Database Management System

Unit-One

Comparison between data and information

S.N.	Data	Information
1	Data is a raw and unorganized fact that required to be processed to make it meaningful.	Information is a set of data which is processed in a meaningful way according to the given requirement.
2	Data is meaningless.	Information is meaningful.
3	It is in unstructured forms.	It is in structured forms.
4	Generally, data comprises facts, observations, perceptions numbers, characters, symbols, image, etc.	Information is processed, structured, or presented in a given context to make it meaningful and useful.
5	It never depends on Information	It depended on Data.
6	Measured in bits and bytes.	Measured in meaningful units like time, quantity, etc.
7	It can't be used for decision making	It is widely used for decision making.
8	Data alone has no significance.	Information is significant by itself.
9	Data may or may not be useful.	Information is useful.
10	Data is never designed to the specific need of the user.	Information is always specific to the requirements and expectations because all the irrelevant facts and figures are removed, during the transformation process.
11	Data is input.	Information is output.
12	Example:- Kathmandu, capital, Nepal, city	12. Example: - Kathmandu is the capital city of Nepal.

Database:-

- Database is an organized collection of related data or information which are shared and used for multiple purposes.
- A database is an organized collection of related data, generally stored and accessed electronically from a computer system.
- A database is an organized collection of structured information, or data, typically stored electronically in a computer system.
- A database is a collection of information that is organized so that it can be easily accessed, managed and updated.
- A database usually controlled by Database Management System (DBMS).
- Example
  a) Consider name and telephone number of the people you know.
  b) Phone diaries.
  c) SEE results

d) Nihareeka college database such as

- Teacher record
- Student record
- Book record

Database Management System:-

- A Database Management System(DBMS) is a collection of program that enable user to create and maintain any database.
- A DBMS is a software design to assist in maintaining and utilizing large collection of data.
- A DBMS is a general purpose software system that facilitates the process of defining, constructing and manipulating data for various application.

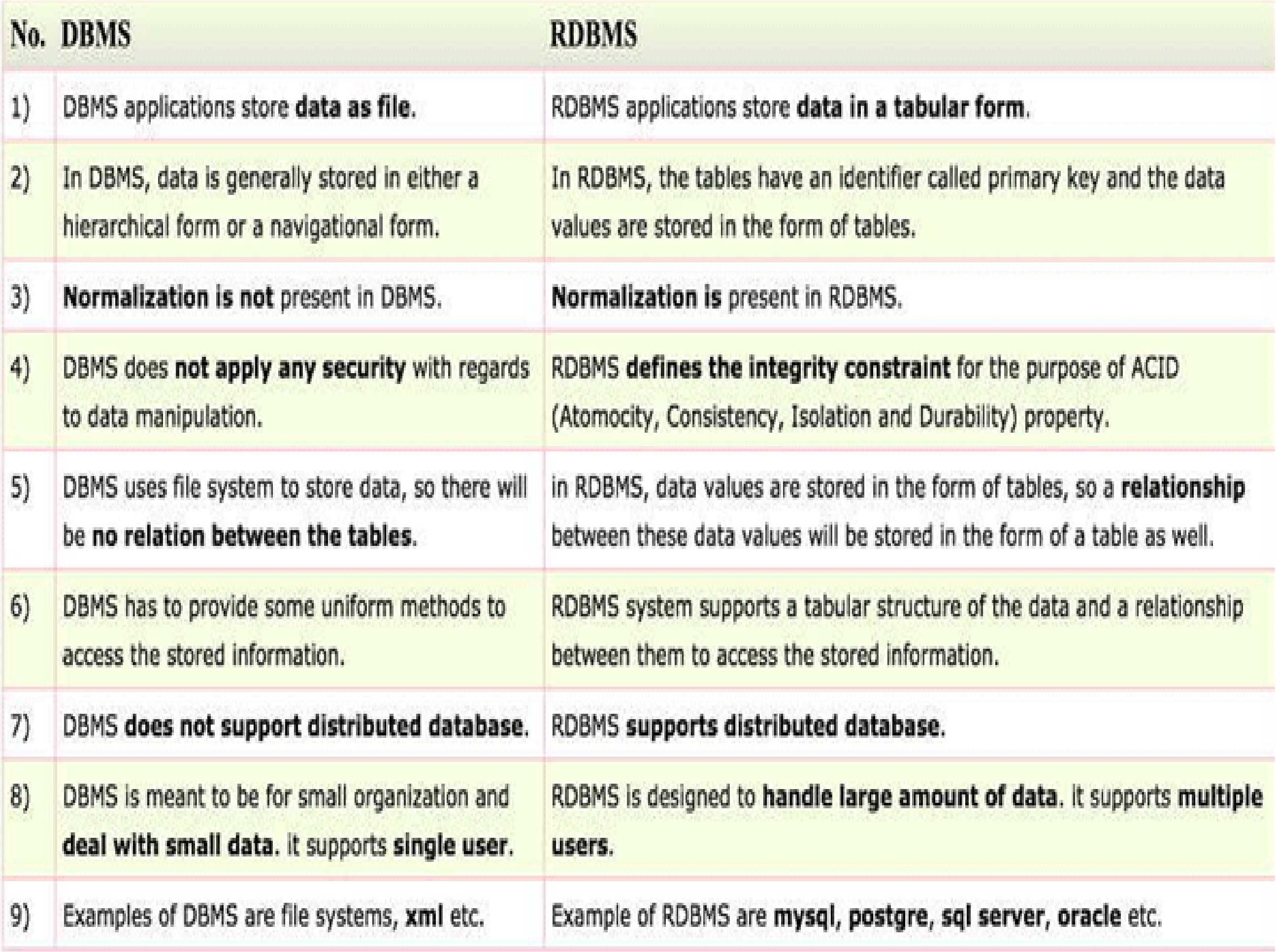
List of DBMS software

1. MS Access
2. MySQL
3. Oracle
4. Microsoft SQL server
5. MongoDB
6. MariaDB
7. PostgreSQL
8. FireBird
9. PhpMyAdmin
10. Hadoop HDFS
11. SQLite
12. Informix
13. SQL developer
14. AltiBase
15. IBM DB2
16. TeraData
17. FoxPro
18. Libre Office Base, etc.

+Difference between DBMS and Flat-Files

DBMS	Flat-Files
Multi-user access	It does not support multi-user access
Design to fulfill the need for small and large businesses	It is only limited to smaller DBMS system.
Remove redundancy and Integrity	Redundancy and Integrity issues
Expensive. But in the long term Total Cost of Ownership is cheap	It's cheaper
Easy to implement complicated transactions	No support for complicated transactions

Difference between DBMS and RDBMS



No.	DBMS	RDBMS
1)	DBMS applications store **data as file.**	RDBMS applications store **data in a tabular form.**
2)	In DBMS, data is generally stored in either a hierarchical form or a navigational form.	In RDBMS, the tables have an identifier called primary key and the data values are stored in the form of tables.
3)	**Normalization is not** present in DBMS.	**Normalization is** present in RDBMS.
4)	DBMS does **not apply any security** with regards to data manipulation.	RDBMS **defines the integrity constraint** for the purpose of ACID (Atomocity, Consistency, Isolation and Durability) property.
5)	DBMS uses file system to store data, so there will be **no relation between the tables.**	in RDBMS, data values are stored in the form of tables, so a **relationship** between these data values will be stored in the form of a table as well.
6)	DBMS has to provide some uniform methods to access the stored information.	RDBMS system supports a tabular structure of the data and a relationship between them to access the stored information.
7)	DBMS **does not support distributed database.**	RDBMS **supports distributed database.**
8)	DBMS is meant to be for small organization and **deal with small data.** It supports **single user.**	RDBMS is designed to **handle large amount of data.** It supports **multiple users.**
9)	Examples of DBMS are file systems, **xml** etc.	Example of RDBMS are **mysql, postgre, sql server, oracle** etc.

