OVERVIEW OF DBMS

Database is a collection of related data and data is a collection of facts and figures that can be processed to produce information.

Mostly data represents recordable facts. Data aids in producing information, which is based on facts. For example, if we have data about marks obtained by all students, we can then conclude about toppers and average marks.

A database management system stores data in such a way that it becomes easier to retrieve, manipulate, and produce information.

Characteristics

Traditionally, data was organized in file formats. DBMS was a new concept then, and all the research was done to make it overcome the deficiencies in traditional style of data management. A modern DBMS has the following characteristics −

Real-world entity − A modern DBMS is more realistic and uses real-world entities to design its architecture. It uses the behavior and attributes too. For example, a school database may use students as an entity and their age as an attribute.

Relation-based tables − DBMS allows entities and relations among them to form tables. A user can understand the architecture of a database just by looking at the table names.

Isolation of data and application − A database system is entirely different than its data. A database is an active entity, whereas data is said to be passive, on which the database works and organizes. DBMS also stores metadata, which is data about data, to ease its own process.

Less redundancy − DBMS follows the rules of normalization, which splits a relation when any of its attributes is having redundancy in values. Normalization is a mathematically rich and scientific process that reduces data redundancy.

Consistency − Consistency is a state where every relation in a database remains consistent. There exist methods and techniques, which can detect attempt of leaving database in inconsistent state. A DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems.

Query Language − DBMS is equipped with query language, which makes it more efficient to retrieve and manipulate data. A user can apply as many and as different filtering options as required to retrieve a set of data. Traditionally it was not possible where file-processing system was used.

ACID Properties − DBMS follows the concepts of Atomicity, Consistency, Isolation, and Durability (normally shortened as ACID). These concepts are applied on transactions, which manipulate data in a database. ACID properties help the database stay healthy in multi-transactional environments and in case of failure.

Multiuser and Concurrent Access − DBMS supports multi-user environment and allows them to access and manipulate data in parallel. Though there are restrictions on transactions when users attempt to handle the same data item, but users are always unaware of them.

Multiple views − DBMS offers multiple views for different users. A user who is in the Sales department will have a different view of database than a person working in the Production department. This feature enables the users to have a concentrate view of the database according to their requirements.

Security − Features like multiple views offer security to some extent where users are unable to access data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage. DBMS offers many different levels of security features, which enables multiple users to have different views with different features. For example, a user in the Sales department cannot see the data that belongs to the Purchase department. Additionally, it can also be managed how much data of the Sales department should be displayed to the user. Since a DBMS is not saved on the disk as traditional file systems, it is very hard for miscreants to break the code.

Users

A typical DBMS has users with different rights and permissions who use it for different purposes. Some users retrieve data and some back it up. The users of a DBMS can be broadly categorized as follows −

Administrators − Administrators maintain the DBMS and are responsible for administrating the database. They are responsible to look after its usage and by whom it should be used. They create access profiles for users and apply limitations to maintain isolation and force security. Administrators also look after DBMS resources like system license, required tools, and other software and hardware related maintenance.

Designers − Designers are the group of people who actually work on the designing part of the database. They keep a close watch on what data should be kept and in what format. They identify and design the whole set of entities, relations, constraints, and views.

End Users − End users are those who actually reap the benefits of having a DBMS. End users can range from simple viewers who pay attention to the logs or market rates to sophisticated users such as business analysts.

FILE SYSTEM VS DBMS

S.No	Difference factor	File System	DBMS
1	Definition	**Database management system (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, Oracle, SAP, dBASE, FoxPro, IBM dB2, SQLite etc.	**A file management system** is an abstraction to store, retrieve, management and update a set of files. A File Management System keep track on the files and also manage them.
2	Data Redundancy	In file system approach, each user defines and implements the needed files for a specific application to run. For example in sales department of an enterprise, One user will be maintaining the details of how many sales personnel are there in the sales department and their grades. Another user will be maintaining the sales person salary details.	Although the database approach does not remove redundancy completely, it controls the amount of redundancy in the database because in database approach, a single repository of data is maintained that is defined once and then accessed by many users. The fundamental characteristic of database approach is that the database system not only contains data's but it contains complete definition or description of the database 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. For e.g. if we want to change or update 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 place then data inconsistency will occur i.e. the appearance of same data will differ from each other.	In DBMS, as there is no or less data redundancy, data remains 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 file system for every operation we have to we have to make different programs.	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.
6	Data isolation	In file system 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 database must follow or satisfy some rules or consistency constraints. For e.g. 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. 8In file system, there is no procedure to check these constraints automatically.	DBMS maintains the data integrity by enforcing the constraints by adding appropriate code.
8	Security problems	In file system there is no or very less security. General security provided by file system are locks, guards etc.	DBMS have high level security like encryption, passwords, biometric security (fingerprint matching, face and voice detection etc) etc.

