MODULE I

Database systems

Data: is the known facts that can be recorded and that have implicit meaning.

Data can be anything such as name, age and address etc. of a person. For example x, '25','student' etc. is a data.

Information: processed data. For example x is 25 years old and is a student is an information.

Database: is a collection of related data.

Database management system (DBMS) : is a collection of programs that enables users to create and maintain database.

Database management system (DBMS) is a general purpose software system that facilitates the process of defining, constructing, manipulating and sharing the database among various users and applications.

Defining a database involves specifying datatypes, structures and constraints of the data to be stored in the database. The database definition or descriptive information is also stored in the database in the form of database catalog or dictionary. It is called **meta-data**. i.e. data about data.

Constructing the database is the process of storing the data on some storage medium that is controlled by DBMS.

Manipulating a database involves function such as querying the database to retrieve the specific data, updating the database and generating reports from data.

Sharing a database allows multiple users and programs to access the database simultaneously.

Database systems: the database and DBMS software are collectively known as database systems.

An **application program** accesses the database by sending queries or requests for data to the DBMS. A **query** typically causes some data to be retrieved. A **transaction** may cause some data to be read and some data to be written into the database.

DBMS include protecting the database. Two types

1) *system protection* : protection against hardware or software malfunction.

2) *security protection* : protection against unauthorized or malicious access.

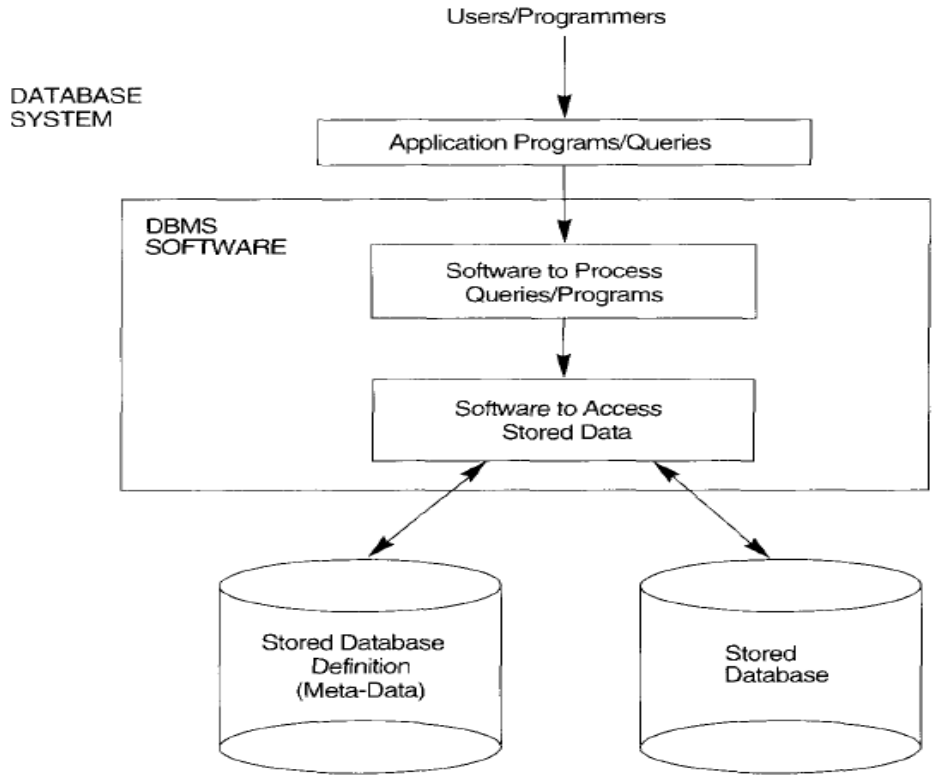

FIGURE 1.1 A simplified database system environment.

EXAMPLE

An example database, **Online Book** database which gives details of books punlished. Five relations namely book,publisher,author,review,author_book gives detials of author and book.

PUBLISHER

P_ID	Pname	Address	State	Phone	Email_id
P001	Hills Publications	12, Park street, Atlanta	Georgia	7134019	h_pub@ hills.com
P002	Sunshine Publishers Ltd.	45, Second street, Newark	New Jersey	6548909	null
P003	Bright Publications	123, Main street, Honolulu	Hawaii	7678985	bright@ bp.com
P004	Paramount Publishing House	789, Oak street, New York	New York	9254834	param_house @param.com
P005	Wesley Publications	456, First street, Las Vegas	Nevada	5683452	null

BOOK

ISBN	Book_title	Category	Price	Copyright _date	Year	Page_count	P_ID
001-354-921-1	Ransack	Novel	22	2005	2006	200	P001
001-987-650-5	Differential Calculus	Textbook	30	2003	2003	450	P001
001-987-760-9	C++	Textbook	40	2004	2005	800	P001
002-678-880-2	Call Away	Novel	22	2001	2002	200	P002
002-678-980-4	DBMS	Textbook	40	2004	2006	800	P002
003-456-433-6	Introduction to German Language	Language Book	22	2003	2004	200	P004
003-456-533-8	Learning French Language	Language Book	32	2005	2006	500	P004
004-765-359-3	Coordinate Geometry	Textbook	35	2006	2006	650	P003
004-765-409-5	UNIX	Textbook	26	2006	2007	550	P003

