स्वाध्याय स्वमन्थन स्वावलम्बन

# UTTAR PRADESH RAJARSHI TANDON OPEN UNIVERSITY

(Established vide U.P. Govt. Act No. 10, of 1999)

Indira Gandhi National Open University

UP Rajarshi Tandon Open University

# BCA-07

## Elements of Systems Analysis and Design

| | | |
|---|---|---|
| FIRST BLOCK | : | System Analysis |
| SECOND BLOCK | : | System Design |
| THIRD BLOCK | : | System Development and Implementation |
| FOURTH BLOCK | : | Management Information System |
| FIFTH BLOCK | : | Case Studies |
| SIXTH BLOCK | : | SAD : Emerging Trends |

Shantipuram (Sector-F), Phaphamau, Allahabad - 211013

Uttar Pradesh
Rajarshi Tandon Open University

# BCA-07

# Elements of Systems Analysis and Design

## Block

## 1

## System Analysis

## COURSE INTRODUCTION

This course deals with the analysis, design, development, implementation and maintenance of computer based information systems. The earlier courses would have made the learner familiar with basic computer hardware and software concepts as well as with some of the programming languages. The programming experience acquired is complemented in this course with systems experience. This would enable the learner to cope with the number of components in a systems development approach and enmesh them correctly to result in a successful project. The first 3 blocks of the course are a run through of the basic stages of a system development life cycle. The recent trend of acquiring computer systems is often with a view to provide management with relevant information. A block in the course is therefore exclusively devoted to MIS. In order to strike a balance between the theoretical and applied aspects of systems analysis, a number of case studies have been included in a special block so that they can serve as a vehicle for applying systems concepts. While working on the case study, a student would have an opportunity to put into a practical context, the tools of analysis and design learned during the course and also in some cases modify the cases and suggest strategies for system improvements. The cases are drawn from actual real life situations and reflect practical problems that would neatly integrate into the learning process. There is a final block on the emerging trends, as development of systems moves from an art form to an industry. Some exposure is given in this block to organisational issues arising out of induction of computer systems, and the new capabilities that the systems analysts of tomorrow must have, when software would be in the form of multi- media and hyper-media with greater emphasis on visuals and sound as means of communication.

## BLOCK INTRODUCTION

This block on Systems Analysis dwells upon the first distinct phase in the development of a system. It is now a fairly well-established fact that errors in a system are more easily and inexpensively rectified; the earlier they are discovered, and hence the importance of the effort given in systems analysis.

Much of what is discussed here is not new or revolutionary; they are tried and tested methodologies. But a familiarity with, and understanding of, the tools and techniques of systems analysis will place the potential analyst in a position of confidence to tackle the tasks of development of new systems and to successfully communicate his strategies and approach to his colleagues both in the development team as well as in the client group.

This block is made up of 4 units.

The first unit provides an overview of systems, the various stages in the life cycle of the systems development, the factors creating a software crisis and the attributes which a good systems analyst must have.

Unit-2 outlines the various sources from which system projects can get initiated. It will go through the various considerations involved in project selection and the elements involved in the approach to a new problem.

The third unit relates to a more rigorous examination of the project in terms of a feasibility study. It also touches upon the issue of description of current system in a narrative form through context data flow diagrams.

The final unit, unit-4 considers in more detail the tools which analysts must possess to be able to arrive at specifications of systems requirements. Out of the numerous tools, techniques, styles and methodologies that prevail in the profession, a few representative and popular ones have been chosen.

At various points during the reading of the block, a student would benefit from referring to the study of case studies which are part of the course. There are sometime specific reference to the case studies during this block, but a general reading through the case studies would help in a better understanding of the concepts of this block by the student.

# UNIT 1 OVERVIEW OF SYSTEM ANALYSIS AND DESIGN

Structure



## 1.0 INTRODUCTION

The study of "SYSTEMS" is by no means a new or even recent endeavour. Systems have been in use for the last thousands of years. The Egyptians used bookkeeping system over 5000 years ago for keeping their accounts, while Phoenician astronomers tried to study the systems of stars for making predictions.

People freely talk of different types of system in their day to day life. However, we shall only concern ourselves with those systems which are of direct and immediate concern to a business in the processing of information to generate useful and meaningful results for management. In other words, we can say that systems analysts will refer to the analysis of business systems.

Systems today are very helpful in running the business efficiently. But a system can function in an effective way only if the users such as the accountants, business manager and other responsible individuals within the company make it function in a proper way. Many times, managers are told that they only need to know how to retrieve required information, thus making them ignorant of the operations of the system as a whole. In accepting such advice, these managers are, in essence, relinquishing a substantial part of the control of the organisation to the system's designers. It is, therefore, necessary that these potential systems analysts should clearly understand many other things also such as what a system is, what its objectives are, what kinds of systems there are, what goes with their creation and maintenance, what are their costs and benefits and how to analyse and monitor systems.

## 1.1 OBJECTIVES

After going through this unit, you should be able to:

- define system, systems study, systems analysis and systems approach
- state the common characteristics in all the systems
- describe the basic elements in systems analysis
- classify different types of systems
- explain what is system development and what is system development life cycle
- illustrate the role of a systems analysL

## 1.2 AN OVERVIEW

Systems analysis and design refers to the process of examining a business situation with the intent of improving it through better procedures and methods. Systems development can generally be thought of as having two major components: Systems Analysis and Systems Design. Systems design is the process of planning a new system or replace or complement an existing system. But before this planning can be done, we must thoroughly understand the existing system and determine how computers can best be used to make its operation more effective. Systems analysis, then, is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvement to the system. In brief, we can say that analysis specifies what the system should do. Design states how to accomplish the objective.

### 1.2.1 What is a system?

The word "SYSTEM" covers a very broad spectrum of concepts. This is derived from the Greek word systema, which means an organised relationship among the functioning units or components. In our daily life, we come into contact with the transportation system, the communication system, the accounting system, the production system, the economic system and for over three decades, the computer system. Similarly, business systems are the means by which business organisations achieve their pre-determined goals. A business system combines policies, personnel, equipment and computer facilities to co-ordinate the activities of a business organisation. Essentially, a business system represents an organised way of achieving the pre-determined objective of an organisation.

There are various definitions of the word system, but most of them seem to have a common idea that suggests that a system is an orderly grouping of interdependent components linked together according to a plan to achieve a specific goal. The word component may refer to physical parts (engines, wheels of car), managerial steps (planning, organising, controlling) or a subsystem in a multi-level structure. The components may be simple or complex, basic or advanced. They may be a single computer with a keyboard, memory and printer or a series of intelligent terminals linked to a mainframe. In either case, each component is part of the total system and has to do its own share of work for the system to achieve the desired goal.

### 1.2.2 Systems study, Systems analysis and Systems approach

Systems study may be defined as "a study of the operations of a set of connected elements and of the inter-connections between these elements". It shows clearly that one cannot ignore any part or element of a system without first finding out the effect that element has on the operation of the system as a whole. We can understand this with the help of systems analysis.

There is a difference between "systems approach" and "systems analysis" also. The systems approach shows a set of procedure for solving a particular problem. It applies scientific methods to observe, clarify, identify and solve a problem with special care being taken to understand the inter- relatedness between elements and their system characteristics. However, systems analysis is a management technique which helps us in designing a new system or improving an existing system.

### 1.2.3 Characteristics of a System

Based on the definition of a system, it is observed that following characteristics are present in all systems:

a) **Organisation:** Organisation implies structure and order. It is the arrangement of components that helps to achieve objectives. In the design of a business system, for example, the hierarchical relationships starting with the president on top and leading downward to the blue-collar workers represents the organisation structure. Likewise a computer system is designed around an input device, a central processing unit, an output device and one or more storage units. When these units are linked together, they work as a whole system for generating information.

b) **Interaction:** Interaction refers to the procedure in which each component functions with other components of the system. In an organisation, for example, purchasing must interact with production, advertising with sales and payroll with personnel. In a computer system also, the central processing unit must interact with other units to solve a problem. In turn, the main memory holds program, and the data that the arithmetic unit uses for computation. The inter-relationship between these components enables the computer to perform.

c) **Interdependence:** Interdependence means that component of the organisation or computer system depend on one another. They are coordinated and linked together in a planned way to achieve an objective.

d) **Integration:** Integration is concerned with how a system is tied together. It is more than sharing a physical part or locations. It means that parts of the system work together within the system even though each part performs a unique function. Successful integration will typically produce a better result as a whole rather than if each component works independently.

e) **Central Objective:** Central objective is the last characteristic of a system. Objectives may be real or stated. Although a stated objective may be the real objective, it is quite common that organisation may set one objective and operate to achieve another. The important point is that users must be aware about the central objective well in advance.

### 1.2.4 Elements of Systems Analysis

There are four basic elements in systems analysis. Brief description of each element has been given below:

a) **Outputs:** First of all, we must determine what the objectives or goals are, what do we intend to achieve, what is the purpose of our work; in other words, what is the main aim behind the system. Defining aim is very vital in system work. If we do not know where we want to go, we will not know when we have reached there. We shall be unnecessarily wasting our time and energy in the process. Once we know our aim, we can try to achieve it in the best possible way. The user department has to define these objectives in terms of their needs. These become the outputs which the systems analyst keeps into mind.

b) **Inputs:** Once we know the output, we can easily determine what the inputs should be. Sometimes, it may happen that the required information may not be readily available in the proper form. This may be because of the existing forms are not properly designed. Sometimes, it may not be possible to get the required information without the help of top management. If the information is vital to the system, we should make all possible efforts to make it available. Sometimes, it might be too costly to get the desired information. It would be better in such cases to prepare a cost-benefit analysis to convince the management of the necessity for acquiring the information. The essential elements of inputs are:

   i) **Accuracy:** If the data is not accurate, the outputs will be wrong.

   ii) **Timeliness:** If data is not obtained in time, the entire system falls into arrears.

   iii) **Proper format:** The inputs must be available in proper format.

iv) Economy: The data must be produced at the least cost.

c) Files: As the word implies files are used to store data. Most of the inputs necessary for the system may be historical data, or it may be possible that these are generated from within the system. These are stored in files either in terms of isolated facts or in large volumes.

d) Processes: Here we come to the details of how the inputs and files are converted into outputs. This involves the programs and the way in which data is processed through the computer. The processing involves a set of logical steps. These steps are required to be instructed to the computer and this is done by a series of instructions called "programs".

### 1.2.5 Types of Systems

Systems have been classified in different ways. Common classifications are:

i) Physical or abstract systems

ii) Open or closed systems

iii) Deterministic or probabilistic systems

iv) Man-made information systems

(i) Physical or Abstract Systems: Physical systems are tangible entities that may be static or dynamic in operation. Abstract systems are conceptual or non-physical entities which may be as straightforward as formulas of relationships among sets of variables or models - the abstract conceptualization of physical situations.

(ii) Open or Closed Systems: An open system continually interacts with its environments. It receives inputs from and delivers outputs output to the outside. An information system belongs to this category, since it must adapt to the changing demands of the user. In contrast, a closed system is isolated from environmental influences. In reality completely closed systems are rare.

(iii) Deterministic or Probabilistic Systems: A deterministic system is one in which the occurrence of all events is perfectly predictable. If we get the description of the system state at a particular time, the next state can be easily predicted. An example of such a system is a numerically controlled machine tool. Probabilistic system is one in which the occurrence of events cannot be perfectly predicted. An example of such a system is a warehouse and its contents.

(iv) Man-made Information Systems: It is generally believed that information reduces uncertainty about a state or event. For example, information that the wind is calm reduces the uncertainty that a trip by boat will be enjoyable. An information system is the basis for interaction between the user and the analyst. It determines the nature of relationship among decision makers. In fact, it may be viewed as a decision centre for personnel at all levels. From this basis, an information system may be defined as a set of devices, procedures and operating systems designed around user-based criteria to produce information and communicate it to the user for planning, control and performance. Many practitioners fail to recognise that a business has several information systems; each is designed for a specific purpose. The major information systems are :

- formal information systems
- informal information systems
- computer based information system

A Formal information system is based on the organisation represented by the organisation chart. The chart is a map of positions and their authority relationships, indicated by boxes and connected by straight lines. It is concerned with the pattern of authority, communication and work flow.

An Informal information system is an employee-based system designed to meet personnel and vocational needs and to help in the solution of work-related problems. It also funnels information upward through indirect channels. In this way, it is considered to be a useful system because it works within the framework of the business and its stated policies.

Third category of information system depends mainly on the computer for handling business applications. Systems analysts develop several different types of information systems to meet a variety of business needs. There is a class of systems known collectively as Computer Based Information Systems. As we have different types of transportation systems such as highway systems, railway systems and airline systems, computer based information systems are of too many types. They are classified as:

- Transaction Processing Systems (TPS)
- Management Information Systems (MIS)
- Decision Support Systems (DSS)
- Office Automation Systems (OAS).

The figure 1.1 shows the organisation chart of computer based information system (CBIS) and figure 1.2 shows the hierarchical view of CBIS.

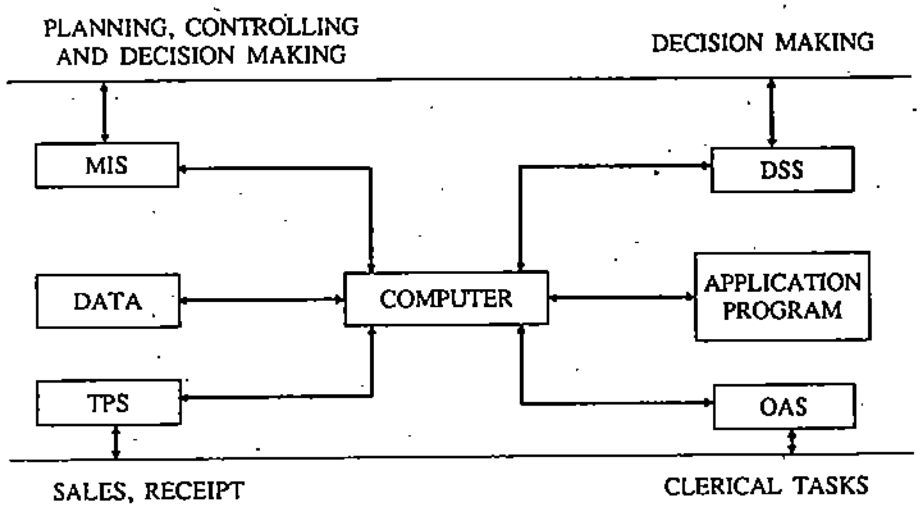



**Figure 1.1: CBIS in an Organisational Context**

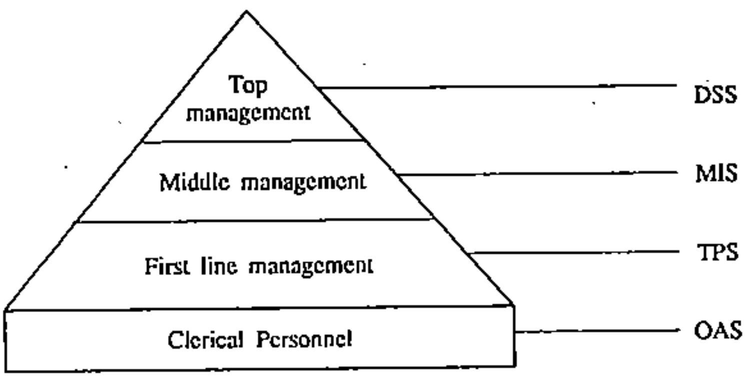



**Figure 1.2: The Hierarchical View of CBIS**

**Transaction Processing Systems:**

The most fundamental computer based system in an organisation pertains to the processing of business transactions. A transaction processing system can be defined as a computer based system that captures, classifies, stores, maintains, updates and retrieves transaction data for record keeping and for input to other types of CBIS. Transaction Processing Systems are aimed at improving the routine business activities on which all organisations depend. A transaction is any event or activity that affects the whole organisation. Placing orders, billing customers, hiring of employees and depositing cheques are some of the common transactions. The types of transactions that occur vary from organisation to organisation.

But this is true that all organisations process transactions as a major part of their daily business activities. The most successful organisations perform this work of transaction processing in a very systematic way. Transaction processing systems provide speed and accuracy and can be programmed to follow routines without any variance.

**Management Information System:**

Data processing by computers has been extremely effective because of several reasons. The main reason being that huge amount of data relating to accounts and other transactions can be processed very quickly. Earlier most of the computer applications were concerned with record keeping and the automation of routine clerical processes. However, in recent years, increasing attention has been focussed on computer applications providing information for policy making, management planning and control purposes. MIS are more concerned with management function. MIS can be described as information system that can provide all levels of management with information essential to the running of smooth business. This information must be as relevant, timely, accurate, complete and concise as is economically feasibl

**Decision Support Systems:**

It is an information system that offers the kind of information that may not be predictable, the kind that business professionals may need only once. These systems do not produce regularly scheduled management reports. Instead, they are designed to respond to a wide range of requests. It is true that all the decisions in an organisation are not of a recurring nature. Decision support systems assist managers who must make decisions that are not highly structured, often called unstructured or semi-structured decisions.. A decision is considered unstructured if there are no clear procedures for making the decision and if not all the factors to be considered in the decision can be readily identified in advance. Judgement of the manager plays a vital role in decision making where the problem is not structured. The decision support system supports, but does not replace, judgement of manager.

**Office Automation Systems:**

Office automation systems are among the newest and most rapidly expanding computer based information systems. They are being developed with the hopes and expectations that they will increase the efficiency and productivity of office workers-typists, secretaries, administrative assistants, staff professionals, managers and the like. Many organisations have taken the first step toward automating their offices. Often this step involves the use of word processing equipment to facilitate the typing, storing, revising and printing of textual materials. Another development is a computer based communications system such as electronic mail which allows people to communicate in an electronic mode through computer terminals. An office automation system can be described as a multi-function, integrated computer based system that allows many office activities to be performed in an electronic mode.

Categories of different information systems with their characteristics have been described briefly in table 1.1 :

**Table 1.1 : Categories of Information Systems**

| Category of Information System | Characteristics |
|---|---|
| Transaction Processing System | Substitutes computer-based processing for manual procedures. Deals with well structured routine processes. Includes record-keeping applications. |
| Management Information System | Provides input to be used in the managerial decision process. Deals with supporting well structured decision situations. Typical information requirements can be anticipated |

| | |
|---|---|
| Decision Support System | Provides information to managers who make judgements about particular situations. Supports decision makers in situations that are not well-structured. |
| Office Automation System | It is a multi-function, integrated computer based system, that allows many office activities to be performed in an electronic mode. |

Check Your Progress 1

1. What is the basic difference between "systems approach" and "systems analysis"?

..................................................................

..................................................................

..................................................................

2. What are the four basic elements in systems analysis?

..................................................................

..................................................................

..................................................................

3. What is a Computer Based Information System?

..................................................................

..................................................................

..................................................................

4. When is a decision considered to be unstructured?

..................................................................

..................................................................

..................................................................

## 1.3 SYSTEM DEVELOPMENT LIFE CYCLE

System development, a process consisting of the two major steps of systems analysis and design, starts when management or sometimes system development personnel feel that a new system or an improvement in the existing system is required. The systems development life cycle is classically thought of as the set of activities that analysts, designers and users carry out to develop and implement an information system. The systems development life cycle consists of the following activities:

- Preliminary investigation
- Determination of system requirements
- Design of system
- Development of software
- Systems testing
- Implementation, evaluation and maintenance

### 1.3.1 Preliminary Investigation

A request to take assistance from information systems can be made for many reasons, but in each case someone in the organisation initiates the request. When the request is made, the first systems activity the preliminary investigation begins. This activity has three parts:

i) request clarification

ii) feasibility study

iii) request approval

**Request Clarification:** Many requests from employees and users in the organisations are not clearly defined. Therefore, it becomes necessary that project request must be examined and clarified properly before considering systems investigation.

**Feasibility Study:** An important outcome of the preliminary investigation is the determination that system requested is feasible. There are three aspects in the feasibility study portion of the preliminary investigation:

(i) **Technical Feasibility:** Can the work for the project be done with current equipment, existing software technology and available personnel? If new technology is needed, what is the likelihood that it can be developed?

(ii) **Economic Feasibility:** Are there sufficient benefits in creating the system to make the costs acceptable? Or, are the costs of not creating the system so great that it is advisable to undertake the project?

(iii) **Operational Feasibility:** Will the system be used if it is developed and implemented? Will there be resistance from users that will undermine the possible application benefits?

The feasibility study is carried out by a small group of people who are familiar with information systems techniques, understand the parts of the business or organisation that will be involved or affected by the project, and are skilled in the systems analysis and design process.

**Request Approval:** It is not necessary that all requested projects are desirable or feasible. Some organisations receive so many project requests from employees that only a few of them can be pursued. However, those projects that are feasible and desirable should be put into a schedule. In some cases, development can start immediately, although usually systems staff members are busy on other ongoing projects. When such situation arises, management decides which projects are most urgent and schedules them accordingly. After a project request is approved, its cost, priority, completion time, and personnel requirements are estimated and used to determine where to add it to any existing project list. Later on, when the other projects have been completed, the proposed application development can be initiated.

A further discussion on **preliminary investigation** is covered in section 2.5 of unit 2.

### 1.3.2 Determination of System Requirements

At the heart of systems analysis is a detailed understanding of all important facets of the business area under investigation. The key questions are:

- What is being done?
- How is it being done?
- How frequently does it occur?
- How great is the volume of transactions or decisions?
- How well is the task being performed?
- Does a problem exist?
- If a problem exists, how serious is it? What is the underlying cause?

To answer the above questions, systems analysts discuss with different category of persons to collect the facts about the business process and their opinions of why things happen as they do and their views for changing the existing process. During analysis, data are collected on the available files, decision points and transactions handled by the present system. Some tools are used in analysis like data flow diagrams, interviews, on-site observations and questionnaires. Detail investigations also require the study of manuals and reports. Once the structured analysis is completed, analyst has a firm understanding of what is to be done?

### 1.3.3 Design of System

The design of an information system produces the details that clearly describe how a system will meet the requirements identified during systems analysis. Systems specialists often refer to this stage as logical design, in contrast to the process of developing program software, which is referred to as physical design.

Systems analysts begin the design process by identifying reports and other outputs system will produce. Then the specific data on each are pinpointed. The systems design also describes the data to be input, calculated or stored. Individual data items and calculation procedures are written in detail. Designers select file structures and storage devices, such as magnetic disk, magnetic tape or even paper files. Procedures they write tell how to process the data and produce the output. The documents containing the design specifications portray the design in many different ways-charts, tables, and special symbols. The detailed design information is passed on to the programming staff for the purpose of software development. Designers are responsible for providing programmers with complete and clearly out lined software specificaitons.

### 1.3.4 . Development of Software

Software developers may install purchased software or they may develop new, custom-designed programs. The choice depends on the cost of each option, the time available to develop software and the availability of programmers. Generally it has been observed that programmers are part of permanent professional staff in a big organisation. In smaller organisation, without programmers, outside programming services may be hired or retained on a contractual basis. Programmers are also responsible for documenting the program, providing an explanation of how and why certain procedures are coded in specific ways. Documentation is essential to test the program and carry on maintenance once the application has been installed.

### 1.3.5 Systems Testing

During systems testing, the system is used experimentally to ensure that the software does not fail. In other words, we can say that it will run according to its specifications and in the way users expect. Special test data are input for processing, and the results examined. A limited number of users may be allowed to use the system so that analyst can see whether they try to use it in unforeseen ways. It is desirable to discover any su ises before the organisation implements the system and depends on it.

### 1.3.6 Implementation, Evaluation and Maintenance

Implementation is the process of having systems personnel check out and put new equipment into use, train users, install the new application and construct any files of data needed to use it. This phase is less creative than system design. Depending on the size of the organisation that will be involved in using the application and the risk involved in its use, systems developers may choose to test the operation in only one area of the firm with only one or two persons. Sometimes, they will run both old and new system in parallel way to compare the results. In still other situations, system developers stop using the old system one day and start using the new one the next.

Evaluation of the system is performed to identify its strengths and weaknesses. The actual evaluation can occur along any of the following dimensions:

(i) Operational Evaluation: Assessment of the manner in which the system functions, including ease of use, response time, overall reliability and level of utilization. .

(ii) Organisational Impact: Identification and measurement of benefits to the organisation in such areas as financial concerns, operational efficiency and competitive impact.

(iii) User Manager Assessment: Evaluation of the attitudes of senior and user manager within the organisation, as well as end-users.

(iv) Development Performance: Evaluation of the development process in accordance with such yardsticks as overall development time and effort, conformance to budgets and standards and other project management criteria.

Maintenance is necessary to eliminate errors in the working system during its working life and to tune the system to any variations in its working environment. Often small system deficiencies are found as a system is brought into operations and changes are made to remove them. System planners must always plan for resource availability to carry out these maintenance functions. The importance of maintenance is to continue to bring the new system to standards.

**Check Your Progress 2**

1. What are the activities which complete the system development life cycle ?

   ..............................................................

   ..............................................................

   ..............................................................

2. In preliminary investigation three types of feasibilities are usually studied. Name those.

   ..............................................................

   ..............................................................

   ..............................................................

3. What are the areas of operational evaluation ?

   ..............................................................

   ..............................................................

   ..............................................................

4. Why is maintenance of a system necessary ?

   ..............................................................

   ..............................................................

   ..............................................................

## 1.4 SOFTWARE CRISIS

The translation of a familiarity with computer hardware and software into the development of useful commercial or business information systems is not a straight-forward or intuitive task. For the last several decades, tens of thousands of people, usually very intelligent and talented have been involved in the building of computer systems. It is now well-known that the rate at which the hardware has been more and more accessible and at lower and lower prices, has created a matching demand for development of software in a similar scale. But the traditional intuitive and ad-hoc approach fails miserably when the quantities of data involved in information systems exceeds say, 10 MB. This is a typical figure at which systems start crossing the barriers of relatively simple and begin to enter the domain of significant complexity.

This has lead to the coining of the phrase "software crisis", and the search for methods and techniques to be able to cope with the ever expanding demands for software. The present course, which is an attempt to teach the ingredients of a structured systems development methodology, and elsewhere in the programme there is a reference to the techniques of software engineering as well. Later on in the subsequent years of the MCA programme, you would also be exposed to a full course on Software Engineering.

However, it is still useful and desirable to have some feel for the kinds of problems which the programmer and the user faces and collectively perceive as the software crisis.

Software crisis can be broadly classified in the following major areas:

### 1.4.1 From Programmer's Point of View

The following types of problems may contribute in maximum cases to software crisis:

- Problem of compatibility
- Problem of portability
- Problem in documentation
- Problem in coordination of work of different people where a team is initiating to develop software.
- Problems that arise during actual run time in the organisation. Some time the errors are not detected during sample run.
- Problem of piracy of software.
- Customers normally expand their specifications after program design and implementation has taken place.
- Problem of maintenance in proper manner.

### 1.4.2 From User's Point Of View

There are many sources of problems that arise out of the user's end. Some of these are as follows:

- How to choose a software from total market availability
- How to ensure which software is compatible with his hardware specifications
- The customerised software generally does not meet his total requirements
- Problem of virus
- Problem of software bugs, which comes to knowledge of customer after considerable data entry
- Certain softwares run only on specific operating system environment
- The problem of compatibility for user may be because of different size and density of floppy diskettes.
- Problem in learning all the facilities provided by the software because companies give only selective information in manual
- Certain software run and create files which expand their used memory spaces and create problem of disk management.
- Software crisis develops when system memory requirement of software is more than the existing requirements and/or availability.

- Problem of different versions of software (user as well as operating system).
- Security problem for protected data in software.

## 1.5 ROLE OF A SYSTEMS ANALYST

### 1.5.1 Who is Systems Analyst?

A systems analyst is a person who conducts a study, identifies activities and objectives and determines a procedure to achieve the objectives. Designing and implementing systems to suit organisational needs are the functions of the systems analyst. He plays a major role in seeing business benefit from computer technology. The analyst is a person with unique skills. He uses these skills to coordinate the efforts of different type of persons in an organisation to achieve business goals.

### 1.5.2 What a Systems Analyst does?

A system analyst carries out the following job:

(a) The first and perhaps most difficult task of systems analyst is problem definition. Business problems are quite difficult to define. It is also true that problems cannot be solved until they are precisely and clearly defined.

(b) Initially a systems analyst does not know how to solve a specific problem. He must consult with managers, users and other data processing professionals in defining problems and developing solutions. He uses various methods for data gathering to get the correct solution of a problem.

(c) Having gathered the data relating to a problem, the systems analyst analyses them and thinks of plan to solve it. He may not come up personally with the best way of solving a problem but pulls together other people's ideas and refines them until a workable solution is achieved.

(d) Systems analysts coordinate the process of developing solutions. Since many problems have number of solutions, the systems analyst must evaluate the merit of such proposed solution before recommending one to the management.

(e) Systems analysts are often referred to as planners. A key part of the systems analyst's job is to develop a plan to meet the management's objectives.

(f) When the plan has been accepted, systems analyst is responsible for designing it so that management's goal could be achieved. Systems design is a time consuming, complex and precise task.

(g) Systems must be thoroughly tested. The systems analyst often coordinates the testing procedures and helps in deciding whether or not the new system is meeting standards established in the planning phase.

### 1.5.3 Attributes of an effective Systems Analyst

Systems analyst must have the following attributes:

(a) **Knowledge of people:** Since a systems analyst works with others so closely, he or she must understand their needs and what motivates them to develop systems properly.

(b) **Knowledge of Business functions:** A systems analyst must know the environment in which he or she works. He must be aware of the peculiarities of management and the users at his installation and realize how they react to systems analyst. A working knowledge of accounting and marketing principles is a must since so many systems are built around these two areas. He must be familiar with his company's product and services and management's policies in areas concerning him.

(c) **Knowledge of Data processing principles:** Most systems today are computer based. The systems analyst must fully aware about the potential and limitations of computers.

(d) **Ability to communicate:** As a coordinator, a systems analyst must communicate properly with people of different levels within an organisation. Systems analyst must listen carefully to what others say and integrate the thoughts of others into the systems development process.

(e) **Flexibility:** Systems analysts must be flexible in their thinking since they often do not get their own way. Different factions in an organisation have conflicting needs and most systems are the result of compromise. The analyst's goal is to produce the system that will be the best for his organisation. This requires an open mind and flexibility in his ideas.

(f) **An analytical mind:** It takes an unusual person to see through problems facing an organisation and develop solutions that will work. Systems analysts often find themselves with more data than they can cope with. It requires an analytical mind to select pertinent data and concentrate on them in defining problems and forming solutions.

(g) **Well educated with sharp mind:** Systems analysts are called upon to work with people at all levels virtually in every aspect of business. They must know how to work with all of them and gain their confidence. Analysts must have sharp mind to learn quickly how people do their jobs and develop ways for them to do it better.

Check Your Progress 3

1. What do you understand by software crisis ?

..........................................................................................

..........................................................................................

..........................................................................................

2. Suppose a system memory requirement is more than the available memory size. Will you call it a software problem although the crisis is with the hardware? Why?

..........................................................................................

..........................................................................................

..........................................................................................

3. Which is in your opinion the most difficult job of a systems analyst?

..........................................................................................

..........................................................................................

..........................................................................................

4. List three important attributes of a system analyst.

..........................................................................................

..........................................................................................

..........................................................................................

## 1.6 SUMMARY

Before going to study Systems Analysis and Systems Design, an initial overall idea should be formed by the learner about what is a system, what are the characteristics of a systems, what is systems approach, what is systems analysis and what is systems design, what are the different types of a system, etc. This unit provides an overview of systems, the components and activities in the life cycle of a system development, what are the various sources which contribute a software crisis, and in details a characteristic study of a system analyst's attributes, it's different jobs. After studying this unit, you might feel eager to go in details of systems analysis and design.

## 1.7 MODEL ANSWERS

**Check Your Progress 1**

1. The systems approach shows a set of procedure for solving a particular problem. Systems analysis is a management technique which helps in designing a new system improving an existing system or solving a system problem.

2. The four basic elements in system analysis are: Outputs, Inputs, Files and Processes.

3. Systems analysts develop several different types of information systems which depend mainly on the computers for handling business applications. This class of systems is known as Computer Based Information Systems.

4. A decision is considered unstructured if there are no clear procedures for making the decision.

**Check Your Progress 2**

1. The following activities complete the life cycle:

   (i) Preliminary investigation
   (ii) Determination of system requirements
   (iii) Design of system
   (iv) Development of software
   (v) Systems testing
   (vi) Implementation, evaluation and maintenance.

2. The three types of feasibilities studied in preliminary investigation are:

   (i) Technical feasibility
   (ii) Economic feasibility
   (iii) Operational feasibility.

3. Assessment of the manner in which the system functions, including ease of use, response time, overall reliability and level of utilization.

4. Maintenance of a system is necessary to eliminate errors in the working system during its working life and to take the system to all variations within the working environment.

**Check Your Progress 3**

1. There are many sources of problems arises out of the Programmer's end or user's end to the softwares on the systems. These contribute to software crisis.

2. Yes, it is also one kind of software problem that arises from users' ends, because the available memory size is constant and known to the system analyst.

3. The most difficult job of a system analyst is Problem-defining, as some business problems are quite difficult to define and no problem can be solved until it is precisely defined.

4. There are many attributes a systems analyst should have.

Some of which are:

(a) Knowledge of business functions

(b) Knowledge of Data processing principles

(c) Ability to communicate.

(d) Flexibility

(e) An analytical mind

# UNIT 2 PROJECT SELECTION

Structure



## 2.0 INTRODUCTION

Systems analysts do not start working on any projects they desire. They receive a lot of requests from the management for starting different type of projects. When projects are formally requested, the systems analysts, under the management's direction, conduct a preliminary investigation to analyse the reasons for the request and collect various facts to respond to the request in a systematic way. Some projects are feasible, while others many not be feasible for various reasons.

## 2.1 OBJECTIVES

After going through this unit you will be able to :

- describe different reasons for developing new systems projects
- state the different sources of project requests
- discuss how to select a project out of a number of project requests
- explain something about the preliminary investigation to see the feasibility of a project
- discuss problem classifications and definitions.

## 2.2 WHY SYSTEM PROJECTS?

Systems projects are initiated for different reasons. The most important reasons are:

(a) CAPABILITY

Business activities are influenced by an organisation's ability to process transactions quickly and efficiently. Information systems add capability in three ways:

(i) Improved processing speed: The inherent speed with which computers process data is one reason why organisations seek the development of systems projects.

(ii) Increased volume: Provide capacity to process a greater amount of activity, perhaps to take advantage of new business opportunities.

(iii) Faster retrieval of information: Locating and retrieving information from storage. The ability in conducting complex searches.

(b) CONTROL

(i) Greater accuracy and consistency: Carrying out computing steps, including arithmetic, correctly and consistently.

(ii) Better security: Safeguarding sensitive and important data in a form that is accessible only to authorised personnel.

(c) COMMUNICATION

(i) Enhanced communication: Speeding the flow of information and messages between remote locations as well as within offices. This includes the transmission of documents within offices.

(ii) Integration of business areas: Coordinating business activities taking place in separate areas of an organisation, through capture and distribution of information.

(d) COST

(i) Monitor costs: Tracking the costs of labour, goods and overhead is essential to determine whether a firm is performing in line with expectations - within budget.

(ii) Reduce costs: Using computing capability to process data at a lower cost than possible with other methods, while maintaining accuracy and performance levels.

(e) COMPETITIVENESS:

(i) Lock in customers: Changing the relationship with and services provided to customers in such a way that they will not think of changing suppliers.

(ii) Lock out competitors: Reducing the chances of entering the competitors in the same market because of good information systems being used in the organisation.

(iii) Improve arrangements with suppliers: Changing the pricing, service or delivery arrangements, or relationship between suppliers and the organisation to benefit the firm.

(iv) New product development: Introducing new products with characteristics that use or are influenced by information technology.

## 2.3 SOURCES OF PROJECT REQUESTS

There are mainly four primary sources of project requests. The requesters inside the organisation are: Department Managers, Senior Executives and Systems Analysts. In addition, government agencies outside the organisation may also ask for information systems projects.

### 2.3.1 Requests from Department Managers

Frequently, department managers who deal with day-to-day business activities, are looking for assistance within their departments. They are often not satisfied with the amount of time that the staff takes to complete the job. Sometimes, they feel that the staff members are involved in duplication of work also. In this case, the manager will discuss this problem with other administrators regarding their clerical as well as processing work and persuade higher authority to approve the development of a computer based system for office administration.

### 2.3.2 Requests from Senior Executives

Senior executives like presidents, vice-presidents usually have more information about the organisation as compared to department managers. Since these executives manage the entire organisation, so naturally they have broader responsibilities. Obviously, systems project requests submitted by them carry more weightage and are generally broader in scope also.

### 2.3.3 Requests from System Analysts

Sometimes systems analysts find areas where it is possible to develop projects. In such cases, they may prefer either writing systems proposal themselves or encouraging a manager to allow the writing of a proposal on their behalf. For instance, in an organisation, an analyst sees that the library information system takes more time in processing and is inefficient, may prepare a project proposal for a new library information system. By the direction of the analyst who is fully aware about the new technology that improves the existing library information system, the librarian may initiate the development of information system to the higher authority for approval.

### 2.3.4 Requests from Outside Groups

Developments outside the organisation also lead to project requests. For example, government contractors are required to use special cost accounting systems with government stipulated features. Generally, it has been observed that new demands from external groups bring about project requests, either for new systems or changes in current ones. Project requests originated from this source are also quite important.

**Check Your Progress 1**

1. Name some important reasons for system projects.

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

2. What are the three ways by which Information systems and capability are related with each other?

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

3. Name some primary sources of project requests.

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

4. Discuss some causes due to which a Department Manager request for development of a computer based system for his office.

   ..........................................................................................

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

5. How 'communication' is also to be considered to be a reason for the initiation of system projects?

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

## 2.4 MANAGING PROJECT REVIEW AND SELECTION

It is true that a number of requests for systems development are generated in the organisation. Someone in the organisation must decide which requests to pursue and which to reject. The criteria to accept or reject a request can be decided in a number of ways. One of the suitable methods commonly in use is by committee. Mainly three committees formats are commonly used :

(i) Steering Committee

(ii) Information Systems Committee

(iii) User-Group Committee

### 2.4.1 Steering Committee

This is one of the most common methods of reviewing and selecting projects for development. Such a committee, consisting of key managers from various departments of the organisation as well as members of information systems group, is responsible for supervising the review of project proposals. This committee receives requests for proposal and evaluates them. The main responsibility of the committee is to take decision, which often requires more information than the proposal provides. It is, therefore, desired to have preliminary investigation to gather more details. The steering committee approach is generally favoured because systems projects are considered as business investments. Management, not systems analysts or designers, selects projects for development. Decisions are made on the basis of the cost of the project, its benefit to the organisation and the feasibility of accomplishing the development within the limits of information systems technology.

### 2.4.2 Information Systems Committee

In some organisations, the responsibility for reviewing project requests is entrusted to a committee of managers and analysts in the information systems department. Under this method, all requests for service and development are submitted directly to a review committee within the information systems department. This committee approves or disapproves projects and sets priorities, indicating which projects are most important and should receive immediate attention. This method can be used when many requests are for routine services or maintenance on existing applications. When major equipment decisions are required or when long- term development commitments are needed to undertake a project, the decision authority is shared with senior executives who decide finally whether a project should proceed or not.

### 2.4.3 User-Group Committee

In some organisations, the responsibility for project decisions is entrusted to the users themselves. Individual departments hire their own analysts and designers who handle project selection and carry out development. Although the practice of having user committees both choose and develop systems does take some of the burden from the systems development group, it can have disadvantages for the users. Some user groups may find themselves with defective or poorly designed systems that require additional time and effort to undo any damage caused by the mis-information that such systems could generate. Although user groups may find the decisions of steering committees and information systems committees disappointing at times, the success rate for users who undertake development job is not very encouraging.

### 2.4.4 The Project Request

The project proposals submitted by the users or the analysts to the project selection committee is a critical element in launching the systems study. There is a general agreement that a project request form should contain the following:

- What is the problem?
- What are the details of the problem?
- How significant is the problem?
- What does user feel is the solution?
- How will the information systems help?
- Who else knows about this and could be contacted?

The project selection committee is responsible to review the proposals carefully and finally selects those projects which are most beneficial to the organisation. Therefore, a preliminary investigation is often requested to gather details which are asked in the project request forms.

## 2.5 PRELIMINARY INVESTIGATION

The first step in the system development life cycle is the preliminary investigation to determine the feasibility of the system. The purpose of the preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the business system in all respect. Rather, it is the collecting of information that helps committee members to evaluate the merits of the project request and make an informed judgement about the feasibility of the proposed project.

Analysts working on the preliminary investigation should accomplish the following objectives:

- clarify and understand the project request.
- determine the size of the project.
- assess costs and benefits of alternative approaches.
- determine the technical and operational feasibility of alternative approaches.
- report the findings to management, with recommendations outlining the acceptance or rejection of the proposal.

### 2.5.1 Conducting the Investigation

The data that the analysts collect during preliminary investigations are gathered through three primary methods : reviewing organisation documents, on-site observations and conducting interviews.

#### Reviewing Organisation Documents

The analysts conducting the investigation first learn about the organisation involved in, or affected by the project. For example, to review an inventory systems proposal means knowing first how the department works and who are the persons directly associated with inventory system. Analysts can get some details by examining organisation charts and studying written operating procedures. The procedures clearly define various important steps involved in receiving, managing and dispensing stock.

### On-site Observations

Another important technique to collect data is on-site observation. In this method, the analysts observe the activities of the system directly. One purpose of on-site observation is to get as chose as possible to the real system being studied. During on-site observation, the analyst can see the office environment, work load of the system and the users, methods of work and the facilities provided by the organisation to the users.

### Conducting interviews

Written documents and the on-site observation technique tell the analysts how the system should operate, but they may not include enough details to allow a decision to be made about the merits of a systems proposal, nor do they present user views about current operations. Analysis use interviews to learn these details. Interviews allow analysts to learn more about the nature of the project request and the reason for submitting it. Interview should provide details that further explain the project and show whether assistance is merited economically, operationally and technically.

## 2.5.2 Testing Project Feasibility

Preliminary investigations examine project feasibility, the likelihood the system will be useful to the organisation. Three important tests of feasibility are studied and described below:

- operational feasibility
- technical feasibility
- economic feasibility

### Operational Feasibility

Proposed projects are beneficial only if they can be turned into information systems that will meet the operating requirements of the organisation. This test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to implementation? Some of the important questions that are useful to test the operational feasibility of a project are given below:

- Is there sufficient support for the project from the management? From users? If the present system is well liked and used to the extent that persons will not be able to see reasons for a change, there may be resistance.
- Are current business methods acceptable to the users? If they are not, users may welcome a change that will bring about a more operational and useful system.
- Have the users been involved in the planning and development of the project? If they are involved at the earliest stage of project development, the chances of resistance can be possibly reduced.
- Will the proposed system cause harm? Will it produce poorer result in any case or area? Will the performance of staff member fall down after implementation?

Issues that appear to be quite minor at the early stage can grow into major problem after implementation. Therefore, it is always advisable to consider operational aspects carefully.

### Technical Feasibility

There are a number of technical issues which are generally raised during the feasibility stage of the investigation. They are as follows:

- Does the necessary technology exist to do what is suggested (and can it be acquired)?
- Does the proposed equipment have the technical capacity to hold the data required to use the new system?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

**Economic Feasibility**

A system that can be developed technically and that will be used if installed, must still be profitable for the organisation. Financial benefits must equal or exceed the costs. The analysts raise various financial and economic questions during the preliminary investigation to estimate the following:

- The cost to conduct a full systems investigation.
- The cost of hardware and software for the class of application being considered.
- The benefits in the form of reduced costs or fewer costly errors.
- The cost if nothing changes (i.e. the proposed system is not developed).

To be judged feasible, a proposal for the specific project must pass all these tests. Otherwise, it is not considered as a feasible project.

### 2.5.3 Handling Infeasible Projects

It is not necessary that all projects that are submitted for evaluation and review are acceptable. In general, requests that do not pass all the feasibility tests are not pursued further, unless they are modified and re-submitted as new proposals. In some cases, it so happens that a part of a newly developed system is unworkable and the selection committee may decide to combine the workable part of the project with another feasible proposal. In still other cases, preliminary investigations produce enough new information to suggest that improvements in management and supervision, not the development of information systems, are the actual solutions to reported problems.

## 2.6 PROBLEM CLASSIFICATIONS AND DEFINITIONS

One of the most difficult tasks of system analysis is developing a clear, in-depth understanding of the problem being investigated, without which it becomes impossible to specify the requirements for a new project with any accuracy. Several questions should be posed for this. Some of those may be:

i) What is the problem?

ii) How complex is it?

iii) What are its likely causes?

iv) Why is it important that the problem be solved?

v) What are possible solutions to the problem?

vi) What types of benefits can be expected once the problem is solved?

### 2.6.1 Defining a Problem

It takes considerable skill to determine the true cause of a systems problem. A systems analyst might begin to define the problem by determining if the problem can be classified according to one or more common types of systems problems. With a knowledge of the common types of problems, the analyst can diagnose a problem by examining its characteristics. The following example illustrates this finding.

A manager comments, "We need a new budgeting system. Our current one seems to vary in quality from one month to the next. Besides, reports are often late, have errors, and contain misleading information. Why we must spend a fortune simply trying to keep the system up and going."

Careful analysis of this statement suggests a number of different problems, the problem of reliability (the system varies in quality from one month to the next), the problem of accuracy (there are too many errors), the problem of timeliness (reports are often late), the problem of validity (reports contain misleading information), and the problem of economy (the system is costly to keep up and going).

Besides the problems of reliability, validity, accuracy, economy and timeliness, the problems of capacity and throughput are also common. Capacity problems occur when a component of a system is not large enough. Two people attempting to do the work of six illustrates a capacity problem. Throughput problems deal with the efficiency of a component of a system. Six people doing the work of two represents a problem of throughput. Let's consider each of these seven problems in more detail.

(a) **The Problem of Reliability:** A system suffers from the problem of reliability when procedures work some but not all of the time, or when use of the same procedure leads to different results. Analysts must work continually to improve the reliability of systems. They strive to do this by running software tests to document that two runs of a computer program lead to identical results, by selecting equipment with low failure rates, and by monitoring processing schedules to ensure that results are on time. With some systems, reliability is essential. Imagine a payroll system that only works some of the time, or for that matter the railway or airlines reservation system.

(b) **The Problem of Validity:** Systems that produce invalid results are often most troublesome to users and systems managers. These systems might be highly reliable. They may work all of the time,but they draw incorrect conclusions. A report might show that demand is increasing and that additional stock should be ordered for inventory. If these conclusions are wrong and demand is actually decreasing, then the stock is unnecessary and the whole operation becomes less efficient.

Maintaining validity in computer software is a troublesome design problem. The objective in design is to produce a flawless product, one that will always reflect actual events. Validity problems result when the environment changes and these changes are not incorporated into the software. As an example, suppose a measure of consumer satisfaction must be placed in a computer program. If the measure is incorrect, the software will draw incorrect conclusions.

(c) **The Problem of Accuracy:** The problem of accuracy is similar to the problems of reliability and validity. A system is inaccurate when processing is error-prone. For example, assume that several people are required to post company expense transactions against departmental budget numbers. If the posting procedure is complex and the number of transactions large, a fair number of errors may occur (for example, 1 percent of all transactions). Because of inaccuracy, the entire budget system might be viewed as unreliable and often invalid. However, these are symptoms of the real problem - namely, the problem of accuracy. Routine, transaction-based manual procedures are basically suitable for conversion to computer-based methods of processing because the computer is far more accurate than human beings, provided that software is written properly.

(d) **The Problem of Economy:** Besides improving processing accuracy, organisations seek to improve processing economy. A system suffers from the problem of economy when existing methods of transmitting, processing, and storing information are very costly. An organisation might discover that the cost of handling the paperwork associated with each purchase order is Rs 25. This cost is determined to be a problem of economy. After the installation of a new method of processing, the cost per purchase is substantially reduced – from Rs 25 per order to Rs 8 per order.

Projects with clear-cut savings are likely to be considered suitable for conversion to computer-based methods of processing. Much like the problem of accuracy, the problem of economy is relatively easy to identify. The danger with the problem of economy is the naive assumption-by both users and system managers-that the computer will eliminate the cause of the problem. Budget managers will say that this assumption is not always true; they will report that some project cost far more than they return. Thus, before moving ahead on a project assignment, the analyst must ask, "Is the project worth doing?". A partial answer to this question follows from determining the return on the investment expected from the project. If the return is low, more economical projects should be selected.

(e) **The Problem of Timeliness:** The problem of timeliness relates more to the transmission of information than to the processing or storing of it. A system suffers from the problem of timeliness if information is available but cannot be retrieved when and where it is needed. As people become more familiar with information systems and how they function, they generally realize how much easier it is to process and store information than it is to retrieve it.

Organisations have committed extensive resources to handle the problem of timeliness in recent years. Fingertip access to information has been the desired objective. The findings to date show that only modest success has been achieved in improving this problem area. Only when retrieval problems are small and well defined has the overall success rate improved.

(f) **The Problem of Capacity:** The problem of capacity occurs when a system component is not large enough. Capacity problems are specially common in organisations that experience peak periods of business. During peak periods, inadequate processing capacity, transmission capacity, storage capacity, staff capacity, and the like may all exist. Capacity problems are also evident in rapidly growing organisations. With growth, smaller-capacity equipment soon becomes too small; smaller staff groups soon become overworked. In either case, some expansion is needed to handle the increasing volume of business.

Many system problems are directed at solving capacity problems. Because it is often difficult to justify the purchase of new equipment or the hiring of new staff, people tend to put off such decisions until the very last moment. Consequently, when the systems group is contacted, the problem of capacity is easy to spot; the difficulty, however, lies in knowing how to handle the problem. For example, an analyst might be forced to suggest a short-term solution to the problem. This is done to gain time toward the formulation of a longer-term solution. For instance, an analyst might recommend: "Let's hire five part-time employees to help us get through the peak period." When a short-term approach fails, the analyst may be tempted to implement a quick-fix computer-based solution. Unfortunately, this solution carries with it the associated danger of creating an even more severe system problem in the near future.

(g) **The Problem of Throughput:** The problem of throughput may be viewed as the reverse of the problem of capacity. Throughput deals with the efficiency of a system. If system capacity is high and production low, a problem of throughput occurs. Consider the following example.

Five programmers are assigned to a fairly straightforward programming assignment consisting of 10,000 lines of computer code. After thirty days of coding, the programming team is evaluated. It is discovered that they have completed 6000 usable lines of code. Now, if each programmer worked eight hours a day, a total of 1200 hours would have been expended on the project. Calculated differently, the average production rate for each programmer would be 5 lines of code per hour (6000 lines divided by 1200 hours). These findings might lead the analyst to conclude that there is a problem of throughput.

Similar to the problem of capacity, the problem of throughput may be much easier to spot than to treat. When repeated equipment breakdowns lead to low rates of production (and when the equipment has been purchased and cannot be returned), an organisation can badger the vendor into fixing the equipment but can achieve little more short of legal action. Likewise, when groups of people exhibit low rates of production, such as the five-person programming team, the problem becomes even more complicated. Badgering and threats may not work at all. Rather, a manager must be able to determine the root of the problem for any improvement in throughput.

### 2.6.2 Evaluating the Problem

Suppose that a problem has been identified. The next step is problem evaluation, which consists of asking the following questions: Why is it important to solve the problem? What are possible solutions to the problem? What types of benefits can be expected once the problem is solved? There will be times when an analyst will recommend that no project be started to resolve a problem, as the next example demonstrates.

Suppose that an analyst discovers that the real problem lies with the supervisor of an area. Because of mistakes made by this man, the throughput rate is 20 percent less than had been expected. However, suppose next that the supervisor is new to the job, is smart enough to realize where mistakes were made, and knows how not to repeat them in the future. Given this situation, the analyst might close the book on this project, recommending that no action be taken at this time.

Consider a different set of circumstances. Suppose that an analyst determines that a problem of low throughput can be traced to a computer printer. Suppose further that the problem must be corrected. Once the problem has been identified, the analyst would prepare a solutions table to list possible problem solutions and the expected benefits from each. Sometimes, the best solution is not at all evident. The analyst might recommend that further study is required to determine which of the possible solutions is best.

In this section, we have spent considerably more time examining how an analyst identifies a problem compared with how the problem is evaluated. This uneven split also occurs in practice. As a general rule, analysts spend 75 percent of the project-definition phase of analysis defining the problem and 25 percent evaluating and documenting their findings. Note also that we have limited our discussion to seven major types of system problems. Because of this limitation, you might ask, "What about the problems of communication" of group conflict? of management? of system security? Are these problems as well? Are these types of problems also evaluated by the analyst?" Although our discussion has been restricted to more technical system problems, individual or group problems also occur in a systems environment and require identification and evaluation.

Still another limitation is the coverage given to determining the feasibility of taking some action to solve a problem. The concept of feasibility entails the joint questions of "Can something be done?" and, if so, "Should it be done given a particular set of circumstances?" For example, is it possible to climb a mountain when we have at our disposal only a forty - foot rope? If it is, a second question is well advised, namely, "Should we attempt such a climb given the size of our rope?" We will examine the question of project feasibility in more detail in the next unit of this block.

A final limitation is the coverage given to tools which the analyst can use to identify and evaluate system problems. These tools are needed when the problems are not self-evident.

Organisations face various types of problems during their course of operations and come across opportunities or situations which could be converted into profitable solutions. Whenever there is an opportunity and/or problem in the existing system of operations or when a system is being developed for the first time, the organisation considers designing a new system for information processing.

### 2.6.3 Sources of Problem/Opportunity

Organisations usually face problems or have opportunity due to the following:

- a new product or plant or branch
- a new market or new process
- failure of an existing system
- inefficiency of an existing system
- structural error in the existing system, etc.

Thus a thorough analysis of the situation need to be required. Not only the above listed reasons but there exist some organisation based reasons too.

### 2.6.4 Problem Identification and Definition

For identifying problems/opportunities, we scan the following:

- the performance of the system
- the information being supplied and its form
- the economy of processing
- the control of the information processing

- the efficiency of the existing system
- the security of the data and software
- the security of the equipment and personnel, etc.

After identification of the problem, it is defined and a general direction or method for solving this problem is also determined. Then project boundaries are defined. The management establishes the term of reference as well as the resources to be provided for the project. System development is an iterative process and the first identifiable stage of it is Problem Definition, whose final output is Terms of Reference.

Check Your Progress 2

1. Name the three committees by which review of projects can be done.

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

2. A project request form should contain information to some basic questionnaire. List some of those.

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

3. (a) What are the objectives the analysts should accomplish during preliminary investigation?

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

   (b) How the data are gathered?

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

4. Which are the causes for which organisations usually face problem (or have opportunity) ?

   ..........................................................................................

   ..........................................................................................

   ..........................................................................................

## 2.7 SUMMARY

In this unit, we have discussed first of all the various possible reasons for system projects. You know here what is the necessity of system projects, why system projects are initiated etc.. Then comes in section 2.3 the various sources who initiate system projects and the reasons for system projects from different angles(In the previous section a general discussion was made; here you study the reasons for system projects which vary from its source to source). Now suppose, project proposals are submitted. How to make a good review of all the projects and how to select or reject project proposal? These are discussed in section 2.4. For this purpose some committees(mainly three) are there. There role/activities are also discussed. The very first step in the System Development Life Cycle is the preliminary investigation to analyse the feasibility of the system. There are different stages to determine the overall feasibility, and you study this in section 2.5. In the section 2.6 various types of problems are pointed out and defined. For your own satisfaction you get some straightforward questions under Check Your Progress 1 and 2, whose model answers you find below.

## 2.8 MODEL ANSWERS

**Check Your Progress 1**

1. Some important reasons for system projects are capability, control, communication, cost and competitiveness.

2. Information system and capability are related to each other in three ways:

   (i) Improved processing speed

   (ii) Increased volume of activities

   (iii) Faster retrieval of information

3. Some primary sources of project requests are:

   (i) Requests from department managers

   (ii) Requests from senior executives

   (iii) Requests from system analysts

   (iv) Requests from outside groups.

4. A Department Manager requests for development of a computer based system for his office for various reasons. The most important reason in case of any manager is the amount of time taken to accomplish different jobs. Certainly, if his office is computerised he will be able to have more output at the cost of equal man-hour. Similarly, Job scheduling, parallel works etc. can also be done by his computer based system, specially when the data to be processed are very large.

5. For speeding up the transmission of important messages and information within his office as well as in remote branches of his organization, "communication" is considered to be a reason for initiating a system project.

**Check Your Progress 2**

1. The main three committees are Steering Committee, Information System Committee and User-Group Committee.

2. A project request form should contain some basic questionnaire. Some of those may be the following:

   (i) What is the problem?

   (ii) What are the details of the problem?

   (iii) How significant is the problem?

   (iv) What does the user feel is the solution?

   (v) How will the information systems help?

   (vi) Who else knows about this and could be contacted?

# UNIT 3 FEASIBILITY STUDY

**Structure**



## 3.0 INTRODUCTION

Once a preliminary area of application has been identified, it may then be subjected to a more rigorous examination in a feasibility study. In the previous unit; we discussed the steps that make up the initial investigation. By the initial investigation, a user has recognized the need, user requirements are determined and the problem has been defined. Apart from this, an initial investigation is launched to study the present system and verify the problem in a systematic way. The next step is to determine exactly what the proposed system is to do by defining its expected performance. This kind of work will be carried out in the feasibility study. A feasibility study is carried out to select the best system that meets performance requirements.

## 3.1 OBJECTIVES

After going through this unit, you should be able to:

- explain what is known as feasibility, and what is feasibility study
- list and illustrate different types of feasibility
- discuss the purposes of feasibility study
- describe different steps in feasibility analysis
- explain in details cost/benefit analysis

## 3.2 PRELIMINARY STUDY

Feasibility is the determination of whether or not a project is worth doing. The process followed in making this determination is called a feasibility study. This type of study determines if a project can and should be taken. Once it has been determined that a project is feasible, the analyst can go ahead and prepare the project specification which finalizes project requirements. Generally, feasibility studies are undertaken within tight time constraints and normally culminate in a written and oral feasibility report. The contents and recommendations of such a study will be used as a sound basis for deciding whether to proceed, postpone or cancel the project. Thus, since the feasibility study may lead to the commitment of large resources, it becomes necessary that it should be conducted competently and that no fundamental errors of judgment are made.

## 3.3 DIFFERENT TYPES OF FEASIBILITY

In subsection 1.3.1 of unit1, you have noted that an important outcome of the preliminary investigation is the determination whether the system requested is feasible or not. That requires the need for a rigorous feasibility study.

In the conduct of the feasibility study, the analyst will usually consider seven distinct, but inter-related types of feasibility. They are:

(1) Technical feasibility

(2) Operational feasibility

(3) Economic feasibility

(4) Social feasibility

(5) Management feasibility

(6) Legal feasibility

(7) Time feasibility

### 3.3.1 Technical feasibility

This is concerned with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may vary considerably, but might include:

- The facility to produce outputs in a given time.
- Response time under certain conditions.
- Ability to process a certain volume of transaction at a particular speed.
- Facility to communicate data to distant location.

In examining technical feasibility, configuration of the system is given more importance than the actual make of hardware. The configuration should give the complete picture about the system's requirements: How many workstations are required, how these units are interconnected so that they could operate and communicate smoothly. What speeds of input and output should be achieved at particular quality of printing. This can be used as a basis for the tender document against which dealers and manufacturers can later make their equipment bids. Specific hardware and software products can then be evaluated keeping in view with the logical needs.

At the feasibility stage, it is desirable that two or three different configurations will be pursued that satisfy the key technical requirements but which represent different levels of ambition and cost. Investigation of these technical alternatives can be aided by approaching a range of suppliers for preliminary discussions. Out of all types of feasibility, technical feasibility generally is the most difficult to determine.

### 3.3.2 Operational feasibility

It is mainly related to human organizational and political aspects. The points to be considered are:

- what changes will be brought with the system?
- what organizational structures are disturbed?
- what new skills will be required? Do the existing staff members have these skills? If not, can they be trained in due course of time?

Generally project will not be rejected simply because of operational infeasibility but such considerations are likely to critically affect the nature and scope of the eventual recommendations. This feasibility study is carried out by a small group of people who are familiar with information system techniques, who understand the parts of the business that are relevant to the project and are skilled in system analysis and design process.

### 3.3.3 Economic feasibility

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as cost/benefit analysis; the procedure is to determine the benefits and savings that are expected from a proposed system and compare them with costs. If benefits outweigh costs, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have a chance of being approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle.

### 3.3.4 Social feasibility

Social feasibility is a determination of whether a proposed project will be acceptable to the people or not. This determination typically examines the probability of the project being accepted by the group directly affected by the proposed system change.

### 3.3.5 Management feasibility

It is a determination of whether a proposed project will be acceptable to management. If management does not accept a project or gives a negligible support to it, the analyst will tend to view the project as a non-feasible one.

### 3.3.6 Legal feasibility

Legal feasibility is a determination of whether a proposed project infringes on known Acts, Statutes, as well as any pending legislation. Although in some instances the project might appear sound, on closer investigation it may be found to infringe on several legal areas.

### 3.3.7 Time feasibility

Time feasibility is a determination of whether a proposed project can be implemented fully within a stipulated time frame. If a project takes too much time it is likely to be rejected.

## 3.4 INVESTIGATIVE STUDY

### 3.4.1 Steps in feasibility analysis

Eight steps are involved in the feasibility analysis. They are:

(i) Form a project team and appoint a project leader.

(ii) Prepare system flowcharts.

(iii) Enumerate potential proposed systems.

(iv) Define and identify characteristics of proposed system.

(v) Determine and evaluate performance and cost effectiveness of each proposed system.

(vi) Weight system performance and cost data.

(vii) Select the best proposed system.

(viii) Prepare and report final project directive to management.

### 3.4.2 Analyzing systems data

After gathering sufficient data to understand how the existing system operates, a proper study on data should be made for evaluating the current operations.

Systems analysis is fact finding followed by analysis of the facts. Data analysis is also considered a pre-requisite condition for cost/benefit analysis. System investigation and data gathering lead to an assessment of current findings. Our interest is in determining how efficiently certain steps are performed to achieve intended goals and the cost of making improvements.

The details of the system learned by the analyst during the investigation tell what is happening, how it is done, when it is carried out. These details help the analyst to evaluate the current system. System analyst tries to find out the efficiency of certain steps and how they contribute to achieve the intended result. After examining the facts collected about the system, the analyst develops a profile of each application area. The systems profile consists of details describing the operating characteristics of the system, such as frequency of occurrence, volume of work or error rate. The analysis of details collected during the investigation phase indicates that are serious gaps in control and a bottleneck exists for processing claims.

### 3.4.3 Identifying design requirements

From the analysis, design requirements are formulated. The requirements for the new system are those features that must be incorporated to produce the improvements. These requirements are determined by comparing current performance with the objectives for acceptable systems performance. The new system should have the following features:

(a) Greater speed of processing

(b) Effective procedure to eliminate errors

(c) Better accuracy

(d) Faster retrieval of information

(e) Integration of data

(f) Larger capacity of storing data with reduced cost

To achieve these features, several alternatives must be studied and evaluated. One alternative may not satisfy all the features. The analyst then selects those that are feasible economically, technically and operationally. The approach may emphasize the introduction of computerised system, replacement staff, changes in operating procedures, or a combination of several options.

The analyst often suggests inputs, process, reporting and control procedures to help the management in decision making techniques. The procedures may be manual or automated but these will be useful in meeting systems requirements. Management will decide whether to accept and use them.

The role of a computer in a design revolves round its capabilities for calculation, storage and retrieval of data, summarizating, sorting, classification and communication of data. The analyst must decide about the speed and storage capacity of a computer required for achieving the design objectives. The analyst does this by matching the computer capabilities with an understanding of the systems requirements.

A new system might, for example, call for the automation of invoice handling so that the invoice could be classified and processed as soon as it is received. All these steps can take place by entering the invoice number, purchase order number and vendor identification through a terminal. The computer in turn can be substituted for human processing. These processes could be faster and accounting balances can be incorporated into the procedure. The results of day's work can be summarized and communicated to supervisors, whether they are sitting in the same building or miles away.

As you know that each approach has its benefits and drawbacks, depending on the particular business situation. Therefore, the analyst selects those alternatives most workable and studies them further and make decision which alternative should be selected? Cost and benefit analysis of each alternative further guides the selection process. Therefore, the analyst needs to be familiar with the cost and benefit categories and the evaluation of various methods before a final selection can be made. This is discussed in the next section.

**Check Your Progress 1**

1. What do you mean by feasibility? What is feasibility study?
2. What are the seven types of feasibility?
3. Which are the technical feasibility concerned with?
4. Name some points which are to be considered in the operational feasibility.

## 3.5 COST/BENEFIT ANALYSIS

Since cost plays quite an important role in deciding the new system, it must be identified and estimated properly. Costs vary by type and consist of various distinct elements. Benefits are also of different type and can be grouped on the basis of advantages they provide to the management. The benefits of a project include four types:

(i) Cost-savings benefits

(ii) Cost-avoidance benefits

(iii) Improved-service-level benefits

(iv) Improved-information benefits

Cost-savings benefits lead to reductions in administrative and operational costs. A reduction in the size of the clerical staff used in the support of an administrative activity is an example of a cost-saving benefit.

Cost-avoidance benefits are those which eliminate future administrative and operational costs. No need to hire additional staff in future to handle an administrative activity is an example of a cost-avoidance benefit.

Improved-service-level benefits are those where the performance of a system is improved by a new computer-based method. Registering a student in fifteen minutes rather than an hour is an example of this third type of benefit.

Improved-information benefits is where computer based methods lead to better information for decision making. For example, a system that reports the most-improved fifty customers, as measured by an increase in sales is an improved-information. This information makes it easier to provide better service to major customers.

**Categories of Costs and Benefits:**

The costs associated with the system are expenses, outlays or losses arising from developing and using a system. But the benefits are the advantages received from installing and using this system.

Costs and benefits can be classified as follows:

(a) Tangible or intangible

(b) Fixed or variable

(c) Direct or indirect

### 3.5.1 Tangible or intangible costs and benefits.

Tangibility refers to the ease with which costs or benefits can be measured. An outlay of cash for any specific item or activity is referred to as a tangible cost. These costs are known and can be estimated quite accurately.

Costs that are known to exist but their financial value cannot be exactly measured are referred to as intangible costs. The estimate is only an approximation. It is difficult to fix exact intangible costs. For example, employee movable problems because of installing new system is an intangible cost. How much moral of an employee has been affected cannot be exactly measured in terms of financial values.

Benefits are often more difficult to specify exactly than costs. For example, suppliers can easily quote the cost of purchasing a terminal but it is difficult for them to tell specific benefits or financial advantages for using it in a system. Tangible benefits such as completing jobs in fewer hours or producing error free reports are quantifiable. Intangible benefits such as more satisfied customers or an improved corporate image because of using new system are not easily quantified. Both tangible and intangible costs and benefits should be taken into consideration in the evaluation process. If the project is evaluated on a purely intangible basis, benefits exceed costs by a substantial margin, then we will call such project as cost effective. On the other hand, if intangible costs and benefits are included, the total costs (tangible+intangible) exceed the benefits which makes the project an undesirable investment. Hence, it is desirable that systems projects should not be evaluated on the basis of intangible benefits alone.

### 3.5.2 Direct or Indirect costs and benefits

Direct costs are those which are directly associated with a system. They are applied directly to the operator. For example, the purchase of floppy for Rs.400/- is a direct cost because we can associate the floppy box with money spent.

Direct benefits also can be specifically attributable to a given project. For example, a new system that can process 30 per cent more transactions per day is a direct benefit.

Indirect costs are not directly associated with a specific activity in the system. They are often referred to as overhead expenses. For example, cost of space to install a system, maintenance of computer centre, heat, light and air-conditioning are all tangible costs, but it is difficult to calculate the proportion of each attributable to a specific activity such as a report.

Indirect benefits are realized as a by-product of another system. For example, a system that tracks sales calls on customers provides an indirect marketing benefit by giving additional information about competition. In this case, competition information becomes an indirect benefit although its work in terms of money cannot be exactly measured.

### 3.5.3 Fixed or Variable costs and benefits

Some costs and benefits remain constant, regardless of how a system is used. Fixed costs are considered as sunk costs. Once encountered, they will not recur. For example, the purchase of an equipment for a computer centre is called as fixed cost as it remains constant whether in equipment is being used extensively or not. Similarly, the insurance, purchase of software etc. In contrast, variable costs are incurred on a regular basis. They are generally proportional to work volume and continue as long as the system is in operation. For example, the cost of computer forms vary in proportion to the amount of processing or the length of the reports desired.

Fixed benefits also remain constant. By using a new system, if 20 percent of staff members are reduced, we can call it a fixed benefit. The benefit of personnel saving may occur every month. Variable benefits, on the other hand, are realized on a regular basis. For example, the library information system that saves two minutes in providing information about a particular book whether it is issued or not, to the borrower compared with the manual system. The amount of time saved varies with the information given to the number of borrowers.

### 3.5.4 How to define Cost-benefit analysis?

We can define cost-benefit analysis as

i) that method by which we find and estimate the value of the gross benefits of a new system specification.

ii) that method by which we find and determine the increased operating costs associated with the above mentioned gross benefits.

iii) the subtraction of these operating costs from the associated gross benefits to arrive at net benefits.

iv) that method by which we find and estimate the monetary value of the development costs that produce the above mentioned benefits.

v) those methods by which we show the time-phased relationship between net benefits and development costs as they relate to cash flow, payback on investment, and time-in-process taking(or not taking) into operation factors such as inflation etc. In short, the calculation of actual net benefit as cash flowback over time.

**Check Your Progress 2**

1. Name different types of benefits.

..........................................................................................

..........................................................................................

..........................................................................................

2. Name different types of costs and benefits.

..........................................................................................

..........................................................................................

..........................................................................................

3. Give a good definition of Cost-benefit analysis.

..........................................................................................

..........................................................................................

..........................................................................................

..........................................................................................

## 3.6 FACT FINDINGS

**What is fact finding?**

Fact finding means learning as much as possible about the present system.

**How to do fact finding?**

To do fact finding, the analyst does the following:

- interviews personnel
- prepares questionnaires
- observes the current system
- gathers forms and documents currently in use
- determines the flow of data through the system, and
- clearly defines the system requirements.

### 3.6.1 Interviewing

By studying this organisation chart, the analyst can confidently schedule interviews with key personnel involved with the system. Of course, there should be preliminary interviews. Later he will conduct a detailed interview with all the people who actually operate the system. Not only will these people use the newly developed system, but they also may be the ones most afraid of change, especially if they feel the computer might replace them. Like an investigative reporter trying to discover the who, what, when, why and how of a story, the analyst should conduct the interview in such a way that people provide honest descriptions of their jobs. The following questions can help accomplish this goal.

- Who is involved with what you do?
- What do you do?
- Where do you do it?
- When do you do it?
- Why do you do it the way you do?
- How do you do it?
- Do you have suggestions for change?

Interviews help gather vital facts about existing problems, such as lack of quality control or sufficient security, but they also allow the analyst to involve people in change, easing them into it. After all, it is the users' system, not the analyst's.

### 3.6.2 Questionnaires

Questionnaires economically gather data from both large and small groups of people. Properly constructed, they do not take long to complete and statistical results can be quickly tabulated. Development of a questionnaire requires in depth planning, and usually more than one draft is necessary.

At this point of time, you may like to go to the block on 'Case Studies' which is included as part of this course. The case studies given here pertain to developing an information systems for a general gymnatorium. More specifically, you will see that in Case 'C', there is an example of a somewhat free format interview from which the crucial problem areas have been identified by the interviewer, who is the Chief Systems Analyst. But in addition, in order to get a more quantitative response of the people to the scheduling and ticketing problems considered therein, there is a questionnaire to gather additional facts to determine the nature of complaints.

You may also have had occasions to respond to such questionnaires, sometimes in newspapers or sometimes from marketing personnel, who do door to door surveys.

Questionnaire design is critical. Questions should be short, easy to understand, unbiased, nonthreatening, and specific. To make sure questions will stimulate needed information, the analyst can test them with one or two outsiders before widespread distribution. Prepaid return envelopes accompanying questionnaires sent to outside help assure prompt response.

The analyst should send questionnaires to everyone involved with the system. A questionnaire works particularly well when the analyst must gather data from a large number of people, when the analyst must ask everyone the same questions, or when facts must be collected from people, such as suppliers, who do not work for the organisation.

**Questions can follow four formats:**

i) **Multiple choice:** This gives respondents a specific set of potential answers. The format is ideal for computer tabulating.

ii) **Open ended:** Respondents must answer the question in their own words. Space is provided under each question for the response.

iii) **Rating:** This is similar to multiple choice except that respondents must rate their satisfaction.

iv) **Rank:** Rank requires respondents to prioritise their responses from high to low or on a percentage basis.

Aware that most people do not spend a lot of time responding to questionnaire. Most analysts decide to mix question formats, including follow-up questions, within the original questionnaire to permit elaboration of certain responses. By so organising a questionnaire, the respondents have an opportunity to express their opinions freely, and yet answer quickly through the use of multiple-choice, rating, and ranking questions. When all the questionnaires are returned, the data can be tabulated.

If the results of a questionnaire survey are incomplete or confusing, the analyst may want to contact selected outsiders by telephone or in person. This requires tact, of course, and an understanding that the analyst's own pressing need may not concern outsiders in the least.

### 3.6.3 Observing the current system

The analyst may want to observe the existing system personally by following transaction, such as in invoice, through it. Direct observation allows the analyst to verify his or her understanding of the system. Instead of getting second-hand impressions about a specific task, the analyst can experience the actual process. However, he or she must remain outside the flow as an observer, so as not to introduce biases or changes in actual procedures. Observing a system requires caution, when people know they are being observed. They usually behave differently, working more efficiently and at higher speeds to impress the analyst.

In some instances, the analyst may find it useful to visit another organisation with a computerised system similar to the one under study. Finding a comparable installation may pose a problem, however. Some competitive organisations may not want to share their experiences, others may be too large or too small for accurate comparisons, and still others may be unwilling to waste employees' time demonstrating their system. Whenever visiting another organisation, an analyst should follow the rules of etiquette: make an appointment, research the organisation beforehand, know what he or she wants to see, and write a follow-up, thank you letter.

Hardware and software vendors can also supply valuable information. Computer sales representatives will gladly share their experiences with potential clients, and software firms will send brochures describing their programs. Although very useful information from such sources should be reviewed carefully because vendors are more interested in promoting their products than in solving your problems.

Buying a product from a new business, such as the explosive software industry poses unusual problems. Customers cannot evaluate decades of performance history by the company, and not enjoying the benefits of an objective "consumer report" on new products, they often feel at the mercy of fast-talking salespeople. Therefore, it is important for people in the market for software to ask some really tough but relevant questions. Any reputable supplier should be able to answer the following 15 questions without backpedaling.

i) **Range of Products:** Can you offer us a complete range of software system designed to work together? Or will we have to piece together a patchwork of systems to fully computerise our organisation?

ii) **Decision Support Systems:** Are your systems just record keepers, or can they really help us make decisions? Can we pull together information from any of our integrated systems in the desired form.

iii) **In-House Development:** Can you provide business software for both mainframe and microcomputers? Do you develop this software yourself or do you simply market it for another company?

iv) **Online:** Are your systems truly online? How many of your systems are online? How secure are they?

v) **Debugging and Testing:** Will my company have to be the one that discovers the bugs in your brand new system? Just how long have your systems actually been used, and how have they been tested?

vi) **Updates:** Will you update your systems as technology advances and regulations change? What are some of your most recent updates? Will you keep us current on regulatory change?

vii) **Flexibility/Adaptability:** Are your systems really adaptable to our unique needs? Or will we have to change or add to them ourselves to get the features we want?

viii) **History/Performance:** How long have you been in business? What are your revenues? What is your growth record? Where will your company be in five years from now? Can you show me an annual report?

ix) **Other Customers:** How many systems has your company installed? How many of these were installed in the past six months? How many of your earlier customers are still using and liking your systems?

x) **Security:** Are your systems secure? Do you provide password type protection and to how many levels? What other type of security provisions do your systems have?

xi) **Networking:** Can you link our executives' personal computers directly to the mainframe, so they can get their own information? Is that software available right now?

xii) **Training Support:** How will you make sure our own people thoroughly understand your system? Do you have educational centres near us or will we have to travel all the way across the country to find one? Will you be there to help during installation and after?

xiii) **In-House Specialists:** How many of your people specialise in software for my industry? How many accountants work for you? Human resource specialists? Manufacturing experts?

xiv) **Special Features:** Do your systems have built-in features that make them easier to use? What happens if someone needs help figuring out a feature? Do you have online documentation that is easy to understand?

xv) **Upgrading:** As my business changes will your system be flexible enough to change with it? Or will we have to pay a lot to revamp it? Or even regenerate it?

### 3.6.4 Determination of DFD

Armed with interview results, tabulated questionnaires, and experience through personal observations, the analyst is ready to describe the current system in narrative form, with a data-flow diagram, or with a system flowchart. Since all organisations have an accounts payable (AP) system let us begin with such an example using a context DFD. A context DFD defines the system under study in a general form, showing:-

Inputs to AP: Packing slips, invoices, checking account balances, payment notifications.

Outputs from AP: reports to management, cheque to suppliers.

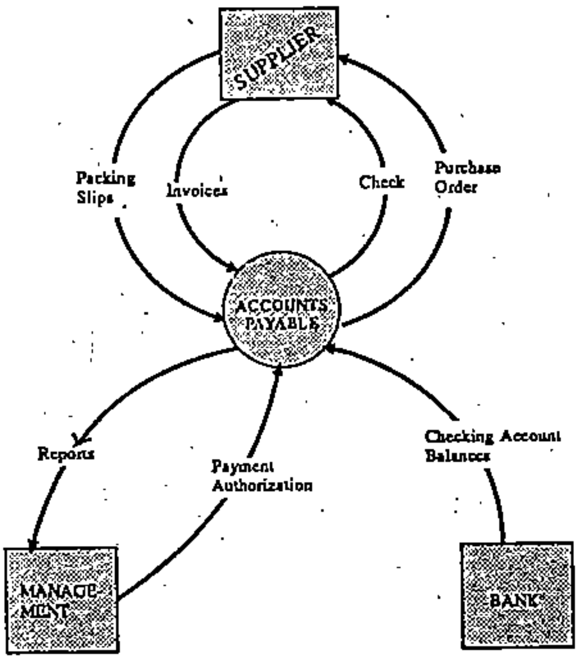



**Figure 3.1: A context data-flow diagram depicts a typical accounts payable system in its broadest perspective, not showing any of the details or internal processes.**

A context DFD does not show any detail but is an overview drawing of the system. It is an excellent diagram to share with management whose interest is general in nature. Context DFDs place a boundary around the system under investigation, saying that this is what will be examined - nothing more and nothing less.

After developing a context DFD, the analyst turns his attention to the details of accounts payable. Management reviews inventory reports and determines what to order from suppliers: orders are placed by the accounting department using a purchase order/requisition: on delivery, merchandise and packing slips enter the warehouse, and packing slips are sent to the accounting department, which receives invoices directly from suppliers, while merchandise stays in the warehouse or goes to a distribution outlet. Accounting clerks compare purchase order requisitions with invoices and packing slips to make sure all invoiced items have actually arrived, and then post the purchase to the supplier's ledger. At the end of each month, the accounting department prepares a report of balances due suppliers and an inventory report for management evaluation.

These detailed activities by the accounting department, management, warehouse personnel, the bank and suppliers add up to six major activities (Figure 3.2):

1. Generation of reports
2. Ordering of stock
3. Printing of cheque
4. Posting of accounts
5. Reconciliation of bank statements
6. Authorisation of payment

During the design phase of the systems process, the analyst will study each of these activities further, leveling the data-flow diagram of Figure . ? into far more details.



To draw the analysis DFD:

1. Look at the system from the inside to the outside
2. Identify the activities
3. Locate the data flows
4. Show the relationships between activities
5. Find the internal inputs or outputs that exist within the system
6. Level complex processes in the DFD into simpler ones
7. Look for duplication of data flows or data stores (files)

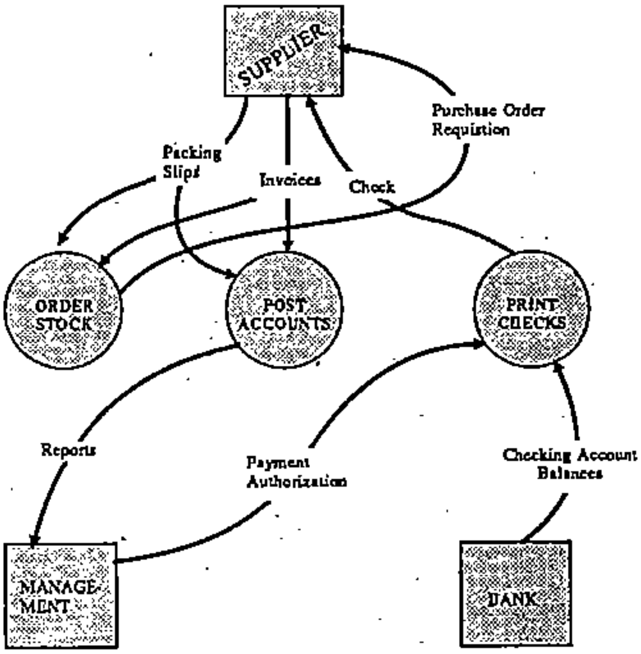



Figure 3.2

While determining the flow of data, the analyst collects samples of all relevant documents, such as sample cheques, invoices, packing slips, and other relevant forms. To create a record of all purchases from and payments to suppliers, a manual system requires that someone prepare a ledger entry for each supplier.

The assembled documents help an analyst understand what data the new system must collect and process. For example, the company can easily obtain the following data from the invoice itself:

1. Supplier name, address, and telephone number
2. Invoice number
3. Invoice date
4. Invoice
5. Terms of invoice
6. Amount of invoice

From the packing slip, it can obtain:

1. Supplier name

2 Shipping date

3 Date goods are received

4. Freight charges

5 Invoice number

Packing slips are carbon copies of invoices omitting certain data, such as the money value of the shipment. The warehouse clerk checks the merchandise received against the packing slip to be sure everything is in the carton and notes any discrepancies. Then the packing slip goes to accounting for comparison with invoices to be sure that the company received what it is paying for.

The ledger offers two categories of facts - supplier data and purchase/payment history:

1. Supplier name
2. Supplier address
3. Supplier telephone number
4. Date of transaction
5. Description of transaction
6. Amount of invoice or payment
7. Discount
8. Balance due the supplier

Each cheque sent to a supplier contains the following data:

1. Invoice number
2. Cheque number
3. Amount of payment
4. Payment date

In addition to these documents, it is useful to have copies of reports prepared by the accounting department.

### 3.6.5 New System

During fact finding, an analyst acts as a researcher, gathering facts, figures, and documents and coming to grips with the entire scope of the problem. Now he must decide what can be done, what it will cost, and the benefits expected to be derived from the new system.

The first step is to generate a list of alternative solutions to the existing accounts payable problem. Possible solutions range from doing nothing to installing a fully computerised AP system. In such a case there could be four alternatives:

i) Do nothing leaving the existing system alone

ii) Hire more staff, partially automate the system, but continue with essentially a manual system

iii) Purchase AP software from an outside software supplier

iv) Design, program, and install a customized AP system

When all necessary facts, figures, documents, data-flow diagrams, questionnaires and observations are complete, the analyst can write the final report. The format of the final report, called the feasibility study, parallels that of the preliminary report. It starts with a restatement of the problems and its importance, followed by a list of the study's objectives, a review of the analyst's findings, tallies of expected costs and savings, and the analyst's recommendations.

In a large organisation, the analyst may use a standardised form for the final report, in smaller organisations, the analyst simply chooses the most logical format. In any case, the analyst distributes the typed, photocopied report to the manager who will decide whether to adopt, modify, or reject the recommended solution.

After management has thoroughly considered the feasibility study, it calls a meeting to discuss the study and to choose a course of action. This meeting should take place a few days after the study's distribution and should be conducted by the manager of the computing services department or whoever requested the analysis. The analyst plays a major role and should be well prepared to answer questions and supply needed information. In fact, the analyst should rehearse the presentation in order to identify and improve upon weak areas.

If the analyst leads the meeting, he or she must exercise control. The following rules are helpful.

i) Never read the feasibility study aloud; instead, summarise it, while trying to lead the audience to support the study's recommendations.

ii) Use visual aids, such as chalkboards, flipcharts, slides, photographs, and overhead transparencies.

iii) If appropriate, demonstrate equipment or software to show how it will work.

Often one key individual must be convinced, and this person will influence the others to follow. If all goes well, the meeting will end with a decision to implement the analyst's recommendations.

After the meeting, management notifies all appropriate staff members of its decision. If management has decided to proceed to the design stage, the notification memo explains the plan briefly and establishes an overall schedule. Even if management decides to maintain or modify the current system, it should still issue a memo, or people will wonder why the company wasted time with a study that produced no results.

After a decision to proceed, analysis ends and design begins. The analyst will organise all the memoranda, questionnaires, interview documents and forms, data-flow diagrams, and reports from both the preliminary and detailed analysis into one file, which becomes the analysis documentation.

## 3.7 SUMMARY

The determination of whether or not a project is worth doing is known as feasibility. Once it has been determined that the project is worth doing i.e. the project is feasible, the analyst can go ahead and prepare the project specification which finalizes project requirements. The process by which we determine whether a project is feasible or not is called feasibility study. There are seven types of feasibility which are discussed in this unit. Different steps involved in the feasibility analysis are listed and an investigative study is made in this unit.

## 3.8 MODEL ANSWERS

**Check Your Progress 1**

1. Feasibility is the determination of whether or not a project is worth doing. The process followed in making this determination is called a feasibility study. Feasibility study determines if a project can and should be done.

2. The seven types of feasibility are:

   i) Technical feasibility

   ii) Operational feasibility

iii) Economic feasibility

iv) Social feasibility

v) Management feasibility

vi) Legal feasibility, and

vii) Time feasibility.

3. Technical feasibility is concerned with specifying equipment and software that will successfully satisfy the user requirement.

4. The main points to be considered in Operational feasibility are :

   i) What changes will be brought with the system?

   ii) What organizational structures are disturbed?

   iii) What new skills will be required?

   iv) Do the existing staff members have these skills?

If not, can they be trained in due course of time?

**Check Your Progress 2**

1. The four types of benefits are:

   i) Cost-savings benefit

   ii) Cost-avoidance benefit

   iii) Improved-service-level benefit

   iv) Improved-information benefit.

2. Costs and benefits can be classified as follows:

   i) Tangible or intangible costs and benefits

   ii) Direct or indirect costs and benefits

   iii) Fixed or variable costs and benefits

3. Costs-benefits analysis can be defined as the method by which we find and estimate the value of the gross benefits of a new system specification.

