Database Management System (DBMS)

Unit-1

BASIC CONCEPTS & DATABASE SYSTEM ARCHITECTURE

Data

- Data may be defined as a known fact that can be recorded and that have implicit meaning.
- Data are raw or isolated facts from which the required information is produced.
- Example:
  - Famous Kesar mangoes are found in Junagadh-Gujarat.
  - Saurashtra region is the largest supplier of cotton.

Data

In salesperson's view	In the view of electricity supplier	In employer's mind
Customer-name	Consumer-name	Employee-name
Customer-account	Consumer-number	Identification-number
Address	Address	Department
Mobile number	Unit consumed	Qualification
	Amount-receivable	Skill-type

Data

Employee data example:

Employee Number	Last Name	First Name	Salary	Home Town
12129	Mathew	Thomas	1200	Jamshedpur

- Data is also known as the plural of datum, which means a single piece of information.
- However, in practice, data is used as both-the singular and the plural form of the word.
- Data can also be objects such as documents, photographic images and even video segments.
- Either numbers, or characters or both can represent data.

Types of Data.

- Data can be of two types:

- **Qualitative data:**
- It is non-numerical data. For eg., the texture of the skin, the colour of the eyes, etc.

- **Quantitative data:**
- Quantitative data is given in numbers. Data in the form of questions such as “how much” and “how many”, gives the quantitative data.

Information

- Data and information are closely related and are often used interchangeably.
- Information is processed, organized or summarized data.
- It may be defined as collection of related data that when put together, communicate meaningful and useful message to a recipient who uses it, to make decision or to interpret the data to get the meaning.
- E

In salesperson's view
M/s Poojara Ltd.
22451/-
Astron Chowk
0281-254458722

Information

- The information could be the amount receivable from M/S Poojara Ltd.Or it may be average amount receivable from all the customers of Gujarat.

- **The answers to such questions are information.**

- It reduces uncertainty, reveals additional alternatives and help to take action.

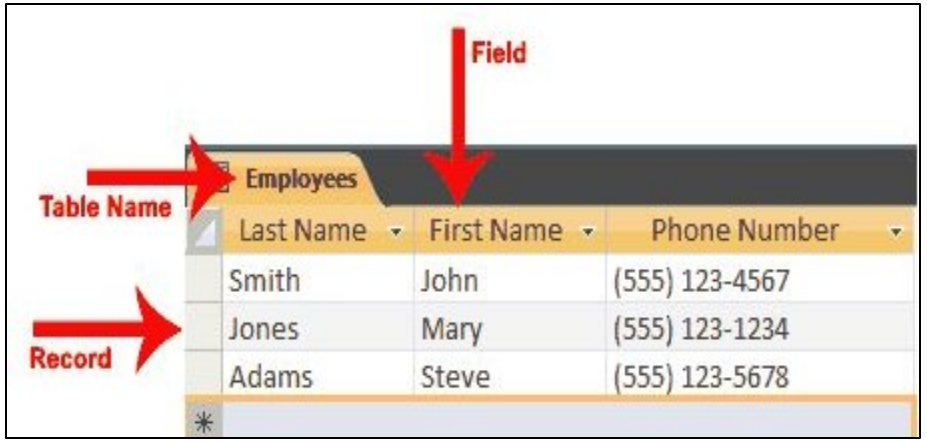
- It gives warning signals before some thing starts going wrong.
- It predicts the future with reasonable level of accuracy and helps the organization to make the best decisions.

Data Item or Fields

- A data item is the smallest unit of the data that has meaning to its user.
- It is traditionally called a field or data element.
- It is an occurrence of the smallest unit of named data.
- **Example**: Names, telephone numbers, bill amount, address and so on in a telephone bill and name, basic allowances, deductions, gross pay, net pay and so on in employee salary slip, are afew examples of data.

Employee ID	Name	Department	Basic Salary	Joining Date
1211	ABC	Database	1200	7-12-2009
1429	CDE	Testing	1100	5-4-2012
1523	GHI	Quality Analyst	1150	9-9-2014

Records



- Data is stored in the form of records.
- A record is simply a set of data stored in a table, for example, a customer record.
- A record in a database is an object that can contain one or more values.
- Groups of records are then saved in a table; the table defines the data that each record may contain.
- For example, a customer record may include items, such as first name, physical address, email address, date of birth and gender.
- A record is also known as a **tuple**.(**A tuple is one record (one row). The information in a database can be thought of as a spreadsheet, with columns (known as fields or attributes) representing different categories of information, and tuples (rows) representing all the information from each field associated with a single record.**)

Metadata

- "Data that provides information about other data" is known as metadata.
- In short we can say like: "Data about data"
- It provides information about a data or certain item's content.
- For example: An image may include metadata that describes how large the picture is, the color depth, the image resolution, when the image was created, and other data.
- Another example: A book's metadata may contain number of pages the book have, who the author is, when the book was written, and a short summary of the book.
- Most web pages include metadata in the form of 'meta tags'.
- Meta tags contains description of the web page.
- Search engines use this data when adding pages to their index.

Metadata

- Term ‘Meta’ in the IT terminology means :
  - “an underlying definition or description.“
- Each relational database system has its own mechanisms for storing metadata.
- Tables of all tables in a database, their names, sizes, and number of rows in each table.
- Tables of columns in each database, what tables they are used in, and the type of data stored in each column.
- In database terminology, this set of metadata is referred to as the catalog.

System Catalog

- A system catalog is a repository (**a place or container in which something is stored in large quantities**) of information describing the data in the database, that is the metadata (or data about the data).
- You can think of the **system catalog** as a metadata repository for your databases.
- System catalog is a system-created database that describes all database objects, data dictionary information and user access information.
- It also describes table-related data such as table names, table creators or owners, column names, data types, data size, foreign keys and primary keys, indexed files, authorized users, user access privileges and so forth.
- The system catalog is created by the database management system and the information is stored in system files, which may be queried in the same

System Catalog

- A fundamental characteristic of the database approach is that the **database system contains not only the database but also a complete definition or description of the database structure and constraints.**

- This definition is stored in the system catalog, which contains information such as the structure of each file, the type and storage format of each data item and various constraints on the data.

Files

- A file is a collection of related sequence of records.
- In many cases, all records in a file are of the same record type (each record having an identical format).
- Example: Employee File , Customer File.
- If every record in the file has exactly the same size (in bytes), the file is said to be made up of fixed-length records.
- If different records in the file have different sizes, the file is said to be made of variable-length records.

Files

- The collection of facts about a particular employee in one line or row (for example, all the fields of all the columns) of the table is an example of record.
- The collection of payroll facts for all of the employees (all columns and rows), that is, the entire table is an example of file.

Employee ID	Name	Department	Basic Salary	Joining Date	View experience/ academic certificates
1211	ABC	Database	1200	7-12-2009	Click here
1429	CDE	Testing	1100	5-4-2012	Click here

Database

- A database is defined as a collection of logically related data stored together that is designed to meet the information needs of an organization.
- It can contain one data file (a very small database) or large number of data files (a large database) depending on organizational needs.
- A database is organized in such a way that a computer program can quickly select desired pieces ofdata.
- A database can be of any size and of varying complexity.
- A database is a systematic collection of data.
- They support electronic storage and manipulation of data.
- Databases make data management easy.

Database

- Let us discuss a database example: An online telephone directory uses a database to store data of people, phone numbers, and other contact details.
- Your electricity service provider uses a database to manage billing, client-related issues, handle fault data, etc.
- Let us also consider Facebook. It needs to store, manipulate, and present data related to members, their friends, member activities, messages, advertisements, and a lot more.
- We can provide a countless number of examples for the usage of databases.

Database

- A database is designed and built for a specific purpose.
- In other words, database has some source from where data is derived, some degree of interaction with events in the real world and an audience that is actively interested in the contents of the database.
- A database is consist of the following four components as shown in the figure:
  - Data Item
  - Relationships
  - Constraints
  - Schema

