



EXPOSURE TO COMPUTER DISCIPLINES

Edited By
Anuj Sharma

ISBN: 978-93-87034-73-0

Printed by

EXCEL BOOKS PRIVATE LIMITED

Regd. Office: E-77, South Ext. Part-I, Delhi-110049

Corporate Office: 1E/14, Jhandewalan Extension, New Delhi-110055



+91-8800697053, +91-011-47520129



info@excelbooks.com/projects@excelbooks.com
internationalalliance@excelbooks.com



www.excelbooks.com



for

Lovely Professional University
Phagwara

CONTENTS

Unit 1:	Data Information	1
	<i>Avinash Bhagat, Lovely Professional University</i>	
Unit 2:	Data Processing	15
	<i>Saurav Prasad, Lovely Professional University</i>	
Unit 3:	Using Operating System	34
	<i>Ajay Kumar Bansal, Lovely Professional University</i>	
Unit 4:	Introduction of Networks	58
	<i>Ajay Kumar Bansal, Lovely Professional University</i>	
Unit 5:	Operations of Network	77
	<i>Sarabjit Kumar, Lovely Professional University</i>	
Unit 6:	Data Communication	102
	<i>Mithilesh Kumar Dubey, Lovely Professional University</i>	
Unit 7:	Graphics and Multimedia	135
	<i>Avinash Bhagat, Lovely Professional University</i>	
Unit 8:	Database System	163
	<i>Mandeep Kaur, Lovely Professional University</i>	
Unit 9:	Software Development	180
	<i>Sarabjit Kumar, Lovely Professional University</i>	
Unit 10:	Programming Language	198
	<i>Balraj Kumar, Lovely Professional University</i>	
Unit 11:	Programming Process	207
	<i>Manmohan Sharma, Lovely Professional University</i>	
Unit 12:	System Development Life Cycle	223
	<i>Pawan Kumar, Lovely Professional University</i>	
Unit 13:	Understanding the Need of Security Measures	233
	<i>Manmohan Sharma, Lovely Professional University</i>	
Unit 14:	Taking Protected Measures	251
	<i>Manmohan Sharma, Lovely Professional University</i>	

SYLLABUS

Exposure to Computer Disciplines

Objectives: This course provides an introduction to all the disciplines provided by computer stream. It concentrates on introducing various sub domains like DBMS, Networking, Programming and Software Development.

S. No.	Description
1	Processing Data: Transforming data into information, How computers represent data, How computers process data, Machine cycles, Memory, Registers, The Bus, Cache Memory
2	Using Operating Systems: Operating system basics, Purpose of the operating system, types of operating system, Providing a user interface, Running Programs, Sharing Information, Managing Hardware, Enhancing an OS with utility software
3	Networks: Sharing data anytime anywhere, Uses of a network, Common types of a network, Hybrid Networks, How networks are structured, Network topologies and Protocols, Network Media, Network Hardware
4	Data Communication: Local and Global reach of the network, Data communication with standard telephone lines and Modems, Using Digital Data Connections, Wireless networks
5	Graphics and Multimedia: Understanding graphics File Formats, Getting Images into your Computer, Graphics Software, Multimedia Basics
6	Data Base Management Systems: The Database, The DBMS, Working with a database, Databases at Work, Common Corporate Database Management Systems
7	Software Programming and Development: What is computer Program, hardware/Software Interaction, Planning a Computer Program, How programs Solve Problems
8	Programming Languages and Programming Process: Categories of Programming Languages, Machine and Assembly Language, Higher Level Languages, WWW development languages, The SDLC of Programming
9	Understanding The Need of Security Measures: Basic Security Concepts, Threats to Users, Threats to Hardware, Threat to Data, Cyber Terrorism
10	Taking Protective Measures: Keeping your System Safe, Protecting Yourself, Protecting your Privacy, Managing Cookies, Spyware and other BUGS, Keeping your data secure, Backing Up data, Safeguarding your hardware

Unit 1: Data Information

Notes

CONTENTS

Objectives

Introduction

1.1 Transforming Data into Information

1.1.1 Functional Units

1.2 Data Representation in Computer

1.2.1 Decimal Representation in Computers

1.2.2 Alphanumeric Representation

1.2.3 Computational Data Representation

1.2.4 Fixed Point Representation

1.2.5 Decimal Fixed Point Representation

1.2.6 Floating Point Representation

1.3 Summary

1.4 Keywords

1.5 Self-Assessment Questions

1.6 Review Questions

1.7 Further Reading

Objectives

After studying this unit, you will be able to:

- Explain data into information
- Discuss data representation in computer

Introduction

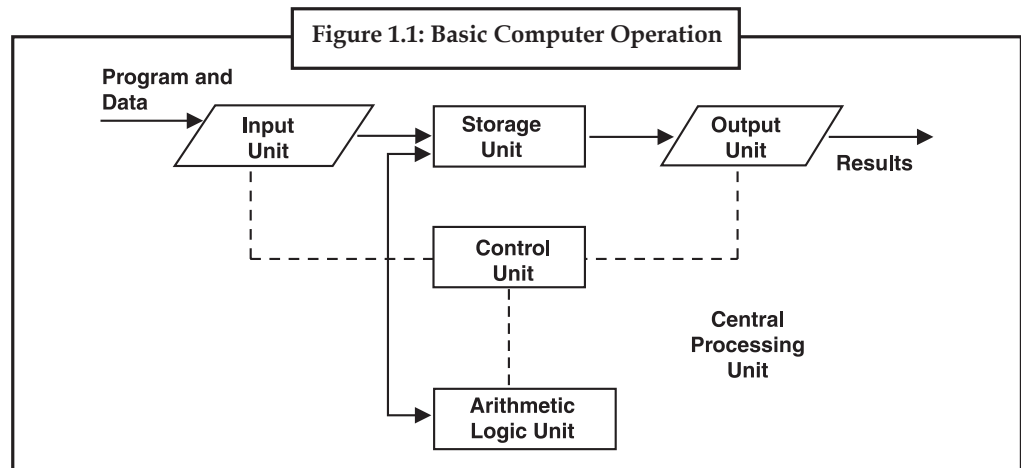
The computer accepts data as an input, stores it process it as the user requires and produces information or processed data as an output in desired format.

The use of Information Technology (IT) is well recognised. The IT has become must for the survival of the business houses with the growing information technology trends. Computer is one of the major components of an Information Technology network and gaining increasing popularity. Today, computer technology has permeated every sphere of existence of modern man. From railway reservations to medical diagnosis; from TV programmes to satellite launching; from matchmaking to criminal catching—everywhere we witness the elegance, sophistication and efficiency possible only with the help of computers.

Basic computer operations are: (i) it accepts data or instruction by way of input, (ii) it stores data, (iii) it can process data as required by the user, (iv) it gives results in the form of output, and (v) it controls all operations inside a computer.

1.1 Transforming Data into Information

This is the **process of producing results** from the data for getting useful information. Similarly the output produced by the computer after processing must also be kept somewhere inside the computer before being given to you in human readable form. Again the output is also stored inside the computer for further processing. Basic computer operation need to be understood first in order to know about data processing. Figure 1.1, shows basic operation units of computer.



Information, thus can be defined as “data that has been transformed into a meaningful and useful form for specific purposes”. In some cases data may not require any processing before constituting information. However, generally, data is not useful unless it is subjected to a process through which it is manipulated and organised, its contents analyzed and evaluated. Only then data becomes information. There is no hard and fast rule for determining when data becomes information. A set of letters and numbers may be meaningful to one person, but may have no meaning to another. Information is identified and defined by its users. For example, when you purchase something in a departmental store, a number of data items are put together, such as your name, address articles you bought, the number of items purchased, the price, the tax and the amount you paid. Separately, these are all data items but if you put these items together, they represent information about a business transaction.

1.1.1 Functional Units

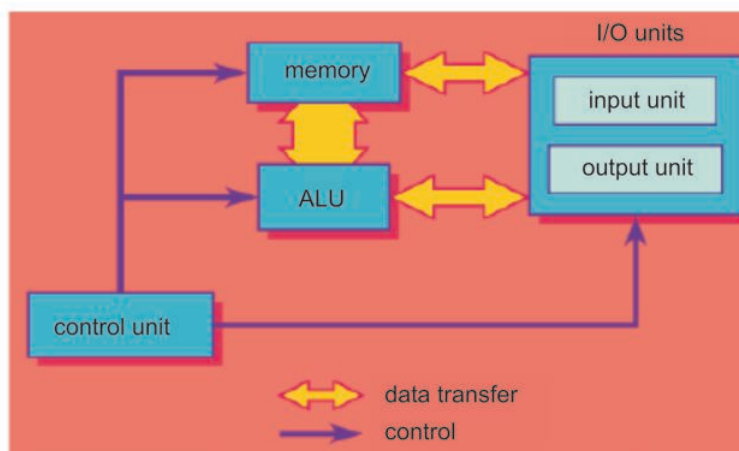
The computer system is divided into three separate units for its operation. They are:

1. Arithmetic logical unit.
2. Control unit.
3. Central processing unit.

1.1.1.1 Arithmetic Logical Unit (ALU)

After you enter data through the input device it is stored in the primary storage unit. The actual **processing of the data and instruction** are performed by Arithmetic Logical Unit. The major operations performed by the ALU are **addition, subtraction, multiplication, division, logic and comparison**. Data is transferred to ALU from storage unit when required. After processing the output is returned back to storage unit for further processing or getting stored.

Figure 1.2: ALU



1.1.1.2 Control Unit (CU)

The next component of computer is the Control Unit, which acts like the **supervisor** seeing that things are done in proper fashion. The control unit determines the sequence in which computer programs and instructions are executed. Things like processing of programs stored in the main memory, interpretation of the instructions and issuing of signals for other units of the computer to execute them.

1.1.1.3 Central Processing Unit (CPU)

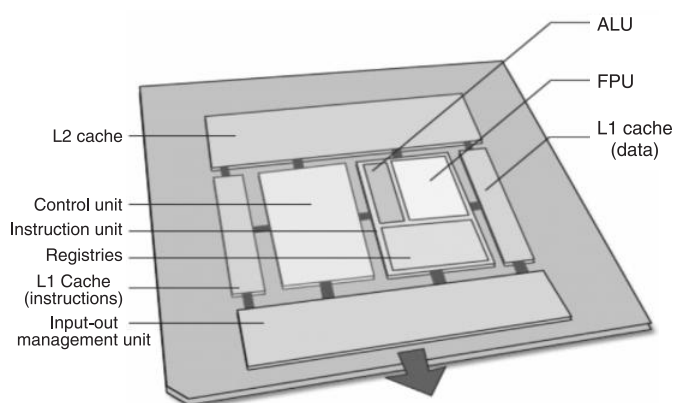
The ALU and the CU of a computer system are jointly known as the central processing unit. You may call CPU as the **brain of any computer system**. It is just like brain that takes all **major decisions**, makes all sorts of calculations and directs different parts of the computer functions by activating and controlling the operations.



Task

Draw a diagram which explains all functional units and their functions of a computer system.

Figure 1.3: CPU



1.2 Data Representation in Computer

Introduction data representation considers how a computer uses numbers to represent data inside the computer. Three types of data are considered at this stage: 1. Numbers including positive, negative and fractions. 2. Text. 3. Graphics.

1.2.1 Decimal Representation in Computers

The binary number system is most natural for computer because of the two stable states of its components. But, unfortunately, this is not a very natural system for us as we work with decimal number system. Then how does the computer do the arithmetic? One of the solutions, which are followed in most of the computers, is to convert all input values to binary. Then the computer performs arithmetic operations and finally converts the results back to the decimal number so that we can interpret it easily. Is there any alternative to this scheme? Yes, there exist an alternative way of performing computation in decimal form but it requires that the decimal numbers should be coded suitably before performing these computations. Normally, the decimal digits are coded in 6-8 bits as alphanumeric characters but for the purpose of arithmetic calculations the decimal digits are treated as four bit binary code. As we know 2 binary bits can represent $2^2 = 4$ different combination, 3 bits can represent $2^3 = 8$ combination and 4 bits can represent $2^4 = 16$ combination. To represent decimal digits into binary form we require 10 combinations only, but we need to have a 4-digit code. One of the common representations is to use first ten binary combinations to represent the ten decimal digits. These are popularly known as Binary Coded Decimals (BCD). The following representation shows the binary coded decimal numbers. Let us represent 43.125 in BCD. It is 0 100 00 1 1.000 1 .00 10 01 0 1

Decimal	Binary Coded Decimal
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	0001 0000
11	0001 0001
12	0001 0010
13	0001 0011
..	
20	0010 0000
..	
30	0011 0000

1.2.2 Alphanumeric Representation

But what about alphabets and special characters like +, - etc.? How do we represent these in computer? A set containing alphabets (in both cases), the decimal digits (10 in number) and special characters (roughly 10-15 in numbers) consist of at least 70-80 elements. One such Code generated for this set and is popularly used is ASCII (American National Standard Code for Information Interchange). This code uses 7 bits to represent 128 characters. Now an extended ASCII is used having 8-bit character representation code on Microcomputers. Similarly binary codes can be formulated for any set of discrete elements e.g., colours, the spectrum, the musical notes, chessboard positions etc. In addition these binary codes are also used to formulate instructions, which are advanced form of data representation.



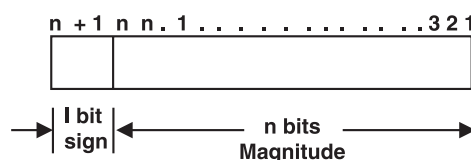
The **binary numeral system**, or **base-2 number system**, represents numeric values using two symbols, 0 and 1. More specifically, the usual base-2 system is a positional notation with a radix of 2.

1.2.3 Computational Data Representation

Till now we have discussed about various number systems and BCD and alphanumeric representation but how do these codes actually use to represent data for scientific calculations? The computer is a discrete digital device and store information in flip-flops, which are two state devices, in binary form. Basic requirements of the computational data representation in binary form are:

- Representation of sign;
- Representation of Magnitude;
- If the number is fractional then binary or decimal point, and exponent.

The solution to sign representation is easy, because sign can be either positive or negative, therefore, one bit can be used to represent sign. By default it should be the left most bit. Thus, a number of n bits can be represented as $n + 1$ bit number, where $n + 1$ bit is the sign bit and rest n bits represent its magnitude.



A $(n + 1)$ bit number: The decimal position can be represented by a position between the flip-flops (storage cells in computer). But, how can one determine this decimal position? Well to simplify the representation aspect two methods were suggested: (i) Fixed point representation where the decimal position is assumed either at the beginning or at the end of a number; and (ii) Floating point representation where a second register is used to keep the value of exponent that determines the position of the binary or decimal point in the number.

But before discussing these two representations let us first discuss the term “complement” of a number. These complements may be used to represent negative numbers in digital computers.

Complement: There are two types of complements for a number of base r , these are called r 's complement and $(r - 1)$'s complement. For example, for decimal numbers the base is 10, therefore, complements will be 10's complement and $(10 - 1) = 9$'s complements. For binary numbers we talk about 2's and 1's complements.

Notes



Example: 1

Find the 9's complement and 10's complement for the decimal number 256.

Solution

9's complement: The 9's complement is obtained by subtracting each digit of the number from 9 (the highest digit value). Similarly, for obtaining 1's complement for a binary number we have to subtract each binary digit of the number from the digit 1 in the same manner.



Example: 2

9's complement of 256

$$\begin{array}{r} 9 \quad 9 \quad 9 \\ -2 \quad -5 \quad -6 \\ \hline = 7 \quad 4 \quad 3 \end{array}$$

Solution

10's complement: adding 1 in the 9's complement produces the 10's complement

$$10's \text{ complement of } 256 = 743 + 1 = 744$$

Please note on adding the number and its 9's complement we get 999 (for this three digit numbers) while on adding the number and its 10's complement we get 1000.



Example: 3

Find 1's and 2's complement of 1010.

Solution

1's complement: The 1's complement of 1010 is

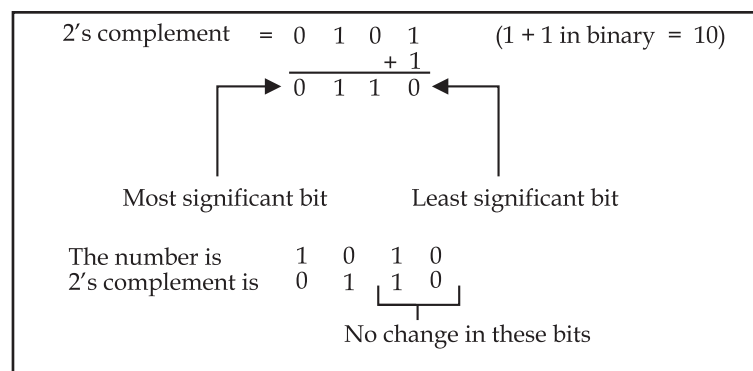
$$\begin{array}{r} 1 \quad 1 \quad 1 \quad 1 \\ -1 \quad -0 \quad -1 \quad -0 \\ \hline 0 \quad 1 \quad 0 \quad 1 \end{array}$$

The number is 1010

The 1's complement is 0101

Please note that wherever you have a digit 1 in number the complement contains 0 for that digit and vice versa. In other words to obtain 1's complement of a binary number, we only have to change all the 1's of the number to 0 and all the zeros to 1's. This can be done by complementing each bit.

2's complement: adding 1 in 1's complement will generate the 2's complement



Notes

The 2's complement can also be obtained by not complementing the least significant zeros till the first 1 is encountered. This 1 is also not complemented. After this 1 rest all the bits are complemented on the left.

Therefore, 2's complement of the following number (using this method) should be (you can check it by finding 2's complement as we have done in the example).

Number	2's complement
100100	011100
	 No change in these bits
100000	100000
101001	010111
	 No change in these bits



Task

Find the 2's complement of 11101 and 1000000011.

1.2.4 Fixed Point Representation

The fixed-point numbers in 1 binary uses a sign bit. A positive number has a sign bit 0, while the negative number has a sign bit 1. In the fixed-point numbers we assume that the position of the binary point is at the end. It implies that all the represented numbers should be integers. A negative number can be represented in one of the following ways:

Signed magnitude representation,

Signed 1's complement representation, or Signed 2's complement representation.

(Assumption size of register = 7 bit, 8th bit is used for error checking and correction or other purposes).

Signed magnitude representation

	0 ↑ Sign bit +6 000110	1 ↑ Sign bit -6 000110
		(No change in the magnitude only sign bit changes)
signed 1's complement	0 ↑ Sign bit 000110	1 ↑ Sign bit 111001
		(complement all the bits including sign bit of the positive number to obtain its complement negative number)
signed 2's complement	0 ↑ Sign bit 000110	111010
		(2's complement of the positive number including sign bit)

Notes

Arithmetic Addition: The complexity of arithmetic addition is dependent on the representation, which have been followed. Let us discuss this with the help of following example.



Example: Add 25 and - 30 in binary using 7 bit register in

Solution signed magnitude representation.
signed 1's complement
signed 2's complement.

Solution: 25 or +25 is

0 01 1001

- 30 in signed magnitude representation is:

+30 is 0 01 1 1 10,

therefore - 30 is 1 01 11 10

To do the arithmetic addition with one negative number we have to check the magnitude of the numbers. The number having smaller magnitude is then subtracted from the bigger number and the sign of bigger number is selected. The implementation of such a scheme in digital hardware will require a long sequence of control decisions as well as circuits that will add, compare and subtract numbers. Is there a better alternative than this scheme? Let us first try the signed 2's complement.

-30 in signed 2's complement notation will be:

+30 is 0011110

-30 is 1100010 (2' complement of 30 including sign bit)

+25 is 0011001

-25 is 1100111

Addition

+25	0 011 001	
<u>+30</u>	<u>0 011 110</u>	+5 is 0 000 101
<u>+55</u>	<u>0 110 111</u>	-5 is 1 111 011

in 2's complement representation.

+25	0 011 001	(Just add the two numbers)
<u>-30</u>	<u>1 100 110</u>	
<u>-05</u>	<u>1 111 011</u>	

-25	1 100 001	
<u>+30</u>	<u>0 011 110</u>	(Just add the two numbers)
<u>+05</u>	<u>0 000 101</u>	

Discard the carry out of the sign bit.

-25	1 100 111	+55 is 0 110 111 in
<u>-30</u>	<u>1 100 010</u>	2's complement representation.
<u>-55</u>	<u>1 001 001</u>	Therefore in the 2's complement notation -55 is

Discard the carry out of the sign bit.

Notes

Please note how easy is to add two numbers using signed 2's Complement. This procedure requires only one control decision in one circuit for adding the two numbers. But it put additional condition that the negative numbers should be stored in signed 2's complement form in the registers. This can be achieved by complementing the positive number bit by bit and then incrementing the resultant by 1 to get signed 2's complement.

Signed 1's Complement Representation: Another possibility, which also is simple, is use of signed 1's complement. Signed 1's complement has a rule. Add the two numbers, including the sign bit. If carry of the most significant bit or sign bit is one, then increment the result by 1 and discard the carry over. Let us repeat all the operations with 1's complement.

+25	0	011	001	(-25)	1	100	110
+30	0	011	110	-30	1	100	001
+25	0	011	001	(-25)	1	100	110
+30	0	011	110	+30	0	011	110
+55	0	110	111	+5	1	0	000 100

↑
Carry out is 1 so add 1 to the sum,
and discard the carry over.
∴ Sum = 0 000 101 which is number 5.

+25	0	011	001
-30	1	100	001
-5	1	111	010
+5 is		0	000 101
-5 in 1's complement	1	111	010
-25		1	100 110
-30		1	100 001
-55	1	1	000 111

↑
Carry out

Since, the carry out is 1, so add 1 to sum and discard the carry

1 000 111

1

1 001 000

+55 is 0 110 111

-55 is 1's complement 1 001 000

Another interesting feature about these representations is the representation of 0. In signed magnitude and 1's complement there are two representations for zero as:

Signed magnitude	+ 0	-0
	0 000000	1 000000
Signed 1's complement	0 000000	1 111111

But in signed 2's complement there is just one zero and there is no positive or negative zero.

+0	000000	-0	2's complement of
+0 =	1	111111	
	1	0	000000
	↑		discard the carry.

Notes

Thus, both +0 and -0 are same in 2's complement notation. This is an added advantage in favour of 2's complement notation. The highest number, which can be accommodated in a register, also depend on the type of representation. In general, in a 8 bit register 1 bit is used as sign, therefore, rest 7 bits can be used for representing the value. The highest and the lowest number, which can be represented, are:

For signed magnitude representation $2^j - 1$ to $-(2^j - 1)$

$$= 128 - 1 \text{ to } -(128 - 1)$$

$$= 127 \text{ to } -127$$

For signed 1's complement 127 to -127

But, for signed 2's complement we can represent +127 to -128. The -128 is represented in signed 2's complement notation as 10000000.

Arithmetic Subtraction: The subtraction can be easily done using the 2's complement by taking the 2's complement of the subtrahend (inclusive of sign bit) and then adding the two numbers. Signed 2's complement provide very simple way for adding and subtracting two numbers. thus, finally computer (including IBM PC) adopt signed 2's complement notation. The reason why signed 2's complement is preferred over signed 1's complement is because it has only one representation for zero.

Overflow: An overflow is said to have occurred when the sum of two n digits number occupies $n + 1$ digits. This definition is valid for both binary as decimal digits. But what is the significance of overflow for binary numbers since it is not a problem for the cases when we add two numbers? Well the answer is in the limits of representation of numbers. Every computer employs a limit for representing number e.g., in our examples we are using 8 bit registers for calculating the sum. But what will happen if the sum of the two numbers can be accommodated in 9 bits? Where are we going to store the 9th bit? The problem will be clearer by the following example.



Example:

Add the numbers 65 and 75 in 8 bit register in signed 2's complement notation.

65	0	1000001
75	0	1001011
140	1	0001100

The expected result is +140 but the binary sum is a negative number and is equal to -16, which obviously is a wrong result. This has occurred because of overflow.

How does the computer know that overflow has occurred?

If the carry into the sign bit is not equal to the carry out of the sign bit then overflow must have occurred.

For example,

-65	1	0111111	-65	1	0111111		
-15	1	1110001	-75	1	0110101		
-80	1	1	0110000	-140	1	0	1110100
carry into sign bit = 1 carry out of sign bit = 1 No overflow			carry into sign bit = 0 carry out of sign bit = 1 Therefore, overflow				

Thus, overflow has occurred, i.e. the arithmetic results so calculated have exceeded the capacity of the representation. This overflow also implies that the calculated results might be erroneous.



Caution

Arithmetic addition	Arithmetic Subtraction
Arithmetic addition: Complexity of arithmetic addition is dependent on the representation	Arithmetic Subtraction: The subtraction can be easily done using the 2's complement by taking the 2's complement of the subtrahend (inclusive of sign bit) and then adding the two numbers.

Notes

1.2.5 Decimal Fixed Point Representation

A decimal digit is represented as a combination of four bits; thus, a four digit decimal number will require 16 bits for decimal digits representation and additional 1 bit for sign. Normally to keep the convention of one decimal digit to 4 bits, the sign sometimes is also assigned a 4-bit code. This code can be the bit combination which has not been used to represent decimal digit e.g., 11 00 may represent plus and 1 101 can represent minus.

Although this scheme wastes considerable amount of storage space yet it do not require conversion of a decimal number to binary. Thus, can be used at places where the amount of computer arithmetic is less than the amount of input output of data e.g. calculators or business data processing. The arithmetic in decimal can also be performed as in binary except that instead of signed one's complement, signed nine's complement is used and instead of signed 2's complement signed 10's complement is used.

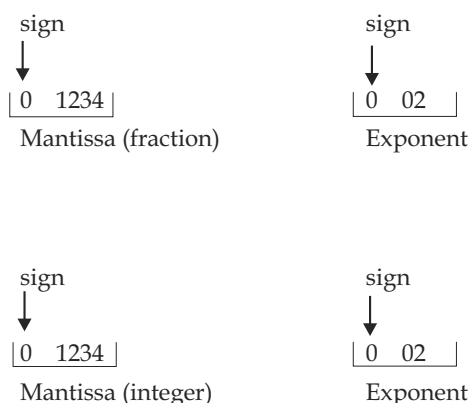
1.2.6 Floating Point Representation

Floating-point number representation consists of two parts. The first part of the number is a signed fixed point number, which is termed as mantissa, and the second part specifies the decimal or binary point position and is termed as an Exponent. The mantissa can be an integer or a fraction. Please note that the position of decimal or binary point is assumed and it is not a physical point, therefore, wherever we are representing a point it is only the assumed position.



Example: A decimal +12.34 in a typical floating point notation is:

Solution



Notes

This number in any of the above form (if represented in BCD) requires 17 bits for mantissa (1 for sign and 4 each decimal digit as BCD) and 9 bits for exponent (1 for sign and 4 for each decimal digit as BCD). Please note that the exponent indicates the correct decimal location. In the first case where exponent is +2, indicates that actual position of the decimal point is two places to the right of the assumed position, while exponent - 2 indicates that the assumed position of the point is two places towards the left of assumed position. The assumption of the position of point is normally the same in a computer resulting in a consistent computational environment.

Floating-point numbers are often represented in normalised forms. A floating point number whose mantissa does not contain zero as the, most significant digit of the number is considered to be in normalised form. For example, a BCD mantissa + 370 which is 0 001 1 01 1 1 0000 is in normalized form because these leading zero's are not part of a zero digit. On the other hand a binary number 001 100 is not in a normalised form. The normalised form of this number is:

0 1100

(sign)

A floating binary number +1010.001 in a 16-bit register can be represented in normalised form (assuming 10 bits for mantissa and 6 bits for exponent):

0 101000100
 ↑
 | sign |
 Mantissa (fraction)

0 00100
 ↑
 | sign |
 Exponent



Did u know?

A zero cannot be normalised as all the digits in mantissa in this case has to be zero. Arithmetic operations involved with floating point numbers are more complex in nature, take longer time for execution and require complex hardware. Yet the floating-point representation is must as it is useful in scientific calculations. Real numbers are normally represented as floating point numbers.



Case Study

Dr. A.P.J Abdul Kalam President of India: May 28, 2003 Indian President calls for greater use of open source software "In India, open source code software will have to come and stay in a big way for the benefit of our billion people" The President of India has called for the extensive of open-source software to replace costly proprietary information technology programmes. In a speech at the country's International Institute of Information Technology, President Abdul Kalam, expressed his concern that so many sectors, including government and education, were still dependent on costly proprietary software packages, calling it a "most unfortunate thing." "In India, open sourcecode software will have to come and stay in a big way for the benefit of our billion people," he added.

The Indian President also reminded the IT industry of the importance of Indian language computing solutions: "We must have (Indian) search engines, word processing tools, optical character recognisers, speech recognisers and machine translators." The Indian President

Contd...

Notes

speaks from experience. He is South Asia's first "techie" national leader, with an impressive scientific background. As Chairman, Technology Information, Forecasting and Assessment Council (TIFAC), President Kalam generated the Technology Vision 2020 documents—a road map for transforming India from Developing India to Developed India. He provided overall guidance to a number of Homegrown Technology Projects and major technology missions. He served as the Principal Scientific Advisor to the Government of India, in the rank of Cabinet Minister, from November 1999 to November 2001. He was primarily responsible for evolving policies, strategies and missions for generation of innovations and support systems for multiple applications.

Questions:

1. How technology for disabled can enhance them?
2. Why is computer education necessary for developing India?

1.3 Summary

- The five basic operations that a computer performs. These are input, storage, processing, output and control.
- A computer accepts data as input, stores it, processes it as the user requires and provides the output in a desired format.
- Computer system into three Functional units, i.e. Arithmetic logic unit (ALU), Control unit (CU), Central processing unit.
- The Binary Numeral system represent numeric values using two digit 081.
- Flooding point number representation consists of two parts. The first part of the number is formed as mantissa and second part specifies as an exponent.

1.4 Keywords

Arithmetic Logical Unit (ALU): The actual **processing of the data and instruction** are performed by Arithmetic Logical Unit. The major operations performed by the ALU are **addition, subtraction, multiplication, division, logic and comparison**.

ASCII: (American National Standard Code for Information Interchange). This code uses 7 bits to represent 128 characters. Now an extended ASCII is used having 8-bit character representation code on Microcomputers.

Data Transformation: This is the **process of producing results** from the data for getting useful information. Similarly the output produced by the computer after processing must also be kept somewhere inside the computer before being given to you in human readable form.

Decimal Fixed Point Representation: A decimal digit is represented as a combination of four bits; thus, a four digit decimal number will require 16 bits for decimal digits representation and additional 1 bit for sign.

Fixed Point Representation: The fixed-point numbers bit 1 binary uses a sign bit. A positive number have a sign bit 0, while the negative number has a sign bit 1. In the fixed-point numbers we assume that the position of the binary point is at the end.

Floating Point Representation: Floating-point number representation consists of two parts. The first part of the number is a signed fixed point number, which is termed as mantissa, and the second part specifies the decimal or binary point position and is termed as an Exponent.

Notes



Lab Exercise

Draw basic computer operation.

1.5 Self-Assessment Questions

1. Floating-point number representation consists of two parts. The first part of the number is a signed fixed point number, which is termed as mantissa and the second part specifies the decimal or binary point position and is termed as an Exponent.
(a) True (b) False
2. Primary memory is an integral part of the computer system and is accessible directly by the processing unit.
(a) True (b) False
3. In computer architecture, a **processor register** (or **general purpose register**) is a small amount of storage available on the CPU whose contents can be accessed more quickly than storage available elsewhere.
(a) True (b) False

1.6 Review Questions

1. Differentiate between the following :
(a) Data and Information
(b) Data processing and Data processing system
2. Define the terms data, data processing and information.
3. Identify various activities involved in manipulation.
4. Draw a block diagram to illustrate the basic organization of computer system and explain the function of various units.
5. Explain Data Processing System.
6. Explain Fixed Point Representation.
7. Explain Decimal Fixed Point Representation.
8. Explain Floating Point Representation.

Answers for Self-Assessment Questions

1. (a) 2. (b) 3. (b)

1.7 Further Reading



Books

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition.



Online link

http://www.webopedia.com/TERM/D/data_process

Unit 2: Data Processing

Notes

CONTENTS

Objectives

- 2.1 Method of Processing Data
 - 2.1.1 The Data Processing Cycle
 - 2.1.2 Data Processing System
- 2.2 Machine Cycles
- 2.3 Memory
 - 2.3.1 Primary Memory
 - 2.3.2 Secondary Storage
- 2.4 Registers
 - 2.4.1 Categories of Registers
 - 2.4.2 Register Usage
- 2.5 Computer Bus
 - 2.5.1 Data Bus
 - 2.5.2 Address Bus
 - 2.5.3 Control Bus
 - 2.5.4 Expansion Bus
- 2.6 Cache Memory
 - 2.6.1 Operation
 - 2.6.2 Applications
 - 2.6.3 The Difference Between Buffer and Cache
- 2.7 Summary
- 2.8 Keywords
- 2.9 Self-Assessment Questions
- 2.10 Review Questions
- 2.11 Further Reading

Notes

Objectives

After studying this unit, you will be able to:

- Explain data processing cycle.
- Discuss data processing system.
- Explain machine cycle.
- Explain different types of memory.
- Discuss different registers.
- Explain computer bus.
- Discuss cache memory in detail.

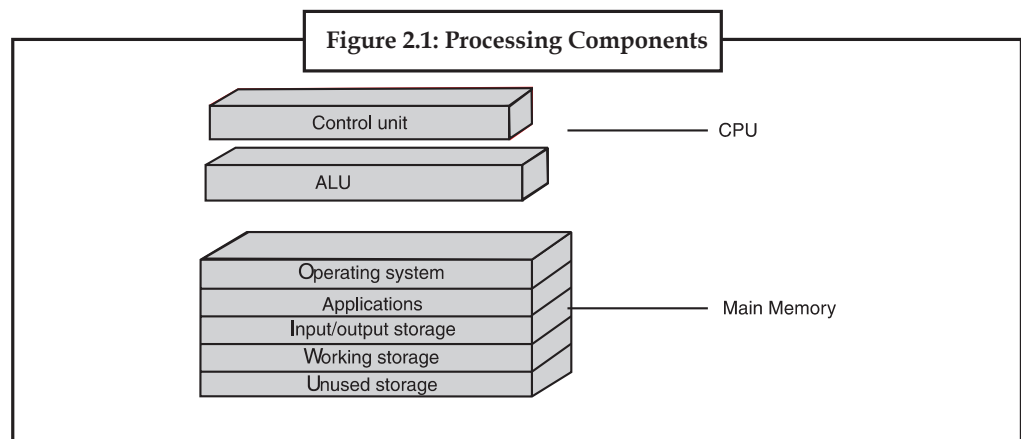
2.1 Method of Processing Data

Data processing consists of those activities which are necessary to transform data into information.

Man has in course of time devised certain tools to help him in processing data. These include manual tools such as pencil and paper, mechanical tools such as filing cabinets, electromechanical tools such as adding machines and typewriters, and electronic tools such as calculators and computers. Many people immediately associate data processing with computers.

Processing is the **thinking** that the computer does - the calculations, comparisons, and decisions.

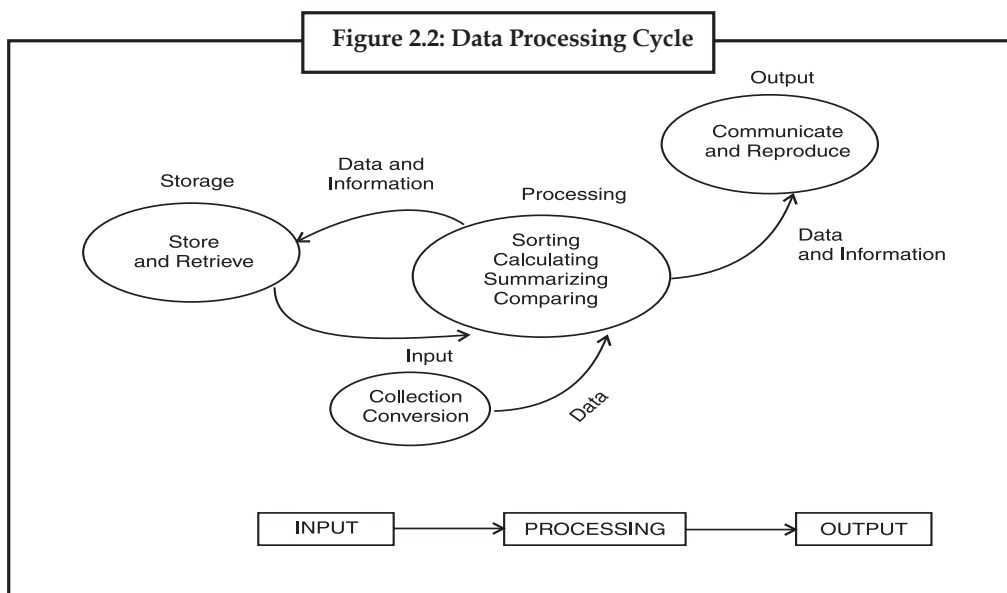
People also process data. What you see and hear and touch and feel is input. Then you connect this new input with what you already know, look for how it all fits together, and come up with a reaction, your output. "That stove is hot. I'll move my hand now!"



2.1.1 The Data Processing Cycle

The data processing activities described above are common to all data processing systems from manual to electronic systems. These activities can be grouped in four functional categories, viz., data input, data processing, data output and storage, constituting what is known as a data processing cycle.

(i) Input: The term input refers to the activities required to record data and to make it available for processing. The input can also include the steps necessary to check, verify and validate data contents.



(ii) Processing: The term processing denotes the actual data manipulation techniques such as classifying, sorting, calculating, summarizing, comparing, etc. that convert data into information.

(iii) Output: It is a communication function which transmits the information, generated after processing of data, to persons who need the information. Sometimes output also includes decoding activity which converts the electronically generated information into human-readable form.

(iv) Storage: It involves the filing of data and information for future use. The above mentioned four basic functions are performed in a logical sequence data processing systems.



Task

Explain steps involved in data processing cycle.

2.1.2 Data Processing System

The activity of data processing system can be viewed as a “system”.

Control unit (CU):

- The control unit determines the sequence in which computer programs and instructions are execute.

ALU

- ALU stands for Arithmetic/Logic Unit
- This is the part that executes the computer's commands.

A command must be either a basic arithmetic operation:

+ - * /

or one of the logical comparisons: > < = not =

“Everything has to be broken down into these few operations.

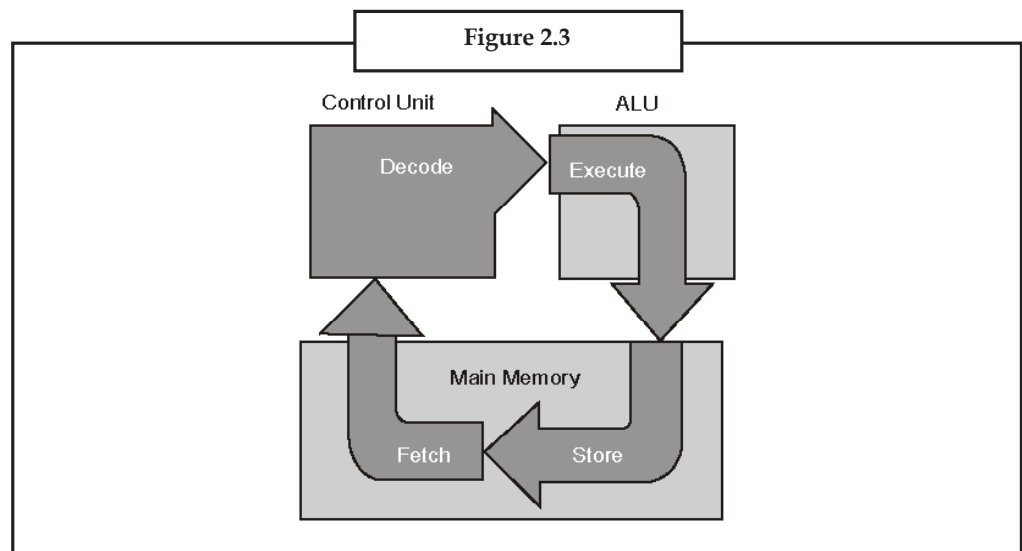
“The ALU can only do one thing at a time but can work very very fast.



Notes

Main Memory

- Main Memory stores the commands that the CPU executes and the results.
- This is where the computer stores the data and commands that are currently being used.
- When the computer is turned off, all data in Main Memory vanishes.
- A data storage method of this type is called volatile since the data "evaporates."
- The CPU can fetch one piece of data in one machine cycle.



2.2 Machine Cycles

A machine cycle, also called a processor cycle or a instruction cycle, is the basic operation performed by a central processing unit (CPU). A CPU is the main logic unit of a computer. A machine cycle consists of a sequence of three steps that is performed continuously and at a rate of millions per second while a computer is in operation. They are fetch, decode and execute. There also is a fourth step, store, in which input and output from the other three phases is stored in memory for later use; however, no actual processing is performed during this step. In the fetch step, the control unit requests that main memory provide it with the instruction that is stored at the address (i.e., location in memory) indicated by the control unit's program counter. The control unit is a part of the CPU that also decodes the instruction in the *instruction register*. A *register* is a very small amount of very fast memory that is built into the CPU in order to speed up its operations by providing quick access to commonly used values; instruction registers are registers that hold the instruction being executed by the CPU. Decoding the instructions in the instruction register involves breaking the *operand field* into its components based on the instructions *opcode*.



Did u know?

Opcode (an abbreviation of *operation code*) is the portion of a *machine language* instruction that specifies what operation is to be performed by the CPU. Machine language, also called *machine code*, refers to instructions coded in patterns of bits (i.e., zeros and ones) that are directly readable and executable by a CPU.

A program counter, also called the *instruction pointer* in some computers, is a register that indicates where the computer is in its instruction sequence. It holds either the address of the instruction currently being executed or the address of the next instruction to be executed,

depending on the details of the particular computer. The program counter is automatically incremented for each machine cycle so that instructions are normally retrieved sequentially from memory.

The control unit places these instructions into its instruction register and then increments the program counter so that it contains the address of the next instruction stored in memory. It then executes the instruction by activating the appropriate circuitry to perform the requested task. As soon as the instruction has been executed, it restarts the machine cycle, beginning with the fetch step.

During one machine cycle the processor executes at least two steps, fetch (data) and execute (command). The more complex a command is (more data to fetch), the more cycles it will take to execute. Reading data from the zero page typically needs one cycle less as reading from an absolute address. Depending on the command, one or two more cycles will eventually be needed to modify the values and write them to the given address.

Control Unit is the part of the computer that controls the **Machine Cycle**.

It takes numerous cycles to do even a simple addition of two numbers.

Fetch - get an instruction from Main Memory

Decode - translate it into computer commands

Execute - actually process the command

Store - write the result to Main Memory

It has to be considered how long a specific command takes to execute when coding, so that the whole timing inside the program altogether and specially the screen output works. This is referred to as "cycle counting". Beginners in assembler programming often throw in NOP commands (takes two cycles) to achieve a stable timing.

2.3 Memory

There are two kinds of computer memory: primary and secondary. Primary memory is an integral part of the computer system and is accessible directly by the processing unit. RAM is an example of primary memory. As soon as the computer is switched off the contents of the primary memory is lost. The primary memory is much faster in speed than the secondary memory. Secondary memory such as floppy disks, magnetic disk, etc., is located external to the computer. Primary memory is more expensive than secondary memory. Because of this, the size of primary memory is less than that of secondary memory. Computer memory is used to store two things: (i) instructions to execute a program and (ii) data.

When the computer is doing any job, the data that have to be processed are stored in the primary memory. This data may come from an input device like keyboard or from a secondary storage device like a floppy disk. As program or the set of instructions is kept in primary memory, the computer is able to follow instantly the set of instructions.

2.3.1 Primary Memory

The primary memory in the computer is in the form of IC's (Integrated Circuits). These circuits are called Random Access Memory (RAM). Each of RAM's locations stores one byte of information. (One byte is equal to 8 bits). A bit is an acronym for binary digit, which stands for one binary piece of information. This primary or internal storage section is made up of several small storage locations (ICs) called cells. Each of these cells can store a fixed number of bits called word length. Each cell has a unique number assigned to it called the

Notes

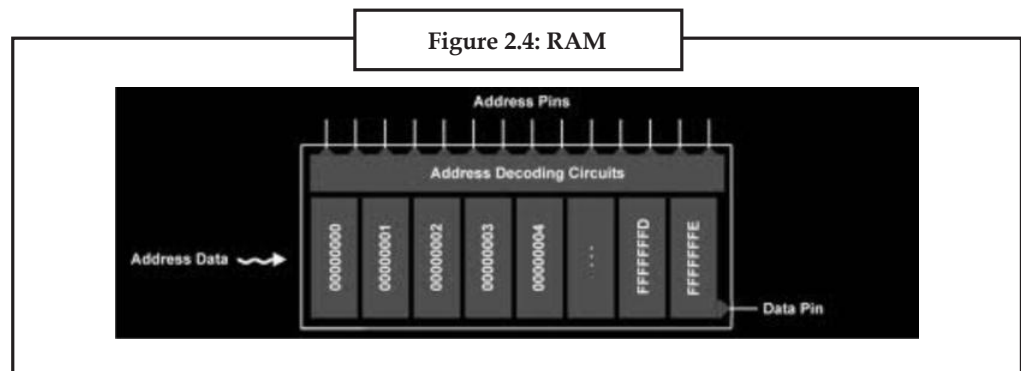
address of the cell and it is used to identify the cells. The address starts at 0 and goes up to (N-1). You should know that the memory is like a large cabinet containing as many drawers as there are addresses on memory. Each drawer contains a word and the address is written on outside of the drawer.

(a) Capacity of Primary Memory You know that each cell of memory contains one character or one byte of data. So the memory capacity is defined in terms of byte or words. The relation is: 1 kilobyte (KB) = 1024 bytes. Thus 64 kilobyte (KB) memory is capable of storing $64 \times 1024 = 32,768$ bytes. The memory size ranges from few kilobytes in small systems to several thousand kilobytes in large mainframe and super computers. In your personal computer you will find memory capacity in the range of 32 MB, 64 MB and even 128 MB (MB = Million bytes and 1 MB = 1024 KB).

The following terms related to memory of a computer are discussed below:

2.3.1.1 Random Access Memory (RAM)

The primary storage is referred to as random access memory (RAM) because it is possible to randomly select and use any location of the memory directly for storing and retrieving data. It takes same time to reach any address of the memory whether it is in the beginning or in the last. It is also called read/write memory. The storage of data and instructions inside the primary storage is temporary. It disappears from RAM as soon as the power to the computer is switched off. The memory, which lose its contents on failure of power supply, are known as volatile memories. So now we can say that RAM is volatile memory.



2.3.1.2 Read Only Memory (ROM)

There is another memory in computer, which is called Read Only Memory (ROM). Again it is the ICs inside the PC that form the ROM. The storage of program in the ROM is permanent. The ROM stores some standard processing programs supplied by the manufacturer to operate the personal computer. The ROM can only be read by the CPU but it cannot be changed. The basic input/output program, which is required to start and initialize equipment attached to the PC, is stored in the ROM. The memories, which do not lose their contents on failure of power supply, are known as non-volatile memories. ROM is a non-volatile memory.



Task

Give the difference between RAM and ROM.

Figure 2.5: ROM



2.3.1.3 Programmable Read Only Memory (PROM)

There is another type of primary memory in computer, which is called Programmable Read Only Memory (PROM). You know that it is not possible to modify or erase programs stored in ROM, but it is possible for you to store your program in PROM chip. Once the programs are written it cannot be changed and remain intact even if power is switched off. Therefore, programs or instructions written in PROM or ROM cannot be erased or changed.

2.3.1.4 Erasable Programmable Read Only Memory (EPROM)

This stands for Erasable Programmable Read Only Memory, which overcomes the limitations of PROM & ROM. EPROM chip can be programmed time and again by erasing the information stored earlier in it. Exposing the chip for some time to ultraviolet light erases information stored in EPROM. The chip can be reprogrammed using a special programming facility. When the EPROM is in use, information can only be read.

2.3.1.5 Cache Memory

The speed of CPU is extremely high compared to the access time of main memory. Therefore, the performance of CPU decreases due to the slow speed of main memory. To minimize the mismatch in operating speed, a small memory chip is attached between CPU and Main memory whose access time is very close to the processing speed of CPU. It is called CACHE memory. CACHE memories are accessed much faster than conventional RAM. It is used to store programs or data currently being executed or temporary data frequently used by the CPU. So cache memory makes main memory to work faster and larger than it really is. It is also very expensive to have bigger size of cache memory and therefore it is available in limited capacity generally Kilo Bytes.

2.3.1.6 Registers

The CPU processes data and instructions with high speed. There is also movement of data between various units of computer. It is necessary to transfer the processed data with high speed. So the computer uses a number of special memory units called registers. These are not part of the main memory but they store data or information temporarily and pass it on as directed by the control unit.

2.3.2 Secondary Storage

Secondary storage (also known as external memory or auxiliary storage), differs from primary storage in that it is not directly accessible by the CPU. The computer usually uses its input/output

Notes

channels to access secondary storage and transfers the desired data using intermediate area in primary storage. Secondary storage does not lose the data when the device is powered down-it is non-volatile. Per unit, it is typically also two orders of magnitude less expensive than primary storage. Consequently, modern computer systems typically have two orders of magnitude more secondary storage than primary storage and data is kept for a longer time there.

In modern computers, hard disk drive are usually used as secondary storage. The time taken to access a given byte of information stored on a hard disk is typically a few thousandths of a second, or milliseconds.

2.3.2.1 Magnetic Tape

Magnetic tapes are used for large computers like mainframe computers where large volume of data is stored for a longer time. In PC also you can use tapes in the form of cassettes. The storage of data in tapes is inexpensive. Tapes consist of magnetic materials that store data permanently. It can be 12.5 mm to 25 mm wide plastic film-type and 500 meter to 1200 meter long, which is coated with magnetic material. The tape unit is connected to the central processor and information is fed into or read from the tape through the processor. It is similar to a cassette tape recorder.

Figure 2.6: Magnetic Tapes



Advantages of Magnetic Tape

Compact: A 10-inch diameter reel of tape is 2400 feet long and is able to hold 800, 1600 or 6250 characters in each inch of its length. The maximum capacity of such tape is 180 million characters. Thus data are stored much more compactly on tape.

Economical: The cost of storing data is very less as compared to other storage devices.

Fast: Copying of data is easier and fast.

Long term Storage and Re-usability: Magnetic tapes can be used for long term storage and a tape can be used repeatedly with out loss of data.

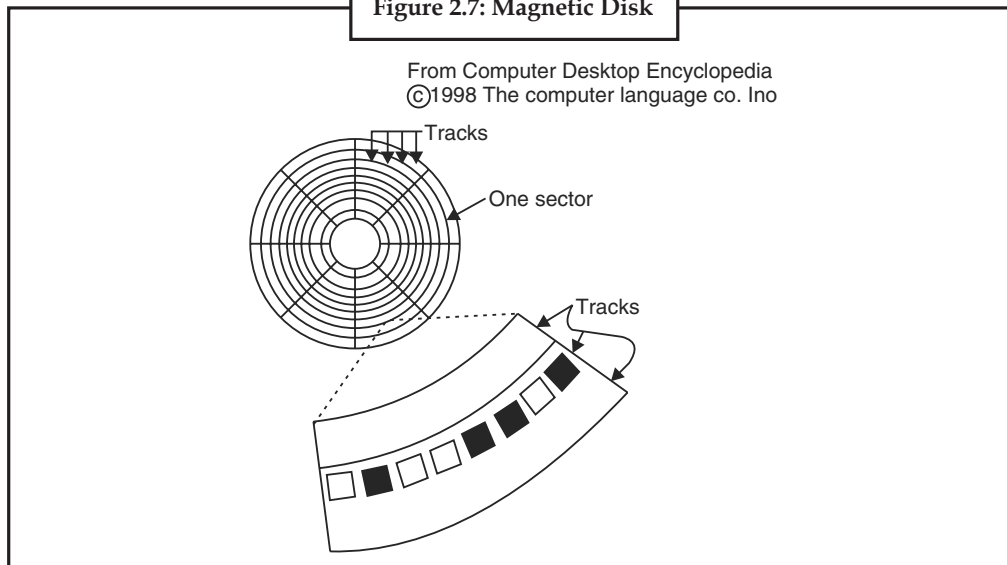
2.3.2.2 Magnetic Disk

You might have seen the gramophone record, which is like a circular disk and coated with magnetic material. Magnetic disks used in computer are made on the same principle.

Notes

It rotates with very high speed inside the computer drive. Data is stored on both the surfaces of the disk. Magnetic disks are most popular as direct access storage device. Each disk consists of a number of invisible concentric circles called tracks. Information is recorded on tracks of a disk surface in the form of tiny magnetic spots. The presence of a magnetic spot represents one bit and its absence represents zero bit. The information stored in a disk can be read many times without affecting the stored data. So the reading operation is non-destructive. But if you want to write a new data, then the existing data is erased from the disk and new data is recorded.

Figure 2.7: Magnetic Disk



2.3.2.3 Floppy Disk

It is similar to magnetic disk discussed above. It is 3.5 inch in diameter. These come in single or double density and recorded on one or both surface of the diskette. The capacity of a high-density 3.5 inch floppy it is 1.44 mega bytes. It is cheaper than any other storage devices and is portable. The floppy is a low cost device particularly suitable for personal computer system.

Figure 2.8: Floppy Disk



Notes

2.3.2.4 Optical Disk

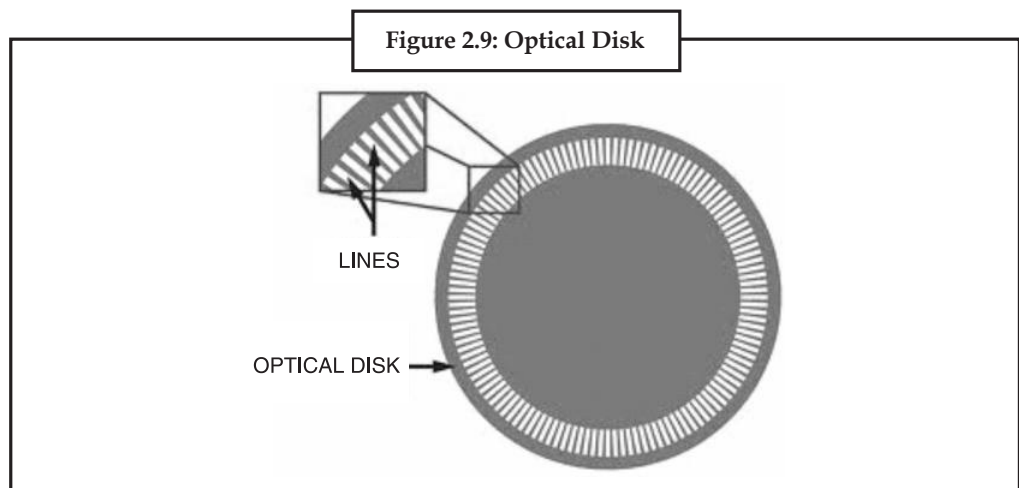
With every new application and software there is greater demand for memory capacity. It is the necessity to store large volume of data that has led to the development of optical disk storage medium. Optical disks read and write the data using light and not the magnetization as in above storage devices. Optical disks can be divided into the following categories:

1. **Compact Disk/Read Only Memory (CD-ROM):** CD-ROM disks are made of reflective metals. CD-ROM is written during the process of manufacturing by high power laser beam.

Here the storage density is very high, storage cost is very low and access time is relatively fast. Each disk is approximately 4 ½ inches in diameter and can have over 600 MB of data. As the CD-ROM can be read only we cannot write or make changes into the data contained in it.

2. **Write Once, Read Many (WORM):** The inconvenience that we cannot write anything onto a CD-ROM is avoided in WORM. A WORM allows the user to write data permanently on to the disk. Once the data is written it can never be erased without physically damaging the disk. Here data can be recorded from keyboard, video scanner, OCR equipment and other devices. The advantage of WORM is that it can store vast amount of data amounting to gigabytes (10⁹ bytes). Any document in a WORM can be accessed very fast, say less than 30 seconds.

3. **Erasable Optical Disk:** These are optical disks where data can be written, erased and re-written. This makes use of a laser beam to write and re-write the data. These disks may be used as alternatives to traditional disks. Erasable optical disks are based on a technology known as magneto-optical (MO). To write a data bit on to the erasable optical disk the MO drive's laser beam heats a tiny, precisely defined point on the disk's surface and magnetises it.



<i>Storage</i>	<i>Speed</i>	<i>Capacity</i>	<i>Relative Cost (\$)</i>	<i>Permanent?</i>
Registers	Fastest	Lowest	Highest	No
RAM	Very Fast	Low/Moderate	High	No
Floppy Disk	Very Slow	Low	Low	Yes
Hard Disk	Moderate	Very High	Very Low	Yes

2.4 Registers

In computer architecture, a **processor register** (or **general purpose register**) is a small amount of storage available on the CPU whose contents can be accessed more quickly than storage available elsewhere. Typically, this specialized storage is not considered part of the normal memory range for the machine. Most, but not all, modern computers adopt the so-called load-store architecture. Under this paradigm, data is loaded from some larger memory be it cache or RAM into registers, manipulated or tested in some way (using machine instructions for arithmetic/logic/comparison) and then stored back into memory, possibly at some different location. A common property of computer programs is locality of reference: the same values are often accessed repeatedly; and holding these frequently used values in registers improves program execution performance.



Did u know?

Processor registers are at the top of the memory hierarchy, and provide the fastest way for a CPU to access data. The term is often used to refer only to the group of registers that are directly encoded as part of an instruction, as defined by the instruction set.

Allocating frequently used variables to registers can be critical to a program's performance. This action (register allocation) is performed by a compiler in the code generation phase.

2.4.1 Categories of Registers

Registers are normally measured by the number of bits they can hold, for example, an "8-bit register" or a "32-bit register". A processor often contains several kinds of registers, that can be classified accordingly to their content or instructions that operate on them:

- (a) **User-accessible registers:** The most common division of user-accessible registers is into data registers and address registers.
- (b) **Data registers:** These are used to hold numeric values such as integer and floating-point values. In some older and low end CPUs, a special data register, known as the accumulator, is used implicitly for many operations.
- (c) **Address registers:** Its hold addresses and are used by instructions that indirectly access memory.
 - Some processors contain registers that may only be used to hold an address or only to hold numeric values (in some cases used as an index register whose value is added as an offset from some address); others allow registers to hold either kind of quantity. A wide variety of possible addressing modes, used to specify the effective address of an operand, exist.
 - A stack pointer, sometimes called a stack register, is the name given to a register that can be used by some instructions to maintain a stack.
- (d) **Conditional registers:** Its hold truth values often used to determine whether some instruction should or should not be executed.
- (e) **General purpose registers (GPRs):** Its can store both data and addresses, i.e., they are combined Data/Address registers.
- (f) **Floating point registers (FPRs):** Its store floating point numbers in many architectures.
- (g) **Constant registers:** Its hold read-only values such as zero, one, or pi.
- (h) **Vector registers:** Its hold data for vector processing done by SIMD instructions (Single Instruction, Multiple Data).

Notes

- (i) **Special purpose registers (SPRs):** Its hold program state; they usually include the program counter (aka instruction pointer), stack pointer, and status register (aka processor status word). In embedded microprocessors, they can also correspond to specialized hardware elements.
- (j) **Instruction registers:** Its store the instruction currently being executed.
- (k) In some architectures, **model-specific registers** (also called *machine-specific registers*) store data and settings related to the processor itself. Because their meanings are attached to the design of a specific processor, they cannot be expected to remain standard between processor generations.
- (l) **Control and status registers:** It has three types – Program counter, instruction registers, Program status word (PSW).
- (m) Registers related to fetching information from RAM, a collection of storage registers located on separate chips from the CPU (unlike most of the above, these are generally not *architectural* registers):
 - Memory buffer register
 - Memory data register
 - Memory address register
 - Memory Type Range Registers (MTRR)

Hardware registers are similar, but occur outside CPUs.

Some examples

<i>Architecture</i>	<i>Integer registers</i>	<i>Double FP registers</i>
x86	8	8
x86-64	16	16
IBM/360	16	4
Z/Architecture	16	16
Itanium	128	128
UltraSPARC	32	32
POWER	32	32
Alpha	32	32
6502	3	0
PIC microcontroller	1	0
AVR microcontroller	32	0
ARM	16	16

The table shows the number of registers of several mainstream architectures. Note that the stack pointer (ESP) is counted as an integer register on x86-compatible processors, even though there are a limited number of instructions that may be used to operate on its contents. Similar caveats apply to most architectures.

2.4.2 Register Usage

The number of registers available on a processor and the operations that can be performed using those registers has a significant impact on the efficiency of code generated by optimizing compilers. The Strahler number defines the minimum number of registers required to evaluate an expression tree.

2.5 Computer Bus

Computer Bus is an electrical pathway through which the processor communicates with the internal and external devices attached to the computer. Bus transfers the data between the computer subsystems and between the computers and sends the instructions and commands to and from the processor the different devices. It connects all internal computer components to the main memory and the central processing unit (CPU).

It can logically connect many peripheral devices that can easily communicate with the CPU. A bus is also known as the data bus, address bus or the local bus. The size of the computer bus is important because it determines that how much data can be transferred at a time. Additionally it has the clock speed, which is measured in the MHz. There are different types of the buses which are discussed below. The name of the bus is determined by the type of the signals it carry and the operations it performs.

2.5.1 Data Bus

It carries the data between the different components of the computer.

2.5.2 Address Bus

It selects the route that has to be followed by the data bus to transfer the data.

2.5.3 Control Bus

It decides that whether the data should be written or read from the data bus.

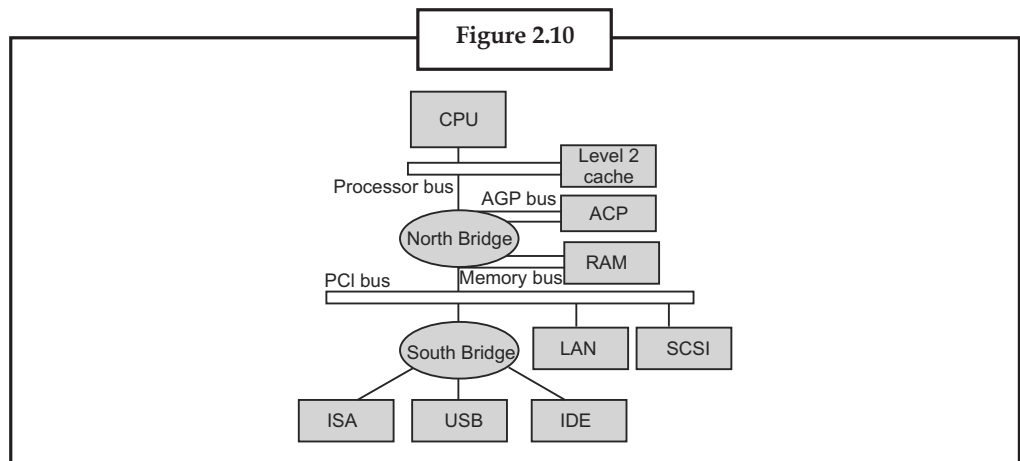
2.5.4 Expansion Bus

It is used to connect the computer's peripheral devices such as printer, modem and scanner with the processor.

Similarly several different types of the buses are being used on the Apple Macintosh computer. In the newer Apple Macintosh systems, the Nubus has been replaced by the PCI.

Buses can be parallel and serial. Today, most of the computers have internal and external buses to connect the internal and external devices to the main memory and CPU. Depending on the type of the bus and computer the bus's operation can be unidirectional or bidirectional.

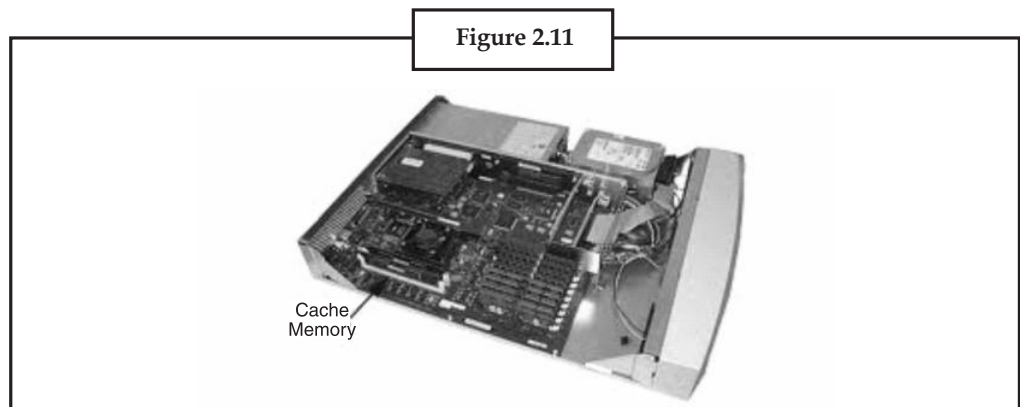
Notes



2.6 Cache Memory

Cache memory is random access memory (RAM) that a computer microprocessor can access more quickly than it can access regular RAM. As the microprocessor processes data, it looks first in the cache memory and if it finds the data there (from a previous reading of data), it does not have to do the more time-consuming reading of data from larger memory. Cache memory is sometimes described in levels of closeness and accessibility to the microprocessor. An L1 cache is on the same chip as the microprocessor. (For example, the PowerPC 601 processor has a 32 kilobyte level-1 cache built into its chip.) L2 is usually a separate static RAM (SRAM) chip. The main RAM is usually a dynamic RAM (DRAM) chip.

In addition to cache memory, one can think of RAM itself as a cache of memory for harddisk storage since all of RAM's contents come from the hard disk initially when you turn your computer on and load the operating system (you are loading it into RAM) and later as you start new applications and access new data. RAM can also contain a special area called a disk cache that contains the data most recently read in from the hard disk.



2.6.1 Operation

Hardware implements cache as a block of memory for temporary storage of data likely to be used again. CPUs and hard drives frequently use a cache, as do web browsers and web servers.

A cache is made up of a pool of entries. Each entry has a datum (a nugget of data) — a copy of the same datum in some backing store. Each entry also has a tag, which specifies the identity of the datum in the backing store of which the entry is a copy.

When the cache client (a CPU, web browser, operating system) needs to access a datum presumed to exist in the backing store, it first checks the cache. If an entry can be found with a tag matching that of the desired datum, the datum in the entry is used instead. This situation is known as a **cache hit**. So, for example, a web browser program might check its local cache on disk to see if it has a local copy of the contents of a web page at a particular URL. In this example, the URL is the tag, and the contents of the web page is the datum. The percentage of accesses that result in cache hits is known as the **hit rate** or **hit ratio** of the cache.

The alternative situation, when the cache is consulted and found not to contain a datum with the desired tag, has become known as a **cache miss**. The previously uncached datum fetched from the backing store during miss handling is usually copied into the cache, ready for the next access.

During a cache miss, the CPU usually ejects some other entry in order to make room for the previously uncached datum. The heuristic used to select the entry to eject is known as the replacement policy. One popular replacement policy, “least recently used” (LRU), replaces the least recently used entry. More efficient caches compute use frequency against the size of the stored contents, as well as the latencies and throughputs for both the cache and the backing store. While this works well for larger amounts of data, long latencies and slow throughputs, such as experienced with a hard drive and the Internet, it is not efficient for use with a CPU cache. When a system writes a datum to the cache, it must at some point write that datum to the backing store as well. The timing of this write is controlled by what is known as the **write policy**.

In a **write-through** cache, every write to the cache causes a synchronous write to the backing store.

Alternatively, in a **write-back** (or **write-behind**) cache, writes are not immediately mirrored to the store. Instead, the cache tracks which of its locations have been written over and marks these locations as **dirty**. The data in these locations are written back to the backing store when those data are evicted from the cache, an effect referred to as a **lazy write**. For this reason, a read miss in a write-back cache (which requires a block to be replaced by another) will often require two memory accesses to service: one to retrieve the needed datum, and one to write replaced data from the cache to the store.

Other policies may also trigger data write-back. The client may make many changes to a datum in the cache, and then explicitly notify the cache to write back the datum.

No-write allocation (a.k.a. write-no-allocate) is a cache policy which caches only processor reads, i.e. on a write-miss:

- Datum is written directly to memory,
- Datum at the missed-write location is not added to cache.

This avoids the need for write-back or write-through when the old value of the datum was absent from the cache prior to the write.

Entities other than the cache may change the data in the backing store, in which case the copy in the cache may become out-of-date or **stale**. Alternatively, when the client updates the data in the cache, copies of those data in other caches will become stale. Communication protocols between the cache managers which keep the data consistent are known as coherency protocols.

2.6.2 Applications

2.6.2.1 CPU Cache

Small memories on or close to the **CPU** can operate faster than the much larger main memory. Most CPUs since the 1980s have used one or more caches, and modern high-end embedded,

Notes

desktop and server **microprocessors** may have as many as half a dozen, each specialized for a specific function. Examples of caches with a specific function are the D-cache and I-cache (data cache and instruction cache).

2.6.2.2 Disk Cache

While CPU caches are generally managed entirely by hardware, a variety of software manages other caches. The page cache in main memory, which is an example of disk cache, is managed by the operating system kernel.

While the hard drive's hardware disk buffer is sometimes misleadingly referred to as "disk cache", its main functions are write sequencing and read prefetching. Repeated cache hits are relatively rare, due to the small size of the buffer in comparison to the drive's capacity. However, high-end disk controllers often have their own on-board cache of hard disk data blocks.

Finally, fast local hard disk can also cache information held on even slower data storage devices, such as remote servers (web cache) or local tape drives or optical jukeboxes. Such a scheme is the main concept of hierarchical storage management.

2.6.2.3 Web Cache

Web browsers and web proxy servers employ **web caches** to store previous responses from web servers, such as web pages. Web caches reduce the amount of information that needs to be transmitted across the network, as information previously stored in the cache can often be re-used. This reduces bandwidth and processing requirements of the web server, and helps to improve responsiveness for users of the web.

Web browsers employ a built-in web cache, but some internet service providers or organizations also use a caching proxy server, which is a web cache that is shared among all users of that network.

Another form of cache is P2P caching, where the files most sought for by peer-to-peer applications are stored in an ISP cache to accelerate P2P transfers. Similarly, decentralised equivalents exist, which allow communities to perform the same task for P2P traffic, e.g. Corelli.

2.6.2.4 Other Caches

The BIND DNS daemon caches a mapping of domain names to IP addresses, as does a resolver library.

Write-through operation is common when operating over unreliable networks (like an Ethernet LAN), because of the enormous complexity of the coherency protocol required between multiple write-back caches when communication is unreliable. For instance, web page caches and client-side network file system caches (like those in NFS or SMB) are typically read-only or write-through specifically to keep the network protocol simple and reliable.

Search engines also frequently make web pages they have indexed available from their cache. For example, Google provides a "Cached" link next to each search result. This can prove useful when web pages from a web server are temporarily or permanently inaccessible.

Another type of caching is storing computed results that will likely be needed again, or memoization, ccache, a program that caches the output of the compilation to speed up the second-time compilation, exemplifies this type.

Database caching can substantially improve the throughput of database applications, for example in the processing of indexes, data dictionaries, and frequently used subsets of data.



Did u know?

Distributed caching uses caches spread across different networked hosts, e.g. Corelli

Notes

2.6.3 The Difference Between Buffer and Cache

The terms “buffer” and “cache” are not mutually exclusive and the functions are frequently combined; however, there is a difference in intent.

A buffer is a temporary memory location, that is traditionally used because CPU instructions cannot directly address data stored in peripheral devices. Thus, addressable memory is used as intermediate stage. Additionally such a buffer may be feasible when a large block of data is assembled or disassembled (as required by a storage device), or when data may be delivered in a different order than that in which it is produced. Also a whole buffer of data is usually transferred sequentially (for example to hard disk), so buffering itself sometimes increases transfer performance or reduce the variation or jitter of the transfer’s latency as opposed to caching where the intent is to reduce the latency. These benefits are present even if the buffered data are written to the buffer once and read from the buffer once.

A cache also increases transfer performance. A part of the increase similarly comes from the possibility that multiple small transfers will combine into one large block. But the main performance-gain occurs because there is a good chance that the same datum will be read from cache multiple times, or that written data will soon be read. A cache’s sole purpose is to reduce accesses to the underlying slower storage. Cache is also usually an abstraction layer that is designed to be invisible from the perspective of neighbouring layers.

2.7 Summary

- Data processing consists of those activities which are necessary to transform data into information.
- OP code (operation code) is the portion of a machine language instruction that specifies is to be performed by the C.P.U.
- Computer memory is basically divided into two type: primary and secondary memory.
- Processor register is a small amount of storage available on the C.P.U. can be accessed more quickly than storage available.
- Data bus carries the data between the different components of the computer.

2.8 Keywords

Computer Bus: Computer Bus is an electrical pathway through which the processor communicates with the internal and external devices attached to the computer.

Data Processing System: A group of interrelated components that seeks the attainment of a common goal by accepting inputs and producing outputs in an organised process.



Lab Exercise

Draw flow of Data Processing Activities.

2.9 Self-Assessment Questions

1. Data processing consists of those activities which are necessary to transform data into information.
(a) True (b) False
2. The data processing activities described above are common to all data processing systems from manual to _____ .
3. Term _____ refers to the activities required to record data and to make it available for processing.
4. Activity of data processing can be viewed as a _____ .
5. RAM is an example of secondary memory.
(a) True (b) False
6. Memory in computer, which is called _____ .
7. Programmable Read Only Memory (PROM) is type of primary memory in computer.
(a) True (b) False
8. User-accessible Registers is not a category of register.
(a) True (b) False

2.10 Review Questions

1. Identify various data processing activities.
2. Define the various steps of data processing cycles.
3. Data processing activities are grouped under following five basic categories.
 - (i) Collection
 - (ii) Conversion
 - (iii) Manipulation
 - (iv) Storage and retrieval
 - (v) Communication
4. Differentiate between
 - (a) RAM and ROM
 - (b) PROM and EPROM
 - (c) Primary memory and Secondary memory
5. Explain cache memory. How is it different from primary memory?
6. Explain The Data Processing Cycle.
7. Explain Registers and Categories of registers.
8. What is Computer Bus?

Answers for Self-Assessment Questions**Notes**

- | | | | |
|--------|---------------------------|-----------------------|-----------|
| 1. (a) | 2. Input | 3. Electronic systems | 4. system |
| 5. (b) | 6. Read Only Memory (ROM) | 7. (a) | 8. (b) |

2.11 Further Reading*Books*

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition.

*Online link*

http://www.webopedia.com/TERM/D/data_process.

Unit 3: Using Operating System

CONTENTS

Objectives

- 3.1 Basics of Operating System
 - 3.1.1 The Operating System: The Purpose
 - 3.1.2 The System Call Model
- 3.2 Types of Operating System
 - 3.2.1 Real-Time Operating System (RTOS)
 - 3.2.2 Single User, Single Task
 - 3.2.3 Single User, Multitasking
 - 3.2.4 Multiprogramming
- 3.3 The User Interface
 - 3.3.1 Graphical User Interfaces (GUIs)
 - 3.3.2 Command-Line Interfaces
- 3.4 Running Programs
 - 3.4.1 Setting Focus
 - 3.4.2 The Xterm Window
 - 3.4.3 The Root Menu
- 3.5 Sharing Files
 - 3.5.1 Directory Access Permissions
 - 3.5.2 File Access Permissions
 - 3.5.3 More Protection Under Linux
- 3.6 Managing Hardware in Operating Systems
 - 3.6.1 Hardware Management Agent Configuration File
 - 3.6.2 Configuring the Hardware Management Agent Logging Level
 - 3.6.3 How to Configure the Hardware Management Agent Logging Level
 - 3.6.4 Configuring your Host Operating System's SNMP
 - 3.6.5 Configuring Net-SNMP/SMA
 - 3.6.6 How to Configure SNMP Gets?
 - 3.6.7 How to Configure SNMP Sets?
 - 3.6.8 How to Configure SNMP Traps?
 - 3.6.9 How to Configure SNMP in Operating Systems?

3.7	Utility Software
3.7.1	Utility Software Categories
3.8	Summary
3.9	Keywords
3.10	Self-Assessment Questions
3.11	Review Questions
3.12	Further Reading

Objectives

After studying this unit, you will be able to:

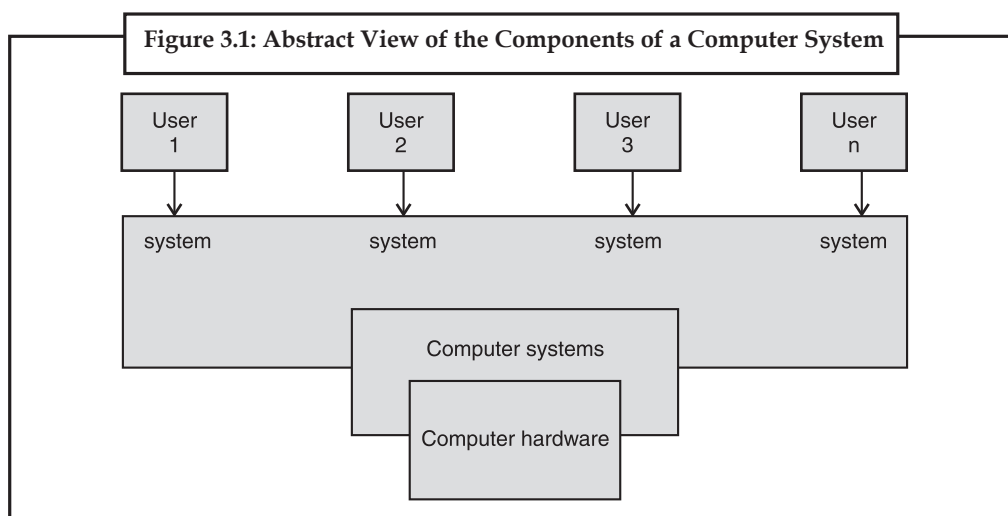
- Basics of operating system.
- Purposes of operating system.
- Types of operating system.
- Discuss of operating system.
- Explain the purpose of operating system.
- Discuss the types of operating system.
- Explain the user interfaces.

Utility Software:

- The user interface
- Running Programme
- Sharing Files
- Managing hardware in operating system

3.1 Basics of Operating System

An operating system is an important part of almost every computer system. A computer system can be divided roughly into four components – the hardware, the operating system, the application programs, and the users (Figure 3.1).



Notes

The hardware-the central processing unit (CPU), the memory, and the input/output (I/O) devices-provides the basic computing resources. The application programs, such as word processors, spreadsheets, compilers, and web browsers-define the ways in which these resources are used to solve the computing problems of the users. The operating system controls and coordinates the use of the hardware among the various application programs for the various users. The components of a computer system are its hardware, software and data.