# UNIT 4 SYSTEM REQUIREMENT SPECIFICATIONS AND ANALYSIS

**Structure**



## 4.0 INTRODUCTION

This is the last unit of this block. In the previous unit we have discussed various types of feasibilities, and cost/benefit analysis. In this unit we present in some detail, DFD (Data Flow Diagram) and Data Dictionaries, their characteristics, various types and applications. Then we discuss HIPO (Hierarchy plus Input Process Output), and the two forms of its diagrams viz. VTOC and IPO. Decision tables and Decision trees are of wide applications in various fields besides computer science. We discuss here these two important techniques in details with several examples. Finally, we describe Warnier-Orr diagram and Nassi-Shneidermann charts which are important tools in systems analysis and design.

## 4.1 OBJECTIVES

After going through this unit, you should be able to :

- define DFD;
- define Data Dictionary, its standard symbols and rules;
- explain HIPO, its two types of diagrams;
- draw decision table;
- display decision trees;
- illustrate Warnier-Orr diagrams; and
- illustrate Nassi-Shneidermann charts.

## 4.2 DATA FLOW DIAGRAMS (DFD)

### 4.2.1 What is DFD?

Graphical description of a system's data and how the processes transform the data is known as Data Flow Diagram (or DFD).

Unlike detail flowcharts, DFDs do not supply detailed descriptions of modules but graphically describe a system's data and how the data interact with the system.

To construct data flow diagrams, we use :

(i) arrows,

(ii) circles,

(iii) open-ended boxes, and

(iv) squares

An arrow identifies data flow – data in motion. It is a pipeline through which information flows. Like the rectangle in flowcharts, circles stand for a process that converts data/into information. An open-ended box represents a data/store–data at rest, or a temporary repository of data. A square defines a source (originator) or destination of system data.

The following seven rules govern construction of data flow diagrams (DFD):

1. Arrows should not cross each other.
2. Squares, circles, and files must bear names.
3. Decomposed data flows must be balanced (all data flows on the decomposed diagram must reflect flows in the original diagram).
4. No two data flows, squares, or circles can have the same name.
5. Draw all data flows around the outside of the diagram.
6. Choose meaningful names for data flows, processes, and data stores. Use strong verbs followed by nouns.
7. Control information such as record counts, passwords, and validation requirements are not pertinent to a data-flow diagram.

If too many events seem to be occurring at a given point, an analyst can decompose a data conversion (circle). The new data conversions form a parent-child relationship with the original data conversion: the child circle in Figure 4.2 belongs to the parent in Figure 4.1.

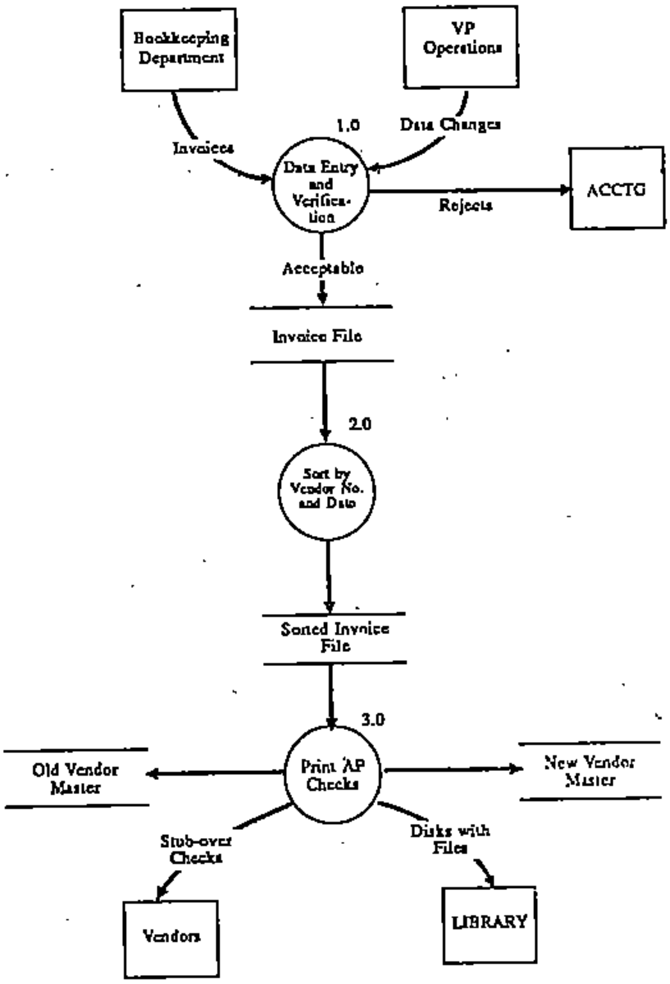



Figure 4.1

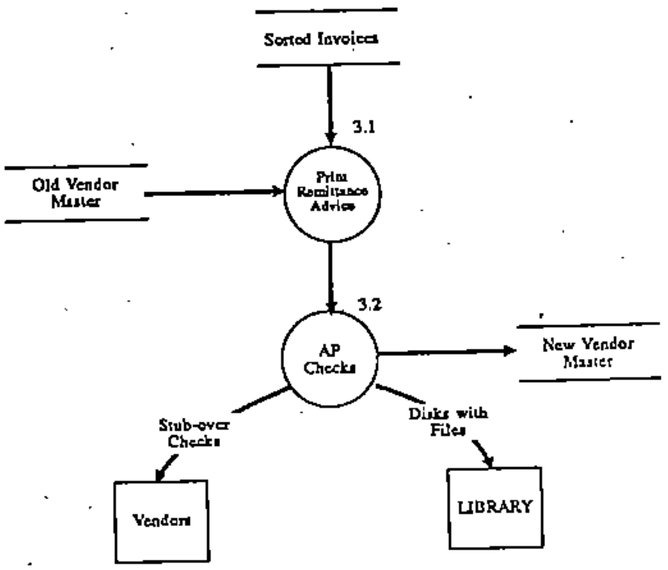



Figure 4.2

Devotees of data-flow diagrams insist that no other analyst's tool expresses so fully the flow of data. After all, don't computer people begin with data flow rather than the processing of the data? Another strong advantage is the balancing feature that builds in an error-detection system other tools lack. For example, if a parent data-flow diagram shows three inputs and two outputs, the leveled child diagrams taken together must have three inputs and two outputs. If there is an imbalance between parent and child data-flow diagrams, an error exists in either the parent or child diagram.

### 4.2.2 Charting tools used for DFDs

The data flow diagram(DFD) is the core specification in this method. Figure 4.3 shows the very few charting forms that are necessary.

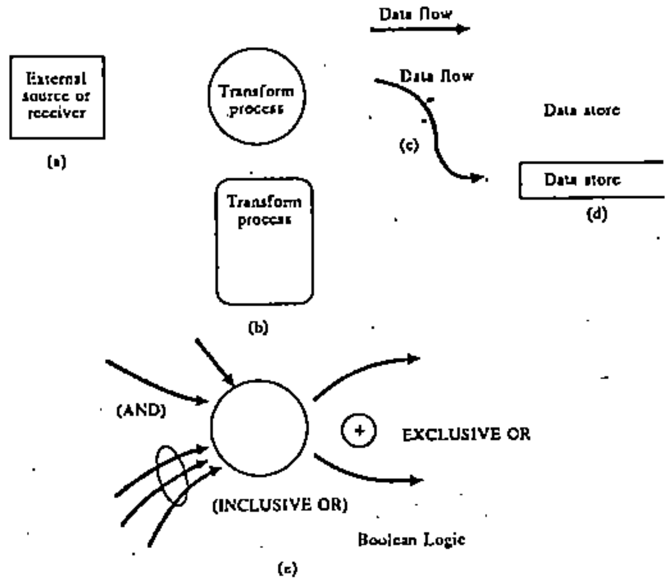



Fig 4.3 Data Flow Diagram Charting Forms

Shown in Figure 4.3(a) as a square box is the external source or receiver of data. Shown in Figure 4.3(b) is the **transform bubble**. Two variations of this have been put forth. The circle or real bubble is the better-known and is used by both Victor Weinberg and Tom DeMarco. The rectangular bubble shown beneath the circle is the form used by Chris Gane and Trish Sarson. The reason for the rectangular bubble is the perceived need to enter more information than can be contained in the bubble. Tom DeMarco, who is the purist in this group of four writers and educators in the structured method, holds that the data content of the bubble must be just the bare bones of one verb and a noun, since our objective is not to explain the process but to partition into leveled transforms. In fact the rectangular bubble is hard to draw freehand and the template containing this special form is not one you are likely to have around the shop, it must be specially ordered.

The line arrow is much more important in this method because it carries the data flow: the data into the transform and the data out of the transform. All lines must be identified by their data. Figure 4.3(c) shows the line arrows. The variations on the use of the line arrow are significant, because again the answer needs that different proponents of this method have perceived. Three different points of view are advanced by the practitioners mentioned above regarding line arrows.

1. When multiple lines go into or leave a transform, Weinberg offers the ability to use Boolean logic describing symbols to represent AND, INCLUSIVE OR, and EXCLUSIVE OR. This clearly indicates a felt need for decision logic above the base level. DeMarco advises against using Boolean decision logic. Our examples will not use Boolean logic beyond showing examples in Figure 4.3 (e), since this seems to the author to represent a consensus view of those who use the DFD approach.

2. DeMarco shows the line arrow as a curved line giving a different "feeling" than the straight lines and right angles of Weinberg and Gane-Sarson. In the DeMarco approach there is more

of a sense of flow - a sense of data in motion. Our examples will use the curved line.

3. Regarding the data flow along the line arrow, this would be a serious problem, like the problem of expressing the process within the confines of the bubble. The answer here is to take advantage of the fact that the method allows multiple lines in and out of a bubble and to break up the wordy data flow into several briefly named data flows. It is either that or lengthen the line. This concentration on detail of form, worrying about whether to use a circle or a square or a curved or straight line, may read as petty to the nonchartist. To those who mean to use this method to specify systems it is just as serious a matter as the concern of the professional tennis player with the type of racket to be used in a tournament. What we are trying to do with these forms is to invent solutions to problems as we move from the fluid to the concrete and from the tentative to the certain. We need a method for all seasons, but especially to communicate the fluid and the tentative new idea.

The final diagramming element shown in Figure 4.3 (d) is the open rectangle or two parallel lines, which indicates the data store (such as a database, file, Kardex or phone book). Gane and Sarson, unlike the others, show the data store as the two parallel lines joined at one side to make an open rectangle.

These are all the charting forms we need to use this methodology. Again, as in the flowcharting forms much can be developed out of a few tools. Essentially a system of any complexity whatever is shown with the bubble, line, data store rectangle, and external box.

It can be seen that although some practitioners prefer to use variation in their notation, the broad style is similar.

## 4.3 DATA DICTIONARIES

### 4.3.1 Why Data Dictionary?

A data dictionary defines each term (called a data element) encountered during the analysis and design of a new system. Data elements can describe files, data flows, or processes. For example, suppose you want to print the vendor's name and address at the bottom of a cheque. The data dictionary might define vendor's name and address as follows:

Vendor name and address = Vendor name +
Street +
City +
State +
Pin +
Phone +
Fax +
e : mail

This definition becomes a part of the data dictionary that ultimately will list all key terms used to describe various data flows and files.

### 4.3.2 Major Symbols

A data dictionary uses the following major symbols:

| | | |
|---|---|---|
| (i) | = | Equivalent to |
| (ii) | + | And |
| (iii) | [ ] | Either/or |
| (iv) | ( ) | Optional entry |

### 4.3.3 Four Rules

Four rules govern the construction of data dictionary entries:

1 Words should be defined to stand for what they mean and not the variable names by which they may be described in the program; use CLIENT_NAME not ABCPQ or

CODE06. Capitalization of words helps them to stand out and may be of assistance.

2 Each word must be unique; we cannot have two definitions of the same client name.

3. Aliases, or synonyms, are allowed when two or more entries show the same meaning; a vendor number may also be called a customer number. However, aliases should be used only when absolutely necessary.

4. Self-defining words should not be decomposed.
We can even decompose a dictionary definition. For instance, we might write

| | | |
|---|---|---|
| Vendor name | = | Company name,<br>Individual's name |

which we might further decompose to:

| | | |
|---|---|---|
| Company name | = | (Contact) +<br>Business name |
| Individual's name | = | Last name +<br>First name +<br>(Middle initial) |

After defining a term, say VENDOR NUMBER, we list any aliases or synonyms, describe the term verbally, specify its length and data type, and list the data stores where the term is found (figure 4.4). Some terms may have no aliases, may be found in many files, or may be limited to specific values. Some self-defining or obvious words and terms may not require inclusion in the data dictionary. For example, we all know what a PIN code and a middle initial are. Data dictionaries seldom include information such as the number of records in file, the frequency a process will run, or security factors such as passwords users must enter to gain access to sensitive data. Rather, data dictionaries offer definitions of words and terms relevant to a system, not statistical facts about the system.

Data dictionaries allow analysts to define precisely what they mean by a particular file, data flow, or process. Some commercial software packages, usually called Data Dictionary Systems (or DDS), help analysts maintain their dictionaries with the help of the computer. These systems keep track of each term, its definition, which systems or programs use the term, aliases, the number of times a particular term is used and the size of the term can be tied to commercial data managers.

| | |
|---|---|
| DATA ELEMENT NAME: | VENDOR_NUMBER |
| ALIASES: | None |
| DESCRIPTION: | Unique identifier for vendors in the accounts payable system. |
| FORMAT: | Alphanumeric, six characters. |
| DATA FLOWS: | Vendor master<br>Accounts payable open item<br>Accounts payable open adjustments<br>Cheque reconciliation |
| REPORTS: | Alphabetic vendor list<br>Numeric vendor list<br>A/P transaction register<br>Open item<br>Vendor account inquiry<br>Cash requirements<br>Pre-cheque-writing<br>Cheque register<br>Vendor analysis |

Fig. 4.4

### 4.3.4 Data Dictionary Types

Figure 4.5 illustrates the different types of data dictionaries and the functions of each addresses.

| Type / Function | Stand-alone<br>Global; manual or automated | Integrated with one database management system (DBMS) |
|---|---|---|
| PASSIVE<br>Documenting function only | Full organizaton documentation possible | |
| ACTIVE<br>Active in program preparation but not during execution | Full organization documentation possible plus:<br>Supports program and operations development with data structures (like program data definitions or even editing and validation code) | Full documentation possible<br>• Supports program and operations development with data structures<br>• Supports database definition language, database definition, and program specification blocks |
| IN-LINE<br>Also active during program execution<br>May have only limited documentation function | | Full documentation not possible in most cases<br>• Checks transaction and report system during job execution<br>• Can edit and validate input transactions at the dictionary level in-line rather than per application |

Figure 4.5 Data-Dictionary Types and Functions

There are two kinds of data dictionaries:

(i) integrated and

(ii) stand-alone.

The **integrated dictionary** is related to one database management system. To the extent the organisation data is under this DBMS it is global or organisationwide. However, very few enterprises have all their data eggs in one basket, so the dictionary documentation (metadata) can be considered as local and fragmented.

The **stand-alone dictionary** is not tied to any one DBMS, although it may have special advantages for one DBMS, such as the IBM DB-DC Data Dictionary, which has special features related to the IBM IMS DBMS but is still a stand-alone variety of dictionary.

#### Data Dictionary Functions

Both these types of dictionaries can be identified by functions as either **passive, active,** or **in-line.** Viewed either way, by type or function, the differences are striking. Passive, active, and in-line dictionaries differ functionally as follows:

#### Passive Data Dictionaries

The functionally passive dictionary performs documentation only. This variety of dictionary could be maintained as a manual rather than an automated database. For more than limited documentation use, the automated passive dictionary has clear advantages. From the organisational view the documentation function is the most important dictionary service with the most potential benefits, so the passive dictionary should not be thought of negatively. It has more limited functionality but may perform its critical function of global documentation best of all.

#### Active Data Dictionaries

Besides supporting documentation to one degree or another, the active data dictionary supports program and operations development by exporting database definitions and program data storage definitions for languages such as COBOL and Job Control Language (JCL) for execution-time performance. The IBM DB/DC Data Dictionary already mentioned is such a stand-alone, active data dictionary. A dictionary such as this is not an in-line data dictionary as delivered, which is not to say that it could not be put in-line by a determined effort of major proportions.

In-line Data Dictionaries

An in-line data dictionary is active during program execution, performing such feats as transaction validation and editing. Such a dictionary would always have some documentation value, but documentation across the organisation about the organisation functions and activities and all the organisation information data stores is not likely. In-line dictionaries are associated with DBMS products such as Cullinet Software Corporation's IDMS-R or Cincom System's TOTAL, to name just two.

### 4.3.5 The Make-up of Data Dictionaries: Data Dictionary Internals

The minimum data dictionary is shown in figure 4.6.

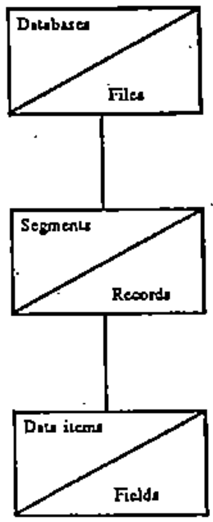



Fig. 4.6 Minimum Data Dictionary

We have a database system consisting of databases or files. These files consist of data groups or segments or records. These data groups consist of data items or fields. There is an implicit relationship here, which needs no additional comment. A certain amount of attribute information is always present. In the case of data items we need to know if it is a primary or secondary key or an attribute field, if it has aliases, what are the field type and field size, what is the name in various languages, and what is the user description of the item. We need to know whether the data item or data group is in test, system test, or production status. We need to know the number of occurrences of this data item on the dictionary.

Addressing these last points, a data item (for instance) on the IBM data dictionary may look strange to the uninitiated. It will look like this:

T,C,BALANCE-ON-HAND,0

We recognize balance-on hand as an inventory quantity. The T is the status code, which we will say is T because the data- item balance-on-hand is on the test-data database. The C is the subject code, which in this case is the primary programming language: COBOL. The 0 is the occurrence number where duplication exists in the common information system. So, in terms of this dictionary, the full description of the data-item consists of the four elements mentioned above. This convention holds for all subjects defined on the IBM data dictionary.

Before discussing the functions of the full-service extended data dictionary we need to review data-dictionary elements. Figure 4.7 shows these elements.

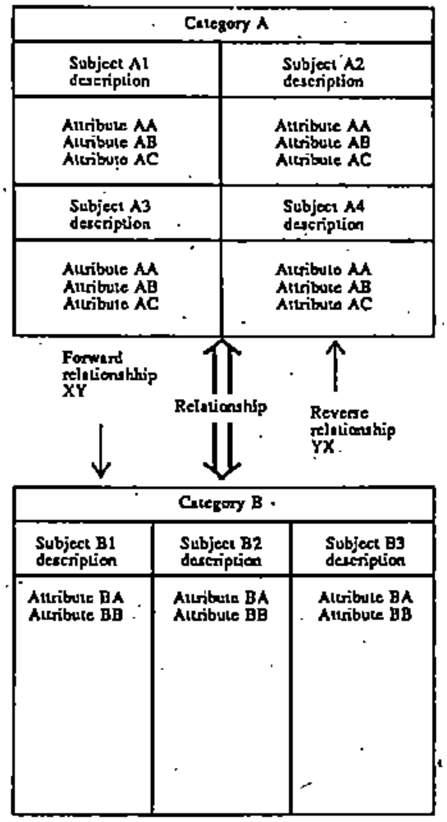



Fig 4.7 Data-Dictionary Elements

We have already referred to categories, subjects, relationships, attributes, and descriptions on other occasions. These are the elements that make up the data dictionary. In figure 4.7, Category A has a forward and reverse relationship to Category B. We have two-way relationships simply because we may want to examine these relationships in both directions. Data-Item is an example of a category. In a full- service dictionary some categories are predefined regarding attributes and relationships, but the dictionary has the capacity to handle user-defined categories. This, in IBM parlance, is an extended use of the dictionary. This "extensibility" feature is the heart of the full-service dictionary, allowing documentation of the whole organization and allowing us to use the dictionary as the software support for strategic and tactical planning.

Category A in figure 4.7 has four subjects. Each subject has the same attribute set as the others (attributes AA, AB, AC). For instance the category may be Projects. The four subjects are four different projects, described by name and description as unique. Perhaps the attributes are Project Leader, Project Due Date, and Percent Accomplished. All four subjects would have identical attribute names. Perhaps Category B is Information Systems, with subjects and attributes defined in a similar fashion. The forward relationship might be Projects ACCOMPLISH Information Systems. Reverse might be ACCOMPLISHED-BY.

Figure 4.8 shows another example of the elements that make up a data-dictionary database. In this case we have the category Business Function (or department) related to the Processes of the organisation such as Provide-Materials. Remember that the subject name looks like this: P,,Provide-Materials,0. Remember that the P is the status code, which in this case stands for Production. The two adjacent commas means the subject code is not used for this kind of category, and the zero is the occurrence (only this occurrence exists).

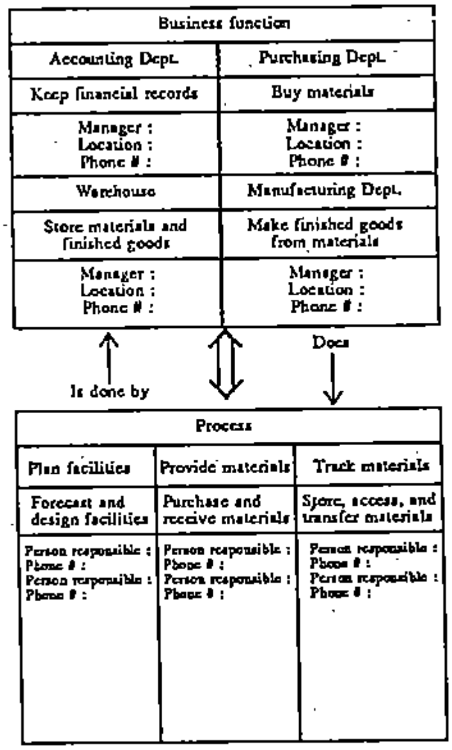



Fig. 4.8 Example of Data-Dictionary Elements

The IBM data dictionary, which is actually six linked databases, each with many segments, consists of standard categories and the infrastructure needed to "customize" installation categories. The standard categories have the attributes prebuilt and ready for the user to fill in. These standard categories are:

- DATA-BASE
- SEGMENT
- ELEMENT
- PROGRAM COMMUNICATION BLOCK
- IMS SYSTEM DEFINITION
- APPLICATION SYSTEM
- JOB
- PROGRAM
- MODULE
- TRANSACTION
- PSB

These categories are all related to servicing the data processing function and are not sufficiently broad in scope to support a dictionary for the entire organisation. The strategic plan cannot be documented with just these categories. It is the ability of this data dictionary to allow the creation of other user-defined categories that allows us to consider the dictionary a serious tool for systems analysis and documentation in support of the current and new system applications.

## 4.4 HIPO

HIPO stands for Hierarchy plus Input Process Output. It consists of two types of diagrams:

(i) Visual Table Of Contents (VTOC)

(ii) Input Process Output (IPO)

Together these diagrams assist in designing programs and their functions.

Following the structured approach that begins with generalities and descends to details, VTOC diagrams break a system or program down into increasingly detailed levels. Therefore, the name of the system appears at the top of the VTOC, the names of the major functions within the system lie on the second level and even smaller subfunctions lie on the third and succeeding levels.

When used to diagram a program, the VTOC arranges the program modules in order of priority, and it reads from the top down and from left to right. Each module of the program appears as a rectangle which contains a brief description of the module's purpose (two to four words, beginning with a verb followed by an object i.e. "compute net pay").

The VTOC for the correct assembly of a bicycle might include five major tasks:

(i) open the carton,
(ii) remove the parts (that is separate them),
(iii) group similar parts,
(iv) assemble the wheels
(v) finish assembling the bicycle (figure 4.9)

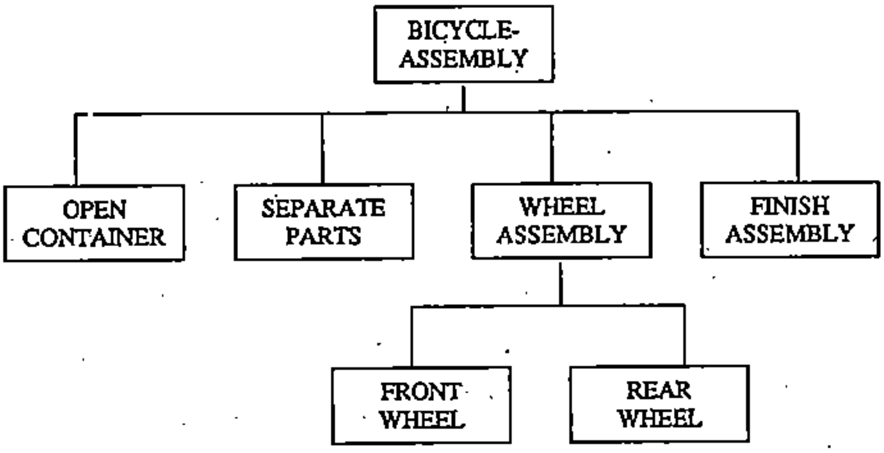



Figure. 4.9 VTOC for the modules to assemble a bicyle.

Compare this diagram with the one figure 4.10.

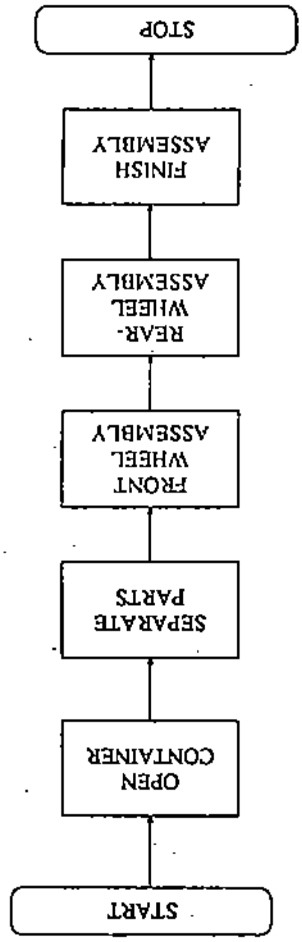



Fig. 4.10 : Flowchart of bycycle assembly

Both indicate hierarchy but the VTOC offers a more complete picture of the overall process. When assembling something, no matter how clear the instructions are, it helps to refer to a picture of the finished product. Within the VTOC, each task must be performed in the order specified, and each task may involve several subtasks. For example, wheel assembly involves the separate subtasks of front-and rear-wheel assembly.

An IPO chart defines the inputs, processing, and outputs for each module in the program. Figure 4.11 is an IPO for the "finish assembly" module, inputs for which include the frame, front-wheel assembly, rear-wheel assembly, seat, handlebars and chain. The processing requirements are bolting wheel assemblies to the frame, and attaching handlebars, chain, and seat. The output is the completed bicycle.

SYSTEM : Bicycle Assembly
MODULE : Finish Assembly

**Author : Jancy Manuel**
**Date : 12/02/94**

| INPUT | PROCESS | OUTPUT |
|---|---|---|
| 1. Frame | 1. Bolt front-and rear-wheel assembly to frame | 1. Completed bicycle |
| 2. Front-wheel assembly | 2. Attach handlebars | |
| 3. Rear-wheel assembly | 3. Attach chain | |
| 4. Seat | 4. Attach seat | |
| 5. Chain | | |
| 6. Handlebars | | |

Fig. 4.11 : Example of an IPO

### 4.4.1 Constructing a VTOC



Now let us learn how to prepare a VTOC for the computerisation of a manual accounts payable system. We first assign all the major outputs (reports) to modules as shown in figure 4.12.

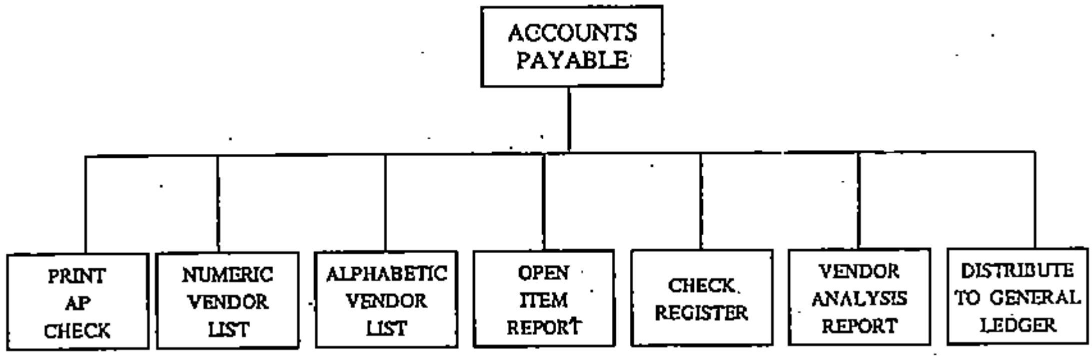



Fig. 4.12 VTOC for an accounts payable system begins with assignment of modules to each output report.

We name modules according to their function so the reader can tell exactly the purpose of the module (again using the verb-object format). After naming, we choose a number for the system (I), we give each module a sub or level number, beginning with 1 for the most general or highest level function. Therefore, we would identify the overall system as 1.0, the accounts payable check module as 1.1, the numeric vendor list as 1.2 and so on (Figure 4.13). Such a number system clarifies the relationships between modules, and allows anyone reading it easily to locate detailed IPO charts with corresponding numbers.

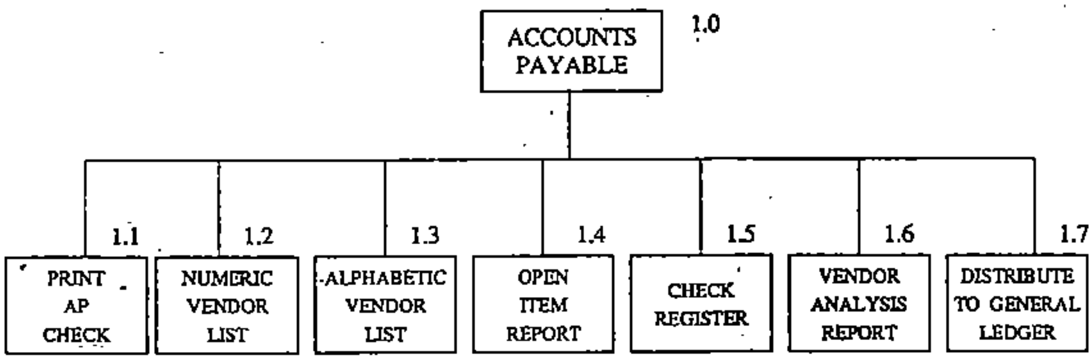



Fig. 4.13 The VTOC for the AP check-printing system with level number

After assigning level numbers to each module, we can consider whether further decomposition is necessary. Take the accounts payable check module, for instance. It must be decomposed to a lower level because it represents two tasks (remember, a module must be single purpose), one for each part or stub of the check (figure 4.14).

VENDOR : 00002 CHEQUE NO.________

| OUR INV. NO. | YOUR REF. NO. | INVOICE DATE | INVOICE AMOUNT | AMOUNT PAID | DISCOUNT TAKEN | NET CHEQUE AMOUNT |
|---|---|---|---|---|---|---|
| 001013 | | 25/[illegible]/94 | 150.00 | 150.00 | .00 | 150.00 |
| 001014 | HJ | 23/06/94 | 263.00 | 200.00 | .00 | 200.00 |
| | | | | | CHEQUE TOTAL | 350.00 |

CHEQUE NO______

| CHEQUE NO | CHEQUE DATE | VENDOR NO. |
|---|---|---|
| 000301 | 25/06/94 | 000002 |

________________19______

PAY ______________________________________________

______________________________________ या धारक को OR BEARER

रुपये RUPEES ______________________________________

______________________________ अदा करें | रु. Rs.

A/c No

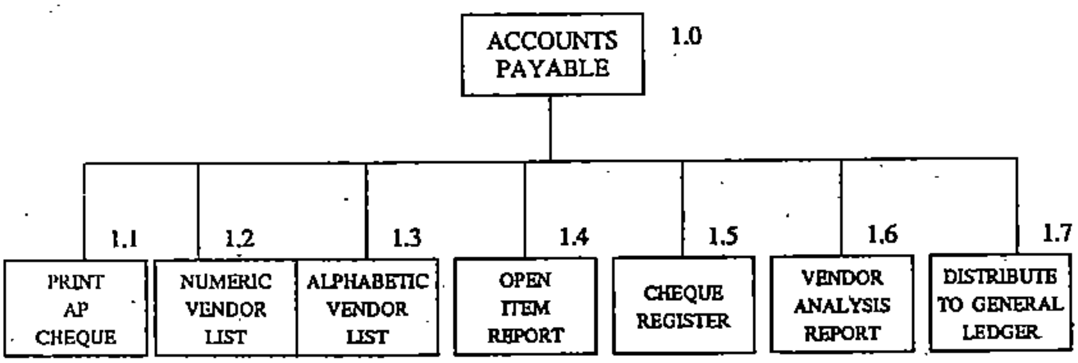
**STATE BANK OF INDIA**

10, SANSADMARG,POST BOX 5, NEW DELHI 110 001

Cheque No. 1201

"642150" 110038002 "

(a) Two-part cheque sent to vendors.




(b) This VTOC reveals the two component modules within the PRINT AP CHECK module.

Fig 4.14

The upper stub is the remittance advice, which contains the date, number, discount, balance a..d total of each invoice covered by the cheque. The lower stub is the cheque itself, complete with cheque number, payment amount, and vendor name and number. Applying the top-down concept to the cheque module, we can add another level of modules: one for printing the remittance advice and one for printing the cheque itself (figure 4.15). We assign this third level of modules a third set of numbers (1.1.1 and 1.1.2). Decomposition ends when all modules are single purpose, single entry, and single exit.

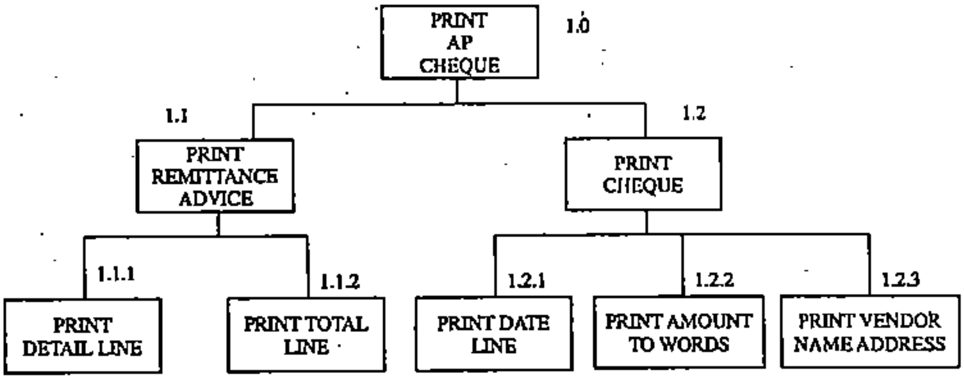



Fig. 4.15 Module development of the AP cheque system from the programmer's perspective

As far as the analyst is concerned, decomposition of the accounts payable cheque writing system can end at the third level of modules. However, the programmer who eventually receives the HIPO chart probably would decompose the modules to even lower levels, thus establishing a new series of numbers (figure 4.15). Programmers sometimes call their VTOCs structure charts, hierarchy charts, or tree charts. When finally programming the accounts payable check system, the programmer would begin at the top left, programming each level before moving down and to the right. Thus the programmer first would code "remittance advice" 1.1 followed by "print detail" (1.1.1) and then "print totals" (1.1.2). Having coded all of module 1.1, the programmer would tackle module 1.2, beginning with "print date line" (1.2.1), moving on to "print amount in words" (1.2.2), and finishing with "print vendor name address" (1.2.3).

Figure 4.15 shows three levels of modules, but complex systems may require many more. Regardless of their number, modules should receive unique and brief names that contains just enough detail for readers to understand their purposes.

### 4.4.2 Constructing an IPO

Let us add the second part of the HIPO diagramming system, the IPO chart. Whereas the VTOC diagram graphically shows an overview of the system, the IPO charts depict program logic, illustrating the steps required to produce desired outputs.

**SYSTEM :** Accounts Payable **Date :** 12/02/94 **Author :** Santosh
**MODULE :** 1.0 **Name :** AP Cheque
**DESCRIPTION :** Prints the stub-over-cheque sent to suppliers.

| INPUT | PROCESS | OUTPUT |
|---|---|---|
| 1. Vendor master file<br>2. Invoice File | 1. Read Invoice record<br>2. Match with vendor<br>3. Total amount<br>4. Print detail line<br>5. Print total line<br>6. Print date line<br>7. Print vendor name and address | 1. Print remittance advice on top stub<br>2. Print cheque on bottom stub |

Fig. 4.16 The IPO chart detail or program logic.

As figure 4.16 shows, the top of the IPO chart identifies the module with its number, title, a brief description, date, and the analyst's name. The chart itself is divided into three units: data input (the names of the files used), processing activities that will require programming, and output, which, in the case of our accounts payable system, would be the printed remittance advice (1.1) and the check (1.2). In the body of the chart, we use a narrative form to describe the input, process, and output as a list of activities. It simply lists activities, not necessarily ordering them in the sequence they should occur.

The HIPO system forms valuable system documentation and helps the analyst prepare reports as the system is being designed and developed. These charts offer several advantages. First, they can be drawn or modified rapidly. Second, they allow the analyst graphically to convey the system to non computer people. Third, standarised symbols enable some future analyst to grasp the system quickly. Finally, HIPO charts facilitate efficient schedules because they make it easy to estimate the time it will take to program a module, thus, simplifying programming assignments. The VTOC offers the analyst an alternative to the system flowcharts whereas the IPO replaces the program or detail flowchart.

## 4.5 DECISION TABLES AND DECISION TREES

Decision tables and trees were developed long before the widespread use of computers. They not only isolate many conditions and possible actions, but they help ensure that nothing has been overlooked.

### 4.5.1 Decision Tables

The decision table is a chart with four sections listing all the logical conditions and actions. In addition the top section permits space for title, date, author, system and comment.

The **condition stub** displays all the necessary tests or conditions. Like the diamond in a flowchart or the IF in pseudocode, these tests require yes or no answers. The condition stub always appears in the upper left-hand corner of the decision table, with each condition numbered to allow easy identification.

Thus **Condition stub** is a list of all the necessary tests in a decision table. In the lower left-hand corner of the decision table we find the action stub where one may note all the processes desired in a given module. Actions, like conditions, receive numbers for identification purposes. Thus **Action Stub** is a list of all the processes involved in a decision table.

The upper right corner provides space for the **condition entry - all** possible permutations of yes and no responses related to the condition stub. The yes or no possibilities are arranged as a vertical column called rules. Rules are numbered 1,2,3, and so on. We can determine the number of rules in a decision table by the formula:

Number of rules = 2 ^ N = $2^N$ where N represents the number of conditions and ^ means exponentiate. Thus a decision table with four conditions has 16 (2^4 = 2x2x2x2 = 16) rules one with six conditions has 64 rules and eight conditions yield 256 rules.

Thus **Condition entry** is a list of all the yes/no permutations in a decision table. The lower right corner holds the action entry. X's or dots indicate whether an action should occur as a consequence of the yes/no entries under condition entry. X's indication action; dots indicate no action.

Thus **Action entry** Indicates via dot or X whether something should happen in a decision table.

**Five sections of a decision table:**

| | |
|---|---|
| TITLE : Author : Comments : | Date : System : |
| Condition Stub | Condition Entry |
| Action Stub | Action Entry |

Fig. 4.17

Returning to the assembly of a bicycle, let us assume we must assemble a variety of containers full of parts. Since a bike can have either hand caliper or foot coaster brakes, the decision table must show the two conditions and five actions (figure 4.18). The two conditions necessitate four condition entries, and the five actions produce 20 possible action entries.

When we build the yes or no rules for the condition entry, we must construct all possible patterns of y's and n's. An arrangement that guarantees thoroughness is to place two y's is succession followed by two n's. In the second row, we place alternating pairs of y's and n's.

(a) **Decision table for bicycle assembly:**

| TITLE: Bicycle Assembly<br>Author :<br>Comments : More than one carton of parts needs to be assembed | DATE :<br>System : | | | |
|---|---|---|---|---|
| | 1 | 2 | 3 | 4 |
| 1. Last Carton ? | y | y | n | n |
| 2. Hand brakes ? | y | n | y | n |
| 1. Open container | . | . | x | x |
| 2. Stack parts | . | . | x | x |
| 3. Assemble whells | . | . | x | x |
| 4. Finish assembly | . | . | x | x |
| 5. End of assembly operations | x | x | . | . |

Fig. 4.18 : Decision table for bicycle assembly

A decision table with four conditions (2^4 = 16) would have 16 different sets of y's and n's and would result in the following pattern of yes and no responses.

The first row therefore will have eight y's followed by eight n's. The second row (corresponding to the second entry in the condition stub) has four y's, four n's, four y's and four n's.

**The complete four-condition entry would read:**

```
y y y y y y y y n n n n n n n
y y y y n n n n y y y y n n n n
y y n n y y n n y y n n y y n n
y n y n y n y n y n y n y n y n
```

This form ensures that the analyst includes all combinations with duplication.

If large number of conditions exist (four conditions result in 16 condition entries, six conditions in 64), decision tables can become unwieldly. To avoid lengthy decision tables, analysts must remove redundancies and yet still take precautions not to overlook anything. On occasion, two or more rules may be combined to reduce or eliminate redundancy. In figures 4.18 and 4.19, rules 1 and 2 cause the last action in the action stub to occur.

Therefore, these two rules could be combined to eliminate redundancy. To indicate redundancy, we put a dash (-) in the condition entry to show that this condition stub is irrelevant and can be ignored.

The decision table in figure 4.19 depicts the AP cheque module. Compare with figure 4.16 (IPO). Although this format is fairly typical, in practice you will encounter several different kinds of decision tables. Figure 4.19 called limited entry, because the condition entry contains yes or no responses for each rule.

**Limited Entry;** A type of decision table listing a y or n response for each condition.

**AP cheque decision table:**

TITLE : AP Cheque | DATE : Sept. 25, 1994
Author : | System : Accounts Payable System
Comments : Two files are to be read until the end of file.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. End of vendor master file ? | y | y | y | y | n | n | n | n |
| 2. End of sorted invoice file ? | y | y | n | n | y | y | n | n |
| 3. Do vendor numbers match ? | y | n | y | n | y | n | y | n |
| 1. Read a vendor master record | . | . | . | . | . | . | . | x |
| 2. Read an invoice record | . | . | . | . | . | . | x | . |
| 3. Add amount to total | . | . | . | . | . | . | x | . |
| 4. Print invoice detail line | . | . | . | . | . | . | x | . |
| 5. Print data line | . | . | . | . | . | . | . | x |
| 6. Print amount in words | . | . | . | . | . | . | . | x |
| 7. Print vendor name/address | . | . | . | . | . | . | . | x |
| 8. End of module | x | x | . | . | . | . | . | . |

Fig. 4.19: A limited entry decision table

**Extended Entry:** Type of decision table displaying values to be tested in the condition entry (Figure 4.20).

**AP cheque written as an extended-entry decision table:**

TITLE : AP Cheque | DATE : Sept. 25, 1994
Author : | System : Accounts Payable System
Comments : Two files are to be read until the end of file.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. Vendor master file ? | End | End | End | End | More | More | More | More |
| 2. Sorted invoice file ? | End | End | More | More | End | End | More | More |
| 3. Vendor numbers ? | End | More | End | More | End | More | End | More |
| 1. Read a vendor master record | . | . | . | . | . | . | . | x |
| 2. Read an invoice record | . | . | . | . | . | . | x | . |
| 3. Add amount to total | . | . | . | . | . | . | x | . |
| 4. Print invoice detail line | . | . | . | . | . | . | x | . |
| 5. Print data line | . | . | . | . | . | . | . | x |
| 6. Print amount in words | . | . | . | . | . | . | . | x |
| 7. Print vendor name/address | . | . | . | . | . | . | . | x |
| 8. End of module | x | x | . | . | . | . | . | . |

Fig. 4.20 : An extended-entry decision table

**Mixed Entry:** A type of decision table mixing values in the condition and action entries.

**AP cheque written as a mixed-entry decision table:** Shown below in Fig. 4.21

TITLE : AP Cheque
Author :
Comments : Two files are to be read until the end of file.

DATE : Sept. 25, 1994
System : Accounts Payable System

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. Is vendor master file at end? | y | y | y | y | n | n | n | n |
| 2. Sorted invoice file ? | End | End | More | More | End | End | More | More |
| 3. Vendor numbers ? | End | More | End | More | End | More | End | More |
| 1. Read a vendor master record | . | . | . | . | . | . | . | x |
| 2. Read an invoice record | . | . | . | . | . | . | x | . |
| 3. Add amount to total | . | . | . | . | . | . | x | . |
| 4. Print invoice detail line | . | . | . | . | . | . | x | . |
| 5. Print data line | . | . | . | . | . | . | . | x |
| 6. Print amount in words | . | . | . | . | . | . | . | x |
| 7. Print vendor name/address | . | . | . | . | . | . | . | x |
| 8. End of module | x | x | . | . | . | . | . | . |

Fig. 4.21 : A mixed-entry decision table

**Open ended** (1) A type of decision table that permits access to another decision table. (2) Questionnaire items that respondents must answer in their own words.

A mixed-entry decision table combines the values and yes or no (figure 4.21), while an open-ended one allows an action entry specifying an additional decision table(figure 4.22). An analyst may want to use one of these other types of decision tables to make the table more readable for a user or manger or to decompose a large (seven conditions leading to 128 rules) table into a series of smaller ones.

**AP cheque decision table (open-ended):**

TITLE : AP Cheque.
Author :
Comments : Two files are to be read until the end of file.

DATE : Sept. 25, 1994
System : Accounts Payable System

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. End of vendor master file ? | y | y | y | y | n | n | n | n |
| 2. End of sorted invoice file ? | y | y | n | n | y | y | n | n |
| 3. Do vendor numbers match? | y | n | y | n | y | n | y | n |
| 1. Read a vendor master record | . | . | . | . | . | . | . | x |
| 2. Read an invoice record | . | . | . | . | . | . | x | . |
| 3. Add amount to total | . | . | . | . | . | . | x | . |
| 4. Print invoice detail line | . | . | . | . | . | . | x | . |
| 5. Print date line | . | . | . | . | . | . | . | x |
| 6. Print amount in words | . | . | . | . | . | . | . | x |
| 7. Print vendor name/address | . | . | . | . | . | . | . | x |
| 8. End of module | x | x | . | . | . | . | . | . |
| 9. Go to next module | x | x | . | . | . | . | . | . |

Fig 4.22 : A open-ended decision table

Besides designing screen layout formats and determining screen specifications, the design must develop input controls for interactive dialogue and illustrate the way in which screens and menus are linked together. Three tools which help the design team in doing this are dialogue trees, decision trees, and picture-frame analysis. With dialogue and decision trees, the team is able to show the flow of control in processing, including the actions users can take to halt or stop an input procedure. With picture-frame analysis, the design team is able to provide a walk-through of how screens will appear once a design becomes operational.

### Constructing a Dialogue Tree

A dialogue tree maps the static and dynamic messages that take place between the computer and the user. Figure 4.23 shows the design of a tree for a simple file processing menu.

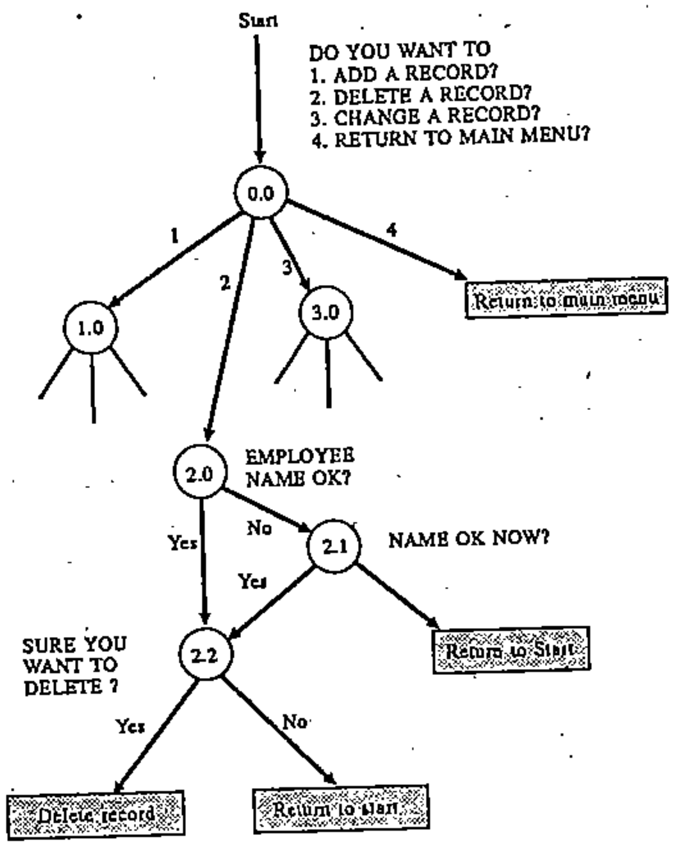



Fig. 4.23 Dialogue Tree showing branches from prompting menus

As shown, a dialogue tree has multiple branch points when menus are used, and forks at yes or no points. If we trace the steps shown in figure 4.23, the dialogue tree should lead you to conclude the following:

1. When an initial response of 2 is received, the program branches to a procedure to DELETE A RECORD from the employee master file.
2. Before a record is deleted, the user is asked to verify that the employee name is correct. The message reads: EMPLOYEE NAME OK?
3. If the name is correct, the tree forks and asks: SURE YOU WANT TO DELETE?
4. If the name is incorrect, the tree forks and asks: NAME OK NOW?
5. If the name is correct (is OK now) and the user responds yes to the question SURE YOU WANT TO DELETE, the record is removed from the file.
6. If the name is not correct, or if the name is correct but the user decides not to delete control is shown as "return to start" - namely, a loop back to the start of the tree.

Isn't this tree incomplete? If the employee name is not correct at node 2.0, how could it be correct at node 2.1? an expanded dialogue tree,like the one shown in figure 4.24, helps fill in the missing messages. The more detailed tree shown a node with an X. This is a non-restricted node, meaning that it is not restricted to a prescribed number of choices. The first nonrestricted node indicates that it is necessary to find an employee record before testing to determine whether the name is correct. Moreover, if a record is found but the name is incorrect, a second attempt (as noted by a second unrestricted node) is made to find the correct employee record. If this second search is successful, the user is aksed: NAME OK NOW?

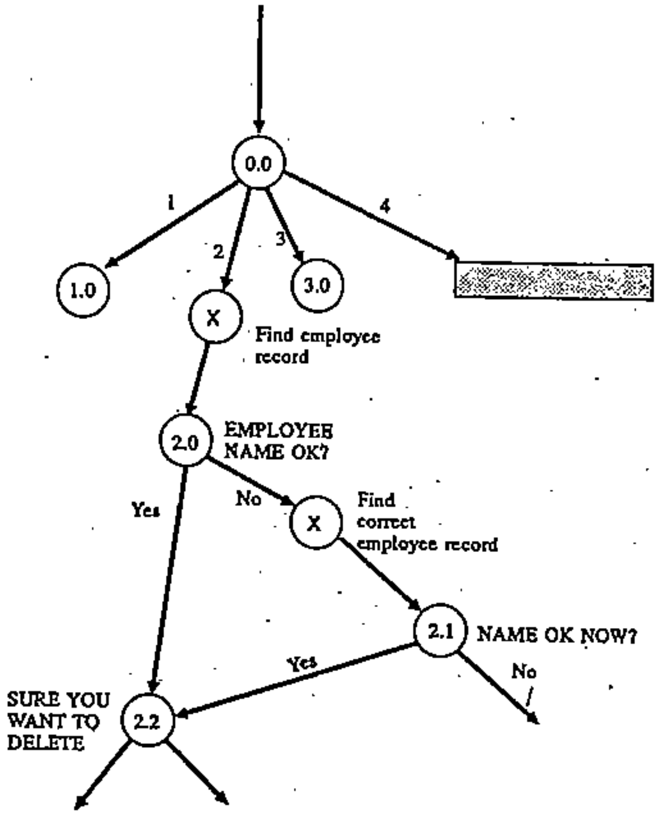



Fig. 4.24

### 4.5.2 Decision Trees

At times, a dialogue tree is too specific for design teams to work with. What they prefer is an easier-to-follow mapping of a complex design. This mapping should show branch points and forks, but not the details of the user dialogue. A decision tree helps to show the paths that are possible in a design following an action or decision by the user. Figure 4.25 illustrates this second type of tree. As indicated, if the user selects 1, followed by M and A, the algebra menu would be displayed.

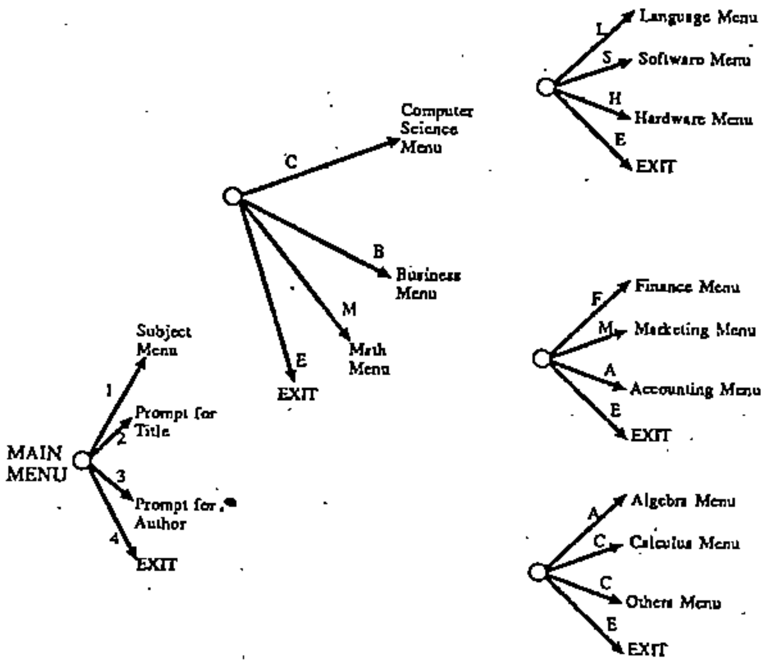



Fig. 4.25

What is the value of a tree such as this? It helps the designer visulaize how the user will move through the design to reach a desired location. Thus, a decision tree provides an overview of the flow of control to be built into computer programs.

Decision trees turn a decision table into a diagram (figure 4.26). This tool is read from left to right, decisions result in a fork, and all branches end with an outcome. Figure 4.26 shows the decision tree for printing the accounts payable check. Trees can be easily read by nontechnical users who find decision tables too complex. Users readily grasp branches, forks, and outcomes.

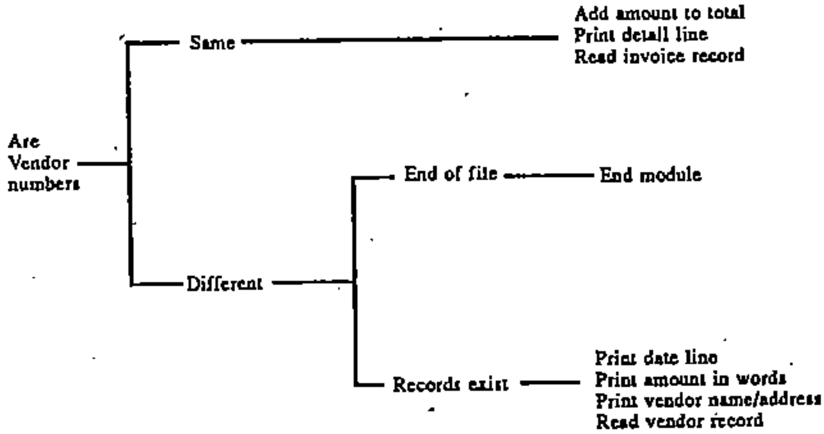



Fig. 4.26 Decision trees are graphic equivalents of decision tables.

## 4.6 WARNIER-ORR DIAGRAMS

Warnier-Orr diagrams are another tool aimed at producing working and correct programs. The Warnier-Orr diagram takes its name from its codevelopers, Jean-Dominique Warnier and Kenneth Orr. Unlike VTOCs, pseudocode, or flowcharts, which read from the top down and then from left to right, the Warnier-Orr diagram reads from left to right, then from the top to down. Whereas a flowchart requires many symbols, Warnier-Orr diagrams employ brackets, circles, parentheses, dots, and bars. Diagrams can depict data-dictionary-type definitions (Figure 4.27) or detailed program logic (Figure 4.28).

The Warnier-Orr uses brackets to group related elements following the sequence control structure. Technically the symbol for a bracket is "[" and a brace is "{" but Warnier-Orr calls "{" a bracket. Thus we see that three elements in Figure 4.27 make up the single element "Systems Process". Two elements, preliminary and detailed, make up Analysis.

Iteration structure (called repetition in the Warnier-Orr notation) is depicted by a parenthesis to the left of the series of elements to be repeated (Figure 4.28). A Number inside the parentheses indicates the amount of times the iteration should be performed. Thus we will repeat the four bracketed elements until there are no more bicycles to assemble.

(a) The systems process drawn in Warnier-Orr fashion.

$$\text{Systems process (1)} \left\{ \begin{array}{l} \text{Analysis} \left\{ \begin{array}{l} \text{Preliminary} \\ \text{Detailed} \end{array} \right. \\ \text{Design} \\ \text{Development} \end{array} \right.$$

Figure-4.27

**(b) Warnier-Orr diagram for bicycle assembly.**

Process (0, End)
- Open container
- Stack parts
- Assemble wheels
- Finish assembly

Figure-4.28

**(c) Warnier-Orr diagram for the accounts payable stub-over-cheque module. This diagram shows the three control structures of sequence, selection, and repetition**

Begin (1)
- Read invoice record
- Initialise totals to zero (1)
- Read vendor record

Process (0, End)
- Discount taken
    - Calculate amount of discount
    - Subtract from amount of invoice
    - Print invoice line on stub
- Discount not taken
    - Print invoice line on stub
- Add amount of invoice to total
- Read invoice record

End (1)
- Print date line on check
- Print amount of invoice in words
- Print vendor name and addres

Fig. 4.29

A plus sign enclosed in a circle indicates an alternation or selection structure. A bar separates the decision into true

(above the bar) and false (below the bar) (see Figure 4.29).

For our AP cheque-printing logic, the Warnier-Orr diagram (Figure 4.29) has brackets surrounding the repetitive operations and a decision point inside the process section to determine whether a discount will be taken. This diagram also has beginning and ending logic that our bicycle example does not require.

Warnier-Orr diagrams show the beginning, processing, and ending parts of the detailed logic quite explicitly. In keeping with the structured methodology, they have a single entry and a single exit, they support the three control structures, and compared with other tools, they employ few symbols. Disadvantages of the Warnier-Orr system include its left-to- right, top-down construction (the opposite of all other tools which are topdown, left to right) and its focus on processing versus data flow.

Introduced by Warnier and later modified by Orr, Warnier-Orr diagrams are thus used to decompose and partition the contents of a data store, much like DFDs are used to decompose the functions of a system.

Figure 4.30 illustrates how a Warnier-Orr diagram would be drawn for the employee master file. As before, the employee master file is defined as a set of employee records, thus leading to the equation:

Employee_master_file = 1{employee_records}n

However, the Warnier-Orr diagram tells us how to define an

employee record. We can write (as before):

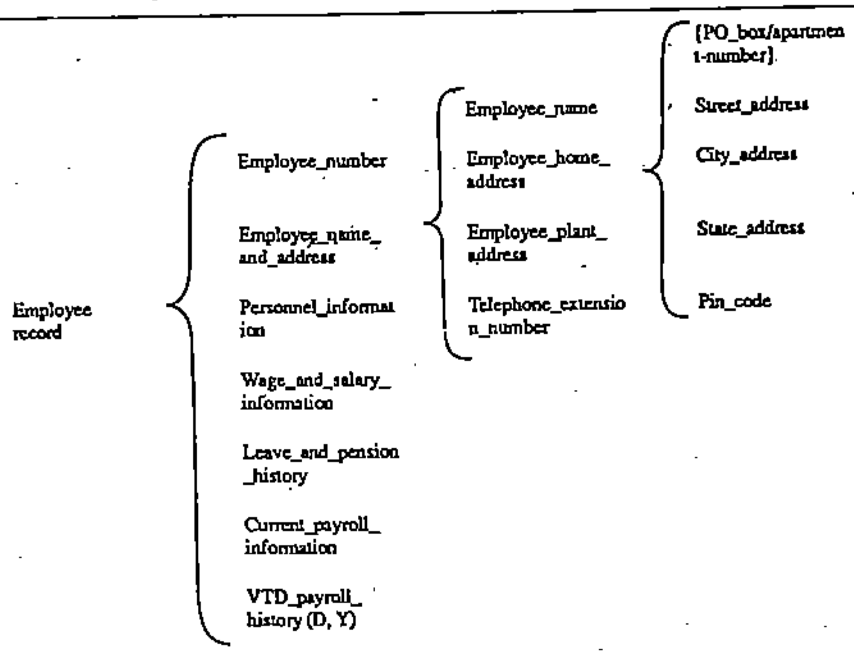
Employee_record = Employee number+ employee_name_and_address + personnel_information + wage_and_salary_information + leave_and_pension_history + current_payroll_history

Let us suppose next that we want to know which attributes make up the category employee name and address. The Warnier-Orr diagram informs us that

Employee_name_and_address = Employee_name + employee_home_address + employee_plant_address + telephone_extension_number

Fig. 4.30

Warnier-Orr Diagram Showing Decomposition of the Employee Record




If we wanted to know which attributes make up employee home address, further decomposition would be necessary. The Warnier- Orr diagram tells us that

Employee_home_address = ({PO_box_/apartment_number}) + street_address + city_address + state_address + Pin_code

where the post office box or the apartment number is optional.

A unique feature of Warnier-Orr diagrams is that they show sequences, either or conditions, and repetitions of a process. As illustrated by Figure 4.30, most Warnier-Orr diagram statements imply a sequential order. For example, Employee_ name_and_address is equal to the employee_name plus the employee_ home_address, plus the employee_plant_address, plus the telephone_extension_number. Finally, repetition of a process is indicated. Current_payroll_history is shown as a variable for each employee.

The Warnier_orr diagram continues to be pushed to the right until all right-hand attributes can be defined by their field length, or physical storage requirements.

City address might be defined as

City_address = 2{character}12

or as requiring from two to twelve characters. Telephone extension might be defined as

Telephone_extension = 4 digits,

where an extension always consists of a four-digit number.

As this example suggests, there are several good reasons for using Warnier-Orr diagrams to describe the contents of data stores. First Warnier-Orr diagrams are easy to construct, read, and interpret. Second, they permit larger, complex terms to be decomposed into constituent parts. Third, Warnier-Orr diagrams describe the three logical constructs used in programming. Fourth, they can be used to analyse both actions and things. Fifth, they simplify the definition and order of terms to be entered into the data dictionary. The main disadvantage of a Warnier_Orr diagram is that it does not show relationships which exist within and between data stores.

## 4.7 NASSI-SHNEIDERMANN CHARTS

Nassi-Shneidermann (N-S) charts offer an alternative to either pseudocode or program flowcharts. Named after their authors, N-S charts are much more compact than program flowcharts, include pseudocode-like statements, and feature sequence, decision, and repetition constructs. Figure 4.31 illustrates an N-S chart designed for processing payroll cheques.

**Nassi-Shneidermann Chart**

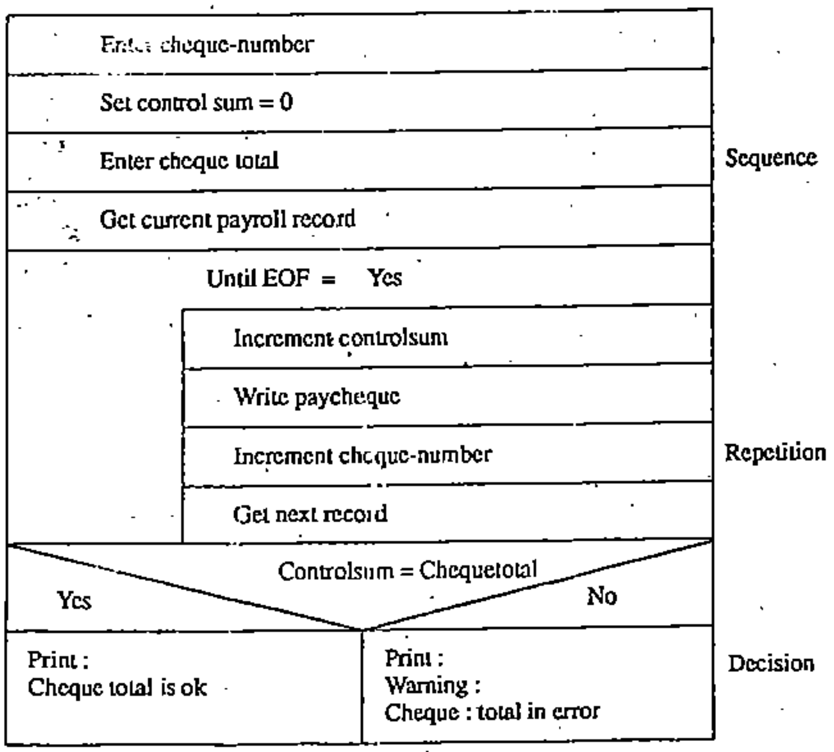



Fig. 4.31

Nassi-Shneidermann charts are also sometimes called the Chapin charts. This system was developed by and named tor I. Nassi and B. Shneidermann in the early 1970s and differs markedly from those we have examined thus far. It uses rectangles divided into halves with an angular line for selection, a horizontal rectangle for sequence, and the word DOWHILE for iteration. The bicycle-assembly chart appears in Figure 4.32, and the AP check-printing system with discounting in Figure 4.33.

Nassi-Shneidermann charts are winning wider acceptance in the computer industry because they so simply illustrate complex logic. Perhaps as analysts evaluate the various tools at their disposal, these charts will gain even more followers. As with other structured tools, they are single entry and single exit, and efficiently accommodate modules. However, they are not as useful for conveying system flow as they are for detailing logic development and they are difficult to maintain.

**Nassi-Shneidermann chart for bicycle assembly**

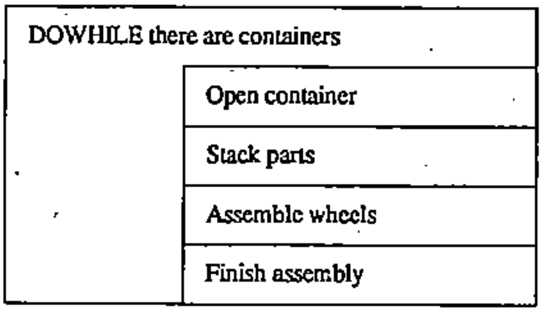



Fig. 4.32

**Nassi-Shneidermann chart for AP cheque-printing logic**

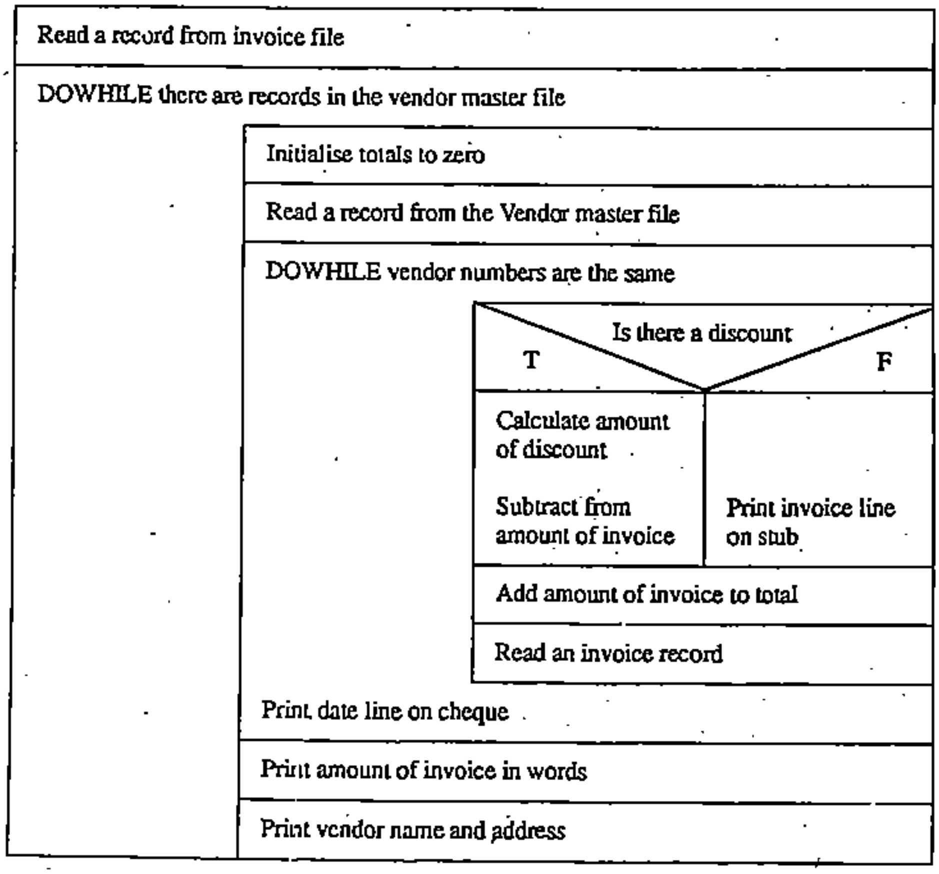



Fig. 4.33

## 4.8 SUMMARY

After the rigorous study on feasibility of a project, this study on system requirement specification is necessary. For this, you have studied in the unit DFD, Data Dictionary and their various characterisations. Next in section 4.4 you have studied HIPO with several examples. In section 4.5 you have studied two important techniques, Decision Tables and Decision Trees with some examples. Finally you have seen how to make Warnier-Orr diagrams and to construct Nassi-Shneidermann charts.

## SUGGESTED READING


1. Analysis & Design of Information Systems—James A Senn, Mc-Graw Hill Book Co. (1986)
2. Structured Analysis and System Specification—Tom De Marco, Prentice Hall (1979)
3. The practical Guide to Structured Systems Design—Page Jones Meilir, The Yourdon Press (1980)
4. Manging the Structured Techniques—Edward Yourdon, Yourdon, Yourdon Press (1979)
5. NCC—Guide to Structured Systems Analysis & Design Method.


## ACKNOWLEDGEMENT

In preparing this unit, considerable reliance has been placed on 'Perry Edwards' book "SYSTEM ANALYSIS , DESIGN, and DEVELOPMENT".

# Notes

Uttar Pradesh
Rajarshi Tandon Open University

# BCA-07

# Elements of Systems Analysis and Design

Block

# 2

## SYSTEM DESIGN

# BLOCK INTRODUCTION

This block deals with the issues and practices involved with the design of systems. The first unit explains the importance of the structured approach to systems design and the advantages of following a modular approach towards development. A system is reflected by its input, output and the processes. The second unit, therefore, addresses itself to the different aspects of input design and the checks and controls that can be applied to it so that it is meaningful and dependable. The next unit enumerates the output devices and the considerations in the design of output reports. The importance of forms and presentation graphics which is now a fairly common part of output design is explained. Before the input data can be processed to give the required output, the information must be held in certain organised forms. The fourth unit of this block talks of file and data base design and therefore reflects upon issues with a practical touch and flavour to which the students would otherwise also have had occasion to learn in some of the other courses which form part of this programme.

# UNIT 1 STRUCTURED SYSTEM DESIGN

## Structure



## 1.0 INTRODUCTION

The systems objectives outlined during the feasibility study serve as the basis from which the work of system design is initiated. Much of the activities involved at this stage is of technical nature requiring a certain degree of experience in designing systems, sound knowledge of computer related technology and thorough understanding of computers available in the market and the various facilities provided by the vendors. Nevertheless, a system cannot be designed in isolation without the active involvement of the user. The user has a vital role to play at this stage too. As we know that data collected during feasibility study will be utilised systematically during the system design. It should, however, be kept in mind that detailed study of the existing system is not necessarily over with the completion of the feasibility study. Depending on the plan of feasibility study, the level of detailed study will vary and the system design stage will also vary in the amount of investigation that still needs to be done. This investigation is generally an urgent activity during the system design as the designer needs to study minutes details in all aspects of the system. Sometimes, but rarely, this investigation may form a separate stage between Feasibility Study and Computer System Design. Designing a new system is a creative process which calls for logical as well as lateral thinking. The logical approach involves systematic moves towards the end-product keeping in mind the capabilities of the personnel and the equipment at each decision making step. Lateral thought implies encompassing of ideas beyond the usual functions and equipment. This is to ensure that no efforts are being made to fit previous solutions into new situations.

## 1.1 OBJECTIVES

At the conclusion of this unit, you should be able to :

- define the system design considerations
- list out various design methodologies
- know the brief outlines on structured design
- explain modularisation, computer system design process and system specifications
- understand the importance of prototype design

## 1.2 SYSTEM DESIGN CONSIDERATIONS

The system design process is not a step by step adherence of clear procedures and guidelines. Though, certain clear procedures and guidelines have emerged in recent days, but still much of design work depends on knowledge and experience of the designer.

When designer starts working on system design, he will face different type of problems. Many of these will be due to constraints imposed by the user or limitations of the hardware and software available in the market. Sometimes, it is difficult to enumerate the complexity of the problems and solutions thereof since the variety of likely problems is so great and no solutions are exactly similar. However, following considerations should be kept in mind during the system disigning phase :

### 1.2.1 Design Objectives

The primary objective of the design, of course, is to deliver the requirements as specified in the feasibility report. In general, the following design objectives should be kept in mind:

(a) Practicality: The system must be stable and can be operated by people with average intelligence.

(b) Efficiency: This involves accuracy, timeliness and comprehensiveness of the system output.

(c) Cost: It is desirable to aim for a system with a minimum cost subject to the condition that it must satisfy all the requirements.

(d) Flexibility: The system should be modifiable depending on the changing needs of the user. Such modifications should not entail extensive reconstructing or recreation of software. It should also be portable to different computer systems.

(e) Security: This is very important aspect of the design and should cover areas of hardware reliability, fall back procedures, physical security of data and provision for detection of fraud and abuse.

System design involves first logical design and then physical construction of the system. The logical design describes the structure and characteristics of features, like the outputs, inputs, files, databases and procedures. The physical construction, which follows the logical design, produces actual program software, files and a working system.

### 1.2.2 Constraints

The designer normally will work under following constraints:

| | | |
|---|---|---|
| Hardware | : | The existing hardware will obviously affect the system design. |
| Software | : | The available software (operating system, utilities, language etc.) in the market will constrain the design. |
| Budget | : | The budget allocated for the project will affect the scope and depth of design. |
| Time-scale | : | The new system may be required by a particular time (e.g. the start of a financial year). This may put a constraint on the designer to find the best design. |
| Interface with other systems | : | The new system may require some data from another computerised system or may provide data to another system in which case the files must be compatible in format and the system must operate with a certain processing cycle. |

### 1.2.3 Processing Techniques

The processing options available to the designers are:

— Batch processing

— Real-time processing

— On-line processing

— A combination of all the above

You are already aware of these techniques. It is quite interesting to note, however, that a combination of these is often found to be ideal in traditional data processing applications. This increases through-put of the system as also brings down the response time of on-line activities. In most of the business applications, 24-hour data is acceptable enough and hence it is possible to update voluminous data after office-hours in batch mode.

### 1.2.4 Operation

Typically, the flow of data through a system has been shown in Figure 1. Throughout the design process as described in the next section, the system designer must consider and specify the requirements of each of these operational areas.

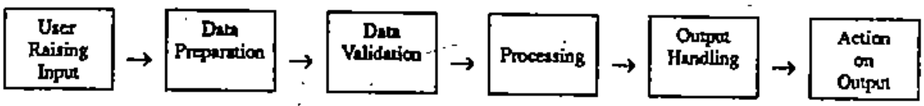



**Figure 1: Data Flow**

## 1.3 DESIGN METHODOLOGIES

The scope of the systems design is guided by the framework for the new system developed during analysis. More clearly defined logical method for developing system that meets user requirements has led to new techniques and methodologies that fundamentally attempt to do the following:

— improve productivity of analysts and programmers

— improve documentation and subsequent maintenance and enhancements.

— cut down drastically on cost overruns and delays

— improve communication among the user, analyst, designer, and programmer.

— standardize the approach to analysis and design

— simplify design by segmentation.

## 1.4 STRUCTURED DESIGN

Structured design is a data flow based methodology. The approach begins with a system specification that identifies inputs and outputs and describes the functional aspects of the system. The specifications, then are used as a basis for the graphic representation. The next step is the definition of the modules and their relationships to one another in a form called a structure chart, using a data dictionary and other structured tools.

Logical design proceeds from the top down. General features, such as reports and inputs are identified first. Then each is studied individually and in more detail. Hence, the structured design partitions a program into small, independent modules. They are arranged in a hierarchy that approximates a model of the business area and is organized in a top-down manner. Thus, structured design is an attempt to minimize the complexity and make a problem manageable by subdividing it into smaller segments which is called Modularisation or decomposition. In this way, structuring minimizes intuitive reasoning and promotes maintainable provovable systems.

A design is said to be top-down if it consists of a hierarchy of modules, with each module having a single entry and a single exit subroutine. The primary advantages of this design are as follows:

- Critical interfaces are tested first.
- Early versions of the design, though incomplete, are useful enough to resemble the real system.

- Structuring the design, perse, provides control and improves morale.
- The procedural characteristics define the order that determines processing.

### 1.4.1 Major System Design Activities

Several development activities are carried out during structured design. They are data base design, implementation planning, system test preparation, system interface specification, and user documentation.

(a) **Data base design:** This activity deals with the design of the physical database. A key is to determine how the access paths are to be implemented.

(b) **Program design:**In conjunction with database design is a decision on the programming language to be used and the flowcharting, coding, and debugging procedure prior to conversion. The operating system limits the programming languages that will run on the system.

(c) **System and program test preparation:** Each aspect of the system has a separate test requirement. System testing is done after all programming and testing are completed. The test cases cover every aspect of the proposed system, actual operations, user interface and so on. System and program test requirements become a part of design specifications - a pre requisite to implementation.

In contrast to the system testing is acceptance testing, which puts the system through a procedure design to convince the user that the proposed system will meet the stated requirements. Acceptance testing is technically similar to system testing but politically it is different. Acceptance testing is conducted in the presence of the user, audit representatives, or the entire staff.

### 1.4.2 System Interface Specification

This phase specifies for the user how information should enter and leave the system. The designer offers the user various options. By the end of the design, formats have to be agreed upon so that machine- machine and human-machine protocols are well defined prior to implementation. Before the system is ready for implementation, user documentation in the form of a operator's manual must be prepared. The manual provides instructions on how to install and operate the system, how to provide input, how to access, update, or retrieve information, how to the display or print output, in what format, and so on.

### 1.4.3 Audit Considerations

A well designed system should have controls to ensure proper operation and routine auditing. A proposed system's failure often results from a lack of emphasis on data control. When designing the system, standards of accuracy, consistency, and maintainability must be specified to eliminate errors and control for fraud. A system design introduces new control elements and changes the control procedures. In a manual system, internal control depends on human judgement, personal care, and division of labour. In a computer-based system, the number of persons involved is considerably reduced. A software packages is an effective substitute for human judgement in processing routines and error checks.

### 1.4.4 Audit Control and Documentation Control

An important function of system controls is to provide an audit trail. An audit trail is a routine designed to allow the analyst, user, or auditor to verify a process or an area in the new system. In a manual system, the audit trail includes journals, ledgers, and other documents that the auditor uses to trace transactions through the system. In a computerized system, record content and format frequently make it difficult to trace a transaction completely. The systems analyst must be familiar with basic auditing or work closely with an auditor to ensure an effective audit trail during the design phase. For auditing a system in a proper way, documentation is required. Documentation is the basis for the review of internal controls by internal or independent auditors. It also provides a reference for system maintenance. Analyst take lot of time in preparing the documentation.

Thus the main aim of auditing is to check that controls built into the design of proposed systems ensure its integrity. Audit considerations must be incorporated at an early stage in the system development so that changes can be made in time.

## 1.5 MODULARISATION

In structure design (already explained in section 1.4) a program is segmented into small, independent modules. These are arranged in a hierarchy that approximates a model of the business area and is organised in a top-down manner with the details shown at the bottom. Thus, in structured design, we try to minimise the complexity of the problem and make it manageable by sub-dividing it into smaller segments which is called modularisation or decomposition. This has been shown in Figure 2.

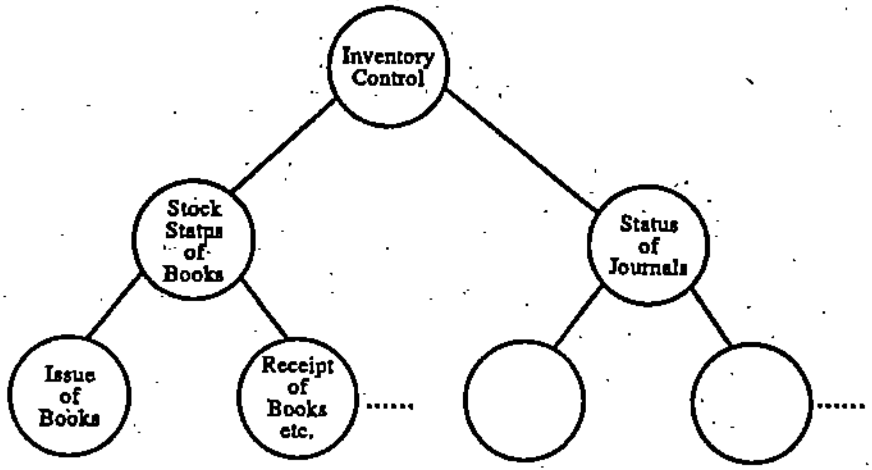



Figure 2: Decomposition - A Framework

**Check Your Progress 1**

1. Name the five objectives of system design.

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

2. List out some of the constraints under which a system designer has to work.

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

3. List out various processing techniques.

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

.....................................................................................................................

4. What do you understand by structured design?

..........................................................................................

..........................................................................................

..........................................................................................

..........................................................................................

So structured design arises from the hierarchical view of the application. The top level as shown in the Figure 2 shows the most important division of work; the lowest level at the bottom shows the details.

## 1.6 DESIGN PROCESS

The computer system design process is an exercise of specifying "how" the system will work. It is an iterative process which is based on "what" the system will do as shown in the feasibility report.

Mainly, following five parts have been included in the system design process:

(i) **Output design:** The starting point of the design process is the proper knowledge of system requirements which will normally be converted in terms of output.

(ii) **Input design:** Once the output requirements have been finalised, the next step is to find out what data need to be made available to the system to produce the desired outputs. The basic documents in which these data are available need to be identified. If necessary, these documents may have to be revised or new documents may have to be introduced.

(iii) **File design:** Once the input data is captured in the system, these may have to be preserved either for a short or long period. These data will generally be stored in files in a logical manner. The designer will have to devise the techniques of storing and retrieving data from these files.

(iv) **Procedure design:** This step involves specifications of how processing will be performed. In this, there are two aspects:

- Computer procedure
- Non-computer procedure

The computer procedure will specify what functions will be carried out on computer, what will be different programs and in what sequence the programs will be run. The non- computer procedure will specify the manual procedures for feeding input data, receiving outputs etc.

(v) **Control design:** The control design indicates necessary procedures which will ensure correctness of processing, accuracy of data, timely output etc. This will ensure that the system is functioning as per plan.

Generally, these steps as mentioned above are inter-dependent and some of them may have to be used together and traversed many times until a satisfactory design is prepared. It is just like the situation of "Two-step forward - one step backward" kind. In the Figure 3 we have tried to present an ideal situation about the progress of a project. But this situation occurs rarely in day-to-day life. Most of the time progress of a project takes different shape which is shown in Figure 4.

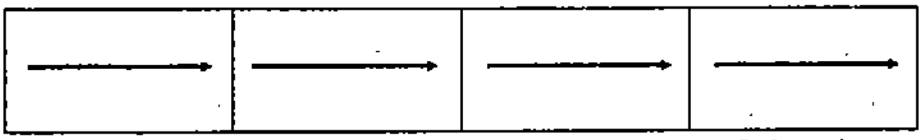



Figure 3: "Ideal" Project Progress

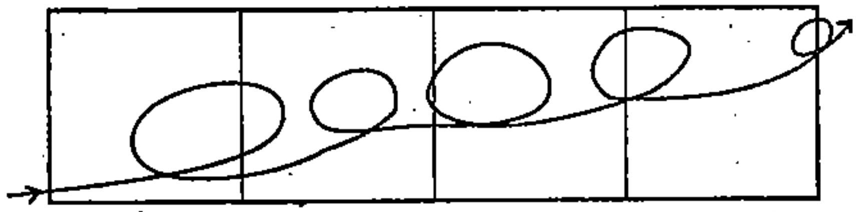



Figure 4: "Reality" of Project Progress

The system design process is, therefore, an iterative process where decisions made or changed at one step will have a "ripple effect" on other steps. For example, if an output report is modified, it may necessitate changes in input design, file design and control design.

In designing a system, if one tries to design a perfect system, it is his wrong conception. He may perhaps land up with no system at all. What is to be aimed at is the most satisfactory and operable design of a system and then gradual improvements over time.

## 1.7 SYSTEM SPECIFICATIONS

The result of the system design process is a document known as "system specifications". The complete details of design about the proposed system have been included in it. It serves as a blue print and helps in developing and implementing the new system. This also forms the primary documentation on which the system maintaining persons will fall back upon after the system is in use. Later on, this document is normally divided into different parts for easy reference. Thus we get system manual, user manual, operational manual. Since the system specifications are prepared as a plan, it becomes sometimes necessary to modify it after taking into consideration the practical difficulties or bottlenecks or errors found during later stages of development. System specifications should include all the details necessary to implement the system and to understand the whole working of the system.

## 1.8 PROTOTYPE DESIGN

Prototype is a working system that is developed to test ideas and assumptions about the new system. Like any computer-based system, it consists of working software that accepts input, performs some operations on it and gives the output. It is the first version of an information system - an original model.

The prototype is actually a pilot or test model. It is designed to be easily changed. Information gained through its use is applied to a modified design that may again be used as a prototype to reveal still better design information. The process is repeated as many times as necessary to reveal essential design requirements. In general, prototypes are considered to be most useful under the following conditions:

- No system with the characteristics of the one proposed has yet been constructed by the developers.
- The essential features of the system are only partially known; others are not identifiable even through careful analysis of requirements.
- Experience in using the system will significantly add to the list of requirements the system should meet.

- Alternate versions of the system will evolve through experience and additional development and refinement of its features.
- The system users will participate in the development process.

The underlying principle of prototyping is as under:

- Users can point out features they like or dislike and so indicate short-comings in an existing and working system more easily than they can describe them in a theoretical or proposed system. Experience and use produce more meaningful comment than analysis of charts and narrative proposals.
- Systems prototyping is an interactive process. It may begin with only a few functions and be expanded to include others that are identified later. It may also start with what both analyst and user believe is a complete set of functions that may expand or contract through use and experience.