Database

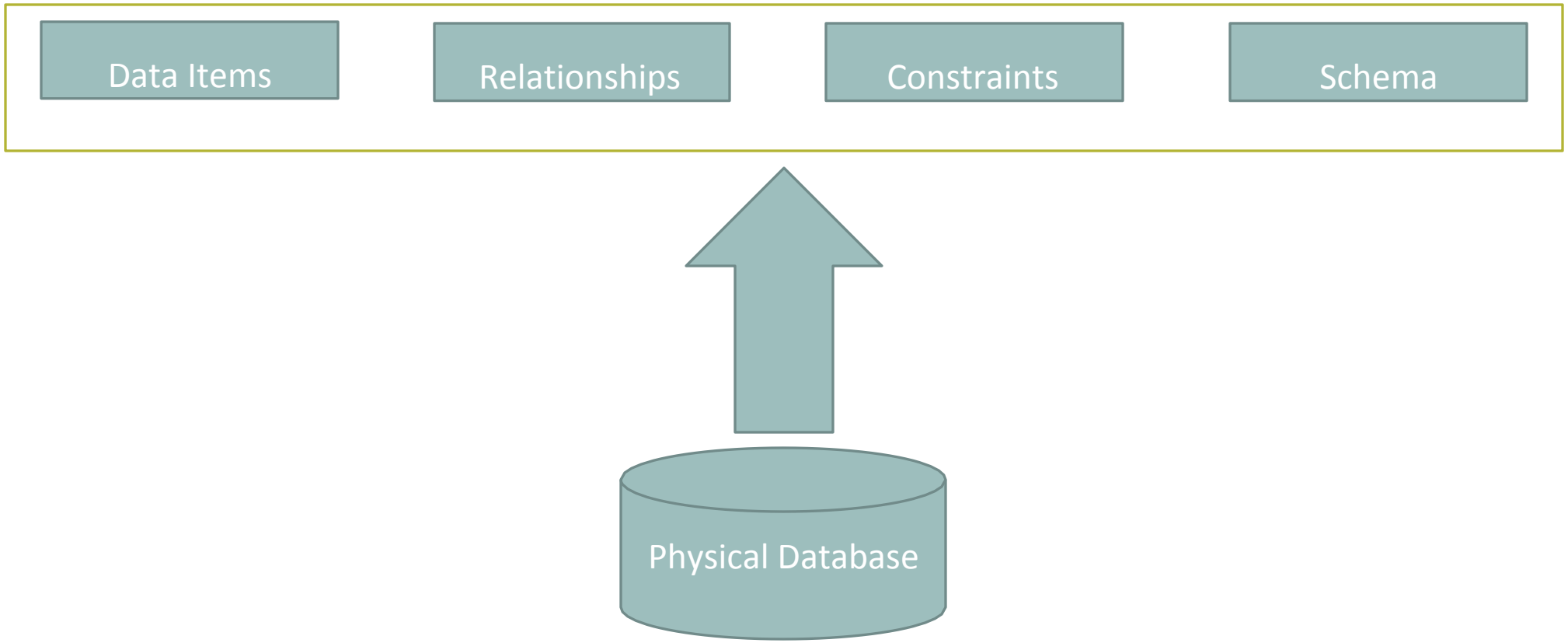


Figure: Components of database

Database

Data item is a distinct piece of information.

Relationships represent a correspondence (or communication) between various data elements.

Constraints are predicates that define correct database states.

Schema is the logical representation of a database, which shows how the data is stored logically in the entire database.

The schema does not physically contain the data itself; instead, it gives information about the shape of data and how it can be related to other tables or models.

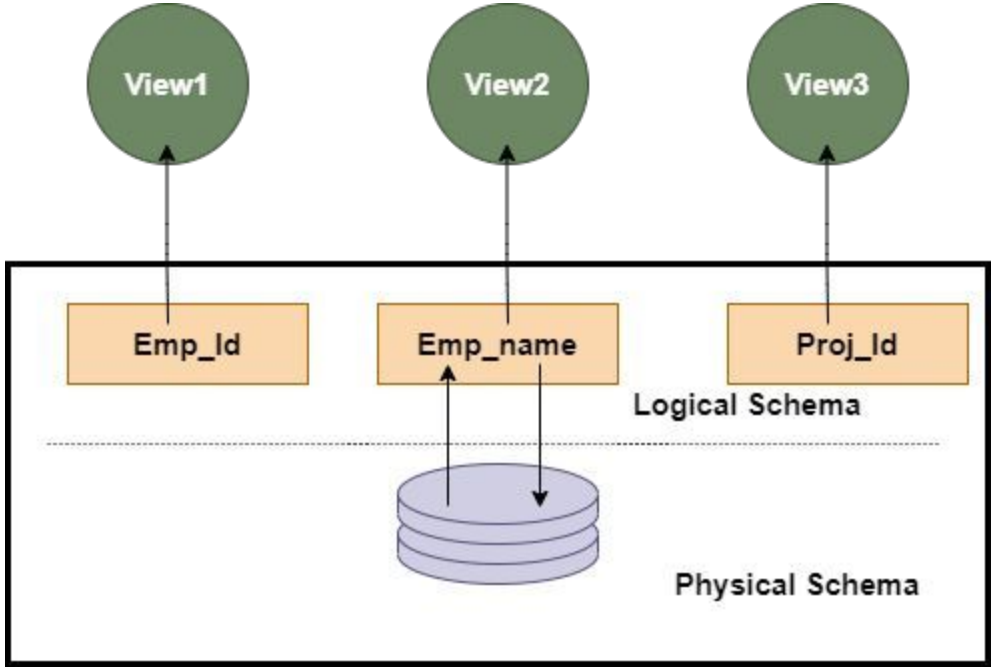
Organization of database

Types of Database Schema

The database schema is divided into three types, which are:

1. Logical Schema
2. Physical Schema
3. View Schema

Note:Write down some theory demonstrated in the lecture.



Database System/Database Management System (DBMS)

- Database system/Database Management system (DBMS) is a collection of programs that enables users to create and maintain database.
- DBMS is basically a computerized record-keeping system; which stores information and allows users to add, delete, change, retrieve and update that information on demand.
- It provides for simultaneous use of a database by multiple users and tool for accessing and manipulating the data in the database.
- It is a general-purpose software system that facilitates the process of defining (specifying the data types, structures and constraints), constructing (process of storing data on storage media) and manipulating (querying to retrieve specific data, updating to reflect changes and generating reports from the data) for

DBMS

DBMS stores data as file.

Examples: XML,
Window Registry, etc.

RDBMS

RDBMS stores data in tabular form.

Examples: MySQL,
PostgreSQL, SQL Server,
Oracle, Microsoft Access etc.

Database System/Database Management System (DBMS)…

Data description language (DDL):

It allows users to define the database, specify the data types, and data structures, and the constraints on the data to be stored in the database, usually through data definition language.

Data manipulation language (DML) and query facility:

It allows users to insert,update, delete and retrieve data from the database, usually through data manipulation language (DML). It provides general query facility through structured query language (SQL).

Software for controlled access of database:

It provides controlled access to the database, for example, preventing unauthorized user trying to access the database, providing a concurrency control system to allow shared access of the database, activating a recovery control system to restore the database to a previous consistent state following a hardware or software failure and so on.

Database system/Database Management System (DBMS)…

- The database and DBMS software together is called a database system.
- A database system overcomes the limitations of traditional file-oriented system such as, large amount of data redundancy, poor data control, inadequate data manipulation capabilities and excessive programming effort by supporting an integrated and centralized data structure.

Operations Performed on Database Systems

1. **Inserting** new data into existing data files
2. Adding new files to the database

Database system/Database Management System (DBMS)…

Data Administrator (DA)

- A data administrator (DA) is an identified individual person in the organization who has central responsibility of controlling data.
- As discussed earlier, data are important assets of an organization.
- Therefore, it is important that someone at a senior level in the organization understands these data and the organizational needs with respect to data.
- Thus, a DA is this senior level person in the organization whose job is to decide what data should be stored in the database and establish policies for maintaining and dealing with that data.
- He **decides** exactly what information is to be stored in the database, identifies the entities of the interest to the organization and the information to be recorded about those entities.
- A DA decides the content of the database at an abstract level.
- This process performed by DA is known as logical or conceptual database design.

Database system/Database Management System (DBMS)…

Data Base Administrator (DBA)

- A database administrator (DBA) is an individual person or group of persons with an overview of one or more databases who controls the design and the use of these databases.
- A DBA provides the necessary technical support for implementing policy decisions of databases.
- Thus, a DBA is responsible for the overall control of the system at technical level and unlike a DA, he or she is an IT professional.
- A DBA is the central controller of the database system who oversees and manages all the resources (such as database, DBMS and related software).
- The database administrator is supported with a number of staff or a team of people such as system programmers and other technical assistants.

Functions and Responsibilities of DBAs

1. Defining conceptual schema and database creation
2. Storage structure and access-method definition
3. Granting authorization to the users
4. Physical organisation modification
5. Routine maintenance
6. Job monitoring

1. Defining conceptual schema and database creation:

- A DBA creates the conceptual schema (using data definition language) corresponding to the abstract level database design made by data administrator (DA).

- The DBA creates the original database schema and structure of the database.

2. Storage structure and access-method definition:

- DBA decides how the data is to be represented in the stored database, the process called physical database design.

- Database administrator defines the storage structure (called internal schema) of the database (using data definition language) and the access method of the data from the database.

3. Granting authorization to the users:

- One of the important responsibilities of a DBA is the liaising with end-users to ensure availability of required data to them.
- A DBA grants access to use the database to its users.
- It regulates the usage of specific parts of the database by various users.
- DBAs assist the user with problem definition and its resolution.

4. Physical organisation modification:

- The DBA carries out the changes or modification to the description of the database or its relationship to the physical organization of the database to reflect the changing needs of the organization or to alter the physical organization to improve performance.

5. Routine maintenance:

- The DBA maintains periodical back-ups of the database, either onto hard disks, compact disks or onto remote servers, to prevent loss of data in case of disasters.
- It ensures that enough free storage space is available for normal operations and upgrading disk space as required.
- A DBA is also responsible for repairing damage to the database due to misuse or software and hardware failures.
- DBAs define and implement an appropriate damage control mechanism involving periodic unloading or dumping of the database to backup storage device and reloading the database from the most recent dump whenever required.

6. Job monitoring:

- DBAs monitor jobs running on the database and ensure that performance is not degraded by very expensive tasks submitted by some users.
- With change in requirements (for example, re-organizing of database), DBAs are responsible for making appropriate adjustment or tuning of the database.

File oriented system

- With increased complexity of business requirements, gradually they were introduced into the business applications.
- The manual method of filing systems of an organization, such as to hold all internal and external correspondence relating to a project or activity, client, task, product, customer or employee, was maintaining different manual folders.
- These files or folders were labelled and stored in one or more cabinets or almirahs under lock and key for safety and security reasons.
- As and when required, the concerned person in the organization used to search for a specific folder or file serially starting from the first entry.
- Alternatively, files were indexed to help locate the file or folder more quickly.
- Ideally, the contents of each file folder were logically related.
- For example, a file folder in a supplier's office might contain customer data; one file folder for each customer.

