Maximizing Efficiency with a Database Management System

Bhuvaneshwari Madadi

Student

Vaagdevi College of Engineering, Warangal.

Abstract: - *An early attempt to computerise the manual system was with file-based systems. The majority of organisations are dealing with a sharp growth in the processing of enormous amounts of data. Consequently, it creates the challenge of collecting, storing, managing, and analysing vast amounts of data that are stored in different formats. Currently, to overcome these challenges a more efficient software is used i.e., database management systems (DBMS). Database systems are more reliable and adaptable compared to file processing systems. Using a database management system enables developers to eliminate duplicate data and provide measures for analysing large data sets. Instead of producing multiple versions of the similar data saved in new files for each application, data may be shared via a DBMS and is therefore more secure and maintained. This paper highlights a general idea of the changes brought about by storing and managing data and the emergence of database management systems (DBMS), which has led to a paradigm shift.*

Keywords: - File system, database management systems (DBMS)

INTRODUCTION: -

Data is information that has been transformed into an efficient form for transit or processing. A file system is a traditional-based method in which each department maintains and controls its own data with the assistance of a data management professional. The amount of information at our fingertips is practically increasing, and the importance of data as an organisational asset is well acknowledged. To manage such a huge amount of data, increasingly sophisticated and adaptable data management solutions are required instead of traditional databases. Consequently, in order to resolve this problem, a database approach is implemented. The database has a history of more than 45 years. The database and database management systems evolved in accordance with the demands and needs of the time [1].

Before the emergence of database systems

Computer data management has gone a long way in the decades that have passed. Today's users take the numerous advantages of a database system for granted. However, not long ago, computers depended on a considerably less sophisticated and costly method to data storage known as the file-based system.

File Systems

A file system represents the organisation of data. It is an application programme that organises and maintains the files on a storage medium. Many files are grouped into directories to form file systems. The directories also include additional folders and files. The file system handles basic tasks including managing files, file naming, setting access limits, and so on. Previously, punched cards were used to store data and, subsequently, files. However, files have no such benefit but rather have some limitations. Digital file systems and files are termed and modelled after traditional paper-based filing systems, while they maintain and access data in a similar logic-based approach. File systems vary from device to device. The file system of a hard disc, for example, holds the data saved on the device. The file system on a computer consists of virtual space for storing data. The file system not only stores data, but it also determines how files are organised on the device. FAT, NTFS, ext2, and ext3 are some examples.

Advantages of File Systems

1. Less complex and easier to use.
2. Multiple access methods.
3. File-based systems are employed by fewer businesses or individuals to store substantially less data. As a result, the data may be retrieved more quickly and conveniently.

Disadvantages of File Systems

One of the earliest and most basic techniques for managing data in computer systems is the file processing system. Although some applications still use these systems today, they have a number of drawbacks when compared to contemporary database management systems.

- **Data redundancy:** In file processing systems, it is possible for identical information to exist repeatedly in distinct files, which may result in data redundancy. This consumes storage space and makes data maintenance more complex.
- **Data inconsistency:** Data inconsistencies could arise as a result of data redundancy. Inconsistency in data, or a conflict between various versions of the same data, costs storage space and necessitates more labour.
- **Data isolation:** Multiple locations and formats are used to store files. Data isolation occurs when a file's data cannot be mapped to its associated data in another file.
- **Atomicity issues:** The "all or nothing" principle, which argues that either all of the operations in a transaction are completed or nothing is done, is related to the atomicity of a transaction. In file processing systems, achieving atomicity is challenging.
- **Integrity Problem:** A set of data can be integrated if it complies with particular consistency constraints. When a file processing system's data integrity is subpar, it is extremely difficult to add more constraints.
- **Lack of Security:** File Processing Systems lack built-in security mechanisms that protect against unauthorised access to data. All users must be subject to some limitations on their ability to access data.