The operating system provides the means for the proper use of these resources in the operation of the computer system. An operating system is similar to a government. Like a government, it performs no useful function by itself. It simply provides an environment within which other programs can do useful work. Operating systems can be explored from two viewpoints: the user and the system.



Example:

```
/* syscall.c
*
* System call "stealing" sample.
*/

/* The necessary header files */

/* Standard in kernel modules */
#include <linux/kernel.h> /* We're doing kernel work */
#include <linux/module.h> /* Specifically, a module */

/* Deal with CONFIG_MODVERSIONS */
#if CONFIG_MODVERSIONS==1
#define MODVERSIONS
#include <linux/modversions.h>
#endif
```

3.1.1 The Operating System: The Purpose

The operating system is the core software component of your computer. It performs many functions and is, in very basic term, an interface between your computer and the outside world. An operating system provides an environment for the execution of programs by providing services needed by those programs.

The services programs request fall into five categories:-

1. Process Control
2. File System Management
3. I/O Operation
4. Interprocess Communication
5. Information Maintenance

The operating system must try to satisfy these requests in a multi-user, multi-process environment while managing-

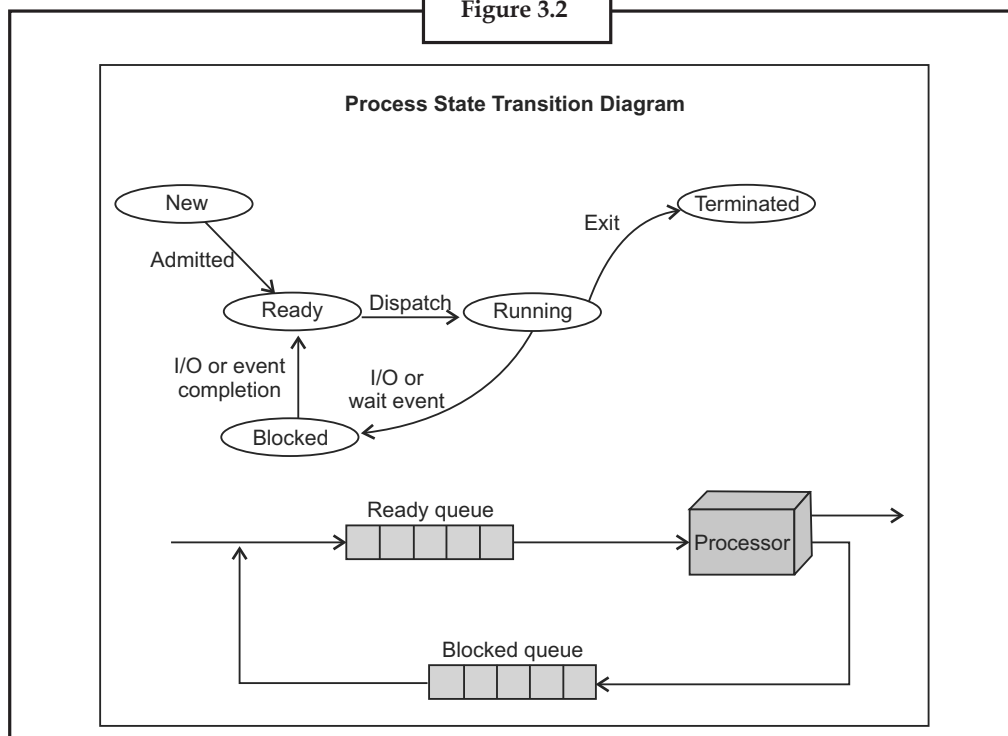
- Resource allocation
- Error Detection
- Protection

1. The Operating System: Process Control

A process is the unit of work. Processes need to be able to control their own execution, as well as spawn new processes to perform tasks concurrently. Processes request services to:

- fork: create a new process
- exit: normal termination, abort: abnormal termination
- execve: load and execute another program
- wait: wait for another process to finish
- Signal Management: handling asynchronous events
 - signal, sigaction: setting signal handlers
 - kill: sending signal
- Threads: creating and managing multiple execution threads in a single process

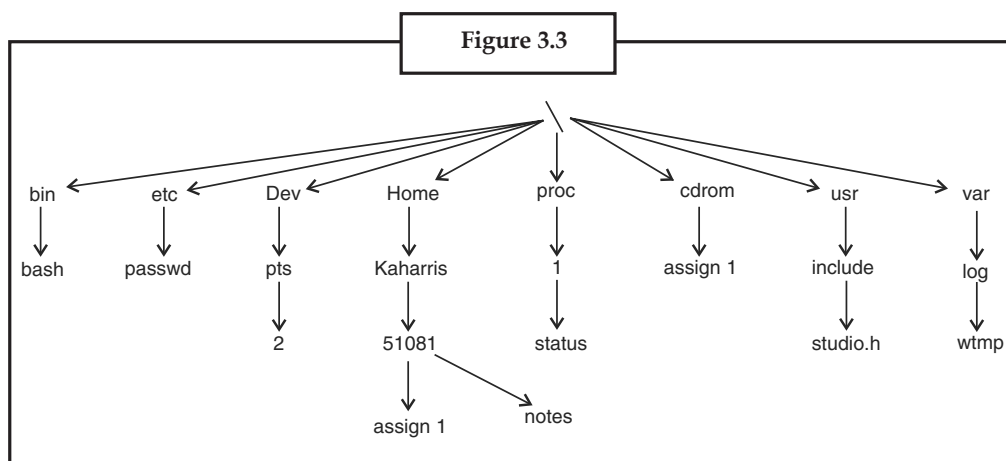
Figure 3.2



2. The Operating System: File System Management

The User sees ...

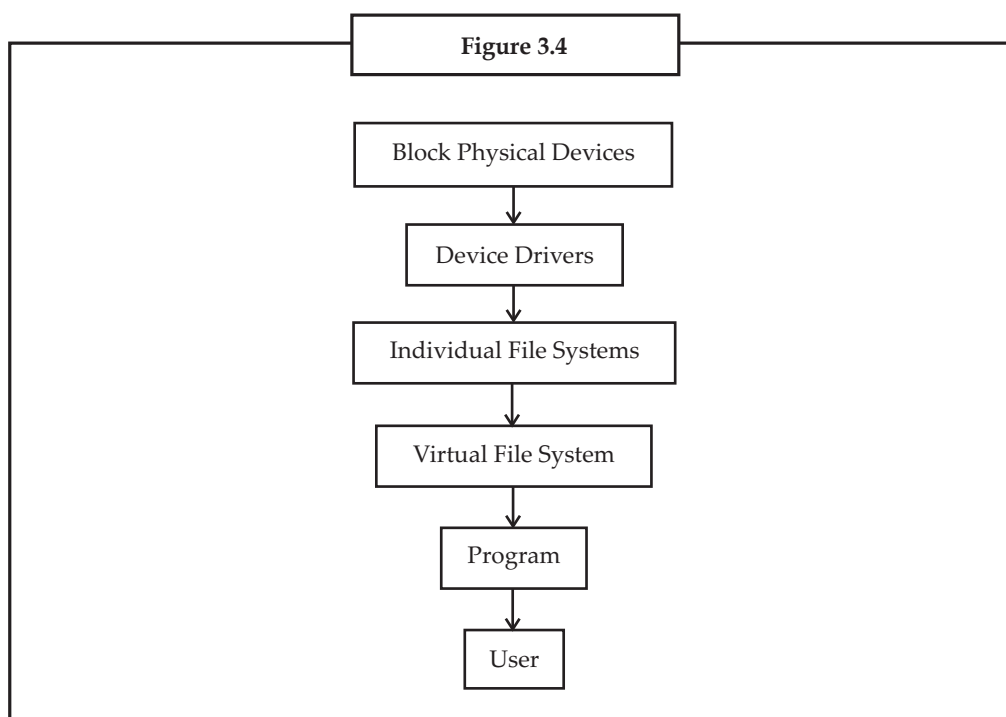
Notes



The Unix File System is much more complicated than it appears:

- The root directory is located on hard disk whose file system is organized using a Linux file system organization, ext2.
- There is a CD-ROM in the cd drive whose file system is organized using High-SierraFile Systems, HSFS.
- The directory /proc does not even exist on a device. The files are not storage locations of data yet they appear to have content: cat /proc/1/status or cat /proc/mounts.
- The “files” in /dev are actually physical devices. For example, /dev/pts/2 is your terminal device, /dev/mouse is my mouse and /dev/lp0 is a printer.

The File System: Layers of Abstraction



The Unix File System is a high-level organization of the resources available to processes and users. The Unix File System is organized as an acyclic directed graph with nodes representing files and arcs representing a containment relation. Directories are nodes with children, so contain files. Processes move around and modify the structure:

- chdir : moving to a new directory
- getcwd : get the current working directory
- opendir, closedir : open and close a directory
- readdir : read from a directory
- stat, fstat : Retrieving file status information
- link, unlink : create and release a hard link (an alias for a file.)
- symlink : create a soft link (a pointer to another file)

3. The Operating System: I/O Operation

Unix uses a uniform device interface that allows the same I/O calls to be used for terminals, disks, audio and network communication. Unix provides a universal interface for I/O:

- open : open a file or device for I/O operations
- close : close a file or device from I/O operations
- read : read from a file or device
- write : write to a file or device
- Refined I/O control:
 - fcntl : getting and setting attributes of an open file
 - ioctl : getting device status information and setting device control options
 - poll,select : handling I/O from multiple sources

4. The Operating System: Interprocess Communication

Concurrently running processes need to communicate to work together effectively. Unix provides a variety of means for processes to communicate with each other:

- pipe : a one-way data stream between related processes.
- mkfifo : a one-way data stream between unrelated processes (called a named pipe or FIFO)
- System V IPC: refined communication channels for unrelated processes
 - Message Queues : Linked lists of messages stored in the operating system
 - Shared Memory : Allows two processes to share a given region of memory
 - Semaphores : Provides controlled access to a shared object
- Sockets : Two-way data stream, used to establish Network connections.

5. The Operating System: System Information

Unix provides means for accessing information about the system for process use or accounting:

- Special directories which provide Start-up and Run-time information. Some of the files are specially configured, so provide special functions to retrieve information.

Notes**Examples**

- /etc/ : System configuration files. Examples
 - * /etc/passwd
 - * /etc/group
- /var/ : Runtime changable files. Examples
 - * /var/run/utmp : Currently logged in users
 - * /var/log/wtmp : All logins and logouts
- /proc/ : Process information
- Time and Date functions in <time.h>
- uname : system identification
- sysconf, pathconf : Information on system limits

3.1.2 The System Call Model

In computing, a system call is the mechanism used by an application to request service from the operating system based on the monolithic kernel or to system services on the operating systems based on the microkernel-structure. System Calls are requests by programs for services to be provided by the Operating System.

The system call represents the interface of a program to the kernel of the Operating System:

1. The system call API (Application Program Interface) is given in C.
2. The system call is actually a wrapper routine typically consisting of a short piece of assembly language code.
3. The system call generates a hardware trap, and passes the user's arguments and an index into the kernel's system call table for the service requested.
4. The kernel processes the request by looking-up the index passed to see the service to perform, and carries out the request.
5. Upon completion, the kernel returns a value which represents either successful completion of the request or an error. If an error occurs, the kernel sets a global variable, err no, indicating the reason for the error.
6. The process is delivered the return value and continues its execution.

3.1.2.1 System Call: Motivation behind the Model

The system call model ensures safety, fairness and modularity. The benefits of the system call model include:

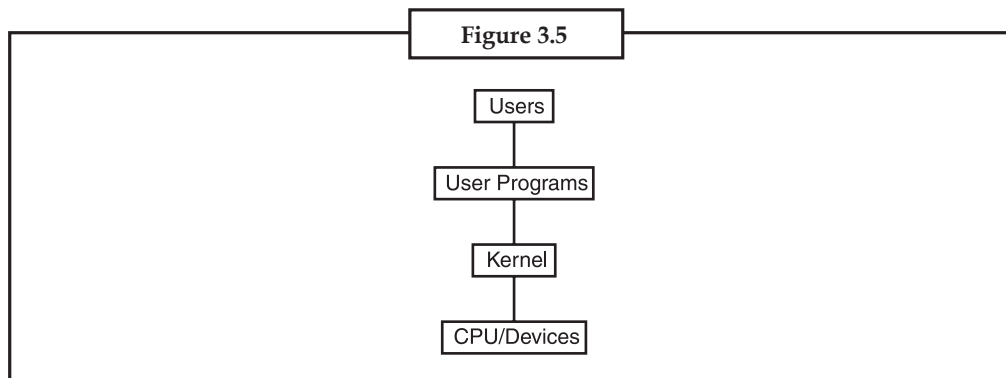
- The Operating System ensures the integrity of its data and the data of other users, by carefully controlling access to these resources.
- The Operating System ensures fair use of system resources.
- The system call interface is standardized by POSIX (Portable Operating System Interface), so is identical for all Unix flavors. (BSD, Linux, Solaris, Mach, etc.) regardless how any operating system actually implements the services the process requests.

- The system call interface simplifies program construction by removing the details of implementing the request to the operating system.

Notes

3.1.2.2 The System Call Model: Layers of Interaction

There are layers of interaction in the system to ensure fair access to resources, protect privacy, and provide convenience.



3.2 Types of Operating System

Within the broad family of operating systems, there are generally four types, categorized based on the types of computers they control and the sort of applications they support. The categories are real-time operating system, single user single task, single user multitasking and multi-user.

3.2.1 Real-Time Operating System (RTOS)

Real-time operating systems are used to control machinery, scientific instruments and industrial systems such as embedded systems (programmable thermostats, household appliance controllers), industrial robots, spacecraft, industrial control (manufacturing, production, power generation, fabrication, and refining), and scientific research equipment.

An RTOS typically has very little user-interface capability, and no end-user utilities, since the system will be a “sealed box” when delivered for use. A very important part of an RTOS is managing the resources of the computer so that a particular operation executes in precisely the same amount of time, every time it occurs. In a complex machine, having a part move more quickly just because system resources are available may be just as catastrophic as having it not move at all because the system is busy.

An RTOS facilitates the creation of a real-time system, but does not guarantee the final result will be real-time; this requires correct development of the software. An RTOS does not necessarily have high throughput; rather, an RTOS provides facilities which, if used properly, guarantee deadlines can be met generally (soft real-time) or deterministically (hard real-time). An RTOS will typically use specialized scheduling algorithms in order to provide the real-time developer with the tools necessary to produce deterministic behavior in the final system. An RTOS is valued more for how quickly and/or predictably it can respond to a particular event than for the given amount of work it can perform over time. Key factors in an RTOS are therefore a minimal interrupt latency (the time between the generation of an interrupt by a device and the servicing of the device which generated the interrupt) and a minimal thread switching latency (the time needed by the operating system to switch the CPU to another thread).

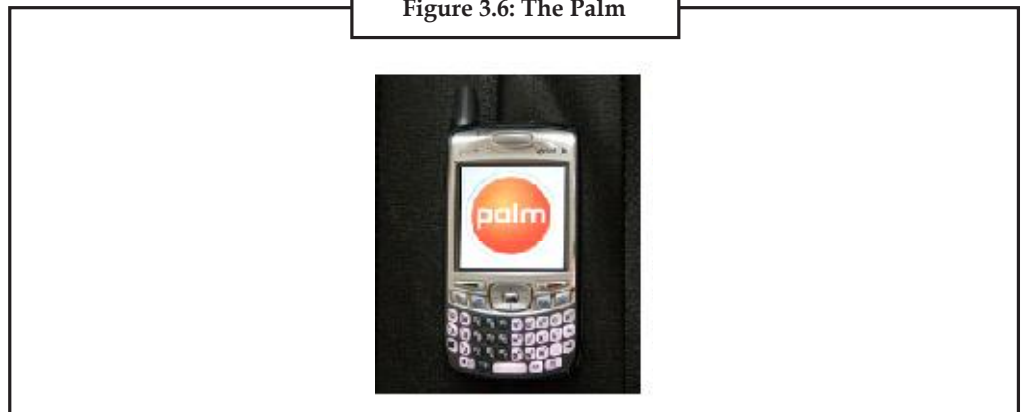
Notes

An early example of a large-scale real-time operating system was Transaction Processing Facility. Current users include Sabre (reservations), Amadeus (reservations), VISA Inc (authorizations), Holiday Inn (central reservations), CBOE (order routing), Singapore Airlines, KLM, Qantas, Amtrak, Marriott International, Worldspan and the NYPD (911 system).

3.2.2 Single User, Single Task

As the name implies, this operating system is designed to manage the computer so that one user can effectively do one thing at a time. The Palm OS for Palm handheld computers is a good example of a modern single-user, single-task operating system.

Figure 3.6: The Palm



Treo 700p is one of many Smartphones produced that combines Palm PDA functions with a cell phone, allowing for built-in voice and data.

3.2.3 Single User, Multitasking

This is the type of operating system most people use on their desktop and laptop computers today. Microsoft's Windows and Apple's Mac OS platforms are both examples of operating systems that will let a single user have several programs in operation at the same time. For example, it's entirely possible for a Windows user to be writing a note in a word processor while downloading a file from the Internet while printing the text of an e-mail message.

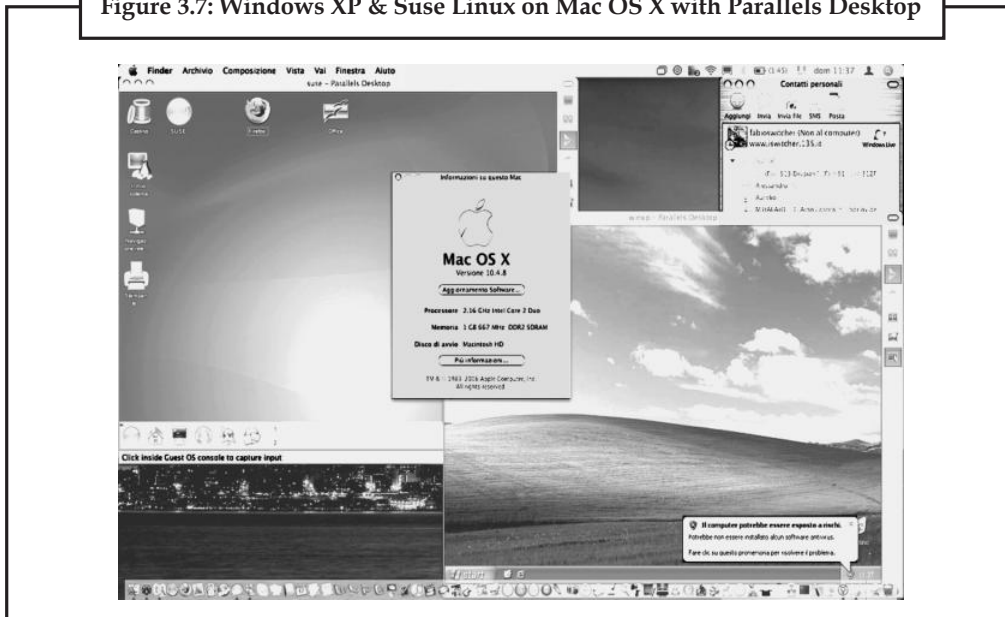
3.2.3.1 Multi-User

Multi-user is a term that defines an operating system or application software that allows concurrent access by multiple users of a computer. A multi-user operating system allows many different users to take advantage of the computer's resources simultaneously. The operating system must make sure that the requirements of the various users are balanced, and that each of the programs they are using has sufficient and separate resources so that a problem with one user doesn't affect the entire community of users. Unix, VMS and mainframe operating systems, such as *MVS*, are examples of multi-user operating systems.

Time-sharing systems are multi-user systems. Most batch processing systems for mainframe computers may also be considered "multi-user", to avoid leaving the CPU idle while it waits for I/O operations to complete. However, the term "multitasking" is more common in this context.

An example is a Unix server where multiple remote users have access (such as via Secure Shell) to the Unix shell prompt at the same time. Another example uses multiple X Window sessions spread across multiple terminals powered by a single machine - this is an example of the use of thin client.

Figure 3.7: Windows XP & Suse Linux on Mac OS X with Parallels Desktop



Management systems are implicitly designed to be used by multiple users, typically one system administrator or more and an end-user community.

It's important to differentiate between multi-user operating systems and single-user operating systems that support networking. Windows 2000 and Novell Netware can each support hundreds or thousands of networked users, but the operating systems themselves aren't true multi-user operating systems. The **system administrator** is the only "user" for Windows 2000 or Netware. The network support and all of the remote user logins the network enables are, in the overall plan of the operating system, a program being run by the administrative user.

3.2.4 Multiprogramming

Multiprogramming is a rudimentary form of parallel processing in which several programs are run at the same time on a uniprocessor. Since there is only one processor, there can be no true simultaneous execution of different programs. Instead, the operating system executes part of one program, then part of another, and so on. To the user it appears that all programs are executing at the same time.

If the machine has the capability of causing an interrupt after a specified time interval, then the operating system will execute each program for a given length of time, regain control, and then execute another program for a given length of time, and so on. In the absence of this mechanism, the operating system has no choice but to begin to execute a program with the expectation, but not the certainty, that the program will eventually return control to the operating system.

If the machine has the capability of protecting memory, then a bug in one program is less likely to interfere with the execution of other programs. In a system without memory protection, one program can change the contents of storage assigned to other programs or even the storage assigned to the operating system. The resulting system crashes are not only disruptive, they may be very difficult to debug since it may not be obvious which of several programs is at fault.

Notes

Distributed Operating System: An operating system that manages a group of independent computers and makes them appear to be a single computer is known as a distributed operating system. The development of networked computers that could be linked and communicate with each other, gave rise to distributed computing. Distributed computations are carried out on more than one machine. When computers in a group work in cooperation, they make a distributed system.

Embedded System: The operating systems designed for being used in embedded computer systems are known as embedded operating systems. They are designed to operate on small machines like PDAs with less autonomy. They are able to operate with a limited number of resources. They are very compact and extremely efficient by design.

Windows CE, FreeBSD and Minix 3 are some examples of embedded operating systems.

The operating systems thus contribute to the simplification of the human interaction with the computer hardware. They are responsible for linking application programs with the hardware, thus achieving an easy user access to the computers.

3.3 The User Interface

The user interface is the space where interaction between humans and machines occurs. The goal of interaction between a human and a machine at the user interface is effective operation and control of the machine, and feedback from the machine which aids the operator in making operational decisions. A user interface is the system by which people (users) interact with a machine. The user interface includes hardware (physical) and software (logical) components. User interfaces exist for various systems, and provide a means of:

- Input, allowing the users to manipulate a system, and/or
- Output, allowing the system to indicate the effects of the users' manipulation.

The following types of user interface are the most common:

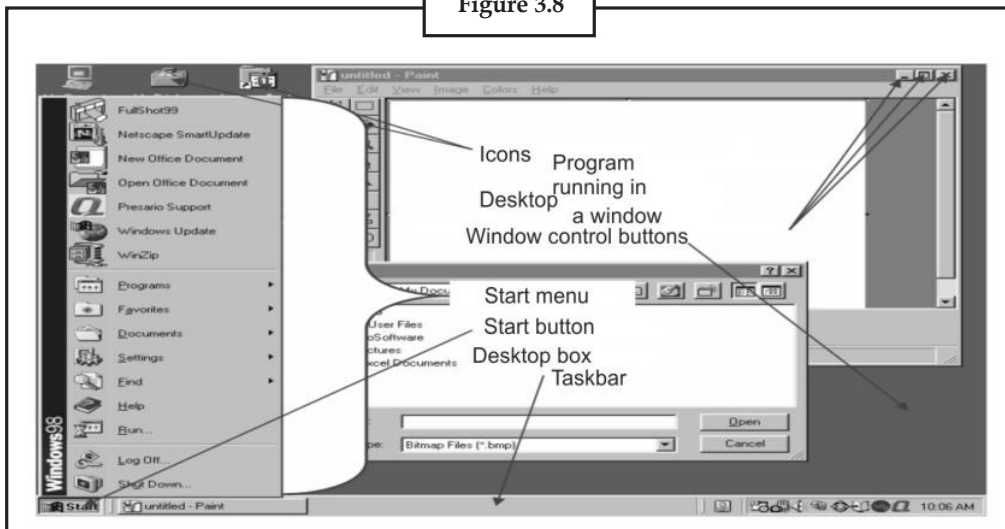
- Graphical User Interfaces (GUIs)
- Command-Line Interfaces

3.3.1 Graphical User Interfaces (GUIs)

Most modern operating systems, like Windows and the Macintosh OS, provide a graphical user interface (GUI). A GUI lets you control the system by using a mouse to click graphical objects on screen. A GUI is based on the desktop metaphor. Graphical objects appear on a background (the desktop), representing resources you can use.

Icons are pictures that represent computer resources, such as printers, documents, and programs. You double-click an icon to choose (activate) it, for instance, to launch a program. The Windows operating system offers two unique tools, called the taskbar and Start button. These help you run and manage programs.

Figure 3.8



3.3.2 Command-Line Interfaces

A CLI (command line interface) is a user interface to a computer's operating system or an application in which the user responds to a visual prompt by typing in a command on a specified line, receives a response back from the system, and then enters another command, and so forth. The MS-DOS Prompt application in a Windows operating system is an example of the provision of a command line interface. Today, most users prefer the graphical user interface (GUI) offered by Windows, Mac OS, BeOS, and others. Typically, most of today's UNIX-based systems offer both a command line interface and a graphical user interface.



Case Study

In the 1980s UNIX, VMS and many others had operating systems that were built this way. GNU/Linux and Mac OS X are also built this way. Modern releases of Microsoft Windows such as Windows Vista implement a graphics subsystem that is mostly in user-space; however the graphics drawing routines of versions between Windows NT 4.0 and Windows Server 2003 exist mostly in kernel space. Windows 9x had very little distinction between the interface and the kernel.

3.4 Running Programs

One of the most important X features is that windows can come either from programs running on another computer or from an operating system other than UNIX. So, if your favorite MS-DOS program doesn't run under UNIX but has an X interface, you can run the program under MS-DOS and display its windows with X on your UNIX computer. Researchers can run graphical data analysis programs on supercomputers in other parts of the country and see the results in their offices.

3.4.1 Setting Focus

Of all the windows on your screen, only one window receives the keystrokes you type. This window is usually highlighted in some way. By default in the **mwm** window manager, for instance, the frame of the window that receives your input is a darker shade of grey. In X jargon,

Notes

choosing the window you type to is called “setting the *input focus*.” Most window managers can be configured to set the focus in one of the following two ways:

- (a) Point to the window and click a mouse button (usually the first button). You may need to click on the titlebar at the top of the window.
- (b) Simply move the pointer inside the window.

When you use **mwm**, any new windows will get the input focus automatically as they pop up.

3.4.2 The Xterm Window

One of the most important windows is an **xterm** window. **xterm** makes a terminal emulator window with a UNIX login session inside, just like a miniature terminal. You can have several **xterm** windows at once, each doing something different. To enter a UNIX command or answer a prompt in a window, set the focus there and type. Programs in other windows will keep running; if they need input from you, they’ll wait just as they would on a separate terminal.

You can also start separate X-based window programs (typically called *clients*) by entering commands in an **xterm** window. Although you can start new clients (**xterm**, **xcalc**, and so on) from any open **xterm** window on your computer, we recommend starting all of them from the first window that you opened.

Here’s an example. To start the calculator called **xcalc**, enter:

```
% xcalc &
```

```
12345 %
```

The shell will print a PID number like 12345 has more information on this subject.) If you forget to add the ampersand (&) at the end of the line, kill (terminate) **xcalc** with your interrupt character (like [CTRL-C] to get another shell prompt—then enter the command correctly.

The new window may be placed and get the focus automatically. Or, the window (or an outline of it) may “float” above the display, following the pointer—until you point somewhere and click the mouse button to place the window.

You can also start a new **xterm** from an existing **xterm**. Just enter **xterm** & (don’t forget the ampersand) at the shell prompt.

The same method works for starting other X programs.

3.4.3 The Root Menu

If you move the pointer onto the root window (the “desktop” behind the windows) and press the correct mouse button (usually the first or third button, depending on your setup), you should see the *root menu*. You may need to hold down the button to keep the menu visible. The root menu has commands for controlling windows. The menu’s commands may differ depending on the system. Your system administrator (or you, if you study your window manager) can add commands to the root menu. These can be window manager operations or commands to open other windows. For example, a “New Window” menu item can open a new **xterm** window for you. A “Calculator” item could start **xcalc**.

3.5 Sharing Files

UNIX makes it easy for users to share files and directories. Controlling exactly who has access takes some explaining, though—more explaining than we can do here. So here’s a cookbook set of instructions.

3.5.1 Directory Access Permissions

A directory's access permissions help to control access to the files in it. These affect the overall ability to use files and subdirectories in the directory. (Once you have access to a directory, the ability to read or modify the contents of specific files is controlled by the file access permissions; see the second of the following two lists.)

In the commands below, replace *dirname* with the directory's pathname. An easy way to change permissions on the working directory is by using its relative pathname, *.* (dot), as in "**chmod 755 .**".

- To keep yourself from accidentally removing files (or adding or renaming files) in a directory, use **chmod 555 dirname**. To do the same, but also deny other users any access, use **chmod 500 dirname**.
- To protect the files in a directory and all its subdirectories from everyone else on your system - but still be able to do anything *you* want to do there — use **chmod 700 dirname**.
- To let other people on the system see what's in a directory — and read or edit the files if the file permissions let them — but not rename, remove, or add files — use **chmod 755 dirname**.
- To let people in your UNIX group add, delete, and rename files in a directory of yours — and read or edit other people's files if the file permissions let them — use **chmod 775 dirname**.
- To give full access to everyone on the system, use **chmod 777 dirname**.

Remember, to access a directory, a user must also have execute (x) permission to all of its parent directories, all the way up to the root.

3.5.2 File Access Permissions

The access permissions on a file control what can be done to the file's *contents*. The access permissions on the *directory* where the file is kept control whether the file can be renamed or removed.

- (a) To make a private file that only you can edit, use **chmod 600 filename**. To protect it from accidental editing, use **chmod 400 filename**.
- (b) To edit a file yourself, and let everyone else on the system read it without editing, use **chmod 644 filename**.
- (c) To let you and all members of your UNIX group edit a file, but keep any other user from reading or editing it, use **chmod 660 filename**.
- (d) To let nongroup users read but not edit the file, use **chmod 664 filename**.
- (e) To let anyone read or edit the file, use **chmod 666 filename**.

3.5.3 More Protection Under Linux

Most Linux systems have a command that gives you more choices on file and directory protection: **chattr**. **chattr** is being developed, and your version may not have all of the features that it will have in later versions of Linux. For instance, **chattr** can make a Linux file *append-only* (so it can't be overwritten, only added to); *compressed* (to save disk space automatically); *immutable* (so it can't be changed at all); *undeletable*, and more. Check your online documentation (type **man chattr**) or ask your system administrator for advice on your system.

3.6 Managing Hardware in Operating Systems

It is possible to supply command-line options when starting the Hardware Management Agent manually. Use the command line options as follows:

`hwagentd OPTIONS`

The command line options are explained in the following table.

Option	Function
-h	Displays the usage message and exits.
-v	Displays the Hardware Management Agent version currently installed on this Sun x64 Server and exits.
-l <i>level</i>	Override the logging level set in the <code>hwagentd.conf</code> file with <i>level</i> .

When using the logging levels option, you must supply a decimal number to set the logging level to use. This decimal number is calculated from a bit field, depending on the logging level you want to specify. For more information on the bit field used to configure different log levels.

3.6.1 Hardware Management Agent Configuration File

Once the Hardware Management Agent and Hardware SNMP Plugins are installed on the Sun x64 Server you want to monitor, you can configure them. There is only one configuration file for the Hardware Management Agent, which configures the level of detail used for log messages. Depending on which host operating system the Hardware Management Agent is running on, the configuration file can be found at the path shown in the following table.

Operating System	Configuration file path
Linux	<code>/etc/sun-ssm/hwagentd.conf</code>
Solaris Operating	<code>/etc/opt/sun-ssm/hwagentd.conf</code>
System Windows	<code>C:\Program Files\Sun Microsystems\SSM\Sun Server Hardware Management Agent\conf\hwmgmt.d.conf</code>

The Hardware Management Agent records log messages into the log file. These messages can be used to troubleshoot the running status of the Hardware Management Agent. The following table shows the path of the log file.

Operating SystemLog	Log File Path
Linux	<code>/var/log/sun-ssm/hwagentd.log</code>
Solaris	<code>/var/opt/sun-ssm/hwagentd.log</code>
Windows	<code>C:\Program Files\Sun Microsystems\SSM\Sun Server Hardware Management Agent\log\hwmgmt.d.log</code>

3.6.2 Configuring the Hardware Management Agent Logging Level

To configure the logging level, modify the `hwagentd_log_levels` parameter in the `hwagentd.conf` file. The `hwagentd_log_levels` parameter is a bit flag set expressed as a decimal integer.

The following table explains the different logging levels that can be configured using the various bit fields.

Log Level	Bit Code	Messages Logged
EMERG	0x0001	Information about the system being unusable
ALARM	0x0002	Information about any immediate action that must be taken
CRIT	0x0004	Information related to the Hardware Management Agent either not starting or stopping because of critical conditions
ERROR	0x0008	Information related to the Hardware Management Agent either not starting or stopping because of critical conditions
WARNING	0x0010	Information about any conditions that generate a warning, which do not stop the Hardware Management Agent
NOTICE	0x0020	Information related to normal functioning which is significant
INFO	0x0040	Informative messages about normal functioning
DEBUG	0x0080	Verbose debug-level messages, useful in troubleshooting
TRACE	0x0100	Highly verbose debug-level messages, useful in troubleshooting



Caution

Levels DEBUG and TRACE generate a lot of detailed messages and are designed for troubleshooting. These levels are not recommended for production usage.

3.6.3 How to Configure the Hardware Management Agent Logging Level

- (a) Depending on the host operating system that the Hardware Management Agent is running on, open the `hwagentd.conf` file from the path shown in the following table. You can use any text editor to modify this file.
- (b) Find the `hwagentd_log_levels` parameter and enter the decimal number calculated using the instructions above.
- (c) Save the modified `hwagentd.conf` file.
- (d) Choose one of the following options to make the Hardware Management Agent reread the `hwagentd.conf` file:
 - On Linux and Solaris operating systems, you can manually restart (Solaris operating system: refresh) the Hardware Management Agent, which forces the `hwagentd.conf` to be reread. Depending on the host operating system that the Hardware Management Agent is running on, restart the Hardware Management Agent.
 - On Windows operating systems, you can restart the service using the Microsoft Management Console Services snap-in.

The Hardware Management Agent rereads the `hwagentd.conf` file with the modified `hwagentd_log_levels` parameter.

3.6.4 Configuring your Host Operating System's SNMP

The Hardware Management Agent uses SNMP for network communications. For the Hardware Management Agent to be able to use SNMP correctly on host operating systems, you must ensure that SNMP is configured correctly. On Linux and Solaris operating systems, the `snmpd.conf` file controls network access to the Hardware Management Agent. On Windows operating systems the SNMP service controls network access to the Hardware Management Agent.

Notes

Incorrect settings can cause the Hardware Management Agent to have limited, or no, network connectivity.

3.6.5 Configuring Net-SNMP/SMA

Depending on which operating system the Hardware Management Agent has been installed on, you can find the `snmpd.conf` file at the path shown in the following table.

Operating System	Path to <code>snmpd.conf</code>
Linux	<code>/etc/snmp/snmpd.conf</code>
Solaris Operating System	<code>/etc/sma/snmp/snmpd.conf</code>

The exact modifications you need to make to the `snmpd.conf` file depend on which host operating system the Hardware Management Agent is running on. The following procedures explain how to configure SNMP gets, sets, and traps.



The following instructions assume you are using an unmodified `snmpd.conf` file. If you have customized your `snmpd.conf` file, please consider these instructions as a guide to how to make sure your `snmpd.conf` file is compatible with the Hardware Management Agent.

3.6.6 How to Configure SNMP Gets?

SNMP gets enable you to read data filled by the Hardware Management Agent. To be able to perform SNMP gets, use the following information to modify your `snmpd.conf` file, depending on which host operating system the Hardware Management Agent is running on.

- (a) Open your `snmpd.conf` file for editing.
- (b) Depending on which operating system you are running, choose one of the following options:
 - For Red Hat Enterprise Linux, add the following line to **`snmpd.conf`**:
`view systemview included .1.3.6.1.4.`
This adds the Hardware SNMP Plugins to the specified view.
 - For SUSE Linux Enterprise Server, add the following line to **`snmpd.conf`**:
`rocommunity public 31`

3.6.7 How to Configure SNMP Sets?

To enable the functionality of setting information via SNMP, use the following information to modify your `snmpd.conf` file, depending on which host operating system the Hardware Management Agent is running on.

- (a) Open your `snmpd.conf` file for editing.
- (b) Depending on which operating system you are running, choose one of the following options:
 - For SUSE Linux Enterprise Server, VMware ESX and Solaris you should add the following line to `snmpd.conf`: `rwcommunity private`
By default the public community is blocked as `rocommunity` on these operating systems.

- For Red Hat Enterprise Linux, change the following line in `snmpd.conf`:
`access notConfigGroup "" any noauth exact systemview none none` to the following:
`access notConfigGroup "" any noauth exact systemview systemview none`

This modification grants write access for the specified view and group. In this example the specified view is *systemview* and the specified group is *NotConfigGroup*. By default, the group uses the public community string.

3.6.8 How to Configure SNMP Traps?

1. Open your `snmpd.conf` file for editing.
2. Depending on the version of SNMP traps you want to send:
 - To be able to send SNMP version 1 traps from the Hardware Management Agent, add the following line to `snmpd.conf`:

```
trapsink host communitystringtrapport
```

- To be able to send SNMP version 2 traps from the Hardware Management Agent, add the following line to `snmpd.conf`:

```
trap2sink host communitystringtrapport
```

Setting SNMP Version 2 Traps

The following example shows the line added to the `snmpd.conf` file to configure SNMP Traps using SNMP version 2:

```
trap2sink 10.18.141.22 public 162
```

3.6.9 How to Configure SNMP in Operating Systems?

- (a) From the Start menu Administrative Tools option, select Services.
- (b) Double-click the SNMP service.
- (c) On the Security tab, configure the community rights.
- (d) On the Traps tab, configure the destination you want to send SNMP traps.



Did u know?

The Microsoft Management Console Services snap-in opens.

3.7 Utility Software

Utility software is a kind of system software designed to help analyze, configure, optimize and maintain the computer. A single piece of utility software is usually called a **utility** (*abbr. util*) or **tool**.

Utility software should be contrasted with application software, which allows users to do things like creating text documents, playing games, listening to music or surfing the web. Rather than providing these kinds of user-oriented or output-oriented functionality, utility software usually focuses on *how* the computer infrastructure (including the computer hardware, operating system, and application software and data storage) operates. Due to this focus, utilities are often rather technical and targeted at people with an advanced level of computer knowledge.

Notes

Most utilities are highly specialized and designed to perform only a single task or a small range of tasks. However, there are also some utility suites that combine several features in one piece of software.

Most major operating systems come with several pre-installed utilities.

3.7.1 Utility Software Categories

- **Disk storage** utilities
- **Disk defragmenters** can detect computer files whose contents are broken across several locations on the hard disk, and move the fragments to one location to increase efficiency.
- **Disk checkers** can scan the contents of a hard disk to find files or areas that are corrupted in some way, or were not correctly saved, and eliminate them for a more efficiently operating hard drive.
- **Disk cleaners** can find files that are unnecessary to computer operation, or take up considerable amounts of space. Disk cleaner helps the user to decide what to delete when their hard disk is full.
- **Disk space analyzers** for the visualization of disk space usage by getting the size for each folder (including sub folders) & files in folder or drive. showing the distribution of the used space.
- **Disk partitions** can divide an individual drive into multiple logical drives, each with its own file system which can be mounted by the operating system and treated as an individual drive.
- **Backup** utilities can make a copy of all information stored on a disk, and restore either the entire disk (e.g. in an event of disk failure) or selected files (e.g. in an event of accidental deletion).
- **Disk compression** utilities can transparently compress/uncompress the contents of a disk, increasing the capacity of the disk.
- **File managers** provide a convenient method of performing routine data management tasks, such as deleting, renaming, cataloging, uncataloging, moving, copying, merging, generating and modifying data sets.
- **Archive** utilities output a stream or a single file when provided with a directory or a set of files. Archive utilities, unlike archive suites, usually do not include compression or encryption capabilities. Some archive utilities may even have a separate un-archive utility for the reverse operation.
- **System profilers** provide detailed information about the software installed and hardware attached to the computer.
- **System monitors** for monitoring resources and performance in a computer system.
- **Anti-virus** utilities scan for computer viruses.
- **Hex editors** directly modify the text or data of a file. These files could be data or an actual program.

- **Data compression** utilities output a shorter stream or a smaller file when provided with a stream or file.
- **Cryptographic** utilities encrypt and decrypt streams and files.
- **Launcher applications** provide a convenient access point for application software.
- **Registry cleaners** clean and optimize the Windows registry by removing old registry keys that are no longer in use.
- **Network utilities** analyze the computer's network connectivity, configure network settings, check data transfer or log events.
- **Screensavers** were desired to prevent phosphor burn-in on CRT and plasma computer monitors by blanking the screen or filling it with moving images or patterns when the computer is not in use. Contemporary screensavers are used primarily for entertainment or security.



Case Study

OS68 for Embedded Systems

OS68 overview: OS68 real-time operating system consists of a small configurable executive kernel and a set of system tools such as debugging tools, target simulation tools. The executive kernel is small in size about 11 kilo bytes in binary. It is simple due to its very limited set of system calls. The design goals of OS68 are easy to use, efficient debugging tools and high performance.

Easy to use: The number of system calls used in OS68 design are reduced compared to other operating systems. 90 - 100% of the processes in a system need 6 or fewer system calls. It is recommended to be restrictive when using tools other than the six basic calls. The six basic calls are:

- Alloc:** Allocate memory
 - Free Buf:** Return allocated memory to OS.
 - Send:** Send a message to a process.
 - Receive:** Receive selected messages, sleep if no selected message is available.
 - Receiv_W_TMO:** Like RECEIVE, but wake up again after a specified time if no message is available.
 - Delay:** Sleep for a specific time.
- Efficient debugging tools improve application development, and high performance is due to configurable OS system according to specific application or specific processor families.
- Process and its States:** A process is an independent program running under real-time operating system. A process running under a real-time operating system can find itself in one of three distinctly different states.
 - Running:** The process is the one currently in control of the CPU. In a single process system, only one process can be in this state at a time.

Contd...

Notes

- (i) **Ready:** The process needs the CPU and will be running as soon as it is permitted to do so. However, another process of higher or equal priority is RUNNING and the READY process must wait until all READY process of higher priority are waiting.
- (j) **Waiting:** The process is waiting for something, for example a message, and has no need of the CPU.
- (k) **Process types and scheduling in OS68:** The most frequently used task/process scheduling paradigm of the currently available commercial real-time operating system kernels is the preemptive priority-based scheme. The OS68 executive is also based on this paradigm / 2 / . The scheduling principle is simple: Always executes the highest-priority process of the ready processes. Preemption in this context means that the scheduler may (virtually) at any time suspend a running low-priority process for a higher-priority process.

The OS68 operating system provides four different types of processes: prioritized process, background processes, interrupt processes and timer processes.

Prioritized processes operate on different priority levels. Processes on higher priority level always run before processes of lower priority. A prioritized process will run as long as it desires unless a process of higher priority becomes ready. If processes of equal priority are ready, one is choose by random. A process can wait for one or more specified signals to arrive. If none of them is present in its queue the process is swapped out and other process is allowed to run.

A process is swapped in if all processes on higher priority levels are not ready and the process itself is or become ready, for example when another process sends a message to it.

Background Processes: These are almost the same as the prioritized processes with one exception: if there is more than one ready background processes at a priority level. The running process will be interrupted by the OS after a time and the next background process will be allowed to run.

Interrupt Processes: An interrupt process is immediately called whenever the corresponding interrupt occurs, provided that no other interrupt of higher or equal hardware priority is running. An OS68 interrupt process will never become waiting; it is run from beginning to the end each time it is activated. It is impossible for an interrupt process to wait for a signal or make a delay.

OS68 timer interrupt processes are identical to interrupt processes for the manner in which they are called. For each time-interrupt process, the user selects a time to elapse between each call of the process. Timer-interrupt process should be short (e.g., execute during one system tick). Timer-interrupt processes have the same hardware priority as system timer-interrupts.

Questions:

1. Give a brief introduction about OS68.
2. What are the processes and states involved in OS68?

3.8 Summary

- Computer system can be divided into four components – hardware, operating system, the application program and the user.
- System call is the mechanism used by an application program to request services from the operating system.

- Operating system is an interface between your computer and the outside world.
- Multiuser is a term that defines an operating system or applications Software that allows concurrent access by multiple users of a computer.
- Utilities are often rather technical and targeted at people with an advanced level of computer knowledge.



Task

1. Configuring your Host Operating System's SNMP.
2. How we manage hardware in Operating Systems?

3.9 Keywords

Directory Access Permissions: A directory's access permissions help to control access to the files in it. These affect the overall ability to use files and subdirectories in the directory.

Driver: A driver is a specially written program which understands the operation of the device it interfaces to, such as a printer, video card, sound card or CD ROM drive.

File Access Permissions: The access permissions on a file control what can be done to the file's contents.

Graphical User Interfaces: Most of the modern computer systems support graphical user interfaces (GUI), and often include them. In some computer systems, such as the original implementation of Mac OS, the GUI is integrated into the kernel.

Multi-User: Is a term that defines an operating system or application software that allows concurrent access by multiple users of a computer.

Process Communication: Mutual exclusion of operations on shared variables makes it possible to make meaningful statements about the effect of concurrent computations.

Real-Time Operating System (RTOS): Real-time operating systems are used to control machinery, scientific instruments and industrial systems.

Single-User, Multitasking: This is the type of operating system most people use on their desktop and laptop computers today.

Single-User, Single Task: As the name implies, this operating system is designed to manage the computer so that one user can effectively do one thing at a time.

Supervisor and User Mode: In computer terms, supervisor mode is a hardware-mediated flag which can be changed by code running in system-level software.

System Calls: In computing, a **system call** is the mechanism used by an application program to request service from the operating system based on the monolithic kernel or to system servers on operating systems based on the microkernel-structure.

The Root Menu: If you move the pointer onto the root window (the "desktop" behind the windows) and press the correct mouse button (usually the first or third button, depending on your setup), you should see the *root menu*.

The xterm Window: One of the most important windows is an xterm window. xterm makes a terminal emulator window with a UNIX login session inside, just like a miniature terminal.

Notes

Utility Software: Kind of system software designed to help analyze, configure, optimize and maintain the computer. A single piece of utility software is usually called a utility (*abbr.* util) or tool.



Lab Exercise

1. Draw flow chart for components of a computer system.
2. Give flow chart of User and Supervisor modes.

3.10 Self-Assessment Questions

1. In computing, a _____ is the mechanism used by an application program to request service from the operating system based on the monolithic kernel or to system servers on operating systems based on the microkernel-structure.
2. _____ calls requires a control transfer which involves some sort of architecture-specific feature.
3. System calls can be roughly grouped into five major categories:
 - (a) Process Control
 - (b) System Control
 - (c) Debugger
 - (d) None of these
4. File management and Device management is used for system call that can be roughly grouped into major categories.
 - (a) True
 - (b) False
 - (c) None of these
5. A driver is not a written program which understands the operation of the device it interfaces to, such as a printer, video card, sound card or CD ROM drive.
 - (a) True
 - (b) False
 - (c) None of these
6. System tools (programs) used to monitor computer performance, debug problems, or maintain parts of the system.
 - (a) True
 - (b) False
 - (c) None of these
7. Real-time operating systems are used to control machinery, scientific instruments and industrial systems in machinery tools.
 - (a) True
 - (b) False
 - (c) None of these
8. Multi-user is a term that defines an operating system or application software that allows concurrent access by multiple users of a computer.
 - (a) True
 - (b) False
 - (c) None of these

Notes

9. Mutual exclusion of operations on shared variables makes it possible to make meaningful statements about the effect of concurrent computations.
 - (a) True
 - (b) False
 - (c) None of these
10. xterm makes a terminal emulator window with a LINUX login session inside, just like a miniature terminal.
 - (a) True
 - (b) False
 - (c) None of these

3.11 Review Questions

1. What is operating system? Give its types.
2. What is Supervisor and User mode in operating system?
3. Define System Calls. Give their types also.
4. What does driver mean in operating system? Briefly explain with their examples.
5. What are the operating system functions?
6. Differentiate between Single user single task and Single user multi task.
7. What are user interface in operating system?
8. Give definition of GUI.
9. What is setting of focus?
10. Define the xterm Window.
11. Define Root Menu.
12. What is sharing of files also give their types?
13. Give steps of Managing hardware in Operating Systems.
14. What are utility software's?
15. Define **Real-Time Operating System** (RTOS).
16. Give description how to run program in Operating system.

Answers for Self-Assessment Questions

1. System call
2. Implementing system
3. (a)
4. (a)
5. (b)
6. (a)
7. (b)
8. (a)
9. (a)
10. (b)

3.12 Further Reading



Books

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition.



Online link

<http://computer.howstuffworks.com/operating-system.htm>

Unit 4: Introduction of Networks

CONTENTS

Objectives

Introduction

4.1 Sharing Data any time any where

4.1.1 Sharing Data Over a Network

4.1.2 Saving Data to a Server

4.1.3 Opening Data from a Network Server

4.1.4 About Network Links

4.1.5 Creating a Network Link

4.2 Use of a Network

4.3 Types of Networks

4.3.1 Based on Server Division

4.3.2 Local Area Network

4.3.3 Personal Area Network

4.3.4 Home Area Network

4.3.5 Wide Area Network

4.3.6 Campus Network

4.3.7 Metropolitan Area Network

4.3.8 Enterprise Private Network

4.3.9 Virtual Private Network

4.3.10 Backbone Network

4.3.11 Global Area Network

4.3.12 Overlay Network

4.3.13 Network Classification

4.4 Summary

4.5 Keywords

4.6 Self-Assessment Questions

4.7 Review Questions

4.8 Further Reading

Objectives

Notes

After studying this unit, you will be able to:

- Explain sharing data any time any where.
- Discuss use of network.
- Explain common types of network.

Introduction

A computer network, often simply referred to as a network, is a collection of computers and devices interconnected by communications channels that facilitate communications among users and allows users to share resources. Networks may be classified according to a wide variety of characteristics. A computer network allows sharing of resources and information among interconnected devices.

4.1 Sharing Data any time any where

4.1.1 Sharing Data Over a Network

In addition to saving placemarks or folders to your local computer, you can also save place data to a web server or network server. Other Google Earth users who have access to the server can then use the data. As with other documents, you can create links or references to KMZ files for easy access. Storing a placemark file on the network or on a web server offers the following advantages:

4.1.1.1 Accessibility

If your place data is stored on a network or the Web, you can access it from any computer anywhere, provided the location is either publicly available or you have log in access.

4.1.1.2 Ease in Distribution

You can develop an extensive presentation folder for Google Earth software and make that presentation available to everyone who has access to your network storage location or web server. This is more convenient than sending the data via email when you want to make it persistently available to a large number of people.

4.1.1.3 Automatic Updates/Network Link Access

Any new information or changes you make to network-based KMZ information is automatically available to all users who access the KML data via a network link.

4.1.1.4 Backup

If for some reason the data on your local computer is corrupt or lost, you can open any of the KMZ files that you have saved to a network location, and if so desired, save it as a local file again.

4.1.2 Saving Data to a Server

To make your placemarks or folders available to other people via a server, you need to first save the file to the appropriate location.

Notes

4.1.2.1 Network Server

To save a folder or placemark to a location on your network, simply follow the steps in Saving Places Data and save the file in a location on your company network rather than to your local file system.

4.1.2.2 Web Server

To save a placemark or folder to a web server, first save the file to your local computer as described in Saving Places Data. Once the file is saved on your local computer as a separate KMZ file, you can use an FTP or similar utility to transfer the file to the web servers.

If you want users to be able to open KML and KMZ files from a web server, you may need to add MIME types for the server. These are:

- application/vnd.google-earth.kml+xml kml
- application/vnd.google-earth.kmz kmz

4.1.3 Opening Data from a Network Server

If you are working in an organization where place data is saved to a network that you have access to, you can open that data in the same way you would open a saved KMZ file on your local computer.

4.1.3.1 From the File menu, select Open (CTRL + O in Windows/Linux, + O on the Mac)

Navigate to your network places and locate the KMZ or KML data you want to open in Google Earth. Select the file and click the Open button. The folder or placemark appears in the 'Places' panel and the 3D viewer flies to the view set for the folder or placemark (if any).

Files opened in this way are NOT automatically saved for the next time you use Google Earth. If you want the placemark or folder to appear the next time you use Google Earth, drag the item to your 'My Places' folder to save it for the next session.

4.1.3.2 Locate the file you want to open

Once you have located the file on your network places, you can simply drag and drop the KMZ file over the 'Places' panel. The 3D viewer flies to the view set for the folder or placemark (if any).

When you use the drag-and-drop method of opening a placemark or folder, you can drop the item over a specific folder in the 'Places' panel. If the 'My Places' folder is closed and you want to drop it there, just hold the item over the 'My Places' folder until the folder opens up and you can place the item within subfolders or in the list. Items dropped in the 'My Places' folder appear the next time you start Google Earth. Otherwise, you can drop the item in the white space below the 'Places' panel so that it appears in the 'Temporary Places' folder. Items opened this way appear only for the current session of Google Earth unless you save them.

4.1.4 About Network Links

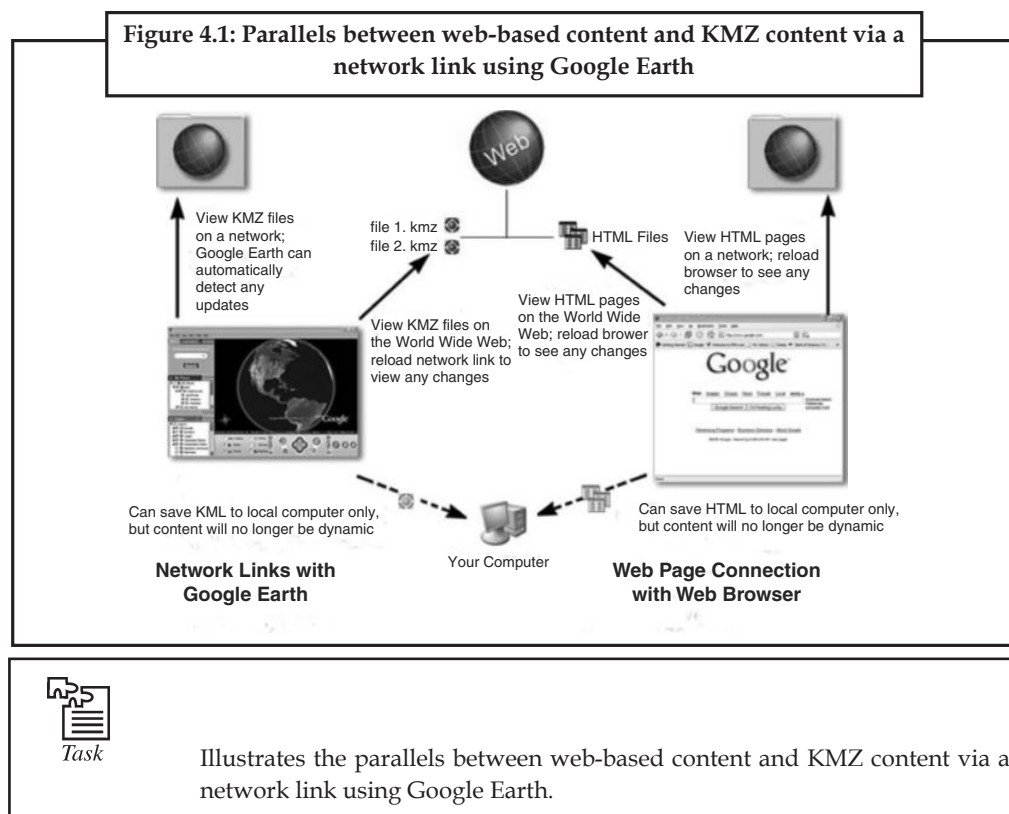
The network link feature in Google Earth provides a way for multiple clients to view the same network-based or web-based KMZ data and automatically see any changes to the content as those changes are made. A network link allows for content publishing in a manner similar to web page/web browser content delivery:

Notes

A network link provides a way to deliver dynamic data to multiple Google Earth users. When users connect to your KMZ file via a network link, either you or they can specify how often to refresh the data in the file. This way, regular updates made to the content by one person are automatically reflected in all connected clients.

A network link is intended as a view-only reference to published content. Just as web pages are viewed by many people but only modifiable by those with permission, place data content can be linked to and seen by multiple users while set to be modifiable only by one author. As with web pages, users who are viewing the content can always save that content to their hard drive, but they can only receive dynamic content via a network link.

The Figure 4.1 illustrates the parallels between web-based content and KMZ content via a network link using Google Earth.



4.1.5 Creating a Network Link

Before you can create a network link, the content you want to link to must exist on the network or web server that you are linking to. If you are authoring the content, if you are only linking to the content, be sure you know either the network location of the file or the URL if it is located on a web server.

1. Choose any one of the following ways to start:

Select **Network Link** From the 'Add' Menu. Select **Network Link** from the pop-up menu.

- Right-click a folder in the 'My Places' Panel. Select **Add > Network Link** from the pop-up menu. When you add a network link in this way, the selected folder is automatically set as the container for the network link.

Notes

The 'New Network Link' dialog box appears. Enter the name of your link in the 'Name' field. Enter the full path of the KMZ file in the 'Link' field, or browse to the file location if the file is located on a network. You can use a URL to reference the KMZ, such as 'http://www.test.com/myKMZ.kmz'. The 3D viewer immediately flies to the default view for the linked data.

2. Enter descriptive text or HTML. You can enter this data in the same way you would for a regular folder. See Editing Places and Folders for more information.
3. (Optional) Click the View tab to change the default view that this network link presents to users. Click Snapshot current view to use your current view in Google Earth. If your network link has more than one placemark, click Reset to display all these placemarks to users.



The text you enter in this description is your description of the link only and is not viewable by anyone else linking to the KMZ file. Only you will be able to see the description you enter here (unless you email your link folder to other people). This is similar to creating a bookmark for a web page and then adding a descriptive note about the bookmark.

4.2 Use of a Network

Connecting computers in a local area network lets people increase their efficiency by sharing files, resources, and more. Local area networking has attained much popularity in recent years – so much that it seems networking was just invented. In reality, local area networks (LANs) appeared more than ten years ago, when the arrival of the microcomputer gave multiple users access to the same computer.

- (a) These are three of the most common benefits of using a LAN.
- (b) Increased efficiency
- (c) Improved communications

4.3 Types of Networks

A network can be classified in following ways:

4.3.1 Based on Server Division

4.3.1.1 Peer-to-Peer Networking

This is a simple network configuration that requires some basic know-how to set up. Each of the interconnected machines share dual capability and responsibility on the network. That is to say, that each machine serves a dual purpose or role, i.e. they are both clients and servers to some extent.

The server capability of the machines is very basic. The services provided by each, is no more than the ability to share resources like files, folders, disk drives and printers. They even have the ability to share Internet access.

However, the server functionality of these machines stops there. They cannot grant any of the benefits mentioned previously, since these are functions provided only by a dedicated server operating system.

Because all machines on the network have equal status, hence the term peers, there is no centralised control over shared resources. Sharing is endorsed or repealed by each machine's user. Passwords can be assigned to each individual shared resource whether it is a file, folder, drive or peripheral, again done by the user.

Notes

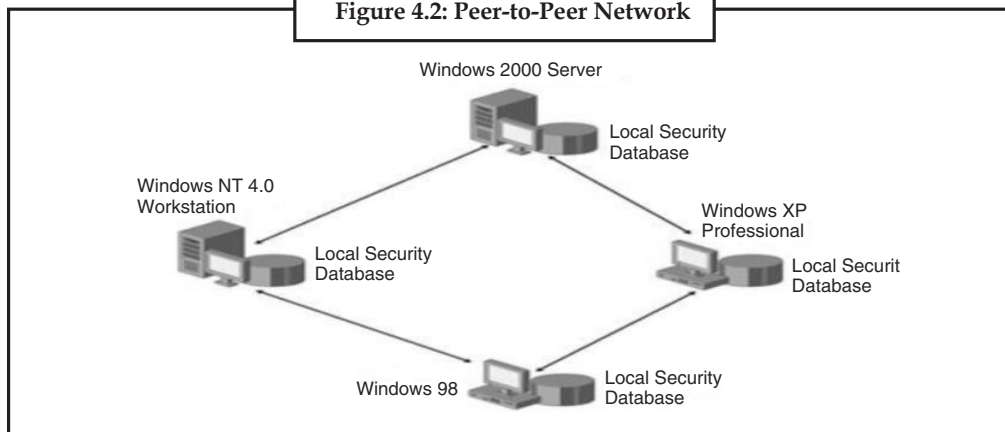
Although this solution is workable on small networks, it introduces the possibility that users may have to know and remember the passwords assigned to every resource, and then re-learn them if the user of a particular machine decides to change them! Due to this flexibility and individual discretion, institutionalised chaos is the norm for peer-to-peer networks.

Security can also be a major concern, because users may give passwords to other unauthorised users, allowing them to access areas of the network that the company does not permit. Furthermore, due to lack of centralisation, it is impossible for users to know and remember what data lives on what machine, and there are no restrictions to prevent them from over-writing the wrong files with older versions of the file. This of course cripples attempts to organise proper backups.

It may appear that peer-to-peer networks are hardly worthwhile. However, they offer some powerful incentives, particularly for smaller organisations. Networks of this type are the cheapest and easiest to install, requiring only Windows95, a network card for each machine and some cabling. Once connected, users can start to share information immediately and get access to devices.

As a result, networks of this type are not scalable and a limit of no more than 10 machines is the general rule.

Figure 4.2: Peer-to-Peer Network



Advantages

- Easy to install and configure.
- No dedicated server required.
- Users control their own resources.
- Inexpensive to purchase and operate.
- No specialist software required.
- No dedicated administrator to run the network required.

Disadvantages

- Difficult to employ security.
- Too many passwords for shared resources.
- Backups difficult to manage.
- No centralisation.
- Limited users.

Notes

4.3.1.2 Client/Server Networks

Server based networks, or client/server networks as they are properly called, has a machine at the heart of its operations called the server. A server is a machine that provides services over a network by responding to client requests. Servers rarely have individuals operating it, and even then, it is usually to install, configure or manage its capabilities. The server's essential role on the network is to be continuously available to handle the many requests generated by its clients.

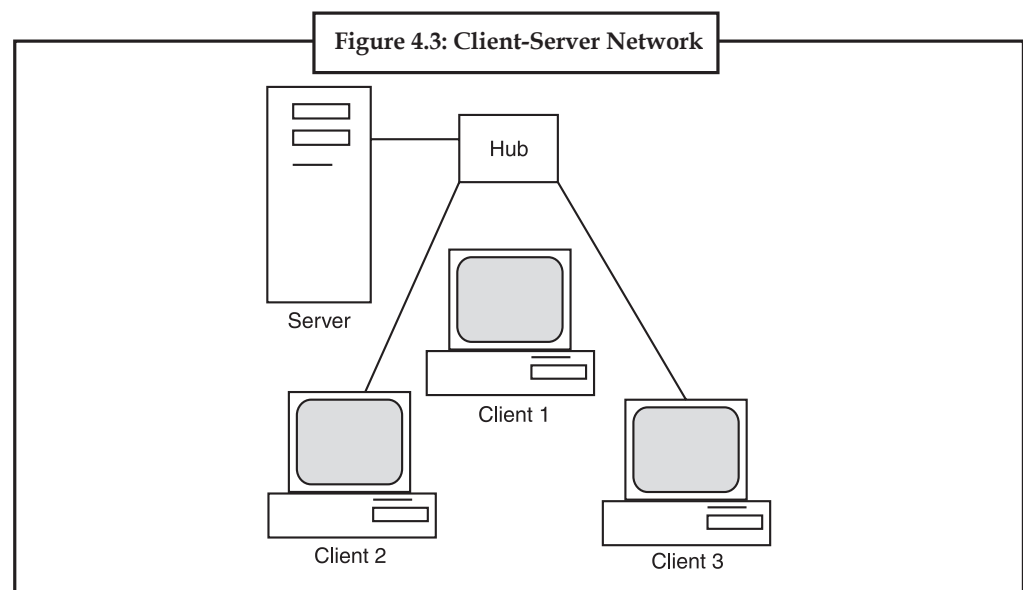
Server-based networks provide centralised control of the entire network environment. The computer systems used for this role are generally more powerful than end-user machines, incorporating faster CPUs, more memory, larger disk drives and other drive types installed, like a tape drive for backup purposes. These are required, because servers are dedicated to handling multiple simultaneous requests from their client communities.

Server based networks provide centralised verification of user accounts and passwords. Only valid account name and password combinations are allowed access to the network. Client/Server networks typically require a single login to the network itself, meaning that users need to remember long password lists to access various resources. Concentrations of resources on a single server, mean that they are easier to find, as opposed to the peer-to-peer model, where resources were distributed throughout the network since they were attached to multiple machines. The server being a central data repository, means that not only is data more accessible to users, but it also makes life much easier in terms of performing backups, since the data is in a location known to the administrator.

Server-based networks are easier to scale. Peer-to-peer networks bog down seriously as they grow beyond ten users, and seriously slow up with 20 users. On the other hand, client/server networks can handle a few users, up to a thousand users as such networks grow to keep pace with an organisation's growth and expansion.

Unlike peer-to-peer networks, client/server networks don't come cheap. The server machine itself may cost several thousands of pounds, along with the software to make it run; another thousand pounds. Because of the complex nature of this kind of networking environment, a dedicated administrator is required to be on site at all times to be involved in the day to day running of the network. Hiring an individual of this nature adds considerably to the cost of client/server networks.

Lastly, because the network's operability is so dependent upon the server, this introduces a single point of failure, if the server goes down the network goes down. There are measures available, that can legislate for such failures, and however these techniques add even more cost to this solution.



Advantages

- Centralised user accounts, security and access controls simplify network administration.
- More powerful equipment means more efficient access network resources.
- Single password login, means access to all resources.
- Supports greater numbers of users, or networks where resources are heavily used.

Notes**Disadvantages**

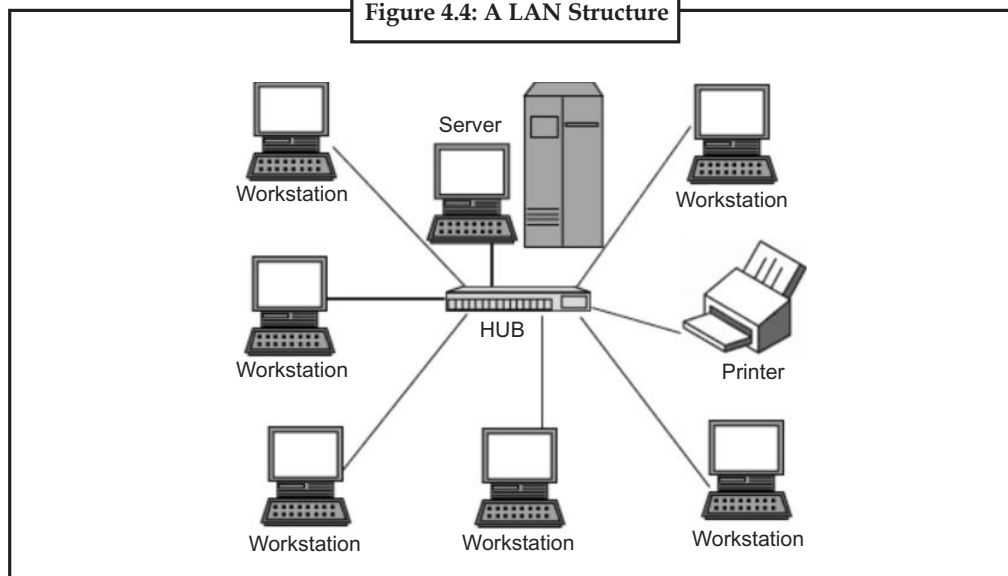
- More costly to install and maintain.
- Single point of failure, server goes down, the network goes down.
- Complex special-purpose software requires appointment of expert staff, increasing costs.
- Dedicated hardware and software increases costs.

4.3.2 Local Area Network

A local area network (LAN) is a network that connects computers and devices in a limited geographical area such as home, school, computer laboratory, office building, or closely positioned group of buildings. Each computer or device on the network is a node. Current wired LANs are most likely to be based on Ethernet technology, although new standards like ITU-T G.hn also provide a way to create a wired LAN using existing home wires (coaxial cables, phone lines and power lines).

All interconnected devices must understand the network layer (layer 3), because they are handling multiple subnets (the different colors). Those inside the library, which have only 10/100 Mbit/s Ethernet connections to the user device and a Gigabit Ethernet connection to the central router, could be called “layer 3 switches” because they only have Ethernet interfaces and must understand IP. It would be more correct to call them access routers, where the router at the top is a distribution router that connects to the Internet and academic networks’ customer access routers.

The defining characteristics of LANs, in contrast to WANs (Wide Area Networks), include their higher data transfer rates, smaller geographic range, and no need for leased telecommunication lines. Current Ethernet or other IEEE 802.3 LAN technologies operate at speeds up to 10 Gbit/s. This is the data transfer rate. IEEE has projects investigating the standardization of 40 and 100 Gbit/s.

Figure 4.4: A LAN Structure

Notes

4.3.3 Personal Area Network

A personal area network (PAN) is a computer network used for communication among computer and different information technological devices close to one person. Some examples of devices that are used in a PAN are personal computers, printers, fax machines, telephones, PDAs, scanners, and even video game consoles. A PAN may include wired and wireless devices. The reach of a PAN typically extends to 10 meters. A wired PAN is usually constructed with USB and Firewire connections while technologies such as Bluetooth and infrared communication typically form a wireless PAN.

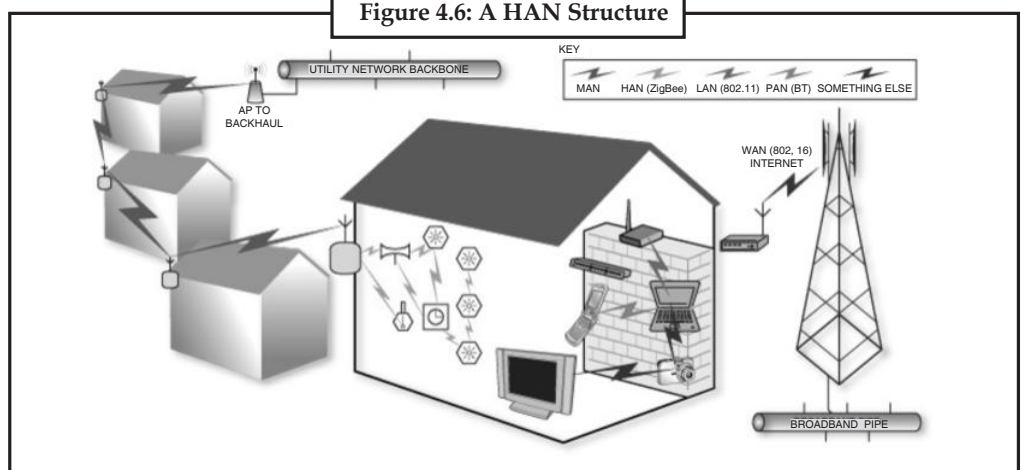
Figure 4.5: A PAN Structure



4.3.4 Home Area Network

A home area network (HAN) is a residential LAN which is used for communication between digital devices typically deployed in the home, usually a small number of personal computers and accessories, such as printers and mobile computing devices. An important function is the sharing of Internet access, often a broadband service through a CATV or Digital Subscriber Line (DSL) provider. It can also be referred to as an office area network (OAN).

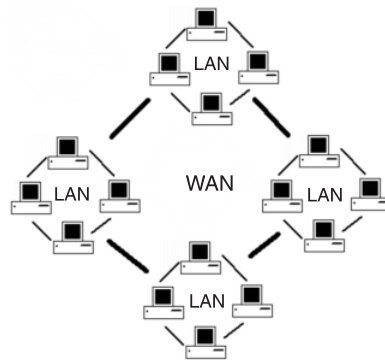
Figure 4.6: A HAN Structure



4.3.5 Wide Area Network

A wide area network (WAN) is a computer network that covers a large geographic area such as a city, country, or spans even intercontinental distances, using a communications channel that combines many types of media such as telephone lines, cables, and air waves. A WAN often uses transmission facilities provided by common carriers, such as telephone companies. WAN technologies generally function at the lower three layers of the OSI reference model: the physical layer, the data link layer, and the network layer.

Figure 4.7: A WAN Structure

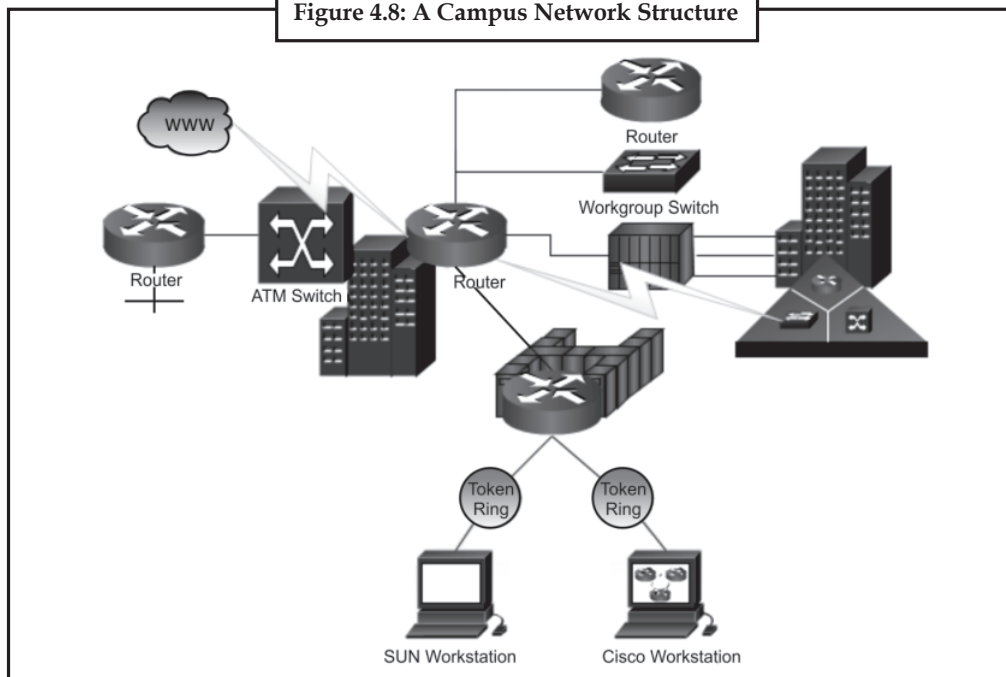


4.3.6 Campus Network

A campus network is a computer network made up of an interconnection of local area networks (LAN's) within a limited geographical area. The networking equipments (switches, routers) and transmission media (optical fiber, copper plant, Cat5 cabling etc.) are almost entirely owned (by the campus tenant / owner: an enterprise, university, government etc.).

In the case of a university campus-based campus network, the network is likely to link a variety of campus buildings including; academic departments, the university library and student residence halls.

Figure 4.8: A Campus Network Structure



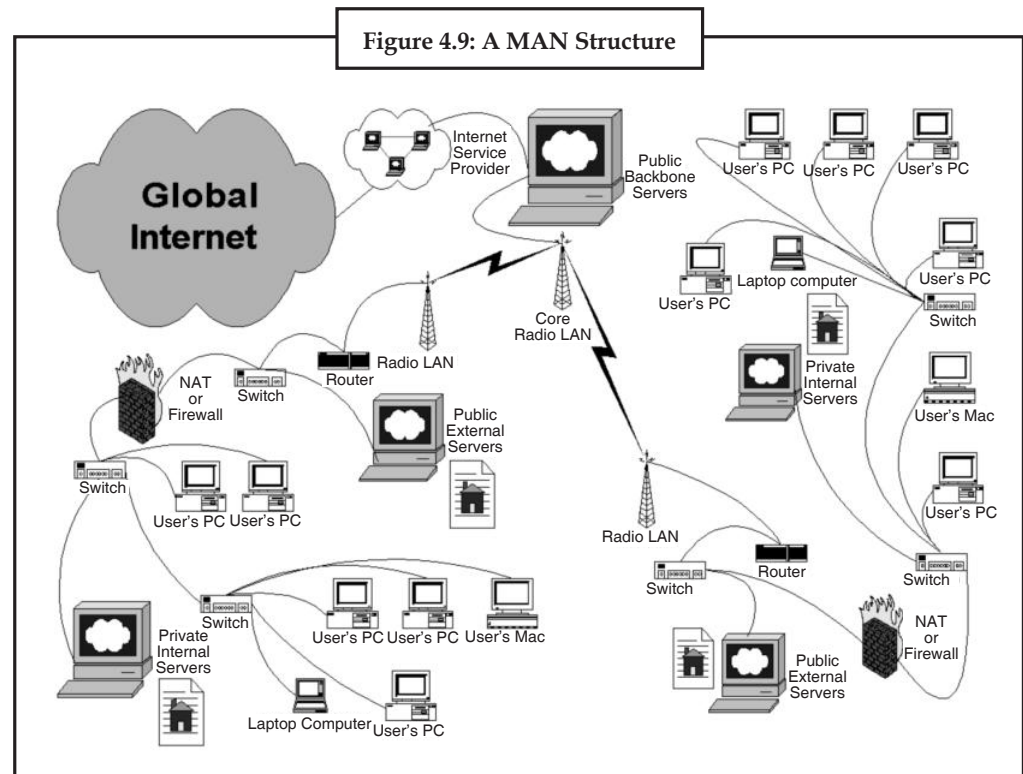
Notes

4.3.7 Metropolitan Area Network

A Metropolitan area network is a large computer network that usually spans a city or a large campus.

Sample EPN made of Frame relay WAN connections and dialup remote access.

Sample VPN used to interconnect three offices and remote users.

