Introduction To Databases

Applications of Data Base Systems:

1. Traditional Database : Which are responsible to store and retreive textual data.
2. Multi media Databases: Which are able to store audio / vedio clips or images
3. Geographical Information System: Which are able to store maps, process and analyses map. And store and process satellite images and weather forecasting data.
4. Real And Active Database: Controlling manufacturing process in industries
5. Data Klare House: It is a large database containing transactions related to organisation / enterprenuer. It is mainly used in analysis purpose to develop business etc. It provides data for decision making.

→ Data Base: Collection of related data.

→ Data: Known facts to be recorded.

* Data base is logical coherent

* Database is specific to application.

* DBMS is general purpose software which is used for creating, manipulating, maintaing and sharing the database among various users. enables

* During creation time user specify some data such as name, columns, constraint etc called Metadata which is stored in "System Catalog".

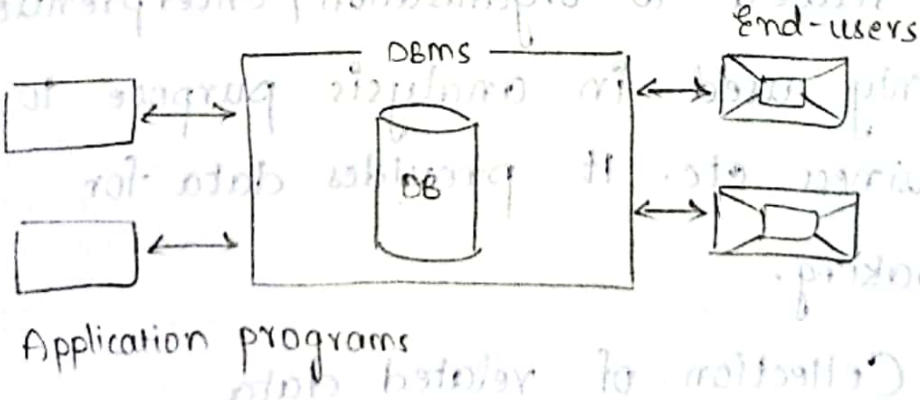
* Data about data is known as 'Meta data'.

* After creation DBMS software is responsible for retreiving, inserting, deleting, updating operations

* How the information is stored in server memory is called maintainance.

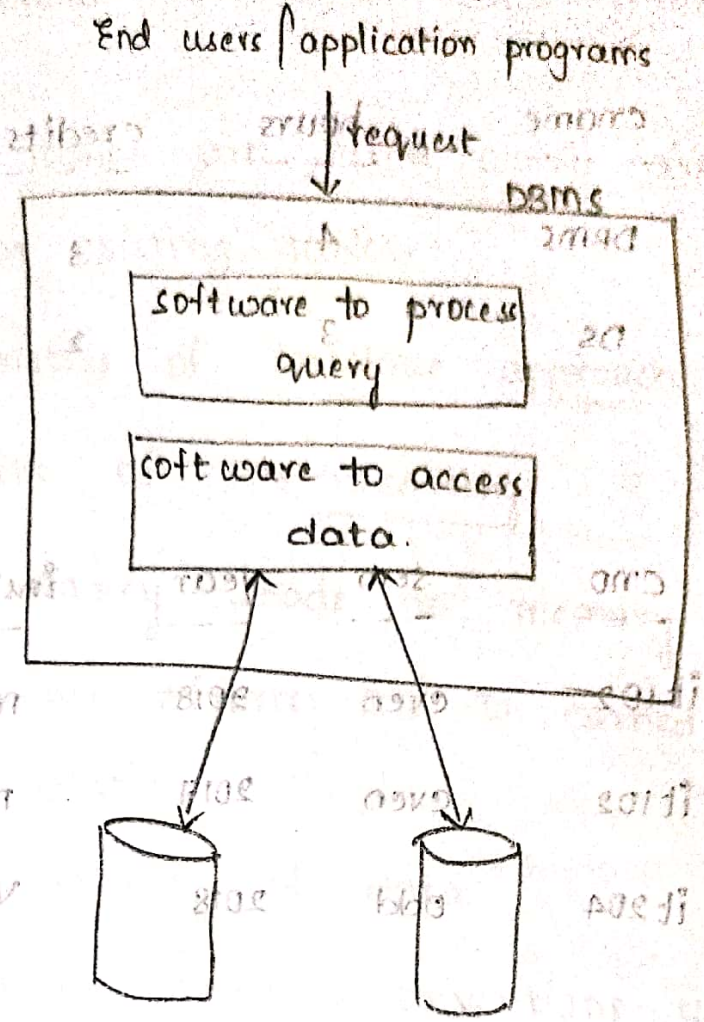
* DBMS software is responsible for sharing of data.

* Simplified view of DB System:



* DBMS also provides security related to modules i.e to pretend the unauthorized attacks.

* DBMS also provides protection from system / hardware failures i.e through recovery modules.



* Generally data is stored in the form of files.

→ Example DataBase – University DB

* Keep track details concern to students, courses, sections, grades, pre-requisited details as five DB files or tables.

Student:

SNO	sname	class	Branch/major
y18it01	smith	1	it
y17cs32	brown	2	cs

Courses:

cno	cname	hours	credits
it102	DBMS	4	3
it204	DS	3	2

Sections:

sid	cno	sem	year	instructor
121	it102	even	2018	mpp
122	it102	even	2017	nn
102	it204	odd	2018	vss

Grades:

sno	cno	sid	grade
418it001	it102	121	O
418it001	204		A

Pre-requisite

cno	prerequisite
it102	it204
!	

* DBMS allows to create views. Views are virtual tables

* Virtual tables are tables which are derived from base or existing tables.

→ Characteristics of Database approach with File Processing

* Drawbacks of FileProcessing:

1. Redundancy leads to memory wastage
2. Duplicate efforts are to carried during manipulation.
3. Inconsistancy of data
4. Naming conflicts conventions used for data fields may differ in files which leads to inconsistency.

All these drawbacks are avoided with databases

* Characteristics of DB :-

1. Self describing of Dr nature of DB
2. Provides insulation between programs & data also it provides exhibiting data abstraction.
3. Support of multiple user views.
4. Sharing data in multi user environment.

1. → All the info about student table is stored in system catalog.

Student - Each record

column	starting byte	length
sno	1	10
sname	11	10
class	21	1
major	22	3

2. → Metadata is stored in system catalog and application programs are stored in another way so this shows independency among them. This will be exibited through Program Data Independence and Program operation independence. Both these will come under "Data Abstraction". The software regarding a particular operation is hidden from user through program operation independence.

4. → Same database is accessed by multi users at a same time. The control access by users is done. Concurrency control module helps to control accessing.