Advantages of DBMS

1. Data Redundancy

Unlike traditional file-system storage, Data Redundancy in DBMS is very less or not present. Data Redundancy occurs when the same data are stored unnecessarily at different places. Data Redundancy is reduced or eliminated in DBMS because all data are stored at a centralized location rather than being created by individual users and for each application.

For e.g: Application A and Application B have the same user MARVEL, and we need to store personal information about the user such as Name, age, address, Date of Birth etc. Not to mention, this user has also access to different application, so in traditional file-based system, there is a need to maintain separate file system for each of the application to store user’s information while in DBMS approach, there could be just one centralized location where information can be down streamed to different application as and when needed.

2. Data Inconsistency

In traditional file system storage, the changes made by one user in one application doesn’t update the changes in other application, given both have the same set of details. While this is not the case with DBMS systems as there is a single repository of data that is defined once and is accessed by many users, and data are consistent.

3. Data Sharing

Data Sharing is the primary advantage of Database management systems. DBMS system allows users and applications to share Data with multiple applications and users. Data are stored in one or more servers in the network and that there is some software locking mechanism that prevents the same set of data from being changed by two people at the same time. While the file system doesn’t have this capability.

4. Data Searching

Searching and retrieving of data is very easy in DBMS systems. The need to write separate programs for each of the search is eliminated as in the case with a traditional file-based approach. In DBMS, we can write small queries to search for multiple information at a time from the data from DB servers.

5. Data Security

DBMS systems provide a strong framework to protect data privacy and security. DBMS ensures that only authorized users have access to data and there is a mechanism to define access privileges.

6. Data Concurrency

In DBMS, Data are stored in one or more servers in the network and that there is some software locking mechanism that prevents the same set of data from being changed by two people at the same time.

7. Data Integration

Data integration is a process of combining the data residing at different locations and present the user with a unified view of data. DBMS systems allow Data Integration with much feasibility.

8. Data Access

While in traditional file-based approach, it might take hours to look for very specific information that might be needed in the context of some business emergency, while DBMS reduces this time to a few seconds. This is a great advantage of DBMS because we can write small queries which will search the Database for you and it will retrieve the information in the fastest way possible due to its inbuilt searching operations.

9. Decision making

Improved Data Sharing and better-managed data allow business to make quality business decisions which will promote the growth of the organization.

10. Data Backup and Recovery

This is another advantage of DBMS as it provides a strong framework for Data backup, users are not required to back up their data periodically and manually, it is automatically taken care by DBMS. Moreover, in case of a server crash, DBMS restores the Database to its previous condition.

11. Data Migration

There are some data which are access very frequently while there are few that aren't. So, DBMS provides the capability to access the frequently accessed data as quickly as possible.

12. Low Maintenance Cost

Though DBMS systems might be costly at the time of purchase but their maintenance involves a very minimal cost.

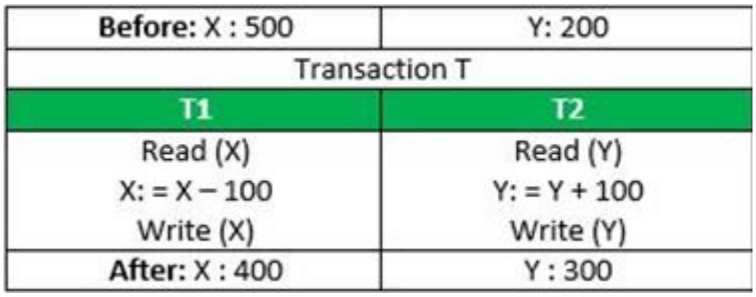
14. Data Loss is almost eliminated

With DBMS, one can keep information for thousands of years, provided we don't see the doomsday. Data security and very low storage cost (as compared to our previous generations) in the current century cut any possibility of Data Loss.

15. Data Atomicity

An atomic transaction is one in which all of the database actions occur or none of them do. It is the duty of DBMS to store a complete transaction in the database. If any transaction is partially completed, then it rolls backs them.

For e.g: If we make an online purchase, money is deducted from our account while if the purchase is somehow failed, then no money is deducted or if it gets deducted, it gets returned within few days.



Before: X : 500	Y: 200
Transaction T	
T1	T2
Read (X) X: = X – 100 Write (X)	Read (Y) Y: = Y + 100 Write (Y)
After: X : 400	Y : 300

16. Easy to add new data.

17. Easy to modify data.

18. Easy to delete data.

19. Organize the data in proper sequence.

20. Eliminate redundancy of data.

21. Easy queries of data in database.

22. Allows multiple users to be active at one time. (Data in database may be shared among several users.)

23. Protecting data against unauthorized access.

Disadvantages of DBMS

1. Cost of purchasing or developing software.
2. Chances of losing data.
3. Chances of data leakage and hacking.
4. Cost of sharing data.
5. Required trained manpower.
6. Cost of maintenance of software.

Importance of DBMS

We have always needed information to be saved somewhere; we started using the practice to save data in many forms since ancient times. You can see we evolved from Stone Age to modern age. Earlier we used to write on stones and caves now we use computers to save the piece of information.

To make this information more useful, easy to access and protected we use database management systems.

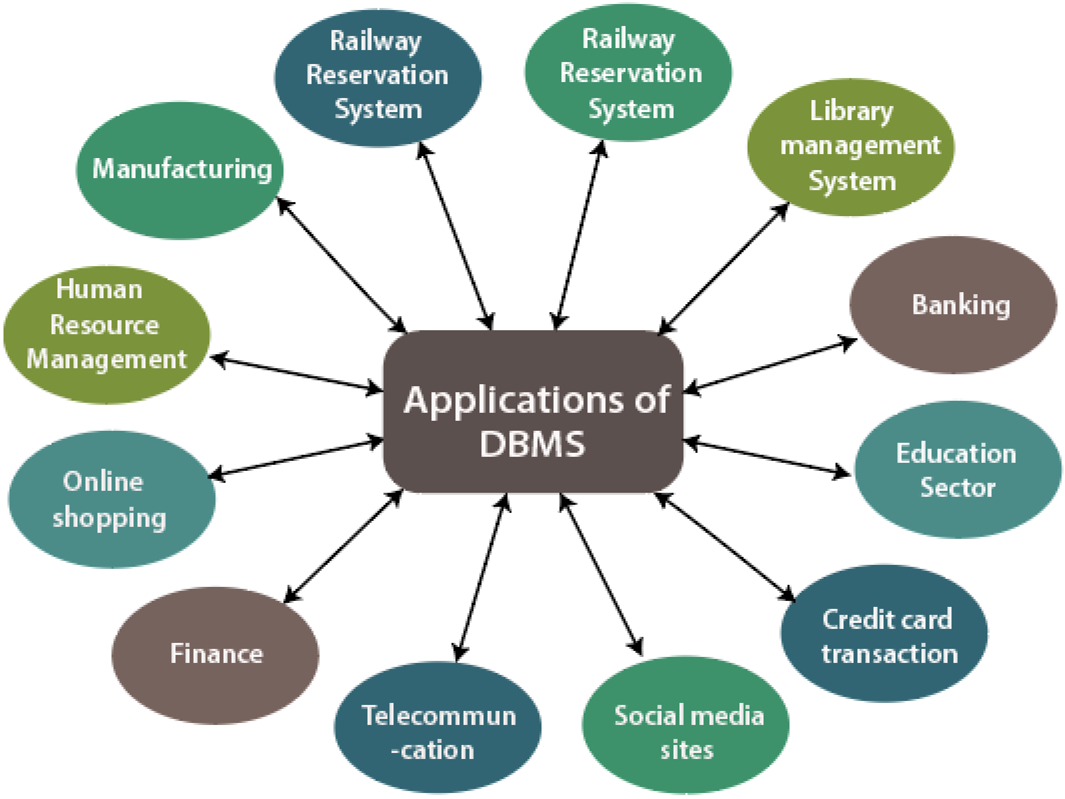
DBMS is important because it manages the data efficiently and allows users to perform multiple tasks on it with the ease.

