B.Sc. (Computer Science)-Part-III
Subject Name- “DBMS”
Paper-I (Video Part –II)

By- Prof. Dileep Kumar Sahu

Assistant Professor

Department of Computer Application

Govt. Vishwanath Yadav Tamaskar Post Graduate Autonomous College, Durg (C.G.)

Email ID: dileepksahu20@gmail.com

Unit- III: Concept of DBMS and Data Models

Chapter-I: Introduction to DBMS

By- Prof. Dileep Kumar Sahu

Content

1. Views of Data
2. **THE LOGICAL DBMS ARCHITECTURE**
3. **THE PHYSICAL DBMS ARCHITECTURE**

Objective

- Define the functions of DBA;
- Explain the three-tier architecture of DBMS, and
- Identify the need for three-tier architecture.

Architecture of a DBMS or **Views of Data**

There are two different ways to look at the architecture of a DBMS:

1. the logical DBMS architecture and
2. the physical DBMS architecture.

The LOGICAL ARCHITECTURE deals with the way data is stored and presented to users, while the PHYSICAL ARCHITECTURE is concerned with the software components that make up a DBMS.

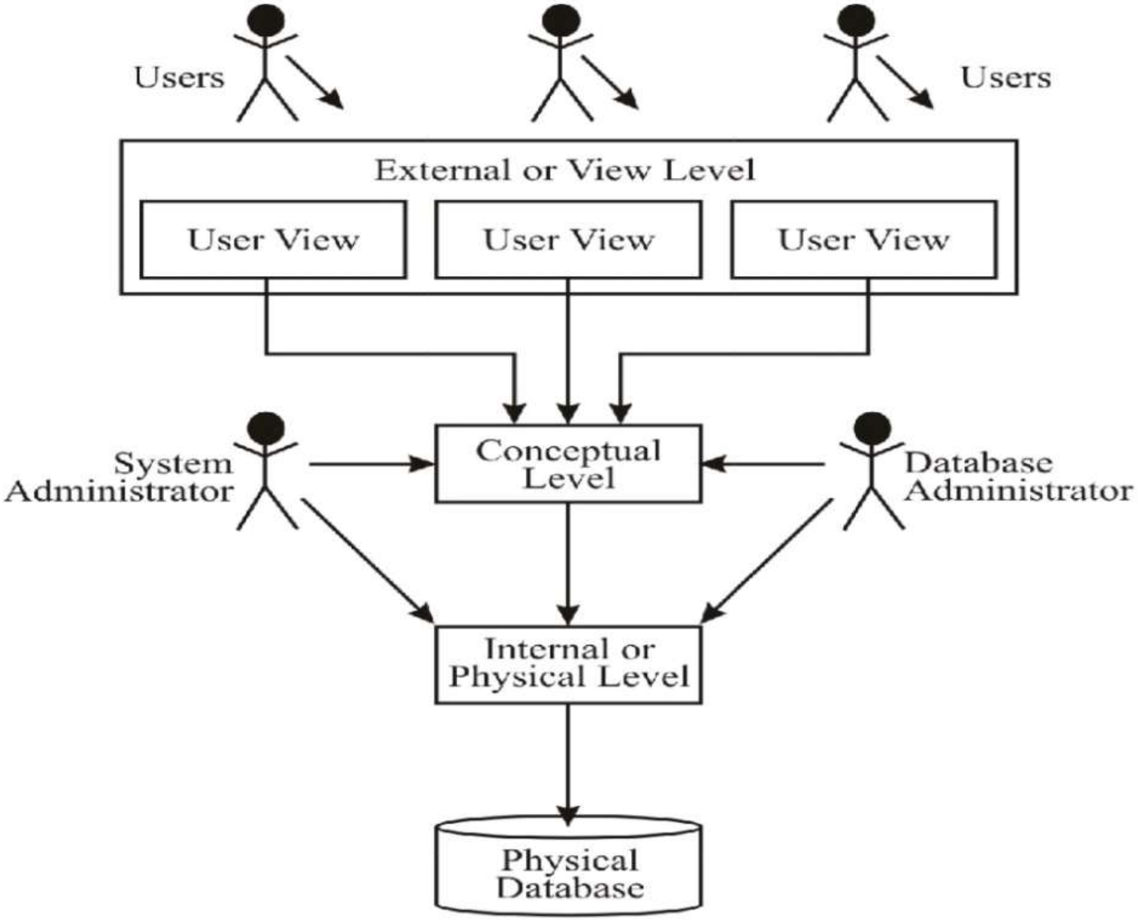
THE LOGICAL DBMS ARCHITECTURE

- The logical architecture describes how data in the database is perceived by users.
- It is not concerned with how the data is handled and processed by the DBMS, but only with how it looks.
- the users can manipulate the data without worrying about where it is located or how it is actually stored.
- This results in the database having different levels of abstraction.