File oriented system

- All data in that folder described only that customer's transaction.
- File-oriented systems were an early attempt to computerize the manual filing system that we are familiar with. Because these systems performed normal recordkeeping functions, they were called data processing (DP) systems.
- Each table represents a file in the system, for example, PRODUCT file, CUSTOMER file, SALES file and so on. Each row in these files represents a record in the file.

Draw a figure demonstrated

It can be seen from the above examples that there is significant amount of duplication of data storage in different departments (for example, CUST-ID and PROD-ID), which is generally true with file-oriented system.

Advantages and Disadvantages of file oriented system

Advantages

- Although the file-oriented system is now largely obsolete, following are the several advantages of learning file-based systems:
  - It provides a useful historical perspective on how we handle data.
  - The characteristics of a file-based system helps in an overall understanding of design complexity of database systems.
  - Understanding the problems and knowledge of limitation inherent in the file-based system helps avoid these same problems when designing database systems and thereby resulting in smooth transition.

Advantages and Disadvantages of file oriented system…

Disadvantages

Data redundancy (or duplication):

Since a decentralized approach was taken, each department used their own independent application programs and special files of data.

This resulted into duplication of same data and information in several files, for example, duplication of PRODUCT-ID data in both PRODUCT and SALES files, and CUST-ID data in both CUSTOMER and SALES files.

This redundancy or duplication of data is wasteful and requires additional or higher storage space, costs extra time and money, and requires increased effort to keep all files up-to-date.

Advantages and Disadvantages of file oriented system…

Disadvantages

Data inconsistency (or loss of data integrity):

Data redundancy also leads to data inconsistency (or loss of data integrity), since either the data formats may be inconsistent or data values (various copies of the same data) may no longer agree or both.

Program-datadependence:

As we have seen, file descriptions (physical structure, storage of the data files and records) are defined within each application program that accesses a given file.

For example, "Account receivable program" accesses both CUSTOMER file and SALES file.

Therefore, this program contains a detailed file description for both these files.

As a consequence, any change for a file structure requires changes to the file description for all programs that access the file.

Advantages and Disadvantages of file oriented system…

Disadvantages

Poor data control:

A file-oriented system being decentralized in nature, there was no centralized control at the data element (field) level.

It could be very common for the data field to have multiple names defined by the various departments of an organization and depending on the file it was in.

This could lead to different meanings of a data field in different context, and conversely, same meaning for different fields. This leads to a poor data control, resulting in a big confusion.

Advantages and Disadvantages of file oriented system…

Excessive programming effort:

There was a very high interdependence between program and data in file-oriented system and therefore an excessive programming effort was required for a new application program to be written.

Even though an existing file may contain some of the data needed, the new application often requires a number of other data fields that may not be available in the existing file.

As a result, the programmer had to rewrite the code for definitions for needed data fields from the existing file as well as definitions of all new data fields.

Database approach

- The database approach represents the change in the way **end user** data are stored, accessed and managed.
- It emphasizes the integration and sharing of data throughout the organization.
- Database systems overcome the disadvantages of file-oriented system.

- Diiference between them (student side)

Database System Environment

- Database environment consist of four main parts:

1. Data
2. Hardware
3. Software
4. Users (people)

1. **Data:**

- From the user's point of view, the most important component of database system is perhaps the
  data.
- These data in a database are both integrated and shared in a system.
- Data integration means that the database can be thought of as a function of several otherwise distinct files.
- Whereas in data sharing, individual pieces of data in the database can be shared among different users and each of those users can have access to the same piece of data, possibly for different purposes.
- Different users can effectively even access the same piece of data concurrently (at the same

Database System Environment…

2. Hardware:

- All the physical devices of a computer are termed as hardware.
- With a large number of users, a very large amount of main memory and disk space is required to maintain and control the huge quantity of data stored in a database.
- In addition, high-speed computers, networks and peripherals are necessary to execute the large number of data access required to retrieve information in an acceptable amount of time.

3. Software:

- Software is the basic interface (or layer) between the physical database and the users.
- It is most commonly known as database management system (DBMS).
- It comprises the application programs together with the operating system software.
- All requests from the users to access the database are handled by DBMS.

Database System Environment…

4. Users:

The users are the people interacting with the database system in any form.

There could be various categories of users.

The first category of users is the application programmers who write database application programs in some programminglanguage.

The second category of users is the end users who interact with the system from online workstations

The third category of users is the database administrators (DBAs), who manage the DBMS and its proper functioning.

The fourth category of users is the database designers who design the database structure.

Advantages of DBMS

Due to the centralized management and control, the database management system (DBMS) has numerous advantages. Some of these are as follows:

a). Minimal data redundancy:

- In a database system, views of different user groups (data files) are integrated during database design into a single, logical, centralized structure.
- Each primary fact is ideally recorded in only one place in the database. The total data storage requirement is effectively reduced.
- Incidentally, we do not mean or suggest that all redundancy can or necessarily should be eliminated.
- Sometimes there are sound business and technical reasons for maintaining multiple copies of the same data.

Advantages of DBMS…

b). Efficient data access:

- DBMS utilizes a variety of sophisticated techniques to store and retrieve data efficiently.
- This feature is especially important if the data is stored on external storage devices.

c). Improved data sharing:

Since, database system is a centralized repository of data belonging to the entire organization it can be shared by all authorized users.

Existing application programs can share the data in the database.

Furthermore, new application programs can be developed on the existing data in the database to share the same data and add only that data that is not currently stored, rather having to define all data requirements again.

Advantages of DBMS…

d). Improved data consistency:

- In database system, such inconsistencies are avoided to some extent by making them known to DBMS.
- DBMS ensures that any change made to either of the two entries in the database is automatically applied to the other one as well.

e). Improved data integrity:

- Integrity is usually expressed in terms of constraints, which are consistency rules that the database system should not violate.
- Another integrity check can be incorporated in the database to ensure that if there is reference to a certain object, that object must exist.
- For example, in the case of bank's automatic teller machine (ATM), a user is not allowed to transfer fund from a non-existent saving to a checking account.

Advantages of DBMS…

f). Improved security:

- Database security is the protection of database from unauthorized users.
- The database administrator (DBA) ensures that proper access procedure is followed, including proper authentication schemes for access to the DBMS and additional checks before permitting access to sensitive data.
- Different levels of security could be implemented for various types of data and operations.
- The access of data by authorized user may be restricted for each type of access (for example, retrieve, insert, modify, update, delete and so on) to each piece of information in the database.

- **g). Enforcement of standards:**
- With central control of the database, a DBA defines and enforces the necessary standards.
- Applicable standards might include any or all of the following: departmental, installation, organizational, industry, corporate, national or international.

Advantages of DBMS…

- Standards can be defined for data formats to facilitate exchange of data between systems, naming conventions, display formats, report structures, terminology, documentation standards, update procedures, access rules and so on.

Advantages of DBMS…

- **h). Economy of scale:**

- Centralizing of all the organization's operational data into one database and creating a set of application programs that work on this source of data resulting in drastic cost savings.

- This enables the whole organization to invest in more powerful processors, storage devices or communication gear, rather than having each department purchase its own (low-end) equipment.

- **i). Reduced program maintenance:**

- The problems of high maintenance effort required in file-oriented system, are reduced in database system.

- In a file-oriented environment, the descriptions of data and the logic for accessing data are built into individual application programs.

- As a result, changes to data formats and access methods inevitably result in the need to modify application programs. In database environment, data are more independent of the application programs.

Advantages of DBMS…

j). Improved backup and recovery services:

- DBMS provides facilities for recovering from hardware or software failures through its back up and recovery subsystem.
- For example, if the computer system fails in the middle of a complex update program, the recovery subsystem is responsible and makes sure that the database is restored to the state it was in before the program started executing.
- Alternatively, the recovery subsystem ensures that the program is resumed from the point at which it was interrupted so that its full effect is recorded in the database.

Disadvantages of DBMS

In spite of the advantages, the database approach entails some additional costs and risks that must be recognized and managed when implementing DBMS.

1. **Increased complexity:**

- A multi-user DBMS becomes an extremely complex piece of softwaredue to expected functionality from it.
- It becomes necessary for database designers, developers, database administrators and end- users to understand this functionality to full advantage of it.
- Failure to understand the system can lead to bad design decisions, which can have serious consequences for an organization.

Disadvantages of DBMS…

2. Requirement of new and specialized manpower:

- Because of rapid changes in database technology and organization's business needs, the organization's need to hire, train or retrain its manpower on regular basis to design and implement databases, provide database administration services and manage a staff of new people.
- Therefore, an organization needs to maintain specialized skilled manpower.

3. Large size of DBMS:

- The large complexity and wide functionality makes the DBMS an extremely large piece of software.
- It occupies many gigabytes of storage disk space and requires substantial amounts of main memory to run efficiently.

Disadvantages of DBMS…

4. Increased installation and management cost:

- The large and complex DBMS software has a high initial cost.
- It requires trained manpower to install and operate and also has substantial annual maintenance and support costs.
- Installing such a system also requires upgrades to the hardware, softwareand c d o a m ta munications systems in the organization.
- Substantial training of manpower is required on an ongoing basis to keep up with new releases and upgrades.
- Additional or more sophisticated and costly database software may be needed to provide security and to ensure proper concurrent updating of shared data.