Without DBMS, we might have to do it manually and would have taken more time.

Also DBMS helps preserving the data in many forms and which we can use anywhere and may be after ages, to keep a record of what we have done to what we will do, everything has to be kept in form of some record. And DBMS provides efficient ways to accomplish that task.

Suppose, take your bank account. How will it be if people at bank will be managing it manually? There is always a chance of human error. And nobody likes miscalculation when it comes to their money. That's why they too use some kind of DBMS.

Application of DBMS



Railway Reservation System

Database is required to keep record of ticket booking, train's departure and arrival status. Also if trains get late then people get to know it through database update.

Library Management System

There are thousands of books in the library so it is very difficult to keep record of all the books in a copy or register. So DBMS used to maintain all the information relate to book issue dates, name of the book, author and availability of the book.

Banking

We make thousands of transactions through banks daily and we can do this without going to the bank. So how banking has become so easy that by sitting at home we can send or get money through banks. That is all possible just because of DBMS that manages all the bank transactions.

Universities and colleges

Examinations are done online today and universities and colleges maintain all these records through DBMS. Student's registrations details, results, courses and grades all the information are stored in database.

Credit card transactions

For purchase of credit cards and all the other transactions are made possible only by DBMS. A credit card holder knows the importance of their information that all are secured through DBMS.

Social Media Sites

We all are on social media websites to share our views and connect with our friends. Daily millions of users signed up for these social media accounts like facebook, twitter, pinterest and Google plus. But how all the information of users are stored and how we become able to connect to other people, yes this all because DBMS.

Telecommunications

Any telecommunication company cannot even think about their business without DBMS. DBMS is must for these companies to store the call details and monthly post paid bills.

Finance

Those days have gone far when information related to money was stored in registers and files. Today the time has totally changed because there are lots f thing to do with finance like storing sales, holding information and finance statement management etc.

Military

Military keeps records of millions of soldiers and it has millions of files that should be keep secured and safe. As DBMS provides a big security assurance to the military information so it is widely used in militaries. One can easily search for all the information about anyone within seconds with the help of DBMS.

Online Shopping

Online shopping has become a big trend of these days. No one wants to go to shops and waste his time. Everyone wants to shop from home. So all these products are added and sold only with the help of DBMS. Purchase information, invoice bills and payment, all of these are done with the help of DBMS.

Human Resource Management

Big firms have many workers working under them. Human resource management department keeps records of each employee’s salary, tax and work through DBMS.

Manufacturing

Manufacturing companies make products and sales them on the daily basis. To keep records of all the details about the products like quantity, bills, purchase, supply chain management, DBMS is used.

Airline Reservation system

Same as railway reservation system, airline also needs DBMS to keep records of flights arrival, departure and delay status.

So in short, one can say the DBMS is used everywhere around us and we cannot rely without DBMS.

Objective of DBMS

The main objective of DBMS are:

1. **Minimal Redundancy:** Data redundancy is the duplicate of same data at more than one storage place. This duplication of data leads to wastage of storage space and time and affect cost also. This redundancy has to be eliminated by integrating the data at one place.

2. **Consistency:** Data duplication create multiple level of updation. At some occasion updation of duplication data entries may supply incorrect or conflicting information. At such time, the database is said to be inconsistent. Consistency of data has to be achieved through redundancy control.
3. **Sharing of Data:** This means various users can use the same data in the database. Moreover new application can be developed according to the needs to operate against the same stored data. Hence the objectives of DBMS is to satisfy the data requirement of various new applications without the need of having separate data for each application.
4. **To Provide Multiple User Interfaces:** In order to allow different users to access the database DBMS provide:-

   a) Query Language: Query language for casual users such as SOL to access the database.

   b) Programming Language Interfaces: For application programmers.

   c) Menu Driven Interfaces: For stand-alone users.
5. **Simplicity:** Another objectives of DBMS is to make application development procedure simple and easier. To achieve this DBMS is accompanied without powerful query manipulation and reports generations tools.
6. **Flexibility:** DBMS allows change to the structure of database without affecting the stored data, the existing application. Thus it should make the application development cheaper, faster and flexible.
7. **Data Migration:** This objective is important to make the database economical. All data are not referenced very frequently. Thus the rarely accessed data can be stored on slower access or on cheap devices. Whereas more frequently accessed data can be stored on fast access or direct access media devices. Thus data migration implies the adjustment of data on costly or cheap media devices.
8. **To Restrict Unauthorized Access:** Data in database must be secured. Hence an important objective of database system is to restrict unauthorized access. To ensure the DBMS must provide:
   a) Identification of user of the database, before they can use the database.
   b) Monitoring users action so that if they do something wrong they are likely to be found.
   c) All content should be proper and not easy to check.
9. **Privacy and Security:** Privacy means when, how and to what extent data should be given to users. Database are costly products and hence their security is very important. Security of data is needed from accidental as well as intentional disposal. Thus to achieve privacy and security is also an important objective of DBMS.
10. **To Enforce Integrity:** Integrity is data accuracy. It also implies that incorrect information cannot be stored in the database. In order to achieve the objective of

integrity some integrity constraints are enforced on database. A DBMS should have the capability for designing imposing consistency constraints.

11. **Maintain Standards:** All applicable standard should be followed in the representation of data such as format, conventions on data names, documentation etc. The standardized data is very helpful during migration or interchanging of data. This will result in uniformity of the entire database as well as its usage.

Some objectives of DBMS are given below

1. Provide for mass storage of relevant data.
2. Making easy access to data for the authorized user.
3. Providing prompt response to users requests for data.
4. Eliminate redundantly (Duplicate) d data.
5. Allow multiple users to be active at one time.
6. Allow the growth of database system
7. Provide data integrity.
8. Protect the data from physical harm and unauthorized access.
9. Serving different types of users. the
10. Provide security with a user access privilege.
11. Combining interrelated data to generate a report.
12. Provide multiple views for same data.

Data Models

Data models define how the logical structure of a database is modeled. Data Models are fundamental entities to introduce abstraction in a DBMS. Data models define how data is connected to each other and how they are processed and stored inside the system.

The very first data model could be flat data-models, where all the data used are to be kept in the same plane.