→ Database Users:

Database users

Actors on the scene

1. Database Administrator
2. System Analyst
3. Application Programmers
4. Database designers
5. End users

Workers behind scene

Responsible for

1. Usage of Database & DBMS
2. Creating Authorized users for access of DB - creating user accounts through DBMS software.
3. Coordinating and maintaining database.
4. Handling poor response time for DBS
5. Recovery of DBS during failures
6. Creating a database

System Analyst & AP

— Who is responsible for interacting with end users community and analyse the data to be stored and how the data is stored what type of data to be transferred.

→ DB Designers: Identifying the data to be populated along db table. Also identifying views of various end user community.

→ End users: Responsible for who are responsible for retreiving data or accessing the database

@ Casual End users: Whose access information differs

from time to time.

(b) Naive / Parametric End users: Users who are responsible for accessing a portion of p database for executing canned transactions. A transaction which is to be executed repeatedly in order to complete a process is known as canned transaction.

(c) Sophisticated End Users: Scientific Engineers, Business analysts

(d) Standalone users: maintains individual db. Limited to their organization.

Workers Behind the Scene:

Group of persons responsible for developing various modules and other utilities known as workers behind the scene.

1. DBMS System Engineers & Implementors: System catalogs, recovery modules etc are maintained by them.
2. Tool Developers: To perform modelling seperate tools are required
3. Operators & Maintainance persons: Monitoring the usage of data through DBMS. Updations & records of previous operations are maintained. Maintain backup copies of data

→ Advantages of Database Management System:

1. Controlling redundancy is acheived.
2. Restricting unauthorized access

Data normalization is used to organize data with minimum redundancy.

3. Providing persistent storage for program objects.
4. Providing storage structures and search techniques fore efficient query processing.
5. Provides backup and recovery.
6. Multiple user interfaces
7. Easy to represent complex relationships among data.
8. Easy to enforce integrity constraints and business rules.
9. Permitting interfacing and actions using rules
   - Deductive DB system
   - Active DB system

Deductive DB : From available data derive new data

Active DB : Generally used in manufacturing plants, decisions are taken on previously available data.

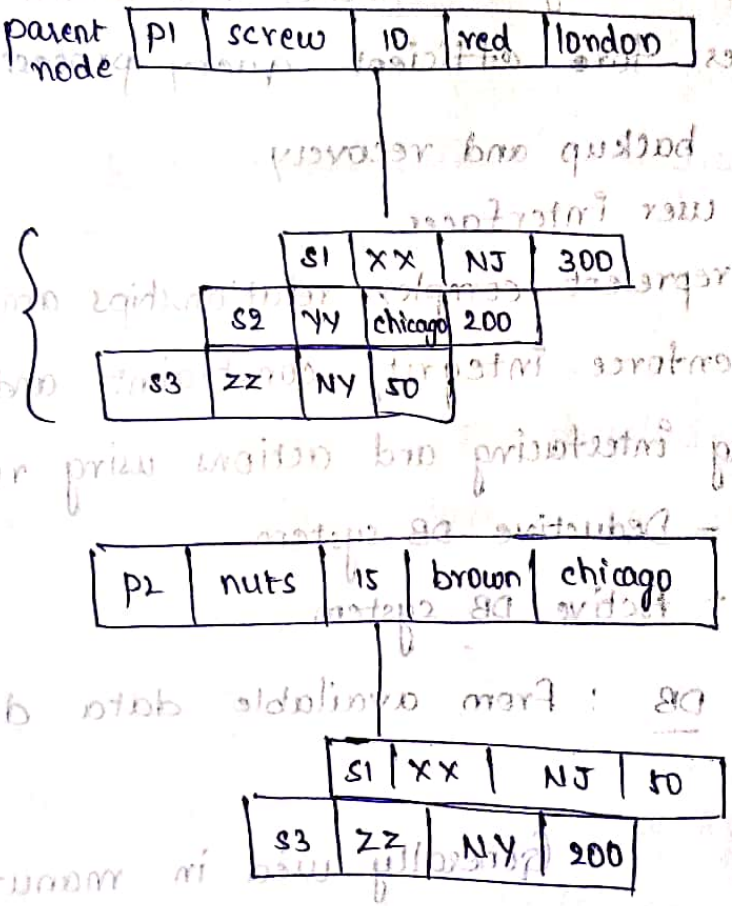
→ History of DBMS applications:

I Based on data model:-

Data model is collection of concepts which can be used to describe structure of database in order to achieve level of dataabstraction.

1. Earlier database applications using hierarchial model
   - Hierarchial model is given by IBM in 1968 used in university, banking, hospitals, trading - - - etc.
   - Data is organized in hierarchial tree structure in which parent & child relation ship are defined explicitly among rows / records of database.
   2. Atmost a child can have only one parent. due to this redundancy is more.

Eg:



Drawbacks: 1) It is not flexible enough to represent all the relation ship proportions.

2) It cannot demonstrate the overall data model for enterprise because of non availability of actual data at the time of designing a data model.

3) It cannot represent many - many relation ships.

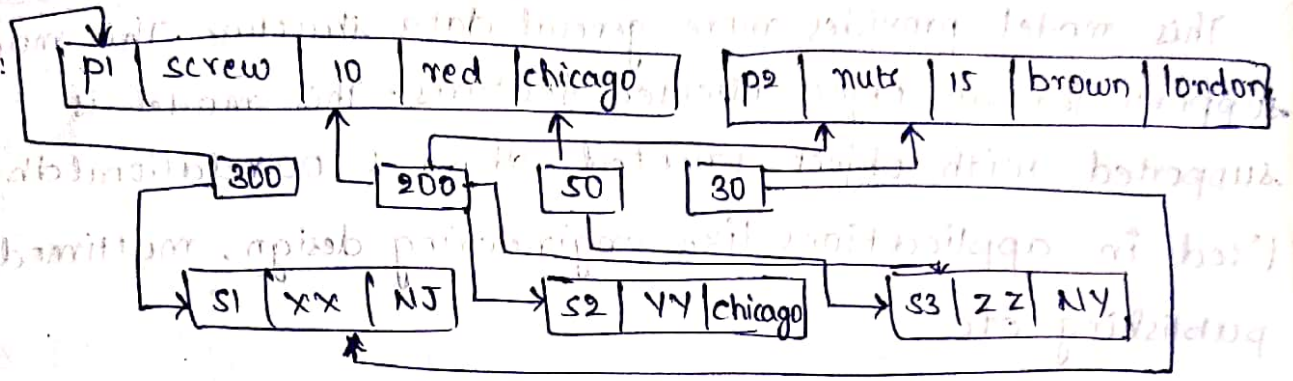
4. This model is used only when concerned data has

clearly hierarchial nature with a single root node.

2. Earlier DB applications using 'Network data model'-

In this model data is organized in graph structure by defining multiple parent-child relation ships. This model provides easy access to data due to multiple access paths available to data elements.

Ex:



Ex: university, banking, hospitals, trading - - - etc.

