

DBMS Data base Management System :-

Introduction :-

Database :- A database is a structured collection of data that is stored and accessed in a computer system.

Database are used to organize data for companies in many industries, including banking, manufacturing, finance and telecommunication.

DBMS :- DBMS are software systems used to store, retrieve and run queries on data.

- * It A DBMS serves as an interface between an end-user and a database, allowing users to ~~creat~~ create, read, update and delete data in the database.
- * It is a computerized data keeping system.
- * DBMS provides an Interface to perform various operations like database creating, storing data in it, updating data, creating a table in the database and a lot more.
- * It provides Protection and security to the database. In the case of multiple users, it also maintains data consistency.

- * It provides protection and security to the database.
- * In the case of multiple users, it also maintains data consistency.
- * 1960 - Charles Bachman designed first DBMS system.

Component of DBMS :-

1. User :- Application programmers, end user, database administrator.
2. Software :- Controls the storage, management, retrieval of data in database.
3. Hardware :- Input device, output device, network device, storage device.
4. Data :- data stored in database, numerical, non numerical or logical data.

Advantages of DBMS :-

1. Controls database redundancy
2. Data sharing
3. Easily Maintenance
4. Reduce time
5. Backup

1 Disadvantages of DBMS :-

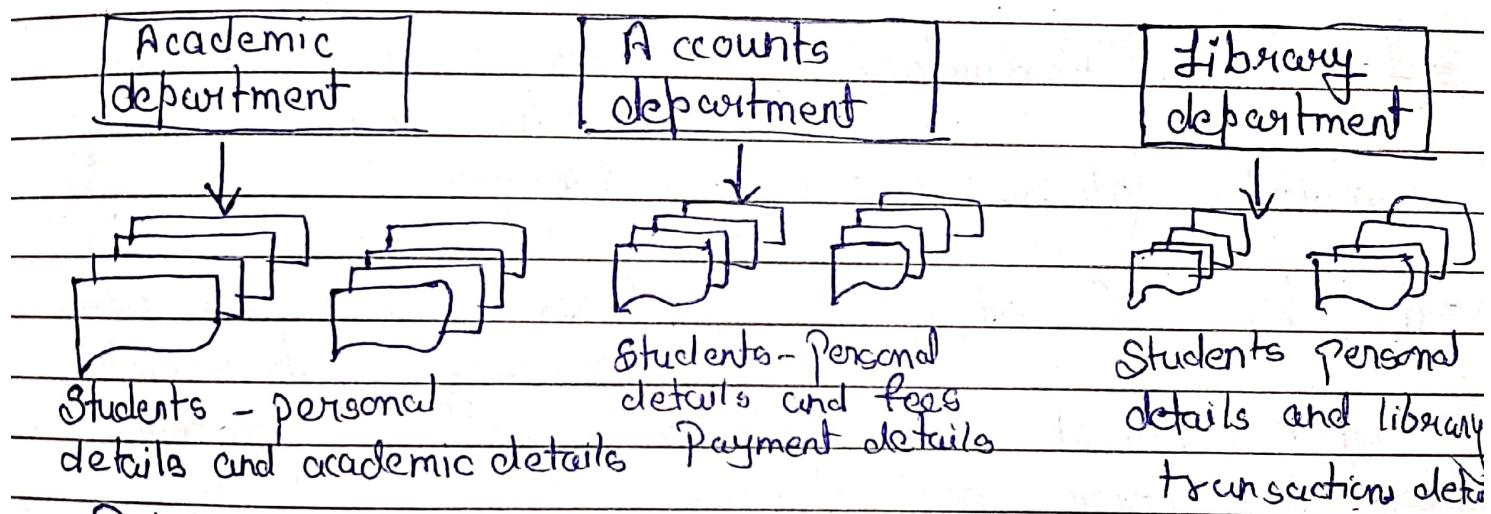
1. Cost of Hardware and Software
2. Size
3. Higher impact of failure

Application of DBMS :-

1. Banking :- For customer information, account activities, Payments, Deposits, Loans etc.
2. Airlines :- For reservation and schedule information
3. Universities :- For student information, course registration, colleges and grades.
4. Telecommunication :- It helps to keep call records, monthly bills, maintaining balance etc.
5. Finance :- For storing information about stock, sales and purchases of financial instruments like stocks and bonds.
6. Sales :- Use for storing customer, product & sales information
7. Manufacturing :- It is used for the management of supply chain and for tracking production of items.

file oriented approach and Database approach :-

File oriented approach :- A file oriented approach to storage creates files in sets as needed when a business sells products or services. Each file operates independently from other files in storage. This means files don't share information with other files stored in the system.



Database Oriented Approach:-

A database is a collection of related data that is organized in a way that allows users to easily access, update, and maintain the data. It stores non-redundant data that can be shared by different application systems.

By non-redundant, we mean that only a single copy of a data exists in the entire system. Non-redundancy ensures data consistency.

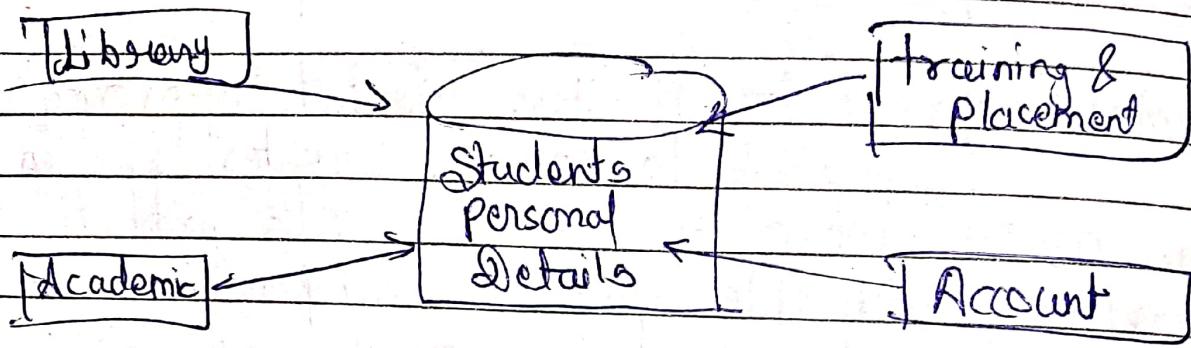


Fig: Database Approach.

