DBMS Unit-1

DATE / /
PAGE

* What is Database Management Systems

Database is a collection of related data. The data we want to keep or need in future use called database.

have some implicit properties

(i) represent some aspect of real world called universe of discourse.

(ii) It is a logical collection of data.

(iii) Designed & built for specific purpose.

E.g.: Student Database

Name	Roll No	Branch	% of marks	Grade

DBMS

→ DBMS is a collection of interrelated data and a set of programs to access this data in a convenient and efficient way.

→ In other words, it is a collection of programs that enables users to create and maintain a database.

→ It is a general purpose s/w that facilitates the processes of defining, constructing, manipulating and sharing of database among various users and applications.

→ Defining a database :- means specifying the data types, structures and constraints for the data to be stored in database.

→ Constructing :- process of storing data itself on some storage medium controlled by DBMS.

→ Manipulating :- processing the database. It includes functions such as quering the database to retrieve specific data, updating database & generating reports from data.

→ Sharing : allows multiple users and programs to access the database concurrently.

* Advantages of DBMS over File Processing System

→ One way to store data is traditional file sharing system.

→ In this method, each data is stored in different files and there should be an appl^n program for each of the appl^n.

→ Disadvantages of File ~~sharing~~ system :

(i) Data redundancy and inconsistency

→ In traditional file systems data may be duplicate Foreg. consider a bank having two accounts. In this case address of customer is stored in two files. Thus, this duplication will result in need of high storage space which will also lead of inconsistency.

(ii) Difficulty in accessing information :

→ Eg, The bank needs a list of customers with an account higher than ₹10,000. To access this info. we have two ways. First is to extract data manually, second develop a new program to satisfy new request. Both are difficult

(iii) Data isolation

→ Data are scattered in different files and files may be in different format. So, it is difficult to extract appropriate data.

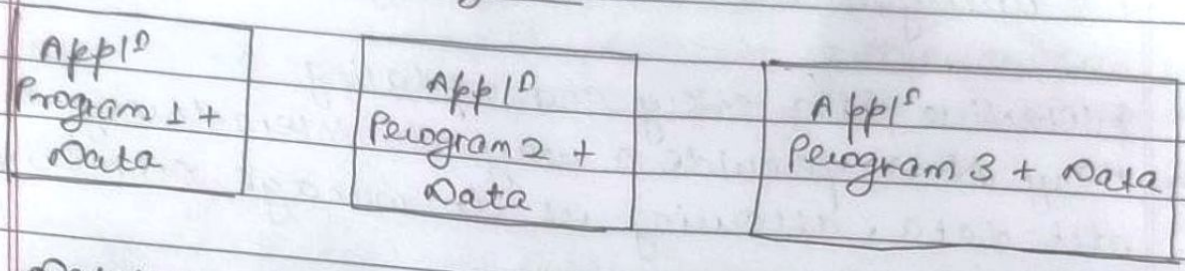
(iv) Integrity problems :

The constraint of data is enforced through the programs by appropriate codes. It is very difficult to add or change the constraints.

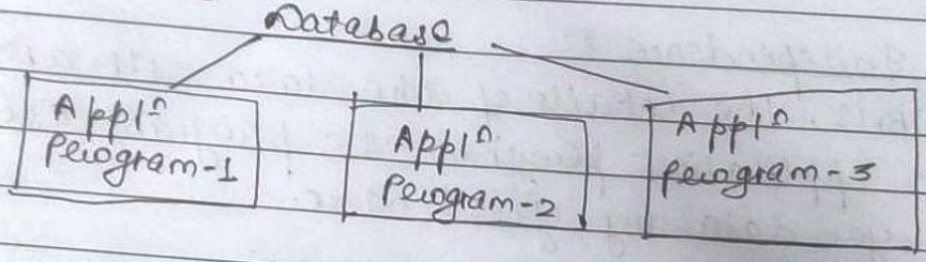
(v) Atomicity Problems :

→ Suppose a failure occurs during execution of program. Then the execution stops in the middle of the program resulting in inconsistency.
For a traditional file sharing the failure mostly result to an inconsistency state.

