

✓ DBMS (Data Base Management System)
A DBMS is a system software for creating and managing databases that allows end users to create, protect, read, update and delete data in a database.

File processing system v/s. DBMS.

- File has only physical access- logical as well as physical in DBMS.
- Flexibility & Scalability in DBMS. File not easily accessible.
- Redundancy control - exists in DBMS not in file.
- Backlog recovery - possible in DBMS but not in file.
- Unauthorised access - restricted in DBMS but not in file & it cannot be controlled.
- Multiple user interface in DBMS not in file.
- Data inconsistency - in file but not occur in DBMS.
- Data isolation - in file but not in DBMS.
- Easy manege, query of transactions - in DBMS but not in file.
- Security - in DBMS not in file

DBMS follows the ACID property

- A - Accuracy
- C - Completeness
- I - Isolation
- D - Durability

- For taking access to database DBA (Data base (Database administrator)) is required.

Structure and description of data

- We have complete description of data in DBMS but not in a file.
- We can restrict or unrestricted data in DBMS at different levels.

Disadvantages of DBMS -

1. Costly as compared to file.
2. High complexity.
3. Database handling staff is required.
4. Database failures.
5. High hardware & software cost.
6. Huge size.
7. Upgradation cost.
8. Cost of data conversion.
9. Concurrency maintenance.
10. Performance monitoring.

Characteristics of DBMS -

1. Stores any kind of data.
2. Supports ACID properties.
3. Represents complex relationships between data.
4. Backup and recovery.
5. Structures & describe data.
6. Data integrity.
7. Concurrent use of database.

✓ Applications of DBMS -

1. Banking (10 marks)

- Creation of account
- Credit & debit
- loan Management
- Tracks of client deposit

2. Airlines

- Track of bookings

3. University

- Attendance detail
- Exam details
- Fees details

4. Telecommunication

- database of phone calls
- applications of phone

5. Financial sectors (Stock Market)

6. Sales and purchase details

7. Hospital

8. Hotel

9. Government sectors.

✓ Purpose of DBMS -

→ (explain in detail each)

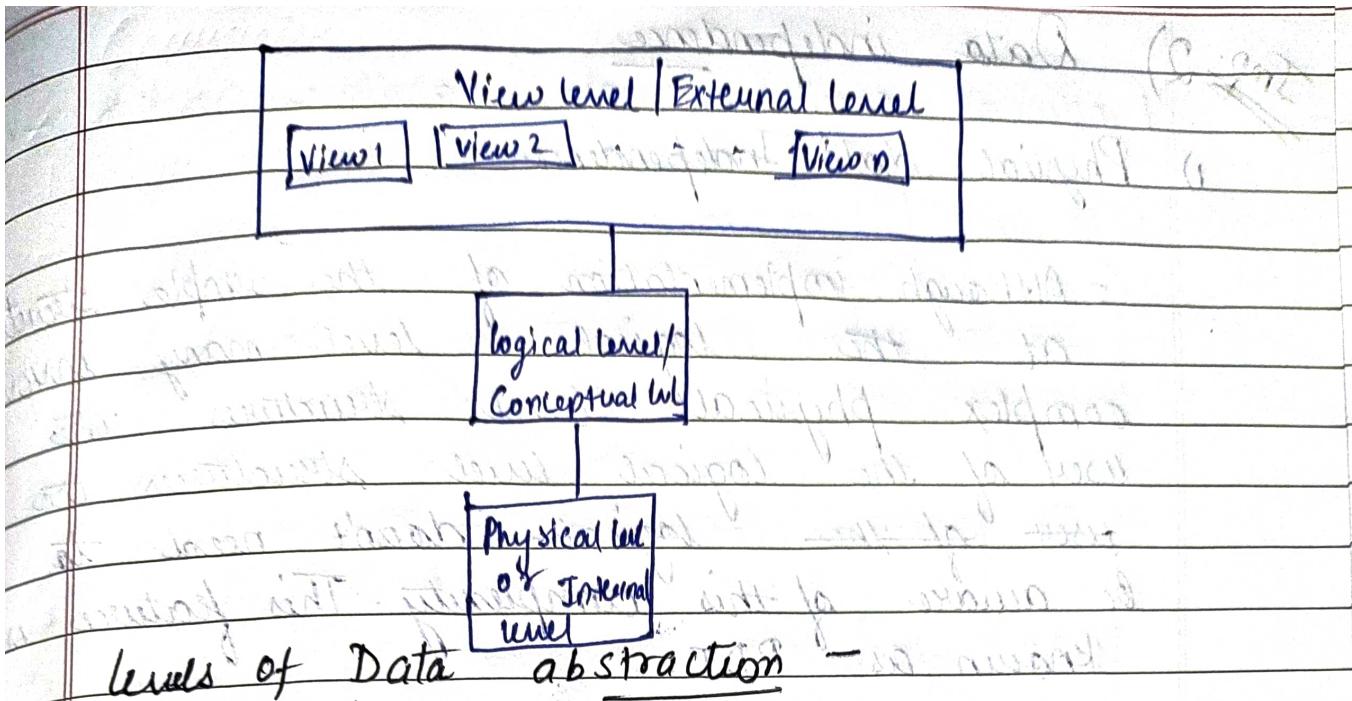
1. Data redundancy and inconsistency reduction.
2. Due to difficulty in accessing of data.
3. Due to problem of data isolation.
4. Data integrity problem.
5. Atomicity problems.
6. Security problems.