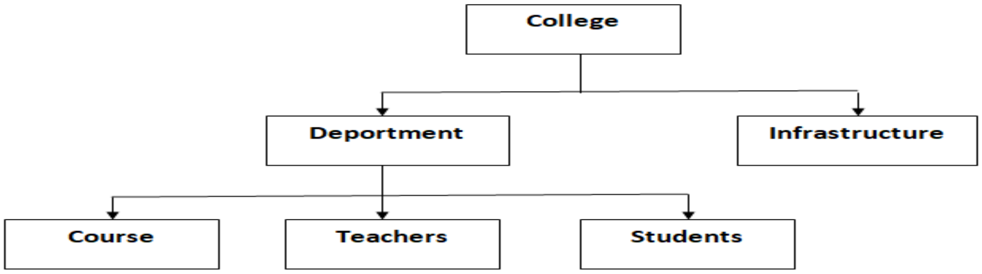
A Database model defines the logical design and structure of a database and defines how data will be stored, accessed and updated in a database management system. While the Relational Model is the most widely used database model, there are other models too:

- Hierarchical Model
- Network Model
- Entity-relationship Model
- Relational Model
- Object-Oriented Model

1.Hierarchical Data Model

In this model, there is a parent-child relationship. In this model, each entity has only one parent and many abstract children. There is only one entity in this model that we call root.

In this model, data is organized in a tree-like structure, which has only one root. In this, the data is stored as records that are connected to each other. It was proposed in 1970.



Advantage of the hierarchical model

- It promotes data sharing.

- There is a parent/child relationship due to which its concepts are simple.

- It provides database security.

- It takes 1 to many relationships.

A disadvantage of the hierarchical model

- It is not flexible

- It does not have a data definition and data manipulation languages.

- It requires knowledge of physical data storage for complex implementation.

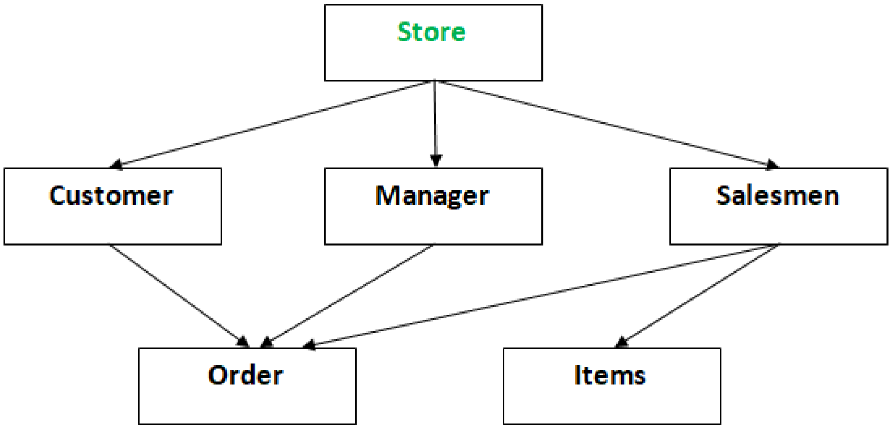
2. Network Data Model

This model is an extension of the hierarchical model.

In a network model, data is organized into graphs. And it can have more than one parent node.

That is, there is more parent/child relationship in it. And in it, some entity can access from multiple paths. So we can say that in this model we store and access the data as a network.

The network model was first used until the relational model was proposed.



Advantage of a network model

- Its concept is as simple as the hierarchical model.
- There is more than one parent/child relationship.
- Data can be accessed easily in it.
- It provides data integrity(primary key, unique key, not null, check, foreign key).
- It contains a data definition language (DDL) (Create, Alter, Drop, Rename, Truncate) and data manipulation language (DML) (Insert, Update, Delete, Select).

Disadvantage of the network model

- Its database structure is very complex (difficult) because all the records in it are maintained using pointers.
- Changes in its structure require changes in all programs.

3. Relational Data Model

In this model, data is stored in relations i.e. tables and each relation has rows and columns. A relational model is a group of tables in which data and relationships are specified.

In this, the data is stored in two-dimensional tables, the tables are also called a relation. And the row of each table is called a tuple. The tuple represents the entity and the column of the table represents the attribute.

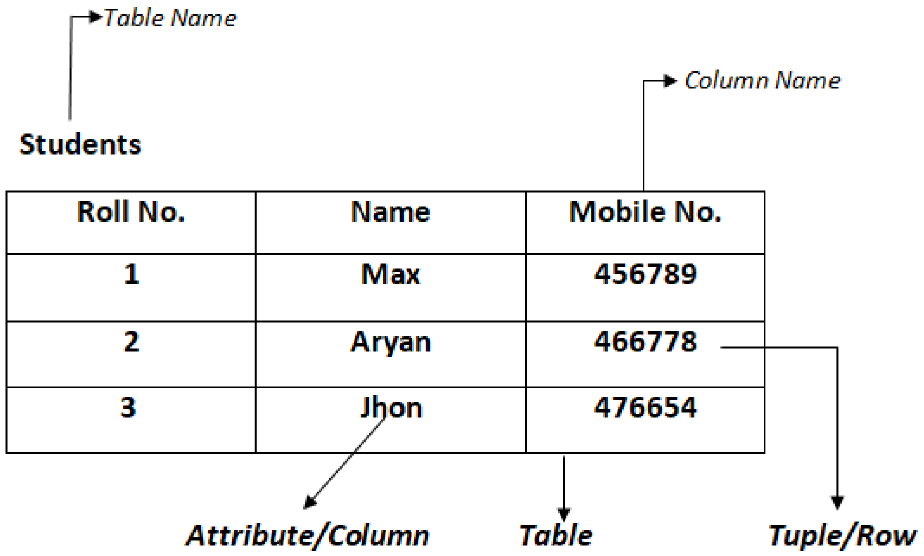
Row=tuple=record

Column=field=attribute

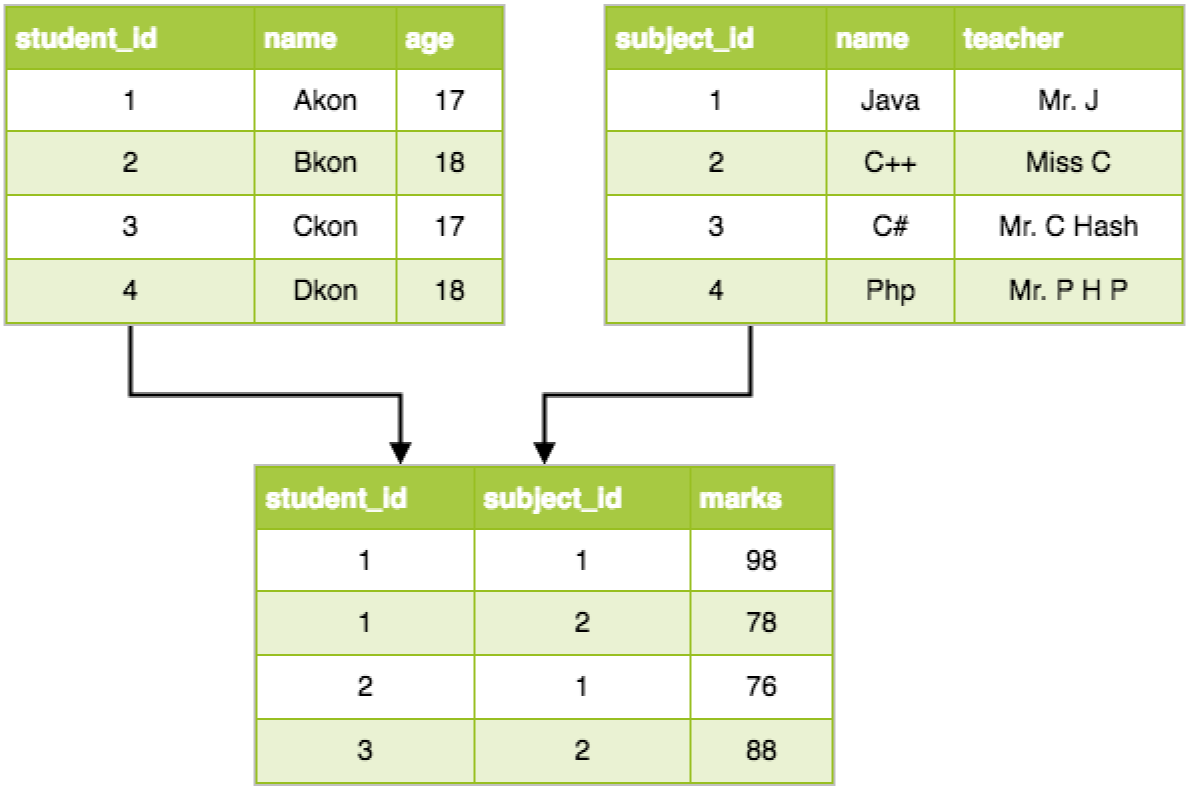
Student(id, name, address, phone, faculty, dob, gender)