Data Redundancy :- Refers to the repetition of the same data in multiple places within a database.

Data Consistency :- Helps ensure that data is accurate, up to date and coherent across different database systems, applications and platforms.

Data Model :-

A data model in a database management system (DBMS) is a visual representation of how data is organized, stored, accessed, and changed within a database.

Categories of data Models :-

There are 3 main categories of data models in which we can categorize data models.

1. High level or conceptual data model
2. Low level or physical data model
3. Representational or implementation data models.

types of Data models :-

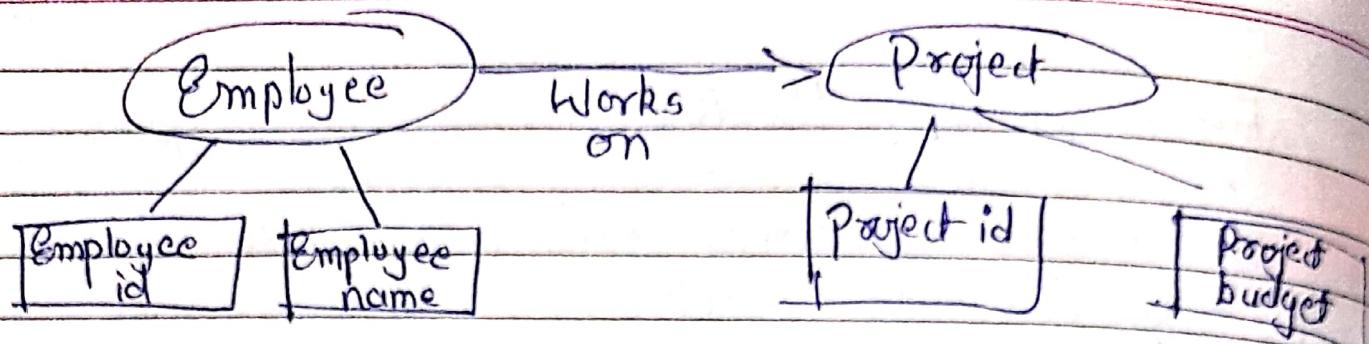
1. Entity Relationship Model 2. Relational data model
3. Network Model 4. Hierarchical model

1. Entity Relationship Model :-

- * In Entity relation data Model, an entity represents a real world object or concept such as an employee or a student.
- * Every entity has its characteristics, which are represented as attributes.
- * The connection between different entities are represented with relations.

Example :-

1. Let say we have these entities:
2. Employee and project
3. These two entities are represented like this:-



2. Relational data Model :-

- * Relational data model uses the concept of database as a collection of tables, where each table can be stored separately as a file.
- * All the database operations are done with relations between the tables.
- * Most relational database use high level language like SQL.

Example :-

Account table		
Account - Number	Branch	Balance amount
1234	Mumbai	400
4567	Bangalore	500
7890	London	600
1234	India	700
9569	Punjab	800

Attribute

domain

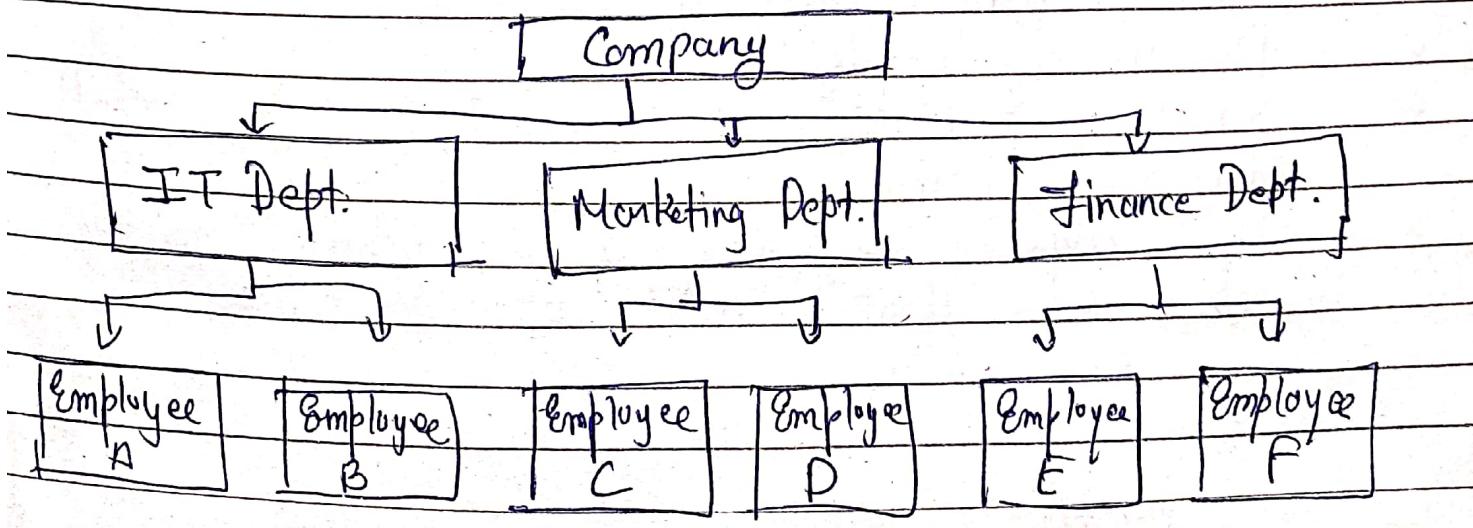
tuple

Relation

Account			Customer Details		
Account Number	Branch	Balance amount	Account Number	Customer name	Customer Mobile number
1234	Mumbai	400	1234	amy	1234357
4567	Baroda	500	4567	seim	128926
7890	London	600	7890	cindy	1349121
12345	London	700	1243	bory	134924

