DBMS

Module 1

Introduction to Database Concepts

Written and Compiled
by
Prof.Priyanka Bhilare

PAGE No.
DATE / /

1. Introduction Database Concepts.

(*) INTRODUCTION :-

1) Data :-

Data usually refers to raw data or unprocessed data. It is the basic form of data, data that hasn't been analyzed or processed in any manner.
eg :- student's mid term marks.

2) Information :-

Once the data is analyzed, it is considered as information. Data and information are interrelated. In fact, they are often mistakenly used interchangeably. Once the data is processed, organized, structured or presented in a given context it becomes an information. eg. Avg. mid term marks of the class.

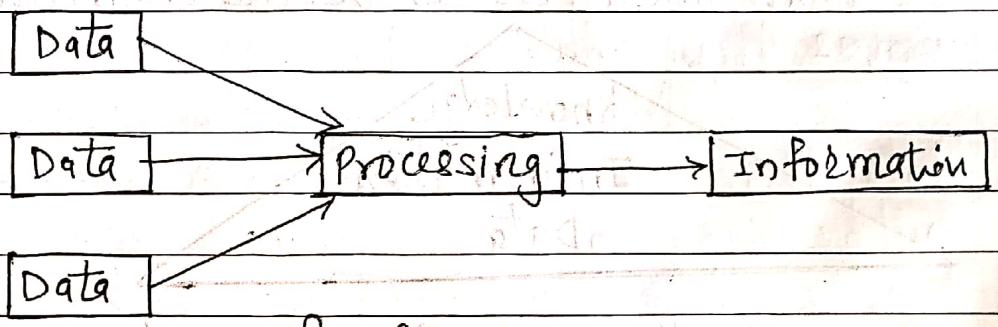

fig. Processing System

Scanned by CamScanner

PAGE No.
DATE / /

Difference between data and information :-

1) Data is used as input for the computer system. Information is the output of data.

2) Data is unprocessed facts, figures. Information is processed data.

3) Data is not specific. Information is specific.

4) Data doesn't carry a meaning. Information carry a meaning. (Logical meaning).

5) Data is the raw material. Information is the product.

* **Knowledge :-**

Knowledge defines what we understood or what we got to know from the information. avg.

eg:- Based on mid term marks we can conclude that one class is better than other class.

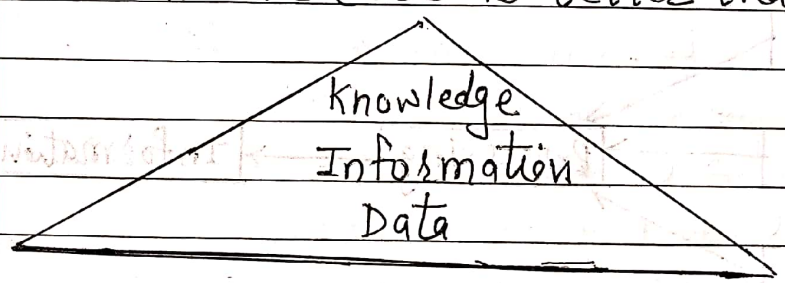

fig. levels of Info. System.

Scanned by CamScanner

PAGE No.
DATE / /

* Database :-

A database is collection of related data organised in a way that data can be easily accessed, managed and updated.

In a database, data is strictly organized in row and column format.

Roll no.	Name	age
1	priyanka	19
2	Snehal	22
3	Vipul	28

fig. Student Table

The rows are called Tuple or Record. The data items within one row may belong to different data types. On the other hand, the columns are often called Domain or Attribute. All the data items within a single attribute are of the same data type. The columns of the relations are called Fields, Attributes or Domains. The rows are also called Tuples or Records. Therefore, Database is also called as collection of related relations.

Scanned by CamScanner

PAGE No.
DATE / /

* Database Management System (DBMS) :-

A database Management System is a set of rules and procedures which help us to create, organize and manipulate database. It is basically a large software package which helps us to add, modify, delete data items in the database.

Need :-

Suppose, there is no database system in the college. Accessing information of particular student among hundreds would be difficult task. An individual has to manually search through all the records of the students, till they find the relevant person and accessing his/her information on grades would be more complicated.

Maintaining database overcomes this this problem. Users can just type in and get all the information regarding the relevant person quickly without any effort. This saves the valuable time.