The relational model was proposed by

E.F Codd in 1969 and since then this model has been used the most.



Roll No.	Name	Mobile No.
1	Max	456789
2	Aryan	466778
3	Jhon	476654



student_Id	name	age
1	Akon	17
2	Bkon	18
3	Ckon	17
4	Dkon	18

subject_Id	name	teacher
1	Java	Mr. J
2	C++	Miss C
3	C#	Mr. C Hash
4	Php	Mr. P H P

student_Id	subject_Id	marks
1	1	98
1	2	78
2	1	76
3	2	88

Advantage of the relational model

- It is very flexible, it can easily make any kind of change.

- In this, the data is kept in tables, so its concept is very simple.

- It provides data integrity(rules and regulations for the database). That is, no user can access the database without the owner’s permission.

A disadvantage of the relational model

- It requires powerful hardware computers, storage devices, and software.

- It is very easy to use but when a user stores data in it incorrectly then it becomes very bad DBMS.

- This is a very simple model, due to its simplicity, some users create their own database, causing the problem of data inconsistency, data duplication.

4. Entity-Relationship Data Model

In this database model, relationships are created by dividing object of interest into entity and its characteristics into attributes.

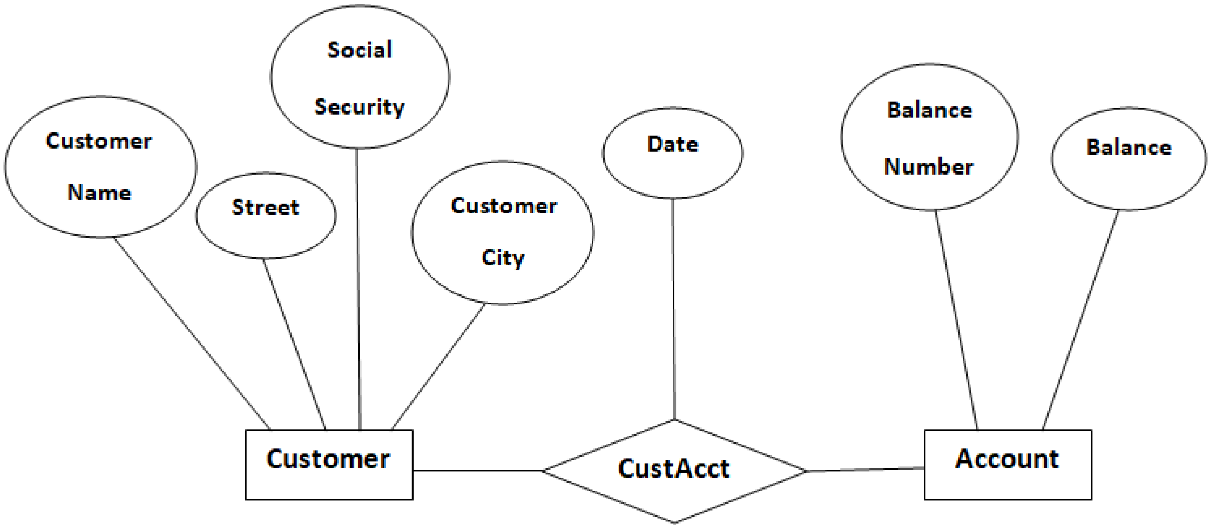
Different entities are related using relationships.

E-R Models are defined to represent the relationships into pictorial form to make it easier for different stakeholders to understand.

This model is good to design a database, which can then be turned into tables in relational model (explained below).

Let's take an example, If we have to design a School Database, then Student will be an entity with attributes name, age, address etc. As Address is generally complex, it can be another entity with attributes street name, pincode, city etc, and there will be a relationship between them.

Relationships can also be of different types. To learn about E-R Diagrams in details, click on the link.



Advantage of the E-R model

- The E-R model is very simple if we know the relationship between entities and attributes.

- This model is presented as a diagram. With which we can understand easily.

- There is no data manipulation.

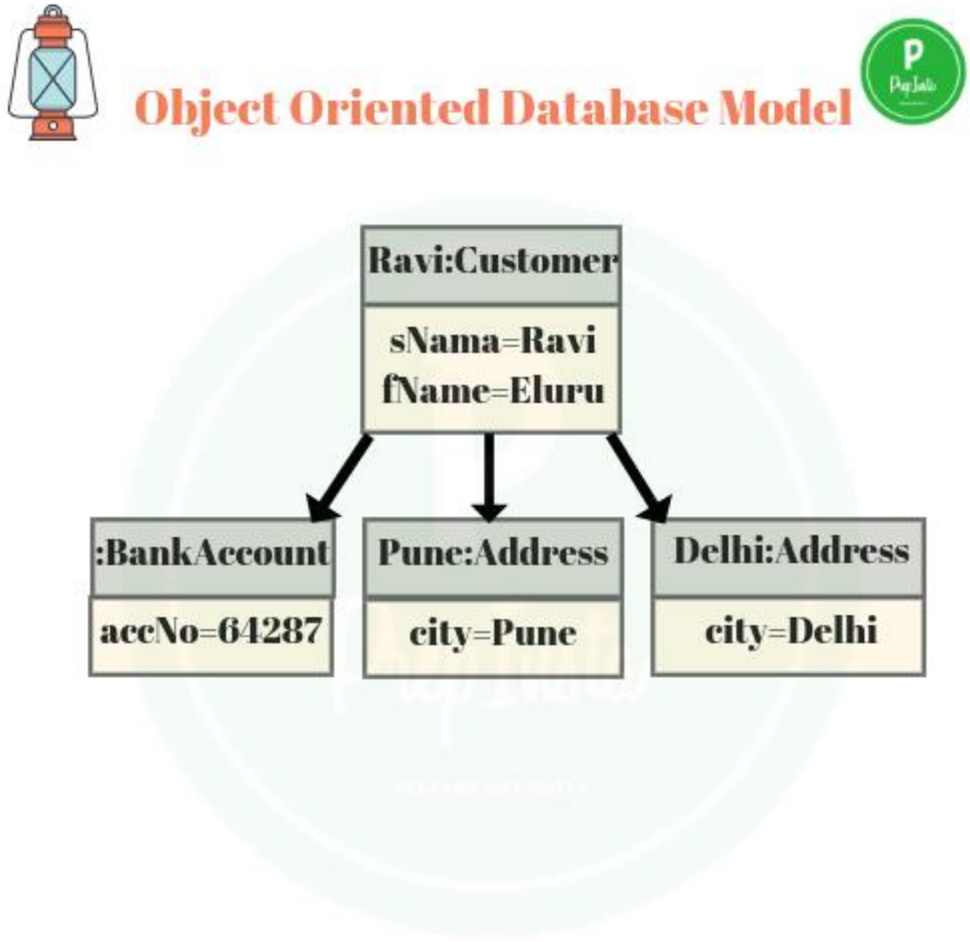
- Its design is of a high level.