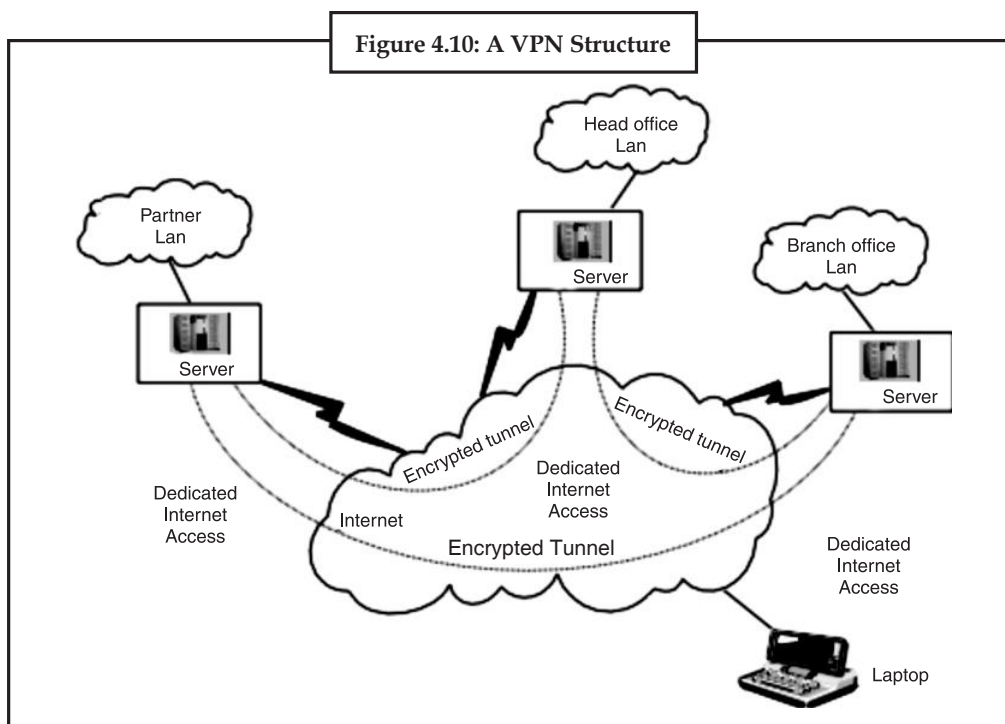
**4.3.8 Enterprise Private Network**

An enterprise private network is a network build by an enterprise to interconnect various company sites, e.g., production sites, head offices, remote offices, shops, in order to share computer resources.

4.3.9 Virtual Private Network

A virtual private network (VPN) is a computer network in which some of the links between nodes are carried by open connections or virtual circuits in some larger network (e.g., the Internet) instead of by physical wires. The data link layer protocols of the virtual network are said to be tunneled through the larger network when this is the case. One common application is secure communications through the public Internet, but a VPN need not have explicit security features, such as authentication or content encryption. VPNs, for example, can be used to separate the traffic of different user communities over an underlying network with strong security features.

VPN may have best-effort performance, or may have a defined service level agreement (SLA) between the VPN customer and the VPN service provider. Generally, a VPN has a topology more complex than point-to-point.



4.3.9.1 Internetwork

An internetwork is the connection of two or more private computer networks via a common routing technology (OSI Layer 3) using routers. The Internet is an aggregation of many internetworks, hence its name was shortened to Internet.

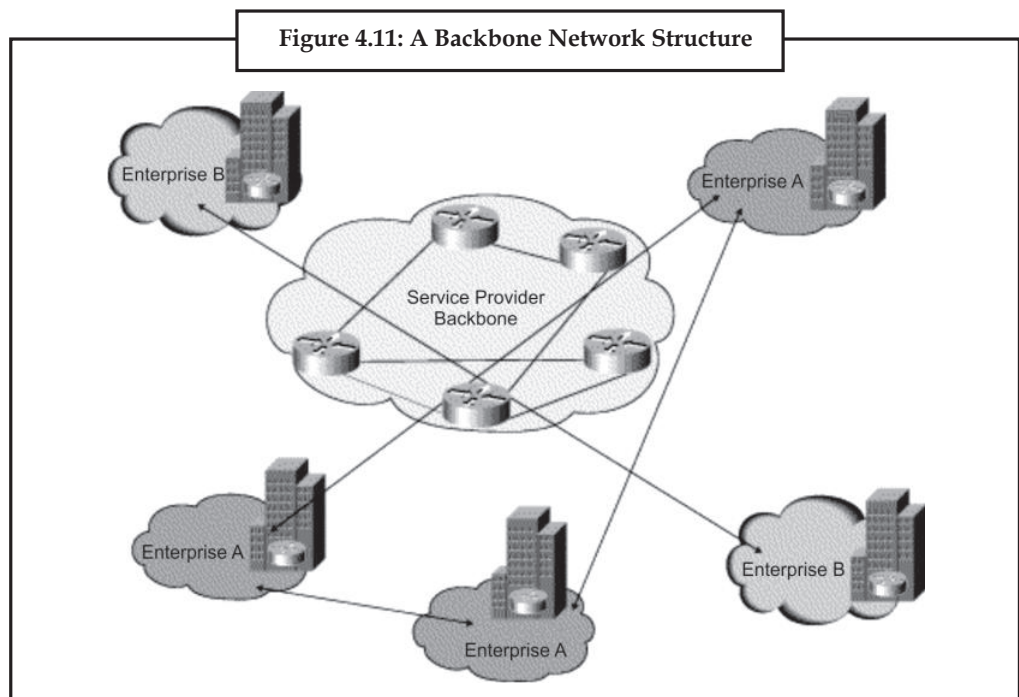
4.3.10 Backbone Network

A Backbone network (BBN) or network backbone is part of a computer network infrastructure that interconnects various pieces of network, providing a path for the exchange of information between different LANs or subnetworks. A backbone can tie together diverse networks in the same building, in different buildings in a campus environment, or over wide areas. Normally, the backbone's capacity is greater than the networks connected to it.

A large corporation that has many locations may have a backbone network that ties all of the locations together, for example, if a server cluster needs to be accessed by different departments of a company that are located at different geographical locations. The pieces of the network connections (for example: ethernet, wireless) that bring these departments together is often mentioned as network backbone. Network congestion is often taken into consideration while designing backbones.

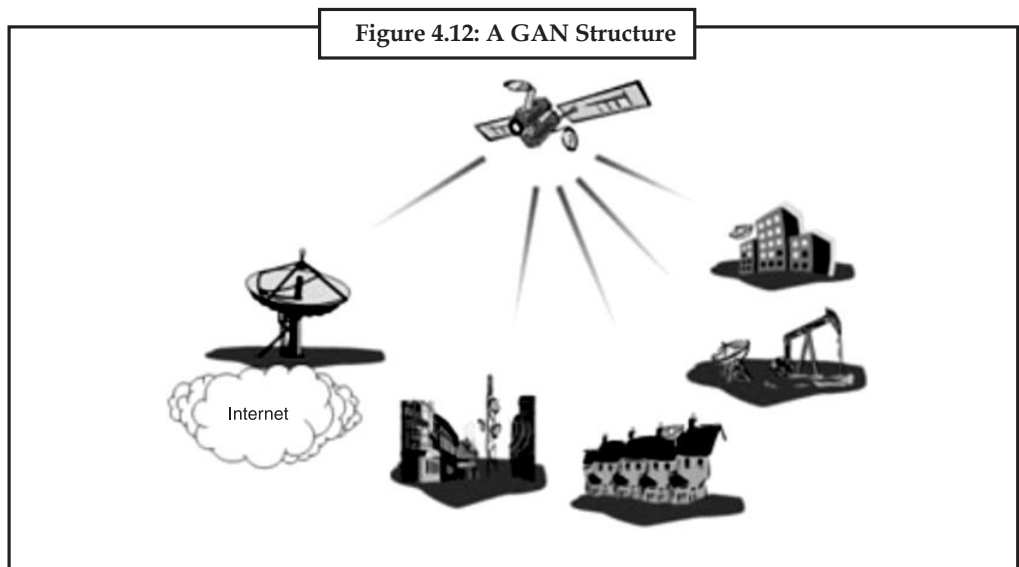
Backbone networks should not be confused with the Internet backbone.

Notes



4.3.11 Global Area Network

A global area network (GAN) is a network used for supporting mobile communications across an arbitrary number of wireless LANs, satellite coverage areas, etc. The key challenge in mobile communications is handing off the user communications from one local coverage area to the next. In IEEE Project 802, this involves a succession of terrestrial wireless LANs.



4.3.11.1 Internet

The Internet is a global system of interconnected governmental, academic, corporate, public, and private computer networks. It is based on the networking technologies of the Internet Protocol Suite. It is the successor of the Advanced Research Projects Agency Network (ARPANET)

developed by DARPA of the United States Department of Defense. The Internet is also the communications backbone underlying the World Wide Web (WWW).

Participants in the Internet use a diverse array of methods of several hundred documented, and often standardized, protocols compatible with the Internet Protocol Suite and an addressing system (IP addresses) administered by the Internet Assigned Numbers Authority and address registries. Service providers and large enterprises exchange information about the reachability of their address spaces through the Border Gateway Protocol (BGP), forming a redundant worldwide mesh of transmission paths.

4.3.11.2 Intranets and Extranets

Intranets and extranets are parts or extensions of a computer network, usually a local area network.

An intranet is a set of networks, using the Internet Protocol and IP-based tools such as web browsers and file transfer applications, that is under the control of a single administrative entity. That administrative entity closes the intranet to all but specific, authorized users. Most commonly, an intranet is the internal network of an organization. A large intranet will typically have at least one web server to provide users with organizational information.

An extranet is a network that is limited in scope to a single organization or entity and also has limited connections to the networks of one or more other usually, but not necessarily, trusted organizations or entities—a company's customers may be given access to some part of its intranet-while at the same time the customers may not be considered trusted from a security standpoint. Technically, an extranet may also be categorized as a CAN, MAN, WAN, or other type of network, although an extranet cannot consist of a single LAN; it must have at least one connection with an external network.

4.3.12 Overlay Network

An overlay network is a virtual computer network that is built on top of another network. Nodes in the overlay are connected by virtual or logical links, each of which corresponds to a path, perhaps through many physical links, in the underlying network.

A sample overlay network: IP over SONET over Optical.

For example, many peer-to-peer networks are overlay networks because they are organized as nodes of a virtual system of links run on top of the Internet. The Internet was initially built as an overlay on the telephone network .

Overlay networks have been around since the invention of networking when computer systems were connected over telephone lines using modem, before any data network existed.

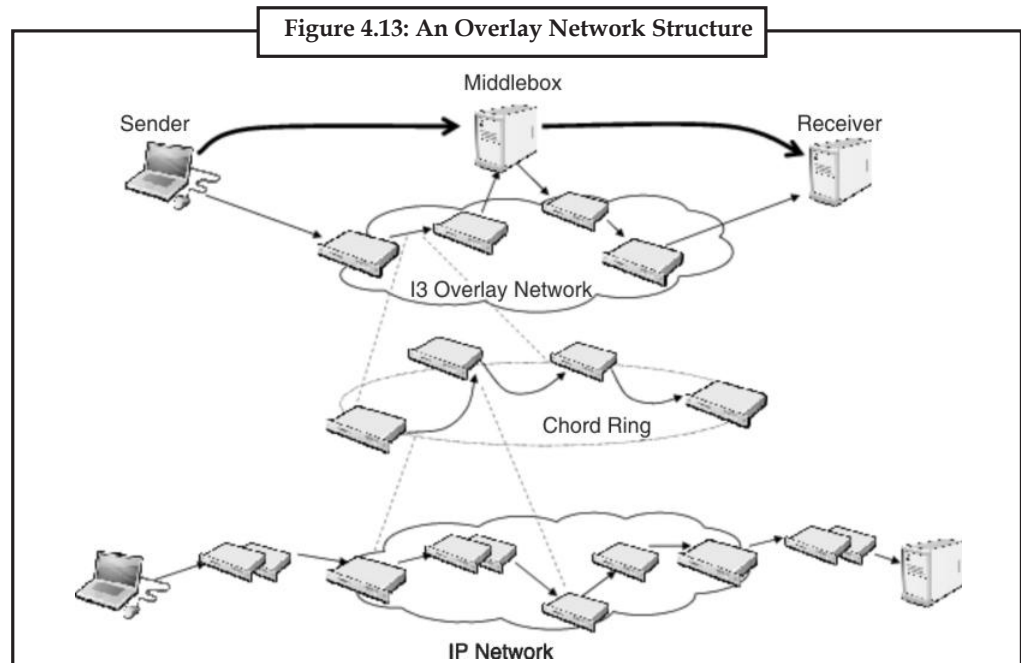
Nowadays the Internet is the basis for many overlaid networks that can be constructed to permit routing of messages to destinations specified by an IP address. For example, distributed hash tables can be used to route messages to a node having a specific logical address, whose IP address is known in advance.

Overlay networks have also been proposed as a way to improve Internet routing, such as through quality of service guarantees to achieve higher-quality streaming media. Previous proposals such as IntServ, DiffServ, and IP Multicast have not seen wide acceptance largely because they require modification of all routers in the network. On the other hand, an overlay network can be incrementally deployed on end-hosts running the overlay protocol software, without cooperation from Internet service providers. The overlay has no control over how packets are routed in the underlying network between two overlay nodes, but it can control, for example, the sequence of overlay nodes a message traverses before reaching its destination.

For example, Akamai Technologies manages an overlay network that provides reliable, efficient content delivery (a kind of multicast). Academic research includes End System Multicast and

Notes

Overcast for multicast; RON (Resilient Overlay Network) for resilient routing; and OverQoS for quality of service guarantees, among others. A backbone network or network backbone is a part of computer network infrastructure that interconnects various pieces of network, providing a path for the exchange of information between different LANs or subnetworks. A backbone can tie together diverse networks in the same building, in different buildings in a campus environment, or over wide areas. Normally, the backbone's capacity is greater than the networks connected to it.



4.3.13 Network Classification

The following list presents categories used for classifying networks.

4.3.13.1 Connection Method

Computer networks can be classified according to the hardware and software technology that is used to interconnect the individual devices in the network, such as optical fiber, Ethernet, wireless LAN, HomePNA, power line communication or G.hn.



Case Study

Success Story on Network Security

The Challenge: A new and upcoming network security server company designed a revolutionary secure server architecture. The company successfully sold the server to niche clients in the government and military space but wanted to expand to the general market. In addition, the vendor needed visibility into the enterprise space, as well as guidance on entry strategies. The company sought Frost & Sullivan's assistance in addressing these challenges.

The Process: Frost & Sullivan's market research and consulting teams leveraged its

Contd...

Notes

ongoing research on security solutions to create timely, actionable research services and simultaneously used its ongoing contacts with end-users to help our client address new markets. The Frost & Sullivan Network Security team was able to tap into the existing knowledge base to immediately analyze the different metrics and trends that would impact our client.

The Solution: The information in the subscription was leveraged to develop recommendations for market entrance strategies. The Frost & Sullivan team conducted interviews with a number of large end-users to determine where they were experiencing difficulties with their server security posture. Frost & Sullivan provided the client with analysis of the addressable market and then set up meetings for the client with potential end-users and partners, such as Rackspace and Nokia. Finally, Frost & Sullivan identified holes in the client's sales strategy and provided the client with sales training to improve its employee's ineffective techniques and establish ones that would be more effective.

Questions:

1. What kind of challenge they have to phase during securing network?
2. Explain the Solution.

Ethernet as it is defined by IEEE 802 utilizes various standards and mediums that enable communication between devices. Frequently deployed devices include hubs, switches, bridges or routers. Wireless LAN technology is designed to connect devices without wiring. These devices use radio waves or infrared signals as a transmission medium. ITU-T G.hn technology uses existing home wiring (coaxial cable, phone lines and power lines) to create a high-speed (up to 1 Gigabit/s) local area network.

4.3.13.2 Wired Technologies

- **Twisted pair wire** is the most widely used medium for telecommunication. Twisted-pair cabling consists of copper wires that are twisted into pairs. Ordinary telephone wires consist of two insulated copper wires twisted into pairs. Computer networking cabling consists of 4 pairs of copper cabling that can be utilized for both voice and data transmission. The use of two wires twisted together helps to reduce crosstalk and electromagnetic induction. The transmission speed ranges from 2 million bits per second to 100 million bits per second. Twisted pair cabling comes in two forms which are Unshielded Twisted Pair (UTP) and Shielded twisted-pair (STP) which are rated in categories which are manufactured in different increments for various scenarios.
- Coaxial cable is widely used for cable television systems, office buildings, and other work-sites for local area networks. The cables consist of copper or aluminum wire wrapped with insulating layer typically of a flexible material with a high dielectric constant, all of which are surrounded by a conductive layer. The layers of insulation help minimize interference and distortion. Transmission speed range from 200 million to more than 500 million bits per second.
- Optical fiber cable consists of one or more filaments of glass fiber wrapped in protective layers that carries a data by means of pulses of light. It transmits light which can travel over extended distances. Fiber-optic cables are not affected by electromagnetic radiation. Transmission speed may reach trillions of bits per second. The transmission speed of fiber optics is hundreds of times faster than for coaxial cables and thousands of times faster than a twisted-pair wire. A recent innovation in fiber-optic cable is the use of colored light. Instead of carrying one message in a stream of white light impulses, this technology can carry multiple signals in a single strand.

Notes

4.3.13.3 Wireless Technologies

- **Terrestrial microwave** - Terrestrial microwaves use Earth-based transmitter and receiver. The equipment looks similar to satellite dishes. Terrestrial microwaves use low-gigahertz range, which limits all communications to line-of-sight. Path between relay stations spaced approx, 30 miles apart. Microwave antennas are usually placed on top of buildings, towers, hills, and mountain peaks.
- **Communications satellites** - The satellites use microwave radio as their telecommunications medium which are not deflected by the Earth's atmosphere. The satellites are stationed in space, typically 22,000 miles (for geosynchronous satellites) above the equator. These Earth-orbiting systems are capable of receiving and relaying voice, data, and TV signals.
- **Cellular and PCS systems** - Use several radio communications technologies. The systems are divided to different geographic areas. Each area has a low-power transmitter or radio relay antenna device to relay calls from one area to the next area.
- **Wireless LANs** - Wireless local area network use a high-frequency radio technology similar to digital cellular and a low-frequency radio technology. Wireless LANs use spread spectrum technology to enable communication between multiple devices in a limited area. An example of open-standards wireless radio-wave technology is IEEE.
- **Infrared communication**, which can transmit signals between devices within small distances not more than 10 meters peer to peer or (face to face) without any body in the line of transmitting.

4.3.13.4 Scale

Networks are often classified as local area network (LAN), wide area network (WAN), metropolitan area network (MAN), personal area network (PAN), virtual private network (VPN), campus area network (CAN), storage area network (SAN), and others, depending on their scale, scope and purpose, e.g., controller area network (CAN) usage, trust level, and access right often differ between these types of networks. LANs tend to be designed for internal use by an organization's internal systems and employees in individual physical locations, such as a building, while WANs may connect physically separate parts of an organization and may include connections to third parties.

*Did u know?*

Network classification and Network types.

4.4 Summary

- A computer network, often simply referred to as a network, is a collection of computers and devices interconnected by communication channels that facilitate communications among users and allows users to share resources.
- A data can be saved on the network so that other users who access the network can share the data from network.
- The network link feature in Google Earth provides a way for multiple clients to view the same network based or web-based KML data and automatically see any changes to the content as those changes are made.
- Connecting computers in a local area network lets people increase their efficiency by sharing files, resources and more.

- Networks are often classified as local area network (LAN) wide area network (WAN), metropolitan area network (MAN), personal area network (PAN), virtual private network (VPN), campus area network (CAN) etc.

4.5 Keywords

Campus network: A campus network is a computer network made up of an interconnection of local area networks (LAN's) within a limited geographical area.

Coaxial cable: Coaxial cable is widely used for cable television systems, office buildings, and other work-sites for local area networks.

Ease in Distribution: You can develop an extensive presentation folder for Google Earth software and make that presentation available to everyone who has access to your network storage location or web server. This is more convenient than sending the data via email when you want to make it persistently available to a large number of people.

Global area network (GAN): A global area network (GAN) is a network used for supporting mobile communications across an arbitrary number of wireless LANs, satellite coverage areas, etc.

Home area network (HAN): A home area network (HAN) is a residential LAN which is used for communication between digital devices typically deployed in the home, usually a small number of personal computers and accessories.

Local area network (LAN): A local area network (LAN) is a network that connects computers and devices in a limited geographical area such as home, school, computer laboratory, office building, or closely positioned group of buildings.

Metropolitan area network: A Metropolitan area network is a large computer network that usually spans a city or a large campus.

Personal area network (PAN): A personal area network (PAN) is a computer network used for communication among computer and different information technological devices close to one person.

Wide area network (WAN): A wide area network (WAN) is a computer network that covers a large geographic area such as a city, country, or spans even intercontinental distances, using a communications channel that combines many types of media.



Lab Exercise

Draw Parallels between web-based content and KMZ content via a network link using Google Earth.

4.6 Self-Assessment Questions

1. An autoassociative network is:
 - (a) a neural network that contains no loops
 - (b) a neural network that contains feedback
 - (c) a neural network that has only one loop
 - (d) all of these

Notes

2. Which of the following is true?
 - (i) On average, neural networks have higher computational rates than conventional computers.
 - (ii) Neural networks learn by example.
 - (iii) Neural networks mimic the way the human brain works.
 - (a) all of them are true
 - (b) (ii) and (iii) are true
 - (c) (i), (ii) and (iii) are true
 - (d) none of these
3. Which of the following is true?

Single layer associative neural networks do not have the ability to:

 - (i) perform pattern recognition
 - (ii) find the parity of a picture
 - (iii) determine whether two or more shapes in a picture are connected or not
 - (a) (ii) and (iii) are true
 - (b) (ii) is true
 - (c) all of them are true
 - (d) all of these
4. LANs decrease the efficiency of workers by letting them exchange data and by eliminating redundant effort.
 - (a) True
 - (b) False

4.7 Review Questions

1. Explain the Network Operating Systems.
2. What is (Wireless / Computer) Networking?
3. Explain network interface card.
4. What is Twisted-pair cable? Explain with suitable examples.
5. Explain the most common benefits of using a LAN.
6. Explain Common types of computer networks.
7. What is hierarchy network?
8. Explain Media Network and its types.

Answers for Self-Assessment Questions

1. (b) 2. (a) 3. (a) 4. (a)

4.8 Further Reading



Books

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition.



Online link

<http://www.networkcomputing.com/> -

Unit 5: Operations of Network

Notes

CONTENTS

Objectives

5.1 Network Structure

5.1.1 Network Architecture

5.1.2 OSI Model

5.1.3 TCP/IP Model

5.2 Network Topology

5.2.1 Basic Topology Types

5.2.2 Classification of Network Topologies

5.3 Network Media

5.3.1 Twisted-Pair Cable

5.3.2 Shielded Twisted-Pair Cable

5.4 Basic Hardware

5.4.1 Network Interface Cards

5.4.2 Repeaters

5.4.3 Bridges

5.4.4 Switches

5.4.5 Routers

5.4.6 Firewalls

5.5 Summary

5.6 Keywords

5.7 Self-Assessment Questions

5.8 Review Questions

5.9 Further Reading

Notes

Objectives

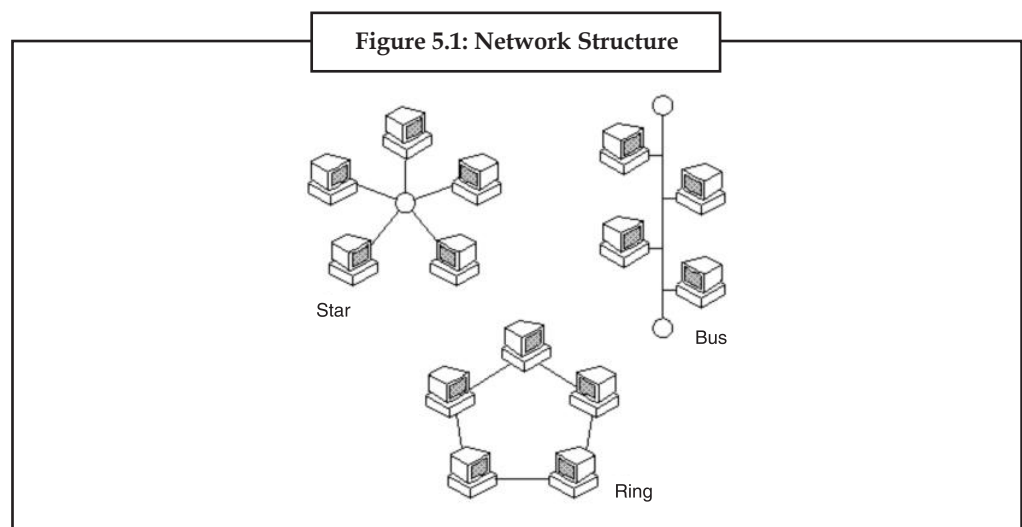
After studying this unit, you will be able to:

- Understanding network structure.
- Discuss network architecture.
- Explain network topologies.
- Explain network protocols.
- Discuss network media.
- Understand network media.

5.1 Network Structure

Networks are usually classified using three properties: Topology, Protocol, and Architecture.

Topology specifies the geometric arrangement of the network. Common topologies are a bus, ring, and star. A bus topology means that each computer on the network is attached to a common central cable, called a bus or backbone. This is a rather simple network to set up. Ethernets use this topology. A ring topology means that each computer is connected to two others, and they are arranged in a ring shape. These are difficult to set up, but offer high bandwidth. A star topology means all computers on the network are connected to a central hub. These are easy to set up, but bottlenecks can occur because all data must pass through the hub. You can consult the diagram below to see these three topologies:



Protocol specifies a common set of rules and signals the computers on the network use to communicate. There are many protocols, each having advantages over others. Let's run through the common ones:

- **TCP/IP** : Transmission Control Protocol / Internet Protocol. This was originally developed by the Defense Department of the US to allow dissimilar computers to talk. Today, as many of us know, this protocol is used as the basis for the internet. Because it must span such large distances and cross multiple, smaller networks, TCP/IP is a routable protocol, meaning it can send data through a router on its way to its destination. In the long run, this slows things down a little, but this ability makes it very flexible for large networks.

- **IPX/SPX:** Developed by Novell for use with its NetWare NOS, but Microsoft built compatibility into both NT and Windows 9x. IPX is like an optimized TCP/IP. It, too, is a routable protocol, making it handy for large networks, but it allows quicker access over the network than TCP/IP. The downfall is that it doesn't work well over analog phone lines. IPX continually checks the status of transmission to be sure all the data arrives. This requires extra bandwidth, where analog phone lines don't have much to begin with. This results in slow access. Of course, the data is more reliable with IPX.
- **NetBEUI:** Designed for small LANs, this protocol developed by Microsoft is quite fast. It lacks the addressing overhead of TCP/IP and IPX, which means it can only be used on LANs. You cannot access networks via a router.

Architecture refers to one of the two major types of network architecture.

Peer-to-peer or client/server. In a Peer-to-Peer networking configuration, there is no server, and computers simply connect with each other in a workgroup to share files, printers, and Internet Access. This is most commonly found in home configurations, and is only practical for workgroups of a dozen or less computers. In a client/server network, there is usually an NT Domain Controller, which all of the computers log on to. This server can provide various services, including centrally routed Internet Access, mail (including e-mail), file sharing, and printer access, as well as ensuring security across the network. This is most commonly found in corporate configurations, where network security is essential.

Now that you have a basic understanding of networks, we'll learn about the type of network most people will want to setup, a Local-Area Network.

5.1.1 Network Architecture

What is Network Architecture?

A network architecture is a blueprint of the complete computer communication network, which provides a framework and technology foundation for designing, building and managing a communication network. It typically has a layered structure. Layering is a modern network design principle which divides the communication tasks into a number of smaller parts, each part accomplishing a particular sub-task and interacting with the other parts in a small number of well-defined ways. Layering allows the parts of a communication to be designed and tested without a combinatorial explosion of cases, keeping each design relatively simple.

If a network architecture diagram is open, no single vendor owns the technology and controls its definition and development. Anyone is free to design hardware and software based on the network architecture. The TCP/IP network architecture, which the Internet is based on, is such a open network architecture and it is adopted as a worldwide network standard and widely deployed in local area network (LAN), wide area network (WAN), small and large enterprises, and last but not the least, the Internet.

The network architecture and design specialization will help you gain the technical leadership skills you need to design and implement high-quality networks that support business needs. You will learn how to design, maintain, and troubleshoot Internet, intranet, and extranet connections, including local- and wide-area networks. This specialization will also build your knowledge of developing security and disaster recovery plans. Upon completion of this online degree, you will be well positioned to assume a senior management or team leader role in network management.

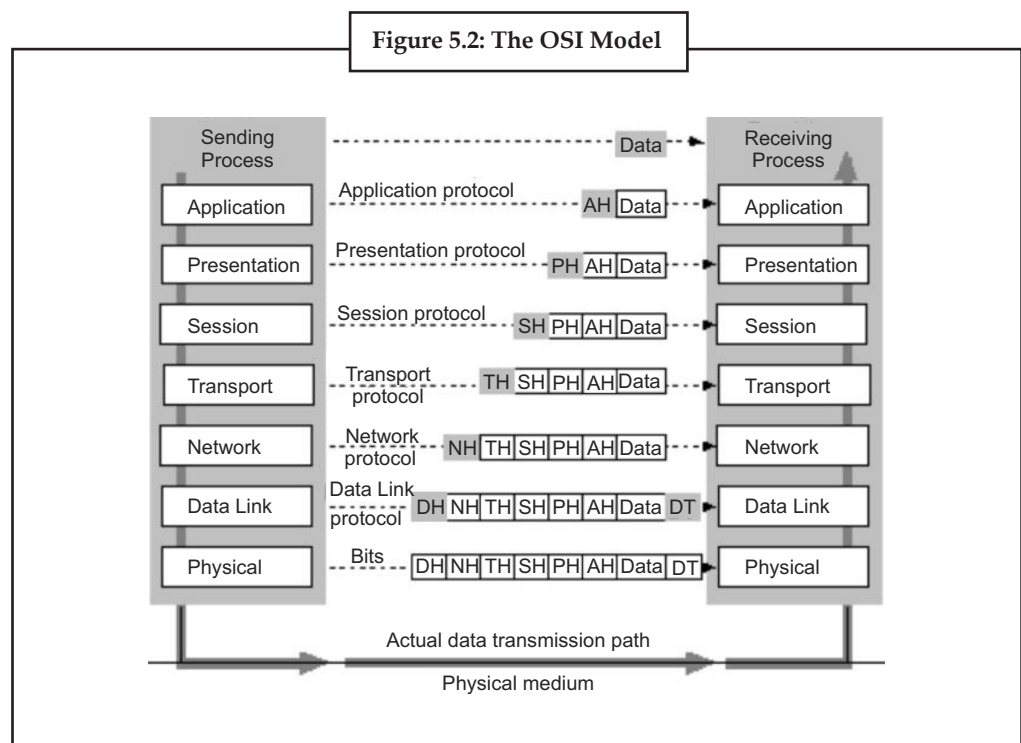
Notes

5.1.2 OSI Model

Open Systems Interconnection (OSI) network architecture, developed by International Organization for Standardization, is an open standard for communication in the network across different equipment and applications by different vendors. Though not widely deployed, the OSI 7 layer model is considered the primary network architectural model for inter-computing and inter-networking communications.

In addition to the OSI network architecture model, there exist other network architecture models by many vendors, such as IBM SNA (Systems Network Architecture), Digital Equipment Corporation (DEC; now part of HP) DNA (Digital Network Architecture), Apple computers AppleTalk, and Novells NetWare.

Network architecture provides only a conceptual framework for communications between computers. The model itself does not provide specific methods of communication. Actual communication is defined by various communication protocols.

**Layer 1: Physical**

Physical layer defines the cable or physical medium itself, e.g., thinnet, thicknet, unshielded twisted pairs (UTP). All media are functionally equivalent. The main difference is in convenience and cost of installation and maintenance. Converters from one media to another operate at this level.

Layer 2: Data Link

Data Link layer defines the format of data on the network. A network data frame, aka packet, includes checksum, source and destination address, and data. The largest packet that can be sent through a data link layer defines the Maximum Transmission Unit (MTU). The data link layer handles the physical and logical connections to the packet's destination, using a network interface. A host connected to an Ethernet would have an Ethernet interface to handle connections to the outside world, and a loopback interface to send packets to itself.

Notes

Ethernet addresses a host using a unique, 48-bit address called its Ethernet address or Media Access Control (MAC) address. MAC addresses are usually represented as six colon-separated pairs of hex digits, e.g., 8:0:20:11:ac:85. This number is unique and is associated with a particular Ethernet device. Hosts with multiple network interfaces should use the same MAC address on each. The data link layer's protocol-specific header specifies the MAC address of the packet's source and destination. When a packet is sent to all hosts (broadcast), a special MAC address (ff:ff:ff:ff:ff:ff) is used.

Layer 3: Network

NFS uses Internetwork Protocol (IP) as its network layer interface. IP is responsible for routing, directing datagrams from one network to another. The network layer may have to break large datagrams, larger than MTU, into smaller packets and host receiving the packet will have to reassemble the fragmented datagram. The Internetwork Protocol identifies each host with a 32-bit IP address. IP addresses are written as four dot-separated decimal numbers between 0 and 255, e.g., 129.79.16.40. The leading 1-3 bytes of the IP identify the network and the remaining bytes identifies the host on that network. The network portion of the IP is assigned by InterNIC Registration Services, under the contract to the National Science Foundation, and the host portion of the IP is assigned by the local network administrators. For large sites, the first two bytes represents the network portion of the IP, and the third and fourth bytes identify the subnet and host respectively.

Even though IP packets are addressed using IP addresses, hardware addresses must be used to actually transport data from one host to another. The Address Resolution Protocol (ARP) is used to map the IP address to its hardware address.

Layer 4: Transport

Transport layer subdivides user-buffer into network-buffer sized datagrams and enforces desired transmission control. Two transport protocols, Transmission Control Protocol (TCP) and User Datagram Protocol (UDP), sits at the transport layer. Reliability and speed are the primary difference between these two protocols. TCP establishes connections between two hosts on the network through 'sockets' which are determined by the IP address and port number. TCP keeps track of the packet delivery order and the packets that must be resent. Maintaining this information for each connection makes TCP a stateful protocol. UDP on the other hand provides a low overhead transmission service, but with less error checking. NFS is built on top of UDP because of its speed and statelessness. Statelessness simplifies the crash recovery.

Layer 5: Session

The session protocol defines the format of the data sent over the connections. The NFS uses the Remote Procedure Call (RPC) for its session protocol. RPC may be built on either TCP or UDP. Login sessions uses TCP whereas NFS and broadcast use UDP.

Layer 6: Presentation

External Data Representation (XDR) sits at the presentation level. It converts local representation of data to its canonical form and vice versa. The canonical uses a standard byte ordering and structure packing convention, independent of the host.

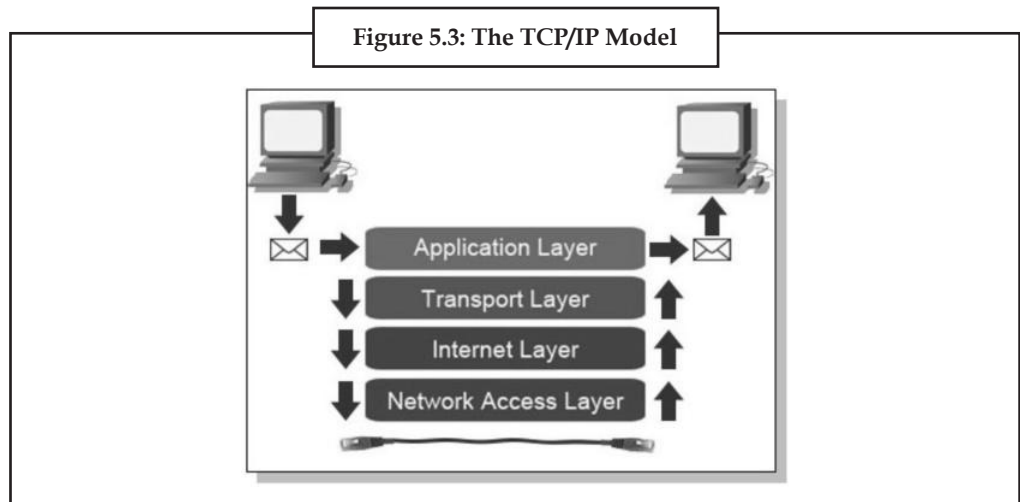
Layer 7: Application

Provides network services to the end-users. Mail, ftp, telnet, DNS, NIS, NFS are examples of network applications.

Notes

5.1.3 TCP/IP Model

TCP/IP (Transmission Control Protocol/Internet Protocol) is the basic communication language or protocol of the Internet. It can also be used as a communications protocol in a private network (either an intranet or an extranet). When you are set up with direct access to the Internet, your computer is provided with a copy of the TCP/IP program just as every other computer that you may send messages to or get information from also has a copy of TCP/IP.



TCP/IP is a two-layer program. The higher layer, Transmission Control Protocol, manages the assembling of a message or file into smaller packets that are transmitted over the Internet and received by a TCP layer that reassembles the packets into the original message. The lower layer, Internet Protocol, handles the address part of each packet so that it gets to the right destination. Each gateway computer on the network checks this address to see where to forward the message. Even though some packets from the same message.

TCP/IP uses the client/server model of communication in which a computer user (a client) requests and is provided a service (such as sending a Web page) by another computer (a server) in the network. TCP/IP communication is primarily point-to-point, meaning each communication is from one point (or host computer) in the network to another point or host computer. TCP/IP and the higher-level applications that use it are collectively said to be “stateless” because each client request is considered a new request unrelated to any previous one (unlike ordinary phone conversations that require a dedicated connection for the call duration). Being stateless frees network paths so that everyone can use them continuously. (Note that the TCP layer itself is not stateless as far as any one message is concerned. Its connection remains in place until all packets in a message have been received.)

Many Internet users are familiar with the even higher layer application protocols that use TCP/IP to get to the Internet. These include the World Wide Web’s Hypertext Transfer Protocol (HTTP), the File Transfer Protocol (FTP), Telnet (Telnet) which lets you logon to remote computers, and the Simple Mail Transfer Protocol (SMTP). These and other protocols are often packaged together with TCP/IP as a “suite.”

Personal computer users with an analog phone modem connection to the Internet usually get to the Internet through the Serial Line Internet Protocol (SLIP) or the Point-to-Point Protocol (PPP). These protocols encapsulate the IP packets so that they can be sent over the dial-up phone connection to an access provider’s modem.

Protocols related to TCP/IP include the User Datagram Protocol (UDP), which is used instead of TCP for special purposes. Other protocols are used by network host computers for exchanging router information. These include the Internet Control Message Protocol (ICMP), the Interior Gateway Protocol (IGP), the Exterior Gateway Protocol (EGP), and the Border Gateway Protocol (BGP).

5.2 Network Topology

Network topology is the layout pattern of interconnections of the various elements (links, nodes, etc.) of a computer network. Network topologies may be physical or logical. Physical topology means the physical design of a network including the devices, location and cable installation. Logical topology refers to how data is actually transferred in a network as opposed to its physical design. In general physical topology relates to a core network whereas logical topology relates to basic network.

Topology can be considered as a virtual shape or structure of a network. This shape does not correspond to the actual physical design of the devices on the computer network. The computers on a home network can be arranged in a circle but it does not necessarily mean that it represents a ring topology.

Any particular network topology is determined only by the graphical mapping of the configuration of physical and/or logical connections between nodes. The study of network topology uses graph theory. Distances between nodes, physical interconnections, transmission rates, and/or signal types may differ in two networks and yet their topologies may be identical.

A local area network (LAN) is one example of a network that exhibits both a physical topology and a logical topology. Any given node in the LAN has one or more links to one or more nodes in the network and the mapping of these links and nodes in a graph results in a geometric shape that may be used to describe the physical topology of the network. Likewise, the mapping of the data flow between the nodes in the network determines the logical topology of the network. The physical and logical topologies may or may not be identical in any particular network.

5.2.1 Basic Topology Types

The study of network topology recognizes seven basic topologies:

- Point-to-point topology
- Bus (point-to-multipoint) topology
- Star topology
- Ring topology
- Tree topology
- Mesh topology
- Hybrid topology

This classification is based on the interconnection between computers – be it physical or logical. The physical topology of a network is determined by the capabilities of the network access devices and media, the level of control or fault tolerance desired, and the cost associated with cabling or telecommunications circuits. Networks can be classified according to their physical span as follows:

- LANs (Local Area Networks)
- Building or campus internetworks
- Wide area internetworks

Notes

5.2.2 Classification of Network Topologies

There are also two basic categories of network topologies:

- (a) Physical topologies
- (b) Logical topologies

The shape of the cabling layout used to link devices is called the physical topology of the network. This refers to how the cables are laid out to connect many computers to one network. The physical topology you choose for your network influences and is influenced by several factors:

- (a) Office Layout
- (b) Troubleshooting Techniques
- (c) Cost of Installation
- (d) Type of cable used

Logical topology describes the way in which a network transmits information from network/computer to another and not the way the network looks or how it is laid out. The logical layout also describes the different speeds of the cables being used from one network to another.

5.2.2.1 Physical Topologies

The mapping of the nodes of a network and the physical connections between them the layout of wiring, cables, the locations of nodes, and the interconnections between the nodes and the cabling or wiring system.

5.2.2.2 Classification of Physical Topologies

Point-to-point

The simplest topology is a permanent link between two endpoints (the line in the illustration above). Switched point-to-point topologies are the basic model of conventional telephony. The value of a permanent point-to-point network is the value of guaranteed, or nearly so, communications between the two endpoints. The value of an on-demand point-to-point connection is proportional to the number of potential pairs of subscribers, and has been expressed as Metcalfe's Law.

Permanent (Dedicated)

Easiest to understand, of the variations of point-to-point topology, is a point-to-point communications channel that appears, to the user, to be permanently associated with the two endpoints. A children's "tin-can telephone" is one example, with a microphone to a single public address speaker is another. These are examples of physical dedicated channels.

Within many switched telecommunications systems, it is possible to establish a permanent circuit. One example might be a telephone in the lobby of a public building, which is programmed to ring only the number of a telephone dispatcher. "Nailing down" a switched connection saves the cost of running a physical circuit between the two points. The resources in such a connection can be released when no longer needed, for example, a television circuit from a parade route back to the studio.

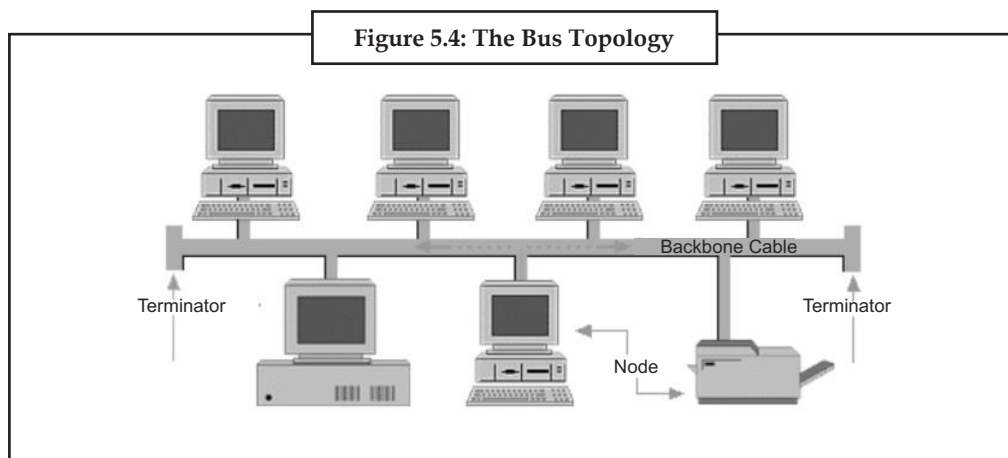
Switched

Using circuit-switching or packet-switching technologies, a point-to-point circuit can be set up dynamically, and dropped when no longer needed. This is the basic mode of conventional telephony.

Bus Network Topology

Notes

In local area networks where bus topology is used, each machine is connected to a single cable. Each computer or server is connected to the single bus cable through some kind of connector. A terminator is required at each end of the bus cable to prevent the signal from bouncing back and forth on the bus cable. A signal from the source travels in both directions to all machines connected on the bus cable until it finds the MAC address or IP address on the network that is the intended recipient. If the machine address does not match the intended address for the data, the machine ignores the data. Alternatively, if the data does match the machine address, the data is accepted. Since the bus topology consists of only one wire, it is rather inexpensive to implement when compared to other topologies. However, the low cost of implementing the technology is offset by the high cost of managing the network. Additionally, since only one cable is utilized, it can be the single point of failure. If the network cable breaks, the entire network will be down.



Linear Bus

The type of network topology in which all of the nodes of the network are connected to a common transmission medium which has exactly two endpoints (this is the 'bus', which is also commonly referred to as the backbone, or trunk) - all data that is transmitted between nodes in the network is transmitted over this common transmission medium and is able to be received by all nodes in the network virtually simultaneously (disregarding propagation delays).



The two endpoints of the common transmission medium are normally terminated with a device called a terminator that exhibits the characteristic impedance of the transmission medium and which dissipates or absorbs the energy that remains in the signal to prevent the signal from being reflected or propagated back onto the transmission medium in the opposite direction, which would cause interference with and degradation of the signals on the transmission medium (See Electrical termination).

Distributed Bus

The type of network topology in which all of the nodes of the network are connected to a common transmission medium which has more than two endpoints that are created by adding branches to the main section of the transmission medium - the physical distributed bus topology functions in exactly the same fashion as the physical linear bus topology (i.e., all nodes share a common transmission medium).

Notes



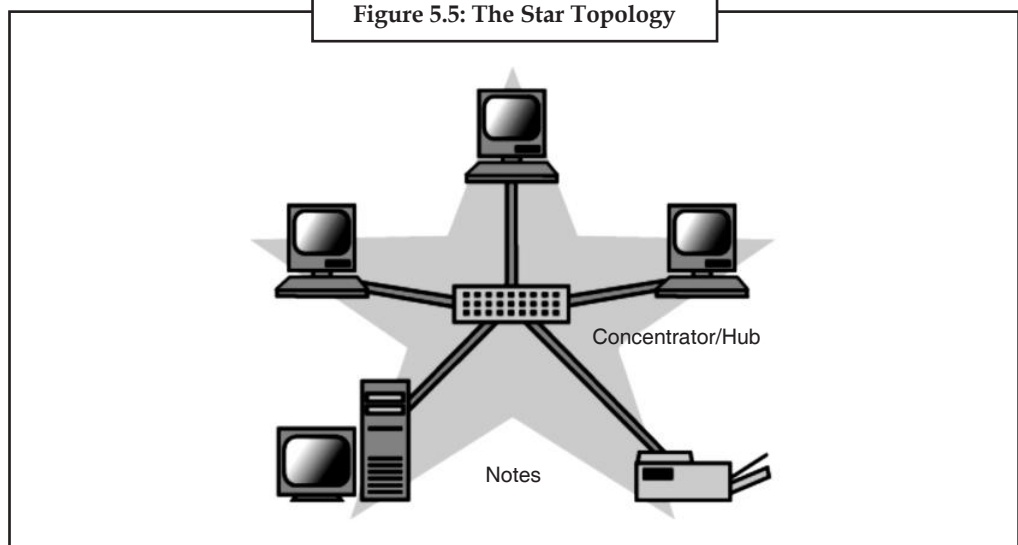
Notes

- (1) All of the endpoints of the common transmission medium are normally terminated with a device called a 'terminator'.
- (2) The physical linear bus topology is sometimes considered to be a special case of the physical distributed bus topology – i.e., a distributed bus with no branching segments.
- (3) The physical distributed bus topology is sometimes incorrectly referred to as a physical tree topology – however, although the physical distributed bus topology resembles the physical tree topology, it differs from the physical tree topology in that there is no central node to which any other nodes are connected, since this hierarchical functionality is replaced by the common bus.

Star Network Topology

In local area networks with a star topology, each network host is connected to a central hub. In contrast to the bus topology, the star topology connects each node to the hub with a point-to-point connection. All traffic that traverses the network passes through the central hub. The hub acts as a signal booster or repeater. The star topology is considered the easiest topology to design and implement. An advantage of the star topology is the simplicity of adding additional nodes. The primary disadvantage of the star topology is that the hub represents a single point of failure.

Figure 5.5: The Star Topology



Did u know?

- A point-to-point link (described above) is sometimes categorized as a special instance of the physical star topology – therefore, the simplest type of network that is based upon the physical star topology would consist of one node with a single point-to-point link to a second node, the choice of which node is the 'hub' and which node is the 'spoke' being arbitrary.
- Star networks may also be described as either broadcast multi-access or nonbroadcast multi-access (NBMA), depending on whether the technology of the network either automatically propagates a signal at the hub to all spokes, or only addresses individual spokes with each communication.

Extended Star

A type of network topology in which a network that is based upon the physical star topology has one or more repeaters between the central node (the 'hub' of the star) and the peripheral or 'spoke' nodes, the repeaters being used to extend the maximum transmission distance of the point-to-point links between the central node and the peripheral nodes beyond that which is supported by the transmitter power of the central node or beyond that which is supported by the standard upon which the physical layer of the physical star network is based.

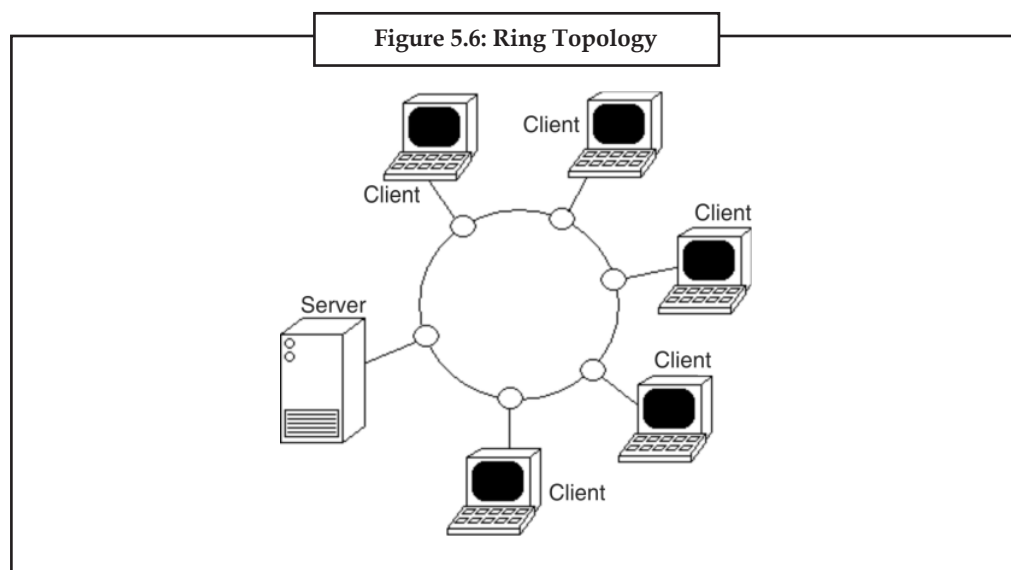
If the repeaters in a network that is based upon the physical extended star topology are replaced with hubs or switches, then a hybrid network topology is created that is referred to as a physical hierarchical star topology, although some texts make no distinction between the two topologies.

Distributed Star

A type of network topology that is composed of individual networks that are based upon the physical star topology connected together in a linear fashion - i.e., 'daisy-chained' — with no central or top level connection point (e.g., two or more 'stacked' hubs, along with their associated star connected nodes or 'spokes').

Ring Network Topology

A network topology that is set up in a circular fashion in which data travels around the ring in one direction and each device on the ring acts as a repeater to keep the signal strong as it travels. Each device incorporates a receiver for the incoming signal and a transmitter to send the data on to the next device in the ring. The network is dependent on the ability of the signal to travel around the ring.

**Mesh Networking**

The value of fully meshed networks is proportional to the exponent of the number of subscribers, assuming that communicating groups of any two endpoints, up to and including all the endpoints, is approximated by Reed's Law.

Fully connected Mesh Topology

The number of connections in a full mesh = $n(n - 1) / 2$

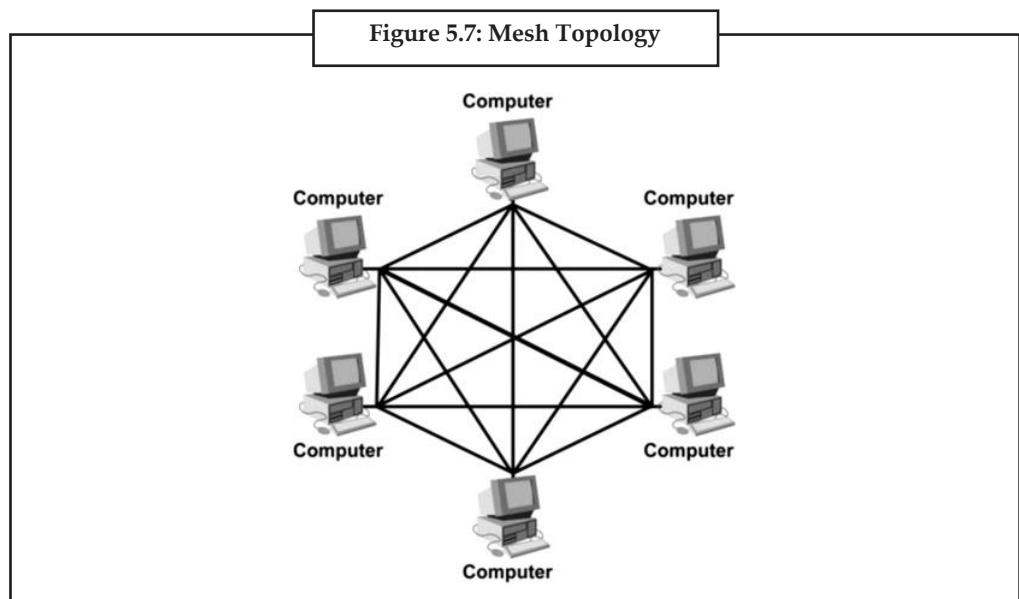
Notes

Fully connected Mesh Topology

The physical fully connected mesh topology is generally too costly and complex for practical networks, although the topology is used when there are only a small number of nodes to be interconnected.

Partially Connected Mesh Topology

The type of network topology in which some of the nodes of the network are connected to more than one other node in the network with a point-to-point link – this makes it possible to take advantage of some of the redundancy that is provided by a physical fully connected mesh topology without the expense and complexity required for a connection between every node in the network.



In most practical networks that are based upon the physical partially connected mesh topology, all of the data that is transmitted between nodes in the network takes the shortest path (or an approximation of the shortest path) between nodes, except in the case of a failure or break in one of the links, in which case the data takes an alternative path to the destination. This requires that the nodes of the network possess some type of logical 'routing' algorithm to determine the correct path to use at any particular time.

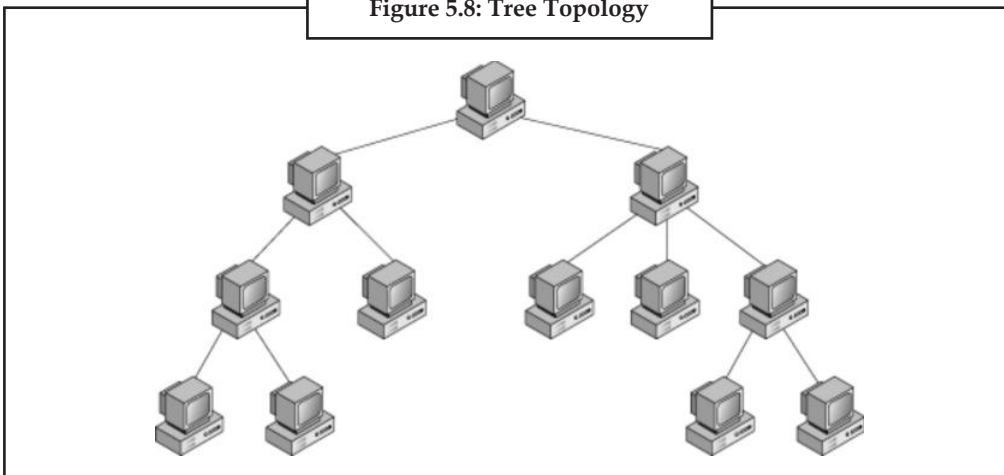
Tree Network Topology

This is also known as a **hierarchy network**. The type of network topology in which a central 'root' node (the top level of the hierarchy) is connected to one or more other nodes that are one level lower in the hierarchy (i.e., the second level) with a point-to-point link between each of the second level nodes and the top level central 'root' node, while each of the second level nodes that are connected to the top level central 'root' node will also have one or more other nodes that are one level lower in the hierarchy (i.e., the third level) connected to it, also with a point-to-point link, the top level central 'root' node being the only node that has no other node above it in the hierarchy (The hierarchy of the tree is symmetrical.) Each node in the network having a specific fixed number, of nodes connected to it at the next lower level in the hierarchy, the number, being referred to as the 'branching factor' of the hierarchical tree. This tree has individual peripheral nodes.

Notes

- (1) A network that is based upon the physical hierarchical topology must have at least three levels in the hierarchy of the tree, since a network with a central 'root' node and only one hierarchical level below it would exhibit the physical topology of a star.
- (2) A network that is based upon the physical hierarchical topology and with a branching factor of 1 would be classified as a physical linear topology.
- (3) The branching factor, f , is independent of the total number of nodes in the network and, therefore, if the nodes in the network require ports for connection to other nodes the total number of ports per node may be kept low even though the total number of nodes is large — this makes the effect of the cost of adding ports to each node totally dependent upon the branching factor and may therefore be kept as low as required without any effect upon the total number of nodes that are possible.
- (4) The total number of point-to-point links in a network that is based upon the physical hierarchical topology will be one less than the total number of nodes in the network.
- (5) If the nodes in a network that is based upon the physical hierarchical topology are required to perform any processing upon the data that is transmitted between nodes in the network, the nodes that are at higher levels in the hierarchy will be required to perform more processing operations on behalf of other nodes than the nodes that are lower in the hierarchy. Such a type of network topology is very useful and highly recommended.

Figure 5.8: Tree Topology



5.2.2.3 Logical topology

The logical topology, in contrast to the “physical”, is the way that the signals act on the network media, or the way that the data passes through the network from one device to the next without regard to the physical interconnection of the devices. A network’s logical topology is not necessarily the same as its physical topology. For example, twisted pair Ethernet is a logical bus topology in a physical star topology layout. While IBM’s Token Ring is a logical ring topology, it is physically set up in a star topology.

The logical classification of network topologies generally follows the same classifications as those in the physical classifications of network topologies but describes the path that the data takes between nodes being used as opposed to the actual physical connections between nodes.

Notes



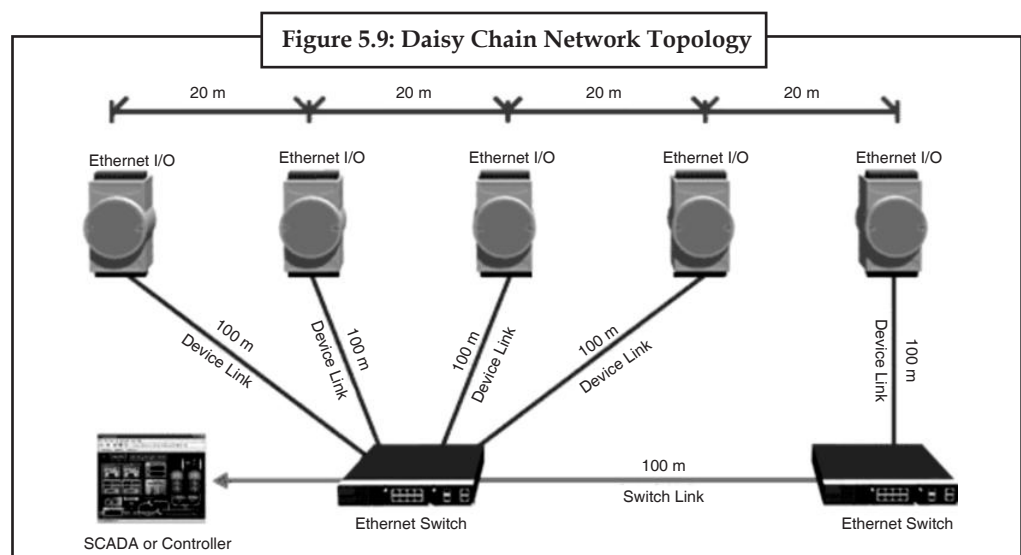
Notes

1. Logical topologies are often closely associated with Media Access Control methods and protocols.
2. The logical topologies are generally determined by network protocols as opposed to being determined by the physical layout of cables, wires, and network devices or by the flow of the electrical signals, although in many cases the paths that the electrical signals take between nodes may closely match the logical flow of data, hence the convention of using the terms logical topology and signal topology interchangeably.
3. Logical topologies are able to be dynamically reconfigured by special types of equipment such as routers and switches.

Daisy Chains

Except for star-based networks, the easiest way to add more computers into a network is by daisy-chaining, or connecting each computer in series to the next. If a message is intended for a computer partway down the line, each system bounces it along in sequence until it reaches the destination. A daisy-chained network can take two basic forms – linear and ring.

- A linear topology puts a two-way link between one computer and the next. However, this was expensive in the early days of computing, since each computer (except for the ones at each end) required two receivers and two transmitters.
- By connecting the computers at each end, a ring topology can be formed. An advantage of the ring is that the number of transmitters and receivers can be cut in half, since a message will eventually loop all of the way around. When a node sends a message, the message is processed by each computer in the ring. If a computer is not the destination node, it will pass the message to the next node, until the message arrives at its destination. If the message is not accepted by any node on the network, it will travel around the entire ring and return to the sender. This potentially results in a doubling of travel time for data.



Centralization

The **star topology** reduces the probability of a network failure by connecting all of the peripheral nodes (computers, etc.) to a central node. When the physical star topology is applied to a

Notes

logical bus network such as Ethernet, this central node (traditionally a hub) rebroadcasts all transmissions received from any peripheral node to all peripheral nodes on the network, sometimes including the originating node. All peripheral nodes may thus communicate with all others by transmitting to, and receiving from, the central node only. The failure of a transmission line linking any peripheral node to the central node will result in the isolation of that peripheral node from all others, but the remaining peripheral nodes will be unaffected. However, the disadvantage is that the failure of the central node will cause the failure of all of the peripheral nodes also.

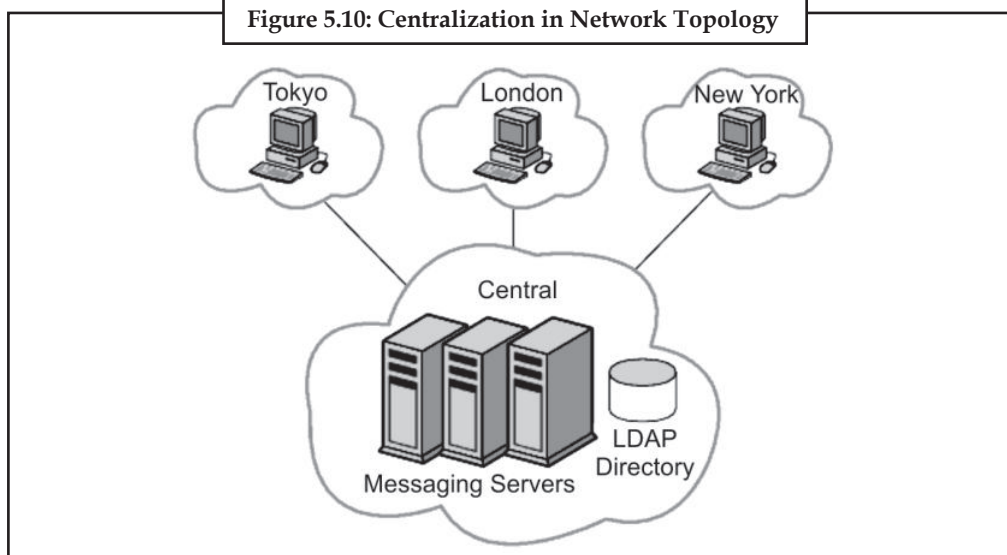
If the central node is passive, the originating node must be able to tolerate the reception of an echo of its own transmission, delayed by the two-way round trip transmission time (i.e. to and from the central node) plus any delay generated in the central node. An active star network has an active central node that usually has the means to prevent echo-related problems.

A **tree topology** (a.k.a. **hierarchical topology**) can be viewed as a collection of star networks arranged in a hierarchy. This tree has individual peripheral nodes (e.g. leaves) which are required to transmit to and receive from one other node only and are not required to act as repeaters or regenerators. Unlike the star network, the functionality of the central node may be distributed.

As in the conventional star network, individual nodes may thus still be isolated from the network by a single-point failure of a transmission path to the node. If a link connecting a leaf fails, that leaf is isolated; if a connection to a non-leaf node fails, an entire section of the network becomes isolated from the rest.

In order to alleviate the amount of network traffic that comes from broadcasting all signals to all nodes, more advanced central nodes were developed that are able to keep track of the identities of the nodes that are connected to the network. These network switches will “learn” the layout of the network by “listening” on each port during normal data transmission, examining the data packets and recording the address/identifier of each connected node and which port it’s connected to in a lookup table held in memory. This lookup table then allows future transmissions to be forwarded to the intended destination only.

Figure 5.10: Centralization in Network Topology



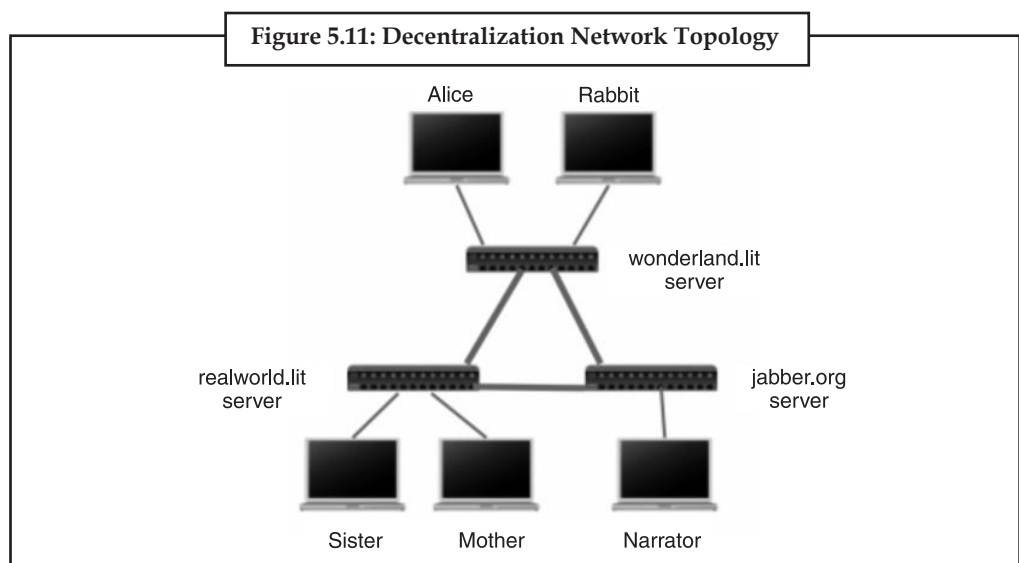
Decentralization

In a **mesh topology** (i.e., a partially connected mesh topology), there are at least two nodes with two or more paths between them to provide redundant paths to be used in case the link providing

Notes

one of the paths fails. This decentralization is often used to advantage to compensate for the single-point-failure disadvantage that is present when using a single device as a central node (e.g., in star and tree networks). A special kind of mesh, limiting the number of hops between two nodes, is a hypercube. The number of arbitrary forks in mesh networks makes them more difficult to design and implement, but their decentralized nature makes them very useful. This is similar in some ways to a **grid network**, where a linear or ring topology is used to connect systems in multiple directions. A multi-dimensional ring has a toroidal topology, for instance.

A **fully connected network, complete topology** or **full mesh topology** is a network topology in which there is a direct link between all pairs of nodes. In a fully connected network with n nodes, there are $n(n-1)/2$ direct links. Networks designed with this topology are usually very expensive to set up, but provide a high degree of reliability due to the multiple paths for data that are provided by the large number of redundant links between nodes. This topology is mostly seen in military applications. However, it can also be seen in the file sharing protocol BitTorrent in which users connect to other users in the “swarm” by allowing each user sharing the file to connect to other users also involved. Often in actual usage of BitTorrent any given individual node is rarely connected to every single other node as in a true fully connected network but the protocol does allow for the possibility for any one node to connect to any other node when sharing files.



Hybrids

Hybrid networks use a combination of any two or more topologies in such a way that the resulting network does not exhibit one of the standard topologies (e.g., bus, star, ring, etc.). For example, a tree network connected to a tree network is still a tree network, but two star networks connected together exhibit a hybrid network topology. A hybrid topology is always produced when two different basic network topologies are connected. Two common examples for Hybrid network are: star ring network and star bus network

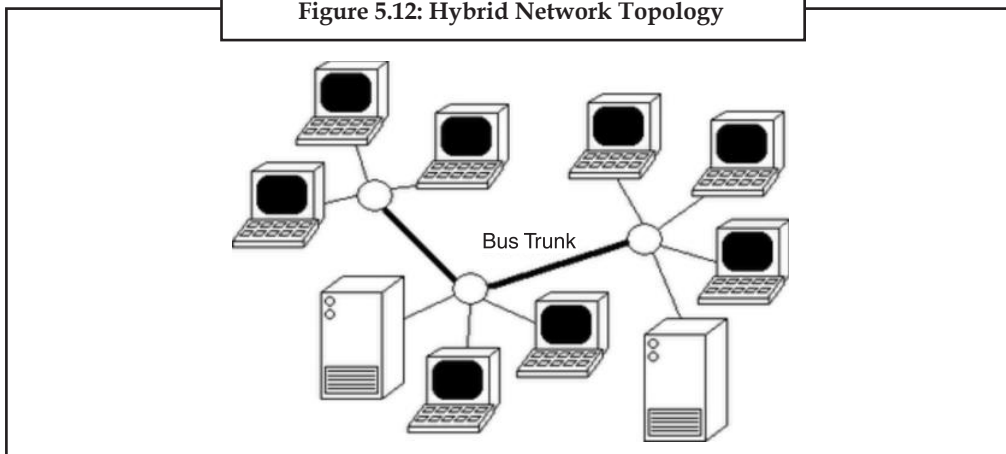
- A Star Ring network consists of two or more star topologies connected using a multistation access unit (MAU) as a centralized hub.
- A Star Bus network consists of two or more star topologies connected using a bus trunk (the bus trunk serves as the network's backbone).

Notes

While grid networks have found popularity in high-performance computing applications, some systems have used genetic algorithms to design custom networks that have the fewest possible hops in between different nodes. Some of the resulting layouts are nearly incomprehensible, although they function quite well.

A Snowflake topology is really a "Star of Stars" network, so it exhibits characteristics of a hybrid network topology but is not composed of two different basic network topologies being connected together.

Figure 5.12: Hybrid Network Topology



5.3 Network Media

At Media Network, we work hand in hand with each of our clients from step one to ensure constant communication and the growth of a long lasting relationship. We take great pride in providing the highest level of quality and top standards. With over 10 years of combined experience and knowledge, we are a cutting edge factor on the internet. Our deep commitment to our performance and our client satisfaction is a fundamental element to our beliefs. Your success guarantees ours! You will NOT deal with 2nd and 3rd party involvement that can lead to poor results and wasted profits. You can be confident in our 1 source, total solution to optimize and maximize your ROI.

Network media is the actual path over which an electrical signal travels as it moves from one component to another. This chapter describes the common types of network media, including twisted-pair cable, coaxial cable, fiber-optic cable, and wireless.

5.3.1 Twisted-Pair Cable

Twisted-pair cable is a type of cabling that is used for telephone communications and most modern Ethernet networks. A pair of wires forms a circuit that can transmit data. The pairs are twisted to provide protection against crosstalk, the noise generated by adjacent pairs. When electrical current flows through a wire, it creates a small, circular magnetic field around the wire. When two wires in an electrical circuit are placed close together, their magnetic fields are the exact opposite of each other. Thus, the two magnetic fields cancel each other out. They also cancel out any outside magnetic fields. Twisting the wires can enhance this cancellation effect. Using cancellation together with twisting the wires, cable designers can effectively provide self-shielding for wire pairs within the network media.

Two basic types of twisted-pair cable exist: unshielded twisted pair (UTP) and shielded twisted pair (STP). The following sections discuss UTP and STP cable in more detail.

Notes

UTP Cable

UTP cable is a medium that is composed of pairs of wires. UTP cable is used in a variety of networks. Each of the eight individual copper wires in UTP cable is covered by an insulating material. In addition, the wires in each pair are twisted around each other.

UTP cable relies solely on the cancellation effect produced by the twisted wire pairs to limit signal degradation caused by electromagnetic interference (EMI) and radio frequency interference (RFI). To further reduce crosstalk between the pairs in UTP cable, the number of twists in the wire pairs varies. UTP cable must follow precise specifications governing how many twists or braids are permitted per meter (3.28 feet) of cable.

UTP cable often is installed using a Registered Jack 45 (RJ-45) connector. The RJ-45 is an eight-wire connector used commonly to connect computers onto a local-area network (LAN), especially Ethernets.

When used as a networking medium, UTP cable has four pairs of either 22- or 24-gauge copper wire. UTP used as a networking medium has an impedance of 100 ohms; this differentiates it from other types of twisted-pair wiring such as that used for telephone wiring, which has impedance of 600 ohms.

UTP cable offers many advantages. Because UTP has an external diameter of approximately 0.43 cm (0.17 inches), its small size can be advantageous during installation. Because it has such a small external diameter, UTP does not fill up wiring ducts as rapidly as other types of cable. This can be an extremely important factor to consider, particularly when installing a network in an older building. UTP cable is easy to install and is less expensive than other types of networking media. In fact, UTP costs less per meter than any other type of LAN cabling. And because UTP can be used with most of the major networking architectures, it continues to grow in popularity.

Disadvantages also are involved in using twisted-pair cabling, however. UTP cable is more prone to electrical noise and interference than other types of networking media, and the distance between signal boosts is shorter for UTP than it is for coaxial and fiber-optic cables.

Although UTP was once considered to be slower at transmitting data than other types of cable, this is no longer true. In fact, UTP is considered the fastest copper-based medium today. The following summarizes the features of UTP cable:

Speed and throughput-10 to 1000 Mbps

Average cost per node-Least expensive

Media and connector size-Small

Maximum cable length-100 m (short)

Commonly used types of UTP cabling are as follows:

Category 1: Used for telephone communications. Not suitable for transmitting data.

Category 2: Capable of transmitting data at speeds up to 4 megabits per second (Mbps).

Category 3: Used in 10BASE-T networks. Can transmit data at speeds up to 10 Mbps.

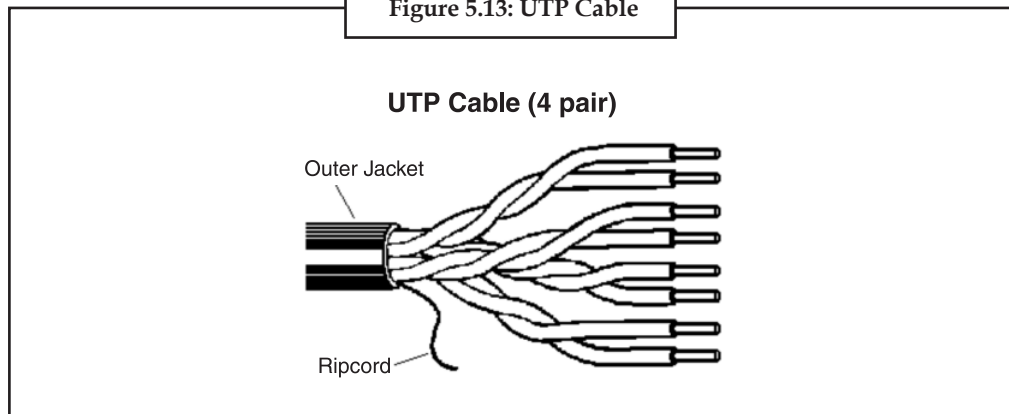
Category 4: Used in Token Ring networks. Can transmit data at speeds up to 16 Mbps.

Category 5: Can transmit data at speeds up to 100 Mbps.

Category 5e: Used in networks running at speeds up to 1000 Mbps (1 gigabit per second [Gbps]).

Category 6: Typically, Category 6 cable consists of four pairs of 24 American Wire Gauge (AWG) copper wires. Category 6 cable is currently the fastest standard for UTP.

Figure 5.13: UTP Cable



5.3.2 Shielded Twisted-Pair Cable

Shielded twisted-pair (STP) cable combines the techniques of shielding, cancellation, and wire twisting. Each pair of wires is wrapped in a metallic foil. The four pairs of wires then are wrapped in an overall metallic braid or foil, usually 150-ohm cable. As specified for use in Ethernet network installations, STP reduces electrical noise both within the cable (pair-to-pair coupling, or crosstalk) and from outside the cable (EMI and RFI). STP usually is installed with STP data connector, which is created especially for the STP cable. However, STP cabling also can use the same RJ connectors that UTP uses.

Although STP prevents interference better than UTP, it is more expensive and difficult to install. In addition, the metallic shielding must be grounded at both ends. If it is improperly grounded, the shield acts like an antenna and picks up unwanted signals. Because of its cost and difficulty with termination, STP is rarely used in Ethernet networks. STP is primarily used in Europe.

The following summarizes the features of STP cable:

Speed and throughput-10 to 100 Mbps

Average cost per node-Moderately expensive

Media and connector size-Medium to large

Maximum cable length-100 m (short)

When comparing UTP and STP, keep the following points in mind:

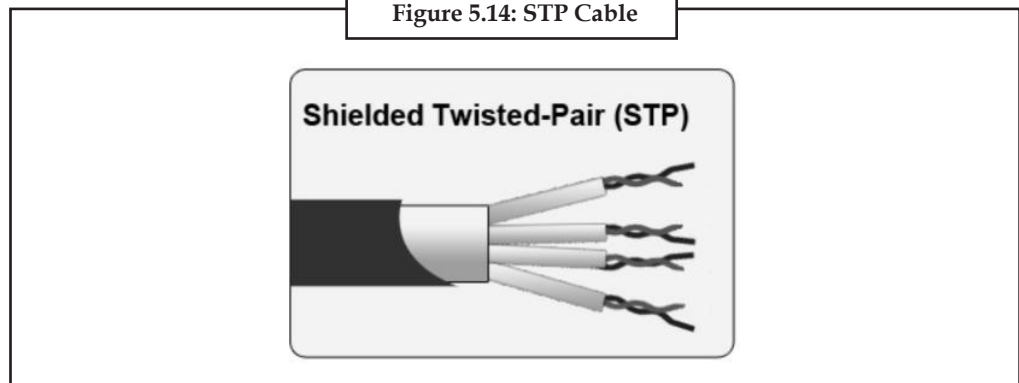
The speed of both types of cable is usually satisfactory for local-area distances.

These are the least-expensive media for data communication. UTP is less expensive than STP.