Traditional File System



Database Systems



* Introduction And Application of DBMS :-

Features of DBMS :-

1- Self-Describing :-

→ The DB system contains complete definition or description and structure of db.

→ The structure is stored in catalog with type, storage format and constraints. The info. stored in db is called meta-data

2. Data Security :-

→ The DBMS can prevent unauthorized users from viewing or updating the db.

→ Using passwords, users are allowed access to the entire database or a subset of it known as "subschema".

3. Data Integrity :-
→ DBMS keep duplicate records out of db. For e.g, no two customers with same customer no. can be intered

4. Interactive Query :-
→ Most DBMS provide query language and report writers that let us obtain data from db and analyze it. We will get all details at any time

5. Interactive data entry and updating :-
→ Many DBMS provide a way to interactively enter edit data, allowing us to manage our own files and databases.

6. Data Independence :-
→ With DBMS, the details of the data str. are not stored in each application program. The program asks the DBMS for data by field name.

(*) DBMS components :-
* Data :-
e.g, intergers, floating point no., characters, pictures & images.

* Standard operations :-
→ St. operations are provided by most DBMS.
→ e.g, sorting, deleting and selecting records

* Data definition language : (DDL)
→ used to describe the contents of the database.
→ e.g, attribute name, data-type, location in the db.

Deductive db → db system that can make conclusion about its data base on set of well-defined rules & facts.

DATE / /
PAGE

* Data manipulation & query language :-
→ Normally a query language is supported by DBMS to form commands for input, edit, analysis, output etc.
→ Some degree of standardization has been achieved with SQL.

* Programming tools :-
→ Besides query languages, the db should be accessed directly from appl° program through function call in conventional programming language.

* File structures :
→ Every DBMS has its own internal st used to organize data.

* Advantages of DBMS :-
1. Controlling Redundancy → disad. of file system
2. Restricting Unauthorized Access :- → feature of DBMS
3. Providing persistent storage for Program objects and Data structures :-
→ Data structure provided by DBMS must be compatible with programming language's data structures.
e.g. Object Oriented DBMS are compatible with C++, and DBMS s/w automatically performs conversion betⁿ programming data structure & file formats.
4. Deductive database systems provide capabilities for defining deduction rules to inference new info. from stored db facts.
5. Providing Multiple user interface

DATE / /
PAGE

* DBMS Disadvantages :-

1. Confidentiality, Privacy and Security :-

→ The centralized data that is made available to users from remote locations, the possibility of abuse are often more than in conventional method.

→ It is necessary to take technical, administrative and possibly legal measures

2. Data Quality :-

→ With increased no of users accessing data directly, there are enormous opportunities for users to damage the data.

3. Data Integrity :-

→ The main threat to data integrity comes from several users attempting to update data at same time.

→ The db needs to be protected against accidental changes by the users.

4. Enterprise vulnerabilities :-

→ Centralizing all data of an enterprise in one db makes the db critical resources.

→ The enterprise becomes vulnerable to the destruction of db as the enterprise survival depends on the info. of db.

5. The cost of using a DBMS :-

→ DBMS is less costly as compared to conventional data processing system.

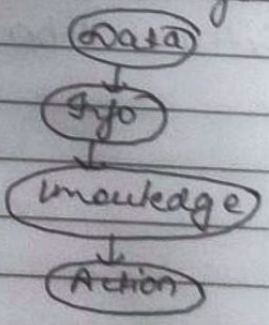
* Applications of DBMS :-

1. Railway Reservation System.
2. Library Management System
→ DBMS is used to maintain all the info. related to book
3. Banking.
4. Universities & colleges
5. Credit card transactions
→ the purchase of items and credit card transactions are made possible by DBMS. A credit card holder knows the importance of their info through DBMS.
6. Social Media Sites
7. Finance → storing sales, holding info.
8. Military.
9. Online shopping
10. Human Resource Management
11. Manufacturing companies.
12. Airline Reservation systems.