Typically, these are the steps in the prototyping process:

- Identify the user's known information requirements anu reatures needed in the system.
- Develop a working prototype.
- Use the prototype, noting needed enhancements and changes. These expand the list of known system requirements.
- Revise the prototype based on information gained through user experience.
- Repeat these steps as needed to achieve a satisfactory system.

As these steps suggest, prototyping is not a trial-and-error development process. Before starting the system design work, user and system analyst sit together and discuss to identify the requirements. These discussions form the basis for the construction of the prototype. System analyst is fully responsible for the development of the working prototype.

Check Your Progress 2

1. List out the various parts of system design process.

..........................................................................................

..........................................................................................

..........................................................................................

..........................................................................................

2. Define "system specifications" briefly.

..........................................................................................

..........................................................................................

..........................................................................................

..........................................................................................

3. What do you know about "Prototype Design"?

..........................................................................................

..........................................................................................

..........................................................................................

..........................................................................................

## 1.9 SUMMARY

The role of system analyst is considered to be very signiticant in designing oı system.

During this stage, he applies his understanding of the procedures and converts them into an efficient system. The design is a solution-translation of requirements into ways of meeting them. The design phase focuses on the detailed implementation of the system recommended in the feasibility study. Emphasis is on translating performance specifications into design specifications. The design phase is a transition from a user-oriented document to a document oriented to the programmers or data base personnel. System design is not an exact science like programming where a set of instructions would lead to a desired result. As the subject involves lot of interaction with the users, it has to be flexible and dynamic to meet the changing needs of the users over a period of time.

## 1.10 MODEL ANSWERS

**Check Your Progress 1**

1. Five objectives are:

   (i) Practicability (ii) Efficiency

   (ii) Cost (iv) Flexibility

   (v) Security

2. Some of the constraints are:

   (a) Hardware (b) Software (c) Budget

   (d) Time-scale (c) Interface with other systems

3. Various processing techniques are:

   (i) Batch Processing

   (ii) Real-time Processing

   (iii) On-line Processing

   (iv) A combination of all the above

4. Structured design is a data flow based methodology. This approach begins with a system specification that identifies inputs and outputs and describes the functional aspects of the system. The next step is the definition of the modules and their relationships to one another in a form called a structured chart, using a data dictionary and other structured tools.

**Check Your Progress 2**

1. Five parts of system design process are:

   (i) Output design

   (ii) Input design

   (iii) File design

   (iv) Procedure design

   (v) Control design

2. The result of system design process is a document known as system specification. The complete details of design about the proposed system have been included in it. It serves as a blue print and helps in developing and implementing the new system.

3. Prototype is a working system that is developed to test ideas and assumptions about the new system. It is actually a pilot or test model. It is designed to be easily changed.

# UNIT 2 INPUT DESIGN AND CONTROL

## Structure



## 2.0 INTRODUCTION

Once the analysis and design of the system has been done, it would be necessary to identify the data that are required to be processed to produce the outputs. Input is one of the most expensive phases of the operation of a computerised system and creates sometimes a major problem. Different type of problems with a system can usually be traced back to faulty input design method. Needless to say, therefore, that the input data are the lifeblood of a system and have to be analysed and designed with utmost care and consideration. Input design features can ensure the reliability of the system and generate correct reports from the accurate data. The input design also determines whether the user can interact efficiently with the system.

## 2.1 OBJECTIVES

After studying this unit, you should be able to understand the following:

Importance of input design for producing the correct report.

- Criteria for selecting the most appropriate input method and medium for an application.
- Development of a system for validating input to a computer application.
- Preparing layout for terminal screens.

## 2.2 PROCESSING TRANSACTION DATA

**Transaction System:**

The transaction processing system keeps records of routine business activities, follows standard operating procedures, and does not require complex decision making. These systems also depend on accurate and detailed data and must be able to process large volumes of data in short periods of time. Examples of transaction system are accounting, inventory, sales-order entry, banking and airline reservation systems. All of these applications follow a transaction - processing cycle, which begins by entering the data, continues by processing the data, and concludes by presenting the output. Data can be entered into a computer system in many ways. At a bank, customers can enter their transaction through an automatic teller terminal. At a super market, prices may be read by an optical scanning device. At a travel agency, airline reservation are entered through a terminal. Regardless of the method used, the data must be entered accurately as well as efficiently. Once the data have been entered into a transaction system, it must be processed. Mainly there are two ways of processing the data. One is batch processing and other method in on-line processing.

### 2.2.1 Batch Processing

In the batch processing the data are collected for a given period of time, and the resulting "batch" of data is processed as a single job. Batch processing is useful when most of the records in a large database must be processed at the same time.

Many organizations also use batch processing methods for billing. Telephone, gas, electric, and cable TV companies, for example, prepare their bills using batch processing. Even Payroll—whether processed weekly, or monthly—is almost handled through batch processing.

### 2.2.2 On-Line Processing

Although batch processing is still used for some applications, today an increasing number of companies choose on-line transaction-processing systems. On-line systems can reduce data-processing costs, offer better customers service, and provide a strategic advantage over competitors.

In on-line systems, data are processed instantly by the CPU. As a result, whenever a user wishes to enter or access data, the request is accommodated within a few seconds. Unlike batch processing, each request is processed individually - there is no waiting while groups of requests are batched and processed together. On-line systems are thus, preferred when selected records must be processed at any single point of time or when the user and computer system must interact.

Because on-line transaction-processing systems can access data instantly and because many transaction systems must accommodate large volumes of data, larger systems generally include several disk drives with capacities of several billion bytes of data. Tapes and optical storage, if used at all, serve as a backup medium, storing additional copies of the data in the event that those stored on disk are lost or damaged.

But an OLTP (ON-LINE TRANSACTION PROCESSING) system that brings data within instantaneous reach can also, if it breaks down, brings many of an organization's activities to a screening halt. Imagine the consequences of the failure in a bank deposit system. Not only would it be difficult, if not impossible, to continue serving customers,but the cost to the bank for even a few hours of service interruption could be very high. Because many organizations have become so dependent on OLTP systems and because failures can be dramatically disruptive, some firms have chosen to purchase fault-tolerant computer systems. These systems use additional hardware and software to help avoid a system failure. One key ingredient is the use of disk mirroring, a scheme in which the system maintains a mirror image of critical disk data on two physical separated disk drives. Whenever data are entered into the system, they are automatically entered into both devices, in the event of a disk failure, the system will automatically retrieve the data from the working disk. In addition, fault-tolerant computer systems can through hardware and software, circumvent portions of the hardware if the electronic circuitry breaks down. Two other important characteristics of an OLTP are its multi-user and multi-tasking capabilities. A multi-user, OLTP serves many users, each executing a different JOB, at what appears to be the same time. For example, it allows several people to sit at terminal scattered throughout an organization and work on problems that may be totally unrelated to one another. One user may be entering accounts receivables transactions, another may be entering accounts payables; and a third may be producing an inventory report. Each user has the impression that no one else is using the system, thanks to complex on-line operating system software that keeps the separate uses of the system disentangled.

## 2.3 ELEMENTS OF INPUT DATA

Inaccurate input data are the most common cause of errors in data processing. Errors entered by data entry operators can be controlled by input design. Input data are collected and organized into groups of similar data. Once identified, appropriate input media are selected for processing.

### 2.3.1 Input Data

The goal of designing input data is to make data entry as easy, logical and error free from errors as possible. In entering data, operators need to know the following:

- The allocated space for each field.
- Field sequence, which must match that in the source document.
- The format in which data fields are entered; for example, filling out the date field is required through the edited format mm/dd/yy.

When we approach input data design, we design the source documents that capture the data and then select the media used to enter them into the computer. Let us elaborate on each step.

### 2.3.2 Source Documents

Source data are captured initially on original paper or a source document. For example, a cheque written against an account is a source document. When it reaches the bank, it is encoded with special magnetic ink character recognition (MICR) so that it can be processed by a reader that is part of the information system of the bank. Therefore, source documents initiate a processing cycle as soon as they are entered into the system. Source documents may be entered into the system from punch cards, from diskettes, or even directly through the keyboard. A source document may or may not be retained in the proposed system. Thus, each source document may be evaluated in terms of:

— its continued use in the proposed system;

— the extent of modification for the proposed system &

— replacement by an alternative source document.

A source document should be logical and easy to understand. Each area in the form should be clearly identified and should specify for the user what to write and where to write it. For example, a field as simple as date of birth may be written in four different ways:

— 5 December, 1994

— Dec. 5 1994

— 12/5/94

— 5/12/94 (European style)

Unless it is clear in a source document that two digits are allowed for the month, day, and year (MM/DD/YY), we could expect such combinations of responses.

## 2.4 INPUT MEDIA AND DEVICES

Source data are input into the system in a variety of ways. The following media and devices are suitable for operation:

a) Punch cards are either 80 or 96 columns wide. Data are arranged in a sequential and logical order. Operators use a keypunch to copy data from source documents onto cards. This means that the source document and card design must be considered simultaneously.

b) Key-to-diskette is modeled after the keypunch process. A diskette replaces the card and stores upto 325000 characters of data - equivalent to the data stored in 4050 punch cards. Like cards, data on diskettes are stored in sequence and in batches. The approach to source document and diskette design is similar to that of the punch card. Data must be in sequence and logically cohesive.

c) MICR translates the special fonts printed in magnetic ink on checks into direct computer input.

d) Mark-sensing readers automatically convert pencil marks in predetermined locations on a card to punched holes on the same card.

e) Optical character recognition (OCR) readers are similar to MICR readers, except that they recognize pencil, ink or characters by their configuration (shape) rather than their magnetic pattern. They are often used in remote locations as free-standing input preparation devices or direct input media to the system.

f) Optical bar code readers detect combination of marks that represent data. The most widely known system is the Universal Product Code (UPC), which codes retail items in stores. Automatic tag reading is a major breakthrough in speeding up customer service and eliminating costly data input errors at the point of sale. It is virtually impossible for the sale clerk to enter incorrect merchandise information such as department and class type data. Automatic tag reading is the ideal way to collect unit inventory information fast, accurately and economically.

g) Cathode-ray tube (CRT) screens are used for on-line data entry. CRT screen generally display 80 characters simultaneously on a television-like screen. They show as many as 24 lines of data.

In addition to determining record media, the analyst must decide on the method of input and the speed of capturing and entering the data into the system. Processing may be batched (a group of records handled as a unit), on-line (records processed directly), sequential (sorted records), or random (unsorted). For example, magnetic tape may be suitable for batch sequential processing, whereas diskettes are ideal for on line processing and random inquiries.

## 2.5 INPUT DESIGN GUIDELINES

The design of input play very significant role in getting the correct output. It covers all phases of input from creation of initial data (original recording) to actual entering the data to the system for processing. The input design is the link that ties the information system into the world of its users. Some features of design may vary depending on whether the system is batch-oriented or on-line. Here, we will discuss the various objectives of input design. They focus on:

— Controlling amount of input

— Avoiding delay

— Avoiding errors in data

— Avoiding extra steps

— Keeping the process simple

Each of the five objectives of input design is briefly discussed below:

### 2.5.1 Controlling Amount of Data

An effective design controls the quantity of data for input for the following reasons:

Firstly, data preparation and data entry operations depend on people. Since labor costs are high, the cost or preparing and entering data is also high. It is quite evident, then, that reducing data requirements mean lowering costs through reduced labor expense. Secondly, the input phase of computing can be a slow process and take many times longer than that needed by computers to carry out their tasks. In fact, the computer itself may sit idle until data is prepared and input for processing. By reducing input requirements, the analyst will speed the entire process from data capture to processing to provide results to users.

### 2.5.2 Avoiding Delay

When processing is delayed owing to data preparation or data entry, the cause is called a bottleneck. Avoid bottlenecks when designing input should always be one of the objective of the analyst.

### 2.5.3 Avoiding Errors in Data

The third objective deals with errors. In one sense, the rate at which errors occur is dependent on the quantity of data. Since the lower the amount of data is inputted, there are fewer opportunities for the error to occur.

Firstly, the analyst can reduce this number by reducing the volume of data that must be entered for each transaction.

Secondly, the analyst can also affect error rates of an operation through design. The manner in which data must be entered can reduce the chance of errors.

Still, a third aspect of error control is the need to detect errors when they do occur. Checks and balances in the data entry programs, called input validation techniques, also detect errors in input.

### 2.5.4 Avoiding Extra Steps

Sometimes the volume of transactions and the amount of data preparation or data entry jobs resulting from them cannot be controlled. For example, in bank cheque processing runs or in large retail sales firms, the number of transactions to process runs into the tens of thousands.

When the volume cannot be reduced, the analyst must be sure that the process is efficient. The experienced analyst will also avoid input designs that cause extra steps. The effect of saving a single step when feeding details of cheque into the banking process is multiplied many times over in the course of a working day. So is the additions of a single step.

### 2.5.5 Keeping the Process Simple

Perhaps the best advice to analysts is to achieve all of the objectives mentioned in the simplest manner possible. The best- designed system fits the people who will use it in the way that is comfortable for them, and at the same time it provides the error control methods management acceptable to the users. In contrast, one will have to work to get users to accept complex or confusing input designs, and there is no guarantee he will succeed in installing and running complex system. So it is advisable to avoid complexity when there are simple alternatives.

### 2.5.6 Major Concerns Regarding Input

Important points to be considered here are as follows:

- What input is needed?
- How and where is the input created?
- How should the source documents be designed?
- What format should be used for the input records?
- What medium should be used for recording the input?

We will discuss each of the major input concerns briefly:

#### The Inputs Needed

The input needed for any program is determined by the output desired. The analyst must ask the following questions. What information is already in the master file or database? What constant data is required that can be entered from some type of control record ? What information must be supplied by using some type of transaction file? What data should be stored in and accessed from tables? What information can be calculated by the program?

Any time the use of a transaction file is being considered, or the data is to be entered from a terminal, the analyst must check each field to determine whether the data is already in a master file or might be included in a table. The analyst must be concerned that all of the data required to produce that output is entered into the program in the most efficient and cost-effective manner.

**How and where data is generated**

How the data is generated, and where it is generated, has a direct impact on a number of other questions. In a cost- accounting system, much of the data is generated when material is put into production. The analyst should attempt to provide a reliable means of entering data directly into the system from the factory. Data collection devices or special terminals can be used to enter some of the data.

In a retail sales system, a type of scanner device - Bar Code Readers may be used to read price tickets. When charge sales are made, special readers are available that make it possible to use the data stored on the customer's charge card. Whenever possible, the manual keeping of data should be eliminated. In a retail sales application, the only variable data that a clerk might need to key in on a Point of Sale (POS) Terminal having special key for various item categories is the quantity of a given item that is purchased. For processing of electricity bills, the computer system itself first generates an input sheet containing House Number-wise table covering street to street the consumer details - Meter No., reading of Last Cycle, Type of Consumer - Domestic/Charitable Society/Govt./Business keeping the single field blank - present meter reading. This input sheet facilitates the Meter Reader to enter single blank entry after verifying correctness in other columns of table.

Railways Reservation System presents an input screen simply on entering the Train code, Travel Class and Date of travel wherein if reservation is asked, the clerk has to enter passenger's details almost in same way as given by him in the reservation slip.

In large Tea Gardens of Assam spread over several kilometers, Hand held terminals in Mobile Vans are used to enter leaf Phreker's code and weight of leaf plucked on daily basis moving from one garden to another and entire data is updated returning back to Computer Centre which generates weekly payment reports for forced workers and next days schedule for labour deployment in new gardens.

**Designing the source document and the Input Format**

The formats for the input records and the source documents should be determined simultaneously.

The source document can be designed as soon as it is determined what data is needed and where and how it is to be entered into the system. The analyst should work with the data entry supervisor. The design of the documents should permit the personnel recording the data to do so as easily and rapidly as possible. Check boxes can be used, which reduce the time needed to fill out documents and minimize recording errors. Take the case of Electricity bills. It has two parts. The first is called 'MAIN' on which cash/cheque receiving clerk enters the amount received with details including the date and hands over to the customer as a payment made by him. The second is called 'STUB' which are bundled together on day end and sent to Computer Centre for data updation of payments received from the customers for future accounting for ease in transcribing data into a machine-processable form, the locations for each field with the record should be specified on the documents. Identical design of source document and related Data Entry Input screen facilitates Data Entry Operators to enter data at high speed since locations of data to be read are easily identified.

The input record should be designed so that the flow of data on it is the same as on the source document. This decreases the time needed to record data and also reduces errors.

The input format must be designed concurrently with the source document. The factors that must be considered are: record length, field size, use of codes, and the relationship of the source document to the input record. When a terminal is used to make the changes or to add new customers to the file, a formatted screen should be designed that looks very much like the source document. The operator fills in the blanks and these may be used to indicate the number of characters that can be entered in much the same way the form would be filled in manually.

The analyst must understand the characteristics of the data entering the system and determine the field size that should be used. For example, if each customer is assigned a number and the firm now has 9,945 customers, the analyst should allow five positions for the field. If five positions are not reserved for the field, as soon as 54 more records are added, the field will not be large enough to handle the account number. Field sizes are usually determined by studying historical data, projecting future needs, and providing for growth.

**Input Medium**

An input method that requires a minimal amount of data conversion should be selected. If punched-card recorders (keypunches), diskette recorders, key-to-key tape recorders, or terminals are used, the data is usually recorded on a source document and then transferred to the machine-processable medium (cards, diskettes, tape, disk, or directly into a transaction file). In case of process control however, these media based programs directly control the target machines.

## 2.6 INPUT VERIFICATION AND CONTROL

If incorrect data enters the system, it is usually very costly to make the necessary corrections. Also, how expensive would it be to have your operator record a quantity of 100 rather than 10 for a shipment of sports cars? The shipping charges for sending the cars to the customer and then of having them returned would be only one of the costs. While the 90 extra cars were in transit, they would not be available to other customers (which could result in a loss of sales) or could be damaged. There are many methods which are commonly used to verify data entering the system as input. Some of the them are :

### 2.6.1 Key Verification

A second operator rekeys the data already recorded. This method is used for verifying data recorded in punched cards or on diskettes and magnetic tape. Then two floppies are compared to correct record by record which mismatched during comparison after verifying, from the original documents. This is most effective method used by Computer Service bureaus for data validation.

### 2.6.2 Use of Self-checking Numbers

The computer can be programmed to reject numbers that have been transposed or have one or more wrong digits. Check digits and self-checking number routines can be effectively used for numbers in a series, such as student roll numbers, account numbers, part numbers, or invoice numbers are popular for such jobs.

### 2.6.3 Visually Displaying an Identifying Characteristics

When using a terminal, a part number is entered. Displayed in the VDT is the description of the part, which is then visually confirmed by the operator.

### 2.6.4 Hash Totals

Sometimes numbers are added to produce a meaningless total called a hash total. For example, totaling is made of the quantity of all items purchased. When the records are entered and processed, the hash total is compared to the original total. If the two totals agree, it is an indication that all quantities were entered correctly and all records were processed.

### 2.6.5 Checking Between a Range of Numbers

The numbers on the orders being processed on a given day should fall between, say, 4999 (the last number from the previous day) and 6001 (the next order number that will be on all of the orders processed by the next day). If the order number recorded on the input record does not fall within that range, an error message will be generated.

### 2.6.6 Reasonableness Test

Based upon past history, some input can be checked to see if it is reasonable. For example, because of long-standing company policy, it is unlikely that any employee will have more than 20 hours of overtime. If more than 20 hours of overtime are recorded in an employee's current transaction record, an error message will be generated as the data is being edited.

Similarly in 'Date of Birth' field, it is checked that no date is more than 31, month number is not more than 12 and the year is not more than the current year or current year minus minimum age prescribed.

### 2.6.7 Verification of Codes

The pay and fringe benefits are calculated for employees based upon their payroll status. Assuming that the valid status code must be either an H (hourly), S (salaried), T (trainee), or a P (part-time), an error message would be generated if the code used was not an H, S, T, or P.

### 2.6.8 Verification of Data Type

Some input fields should contain only numeric data while others should contain only alphabetic data. The fields can be edited to make certain that only the right type of data is recorded in each field.

### 2.6.9 Verification That Certain Combinations of Data Exist

For example, all students may be coded with either a W or a V. The V denotes a non-work-study student while the W indicates that the student is on work-study. The only valid account numbers for a work-study student are 2155 and 2156. Any other account number for a W-coded student is invalid.

### 2.6.10 Sequence Check

If the numbers in the source documents are serial and the documents are in order, the input records will also be in numerical sequence. A check can be made by the program to determine whether the records are in either ascending or descending order.

**Check Your Progress 1**

1. List out various input devices for feeding the raw data into the system.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

2. Explain briefly the objectives of input design.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

3. List out the various methods commonly used for input verification and control.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

## 2.7 DATA DICTIONARIES

Data dictionary stores description of data items and structures as well as systems processes. It is intended to be used to understand the system by analyst who retrieves the details and descriptions it stores. He takes the help of data dictionary during system design, when information about such concerns as data length, alternate names (aliases) and data use in particular processes must be available. The data dictionaries have also validation information in storage to help the analysts in specifying controls for the system's acceptance of data. The dictionary also contains definitions of data flows, data stores and processes. Data dictionaries can be developed manually or using automated systems. Automated systems offer the advantage of automatically producing data element, data structure and process listings. They also perform cross-reference checking and error detection. Automated dictionary systems are becoming the norm in the development of computer information systems. For further study about data dictionary, please consult unit 4 of block 1.

## 2.8 HOW TO LAYOUT TERMINAL SCREEN

Software is available that make it easy to layout screens. The programmer keys in the required format on the screen, gives the format a name and then stores the format in a file. Whenever the format is to be used as a display in a program, it can be called into the program by using its name. A programmer can also create display screens within a program.

### 2.8.1 Designing of CRT- Input Display Screen

Special considerations are needed for input designs in on-line environments. The analyst must design CRT screens that tell the user what to do and what steps to take next in a way that is brief, yet easy to understand, Menus are often used to present options to users and data fields are marked to show their length while telling the user where to enter the data. Data entry in on-line systems also includes the ability to edit data. In each of these cases, valid entries must be identified and communicated to programmers so that they develop the software to accept correct entries and reject those that are invalid.

### 2.8.2 Basic Rules for CRT-Input Display Screens

There are a few basic rules that must be followed in displaying information on a screen. The important points to remember are:

- Clear the entire screen between formats. There is usually a "clear screen" command that can be used.
- Format the output so that it is easy to read. For example, don't clutter up the screen with unnecessary information. Always display directions or error messages in the same place on the screen, and leave space between items so that the information is easy to read.
- Prevent scrolling. Unless delays are coded into programs, information is displayed on a VDT faster than most people can read. One screen of information should be displayed at a time. When the operator is ready, a specified key is depressed and a new screen of information is displayed.
- Don't overuse color. Often monitors that display information in color are used for terminals or for microcomputers, Carefully controlled use of color can make the information more understandable; uncontrolled use of color adds confusion.
- Be consistent. For example, all instructions may be displayed at the bottom of the screen.
- Develop and use simple conventions such as having an operator enter a '1' or 'Y' for a positive response to a question or a statement.
- Make certain that all directions are clearly stated.
- Test all screens. Have someone totally unfamiliar with the program to load the program and enter the required data without using the help option before same is released commercially.

## 2.9 MAJOR CONCERNS REGARDING CRT— INPUT SCREEN DESIGN

Major concerns regarding CRT - input screen designs are as follows:

- Ease of use
- Improved processing speed
- Menu driven screens
- Emphasizing information on display screens
- Colour use in screen design
- Colour selection
- Editing through display screens

We will discuss each of them below:

### 2.9.1 Ease of Use

One of the most common approaches in designing easy-to-use CRT screen displays is the fill-in-the-blank approach. The analyst simply formats the initial input display so that all the required data elements are clearly labeled and a space is provided for data entry. The display should be aesthetically designed and all descriptions and error messages should be meaningful in clear statements. In other words, care should be taken to avoid symbols and over abbreviations.

### 2.9.2 Improved Processing Speed

Some of the ways to reduce data entry requirements include:

— Designing the screen display so that responses can be abbreviated(for example, entering "Y" instead of "Yes").

— Designing the screen format so that the order of data entry is consistent with the business transaction. This feature eliminates unnecessary "tabbing" around the screen.

— Designing the screen format so that data can be changed are "unprotected" and data that cannot be or should not be changed are protected.

— Using program function keys, which are available on many of the sophisticated terminals and can literally trigger a transaction when a single button is pushed.

— Using terminals that have additional application-specific features (for example, a number pad can be helpful for accounting systems, Point of Sale Terminals in Hotels, Super Bazars etc.

### 2.9.3 Menu Driven Screens

Since on-line systems provide several input and processing options to users, a method of showing the options the user can choose from is needed. Menus serve this purpose. A menu is a screen of information displayed on the CRT that shows the user what functions can be performed and how to select them.

Menus that provide selections to users in a top-down fashion ensure that systems are easy to use, while making the choice of what to do next should be a simple procedure. The system leads the user through a series of decisions until the correct procedure is selected. For instance, a narrative dialogue to lead someone through the steps in editing sales budget data would probably sound something like this:

— You are using the sales system. Which function do you wish to select? (The user depresses "3" for the EDIT MENU option shown on the main menu.)

— You have selected the edit option from the main menu. Which of these editing options do you wish? (The user depresses "1" for the EDIT BUDGET option shown on the edit menu.

— You have selected the edit budget option from the edit Menus accomplish the same thing with few words. This is why analysts and users alike prefer them to write instructions or the display of narrative information on the screen. Notice the uncluttered look of the screen, even after all headings and options are displayed. It would be difficult to preserve the easy-to-read displays if the narrative above was shown instead of the menu option.

### 2.9.4 Emphasizing Information on Display Screens

Often the analyst will use features built into hardware and software to call information or messages to the attention of users. For example, error messages or reports of unacceptable actions (such as submitting invalid data or asking the system to perform a function now expected by the program) are best displayed by using one of the emphasis techniques listed below. Likewise, when the user enters data for processing, the analyst may display a message informing the user the data has been accepted and processing has begun.

The methods of emphasis that many systems offer are:

- Blinking
- Underlining
- Increased/reduced light intensity
- Inverse video (black letters on light screen)

### 2.9.5 Colour Use in Screen Design

When large amounts of information must be presented on a display screen, the analyst may use color to provide better structure and meaning to the information. Related items can be tied together by color so the user can spot them more quickly. For instance, a report showing sales, costs, and profits for various regions may be presented using colors to show each type of cost across departments.

If properly used in the design, color will assist the user in understanding information on the screen, determining what steps are valid, and reducing errors.

Color has four primary uses: (1) identifying valid operations the user can carry out (2) typing related data together (3) highlighting information about organisation performance and (4) communicating messages about system performance.

### 2.9.6 Color Selection

Color meaning are lost if they are used improperly or if excessive color is used. No more than four colors can be quickly recognized by users. The most useful colors are red, green, yellow, and blue. Contrasts are most effectively presented through the color pairs of red and green, yellow and blue, or white and blue.

Color can be effective when there is a reason for its use. However, if the analyst specifics color only for the sake of color, the impact of information may be lost through the distraction of color. Thus its utilization must be carefully planned. The relative contrast/order of legibility amongst most popular colour ways and the background have been shown in the tabular form as given below:

| Order of Legibility | Printing Colour | Background Colour |
|---|---|---|
| 1 | Black | Yellow |
| 2 | Green | White |
| 3 | Red | White |
| 4 | Blue | White |
| 5 | White | Blue |
| 6 | Black | White |
| 7 | Yellow | Black |

| | | |
|---|---|---|
| 8 | White | Red |
| 9 | White | Green |
| 10 | White | Black |
| 11 | Red | Yellow |
| 12 | Green | Red |
| 13 | Red | Green |

### 2.9.7 Editing Through Display Screens

Editing refers to any changes made to records that are stored in the system or that have been submitted for processing but have not yet been stored. Editing also includes deletion of records.

To design an edit function, you must first provide a way for users to tell the system which record of data they wish to edit. Deleting records in on-line systems requires the analyst to provide a way for the users to indicate the proper record, as well as instructing the system that the transaction is a deletion.

Two ways are common. The first allows the user to depress a key that instructs the system to delete the current record on the screen. The other way analysts build delete procedures asks the user to identify the proper record by entering the record key (such as item number) and then depressing a key telling the system to delete the record.

Both methods are common, although the first one is preferred since the user views the record before telling the system to delete the record.

**Check Your Progress 2**

1. What do you understand by 'Data Dictionary'?

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

2. Explain briefly major concerns regarding CRT-Input screen design.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

3. What is the importance of color use in screen design?

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

## 2.10 SUMMARY

In this unit, we have discussed the responsibilities of analysts for the design of input specifications. Various methods for capturing data and validating its accuracy have also been studied. The overall objectives of input design stress minimizing the quantity of data for input while controlling errors and delay. An effective design will also avoid extra steps in input while ensuring that the entire process is quite simple for users and data entry operators. Data dictionary and input screen design have also been discussed briefly.

## 2.11 MODEL ANSWERS

**Check Your Progress 1**

1. Various input devices are:

   (i) Punch card reader

   (ii) Key-to-tape, key-to-disk, key-to-diskette

   (iii) MICR

   (iv) Mark-sensing reader

   (v) OCR

   (vi) Optical bar code reader

   (vii) CRT

2. Various objectives of input design are:

   (i) Controlling amount of input

   (ii) Avoiding delay

   (iii) Avoiding errors in data

   (iv) Avoiding extra steps

   (v) Keeping the process simple

3. Various methods used for input verification and control are:

   (i) Key verification

   (ii) Use of self-checking numbers

   (iii) Hash totals

   (iv) Checking between a range of numbers

   (v) Reasonableness test

   (vi) Verification of codes

   (vii) Verification of data type

   (viii) Verification that certain combinations of data exist

   (ix) Sequence check

**Check Your Progress 2**

1. Data dictionary stores description of data items and structures as well as systems processes. It is intended to be used to understand the system by analyst who retrieves the details and descriptions it stores.

2. Major concerns regarding CRT-input screen designs are:

   (i) Ease of use

(ii) Improved processing speed

(iii) Menu driven screens

(iv) Emphasizing information on display screens

(v) Colour use in search design

(vi) Colour selection

(vii) Editing through display screens

3. Main uses of colour are:

(i) Identification of valid operation properly

(ii) Typing related data together

(iii) Highlighting information about the achievements of organisation

(iv) Communicating messages about system performance

# UNIT 3 OUTPUT SYSTEM DESIGN

Structure



## 3.0 INTRODUCTION

Presenting the data processed by a computer-based information system in an attractive and usable form has become very essential these days. Success and acceptance of a system to some extent depends on good presentation. Therefore, system analyst must know fully how to design output report in an attractive way. Many new output devices are being introduced in the market because of recent development in computer technology. System analyst must be aware of these new technology and try to use these new output devices if possible. Currently, excellent graphic displays are widely available. Speech output systems are also fast emerging.

There are three main reasons why outputs from the computer are required. They are:

(i) For communicating to the persons concerned.

(ii) For re-input to the computer for being connected with other data and further processing.

(iii) For permanent storage.

## 3.1 OBJECTIVES

After going through this unit, you should be able to learn:

- the devices used to output information from a computer
- the design consideration
- the form design
- the design of screen output
- the role of graphics in output
- the record structure and report layout consideration.

## 3.2 TYPES OF OUTPUT

Outputs of a system can take different forms. The most common are reports, displays on screen, printed forms etc. The outputs also vary in terms of their contents, type of stationery, frequency and timing etc. Besides, due consideration also need to be given as to who will use the output and for what purpose. All these points must be kept in mind while designing outputs so that the objectives of the system are met in the best possible way. Outputs of a data-processing system can be placed into two categories:

- Application Output
- Operating Output

### 3.2.1 Application Output

These are the outputs desired out of the system to meet its objectives. These are of three types:

(i) Output as a basis for decision making. This type of output is generally required by management for decision making purposes.

(ii) Output as a requirement to meet a functional objective. Invoices, Excise Gate Pass, Purchase Orders are the examples of such output.

(iii) Statutory Outputs: All organisations are required to produce a certain amount of reports and forms as required by law. Examples are 'C' forms, '3A' and '6A' forms for provident fund, income tax certificates etc.

### 3.2.2 Operating Output

These outputs are mainly generated for use of E.D.P. staff and give various indications as to how the system operates. System logs, error messages, status indicators etc. are the examples of such output. These types of output are not concerned for the users.

## 3.3 OUTPUT DEVICES

The most important output devices are printers, video display units (VDUs) computer output microfilm (microfiche). Printers are mainly used in the following situations:

- when large volume of output is required
- when output is to be distributed to various persons inside/outside the organization
- when batch processing systems are used.

Printer is one of the most common output devices. It provides the user with a permanent visual record of the data output from the computer. Printers can print on ordinary paper or on specially designed forms such as dispatch notes invoices or packing slips. Printers can print 150-2500 lines per minute, each line consisting of as many as 150 characters.

Printers are mainly of two types : impaet printers and non- impact printers. In impact printers, a print head strikes a print ribbon which prints a character on paper. Non-impact printers have no mechanical print head. Characters on such printers are printed using either a jet of ink or a laser beam. These printers are little bit costlier. For example, laser printers are widely used in Desk Top Publishing (DTP) system because of producing excellent quality prints. Impact printers are of two types: line printers in which a whole line is printed at a time and character printers which allow printing of one character at a time. Line printers are quite impact and fast in printing. Generally these printers print in the range of 1000-1200 lines per minute. They are little bit expensive and mainly used for large volume of printing. Character printers are cheaper but slower in speed (around 80 to 90 characters per minute). They are generally used for low volume printing such as in word-processing.

Video display units use Cathode Ray Tubes (CRTs) for display purposes. They have keyboard which is used for entering data. Twenty four lines, each 80 characters long, can be displayed on the display unit. VDUs are widely used in on-line systems to display results and answers to queries. Colour display units can display different colours. The information displayed on a VDU may be plotted on a plotter or recorded on a floppy diskette or video tape recorder.

Computer Output Microfilm(COM) is an output technique that records output from a computer as microscopic images on roll or sheet film. The images stored on COM are the same as the images which would be printed on paper. The COM recording process reduces the size of character 24, 42 or 48 times smaller than would be produced from a printer. The information is then recorded on sheet film called 16mm, 35mm microfilm or 105mm microfiche. The data to be recorded on the microfilm can be retrieved directly from the computer (on-line) or from magnetic tape which is produced by the computer (off-line). The data is read into a recorder where, in most of the system, it is displayed internally on a CRT. As data is displayed on CRT, a camera takes a picture of it and keeps it on the film. After this, the film is processed, either in the recorder unit or separately. Then it can be retrieved and viewed by the user.

A new type of output unit emerging in the market is an audio output unit. Such an output device can be used to announce a result on a loudspeaker. Currently, the most common audio output gives digits or numbers. Short words are also synthesized by the computer. The pattern of bits corresponding to a word is stored in an audio output buffer storage. With the help of speech synthesizer, it is converted to spoken words. Presently, the speech output of synthesizers does not sound in a natural way. Lot of improvement is expected in this direction.

The most common audio output unit answers enquiries such as a request for telephone number, balance in one's saving account number in a particular bank etc.

## 3.4 OUTPUT DESIGN CONSIDERATION

Output to be produced usually depends upon the following consideration:

| | | |
|---|---|---|
| Type of user and purpose | : | Generally different levels of users will have different requirements from the system. Some want exception reports (e.g. when sales fall below a certain level), some want summary reports (e.g. sales quantity and value for each region) while some want details (e.g. list of invoices for a period). Again statutory reports will normally be as per requirement specified under the law and the designer will not have much flexibility to change the format. |
| Content | : | The data that are needed to be included in the output. These will be related to the purpose of the output. |
| Format | : | This refers to the arrangement of data on the report, size of the paper, titles, headlines, colour of the paper etc. |
| Frequency and timing | : | At what frequency (daily, weekly, monthly, annually etc.) and when (after annual closing of accounts, after the end of the fiscal year, before the last day of every month etc.) |
| Volume | : | Often sheer volume of the output deters one from using the output. The sheer bulk of the report may also create problems for handling, filing or printing time. |
| Sequence | : | The usefulness of an output very often depends on the sequence of data printed. A proper sequence will also help distribution of outputs to different users (e.g. pay-slips printed department-wise facilitates easier payment). |
| Quality | : | This relates to the content, appearance and accuracy of the output. Outputs generated for external users should be given special attention in respect of its get-up, quality of paper etc. |
| Type of stationery | : | Reports can be generated on ordinary blank stationery or on specially printed stationery which is useful when most of the contents of the output (e.g. Invoice, Pay-slips etc.) are constant. This type of stationery has the following advantages: |

- Saves computer time.
- Attractive appearance convenient to use by the users.

However, these stationery will normally be costlier than ordinary stationery.

## 3.5 DESIGN OF OUTPUT REPORTS

A report normally has the following structure.

- A report heading which generally appears only on the first page of the report.
- A page heading and sub heading are given at the top of each page of the report.
- A set of records containing some common features may be grouped together. Such a group is named as control group. Control heading can be named as for this group.

**Table 1: Illustration of terminology used in defining the structure of reports**

INDIRA GANDHI NATIONAL OPEN UNIVERSITY
ROLL LIST OF MCA STUDENTS

ROLL LIST OF STUDENTS FOR DIFFERENT SUBJECTS
OF SEMESTER 1/1994-95

LIST OF STUDENTS IN PASCAL

| ROLL NO. | NAME OF STUDENT |
|---|---|
| 1001 | Pankaj Kumar Goel |
| 1002 | Dinesh Kumar |
| 1003 | Upma Rani |
| 1004 | Rajesh K. Aggarwal |
| 1005 | Shoba |
| 1006 | Devender Garg |
| 1007 | Suman |
| . | |
| . | |
| . | |
| 1049 | Bimla Devi |
| 1050 | Hari Om |
| 1051 | Chand Aggarwal |
| 1052 | B.L. Goel |

TOTAL NO. OF STUDENTS IN PASCAL = 52.

**LIST OF STUDENTS IN 'C' LANGUAGE**

| ROLL NO. | NAME OF STUDENT |
|---|---|
| 1070 | KRISHNA GUPTA |
| 1071 | NARENDER KR. GUPTA |
| 1072 | ATMA RAM |

| | |
|---|---|
| 1073 | RAM KUMAR |
| 1074 | RAMESH KUMAR |
| 1075 | KUSHUM RANI |
| 1076 | ANITA KUMARI |
| 1077 | SNEH LATA |
| 1078 | SURENDER KUMAR |
| 1079 | PRADIP KUMAR |
| | |
| 1088 | ANGURI DEVI |
| 1089 | PYARE LAL |

TOTAL NUMBER OF STUDENTS IN 'C' LANGUAGE = 20

TOTAL NO. OF STUDENTS IN SEMESTER 1/1994-95 = 570

END OF IGNOU ROLL LIST FOR SEMESTER 1/1994-95

---

Using table 1, we will explain the terminology:

(i) The report heading appearing once for the report is :

INDIRA GANDHI NATIONAL OPEN UNIVERSITY – ROLL LIST OF MCA STUDENTS

(ii) The page heading which will appear on top of each page is:

ROLL LIST OF STUDENTS FOR DIFFERENT SUBJECTS OF SEM. 1/1994-95

(iii) The control headings and sub headings are as follows:

LIST OF STUDENTS IN PASCAL

ROLL NO. NAME OF STUDENT

Another control heading and sub headings in the same table are:

LIST OF STUDENTS IN C LANGUAGE

ROLL NO. NAME OF STUDENT

(iv) In the above table the line

1001 Pankaj Kumar Goel

appearing below the heading

ROLL NO. NAME OF STUDENT

is called a detailed line.

(v) Abstract of the information at the end of a control group is called the control footing. In the above table, the following lines

TOTAL NO. OF STUDENTS IN PASCAL = 50

TOTAL NO. OF STUDENTS IN 'C' LANGUAGE = 20

are control footings.

We have final control footing also in the report.

The line

TOTAL NO. OF STUDENTS IN SEMESTER 1/1994-95 = 570

is called final control footing.

(vi) Information written in the end of each page is called page footing.

(vii) Information printed in the end of each report is called report footing. The report footing in the said report is

END OF IGNOU ROLL LIST FOR SEMESTER 1/1994-95.

Having decided what report groups should appear in a report, an analyst decides the layout of the report in a print chart which depicts the no. of columns and lines in a report. A sample is shown in Fig. 1. The chart has a number of columns normally equal to that in a printer. The report headings, etc. are entered on the chart by the analyst to examine the way in which the printed output will appear. The most important consideration in designing the print chart is proper form layout for easy readability. The print chart helps in selecting appropriate headings, entering headings, picking control headings, etc. Once a form layout is determined using a print chart, it may be handed over to a programmer who may use an appropriate report generator program. For example, COBOL has a report generation program with its own rules of syntax and semantics. In the print chart, the no. of columns reserved for each detail line to be printed by the program, including control footing and report footing, is indicated. One may use a convention used in a language such as COBOL to describe the format of the individual fields. In the Fig. 1, for example, the convention used is 9 for digits, X for characters, B for Blank, Z for digit or leading zeros and $ for currency symbol.

The general principles to be used in designing outputs reports are:

- The design must be such that it can be read from left to right and from top to bottom.
- The most important item, such as the key field, should be easily available.
- All pages must have a heading and a page number. The date on which the report was prepared should also be printed.
- All columns must be labelled with meaningful labels.
- Too many details should be avoided.
- Control footing abstracts information about groups of detail lines must be effectively used.
- Similarly, page and report footings must be properly defined.

Reports are often sent to government agencies such as the tax department periodically. The format for such reports are specified by government agencies and must be strictly followed.

There are situations in which a large number of forms are to be printed, e.g. the marks record of high school examination may have to be printed for 10 lakh candidates. In such a case, the name of the board which conducted the examination, subjects offered, etc. can be pre-printed in the form and only marks obtained by the candidates are printed by the computer. Similarly, to print documents like share certificates, dividend warrant, etc., pre-printed special stationery is used and only amounts, name, etc. are printed by a program.

PRINT CHART FOR THE PAY REPORT EXAMPLE
PAY REPORT

| EMPLOYEE - NO. | NAME | DAY MON YR | TOTAL PAY |
|---|---|---|---|
| xx99999 | xxxxxxxxxxxxxx | 99B99B99 | 99,999.99 |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 |

** PAY-TOTALS ** 999,999.99

Fig 1: A Sample for Print Chart

**Check Your Progress 1**

1. Fill in the blanks.

   (i) Outputs of a system can take different forms. They are ……………, …………… and …………….

   (ii) Outputs of a data processing system can be categorised as …………… and …………….

   (iii) Two types of printers are …………… and …………….

   (iv) …………… are widely used in DeskTop Publishing system because of producing excellent quality prints.

   (v) …………… is an output technique that records output from a computer as microscopic images on roll or sheet film.

2. List out various considerations on which output to be produced usually depends.

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

3. What are the general principles to be used in designing output reports?

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

……………………………………………………………………………………

## 3.6 DESIGNING SCREEN OUTPUT

Most of the principles for designing printed outputs as discussed in the previous section are valid for displaying screen output also. However, some differences are there. A video display screen has normally 80 columns and 24 rows. Active involvement of the user is required as it is displayed on-line.. User must be given proper instructions for retrieving the required information.

A screen layout for MCA student information system is given below:

**SCREEN FOR MCA STUDENT INFORMATION**

INDIRA GANDHI NATIONAL OPEN UNIVERSITY
MCA STUDENT INFORMATION SYSTEM

| ROLL NO. | NAME | SUBJECTS TAKEN | SEMESTER |
|---|---|---|---|
| 1001 | PANKAJ KR. GOEL | PASCAL | I |
| 1060 | RAJ RANI JINDAL | SADP | II |
| 1230 | SURENDER KUMAR | ORACLE | IV |
| 1231 | NITIN | C++ | III |
| – | – | – | – |

PRESS D FOR DETAILS OF A STUDENT

PRESS C TO CONTINUE

PRESS X TO QUIT

As we have seen in a printed report, it has a screen heading and headings for various fields. At the bottom of the screen, instructions are given to help the user in getting the next screen. By following these instructions, he can exit the system also. The system should be in a position to recover if the user presses the wrong key by mistake. For example, in the above case (MCA student information system), if a user presses E by mistake, the system should give a message that a wrong key is pressed and provide the alternatives again to the user. Now we illustrate a screen for detailed information about a particular student.

**Screen for Detailed Student Information**

INDIRA GANDHI NATIONAL OPEN UNIVERSITY
MCA STUDENT INFORMATION SYSTEM

| | |
|---|---|
| ROLL NO. | 1001 |
| NAME | PANKAJ KR. GOEL |
| DEPTT. | COMPUTER SCIENCE |
| YEAR | 1994-95 |
| ADDRESS | 162 K. VIHAR PITAM PURAM, NEW DELHI. |

PRESS D FOR MORE DETAILS

PRESS ANY KEY TO COME TO MAIN MENU

## 3.7 MENU DESIGN

Designing menu for a system is relatively easy exercise. However, following points should be kept in mind while designing menu:

(i) Hierarchical: Menu should be designed in a structured hierarchical manner from higher level of functions to lower levels. For example, in an order-processing system, the menu should look like as given below:

ORDER PROCESSING
1. DATA ENTRY
2. FILE MAINTENANCE
3. PROCESSING
4. INQUIRY
5. MIS REPORTS
6. EXIT

WHICH ONE ?

Depending on the selection, the next level will be displayed. For example, if we select 2, the screen may be as follows:

```
ORDER PROCESSING
(FILE MAINTENANCE)
1.  CUSTOMER MASTER
2.  PRODUCT MASTER
3.  DISCOUNT MASTER
4.  EXIT
    WHICH ONE ?
```

Again depending on the selection, the screen for the next level will be displayed. For example, if we select 1, the screen may be as follows:

```
FILE MAINTENANCE
(CUSTOMER MASTER)
1.  ADD A RECORD
2.  MODIFY
3.  DELETE
4.  EXIT
    WHICH ONE?
```

Depending on the selection of option, the next level of screen can also be displayed.

(ii) Termination: When a particular program under the control of a menu is over, the menu screen from which the program was initiated should be displayed again. The principle to be followed is that termination from one level should lead to the menu of the next level. However, it is desirable to provide an option in each menu screen to terminate the setting completely or to revert to the highest level screen. This makes the processing faster and operation becomes easier.

(iii) Skipping Menu: Sometimes it happens that certain functions are restricted to particular individual. In such situations, navigating through levels of menu may be irritating and may affect efficiency of operation. In such cases, it is desirable to provide facilities for jumping to the particular screen directly depending on the user-code.

(iv) Security control: Generally, all the individuals will not be allowed to carry out all the functions. For example, a clerk in the Income Tax Deptt. may be allowed only to enter income tax returns. In such cases, menu may be segmented by function to control access to the system functions.

## 3.8 FORM DESIGN AND CONTROL

### 3.8.1 Form Design

We know that data provide the basis for information systems. Without data there is no system,but data must be fed in correct way so that the information produced must be in a format acceptable to the user. In either case, it is still data - the basic element of a printed form.

### 3.8.2 What is Form?

People read from forms, write on forms, and spend many hours in handling forms and filing forms. The data the forms carry come from people, and the informational output of the system goes to people. So the form is a tool with a message; it is the physical carrier of data - of information. It also can constitute authority for action. For example, a purchase order says BUY a customer's order says SHIP, and a paycheck says PAY TO THE ORDER OF. Each form is a request for action. It provides information for making decisions and improving operations.

With this in mind, it is hard to imagine a business operating without using forms. They are the vehicles for most communications and the blue print for many activities. As important as a printed form is, however, the majority of forms are designed by poorly trained people. People are puzzled by confusing forms; they ask for directions on how to read them and how to fill them out. When a form is poorly designed, it is a poor (and costly) administrative tool.

### 3.8.3 Classification of Forms

A printed form is generally classified by what it does in the system. There are three primary classifications: action, memory, and report forms. An action form requests the user to do something - get action. (Examples and purchase orders and shop orders.) A memory form is a record of historical data that remains in a file, isused for reference, and serves as control on key details. (Examples are inventory records, purchase records, and bond registers.) A report form guides supervisors and other administrators in their activities. It provides data on a project or a job. (Examples are profit and loss statements and sales analysis reports.) Fig. 2 is a summary of the characteristics and examples of these forms.

| Class | Characteristics | Examples |
|---|---|---|
| Action | 1. Orders, instructs, authorizes | Application form |
| | 2. Achieves results | Purchase order |
| | 3. Goes from one place(person) to another | Sales slip<br>Shop order<br>Time card |
| Memory | 1. Represents historical data | Bond register |
| | 2. Data generally used for reference | Inventory record |
| | 3. Stationary and remains in one place, usually in a file | Journal sheet<br>Purchase record |
| | 4. Serves as control on certain details | Stock ledger |
| Report | 1. Summary picture of a project | Balance sheet |
| | 2. Provides information about job or details that need attention | Operating statement |
| | | Profit and loss statement |
| | 3. Used by a manager with authority to effect change | Sales analysis<br>Trial balance |
| | 4. Used as a basis for decision making | |

Fig 2: Three Classes of Forms - A Summary

### 3.8.4 Factors to be Considered in Form Design

Form design plays an important role in data processing. Form must have the appearance of a well conceived and attractive design. Some of the important factors which should be taken care of are given below:

(i) Size and shape of the form should be such that it is convenient for handling, filing, sorting etc.

(ii) Arrange the material in a logical order so that it becomes easy to fill it up.

(iii) The form title must clearly identify its purpose. Columns and rows should be labelled to avoid confusion.

(iv) Precise contents should be recorded. Adequate and compact space should be provided for items to be recorded. Pre- printed entries should be taken care of.

(v) Special features like security and control are to be considered.

(vi) Introduce emphasis by shading columns, heavy lines, etc. If the form is to be used for specific clerical operation, for example copying or checking, see that the detail is arranged and spaced to provide maximum help to the operation.

(vii) The form designer should design the form in such a way so as to cover the specific needs of the purpose for which it is designed.

### 3.8.5 Forms Control

Controlling the number as well as the quality of forms in an organisation can be a substantial work. Forms have a tendency to multiply and unless they are checked, it can be a costly affair in many organisations. To control this type of situation, most large organisations establish a formal forms control program.

The first objective of this form control program is to establish standards. Different departments using different forms to accomplish the same task is an unnecessary expense. The job of forms control specialist is to eliminate redundancies among forms to reduce clerical cost.

The forms control specialist also seeks to reduce the number of copies of each form used. Routing one copy of a form through several departments is probably the best way to achieve this.

Forms should be titled, numbered and contain the date of the most recent revision. It is quite helpful to have the form numbers organised so that all forms in a given system can easily be located when that system is under study.

Normally, a form is designed originally by a systems analyst working with the users. When the original supply of the form is reduced to a reorder level, a forms control specialist is generally responsible for its reorder and possible revision. The form is routed to the users for comments and suggested changes. The forms controller coordinates these suggestions and orders the most economical list. For routine office forms which are not likely to change frequently, a reorder of one year's supply is normal.

**Check Your Progress 2**

1. List out various points to be considered in designing menu.

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

2. List out three primary classification of forms.

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

3. Describe various factors to be considered in Form Design briefly.

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

4. What do you understand by 'Form Control'? Explain briefly.

......................................................................................................................

..........................................................................................................................

..........................................................................................................................

..........................................................................................................................

..........................................................................................................................

## 3.9 COMPUTER GRAPHICS

When designing an information system, it should be considered carefully about how the output can best be presented. Text output in many cases is O.K. But for many applications it does not look nice to present the user with pages and pages of textual output. It may take hours together to go through such a large output.

When it is not desired to present volumes of textual data but only summary of data is required, the data are often best presented in graphical form.

Presentation of data in graphics forms was little bit difficult in earlier computer systems. But with the advent of end-user computing, DBMS, electronic spreadsheets, sophisticated graphics software and high-resolution output devices, the use of graphics has increased tremendously.

Presentation of output in graphics format has the following benefits:

- More effective conversion of data into information
- Easier recognition of relationships and trends
- Quick decisions can be taken to make decisions
- Better presentation of output
- Ability to focus attention on important issues
- Capability of presenting ideas in an attractive format that may readily receive attention.

Behind these benefits lies the fact that the mind can absorb information more rapidly from an effective picture than it can from words or numbers. If they are used when appropriate, computer graphics can bridge the gap between computer data and the human mind. To the business professional this means more information in less time.

Using the graphic capabilities provided by integrated spreadsheet packages, data can be graphed in different ways. But some users need more. For example, it can be helpful to construct graphs from the company's central databases, to design more customized graphics than those offered by standard packages, and to use graphics for a wider range of applications than merely spreadsheets. And indeed, a host of more sophisticated computer graphics packages is available. Computer graphics software can be divided into two categories, presentation graphics and decision support graphics. We will briefly describe how each is used.

### 3.9.1 Presentation Graphics

Presentation graphics are used to communicate ideas to those who might be unfamiliar with a situation or who need a simple but highly effective overview of a topic. For example presentation graphics might be used by a sales person to show a customer how several insurance policies compare, by a marketing manager at a long range planning session to show the change in market share between competitive products, or by manufacturing management at a budget session to give an overview of the expected work load in the next quarter.

Those who used presentation graphics need a system that can

- Produce high-quality illustrations.
- Produce a range of colors.
- Allow the user to choose among a variety of print styles or "fonts".
- Reduce and enlarge illustrations.

- Produce high-quality 35-millimeter slides or transparencies.

The data used in presentation graphics may come from different databases in the organisation, from non-computer sources in the organization, and from outside sources. Most illustrations are accompanied by explanatory text, and so any graphics system must be capable of mixing text with graphics.

A simple example illustrates how a pie chart is constructed. The user first types the title for the chart on the keyboard and then enters the following information for each "slice" of the pie : label, value, whether or not the slice should be "exploded" out of the pie for attention purposes, color, and design code (texture). Provided with this input, the graphics software does the rest.

### 3.9.2 Decision Support Graphics

The second computer graphics category is decision support graphics or analytical graphics. Here graphics are used as a vehicle for understanding patterns, trends or relationships in data. Because the objective in using decision support graphics is to learn something about data, the demands made on the quality of the graphics, the type of presentation, and the source of data are quite different from the demands made on presentation graphics.

First, the quality of the illustration is not nearly so important as its ability to present the information in a way that can support the problem solving and decision making process. Second, the color and special graphical effects are not usually necessary. Third, the data for decision support graphics usually come from spreadsheets, local databases, or the firm's central database. If the data are stored centrally, then the graphics system must be able to access the data and use them to produce graphs with a minimum of user involvement. The effective use of graphics offers finely "distilled" information for quick comprehension by decision makers. The speed of comprehension is not merely a matter of convenience but also of being able to make timely decisions.

### 3.9.3 Graphics Hardware/Software

The hardware used in a graphics system falls into several categories, including graphics terminals, graphics boards, graphics printers, and interface devices.

In addition to hardware, a graphics system needs software. It is the software that provides the capability of using different fonts, adjusting the size of the fonts, selecting colors, moving the image from one location on the screen to another, incorporating graphics into the text, and supporting the use of interface devices.

A wide variety of software packages is available in the market. Many provide a standard set of line, bar and pie charts; some offer the option of displaying these in three dimensions; and others allow several charts to be graphed on a single plot

**Check Your Progress 3**

1. List out various benefits for presenting the output in graphic format.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

2. Define the following terms briefly.

   (a) Presentation Graphics

   (b) Decision support Graphics

## 3.10 SUMMARY

In this unit, we have discussed the responsibility of system analyst for output system design. Different type of output devices have been discussed. Output to be produced usually depends upon many considerations. They have been explained in details. Designing screen output or menu design play quite significant role in business applications. Computer graphics are also very helpful in presenting the output in an effective way.

## 3.11 MODEL ANSWERS

**Check Your Progress 1**

1. (i) Report, display on screen and printed forms
   (ii) Application output and Operating output
   (iii) Impact printer and non-impact printer
   (iv) Laser printers
   (v) Computer Output Microfilm

2. Various considerations are:
   (i) Type of user and purpose
   (ii) Contents
   (iii) Format
   (iv) Frequency and timing
   (v) Volume
   (vi) Sequence
   (vii) Quality
   (viii) Type of stationary

3. The general principles are:
   (i) The design must be such that it can be read from left to right and from top to bottom.
   (ii) An important item, such as the key field, should be easiest to find.
   (iii) All pages must have a heading and a page number.
   (iv) All columns must be labelled with meaningful labels.
   (v) Too many details should be avoided.
   (vi) Control footing abstracts information about groups of detail lines must be effectively used.

**Check Your Progress 2**

1. Various points to be considered in menu design are:
   (i) Hierarchical
   (ii) Termination
   (iii) Skipping menu
   (iv) Security control

2. Three primary classifications are:
   (i) Action
   (ii) Memory
   (iii) Report

3. Various factors to be considered in form design are:
   (i) Size and shape of the form should be such that it is easy in handling, filing, sorting, etc.
   (ii) Arrange the material in a logical order so that it is easily filled up.
   (iii) The form title must clearly identify its purpose.
   (iv) Precise contents should be recorded.
   (v) Special features like security and control are to be considered.
   (vi) Introduce emphasis by sharing columns, heavy lines, etc.

4. Form control is a procedure for (i) providing improved and effective forms (ii) reducing printing costs, and (iii) securing adequate stock for all times.

**Check Your Progress 3**

1. Various benefits are:
   (i) More effective conversion of data into information
   (ii) Easier recognition of relationships and trends
   (iii) Quick decision is possible
   (iv) Better presentation of output
   (v) Important issues can be highlighted
2. (a) Presentation graphics are used to communicate ideas to those who might be unfamiliar with a situation or who need a simple but highly affective overview of a topic.
   (b) These graphics are used as a vehicle for understanding patterns, trends or relationships in data.

# UNIT 4 FILE AND DATABASE DESIGN

## Structure



## 4.0 INTRODUCTION

After designing the input and output, the designer begins to pay his attention on the work of file designing or how data should be organised around user requirement. How data are organised depends on the data and response requirements that determine hardware configurations. System analyst is responsible for designing the files and selecting their contents, selecting from options available for organising the data. File organisation may be sequential, index sequential, inverted list or random. Each method has its own uses and abuses.

An integrated approach to file design is the database. The general theme is to handle information as an integrated whole, with a minimum of redundancy and improved performance. Various software techniques are applied to manipulate, describe and manage data. Irrespective of type of data structure used, the main objectives of database are accuracy and integrity, successful recovery from failure, privacy and security of data.

## 4.1 OBJECTIVES

After going through this unit, you will be in a position to learn:

- file concept and its different type
- various methods of selecting data storage medium
- file Organisation
- file design
- database design
- coding system

## 4.2 SELECTING DATA STORAGE MEDIA

In theory, there is a wide selection of devices and media available for file storage, ranging from punched cards to high speed internal memory. In practice, system analyst is interested with bulk storage devices using magnetic media such as magnetic tape and exchangeable magnetic discs. Magnetic drums and fixed discs are also very helpful but these are generally used for special purpose files. Magnetic card devices, although provide large storage capacity, are slow and not so much reliable. Punched cards and paper tape have become obsolete now-a-days and are not considered for bulk storage of data. Magnetic tape units generally used reels holding plastic tape 0.5 inch wide and 2400 ft. in length. It has magnetizable coating on one side. For reading or writing purpose, the tape is transported from one spool to another. It is not possible at all to read from and immediately write on to one reel of tape. Therefore, when we update a file, it becomes necessary to read the existing file from one reel and to create an entirely new version by writing on the another reel. When dealing with a master file updated from time to time, the file being read is said to be the brought- forward file and the newly created one is the carried-forward file (because the former has been brought forward from a previous updating process and the letter will be carried-forward to a future one). Apart from this, these different versions of master file referred to as 'generations' are very helpful for recovery purposes in case the latest versions get corrupted. Three generations (grandfather, father, son) are commonly created and retained before re-using the oldest version.

The magnetable surface of the tape is densely packed with spots of magnetism measured in bits per inch (bpi) along the length of tape. They are arranged in rows across the tape- commonly either seven or nine spots per row-each spot representing a bit. These arrangements are also known as seven or nine track tape. Each row or frame represents a character or byte. Data is recorded on the tape in terms of blocks and size of the block is specified.

### 4.2.1 File Concepts

Files are the heart of a computer application. Before constructing files, we must understand the basic terms used to describe the file hierarchy. The most commonly used terms are data item, record, file and database.

Data Item: A basic or individual element of data is called data item. Each data item is identified by a name and is assigned a value. For example, in a payroll system, Employee Name, Employee Identification Number are the data items assigned the values PANKAJ and 1775 respectively. Data item is sometimes referred to as a field.

Record: The collection of related data items is called a record. For example, in a payroll system, each employee record may have eight separate fields, which are related to print a pay cheque, such as Employee Name, Employee Code, Sex, Designation, Basic Pay, HRA, DA, and Deductions. The analyst also determines the length and type of each field while designing the record. For example, layout of an employee record in a payroll system can be as follows:

| Name of Data item | Type | Length | Decimal |
|---|---|---|---|
| NAME | C | 20 | |
| CODE | N | 5 | |
| SEX | C | 1 | |
| DESIGNATION | C | 15 | |
| BASIC | N | 8 | 2 |
| HRA | N | 6 | 2 |
| DA | N | 6 | 2 |
| Deduction | N | 6 | 2 |

It is necessary to distinguish one specific record from another. System analyst select one data item in the record that is likely to be unique in all the records of a file which is used to identify the record for further processing. This item is called the key field or record key.

File: File is a collection of related records. Each record in a file is included because it pertains to the same entity. In the payroll system, employee file will contain only employee details records and the inventory or stock records will not be kept in the employee file because these are not related to employee details.

**Database:** The highest level in the hierarchy is the database. It is a set of inter-related files for real time processing. It contains the necessary data for problem solving and can be used for several users who are accessing data concurrently.

## 4.3 TYPES OF FILE

There are various types of files in which the records are collected and maintained. They are categorised as:

— Master file
— Transaction file
— Table file
— Report file
— Back-up file
— Archival file
— Dump file
— Library file

### 4.3.1 Master

Master files are the most important type of file. Most file design activities concentrate here. In a business application, these are considered to be very significant because they contain the essential records for maintenance of the organisation's business. A master file can be further categorised. It may be called as a reference master file, in which the records are static or unlikely to change frequently. For example, a product file containing descriptions and codes; a customer file containing name, address and account number are example of reference files. Alternatively, it may be described as a dynamic master file. In this file, we keep records which are frequently changed (updated) as a result of transactions or other events. These two types of master file may be kept as separate files or may be combined, for example, a sales ledger file containing reference data, such as name, address, account number, together with current transaction and balance outstanding for each customer.

### 4.3.2 Transaction

A transaction is a temporary file used for two purposes. First of all, it is used to accumulate data about events as they occur. Secondly, it helps in updating master files to reflect the result of current transactions. The term transaction refers to any business event that affects the organisation and about which data is captured. Examples of common transactions in the organisation are making purchases, hiring of workers and recording of sales.

### 4.3.3 Table

A special type of master file is included in many systems to meet specific requirements where data must be referenced repeatedly. Table files are permanent files containing reference data used in processing transactions, updating master file or producing output. As the name implies, these files store reference data in tabular form. Table files conserve memory space and make the program maintenance easier by storing data in a file, that otherwise would be included in programs or master file records.

### 4.3.4 Report

Report files are collected contents of individual output reports or documents produced by the system. They are created by the system where many reports are produced by the system but printer may not be available for all the reports. This situation frequently arises when the computer carry out three functions - input, processing and output simultaneously, rather than executing each function in sequence. In this case, the computer writes the report contents to a file on a magnetic tape or disk, where it remains until it can be printed. That file is called the report file which contains the unprinted output data. The process of creating it is known as spooling which means that output that cannot be printed when it is produced is spooled into a report file. Then, depending on the availability of printer, the system will be instructed to read the report file and print the output on the printer.

### 4.3.5 Backup

It is a copy of master, transaction or table file that is made to ensure a copy is available if anything happens to the original.

### 4.3.6 Archival

These files are copes made for long term storage of data that may be required at a much later date. Usually, archival files are stored far away from the computer centre so that they cannot be easily retrieved for use.

### 4.3.7 Dump

This is a copy of computer-held data at a particular point of time. This may be a copy of master file to be retained to help recovery in the event of a possible corruption of the master file or it may be part of a program in which error is being traced.

### 4.3.8 Library

Library file generally contains application programs, utility programs and system software packages.

## 4.4 FILE ORGANISATION

A file is organised to ensure that records are available for processing. Before a file is created, the application to which the file will be used must be carefully examined. Clearly, a fundamental consideration in this examination will concern the data to be recorded on the file. But an equally important and less obvious consideration concerns how the data are to be placed on the file.

### 4.4.1 Sequential

It is the simplest method to store and retrieve data from a file. Sequential organisation simply means storing and sorting in physical on tape or disk. In a sequential organisation a records can be added only at the end of the file.. That is in a sequential file, records are stored one after the other without concern for the actual value of the data in the records. It is not possible to insert a record in the middle of the file without re-writing the file. In a sequential file update, transaction records are in the same sequence as in the master file. Records from both files are matched, one record at a time, resulting in an updated master file.

It is a characteristic of sequential files that all records are stored by position; the first one is at the first position, the second one occupies the second position and third is at third and so on. There are no addresses or location assignments in sequential files.

To read a sequential file, the system always starts at the beginning of the file. If the record sought is somewhere in the file, the system reads its ways upto it, one record at a time. For example, if a particular record happens to be the fifteenth one in a file, the system starts at the first one and reads ahead one record at a time until the fifteenth one is reached. It cannot jump directly to the fifteenth one in a sequential file without starting from the beginning.

Using the key field, in a sequential file the records have been arranged into ascending or descending order according to a key field. This key field may be numeric, alphabetic, or a combination of both, but it must be occupy the same place in each record, as it forms the basis for determining the order in which the records will appear on the file.

When we start searching for a particular record in a sequential file the system do not use the physical record key. The system assigns the value of the particular record key as a search key. For example, let a sequential file consists of the records of employee number from 1200 to 1250. Then how to locate or retrieve a record for an employee number 1234? Here employee number 1234 is the search key. When searching for the employee number 1234, the program controls all the processing steps that follows. The first record is read and its employee number is compared with a search key. 1200 versus 1234. Since they do not match, the process is repeated. The employee number for the next record is 1201, and it also does not match the search key. Therefore, the process of reading and comparing records continues until the employee number match the search key. If the file does not contain the employee number 1234, the read and compare process will continue until the end of file is reached. Sequential files are generally maintained a magnetic tape, disk or a mass storage system. The advantages and disadvantages of the Sequential File organisation are compared and given below:

| Advantages | Disadvantages |
|---|---|
| Simple to understand this approach | Entire file must be processed even when the activity rate is low. |
| Locating a record requires only the record key. | Transactions must be sorted and placed in sequence prior to processing |
| Efficient and economical if the activity rate is high | Timeliness of data in file deteriorates while batches are being accumulated |
| Relatively inexpensive I/O media and devices may be used. | |
| Files may be relatively easy to reconstruct since a good measure of built in backup is usually available. | Data redundancy is typically high since the same data may be stored in several files sequenced on different keys. |

## 4.4.2 Random or Direct

For a proposed system, when the sequential files are assumed as a disadvantage, another file organisation called Direct organisation is used. As with a sequential file, each record in a direct file must contain a key field. However the records need not appear on the file in key field sequence. In addition any record stored on a direct file can be accessed if its location or address is known. All previous records need not to be accessed. The problem, however is to determine how to store the data records so that, given the key field of the desired record, its storage location on the file can be determined. In other words, if the program knows the record key, it can determine the location address of a record and retrieve it independently of any other records in the file.

It would be ideal if the key field could also be the location of the record on the file. This method is known as direct addressing method. This is quite simple method but the requirements of this method often prevents its use. Because of many other factors, this method could not become popular. Hence it is rarely used.

Therefore, before a direct organised file can be created, a formula or method must be devised to convert the key field value for a record to the address or location of the record on the file. This formula or method is generally called an algorithm. Otherwise called the Hashing addressing. Hashing refers to the process of deriving a storage address from a record key. There are many algorithms to determine the storage location using key field. Some of the algorithms are:

**Division by Prime:** In this procedure, the actual key is divided by any prime number. Here the modular division is used. That is quotient is discarded and the storage location is signified by the remainder. If the key field consists of large number of digits, for instance, 10 digits (e.g. 2345632278) then strip off the first or last 4 digits and then apply the division by prime method.

For example, the key field is 2345632278 strip off first 4 digits. Then the new key is 632278. Divide the new key by a prime number. Let it be 41. The quotient is 15421, remainder is 17. Hence 17 is the storage address.

Various other common algorithms are also given as under

- — Folding
- — Extraction
- — Squaring

The advantages and disadvantages of direct file organisation are as follows:

| **Advantages** | **Disadvantages** |
|---|---|
| Immediate access to records for inquiry and updating purposes is possible. | Records in the on-line file may be exposed the risk of a loss of accuracy and a procedure for special backup and reconstruction is required. |
| Immediate updating of several files as a result of single transaction is possible. | As compared to sequentially organised, this may be less efficient in using the storage space. |
| Time taken for sorting the transactions can be saved. | Adding and deleting of records is more difficult then with sequential files. |
| | Relatively expensive hardware and software esources are required. |

### 4.4.3 Indexed

The third way of accessing records stored in the system is through an index. The basic form of an index includes a record key and the storage address for a record. To find a record, when the storage address is unknown it is necessary to scan the records. However, if an index is used, the search will be faster since it takes less time to search an index than an entire file of data.

Indexed file offers the simplicity of sequential file while at the same time offering a capability for direct access. The records must be initially stored on the file in sequential order according to a key field. In addition, as the records are being recorded on the file, one or more indexes are established by the system to associate the key field value(s) with the storage location of the record on the file. These indexes are then used by the system to allow a record to be directly accessed.

To find a specific record when the file is stored under an indexed organisation, the index is searched first to find the key of the record wanted. When it is found, the corresponding storage address is noted and then the program can access the record directly. This method uses a sequential scan of the index, followed by direct access to the appropriate record. The index helps to speed up the search compared with a sequential file, but it is slower than the direct addressing.

The indexed files are generally maintained on magnetic disk or on a mass storage system. The primary differences between direct and indexed organised files are as follows:

Records may be accessed from a direct organised file only randomly, where as records may be accessed sequentially or randomly from an indexed organised files.

Direct organised files utilise an algorithm to determine the location of a record, whereas indexed organised files utilize an index to locate a record to be randomly accessed.The advantages and disadvantages of indexed sequential file organisation are as follows:

| Advantages | Disadvantages |
|---|---|
| Permits the efficient and economical use of sequential processing techniques when the activity rate is high. | Less efficient in the use of storage space than some other alternatives. |
| Permits quick access to records in a relatively efficient way this activity is a small fraction of the total workload. | Access to records may be slower using indexes than when when transform algorithms are used. Relatively expensive hardware and software resources are required. |

## 4.5 FILE DESIGN:

The basic factors to be considered in the selection of file media and file organisation method are:

- the method of processing for updating files.
- size of the file
- file inquiry capabilities
- activity ratio of records in the file
- file volatility and
- the response time.

Each of these factors is briefly described in the following paragraphs.

In batch processing, the sequential method of processing, using magnetic tapes, is employed. This method of updating involves re-creation of the master file every time the file is updated. In order to reduce the set-up time, the updating of the file is done after accumulation of a fairly large batch of transactions. Fig.1 illustrates the updating procedure.

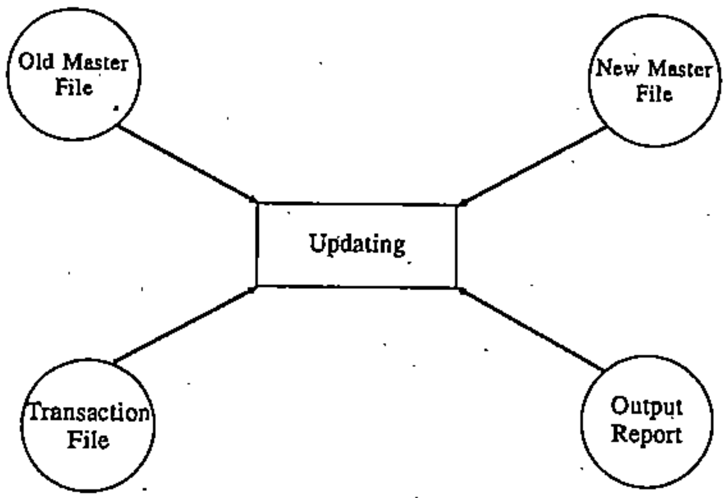



Fig.1: Updating File in Batch Processing

The method of random processing is used to update the files as the transactions occur. This is a widely used method for on- line processing of files through remote terminals. Fig.2 illustrates this method.

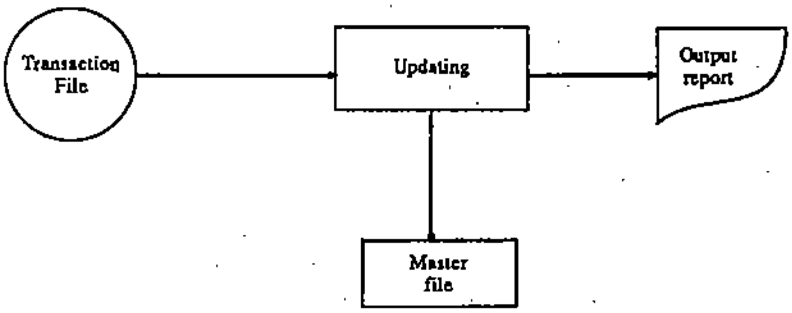



Fig.2: Updating Direct Access File

On-line processing allows for dispersing input/output terminals throughout the organisation, so that various users can have access to the files. This can also be used in a batch processing mode where several files can be updated simultaneously, as illustrated in Fig.3. While the new open order file is created from a batched file of current orders, the customer file and the product file are accessed simultaneously and updated.

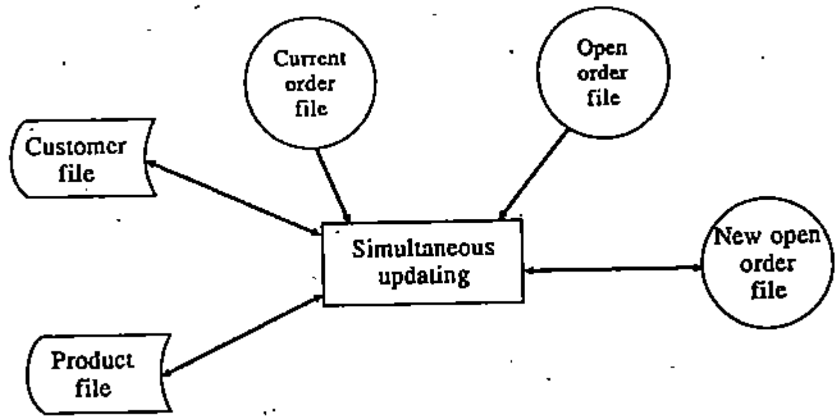



Fig.3: Simultaneous Updating of Multiple Files

Magnetic tape and magnetic disk can accommodate large files. Magnetic tape is more convenient and economical for holding large files, particularly when the records are processed sequentially in a batch mode. The magnetic disk can also hold large volumes of data, but it is relatively expensive. However, it is really suited for on-line processing. In designing the files, the growth potential should be taken into account.

The inquiry capabilities relate to the ease of referring to a specific record in file without any alteration. Direct Access Storage Devices (DASD) can quickly handle file inquiries. A teletype or CRT is used to enter the inquiry specifying the record and the information desired. After the record in the file is accessed, the desired information is communicated by the CPU to the CRT or the teletypewriter. All these steps are completed within a few seconds after entering the inquiry. This inquiry capabilities were severely restricted before the advent of DASD. The inquiry capability of the DASD can be advantageously used without interrupting the normal processing operations.

The time interval between entering an inquiry and getting the reply is called response time. Some applications require a fast response as in the case of airline or hotel reservations, stock quotations in stock exchange, etc. The most suitable medium for such purposes is the DASD.

The activity ratio of a file is an important feature in file design. It is a measure of the proportion of records processed in an updating run. A file with high activity ratio can be processed more economically using the sequential processing method. If more than thirty per cent of the records in a file are used for updating, the activity ratio is considered high. Random or direct accessing is more suitable for updating files whose activity ratio is low, i.e., less than thirty per cent.

Another characteristic of a file is its volatility, which indicates the additions, deletions and changes to the file. When a file is accessed frequently, as in banks, stock exchanges, airlines, etc. in a working period, the file is regarded as highly volatile.

Before designing the file, the analyst should consider specific aspects relating to a file. These aspects should be recorded on a work sheet as shown below:

**File Work Sheet**

File Name : Analyst:

1. File update: Batch [ ] Direct [ ]
2. File Organisation: Sequential [ ] Indexed Sequential [ ] Direct [ ]
3. Processing frequency: On demand [ ] Daily [ ] Weekly [ ] Monthly [ ] Any other [ ]
4. Activity ratio: Low [ ] Moderate [ ] High [ ]
5. Direct access: Yes [ ] No [ ] Occasional [ ]
6. Volatility: Low [ ] Moderate [ ] High [ ]
7. Record Characteristics:

   Type: Fixed [ ] Variable [ ]

   Blocking Factor

   Number of Characters
8. File Dynamics:

   (i) Number of records
   (ii) Yearly additions
   (iii) Yearly deletions
   (iv) Percent growth
9. File media: Tape [ ] Disk [ ]
10. Sources of data:
11. Type of information required and reported:

The above file work sheet should be accompanied by detailed record layouts, which describe in detail the layout of the records in the file.

**Check Your Progress 1**

1. Explain the file concept briefly.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

2. List out various types of file.

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

3. List out the three methods of organizing a file.

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

4. List out the basic factors responsible for selecting the appropriate media and file organisation methods.

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

......................................................................................................................

## 4.6 DATABASE DESIGN

As we have discussed that the organization selected for a particular file mainly depends on the nature of the application for which it will be used. Historically, files have been designed based on specific application. Payroll files are created containing all the data pertinent to a company's payroll system. Similarly, individual files are created for use with the company's personnel, accounts receivable, inventory, and other systems. If the data contained on these files are not carefully delineated, it is very likely that the same data will appear on several of these files. In other words, these files would contain redundant data. For example, both a company's personnel file and payroll file could contain the name and address of each employee. This would mean that a simple change of address would have to be processed twice and possibly three or four times, depending on the number of other files on which these data appear. Clearly, it would be more practical to have each employee's name and address on one file from which it can be accessed by all programs requiring these data. This would reduce the amount of redundant data and minimise the possibility that data contained on a file might be inaccurate because they were never updated. This is but one of the reasons that database technology was developed.

A DATABASE can be thought of as a set of logically related files organised to facilitate access by one or more applications programs and to minimise data redundancy. In other words, a database can be defined as a stored collection of data, organised on the basis of relationships in the data rather than the convenience of storage structures. It is not a replacement for files.

Some general objectives in establishing a database are as follows:

- Eliminate redundant data as much as possible.
- Integrate existing data files.

— Share data among all users.
— Incorporate changes easily and quickly.
— Simplify the use of data files.
— Lower the cost of storing and retrieving data.
— Improve accuracy and consistency.
— Provide data security from unauthorised use.
— Exercise central control over standards.

In addition to the database itself, a set of programs is necessary to facilitate adding new data as well as modifying and retrieving existing data within a database. This set of programs is referred to as a Data Base Management System (DBMS). A data base system merge data into one pool shared by all systems so that any change automatically affects all relevant systems. The following figures defines the difference between the traditional file systems and database management system.

Fig.4 shows the Traditional file systems in which each system is responsible for its own data.

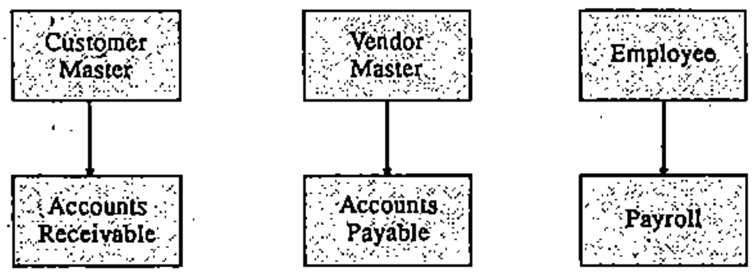



**Fig.4: Traditional File System**

Fig. 5 shows the Data Base Management Systems in which data is centralised.

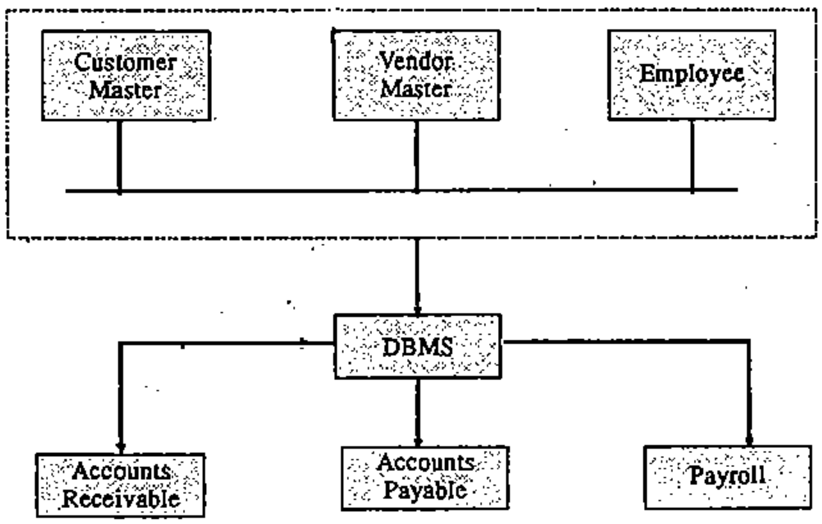



**Fig.5: Data Base Management System**

Specific advantages of data base are:

1. **File Consolidation:** Pooling data reduces redundancy and inconsistency and promotes cooperation among different users. Since data bases link records together logically, a data change in one system will cascade through all the other system using the data.

2. **Program and file independence:** This feature separates the definition of the files from their programs, allowing a programmer to concentrate on the logic of the program instead of precisely how to store and retrieve data.

3. **Access Versatility:** Users can retrieve data in many ways. They enjoy the best of both worlds - sequential access for reporting data in a prescribed order and random access for rapid retrieval of a specific record.

4. **Data Security:** Usually a DBMS includes a password system that controls access to sensitive data. By limiting their access to read-only, write-only, or specified records, or even fields in records, passwords can prevent certain users from retrieving unauthorised data.

5. **Program Development:** Programmers must use standard names for data items rather than invent their own from program to program. This allows the programmer to focus on desired function.

6. **Program Maintenance:** Changes and repairs to a system are relatively easy.

7. **Special Information:** Special-purpose report generators can produce reports with minimum effort.

### 4.6.1 Logical and Physical view of Data

In database design, several views of data must be considered along with the persons who use them. In addition to data structuring, where relationships are reflected between and within entities, we need to identify the application program's logical views of data within an overall logical data structure. The logical view is what the data look like, regardless of how they are stored. The physical view is the way data exist in physical storage. It deals with how data are stored, accessed, or related to other data in storage. There are four views of data out of which three are logical and one is physical. The logical views are the user's view, the programmer's view and the overall logical view, called a schema.

### 4.6.2 SCHEMA

Once a database system has been designed, it will be possible to identify each type of data item, data aggregate, record and set by a name or code. It will be possible to state which data item types go together to make data aggregate types and record types, and to identify which record types are members and owners of set types. A coded set of tables describing this information and stored in the computer system on direct access devices is called a SCHEMA. It is a description of the data structure which is separate from the data itself. The schema describes the areas, their identifiers and page sizes, and indicates how these are related to the records and sets. In other systems, a different set of tables is used for this.

The schema therefore, is the view of the data, the overall logical data structure which is held by the DBMS. Each time a program requires data, the DBMS will look up in the schema for the details of the structure of the data requested. For example if the program requires an occurrence of a set, the DBMS will look up in the schema which record types are required, how to find the relevant records given a certain key by the program, and perhaps also which areas the pages containing the relevant data are stored in.

### 4.6.3 Sub-Schema

In a database system, it is not always possible to allow programmers to write the data division of their choice for reasons of security or control. It is more useful to provide the programmer with a standard description of the logical data to be used in a particular application. All references to data within the program will be for this description, which is called a SUBSCHEMA and is similar to the Schema in structure. The DBMS has the job of matching data requests on a subschema and data requests based on the schema.

## 4.7 TYPES OF DATABASE

In conventional file systems, groups of bytes constitute a field, one or more fields make a record, and two or more records make a file. In a database environment, a group of bytes constitutes a data item or segment, a collection of segments a data entry, and a series of data entries a data set. The complete collection of data sets is the database itself. With traditional processing of files, records are not automatically related, so a programmer must be concerned with record relationships. Often the files are stored and processed by record key, just as we sorted the transaction file. Data bases relate data sets in one of three models: hierarchical, network, or relational.

### 4.7.1 Hierarchical Model

In a hierarchical structure, sometimes referred to as a tree structure, the stored data get more and more detailed as one branches further and further out on the tree. Each segment, or node, may be subdivided into two or more subordinate nodes, which can be further subdivided into two or more additional nodes. However, each node can have only one "parent" from which it emanates. The Fig.6 shows the hierarchical structure.

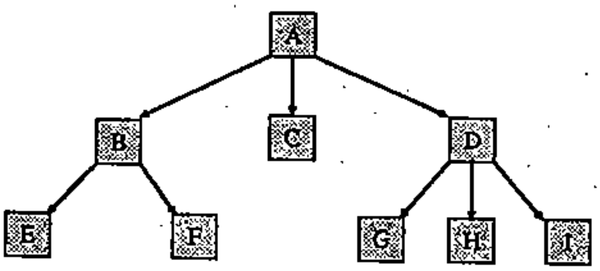



Fig.6: Hierarchical Structure

### 4.7.2 Network Model

The network related data sets are similar to hierarchical ones, except that a node may have more than one parent. Thus a hierarchical DBMS is a subset of network DBMS. The trade off between the simplicity of design of a hierarchical structure and the storage efficiency of a network structure is a very important consideration in database implementation. The Fig.7 shows the Network structure.

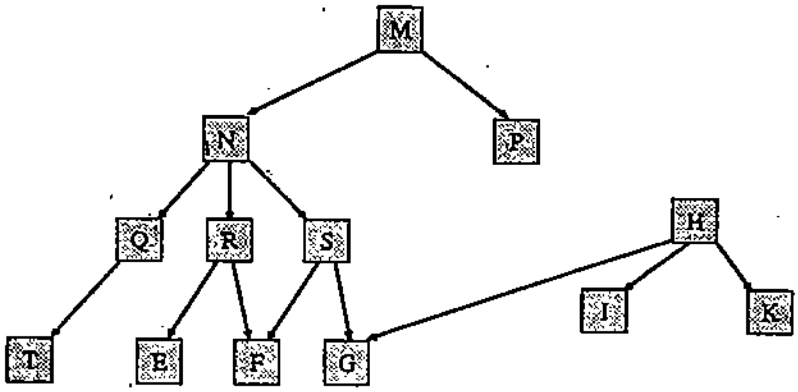



Fig.7: Network Structure

### 4.7.3 Relational Model

The relational structure, however, organises the data in terms of two dimensional tables. That is, Relational data sets order data in a table of rows and columns and differ markedly from their hierarchical or network counterparts. There are no parent or node data sets as shown in Fig.8. In a relational database management systems, we have the same concept of files, records, and fields. Files are represented by two-dimensional tables, each of which is called a "relation". Records, which can be visualized as rows in the table, are called "Luples". Fields can be visualised as columns, and are called by attribute names, or domains.