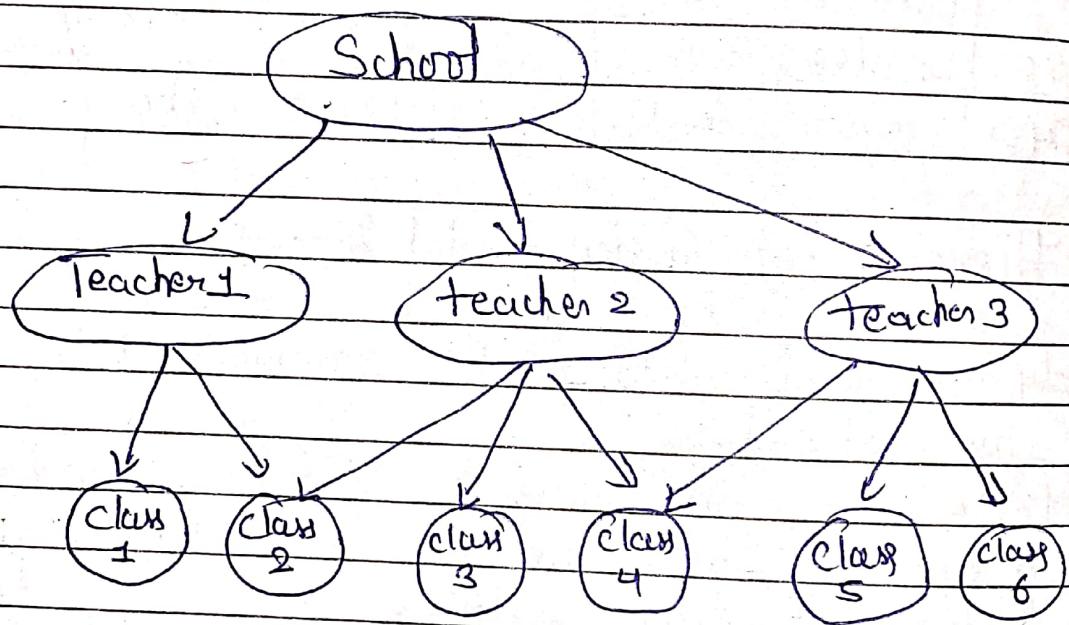
3. Hierarchical Data model :-

- * Hierarchical data model organises data in a tree like structure.
- * Let's say, we have a company and that company has multiple departments. Each department has multiple employees.
- * But one important factor of hierarchical data model is that each child node will have a single parent node.



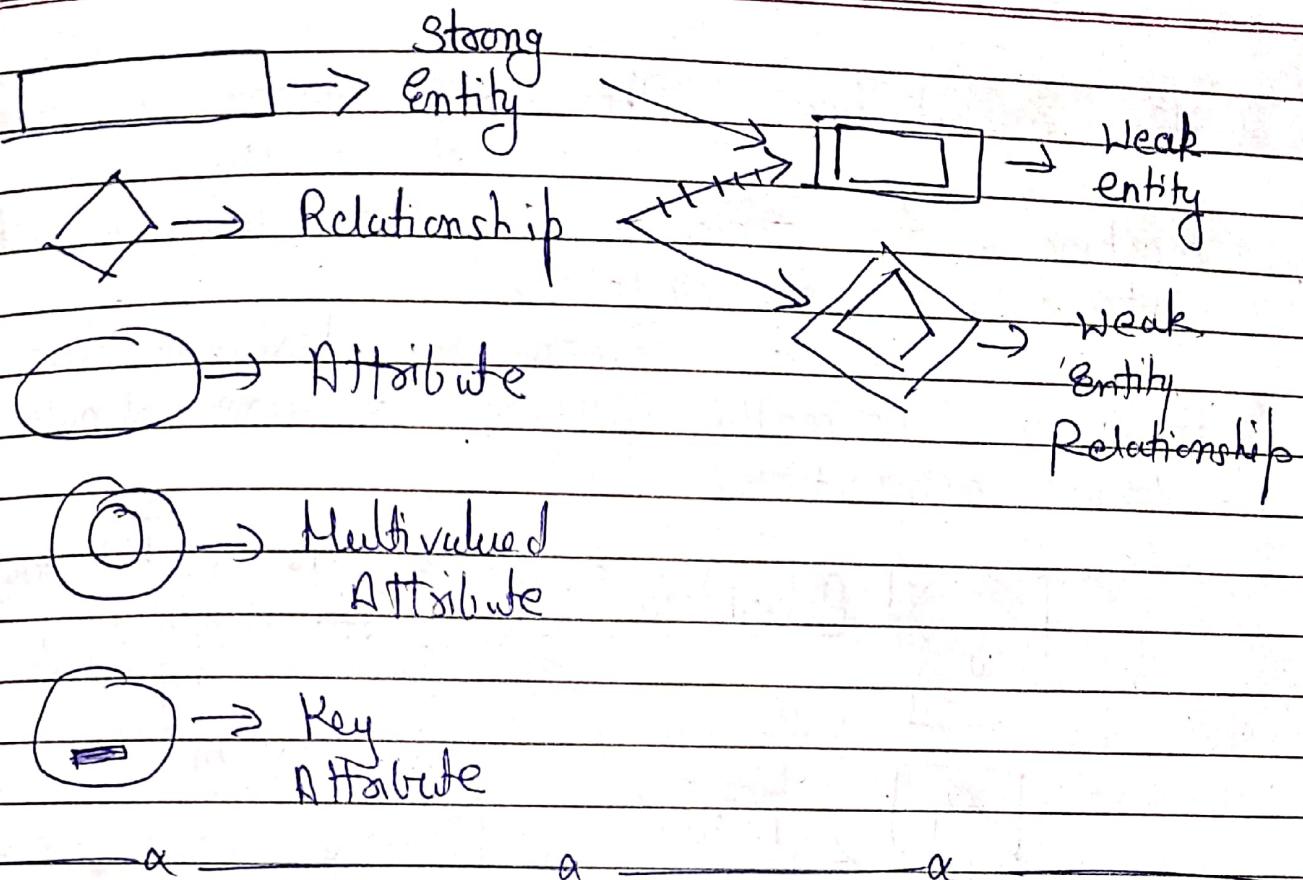
4. Network data model :-

- * Network data model is like a graph and the difference between the hierarchical data model and Network data model is that in network data model, a child node can have multiple parent node.



Other types of Model :-

- * Object - oriented data model
- * Document model
- * Entity attribute value model
- * Star schema
- * The object relation model, which combines the two that make up its name.



Architecture of Database System :-

The Design of a "DBMS" depends on its architecture and it can be centralized or decentralized. It defines how users interact with a "database system" and how data is managed and accessed.