Most modern enterprises have data that they need to store in ways which will be easy to retrieve later. One of the most type of database is list of names and addresses of people who regularly deal with the enterprise. Originally when a business is small such lists may have been written on paper then in word processor, spread sheet, but as

Scanned by CamScanner

the business and list grew, it becomes difficult to deal with the information stored in word processor or a spreadsheet, therefore Database technology is needed.

With the advent of W.W.W. and its use data being generated and stored by companies like amazon.com, eBay, google, Youtube, Wikipedia needs needs to be managed. On the other hand, consider the amount of data being generated by mobile phones in India. A lot of data must be stored for accounting purposes while other data like the content of various Sms messages must also be stored for retrieval in case of security checks or more serious criminal proceedings. Thus, amount of data being generated in the modern world using new technologies is mind bobbling. And therefore need of DBMS has emerged.

(*) There are some unique problems if not using database.

1) Size of data :-

The small amount of data storing into spreadsheet is fine, however it might turns into a large amount of data then spreadsheet solution will not work. Even if the size of data records goes into millions then storing data into multiple spreadsheets which will create problem of speed. It will take long time to find a record from the multiple spreadsheet files.

Scanned by CamScanner

PAGE No.
DATE / /

2) **Ease of updating data :-**

Multiple people cannot edit the same file on the same time. Other people must wait until files are available to update which results into wastage of time.

3) **Accuracy :-**

When user is doing data entry in files then it might be possible to have incorrect data due to no validation present like one can enter wrong spelling, wrong dates and wrong amount. So data accuracy is hard to maintain and accuracy is in question.

4) **Security :-**

You cannot secure data in the text files and spreadsheet. Anyone can access the file and read any data present in the file. So storing data will not work with banking, healthcare application, payroll department where privacy is difficult to maintain.

5) **Redundancy :-**

The duplication of data ~~cannot~~ can be possible using text files or spreadsheet. Chances of adding multiple copies of data cannot be limited here. This will leads to accuracy issues. Maintaining and updating multiple copies is not an easy task.

Scanned by CamScanner

PAGE No.
DATE / /

6) Incomplete data:-
Some of the data is not considered and not important, so such data is not entered in the files as no validation in place which leads the data integrity in question.

* Difference between File System and Database Management System:-

Sr.No.	Difference factor	file system	DBMS
1.	Definition	A file system is an abstraction to store retrieve, manage and update ~~to~~ a set of files.	DBMS is a collection of interrelated data and a set of programs to access those data. Some of the very well known DBMS are Microsoft access, Microsoft SQL server, SAP, Oracle, dBase.
2.	Data Redundancy	In file system approach, each user defines and implements the needed files for a specific applicat^n to run.	Although the database approach does not remove redundancy completely, it controls the amount of

Scanned by CamScanner

		for eg. in sales dept. of an enterprise, one user will be maintaining the details of how many sales personnel are there in the sales dept. and their grades. Another user will be maintaining the sales person details.	redundancy in DB approach, a single repository of data is maintained i.e. defined once and then accessed by many users. The fundamental characteristic of DB System not only contains data but also contains complete definition or description of the DB structure and constraints
3.	Sharing of data	File system doesn't allow sharing of data or data sharing is very complex.	In DBMS, data can be shared very easily due to centralized system.
4.	Data Consistency	When data is redundant, it is difficult to update. eg. If we want to change or	In DBMS, as there is no or less data redundancy, data remains

Scanned by CamScanner

		update ~~or~~ employee's address, then we have to make changes at all the places where data of that employee is stored. If by mistake, we forgot to change or update the address at one or more places then data inconsistency will occur i.e. the appearance of same data will differ from each other.	consistent.
5.	Difficult to search/ access data	In conventional file system, if we want to search/ retrieve/access some data item, it becomes very difficult because in the file system for every operation we have to make different programe	In DBMS, searching retrieval, accessing of data item is very easy and user friendly because searching and querying operations are already available in the system.

Scanned by CamScanner

6. Data Isolation	In file sys, there is no standard format of data or we can say data is scattered in various formats or files which also make data retrieval difficult.	In DBMS, due to centralized system the format of similar type of data remains same.
7. Data Integrity	The value of data in DB, must follow or satisfy some rules or consistency constraints. eg:- A company have a policy that the age of an employee must be >18. The value which is not satisfying this constraint must not be stored in the respective column. In file system there is no procedure to check these constraints automatically.	DBMS maintains the data integrity by enforcing the constraints by adding appropriate codes.