Database management systems (DBMS)

An organized collection of details or information that is usually stored digitally in a computer system is known as a database (DB). A DBMS, or database management system, is a generalized tool for handling huge databases that is accessible via specialized applications for data interrogation, preservation, and analysis. Many of its user interfaces offer a variety of languages to help all users, from clerks to data administrators. [2]. MySQL, Oracle, and other significant commercial databases are used in a variety of applications.

Characteristics of DBMS

- Provides security
- Removes redundancy
- Establishing a robust backup and recovery system
- Maintains data integrity
- Offers a multi-user environment that enables concurrent data access and manipulation.
- Provides multiple views of data for various users in a single organization.
- The database management system must provide the attributes of atomicity, consistency, isolation, and durability (ACID).
- Provides users with a query language that can be used to insert, retrieve, update, and delete data in a database.

Applications of DBMS

A database management system is used in a variety of fields. Following are a few examples of applications:

1. **Railway Reservation System -** In order to maintain track of ticket sales, provide users with information about delayed trains, and track train arrival and departure information, the database for the railway reservation system is essential.
2. **Library Management System -** The library has a large number of volumes, making it challenging to keep track of them all. A database management system (DBMS) is used to keep track of all the data related to publication dates, title, author, and availability of books.
3. **Banking -** Every day, banks will process a thousand transactions, and we're doing it all online. All of this is only possible due to the DBMS that controls all bank transactions.
4. **Education Sector -** In schools and colleges, database systems are widely used to store and retrieve information about students, staff, courses, examinations, attendance, fees, and other types of data.
5. **Social media sites -** Millions of individuals register for social media accounts every day on sites like Facebook, Twitter, LinkedIn, and Google Plus. All of the client data is stored in the information base using the data set administration framework, which also makes us ready to interact with other people.

6. **Finance:** There are many things to deal with in finance these days, such as managing financial statements, keeping information, and storing sales data. Maintaining and analysing historical data is made easier with the help of a financial database.
7. **Telecommunication:** A telecom database is needed to store data like client names, phone numbers, calling information, prepaid and post-paid connection records, network activity, bill details, balance details, etc. The telecom industry generates enormous amounts of data, and managing big data without a database is very challenging.

Data abstraction

System usability depends on how well the data was obtained. A database consists of sophisticated data structures. Developers use abstraction to simplify the system's usability for users and make data retrieval more effective [3]. There are three different levels of data abstraction in a DBMS, and they are as follows:

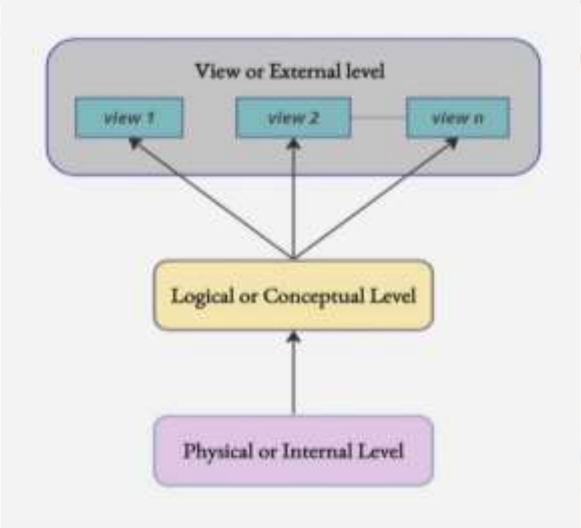


Fig.1 Levels of data abstraction

- **Physical Level:** In a database management system, the physical or internal layer represents the lowest abstract level of data. This layer of data abstraction is where unprocessed data is physically kept as files. It explains complex data structures in great detail, making them difficult to understand, which is why it is concealed from the end user. This level is only accessible to the database administrator (DBA).
- **Logical level:** The logical or conceptual level is a level higher than the physical level. It describes what information will be saved in the database and how it will be related. This layer contains no information on how the data will be seen by the end user. Data administrators (DBAs) extract information from the raw data that is present at the physical level with the help of the logical level.
- **View Level:** The highest level of data abstraction is known as the View or External Level. Several distinct views of the same data can be described at the view level. Here, the user accesses the database using a separate application to obtain the data. At this level, end users can communicate with the system and obtain the data depending on their queries.

