DATABASE MANAGEMENT SYSTEM(ITC303)

Subject Incharge

B. Vijayalakshmi
Assistant Professor
email: balarajulakshmi@sfit.ac.in

Content

- Introduction
- Characteristics of databases
- File system V/s Database system
- Applications of DBMS

Introduction

Data: Collection of raw facts

ex: figures,images,statics,images,videos etc having no particular meaning

Information: processed or organised /summarized data .

Which has particular meaning

Database: Collection of related or Interrelated data is known as Database

examples: Banking System

Computerized medical records

Online Shopping System

Library Management System

DBMS:

Database Management Systems is a software or set of programs where we can manipulate the data

or

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

Technical Definition for DBMS

Def: A Database Management System (DBMS) is a large software package that controls the specification, organization,storage,retrieval and update of data in a database.

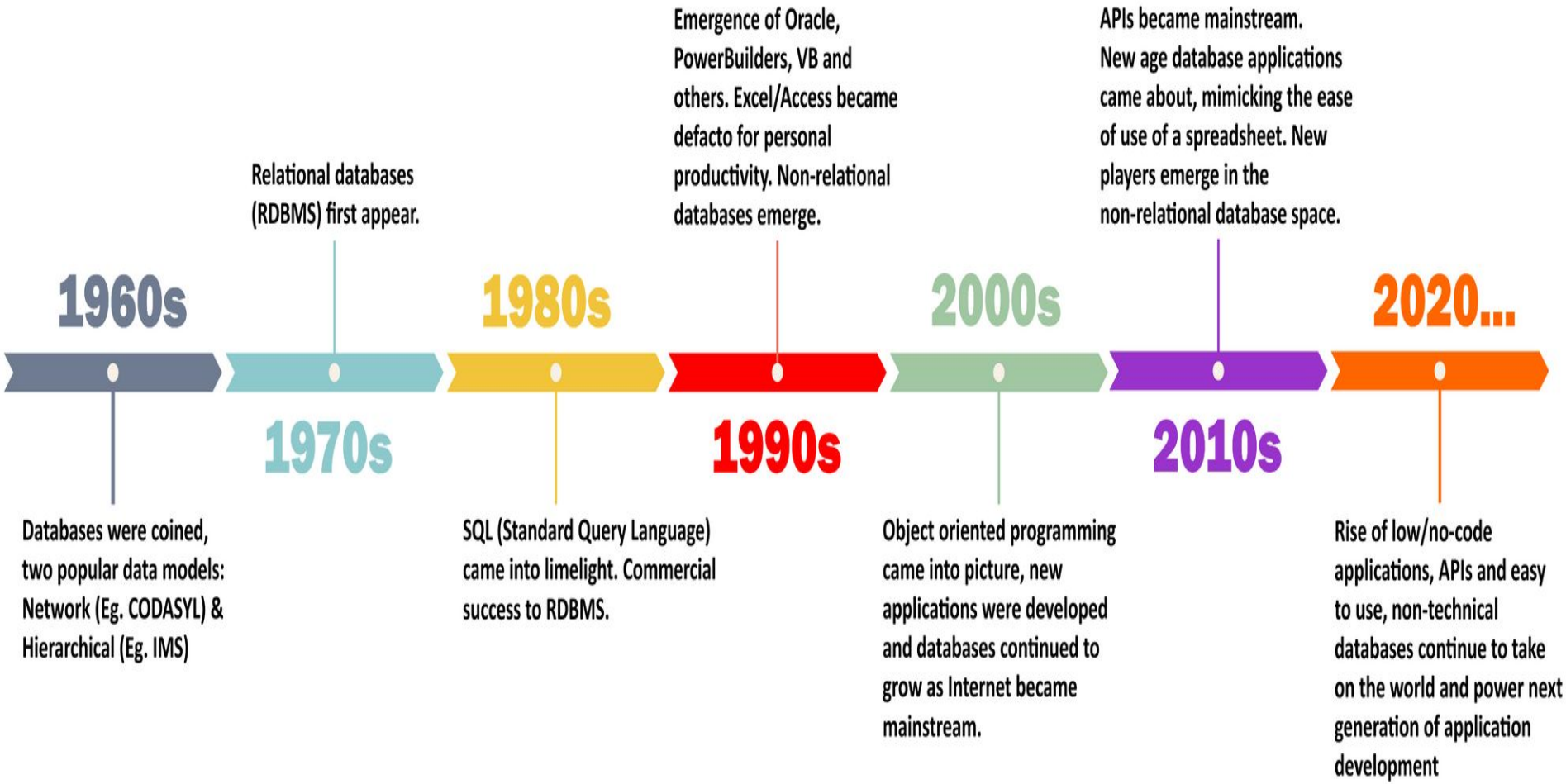
Introduction to history of Database Software

Early days of Computing :Computers were used for solving Mathematical Problems

In late 1950's Frequent data processing tasks were observed(Sorting)
Which leads to the development of generalized software modules

In 1963 first DBMS was developed
IBM developed a system called IMS(Information Management System) which is still in use for processing data

History of Databases (1960-2020)