Scanned by CamScanner

8.	Security Problems	In file system, there is no or very less security. General security provided by the file systems are locks, guards etc.	DBMS have high level security like encryption, passwords, biometric (finger-print matching, face and voice detection etc.) security.
9.	Atomicity	Atomicity means a transaction must be all-or-nothing i.e. the transaction must either fully happen or not happen at all. It must not complete partially. eg. If A wants to transfer 5000Rs to B's account. In this case A's account should be debited and B's account should be credited with the same amount. Let suppose A's a/c is debited with 5000 Rs. and then transaction fails. Now the transaction is incomplete because	

Scanned by CamScanner

		B's account is not credited. These type of problems occur in file system because there is no procedure to stop such type of anomalies	Transaction Atomicity is a special feature of DBMS. In DBMS either a transaction completed fully or none of the action is performed. For this, DBMS maintains transaction log in which intermediate values are stored.
10.	Concurrent access Anomalies	Any multi-user DB application require a method for dealing with concurrent access to data when more than one user is accessing the same data at the same time. A problem occurs when user X reads a row ~~from~~ for editing, user Y reads the same row	DBMS along with an appropriate application provides safety towards concurrent access. For this locks are available in DBMS. If 2 or more transactions want to change/update

Scanned by CamScanner

		for editing. User Y saves changes then user X saves changes. The changes made by user Y are lost unless something prevents user X from blindly overwriting the new. File system does not provide any procedure to stop such type of anomalies.	or write a data item an exclusive lock is issued to one of these transactions. until and unless transaction release that lock no other transaction can acquire the lock and hence cannot update/write data item.

(*) Database Architecture:-

DB architecture defines the manner in which the components of DB are organized and integrated. DBMS can be centralized (All the data stored at one location), decentralized (multiple copies of DB at different locations) or hierarchical.

(*)(a) 1 tier architecture:-

In 1-tier architecture DB is directly available to the users to perform various operations. Any changes done here are directly done on the DBMS itself. Mainly DB designers

and programmers prefer to use single tier architecture.

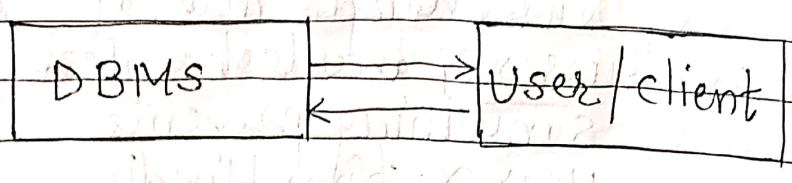

fig. 1 tier architecture.

(*)(6) 2-tier architecture :-

2 tier DBMS architecture includes an Application layer between the user and the DBMS, which is responsible to communicate the user's request to the database management system and then send the response from the DBMS to the user.

An application interface known as ODBC (Open Database Connectivity) provides an API that allow client side program to call the DBMS. Most DBMS vendors provide ODBC drivers for their DBMS.

fig. 2-tier architecture

Such an architecture provides the DBMS extra security as it not exposed to the end

User directly. Also, security can be improved by adding security and authentication checks in the application layer too.

(*) (c) 3-tier architecture :-

3-tier architecture is the most commonly used architecture for web applications. It is an extension of the 2 tier architecture. In the 2-tier architecture, we have an application layer which can be accessed programmatically to perform various operations on the DBMS. The application generally understands the Database Access Language and processes end users requests to the DBMS.

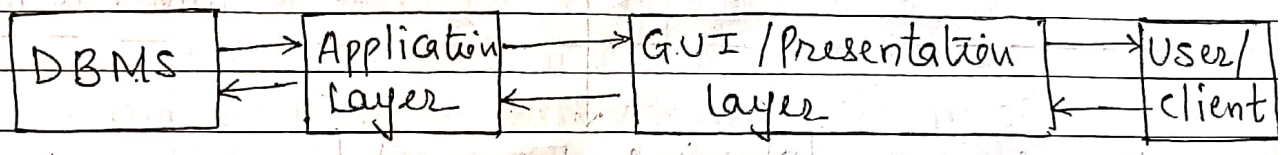

fig. 3-tier architecture

In 3-tier architecture, an additional Presention or GUI layer is added, which provides a graphical user Interface for the end user to interact with the DBMS.