Disadvantages of DBMS…

5. **Conversion cost:**

The cost of conversion (both in terms of money and time) from legacy system (old file-oriented and/or older database technology) to modern DBMS environment is very high.

In some situations, the cost of DBMS and extra hardware may be insignificant compared with the cost of conversion.

This cost includes the cost of training manpower (staff) to use these new systems and cost of employing specialists manpower to help with the conversion and running of the system.

6. **Need for explicit backup and recovery:**

For a centralized shared database to be accurate and available all times, a comprehensive procedure is required to be developed and used for providing backup copies of data and for restoring a database when damage occurs.

A modern DBMS normally automates many more of the backup and recovery tasks than a file-oriented system.

Transaction Management

- All work that logically represents a single unit is called transaction.
- A transaction can update a record, delete a record, modify a set of records and so on. When the DBMS does a 'commit', the changes made by transaction are made permanent.
- If the changes are not be made permanent, the transaction can be 'rollback' and the database will remain in its original state.
- When updates are performed on a database, we need some way to guarantee that a set of updates will succeed all at once or not at all.
- This is necessary in order to keep the database in a consistent state.
- For example, a transaction might involve transferring money from a bank saving account of a person to a checking account.
- While this would typically involve two separate database operations.

Transaction Management…

- First a withdrawal from the savings account and then a deposit into the checking account.
- It is logically considered one unit of work.
- It is not acceptable to do one operation and not the other operation because that would violate integrity of the database.
- Thus, both withdrawal and deposit must be completed (committed) or partial transaction must be aborted (rolled-back), so that uncompleted work does not affect database.
- Another example: Railway reservation.
- Transaction has, generally, following four properties, called ACID:

  (1). Atomicity (2). Consistency

  (3). Isolation (4). Durability

Transaction Management…

1). Atomicity:

- Atomicity means that either all the work of a transaction or none of it is applied.
- With atomicity property of the transaction, other operations can only access any of the rows involved in transactional access either before the transaction occurs or after the transaction is complete, but never while the transaction is partially complete.

2.) Consistency:

- Consistency means that the transaction's work will represent a correct (or consistent) transformation of the database's state.

3). Isolation:
Isolation requires that a transaction not to be influenced by changes made by other concurrently executing transactions.

4).Durability:

Durability means that the work associated with a successfully completed transaction is applied to the database and is guaranteed to survive system or media failures.

Data Warehouse

- In computing, a data warehouse (DW or DWH), also known as an enterprise data warehouse (EDW), is a system used for reporting and data analysis, and is considered a core component of business intelligence.
- DWs are central repositories of integrated data from one or more disparate (different) sources to support management in decision making.
- They store current and historical data in one single place that are used for creating analytical reports for workers throughout the organization.
- The data stored in the warehouse is uploaded from the operational systems(such as marketing or sales).
- The data may pass through an operational data store and may require data cleansing for additional operations to ensure data quality before it is used in the DW for reporting.

Data dictionary

- It is a repository of information about a database.
- The data dictionary is an integral part of the database management systems (DBMSs) and stores metadata, or information about the database, attribute names and definitions for each table in the database.
- Data dictionaries aid the database administrator in the management of a database.
- Data dictionary is usually a part of the system catalog that is generated for each database.
- A useful data dictionary system usually stores and manages the following types of information:
  - Description of the database users, their responsibilitiesand their access rights.
  - Usage statistics such as frequencies of queries and transactions and access counts to different portions of the database.

Data

Let us take an example of a manufacturing company M/s ABC [illegible] activities related to various departments.

INVENTORY

MOD-NO	MOD-NAME	MOD-DESC	UNIT-PRICE
L-800	Legend	Luxury car	4000000
M-1000	Maharaja	Luxury car	3000000
C-1200	Cruze	Zip drive	1200000
P-2000	Panthera	Sports ride	800000
R-121	Rover	Sports ride	2000000

EMPLOYEE

EMP-NO	EMP-LNAME	EMP-FNAME	EMP-SALARY
106519	Mathew	Thomas	4000
112233	Smith	John	4500
123456	Kumar	Rajeev	6000
123243	Martin	Jose	3500

Data dictionary…

As it can be seen from below figure that all data fields of both the files are included in field's file and both the files (INVENTORY and EMPLOYEE) in the file's file.

Data dictionary of M/s ABC Motors Limited

Field's File

FIELD-NAME	FIELD-TYPE	FIELD-LENGTH
MOD-NO	CHAR	6
MOD-NAME	ALPHA	10
MOD-DESC	ALPHA	15
UNIT-PRICE	NUMERIC	7
EMP-NO	NUMERIC	6
EMP-LNAME	ALPHA	10
EMP-LNAME	ALPHA	15
EMP-SALARY	NUMERIC	4

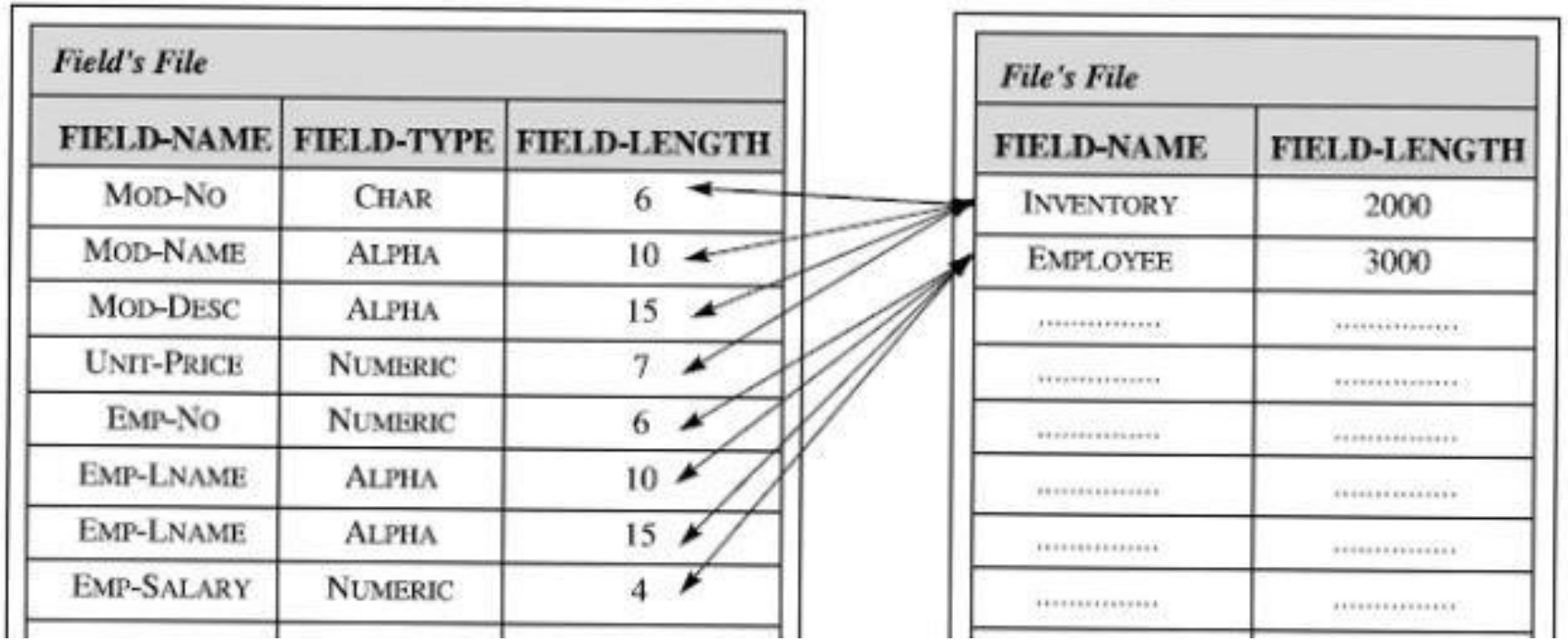
File's File

FIELD-NAME	FIELD-LENGTH
INVENTORY	2000
EMPLOYEE	3000
.....	.....
.....	.....
.....	.....
.....	.....
.....	.....
.....	.....

Data dictionary…

Data dictionary also keeps track of the relationships among the entities, which is important in the data processing environment as how these entities interrelate.

Below figure shows the links (relationship) between fields and files.



Field's File

FIELD-NAME	FIELD-TYPE	FIELD-LENGTH
MOD-NO	CHAR	6
MOD-NAME	ALPHA	10
MOD-DESC	ALPHA	15
UNIT-PRICE	NUMERIC	7
EMP-NO	NUMERIC	6
EMP-LNAME	ALPHA	10
EMP-LNAME	ALPHA	15
EMP-SALARY	NUMERIC	4

File's File

FIELD-NAME	FIELD-LENGTH
INVENTORY	2000
EMPLOYEE	3000
.....	.....
.....	.....
.....	.....
.....	.....
.....	.....
.....	.....

Components of data dictionaries

- Data dictionary contains the following components:
  1. Entities
  2. Attributes
  3. Relationships
  4. Key

1. <u>Entities</u>

- Entity is the real physical object or an event; the user is interested in keeping track of.
- In other words, any item about which information is stored is called entity.
- Thomas Mathew is a real living person and an employee of M/s ABC Motors Ltd., is an entity for which the company is interested in keeping track of the various details or facts.
- Maharaja model car (Model no. M-1000) is a real physical object manufactured by M/s ABC Motors Ltd., is an entity.