Drawbacks: 1. Intermixing of conceptual relational ships with physical storage and placement of records on disk may require more tracing time or retrieval time

2. Difficult to formulate new kinds of queries, if needed compulsarily to be implemented through different application programs.

3. Relational Database - In this datamodel data is organized in the form of rows & columns in a table known as a relation. There will be no hard coded relationships among the tables. This model allows to define a relationship between one table with another table based on columns of a table through referential integrity constraint (foreign key approach)

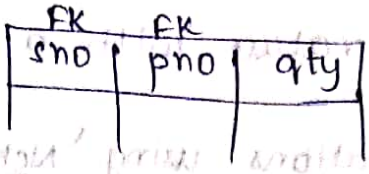
Ex: Supplier table -

PK			
sno	sname	status	city

parts table -

pno	pname	pcolor	pweight

supply table -



This model separates physical storage of data from contextual representation. It exhibits higher level data abstraction. It supports for application flexibility.

4. Database Applications using 'Object Oriented Data model'

This model provides more general data structure. This model supports for all object oriented features. This model is supported with object oriented dbms & OO relational dbms Used in applications like engineering design, multimedia publishing etc.

II Based on E-commerce application using XML databases:

XML db supports for interchanging data on web through XML files. XML db also supports interchanging of data among various types of db and with webpages also.

III Emerging applications:

1. Scientific applications - Human gene mapping
High energy physics.

2. Multimedia database applications - Allow storage & retrieve audio & vedio / image file.

3. Datamining applications - Which derive new kind of knowledge from available data.

Ex: Identifying the outlier from shopping data which are deviated from regular behaviour

4. Spatial db - Maps, satellites data

5. Time-series db - which is to be collected & stored in a time interval.

IV 6. Information retrieval system (IR) - Information retrieval deals with books, manuscripts, various forms of library based articles. In IRS the data is indexed, catalogued and annotated with key words. IR is concerned with searching for material or documents based on the key words

Ch 2

→ Overview of Database Languages and Architectures:

* Data Models, Database Schema, Database State:

- Data model:- is a collection of concepts used to describe the structure of DB, in order to achieve data abstraction.

Structures:-
1. Relations
2. Data elements (Attributes)
3. Types
4. Constraints
5. Storage organisation
6. Operations — Basic set of operations
   — Dynamic operations

3 kinds of data models
1. High level data model or conceptual data model
2. Low level data model or physical data model
3. Representational data model or Implementational data model

1. → Structure in high dm will help user to perceive nature of data available in db.

The structure in conceptual dm is given in the form of Entity, Attributes, Relations

Entity: physically/conceptually existing object instances for which we need to record data.

Attributes: Dataelements used to describe the enity

Relations: How one entity is related to other entity

2. → It is concerned to db developers to perceive how data is organised internally.

Physical model will specify data in record is structured. also contains indexes of columns of db table.

3. → 1. Hierarchial dm
2. Network dm
3. Relational dm — object based model
   - i) OO DBMs
   - ii) OO RDBMs

- Database Schema:-

DB System → DB structure + Actual DB

↓

DB Schema

Displayed DB Schema is called Schema diagram in the form of diagram

Student

sno	sname	class	branch

course

cno	cname	credits	marks	dept

section

sid	cno	instructor	sem	year

Grade report

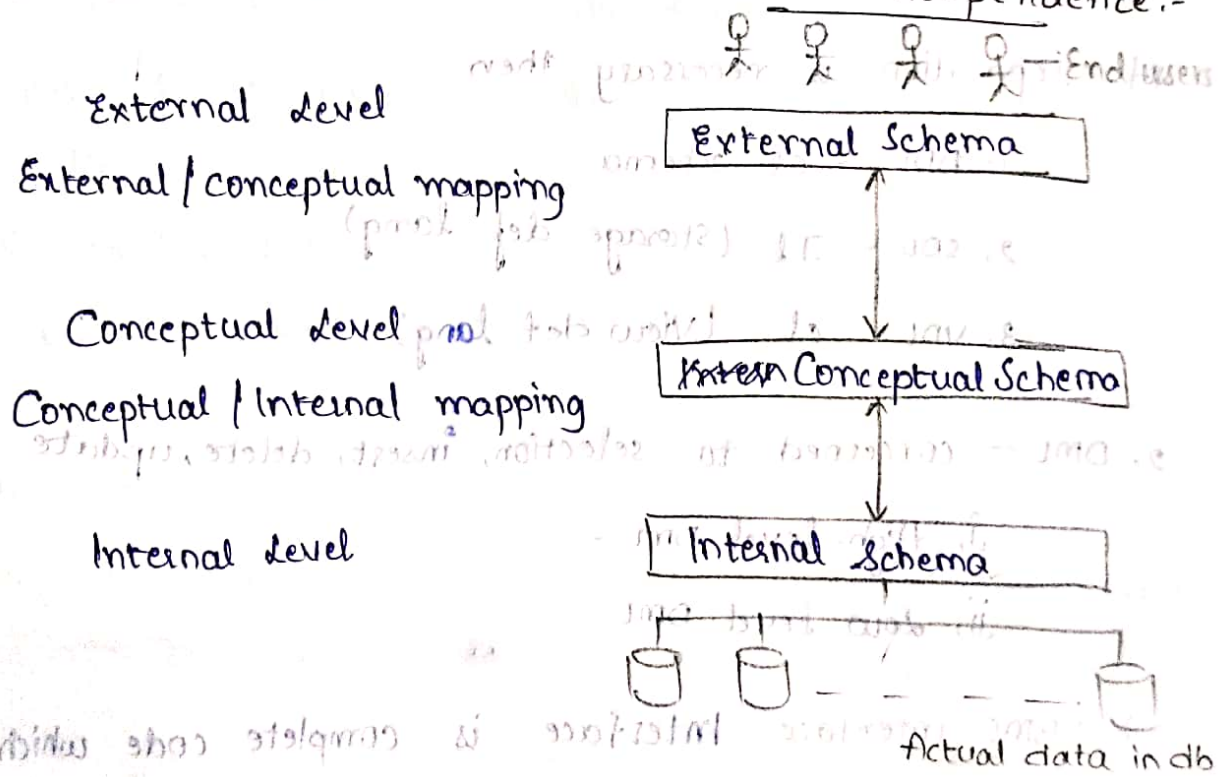
sno	sid	Grade

pre-requisite

cno	pc

- Database State: It constitutes no. of rows (instances) available in db at a particular time.

→ * Three-Schema Architecture and Data Independence:-



3- Schema architecture is used to visualize 3 important characteristics of DB system

1. DB catalog
2. Abstraction
3. Multiple User Support.

- Data Independence: Capacity to change the schema at one level without effecting the higher level in 3-schema structure

1. Logical Data Independence - Based on External & conc
2. Physical Data Independence - Capacity to change schema at internal level without effect on its higher levels called external and conceptual.