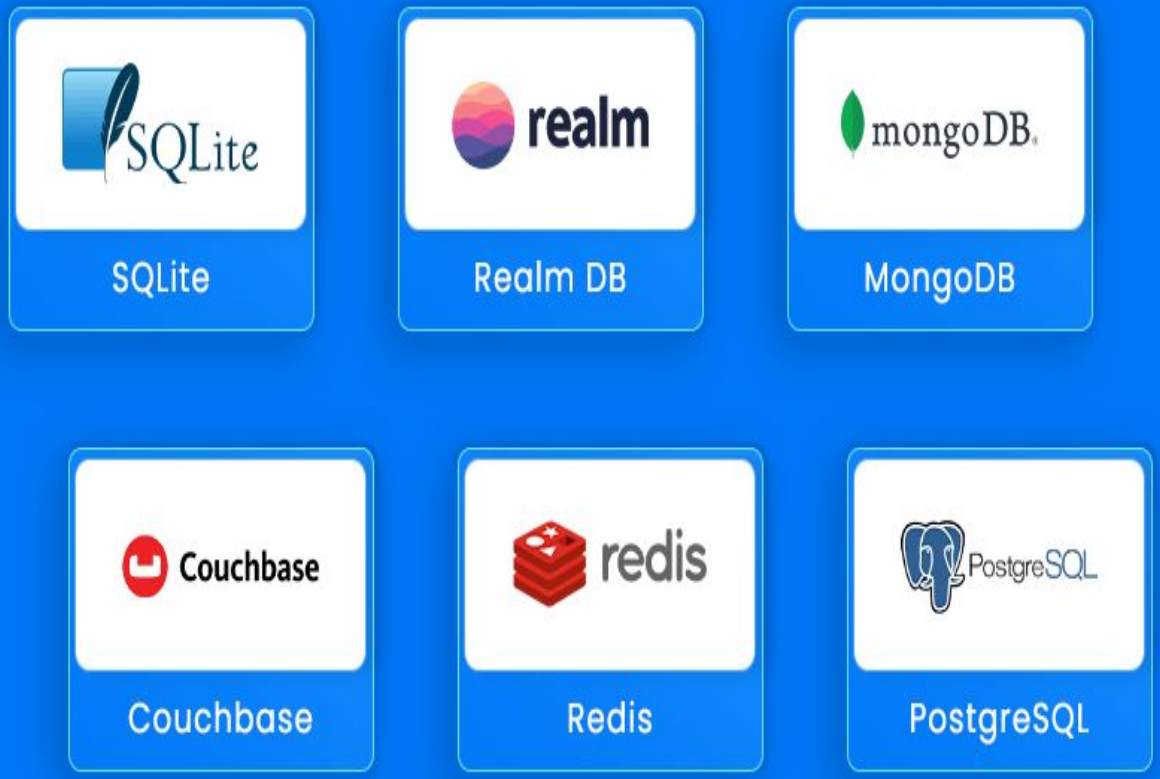


Popular Database Management Systems

POPULAR DATABASES FOR WEB APPLICATIONS

Most Popular RDBMS

Popular Mobile App Databases



Characteristics of databases

- **Self – describing** Nature of a Database system
- Insulation between **Programs** and **Data**, Data Abstraction
- Support of **Multiple Views** of the Data
- Sharing of **Data and Multiuser Transaction** Processing
- Uses a **digital repository** established on a server to store and manage the information
- DBMS contain **automatic backup** and **recovery procedures**

Example of a simplified database catalog

RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

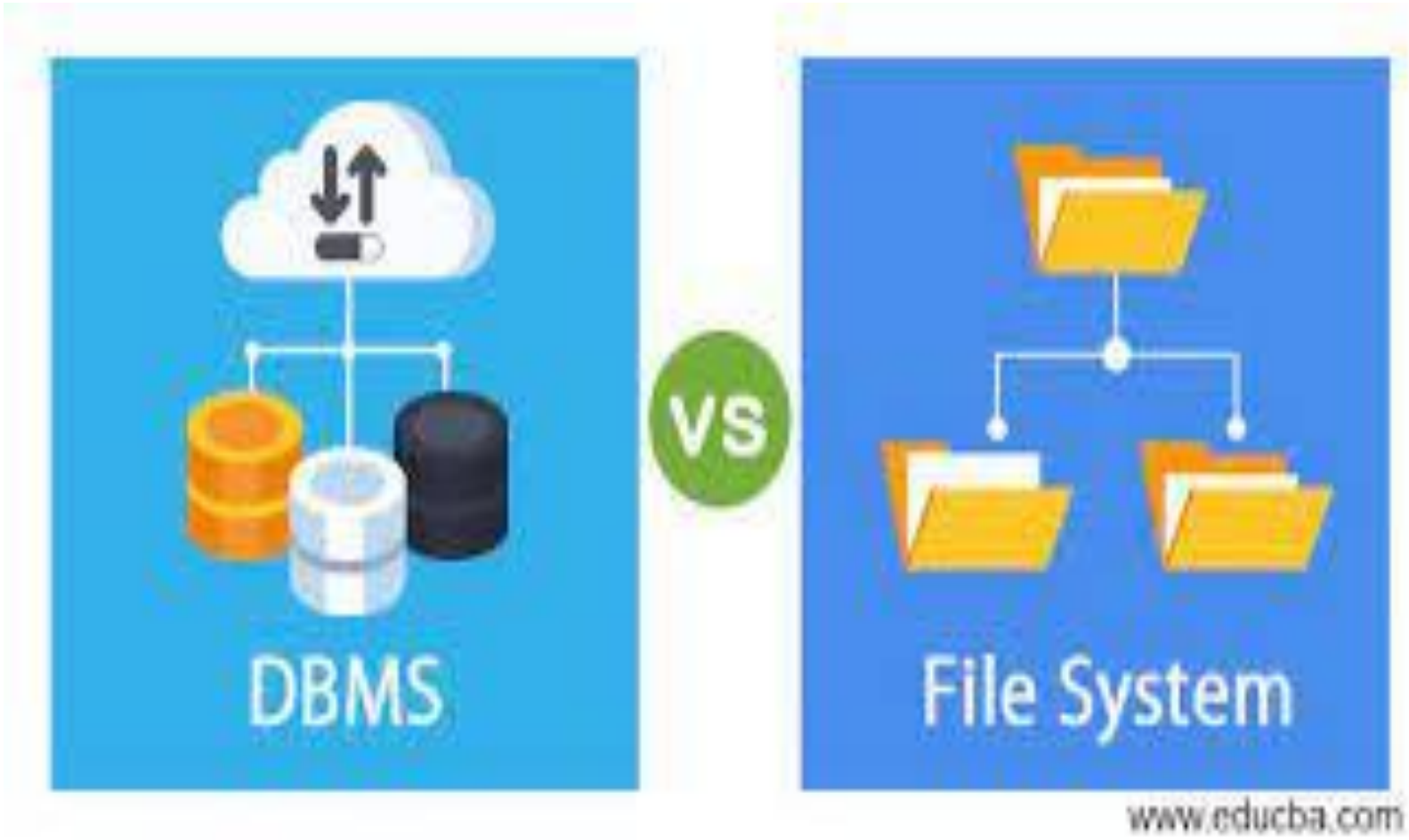
COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
....	....	.....
....	....	.....
....	....	.....
Prerequisite_number	XXXXNNNN	PREREQUISITE

Characteristics of databases

- It contains ACID properties which maintain data in a healthy state in case of failure
- It can reduce the complex relationship between data.
- It is used to support manipulation and processing of data.
- It is used to provide security of data.

File System Vs Database Management System



DBMS vs. File System

DBMS	File System
DBMS is a collection of data. In DBMS, the user is not required to write the procedures.	File system is a collection of data. In this system, the user has to write the procedures for managing the database.
DBMS gives an abstract view of data that hides the details.	File system provides the detail of the data representation and storage of data.
DBMS provides a crash recovery mechanism, i.e., DBMS protects the user from the system failure.	File system doesn't have a crash mechanism, i.e., if the system crashes while entering some data, then the content of the file will lost.
DBMS provides a good protection mechanism.	It is very difficult to protect a file under the file system.
DBMS contains a wide variety of sophisticated techniques to store and retrieve the data.	File system can't efficiently store and retrieve the data.
DBMS takes care of Concurrent access of data using some form of locking.	In the File system, concurrent access has many problems like redirecting the file while other deleting some information or updating some information.

