Material – 1

Data : Collection of raw facts

Information : Processed data

Database : Collection of raw facts of one particular organization

DataBase Management Systems : The ordering and operations performed on a collection of data of one particular organization through some application programs is known as DataBase Management Systems

Or

It is a collection of interrelated data and a set of programs to access those data.

Goals of DBMS:

1. To store data
2. To retrieve data

Drawbacks of File System (File System Vs DBMS) :

1. **Data Redundancy & Data Inconsistency:** Redundancy means repetition. In file system data is redundant i.e. data is repeated. Redundant data results in high memory cost. For example address of a person in a college database is repeated for a number of times in various files. If redundant data is modified in one file and not modified in another file results in another problem in file system i.e. data inconsistency. For example if address of a person is modified in one file and not modified in another file results in inconsistent address. This problem is eliminated by normalization in DBMS.
2. **Difficulty in accessing data :** For every retrieval in file system an application program must be written or an existing program must be modified by the programmer. For example a client needs details of students who obtained first class. Application programmer writes an application to meet the requirement. This application may not meet the needs of client at all the times. Later the same client is in need to have the details of first class students who are resident of a particular place. Either the application programmer writes a new application or modifies an existing program to meet the need. Writing new applications or modifying existing applications makes difficult in accessing data in file system. In DBMS simple queries are written for retrieval with less effort and more effective.
3. **Data Isolation :** Data in file system is scattered in various files in different formats makes difficult for retrieving data. In DBMS data is not scattered.
4. **Integrity Problems :** Data values in database must satisfy certain constraints. For example roll number of a student is to be represented in a particular format, to have this constraint in file system; various application programs retrieving information from the database must be included with the constraint. Constraints in DBMS are included with less effort and more effective.
5. **Atomicity Problems:** Any system may subject to failure. Despite of any failures data lost during failures must be restored to the consistent state that existed prior to the failure. For example account A has 100/- and account B has 200/-, together there is 300/-. Consider a transaction of transferring an amount of 50/- from account B to account A. 50/- was debited from account B followed by a system failure. There was no amount credited into account A, together there is 250/- but not 300/-. Now the database is in inconsistent state. Either debit and credit operations must be done or none of the operations must be done to maintain consistency (called atomicity). Recovery policies like Log based recovery are used in DBMS to recover the lost data.
6. **Concurrent access anomalies:** Accessing a data item by different transactions at the same time may result in inconsistent data. For example account A has 200/-. If two transactions want to debit money at the same time, result of concurrent executions may leave the database into inconsistent state. Consider one transaction debits 100/- and the other 50/-. If both transactions are executed at the same time, both

transactions reads account A with 200/-, depending upon which transaction writes the last either 100/- or 150/- is written into database, but account A must be written with 50/- after two debits. To overcome this problem in file systems leads to complex programming whereas it is easy in DBMS by using different concurrency control mechanisms.

7. **Security Problems:** Security levels in files system is less when compared with DBMS. For example in a banking system using file system account holder details and other details are known by most of the employees in bank when a transaction of a customer is executed.

View of Data / Data Abstraction

Physical Level:

- Describes HOW data are actually stored
- It deals with the data structures used in physical
- It is otherwise called as Internal level

Logical Level:

- Describes WHAT data is stored in database and their relationships among data
- This level is usually designed by using ER diagrams
- It is otherwise called as Conceptual Level, Schema level

External Level:

- Describes only part of the database i.e. only part of the database can be viewed at a time but not the entire database
- Database has many views
- It is otherwise called as Subschema level, View level

Instance:

The collection of data stored in database at a particular moment or time is called Instance

Schema:

The overall description of a database is called as Schema

Data Models:

Data model is a collection of conceptual tools for describing data, data relationships, data semantics and consistency constraints.

Or

A data model depicts the dataflow and logical interrelationships among different data elements.

- ✓ *A data model has 3 uses in an application which are getting data in, integrating data and getting data out.*

Data models are classified into two categories. They are:

1. Object based logical models
2. Record based logical models

1. **Object Based Logical Models:**
   - There is no fixed format for representing data
   - These models consists of conceptual tools
   - These are used for designing the logical/ conceptual level of a database
   - Otherwise called as semantic models
   - For example: ER Model. ER diagrams are used for designing logical level of a database
2. **Record Based Logical Models:**
   - There is fixed format for representing data
   - Data Models in this category are: *Network Model, Hierarchical Model & Relational Model*
   - Network Model uses the data structure Graph for representing data

- Hierarchical Model uses the data structure Tree for representing the data
- Relational Model uses table (set of rows & columns) for representing data
- Both network and hierarchical models are involved with more number of pointers. A single misplaced pointer corrupts the entire database. Relational Model is free from pointers

Relational Model:

- The relational model uses a collection of tables to represent both data and the relationships among those data
- Relational Model uses table (set of rows & columns) for representing data
- Table consists of two parts : *head & body*
- Head consists of column names. Column names are unique
- Body consists of tuples. A row in a table is called tuple. A tuple is a combination of different column values of one real world object.
- Number of columns represents *degree* of the table
- Number of tuples represents *cardinatily* of the table
- Tuple is identified uniquely by using primary key

Student

	Sid	**Sname**	**City**
Head			
Body	S1	Sree	Guntur
	S2	Hari	Tenali
	S3	Varun	Tenali
	S4	Varun	Tenali

Note: Name of the table is student. Degree of the table is 3 and cardinality is 4. In a table if a column (set of columns) is underlined with double line then it is called primary key of the table.

Properties of Relation:

1. All values in a table are atomic
2. Rows are unordered from top to bottom
3. Columns are unordered from left to right
4. Relation must contain primary key for identifying rows

Note: The term “Relation” is abstract and “Table” is concrete