→ Database Languages:

If clear separation among conceptual & external level & il is not necessary then

1. DDL - is used to define structure. for cl, il, el

If separation is necessary then

1. DDL - cl schema
2. SDL - Il (Storage def lang)
3. VDL - el (View def lang)

2. DML -- concerned to selection, insert, delete, update

i. High level DML -
ii. Low level DML

→ DBMS Interface: Interface is complete code which asks input from user at runtime and submits to software. It is a communication path between user and dbms system. User can view information from system.

1. Menu-based interfaces: It is for web clients or browsers. List of all possible operations are displayed on menu. Used for developing web pages.

2. Form based interfaces: Thise will allow effectively to collect data and submit to dbms system. for business applications.

3. Graphical User Interfaces: Based on displayed data used performs drag & drop activities. It is mixture of menu & form based interface

4. Natural Language Interface: It will receive information in natural language and then that will convert into dbms understandable language. then dbms supplies data to user.

5. Speech based Input & Output Interface: The input will be taken in form of speech and output also will be in speech.

6. Interfaces for Parametric Endusers: Parametric end users are the users who access only a portion of db and the interfaces are developed for those users.

→ Classification of Database Management System:

1. Based on Datamodel - implementation Model

Legacy datamodel:
- (a) Relation datamodel – RDBMS, OODBMS, OORDBMS
- (b) Network datamodel – IMAGE (HP), IDMS (computer associates), DMS 1100 (Unisys)
- (c) Hierarchial datamodel - Ims (IBM), System 2k (SAS Inc)

(d) XML model - data organised in XML files

2. Based on no. of users:

(a) Single user DBMS - for single specific purpose. Only one person is allowed to access data.

(b) Multi user DBMS - Many users use dbms concurrently at a time.

3. Based on no. of Sites over database is distributed

(a) Centralized DBMS -

(b) Distributed DBMS

(a) - Data is maintained centrally at one server. Performance of query is degraded when accessed remotely.

(b) - Data is replicated & distributed among multiple sites.

i, Homogenous DDBMS

ii, Heterogenous DDBMS - different sites are different in distributed management system. i.e uses various db softwares

4. Based on cost:

(a) Free ware DBMS - mysql, postgradeSQL

(b) Properitary version DBMS -

5. Based on significance of users

(a) General purpose DBMS

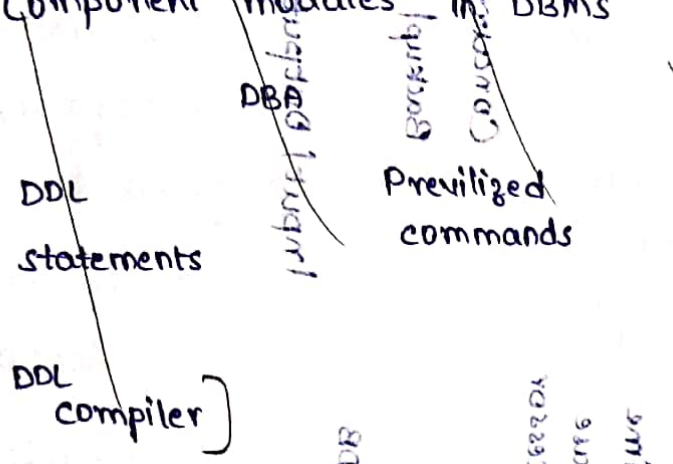
(b) Special purpose DBMS - Only specialized end users can access

6. Based on types of access paths: Inverting paths dbms.

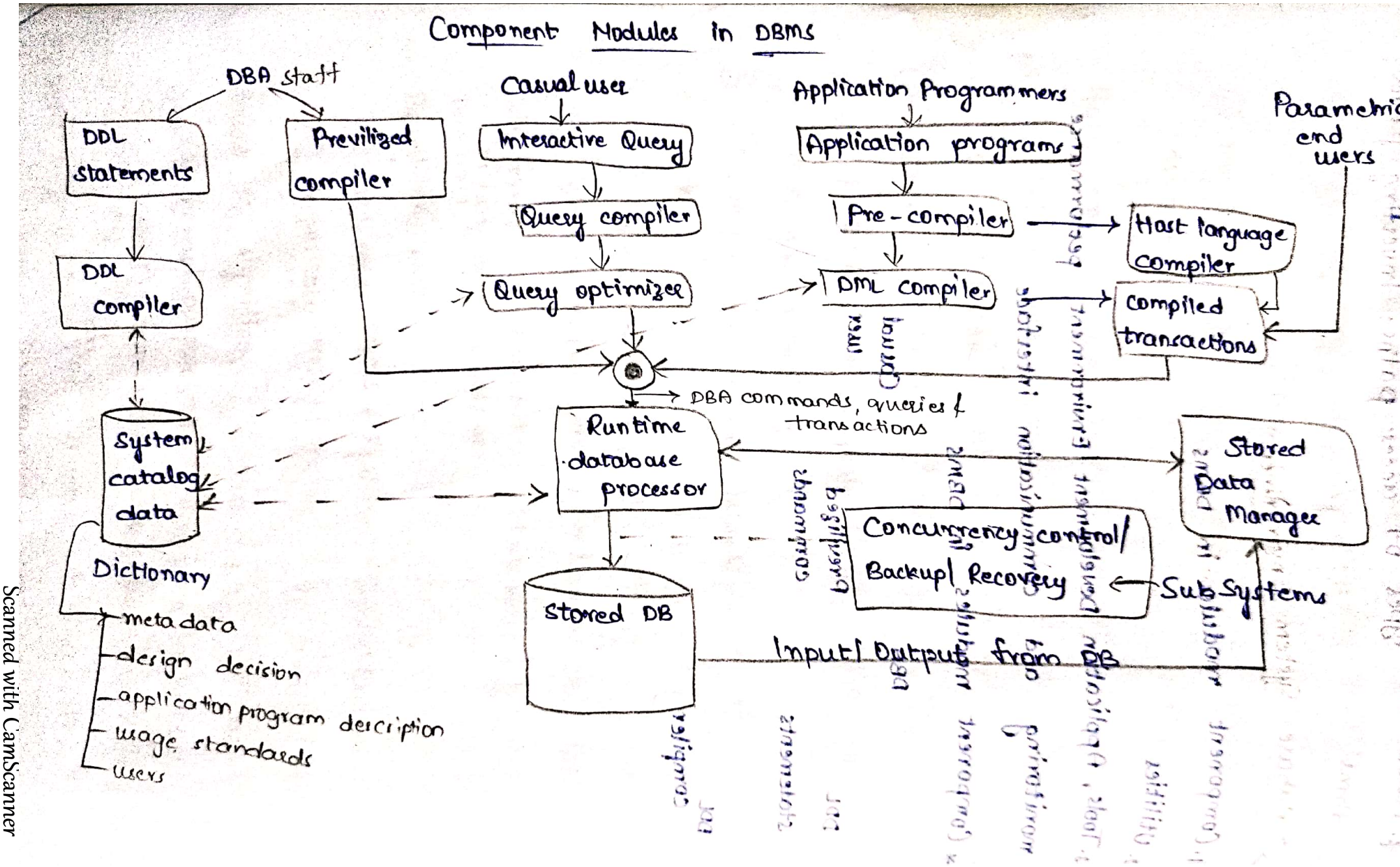
→ Database System Environment:

1. Component modules in DBMs
2. Utilities
3. Tools, Application Development Environment, performance monitoring and communication interface

* Component modules in DBMS



Component Modules in DBMS



* Data Dictionary: One of the component module in dbms which is used to store metadata, design decision, application program description, usage standards, user info.