Because most buildings are already wired with UTP, many transmission standards are adapted to use it, to avoid costly rewiring with an alternative cable type.

Notes

Figure 5.14: STP Cable



5.4 Basic Hardware

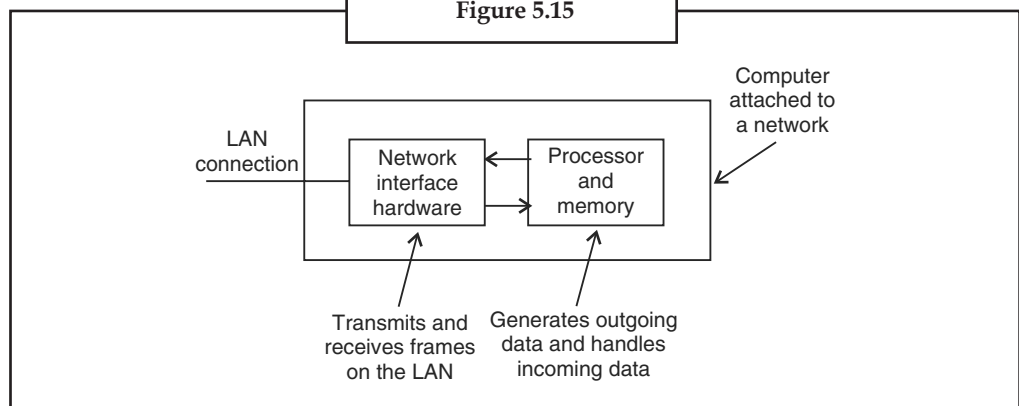
All networks are made up of basic hardware building blocks to interconnect network nodes, such as Network Interface Cards (NICs), Bridges, Hubs, Switches, and Routers. In addition, some method of connecting these building blocks is required, usually in the form of galvanic cable (most commonly Category 5 cable). Less common are microwave links (as in IEEE 802.12) or optical cable ("optical fiber").

5.4.1 Network Interface Cards

A network card, network adapter, or NIC (network interface card) is a piece of computer hardware designed to allow computers to communicate over a computer network. It provides physical access to a networking medium and often provides a low-level addressing system through the use of MAC addresses.

Each network interface card has its unique id. This is written on a chip which is mounted on the card.

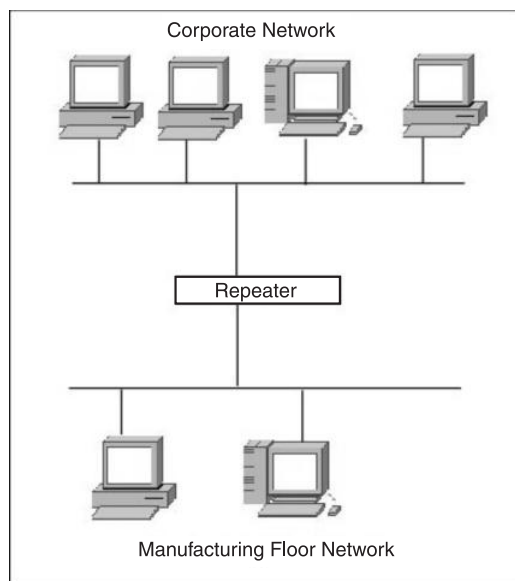
Figure 5.15



5.4.2 Repeaters

A repeater is an electronic device that receives a signal, cleans it of unnecessary noise, regenerates it, and retransmits it at a higher power level, or to the other side of an obstruction, so that the signal can cover longer distances without degradation. In most twisted pair Ethernet configurations, repeaters are required for cable that runs longer than 100 meters. A repeater with multiple ports is known as a hub. Repeaters work on the Physical Layer of the OSI model. Repeaters require a small amount of time to regenerate the signal. This can cause a propagation delay which can affect network communication when there are several repeaters in a row. Many network architectures limit the number of repeaters that can be used in a row (e.g. Ethernet's 5-4-3 rule).

Figure 5.16

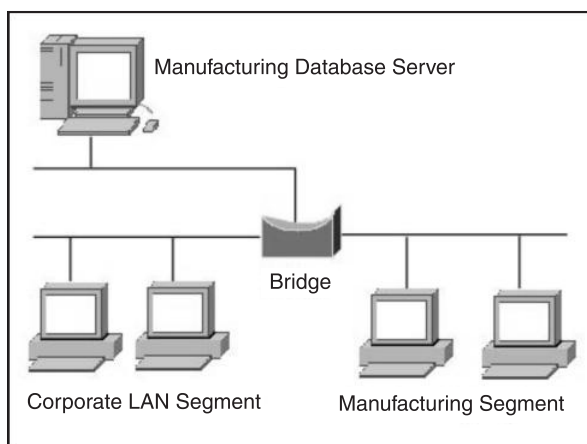


5.4.3 Bridges

A network bridge connects multiple network segments at the data link layer (layer 2) of the OSI model. Bridges broadcast to all ports except the port on which the broadcast was received. However, bridges do not promiscuously copy traffic to all ports, as hubs do, but learn which MAC addresses are reachable through specific ports. Once the bridge associates a port and an address, it will send traffic for that address to that port only.

Bridges learn the association of ports and addresses by examining the source address of frames that it sees on various ports. Once a frame arrives through a port, its source address is stored and the bridge assumes that MAC address is associated with that port. The first time that a previously unknown destination address is seen, the bridge will forward the frame to all ports other than the one on which the frame arrived.

Figure 5.17: A Bridge in OSI Model Layer



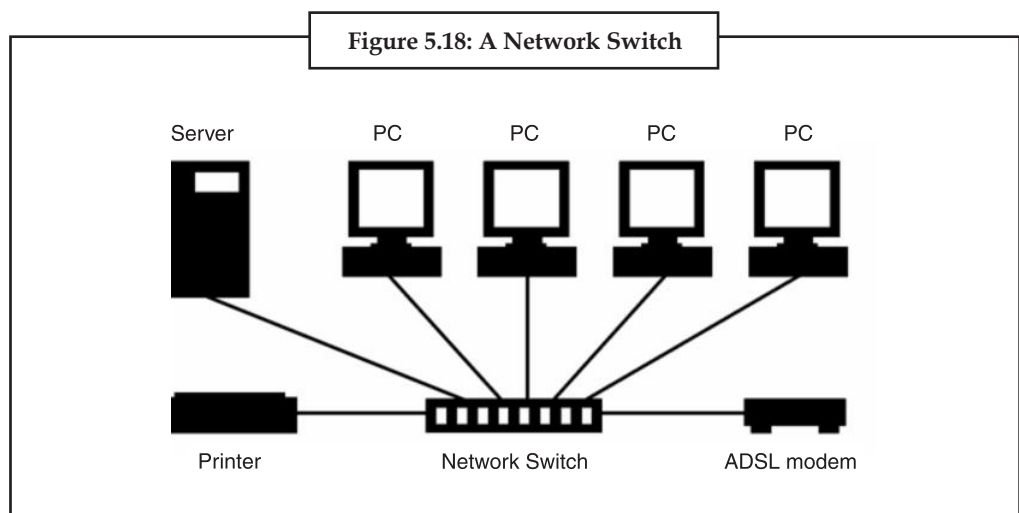
Notes

5.4.3.1 Bridges come in three basic types

- **Local bridges:** Directly connect local area networks (LANs).
- **Remote bridges:** Can be used to create a wide area network (WAN) link between LANs. Remote bridges, where the connecting link is slower than the end networks, largely have been replaced with routers.
- **Wireless bridges:** Can be used to join LANs or connect remote stations to LANs.

5.4.4 Switches

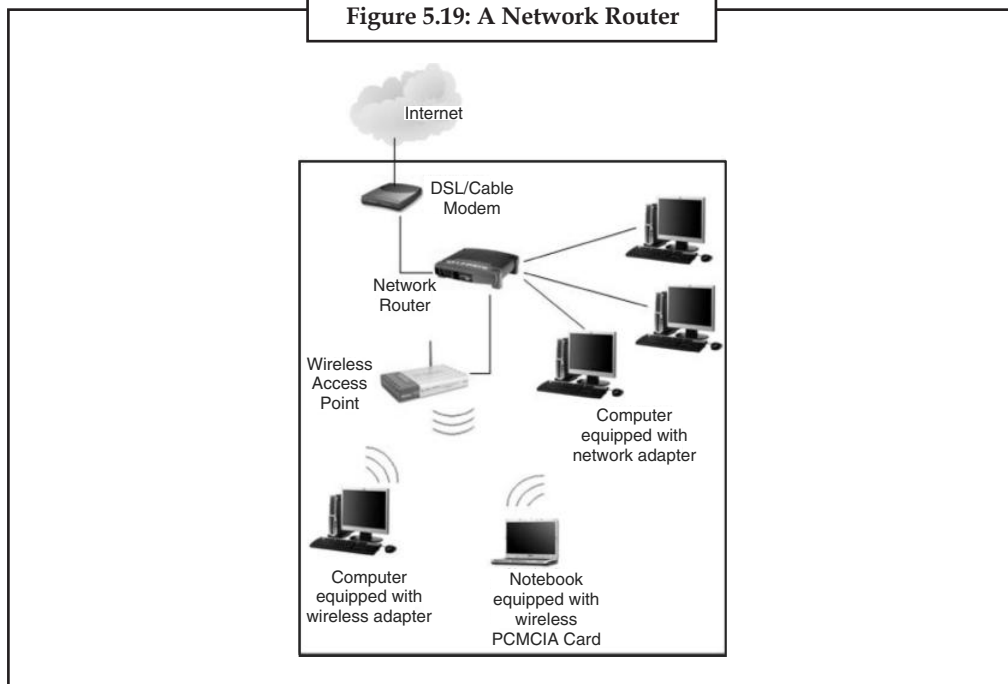
A network switch is a device that forwards and filters OSI layer 2 datagrams (chunks of data communication) between ports (connected cables) based on the MAC addresses in the packets. A switch is distinct from a hub in that it only forwards the frames to the ports involved in the communication rather than all ports connected. A switch breaks the collision domain but represents itself as a broadcast domain. Switches make forwarding decisions of frames on the basis of MAC addresses. A switch normally has numerous ports, facilitating a star topology for devices, and cascading additional switches. Some switches are capable of routing based on Layer 3 addressing or additional logical levels; these are called multi-layer switches. The term switch is used loosely in marketing to encompass devices including routers and bridges, as well as devices that may distribute traffic on load or by application content (e.g., a Web URL identifier).



5.4.5 Routers

A router is an internetworking device that forwards packets between networks by processing information found in the datagram or packet (Internet protocol information from Layer 3 of the OSI Model). In many situations, this information is processed in conjunction with the routing table (also known as forwarding table). Routers use routing tables to determine what interface to forward packets (this can include the “null” also known as the “black hole” interface because data can go into it, however, no further processing is done for said data).

Figure 5.19: A Network Router



5.4.6 Firewalls

Firewalls are the most important aspect of a network with respect to security. A firewalled system does not need every interaction or data transfer monitored by a human, as automated processes can be set up to assist in rejecting access requests from unsafe sources, and allowing actions from recognized ones. The vital role firewalls play in network security grows in parallel with the constant increase in 'cyber' attacks for the purpose of stealing/corrupting data, planting viruses, etc.

5.5 Summary

- A network architecture is a blue print of the complete computer communication network, which provides a framework and technology foundation of network.
- Network topology is the layout pattern of interconnections of the various elements (links, nodes etc.) of a computer network.
- Protocol specifies a common set of rules and signals of computers on the network use to communicate.
- Network media is the actual path over which an electrical signal travels as it moves from one component to another.
- All networks are made up of basic hardware building blocks to interconnect network nodes such as NICs, bridges, hubs, switches and routers.

5.6 Keywords

Optical fiber cable: Optical fiber cable consists of one or more filaments of glass fiber wrapped in protective layers that carries a data by means of pulses of light.

Notes

Overlay network: An overlay network is a virtual computer network that is built on top of another network. Nodes in the overlay are connected by virtual or logical links, each of which corresponds to a path, perhaps through many physical links, in the underlying network.

Twisted pair wire: Twisted pair wire is the most widely used medium for telecommunication. Twisted-pair cabling consist of copper wires that are twisted into pairs.

Virtual private network (VPN): A virtual private network (VPN) is a computer network in which some of the links between nodes are carried by open connections or virtual circuits in some larger network (e.g., the Internet) instead of by physical wires.



Draw Network Structure.

Lab Exercise

5.7 Self-Assessment Questions

1. A perceptron is:
 - (a) A single layer feed-forward neural network with preprocessing
 - (b) An autoassociative neural network
 - (c) A double layer autoassociative neural network
 - (d) None of these
2. Which of the following is true for neural networks?
 - (i) The training time depends on the size of the network.
 - (ii) Neural networks can be simulated on a conventional computer.
 - (iii) Artificial neurons are identical in operation to biological ones.
 - (a) all of them are true.
 - (b) (ii) is true.
 - (c) (i) and (ii) are true.
 - (d) all of these
3. Storing a placemark file on the network or on a web server offers the following advantages:
 - (a) Accessibility
 - (b) Automatic Updates/Network Link Access
 - (c) Backup.
 - (d) All of these
4. To save a placemark or folder to a web server, first save the file to your local computer as described in Saving Places Data.
 - (a) True
 - (b) False
5. Sharing hardware. In a networked environment, each computer on a network may access and use hardware resources on the network, such as printing a document on a shared network printer.
 - (a) True
 - (b) False

5.8 Review Questions

1. How will you Create a Network Link?
2. What is the Purpose of networking?
3. Explain Network classification.
4. Explain Network Topology.
5. Explain Network Protocol.
6. Explain Network Architecture.
7. Explain Basic topology types.
8. Explain basic hardware.

Answers for Self-Assessment Questions

1. (a) 2. (c) 3. (d) 4. (a) 5. (b)

5.9 Further Reading



Books

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition



Online link

<http://www.networkcomputing.com/> -

Unit 6: Data Communication

CONTENTS

Objectives

Introduction

6.1 Local and Global Reach of the Network

6.1.1 Views of Networks

6.1.2 Networking Methods

6.2 Data Communication with Standard Telephone Lines

6.2.1 Dial-Up Lines

6.2.2 Dedicated Lines

6.3 Data Communication with Modems

6.3.1 Narrow-Band/Phone-Line Dialup Modems

6.3.2 Radio Modems

6.3.3 Mobile Modems and Routers

6.3.4 Broadband

6.3.5 Home Networking

6.3.6 Deep-space Telecommunications

6.3.7 Voice Modem

6.4 Data Communication using Digital Data Connections

6.4.1 Digital Data with Analog Signals

6.4.2 Analog Data with Digital Signals

6.4.3 Digital Data with Digital Signals

6.4.4 Some Digital Data Connection Methods

6.5 Wireless Networks

6.5.1 Types of Wireless Connections

6.5.2 Uses

6.5.3 Environmental Concerns and Health Hazard

6.6 Summary

6.7 Keywords

6.8 Self-Assessment Questions
 6.9 Review Questions
 6.10 Further Reading

Objectives

After studying this unit, you will be able to:

- Discuss local and global reach of network
- Explain data communication with standard telephone lines
- Explain data communication with modems
- Understand data communication using digital data connections
- Explain wireless networks

Introduction

Data Communications is the transfer of data or information between a source and a receiver. The source transmits the data and the receiver receives it. The actual generation of the information is not part of Data Communications nor is the resulting action of the information at the receiver. Data Communication is interested in the transfer of data, the method of transfer and the preservation of the data during the transfer process.

In Local Area Networks, we are interested in “connectivity”, connecting computers together to share resources. Even though the computers can have different disk operating systems, languages, cabling and locations, they still can communicate to one another and share resources.

The purpose of Data Communications is to provide the rules and regulations that allow computers with different disk operating systems, languages, cabling and locations to share resources. The rules and regulations are called protocols and standards in Data Communications.

6.1 Local and Global Reach of the Network

Data transmission, digital transmission, or digital communications is the physical transfer of data (a digital bit stream) over a point-to-point or point-to-multi-point communication channel. Examples of such channels are copper wires, optical fibers, wireless communication channels, and storage media. The data is represented as an electromagnetic signal, such as an electrical voltage, radiowave, microwave, or infrared signal.

The messages are either represented by a sequence of pulses by means of a line code (baseband transmission), or by a limited set of continuously varying wave forms (passband transmission), using a digital modulation method. The passband modulation and corresponding demodulation (also known as detection) is carried out by modem equipment. According to the most common definition of digital signal, both baseband and passband signals representing bit-streams are considered as digital transmission, while an alternative definition only considers the baseband signal as digital, and passband transmission of digital data as a form of digital-to-analog conversion.

Notes

Data transmitted may be digital messages originating from a data source, for example a computer or a keyboard. It may also be an analog signal such as a phone call or a video signal, digitized into a bit-stream for example using pulse-code modulation (PCM) or more advanced source coding (analog-to-digital conversion and data compression) schemes. This source coding and decoding is carried out by code equipment.

Computer networking or **Data communications (Datacom)** is the engineering discipline concerned with the communication between computer systems or devices. A computer network is any set of computers or devices connected to each other with the ability to exchange data. Computer networking is sometimes considered a sub-discipline of telecommunications, computer science, information technology and/or computer engineering since it relies heavily upon the theoretical and practical application of these scientific and engineering disciplines. The three types of networks are: the Internet, the intranet, and the extranet. Examples of different network methods are:

- Local area network (LAN), which is usually a small network constrained to a small geographic area. An example of a LAN would be a computer network within a building.
- Metropolitan area network (MAN), which is used for medium size area. examples for a city or a state.
- Wide area network (WAN) that is usually a larger network that covers a large geographic area.
- Wireless LANs and WANs (WLAN & WWAN) are the wireless equivalent of the LAN and WAN.

All networks are interconnected to allow communication with a variety of different kinds of media, including twisted-pair copper wire cable, coaxial cable, optical fiber, power lines and various wireless technologies. The devices can be separated by a few meters (e.g. via Bluetooth) or nearly unlimited distances (e.g. via the interconnections of the Internet). Networking, routers, routing protocols, and networking over the public Internet have their specifications defined in documents called RFCs.

6.1.1 Views of Networks

Users and network administrators typically have different views of their networks. Users can share printers and some servers from a workgroup, which usually means they are in the same geographic location and are on the same LAN, whereas a Network Administrator is responsible to keep that network up and running. A community of interest has less of a connection of being in a local area, and should be thought of as a set of arbitrarily located users who share a set of servers, and possibly also communicate via peer-to-peer technologies.

Network administrators can see networks from both physical and logical perspectives. The physical perspective involves geographic locations, physical cabling, and the network elements (e.g., routers, bridges and application layer gateways that interconnect the physical media. Logical networks, called, in the TCP/IP architecture, subnets, map onto one or more physical media. For example, a common practice in a campus of buildings is to make a set of LAN cables in each building appear to be a common subnet, using virtual LAN (VLAN) technology.

Both users and administrators will be aware, to varying extents, of the trust and scope characteristics of a network. Again using TCP/IP architectural terminology, an intranet is a community of interest

under private administration usually by an enterprise, and is only accessible by authorized users (e.g. employees). Intranets do not have to be connected to the Internet, but generally have a limited connection. An extranet is an extension of an intranet that allows secure communications to users outside of the intranet (e.g. business partners, customers).

Unofficially, the Internet is the set of users, enterprises, and content providers that are interconnected by Internet Service Providers (ISP). From an engineering viewpoint, the Internet is the set of subnets, and aggregates of subnets, which share the registered IP address space and exchange information about the reachability of those IP addresses using the Border Gateway Protocol. Typically, the human-readable names of servers are translated to IP addresses, transparently to users, via the directory function of the Domain Name System (DNS).

Over the Internet, there can be business-to-business (B2B), business-to-consumer (B2C) and consumer-to-consumer (C2C) communications. Especially when money or sensitive information is exchanged, the communications are apt to be secured by some form of communications security mechanism. Intranets and extranets can be securely superimposed onto the Internet, without any access by general Internet users, using secure Virtual Private Network (VPN) technology.

6.1.2 Networking Methods

One way to categorize computer networks is by their geographic scope, although many real-world networks interconnect Local Area Networks (LAN) via Wide Area Networks (WAN) and wireless wide area networks (WWAN). These three (broad) types are:

6.1.2.1 Local Area Network (LAN)

A local area network is a network that spans a relatively small space and provides services to a small number of people.

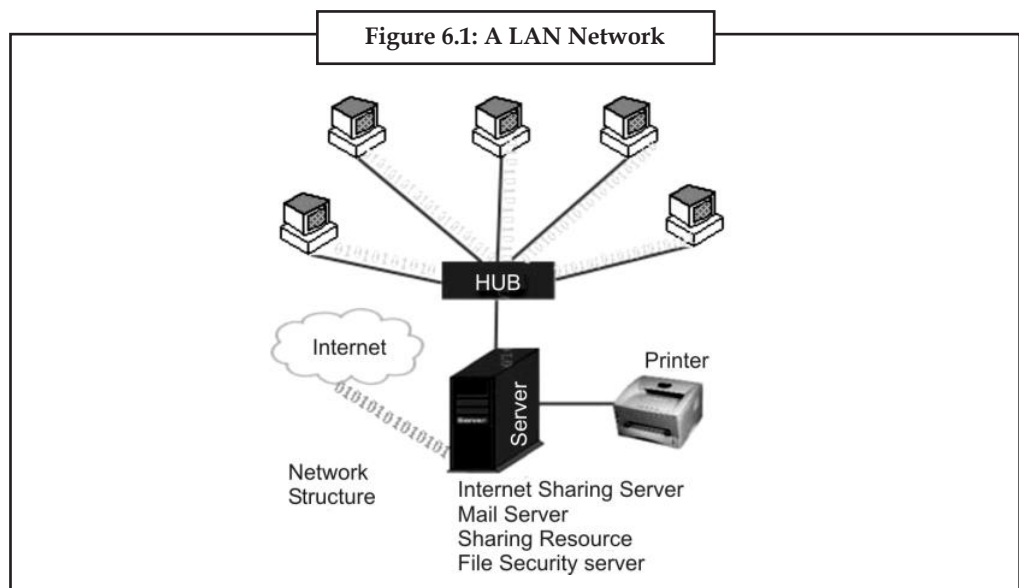
A peer-to-peer or client-server method of networking may be used. A peer-to-peer network is where each client shares their resources with other workstations in the network. Examples of peer-to-peer networks are: Small office networks where resource use is minimal and a home network. A client-server network is where every client is connected to the server and each other. Client-server networks use servers in different capacities. These can be classified into two types:

1. Single-service servers
2. Print servers

The server performs one task such as file server, while other servers can not only perform in the capacity of file servers and print servers, but also can conduct calculations and use them to provide information to clients (Web/Intranet Server). Computers may be connected in many different ways, including Ethernet cables, Wireless networks, or other types of wires such as power lines or phone lines.

The ITU-T G.hn standard is an example of a technology that provides high-speed (up to 1 Gbit/s) local area networking over existing home wiring (power lines, phone lines and coaxial cables).

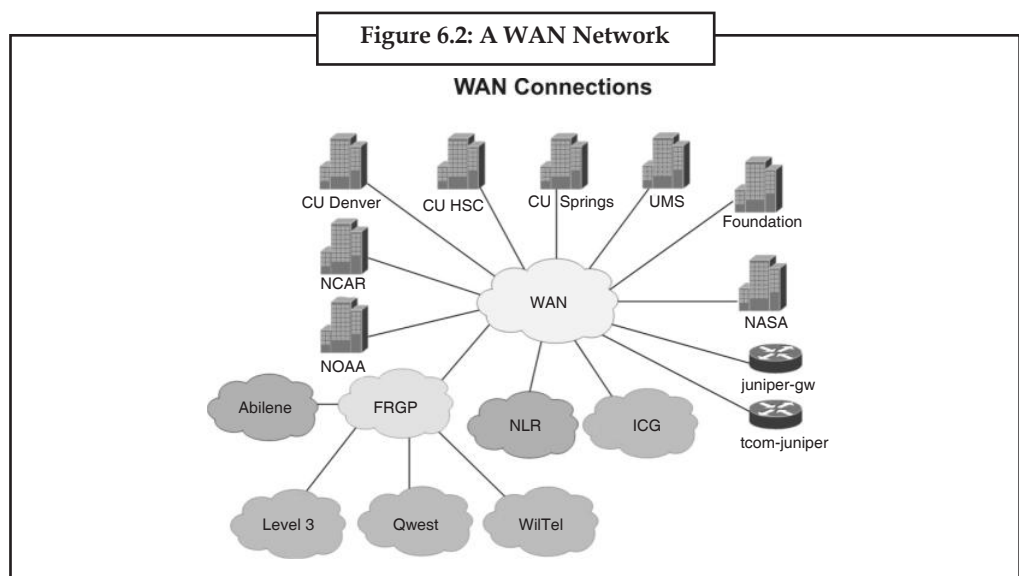
Notes



6.1.2.2 Wide Area Network (WAN)

A wide area network is a network where a wide variety of resources are deployed across a large domestic area or internationally. An example of this is a multinational business that uses a WAN to interconnect their offices in different countries. The largest and best example of a WAN is the Internet, which is a network composed of many smaller networks. The Internet is considered the largest network in the world. The PSTN (Public Switched Telephone Network) also is an extremely large network that is converging to use Internet technologies, although not necessarily through the public Internet.

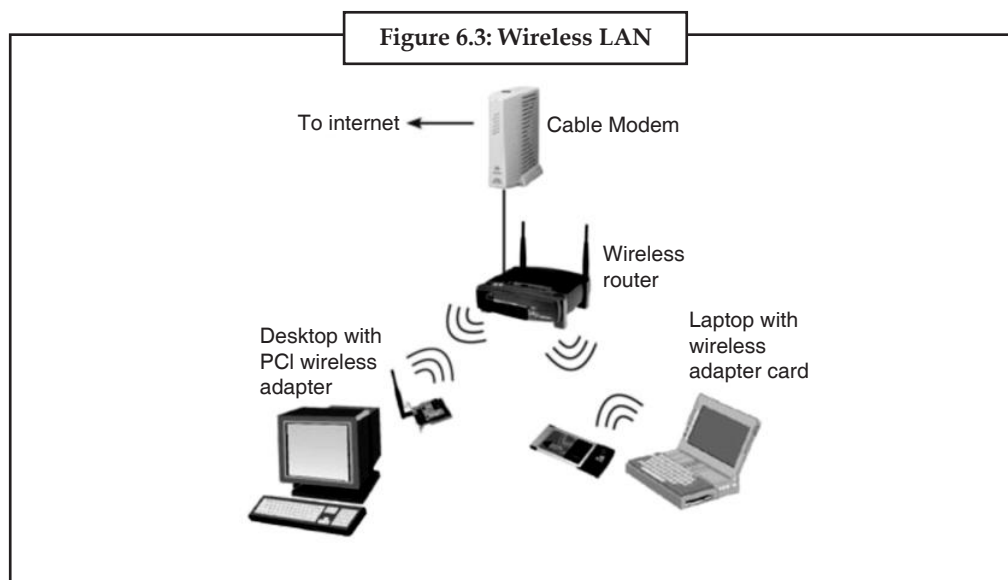
A Wide Area Network involves communication through the use of a wide range of different technologies. These technologies include Point-to-Point WANs such as Point-to-Point Protocol (PPP) and High-Level Data Link Control (HDLC), Frame Relay, ATM (Asynchronous Transfer Mode) and Sonet (Synchronous Optical Network). The difference between the WAN technologies is based on the switching capabilities they perform and the speed at which sending and receiving bits of information (data) occur.



6.1.2.3 Wireless Networks (WLAN, WWAN)

A wireless network is basically the same as a LAN or a WAN but there are no wires between hosts and servers. The data is transferred over sets of radio transceivers. These types of networks are beneficial when it is too costly or inconvenient to run the necessary cables. For more information, see Wireless LAN and Wireless wide area network. The media access protocols for LANs come from the IEEE.

The most common IEEE 802.11 WLANs cover, depending on antennas, ranges from hundreds of meters to a few kilometers. For larger areas, either communications satellites of various types, cellular radio, or wireless local loop (IEEE 802.16) all have advantages and disadvantages. Depending on the type of mobility needed, the relevant standards may come from the IETF or the ITU.



6.2 Data Communication with Standard Telephone Lines

The public switched telephone network (PSTN) is the world wide telephone system and usually this network system uses the digital technology. In the past, it was used for voice communication only but now it is playing very important role for data communication in the computer network such as in the Internet. There are different types of telephone lines that are used for data communication in the network. These are discussed below.

6.2.1 Dial-Up Lines

It is a temporary connection that uses communication. Modern is used at the s telephone number is dialed from the sender the receiving end answers the call. In the communication between computers or e the cost of data communication is very Internet through this connection.

6.2.2 Dedicated Lines

It is a permanent connection that is establish connection between two permanently. It is better than dial-up lines connection because dedicated lines provide a constant connection. These types of connections may be digital or analog. The data transmission speed, of digital lines is very fast as compare to analog dedicated line. The data transmission speed is also measured in bits per second (bps). In dial-up and dedicated lines, it is up to 56 Kbps. The dedicated lines are mostly used for business purposes. The most important digital dedicated lines are described below.

Notes

6.2.2.1 ISDN Lines

ISDN stands for Integrated Services Digital Network. It is a set of standards used for digital transmission over telephone line. The ISDN uses the multiplexing technique to carry three or more data signals at once through the telephone line. It is because the data transmission speed of ISDN line is very fast. In ISDN line, both ends of connections require the ISDN modem and a special telephone set for voice communication. Its data transmission speed is up to 128 Kbps.

6.2.2.2 DSL

DSL stands for Digital Subscriber Line. It is another digital line. In DSL, both ends of connections require the network cards and DSL modems for data communication. The data transmission speed and other functions are similar as ISDN line. DSL transmits data on existing standard copper telephone wiring. Some DSLs provide a dial tone, which allows both voice and data communication.

6.2.2.3 ADSL

The ADSL (Asymmetric Digital Subscriber Line) is another digital connection. It is faster than DSL. ADSL is much easier to install and provides much faster data transfer rate. Its data transmission speed is from 128 Kbps up to 10 tvlbps. This connection is ideal for Internet access.

6.2.2.4 Cable Television Line

The Cable Television (CATV) line is not a standard telephone line. It is a dedicated line used to access the Internet. Its data transmission speed is 128 Kbps to 3 Mbps.

A cable, modem is used with the CATV it provides a high speed Internet connections through the cable television network. A cable modem sends and receives digital data over the cable television network.

To access the Internet using the CATV network, the CATV Company installs a splitter inside your house. From the splitter, one part of the cable runs to your television and other part connects to the cable modem. A cable modem usually is an external device, in which one end of a cable connects to a CATV wall outlet while the other end plugs into a port (such as on an Ethernet card) in the system unit.

6.2.2.5 T-Carrier Lines

It is very fast digital line that can carry multiple signals over a single communication line whereas a standard dialup telephone line carries only one signal. 1-carrier lines use multiplexing so that multiple signals share the line. T-carrier lines provide very fast data transfer rates. The T-carrier lines are very expensive and large companies can afford these lines. The most popular T-carrier lines are:

- (i) T1 Line
- (ii) T3 Line

T1 Line The most popular 1-carrier line is the T1 line (dedicated line). Its data transmission speed is 1.5 Mbps. Many ISPs use T1 lines to connect to the Internet backbone. Another type of T1 line is the fractional T1 line. It is slower than T1 line but it is less expensive. The home and business users use this line to connect to the Internet and share a connection to the T1 line with other users.



Notes

Businesses often use T1 lines to connect to the Internet.

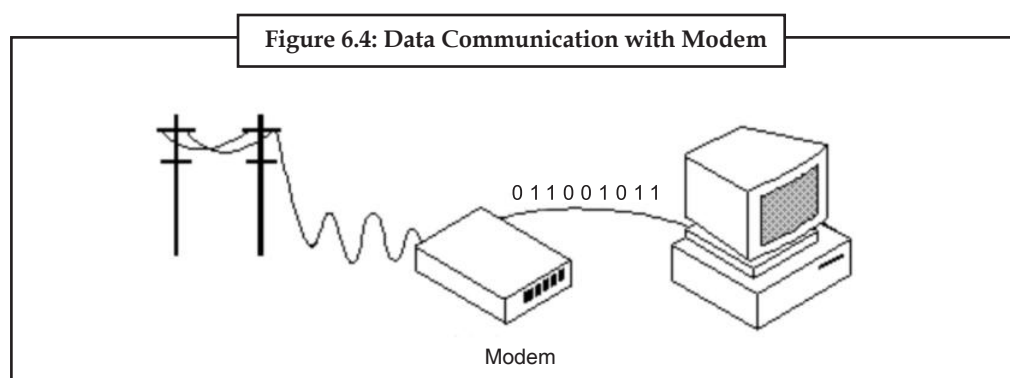
T3 Line Another most popular and faster 1-carrier line is T3 line. Its data transmission speed is 44 Mbps. It is more expensive than T1 line. The main users of T3 line are telephone companies and ISPs. The Internet backbone itself also uses T3 lines.

6.2.2.6 Asynchronous Transfer Mode (ATM)

It is very, fast data transmission connection line that can carry data, voice, video, multimedia etc. Telephone networks, Internet and other network use ATM. In near future, ATM will become the Internet standard for data transmission instead of T3 lines. Its data transmission speed is from 155 Mbps to 600 Mbps.

6.3 Data Communication with Modems

A **modem (modulator-demodulator)** is a device that modulates an analog carrier signal to encode digital information, and also demodulates such a carrier signal to decode the transmitted information. The goal is to produce a signal that can be transmitted easily and decoded to reproduce the original digital data. Modems can be used over any means of transmitting analog signals, from driven diodes to radio.



The most familiar example is a voice band modem that turns the digital data of a personal computer into modulated electrical signals in the voice frequency range of a telephone channel. These signals can be transmitted over telephone lines and demodulated by another modem at the receiver side to recover the digital data.

Modems are generally classified by the amount of data they can send in a given time unit, normally measured in bits per second (bit/s, or bps). They can also be classified by the symbol rate measured in baud, the number of times the modem changes its signal state per second. For example, the ITU V.21 standard used audio frequency-shift keying, aka tones, to carry 300 bit/s using 300 baud, whereas the original ITU V.22 standard allowed 1,200 bit/s with 600 baud using phase shift keying.



(modem) **(n.)** Short for **modulator-demodulator**. A modem is a device or program that enables a computer to transmit data over, for example, telephone or cable lines.

Fortunately, there is one standard interface for connecting external modems to computers called RS-232. Consequently, any external modem can be attached to any computer that has an RS-232 port, which almost all personal computers have. There are also modems that come as an expansion board that you can insert into a vacant expansion slot. These are sometimes called onboard or internal modems.

Notes

While the modem interfaces are standardized, a number of different protocols for formatting data to be transmitted over telephone lines exist. Some, like CCITT V.34, are official standards, while others have been developed by private companies. Most modems have built-in support for the more common protocols at slow data transmission speeds at least, most modems can communicate with each other. At high transmission speeds, however, the protocols are less standardized.

Aside from the transmission protocols that they support, the following characteristics distinguish one modem from another:

BPS : How fast the modem can transmit and receive data. At slow rates, modems are measured in terms of baud rates. The slowest rate is 300 baud (about 25 cps). At higher speeds, modems are measured in terms of bits per second (bps). The fastest modems run at 57,600 bps, although they can achieve even higher data transfer rates by compressing the data. Obviously, the faster the transmission rate, the faster you can send and receive data. Note, however, that you cannot receive data any faster than it is being sent. If, for example, the device sending data to your computer is sending it at 2,400 bps, you must receive it at 2,400 bps. It does not always pay, therefore, to have a very fast modem. In addition, some telephone lines are unable to transmit data reliably at very high rates.

Voice/data: Many modems support a switch to change between voice and data modes. In data mode, the modem acts like a regular modem. In voice mode, the modem acts like a regular telephone. Modems that support a voice/ data switch have a built-in loudspeaker and microphone for voice communication.

Auto-answer: An auto-answer modem enables your computer to receive calls in your absence. This is only necessary if you are offering some type of computer service that people can call in to use.

Data compression: Some modems perform data compression, which enables them to send data at faster rates. However, the modem at the receiving end must be able to decompress the data using the same compression technique.

Flash memory: Some modems come with flash memory rather than conventional ROM, which means that the communications protocols can be easily updated if necessary.

Fax capability: Most modern modems are fax modems, which means that they can send and receive faxes.



Modems grew out of the need to connect teletype machines over ordinary phone lines instead of more expensive leased lines which had previously been used for current loop-based teleprinters and automated telegraphs. George Stibitz connected a New Hampshire teletype to a computer in New York City by a subscriber telephone line in 1940.

6.3.1 Narrow-Band/Phone-Line Dialup Modems

A standard modem of today contains two functional parts: an analog section for generating the signals and operating the phone, and a digital section for setup and control. This functionality is often incorporated into a single chip nowadays, but the division remains in theory. In operation the modem can be in one of two modes, data mode in which data is sent to and from the computer over the phone lines, and command mode in which the modem listens to the data from the computer for commands, and carries them out. A typical session consists of powering up the modem (often inside the computer itself) which automatically assumes command mode, then sending it the command for dialing a number. After the connection is established to the remote modem, the modem automatically goes into data mode, and the user can send and receive data.

When the user is finished, the escape sequence, “+++” followed by a pause of about a second, may be sent to the modem to return it to command mode, then a command (e.g. “ATH”) to hang up the phone is sent. Note that on many modem controllers it is possible to issue commands to disable the escape sequence so that it is not possible for data being exchanged to trigger the mode change inadvertently.

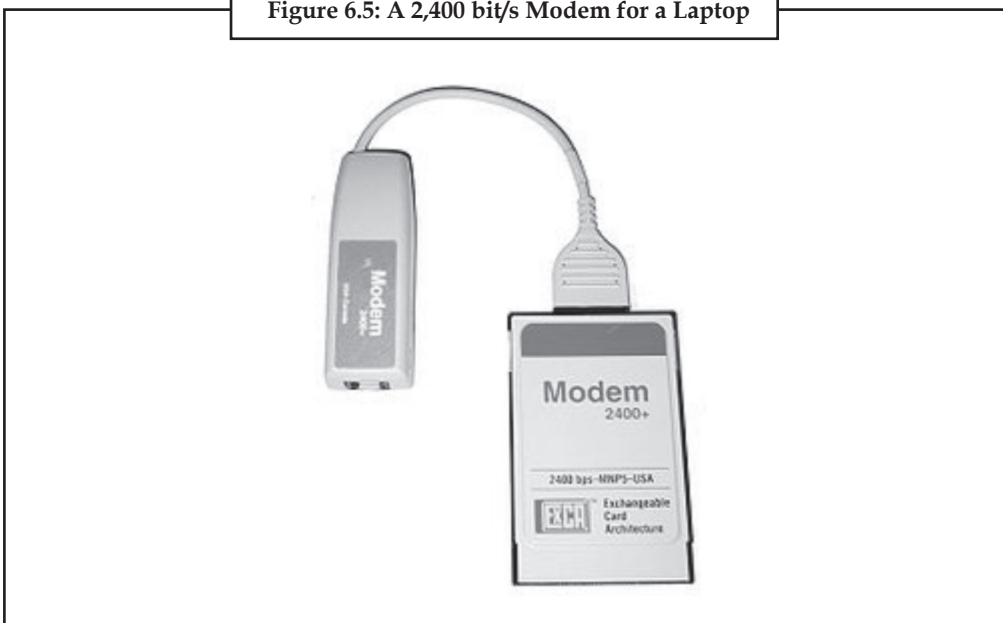
The commands themselves are typically from the Hayes command set, although that term is somewhat misleading. The original Hayes commands were useful for 300 bit/s operation only, and then extended for their 1,200 bit/s modems. Faster speeds required new commands, leading to a proliferation of command sets in the early 1990s. Things became considerably more standardized in the second half of the 1990s, when most modems were built from one of a very small number of chipsets. We call this the Hayes command set even today, although it has three or four times the numbers of commands as the actual standard.

6.3.1.1 Increasing Speeds (V.21, V.22, V.22bis)

The 300 bit/s modems used audio frequency-shift keying to send data. In this system the stream of 1s and 0s in computer data is translated into sounds which can be easily sent on the phone lines. In the Bell 103 system the originating modem sends 0s by playing a 1,070 Hz tone, and 1s at 1,270 Hz, with the answering modem putting its 0s on 2,025 Hz and 1s on 2,225 Hz. These frequencies were chosen carefully, they are in the range that suffer minimum distortion on the phone system, and also are not harmonics of each other.

In the 1,200 bit/s and faster systems, phase-shift keying was used. In this system the two tones for any one side of the connection are sent at the similar frequencies as in the 300 bit/s systems, but slightly out of phase. By comparing the phase of the two signals, 1s and 0s could be pulled back out, for instance if the signals were 90 degrees out of phase, this represented two digits, 1, 0, at 180 degrees it was 1, 1. In this way each cycle of the signal represents two digits instead of one. 1,200 bit/s modems were, in effect, 600 symbols per second modems (600 baud modems) with 2 bits per symbol.

Figure 6.5: A 2,400 bit/s Modem for a Laptop



Notes

Voiceband modems generally remained at 300 and 1,200 bit/s (V.21 and V.22) into the mid 1980s. A V.22bis 2,400-bit/s system similar in concept to the 1,200-bit/s Bell 212 signalling was introduced in the U.S., and a slightly different one in Europe. By the late 1980s, most modems could support all of these standards and 2,400-bit/s operation was becoming common.

6.3.1.2 Increasing Speeds (One-way Proprietary Standards)

Many other standards were also introduced for special purposes, commonly using a high-speed channel for receiving, and a lower-speed channel for sending. One typical example was used in the French Minitel system, in which the user's terminals spent the majority of their time receiving information. The modem in the Minitel terminal thus operated at 1,200 bit/s for reception, and 75 bit/s for sending commands back to the servers.

Three U.S. companies became famous for high-speed versions of the same concept. Telebit introduced its Trailblazer modem in 1984, which used a large number of 36 bit/s channels to send data one-way at rates up to 18,432 bit/s. A single additional channel in the reverse direction allowed the two modems to communicate how much data was waiting at either end of the link, and the modems could change direction on the fly. The Trailblazer modems also supported a feature that allowed them to spoof the UUCP g protocol, commonly used on Unix systems to send e-mail, and thereby speed UUCP up by a tremendous amount. Trailblazers thus became extremely common on Unix systems, and maintained their dominance in this market well into the 1990s.

U.S. Robotics (USR) introduced a similar system, known as HST, although this supplied only 9,600 bit/s (in early versions at least) and provided for a larger backchannel. Rather than offer spoofing, USR instead created a large market among Fidonet users by offering its modems to BBS sysops at a much lower price, resulting in sales to end users who wanted faster file transfers. Hayes was forced to compete, and introduced its own 9,600-bit/s standard, Express 96 (also known as Ping-Pong), which was generally similar to Telebit's PEP. Hayes, however, offered neither protocol spoofing nor sysop discounts, and its high-speed modems remained rare.

6.3.1.3 4,800 and 9,600 bit/s (V.27ter, V.32)

Echo cancellation was the next major advance in modem design. Local telephone lines use the same wires to send and receive, which results in a small amount of the outgoing signal bouncing back. This signal can confuse the modem, which was unable to distinguish between the echo and the signal from the remote modem. This was why earlier modems split the signal frequencies into 'answer' and 'originate'; the modem could then ignore its own transmitting frequencies. Even with improvements to the phone system allowing higher speeds, this splitting of available phone signal bandwidth still imposed a half-speed limit on modems.

Echo cancellation got around this problem. Measuring the echo delays and magnitudes allowed the modem to tell if the received signal was from itself or the remote modem, and create an equal and opposite signal to cancel its own. Modems were then able to send over the whole frequency spectrum in both directions at the same time, leading to the development of 4,800 and 9,600 bit/s modems.

Increases in speed have used increasingly complicated communications theory. 1,200 and 2,400 bit/s modems used the phase shift key (PSK) concept. This could transmit two or three bits per symbol. The next major advance encoded four bits into a combination of amplitude and phase, known as Quadrature Amplitude Modulation (QAM). Best visualized as a constellation diagram, the bits are mapped onto points on a graph with the x (real) and y (quadrature) coordinates transmitted over a single carrier.

The new V.27ter and V.32 standards were able to transmit 4 bits per symbol, at a rate of 1,200 or 2,400 baud, giving an effective bit rate of 4,800 or 9,600 bit/s. The carrier frequency was 1,650 Hz. For many years, most engineers considered this rate to be the limit of data communications over telephone networks.

6.3.1.4 Error Correction and Compression

Operations at these speeds pushed the limits of the phone lines, resulting in high error rates. This led to the introduction of error-correction systems built into the modems, made most famous with Microcom's MNP systems. A string of MNP standards came out in the 1980s, each increasing the effective data rate by minimizing overhead, from about 75% theoretical maximum in MNP 1, to 95% in MNP 4. The new method called MNP 5 took this a step further, adding data compression to the system, thereby increasing the data rate above the modem's rating. Generally the user could expect an MNP5 modem to transfer at about 130% the normal data rate of the modem. Details of MNP were later released and became popular on a series of 2,400-bit/s modems, and ultimately led to the development of V.42 and V.42bis ITU standards. V.42 and V.42bis were non-compatible with MNP but were similar in concept: Error correction and compression.

Another common feature of these high-speed modems was the concept of fallback, or speed hunting, allowing them to talk to less-capable modems. During the call initiation the modem would play a series of signals into the line and wait for the remote modem to respond to them. They would start at high speeds and progressively get slower and slower until they heard an answer. Thus, two USR modems would be able to connect at 9,600 bit/s, but, when a user with a 2,400-bit/s modem called in, the USR would fallback to the common 2,400-bit/s speed. This would also happen if a V.32 modem and a HST modem were connected. Because they used a different standard at 9,600 bit/s, they would fall back to their highest commonly supported standard at 2,400 bit/s. The same applies to V.32bis and 14,400 bit/s HST modem, which would still be able to communicate with each other at only 2,400 bit/s.

6.3.1.5 Breaking the 9.6k Barrier

In 1980, Gottfried Ungerboeck from IBM Zurich Research Laboratory applied powerful channel coding techniques to search for new ways to increase the speed of modems. His results were astonishing but only conveyed to a few colleagues. Finally in 1982, he agreed to publish what is now a landmark paper in the theory of information coding. By applying powerful parity check coding to the bits in each symbol, and mapping the encoded bits into a two-dimensional diamond pattern, Ungerboeck showed that it was possible to increase the speed by a factor of two with the same error rate. The new technique was called mapping by set partitions (now known as trellis modulation).

Error correcting codes, which encode code words (sets of bits) in such a way that they are far from each other, so that in case of error they are still closest to the original word (and not confused with another) can be thought of as analogous to sphere packing or packing pennies on a surface: the further two bit sequences are from one another, the easier it is to correct minor errors.

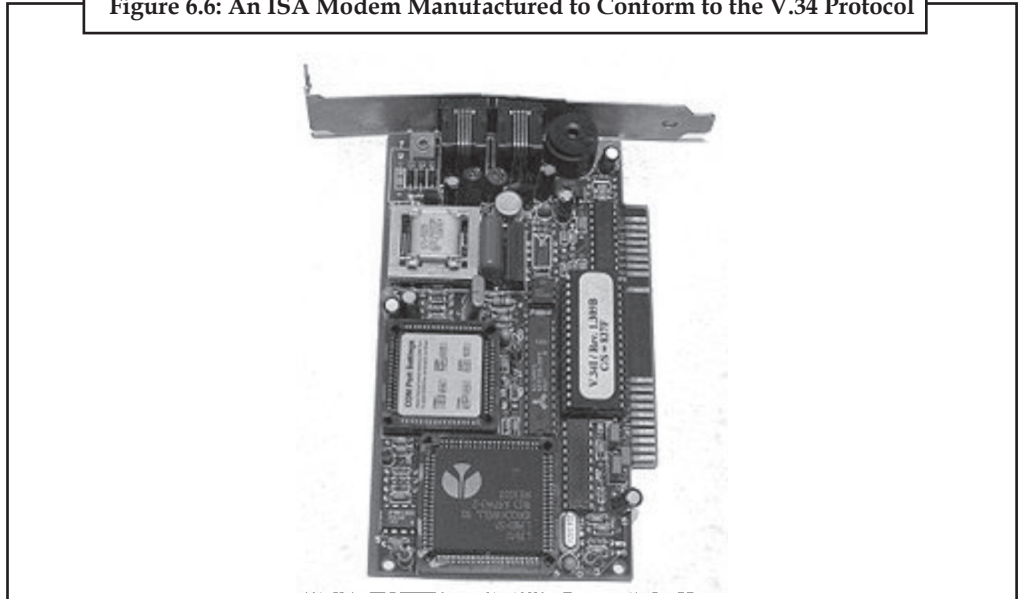
V.32bis was so successful that the older high-speed standards had little to recommend them. USR fought back with a 16,800 bit/s version of HST, while AT&T introduced a one-off 19,200 bit/s method they referred to as V.32ter (also known as V.32 terbo or tertiary), but neither non-standard modem sold well.

6.3.1.6 V.34/28.8k and 33.6k

Any interest in these systems was destroyed during the lengthy introduction of the 28,800 bit/s V.34 standard. While waiting, several companies decided to release hardware and introduced modems they referred to as V.FAST. In order to guarantee compatibility with V.34 modems once the standard was ratified (1994), the manufacturers were forced to use more flexible parts, generally a DSP and microcontroller, as opposed to purpose-designed ASIC modem chips.

Notes

Figure 6.6: An ISA Modem Manufactured to Conform to the V.34 Protocol



Today, the ITU standard V.34 represents the culmination of the joint efforts. It employs the most powerful coding techniques including channel encoding and shape encoding. From the mere 4 bits per symbol (9.6 kbit/s), the new standards used the functional equivalent of 6 to 10 bits per symbol, plus increasing baud rates from 2,400 to 3,429, to create 14.4, 28.8, and 33.6 kbit/s modems. This rate is near the theoretical Shannon limit. When calculated, the Shannon capacity of a narrowband line is, with the (linear) signal-to-noise ratio. Narrowband phone lines have a bandwidth from 300-4000 Hz, so using (SNR = 30dB): capacity is approximately 35 kbit/s.

Without the discovery and eventual application of trellis modulation, maximum telephone rates using voice-bandwidth channels would have been limited to $3,429 \text{ baud} \times 4 \text{ bit/symbol} \approx$ approximately 14 kbit/s using traditional QAM. (DSL makes use of the bandwidth of traditional copper-wire twisted pairs between subscriber and the central office, which far exceeds that of analog voice circuitry.)

6.3.1.7 V.61/V.70 Analog/Digital Simultaneous Voice and Data

The V.61 Standard introduced Analog Simultaneous Voice and Data (ASVD). This technology allowed users of v.61 modems to engage in point-to-point voice conversations with each other while their respective modems communicated.

In 1995, the first DSVD (Digital Simultaneous Voice and Data) modems became available to consumers, and the standard was ratified as v.70 by the International Telecommunication Union (ITU) in 1996.

Two DSVD modems can establish a completely digital link between each other over standard phone lines. Sometimes referred to as "the poor man's ISDN," and employing a similar technology, v.70 compatible modems allow for a maximum speed of 33.6 kbps between peers. By using a majority of the bandwidth for data and reserving part for voice transmission, DSVD modems allow users to pick up a telephone handset interfaced with the modem, and initiate a call to the other peer.

One practical use for this technology was realized by early two player video gamers, who could hold voice communication with each other while in game over the PSTN.

Advocates of DSVD envisioned whiteboard sharing and other practical applications for the standard, however, with advent of cheaper 56kbps analog modems intended for Internet connectivity, peer-to-peer data transmission over the PSTN became quickly irrelevant. Also, the

standard was never expanded to allow for the making or receiving of arbitrary phone calls while the modem was in use, due to the cost of infrastructure upgrades to telephone companies, and the advent of ISDN and DSL technologies which effectively accomplished the same goal.

Today, Multi-Tech is the only known company to continue to support a v.70 compatible modem. While their device also offers v.92 at 56kbps, it remains significantly more expensive than comparable modems sans v.70 support.

6.3.1.8 Using Digital Lines and PCM (V.90/92)

In the late 1990s Rockwell/Lucent and U.S. Robotics introduced new competing technologies based upon the digital transmission used in modern telephony networks. The standard digital transmission in modern networks is 64 kbit/s but some networks use a part of the bandwidth for remote office signaling (e.g., to hang up the phone), limiting the effective rate to 56 kbit/s DS0. This new technology was adopted into ITU standards V.90 and is common in modern computers. The 56 kbit/s rate is only possible from the central office to the user site (downlink). In the United States, government regulation limits the maximum power output, resulting in a maximum data rate of 53.3 kbit/s. The uplink (from the user to the central office) still uses V.34 technology at 33.6 kbit/s.

Figure 6.7: Modem Bank at an ISP



Later in V.92, the digital PCM technique was applied to increase the upload speed to a maximum of 48 kbit/s, but at the expense of download rates. For example a 48 kbit/s upstream rate would reduce the downstream as low as 40 kbit/s, due to echo on the telephone line. To avoid this problem, V.92 modems offer the option to turn off the digital upstream and instead use a 33.6 kbit/s analog connection, in order to maintain a high digital downstream of 50 kbit/s or higher. V.92 also adds two other features. The first is the ability for users who have call waiting to put their dial-up Internet connection on hold for extended periods of time while they answer a call. The second feature is the ability to quickly connect to one's ISP. This is achieved by remembering the analog and digital characteristics of the telephone line, and using this saved information to reconnect at a fast pace.

6.3.1.9 Using Compression to Exceed 56k

Today's V.42, V.42bis and V.44 standards allow the modem to transmit data faster than its basic rate would imply. For instance, a 53.3 kbit/s connection with V.44 can transmit up to $53.3 \times 6 = 320$ kbit/s using pure text. However, the compression ratio tends to vary due to noise on the line, or due to the transfer of already-compressed files (ZIP files, JPEG images, MP3 audio, MPEG video). At some points the modem will be sending compressed files at approximately 50 kbit/s, uncompressed files at 160 kbit/s, and pure text at 320 kbit/s, or any value in between.

Notes

In such situations a small amount of memory in the modem, a buffer, is used to hold the data while it is being compressed and sent across the phone line, but in order to prevent overflow of the buffer, it sometimes becomes necessary to tell the computer to pause the data stream. This is accomplished through hardware flow control using extra lines on the modem-computer connection. The computer is then set to supply the modem at some higher rate, such as 320 kbit/s, and the modem will tell the computer when to start or stop sending data.

6.3.1.10 Compression by the ISP

As telephone-based 56k modems began losing popularity, some Internet service providers such as Netzero and Juno started using pre-compression to increase the throughput and maintain their customer base. As example, the Netscape ISP uses a compression program that squeezes images, text, and other objects at the modem server, just prior to sending them across the phone line. Certain content using lossy compression (e.g., images) may be recompressed (transcoded) using different parameters to the compression algorithm, making the transmitted content smaller but of lower quality. The server-side compression operates much more efficiently than the on-the-fly compression of V.44-enabled modems due to the fact that V.44 is a generalized compression algorithm whereas other compression techniques are application-specific (JPEG, MPEG, Vorbis, etc.). Typically Website text is compacted to 4% thus increasing effective throughput to approximately 1,300 kbit/s. The accelerator also pre-compresses Flash executables and images to approximately 30% and 12%, respectively.

The drawback of this approach is a loss in quality, where the GIF and JPEG images are lossy compressed, which causes the content to become pixelated and smeared. However the speed is dramatically improved such that Web pages load in less than 5 seconds, and the user can manually choose to view the uncompressed images at any time. The ISPs employing this approach advertise it as “surf 5× faster” or simply “accelerated dial-up”.

6.3.1.11 List of Dialup Speeds

Note that the values given are maximum values, and actual values may be slower under certain conditions (for example, noisy phone lines). For a complete list see the companion article list of device bandwidths. A baud is one symbol per second; each symbol may encode one or more data bits.

Connection	<u>Bitrate (kbit/s)</u> Year Released	
110 baud Bell 101 modem	0.1	1958
300 baud (<u>Bell 103</u> or <u>V.21</u>)	0.3	1962
1200 modem (1200 baud) (<u>Bell 202</u>)	1.2	
1200 Modem (600 baud) (Bell 212A or <u>V.22</u>)	1.2	
2400 Modem (600 baud) (<u>V.22bis</u>)	2.4	
2400 Modem (1200 baud) (<u>V.26bis</u>)	2.4	
4800 Modem (1600 baud) (<u>V.27ter</u>)	4.8	
9600 Modem (2400 baud) (<u>V.32</u>)	9.6	
14.4k Modem (2400 baud) (<u>V.32bis</u>)	14.4	
28.8k Modem (3200 baud) (<u>V.34</u>)	28.8	

		Notes
33.6k Modem (3429 baud) (<u>V.34</u>)	33.6	
56k Modem (8000/3429 baud) (<u>V.90</u>)	56.0/33.6	
56k Modem (8000/8000 baud) (V.92)	56.0/48.0	
Bonding modem (two 56k modems)) (V.92)	112.0/96.0	
Hardware compression (variable) (<u>V.90/V.42bis</u>)	56.0-220.0	
Hardware compression (variable) (V.92/ <u>V.44</u>)	56.0-320.0	
Server-side web compression (variable) (Netscape ISP)	100.0-1,000.0	

6.3.2 Radio Modems

Direct broadcast satellite, WiFi, and mobile phones all use modems to communicate, as do most other wireless services today. Modern telecommunications and data networks also make extensive use of radio modems where long distance data links are required. Such systems are an important part of the PSTN, and are also in common use for high-speed computer network links to outlying areas where fibre is not economical.

Even where a cable is installed, it is often possible to get better performance or make other parts of the system simpler by using radio frequencies and modulation techniques through a cable. Coaxial cable has a very large bandwidth, however signal attenuation becomes a major problem at high data rates if a digital signal is used. By using a modem, a much larger amount of digital data can be transmitted through a single piece of wire. Digital cable television and cable Internet services use radio frequency modems to provide the increasing bandwidth needs of modern households. Using a modem also allows for frequency-division multiple access to be used, making full-duplex digital communication with many users possible using a single wire.

Wireless modems come in a variety of types, bandwidths, and speeds. Wireless modems are often referred to as transparent or smart. They transmit information that is modulated onto a carrier frequency to allow many simultaneous wireless communication links to work simultaneously on different frequencies.

6.3.2.1 WiFi and WiMax

Wireless data modems are used in the WiFi and WiMax standards, operating at microwave frequencies.

WiFi is principally used in laptops for Internet connections (wireless access point) and wireless application protocol (WAP).

6.3.3 Mobile Modems and Routers

Modems which use a mobile telephone system (GPRS, UMTS, HSPA, EVDO, WiMax, etc.), are known as wireless modems (sometimes also called cellular modems). Wireless modems can be embedded inside a laptop or appliance or external to it. External wireless modems are connect cards, usb modems for mobile broadband and cellular routers. A connect card is a PC card or ExpressCard which slides into a PCMCIA/PC card/ ExpressCard slot on a computer. USB wireless modems use a USB port on the laptop instead of a PC card or ExpressCard slot. A cellular router may have an external datacard (AirCard) that slides into it. Most cellular routers do allow such datacards or USB modems. Cellular Routers may not be modems per se, but they contain modems or allow modems to be slid into them. The difference between a cellular router and a wireless modem is that a cellular router normally allows multiple

Notes

people to connect to it (since it can route, or support multipoint to multipoint connections), while the modem is made for one connection.

Figure 6.8: T-Mobile Universal Mobile Telecommunications System PC Card Modem



Most of the GSM wireless modems come with an integrated SIM cardholder (i.e., Huawei E220, Sierra 881, etc.) and some models are also provided with a microSD memory slot and/or jack for additional external antenna such as Huawei E1762 and Sierra Wireless Compass 885. The CDMA (EVDO) versions do not use R-UIM cards, but use Electronic Serial Number (ESN) instead.

Figure 6.9: Huawei CDMA2000 Evolution-Data Optimized USB Wireless Modem



The cost of using a wireless modem varies from country to country. Some carriers implement flat rate plans for unlimited data transfers. Some have caps (or maximum limits) on the amount of data that can be transferred per month. Other countries have plans that charge a fixed rate per data transferred per megabyte or even kilobyte of data downloaded; this tends to add up quickly in today's content-filled world, which is why many people are pushing for flat data rates.

The faster data rates of the newest wireless modem technologies (UMTS, HSPA, EVDO, WiMax) are also considered to be broadband wireless modems and compete with other broadband modems below.

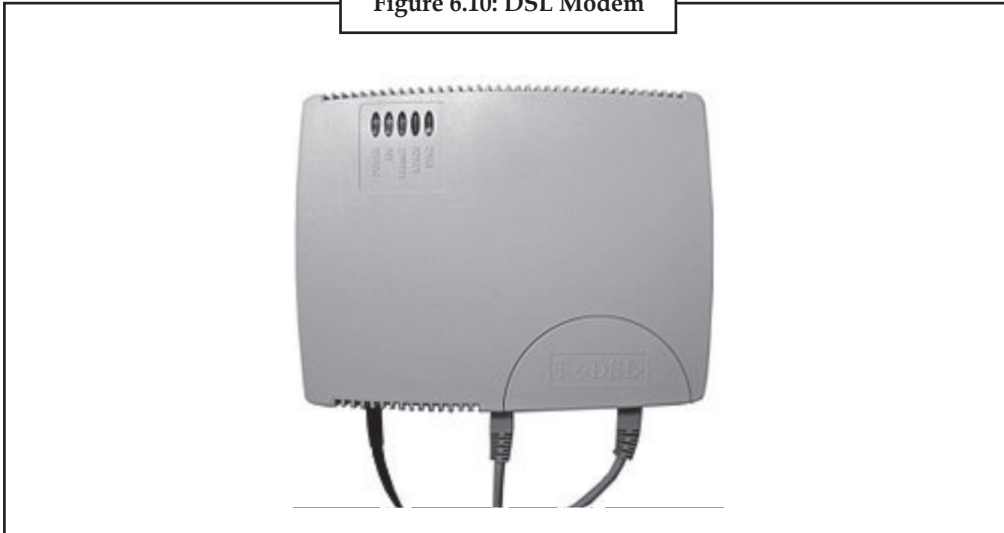
6.3.4 Broadband

ADSL modems, a more recent development, are not limited to the telephone's voiceband audio frequencies. Some ADSL modems use coded orthogonal frequency division modulation (DMT, for Discrete MultiTone; also called COFDM, for digital TV in much of the world).

Notes

Cable modems use a range of frequencies originally intended to carry RF television channels. Multiple cable modems attached to a single cable can use the same frequency band, using a low-level media access protocol to allow them to work together within the same channel. Typically, 'up' and 'down' signals are kept separate using frequency division multiple access.

Figure 6.10: DSL Modem



New types of broadband modems are beginning to appear, such as doubleway satellite and power line modems.

Broadband modems should still be classed as modems, since they use complex waveforms to carry digital data. They are more advanced devices than traditional dial-up modems as they are capable of modulating/demodulating hundreds of channels simultaneously.

Many broadband modems include the functions of a router (with Ethernet and WiFi ports) and other features such as DHCP, NAT and firewall features.

When broadband technology was introduced, networking and routers were unfamiliar to consumers. However, many people knew what a modem was as most internet access was through dial-up. Due to this familiarity, companies started selling broadband modems using the familiar term modem rather than vaguer ones like adapter or transceiver, or even "bridge".

Many broadband modems must be configured in bridge mode before they can use a router.

6.3.5 Home Networking

Although the name modem is seldom used in this case, modems are also used for high-speed home networking applications, specially those using existing home wiring. One example is the G.hn standard, developed by ITU-T, which provides a high-speed (up to 1 Gbit/s) Local area network using existing home wiring (power lines, phone lines and coaxial cables). G.hn devices use orthogonal frequency-division multiplexing (OFDM) to modulate a digital signal for transmission over the wire.

The phrase "Null modem" was used to describe attaching a specially wired cable between the serial ports of two personal computers. Basically, the transmit output of one computer was wired to the receive input of the other; this was true for both computers. The same software used with modems (such as Procomm or Minicom) could be used with the null modem connection.

Notes

6.3.6 Deep-space Telecommunications

Many modern modems have their origin in deep space telecommunications systems of the 1960s.

Differences between deep space telecom modems and landline modems:

- (a) digital modulation formats that have high doppler immunity are typically used
- (b) waveform complexity tends to be low, typically binary phase shift keying
- (c) error correction varies mission to mission, but is typically much stronger than most landline modems.

6.3.7 Voice Modem

Voice modems are regular modems that are capable of recording or playing audio over the telephone line. They are used for telephony applications. See Voice modem command set for more details on voice modems. This type of modem can be used as an FXO card for Private branch exchange systems (compare V.92).

6.4 Data Communication using Digital Data Connections

A communications system may be digital either by the nature of the information (also known as data) which is passed or in the nature of the signals which are transmitted. If either of these is digital then for our purposes it is considered to be a digital communications system. There are four possible combinations of data and signal types:

1. Analog data, analog signal;
2. Digital data, analog signal;
3. Analog data, digital signal;
4. Digital data, digital signal.

6.4.1 Digital Data with Analog Signals

This method is used to send computer information over transmission channels that require analog signals, like a fiber optic networks, computer modems, cellular phone networks, and satellite systems. In each of this systems, an electromagnetic carrier wave is used to carry the information over great distances and connect digital information users at remote locations. The digital data is used to modulate one or more of the parameters of the carrier wave, This basic process is given the name “shift-keying” to differentiate it from the purely analog systems like AM and FM. As with analog modulation, there are three parameters of the carrier wave to vary and therefore three basic types of shift keying:

1. Amplitude Shift Keying (ASK)
2. Frequency Shift Keying (FSK), and
3. Phase Shift Keying (PSK).

6.4.1.1 ASK

In amplitude shift keying, the carrier wave amplitude is changed between discrete levels (usually two) in accordance with the digital data.

The digital data to be transmitted is the binary number 1011. Two amplitudes are used to directly represent the data, either 0 or 1. In this case, the modulation is called binary amplitude shift keying or BASK. The signal is divided into four pulses of equal duration which represent the bits in the digital data. The number of bits used for each character is a function of the system, but is typically eight, seven of which represent the 128 possible characters, the last bit is used to check for errors, and is explained at the end of this chapter.

6.4.1.2 FSK

In frequency shift keying, the carrier frequency is changed between discrete values. If only two frequencies are used then this will be called BFSK, for binary frequency shift keying.

6.4.1.3 PSK

The phase of the carrier wave at the beginning of the pulse is changed between discrete values. This particular case is the same code shown above but in BPSK.

6.4.1.4 M-ary Frequency/Phase Keying

In binary shift keying, there were only two choices for the parameter of the carrier wave which was varied in accordance with the digital data. In BASK, there are two possibilities for amplitude, which corresponded to zero and one. Likewise for BFSK and BPSK. This matches nicely with the binary number system, which also uses two possibilities for each bit, 0 and 1. It is possible to increase the data transfer rate by putting more choices into each bit. As an example, 4-ary (or Quaternary PSK) uses four different phases: 0, 90, 180, and 270 degrees. This gives four possible values at each pulse, corresponding to two independent streams (channels) of data. Likewise, 16-ary FSK can send four channels of data at the same time.

6.4.1.5 Amplitude-Phase Keying

This process uses combinations of amplitude and phase keying. For example, if we use two levels of amplitude and two levels of phase together, there will be a total of four possibilities. This is used to transmit two independent channels of digital data simultaneously. This particular case is called Quadrature AM or Quaternary PSK. They are identical, although it may not be obvious at this level. Because of the equivalence, the basic process is called amplitude-phase keying. This process may be extended to higher numbers of possibilities. The completely general term is M-ary APK, which is not specific about which parameter has which number of possibilities. 16-APK may have 2 amplitudes and 8 phases or 4 each, it matters little. The upshot is that the number of separate channels that can be sent simultaneously is increased. If M designates the number of possible combinations, from the M-ary APK system, then the number of channels of digital data that may be transmitted simultaneously is given by

$$N = \log_2 M$$

6.4.1.6 Capacity

All of these methods which utilize a sequence of equally spaced pulses to modulate a carrier wave have similar bandwidths. The bandwidth determined by the duration of each pulse, designated as t_d . It is a general result, that the minimum bandwidth required to create this pulse, W , is given by

$$W = 1/(2t_d)$$

Notes

Given a specific bandwidth limitation, the rate at which data can be transferred can be determined. If the bandwidth is W (in Hz), and the modulation type is M -ary, the rate at which data can be transferred, given in bits per second (also known as the baud rate), R , is given by

$$R = W \log_2 M$$

It would now appear that the free lunch principle (i.e. there is none) has been violated. Given the same bandwidth, which is determined by the pulse duration, the data rate may be extended by using a higher M -ary modulation type. As you may suspect, this will not succeed indefinitely. Ultimately, increasing the bit content of each pulse has the effect of lowering the signal-to-noise ratio. A way to illustrate this is to consider M -ary FSK. Starting with BFSK, the bandwidth limits the difference between the two frequencies. If the same interval is further subdivided to make 16-ary FSK, the difference between any two adjacent frequencies has been reduced by $1/16$ making it more difficult to tell them apart (especially in the presence of noise). This is quantified as a reduction in the signal-to-noise ratio. This is also true for all other M -ary systems. Continued operation of a system with low SNR will lead to an increase in the error rate

Probability of Error as SNR

Clearly, the data rate cannot be increased indefinitely without affecting performance. This result is expressed in the Hartley-Shannon law for the capacity:

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

where:

C = capacity in bits per second (bps)

S/N = signal-to-noise ratio (depends of modulation type and noise).

6.4.1.7 Minimum Shift Keying (MSK)

This is a technique used to find the minimum signal bandwidth for a particular method (usually FSK). In BFSK, the two frequencies are not chosen to be far enough apart, then it will become impossible to discriminate the two levels. The condition for the difference in frequencies, D_{fMSK} , such that the two levels can be determined accurately is

$$D_{fMSK} = 1/(4t_d)$$

where t_d is the pulse duration as previously discussed. MSK is considered to be the most efficient way to use a given bandwidth. It maximizes the reliability (which is related to S/N) within a given bandwidth.

6.4.2 Analog Data with Digital Signals

A digital signal can be transmitted over a dedicated connection between two or more users. In order to transmit analog data, it must first be converted into a digital form. This process is called sampling, or encoding. Sampling involves two steps:

- (a) Take measurements at regular sampling intervals, and
- (b) Convert the value of the measurement into binary code.

6.4.2.1 Sampling

The amplitude of a signal is measured at regular intervals. The interval is designated as t_s , and is called the sample interval. The sample interval must be chosen to be short enough that the signal does not change greatly between measurements. The sampling rate, which is the inverse of the sample interval should be greater than twice the highest frequency component of the signal which is being sampled. This sample rate is known as the Nyquist frequency. If you sample at a lower rate, you run the risk of missing some information, known as aliasing.

6.4.2.2 Encoding

Once the samples are obtained, they must be encoded into binary. For a given number of bits, each sample may take on only a finite number of values. This limits the resolution of the sample. If more bits are used for each sample, then a higher degree of resolution is obtained. For example, if the sampling is 8-bit, each sample may only take on 256 different values. 16-bit sampling would give 65,536 unique values for the signal in each sample interval. Higher bit sampling requires more storage for data and requires more bandwidth to transmit.

6.4.3 Digital Data with Digital Signals

We have already discussed how computers use a binary number system to perform operations. In its simplest form, digital data is a collection of zeroes and ones, where the value at any one time is called a bit. In order for two digital users (like computers) to communicate there must be an agreement on the format used. There are several different ways in which a binary number may be formatted. This is called pulse code modulation or PCM. The most straightforward PCM format is designated as NRZ-L, for non return to zero level. In this format, the level directly represents the binary value: low level = 0, high level = 1.

There are many other varieties, which are explained below:

- (a) NRZ-M (non return to zero mark). 1: no change in level from last pulse. 0: level changes from last pulse.
- (b) NRZ-S (non return to zero space). This is the same as NRZ-M but with the logic levels reversed. 1: level changes from last pulse. 0: no change in level from last pulse.
- (c) Bi-Phase-L (bi-phase level). The level always changes in the middle of the pulse. 1: level changes from high to low. 0: level changes from low to high.
- (d) Bi-Phase-M (bi-phase mark). The level always changes at the beginning of each pulse. 1: level changes in the middle of the pulse. 0: no level change in the middle of the pulse.
- (e) Bi-Phase-S (bi-phase space). This is the same as Bi-Phase-L but with the logic levels reversed. 1: no level change in middle of pulse. 0: level changes in the middle of the pulse.
- (f) DBi-Phase-M (differential bi-phase mark). The level always changes in the middle of the pulse. 1: no level change at beginning of the pulse. 0: level change at beginning of the pulse.
- (g) DBi-Phase-S (differential bi-phase space). This is the same as DBi-phase-M but with the logic levels reversed. 1: level change at beginning of the pulse. 0: no level change at the beginning of the pulse.

Notes

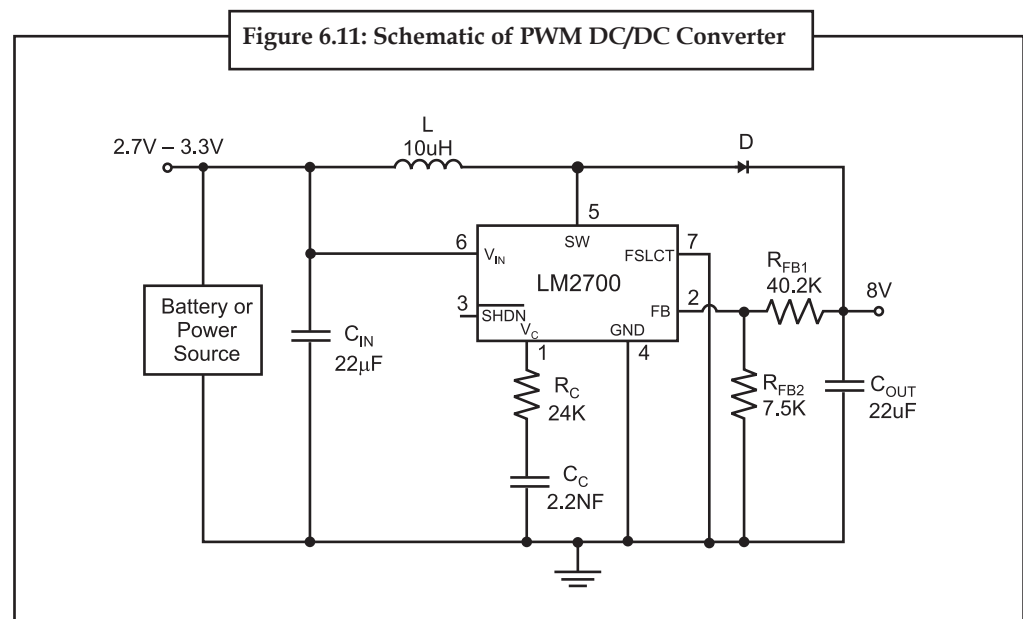
6.4.3.1 Parity Checksum

It is possible for an error to occur somewhere in the transmission process. One way to increase the reliability of transmitted PCM signals is to add a checksum bit to each piece of data. For example, in an eight-bit byte, seven of the bits can be used for data and the last reserved for a checksum bit. In one method, the checksum bit is determined by parity (meaning an even or odd number). In even parity checksums, a 0 or 1 is added to make the overall number of ones (including the checksum) even. In odd parity, a 0 or 1 is added to make the overall number of ones odd.

6.4.4 Some Digital Data Connection Methods

6.4.4.1 LM2700 Step-Up PWM DC/DC Converter Datasheet

LM2700 step-up PWM DC/DC converter is an ideal part for biasing LCD displays. It features internal switch, pin selectable <http://datasheetoo.com/wp-content/uploads/2009/12/Step-Up-PWM-DC-DC-Converter-Datasheet-and-Circuit-Diagram.jpg> frequency for easy filtering and low noise, over temperature protection circuit and adjustable output voltage up to 17.5V. It used to be applied on applications such as handheld, GSM/CDMA Phones, digital cameras and portable applications.

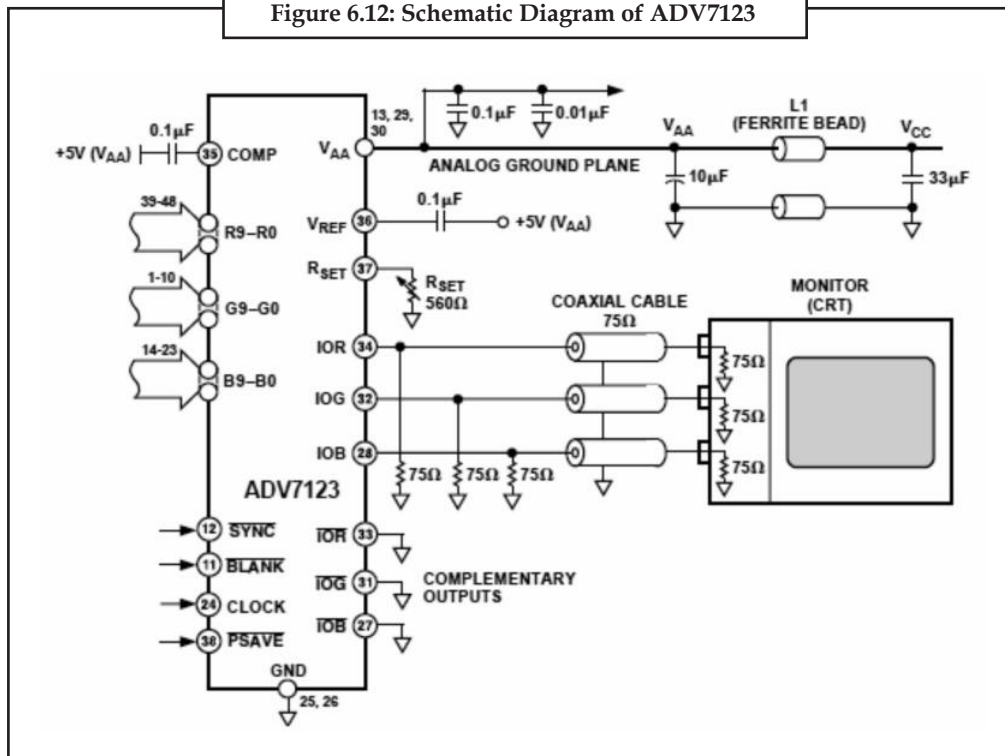


Find sections such the typical application circuit, 14-Lead TSSOP or LLP package connection diagram, pin description, operation circuit example of LM2700 (continuous conduction mode, setting the output voltage, DC gain and Loop gain, inductor and diode selection, input and output capacitor selection, and etc), and the application circuit information related to the operation functions in the datasheet.