Views of Data (THE LOGICAL DBMS ARCHITECTURE)

- It is also called as the ANSI/SPARC model.
- It divides the system into three levels of abstraction:

1. The Internal Or Physical Level
2. The Conceptual Level
3. The External Or View Level.



- The diagram below shows the logical architecture for a typical DBMS

By- Prof. Dileep Kumar Sahu

External or View Level

- It is the highest level of abstraction of database.
- It provides a window on the conceptual view, which allows the user to see only the data of interest to them.
- The user can be either an application program or an end user.
- There can be many external views as any number of external schema can be defined and they can overlap each other.
- It consists of the definition of logical records and relationships in the external view.
- It also contains the methods for deriving the objects such as entities, attributes and relationships in the external view from the Conceptual view.

Conceptual Level

- The conceptual level presents a logical view of the entire database as a unified whole.
- It allows the user to bring all the data in the database together and see it in a consistent manner. Hence, there is only one conceptual schema per database.
- The first stage in the design of a database is to define the conceptual view, and a DBMS provides a data definition language for this purpose.
- It describes all the records and relationships included in the database.
- It does not provide any storage or access details, but defines the information content only.

The Internal or Physical Level

- The collection of files permanently stored on secondary storage devices is known as the physical database.
- The physical or internal level is the one closest to physical storage, and it provides a low-level description of the physical database, and an interface between the operating systems file system and the record structures used in higher levels of abstraction.
- It is at this level that record types and methods of storage are defined, as well as how stored fields are represented, what physical sequence the stored records are in, and what other physical structures exist.

Mappings between Levels and Data Independence

- The three levels of abstraction in the database do not exist independently of each other.
- There must be some correspondence, or mapping, between the levels.
- There are two types of mappings: the conceptual/internal mapping and the external/conceptual mapping.
- The conceptual/internal mapping lies between the conceptual and internal levels, and defines the correspondence between the records and the fields of the conceptual view and the files and data structures of the internal view.
- *If the structure of the stored database is changed, then the conceptual/ internal mapping must also be changed accordingly so that the view from the conceptual level remains constant.*

Mappings between Levels and Data Independence

- The external/conceptual view lies between the external and conceptual levels, and defines the correspondence between a particular external view and the conceptual view.
- some elements found in a particular external view may be different from the conceptual view.
- For example, several fields can be combined into a single (virtual) field, which can also have different names from the original fields.
- If the structure of the database at the conceptual level is changed, then the external/conceptual mapping must change accordingly so that the view from the external level remains constant.

The need for three level architecture

- The objective of the three level architecture is to separate each user’s view of the database from the way the database is physically represented.
- **Support of multiple user views:** Each user is able to access the same data, but have a different customized view of the data.
  - Each user should be able to change the way he or she views the data and this change should not affect other users.
- **Insulation between user programs and data that does not concern them:** Users should not directly deal with physical storage details, such as indexing or hashing.
  - The user’s interactions with the database should be independent of storage considerations

Insulation between conceptual and physical structures

- It can be defined as:

1. The Database Administrator should be able to change the storage structures without affecting users' views.

2. The internal structure of the database should be unaffected by the changes to the physical aspects of the storage, such as changing to a new storage device.

3. The DBA should be able to change the conceptual structure of the database without affecting all users.

DBMS Architecture

- The physical architecture describes the software components used to enter and process data, and how these software components are related and interconnected.
- It is possible to identify a number of key functions which are common to most database management systems.
- The components that normally implement these functions are shown in Figure , which depicts the physical architecture of a typical DBMS.