* Utilities: It is also application programs which helps dbms software to perform additional functionalities.

Loading utilities: Will take input data from files / data on manual papers and restructured to be stored in db

Backup utilities : To maintain backup data they run in background.

DB storage reorganization utilities: These will take data format in existing db and convert into new format. It also changes access strategy

* Performance (maintaining) monitoring & communication system are used to measure performance of system like speed, response, throughput etc

(* Tools : DD)*

* Application Development Environment

Ex: Power Builder (Sysbase)

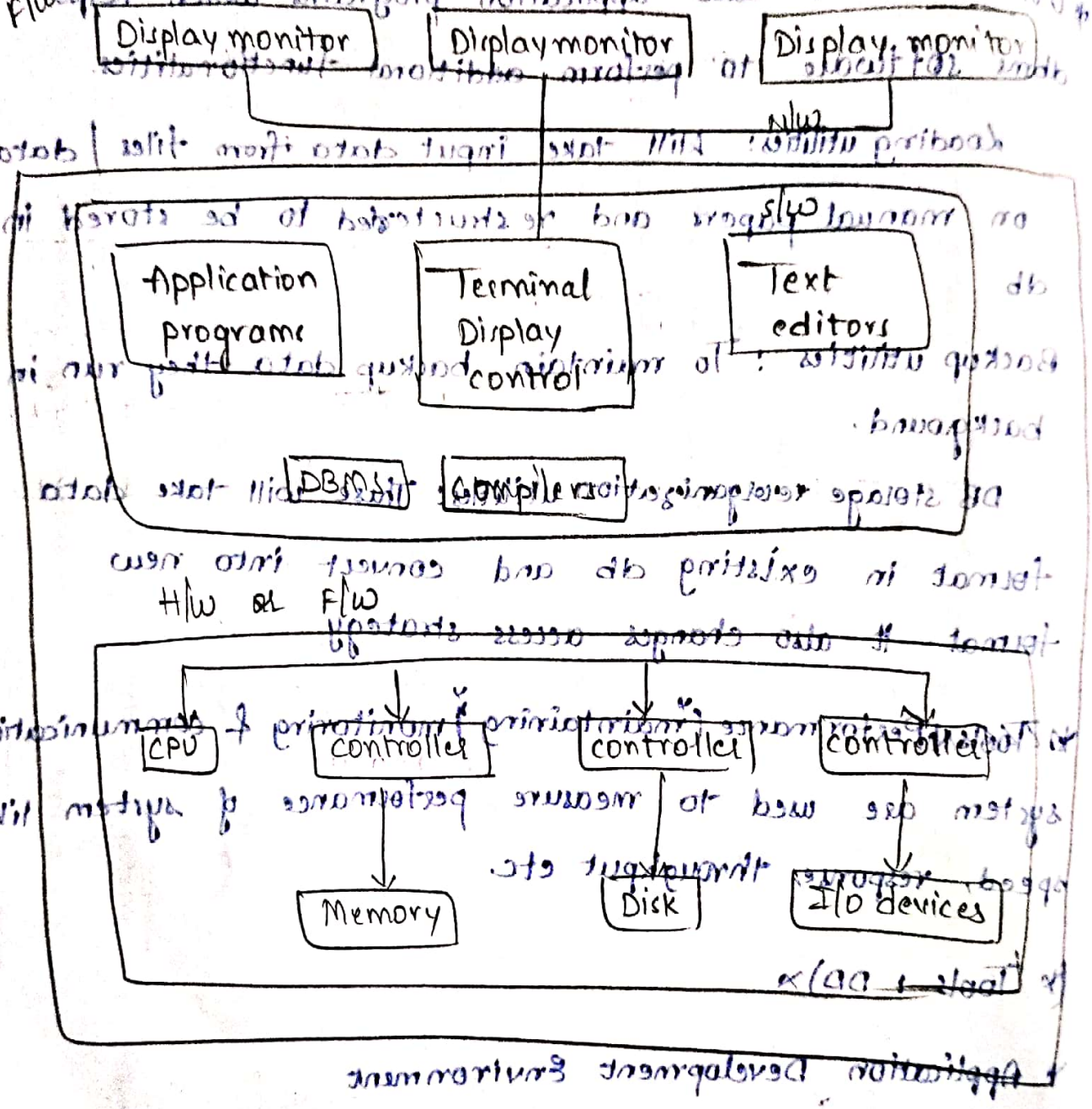
Jbuilder (Bauland)

* Communication Interface : LAN, WAN, INTRANET, INTERNET

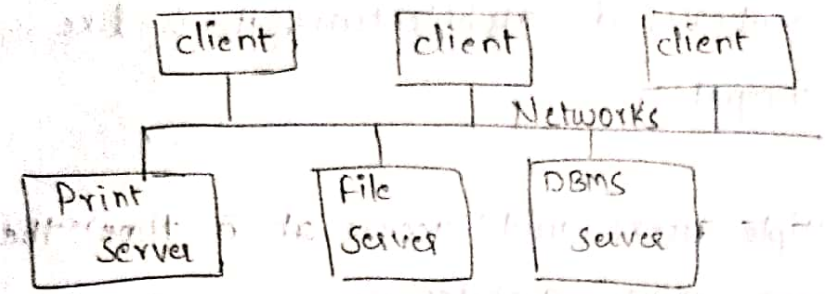
Centralized and Client/Server architectures for DBMs