Drawbacks of File Systems

- Data redundancy
- Data inconsistency
- Data Isolation
- Dependency on application programs
- Atomicity issues
- Data Security

Advantages of DBMS

- **Controls database redundancy:**

  It can control data redundancy because it stores all the data in one single database file and that recorded data is placed in the database.
- **Data sharing:**

  DBMS, the authorized users of an organization can share the data among multiple users.
- **Easily Maintenance:**

  It can be easily maintainable due to the centralized nature of the database system.

Advantages of DBMS

Reduce time: It reduces development time and maintenance need.

Backup: It provides backup and recovery subsystems which create automatic backup of data from hardware and software failures and restores the data if required.

Multiple user interface: It provides different types of user interfaces like graphical user interfaces, application program interfaces

Applications of DBMS

- Railway Reservation System
- Library Management System
- Banking
- Education Sector
- Social Media Sites
- Broadcast communications
- Account
- Manufacturing
- Online Shopping
- Human Resource Management

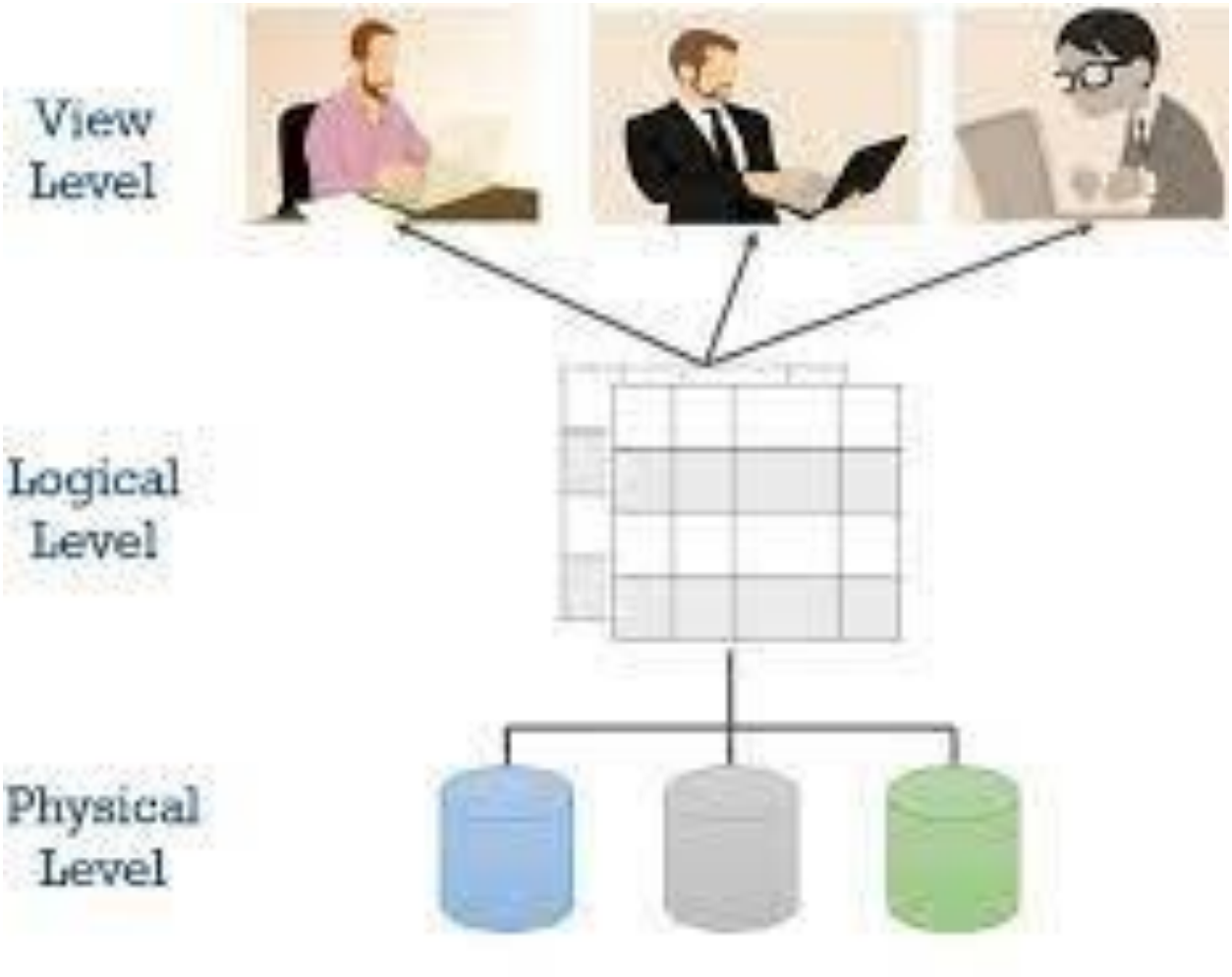
Data Abstraction

- Data Abstraction is **a process of hiding unwanted or irrelevant details from the end user**.
- Three levels of data abstraction:

  View Level

  Conceptual Level

  Physical Level

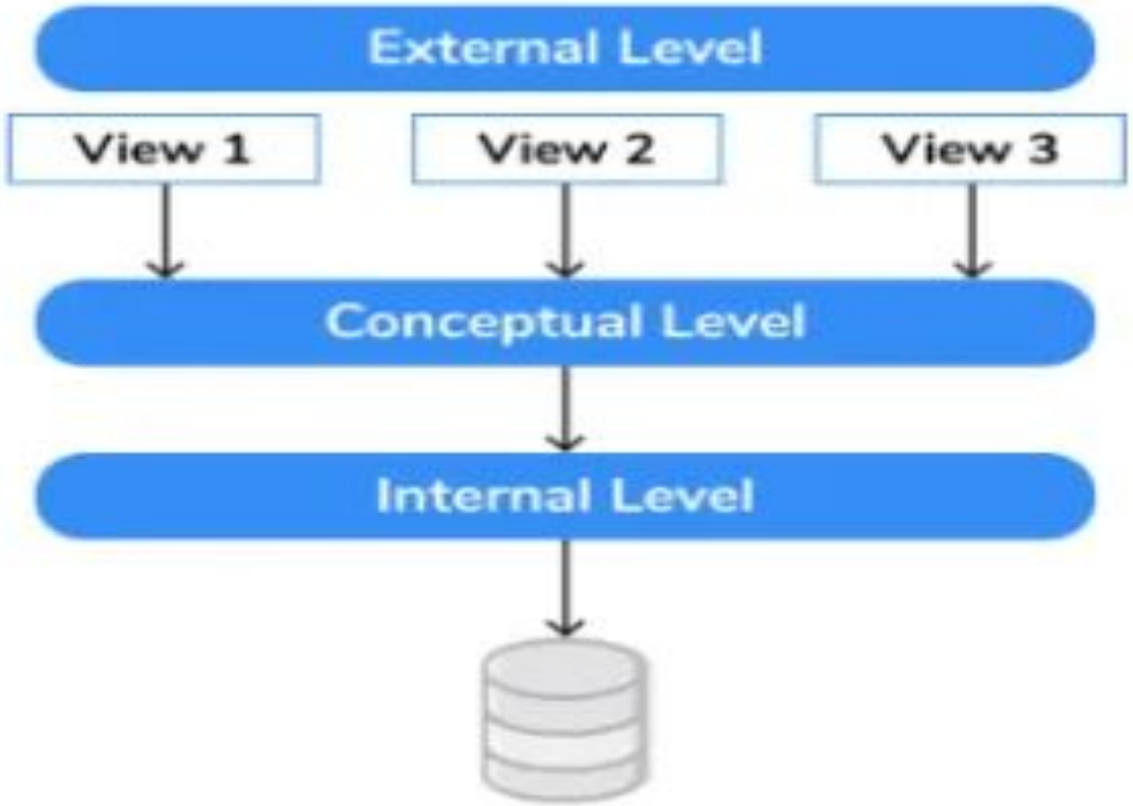


Front end

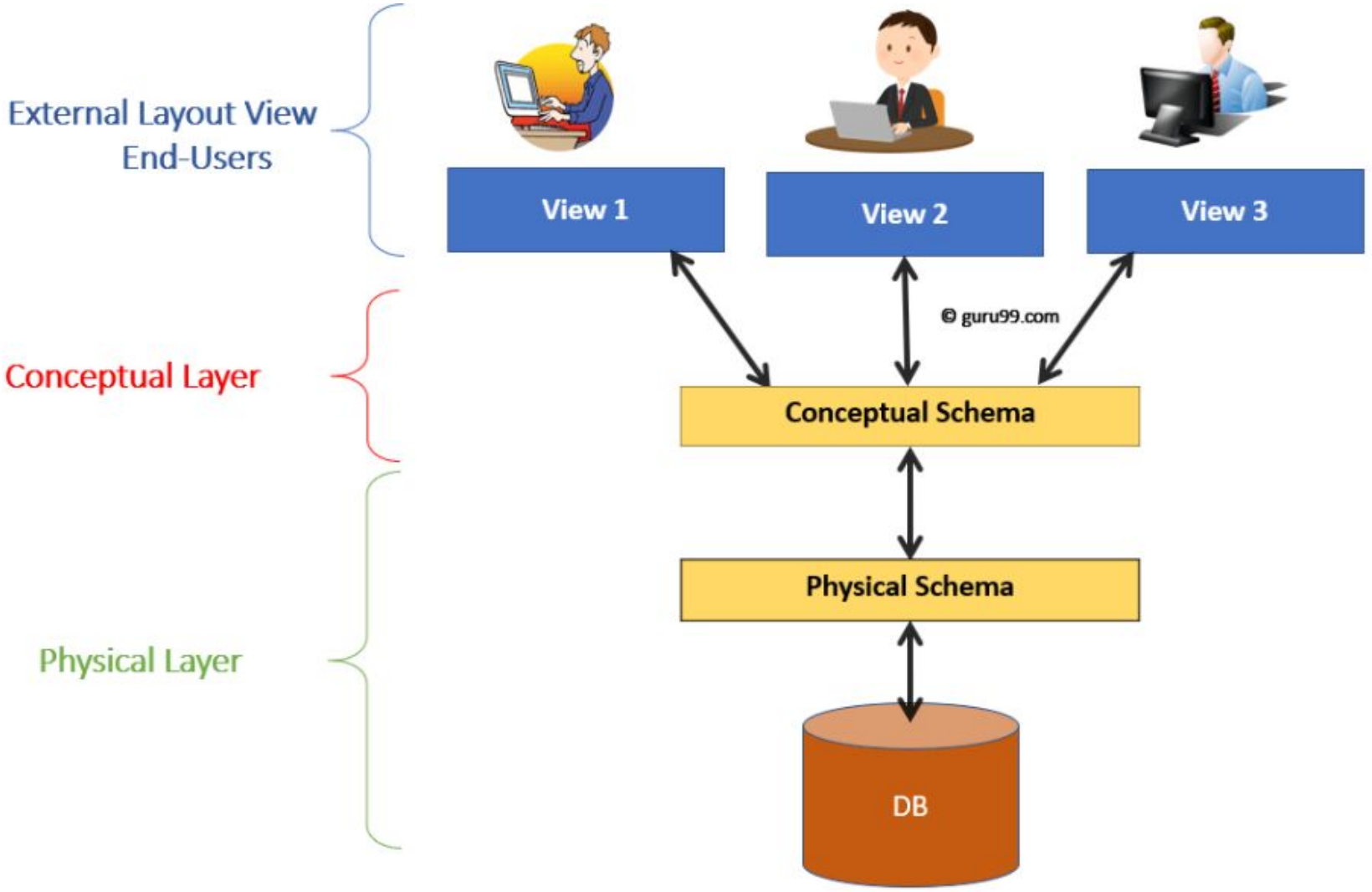
Design

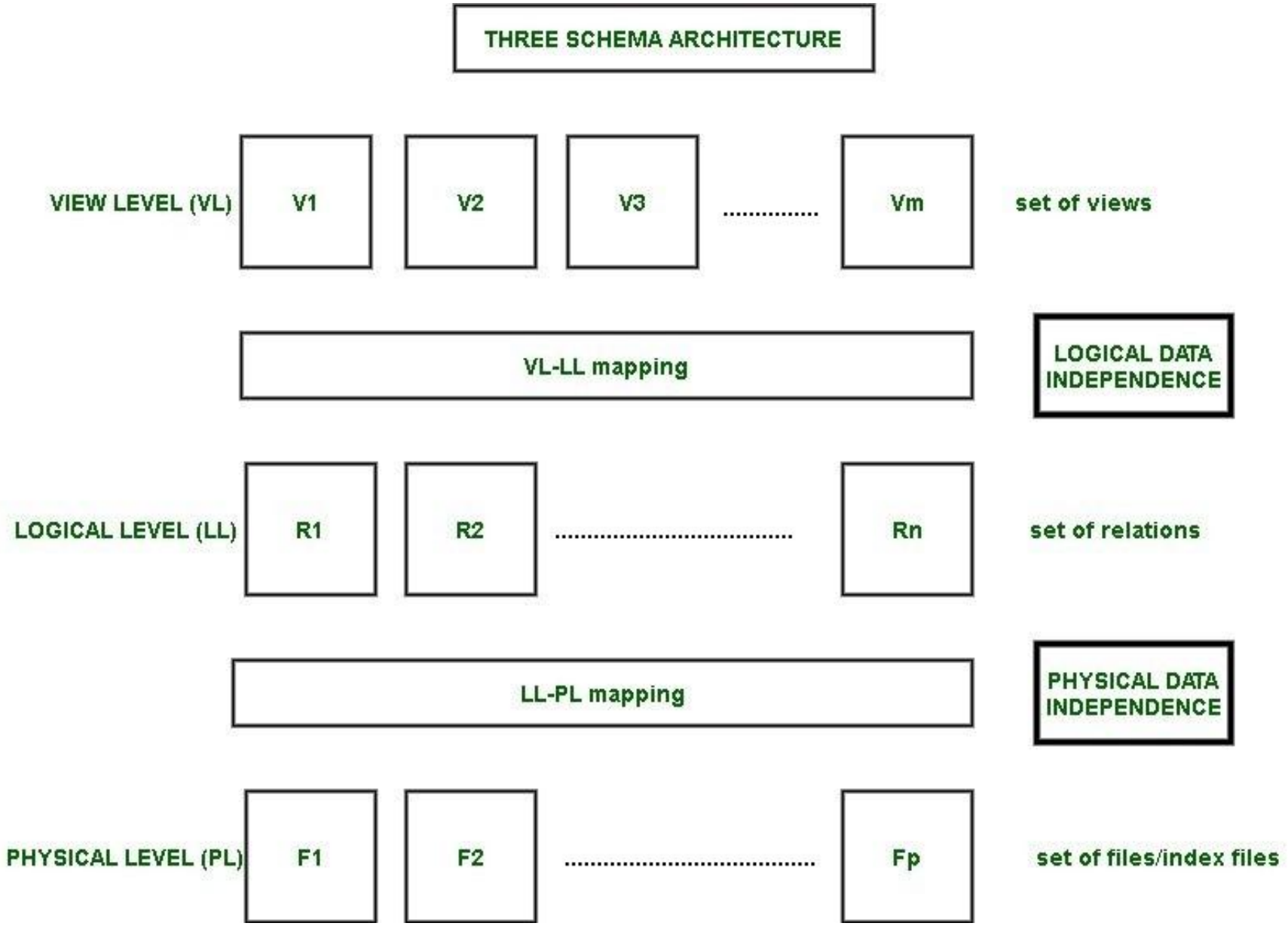
Data Admistrator

Data Abstraction and Independence



Data Independence




THREE SCHEMA ARCHITECTURE
VIEW LEVEL (VL)
V1
V2
V3
.....
Vm
set of views
VL-LL mapping
LOGICAL DATA INDEPENDENCE
LOGICAL LEVEL (LL)
R1
R2
.....
Rn
set of relations
LL-PL mapping
PHYSICAL DATA INDEPENDENCE
PHYSICAL LEVEL (PL)
F1
F2
.....
Fp
set of files/index files

Data Independence:

•The ability to modify a schema definition in one level without affecting a schema definition in the next higher level .

Physical Data Independence:

• It is the ability to modify the physical schema without causing application programs to be rewritten.

• Modifications at the physical level are occasionally necessary to improve performance.

Logical Data Independence:

• It is ability to modify the logical schema without causing application programs to be rewritten.

• Modifications at the Logical level are necessary whenever the logical structure of the database is altered.

•

External View in C language

```
Struct Student {
Char stud_name(10);
Int matriculation_number(6); Char
sex(1);
```

Conceptual View

```
STUDENT
STUD_NAME CHARACTER(10);
MATRICULATION_NUMBER INT(6);
SEX CHARACTER(1);
```

Internal view STORED_STUDENT

BYTES=20 PREFIX

BYTES=6,OFFSET=0

Database Users:

- Users may be divided into those who actually use and control the content (called “Actors on the Scene”) and those who enable the database to be developed and the DBMS software to be designed and implemented (called “Workers Behind the Scene”).

System Structure DBMS

- The design of Database System must include the considerations of the interface between the **database system** and the **operating system.**

Functional Components of a database system are:

Query Processor Components

Storage Manager Components

Transaction Management

Storage Management

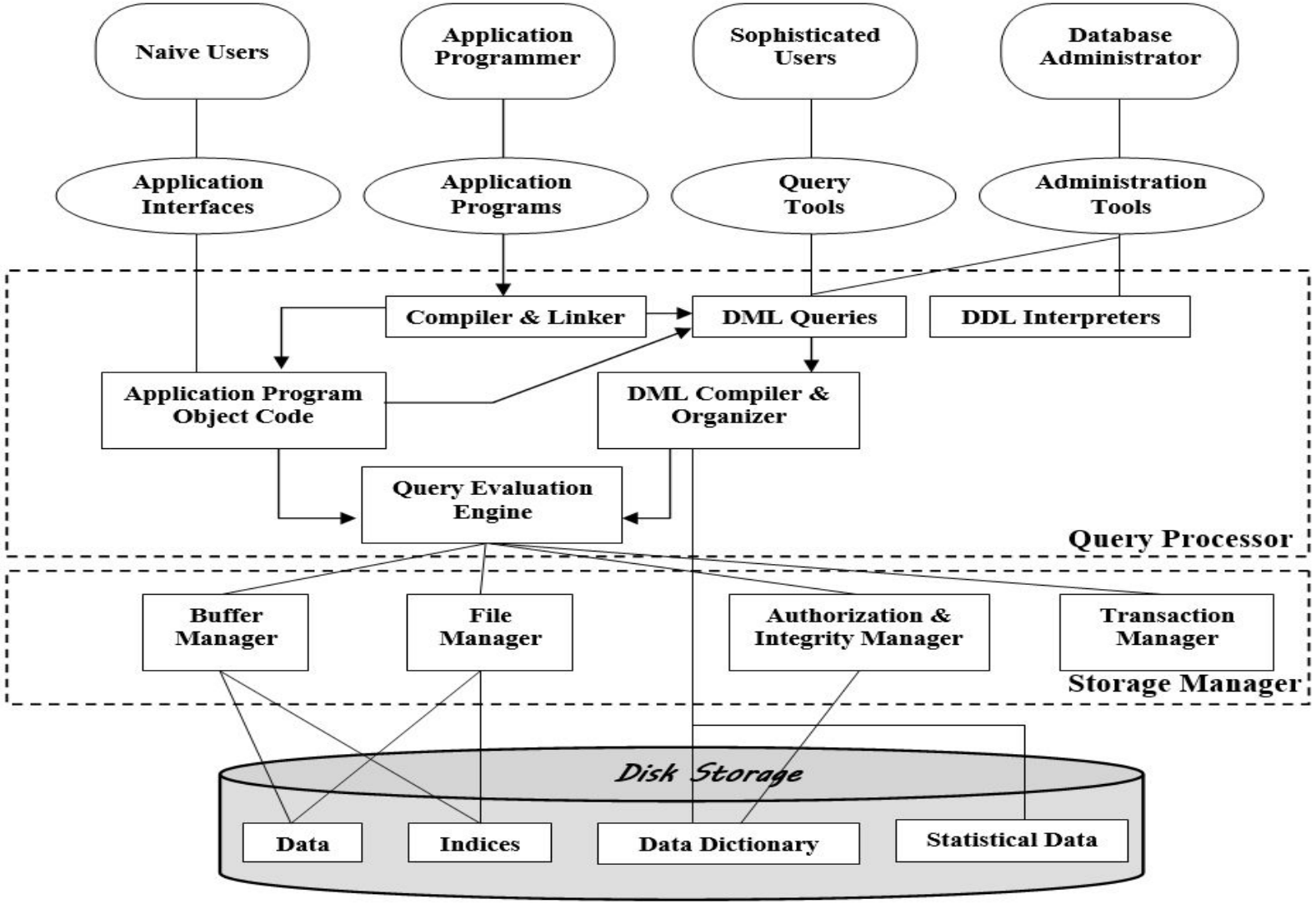


Figure: System Architecture

Query Processor Components

- DML compiler: Translates DML statements into low level instructions that query evaluation engine can understand.
- Embedded DML precompiler: Converts DML statements embedded in an application program to normal procedure calls in the host language.
- DDL interpreter : interprets DDL statements and records them in a set of tables containing metadata.
- Query Evaluation Engine: Executes low-level instructions generated by DML compiler

Storage Manager Components

- Authorization & Integrity Manager: Test satisfaction of integrity constraints and checks authority of users to access data.
- Transaction Manager: Ensures that the database remains in a consistent state despite the system failures, and concurrent transaction execution without confliction.
- File Manager: Manages the allocation of space on disk storage and data structures used to represent information.
- Buffer Manager: Responsible for fetching data from disk storage into main memory and decide what data to cache in memory.

Transaction Management

ACID Properties

- A Transaction is a collection of operations that performs a single logical function in a database application.
- Several operations on the database form a single logical unit of work.
- To ensure integrity of the data , the database system maintains the following properties of transaction:

Atomicity: Either all operations of the transaction are reflected properly or none are.

Consistency : execution of a transaction in isolation (i.e. no other transaction executing concurrently) preserves the consistency of the database.

Isolation: Even though multiple transactions execute concurrently, each transaction is unaware of other transactions executing concurrently in the system.

Durability: After a transaction completes successfully, the changes it has made to the database persist, even if there are system failures.

Storage Management

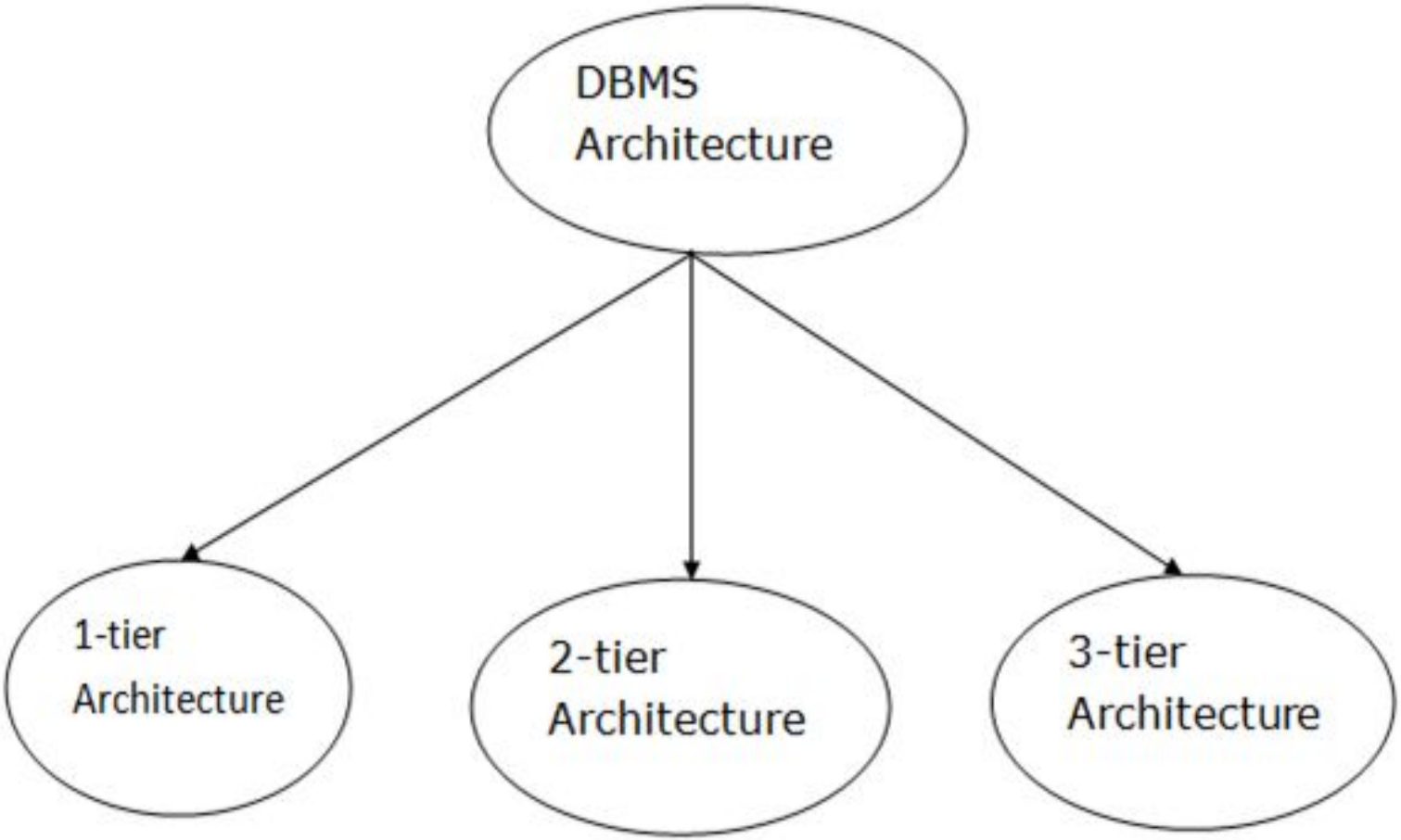
- A storage manager is a program module that provides the interface between the low-level data stored in the database and the application programs and the queries submitted to the system.
- A storage manager is responsible for the interaction with the file manager.
- The storage manager translates the various DML statements into low –level file system commands.
- Thus storage manager is responsible for storing, retrieving, and updating data in the database.