6.4.4.2 ADV7123 Digital-to-Analog Converter Connection Diagram and Datasheet

This digital-to-analog converter (DAC) integrated circuit is designed for lowest noise performance, both radiated and conducted noise.

Figure 6.12: Schematic Diagram of ADV7123



According to the ADV7123 datasheet, this device consists of three high speed, 10-bit, video DACs with complementary outputs, a standard TTL input interface, and a high impedance, analog output current source. It used to be applied in digital video systems, image processing, digital radio modulation, color graphics and more.



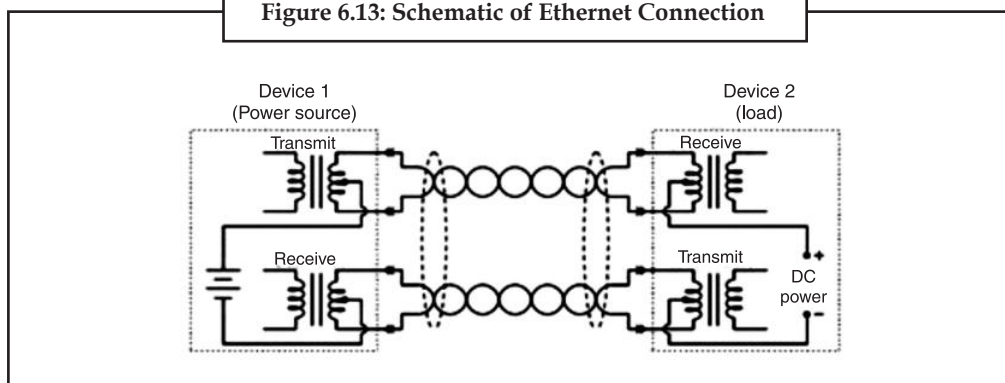
Task

Explain the schematic diagram of ADV7123.

6.4.4.3 Ethernet DC Power and Digital Data Connection Circuit Schematic

You can see the schematic shown how two Ethernet (802.3af) devices connect together via a “Cat 5” twisted-pair cable.

Figure 6.13: Schematic of Ethernet Connection



Notes

The 802.3af Ethernet standard allowing both DC power and digital data to be communicated over the pairs of wires.

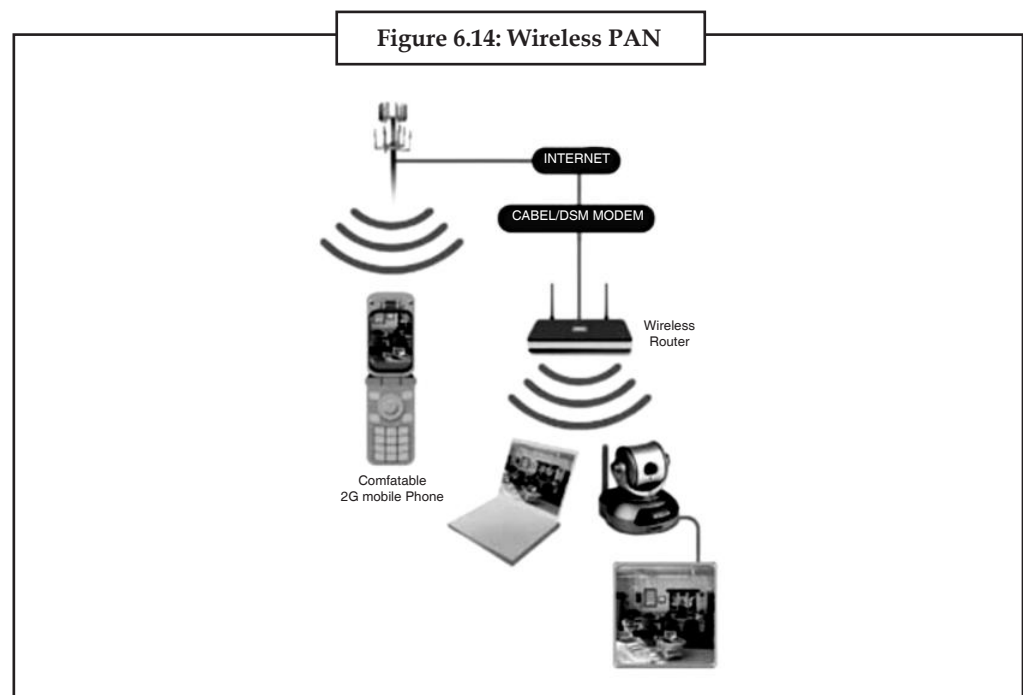
6.5 Wireless Networks

Wireless network refers to any type of computer network that is not connected by cables of any kind. It is a method by which telecommunications networks and enterprise (business), installations avoid the costly process of introducing cables into to a building, or as a connection between various equipment locations. Wireless telecommunications networks are generally implemented and administered using a transmission system called radio waves. This implementation takes place at the physical level, (layer), of the network structure.

6.5.1 Types of Wireless Connections

6.5.1.1 Wireless PAN

Wireless Personal Area Networks (WPANs) interconnect devices within a relatively small area, generally within a persons reach. For example, both Bluetooth radio and invisible Infrared light provides a WPAN for interconnecting a headset to a laptop. ZigBee also supports WPAN applications. Wi-Fi PANs are becoming commonplace (2010) as equipment designers start to integrate Wi-Fi into a variety of consumer electronic devices. Intel “My WiFi” and Windows 7 “virtual Wi-Fi” capabilities have made Wi-Fi PANs simpler and easier to set up and configure.

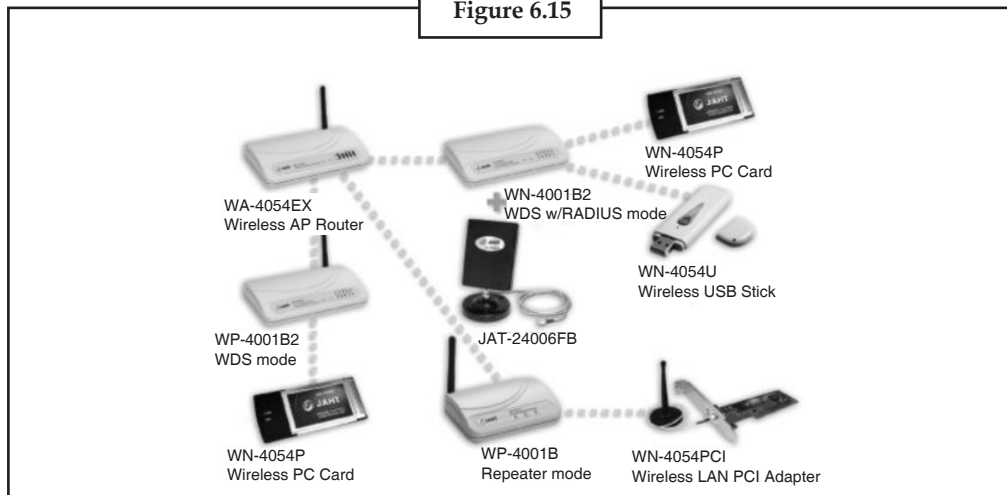


6.5.1.2 Wireless LAN

A wireless local area network (WLAN) links two or more devices using a wireless distribution method, providing a connection through an access point to the wider internet. The use of spread-spectrum or OFDM technologies also gives users the mobility to move around within a local coverage area, and still remain connected to the network.

- **Wi-Fi:** “Wi-Fi” is a term used to describe 802.11 WLANs, although it is technically a declared standard of interoperability between 802.11 devices.
- **Fixed Wireless Data:** This implements point to point links between computers or networks at two distant locations, often using dedicated microwave or modulated laser light beams over line of sight paths. It is often used in cities to connect networks in two or more buildings without installing a wired link.

Figure 6.15

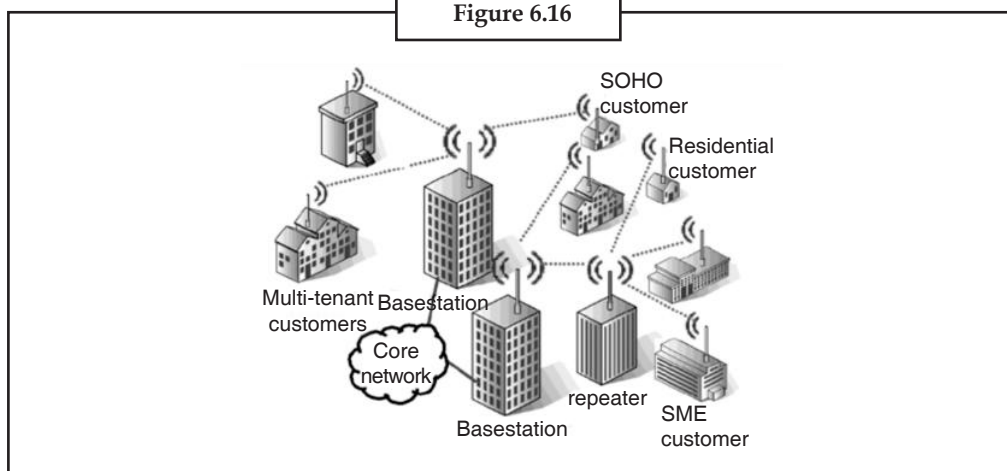


6.5.1.3 Wireless MAN

Wireless Metropolitan Area Networks are a type of wireless network that connects several Wireless LANs.

- WiMAX is a type of Wireless MAN and is described by the IEEE 802.16 standard.

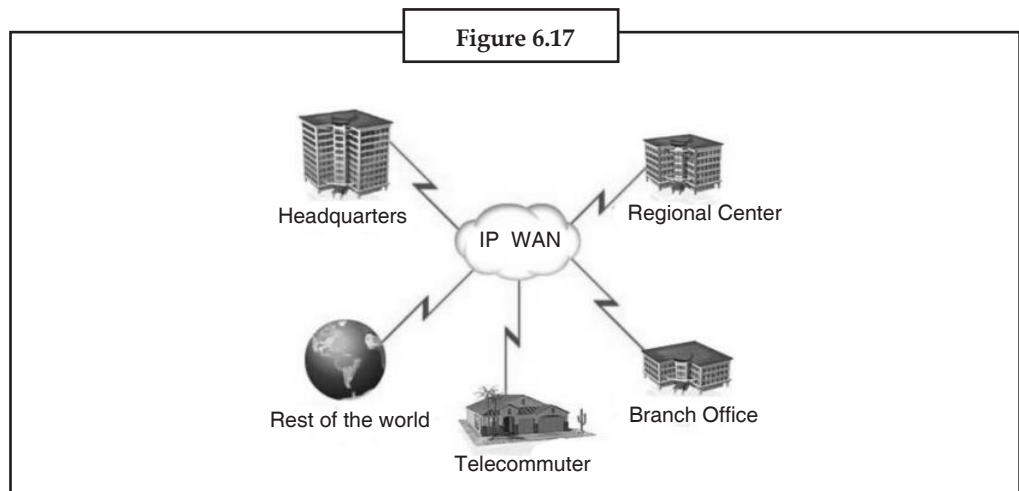
Figure 6.16



6.5.1.4 Wireless WAN

Wireless wide area networks are wireless networks that typically cover large outdoor areas. These networks can be used to connect branch offices of business or as a public internet access system. They are usually deployed on the 2.4 GHz band. A typical system contains base station gateways, access points and wireless bridging relays. Other configurations are mesh systems where each access point acts as a relay also. When combined with renewable energy systems such as photovoltaic solar panels or wind systems they can be stand alone systems.

Notes



6.5.1.5 Mobile Devices Networks

With the development of smart phones, cellular telephone networks routinely carry data in addition to telephone conversations:

- (a) **Global System for Mobile Communications (GSM):** The GSM network is divided into three major systems: the switching system, the base station system, and the operation and support system. The cell phone connects to the base system station which then connects to the operation and support station; it then connects to the switching station where the call is transferred to where it needs to go. GSM is the most common standard and is used for a majority of cell phones.
- (b) **Personal Communications Service (PCS):** PCS is a radio band that can be used by mobile phones in North America and South Asia. Sprint happened to be the first service to set up a PCS.
- (c) **D-AMPS:** Digital Advanced Mobile Phone Service, an upgraded version of AMPS, is being phased out due to advancement in technology. The newer GSM networks are replacing the older system.

6.5.2 Uses

An embedded Router Board 112 with U.FL-RSMA pigtail and R52 mini PCI Wi-Fi card widely used by wireless Internet service providers (WISPs) in the Czech Republic.

Wireless networks continue to develop, usage has grown in 2010. Cellular phones are part of everyday wireless networks, allowing easy personal communications. Inter-continental network systems use radio satellites to communicate across the world. Emergency services such as the police utilize wireless networks to communicate effectively. Individuals and businesses use wireless networks to send and share data rapidly, whether it be in a small office building or across the world.

Another use for wireless networks is a cost effective means to connect to the Internet, in regions where the telecommunications infrastructure is both poor and lacking in resources, typically in rural areas and developing countries.

Compatibility issues also arise when dealing with wireless networks. Different devices may have compatibility issues, or might require modifications to solve these issues. Wireless networks are often typically slower than those found in modern versions of Ethernet cable connected installations.

A wireless network is more vulnerable, because anyone can intercept and sometimes divert a network broadcasting signal when point to point connections are used. Many wireless networks

use WEP Wired Equivalent Privacy security systems. These have been found to be still vulnerable to intrusion. Though WEP does block some intruders, the security problems have caused some businesses to continue using wired networks until a more suitable security system can be introduced. The use of suitable firewalls overcome some security problems in wireless networks that are vulnerable to attempted unauthorized access.

6.5.3 Environmental Concerns and Health Hazard

Starting around 2009, there have been increased concerns about the safety of wireless communications, despite little evidence of health risks so far. The president of Lakehead University refused to agree to installation of a wireless network citing a California Public Utilities Commission study which said that the possible risk of tumors and other diseases due to exposure to electromagnetic fields (EMFs) needs to be further investigated.



Case Study

Abuzz Technologies

Abuzz Technologies designs and manufactures interactive touch-screen kiosks. Founded in 1995 as a design agency, Abuzz produced its first kiosks as an innovative way to present its portfolio of designs at trade shows. The overwhelming response to these prototypes convinced the company to shift its focus to interactive touch-screen products in 1997. A kiosk is a self-service terminal comprising a secure enclosure that houses hardware and software. Kiosks usually incorporate a touch-screen monitor, printer, speakers and keyboard as well as a custom-developed software application. The company is now one of the largest kiosk manufacturers in Australia. Based in Sydney, with approximately 50 staff, Abuzz exports to 13 countries around the world, including Germany, the Netherlands, the United Kingdom and the United States. Abuzz's impressive client list includes AMP, BMW, Coles Myer, Greater Union, Hoyts, Nokia, Toyota, Westfield and Westpac. Abuzz designs and assembles the kiosks and develops specialised applications to ensure each kiosk meets the needs of each customer. For example, shopping centre management company Westfield uses kiosks to enable shoppers to search for and locate the stores they need, while cinema chain Greater Union uses them for self-service ticket sales. Banks such as the Commonwealth Bank of Australia and Westpac use Abuzz kiosks as personnel directories.

Business Challenge

Abuzz was experiencing rapid revenue growth of about 50 per cent each year, but the company's data and telephony systems did not have the scale or flexibility required to meet its evolving business needs. Its outdated PABX phone system was proving expensive in terms of support and maintenance.

However, Abuzz's telephony system wasn't its only problem. The company was also dissatisfied with its data network. Staff working remotely were unable to connect to critical business applications. Network security was also a concern. Abuzz's disconnected systems meant it had to manage multiple vendors – one for its PABX phone system, another for Linux firewall and a third for its data network. This increased the cost of managing and maintaining the company's technology systems.

"We wanted a telephony environment that would enable us to manage call flows ourselves and that would integrate with our customer relationship management system," said Drew.

Contd...

Notes

“We also wanted to allow staff in the office to connect remotely to kiosks around Australia. With our existing data and telephony systems, this wasn’t going to happen.”

Solution

Abuzz investigated solutions from a number of vendors before deciding on a Cisco Unified Communications solution that incorporated voice over internet protocol (VoIP) telephony and a secure data network. Cisco Partner Efficient Data Communications (EDC) acted as the systems integrator for the project. “Abuzz was looking to upgrade its telephone system and at the same time provide a new data and communications infrastructure for the office,” explained Andrew Lowy, Director, Efficient Data Communications. “It wanted a complete solution - telephony, data, a secure infrastructure and remote access. Abuzz’s previous infrastructure was pretty rough and ready and not very well designed. It was clear the company would reap enormous benefits by moving to an integrated network.” EDC implemented Cisco Call Manager Express on a Cisco 2811 Integrated Services Router and a Catalyst 3560 48-port switch. It implemented the Call Manger Express security feature set, which provides a firewall built into the router. Abuzz also connected close to 50 Cisco 7940 IP Phones. The solution enables voice and data to run over the one network, halving Abuzz’s cabling requirements. Abuzz also implemented Cisco 800 Series Routers at all kiosk locations, allowing the company’s technical staff to remotely managed and update kiosks over a secure virtual private network (VPN). The Cisco VPN also provides secure network access for staff working remotely.

In addition, EDC installed a front-door station at Abuzz’s office and integrated this with the phone system. The company’s hosted customer relationship management (CRM) system was also integrated with the new telephony infrastructure. The implementation was completed in January 2006.

Secure, Reliable Network

“Installing the Cisco VPN has ensured our network is very secure,” said Drew. “This has made a big difference for us. Now, when staff are overseas, they can log in as if they were part of the network here in Sydney. This has boosted productivity and our customer service levels don’t drop if staff are out of the office.” A secure network has also made it easier for Abuzz to remotely maintain kiosks at client sites around Australia.

“The secure, low-cost VPN connection between our office and remote sites means we can measure statistics, monitor system health, provide 24x7 support and update content remotely,” said Drew.

“We thought Cisco would be a safe bet as it’s a premium product that offers very high levels of security. That’s what the brochures say, and this has certainly been our experience.”

Questions:

1. What are the main concerns for data transmission over a communication channel?
2. Why security is needed in data transmission?

6.6 Summary

- Digital communication is the physical transfer of data over a point to point or point to multipoint communication channel.
- The public switched telephone network (PSTN) is a world wide telephone system and usually this network system uses the digital technology.

- A modem is a device that modulates an analog carrier signal to encode, digital information, and also demodulates such a carrier signal to decode the transmitted information.
- Wireless network refers to any type of computer network that is not connected by cables of any kind.
- Wireless telecommunication network are generally implemented and administered using a transmission system called radio waves.

6.7 Keywords

Computer networking: A computer network, often simply referred to as a network, is a collection of computers and devices interconnected by communications channels that facilitate communications among users and allows users to share resources. Networks may be classified according to a wide variety of characteristics.

Data transmission: Data transmission, digital transmission, or digital communications is the physical transfer of data (a digital bit stream) over a point-to-point or point-to-multipoint communication channel.

Dial-Up lines: Dial-up networking is an important connection method for remote and mobile users. A dial-up line is a connection or circuit between two sites through a switched telephone network.

DNS: The Domain Name System (DNS) is a hierarchical naming system built on a distributed database for computers, services, or any resource connected to the Internet or a private network.

DSL: Digital Subscriber Line (DSL) is a family of technologies that provides digital data transmission over the wires of a local telephone network.

GSM: Global System for Mobile Communications, or GSM (originally from Groupe Spécial Mobile), is the world's most popular standard for mobile telephone systems.

ISDN Lines: Integrated Services Digital Network (ISDN) is a set of communications standards for simultaneous digital transmission of voice, video, data, and other network services over the traditional circuits of the public switched telephone network.

LAN: A local area network (LAN) is a computer network that connects computers and devices in a limited geographical area such as home, school, computer laboratory or office building.

MAN: A metropolitan area network (MAN) is a computer network that usually spans a city or a large campus.

Modem: A modem (modulator-demodulator) is a device that modulates an analog carrier signal to encode digital information, and also demodulates such a carrier signal to decode the transmitted information.

Network topology: Network topology is the layout pattern of interconnections of the various elements (links, nodes, etc.) of a computer network.

PSTN: The public switched telephone network (PSTN) is the network of the world's public circuit-switched telephone networks. It consists of telephone lines, fiber optic cables, microwave transmission links, cellular networks, communications satellites, and undersea telephone cables all inter-connected by switching centers which allows any telephone in the world to communicate with any other.

VPN: A virtual private network (VPN) is a computer network that uses a public telecommunication infrastructure such as the Internet to provide remote offices or individual users secure access to their organization's network.

Notes

WAN: A wide area network (WAN) is a computer network that covers a broad area (i.e., any network whose communications links cross metropolitan, regional, or national boundaries).

WISP: Wireless Internet Service Providers (WISPs) are Internet service providers with networks built around wireless networking.



1. Draw a schematic diagram of ADV7123 digital data communication.
2. Draw an Ethernet DC Power and Digital Data Connection Circuit schematic.

6.8 Self-Assessment Questions

1. What is Data Communication?
 - (a) The transmission of data from one location to another for direct use or further processing.
 - (b) A system made up of hardware, software and communication facilities, computer terminals and input\output devices linked locally or globally.
 - (c) A telegraphy system.
 - (d) A channel used to transmit data at a rate of 1000-8000cps.
2. The three classes of channel that makes up bandwidth are:
 - (a) Data, communication, transmission.
 - (b) Analogue signals, digital signals, standard telephone line.
 - (c) Narrow-band, voice-band, broad-band channels.
 - (d) Automatic dialing, signal transmission, modems.
3. A data communication system is made up of
 - (a) The transmission of data from one location to another for direct use or further processing.
 - (b) A system made up of hardware, software and communication facilities, computer terminals and input\output devices linked locally or globally.
 - (c) A telegraphy system.
 - (d) A channel used to transmit data at a rate of 1000-8000cps.
4. Coaxial cables, fiber optics cables, telegraph system and telephone line are grouped into which bandwidth classes of channel?
 - (a) Narrow-band, broad-band, voice-band.
 - (b) Digital signals, analogue signals, transmission channel.
 - (c) Bandwidth, hardware, software.
 - (d) Broad-band, narrow-band, voice-band.
5. Which bandwidth classes of channel can transmit data at a rate of 1000 -8000 cps?
 - (a) Broad-band.
 - (b) Narrow-band.
 - (c) Voice-band.
 - (d) Internal modem.

Notes

6. What does a data communication system need to carry data from one location to another?
 - (a) Bandwidth.
 - (b) Data transmission channels.
 - (c) Standard telephone line.
 - (d) Telegraph system.
7. A bandwidth determines the volume of data that can be transmitted in a given time?
 - (a) True
 - (b) False
8. Broad-band channel transmit data at speeds over 100 000cps. Which cable provides very high frequency signals that can be transmitted over space?
 - (a) Coaxial cables.
 - (b) Fiber Optics Cables.
 - (c) Communication Satellites.
 - (d) Microwave Signals.
9. Which transmission lines permits data to flow in one direction only i.e. you can send or receive data but not both?
 - (a) Full-duplex line.
 - (b) Half-duplex line.
 - (c) Simplex line.
 - (d) Coaxial line.
10. Which transmission line permits data to be sent alternatively, i.e. you can send and receive but not at the same time?
 - (a) Simplex line.
 - (b) Coaxial line.
 - (c) Half-duplex line.
 - (d) Full-duplex line.

6.9 Review Questions

1. What do you mean by data communication?
2. Explain the general model of data communication. What is the role of modem in it?
3. Explain the general model of digital transmission of data. Why is analog data sampled?
4. What do you mean by digital modulation?
5. Explain various digital modulation techniques.
6. What are computer networks?
7. What do you mean by network topology?
8. How data communication is done using standard telephone lines?
9. What is ATM switch? Under what condition it is used?
10. What do you understand by ISDN?
11. What are the different network methods? Give brief introduction about each.

Notes

12. What do you understand by wireless networks?
13. Give the types of wireless networks.
14. What is the use of wireless network?
15. What is the difference between broadcast and point to point network?

Answers for Self-Assessment Questions

- | | | | | | |
|--------|--------|--------|---------|--------|--------|
| 1. (a) | 2. (c) | 3. (b) | 4. (b) | 5. (c) | 6. (b) |
| 7. (a) | 8. (c) | 9. (c) | 10. (c) | | |

6.10 Further Reading



Books

Computing Fundamentals by Peter Norton.



Online link

<http://www.techbooksforfree.com>

Unit 7: Graphics and Multimedia

Notes

CONTENTS

Objectives

Introduction

7.1 Information Graphics

7.1.1 Visual Devices

7.1.2 Elements of Information Graphics

7.1.3 Interpreting Information Graphics

7.1.4 Interpreting with a Common Visual Language

7.2 Multimedia

7.2.1 Major Characteristics of Multimedia

7.2.2 Word Usage and Context

7.2.3 Application

7.3 Understanding Graphics File Formats

7.3.1 Raster Formats

7.3.2 Vector formats

7.3.3 Bitmap Formats

7.3.4 Metafile Formats

7.3.5 Scene Formats

7.3.6 Animation Formats

7.3.7 Multimedia Formats

7.3.8 Hybrid Formats

7.3.9 Hypertext and Hypermedia Formats

7.3.10 3D Formats

7.3.11 Virtual Reality Modeling Language (VRML) Formats

7.3.12 Audio Formats

7.3.13 Font Formats

7.3.14 Page Description Language (PDL) Formats

7.4 Graphics Software

7.5 Multimedia Basics

7.5.1 Text

7.5.2 Video and Sound

7.5.3 What is Sound?

7.6 Summary

7.7 Keywords

7.8 Self Assessment Questions

7.9 Review Questions

7.10 Further Reading

Notes

Objectives

After studying this unit, you will be able to:

- Explain information graphics.
- Discuss multimedia.
- Understand graphics file format.
- Explain getting images into computer.
- Discuss graphic software.

Introduction

Graphics (from Greek γραφικός graphikos) are visual presentations on some surface, such as a wall, canvas, computer screen, paper, or stone to brand, inform, illustrate, or entertain. Examples are photographs, drawings, Line Art, graphs, diagrams, typography, numbers, symbols, geometric designs, maps, engineering drawings, or other images. Graphics often combine text, illustration, and color. Graphic design may consist of the deliberate selection, creation, or arrangement of typography alone, as in a brochure, flier, poster, web site, or book without any other element. Clarity or effective communication may be the objective, association with other cultural elements may be sought, or merely, the creation of a distinctive style.

Graphics can be functional or artistic. The latter can be a recorded version, such as a photograph, or an interpretation by a scientist to highlight essential features, or an artist, in which case the distinction with imaginary graphics may become blurred.

Multimedia may be viewed by person on stage, projected, transmitted, or played locally with a media player. A broadcast may be a live or recorded multimedia presentation. Broadcasts and recordings can be either analog or digital electronic media technology. Digital online multimedia may be downloaded or streamed. Streaming multimedia may be live or on-demand.

7.1 Information Graphics

Information graphics or infographics are graphic visual representations of information, data or knowledge. These graphics present complex information quickly and clearly, such as in signs, maps, journalism, technical writing, and education. With an information graphic, computer scientists, mathematicians, and statisticians develop and communicate concepts using a single symbol to process information.

Following are information graphics subjects.

7.1.1 Visual Devices

Information graphics are visual devices intended to communicate complex information quickly and clearly. The devices include, according to Doug Newsom (2004), charts, diagrams, graphs, tables, maps and lists. Among the most common devices are horizontal bar charts, vertical column charts, and round or oval pie charts, that can summarize a lot of statistical information. Diagrams can be used to show how a system works, and may be an organizational chart that

shows lines of authority, or a systems flowchart that shows sequential movement. Illustrated graphics use images to related data. The snapshots features used every day by USA Today are good examples of this technique. Tables are commonly used and may contain lots of numbers. Modern interactive maps and bulleted numbers are also infographic devices.

7.1.2 Elements of Information Graphics

The basic material of an information graphic is the data, information, or knowledge that the graphic presents. In the case of data, the creator may make use of automated tools such as graphing software to represent the data in the form of lines, boxes, arrows, and various symbols and pictograms. The information graphic might also feature a key which defines the visual elements in plain English. A scale and labels are also common. The elements of an info graphic do not have to be an exact or realistic representation of the data, but can be a simplified version.

7.1.3 Interpreting Information Graphics

Many information graphics are specialised forms of depiction that represent their content in sophisticated and often abstract ways. In order to interpret the meaning of these graphics appropriately, the viewer requires a suitable level of graphicacy. In many cases, the required graphicacy involves comprehension skills that are learned rather than innate. At a fundamental level, the skills of decoding individual graphic signs and symbols must be acquired before sense can be made of an information graphic as a whole. However, knowledge of the conventions for distributing and arranging these individual components is also necessary for the building of understanding.

7.1.4 Interpreting with a Common Visual Language

In contrast to the above, many other forms of infographics take advantage of innate visual language that is largely universal. The disciplined use of the color red, for emphasis, on an otherwise muted design, demands attention in a primal way even children understand. Many maps, interfaces, dials and gauges on instruments and machinery use icons that are easy to grasp and speed understanding for safe operation.

7.2 Multimedia

Multimedia is media and content that uses a combination of different content forms. The term can be used as a noun (a medium with multiple content forms) or as an adjective describing a medium as having multiple content forms. The term is used in contrast to media which only use traditional forms of printed or hand-produced material. Multimedia includes a combination of text, audio, still images, animation, video, and interactivity content forms.

Multimedia is usually recorded and played, displayed or accessed by information content processing devices, such as computerized and electronic devices, but can also be part of a live performance. Multimedia (as an adjective) also describes electronic media devices used to store and experience multimedia content. Multimedia is distinguished from mixed media in fine art; by including audio, for example, it has a broader scope. The term "rich media" is synonymous for interactive multimedia. Hypermedia can be considered one particular multimedia application.

7.2.1 Major Characteristics of Multimedia

Multimedia presentations may be viewed by person on stage, projected, transmitted, or played locally with a media player. A broadcast may be a live or recorded multimedia presentation. Broadcasts and recordings can be either analog or digital electronic media technology. Digital online multimedia may be downloaded or streamed. Streaming multimedia may be live or on-demand.

Notes

Multimedia games and simulations may be used in a physical environment with special effects, with multiple users in an online network, or locally with an offline computer, game system, or simulator.

The various formats of technological or digital multimedia may be intended to enhance the users' experience, for example to make it easier and faster to convey information or in entertainment or art, to transcend everyday experience.

Figure 7.1: A Lasershow is a Live Multimedia Performance



Enhanced levels of interactivity are made possible by combining multiple forms of media content. Online multimedia is increasingly becoming object-oriented and data-driven, enabling applications with collaborative end-user innovation and personalization on multiple forms of content over time. Examples of these range from multiple forms of content on Web sites like photo galleries with both images (pictures) and title (text) user-updated, to simulations whose co-efficients, events, illustrations, animations or videos are modifiable, allowing the multimedia “experience” to be altered without reprogramming. In addition to seeing and hearing, Haptic technology enables virtual objects to be felt. Emerging technology involving illusions of taste and smell may also enhance the multimedia experience.

7.2.2 Word Usage and Context

Since media is the plural of medium, the term “multimedia” is a pleonasm if “multi” is used to describe multiple occurrences of only one form of media such as a collection of audio CDs. This is why it’s important that the word “multimedia” is used exclusively to describe multiple forms of media and content.

The term “multimedia” is also ambiguous. Static content (such as a paper book) may be considered multimedia if it contains both pictures and text or may be considered interactive if the user interacts by turning pages at will. Books may also be considered non-linear if the pages are accessed non-sequentially. The term “video”, if not used exclusively to describe motion photography, is ambiguous in multimedia terminology. Video is often used to describe the file format, delivery format, or presentation format instead of “footage” which is used to distinguish motion photography from “animation” of rendered motion imagery. Multiple forms of information content are often not considered modern forms of presentation such as audio or video. Likewise, single forms of information content with single methods of information processing (e.g. non-interactive audio) are often called multimedia, perhaps to distinguish static media from active media. In the Fine arts, for example, Leda Luss Luyken’s ModulArt brings two key elements of musical composition and film into the world of painting: variation of a theme and movement of and within a picture, making ModulArt an interactive multimedia form of art. Performing arts may also be considered multimedia considering that performers and props are multiple forms of both content and media.

Figure 7.2: A presentation using Powerpoint. Corporate presentations may combine all forms of media content



Figure 7.3: Virtual reality uses multimedia content. Applications and delivery platforms of multimedia are virtually limitless



Figure 7.4: VVO Multimedia-Terminal in Dresden WTC (Germany)



Notes

7.2.3 Application

Multimedia finds its application in various areas including, but not limited to, advertisements, art, education, entertainment, engineering, medicine, mathematics, business, scientific research and spatial temporal applications. Several examples are as follows:

Creative Industries

Creative industries use multimedia for a variety of purposes ranging from fine arts, to entertainment, to commercial art, to journalism, to media and software services provided for any of the industries listed below. An individual multimedia designer may cover the spectrum throughout their career. Request for their skills range from technical, to analytical, to creative.

Commercial

Much of the electronic old and new media used by commercial artists is multimedia. Exciting presentations are used to grab and keep attention in advertising. Business to business, and interoffice communications are often developed by creative services firms for advanced multimedia presentations beyond simple slide shows to sell ideas or liven-up training. Commercial multimedia developers may be hired to design for governmental services and nonprofit services applications as well.

Entertainment and Fine Arts

In addition, multimedia is heavily used in the entertainment industry, especially to develop special effects in movies and animations. Multimedia games are a popular pastime and are software programs available either as CD-ROMs or online. Some video games also use multimedia features. Multimedia applications that allow users to actively participate instead of just sitting by as passive recipients of information are called Interactive Multimedia. In the Arts there are multimedia artists, whose minds are able to blend techniques using different media that in some way incorporates interaction with the viewer. Although multimedia display material may be volatile, the survivability of the content is as strong as any traditional media. Digital recording material may be just as durable and infinitely reproducible with perfect copies every time.

Education

In Education, multimedia is used to produce computer-based training courses (popularly called CBTs) and reference books like encyclopedia and almanacs. A CBT lets the user go through a series of presentations, text about a particular topic, and associated illustrations in various information formats. Edutainment is an informal term used to describe combining education with entertainment, especially multimedia entertainment.

Learning theory in the past decade has expanded dramatically because of the introduction of multimedia. Several lines of research have evolved (e.g. Cognitive load, Multimedia learning, and the list goes on). The possibilities for learning and instruction are nearly endless.

The idea of media convergence is also becoming a major factor in education, particularly higher education. Defined as separate technologies such as voice (and telephony features), data (and productivity applications) and video that now share resources and interact with each other, synergistically creating new efficiencies, media convergence is rapidly changing the curriculum in universities all over the world. Likewise, it is changing the availability, or lack thereof, of jobs requiring this savvy technological skill.

Journalism

Newspaper companies all over are also trying to embrace the new phenomenon by implementing its practices in their work. While some have been slow to come around, other major newspapers like The New York Times, USA Today and The Washington Post are setting the precedent for the positioning of the newspaper industry in a globalized world.

News reporting is not limited to traditional media outlets. Freelance journalists can make use of different new media to produce multimedia pieces for their news stories. It engages global audiences and tells stories with technology, which develops new communication techniques for both media producers and consumers. Common Language Project is an example of this type of multimedia journalism production.

Engineering

Software engineers may use multimedia in Computer Simulations for anything from entertainment to training such as military or industrial training. Multimedia for software interfaces are often done as a collaboration between creative professionals and software engineers.

Industry

In the Industrial sector, multimedia is used as a way to help present information to shareholders, superiors and coworkers. Multimedia is also helpful for providing employee training, advertising and selling products all over the world via virtually unlimited web-based technology.

Mathematical and Scientific Research

In mathematical and scientific research, multimedia is mainly used for modeling and simulation. For example, a scientist can look at a molecular model of a particular substance and manipulate it to arrive at a new substance. Representative research can be found in journals such as the Journal of Multimedia.

Medicine

In Medicine, doctors can get trained by looking at a virtual surgery or they can simulate how the human body is affected by diseases spread by viruses and bacteria and then develop techniques to prevent it.

Document Imaging

Document imaging is a technique that takes hard copy of an image/document and converts it into a digital format (for example, scanners).

Disabilities

Ability Media allows those with disabilities to gain qualifications in the multimedia field so they can pursue careers that give them access to a wide array of powerful communication forms.

Miscellaneous

In Europe, the reference organisation for Multimedia industry is the European Multimedia Associations Convention (EMMAC).

Notes

Multimedia represents the convergence of text, pictures, video and sound into a single form. The power of multimedia and the Internet lies in the way in which information is linked.

Multimedia and the Internet require a completely new approach to writing. The style of writing that is appropriate for the 'on-line world' is highly optimized and designed to be able to be quickly scanned by readers.

A good site must be made with a specific purpose in mind and a site with good interactivity and new technology can also be useful for attracting visitors. The site must be attractive and innovative in its design, function in terms of its purpose, easy to navigate, frequently updated and fast to download.

When users view a page, they can only view one page at a time. As a result, multimedia users must create a 'mental model of information structure.

7.3 Understanding Graphics File Formats

7.3.1 Raster Formats

7.3.1.1 JPEG/JFIF

JPEG (Joint Photographic Experts Group) is a compression method; JPEG-compressed images are usually stored in the **JFIF** (JPEG File Interchange Format) file format. JPEG compression is (in most cases) lossy compression. The JPEG/JFIF filename extension is **JPG** or **JPEG**. Nearly every digital camera can save images in the JPEG/JFIF format, which supports 8 bits per color (red, green, blue) for a 24-bit total, producing relatively small files. When not too great, the compression does not noticeably detract from the image's quality, but JPEG files suffer generational degradation when repeatedly edited and saved. The JPEG/JFIF format also is used as the image compression algorithm in many PDF files.

7.3.1.2 JPEG 2000

JPEG 2000 is a compression standard enabling both lossless and lossy storage. The compression methods used are different from the ones in standard JFIF/JPEG; they improve quality and compression ratios, but also require more computational power to process. JPEG 2000 also adds features that are missing in JPEG. It is not nearly as common as JPEG, but it is used currently in professional movie editing and distribution (e.g., some digital cinemas use JPEG 2000 for individual movie frames).

7.3.1.3 Exif

The **Exif** (Exchangeable image file format) format is a file standard similar to the JFIF format with TIFF extensions; it is incorporated in the JPEG-writing software used in most cameras. Its purpose is to record and to standardize the exchange of images with image metadata between digital cameras and editing and viewing software. The metadata are recorded for individual images and include such things as camera settings, time and date, shutter speed, exposure, image size, compression, name of camera, color information, etc. When images are viewed or edited by image editing software, all of this image information can be displayed.

7.3.1.4 TIFF

The **TIFF** (Tagged Image File Format) format is a flexible format that normally saves 8 bits or 16 bits per color (red, green, blue) for 24-bit and 48-bit totals, respectively, usually using either the **TIFF** or **TIF** filename extension. TIFF's flexibility can be both an advantage and disadvantage, since a reader that reads every type of TIFF file does not exist. TIFFs can be lossy and lossless; some offer relatively good lossless compression for bi-level (black&white) images. Some digital

cameras can save in TIFF format, using the LZW compression algorithm for lossless storage. TIFF image format is not widely supported by web browsers. TIFF remains widely accepted as a photograph file standard in the printing business. TIFF can handle device-specific color spaces, such as the CMYK defined by a particular set of printing press inks. OCR (Optical Character Recognition) software packages commonly generate some (often monochromatic) form of TIFF image for scanned text pages.

7.3.1.5 RAW

RAW refers to a family of raw image formats that are options available on some digital cameras. These formats usually use a lossless or nearly-lossless compression, and produce file sizes much smaller than the TIFF formats of full-size processed images from the same cameras. Although there is a standard raw image format, (ISO 12234-2, TIFF/EP), the raw formats used by most cameras are not standardized or documented, and differ among camera manufacturers. Many graphic programs and image editors may not accept some or all of them, and some older ones have been effectively orphaned already. Adobe's Digital Negative (DNG) specification is an attempt at standardizing a raw image format to be used by cameras, or for archival storage of image data converted from undocumented raw image formats, and is used by several niche and minority camera manufacturers including Pentax, Leica, and Samsung. The raw image formats of more than 230 camera models, including those from manufacturers with the largest market shares such as Canon, Nikon, Sony, and Olympus, can be converted to DNG. DNG was based on ISO 12234-2, TIFF/EP, and ISO's revision of TIFF/EP is reported to be adding Adobe's modifications and developments made for DNG into profile 2 of the new version of the standard.

As far as videocameras are concerned, ARRI's Arriflex D-20 and D-21 cameras provide raw 3K-resolution sensor data with Bayern pattern as still images (one per frame) in a proprietary format (.ari file extension). Red Digital Cinema Camera Company, with its Mysterium sensor family of still and video cameras, uses its proprietary raw format called REDCODE (.R3D extension), which stores still as well as audio+video information in one lossy-compressed file.

7.3.1.6 PNG

The **PNG** (Portable Network Graphics) file format was created as the free, open-source successor to the GIF. The PNG file format supports truecolor (16 million colors) while the GIF supports only 256 colors. The PNG file excels when the image has large, uniformly colored areas. The lossless PNG format is best suited for editing pictures, and the lossy formats, like JPG, are best for the final distribution of photographic images, because in this case JPG files are usually smaller than PNG files. The Adam7-interlacing allows an early preview, even when only a small percentage of the image data has been transmitted.

PNG provides a patent-free replacement for GIF and can also replace many common uses of TIFF. Indexed-color, grayscale, and truecolor images are supported, plus an optional alpha channel.

PNG is designed to work well in online viewing applications like web browsers so it is fully streamable with a progressive display option. PNG is robust, providing both full file integrity checking and simple detection of common transmission errors. Also, PNG can store gamma and chromaticity data for improved color matching on heterogeneous platforms.

Some programs do not handle PNG gamma correctly, which can cause the images to be saved or displayed darker than they should be.

Animated formats derived from PNG are MNG and APNG. The latter is supported by Mozilla Firefox and Opera and is backwards compatible with PNG.

Notes

7.3.1.7 GIF

GIF (Graphics Interchange Format) is limited to an 8-bit palette, or 256 colors. This makes the GIF format suitable for storing graphics with relatively few colors such as simple diagrams, shapes, logos and cartoon style images. The GIF format supports animation and is still widely used to provide image animation effects. It also uses a lossless compression that is more effective when large areas have a single color, and ineffective for detailed images or dithered images.

7.3.1.8 BMP

The **BMP file format** (Windows bitmap) handles graphics files within the Microsoft Windows OS. Typically, BMP files are uncompressed, hence they are large; the advantage is their simplicity and wide acceptance in Windows programs.

7.3.1.9 WEBP

WebP is a new image format that uses lossy compression. It was designed by Google to reduce image file size to speed up web page loading; its principal purpose is to supersede JPEG as the primary format for photographs on the web.

WebP is based on VP8's intra-frame coding and uses a container based on RIFF.

7.3.1.10 Others

Other image file formats of raster type include:

- JPEG XR (New JPEG standard based on Microsoft HD Photo)
- TGA (TARGA)
- ILBM (InterLeaved BitMap)
- PCX (Personal Computer eXchange)
- ECW (Enhanced Compression Wavelet)
- IMG (ERDAS IMAGINE Image)
- SID (multiresolution seamless image database, MrSID)
- CD5 (Chasys Draw Image)
- FITS (Flexible Image Transport System)
- PGF (Progressive Graphics File)
- XCF (eXperimental Computing Facility format, native GIMP format)
- PSD (Adobe PhotoShop Document)
- PSP (Corel Paint Shop Pro)

7.3.2 Vector Formats

As opposed to the raster image formats above (where the data describes the characteristics of each individual pixel), vector image formats contain a geometric description which can be rendered smoothly at any desired display size.

Vector file formats can contain bitmap data as well. 3D graphic file formats are technically vector formats with pixel data texture mapping on the surface of a vector virtual object, warped to match the angle of the viewing perspective.

At some point, all vector graphics must be rasterized in order to be displayed on digital monitors. However, vector images can be displayed with analog CRT technology such as that used in some electronic test equipment, medical monitors, radar displays, laser shows and early video games. Plotters are printers that use vector data rather than pixel data to draw graphics.

7.3.2.1 Computer Graphics Metafile (CGM)

Computer Graphics Metafile (CGM) is a file format for 2D vector graphics, raster graphics, and text, and is defined by **ISO/IEC 8632**. All graphical elements can be specified in a textual source file that can be compiled into a binary file or one of two text representations. CGM provides a means of graphics data interchange for computer representation of 2D graphical information independent from any particular application, system, platform, or device. It has been adopted to some extent in the areas of technical illustration and professional design, but has largely been superseded by formats such as SVG and DXF.

7.3.2.2 Scalable Vector Graphics (SVG)

Scalable Vector Graphics (SVG) is an open standard created and developed by the World Wide Web Consortium to address the need (and attempts of several corporations) for a versatile, scriptable and all-purpose vector format for the web and otherwise. The SVG format does not have a compression scheme of its own, but due to the textual nature of XML, an SVG graphic can be compressed using a program such as gzip. Because of its scripting potential, SVG is a key component in web applications: interactive web pages that look and act like applications.

7.3.2.3 Others

Other image file formats of vector type include:

- AI (Adobe Illustrator)
- CDR (CorelDRAW)
- EPS (Encapsulated PostScript)
- HVIF (Haiku Vector Icon Format)
- ODG (OpenDocument Graphics)
- PDF (Portable Document Format)
- PGML (Precision Graphics Markup Language)
- SWF (Shockwave Flash)
- VML (Vector Markup Language)
- WMF / EMF (Windows Metafile / Enhanced Metafile)
- XPS (XML Paper Specification)

7.3.3 Bitmap Formats

Bitmap formats are used to store bitmap data. Files of this type are particularly well-suited for the storage of real-world images such as photographs and video images. Bitmap files, sometimes called raster files, essentially contain an exact pixel-by-pixel map of an image. A rendering application can subsequently reconstruct this image on the display surface of an output device.

Microsoft BMP, PCX, TIFF, and TGA are examples of commonly used bitmap formats.

Notes

7.3.4 Metafile Formats

Metafiles can contain both bitmap and vector data in a single file. The simplest metafiles resemble vector format files; they provide a language or grammar that may be used to define vector data elements, but they may also store a bitmap representation of an image. Metafiles are frequently used to transport bitmap or vector data between hardware platforms, or to move image data between software platforms.

WPG, Macintosh PICT, and CGM are examples of commonly used metafile formats.

7.3.5 Scene Formats

Scene format files (sometimes called scene description files) are designed to store a condensed representation of an image or scene, which is used by a program to reconstruct the actual image. What's the difference between a vector format file and a scene format file? Just that vector files contain descriptions of portions of the image, and scene files contain instructions that the rendering program uses to construct the image. In practice it's sometimes hard to decide whether a particular format is scene or vector; it's more a matter of degree than anything absolute.

7.3.6 Animation Formats

Animation formats have been around for some time. The basic idea is that of the flip-books you played with as a kid; with those books, you rapidly displayed one image superimposed over another to make it appear as if the objects in the image are moving. Very primitive animation formats store entire images that are displayed in sequence, usually in a loop. Slightly more advanced formats store only a single image but multiple color maps for the image. By loading in a new color map, the colors in the image change, and the objects appear to move. Advanced animation formats store only the differences between two adjacent images (called frames) and update only the pixels that have actually changed as each frame is displayed. A display rate of 10-15 frames per second is typical for cartoon-like animations. Video animations usually require a display rate of 20 frames per second or better to produce a smoother motion.

TDDD and TTDDD are examples of animation formats.

7.3.7 Multimedia Formats

Multimedia formats are relatively new but are becoming more and more important. They are designed to allow the storage of data of different types in the same file. Multimedia formats usually allow the inclusion of graphics, audio, and video information. Microsoft's RIFF, Apple's QuickTime, MPEG, and Autodesk's FLI are well-known examples, and others are likely to emerge in the near future.

7.3.8 Hybrid Formats

Currently, there is a good deal of research being conducted on the integration of unstructured text and bitmap data ("hybrid text") and the integration of record-based information and bitmap data ("hybrid database"). As this work bears fruit, we expect that hybrid formats capable of efficiently storing graphics data will emerge and will steadily become more important.

7.3.9 Hypertext and Hypermedia Formats

Hypertext is a strategy for allowing nonlinear access to information. In contrast, most books are linear, having a beginning, an end, and a definite pattern of progression through the text. Hypertext, however, enables documents to be constructed with one or more beginnings, with one, none, or multiple ends, and with many hypertext links that allow users to jump to any available place in the document they wish to go.

Hypertext languages are not graphics file formats, like the GIF or DXF formats. Instead, they are programming languages, like PostScript or C. As such, they are specifically designed for serial data stream transmission. That is, you can start decoding a stream of hypertext information as you receive the data. You need not wait for the entire hypertext document to be downloaded before viewing it.

The term hypermedia refers to the marriage of hypertext and multimedia. Modern hypertext languages and network protocols support a wide variety of media, including text and fonts, still and animated graphics, audio, video, and 3D data. Hypertext allows the creation of a structure that enables multimedia data to be organized, displayed, and interactively navigated through by a computer user.

Hypertext and hypermedia systems, such as the World Wide Web, contain millions of information resources stored in the form of GIF, JPEG, PostScript, MPEG, and AVI files. Many other formats are used as well.

7.3.10 3D Formats

Three-dimensional data files store descriptions of the shape and color of 3D models of imaginary and real-world objects. 3D models are typically constructed of polygons and smooth surfaces, combined with descriptions of related elements, such as color, texture, reflections, and so on, that a rendering application can use to reconstruct the object. Models are placed in scenes with lights and cameras, so objects in 3D files are often called scene elements.

Rendering applications that can use 3D data are generally modeling and animation programs, such as NewTek's Lightwave and Autodesk's 3D Studio. They provide the ability to adjust the appearance of the rendered image through changes and additions to the lighting, textures applied to scene elements, and the relative positions of scene elements. In addition, they allow the user to animate, or assign motions to, scene elements. The application then creates a series of bitmap files, or frames, that taken in sequence can be assembled into a movie.

It's important to understand that vector data historically has been 2D in nature. That is, the creator application with which the data originated made no attempt to simulate 3D display through the application of perspective. Examples of vector data include CAD drawings and most clip art designed to be used in desktop publishing applications. There is a certain amount of confusion in the market about what constitutes 3D rendering. This is complicated by the fact that 3D data is now supported by a number of formats that previously stored only 2D vector data. An example of this is Autodesk's DXF format. Formats like DXF are sometimes referred to as extended vector formats.

7.3.11 Virtual Reality Modeling Language (VRML) Formats

VRML (pronounced "vermel") may be thought of as a hybrid of 3D graphics and HTML. VRML v1.0 is essentially a subset of the Silicon Graphics Inventor file format and adds to it support for linking to Uniform Resource Locators URLs in the World Wide Web.

VRML encodes 3D data in a format suitable for exchange across the Internet using the Hypertext Transfer Protocol (HTTP). VRML data received from a Web server is displayed on a Web browser that supports VRML language interpretation. We expect that VRML-based 3D graphics will soon be very common on the World Wide Web.

This book does not contain an in-depth discussion of VRML for some of the same reasons that we do not provide detailed descriptions of hypertext, hypermedia, and 3D formats.

7.3.12 Audio Formats

Audio is typically stored on magnetic tape as analog data. For audio data to be stored on media such as a CD-ROM or hard disk, it must first be encoded using a digital sampling process similar

Notes

to that used to store digital video data. Once encoded, the audio data can then be written to disk as a raw digital audio data stream, or, more commonly, stored using an audio file format.

Audio file formats are identical in concept to graphics file formats, except that the data they store is rendered for your ears and not for your eyes. Most formats contain a simple header that describes the audio data they contain. Information commonly stored in audio file format headers includes samples per second, number of channels, and number of bits per sample. This information roughly corresponds to the number of samples per pixel, number of color planes, and number of bits per sample information commonly found in graphics file headers.

Where audio file formats differ greatly is in the methods of data compression they use. Huffman encoding is commonly used for both 8-bit graphical and audio data. 16-bit audio data, however, requires algorithms specially adapted to the problems of compressing audio data. Such compression schemes include the CCITT (International Telegraph and Telephone Consultative Committee) recommendations G.711 (uLAW), G.721 (ADPCM 32) and G.723 (ADPCM 24), and the U.S. federal standards FIPS-1016 (CELP) and FIPS-1015 (LPC-10E).

Because audio data is very different from graphics data, this book does not attempt to cover audio file formats.

7.3.13 Font Formats

Another class of formats not covered in this book are font files. Font files contain the descriptions of sets of alphanumeric characters and symbols in a compact, easy-to-access format. They are generally designed to facilitate random access of the data associated with individual characters. In this sense, they are databases of character or symbol information, and for this reason font files are sometimes used to store graphics data that is not alphanumeric or symbolic in nature. Font files may or may not have a global header, and some files support sub-headers for each character. In any case, it is necessary to know the start of the actual character data, the size of each character's data, and the order in which the characters are stored in order to retrieve individual characters without having to read and analyze the entire file. Character data in the file may be indexed alphanumerically, by ASCII code, or by some other scheme. Some font files support arbitrary additions and editing, and thus have an index somewhere in the file to help you find the character data.

Some font files support compression, and many support encryption of the character data. The creation of character sets by hand has always been a difficult and time-consuming process, and typically a font designer spent a year or more on a single character set. Consequently, companies that market fonts (called foundries for reasons dating back to the origins of printing using mechanical type) often seek to protect their investments through legal means or through encryption. In the United States, for instance, the names of fonts are considered proprietary, but the outlines described by the character data are not. It is not uncommon to see pirated data embedded in font files under names different from the original.

Historically there have been three main types of font files: bitmap, stroke, and spline-based outlines, described in the following sections.

We choose not to cover font files in this book because font technology is a world to itself, with different terminology and concerns. Many of the font file formats are still proprietary and encrypted and, in fact, are not available to the general public. Although there are a few older spline-based font formats still in use, font data in the TrueType and Adobe Type 1 formats is readily available on all the major platforms and is well-documented elsewhere in publications readily available to developers.

Bitmap fonts

Bitmap fonts consist of a series of character images rendered to small rectangular bitmaps and stored sequentially in a single file. The file may or may not have a header. Most bitmap font files are monochrome, and most store fonts in uniformly sized rectangles to facilitate speed of access. Characters stored in bitmap format may be quite elaborate, but the size of the file increases, and, consequently, speed and ease of use decline with increasingly complex images.

The advantages of bitmap files are speed of access and ease of use--reading and displaying a character from a bitmap file usually involve little more than reading the rectangle containing the data into memory and displaying it on the display surface of the output device. Sometimes, however, the data is analyzed and used as a template for display of the character by the rendering application. The chief disadvantages of bitmap fonts are that they are not easily scaled, and that rotated bitmap fonts look good only on screens with square pixels.

Most character-based systems, such as MS-DOS, character-mode UNIX, and character terminal-based systems use bitmap fonts stored in ROM or on disk. However, bitmap fonts are seldom used today when sufficient processing power is available to enable the use of other types of font data.

Stroke fonts

Stroke fonts are databases of characters stored in vector form. Characters can consist of single strokes or may be hollow outlines. Stroke character data usually consists of a list of line endpoints meant to be drawn sequentially, reflecting the origin of many stroke fonts in applications supporting pen plotters. Some stroke fonts may be more elaborate, however, and may include instructions for arcs and other curves. Perhaps the best-known and most widely used stroke fonts were the Hershey character sets, which are still available online.

The advantages of stroke fonts are that they can be scaled and rotated easily, and that they are composed of primitives, such as lines and arcs, which are well-supported by most GUI operating environments and rendering applications. The main disadvantage of stroke fonts is that they generally have a mechanical look at variance with what we've come to expect from reading high-quality printed text all our lives.

Stroke fonts are seldom used today. Most pen plotters support them, however. You also may need to know more about them if you have a specialized industrial application using a vector display or something similar.

Spline-based outline fonts

Character descriptions in spline-based fonts are composed of control points allowing the reconstruction of geometric primitives known as splines. There are a number of types of splines, but they all enable the drawing of the subtle, eye-pleasing curves we've come to associate with high-quality characters that make up printed text. The actual outline data is usually accompanied by information used in the reconstruction of the characters.

The advantages of spline-based fonts are that they can be used to create high-quality character representations, in some cases indistinguishable from text made with metal type. Most traditional fonts, in fact, have been converted to spline-based outlines. In addition, characters can be scaled, rotated, and otherwise manipulated in ways only dreamed about even a generation ago.

Unfortunately, the reconstruction of characters from spline outline data is no trivial task, and the higher quality afforded by spline outlines comes at a price in rendering time and program development costs.

Notes

7.3.14 Page Description Language (PDL) Formats

Page description languages (PDLs) are actual computer languages used for describing the layout, font information, and graphics of printed and displayed pages. PDLs are used as the interpreted languages used to communicate information to printing devices, such as hardcopy printers, or to display devices, such as graphical user interface (GUI) displays. The greatest difference is that PDL code is very device-dependent. A typical PostScript file contains detailed information on the output device, font metrics, color palettes, and so on. A PostScript file containing code for a 4-color, A4-sized document can only be printed or displayed on a device that can handle these metrics.

Markup languages, on the other hand, contain no information specific to the output device. Instead, they rely on the fact that the device that is rendering the markup language code can adapt to the formatting instructions that are sent to it. The rendering program chooses the fonts, colors, and method of displaying the graphical data. The markup language provides only the information and how it is structured.

Although PDL files can contain graphical information, we do not consider PDLs to be graphics file formats any more than we would consider a module of C code that contains an array of graphical information to be a graphics file format. PDLs are complete programming languages, requiring the use of sophisticated interpreters to read their data; they are quite different from the much simpler parsers used to read graphics file formats.



Did u know?

If your image size is say 3000x2000 pixels, then this is $3000 \times 2000 = 6$ million pixels (6 megapixels). If this 6 megapixel image data is RGB color (if 24 bits, or 3 bytes per pixel of RGB color information), then the size of this image data is 6 million $\times$ 3 bytes RGB = 18 million bytes. That is simply how large your image data is then file compression like JPG or LZW can make the file smaller, but when you open the image in computer memory for use, the JPG may not still have the same image quality, but it is always still 3000x2000 pixels and 18 million bytes. This is simply how large your RGB image data is (megapixels $\times$ 3 bytes per pixel).

7.4 Graphics Software

Graphics software or Image editing software encompasses the processes of altering images, whether they be digital photographs, traditional analog photographs, or illustrations. Traditional analog image editing is known as photo retouching, using tools such as an airbrush to modify photographs, or editing illustrations with any traditional art medium. Graphic software programs, which can be broadly grouped into vector graphics editors, raster graphics editors, and 3d modelers, are the primary tools with which a user may manipulate, enhance, and transform images. Many image editing programs are also used to render or create computer art from scratch.

In computer graphics, **graphics software or image editing software** is a program or collection of programs that enable a person to manipulate visual images on a computer.

Computer graphics can be classified into two distinct categories: raster graphics and vector graphics. Before learning about computer software that manipulates or displays these graphics types, you should be familiar with both.

Many graphics programs focus exclusively on either vector or raster graphics, but there are a few that combine them in interesting and sometimes unexpected ways. It is simple to convert from vector graphics to raster graphics, but going the other way is harder. Some software attempts to do this.

Most graphics programs have the ability to import and export one or more graphics file formats.

The use of a swatch is a palette of active colours that are selected and rearranged by the preference of the user. A swatch may be used in a program or be part of the universal palette on an operating system, it is used to change the colour of a project, that may be text, image or video editing.

Several graphics programs support animation, or digital video. Vector graphics animation can be described as a series of mathematical transformations that are applied in sequence to one or more shapes in a scene. Raster graphics animation works in a similar fashion to film-based animation, where a series of still images produces the illusion of continuous movement.

7.5 Multimedia Basics

This term is often used in reference with creativity using the PC. In simple words Multimedia means multi (many) and media (communication/transfer medium). Hence it means many mediums working together or independently. It can be considered to be a combination of several mediums:

1. Text
2. Graphics
3. Animation
4. Video And Sound

They all are used to present information to the user via the computer.

The main feature of any multimedia application is its human interactivity. This means that it is user friendly and basically caters to the commands the user dictates. For instance, users can be interactive with a particular program by clicking on various icons and links, the program subsequently reveals detailed information on that particular subject. The above theme can be considered to be a GUI (Graphical User Interface).

A Multimedia System can be defined as below:

A communications network, computer platforms or a software tool is a multimedia system if it supports the interaction with atleast one of the types of the above-mentioned mediums.

The components of a multimedia package are:

7.5.1 Text

It is one of the most popularly used mediums of appearance, In 99% of the occasions text provides the core structure to the package.

A major drawback of using text is that it is not user friendly as compared to the other medium. Also it normally has a lack luster performance when judged with its counterparts. It is for instance harder to read from a screen as it tires the eyes more than reading it in its print version. But now with the availability of TEXT to SPEECH software this drawback is speedily disappearing.

We can also provide Hypertext. Hypertext in a way provides a choice to the user in reading or not reading the information in detail, attached to a particular word of text.

7.5.1.1 What is Text?

Text is the graphic representation of speech. Unlike speech, however, text is silent, easily stored, and easily manipulated. Text in multimedia presentations makes it possible to convey large

Notes

amounts of information using very little storage space. Computers customarily represent text using the ASCII (American Standard Code for Information Interchange) system. The ASCII system assigns a number for each of the characters found on a typical typewriter. Each character is represent as a binary number which can be understood by the computer. On the internet ASCII can be transmitted from one computer to another over telephone lines. Non-text files (like graphics) can also be encoded as ASCII files for transmission. Once received, the ASCII file can be translated by decoding software back into its original format.

7.5.1.2 Fonts

The graphic representation of speech can take many forms. These forms are referred to as fonts or typefaces. Fonts can be characterized by their proportionality and their serif characteristics.

Non-proportional fonts, also known as monospaced fonts, assign exactly the same amount of horizontal space to each character. Monospaced fonts are ideal for creating tables of information where columns of characters must be aligned. Text created with non-proportional fonts often look as though they were produced on a typewriter. Two commonly-used non-proportional fonts are Courier and Monaco on the Macintosh and Courier New and FixedSys on Windows.

Proportional fonts vary the spacing between characters according to the width required by each letter. For example, an "l" requires less horizontal space than a "d." Words created with proportional fonts look more like they were typeset by a professional typographer. Two commonly-used proportional fonts are Times and Helvetica on the Macintosh and Times New Roman and Arial on Windows. This article is written using proportional fonts.

Serif fonts are designed with small ticks at the bottom of each character. These ticks aid the reader in following the text. Serif fonts are generally used for text in the body of an article because they are easier to read than Sans Serif fonts. The body text in this article is written using a serif font. Two commonly-used serif fonts are Times and Courier on the Macintosh and Times New Roman and Courier New on Windows. The body text in this article uses a serif font that is proportional.

Sans Serif fonts are designed without small ticks at the bottom of each character. Sans Serif fonts are generally used for headers within an article because they create an attractive contrast with the Serif fonts used in the body text. The section headers in this article are written using a sans serif font. Two commonly-used sans serif fonts are Helvetica and Monaco on the Macintosh and Arial and FixedSys on Windows. The headings in this article use a sans serif font that is proportional.

7.5.1.3 Font Samples

Times New Roman and Georgia are proportional serif fonts.

Verdana and Arial are proportional sans serif fonts.

Courier New and Courier are non-proportional serif fonts.

Monaco and FixedSys are non-proportional sans serif fonts.

7.5.1.4 Font Standards

There are basically two font standards of interest today. The first is called Postscript. Postscript fonts are designed to produce exceptionally good looking type when printed on a high-resolution printer. To use a Postscript font, a set of files must be installed on the host computer. These files include a printer font that is downloaded to the printer when a page containing the font is printed, and a set of screen fonts which represent the font on screen at various point sizes. If the user chooses to view the font at a size not provided for by the font file, the computer interpolates and produces an unattractive font on screen. The printed output, however, will always appear attractive.

Postscript is a complete page description language that encompasses all elements of a printed page including high-resolution graphics. Postscript was created by Adobe in the mid 1980s and, combined with the introduction of the Macintosh and the Apple LaserWriter printer, created an industry called desktop publishing.

The second standard is called TrueType. TrueType fonts use a variant of postscript technology. To use a TrueType font only one file must be installed on the host computer. This file is used by the printer and by the screen to produce attractive text at any point size. TrueType technology, however, is limited to text. For high-resolution graphics, Postscript is the standard to use. TrueType was created in the early 1990s by Microsoft in cooperation with Apple Computer and others.

Both Macintosh and Windows laptop and desktop computers commonly use TrueType fonts. Postscript technology, however, is much more commonly available on the Macintosh platform because of its dominance in the desktop publishing and multimedia production industries.

7.5.1.5 Styles and Sizes

Styles such as Bold, Underlined, and Italics can be applied to most fonts.

The size of the font also can be altered through software commands.

7.5.1.6 File Formats

Text created on a computer is stored as a file on a hard disk or floppy disk. The ASCII file format, aka plain text, is universally understood by all computer systems. A more complex standard called Rich Text Format (RTF) was developed by Microsoft to allow for the exchange of word processing files that include formatting such as text alignment, font styles, and font sizes. Although RTF is proprietary technology, it has become a defacto standard for exchanging formatted text documents. A quickly-emerging replacement for RTF, however, is HTML (HyperText Markup Language) which is used for creating Web pages. HTML files are really just ASCII text files. The content of HTML files, however, contains a standard set of markings to indicate text styles, alignments, hypertext links, graphics, and other formatting essentials. HTML files can be read by Web browser software like Netscape Navigator. Many word processors today are also equipped to interpret HTML. Other file formats such as the native file formats used by Microsoft Word, WordPerfect, and AppleWorks are proprietary and not universally understood. When preparing electronic documents for a wide audience, therefore, it is best to use ASCII, RTF, or HTML.

7.5.2 Video and Sound

Sound is used to set the rhythm or a mood in a package. Speech gives an effect of a language (pronunciation) for instance.

Sound files in various Sound File Formats like the MP3 can be easily transmitted through the NET. Voice over IP VOIP is an upcoming field with a great future. Sound can be recorded into a mic or from any other medium like tape or cassette player onto the PC. Some of the factors effecting the size of sound files are:

The method of storage used, whether the format any kind of compression, resolution i.e bit rate, whether the sound is mono or stereo, the quality of sound desired. If pictures can paint a thousand words than motion pictures can paint a million. Digital video is usually produced from analog video as it is much easier to transmit digital data and the advantages of digital over analog are quite pronounced.

7.5.3 What is Sound?

If a tree falls in the forest and no living creature is there to hear it, does it make a sound? The answer is no. Sound is a perceptual phenomenon only. When a tree falls, a person speaks, or a violin string

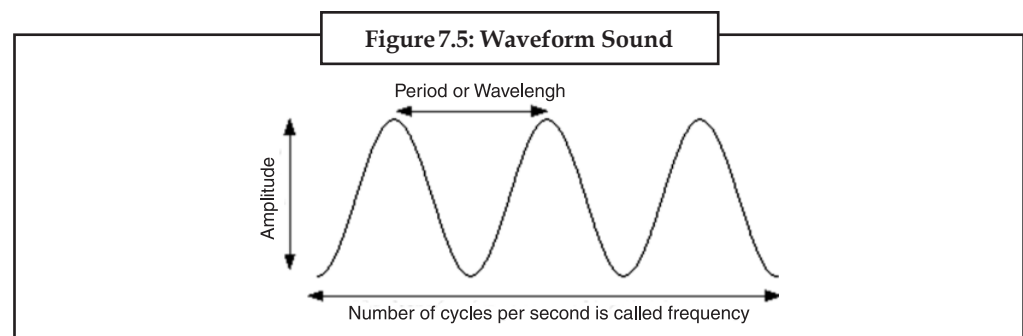
Notes

vibrates, the surrounding air is disturbed causing changes in air pressure that are called sound waves. When sound waves arrive at our ears they cause small bones in our ears to vibrate. These vibrations then cause nerve impulses to be sent to the brain where they are interpreted as sound.

7.5.3.1 How is Sound Recorded?

Sound waves can be transduced (converted to another form) using a microphone. A microphone is similar to the human ear in that it has a diaphragm which vibrates in response to changes in air pressure. The movements of the diaphragm within an electromagnetic field cause changes in electrical voltage. These voltage changes can be directed to a tape recorder which alters the magnetic particles on the tape to correspond to the voltage changes. A “picture” of the sound then exists on the tape. When you press play on the tape recorder, the “picture” is read back as a series of voltage changes which are then sent to a speaker. The voltage changes cause an electromagnet within the speaker to push and pull on a diaphragm. The movement of the diaphragm then causes air pressure changes which our ears interpret as the original sound. This process is known as analog recording because the picture of the sound on the tape is analogous to the original changes in air pressure caused by the sound event.

Usually we represent sound visually as a waveform. The height is called the amplitude and represents volume. The distance between cycles is called the period or wavelength. The number of cycles per second is called frequency and is interpreted by our ears as pitch. Frequency is measured in Herz (Hz) or kilohertz (kHz).



The waveform above is a simple sine wave. Typical sounds are more complex in appearance. Here is a waveform of a short spoken phrase. Note the frequent changes in wavelength, amplitude, and frequency.

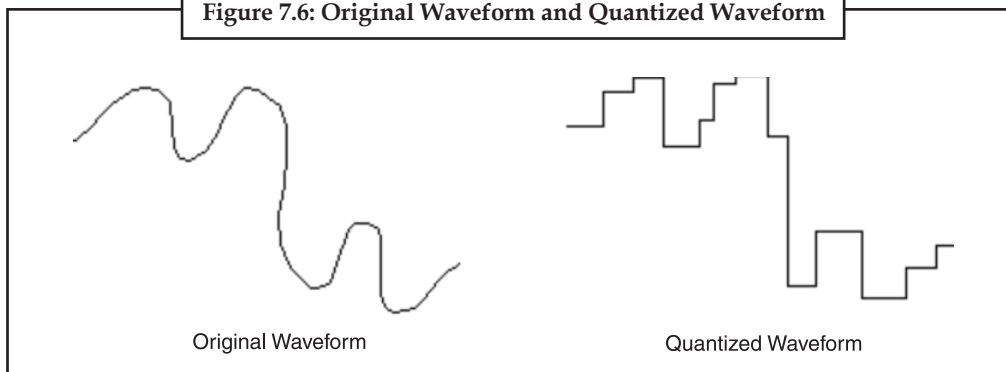
Digital recording differs from analog recording in that the “picture” of the sound is created by measuring the voltage changes coming from the microphone and assigning numbers to each measurement. The term “sampling” is used to describe the process of measuring an electrical signal’s voltage thousands of times per second at a given level of precision (resolution). The number of measurements per second is called the “sampling rate” and is expressed as kilohertz (kHz). A rate of 11,000 measurements per second is thus designated as 11 kHz. Sampling rates range from 5 kHz to 48 kHz with higher rates being used for the best quality recordings. The frequency range of a digitized sound is limited to one-half the sampling rate. Since humans can hear frequencies in a range of 20 hertz to about 20 kilohertz, it is necessary to sample at more than 40 kilohertz to capture the full range of frequencies perceptible to the human ear.

The number of measurements per second, however, is only part of the picture. The degree of precision within each measurement is also important. This is known as “sampling resolution”. Sampling resolution is used to divide the total range of the electrical voltage into discrete parts. Common sampling resolutions in use today are 8-bit and 16-bit. Sampling at 8-bits divides the voltage into 256 parts

(2 to the 8th power). Sampling at 16-bits divides the voltage into 65,536 parts (2 to the 16th power). Using a higher sampling resolution creates cleaner recordings with less background noise. Higher sampling resolutions also capture a wider dynamic range. For example an 8-bit digitizer will only capture sounds up to 48 decibels (DB). Any portion of the sound that is louder than 48 DB will be clipped and the resulting sample will sound distorted. 16-bit digitizers, however, capture up to 96 DB of volume. The dynamic range of the human ear extends to 120 DB.

Quantization is the term that describes the process of measuring the amplitude of a sound and rounding off the measurements according to the sampling resolution. For example, an 8-bit sound digitizer will assign integer values of between 0 and 255 for the amplitude of each sample. The result is that the original smooth waveform is reconstructed as a staircase shape with only 256 discrete levels of amplitude and noise is introduced into the signal. 16-bit digitizers, on the other hand, assign amplitude values on a scale of 0 to 65,535. At that level of precision, the reconstructed waveform is almost identical to the original and almost no noise is introduced.

Figure 7.6: Original Waveform and Quantized Waveform



All of these measurements are made by an analog-to-digital converter. The measurements can then be stored as binary numbers in a file on a computer's hard disk. To play back the sound, the computer sends the information in the file to a digital-to-analog converter which reproduces the original electrical signal. That signal is then sent to a speaker which produces the sound as described earlier.

Maximum precision per measurement combined with maximum sampling rates produces the highest quality recordings. To describe a digital recording of a sound, therefore, one can speak of the sampling rate and resolution. For example, sound recorded at a sampling rate of 22 kHz with 8-bit resolution is considered to be of a quality similar to that of a telephone call. Sound recorded at 44 kHz and 16-bits is considered the minimum quality for compact disc recordings because it captures the full range of human hearing. In multimedia production work, 11 kHz, 8-bit sound is sometimes acceptable for speech recordings and 22 kHz, 8-bit resolution or 11 KHz, 16-bit resolution is often considered acceptable for music. For the highest-level multimedia work, however, nothing short of 44 kHz, 16-bit sound is acceptable.

When sound waves strike a microphone, they are converted to an electrical signal which is measured many thousand times per second by an analog-to-digital converter chip. The measurements are stored in the computer as binary numbers.

The higher the quality of sound, the more space it takes to store the sound. A compact disc can store about 74 minutes of stereo sound at 44 kHz, 16-bit. If you reduce the quality to 22 kHz, 8-bit stereo sound, however, you can store approximately 300 minutes of audio on the same disc. In other words, one minute of stereo sound takes 10 megabytes of storage at 44 kHz, 16-bit quality, and only 2.5 megabytes of storage at 22 kHz, 8-bit quality. When producing sound for multimedia, therefore, one must consider not only sound quality, but also how the sound will be distributed. If your multimedia program will be distributed on CD then you may have enough storage space

Notes

to justify using the best quality. If the program will be distributed on disk or through the internet, however, you would consider using lower quality sound to avoid having to distribute many disks or subject your users to long download times.

7.5.3.2 Sound File Formats

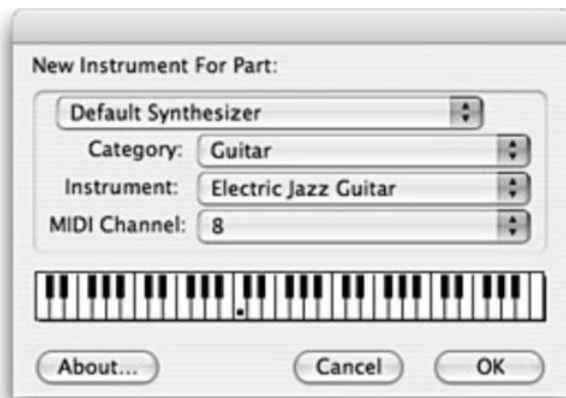
When sound is digitally recorded to a hard disk, a file format is assigned by the recording software. Sound files are either RAM-based or Disk-based. To play back a RAM-based file, your computer must have enough random access memory (RAM) to hold the entire file. For example a computer with 8 megabytes of RAM might not be able to play a large RAM-based sound file but a computer with 16 megabytes of RAM might have no problem with it. As a result, RAM-based sound file formats are appropriate for use with short sound samples. On the Macintosh, System 7 sound and SND resource are common RAM-based file formats. System 7 sounds are used to generate the various beeps and alert sounds used on the Macintosh. SND resources are often used as sound resources in HyperCard stacks. A Macintosh sound recording program, such as Macromedia's SoundEdit 16 or the freeware SoundHandle 1.0.3 can be used to create SND resources that can be saved directly into the resource fork of a HyperCard stack. System 7 and SND file formats are most commonly used with 22 kHz, 8-bit sound samples.

Disk-based sound file formats allow you to record music of any length and quality. You are only limited by the amount of available storage space on your hard drive. Disk-based sound file formats are ideal for longer and/or higher-quality samples. AIFF (Audio Interchange File Format) is one of the most commonly-used disk-based file formats on Macintosh, Windows, and even Unix computers. Stereo AIFF sound files recorded at 44 kHz, 16-bit quality are ideal for multimedia productions that will be distributed on CD. Monophonic AIFF sound files recorded at 22 kHz, 16-bit quality are better for multimedia productions that will be distributed via the internet because their file sizes are smaller than higher-quality samples. If you use the internet frequently you have probably encountered sound files in WAV and AU formats. The WAV format is used by Microsoft Windows and the AU file format is used by computers running the UNIX operating system. Sound editing software can convert among these and many other file formats.

7.5.3.3 MIDI

The Musical Instrument Digital Interface (MIDI) is a hardware and software standard that, among other things, allows users to record a complete description of a lengthy musical performance using only a small amount of disk space. Standard MIDI Files can be played back using the sound synthesis hardware of a Mac or PC. Using a digital audio file format like AIFF, the same symphony uses over 300 megabytes of hard disk storage. One problem with MIDI is that the quality of the actual sound you hear will vary depending on the quality of your computer's sound hardware. For educational applications, however, MIDI-generated sound can be used to demonstrate musical ideas quite effectively. Another problem with MIDI in the past was the lack of a standard sound set. A MIDI file designed to be played with piano and flute sounds might be realized with organ and clarinet on another person's computer. This problem was partially solved by the advent of the General MIDI standard which created a standard set of 128 sounds. Virtually all MIDI files today are distributed in General MIDI format. Still it was left to the owner of each computer to be sure their sound hardware could play the General MIDI sounds. Apple Computer solved the problem with the latest version of its QuickTime software.

Figure 7.7: Demonstration of MIDI and QuickTime



The final product is embedded below.

You are welcome to download the original MIDI file, *bachinv4.mid*, for use in your music sequencing or music notation applications.

Web sites can be used to exchange MIDI files, collaborate on MIDI sequences, and engage in group compositions. If you convert your MIDI files to QuickTime movies then multiple MIDI files can be embedded in a single page, allowing visitors to participate in a jam session.

The conversion of analog video into its digital equivalent requires a special hardware called video capture card. For the past 15 years three of the most influential communication industries have been converging:

- (a) Publishing
- (b) Broadcasting
- (c) Computing

Interactive multimedia and electronic publishing are products of this convergence and their impact on all kinds of communication from marketing to education is immense.

Multimedia Development Tools: They are required for building applications and reviewing some of the products that are commercially available commercially.

Presentation Tools: These tools are necessary to create multimedia presentations on a PC. Presentation tools are tools like overhead projectors, these tools actually improve the overall effect and helps the speaker to get his message across in a professional manner.