Views of data — Views of Data —

1) Data abstraction

→ The need for efficiency has led designers to use complex data structures for the representation of data in a database.

→ Developers hide the complexity from the users through several levels of extraction to simplify user interaction with the system.



1) Physical level

- This is the lowest level of abstraction that describes how the data are actually stored.

2) logical level

- It describes what data are stored in the database and what relationships exists among those data.

3) View level

- It describes entire database. The view level of abstraction exists to simplify interaction with the system.

Ques 2) Data independence

1) Physical data Independence

- Although implementation of the simple structures at the logical level many involves complex physical structures the user of the logical level structures the user of the logical level doesn't need to be aware of this complexity. This feature is known as PDI.

2) Logical data Independence

- Ability to make changes in structure of middle level of DBMS without affecting the highest level schema or application program i.e. modification in logical level should not result in any changes in the view level. This feature is known as LDI.

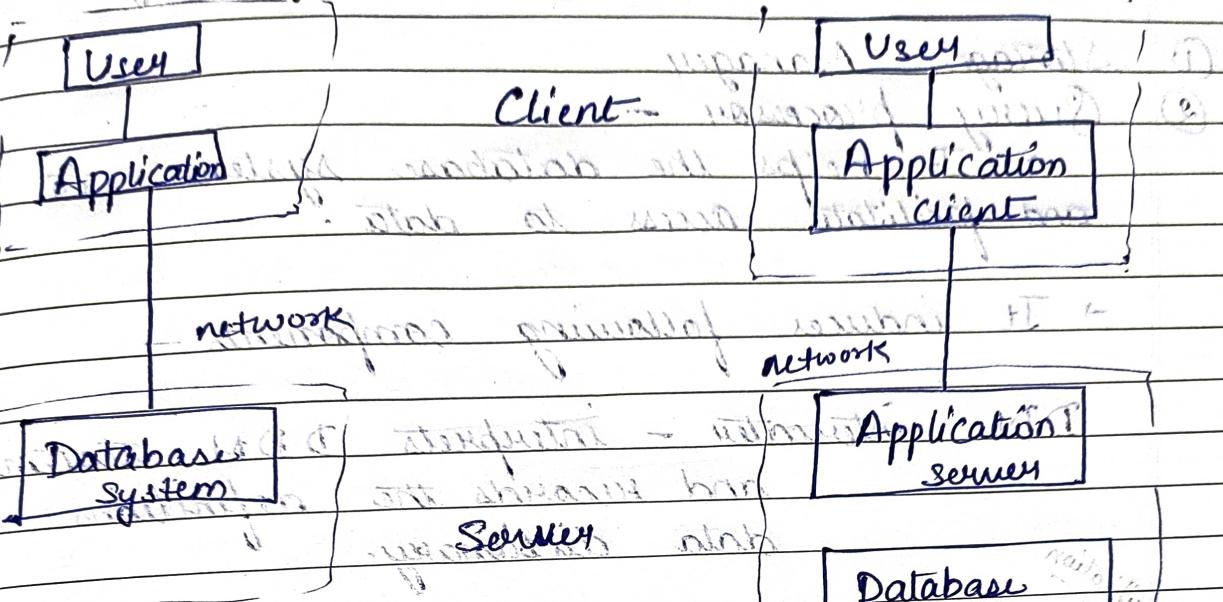
3) Instance & Schema

- The collection of information in the database at a particular moment is called instance of database.
- The overall design of the database is called database schema.

Instances and Schemas

- **Schema** – the logical structure of the database
 - e.g., the database consists of information about a set of customers and accounts and the relationship between them)
 - **Physical schema**: database design at the physical level. Describes how the data is stored in blocks of storage.
 - **Logical schema**: database design at the logical level. programmers and database administrators work at this level. At this level data can be described as certain types of data records that get stored in data structures.
 - **View schema** : Design of database at view level . This generally describes end user interaction with database systems.
- **Instance** – the actual content of the database at a particular point in time

1. Database System Architectures



2-tier Arch. statement - individual

and to individual query

3-tier Arch.

→ Database applications are partitioned into 2 or 3 parts.

→ In 2-tier architecture, the application resides at the client machine where it now invokes database system functionalities at query language statement.

- In 3-tier arch., the client machine acts as a front-end and does not contain any direct database calls, instead the client end communicates with application server through forms interface.

✓ Functional components of database system :-

① Storage Manager

