**Data Base Management System**

A Data Base Management System is a system software for easy, efficient and reliable data processing and management. It can be used for:

Creation of a database.

Retrieval of information from the database.

Updating the database.

Managing a database.

It provides us with the many functionalities and is more advantageous than the traditional file system in many ways listed below:

1) Processing Queries and Object Management:

In traditional file systems, we cannot store data in the form of objects. In practical-world applications, data is stored in objects and not files. So in a file system, some application software maps the data stored in files to objects so that can be used further.

We can directly store data in the form of objects in a database management system. Application level code needs to be written to handle, store and scan through the data in a file system whereas a DBMS gives us the ability to query the database.

2) Controlling redundancy and inconsistency:

Redundancy refers to repeated instances of the same data. A database system provides redundancy control whereas in a file system, same data may be stored multiple times. For example, if a student is studying two different educational programs in the same college, say ,Engineering and History, then his information such as the phone number and address may be stored multiple times, once in Engineering dept and the other in History dept. Therefore, it increases time taken to access and store data. This may also lead to inconsistent data states in both places. A DBMS uses data normalization to avoid redundancy and duplicates.

3) Efficient memory management and indexing:

DBMS makes complex memory management easy to handle. In file systems, files are indexed in place of objects so query operations require entire file scans whereas in a DBMS , object indexing takes place efficiently through database schema based on any attribute of the data or a data-property. This helps in fast retrieval of data based on the indexed attribute.

4) Concurrency control and transaction management:

Several applications allow user to simultaneously access data. This may lead to inconsistency in data in case files are used. Consider two withdrawal transactions X and Y in which an amount of 100 and 200 is withdrawn from an account A initially containing 1000. Now since these transactions are taking place simultaneously, different transactions may update the account differently. X reads 1000, debits 100, updates the account A to 900, whereas Y also reads 1000, debits 200, updates A to 800. In both cases account A has wrong information. This results in data inconsistency. A DBMS provides mechanisms to deal with this kind of data inconsistency while allowing users to access data concurrently. A DBMS implements ACID(atomicity, durability, isolation,consistency) properties to ensure efficient transaction management without data corruption.

5) Access Control and ease in accessing data:

A DBMS can grant access to various users and determine which part and how much of the data can they access from the database thus removing redundancy. Otherwise in file system, separate files have to be created for each user containing the amount of data that they can access. Moreover, if a user has to extract specific data, then he needs a code/application to process that task in case of file system, e.g. Suppose a manager needs a list of all employees having salary greater than X. Then we need to write business logic for the same in case data is stored in files. In case of DBMS, it provides easy access of data through queries, (e.g., SELECT queries) and whole logic need not be rewritten. Users can specify exactly what they want to extract out of the data.

6) Integrity constraints

Data stored in databases must satisfy integrity constraints. For example, Consider a database schema consisting of the various educational programs offered by a university such as(B.Tech/M.Tech/B.Sc/M.Sc/BCA/MCA) etc. Then we have a schema of students enrolled in these programs. A DBMS ensures that it is only out of one of the programs offered schema , that the student is enrolled in, i.e. Not anything out of the blue. Hence, database integrity is preserved.

Apart from the above mentioned features a database management also provides the following:

Multiple User Interface

Data scalability, expandability and flexibility: We can change schema of the database, all schema will be updated according to it.

Overall the time for developing an application is reduced.

Security: Simplifies data storage as it is possible to assign security permissions allowing restricted access to data.