* Purpose of DBMS :-

→ The purpose of DBMS is to transform the following :-

(i) Data into information
(ii) Information into Knowledge
(iii) Knowledge to the action.



* Degree of Data abstraction :

→ We all know that each appl° program have some data relevant to a particular task.
→ Each appl° program uses its own data, details concerning the structure of data as well as the access and to interpret each data.

→ The appl° program are implemented independently and by hence utely, any change in storage media requires changes to these structures & access methods.

→ ~~e.g Two appl° program that require data on existing set EMPLOYEE.~~

→ The file were structured for one appl°, it was difficult to use the data in these files to new applications requiring data from several files belonging to different existing application

* Three Schema Architecture in DBMS :-

→ The generalized architecture of DBMS is called ANSI/SPARC model.

→ The architecture is divided in 3-levels

→ A schema describe the view of this level. Also describes the records and its relationship in the view.

(i°) External view or User view :-

→ This includes only portions of database concern to a user.

→ It is the highest level of data abstraction.

→ Each user has a different external view and is described by means of a scheme called external schema

→ External schema contains – definition of logical view and relationship in external view
– methods of deriving the objects in external view from objects in conceptual view.

(ii°) Conceptual view :-

→ all the database entities and the relationship among them are included.

→ One conceptual view represents the entire database called conceptual schema.

→ Specify the check to retain data consistency and integrating

REDMI NOTE 7 PRO

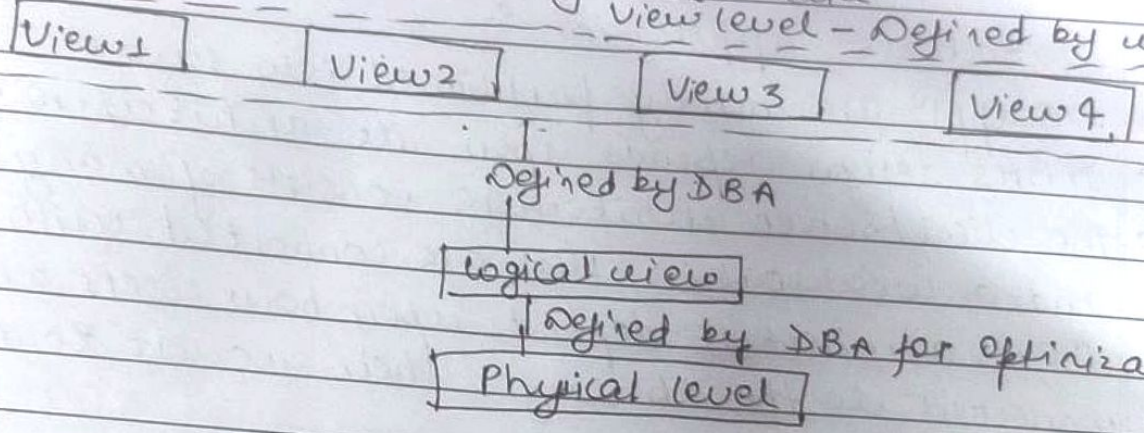
→ It describes the method of deriving the objects in conceptual view from object in internal view.

(iii) Internal view :-

→ Lowest level of abstraction, closest to physical storage method.

→ Represented by internal schema.

→ Describes how data is stored, structure of data stored and methods of accessing these data.



* Data Independence :-

→ Capacity to change the schema at one level of db system without having to change next high levels.

→ Two types :-

(i) Logical data independence :-

→ Capacity to change the conceptual schema without having to change external schema.

→ We may need to change conceptual schema to expand the database, to change constraints or to reduce the database.

→ Only the view definitions and mappings need to be changed that support logical data independence.

→ The appl^n programmer will not feel any change in schema content.

(ii) Physical Data independence :-

→ Capacity to change the internal schema without having to change conceptual & external schema.
→ The internal schema may change to improve the performance of retrieval or update.
→ e.g. we need not to change the Query to retrieve a student progress report even though the DBMS take a new method to store the student record.