For the end user, the GUI layer is the DB system, and the end user has no idea about the application layer and the DBMS system. PHPMyAdmin is the best example of a 3 tier DBMS.

Scanned by CamScanner

PAGE No.
DATE / /

(*) Three Level Architecture of DBMS OR Three Views of Data :-

The DB can also be viewed from different levels of abstraction to reveal different levels of details.

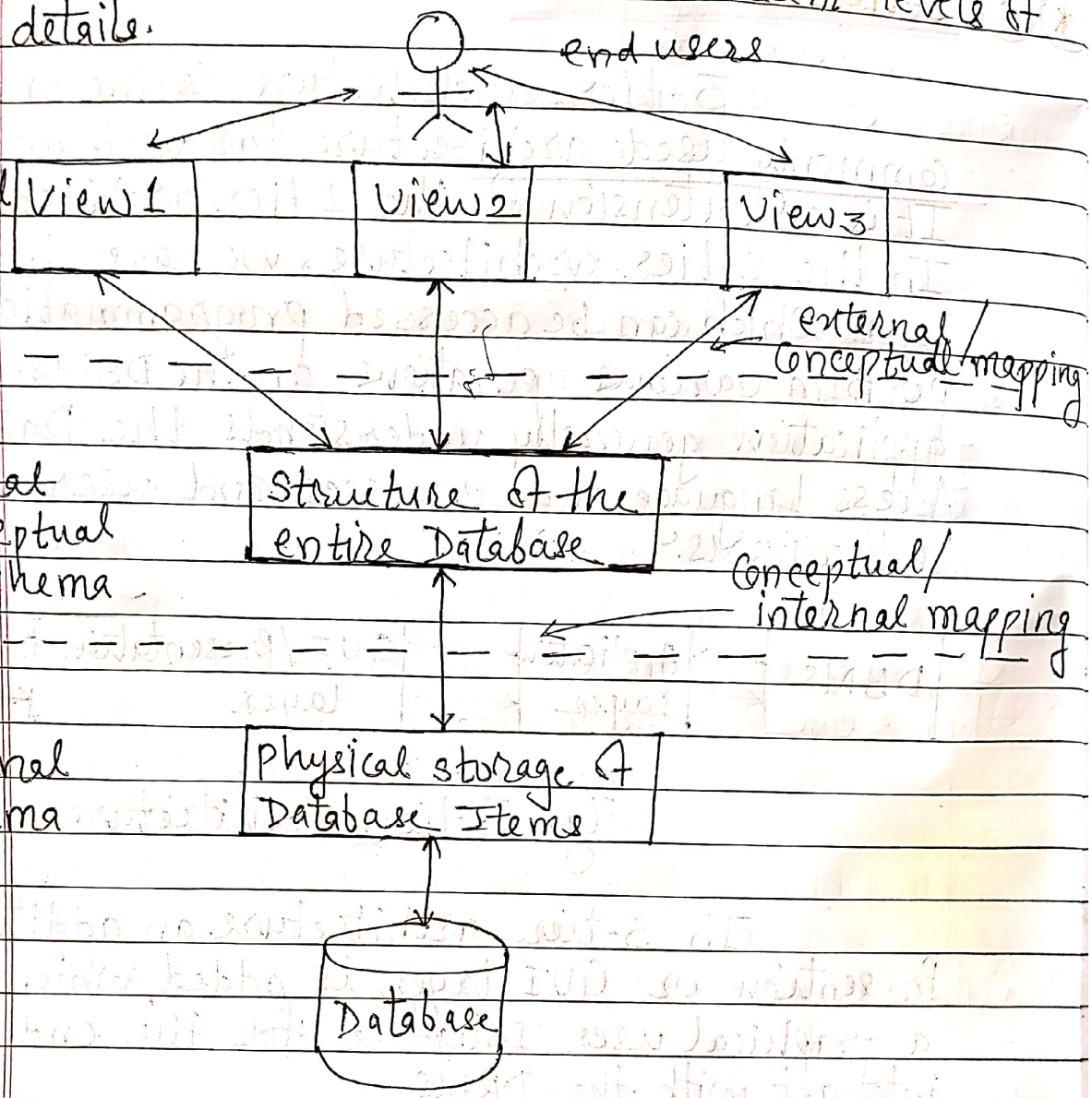

fig. 3-level Architecture of DBMS.