H/w - hardware
S/w - software
Firmware

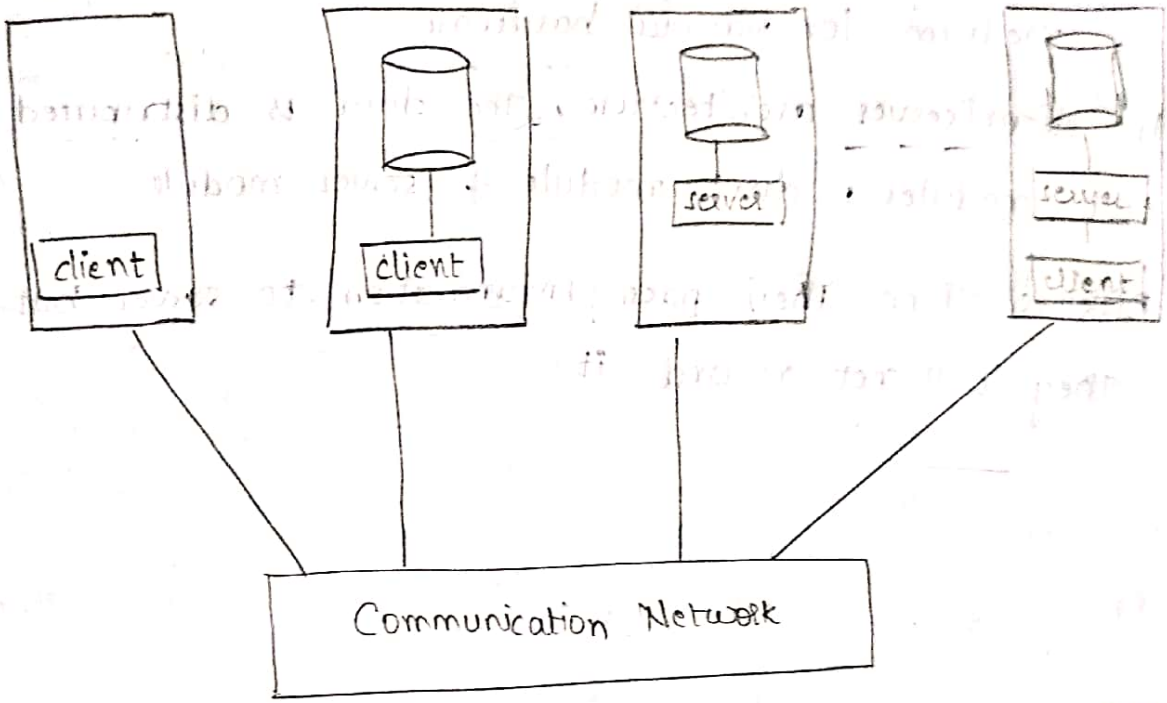
Centralized architecture for DBMS



2. Logical two-tier client server architecture:

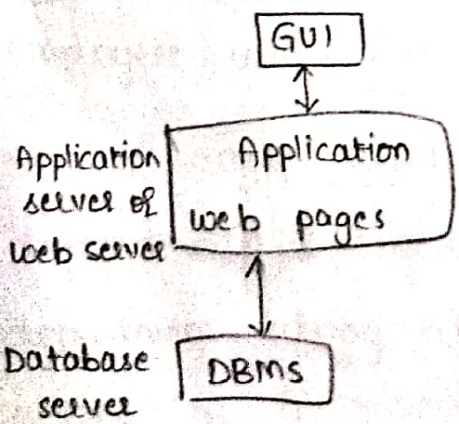


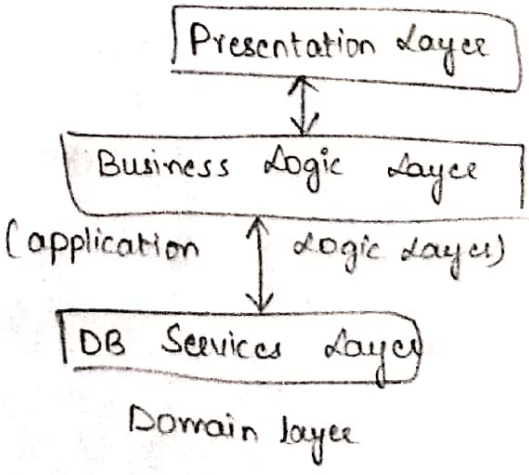
3. Physical two-tier c/s architecture



Communication Network may be LAN, MAN, INTRANET, WAN etc.

4. Logical three tier c/s architecture





Centralized : Single access or multiple access is limited then we use centralized architecture. It is like general purpose computer.

Client / server : Multiple users will access at a time then we use client / server architecture.

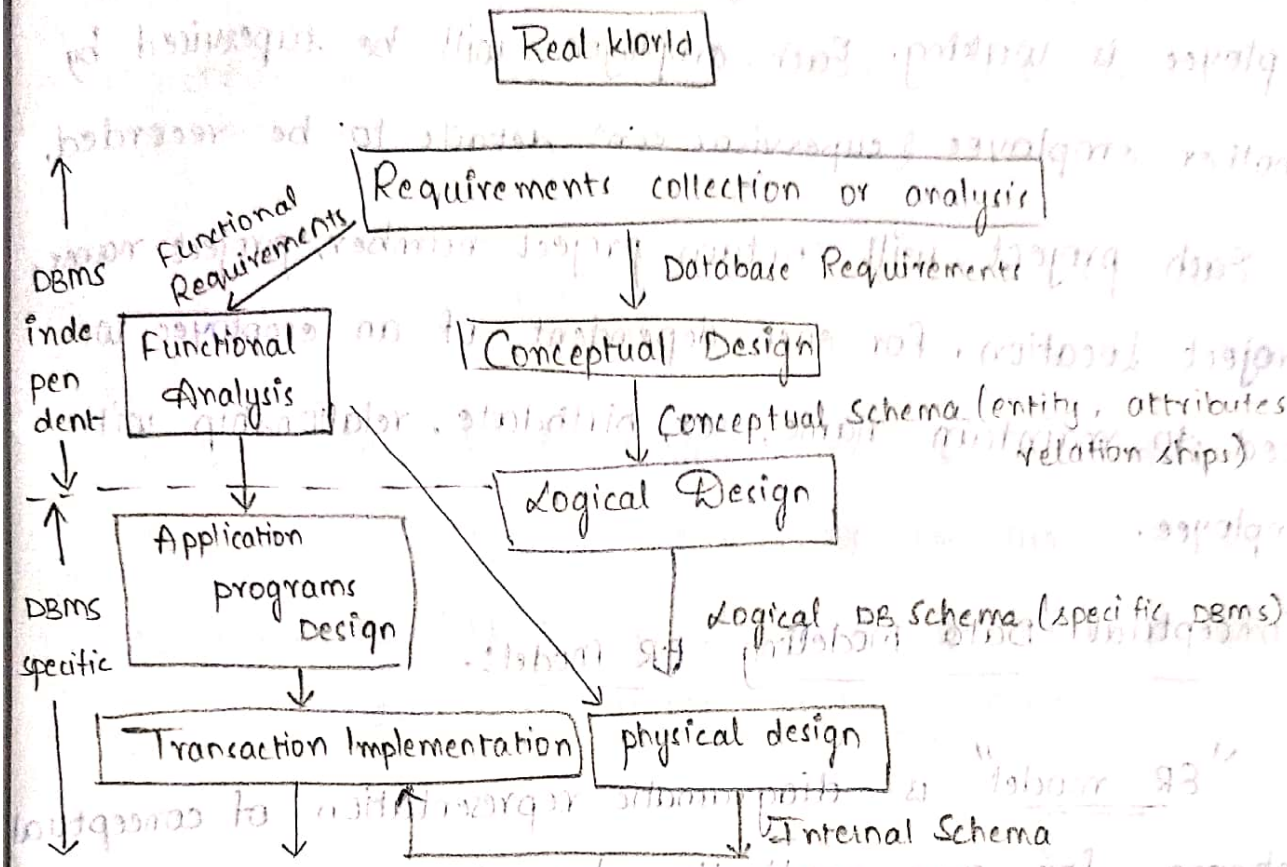
Formware: It is like software program, which acts like medium for various hardware.

