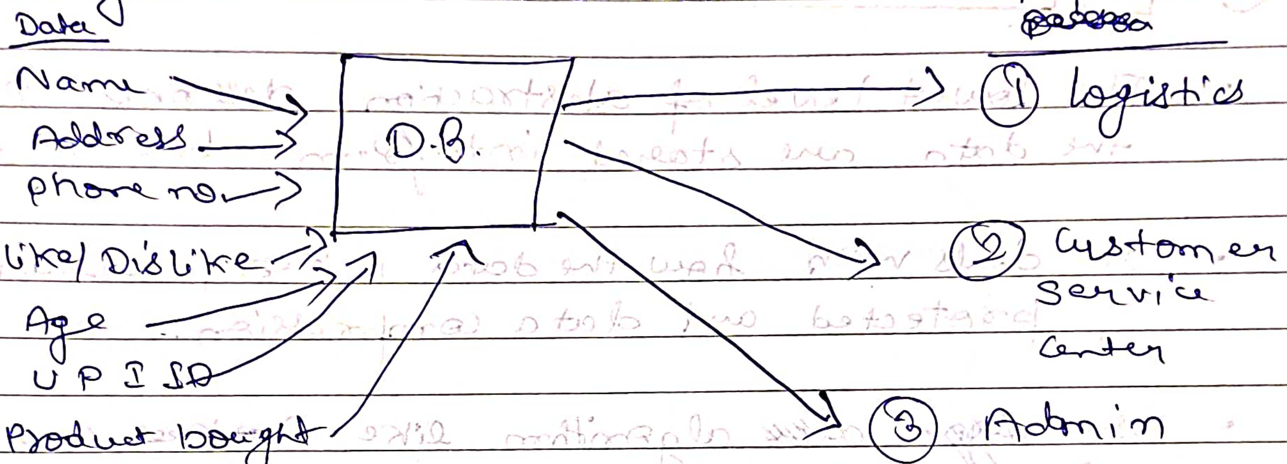
Date ______
Page ______

{Lecture :- 2}

DBMS Architecture

Eg.⇒ let's take an example of ~~an~~ "amazon.com" website. As all the data are stored on a single central database.



Here; ① logistics department will given the data like name, Phone no. and address. That department doesn't need other information like the Age, like/Dislike, UPI ID etc.

In the same way ② customer Service Center needs info like Name, Age, like/Dislike, Product bought. ③ Admin have all the information about the user.

- To hide the data from any particular department, we use the term "abstraction."

- Abstraction ⇒ hiding the complexity of the code or $\overset{data}{\wedge}$ of the program, to make the user interaction more easy.

- In DBMS, there are 3 level of abstraction ① physical level, ② logical level / ~~8~~ conceptual level ③ view level / External level

① **Physical level** :-

- The lowest level of abstraction describes how the data are stored in the form bits & bytes.

- It deals with how the data is kept, arranged, protected and data compression.

- use ~~algo~~ ~~Al~~ algorithm like B-trees and B+ trees, hashing for quickly locate, Huffman coding for compression.

② **logical level** :-

- Describes the design of a database at the conceptual ~~dat~~ level, What data are stored and what relationships exist among ~~a~~ those data.

- Goal : Ease to use.

③ **View level** :-

- Highest level of abstraction, which simplify users' interaction with the system by providing different view to different end-user

- e.g. => ~~A~~ In a big department store with several sections, when ~~yo~~ we enter electronic section, we

See only electronic things and not the other things.

e.g. Physical level

↓

Name , Phone , address , Batch

Logical lelel

Student

S.NO	Name	Phone	address	Batch
1				
2				
3				

• Here, in the physical level only data are stored like name, phone, ... etc. We cann't decide which type of data it is.

In the logical level we grouped it in the table form and define it as a student data, and now we can simply decide it is an student.

→ Instances and Schemas

- The collection of information stored in the DB at a particular moment.

- The overall design of the Database is called DB schema.

- Schema is not frequently change but data may change.

- We have 3 types of Schemas :- Physical, logical and ~~views~~ several view schemas called sub-schemas.

- Logical schema is the most important as programmers construct apps on it and physical schema change does not affect logical schema.

→ Data Models:-

- Provides a way to describe the design of a DB at logical level.

- E.g. ⇒ ER model, Relation model, object-oriented model etc.

→ Database languages

① DDL:- It is used to define and modify the structure of the database itself.

Like creation, altering and deleting tables.

(2) DML :- It is used to work with the data stored in the database.

Like query, insert, update etc.

Both the features are present in a single DB language called, SQL language.

→ **How application programs accessed Database ?**

- Apps written in C/C++/Java/python cannot directly interact with the DB.

- API is used to do this like, open Database ~~[illegible]~~ connectivity (ODBC), made by microsoft for c/c++, Java Database connectivity (JDBC), Java.

→ **Database Administrator (DBA)**

- A person who control of both the data and the program. Perform action like, schema definition, Authorization control, Storage structure, routine maintenance etc.

→ **DBMS Application Architecture.**

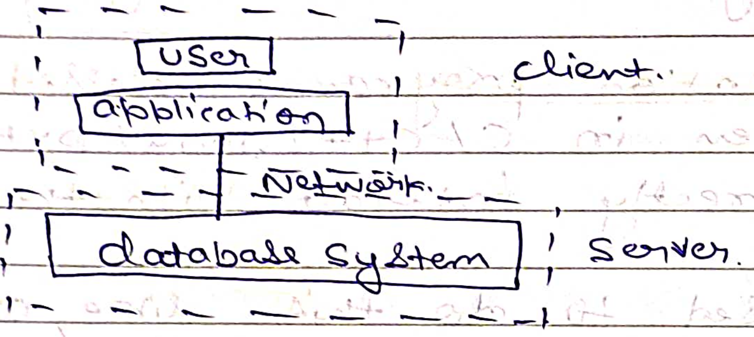
T1, T2, T3

ⓐ T1 architecture

- The client, server & DB all present on the same machine.

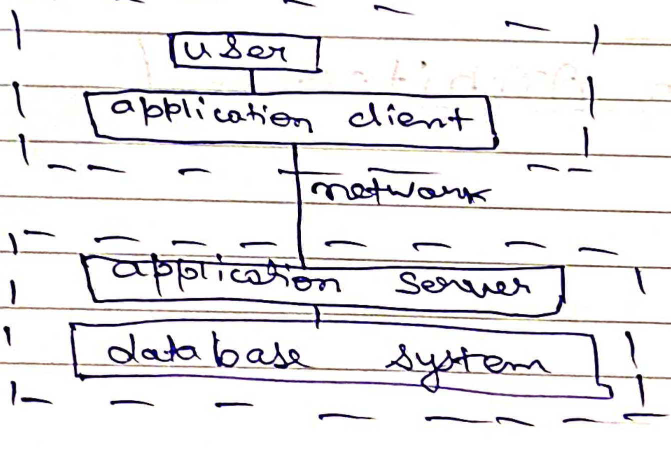
ⓑ T2 architecture

- It is of two - components, client machine invokes DataBase system functionality at Server end through query language stateme
- API's like ODBC & JDBC are used to conne



ⓒ T3 architecture

- It is of 3 - components.
- client is just a frontend and does not do any direct DB a
- App server communicated with DB System.
- T3 arc. best use in WWW application.
- Advantages :- ① Scalability due to distributed app. server
  ② Data integrity
  ③ Security, client can't directly access DB.



user
application client
network
application server
database system