For example, note that in the supplier table in Fig.8 we have three luples, or rows, and three attribute names or columns. If we need to know the name of the supplier of blue chairs, the relational DBMS searches the type and color columns of the Furniture, Table and finds supplier number 30, and then it scans the supplier table for number 30, which turns out to be PANKAJ'S. Since each "record" is a row in the table and each "field" a column, an inventory system of 1600 Luples, each with 5 attributes, would create a table of 1600 rows and 5 columns.

**FURNITURE**

| Product Number | Type | Color | Quantity in stock | Supplier Number |
|---|---|---|---|---|
| 2589 | Table | White | 4 | 26 |
| 2892 | Chair | Blue | 6 | 30 |
| 3471 | Chair | Light Green | 20 | 133 |
| 3678 | Desk | Brown | 9 | 150 |
| 3689 | Stool | Brown | 25 | 159 |

**SUPPLIER**

| Supplier Number | Supplier Name | Amount of Purchases This year |
|---|---|---|
| 30 | PANKAJ | 26,035.00 |
| 26 | DINESH | 13,960.00 |
| 159 | RAJESH | 75,286.00 |

Fig.8: Relational Structure

A relational DBMS can perform the following basic operations:

- create or delete tables
- update, insert, or delete rows
- add or delete columns
- copy data from one table into another
- retrieve or query a table, row or column
- print, recognize, or read a table or row
- join or combine tables based on a value in a table.

Since the relational structure organises the data in terms of two dimensional tables, they offer great flexibility and a high degree of data security. The relational structure uses relatively little memory or secondary storage. Unfortunately, the process of creating the tables is a rather elaborate procedure. Another disadvantage of this structure is that it generally requires more time to access information than either of the other two structures. This is because much more information must be searched in order to answer queries posed to the systems. In addition, some implementation use a fixed amount of storage for each field, resulting in insufficient storage utilisation. In spite of these disadvantages, the relational structure has gained rapid acceptance and is currently the most popular of the three structures. Many experts predict that it will eventually replace the others completely.

## 4.8 CODING SYSTEM

The information systems are designed with space,time, and cost savings in mind. The coding systems are used to reduce the input, control errors and speed up the entire process. So coding systems are methods in which conditions, words, ideas or relationships are expressed by a code. A code is an ordered collection of symbols designed to provide unique identification of an entity or attribute. It may be a brief number, title or symbol. The main purpose of codes is to facilitate the identification and retrieval of items of information from the system.

## 4.9 TYPES OF CODE

There are many possible coding structures. The main types of codes are described below;

### 4.9.1 Classification Code

Classification is the best described as the establishment of categories of entities, types and attributes in a way that brings like or similar items together according to pre-determined relationships. A classification is by nature an ordered systematic structure. So the classification code places separate entities like events, people, or objects, into distinct groups called classes. A code is used to identify one class from another. Using this code the user classifies the event into one of several possible categories and records the code. Classification codes vastly simplify the input process because only a single-digit code is required. The need for writing lengthy descriptions or making judgements is eliminated.

### 4.9.2 Function Code

Function codes state the activities or work to be performed without spelling out all of the details in narrative statements. Analysts use this type of code frequently in transaction data to tell the system how to process the data. For example, the design for file processing the codes given like the following to perform certain activities.

| | | |
|---|---|---|
| A | — | To add record |
| D | — | To delete record |
| U | — | To update record |
| E | — | To edit record |

otherwise,

| | | |
|---|---|---|
| 1 | — | Addition of records |
| 2 | — | Modification of records |
| 3 | — | Deletion of records |

### 4.9.3 Card Code

Card codes allow the program to distinguish between the type of card and to determine whether the contents of a specific card are correct. These card codes are used in the punched cards only which is used for batch process.

### 4.9.4 Sequence Code

Sequence codes are numbers or letters assigned in series. They tell the order in which events have occurred. This code is simple to use and apply. For example, employee numbers might be assigned consecutively to employees as they hired. It makes no provision for classifying groups of like items according to specific characteristics. An advantage of the sequence code is that it can cover an unlimited number of items by using the fewest possible code digits. As new items occur they are simply assigned to the next higher unused number in.

### 4.9.5 Significant-digit Subset Code

A well conceived coding scheme, using subcodes within larger codes or numbers, can provide a wealth of information to users. Suppose item number will be assigned to the different materials and products a firm stocks or sells. One way is to assign numbers in sequence, starting with the first and going through to the last one. Or a prefix can be added to the identification numbers to further describe the type of item. For example:

PL — refers the product is Plastic and

ST — refers the product is steel.

The codes can be divided into subsets or subcodes, or characters that are part of the identification number that have special meaning. The subcodes tell the user additional information about the item. For example, in a bank examination, the registration number is assigned to each candidate which gives many informations,

Let the assigned number is 11021978

| | | |
|---|---|---|
| 11 | — | centre where the examination is to be held |
| 02 | — | The post number for which the candidate has applied. |
| 1978 | — | The number assigned to the candidate. |

A frequent method of coding is by abbreviation of the name of an entity or attribute. The main ways of doing this are by mnemonic codes and acronyms.

### 4.9.6 Mnemonic Code

Mnemonic code construction is characterised by the use of either letters or numbers, or letters and numbers combinations, which describe the items coded, the combinations having been derived from descriptions of the items themselves. Mnemonic codes produce fewer errors than other types of code where the number of items is relatively small and stable. For example, M and F are more reliable for Male and Female than 1 and 2, and Y and N for Yes and No respectively. Another example for combination of letters and numbers is, to describe 16 inch color television set, a use code is TV-CL-16. Also the unit of measure codes are frequently mnemonic codes. For example, cm - centimeter, m - meter, km - kilometer

### 4.9.7 Acronym

The acronym is a particular type of mnemonic representation formed from the first letter or letters of several words. An acronym often becomes a word in itself. For example,

MCA — Master of Computer Applications

RADAR — Radio Detecting and Ranging

**Check Your Progress 2**

1. List out the specific advantages of database.

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

2. List out various types of database structure.

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

3. What do you understand by Coding System ?

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

4. List out various types of code.

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

## 4.10 SUMMARY

In this unit, we have discussed file concept and several categories of data files. Three most important methods of file organisation have also been explained in a systematic way. Basic factors to be considered in the selection of file media and file organisation have been pointed out in this unit. Database design, its various types are also included. Coding design also play significant role in business data processing system. Different types of code have been defined.

## 4.11 MODEL ANSWERS

**Check Your Progress 1**

1. File is a collection of related records. Each record in a file pertains to the same entity.

2. Various types of files are:

| | | |
|---|---|---|
| (i) Master | (ii) Transaction | (iii) Table |
| (iv) Report | (v) Back-up | (vi) Archival |
| (vii) Library | (viii) Dump | |

3. Three methods are:

| | | |
|---|---|---|
| (i) Sequential | (ii) Random | (iii) Indexed |

4. The basic factors to be considered in the selection of file media and file organisation method are:
   (i) Method of processing for updating files
   (ii) Size of the file
   (iii) File inquiry capabilities
   (vi) Activity ratio of records in the file
   (v) File volatility
   (vi) Response time

**Check Your Progress 2**

1. Specific advantages of database are:
   (i) File consolidation
   (ii) Program and file independence
   (iii) Access versatility
   (iv) Data security
   (v) Program development
   (vi) Easier way of program maintenance

2. (i) Hierarchical (ii) Network (iii) Relational

3. The coding systems are used to reduce the input, control errors and speed up the entire process. Coding systems are considered as methods in which conditions, words, ideas or relationships are expressed by a code. A code is an ordered collection of symbols designed to provide unique identification of an entity or attribute. It may be a brief number, title or symbol.

4. Various types of code are:

| | |
|---|---|
| (i) Classification code | (ii) Function code |
| (iii) Card code | (iv) Sequence code |
| (v) Significant digit subset code | (vi) Mnemonic code |
| (vii) Acronym | |

# Notes

Uttar Pradesh
Rajarshi Tandon Open University

# BCA-07

## Elements of Systems Analysis and Design

Block

# 3

## SYSTEM DEVELOPMENT AND IMPLEMENTATION

# BLOCK INTRODUCTION

After the requirements of a system have been spelt out and the design considerations finalised, the task of the development and implementation of the system is to be taken up.

This block covers the different aspects of this phase of the systems life cycle. The first unit focuses on the different tasks involved in system development and outlines the relevant factors to be considered prior to the beginning of the developmental work. Attention is drawn towards the importance of benchmarking for this purpose. The second unit reflects the issue of quality, which is becoming more and more important in current times. In fact, there is a clear shift from attitudes of quality control to quality assurance, because failure of software in mission critical applications is just not acceptable. The work of the development and implementation could never be carried out successfully if it is not accompanied by proper documentation, which is necessary for programmers and software engineers as well as the users to understand clearly what is being attempted. This important issue of documentation is covered in unit 3 of this block. The fourth unit of this block addresses itself to the issues involved in the implementation. Many of these are not programming-oriented, but related to criteria for selection of hardware and software, alternative ways of obtaining the required services and the issue of drawing out contracts for computer related resources.

[illegible]

# UNIT 1 SYSTEM DEVELOPMENT

## Structure



## 1.0 INTRODUCTION

The fourth phase in the life cycle of a system is the software development. In this development phase, the computer- based business system is developed to conform to the design specification prepared in the preceding phase. This phase involves heavy expenditure because of recruiting additional staff for the purpose of software development, purchase of machine, materials and the use of computer facilities. The principal activities performed during the development phase are: (a) External system development and (b) Internal system development. The major activities that comes under external system development are implementation planning; preparation of manuals and personnel in-house training; and equipment acquisition and installation. Software development and performance testing are considered to be the principal activities of internal system development.

## 1.1 OBJECTIVES

After completing this unit, you should be able to:

- understand different tasks of system development
- know the importance of prototype
- find out various factors to be considered prior to system selection
- understand the term 'benchmark'
- define the various parameters responsible in considering the selection of a language

## 1.2 TASKS OF SYSTEM DEVELOPMENT

Major tasks of system development have been summarised below:

(i) Implementation Planning: After the initiation of development phase is approved, implementation planning starts. Essential parts of implementation plan are:

(a) A plan for testing the computer program component, both as the integrated assembly of its individual programs and as an element of the overall business system.

(b) A plan for training the personnel associated with the development of software on the new system. This includes persons who will provide inputs to, receive outputs from, operate or maintain the new system.

(c) A conversion plan that provides for the conversion of procedures, programs and files preparatory to actual changeover from the old system to the new one.

(ii) Software development phase: Software development phase can work along with the implementation planning efforts. If it is necessary, system flowcharts are expanded to show additional detail for the computer program components. The complete database is developed. Input and output files are identified and computer program logic flowcharts are prepared for each computer program component.

(iii) User Review: Reviews are held with the principal user throughout the development phase. A review of test plans, training plans, and conversion plan is quite important because users are directly involved in implementation activities. Users' concurrence with the implementation plan is extremely important to carry on the software development work in an efficient way.

(iv) Equipment acquisition and installation: In the design phase, special hardware required to support the system may have identified. If the hardware is not ordered during the design phase, it is proper time to go for it. It is also true that all hardware components are not required at a particular time because the needs vary depending upon the type of software being developed. It is, therefore, necessary that a proper schedule should be prepared for acquisition of hardware components.

(v) Coding, debugging and testing of computer program: Each of the computer programs that make up the entire system is coded and debugged. This means that each computer program is compiled error free and successfully executed using the test data prepared by the programmer.

(vi) System testing: System tests are performed to verify that the computer based business system has met its design objectives.

(vii) Reference manual preparation: Proper reference manual for the various individuals who will be associated with the new computer based information system must be prepared.

(viii) Personnel training: Operating, programming and user personnel are trained using the reference manuals, forms and procedures as training aids. All sort of training activities must be completed prior to the user acceptance review which occurs at the end of the development phase.

(ix) User acceptance review: At the end of development phase, the computer based system is reviewed by the management of the user organisation. Representatives of the information service organisation and other affected organisations take part in this review. Design phase report, development phase report and test reports are some of the important documents which are responsible for the acceptance review.

## 1.3 PROTOTYPE INSTALLATION

A prototype is the process of creating, developing and refining a working model of a final system. It does not contain all the features or perform all the necessary functions of the final system. Rather, it includes large number of elements to enable individuals to use the *proposed* system to determine what they like and do not like and to identify features to be added or changed. Application prototyping, the process of developing and using the prototype, has the following characteristics:

(i) The prototype is a live, working application.

(ii) Its main purpose is to test out the assumptions made by the analysts and users about the features of required system.

(iii) Prototypes can be quickly created.

(iv) They follow an iterative process.

(v) They are relatively cheap.

Application prototyping has two primary uses. The first one is that it is an effective device for clarifying user requirements. Written specifications are typically created as a vehicle for defining application features and the requirements that must be satisfied.

A second use of application prototyping is to verify the feasibility of a system design. Analysts can make experiment using different application characteristics, evaluating user reaction and response.

The rationale for application prototyping is a direct outgrowth of the need to design and develop information systems quickly, efficiently and effectively.

Application prototyping is a proven technique that improves the overall effectiveness of the development effort for the benefit of user, analyst and the organisation as a whole.

## 1.4 HARDWARE AND SOFTWARE SELECTION AND PERFORMANCE

### 1.4.1 Hardware Selection

The decision to acquire computer hardware or software must be handled in the same way as any other business decision. The variety of sizes and types of computing resources available puts a burden on the analyst who must select suitable hardware, software or services and advise the top management accordingly.

Today, selecting a system is a serious and time-consuming business. The time spent on the selection process is a function of the applications and whether the system is a basic micro-computer or a mainframe. In either case, planning system selection and acquiring experienced help where necessary pay off in the long run.

There are various important factors which should be considered prior to system selection. They are:

(a) Define system capabilities that make sense for the business.

(b) Specify the magnitude of the problem; that is, clarify whether selection entails a few peripherals or a major decision concerning the mainframe.

(c) Assess the competence of the in-house staff.

(d) Hardware and software should be considered as a package.

(e) Develop a time frame for the selection process.

(f) Provide user indoctrination. This is crucial, especially for first-time users. Selling the system to the user staff, provide adequate training and creating an environment conductive to implementation are pre-requisites for system acquisition.

The selection process should be viewed as a project and a project team should be formed with the help of management. The selection process consists of several steps which are discussed below:

1. **Requirements analysis:** The first step in selection is understanding the user's requirements within the framework of the organisation's objectives and the environment in which the system is being installed.

2. **System specifications:** System specifications must be clearly defined. These specifications must reflect the actual applications to be handled by the system and include system objectives, flowcharts, input-output requirements, file structure and cost.

3. **Request for proposal:** After the requirement analysis and system specifications have been defined, a request for proposal is prepared and sent to selected vendors for bidding.

4. **Evaluation and validation:** The evaluation phase ranks various vendor proposals and determines the one best suited to the user's requirements. It looks into items such as price, availability and technical support. System validation ensures that the vendor can, in fact, match his/her claims, especially system performance.

5. Vendor selection: This step determines the vendor with the best combination of reputation, reliability, service record, training, delivery time, lease/finance terms. The selected vendors are invited to give a presentation of their system. The system chosen goes through contract negotiations before implementation.

### 1.4.2 Software Selection

Software selection is a critical aspect of system development. There are two ways of acquiring software: custom- made or "off-the-shelf" package. Today, there is great demand for these packages because they are quite cheap. There are other benefits also.

(i) A good package can get the system running quickly.

(ii) MIS personnel are released for other projects.

(iii) 'Home-grown' software can take more time and its cost cannot be predicted.

(iv) Package can be tested before purchasing it.

Some drawbacks of software packages are:

(i) These packages may not meet user requirements in all respect.

(ii) Extensive modifications of a package usually results in loss of the vendor's support.

It can be observed that price alone cannot determine the quality of software. A systematic review is crucial for selecting the desired software. Prior to selecting the software, the project team must set up criteria for selection. The criteria for software selection are:

(a) **Reliability:** gives consistent results without any failure for a specified time period.

(b) **Functionality:** functions to standards.

(c) **Capacity:** satisfies volume requirements of the user.

(d) **Flexibility:** adapts to the changing needs

(e) **Usability:**is user-friendly.

(f) **Security:** maintains integrity and prevents unauthorised user.

(g) **Performance:** delivers the results as expected.

(h) **Serviceability:** has good documentation and vendor support.

(i) **Ownership:** has right to modify and share use of package.

(j) **Minimal costs:** is justified and affordable for intended application.

**Check Your Progress 1**

1. List out the major tasks of system development.

.........................................................................................................

.........................................................................................................

.........................................................................................................

.........................................................................................................

.........................................................................................................

2. What are the important factors to be considered prior to system selection?

.........................................................................................................

.........................................................................................................

.........................................................................................................

.........................................................................................................

..........................................................................................................

3. Explain briefly about the criteria for software selection.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

## 1.5 BENCHMARK TESTING

The term "benchmark" was derived from the days when the machinist in a factory would use measurements at each bench to determine if the parts he was machining were satisfactory. In the computing field, to compare one system with another, you would run the same set of "benchmark" programs, through each system.

A benchmark is a sample program specially designed to evaluate the performance of different computers and their software. This is necessary because computers will not generally use the same instructions, words of memory or machine cycle to solve particular problem. As regard, evaluation of software, benchmarking is mainly concerned with validation of vendor's claims in respect of following points:

- minimum hardware configuration needed to operate a package.
- time required to execute a program in an ideal environment and how the performance of own package and that of other programs under execution is affected, when running in a multi-programming mode.

The more elaborate the benchmarking, the more costly is the evaluation. The user's goals must be kept in mind. Time constraints also limit how thorough the testing process can be. There must be a compromise on how much to test while still ensuring that the software (or hardware) meets its functional criteria.

Benchmarks can be run in almost all type of systems environment including batch and on-line jobs streams and with the users linked to the system directly or through telecommunications methods.

Common benchmarks test the speed of the central processor, with typical instructions executed in a set of programs, as well as multiple streams of jobs in a multiprogramming environment. The same benchmark run on several different computers will make apparent any speed and performance differences attributable to the central processor.

Benchmarks can also be centered around an expected language mix for the programs that will be run, a mix of different set of programs and applications having widely varying input and output volumes and requirements. The response time for sending and receiving data from terminals is an additional benchmark for the comparison of systems.

Benchmark is one of the evaluation techniques used by the computer purchasers to determine which marking is best for them in marking out their requirements in terms of both speed and cost.

## 1.6 PREPARING SOFTWARE DEVELOPMENT CYCLE

Software development, which involves writing of programs, begins after systems design has been completed. Again, the work done in the previous phase is the foundation on which the work in this phase will be built. As long as system design has followed the principles of structured design, software development phase will start properly. But if the structured principle have not been properly followed, the results may be disastrous.

In the structure chart (Fig. 1), it is quite clear that an information system can be decomposed into a group of related modules, each of which represents a self-sufficient function within the new system. This modular approach, together with certain programming guidelines and regular reviews, represent the fundamental working concepts of structured programming.

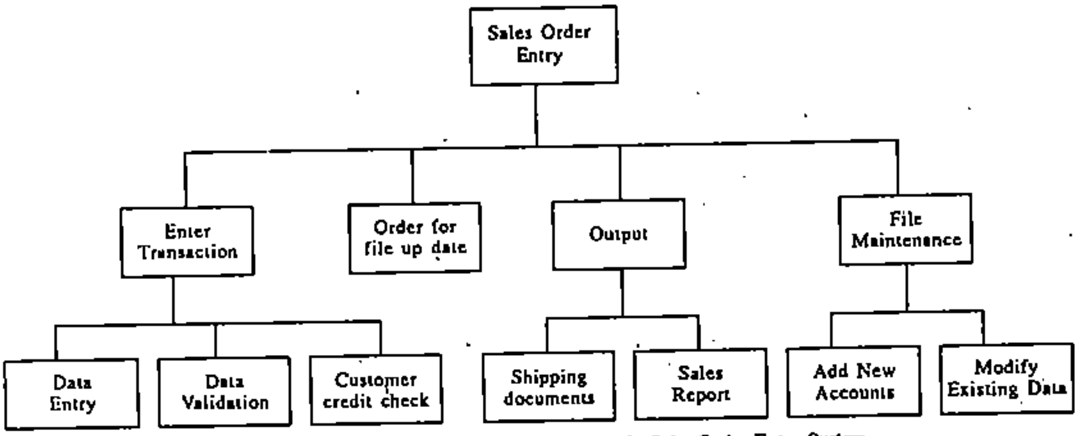



**Fig. 1: A Structure Chart of a Sales Order Entry System**

### 1.6.1 Identifying Programs

Every large system consists of number of small programs because these smaller programs are easier to write, modify, troubleshoot and maintain as compared to very large and complex program. The modular approach serves as the basis for identifying separate programs.

### 1.6.2 Program Logic and Flowcharts

As we move closer to the time when programmers will start writing programs called coding. The detailed logic behind each program is generally the program flowcharts, data flow diagrams or in a more English-like form called pseudocode.

The program flowchart is a detailed graphical representation of the logical flow of data within that module. It serves as the logical road map that the programmers will use to write programming code. When following structured programming principles, the program flowchart must use the standard symbols as shown in Fig. 2 and certain guidelines concerning control structures which will now be described:

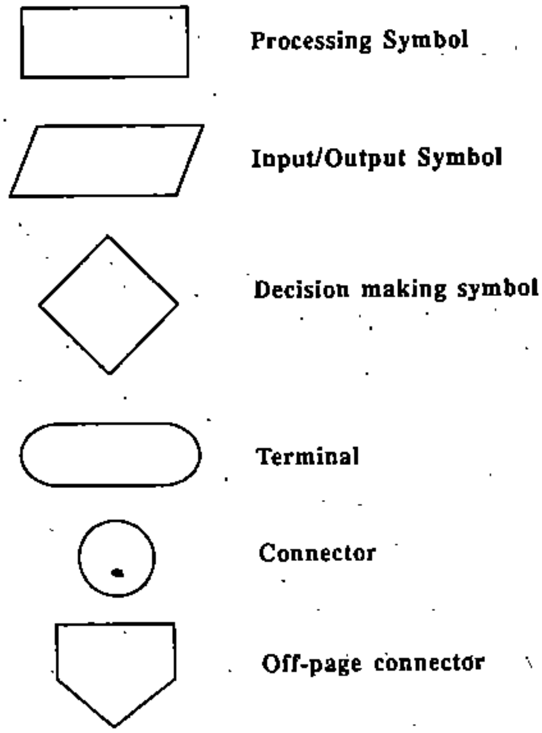



**Fig. 2: Standard Symbols Used to Construct a Program flowchart**

### 1.6.3 Control Structures

Any program whether it may be simple or complex, can be developed using only three basic control structures: (i) simple sequence (ii) if-then and (iii) do-while. Let us describe each briefly.

**Simple Sequence:** A simple sequence structure is used when a series of steps must be carried out in linear sequence. These steps begin with the first and end with the last. Fig. 3 shows a simple sequence containing some of the steps necessary to validate employee wage data, one module in a payroll system.

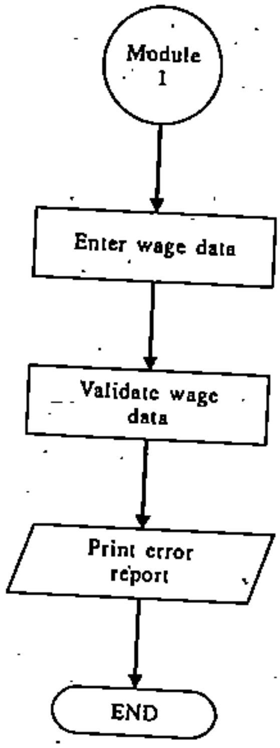



Fig. 3: A Simple Sequence Structure Example

**If-then Structure:** The if-then structure is used to transfer control from one point in a program to another. This transfer is based on meeting out certain condition. Suppose we want to write a program which all employees with more than 10 years of experience are to be listed in a specific report. Fig. 4 shows how the if-then structure can be used to perform this test.

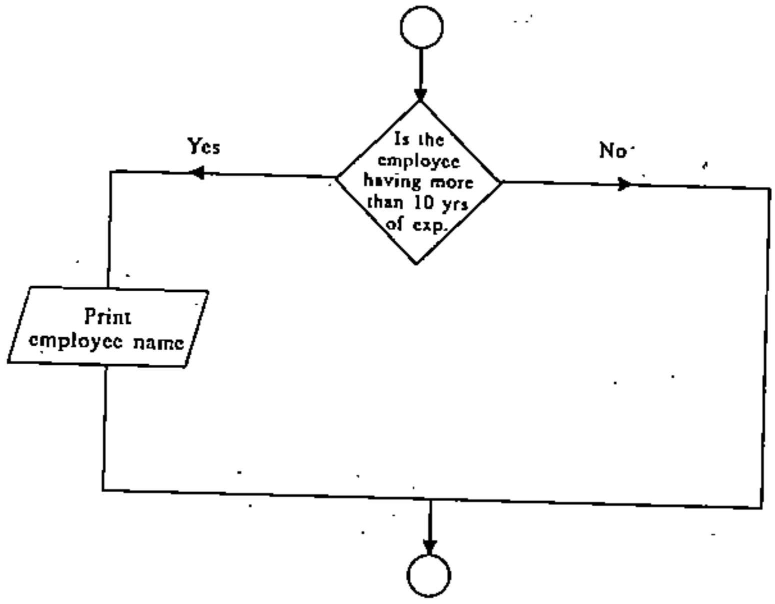



Figure 4: If-then Structure Example

**Do-while Structure:** The do-while structure is used when it becomes necessary to loop through or repeat over and over a sequence of steps. Looping starts at the entrance of the structure and continues while a pre-specified condition still exists.

It is important to remember that a do-while structure tests immediately for a pre-specified condition and does not allow any processing between the entry of the do-while structure and this test. Processing within the structure can only be done after the test has been made and in the line that returns to the top of the do-while structure. Therefore, a record must be read before a do-while structure is entered for the first time.

Suppose we want to access an employee file, read each record and print its contents. As you can see from Fig. 5 a record is read before the do-while structure is entered for the first time. Immediately upon entering, the test for more record is made and if there are more record then the contents of the first record will be printed and another record will be read. This process continues until the last record. At that time, control passes out of the structure and to the next program step.

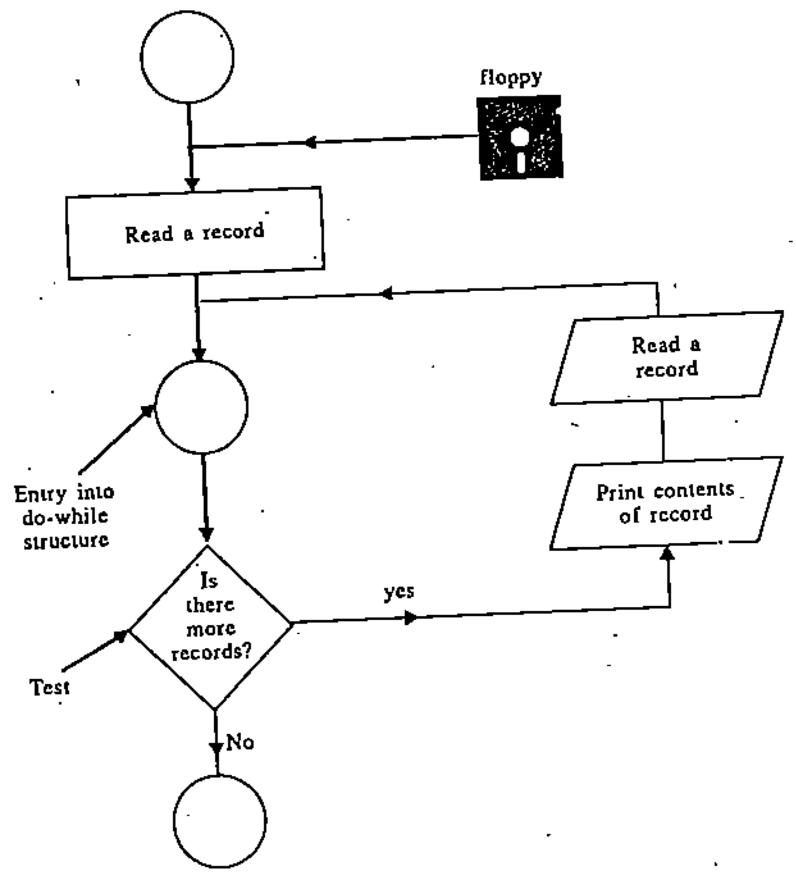



Fig. 5: A Do-while Structure Example

**Combining Control Structure:** Each of these three control structures can be combined to form complex flowcharts and complete programs. Fig. 6 is a complete program flowchart for a process that reads accounts receivable records from a file and prints a list of all those customers whose accounts are more than 60 days overdue and whose balance is greater than Rs. 1000.00.

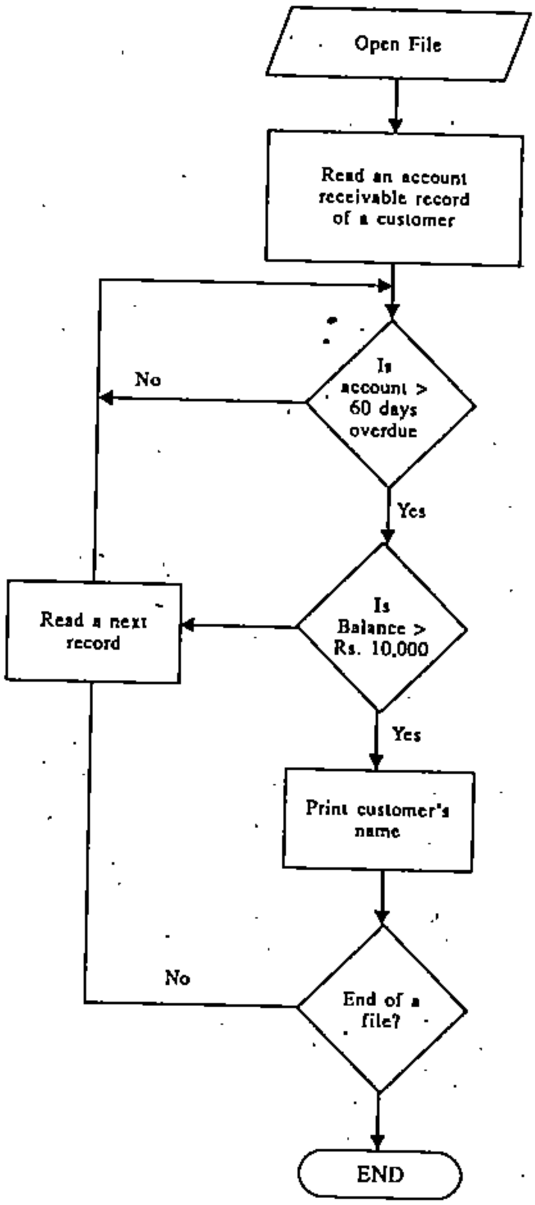



Fig. 6: A Complete Program Flowchart

### 1.6.4 Pseudocode

These program flowcharts show the logic of a program. Pseudocode often used as an alternative to this technique, expresses the logic of a program in English statements. It is a verbal rather than a graphic in nature. Fig. 7 shows various steps necessary to read and print the contents of a file with the help of flowchart and pseudocode.

One advantage of pseudocode is that it closely resembles the form that the actual programming code will take. Another advantage is that it avoids laying out symbols on paper.

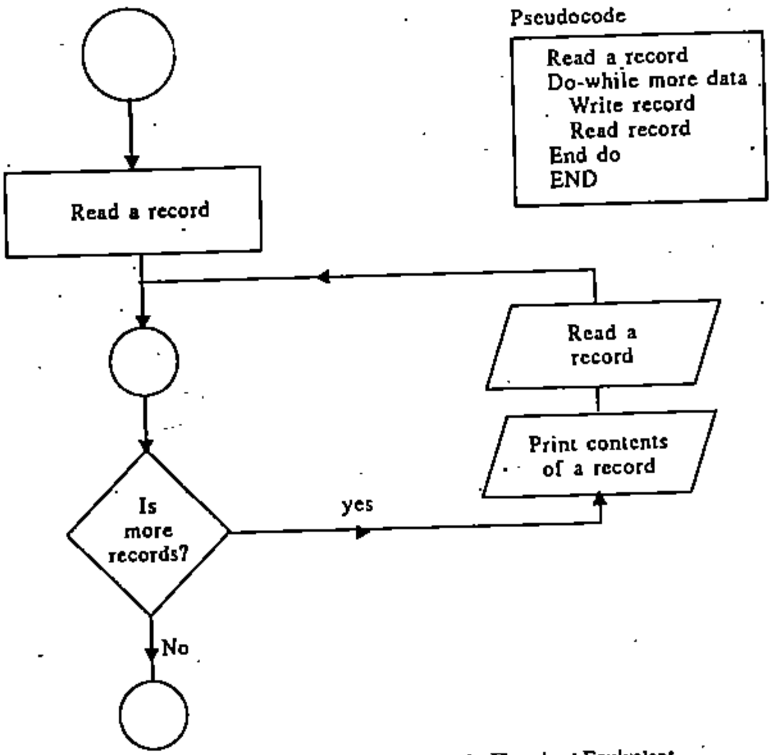



Fig. 7: Pseudocode and a Flowchart Equivalent

## 1.7 SOFTWARE SPECIFICATION LANGUAGE SELECTION CRITERIA

After the program flowcharts or pseudocode have been completed, a programming language must be chosen. Often that choice will depend on the language in which most of an organisation's other programs are written. This ensures consistency and eases the maintenance problem. Nevertheless, a number of different languages are currently in use, only a few of them need to be considered for the purpose of appreciating the issues involved in choosing a programming language. The languages that are discussed are: FORTRAN, COBOL, BASIC, C and dBASE.

**FORTRAN (FORmula TRANslation)** is a compact language that serves the needs of both the Scientists and Economists. Main advantage of this language is that it supplies large-library of Mathematical and Engineering sub-routines being utilised by the programmer in solving different type of problems of numerical and scientific in nature. Handling files containing voluminous data is definitely not a strength of this language.

**COBOL(COmmon Business Oriented Language)** is one of the most popular languages used for large data processing problems. Its main strengths lie in handling files of large size and the ease of understanding and editing the program. COBOL can solve simple arithmetic problems but does not help in solving any complex mathematical problems.

**BASIC (Beginners' All-purpose Symbolic Insrruction Code)** is the most popular conversational programming language. It is simple to understand. Various versions of BASIC have been developed by computer manufacturers for their computers. It is suitable for both mathematical and business problems. It has been specially designed for use in time-sharing environment but can also be used as a standard programming language in a batch-processing environment. The main problem in BASIC language is that most versions of this language do not support indexed files.

C is a quite powerful programming language which is as compact as its name. C can be used where programmer wants to have more control over the hardware because it provides features that would typically be possible with machine/assembly languages only. This language is quite popular among people who are engaged in developing system software like operating systems or other utilities. The disadvantage of this language is that the program is not too easily decipherable and it takes a long time to learn.

dBASE is a Fourth Generation Language which is more an application development tool rather than a programming language. It provides features to store and retrieve data. The major advantage of this language is the query and reporting facility which helps in generating the report quickly. It is quite simple and easy to understand.. If the volume of data to be processed is very large, the program can become quite slow.

Numbers of parameters responsible in considering the selection of a language are:

- volume of data
- complexity of processing
- compatibility with other systems
- types of input/output
- development efforts

### 1.7.1 Volume of Data

This covers two aspects:

(i) Number of files

(ii) Number of records

Some languages put a restriction on the number of files that can be accessed at a time. For example, dBASE III + can open ten data files only whereas COBOL does not have such restriction. It is also observed that most languages tend to become slower when the file size become larger. This slowing down feature due to large number of records is usually found in Fourth Generation Languages. There is also the other extreme where certain tools are not desirable for a very low volume of data. This is typically true of DBMS based products.

### 1.7.2 Complexity of Processing

Some applications take little bit input, do small calculation and generate output. There are quite simple reporting programs. These programs would not require much computation work. On the other hand, some applications involve plenty of computation, for example analysing data from a drilling rig exploring for oil. These would also benefit from the use of mathematical co-processors. FORTRAN is a language that is desired for such type of applications. Some spreadsheets also help in doing complex computations. In case the application does not involve complex computation but require repetitive computations, a language like COBOL or dBASE III would be desirable.

### 1.7.3 Compatibility with other Systems

Any new system cannot exist in isolation but has to co-exist with other systems. Generally we face the problem that some data files may have to be accessed by the old and new systems. If this is not adhered to, the result could be chaotic and may also require additional overheads. It is also possible that the two systems, if in different languages, may not be able to read the same data files e.g. COBOL programs cannot directly read a dBASE III file. This compatibility issue has to be kept in mind even when two parts of the system are written in different languages. In such a situation, it may not suffice that sharing of data is permitted, and two programs may need to invoke each other.

### 1.7.4 Types of Input/Output

Two issues involved are:

(i) Types of Input/Output device

(ii) Complexity of Format

While considering the types of input/output devices that are required, the answer is normally yes or no. For example, if a program needs to give output to a plotter, dBASE may not be suitable.

Complexity of format is little bit more difficult to decide. Some applications may definitely need graphic output, in which case the options are to use an integrated package like LOTUS 1-2- 3 or VP Planner, which have the capability of giving graphic output. There may be some applications which require complex formatted screens or printouts. In such cases, one method to be explored is "How critical is the format? Can it be modified to make programming efforts simpler?"

For example, the user may have asked page number at the bottom of the report whereas the REPORT FORM of dBASE would print the page number on top of the page. Printing the page number at the bottom of the page may mean program writing instead of using the tools available. If the user wants adhoc queries then a powerful query language like dBASE III would be suitable.

### 1.7.5 Development Effort

The parameter could sometimes be the decider. The main reasons for different languages taking different amounts of effort for the same task are:

(i) features available in the language

(ii) learning time

A simple example of feature availability is that most versions of BASIC do not support indexed files whereas COBOL supports indexed files. The learning time required to gain proficiency could become a critical factor if software has to be developed in a language that the development taken is not familiar with. It is possible that even a particular language offers many good features but might take enough time to learn. It is very possible that programming language is chosen based on what language the people in the organisation are familiar with.

Check Your Progress 2

1. Describe "Benchmark Testing" briefly.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

2. List out various parameters responsible in considering the selection of suitable language in a big organisation.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

## 1.8 SUMMARY

System analyst play very important role in developing information systems that are useful to management and employees in business systems. The systems development life cycle, the set of activities that analysts and designers carry out to develop and implement an information system.

Prototyping is an appropriate development strategy when predicting the user's requirement is not possible. A prototype, a version of an information system having the essential features but not necessarily all details of the user interface or performance efficiency, is developed and put into use.

In hardware/software selection also, system analyst should act very carefully. He must consider all important factors prior to system selection.

Systems analysts rely on a wide variety of tools to fulfill their responsibilities. These tools can be very helpful in the process of system development and implementation.

## 1.9 MODEL ANSWERS

Check Your Progress 1

1. Major tasks of system development are:

   (a) Implementation planning

   (b) Software development phase

   (c) User review

   (d) Equipment acquisition and installation

   (e) Coding, debugging and testing of computer program

   (f) System testing

   (g) Reference manual preparation and training

   (h) User acceptance review

2. Important factors to be considered are:

   (a) Define system capabilities

   (b) Specify the magnitude of the problem

   (c) Assess the competence of the in-house staff

   (d) Consider hardware and software as a package

   (e) Develop a time frame for selection process

   (f) Provide user indoctrination

3. Criteria for software selection are:

| | | | |
|---|---|---|---|
| (a) Reliability | (b) Functionality | (c) Capacity | (d) Flexibility |
| (e) Usability | (f) Security | (g) Performance | (h) Serviceability |
| (i) Ownership | (j) Minimal costs | | |

Check Your Progress 2

1. A benchmark is a simple program specially designed to evaluate the performance of different computers and their software. This is necessary because computers often do not use the same instructions, words of memory or machine cycle to solve a particular problem.

2. Various parameters responsible in considering the selection of suitable language are:

   (a) Volume of data

   (b) Complexity of processing

   (c) Comparability with other systems

   (d) Types of input/output

   (e) Development efforts

# UNIT 2 SYSTEM CONTROL AND QUALITY ASSURANCE

## Structure



## 2.0 INTRODUCTION

The amount and complexity of software developed today stagger the imagination. Software development strategies have not come up with the standard and because of this software products do not meet the application objectives. Subsequently control must be developed to ensure a quality product. Basically quality assurance is the review of software products and its related documentation for completeness, correctness, reliability and maintainability. This unit deals with various issues involved in the quality assurance, testing and system control.

## 2.1 OBJECTIVES

The objectives of this unit are as follows:

- To highlight various issues involved in quality assurance, testing and system control.
- To point out the role of quality assurance in various stages of a Software Development Life Cycle.
- To illustrate the significance of various levels of tests and their different functions.
- To explain the role played by system control in a computer system.

## 2.2 QUALITY ASSURANCE IN SOFTWARE LIFE CYCLE

The software life cycle includes various stages of development and each stage has the goal of quality assurance. Steps taken in this regard are summarised below:

### 2.2.1 Quality Factors Specifications

The goal of this stage is to describe various factors mainly responsible for quality of the proposed system. They are as follows:

(i) **Correctness:** The extent to which a program meets system specifications and user objectives.

(ii) **Reliability:** The degree to which the system performs its intended functions over a time.

(iii) **Efficiency:** Computer resources required by a program to perform a particular function.

(iv) **Usability:** The efforts required to understand and operate a system.

(v) **Maintainability:** The ease with which the program errors are detected and removed.

(vi) **Testability:** The effort required to test a program to ensure its correct performance.

(vii) **Portability:** The ease of transporting a program from one hardware configuration to another.

(viii) **Accuracy:** The required precision in input, editing, computations, and output.

(ix) **Error tolerance:** Error detection and correction versus error avoidance.

(x) **Expandability:** Ease of expanding the existing database.

(xi) **Access control and audit:** Control of access to the system and the extent to which that access can be audited.

(xii) **Communicativeness:** Usefulness and effectiveness of the inputs and outputs of the system.

### 2.2.2 Software Requirements Specifications

The quality assurance goal of this stage is to generate the requirements document that helps in providing technical specifications for developing the software.

### 2.2.3 Software Design Specifications

In this stage, the software design document defines the overall architecture of the software that provides the functions and features given in the software requirements document.

### 2.2.4 Software Testing and Implementation

The quality assurance goal of the testing phase is to ensure that completeness and accuracy of the system and minimize the retesting process. In the implementation phase, the goal is to provide a logical order for the creation of the modules and, in turn, the creation of the system.

### 2.2.5 Maintenance and Support

This phase provides the necessary software development for the system to continue to comply with the original specifications. The quality assurance goal is to develop a procedure for correcting errors and enhancing software.

## 2.3 LEVELS OF QUALITY ASSURANCE

Analysts use three levels of quality assurance: testing, verification with validation and certification.

### 2.3.1 Testing

System testing is quite expensive and time consuming process. The common view of testing held by users is that it is performed to prove that program is error free. But this is quite difficult since the analyst cannot prove that software is free from all sort of errors.

Therefore the most useful and practical approach is with the understanding that testing is the processing of executing a program with the explicit intention of finding errors, that is, making the program fail. A successful test, then, is one that finds an error.

### 2.3.2 Verification with Validation

Like testing, verification is also intended to find errors. It is performed by executing a program in a simulated environment. Validation refers to the process of using software in a live environment to find errors.

When commercial systems are developed with the main aim of distributing them to dealers for sale purposes, they first go through verification, sometimes called alpha testing. The feedback from the validation phase generally brings some changes in the software to deal with errors and failures that are uncovered. Then a set of user sites is selected for putting the system into use on a live basis. These beta test sites use the system in day-to-day activities; they process live transactions and produce normal system output. Validation may continue for several months. During the course of validating the system, failure may occur and the software will be changed. Continued use may bring more failures and the need for still more changes.

### 2.3.3 Certification

The last level of quality assurance is to certify that the software package developed conforms to standards. With a growing demand for purchasing ready-to-use software, importance of certification has increased. A package that is certified goes through a team of computer specialists who test, review and determine how well it meets the user's requirements and vendor's claims. Certification is issued only if the package is successful in all the tests. Certification, however, does not mean that it is the best package to adopt. It only attests that it will perform what the vendor claims.

## 2.4 DESIGN OBJECTIVES: RELIABILITY AND MAINTENANCE

The two operational design objectives continually sought by developers are systems reliability and maintainability. This section will discuss about the importance of these objectives and ways to achieve them.

### 2.4.1 Designing Reliable Systems

A system is said to be reliable if it does not produce dangerous or costly failures during its normal use. The definition recognises that systems may not always be used according to the designer's expectation. There are changes in the ways users use a system and also in business operations. However, there are steps analysts can follow to ensure that the system is reliable at the installation stage and its reliability will continue even after implementation.

There are two levels of reliability. The first level shows that the system is meeting the right requirements. This is possible only if a thorough and effective determination of systems requirements was performed by the analyst. A careful and thorough systems study is required for this aspect of reliability. The second level of systems reliability involves the actual working of the system delivered to the user. At this level, systems reliability is interwoven with software engineering and development.

Reliability has three approaches shown in the Table 1.

**Table 1 : Approaches to Reliability**

| *Sl. No.* | *Approach* | *Description* | *Example* |
|---|---|---|---|
| 1. | Error avoidance | Prevents errors from occuring in the software | It is impossible in large systems. |
| 2. | Error detection and correction | Recognises errors when they are encountered and corrects the error or the effect of the error so tat system does not fail. | Traps and modifies illegal arithmetic steps : Compensates for unexpected data values. |
| 3. | Error Tolerance | Recognises errors when they occur, but enables the system to keep running through degraded performance. | Shuts down part of the system. Does not perform some processing but keeps the system operational. |

### 2.4.2 Designing Maintainable Systems

When the systems are installed, they are normally used for a considerable period. The average life of a system is generally 4 to 6 years, with the oldest applications often in use for over 10 years. However, this period of constant use brings with it the used to contimually maintain the system. When system is fully implemented, analyst must take precautions to ensure that the need for maintenance is controlled through design and testing and the ability to perform it is provided through proper design practices.

**Check Your Progress 1**

1. List out various factors which are responsible for the quality of a system.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

2. Explain briefly the different levels of quality assurance.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

3. Explain briefly the importance of system reliability.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

## 2.5 MAINTENANCE ISSUES

Many studies at the private, University and government level have been conducted to learn about maintenance requirements for information systems. These studies reveal the following facts:

(a) From 60 to 90 per cent of the overall cost of software during the life of a system is spent on maintenance.

(b) Often maintenance is not done very efficiently.

(c) Software demand is growing at a faster rate than supply. Many programmers are spending more time on systems maintenance than on new software development.

Several studies of maintenance have examined the type of tasks performed under maintenance (Lientz and Swanson, 1980). Table 2 summarizes the broad classes of maintenance found in information systems environments.

Table 2: Types of System Maintenance

| Category | Activity | Relative Frequency |
|---|---|---|
| Corrective | Emergency fixes, routine debugging. | 20% |
| Adaptive | Accommodation of changes to data and files and to hardware and system software. | 20% |
| Perfective | User enhancement, improved documentation, recording for computational efficiency. | 60% |

## 2.6 MAINTAINABLE DESIGNS

The keys to reduce the need for maintenance, while making it possible to do essential tasks more efficiently, are as follows:

(a) More accurately defining the user's requirements during systems development.

(b) Making better systems documentation.

(c) Using proper methods of designing processing logic and communicating it to project team members.

(d) Utilising the existing tools and techniques in an effective way.

(e) Managing the systems engineering processin a better and effective way.

Now it is clear that design is both a process and a product. The design practices followed for software has a great effect on the maintainability of a system. Good design practices produce a product that can be maintained in a better way.

## 2.7 TESTING PRACTICE AND PLANS

It should be clear in mind that the philosophy behind testing is to find errors. Test cases are devised with this purpose in mind. A test case is a set of data that the system will process as normal input. However, the data are created with the express intent of determining whether the system will process them correctly. For example, test cases for inventory handling should include situations in which the quantities to be withdrawn from inventory exceed, equal and are less than the actual quantities on hand. Each test case is designed with the intent of finding errors in the way the system will process it. There are two general strategies for testing software: Code testing and Specification testing. In code testing, the analyst develops that cases to execute every instructions and path in a program. Under specification testing, the analyst examines the program specifications and then writes test data to determine how the program operates under specific conditions. Regardless of which strategy the analyst follows, there are preferred practices to ensure that the testing is useful. The levels of tests and types of test data, combined with testing libraries, are important aspects of the actual test process.

## 2.8 LEVELS OF TESTS

Systems are not designed as entire systems nor are they tested as single systems. The analyst must perform both unit and system testing.

### 2.8.1 Unit Testing

In unit testing the analyst tests the programs making up a system. For this reason, unit testing is sometimes called program testing. Unit testing gives stress on the modules independently of one another, to find errors. This helps the tester in detecting errors in coding and logic that are contained within that module alone. The errors resulting from the interaction between modules are initially avoided. For example, a hotel information system consists of modules to handle reservations; guest checkin and checkout; restaurant, room service and miscellaneous charges; convention activities; and accounts receivable billing. For each, it provides the ability to enter, modify or retrieve data and respond to different types of inquiries or print reports. The test cases needed for unit testing should exercise each condition and option.

Unit testing can be performed from the bottom up, starting with smallest and lowest-level modules and proceeding one at a time. For each module in bottom-up testing a short program is used to execute the module and provides the needed data, so that the module is asked to perform the way it will when embedded within the larger system.

### 2.8.2 System Testing

The important and essential part of the system development phase, after designing and developing the software is system testing. We cannot say that every program or system design is perfect and because of lack of communication between the user and the designer, some error is there in the software development. The number and nature of errors in a newly designed system depend on some usual factors like communication between the user and the designer; the programmer's ability to generate a code that reflects exactly the systems specifications and the time frame for the design.

Theoretically, a newly designed system should have all the parts or sub-systems are in working order, but in reality, each sub-system works independently. This is the time to gather all the subsystem into one pool and test the whole system to determine whether it meets the user requirements. This is the last change to detect and correct errors before the system is installed for user acceptance testing. The purpose of system testing is to consider all the likely variations to which it will be subjected and then push the system to its limits.

Testing is an important function to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, the goal will be successfully activated. Another reason for system testing is its utility as a user-oriented vehicle before implementation.

System testing consists of the following five steps:

- Program testing
- String testing
- System testing
- System documentation
- User acceptance testing

**Program Testing**

A program represents the logical elements of a system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. It is the responsibility of a programmer to have an error free program. At the time of testing the system, there exists two types of errors that should be checked. These errors are syntax and logic. A syntax error is a program statement that violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted key words are common syntax errors. These errors are shown through error messages generated by the computer. A logic error, on the other hand, deals with incorrect data fields out of range items, and invalid combinations. Since the logical errors are not detected by compiler, the programmer must examine the output carefully to detect them.

When a program is tested, the actual output is compared with the expected output. When there is a discrepancy, the sequence of the instructions, must be traced to determine the problem. The process is facilitated by breaking the program down into self-contained portions, each of which can be checked at certain key points.

**String Testing**

Programs are invariably related to one another and interact in a total system. Each program is tested to see whether it conforms to related programs in the system. Each part of the system is tested against the entire module with both test and live data before the whole system is ready to be tested.

**System Testing**

System testing is designed to uncover weaknesses that were not found in earlier tests. This includes forced system failure and validation of total system as it will be implemented by its user in the operational environment. Under this testing, generally we take low volumes of transactions based on live data. This volume is increased until the maximum level for each transaction type is reached. The total system is also tested for recovery and fallback after various major failures to ensure that no data are lost during the emergency. All this is done with the old system still in operation. When we see that the proposed system is successful in the test, the old system is discontinued.

**System Documentation**

All design and test documentation should be well prepared and kept in the library for future reference. The library is the central location for maintenance of the new system.

**User Acceptance Testing**

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system's procedures operate to system specifications and that the integrity of important data is maintained. Performance of an acceptance test is actually the user's show. User motivation is very important for the successful performance of the system. After that a comprehensive test report is prepared. This report shows the system's tolerance, performance range, error rate and accuracy.

## 2.9 SPECIAL SYSTEMS TESTS

There are other six tests which fall under special category. They are described below :

(i) **Peak Load Test:** It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand. For example, test the system by activating all terminals at the same time.

(ii) **Storage Testing:** It determines the capacity of the system to store transaction data on a disk or in other files. For example, verify documentation statements that the system will store 10,000 records of 400 bytes length on a single flexible disk.

(iii) **Performance Time Testing:** It determines the length of time system used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response.

(iv) **Recovery Testing:** This testing determines the ability of user to recover data or re-start system after failure. For example, load backup copy of data and resume processing without data or integrity loss.

(v) **Procedure Testing:** It determines the clarity of documentation on operation and use of system by having users do exactly what manuals request. For example, powering down system at the end of week or responding to paper-out light on printer.

(vi) Human Factors Testing: It determines how users will use the system when processing data or preparing reports.

**Check Your Progress 2**

1. What are the different types of system maintenance? Explain them briefly.

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

2. Describe briefly about 'Unit Testing'.

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

3. What do you know about various special systems tests? Explain briefly.

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

## 2.10 DESIGNING TEST DATA

The proper designing of test data is as important as the test itself. If test data as input are not valid or representation of the data to be provided by the user, then the reliability of the output is doubtful. Test data may be live or artificial. The live data is that which is actually extracted from the users' files. After a system is partially constructed, the programmers or analysts ask the users to key in a set of data from their normal activities. It is difficult to obtain live data in sufficient amount to conduct extensive testing.

The artificial test data is created solely for test purposes. Properly created artificial data should provide all combinations of values and formats and make it possible to test all logic and control paths through the program. Unlike live data, which are biased toward typical values, artificial data provide extreme values for testing the limits of the proposed system.

## 2.11 SYSTEM CONTROL

In a computer system, controls essentially mean the extent to which the system is secure against human errors, machine malfunction or deliberate mischief. The amount of control to be applied can vary depending upon the importance, critically and volume of the output. Depending on the nature of the system or the amounts of money at stake, the designer will need to build in different type of controls within system procedures, programs and operations.

### 2.11.1 Objective of System Control

Control mechanisms are designed to achieve the following objectives:

(a) Accuracy: The system should provide information and reports which are accurate in all respect.

(b) Reliability: The system should continue to function as it is designed to function. It should not break down due to malfunction of equipments.

(c) Security: Files and programs in the system must be secured against accidental damage or loss. Procedures must be established to restrict access to data by authorised user only.

(d) Efficiency: Efficiency of the system will essentially mean performing the tasks in the best manner and producing output of highest quality with least efforts.

(e) Audit: System controls should cover the aspect of auditing procedures. Facilities should be designed so that a transaction can be traced from its creation to its final use. This aspect of control assumes added significant in case of on-line systems where written documents may not exist for some transactions.

(f) Adherence to organisation policies: System controls should not conflict with organisation policies and must promote such policies. For example, by ensuring that payroll is produced accurately and by a specified time, the organisation objective of paying all employees in time will be promoted.

It may not always be possible to provide adequate controls in the system to meet the above objectives fully. Sometimes, it may happen that too many controls may adversely affect the performance of the system. On the other hand, lack of adequate control may create chaos and dissatisfaction. It is, therefore, a question of trade-off between the control and their objectives and as stated earlier, the decisions will depend on the nature of the system and criticality of its functions.

### 2.11.2 Types of Control

Two types of controls which affect the operation of a system are:

- External control
- Internal control

External controls to a system, as the name implies are laws, regulations, procedures and policies outside the scope of the system which affect the operations of the system.

Internal controls are basically plans, procedures, guidelines, rules and checks under which the system must function. Much of these internal controls will be specified as a part of system design, but some of these may be internal controls of the organisation - with or without computerisation.

## 2.12 AUDIT TRAIL

In on-line systems, unlike batch environments, there may not be copies of input source documents to fall back on if the system fails during processing. It is also possible for on-line users to sign on to a system, make changes in data already stored in files and sign off again without leaving a visible clue as to what happened. Unless the systems analyst develops an audit trail, no such protection exists in on-line systems.

Audit trail is the path which a transaction traces through a data processing system from source documents to summary reports. In other words, it refers to the facilities or procedures which allow a transaction to be traced through all stages of data processing beginning with its appearance on a source document and ending with its transformation into information on a final output document. The audit trail contains complete details such as reference numbers, dates, names which are recorded in files, ledgers and journals so that trailing of these records to source documents becomes easier. It is generally available in manual accounting system. But in computerised system, it is generally not available unless the system is specially designed to do so.

Audit trails are not primarily for the use of auditors. Rather they are quite important tools that are designed to help the management. The auditors use these tools which management has found necessary for internal purposes.

**Check Your Progress 3**

1. List out the objectives of system control.

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

2. What is the difference between internal and external control?

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

3. Explain briefly about 'Audit Trail'.

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

## 2.13 SUMMARY

It is thus seen that quality assurance, testing and system control play a vital role and each of them contribute to the development of an efficient software. Quality assurance has to be implemented at every stage of the software development life cycle. Implementation at each level thus paves way for a better step in the next stage. Both the system testing and the unit testing should be done so as to avoid the occurrence of error during implementation stage. Using Audit trail all the transactions occurring during a period of time can be tracked.

## 2.14 MODEL ANSWERS

**Check Your Progress 1**

1. Various factors are as follows:

   (i) Correctness (ii) Reliability (iii) Efficiency (iv) Usability
   (v) Maintainability (vi) Testability (vii) Portability (viii) Accuracy
   (ix) Error tolerane (x) Expandability (xi) Access control and audit
   (xii) Communicativeness

2. Different levels of quality assurance are;

   (i) Testing

   (ii) Verification with validation

   (iii) Certification

3. A system is said to be reliable if it does not produce dangerous or costly failures during its normal use. There are two levels of reliability. The first level shows that system is meeting the right requirements. This is possible only if a thorough and effective determination of system requirement, was properly studied by the analyst. The second level of system reliability involves the actual working of the system delivered to the user. At this level, system's reliability is interwoven with software engineering and development.

**Check Your Progress 2**

1. Different types of system maintenance are:

   (a) Corrective

   (b) Adaptive

   (c) Perfective

2. In unit testing, the analyst tests the programs making up a system. Unit testing gives stress on the modules independently of one another, to find errors. This helps the tester in detecting errors in coding and logic that are contained within that module alone. The errors resulting from the interaction between modules are initially avoided.

3. Various special system tests are:

| | | |
|---|---|---|
| (i) Peak load test | (ii) Storage testing | (iii) Performance time testing |
| (iv) Recovery testing | (v) Procedure testing | (vi) Human factors testing |

**Check Your Progress 3**

1. Objective of system control are as under:

   (i) Accuracy (ii) Reliability (iii) Security (iv) Efficiency
   (v) Audit (vi) Adherence to organisation policies

2. Internal controls: They are basically plans, procedures, guidelines, rules and checks under which the system must function.

   External controls: They are mainly laws, regulations, procedures and policies outside the scope of the system which affect the operations of the system.

3. Audit trail is the name given to the facility to trace individual transactions through a system from source to completion.

# UNIT 3 DOCUMENTATION

## Structure



## 3.0 INTRODUCTION

A system cannot be completely effective unless it is adequately documented. It should be documented as it is being created. That is, at various stages of the system development, status reports should be prepared for those management personnel for whom the system is being designed. Such reports could include flowcharts, decision tables, output forms and other documents thus far developed. It also includes various problems encountered, suggested solutions and resulting schedule revisions. In this way, management remains fully aware of system's progress so that they can offer criticisms or suggest change while it is still economically and physically possible to make these changes without it being necessary to revise the entire system. These progress reports provide an excellent basis on which to build additional documentation.

Instructions and narrative descriptions must be prepared for every phase and part of the system, including system logic, tunings, user instructions, guidelines for operations staff in the data processing centre and instructions relating to transmission of data and results. Much of this can be incorporated into procedures manual. This manual stipulates the relationship between personnel in the application areas affected by the system and the data processing centre. It should relate in details, exactly what procedures must be employed by the user to operate the system efficiently and effectively.

## 3.1 OBJECTIVES

The following are the objectives of this unit

- To explain the characteristics of good documentation and its various types
- The tools needed for documentation
- The guidelines/format to be followed for preparing good documentation package.

## 3.2 CHARACTERISTICS OF A GOOD DOCUMENTATION

Documentation is considered to be good if it has the following qualities:

(a) Availability: It should be accessible to those for whom it is intended.

(b) Objectivity: It must be clearly defined in a language that is easily understood.

(c) Cross-referable: It should be possible to refer to other documents.

(d) Easy to maintain: When the system gets modified, it should be easy to update the documentation.

(e) Completeness: It should contain everything needed, so that those who have gone through it carefully can understand the system.

## 3.3 TYPES OF DOCUMENTATION

These are five major types of documentation. They are:

(i) Program Documentation

(ii) Operation Documentation

(iii) User Documentation

(iv) Management Documentation

(v) Systems Documentation

### 3.3.1 Program Documentation

Many companies discuss about programming documentation but fail to provide it adequately. Before a program is developed, the systems analyst should provide the programmer with the required documentation. The logic in some programs is best described by a flowchart. Sometimes, decision tables are most appropriate for explaining the logic of a program. Programmers should insist on proper documentation before starting a job.

Four items constitute normal documentation required for each program.

- Copying in final form of all input/output documents affecting the program.
- Statement of standards for coding structures and input/output layouts.
- Clarification of the program's interface with other related programs.
- General flowchart or decision table.

The programmer's responsibility in documentation is to provide information to enable future programmers to make necessary changes. Personnel turnover is normal feature in any business, and turnover is particularly high among programmers. A company can never think that a programmer assigned to a specific program will be available in two years, when some modifications to that program are required. For continuity of information a company must insist on complete and meaningful documentation. Typically a documentation folder is provided for each program which contains all the input/output forms associated with the program, a detailed flowchart or decision table for the program use a set of operator and user instructions.

Maintaining this type of documentation is costly and time consuming, for, programmers do not take interest in spending time for this type of work. Routine changes occur frequently in a program and all changes must be covered in the documentation folder. But the very changes which require the updating of existing documentation are the reasons for maintaining accurate documentation.

### 3.3.2 Operations Documentation

A well designed system may run for a long time with little or no assistance from the systems department. This can happen only when the system has been documented in a proper way. For smooth running of the system, the console operator must have complete knowledge about the job. Providing the computer centre with a set of operating instructions will not serve the purpose. The instructions must be in a form readily accessible to the console operator and written in simple and understandable style. A systems analyst must thoroughly discuss all the requirements of new jobs with the operations staff before the job can be properly transferred.

The run book is traditional in computer centres. It is a collection of operator instructions for each program at an installation and typically contains:

(i) Narrative, describing the run

(ii) Listing of the programmed error conditions

(iii) Detailed information for running the job, including:

- input/output forms to be used
- anticipated problem areas and how to handle them
- detailed description of file assignment of each input/output device
- disposition of data files after completing the job
- general block diagram of the programming logic
- restart procedures

The run book generally takes the form of a loose leaf notebook because of the ease of substituting sheets as programs change. It should be kept in mind that an operator in a multiprogramming environment must monitor many programs simultaneously. Instructions must be simple and complete enough for executing the job correctly.

### 3.3.3 User Documentation

Systems users require proper documentation to prepare a developing system and to smoothly carry out existing ones. To meet this requirement, each system should have a manual that spells everything the users must know to do their job correctly. Users require two general types of information; complete details to handle each case the system processes, and overall picture of the system so that they can see their role in the total operation of the company.

The manual should supply the following information.

- General flowchart of the system
- Assignment of responsibility for specific tasks
- Standards for work flow, including target dates and deadlines for specific tasks
- Simple input and output documents
- Detailed procedures
- Anticipated exceptions and instructions on how to handle them
- Accuracy standards for data in the system

The systems department must write a thoroughly detailed narrative of each system, including the proper handling of routine cases, as well as exception handling. A staff member in the user department must have an authority to consult when faced with a case not handled before. Properly prepared manual which is always available can provide the information needed by the user. Supervising staff in user areas must understand the overall picture in each system just as staff members must understand the details of their function. This requires documentation, in the form of charts, graphs and illustrations, so that the supervising staff have a clear grasp of their department's role in the total system.

### 3.3.4 Management Documentation

The documentation required by corporate management differs quite a lot from that required by users. The systems designer must know the requirements of the management and provide documentation to enable management to perform three functions:

(i) Evaluate progress on systems development

(ii) Monitor existing systems

(iii) Understand the objectives and methods of new and existing systems

Management is primarily interested to know in general the system's overall objectives and basic operations. A brief manual highlighting the key steps in each system may be prepared for management. Good managers have an exceptional ability to get to the root of a system and, their experience should enable them to retrieve information from a systems summary or chart which may not be apparent to the systems analyst.

### 3.3.5 Systems Documentation

Each phase in the systems development cycle is accompanied by appropriate documentation. The systems request, even if it is initially mark verbally, eventually must be written. It is desirable for the client and a systems analyst to work jointly in writing the request since each can contribute knowledge the other does not have. The written system's request is merely a statement of the user's problem.

In documenting the results of its deliberations, the selection committee must specify the following:

(a) The objectives of the impending feasibility study

(b) The extent of the authority of the feasibility team

(c) The individual or group responsible for completing the study

A feasibility report is probably the most important form of documentation in the system development cycle. It accomplishes the following two purposes:

(i) It defines the objectives of the proposed system's change in reasonable detail after a sufficiently detailed study.

(ii) It gives a plan to attain these objectives.

The documentation of this plan must be thorough enough so that system designers could produce a complete and effective system. At various points during systems design, the designing team produces the following additional forms of documentation:

(i) File Specification: detailed definitions of each file in the system, best done in graphic form.

(ii) Transaction Specifications: detailed descriptions of each type of input in the system, including a layout of each transaction and a narrative description of how it is used.

(iii) Output Specifications: detailed descriptions of all output anticipated from the system.

Documentation also includes plans to test the system and convert from the old to the new one. The systems analyst must also provide a plan to train the personnel affected by the changes.

During the life cycle of the completed system, the system itself must provide documentation of how well it is operating and consequently should be designed to yield data about itself as a normal by-product.

Check Your Progress 1

1. Describe the characteristics of a good documentation.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

2. List out various types of documentation.

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

What is user documentation?

..........................................................................................................................

..........................................................................................................................

..........................................................................................................................

..........................................................................................................................

..........................................................................................................................

## 3.4 SOFTWARE DESIGN AND DOCUMENTATION TOOLS

Well designed, modular software is more likely to meet the maintenance, reliability and testing requirement. Three specific tools are described below :

- Structured flowchart
- HIPO diagrams
- Warnier/Orr diagrams

### 3.4.1 Structured Flowchart

Structured flowcharts, also called Nassi-Schneiderman charts, are graphic tools that force the designer to structure software in modular as well as top-down form. They provide a proper structure that can be retained by the programmer for developing the application software. The programmer should be expert in using the structured flowcharts.

The basic elements are used in developing structured flowcharts are :

- Process
- Decision
- Iteration

**Process:** Simple processes or steps in a program are shown by a rectangular box, the process symbol. This symbol represents initialisation of values, input and output operations and calls to execute other procedures.

**Decision:** The decision symbol represents alternative conditions that can occur and that the program must have a manner of handling. The decision symbol may show actions for more than two alternatives at the same time.

**Iteration:** The iteration symbol represents looping and repetition of operations while a certain condition exists or until a condition exists.

The structured flowcharts use no arrows or continuations on separate pages. Each structured flowchart is shown on a single sheet of paper. When designing a structured flowchart, the systems analyst specifies the logic in a top down fashion. The first consideration in a process is the top element. The second in sequence is next one shown and so forth. Similarly, there is a single exist from the process.

### 3.4.2 HIPO Diagram

It is another tool commonly used for developing systems software. It is an acronym for Hierarchical Input Process Output. This method was originally developed to provide documentational assistance for programmers/analysts.

The major concept upon which HIPO is based is the highly structured modular design. The HIPO documentation uses a structure that is similar to an organisation chart. This type of structure allows the enforcement of major principles to HIPO, a top-to-bottom approach to design. The emphasis is made on forcing the flow of data down through the system, not in the opposite direction.

The main idea behind the top-to-bottom approach is the elimination of "output-oriented" systems solutions. An output- oriented system is concerned with providing output and does not bother about the sound principles of system design. In essence, an output-oriented system often gets the job done without delay.

Unfortunately, many data processing organisation try to employ this type of rationale in their system designs. Output-oriented systems are often fragmentary, with large gaps evident in the logic and flow of data throughout the system. Programs written for this type of system often duplicate each other in part. The net effect is that more programming efforts are required, with a resultant, loss of manpower and time.

The HIPO concept, with its highly ordered structure and top- to-bottom approach, tries to eliminate piecemeal system design. A view of general HIPO structure is shown in Figure 1.

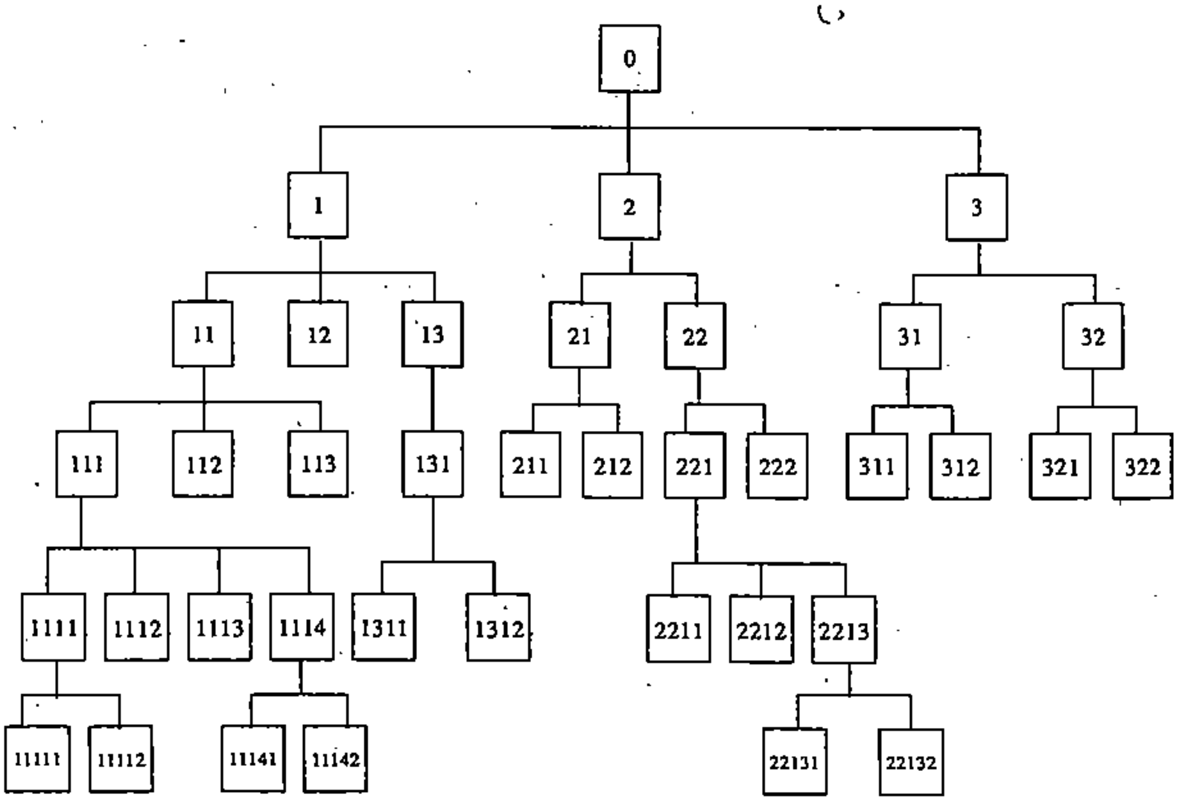



**Figure 1: Structure of HIPO chart identifying all of the components within the chart by number at each of the sub-level.**

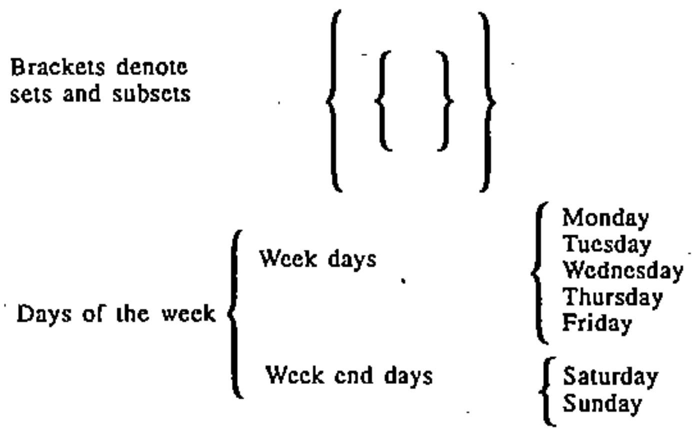
As you can observe that HIPO structure is quite similar to that of a manager's organisational chart. The numbers shown in various boxes of HIPO chart provide a means of identifying each of this sub-levels and component blocks on the chart. The rationale of subdividing and identifying the component blocks within a HIPO design is extremely important. Applying this concept, the analyst is capable of defining and completely laying out the overall structure of the entire system under study. The HIPO approach is mainly designed to accommodate the development of a system.

### 3.4.3 Warnier/Orr Diagram

Warnier/Orr diagrams, also known as logical construction of programs/logical construction of systems are powerful tools aimed at designing of program structures by identifying the output and processing results and then working backwards to determine the steps and combinations of input needed to produce them. The simple graphic methods used in Warnier/Orr diagrams make the levels in the system evident and movement of the data between them vivid.

Warnier/Orr diagrams clearly show the various processes and sequences in which they are performed. Each process is defined in a hierarchical way. At each level, the process is shown in a bracket that groups its components (figure 2). Since a process consists of different subprocesses, a Warnier/Orr diagram employs a set of brackets to indicate each level of system clearly.

Brackets denote set and subsets




The set of days in the week has week days and week end as subsets.

**Figure 2: Set Notation used in Warnier/Orr Diagrams**

Warnier/Orr diagrams are very powerful design tools and offer some distinct advantages to systems experts. They are quite simple in appearance and easy to understand.

**Check Your Progress 2**

List out various software design and documentation tools.

2. What do you mean by HIPO diagram? Explain briefly.

3. Explain briefly the Warnier/Orr Diagram.

## 3.5 NEED FOR DOCUMENTATION

Preparation of documentation is quite important as it depicts what the system is supposed to be and how it should perform its functions. It illustrates both technically and economically how a system would better serve the objectives and goals of the company. Documentation improves overall operation in addition to management and audit control.

It also serves the following purposes:

(i) Reviews the progress or development of an application software.

(ii) Communicates facts about system to users.

(iii) Communicates between personnel working on a development project.

(iv) Provides necessary guidelines to allow correction or revision of a system or its computer programs.

(v) Provides operating instruction to users and operating staff.

(vi) Assists in the reconstruction of the system in case it is destroyed.

(vii) It helps the management to determine if the new design achieves the objectives of the company within the established constraints and if it is justifiable from a cost standpoint.

(viii) Documentation serves as a focal point from which the analysts' design can be assessed and as a standard to be utilised as a reference once the system is implemented.

## 3.6 GUIDELINES/FORMAT FOR PREPARING DOCUMENTATION PACKAGE

There are, as yet, no universal documentation standards, since systems vary greatly in form, contents and requirements. The format of each documentation package will be based on the following points:

(i) **Characteristics of system:** Some designs require descriptive while others can be explained with the help of diagrams.

(ii) **Management's attitude toward documentation:** The analyst must prepare the documentation package within the limitations established by the management.

(iii) **Equipment restraints:** A company with large and integrated computer system having teleprocessing facilities will require more formalised and technical documentation than a company with a more conservative and small computer system.

## 3.7 ELEMENTS THAT COMPRISE A DOCUMENTATION PACKAGE

A documentation package consists of the following elements :

- Cover letter
- Table of contents
- Narrative
- Flow charts
- File specification
- Program specification
- Cost of the proposed system and of its alternatives
- Test brochures

(i) **Cover letter:** The cover letter is a correspondence primarily to management, that describes the benefits of the new design and that generally helps in selling the system. It should be kept in mind that unless the documentation is approved, the new system will never be implemented and the anplysts' work will be of no use. Thus the analyst must try to convince the management that the new design presented is feasible and appropriate to satisfy the objectives of the company. The cover letter should describe the purpose and function of the new system clearly. It should be written in concise language to facilitate executive understanding, without requiring complete familiarity with the intricacies of the system.

(ii) **Table of Contents:** The inclusion of a table of contents is an absolute necessity. Pages in the documentation package must be numbered and cross referenced in this table of contents.

(iii) **Narrative:** With the narrative, we begin the detailed formulation of the new system.

(iv) **Flowcharts:** Each subsystem within the analyst's formal design should be explained with the help of flowchart.

(v) **File Specification:** Each file within the formal design must be described with regard to:

- Purpose
- Programs that will use the file
- Volume
- Frequency of use
- Source from which the file is obtained
- Description of fields
- Layout and samples

(vi) **Program Specification:** At this point, the analyst must segment the new design so that each unit will have separate program, assuming that the design is itself approved by the management.

(vii) **Cost of the proposed system and of its alternatives:** The details of cost must be shown as part of documentation.

(viii) **Test Brochures:** The analyst should describe the operations and procedures (including test data) that will be employed to test the new system, once it is approved.

Check Your Progress 3

1. Why is documentation needed ?

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

2. What are the elements that comprise a documentation package?

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

.................................................................................................................

## 3.8 SUMMARY

The unit stresses the need for effective documentation at all stages of the software development. The various type of documents discussed above are helpful to various personnel in their respective functional areas. It also stresses the need to use various tools for designing and documenting a software package.

## 3.9 MODEL ANSWERS

**Check Your Progress 1**

1. Various characteristics of a good documentation are:
   - (i) Availability
   - (ii) Objectivity
   - (iii) Cross-referable
   - (iv) Easy to maintain
   - (v) Completeness
2. Major types of documentation are:
   - (i) Program documentation
   - (ii) Operation documentation
   - (iii) User documentation
   - (iv) Management documentation
   - (v) Systems documentation
3. Systems users require proper documentation to prepare or developing system and to smoothly carry out existing ones. To meet this requirement, each system should have a manual that spells everything the users must know to do their job correctly.

**Check Your Progress 2**

1. Various software design and documentation tools are:
   - (i) Structured flowchart
   - (ii) HIPO diagrams
   - (iii) Warnier/Orr diagrams
2. It is an acronym for Hierarchical Input Process Output. It is quite important tool commonly used for developing systems software. This method was originally developed to provide documentational assistance for programmers/analysts.
3. Warnier/Orr diagrams are powerful tools aimed at designing of program structure by identifying the output and processing results and then working backwards to determine the steps and combinations of input needed to produce them. They are important design tools and offer some distinct advantages to systems experts.

**Check Your Progress 3**

1. Preparation of documentation is quite important as it depicts what the system is supposed to be and how it should perform its functions. It illustrates both technically and economically how a system would better serve the objectives and goals of this organisation.
2. A documentation package contains the following elements:

| | | | | | |
|---|---|---|---|---|---|
| (i) | Cover letter | (ii) | Table of contents | (iii) | Narration |
| (iv) | Flow charts | (v) | File specification | (vi) | Program specification |
| (vii) | Cost of proposed system and of its alternatives | | | (viii) | Test brochures |

# UNIT 4 SYSTEM IMPLEMENTATION

## Structure



## 4.0 INTRODUCTION

A crucial phase in the system life cycle is the successful implementation of the new system design. Implementation includes all those activities that take place to convert from the old system to the new one. The new system may be completely new, replacing an existing manual or automated system or it may be major modification to an existing system. In either case, proper implementation becomes necessary so that a reliable system based on the requirements of the organisation can be provided. Successful implementation may not guarantee improvement in the organisation using the new system, but improper installation will prevent it. It has been observed that even the best system cannot show good result if the analysts managing the implementation do not attend to every important details. This is an area where the systems analysts need to work with utmost care.

This unit discusses the three aspects of implementation:

- Training personnel
- conversion procedures
- Post-implementation review

In each area, the particular elements of that aspect are discussed, along with the methods of handling each aspect efficiently and effectively.

## 4.1 OBJECTIVES

After completing this unit, you should be able to :

- understand the importance oftraining of personnel involved with system.
- get familarity with various training methods conversion and operation plans.
- explain post implementaton reveiw.
- highlights various issues involved in system maintenance and desgning of computer contract.
- get proper understanding about various factors for hardware/software acquistion and vendor's selection.
- understand the importance of service bureau and financing use of computers.

## 4.2 TRAINING OF PERSONNEL INVOLVED WITH SYSTEM

Even well designed system can succeed or fail because of the way they are operated and used. Therefore, the quality of training received by the personnel involved with the system in various capacities helps or hinders and may even prevent the successful implementation of management information system. Those who are directly or indirectly related with the system development work must know in detail what their roles will be, how they can make efficient use of the system and what the system will or will not do for them. Both systems operators and users need training.

### 4.2.1 Systems Operators Training

Running of the system successfully depend on the personnel working in the computer centre. They are responsible for providing the necessary support. Their training must ensure that they are able to handle all possible operations, both routine and extra-ordinary in nature.

If the system calls for the installation of new equipment, such as a new computer system, special terminals or different data entry machines, the operators' training should include such fundamentals as how to turn the equipment on and use it, how to power off and a knowledge of what constitutes normal operation. The operators should also be trained on different type of malfunctioning, how to recognise them and what steps should be taken whenever they arise. As part of their training, operators should be given both a troubleshooting list that identifies possible problems and remedies for them, as well as the names and telephone numbers of individuals to contact when unexpected or unusual problems arise. Training also involves familiarisation with run procedures, which involves working through the sequence of activities needed to use a new system on an ongoing basis.

### 4.2.2 User Training

User may be trained on use of equipment, particularly in the case where, for example, a micro-computer is in use and the individual involved is both operator and user. In such cases, user must be given training on how to operate the system also. Questions that may be trivial to the analyst, such as how to turn on a terminal, how to insert a diskette into a micro-computer, or when it is safe to turn off equipment without danger of data loss, are significant problems to new users who are not familiar with computers.

In most of the cases, user training deals with the operation of the system itself, with proper attention given to data handling techniques. It is imperative that users be properly trained in methods of entering transactions, editing data, formulating inquiries, deleting and inserting of records. No training is complete without familiarising users with simple systems maintenance activities. Weakness in any aspect of training may lead to awkward situations that create user frustration and errors.

## 4.3 TRAINING METHODS

Training of operators and users can be organised in several different ways.Most important are :

(i) Vendor and in-service training

(ii) In-house training

### 4.3.1 Vendor and In-service Training

Often the best source of training on equipment is the vendor who supplies the equipment. Most vendors offer extensive educational programs as part of their services. For example, IBM offers complimentary two or three days short-term courses to purchasers of many of their mini and mainframes. The courses, offered by experienced trainers and sales personnel, cover all aspects of using the equipment, from how to turn it on and off, to the storage and retrieval of data. One session is kept for hands-on training also so that the participants can freely use the system in the presence of the trainers.

If special software such as teleprocessing package or database management system is being installed, sending personnel to off-site short term courses providing in-depth training is preferable to in-service training. These courses, which are generally provided by charging a fee, are presented to personnel from many organisations that are acquiring or using the same system. The benefit of sharing questions, problems and experiences with persons from other companies is substantial.

### 4.3.2 In-house Training

The main advantage of offering in-house training is that instruction can be tailored according to the requirements of the organisation. Often the vendors negotiate fees and charges that are more economical so that company can involve more personnel in the training program than is possible when travel is required. However, the disadvantage of distracting telephone calls, business emergencies and other interruptions must not be overlooked.

The other common approach is to evaluate by taking case- study example that contains all sort of frequently encountered situations that system is able to handle. Then the user must use the system to handle the actual situations; that is, enter data as required, process the data and prepare the desired reports.

Although high-quality training is an essential step in systems implementation, yet it is not sufficient by itself.

## 4.4 CONVERSION METHODS

Conversion is the process of changing from the old system to the new one. It must be properly planned and executed. Four methods are common in use. They are: parallel systems, direct conversion, pilot system and systems phase-in. Each method should be considered in the light of the opportunities that it offers and problems that it may create. However, it may be possible that sometimes, we may be forced to apply one method over others, even though other methods may be more beneficial. In general, systems conversion should be accomplished in shortest possible time. Long conversion periods create problems for all persons involved including both analysts and users.

### 4.4.1 Parallel Systems

The most secure method of converting from an old to new system is to run both systems in parallel. Under this approach, users continue to operate the old system in the usual manner but they also start using the new system. This method is the safest one because it ensures that in case of any problems in using the new system, the organisation can still fall back to the old system without loss of time and money.

The disadvantages of the parallel systems approach are:

- It doubles operating costs
- The new system may not get fair trial.

### 4.4.2 Direct Conversion

This method converts from the old to the new system abruptly, sometimes over a weekend or even overnight. The old system is used until a planned conversion day, when it is replaced by the new system. There are no parallel activities. The organisation relies fully on the new system. The main disadvantages of this approach are: no other system to fall back on, if difficulties arise with new system. Secondly, wise and careful planning is required.

### 4.4.3 Pilot System

Pilot approach is often preferred in the case of the new system which involves new techniques or some drastic changes in organisation performance. In this method, a working version of the system is implemented in one part of the organisation, such as a single work area or department. The users in this area are aware that they are piloting a new system and that changes can be made to improve the system. Based on the feedback, changes are made and the system is installed in the remaining departments of the organisation, either all at once (direct conversion method) or gradually (phase-in method). This approach provides experience and live test before implementation.

### 4.4.4 Phase-in Method

This method is used when it is not possible to install a new system throughout an organisation all at once. The conversion of files, training of personnel or arrival of equipment may force the staging of the implementation over a period of time, ranging from weeks to months. It allows some users to take advantage of the new system early. Also it allows training and installation without unnecessary use of resources.

## 4.5 CONVERSION AND OPERATION PLANS

After the system conversion is completed using any one of the methods mentioned above, the conversion plan starts. In the conversion plan, all the activities that must occur to implement the new system are properly defined and put it into operation. It identifies the persons responsible for each activity and includes a time schedule for each activity.