The word schema means arrangement - how we want to arrange things that we have to store. Mapping is the process of transforming

request and response between various database levels of architecture.

In external/conceptual schema, request is from external to conceptual schema, and in conceptual/internal mapping the request is from conceptual to internal levels.

- External/view schema :-

It is the view that individual user has. Each individual user has his/her own view of his/her enterprise database. This view may be restricted view of the database and the same database is likely to provide a number of different views for different classes of users.

eg. A department head in a university may only be interested in the departmental finances and student enrolments in his/her department but not in the buildings and grounds information. The librarian would not be expected to have any interest in the information about departmental finances. The staff payroll office may have no particular interest in student enrollments in computer science unless those numbers have an impact on staff salaries.

- Conceptual/Logical schema :-

It deals with the structure of the entire database. It is the community view of the database which contains the view of the whole enterprise without any concern for its physical implementation. This view is normally more stable than two. In this, we want to know informat

PAGE No.
DATE / /

about the attributes of each table, the common attributes in different tables that help them to be combined, what kind of data can be input these attributes and so on. It represents entity, attributes and their relationships. Conceptual or logical schema is very useful for database administrators whose responsibility is to maintain the entire database.

- **Internal / Physical schema :-**

It is the physical representation of database on the computer. It describes how data is physically stored in the database. It also describes the data type of these data items, the size of the items in the storage media, the location (physical address) of the items in the storage device and so on. This schema is useful for database application developers and database administrators.

* Data Independence :-

Data Independence is ability to modify a schema definition in one level without affecting a schema definition in the next higher level.

A DBMS may change the structure of data without having to change the application programs. (eg. Accounting systems whose primary purpose is to enter and retrieve information from a computerized

Scanned by CamScanner

PAGE No.
DATE / /

database). This is not possible in conventional file processing systems which will have a very low level of data independence as the file structure and the application programs are tightly coupled together.

* Logical Data Independence :-

It is data about database i.e. it stores information about how data is managed inside. for eg. Table stored in the DB and all its constraints applied on the relation (table).

Logical data independence is a kind of mechanism which changes itself from actual data stored on the disk. It refers to the capacity of changing the conceptual schema without any changes to the external schema. eg. Adding a new column in a table without affecting the external views.

* Physical Data Independence :-

Physical data independence is the power to change the physical data without impacting the logical schema.

eg.- Consider a case where storage device on which the DB files are stored is changed. A user writes a query to retrieve some data. for example, select fname, lname, age from records;

The above query states that tables name which contains the data, not the location of the file where database is stored

Scanned by CamScanner

PAGE No.
DATE / /

So therefore even when internal schema is changed, it will not change the conceptual schema. The result of query will remain same whether the DB storage device is changed or not.

* Characteristics of database :-

1) Self describing nature of DB and metadata :-
DB system not only contains DB itself but also the descriptions of data structures and its constraints. This is stored in system catalog or data dictionary. The info. stored in system catalog is called metadata (data about data)

eg:- Data dictionary in oracle stores info. about the users in DB, the tables that they have created, the files in which data is stored and security permissions.

Some examples of the contents of the data dictionary are :-

i) What data is available?
ii) Where data is stored?
iii) Who owns the data?
iv) How the data is issued?
v) Who can access the data?
vi) What relationship exists between data items?

2) Data Independence :-
The process of hiding irrelevant details from the user is called

Scanned by CamScanner

PAGE No.
DATE / /

data abstraction.

eg:- We are storing customer information in customer table. At physical level these records can be described as blocks of storage (bytes, gigabytes) ~~described as blocks of storage~~ in memory. The details are often handled by the programmers.

At the logical level, these records can be described as fields and attributes along with their data types, their relation among each other can be logically implemented.

At view level, user just interact with the system with the help of GUI and enter the details on the server. They are not aware of how data is stored and what data is stored. Such details are hidden from them.

3) Multiple views of data and Security :-

A DB typically has many users. Each of them may require a different perspective or view of the DB. A view may be subset of database, ~~A view may be subset of~~ or it may contain virtual data that is derived from the Database files but is not explicitly stored.

In modern DB systems restrictions are placed on the content the users can access unless the user is not authorized user to access some data.

4) Persistent Data :-

A DB only stores persistent data

Scanned by CamScanner

and once data has been stored in DB system. It always stays there unless deleted by an explicit request.