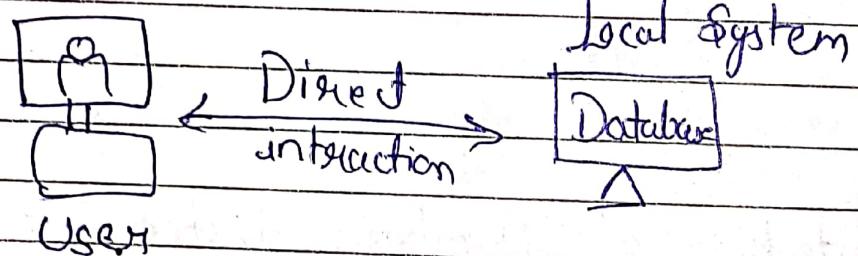
DBMS Architecture derived into three tier:-

- 1. 4, tier architecture
- 2. 2, tier architecture
- 3. 3, tier architecture

① 1 tier architecture :- The client Server and database are all on the same machine. In this architecture user directly interact with the database.

Programmer normally prefer to use single tier architecture.

- Advantages :-
 - ① Simple Architecture
 - ② Cost effective
 - ③ Easy to implement

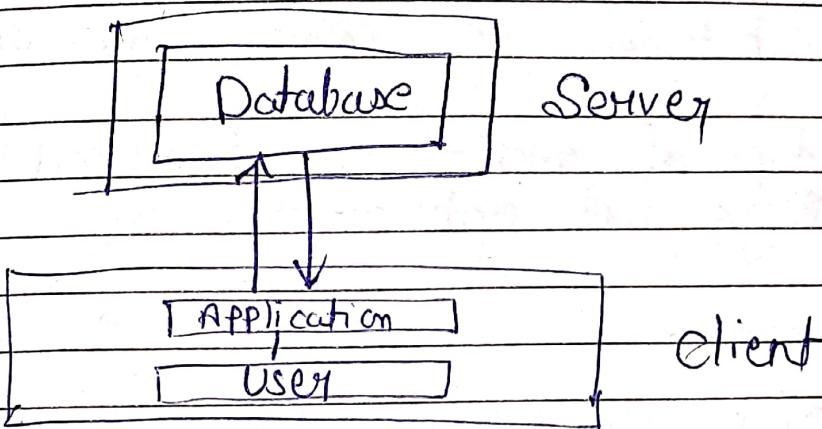


② 2-tier architecture :- In this architecture must have an application through which the database can be access. Here, client directly interacts with the server.

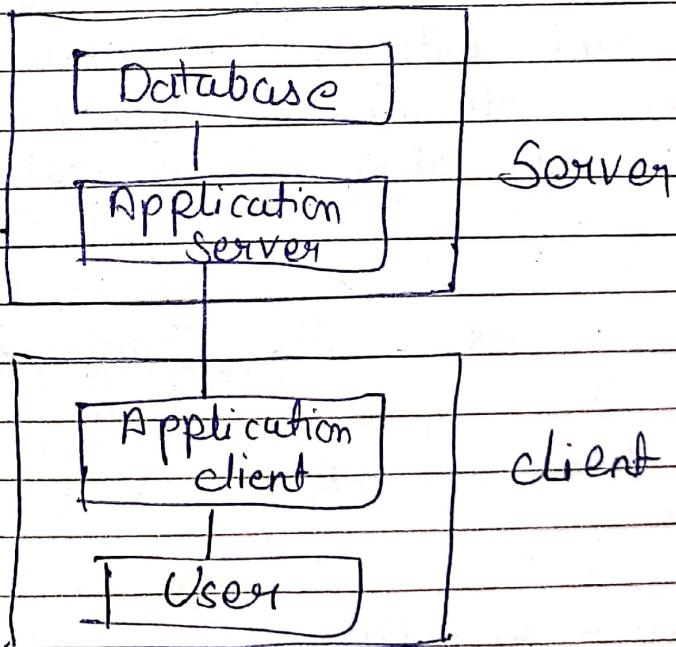
The 2 tier architecture is similar to a basic client server model. The application at the client end directly communicates with the database on the server side. APIs like ODBC and JDBC are used for this interaction.

The server side is responsible for providing query processing and transaction management functionalities.

Advantages :- ① Easy to Access ② Scalable ③ Low Cost
④ Easy Deployment ⑤ Simple



③ 3-tier Architecture :- In this architecture client can't directly interact with server. It separates the database, application logic, and user interface into three distinct layers.



Adv :- ① Security
② Data Integrity
③ Enhanced Scalability.

In, this intermediate layer acts as a medium for the exchange of partially processed data between the server and the client.

This type of architecture is used in the case of large web application.



DBA (Database Administrator) :-

A Database administrator (DBA) manages computer databases to ensure they run efficiently and securely.

A DBA is an individual person or a group of persons with an overview of one or more databases so that he / she can control the design and the use of these database.

DBA Control the database structure and sets up the definition of physical as well as logical implementation of the database.

Function of an DBA :-

- (i) Defining Conceptual schema
- (ii) Physical database design

- (iii) Security and Integrity checks } authentication
- (iv) Backup and Recovery strategies } authorization
- (v) Controlling User Access {
 - U_i → select
 - U_j → delete