Instance and Schema

A database instance is essentially a collection of information as well as data that it stores at any particular moment. These instances are easily modifiable utilising CRUD operations, such as the insertion and deletion of data and information. It should be emphasised that no search query makes any kind of change in any instances. The concept of schema refers to the general description of a database. The schema remains constant throughout the database.

DBMS ARCHITECTURE

The architecture of the DBMS determines its design. Users can use the DBMS architecture to fulfil their requests while connecting to the database. The basic client-server design manages an enormous number of PCs, internet servers, database servers, and other network-connected components. The client-server architecture comprises of a workstation and different PCs linked by a network.

Types of DBMS architecture

DBMS architecture is classified into three types:

1. **1-Tier Architecture:** In this architecture, each client request does not need a network association to execute the action on the database because the database is promptly accessible on the client machine. Changes or updates made here will be reflected in the database. The 1-tier architecture is utilised for local application development, where computer specialist may specifically interface with the database for immediate feedback.
2. **2-Tier Architecture:** End-users benefit more from this type of architecture. In a two-tier design, the database framework is located on the server computer and the application of DBMS is located on the client computer; these two computers are linked by a reliable network. For server-client interaction, application association interfaces such as JDBC and ODBC are employed. The server side is in charge of query processing and transaction management functions. User interfaces and application programmes are executed on the client side.

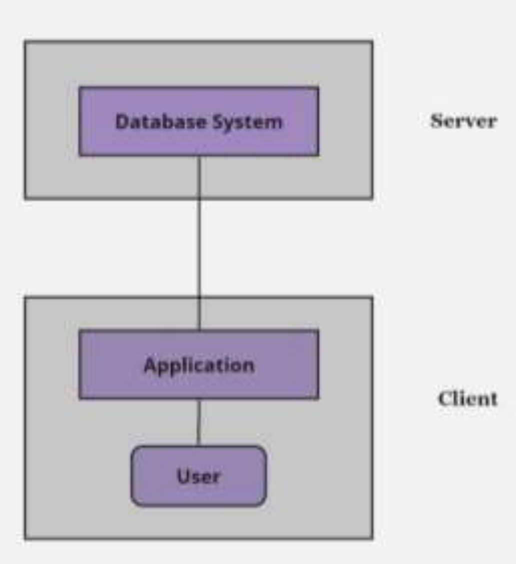


Fig.2 2-Tier Architecture

3. **3-Tier Architecture:** There another layer that exists between the client and server machines in a three-tier architecture. This intermediary layer is known as the application layer. This layer is in charge of storing connection software and limiting data transfer. In this architecture, client applications interact with server applications, which then internally interact with the server's database systems. This prevents client applications from directly accessing the database systems installed on the server computer.

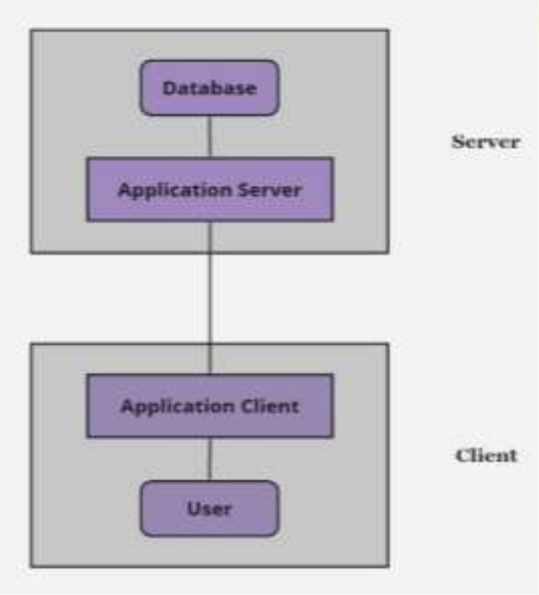


Fig.3 3-Tier Architecture

DATA MODELS

Data models in DBMS make it easier for programmers to create a physical database by providing a clear image of the data. This aids comprehension of the architecture on conceptual, physical, and logical levels. Data modeling is the process of modeling the requirements for data description, data semantics, and data consistency.