DBMS Architecture

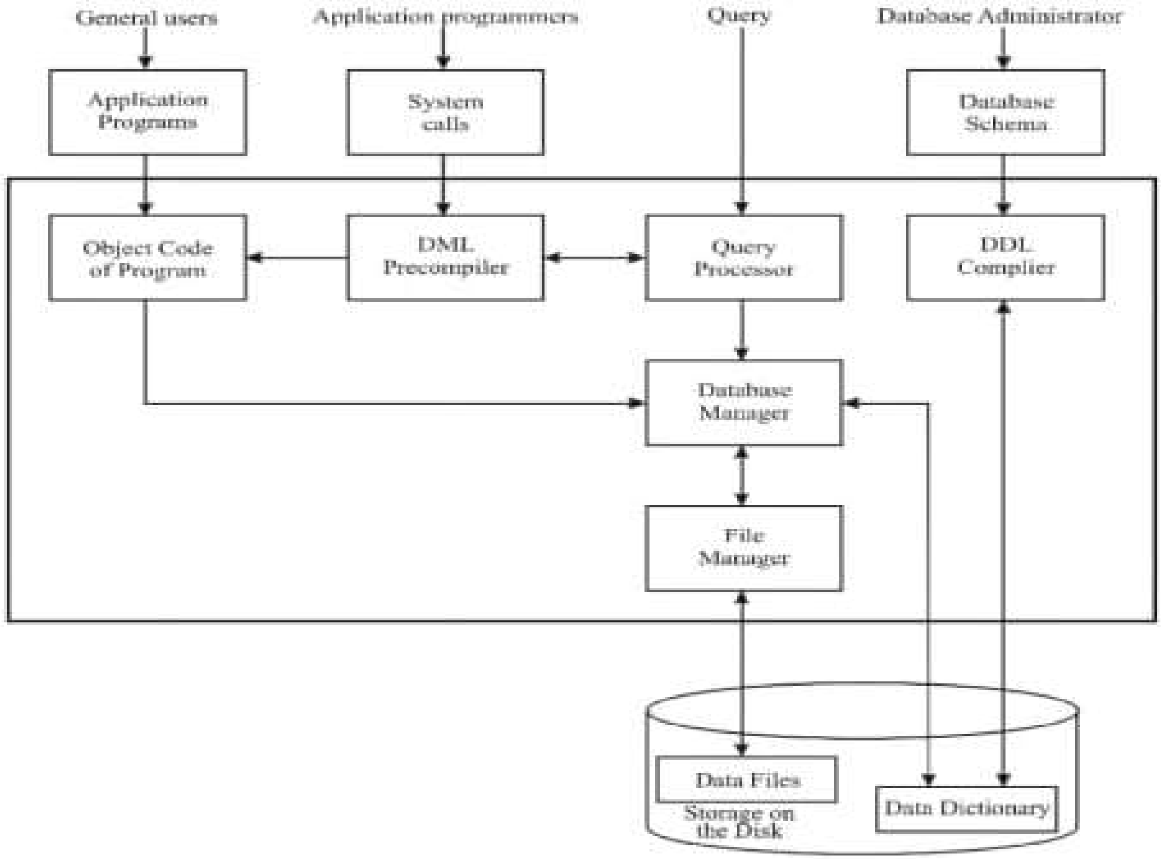


Figure 4: DBMS Structure

By- Prof. Dileep Kumar Sahu

DBMS Architecture: Database Languages

The database system may be partitioned into the following modules. Some functions (for example, file systems) may be provided by the operating system.

Database Languages:

- All the Database Management systems have two basic sets of Languages –
  - Data Definition Language (DDL) that contains the set of commands required to define the format of the data that is being stored and
  - Data Manipulation Language (DML) which defines the set of commands that modify, process data to create user definable output.
  - The DML statements can also be written in an application program.

DBMS Architecture

DML Pre-compiler:

- The DML pre-compiler converts DML statements (such as SELECT…FROM in Structured Query Language (SQL) covered in Block 2) embedded in an application program to normal procedural calls in the host language.
- The pre-compiler interacts with the query processor in order to generate the appropriate code.

DDL Compiler:

- The DDL compiler converts the data definition statements (such as CREATE TABLE …. in SQL) into a set of tables containing metadata tables.
- These tables contain information concerning the database and are in a form that can be used by other components of the DBMS.
- These tables are then stored in a system catalog or data dictionary.

DBMS Architecture

File manager:

- It manages the allocation of space on disk storage.
- It establishes and maintains the list of structures and indices defined in the internal schema that is used to represent information stored on disk.
- However, the file manager does not directly manage the physical output and input of data.
- It passes the requests on to the appropriate access methods, which either read data from or write data into the system buffer or cache.
- The file manager can be implemented using an interface to the existing file subsystem provided by the operating system of the host computer or it can include a file subsystem written especially for the DBMS.

DBMS Architecture: Database Manager

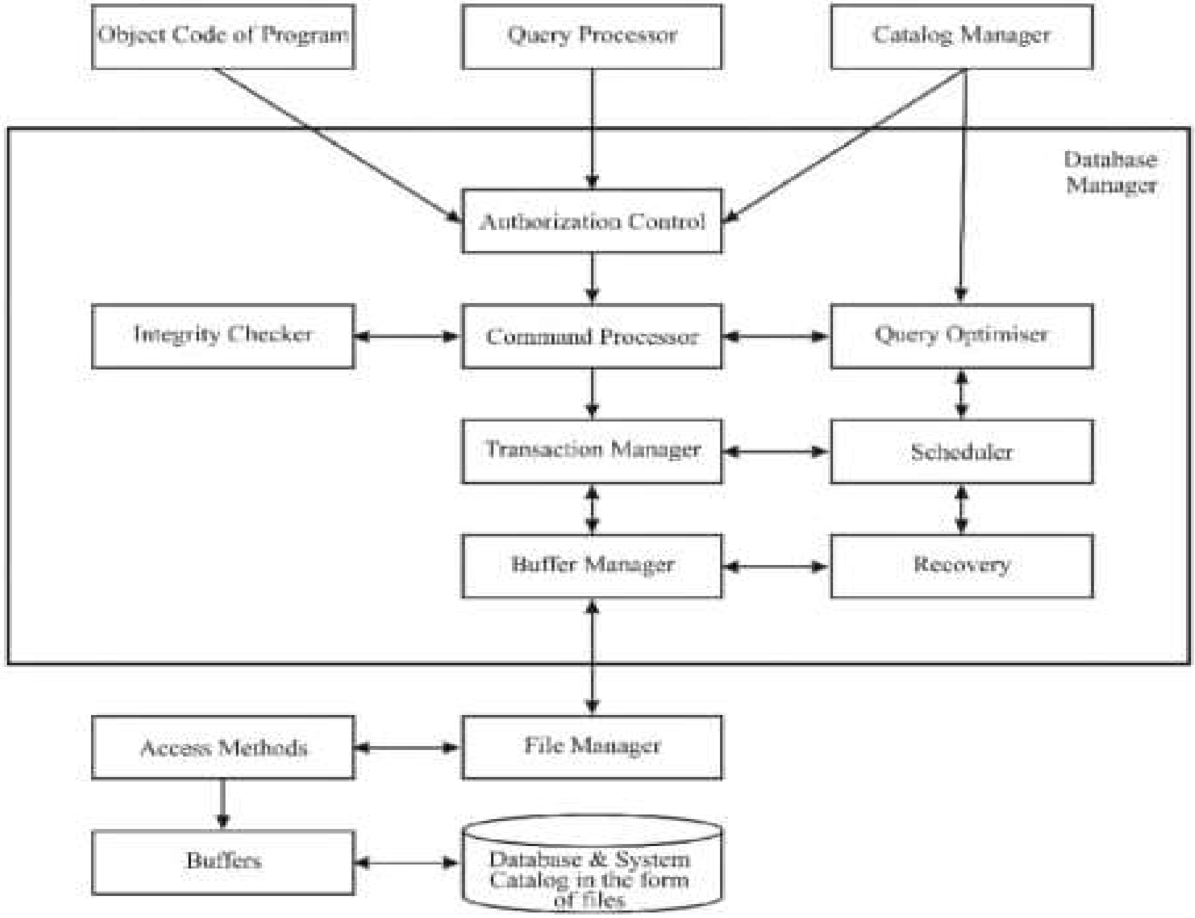
Database Manager:

- A database manager is a program module responsible for interfacing with the database file system to the user queries. The responsibilities of Database Manager are:
- **Interaction with file manager:** The database manager translates the various DML statements into low-level file system commands. Thus, the database manager is responsible for the actual storing, retrieving and updating of data in the database.
- **Integrity enforcement:** The data values stored in the database must satisfy certain types of consistency constraints.
  - For example, the balance of a bank account may never fall below a prescribed amount (for example, Rs. 1000/-).
  - Similarly the number of holidays per year an employee may be having should not exceed 8 days.
  - These constraints must be specified explicitly by the DBA.
  - If such constraints are specified, then the database manager can check whether updates to the database result in the violation of any of these constraints and if so appropriate action may be imposed.