Data independence :-

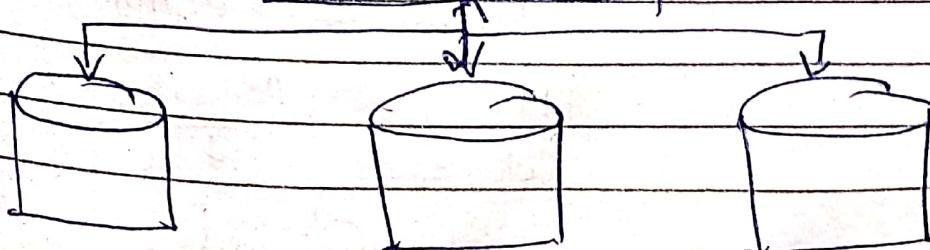
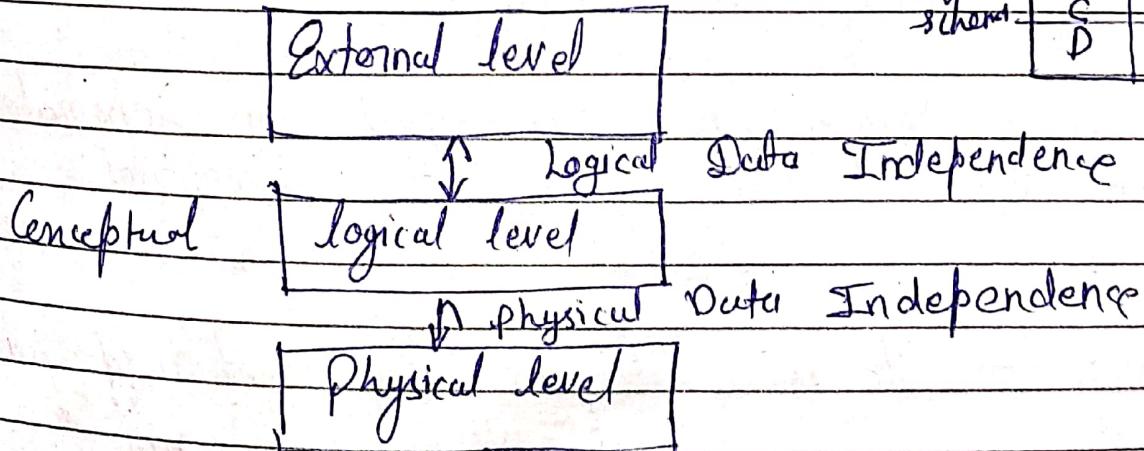
- * The ability to modify a schema definition in one level without affecting a schema definition in the next higher level is called Data Independence.
- * Data independence is one of the main advantages of DBMS.

Data independence is two types:-

- * physical Data independence
- * logical Data independence

Name	Roll	Branch	Address
A	1	X	P
B	2	Y	Q
C	3	Z	R
D	4	U	S

Physical schema



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.

It occurs at the logical Interface level.

Logical Data Independence :- It is the ability to modify the conceptual schema without causing application program to be rewritten.

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.

Schema :- It is a logical Representation of database.

RDBMS

tables

Student ← entity

Rollno	Name	Address
--------	------	---------

Course ← entity

Cid	Name	Duration
-----	------	----------

Data Dictionary → A Data dictionary contains metadata i.e. data about the database.

- # It contains information such as what is in the database, who is allowed to access it, where is the database physically stored etc.
- # It plays an important role in building a database.

→

Field	client_id	client_name	Password	Contact no.	Email
1	Rajiv	12345	777776	rajiv@gmail.com	
2	Rohan	AbcdEf	666667	rohan@gmail.com	
3	Sohan	5432)	555552	sohan@email.com	
4	Mohan	14145	222225	mohan@email.com	

Field Name	Data type	Field length	Constraints	Description
client_id	number	10	Primary Key	client_id, Auto generated
client_name	varchar	20	not null	Name of client
Password	varchar	30	not null	Logic password
Contact_no	number	10	not null	Contact of client
Email	varchar	40	not null	client email

The data dictionary in general contains information about the following:

- # Names of all tables in the database
- # Names of each field in the tables of the database

- * Constraints defined on tables.
- * Physical information about tables, like their storage location, storage method etc.
- * A data dictionary is also called a metadata repository.

There are two types of data dictionary:-

- ① Active :- It may happen that the structure of the database has to be changed, like adding new attributes or removing older ones. If those changes are updated automatically in the data dictionary by the DBMS, then the data dictionary is active one.
It is also known as integrated data dictionary (updated).
- ② Passive :- When the DBMS maintains the data dictionary separately and it has to be updated manually, then the data dictionary is an passive one.
It is also known as non-integrated data dictionary.

Primary key :-

Key :- Key is the data item in RDBMS which exclusively identify the records of tables.

① Primary Key :- Primary key is the key field which uniquely identify each record of table.

Example :- Student table

Select * from student;

	Roll No	Name	Add	Phone
Primary Key	1024	ANISH	Delhi	6201
	2024	ANKIT	Mumbai	7034
	3024	ANKUSH	Groh	2024
	4044	AKHIL	TATA	5674
	5042	KISAN	Jammu	7899

SQL > Create table Student (roll number primary key, name varchar(12), address varchar(22), phone number);

An attribute or set of attribute that uniquely identifies a row of record in a Relation is known as primary key.

Some most important points about primary key are:-

1. A Relation can have only one primary key.
2. Each value in primary key attribute must be unique.

3) primary key can not contain null values:-

(*) other types of key:-

- (2) Candidate Key
- (3) Alternate Key
- (4) Composite Key
- (5) Foreign Key
- (6) Unique Key

Database language:-

1. DDL (Data Definition Language) :-

SQL commands are essential for managing database effectively. The SQL stands for Structured Query Language.

These commands are divided into categories:-

- * DDL (Data Definition Language)
- * DML (Data Manipulation Language)
- * DCL (Data Control Language)
- * DQL (Data Query Language)
- * TCL (Transaction Control Language)

1. DDL :- DDL or Data Definition Language actually consists of the SQL commands that can be used to creating, defining, altering and deleting database structures such as tables, indexes and schemas.

Common DDL Commands :-

- * Create & Drop
- * Alter & Truncate
- * Rename

(a) CREATE :- It is used to create a new table in the database.

Syntax :- CREATE TABLE TABLE-NAME (COLUMN NAMES DATATYPES [...]);

(b) DROP :- It is used to delete both the structure and record stored in the table.

Syntax :- DROP TABLE table-name [cascade constraint];

(c) Alter :- It is used to alter the structure of the database. This change could be either to modify the characteristics of an existing attribute or probably to add a new attribute.

Syntax :- ALTER TABLE table-name ADD column-name column-definition;

(d) TRUNCATE :- It is used to delete all the rows from the table and free the space containing the table.

Syntax :- TRUNCATE TABLE table-name;

(e) Rename :- It is used to rename the table.

Syntax :- Rename <OLD-TABLENAME> to <NEW - TABLENAME>;

2. Data Manipulation language :- DML commands are used to modify the database. It is responsible for all form of changes in the database.

Common DML Command :-
 * Insert + Update
 * Delete + Select

(a) INSERT :- The INSERT statement is a SQL query. It is used to insert data into the row of a table.

Syntax :- To add row in a table

`INSERT INTO TABLE_NAME
 (col1, col2, col3, -- colN)
 VALUES (value1, value2, value3, -- valueN);`

(b) UPDATE :- This command is used to update or modify the value of a column in the table.

