UNIT-1

File Processing System.

FPS-Definition:

A file is a named collection of data.
Ex:- a.txt,a.doc,a.pptx,a.c,a.exe

- ★ File processing system (FPS) is a collection of files and set of Application programs called software written in c like host language.

- ★ File processing system facilitates management of files i.e. creation, modification and deletion of files.

- ★ And the File processing system facilitates manipulation of data in the files i.e. insertion, update and deletion of data.

- ★ Before the advent of Database management systems FPS was used.

Disadvantages of FPS:

The following are disadvantages of the file process system.

1. Data Mapping and Access:

- ➢ Although all the related information is grouped and stored in different files, there is no mapping between any two files. i.e.; any two dependent files are not linked.

- ➢ Even though Student files and Student_Report files
  are related, they are two different files and they are not linked
  by any means.

2.Data Redundancy:

- ➔ Data redundancy refers to the same data located in multiple places in the database.
- ➔ It also makes data retrieving less efficient.
- ➔ Furthermore, data redundancy consumes more resources in the database.
- ➔ Over time, data redundancy makes database corruption, causing the data to be unusable.

3.Data inconsistency:

- ➔ Data inconsistency refers to a situation of keeping the same data in different formats in two different tables or a situation where it requires matching the data between tables.

4.Data Dependence:

- ➔ In the files, data are stored in a specific format, say tab, comma, or semicolon.
- ➔ If the format of any of the files is changed, then the program for processing this file needs to be changed.

5.Data Isolation:

- ➔ Since all the data is isolated from each other in different files, programming becomes Difficult.

6.Security:

- ➔ Each file can be password protected.This is very difficult in the file system.

7.Integrity:

- ➔ If we need to check for certain insertion criteria while entering the data into the file it is not possible directly.
- ➔ There is no direct checking facility in the file system.
- ➔ Hence these kinds of integrity checks are not easy in the file systems.

8.Atomicity:

- ➔ Atomicity means either all or nothing i.e it must successfully get completed or come back to the previous state.
- ➔ This atomicity is there in FPs

9.Concurrent Access:

- ➔ Accessing the same data from the same the file is called concurrent access.
- ➔ In the file system, concurrent access leads to incorrect data.

Advantages of Database Management System (DBMS)

1. Improved data sharing:

- ➔ An advantage of the database management approach is, the DBMS helps to create an environment in which end users have better access to more and better-managed data.

2. Improved data security:

- ➔ The more users access the data, the greater the risks of data security.
- ➔ A DBMS provides a framework for better enforcement of data privacy and security policies.

3. Better data integration:

- ➔ The term data integrity refers to the accuracy and consistency of data.
- ➔ When creating databases, attention needs to be given to data integrity and how to maintain it.
- ➔ A good database will enforce data integrity whenever possible.

4. Minimized data redundancy

5. Data Independence and Consistency:

- ➔ Since DBMS defines each column and rows at the beginning itself and controls the way data is entered, there is no effect on the programs or any other tables or data.
- ➔ Hence the consistency of the data also maintained.

6. Concurrent Access:

- ➔ DBMS provides access to multiple users to access the database at the same time.
- ➔ It has its own concurrency control mechanisms to have concurrency accesses and hence avoid any incorrect data in the system.

Levels of abstraction

In order to make the system efficient in terms of retrieval of data, and reduce complexity in terms of usability of users, developers use abstraction i.e. hide irrelevant details from the users.

This process of hiding irrelevant details from the user is called data abstraction.

There are mainly three levels of data abstraction:

1.Physical level:

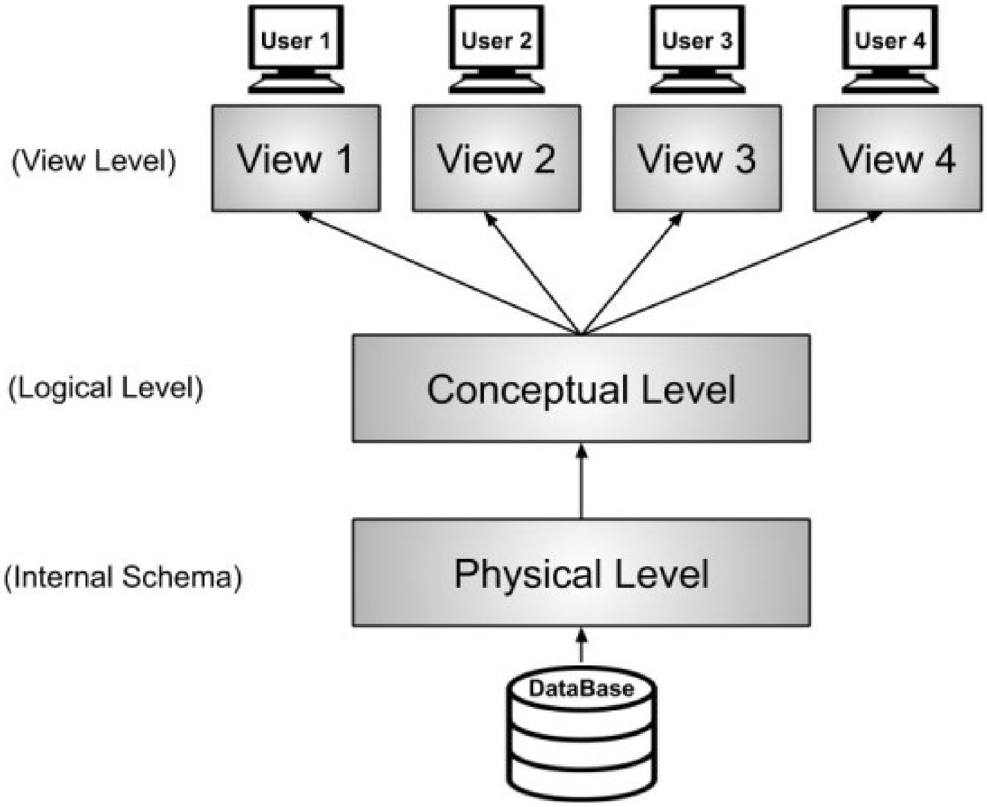
- ★ This is the lowest level of data abstraction.
- ★ It tells us how the data is actually stored in memory.
- ★ The complex data structure details can be obtained at this level.

2.Logical level:

- ★ This is the middle level of data abstraction.
- ★ It comprises of the what data is actually stored in the database in the form of tables.
- ★ It also stores the relationship among the data entities in relatively simple structures.

3.View level:

- ★ This is the highest level of abstraction.
- ★ Only a part of the actual database is viewed by the users.
- ★ This level exists to ease the accessibility of the database by an individual user.
- ★ Users view data in the form of rows and columns.
- ★ Tables and relations are used to store data.
- ★ Multiple views of the same database may exist.



Levels of Data Abstraction

Data Independence:

The main purpose of data abstraction is achieving data independence
in order to save time and cost required when the database is
modified or altered.

Data Independence is defined as a property of DBMS that helps you to change the Database schema at one level of a database system without requiring to change the schema at the next higher level.

We have namely two levels of data independence arising from these
levels of abstraction

Physical level data independence :

1. It refers to the characteristic of being able to modify the physical schema without any alterations to the conceptual or logical schema, done for optimisation purposes
2. E.g., Conceptual structure of the database would not be affected by any change in storage size of the database system server.
3. Changing from sequential to random access files is one such example.
4. These alterations or modifications to the physical structure may include:
    - Utilising new storage devices.
    - Modifying data structures used for storage.
    - Altering indexes or using alternative file organisation techniques etc.

Logical level data independence:

1. It refers characteristic of being able to modify the logical schema without affecting the external schema or application program.
2. The user view of the data would not be affected by any changes to the conceptual view of the data.
3.These changes may include insertion or deletion of attributes, altering table structures entities or relationships to the logical schema etc.

DIFFERENT TYPES OF USERS IN DBMS

There are four types of users in DBMS based on their usage.
They are:
1.Database Implementers
2.End Users
3.Application Programmers
4.Data Base Administrators (DBA)

1.Database Implementers:
The database implementers build DBMS software and they are particularly work for vendors such as IBM or Oracle.

2.End Users:
End users are the users who wish to store and use data in a DBMS.
End users may come from different fields.
Many end users simply use applications written by database application programmers and so require little technical knowledge about DBMS software.
Some end users make more extensive use of a DBMS, such as writing their own queries.
They require a deeper understanding of its features.

3.Application Programmers:
The Database application programmers develop packages that provides data access for end users.
For package development the application programmers use the host or data languages and software tools that DBMS vendors provide.

4.Database Administrators (DBA):
Database Administrator (DBA) is a professional user of DBMS.
The DBA is authorized for designing and maintaining the complex and important databases related to corporate and enterprise sectors.