Components of data dictionaries…

- A collection of the entities of the same type, for example “all” of the company’s employees are called an entity set.
- In other words, we can say that, a record describes the entity and a file describes an entity set.

INVENTORY

MOD-NO	MOD-NAME	MOD-DESC	UNIT-PRICE
L-800	Legend	Luxury car	4000000
M-1000	Maharaja	Luxury car	3000000
C-1200	Cruze	Zip drive	1200000

EMPLOYEE

EMP-NO	EMP-LNAME	EMP-FNAME	EMP-SALARY
106519	Mathew	Thomas	4000
112233	Smith	John	4500
123456	Kumar	Rajeev	6000

Components of data dictionaries…

2. Attributes

An attribute is a property or characteristic (field) of an entity.
Mathew's EMP-NO, EMP-SALARY and so forth, all are his attributes.
In other words, we can say that, values in all the fields are

attributes. Below table shows an example of an entity set and its

attributes:

	Entity set	Attributes
(*a*)	INVENTORY	MOD-NO
		MOD-NAME
		MOD-DESC
		UNIT-PRICE
(*b*)	EMPLOYEE	EMP-NO
		EMP-LNAME
		EMP-FNAME
		EMP-SALARY

Components of data dictionaries…

3. Relationships

- The associations or the ways that different entities relate to each other is called relationships.
- The relationship between any pair of entities of a data dictionary can have value to some part or department of the organization.
- Some data dictionaries define limited set of relationships among their entities, while others allow the relationship between every pair of entities.

Relationships could be of followingtypes:

- One-to-one (1:1) relationship
- One-to-many (1:m) relationships
- Many-to-many (m:m) relationships

Components of data dictionaries…

- Employee number allocated to each employee is unique and each employee number is allocated to an individual employee. This is called one-to-one relationship (1:1) relationship.

(*a*) One-to-one relationship

- While for a given employee there is one manufacturing department, in the manufacturing department there may be many employees.
- Thus, in this case, there is one-to-one relationship in one direction and a multiple association in the other direction.
- This combination is called one-to many (1:m) relationship

Components of data dictionaries…

(*b*) One-to-many relationship

- Finally, consider the situation in which an employee gets a particular salary.
- While for a given employee there is one salary amount (for example, 4000), the same amount may be given to many employees in the department.
- In this case, there is multiple associations in both the direction, and this combination is called many-to-many (m:m) relationship.

(*c*) Many-to-many relationship

Components of data dictionaries…

4. Key

- The data item (or field) for which a computer uses to identify a record in a database system is referred to as key.
- In other words, key is a single attribute or combination of attributes of an entity set that is used to identify one or more instances of the set.
- There are various types of keys:

1. Primary key
2. Concatenated key
3. Secondary key
4. Super key

Components of data dictionaries…

1. **Primary key**

   Primary key is used to uniquely identify a

   record. It is also called entity identifier.

   Example: EMP-NO in the EMPLOYEE file and MOD-NO in the INVENTORY file.

2. **Concatenated key**

   When more than one data item is used to identify a record, it is called concatenated key.

   Example: EMP-NO and EMP-FNAME in EMPLOYEE file and both MOD-NO and MOD-TYPE in INVENTORY file.

Components of data dictionaries…

3. Secondary key

- Secondary Key is the key that has not been selected to be the primary key.
- However, it is considered a candidate key for the primary key.
- Therefore, a candidate key not selected as a primary key is called secondary key.
- Candidate key is an attribute or set of attributes that you can consider as a Primary k

Student_ID	Enrollment	Student_Name	Age	Student_Email
27	9122717	Manish	20	abc@gmail.com
54	9122655	Manan	21	def@gmail.com
81	9122699	Mohan	25	ghi@gmail.com

Components of data dictionaries…

4. Super key:

- A super key is a combination of all possible attribute which can uniquely identify the rows (tuples) in a table.
- This means that a super key may have some extra attribute which isn't necessary for uniquely identifying the rows in the table.

Roll_no	Name	Registration_no
1	Andrew	895
2	Angel	564
3	Augusto	567

Components of data dictionaries…

1. {Roll_no}
2. {Registration_no}
3. {Roll_no, Registration_no}
4. {Roll_no, Name}
5. {Name, Registration_no}
6. {Roll_no, Name, Registration_no}

- All the above keys are able to uniquely identify each row.
- So, each of these keys is super key.
- Here you can see that by using Roll_no only, we can uniquely identify the rows but if you are making a super key, then you will try to find all the possible cases of keys that can be used to identify data uniquely.

Active and Passive data dictionaries

Active data dictionary:

- A dictionary is called active if it is checked by DBMS every time a data base is accessed.
- It is always consistent with actual data base structure.
- It is automatically maintained by the DBMS.
- Hence there will not be any mismatch between the actual structure and the data dictionary details.
- Such data dictionary is called active data dictionary.

Passive data dictionary:

- It is also known as non-integrated data dictionary.
- This is normally used for only documentation purpose.
- User of the system can control passive data dictionary and it is modified manually when the database structure change.
- In such case, an effort is required to keep data dictionary in sync with the database objects.
- In this case, there is a chance of mismatch with the database objects and the data dictionary.

Database System Architecture

Introduction

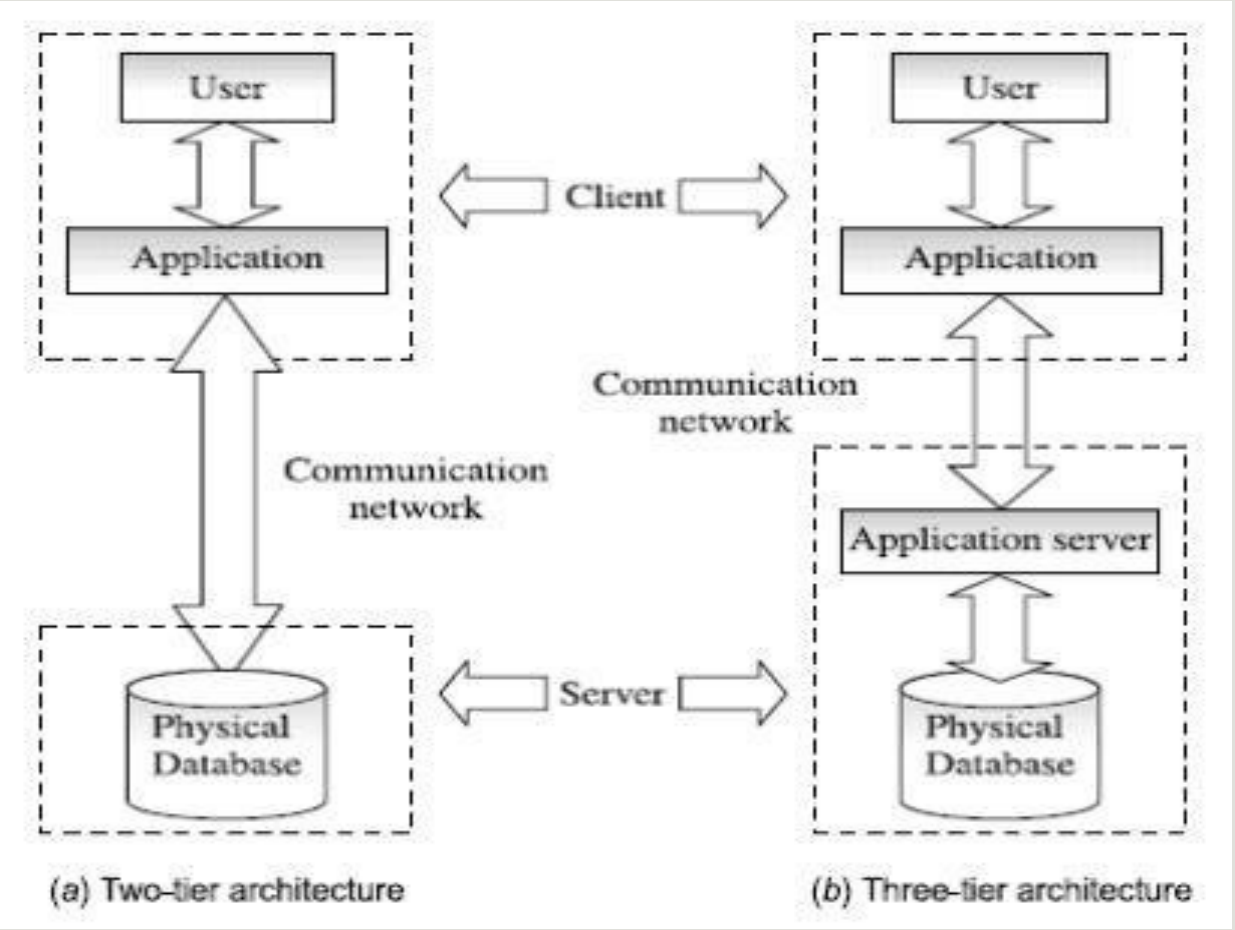
- An organization requires an accurate and reliable data and efficient database system for effective decision-making.
- Users of a database system in the organization look for an abstract view of data they are interested in.
- Furthermore, since database is a shared resource, each user may require a different view of the data held in the database.
- To satisfy these needs, we need to develop architecture for the database systems.
- The database architecture is a framework in which the structure of the DBMS is described.

Introduction…

- The database applications are usually portioned into a two-tier architecture or a three-tier architecture.