(*) Users involved in DB systems :-

On the basis of interaction with the system, users are differentiated to following types:

1) Database Administrator (DBA) :-

Usually a person (or a group of persons) centrally located with an overall view of the database is needed to keep DB running smoothly such a person is often called the Database Administrator (DBA). DBA is responsible for managing software and hardware resources. In a large enterprise DBA must be assisted by an advisory committee which includes members from the different units of the company.

DBA performs the following tasks :-

a) Developing Conceptual Schema :-

DBA in consultation with senior management of the enterprise is responsible for defining conceptual schema (It is used for defining data structure, entities and relationships used for DB. It also describes semantics (logic) for an organization.

Scanned by CamScanner

PAGE No.
DATE / /

b) deciding which DBMS to use :-
Once the conceptual schema has been decided, a decision regarding which DBMS software to be used is made.

c) Defining the DB and loading DB content :-
Once the DBMS has been acquired and installed the DBA would be responsible for implementing conceptual schema by using DDL.

d) Assisting and approving application and access :-
DBA would normally provide assistance to end users as well as programmers in writing application programs to assess the DB.

e) Deciding Data Structure :-
Once DB contents have been decided, the DBA would normally take decisions regarding manner in which data is to be stored. DBA monitors the performance and makes changes to the data structure for the performance.

f) Backup and recovery :-
DB is valuable asset and therefore DBA must make all possible efforts to ensure that the database is not damaged or lost. Therefore regular recovery and backups are carried out to check the

Scanned by CamScanner

failures. If any data entry error or hardware failure occurs DBA can bring the DB backward in time to its state before the damage was done.

g) Integrity :-

Integrity means accuracy and consistency of data.

for eg :- Retailer may have business rule that individual customer can place order. so every order must identify one and only one customer. These constraints are enforced and checked by the DBA.

h) Security :-

Only authorized users can assess and change the policies of the business. This privilege is provided by the DBA.

2) Database Designers :-

DB designers are responsible for identifying data to be stored in DB and choosing the appropriate structures to represent and store data. It is responsibility of DB designers to communicate with all DB users in order to understand their requirements and to come up with designs that meets these requirements.

Scanned by CamScanner

PAGE No.
DATE / /

3) End Users :-

End users are responsible for accessing the DB for querying, updating, generating reports.

a) Casual End Users :-

They occasionally access the DB but they need different information each time.

b) Naive or Parametric end users :-

Their job is to constantly querying and updating DB. eg. Railway reservation clerks, persons in ATM services.

c) Sophisticated end users :-

Sophisticated end users include engineers, scientists, business analyst who themselves learn the facilities of DBMS to implement their application.

d) Stand-alone users :-

They maintain personal DB by using ready-made packages.
eg. User of tax package maintains personal financial data for tax purpose.

4) Security Officers :-

The hackers who try to break into corporate DB and steal sensitive information So, in todays environment the computer resources

Scanned by CamScanner

and databases need specialized professionals whose sole responsibility is to protect data from unauthorized access.

(*) Database System structure :-

A DB is partitioned into modules that deal with each of the responsibilities of the overall system. The functional components of a DB system can be broadly divided into storage Manager and query processor component.

1) Storage Manager :-

Storage Manager is important because DB typically require a large amount of storage space. Since main memory of computers cannot store much of this info., the info. is stored on disks. Data are moved between disk storage & main memory as needed.

Storage Manager is responsible for providing interface between the low level data stored in the DB and application programs and queries submitted to the system.
It includes following components :-

a) Authorization and Integrity Manager :-

It tests for the satisfaction of integrity constraints and checks the authority of the users.

Scanned by CamScanner

PAGE No.
DATE / /

b) Transaction Manager :-
It ensures that DB remains in consistent state despite of system failure.

c) File Manager :-
It manages the allocation of space on disk storage and the data structures used to represent info. stored on disk.

Data Structures :-
(i) Data-files :- stores data itself.
(ii) Data Dictionary :- stores ~~Medat~~ Metadata about structures of the DB, in particular the schema of the DB.
(iii) Indices :- It provides fast access to data items that hold particular values.

d) Buffer Manager :-
It is responsible for fetching data from disk storage to Main Memory & deciding which data to cache in main memory.

2) Query Processor :-
It includes following components

a) DDL interpreter :-
It interprets DDL statements & records the definitions in the data dictionary

Scanned by CamScanner

PAGE No.
DATE / /

b) DML Compiler :-

It translates DML statements in a query language into an evaluation plan consisting of low level instructions that the query evaluation engine understands.