Syntax :- to update record in a table

`UPDATE table-name SET [column-name1 =
 value1, -- column-nameN = valueN]
 [WHERE CONDITION]`

Example :-

`UPDATE students
 SET User-Name = 'Sandeep'
 WHERE Student-ID = '3'`

(c) DELETE :- It is used to remove one or more row from a table.

Syntax :- To Delete a record from table.

DELETE FROM table-name [WHERE condition];

Ex :- DELETE FROM javapoint
WHERE Author = "SOMOV";

(d) SELECT :- It is used to select the attribute based on the condition described by WHERE clause.

Syntax :- SELECT expression
FROM TABLES
WHERE conditions;

Ex :- SELECT emp-name
FROM employee
WHERE age > 20;

Cloud Computing :- Cloud computing is the delivery of computing resources over the internet, or "the cloud", on an as-needed basis.

It allows users to access computing resources like storage, databases, software and networking without having to purchase and maintain their own physical infrastructure.

Advantages :-

- ① Cost saving
- ② Scalability
- ③ Flexibility
- ④ Innovation

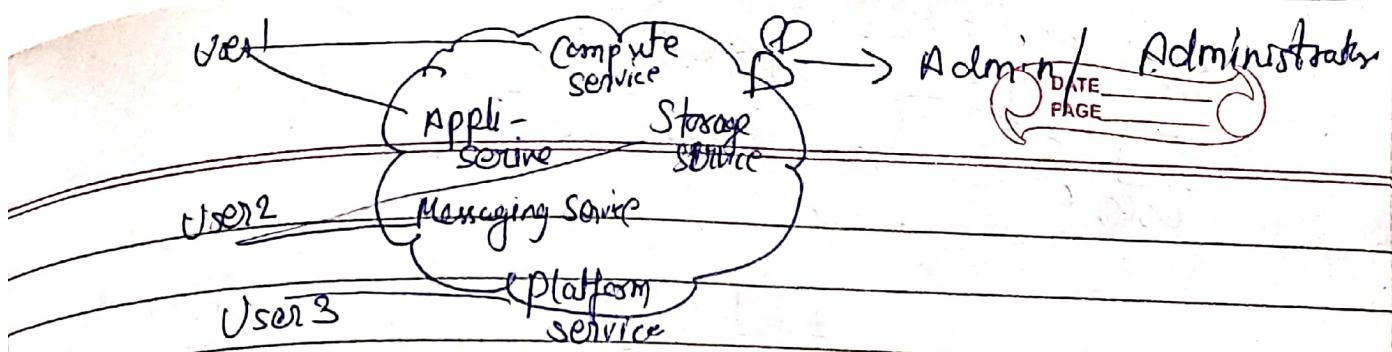
Some examples of cloud computing include:-

- ① Google Docs and Microsoft 365
- ② Gmail, calendar, Skype, and WhatsApp
- ③ Zoom meeting
- ④ AWS Lambda
- ⑤ Salesforce

Types of cloud :-

- ① public cloud :- open to all to store & access information via internet
 - pay as per use (for the services)
 - managed by third parties (cloud service provider)

Ex:- dropbox, Google drive etc.



Advantage :- ① It is maintained by cloud service provider. So we need not maintain it.

② location independent b/c its services are delivered through the internet.

③ high Scalability (eg → gmail offers 15gb. we can increase anytime & dec also after increasing)

④ Cost effective

Disadvantage :-

- ① Less secure b/c resources are shared publicly.
- ② less customizable as compared to private cloud.

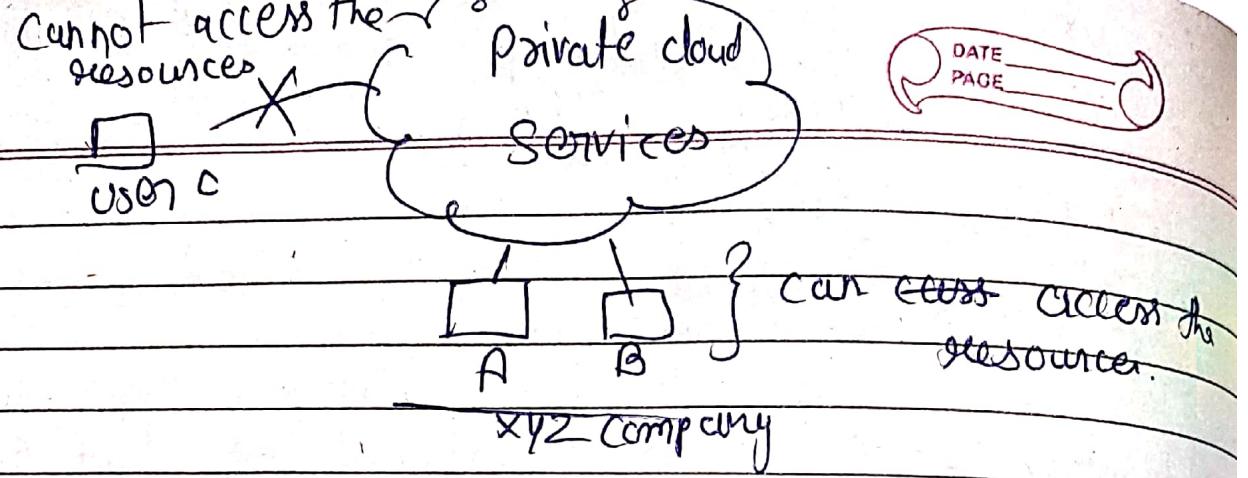
private cloud :-

* Services accessible within an organization i.e.

it belongs to a specific organization

* Sometimes also called ^{internal} internet / corporate cloud.

* Can be managed by → organization, 3rd party also



Advantages:- ① high Security ② data privacy
③ more customizable

Disadvantages:- ① High Cost ② Limited Scalability

③ Hybrid cloud :- It is combined the feature of public & private cloud.

⇒ Critical activities performed by private cloud
non critical activities performed by public cloud.