Types of data models:

Hierarchical Model: Data is arranged into a tree-like structure in this data model, with only one root to which all other data is related. Data is organised hierarchically (either bottom-up or top-down). In this model, parent-child relationships are used to represent data, and there will only be one parent node for each child node.

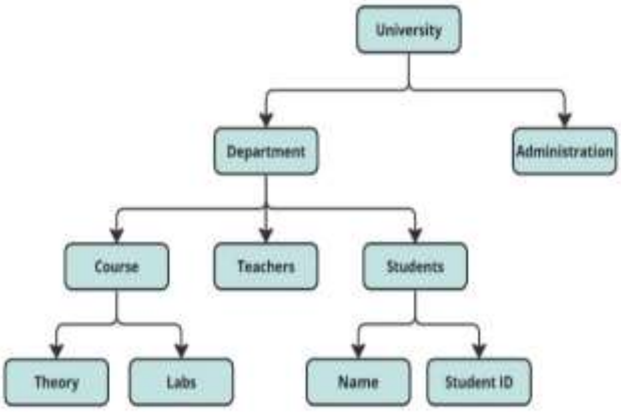


Fig.4 Hierarchical Model

Network Model: A network model is only a broader version of a hierarchical data model. This paradigm arranges data more like a network and permits it to have numerous parent nodes. It assists you in addressing the requirement to model increasingly complicated relationships, such as the many-to-many relationship. Entities are arranged in a graph that has several pathways leading to them.

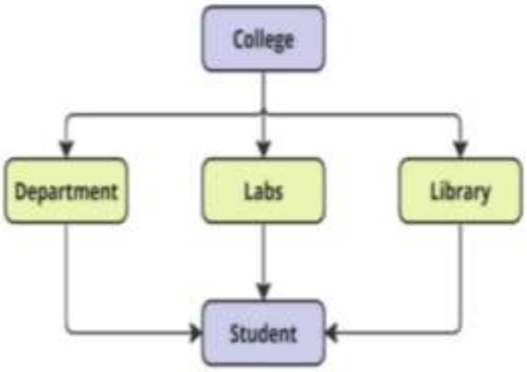


Fig.5 Network Model

Entity-Relationship Model: An entity-relationship model uses an ER diagram to visually depict the database's structure. While incorporating scenarios from the real world into the underlying database model, the ER Model generates an entity set, a relationship set, general characteristics, and constraints. Entities, their properties, and the interactions between them form the foundation of the ER Model.

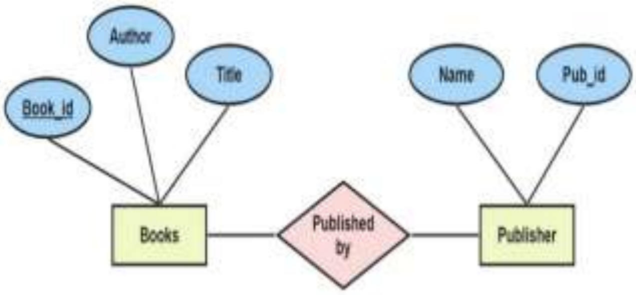
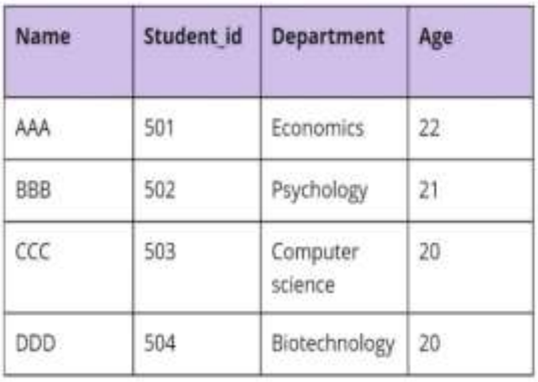


Fig.6 Entity-Relationship Model

Relational Model: This data model is the most extensively used. In this model, the database is shown as a set of relationships that are displayed as rows and columns in a two-dimensional table. Each column represents an attribute, and each row is referred to as a tuple. A relational model that is stored in set structures and controlled via SQL.