- In a two-tier architecture, the application is partitioned into a component that resides at the client machines, which evokes database system functionality at the server machine through query language statements.
- Application program interface standards are used for interaction between the client and the server.
- In a three-tier architecture, the client machine acts as merely a front-end and does not contain any direct database calls.
- Instead, the client end communicates with an application server, usually through a forms interface.
- The application server in turn communicates with a database system to access data.

Figure of two tier architecture and three tier architecture…



Schemas, Sub-schemas and Instances

Introduction

- The data in the database changes frequently, while the plans remain the same over long periods of time (although not necessarily forever).
- The database plans consist of types of entities that a database deals with, the relationships among these entities.
- The users' view of the data (also called logical organization of data) should be in a form that is most convenient for the users and they should not be concerned about the way data is physically organized.
- Therefore, a DBMS should do the translation between the logical (users' view) organization and the physical organization of the data in the database.

Schema

- It is a framework into which the values of the data items (or fields) are fitted.
- In other terms, schema mean an overall plan of all the data item (field) types and record types stored in a database.
- Schema gives the names of the entities and attributes.
- It specifies the relationship among them.
- The plans or the format of schema remains the same but, the values fitted into this format changes from instance to instance.
- Schema includes the definition of the database name, the record type and the components that make up those records.
- Let us look an example of M/s ABC, a manufacturing company
- The structure of the database consisting of three tables namely, PRODUCT, CUSTOMER and SALES files is the schema of the database.

Schema…

PRODUCT		
PROD-ID	PROD-DESC	UNIT-COST

CUSTOMER				
CUST-ID	CUST-NAME	CUST-STREET	CUST-CITY	CUST-BAL

SALES			
CUST-ID	PROD-ID	PROD-QTY	PROD-PRICE

Schema…

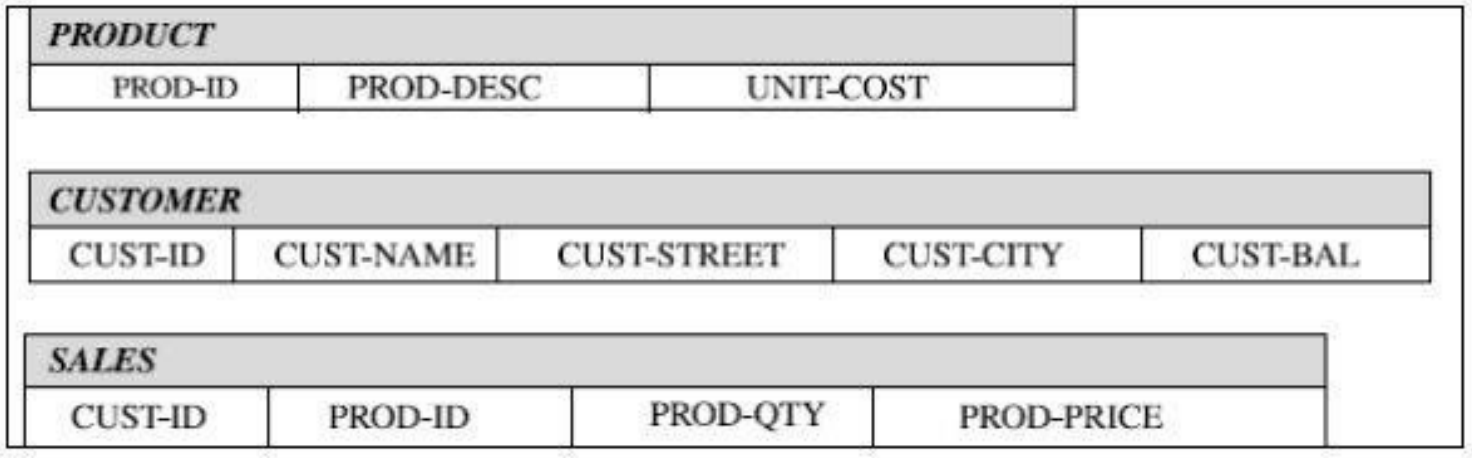
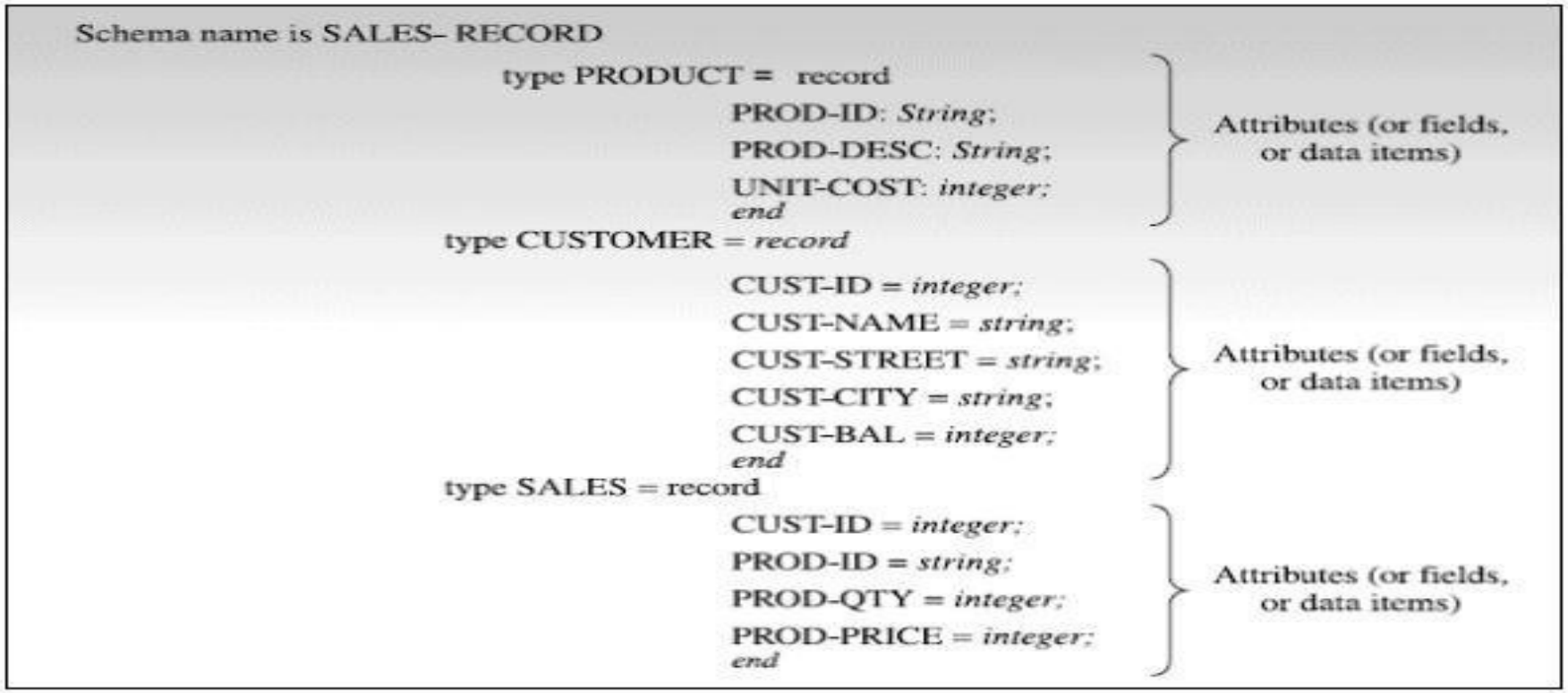


Fig: Schema diagram for sales record database of M/s ABC company.

- The structure of the database consisting of three files (or tables) namely, PRODUCT, CUSTOMER and SALES files is the schema of the database.
- A database schema corresponds to the variable declarations (along with associated type definitions) in a program.

Schema…

- The schema diagram displaysthe structure of each record type but not the actual instances of records.
- Each object schema cor



Types of schema

1) Physical schema
2) Logical schema

1). Physical schema:

- The physical schema deals with the manner in which the conceptual database shall get represented in the computer as a stored database.
- The physical schema is hidden beneath the logical schema and can usually be changed easily without affecting application programs.

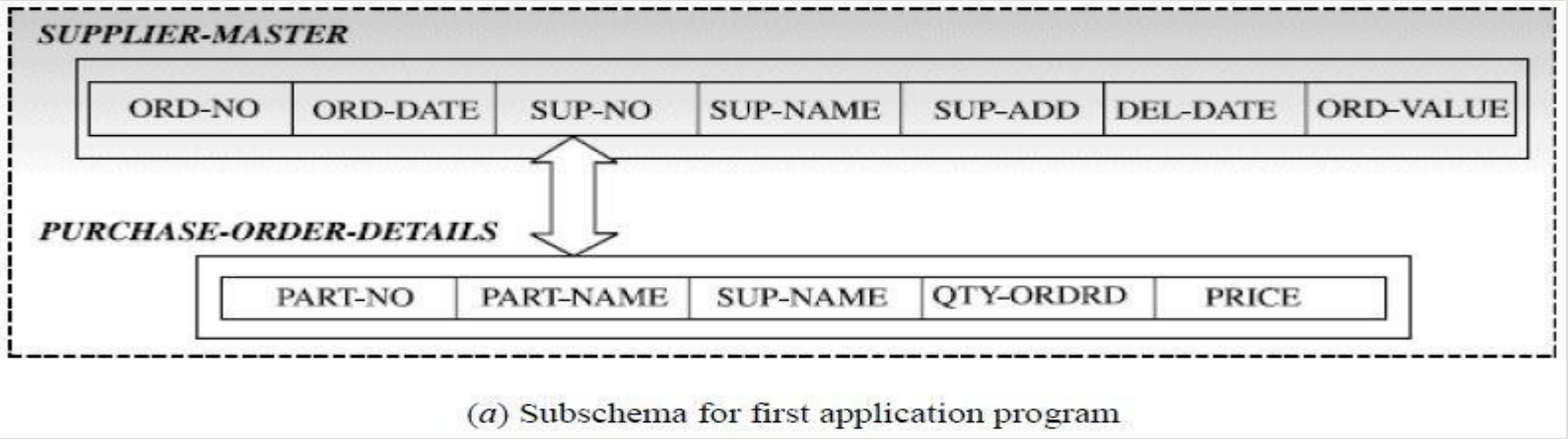
2). Logical schema:

- The logical schema is concerned with exploiting the data structures offered by a DBMS in order to make the scheme understandable to the computer.
- The logical schema is the most important as programs use it to construct applications.