Advantages :- * Scalability * Security * Low cost
* Flexibility

Disadvantage :-

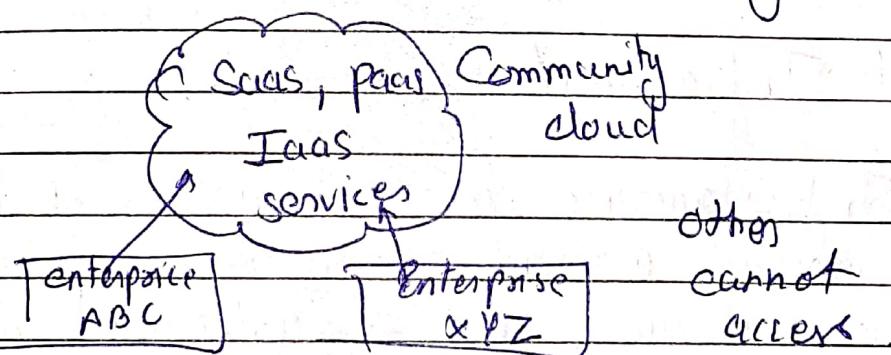
A Managing is difficult

At Complex

* dependency on infrastruct structure.

④ Community cloud :-

- Allows service to be accessible by a group of several organization to share the info between the organization & a specific community.
- Owned, managed & operated by one or more organizations in the community or 3rd party.



Advantages :-

- ① Cost reduction / Cost effective :- It is cheaper than private cloud. (Multiple companies share the bill, which lowers the cost)
- ② Sharing among companies (the resources)
- ③ More secure than public cloud but less than private cloud.

Disadvantages:-

- ① data is accessible between organization.
- ② Overall increased cost.
- ③ Consistent Maintenance cost.

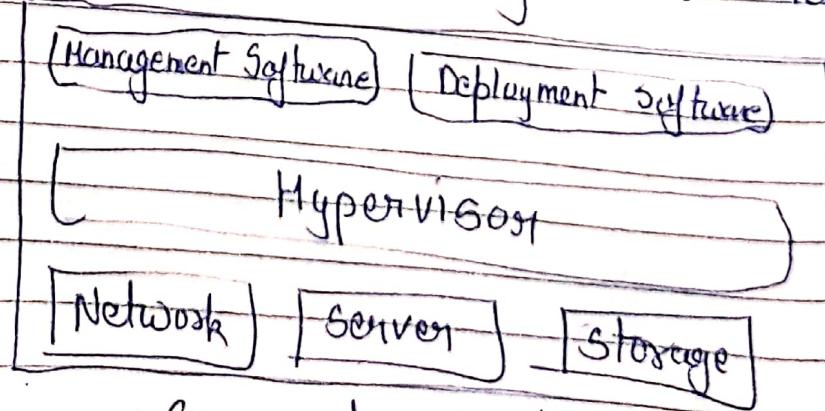
Cloud Infrastructure :-

Cloud Infrastructure is the collection of hardware and software elements such as computing power, networking, storage, and virtualization resources needed to enable cloud computing.

Cloud Infrastructure types usually also include a user interface (UI) for managing these virtual resources.

Cloud infrastructure components :-

1. Computing
2. Networking
3. Storage



Components of cloud Infrastructure

1. Hypervisor :- It is used to divide and allocate cloud resources between several customers.

As it monitors and manages cloud services / resources that's why hypervisor is called as VMM (Virtual Machine Monitor) or (Virtual Machine Manager)

2. Management Software :- Cloud management Software monitors and optimizes resources, data, applications and services.

3. Deployment Software :- Helps in deploying and integrating the application on the cloud.

4. Network :- It is one of the key component of cloud infrastructure which is responsible for connecting cloud services over the internet.

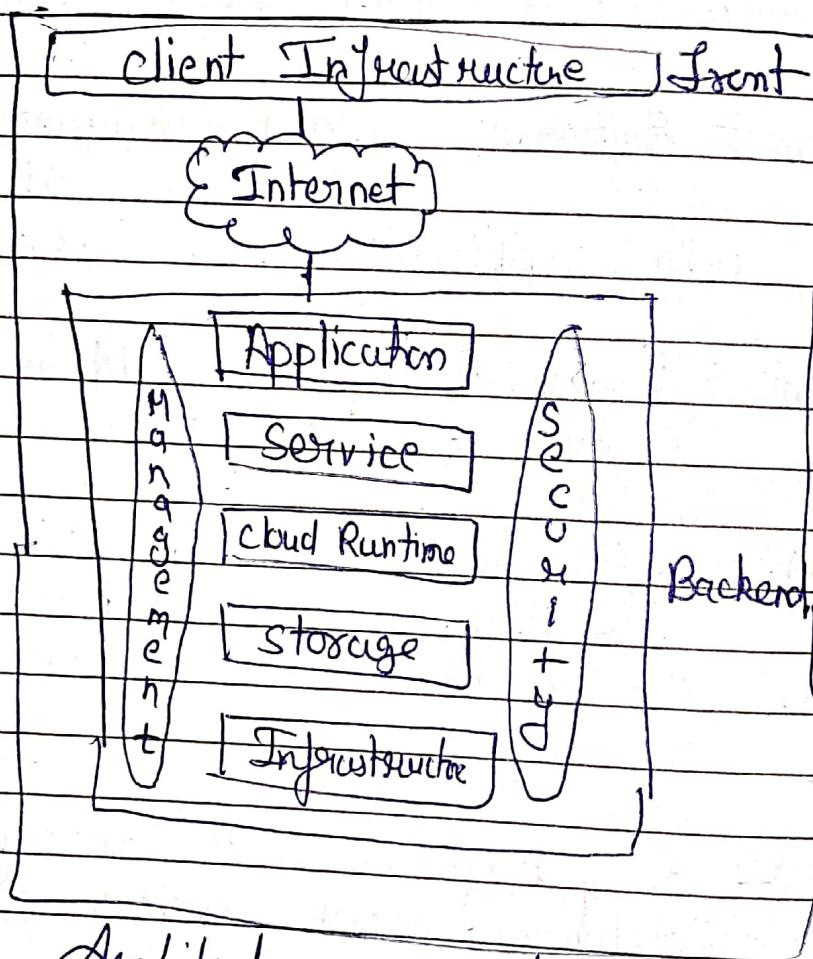
5. Server :- Is responsible for managing and delivering cloud services for various services and partners, maintaining security etc.

6. Storage :- Represents the storage facility which is provided to different organizations for storing and managing data.

cloud Computing Architecture :-

The cloud architecture is divided into 2 parts

1. Frontend
2. Backend.



Architecture of cloud Computing

1. Frontend :- Frontend of the cloud architecture refers to the client side of cloud computing system. Means it contains all the user interfaces and applications which are used by the client to access the cloud computing services/resources. For example, use of a web browser to access the cloud platform.