During the pre-implementation stages, when the conversion is being planned, analysts should prepare a list of all tasks, including the following:

- List all files of conversion
- Identify all data required to build new files during conversion
- List all new documents and procedures used during conversion
- Identify all controls to be made during conversion
- Assign responsibility for each task
- Verify conversion schedules

The conversion plan should anticipate possible problems and methods for controlling them. The missing documents, mixed data formats between current and new files, errors in data translation and situations that were overlooked during systems development are the common problems. The conversion manager must guard against the omission of steps in the conversion. This manager is also responsible for reviewing conversion plans, verifying the delivery of equipment, software and preparing the site.

### 4.5.1 Site Preparation

A major aspect of conversion is the preparation of the systems site. Preparation activities include electrical and air- conditioning preparation, site layout and installation of the equipment. It is the best to have the site preparation completed prior to the arrival of the equipment, since vendors are not in favour of delivering the system if the construction work is going on.

If the system is micro-computer, little site preparation work is needed. However, the electric lines should be checked to ensure that they are free of static or power fluctuations. It is desirable to install a "clean" line that is not shared by any other equipment. Static electricity is very harmful for computers. Carpet should be avoided in the computer room. If carpet is necessary, it should be the anti-static type that will not allow static build-up.

The site layout should allow sufficient space for moving equipment in and setting it up for normal operation. Vendors will provide clearance requirements for performing service and maintenance and for air circulation. These requirements must be strictly adhered to or warranties can be voided and maintenance discontinued until specifications are met.

### 4.5.2 File and Data Conversion

Data and file preparation consumes a large proportion of conversion time. Not only m st the data be converted to a format acceptable in the new system, but analysts must ensure that this is due without loss of detail or accuracy. By using record counts, financial controls and hash totals, analysts are able to detect correct problems quickly, before they get out of control, even if the conversion involves data transmission.

## 4.6 POST IMPLEMENTATION REVIEW

After the system is implemented and conversion is complete, a review should be conducted to determine whether the system is meeting expectations and where improvements are needed. A post implementation review measures the systems' performance against pre-defined requirements. It determines how well the system continues to meet performance specifications. It also provides information to determine whether major re-design or modification is required.

A post-implementation review is an evaluation of a system in terms of the extent to which the system accomplishes stated objectives and actual project costs exceed initial estimates. It is usually a review of major problems that need converting and those that surfaced during the Implementation phase.

The post implementation study begins with the review team, which gathers and reviews requests for evaluation. Unexpected change in the system that affects the user or system performance is a primary factor that prompts system review. Once request is filed, the user is asked how well the system is functioning to specifications or how well the measured benefits have been realised. Suggestions regarding changes and improvements are also asked for.

**Check Your Progress 1**

1. List out the various training methods for imparting training to operators and users.

   ......................................................................................................

   ......................................................................................................

   ......................................................................................................

   ......................................................................................................

   ......................................................................................................

2. List out the four methods of system convesion.

   ......................................................................................................

   ......................................................................................................

   ......................................................................................................

..............................................................................................................

..............................................................................................................

3. Explain the importance of post implementation review briefly.

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

..............................................................................................................

## 4.7 REVIEW PLAN

The review team prepares a formal review plan around the objectives of the review, the type of evaluation to be carried out and the time schedule required. An overall plan covers the following areas:

(i) **Administrative Plan:** Review area objectives, operating costs, actual operating performance and benefits.

(ii) **Personnel requirements Plan:** Review performance objectives and training performance to data.

(iii) **Hardware Plan:** Review performance specifications.

(iv) **Documentation Review Plan:** Review the system development effort.

The review not only assesses how well the current system is designed and implemented, but also is a valuable source of information that can be applied to the next systems project.

The review team prepares a formal review plan around the objectives of the review, the type of evaluation to be carried out and the time schedule required. The review plan cover the following areas:

(i) Administrative Plan: The following two activities are reviewed under this plan:

  (a) User Objective: This is an extremely critical area since it may be possible that over a period of time either the system does not meet the initial objectives of the user or the user objectives get changed as a result of changes in the overall objectives of the organisation. The results of the evaluation are documented for future reference.

  (b) Operating Costs and Benefits: Under the administration plan, current budget designed to manipulate the costs and savings of the system is closely reviewed.

(ii) Personnel Requirement Plan: Under this plan, all activities involving system personnel and staff members associated with the system are evaluated. After the plan is developed, the review group evaluates:

  (a) Personnel performance objectives compared with current performance levels

  (b) Training performance through testing, conducting interviews and other data gathering techniques.

(iii) Hardware Plan: The hardware of the new system is also reviewed including terminals, CRT and communication network. The main target is a comparison of current performance specifications with design specifications. It also points out necessary modification to be made.

(iv) Documentation Review Plan: The reason for review plan is to evaluate the accuracy and completeness of the documentation compiled to date and to its conformity with documentation standards established earlier.

## 4.8 SYSTEM MAINTENANCE

The last part of the system development life cycle is system maintenance which is actually the implementation of the post- implementation review plan. When systems are installed, they are generally used for long periods. The average life of a system is 4 to 6 years, with oldest application often in use for over 10 years. However, this period of use brings with it the need to continually maintain the system. Programmers/Analyst spend sufficient time for maintaining programs. The study on the maintenance requirement for the information system revealed that

(a) 60-90 per cent of the overall cost of software during the life of a system is spent on maintenance.

(b) In documented cases, the cost of maintenance, when measured on the basis of writing each instruction in coding form, is more than 50 times the cost of developing a system.

(c) The software demand is increasing at faster rate than supply. Many programmers are devoting more time on systems maintenance than on new software development. There is a backlog of new development work.

The maintenance can be classified as corrective, adaption or perfective. Corrective maintenance means repairing, processing or performance failures or making alterations because of previously ill-defined problems.

Adaption maintenance means changing the program functions. Enhancing the performance or modifying the programs according to user's additional or changing needs are included in perfective maintenance. The greatest amount of maintenance work is for user enhancement and improved documentation of the system for better efficiency. More time and money are spent on perfective than on corrective and adaptive maintenance together.

Maintenance covers a wide range of activities including correcting coding and design errors, updating documentation and test data and upgrading user support. Many activities classified as maintenance actually fall under enhancements. Maintenance means restoring something to its original position. Unlike hardware, software does not wear out; it is corrected. In contrast, enhancement means adding, modifying or re-developing the code to support changes in the specifications. It is to keep with changing user needs and the operational environment.

The keys to reduce the need for maintenance while making it possible to carry on with essential tasks more efficiently are as follows:

(a) More accurately defining the user's requirement during systems development.

(b) Preparation of system documentation in a better way.

(c) Using more effective ways for designing processing logic and communicating it to project team members.

(d) Making better use of existing tools and techniques.

(e) Managing the systems engineering process effectively.

An addition factor in the success of the maintenance programmer is the work environment. Maintenance programmers have generally been paid less amount and receives less recognition than other programmers. Little attention has been paid to their training and career plans within the MIS function. Maintenance demands more orientation and training then any other programming activities, especially for entry-level programmers. The environment must recognize the needs of the maintenance programmer for tools, methods and training.

## 4.9 DRAWING UP COMPUTER CONTRACT

Generally it is observed that vendors have their own standard form of contract which is prepared keeping into mind the interest of the vendor only. But during final discussion with the vendor, it can be negotiated to add, delete or modifying certain clauses so that terms and conditions of the contract become reasonably equitable for both the sides.

The buyer just look into the following provisions in the contract to safeguard his interest.

### 4.9.1 Respective Responsibilities of Vendors and Buyers

The responsibilities and remedies available to the two sides in the event of non or faulty performance of the system should be clearly defined in the contract. The remedies include special remedies, damages (actual, consequential and liquidated) and specific performance. 'special remedies' may be invoked when the vendor fills to give the delivery on time or is found deficient at the time of carrying out acceptance tests. Alternatives to special remedies are claims for actual consequential and liquidated damages. 'Actual damages' compensate a party for the advantages lost in the actual bargain. 'consequential damages' compensate the parties in respect of all foreseeable losses because of breach of contract. 'Liquidated damages' fix a certain amount in the event of breach of contract condition. 'Specific performance' clauses are attracted when a system fails the acceptance tests because of memory storage. In this situation, buyer can ask for additional memory at nil or nominal cost within a specified time.

### 4.9.2 Documentation

The customer should have the right to stop payment if proper documentation required for the effective running of the system is not provided by the vendor.

### 4.9.3 Hardware

It is necessary that each component of the contracted hardware is identified and its performance criteria is clearly stated in normal operating condition.

### 4.9.4 Delivery and Acceptance

It should be clearly described about the delivery schedule and the acceptance test/standards to be met by the computer during normal operating conditions throughout a specific period of use.

### 4.9.5 Right of Use of Equipment from Other Vendors

The customer should have liberty to link peripherals from other manufacturers to the CPU. The vendor must have protection from any damage or extra costs resulting therefrom.

### 4.9.6 Warranties

Warranties should be included in the contract.

### 4.9.7 Guarantees

For assuring reliability of the system, guarantees should encompass the following points:

- minimum hours of usable time per day.
- mean-time-between failures (MTBF).
- maximum time to repair.

### 4.9.8 Payments

It should be clearly explained in the contract whether the rental/lease payments made by the customer are in respect of number of hours actually used per month/shift or they are to make at a flat monthly rate. In lease contracts, provision of buying the equipment at certain stipulated prices should also be kept.

The contract should provide the facility to protect the already negotiated prices. Also, if the vendor insists on escalation clauses, the customer should have the right to terminate the contract and recover compensation for costs incurred on preparation.

### 4.9.9 Bankruptcy

The contract should provide explicit protection to the customer in the event of the vendor becoming bankrupt.

It is fact that all the areas of problems and disputes between vendors and customers of computer hardware cannot be covered in the contract. Still, the aforementioned check-lists try to cover the problems areas usually encountered.

## 4.10 HARDWARE ACQUISITIONS

Once a decision has been reached to install an in-house computing device, the next step is to prepare a list of specifications of the proposed system so that suitable vendors would be invited for meeting the specific requirements.

The tender specifications are prepared as per norms of approved feasibility report. Main technical parameters of the various units of the required hardware objectives of the project and implementation schedule are also included in the tender specifications. Vendors may also be asked to quote separately in respect of 'leasing' and 'buying' options. In addition to this, the vendors may be required to furnish the details of the infrastructure which the customer will have to arrange and the likely cost thereof.

### 4.10.1 Tender Evaluations

It is often seen that requirements as indicated by the customer do not match with the offer made by individual vendors where the specification given by the vendors are far below the essential requirements of the customers, such offers may be rejected straightway from the purview of short listing. Marginal shortfalls may be considered on merits. However, in case of additional features in the offers which could be categorised as 'desirable', it becomes necessary to assign appropriate weights to such features, in order to bring all the bids on equal footing. The additional features include quantifiable differences are:

- One time costs as well as in the continually running and maintenance costs.
- Equipment characteristics such as storage capacities, speeds of various units of computing device.
- In-built spare capacity as well as capability of the system to support additional peripherals.
- Additional support to be provided by the vendor.

### 4.10.2 Costing Factor

Cost consideration is quite important factor in computer acquisitions. Costs are of two types:

(i) One-time costs such as cost of site preparation (space, false ceilings, special floorings, electrical fittings, air- conditioning etc.)

(ii) Continued running and maintenance costs of the entire installations.

### 4.10.3 Equipment Characteristics

Hardware device which provides higher transfer rates (that is arrange number of bytes passing between various functional units per unit of time), or has large storage capacity or in case of printers, if the printed characters per line are quite high, then such additional characteristics get entitled to weightage to the extent of their practical utility to the buying organisation. Appropriate weightage can also be given to such characteristics as high mean-time-between-failures (MTBF), compatibility with the equipment, peripherals etc. available in the market.

### 4.10.4 Potential for Growth

Following features can be considered in this category:

- Potentiality of the system to grow beyond the currently specified capacity by adding on certain components,
- Potentiality of growth within a particular family of computer models,
- Capacity of the system to handle a large variety of peripherals,
- Ability of the system to handle additional workloads after considering the peak hour load.

### 4.10.5 Vendor Support

The features to be given weightage under the vendor support include:

- hardware maintenance facilities offered,
- training facilities provided
- assistance to be provided in software development
- back-up facilities provided by vendor in case of system failure
- comparative delivery periods offered by different vendors.

## 4.11 CRITERIA FOR VENDOR'S SELECTION

Mandatory requirement is that, if a vendor fails to meet them, he would be screened out without any reason. The desirable characteristics would surely be little bit difficult to evaluate because he may offer several alternatives in lieu of them. The criteria of vendor selection may be listed in descending order by importance as below:

### 4.11.1 Economic Factors

- Cost comparisons
- Return on investment
- Acquisition method

### 4.11.2 Hardware Factors

- Hardware performance and its reliability
- Facilities for modularity
- Provision for back up facilities
- Firmness of delivery date
- Compatibility with existing systems
- Expandability

### 4.11.3 Software Factors

- Performance of software and its price
- Efficiency and reliability of available software
- Programming languages available
- Availability of well documented package programs
- Firmness of delivery date for a promised software
- Ease of use and modification as per user requirements
- Portability and its capacity to interface with the environment.

### 4.11.4 Service Factors

- Facilities provided by the manufacturer for detecting errors in the new programs
- Providing of good training facilities
- Assistance in software development and conversion facilities provided
- Maintenance terms and quality

### 4.11.5 Reputation of Manufacturer

- Financial stability
- Past history for keeping promises

These criteria may have to be further sub-divided particularly for hardware performance and support services.

## 4.12 ACQUISITION FOR PROPRIETARY SOFTWARE PACKAGES

Apart from few cases where software packages are developed by outside agencies as per requirement and at the expenses of the user organisation, the proprietary rights in the package remain with the original developer. This means that software package developed by outside agencies can only be licensed for use. The mode of payment for the purpose can be possible in one of the following ways:

- Off-the-shelf acquisition of programs stored in a floppy or in a plug-in ROM module. (These modes are quite popular in the case of software to be used on PCs).
- Acquisition of the right to use a package by means of a monthly/periodical licence fee.
- One-off payment for a fixed period.
- Lump-sum payment in the beginning and periodical service charges later on.

Some computer vendors are also interested to supply software packages along with the system and thus they form an integral part of computer operations. However, they like to charge a nominal fee for the use of such packages just to establish ownership.

### 4.12.1 Technical Aspect of Proprietary Software

While considering the acquisition of a proprietary software package, following technical characteristics must be taken into account:

| | | |
|---|---|---|
| Memory size | : | Amount of memory required when resident in the CPU. |
| Run time | : | Some programs may occupy lesser memory but much larger run times. |
| Adaptability | : | Ability to mix into a multi-programming environment and to utilise the existing resources such as peripherals, operating systems, various utility programs, etc. |
| File storage | : | File storage requirements and how efficiently it can store and retrieve data. |
| Modularity | : | A package with a high degree of modularity has the capacity to operate on various machine with different configurations. |
| Expandability | : | It emphasizes the sensitivity of a software package to handle an increased volume of transactions or to integrate with other programs. |
| Reliability | : | Significant reliability related aspects also include: • the extent to which a package can still be used when a particular module fails, • types of errors on the part of the user which affect performance of the packages, • ease of recovery in case of failure, • capability of the package to run on different configuration. |
| Efficiency | : | This aspect encompasses capability of the package to perform under peak load conditions. Efficiency of a package much depends on the language in which it is written and the operating system used. |
| Ease of Implementation | : | Implementation of an application package refers to all the associated activities and vendor support up to the point where the package satisfies users needs and the user can run it independently. |
| Usability | : | It refers to the effort required to operate, prepare the input and interpret the output of a program. Additional points to be considered are portability and understandability. |
| Vendor support | : | This is critically important in such areas as documentation, software, modification/enhancement/maintenance and educating the user staff at different levels-clerical, technical, operational and managerial. |

Security : This refers to the in-built capability of the package to prevent unauthorised access to software and data and to maintain proper integrity of the system.

### 4.12.2 Approaches to Software Evaluation

For software evaluation, following approaches have been discussed:

(a) **Benchmarking:** Benchmark is nothing but a sample program specially designed to evaluate the comparative performance of hardware or software.

(b) **Experience of other users:** Vendors generally gives a list of users who are satisfied with their work. But it is advisable to seek the opinion independently from the existing users whose configuration and operational environment is closely identical.

(c) **Report of independent research organisation:** Now-a-days many research organisations undertake project of evaluating the proprietary software offered by various software agencies. These evaluations are objective and comprehensive in nature. They publish the report at regular interval. The prospective buyer of a software package can have faith in their evaluation.

## 4.13 SERVICE BUREAUX

It has been observed that installing in-house computer is quite costly affair. Thus a small user faces the problem of justifying such a large amount in the initial stage and recurring expenditure later on. It may also be possible that he may not like to get involved in the setting up and staffing of a computer department until a few applications have shown results. For such a user, number of consulting and service bureaux exist that provide computer facilities to their clients on a fee basis. This service is available on an as-needed basis or on a continuing contract basis.

Large number of such service bureaux are functioning in our country. Quality of service rendered by these vary widely. Some bureaux are run by companies who have acquired a computer for meeting their own requirements and rent out the spare time. Their rates are low but customer service is negligible. The user has to arrange for his own computer operator and related computer stationery. This type of bureaux is good for persons having knowledge of EDP and is available at a cheaper rate. On the other hand, some service bureaux provide everything such as computer operator, tapes and disks etc. Such bureaux are little bit expensive but they are customer oriented.

Alternatively, the user can entrust his entire job to a service bureau or data centre. The charges of such bureaux vary considerably depending on the responsibilities taken by them and the reputation of the firm. The charges are made up of the fixed cost for systems development work and also variable cost which depends on the volume of the data processed.

### 4.13.1 Advantages of Using Data Centres

The various advantages of using data centres instead of an in-house computer facility are given below:

(i) The major benefit of using data centre is that they can make use of computer without spending large initial amount.

(ii) It eliminates staff and management problems caused by the employment of a team of highly paid technical professionals in a rapidly changed field of computer.

(iii) The small organisation can utilise the expertise knowledge of experienced and qualified staff of data centre in his data processing job which is not available within the organisation.

(iv) There is no fear of equipments becoming obsolete.

(v) The small organisation can get valuable experience of working on computer before deciding whether or not to install an in-house computer.

### 4.13.2 Disadvantages of Using Data Centre

(i) Using data centre services relinquishes control of vital business data whereas in-house computer offers the possibility of strict security measure.

(ii) Loss of control over the time taken to process data is suffered by an organisation due to data centre. Off-premise computer processing may be inconvenient.

(iii) In using the data centre, staff members of the organisation are not getting familarity with working on computers.

(iv) Use of data centre makes the organisation dependent on a second party which is not aware of the need of the organisation.

## 4.14 FINANCING USE OF COMPUTERS

After deciding which computer to acquire, the basic configuration and a general plan for its expansion, a further decision has to be taken as how to finance the use of computers. There are three major approaches for financing the use of computers: renting, leasing or outright purchase. Determining which approach is appropriate depends on the characteristics and plans of the organisation at the time the acquisition is made. None of the above approaches have an edge over others. The features of each method of acquisition have been summarized in the following table:

**Comparison of Computer Systems Financing Options**

| Method of Acquistion | Advantages | Disadvantages |
|---|---|---|
| Renting | Short-term commitment. High level of flexibility. Does not require cash up front. | Most expensive option. Little control of equipment change. All vendors are not in favour of renting. |
| Leasing | Pre-determined payments for fixed period. Does not require cash up front. Usually better service from vendor than under rental. Little risk of obsolescence. | More expensive than purchase. May have limitations on hours or equipment use. |
| Outright purchase | Least cost in the long run. Distinct tax advantages in case of profit-making firm. A business investment. Full control over equipment use. | Risk of obsolescence. Permanent commitment. Full responsibility for all types of problems. Immediate and more requirements as compared to other options. |

### 4.14.1 Renting

The renting method is generally popular. Rent is paid for using the system on a short-term duration, generally from 1 to 12 months. It is paid on a monthly basis. Both the user and supplier have the option of cancelling the rental with advance notice, usually 30 or 60 days ahead of the termination date.

Because of short-term commitment, the renter (user) has lot of flexibility. The decision for purchasing a system can be delayed until financing is adequate. Flexibility can be particularly important when an organisation is experiencing planned rapid growth and will outgrow a specific system in a short period. Another advantage is that the user can obtain better maintainability, as the manufacturer is responsible for the maintenance of the equipment.

Rental is quite expensive as compared with other acquisition methods. Monthly payments are higher and the user organisation does not get any tax benefits, other than deduction of the monthly rental as a business expense.

### 4.14.2 Leasing

A lease is a commitment to use a system for a specific time, generally from three to seven years. Payments are pre-determined and do not change throughout the course of the lease. Depending on the terms of the lease, payments are monthly, quarterly, semi- annual or annual and include the cost of equipment service and maintenance. At the end of lease period, the lessor generally does not become the owner of equipment.

Leasing is less costly as compared with rental. Because of commitment for a longer duration, the supplier will generally provide better services and facilities to user. Leasing protects against technical absolescence also.

No capital investment is required to lease a computer system. Leasing offers specific tax benefits. In addition to deducting the cost of the lease as a business expense, tax credits are sometimes available for the investment, which directly lowers the income tax a business pays.

### 4.14.3 Outright Purchase

The ownership of computers through outright purchase is the most common method of computer acquisition and is becoming popular as lease costs rise. In due course of time, the purchase option frequently costs the least, especially in the light of the tax benefit that can sometimes be achieved.

Under purchase, the user organisation takes title to the equipment. Of course, the money for the purchase must be taken from operating funds or borrowed. And, in a sense the user organisation is locked into the system it purchases, since changing to a different system is difficult problem.

The user organisation must arrange its own maintenance, either by engaging his own maintenance staff, of by entering into maintenance contract with the manufacturer. The former method involves stocking of components and all the associated attendant problems. In addition, if the equipment was financed, payment on the loan must be made periodically, the cash outflow still may be lower than with renting or leasing, depending on the terms arranged by the purchaser. In return for the outgoing cash, purchase offers some tax benefits also.

**Check Your Progress 2**

1. What do you know about system maintenance?

   ......................................................................................................................................
   ......................................................................................................................................
   ......................................................................................................................................
   ......................................................................................................................................
   ......................................................................................................................................

2. What is the criteria for vendor's selection?

   ......................................................................................................................................
   ......................................................................................................................................
   ......................................................................................................................................
   ......................................................................................................................................
   ......................................................................................................................................

3. List out the different approaches for software evaluation.

   ......................................................................................................................................
   ......................................................................................................................................
   ......................................................................................................................................

..........................................................................................................

..........................................................................................................

What are the advantages of using Data Centres ?

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

What are the different ways for financing the use of computers?

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

..........................................................................................................

## 4.15 SUMMARY

This unit explains the significance of the implementation phase of the software development life cycle. Even though it occurs at the fag end of the life cycle, it can't be ignored because by doing so, all the efforts that has been put till the previous phase will become void. Each aspect of implementation i.e. training of personnel, converting the system and reviewing the system after implementation has a significant role to play in the successful implementation of the system.

## 4.16 MODEL ANSWERS

**Check Your Progress 1**

1. Training of operations and users can be organised in several different ways. Most important are:

   (i) Vendor and in-service training

   (ii) In-house training

2. Methods of system conversion are:

   (i) Parallel system

   (ii) Direct conversion

   (iii) Pilot system

   (iv) Phased-in method

3. A post-implementation review is an evaluation of a system in terms of the extent to which the system accomplishes stated objectives and actual project costs exceed initial estimates. It is usually a review of major problems that need converting and those that surfaced during the implementation phase.

**Check Your Progress 2**

1. The last part of the system development life cycle is system maintenance which is actually the implementation of the post-implementation review plan. Maintenance covers a wide range of activities including correcting coding and design errors, updating documentation and test data and upgrading user support.

2. The criteria of vendor selection may be as follows:
   (i) Economic factors
   (ii) Hardware factors
   (iii) Software factors
   (iv) Service factors
   (v) Reputation of manufacturer
3. Different approaches for software evaluation are:
   (i) Benchmarking
   (ii) Experience of other users
   (iii) Report of independent research organisation
4. Various advantages are:
   (i) One can make use of computers without spending large initial amount.
   (ii) Expert knowledge of qualified staff of data center can be utilised.
   (iii) There are no fear of equipments becoming obsolete
   (iv) It eliminates staff and management problems caused by employing highly paid technical professionals in a rapidly changed field of computer.
5. Different ways for financing the use of computers are:
   (i) Renting
   (ii) Leasing
   (iii) Outright purchase

Uttar Pradesh
Rajarshi Tandon Open University

# BCA-07

# Elements of Systems Analysis and Design

Block

# 4

## MANAGEMENT INFORMATION SYSTEM

# BLOCK INTRODUCTION

This block consists of four units. Unit 1 develops the basic idea of Management System (MIS) and our views on the contemporary role of MIS. It also traces the history of MIS over the past four decades and provides a perspective for the future direction in MIS. Unit 2 provides an overview of the Information Technology components that form the core of the delivery vehicle for MIS. Unit 3 touches upon some organisational issues relevant to successful application of MIS in an organisation. Unit 4 details the steps involved in building successful information systems in contemporary organisations.

# UNIT 1 INTRODUCTION TO MIS

## Structure



## 1.0 INTRODUCTION

This unit introduces the subject matter. Since MIS is studied both by computer professionals and management professionals, both the technical and organisational perspectives are provided in this module. Further elaboration of these ideas form the rest of the course. This unit will trace the history of computer applications in office management through Electronic Data Processing and its evolution to Management Information System. Recent trends like Decision Support Systems, Model Management Systems, Knowledge Based Systems will be classified within the broad framework of an evolutionary trend. The significant impact of Information Systems on the society at large and the emergence of Information Technology as a dominant industry will be outlined along with future implications in the Indian context.

## 1.1 OBJECTIVES

After going through this unit you should be able to :

- understand the notion of MIS
- appreciate the importance of MIS in organisation
- appreciate the relative role of computer systems & information systems
- appreciate the importance of commercial applications of computers
- develop a perspective to the evolving concept of MIS through the myriad term like EDP, MIS, DSS, ESS, KBS, etc.
- appreciate the significance of Information systems to the society at large.

## 1.2 WHAT IS MANAGEMENT INFORMATION SYSTEM (MIS)?

The *term* Management Information System (MIS) is familiar to most managers in the private sector, public sector and the government sector. Many MIS systems are in widespread use by the managers at all levels: operational, middle and senior management. Substantially large number of programmers and systems analysis are employed by many managers to build a variety of MIS systems. Naturally in the education of programmers and systems analysts as well as general managers, the subject of MIS occupies a key role.

The *concept* of Management Information System (MIS), however, is not clearly understood by many developers as well as the end users of these MIS systems. Some consider any computer based system to be an MIS. In the governmental sector, MIS primarily means a variety of reports, thanks to the emphasis on report generation by several funding agencies. For many, MIS is an evolutionary concept that evolved when the Electronic Data Processing Systems (EDP) matured. Some others consider MIS to be a discipline of management education similar to other disciplines like Accounting and Marketing. Many view MIS as a philosophy of providing help to managers in decision making. With the revolution in microprocessors or telecommunications and their impact on many of the MIS systems development these days, many managers view MIS as a bunch of technologies. The traditional Organisation and Methods (O & M) specialists view MIS as an implementation of the organisational systems and procedures. For many of the students of Computer Science, MIS is just a course on File Structures and File Processing with a relatively less demanding intellectual component!

In our view, Management Information Systems (MIS) involve all these and much more. The three sub-components — Management, Information and Systems — together bring out the focus clearly and effectively — *Systems* emphasizing a fair degree of integration and a holistic view; *Information* stressing on processed data in the context in which it is used by end users; *Management* focusing on the ultimate use of such information systems for managerial decision making rather than mere technology. While many professionals may like to substitute the term MIS by relatively nascent terms like Decision Support Systems (DSS) and Executive Support Systems (ESS), in our opinion MIS is the most widely used and understood terminology and retains the spirit of Management Information more comprehensively than most other terms.

While the conceptual MIS does not have to rely on a computational device, in the contemporary world where computers are ubiquitous by their presence, if not use, any meaningful MIS would exploit the power of computing and communication devices to the maximum extent.

For the purpose of this text, we would mean Management Information System (MIS) to be the set of computer based systems and procedures, designed to improve the managerial decision making process - that involve collection, organisation, distribution and storage of information for analysis and control.

Needless to mention, such a definition only includes formal decisions that can be codified in the form of systems and procedures. Care must be exercised to note that some decision situations may have far reaching consequences and yet be out of the reach of MIS. This includes innovative and intuitive decisions, emotional and personal decisions that may not lend themselves to formal frame work of analysis.

## 1.3 HISTORIC DEVELOPMENT

The modern digital computer was primarily designed for scientific calculations. Ever since the use of computer for the American Census work in 1954, however, the commercial application of computers far exceed the scientific applications. It is estimated that the commercial applications amount for as high as 80% of the investment in hardware and software. A significant portion of such commercial applications aims at providing the management of an enterprise, some form of information system support. The broad area that deals with such an information support came to be widely known as Management Information System (MIS).

The concept of Management Information System has changed substantially over the years. In the 50's and 60's, the management saw the potential of computers to process large amounts of data speedily and accurately. The speed and accuracy of such data processing equipment far exceeded that of human clerks, who used to undertake data processing in those days. The departments that were involved with such activities were known as *Electronic Data Processing* (EDP) departments. The focus of EDP was Record Keeping — an activity statutorily required in many organisations. The majority items whose records need to be kept were primarily accounting data — symbolically described by many users in India as Payroll data. So, many EDP departments fell in the administrative control of Accounting Departments. Interestingly in India, we had taken as early lead in such applications. A classic example in the Indian context is the extensive use of EDP by Indian Railways.

The next stage of development was concerned more with the insight and analysis that can possibly be provided by the routine availability of such accurate and timely data. In the 70's, when many organisations took to EDP with the availability of more powerful computers, there was a discernible shift from *data* to *information*. The focus was not on data per se but on the analysis of corporate data. There was a shift in the philosophy. Such a concept came to be widely known as *'Management Information System'*. Our use of MIS in contemporary world goes much beyond the MIS of the 70's but we retain the word as such. While many new jargon words are being proposed we find the term MIS more widely accepted and well understood (Davis[]).

BCA-07 (13B)

Without an excellent record keeping mechanism provided by EDP, no meaningful analysis of data would be possible. As such the, MIS presupposes, even today, an excellent record keeping function, through it is no longer fashionable to use the term EDP. The blame for EDP falling into disrepute lies squarely with the early EDP enthusiasts themselves; they were more interested in generating piles of data accurately than with their ultimate use of management. The very effort of such large data processing made it impossible to process the data in a timely manner. This was in spite of spectacular progress made in the processing capabilities of the computers of the 70's. Probably what went wrong with the EDP was their lack of focus on *levels of management*. Particularly the top management had neither the time nor the inclination to assimilate the large volumes of data presented by EDP departments. What was in fact needed was a mechanism of filtering the data to suit the level of the management. In the word of jargon what MIS provided was right information in right time people and not just all information to all people.

The 80's saw the Personal Computers (PC) revolution. In the 70's the top management relied on the staff departments of EDP and MIS to supply the necessary information. Their access to information was always indirect, until then. The Personal Computer and the desk-top metaphor changed the picture completely. With new generation software consisting of Word Processing, Database and Spreadsheet a new avenue of direct interaction with corporate data was open. At least those managers who did not mind 'getting hooked on' to the Personal Computers saw an immense potential through this avenue. Their demands for information were much higher, they were no longer content with indirect information support. The biggest pay-off for such direct use was the "what-if" analysis capability. Suddenly, the executives realized the "decision support" capability of the Personal Computer. This led to the emerge of *Decision Support Systems* — a new generation of systems with a new philosophy, pioneered by Keen [ ]. Such DSS had to be more flexible, demanded a better user-interface and adaptable to the decision style of the decision maker. The emergence of business graphics provided further boost to the development of such systems.

The spectacular growth in the Artificial Intelligence area, particularly in the Expert Systems sub-area (in the 60's and 70's) had to wait for the 80's for the arrival of powerful machines. Until then the theoretical developments could not be translated into useful, value adding products. The information and decision hungry managers of 80's saw a huge potential in the expert systems area. Combined with DSS philosophy the expert systems could supply a superior class of managerial information support. Such systems generally come to be known as *Knowledge Based Systems* (KBS).

The exciting results of Operation Research and Management Science, in a similar vein, had to await the arrival of PC revolution. Without the power of desk top computing and the subsequent easy access to information, tangible benefits from the models could not be derived. The philosophy of DSS, combined with the power of the OR models, working behind the screen are transforming the "What-If" capability to "What-Is-Best" capability in the past few years [ ]. Such systems that truly integrate Management Information Systems, DSS philosophy and OR/MS models came to be known as Model Management Systems (MMS) [ ].

The EDP targeted the operational level of the management. The MIS/DSS/MMS target the middle management. Very recently attempts have been made to provide information support to top management as well. Such systems are known as *Executive Information Systems* (EIS). The focus here is in the use of systems by direct end user whose time value is extremely high. Depending on the context of system may be MIS, DSS, EIS or KBS; but in either case the user interface must be superior. Some of the interfaces that are being researched for such EIS applications are Natural Language Interface, Voice Processing and Voice Response, Multi-Media like Graphics, Sound and Video images.

## 1.4 COMPUTER SYSTEMS AND MIS

The widespread availability of computer explains to a large extent the extensive use of computers to implement any meaningful MIS today. Since the major applications of computers today are for the design, development and operation of MIS systems, to a distant observer Computer Systems (CS) and Information Systems (IS) appear synonymous. However, Computer Systems provide only the technology component and successful MIS calls for understanding of the organisational systems and procedures. While the importance of good Computer Systems for the success of MIS cannot be under estimated, one should not make the mistake of over-emphasizing Computer Systems component in the design of MIS systems. Computer Systems (CS) and Information Systems (IS) are two disciplines that sufficiently overlap and yet have quit independent existence. While the Information System (IS) is man- centric, Computer Systems (CS) is machine-centric. IS is an applied area while CS has a strong theoretical foundation. IS is specific to managerial and organisational context, while CS is far more generic. The intellectual challenge is very high in CS discipline while IS calls for a very high conceptual challenge. While many of the tools of IS are context specific, the concepts of CS are by and large context independent.

## 1.5 ORGANISATIONAL SYSTEMS AND MIS

Since MIS is concerned largely with managerial applications, an appreciation of the theory of organisation is a necessary pre- requisite for successful application of MIS. Some professionals in the MIS area bring forth this point in the cryptic definition of MIS as "the supply of right information, in right time, at right level".

The level of management in the context of MIS refers to a classification of management, originally developed by Anthony. The different levels of management are generally referred to as a Pyramid in a pictorial form to emphasize the fact that in any organisation there are a few top positions, a large number of supervisory staff and a much large number of operational staff. Placing these three staff positions in order from top to bottom would lead to a structure loosely resembling the historic Egyptian pyramid.

Anthony classified the three levels as *strategic, tactical* and *operational*. The strategic management is concerned with long term policy decisions like new plant location, new products, diversification, etc.; they typically need a summary of plant/organisational level information, but need lots of unstructured and even vague information pertaining to the environment — the competitors, fiscal policy changes of the Government, emerging technologies etc. The tactical management comprising functional managers would need some external information but a lot of organisation-wide information to exercise control like budgeting, quality level, service level, inventory level etc. The operational management is only concerned with plant/organisational level information but in a far detailed manner like individual operator specific, machine specific and shift specific performance measures. The point will be further elaborated in Section 4.4 of this Block.

To be successful, MIS is an organisation must explicitly take into account this classification of management. Since the summary information to be provided to the tactical and strategic management must be culled out of operational information, the *accuracy* and *timeliness* of information collection and dissemination is important at the operational level. However at the tactical and operational level, *relevance* is the watch-word. A relevant but slightly inaccurate data is better than irrelevant but accurate data though the goal must be for the relevant and accurate data. The context decides the tradeoffs, particularly when cost considerations in data collection and processing are involved. historically the failures of early generation data processing departments are attributed to the lack of appreciation of this point. Some professionals call this process as the provision of *Information Filter*, meaning that only filtered information called out of operational data must be presented to the middle and top management. Some others put it more effectively in the form of emphasizing *efficiency* at the operational level and *effectiveness* at the tactical and strategic level. In fact there is a school of thought that believes that data processing serves the need of operational management. MIS serves the needs of tactical management and strategic level of management cannot be supported by information systems at all!

The point we would like to emphasize here, is the importance of mapping organisational level and structure into the design of any MIS for successful implementation. This once again brings home the point we made earlier that knowledge of Computer Systems comprise only a small fraction of the design of MIS; the major part calls for a clear understanding of the organisational dynamics.

## 1.6 LOGICAL FOUNDATION OF MIS

MIS as a discipline has no intrinsic theory of its own. The data processing applications had a simple goal of accurate and efficient data processing needing just simple control mechanism. With the refinement of data processing into MIS and later DSS there was increasing demand for "What-If capability. Such a demand necessitates the extensive use of mathematical, statistical, optimization and simulation models collectively known as the models of Operations Research and Management Science. While the models were powerful, they could not be applied to real world problems due to the non-availability of data. With streamlined data available to the decision maker through MIS systems, the power of statistical and operations research models could be very well explained. The classic example being the extensive use of Operations Research in Airline Industry [.]. Recently a new generation of spread sheet software with optimization capability reinforces this trend [ ]. Thus the quantitative tools of Operation Research and Management Science form one of the logical foundation of MIS.

Since MIS is primarily concerned with managerial decision making, the theory of organisation behaviour and the underlying understanding of human behaviour in the organisational context from another logical foundation for MIS area. The socioeconomic impact of competition, globalisation, democratisation and their impact on organisational structure must be well understood before one can design a successful MIS system today.

Computing Science would form the third logical foundation to the MIS field. Since every major MIS system developed today is built on some computational device (PC to mini to mainframe to super computer with or without an underlying computer network), a clear understanding of the various facets of computing becomes a major pre-requisite to the success of a MIS professional. While all areas of computing science directly or indirectly have significance in MIS work, data and file structures, database theory, design and implementation, computer networking, expert systems and artificial intelligence tools are by far the most important areas of computing science that contribute significantly to MIS discipline as of today.

Information theory [ ] would form the fourth logical foundation of MIS today. Information theory developed independently an ability to analytically quantify the information context of a message. This area also developed related areas like noise reduction, error detection and control, signal processing, image processing, compression schemes, image restoration and enhancement etc. In short it provides insight into the fundamental aspects of information processing at an abstract level. With the emergence of Multi-Media, Recognition Technology etc., Information Theory will play a key role in the future development of MIS. The related areas of systems design, systems analysis and system engineering [ ] provide further theoretical framework for structuring the information.

## 1.7 TYPICAL SYSTEMS

The typical MIS systems that will be in operation in any organisation can be classified in many ways. One classification is through the functional disciplines of management like Marketing, Finance, etc. Typical systems with examples in the operational, tactical and strategic level are as follows:

**Example of Typical MIS System**

| | Production | Finance | Personnel | Marketing |
|---|---|---|---|---|
| Strategic | New Plant Location | Alternative Financing | Welfare Policy | Competitor Survey |
| Tactical | Production Bottleneck | Variance Analysis | Performance Appraisal | Advertising |
| Operational | Daily Scheduling | Payroll | Leave Records | Sales Analysis |

While this classification convinces the reader of the usefulness of MIS in the organisational context it still does not provide a perspective into the overall impact of MIS in an organisation. A typical young executive entering an organisation is bewildered by a plethora of terms describing various computer based systems — Payroll, Inventory System, Marketing Support, Word Processing, Electronic Mail, Bulletin Board, Board Room Presentation, DBMS etc. Where and how do these systems fit into the organisation is not at all clear to him/her. It takes years of understanding of the organisation, reading manuals pertaining to the various systems and working experience in the actual design of one or more systems before a clear picture emerges. The remaining units in the block along with units in the other blocks like Systems Analysis, Database Management are to be studied in conjunction with the units in this block to get a better application of organisation-wide MIS system.

## 1.8 THE FUTURE

The previous section brought to focus the increasing impact of information processing for organisational decision making. The changing employment scene the world over brings this development into sharper focus. The dominant profession of mankind throughout the world used to be agriculture for many centuries. This is still true of countries like India which are yet to attain full economic development, the scene, however, is far difficult in economically developed countries, particularly after the Industrial Revolution. In the past industrial revolution years, the percentage of agriculture related jobs dwindled to a mere single digit percentage. Until the seventies there was a substantial growth in manufacturing oriented jobs. With the onset of *Information Revolution* the percentage of manufacturing related jobs has also been reduced substantially. The fastest growing sector is the *service sector* which is likely to account for almost 90% of the entire job market [ ]. The workers in this category are generally known as "white-collar" or "knowledge worker". This sector precisely is likely to be influenced by the information service to a maximum extent.

Services include banking, financial organisation, health care, entertainment, travel and tourism and education. All these sectors depend heavily on the information service for their very survival. Consequently the share of total business by Information Technology (IT) industry is likely to foreshadow the Oil and Automobile industry by the turn of the century. A recent publication — "IT - The Thrillion Dollar Opportunity" [ ] brings home this point very effectively.

Even in India, the IT industry is growing at a very respectable rate. While we got on to EDP in the early seventies, the overprotection by the political powers in the seventies and early eighties kept India very much behind the International scene in the IT area. Luckily we have been 'catching up' in the past six years and just this year the Indian IT industry crossed the Billion Dollar Mark []. With the recent liberalisation policy of the Government, it is likely to grow further.

The importance of IT to a country is so crucial these days, countries like Singapore and Hongkong have identified IT as their "strategic importance" areas for development. Even India has identified recently Software as the strategic area and plans to make it into a priority area.

The recent trends in the world scene also suggest a strong growth pattern for the information services. The collapse of Soviet Union, the emergence of Unified Germany, the thirst for democracy in several East European States, the unification efforts of Korea, Vietnam form a political pattern, the globalisation trend also pre-supposes spectacular investments in information services. The increasing competition and the resulting reduced product life cycle bring further emphasis for information sector. It is clear that Information Processing will form the dominant industry of the 21st century.

In the previous section we argued at length, the likely dominance of the information sector in the future. A natural off short of such a change will be a perceptible change in the quality of life (other things remaining unaffected). Since transportation and communication generally have the primal and dual role, improvements in information and communication will lead to substantial reduction in transportation. Routine travel to find, chase and expedite information can be more effectively accomplished by communication. Already the emergence of FAX service has reduced the transportation of over-night express courier mail substantially. With the wide spread access to information networks substantial information in the form of tables, reports, etc. will be transferred electronically rather than through bulky paper. Such developments will lead to reduced energy consumption, reduced pollution and a better ecological balance. While such ideas may appear far fetched in the Indian context at present, they are bound to improve the quality of life in future.

Even in the not so distant future, information management can substantially improve the quality of life in the Indian context in the following manner:

a) better land record management leading to less litigation and violence in the rural sector;

b) better natural resource planning leading to optimum utilisation of water, oil, electricity, coal, etc. and thereby reducing unnecessary inter-state tension, rivalry, etc.;

c) better banking and financial services with a superior service to the customers; leading to better money- management;

d) improved legal delivery system helping a large number of small farmers, business and common man by way of speedy disposal of cases;

e) networked educational institutions leading to better sharing of resources among larger number of citizens.

## 1.9 SUMMARY

This unit has provided an overview of the concept of MIS in an organisational context. While it has not told you about the steps in the development of a typical MIS, it has outlined the context and foundation. The next units will provide more information into the mechanics of MID. The examples of typical systems will help you to relate the various MIS subsystems. This unit has also provided an overview of the evolution of MIS over the past three decades and the emergence of the powerful information technology industry. It outlined the possible implications of such developments in the global and the Indian context.

## 1.10 SELF-ASSESSMENT EXERCISES

1. In your own words define Management Information Systems.
2. Using your employment background argue about the relevance of MIS to your day-to-day and long term planning.
3. Draw an Organisation Structure of your organisation. Map the Information Organisation with respect to this Organisation Chart.
4. What is meant by Anthony's Pyramid?
5. In your own words define the terms EDP, MIS, DSS, EIS & KBS.
6. What is the role of the Personal Computer (PC) in the information revolution?
7. Why is the IT industry emerging as the dominant industry?
8. Give two examples of the applications of IT in your organisation.

# UNIT 2 THE TECHNOLOGY COMPONENT

**Structure**



## 2.0 INTRODUCTION

Computers have fundamentally changed MIS from an abstract concept to concrete system that provide insight, competitive advantage and avenues to leverage business. The unprecedented technology breakthroughs and the price performance advantages that accompany these breakthroughs provide ample business opportunities to adapt, innovate and modify existing MIS systems. This unit surveys these technology developments from a used perspective so as to provide insight into the ever changing scene. The database management system is covered in more detail: historically DBMS (Data Base Management System) and their earlier incarnation of file processing systems provided the primary vehicle for developing Management Information Systems.

## 2.1 OBJECTIVES

After going through this unit you should be able to :

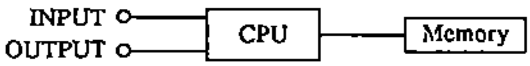
- appreciate the ever changing hardware and software scene
- appreciate dramatic developments in the telecom area
- get an insight into the tools of database management
- appreciate the impact of the emerging expert systems and the tools of artificial intelligence.

## 2.2 OVERVIEW OF COMPUTING TECHNOLOGY

The engine that runs the management information systems is the modem digital computer. Starting with the first ENIAC [ ] machine, computers have undergone spectacular growth in speed, processing power; interestingly there has been a steady decrease in costs and the size leading to what is generally known as micro-miniaturisation. To add to the attraction is the steady increase in reliability of the equipment.

Starting with the *first generation* that was built on vacuum tubes, the *second generation* machines were built using transistors; the *third generation* made use of large-scale integration while the *fourth generation* machines (the current ones) use Very Large Scale Integration (VLSI) as the basis of the technology. Current research efforts are aimed at the replacement of silicon that forms the basis of VLSI technology of the present with non-silicon materials. At least two companies Cray & Convex have already built their first GaA processor.

Conceptually all the present day computers are built on an architecture originally proposed by Von Neuman. The concept is quite simple. It is a sequential processing machine that can store programs and data. Even though the programs are instructions that operate on data they both can be stored together and can change meanings contextually. Using primary memory, control unit and arithmetic-logic unit the system can take data from an input data channel and output data on an output channel. Such a simple model is so powerful that it encompasses the entire spectrum of the machines today (excepting the few parallel processors being built in the recent years).




Computers are being built in so many sizes, shapes and with varying capabilities that it is difficult for any one including a professional to get a clear holistic picture. One classification that is generally used to classify them as micros, minis, mainframes and super computers. Even though for convenience the classification is based on some specifications, it gets very confusing with rapid strides being made in the processing power. Yesterday's mainframe becomes today's micro! Nevertheless the relative picture stays even though the individual specifications keep changing.

| | Micro | Mini | Mainframe | Super |
|---|---|---|---|---|
| **Cost** | <Rs.1,00,000 | <Rs.1 Million | <Rs.10 Million | <Rs. 20 Million |
| **Powers in MIPS** | 1+ | 10+ | 20+ | 100+ |
| **Data bus** | 8 bit | 16 bit | 32 bit | 64 bit |
| **Memory** | 1 MB | 10 MB | 100 MB | 400 MB |
| **Example** | IBM PC | HCL Horizon | IBM 3090 | CRAY YMP |

Adding to the confusion is the super-micro, super-mini classifications that describe positions in between.

An interesting characteristic of computer technology is the rapid strides being made in every sector. The processors are becoming more and more powerful. The *microprocessors* (which are actually a computer on a chip) from independent vendors like Intel, Motorola, AMD, NEC, Hitachi, etc. are so dominant in the market that proprietary processors are practically non existent the lower end. Even at the higher end the new generation RISC (Reduced Instruction Set Computers) form MIPS, Sun, DEC, IBM, Apple etc. are becoming more dominant. Only at the still higher end we find proprietary *macro processors* like CRAY and IBM processors.

The memory segment is changing fast. While in the 70's main memory (also known as Random Access Memory — RAM) of 1 MB was considered large, today we talk of large machines with 256 MB — 1 GB of RAM. In early 80's the RAM chips used to be 16K chips while today we mostly come across 256 KB DRAM chips and increasingly tend to use large capacity DRAM chips.

The secondary storage devices used to be dominated by magnetic media, like the ubiquitous magnetic tape. Miniaturised tape cartridges are slowly replacing the bulky spool type magnetic tapes. While the tapes are sequential access devices, we have a whole gamut of direct access devices. The floppy drive (changeable media) is the most popular direct access storage media. The early generation of 8" floppy disks have disappeared even in the Indian scene practically giving way to 5.25" disks in several categories (Double Sided Double Density DSDD with 360KB capacity, High Density with 1.2MB capacity being the most popular in India). The new 'pocket' size 3.5" (1.44MB capacity) mini floppy drive will definitely replace the 5.25" floppies in the next two or three years. The 3.5" disk is better made, well protected, more handy and has larger storage capacity. Both the media (floppy disks) and floppy disk drives (device to read the floppy disk) are locally manufactured in India (Amkette, Verbatim diskettes and L&T floppy disk drives).

The hard disk scene also offers a bewildering variety. By the time the Indian vendors got into the PC business the 10MB disks had become obsolete. So, we have disks drives of 20MB, 40MB & 80MB capacity being manufactured locally (L&T). A large variety of disk drives with capacities up to 1.3GB (1 Gigabyte = 10,000 MB) are available in the market to suit the varying needs.

A new generation of optical storage technologies are slowly appearing in the market. These devices being non mechanical have a superior reliability and life. Once again we have a variety of technologies starting from CD_ROM (a read only sequential device) to floptical drives (optical floppy drives) to optical disks. This technology is yet to arrive in India (by way of local manufacture).

The peripheral devices (incidentally no longer peripheral in overall investment) offer a bewildering variety as well. Luckily card readers, card printers have moved to the museums. The most common peripheral, viz. a CRT based VDU, often called terminal is available from a large number of vendors. Based on a standard emulation originally developed at DEC (Digital Equipment Corporation) they are available as VT-1—, VT-220, VT-320 terminals. The more sophisticated ones emulate Tektronics graphics terminals and are loosely called Tek terminals. Specially designed to display local scripts we have Indian Script Terminals too, generally called GIST terminals. Terminals which can generate PC scan codes, called PC terminals are also available.

While the terminal are both input and output devices we have a large number of output devices like printers and plotters. Printers constitute a sizable peripheral investment particularly in business applications that rely on "black and white" records. Starting from low cost, draft quality 9 pin Dot Matrix Printers (DMP) to 24 pin Letter Quality Dot Matrix Printers, we have line printers (300 lpm/600 lpm), line matrix printers (dot matrix printers that print like line printers) and the expensive pretty printing laser printers. In the Indian context we have an excellent choice — Dot Matrix printers from Wipro-Epson, TVSE, Line printers from Lipi, Data Products and Transmatic and Laser Printers from HCL-HP and Modi Xerox. The plotter market is more focused on engineering segment.

A number of other devices like Mouse, Digitizer/tablet, Scanner, etc. are used for more specific applications and are not covered in this quick overview.

On the software side, there is an equally impressive growth. The first generation machines needed to be programmed in their native machine language. The next generation could be programmed in a more human (and hence higher level) assembler language. The third generation machines could be programmed in a still more human application specific (and hence called higher level languages) like FORTRAN (Scientific), COBOL (Business), ALGOL (algorithm), SNOBOL (String Processing), LISP and PROLOG (AI), Pascal (instructional), making computer programming into a major learning activity with millions of people acquiring skills in programming using these languages. So much so, that many people carry the feeling that Computer Science is nothing but learning a series of Cacophony languages like COBOL, Pascal, C in street corner shops!

Recently there has been a generation of languages like FOCUS, Application by Forms, SQL, QBE which make the job of developing applications much faster. These application generators are generally known as the 4GL (4th Generation Languages).

To cater to the need of Executive Support Systems there are more specialised languages like IFPS (Interactive Financial Planning Systems) from ExecuCom and INTELECT a natural language based query system from IntelliCorp.

The programming languages and the applications interface with the hardware through another piece of software called Operating Systems, information systems professionals may never program the operating systems directly but still need an appreciation of the capabilities of such software. They vary in complexity considerably too. While there are proprietary operating systems like IBM's MVS, DEC's VMS many are converging towards the Unix operating system or its variant. Unix originally was developed in several academic/research institutions under AT&T's sponsorship. Currently it is marketed by AT&T. Several vendors port it on to their hardware platform and sell it under their names like HP-UX from Hewelett Packard, A/IX from IBM, Convex-OS from Convex. Unix like operating systems are multi user operating systems allowing several people access the system simultaneously. On the PC side we have the Disk Operating System DOS (in several versions) that is primarily a single user and single tasking. A graphical user interface GUI that permits multi-tasking and provide a consistent interface across all applications is available as Windows Operating System on PC class machines. On the Unix O/S also GUI's like open windows from Sun, Motif from OSF (Open Software Foundation), NewWave from HP are becoming available.

Among the application software that has the profound impact on business application is trio - word Processing, Spreadsheet and Database. In the Indian context the dominant Word Processor is Word Star, the dominant Spreadsheet is Lotus 1-2-3 and the dominant database is dBase. They are marketed respectively by Sujata, ITC and Wipro Systems. The business graphics software is slowly catching up — thanks to the popularity of Harvard Graphics (marketed by NIIT). As a desktop publishing software Ventura from Xerox (marketed by Microland) is the most popular along with Indian Language support (Prakashak). The high end database management system (DBMS) market is dominated by Oracle, Ingres, unify, Sybase and Informix.

The scene in this area will be changing continuously. What is attempted here is to present the current picture in a perspective that end users can develop an appreciation and insight without getting lost in the immense variety of options available in every segment.

## 2.3 OVERVIEW OF COMMUNICATION TECHNOLOGY

The communication revolution has not logged behind either. The POT (Plain Old telephone) got a metamorphosis with the advent of low cost phone chips into a smart phone. Offering such frills like repeat dial to start with they offer increasingly complex features like memory dial, tune/pulse changer, hands-free dialing, timer, wake up calls, hold features, etc. Ultimately features like picture will also be supported in addition to cordless, radio cellular and other features, leading to an Intelligent Customer Premise Equipment providing integrated voice, data, fax and picture capability. Other value added schemes are answering paging, etc.

As the exchange front things are moving too. The mechanical exchanges are giving way to analog electronic exchanges and ultimately digital exchanges. They provide special services like call forward, conferencing, etc. that significantly help business. Special services like directory services, toll-free services, call-collect services along with itemized accounting are natural by products of such exchanges with significant impact on business practices. The innovative experiment of C-DOT contributed significantly to this arena.

With the increase in sophistication of exchanges and computers the two are getting to influence each other significantly. The computer communication in a local area network (LAN) does not make use of telephone network, but wide area networks (WAN) primarily make use of telephone network. Remote communication using a low cost PC & modem is possible even in India today. With the distance no longer the problem, emerging services called network sauces like electronic mail, remote login, remote databases access and search are becoming possible. These also can affect business practices significantly.

## 2.4 DATABASE TECHNOLOGY

### 2.4.1 Introduction

At the heart of the information systems of an organisation is the central repository of corporate data that must be accessed and used by every section of the organisation. This data resource is traditionally called the database of the organisation. The modern information system primarily aims at organising this data resource in an effective manner - for flexible yet controlled access keeping the security and privacy issues in a cost effective yet very fast, corporate wide access. This is precisely the technology of the database. A typical analogy is the management of corporate finance. The financial management structure of the organisation provides for an operating procedure, authority, control in the form of budget and security measures; yet with a flexibility that day-to-day operations run smoothly. Just as financial management has evolved over the years - from barter to currency trade to financial market to instruments of finance, databases have evolved from traditional paper files to computer files to a modern sophisticated software widely known as database management systems (DBMS).

The early generation of file processing system relied on islands of data across the organisation - every functional unit maintaining its own set of application programs and data - finance sub-system, purchase sub-system, etc. such an organisation was simple to operate and had local control. Historically the functional units took to computerisation based on individual initiative and this model was viable. With the corporate vision including information system in its agenda, this model has to change. The data redundancy and the resulting inconsistencies had to go; the constantly changing corporate information must be instantly available to every corner of the organisation thereby propelling the need for DBMS.

The superiority of Database Management Systems over conventional file processing systems can be explained with the help of the following example.

Consider the student record administration in a typical College or University. Assume that in our hypothetical university the DOSA (Dean of Students Affairs) maintains all the personal record like permanent address, scholarship status, medical records, hostel dues, etc. and the DOAA (Dean of Academic Affairs) maintains all the academic information like courses, grades, ranks, awards, summer projects, theses projects, etc. In a traditional approach the DOSA office and the DOAA office will have independent files and independent programs. By the very nature of the operation some information like student name, address, department, etc. will have to be duplicated in both the files. Since the final certificates are printed by the DOAA office the students are likely to point out any minor errors in their names like misspellings, last and first name interchanges which are rather common among Indian names. The same student may not take the same care to correct the name errors in the DOSA office because of its inconsequential position. Hence within the same institute/college/university the same information is maintained to varying degrees of accuracy. This obviously is not serious but an information about a student getting terminated on academic efficiency may not be reflected in the DOSA office leading to a lost of opportunity of not collecting a substantial amount of dues/fine from the library/labs/hostels of the institute from the dismissed students. Even delay of this vital information is reaching DOSA office may lead to substantial losses for the institute. Such deficiencies arise due to inconsistency of data among different units of the organisation. Replicating the same information in many departments also leads to substantial data redundancy leading to large scale store inefficiency. Often such an redundancy is uncontrolled; the extent of redundancy may not be known at all. For performance consideration some amount of redundancy may be introduced but that is *controlled redundancy* by design and not by default.

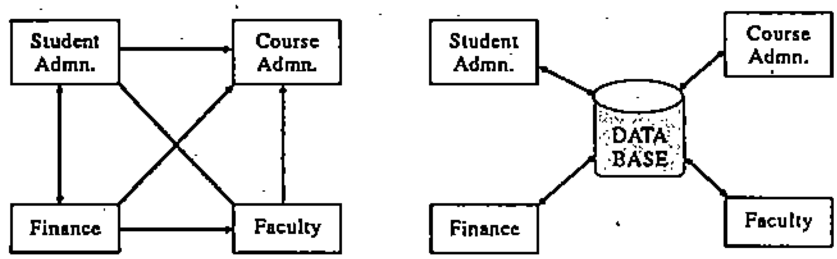
Another drawback of multiplication of data may be representation inconsistency; for example, one of the files may contain the address fields neatly decomposed into street address, city and postal address by a not so imaginative department may have kept the entire address field as single field making a query about all the addresses in a specific city nearly impossible. The data is there and yet not there!

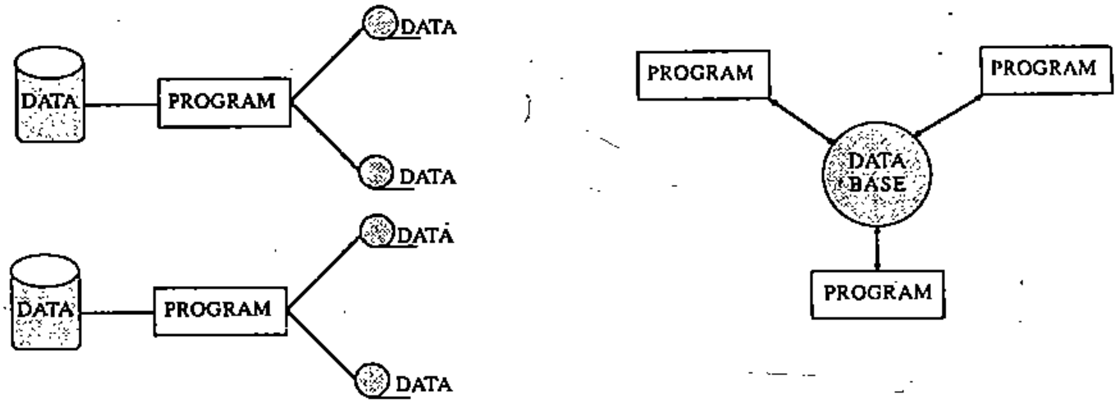
A related but more common side effect of duplication is unit inconsistency; for example DOSA office may maintain the students annual scholarship value, while DOAA maintain the monthly scholarship value, while DOAA maintain the monthly scholarship amount. Unfortunately the files may not contain information about the units, i.e., annual/monthly, etc. so that a not-so careful application program involving both the files may generate a lot of garbage output!

The more serious draw backs of traditional file processing system is the lack of privacy/security/controlled access mechanism that modern DBMS provides. With the increasing importance attached to data and its view as corporate resource, such issues are of paramount importance. The modern DBMS provide for precisely such features.

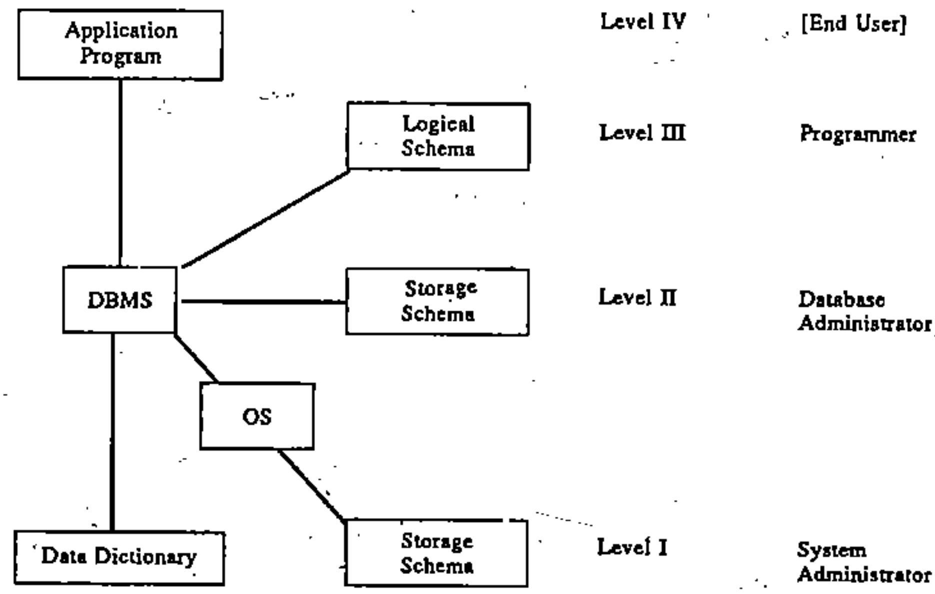
At a conceptual level all these deficiencies can be attributed to just one short coming of the file processing system. They impose a close dependency between programs that use the data and data themselves. The modern DBMS on the other hand provide *program-data independence* that basically separate a physical structure of the data (file) from the logical structure of the data used inside programs. By providing such an independence DBMS provide a superior system of file handling leading to a very high level of flexibility, consistency and simplicity. Essentially they allow applications to write programs that are general enough to operate on files whose structures can be made available to the program. In other words DataBase Management Systems are generalised file processing systems.

The precise distinction between traditional file systems and DBMS are explained through the following diagrams:



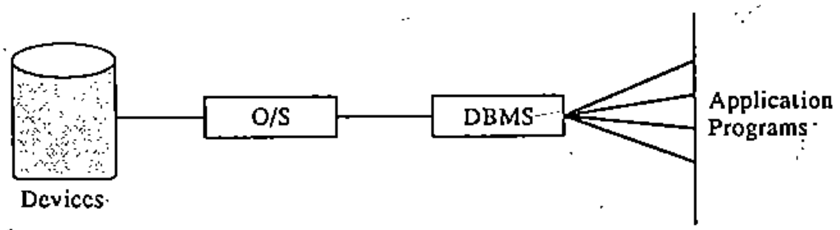




Another significant contribution of DBMS to the data processing field is the introduction of concept of levels of abstraction of data. The physical and logical independence of data permits multiple levels in which the end user/programmer/data administrator may view the data, every view not affecting the other views of data. For example data administration having to change the physical device or the media or location of a file may not bother the programmer at all. Similarly the introduction of a new and more efficient data structure to store a file may not be seen by an end user; in fact an end user may even able to view the data stored in a large DBMS like Oracle/Ingres using a more familiar Personal Computer database system like dBase. Such different *views* of data by different persons, generally called *schema* permit a superior architecture for the organisation of corporate data that is at once appealing. This permits multiple specialists like system administrator/programmer and end-user to coexist rather peacefully. Such an architecture can be diagramatically pictured as follows [Date].



Such a division of responsibilities so familiar to organisation pundits also permit a clearly distinguishable partition of the basis activities of file processing: *data definition* and *data manipulation*. The data definition phase consists of describing the structure of the data and not the contents. Naturally the structure remains invariant over longer periods of time and need be the concern of only the system administrators (a minority). A large number of programmers and end users whose prime concern will be data manipulation adding/deleting/modifying/viewing/querying/searching/sorting data that is subject to a structure that is predetermined. Since the technical back ground of people involved in data definition and data manipulation are likely to be significantly different it is possible to have every specialised data definition construct but very friendly, simple yet powerful languages like SQL (Structure Query Language), QBE (Query By Example), QBF (Query By Form), ABF (Application By Form) which we will consider in later sections.

To the typical user DBMS, which is nothing but a very sophisticated piece of software appears like this -




## 2.4.2 Data Modelling

The database management systems being generalised file processing systems need a model of the data that forms the internal structure of the DBMS software. The earliest of the DBMS viewed the data as a hierarchy — possibly a fall out of the nested level structure of a COBOL record. Such a structure allowed data to be abstracted at different levels of parentage: root, first level child, second level etc. and formed the basis of many of the early implementation of large scale databases. The classic hierarchical model based DBMS is IBM's IMS (Information Management Systems).

*Hierarchical data model* effectively use the idea of a hierarchy but cannot model the many-to-one and many-to-many relationships that are common in many business examples. The concept of a boss with several subordinates can be well abstracted by hierarchical model but the same boss may be a subordinate of one of this subordinates, may be in another context, committee, etc. cannot be modeled by hierarchical data model. To circumvent this difficulty the *Network data model* was introduced which can model the relationships as a network that is more general than the hierarchy. However this extension also was also of limited help only.

Both the hierarchical and network models suffer from their inflexibility to handle anging database structure. In addition querying the contents of the databases needed complex access mechanism depending on the data item that is being searched. The root level data item could be retrieved much faster than an item at the deeper levels of hierarchy. Over and above this both the hierarchical and network models were not elegant.

## 2.4.3 Relational Model

Codd [ ] in a seminal paper in the mid 70's proposed the *relational model* based on the mathematical structure of the *relation*. Intuitively in a relational model any database can be viewed as a set of relations with each relation having a simple structure of a *table* consisting of a set of *rows* and *columns*. Every row consists of a multiple tuple with values corresponding to a every column. Every row consists of a tuple with values of the different columns taking values different from that of the different row. The values permitted for every column (defined as domain) is pre-specified so that every tuple looks alike though different. It is generally expected that the rows contain values such that two entries contain values different from each other at least for one column. No ordering of the rows is implied but the columns have specific order; but the column entries take values from a specified domain for every column. In the traditional file processing sense every row is a *record* and every column is a *field* (sometimes also known as attribute). Such a simple construct is so elegant it is at once appealing. Codd [ ] could also provide a mathematical foundation for relational DBMS using the mathematical notion of relation. Naturally relational DBMS received substantial attention from the developers today almost all systems under development are relational.

In addition to elegance relational systems have some deep conceptual beautifies also. The most striking is the *completeness:* that is in the relational system one operates on relations and get the results of any query or update in the form of relations. This is true for any database that is modeled as a relational database. Because it is sufficient to have operations that can efficiently handle just one data object called table, relational systems have built in simplicity and the accompanying efficiency.

### 2.4.4 Structured Query Language (SQL)

Another major attraction of relational system is the simplicity of operation needed to manipulate the relations. Any query on a single table can be performed using just two operations SELECT that selects a set of rows (intuitively a horizontal cut of the table), followed by a PROJECT operator that selectively limits the fields (columns) for displaying (intuitively a vertical cut of the table). These two operations can be shown to be sufficient to perform any query or update of arbitrary complexity.

When query or update involves more than one table we need one more operator JOIN that creates momentarily a large table that matches the rows of two tables joined together over the common field. A simple example will illustrate the point. Consider the relations

Course

| cno | ctitle |
|---|---|

Student

| sno | sname |
|---|---|

Enrollment

| sno | cno |
|---|---|

Join of Student & Enrollment would yield

| sno | sname | cno |
|---|---|---|

The SELECT, PROJECT & JOIN operators together are powerful enough that any query of arbitrary complexity can in fact be written in just one sentence though a complicated one! This triplet forms the core of a 4GL called SQL (Structured Query Language) that has become the de-facto standard query language with even an ANSI (American National Standards Institute) stamp. Every major database vendor provides a SQL capability.

Last but not the least SQL stores the tables and their columnar structure itself as a table so that one can query the table of tables also using similar construct permitting a tremendous flexibility to the end user.

### 2.4.5 Fourth Generation Language (4GL)

Because of the sheer size of applications of databases the large numbers of end users of databases have been demanding special languages particularly fine tuned for their applications. The early generation of Report Generators belong to this category. In the last few years almost every database vendor is packing a 4GL product with his RDBMS engine.

These 4GL's typically include SQL as their core product. SQL being primarily an 'ad-hoc' query language lacks formatting features. 4GL's provide formatting feature to produce quality reports. Many of the 4GL's also provide a means of 'quick application development' like Application-By-Form. Here the user quickly design a few FORM screens for user input and some other FORM screens for user-output. By filling (either full or part) of the input screen the necessary outputs can be generated even by an occasional end user who may be a novice to computers and databases.

Facilities exit to edit/format the user data (like case conversion, numeric/character alignment), databse validation (alpha, numeric, range check) databse look up and feed back (like code number decoding into the actual values, e.g., Roll No. to name).

### 2.4.6 Complex Database

Recently data models that are more sophisticated are being researched. These models can capture more complex objects than the traditional DBMS could do. The traditional databases were limited to a few data types numeric (integer & real), character strings and special cases (Y/N character string for boolean, dd/mm/yy like character string for date). However, the increasing complexity of database applications calls for handling more complex data objects like scanned images (maps, line drawings, pictures, photographs etc.), audio and video images. Databases to handle such complex data objects are known as *Multi- Media* databases.

## 2.5 DECISION SUPPORT SYSTEMS

Systems that support decisions are simply the Decision Support Systems (DSS). DSS represents both a philosophy and a set of tools. At the philosophical level, it is the next stage of evolution of EDP & MIS. While EDP concentrated on accurate and timely processing of data, MIS emphasized the value added component of data, namely information. DSS goes one step further in emphasizing the organisation and presentation of information in a manner that has direct releance to the supporting phase of a decision maker. Here the goal is to improve the quality of decision making through information. Naturally the issue of access and flexibility are also important in organising the information.

A more deeper concept that also gets emphasized in DSS philosophy is the recognition of the central role played by decision maker — a human manager. The salutary importance of this philosophy indirectly removed by the possible threat posed by some early passionate MIS professionals who proposed that large corporate wide databases would replace the managers. DSS disproved such a myth thereby creating a clearer picture of the role of the machine Vs man.

There are several techniques of decision support. All of them involve an innovative integration of 'what-if' capabilities with data processing/information processing systems. In his seminal text Keen [] refers to several early generation of DSS systems that took great pains to illustrate integration. Such an integration calls for multiple integration at different levels man-machine integration, hardware-software integration and data- model integration. One of the pioneering early DSS implementation GDSS [] integrates a natural resource planner who will use maps — a natural information resource for his profession — with models of location. GDSS [] also used an early integration of a computer graphics software with an OR modeling software. An excellent example of a similar work in the in Indian context is the Gujrat [].

One of the technique that evolved as part of the DSS growth is the influence diagram [Plane, ]. This is a graphical tool to document the decision process in the form of logical diagram. It also allows an excellent top-down version of the documentation process of the logic. Using very simple graphical symbols to indicate goals, decisions and outcomes, influence diagrams capture the essence of the decision maker excellently. Coupled with a powerful decision analysis software like IFPS [], influence diagrams can be an excellent decision tool for the analysis oriented managers.

## 2.6 KNOWLEDGE BASED SYSTEMS

Knowledge Based Systems (KBS) goes beyond the decision support philosophy to indicate the expert system technology into the decision making framework. Expert Systems (ES) have been the tools and the techniques perfected by artificial intelligence (AI) researchers to deduce decision influences based on codified knowledge. The codification of knowledge uses the principles of knowledge representation (part of the large theoretical ideas of knowledge engineering). Typically such codification uses rules like IF-THEN rules to represent logical implications. Using first order predicate calculus, it is fairly easy to construct inference engines that use forward chaining or backward chaining to perform the induction.

One of the earliest expert system that had considerable success was the MYCIN [] project that used expert system based inferencing to diagnose diseases using clinical measurements. Fueled by the early success several successful systems have been built to cater the different applications the important one being PROSPECTOR [] for mineral exploration, XCON [] for configuring VAX computer, DENDRAL []. When many other researchers wanted to apply the expert systems ideas for diverse applications, expert system shells [] were built. These shells have a flexibility that they are independent of the knowledge bases so that one can build just the knowledge base and use the same shell for different applications. Such de-linking of the knowledge base from the inference engine using expert system shells has helped significantly in the large scale application of expert systems to a variety of situations.

## 2.7 SUMMARY

This unit outlines the spectacular developments taking place in the computing and communication fields; these developments are also leading to changing the role of MIS from a passive data support role to the role of providing new business opportunities and a competitive weapon in the era of ever increasing competitive world. While the mastery of these technologies takes years even for specialized professionals the demanding job of a contemporary MIS professional compels him/her to keep abreast with the changing scenario to exploit the potential of these technologies in an optimal manner.

## 2.8 SELF-ASSESSMENT EXERCISES

1) Briefly outline the computers and communication revolution.
2) Explain briefly the concept of independence in the context of DBMS.
3) Define DBMS in general and Relational DBMS in particular.
4) What are DSS and KBS?
5) What is an Expert Systems?

# UNIT 3 THE ORGANISATIONAL IMPACT OF MIS

## Structure



## 3.0 INTRODUCTION

Unit 1 underlines the importance of understanding organisation for a successful implementation of MIS. In this Unit we reaffirm some of the ideas and provide a frame work that calls for viewing information as a corporate resource just as other resources like Finance, Personnel or Machines. With the changing role of MIS this unit also outlines that competitive advantage that can accrue due to effective use of MIS.

## 3.1 OBJECTIVES

After going through this unit you should be able to :

- view information as a resource
- understand the role of MIS in future
- appreciate the importance of MIS profession.

## 3.2 INFORMATION AS A RESOURCE

Traditional economists of the pre-industrial era considered only land and labour and capital as the resources. With the emergence of industrial revolution and the availability of powered engines the importance of the resources changes relatively. The technology of mass manufacture placed a premium on the "technology" which emerged as the major resource. In fact this resource particularly in the context of changing job mix, as outlined in Unit 1, completely overshadowed the traditional resources. A shining example to illustrate this point is the emergence of Japan as the major international power.

With the recent information revolution, the prized resource is actually information. This point can be better appreciated when one considers the success or failure of products that appeared in the market at the right time or wrong time. While the technology of two products would be comparable, the vendor who brings in the product a few months ahead of the competitor wins substantially more market than the later. Late introduction a new technology product in the market may even translate to a virtual collapse of an organisation.

The information being such a vital resource need to be managed just as other resource like money. Formal methods to control, monitor and evaluate this resource are needed to manage the organisation successfully. We have discussed this issue in section 2.4.1. There is also a need for corporate strategy to manage such a vital resource.

## 3.3 INFORMATION FOR COMPETITIVE ADVANTAGE

Recently organisations have not only found out the immense utility of information systems to better manage their organisations but are also feeling the potential of the competitive advantage provided by information as a resource. The classic example of the use of information for competitive advantage is SABRE Airline Reservation System [ ] of American Airlines and the Frequent Filer Option introduced by Airlines the world over. These are examples that illustrate the use of information not just to run business effectively but to open up new business vistas.

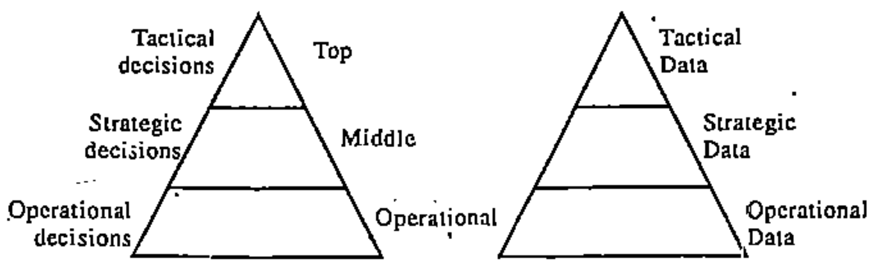
Even in the Indian context, the Citi Bank's expansion into a larger market share compare to Grindlays Bank is an example. Due to Government restrictions Citi Bank had to contend with their existing 6 branches compared to 26 branches of Grindlays. Using telecommunication, Citi Bank could still manage to capture large percentage of the business.

Information as a resource can provide new value added business prospects as well. The reduction in Telephone Tariff at off peak hours by the Dept. of Telecommunications last year is an example to illustrate this point. Based on the data collection capabilities of the new generation trunk exchanges it was much easier to detect the poor utilisation of trunk lines. Similar exercises would not have been easily possible but for the information capture capabilities of the electronic exchanges.

## 3.4 ORGANISATION, INFORMATION AND DECISION

Successful development of information systems call for a deep understanding of the organisational structure and dynamics of the enterprise. Some organisation are goal oriented, the analyst must be clear as to what information exactly need to be collected, stored and analysed. Since every information must have a context, only operational information that ultimately has some decision making contribution must be collected. Second the information collected and processed must be consistent with the level of the organisation to which it is to be presented. In this context, a classification of organisation by Anthony [] is very useful.

According to Anthony's classification, there are basically three levels of management, independent of the size of the enterprise: operational level, middle level and top level management. Operational decisions that call for large volumes of internal data (local to the enterprise). The middle management is concerned with medium range (tactical) decisions that call for much less information. The top management being concerned with long term (strategic) decisions calling for just a few vital internal information but a lot of external information as well. Any successful information system should take into account such a pattern of information needs by the management. This is generally pictorially displayed in the form of Management vs Information Pyramid..




The importance of information to management is further emphasized by the fact that much of management is primarily decision making. While there are several views of what constitute management, the generally accepted planning, organising, coordinating, directing and control are all concerned with decision making. In this text we take such a view of management and we perceive management information systems to support such managerial decision making. We also would like to emphasize that information systems should address clearly the situations of programmed decisions and non-programmed[] decisions by properly structuring the appropriate information. Failure to recognize intrinsic difference may lead to a failure of the information system.

Several functional areas of management viz. personnel, marketing, production, finance and services will be significantly influenced by the information systems that are to be implemented in the organisation. Care must also be taken to recognize the fact that in every functional area the mapping of the informational pyramid must be carefully worked out. Typical example of the three levels of information among the functional areas of management are as follows (reproduced from Section 1.7).

**Example of Typical MIS System**

| | Production | Finance | Personnel | Marketing |
|---|---|---|---|---|
| **Strategic** | New Plant Location | Alternative Financing | Welfare Policy | Competitor Survey |
| **Tactical** | Production Bottleneck | Variance Analysis | Performance Appraisal | Advertising |
| **Operational** | Daily Scheduling | Payroll | Leave Records | Sales Analysis |

**Operational Information :**

Daily schedules refers to the detailed assignment of jobs to machines or machines to operators in a production environment. In a service organisation like hospital it may be a duty roster for nurses, doctors or other paramedical staff. Such schedules must be detailed, unambiguous reports produced in large number at relatively low cost. Most of the information contained in such reports is internal - to the special shop or hospital.

The payroll represents a typical operational system for the financial management. Since such systems must execute accurately data pertaining to a large number of staff in a timely manner, month after month, cost based efficiency and speed would be a major consideration in the design of such a system.

Leave records constitute a major statutory record that must be maintained for every employee, through the many years the employee serves the organisation. The MIS system built for such an application has a primary aim of minimizing the chore involved in such large scale mundane record keeping. Once again the speed and accuracy are the major performance measures of such systems.

Detailed Sales Analysis is a must for any sales and marketing function. This might involve a very detailed data collection and processing pertaining to every salesman, every product over a long time span of an entire year or a quarter with even details of the region, market segment etc. Naturally it is an involved data collection and processing function. Accuracy and timeliness of this effort would decide the quality of tactical and strategic decisions that may be based on summary information generated out of this data. Accuracy and disaggregation would be the watch word for such detailed data collection effort.

**Tactical Information :**

At the tactical one could visualize a production bottleneck analysis in a manufacturing environment. Such analysis would call for senior management involvement by people with years of experience. Results arising out of such analysis are likely to have medium range impact. Naturally information systems to provide such information must have summary information, with provision for detailed information when called for. Comparative analysis shift wise, plant-wise, machine-wise, operator-wise etc. may be necessary. The watch word would be insight and analysis and not mere reporting of data.

Variance Analysis for the finance function would call for systems that point out deficiencies, cost over runs, budget excess by carefully matching goal or target information with summary information generated by operational data. Once again the accuracy is important; but what is more important is the clear recognition and highlighting of patterns that can help the decision maker to initiate action and bring systems under control. Timeliness is a must as the systems must be re-set before they are too late.

A tactical systems under personnel function is the Performance Appraisal which would take into account among several other things, the individual employee leave records. The details may not be that important but patterns are. Whenever they are clearly visible. Certain policy conformance may also be checked through such analysis. Since decisions based on such systems are likely to have medium to long term impact accuracy should not be underestimated, while keeping the priority of analysis.

Advertising Information System is another examples of tactical information system for the marketing function of management. Needless to say planning for advertisement would use much of the information generated by detailed sales analysis (an operational MIS system). But to be able to decide on the levels of advertisement, mix of advertisement, budget for advertisement etc. calls for certain policy parameters as well as environmental (outside organisation) information. This point must be clearly borne in mind in designing MIS for tactical decision situations.

**Strategic Information:**

A strategic decision by the production function of the management is a New Plant Location. This would use much of the internal information generated by the tactical system designed to analyze the production bottleneck that is internal to the organisation. But a long term decision like location of a new plant is likely to be influenced far more by environmental information like changing market, changing technology, changing fiscal and governmental policy like de-regulation, tax incentive for backward area etc. Obviously strategic information systems should have a mechanism of scanning and assimilating environmental information that are likely to influence and strategic decisions in a systematic way.

For the finance discipline Alternative Financing is a strategic decision. It will use a summary status information about the internal finances of the company including payroll, budget, overruns, variance analysis etc. but will be primarily governed by long term policy, business environment etc. that are strategically important to the survival and growth of the organisation. Information support for such strategic decision would call for substantial external information supplemented with internal financial health indicators.

Decision concerning Welfare Policy of an organisation is a strategic decision that must be faced by personnel (also known as human resource development) management. It will be tempered by internal information about staff size, their quality, the compensation package etc. but the strategic decision will be governed by the future vision for the Organisation as seen by the top management as well as the labour market conditions of the environment. Being a long term decision with major impact on the corporate health of the organisation this strategic decision need to be supported by a balance blend of external and internal information.

A strategic decision for the marketing function is a survey of competition and the resulting strategy of gaining market share. While a detailed analysis of in-company sales and advertising is a necessary pre-requisite, much will depend on the present and future strategies of the competitors who are external to the environment. Once again this brings to the focus the importance of external for strategic decision.

While we take primarily the decision making role of the management in the design of the information systems, care must be taken to take into account the stark realities of managerial work-style. The diversity of managerial work, the inter personal dynamics, politics of people, resistance to change, etc. must also be taken into account. These are issues beyond the scope of this text but must be taken into account in the actual implementation.

## 3.4.1 Data and Information

Early generation of textbooks in MIS used to elaborate in detail the definition of data and information and the important differences between them. With so many information systems in daily use and access even by the common man (like the Banking Information Systems, Library Information Systems, Train, Air, Bus-Ticketing and Reservation systems even in India), most people understand and appreciate that data is relatively raw and information is refined form of data that is more useful for human understanding and decision processes. The exact form of refinement may vary according to the needs and the nature of the applications. In some cases it is merely packaging in the form of a neatly formatted report, an eye-catching graphics on the TV or print media or a slide show presentation with brilliant effects. In some other applications it may be a form of summarizing data often using statistical tools and techniques viz. mean, median, standard deviation, probability estimates, approximation of distribution etc. In yet another situations, it may be a detailed simulation study using sophisticated modelling techniques. In all these cases the basic difference is clear - information provides insight into situations using data culled out of the processes that characterize the situations. Normally information systems should concentrate on information and not merely data.

From an information processing point of view there are some basic characteristics of data that must be kept in mind. Minimally the data must be accurate, timely and relevant. Several other attributes like reliability, source of data, consistency over time, value aspects including threat to individuals, society and to the world at large, privacy, protection of intellectual property etc. are important but are needs specific to some applications. For brevity we will elaborate only the former three attributes viz. accuracy, relevance and timeliness of data.

Data accuracy is obviously an important characteristic that any information system should guarantee to maintain. Inaccurate data-processed using analysis techniques however sophisticated in terms of tools is unlikely to be of use for any real world decision making. In computer jargon this is referred to as garbage in and garbage out (GIGO) rule. It is true that data collection costs money and more accurate data collection costs more money. To generate data up to an accuracy of tenth decimal digit may be prohibitively expensive. Yet large inaccuracies must be clearly avoided and a golden mean between cost of processing and value of accuracy must be found. There can be no general prescription about the absolute accuracy if data in general. Many monetary transactions get rounded to just two decimal digits (like Rupees and Paise in the Indian context and dollars and cents in the context of the U.S.). Such definitions of accuracy are based on the definition of currency. There may be no such equivalents in many other forms of data. Even a financial data like interest may have no fixed definition of accuracy. In a competitive business environment some bankers may compute interest rate to far more numbers of digits. In millions of transactions across the globe over electronic networks like SWIFIT, even a fourth digit after the decimal point might translate to millions of dollars! This aspect of the application specific accuracy should be borne in mind in the design of information systems. As another illustration of the needs of accuracy one can consider the time table information of the Indian Railways. The arrival and departure data of long distance express trains need be stored to an accuracy of five or ten minutes on even up to a quarter of an hour. However the time table concerned with commuter trains that operate over short distances within frequent intervals accuracies of arrival departure information may go down to a minute. During peak hours when the time gap between trains approach two minutes or so the arrival/departure information may even have to reach sub-minute accuracies! A related attribute of completeness is equally important. Extending the example of the train time table, information about the time and the time zone may have to be provided, in a train system operating over multiple time zones like EST, CST, MST, and PST in US or the time zones of different countries for trains that operate across the continent crossing several countries.

The second aspect of the data is the timeliness. Data however accurate if not available at the right time is of little use. Large scale use of computers in the early census calculations was primarily due to the power of computers to deliver data in a timely manner. Manual data collection of large census data, taken once in ten years. in many countries including India, does take more than a year if not many years. If the data of the decade spanning a ten year interval taken ten years to process, it would hardly be useful!. Thanks to the computers and the National Informatics Centre, for the first time in India, the census data of 1991 was available to planners in early 1993. In the early EDP era, there were several examples of data processing centers where enthusiastic data entry personnel used to process large, even accurate data often out of tme and time. With large improvements in processing times of computers over the past several years this is no longer a problem. Yet this aspect should not be lost sight of. In the Indian context, clearing tax dues, duty collections etc. take so long that by the time the tax evasion or duty avoidance is noticed the concerned entities may not even be there. In such cases either the speed of processing must dramatically improve or the data collected must be drastically pruned, and may be both the strategies must be attempted simultaneously. Otherwise the information systems will lack any practical utility. After all insight is more important than numbers.