AUTHOR

A_ID	Aname	State	City	Zip	Phone	URL
A001	Thomas William	New York	New York	14998	923673	www.thomas.com
A002	James Erin	Georgia	Atlanta	31896	376045	www.ejames.com
A003	Charles Smith	California	Los Angeles	95031	419562	www.csmith.com
A004	Lewis Ian	Washington	Seattle	98012	578932	www.au_lewis.com
A005	Allen Ford	Alaska	Juneau	99502	581231	www.allenford.com
A006	Jones Martin	New York	Albany	14521	218161	www.martin.com
A007	John Stephen	Texas	Austin	75112	316292	www.john_stepen.com
A008	Annie George	Michigan	Detroit	48011	321313	www.annie.com
A009	Suzanne Hedley	Washington	Seattle	98001	785236	www.suzz.com
A010	Richard Flaming	Virginia	Virginia Beach	22111	456163	www.richard.com

AUTHOR_BOOK

A_ID	ISBN
A001	001-987-760-9
A002	001-678-980-4
A003	001-987-760-9
A004	003-456-433-6
A005	001-354-921-1
A006	002-678-880-2
A007	001-987-650-5
A008	002-678-980-4
A008	004-765-409-5
A009	004-765-359-3
A010	003-456-533-8

REVIEW

R_ID	ISBN	Rating
A001	002-678-980-4	2
A002	001-987-760-9	6
A003	002-678-980-4	5
A003	004-765-409-5	4
A004	003-456-533-8	9
A005	002-678-980-4	7
A006	001-354-921-1	7
A006	002-678-880-2	4
A007	004-765-359-3	3
A008	001-987-760-9	7
A009	001-987-650-5	8
A010	003-456-433-6	5

Fig.1.2 Relations in Online Book database

ADVANTAGEOUS OF DATABASE SYSTEM

In the database approach data is stored at a central location and is shared among multiple users. Thus the main advantage of DBMS is **centralized data management**, which overcome the limitations of *conventional file processing system*.

- *Controlled redundancy*
  During database design, various files are integrated and each logical item is stored at central location. This eliminates replicating the data item in different files and ensures consistency and saves the storage space.
- *Enforcing data integrity*
  Various integrity constraints are identified by the database designer during the database design.
- *Data sharing*
  Data stored in the database can be shared among multiple users or application programs.
- *Ease of application development*
  The application programmer needs to develop application programs according to the user's needs. The other issues like concurrent access, security and data integrity etc are handled by DBMS itself. This makes application development is an easier task.
- *Data security*
  DBMS provides security tools such as user codes and passwords. Different checks can be established for each access.

- *Backup and recovery*
  The DBMS provides backup and recovery subsystem that is responsible for recovery from hardware and software failures.
- *Multiple user interfaces*
  DBMS provides various type of interfaces such as query languages and application program interfaces and GUI that includes form-style and menu driven interfaces.

In addition to centralized data management, DBMS has some other advantages, they are

- *Program-data independence*
  The independence between program and data is known as program-data independence. i.e. DBMS allows changing the structure of the database without making any changes in application programs that are using the database.
- *Data abstraction*
  The property of the DBMS that allow program-data independence is known as data abstraction. Thus database system provides an abstract view of the data to its users without giving the physical storage and implementation details.
- *Supports multiple views of data*
  A database can be accessed by many users and each of them have different view of data.

DISADVANTAGEOUS OF DATABASE SYSTEM

- **High cost**: Installing new database may investment in hardware and software. The DBMS require more memory and disk storage.
- **Training new personnel**: when an organization plans to adopt a database system, it may need to recruit professionals in order to work with database system smoothly.
- **Explicit backup and recovery** : Database must be accurate and available at all times. Thus a system using online updation requires explicit backup and recovery.
- **System failure**: when a system containing database fails, all users have to wait until the system function again.
- **Size:** The complexity and breadth of functionality makes the DBMS an extremely large piece of software, occupying many megabytes of disk space and requiring substantial amounts of memory to run efficiently.

APPLICATION AREAS OF DATABASE SYSTEM

- ✓ Airlines and railways reservation
- ✓ Banking
- ✓ Education
- ✓ Health care information systems
- ✓ Digital libraries
- ✓ Finance
- ✓ Sales
- ✓ E-commerce

NEED OF A DATABASE SYSTEM

The need of a database system arose in response to the traditional file processing system. In file processing system, datas are stored in form of files and number of application programs to sdd, modify, delete and retrieve data to and from appropriate files.

- Same information is duplicated in several files. This is known as data redundancy, this leads to wastage of storage space. The other problem is that the data may not be updated consistently. i.e if any change is made in one file, it must change in every file that contains the changed data.
- Integrity rules can easily maintained. In file processing system, all the rules need to be explicitly programmed in all application programs, which are using that particular data item.
- In file processing system lacks the insulation between program and data. This is because the file structure is embedded in the application program itself.thus it is difficult to change the structure of a file as it requires changing all application programs accessing it.
- Handling new queries is difficult in file processing system, since it requires change in the application program or requires a new application program.
- Since many users are involved in creating files and writing application programs, various files in the system may have different file structures. Moreover programmers may choose different programming languages to write application programs.
- Details of each file are easily available to every user. Thus file processing system lacks security.

CHARACTERISTICS OF A DATABASE SYSTEM

- **Self-describing nature of a database system:** A DBMS catalog stores the *description* of the database. The description is called meta-data. This allows the DBMS software to work with different databases.
- **Insulation between programs and data:** Called program-data independence. Allows changing data storage structures and operations without having to change the DBMS access.
- **Data Abstraction:** A data model is used to hide storage details and present the users with a *conceptual view* of the database.
- **Support of multiple views of the data:** Each user may see a different view of the database, which describes *only* the data of interest to that user.
- **Sharing of data and multiuser transaction processing** : allowing a set of concurrent users to retrieve and to update the database. Concurrency control within the DBMS guarantees that each transaction is correctly executed or completely aborted.

DATABASE SCHEMA VERSUS DATABASE INSTANCE

The overall design or description or structure of the database is known as schema or database schema. It is also known as intension of the database.

The data in the database at a particular point of time is called database instance or database state or snapshot. It is also known as extension of the database.

The various sate of database are

- Empty state : when a new database is defined only its schema is specified. At this point, the database is said to be in empty state as it contains no data.
- Initial state: when a database is loaded with data for the first time, it is said to be in initial state.
- Current state: the data in the database is updated frequently. Thus at any point of time, the database is said to in current state.

The DBMS ensures that the database remains in valid state after each updation.

Subschema: - User view of a portion of the database.
Can have many subschemas for one database

BOOK

ISBN	Book_title	Category	Price	Copyright_date	Year	Page_count	P_ID

PUBLISHER

P_ID	Pname	Address	State	Phone	Email_id

AUTHOR

A_ID	Aname	City	State	Zip	Pnone	URL

AUTHOR_BOOK

A_ID	ISBN

REVIEW

R_ID	ISBN	Rating

Fig 1.3 database schema for Online Book database

P_ID	Pname	Address	State	Phone	Email_id
P001	Hills Publications	12, Park street, Atlanta	Georgia	7134019	h_pub@hills.com
P002	Sunshine Publishers Ltd.	45, Second street, Newark	New Jersey	6548909	Null
P003	Bright Publications	123, Main street, Honolulu	Hawai	7678985	bright@bp.com
P004	Paramount Publishing House	789, Oak street, New York	New York	9254834	param_house@param.com
P005	Wesley Publications	456, First street, Las Vegas	Nevada	5683452	Null

Fig1.4 An example of instance-PUBLISHER schema

- .

DBMS ARCHITECTURE and DATA INDEPENDENCE

The architecture describes how data in the database is viewed by users. I architecture overall description can be defined at three levels namely internal, conceptual and external levels and thus named three level DBMS architecture. This is proposed by ANSI/SPARC, hence it also known as ANSI/SPARC architecture.

The three levels are:

- Internal: It is the lowest level of the data abstraction that deals with physical representation of the database on a computer and thus is also known as physical level. It describes how data is physically stored and organized.
- Conceptual: deals with logical structure of entire database and thus also known as logical level. It describes what data is stored in the database.the relationship among the data. It hides the complexity of physical storage structures.
- External: it is the highest level of data abstraction, deals with the users view of the database. It also known as view level. The external level describes part of the database for a particular group of users.

The internal level has an internal schema, which describes the physical storage structure of the database. The conceptual level has conceptual schema which describes the structure of entire database. The external level has external schema, which describes the part of the database according to particular users requirements and hides the rest of the database from that user. The three schema architecture is shown in fig.

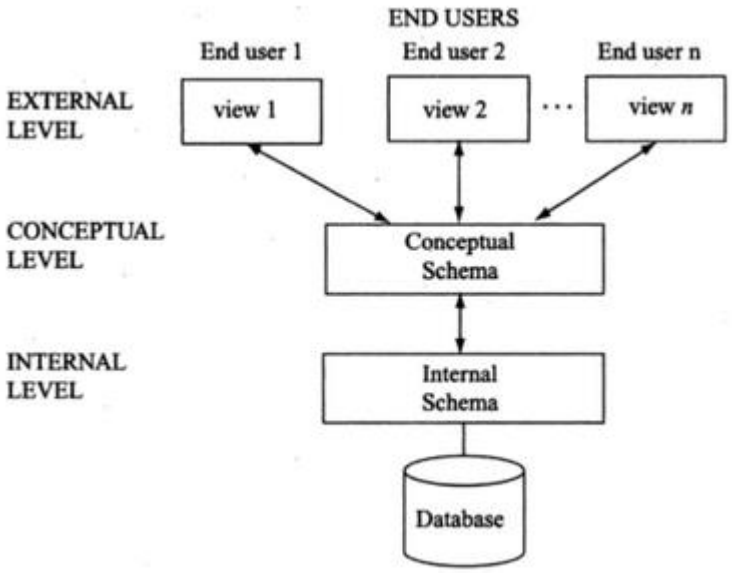

Fig.1.5 Three schema architecture

For example consider three levels of BOOK file in Online Book database is shown in figure . In this figure two views (view1 and view2) of BOOK file have been defined at the external level. Different database users can see these views. The details of the datatype are hidden from the users. At conceptual level the BOOK records are described by type definition. The application programmers and DBA generally work at this level. At internal level the BOOK records are described as a block of consecutive storage locations.the database users and tha application programmers are not aware of these details however the DBA may be aware of certain details of the physical organization of the data.

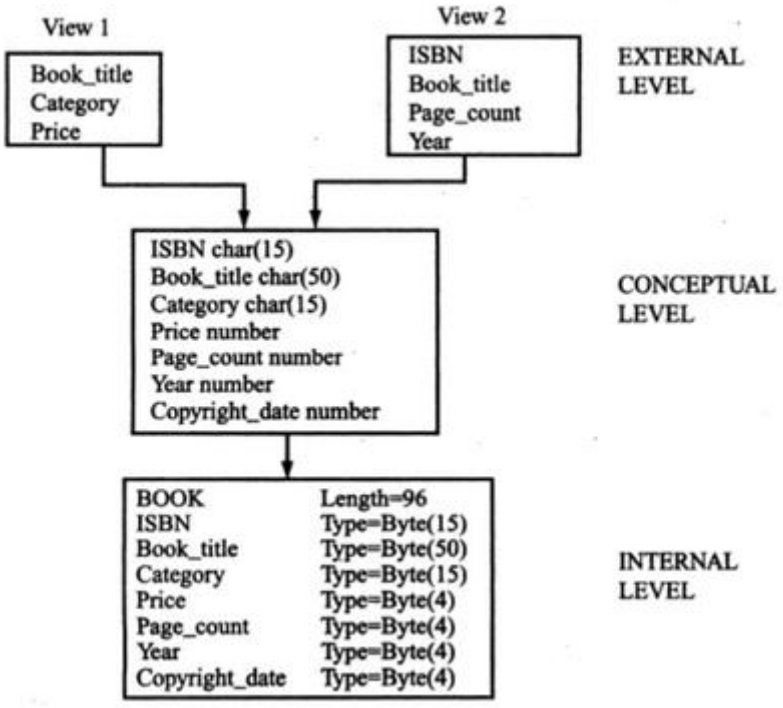

Fig1.6 The three levels of Online Book database (Book file)

In the three schema architecture each user group refers only to its own external view. Whenever a user specifies a request to generate a new external view, the DBMS must transform the request specified at external level into a request at conceptual level, and then into a request at physical level. If the user requests for data retrieval, the dta extracted from the database must be presented according to the needs of the user. This process of transforming the requests and results between various levels are known as mapping.

The main advantage is that it provides data independence. Data independence is the ability to change the schema at one level of the database system without having to change the schema at the other levels. Two types

- ❖ Logical data independence: It is the ability to change the conceptual schema without affecting the external schema.
- ❖ Physical data independence: It is the ability to change the internal schema without affecting the conceptual or external schema.

Logical data independence is difficult to achieve than physical data independence because application programmers are always dependent on the logical structure of the database. Therefore the change in the logical structure of the database may require change in the application programs.

DATA MODELS

Data model is an abstract model that describes how data is represented and used. Data model is a collection of concepts that can be used to describe the structure of a database. By structure of a database we mean data types, relationships and constraints that should hold the data. Mainly three types

1. Conceptual or high level data model
2. Representational or implementation data model
3. Physical or low level data model

Conceptual or high level data model

It provides the concepts that are close to the way many users perceive data. The main advantage is that the model is independent of implementation details and hence can be understood by even end users having nontechnical background.

Physical data model

It provides the concepts that describe the details of how data is stored in the computer. Concepts provided by low level model are generally meant for computer specialists, not for typical end users.

Representational or implementation data model

This data model hides some data storage details from the users; however can be implemented directly on a computer system. Various Representational or implementation data model are

a. Hierarchical data model
b. Network data model

c. Relational data model
d. Object-based data model
e. Semistructured data model

Hierarchical data model

In this data model data is organized in tree-like structure in which each child node (also known as dependents) can have only one parent node. The database based on hierarchical data model comprises a set of records connected to one another through links. The link is an association between two or more records. The top of the structure consist of a single node that does not have any parent and is called root node.

The root may have any number of dependents; each of these dependent may have any number of lower level dependents. E ach child node can have only one parent node and parent node can have any number of child node. i.e represents only one-to-one and one-to-many relationships.

The main advantage is that the data retrieval and updates are highly optimized by the DBMS. Drawback is the links are 'hard-coded' into the data structure that is link is permentantly established and cannot be modified. Likewise database modification also difficult.

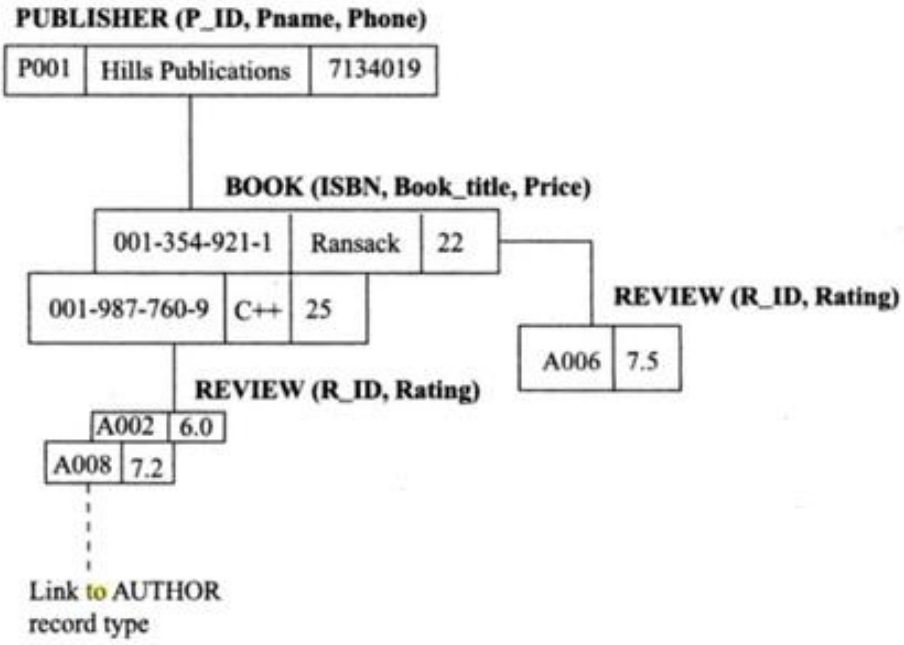

Fig 1.7. Hierarchical data model of Online Book database

Network data model

Data is represented as collection of records, and relationship among data are represented by links. However link in network model represents association between precisely two nodes. Like hierarchal model, each record of particular type represents a node. However unlike hierarchal mode, all nodes are linked to each other without any hierarchy. The main difference is that in hierarchal model data is organized in form of trees and in network data model data is organized in form of graphs.

The main advantage is that parent node can have many child node and a child can have many parent node. Thus it represents many-to-many relationships. The main limitation is complication in maintenance of links and a single broken link can lead to problem in database.

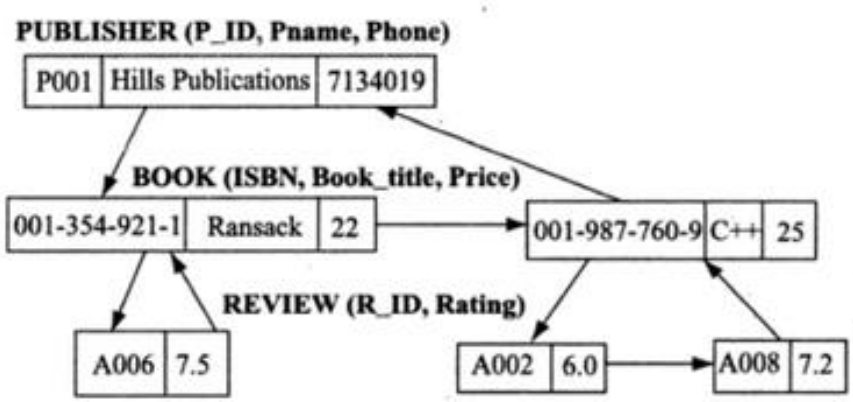

Fig.1.8 Network data model of Online Book database

Relational data model

There are no physical links. All data is maintained in form of tables (known as relations) consisting of rows and columns. Each row (record) represents an entity and column (field) represents an attribute of an entity. The relationship between two tables is implemented through a common attribute in the tables and not by physical links.

BOOK

ISBN	Book_title	Category	Price	Copyright_date	Year	Page_count	P_ID
001-354-921-1	Ransack	Novel	22	2005	2006	200	P001
001-987-760-9	C++	Textbook	25	2004	2005	800	P001

PUBLISHER

P_ID	Pname	Address	State	Phone	Email_ID
P001	Hills Publications	12, Park street, Atlanta	Georgia	7134019	H_pub@hills.com

REVIEW

R_ID	ISBN	Rating
A002	001-987-760-9	6.0
A006	001-354-921-1	7.5
A008	001-987-760-9	7.2

Fig.1.9 Relational data model of Online Book database

Object-based data model

Two types

i. *Object oriented data model* : extends the concepts object oriented programming languages

ii. *Object relational data model*: is an extension of relational data model. It combines the features of both relational and object oriented data model.

Semistructured data model

Unlike other data model, where every data model of particular datatype must have same set of attributes, the semistructured data model allows individual data items of the same type to have different set of attributes. This facilitates data exchange among heterogeneous data sources and querying the database easily without knowing its type.

COMPONENT MODULES OF DBMS

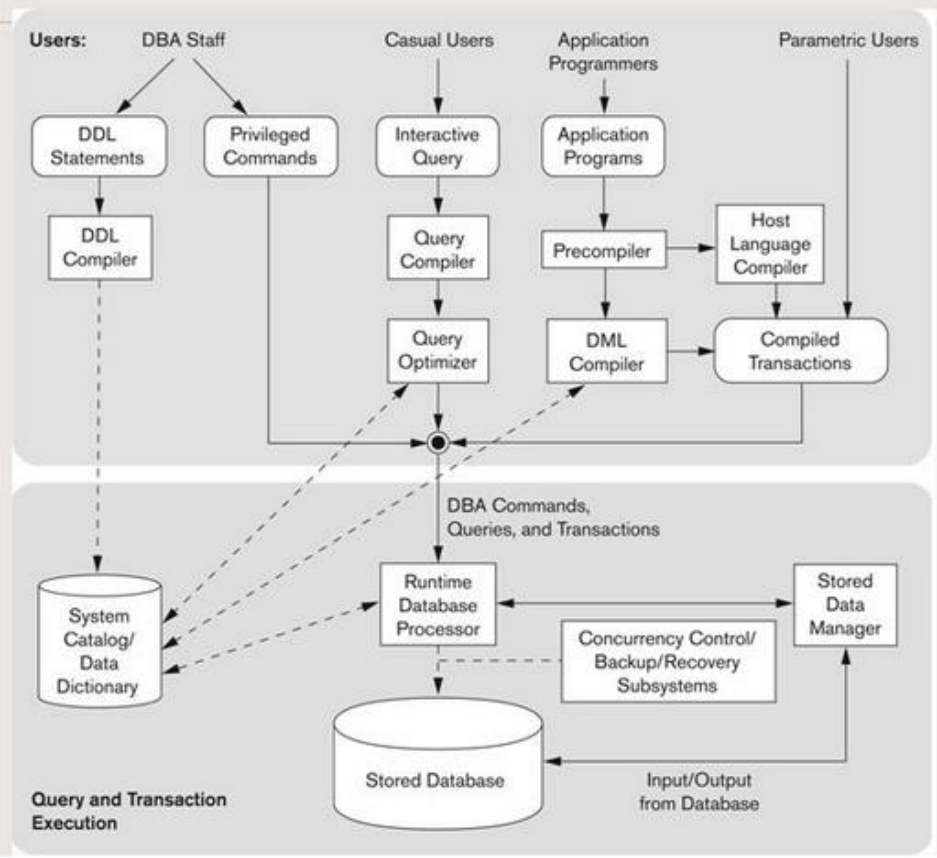

Fig.1.10 Component modules of DBMS

The figure is divided into two halves. The top half refers to various users of database environment and their interfaces. The lower half shows the internals of the DBMS responsible for storage of data and processing of transactions.

The database and DBMS catalog are usually stored on disk. Access to the disk is controlled primarily by OS which schedules disk input/output. The high level storage module of the DBMS controls access to the DBMS information stored on disk, whether it is a part of database or catalog.

The top half shows the interface for the DBA staff, casual users who work with interactive interfaces to formulate queries, application programmers who program using host languages and parametric users who do data entry work by supplying parameters to

predefined transactions. The DBA staff works on defining database and tuning it by making changes to its definition using DDL and other privileged commands.

The DDL compiler processes the schema definition, specified in DDL, and stores the description of the schemas in the DBMS catalog. The catalog includes the information such as name and size of files, name and data type of data items, storage details of each file, mapping information among schemas and constraints.

Casual users interact with the interface called interactive query interface. Query compiler compiles them into an internal form and this is subjected to query optimization which is concerned with rearrangement and possible reordering of operations, elimination of redundancies.

Application programmers write programs in host language such as c++, java that are submitted to precompiler. The precompiler extracts the DML commands from an application program and sent to DML complier for compilation into object code for database s=access. The rest of the program is sent to the host language compiler. The object code for DML commands and rest of the program are linked, forming a canned transaction whose executable code includes calls to the runtime database processor. These canned transactions are useful to parametric users who simply supply parameters to these canned transactions so that they can run repeatedly as separate transactions.

In the lower half of the figure runtime database processor is shown to execute (1) privileged commands (2) executable query plans (3) canned transactions with runtime parameters. It works with system dictionary and may update with its statistics. It works with stored manager, which in turn uses basic OS services for carrying out low level input/output operations between disk and memory.

CENTRALIZED AND CLIENT SERVER DATABASE SYSTEMS

Centralized DBMS: combines everything into single system including- DBMS software, hardware, application programs and user interface processing software.

Client/server Architectures:

Client is a computer system that sends request to the server connected to the network, and a server is a computer system that receives the request, processes it and requested information back to the client. Servers can be of several types for example print server, web server, database server etc. the client machine have user interface that help the users to utilize the servers. The architecture described here is called *two-tier architecture*, where components are distributed over two system: client and server.

Many web applications use an architecture called *three-tier architecture*; it adds an intermediate layer called application server (or web server). This plays as an intermediary role by storing business rules that are used to access data from the database server. It checks the clients credentials before forwarding a request to database server. Hence it improves data security. Hence a client request or information, the application server accepts the request, process it, and sends the corresponding database commands to database server. The database server sends the result back to an application server which is converted to GUI format and presented to the client.

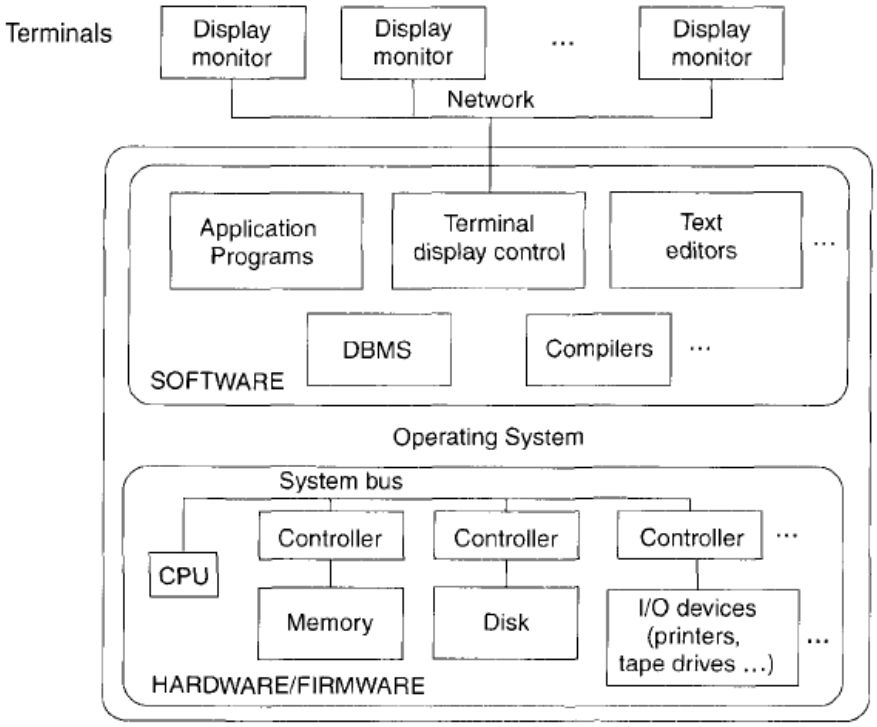

Fig.1.11 A physical centralized architecture

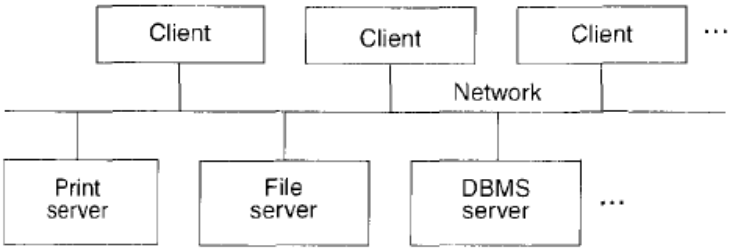

Fig1.12 Logical two-tier client/server architecture

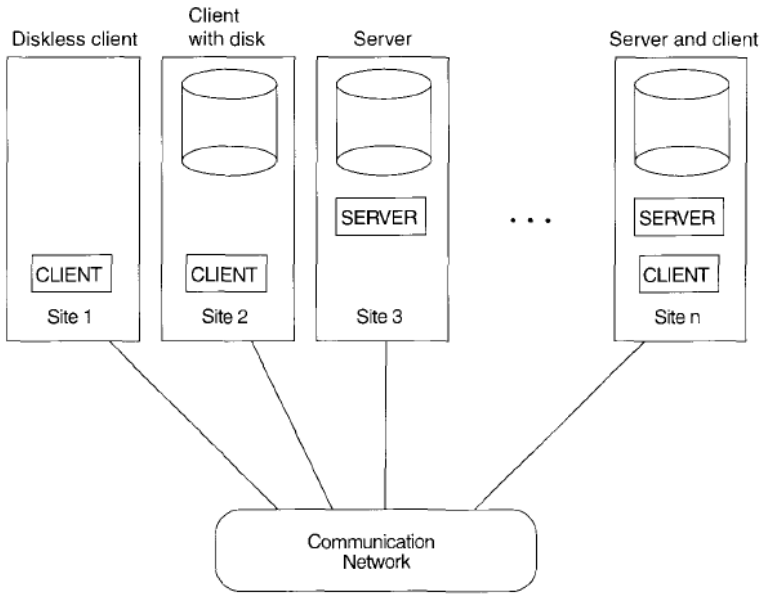

Fig 1.13 Physical two-tier client/server architecture

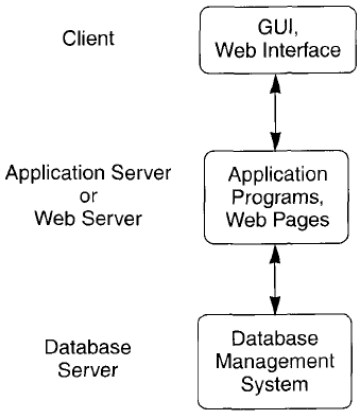

Fig1.14 Logical three-tier client/server architecture

CLASSIFICATIONS OF DBMS

The DBMS can be classified into different categories on the basis of several criteria such as data model they are using, number of users they support number of sites over which the database is distributed and the purpose they serve.

Based on Data Models

- Hierarchical
- Network
- Relational
- Object oriented
- Object relational

Hierarchical and network are oldest model, and is also known *as legacy data model*.

Based on Number of Users

- *Single user systems:* the database resides on one computer system and is only accessed by one user at a time.
- *Multi-user systems:* Multiple users can access the database simultaneously.

Based on Number of sites

- *Centralized database system*: runs on single computer. Both database and DBMS software residing at a single computer.
- *Distributed database system*: Database and DBMS software are distributed over several computers located at different sites. This can be classified as homogeneous and heterogeneous. In **homogeneous** distributed database system all sites can have identical database management system software, but **heterogeneous** distributed database system different sites use different database management system software.

Based on the purpose

- *General purpose:* can use for all applications.
- *Special purpose:* can design for specific applications such as banking, hospital etc.

PEOPLE WHO INTERACT WITH THE DATABASE

People who work with database include database users, system analysts, application programmers and database administrator (DBA).

Database users are those who interact with the database inorder to query and update the database, generate reports. Database users are classified into following categories.