* Database Application :-

→ DB appl° are mainly partitioned into two or 3 parts:
→ DBMS design depends upon its architecture.
→ The client/server architecture consists of many PCs and a workstation that are connected with n/w
→ DBMS architecture depends upon how users are connected to the db to get their request done.

* Types of DBMS architecture :-

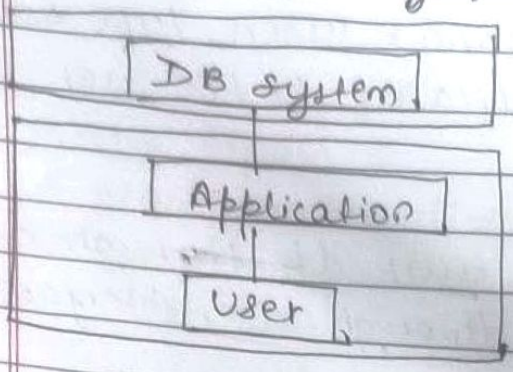
(i) 1-Tier :-

→ DB is directly available to the user, which means user can directly sit on DBMS & use it
→ Any changes done here will directly be done on db itself.
→ Doesn't provide a handy tool for users.
→ Used for development of local applications

(ii) 2-Tier :-

→ Same as basic client/server architecture.
→ The appl°s on client end can directly communicate with db on server side. by establishing a connection
→ On client side :- user interface and appl° programs are run.

→ On server side :- query processing & transaction management.



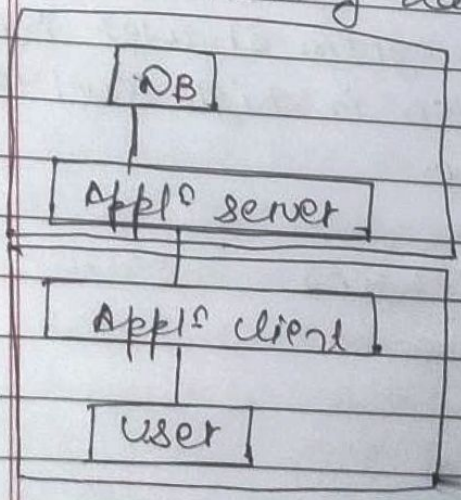
(iii) 3-Tier :

→ Contains another layer bet^n client & server called appl^n server.

→ The appl^n on the client side - interacts with an appl^n server which further communicates to DB system.

→ End user has no idea about existence of db beyond appl^n server.
The db has no idea about any user beyond the appl^n

→ Uses in case of large web application.



* Database users and DBA :-
Types of users :-

1. Naive users :-

→ They need not to be aware of the presence of db system.

→ They are end users of the db who work through menu driven appl° program, where type & range of response is always indicated to the user.

2. Online users
→ Can communicate directly with db through online terminal and indirectly through user interface and appl° program.

3. Sophisticated users
→ they are SQL programmers, who are going to deal directly with the db.
→ They write queries to delete, insert, & update the db.

4. Specialized users
→ who write specialized db appl° that do not fit into the traditional db processing framework.

5. Appl° Programmer
→ For developing the appl° program of user interface
→ This program could be written in high level lang. such as Java, .net, Php etc.

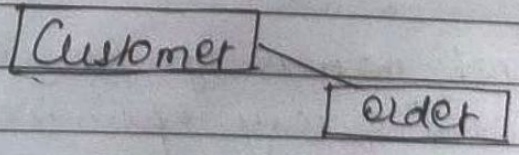
* DBMS mainly classified into 3- users
End users
Appl° Programmers
DB Administrator

* DB Administrator (DBA) :
→ A person or group in charge of implementing the db system within the organization.
→ the DBA has all the privileges allowed by

DBMS and can assign or remove privileges from the users.

* Data Model :
→ A plan for building a db.
→ the model represents data conceptually, the way user sees it rather than how computer stores it.
→ Used do represent db's overall structure.
→ Mainly used data modeling techniques are
(i) Entity-Relation Model
(ii) Hierarchial "
(iii) network "
(iv) Object-oriented "

Hierarchial Model :-
→ Organizes data in a tree structure. and there is a hierarchy of parent & child data segments.
→ This structure implies a record can have repeating info generally in child data segment.
→ Data in a series of records have a set of field values attached to it.
→ It collects all the instances of specific record as a record type.
→ These record types are equivalent to tables in relational model, with individual record being equivalent of rows.
→ To create link bet these record types, the hierarchial model uses parent child relationship.