Authoring Tools: These in contrast with the presentation tools support features such as layout graphic design animation control of branching and navigation the manner in which the end user will be able to move through the application. Authoring tools may also provide screen design help to harness the layout of text images and places where user interaction is required. Libraries may support audiovisual and graphics functions and implement multitasking capabilities under different operating systems. Some of these authoring tools are Authorware and Director from Macromedia, Hypercarn from Apple Inc. etc.

Notes

What is Authoring?

It is the whole process of developing a multimedia package. It usually describes the integration of all the multimedia data into a single coherent entity, i.e. the package itself.

All the components of the package must work in tandem with each other and have a sort of flow and synchronism.

The methods adopted for authoring purposes can be grouped as:

1. FRAME or Multimedia PAGE based
2. ICON type



Case Study

Success Story of Apple Computer's QuickTime Software:

One of Apple Computer's most brilliant innovations is the continuing development of QuickTime. QuickTime began as a set of system extensions to Macintosh System 7 to allow users to play digitized video in a small window on the screen. Today QuickTime is a comprehensive multimedia tool for storing video, animations, and sound in a variety of formats. It is also a cross-platform tool, meaning that QuickTime movies can be viewed and heard using computers running Mac OS, Windows, or even UNIX.

MoviePlayer can be used to convert audio from compact discs into QuickTime movies that can be used in multimedia presentations. MoviePlayer can be used to add sound and text tracks to digital video. Using a video recorder, Apple's free Video Player software, and a Mac equipped with video input, you could record a movie demonstrating instrumental techniques and then use MoviePlayer to add a voiceover narrative. You could also add a descriptive voice narrative to a QuickTime MIDI movie containing a full performance of a complex work. QuickTime comes with several software CODECs (compressor/decompressor) to reduce file size while retaining quality. For music, the QDesign Music Compressor is excellent. For speech, try the Qualcomm PureVoice Compressor is a good choice. For video, the Sorenson compressor does an impressive job of reducing file size for the visual portion of the video. When used in combination with the QDesign or Qualcomm audio compressors, file size can be made manageable for transmission over the internet. A "Fast Start" feature is also available to allow the movie to begin playing while still downloading to the user's computer. The next version of QuickTime, version 4.0 currently in Beta testing, allows for streaming live content as well.

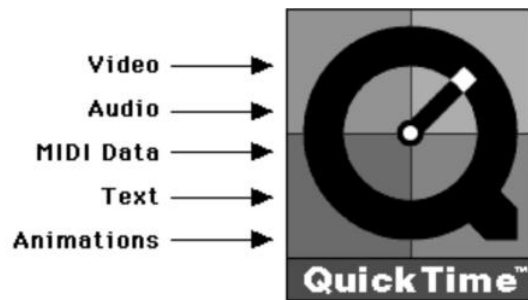
QuickTime movies can be loaded onto any web server and included in web pages by using the appropriate EMBED code.

```
<EMBED SRC="doodle16.mov" AUTOPLAY=FALSE WIDTH=150 HEIGHT=24>
```

For more detailed and advanced editing of video and audio, of course, you might purchase professional software like Adobe Premiere and MacroMedia SoundEdit 16. Using free and shareware software available from Apple and others, however, you can create multimedia presentations to inspire and educate your students.

Contd...

Figure 7.8: QuickTime



Apple Computer's QuickTime software can be used to create movies with any combination of video, audio, MIDI data, text, and animations.

Questions:

1. What is Apple Computer's most brilliant innovations?
2. What are various uses of Apple Computer's QuickTime software?

7.6 Summary

- Multimedia is usually recorded and played, displayed or accessed by information content processing devices.
- Graphics software or image editing software is a program or collection of programs that enable a person to manipulate visual image on a computer.
- Most graphics programs have the ability to import one or more graphics file formats.
- Multimedia means multicomcommunication.

7.7 Keywords

BMP file format: The BMP file format (Windows bitmap) handles graphics files within the Microsoft Windows OS. Typically, BMP files are uncompressed, hence they are large; the advantage is their simplicity and wide acceptance in Windows programs.

CGM (Computer Graphics Metafile): CGM (Computer Graphics Metafile) is a file format for 2D vector graphics, raster graphics, and text, and is defined by ISO/IEC 8632.

Etching: Etching is an intaglio method of printmaking in which the image is incised into the surface of a metal plate using an acid.

JPEG 2000: JPEG 2000 is a compression standard enabling both lossless and lossy storage.

Line art: Line art is a rather non-specific term sometimes used for any image that consists of distinct straight and curved lines placed against a (usually plain) background, without gradations in shade (darkness) or hue (color) to represent two-dimensional or three-dimensional objects. Line art is usually monochromatic, although lines may be of different colors.

Metafile formats: Metafile formats are portable formats which can include both raster and vector information.

Notes

Raster formats: These formats store images as bitmaps (also known as pixmaps).

RAW: RAW refers to a family of raw image formats that are options available on some digital cameras.

SVG (Scalable Vector Graphics): SVG (Scalable Vector Graphics) is an open standard created and developed by the World Wide Web Consortium to address the need (and attempts of several corporations) for a versatile, scriptable and all-purpose vector format for the web and otherwise.

TIFF (Tagged Image File Format): The TIFF (Tagged Image File Format) format is a flexible format that normally saves 8 bits or 16 bits per color (red, green, blue) for 24-bit and 48-bit totals, respectively, usually using either the **TIFF** or **TIF** filename extension.

Vector file formats: Vector file formats can contain bitmap data as well. 3D graphic file formats are technically vector formats with pixel data texture mapping on the surface of a vector virtual object, warped to match the angle of the viewing perspective.



Lab Exercise

1. Draw Original wave form and aquantized wave form.
2. Demonstration 2 - MIDI and QuickTime.

7.8 Self-Assessment Questions

1. Etching is only used in the manufacturing of printed circuit boards and semiconductor devices.
(a) True (b) False
2. Multimedia is usually recorded and played, displayed or accessed by information content processing devices, such as computerized and electronic devices, but can also be part of a live performance.
(a) True (b) False
3. What image types Web pages require?
(a) JPG (b) GIF
(c) PNG (d) All of the above
4. Photo images have
(a) Discontinuous tones (b) Continuous tones,
(c) Wavy tones (d) None of these
5. Files are very small files for continuous tone photo images.
(a) JPG (b) GIF
(c) PNG (d) All of these
6. In addition to straight image formats, which formats are portable formats.
(a) Raster formats (b) JPEG
(c) Metafile (d) All of these

7. RAW refers to a family of raw image formats that are options available on some digital cameras.
- (a) True (b) False
8. WebP is a old image format that uses lossy compression.
- (a) True (b) False
9. What is Graphics software?
- (a) It is a program or collection of programs that enable a person to manipulate visual images on Web.
- (b) It is a program or collection of programs that disable a person to manipulate visual images on a computer.
- (c) It is a program or collection of programs that enable a person to manipulate visual images on a computer.
- (d) All of the above.
10. What is the major drawback of using text?
- (a) It is not user friendly as compared to the other medium.
- (b) It is not flexible as compared to the other medium.
- (c) It is not supportive as compared to the other medium.
- (d) None of the above

7.9 Review Questions

1. Explain Graphics and Multimedia.
2. What are Major characteristics of multimedia?
3. Finds application of Multimedia.
4. Explain Image File Formats (TIF, JPG, PNG, GIF).
5. Find Difference in photo and graphics images.
6. What is Image file size?
7. What is Image file compression?
8. Explain Major graphic file formats.
9. What is Authoring?
10. How will you Transfer Pictures to Your PC with a USB Data Cable?
11. How will you Transfer Pictures to Your PC Using BlueTooth?
12. Explain components of a multimedia package.
13. What is Text?
14. What is Sound and how is Sound Recorded?
15. What is Musical Instrument Digital Interface (MIDI)?
16. Explain Color data modeBits per pixel.

Notes

Answers for Self-Assessment Questions

- | | | | | |
|--------|--------|--------|--------|---------|
| 1. (b) | 2. (a) | 3. (d) | 4. (b) | 5. (a) |
| 6. (c) | 7. (a) | 8. (b) | 9. (c) | 10. (a) |

7.10 Further Reading



Books

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition.



Online link

<http://www.symbio.com/solutions/mobile-embed>

Unit 8: Database System

Notes

CONTENTS

Objectives

Introduction

8.1 Database

8.1.1 Types of Database

8.1.2 Database Models

8.2 The DBMS

8.2.1 Building Blocks of DBMS

8.3 Working with Database

8.3.1 Relational Databases

8.3.2 Three Rules for Database Work

8.4 Database at Work

8.4.1 Database Transaction

8.5 Common Corporate DBMS

8.5.1 ORACLE

8.5.2 DB2

8.5.3 Microsoft Access

8.5.4 Microsoft SQL Server

8.5.5 PostgreSQL

8.5.6 MySQL

8.5.7 Filemaker

8.6 Summary

8.7 Keywords

8.8 Self-Assessment Questions

8.9 Review Questions

8.10 Further Reading

Objectives

After studying this unit, you will be able to:

- Explain Database.
- Discuss DBMS.
- Understand working with database.
- Explain database at work.
- Explain common corporate DBMS.

Introduction

All of us are familiar with the term data. In fact, unknowingly we come across data in our day-to-day life everyday. The age of a person, price of potato, number of students in a school, pin code of a city, etc. are some examples of data. In our life we have to remember so much of data. But it is easier for us to remember all information for a few individuals. For example, you may be in a position to tell accurately the age, height, complexion, income, educational qualification, residential address, etc. of your close friends. But it is too difficult for you to memorise all these information for a large number of individuals. Let us consider the example of National Open School (NOS). Every year about one lakh students take admission in NOS. If you are asked to memorise records of date of birth, subjects offered and postal address of all these students, it will not be possible for you.

To deal with such problems we construct a database. We arrange all information about students in a tabular form. We keep all the records so that if I am asked, 'How many students are there in Economics?' I am in a position to answer.

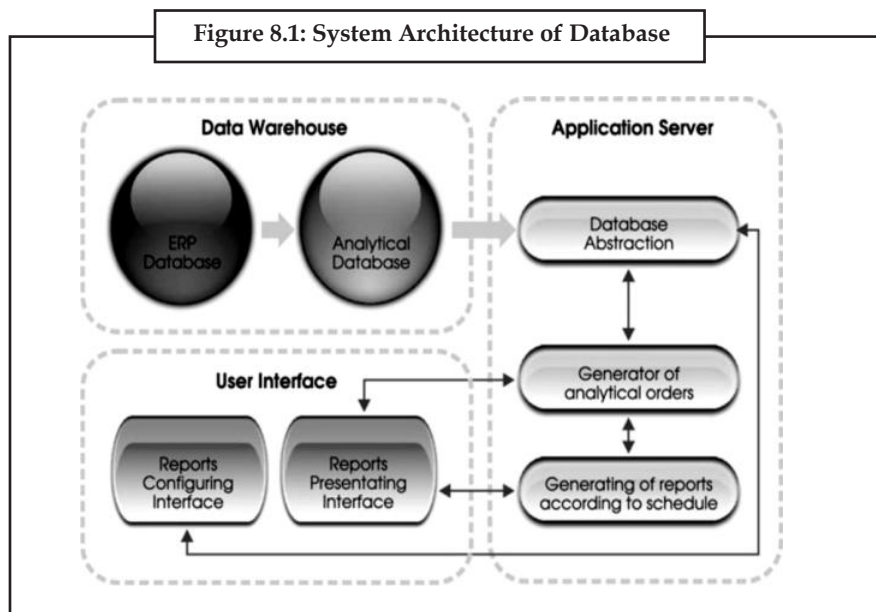
8.1 Database

A **database** is a system intended to organize, store, and retrieve large amounts of data easily. It consists of an organized collection of data for one or more uses, typically in digital form. One way of classifying databases involves the type of their contents, for example: bibliographic, document-text, statistical. Digital databases are managed using database management systems, which store database contents, allowing data creation and maintenance, and search and other access.

A database is a collection of information that is organized so that it can easily be accessed, managed, and updated. In one view, databases can be classified according to types of content: bibliographic, full-text, numeric, and images.

In computing, databases are sometimes classified according to their organizational approach. The most prevalent approach is the relational database, a tabular database in which data is defined so that it can be reorganized and accessed in a number of different ways. A distributed database is one that can be dispersed or replicated among different points in a network. An object-oriented programming database is one that is congruent with the data defined in object classes and subclasses.

Computer databases typically contain aggregations of data records or files, such as sales transactions, product catalogs and inventories, and customer profiles. Typically, a database manager provides users the capabilities of controlling read/write access, specifying report generation, and analyzing usage. Databases and database managers are prevalent in large mainframe systems, but are also present in smaller distributed workstation and mid-range systems such as the AS/400 and on personal computers. SQL (Structured Query Language) is a standard language for making interactive queries from and updating a database such as IBM's DB2, Microsoft's Access, and database products from Oracle, Sybase, and Computer Associates.



A **database** is a system intended to organize, store, and retrieve large amounts of data easily.

8.1.1 Types of Database

Analytical Database

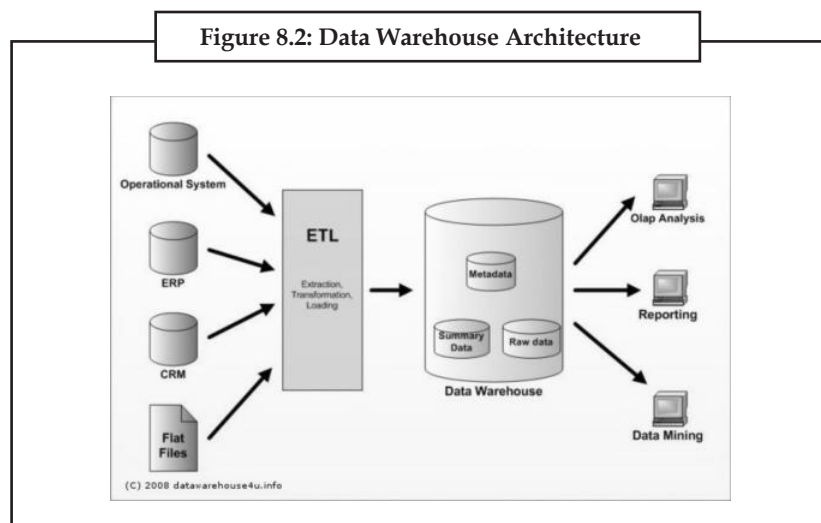
Analysts may do their work directly against a data warehouse or create a separate analytic database for Online Analytical Processing. For example, a company might extract sales records for analyzing the effectiveness of advertising and other sales promotions at an aggregate level.

Data Warehouse

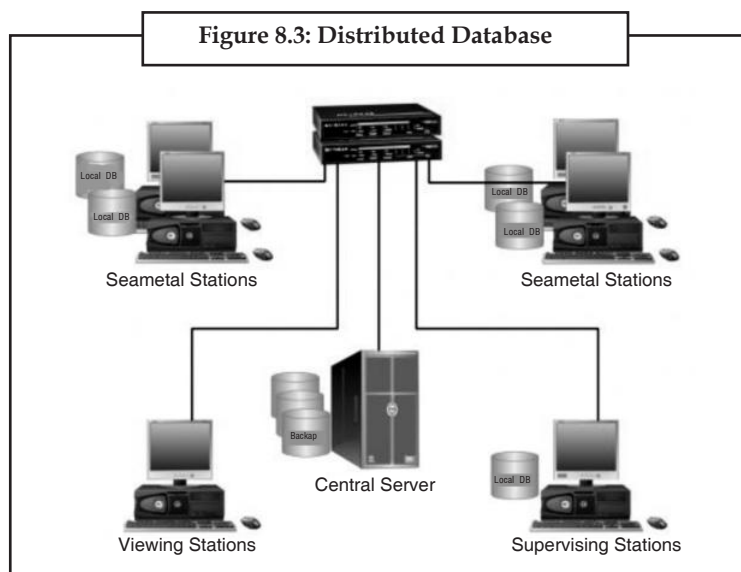
Data warehouses archive modern data from operational databases and often from external sources such as market research firms. Often operational data undergoes transformation on its way into the warehouse, getting summarized, anonymized, reclassified, etc. The warehouse becomes the central source of data for use by managers and other end-users who may not have access to operational data. For example, sales data might be aggregated to weekly totals and converted from internal product codes to use UPC codes so that it can be compared with ACNielsen data. Some basic and essential components of data warehousing include retrieving and analyzing data, transforming, loading and managing data so as to make it available for further use.

Operations in a data warehouse are typically concerned with bulk data manipulation, and as such, it is unusual and inefficient to target individual rows for update, insert or delete. Bulk native loaders for input data and bulk SQL passes for aggregation are the norm.

Notes

*Distributed Database*

These are databases of local work-groups and departments at regional offices, branch offices, manufacturing plants and other work sites. These databases can include segments of both common operational and common user databases, as well as data generated and used only at a user's own site.

*End-user Database*

These databases consist of data developed by individual end-users. Examples of these are collections of documents in spreadsheets, word processing and downloaded files, even managing their personal baseball card collection.

External Log Database

These databases contain data collected for use across multiple organizations, either freely or via subscription. The Internet Movie Database is one example.

Hypermedia Log Databases

The World Wide Web can be thought of as a database, albeit one spread across millions of independent computing systems. Web browsers “process” this data one page at a time, while Web crawlers and other software provide the equivalent of database indexes to support search and other activities.

Operational Log Database

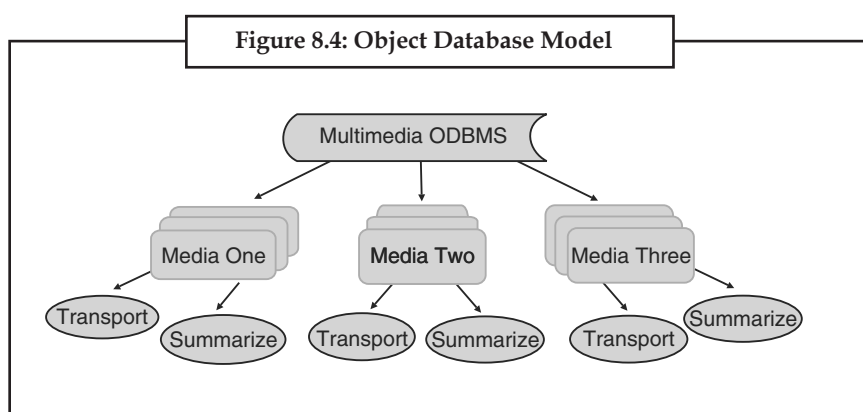
These databases store detailed data about the operations of an organization. They are typically organized by subject matter, process relatively high volumes of updates using transactions. Essentially every major organization on earth uses such databases. Examples include customer databases that record contact, credit, and demographic information about a business’ customers, personnel databases that hold information such as salary, benefits, skills data about employees, Enterprise resource planning that record details about product components, parts inventory, and financial databases that keep track of the organization’s money, accounting and financial dealings.

8.1.2 Database Models

8.1.2.1 Object Database Models

In recent years, the object-oriented paradigm has been applied in areas such as engineering and spatial databases, telecommunications and in various scientific domains. The conglomeration of object oriented programming and database technology led to this new kind of database. These databases attempt to bring the database world and the application-programming world closer together, in particular by ensuring that the database uses the same type system as the application program. This aims to avoid the overhead (sometimes referred to as the impedance mismatch) of converting information between its representation in the database (for example as rows in tables) and its representation in the application program (typically as objects). At the same time, object databases attempt to introduce key ideas of object programming, such as encapsulation and polymorphism, into the world of databases.

A variety of these ways have been tried [by whom?] for storing objects in a database. Some products have approached the problem from the application-programming side, by making the objects manipulated by the program persistent. This also typically requires the addition of some kind of query language, since conventional programming languages do not provide language-level functionality for finding objects based on their information content. Others [which?] have attacked the problem from the database end, by defining an object-oriented data model for the database, and defining a database programming language that allows full programming capabilities as well as traditional query facilities.



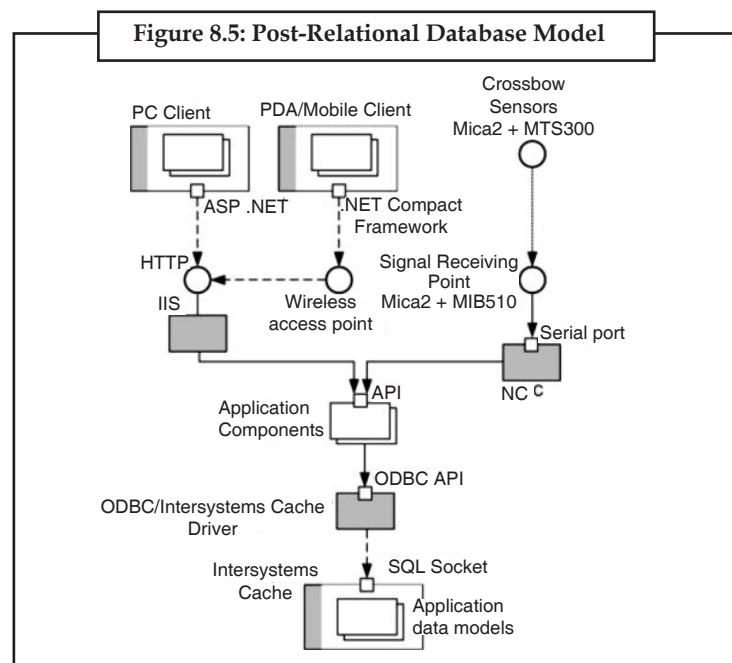
Notes

8.1.2.2 Post-relational Database Models

Products offering a more general data model than the relational model are sometimes classified as post-relational. Alternate terms include “hybrid database”, “Object-enhanced RDBMS” and others. The data model in such products incorporates relations but is not constrained by E.F. Codd’s Information Principle, which requires that all information in the database must be cast explicitly in terms of values in relations and in no other way.

Some of these extensions to the relational model integrate concepts from technologies that pre-date the relational model. For example, they allow representation of a directed graph with trees on the nodes.

Some post-relational products extend relational systems with non-relational features. Others arrived in much the same place by adding relational features to pre-relational systems. Paradoxically, this allows products that are historically pre-relational, such as PICK and MUMPS, to make a plausible claim to be post-relational.

**8.2 The DBMS**

As one of the oldest components associated with computers, the database management system, or DBMS, is a computer software program that is designed as the means of managing all databases that are currently installed on a system hard drive or network. Different types of database management systems exist, with some of them designed for the oversight and proper control of databases that are configured for specific purposes. Here are some examples of the various incarnations of DBMS technology that are currently in use, and some of the basic elements that are part of DBMS software applications.

As the tool that is employed in the broad practice of managing databases, the DBMS is marketed in many forms. Some of the more popular examples of DBMS solutions include Microsoft Access, FileMaker, DB2, and Oracle. All these products provide for the creation of a series of rights or privileges that can be associated with a specific user. This means that it is possible to designate

Notes

one or more database administrators who may control each function, as well as provide other users with various levels of administration rights. This flexibility makes the task of using DBMS methods to oversee a system something that can be centrally controlled, or allocated to several different people.

There are four essential elements that are found with just about every example of DBMS currently on the market. The first is the implementation of a modeling language that serves to define the language of each database that is hosted via the DBMS. There are several approaches currently in use, with hierarchical, network, relational, and object examples. Essentially, the modeling language ensures the ability of the databases to communicate with the DBMS and thus operate on the system.

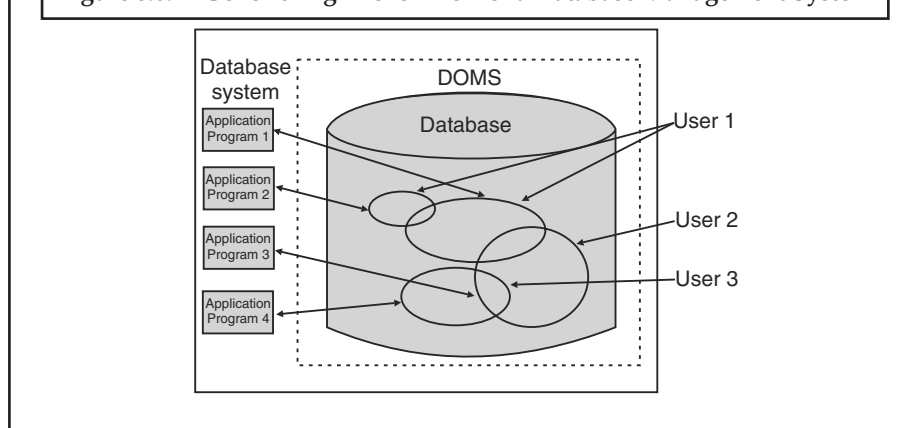
Second, data structures also are administered by the DBMS. Examples of data that are organized by this function are individual profiles or records, files, fields and their definitions, and objects such as visual media. Data structures are what allow DBMS to interact with the data without causing and damage to the integrity of the data itself.

A third component of DBMS software is the data query language. This element is involved in maintaining the security of the database, by monitoring the use of login data, the assignment of access rights and privileges, and the definition of the criteria that must be employed to add data to the system. The data query language works with the data structures to make sure it is harder to input irrelevant data into any of the databases in use on the system.

Last, a mechanism that allows for transactions is an essential basic for any DBMS. This helps to allow multiple and concurrent access to the database by multiple users, prevents the manipulation of one record by two users at the same time, and preventing the creation of duplicate records.

A database management system (DBMS) consists of software that operates databases, providing storage, access, security, backup and other facilities. Database management systems can be categorized according to the database model that they support, such as relational or XML, the type(s) of computer they support, such as a server cluster or a mobile phone, the query language(s) that access the database, such as SQL or XQuery, performance trade-offs, such as maximum scale or maximum speed or others. Some DBMS cover more than one entry in these categories, e.g., supporting multiple query languages. Examples of some commonly used DBMS are MySQL, PostgreSQL, Microsoft Access, SQL Server, FileMaker, Oracle, Sybase, dBASE, Clipper, FoxPro etc. Almost every database software comes with an Open Database Connectivity (ODBC) driver that allows the database to integrate with other databases.

Figure 8.6: A Generic High Level View of a Database Management System



Notes

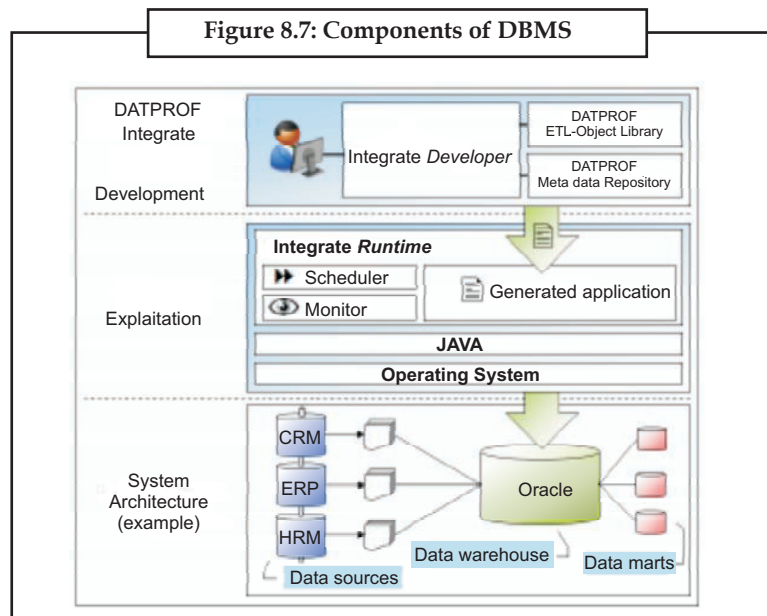


A database management system (DBMS) consists of software that operates databases, providing storage, access, security, backup and other facilities.

8.2.1 Building Blocks of DBMS

8.2.1.1 Components

- (a) **DBMS Engine** accepts logical requests from various other DBMS subsystems, converts them into physical equivalents, and actually accesses the database and data dictionary as they exist on a storage device.
- (b) **Data Definition Subsystem** helps the user create and maintain the data dictionary and define the structure of the files in a database.
- (c) **Data Manipulation Subsystem** helps the user to add, change, and delete information in a database and query it for valuable information. Software tools within the data manipulation subsystem are most often the primary interface between user and the information contained in a database. It allows the user to specify its logical information requirements.
- (d) **Application Generation Subsystem** contains facilities to help users develop transaction-intensive applications. It usually requires that the user perform a detailed series of tasks to process a transaction. It facilitates easy-to-use data entry screens, programming languages, and interfaces.
- (e) **Data Administration Subsystem** helps users manage the overall database environment by providing facilities for backup and recovery, security management, query optimization, concurrency control, and change management.



8.2.1.2 Modeling Language

A modeling language is a data modeling language to define the schema of each database hosted in the DBMS, according to the DBMS database model. Database management systems (DBMS)

are designed to use one of five database structures to provide simplistic access to information stored in databases. The five database structures are:

- the hierarchical model,
- the network model,
- the relational model,
- the multidimensional model, and
- the object model.

Inverted lists and other methods are also used. A given database management system may provide one or more of the five models. The optimal structure depends on the natural organization of the application's data, and on the application's requirements, which include transaction rate (speed), reliability, maintainability, scalability and cost.

The **hierarchical structure** was used in early mainframe DBMS. Records' relationships form a treelike model. This structure is simple but nonflexible because the relationship is confined to a one-to-many relationship. IBM's IMS system and the RDM Mobile are examples of a hierarchical database system with multiple hierarchies over the same data. RDM Mobile is a newly designed embedded database for a mobile computer system. The hierarchical structure is used primarily today for storing geographic information and file systems.

The **network structure** consists of more complex relationships. Unlike the hierarchical structure, it can relate to many records and accesses them by following one of several paths. In other words, this structure allows for many-to-many relationships.

The **relational structure** is the most commonly used today. It is used by mainframe, midrange and microcomputer systems. It uses two-dimensional rows and columns to store data. The tables of records can be connected by common key values. While working for IBM, E.F. Codd designed this structure in 1970. The model is not easy for the end user to run queries with because it may require a complex combination of many tables.

The **multidimensional structure** is similar to the relational model. The dimensions of the cube-like model have data relating to elements in each cell. This structure gives a spreadsheet-like view of data. This structure is easy to maintain because records are stored as fundamental attributes — in the same way they are viewed — and the structure is easy to understand. Its high performance has made it the most popular database structure when it comes to enabling online analytical processing (OLAP).

The **object oriented structure** has the ability to handle graphics, pictures, voice and text, types of data, without difficulty unlike the other database structures. This structure is popular for multimedia Web-based applications. It was designed to work with object-oriented programming languages such as Java.

The dominant model in use today is the ad hoc one embedded in SQL, despite the objections of purists who believe this model is a corruption of the relational model since it violates several fundamental principles for the sake of practicality and performance. Many DBMSs also support the Open Database Connectivity API that supports a standard way for programmers to access the DBMS.

Before the database management approach, organizations relied on file processing systems to organize, store, and process data files. End users criticized file processing because the data is stored in many different files and each organized in a different way. Each file was specialized to be used with a specific application. File processing was bulky, costly and nonflexible when it came to supplying needed data accurately and promptly. Data redundancy is an issue with

Notes

the file processing system because the independent data files produce duplicate data so when updates were needed each separate file would need to be updated. Another issue is the lack of data integration. The data is dependent on other data to organize and store it. Lastly, there was not any consistency or standardization of the data in a file processing system which makes maintenance difficult. For these reasons, the database management approach was produced.

8.2.1.3 Data Structure

Data structures (fields, records, files and objects) optimized to deal with very large amounts of data stored on a permanent data storage device (which implies relatively slow access compared to volatile main memory).

8.2.1.4 Database Query Language

A database query language and report object allows users to interactively interrogate the database, analyze its data and update it according to the users' privileges on data. It also controls the security of the database. Data security prevents unauthorized users from viewing or updating the database. Using passwords, users are allowed access to the entire database or subsets of it called subschemas. For example, an employee database can contain all the data about an individual employee, but one group of users may be authorized to view only payroll data, while others are allowed access to only work history and medical data.

If the DBMS provides a way to interactively enter and update the database, as well as interrogate it, this capability allows for managing personal databases. However, it may not leave an audit trail of actions or provide the kinds of controls necessary in a multi-user organization. These controls are only available when a set of application programs are customized for each data entry and updating function.

8.2.1.5 Transaction Mechanism

A database transaction mechanism ideally guarantees ACID properties in order to ensure data integrity despite concurrent user accesses (concurrency control), and faults (fault tolerance). It also maintains the integrity of the data in the database. The DBMS can maintain the integrity of the database by not allowing more than one user to update the same record at the same time. The DBMS can help prevent duplicate records via unique index constraints; for example, no two customers with the same customer numbers (key fields) can be entered into the database. See ACID properties for more information (Redundancy avoidance).

8.3 Working with Database

Almost all of the most useful sites on the web use databases to organise their content, and they often use them to allow users to register and leave comments too. Any time you do something that a website seems to 'remember' the next time, the chances are that a database is involved.

Yet, despite how common databases are, they aren't very well understood. Every day, new webmasters become database administrators without even understanding the first thing about databases. When you use a database on the web today, you're not just using any database: you're using ones that rely on concepts built up over decades of database development and proven effective. Here are some of those concepts.

8.3.1 Relational Databases

The most common database model in use today is that of the relational database — others include hierarchical databases (where data is organised in ‘trees’, like an organisation’s management structure), and flat file databases (where data is stored in ‘records’ in a text document).

In a relational database, data is stored in tables. The columns are called fields and the rows are called records. So, for example, a table might have two fields — firstname and lastname. If you then added a record to this table, it could be ‘Bob’ and ‘Smith’. Instead of just having that data, you have labelled it with what it is, and that lets you refer to it and search through it much more easily.

Where the ‘relational’ part is really significant, though, is when it comes to the way tables in a database relate to the other tables. Each record of each table has an ID number (technically known as the ‘primary key’) — for example, the Bob Smith record might be ID number 123. This then lets you refer to his record in a new table.

Let’s say you were storing records of people’s orders. You could have two columns: customer number and date. This lets you simply store 123 and the date in the table each time Bob Smith orders from you - the relational nature of the database will tell you later on that customer number 123 is Bob Smith. When it comes to things like, for example, storing posts made by multiple authors, this is powerful.

8.3.2 Three Rules for Database Work

Remote development is slow and difficult.

Avoid using a shared database at all costs, as they ultimately waste time and help produce bugs.

Always Have a Single, Authoritative Source For Your Schema

Ideally, this single source will be your source control repository (see rule #3). Consider the following conversation:

Developer 1: It’s time to push the app into testing. Do we copy the database from Jack’s machine, or Jill’s machine?

Developer 2: Un’t remember which one is up to date.

Developer 1: We’re screwed.

Everyone should know where the official schema resides, and have a frictionless experience in getting a fresh database setup. I should be able to walk up to a computer, get the latest from source control, build, and run a simple tool to setup the database (in many scenarios, the build process can even setup a database if none exists, so the process is one step shorter).

How you put your database into source control depends on your situation and preferences. Any decent O/R mapping tool should be able to create a database given the mappings you’ve defined in a project. You can also script out the database as a set of one or more files full of SQL DDL commands. I generally prefer to keep database views and programmatic features (including functions, triggers, and stored procedures) as separate files — but more on this in a later post.

Always Version Your Database

There are many ways to version databases, but the common goal is to propagate changes from development, to test, and ultimately to production in a controlled and consistent manner. A second goal is to have the ability to recreate a database at any point in time. This second goal is particularly important if you are shipping software to clients. If someone finds a bug in build 20070612.1 of your application, you must be able to recreate the application as it appeared in that build - database and all.

Notes

In a future post, I'll describe an approach I've used for database versioning that has worked well for many years of commercial development.



Task

Prepare database for school management using students and applications received.

8.4 Database at Work

Databases have been a staple of business computing from the very beginning of the digital era. In fact, the **relational database** was born in 1970. Since then, relational databases have grown in popularity to become the standard. Originally, databases were **flat**. This means that the information was stored in one long text file, called a **tab delimited file**. Each entry in the tab delimited file is separated by a special character, such as a vertical bar (|). Each entry contains multiple pieces of information (**fields**) about a particular object or person grouped together as a **record**. The text file makes it difficult to search for specific information or to create reports that include only certain fields from each record.

You can see that you have to search sequentially through the entire file to gather related information, such as age or salary. A relational database allows you to easily find specific information. It also allows you to sort based on any field and generate reports that contain only certain fields from each record. Relational databases use **tables** to store information.

In the relational database example, you can quickly compare salaries and ages because of the arrangement of data in columns. The relational database model takes advantage of this uniformity to build completely new tables out of required information from existing tables. In other words, it uses the relationship of similar data to increase the speed and versatility of the database.

The “relational” part of the name comes into play because of mathematical relations. A typical relational database has anywhere from 10 to more than 1,000 tables. Each table contains a column or columns that other tables can key on to gather information from that table. Look at the table below that matches the number in the City column of the above table with the name of a city.

By storing this information in another table, the database can create a single small table with the locations that can then be used for a variety of purposes by other tables in the database. A typical large database, like the one a big Web site, such as Amazon would have, will contain hundreds or thousands of tables like this all used together to quickly find the exact information needed at any given time.

Relational databases are created using a special computer language, **structured query language (SQL)**, that is the standard for database interoperability. SQL is the foundation for all of the popular database applications available today, from **Access** to **Oracle**.

8.4.1 Database Transaction

A **database transaction** comprises a unit of work performed within a database management system (or similar system) against a database, and treated in a coherent and reliable way independent of other transactions. Transactions in a database environment have two main purposes:

1. To provide reliable units of work that allow correct recovery from failures and keep a database consistent even in cases of system failure, when execution stops (completely or partially) and many operations upon a database remain uncompleted, with unclear status.

2. To provide isolation between programs accessing a database concurrently. Without isolation the program's outcomes are possibly erroneous.

Notes

A database transaction, by definition, must be atomic, consistent, isolated and durable. Database practitioners often refer to these properties of database transactions using the acronym ACID.

Transactions provide an “all-or-nothing” proposition, stating that each work-unit performed in a database must either complete in its entirety or have no effect whatsoever. Further, the system must isolate each transaction from other transactions, results must conform to existing constraints in the database, and transactions that complete successfully must get written to durable storage.

8.5 Common Corporate DBMS

Additional types of software applications have been used in the past and may be still in use on older, legacy systems at various organizations around the world. However, these examples provide an overview of the most popular and most-widely used by IT departments. Some typical examples of DBMS include: Oracle, DB2, Microsoft Access, Microsoft SQL Server, PostgreSQL, MySQL, and FileMaker.

8.5.1 ORACLE

The **Oracle Database** (commonly referred to as Oracle RDBMS or simply as Oracle) is an object-relational database management system (ORDBMS) produced and marketed by Oracle Corporation.

Larry Ellison and his friends and former co-workers Bob Miner and Ed Oates started the consultancy Software Development Laboratories (SDL) in 1977. SDL developed the original version of the Oracle software. The name Oracle comes from the code-name of a CIA-funded project Ellison had worked on while previously employed by Ampex.

8.5.2 DB2

The **IBM DB2 Enterprise Server Edition** is a relational model database server developed by IBM. It primarily runs on UNIX (namely AIX), Linux, IBM i (formerly OS/400), z/OS and Windows servers. DB2 also powers the different IBM InfoSphere Warehouse editions. Alongside DB2 is another RDBMS: Informix, which was acquired by IBM in 2001.

8.5.3 Microsoft Access

Microsoft Office Access, previously known as **Microsoft Access**, is a relational database management system from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software-development tools. It is a member of the Microsoft Office suite of applications, included in the Professional and higher editions or sold separately. In mid-May 2010, the current version of Microsoft Access 2010 was released by Microsoft in Office 2010; Microsoft Office Access 2007 was the prior version.

Access stores data in its own format based on the Access Jet Database Engine. It can also import or link directly to data stored in other applications and databases.

Software developers and data architects can use Microsoft Access to develop application software, and “power users” can use it to build simple applications. Like other Office applications, Access is supported by Visual Basic for Applications, an object-oriented programming language that can reference a variety of objects including DAO (Data Access Objects), ActiveX Data Objects, and many other ActiveX components. Visual objects used in forms and reports expose their methods

Notes

and properties in the VBA programming environment, and VBA code modules may declare and call Windows operating-system functions.

8.5.4 Microsoft SQL Server

Microsoft SQL Server is a relational model database server produced by Microsoft. Its primary query languages are T-SQL and ANSI SQL.

Since parting ways, several revisions have been done independently. SQL Server 7.0 was a rewrite from the legacy Sybase code. It was succeeded by SQL Server 2000, which was the first edition to be launched in a variant for the IA-64 architecture.

8.5.5 PostgreSQL

PostgreSQL, often simply Postgres, is an object-relational database management system (ORDBMS). It is released under an MIT-style license and is thus free and open source software. As with many other open-source programs, PostgreSQL is not controlled by any single company – a global community of developers and companies develops the system.

The mixed-capitalization of the PostgreSQL name can confuse some people on first viewing. The several pronunciations of “SQL” can lead to this confusion. It is abbreviated as “**Postgres**”, its original name. Because of ubiquitous support for the SQL Standard amongst most relational databases, the community considered changing the name back to Postgres. However, the PostgreSQL Core Team announced in 2007 that the product would continue to use the name PostgreSQL. The name refers to the project’s origins as a “post-Ingres” database, the original authors having also developed the Ingres database. (The name Ingres was an abbreviation for INteractive Graphics REtrieval System.)

8.5.6 MySQL

MySQL is a relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases. It is named after developer Michael Widenius’ daughter, My. The SQL phrase stands for Structured Query Language.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

Free-software projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Some free software project examples: Joomla, WordPress, Mob, phpBB, Drupal and other software built on the LAMP software stack. MySQL is also used in many high-profile, large-scale World Wide Web products, including Wikipedia, Google (though not for searches) and Face book.

8.5.7 Filemaker

FileMaker Pro is a cross-platform relational database application from FileMaker Inc., formerly Claris, a subsidiary of Apple Inc. It integrates a database engine with a GUI-based interface, allowing users to modify the database by dragging new elements into layouts, screens, or forms. Current versions are: FileMaker Pro 11, FileMaker Pro 11 Advanced, FileMaker Pro 11 Server, FileMaker Pro 11 Server Advanced and FileMaker Go.

FileMaker evolved from a DOS application, but was then developed primarily for the Apple Macintosh. Since 1992 it has been available for Microsoft Windows as well as Mac OS, and can be used in a heterogeneous environment. It is available in desktop, server, iOS and web-delivery configurations.

8.6 Summary

- Database is a system intended to organise, store and retrieve large amounts of data easily.
- DBMS is a tool that is employed in the broad practice of managing databases.
- Distributed database management system (ODBMS) is collection of data which logically belong to the same system but are spread out over the sites of the computer network.
- A modelling language is a data modelling language to define the scheme of each database hosted in DBMS.
- Data structures optimized to deal with very large amount of data stored on a permanent data storage device.

8.7 Keywords

Analytical database: Analysts may do their work directly against a data warehouse or create a separate analytic database for Online Analytical Processing.

Data definition subsystem: It helps the user create and maintain the data dictionary and define the structure of the files in a database.

Data structure: Data structures optimized to deal with very large amounts of data stored on a permanent data storage device.

Data warehouse: Data warehouses archive modern data from operational databases and often from external sources such as market research firms.

Database: A database is a system intended to organize, store, and retrieve large amounts of data easily. It consists of an organized collection of data for one or more uses, typically in digital form.

Distributed database: These are databases of local work-groups and departments at regional offices, branch offices, manufacturing plants and other work sites.

End-user database: These databases consist of data developed by individual end-users.

Hypermedia databases: The World Wide Web can be thought of as a database, albeit one spread across millions of independent computing systems.

Microsoft access: It is a relational database management system from Microsoft that combines the relational Microsoft Jet Database Is.

Modeling language: A modeling language is a data modeling language to define the schema of each database hosted in the DBMS, according to the DBMS database model.

Object database models: In recent years, the object-oriented paradigm has been applied in areas such as engineering and spatial databases, telecommunications and in various scientific domains.

Operational database: These databases store detailed data about the operations of an organization.

Post-relational database models: Products offering a more general data model than the relational model are sometimes classified as post-relational.

The DBMS: It is a computer software program that is designed as the means of managing all databases that are currently installed on a system hard drive or network.

Notes



Lab Exercise

1. Draw relational model, related records are linked together with a “key”.
2. Prepare database for hospital management.

8.8 Self-Assessment Questions

1. _____ is the most commonly used today.
(a) Relational structure (b) Network structure
2. _____ was used in early mainframe DBMS. Records’.
(a) Object structure (b) Hierarchical structure
3. The multidimensional structure is similar to the relational model.
(a) True (b) False
4. DBMS, is a computer Hardware program.
(a) True (b) False
5. The network structure consists of more
(a) Complex relationships (b) Single relationship
(c) All of these (d) Double relationship
6. The relational database was born in
(a) 1970 (b) 1975
(c) 1980 (d) 1960

8.9 Review Questions

1. What is Database?
2. How many types of database?
3. Define the Data Definition Subsystem.
4. What is Data structure?
5. What is Microsoft Access?
6. Write the full form of DBMS.
7. What is Post-relational database models?
8. Describe working with Database.
9. What is Object database models?
10. Define the Modeling language .
11. Describe the common corporate DBMS .
12. How many rules for Database work?

Answers for Self-Assessment Questions

1. (a) 2. (b) 3. (a) 4. (b) 5. (a) 6. (a)

8.10 Further Reading



Books

Maran illustrated Computers Guided Tour, by Ruth Maran; Kelleigh Johnson, Publisher:
Course Technology PTR



Online link

http://www.webopedia.com/TERM/D/database_management_system_DBMS.html

Unit 9: Software Development

CONTENTS

Objectives

- 9.1 History of Programming
 - 9.1.1 Quality Requirements in Programming
 - 9.1.2 Readability of Source Code
 - 9.1.3 Algorithmic Complexity
 - 9.1.4 Methodologies
 - 9.1.5 Measuring Language Usage
 - 9.1.6 Debugging
 - 9.1.7 Programming Languages
 - 9.1.8 Paradigms
 - 9.1.9 Compiling or Interpreting
 - 9.1.10 Self-Modifying Programs
 - 9.1.11 Execution and Storage
 - 9.1.12 Functional Categories
- 9.2 Hardware/Software Interactions
 - 9.2.1 Software Interfaces
 - 9.2.2 Hardware Interfaces
- 9.3 Planning a Computer Program
 - 9.3.1 The Programming Process
- 9.4 Summary
- 9.5 Keywords
- 9.6 Self Assessment Questions
- 9.7 Review Questions
- 9.8 Further Reading

Objectives

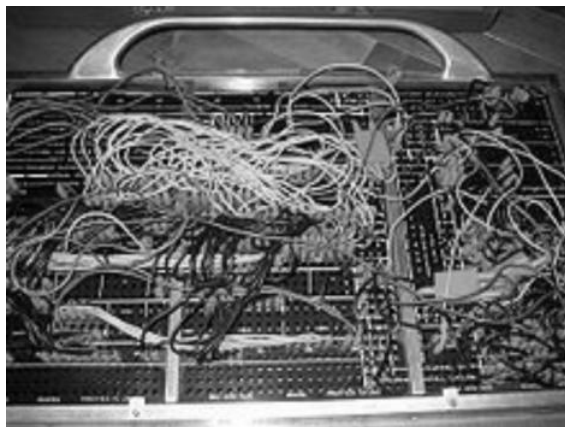
After studying this unit, you will be able to:

- Explain history of programming.
- Discuss Hardware/Software Interaction.
- Understand Planning Computer Program.

9.1 History of Programming

The Antikythera mechanism from ancient Greece was a calculator utilizing gears of various sizes and configuration to determine its operation which tracked the metonic cycle still used in lunar-to-solar calendars, and which is consistent for calculating the dates of the Olympiads. Al-Jazari built programmable Automata in 1206. One system employed in these devices was the use of pegs and cams placed into a wooden drum at specific locations which would sequentially trigger levers that in turn operated percussion instruments. The output of this device was a small drummer playing various rhythms and drum patterns. The Jacquard Loom, which Joseph Marie Jacquard developed in 1801, uses a series of pasteboard cards with holes punched in them. The hole pattern represented the pattern that the loom had to follow in weaving cloth. The loom could produce entirely different weaves using different sets of cards. Charles Babbage adopted the use of punched cards around 1830 to control his Analytical Engine. The synthesis of numerical calculation, predetermined operation and output, along with a way to organize and input instructions in a manner relatively easy for humans to conceive and produce, led to the modern development of computer programming. Development of computer programming accelerated through the Industrial Revolution.

Figure 9.1. Wired Plug Board for an IBM 402 Accounting Machine



In the late 1880s, Herman Hollerith invented the recording of data on a medium that could then be read by a machine. Prior uses of machine readable media, above, had been for control, not data. "After some initial trials with paper tape, he settled on punched cards". To process these punched cards, first known as "Hollerith cards" he invented the tabulator, and the keypunch machines. These three inventions were the foundation of the modern information processing industry. In 1896 he founded the Tabulating Machine Company (which later became the core of IBM). The addition of a control panel (plugboard) to his 1906 Type I Tabulator allowed it to do different jobs without having to be physically rebuilt. By the late 1940s, there were a variety of plug-board programmable machines, called unit record equipment, to perform data-processing tasks (card reading). Early computer programmers used plug-boards for the variety of complex calculations requested of the newly invented machines.

Notes

Figure 9.2: Data and Instructions could be Stored on External Punched Cards, Which were Kept in Order and Arranged in Program Decks



The invention of the von Neumann architecture allowed computer programs to be stored in computer memory. Early programs had to be painstakingly crafted using the instructions (elementary operations) of the particular machine, often in binary notation. Every model of computer would likely use different instructions (machine language) to do the same task. Later, assembly languages were developed that let the programmer specify each instruction in a text format, entering abbreviations for each operation code instead of a number and specifying addresses in symbolic form (e.g., ADD X, TOTAL). Entering a program in assembly language is usually more convenient, faster, and less prone to human error than using machine language, but because an assembly language is little more than a different notation for a machine language, any two machines with different instruction sets also have different assembly languages.

In 1954, FORTRAN was invented; it was the first high level programming language to have a functional implementation, as opposed to just a design on paper. (A high-level language is, in very general terms, any programming language that allows the programmer to write programs in terms that are more abstract than assembly language instructions, i.e. at a level of abstraction “higher” than that of an assembly language.) It allowed programmers to specify calculations by entering a formula directly (e.g. $Y = X^2 + 5 \cdot X + 9$). The program text, or source, is converted into machine instructions using a special program called a compiler, which translates the FORTRAN program into machine language. In fact, the name FORTRAN stands for “Formula Translation”. Many other languages were developed, including some for commercial programming, such as COBOL. Programs were mostly still entered using punched cards or paper tape. By the late 1960s, data storage devices and computer terminals became inexpensive enough that programs could be created by typing directly into the computers. Text editors were developed that allowed changes and corrections to be made much more easily than with punched cards.



Notes

Usually, an error in punching a card meant that the card had to be discarded and a new one punched to replace it.

As time has progressed, computers have made giant leaps in the area of processing power. This has brought about newer programming languages that are more abstracted from the underlying hardware. Although these high-level languages usually incur greater overhead, the increase in speed of modern computers has made the use of these languages much more practical than in the past. These increasingly abstracted languages typically are easier to learn and allow the programmer to develop applications much more efficiently and with less source code. However, high-level languages are still impractical for a few programs, such as those where low-level hardware control is necessary or where maximum processing speed is vital.

Throughout the second half of the twentieth century, programming was an attractive career in most developed countries. Some forms of programming have been increasingly subject to offshore outsourcing (importing software and services from other countries, usually at a lower wage), making programming career decisions in developed countries more complicated, while increasing economic opportunities in less developed areas. It is unclear how far this trend will continue and how deeply it will impact programmer wages and opportunities.

Notes

9.1.1 Quality Requirements in Programming

Whatever the approach to software development may be, the final program must satisfy some fundamental properties. The following properties are among the most relevant:

- (a) **Efficiency/performance:** the amount of system resources a program consumes (processor time, memory space, slow devices such as disks, network bandwidth and to some extent even user interaction): the less, the better. This also includes correct disposal of some resources, such as cleaning up temporary files and lack of memory leaks.
- (b) **Reliability:** how often the results of a program are correct. This depends on conceptual correctness of algorithms, and minimization of programming mistakes, such as mistakes in resource management (e.g., buffer overflows and race conditions) and logic errors (such as division by zero or off-by-one errors).
- (c) **Robustness:** how well a program anticipates problems not due to programmer error. This includes situations such as incorrect, inappropriate or corrupt data, unavailability of needed resources such as memory, operating system services and network connections, and user error.
- (d) **Usability:** the ergonomics of a program: the ease with which a person can use the program for its intended purpose or in some cases even unanticipated purposes. Such issues can make or break its success even regardless of other issues. This involves a wide range of textual, graphical and sometimes hardware elements that improve the clarity, intuitiveness, cohesiveness and completeness of a program's user interface.
- (e) **Portability:** the range of computer hardware and operating system platforms on which the source code of a program can be compiled/interpreted and run. This depends on differences in the programming facilities provided by the different platforms, including hardware and operating system resources, expected behaviour of the hardware and operating system, and availability of platform specific compilers (and sometimes libraries) for the language of the source code.

Notes

- (f) **Maintainability:** The ease with which a program can be modified by its present or future developers in order to make improvements or customizations, fix bugs and security holes, or adapt it to new environments. Good practices during initial development make the difference in this regard. This quality may not be directly apparent to the end user but it can significantly affect the fate of a program over the long term.

9.1.2 Readability of Source Code

In computer programming, readability refers to the ease with which a human reader can comprehend the purpose, control flow, and operation of source code. It affects the aspects of quality above, including portability, usability and most importantly maintainability.

Readability is important because programmers spend the majority of their time reading, trying to understand and modifying existing source code, rather than writing new source code. Unreadable code often leads to bugs, inefficiencies, and duplicated code. A study found that a few simple readability transformations made code shorter and drastically reduced the time to understand it.

Following a consistent programming style often helps readability. However, readability is more than just programming style. Many factors, having little or nothing to do with the ability of the computer to efficiently compile and execute the code, contribute to readability. Some of these factors include:

- (i) Different indentation styles (whitespace)
- (ii) Comments
- (iii) Decomposition
- (iv) Naming conventions for objects (such as variables, classes, procedures, etc.)

9.1.3 Algorithmic Complexity

The academic field and the engineering practice of computer programming are both largely concerned with discovering and implementing the most efficient algorithms for a given class of problem. For this purpose, algorithms are classified into orders using so-called Big O notation, $O(n)$, which expresses resource use, such as execution time or memory consumption, in terms of the size of an input. Expert programmers are familiar with a variety of well-established algorithms and their respective complexities and use this knowledge to choose algorithms that are best suited to the circumstances.

9.1.4 Methodologies

The first step in most formal software development projects is requirements analysis, followed by testing to determine value modeling, implementation, and failure elimination (debugging). There exist a lot of differing approaches for each of those tasks. One approach popular for requirements analysis is Use Case analysis. Nowadays many programmers use forms of Agile software development where the various stages of formal software development are more integrated together into short cycles that take a few weeks rather than years. There are many approaches to the Software development process.

Popular modeling techniques include Object-Oriented Analysis and Design (OOAD) and Model-Driven Architecture (MDA). The Unified Modeling Language (UML) is a notation used for both the OOAD and MDA.

A similar technique used for database design is Entity-Relationship Modeling (ER Modeling).

Implementation techniques include imperative languages (object-oriented or procedural), functional languages, and logic languages.

9.1.5 Measuring Language Usage

It is very difficult to determine what are the most popular of modern programming languages. Some languages are very popular for particular kinds of applications (e.g., COBOL is still strong in the corporate data center, often on large mainframes, FORTRAN in engineering applications, scripting languages in web development, and C in embedded applications), while some languages are regularly used to write many different kinds of applications. Also many applications use a mix of several languages in their construction and use.

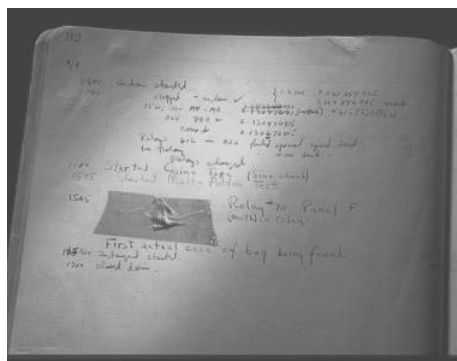
Methods of measuring programming language popularity include: counting the number of job advertisements that mention the language, the number of books teaching the language that are sold (this overestimates the importance of newer languages), and estimates of the number of existing lines of code written in the language (this underestimates the number of users of business languages such as COBOL).

9.1.6 Debugging

Debugging is a very important task in the software development process, because an incorrect program can have significant consequences for its users. Some languages are more prone to some kinds of faults because their specification does not require compilers to perform as much checking as other languages. Use of a static analysis tool can help detect some possible problems.

Debugging is often done with IDEs like Eclipse, Kdevelop, NetBeans, Code, Blocks, and Visual Studio. Standalone debuggers like gdb are also used, and these often provide less of a visual environment, usually using a command line.

Figure 9.3: A Bug, which was Debugged in 1947



9.1.7 Programming Languages

Different programming languages support different styles of programming (called programming paradigms). The choice of language used is subject to many considerations, such as company policy, suitability to task, availability of third-party packages, or individual preference. Ideally, the programming language best suited for the task at hand will be selected. Trade-offs from this ideal involve finding enough programmers who know the language to build a team, the availability of compilers for that language, and the efficiency with which programs written in a given language execute. Languages form an approximate spectrum from “low-level” to “high-level”; “low-level” languages are typically more machine-oriented and faster to execute, whereas “high-level” languages are more abstract and easier to use but execute less quickly. It is usually easier to code in “high-level” languages than in “low-level” ones.

The details look different in different languages, but a few basic instructions appear in just about every language:

Notes

- **input:** Get data from the keyboard, a file, or some other device.
- **output:** Display data on the screen or send data to a file or other device.
- **arithmetic:** Perform basic arithmetical operations like addition and multiplication.
- **conditional execution:** Check for certain conditions and execute the appropriate sequence of statements.
- **repetition:** Perform some action repeatedly, usually with some variation.

Many computer languages provide a mechanism to call functions provided by libraries such as in .dlls. Provided the functions in a library follow the appropriate run time conventions (e.g., method of passing arguments), then these functions may be written in any other language.

9.1.8 Paradigms

Computer programs can be categorized by the programming language paradigm used to produce them. Two of the main paradigms are imperative and declarative.

Programs written using an imperative language specify an algorithm using declarations, expressions, and statements. A declaration couples a variable name to a datatype. For example: `var x: integer;` . An expression yields a value. For example: `2 + 2` yields 4. Finally, a statement might assign an expression to a variable or use the value of a variable to alter the program's control flow. For example: `x := 2 + 2;` if `x = 4` then `do_something()` ; One criticism of imperative languages is the side effect of an assignment statement on a class of variables called non-local variables.

Programs written using a declarative language specify the properties that have to be met by the output. They do not specify details expressed in terms of the control flow of the executing machine but of the mathematical relations between the declared objects and their properties. Two broad categories of declarative languages are functional languages and logical languages. The principle behind functional languages (like Haskell) is to not allow side effects, which makes it easier to reason about programs like mathematical functions. The principle behind logical languages (like Prolog) is to define the problem to be solved, the goal and leave the detailed solution to the Prolog system itself. The goal is defined by providing a list of subgoals. Then each subgoal is defined by further providing a list of its subgoals, etc. If a path of subgoals fails to find a solution, then that subgoal is backtracked and another path is systematically attempted.

The form in which a program is created may be textual or visual. In a visual language program, elements are graphically manipulated rather than textually specified.

9.1.9 Compiling or Interpreting

A computer program in the form of a human-readable, computer programming language is called source code. Source code may be converted into an executable image by a compiler or executed immediately with the aid of an interpreter.

Either compiled or interpreted programs might be executed in a batch process without human interaction, but interpreted programs allow a user to type commands in an interactive session. In this case, the programs are the separate commands, whose execution occurs sequentially, and thus together. When a language is used to give commands to a software application (such as a shell) it is called a scripting language.



Compilers are used to translate source code from a programming language into either object code or machine code. Object code needs further processing to become machine code, and machine code is the central processing unit's native code, ready for execution.

Interpreted computer programs -in a batch or interactive session— are either decoded and then immediately executed or are decoded into some efficient intermediate representation for future execution. BASIC, Perl, and Python are examples of immediately executed computer programs. Alternatively, Java computer programs are compiled ahead of time and stored as a machine independent code called bytecode. Bytecode is then executed on request by an interpreter called a virtual machine.

The main disadvantage of interpreters is that computer programs run slower than when compiled. Interpreting code is slower than running the compiled version because the interpreter must decode each statement each time it is loaded and then perform the desired action. However, software development may be faster using an interpreter because testing is immediate when the compiling step is omitted. Another disadvantage of interpreters is that at least one must be present on the computer during computer program execution. By contrast, compiled computer programs need no compiler present during execution.

No properties of a programming language require it to be exclusively compiled or exclusively interpreted. The categorization usually reflects the most popular method of language execution. For example, BASIC is thought of as an interpreted language and C a compiled language, despite the existence of BASIC compilers and C interpreters. Some systems use just-in-time compilation (JIT) whereby sections of the source are compiled 'on the fly' and stored for subsequent executions.

Notes

9.1.10 Self-Modifying Programs

A computer program in execution is normally treated as being different from the data the program operates on. However, in some cases this distinction is blurred when a computer program modifies itself. The modified computer program is subsequently executed as part of the same program. Self-modifying code is possible for programs written in machine code, assembly language, Lisp, C, COBOL, PL/1, Prolog and JavaScript (the eval feature) among others.

9.1.11 Execution and Storage

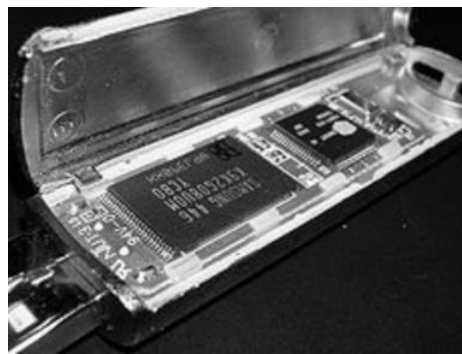
Typically, computer programs are stored in non-volatile memory until requested either directly or indirectly to be executed by the computer user. Upon such a request, the program is loaded into random access memory, by a computer program called an operating system, where it can be accessed directly by the central processor. The central processor then executes ("runs") the program, instruction by instruction, until termination. A program in execution is called a process. Termination is either by normal self-termination or by error, software or hardware error.

Notes

9.1.11.1 Embedded Programs

Some computer programs are embedded into hardware. A stored-program computer requires an initial computer program stored in its read-only memory to boot. The boot process is to identify and initialize all aspects of the system, from processor registers to device controllers to memory contents. Following the initialization process, this initial computer program loads the operating system and sets the program counter to begin normal operations. Independent of the host computer, a hardware device might have embedded firmware to control its operation. Firmware is used when the computer program is rarely or never expected to change, or when the program must not be lost when the power is off.

Figure 9.4: The Microcontroller on the Right of this USB Flash Drive is Controlled with Embedded Firmware

**9.1.11.2 Manual Programming**

Computer programs historically were manually input to the central processor via switches. An instruction was represented by a configuration of on/off settings. After setting the configuration, an execute button was pressed. This process was then repeated. Computer programs also historically were manually input via paper tape or punched cards. After the medium was loaded, the starting address was set via switches and the execute button pressed.

Figure 9.5: Switches for Manual Input on a Data General Nova 3

**9.1.11.3 Automatic Program Generation**

Generative programming is a style of computer programming that creates source code through generic classes, prototypes, templates, aspects, and code generators to improve programmer productivity. Source code is generated with programming tools such as a template processor or an integrated development environment. The simplest form of source code generator is a macro processor, such as the C preprocessor, which replaces patterns in source code according to relatively simple rules.

Software engines output source code or markup code that simultaneously become the input to another computer process. The analogy is that of one process driving another process, with the computer code being burned as fuel.



Application servers are software engines that deliver applications to client computers. For example, a Wiki is an application server that lets users build dynamic content assembled from articles. Wikis generate HTML, CSS, Java, and JavaScript which are then interpreted by a web browser.

9.1.11.4 Simultaneous Execution

Many operating systems support multitasking which enables many computer programs to appear to run simultaneously on one computer. Operating systems may run multiple programs through process scheduling a software mechanism to switch the CPU among processes often so users can interact with each program while it runs. Within hardware, modern day multiprocessor computers or computers with multicore processors may run multiple programs.

One computer program can calculate simultaneously more than one operation using threads or separate processes. Multithreading processors are optimized to execute multiple threads efficiently.

9.1.12 Functional Categories

Computer programs may be categorized along functional lines. The main functional categories are system software and application software. System software includes the operating system which couples computer hardware with application software. The purpose of the operating system is to provide an environment in which application software executes in a convenient and efficient manner. In addition to the operating system, system software includes utility programs that help manage and tune the computer. If a computer program is not system software then it is application software. Application software includes middleware, which couples the system software with the user interface. Application software also includes utility programs that help users solve application problems, like the need for sorting.

Sometimes development environments for software development are seen as a functional category on its own, especially in the context of human-computer interaction and programming language design. Development environments gather system software (such as compilers and system's batch processing scripting languages) and application software (such as IDEs) for the specific purpose of helping programmers create new programs.

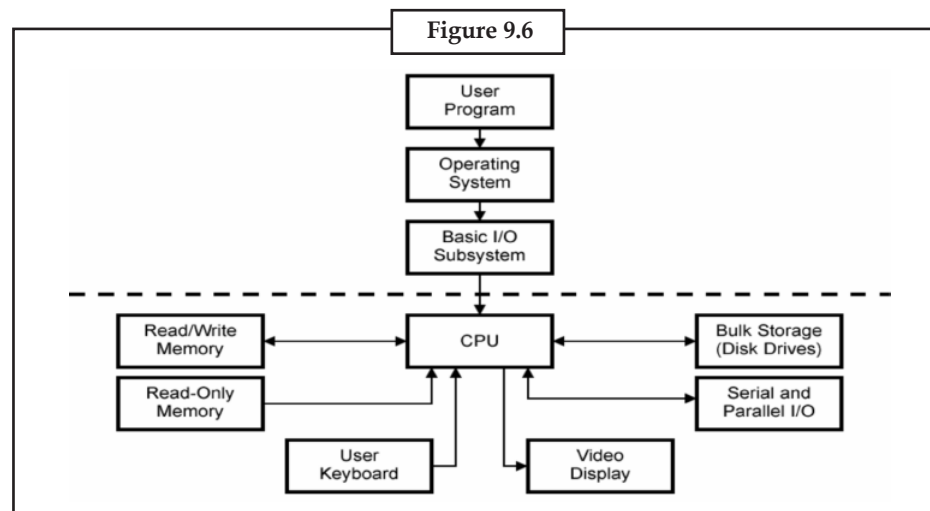
9.2 Hardware/Software Interactions

In the field of computer science, an **interface** refers to a point of interaction between components, and is applicable at the level of both hardware and software. This allows a component, whether a piece of hardware such as a graphics card or a piece of software such as an internet browser, to function independently while using interfaces to communicate with other components via an input/output system and an associated protocol.

In addition to hardware and software interfaces, a computing interface may refer to the means of communication between the computer and the user by means of peripheral devices such a monitor or a keyboard, an interface with the internet via Internet Protocol, and any other point of communication involving a computer.

Most modern computer systems are built as a series of layers, or levels. The lowest level is usually considered the physical "hardware" layer and the topmost layer is usually a user application or source code.

Notes



9.2.1 Software Interfaces

A software interface may refer to a range of different types of interface at different “levels”: an operating system may interface with pieces of hardware, applications or programs running on the operating system may need to interact via streams, and in object oriented programs, objects within an application may need to interact via methods.

9.2.1.1 Software Interfaces in Practice

A piece of software provides access to computer resources (such as memory, CPU, storage, etc.) by its underlying computer system; the availability of these resources to other software can have major ramifications sometimes disastrous ones for its functionality and stability. A key principle of design is to prohibit access to all resources by default, allowing access only through well-defined entry points, i.e. interfaces.

The types of access that interfaces provide between software components can include: constants, data types, types of procedures, exception specifications and method signatures. In some instances, it may be useful to define public variables as part of the interface. It often also specifies the functionality of those procedures and methods, either by comments or (in some experimental languages) by formal logical assertions and preconditions.

The interface of a software module A is deliberately kept separate from the implementation of that module. The latter contains the actual code of the procedures and methods described in the interface, as well as other “private” variables, procedures, etc.. Any other software module B (which can be referred to as a client to A) that interacts with A is forced to do so only through the interface. One practical advantage of this arrangement is that replacing the implementation of A by another one that meets the same specifications of the interface should not cause B to fail-as long as its use of A complies with the specifications of the interface.

9.2.1.2 Software Interfaces in Object Oriented Languages

In object-oriented languages the term “interface” is often used to define an abstract type that contains no data but exposes behaviors defined as methods. A class having all the methods corresponding to that interface is said to implement that interface. Furthermore, a class can implement multiple interfaces, and hence can be of different types at the same time.

An interface is hence a type definition; anywhere an object can be exchanged (in a function or method call) the type of the object to be exchanged can be defined in terms of an interface instead

of a specific class. This allows later code to use the same function exchanging different object types; hence such code turns out to be more generic and reusable.

Usually a method in an interface cannot be used directly; there must be a class implementing that object to be used for the method invocation. For example, one can define an interface called “Stack” that has two methods: `push()` and `pop()` and later implement it in two different versions, say, `FastStack` and `GenericStack`-the first being faster but working with a stack of fixed size, and the second using a data structure that can be resized but at the cost of somewhat lower speed.

This approach can be pushed to the limit of defining interfaces with a single method; e.g. the Java language defines the interface `Readable` that has the single `read()` method and a collection of implementations to be used for different purposes, among others: `BufferedReader`, `FileReader`, `InputStreamReader`, `PipedReader`, and `StringReader`.



Did u know?

In its purest form, an interface (like in Java) must include only method definitions and constant values that make up part of the static interface of a type. Some languages (like C#) also permit the definition to include properties owned by the object, which are treated as methods with syntactic sugar.

9.2.1.3 Programming against Software Interfaces

The use of interfaces allows implementation of a programming style called programming against interfaces. The idea behind this is to base the logic one develops on the sole interface definition of the objects one uses and not to make the code depend on the internal details. This allows the programmer the ability to later change the behavior of the system by simply swapping the object used with another implementing the same interface.

Pushing this idea to the limit one can introduce the inversion of control which means leaving the context to inject the code with the specific implementations of the interface that will be used to perform the work.

9.2.2 Hardware Interfaces

Hardware interfaces exist in computing systems between many of the components such as the various buses, storage devices, other I/O devices, etc. A hardware interface is described by the mechanical, electrical and logical signals at the interface and the protocol for sequencing them (sometimes called signaling). A standard interface, such as SCSI, decouples the design and introduction of computing hardware, such as I/O devices, from the design and introduction of other components of a computing system, thereby allowing users and manufacturers great flexibility in the implementation of computing systems.

9.3 Planning a Computer Program

9.3.1 The Programming Process

Developing a program involves steps similar to any problem-solving task. There are five main ingredients in the programming process:

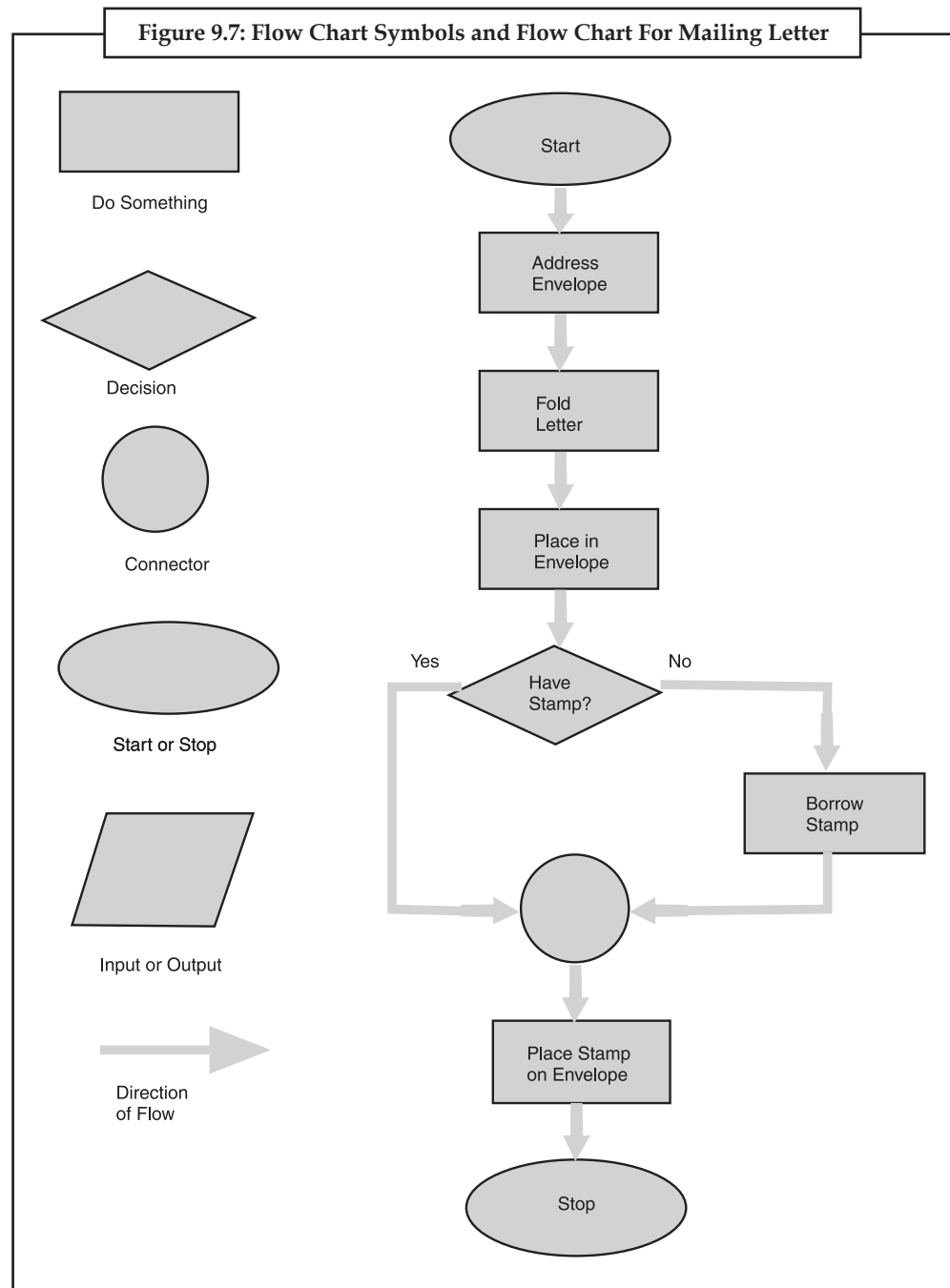
1. Defining the problem
2. Planning the solution
3. Coding the program
4. Testing the program
5. Documenting the program

Let us discuss each of these in turn.

Notes

9.3.1.1 Defining the Problem

Suppose that, as a programmer, you are contacted because your services are needed. You meet with users from the client organization to analyze the problem, or you meet with a systems analyst who outlines the project. Specifically, the task of defining the problem consists of identifying what it is you know (input-given data), and what it is you want to obtain (output-the result). Eventually, you produce a written agreement that, among other things, specifies the kind of input, processing, and output required. This is not a simple process.

9.3.1.2 Planning the Solution

**Task**

Using the above flow chart draw a flow chart for adding two numbers and find the output if it is odd or even for different cases.

Two common ways of planning the solution to a problem are to draw a flowchart and to write pseudocode, or possibly both. Essentially, a flowchart is a pictorial representation of a step-by-step solution to a problem. It consists of arrows representing the direction the program takes and boxes and other symbols representing actions. It is a map of what your program is going to do and how it is going to do it. The American National Standards Institute (ANSI) has developed a standard set of flowchart symbols. Figure 1 shows the symbols and how they might be used in a simple flowchart of a common everyday act-preparing a letter for mailing.

Pseudocode is an English-like nonstandard language that lets you state your solution with more precision than you can in plain English but with less precision than is required when using a formal programming language. Pseudocode permits you to focus on the program logic without having to be concerned just yet about the precise syntax of a particular programming language. However, pseudocode is not executable on the computer. We will illustrate these later in this chapter, when we focus on language examples.

9.3.1.3 Coding the Program

As the programmer, your next step is to code the program-that is, to express your solution in a programming language. You will translate the logic from the flowchart or pseudocode-or some other tool-to a programming language. As we have already noted, a programming language is a set of rules that provides a way of instructing the computer what operations to perform. There are many programming languages: BASIC, COBOL, Pascal, FORTRAN, and C are some examples. You may find yourself working with one or more of these. We will discuss the different types of languages in detail later in this chapter.

Although programming languages operate grammatically, somewhat like the English language, they are much more precise. To get your program to work, you have to follow exactly the rules-the syntax-of the language you are using. Of course, using the language correctly is no guarantee that your program will work, any more than speaking grammatically correct English means you know what you are talking about. The point is that correct use of the language is the required first step. Then your coded program must be keyed, probably using a terminal or personal computer, in a form the computer can understand.

One more note here: Programmers usually use a text editor, which is somewhat like a word processing program, to create a file that contains the program. However, as a beginner, you will probably want to write your program code on paper first.

9.3.1.4 Testing the Program

Some experts insist that a well-designed program can be written correctly the first time. In fact, they assert that there are mathematical ways to prove that a program is correct. However, the imperfections of the world are still with us, so most programmers get used to the idea that their newly written programs probably have a few errors. This is a bit discouraging at first, since programmers tend to be precise, careful, detail-oriented people who take pride in their work. Still, there are many opportunities to introduce mistakes into programs, and you, just as those who have gone before you, will probably find several of them.

Notes

Eventually, after coding the program, you must prepare to test it on the computer. This step involves these phases:

- (i) **Desk-checking.** This phase, similar to proofreading, is sometimes avoided by the programmer who is looking for a shortcut and is eager to run the program on the computer once it is written. However, with careful desk-checking you may discover several errors and possibly save yourself time in the long run. In desk-checking you simply sit down and mentally trace, or check, the logic of the program to attempt to ensure that it is error-free and workable. Many organizations take this phase a step further with a walkthrough, a process in which a group of programmers-your peers-review your program and offer suggestions in a collegial way.
- (ii) **Translating.** A translator is a program that (1) checks the syntax of your program to make sure the programming language was used correctly, giving you all the syntax-error messages, called diagnostics, and (2) then translates your program into a form the computer can understand. A by-product of the process is that the translator tells you if you have improperly used the programming language in some way. These types of mistakes are called syntax errors. The translator produces descriptive error messages. For instance, if in FORTRAN you mistakenly write $N=2*(I+J))$ -which has two closing parentheses instead of one-you will get a message that says, "UNMATCHED PARENTHESES." (Different translators may provide different wording for error messages.) Programs are most commonly translated by a compiler. A compiler translates your entire program at one time. The translation involves your original program, called a source module, which is transformed by a compiler into an object module. Prewritten programs from a system library may be added during the link/load phase, which results in a load module. The load module can then be executed by the computer.
- (iii) **Debugging.** A term used extensively in programming, debugging means detecting, locating, and correcting bugs (mistakes), usually by running the program. These bugs are logic errors, such as telling a computer to repeat an operation but not telling it how to stop repeating. In this phase you run the program using test data that you devise. You must plan the test data carefully to make sure you test every part of the program.