② Query processor -

→ It helps the database systems to simplify and facilitate access to data.

→ It induces following components -

DDL interpreter - interprets DDL statements and records the definitions in data dictionary.

Classification
③ DML compiler - translates DML statements in a query language into evaluation plan consisting of low level instruction that query evaluation engine understands. It also performs query optimization.

→ Transaction manager

↳ Transaction manager maintains consistency of data in database by performing various operations like committing, aborting, serializing, etc.

✓ Functional components of a DBMS:

User program. \rightarrow application

Sequential indexed random list

logical I/O

Basic file system structure

Device drivers (disk, tape, etc.)

Controllers

Actual Device

Fig - File system layered architecture

File manager

- Manages the allocation space on disk

storage and the data structure used to represent information stored on other media

Buffer manager

- It transfers blocks between disks (other devices) and main memory (DMA).

Query Parser

Translates statements in query languages into low level language.

- Authorisation & Integrity Manager
- Recovery Manager
- Concurrency Controller

(i) User

entitäten mitte] die sind

(ii) Data and Database

untertopics

(iii) DBMS

ausführung

(iv) Database Applications

Database

- 1) Database Administrator - responsible for maintaining and monitoring the database system
 - responsible for authorizing access to database
 - coordinating and monitoring its use
 - Acquiring software and hardware resources as needed and maintaining IT equipment
 - managing data and its security

2) Database designers

- Identifies data to be stored in the database
- Chooses appropriate structure to represent and store data.
- Communicates with users to understand their requirements.
- Comes up with a design that meets user's requirement.

3) End users

- (i) Casual End Users
- (ii) Naive / Parametric end users
- (iii) Sophisticated end users
- (iv) Standalone users.

✓ Database languages.

→ It can be categorised as - ~~DB~~

1) DDL (Data Definition language)

 DML (Data Manipulation language)

 TCL (Transaction control language)

 DCL (Data control language)

DDL (Data definition language).

- Used to define database patterns ~~to~~ ~~#~~ structures.

→ Create - used to create object in database

ALTER - alters the pattern of "

DROP - helps in ~~in~~ deleting objects.

TRUNCATE - erase all records from table.

COMMENT - adding comment to data dictionary

RENAME - renaming an object.

DML

- used to manage data within the patterns of objects.

SELECT - used in holding data from database

INSERT - help in inserting data into a table

UPDATE - used in updating the data

DELETE - deletes a particular record.

MERGE - performs insert or update operation

~~CALL UPDATE~~ - this calls sub

EXPLAIN PLAN - has a capacity of explaining data

LOCK TABLE - controls concurrency

DCL

GRANT - permits particular users to perform particular task

REVOKE - ~~remove~~ it blocks granted permissions

TCL

- used to save work

SAVE POINT - helps in identifying a point in the transaction, can be rolled back to the identified point.

ROLL-BACK - feature of restoring the database to the point since the last commit

SET TRANSACTION - parameters for changing
settings like roll back point.

e.g -

SELECT * FROM table-name;
INSERT INTO table-name values;

Importance of data models.

- Describe structure of data in a database
- Support development of information systems.
- Guideline for development.
- Referred as data structure

Advantages

- Prevents system from future risks & failures.
- Reduces cost of project by proper planning and cost.
- Data repetition and data type compatibility can be checked.
- Improves quality of system.

Classification of data models.

Data models can be classified into following heads —

Hierarchical model & Network model

Network

Hierarchical model

- In this, a hierarchical relationship is formed by collection of relations and forms a three layered structure.

- It can be defined in the form of parent child relationship

Advantages -

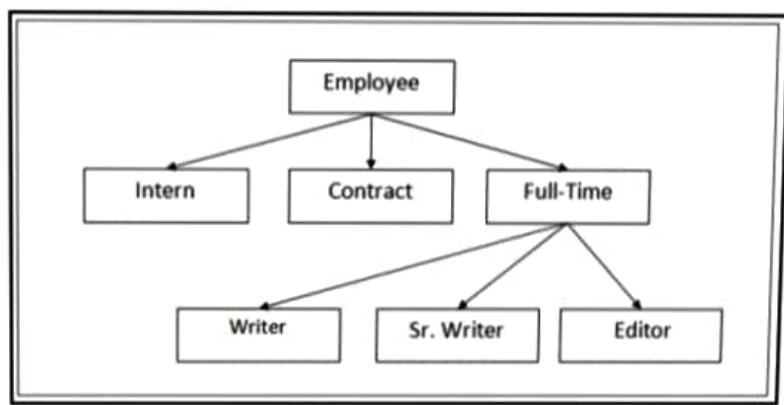
- ① Design is simple
- ② Provides data integrity
- ③ Data sharing is feasible
- ④ Even for large volumes of data, the model works perfectly.

Data is more manageable.
Simple creation & updation process.

Disadvantages -

- Implementation is complex
- Has to deal with anomalies
- Maintenance is difficult
- Complex application programming.

Example: The hierarchy shows an Employee can be an Intern, on Contract or Full-Time. Sub-levels show that Full-Time Employee can be hired as a Writer, Senior Writer or Editor:



Network Model

- This model has graph & links.
- The relationship can be defined in the form of links and it handles many to many relationships.
- It states that a record can have more than one parent.

Advantages: easier to maintain & modify and

- ① easy to design
- ② can handle one to one, one to many & many to many relationships

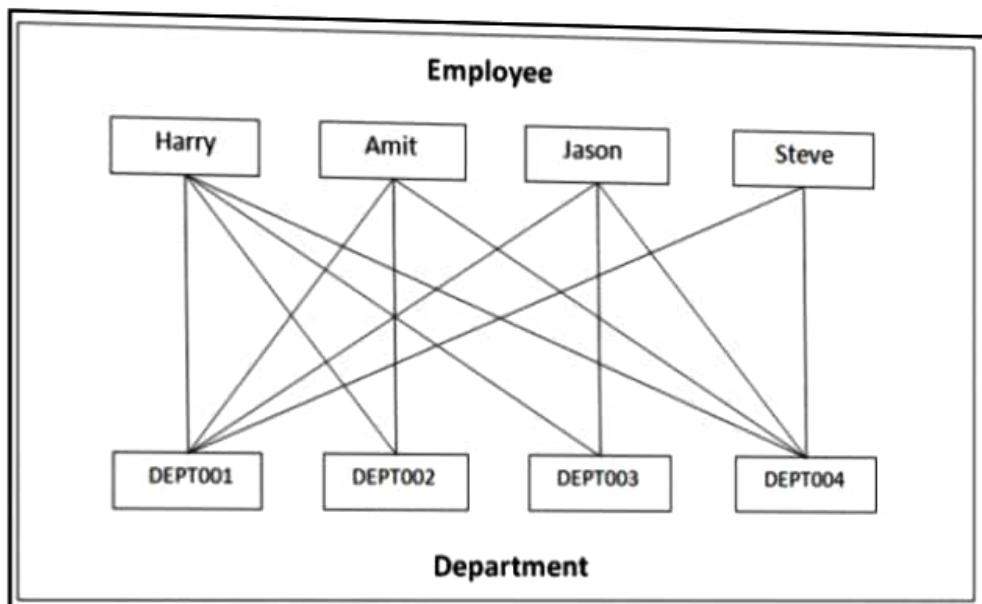
DATE / /
PAGE

- (3) It isolates the program from other details.
- (4) Based on standards and conventions.
- (5) Ease of data access.

Disadvantages:

- (1) Pointers bring complexity.
- (2) Changes in database is not easy.
- (3) lack of structural independence.
- (4) system complexity

Example



Relational model

- A relational model groups data into one or more tables.
- The tables are related to each other using common records.
- The data is represented in the form of rows and columns i.e. tables.

Advantages

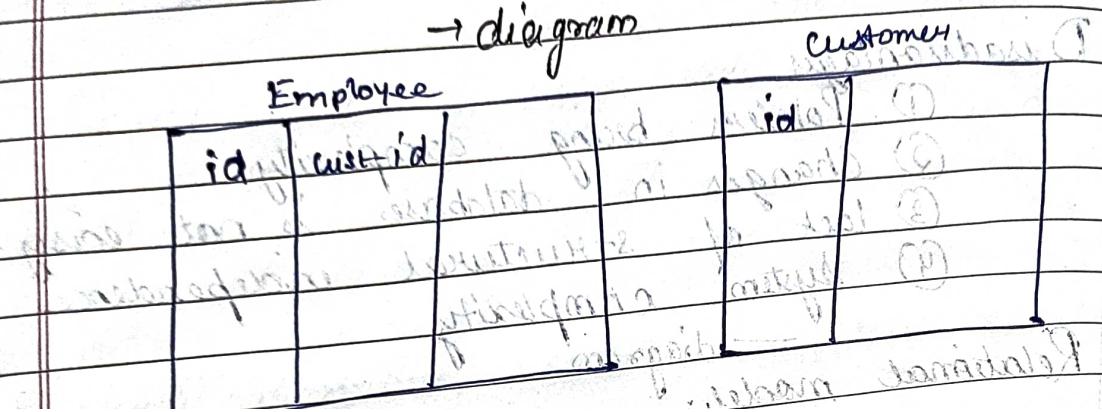
- ① This model does not have any scope related to anomalies.
- ② Data independence.
- ③ Implementation is easy.
- ④ Maintenance is easy.

Disadvantage

- ① Database when model has large volumes of data.

② Overheads of using this model, comes with the cost of using powerful hardware & devices.

→ diagram



- The data is represented in the form of rows and columns i.e. tables:

	Column1	Column2	Column3
Row1			
Row2			
Row3			
Row4			
Row5			

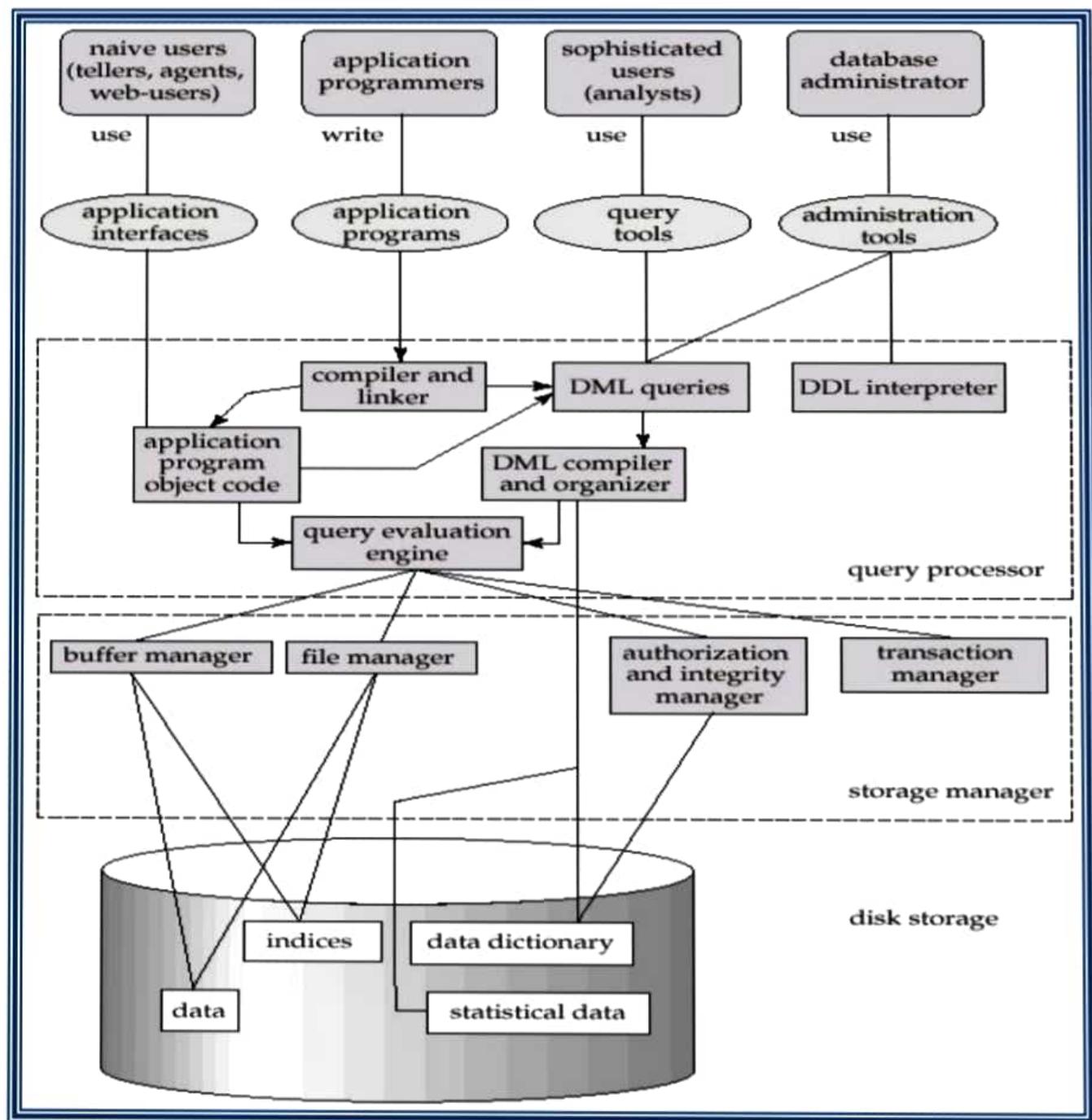
Example

Let us see an example of two relations **<Employee>** and **<Department>** linked to each other, with **DepartmentID**, which is Foreign Key of **<Employee>** table and Primary key of **<Department>** table.

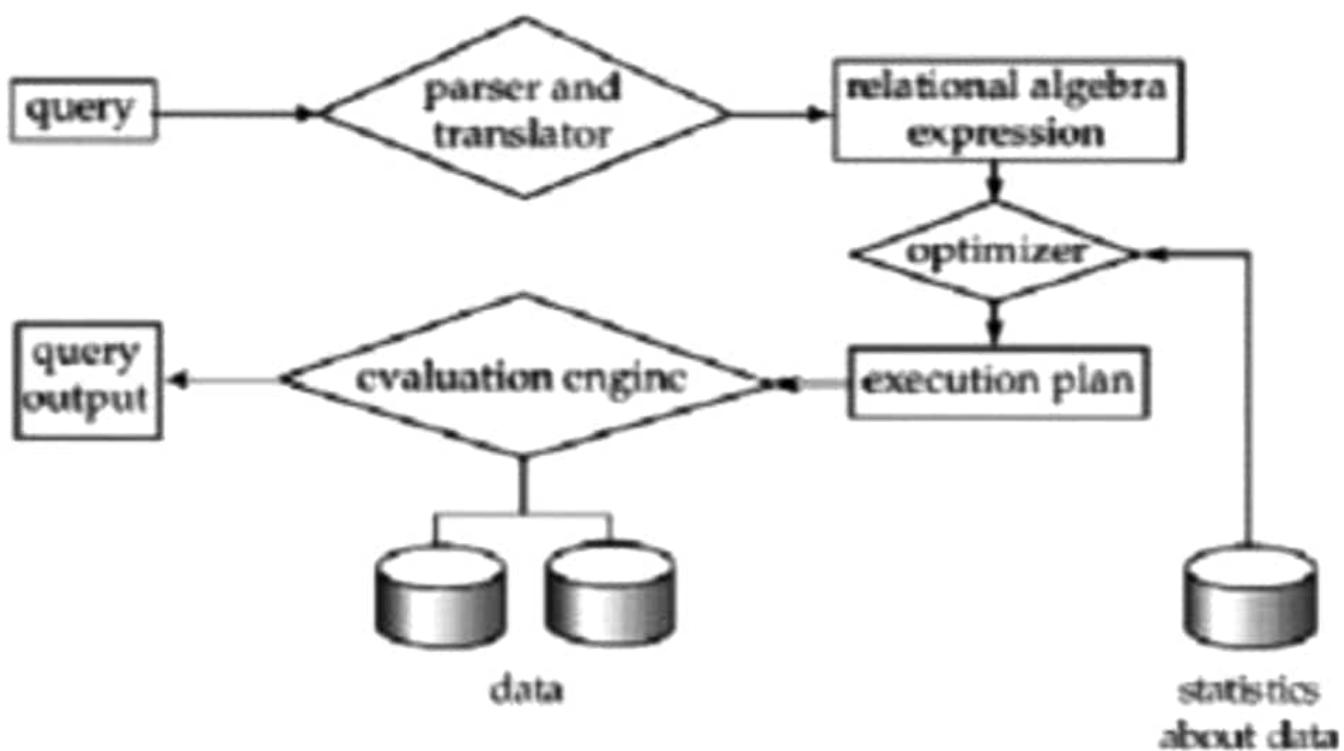
<Employee>		
EmployeeID	EmpName	DepartmentID
E09	Emily	D001
E04	Tom	D002
E11	Emma	D003

<Department>		
DepartmentID	DeptName	DeptZone
D001	Finance	North
D002	Operations	East
D003	Marketing	West

Database System Architecture



Query processor



Storage Manager

- Storage Manager is a program that provides an interface between the data stored in the database and the queries received.
- It is also known as Database Control System.
- It maintains the consistency and integrity of the database by applying the constraints and executes the DCL statements.
- It is responsible for updating, storing, deleting, and retrieving data in the database.

Lec 4 27 Jan 2021

Storage Manager

Contains the following components :

- **Authorization Manager** – It ensures role-based access control, i.e. checks whether the particular person is privileged to perform the requested operation or not.
- **Integrity Manager** – It checks the integrity constraints when the database is modified.
- **Transaction Manager** – It controls concurrent access by performing the operations in a scheduled way that it receives the transaction. Thus, it ensures that the database remains in the consistent state before and after the execution of a transaction.
- **File Manager** – It manages the file space and the data structure used to represent information in the database.
- **Buffer Manager** – It is responsible for cache memory and the transfer of data between the secondary storage and main memory.

26-27/40

Disk Storage

Contains the following components:

- **Data Files** – It stores the data.
- **Data Dictionary** – It contains the information about the structure of any database object. It is the repository of information that governs the metadata.
- **Indices** – It provides faster retrieval of data item.