DBMS Architecture: Database Manager

- **Security enforcement:** As discussed above, not every user of the database needs to have access to the entire content of the database. It is the job of the database manager to enforce these security requirements.
- **Backup and recovery:** There are a variety of causes of failure, including disk crash, power failure and software errors.
  - In each of these cases, information concerning the database is lost.
  - It is the responsibility of database manager to detect such failures and restore the database to a state that existed prior to the occurrence of the failure.
  - This is usually accomplished through the backup and recovery procedures.
- **Concurrency control:** When several users update the database concurrently, the consistency of data may no longer be preserved.

Components of Database Manager



By- Prof. Dileep Kumar Sahu

Figure 5: Components of Database Manager

DBMS Architecture: Database Manager

The major components of a DATABASE MANAGER are:

- **Authorization control:** This module checks that the user has necessary authorization to carry out the required function.
- **Command Processor:** command processor converts commands to a logical sequence of steps.
- **Integrity checker:** The integrity checker checks that the requested operation satisfies all necessary integrity constraints such as key constraints.
- **Query Optimizer:** This module determines an optimal strategy for the query execution.
- **Transaction Manager:** This module performs the required processing of operations of various transactions.
  - The transaction manager maintains tables of authorization Concurrency.
  - The DBMS may use authorization tables to allow the transaction manager to ensure that the user has permission to execute the desired operation on the database.

DBMS Architecture: Database Manager

- **Scheduler:** This module is responsible for ensuring that concurrent operations or transactions on the database proceed without conflicting with one another.
  - It controls the relative order in which transaction operations are executed.
  - The DBMS checks the concurrency control tables before executing an operation to ensure that the data used by it is not locked by another statement.
- **Recovery Manager:** It is responsible for transaction commit and abort, that is success or failure of transaction.
- **Buffer Manager:** This module is responsible for the transfer of data between main memory and secondary storage, such as disk and tape.

DBMS Architecture

- **Query Processor:**
- The query language processor is responsible for receiving query language statements and changing them from the English-like syntax of the query language to a form the DBMS can understand.
- The query language processor usually consists of two separate parts: the parser and the query optimizer.
- The parser receives query language statements from application programs or command-line utilities and examines the syntax of the statements to ensure they are correct.
- The query optimiser examines the query language statement, and tries to choose the best and most efficient way of executing the query, such as: CPU time, disk time, network time, sorting methods, and scanning methods.

DBMS Architecture: Database Administrator(DBA)

- The DBA administers the three levels of the database and defines the global view or conceptual level of the database.
- The DBA also specifies the external view of the various users and applications and is responsible for the definition and implementation of the internal level, including the storage structure and access methods to be used for the optimum performance of the DBMS.
- Changes to any of the three levels due to changes in the organization and/or emerging technology are under the control of the DBA.
- Conceptual and External levels, are also defined by the DBA.
- The DBA is responsible for granting permission to the users of the database and stores the profile of each user in the database.
- The DBA is also responsible for defining procedures to recover the database from failures due to human, natural, or hardware causes with minimal loss of data.

DBMS Architecture: Database Administrator(DBA)

The Functions of DBA:

- Schema definition
- Storage Structure and access method definition.
- Schema and Physical organization modification.
- Granting of authorization for data access.
- Integrity constraint specification.

DBMS Architecture

Data Dictionary: A Data Dictionary stores information about the structure of the database.

  - A dictionary provides definitions of things.
  - A directory tells you where to find them.
  - It is also called data about data.

- A data dictionary/directory contains information (or data) about the data.
- A data dictionary would provide the definition of data items, how they fit into the data structure and how they relate to other entities in the database.
- In DBMS, the data dictionary stores the information concerning the external, conceptual and internal levels of the databases.
- It would combine the source of each data field value, that is from where the authenticate value is obtained. The frequency of its use and audit trail regarding the updates including user identification with the time of each update is also recorded in Data dictionary.

DBMS Architecture

An ideal data dictionary should include everything a DBA wants to know about the database:

- External, conceptual and internal database descriptions.
- Descriptions of entities (record types), attributes (fields), as well as cross-references, origin and meaning of data elements.
- Synonyms, authorization and security codes.
- Which external schemas are used by which programs, who the users are, and what their authorizations are.
- Statistics about database and its usage including number of records, etc.

Thank You

By- Prof. Dileep Kumar Sahu