The third aspect relevance is indeed the watch word for data. The emerging Decision Support Systems and Executive Support System that are characterized by end user computing underscores the importance of relevance in information systems. There is a growing feeling among a number of senior managers and end users that information systems professionals are able to provide accurate and timely information which are often not relevant; there is a whole bunch of information that the end users feel to be relevant but such information is not captured by many information systems. This mismatch once again underscores the importance of understanding organisations and the information needs of the organisation, which often may be outside the organisation, before designing information systems. Many library information systems provide selective dissemination service of the books and journals received recently in the library; often it takes months to prepare. What the users want often is the current contents of the journals, that are subscribed to by the Library as well as those not subscribed by the Library. They want information about all new books in their area, not necessarily those the Library has. This explains the role of relevance information systems.

The unprecedented developments in data capture technologies over the past few years had re-defined the role of data and information. The low cost auto identification technology of Bar Codes has totally done away with large scale data entry that was necessary to process a large number of transactions - sales data in super markets, issue and return of materials into and out of stores, libraries, drug-houses, hospitals etc. The smart cards will eventually replace Bar Codes and may lead to even further reduced data processing. Such developments also change fundamentally the design of information systems. A large number of examples of such automatic data capture and on-like access for verification in banking, insurance and government services (passport, driver license, crime records, voter identity card, Permanent Account Number for Income Tax etc.) will change information systems radically. Coupled with the Re-engineering exercise, one may witness such radical developments like Ticket- less Travel in aircraft recently introduced by South West Airlines.

### 3.4.2 Information and Management

Much of the management is decision making, according to one of many approaches to management. While there are several views of what constitutes management, according to the decision oriented view, management mainly comprises the following :

Planning
Organising
Coordinating
Directing and
Control.

Each one of these functions may be at the strategic, tactical or operational level. To illustrate this point we will use a series of examples of strategic, tactical and operational decisions and the information needs, in each of these functional areas.

**Planning:** Strategic level planning would call for a lot of environmental information like shifting markets, changing technology as well as internal information like core-competitive, strength of the organisation. Tactical planning activities like vendor development, make-or-buy decisions would call for cost and availability information pertaining to materials, production capacities both internal to the organisation as well as outside. Operational planning like staff scheduling would need large amounts of internal information like schedules, attendance, up- times of equipment.

**Organising:** Strategic Organising would need external and internal data to decide on re-structuring as well as forge strategic partnerships. Tactical organising would need changing wage level data of both the organisation as well as that of competitors. Operational organising would need data relating to skills and training requirements of the operational staff.

**Co-ordinating:** Strategic co-ordination would call for industry wide data corresponding to technology availability. Tactical co-ordination would call for plant wide and supplier wise bottleneck data which reflects the deficiencies both inside the organisation and outside. Operational co-ordination would require itemized break up of plant and machinery performance, failures etc.

**Directing:** Strategic directing functions like introduction of office automation would call for detailed cost, benefit analysis of new techniques. Tactical directing like innovating marketing strategy would call for detailed market and production data. Operational directing function would need data pertaining to the individual managers detailed skills.

**Control:** Strategic control decisions like total quality management would need detailed performance data and bench marking data from outside the organisation. Tactical control decisions like maintaining steady market share in the medium run would necessitate continuous monitoring of plant data. Operational control may call for techniques of statistical process control that involves the collection of substantial sampling information that must be collected and processed continuously during the entire production period.

In essence each and every area of managerial decision making would be it planning, organising, co-ordinating, directing or control, it calls for substantial amounts of information processing. While these functions of management need information support for decision making, there are subtle differences between the decisions that can significantly benefit from information; equally there are decisions that are unlikely to be benefited substantially by the availability of information. This subtle difference between the decisions, in a managerial context was very ably pointed out by the pioneering decision- theorist and Nobel Laureate Siman in his New Science of Management Decisions. According to Simon, all decisions cutting across the disciplines and levels of management can be classified into two types of decisions.

Programmed decisions are those that can easily be automated, like the determination of optimal product mix, minimum cost production schedule, optimal sequencing of machines to minimize mean flow time etc. Generally such decisions are characterized by large data and a few decision rules or algorithms that use the data in an automated fashion to arrive at an optimal plan. Techniques of Operations Research like Linear Programming represents a typical example of this category of decisions that use formal data and algorithms. Naturally such decisions are easily programmed. In other words they can be represented as an algorithmic procedure into unambiguous instructións whose step- by-step execution will lead the optimal result. Since these algorithms are likely to be codified in the form of a computer program and run on a digital computer, they care programmable or programmed decisions. The key to such programmability is the underlying structure of these decision situations that permit an algorithmic translation. By no means it is intended that such programmed decisions are unimportant, trivial or simple. There are no value judgements to such programmed decisions either. What is intended simply is an appreciation of the nature of cognitive or mental burden associated with such decisions. Many of the programmed decisions may need the most challenging algorithms involving the best brains available at that moment for their solution. Nevertheless they are translatable into algorithmic procedures. Simon's predictions of the fifties have practically come true in the nineties. This is borne out by the large number of decision support systems that use data and models to automate such decisions - re-order levels in materials and production control, control limits for process re-setting etc. Information support for such programmed decisions can be designed rather easily.

Non-programmed decisions on the other hand dō not lend themselves to easy automation. The model support for such programmed decisions would need more of heuristics rather than optimal algorithms. Optimization Algorithms as mentioned earlier also are formalized procedures readily implementable as a finitely terminating, computer programs with guaranteed outcome of optimal solutions, whenever they exist. Heuristics, on the contrary, generally yield near optimal if not the optimal solution but cannot guarantee optimality. The heuristics themselves may be implemented in the form of an algorithmic procedure and solved on a computer; however the decision situation may not have admit any optimal algorithm. The interesting developments in complexity theory of the theoretical computer science also led to an interesting observation that for many interesting optimization problems, heuristics are necessary as no optimisations algorithms with reasonable estimates of computational performance are known to exist. This idea led to the heuristic problem solving approach where one gets good solutions and even optimal solutions but optimality cannot be guaranteed. Such heuristic procedures for the solution of many real-world problems in the area of scheduling and resource allocation have been found to be extremely successful. Such heuristic problem solving vindicate a related theory, also proposed by Simon in the form of satisficing principle. Under this principle most decisions makers use a model where they fix initially a set of aspiration level; a level on the optimization criteria like profit for which they aspire. They identify strategies that take them to the pre-specified aspiration level. This re-evaluation process of the aspiration level stops after a while and the decision maker declares the best solution obtained in the process as the satisficing solution. Simon established that in many decision situations the decision maker uses the satisficing principle in place of optimizing principle.. Most of these decisions being to the non-programmed variety. Often times but not always, the tactical and strategic decisions belong to this category.

Information support for non-programmed decisions, naturally would be very different. The programmed decisions often use the algorithmic approach using high quality (accurate and precise), predominantly internal and low volume data. Non-programmed decisions, on the other hand, would often use unstructured and uncertain data. The programmed decisions like statistical process control would need one type of data support; the non- programmed decisions like the Just-in-time manufacturing process, Total Quality Management process would need entirely different type of decisions and naturally a very different information support. Over the years the programmed and non-programmed decisions have also come to be known as structured and unstructured decisions. Information systems designed should keep this vital difference between the two types of decisions in mind. Flexible access to data, user friendliness, graphical user interfaces, natural language support, what-if analysis, capability, etc. characterise information support for non- programmed decisions.

### 3.4.3 Information Support and Nature of Management

Office Automation Systems are characterised by repetitive, short-term, input-output oriented systems used directly by a large number of end users like clerks, typists, accountants etc. A large number of accounting systems like payroll, invoicing, billing, inquiry etc. belong to this category. These systems are more tuned to generation of information rather than the use of information. Office Automation Systems generally form the bed rock of all other information systems and the information generated by them become the corporate database. Information support for such systems must be simple, flexible and user- friendly. User friendliness in the Indian context would include provisions such as Multi-lingual support i.e. applications presented through Indian scripts like Tamil, Devanagari etc. The contribution of GIST (Generalized Indian Script Technology) and its widespread use in large systems like Railway Reservation System are prime examples of user friendliness in the office automation context. Many organisations world wide are moving away from IBM Mainframes and VAX minicomputers (that were mainly used upto the eighties for office automation) to PC based software tools like Word-Processing, Spreadsheet, Database, Business Graphics and Communication software. Standardization, Application Portability, Data and Format Compatibility etc. are the issues that must be resolved among various office automation system Components, particularly in large organisation. Use of standardized software for such purposes like Accounting is the current development in this area with an all time high emphasis on cost reduction. The emergence of integrated software and the bundled software bundles like Microsoft Office, WordPerfect Office, Borland Office suites along with electronic mail software like cc: Mail represent this trend of changing profile of applications in this area. The emergencies of work flow automation through software like Lotus Notes represent a still further integration of software over corporate network with many applications becoming mail-enabled. Integration with other office aids like copying, phone, fax, dictation, equipment, TV and other projection represent yet another development that will lead to exciting applications, changing the very concept of office in the organisations. Desk Top publishing and the emerging Desk Top Video including Desk Top Video Conferencing represent another office automation trend.

Transaction Processing refers to the traditional applications of computers like invoicing, billing, order entry, despatch, delivery, stores accounting etc. In all these application any single transaction like raising an invoice, accepting an order, shipping an item, receipt of an item, payment of a bill etc. would need an updating of multiple data stores (databases like accounts receivable, order status, pending list etc. A transaction can be said to be complete if and only if all the associated databases that are affected by this transaction are updated and all the updates are completed. It is important to emphasize the necessity of completing the updation of all databases before the transaction is complete. Any deviation from this would lead to major disasters as incomplete information may have serious consequences. Imagine the case of updating the line status of a telephone as disconnected but not updating the customer database to reflect the discontinuation of billing. The particular connection would be provided to some other customer and the old customer would keep getting the bill. More embarrassing example would be the updating of the guest status as booked. The next guest checking in may get the same room allotted to him by the hotel information system and the poor guest may slip into the room occupied already by the former guests! Naturally major corporate-wide information systems like Airline Reservations, Hotel Reservations, Library Automation Systems etc. have to reckon with this problem of transaction processing, and account for bulk of the applications. Many transaction processing systems also use very large databases running on large mainframes and minicomputers. They also call for very large processing of data by hundreds of users using equipment distributed over dozens of location spread over a vast geographical area. By their very nature, issues like performance, speed, accuracy, response time, backup, recovery, security etc., are extremely important in these applications. Naturally such systems receive substantial attention from the researchers and application specialists. Each one of the issues mentioned here is important enough to warrant several texts to discuss them in detail. For the sake of brevity, we shall discuss just a couple of issues.

Measurement of transaction processing speed and not merely the processing power of the processor is an important activity that must be continuously undertaken in any large scale transaction processing environment. The time taken to complete a transaction depends on the size of the databases, their distribution across the machines and/or across the network, the relative speeds of the processor, network, input/output device speeds like data transfer rate etc. as well as the nature of applications like simple query using one or more table, selective searching, searching and sorting, special processes like computation of statistical measures and the subsequent processing etc. An Organisation Transaction Processing Council has been developing a set of benchmarks in the form of TPC-A, TPC-B and TPC-C ratings for various combinations of databases, servers and networks. Constantly running these test suites, calibrating these test suites and actual benchmarking of the database applications constitute a major activity. Backup and recovery issues including disaster management (when data pertaining to thousands of users, transactions, bills etc. are lost, damaged or corrupted) is itself a major development activity. Security issues including encryption schemes that crypt the data and the converse schemes to decipher the encrypted data form another interesting research and development activity. Consequently information systems analysis should be able to deal with large scale, complex, organisation wide, mission critical applications in this transaction processing area that demands substantial hardware, software and manpower investments. Because of the dominance of transaction processing systems in many organisations, MIS professionals in these organisations associate information support only with this area, often at the cost of other areas. While this may be a default it should not be made out by design. Well planned information systems for office automation, decision support and executive support are needed with the same priority, though often as a small scale as well. In fact without the low level office automation systems in place, organisation wide transaction processing systems may not take off. Without some decision support systems in place many managers may not see much value for the transaction processing systems. The handful of executive support systems may be a positive factor to get top management involvement, commitment and investment approvals needed to continue the running of organisation wide information systems. In passing it may be pointed out that office automation and transaction processing systems capture much of the information needed for programmed decisions, described in the earlier section. Decision support and executive support systems to be discussed in the next set of paragraphs caters to the non-programmed decisions.

**Decision support** systems are intended to help individual managers in their decision making capability, involving generally the decisions belonging to the non-programmed category. Such system would need access to the large information generated by office automation and transaction processing systems However, what they need is mainly the summary information, exception information or some patterns and trends. To generate such summarised information like trends, patterns, exceptions etc. they may need access to analytical models. Since the decisions are non-programmed the exact report, formats and the contents of decision support systems cannot be decided 'a priori'. The end users would, therefore, need tools that enable them to generate on the fly reports with ease in a flexible manner. Since the end users are likely to be conversant with but not skilled in programming, the interaction will have to be at the higher level of abstraction. Fourth Generation Languages, Menu Driven Packages, use of point and clock or pull down menu based access, form based database access (with a flexibility to paint screens easily with different data, format etc.) are likely to be more easily acceptable. The user must be convinced that he/she is using his/her time productively to get insights into the decision processes without spending too much time in learning the commands of the computer packages. Related features that are generally demanded by decision support applications are the provisions for graphics (generation of different type of charts), integration with other modules (presentation software, slides preparation etc.). In short information for decision support calls for summary information processed in innovative ways in a flexible manner by end users directly. Decision support generators like IFPS (Interactive Financial Planning Systems), to be discussed in the next chapter, were designed primarily with these features in mind.

**Executive Support** Systems provide yet another challenge to the information systems professional. These systems are meant to be used by senior managers directly to provide support to non-programmed decisions in the strategic management category. The information needed will be largely external, unstructured and even uncertain. The exact scope and context of the information needs is not likely to be known a priori, to the analyst. The requirement is for intelligent information market intelligence, investment intelligence and technology intelligence. This must often be collected and presented to the decision makes in a pro-active manner. Often the information is likely to be spread throughout the organisation as well as in the outside environment. Access to external databases, technology information like patent records, technical reports by consultants and consultancy organisations, market report by market intelligence agencies, confidential information regarding competitors, speculative information about markets, sources of financial information, like government policies etc. being to the category of information that must support executive decisions. One can easily appreciate the extent of the lack of any structure among this diverse information needs. Since executives often work in groups, group communication, brainstorming etc. are often needed. Executive support systems, therefore, provide the highest levels of challenge for the information systems design. Tools of Artificial Intelligence (AI) and Export Systems (ES) are often used to cope with some of the complexity of executive decision making. Natural Languages Support to provide a natural, intuitive and less demanding interface to databases and other sources of information is yet another but difficult requirement of some of the executive support systems.

In the past few paragraphs, we have argued about the need of the analyst to appreciate the differences among office automation, transaction processing, decision support and executive support systems. This is absolutely necessary as the requirements of quality of information, quantity of information, access strategies, presentation of information all vary substantially across the different systems. The strategies, skills and resources necessary to succeed in one area may not necessarily lead to success in other areas. The careful analyst should arm oneself with all the necessary tools and apply the most appropriate tools for the situation. It may be noted that every organisation will have some elements of all the four information processing systems; only the extent and relative proportions vary. For reasons of brevity, this will not address all the techniques of each of these systems. Being a basic text, own intention is to clarity the basics. The interested reader may refer to advance texts in each one of this areas.

## 3.5 MIS AS A PROFESSION

What constitute a MIS professional? After going through the spectacular growth in the Information Technology (Unit 2) one might tend to feel that a good computer scientist can make a good MIS professional. Quite often such an assumption is not true. Developing information systems call for people interaction, organisational understanding and inter personal skills that are generally not the thrust areas of the computer science professional. Keeping this view in mind ACM Curriculum Recommendation [] clearly outlines a separate profession with a mastery of organisation, viz. the Information System Executive. This text is clearly addressed to such professionals with a strong back ground in technical computing.

## 3.6 SUMMARY

This unit provides the organisational context of MIS. After reviewing the resource view of information, the emerging role of MIS as a possible competitive weapon is outlined. This is followed by a possible profile of an MIS professional.

## 3.7 SELF-ASSESSMENT EXERCISES

1) What is meant by "Information Revolution"?
2) How can information provide "Competitive Advantage"?
3) Explain in your own words the terms "Organisation Pyramid" and "Information Pyramid".
4) What are the major attributes of an MIS professional?

# UNIT 4 BUILDING MANAGEMENT INFORMATION SYSTEMS

**Structure**



## 4.0 INTRODUCTION

This final unit outlines some of the steps involved in building MIS. A detailed study of the techniques would be taken up in a separate Block (System Analysis and Design). This unit is intended to provide a quick overview and to enable the reader to appreciate the role of System Analysis tool in the broader MIS context. Such a coverage will provide an overall perspective to the broad area of MIS, the subject matter of this block.

## 4.1 OBJECTIVES

After going through this unit, you should be able to:

- focus on the steps that form part of systems analysis that have special relevance to MIS
- appreciate the need for 're-engineering' an organisation rather than merely computerising existing manual procedures
- realise that development of management information systems is fundamentally different from programming
- focus on the tools and techniques used in actually building real world management information systems.

## 4.2 SYSTEM ANALYSIS

Building computer based information system needs a careful analysis and design before implementation. Detailed study to understand the information requirement forms the major component of the analysis. Providing a system that would meet the information requirement in an efficient manner is the subjects of information system design. With the increased availability of new products and services ever-improving price performance ratio, such analysis and design become far more challenging. The newer products and services provides new ways of daring business as well. While the conventional analysis of information system requirement may indicate some possible alternatives, an alert analyst who is abreast of current and emerging developments in information technology may propose entirely new systems. Such exercises known as business "re-engineering" are the by products of good systems analysis and design. It should be clearly understood that good systems analysis is not merely an efficient way of getting computers to do what has been done manually for years. Such mere mechanisation fall short of the real objectives of good systems design, particularly these days when Information Technology is moving leaps and bounds.

The techniques of systems analysis are the ones that get covered in courses an MIS. While the techniques are definitely important, it must be clearly borne in mind that systems analysis is fundamentally different from programming. In the opinion of the author much of the ills of the customers getting obnoxious university results and atrocious utility bills, reported time and again in the Indian media over the past few years precisely arise out of programming efforts without going through the necessary systems analysis. As an analogy, systems analysis can be viewed as the job of an architect, while the job of programming of comparable to that of construction. Naturally systems analysis is for more conceptual in nature as contrasted to programming that is one of details. It is no way to belittle the intellectual process that can be demanded by programming profession. Programming is clearly analytical in nature but system analysis is far deeper conceptually. The input of system analysis naturally is far reaching compared to immediate impact of programming. Yet another way of distinguishing between these two different, yet related disciplines is that programming is skill oriented but systems analysis is more knowledge oriented. Systems analysis calling for very generous skills in limited to a handful of successful individuals; while programming skills being specific are easily acquired by fairly intelligence people. Naturally mass producing programmers is fairly easy compared to training a large number of analysts. In summary systems analysis is holistic in nature while programming is individualistic in nature.

Such detailed comparative study of programming and systems analysis was necessary to reinforce the fact that building information system gives far beyond simple programming and coding. With a substantial emphasis laid an programming (thus street corner shops to train people in BASIC, FORTRAN, COBOL & C) in the Indian scene, this point can never be over emphasized. Teaching programming to develop on intellectual maturity is perfectly fine. This is similar to training in basic mathematical skills. But simply providing training in programming and expecting people to develop information systems in just not the right thing to do.

The steps in system analysis include the following eight steps generally known as systems life cycle.

1. Requirement Analysis
   (a) Determination
   (b) Specification
2. Feasibility Analysis
3. Basic system specifications
4. Hardware/Software Study
5. System Design
6. System Implementation
7. Evaluation and Acceptance
8. System Maintenance

(for a detailed study of the activities involved in each of the steps the reader is referred to the excellent text [Rajaraman p. 29].)

## 4.3 TECHNIQUES OF SYSTEMS ANALYSIS

Over the past decades, based on extensive empirical studies of the different steps of systems analysis enumerated in the last section, several techniques have emerged. We will briefly look at some of the fundamental techniques.

### 4.3.1 Requirement Analysis

Requirement Determination is generally done though extensive study of the system including the understanding of the goals, processes and constraints of the system for which information systems are designed. Several forms are also designed and illustrated in the texts of system analysis. In the view of the author such techniques must be left to the ingenuity of the analyst; there is no straight forward algorithm to elicit the requirement from the user. It is an iterative process which the analysts use while interviewing several user/users groups. It will continue to remain an art rather than science.

For Requirement Specification both at the preliminary as well as the detailed stage, several diagramming techniques have evolved. In fact such diagrams have become the language of the analysts just as blue prints have become the language of the designer or balance sheet becoming the language of the accountant. We will detail below some of the diagramming techniques.

### 4.3.2 Diagramming Techniques

Data Flow & Document Flow Diagrams represent perhaps the most widely used diagramming technique of the systems analysis. The **document flow diagram** graphically represents the various documents (typically paper developments in most cases) that flow across the system; the information carried by the paper documents must be generated and processed by the proposed information system. Often the documents themselves may continue in paper form for administrative control, archives etc. as the necessary imaging techniques are still not cost effective, at least in India. Given the constraints of electrical power, paper may continue to be the major source of document, for many more years to come.

**Data Flow Diagram** (DFD) is a powerful diagram that can be used to document the information flow. It also presents itself to be broken down in top-down fashion. At the top level, data flows are represented at very abstract aggregate level. Each component of the data flow is further broken down to different levels, so that at each level we have just a few entities to concentrate on. DFD have developed a representation scheme to represent **data store** (storage & retrieval of data), **processes** (where some changes are made to the system) and **entities** (the player in the game) and the actual information flows.

The example data flow diagrams explains the concept. Breaking down the DFDs into different levels is generally known as Leveling and the DFDs are correspondingly termed Top level DFD and Leveled DFD.

A diagram similar to DFD is the **context diagram** that is generally a summary diagram that fully depicts the information flows and the entities involved. It is more useful to the end user than the analyst. The detailed DFDs carry enough details that the analyst can hand them over to the programmer for detailed coding.

### 4.3.3 Data Dictionary

Another powerful tool that is extensively used is system analysis is the data dictionary. DDs as they are called provide a detailed reference to every data item - the different names by which the item is represented, in different program modules, different data structures used to represent the item in different modules, the modules where the data item is generated, where it is stored and destroyed. In essence it provides a quick snapshot of every data item used by the information system.. Needless to say it is extremely detailed and very useful for consistency checks, system modification and completeness checking.

A typical data dictionary appears as follow :

1 Page Data dictionary (*** ??? ***)

While these techniques are general in nature and used by the analyst in the different stages of the system life cycle the following are specific to some of the steps of the system life cycle.

### 4.3.4 Feasibility Report

A typical structure of the feasibility report will be as under:

A preamble that sets the stage for the project, followed by *goals* statement that quantifies precisely the goals of the proposed information system. This is followed by a short narrative that describes in unambiguous yet jargon free language the present system. This must be understandable to any intelligent person not necessary a computer professional or even a computers literate. The proposed alternatives are then described once again in a reasonably jargon free language. Being a feasibility study the alternatives are unlikely to be detailed to the full extent. Until the full system is developed in its entirety, the full details are unlikely to be known. Yet we cannot go ahead with the final system without doing a feasibly analysis. In a sense the details of the system to be built will unfold gradually from feasibility study stage to system specification stage to final implementation stage. These steps must be clearly understood by the user as well as the analyst. Based on 'sketchy' design of the proposed alternatives, an order of magnitude cost benefit study is prepared. It will be order of magnitude only as the costs and benefit are also unlikely to be in fully quantifiable form. Based on the detailed human/organisational/technological problems likely to be encountered in the project detailed in the feasibility study the end user decides a particular alternative that is worked out in detail for further implementation. The detailed design phase starts here.

### 4.3.5 Detailed Design

Roughly the detailed specifications are worked out followed by hardware/software plan. This constitute system design which once again need to be whetted by the user. Once this is done detailed system design starts. Effectively one can say that the analysis phase ends here and the design phase begins. Being a detailed design it may involve substantial effort on the part of technical system analysts, hardware, software, communication specialists etc. A major component of detailed system design is the database design covered in the next section. Actual coding is undertaken after the database design is complete.

### 4.3.6 Database Design

While DBMS permit efficient storage and manipulation of data files they do not cater to the structuring of the database themselves. After extensive uses of database in real world applications, several early analysts have felt the need for the right abstraction of data into the database so that any update/query operation captures the spirit of the meaning of the data stored in the data bases. One of the basic observation made by the early analysts was the possible "loss of information" by careless update operations on databases. This led to the concepts of normalisation pioneered by Codd []. Intuitively normalisation leads to the decomposition in such a way that no information is lost due to processing of data. Normalization ensures no loss of information and avoids insertion update and other anomalies. A table (relation) is said to be in First Normal Form (1NF) if there is an identifying key and there are no repeating groups of attributes. Intuitively First Normal Form ensures that all the table entries are atomic. A formal definition of a relation will be as follows:

A relation is in (1NF) iff attributes of non-key fields are dependent on a key.

For example, consider a student table with data as follows:

{Student_Name, {Course_No., Course_Name},
{Course_No., Course_Name},......}

with repeating groups. Obviously such a varying field leads to blank fields (that do not depend on the key). A simple way to transform a relation that is not in (1NF) is to replace it with a simple relation with as many records as the non-full repeating groups. For example,

Student_Name, Course_No, Course_Name
Student_Name, Course_No, Course_Name
...
...

for as many courses as the student has. A further refinement of the First Normal Form (INF) is the Second Normal Form (2NF) where we avoid any dependencies among the relationships. For example in the earlier example course name depends on Course_No which is not a key field. Such a structure may lead to inconstancy where there could be different Course_Name combinations for the same Course_No. To ensure that the relations (tables) are free of such inconsistency we break down the relation further into

{Student_Name, Course_No.}　　{Course_No, Course_Title}
{Student_Name, Course_No.}

A still further refinement is Third Normal Form (3NF) where we further specify that there is an identifying key, no repeating of attributes, no attribute exist that does not require the whole of the key and there is no transitive dependency (i.e., an attribute depending an another non-key attribute) i.e., in general attributes are mutually independent.

The benefits of normalisation are the avoidance of update anomalies explained through the following example :

Example　(**** ???? ****)

Note that the 1NF, 2NF, & 3NF are upwards inclusive, that is 2NF relation is necessarily in 1NF and a relation that is in 3NF is necessarily in 2NF. They represent progressively increasing refinement of data relationships represent.

Database theory details further degrees of normalisation including 4NF and 5NF. While theoretically sound, such further refinements add(?) like, if any, to data modeling real world data. Since our text is primarily on Information System and not on Database theory we will not further elaborate an advanced normalisation. It may be mentioned in passing that a slightly modified 3NF known as Boyce-Codd Normal Form (BCNF) lends itself to automatic decomposition from an un-normalised form using Bernstein algorithm [].

Physical design of database includes the storage devices and their management. This calls for detailed knowledge of the specific hardware and software on which the information system is likely to be implemented. Hardware/Software Selection is indeed a difficult job considering the fact that many competing products are being introduced in the market with improved benefit cost ratio. An alert analyst has to keep himself posted continuously about the trends in technology, obsoleteness of technology, viability of emerging technology, future trends and even the financial and market viability of the hardware/software vendor in this fiercely competitive area of Information Technology. Access to specific experts in areas, use of bench marketing literature and user group and peer opinion are some of the resources used by most analysts in this difficult task.

### 4.3.7 System Implementation

System implementation includes the detailed design of the process, their validation and thorough checking. While the formal methods of proving program correctness [] are evolving, they are still not useful to test out large commercial software to help information system planning. Many of the analysts use experimental version using what is known as "parallel runs". Here both the current system and proposed new system are run in parallel for a specified time period and the current system is used to validate the purposed system. (This is like the practice of the incumbent senior officer spending some time with the outgoing office before actually taking over the full responsibility.)

## 4.4 SUMMARY

This module provides a quick introduction to the techniques of system analysis and design that are to be used in actually building real world management information system. Where the full details will be shown in details in a separate course, this unit is intend to provide a quick overview of all the issues involved in MIS development. Accordingly many of the tools System Analysis are only touched upon.

## 4.5 SELF-ASSESSMENT EXERCISES

1) Outline the steps of system analysis.
2) Understand the difference between programming and system analysis.
3) What is a Data Flow Diagram (DFD)?
4) Briefly describe the structure of a Feasibility Report?
5) Explain the role of Normalisation in relational database design.

Uttar Pradesh
Rajarshi Tandon Open University

# BCA-07
# Elements of Systems Analysis and Design

Block

# 5

## Case Studies

## BLOCK INTRODUCTION

By now, you would have had some exposure to the fundamentals of computer hardware and software. The usefulness of computers to organisations come about only when they can be gainfully employed to facilitate the working of the organisation and help streamline its operations.

The subject of systems analysis and design addresses itself to this very issue. It is an intensively practical subject and the analysts must be able to translate his technical knowledge into real-world applications. The preceding blocks in this course on systems analysis and design have set the pace for what is conventionally taught under such a subject.

However, in order to have a flavour of the kind of work system analysts do, and the surroundings and constraints within which they must perform their functions, it is desirable to work through several case studies. This block, therefore, presents a series of case studies numbered as A to F which give such as flavour.

The organisation in question is a gymnatorium, which is trying to modernise its operations, and finds that mere investment of money is not enough. You can go through these case studies at a number of stages of the course, in a manner of what is called "Spiral Learning". In other words when you read the same case again after a gap of sometime, you will be able to appreciate it better and on a higher level. To facilitate your understanding, each of the case studies A to F also carries with it a model solution. These solutions are indicative and not expected to close your mind to further thinking. Nor are you expected to memorise the solution and present them in any examination. They are for you to understand it better so that you can handle the assignments, the projects, the examinations, and more important actual real life problems in your employer's organisation after you have completed the course.

You will notice that the key players in these case studies are about 20 in number and all of them play a significant role in the implementation of the new system. This is to specifically draw attention to the fact that creation and development of computer based systems to help any organisation is not the domain of the programmer alone or the EDP department alone.

For your convenience the dramatis personae in these case studies have been listed below with their functional titles.

If you follow this as you read through the case studies it will keep you aware of the various interactions between different persons in the organisations.

**DRAMATIS PERSONAE**

| | | | | Case and Frequency of Occurrence |
|---|---|---|---|---|
| 1. | Mukul Gupta | : | Director, MCG | A-6, B-6, C-1, D-2, E-1, F-4 |
| 2. | Neera Tiwari | : | EDP Manager | A-1, B-5, C-1, D-2, E-1, F-3 |
| 3. | Anoop Sarin | : | Head of Scheduling and Ticketing | B-14, C-1, E-1, F-1 |
| 4. | Sarita Kakkar | : | Chief Systems Analyst | B-11, C-8, D-8, E-7, F-1 |
| 5. | Anil Abraham | : | Systems Analyst | B-12, C-6, D-8, E-3, F-2 |
| 6. | Pran Sharma | : | Member Management Committee | B-1 |
| 7. | Feroze Ticketwallah | : | Head Cashier, Ticket Sales | C-3, F-1 |
| 8. | Suresh Srivastav | : | Accounting Manager | C-1, F-1 |
| 9. | Dinesh Malik | : | Activities Manager | C-1, F-1 |
| 10. | Balwant Singh | : | Facilities Manager | C-1, F-1 |
| 11. | Nasrin Khan | : | Chief Accountant | C-1, F-1 |
| 12. | Dilip Dalal | : | Publicity Manager | C-1, F-1 |
| 13. | Sanjay Sharma | : | Prog. Coordinator | C-1, F-1 |

| | | | |
|---|---|---|---|
| 14. Tirath Kumar | : | Assistant | C-1, F-1 |
| 15. Keshav Dutt | : | Assistant | C-1 |
| 16. Minakshi Dutt | : | Ticket Seller | C-1, F-1 |
| 17. Rohan Kumar | : | Ticket Seller | C-1, F-1 |
| 18. Samir Rawat | : | Asstt. to the Program Coordinator | F-1 |

# CASE (A)

## MAHARAJA COLLEGE'S GYMNATORIUM (MCG) INFORMATION SYSTEMS PLANNING

## THE SCENARIO

Mukul Gupta, the Director of Maharaja College's Gymnatorium, sits at his desk wondering how to develop solutions to the problems facing the gymnatorium (gymnasium cum auditorium). Since the expansion of the gymnatorium was completed a year ago, the number of programs conducted have doubled, as college management wants to capitalize on the additional facilities available. Since the college is new and management has modern outlook the gymnatorium is run as a profit centre and has various divisions to look after its numerous activities.

The information needs were easily satisfied before the expansion when the gymnatorium hosted only basketball games, an occasional Youth Forum meeting, and three to four cultural programs a year. Additional building space and seating has attracted the promoters of rock concerts, conventions, and professional matches to the gymnatorium.

Recently several programs have almost ended in disasters because of coordination problems. On one occasion the advertising division learned only a few days before the program that a souveneir to be sold at the door had not been produced. Although the printing was completed on time, the program resulted in a loss because the rush delivery charges levied by the printer were large, and the advertising division was unable to solicit the required number of advertisers for the souveneir.

For every event, the facilities planning division complaints about problems encountered preparing the gymnatorium for a program. Requirements for the program, such as the seating arrangement and the number of chairs and tables needed, often are not communicated in a timely fashion. The facilities planning division has found itself short of chairs, tables and other resources because two programs events are scheduled in different parts of the gymnatorium at the same time, each competing for the same resources. Personnel division has had difficulty in scheduling ushers, attendants, and ticket checkers. To prepare properly they feel that they should be notified as soon as the program is booked by the activities division. The lack of information about the resources available has caused the activities division to take up to a week to inform all other divisions of the requirements for a given program. Even the accounting division has had problems paying the promoters because of delays in receiving information concerning ticket revenue and expenses.

The worst problems seem to be in the ticketing division. The ticketing division handled only general admission tickets before the expansion. Now, the division provides tickets for reserved seating, special seating and general seating for different kinds of programs that have varied seating plans.

## GETTING STARTED

Mr. Gupta knows he has to address these problems. The Managing Committee of the college approved the expansion of the facilities assuming that the gymnatorium would host more programs, increase the variety of programs, and show a profit. At present, the gymnatorium is losing money and credibility with the promoters of programs. Staff members are also unhappy because they feel that the problems are a direct result of their performance. Mr. Gupta, indeed, recognizes that the only way to solve this growing problem is to take action to meet the increased information requirements of the gymnatorium.

# PLANNING FOR INFORMATION SYSTEMS

Mr. Gupta realizes that the planned goal of increasing the business operations of the gymnatorium has been met since expanding the facilities. They could meet the goals set forth by the Managing Committee much more efficiently, only if they could achieve better communication between the various division in the gymnatorium. An information system that integrates all the division of the gymnatorium would provide the communication necessary to meet this requirement.

Mr. Gupta sends a memo to all his division heads asking them to meet with him and Neera Tiwari, the EDP Manager, next Monday for a strategic-level conference about the development of a new information system. He also requires each division head to bring systems project request forms (attached to his memo) filled out, ready for presentation and prioritization.

On Monday, with aid and direction from Neera Tiwari and several assistants, the division heads and Gupta hammer out a systems plan after some heated discussions. See Exhibit A-I for the modular design of an integrated information systems project for the gymnatorium.

After the brainstorming that went into preparing the information systems plan, a consensus is reached that the highest-priority projects will solve immediate critical problems. The others of lower priority will fulfill long-term management goals. Based on this philosophy various ratings are provided on a 10 point scale for TELOS (Technical, Economic, Legal, Operation and Scheduling) feasibilities this listing is consolidated in Exhibit A-II.

**EXHIBIT A-I**

**LISTING OF PROPOSED MODULES**

**A. SCHEDULING AND TICKETING SYSTEM (STS)**

1. Integrate the activities and ticketing functions.
2. Provide the activities office with a system for setup and management for each program.
3. Provide the ticketing with a system for issuing tickets and provide the activities office status on demand for each program.

**B. FACILITATING AND SEATING SYSTEM (FSS)**

1. Provide information to the facilities division necessary to prepare the gymnatorium for each program.
2. Provide feedback to the activities office about availability of resources and progress reports on facility set up for upcoming programs.

**C. ADVERTISING AND PRINTING SYSTEM (APS)**

1. Provide information about printing and advertising needs to the advertising department to prepare for each program.
2. Provide feedback to the activities office on the status of advertising and printing efforts.

**D. PERSONNEL AND SCHEDULING SYSTEM (PSS)**

1. Provide information about required staffing for each event to the personnel division to schedule staff.
2. Provide feedback to the activities office on staffing and scheduling status.

**E. BOOKING TRACKING SYSTEM (BTS)**

1. Provide the activities office with the information necessary to contact representative for programs scheduled.

2. Provide the activities office with the information necessary to manage current booking at the gymnatorium.

**F. ACCOUNTING AND TRACKING SYSTEM (ATS)**

1. Provide accounting information to all levels of management in a timely manner.

2. Provide accurate reporting for all financial data to Maharaja College Central financial management division.

**G. PROGRAM CATALOGING SYSTEM (PCS)**

1. Provide master catalog of types booking criteria and other information for all programs which may potentially be booked in the future.

2. Provide cost information for all potential future programs.

| PROJECT NAME | STRATEGIC FACTORS | | | STRATEGIC FACTOR SCORE | FEASIBILITY FACTORS | | | | | FEASIBILITY FACTOR SCORE |
|---|---|---|---|---|---|---|---|---|---|---|
| | * PROFIT | ** SERVICE | *** DECISION MAKING | | TECHNICAL | ECONOMIC | LEGAL | OPERATIONAL | SCHEDULE | |
| A. Scheduling and Ticketing System (STS) | 9 | 8 | 7 | 8.0 | 8 | 8 | 9 | 7 | 9 | 8.2 |
| B. Facilitating and Seating System (FSS) | 5 | 6 | 7 | 6.0 | 6 | 3 | 9 | 7 | 5 | 6.0 |
| C. Advertising and printing System (APS) | 7 | 4 | 3 | 4.7 | 8 | 4 | 6 | 7 | 5 | 6.0 |
| D. Personnel and Scheduling System (PSS) | 2 | 2 | 2 | 2.0 | 8 | 8 | 7 | 7 | 9 | 7.8 |
| E. Booking Tracing System (BTS) | 4 | 5 | 7 | 5.3 | 7 | 3 | 9 | 3 | 1 | 4.6 |
| F. Accounting and Tracking System (ATS) | 8 | 4 | 6 | 6.0 | 2 | 1 | 7 | 2 | 4 | 3.2 |
| G. Program Cataloging System (PCS) | 1 | 5 | 3 | 3.0 | 2 | 3 | 2 | 4 | 9 | 4.0 |

EXHIBIT A-II
PROJECT REQUEST PRIORITY WORKSHEET

* Profits : Increase Revenue and Decrease Expenses
** Service : Provide Organized and Problem-Free Programs
*** Decision Making : Coordination Between Divisions of MCG

## ASSIGNMENT ON MCG CASE (A)

1. Prepare a mission statement and systems goal list based on outline of MCG's information systems plan.

2. Discuss the validity of various FACTOR rating given by Mr. Gupta and his information systems planning team.

3. How can Neera Tiwari, the EDP Manager can prioritize the various modules listed for analysis and design?

ANSWERS TO ASSIGNMENT

Ans 1

Mission Statement

To develop an information system that will increase the effectiveness and efficiency of the planning and controlling functions of the gymnatorium by providing more timely, relevant and accurate information to managers and their staff.

Goals Listing

1. Better communication between the various divisions of MCG
2. Adequate reporting to each division.
3. Centralised management reporting and decision support
4. Accurate and timely control of programme
5. Timely reporting of accounting activity
6. Efficient and controlled ticket handling
7. Cost-effective system

Ans 2:

Factor ratings are clubbed into two broad categories viz. Strategic Factors and Feasibility Factors. Strategic factors include; profits, service and decision making whereas the feasibility factors comprise of TELOS—Technical, Economic, Legal, Operational and Scheduling.

Let us take example of STS module listed as A in the Exhibit A- II. On strategic factors it scores 9, 8, 7 on a 10-point scale. High strategic factor scores were recommended by Gupta and his team because this module has strong influence on strategic factors of profit making, providing service and quick decision about the program selection. All these factors are essential for a business to succeed in a competitive market (similar discussions can be done for other modules also).

On TELOS feasibility also STS module consistently seems high resulting in an average score of 8.2. The reason again in Mr. Gupta's mind is that STS is justifiable in terms of TELOS feasibility and is executable in terms of technical competency has sufficient funding and meets legal, operational and scheduling constraints.

Ans 3:

To prioritize Neera can utilise a system project priority grid (a concept akin to BCG matrix analysis in marketing).

This grid is shown as below:

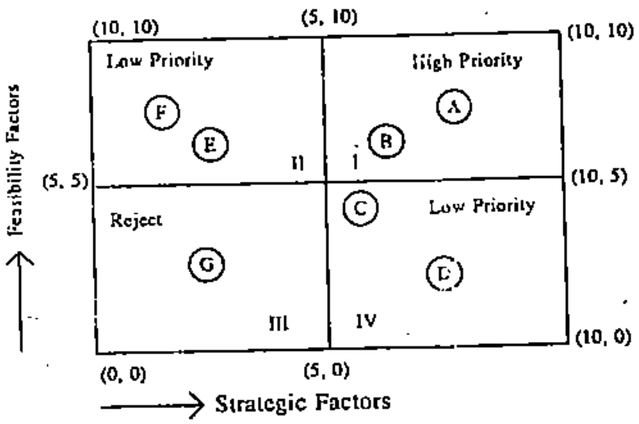



Figure A-1: System Priority Request Grid

In this Grid projects in quadrent I are of immediate priority, ones in quadrents II, IV can be taken up slightly later and the ones in quadrant III can be put on blackburner.

# CASE (B)

## MAHARAJA COLLEGE'S GYMNATORIUM (MCG)
## PREPARING FOR SYSTEMS ANALYSIS

## GETTING APPROVAL FOR SYSTEMS DEVELOPMENT

The accounting division with guidance from Gupta and Tiwari, prepared a cost estimate of Rs. 20 Lacs to support the Information Systems plan. Mukul Gupta then requested sanction for this amount to support systems development effort. Realizing the urgency of this request, the Management Committee immediately allocated Rs. 12 Lacs, less than what was requested, but possibly enough to complete some of the higher-priority modules over the next year.

Mukul Gupta was elated and quickly got in touch his divisional heads and Neera to tell them the good news. The budget of Rs. 12 Lacs enables a substantial portion of the systems plan to be funded.

Then Mr. Gupta gave a general directive for systems work to begin immediately starting with the scheduling and ticketing system, and within two weeks, also the modules B, C and D. Mr. Gupta also had a meeting with Mr. Anoop Sarin, Head of Scheduling and ticketing department. He was told to get his people to interact with systems people first thing in the morning.

## REQUESTING SYSTEMS ANALYSIS

Mr. Anoop Sarin called Neera Tiwari over the intercom and told her, "Well, it looks like we are ready to get started on my scheduling and ticketing system." "Yes, I heard, "Neera said, "What I'm going to do is to ask Sarita Kakkar, our chief systems analyst to get in touch with you. She will be able to get us off to a good start. Why don't I transfer your call to her office?"

"OK, thanks."
"Hello, Sarita Kakkar speaking."
"Oh, yes. Hi! I'm Anoop Sarin, Head of scheduling and ticket sales at the gymnatorium. Neera told me that you are going to do our systems work. Great to have you on our side.

"Thanks, I'm looking forward to working with you, "Sarita responded. "Actually, Anil Abraham, one of our systems analysts is all set to be with you this morning and prepare an information systems service request."

"Oh. I've already filled out one of those forms, "said Anoop.
"You know, the one we did for planning."

"No, that was a systems project request form that you people use for planning and for budgeting purpose, "responded Sarita. "The information systems service request form that you fill out for us is just that, a request from you for our service. It's a little more specific than what you did earlier, and it gives us a starting point and serves as an agreement, if you will, "between our users and the information systems group. It's a 'meeting of the minds,' so to speak."

"Another form?" complained Anoop, "I don't know anything about it or how to fill it out. And we just started using the systems project request form."

"Well, you know us, we couldn't operate without forms, "laughed Sarita. "Don't worry, Anil Abraham will help you fill it out. Then we can get started."

"Thanks, that will help me a lot," sighed Anoop, mentally relieved that he was going to receive help.

"By the way, "Sarita said, "As soon as my staff and I go over your information systems service request, we will start mapping out our work. One thing we will need to do at the start, I'm sure, will involve a great deal of interviewing of you and your people. Will you please make them available to us starting Tuesday at 8.00 AM.?"

"We're all set, "exclaimed Anoop.
"Fine. I'll see you Tuesday morning, "Sarita responded.
"Good-bye."
"Bye."

## PREPARATION OF THE INFORMATION SYSTEMS SERVICE REQUEST FORM

Shortly after 10:30 that morning, Anil picked up his briefcase and went to Anoop's office.

"Hi, I'm Anil Abraham, one of the systems analysts from EDP department. I guess Sarita told you that I was coming to see you, "said Anil.

"Oh yes, pleased to meet you, "said Anoop. "I was on the phone with Sarita a little earlier this morning, and she told me that you were coming."

Great!" replied Anil. "If you're ready, we'll get started."

"Well," responded Anoop. "I've never been involved in a complete information systems overhaul as we're doing here. Where do I begin?"

"What would be helpful, "Anil explained, "is if you could start by telling me some of your basic and common problem."

"To begin with, "Anoop started, "we are having trouble with our seating system. There are times when the same seat is sold to at least two different people. All of us in this department feel that once is once too often and we try to do our best to keep the public happy. But this is now a common problem."

"So, you are having a ticket sales problem," said Anil.

"A problem?" exclaimed Anoop. "It's becoming a catastrophe! Some of our ushers came back with reports from several people recently that this was the worst program venue they had ever been to. Another thing, I've had five people in the last month personally call me up and tell me that they were sold tickets for the wrong program! I looked into the matter, and sure enough, they were sold tickets for the wrong program."

"Oh! God" remarked Anil. "I can see exactly what you mean. That would be enough to make practically anyone want to throw in the towel. You are having big problems with your scheduling of events and ticket system. Why don't we call the first project Scheduling and Ticketing System (STS)."

"Sounds good to me," said Anoop. "STS.....I like it."

"Why don't we get started on the information systems service request form?" prodded Anil.

"OK, I'm ready. What do you need to know first?"

Anil and Anoop spent over an hour filling out the information systems service request form. It is displayed in EXHIBIT B-I.

## PREPARATION AND PRESENTATION OF THE PROPOSAL TO CONDUCT SYSTEMS ANALYSIS REPORT

After going over the information systems service request form prepared by Anoop and Anil, Sarita talked with Anoop and some of his staff to get a better feel for the problems, reasons for the request and scope of the project. She and Anil worked together to define the systems project scope, to determine the study facts they would need to collect, and the sources of these facts. A schedule of major events was also prepared..

**EXHIBIT B-I Information systems service request form for INFORMATION SYSTEMS SERVICE REQUEST**

SYSTEM NAME: Scheduling and __x__ NEW<br>Ticketing System (STS) __ REVISED

REQUESTED DATE:MM/DD/YY<br>REQUIRED DATE: MM/DD/YY

**ANTICIPATED BENEFITS:**

* Schedule with more accured and control ticketing for programs for gymnatorium.
* Control the issuing and reserving of tickets.
* Account for and control ticket revenue.

| OUTPUT | INPUT |
|---|---|
| 1. TITLE: Program Schedule Report<br>FREQUENCY: Weekly quantity: 4 copies<br>COMMENT: Lists name of program, cost, date and duration, Ticketing queries about the same items should be provided. | 1. TITLE: Program Input<br>FREQUENCY: Random Quantity: 200 per year<br>COMMENT : Includes seating requirements for program beginning date, ending date, and special accommodations |
| 2. TITLE: Reservation Status and Ticket Control Report<br>FREQUENCY: Weekly quanity: 4 copies<br>COMMENT : Lists reservation by name and event, tickets logged by number, tickets issued and receipts. Online inquires about same items should be provided. | 2. TITLE: Ticket Sales<br>FREQUENCY: Random Quantity: Thousands per week<br>COMMENT : Reservation from should include name of patron, number of reservations program name, ticket transaction log should contain ticket number, date and price. |
| 3. TITLE : Profit and Loss Report<br>FREQUENCY : Weekly quantity : 4 copies<br>COMMENT : Lists revenue or expense item by event, data and type of revenue or expense incurred. | 3. TITLE : Program Revenue and expense input<br>FREQUENCY : Random Quantity 200 per year.<br>COMMENT : Includes all expenses incurred for Program date for expense and amount and type of expense. |
| 4. TITLE : Resources Inventory Listing<br>FREQUENCY : Weekly quantity : 4 copies<br>COMMENT : Lists available resources such as chairs, tables, lighting, staging products. | 4. TITLE : Resources Add/Change/ Delete<br>FREQUENCY : Random Quantity : As needed.<br>COMMENT : Input all equipment in stock at the centre including tables, chairs, lighting and staging products for inventory. |

REQUESTED BY: MUKUL GUPTA DEPARTMENT: MCG : APPROVED BY : PRAN SHARMA

TITLE: Director<br>TITLE: Member, Mgmt. Committee

—x— APPROVED

_____ APPROVED ON CONDITION. REASON:______________________

____ REJECTED REASON:______________________

____ SIGNATURE:____________ DEPARTMENT : EDP
TITLE : EDP Manager
DATE : DD/MM/YY

COMMENTS: Sarita Kakkar is the Chief Systems Analyst and her assistant is Anil Abraham. Systems Analysis begins next Monday. The systems analysis phase will take approximately two weeks. Please notify employees at the event centre to make themselves available for Sarita and Anil to interview. Full cooperation will expedite the systems analysis work and help to ensure a successful systems project.

Sarita and Anil worked late in the night to prepare the Proposal to Conduct Systems Analysis Report. Sarita knew from earlier bad experiences that typically one type of prospective systems users don't know exactly what they need, and the other type expects everything to be computerized on a personal computer immediately.

Sarita knew that if one tries to leapfrog several phases of the Structured Design Methodology, one nearly always winds up with a system nobody wants or needs. As a matter of fact, following the lead of typical users, and in essence letting them guide systems development, is similar to a doctor allowing a patient to guide surgical procedures or prescribe medicine.

So, to make sure that everybody is still on the same wavelength, the proposal to Conduct Systems Analysis Report, the first major documented deliverable in the SDM Process was presented both orally by Sarita and backed up by a written document.

## ASSIGNMENT

Consider yourself to be in Sarita Kakkar's shoes. Based on the narrative and the filled form as shown in EXHIBIT B-I. Write a crisp formal proposal to conduct systems analysis.

### SOLUTION TO ASSIGNMENT IN MCG CASE (B)

### A PROPOSAL TO CONDUCT SYSTEMS ANALYSIS

Month xx, 19xx

| | | |
|---|---|---|
| To | : | All Division Heads |
| From | : | Sarita Kakkar, Chief Systems Analyst |
| Subject | : | Scheduling and Ticketing System (STS) |
| Copies to | : | Mukul Gupta, Director, MCG and Neera Tiwari, EDP Manager |

**Reasons for the Analysis**

The recent expansion of the gymnatorium has increased the frequency of program which has in turn, added to the complexity of hosting them. The current manual information system is not meeting the information needs of those responsible for implementing the gymnatorium's business plan.

**Performance Requirements of the Systems**

The information systems plan previously completed provides the basis for establishing the following users requirements for the information system:

1. To increase organisation integration
2. To improve organisation communication
3. To provide more timely reporting
4. To improve the quality of service to the public at large
5. To reduce expenses
6. To increase revenues,

### Scope of Systems Analysis

The systems analysis work will determine the feasibility and direction(s) of the information needs as outlined in the systems plan and the information systems service request form. The findings of systems analysis will be subsequently be recorded in the Systems Analysis Completion Report. We will be responsible for the delivery of this report within two weeks from beginning of the analysis. Responsibility for systems design will be determined at the end of this systems analysis phase. We solicit your support for this effort because it is critical to undertake such a development if we are to improve our current management situation.

### Facts to be collected

To develop a complete picture of the proposed system, several details must be investigated further:

1. The sequence of steps and time required for each step in the process of hosting an event (Gantt or PERT chart).
2. Identify potential bottlenecks and a critical path for each type of event.
3. Determine the feasibility of integration of information between departments.
4. Determine the type of report needed by each division and management for each type of program.
5. Identify problems that exist in the current system.
6. Collect possible solutions from users.
7. Determine the feasibility of computerisation as the solution.
8. Determine the opportunities and problems associated with each alternative system.
9. Determine the feasibility of in-house development.
10. Determine the impact on the gymnatorium's work culture.
11. Determine the timetable for development and implementation.

## POTENTIAL SOURCES OF THE FACTS

The current organisation should be investigated first to provide baseline information necessary for identifying any proposed enhancements. This falls into several specific categories:

1. **The Present Information System:** The gymnatorium currently has separate systems within each division. All are manual systems with the exception of accounting. We propose to evaluate the effectiveness of the present system to determine whether a new system should be developed or whether the current system should be modified. An analysis of the old system will provide a starting point for any new development and provide design ideas. Understanding the current system enables us to evaluate the resources necessary for modifying the current system or developing a new one. Familiarity with the current system will help in planning for the conversion to a new computer based information system.

2. **Management and Employees of the Gymnatorium:** The use of questionnaires, interviews and informal discussions with the stakeholders of the gymnatorium allows us to develop an understanding of the needs of the users and how the current system operates.

3. **Current Paper Documentation:** Accounting procedure manuals, financial statements, management policies, organisational charts, job descriptions and other forms of paper evidence will be reviewed to determine how the organisation is operating and how it plans to operate. Current paper documentation will be used to reinforce the definition of relationship between the various divisions of the gymnatorium, interdivisional functions and people within the divisions.

4. **Contract with other Program Venues:** Program Venues with similar characteristics will be contacted to determine the type of system they use. This will provide an insight to document the types of problems which might be encountered with alternative systems. It will also enable to gain an understanding of alternative solutions available.

## SCHEDULE OF MAJOR EVENTS OR MILESTONES

The systems analysis will be undertaken as a series of tasks, each of which will have a specific scope and completion date.

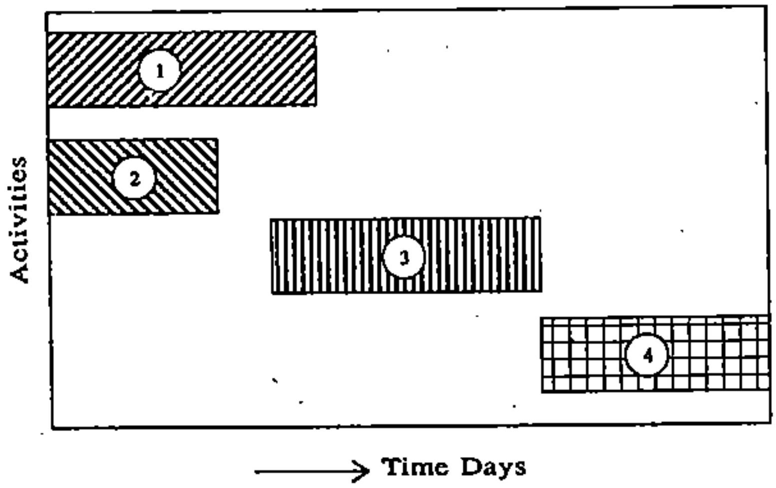
1. Review of the current system, interviews with people and review of documentation will take approximately one person two weeks to complete and will be performed by Sarita Kakkar (80 person hours).

2. Contacting other programs venue will take approximately one person one week to complete and will be performed by Anil Abraham (40 person hours) concurrent with Task 1.

3. Analysis of facts, determine of feasibility, definition of user requirements and direction will take two people approximately two weeks to complete and be performed by both analysts (80 person hours) after completion of Tasks 1 and 2.

4. Study facts synthesis and the Systems Analysis Completion Report will another take two weeks to complete by both analysts (80 person hours) after completion of Task 3.

GANTT CHART




Based on our discussions in the past few weeks I feel that we must give serious consideration to beginning this analysis for STS as soon as possible, ideally, we should begin our analysis within the next two weeks so that we can have answers to the requirements issues prior to our next financial year.

## CASE (C)

### MAHARAJA COLLEGE'S GYMNATORIUM
### SYSTEMS ANALYSIS COMPLETION

## INTERVIEWING

Sarita did most of the interviewing. Some of the study facts that came out from these interviews are shown in EXHIBIT C-1 which was her interview with Feroze Ticketwallah.

Interviewee : Feroze Ticketwallah
Interviewer : Sarita Kakkar
Duties : Head Cashier, Ticket Sales
Date : Month dd, 19xx

As cashier at the ticket window, he sees that several obvious problems exist:

- Often difficult to determine from the charts how many seats are in a group (for telephonic reservations for a block of seats), customers tend to get angry when it takes a few minutes to figure out what's available. In fact more than once customers have become angry and left without purchasing any tickets.
- Encountered numerous problems with having several people selling tickets concurrently, it's tough to keep two people from selling the same seat at the same time.
- The time delays are especially difficult with a group with young kids who get problematic at having to wait while their parents get tickets.
- Getting an accurate accounting of the money collected for each event only once a day is inefficient, management needs a more timely report but there just is not staff to do it.
- If they could only have a picture on a screen of the layout of the center with a seating chart for a given event which all of the cashiers could see simultaneously. It would make it a lot easier to give the customers tickets for good seats as fast as they want them.
- Returning tickets create a real problem because you have to run around to all of the cardboard charts and correct the blocked out seats.
- Ticket takers are frustrated from being bullied around and yelled at by angry customers, impatient management and nosy accountants who need the results of what they do faster, faster, faster......

EXHIBIT C-I Sarita's Interview Notes

## QUESTIONNAIRE

A questionnaire for clients of gymnatorium was sent out to a random sample of users. Sarita wanted to determine the nature of complaints and gather additional study facts to design the system better. EXHIBIT C-II illustrates this questionnaire.

**MAHARAJA COLLEGE GYMNATORIUM**
**SCHEDULING AND TICKETING QUESTIONNAIRE**

Dear Client :

As you know, we have enjoyed major expansion in the MCG recently. To serve you better, we need your help by answering the following questions.

1. You have been able to buy tickets without having to wait
   Strongly Disagree 1 2 3 4 5 6 7 8 9 10 Strongly Agree

2. You learn of coming programs in time to plan to attend
   Strongly Disagree 1 2 3 4 5 6 7 8 9 10 Strongly Agree

3. You always get your assigned seat
   Strongly Disagree 1 2 3 4 5 6 7 8 9 10 Strongly Agree

4. You can reserve seats easily
   Strongly Disagree 1 2 3 4 5 6 7 8 9 10 Strongly Agree

5. When you enter the center for a particular event, your ticket is checked and verified properly
   Strongly Disagree 1 2 3 4 5 6 7 8 9 10 Strongly Agree

6. You are directed to your seat efficiently and without hassle
   Strongly Disagree 1 2 3 4 5 6 7 8 9 10 Strongly Agree

Comments:

**EXHIBIT C-II : Clients Questionnaire**

## GATHERING AND ANALYZING ADDITIONAL STUDY FACTS

One of Anil's first tasks was to determine how the information flowed, or at least how it was supposed to flow in the present system. This general flow is represented in Figure C-1.

After some further analysis, Anil prepared the HIPO visual table of contents (VTOC), illustrated in EXHIBIT C-III. Anil also included a seating chart and physical layout of the event center which is shown in Figure C-2. An organisation chart for the gymnatorium did not exist, so Anil sketched the one illustrated in Figure C-3.

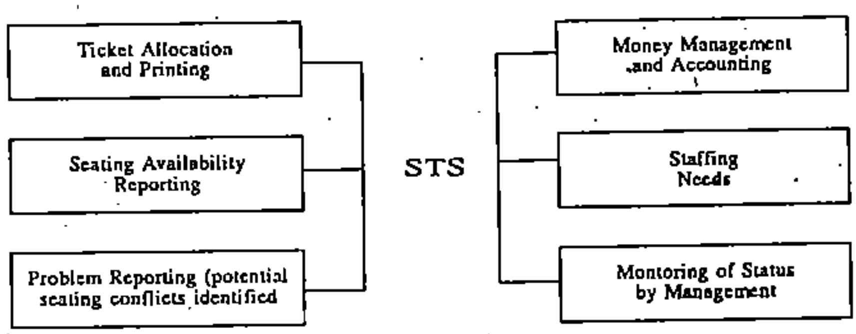



Fig. C-1 : Flow of Information in System

## DETERMINING RELATIONSHIPS IN THE TICKETING FUNCTION

Anil and Sarita spent a great deal of time trying to determine and put together relationships in the total ticketing process. After a great deal of frustration and false starts, they finally came to the realisation that tickets and ticketing was essentially an inventory control and marketing system. Once this assumption was made, they developed two entity=relationship (ER) diagrams that reflect this concept. EXHIBIT C-IV illustrates an ER diagram that models a ticket transaction from a selling viewpoint. EXHIBIT C-V demonstrates an ER diagram from an accounting and control viewpoint.

## PREPARING THE SYSTMES ANALYSIS COMPLETION REPORT

Sarita conducted most of the interviews and handled the questionnaire. Anil performed the data processing and flow analysis. Both combined their study facts and after several meetings, hammered out the Systems Analysis Completion Report.

## ASSIGNMENT

Assume the role of Sarita and Anil, and based on the narrative, figures and EXHIBITS, write a crisp "Systems Analysis Completion Report".

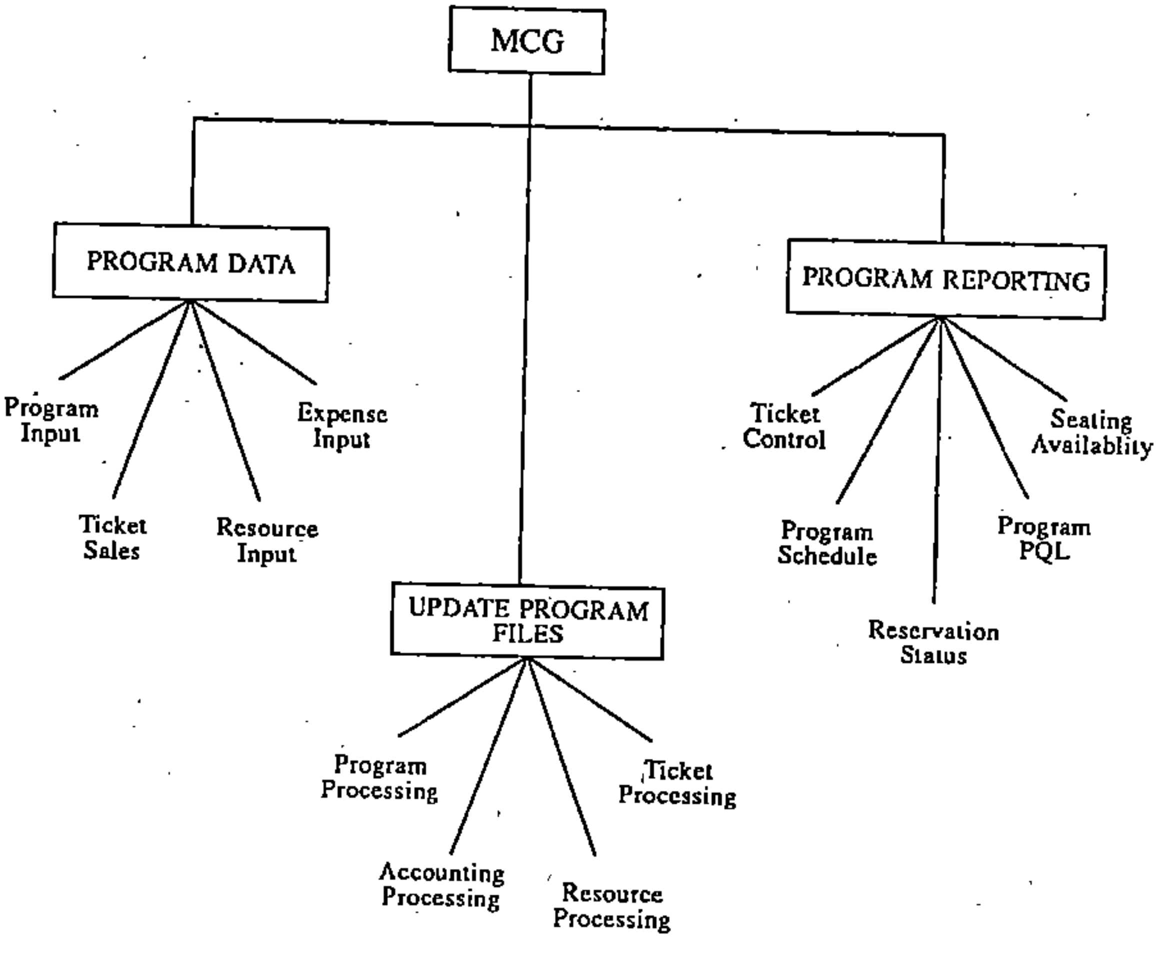



EXHIBIT C-III : HIPO VISUAL TABLE OF CONTENT

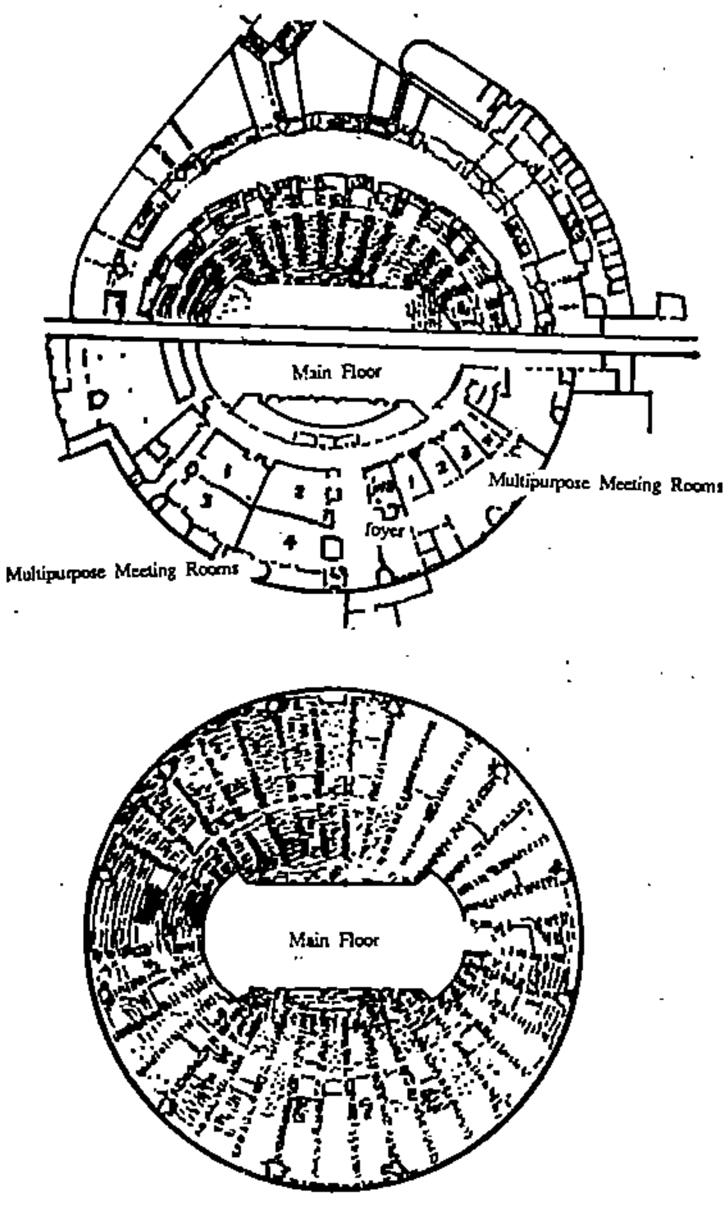



Figure C-2: Gymnatorium Seating Chart and Physical Layout.

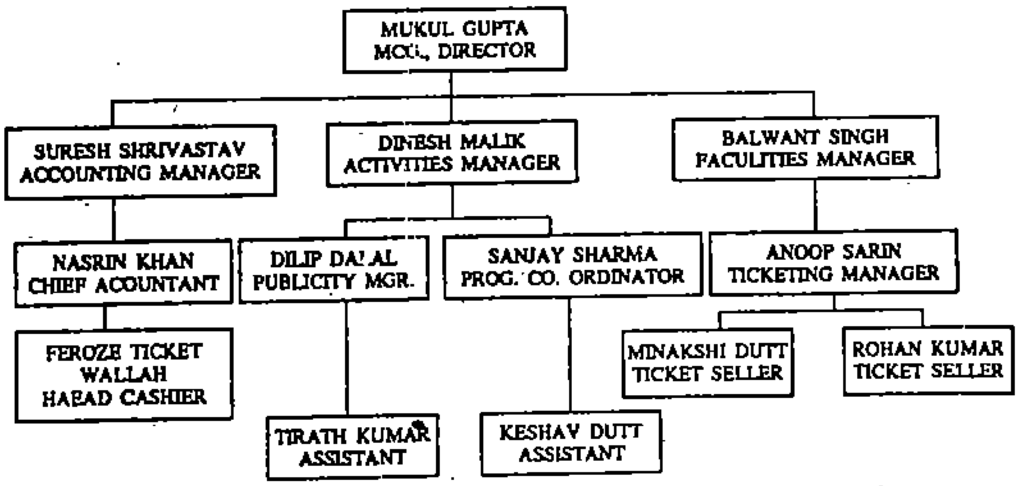
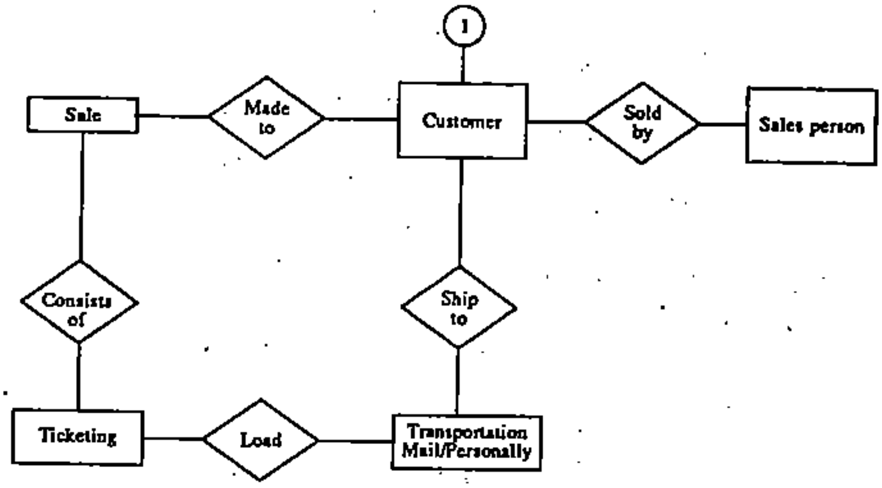
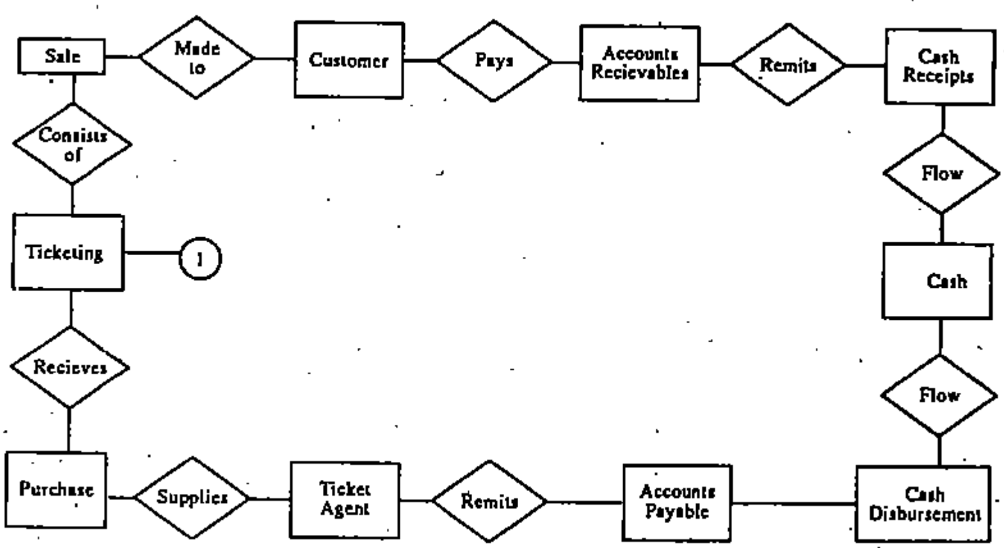



Figure C-3 Organisation chart for MCG .




| CUSTOMER DATA MODEL | | | | | | | |
|---|---|---|---|---|---|---|---|
| Customer Name | Customer Number | Telephone Number | Program Classi-fication | Payment | Mailing Address | Number of Tickets | Type |

EXHIBIT C-IV : An ER diagram that models a ticket transaction from a selling viewpoint.




| TICKETING | | | | | | | |
|---|---|---|---|---|---|---|---|
| Ticket No. | Program | Date | Price | Seat | | | |
| | | | | Type | Section | Row | Number |
| | | | | | | | |

EXHIBIT C-V : An ER diagram that models a ticketing transaction from an accounting and control viewpoint.

# SOLUTION TO ASSIGNMENT FOR MCG CASE (C)

## SYSTEMS ANALYSIS COMPLETION REPORT

Month dd, 19xx

To : All Division Heads
From : Sarita Kakkar, Chief Systems Analyst
Subject : Scheduling and Ticketing System (STS)
Copies : Mukul Gupta, Gymnatorium Director and Neera Tiwari, EDPManager

**Reasons and Scope**

The systems analysis was conducted to determine the feasibility and directions of STS. This report contains the findings of the systems analysis.

**Major Problem Identified**

Several major problems routinely experienced by the gymnatorium personnel:

1. Lack of coordination between divisions.
2. Lack of information on the status of resources.
3. Lack of information on scheduling required resources.
4. Lack of sales and expense status information for events.
5. Poor internal control of cash and tickets.

**Specific Problems Identified**

Consider some specific examples of how these problems have affected the ability of the gymnatorium personnel to meet their commitments:

1. The activities office is unable to communicate in a timely fashion the special accommodations required for special program such as boxing, concerts, and conventions. For example, for concerts and conventions, more communication is required because they are often not informed of special requirements such as room set-ups or planning for an associated exhibition until much too late to plan properly for and notify the relevant divisions of the extra needs for the program. On the other hand, inter- college basketball games require little coordination and the current level of communication is still adequate.

2. The current system of booking of programs has failed on several occasions, resulting in double bookings of programs. The lack of a central system for tracking the status of negotiation has led to different representatives of the gymnatorium promising and booking two conflicting types of programs in the same time period, which were extremely difficult to conduct.

3. The activities office is often unable to determine the status of specific resources available at the gymnatorium for a certain date. The result is the inability to meet commitments to promoters of programs and casual staff on a timely basis because casual staff was not scheduled early enough.

4. The inefficiencies relating to scheduling and the underestimating of the resources available in the gymnatorium have led to a policy of allowing slack-time between events to maintain quality of service. By computerizing we will be able to increase the number of events and schedule them more closely together. This will allow us to increase the usage of gymnatorium from the current 30 percent rate.

5. The activities office is not able to determine the status of ticket sales to date on a timely and cost-effective basis. This information is required by the staff at the scheduling office to determine whether to cancel a program before the contracted date.

6. The ticket office is unable to control ticket sales between the sales staff in the ticket office. Control can only be attained by cutting staff or centralising the information. Reducing staff is not feasible considering the volume of ticket sales. The only alternative is centralization of information.

7. The reconciliation of cash is time consuming. We never know what we should have in receipts because ticket dispensing is not controlled.

8. Poor control of ticket sales for assigned seating has resulted in the same seat being sold twice or more. Concurrently, some seats remain empty. This situation contributes to lost revenue and unhappy clientale.

**Statement of All User Requirement**

During the interviews and from other research we identified the following specific requirements:

1. Timely and efficient communication of program requirements to all divisions.

2. Efficient and accurate production and distribution of tickets.

3. Timely information on the status of tickets sales and expenses incurred for each program.

4. A system to ensure proper internal control of ticket sales and cash within the ticket office

5. A simple and efficient system to reconcile cash.

6. A system to manage credit card sales.

**Statement of Critical Assumptions**

All possible future constraints cannot be determined in the systems analysis phase. Future development is based on several critical assumptions.

1. After a period of parallel conversion from the current system the old system will be eliminated and the only system to be used will be the one developed as a result of this proposed project.

2. The final system developed will require that personnel be trained and the funding and support for the training will be committed.

3. The hardware and software required can be installed. No physical limitations of facilities will preclude implementation.

**Resources Required**

If the commitment to systems development is made, the gymnatorium must identify additional resources that do not currently exist or recommit staff time from current commitments. These resource requirements include:

1. The development phase will require additional personnel to form a development team. We anticipate the need for two system designers They should have experience in systems design and programming experience in one language

2. If the development is done in-house, acquisition of hardware and software as specified in the systems design is required If programs are acquired from an outside vendor, a similar acquisition is anticipated to support the product obtained. At this time it is anticipated that two personal computers with word processing and program development software will be required

3. The system design phase will be require 1000 person hours to complete at a cost of approximately Rs. 50 000

4. Management support is a key to success.

**Recommendations**

The discussions with division heads and other personnel indicate that the centralisation of information would enable the organisation to function more efficiently. It was universally felt that a computerised system that linked all the divisions together would improve coordination by improving communication through the use of centrally accessible specialized reports.

The systems development should be started as soon as possible. Problems experienced with the current information system have caused lost revenues and additional nonmonetary costs. A Systems Design Proposal Report will be produced in a few weeks that will provide feasible information systems design alternatives.

# CASE (D)

## MAHARAJA COLLEGE'S GYMNATORIUM (MCG) SYSTEM DESIGN PROPOSAL

## EMERGENCE OF DESIGN

To get a better perspective on the STS project, Sarita and Anil prepared a functional diagram, depicted in Figure D-1. This diagram, combined with a thorough review of their study facts gathered during systems analysis phase (Ref: MCG Cases (B) & (C) ) began to merge together helping to form building blocks, giving rise to several design issues.

## DESIGN ISSUES

Sarita had told Anil on several occasions, "The right hand doesn't know what the left hand is doing, at the gymnatorium. Moreover, they need a system that will integrate their activities. Otherwise they will be doomed".

"Yes, I know", Anil responded, "They're trying but they don't know how to go about it".

"Also, we've got to think about users' level of expertise. Most of them will be unskilled. Therefore the user/system interface must be carefully planned."

"They better start getting serious" said Anil. "The municipality is building its own general purpose auditorium, so competition is bound to increase".

"They're getting some information now, if you want to call it that but it's not usable. It's too late, so it is no way better than zero information. The timeliness for information seems to me to be a vital consideration in design. And for them timely information is key to be able to compete, "said Sarita.

I'm sure we'll be able to meet the systems requirements. The idea of building system's bridge to other businesses for ticket sales is a good one, though I believe it's a little premature at this stage of the design, "said Anil.

"Yes, First things first, "Sarita agreed, "but we can direct out designs towards enabling this kind of bridging if they want to go that way in the future.

"If they approve the kind of systems design I have in mind, data processing requirements will be very easy, "said Anil.

"You know, I've been thinking, "said Sarita, "the gymnatorium is just like any other service. Sometimes, I tend to look at systems differently if they are in an academic setting, but in reality all systems can pretty much be described in the flow of people, materials and data. The gymnatorium is organised along divisional lines and has decentralised management because Gupta lets them run their own areas."

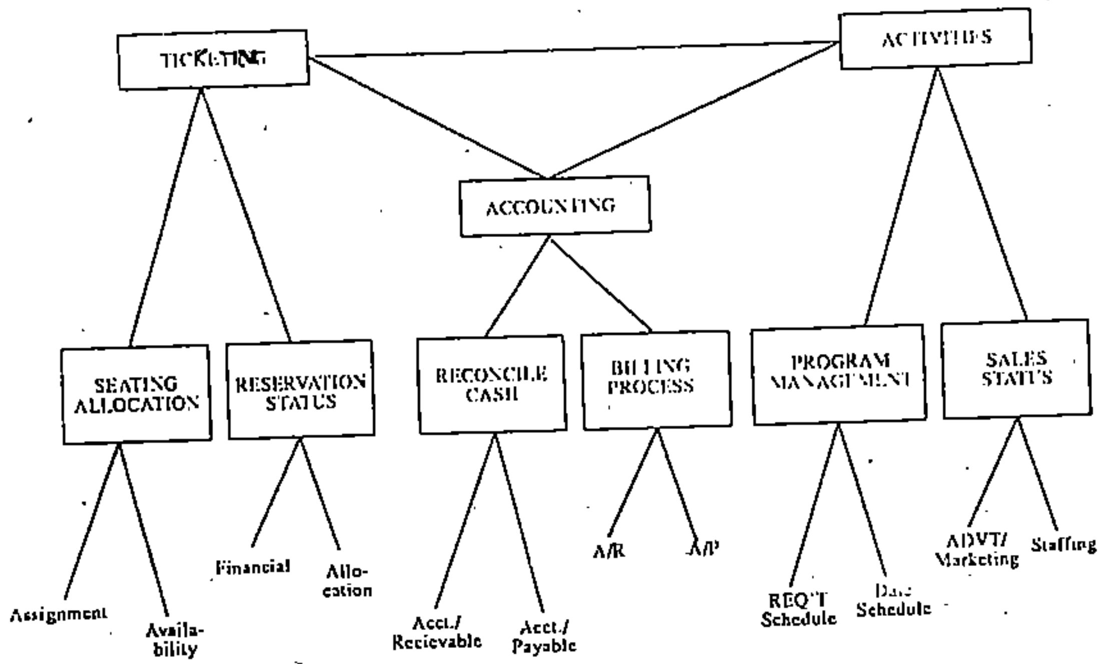



Figure D-1 : STS Design Levels

"That's true. It's just a typical business organisation.

"Something else that's really going to affect us is the cost- effectiveness. All the top people and especially the Management Committee are coming down hard on cost-effectiveness. I overheard Mr. Gupta tell Neera that though the Committee has allocated the money in the budget, but they're not going to approve anything unless it's cost-effective."

"Most of the time they just look at costs. That is, if it costs less than they are saving money, and that's not always true. Sometimes, cutting costs is the worst decision.

"You bet, "agreed Sarita. "Just look at what they're done over at the gymnatorium. They haven't spent much money, and look how they are just scraping through.

"I totally agree, "Sarita said. "They're really have a morale problem there. A few people told me they were leaving. They just can't take the hassling and confusion any more."

"We may look like real heroes when this is over," Anil said enthusiastically.

"If they can keep the right kind of people on staff. I agree. We can do some good things for them, but we're getting ahead of ourselves. First, we've got to come up with some viable designs for them to look at , and go from there. And what will make them viable is to consider and be aware of the design issues we've been discussing."

Sarita and Anil continued to weigh the impact of design issues on their design conceptualisations. From what they had learnt so far, it indicated to them some design issues should be weighed heavily in the design process. Their design issues weighting scheme is shown in EXHIBIT D-I.

**EXHIBIT D-I**
**WEIGHTING SCHEME FOR DESIGN ISSUES PERTAINING TO GYMNATORIUM**

| DESIGN FORCE | DESCRIPTION | WEIGHTING FACTORS (Total 100) |
|---|---|---|
| Integration | * Needed among accounting, activities and ticketing<br>* Users need ready access to system | 15 |
| Users/System Interface | * Many users are novices<br>* Need ease of use and limited training<br>* Automatic Monitoring and reporting | 15 |
| Competitive Forces | * Pooled and sharing of resources<br>* Concurrent ticketing for several program<br>* Need for immediate access to ticketing information<br>* Patron mailing list for promotion<br>* Service to Promoters of programs<br>* Extend reach into northern part of bordering state | 10 |
| Information Quality and Usability | * Timely and accurate seating availability<br>* Timely and accurate ticket disbursement<br>* Timely and accurate details for tickets disbursed | 10 |
| Systems Requirements | * Reliability goal is 100 percent in scheduling programs and ticket sales<br>* Availability goal is 100 percent to all users.<br>* Flexibility goal is to extend ticket selling into selected businesses in north-western region in the country<br>* Installation schedule is targeted at the end of next quarter<br>* Life expectancy is set for five years with 100 percent gwoth potential<br>* Maintainability is ensured by use of structured design standards documentation and modular programming | 10 |
| Data Processing Requirements | * Volume is over 2 million separate transactions per year with projections on nearly 4 million within five years<br>* Complexity involves scheduled of events and matching resources to support each event<br>* Time constraints for maximum productivity require online real-time processing<br>* Computational demands are relatively low | 5 |
| Organisational Factors | * Nature of the gymnatorium's business makes proper scheduling and ticket selling an integral part of its operation<br>* Type is divisional which facilities reporting requirements<br>* Size of the gymnatorium is one of the largest in the country that can accommodate a wide variety of programs<br>* Structure is a stand-alone operation with limited ties directly to Maharaja College so for all intents and purposes the gymnatorium is to be run like any other profit-making business<br>* Management style is decentralized. | 5 |
| Cost Effectiveness | * More efficient ticket disbursement with fewer staff<br>* Must be developed for current staff | 10 |

| | | |
|---|---|---|
| | * Management Committee require a formal cost-effectiveness report<br>* An audit is conducted 6 and 12 months after implementation to verify cost-effectiveness analysis | |
| Human Factors | * Ease of use<br>* Users have feeling they are in control<br>* Reduce frustration in dealing with angry patrons<br>* Raise Morale | 15 |
| Feasibility Requirements | * Technical feasibility is not a problem<br>* Economic pressures are to stay withinbudget<br>* Legal impact means safeguarding patron and promoter data<br>* Operational means the present staff and future personnel with limited expertise must be able to use the system with limited training | 5 |

## PREPARING A DATA FLOW DIAGRAM

Sarita had prepared a data flow diagram (DFD) of the present system. It is displayed in Figure D-2. Form this model, she and Anil developed a number of models on a CASE based workstation. The one they finally selected is illustrated in Figure D-3.