In client/server architecture, the data is distributed among two modules : client module & server module.

Diskless client: They pass information to server but they will not record it.

→ Conceptual Datamodeling using entities & relationships:

Role of conceptual data model for database design -



Simple Database Application Requirements - Company

On a company, it is organised into various department, each department contains a set of employees, each department handles various projects on which one or more employees are working, while working with projects employees will give dependent details for insurance purposes.

For each department we need to record deptno, deptname, deptlocation and an employee who manages that department along with the day on which employee started acting as a manager.

For each employee we need to record social security number (SSN), name, birthdate, address, sex/gender and

salary. Each employee is allowed to work for a particular dept and he is allowed to work on 1 or more projects may not be controlled by same dept to which an employee is working. Each employee will be supervised by another employee (supervisor ssn) details to be recorded.

Each project will contain project number, project name, project location. For each dependent of an employee we need to maintain name, sex, birthdate, relationship with employee.

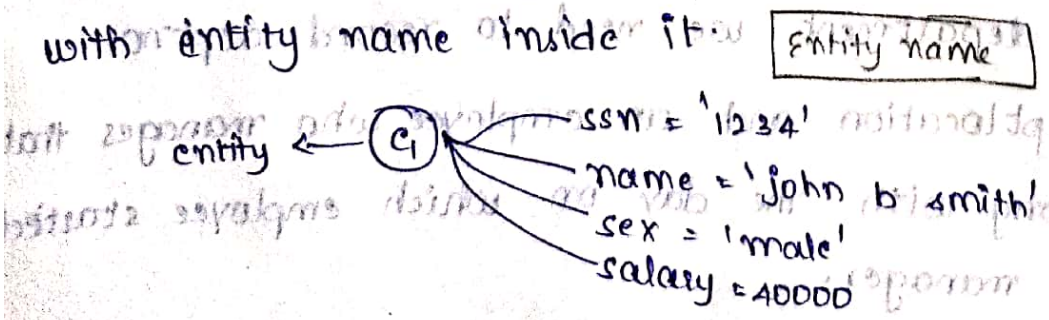
Conceptual Data Modeling ER model:-

"ER model" is diagramatic representation of conceptual schema for any application by using set of entities, attributes & relationships.

Entity :- It is an object (thing) person with physical or conceptual existence for which we need to record data

Attribute:- These are data elements / properties which are used to describe further an entity structure.

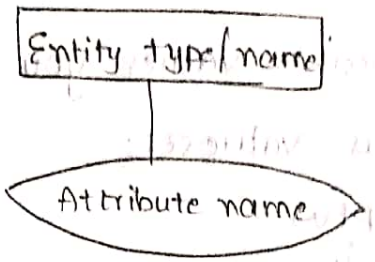
In E-R model entity is represented with rectangle along with entity name inside it.



Entity which

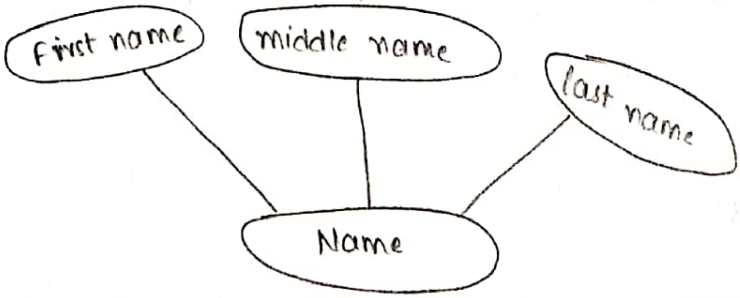
Collection of entities with common structures / properties are grouped together as 'ENTITY TYPE'.

In ER model attributes are represented by oval and is attatched to respective entity with a solid line.



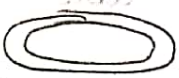
1. Simple Attribute - which cannot be divided into sub parts further eg: DOB, SSN, salary

2. Composite Attribute - which can be divided into sub parts further eg: name, address



3. Single valued Attribute : which holds a single value for any row - eg: age

4. Multi valued Attribute : which we can supply more than one value as value. eg: qualification, dept location



5. Stored Attribute : which supplies data for another attribute. eg: DOB, joining data

6. derived Attribute : An attribute which is derived from another attribute. eg: age, experience

7. Null valued Attribute : The value for that attribute is not applicable at the time entering data
eg: comission

8. key valued Attribute: Based on an attribute value if we are able to extract one individual row then they are called as key attribute. or it is simply not null & unique in er model it is represented by (Attribute name)

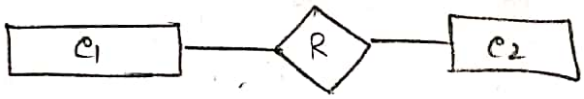
Value set : Set of possible values you can give to that attribute is known as valueset.

mathematically, $V: E \rightarrow P(V)$

Valueset V from Attribute A in ER model

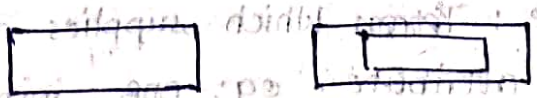
Relation Ship Type: It is an association between entities belonging to various entity type.

In er model relation ship is represented by.



Entity types:

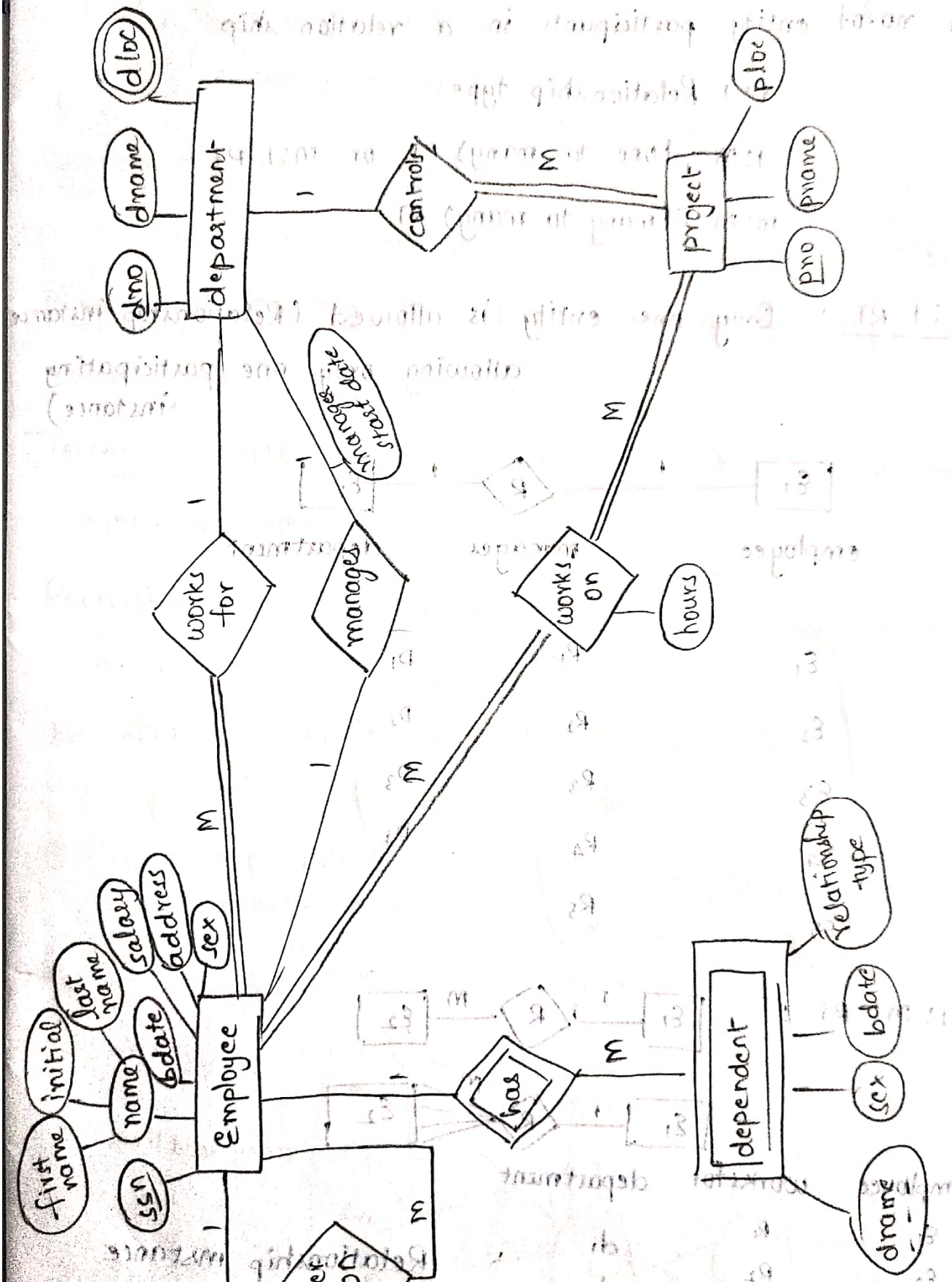
1. Strong ET or strong entity: based on key attribute if you are able to determine single entity
2. Weak ET or weak entity: They do not contain their own key attributes.



strong entity weak entity

A relation ship blw owner entity type and weak entity type is represented by  - Identifying relation ship

ER Diagram for company DB application



Constraints on "Relationship types":

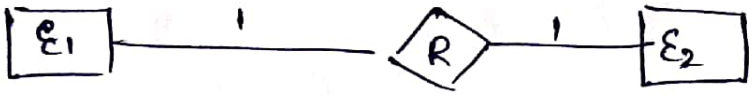
1. Cardinality constraint: These are used to allow the limit the no.of entity participants in a relation ship

1:1 Relationship type

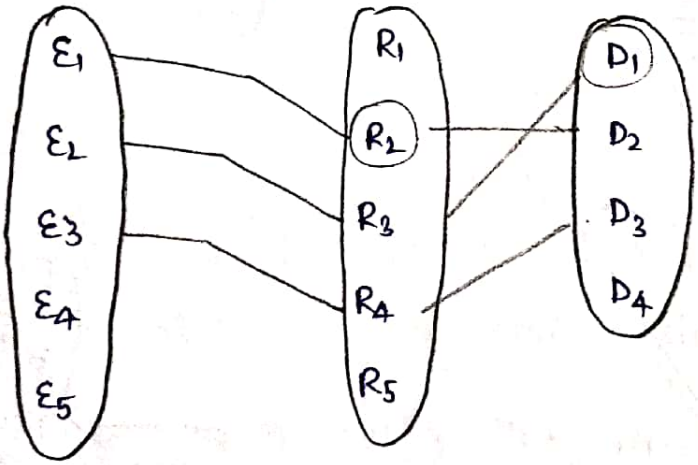
1:M (one to many) RT or M:1 RT

M:M (many to many) RT

1:1 RT :- Only one entity is allowed (Relationship instance allowing only one participating instance)



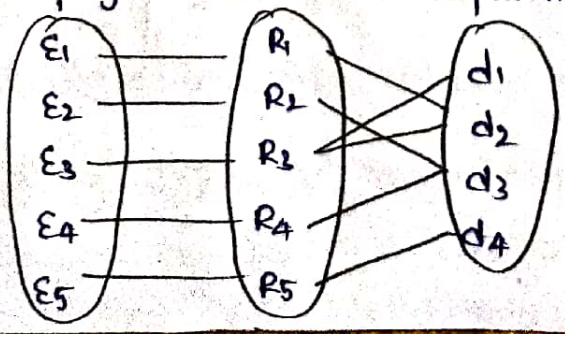
employee manages department



1:M RT :

many solid lines

Employee works for department



Relationship instance or Relation ship type

2. Participation constraints : The entity available in entities are allowed to participate as total participant

1. Total participation (═══)
2. Partial participation (——)

Degree of Relationship type :

A no. of participating entity types in relationship

Binary Relationship type : If no. of participating entity types is 'two'.

Ternary Relationship type : If no. of participating entity types is 'three'.

Recursive Relationship type : One entity type participate in more than one relationship type.

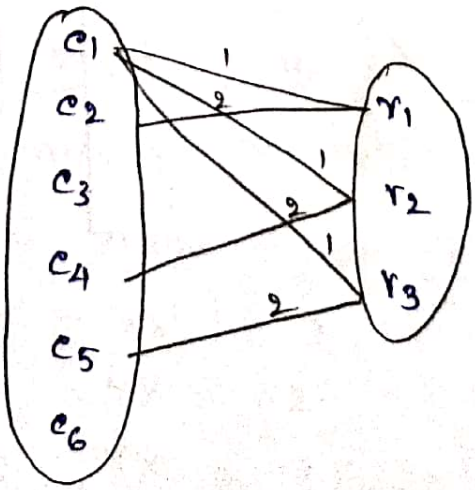
We should compulsorily specify the "role names".

supervision

supervisor - 1
supervisee - 2

$e_1 \rightarrow e_2, e_4, e_5$

employee supervision



ER model for university DB: