UNIT 1

Data :

It is a fact which is to be recorded. such as alphabates, numbers, images, pictures videos, etc.

Data may be processed or unprocessed.

Database:

It is the collection of related information grouped together as a single item.It is a structured set of data stored in computer system in such a way that it can be accessed in simple and different ways

eg. Consider student data base having attributes, rollno, name & class.

S.Roll no.	Name	Class
76	Pratiksha	B.Sc (CS)
77	Sneha	B.Sc (CS)
78	Vaishali	B.Sc (CS)

DBMS:

It is a software which is used to provide simple and effictive method for accessing data from the database.

The aim of DBMS is to provide different ways and effective method for defining, constructing and manipulationg data base for various application..

Defining: It gives detail discription about each type of data...

Constructing :It gives information about storage medium and that is controlled by DBMS

Manipulation: Manipulating means processing on data such as update, add, delete, etc.

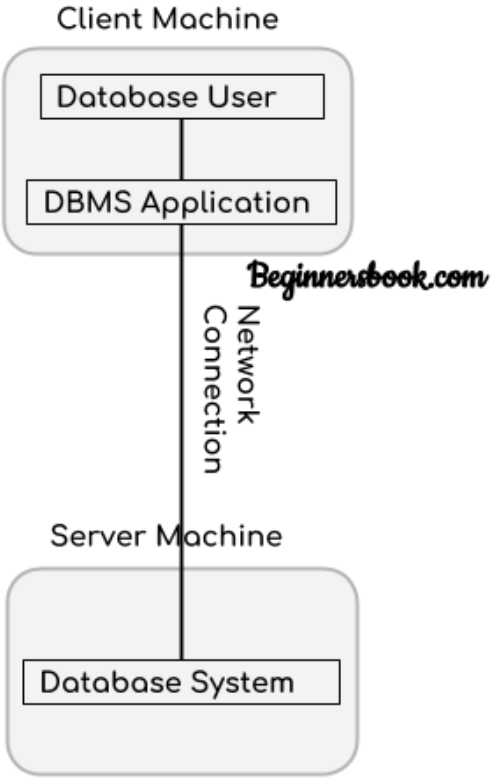
ARCHITECTRE OF DBMS

1. One/ Single tier architecture

In this type of architecture, the database is readily available on the client machine, any request made by client doesn't require a network connection to perform the action on the database.

For example, lets say you want to fetch the records of employee from the database and the database is available on your computer system, so the request to fetch employee details will be done by your computer and the records will be fetched from the database by your computer as well. This type of system is generally referred as local database system.

2. Two tier architecture

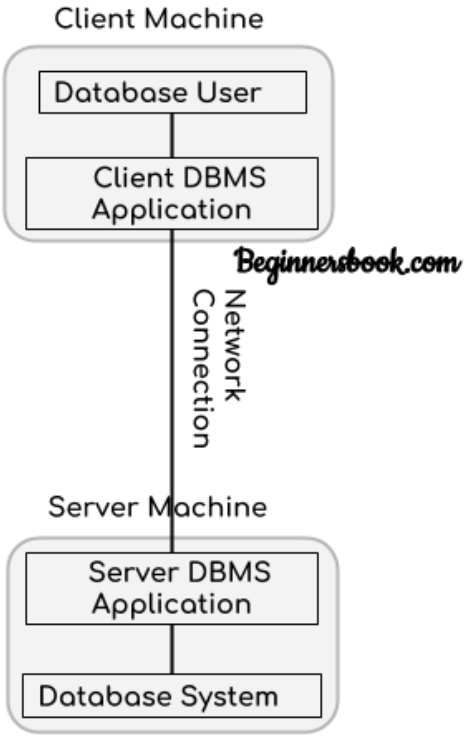


Two-Tier architecture

In two-tier architecture, the Database system is present at the server machine and the DBMS application is present at the client machine, these two machines are connected with each other through a reliable network as shown in the above diagram.

Whenever client machine makes a request to access the database present at server using a query language like sql, the server perform the request on the database and returns the result back to the client. The application connection interface such as JDBC, ODBC are used for the interaction between server and client.

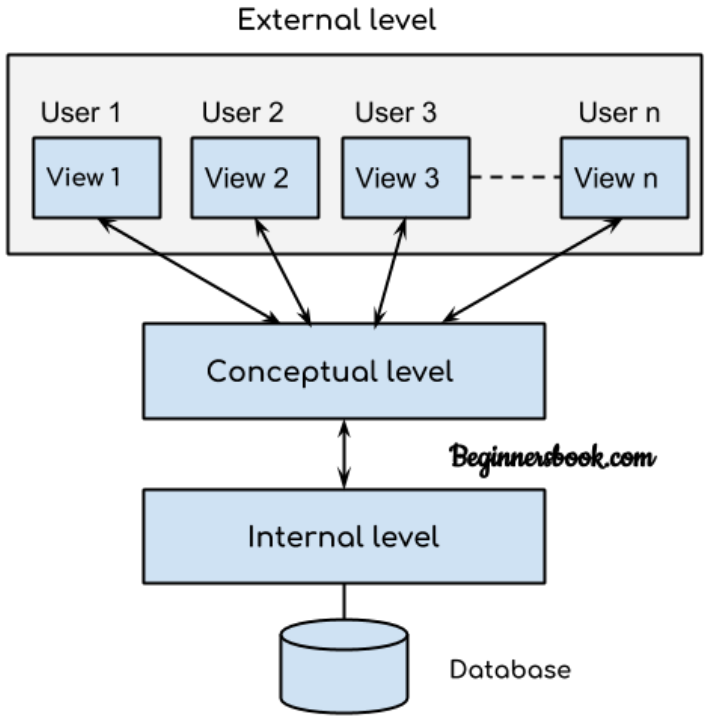
3. Three tier architecture



Three-Tier architecture

In three-tier architecture, another layer is present between the client machine and server machine. In this architecture, the client application doesn't communicate directly with the database systems present at the server machine, rather the client application communicates with server application and the server application internally communicates with the database system present at the server.

DBMS Three Level Architecture Diagram



This architecture has three levels:
1. External level
2. Conceptual level
3. Internal level /physical level

1. External level
It is also called **view level**. The reason this level is called “view” is because several users can view their desired data from this level which is internally fetched from database with the help of conceptual and internal level mapping.

The user doesn’t need to know the database schema details such as data structure, table definition etc. user is only concerned about data which is what returned back to the view level after it has been fetched from database (present at the internal level).

External level is the “**top level**” of the Three Level DBMS Architecture.

2. Conceptual level

It is also called **logical level**. The whole design of the database such as relationship among data, schema of data etc. are described in this level.

Database constraints and security are also implemented in this level of architecture. This level is maintained by DBA (database administrator).

3. Internal level

This level is also known as physical level. This level describes how the data is actually stored in the storage devices. This level is also responsible for allocating space to the data. This is the lowest level of the architecture.

ENTITY-

Entity is an object with physical existence or anything in the real word which is going to be recorded in database.

Ex-in a college database student can be consider as an entity.

Entity is an denoted in rectangle for ER diagram.

Entity set-

An entity set is a collection of similar type of entities. An entity set contains entity with some attributes such as roll no, name, class etc.

Ex-a student entity set contains all students of college.

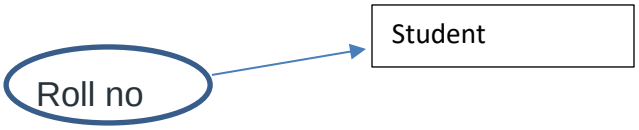
Attributes-

An attributes is properly of an entity or attribute is the name of column.

There are five types of attributes

1.single valued attributes

An attribute that cannot be further subdivided into components is a simple attribute.Which has only one value for particular entity
Example: The roll number of a student, the id number of an employee.



2. composite attributes

An attribute that can be split into components is a composite attribute.

.which has there own attribute.

Example: The address can be further split into house number, street number, city, state, country, and pin code, the name can also be split into first name middle name, and last name.

3. Multivalued attributes

The attribute which takes up more than a single value for each entity instance is a multi-valued attribute. It denoted by double ellipse.

Example: Phone number of a student: Landline and mobile.

4. Derived attributes

An attribute that can be derived from other attributes is derived attributes.

Example: Age of student can be calculated from date of birth. Therefor age is a derived attribute and derived attribute denoted as dashed eclipse.

5. key attributes

is an ID, key, letter or number that uniquely identifies that item.

It is an attribute that can be act as primary key for a table for uniquely identification of each recordin that table and it is denoted with ellipse and underline to attribute.

Ex-In a student database roll no is the key attribute and roll no gives unique identification for every student.

Relationship-

Any association between two entity types is called a relationship. Entities take part in the relationship. It is represented by a diamond shape.

For example, A teacher teaches students. Here, "***teaches***" is a relationship and this is the relationship between a Teacher entity and a Student entity.

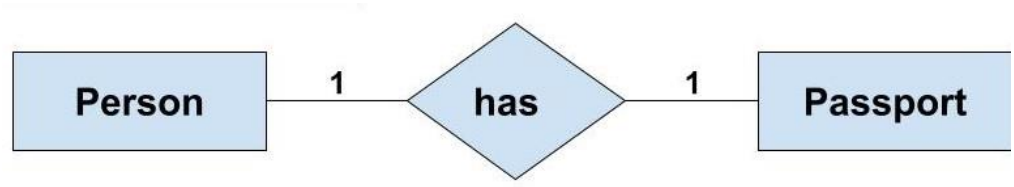
We have two entity types of 'Customer'(Customer_id, Name, City, Phone) and 'Account'(Account_no, Type, Balance). We store the data of 'Customer' in one table and his accounts details in the 'Account' table. Now, to link these two tables we need to insert the primary key 'Customer_id' of the 'Customer' table in the 'Account' table. This key acts as a foreign key for the 'Account' table and refers to a column with the same name in the 'Customer' table. This is how a relationship between two tables is established. There are three types of relationships that can exist between two entities.

Types of Relationship-

1.One to one –

Such a relationship exists when each record of one table is related to only one record of the other table.

For example, If there are two entities 'Person' (Id, Name, Age, Address)and 'Passport'(Passport_id, Passport_no). So, each person can have only one passport and each passport belongs to only one person.

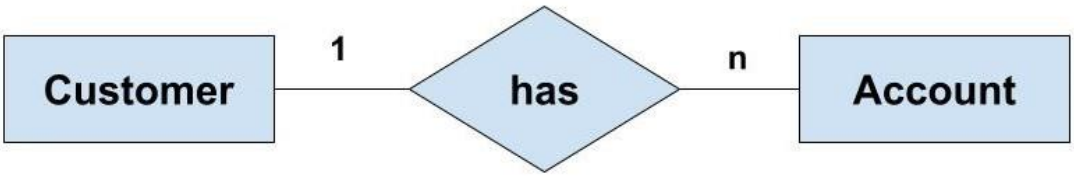


such a relationship is used for security purposes. In the above example, we can easily store the passport id in the ‘Person’ table only. But, we make another table for the ‘Passport’ because Passport number may be sensitive data and it should be hidden from certain users. So, by making a separate table we provide extra security that only certain database users can see it.

2.One to many-

Such a relationship exists when each record of one table can be related to one or more than one record of the other table.

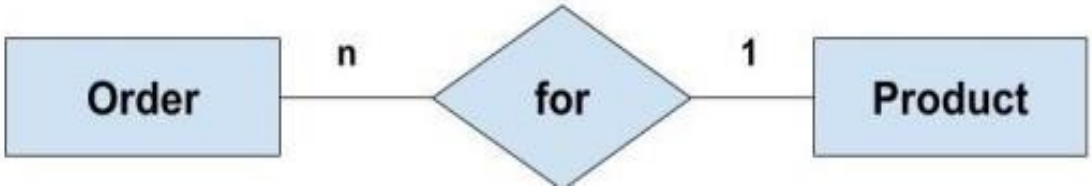
For example, If there are two entity type ‘Customer’ and ‘Account’ then each ‘Customer’ can have more than one ‘Account’ but each ‘Account’ is held by only one ‘Customer’. In this example, we can say that each Customer is associated with many Account. So, it is a one-to-many relationship.



3.Many to one-

More than one entity from entity set A can be associated with at most one entity from entity set B and one entity from entity set B can be associated with more than one entity from entity set A.

Ex-



4.Many to many-

More than one entity from entity set A can be associated with more than one entity from entity set B.

Example: If there are two entity type 'Customer' and 'Product' then each customer can buy more than one product and a product can be bought by many different customers.

DBMS users-

Native users/Navie users − The native users need not be aware of the presence of the database system. They are end users of the database who works through a menu driven application programs, where the type and range of response is always indicated to the user

Ex-ATM users use ATM machines for giviresponse according to instruction of machine such as enter pin no,etc.

Online users − Online users may communicate with databases directly through an online terminal or indirectly through user interface and application programs.

Sophisticated Users − They are those users who interact with the system without writing the program instead they form their request in database query language. They are the SQL programmers, who are going to deal directly with the database. They write queries to delete or select or insert and update the database.

Ex- Scientist ,businessman etc.

Specialized Users − Specialized users who write specialized database applications that do not fit into the fractional database processing framework.it writes specialized database application using complex data types such graphics,audio,etc.

Ex- computer Aided Designing system, knowledge based expert system.

Application Programmer − The application programmer users who are responsible for developing the application programs or user interface. The application programs could be written in high level language. For example − Java, .net, php etc,

Database Administrator (DBA) − It is a person or the group in charge of implementing the database system within the organization. The DBA has all the privileges allowed by the DBMS and can assign or remove the privileges from the users.

Some funtions of DBA-

1. Database design-

   The logical and physical design of database is designed by DBA.

2. Database accessibility-

DBA decides accessibility of database DBA decides the users of database and also he decides which data is to be used by which user.

3. Monitoring the performance-

   DBA has to monitor the performance of database. DBA monitors CPU, memory, etc.

4. Decides content of database-

   Database system has many kind of data in it and DBA decides the data in it and DBA decides the data type of field and range of the values .i.e. DBA decides the structure of database file.

5. Decides checks on data-

   DBA always puts validation check on data to make data more accurate.

6. Provide help and support to user –

   If any user needs help then it is the duety of DBA to help them and DBA gives complete support to the use who are new to the database.

Advantages of DBMS-

1. **Data integrity-**
   Integrity of database means that data in the database is always accurate so that the incorrect information can not be stored in database. Therefor to maintain! integrity of data some integrity constrain 1-ts are applied on database.

2. **Inconssitency can be avoided-**
   If the same data is duplicated and changes are made at one o e place) side then in 4 DBMS it make update at other side automatically if -you make update at one places

   e.g. Roll no. 150 shifted from pune to latur then address information of Roll no. 150 must be updated and this single update operation changes the address in hole database.

3. **Data can be shared-**

   Data about name, class, etc. can bel shared by different department in the college database. eg.. office file, Library file

4. **Controlling data redundancy-**
   Data redundancy means duplication of data. In data base system each data item is stored in only one place

5. **Providing backup and Recovery -**
   DBMS provides facilities for recoverin database from any failures of hardwork and software.

6. **Concurrent access-**
   DBMS provides multiusers environment and allows them to access the same database at the same time.

7. **Restriction to unauthorized access-**
   All users of DBMS do not have same access for database .
   Eg. Some user have read only access i.e. this users can read the content of file but can not make changes in the file.
   While some users has read and write access. Therefore they can read the file and make changes in the file.

Disadvantages of DBMS

The disadvantages of DBMS are as follows:

- **Complexity**

The provision of the functionality that is expected of a good DBMS makes the DBMS an extremely complex piece of software. Database designers, developers, database administrators and end-users must understand this functionality to take full advantage of it.

Failure to understand the system can lead to bad design decisions, which leads to a serious consequence for an organization.

- **Size**

The functionality of DBMS makes use of a large piece of software which occupies megabytes of disk space.

- **Performance**

Performance may not run as fast as desired.

- **Higher impact of a failure**

The centralization of resources increases the vulnerability of the system because all users and applications rely on the availability of DBMS, the failure of any component can bring operation to halt.

- **Cost of DBMS**

The cost of DBMS varies significantly depending on the environment and functionality provided. There is also the recurrent annual maintenance cost.

DATA MODEL-

Data model is used to show the logical design and structure of database and if also gives information in which manner data is to be stored, accessed, organized, manipulate.

There are main three types of data model

1.hierarchical model

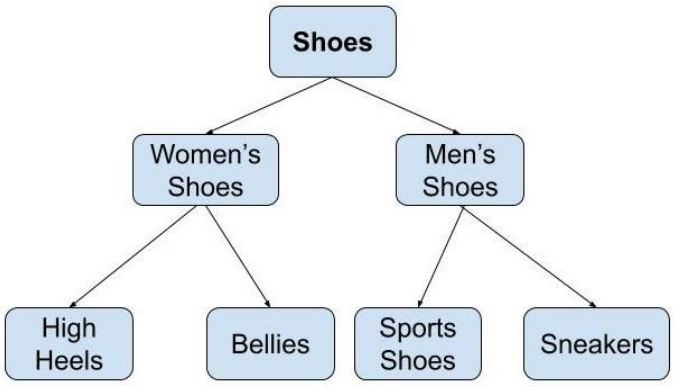
2.network model

3.relational model

Hierarchical Model-

Hierarchical Model was the first DBMS model. This model organises 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. In this model one to many relationship between two different type of data.

Example: We can represent the relationship between the shoes present on a shopping website in the following way:



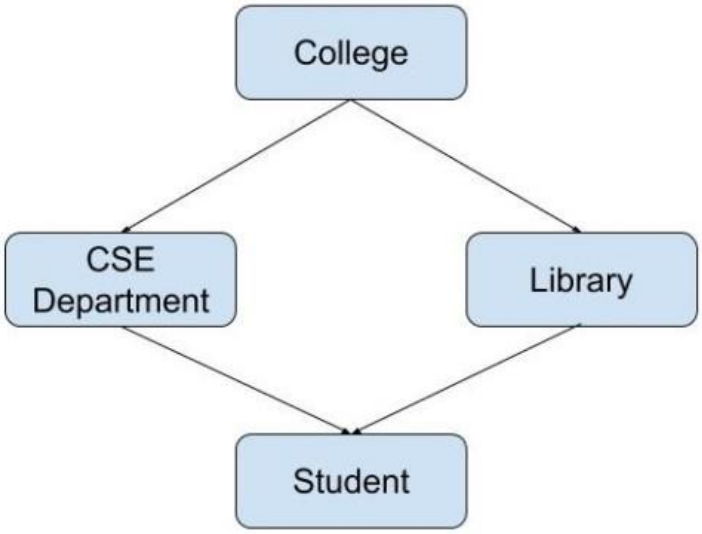
Hierarchical Model

Features of a Hierarchical Model

1. **One-to-many relationship:** The data here is organised in a tree-like structure where the one-to-many relationship is between the datatypes. Also, there can be only one path from parent to any node. **Example:** In the above example, if we want to go to the node sneakers we only have one path to reach there i.e through men's shoes node.
2. **Parent-Child Relationship:** Each child node has a parent node but a parent node can have more than one child node. Multiple parents are not allowed.
3. **Deletion Problem:** If a parent node is deleted then the child node is automatically deleted.
4. **Pointers:** Pointers are used to link the parent node with the child node and are used to navigate between the stored data. Example: In the above example the 'shoes' node points to the two other nodes 'women shoes' node and 'men's shoes' node.

Network Model

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 only difference 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

1. **Ability to Merge more Relationships:** In this model, as there are more relationships so data is more related. This model has the ability to manage one-to-one relationships as well as many-to-many relationships.
2. **Many paths:** As there are more relationships so there can be more than one path to the same record. This makes data access fast and simple.
3. **Circular Linked List***:* The operations on the network model are done with the help of the circular linked list. The current position is maintained with the help of a program and this position navigates through the records according to the relationship.

Relational Model

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:*** In this example, we have an Employee table.

Emp_id	Emp_name	Job_name	Salary	Mobile_no	Dep_id	Project_id
AfterA001	John	Engineer	100000	9111037890	2	99
AfterA002	Adam	Analyst	50000	9587569214	3	100
AfterA003	Kande	Manager	890000	7895212355	2	65

EMPLOYEE TABLE

Features of Relational Model

- **Tuples**: Each row in the table is called tuple. A row contains all the information about any instance of the object. In the above example, each

row has all the information about any specific individual like the first row has information about John.

- **Attribute or field:** Attributes are the property which defines the table or relation. The values of the attribute should be from the same domain. In the above example, we have different attributes of the employee like Salary, Mobile_no, etc.

DATABASE LANGUAGES-

- A DBMS has appropriate languages and interfaces to express database queries and updates.
- Database languages can be used to read, store and update the data in the database.

Types of database languages-

1.DDL Data Definition Language=

- **DDL** stands for **D**ata **D**efinition **L**anguage. It is used to define database structure or pattern.
- It is used to create schema, tables, indexes, constraints, etc. in the database.
- Using the DDL statements, you can create the skeleton of the database.
- Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

Here are some tasks that come under DDL:

- **Create:** It is used to create objects in the database.
- **Alter:** It is used to alter the structure of the database.
- **Drop:** It is used to delete objects from the database.
- **Truncate:** It is used to remove all records from a table.
- **Rename:** It is used to rename an object.
- **Comment:** It is used to comment on the data dictionary.

These commands are used to update the database schema that's why they come under Data definition language.

2.DML Data Manipulation Language=

DML stands for **D**ata **M**anipulation **L**anguage. It is used for accessing and manipulating data in a database. It handles user requests.

Here are some tasks that come under DML:

- **Select:** It is used to retrieve data from a database.
- **Insert:** It is used to insert data into a table.
- **Update:** It is used to update existing data within a table.
- **Delete:** It is used to delete all records from a table.
- **Merge:** It performs UPSERT operation, i.e., insert or update operations.
- **Call:** It is used to call a structured query language or a Java subprogram.
- **Explain Plan:** It has the parameter of explaining data.
- **Lock Table:** It controls concurrency.

3. DCL Data Control Language=

- **DCL** stands for **D**ata **C**ontrol **L**anguage. It is used to retrieve the stored or saved data.
- The DCL execution is transactional. It also has rollback parameters.

  (But in Oracle database, the execution of data control language does not have the feature of rolling back.)

Here are some tasks that come under DCL:

- **Grant:** It is used to give user access privileges to a database.
- **Revoke:** It is used to take back permissions from the user.

There are the following operations which have the authorization of Revoke.

4.TCL Transaction Control Language=

TCL is used to run the changes made by the DML statement. TCL can be grouped into a logical transaction.

Here are some tasks that come under TCL:

- **Commit:** It is used to save the transaction on the database.
- **Rollback:** It is used to restore the database to original since the last Commit.

DBMS STRUCTURE-

How exactly database works

- Overall structure of dbms

-about users and programers

-about query processor

-about storage manager

-about disk storage

First is a user view users always present in view level. They interact with query processor. It is complete logical level.

Following are the database users.

USERS-

1.Native users-

It is also called front end system. The native users need not be aware of the presence of the database system. They are end users of the database who works through a menu driven application programs, where the type and range of response is always indicated to the user

Ex-ATM users use ATM machines for giviresponse according to instruction of machine such as enter pin no,etc.

2. Application programmer− The application programmer users who are responsible for developing the application programs or user interface. The application programs could be written in high level language. For example − Java, .net, php etc,

3. Sophisticated user- They are those users who interact with the system without writing the program instead they form their request in database query language. They are the SQL programmers, who are going to deal directly with the database. They write queries to delete or select or insert and update the database.

Ex- Scientist ,businessman etc.

4.Database Administrator- It is a person or the group in charge of implementing the database system within the organization. The DBA has all the privileges allowed by the DBMS and can assign or remove the privileges from the users.

Query processor-

1.DML compiler-DML stands for **D**ata **M**anipulation **L**anguage. It is used for accessing and manipulating data in a database. It handles user requests.

2.DDL compiler- **DDL** stands for **D**ata **D**efinition **L**anguage. It is used to define database structure or pattern.

It is used to create schema, tables, indexes, constraints, etc. in the database.

Using the DDL statements, you can create the skeleton of the database.

Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

3.compiler and linker- It compile all query statement written in application program also they handle different procedural call. It linkers some library files or some supporting files that is work of compiler and linker.

4.query evaluation engine- It executes all instruction generated by DML compiler

Query pass to the evaluation engine.

About storage manager-

Queries received in query evaluation engine

Storage Manager is a program that provides an interface between the data stored in the database and the queries received. It is also known as Database Control System. It maintains the consistency and integrity of the database by applying the constraints and executes the DCL statements. It is responsible for updating, storing, deleting, and retrieving data in the database.
It contains the following components –

- **Authorization Manager –**
  It ensures role-based access control, i.e,. checks whether the particular person is privileged to perform the requested operation or not.

- **Integrity Manager –**
  It checks the integrity constraints when the database is modified.

- **Transaction Manager –**
  It controls concurrent access by performing the operations in a scheduled way that it receives the transaction. Thus, it ensures that the database remains in the consistent state before and after the execution of a transaction.

- **File Manager –**
  It manages the file space and the data structure used to represent information in the database.

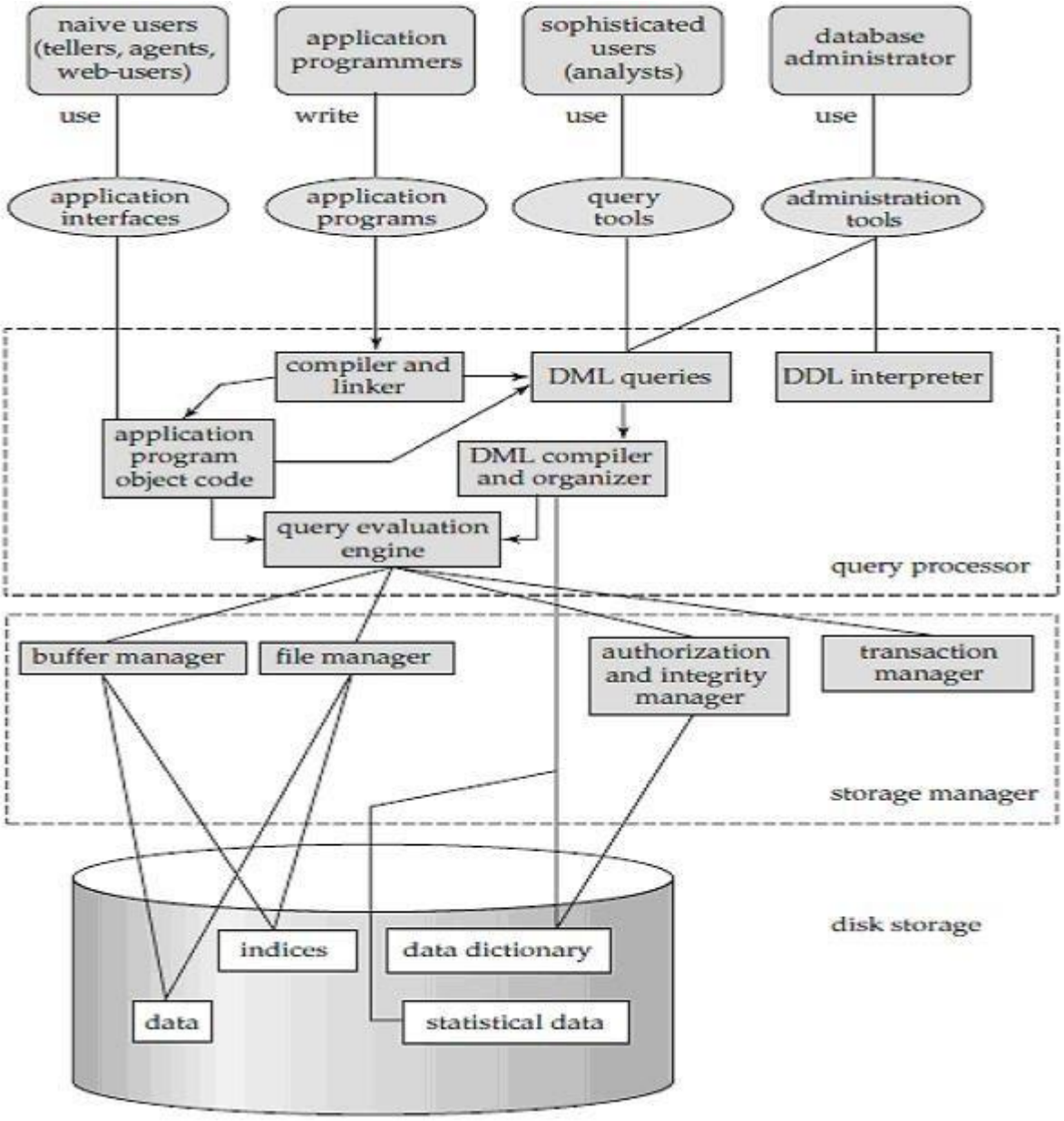
- **Buffer Manager –**
  It is responsible for cache memory and the transfer of data between the secondary storage and main memory.

3. Disk Storage: It contains the following components –

- **Data Files –**
  It stores the data.

- **Data Dictionary –**
  It contains the information about the structure of any database object. It is the repository of information that governs the metadata.

- **Indices –**
  It provides faster retrieval of data item.


naive users (tellers, agents, web-users)
application programmers
sophisticated users (analysts)
database administrator
use
write
use
use
application interfaces
application programs
query tools
administration tools
compiler and linker
DML queries
DDL interpreter
application program object code
DML compiler and organizer
query evaluation engine
query processor
buffer manager
file manager
authorization and integrity manager
transaction manager
storage manager
disk storage
indices
data dictionary
data
statistical data