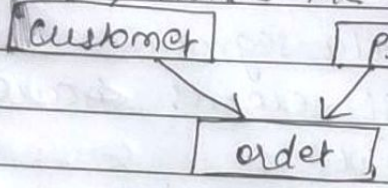


DATE / /
PAGE

* Adv:
(i) Simple to construct and operate on
(ii) lang. is simple, uses commands like GET, GET UNIQUE, GET NEXT, etc.

* Disadv:-
(i) Navigational & procedural nature of processing.
(ii) Little scope for "query optimization".
(iii) DB is visualized as a linear arrangement of records.

Network Model
→ The basic data-modeling construct in the network model is the set construct
→ A set consists of an owner record type, a set name, and a member record type.
→ In network db, a record type can have multiple owners, hence the multiparent concept is supported



* Adv:-
(i) Able to model complex relationships
(ii) Language is navigational, uses commands like FIND, FIND member, FIND owner etc.
(iii) Can handle most situations for modeling using record type and relationship types.

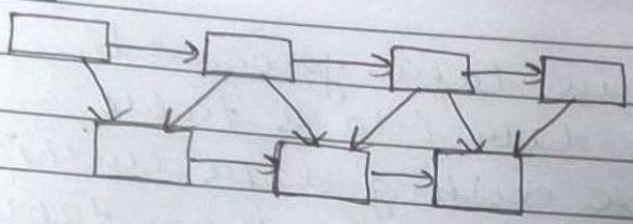
* Disadv.
(i) Navigational & procedural nature of processing
(ii) Little scope for automated query optimization
(iii) DB contains a complex array of pointers that threads through a set of records.

DATE / /
PAGE

Object-Oriented Model :-

→ object OBMS add of OB functionality to object programming languages.
→ Object DBMS extend the semantics of C++, Java to provide full-featured db programming capability, while retaining native language compatibility.
→ A major adv. of this approach is unification of the application and db development into a seamless data model and lang. environment.
→ As a result, applications require less code, and code bases are easier to maintain.

Objects



* OB language :-

→ A DBMS has appropriate languages and interfaces to express db queries and updates.
→ OB languages can be used to read, store and update the data in the db.
→ Types of DB lang

1. DDL → Data Definition language
2. DCL → " Manipulation "
3. DHL → " control "
4. TCL → Transaction " "

1. Data Definition language :-

→ It is used to define db structure or pattern.
→ Used to create schemas, indexes, constraints etc in db.
→ Used to store the info. of metadata like no. of

tables and schemas, their names, indexes etc.

* Some tasks that come under DDL :-
  Create → used to create object in db
  Alter → " " alter the structure of db
  Drop → " " delete objects from db
  Truncate → " " remove all records from a table
  Rename → " " rename an object
  Comment → " " comment on the data dictionary.

2. Data Manipulation Language :-
→ used for accessing and manipulating data in a db. Handles user request.

  Select → retrieve data from db
  Insert → insert data into a table
  Update → update existing data within a table
  Delete → delete all records from table
  Merge → performs UPSERT operation i.e. insert or update operations
  Call → used to call a SQL
  Explain Plan → has the parameter of explaining data
  Lock Table → controls concurrency.

3. Data Control Language :-
→ Retrieve the stored or saved data
→ DCL execution is transactional and also has rollback parameters.

  Grant → to give user access privileges to db
  Revoke → used to take back permissions from the user.

There are following operations which have

DATE / /
PAGE

authorization of Revoke
CONNECT, INSERT, USAGE, EXECUTE, DELETE, UPDATE and SELECT

4. Transactional control language
→ used to undo the changes made by DML statements.

Commit → used to save transaction on the db
Rollback → " " restore the db to original