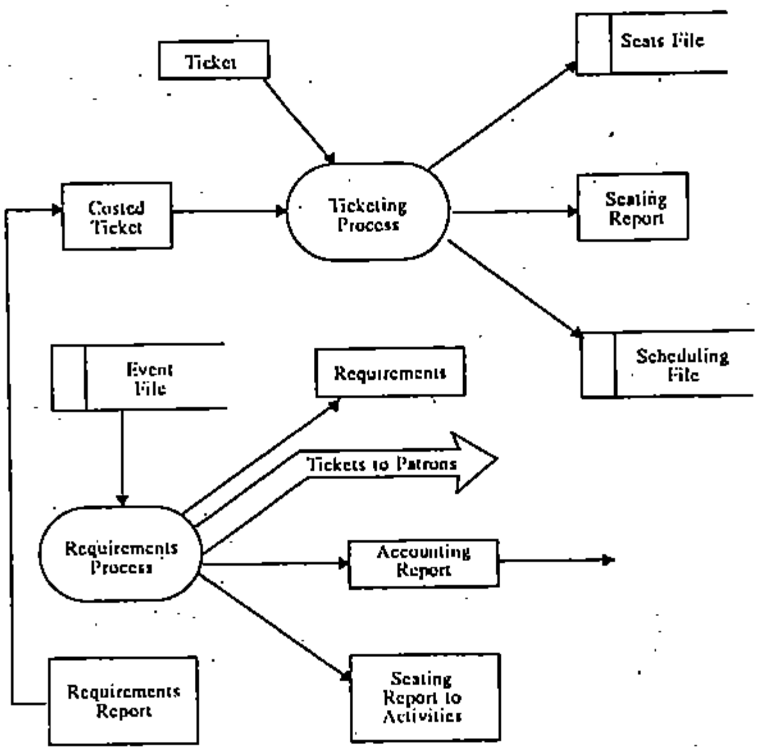



Figure D-2: Data Flow diagram of the present ticketing process

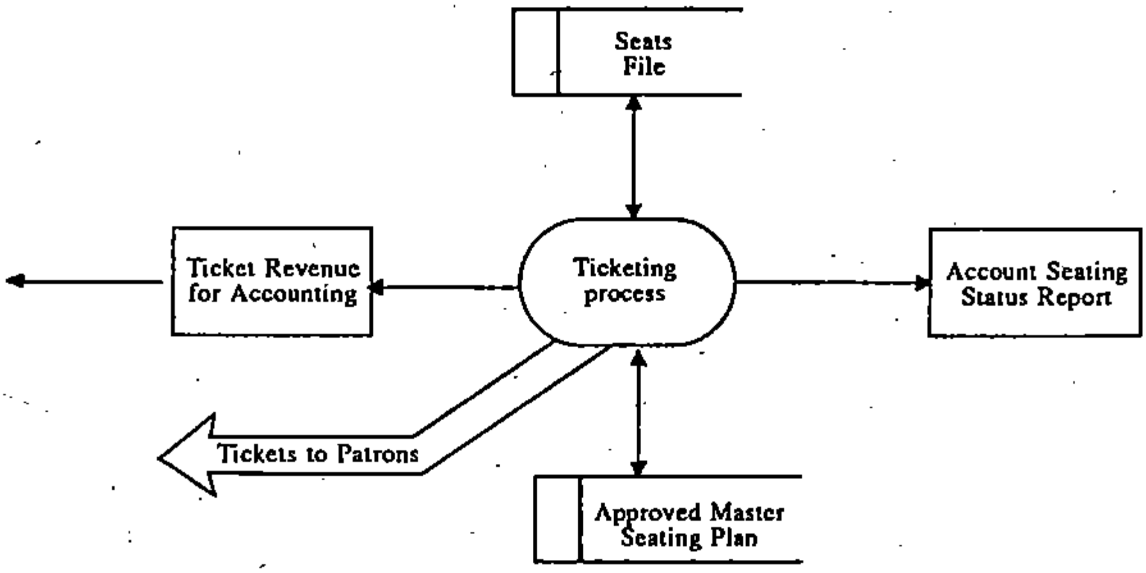



Figure D-3 . Proposed data flow diagram for the ticketing process.

## ASSIGNMENT

1. Consider yourself to be in place of Sarita and Anil come up with a systems design proposal report with three design alternatives, viz.

   a) The Stand-alone Batch Alternative

   b) The Network Based Alternative

   c) The Centralized Alternative

# SOLUTION TO ASSIGNMENT ON MCG CASE (D)

**SYSTEMS DESIGN PROPOSAL REPORT**

Month XX, 19xx

To : All Divisional Heads
From : Sarita Kakkar, Chief Systems Analyst
Subject : Scheduling and Ticketing System STS
Copies : Mukul Gupta, Gymnatorium Director and Neera Tiwari, EDP Manager

## DESIGN IMPERATIVE:

The present system does not meet the information and accounting control needs of the gymnatorium. Present operations are confusing and inefficient from all users and stakeholders viewpoints. For business continuity and to support a major expansion, it is imperative that new, viable systems be considered for further evaluation design and implementation.

**Possible Systems Design Models**

We have prepared thre general models for your consideration. We do not recommend one over the other. They all are viable based on our work upto this point. The one or ones you select or a combination thereof, will be evaluated further as to the best technology platform available to support the designs. Once we have defined precisely all the resources necessary to implement the cost-effectiveness analysis to determine the one that yields the best effectiveness to-cost ratio.

The three general systems design alternative can be classified as (1) stand-alone batch alternative (2) network-based alternative and (3) centralized alternative. The building blocks of each system are described in the following pages.

1. **THE STAND-ALONE BATCH ALTERNATIVE**
   **Building Block Design Sheet**

COMPANY : MCG
SYSTEM : STS

**INPUT :**

- Program Input Sheet
- Ticket Sales Recap Sheet
- Resources Available Input Sheet
- Ticket Sales Revenue Log
- Program Expenses
- Ticket Deposit Slip

**MODELS:**

- Matching old Program Requirements Against Resources Available
- Profit and Loss for each program (Income = Revenue - Expenses)

**DATA BASE:**

* Resources Master File
* Ticket Master File
* Program Master File
* Accounting File

**TECHNOLOGY:**

* Manual Systems in Ticket Office
* Personnel Computer in Accounting
* Personal Computer in Activities Office

**OUTPUT:**
- Ticket Control Report
- Program Schedule Report
- Reservation Status Report
- Seat Availability Report
- Program Profit and Loss
- Resources Inventory Listing

**CONTROLS**
- Preprinted Tickets
- Tickets Equal Seats
- Backup on Floppies
- Seating Chart Marked as seats are sold

**(1) INPUT**

A. The activities office fills out a program input sheet when A program is booked. The data sheet contains the following information:

- A program ID number (accounting number)
- Name of the Program or performer
- Estimated cost of the program
- Date of the program
- Duration of the program
- Resource requirements (i.e. staging, lighting, seating)
- Selection of resources configuration layout

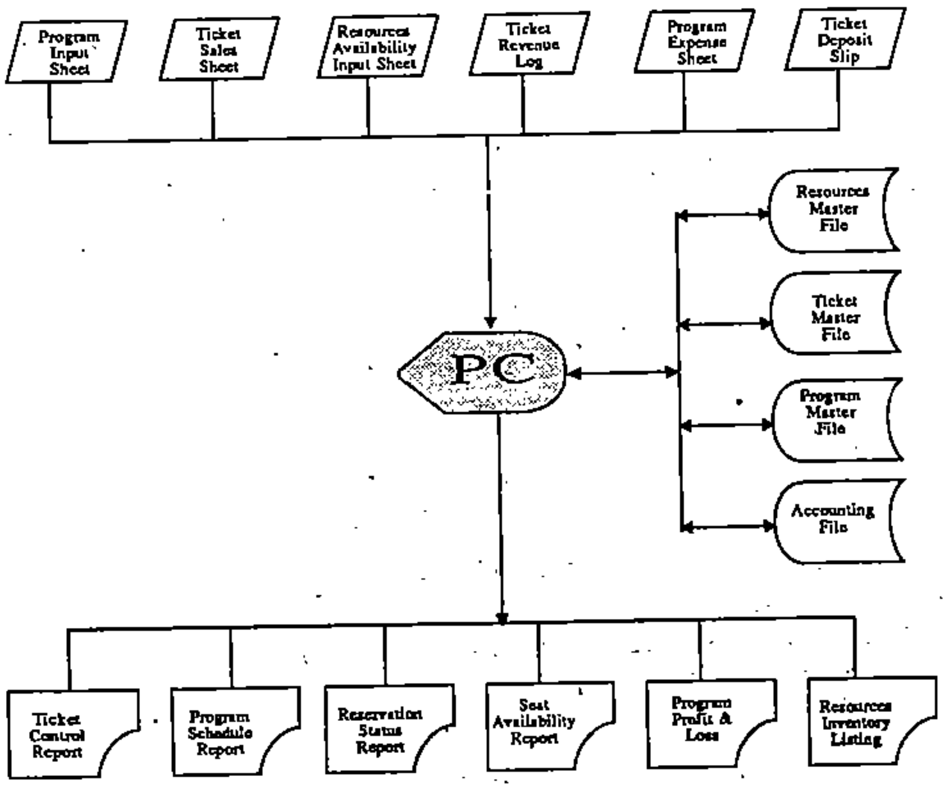
The data sheets are batched and entered into the activities office PC on a daily basis.

B. As preprinted tickets are sold, they are entered on a ticket sales revenue log sheet by the ticket office. The data sheets are sent to the activities office on a daily basis and entered in the PC.

Ticket deposit slips (bank deposit slip) and the ticket sales recap sheet are sent to the accounting office on a daily basis. The accounting office batches and posts these data sheets as part of the normal accounting cycle. The data sheets contain the following information:

- Log sheet line number
- Name of the patron (If phone order)
- Number of reservation (ticket numbers)
- Program, Program name, performer name
- Date purchased
- Purchase price
- Amount deposited.

STAND ALONE BATCH ALTERNATIVE



C. As resources become available or the availability changes, the facilities division fills out a resources available input sheet. This data sheet is sent to the activities office and entered on the PC. An expense sheet containing a code for each program and costs allocated to it is sent to accounting.

(2) MODELS

A. The resources master file produces a report that identifies resources available for each program.

B. Accounting models require the analysis of revenue and expense items per program to produce a profit and loss statement for each event on a weekly basis. An accounting general ledger code is used to identify the event and the type of revenue and expense item incurred.

(3) OUPUT

A. A ticket control report is produced to show tickets sold against seats for a program. Any variances or duplications are highlighted. This report is also used by accounting to verify cash deposited. Processing of the ticket control report is illustred in the following Warnier-Orr (W-O)

- PRODUCE TICKET CONTROL REPORT
  - BEGIN PROGRAM
  - UPDATE SEATS FILE
    - ID SEAT ASSIGNMENT (O.1)
      - SEAT NUMBER (1)
      - PRINT TICKET (1)
    - ISSUE TICKET
      - SEAT NUMBER(S) (1)
      - POST PAYMENT (1)
    - ADD SEATING RECORD (0.1) (UPDATE MASTER EVENT FILE)
      - SEAT NUMBER (1)
      - SEAT DESCRIPTION (1)
      - COST (1)
      - PRICE (1)
      - TYPE OF PAYMENT (1)
      - RECEIVED PAYMENT (1)
      - ISSUED TICKET (1)
      - QUANTITY ON HAND (1)
      - SOLD OUT (1)
      - TICKET REMAINING (1)
    - DELETE SEATING RECORD (0.1) CANCELLATION
      - SEAT AVAILABILITY (1)
  - END PROGRAM

B. A program schedule report is produced by listing the names of all programs, performers, cost of programs/performers, data of program and duration. The report can be printed and distributed weekly by the activities office or more often if needed. Online inquiry will only be available at the activities office.

C. Reservation status report is produced analyzing the status of ticket sales in the ticket master file and the program master file. A separate report is produced for each program. Each report contains a list of reservations by ID/name, tickets inventory logged by number and tickets issued.

D. A seating availability report is produced using the ticket master file. The report indicates the seats and tickets that are still available for each program.

E. The program profit and loss statement is produced from the accounting files. The report contains ticket receipts to date and expenses incurred to date.

F. A resources inventory listing is produced using the resources master file. This report contains a listing of available staging, lighting, seating and canteen facilities available in the gymnatorium.

## (4) TECHNOLOGY

A. The ticket office works on a manual system. Tickets are preprinted and the results of daily sales are entered on data sheets. The data sheets are sent to the activities office for processing on their personal computer.

B. The facilities division is supplied with input, change and delete forms that they submit to the activities office to maintain accuracy of the resources master file.

C. The activities office processes the forms from the ticket office and the facilities department on their personal computer. Forms generated from their own division are also processed. The appropriate menu-driven software will be available to maintain the files and produce the reports. Batch processing of data sheet is also performed.

D. The accounting department uses a personal computer to maintain the accounting file used to produce the program profit and loss statements.

## (5) DATA BASE

A. The resources master file contains all the resources available in the gymnatorium. The items are classified and coded by asset type such as staging, seating or canteen.

B. The ticket master file contains the ticket inventory for each program, the price of each ticket, and the seat assigned if applicable.

C. The program master file contains the name and ID of the program or performer, estimated cost of the program, date of the program, duration of the program, and resource requirements.

D. The accounting file contains information on revenue and expenses incurred for each program.

## (6) CONTROLS

A. System controls are as follows:

1. Batch processing control totals of ticket office activities.

2. Preprinted tickets accounted for and under tight access control.

3. Seating chart is marked as tickets are sold.

4. Tickets sold equals the number of seats marked.

5. Files are backed up on magnetic tape.

B. Access control is achieved by issuing keys to rooms that have computers.

## II. NETWORK BASED ALTERNATIVE.
Building Block Design Sheet

**COMPANY : MCG**
**SYSTEM : STS**

**INPUT**
- Online Ticket Deposit Slip
- Online Program Input
- Online Ticket Sales
- Online Resources Input
- Online Program Expenses Input

**OUTPUT**
- Online Ticket Control Report
- Online Schedule Report
- Online Reservation Status Report
- Online Program Profit and Loss Report
- Online Seat Availability Report
- Online Resources Inventory Listing
- Online Ticket Sales Recap Report

**MODELS**

(same as alternative 1)

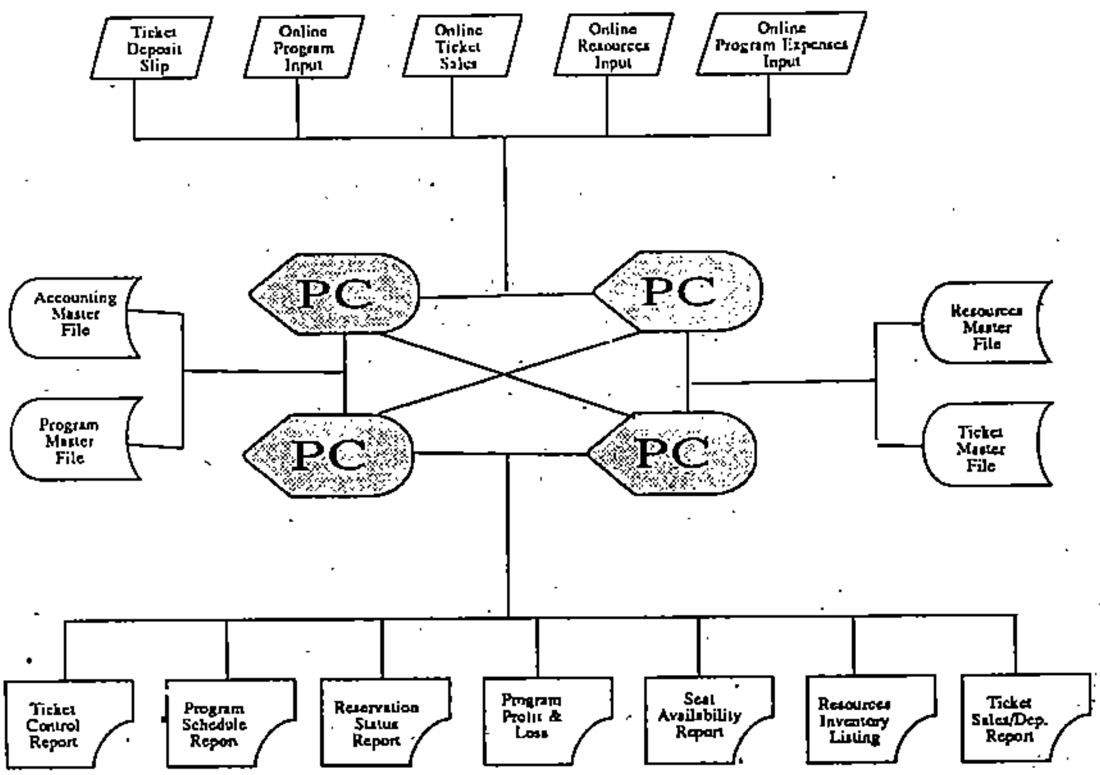
**TECHNOLOGY**
- Local Area Network Connecting Divisions
- Personal Computer in Accounting Division
- Personal Computer in Ticket Office
- Personal Computer in Facilities Division
- Real Time Processing of Program and Ticketing
- Batch Processing in Accounting

**DATA BASE**

(same as alternative I)

**CONTROLS**
- Preprinted Tickets
- Central Backup on Magnetic Tape
- Ticket Equal Seats
- Seating Chart Marked as Tickets as Sold
- Passwords

NETWORK BASED ALTERNATIVE



(1) INPUT

A. The activities office inputs directly into their PC program data when a program is booked. The input screen contains the following information:

* A program ID number (accounting number)
* Name of the program or performer
* Estimated cost of the program
* Date of the Program
* Duration of the program
* Resource requirements (i.e. stagiing, lighting, seating)
* Selection of resources configuration layout

B. As preprinted tickets are sold, they are entered directly into a PC in the ticket office. Ticket deposit slips and the ticket sales recap sheet are sent to the accounting office on a daily basis. The accounting office batches and posts these data sheets as part of the normal accounting cycle. The input screen contains the following information:

* Log sheet line number
* Name of the patron (if phone order)
* Number of reservation (ticket numbers)
* Program name or performer name
* Date purchased
* Purchase price
* Amount deposited

C. As resources become available or the availability changes, the facilities division enters add, change or delete information directly into its PC.

(2) MODELS

A. The resources master file produces a report that identifies resources available for each program.

B. Accounting models produce a profit and loss statement for each program on a weekly basis. An accounting general ledger code is used to identify the program and type of revenue or expense item incurred.

(Note: The same data flow diagram presented earlier depicts this model).

(3) OUTPUT

A. A ticket control report is produced to show tickets sold against seats. Any variances or duplications are highlighted. This report is also used by accounting to verify cash deposited.

B. A program schedule report is produced by listing the names of all programs, Cost of programs/performers, date of the program and duration. Online inquiry or report output is available at all offices.

C. A reservation status report is produced analysing the status of ticket sales in ticket master file and the program master file. A separate report is produced for each program. Each report contains a list of reservations by ID name, tickets inventory logged by number and tickets issued. Online enquiry or report output is available at all offices. (Note: the same W-O diagram presented earlier models this alternative).

D. The program profit and loss statment is produced using the accounting file. The report is available only through the accounting office.

E. A seat availability report is produced using the ticket master file. The report indicates the seats and tickets that are still available for each program. Online inquiry or report output is available at all offices.

F. A resources inventory listing is produced from the resources master file. This report contains a listing of the available staging, lighting, seating, and canteen services available in the gymnatorium. Online inquiry or report output is available at all offices.

G. A ticket sales recap sheet and deposit slip is produced and used for accounting input and control.

(4) TECHNOLOGY

A. The accounting division, the activities office, the ticket office and facilities division are linked together by a local network system. These divisions enter information directly into a personal computer in their division or office. The information is shared by the other divisions.

B. The ticket office preprints tickets.

C. Information from the activities office and the ticket office is provided in real time.

D. Information produced in the accounting division is restricted for inquiry in the other divisions or offices. The accounting division processes transactions in batch processing mode.

(5) DATA BASE

A. The resources master file contains all resources available in the gymnatorium. The items are classified and coded by assest type such as staging, seating or canteen.

B. The ticket master file contains the ticket inventory for each program, price of each ticket, and seat assigned if applicable.

C. The program master file contains name and ID of the program or performer, estimated cost of the program, date of the program, duration of the program, and resource requirements.

D. The accounting master file contains information on revenue and expenses incurred for each program and other financial data.

(6) CONTROL

A. System control are as follows :

1. Printed tickets are accounted for and are under tight access control.
2. Seating chart is marked as tickets are sold.
3. Tickets sold equals the number of seats marked.
4. Files are backed up on magnetic medium.

B. Access control is achieved by passwords and issuing keys to the rooms containing computers.

## III. CENTRALIZED ALTERNATIVE

Building Block Design Sheet

COMPANY : MCG
System : STS

INPUT

↓

(same as alternative II)

OUTPUT

(same as alternative II)

MODELS

↓

(same as alternative II)

TECHNOLOGY

- Multitasking, Multi-user Operating System
- Mini Computer Processing Unit
- CRT's for All Divisions and Offices
- Real-time Processing of Programs, Ticketing and Accounting
- Dialup Access from Ticket Office in Central market

DATA BASE

(same as alternative II)

CONTROLS

* Automatic Ticket Printing
* Central Backup
* Passwords

(1) INPUT

(same as alternative II)

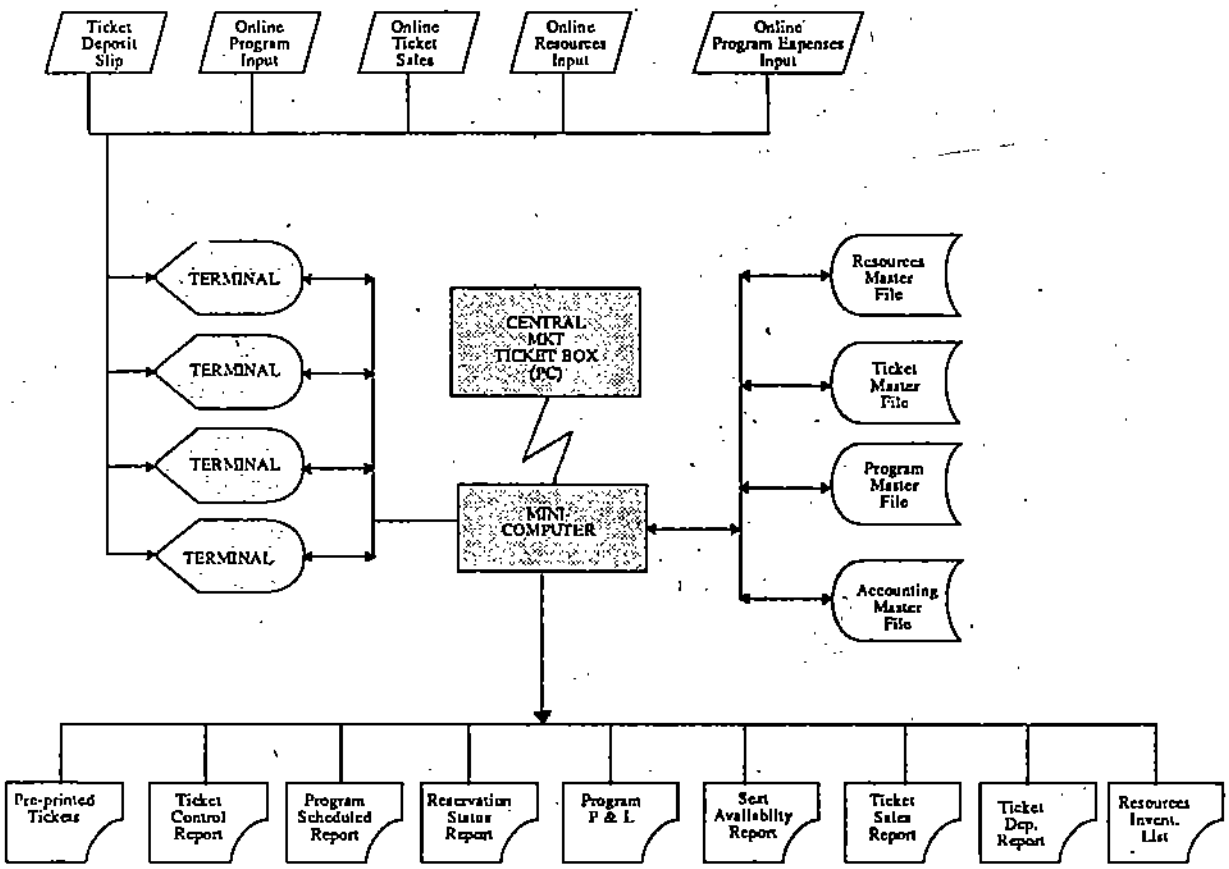
(2) MODELS

(same as alternative II)

(3) OUTPUT

(same as alternative II except for automatic printing of tickets)

THE CENTRALIZED ALTERNATIVE




(4) TECHNOLOGY

A. All Division and offices enter data by online terminals.

B. The system has its own ticket printing device.

C. Information from the activities office and the ticket office is provided in real time.

D. Information produced in the accounting division is restricted. The accounting division operates in a real time processing mode.

E. Access to the ticket reservation system is provided to the ticket office in central market of the city. This is expected to improve access of public to program.

(5) DATA BASE

(same as alternative II)

(6) CONTROLS

A. System controls are as follows:

1. Tickets are not printed until sold.
2. Electronic seating chart is marked as tickets are sold.
3. Tickets sold equals the number of seats marked.
4. Files are backed up on magnetic medium

B. Access control is achieved by issuing keys to the rooms containing terminals or the computer and by issuing passwords.

# CASE (E)

## MAHARAJA COLLEGE'S GYMNATORIUM EVALUATION AND SELECTION OF SYSTEMS

## CONDUCTING EVALUATION

Once the three generic systems design alternative were finalised, Sarita asked Anoop Sarin to schedule a meeting of the project team to review them. Because the final choice of a systems alternative would have a major impact on the function of the gymnatorium, thus it was essential project team be involved in the review. Sarita also felt that by making a formal presentation, she and Anil would be forced to consolidate the work completed to date.

After Sarita presented the findings, the three suggested alternatives were discussed threadbare. The meeting was long and often heated. A gist of the discussions is presented as under.

The stand-alone batch alternative automates the current manual system. Sarita and Anil estimated that at best it would provide a 20 percent increase in efficiency. Although they felt this increase would adequately solve the problems in the gymnatorium, they were concerned that this alternative revolved around manual handling of ticket delivery process for a given program. Most programs require using three to four ticket salespeople simultaneously and only one centralized preprinted batch of tickets for program. Either the ticket sales staff would be bumping into each other or they would have to break up the tickets into smaller batches and allow each sales person to sell for only one section of the gymnatorium. They would also have to share one seating chart or face difficulties in not duplicating seats offered to customers. However, the computer system portion of this alternative seemed extremely inexpensive to implement and administer. Costs of maintaining portions of the current manual system were yet to be determined. The important questions according to Sarita was, "Is this the best alternative when total costs are compared to total benefits?"

The project team felt the network-based alternative provided a means for each department to share information. A 50 to 60 percent improvement in efficiency is anticipated. This coordination and efficiency improvement would allow the Gymnatorium to operate at near capacity.

The centralized alternative, although most attractive to all concerned, but MCG budget had been traditionally low,, the team felt that requesting the funding for such an elaborate system might not find support in Management Committee. As a systems professional, however, Sarita was morally bound to present the centralized alternative because benefits projected far exceeded the costs. thus resulting in a better benefits-to-cost ratio than the other two alternatives.

By the end of the meeting the committee concluded that they could not reach a decision on the generic systems design alternatives until vendor proposals were evaluated and facts concerning the specifics on each technology platform were gathered and a cost- effectiveness analysis was performed.

Before developing the request for proposal (RFP), Sarita felt that one enhancement should be made. For each ticket disbursement by a given sales agent, a summary statement should be printed that would serve as a receipt for the transaction. This statement would include the number of tickets, type of tickets, type of payment, amount of payment and so forth, and would serve as input to the accounting system. Then, if the customer needed further service, a record of the prior service could be obtained.

## REQUEST FOR PROPOSAL

Sarita and Anil then sent out RFP (Request for Proposal) in the format given below:

**Introduction**

The MCG requests proposals for a Seating and Ticketing System (STS) to be provided on or before month dd, 19xx. This system should enhance current administrative and accounting support for the ticketing function of the gymnatorium for all types of programs currently scheduled. These events are of three types: (1) Sporting programs for which open seating is required, (2) programs such as concerts for which designated categories of tickets and set-up are required, and (3) academic programs such as professional conferences that have specialised requirements for ticketing and seating. The system provided should enhance the ability of the gymnatorium to maintain a mix of types and numbers of programs on a tight schedule, properly maintaining accounting and other requisite record keeping for the center on timely basis. The proposals should address the mandatory requirements in full. The MCG welcomes suggestions for enhancements and options within the spending guidelines.

**Imperatives**

The following issues must by addressed for the RFP to be considered:

Centrally accessible data : Ability to share data and printed reports among the activities office, the ticket office, and the resources division.

Menu driven : User friendly, menu-based system for commonly used capabilities.

Expandibility and Upgradability: As the gymnatorium expands, the system must be expandable to meet the increased need.

**Desirable**

Addressing the following items will improve the consideration for the proposed solution:

User report generation : Independence in production of reports necessary for each departmental user of the system.

Online help: Access to information on software use from all workstations at all times.

## PROPOSAL PREPARATION GUIDELINES

Vendors should prepare and deliver proposals to the MCG's Purchasing Division Office to meet the specific needs designated in the attachments on or before Month dd, 19xx.

**Vendor Performance and Financial Condition**

Proof of the vendor performance history, growth and financial strength is required. An audited financial statement is also required.

A biographical sketch (resume) for all persons who will be involved in installation and support is required. Also, a list of the responsibilities and tasks assigned to each participant is required.

A list of customers (10-15) who have acquired the similar technology being proposed is required.

**Systems Documentation**

Samples of documentation for systems similar to the ones being proposed must be included. Complete review of the available online and offline documentation must be provided on request by vendors who make the final short list.

### Vendors Policies and Procedures

Relevant policy information for hardware and software testing and beta test results should be described.

### Legal and Business Procedures

Submit copies of your purchase and service and maintenance agreements, include information on discounts, payment schedules and cancellation policies.

### Generic Systems Design Proposal Report

A copy of the Generic Systems Design Proposal Report developed by the MCG Gymnatorium is included. This document will provide information on user requirements, building blocks and the systems working environment for use in preparing proposals.

### Data Processing Requirements

Provide definition of hardware and software set-up criteria. Include responses to the following questions:

* Disk capacity?
* Disk speed?
* Disk access requirements?

Note: It is proposed to evaluate computing platform proposals, using benchmarks.

### Systems Requirements

Items that contribute to a better functioning of system and effect good systems design should be recommended as needed, including:

* Reliability measured by MTBF
* Modularity
* Compatibility
* Installation schedule
* Ease of use
* Maintainability measured by MTTR
* Vendor support

### Full Description of Building Blocks of Computing Platform

A full description of each recommended computing platform is required, including:

* Network topology
* Architecture
* Model number
* Operating system
* Primary storage capacity
* Front-end and back-end processors
* Language processors
* Security and control features
* Application program packages
* Peripherals

### Price and Financing Plans

Include full pricing, financing and acquisition methods.

### Evaluation methodology

Simulations and benchmark tests may be scheduled during evaluation as required. Customers will be contacted about their experiences with the proposed technology, its performance and ease of use.

## ASSIGNMENT ON MCG CASE (E)

Assume that you are the team consisting of Sarita and Anil that is conducting the evaluation and selection process. Write a comprehensive report on this analysis using 8 level screening process. The 8 Level Screening Process consists of following screens:

1. Desk Checking of Proposal Elements
2. Technical Performance Criterion Comparison
3. General Performance Criterion Comparison
4. Benchmark and Simulation Tests
5. Single Vendor Vs. Multiple Vendor
6. Price and Contract negotiations
7. Acquisition and Financing Methods
8. Cost Benefit Analysis

(Hint: You can send a fictitious RFP to selected vendors to get cost and technical data to perform this analysis).

## SOLUTION TO ASSIGNMENT IN MCG CASE (E)
## SYSTEMS EVALUATION AND SELECTION REPORT

Month dd, 19xx

To : All Division Heads
From : Sa. Kakker, Chief Systems Analyst
Subject : Scheduling and Ticketing System (STS)
Copies : Mukul Gupta, Gymnatorium Director and Neera Tiwari, EDP Manager.

## GENERAL EVALUATION STRATEGY

Vendors who submitted acceptable proposals were subjected to a seven-screen evaluation process to select the best vendors with the best technology platforms. The eighth evaluation screen subjected each systems design alternative with its technology platform to a cost-effectiveness analysis. The one with the best effectiveness-to-cost ratio should be selected for implementation.

**Response from Requests for Proposals**

The RFP and the Generic Systems Design Proposal Report were sent to 15 vendors. Three of the vendors were computer manufacturers, seven general computer systems vendors, and five were OEMs (Original Equipment Manufacturers) specializing in theater/program center ticketing and control systems.

One general computer systems vendor did not respond and another general computer systems vendor was eliminated in the general review of the proposal because of unclear substantive commitments. One computer manufacturer proposal was eliminated because it contained only hardware specifications without software. Twelve vendors remained after the general review and a seven-step screening process was started to consider the proposals of two computer manufacturers, five general computer system vendors and five OEMS. The following is a list of the vendors and the alternative to which they responded. Alternative 1 is the stand-alone batch generic systems design. Alternative 2 is the network-based generic systems design. Alternative 3 is the centralized generic systems design.

| Vendors | Generic Systems Design Alternative |
|---|---|
| Computer Manufacturer | |
| VENDOR 1 — Wipro Systems | Alternative 3 |
| VENDOR 2 — Digital Equipment | Alternative 3 |
| General Computer Vendors | |
| VENDOR 3 — The computer Land | Alternative 1, 2 |
| VENDOR 4 — Computer Store | Alternative 1 |
| VENDOR 5 — Computer Point | Alternative 1, 2 |
| VENDOR 6 — Micro Land | Alternative 2 |
| VENDOR 7 — Computer Joint | Alternative 1 |
| Original Equipment Manufacturer | |
| VENDOR 8 — Central Information COs | Alternative 2, 3 |
| VENDOR 9 — Program Data Processing | Alternative 2 |
| VENDOR 10— Tickets systems Unlimited | Alternative 3 |
| VENDOR 11— Online Reservations Pvt. Ltd. | Alternative 2 |
| VENDOR 12— The Schedulers Inc. | Alternative 3 |

### Evaluation Screen One: Desk-Checking Proposal Elements

The following criteria were used to screen vendors in the first evaluation.

A. Vendor personnel

B. Vendor customers

C. Vendor documentation

D. Vendor testing procedures and policies

E. Vendor financial condition

F. Vendor general, legal and business procedures

Based on this evaluation, Xs were placed in appropriate cells of the following grid when unacceptable:

Vendor Number

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A. | . | . | . | . | . | x | . | . | . | . | . | . |
| B. | . | . | x | . | . | . | . | . | . | . | . | . |
| C. | . | . | . | . | . | x | x | . | . | . | . | . |
| D. | . | . | . | . | . | . | . | . | . | . | . | . |
| E. | . | . | x | . | . | . | . | . | . | . | . | . |
| F. | . | . | . | . | . | . | x | . | . | . | . | . |

Vendor 3 was eliminated based on phone calls made to this vendor's customers. They complained about training and installation support and unexpected costs that were incurred. Vendor 3 also presented an audited financial statement that had looks of creative accounting. Vendor 6 is a well-known vendor with a reputation for rapid turnover of personnel and was therefore eliminated. The documentation for vendor 6 was also unclear and difficult to follow. Vendor 7 was eliminated because the contract included a license agreement that called for a large yearly payment for a minimum of 10 years without the possibility of ownership. Vendor 7 was also eliminated for unclear and incomplete documentation. Vendor 10 did not provide evidence that testing procedures were conducted by a test group that was separate and independent from the development group.

Based on the outcome of this screening, vendors 1, 2, 4, 5, 8, 9, 11, and 12 remained in the running.

**Evaluation Screen Two : Technical Performance Criteria Comparisons**

This section contains:

- Hardware technical performance criteria comparisons
- Software technical performance criteria comparisons

| Hardware Technical Performance Criteria | Vendors | 4 | Computing Platform for Standalone Batch System (Alternative I) 5 |
|---|---|---|---|
| Purchase Price | | Rs. 150,000 | Rs. 300,000 |
| CPU | | 80386 | 80386 |
| CPU Speed | | 33 Mhz | 50 Mhz |

| | | |
|---|---|---|
| Main Storage | 1 Mbytes | 1 Mbytes |
| Hard Disk Provided | 240 Mbytes | 300 Mbytes |
| Hard Disk Access Time | 60 MIs | 40 MIs |
| MTBF Rating | 40,000 hours | 30,000 hours |
| Display Provided | EGA colour | Monochrome |
| Display Resolution | 640X400 PPI | 1000x800 PPI |
| Display Graphics | Yes | Yes |
| Backup System | Floppies | Mag tape |
| Network Upgradable | Yes | Yes |
| Printer Provided | 24-pin DMP | Laser |
| Expansion slots | 8 | 7 |

| Software Technical Performance Criteria / Vendors | 4 | 5 (Computing Platform for Standalone Batch System (Alternative I)) |
|---|---|---|
| Operating System | MS-DOS | MS-WINDOWS |
| Programming languages Provided | COBOL<br>C | COBOL<br>SQL/RDB |
| Programming languages used | C | SQL/RDB |
| Source code provided | No | Yes |
| Main storage required | 640K | 512K |
| Disk storage required | 20 Mbytes | 30 Mbytes |
| Menu driven | Yes | Yes |
| Network upgradable | No | Yes |
| Access control | None | Passwords |
| Backup program | Yes | No |
| Structured Design | No | Yes |
| Integrated | Yes | Yes |
| Documentation quality | Fair | Good |
| Modules: | | |
| Seating allocation | No | No |
| Reservation status | No | No |
| Reconcile cash | No | Yes |
| Billing process | Yes | Yes |
| Program management | Yes | Yes |
| Sales status | Yes | Yes |

After reviewing the two proposals, vendor 5 was chosen as having the best proposal for alternative 1 because of the ability to upgrade to a LAN system. We then compared vendors 5, 8, 9 and 11 for alternative 2.

| Hardware Technical Performance Criteria/Vendors | 5 | 8 | 9 | 11 |
|---|---|---|---|---|
| | | | Computing Platform for Network-Based System (Alternative 2) | |
| Price | Rs. 540,000 | Rs. 750,000 | Rs. 800,000 | Rs. 600,000 |
| Workstations | Olliveti 1 | IBM AS 1 | IBM AS4 | Microvax |
| CPU speed | 18 Mhz | 25 Mhz | 25 Mhz | 20 Mhz |
| Main storage | 1MB | 1MB | 4MB | 1MB |
| Network cards | Enable/LAN | Ethernet | Net II | G-Net |
| Hard disk | 20 Mbytes | 30 Mbytes | 30 Mbytes | 40 Mbytes |
| Harddisk speed | 60 MIs | 45 MIs | 45 MIs | 40 MIs |
| MTBF rating | 40,000 hours | 30,000 hours | Unavailable | Unavailable |
| Display | CGA colour | VGA colour | Monochrome | EGA colour |
| Display resolution | 320x200 PPI | 640x200 PPI | 640x200 PPI | 720x400 PPI |

| | | | | |
|---|---|---|---|---|
| Display graphics | Yes | Yes | No | Yes |
| LAN system | 3-Com | IBM ring | IBM PC Net | Dec Net |
| Server | Ollivetti III+ | IBM AS1 | IBM AS 4 | Micro Vax |
| Server speed | 25 Mhz | 25 Mhz | 25 Mhz | 25 Mhz |
| Hard disk | 80 Mbytes | 80 Mbytes | 240 | 240 |
| Backup system | 60 Mbytes Tape | 40 Mbytes Tape | None | 60 Mbytes Tape |
| Data transfer | 10 Mbps | 15 Mbps | 12 Mbps | 9 Mbps |

| Software Technical Performance Criteria/Vendors | 5 | 8* | 9 | 11 |
|---|---|---|---|---|
| | | | Computing Platform for Network-Based System (Alternative 2) | |
| Price | Rs. 450,000 | Rs. 420,000 | Rs. 400,000 | Rs. 360,000 |
| LAN system | Novell | Banyan | PC-Net | DECNET |
| Programming language provided | SQL/DB | Pascal Cobol | SQL/DDL C | SQL/FTP |
| Programming language used | SQL/DB | Pascal Cobol | SQL/DDL C | SQL/FTP |
| Main memory required | 500K | 400K | 640K | 1 Mbyte |
| Main storage required | 20 Mbytes | 12 Mbytes | 10 Mbytes | 30 Mbytes |
| File locking | Yes | No | Yes | Yes |
| Passworks | Yes | Yes | Yes | Yes |
| Menu driven | Yes | Yes | Yes | Yes |
| report generators | Yes | No | No | Yes |
| Documentation | Excellent | Poor | Good | Fair |
| Service rates | Rs. 250/months | Rs. 300/Month | Rs. 400/month | Rs. 4000/Yr |

* Eliminated vendor 8 because of poor Software documentation.

| Hardware Technical Performance Criteria/Vendors | 1 | 2 | 8 | 12 |
|---|---|---|---|---|
| | | | Computing Platform for Centralized System (Alternative 3) | |
| Price | Rs. 900000 | Rs. 1400000 | Rs. 1000000 | Rs. 1500000 |
| CPU | Supreme 3020 | DEC Vax II | IBM RC | IBM 9480 |
| CPU speed | 33 | 30 Mhz | 50 Mhz | 66 Mhz |
| Main storage | 33 Mbytes | 20 Mbytes | 20 Mbytes | 20 Mbytes |
| Hard disk | 344 Mbytes | 240 Mbytes | 120 Mbytes | 260 Mbytes |
| Hard disk speed | 25 Mls | 30 Mls | 40 Mls | 30 Mls |
| MTBF rating | 100,000 hr | 70,000 hr | 40,000 hr | 95,000 hr |
| Backup system | Mag Tape | Mag Tape | Mag Tape | Disk |
| Backup power | None | 15 Minutes | 30 Minutes | Auto Save |
| Input devices | Mouse | Lightpen | Touch screen | Mouse |
| Terminals | Wyze 50 | DECUT 100 | IBM 3164 | TISL 200 |
| Terminals | 720x512 PPI | 1000x480 PPI | 860x600 PPI | 640x480 PPI |

| Software Technical Performance Criteria/Vendors | 1* | 2 | Computing Platform for Centralized System (Alternative 3) 8 | 12* |
|---|---|---|---|---|
| Price | Included | Included | Included | Included |
| Operating system | Pick | VMS 2.0 | Unix system 5 | MVS |
| Programming language provided | COBOL COBOL | FOCUS | C | C and FOCUS |
| Programming language used | COBOL | SQL/DA | ORACAL | SQL/DP |
| Main storage required | 10 Mbytes | 15 Mbytes | 15 Mbytes | 20 Mbytes |
| File locking | No | Yes | Yes | Yes |
| Menu driven | Yes | Yes | Yes | Yes |
| Passwords | Yes | Yes | No | Yes |
| Documentation | Fair | Good | Good | Excellent |
| Report generators | No | Yes | Yes | No |
| Online help | No | Yes | Yes | No |
| Monthly service | Rs. 1000 | Rs. 1300 | Rs. 1200 | Rs. 1600 |

*Eliminated vendor 1 because it does not provide a 4GL Vendor 12 was eliminated because its proposal system did not contain online help.

**Evaluation Screen Three : General Performance Criteria Comparisons**

Because vendor 5 was the only one remaining for alternative 1, it becomes the vendor of choice for alternative 1, Vendor 8 was eliminated from alternative 2 because the software documentation was poor. Vendor 1 was eliminated from alternative 3 because programming in a fourth generation language is preferred over COBOL. Vendor 12 was eliminated from alternative 3 because it was deemed inadequate without a report generator or online help. Further screening for alternative 2 and alternative 3 is conducted based on the following general performance criteria:

* Compatibility: Flexible, Independent, Portable and Universal
* Modularity : Segments merged readily and modules changed easily.
* Maintainability : Changes in the program are made easily and low MTTR
* Reliability : Safe Shutdowns, high MTBF
* Life expectancy : Newer technology and support system over its life
* Installation schedule : Time to install
* Vendor support : Quality and cost of support when needed

| NETWORK BASED SYSTEM (Alternative 2) | | | | | | | |
|---|---|---|---|---|---|---|---|
| General Performance Criteria | | Vendor 5* | | Vendor 9 | | Vendor 11* | |
| | Weight | Value | Score | Value | Score | Value | Score |
| Compatibility | 20 | 7 | 140 | 5 | 100 | 8 | 160 |
| Modularity | 20 | 9 | 180 | 8 | 160 | 9 | 180 |
| Maintainability | 10 | 4 | 40 | 2 | 20 | 1 | 10 |
| Realiability | 10 | 8 | 80 | 9 | 90 | 10 | 100 |
| Life expectancy | 5 | 9 | 45 | 7 | 35 | 4 | 20 |
| Installation schedule | 5 | 10 | 50 | 8 | 40 | 5 | 25 |
| Vendor Support | 30 | 6 | 180 | 10 | 300 | 4 | 120 |
| Total | 100 | | 715 | | 745 | | 615 |

* Eliminatied : Although vendors 5 and 11 received higher marks form compatibility and modularity vendor 9 prevailed because of a very high mark for vendor support. Vendor 9 offered a five year support contract included in the price of the system. Vendor 9 received favorable commants from customers concerning their level of support and users group has been formed. Vendor 9 also provides training facilities with experienced personnel.

| CENTRALIZED SYSTEM (Alternative 3) | | | | | |
|---|---|---|---|---|---|
| General Performance Criteria | | Vendor 2 | | Vendor 8* | |
| | Weight | Value | Score | Value | Score |
| Compatibility | 10 | 6 | 60 | 5 | 50 |
| Modularity | 10 | 9 | 90 | 8 | 80 |
| Maintainability | 5 | 1 | 5 | 2 | 10 |
| Reliability | 10 | 9 | 90 | 7 | 70 |
| Life expectancy | 20 | 9 | 180 | 7 | 140 |
| Installaton Schedule | 5 | 7 | 35 | 9 | 45 |
| Vendor Support | 40 | 8 | 320 | 6 | 240 |
| Total | 100 | | 780 | | 635 |

* Eliminated : Although vendor 8 provided higher performing hardware, it was felt compatibility, modularity, maintainability and vendor support were more important. For these reasons vendor 2 prevailed.

**Evaluations Screen Four: Benchmark and Simulation Tests**

The vendors left in the running up to this point are reasonably close to each other in what they are offering . The question now becomes which one actually performs the best under operating conditions.

At this point, vendor 5 supports alternative 1, vendor 9 supports alternative 2, and supports alternative 3.

**Benchmark tests are based on :**

* Anticipated work load
* Compilers
* Operating system
* Application and utility packages
* Input–bound and process bound speed

| Applications | Vendor 8 Alternative 1 | Vendor 9 Alternative 2 | Vendor 2 Alternative 3 |
|---|---|---|---|
| Ticket entry | NA | NA | 80 sec |
| Backup of disk | 30 min | 10 min | 45 sec |
| Print event listing | 10 min | 9 min | 4 min |
| Process 20 events | 2 min | 1 min | 20 sec |
| Query data base | NA | 5 sec | 3 sec |
| Process a ticket request | 5 sec | 4 sec | 1 sec |

Simulation tests considered were:–

* Turnaround time
* Clock time
* Systems capacity
* Define optimum equipment configuration

Note : Simulation tests could not be performed because of the unavailability of appropriate programs and personnel.

Vendor 2 performed better based on the benchmark tests over vendors 5 and 9. Similarly, vendor performed better than vendor 5. This difference in the performance does not serve to eliminate any of the three at this point because of the price and configuration differences.

**Evaluation Screen Five : Single Vendor Versus Multiple Vendor**

The evaluation team decided that dealing with a single vendor is advantageous. A single vendor involves fewer complexities and better support. Upgrades and changes are easier to obtain. It is also difficult to combine software packages from different vendors.

**Evaluation Screen Six : Price and Contract negotiations**

After negotiations with the remaining vendors, we determined the actual selling price. The terms of the bids are now in writing and the price and financial arrangements are specified in detail. Delivery dates are mutuslly agreed upon and specified and acceptance criteria have been outlined. The specifics of all warranties have been spelled out and approved by legal counsel and purchasing & division:

| Contract terms | Vendor 5 Alt 1 | Vendor 9 Alt 2 | Vendor 2 Alt 3 |
|---|---|---|---|
| Selling Price | Rs. 300.000 | Rs. 1,200.000 | Rs. 1,400.000 |
| Delivery date | DD/MM/YY | DD/MM/YY | DD/MM/YY |
| Financing available | Purchase or rent | Purchase or installment purchase | Operating lease of finance lease |
| Warranties | 1 year | 1 year | 5 years |
| Service contract | Rs. 3000/ yr. Level 1 | Rs. 12000/yr. Level 2 | Rs. 15,000/ yr. Level 2 |
| Acceptance criteria | Letter of approval within 30 days | Letter of approval within 90 days | Acceptance test |

**Evaluation Screen Seven: Acquisiton and Financing Methods**

Before determining costs versus benefits of the three altamative, it is necessary to determine the best acquisition and financing method.

Based on our analysis, it is best to purchase the technology block for alternative 1, to use the installment purchase method for the technology platform for alternative 2, and lease using the operating method for the technology block for alternative 3. A summary of the financial details are as follows:

| Contract terms | Vendor 5 Altomative 1 | Vendor 9 Alternative 2 | Vendor 2 Alternative 3 |
|---|---|---|---|
| Financing method | Purchase | Installent purchase | Operating lease |
| Cost | Rs. 100.000 | Rs. 17450/month | Rs. 20.000 / month |
| Term | NA | 7 years | NA |
| Depreciation | Rs. 20,000 for 5 years | Rs. 90,000 for 5 years | Na |
| Salvage value | Zero | Zero | NA |
| Residual | NA | NA | Rs. 500,000 |

**Evaluation Screen Eight : Cost– effectiveness Analysis of General Systems Design Alternatives**

We use the present value (PV) of cash flows method to determine which alternative over a period of ten years will provide the best financial benefit to the gymnatorium. Our analysis is based on the following costs and benefits:

**Benefits**

I. Tangible
   a. Labour savings form more efficient scheduling and coordination
   b. Revenue from additional program bookings
   c. Cost savings from improved scheduling
   d. Efficient and controlled ticket handling
   e. Increased revenue from the ability to sell tickets in central market (for Alternative 3 only)

2. Intangible
   a. Customer goodwill from efficient and controlled ticket handling.

Cost

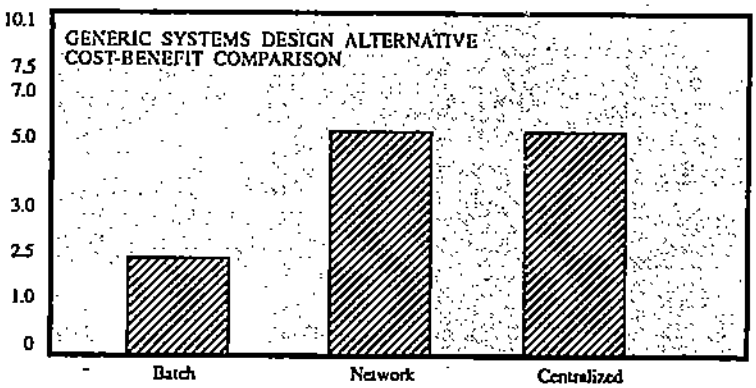
1. Systems implementation costs
2. Computing platiform costs
   a. Cost to acquire the initial system with finace charges
   b. Cost of additional programming and hardware if required
3. Cost to operate the computer facility
   a. Power requirements
   b. Air conditioning
   c. Furniture and fixtures
   d. Supplies
   e. Staffing
   f. Maintenance contract fees
   g. Insurance

| STAND-ALONE BATCH ALTERNATIVE 1 | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Years | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Labour saving | 30,000 | 40,000 | 40,000 | 50,000 | 50,000 | 60,000 | 80,000 | 90,000 | 10,000 | 1,20,000 |
| Revenue | 4,00,000 | 6,00,000 | 8,00,000 | 8,00,000 | 8,00,000 | 8,00,000 | 8,00,000 | 8,00,000 | 8,00,000 | 10,00,000 |
| Cost savings | 60,000 | 60,000 | 50,000 | 40,000 | 40,000 | 30,000 | 30,000 | 30,000 | 20,000 | 20,000 |
| Ticketing | 0 | 0 | 10,000 | 10,000 | 10,000 | 15,000 | 15,000 | 20,000 | 20,000 | 20,000 |
| Benefits | 4,90,000 | 7,00,000 | 9,00,000 | 9,00,000 | 9,00,000 | 9,15,000 | 9,25,000 | 9,40,000 | 9 40,000 | 9,60,000 |
| Hardware | 3,00,000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Programming | 35,000 | 0 | 0 | 0 | 50,000 | 0 | 0 | 0 | 50,000 | 0 |
| Maintenance | 3,000 | 3,000 | 3,000 | 3,000 | 3,000 | 3,000 | 3,000 | 3,000 | 3,000 | 3,000 |
| Operations | 1,00,000 | 1,50,000 | 1,50,000 | 1,60,000 | 1,30,000 | 1,80,000 | 2,00,000 | 2,50,000 | 2,50,000 | 2,50,000 |
| Costs | 4,38,000 | 1,53,000 | 1,53,000 | 1,63,000 | 2,33,000 | 1,33,000 | 2,03,000 | 2,03,000 | 3,03,000 | 2,53,000 |
| Net cash | 52,000 | 5,47,000 | 7,47,000 | 7,37,000 | 6,67,000 | 7,32,000 | 7,22,000 | 7,37,000 | 6,37,000 | 9,07,000 |
| Factor @ 20% | .883 | .696 | .578 | .482 | .402 | .335 | .279 | .233 | .186 | .162 |
| PV | 43,316 | 3,80,712 | 4,31,766 | 3,55,234 | 12,68,134 | 2,45,220 | 2,01,438 | 1,171,721 | 1,18,482 | 1,46,934 |
| TOTAL PV | 23,62,957 | | | | | | | | | |

TOTAL BENEFITS 72,60,000
RATIO 3

| NETWORK-BASED ALTRNATIVE-2 | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| a. Labour savings | 40,000 | 50,000 | 70,000 | 3,80,000 | 1,00,000 | 1,14,000 | 1,20,000 | 1,40,000 | 1,40,000 | 1,50,000 |
| b. Revenue | 1,10,000 | 1,10,000 | 1,15,000 | 1,20,000 | 1,20,000 | 1,30,000 | 1,30,000 | 1,30,000 | 1,30,000 | 1,30,000 |
| c. Cost Savings | 60,000 | 60,000 | 80,000 | 80,000 | 1,00,000 | 1,00,000 | 1,50,000 | 1,50,000 | 1,3 0,000 | 2,00,000 |
| d. Ticketing | 40,000 | 40,000 | 50,000 | 50,000 | 60,000 | 60,000 | 60,000 | 60,000 | 60,000 | 60,000 |
| e. Benefits (A+B+C) | 2,50,000 | 2,60,000 | 3,15,000 | 3,30,000 | 3,30,000 | 4,00,000 | 4,60,000 | 4,80,000 | 5,10,000 | 5,30,000 |
| f. Hardware Installments | 17,143 | 17,143 | 17,143 | 17,143 | 17,143 | 17,143 | 17,143 | 0 | 0 | 0 |
| g. Programming | 10,000 | 10,000 | 10,000 | 10,000 | 50,000 | 10,000 | 10,000 | 10,000 | 50,000 | 10,000 |
| h. Maintenance | 12,000 | 12,000 | 12,000 | 12,000 | 12,000 | 12,000 | 12,000 | 12,000 | 12,000 | 12,000 |
| i. Operations | 1,50,000 | 1,50,000 | 1,60,000 | 1,80,000 | 2,00,000 | 2,00,000 | 2,00,000 | 2,00,000 | 2,10,000 | 2,10,000 |
| j. Cost (F+G+H+I) | 1,89,143 | 1,89,143 | 1,99,143 | 1,99,143 | 2,19 ,143 | 2,19,143 | 2,39,143 | 2,22,000 | 2,82,000 | 2,42,000 |
| k. Net Cash | 60,857 | 76,857 | 1,15,857 | 1,30,857 | 1,20,857 | 1,80,857 | 2,20,857 | 2,58,000 | 2,28,000 | 2,88,000 |
| Factor @ 20% | 0.833 | 0.696 | 0.579 | 0.482 | 0.402 | 0.335 | 0.279 | 0.233 | 0.196 | 0.162 |
| PV | 50,694 | 49,316 | 67,081 | 63,073 | 48,584 | 60,587 | 61,619 | 60,114 | 44,688 | 46,656 |
| Total PV | 5,52,412 | | | | | | | | | |
| Total Benefits | 39,15,000 | | | | | | | | | |
| Ratio | 7.1 | | | | | | | | | |

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| NETWORK-BASED ALTERNATIVE 3 | | | | | | | | | | |
| a. Labor savings | 50,000 | 70,000 | 80,000 | 90,000 | 1,10,000 | 1,30,000 | 1,30,000 | 1,30,000 | 1,30,000 | 1,30,000 |
| b. Revenue | 1,20,000 | 1,20,000 | 1,60,000 | 1,60,000 | 1,50,000 | 1,50,000 | 1,60,000 | 1,60,000 | 1,80,000 | 1,80,000 |
| c. Cost Savings | 30,000 | 1,20,000 | 1,50,000 | 1,80,000 | 1,30,000 | 1,30,000 | 1,30,000 | 1,80,000 | 1,80,000 | 1,80,000 |
| d. Ticketing ( A+B+C) | 60,000 | 1,00,000 | 1,00,000 | 1,00,000 | 1,00,000 | 1,00,000 | 1,00,000 | 1,20,000 | 1,20,000 | 1,20,000 |
| e. Contract Mkt Ticketing | 26,000 | 26,000 | 28,000 | 28,000 | 30,000 | 32,000 | 32,000 | 32,000 | 36,000 | 36,000 |
| f. Benefits (A+B+C+D+E) | 3,36,000 | 4,36,000 | 5,18,000 | 5,58,000 | 5,70,000 | 5,90,000 | 6,02,000 | 6,22,000 | 6,46,000 | 6,46,000 |
| g. Hardware lease | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 |
| h. Programming | 10,000 | 30,000 | 30,000 | 30,000 | 15,000 | 30,000 | 30,000 | 30,000 | 20,000 | 30,000 |
| i. Maintenance | 1,55,000 | 15,000 | 15,000 | 15,000 | 15,000 | 15,000 | 15,000 | 15,000 | 15,000 | 15,000 |
| j. Operations | 2,00,000 | 2,00,000 | 2,50,000 | 2,50,000 | 2,50,000 | 2,80,000 | 2,80,000 | 3,00,000 | 3,00,000 | 3,00,000 |
| k. Costs (G+H+I+J) | 2,45,000 | 2,65,000 | 3,15,000 | 3,00,000 | 3,00,000 | 3,00,000 | 3,45,000 | 3,65,000 | 3,55,000 | 3,65,000 |
| l. Net cash | 91,000 | 1,71,000 | 2,03,000 | 2,43,000 | 2,70,000 | 2,55,000 | 2,57,000 | 2,27,000 | 2,91,000 | 22,81,000 |
| Factor @20% | 0.833 | 0.696 | 0.579 | 0.482 | 0.402 | 0.335 | 0.279 | 0.233 | 0.196 | 0.162 |
| PV | 75,803 | 1,19,016 | 1,17,126 | 1,17,126 | 1,08,546 | 85,425 | 71,703 | 59,881 | 54,126 | 45,522 |
| Tital PV | 7,94,803 | | | | | | | | | |
| Total Benefits | 55,24,000 | | | | | | | | | |
| Ratio | 7.0 | | | | | | | | | |




MC GYMNATORIUM

# CASE (F)

## MAHARAJA COLLEGE'S GYMNATORIUM SYSTEMS IMPLEMENTATION IMPLEMENTATION PLAN AND ACTIVITIES

## OVERALL IMPLEMENTATION PLAN

The PERT diagram of the overall implementation plan for STS is shown in Figure F-1.

## IMPLEMENTATION ACTIVITIES

**Order Technology**

Upon MCG officials signing the contract with DEL, the purchasing department prepared and released the purchase order to DEL. Delivery dates for the computing platform were mutually decided to coincide with the completion of the site for installation.

**Review of Specifications**

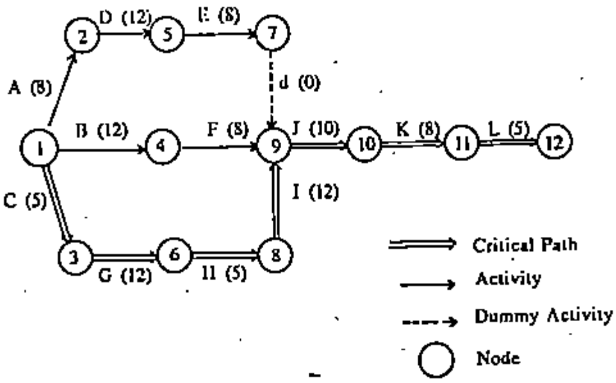
An analyst from DEL reviewed the specifications for input, models, output and the data base, to customize the necessary program code for ticketing, event recording and resources planning, accounting and data base maintenance models, as described in the Systems Design Report. (Ref Case-D)

**Site Preparation and Installation**

The goal of site preparation is to ensure quick and easy installation of the computing platform.

The equipment will be burnt in by DEL at their plant to simulate four weeks of continuous operation. During this time, the facility division of MCG has been informed to prepare the site for the equipment. The environment must be clean and air- conditioned. Sufficient space is to be provided for shelving, stands, tables and disk and tape storage trays and cabinets.

Representatives from DEL will be at the MCG to coordinate the installation of cabling. They will provide recommendations on improving CHI (Computer Human Interface) viz. acoustics, privacy panels, printer enclosures and ergonomically designed furniture and workstations.



A = Order Computing Platform
B = Prepare Site
C = Review Specifications
D = Install Equipment
E = Test Hardware
F = Training
G = Write Programs
H = Test Programs
I = Test Software
J = Convert Systems
K = Implement followup
L = Accept

**Figure F-1: PERT Chart of System Installation Activities**

### Test Technology

DEL plans to install hardware monitors on the CPU and disk drives for three weeks to monitor their operation. A software monitor will be installed permanently to monitor the usage of the software. Code modifications will be performed to increase efficiency. DEL will provide utilities such as data set optimizers to improve throughput by reducing wasted file storage on disk. They will also provide schedulers to help meet timing demands and balance job mix.

### Write Programs

Comparing the detailed programming specifications against the turnkey system provided by DEL expressed the need for additional programming, to customize the software. A team of DEL programmers worked on this job in consultation with Sarita and Anil.

### Test Programs

Anil Abraham performed a structured walkthrough of all the amended program modules written by DEL. After the modules were tested and integrated with DEL's core software packages, both valid and invalid test transactions were run to test the total software system. No errors were found, and Anil Abraham certified the software system 100 percent reliable.

Programming changes to the core system in the future will be performed by DEL as specified in the contract between the MCG and DEL. The MCG is allowed to produce custom reports and develop programs that read from the data base. Anil Abraham will receive specialized training in SQL to aid users in preparing ad hoc reports.

### User Training

Based on an assessment of the tasks required to operate the new system, it is deemed unnecessary to hire new employees.

DEL will conduct one week of on-site training before the hardware is installed. The company will organize seminars and group instruction during this week. DEL remains on-site two weeks after the hardware is installed to provide on-the-job training to operators and users. Interactive video programs on CD-ROM are also provided for personnel who wish to review training material. DEL also provides a voice-mail based hot line to answer any questions.

### Input Testing

One week prior to delivery of hardware, the divisions will be instructed on how to fill out the forms. Users will be tested to determine if they are completing the forms correctly. Accuracy and speed of data entry personnel will be evaluated. The efficiency of screen input layouts will be evaluated. Any screens that are cluttered or contain unnecessary data will be identified and corrected.

## Output Testing

Reports and enquiry screens will be tested for accuracy based on live data that are entered during the training sessions. All reports will be issued to the final users, and each user will be required to fill out a questionnaire to see if the reports meet his or her information needs. The questionnaire will address the following issues:

1. Are headings accurate and understandable?
2. Are editing characteristics of report fields correct?
3. Are debit/credit and notations correct?
4. Are page numbers in correct sequence?
5. Are end-of-report indicators understandable?
6. Are reports issued with the correct date?
7. Are reports understandable and accurate?
8. Comments?

## Data Base Testing

After live data are entered, data files are tested for completeness by comparing batch control totals to totals produced from the database. Other database tests are performed by DEL during development.

## Control Testing

Controls testing emphasizes the way transactions are prepared and entered. Specific items to be tested are as follows:

1. Are new programs being booked with proper authorization?
2. Are customer accounts being created with proper authorization?
3. Are ticket transactions being performed in proper sequence?
4. Is the use of passwords being controlled?
5. Are control totals being prepared and reported back to the control group?

Controls built into the system by DEL to be tested are:

1. Numeric, alphabetic and special character checks
2. Validity checks on key data fields
3. Limit and reasonableness checks

## Systems Conversion

The systems conversion process is performed in parallel with the manual system currently in place for six months in all areas except ticketing, which will be converted directly on DD/MM/YY. Parallel conversion in accounting, programs and resources divisions is necessary to reconcile differences that may occur. Parallel conversion of ticketing is not practical. The new system is converted directly and must perform on its own.

All current data are converted directly and reconciled during the training sessions in cooperation with DEL. A rehearsal for direct conversion of the ticketing operations is to be performed on Sunday, Month DD 19XX. The ticket office is normally closed on Sunday and will remain closed during the trial, if the trial is performed successfully, then the STS ticketing system will be used on Monday, Month DD, 19XX.

## IMPLEMENTATION FOLLOW-UP AND ACCEPTANCE MEETING

### Implementation Follow-up

A user representative assigned by Mukul Gupta will perform implementation follow-up. Sarita and Anil will be available to answer questions. The representative will be responsible to review and report on a monthly basis how STS is operating.

The review covers the following areas:

1. Input, processing and output schedules
2. Activities of input preparation personnel
3. Backup procedures
4. Computer operator procedures
5. Report utilisation
6. Hardware maintenance

The user representative report will contain:

1. Areas that need improvement
2. Recommended methods for reconciling errors
3. Recommended areas where additional control is required.
4. Areas where old material will be eliminated from the system.
5. Requested upgrades to the system.

### Acceptance Meeting

An acceptance meeting will be held on DD/MM/YY to discuss the achievements upon completion of training sessions and to determine if the gymnatorium is ready to perform parallel conversion. The meeting will be attended by systems analysts, systems operating personnel and users. If a consensus is reached that the system installed is ready for use then the systems analysts will be released from working on the development of STS to perform work on the Facilitating and Seating System, a follow- up to STS. After systems conversion, STS will become the responsibility of Mukul Gupta and his staff.

## ASSIGNMENT ON MCG CASE (F)

Q.1 Consider the PERT diagram for systems implementation as shown in Figure F-1. Based in the data available in MCG cases A-F assign reasonable time values to various activities listed and determine the implementation time.

Q.2 Assume that DEL software engineer has found a problem with ticketing model as described in MCG Case (D). Try to identify the problem and write the psuedo-code of that model again.

Q.3 Prepare a plan for training and educating users of new information system at MCG.

Q.4 Write a Acceptance Report (by considering yourself to be Sarita Kakkar), consequent to conduct of an Acceptance meeting.

Ans. 1. The Activities can be named as:

| Activity | Name | $t_p$ | $t_m$ | $t_o$ | $\left(t_e = \frac{t_p + 4t_m + t_o}{6}\right)$ |
|---|---|---|---|---|---|
| A(1-2) | Order Equipment | 10 | 9 | 2 | 8 |
| B(1-4) | Prepare Site | 20 | 12 | 4 | 12 |
| C(1-3) | Review Specs | 8 | 5 | 2 | 5 |
| D(2-5) | Install Equipment | 18 | 7.5 | 4 | 12 |
| E(5-7) | Test Hardware | 10 | 8.5 | 4 | 8 |
| d(7-9) | Dummy | 0 | 0 | 0 | 0 |
| F(4-9) | Training | 10 | 9 | 2 | 8 |
| G(3-6) | Write Programs | 20 | 12 | 4 | 12 |
| H(6-8) | Test Programs | 8 | 5 | 2 | 5 |
| I(8-9) | Test Software | 18 | 7.5 | 4 | 12 |
| J(9-10) | Convert Systems | 16 | 10 | 4 | 10 |
| K(10-11) | Implement Follow-up | 10 | 8.5 | 4 | 8 |
| L(11-12) | Accept | 8 | 5 | 2 | 5 |

Possible paths are

| | | |
|---|---|---|
| 1-4-9-10-11-12 | = | 43 days |
| 1-2-5-7-9-10-11-12 | = | 51 days |
| 1-3-6-8-9-10-11-12 | = | 57 days |

Hence critical path is

1-3-6-8-9-10-11-12 with tasks C, G, H, I, J, K, L and the time required for implementation of MCG systems project is 57 days.

Ans. 2.

The following is a description and a psuedo-code listing of the corrected program ticketing model earlier presented in case (D).

1. Ticketing Model:

   It allows the user to input seating assignments. When a ticket is sold and paid for, the input screen records information about the customer, the amount paid, the purchase date, and the seats assigned. The "available" relation is updated and the "issued" attribute is updated to read "y" for Yes, which means the ticket or seat has been issued. The ticket is then printed.

> In this module, the relation called "Programs" is opened. The system looks up the program number for the event the user has selected. The program number is stored in a temporary memory variable called "number" for further use by the model.

OPEN THE PROGRAMS RELATION
    LOOKUP THE PROGRAM_NUMBER WHERE THE RECORD NUMBER =
    Select STORE Program_number TO number.

> In this module, the relation called "assigned" is opened. The input screen is displayed and the customer's ticket order is input and stored in the "assigned" relation. The event number is stored in the "assigned" relation using the data stored in the temporary memory variable called "number".

OPEN THE assigned RELATION

DISPLAY THE INPUT SCREEN

```
INPUT log_number
INPUT customer
INPUT phone
INPUT reservation_date
INPUT number_of_reservations
INPUT amount_paid
INPUT amount_due
INPUT purchase_date
INPUT seat_assigned_1
INPUT seat_assigned_2
INPUT seat_assigned_3
INPUT seat-assigned_4
INPUT seat_assigned_5
INPUT seat_assigned_6
INPUT seat_assigned_7
INPUT seat_assigned_8
INPUT seat-assigned_9
INPUT seat_assigned_10
INPUT seat_assigned_11
INPUT seat_assigned_12
INPUT seat_assigned_13
INPUT seat-assigned_14
INPUT seat_assigned_15
INPUT seat_assigned_16
INPUT seat_assigned_17
INPUT seat_assigned_18
INPUT seat-assigned_19
INPUT seat_assigned_20

REPLACE program_number WITH number
```

In this module, the routine is repeated 20 times to check if the user has entered a seat assignment or ticket number in 1 of 20 input areas provided in the input screen, if a seat assignment or ticket number has been entered, the system looks up the ticket number in the "available" relation and stores a "y" for yes in the "issued" data attribute to indicate that the ticket or seat has been issued. The model then prints the ticket.

```
DO A BIG LOOP UNTIL loop > 20

IF Seat_assigned_1      NOT BLANK AND loop = 1
THEN PERFORM START
IF Seat_assigned_2      NOT BLANK AND loop = 2
THEN PERFORM START
IF Seat_assigned_3      NOT BLANK AND loop = 3
THEN PERFORM START
IF Seat_assigned_4      NOT BLANK AND loop = 4
THEN PERFORM START
IF Seat_assigned_5      NOT BLANK AND loop = 5
THEN PERFORM START
IF Seat_assigned_6      NOT BLANK AND loop = 6
THEN PERFORM START
IF Seat_assigned_7      NOT BLANK AND loop = 7
THEN PERFORM START
IF Seat_assigned_8      NOT BLANK AND loop = 8
THEN PERFORM START
IF Seat_assigned_9      NOT BLANK AND loop = 9
THEN PERFORM START
IF Seat_assigned_10     NOT BLANK AND loop = 10
THEN PERFORM START
IF Seat_assigned_11     NOT BLANK AND loop = 11
THEN PERFORM START
IF Seat_assigned_12     NOT BLANK AND loop = 12
THEN PERFORM START
```

```
IF Seat_assigned_13   NOT BLANK AND loop = 13
THEN PERFORM START
IF Seat_assigned_14   NOT BLANK AND loop = 14
THEN PERFORM START
IF Seat_assigned_15   NOT BLANK AND loop = 15
THEN PERFORM START
IF Seat_assigned_16   NOT BLANK AND loop = 16
THEN PERFORM START
IF Seat_assigned_17   NOT BLANK AND loop = 17
THEN PERFORM START
IF Seat_assigned_18   NOT BLANK AND loop = 18
THEN PERFORM START
IF Seat_assigned_19   NOT BLANK AND loop = 19
THEN PERFORM START
IF Seat_assigned_20   NOT BLANK AND loop = 20
THEN PERFORM START

START
OPEN THE available RELATION
REPLACE issued Y/N IN available WITH "Y"

for PROGRAM_NUMBER = NUMBER and DATA =
RESERVATION_date
PRINT THE TICKET STORE loop + TO loop

END OF THE BIG LOOP
```

**Ans.3.**

Training is conducted by three representatives from DEL with assistance from Sarita Kakkar and Anil Abraham. Training is customized for top management, supervisors, operating personnel and direct users. Personnel are advised to make available the time necessary to attend appropriate sessions based on the following training schedule:

1. Top Management Overview Seminars
   Attendees:
   Mukul Gupta, Director, MCG
   Neera Tiwari, EDP Manager
   Seminar Session 1 (Monday 8:00-12:00)
   a Computing Platform Review
   b. STS Design Overview
   c. STS Data Base

   Seminar Session 2 (Wednesday 8:00-12:00)
   a. Administrative and Procedural Controls
   b. Maintenance
   c. Reports

2. Supervisors Group Instruction
   Attendees:
   Suresh Srivastava, Accounting Manager
   Dinesh Malik, Activities Manager
   Balwant Singh, Facilities Manager

   Group Instruction Session 1 (Monday 1:00-5:00)
   a. STS Design Overview
   b. Menus and Screens Descriptions
   c. Logical Relationship of STS Data Base

   Group Instruction Session 2 (Tuesday 1:00-5:00)
   a. Program Module Specifics
   b. Accounting Module specifics
   c. Ticketing Module Specifics

   Group Instruction Session 3 (Wednesday 1:00-5:00)
   a. Procedural Controls
   b. Administrative Controls
   c. File Maintenance

3. Operating Personnel Procedural Training
Attendees:
Anoop Sarin, Ticketing Manager
Nasrin Khan, Principal Accountant
Dalip Dalal, Publicity Manager
Sanjay Sharma, Programs Coordinator

Events Module Procedural Training Session 1 (Tuesday 8:00-12:00)
a. Date Scheduling
b. Advertising and Marketing Requirements
c. Staffing Requirements
d. Reports

Ticketing Module Procedural Training Session 2 (Thursday 8:00-12:00)
a. Seating Assignments and Ticket Sales
b. Ticketing Pricing Structure Setup
c. Financial Inquiry
d. Ticket Sales Inquiry
e. Reports

Ticketing Module Procedural Training Session 3 (Thursday 8:00-12:00)
a. Accounts Receivable
b. Accounts Payable
c. General Journal
d. Reporting

4 Direct Users On-the-job and Tutorial Training
Attendees:
Feroze Ticketwallah, Head Cashier, Ticket sales
Minakshi Dutt, Ticket Seller
Rohan Kumar, Ticket Seller
Tirath Kumar, Assistant to the Publicity Manager
Samir Rawat, Assistant to the Programs Coordinator

On-the-Job Training Session 1 (Thursday 1:00-5:00)
a. Terminals
b. Printers
c. Documentation
d. Online Help Screens
e. Source Data
f. Problem Resolution

On-the-Job Training Session 2 (Friday 1:00-5:00)
a. Program Menus and Screens
b. Ticketing Menus and Screens
c. Accounting Menus and Screens

Ans. 4 : ACCEPTANCE REPORT

Month dd, 19XX
To : All Division Heads
From : Sarita Kakkar, Chief Systems Analyst
Sub : STS Acceptance Report
Copies : Mukul Gupta, Gymnatorium Director and Neera Tiwari, EDP Manager

Consequent to the acceptance meeting held on month dd, 19XX the following points were noted:

a STS module is working fine after the systems conversion.

b All hardware/software components of the system are performing well.

c Training has been imparted to satisfaction of all concerned.

d A five working day formal acceptance tests have been satisfactorily carried out.

Hence it is decided to accept STS from Systems Development Group by MCG from Month dd, 19XX.

Signatory:
For System
Development
Group ________________

Sarita Kakkar

For MCG
All Concerned ____________
Divisional Heads ____________

Approval :

____________

Neera Tiwari

EDP Manager

Approval:

____________

Mukul Gupta

MCG Director

Uttar Pradesh
Rajarshi Tandon Open University

# BCA-07

# Elements of Systems Analysis and Design

Block

# 6

## SAD : Emerging Trends

# BLOCK INTRODUCTION

The first 3 blocks of this course covered a run through of the basic stages of Systems Analysis and Design. In view of the importance of Management Information Systems, especially in the recently emerging globally opened and competitive business environment, a special block was devoted to this topic. The next block comprised a set of case studies, which went through the different stages taught earlier in the context of a specific need.

The history of software development is but 50 years old, and during this short period there have been rapid developments in the hardware, and significant shifts in the approaches to software. The impact of this rapid development is such that the analysts of tomorrow would have to substantially change the way in which their work is carried out. This block deals with some such developments, so that on the strength of the foundations laid in the last 5 blocks, you will be able to cope with the new working environment which would be available to a practitioner over the next few years.

The first unit of this block deals with the attributes which a good system analyst must have in order to do justice to his profession. An awareness is created in this unit of organisational issues, and it also illustrates the impact that law has on a working of a systems analyst.

The second unit of this block deals with human computer interaction. It introduces the what, why, when and where of HCI and proceeds further with the issues of current areas such as communicating with computers, user interfaces and ergonomics.

The third unit of this block begins with a description of the multi-media tools that have recently become available, and the basic elements that go into the design of a multi-media based application. The systems analyst of tomorrow will hav· to work in an environment where the multimedia P.C. units be most common on the desk top.

# UNIT 1 THE ANALYST AS A PROFESSIONAL

## Structure



## 1.0 INTRODUCTION

In the previous units, we have gone through the various stages of Systems Analysis and Design. This unit shall take you to the general but very interesting and important aspects of the multifaceted personality that every Systems Analyst should have. It draws your attention to the knowledge and skills that a systems analyst must acquire, the code of ethics and standards of practice that are expected from him. This unit also emphasises the practical need of the situation i.e. the role of the systems analyst while an organisation goes through different stages in the assimilation of new technology and at the same time planning for averting and coping with disasters. You will also find a reference to the computer and software related crimes and the systems analyst's duty to take care of privacy and data protection while designing any system.

## 1.1 OBJECTIVES

After going through this unit you shall be able to

- enumerate the attributes which a good system analyst must have,
- have an idea of the factors which make it possible to implement a new system in a changing environment,
- appreciate the precautionary measures to be taken to cope with possible of disasters,
- know about the computer crimes and software piracy.

## 1.2 ATTRIBUTES OF A GOOD ANALYST

Having gone through the earlier courses on hardware software and a variety of programming languages, it would have become apparent that system analysis is an activity that requires inputs from a number of directions. The task of a Systems Analyst is not algorithmic in nature, where he can pursue vigorously a series of steps and come out with the appropriate systems.

On the contrary, it would seem that an Analyst is like a juggler who maintains the delicate balance between different objects and the pressures and pulls arising from them to ensure that the show goes on successfully.

In fact, there have been systematic research studies to find out as to who can be a good Systems Analyst.

### 1.2.1 Knowledge and Skills Repertory

In the context of what has been referred to above, G.Weinberg and I. Shemer have summarised that a good Systems Analyst should demonstrate:

> logical ability, mature judgement, thoroughness, practicality, ability to observe, ability to work with others, resourcefulness, dislike of inefficiency, imagination, initiative, oral ability, integrity, intelligence, abstract reasoning, emotional balance, interest in technology, interest in analysis, interest in staff work, writing ability, numerical ability, curiosity, open-mindedness, decisiveness, selling ability, empathy, and intuition;

and be well versed in:

> organisational theory, the art of expression, law, information analysis, the art of interviewing, software engineering, project management, programming, economics, databases, user training.

Apparently, the education of such a person is going to take time, possibly a lifetime. May be even more than a lifetime.

What has been stated above seems to be a tall order, and there could be valid sceptism whether such training can be imparted at all. The natural conclusion would be that on the strength of some technical background as provided by good quality courses and training materials, a person aspiring to become a good Systems Analyst must continue to learn and gather knowledge from associate disciplines as well as keep in touch with the progress in the main stream. There is, therefore, no alternative but to adopt a commitment to life long learning.

### 1.2.2 Attitudes and Beliefs

Because the Systems Analyst is in some sense at the core of the organisation, apart from his knowledge in the computer field and even in the somewhat relevant social sciences, his personality must reflect attitudes and beliefs which further his work and lead to greater success. Some of the important parameters which such a person must build in to his working style, is one of increased efficiency. Since it is not possible to cope with the vast amount of information required to do the job, and the multifarious interactions with the numerous persons, the analyst is well advised to follow certain principles which help in doing work more efficiently. These have been encapsulated by Winston Fletcher in his work on "super-efficiency", as follows :

(a) Persuasive communication
(b) Time management
(c) Stopping procrastination
(d) Dominating data
(e) Innovative Ideas
(f) Travelling creatively
(g) People management

This is not to say that this is the only model which can be followed but to indicate that these should be major concerns and positive effort would have to be made towards inculcating such a style.

### 1.2.3 Ecological Awareness and Green P.C.

In recent times, the consciousness and awareness of the finiteness of the earth, our resources and the rapidly increasing consumptive pattern of society have become a cause of global concern. There is, therefore, now a backlash towards conservationism towards lesser exploitation of our natural resources and towards a more positive attitude to reconstruction and rehabilitation of man-made destruction that may have been caused recently.

The above discussion translated into the context of computer based information systems is to urge the systems analyst to apply this consciousness so that unnecessary consumption is reduced. The consequences of higher rated power supply, hazardous radiation coming through video display units, the use of non bio-degradable materials in the hardware system and so on.

There are as of now no existing standards or regulations in India to enforce strict control measures. In contrast to say the laws for air and water pollution control. Even noise pollution is typically covered only under the general law of nuisance.

For example, it is common practice to suggest that any computer system should be placed only in air-conditioned surroundings. While this may add to the comfort level of the workers and to that extent contribute to some increase in productivity, the overall requirement of the electricity and the heat generated would definitely cause fair amount of thermal pollution. In contrast the storage of documents in digital form through high density compact storage media, results in saving of vast quantities of paper. Savings on this account could be equated to saving large tracts of forest areas from deforestation on account of otherwise having to fulfil the need for paper production. Of course, the fact that the plastic used in some of these media is not bio-degradable is also to be borne in mind.

The concern for the environmental factors has given rise to the emergence of the term "Green P.C." to refer to a machine whose design has borne in mind such considerations. After the lead taken by many international companies in the field, in India too, various vendors have announced their own 'Green P.C.'. They are characterised by consumption of less power and the feature of consuming only say 30 watts in the stand-by mode. It does not use chloro-fluoro carbons in the manufacturing process and uses recyclable packaging material.

**Green P.C.**

**Defining A Green PC**

A green PC consumes less than 30W of power in STANDBY mode. Its distinctive feature of vital importance is that its manufacturing process does not include the CHLORO FLUORO CARBONS which are responsible for harming the ozone layers. It also makes use of the recyclable packaging material.

There are two types of green PCs, viz. **LIGHT GREEN** and **DARK GREEN** PC's. A Dark Green PC consumes 15-20W in standby mode. It spins down the IDE and shuts off display. The use of it reduces system's clock speed to 8 MHZ. Its user can set the Power down timings of it. Also the options available with Hot Keys of it, facilitate the powering down system.

A Light Green PC consumes 25-30W in standby mode, which is more compared to a Dark Green PC. It also spins down IDE and shuts off display. But unlike the Dark Green PC, its IDE spin down timing is fixed.

**Origin of the Green PC**

In the US, PCs account for nearly 50% of all office automation equipment power consumption. A strange fact about the use of these PCs in the US is that these are rarely switched off over there. Hence, even during inactivity the PC continues to draw the same power. In June 1992 the **Environment Protection Agency (EPA)** of USA announced the **Energy Star Programme** to reduce electrical consumption and the resources required to produce electricity. If power consumption on all PCs in the US could be cut down, then they calculated that:

TWO MEGA SIZED POWER PLANTS NEED NOT BE CONSTRUCTED

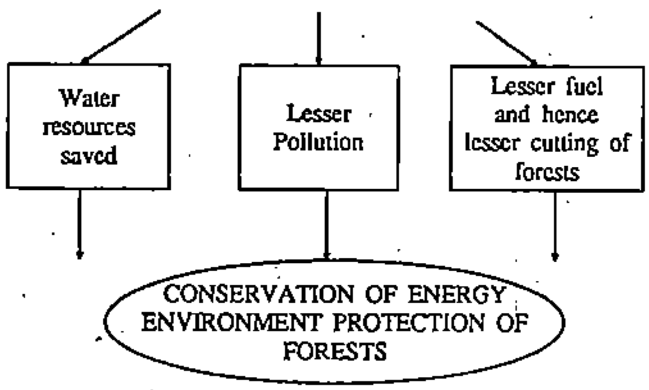



Figure 1

PCs which were compliant with E.P.A. norms would be certified with the ENERGY STAR logo. In March, 1993, President Clinton ordered that within six months all PCs purchased by the Federal Government would have to be Energy Star compliant.

**Designing a Green PC**

**POWER CONSUMPTION IN DARK GREEN PCs**

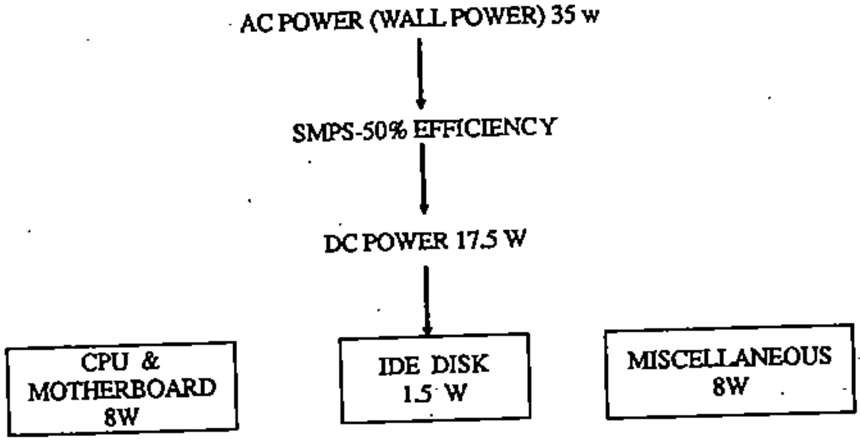



**Figure 2**

1. **Motherboard:** Power reduction is brought about by reducing system clock to 8 MHz (from 33 MHz) (Full power is 25.54 W and Green mode is 7.2 W).
2. **IDE Drives:** Green drives are different from the normal drives. (Seek mode (read or write) is 5.7 W, IDLE is 3.5 W and Green/inactive mode is 1.5 W).
3. **CPU:** An implementation called FREQUENCY SCALING uses an interrupt called STPCLK to reduce operating frequency from 33 MHz to 8 MHs. This is the minimum frequency required to keep the system functions operational. Here the full power is 4.75 W and green mode is 0.80 W.
4. **Display:** Monitors should be designed to be DPMS (Display Power Management Signalling) compatible. Vertical and horizontal SYNC signals in a monitor are blocked for effective power management. Here the normal mode is 60 W A/C and green mode is 30 W A/C.

**Power Down Setting**

There are three options for setting the time after which the PC centres green mode.

**Timer:** Sleep timing is set by the user. The set up has different user definable time (e.g. 5/10/15 minutes) after which PC centres green mode.

**Hot key:** This is a combination of keys (similar to Ctrl-Alt-Del) which power down the PC.

**Fixed:** In light green PCs the IDE powers down automatically after a fixed duration of 10 minutes. The display power down can be set by the above two options.

### 1.2.4 Ethical Issues

All professionals, when they get well established, and become responsible for their actions to society at large, as well as to the direct users of their services, then become in significant measures accountable for their actions. There are well defined qualifications and procedures for entry into the profession, there are bodies which regulate the conduct of the members of the profession and if necessary take disciplinary action including if necessary, debar a member from practice of the profession. This happens in the field of medicine, engineering, law, chartered accountants and so on. The recent emergence of the computer professions has been so new and a computer professional can be inducted from a number of different directions, that strict regulations for the profession have not so far been possible. There is today, no equivalent of the computer 'quack' as is understood for example in the field of medicine.

It is only recently that professional societies at a number of levels have started taking an interest in creating a code of ethics for computer professionals.

The International Federation for Information Processing(IFIP) is a multinational federation of professional and technical organizations (or national groupings of such organizations) concerned with information processing and computer science. There are currently 43 such organizations in IFIP representing 53 countries.

The aims of IFIP are to promote information science and technology by:

- fostering international cooperation in the field of information processing
- stimulating research, development and the application of information
- furthering the dissemination and exchange of information about the subject
- encouraging education in information processing.

Although the IFIP mentioned above has been in existence since January 1960, it has formally attempted to carry out a code of ethics project only from 1988. The project made a detailed analysis and comparison of 21 codes from Members Societies of IFIP and 7 other codes and it is only in 1994 that the code of ethics was put up for approval by the General Assembly.

In our own country, the Computer Society of India(CSI) which is an apex body of computer professionals has also created a code of ethics which it expects its members to observe. There is also a declaration which the members are required to sign.

**Code of Ethics for IT Professionals** (Applicable to members of CSI)

1. A professional member of the Computer Society of India (CSI) shall:
   - organise the resources available to him and optimise these in attaining the objectives of his organisation.
   - use the codes of practice conveyed by the CSI from time to time in carrying out his tasks.
   - not misuse his authority or office for personal gains.
   - comply with the Indian laws relating to the management of his organisation particularly with regard to Privacy and Piracy and operate within the spirit of these laws.
   - conduct his affairs so as to uphold, project and further the image and reputation of the CSI.
   - maintain integrity in research and publications.

**Codes of Practice**

2. As regards his ORGANISATION an IT professional should:
   - act with integrity in carrying out the lawful policy and instructions of his organisation and uphold its image and reputation.
   - plan, establish and review objectives and tasks for himself and his subordinates which are compatible with the Codes of Practice of other professionals in the enterprise, and direct all available effort towards the success of the enterprise rather than of himself.
   - fully respect the confidentiality of information which comes to him in the course of his duties, and not use confidential information for personal gain or in a manner which may be detrimental to his organisation or his clients.
   - not snoop around in other people's computer files.
   - in his contacts and dealings with other people, demonstrate his personal integrity and humanity and when called to give an opinion in his professional capacity, shall, to the best of his ability, give an opinion that is objective and reliable.
3. As regards the EMPLOYEES, an IT professional should:
   - set an example to his subordinates through his own work and performance, through his leadership and by taking account of the needs and problems of his subordinates.
   - develop people under him to become qualified for higher duties.

- pay proper regard to the safety and well-being of the personnel for whom he is responsible.
- share his experience with fellow professionals.

4. As regards the CLIENTS, an IT professional should:
   - ensure that the terms of all contracts and terms of business be stated clearly and unambiguously and honoured.
   - in no circumstance supply inherently unsafe goods or services.
   - not use the computer to harm other people or to bear false witness.
   - be objective and impartial when giving independent advice.
5. As regards the COMMUNITY, an IT professional should:
   - make the most effective use of all natural resources employed.
   - be ready to give professional assistance in community affairs.
   - not appropriate other people's intellectual output.
   - always use a computer in ways that ensure consideration and respect for fellow humans.

**Code of Ethics**
**UNDERTAKING**

I, ______________________________ affirm that as a professional member, I shall abide by the Code of Ethics of the Computer Society of India (CSI). I further undertake that I shall uphold the fair name of the Computer Society of India by maintaining high standards of integrity and professionalism.

I am aware that any breach of the Code of Ethics may lead to disciplinary action against me under the Byelaws and rules of the CSI. I hereby confirm that I shall be bound by any decision taken by the CSI in such matters.

Place:_______________

Date:_______________

_______________
(Signature)

However, such societies and professional groups have only a limited authority in meeting with the requirements. There are occasions where the issues go beyond the regime of the professional society and into the courts of law. Some of these matters are discussed in a later section which relate to the interactions between the system analyst and the law.

## 1.3 ORGANISATIONAL ISSUES

In the previous sub-section, attention was drawn to the knowledge and skills that an analyst must acquire and the kind of attitudes and beliefs that he must inculcate.

Beyond the self, the analyst has to be involved in the organisation. If he is working for a software development company itself, then in fact, he has to go across two organisational cultures that of his own and that of a client. These organisational issues are of direct interest of the systems analyst. The most important of these is the different stages through which an organisation goes through in the assimilation of new technology. It is important to understand these phases of induction in new technology so that the activity is done in a successful manner.

### 1.3.1 Organisational Characteristics

Organisations that may decide to use computer based information systems may vary from small to large mega corporations. The decision making authorities and hierarchical structure prevailing in organisations vary significantly and the systems analysts must be able to appreciate the specific nature of authority in a given organisation. But for most situations, the characteristics which concerned the analysts are related to response of an organisation to the induction of new technology. This has an almost similar pattern across different organisations.

The first stage of course, is the decision to make an Investment in the new technology. This is the result of some feasibility studies resulting in a commitment by the organisation to experiment with a new technology. Computers and other related equipment is bought, the organisation identifies to undertake one or a few development projects to begin with. New staff is sometimes brought in and the potential users undergo a training programme.

At this stage normally, two types of attitudes prevail throughout the organisation, one welcoming the change and expecting a lot of benefits, possibly being over optimistic in their expectations and the contrary current urging continuance of the status quo. Both the streams have their leaders and a balance has to be maintained by curtailing the enthusiasm of the supporters and explaining to the critics the benefits of the expected new technology.

If the first or the first few projects which are undertaken turn out to be successful, and the user groups begin to see an advantage by way of speeding up of their work and increased productivity, there is a clamour for new applications beyond those that were considered in the original plan.

User groups take a greater interest in refining their understanding of the technology and urge management for expanding the application areas.

Depending upon the success of the first phase, the computer professional gives increasing pressure from both directions. On the one hand, an increasing demand for new applications as end users become more excited and computer become widespread and cheap. On the other hand, since the organisation is now become more dependent upon the new technology, there is an increasing problem of maintaining the old applications and seeing that the quality of support is maintained at the previous levels. The result of the third phase is that of the development of precise controls to guide the use of the technology. These controls become necessary to prevent duplication, to promote standardisation and to see that in the enthusiasm and excitement, energies of workers are not distracted or dissipated. Also on the basis of the experience that the organisation would have gained such controls would help in seeing that the later applications of the technology are more cost efficient than the first.

The next phase is an acceptance of the technology by other groups in the organisation and hence a greater pressure to speed up. This may then want networking strategies, E-Mail communication and if an organisation progress sufficiently fast successfully, may be used of Electronic Data Inter-change (EDI) as a means of application to application communication between departments and organisations.

### 1.3.2 Working in Teams

The Analyst as mentioned earlier has to work in a group which comprises for computer professional usually programmers and also a management and administrative representatives of the organisation. As the size of the project increases, even the computer group itself form a sub-team and they must comprise certain styles of functioning in order to be effective. We of course, begin with the assumption that the members from this team are competent and efficient in their respective areas of functioning. The kind of the structure that is successful for the software development activity has been adverted to by Edward Yourdon, who is regarded as a Software Guru. His conclusions are that a successful software team typically needs the following different kinds of people. Firstly, you need a chairman or leader. Then you need a visionary - the architect who has got the entire system in his head. You need a sceptic. You need a provocator, someone to provoke radically new ideas. You need an ordinary worker. A lot of projects, to be successful, need a scavenger - a resources manager who can get things for the project in the right time.

And then you need a diplomat, somebody who can sense the mood of the team members. With software people there are often emotional arguments. Some people believe that women are more suitable in that role. They are aware of those nuances. Men are often very macho and aggressive. And lastly, you need the completer, whose primary passion is to see the end of the project—who is desperate to bring in all the pieces together. A lot of other members of the team see the project as a lifetime exercise—as if it will go on for ever.

The visionary has to be a technical person. The chairman who gives leadership and direction doesn't necessarily have to be one, but the visionary has to be.

The above observations of Yourdon deserve to be imbibed by any practitioner, because in the absence of a well knit and cohesive team, group work instead of proceedings towards completion is more likely to dissipate resulting in an infructuous endeavour.

### 1.3.3 Disaster Recovery Planning

It is common for an analyst to be so much enthusiastic and excited about conversion from an existing manual system to a computerised information system or re-engineering an existing information system to put in place a more appropriate information system, that the scenario when the new systems are in place and are heavily depended upon tend to be lost sight of.

As users begin to rely more and more on a new efficient information system, they discover to their chagrin that non- availability of the system almost spells disaster for the organisation. Of course one anticipates possible threats to a system and provides for elementary remedial measures such as adequate stabilised power supply, backing up of software and data and so on. However, inspite of the preventive action that may be contemplated, there are always events beyond one's control. There are a number of examples even in the developed countries which illustrate the fragility of advanced planning against disaster.

Hurricane Hugo, the California earthquakes, the AT&T brownout, the Chicago floods, the Hinsdale telephone switch fire and the Penn Mutual and First interstate fires are all examples of natural disasters beyond human control, which have destroyed many company's information systems and in a number of cases resulted in the termination of the business itself. Developing countries such as India, which are rapidly moving towards dependence on computer based systems for their work, are even more prone to such disasters. The Latoor earthquake, or for that matter, the earthquake in Uttar Kashi took place in non-commercial areas. The impact of the kind of Bomb Blast that took place in the Bombay Stock Exchange can easily be seen to have effected the fortunes of many in a few years from now. The importance of Information System Disaster Recovery(IS-DR) planning, therefore, cannot be under-mined. The purpose of this discussion is to draw attention to this issue and to give some pointers as to the manner in which such a plan could be formulated.

The financial implication of the proposed disaster recovery plan must be borne in mind. Of course, some non-quantifiable aspects such as the reputation and goodwill of the organisation may also have to be assigned notional financial values. We do not have any specific analysis for the Indian context, but it has been suggested by experts in the field that a manufacturer and distributor in the developed world whose gross sales are in the range of $200 million annually, will lose more than $100,000 after 4 days of being deprived of Information Systems Services and $1 million after 10 days. It has also been estimated that almost half of the firms which fail to recover within a period of about 10 days will never recover at all and possibly go bankrupt.

Having thus seen the importance of time in the recovery plan, it is important that the organisation is able to prioritize its recovery needs. Although some detailed contextual analysis would have to be done to determine the priorities, a rough and ready guide is to classify recovery needs in the following four classes :

1. Critical – Critical needs are those which are absolutely essential to the running of the business and because of their high complexity cannot be replaced with manual methods.
2. Vital – These needs are those where the organisation could somehow continue for the better part of a working week, that is 4-5 days but the services must be restored in that time. These can not be replaced by manual operations at all but the organisation could bear with them for some time.
3. Sensitive – These belong to such tasks and operations that are more efficiently performed with the help of computers, but with some difficulty could be performed manually as well. For example typing of some kind of letters and making of indents and orders.
4. Non-Critical – Non-critical needs are those whose effects are seen over larger time scale of several weeks. If the services are restored over that period, externally the organisation is not much the worse for the breakdown. There would of course, be some loss in money terms and possibly in terms of the quality of services although it is not likely to lead to total business failure.

Having identified the nature of the different information system, the needs of the organisation and their criticality, the disaster recovery options would have to be exercised.

There are more or less five options that are available to provide services for a site which has been afflicted with the kind of disaster referred to in the earlier part of the section.

(A) Hot Sites

The term hot site is used for a site which is almost a replica of the afflicted site and is commercially available 'on demand'. The advantage of such an arrangement is that it is very fast and a continuity of operations can be maintained fairly easily. The pre-dominant disadvantage is that it is likely to be quite expensive.

(B) Private Hot Sites

In the absence of the availability of a hot site, or the decision not to use such a site for confidentiality or other strategic reasons, some organisations establish their very own hot site for such emergencies. The biggest advantage is that there is no dependency on any other organisation and, therefore, no sharing of resources. Naturally, the biggest advantage also flows from this and that is that it would cost exactly twice cost of the original site. Since this site would not be used very often, it effectively doubles the cost of the capital invested in the business.

(C) Warm or Cold Sites

Such a site does not have the entire physical equipment in place and is essentially a shell that is ready to be installed with the required equipment. In some cases, this may even be a mobile unit. There would of course, be a separate contract with a supplier to supply within the specified time frame (on an emergency scale) the required hardware and software. This would be a cheaper location, but the time required for the services to be up and ready is greater. Of course, it is assumed that the most critical software and data have been archived in a safe manner at a site sufficiently far away from the afflicted site.

(D) Service Bureaus

Service Bureaus usually could not be a good solution for rapid recovery in an emergency. Most service bureaus would manage their operations on a scale which leads to an almost complete utilisation of their hardware, software and human resources. It is extremely difficult to visualize the situation where a service bureau would be able to provide the level, quality and timely needs of services that would be critical to an organisation. Of course, if a service bureau arrangement would be workable, it would be the cheapest solution.

(E) Reciprocal Agreements

Such agreements are made between two organisations, who may have used a fairly similar strategy in their information systems planning and implementation. It would be a first requirement for such arrangement to be workable that each one's systems would run on the other one's infrastructure. If this is so and both the parties are able to provide that much spare capacity in the eventuality of a crisis in the other organisation, then this usually would be the most desirable of all solutions. However, this advantage must be weighed against the possibility of both the organisations being simultaneously affected. Such a situation can arise during flash floods, cyclones, earthquakes, etc.

## 1.4 THE SYSTEMS ANALYST AND LAW

It is not immediately apparent that a person who has equipped himself with knowledge of computers and hopes to be involved in the development of computer based information systems would have anything to do with law as part of his work. However, as the penetration of information systems into organisations increases and their utility becomes apparent and dependance on computer based systems for provision of a variety of services becomes routine, the impact of the law is also felt.

This section draws the attention to those aspects which would concern the systems analysts and awareness and understanding of which would hold him in good standing.

### 1.4.1 Software Piracy

What is Software Piracy?

The PC industry is barely 20 years old. In that time, both the quality and quantity of available software programs have increased dramatically.

Although approximately 70 percent of the worldwide market is today supplied by developers in the United States, important development work is carried out in scores of nations around the world.

But in both the United States and elsewhere, unauthorized copying of personal computer software is a serious problem. On the average, for every authorized copy of PC software in use, at least one unauthorized copy is made.

Unauthorized copying is known as software piracy, and in 1993 it cost the software industry in excess of US $ 12.8 billion.

Software theft is widely practiced and widely tolerated. In some countries, legal protection for software is nonexistent, laws are unclear or not enforced with sufficient public commitment to cause those making unauthorized copies to take legal prohibitions on copying seriously.

Significant piracy losses are suffered in virtually every region of the world. In some areas the rate of unauthorized copies is believed to be in excess of 99 percent.

Software piracy harms all software companies and ultimately the end-user.

Piracy results in higher prices for honest users and reduced level of support for customers.

**Why is Piracy so Prevalent?**

Software presents a unique problem because it is so easy to duplicate and because the copy is often undistinguishable from the original. Unlike other works, such as audio and video tapes, there is no degeneration in quality from copy to copy.

A program that reflects many years of effort by a team of software developers and large investments in money, takes only seconds to copy. Although software is expensive to develop, it costs little or nothing to duplicate and virtually any PC can be used to make unauthorized copies.

Software piracy takes many forms. The reasons for unauthorized software copying range from sheer carelessness, lack of awareness of the law and general disregard for the importance of treating software as valuable intellectual property.

Even in the "best" countries, piracy remains a significant and extremely costly problem for the individual, local economics and the software industry as a whole.

**Forms of Software Piracy**

The forms of software piracy seen around the world include:

**1. Hard Disk Loading**

Hard disk loading occurs when computer dealers load unauthorized copies of software onto the hard disk of personal computer as an incentive for the end-user to buy hardware from the particular dealer.

These dealers do not provide original disks, documentation or the end-user license agreement that comes with a legitimate copy of the product.

Hard disk loading is a wide-spread problem, but this form of piracy is easy to detect.

Some end-users unwittingly receive illegitimate software already installed, but the absence of disks, documentation, registration forms and software licensing should alert them to the problem. Hard disk loading leaves enough evidence to make prosecution straightforward.

Industry focus on hard disk loading over the last two years has begun to reduce the practice in countries around the world.

Dealer cases involving hard disk loading are often the first enforcement efforts undertaken in many countries due to the relative ease of proving infringement.

**2. Softlifting**

Unauthorized copying of personal computer software within organizations, also called 'softlifting', occurs when extra copies are made within an organization for employees to use in the office, or to take home.

Disk swapping among friends and associates outside of the corporate environment is also included in this category. Software is often copied from the corporate work-place and distributed to friends outside the work-place, but other sharing of software is also quite common.

Unauthorized copying of software within organizations is the most pervasive form of software piracy faced by many software publishers. It is estimated to be responsible for more than half of the total revenues lost by the personal computer software industry worldwide.

This practice is widespread not only in corporations, but also within institutions such as schools, public agencies, government offices and nonprofit organizations.

**3. Software Counterfeiting**

Software counterfeiting is the illegal duplication and sale of copyrighted software in a form usually designed to make it appear to be legitimate.

Unlike corporate violators, software counterfeiters operate purely for profit. Counterfeiting occurs in all regions of the world, but the problem is particularly acute in Pacific Rim areas.

Counterfeiters range from individuals running mail-order operations out of their homes to dealers who duplicate and sell software.

There are several form of counterfeiting. One involves the entire copying of the product package, whereby end-users are deliberately misled into thinking that the product they are buying is the legitimate product from the original source.

Another form includes the sale of illegal duplicated software marketed under a completely different name, with no attempts made to represent the copy as having been distributed by the original software developer.

**4. Bulletin Board Piracy**

Downloading copy-righted software to users connected by modem to an electronic bulletin board is another form of piracy.

Piracy of copyrighted software via electronic bulletin boards should not be confused with sharing public domain software or providing 'shareware'.

Shareware is software that may or may not be copyrighted but is specifically offered by the author for nearly unrestricted use, including copying and sharing with others, with usually a small fee given to the developer if the user finds it useful.

**5. Software Rental**

The industry has encountered three forms of pirate software rental; product rented from a retail outlet for use on the renter's home or office computer, product rented through mail order 'clubs'; and product installed on computers which are in turn rented for temporary use.

Those establishments which rent software only, whether a retail storefront or mail order operation, operate in much the same way as video rental stores.

The customer chooses software, pays a small sum, and takes the product away for a limited period often, to make another copy for permanent use on their personal computer.

Some companies have asked for and been granted permission to provide software with rented computer hardware, as there are clearly circumstances under which such arrangements are legitimately required.

The explicit right to restrict rental is unfortunately often unclear or absent from national copy-right laws.

**Forms of Agreement**

There is obviously a large variety of possible relations between software developers and users. The right which a person acquires upon purchase of a software product is not absolutely ownership rights in the sense of some other products, but a right to its usage while complying with certain conditions. It is in the sense that the word licencee is used when we refer to a purchase of software product. This is not the appropriate place to have an exhaustive discussion on how such contract should be framed but for giving a feel of the kind of issues that can be involved in a typical software release agreement is given below. In order to keep this general, the acronym NOSP has been used for the Name of the Software Product and NOSS has been used for Name of the Software Supplier. It is common practice in such matters of contract to have 2 copies signed and one copy retained by the licencee and the other by the licensor.

## SOFTWARE RELEASE AGREEMENT

**Licensee's Copy**

The undersigned, being an authorised person representing the Institute or Company named and to be referred to as the Licensee, accepts the software and associated documentation known as Name of Software Product (NOSP) and agrees to the terms and conditions as laid out in the Terms and Conditions of Software Release Agreement. The Name of Software Supplier (NOSS) to be referred as the Licenser, grants the Licensee a non-exclusive, non-transferable licence to use NOSP on the specified computer at the named address. New releases are not automatically covered by this licence.

Two types of licences are available for the NOSP system. One is the licence for a limited use of one year from the date of purchase of NOSP. The other is the licence for permanent use.

| | | |
|---|---|---|
| Licensee | : | |
| Institution/Company | : | |
| Authorised person | : | |
| Licence type | : | |
| Machine type | : | |
| Address | : | |
| Signature | : | |
| Date | : | |
| | | For Official Use |
| Licensor | : | NOSS |
| Authorised signature | : | |
| Date | : | |

Note : Please complete this form and return it to NOSS. This will be returned to you with the signature of the authorised person from NOSS.

## TERMS AND CONDITIONS OF SOFTWARE RELEASE AGREEMENT

**NOSP**

**1. Prerequisites**

The licensee is responsible for obtaining any further licence that may be necessary to provide the computing environment required such as MS-DOS, UNIX and MS-WINDOWS.

**2. Non-transferable rights**

NOSP will be used only on machines which are located at the address of the Licensee as filled-in in the Software Release agreement. The Licensee shall not distribute NOSP or any part of NOSP to others.

**3. Software Protection**

The Licensee shall take all precautions to prevent copies of NOSP being made. These precautions shall be equivalent to those employed by the Licensee to protect their own documents and software from being copies.

**4. Non-exclusive rights**

The Licensee recognises that NOSP is released on a non-exclusive basis and that the Licensor shall have the exclusive right to grant licences to others or make such use of NOSP as it shall desire.

5. **Credits**

All credits in NOSP, both in listings and/or documentation, whether names of individuals or organisations, will be retained in place by the Licensee. The Licensee will acknowledge in any published documentation or in any other use of NOSP the authorship of NOSP and the fact that NOSP was developed by the Licensor.

6. **Product Warranty**

NOSP is released on an "as is" basis, and there is no warranty expressed or implied as to the functioning or performance or effect on the hardware or other software. The Licensee recognises that the Licensor is not obliged to provide maintenance, consultation or revision of NOSP.

7. **Liability**

Neither NOSS nor the individuals responsible for the development and/or maintenance of NOSP accept any liability for indirect, consequential or special damages of any kind.

8. **Future Releases and Versions**

New releases will be announced if and when there are major changes. New versions will be announced when there are improvements of relatively lower significance. Version numbering will be x.0, x.1, x.2, etc. This licence does not automatically entitle the Licensee to updates of NOSP when they become available. The Licensor, however, will inform all those who have a NOSP licence about updates within the same release and the charges for these updates,

## 1.4.2 Civil Liability

The concept of intellectual property in general have undergone a tremendous change since the last few decades in India as well as the world. New issues have emerged in this field in the recent years. Traditional intellectual property laws were enacted either to safeguard the rights of the authors, or to put a stop on its illegitimate use and in cases of patents to assure proper use of the discovery. But the rule of the game has changed. Intellectual property right owners now have a duty to assure that others are not harmed due to their work. Due care and precaution has been added as the liability of an author. Previously such sanctions were available with the State only and that to in a limited sense as it was used in cases where it opposed to the public policy.

Examples of such change in regulation can be seen in country guide books that carry the caption "the author is not liable if the reader suffers loss due to the information provided in this book". Such captions are followed on cookery books and other work. There is complete paucity of case law in this field in India but the message is very clear that if the author can lead the reader in a situation where his life or property is endangered, he has a duty to compensate for such loss.

This new line of thought has become more popular due to the fact that such danger of loss is far more in the electronic media and has seeped from there to the normal book writing. There is no doubt that the electronic media is much more sensitive and more prone to causing loss. Take for example the case of computer virus, where a few line of information can completely paralyse a computer network. The authors of such virus are liable for the loss their intellectual property have caused to others. Similarly, a computer program designed to cheat non- active accounts to transfer large sums of money is not only a criminal theft or cheating but also carries with it the civil liability of compensation.

The question of law is becoming more and more important as software becomes internalized in many computer based products. Modern VCRs, Washing Machine, Micro-wave Oven etc., all have software of a few kilobytes which is an integral part of equipment. In malfunctioning, a part of the software would have a direct bearing on the functioning of the gadget and if not under principles of conventional civil law, atleast under the new found umbrella of consumer protection, the manufacturer would have to take responsibility for the error in the software. Eventually, it is the system designers and developers who would be answerable.

The usage of Intelligent systems in context beyond experimental and laboratory situations also creates significant liability issues. An expert system based product which helps in analysis of a medical case or a financial expert system product in making financial investment or for that matter a Project Manager implementing a project would also create a liability on the developer of such a system. As a hypothetical example, should the Bhopal Gas disaster have taken place because of a malfunctioning in the software for process control, liability and responsibility would eventually have to be fixed upon the person responsible for that piece of software.

### 1.4.3 Computer Crimes

In the previous sub-sections, attention has been drawn to the property aspects of computer software and hardware, which is not always obvious at first sight. Just like theft of any other property, there could be theft of data, software and other intellectual property and components of a computer system. In countries, like the USA where Computers have been in large scale commercial use for quite some time, the American Bar Association have been collecting data on computer crimes for the last 20 years. An interesting conclusion which comes out of such studies is that, a traditional bank robbery results in a typical loss of about $5,000. Although, electronic thefts are still few and far between, the loss due to an electronic heist is $500,000. The corresponding figure for India, is Rs.1.2 lakh for a traditional bank robbery and there is not enough information yet on electronic larceny.

Computer crime can be described as one or any combination of the following events in a computer environment:

- Unauthorised attempt to access, alter, add, delete or hide data;
- Unauthorised attempt to access, alter, add, delete or hide a program or system;
- Stealing of data or programs in any manner ;
- Unauthorised (physical and/or logical) entry into computer work environment;
- Change or alter the defined systems.

Computer crimes are classified in three broad categories: (i) Data-related crimes (ii) Software-related crimes, and (iii) Physical crimes.

**Software-related Crimes**

Software-related crimes are sub-classified into several categories. Like the legendary wooden horse used in Troy, the Trojan Horse is a spy sitting comfortably amongst the programs and keeping an eye over the usage of the system. It is either a logic routine embedded into a key program or a separate program, can be a Terminate and Stay Resident (TSR) program, which monitors and/or records the usage of the system.

Trap door is a logic used by system developers frequently during the system development stage, for the testing of programs and systems. The developer/programmer writes an escape route to bypass all security checks, to save extra keystrokes and time, and thus enter the system or specific programs directly. Very often, system developers deliberately leave these trap doors for future usage in case the user forgets the password, or to hedge against future payments from the user.

Typical losses due to computer crimes include money, goodwill, image, quality, service, competitive edge, and credibility. Some of the debilitating effects on an organisation due to computer crimes include-leakage of sensitive data, operations coming to a standstill, corruption of data, blackmail, communications breakdown, tampering of programs, industrial espionage, etc.

While importance of the kind of the crimes that have been mentioned earlier are esoteric and require a high degree of technical expertise to accomplish them, there is one kind of criminal activity which is highly pervasive and to which a lot of attention is being drawn. A good system analyst must be very conscious of unwittingly becoming a party to software piracy, a term which is used to convey the unauthorized copying of data or software. In the early days of usage of pirated software, the companies whose software was copied in an unauthorized manner did not react strongly enough because the expense of attempting to curtail, to go to court or even to educate against software piracy would not have been cost-effective.

But in the recent past, things have changed. The biggest software companies such as Microsoft, Lotus and Autodesk have joined together to form Business Software Alliance (BSA), which is taking very active and firm action to curb the software piracy in India. According to BSA, India has a piracy rate of 76%, which is much lower than Thailand at 99% or Malaysia with 98%, China with 95% or Taiwan with 94%.

However, there is no scope for smugness because these figures must be compared to US which has the rate of 35% and countries like Canada, Australia, Western European countries where it would be between 35 to 50%.

The System Analyst has an important role in the curbing of software piracy by advising that software is something which ought to be paid for. The feasibility of any project ought to be determined by a full comprehensive look at the hardware cost, the software cost, the software maintenance cost and the cost related to creating a secure system.

### 1.4.4 Privacy/Data Protection

The issue of Data Protection, or Privacy, or Confidentiality and Control of Data, is one which is not usually raised as a concern for the Systems Analyst. This is probably because it tends to apply to all records whether paper or computer based. The issue has however been perceived as more significant in recent times because of the ability of computer systems to store, manipulate, retrieve and correlate (match) quantities of data that were unthinkable in paper based systems.

As there is increasing emphasis in the world on this issue it was felt that students should have some exposure, albeit a fairly superficial one, to the issues, and Government and community concerns and expectations.

In India as of now, any action towards a Data Protection Act does not seem to be in progress. However, since dependence of Software and its usage in business may involve data flow through several nations, it is desirable that one is aware of the existence of such data protection act while one's data is flowing through such a country.

**The Issue**

Systems Analysts developing information system must be sensitive to issues of privacy and confidentiality. Whereas the former tends to be a concern of citizens, Governments (often under pressure from their citizens) are becoming increasingly aware of the need to prevent improper collection and use of data relating to citizens or businesses. Data protection laws enacted by Governments seek to safeguard the personal privacy and corporate information. Other motives are sometimes attributed to such moves including accusations of "economic protectionism" where Governments have sought to limit cross-border data flows.

India has traditionally been a country where information is controlled and regulated. There are a number of provisions in the law, which permit the state to curtail the free flow of information including the seizing of books, newspapers, magazines etc. The right to privacy, in a form other than the right to a good reputation (which is available in law of defamation) is otherwise not very well enforced. To some extent, issues such as right of access to information may come under the broad interpretation under the Constitution of the right to life, but there is still greater emphasis on control and confidentiality of information than openness of information as a right to its citizens. The greater use of computer and information technology would probably change this, but it is not clear at the moment whether the government would steer more rights for itself or a greater freedom of access to information for the citizens. Even the recent Ordinance regarding Cable Operators puts a number of restrictive and onerous conditions on what they may offer to their viewers.

**The Duty of Care**

Allowing access to data by unauthorised persons either deliberately, accidentally or negligently may result in an organisation or person facing commercial or legal penalties.

To avoid claims ("liability in negligence") organisations need to:

- implement security procedures that are reasonable in all circumstances;
- specify what security procedures should be implemented, including standards for creating and storing records;
- insert a clause into the relevant agreements for failure to meet reasonable standards; this will however be viewed with suspicion by the courts, and, if inserted by the dominant partner, may be regarded as evidence of "unconscionable conduct", and therefore not enforceable.

The UK which took a lead in creating a Data Protection Act in 1984, enjoined a duty of care for information held in electronic form upon the custodian of such information.

**Legislative Trends**

Many countries are concerned with privacy or data protection. In Europe 17 countries have enacted forms of data protection legislation.

It was recently reported that the Law Reform Commission in Hong Kong has spend three years drafting new legislation covering the area of data protection. This has been accelerated by developments in this area by the European community which seek to limit data transfers to countries that do not have data protection and privacy laws as stringent as those in force in the EEC. Hong Kong has recognised that any restrictions on information flow could adversely affect trade. The practical effects of the bill are to create an Office of the Privacy Commissioner to which all registered companies would have to submit returns covering "the main features of the data that they hold".

The commissioner would have the power of inspection, in the event of a complain relating to the unfair use of information or invasion of privacy, or on his own recognisances if there is a suspicion of "wrong doing". The Hong Kong definition of data incorporates paper, as well as computer files.

It should be noted that what is viewed as the extremism of the Hong Kong and European draft data protection directives have attracted criticism from various quarters.

In Hong Kong there were concerns that the proposed legislation was "so wide they might be open to abuse in some cases, too restrictive in others and could be too costly for many companies to implement".

In Europe the criticism has centred on the proposals being "too extreme", getting the balance wrong between privacy and public policy objectives such as "freedom of information", putting too much emphasis on the "holding of data" rather than the "abuses arising from the use to which the data is put" and being too bureaucratic, onerous and costly. Other issues raised related to what "the status of intra-company data exchanges" might be, the concern that the directives could "have serious implications for data exchange with parties in third countries" and that "increasingly important economic activities within the EEC such as broad-casting, direct mailing, market research, credit assessment might be seriously affected.

In the UK there is now legislation to control access to personal data through the UK Data Protection Act 1984. Whilst there is yet no definition of what constitutes "appropriate security measures" under Principle 8 of the Act, these will be forthcoming. Much has been written by security consultants on what constitutes acceptable access controls in operating systems. Their views will no doubt be highly relevant to the definition of appropriate security measures. Users have different access control requirements and are subject to varying financial limits. However, by building in the facility to enable implementation of security processing, the degree of control can be varied to accommodate both individual users and future requirements.

India does not seem to have made any move yet towards a data protection act, but in keeping with the international trends, it will possibly be having one in the future. Also in view of the tradition of English law, it is likely to be closest to the corresponding UK Data Protection Act 1984.

## 1.5 SUMMARY

In this unit, a number of concerns have been addressed, which are not readily apparent but are likely to become more important in the future. The unit begins with an exhortation that a systems analyst's training is never finished. Since an analyst does not work in isolation, some features of teams and organisation characteristics have been discussed. Systems have to be reliable but if the systems fail, anticipation of needs in case of disastrous failures have been drawn attention to. Issues related to law such as arising in software piracy, computer crimes and privacy, data protection have also been drawn attention to, so that the analyst is aware of these considerations while working on the development of appropriate systems.

# UNIT 2 HUMAN COMPUTER INTERACTION

## Structure



## 2.0 INTRODUCTION

In designing computer systems, it must be borne in mind that they are meant for use eventually by humans, and therefore, the way in which the system interacts with humans is of great importance in the design of such systems. In the initial stages, when the users were few, this was not important but as information technology is becoming ubiquitous, the need for more "humane" interface is gaining attention in the industry. Till sometime back the interface was thought as going beyond data entry to direct data capture using OMR, OCR, etc. and in terms of pen-based input, speech and gesture recognition, possibly a variety of languages such as English, Hindi, Japanese, Chinese, etc. and other input/output technologies. However, in recent times, a realisation has come that the top layer of the software programs has also a major component. Since this is the layer experienced by the user, it is very important. The success of Apple Macintosh and Windows have demonstrated this and most software designers are giving utmost attention to it.

This unit is of course a brief overview of some of the issues involved and by no means exhaustive.

## 2.1 OBJECTIVES

After completing this unit, you will be able to :

- appreciate that computers are meant for use by humans;
- realise that there cannot be a single user-friendly interface;
- appreciate the relevant factors in designing the human- machine interface;
- appreciate the human problems in an organisation moving towards information technology; and
- give adequate importance to ergonomics in the design of computer-based systems.

## 2.2 THE WHAT, WHY, WHEN AND WHERE OF HUMAN COMPUTER INTERACTION

Human Computer Interfaces (HCI) are tools for helping human beings and machines to work together more effectively. Every tool used by human being has a human interface. When one rides about in a Bicycle, the pedal and the handle are the interfaces through which riding the bicycle is possible. In the case of a car, this may change to the steering wheel, the brake, the clutch, the accelerator and the speedometer as constituting the human interface. Even when one enters in a building by opening a door, the door knob is a human interface. For automatically opening doors, the light sensing mechanism would be the human interface.

In the case of computers, the evolution of the technology has taken place at a really fast pace, and in the enthusiasm and excitement of the new emerging frontiers, not much attention was paid to its being easily usable. The inner workings of most computers were initially accessible only to the experts who were well-versed in programming languages. A question was raised by some people in the 1950s itself as to whether instead of training people in low, medium and high level programming languages, is it not possible to design computers that would be able to communicate more directly.

It was only with the dramatic increase in numbers that was brought about during the P.C. revolution that it was possible to seriously address the question. The effect of the PC revolution was the emergence of a whole new set of tools for tasks ranging from Word-processing, personal banking, management, decision support and onwards upto robotic control. The other emerging dimension was that, instead of just a handful of specifically trained programmers being the only users, personal computers began to be used by people from all walks of life. Such as farming, retailing, manufacturing, commerce, entertainment, medicine, politics etc.

HCI, therefore, is about understanding the task, user and environmental factors in order to design systems that can be used effectively by humans. This can be done by understanding the factors including psychological, ergonomics and social factors that determine how people operate and make use of the technology. Having acquired this understanding, the designers have a role in seeing that the computer systems meet the task in user needs. The net result would, therefore, be to achieve efficient and effective interaction.

It is not always transparent and obvious as to how the information should pass between the human and the machine. In complex systems such as computers which comprise both hardware and software, the design of the interface become even more complicated. Many of us have had experience in using gadgets which were not easy to operate and gave problems because the user was not prevented from taking action which was damaging for the equipment. It is not only in the use of gadgets that the human interface may be perceived as difficult. Even in administrative systems such as doing the work in offices, in using forms, in following procedures and in trying to get information or decisions which pertain to one's problem, there are numerous instances of the interface being extremely hostile to the prospective user.

While using computers, when a beginner tries to work through an operating system like UNIX or DOS, he is often intimidated by an unclear and cryptic error message or a system response.

In the context of computer systems, the challenge to the system designer from the point of view of HCI is at several levels. At one level it relates to selecting the most appropriate input devices such as the Keyboard, Mouse, Light Pen, etc. for the task and likewise the most appropriate output display devices, such as video, printed text, printed graphics bearing in mind both aspects of colour and speed. Apart from the physical object used for input, the means of transfer of information, for example, by use of forms, query and answer based languages, natural languages, comments or graphic based structures such as spreadsheets, Icons, Menus etc.

Whenever one tries to see the advantages and limitations of the alternatives of the device methodology that can be used, the approach has to be somewhat qualitative in nature. Of course, HCI is also emerging to be a field in which quantitative techniques are now being used in order to have a better confidence in the decisions that have been taken. In this brief discussion such approaches relying on statistical and mathematical techniques will not be referred to but it is still worthwhile enumerating some considerations that the designers would like to bear in mind when selecting the most appropriate input device.

### 2.2.1 The Task Features

If the task is one where large amounts of the text are to be read from some other document and essentially typed, then the Keyboard is the most useful advice. If a more graphical approach is to be used and a cursor on the screen is required to be moved around and positioned for an operation to take place then the Mouse is a better device rather than using the arrows on the keyboard. If several objects have to be drawn on the screen of the user then the Mouse too would be somewhat clumsy device.

### 2.2.2 User Characteristics

In the same context one would find that young children are more excited and happier trying to use a Mouse. But users who have been using typewriters for quite sometime and are comfortable with the keyboard may still prefer to be typing rather than trying to master a new experience on a pointing device and fumble with the device while learning to use it.

### 2.2.3 Contextual and Environmental Factors

The lighting, the noise, the ventilation and the general comfort in the working place is also an important factor to be borne in mind. Ergonomics is referred to a little later in this unit and is also a major consideration.

The benefits that accrue out of a good HCI implementation is an improvement in the quality of work and often the benefits are largely hidden, intangible and unquantifiable.

But Management all over is now becoming more sensitive to the issue of quality as recent success stories in other commercial and engineering products have proved beyond doubt that eventually the customers perception of the quality of the product is what leads to it becoming a success in the market.

In the preceding unit, it has been pointed out that team activity is an essential part of the designing and developing computer systems. As it is clear from what has been said about HCI expertise, it is actually encapsulation of knowledge obtained from a variety of different disciplines. Somewhat in the manner of what had been said earlier in terms of what constitutes the Knowledge and Skills Repertory of a good System Analyst, human computer interaction also requires an appreciation of computer science, cognitive psychology, ergonomics and human factors, engineering, designing, anthropology, sociology, philosophy and in many cases now artificial intelligence.

As an example, let us consider the use of ATMs (Automated Teller Machines) which are used universally (and now have begun to make their presence in India) also as a device through which transactions can be made by way of a valid card such as credit card. Humans interact with the machines and the machines deliver the cash against credit cards. It was found that quite often the user receiving the cash left the machine immediately after picking up the cash and forgot to take back the credit card. Designers responded to this problem by introducing a change that required users to retrieve their bank cards before receiving cash. This was an enhanced feature to overcome the problem of users who inadvertently left behind the credit cards in the machines. The concept on the basis of which the device was improved was from the principle that a "task" (such as retrieving the credit card) becomes more important to a person if that task is "instrumental" to achieving the desired goal (such as receiving money).

The purpose of the section was to draw attention to the fact that while the computer programmer would tend to look upon a computer based system primarily from the point of view of the functionality of the system, the actual design should bear in mind that it is user centered, user participative, experimental, interactive and user supportive.

## 2.3 COMMUNICATING WITH COMPUTERS

We have experienced on several occasions, that information or instructions given by one person to another person have sometimes been misunderstood and tasks incorrectly carried out. When we use natural language for communicating between human beings, there is scope for ambiguity, lack of precision and possible variation depending upon the context. Natural languages have situations where the same word could have carried different meanings depending upon the context and different words carry similar meaning. Words which sound more or less the same to the human ear can have entirely different meanings. However, because the humans communicate with each other constantly paying attention to the tone of voice, the facial expression, gesture, posture and even the attire adds to the messages being communicated. Even silence can carry messages. This is so because a person at the other end of the communication activity is also capable of human reasoning and interpretation and carries a lot of contextual information and experience.

Communication between humans and computer in contrast is primarily for the purpose of making the system carry out specific tasks. The purpose of the communication is to convey suitable instructions to the machines and observe the machines response to those instructions.

The systems designer, therefore, has to make special effort for making this communication between unequal partners effective and fruitful.

Computers have now pervaded many walks of life and, therefore, the tasks for which computers could be used may include-office procedural tasks, engineering tasks, information retrieval tasks, updating of information tasks, controlling of operation tasks etc. The amount of human interaction of course, varies with the nature of task and some of the considerations that have to be borne in mind while carrying out a task analysis would be :

- to what extent the task is repetitive
- the variation caused by the environment in which the task is performed
- variation between different occasions for performing the task
- the frequency at which the task is carried out
- the knowledge expected of the user who is carrying out the task
- the skills in which the user carrying out the task is efficient
- the criticality in time of the task
- possibility of the safety hazards
- whether the task is done alone or in a group
- whether the task is carried out for a long duration or there is switching between several tasks.

Once these considerations have been deliberated upon and their bearing to the particular system has been analyzed, they must be classified into those tasks which ought to be done by the computer and those that ought to be done by human. Of course, a technology oriented designer would be inclined to automate more and more all the tasks. But an over-riding general principle should be borne in mind that humans should do those tasks that require a high degree of intelligence or dexterity which they can carry out better and the computers should be made to do those tasks which they can perform better than humans.

There is of course, a great variety in human characteristics and attitudes. Individual differences such as gender, age, socio-cultural background, end up in creating a wide range of human participative. It is, therefore, obvious that one design may not suit all. For example, the design of devices which are to be manipulated by hand, must bear in mind the left-handedness or right handedness of the user.

It is an interesting aside to know that although the first impacts of mass technology methods was to force a standard degree of products and a consequent reduction on personal choice, information technology based products have the possibility of greater degree of possible customization and personalization because software can be made quite flexible. Many of today's software has have the possibilities of setting up parameters to suit one circumstances and to that extent customize it and make it more specific to an individual user's needs.

It has been seen earlier that communication of messages from the user to the computer is done by means of input devices, which can broadly be categorized as those which are activated by limb movement and touch, and those activated by speech and those activated by eye or head movement. Apart from the traditional QWERTY keyboard, there are alternative design keyboards and also Mice, Light pen, Digital, Joy sticks, Track ball and data gloves which are used as Data Entry Devices.

The choice of particular device is intimatedly linked to the conclusion of the task analysis carried out earlier. For example, where large amounts of text entry are to be done primarily in a typing move, but traditional keyboard on which large number of trained staff are available would be useful. If more adhoc queries and short elements of text are to be entered may be a keyboard in a dictionary style could be more useful. If the task involved primarily pointing and selecting from specified items than the Mouses are very appropriate.

The traditional approach to writing of computer programmes was one of creation of data files, results being obtained in form of certain output files as a result of processing. These output files could then be appropriately formatted and possibly either on screens or as hard copies on a DOT matrix, Line Printer, Ink- jet or Laser Printer. The movement towards more user friendly communication styles have created a de-facto standard of a dialogue style of conversation between the human being and the computer. A dialogue with the computer, therefore, consists of some instructions which are given in a specified form to the computer which may be by typing a command, pointing with the help of a mouse or for that matter as is likely to be in the future, by simply speaking. These instructions are processed by the computer and the result of the processing is displayed again for the benefit of the user. The dialogues can be carried out in different formats. It may either be small dialogue boxes or it may require the filling of a form by indicating values for different variables.

There is a constant movement towards making this dialogue easier for the user. The traditional command language oriented style, which displayed cryptic and intimidating error messages being replaced by more visually appealing menu driven approaches.

A menu is a set of options displayed on the screen from which the selection on execution of one or more of the options can be carried out, and having made the choice results in a change in the set of the interface. Unlike command driven systems, the user of the computer, therefore, does not have to memorise names or even the first four letters of the various commands. He should be able to understand the functioning of each of the options or these should be listed in the menu. Of course, for such systems to be really effective, the names over Icons used in the display of the menu should be self-explanatory to a typical user with the requisite knowledge and skills for the software. Typically, there is a permanently visible fixed matrix menu. Sometimes pull-down or pop up menus also are used by using the series of them one as what is called as a cascading menu. Although menus may appear at first sight to be extremely attractive and visually appealing, the experienced users may find it easier without taking up lot of steps on the screen unnecessarily. Also being forced to navigate through a menu to arrive at an operation may be not exciting to a user who would later prefer directly typing the specific command. These observations have been made primarily to draw the attention to the fact that, as has been said earlier, different kind of users would prefer different approaches and, therefore, there can not be just one single model or a user friendly interface for all.

Spreadsheets with which you may already have become familiar can also be thought of as an interface which is especially useful in instantly recalculating through a table of numbers to carry out a "what-if" analysis of different situations. Query languages such as SQL which is discussed at greater length on course on 'Database Management System'. Also useful to non-programmers and for making adhoc queries with a rudimentery understanding of the database and the text of the language.

Communication by human takes place through the five senses, and with the availability of digitised pictures and speech and animated images, multi-media provides a very versatile way of displaying and interacting with information.

The student may have heard of virtual reality in which almost a higher level of communication takes place and a user can interact with the system using a data glove. A data glove is a wired glove which enables the wearer to touch objects in three dimensional space. It also conveys a certain feel for the hardness, softness and smoothness of the object that seems to be held by using the glove in this virtual space. The glove can also be used for gesturing. Device like the data glove and computer systems depicting virtual reality are very expensive as of now and they are basically being used for exploratory research programmes. However, like the past experience in other aspect of computer and information technology, the availability of these systems is also likely to undergo an important tremendous change in terms of greater availability at lower costs in the future.

## 2.4 ERGONOMICS

Even before the usage of computers, various other kinds of automation devices have been used in the office and work environment. The typewriter, the telephone, the office furniture, the filing cabinets are all part of a normal work place. It was realised early enough that these must be so devised as to be conducive to long hours of work. The discipline which concerned itself with such aspects of work simplification came to be known as Ergonomics. The system analyst must realise that the basic ideas carry through into the development and use of computer information system as well.

### 2.4.1 What is Ergonomics?

Ergonomics is the name given to a new discipline related to work and work simplification and derives from the Greek words (Ergon (work) and Nomos (law)).

The aims of ergonomics are primarily twofold :

(a) to improve the efficiency of the interaction between the human and machine through enhancing the effectiveness and efficiency of human activities at work or at home.

(b) to maintain the individuality of the worker by considerations of human welfare and human well-being in the design procedure, namely in areas of health, safety, satisfaction and comfort of the worker.

The study of human factors is a multi-disciplinary subject and its approach lies in the systematic application of available data such as human characteristics in the design of human machine systems, work procedures and design guidelines. The human characteristics include the physical and psychological capabilities as well as limitations. A typical set of ergonomics properties relevant to the design and functioning of human machine systems are listed at Table 1. The ergonomic properties in an integrated form determine the overall efficiency, reliability and degree of human involvement in the operations.

Ergonomics essentially is the study of the relationship between the human being and the environmental factors affecting the physiological, psychological or anatomical state. For example, the effect of the lighting of an office on the efficiency of its workers fall in this category. Similarly, the height of the computer operator's chair or the colour of the screen may also be considered as part of ergonomics. Ergonomics considerations play a very important role for designing devices involving physical involvement of human being. The design of the driver's seat, the cockpit of an aeroplane, the location of the kick starter on a motor byke-all are supposed to be guided by the ergonomics factors like comfort, safety, efficiency etc. In recent times, with the proliferation of mechanical/electrical/ electronic gadgets- ergonomics has emerged as a major branch of study. Design of any human operated device invariably takes the ergonomics factors into consideration.

Ergonomics, as stated earlier, also involves psychological factors. The impact of any efficiency enhancing device on factors such as morale, motivation, emotional stability has to be analysed very carefully. A device, as such, may increase efficiency. But, if these factors are overlooked, then it may be difficult to sustain it and in the long run it becomes even counter-productive. Efforts should be made to ensure that the device and the human beings operate in a harmonious environment.

| Controllability | Serviceability | Assimilation | Habitability |
|---|---|---|---|
| Distribution of functions between operator(s) and the human operator playing dominant role | Design for optimal operations, maintenance and reports | Ease of skills for operations and maintenance of, standards for functions under normal conditions | Machine operational factors congenial to human health and work capacity |

**Table 1: Ergonomic Properties of Human-Machine Systems**

### 2.4.2 Ergonomics in System Design

The fundamental approach and emphasis of ergonomics in system design is to consider the human operator as an integral part of the system to be designed. The task components are allocated to the human and machine, in such a way as to utilise strengths and virtues of each and to complement them efficiently through good user interface design. A chart showing the Human-Computer strength and weaknesses related to office information activities is at Table 2. A synergistic combination of the human/computer capabilities is also shown.

| Office Information Functions | Human Weaknesses | Computer Strength | Human Machine Synergy | Human Strength | Computer Weaknesses |
|---|---|---|---|---|---|
| INPUT | Slow limited attention Inaccurate Biased | Fast Voluminous Data Accurate Objective | Computer helps higher volumes of data, Human inputs non-computer readable data | Able to capture in original form, can shift control as needed | Unable to Capture data, not computer readable, all actions, prespecified. |
| PROCESSING | Slow Inaccurate Limited Capacity | Fast Accurate Large Capacity | Human creates alternative options, Computer calculate and analyse | Creative | Prespecified |
| STORAGE | Retrieval Inaccurate | Retrieval accurate Restricted cues but quick retrieval | Computer reduces human memory tasks to a few simple cues, Human provides sophisticated links to cues to enable speedy retrieval | Can create sophisticated retrieval cues | Pre-spec actions, simple Inflexible retrieval cues |
| OUTPUT | Slow Limited Capacity, inaccurate, inconsistent | Fast, High Capacity accurate consistent | Machine enables Human to produce many high quality outputs, Human communicates subtle aspects of computer outputs | Many communication modes | Limited variety of output |

**Table 2 : Human Computer Synergy**

**Check Your Progress**

1. Define the term ergonomics.

   ..............................................................................................................................

   ..............................................................................................................................

2. Mention three safeguards you would recommend during system design to ensure the ergonomic property of serviceability of computer hardware and software.

   ..............................................................................................................................

   ..............................................................................................................................

   (No model answer is given.)

## 2.5 HUMAN PROBLEMS IN THE AUTOMATED OFFICE

The complex structure of information technology changes the way people work. It affects organisational structure, worker's job content, decision making, the interaction patterns, and the physical environment with far-reaching consequences. The assessment of the impact of these changes should encompass the complete operating environment of the system. It should include not only the workers and operating personnel, but also the people like clients, customer and the socio-technical environment which are external to the system. While introducing new technology, due regard to organisational needs and user requirements will ensure smooth transition as well as acceptance and assimilation by the personnel. The projected benefits and productivity gains through automation would not be realised if insufficient attention is paid to human issues.

One critical variable affecting the success or the lack of success in office automation is user acceptance. No matter how technologically superior the equipment is, it is a failure if users reject it. It is noted by systems analysts that the rate of changes due to introduction of technology depends significantly on the degree to which technology will be adopted within the organisation by the different users. The impetus for automation should be focused more on the user requirement and reaction than purely on the technological capability. Therefore, it is essential that the role of office automation should aim at functional-integration of the three components of the system i.e. the people (the office worker and user) the organisation and the technology.

We will, however, confine our study to the human aspects i.e. people and the organisation.

You will see that all the office structures are closely interrelated to the objectives of the organisation which the office supports. Therefore, the systems and procedures in the office will necessarily have to suit the organisational requirements. More importantly, since workers constitute the primary resource in the office environment, human aspects need to be considered with great seriousness. It should be noted that the worker in the office is both an individual and yet a part of the organization and has to be recognised from both these angles. First, let us consider the individual worker.

(a) **Individual Worker** : The first consideration is whether the technology change involves any physically harmful effect on the individual worker. For example, prolonged exposure to VDU (Video Display Unit) terminal has been known to generate mental and physical hazard for the user. Visual fatigue and deterioration of eyesight are also common problem usually caused by improper office lighting and poorly designed VDUs. Many operators suffer from muscular injuries and strains related to poor postural positions due to poor workstation design. Whilst poor working conditions and environment may present immediate hazards some problems may not immediately take effect but only over a long period of time.

The psychological problems of mental health and job satisfaction of workers may also be affected by new technology in the office. Some people who work with computers find that the difficulty to learn and use the machine as being restrictive and intolerable. They may also find that the software tools they use do not satisfy their task requirements. At times, hostility towards using computers may be generated by early traumatic and embarrassing experiences. While starting to use a computer, a user may make some errors that causes situations like wrong billing or incorrect printing of payroll, causing public ridicule. This may very well leave a lasting negative attitude towards the machine.

Changing work patterns as a result of implementing new technology does not necessarily lead to productivity optimization. While the workers try to adapt to new ways of working, changes in procedures may be necessary to formulate efficient working pattern. Work study techniques may be used for such re-structuring. Now let us consider the other human aspects i.e. the organisation.

(b) **Organisational Aspects** : On the organisational side, some workers may perceive the new technology as an adjunct to their jobs, whilst many others may resist possible changes to their ways of working. This is caused due to changes in the nature of job and responsibilities which need additional training and skill. This in turn affects work procedures and the structure of career and reward systems. There may be changes in power, influence and status of personnel depending on the level of access to information. Security and privacy issues have also become important especially when sensitive information is stored in a computer.

In an automated system, work may appear to become depersonalised, fragmented and restrictive. Workers may miss the personal contact with their work and its associated challenges. The boring and repetitive nature of an automated system thus, may cause stress, strain and resentment. Problems, such as these may result in many organisations suffering from —

(i) Increased labour turnover, absenteeism and low motivation

(ii) Increased probability of first errors and accidents

(iii) Increased lethargy and higher medical costs as a result of work related illnesses and injuries.

Many organisations have learnt that improvement in office ergonomics is not just a means of preventing work place from health-hazards but it also makes sound business sense. What is now better recognized is that through adjusting to organisational work environment and improved work stations design, can boost productivity. Office automation need not necessarily have a negative impact if properly introduced and implemented and the primary lacuna of lack of attention towards human issues is overcome.

## 2.6 DESIGNING HUMAN-MACHINE SYSTEMS

An integrated approach is essential to overcome the design complexities of human-machine systems. Different environmental factors affecting the performance of individuals as well as the organisation must be analysed carefully. Apart from this, a strategy must be formulated to test the system before it is adopted for practical applications. Let us consider the environmental factors first.

### 2.6.1 Environmental Factors

The environment of the entire system can be broken down into smaller sub-systems. We will consider these sub-systems one by one.

**a) Human-Machine Environment :** The statement of objectives for a human-machine system design should define the acceptable range of inputs and outputs of the systems together with their temporal relationships and response times. The system has to be separated or broken down into functional components which again are allocated to the human or to the machine. The approach to job allotment for human beings involve a series of activities. These include drawing up job specifications, deciding on the level of skill and the requirement of training as well as the selection process. The next step is to design the human-machine interface. The human and the machine have to be matched to form an integral working unit and close attention must be paid to the area of contact or interface between them. The design of interfaces should combine the strengths and counter the weaknesses of both man and machine in a complementary way to get the best out of the combination.

For example, the information output of the machine must be adapted to the ways in which the human can receive information outputs through visual, auditory or through other means in most efficient ways. Similarly the control input devices of the machine should be adapted to the physical output of the human. The system should undergo through trials and pilot studies for proper evaluation and feedback should be conducted before introducing for practical applications.

**b) The Organisational Environment :** While designing system, apart from human-machine environment, the design must take into account, the organisational environment. An organisation is characterised by personnel who form the user community of the system. During System design phase an early interaction with the user should be brought about and the user should be involved in the design process. Simulations and prototype designs should be tested by users and system performances correlated through tests specifically designed. User involvement is a key element of the design process. The effectiveness of the user involvement depends on the degree and depth of participation and influence the users have on the design process. The traditional approach to design was to carry out the design and implementation through specialist. The dominant theme in that approach is to put together a working technological system. The result is that often user needs are not even taken into account. Other design approaches suggest user participation and specially definition of needs by users in drawing up requirements specifications. User participation and user involvement in the design process does not always guarantee improved user acceptance. It is important that the users are properly represented and the representatives are encouraged to participate in design decisions in an objective way without fears or peer pressures. Users should also be helped to understand the design alternatives, limitations and constraints.

**(c) The Physical Environment :** The preferred approach should be for system design not only to involve practical applications of economic principles but also consideration of the physical environment. The three broad areas which are covered in the physical environment are —

(a) the workstation environment,

(b) the human machine interface, and

(c) the office environment.

The workstation environment covers three important areas of design consideration which are the workspace dimensions, the arm's reach and movements, height and the distances. Work surfaces may have to take into account the nature of the work carried out so as to promote proper working postures. One of the objectives in the application of anthroprometric data in equipment design is to enhance the possibility of maintaining postures.

**(d) Office Environment :** This includes noise, lighting and thermal conditions, office layout and design. Noise has high attention drawing properties. It can affect tasks that require vigilance, speed, accuracy, concentration and memory. Although some noise in office environment is tolerable it should not exceed 60 decibels. If required, noise may be regularised and minimised by using sound absorbing insulating materials such as stands, supports, panels etc. and through physical isolation using sound masking systems and so on. Good lighting increases normal efficiency and reduces visual fatigue. In the video workstation environment, lights should not be directly visible nor in the direct line of vision when viewing the videos, neither should light be reflected off the video screen. The reflecting levels of walls ceilings, and floors are affected by the type of surface. These should be considered while designing office lighting. The environment should be regulated to provide a suitable temperature 21 to 25 degree centigrade and humidity of 50 to 70 percent. A good office layout and design may help to promote a pleasing and comfortable working environment using both special layout and design of open plan offices etc. A chart stipulating some guidelines for ergonomic design is at Table 3. These are broad-based guidelines and only indicate desirable characteristics. These can be adopted suitably in relation to your environment, objectives, weather and the function of your organisations.

**Work Space Considerations**

Work space for operations and maintenance

Optimal space organisation

Levels of natural and artificial illumination for operations and maintenance.

Permissible levels of acoustic noise

Safety devices

Placement of Terminal, detachable Keyboards, Tiltable screens, Working position and adjustable chairs, back rests, movable seats, wrist rests.

Reach

Anthropometric data

**Visual Information** (spatial characteristics)

| | | |
|---|---|---|
| Optimal angles | 30 - 40 | degrees for perspective images |
| Horizontal plane | 50 - 60 | degrees for plane images |
| | 90 | degrees head and eye movement |
| | 180 | (slow) head and eye movement |
| Vertical plane | 15 | degree horizontal |
| | 70 | degrees maximum for eyes only |
| | +90 | head and eyes measurement |

Screen-observer distances 2/2.5 × width of screen (small screen) 8 × width of screen (large screen) orange phospor characters on light brown (amber)background more readable.

**Environment**

**VDU's Lighting should be shielded to prevent reflection on the screen. Use of antiglare shields.**

A clear, stable image should be provided on the screen. Luminescence contrast 3:1 with office lighting.

Lighting Office work typing, lighting 500 lux (× 1.5 for persons over 50 years)

Static Electricity (or Electrostatic discharge)

- Printers, terminals and video display may be affected.
- Use antistatic spray

Temperature : $20^0C$

Health Hazard : Viewing of CRT: 15 minutes break after 2 hours of moderate work. 15 minutes break after 1 hour high demand work

**e) Job Environment :** If human issues are not considered in the design, the impact of new technologies may lead to routinisation in implementation and fragmentation of jobs. Job design is basically meant to promote greater job satisfaction, higher work motivation and better performance. Changes in skill requirements mean that current staff may need to undergo training. User support should help users to adjust, adapt and learn changes in their job responsibility. Rewards, career development and incentive schemes should be devised to accommodate this. The computerisation of manual tasks should not leave the workers with tasks that are dull, boring and insignificant. If new jobs are redefined the job content should continue to be meaningful and result in some significant output. These are usually achieved through mechanism such as job rotation, job enlargement and job enrichment.

## 2.6.2 Human-Machine Interface Design

After the study and analysis of the components of the total environment, let us consider the factors for designing a human- machine interaction.

The design of the human machine interface is one of the most important aspects of system design. A good interface design should take into account the following factors :

a) **User characteristics :** It includes consideration of the kinds of users who will use the equipment, their diverse backgrounds and skills, the user expectations as well as their physical characteristics. The users who posses high degree of skill often prefer more powerful functions which usually means greater complexity. Unskilled operators, on the other hand, would prefer simple functions which are easier to learn and use. Stereo-type expectations also affects the users perception as to how one thinks the system will behave.

b) **Task characteristics :** The nature of users tasks differ and therefore the need for specific kinds of services from the system. For example unstructured tasks usually require a more flexible mode of interaction to meet the varying needs of users as opposed to structured task which are more predictable and repetitive. The sequence and frequency with which certain tasks are performed will also affect the optimal design of the user interface.

c) **Functional characteristics :** It refers to the various functions required to perform the tasks and the ease with which these functions can be learnt and made use of by the users. While ascertaining the functional characteristics, the support facilities required to perform the functions, also need to be taken into account. These include facilities like training, on-line help, documentation, expert systems, etc. The other aspect of functional characteristic of a system is its performance criteria like response time, fault tolerance, etc.

Although traditional applications of ergonomics have been a guiding principle in hardware design, similar design principles are applicable to software design, especially that of the software user interface. The software should provide an interface in the system to enable the user to perform the tasks efficiently and effectively. The human factors guiding the design of the software interface is a challenge to design as the interaction between the user and the system is much more complex than in the static hardware interface. Whilst software interfaces are much more dynamic and interactive, the mental and cognitive processes of the user during interaction still remain as gray areas. Consistent with stability, provision of feedback and minimising the users' mental loads are the most important considerations in defining software user interfaces Good software design should take into account the kinds of people who use the program, the diversified user background, the complexity of use the need to use any other program and the consequences of human error.

## 2.7 SUMMARY

In this unit, you have been able to have a glimpse of who works at the other end of the computer—a human, and ease of use is a major aspect of systems design.

Whether it is input-output devices or the software interfaces, it is brought out that different people react differently. There is therefore no such thing as a single user friendly interface for all.

The relevance of ergonomics in office automation systems has also been drawn attention to, because this is the most common application of computers and affects a large number of persons.

Human-Computer Interaction is now a flourishing topic with inter-disciplinary inputs and support from various industry quarters, as it is not perceived to be crucial to the effective use of computer based systems.

# UNIT 3 INTRODUCTION TO MULTIMEDIA

## Structure



## 3.0 INTRODUCTION

Multimedia is a new technology born, in this world of ever changing and demanding technology and diversifying world of computer. Since last few years, it seems to be much sought after and talked about, not only in the world of Information Technology, but also in various functional fields like advertisement, corporate sector, cinema, fashion design and education, to name a few. More and more research work on this new technology of sound, animation and text, is making it better and better with every passing day. It is one of the most realistic way of working even for a computer literates, where it targets people from almost all ages of life from a toddler to a aged one. In fact it does not have any upper limit for its target audience.

## 3.1 OBJECTIVES

At the end of this session, you would be able to

- understand the basic concept of multimedia
- appreciate multimedia as a new chapter in information technology
- identify and describe various components of multimedia like, sound, animation and graphics
- understand various hardware and software for multimedia
- describe how images and sound works in multimedia.

## 3.2 MULTIMEDIA – THE CONCEPT

You can call it by any name, multimedia, intellimedia, hypermedia or newmedia. In its simplest sense, multimedia means the combination of the text, sound, and graphics. But no body is sure, whether it is a computer itself or a computer software product. In practical sense it is the combination of both. And the fact remains is that it has the best potential to be one of the most powerful form of communicating ideas, searching for information and experiencing new concept of common media ever developed. One might compare it with the conventional form of media which also uses sound, graphics and text like TV, audio etc. Every program you see in TV is a combination of all these. But all these media elements can not be attributed to multimedia, because they do not have the interactive feature. For example a multimedia version of news bulletin would be that where you can request the broadcaster the type of news you are interested in and when you want to hear it. And with a press of a button or a click on mouse you can hear the news.

Non-availability of a wide range of multimedia products in the market might be one of the strong reason that it remains a mystery to the general public. But if you think a bit seriously, you can see one way or the other, multimedia has started creeping into our day to day life in various form, and information has started becoming available; in the digital format. And when the information is stored in the digital format you have lot of flexibility in handling it. It can be edited according to specific requirement and produced in various style and taste.

Use of multimedia in various application can be justified by the levels of the user friendliness, it provides to its user with its interactivity and customization. Multimedia is widely used now a days in various applications and business is one of them. As the market competition is increasing with the introduction of better products in the market everyday, it has become absolutely necessary to provide better service and timely information to your client in a brief, precise and more understandable manner in a short time. It provides various ways to maintain a competitive edge for a company specially in training, market speculation and trend and public relation. The interactive feature of multimedia, brings life back to the business presentations, where one can present various aspects of a business such as, marketing plan for a new product, its impact in the market, consumer reaction etc. simultaneously or even you can combine all these to make the assessment of the consumer feedback on the product launching. The information which is digitised in multimedia makes it more assessable as compared to the conventional way of presentation using video, chart etc. where you cannot edit or modify whatever already video taped. Even if you do it, the cost involved will be high. Now a days many business organizations have already started creating powerful databases that can store and distribute digital media as easily as text.

Another critical application of multimedia is educational sector. It enhances the education quality with its theory of naturalist. It provides new way for teacher to encourage one of the most rare and important element of learning i.e. curiosity to know and explore. Topics can be linked with other related information as possible with graphics, text and sound (may be a lecture session from the subject teacher and so on). With a large screen projector and a multimedia playback system, teachers can use multimedia as a way to enhance their standard lesson plan and stimulate questions. On the other hand students will be able to further explore the topics using standard multimedia platform. And the assignment which requires students to make their own interpretation of the fact, as represented in the multimedia topics. There are several application possible in the area of education using multimedia. To fully explore the use of the information technology in education, definitely the use of multimedia in education would go a long way.

In a complete sense graphics, text and sound is the essence of multimedia and it provides the user friendly environment with most flexible interactive facilities.

## 3.3 MULTIMEDIA – DESIGN, PRODUCTION AND DISTRIBUTION

All the developmental works based on three basic principles; planning and design, production and distribution or the final product, so also multimedia.

Planning/Design → Production → Distribution

### 3.3.1 Planning/Design of Multimedia

During the initial design phase all the aspects of the related multimedia project is carefully thought about and planned well before the production started. Every key element involved is discussed. There are various tools available to plan the project. A flowchart is one of them. It is one of the most commonly used tool to prepare a blue print of the planning, organizing everything during the design phase. It shows how all the elements involved in the project are related to each othe : with adequate information about their controlling features.

Like any other video program or movie, multimedia programs requires a script on the subject involved in the project to work on and to keep track of the elements in the show. The script and flowchart works together to provide a printed version of text, graphics and sound used in the production of the multimedia program.

### 3.3.2 Production of Multimedia

The process of creating the related media elements in a multimedia project including graphics, sound, animation and digital video is called production. In multimedia all these creation of media elements, can even be independent in terms of tools used to produce them. For example graphics elements are created independently using various computer graphics package such as paintbrush, 3D studio etc. These are created and modified accordingly so as to work under various multimedia platform. All the scene of the project is digitised and edited into various sequence and when the digitised video sequence has all the required scenes, titles and transition from one scene to another, they are compressed as to playback in the required speed and pre-defined sequence. Similarly sound and music effects relevant to the project, and digitised scene sequences are created using digital recording system and MIDI (Musical Instrument Design Interface) equipment in a music studio, edited and recorded in the computer as a digital audio file. All textual matter in project are edited and converted into ASCII format and then modifies latter on and placed in the appropriate location within the scene sequences.

One of the most important production task in a multimedia project is regular checking of all the sequences. And this is very true in case of creating a multimedia program for CD-ROM, because a CD-ROM can only transfer data to and from computer at a fraction of speed of a regular hard disk drive.

Once all the related media elements are created and digitised according to the specifications, then comes the final stage of the production process i.e. joining or combining these media elements which is also known as authoring. This process is used to join all the independent media elements such as graphics, text, animation and sound to produce an interactive framework. And all the testing and debugging of the entire package is done at the stage before putting them into various distribution media.

The first step of authoring is to bring all the independent media elements used in the multimedia project, as each of them are developed using various graphics, audio, video and animation tools.

Once all graphics, text, animation and digital video are combined together, different relationship and actions for each of the media element can be worked out by adding interactive control. When multimedia program is executed, the interpreter part of the authoring program takes all necessary commands and relationship that has been defined for each media elements and convert them into binary code or machine code that computer operating system can understand. The converted program is then executed by the CPU of the computer system, which send out instructions to the system to play the music or any audio, display all the text, video and animation and also execute any external commands is defined.

### 3.3.3 Distribution of Multimedia

Once all the media elements are combined together using authoring software they are bundled into package and stored in various storage media for distribution. Creating multimedia package for various platform is one of the most crucial stage that multimedia procedure face today. There are many platform available for multimedia, each of which has their own advantages and disadvantages. To work with full potential, multimedia package need to be available in as many platform as possible. Various storage media used to store multimedia package are CD-ROM, Interactive laser Disk etc.

## 3.4 COMPONENTS OF MULTIMEDIA

As already discussed, multimedia is a combination of sound, text and graphics, which are also the essence of the multimedia. Printed matter continue to be the most common form of conventional way of putting the ideas and with the electronics publishing technology it is possible to produce better printed material but they are static and non-interactive in nature in the sense that sound and motion can not be added to it to give it the interactive feature. But this is possible in multimedia where textual matter can be combined with motion and sound. All multimedia disk contain some amount text, and even some might contain a large amount of textual matter, in various type of fonts and type size to suit the professional presentation of the multimedia software.

Another important and interesting components of multimedia is graphics. One of the very basic facts in multimedia production is that, people do not like reading large amount of textual matter on the screen. And also it is a fact about human nature that a subject is better explained to them when represented in pictorial or graphical form, instead of textual matter i.e. graphics are used more often than text to explain a concept, present background information etc.

Unlike text which is represent in universal ASCII format, graphics does not have any single agreed format. To start with there are two different ways in which graphs or images can be described, Bitmap and Vectors.

### 3.4.1 Bitmap Image

A bitmap image uses the pixel or dots on the screen to form itself and the size of such a image depends on the pixel density and number of colours it uses. For examples for a standard VGA screen uses $640 \times 480$ i.e. a total of 307200 dots or pixels to display an image. If your image is in black and white, then only one digital bit is required to store this information about the image for each dot (0 for black and 1 for white), and the position of the dots can be taken as the order in which they are produced. As a byte is consists of 8 bits, one can store a black and white image of this type in $((640 \times 480)/8) = 38400$ bytes = 37.5KB, so imagine the storage space required to store all the images involved in the multimedia production. If you are using color for the images then the size of the bitmap image becomes larger depending upon the number of color used. A standard VGA which uses 16 colors needs 4 bits of storage to store the information, so only two dots can be coded in a single byte. Hence $((640 \times 480)/2)=153600$ bytes of space is required to store a single image with 16 colors. Similarly for a image with 256 colors (8 bits per dot), 32000 colors (16 bits per dot) or 16,000,000 colors (24 bit per dot) storage requirement will be very high. So a standard image with 16 colors will be ideal solution for a low cost multimedia package.

As discussed, bitmap images are stored as large files and you require large amount of disk space to deal with it. To avoid this constraint, the images can be compressed, which make the use of the fact that many entry in a bitmap file has repeated information or contain very little information. But there is no single standard compression method for image files. There are various formats such as PCX, TIF, BMP, GIF etc., in which images are stored.

### 3.4.2 Vector Image

Other format for storing images is vector or object oriented format. In this format, an images is formed as a set of straight or curved lines instead of dots. A line can be represented by a mathematical equation, whose number can be stored as set of binary codes. Because this form of coding is potentially more economical of disk storage space. The drawback is that it requires software is calculate the number to form the image and also it is very time taking. CAD can be one example for this type of software. A vector image file is always of the same size no matter how large the image is. For visual clarity this type of image depends on the quality of the display media.

There are several technology and sources available to produce images or graphics for multimedia package. Image can be created in Paint program, scan photographs using a scanner and hand drawn artwork, generate 3D graphics and animation using various sophisticated software. You can combined these images using images manipulation program which can combined many different types of graphics files, to create new images.

### 3.4.3 Animation

Animation also plays as a vital part of the multimedia program. The dedicated hardware and software built into the system increases the animation speed. Like a movie, an animation is just a continuous series of still images that are displayed in a sequence. There are mainly two types of animation used in multimedia, namely 2D and 3D animation. 2-D animation which also known as cell animation is the most common kind of animation, where flat images are drawn one frame at a time. This process is very time consuming but result obtained is spectacular in nature. Computer animation has increase the efficient and result of cell animation with introduction of wide ranges of color and speed. In 3-D animation a mathematical model of a 3-D object is created to realistically portray with depth. Now it has become a common media element in film, video and multimedia packages. A 3-D animation follows mainly three steps; modelling, animation and rendering. Of these three first two are crucial and take long time to finish. Modelling is the design phase where a 3-D object is created. In a 2-D animation an object can moved up (called Y axis) and sideways (called X axis). But in the case of a 3-D model, a third axis is used; depth or the Z axis. Once an object is created along these three axis, color, shading and light source can be added to the image to make it more realistic.

In the second phase, the 3-D images is moved along a motion path, which is defined using key frames of the animation sequence. These key frames are used to create the inbetween frames in the sequence automatically.

And in the final stage, the entire sequence in render to create a 3-D animation. Blending texture maps into the model to add realism causes one of the main slow downs during rendering. A texture map is a wallpaper for 3-D models, in which a graphic image is wrapped over the surface of a model. When a 3-D animation program renders an images, it mix up intimately all the color, shade, texture maps, light source and surface attributes in each frame of the 3-D animation sequence.

### 3.4.4 Morphing and Warping

Beside animation there are some special effects used in multimedia. Morphing and warping are two commonly used special effects. Morphing takes two images and seamlessly changes one images to another. The second image actually seems to grow out of the first one. Morphing can also be used to show the pace of changes more clearly than photos. Warping is variation of the morphing where one images is used to show various changes that take place. It uses the key points of one image to create different effects, instead of mixing up two images.

### 3.4.5 Digital Audio

Digitised audio is used as means for providing an interactive solution to the multimedia. The most common reason for using digital audio in computer is so as to be able to use multimedia in its full potential. The most common requirement is to be able to input sound such as a spoken commentary on an image or a document.

As all of you know, sound is a repeated pattern of pressure in the air and a microphone converts a sound wave into an electrical wave. The shape and frequency of the electrical wave is identical to the shape and wave of the sound wave and the clarity of what we hear is entirely depend on the shape and frequency of the sound wave. Sound can also be recorded and reproduced using digital signals and the errors can be reduce drastically in digital recording of the sound. As in case of the video, audio has to be converted into digital form to produce digital audio in order to use it in the multimedia. And the digital audio system will then reconvert the entire digitised audio into analog form, which can be heard on the speaker.

And this two way transformation of audio is known as analog-to-digital conversion. But the storage space required for digital audio is huge, something around more than 1MB for one minute of audio. The entire process of digitisation is a simple process of converting analog or electrical signal of audio to a computer data file in the digital format. The microphone (which is normally used for recording) convert the voice into electrical signal or analog audio signal. Then the analog signal is passed into the audio input of a digital card or sound card (discussed latter in this unit). Once inside, the signal is fed into the analog-digital converter, which convert the analog audio signal to digital form and store it as a computer data file. Playing back a digital audio data file is just the reverse process of converting analog to digital. Once the digital audio file is send to the sound card for playback the file is channel through a converter and only the digitised version of the original file is converted into analog and put into the speaker. As constant or frequent conversion can reduce the sound quality, it is always advisable to keep all audio in digital format while working on a multimedia program.

### 3.4.6 Digital Video

Digitised video is one of the many technologies used in the development of interactive multimedia. It is one of the way to play back and record video in multimedia program. It offers a wide range of flexibility as compared to standard video signal. Unlike regular video, quality of image would not degrade from copy to copy. As digital video is made up of a digital code and not an electrical analog signal, it contain the exact information as in the original. However the final output depends on how the video images converted to digital form during the development phase. Video come in from an external source either from TV or VCR or camera to the video digitiser card inside the system. Some systems use card which has dual function for both audio and video digitiser conversion. The process of converting analog video signal to digital format is called sampling. Using this process the converter card in the system converts or process the analog video signals into digital data streams so that these signals can be stored in the binary data structure format of 1s and Os. The size of the digital data file is then compressed to a considerable amount using some compression program. During this process the digitiser fuses the digital video into a digital movie format and save that all the section of the compressed movie format in the hard disk. Once the conversion and compression process is complete, the file can be played back on the computer screen. These digitised files can also be edited according to requirement using various video editing software.

## 3.5 SOFTWARE AND HARDWARE FOR MULTIMEDIA

By now it must have been clear to you that, sound, text animation and graphics are the integral part of a multimedia software. To produce these media elements, there are various software available in the market, such as Paint Brush, Photo Finish, Animator, Photo Shop, 3D Studio, Corel Draw, Sound Blaster, Master Blaster etc. Software are also available in the market to combined these independently created media elements, such as IMAGINET Apple Hyper Card, Script X etc.

### 3.5.1 Software for Multimedia

Among various types of software for creation of graphics, animation etc. Designer, Corel Draw, Picture Publisher, Photo Magic, Animator pro etc. are most commonly used programs.

Designer is one of the professional rated drawing and graphics package for windows which is also available in OS/2 system. It is specially meant for graphic artists, technical illustrators and has a wide range of drawing tools and powerful text handling features. Similarly Photo Publisher is a professional photo retouching or image editing package designed to enable to retouch and enhance photos faster. It has powerful masking and retouching tools and more than 30 special effects and filters. It has an image browser, precise scanner, printer calibration, a stitching tool and over 50 ready to use texture and image.

Similarly other multimedia related software also works on the same type of working environment, but they have their own features to experiment upon.

### 3.5.2 Multimedia Extension in Windows

All these software do not have a standard platform. To provide all PC the same standard to multimedia, Microsoft developed multimedia extension for Windows operating environment. This feature add several multimedia capabilities to the Windows operating system, including Resource Interchange File Format (RIFF), a standard format for multimedia data, including Bitmap graphics, animation, digital video and audio recording and playback facility for digital audio. It also include MIDI (Musical Instrument Digital Interface) files and MCI (Media Control Interface) to interface a work with external devices such as CD-ROM, Sound card, laser disc players etc. Most of the multimedia extension software for windows works in the background. The utilities like MIDI mapper, sound recorder and media player are the direct interface to the software sections within the multimedia extension for windows.

Most standard window graphics are limited to 256 colors if it is not interfaced with any external custom software and graphics card. All external devices outside the multimedia extension for windows are controlled by MCI for windows. The main interface to MCI is the media player which in turns control any external devices such as CD-ROM, Video Player, Laser Disc Player etc. Whereas sound recorder utility provides facility to playback and record digital audio as .WAV files, directly into the PC with the external sound device such as sound card. It has the function like start, record, stop, play etc. using transport control. Similarly MIDI mapper makes sure that events that are specified in the MIDI file are sent to the correct MIDI instruments.

The video for windows is an external set of software works along with multimedia extension for Windows. It has the feature for video digitised recording, playback and editing capabilities to the multimedia extension for Windows. The videocap utility of this software is used to capture the video and audio clips using external hardware. The captured sequence can be viewed in a number of different sizes and sped and also different color palate can be created from individual frame. Once captured, the entire sequence can be edited frame by frame according to the requirement. Video for windows has four different types of editing features named as VidEdit, PalEdit, WavEdit and BitEdit. As the name suggests VidEdit is used to cut and paste captured video segments together, WavEdit is the feature work with the recorded digital audio and helps you to edit it. Where as PalEdit is the work with the color palates within the captured video to improve the color, BitEdit helps clean up the rough patches in the images. It also has the interface to the media control panel to control digital video files.

### 3.5.3 Hardware for Multimedia

A typical multimedia package usually consists of CD-ROM player, sound card such as Sound Blaster, Master Blaster and some times a microphone and a range of multimedia software. And probably with either a bulletin MIDI interface on the card as an add on utility.

### 3.5.4 Sound Card

Sound output from a computer has been a feature of machine usually used for sending warning error message or games. If better quality and capabilities for sound output or inputs are required, then there must be a device which can be added to the basic PC machine. This device is known as Sound Card which is added to the basic PC machine by inserting it in free slot. The most common reason for adding a sound card to a PC is so as to be able to use multimedia fully by recording and playing back the digital audio. While selecting an add on sound card, quality of sound from the loudspeaker has to be taken into consideration and the software provided with a sound card should be adequate enough to suit your requirement for handling sound effects in multimedia.

Normally a machine of type 386 or 486 or latter version is used for multimedia, with ample RAM memory usually 4MB or more, adequate secondary storage space with high speed reading facilities. While installing a sound card one should keep in mind that the installation of sound card should not conflict with any other external devices installed in the computer system. Installation of sound card might require some default settings to be changed to make it compatible with the computer system.

Once all the setting are done correctly, place the card in a free slot and check all the connected cables to ensure that all connections are fitted properly. After this task is over, run the designated installation software to install the sound card i.e. to make necessary changes in the computer software settings.

The most common element of software is install program which install drivers to work with the sound card. In addition to adding drivers, the installation program modify the AUTOEXEC.BAT and CONFIG.SYS so as to active the sound card when the computer is switched on.

### 3.5.5 CD-ROM (Compact Disc Read Only Memory)

Multimedia uses digitised audio and video which not only takes huge amount of storage space, but they also required high speed storage media, which can send large amount of data back and forth quickly to keep audio and video playing and recording smoothly. Because of this requirement most of multimedia packages uses hard disk drive which offer high speed, high data throughout and plenty of storage space like 1GB or more. Sometimes, two or more hard disk drive are used as an array to work together so that they act like a single hard drive.

However using a disk array of two or more disk for multimedia storage, can no doubt boost the processing speed for the multimedia element, but these are not durable in nature. Magnetic optical disk media offer large amount of storage in a durable package. CD-ROM is now widely used for multimedia storage as it has an amazing amount of flexibility and potential as a multimedia storage and distribution medium. One CD-ROM can hold up to one and half hours of digital audio or around 700 to 750 MB of computer data for that matter.

The production of CD-ROM starts as a glass plate, whose surface contain no deformities that can be detected by the light beam. The glass plate is then coated with photo resists, a method which harden on exposure to light. The image is produce by treating the glass plate as if it were a compact disc. The CD-ROM recording uses the principle of write once and read many times. It uses the optical recording method, using a beam of light from a miniature semiconductor laser. Once the photo resists has been processed, the pattern of pits will develop and this comprises the glass matter. The surface of the disc is then slivered so that the pits are protected and a thick layer of nickel is then plated over the surface.

To install a CD-ROM drive, there must be a free drive bay and the computer system must also have a sound card installed or a spare slot into which either a sound card or a CD-ROM drive can be placed. Before installation all the jumper settings should be properly done according to the specification. Once the CD-ROM drive is fitted in the drive bay with proper jumper settings, necessary software can be installed to make the CD-ROM drive work.

As far as CD-ROM is concerned, the main feature of it is the huge storage space and very high immunity from damage, the control system that is built into the CD systems and it is easy to locate different pieces of data, so that comparatively simple software can be used to guide the laser beam to find what is required.

### 3.5.6 Laser Disc

As CD-ROM, a laser disc uses the same recording media can record and play back high quality digital audio files. As compared to the CD-ROM, a laser disc can play back four channels at a time (two digital and two analog channels) whereas CD-ROM can play only two channels at a time. But the biggest difference between them is that laser disc stores video in analog signals. The disadvantages of using a laser disc is you can not work with video as through it were a digital video file, because the analog output of a laser disc is just like any other video source.

CD-ROM stores digital data on only one side of the disc, whereas the laser disc stores an analog video signal on both sides of the disc.

## 3.6 SUMMARY

This unit introduce you to a new concept in information technology; Multimedia. With a number of examples it explains how slowly but steadily it has started to creep into our life in one or the other way. You were also explained various components of multimedia like graphics, sound, animation, digital audio and video and it talked about their characteristics and various storage formats they use. In a great length it introduces you to various storage media used for multimedia elements and their functions.

## 3.7 FURTHER READINGS

**Graphics and Drawing**

Corel Draw S/S by John Campbell & Marion Pye (Butterworth- Heinemann)

Computer-aided Design on a Shoestring by Ian Sinclair (BSP)

Illustrated AutoSketch 3 by Ian Sinclair (David Fulton Publishers)

**Sound and MIDI**

Practical MIDI Handbook by R.A. Penfold (PC Publishing)

Advanced MIDI Users Guide by R.A. Penfold (PC Publishing)

Introducing Digital Audio by Ian Sinclair (PC Publishing)

PC Music Handbook by Heywood and Evan (PC Publishing)

MIDI Survival Guide by Vic Lennard (PC Publishing)