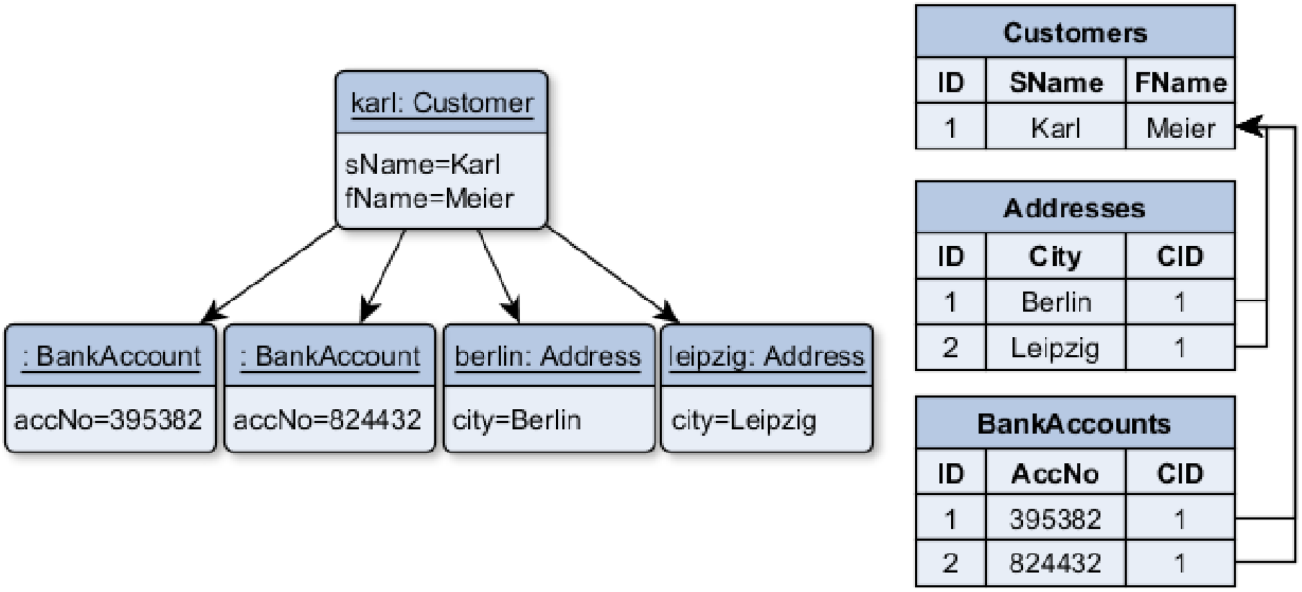
5. Object Oriented Data Model

In an object-oriented model, information or data is displayed as an object and these objects store the value in the instance variable.

In this model, object-oriented programming images are used.

This model works with object-oriented programming languages like – python, java, VB.net, and Perl, etc. It was constructed in the 1980s.





Customers

ID	SName	FName
1	Karl	Meier

Addresses

ID	City	CID
1	Berlin	1
2	Leipzig	1

BankAccounts

ID	AccNo	CID
1	395382	1
2	824432	1

Advantage of the object-oriented model

- Semantic content can be put in it.
- It supports inheritance which increases data integrity.
- It improves performance.

A disadvantage of the object-oriented model

- It requires a powerful system due to which the transaction is very slow.
- It is a very complex model.
- To use it, one has to learn it first.
- There is very little security in it.

Structure of DBMS

DBMS acts as interface between the user and the database. The user requests the DBMS to perform various operations such as insert, delete, update and retrieval on the database.

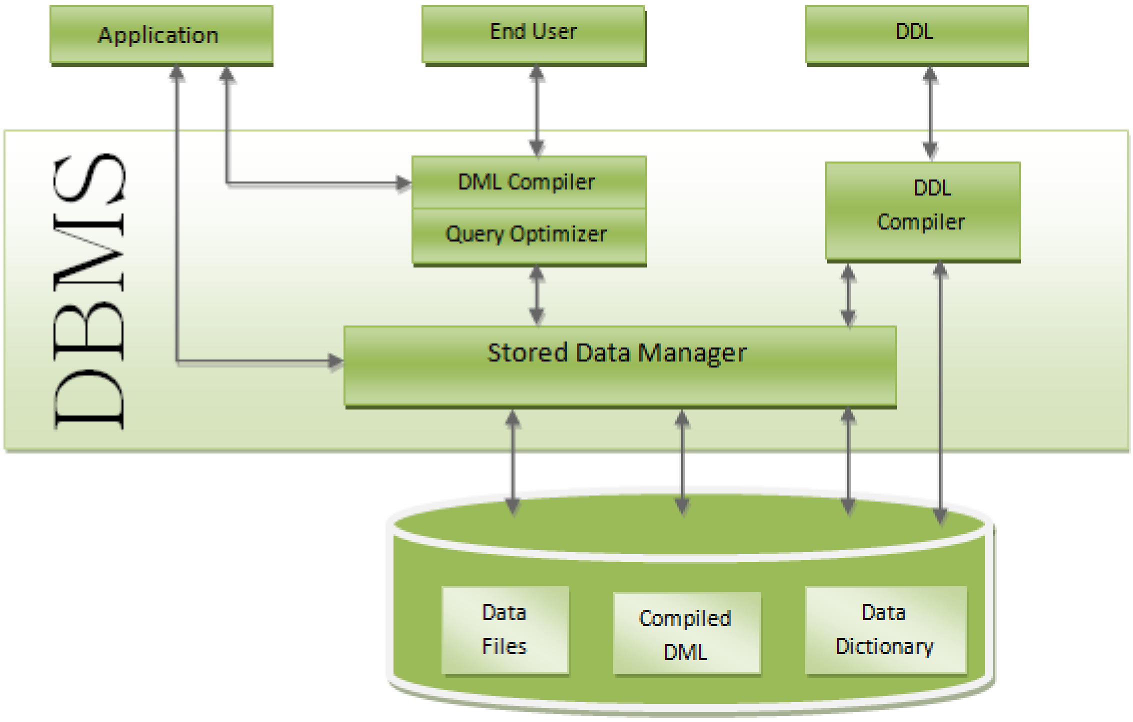


Fig: Structure of Database Management System

- **Applications:** – It can be considered as a user-friendly web page where the user enters the requests. Here he simply enters the details that he needs and presses buttons to get the data.
- **End User:** – They are the real users of the database. They can be developers, designers, administrators, or the actual users of the database.
- **DDL:** – Data Definition Language (DDL) is a query fired to create database, schema, tables, mappings, etc in the database. These are the commands used to create objects like tables, indexes in the database for the first time. In other words, they create the structure of the database.
- **DDL Compiler:** – This part of the database is responsible for processing the DDL commands. That means this compiler actually breaks down the command into machine-understandable codes. It is also responsible for storing the metadata information like table name, space used by it, number of columns in it, mapping information, etc.
- **DML Compiler:** – When the user inserts, deletes, updates or retrieves the record from the database, he will be sending requests which he understands by pressing some buttons. But for the database to work/understand the request, it should be broken down to object code. This is