Name	Student_id	Department	Age
AAA	501	Economics	22
BBB	502	Psychology	21
CCC	503	Computer science	20
DDD	504	Biotechnology	20

Fig.7 Relational Model

DATABASE LANGUAGES

In the realm of data, we require a certain set of programming languages so that the DBMS software can comprehend our requirements and manage the data contained in the databases appropriately. Database languages and query languages are terms used to describe these programming languages. Data in a database is read, updated, and stored using database languages.

Types of Databases Languages

1. **Data Definition Language (DDL)**

   A collection of unique commands called Data Definition Language (DDL) enables us to specify and alter the metadata and database structure. It is a language that lets the user specify the information and its connection to various forms of data. The DDL commands are: create, alter, rename, drop, and truncate.

2. **Data Manipulation Language (DML)**

   The Data Manipulation Language (DML) is a collection of unique commands that enables us to retrieve and alter data held in already-existing schema objects. It is a language that offers a collection of functions to facilitate fundamental data manipulation activities on the data in databases. DML commands include insert, remove, update, select, merge, and call.

3. Data Control Language (DCL)

The Data Control Language (DCL) is a collection of special commands designed for managing user access in a database system. To access the stored data, DCL is required. It is mostly used to revoke and provide access to a database. Grant and Revoke are the DCL commands.

4. Transaction Control Language (TCL)

A collection of unique instructions known as Transaction Control Language (TCL) is used for handling transactions within databases. It is responsible for carrying out the modifications generated through the data manipulation language statements. Commit and rollback are TCL commands.

TRANSACTION MANAGEMENT

The practise of managing several transactions submitted by many database server clients while ensuring that the ACID contract, or the atomicity, consistency, isolation, and durability of each individual transaction, can be met, is known as transaction management [4,5]. One or more database access operations are included in a database transaction, which is a logical processing unit in a database management system (DBMS). Database transactions are essentially representations of real-world business events.

ACID PROPERTIES

Atomicity, consistency, isolation, and durability, or ACID in DBMS, are four key concepts. To keep the database consistent, these properties are checked both before and after the transaction.

Atomicity: The word atomicity refers to the fact that data remains atomic. A logical unit of execution is a transaction. You can either finish it completely or not at all. There is no option for partial execution.

Consistency: The term consistency implies that the value should always be retained. The transition from a single consistent state to another should happen once the transaction is finished.

Isolation: The word 'isolation' refers to separation. This characteristic guarantees that multiple transactions can take place simultaneously without leading to an inconsistent database state.

Durability: The permanence of anything is ensured by its durability. The modifications in the database ought to persist once a transaction is successfully completed. Even in the event of a system breakdown.

CONCLUSION

When compared to current database management systems, file processing systems have significant shortcomings. While file systems continue to be utilised in some applications, they are inadequate for maintaining sophisticated data structures or vast volumes of data. Modern database management systems make data administration in computer systems more efficient and effective. A database management system (DBMS) can be quicker than a file system for some processes, such as sophisticated queries involving several tables or enormous volumes of data.

REFERENCES

[1] A. A. Govande, “Paradigm Shift of Database,” International Journal of Engineering Development and Research, 2016, Vol 4, No. 4, pp.395-400.
[2] J.P. Fry, E.H.Sibley, “Evolution of data-base management systems,” ACM Computing Surveys (CSUR), 1976, vol 8, No. 1, pp.7-42.
[3] Vairavel C, “Introduction of Database Management System,” Journal of Emerging Technologies and Innovative Research, March 2019, Vol 6, No. 4, pp.541-545.
[4] Bernstein P.A, Newcomer E, “Principles of Transaction Processing,” 2nd ed. Morgan Kaufmann, Burlington, 2009.
[5] Gray J, Reuter A, “Transaction processing: concepts and techniques.” Elsevier, 1992.