9.3.1.5 Documenting the Program

Documenting is an ongoing, necessary process, although, as many programmers are, you may be eager to pursue more exciting computer-centered activities. Documentation is a written detailed description of the programming cycle and specific facts about the program. Typical program documentation materials include the origin and nature of the problem, a brief narrative description of the program, logic tools such as flowcharts and pseudocode, data-record descriptions, program listings, and testing results. Comments in the program itself are also considered an essential part of documentation. Many programmers document as they code. In a broader sense, program documentation can be part of the documentation for an entire system.

The wise programmer continues to document the program throughout its design, development, and testing. Documentation is needed to supplement human memory and to help organize program planning. Also, documentation is critical to communicate with others who have an interest in the program, especially other programmers who may be part of a programming team. And, since turnover is high in the computer industry, written documentation is needed so that those who come after you can make any necessary modifications in the program or track down any errors that you missed.



Case Study

Apple Computer

Apple Computer is an American Multinational corporation with a focus on designing and manufacturing consumer electronics and even develops software products. It was co-founded by Steve Wozniak and Steve Jobs. Steve Wozniak met Steve Jobs while he was working at Hewlett-Packard. Steve Jobs worked part time, where he would finish up games that they designed in Grass Valley.

In 1975, the first personal computer kit, the Alistair 8800 was announced. Since Steve Wozniak couldn't afford an Alistair 8800 he decided to build his own personal computer by using cheaper chips. As circuit board alone, it could do more than Alistair. He and Steve Jobs called it Apple I, Jobs handled on marketing it while Wozniak continued to improve it. By 1977, Wozniak had built Apple II, then he and Jobs decided to form Apple Computer Inc. when it went public on 1980, its stock value was \$117 million, three years later it was \$985 million.

This story of how Apple started is one of my favorite. We watched a film titled "Pirates of the Silicon Valley" last year that's why Apple story is a little familiar for me. We can see that Steve Wozniak is determined that he will develop a computer even if he was still a child. I can say that if we really put out hard work and commitment to do something then we can be successful. According to Wozniak, if we try to start our own company, we must have the highest ethics and be open and truthful about things, not hide. We must not lead people. Know in your heart that you're a good person with good goals because it will carry over to our own self-confidence: make our own product better than the average person would.

Questions:

1. Mention some attributes of Apple computers in software development.
2. What is the vision behind establishing Apple computers?

9.4 Summary

- Debugging is often done with IDE like Eclipse, kddevelop, net bank and visual studio.
- Implementation techniques include imperative languages (object-oriented or procedural) functional languages, and logic languages.
- Computer programs can be categorized by the programming language paradigms used to produce them. Two of the main paradigms are imperative and declarative.
- Compilers are used to translate source code from a programming language into either object code or machine code.
- Computer programs are stored in non-volatile memory until requested either directly or indirectly to be executed by the computer user.

9.5 Keywords

Compiler: A **compiler** is a computer program (or set of programs) that transforms source code written in a programming language (the source language) into another computer language (the target language, often having a binary form known as object code).

Computer programming: Computer programming is the process of designing, writing, testing, debugging / troubleshooting, and maintaining the source code computer programs.

Debugging: Debugging is a methodical process of finding and reducing the number of bugs, or defects, in a computer program or a piece of electronic hardware, thus making it behave as expected.

Hardware interfaces: A hardware interface is described by the mechanical, electrical and logical signals at the interface and the protocol for sequencing them (sometimes called signaling).

Paradigms: A programming paradigm is a fundamental style of computer programming. (Compare with a methodology, which is a style of solving specific software engineering problems.)



1. What is the difference in memory management between Java and C++?
Lab Exercise 2. Is it possible to create a memory leak in Java?

9.6 Self-Assessment Questions

1. Errors in computer results could be due to
 - (a) Encoding of data
 - (b) Transmission of data
 - (c) Manipulation of data
 - (d) All of the above
2. Defect prevention is defined as:
 - (a) Finding and fixing errors after insertion
 - (b) Finding and fixing errors before release but after insertion
 - (c) Finding and fixing errors after release
 - (d) Avoiding defect insertion
3. Product quality is defined as:
 - (a) Delivering a product with correct requirements
 - (b) Delivering a product using correct development procedures
 - (c) Delivering a product which is developed iteratively
 - (d) Delivering a product using high quality procedures
4. Transistors are associated with which computer system?
 - (a) First generation
 - (b) Fifth generation
 - (c) Second generation
 - (d) None of these

5. Charles Babbage invented

Notes

- | | |
|-------------------------|-----------------------|
| (a) ENIAC | (b) Difference engine |
| (c) Electronic computer | (d) Punched card |

6. MS-DOS is the name of a/an

- | | |
|--------------------------|-------------------|
| (a) Application software | (b) Hardware |
| (c) System software | (d) None of these |

7. The retrieval of information from the computer is defined as

- | | |
|------------------------|-------------------------------|
| (a) Collection of data | (b) Data retrieval operations |
| (c) Output | (d) Data output collection |

9.7 Review Questions

1. What is meant by readability of source code?
2. List the basic instructions which appear in programming languages.
3. How many types of paradigms are used in computer program? Explain them.
4. How programs are executed and stored?
5. What do you mean by software interfaces?
6. Explain the planning process.
7. What is meant by machine language?
8. What do you know about FORTRAN?

Answers for Self-Assessment Questions

- | | | | | |
|--------|--------|--------|--------|--------|
| 1. (d) | 2. (d) | 3. (a) | 4. (c) | 5. (b) |
| 6. (c) | 7. (c) | | | |

9.8 Further Reading



Books

Computing Fundamentals by Peter Norton



Online link

<http://www.methodsandtools.com/archive/>

Unit 10: Programming Language

CONTENTS

Objectives

10.1 Basic of Programming

10.1.1 Why Programming?

10.1.2 What Programmers Do?

10.2 Levels of Language in Computer Programming

10.2.1 Machine Language

10.2.2 Assembly Languages

10.2.3 High-Level Languages

10.2.4 Very High-Level Languages

10.2.5 Query Languages

10.2.6 Natural Languages

10.2.7 Choosing a Language

10.3 Summary

10.4 Keywords

10.5 Self-Assessment Questions

10.6 Review Questions

10.7 Further Reading

Objectives

After studying this unit, you will be able to:

- Understand Basic of Programming.
- Explain Levels of Language in Computer Programming.

10.1 Basic of Programming

Computer programming (often shortened to programming or coding) is the process of designing, writing, testing, debugging / troubleshooting, and maintaining the source code of computer programs. This source code is written in a programming language. The purpose of programming is to create a program that exhibits a certain desired behaviour. The process of writing source code often requires expertise in many different subjects, including knowledge of the application domain, specialized algorithms and formal logic.

Computer programming is the process of transforming a mental plan in familiar terms into one compatible with the computer. Said another way, programming is the craft of transforming requirements into something that a computer can execute.

10.1.1 Why Programming?

You may already have used software, perhaps for word processing or spreadsheets, to solve problems. Perhaps now you are curious to learn how programmers write software. A program is a set of step-by-step instructions that directs the computer to do the tasks you want it to do and produce the results you want.

There are at least three good reasons for learning programming:

- (a) Programming helps you understand computers. The computer is only a tool. If you learn how to write simple programs, you will gain more knowledge about how a computer works.
- (b) Writing a few simple programs increases your confidence level. Many people find great personal satisfaction in creating a set of instructions that solve a problem.
- (c) Learning programming lets you find out quickly whether you like programming and whether you have the analytical turn of mind programmers need. Even if you decide that programming is not for you, understanding the process certainly will increase your appreciation of what programmers and computers can do.

A set of rules that provides a way of telling a computer what operations to perform is called a programming language. There is not, however, just one programming language; there are many. In this chapter you will learn about controlling a computer through the process of programming. You may even discover that you might want to become a programmer.

An important point before we proceed: You will not be a programmer when you finish reading this chapter or even when you finish reading the final chapter. Programming proficiency takes practice and training beyond the scope of this book. However, you will become acquainted with how programmers develop solutions to a variety of problems.

10.1.2 What Programmers Do?

In general, the programmer's job is to convert problem solutions into instructions for the computer. That is, the programmer prepares the instructions of a computer program and runs those instructions on the computer, tests the program to see if it is working properly, and makes corrections to the program. The programmer also writes a report on the program. These activities are all done for the purpose of helping a user fill a need, such as paying employees, billing customers, or admitting students to college.

The programming activities just described could be done, perhaps, as solo activities, but a programmer typically interacts with a variety of people. For example, if a program is part of a system of several programs, the programmer coordinates with other programmers to make sure that the programs fit together well. If you were a programmer, you might also have coordination meetings with users, managers, systems analysts, and with peers who evaluate your work-just as you evaluate theirs.

10.2 Levels of Language in Computer Programming

Programming languages are said to be "lower" or "higher," depending on how close they are to the language the computer itself uses (0s and 1s = low) or to the language people use (more English-like-high). We will consider five levels of language. They are numbered 1 through 5 to

Notes

correspond to levels, or generations. In terms of ease of use and capabilities, each generation is an improvement over its predecessors. The five generations of languages are:

- (a) Machine language
- (b) Assembly languages
- (c) High-level languages
- (d) Very high-level languages
- (e) Natural languages

Let us look at each of these categories.

10.2.1 Machine Language

Humans do not like to deal in numbers alone—they prefer letters and words. But, strictly speaking, numbers are what machine language is. This lowest level of language, machine language, represents data and program instructions as 1s and 0s—binary digits corresponding to the on and off electrical states in the computer. Each type of computer has its own machine language. In the early days of computing, programmers had rudimentary systems for combining numbers to represent instructions such as add and compare. Primitive by today's standards, the programs were not convenient for people to read and use. The computer industry quickly moved to develop assembly languages.

10.2.2 Assembly Languages

Figure 10.1: Example Assembly Language Program

PROG8	PRINT	NOGEN	
CARDFIL	START	0	
	DFTCD	DEVADDR=SYSRDR, RECFORM=FIXUNB, IOAREA1=CARDREC, C	
		TYPEFILE=INPUT, BLKSIZE=80, EOFADDR=FINISH	
REPTFIL	DTFPR	DEVADDR=SYSLSLST, IOAREA1=PRNTREC, BLKSIZE=132	
BEGIN	BALR	3, 0	REGISTER 3 IS BASE REGISTER
	USING	*, 3	
	OPEN	CARDFIL, REPTFIL	OPEN FILES
	MVC	PRNTREC, SPACES	MOVE SPACES TO OUTPUT RECORD
READLOOP	GET	CARDFIL	READ A RECORD
	MVC	OFIRST, IFIRST	MOVE ALL INPUT FIELDS
	MVC	OLAST, ILAST	TO OUTPUT RECORD FIELDS
	MVC	OADDR, IADDR	
	MVC	OCITY, ICITY	
	MVC	OSTATE, ISTATE	
	MVC	OZIP, IZIP	
	PUT	REPTFIL	WRITE THE RECORD
	B	READLOOP	BRANCH TO READ AGAIN
FINISH	CLOSE	CARDFIL, REPTFIL	CLOSE FILES
	EOJ		END OF JOB
CARDREC	DS	OCL80	DESCRIPTION OF INPUT RECORD
IFIRST	DS	CL10	
ILAST	DS	CL10	
IADDR	DS	CL30	
ICITY	DS	CL20	
ISTATE	DS	CL2	
IZIP	DS	CL5	
	DS	CL3	
PRNTREC	DS	OCL32	DESCRIPTION OF OUTPUT RECORD
	DS	CL10	
OLAST	DS	CL10	
	DS	CL5	
OFIRST	DS	CL10	
	DS	CL15	
OADDR	DS	CL30	
	DS	CL15	
OCITY	DS	CL20	
	DS	CL5	
OSTATE	DS	CL2	
	DS	CL5	
OZIP	DS	CL5	
SPACES	DC	CL132''	
	END	BEGIN	

Today, assembly languages are considered very low level-that is, they are not as convenient for people to use as more recent languages. At the time they were developed, however, they were considered a great leap forward. To replace the Is and Os used in machine language, assembly languages use mnemonic codes, abbreviations that are easy to remember: A for Add, C for Compare, MP for Multiply, STO for storing information in memory, and so on. Although these codes are not English words, they are still- from the standpoint of human convenience-preferable to numbers (Os and 1s) alone. Furthermore, assembly languages permit the use of names- perhaps RATE or TOTAL-for memory locations instead of actual address numbers just like machine language, each type of computer has its own assembly language.

The programmer who uses an assembly language requires a translator to convert the assembly language program into machine language. The translator is an assembler program, also referred to as an assembler. It takes the programs written in assembly language and turns them into machine language. Programmers need not worry about the translating aspect; they need only write programs in assembly language. The translation is taken care of by the assembler.



A translator is needed because machine language is the only language the computer can actually execute.

Although assembly languages represent a step forward, they still have many disadvantages. A key disadvantage is that assembly language is detailed in the extreme, making assembly programming repetitive, tedious, and error prone. This drawback is apparent in the program in Figure 7.7. Assembly language may be easier to read than machine language, but it is still tedious.

10.2.3 High-Level Languages

The first widespread use of high-level languages in the early 1960s transformed programming into something quite different from what it had been. Programs were written in an English-like manner, thus making them more convenient to use. As a result, a programmer could accomplish more with less effort, and programs could now direct much more complex tasks.

These so-called third-generation languages spurred the great increase in data processing that characterized the 1960s and 1970s. During that time the number of mainframes in use increased from hundreds to tens of thousands. The impact of third-generation languages on our society has been enormous.

Of course, a translator is needed to translate the symbolic statements of a high-level language into computer-executable machine language; this translator is usually a compiler. There are many compilers for each language and one for each type of computer. Since the machine language generated by one computer's COBOL compiler, for instance, is not the machine language of some other computer, it is necessary to have a COBOL compiler for each type of computer on which COBOL programs are to be run. Keep in mind, however, that even though a given program would be compiled to different machine language versions on different machines, the source program itself-the COBOL version-can be essentially identical on each machine.

Some languages are created to serve a specific purpose, such as controlling industrial robots or creating graphics. Many languages, however, are extraordinarily flexible and are considered to be general-purpose. In the past the majority of programming applications were written in BASIC, FORTRAN, or COBOL-all general-purpose languages. In addition to these three, another popular high-level language is C, which we will discuss later.

10.2.4 Very High-Level Languages

Languages called very high-level languages are often known by their generation number, that is, they are called fourth-generation languages or, more simply, 4GLs.

Definition: Will the real fourth-generation languages please stand up? There is no consensus about what constitutes a fourth-generation language. The 4GLs are essentially shorthand programming languages. An operation that requires hundreds of lines in a third-generation language such as COBOL typically requires only five to ten lines in a 4GL. However, beyond the basic criterion of conciseness, 4GLs are difficult to describe.

Characteristics: Fourth-generation languages share some characteristics. The first is that they make a true break with the prior generation—they are basically non-procedural. A procedural language tells the computer how a task is done: Add this, compare that, do this if something is true, and so forth—a very specific step-by-step process. The first three generations of languages are all procedural. In a nonprocedural language, the concept changes. Here, users define only what they want the computer to do; the user does not provide the details of just how it is to be done. Obviously, it is a lot easier and faster just to say what you want rather than how to get it. This leads us to the issue of productivity, a key characteristic of fourth-generation languages.

Productivity: Folklore has it that fourth-generation languages can improve productivity by a factor of 5 to 50. The folklore is true. Most experts say the average improvement factor is about 10—that is, you can be ten times more productive in a fourth-generation language than in a third-generation language. Consider this request: Produce a report showing the total units sold for each product, by customer, in each month and year, and with a subtotal for each customer. In addition, each new customer must start on a new page. A 4GL request looks something like this:

```
TABLE FILE SALES
SUM UNITS BY MONTH BY CUSTOMER BY PRODUCT
ON CUSTOMER SUBTOTAL PAGE BREAK
END
```

Even though some training is required to do even this much, you can see that it is pretty simple. The third-generation language COBOL, however, typically requires over 500 statements to fulfill the same request. If we define productivity as producing equivalent results in less time, then fourth-generation languages clearly increase productivity.

Downside: Fourth-generation languages are not all peaches and cream and productivity. The 4GLs are still evolving, and that which is still evolving cannot be fully defined or standardized. What is more, since many 4GLs are easy to use, they attract a large number of new users, who may then overcrowd the computer system. One of the main criticisms is that the new languages lack the necessary control and flexibility when it comes to planning how you want the output to look. A common perception of 4GLs is that they do not make efficient use of machine resources; however, the benefits of getting a program finished more quickly can far outweigh the extra costs of running it.

Benefits: Fourth-generation languages are beneficial because

- They are results-oriented; they emphasize what instead of how.
- They improve productivity because programs are easy to write and change.
- They can be used with a minimum of training by both programmers and nonprogrammers.
- They shield users from needing an awareness of hardware and program structure.

It was not long ago that few people believed that 4GLs would ever be able to replace third-generation languages. These 4GL languages are being used, but in a very limited way.

10.2.5 Query Languages

A variation on fourth-generation languages are query languages, which can be used to retrieve information from databases. Data is usually added to databases according to a plan, and planned reports may also be produced. But what about a user who needs an unscheduled report or a report that differs somehow from the standard reports? A user can learn a query language fairly easily and then be able to input a request and receive the resulting report right on his or her own terminal or personal computer. A standardized query language, which can be used with several different commercial database programs, is Structured Query Language, popularly known as SQL. Other popular query languages are Query-by-Example, known as QBE and Intellect.

10.2.6 Natural Languages

The word “natural” has become almost as popular in computing circles as it has in the supermarket. Fifth-generation languages are, as you may guess, even more ill-defined than fourth-generation languages. They are most often called natural languages because of their resemblance to the “natural” spoken English language. And, to the manager new to computers for whom these languages are now aimed, natural means human-like. Instead of being forced to key correct commands and data names in correct order, a manager tells the computer what to do by keying in his or her own words.

Figure 10.2: Example of Natural Language Interaction

```

Hello
How may I help you?
Who are my customers in Chicago?
Just a sec. I'll see.
The customers in that city are:
I.D.           Name
-----
Ballard        Ballard and Sons, Inc.
Fremont        Henry Fremont Associates
Greenlake      Greenlake Consortium
Wallingford    Wallingford, Inc.
What can I do for you now?
What is Fremont's balance?
Hang on. I'll see.
Accounts Receivable    563.47
Unapplied Credit       79.16
Balance                484.31
What else can I do for you?
Give me Fremont's phone number!
Please wait until I check the files.
(312) 789-5562
What can I do for you now?
  
```

Notes

A manager can say the same thing any number of ways. For example, "Get me tennis racket sales for January" works just as well as "I want January tennis racket revenues." Such a request may contain misspelled words, lack articles and verbs, and even use slang. The natural language translates human instructions-bad grammar, slang, and all-into code the computer understands. If it is not sure what the user has in mind, it politely asks for further explanation.

Natural languages are sometimes referred to as knowledge-based languages, because natural languages are used to interact with a base of knowledge on some subject. The use of a natural language to access a knowledge base is called a knowledge-based system.

Consider this request that could be given in the 4GL Focus: "SUM ORDERS BY DATE BY REGION." If we alter the request and, still in Focus, say something like "Give me the dates and the regions after you've added up the orders," the computer will spit back the user-friendly version of "You've got to be kidding" and give up. But some natural languages can handle such a request. Users can relax the structure of their requests and increase the freedom of their interaction with the data.

Here is a typical natural language request:

REPORT THE BASE SALARY, COMMISSIONS AND YEARS OF
SERVICE BROKEN DOWN BY STATE AND CITY FOR SALESCLERKS
IN NEW JERSEY AND MASSACHUSETTS.

10.2.7 Choosing a Language

How do you choose the language with which to write your program?

There are several possibilities:

- In a work environment, your manager may decree that everyone on your project will use a certain language.
- You may use a certain language, particularly in a business environment, based on the need to interface with other programs; if two programs are to work together, it is easiest if they are written in the same language.
- You may choose a language based on its suitability for the task. For example, a business program that handles large files may be best written in the business language COBOL.
- If a program is to be run on different computers, it must be written in a language that is portable-suitable on each type of computer-so that the program need be written only once.
- You may be limited by the availability of the language. Not all languages are available in all installations or on all computers.
- The language may be limited to the expertise of the programmer; that is, the program may have to be written in a language the available programmer knows.
- Perhaps the simplest reason, one that applies to many amateur programmers, is that they know the language called BASIC because it came with-or was inexpensively purchased with-their personal computers.

10.3 Summary

- The programmer prepares the instructions of a computer program and runs those instructions on the computer tests the program to see if it is working properly and makes corrections to the program.
- Programming languages are said to be “lower” or “higher”, depending on how close they are to the language the computer itself uses or the language people use.
- The programmer who uses an assembly languages requires a translator to convert the assembly language program into machine language.
- Languages called very high-level languages are often known by their generation number, that is they are called fourth-generation languages or more simply 4 GLS.
- A standardized query language which can be used with several different commercial data base programs, is structured query language, popularly known as SQL.

10.4 Keywords

Programming language: A programming language is an artificial language designed to express computations that can be performed by a machine, particularly a computer.

Self-modifying programs: Self-modifying program is program that alters its own instructions while it is executing – usually to reduce the instruction path length and improve performance or simply to reduce otherwise repetitively similar code, thus simplifying maintenance.

Software interfaces: A software interface may refer to a range of different types of interface at different “levels”: an operating system may interface with pieces of hardware, applications or programs running on the operating system may need to interact via streams, and in object oriented programs, objects within an application may need to interact via methods.



Lab Exercise Write a function of factorial using C language.

10.5 Self-Assessment Questions

1. A computer programmer
 - (a) enters data into computer
 - (b) writes programs
 - (c) changes flow chart into instructions
 - (d) provides solutions to complex problems
 - (e) does total planning and thinking for a computer
2. The most widely used commercial programming computer language is

(a) BASIC	(b) COBOL
(c) FORTRAN	(d) PASCAL

Notes

3. Which one of the following is not a programming language of a computer?
- | | |
|-----------|-------------|
| (a) BASIC | (b) FORTRAN |
| (c) LASER | (d) PASCAL |

10.6 Review Questions

1. What are computer programs?
2. What are quality requirements in programming?
3. What is the term debugging mean?
4. Why programming is needed? What are its uses?
5. Give the levels of programming languages.
6. What are the characteristics of very high-level languages and give its uses as well?
7. Give a brief introduction of major programming languages.

Answers for Self-Assessment Questions

1. (e) 2. (b) 3. (c)

10.7 Further Reading



Books

Computing Fundamentals by Peter Norton



Online link

<http://www.methodsandtools.com/archive/>

Unit 11: Programming Process

Notes

CONTENTS

Objectives

Introduction

11.1 Categories of Programming Language

11.1.1 Scripting

11.1.2 Programmer's Scripting

11.1.3 Application Development

11.1.4 Low-level

11.1.5 Pure Functional

11.1.6 Complete Core

11.2 Machine and Assembly Language

11.2.1 Machine Language

11.2.2 Reading Machine Language

11.2.3 Assembly Language

11.3 High Level Languages

11.4 World Wide Web (WWW) Development Language

11.4.1 Function

11.4.2 Linking

11.4.3 Dynamic Updates of Web Pages

11.4.4 WWW Prefix

11.4.5 Privacy

11.4.6 Security

11.4.7 Standards

11.4.8 Accessibility

11.4.9 Internationalization

11.4.10 Statistics

11.4.11 Speed Issues

11.4.12 Caching

Notes

11.5 Summary

11.6 Keywords

11.7 Self-Assessment Questions

11.8 Review Questions

11.9 Further Reading

Objectives

After studying this unit, you will be able to:

- Understand the categories of computer language.
- Discussed machine and assembly language.
- Explained high level languages.
- Understand the www development language.

Introduction

A **programming language** is an artificial language designed to express computations that can be performed by a machine, particularly a computer. Programming languages can be used to create programs that control the behavior of a machine, to express algorithms precisely, or as a mode of human communication. The earliest programming languages predate the invention of the computer, and were used to direct the behavior of machines such as Jacquard looms and player pianos. Thousands of different programming languages have been created, mainly in the computer field, with many more being created every year. Most programming languages describe computation in an imperative style, i.e., as a sequence of commands, although some languages, such as those that support functional programming or logic programming, use alternative forms of description. A programming language is usually split into the two components of syntax (form) and semantics (meaning) and many programming languages have some kind of written specification of their syntax and/or semantics. Some languages are defined by a specification document, for example, the C programming language is specified by an ISO Standard, while other languages, such as Perl, have a dominant implementation that is used as a reference.

11.1 Categories of Programming Language

There are literally thousands of programming languages in the world and each has its own strengths and weakness. Many are simply for teaching or language research. These languages do not interest or concern me. They are frequently limited and almost useless. There is a certain set of programming fields/categories that I believe are currently distinct enough to have an independent programming language to represent them. Each is however, narrow enough that there is no need for more than one language.

Language Role**Best Candidate**

Scripting

Programmer's scripting

Better Scheme (+OO)

Application Development

Notes

Low-level

C/C++

Pure Functional

Complete Core

11.1.1 Scripting

The most basic need of for scripting languages is one which is simple to use because it will be used by those with a minimum of programming knowledge or in situations where cranking out code which does the job is all the matters. An example of where this language might be used by people with more of a programming background is in web scripting. Examples of languages that might fall into this category are PHP, JavaScript, Perl, and Python. VB could also be considered in this family but it is too frequently used for application development today.

11.1.2 Programmer's Scripting

Real programmers also often have a need for scripting capabilities. However, they need a language which is simple and complete. It must never stand in there way as is all to often the case with standard scripting languages. It must be able to easily integrate into other languages.

Better Scheme is an excellent language for use in Programmer's Scripting. It is simple, complete, powerful and concise. In addition many programmers are exposed to Scheme or Lisp during their education and so are already familiar with it. One thing which will most likely be needed is a good extension to Better Scheme which provides solid support for object oriented programming because Better Scheme scripts will frequently need to interact with objects created in other languages.

11.1.3 Application Development

To develop applications a language is needed which is complete and powerful but very safe. The complexity of large application mandates a compile time safe language. It should also be fairly high level and object oriented.

11.1.4 Low-level

For the purpose of operating systems and other low level code we need a language that operates fast and just above the machine level. Of course there will always be occasion to slip into assembly but that is by its nature machine specific and so not considered here.

Both C and C++ can be used for this. They provide the low level power needed while giving a reasonable level of safety and abstraction.

11.1.5 Pure Functional

Certain tasks can best be done working in a purely function environment. And these languages while not as widely used today are in many ways so distinct from others that it is important to keep there legacy alive in the hopes that they may positively influence more common languages. The only real example of this today is Haskell since languages like ML are not purely functional.

11.1.6 Complete Core

This language may never be used for real programming but I still think it would be important to have in mind. It would be a core language which the application, scripting and pure functional

Notes

languages could build on and modify. It would have everything truly necessary and little more. One might think that the lambda calculus would be a minimal example of this. However, it makes no provision for mutability which is clearly a key concept to many languages. In addition the lambda calculus has no concept of types, another important feature of modern languages. There are no examples of a complete core language today.

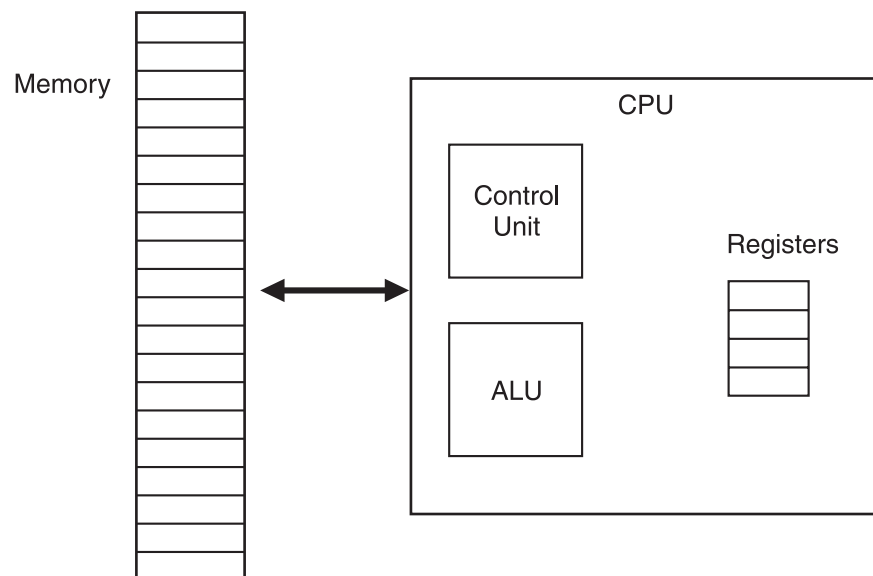
11.2 Machine and Assembly Language

11.2.1 Machine Language

Machine language: A language that need not be modified, translated, or interpreted before it can be used by the processor for which it was designed.

1. The operation codes and addresses used in instructions written in machine language can be directly sensed by the arithmetic and control unit circuits of the processor for which the language is designed.
2. Instructions written in an assembly language or a high-level language must be translated into machine language before they can be executed by a processor.
3. Machine languages are usually used by computer designers rather than computer users.
 - (a) The machine language for a particular computer is tied to the architecture of the CPU.
 - (b) For example: G4 Macs have a different machine language than Intel PC's.
 - (c) We will look at the machine language of a simple, simulated computer.

Von Neumann Architecture





- Example:* (a) Our computer has 4 registers and 32 memory locations.
 (b) Each instruction is 16 bits.
 (c) Here is a machine language program for our simulated computer:

```
1000000100100101
1000000101000101
1010000100000110
1000001000000110
1111111111111111
```

Instruction ID **Register #** **Memory Location**

LOAD contents of memory location into register	100000010 RR MMMMM example: R0 = Mem[3] 100000010 00 00011
LOAD contents of memory location into register	100000010 RR MMMMM ex: R0 = Mem [3] 100000010 00 00011
Store contents of register into memory location	100000010 RR MMMMM ex: Mem[4] = R0 100000100 00 00100
MOVE contents of one register into another register	100100010000 RR RR ex: R0 = R1 100100010000 00 01
ADD contents of 2 registers, store result in third. SUBTRACT contents of 2 registers, store result into third Halt the program	1010000100 RR RR RR ex: R0 = R1 + R2 1010000100 00 01 10 1010001000 RR RR RR ex: R0 = R1 – R2 1010001000 00 01 10 1111111111111111

11.2.2 Reading Machine Language

- (a) In our case, first nine bits specifies the operation, last 6 (or 7) bits specifies the arguments:

```
100000010 01 00101      Load Memory 5 – > R1
100000010 10 00101      Load Memory 5 – > R2
1010000100 00 01 10      R1 + R2 – > R0
1111111111111111      Store R0 – > Memory 6
```

- (b) It is very tedious to program in machine language.

Notes

Notes

11.2.3 Assembly Language

An assembly language is a low-level programming language for computers, microprocessors, microcontrollers, and other programmable devices. It implements a symbolic representation of the machine and other constants needed to program a given CPU architecture. This representation is usually defined by the hardware manufacturer, and is based on mnemonics that symbolize processing steps (instructions), processor registers, memory locations, and other language features. An assembly language is thus specific to a certain physical (or virtual) computer architecture. This is in contrast to most high-level programming languages, which, ideally, are portable.

A utility program called an assembler is used to translate assembly language statements into the target computer's machine code. The assembler performs a more or less isomorphic translation (a one-to-one mapping) from mnemonic statements into machine instructions and data. This is in contrast with high-level languages, in which a single statement generally results in many machine instructions.

- (a) Assembly instructions are just shorthand for machine instructions:

Machine Language	Equivalent Assembly
1000000100100101	Load R1 5
1000000101000101	Load R2 5
1010000100000110	ADD R0 R1 R2
1000001000000110	SAVE R0 6
1111111111111111	HALT

- (b) (For all assembly instructions that compute a result, the first argument is the destination.)
 (c) Very easy to write an Assembly Language > Machine language translator.



Task

What would be the assembly instruction to swap the contents of registers 1 & 2?



Caution

- We are missing some crucial functionality...
- Loops!



Example:

R0	3
R1	1
R2	Number
R3	0

0	ADD R3 R2 R3
R1	SUB R0 R0 R1
R2	BZERO 4
R3	BRANCH 4
4	MOVE R2 R3
5	HALT

In Matlab**Notes**

- The same program in Matlab would be the following:

```
z = 0;
x = 3;
while x ~ = 0
    z = z + y;
    x = x - 1;
y = z;
```

- Or the following:

```
y = y*3;
```

11.3 High Level Languages

A **high-level programming language** is a programming language with strong abstraction from the details of the computer. In comparison to low-level programming languages, it may use natural language elements, be easier to use, or be more portable across platforms. Such languages hide the details of CPU operations such as memory access models and management of scope.

Very early in the development of computers attempts were made to make programming easier by reducing the amount of knowledge of the internal workings of the computer that was needed to write programs. If programs could be presented in a language that was more familiar to the person solving the problem, then fewer mistakes would be made. High-level programming languages allow the specification of a problem solution in terms closer to those used by human beings. These languages were designed to make programming far easier, less error-prone and to remove the programmer from having to know the details of the internal structure of a particular computer. These high-level languages were much closer to human language. One of the first of these languages was Fortran II which was introduced in about 1958. In Fortran II our program above would be written as:

```
C = A + B
```

which is obviously much more readable, quicker to write and less error-prone. As with assembly languages the computer does not understand these high-level languages directly and hence they have to be processed by passing them through a program called a compiler which translates them into internal machine language before they can be executed.

Another advantage accrues from the use of high-level languages if the languages are standardized by some international body. Then each manufacturer produces a compiler to compile programs that conform to the standard into their own internal machine language. Then it should be easy to take a program which conforms to the standard and implement it on many different computers merely by re-compiling it on the appropriate computer. This great advantage of portability of programs has been achieved for several high-level languages and it is now possible to move programs from one computer to another without too much difficulty. Unfortunately many compiler writers add new features of their own which means that if a programmer uses these features then their program becomes non-portable. It is well worth becoming familiar with the standard and writing programs which obey it, so that your programs are more likely to be portable.

As with assembly language human time is saved at the expense of the compilation time required to translate the program to internal machine language. The compilation time used in the computer

Notes

is trivial compared with the human time saved, typically seconds as compared with weeks.

Many high level languages have appeared since Fortran II (and many have also disappeared!), among the most widely used have been:

COBOL: Business applications

FORTRAN: Engineering & Scientific Applications

PASCAL: General use and as a teaching tool

C & C++: General Purpose - currently most popular

PROLOG: Artificial Intelligence

JAVA: General Purpose - gaining popularity rapidly

All these languages are available on a large variety of computers.

11.4 World Wide Web (WWW) Development Language

The **World Wide Web**, abbreviated as **WWW** or **W3** and commonly known as **the Web**, is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and navigate between them via hyperlinks. Using concepts from earlier hypertext systems, English engineer and computer scientist Sir Tim Berners-Lee, now the Director of the World Wide Web Consortium, wrote a proposal in March 1989 for what would eventually become the World Wide Web.

“The World-Wide Web was developed to be a pool of human knowledge, and human culture, which would allow collaborators in remote sites to share their ideas and all aspects of a common project?”

11.4.1 Function

The terms Internet and World Wide Web are often used in everyday speech without much distinction. However, the Internet and the World Wide Web are not one and the same. The Internet is a global system of interconnected computer networks. In contrast, the Web is one of the services that runs on the Internet. Viewing a web page on the World Wide Web normally begins either by typing the URL of the page into a web browser, or by following a hyperlink to that page or resource. The web browser then initiates a series of communication messages, behind the scenes, in order to fetch and display it. First, the browser resolves the server-name portion of the URL (en.wikipedia.org) into an Internet Protocol address using the global, distributed Internet database known as the Domain Name System (DNS); this lookup returns an IP address such as 208.80.152.2. The browser then requests the resource by sending an HTTP request across the Internet to the computer at that particular address. It makes the request to a particular application port in the underlying Internet Protocol Suite so that the computer receiving the request can distinguish an HTTP request from other network protocols such as e-mail delivery; the HTTP protocol normally uses port 80. The content of the HTTP request can be as simple as the two lines of text.

The computer receiving the HTTP request delivers it to Web server software listening for requests on port 80. If the web server can fulfill the request it sends an HTTP response back to the browser indicating success, which can be as simple as:

HTTP/1.0 200 OK

Content-Type: text/html; charset=UTF-8

followed by the content of the requested page. The Hypertext Markup Language for a basic web page looks like

```
<html>
<head>
<title>World Wide Web - Wikipedia, the free encyclopedia</title>
</head>
<body>
<p>The <b>World Wide Web</b>, abbreviated as <b>WWW</b> and commonly known ...</p>
</body>
</html>
```

The web browser parses the HTML, interpreting the markup (<title>, for bold, and such) that surrounds the words in order to draw that text on the screen.

Many web pages consist of more elaborate HTML which references the URLs of other resources such as images, other embedded media, scripts that affect page behavior, and Cascading Style Sheets that affect page layout. A browser that handles complex HTML will make additional HTTP requests to the web server for these other Internet media types. As it receives their content from the web server, the browser progressively renders the page onto the screen as specified by its HTML and these additional resources.

11.4.2 Linking

Most web pages contain hyperlinks to other related pages and perhaps to downloadable files, source documents, definitions and other web resources. Graphic representation of a minute fraction of the WWW, demonstrating hyperlinks such a collection of useful, related resources, interconnected via hypertext links is dubbed a web of information. Publication on the Internet created what Tim Berners-Lee first called the Worldwide Web (in its original CamelCase, which was subsequently discarded) in November 1990.

Over time, many web resources pointed to by hyperlinks disappear, relocate, or are replaced with different content. This makes hyperlinks obsolete, a phenomenon referred to in some circles as link rot and the hyperlinks affected by it are often called dead links. The ephemeral nature of the Web has prompted many efforts to archive web sites. The Internet Archive, active since 1996, is one of the best-known efforts.

11.4.3 Dynamic Updates of Web Pages

JavaScript is a scripting language that was initially developed in 1995 by Brendan Eich, then of Netscape, for use within web pages. To overcome some of the limitations of the page-by-page model described above, some web applications also use Ajax (asynchronous JavaScript and XML). JavaScript is delivered with the page that can make additional HTTP requests to the server, either in response to user actions such as mouse-clicks, or based on lapsed time. The server's responses are used to modify the current page rather than creating a new page with each response. Thus the server only needs to provide limited, incremental information. Since multiple Ajax requests can

Notes

be handled at the same time, users can interact with a page even while data is being retrieved. Some web applications regularly poll the server to ask if new information is available.

11.4.4 WWW Prefix

Many domain names used for the World Wide Web begin with *www* because of the long-standing practice of naming Internet hosts (servers) according to the services they provide. The hostname for a web server is often *www*, in the same way that it may be *ftp* for an FTP server, and *news* or *nntp* for a USENET news server. These host names appear as Domain Name System (DNS) subdomain names, as in *www.example.com*. The use of ‘*www*’ as a subdomain name is not required by any technical or policy standard; indeed, the first ever web server was called *nxoc01.cern.ch*, and many web sites exist without it. Many established websites still use ‘*www*’, or they invent other subdomain names such as ‘*www2*’, ‘*secure*’, etc. Many such web servers are set up such that both the domain root (e.g., *example.com*) and the *www* subdomain (e.g., *www.example.com*) refer to the same site; others require one form or the other, or they may map to different web sites.

The use of a subdomain name is useful for load balancing incoming web traffic by creating a CNAME record that points to a cluster of web servers. Since, currently, only a subdomain can be *cname*’ed the same result cannot be achieved by using the bare domain root.

In English, *www* is pronounced by individually pronouncing the name of characters (*double-u double-u double-u*). Although some technical users pronounce it *dub-dub-dub* this is not widespread. The English writer Douglas Adams once quipped in *The Independent* on Sunday (1999): “The World Wide Web is the only thing I know of whose shortened form takes three times longer to say than what it’s short for,” with Stephen Fry later pronouncing it in his “Podgrammes” series of podcasts as “wuh wuh wuh.” In Mandarin Chinese, *World Wide Web* is commonly translated via a phono-semantic matching to *wàn wéi w?ng*), which satisfies *www* and literally means “myriad dimensional net” a translation that very appropriately reflects the design concept and proliferation of the World Wide Web.



In English, *www* is pronounced by individually pronouncing the name of characters (*double-u double-u double-u*). Although some technical users pronounce it *dub-dub-dub* this is not widespread.

11.4.5 Privacy

Computer users, who save time and money, and who gain conveniences and entertainment, may or may not have surrendered the right to privacy in exchange for using a number of technologies including the Web Worldwide, more than a half billion people have used a social network service, and of Americans who grew up with the Web, half created an online profile] and are part of a generational shift that could be changing norms. Facebook progressed from U.S. college students to a 70% non-U.S. audience, and in 2009 estimated that only 20% of its members use privacy settings. In 2010 (six years after co-founding the company), Mark Zuckerberg wrote, “we will add privacy controls that are much simpler to use”.

Privacy representatives from 60 countries have resolved to ask for laws to complement industry self-regulation, for education for children and other minors who use the Web, and for default protections for users of social networks. They also believe data protection for personally identifiable information benefits business more than the sale of that information. Users can opt-in to features in browsers to clear their personal histories locally and block some cookies and advertising

networks but they are still tracked in websites' server logs, and particularly web beacons. Berners-Lee and colleagues see hope in accountability and appropriate use achieved by extending the Web's architecture to policy awareness, perhaps with audit logging, reasoners and appliances.

In exchange for providing free content, vendors hire advertisers who spy on Web users and base their business model on tracking them. Since 2009, they buy and sell consumer data on exchanges (lacking a few details that could make it possible to de-anonymize, or identify an individual). Hundreds of millions of times per day, Lotame Solutions captures what users are typing in real time, and sends that text to OpenAmplify who then tries to determine, to quote a writer at *The Wall Street Journal*, "what topics are being discussed, how the author feels about those topics, and what the person is going to do about them".

Microsoft backed away in 2008 from its plans for strong privacy features in Internet Explorer, leaving its users (50% of the world's Web users) open to advertisers who may make assumptions about them based on only one click when they visit a website. Among services paid for by advertising, Yahoo! could collect the most data about users of commercial websites, about 2,500 bits of information per month about each typical user of its site and its affiliated advertising network sites.

11.4.6 Security

The Web has become criminals' preferred pathway for spreading malware. Cybercrime carried out on the Web can include identity theft, fraud, espionage and intelligence gathering. Web-based vulnerabilities now outnumber traditional computer security concerns, and as measured by Google, about one in ten web pages may contain malicious code. Most Web-based attacks take place on legitimate websites, and most, as measured by Sophos, are hosted in the United States, China and Russia. Through HTML and URIs the Web was vulnerable to attacks like cross-site scripting (XSS) that came with the introduction of JavaScript and were exacerbated to some degree by Web 2.0 and Ajax web design that favors the use of scripts. Today by one estimate, 70% of all websites are open to XSS attacks on their users.

Proposed solutions vary to extremes. Large security vendors like McAfee already design governance and compliance suites to meet post-9/11 regulations, and some, like Finjan have recommended active real-time inspection of code and all content regardless of its. Some have argued that for enterprise to see security as a business opportunity rather than a cost center, "ubiquitous, always-on digital rights management" enforced in the infrastructure by a handful of organizations must replace the hundreds of companies that today secure data and networks. Jonathan Zittrain has said users sharing responsibility for computing safety is far preferable to locking down the Internet.

11.4.7 Standards

Web Standards

Many formal standards and other technical specifications and software define the operation of different aspects of the World Wide Web, the Internet, and computer information exchange. Many of the documents are the work of the World Wide Web Consortium (W3C), headed by Berners-Lee, but some are produced by the Internet Engineering Task Force (IETF) and other organizations.

Usually, when web standards are discussed, the following publications are seen as foundational:

- (a) Recommendations for markup languages, especially HTML and XHTML, from the W3C. These define the structure and interpretation of hypertext documents.
- (b) Recommendations for stylesheets, especially CSS, from the W3C.

Notes

- (c) Standards for ECMAScript (usually in the form of JavaScript), from Ecma International.
- (d) Recommendations for the Document Object Model, from W3C.

Additional publications provide definitions of other essential technologies for the World Wide Web, including, but not limited to, the following:

- (i) Uniform Resource Identifier (URI), which is a universal system for referencing resources on the Internet, such as hypertext documents and images. URIs, often called URLs, are defined by the IETF's RFC 3986 / STD 66: Uniform Resource Identifier (URI): Generic Syntax, as well as its predecessors and numerous URI scheme-defining RFCs;
- (ii) HyperText Transfer Protocol (HTTP), especially as defined by RFC 2616: HTTP/1.1 and RFC 2617: HTTP Authentication, which specify how the browser and server authenticate each other.

11.4.8 Accessibility

Web Accessibility

Access to the Web is for everyone regardless of disability including visual, auditory, physical, speech, cognitive, or neurological. Accessibility features also help others with temporary disabilities like a broken arm or the aging population as their abilities change. The Web is used for receiving information as well as providing information and interacting with society, making it essential that the Web be accessible in order to provide equal access and equal opportunity to people with disabilities. Tim Berners-Lee once noted, "The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect."

11.4.9 Internationalization

The W3C Internationalization Activity assures that web technology will work in all languages, scripts, and cultures. Beginning in 2004 or 2005, Unicode gained ground and eventually in December 2007 surpassed both ASCII and Western European as the Web's most frequently used character encoding. Originally RFC 3986 allowed resources to be identified by URI in a subset of US-ASCII. RFC 3987 allows more characters-any character in the Universal Character Set-and now a resource can be identified by IRI in any language.

11.4.10 Statistics

Between 2005 and 2010, the number of Web users doubled, and was expected to surpass two billion in 2010. A 2002 survey of 2,024 million Web pages] determined that by far the most Web content was in English: 56.4%; next were pages in German (7.7%), French (5.6%), and Japanese (4.9%). A more recent study, which used Web searches in 75 different languages to sample the Web, determined that there were over 11.5 billion Web pages in the publicly indexable Web as of the end of January 2005. As of March 2009, the indexable web contains at least 25.21 billion pages.http://en.wikipedia.org/wiki/World_Wide_Web - cite_note-71 On July 25, 2008, Google software engineers Jesse Alpert and Nissan Hajaj announced that Google Search had discovered one trillion unique URLs. As of May 2009, over 109.5 million websites operated. Of these 74% were commercial or other sites operating in the .com generic top-level domain.

Statistics measuring a website's popularity are usually based either on the number of page views or associated server 'hits' (file requests) that it receives.

11.4.11 Speed Issues

Frustration over congestion issues in the Internet infrastructure and the high latency that results in slow browsing has led to a pejorative name for the World Wide Web: the World Wide Wait.

Speeding up the Internet is an ongoing discussion over the use of peering and QoS technologies. Guidelines for Web response times are:

- (i) 0.1 second (one tenth of a second). Ideal response time. The user doesn't sense any interruption.
- (ii) 1 second. Highest acceptable response time. Download times above 1 second interrupt the user experience.
- (iii) 10 seconds. Unacceptable response time. The user experience is interrupted and the user is likely to leave the site or system.

11.4.12 Caching

If a user revisits a Web page after only a short interval, the page data may not need to be re-obtained from the source Web server. Almost all web browsers cache recently obtained data, usually on the local hard drive. HTTP requests sent by a browser will usually only ask for data that has changed since the last download. If the locally cached data are still current, it will be reused. Caching helps reduce the amount of Web traffic on the Internet. The decision about expiration is made independently for each downloaded file, whether image, stylesheet, JavaScript, HTML, or whatever other content the site may provide. Thus even on sites with highly dynamic content, many of the basic resources only need to be refreshed occasionally. Web site designers find it worthwhile to collate resources such as CSS data and JavaScript into a few site-wide files so that they can be cached efficiently. This helps reduce page download times and lowers demands on the Web server.

There are other components of the Internet that can cache Web content. Corporate and academic firewalls often cache Web resources requested by one user for the benefit of all. (See also Caching proxy server.) Some search engines also store cached content from websites. Apart from the facilities built into Web servers that can determine when files have been updated and so need to be re-sent, designers of dynamically generated Web pages can control the HTTP headers sent back to requesting users, so that transient or sensitive pages are not cached. Internet banking and news sites frequently use this facility. Data requested with an HTTP 'GET' is likely to be cached if other conditions are met; data obtained in response to a 'POST' is assumed to depend on the data that was POSTed and so is not cached.



Did u know?

Web caching is the caching of web documents (e.g., HTML pages, images) to reduce bandwidth usage, server load, and perceived lag. A web cache stores copies of documents passing through it; subsequent requests may be satisfied from the cache if certain conditions are met.



Case Study

The Issue

Eight of the top 50 e-services consulting, creative, and technology companies joined to form this client company — eight separate companies with eight separate sales forces combined into one. Our contact was the CEO of the largest company. He understood that a sale is a process and that to effect a smooth merger, the newly combined sales teams would need to be on the same page with a common sales methodology. When our contact was appointed EVP for corporate development for the newly formed company, he knew that he had to bring these eight companies together through common language and process.

Contd...

Notes**The Solution**

Applying the methodology they learned from Miller Heiman's Strategic Selling®, the newly formed team began working very closely with their client to really understand what their needs were. They got the go-ahead to make a presentation to a panel charged with making the final decision. The meeting was to take place in the afternoon. That morning, the team met and went through a dry run of the presentation. They walked through the sales plan and identified all the people who were going to be in the meeting and what messages they had to get to each one of those people. They asked confirmation questions to determine that everyone in the room was on the same page. Before the presentation, everyone was that they had covered all of the bases.

The Result

How Miller Heiman helped this client is succinctly expressed by our contact:

"The Blue sheets and the Green sheets were tremendous for facilitating clear communication among the members of the team that led to this success.

Now, salespeople throughout the company are talking about Economic Buyers, Red Flags, Blue Sheets, and Confirmation Questions [components of the Miller Heiman Sales System]. It's becoming part of our language and becoming part of our culture. It will be so embedded that it will simply be how our company does business. We're not there yet, but we've got a great start and we're training more of our top people all of the time."

Question:

Explain the methodology explain Miller Heiman's.



Lab Exercise

1. Draw SDLC diagram.
2. Draw Waterfall model of Software development Life Cycle.

11.5 Summary

- Programming languages can be used to create programs that control the behavior of a machine, to express algorithms precisely, or as a mode of human communication.
- There are certain set of programming categories such as scripting, programmer's scripting, application development, low-level, pure functional and complete core.
- Machine language is a collection of binary digits or bits that the computer reads and interprets and assembly language is a low level programming language using the human readable instruction of the CPU.
- A high-level language isolates the execution semantics of a computer architecture from the specification of the program.
- www is a system of interlinked hypertext documents accessed via the Internet.

11.6 Keywords

A high-level programming language: This is a programming language with strong abstraction from the details of the computer.

ISO 15504: It is also known as Software Process Improvement Capability Determination (SPICE), is a “framework for the assessment of software processes”.

ISO 9000: It describes standards for a formally organized process to manufacture a product and the methods of managing and monitoring progress.

Machine language: The machine language for a particular computer is tied to the architecture of the CPU.

11.7 Self-Assessment Questions

1. A _____ is an artificial language designed to express computations that can be performed by a machine, particularly a computer.
2. It must never stand in there way as is all to often the case with standard scripting languages
(a) True (b) False
3. The purpose of operating systems and other low level code we need a language that operates fast and just above the machine level.
(a) True (b) False
4. _____ language for a particular computer is tied to the architecture of the CPU.
5. _____ are just shorthand for machine instructions.
6. A _____ is a programming language with strong abstraction from the details of the computer.
7. Abstraction penalty is the barrier that prevents high-level programming techniques from being applied in situations where computational resources are limited.
(a) True (b) False
8. Most web pages contain “www” to other related pages and perhaps to downloadable files, source documents, definitions and other web resources.
(a) False (b) True

11.8 Review Questions

1. Define programming language and its categories.
2. What is scripting? Differentiate between programmer scripting and scripting.
3. Give brief discussion on Machine and Assembly Language.
4. Describe reading machine language.
5. Give compilation and interpretation of high level languages.

Notes

Answers for Self-Assessment Questions

- | | |
|--------------------------|------------------------------------|
| 1. programming language | 2. (a) |
| 3. (a) | 4. Machine |
| 5. Assembly instructions | 6. high-level programming language |
| 7. (a) | 8. (a) |

11.9 Further Reading



Books

Maran illustrated Computers Guided Tour, by Ruth Maran; Kelleigh Johnson,
Publisher: Course Technology PTR



Online link

<http://www.alternatives.rzero.com/lang.html>

Unit 12: System Development Life Cycle

Notes

CONTENTS

Objectives

Introduction

12.1 Waterfall Model

12.1.1 Feasibility

12.1.2 Requirement Analysis and Design

12.1.3 Implementation

12.1.4 Testing

12.1.5 Maintenance

12.2 Software Development Activities

12.2.1 Planning

12.2.2 Implementation, Testing and Documenting

12.2.3 Deployment and Maintenance

12.3 Spiral Model

12.4 Iterative and Incremental Development

12.4.1 Agile Development

12.5 Process Improvement Models

12.5.1 Formal Methods

12.6 Summary

12.7 Keywords

12.8 Review Questions

12.9 Further Reading

Notes

Objectives

After studying this unit, you will be able to:

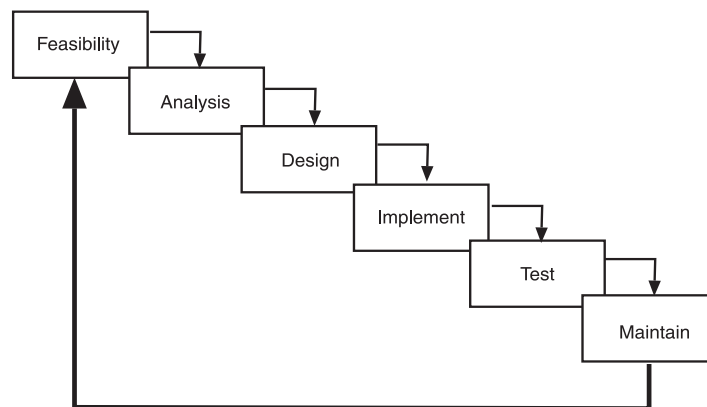
- Explained classical waterfall model.
- Understand software development activity.
- Discussed spiral model.
- Understand the process improvement model.

Introduction

The Systems Development Life Cycle (SDLC) is a conceptual model used in project management that describes the stages involved in an information system development project from an initial feasibility study through maintenance of the completed application. Various SDLC methodologies have been developed to guide the processes involved including the waterfall model (the original SDLC method), rapid application development (RAD), joint application development (JAD), the fountain model and the spiral model. Mostly, several models are combined into some sort of hybrid methodology. Documentation is crucial regardless of the type of model chosen or devised for any application, and is usually done in parallel with the development process. Some methods work better for specific types of projects, but in the final analysis, the most important factor for the success of a project may be how closely particular plan was followed.

12.1 Waterfall Model

The image below is the classic Waterfall model methodology, which is the first SDLC method and it describes the various phases involved in development.



12.1.1 Feasibility

The feasibility study is used to determine if the project should get the go-ahead. If the project is to proceed, the feasibility study will produce a project plan and budget estimates for the future stages of development.

12.1.2 Requirement Analysis and Design

Analysis gathers the requirements for the system. This stage includes a detailed study of the business needs of the organization. Options for changing the business process may be considered. Design focuses on high level design like, what programs are needed and how are they going to interact, low-level design (how the individual programs are going to work), interface design (what are the interfaces going to look like) and data design (what data will be required). During these phases, the software's overall structure is defined. Analysis and Design are very crucial in the whole development cycle. Any glitch in the design phase could be very expensive to solve in the later stage of the software development. Much care is taken during this phase. The logical system of the product is developed in this phase.

12.1.3 Implementation

In this phase the designs are translated into code. Computer programs are written using a conventional programming language or an application generator. Programming tools like Compilers, Interpreters, Debuggers are used to generate the code. Different high level programming languages like C, C++, Pascal, Java are used for coding. With respect to the type of application, the right programming language is chosen.

12.1.4 Testing

In this phase, the system is tested. Normally programs are written as a series of individual modules, these subject to separate and detailed test. The system is then tested as a whole. The separate modules are brought together and tested as a complete system. The system is tested to ensure that interfaces between modules work (integration testing), the system works on the intended platform and with the expected volume of data (volume testing) and that the system does what the user requires (acceptance/beta testing).

12.1.5 Maintenance

Inevitably the system will need maintenance. Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period

A **software development process**, also known as a **software development lifecycle**, is a structure imposed on the development of a software product. Similar terms include software life cycle and software process. There are several models for such processes, each describing approaches to a variety of tasks or activities that take place during the process. Some people consider a lifecycle model a more general term and a software development process a more specific term. For example, there are many specific software development processes that 'fit' the spiral lifecycle model.

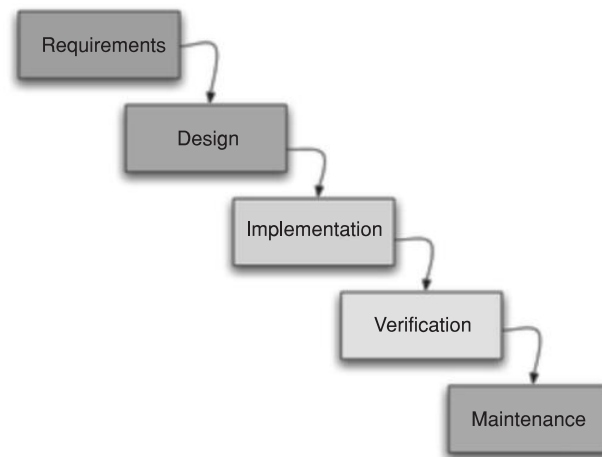


Task

Give usability of each step involved in SDLC.

12.2 Software Development Activities

The large and growing body of software development organizations implement process methodologies. Many of them are in the defense industry, which in the U.S. requires a rating based on 'process models' to obtain contracts.



The international standard for describing the method of selecting, implementing and monitoring the life cycle for software is ISO 12207.

A decades-long goal has been to find repeatable, predictable processes that improve productivity and quality. Some try to systematize or formalize the seemingly unruly task of writing software. Others apply project management techniques to writing software. Without project management, software projects can easily be delivered late or over budget. With large numbers of software projects not meeting their expectations in terms of functionality, cost, or delivery schedule, effective project management appears to be lacking.

Organizations may create a Software Engineering Process Group (SEPG), which is the focal point for process improvement. Composed of line practitioners who have varied skills, the group is at the center of the collaborative effort of everyone in the organization who is involved with software engineering process improvement. The activities of the software development process represented in the waterfall model. There are several other models to represent this process.

12.2.1 Planning

The important task in creating a software product is extracting the requirements or requirements analysis. Customers typically have an abstract idea of what they want as an end result, but not what software should do. Incomplete, ambiguous, or even contradictory requirements are recognized by skilled and experienced software engineers at this point. Frequently demonstrating live code may help reduce the risk that the requirements are incorrect. Once the general requirements are gathered from the client, an analysis of the scope of the development should be determined and clearly stated. This is often called a scope document. Certain functionality may be out of scope of the project as a function of cost or as a result of unclear requirements at the start of development. If the development is done externally, this document can be considered a legal document so that if there are ever disputes, any ambiguity of what was promised to the client can be clarified.

12.2.2 Implementation, Testing and Documenting

Implementation is the part of the process where software engineers actually program the code for the project. Software testing is an integral and important part of the software development process. This part of the process ensures that defects are recognized as early as possible. Documenting the internal design of software for the purpose of future maintenance and enhancement is done throughout development. This may also include the writing of an API, be it external or internal. It is very important to document everything in the project.

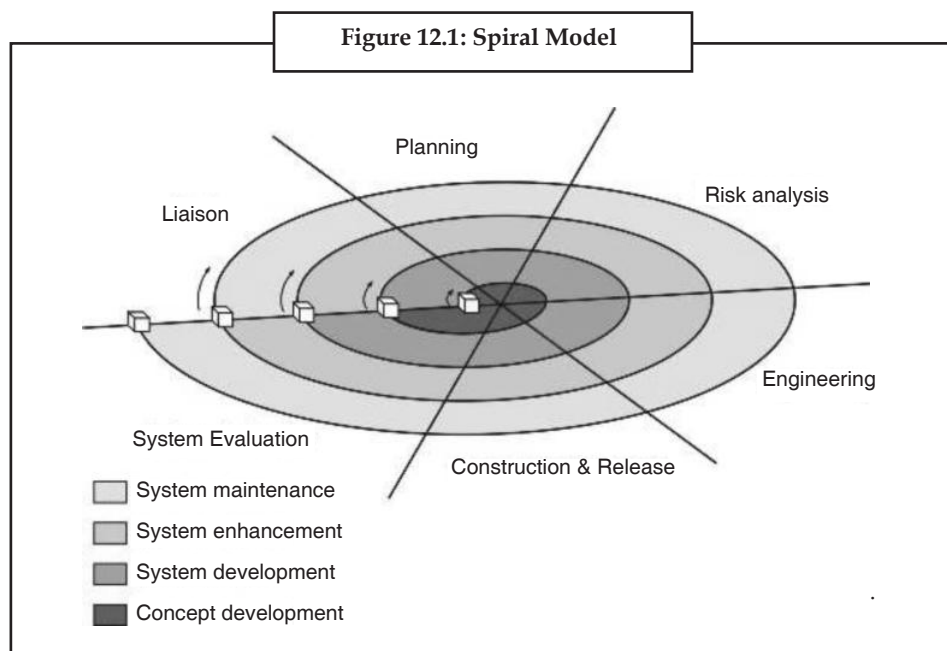
12.2.3 Deployment and Maintenance

Deployment starts after the code is appropriately tested, is approved for release and sold or otherwise distributed into a production environment. Software Training and Support is important and a lot of developers fail to realize that. It would not matter how much time and planning a development team puts into creating software if nobody in an organization ends up using it. People are often resistant to change and avoid venturing into an unfamiliar area, so as a part of the deployment phase, it is very important to have training classes for new clients of your software.

Maintaining and enhancing software to cope with newly discovered problems or new requirements can take far more time than the initial development of the software. It may be necessary to add code that does not fit the original design to correct an unforeseen problem or it may be that a customer is requesting more functionality and code can be added to accommodate their requests. If the labor cost of the maintenance phase exceeds 25% of the prior-phases' labor cost, then it is likely that the overall quality of at least one prior phase is poor. In that case, management should consider the option of rebuilding the system (or portions) before maintenance cost is out of control.

12.3 Spiral Model

The key characteristic of a Spiral model is risk management at regular stages in the development cycle. In 1988, Barry Boehm published a formal software system development "spiral model", which combines some key aspect of the waterfall model and rapid prototyping methodologies, but provided emphasis in a key area many felt had been neglected by other methodologies: deliberate iterative risk analysis, particularly suited to large-scale complex systems.



Notes

The Spiral is visualized as a process passing through some number of iterations, with the four quadrant diagram representative of the following activities:

- (a) formulate plans to: identify software targets, selected to implement the program, clarify the project development restrictions;
- (b) Risk analysis: an analytical assessment of selected programs, to consider how to identify and eliminate risk;
- (c) the implementation of the project: the implementation of software development and verification;

Risk-driven spiral model, emphasizing the conditions of options and constraints in order to support software reuse, software quality can help as a special goal of integration into the product development. However, the spiral model has some restrictive conditions, as follows:

- (i) The spiral model emphasizes risk analysis, and thus requires customers to accept this analysis and act on it. This requires both trust in the developer as well as the willingness to spend more to fix the issues, which is the reason why this model is often used for large-scale internal software development.
- (ii) If the implementation of risk analysis will greatly affect the profits of the project, the spiral model should not be used.
- (iii) Software developers have to actively look for possible risks, and analyze it accurately for the spiral model to work.

The first stage is to formulate a plan to achieve the objectives with these constraints, and then strive to find and remove all potential risks through careful analysis and, if necessary, by constructing a prototype. If some risks can not be ruled out, the customer has to decide whether to terminate the project or to ignore the risks and continue anyway. Finally, the results are evaluated and the design of the next phase begins.



Task

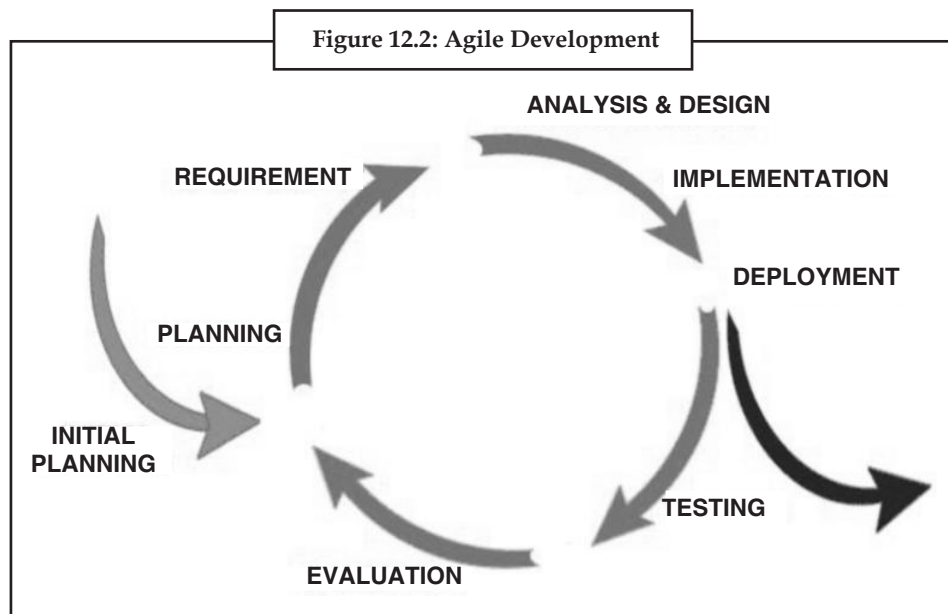
Give a spiral model representation of SDLC.

12.4 Iterative and Incremental Development

Iterative development prescribes the construction of initially small but ever larger portions of a software project to help all those involved to uncover important issues early before problems or faulty assumptions can lead to disaster. Iterative processes are preferred by commercial developers because it allows a potential of reaching the design goals of a customer who does not know how to define what they want.

12.4.1 Agile Development

Agile software development uses iterative development as a basis but advocates a lighter and more people-centric viewpoint than traditional approaches. Agile processes use feedback, rather than planning, as their primary control mechanism. The feedback is driven by regular tests and releases of the evolving software.



There are many variations of agile processes:

- In Extreme Programming (XP), the phases are carried out in extremely small (or “continuous”) steps compared to the older, “batch” processes. The (intentionally incomplete) first pass through the steps might take a day or a week, rather than the months or years of each complete step in the Waterfall model. First, one writes automated tests, to provide concrete goals for development. Next is coding (by a pair of programmers), which is complete when all the tests pass, and the programmers can’t think of any more tests that are needed. Design and architecture emerge out of refactoring, and come after coding. Design is done by the same people who do the coding. (Only the last feature - merging design and code - is common to all the other agile processes.) The incomplete but functional system is deployed or demonstrated for (some subset of) the users (at least one of which is on the development team). At this point, the practitioners start again on writing tests for the next most important part of the system.
- Scrum.

12.5 Process Improvement Models

Capability Maturity Model Integration

The Capability Maturity Model Integration (CMMI) is one of the leading models and based on best practice. Independent assessments grade organizations on how well they follow their defined processes, not on the quality of those processes or the software produced. CMMI has replaced CMM.

ISO 9000

ISO 9000 describes standards for a formally organized process to manufacture a product and the methods of managing and monitoring progress. Although the standard was originally created for the manufacturing sector, ISO 9000 standards have been applied to software development as well. Like CMMI, certification with ISO 9000 does not guarantee the quality of the end result, only that formalized business processes have been followed.

Notes

ISO 15504

ISO 15504, also known as Software Process Improvement Capability Determination (SPICE), is a “framework for the assessment of software processes”. This standard is aimed at setting out a clear model for process comparison. SPICE is used much like CMMI. It models processes to manage, control, guide and monitor software development. This model is then used to measure what a development organization or project team actually does during software development. This information is analyzed to identify weaknesses and drive improvement. It also identifies strengths that can be continued or integrated into common practice for that organization or team.

12.5.1 Formal Methods

Formal methods are mathematical approaches to solving software (and hardware) problems at the requirements, specification and design levels. Examples of formal methods include the B-Method, Petri nets, Automated theorem proving, RAISE and VDM. Various formal specification notations are available, such as the Z notation. More generally, automata theory can be used to build up and validate application behavior by designing a system of finite state machines.

Finite state machine (FSM) based methodologies allow executable software specification and bypassing of conventional coding (see virtual finite state machine or event driven finite state machine). Formal methods are most likely to be applied in avionics software, particularly where the software is safety critical. Software safety assurance standards, such as DO178B demand formal methods at the highest level of categorization (Level A). Formalization of software development is creeping in, in other places, with the application of Object Constraint Language (and specializations such as Java Modeling Language) and especially with Model-driven architecture allowing execution of designs, if not specifications.

Another emerging trend in software development is to write a specification in some form of logic (usually a variation of FOL), and then to directly execute the logic as though it were a program. The OWL language, based on Description Logic, is an example. There is also work on mapping some version of English (or another natural language) automatically to and from logic, and executing the logic directly. Examples are Attempto Controlled English, and Internet Business Logic, which does not seek to control the vocabulary or syntax. A feature of systems that support bidirectional English-logic mapping and direct execution of the logic is that they can be made to explain their results, in English, at the business or scientific level.

The Government Accountability Office, in a 2003 report on one of the Federal Aviation Administration’s air traffic control modernization programs, recommends following the agency’s guidance for managing major acquisition systems by

- establishing, maintaining, and controlling an accurate, valid, and current performance measurement baseline, which would include negotiating all authorized, unpriced work within 3 months;
- conducting an integrated baseline review of any major contract modifications within 6 months; and
- preparing a rigorous life-cycle cost estimate, including a risk assessment, in accordance with the Acquisition System Toolset’s guidance and identifying the level of uncertainty inherent in the estimate.

12.6 Summary

- System development life cycle is a process of creating or altering systems, and the models and methodologies that people use to develop these system.
- The waterfall model is a sequential software development model in which development is seen as flowing steadily downwards through several phases.
- A software development activity is a structure imposed on the development of a software product.
- The spiral model is intended for large, expensive and complicated projects.
- Process improvement is a series of actions taken by a process owner to identify, analyze and improve existing processes with in on organization to meet new goals and objectives.

12.7 Keywords

A software development process: It is also known as a *software development lifecycle*, is a structure imposed on the development of a software product. Similar terms include software life cycle and software process.

Agile development: Agile software development uses iterative development as a basis but advocates a lighter and more people-centric viewpoint than traditional approaches.

Capability maturity model integration: The Capability Maturity Model Integration (CMMI) is one of the leading models and based on best practice.

Finite state machine (FSM): Its based methodologies allow executable software specification and by-passing of conventional coding (see virtual finite state machine or event driven finite state machine).

Software development models: Several models exist to streamline the development process. Each one has its pros and cons, and it's up to the development team to adopt the most appropriate one for the project. Sometimes a combination of the models may be more suitable.

Spiral model: The key characteristic of a Spiral model is risk management at regular stages in the development cycle.

Water fall model: The waterfall model shows a process, where developers are to follow these phases in order

12.8 Review Questions

1. What are execution models?
2. Define System Development Life Cycle.
3. Define Waterfall Model.
4. Define Spiral Model.
5. What is Agile development?
6. Briefly explain Process Improvement Models.

Notes

12.9 Further Reading



Books

Maran illustrated Computers Guided Tour, by Ruth Maran; Kelleigh Johnson,
Publisher: Course Technology PTR



Online link

<http://www.alternatives.rzero.com/lang.html>

Unit 13: Understanding the Need of Security Measures

Notes

CONTENTS

Objectives

Introduction

13.1 Basic Security Concepts

13.1.1 Technical Areas

13.1.2 Security is Spherical

13.1.3 The Need For Security

13.1.4 Security Threats, Attacks and Vulnerabilities

13.1.5 Security Threats

13.2 Threats to Users

13.2.1 Viruses: One of the Most Common Computer Threats

13.2.2 Trojans: The Sneaky Computer Threats

13.2.3 Worms: The Self-replicating Computer Threats

13.2.4 Spyware: Annoying Threats to your Computer

13.2.5 Problems Caused by Common Computer Threats

13.2.6 Protection for Users

13.3 Threats to Hardware

13.3.1 Power Faults

13.3.2 Incompatibilities

13.3.3 Finger Faults

13.3.4 Malicious or Careless Damage

13.3.5 Typhoid Mary

13.3.6 Magnetic Zaps

13.3.7 Bottom Line

13.4 Threat to Data

13.4.1 Main Source

13.4.2 Data Protection

13.5 Cyber Terrorism

13.5.1 Protection against Cyber Terrorism

13.6 Summary

13.7 Keywords

13.8 Self-Assessment Questions

13.9 Review Questions

13.10 Further Reading

Notes

Objectives

After studying this unit, you will be able to:

- Discussed the basic security concepts.
- Explained threats to users.
- Understand threats to hardware.
- Understand threats to data.
- Explained cyber terrorism.

Introduction

The term computer security is used frequently, but the content of a computer is vulnerable to few risks unless the computer is connected to other computers on a network. As the use of computer networks, especially the Internet, has become pervasive, the concept of computer security has expanded to denote issues pertaining to the networked use of computers and their resources. Threat is defined as a computer program, a person, or an event that violates the security system. A threat causes loss of data and attacks the data privacy. Cyber terrorism describe the use of Internet based attacks in terrorist activities, including acts of deliberate, large-scale disruption of computer networks, especially of personal computers attached to the Internet, by the means of tools such as computer viruses.

13.1 Basic Security Concepts

Computer security means to protect information. It deals with the prevention and detection of unauthorized actions by users of a computer. Lately it has been extended to include privacy, confidentiality and integrity.

This unit provides an overview of security concepts, focusing on the following areas:

- Application-Level Security
- Transport-Level Security

These are two basic categories of security that can be independently configured but are often interrelated. The former mostly determines who can access data and what tasks they are allowed to perform; the latter mostly determines the security of data as it is transmitted.

Note that application-level configuration can include transport-level specifications, such as having an application-level constraint requiring Secure Sockets Layer and transport-level security can also involve authentication (limiting data access to appropriate users), such as when client certification is requested as part of the transport-level functionality.

13.1.1 Technical Areas

The major technical areas of computer security are usually represented by the initials CIA: confidentiality, integrity, and authentication or availability. Confidentiality means that information cannot be access by unauthorized parties. Confidentiality is also known as secrecy or privacy; breaches of confidentiality range from the embarrassing to the disastrous. Integrity means that information is protected against unauthorized changes that are not detectable to authorized users; many incidents of hacking compromise the integrity of databases and other resources. Authentication means that users are who they claim to be. Availability means that resources are accessible by authorized parties; “denial of service” attacks, which are sometimes the topic of

national news, are attacks against availability. Other important concerns of computer security professionals are access control and non-repudiation. Maintaining access control means not only that users can access only those resources and services to which they are entitled, but also that they are not denied resources that they legitimately can expect to access. Non-repudiation implies that a person who sends a message cannot deny that he sent it and, conversely, that a person who has received a message cannot deny that he received it. In addition to these technical aspects, the conceptual reach of computer security is broad and multifaceted. Computer security touches draws from disciplines as ethics and risk analysis, and is concerned with topics such as computer crime; the prevention, detection, and remediation of attacks; and identity and anonymity in cyberspace.

While confidentiality, integrity, and authenticity are the most important concerns of a computer security manager, privacy is perhaps the most important aspect of computer security for everyday Internet users. Although users may feel that they have nothing to hide when they are registering with an Internet site or service, privacy on the Internet is about protecting one's personal information, even if the information does not seem sensitive. Because of the ease with which information in electronic format can be shared among companies, and because small pieces of related information from different sources can be easily linked together to form a composite of, for example, a person's information seeking habits, it is now very important that individuals are able to maintain control over what information is collected about them, how it is used, who may use it, and what purpose it is used for.

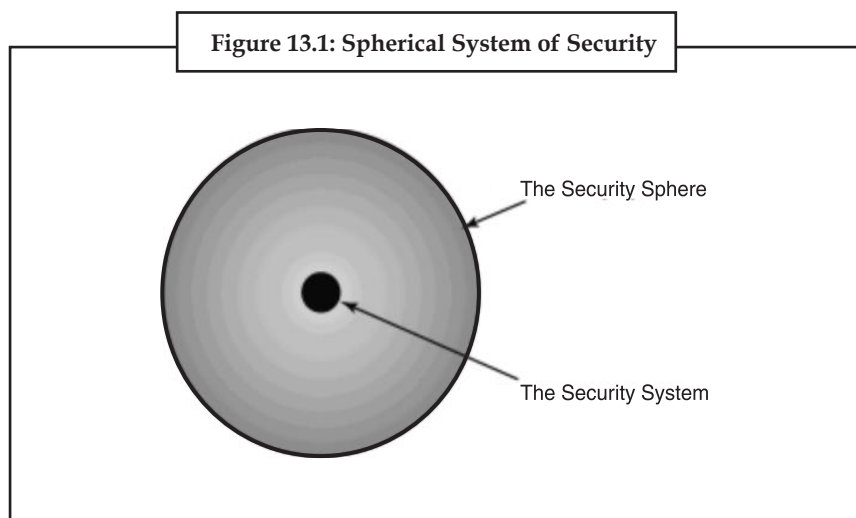
13.1.2 Security is Spherical

Computer systems can never have absolute security in real life. They exist to be used; not to be admired in a locked room sealed away from the outside world. Systems can, however, be made more secure than they would be otherwise. Let's see how we can conceptualize this.

Security is spherical, but has markers

Threats to a system can originate from any source, not just the ones that you have considered or defended against. Think of the threat universe as a sphere around the target, each incoming threat made up of the results of many different vector components. Like a color wheel, it gradiates as the radius increases.

