UNIT 1

Introduction to Database Management System

Data and Information

- Data is meaningful known raw facts that can be processed and stored as information.
- For e.g.
  - Ram
  - Korth
  - Samsung

Database

- Database is a collection of interrelated and organized data.
- For e.g.
  - STUDENT
  - BOOK
  - LIBRARY

DBMS

- Database Management System (DBMS) is a collection of interrelated data [usually called database] and a set of programs to access, update and manage those data [which form part of management system].
- It is a software package to facilitate creation and maintenance of computerized database. It is general purpose software that facilitates the following:
  - Defining: Specifying data types and structures, and constraints for data to be stored.
  - Constructing: Storing data in a storage medium.
  - Manipulating: Involves querying, updating and generating reports.
  - Sharing: Allowing multiple users and programs to access data simultaneously.

Examples Of DBMS

- MS Access
- dBase
- FileMaker Pro
- FoxBASE
- ORACLE
- Ingress
- Informix
- MySQL
- MS SQL Server
- IBM DB2
- PostgreSQL

Primary goals of DBMS

- To provide a way to store and retrieve database information that is both convenient and efficient.
- To manage large and small bodies of information. It involves defining structures for storage of information and providing mechanism for manipulation of information.
- It should ensure safety of information stored, despite system crashes or attempts at unauthorized access.
- If data are to be shared among several users, then system should avoid possible anomalous results.

Disadvantages of File Processing System

- Data Redundancy — Since different programmers create the files and application programs over a long period, the various files are likely to have different formats and the programs may be written in several programming languages. Moreover, the same information may be duplicated in several files, this duplication of data over several files is known as data redundancy.
- Eg. The address and telephone number of a particular customer may appear in a file that consists of saving-account records and in a file that consists of checking-account records.

Disadvantages of File Processing System

- Data Inconsistency — The various copies of same data may no longer agree i.e. various copies of the same data may contain different information.
- Eg. A changed customer address may be reflected in savings-account records but not elsewhere in the system.

Disadvantages of File Processing System

- Difficulty in accessing data — In a conventional file processing system it is difficult to access the data in a specific manner and it is require creating an application program to carry out each new task.
- E.g. Suppose that one of the bank officers needs to find out the names of all customers who live within a particular postal-code area. The officer asks the data-processing department to generate such a list. Because the designer of the original system did not anticipate this request, there is no application program on hand to meet it.

Disadvantages of File Processing System

- Data Isolation — Because data are scattered in various files, and files may be in different formats, writing new application programs to retrieve the appropriate data is difficult.

Disadvantages of File Processing System

- Integrity problems - The data values stored in the database must satisfy certain types of consistency constraints.
- E.g. The balance of a bank account may never fall below a prescribed amount (say $25).
- Developers enforce these constraints in the system by adding appropriate code in the various application programs. However, when new constraints are added, it is difficult to change the programs to enforce them. The problem is compounded when constraints involve several data items from different files.

Disadvantages of File Processing System

- Atomicity problems - A computer system, like any other device, is subject to failure. In many applications, it is crucial that, if a failure occurs, the data be restored to the consistent state that existed prior to the failure.
- E.g. Consider a banking system with a program to transfer $500 from account *A* to account *B*. If a system failure occurs during the execution of the program, it is possible that the $500 was removed from the balance of account *A* but was not credited to the balance of account *B*, resulting in an inconsistent database state. Clearly, it is essential to database consistency that either both the credit and debit occur, or that neither occur. That is, the funds transfer must be *atomic*—it must happen in its entirety or not at all.
- It is difficult to ensure atomicity in a conventional file-processing system.

Atomicity Problems (Consistent State)

A	B	A+B
1000	2000	3000

A = A + 100	B = B - 100
B = B - 100	A = A + 100

A	B	A+B
1100	1900	3000

Atomicity Problems (In - consistent State)

A	B	A+B
1000	2000	3000

- A = A + 100
- B = B - 100

A	B	A+B
1100	2000	3100

Atomicity Problems (In - consistent State)

A	B	A+B
1000	2000	3000

- B = B - 100
- A = A + 100

A	B	A+B
1000	1900	2900

Disadvantages of File Processing System

- Concurrent-access anomalies - For the sake of overall performance of the system and faster response, many systems allow multiple users to update the data simultaneously. In such an environment, interaction of concurrent updates is possible and may result in inconsistent data.
- Consider account *A*, with a balance of $10,000. If two bank clerks debit the account balance (by say $500 and $100, respectively) of account *A* at almost exactly the same time, the result of the concurrent executions may leave the account balance in an incorrect (or inconsistent) state. Suppose that the programs executing on behalf of each withdrawal read the old balance, reduce that value by the amount being withdrawn, and write the result back. If the two programs run concurrently, they may both read the value $10,000, and write back $9500 and $9900, respectively. Depending on which one writes the value last, the balance of account *A* may contain either $9500 or $9900, rather than the correct value of $9400.
- To guard against this possibility, the system must maintain some form of supervision. But supervision is difficult to provide because data may be accessed by many different application programs that have not been coordinated previously.

Concurrent Access Anamalies

• A	B	(Joint Accounts)
• 1000	1000	(Balance – 1000)
• A = A - 1000	B = B - 1000	

Disadvantages of File Processing System

- Security problems - Not every user of the database system should be able to access all the data.
- For example, in a university, payroll personnel need to see only that part of the database that has financial information. They do not need access to information about academic records. But since application programs are added to the file-processing system in an ad hoc manner, enforcing such security constraints is difficult.

For Video lecture on this topic please subscribe to my youtube channel.

The link for my youtube channel is

https://www.youtube.com/channel/UCRWGtE76JlTp1iim6aOTRuw?sub_confirmation=1