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. E.g. if A want to transfer 5000rs to B's a/c. In this case A's a/c should be debited and B's a/c should be credited with the same amount. Let suppose A's a/c is debited with 5000rs and then transaction fails. Now the transaction is incomplete because B's a/c 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 the transaction log in which intermediate values are stored.
10	Concurrent Access Anomalies	Any multi-user database application has to have some 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 for editing, user Y reads the same row 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 row. File system does not provide any procedure to stop such type of anomalies.	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 or write a data item, an exclusive lock is issued to one of these transactions. Until and unless the transaction release that lock no other transaction can acquire the lock and hence cannot update/write the data item.

ADVANTAGES OF DBMS

Controlling Data Redundancy

In non-database systems each application program has its own private files. In this case, the duplicated copies of the same data is created in many places. In DBMS, all data of an organization is integrated into a single database file. The data is recorded in only one place in the database and it is not duplicated.

Sharing of Data

In DBMS, data can be shared by authorized users of the organization. The database administrator manages the data and gives rights to users to access the data. Many users can be authorized to access the same piece of information simultaneously. The remote users can also share same data. Similarly, the data of same database can be shared between different application programs.

Data Consistency

By controlling the data redundancy, the data consistency is obtained. If a data item appears only once, any update to its value has to be performed only once and the updated value is immediately available to all users. If the DBMS has controlled redundancy, the database system enforces consistency.

Integration of Data

In Database management system, data in database is stored in tables. A single database contains multiple tables and relationships can be created between tables (or associated data entities). This makes easy to retrieve and update data.

Integration Constraints

Integrity constraints or consistency rules can be applied to database so that the correct data can be entered into database. The constraints may be applied to data item within a single record or the may be applied to relationships between records.

Data Security

Form is very important object of DBMS. You can create forms very easily and quickly in DBMS. Once a form is created, it can be used many times and it can be modified very easily. The created forms are also saved along with database and behave like a software component. A form provides very easy way (user-friendly) to enter data into database, edit data and display data from database. The non-technical users can also perform various operations on database through forms without going into technical details of a fatabase.

Report Writers

Most of the DBMSs provide the report writer tools used to create reports. The users can create very easily and quickly. Once a report is created, it can be used may times and it can be modified very easily. The created reports are also saved along with database and behave like a software component.

Control Over Concurrency

In a computer file-based system, if two users are allowed to access data simultaneously, it is possible that they will interfere with each other. For example, if both users attempt to perform update operation on the same record, then one may overwrite the values recorded by the other. Most database management systems have sub-systems to control the concurrency so that transactions are always recorded with accuracy.

Backup and Recovery Procedures

In a computer file-based system, the user creates the backup of data regularly to protect the valuable data from damage due to failures to the computer system or application program. It is very time consuming method, if amount of data is large. Most of the DBMSs provide the 'backup and recovery' sub-systems that automatically create the backup of data and restore data if required.

Data Independence

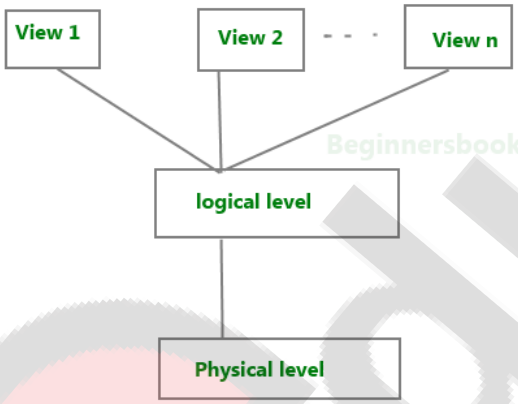
The separation of data structure of database from the application program that uses the data is called data independence. In DBMS, you can easily change the structure of database without modifying the application program.