DUTIES OF DBA:

1.Design of the Conceptual and Physical Schemas:

The DBA interacts with the different types of users in DBMS to understand what data is to be stored in the DBMS and how it is to be used.

Based on this knowledge, the DBA must design the conceptual schema and the physical schema.

The DBA may also design widely used portions of the external schema.

2.Security and Authorization:

The DBA is responsible for ensuring that unauthorized data access is not permitted. In a relational DBMS, users can be granted permission to access only certain views and relations.

3.Data Availability and Recovery from Failures:

The DBA must take steps to ensure that if the system fails, users can continue to access as much of the uncorrupted data as possible.

The DBA must also work to restore the data to a consistent state.

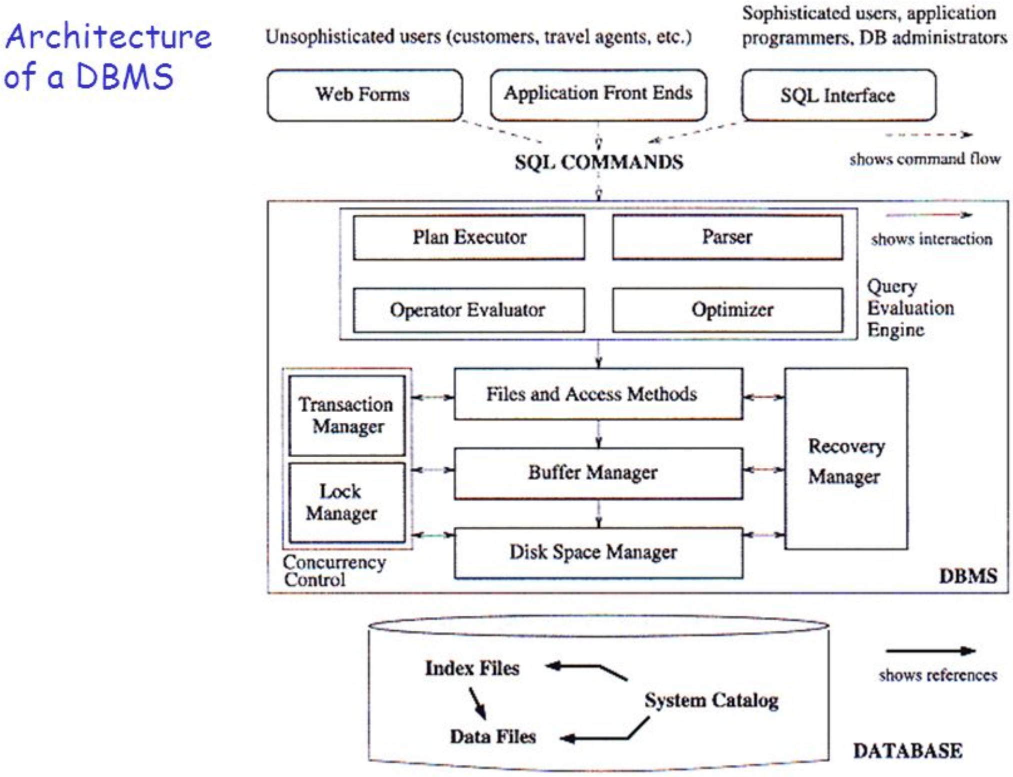
The DBMS provides software support for these functions.

The DBA is also responsible for implementing procedures to back up the data periodically and maintain logs of system activity to facilitate recovery from a crash.

4.Database Tuning:

The DBA is responsible for modifying the database, in particular the conceptual and physical schemas, to ensure adequate performance as requirements change.

STRUCTURE OF DBMS



This diagram represents the structure of the DBMS.
The DBMS accepts SQL commands generated from a variety of user interfaces such as web forms, Applications and SQL Interface.

Query optimizer produce an efficient execution plan for evaluating the query.

An execution plan is a blueprint for evaluating a query, usually represented as a tree of relational operators.

The file and access methods layer supports the concept of a file and indexes.

The buffer manager brings pages in from disk to main memory in response to read requests.

The disk space manager is the lowest layer of the DBMS deals with management of space on disk.

The transaction manager is responsible for executing the transactions in the DBMS environment.

During execution of transactions they request and release locks according to a suitable locking protocol.

The lock manager keeps track of requests for locks and grants locks on database objects when they become available.

The recovery manager is responsible for maintaining a log and restoring the system to a consistent state after a crash.

The disk space manager, buffer manager, and file and access method layers must interact with the transaction manager, lock manager and recovery manager.