2. Backend :- Backend refers to the cloud itself which is used by the service provider. It contains the resources as well as manages the resources and provides security mechanisms.

Along with this, it includes huge storage, virtual application, virtual machines, traffic control mechanisms, deployment models etc.

Cloud Segments or Service Delivery model:-

There are the following three types of cloud service models:-

1. Infrastructure as a Service (IaaS)
2. Platform as a Service (PaaS)
3. Software as a Service (SaaS)

① IaaS :- IaaS is also known as Hardware as a service (HaaS). It is a computing infrastructure managed over the internet. The main advantage of using IaaS is that it helps users to avoid the cost and complexity of purchasing and managing the physical servers.

Example :- Digital Ocean, Amazon Web Services (AWS), Microsoft Azure, Google Compute Engine (GCE) etc.

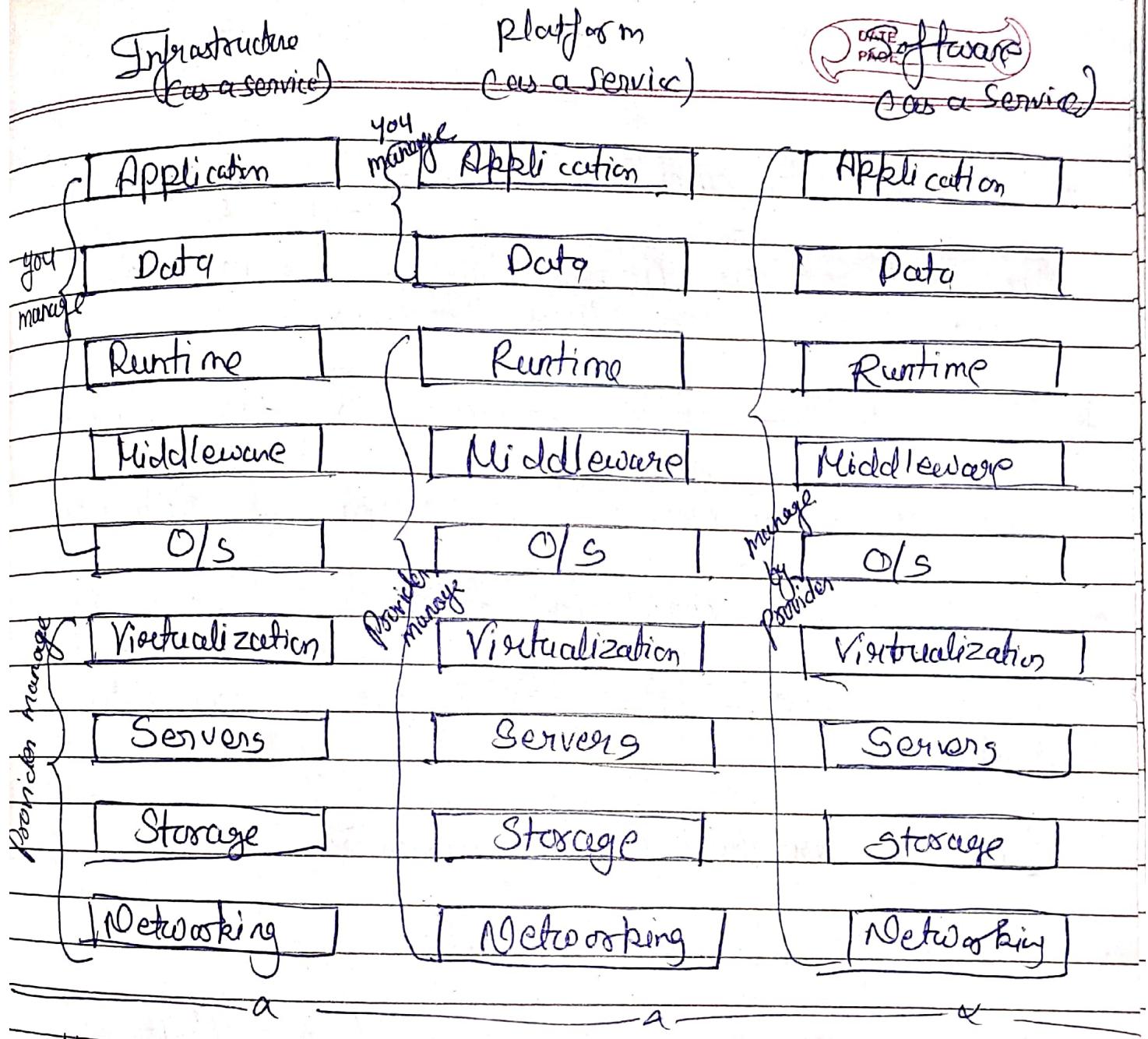
(2) PaaS :- PaaS cloud computing platform is created for the programmer to develop, test, run and manage the application Accessible to various users via the same development application.

Integrates with web service and database. Support multiple languages and framework.

Example :- Windows Azure, Force.com, Apache Struts, Google App Engine

(3) SaaS :- is also known as "on-demand software". It is a software in which the applications are hosted by a cloud service provider. Users can access these applications with the help of internet connection and web browser.

Example :- WebEx, Dropbox, Google Apps, BigCommerce, Salesforce etc,



pros and cons of cloud Computing :-

Pros (Benefits) :- of cloud Computing

* Cost efficiency :- cloud computing can help businesses shift from capital expenses to predictable monthly costs.

* Scalability :- cloud system can grow with your business.

- * Accessibility and mobility :- User can access application through the web or an API, without needing to install them on their devices.
- * Security :- cloud storage is often backed up regularly, stored off-site, and monitored for suspicious activity.
- * Disaster recovery and data backup :- cloud computing can help with disaster recovery and data backup.

Cons (drawback / Disadvantages) :-

- * Data privacy and security concerns :- There are risks of data loss, theft, leakage, and accidental exposure.
- * Bandwidth limitations :- There may be bandwidth limitation.
- * Downtime and reliability
- * Limited control :- There may be limited control over the cloud.

- + Weaker Security :- Some customers, like financial institutions, may be concerned about trusting a third party.
- + Dependency on service provider.