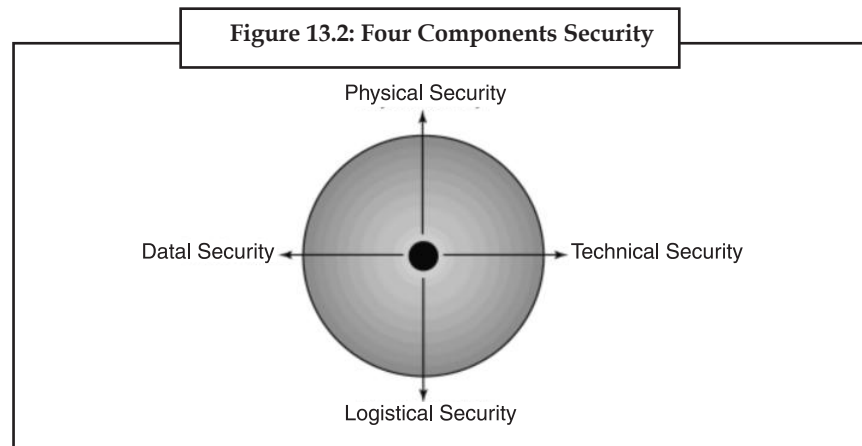
Think of the system at the center of a sphere made up of hostile intentions. Let's cut a circular plane out of the sphere in the middle of it (Figure 13.1).



Notes

Let's then mark four orthogonal vectors like the main points on a compass, except that they point to four security concepts.

These concepts are physical security, logistical security, data security and technical security (Figure 13.2).



Each concept by itself is only a part of the overall solution to the risk management problem. Combined in the proportions necessary for the job at hand, they can have a powerfully deflective effect.



Task

Explain four components security.

13.1.3 The Need For Security

Administrators normally find that putting together a security policy that restricts both users and attacks is time consuming and costly. Users also become disgruntled at the heavy security policies making their work difficult for no discernable reason, causing bad politics within the company. Planning an audit policy on huge networks takes up both server resources and time, and often administrators take no note of the audited events. A common attitude among users is that if no secret work is being performed, why bother implementing security.

There is a price to pay when a half-hearted security plan is put into action. It can result in unexpected disaster. A password policy that allows users to use blank or weak passwords is a hacker's paradise. No firewall or proxy protection between the organization's private local area network (LAN) and the public Internet makes the company a target for cyber crime.

Organizations will need to determine the price they are willing to pay in order to protect data and other assets. This cost must be weighed against the costs of losing information and hardware and disrupting services. The idea is to find the correct balance. If the data needs minimal protection and the loss of that data is not going to cost the company, then the cost of protecting that data will be less. If the data is sensitive and needs maximum protection, then the opposite is normally true.

13.1.4 Security Threats, Attacks and Vulnerabilities

Information is the key asset in most organizations. Companies gain a competitive advantage by knowing how to use that information. The threat comes from others who would like to acquire the information or limit business opportunities by interfering with normal business processes.

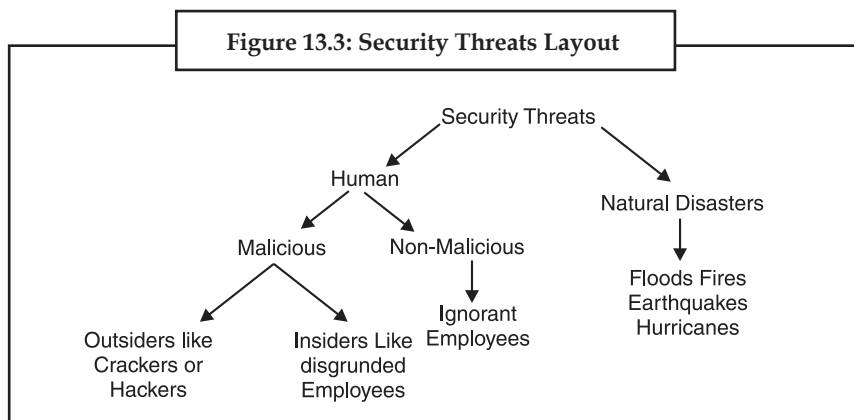
Notes

The object of security is to protect valuable or sensitive organizational information while making it readily available. Attackers trying to harm a system or disrupt normal business operations exploit vulnerabilities by using various techniques, methods, and tools. System administrators need to understand the various aspects of security to develop measures and policies to protect assets and limit their vulnerabilities.

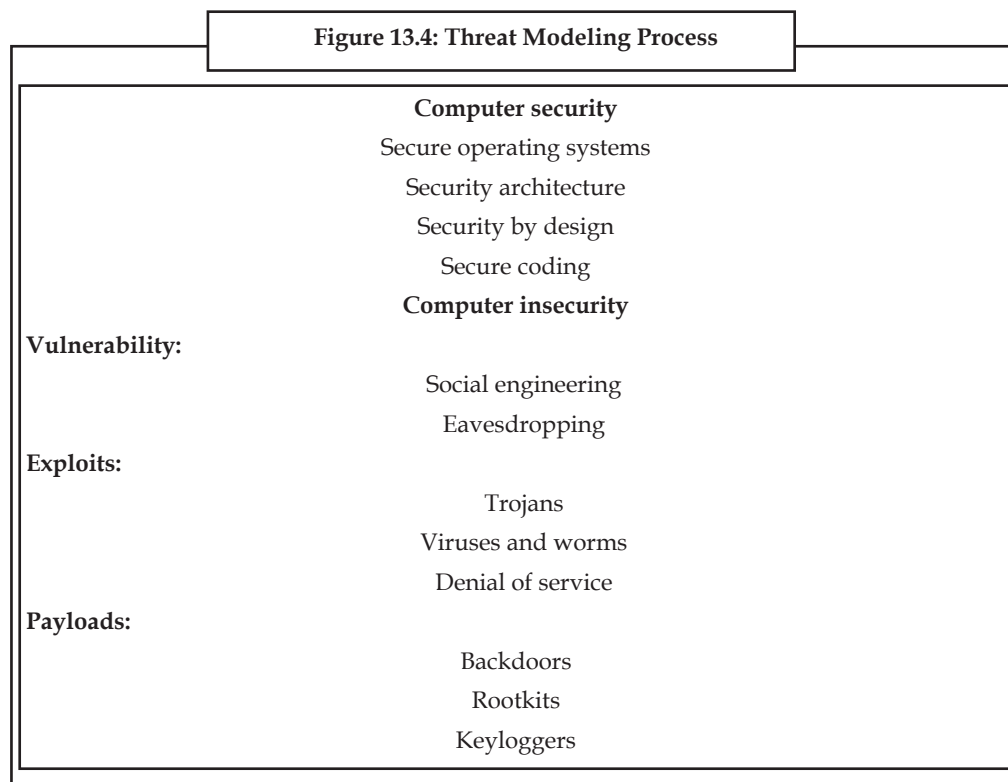
Attackers generally have motives or goals for example, to disrupt normal business operations or steal information. To achieve these motives or goals, they use various methods, tools, and techniques to exploit vulnerabilities in a computer system or security policy and controls.

13.1.5 Security Threats

Figure 13.3 introduces a layout that can be used to break up security threats into different areas.



Tabular Representation:



Notes



Computer security is critical in almost any technology-driven industry which operates on computer systems. Computer security can also be referred to as computer safety. The issues of computer based systems and addressing their countless vulnerabilities are an integral part of maintaining an operational industry.

13.1.5.1 Cloud Computing Security

Security in the cloud is challenging, due to varied degree of security features and management schemes within the cloud entities. In this connection one logical protocol base need to evolve so that the entire gamut of components operates synchronously and securely.

13.1.5.2 In Aviation

The aviation industry is especially important when analyzing computer security because the involved risks include human life, expensive equipment, cargo, and transportation infrastructure. Security can be compromised by hardware and software malpractice, human error, and faulty operating environments. Threats that exploit computer vulnerabilities can stem from sabotage, espionage, industrial competition, terrorist attack, mechanical malfunction, and human error.

The consequences of a successful deliberate or inadvertent misuse of a computer system in the aviation industry range from loss of confidentiality to loss of system integrity, which may lead to more serious concerns such as data theft or loss, network and air traffic control outages, which in turn can lead to airport closures, loss of aircraft, loss of passenger life. Military systems that control munitions can pose an even greater risk.

A proper attack does not need to be very high tech or well funded; for a power outage at an airport alone can cause repercussions worldwide. One of the easiest and, arguably, the most difficult to trace security vulnerabilities is achievable by transmitting unauthorized communications over specific radio frequencies. These transmissions may spoof air traffic controllers or simply disrupt communications altogether. These incidents are very common, having altered flight courses of commercial aircraft and caused panic and confusion in the past. Controlling aircraft over oceans is especially dangerous because radar surveillance only extends 175 to 225 miles offshore. Beyond the radar's sight controllers must rely on periodic radio communications with a third party.

Lightning, power fluctuations, surges, brown-outs, blown fuses, and various other power outages instantly disable all computer systems, since they are dependent on an electrical source. Other accidental and intentional faults have caused significant disruption of safety critical systems throughout the last few decades and dependence on reliable communication and electrical power only jeopardizes computer safety.

13.1.5.3 Notable System Accidents

In 1994, over a hundred intrusions were made by unidentified crackers into the Rome Laboratory, the US Air Force's main command and research facility. Using trojan horse viruses, hackers were able to obtain unrestricted access to Rome's networking systems and remove traces of their activities. The intruders were able to obtain classified files, such as air tasking order systems data and furthermore able to penetrate connected networks of National Aeronautics and Space Administration's Goddard Space Flight Center, Wright-Patterson Air Force Base, some Defense contractors, and other private sector organizations, by posing as a trusted Rome center user.

13.2 Threats to Users

13.2.1 Viruses: One of the Most Common Computer Threats

A computer virus is a small program that can replicate itself to infect computers. They were a problem even in the beginning phases of the internet. The first known virus was the Creeper virus, which spread via ARPANET, a progenitor of the internet, in the early 1970s. Today, there are literally thousands of viruses. Not all of them are particularly troublesome, but some of them can cause significant damage. Computer viruses are considered as one of the most well known computer security threat. This is a program which is written to alter the way a computer operates. And it is obviously without the permission or knowledge of the PC user. A virus replicates and executes itself and damages the files and folders of the computer.

There are several ways to contract these computer threats. They are commonly found in email attachments and downloads from malicious web sites.

13.2.2 Trojans: The Sneaky Computer Threats

If you use an operating system that was released within the past couple years, then you might occasionally see warning messages about Trojans when you download files. Trojans are small viruses that hide within other programs. They can essentially exist anywhere. Funny pictures, illegal downloads, and pirated software are some of the most common sources.

13.2.3 Worms: The Self-replicating Computer Threats

Worms are similar to viruses in that they are small, malicious programs. The big difference, though, is that worms pose a threat even when computer users don't download any files. These programs use the internet to search for vulnerable computers. Once they locate them, they move in. The computer users probably won't even recognize that anything has happened until he or she begins to experience computer problems.

13.2.4 Spyware: Annoying Threats to your Computer

Spyware is another serious computer security threat. Spyware might seem like viruses, but they don't replicate themselves. This technicality will mean little, though, to those who unintentionally download spyware to their computers. This common threat is usually contracted through peer-to-peer file sharing. If you download music, movies, or software illegally, then you have an increased risk of installing spyware. Once installed, these programs start monitoring your online activities. They also have potential to install programs without the consent of the user and capture personal information. To combat spyware threats and stay safe online, it is suggested to consult a spyware removal service provider.

13.2.5 Problems Caused by Common Computer Threats

These computer threats can cause lots of problems. Viruses, worms, and spyware are often designed to gather information. This means that criminals can get private information such as bank account numbers, passwords, and credit card numbers. This makes it much easier for them to commit fraud and identity theft: two of the scariest problems that modern internet users face.

Once criminals have this info, they can take money from your bank accounts, sign up for credit cards in your name, or even purchase high ticket items from online retailers.

Other viruses take over small parts of your computer. This allows people to use your computer to send spam emails or participate in guerilla techniques that bombard web sites with requests until

Notes

their owners pay off the criminal to regain control of his or her site. Chances are that you won't see these activities taking place. Your computer or internet connection, however, will usually slow down: a symptom that a computer threat is causing problems with unknown repercussions.

There are varieties of malicious objects online and Malware is one of them. You will be astonished to know that malware was rated as the second highest ranked threat to the businesses by Perimeter E-Security. Hackers and cyber criminals use varieties of methods to install malware on the user's computer. This is used to break into computer systems of the user to steal, change or destroy information. If proper measures are not taken your bank account details and other important information like credit card details, etc. could be misused by them.

To offer you sleepless nights, hackers use wide selection of malicious objects and Trojan horses are commonly used. Unlike viruses Trojans it do not replicate and spread like a virus. These programs come with the disguise of pictures and PowerPoint presentations. Once the program is run, a virus is placed on your PC to allow a hacker to gain access to your computer. These types of Trojans are called Remote Access Trojans (RATs). You will be amazed to know that more than 50% of all spam (unsolicited email) is sent from home or work computers are compromised by RATs.

Now the million dollar question is how you can protect your computer as well as your online identity and crucial information from all these malicious objects. Well, there are plenty of options to combat all these. Firstly, you can install antivirus software, firewalls, etc. If you are not familiar with all these, you can take help from a remote computer support providers.

13.2.6 Protection for Users

One of the first, and most basic measures, that you should take to protect your sensitive data and computer resources is placing an anti-virus program on your system and allowing it to run at all times. This type of software works to detect any and all types of software that is deemed malicious. If malicious types of software do, in fact, invade your computer system, this software will work to remove it and all threats that it pose. Viruses can be quite destructive when it comes to the home computer system. They can interfere with the basic and higher level functions of the system, corrupt files and other forms of data, and even spread throughout the system and result in instabilities.

The next step to basic computer security for the home user is to turn on the firewall that is built in to the operating system. This is a program that works to prevent dangerous intrusions from hackers, different types of worms, and even viruses in the computer system. The firewall should always be used in conjunction with the anti-virus program that you have on your computer to optimize the protection that your system receives. In addition to software firewalls, there are also hardware firewalls that you can purchase to increase the effectiveness of the amount of protection that your computer receives.

The third step to increasing the security of your home computer if you have a Windows based operating system is to keep up with all the Windows Updates that are available for your system. Microsoft developers work on a daily basis to create software updates and patches that allow Windows to run smoothly and effectively, without the threat of security infringement. It is absolutely necessary to ensure that you check for these updates on a regular basis. To make this easier, Windows has a feature that allows you to turn on automatic updates. This means that your computer will automatically download and install these updates at a time that you specify. As a home computer user, it is important to take advantage of this feature to ensure computer security on your home system.

As you can see, there are many different steps that can be taken to ensure computer security. If you are a home user, it is especially important to take full advantage of the features that are listed here. Not only will these computer security steps protect your computer system as a whole, but they will also protect your personal information.

13.3 Threats to Hardware

Hardware problems are all too common. We all know that when a PC or disk gets old, it might start acting erratically and damage some data before it totally dies. Unfortunately, hardware errors frequently damage data on even young PCs and disks.

Here are some examples.

13.3.1 Power Faults

Your PC is busy writing data to the disk and the lights go out! “Arghhhh!” Is everything OK? Maybe so, maybe not; it’s vital to know for sure if anything was damaged.

Other power problems of a similar nature would include brownouts, voltage spikes, and frequency shifts. All can cause data problems, particularly if they occur when data is being written to disk (data in memory generally does not get corrupted by power problems; it just gets erased if the problems are serious enough).

- **Brownout:** Lower voltages at electrical outlets. Usually they are caused by an extraordinary drain on the power system. Frequently you will see a brownout during a heat wave when more people than normal have air conditioners on full. Sometimes these power shortages will be “rolling” across the area giving everyone a temporary brownout. Maybe you’ll get yours just as that important file is being written to disk.
- **Voltage Spikes:** Temporary voltage increases are fairly common. Large motors or circuit breakers in industry can put them on the electrical line. Sudden losses (e.g., a driver hits a power pole) can cause spikes as the circuits balance. An appliance in your home can cause a spike, particularly with older wiring. Lightning can put large spikes on power lines. And, the list goes on. In addition to current backups and integrity information for your software and data files, including a hardware voltage spike protection device between the wall and your computer hardware (don’t forget the printer and monitor) can be very helpful.
- **Frequency Shifts:** While infrequent, if the line frequency varies from the normal 60 Hertz (or 50 Hertz in some countries), the power supply on the computer can be affected and this, in turn, can reflect back into the computer causing data loss.

13.3.2 Incompatibilities

You can have hardware problems on a perfectly healthy PC if you have devices installed that do not properly share interrupts. Sometimes problems are immediately obvious, other times they are subtle and depend upon certain events to happen at just the wrong time, then suddenly strange things happen! (Software can do this too!)

Solution: Make a really good backup before installing anything (hardware or software) so you can revert the system back to a stable state should something crop up.

13.3.3 Finger Faults

These are an all too frequent cause of data corruption. This commonly happens when you are intending to delete or replace one file but actually get another. By using wild cards, you may experience a really “wild” time. “Hmmm I thought I deleted all the .BAK files; but they’re still here; something was deleted; what was it? Or was I in the other directory?” Of course if you’re a programmer or if you use sophisticated tools like a sector editor, then your fingers can really get you into trouble!

Notes

Another finger fault problem arises with touchpads below the space bar on notebook computers. It's very easy to brush the touchpad when you are typing away and suddenly find yourself entering characters in a screen location very different from where you were before you touched the pad.

Solution: Be careful and look up now and again to make certain your cursor is where you want it.

13.3.4 Malicious or Careless Damage

Someone may accidentally or deliberately delete or change a file on your PC when you're not around. If you don't keep your PC locked in a safe, then this is a risk. Who knows what was changed or deleted? Wouldn't it be nice to know if anything changed over the weekend? Most of this type of damage is done unintentionally by someone you probably know. This person didn't mean to cause trouble; they simply didn't know what they were doing when they used your PC.

Solution: Never run the computer as an administrative user and have guest accounts available for others who use the computer. Keep up-to-date backups as well.

13.3.5 Typhoid Mary

One possible source for computer infections is the Customer Engineer (CE), or repairman. When a CE comes for a service call, they will almost always run a diagnostic program from diskette. It's very easy for these diskettes to become infected and spread the infection to your computer. Sales representatives showing demonstrations via floppy disks are also possibly spreading viruses. Always check your system after other people have placed their floppy disk into it. (Better yet, if you can, check their disk with up-to-date anti-virus software before anything is run.)

Solution: Insist on testing their disk before use or make certain they've used an up-to-date anti-virus before coming to your location.

13.3.6 Magnetic Zaps

Computer data is generally stored as a series of magnetic changes on disks. While hard disks are generally safe from most magnetic threats because they are encased within the computer compartment, floppy disks are highly vulnerable to magnets. The obvious threat would be to post a floppy disk to the refrigerator with a magnet; but there are many other, more subtle, threats.

Some of the more subtle sources of magnetism include:

- (a) **Computer Monitor.** Don't put floppy disks anywhere near the monitor; it generates a magnetic field.
- (b) **Telephone.** When ringing, telephones (particularly older phones with a bell) generate a magnetic field.
- (c) **Bottom Desk Drawer.** While the desk drawer does not generate a magnetic field, the vacuum cleaner that the maintenance people slide under the desk to clean the floor does.
- (d) **Bottom Bookcase Shelf and File Cabinet Drawer.** Same comment as the desk drawer just above.
- (e) **Pets.** Pet fur generates a strong electrostatic charge which, if discharged through a disk, can affect files on the disk. Instead of "The dog ate my homework," today it could just as easily be: "The cat sat on my homework." (I once had a student where this exact problem happened; a cat sat on her floppy disk and static wiped out the data on the disk.)

Solution: Stay away from magnets or sources of static of all kinds when working with a computer.

13.3.7 Bottom Line

There are tools to assist in recovery from disk problems, but how do you know all the data is OK? These tools do not always recover good copies of the original files. Active action on your part before disaster strikes is your best defense. It's best to have a good, current backup and, for better protection, a complete up-to-date integrity-check map of everything on your disk.



Did u know?

Hardware attackers' goals are usually tied to the IC's applications. Most goals can be classified into one or more of three categories:

- (a) Information leakage attackers extract information directly from an IC, passively or actively, as an individual component, and/or as a deployed element of an integrated system. Information to be protected includes the IP associated with a chipset and its design, data associated with both the hardware and deployed software, and data embedded or downloaded to the IC either prior to or during operation
- (b) Tampering attackers eavesdrop on or modify the data associated with the IC once it is deployed in operation, independently or as part of an integrated subsystem, by prolonged inspection and monitoring; and
- (c) Denial of service attackers modify the internal circuit structure of an IC to cause the circuit to malfunction or shut down under certain operating conditions.

13.4 Threat to Data

Threat is defined as a computer program, a person, or an event that violates the security system. A threat causes loss of data and attacks the data privacy. Most of the data of an organization stored inside the computer is very important and more valuable than the computer hardware and software. It can be damaged due to many reasons. You must protect your data from illegal access or from damage.

13.4.1 Main Source

The following are the main threats to data security.

- (a) Some authorized user of the data may unintentionally delete or change sensitive data. There are two solutions to this problem.
- (b) Firstly, the users must be assigned proper rights to minimize such events. Only the authorized user with certain rights may be allowed to delete or modify data after following a step-by-step process.
- (c) Secondly, periodic backup of data should be taken to recover the deleted data.
- (d) A proper password protection should be used to use any resource. A log file should also be maintained to keep track of all the activities performed on the data.
- (e) Some strong encryption algorithm should be used, so that if anyone gets access to the data, he could not be able to make any sense out of it.
- (f) Latest antivirus software should be used to scan all data coming into the organization.
- (g) Computers and all backing storage devices should be placed in locked rooms. Only authorized users can access these resources.
- (h) Authorized users must be asked to change their passwords periodically.

Notes

13.4.2 Data Protection

Data protection means making sure that private data belonging to a person or organization is kept hidden from those who are not authorized to use it.

Many organization collect data of their customers. Some of this data is needed for efficient processing the business transactions but much of this data is personal information of the customer. For example, a hospital collects data about the disease history of patients. All the personal data kept by different organizations may be disclosed for some legal purpose. The data protection rules do not allow anyone to disclose personal data of any person. It means that any personal data kept by some organization should never be disclosed to unauthorized persons or organizations under any circumstances.

Caution: Data protection and Threat to data.

13.5 Cyber Terrorism

Cyber terrorism is a phrase used to describe the use of Internet based attacks in terrorist activities, including acts of deliberate, large-scale disruption of computer networks, especially of personal computers attached to the Internet, by the means of tools such as computer viruses.

Cyber terrorism is a controversial term. Some authors choose a very narrow definition, relating to deployments, by known terrorist organizations, of disruption attacks against information systems for the primary purpose of creating alarm and panic. By this narrow definition, it is difficult to identify any instances of cyber terrorism.

Cyber terrorism can also be defined much more generally as any computer crime targeting computer networks without necessarily affecting real world infrastructure, property, or lives.

There is much concern from government and media sources about potential damages that could be caused by cyber terrorism, and this has prompted official responses from government agencies.

Cyber terrorism is defined by the Technolytics Institute as “The premeditated use of disruptive activities, or the threat thereof, against computers and/or networks, with the intention to cause harm or further social, ideological, religious, political or similar objectives. Or to intimidate any person in furtherance of such objectives.” The term was coined by Barry C. Collin.

The National Conference of State Legislatures, an organization of legislators created to help policymakers issues such as economy and homeland security defines cyber terrorism as:

“The use of information technology by terrorist groups and individuals to further their agenda. This can include use of information technology to organize and execute attacks against networks, computer systems and telecommunications infrastructures, or for exchanging information or making threats electronically. Examples are hacking into computer systems, introducing viruses to vulnerable networks, web site defacing, Denial-of-service attacks, or terroristic threats made via electronic communication.

Cyber terrorism can also include attacks on Internet business, but when this is done for economic motivations rather than ideological, it is typically regarded as cybercrime.

As shown above, there are multiple definitions of cyber terrorism and most are overly broad. There is controversy concerning overuse of the term and hyperbole in the media and by security vendors trying to sell “solutions”.

Cyber terrorism is the union of terrorism and the cyberspace. It refers to the use of computer networks to threaten the masses. It involves the utilization of computer infrastructure to terrorize society. According to some, cyber terrorism is the use of Internet by terrorist organizations to spread fear while others refer to it as the disruption of sensitive information stored on the web, thus causing loss of sensitive data. The disruption of information stored on computer networks is also a form of cyber terrorism.

Electronic means cannot cause physical harm. They can spread fear among the common masses but they are very unlikely to result in a massive physical destruction or death. Considering the advanced Internet security measures and the protective technologies in use, there are lesser chances of the Internet leading to terror attacks. Many computer theorists deny the existence of cyber terrorism. They prefer to call it as hacking or information warfare, wherein confidential information of individuals or organizations is put on stake by unethical Internet users. They define cyber terrorism as the group of activities that risk Internet safety of individuals and organizations. However, many consider cyber terrorism as a serious threat to society. It is not right to underestimate the devastating effects of criminal practices like hacking and phishing. Hacking is the activity of breaking into a computer system in order to gain an unauthorized access to it. The unauthorized revelation of passwords, the hacking of IP addresses can prove being severe threats to the well-being of society. There have been instances of terrorist agencies hacking computer systems to gain access to sensitive and critical information. One example is of terrorists in Romania gaining access to the computers controlling the life support systems at the Antarctic research station. A relatively recent example is of the website of the Ukrainian president being attacked by hackers. Another simple example of cyber terrorism could be the hacking of a hospital database system and changing medical prescriptions of the patients. Isn't that scary?

13.5.1 Protection against Cyber Terrorism

In order to take effective measures against cyber terrorism, it is important for the computer professionals to understand its adverse effects and be able to identify the loopholes in Internet security. It is necessary to strengthen Internet security measures so that the society can be assured a safe life on the Internet.

Critical information should be isolated from the outside world. It should be protected by the means of firewalls, antivirus software and complex password systems. Government organizations should be well-equipped to deal with cyber-terrorist activities. Similarly, banks, financial organizations and other critical information units should be well protected from practices such as phishing, hacking and identity theft. The information pertaining to national security is of highest importance for the people of any country. It needs to be effectively protected from unethical Internet users.

It is also important to ensure protection on an individual level. It is essential for all the Internet users to ensure Internet safety. In order to ensure protection from cyber terrorism, one's email accounts should be protected through passwords that are not easily guessable. One should change the network configurations if defects are known and pay heed to network security issues on an urgent basis. One should not visit websites that can seem suspicious in order to keep away from Internet scams. It is important for every Internet user to know about the different types of computer crimes and take due precaution against all the ill-practices that eclipse the advantages of the Internet.

Notes



Computer Security website Kaspersky Secures Top Connectivity with Dell



Kaspersky, based in Russia, is a computer security company. The website offers a comprehensive one-stop shop to help customers research and buy a wide range of software, downloads and updates to suit their requirements.

Kaspersky's website acts as an e-commerce portal for its products, as well as offering a range of downloads, including free trials, a virus scan and virus removal tools. This means the demand on the site is always extremely high as many users also download product updates simultaneously, which can put a strain on the site.

Kaspersky required a hosting company with superior connectivity and bandwidth to support the constant demands on the Kaspersky website for files and virus updates.

To ensure that all downloads on the site were always available, a quality Dell server was provisioned by LeaseWeb, equipped with 10 Gbps dedicated fibre port capable of pushing large amounts of traffic to many Kaspersky customers simultaneously. This large server, combined with LeaseWeb's superior network connectivity, was the key to meeting Kaspersky's requirements in a cost efficient, highly effective approach that satisfied all Kaspersky's needs.

Questions:

1. What is Kaspersky?
2. Why Kaspersky required?

13.6 Summary

- Computer security means to protect information, it has been extended to include privacy, confidentiality and integrity.
- Computer viruses are considered as one of the most well known computer security threat.
- Hardware threats involve threats of physical damage to the router or switch hardware.

- Data can be damaged due to many reasons. You must protect your data from illegal access or from damage.
- Cyber terrorism can be defined in different ways viz. it can be politically motivated hacking operations intended to cause grave harm such as loss of life or severe economic damage.

13.7 Keywords

Authentication: The process of verifying that users are who they claim to be when logging onto a system. Generally, the use of user names and passwords accomplishes this. More sophisticated is the use of smart cards and retina scanning. The process of authentication does not grant the user access rights to resources-this is achieved through the authorization process.

Availability: The prevention of unauthorized withholding of information or resources. This does not apply just to personnel withholding information. Information should be as freely available as possible to authorized users.

Brownout: Lower voltages at electrical outlets. Usually they are caused by an extraordinary drain on the power system.

Computer security: Computer security means to protect information. It deals with the prevention and detection of unauthorized actions by users of a computer.

Confidentiality: The prevention of unauthorized disclosure of information. This can be the result of poor security measures or information leaks by personnel. An example of poor security measures would be to allow anonymous access to sensitive information.

Cyber terrorism: It can be defined as any computer crime targeting computer networks without necessarily affecting real world infrastructure, property, or lives.

Data protection: It means making sure that private data belonging to a person or organization is kept hidden from those who are not authorized to use it.

Detection: Take measures that allow you to detect when information has been damaged, altered, or stolen, how it has been damaged, altered, or stolen, and who has caused the damage. Various tools are available to help detect intrusions, damage or alterations, and viruses.

Finger faults: These are an all too frequent cause of data corruption. This commonly happens when you are intending to delete or replace one file but actually get another.

Hacking: It is the activity of breaking into a computer system in order to gain an unauthorized access to it. The unauthorized revelation of passwords, the hacking of IP addresses can prove being severe threats to the well-being of society.

Integrity: The prevention of erroneous modification of information. Authorized users are probably the biggest cause of errors and omissions and the alteration of data. Storing incorrect data within the system can be as bad as losing data. Malicious attackers also can modify, delete, or corrupt information that is vital to the correct operation of business functions.

Prevention: Take measures that prevent your information from being damaged, altered, or stolen. Preventive measures can range from locking the server room door to setting up high-level security policies.

Reaction: Take measures that allow recovery of information, even if information is lost or damaged.

Notes

Threat: Any circumstance or event with the potential to harm an information system through unauthorized access, destruction, disclosure, modification of data, and/or denial of service. Threats arise from human actions and natural events.

Threat source: Either (1) intent and method targeted at the intentional exploitation of a vulnerability or (2) the situation and method that may accidentally trigger a vulnerability.

Threat analysis: The examination of threat sources against system vulnerabilities to determine the threats for a particular system in a particular operational environment.

Trojans: Trojans are small viruses that hide within other programs. They can essentially exist anywhere.

Virus: A computer virus is a small program that can replicate itself to infect computers. They were a problem even in the beginning phases of the internet.

Voltage spikes: Temporary voltage increases are fairly common. Large motors or circuit breakers in industry can put them on the electrical line.

Worms: Worms are similar to viruses in that they are small, malicious programs. The big difference, though, is that worms pose a threat even when computer users don't download any files.



Lab Exercise

1. Draw Spherical system of security.
2. Draw Security Threats layout.

13.8 Self-Assessment Questions

1. What are the major technical areas of computer security are usually represented by the initials CIA?
 - (a) Confidentiality
 - (b) Integrity
 - (c) Authentication
 - (d) All of these
2. A password policy that allows users to use blank or weak passwords is a hacker's paradise.
 - (a) True
 - (b) False
3. Security in the cloud is challenging, due to varied degree of
 - (a) Security features and arrangement
 - (b) Security features and management
 - (c) Security wall and management
 - (d) All of the above
4. The consequences of a successful deliberate or inadvertent misuse of a computer system in the aviation industry range from loss of system integrity to loss of confidentiality.
 - (a) True
 - (b) False
5. A computer virus is a small program that can replicate itself to infect computers.
 - (a) True
 - (b) False

Notes

6. What Trojans are called?
 - (a) RAM Access Trojans (RATs).
 - (b) ROM Access Trojans (RATs).
 - (c) Remote Access Trojans (RATs).
 - (d) None of the above
7. Computer infections source is the Customer Engineer (CE), or repairman.
 - (a) True
 - (b) False
8. What does Data protection means?
 - (a) Means making sure that private data belonging to a person or organization is kept hidden from those who are authorized to use it.
 - (b) Making sure that private data belonging to a person or organization is kept hidden from those who are not authorized to use it.
 - (c) making sure that organizational data belonging to a person or organization is kept hidden from those who are not authorized to use it.
 - (d) All of the above
9. Cyber terrorism can not include attacks on Internet business,
 - (a) True
 - (b) False
10. Hardware attackers' goals are usually tied to the IC's applications.
 - (a) True
 - (b) False

13.9 Review Questions

1. What are security issues related to computer hardware?
2. Elaborate the importance of security in an organization.
3. Define computer security and write down the major components of spherical security system.
4. What are viruses and enumerate and explain briefly about the related risk agents?
5. How important is hardware security and briefly explain the important risks associated with hardware threats?
6. Elaborate and explain about CIA.
7. What is cyber terrorism and why it is important from national welfare point of view?
8. What is electronic cyber terrorism?
9. Enumerate and explain various threats related to data security.
10. Enlist various security requirement parameters.
11. Define the following terms:
 - (a) Malware
 - (b) Spyware
 - (c) Hacker
 - (d) Trojan
12. Write the steps of protection for users.

Notes

Answers for Self-Assessment Questions

1. (d) 2. (a) 3. (b) 4. (b) 5. (a) 6. (c) 7. (a)
8. (b) 9. (b) 10. (a)

13.10 Further Reading



Books

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition



Online link

<http://www.net-security.org/article.php%3Fid>

Unit 14: Taking Protected Measures

Notes

CONTENTS

Objectives

Introduction

14.1 Keeping Your System Safe

14.1.1 Get Free Wireless Network Protection Software

14.1.2 Use a Free Firewall

14.1.3 Encrypt Your Data

14.1.4 Protect Yourself Against Phishers

14.1.5 Disable File Sharing

14.1.6 Surf the Web Anonymously

14.1.7 Say No to Cookies

14.1.8 Protect yourself against E-mail "Nigerian Scams"

14.1.9 Virus Scan

14.1.10 Kill Spyware

14.1.11 Stay Up-To-Date

14.1.12 Secure Your Mobile Connection

14.1.13 Don't Forget the Physical

14.2 Protect Yourself

14.3 Protect Your Privacy

14.3.1 Avoid Identity Theft

14.3.2 Identity Theft

14.3.3 Spying

14.4 Managing Cookies

14.4.1 Cookies

14.4.2 Internet Explorer

14.4.3 Mozilla Firefox

14.4.4 External Tools

14.5 Spyware and Other BUGS

14.5.1 Spyware

14.5.2 Other Web Bugs

14.6 Keeping your Data Secure

14.6.1 The Data Protection Act

Notes

14.7	Backing up Data
14.8	Safeguarding Your Computer Hardware
14.8.1	Physical Access and Data Security
14.9	Summary
14.10	Keywords
14.11	Self-Assessment Questions
14.12	Review Questions
14.13	Further Reading

Objectives

After studying this unit, you will be able to:

- Understand keeping your system safe.
- Explain protect yourself.
- Understand protect your privacy.
- Explain managing cookies.
- Discuss spyware and other bugs.
- Explain keeping your secure.
- Discuss backing up data.
- Explain safeguarding your computer hardware.

Introduction

Your home computer is a popular target for intruders. Why? Because intruders want what you've stored there. They look for credit card numbers, bank account information, and anything else they can find. By stealing that information, intruders can use your money to buy themselves goods and services. But it's not just money-related information they are after. Intruders also want your computer's resources, meaning your hard disk space, your fast processor, and your Internet connection. They use these resources to attack other computers on the Internet. In fact, the more computers an intruder uses, the harder it is for law enforcement to figure out where the attack is really coming from. If intruders can't be found, they can't be stopped, and they can't be prosecuted. Why are intruders paying attention to home computers? Home computers are typically not very secure and are easy to break into. When combined with high-speed Internet connections that are always turned on, intruders can quickly find and then attack home computers. While intruders also attack home computers connected to the Internet through dial-in connections, high-speed connections (cable modems and DSL modems) are a favorite target.

No matter how a home computer is connected to the Internet, intruders' attacks are often successful. Many home computer owners don't realize that they need to pay attention to computer security. In the same way that you are responsible for having insurance when you drive a car, you need to also be responsible for your home computer's security. This document explains how some parts of the Internet work and then describes tasks you can do to improve the security of your home computer system. The goal is to keep intruders and their programs off your computer.

How do intruders break into your computer? In some cases, they send you email with a virus. Reading that email activates the virus, creating an opening that intruders use to enter or access your computer. In other cases, they take advantage of a flaw or weakness in one of your computer's programs - vulnerability — to gain access.

Once they're on your computer, they often install new programs that let them continue to use your computer—even after you plug the holes they used to get onto your computer in the first place. These backdoors are usually cleverly disguised so that they blend in with the other programs running on your computer. The next section discusses concepts you need to know, especially trust. The main part of this document explains the specific issues that need your attention. There are examples of how to do some of these tasks to secure a Microsoft Windows 2000-based computer. We also provide checklists you can use to record information about the steps you have taken to secure your computer. Finally, a glossary defines many of the technical terms used in this document. Unless otherwise stated in the glossary, the definitions come from the Webopedia Online Dictionary for Computer and Internet Terms. Whether your computer runs Microsoft® Windows®, Apple's Mac OS, LINUX, or something else, the issues are the same and will remain so as new versions of your system are released. The key is to understand the security-related problems that you need to think about and solve.

14.1 Keeping Your System Safe

Before diving into the tasks you need to do to secure your home computer, let's first think about the problem by relating it to something you already know how to do. In this way, you can apply your experience to this new area.

So, think of your computer as you would your house, your apartment, or your condo. What do you know about how that living space works, what do you routinely do to keep it secure and what have you installed to improve its security? (We'll use this "computer-is-like-a-house-and-the-things-in-it" analogy throughout, departing only a few times to make a point.)

For example, you know that if you have a loud conversation, folks outside your space can probably hear you. You also routinely lock the doors and close the windows when you leave, and you don't give the keys to just anyone. Some of you may install a security system to complement your practices. All of these are part of living in your home.

Let's now apply similar thinking to your home computer. Email, instant messaging, and most web traffic go across the Internet in the clear; that is, anyone who can capture that information can read it. These are things you ought to know. You should always select and use strong passwords and exercise due care when reading all email, especially the unsolicited variety. These are things you ought to do. Finally, you can add a firewall, an anti-virus program, patches, and file encryption to improve the level of security on your home computer.



Task

Write down the steps involved in creating password for your computer.

The rest of this document describes the things you ought to know, do, and install to improve the security of your home computer.

On starting point for solving home computer security problems is being aware of how the Internet and some of its technologies work. If you know how they work, you can evaluate solutions to the problems that come up. You can also use the Internet more safely and responsibly. In this section, we'll talk about two topics: trust and information in the clear as it crosses the Internet.

Notes

14.1.1 Get Free Wireless Network Protection Software

Most home networks are vulnerable to passing “war drivers” who hack into unsuspecting wireless networks. There are plenty of ways you can muck around with your router settings to protect yourself.

But what if you don’t want to fiddle around with filtering MAC addresses, changing your SSID (network name), and disabling SSID broadcast? You can get a free program that will do most of that for you. Network Magic comes in two versions, a free version and a for-pay version, but if all you want to do is configure your wireless network for maximum security, the free version will work just fine.

Install it, and it examines your router and entire network, and builds a network map of all of your connected devices. It examines your router’s security settings and issues a report on what it finds. If, for example, it discovers that you’re broadcasting your SSID, it will alert you. A single click of a check box and Network Magic will stop the broadcasting for you. The for-pay version includes other features, such as configuring folder and printer sharing, but if you’re only interested in security, you don’t really need it.

Note that there’s not much this program can do that you can’t do on your own, if you’re willing to dig in and get your hands dirty. But if you’d like to keep them clean and still have a secure wireless network, you can’t do any worse than free.

14.1.2 Use a Free Firewall

It’s this simple you need a firewall. It’s one of the best ways to protect yourself against Trojans, to keep your PC from becoming a zombie that obeys the commands of a distant hacker, and to stop attackers from worming their way into your PC. If you have Windows XP Service Pack 2, you have a halfway useful firewall built in. (If you haven’t installed SP2, immediately upgrade by going to Windows Update.) By default, when you install SP2, the firewall is turned on. But if you suspect it’s accidentally been turned off, you can check by clicking the Security Center icon in the system tray. The Security Center screen will pop up. (If the Security Center icon doesn’t appear in your system tray or Taskbar, select Control Panel > Security Center.) Look at the top of the screen to make sure the firewall is turned on. If it’s not, click the Windows Firewall icon at the bottom of the screen, select On, and click OK. The firewall will now be turned on. But the firewall built into XP only offers inbound protection -- in other words, it blocks unsolicited incoming connections, but not outbound connections. Spyware and Trojans often “phone home,” making outbound connections from your PC without your knowledge. If you want to block outbound connections, you need a two-way firewall. The best free one you can find is ZoneAlarm from Zone Labs. If you’re only looking for a two way firewall, there’s no need to buy one of ZoneAlarm’s for-pay versions, which offer extra features such as virus protection. This is one area where users of older versions of Windows don’t have much in the way of free options. ZoneAlarm no longer supports Windows 98 and ME, so if you’re using one of those operating systems, you’ll need to shell out for a commercial firewall such as Symantec’s Norton Personal Firewall or Trend Micro’s PC-cillin Internet Security.

14.1.3 Encrypt Your Data

No matter how good you are at making sure no one else has access to your PC, someone might be able to get in. It could be a hacker, or someone who is on the network you use. If you’re at work, it might even be a coworker who sits down at your PC when you’re out of the office.

Encrypt data that you don’t want others to see. Most encryption programs cost money, and many aren’t particularly easy to use. But Cryptainer LE from Cypherix is both free and simple to use.

Install it, and it creates a new, encrypted volume on your PC. Create files inside that volume, or move files into the volume, and they're encrypted on the fly. You can work with them as you would any other files, without having to use a password.

When you want any files or folders hidden from prying eyes, highlight them and click the Unload button in Cryptainer LE. They'll suddenly vanish. To make them appear, click the Load button, and they're back after you type in a password. Only those with access to the password will be able to see them. The software is also useful for those who use small USB flash drives to carry around data. You can encrypt the entire drive so that if you lose it, no one else can see the files on it.

14.1.4 Protect Yourself Against Phishers

Phishing is one of the most insidious and nastiest attacks out there. You get a legitimate-looking e-mail from your bank, eBay, PayPal, or other financial institution warning you that you must click a link to log into your account for some reason -- to update it, confirm your information, or even for protective purposes.

There are simple ways to thwart phishing attacks. Never click on a log-in link from an e-mail purporting to be from your financial institution, eBay, or PayPal, no matter how legitimate it looks. Instead, go to the site yourself and log in.

Second, use an anti-phishing toolbar, which will block you from visiting a phishing site or warn you when you're visiting one. There are plenty of good ones out there. The Google Toolbar includes an anti-phishing feature that will block you from visiting a phishing site and pop up a warning about it. After you install the toolbar, click on its "Options" button. Then, under the "Browsing" tab, check the box next to "Safe Browsing." Click the "Safe Browsing Settings" button and configure your level of protection. Click "OK."

Don't go there! The Google Toolbar includes an anti-phishing feature to protect you from phishing scams.

Another good anti-phishing toolbar is the Netcraft Toolbar, which offers similar protection. Soon you won't need any anti-phishing toolbars, because both Internet Explorer 7 and Firefox 2.0 will include anti-phishing tools built right into the browser. In preliminary tests, the IE7 anti-phishing tools caught more phishing attacks than did Firefox 2.0, but both products are still in beta.

14.1.5 Disable File Sharing

One of your biggest security dangers is sitting in plain sight, and you probably don't even know it. If you've set up your PC to share files and folders, it's exceptionally easy for people to look through all your files, grab personal information, and even delete files and folders as well. Odds are, though, that you don't know if you're set up for sharing.

It's easy to find out, and then to turn off sharing. Open Windows Explorer and look at all of your folders. Any folder that has a small hand beneath it means that folder is being shared, and that anyone connected to your network can gain access to it. To turn off sharing, right-click the folder, select Sharing and Security, click the Sharing tab, select "Do not share this folder," and click OK.

By the way, when a folder is shared, all the subfolders beneath it are automatically shared as well. But those subfolders won't show the small hand beneath the folder or indicate that it's shared in the Sharing tab. So be careful to look at all top-level folders to see if they're shared. And you should always check to make sure that your root drives aren't shared, because if they are, others have access to all the folders and files on your system.

Notes

14.1.6 Surf the Web Anonymously

When you surf the Web, your life is an open book. Web sites can track your online travels, know what operating system and browser you're running, find out your machine name, peer into your clipboard, uncover the last sites you visited, examine your history list, and delve into your cache.

They can also examine your IP address to learn basic information about you, such as your geographic location. Pretty scary stuff.

But if you'd like, you can browse in perfect anonymity. There's plenty of software you can buy that will do this for you, but you can do it for free by using an anonymous proxy server that sits between you and the Web sites you visit. When you use an anonymous proxy server, your browser doesn't contact Web sites directly—the proxy server acts as a buffer, which means the sites see the IP address of the proxy server, not your PC's IP address. Web sites can't read your cookies, see your history list, or examine your clipboard and cache because your PC is never in direct contact with them. You can surf without a trace.

One way to do it is to head to the free site The Cloak. Click the "Surf" link on the left. From there, type in the URL you want to visit, and the site acts as your proxy, with all your information hidden.

If you want, you can instead manually set your browser to use an anonymous proxy server. Find an anonymous proxy at the AiS Alive Proxy List.

Write down the server's IP address and the port it uses. For example, in the listing 24.236.148.15:80, the IP address is 24.236.148.15, and the port number is 80.

Then, in Internet Explorer, select Tools > Internet Options, click the Connections tab, and click the LAN Settings button. Check the "Use a proxy server for your LAN" box. In the Address field, type in the IP address of the proxy server. In the Port field, type in its port number. Check the "Bypass proxy server for local addresses" box; you don't need to remain anonymous on your local network. Click OK twice to close the dialog boxes.

In Firefox, select Tools > Options > General >

Connection Settings, click the "Manual proxy configuration" button, enter your proxy information, and click OK twice.

14.1.7 Say No to Cookies

Online ad networks have the potential to create in-depth profiles of your Web travels and personal interests. They place cookies on your hard disk that track you across multiple sites.

You can fight back by placing an opt-out cookie—provided by the ad network—on your hard disk that will tell sites to keep their mitts off your surfing habits.

To opt out of the massive Double Click online advertising network, go to its opt-out page and click on the "Ad Cookie Opt-Out" button at the bottom of the screen.

Some other advertising networks let you opt out as well. For details, go to the Network Advertising Initiative site; check the Opt-Out box next to any ad networks you want to opt out of, and then click Submit.

14.1.8 Protect yourself against E-mail "Nigerian Scams"

E-mail "Nigerian scams" are among the oldest and well-known on the Internet, in which you're sent an unsolicited e-mail asking for help to transfer millions of dollars out of Nigeria—but somehow, it's your bank account that gets emptied.

Well, the scam has morphed, and Nigerian scams are now rife on eBay. This time around they're often pointed at sellers of items, not buyers.

Here's how it works. You put an item up to bid. At the end of the auction, the winning bidder gets in touch with you and asks that you ship the item to Nigeria, or somewhere else overseas. Often, there's a strange story attached – a common one is that the bidder lives in the U.S., but has just adopted a child in Nigeria, and wants the item sent directly to the child there.

The winning bidder sends you what appears to be a PayPal notification, saying that the item has been paid for. Or else he sends you an e-mail saying that as soon as you send him a confirmation that you've shipped the item, he'll pay you via PayPal. Ship the item, and you've been scammed. The PayPal notification was in fact a forgery, and, of course, if you first ship it before getting payment, you'll never get paid.

How to protect against it? First, never ship an item until you confirm that you've been paid. Don't trust an e-mail from a bidder, or from PayPal itself, that appears to say a payment has been made. Instead, log into your PayPal account and see if there has in fact been a payment.

Second, only sell items to people who have already bought items at other auctions. Scammers often set up new accounts for scams, and these accounts have zero activity. If you see a high bidder on an item of yours with zero activity, go to the Canceling bids placed on your listing page and fill out the form for canceling a bidder.

14.1.9 Virus Scan

Sometimes, typically via email, virus are able to cross the wall and end up on your computer anyway. A virus scanner will locate and remove them from your hard disk. A real time virus scanner will notice them as they arrive, even before they hit the disk, but at the cost of slowing down your machine a little. Important: because new virus are arriving every day, it's important to keep your virus definitions up-to-date. Be sure to enable the scanning software's automatic-update feature and have it do so everyday.

"It all might seem overwhelming, but it's not nearly as overwhelming as an actual security problem if and when it happens to you."

14.1.10 Kill Spyware

Spyware is similar to virus in that they arrive unexpected and unannounced and proceed to do something undesired. Normally spyware is relatively benign from a safety perspective, but it can violate your privacy by tracking the web sites you visit, or add "features" to your system that you didn't ask for. The worst offenders are spyware that hijack normal functions for themselves. For example, some like to redirect your web searches to other sites to try and sell you something. Of course some spyware is so poorly written that it might as well be a virus, given how unstable it can make your system. The good news is that, like virus scanners, there are spyware scanners that will locate and remove the offending software.

14.1.11 Stay Up-To-Date

I'd wager that over 90% of virus infections don't have to happen. Software vulnerabilities that the viruses exploit usually already have patches available by the time the virus reaches a computer. The problem? The user simply failed to install the latest patches and updates that would have prevented the infection in the first place. I still see this constantly, as some of the most popular articles here on Ask Leo! Deal with exploits that were patched nearly 2 years ago. The solution is simple: enable automatic updates, and visit Windows Update periodically.

Notes

14.1.12 Secure Your Mobile Connection

If you're traveling and using internet hot spots, free Wifi or internet cafes, you must take extra precautions. Make sure that your web email access is via secure (https) connections, or that your regular mail is over an encrypted connection as well. Don't let people "shoulder surf" and steal your password by watching you type it in a public place. Make sure your home Wifi has WPA security enabled if anyone can walk within range.

14.1.13 Don't Forget the Physical

An old computer adage is that "if it's not physically secure, it's not secure." All of the precautions I've listed above are pointless if other people can get at your computer. They may not follow the safety rules I've laid out. A thief can easily get at all the unencrypted data on your computer if they can physically get to it. The common scenario is a laptop being stolen during travel, but I've gotten reports of people who've been burned because a family member or roommate accessed their computer without their knowledge.

It all might seem overwhelming, but it's not nearly as overwhelming as an actual security problem if and when it happens to you. While we might want it to be otherwise, the practical reality of the internet, and computing today, is that we each must take responsibility for our own security online.



If you're traveling and using internet hot spots, free Wifi or internet cafes, you must take extra precautions. Make sure that your web email access is via secure (https) connections, or that your regular mail is over an encrypted connection as well. Don't let people "shoulder surf" and steal your password by watching you type it in a public place. Make sure your home Wifi has WPA security enabled if anyone can walk within range.

14.2 Protect Yourself

The best defense against identity theft is to do business only with reputable merchants. Apply the proverbial sniff-test when selecting an online retailer. Does it feel and look legit? Check the merchant at the Better Business Bureau, or look for their ranking on the Internet Retailer 500 list. Check out for feedback on opinion sites like (Epinionsandbizrate). In other words, do your homework before you plunk down your credit card.

Check the retailer's privacy policy, too. It should specify what they will and won't do with your information. (Note that those policies change, especially when a company changes hands). You'd be surprised how many sell online merchants sell your e-mail and address to third parties. Search the privacy policy for terms like third-party, e-mail address, or personally identifiable information; that should help you find out what a retailer plans to do with your information.

Assuming you are working with a reputable merchant, be sure you are actually on the correct site. A phishing attack or network redirection attack can direct you from e-mail or another Web address to a fake site (which may look exactly like the real thing). One Firefox plug-in, LocationBar 2 can help: It makes clear what Website you are using.

Regardless of your best efforts, your information may still be compromised. That's why it's also important to monitor your identity through services such as Debix, Citi IdentityMonitor, and Experian). They will alert you when any new credit requests appear in your file. Get into the habit of checking your credit card and banking accounts frequently to make sure there are no unauthorized charges.

As mentioned above, reputable retailers spell out how they use your information in their privacy policies. Check to see if your favorites sell your information to other merchants. If so, then see if you can opt-out of offers from third parties.

Consider connecting to the Internet through a VPN or private proxy, which will obscure your IP address and therefore your location. This is only effective when you're browsing--once you decide to buy, your identity will be exposed.

Finally, if you're really paranoid, shop at a variety of sites to limit the depth of knowledge about you that any one retailer can acquire. This has its risks, however; buying from that guy on 33rd Street just to avoid shopping too often at Nordstroms is counterproductive.

14.3 Protect Your Privacy

14.3.1 Avoid Identity Theft

When you're buying online, it can sometimes be hard to tell whether you're dealing with a legitimate merchant or the online equivalent of that guy selling counterfeit watches on 33rd Street. Most shoppers focus on maintaining the privacy of their credit card data, and that's good. But that's not the only privacy concern you should have while shopping online.

14.3.2 Identity Theft

The risk: Any time you use your credit card online, your identity is at risk. Organized crime factions from all over the world have streamlined the process of extracting your personal details from all sorts of places, especially shopping sites. These attackers can harvest thousands or even millions of credit cards in one fell swoop. That's a problem for two reasons.

First, and most obviously, there's the problem of having your credit card used to purchase all kinds of goods in places you've never been. Usually your bank protects you from such fraudulent charges. But it's still a hassle to change your account numbers.

But these days, the bad guys aren't satisfied with credit cards alone. In fact, individual card numbers aren't worth that much. But when those numbers are combined with other commonly available bits of personal information—such as addresses and birthdays—attackers can then assemble a virtual dossier of your private information. Identity thieves can parlay that information to secure credit in your name. That credit can be more valuable to them than your credit card and far more difficult and expensive for you to fix: Bank protections focus on credit cards. They are much less help when money is directly extracted from accounts, via debit cards or electronic fund transfers.

14.3.3 Spying

The risk: It may be convenient when your favorite online merchants e-mail offers for products you were thinking about buying. But wait—how did they know that? Unfortunately, if you've been using Google to browse pages for similar products, or perusing the merchant's Website, it's not a secret. You've been telling them what you like to buy and when—they're just listening.

With the advent of tracking technologies and sophisticated analytics, many Web merchants know exactly who you are and what you are most likely to buy. They know because you tell them through your buying and surfing patterns. This is valuable data, and merchants can (and do) sell it to each other.

14.4 Managing Cookies

14.4.1 Cookies

An internet cookie is a packet of information sent by a server to a browser, which is then sent by the browser each time it accesses the server. Cookies are typically used to authenticate a registered user of a web site, personalizing the site, maintaining an online shopping cart, etc. Originally developed by Netscape, cookies offer convenience to the visitor if care is taken by the website. Usually internet cookies are specific to one domain – meaning a cookie set by one domain cannot be read by other domains. Many websites subscribe to media services that place advertisements for them. One of the controversies surrounding cookies is the use of cookies to build a personal profile of the user's browsing and purchasing habits. One can set the browser to disable cookies, or use Internet filter software to filter out cookies.

When you view a web page, the web server which sends it to you can store a small parcel of text on your computer, which will be sent back to the server each time you request the same or another page from the same web site. This bit of text is called an HTTP cookie, web cookie, or, most commonly, just cookie. In addition to the text data, a cookie can have an expiration date, at which time it will automatically be deleted – if it doesn't have an expiration date, it will be deleted when you exit your web browser.

The maximum amount of data that can be stored in one cookie is four kilobytes in most browsers, the equivalent of about two pages of typewritten text. (By comparison, a standard 3.5" floppy disk can hold 1,440 kilobytes of data, and a typical three-minute MP3 song takes about twice that.), but most cookies use a fraction of that space. The number of cookies a browser will store for a single web site (domain name, to be more specific) varies from browser to browser, but most will allow 30 or more.

To understand how cookies work, you first must understand a bit about how the HTTP protocol works. Here are the basics: When you enter an address in your browser's address bar or click on a link a page loads an image, video, or other file, what your browser is really doing is sending an HTTP request to a web server. When the server receives your request, it loads or generates the requested web page, image, video, etc. and sends it – in the form of an HTTP response – to your web browser, which then displays it for you.

Both requests and responses can include extra information like browser type, date and time, and so on in the form of "headers" which are used by your computer and the server, but not displayed on your screen. When a server sends a response to your web browser, one of the headers it can include is a "Set-Cookie" header, which gives the browser text data and an expiration date to store in a cookie. Then, the next time you send another request to the same server, that cookie – assuming it hasn't expired – will be sent back, unchanged, to the server along with the request.

Now that you know how cookies work, you might be wondering what they're good for--what use is it for a web server to store tiny bits of data on your computer? Well, mainly web sites use cookies to remember information about you and how you use them. For example, when you view an item on Amazon, Amazon stores a cookie on your computer, and when you return to Amazon's front page, your browser sends the cookie back, and Amazon uses it to give you quick access to the item you looked at before, or show you related items. A weather web site could use a cookie to remember your ZIP code so you don't have to enter it every time you visit. In most cases if you log in to a web site and the site is able to "remember" you the next time you come back, it does so using cookies. In many cases web sites don't store the actual information in cookies – that could be a security risk--but rather the information is stored in a database on the web server, and a unique but meaningless value associated with the database record is stored in a cookie on your computer.

Cookies are not anything like viruses or spyware, despite popular misconceptions, and they can't harm your computer or your files. Your web browser will only send a cookie to the same web server that created it, and web servers have no way to retrieve information from your computer other than the cookies it created.

On the other hand, there are some privacy issues to be aware of with cookies. While your web browser will only send cookie data to the same web server that sent it to you, cookies nevertheless can be used, in some cases, to track your activities across multiple web sites. Here's how: A web page can include images, scripts, and other data, that is actually hosted on other web servers. For example, you can display a video from YouTube on your own blog. This is a good thing. The flip side is that when multiple web sites use, for example, the same ad network, because the advertisements are all sent from the same web server or servers (the ad network's), the ad network knows which of the sites you have visited. You are still anonymous to the ad network — it cannot find out your real name or your credit card numbers, for example—but it can use the information it has learned about your browsing habits to display ads targeted to your interests when you visit those sites. Some people consider behavior like this to be a violation of their privacy. In the next section I'll give you some tips for dealing with these kinds of cookies.

Every web browser has some built-in functionality to view and manage cookies. I'll step you through finding and using those tools in Mozilla Firefox and Internet Explorer below. For other browsers, check their help documentation.

14.4.2 Internet Explorer

In Internet Explorer, you can manage your cookies by clicking on the Tools menu, choosing Internet Options. If you want to turn cookies on or off, click on the Privacy tab and then the Advanced button. If you check the "Override automatic cookie handling" checkbox, you can select "Block" to never allow cookies (which I don't recommend, as it will severely limit your use of some web sites) or "Prompt" to be prompted every time a web site tries to set a cookie (this will get annoying very quickly). In this dialog, "First-party cookies" refers to cookies set by the server that hosts the page you're looking at, and "Third-party cookies" refers to cookies set by other servers whose images, etc. are included on the page you're viewing, such as in the ad network scenario mentioned above. "Always allows session cookies" refers to those cookies that are automatically deleted when you exit Internet Explorer.

If you want to manage the cookies that have already been set in Internet Explorer, go to the General tab in Internet Options. If you want to delete all of the cookies that are currently being stored, click on Delete Cookies... under Temporary Internet Files in Internet Explorer 6, or, in IE7, the Delete button under Browsing History followed by Delete Cookies. If you'd rather see all of the cookies and delete them individually, click on Settings under Temporary Internet Files (for IE6) or Browsing History (for IE7) and then click on View Files. This will take you to Internet Explorer's cache, where cookie files have names that start with "Cookie:". Each cookie will show an address, and if you open the file in Notepad you can see its contents, although it's unlikely to be intelligible. To delete a cookie, just select it and press delete.

14.4.3 Mozilla Firefox

To manage your cookies in Firefox, click on the Tools menu and choose Options... Then click on the Privacy tab. There, if you never want to accept cookies, you can uncheck the "Accept cookies from sites" box. Again, this will severely limit your use of some sites. If you want, you can click on Exceptions... to specify sites from which you never, or always, wish to accept cookies from. If you click on the Show Cookies... button, you can see all of the sites which have stored cookies on your computer, and complete data about each cookie and you can manually delete any you

Notes

want to get rid of. If you'd rather get rid of all of your cookies at once, click on Clear Now... under Private Data on the Privacy tab, and check to Cookies checkbox and uncheck all the others before clicking on Clear Private Data Now.

If you want even easier access to cookies in Firefox, I highly recommend the View Cookies add-on. It places a new tab on the Page Info dialog, allowing you to see all of the cookies for the site you're currently looking at with just a few clicks.

14.4.4 External Tools

There are a lot of third-party tools out there that are designed to make managing your cookies easier. Many of them can be found right here on Tucows. If you're just worried about those tracking cookies that have the potential to violate your privacy, you'll find that most good anti-spyware programs will detect and optionally delete the worst of them.

14.5 Spyware and Other BUGS

14.5.1 Spyware

Some companies place "spyware" through their software installations, usually without the user's permission. Spyware passes on information about software, browsing habits and purchasing habits of the user to the company's data collection facilities. Spyware programs do not collect information specifically, but report general demographics. Spyware could also have the capability to take names, credit card and other personal information.

The information gathered by such companies are usually sold and combined with other databases to build a profile of individual web users. This profile is mainly used for direct marketing purposes. But proper ad-ware is a serious revenue model for many companies and when used correctly do not pose a privacy threat.

Guard yourself against Spyware, use:

- Software tools
- Modify the hosts file on the system
- Remove files and making changes manually. This is a complex task as each type of spyware behaves differently and requires a certain amount of technical know-how Another software tool that threatens affiliate marketing is "steal ware". These are products that modify affiliate-tracking codes to change the person to whom the payment is due. Such tools completely invalidate affiliate marketing as a valid revenue model. Some peer-to-peer packages are reported to include these steal ware.



Task

Explain how bugs can harm our data storage devices.

14.5.2 Other Web Bugs

A web bug is a graphics on a web page or an email message that is designed to monitor who is reading the web page or email message. A web bug is often invisible as its size is only 1 pixel by 1 pixel. It is represented as HTML IMG tags. Any graphics used for monitoring is a web bug. All

invisible gif images are not web bugs, as some are used for alignment purposes. Web bugs are also known as clear gifs or 1 by 1 gifs or invisible gifs.

The informations sent to the server are:

- The IP address of the system that fetched the Web Bug
- The URL of the page that the Web Bug is located on
- The URL of the Web Bug image
- The time the Web Bug was viewed
- The type of browser that fetched the Web Bug image
- A previously set cookie value.

Networks use web bugs to add information to a personal profile of what sites a person is visiting. This information is stored in a database belonging to the ad network. This in turn determines what banner ad the user is shown. Web bugs are also used to gather statistics about web browser usage and independent accounting of the number of people who have visited a particular web site. Web bugs can be found by using the HTML source of a web page. The web bug is usually loaded from a different server than the rest of the page.

14.6 Keeping your Data Secure

After a series of catastrophic errors by big businesses and the government, data protection issues have been thrown into sharp relief. As criminals become more savvy to how they can use data and the public becomes more alert to the threats identity theft and careless handling of data pose, it's become increasingly important for businesses to ensure their approach to data protection is as scrupulous as possible.

14.6.1 The Data Protection Act

- (a) The Data Protection Act came into force in 1998 and provides businesses and public bodies with a series of regulations governing how they handle people's personal information. If you store any personal data on customers or employees, the Data Protection Act applies to you.
- (b) The information the Data Protection Act covers can be described as any information about living, identified or identifiable individuals. It includes names, addresses, email addresses, dates of birth, bank details and opinions expressed about an individual.
- (c) The Data Protection Act is split into eight principles, which require that you:
 1. Process data fairly and according to the law
 2. Process data for a limited, lawful purpose
 3. Only hold enough information for your purpose – nothing excessive or extraneous
 4. Ensure the data you hold is accurate, relevant and up-to-date
 5. Don't hold the data for any longer than necessary
 6. Process the data in line with individuals' rights (below)

Notes

7. Ensure the data is kept physically secure
 8. Don't transfer the data outside the European Economic Area unless it is adequately protected.
- (d) You need to be aware of the rights the Data Protection Act grants individuals. These include:
1. The right of subject access, which allows individuals to see the data you hold on them
 2. The right to prevent direct marketing, which means individuals, can opt out of being targeted with direct marketing, either online or by phone or mail. Once an individual has put their request in writing, you have up to 28 days to stop.
 3. The right to have personal information corrected
 4. The right to prevent automated decisions, which prevents you from making decisions on an individual using an automated process or algorithm. For example, it would be against the law to employ someone based purely on the results of a psychometric test.
- (e) In some cases, you may be required to notify the Information Commissioner's Office (ICO) that you are holding data. The ICO allows people to find out what information organizations are holding on them and what the information is being used for. If you use individuals' information for any purpose other than staff administration (payroll, etc), marketing or PR for your own business (rather than selling the information to a third party), or accounts and records, you will be required to notify the ICO. If you're at all uncertain, it's best to contact the ICO using the contact details below.
- (f) Losing data will put your business at risk, so make sure you follow best practice at all times. If you have any doubts over how you are handling your data, contact the information commissioner's office or visit its website.
- (g) Carry out a risk assessment to identify physical risks to your data. Could it be affected by power cuts, theft or fire? Make a plan which details how you will take action if your data is affected by any of these threats.
- (h) Make a list of who has access to sensitive data and who is responsible for inputting it, so you can identify who you need to train and who is at fault if something does happen to your data. Make sure these people are aware of the Data Protection Act and know how to handle data correctly.
- (i) It might seem obvious, but run regular virus scans to minimize the risks computer viruses pose. A recent report indicated more than three quarters of business computers are affected by viruses - and if your computer is hit by a bad one, the result could be catastrophic.
- (j) Implement an IT security policy to make clear to your staff exactly how they should be handling data. This should include rules on how to handle customer and business information, limitations on the amount of access your employees have to data, and an acceptable use policy for the internet and email.
- (k) As well as the increased threat of getting a virus, misuse of the internet could have a damaging effect on your business in other ways—including exposing your business to an increased risk of legal action, a loss of productivity, and damage to your reputation if one of your employees sends a badly-worded email. Be vigilant on this point and remind your employees personal emails are representing the company as well as the individual.
- (l) Create a data backup routine to make sure your business isn't affected if something happens to your servers. This should take place at least once a week, but ideally everyday.

14.7 Backing up Data

In Information Technology, a backup or the process of backing up refers to making copies of data so that these additional copies may be used to restore the original after a data loss event. The verb form is back up in two words, whereas the noun is backup (often used like an adjective in compound nouns).

Backups have two distinct purposes. The primary purpose is to recover data as a reaction to data loss, be it by data deletion or corrupted data. Data loss is a very common experience of computer users. 67% of internet users have suffered serious data loss. The secondary purpose of backups is to recover data from a historical period of time within the constraints of a user-defined data retention policy, typically configured within a backup application for how long copies of data are required. Though backups popularly represent a simple form of disaster recovery, and should be part of a disaster recovery plan, by themselves, backups should not alone be considered disaster recovery. Not all backup systems and/or backup applications are able to reconstitute a computer system, or in turn other complex configurations such as a computer cluster, active directory servers, or a database server, by restoring only data from a backup.

Since a backup system contains at least one copy of all data worth saving, the data storage requirements are considerable. Organizing this storage space and managing the backup process is a complicated undertaking. A data repository model can be used to provide structure to the storage. In the modern era of computing there are many different types of data storage devices that are useful for making backups. There are also many different ways in which these devices can be arranged to provide geographic redundancy, data security, and portability.

Before data is sent to its storage location, it is selected, extracted, and manipulated. Many different techniques have been developed to optimize the backup procedure. These include optimizations for dealing with open files and live data sources as well as compression, encryption, and de-duplication, among others. Many organizations and individuals try to have confidence that the process is working as expected and work to define measurements and validation techniques. It is also important to recognize the limitations and human factors involved in any backup scheme.

14.8 Safeguarding Your Computer Hardware

In small and home based businesses, the aspect of physical protection is often overlooked. Sufficient measures should also be put in place and maintained for protection against climate and environmental factors such as fire, dust, power, excessive heat and humidity, electrical emanations, and natural calamities. Regulation of power supply is vital.

In a good security policy, special emphasis should be given to protect all equipment handling or containing sensitive information. It should lessen the possibilities for an intruder to access these devices.

In order to have a secure computer network, it is necessary to have a proper mechanism for protecting the computer hardware and other equipments from external physical threats such as theft, natural calamities (earthquake, floods, or even accidents in the home, etc.)

14.8.1 Physical Access and Data Security

The first line of defense locally to protect network equipment such as servers, switches, and routers is to keep them in a locked, climate controlled, and fire protected environment. If equipment is not physically accessible to unauthorized personnel, there is less chance of accidental or intentional tampering.

Notes

It is important that access to critical system components such as the server is restricted to a small number of individuals (usually the administrator and his backup). The server should be located in a locked room to which access is restricted. Other considerations should include protection of equipment against theft, fire, and electrical hazards.

No one must be able to remove a disk containing sensitive information or to install devices to record confidential information. For this problem to be solved security policy must be created to maintain an environment secure enough to contain and keep the information handled by the equipment safe from any damage or loss. See the section entitled "Controlling Access to your PC" for a more in-depth look at dealing with access.

There are other specific items left to deal with in protecting computer data besides keeping data from the eyes of unethical immoral people – that of machine failure. Computers crash – Especially when you least want them to, like when you have a big project due or an important email that needs to be responded to immediately – It seems that Murphy's Law is the only law that holds any sway when it comes to your computer's hard drive – Unlike other issues discussed in this book, there are no common sense approaches to keep it from happening, no programmers out there hard at work trying to protect your computer from crashing – There aren't even any hackers or attackers to blame – The fact is that human beings are fallible and so are the things they make and therefore, hard drives are not perfect – Just like you would never expect to drive a car without it breaking down once in a while, so you should expect and learn to deal with crashing hard drives.

There are several very important things you can do to protect your hard drive, the most important being: back up all your data regularly, keep your computer in a cool environment, make an emergency bootable floppy disk, keep a fire extinguisher nearby, and keep programs to a minimum, uninstalling those you don't use, monitor and regulate the power supply and restrict authentication and access to your computer(s). Keeping programs to a minimum: running multiple programs also cause stress on your computer, decreasing your hard drive's life span – Keep beta versions and cracked software off your computer and if you don't use a program anymore, uninstall it.

There are several other security implications that arise from the fact that computers run on electricity. These include radio interference, which can be used for eavesdropping and sabotage, plus radiation, a potential liability threat.



Case Study

Employee Data - Appropriate Security Measures - Disclosure

A large organisation, whose staffs are employed at several locations throughout the country, used a central database to record information relating to its employees and their work. The complainant questioned the security arrangements in respect of his personal data, and the extent of access to such data throughout the organisation.

The organisation's computer system comprised about a hundred personal computers nationwide connected to a central computer in the Dublin head office. Some sixty laptop computers were also provided for use by employees when away from their offices. These laptops contained a version of the organisation's main database which was downloaded from the main computer and updated periodically. Accordingly, data kept by the organisation on its main database was available to staff in the head office, in the local offices, and at off-site locations.

Contd...

Notes

The complainant, an employee, made his complaint while the computer system was still being developed and implemented by the organisation. He made the following points. First, he alleged there had been a breach of security because the laptops were without any password protection for a period during the development of the system. Second, the complainant objected to certain of his personnel data and details of his work activity being generally available to staff, and argued that such data should only be available to those who needed them to perform their managerial functions.

Section 2(1) (d) of the Data Protection Act provides that “appropriate security measures shall be taken against unauthorised access to, or alteration, disclosure or destruction of, the data and against their accidental loss or destruction.” The question of the security of access to the laptop computers was considered in the light of this provision.

My investigation established that each laptop required use of a password for access to the local version of the database. Where a laptop was establishing a connection to the main computer, another password was needed, and access to the main database itself required the use of a third password. In principle this approach appeared to conform well to the requirements of section 2(1) (d) above. However, the apparent effectiveness of this approach had been compromised. In the interests of simplicity of operation the organisation issued a unique centrally-generated password to each member of staff (so that each staff member would only need to remember one password) thus reducing the effectiveness of the password system as a whole. Furthermore, in the course of training staff on an upgraded version of the software, the password security system was modified to allow trainees ease of access to the system. This modification gave open access to the main database from a number of laptops.

As soon as this fact was discovered, the data controller took steps to rectify the matter. It is not appropriate for a data controller to allow his standards of security to slip, so that personal data becomes more widely accessible than is necessary. However, I noted the prompt action taken by the data controller to put matters right, and - given that my investigation did not discover any evidence of unauthorised access or use of the data during the period when the passwords were not in operation - I did not uphold this part of the complaint.

The second ground for complaint put forward was the alleged wide availability throughout the organisation of details relating to the complainant’s work activities including particulars of annual and sick leave. This raised two separate but related issues: first, whether this wide availability constituted “disclosure” for the purposes of the Data Protection Act; and second, whether the wide availability of data was consistent with the organisation’s duty to take “appropriate security measures ... against unauthorised access to, or alteration, disclosure or destruction of, the data and against their accidental loss or destruction.”

On the first question, I noted that the only people with access to the main database were the staff of the data controller. The definition of “disclosure” given in section 1(1) of the Act, specifically states that disclosure “does not include a disclosure made ... to an employee ... for the purpose of enabling the employee ... to carry out his duties”. In my opinion, these words require a data controller to make an assessment, in respect of particular employees, as to whether such employees need to have access to particular holdings of personal data, and to provide accordingly. Thus, one would expect a Human Resources Manager to have access to personal data not necessarily available to the manager of a client database, and vice versa. Data controllers should, in my view, take reasonable steps to prevent personal data from being made available to employees who may have no work-related interest in the data.

On the second question, I consider that sensible restriction of the availability of personal data

Contd...

Notes

is one of the “appropriate security measures” that data controllers must consider. The more people who have access to personal data, the greater is the risk of unauthorised access or disclosure. These issues were discussed with the data controller in detail. The organisation explained that the wide availability of personnel information and staff operational details was due in part to business requirements, and in part to the culture and tradition of the organisation. Following discussions, the data controller made a number of significant changes to the computer system, at some expense, in order to restrict access to the personal data of employees. It is my view that, in a case such as this, an appropriate balance must be struck between the concerns of the employee as data subject, the real operational requirements of the organisation and the costs to the organisation. I took the view that, following the changes referred to above; the data controller was compliant with the Act.

Questions:

1. Why do the protecting privacy of a data over a communication channel is needed?
2. Explain the process involved in data protection.

14.9 Summary

- Home computers are typically not very secure and are easy to break into, when combined with high-speed Internet connection that are always turned on, intruders can quickly find and then attack home computers.
- Encrypt data means that you don't want to see.
- Internet Explorer manage your cookies by clicking on the tool menu.
- A web bug is a graphics on a web or an email message that is designed to monitor who is reading the web page or email message.
- In a good security policy, special emphasis should be given to protect all equipment handling or containing sensitive information.

14.10 Keywords

ARPA: Stand for the Advanced Research Projects Agency - that funded and managed the project.

Cookies: An internet cookie is a packet of information sent by a server to a browser, which is then sent by the browser each time it accesses the server.

Firewall: A firewall is a piece of software or hardware that sits between your computer and the internet and only allows certain types of things to cross the wall.

ICO(Information Commissioner's Office): It allows people to find out what information organizations are holding on them and what the information is being used for.

Internet Explorer: In Internet Explorer, you can manage your cookies by clicking on the Tools menu, choosing Internet Options.

Phishing: It is a way that internet scammers trick you into providing your personal and financial details. Phishing opens the door to identity theft, and more.

Remote Procedure Calls (RPC): It is a vulnerabilities in RPC that allowed for one of the more recent worms to propagate.

Spyware: Spyware is similar to virus in that they arrive unexpected and unannounced and proceed to do something undesired.

War drivers: One who hack into unsuspecting wireless networks.

Web bug: It is a graphics on a web page or an email message that is designed to monitor who is reading the web page or email message. A web bug is often invisible as its size is only 1 pixel by 1 pixel.



Lab Exercise

Understand the security-related problems that you need to think about and solve them.

14.11 Self-Assessment Questions

- The Data Protection Act came into force in _____.
 (a) 1995 (b) 1996
 (c) 1998 (d) 1997
- The Data Protection Act sets out _____ Principles for the safe and legal handling of data.
 (a) Five (b) Eight
 (c) Six (d) Seven
- 1960s, computers were very expensive and slow by today's standards.
 (a) True (b) False
- A standard _____ floppy disk can hold 1,440 kilobytes of data.
 (a) 1.5'' (b) 2.5''
 (c) 3.5'' (d) 4.5''
- The number of cookies a browser will store for a single web site varies from browser to browser, but most will allow _____ or more.
 (a) 10 (b) 20
 (c) 25 (d) 30
- The main function of Virus scan is to maximize the risks.
 (a) True (b) False
- Encryption uses _____ to conceal information.
 (a) English (b) Mathematics
 (c) Physics (d) Chemistry
- Cookies can harm your computer or your files.
 (a) True (b) False
- A web bug is often invisible as its size is only _____.
 (a) 1 pixel by 3 pixels. (b) 1 pixel by 2 pixels.
 (c) 2 pixels by 1 pixel. (d) 1 pixel by 1 pixel.
- Spyware is similar to _____.
 (a) Firewall. (b) Virus.
 (c) Both (a) & (b). (d) None of these.

Notes

14.12 Review Questions

1. What is a cookie?
2. What is Spyware?
3. What is a Web Bug?
4. How can you guard yourself against Spyware?
5. How to Clear All Files from a Computer Running Windows XP?
6. How to Create a System Restore Point?
7. How to Keep Your Computer Running Smoothly?
8. How to Organize Your Computer?
9. How to Do a Scan to Clean a Hard Drive?
10. How to Fix a Slow Internet on Windows Vista?
11. How to Clean a Computer of Viruses?
12. How to Make a Computer Fast by Deleting Viruses?
13. How to Clean & Restore Computers?
14. What is a firewall? Why one should need it?
15. How do we keep our system safe from Viruses?
16. How do cookies work and what are cookies for?

Answers for Self-Assessment Questions

- | | | | | |
|--------|--------|--------|--------|---------|
| 1. (c) | 2. (b) | 3. (a) | 4. (c) | 5. (d) |
| 6. (b) | 7. (b) | 8. (b) | 9. (d) | 10. (b) |

14.13 Further Reading



Books

Introduction to Computers, by Peter Norton, Publisher: McGraw Hill, Sixth Edition.



Online link

<http://www.cert.org/homeusers/HomeComputerSecurity/>
www.cypherix.com/docs/cryptainer_information_week.pdf