VIEW OF DATA

DATA ABSTRACTION

Database systems are made-up of complex data structures. To ease the user interaction with database, the developers hide internal irrelevant details from users. This process of hiding irrelevant details from user is called data abstraction.

3 levels of abstraction



Three Levels of data abstraction

We have three levels of abstraction:

Physical level: This is the lowest level of data abstraction. It describes how data is actually stored in database. You can get the complex data structure details at this level.

Logical level: This is the middle level of 3-level data abstraction architecture. It describes what data is stored in database.

View level: Highest level of data abstraction. This level describes the user interaction with database system.

Example: Let's say we are storing customer information in a customer table. At physical level these records can be described as blocks of storage (bytes, gigabytes, terabytes etc.) in memory. These details are often hidden from the programmers.

At the logical level these records can be described as fields and attributes along with their data types, their relationship among each other can be logically implemented. The programmers generally work at this level because they are aware of such things about database systems.

At view level, user just interact with system with the help of GUI and enter the details at the screen, they are not aware of how the data is stored and what data is stored; such details are hidden from them.

INSTANCES & SCHEMA

Definition of schema: Design of a database is called the schema. Schema is of three types: Physical schema, logical schema and view schema.

The design of a database at physical level is called physical schema, how the data stored in blocks of storage is described at this level.

Design of database at logical level is called logical schema, programmers and database administrators work at this level, at this level data can be described as certain types of data records gets stored in data structures, however the internal details such as implementation of data structure is hidden at this level (available at physical level).

Design of database at view level is called view schema. This generally describes end user interaction with database systems.

Definition of instance: The data stored in database at a particular moment of time is called instance of database. Database schema defines the variable declarations in tables that belong to a particular database; the value of these variables at a moment of time is called the instance of that database.

DATABASE LANGUAGES

Data Definition Language

Data Definition Language (DDL) statements are used to classify the database structure or schema. It is a type of language that allows the DBA or user to depict and name those entities, attributes, and relationships that are required for the application along with any associated integrity and security constraints. Here are the lists of tasks that come under DDL:

CREATE – used to create objects in the database

ALTER – used to alters the structure of the database

DROP – used to delete objects from the database

TRUNCATE – used to remove all records from a table, including all spaces allocated for the records are removed

COMMENT – used to add comments to the data dictionary

RENAME – used to rename an object

Data Manipulation Language

A language that offers a set of operations to support the fundamental data manipulation operations on the data held in the database. Data Manipulation Language (DML) statements are used to manage data within schema objects. Here are the lists of tasks that come under DML:

SELECT – It retrieve data from the a database

INSERT – It inserts data into a table

UPDATE – It updates existing data within a table

DELETE – It deletes all records from a table, the space for the records remain

MERGE – UPSERT operation (insert or update)

CALL – It calls a PL/SQL or Java subprogram

EXPLAIN PLAN – It explains access path to data

LOCK TABLE – It controls concurrency

Data Control Language

There is another two forms of database sub-languages. The Data Control Language (DCL) is used to control privilege in Database. To perform any operation in the database, such as for creating tables, sequences or views we need privileges. Privileges are of two types,

System – creating session, table etc are all types of system privilege.

Object – any command or query to work on tables comes under object privilege. DCL is used to define two commands. These are:

Grant – It gives user access privileges to database.

Revoke – It takes back permissions from user.

Transaction Control Language (TCL)

Transaction Control statements are used to run the changes made by DML statements. It allows statements to be grouped together into logical transactions.