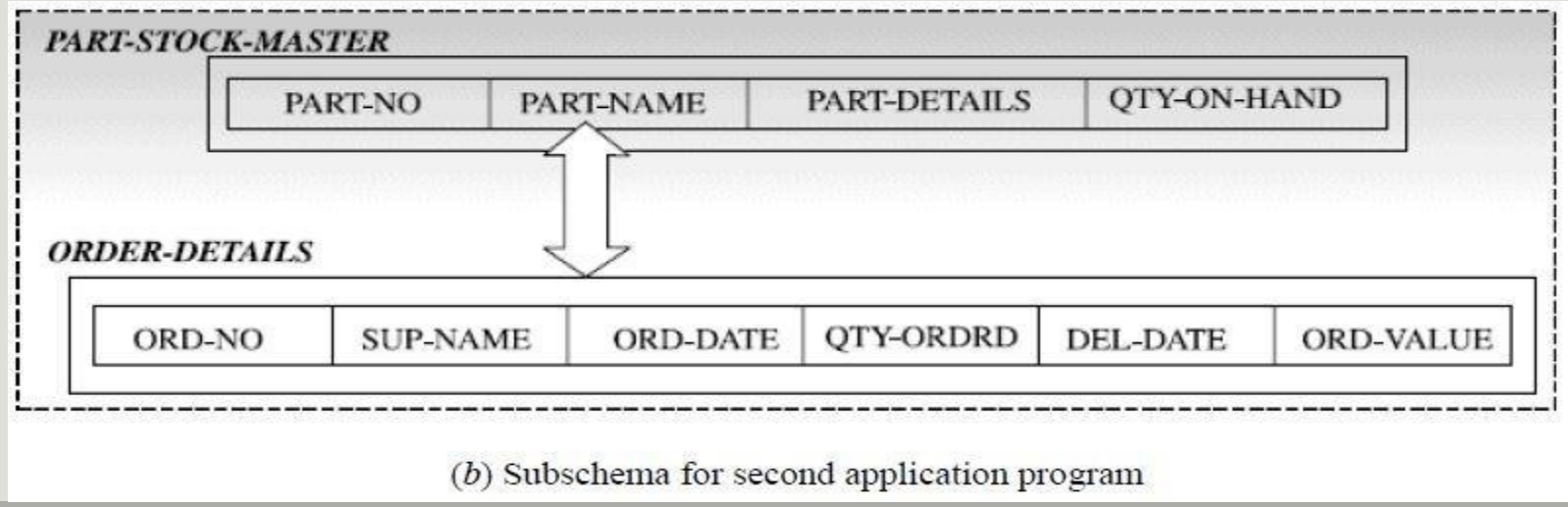
Sub-Schema

- A subschema is a subset of the schema and inherits the same property that a schema has.
- The plan (or scheme) for a view is often called subschema.
- Subschema refers to an application programmer's (user's) view of the data item types and record types, which he or she uses.
- It gives the users a window through which he or she can view only that part of the database, which is of interest to him.
- Therefore, different application programs can have different view of data.
- Individual application programs can change their respective subschema without effecting subschema views of others.
- The database administrator (DBA) ensures that the subschema requested by application programs is derivable from schema.

Sub-Schema…



(*a*) Subschema for first application program



(*b*) Subschema for second application program

Sub-Schema…

- With the change in physical organization of data, application programs for subschema need not be changed or modified.
- The subschema is sometimes referred to as an LVIEW or logical view.

Instances/State

- 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.
- The term instance is also called as state of the database or snapshot.
- The difference between database schema and database state or instance is very distinct.
- In the case of a database schema, it is specified to DBMS when new database is defined, whereas at this point of time, the corresponding database instance/state is empty with no data in the database.
- At any point of time, the current state of the database is called the instance.

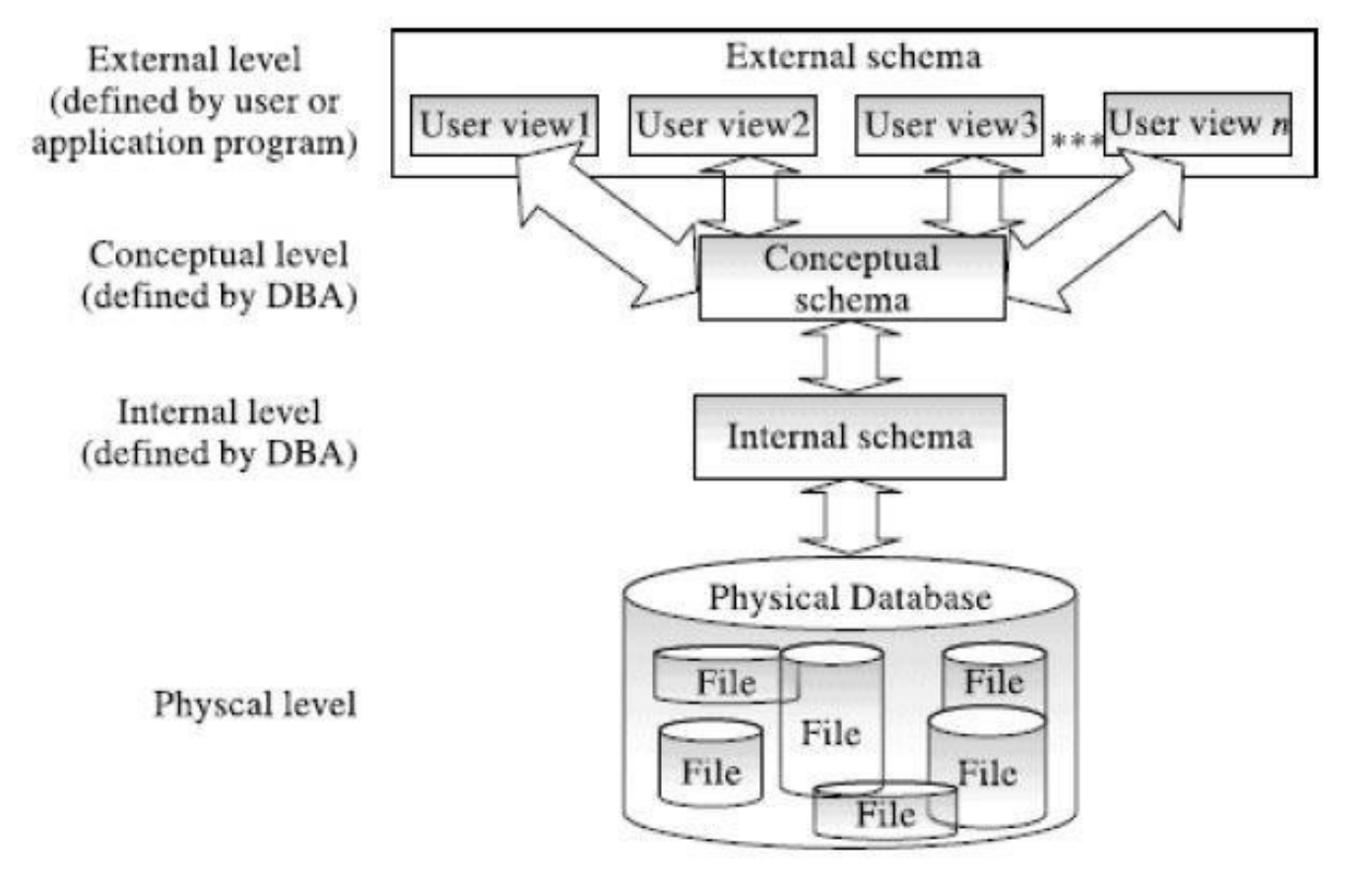
Instances/State…

SALES			
CUST-ID	**PROD-ID**	**PROD-QTY**	**PROD-PRICE**
1001	A12345	100	6700
1000	B23412	250	4000
1010	B44332	120	14000
1005	A98765	110	5500
1001	A29834	300	12999

Three level ANSI-SPARC DB Architecture

- For the first time in 1971, Database Task Group (DBTG) proposed a two-tier architecture (see slide no. 78).
- In 1975, ANSI-SPARC (American National Standards Institute — Standards Planning and Requirements Committee) produced a three-tier architecture with a system catalog.
- The architecture of most commercial DBMSs available today is based to some extent on ANSI- SPARC proposal.
- ANSI-SPARC three-tier database architecture consists of following three levels:
  1. Internal level
  2. Conceptual level
  3. External level

Three level ANSI-SPARC DB Architecture



Three level ANSI-SPARC DB Architecture...