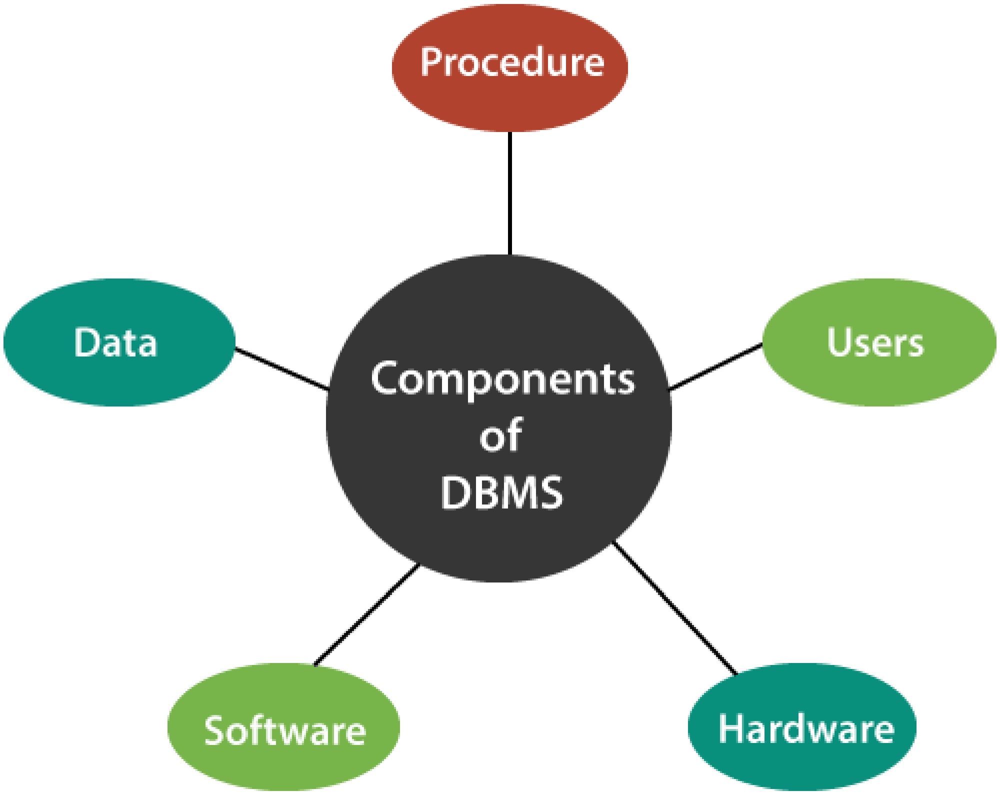
done by this compiler. One can imagine this as when a person is asked some question, how this is broken down into waves to reach the brain!

- **Query Optimizer: –** When a user fires some requests, he is least bothered how it will be fired on the database. He is not all aware of the database or its way of performance. But whatever be the request, it should be efficient enough to fetch, insert, update, or delete the data from the database. The query optimizer decides the best way to execute the user request which is received from the DML compiler. It is similar to selecting the best nerve to carry the waves to the brain!
- **Stored Data Manager: –** This is also known as Database Control System. It is one of the main central systems of the database. It is responsible for various tasks
  - It converts the requests received from query optimizer to machine-understandable form. It makes actual requests inside the database. It is like fetching the exact part of the brain to answer.
  - It helps to maintain consistency and integrity by applying the constraints. That means it does not allow inserting/updating / deleting any data if it has child entry. Similarly, it does not allow entering any duplicate value into database tables.
  - It controls concurrent access. If there are multiple users accessing the database at the same time, it makes sure, all of them see correct data. It guarantees that there is no data loss or data mismatch happens between the transactions of multiple users.
  - It helps to back up the database and recovers data whenever required. Since it is a huge database and when there is any unexpected exploit of the transaction, and reverting the changes is not easy. It maintains the backup of all data so that it can be recovered.
- **Data Files: –** It has the real data stored in it. It can be stored as magnetic tapes, magnetic disks, or optical disks.
- **Compiled DML: –** Some of the processed DML statements (insert, update, delete) are stored in it so that if there are similar requests, it will be re-used.
- **Data Dictionary: –** It contains all the information about the database. As the name suggests, it is the dictionary of all the data items. It contains a description of all the tables, view, materialized views, constraints, indexes, triggers, etc.

Components of DBMS

There are the following components of DBMS:

1. Software
2. Hardware
3. Procedures
4. Data
5. Users



Software

- The main component of a Database management system is the software. It is the set of programs which is used to manage the database and to control the overall computerized database.
- The DBMS software provides an easy-to-use interface to store, retrieve, and update data in the database.
- This software component is capable of understanding the Database Access Language and converts it into actual database commands to execute or run them on the database.

Hardware

- This component of DBMS consists of a set of physical electronic devices such as computers, I/O channels, storage devices, etc that create an interface between computers and the users.
- This DBMS component is used for keeping and storing the data in the database.

Procedures

- Procedures refer to general rules and instructions that help to design the database and to use a database management system.
- Procedures are used to setup and install a new database management system (DBMS), to login and logout of DBMS software, to manage DBMS or application programs, to take backup of the database, and to change the structure of the database, etc.

Data

- It is the most important component of the database management system.
- The main task of DBMS is to process the data. Here, databases are defined, constructed, and then data is stored, retrieved, and updated to and from the databases.

- The database contains both the metadata (description about data or data about data) and the actual (or operational) data.

Users

- The users are the people who control and manage the databases and perform different types of operations on the databases in the database management system.

There are three types of user who play different roles in DBMS:

- Application Programmers
- Database Administrators
- End-Users

1. **Application Programmers**

The users who write the application programs in programming languages (such as Java, C++, or Visual Basic) to interact with databases are called Application Programmer.

2. Database Administrators (DBA)

A person who manages the overall DBMS is called a database administrator or simply DBA.