Several data structures are required as part of physical system implementation

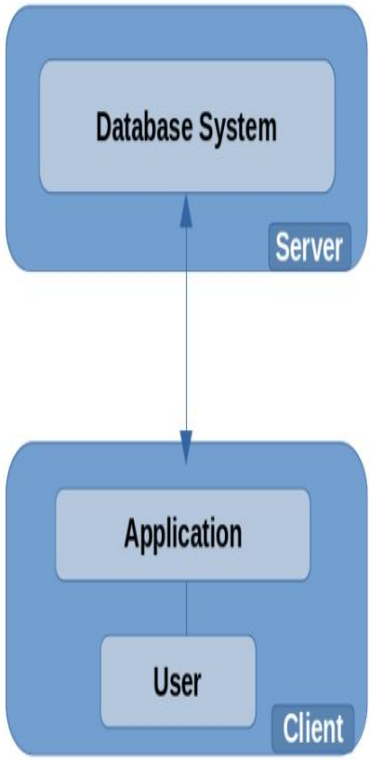
- **Data Files :** which stores the database itself
- **Data Dictionary** : which stores metadata about the structure of the database.
- **Indices :** which provide fast access to data items that hold particular values.
- **Statistical Data:** which stores statistical information about the data in the database.
- This info is used by query processor to select efficient ways to execute a query.

DBMS Application Architecture

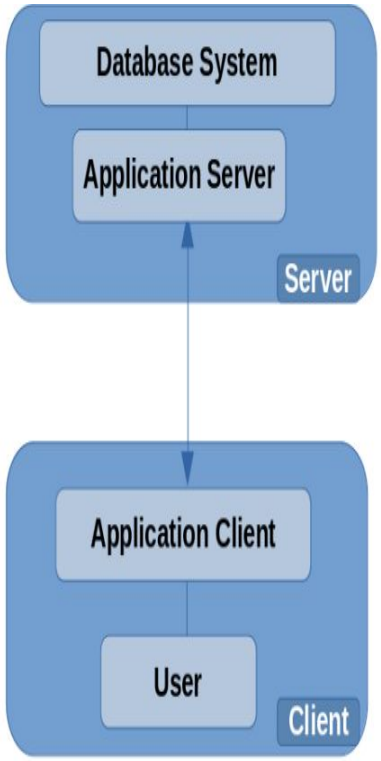
- The DBMS design depends upon its architecture.
- The basic client/server architecture is used to deal with a large number of PCs, web servers, database servers and other components that are connected with networks.
- The client/server architecture consists of many PCs and a workstation which are connected via the network.
- DBMS architecture depends upon how users are connected to the database to get their request done


DBMS
Architecture
1-tier
Architecture
2-tier
Architecture
3-tier
Architecture

Single Tier Architecture



2-Tier Architecture

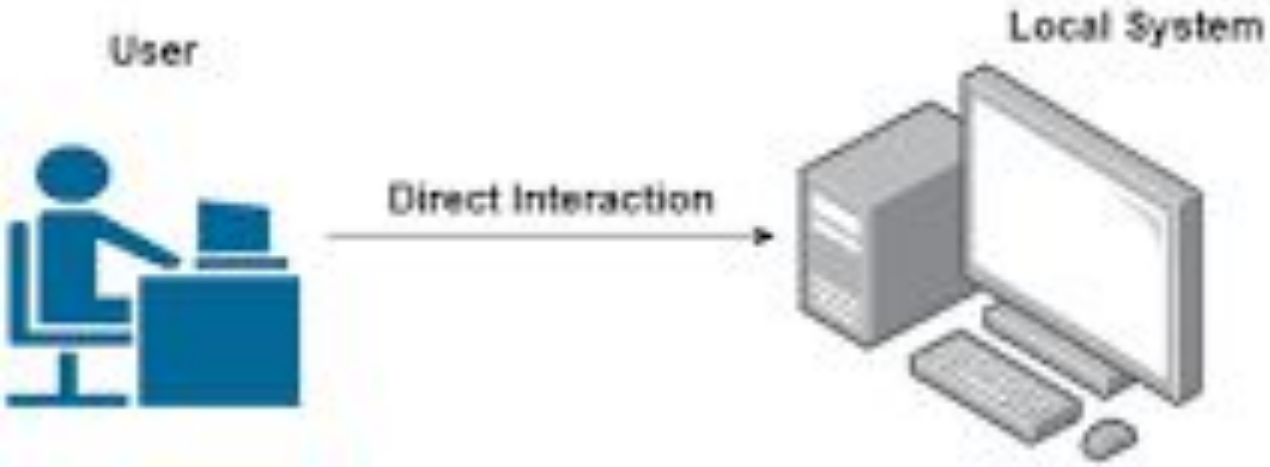


3-Tier Architecture

1-Tier Architecture

- In this architecture, the database is directly available to the user.
- It means the user can directly sit on the DBMS and uses it.
- Any changes done here will directly be done on the database itself. It doesn’t provide a handy tool for end users.
- The 1-Tier architecture is used for development of the local application, where programmers can directly communicate with the database for the quick response.

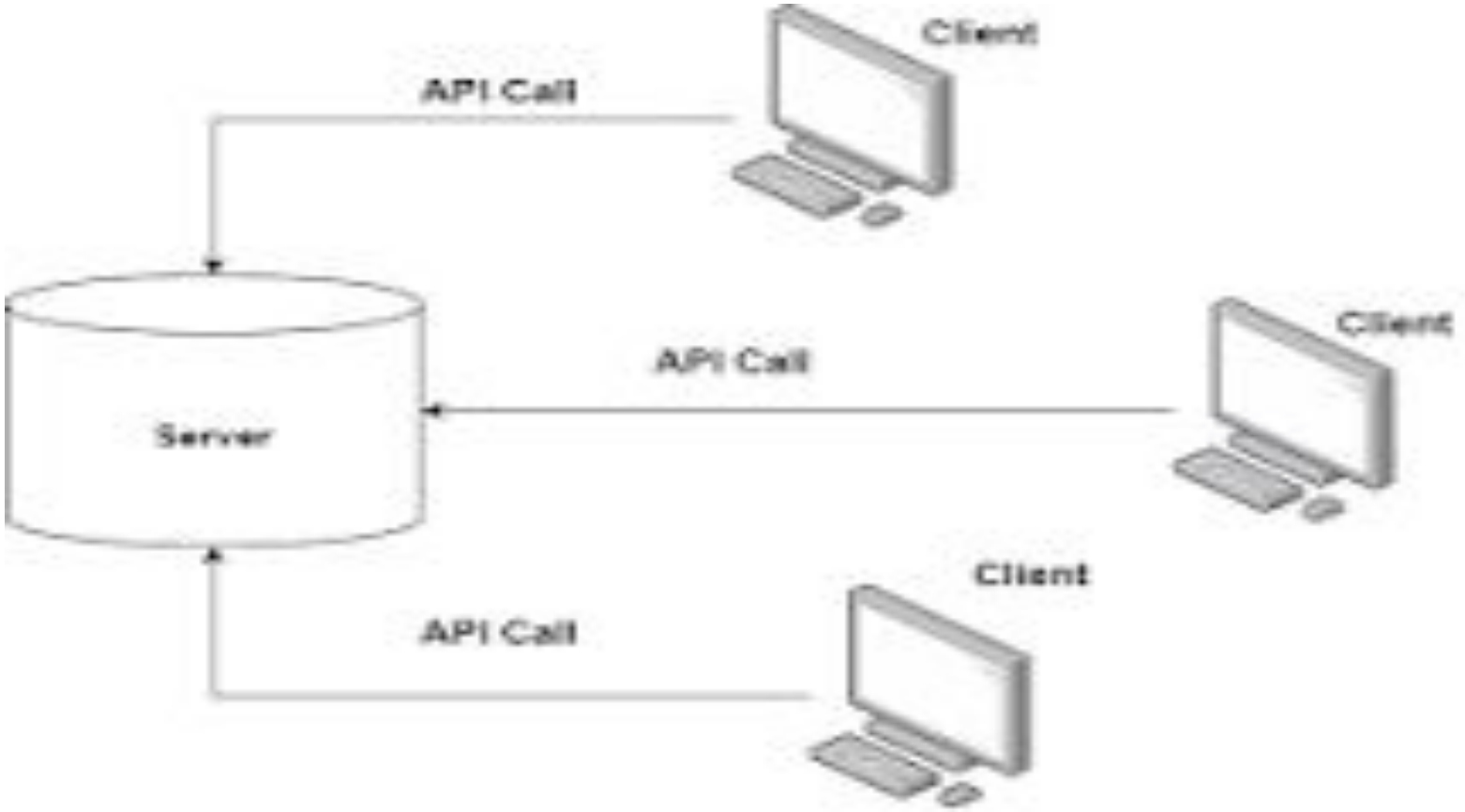
Example for 1-Tier Architecture



2-Tier Architecture

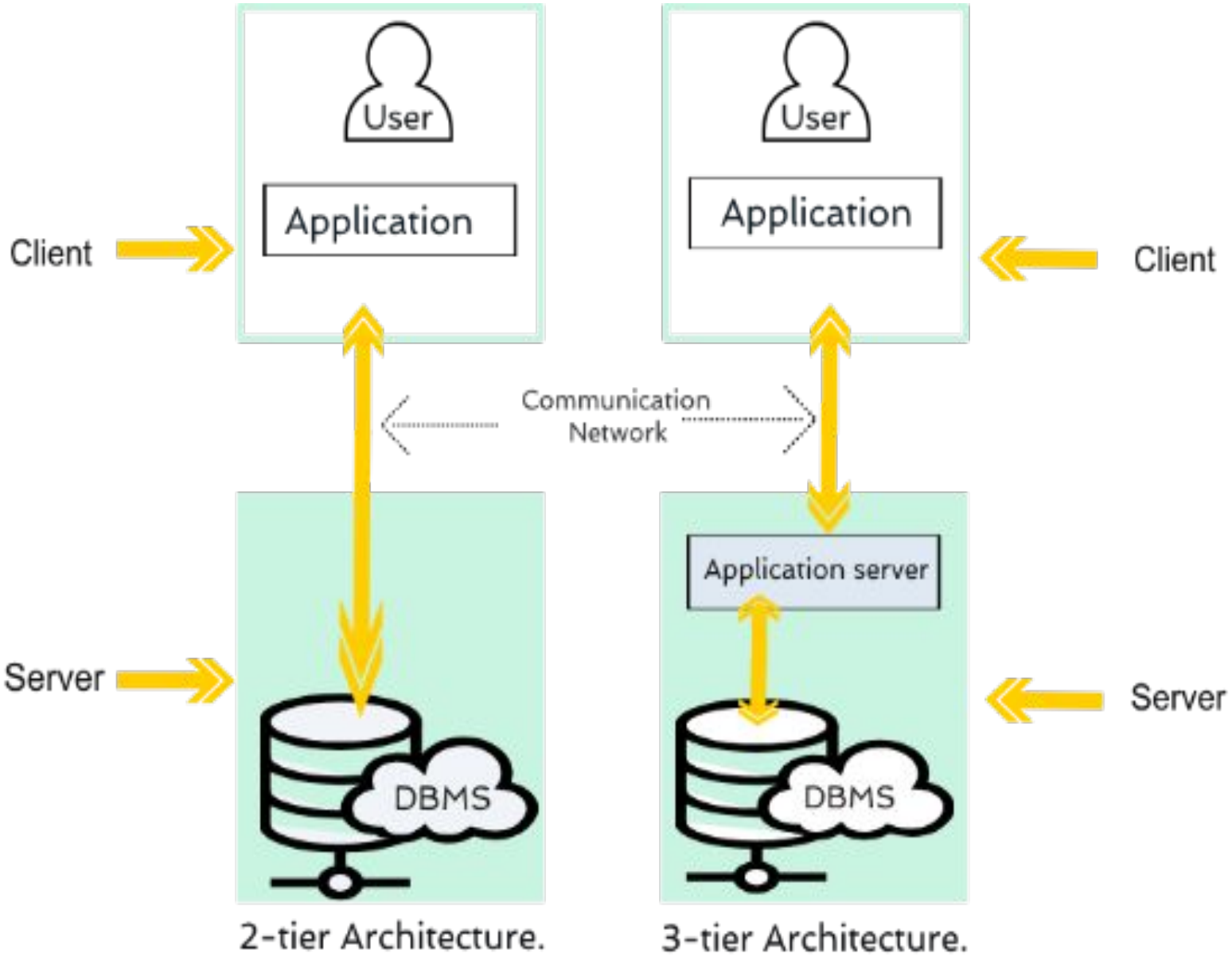
- It is same as basic client-server. But Applications on the client end can directly communicate with the database at the server side.
- For this interaction, API's like: ODBC, JDBC are used. The user interfaces and application programs are run on the client-side.
- The server side is responsible to provide the functionalities like: query processing and transaction management.
- To communicate with the DBMS, client-side application establishes a connection with the server side.

Example for 2-Tier Architecture



3-Tier Architecture

- The 3-Tier architecture contains another layer between the client and server. In this architecture, client can’t directly communicate with the server.
- The application on the client-end interacts with an application server which further communicates with the database system.
- End user has no idea about the existence of the database beyond the application server.
- The database also has no idea about any other user beyond the application.
- The 3-Tier architecture is used in case of large web application.


User
Application
Client
Server
DBMS
2-tier Architecture.
User
Application
Client
Communication
Network
Application server
Server
DBMS
3-tier Architecture.