- Information about the internal, conceptual and external schemas is stored in the system catalog.
- Let us take an ex

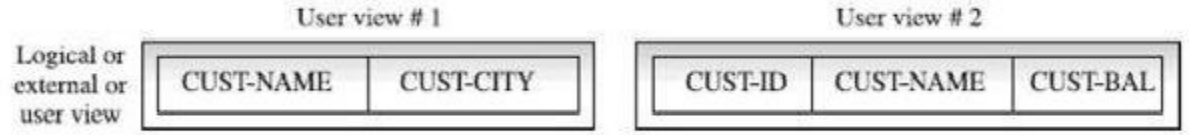
CUSTOMER				
CUST-ID	CUST-NAME	CUST-STREET	CUST-CITY	CUST-BAL

(*a*) CUSTOMER record

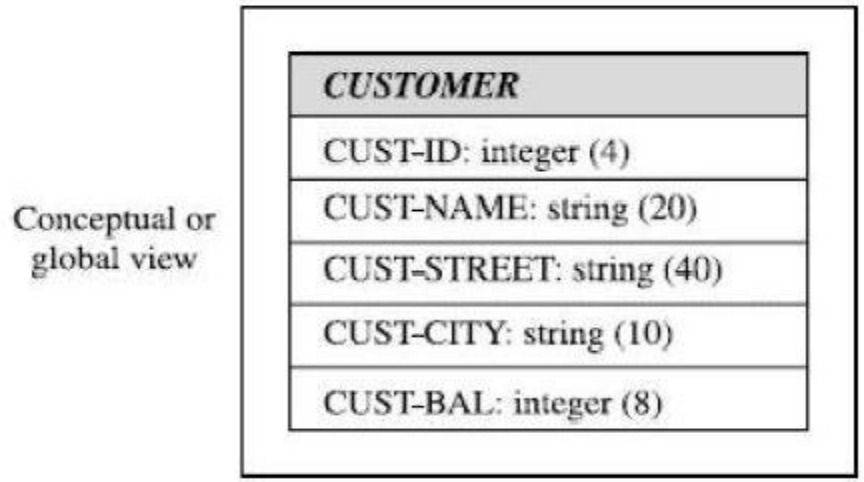
- The integrated record definition of CUSTOMER record is shown in below figure:

```
type CUSTOMER = record
                    CUST-ID = integer;
                    CUST-NAME = string;
                    CUST-STREET = string;
                    CUST-CITY = string;
                    CUST-BAL = integer;
                end
```

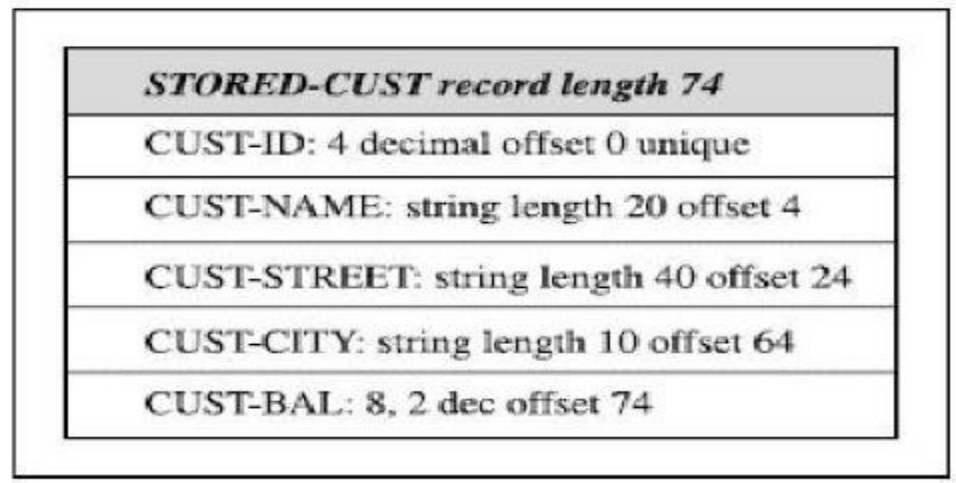
(*b*) Integrated record definition of CUTOMER record



(*a*) Logical records



(*b*) Conceptual records



(*c*) Internal record

Three level ANSI-SPARC DB Architecture…

1) Internal level:

- Internal level is the physical representation of the database on the computer and this view is found at the lowest level of abstraction of database.
- This level indicates howthe data will be stored in the database and describes the data structures, file structures and access methods to be used by the database.
- Just below the internal level there is physical level data organization.
- The internal schema defines the internal level.
- The internal schema contains the definition of the stored record and the method of representing the data fields.
- Internal schema summarizes how the relations described in the conceptual schema are actually stored on secondary storage devices.
- It interfaces with the operating system access methods also called file management techniques for storing and retrieving data records.

Three level ANSI-SPARC DB Architecture…

2) Conceptual Level

- The conceptual level is the middle level in the three-tier architecture.
- Conceptual level provides the community view of the database and describes what data is stored in the database and the relationships among the data.
- It contains the logical structure of the entire database.
- One conceptual view represents the entire database of an organisation.
- The conceptual schema defines conceptual view, which is also known as logical schema.
- This schema contains the method of deriving the objects in the conceptual view from the objects in the internal view.
- The conceptual level supports each external view.
- However, this level must not contain any storage-dependent details such as number of bytes occupied by the data.

Three level ANSI-SPARC DB Architecture…

3). External Level:

- The external level is the user's view of the database.
- This level is at the highest level of data abstraction where only those data is described that is relevant/useful to the user.
- Any number of user views, even identical, may exist for a given conceptual or global view of the database.
- In the external level, the different views may have different representations of the same data.
- For example, one user may view data in the form of DD:MM:YY while another may view as YY:MM:DD.
- Any given database has exactly one internal or physical schema and one conceptual schema because it has just one set of stored relations.
- But, it may have several external schemas, each tailored to a particular group of users.

Advantages of Three-tier Architecture

- Each user is able to access the same data in their own view of the data as per their own needs.
- Each user can change the way he or she views the data and this change does not affect other users of the same database.
- The user's interaction with the database is independent of physical data storage organization.
- The internal structure of the database remains unaffected when the data is shifted to a new storage device.
- The database administrator (DBA) is able to change the database storage structures without affecting the user's view.
- The DBA is able to change the conceptual structure of the database without affecting all users.

Functions of DBMS

1). Data storage management:

- The DBMS creates the complex structures required for data storage in the physical database.
- It provides a mechanism for management of permanent storage of the data.
- The internal schema defines how the data should be stored by the storage management mechanism and the storage manager interfaces with the operating system to access the physical storage.

2). Data manipulation management:

- A DBMS furnishes users with the ability to retrieve, update and delete existing data in the database or to add new data to the database.

Functions of DBMS…

3). Data dictionary system catalog management:

- The DBMS provides a data dictionary or system catalog function in which descriptions of data items are stored and which is accessible to users.
- System catalog or data dictionary is a system database, which is a repository of information describing the data in the database.
- It is the data about the data or metadata.
- DBMS will consult the system catalog to verify that a requested table exists and that the user issuing the request has the necessary access privileges.

4). Database Communication Interfaces:

- The end-user's requests for database access from remote location through network are transmitted to DBMS in the form of communication messages.

Functions of DBMS…

- The DBMS integrates with a communication software component called data communication manager (DCM) which controls such message transmission activities.
- DCM is not a part of DBMS, both work in harmony in which the DBMS looks after the database and the DCM handles all messages to and from the DBMS.
- **5). Authorisation / Security Management:**
- The DBMS protects the database against unauthorized access, either intentional or accidental. It furnishes mechanism to ensure that only authorized users can access the database.
- Security rules determine which users can access the database, which data items each user may access and which data operations (read, add, delete and modify) the user may perform.
- This is especially important in multi-user environment where many users can access the database simultaneously.
- DBMS monitors and controls the level of access for each user and the operations that each user can perform on the data depending on the access privileges or access rights of the users

Functions of DBMS…

6). Backup and Recovery Management:

- The DBMS provides mechanisms for backing up data periodically and recovering from different types of failures.
- It ensures that the aborted or failed transactions do not create any adverse effect on the database or other transactions.
- The recovery mechanisms of DBMSs make sure that the database is returned to a consistent state after a transaction fails or aborts due to a system crash, media failure, hardware or software errors, power failure, and so on.
- Many DBMSs enable users to make full or partial backups of their data. A full backup saves all the data in the target resource, such as the entire file or an entire database.
- Partial, or incremental, backups usually record only the data that has been changed since the last full backup.

Functions of DBMS…

7). Concurrency Control Services:

- Since DBMS support sharing of data among multiple users, they must provide a mechanism for managing concurrent access to the database.
- DBMS ensure that the database is kept in consistent state and that the integrity of the data is preserved.
- It ensures that the database is updated correctly when multiple users are updating the database concurrently.

8). Transaction Management:

A transaction is a series of database operations, carried out by a single user or application program, which accesses or changes the contents of the database.

Therefore, a DBMS must provide a mechanism to ensure either that all the updates corresponding to a given transaction are made or that none of them is made.

Functions of DBMS…

9). Utility Services:

- The DBMS provides a set of utility services used by the DBA and the database designer to create, implement, monitor and maintain the database.
- These utility services help the DBA to administer the database effectively.

10). Database Access and Application Programming Interfaces:

- All DBMSs provide interface to enable applications to use DBMS services.
- They provide data access via structured query language (SQL).
- The DBMS query language contains two components: (a) a data definition language (DDL) and (b) a data manipulation language (DML).
- DDL defines the structure in which the data are stored and the DML allows end users to extract the data from the database.