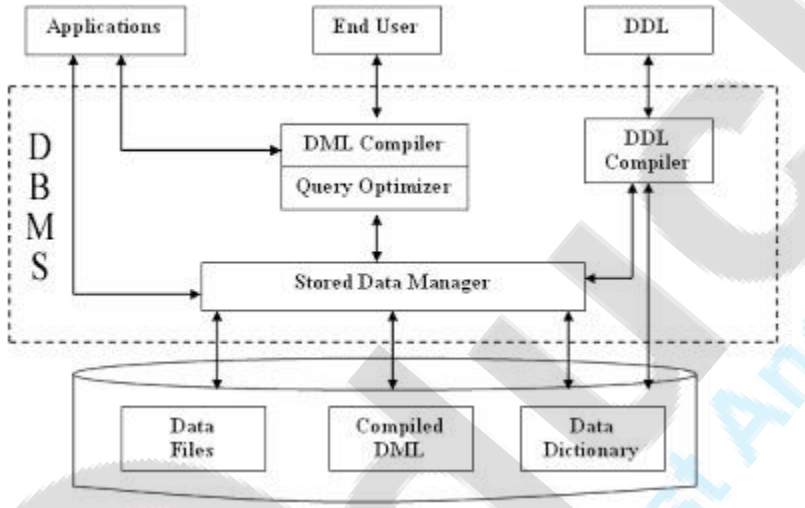
COMMIT – It saves the work done

SAVEPOINT – It identifies a point in a transaction to which you can later roll back

ROLLBACK – It restores database to original since the last COMMIT

SET TRANSACTION – It changes the transaction options like isolation level and what rollback segment to use.

STRUCTURE OF DBMS



DATA MODELS

A database model shows the logical structure of a database, including the relationships and constraints that determine how data can be stored and accessed. Individual database models are designed based on the rules and concepts of whichever broader data model the designers adopt. Most data models can be represented by an accompanying database diagram.

Types of database models-

There are many kinds of data models. Some of the most common ones include:

Hierarchical database model

Relational model

Network model

Object-oriented database model

Entity-relationship model

Document model

Entity-attribute-value model

Star schema

The object-relational model, which combines the two that make up its name.

ROLE OF DBA

A Database Administrator is a person or a group of person who are responsible for managing all the activities related to database system. This job requires a high level of expertise by a person or group of person. There are very rare chances that only a single person can manage all the database system activities so companies always have a group of people who take care of database system.

Deciding the hardware device-

Depending upon the cost, performance and efficiency of the hardware, it is DBA who have the duty of deciding which hardware devise will suit the company requirement. It is hardware that is an interface between end users and database so it needed to be of best quality.

Managing Data Integrity-

Data integrity should be managed accurately because it protects the data from unauthorized use. DBA manages relationship between the data to maintain data consistency.

Decides Data Recovery and Back up method-

If any company is having a big database, then it is likely to happen that database may fail at any instance. It is require that a DBA takes backup of entire database in regular time span. DBA has to decide that how much data should be backed up and how frequently the back should be taken. Also the recovery of data base is done by DBA if they have lost the database.

Tuning Database Performance-

Database performance plays an important role for any business. If user is not able to fetch data speedily then it may loss company business. So by tuning an modifying sql commands a DBA can improves the performance of database.

Capacity Issues-

All the databases have their limits of storing data in it and the physical memory also has some limitations. DBA has to decide the limit and capacity of database and all the issues related to it.

Database design-

The logical design of the database is designed by the DBA. Also a DBA is responsible for physical design, external model design, and integrity control.

Database accessibility-

DBA writes subschema to decide the accessibility of database. He decides the users of the database and also which data is to be used by which user. No user has to power to access the entire database without the permission of DBA.

Decides validation checks on data-

DBA has to decide which data should be used and what kind of data is accurate for the company. So he always puts validation checks on data to make it more accurate and consistence.

Monitoring performance-

If database is working properly then it doesn't mean that there is no task for the DBA. Yes of course, he has to monitor the performance of the database. A DBA monitors the CPU and memory usage.

Decides content of the database-

A database system has many kind of content information in it. DBA decides fields, types of fields, and range of values of the content in the database system. One can say that DBA decides the structure of database files.

Provides help and support to user-

If any user needs help at any time then it is the duty of DBA to help him. Complete support is given to the users who are new to database by the DBA.

Database implementation-

Database has to be implemented before anyone can start using it. So DBA implements the database system. DBA has to supervise the database loading at the time of its implementation.

Improve query processing performance-

Queries made by the users should be performed speedily. As we have discussed that users need fast retrieval of answers so DBA improves query processing by improving their performance.