Dhi-Qar University
College of education for pure science
Computer department
2nd class

System analysis and databases design lectures

Lecture SURA ISMAEL HAMAD

System Analysis and Design - Overview

Systems development is systematic process, which includes phases such as planning, analysis, design, deployment, and maintenance. Here, in this tutorial, we will primarily focus on:

1- Systems Analysis
2- Systems Design

Systems analysis

It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components.

System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose. Analysis specifies what the system should do.

Systems Design

It is a process of planning a new business system or replacing an existing system by defining its components or modules to satisfy the specific requirements. Before planning, you need to understand the old system thoroughly and determine how computers can best be used in order to operate efficiently.

System Design focuses on how to accomplish the objective of the system.

System Analysis and Design (SAD) mainly focuses on

- Systems
- Processes
- Technology

What is a System?

The word System is derived from Greek word Systema, which means an organized relationshipbetween any set of components to achieve some common cause or objective.

A system is "an orderly grouping of interdependent components linked together according to a plan to achieve a specific goal."

Constraints of a system

A system must have three basic constraints :-

1- A system must have some structure and behavior which is designed to achieve apredefined objective.
2- Interconnectivity and interdependence must exist among the system components.
3- The objectives of the organization have a higher priority than the objectives of itssubsystems.

For example, traffic management system, payroll system, automatic library system, humanresources information system.

Properties of a System

A system has the following properties : -

1- Organization

Organization implies structure and order. It is the arrangement of components that helps toachieve predetermined objectives.

2- Interaction

It is defined by the manner in which the components operate with each other.For example, in an organization, purchasing department must interact with production department and payroll with personnel department.

3- Interdependence

Interdependence means how the components of a system depend on one another. For proper functioning, the components are coordinated and linked together according to a specified plan. The output of one subsystem is the required by other subsystem as input.

4-Integration

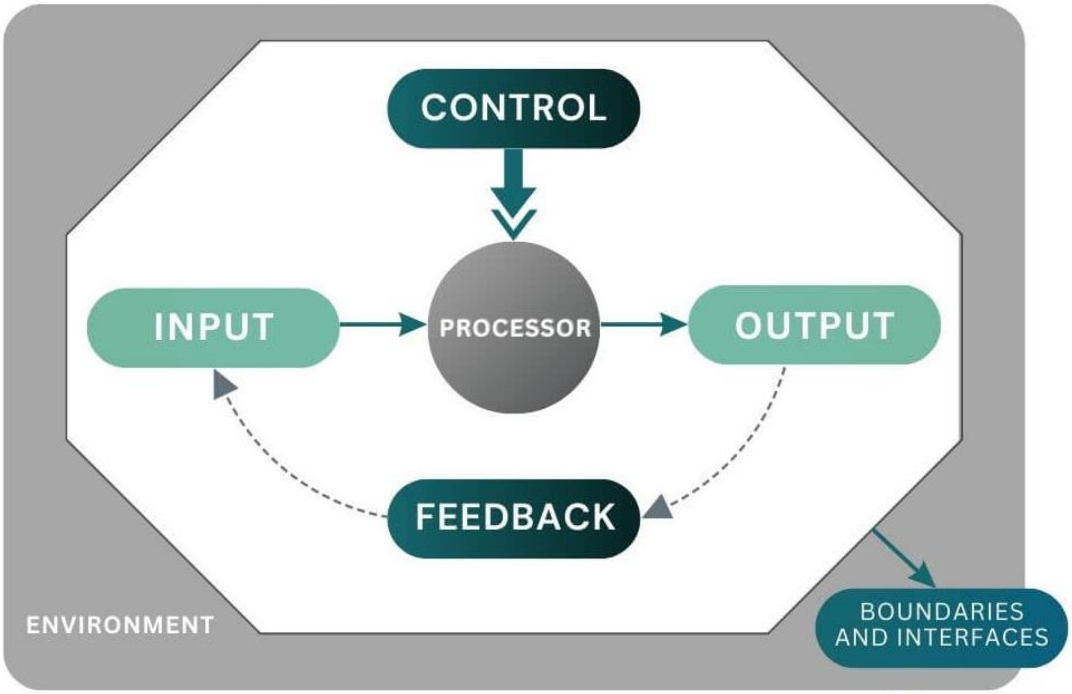
Integration is concerned with how a system components are connected together. It means that the parts of the system work together within the system even if each part performs a unique function.

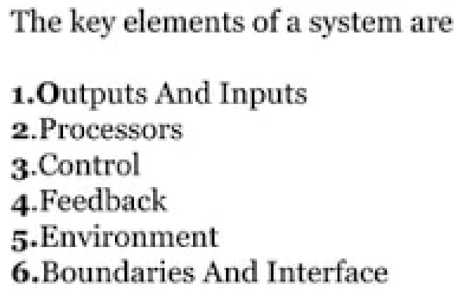
5- Central Objective

The objective of system must be central. It may be real or stated. It is not uncommon for a organization to state an objective and operate to achieve another. The users must know the main objective of a computer application early in the analysis for a successful design and conversion.

Elements of a System

The following diagram shows the elements of a system:





The key elements of a system are

1.Outputs And Inputs
2.Processors
3.Control
4.Feedback
5.Environment
6.Boundaries And Interface

1- Outputs and Inputs

The main aim of a system is to produce an output which is useful forits user. Inputs are theinformation that enters into the system for processing. Output is the outcome of processing.

2- Processor(s)

The processor is the element of a system that involves the actual transformation of input into output. It is the operational component of a system. Processors may modify the input either totally or partially, depending on the output specification. As the output specifications change, so does the processing. In some cases, input is also modified toenable the processor for handling the transformation.

3- Control

The control element guides the system. It is the decision–making subsystem that controls the pattern of activities governing input, processing, and output. The behavior of a computer System is controlled by the Operating System and software. In order to keep system in balance, what and how much input is needed is determined by Output Specifications.

4- Feedback

Feedback provides the control in a dynamic system.

Positive feedback is routine in nature that encourages the performance of the system.

Negative feedback is informational in nature that provides the controller with information foraction.

5- Environment

The environment is the “super system” within which an organization operates. It is the source of external elements that strike on the system. It determines how a system must function. For example, vendors and competitors of organization’s environment, may provide constraints that affect the actual performance of the business.

6- Boundaries and Interface

A system should be defined by its boundaries. Boundaries are thelimits that identify its components, processes, and interrelationshipwhen it interfaces with another system. Each system has boundaries that determine its sphere of influence and control. The knowledge of the boundaries of a given system is crucial in determining the nature ofits interface with other systemsfor successful design.

Types of Systems

The systems can be divided into the following types :-

1-Physical or Abstract Systems

Physical systems are tangible entities. We can touch and feel them. Physical System may be static or dynamic in nature. For example, desks and chairs are the physical parts of computer center which are static. A programmed computer is a dynamic system inwhich programs, data, and applications can change according to the user's needs. Abstract systems are non-physical entities or conceptual that may be formulas, representation or model of a real system

2- Open or Closed Systems

An open system must interact with its environment. It receives inputs from and delivers outputs to the outside of the system. For example, an information system which must adapt to the changing environmental conditions.

A closed system does not interact with its environment. It is isolated from environmental influences. A completely closed system is rare in reality.

3- Adaptive and Non Adaptive System

Adaptive System responds to the change in the environment in a way to improve theirperformance and to survive. For example, human beings, animals.

Non Adaptive System is the system which does not respond to the environment. Forexample, machines.

4 - Permanent or Temporary System

Permanent System persists for long time. For example, business policies. Temporary System is made for specified time and after that they are demolished. (set up for a program and it is dissembled after the program).

5 - Natural and Manufactured System

Natural systems are created by the nature. For example, Solar system, seasonal system. Manufactured System is the man-made system. For example, Rockets, dams, trains.

6 - Deterministic or Probabilistic System

Deterministic system operates in a predictable manner and the interaction between system components is known with certainty. For example, two molecules of hydrogenand one molecule of oxygen makes water.

Probabilistic System shows uncertain behavior. The exact output is not known. For example, Weather forecasting, mail delivery.

7 - Social, Human-Machine

Machine System Social System is made up of people. For example, social clubs, societies. In Human-Machine System, both human and machines are involved to perform a particular task. For example, Computer programming. Machine System is where human interference is neglected.

All the tasks are performed by the machine. For example, an autonomous robot.

8 - Man–Made Information Systems

It is an interconnected set of information resources to manage data for particular organization, under Direct Management Control (DMC). This system includes hardware, software, communication, data, and application for producing information according to the need of an organization.

Man-made information systems are divided into three types − Formal Information System − It is based on the flow of information in the form of memos, instructions, etc.,from top level to lower levels of management.

Informal Information System − This is employee based system which solves the day to

day work related problems.

9 - Computer Based System

This system is directly dependent on the computer for managing business applications. For example, automatic library system, railway reservation system, banking system, etc.

Systems Models

1- Schematic Models

A schematic model is a 2-D chart that shows system elements and their linkages. Different arrows are used to show information flow, material flow, and information feedback.

2- Flow System Models

A flow system model shows the orderly flow of the material, energy, and informationthat hold the system together. Program Evaluation and Review Technique (PERT), for example, is used to abstract a real world system in model form.

3- Static System Models

They represent one pair of relationships such as activity–time or cost–quantity. The Gantt chart, for example, gives a static picture of an activity-time relationship.

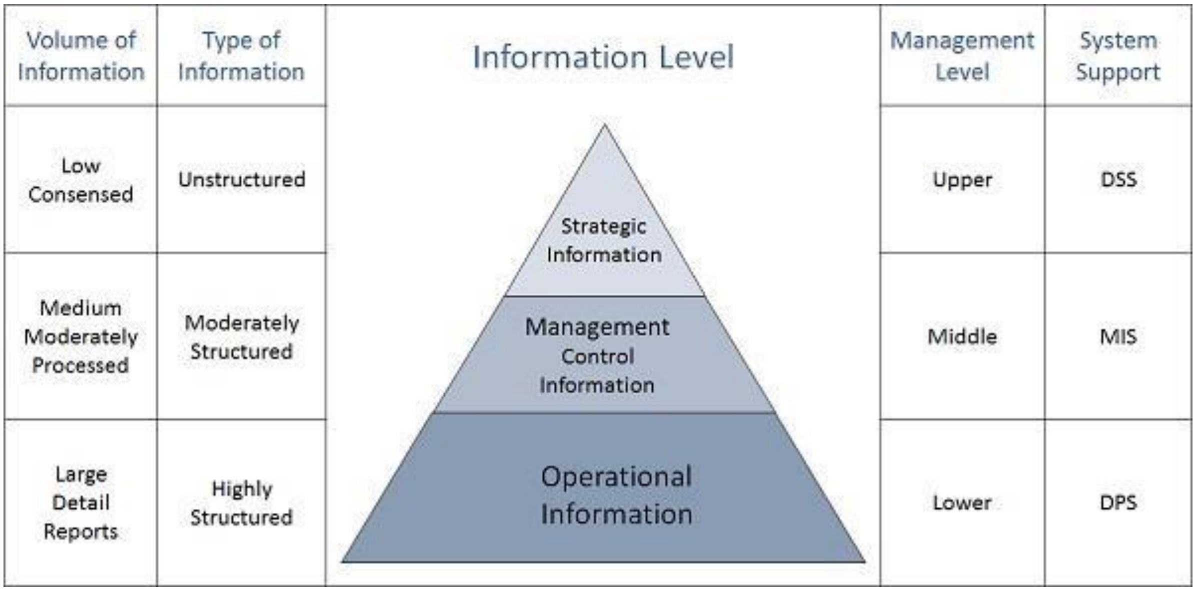
4- Dynamic System Models

Business organizations are dynamic systems. A dynamic model approximates the typeof organization or application that analysts deal with. It shows an ongoing, constantly changing status of the system.

It consists of Inputs that enter the system

- The processor through which transformation takes place
- The program(s) required for processing
- The output(s) that result from processing

Categories of Information



Volume of Information	Type of Information	Information Level	Management Level	System Support
Low Consensed	Unstructured	Strategic Information	Upper	DSS
Medium Moderately Processed	Moderately Structured	Management Control Information	Middle	MIS
Large Detail Reports	Highly Structured	Operational Information	Lower	DPS

There are three categories of information related to managerial levels and thedecision managers make.

1- Strategic Information:

This information is required by topmost management for long range planning policies for next few years. For example, trends in revenues, financial investment, and human resources, and population growth. This type of information is achieved with the aid of Decision Support System (DSS).

2- Managerial Information :

This type of Information is required by middle management for short and intermediate range planning which is in terms of months. For example, sales analysis, cash flow projection, and annual financial statements. It is achieved with the aid of Management Information Systems (MIS).

3- Operational information:

This type of information is required by low management for daily and short term planning to enforce day-to-day operational activities. For example, keeping employee attendance records, overdue purchase orders, and current stocks available. It is achieved with the aid of Data Processing Systems (DPS).

Structure Analysis :

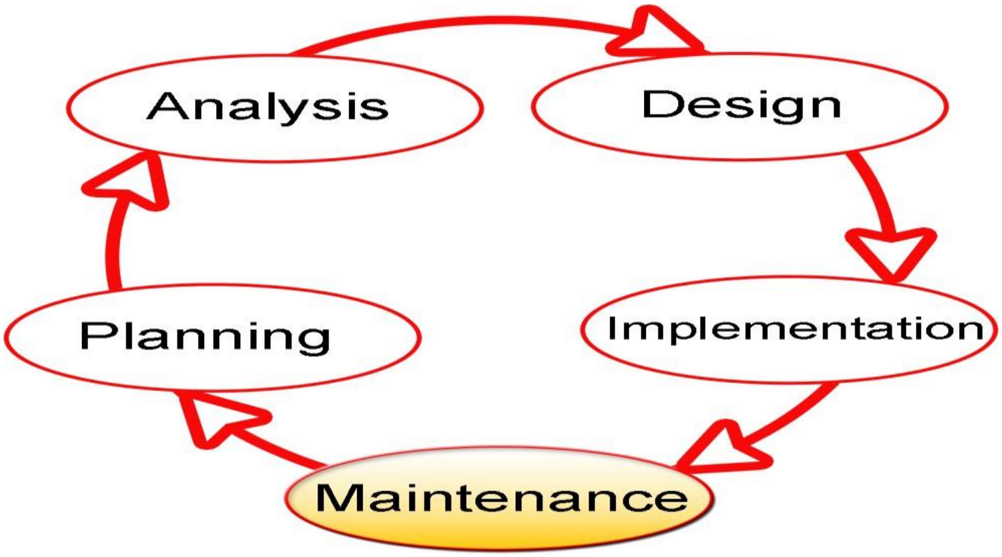
The life cycle of the SDLC system

The term "**System Development Life Cycle**" or **SDLC**' is an standard industry term used in systems engineering, information systems and software engineering to describe a process for planning, creating, testing, and deploying an information system.

Each of the phases has a series of events or steps that are completed in order to arrive at the final product. It comprises of 5 phases:

SDLC Phases:

- Planning
- Designing
- Developing
- Testing
- Maintenance



1- Planning and analysis

This phase is the most fundamental in the SDLC process. Business requirements are compiled and analyzed by a business analyst, domain expert, and project manager. The business analyst interacts with stakeholders to develop the business requirements document. They also write use cases and share this information with the project team. The aim of the requirements analysis is for quality assurance, technical feasibility, and to identify potential risks to address in order for the software to succeed.

2- Designing the product architecture

During the design phase, lead developers and technical architects create the initial high-level design plan for the software and system. This includes the delivery of requirements used to create the Design Document Specification (DDS). This document details database tables to be added, new transactions to be defined, security processes, as well as hardware and system requirements.

3- Developing and coding

In this phase, the database admin creates and imports the necessary data into the database. Programming languages are defined by requirements. Developers create the interface as per the coding guidelines and conduct unit testing. This is an important phase for developers. They need to be open-minded and flexible if any changes are introduced by the business analyst.

4- Testing

Testers test the software against the requirements to make sure that the software is solving the needs addressed and outlined during the planning phase. All tests are conducted as functional testing, including unit testing, integration testing, system testing, acceptance testing, and non-functional testing.

5- Maintenance

In a post-production, live software environment, the system is in maintenance mode. No matter the number of users, the sophistication of the software and testing, issues will occur. That's the nature of software with managing data, integration, and security, and real world usage.

Access to knowledgeable, reliable support resources is essential, as is routine maintenance and staying up to date on upgrades.

Role of System Analyst

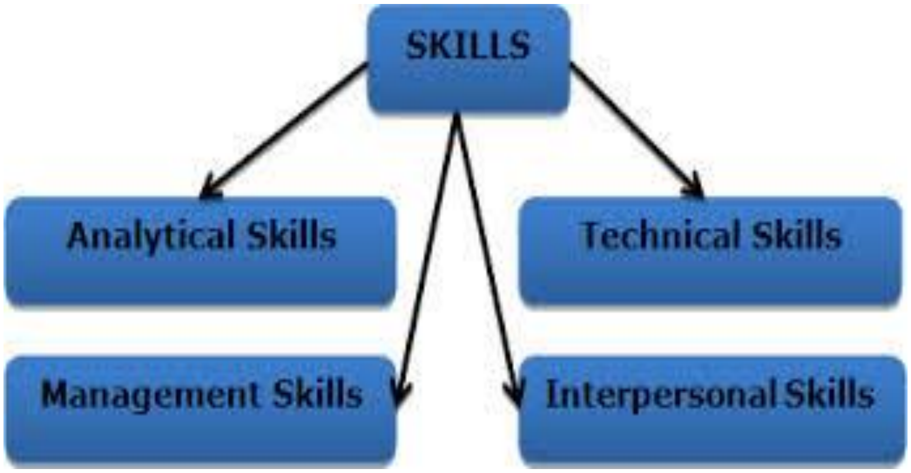
The system analyst is a person who is thoroughly aware of the system and guides the system development project by giving proper directions. He is an expert having technical and interpersonal skills to carry out development tasks required at each phase. He pursues to match the objectives of information system with the organization goal.

Main Roles

1. Defining and understanding the requirement of user through various Fact findingtechniques.
2. Prioritizing the requirements by obtaining user consensus.
3. Gathering the facts or information and acquires the opinions of users.
4. Maintains analysis and evaluation to arrive at appropriate system which is more userfriendly.
5. Suggests many flexible alternative solutions, pick the best solution, and quantify cost andbenefits.
6. Draw certain specifications which are easily understood by users and programmer inprecise and detailed form.
7. Plan the periodicity for evaluation after it has been used for some time, and modify thesystem as needed.

Attributes of a Systems Analyst

The following figure shows the attributes a systems analyst should possess :



Interpersonal Skills

1. Interface with users and programmer,
2. Facilitate groups and lead smaller teams,
3. Managing expectations,
4. Good understanding, communication, selling and teaching abilities.
5. Motivator having the confidence to solve queries.

Analytical Skills

1. System study and organizational knowledge
2. Problem identification, problem analysis, and problem solving
3. Sound commonsense
4. Ability to access trade
5. off - Curiosity to learn about new organization

Management Skills

1. Understand users jargon and practices.
2. Resource & project management.
3. Change & risk management.
4. Understand the management functions thoroughly.

Technical Skills

1 - Knowledge of computers and software.

2 - Keep abreast of modern development.

3 - Know of system design tools.

4 - Breadth knowledge about new technologies.

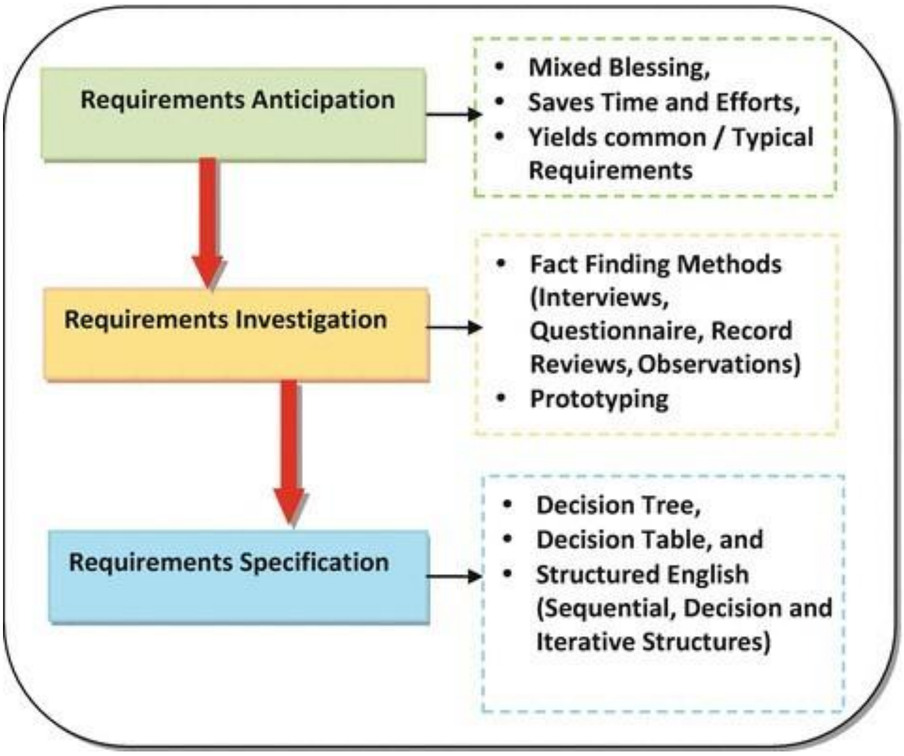
System Planning

What is Requirements Determination?

A requirement is a vital feature of a new system which may include processing or capturing of data, controlling the activities of business, producing information and supporting the management. Requirements determination involves studying the existing system and gathering details to find out what are the requirements, how it works, and where improvements should be made.

Major Activities in requirement Determination :

1. Requirements Anticipation.
2. Requirements Investigation.
3. Requirements Specifications.


Requirements Anticipation
Mixed Blessing,
Saves Time and Efforts,
Yields common / Typical Requirements
Requirements Investigation
Fact Finding Methods (Interviews, Questionnaire, Record Reviews, Observations)
Prototyping
Requirements Specification
Decision Tree,
Decision Table, and
Structured English (Sequential, Decision and Iterative Structures)

Data Modeling Using the Entity-Relationship (ER) Model

Example COMPANY Database

- We need to create a database schema design based on the following (simplified) **requirements** of the COMPANY Database:
  - The company is organized into DEPARTMENTs. Each department has a name, number and an employee who *manages* the department. We keep track of the start date of the department manager. A department may haveseveral locations.
  - Each department *controls* a number of PROJECTs. Each project has a unique name, unique number and is located at a single location.
  - We store each EMPLOYEE’s social security number, address, salary, sex, and birthdate.
    - Each employee *works for* one department but may *work on* severalprojects.
    - We keep track of the number of hours per week that an employeecurrently works on each project.
    - We also keep track of the *direct supervisor* of each employee.
  - Each employee may *have* a number of DEPENDENTs.
    - For each dependent, we keep track of their name,
    - sex, birthdate, and relationship to the employee.

ER Model Concepts

ER model has three main concepts: Entities (and their entity types and entity sets), Attributes (simple, composite, multivalued) and Relationships(and their relationship types and relationship sets)

- Entities and Attributes
  - Entities are specific objects or things in the mini-world that are representedin the database.
  - For example the EMPLOYEE John Smith, the Research DEPARTMENT, the ProductX PROJECT
  - Attributes are properties used to describe an entity.

    - For example an EMPLOYEE entity may have the attributes Name, SSN, Address, Sex, BirthDate

  - A specific entity will have a value for each of its attributes.
    - For example a specific employee entity may have Name='John Smith', SSN='123456789', Address ='731, Fondren, Houston, TX', Sex='M', BirthDate='09-JAN-55'

  - Each attribute has a *value set* (or data type) associated with it – e.g.integer, string, subrange, enumerated type, …

Types of Attributes (1)

- Simple

  - Each entity has a single atomic value for the attribute. For example, SSN or Sex.

- Composite

  - The attribute may be composed of several components. For example:

    Address(Apt#, House#, Street, City, State, ZipCode, Country), or

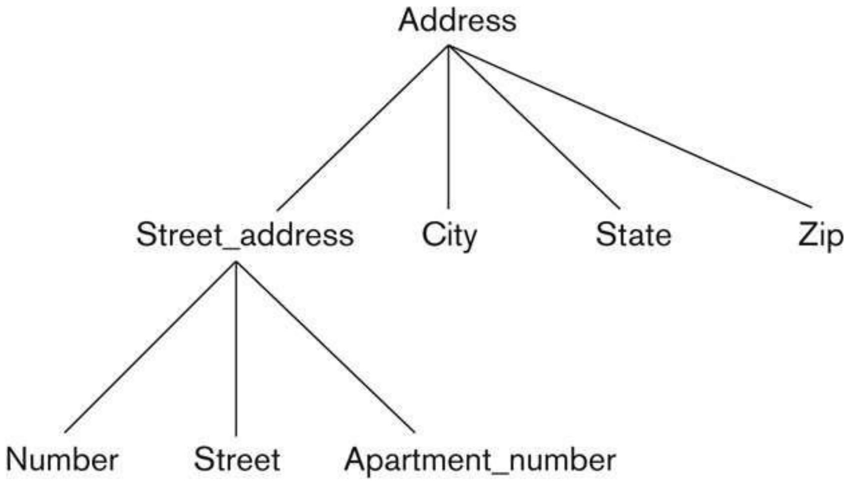
    Name(FirstName, MiddleName, LastName).

    Composition may form a hierarchy where some components are themselves composite.

- Multi-valued
  - An entity may have multiple values for that attribute. For example, Color of a CAR or PreviousDegrees of a STUDENT.

    Denoted as {Color} or {PreviousDegrees}.

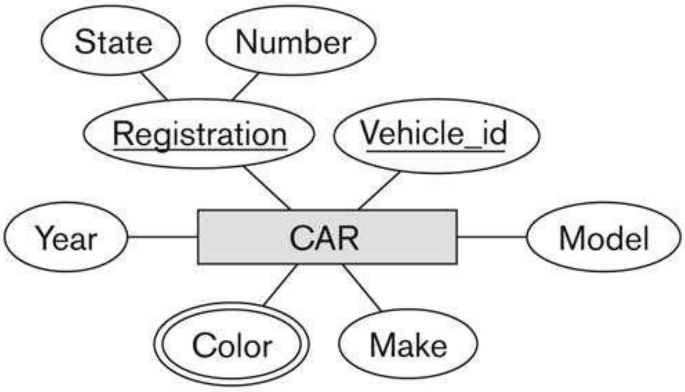
Example of a composite attribute



Displaying an Entity type

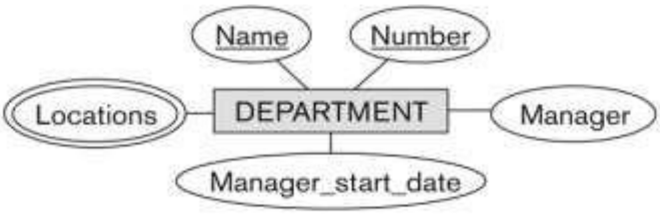
- In ER diagrams, an entity type is displayed in a rectangular box
- Attributes are displayed in ovals
- Each attribute is connected to its entity type
- Components of a composite attribute are connected to the oval representing thecomposite attribute
- Each key attribute is underlined
- Multivalued attributes displayed in double ovals

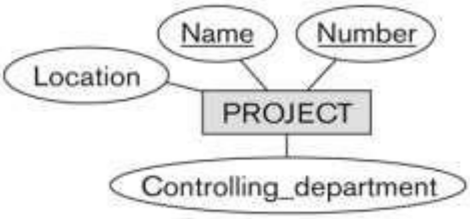
Entity Type CAR with two keys and a corresponding Entity Set

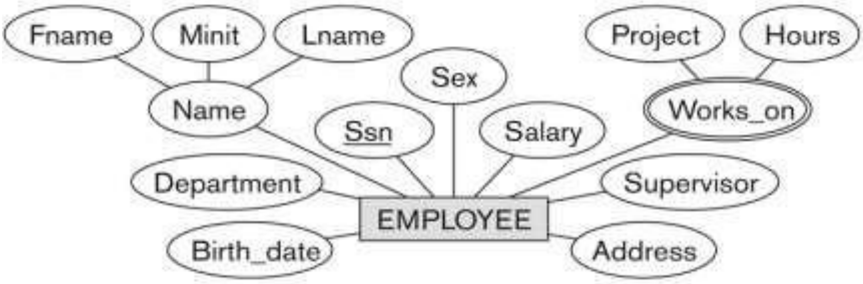


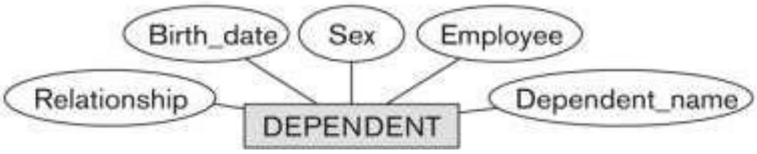
Initial Design of Entity Types for the COMPANY Database Schema

- Based on the requirements, we can identify four initial entity types in theCOMPANY database:
  - DEPARTMENT
  - PROJECT
  - EMPLOYEE
  - DEPENDENT

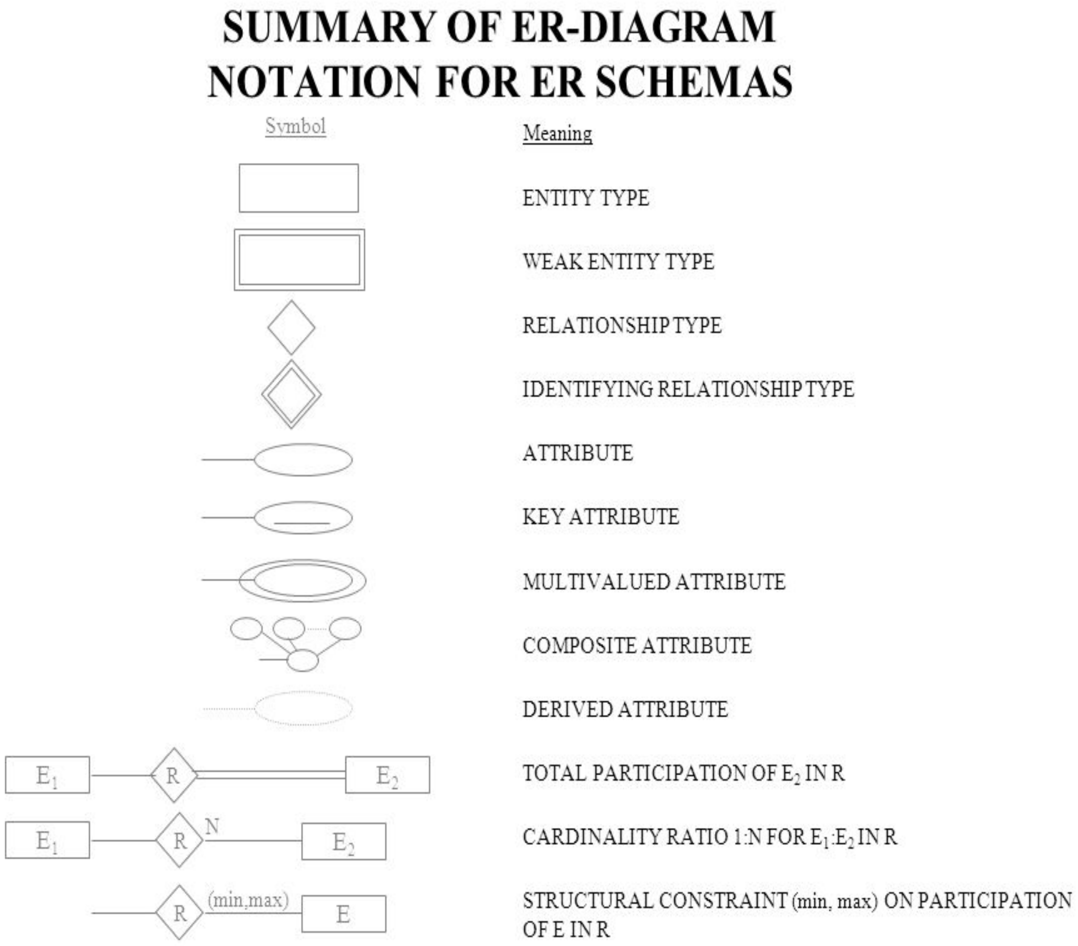

Name
Number
Locations
DEPARTMENT
Manager
Manager_start_date


Name
Number
Location
PROJECT
Controlling_department


Fname
Minit
Lname
Project
Hours
Sex
Name
Ssn
Salary
Works_on
Department
Supervisor
EMPLOYEE
Birth_date
Address


Birth_date
Sex
Employee
Relationship
Dependent_name
DEPENDENT

Summary of notation for ER diagrams



SUMMARY OF ER-DIAGRAM NOTATION FOR ER SCHEMAS

Symbol

Meaning

ENTITY TYPE

WEAK ENTITY TYPE

RELATIONSHIP TYPE

IDENTIFYING RELATIONSHIP TYPE

ATTRIBUTE

KEY ATTRIBUTE

MULTIVALUED ATTRIBUTE

COMPOSITE ATTRIBUTE

DERIVED ATTRIBUTE

TOTAL PARTICIPATION OF E_2 IN R

CARDINALITY RATIO 1:N FOR E_1:E_2 IN R

STRUCTURAL CONSTRAINT (min, max) ON PARTICIPATION OF E IN R

Refining the initial design by introducing relationships

- The initial design is typically not complete
- Some aspects in the requirements will be represented as relationships

Introduction to Database

Data: It's a facts material, and it can be a text, numbers orimages which are stored and processed by computer.

Example: Customer

1. cname.
2. cno.
3. ccity

Information: To understand these data, they need to translate or interpret to become information. Information is the meaningthat is given to the data through interpreted appropriately.

Database: is an organized collection of data.

It is a collection of interrelated data. These can be stored in theform of tables. A database can be of any size and varying complexity. A database may be generated and manipulated manually or it may be computerized.

Example: Customer database consists the fields as cname, cno,and ccity

Cname	Cno	Ccity

Types of Databases

1- Bibliographic Databases
2- Knowledge Databases
3- Graphic-Oriented Databases
4- Decision-making Databases

Database System: It is computerized system, whose overall purpose is to maintain the information and to make that theinformation is available on demand.

Advantages:

1.Redundency can be reduced.

2.Inconsistency can be avoided.

3.Data can be shared.

4.Standards can be enforced.

5.Security restrictions can be applied.

6.Integrity can be maintained.

7.Data gathering can be possible.

8.Requirements can be balanced.

Disadvantades of DBS :

1. Complexity

2. Size

3. Cost of DBMS

Database Management System (DBMS) :

A DBMS is essentially a collection of interrelated data and a setof programs to access this data. This collection of data is called the Database which facilitates storage,retrieval and managementof information.

Database Applications:

1- Banking: all transactions
2- Airlines: reservations, schedules
3- Universities: registration, grades
4- Sales: customers, products, purchases
5- Online retailers: order tracking, customized recommendations
6- Manufacturing: production, inventory, orders, supply chain
7- Human resources: employee records, salaries, tax deductions

Advantages of DBMS:

1.Data Independence.
2.Efficient Data Access.

3.Data Integrity and security.

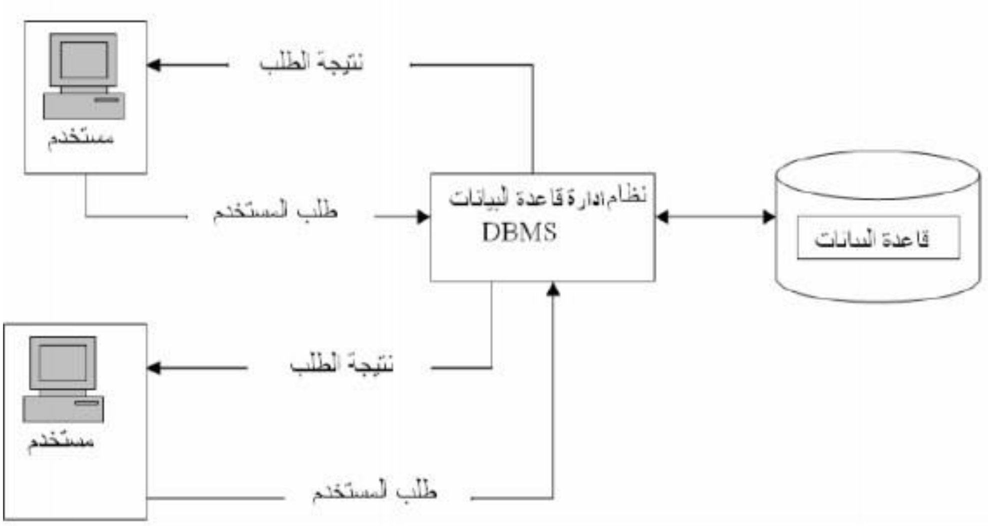
4.Data administration.

5.Concurrent access and Crash recovery.

6.Reduced Application Development Time.

- Examples:
- Oracle
- DB2 (IBM)
- MS SQL Server
- MS Access
- Ingres
- PostgreSQL

Database Management System (DBMS)



Database Models

The Database models in the Database Management System explains the logic behind the structure of a Database system that should usually include all the tables, which are represented as entities in ER model, the relationships between the tables and objects, and the requirement providedby the project team in order to settle on how data can be stored & accessed, granted the aimed Database System needs to be designed with respect to the rules and notions of the given data model the Database Architect prefers to be implemented.

Some of the Data Models in DBMS are:

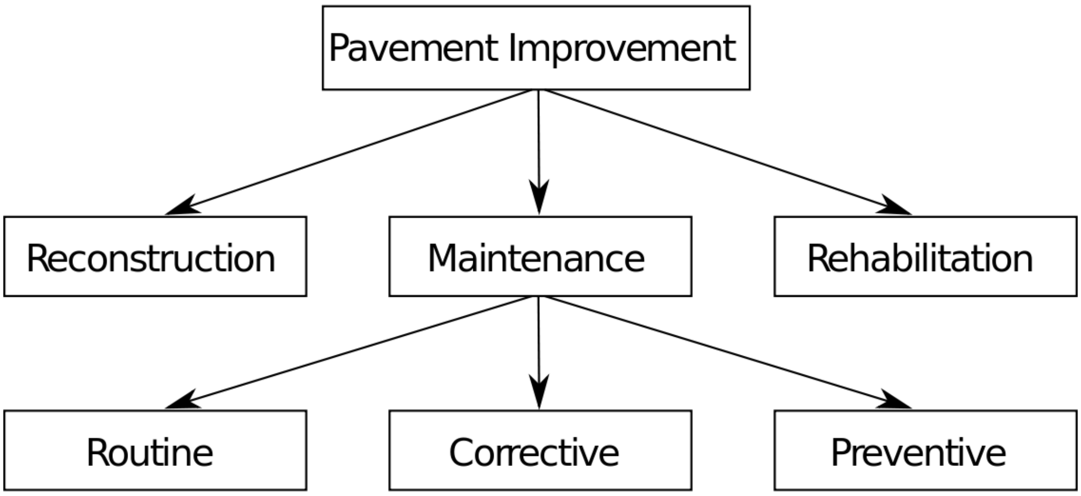
1. Hierarchical Model
2. Network Model
3. Entity-Relationship Model
4. Relational Model
5. Object-Oriented Data Model
6. Object-Relational Data Model
7. Flat Data Model
8. Semi-Structured Data Model
9. Associative Data Model
10. Context Data Model

1- Hierarchical Database Management Systems :

Hierarchical Model was the first DBMS model. This model organizes the data in the hierarchical tree structure. The hierarchy starts from the root which has root data and then it expands in the form of a tree adding child node to the parent node. This model easily represents some of the real-world relationships like food recipes, sitemap of a website etc.

Features of a Hierarchical Model

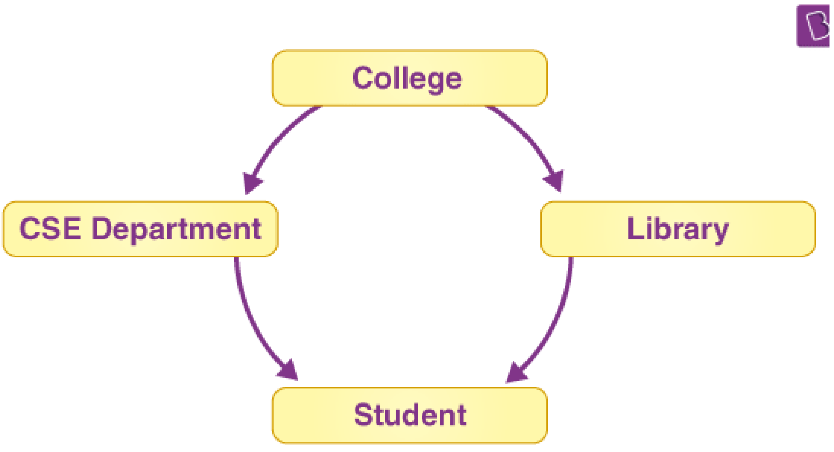
1. ***One-to-many relationship:***



2 - Network Database Management System:

This model is an extension of the hierarchical model. It was the most popular model before the relational model. This model is the same as the hierarchical model, the onlydifference is that a record can have more than one parent.It replaces the hierarchical tree with a graph.

Example: In the example below we can see that node student has two parents i.e. CSE Department and Library.This was earlier not possible in the hierarchical model.



Network Model

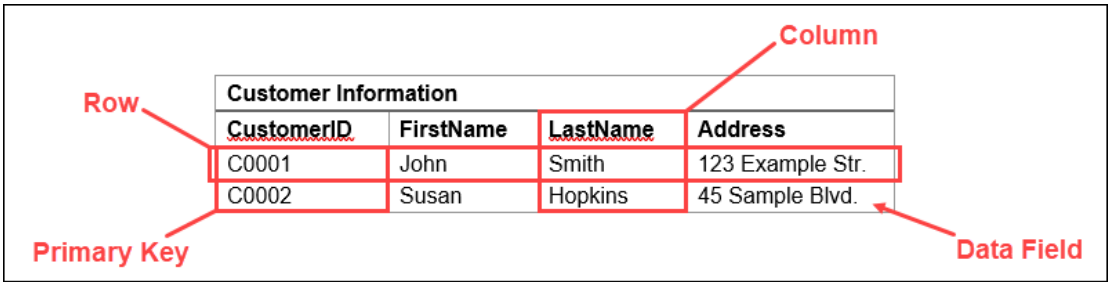
Features of a Network Model

many-to-many relationships.

3 -Relational Database Management System

Relational Model is the most widely used model. In this model, the data is maintained in the form of a two-dimensional table. All the information is stored in the form of row and columns. The basic structure of a relational model is tables. So, the tables are also called *relations* in the relational model

Example:



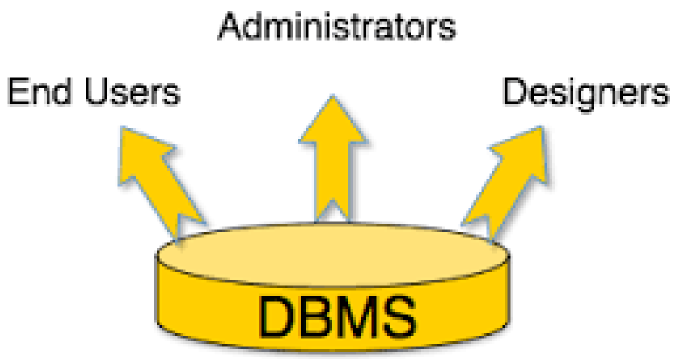
Customer Information			
CustomerID	FirstName	LastName	Address
C0001	John	Smith	123 Example Str.
C0002	Susan	Hopkins	45 Sample Blvd.

Column
Row
Primary Key
Data Field

Relational Database Components

- Tables
- Field (Row , Column)
- Record
- Primary Key
- Relations

Database Users



1- Administrator DB

Who manages databases, control the permissions, monitor thesystem and improve the performance of databases.

2- Designer DB

He is designing databases to be created and built with highlyefficient manner, according to user requirements.

3- User End DB

Some users have a sufficient experience to the preparation of the inquiries by query language, and others have no experiencetherefore a special programs are created for them Database Features.

Classification of Database Management System

- Data model
  - Relational
  - Object
  - Hierarchical and network (legacy)
  - Native XML DBMS
- Number of users
  - Single-user
  - Multi – user
- Number of sites
  - Centralized
  - Distributed
  - Homogeneous
  - Heterogeneous
- Cost
  - Open source
  - Different types of licensing
- Types of access path options
- General or special-purpose

DBMS Languages

- Data definition language (DDL)
  - Defines both schemas
- Storage definition language (SDL)
  - Specifies the internal schema
- View definition language (VDL)
  - Specifies user views/mappings to conceptualschema
- Data manipulation language (DML)
  - Allows retrieval, insertion, deletion, modification
- High-level or nonprocedural DML
  - Can be used on its own to specify complexdatabase operations concisely
  - Set-at-a-time or set-oriented
- Low-level or procedural DML
  - Must be embedded in a general-purpose programming language
  - Record-at-a-time

DBMS Interfaces

- Menu-based interfaces for Web clients orbrowsing
- Forms-based interfaces
- Graphical user interfaces
- Natural language interfaces
- Speech input and output
- Interfaces for parametric users
- Interfaces for the DBA

Joint Application Development (JAD)

It is a new technique developed by IBM which brings owners, users, analysts, designers, and builders to define and design the system using organized and intensive workshops. JAD trained analyst act as facilitator for workshop who has some specialized skills.

Advantages of JAD

- It saves time and cost by replacing months of traditional interviews and follow-up meetings.

- It is useful in organizational culture which supports joint problem solving.

- Fosters formal relationships among multiple levels of employees.

- It can lead to development of design creatively.

- It Allows rapid development and improves ownership of information system.

Secondary Reaseach or Background Reading

This method is widely used for information gathering by accessing the gleaned information. It includes any previously gathered information used by the marketer from any internal or external source.

Advantages

- It is more openly accessed with the availability of internet.
- It provides valuable information with low cost and time.
- It act as forerunner to primary research and aligns the focus of primary research.
- It is used by the researcher to conclude if the research is worth it as it is available with procedures used and issues in Collecting them.

Feasibility Study

Feasibility Study can be considered as preliminary investigation that helps the management to take decision about whether study of system should be feasible for development or not. It identifies the possibility of improving an existing system, developing a new system, and produce refined estimates for further development of system. It is used to obtain the outline of the problem and decide whether feasible or appropriate solution exists or not. The main objective of a feasibility study is to acquire problem scope instead of solving the problem. The output of a feasibility study is a formal system proposal act as decision document which includes the complete nature and scope of the proposed system.

Steps Involved in Feasibility Analysis

The following steps are to be followed while performing feasibility analysis:

- Form a project team and appoint a project leader.
- Develop system flowcharts.
- Identify the deficiencies of current system and set goals.

- Enumerate the alternative solution or potential candidate system to meet goals.

- Determine the feasibility of each alternative such as technical feasibility, operational feasibility, etc.

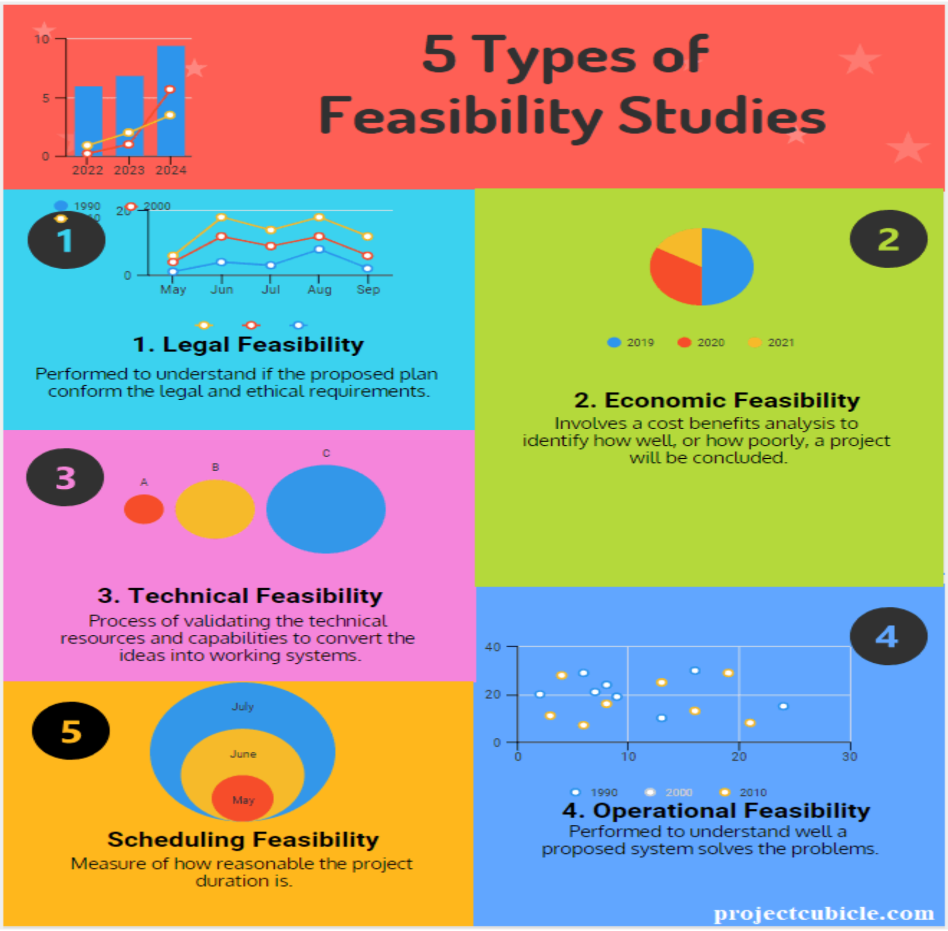
- Weight the performance and cost effectiveness of each candidate system.

- Rank the other alternatives and select the best candidate system.

- Prepare a system proposal of final project directive to management for approval.

Types of Feasibilities

1. Economic Feasibility
2. Organization Feasibility
3. Legal Feasibility
4. Behavioral Feasibility
5. Schedule Feasibility



5 Types of Feasibility Studies

1. Legal Feasibility
Performed to understand if the proposed plan conform the legal and ethical requirements.

2. Economic Feasibility
Involves a cost benefits analysis to identify how well, or how poorly, a project will be concluded.

3. Technical Feasibility
Process of validating the technical resources and capabilities to convert the ideas into working systems.

4. Operational Feasibility
Performed to understand well a proposed system solves the problems.

Scheduling Feasibility
Measure of how reasonable the project duration is.

projectcubicle.com

Analysts use various tools to understand and describe the information system. One of the ways is using structured analysis.

What is Structured Analysis?

Structured Analysis is:

- development method that allows the analyst to understand the system and its activities in a logical way.

- It is a systematic approach, which uses graphical tools that analyze and refine the objectives of an existing system and develop a new system specification which can be easily understandable by user.

- It is graphic which specifies the presentation of application.

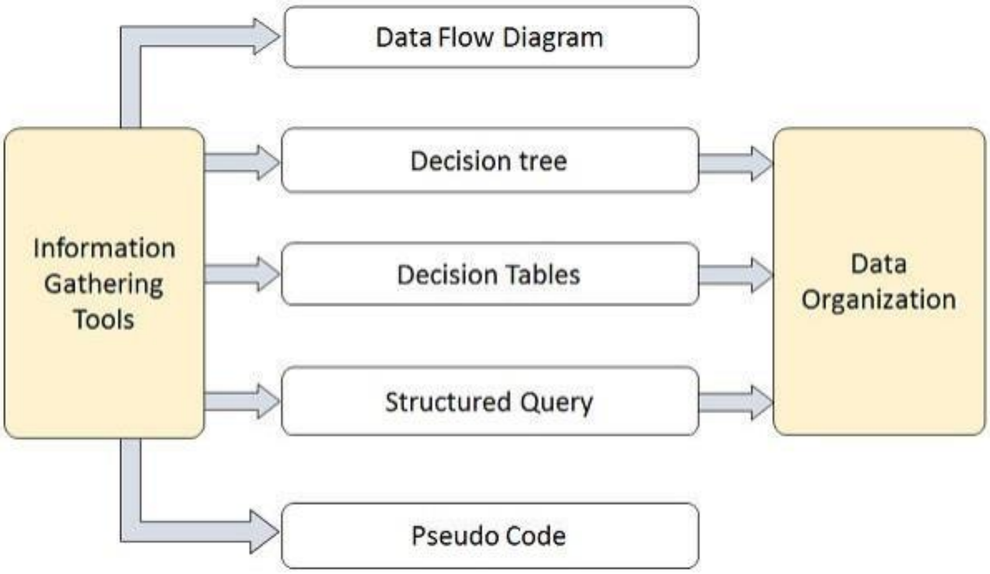
- It divides the processes so that it gives a clear picture of system flow.

- It is logical rather than physical.

Structured Analysis Tools

During Structured Analysis, various tools and techniques are used for system development. They are:

Data Flow Diagrams, Data Dictionary, Decision Trees, Decision Tables, Structured English, Pseudocode



Data Flow Diagrams (DFD) or Bubble Chart

1- It is a technique developed by Larry Constantine to express the requirements of system in a graphical form.

2 - It shows the flow of data between various functions of system and specifies how the current system is implemented.

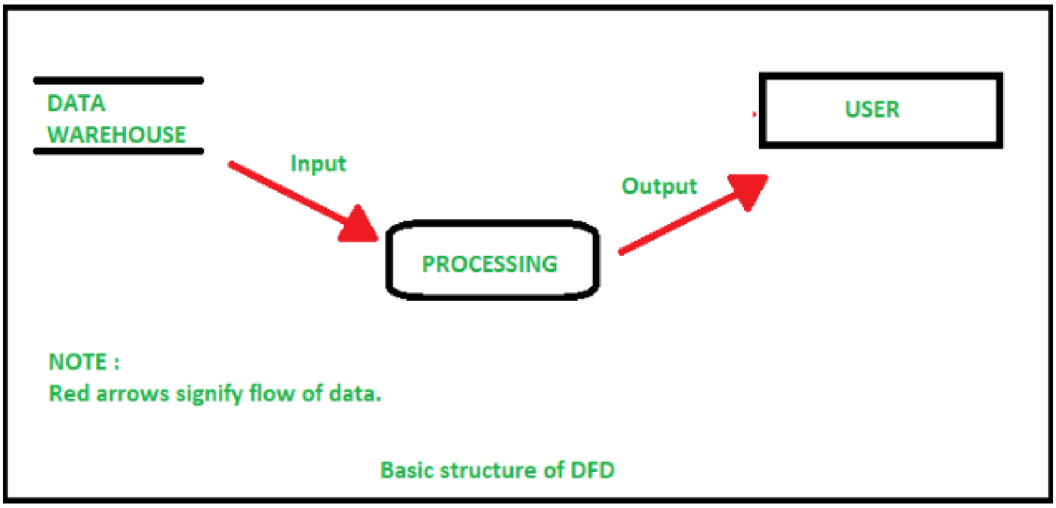
3- It is an initial stage of design phase that functionally divides the requirement specifications down to the lowest level of detail.

4- It is graphical nature makes it a good communication tool between user and analyst or analyst and system designer.

5 - It gives an overview of what data a system processes, what transformations are performed, what data are stored, what results are produced and where they flow.

Basic Elements of DFD

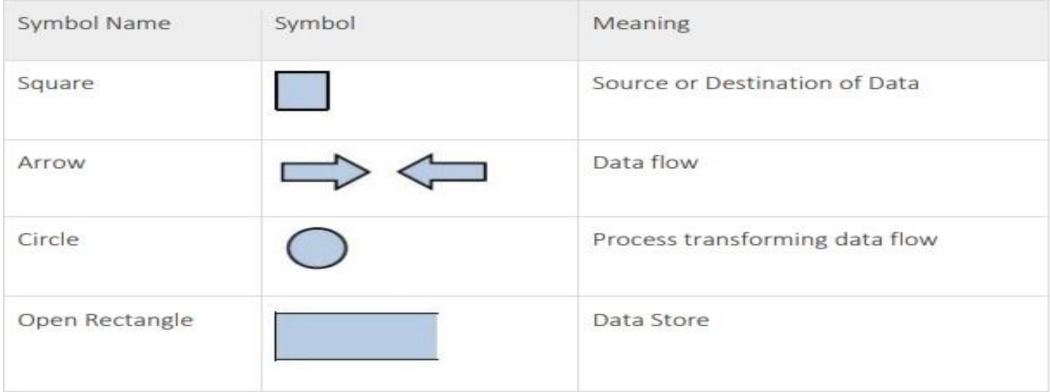
Is easy to understand and quite effective when the required design is not clear and the user wants a notational language for communication. However, it requires a large number of iterations for obtaining the most accurate and complete solution.



NOTE :
Red arrows signify flow of data.

Basic structure of DFD

The following table shows the symbols used in designing a DFD and their significance –



Symbol Name	Symbol	Meaning
Square	□	Source or Destination of Data
Arrow	⇨ ⇦	Data flow
Circle	○	Process transforming data flow
Open Rectangle	▭	Data Store

Types of DFD DFDs are of two types:

1-Physical DFD.
2-Logical DFD.

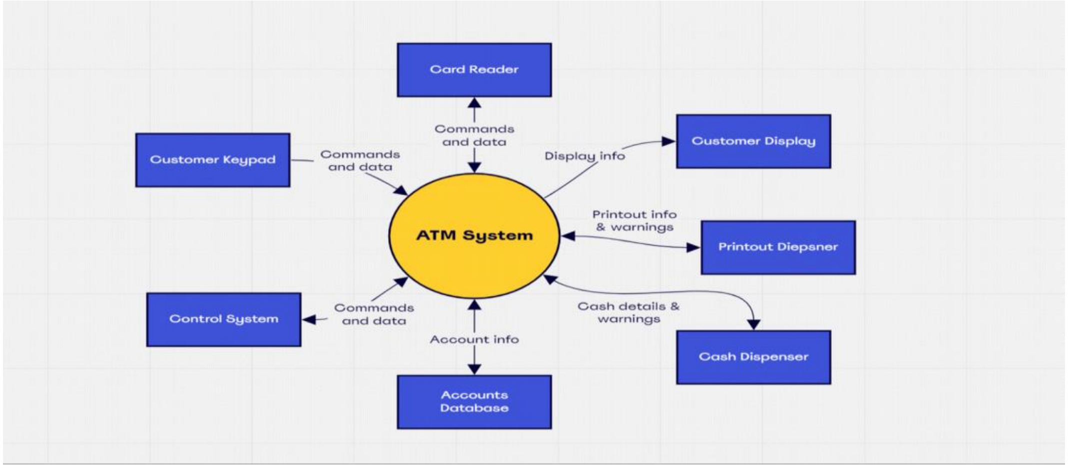
Levels of DFD

0-level DFD
1-level DFD
2- level DFD
3-level DFD

Context Diagram

A context diagram helps in understanding the entire system by one DFD which gives the overview of a system. It starts with mentioning major processes with little details and then goes onto giving more details of the processes with the top- down approach.

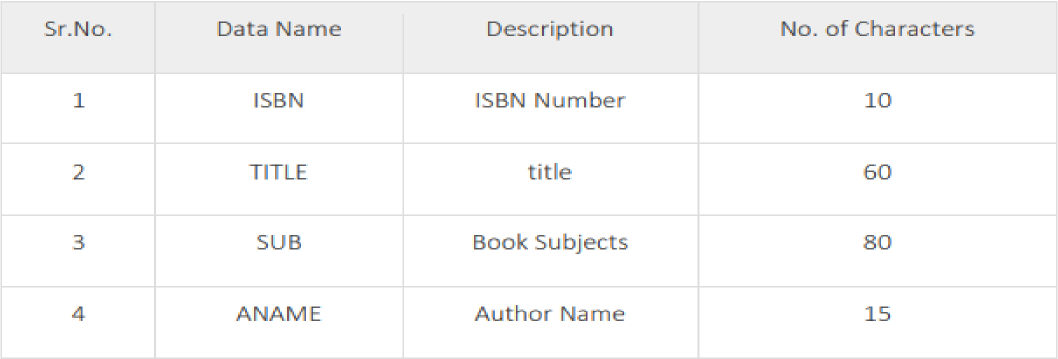
The context diagram of mess management is shown below.



Data Dictionary

A data dictionary is a structured repository of data elements in the system. It stores the descriptions of all DFD data elements that is, details and definitions of data flows, data stores, data stored in data stores, and the processes. A data dictionary improves the communication between the analyst and the user. It plays an important role in building a database. Most DBMSs have a data dictionary as a standard feature.

For example, refer the following table

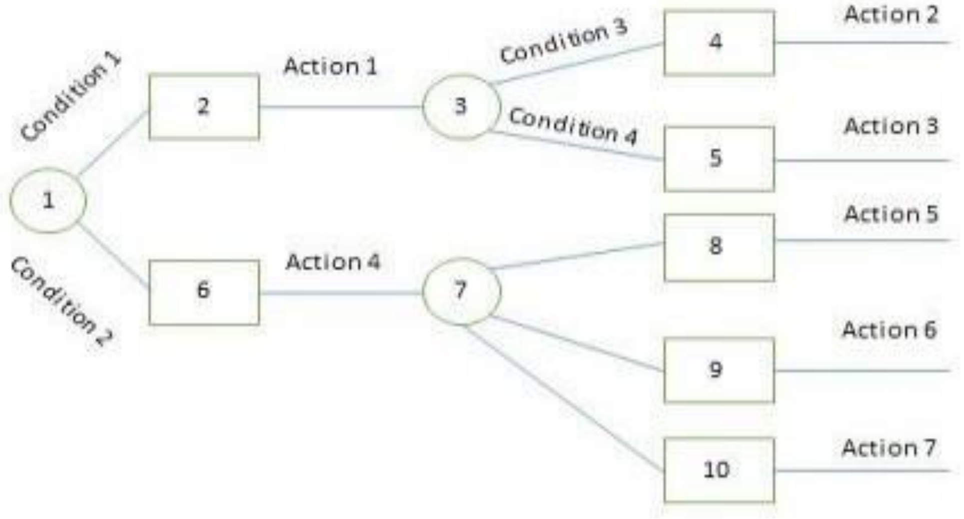


Sr.No.	Data Name	Description	No. of Characters
1	ISBN	ISBN Number	10
2	TITLE	title	60
3	SUB	Book Subjects	80
4	ANAME	Author Name	15

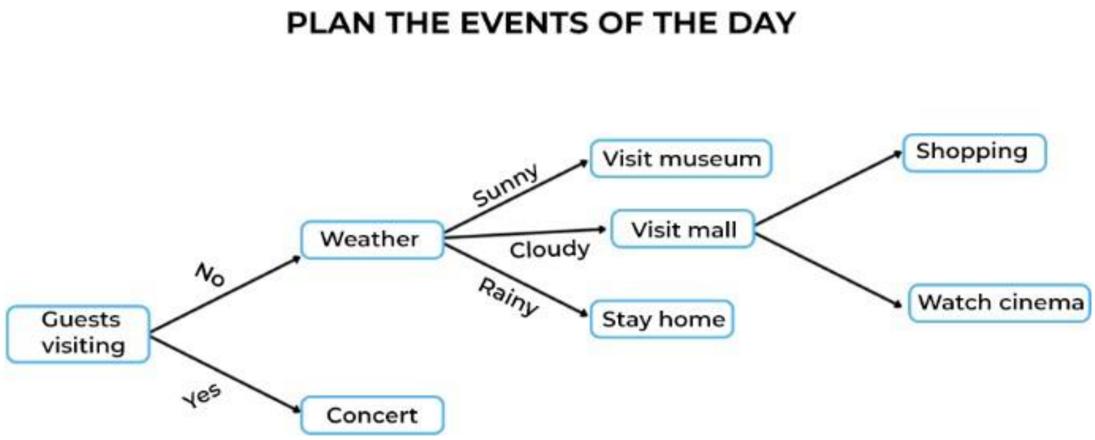
Decision Trees

Decision trees are a method for defining complex relationships by describing decisions and avoiding the problems in communication. A decision tree is a diagram that shows alternative actions and conditions within horizontal tree framework. Thus, it depicts which conditions to consider first, second, and so on.

Decision trees depict the relationship of each condition and their permissible actions. A square node indicates an action and a circle indicates a condition. It forces analysts to consider the sequence of decisions and identifies the actual decision that must be made. The major limitation of a decision tree is that it lacks information in its format to describe what other combinations of conditions you can take for testing. It is a single representation of the relationships between conditions and actions.



For example, refer the following decision tree



Decision Tables

Decision tables are a method of describing the complex logical relationship in a precise manner which is easily understandable.

1- It is useful in situations where the resulting actions depend on the occurrence of one or several combinations of independent conditions.

2- It is a matrix containing row or columns for defining a problemand the actions.

Components of a Decision Table:

- **Condition Stub −** It is in the upper left quadrant which lists all the conditionto be checked.
- **Action Stub −** It is in the lower left quadrant which outlines all

the action tobe carried out to meet such condition.

- **Condition Entryi −** It is in upper right quadrant which provides answers toquestions asked in condition stub quadrant.
- **Action Entryi −** It is in lower right quadrant which indicates the appropriateaction resulting from the answers to the conditions in the condition entry quadrant.

Conditions
Actions

Stubs	Entries

Condition Stub	Condition Entry
Action Stub	Action Entry

Structured English

Structure English is derived from structured programming languagewhich gives more understandable and precise description of process. It is based on procedural logic that uses construction and imperative sentences designed to perform operation for action.

- It is best used when sequences and loops in a program must be considered and the problem needs sequences of actions with decisions.

- It does not have strict syntax rule. It expresses all logic in terms of sequential decision structures and iterations.

For example, see the following sequence of actions –

```
if customer pays advance
then
Give 5%
Discountelse
if purchase amount >=10,000
then
if the customer is a regular customer
```

```
then Give 5% Discount
else No Discount
end if
else No Discount
end if
end if
```

Pseudocode

A pseudocode does not conform to any programming language andexpresses logic in plain English.

1- It may specify the physical programming logic without actualcoding during and after the physical design.

2- It is used in conjunction with structured programming.

3- It replaces the flowcharts of a program.

Guidelines for Selecting Appropriate Tools

Use the following guidelines for selecting the most appropriate tool that would suit your requirements –

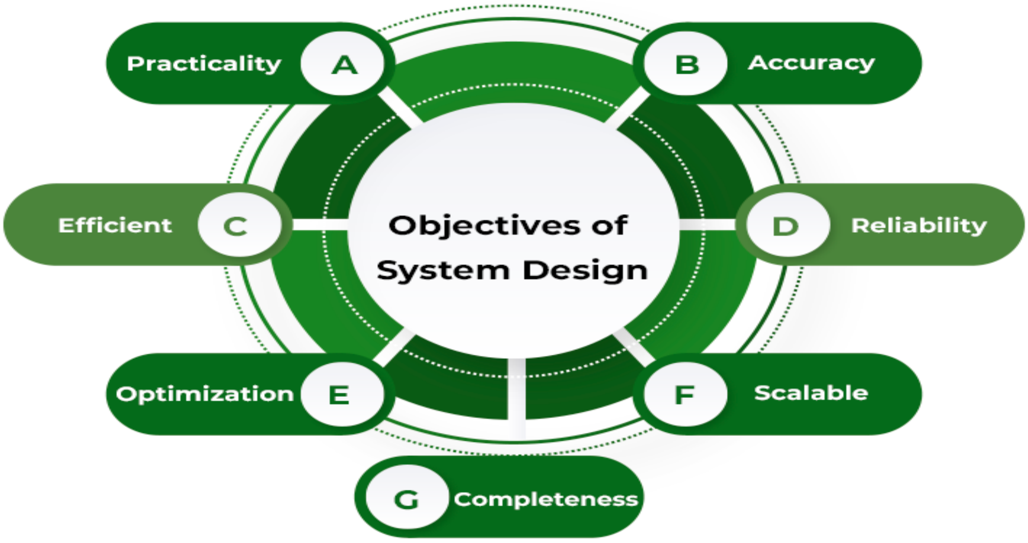
- Use DFD at high or low level analysis for providing good system documentations.
- Use data dictionary to simplify the structure for meeting the data requirement of the system.
- Use structured English if there are many loops and actions are complex.
- Use decision tables when there are a large number of conditions tocheck and logic is complex.
- Use decision trees when sequencing of conditions is important andif there are few conditions to be tested.

System Design

System design is the phase that bridges the gap between problem domain and the existing system in a manageable way. This phase focuses on the solution domain, i.e. *“how to implement?”*

It is the phase where the SRS document is converted into a format that can beimplemented and decides how the system will operate.

In this phase, the complex activity of system development is divided into several smaller sub-activities, which coordinate with each other to achieve the main objectiveof system development.



Inputs to System Design

System design takes the following inputs :-
1- Statement of work
2- Requirement determination plan
3- Current situation analysis
4-Proposed system requirements including a conceptual data model, modified DFDs,and Metadata (data about data).

Outputs for System Design

System design gives the following outputs :-
1-Infrastructure and organizational changes for the proposed system.
2-A data schema, often a relational schema.
3-Metadata to define the tables/files and columns/data-items.
4-A function hierarchy diagram or web page map that graphically describes theprogram structure.
5-Actual or pseudocode for each module in the program.
6-A prototype for the proposed system.

Types of System Design

1. Logical Design
2. Architectural Design

Documentation Control

Documentation is a process of recording the information for any reference or operational purpose. It helps users, managers, and IT staff, who require it. It is important that prepared document must be updated on regular basis to trace theprogress of the system easily.
After the implementation of system if the system is working improperly, then documentation helps the administrator to understand the flow of data in the systemto correct the flaws and get the system working.
Programmers or systems analysts usually create program and system documentation. Systems analysts usually are responsible for preparing documentation to help users learn the system. In large companies, a technical support team that includes technicalwriters might assist in the preparation of user documentation and training materials.

Types of Documentations

When it comes to System Design, there are following four main documentations –

1- Program documentation
2- System documentation
3- Operations documentation
4- User documentation

Program Documentation

It describes inputs, outputs, and processing logic for all the program modules. The program documentation process starts in the system analysis phase and continuesduring implementation.
This documentation guides programmers, who construct modules that are well supported by internal and external comments and descriptions that can be understood and maintained easily.

Operations Documentation

Operations documentation contains all the information needed for processing and distributing online and printed output. Operations documentation should be clear,concise, and available online if possible. It includes the following information :-

1- Program, systems analyst, programmer, and system identification.
2- Scheduling information for printed output, such as report, execution frequency,and deadlines.
3- Input files, their source, output files, and their destinations.4- E-mail and report distribution lists.
5- Special forms required, including online forms.
6- Error and informational messages to operators and restart procedures.
7- Special instructions, such as security requirements.

User Documentation

It includes instructions and information to the users who will interact with the system.For example, user manuals, help guides, and tutorials. User documentation is valuablein training users and for reference purpose. It must be clear, understandable, and readily accessible to users at all levels.

System Documentation

System documentation serves as the technical specifications for the IS and how theobjectives of the IS are accomplished. Users, managers and IS owners need never reference system documentation.

Testing

Testing is the process or activity that checks the functionality and correctness of software according to specified user requirements in order to improve the quality andreliability of system. It is an expensive, time consuming, and critical approach in system development which requires proper planning of overall testing process.

A successful test is one that finds the errors. It executes the program with explicit intention of finding error, i.e., making the program fail. It is a process of evaluating system with an intention of creating a strong system and mainly focuses on the weakareas of the system or software.

Characteristics of System Testing

System testing begins at the module level and proceeds towards the integration of the entire software system. Different testing techniques are used at different timeswhile testing the system. It is conducted by the developer for small projects and byindependent testing groups for large projects.

Stages of System Testing

The following stages are involved in testing –

A- Test Strategy

It is a statement that provides information about the various levels, methods, tools,and techniques used for testing the system. It should satisfy all the needs of an organization.

B- Test Plan

It provides a plan for testing the system and verifies that the system under testing fulfils all the design and functional specifications. The test plan provides the following information:

1-Objectives of each test phase
2- Approaches and tools used for testing
3- Responsibilities and time required for each testing activity
4- Availability of tools, facilities, and test libraries
5- Procedures and standards required for planning and conducting the tests
6- Factors responsible for successful completion of testing process

Test Case Design

Test cases are used to uncover as many errors as possible in the system. A number oftest cases are identified for each module of the system to be tested.

Each test case will specify how the implementation of a particular requirement ordesign decision is to be tested and the criteria for the success of the test.

The test cases along with the test plan are documented as a part of a system specification document or in a separate document called test specification or testdescription.

Test Procedures

It consists of the steps that should be followed to execute each of the test cases. These procedures are specified in a separate document called test procedure specification. This document also specifies any special requirements and formats forreporting the result of testing.

Test Result Documentation

Test result file contains brief information about the total number of test cases executed, the number of errors, and nature of errors. These results are then assessedagainst criteria in the test specification to determine the overall outcome of the test.

Types of Testing

Testing can be of various types and different types of tests are conducted dependingon the kind of bugs one seeks to discover :-

1- Unit Testing

Also known as Program Testing, it is a type of testing where the analyst tests or focuses on each program or module independently. It is carried out with the intentionof executing each statement of the module at least once.

2- Integration Testing

In Integration Testing, the analyst tests multiple module working together. It is usedto find discrepancies between the system and its original objective, current specifications, and systems documentation.

Here the analysts are try to find areas where modules have been designed withdifferent specifications for data length, type, and data element name

3- **Functional Testing**

Function testing determines whether the system is functioning correctly according toits specifications and relevant standards documentation. Functional testing typically starts with the implementation of the system, which is very critical for the success of the system.

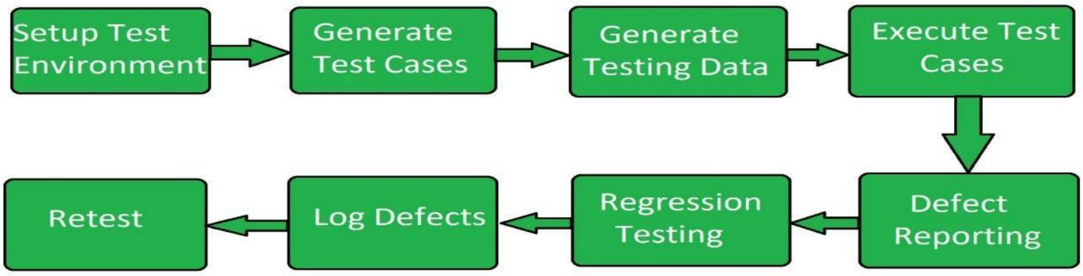
Functional testing is divided into **two categories**

A- Positive Functional Testing – It involves testing the system with valid inputs toverify that the outputs produced are correct.

B- Negative Functional Testing – It involves testing the software with invalid inputsand undesired operating conditions.

Rules for System Testing

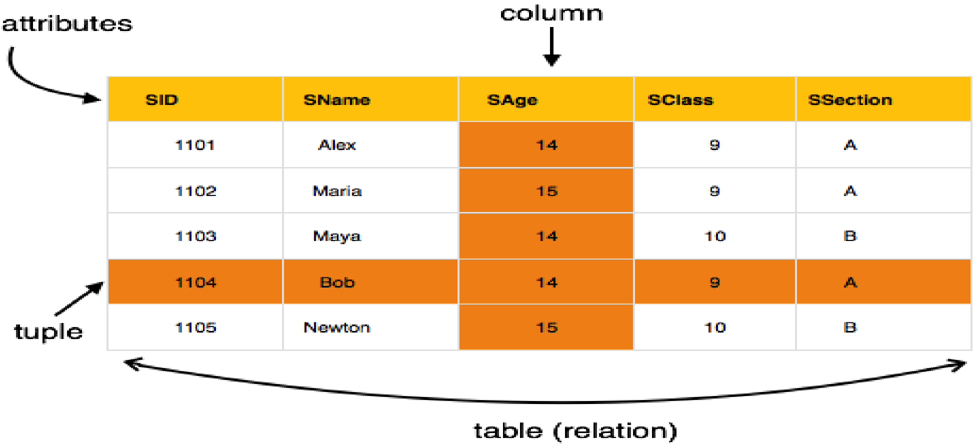
To carry out system testing successfully, you need to follow the given rules:



Relational Model

The main highlights of this model are –

1. Data is stored in tables called relations.
2. Relations can be normalized.
3. In normalized relations, values saved are atomic values.
4. Each row in a relation contains a unique value.
5. Each column in a relation contains values from a same domain.



SID	SName	SAge	SClass	SSection
1101	Alex	14	9	A
1102	Maria	15	9	A
1103	Maya	14	10	B
1104	Bob	14	9	A
1105	Newton	15	10	B

Normalization

Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy (repetition) and undesirable characteristics like Insertion, Update andDeletion anomalies.It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

Normalization usually involves dividing a database into two or more tables and defining relationships between the tables. The objective is to isolate data so that additions, deletions, and modifications of an attribute can be made in just one table and then propagated through the rest of the database via the defined relationships.

There are three main normal forms, each with increasing levels of normalization:

1. First Normal Form (1NF).
2. Second Normal Form (2NF).
3. Third Normal Form (3NF).

And other types Boyce Code Normal Form (BCNF),
fourth normal form (4NF) and
fifth normal form (5NF).

Normalization is used for mainly 3 purposes,

1- Eliminating redundant (useless) data.
2- Ensuring data dependencies make sense i.e data is logically stored.
3- Simplifying the process of insertion, deletion and updating DB.
Problems without Normalization

If a table is not properly normalized and have data redundancy then it will not only eat up extra memory space but will also make it difficult to handle and update the database, without facing data loss. Insertion, Updation and Deletion Anamolies arevery frequent if database is not normalized. To understand these anomalies let us take an example of a Student table.

roll no	Name	Branch	HOD	office_tel
401	A	CSE	Mr. X	53337
402	B	CSE	Mr. X	53337
403	C	CSE	Mr. X	53337
404	D	CSE	Mr. X	53337

In the table above, we have data of 4 Computer Sci. students. As we can see, datafor the fields branch, hod(Head of Department) and office_tel is repeated for the students who are in the same branch in the college, this is Data Redundancy.

Insertion Anomaly

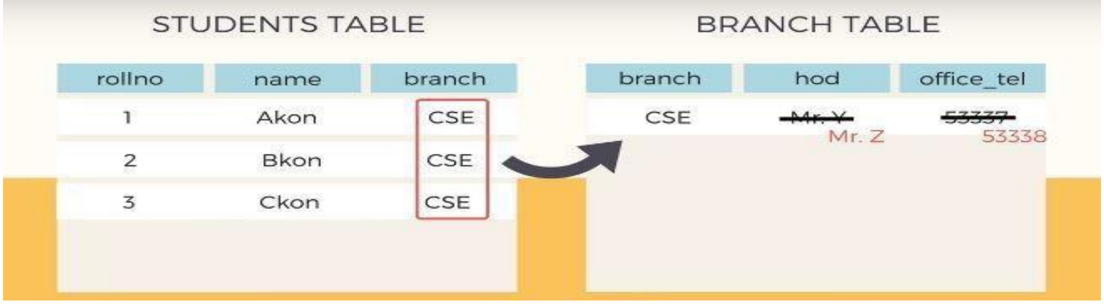
Suppose for a new admission, until and unless a student opts for a branch, data of the student cannot be inserted, or else we will have to set the branch information as NULL.Also, if we have to insert data of 100 students of same branch, then the branch information will be repeated for all those 100 students.

Updation Anomaly

What if Mr. X leaves the college? or is no longer the HOD of computer science department? In that case all the student records will have to be updated, and if bymistake we miss any record, it will lead to data inconsistency.

Deletion Anomaly

In our Student table, two different information are kept together, Student informationand Branch information. Hence, at the end of the academic year, if student records are deleted, we will also lose the branch information. This is Deletion anomaly.

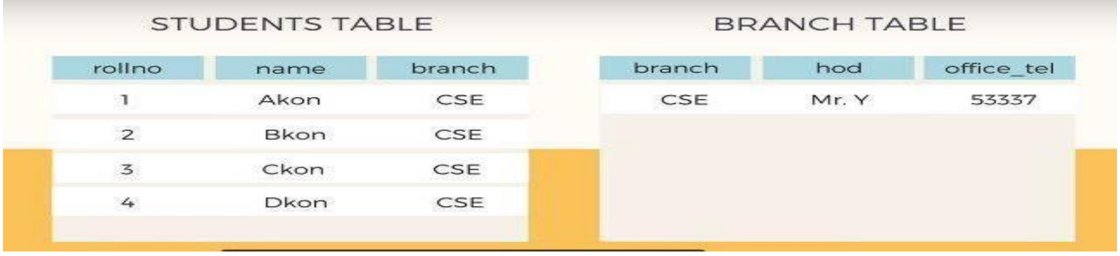


STUDENTS TABLE

rollno	name	branch
1	Akon	CSE
2	Bkon	CSE
3	Ckon	CSE

BRANCH TABLE

branch	hod	office_tel
CSE	~~Mr. Y~~ Mr. Z	~~53337~~ 53338

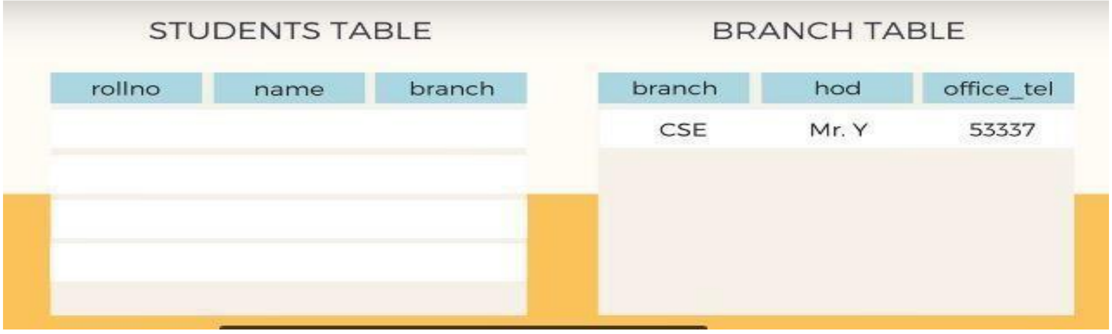


STUDENTS TABLE

rollno	name	branch
1	Akon	CSE
2	Bkon	CSE
3	Ckon	CSE
4	Dkon	CSE

BRANCH TABLE

branch	hod	office_tel
CSE	Mr. Y	53337



STUDENTS TABLE

rollno	name	branch

BRANCH TABLE

branch	hod	office_tel
CSE	Mr. Y	53337

DBMS SQL

Structured Query Language, popularly known as **SQL** is a domain specific language used to manage data in relational database management systems (RDBMS)

- It is a standard language for Relational Data base System. It enables a user to create, read, update and delete relational databases and tables.

- All the RDBMS like MySQL, Informix, Oracle, MS Access and SQL Server use SQL as their standard database language.

- SQL allows users to query the database in a number of ways, using English-like statements.

What is SQL Process?

SQL processing is **the parsing, optimization, row source generation, and execution of a SQL statement**. Depending on the statement, the database may omit some of these stages.

Important components included in this SQL process are:

- SQL Query Engine
- Optimization Engines
- Query Dispatcher

- Classic Query Engine

A classic query engine allows you to manage all the non-SQL queries

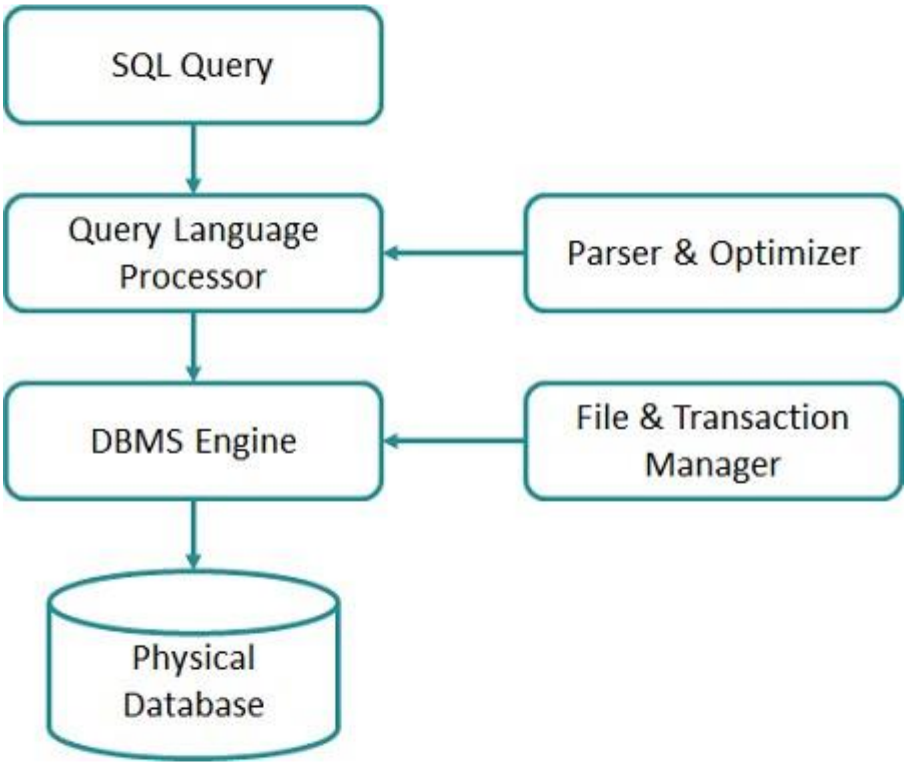


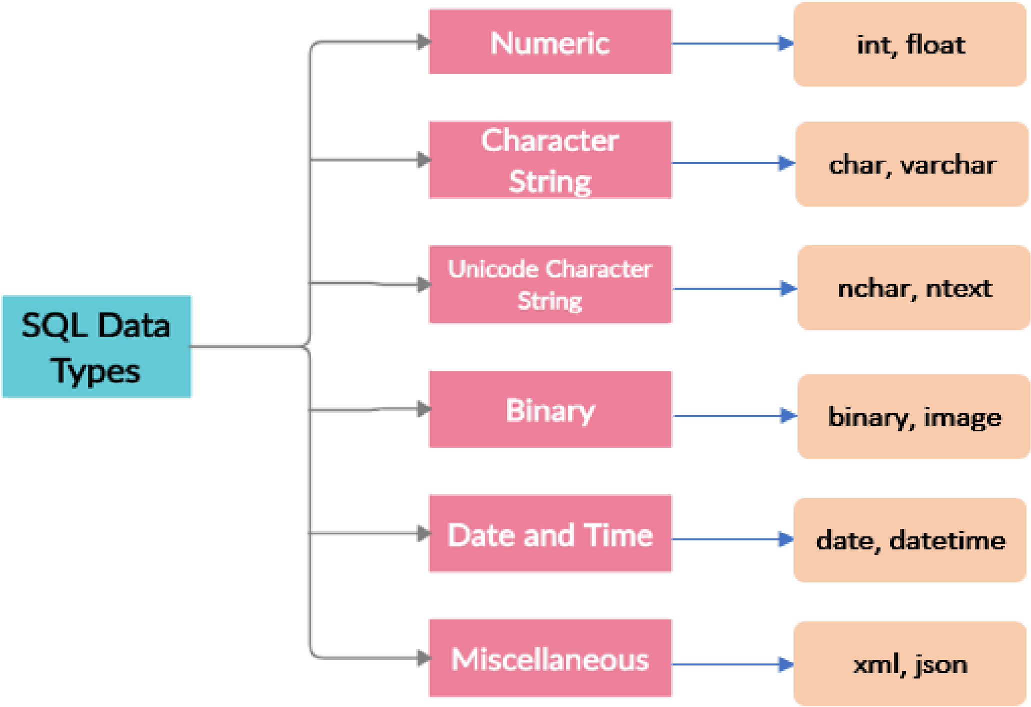
Figure (1): SQLprocess

Types of SQL Statements

Here are **five types** of widely used SQL queries.

- Data Definition Language (DDL)
- Data Manipulation Language (DML)
- Data Query Language (DQL)
- Data Control Language (DCL)
- Transaction Control Language (TCL)

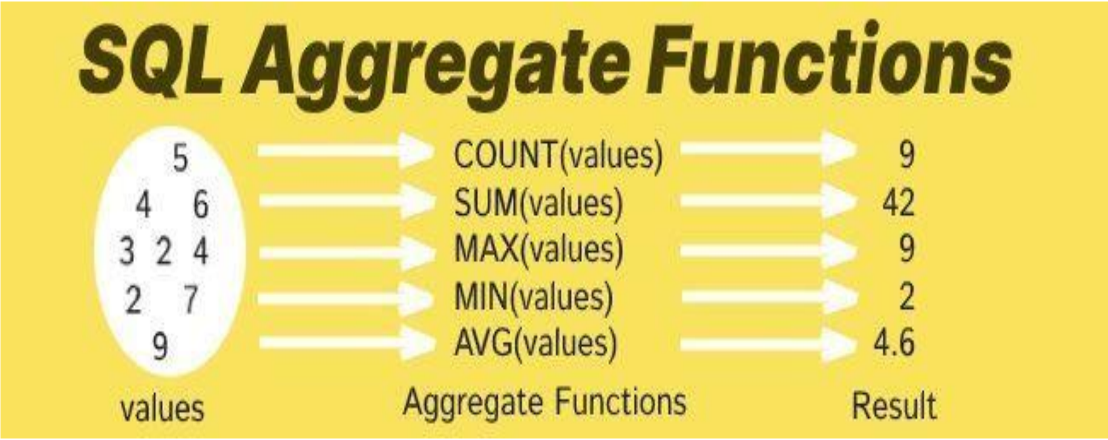
SQL Data type



SQL Aggregate Functions

- SQL aggregation function is used to perform the calculations on multiple rows of a single column of a table. It returns a single value.
- It is also used to summarize the data.

Types of SQL Aggregation Function


SQL Aggregate Functions
5
4 6
3 2 4
2 7
9
values
COUNT(values)
SUM(values)
MAX(values)
MIN(values)
AVG(values)
Aggregate Functions
9
42
9
2
4.6
Result