- ✓ *Naïve users:* the users who query and update the database by invoking some already written application programs.
- ✓ *Sophisticated users:* the users such as business analysts, scientists etc., who are familiar with the facilities provided by a DBMS interact with the system without writing any application programs.
- ✓ *Specialized users:* the users who write specialized database programs such as banking, hospital management etc.

System analysts determine the requirements of database users to create a solution for their business need, and focus on non-technical and technical aspects. The non-technical aspects involve defining system requirement, facilitating interaction business users and technical staff etc. technical aspects involve developing specification for user interface.

Application programmers are the computer professionals who implement the specification given by the system analystsand develop application program to facilitate easy data access for database users.

Database administrator (DBA) is a person who has central control over both data and application programs. Some of the responsibilities of DBA are

- ✓ *Schema definition and modification*: it is the responsibility of DBA to create database schema by executing a set of statements in DDL. DBA also carries out changes to the schema according to the changing needs of organization.
- ✓ *New software installation*: It is the responsibility of DBA to install new DBMS software, application software and other related software.After installation DBA must test the new software.
- ✓ *Security enforcement and administration*: DBA is responsible for establishing and monitoring the security of the database system. It involves adding and remving users, auditing and checking for security problems.
- ✓ *Data analysis*: DBA is responsible for analyzing the data stored in the database.
- ✓ *Preliminary database design*: DBA work along with development team during the database design due to which many potential problems that can arise late can be avoided.
- ✓ *Physical organization modification*: DBA is responsible for carrying out modification in the physical organization of the database for better performance.
- ✓ *Routine maintenance check*: DBA is responsible for taking database backup periodically inorder to recover from any hardware and software failures.

- ***Actors on the scene***: People whose jobs involve the day-to-day use of a large database. They are database administrator, Database designers, End users, System analysts and Application programmers.

- ***Workers behind the scene-*** those who work to maintain the database system environment but who are not actively interested in the database itself. They are system designers and implementers, Tool developers, Operators and maintenance personnel

DBMS LANGUAGES

- **Data Definition Language (DDL):** DDL is used for specifying the database schema. SQL commands in DDL are CREATE, ALTER, DROP, TRUNCATE, RENAME.

- **Data Manipulation Language(DML):**
  DML is used for accessing and manipulating data in a database. SQL commands are Insert, update, delete.

- **Data Query Language (DQL):** SQL commands are Select
  In some text books select command is also under DML.

- **Tata Control Langauge (TCL):**
  SQL commands are commit, rollback.

- **Data Control Language(DCL):**
  DCL is used to provide certain privileges to a particular user. Privileges are rights to be allocated. SQL commands are GRANT and REVOKE.