A query can usually be translated into any number of alternative evaluation plans that all give same result. The DML compiler also performs query optimization i.e. it picks the lowest evaluation plan from among the alternatives.

c) Query evaluation engine

It executes low level instructions generated by the DML compiler.

Scanned by CamScanner

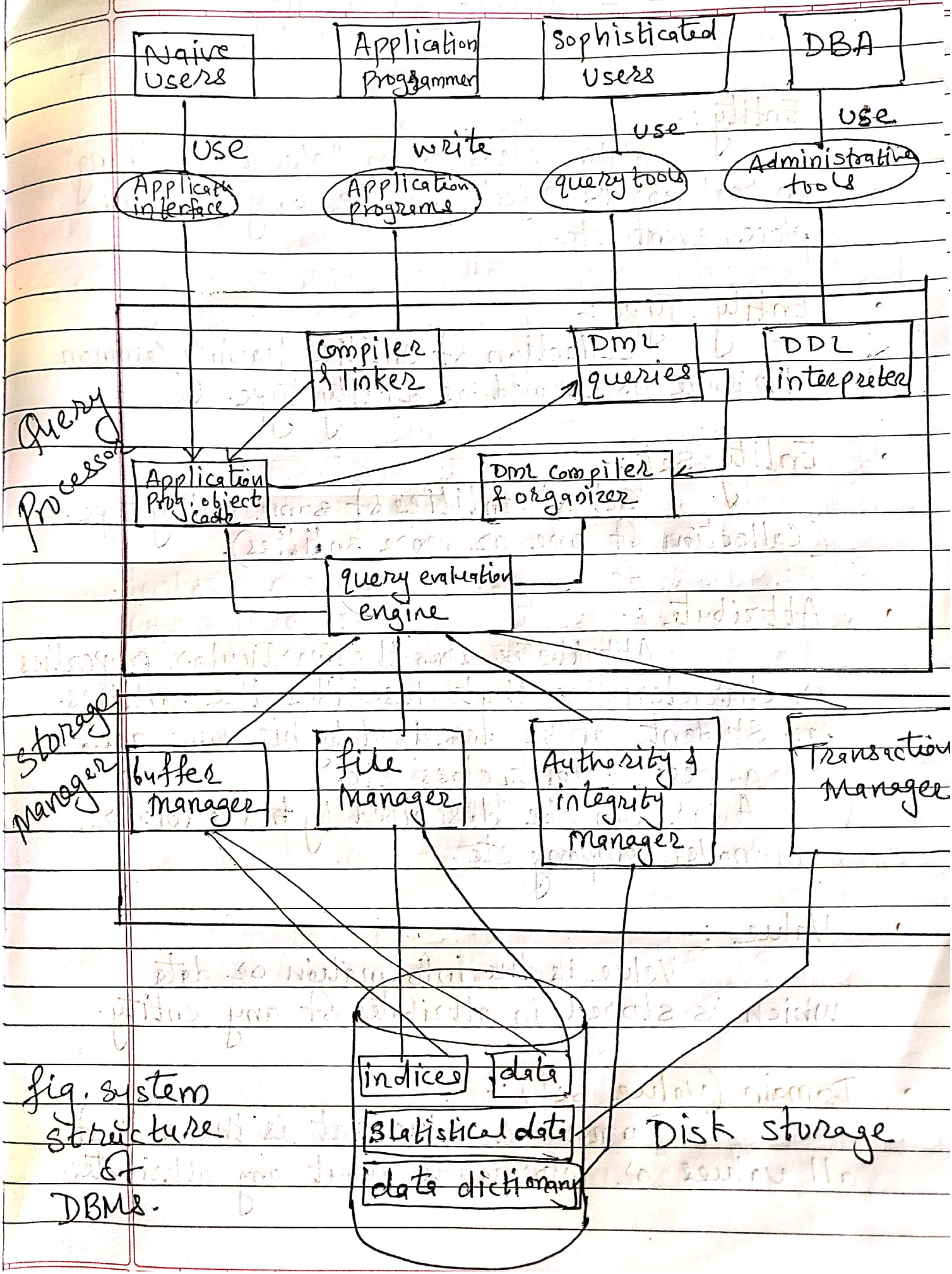

fig. system structure of DBMS.

Scanned by CamScanner