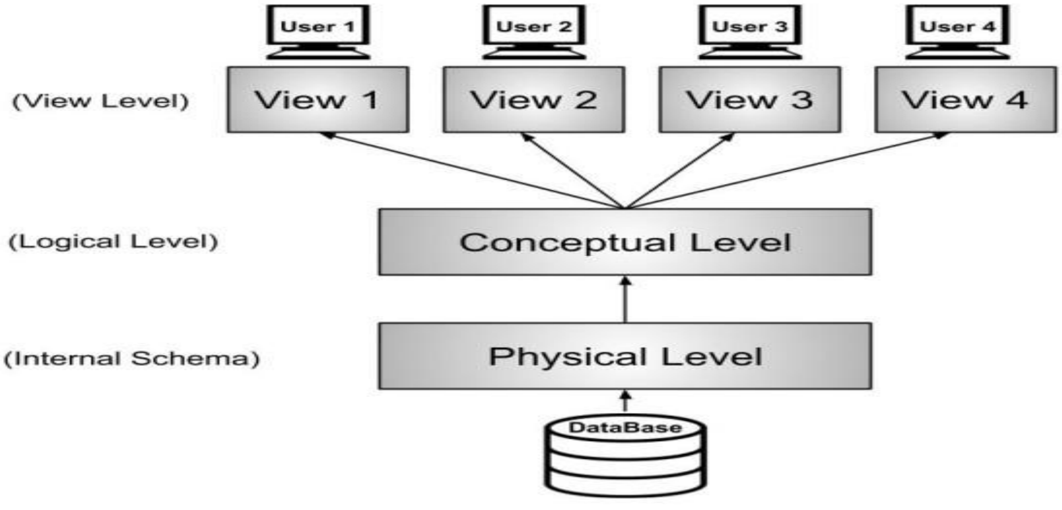
3. End-Users

The end-users are those who interact with the database management system to perform different operations by using the different database commands such as insert, update, retrieve, and delete on the data, etc.

Data Abstraction in DBMS

View of data in DBMS narrate how the data is visualized at each level of data abstraction? **Data abstraction** allow developers to keep complex data structures away from the users. The developers achieve this by hiding the complex data structures through **levels of abstraction**.

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



Levels of Data Abstraction

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

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

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

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

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

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

Data Independence

Data independence refers characteristic of being able to modify the schema at one level of the database system without altering the schema at the next higher level.

Data independence can be explained using the three-schema architecture.

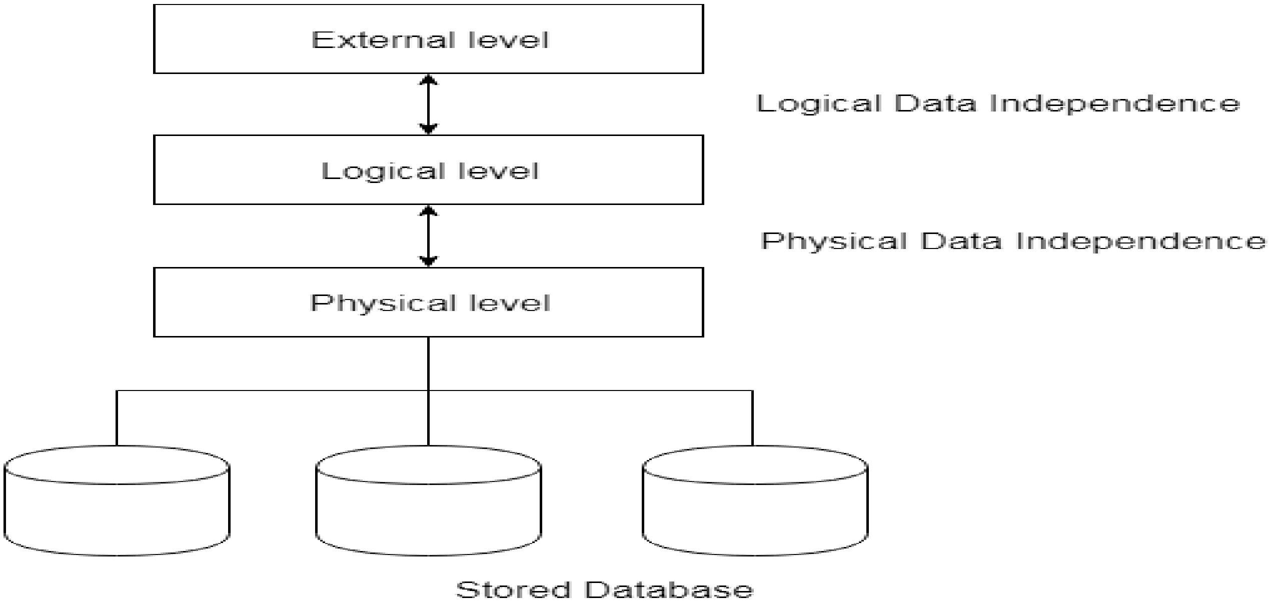


Fig: Three Schema Architecture

There are two types of data independence:

1. Logical Data Independence

- Logical data independence refers characteristic of being able to change the conceptual schema without having to change the external schema.
- Logical data independence is used to separate the external level from the conceptual view.
- If we do any changes in the conceptual view of the data, then the user view of the data would not be affected.
- Logical data independence occurs at the user interface level.

2. Physical Data Independence

- Physical data independence can be defined as the capacity to change the internal schema without having to change the conceptual schema.

- If we do any changes in the storage size of the database system server, then the Conceptual structure of the database will not be affected.
- Physical data independence is used to separate conceptual levels from the internal levels.
- Physical data independence occurs at the logical interface level.

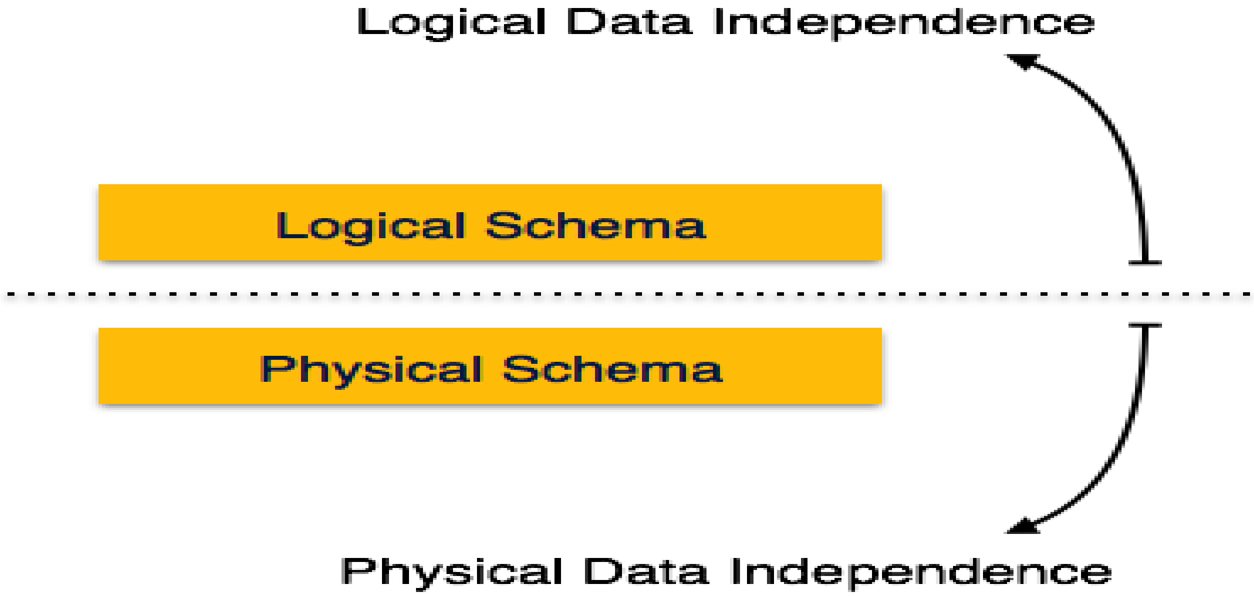


Fig: Data Independence

Oracle Data Types

https://www.w3resource.com/oracle/index.php