Data Management And Analytics

Karuna Prasad

karunap@cdac.in

Topics covered

- RDBMS – Oracle, MySQL
- NoSQL - MongoDB
- Data Analytics – Apache Spark

- Total duration – 3/09/2020 to 25/09/2020
- Assignments

Introduction

Databases and Systems to manage them have become significant components of any present day business of any nature. It has a major impact on growing use of computers.

These databases help businesses to perform their day-to-day activities in an efficient and effective manner.

- Banking
- Travel ticket reservation
- Library catalog search

Here some program access the database.
Advances in technology have given raise to new concepts-

❑ Multimedia databases
❑ GIS
❑ Web data
❑ Data warehousing and mining

Database Applications

- ✓ Banking: All transactions
- ✓ Airlines: Reservations, Schedules
- ✓ Universities: Registration, Grades
- ✓ Sales: Customers, Products, Purchases
- ✓ Manufacturing: Production, Inventory, Orders, Supply chain
- ✓ Human resources: employee records, salaries, tax deductions
- ✓ IT professionals.
- ✓ Business management professionals
- ✓ Marketing professionals : to analyze sales data
- ✓ Human resource managers : to evaluate employees
- ✓ Operations managers : to track and improve quality

➔ Databases touch all aspects of our lives

Data: Known fact that can be recorded and that has implicit meaning.

Ex. *Name*, *Tel_no*, *city* etc.

This data can be stored in a file on a computer.

Database: Is a collection of related data.

❖ It is a collection of logically related data.

❖ A database is designed, built and populated with data for a specific purpose.

DBMS

DBMS: Is a collection of programs that enables users to create and maintain databases in a convenient and effective manner.

DBMS is a software system that facilitates the following:

1.*Defining the database:* This includes defining the structures, data types, constraints, indexes etc.

Database catalog/Data dictionary/ called as *Meta-data*

2.*Constructing the database:* This means storing data into the database structures and storing on some storage medium.

3.*Manipulating database for various applications:* This encompasses activities like – *querying* the database, *inserting* new records into the database, *updating* some data items, and *deleting* certain items from the database.

What is DBMS?
What is a Database System?

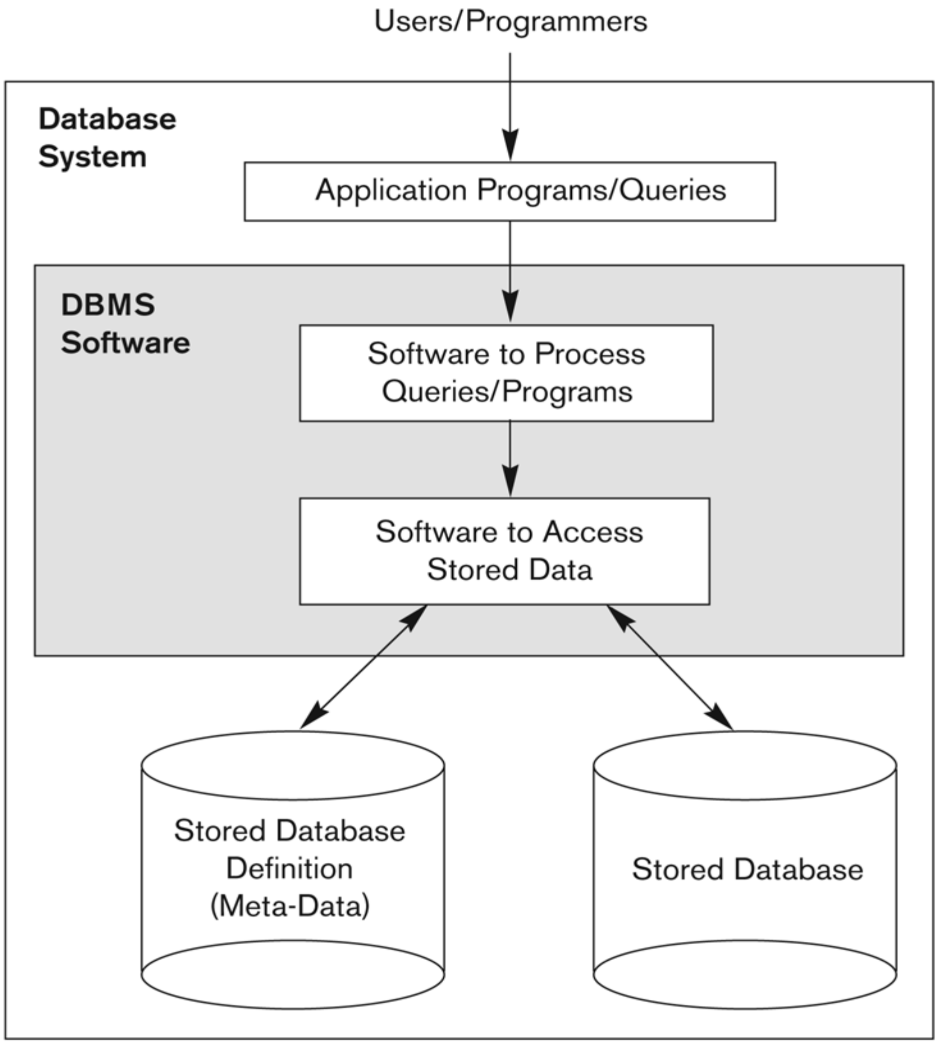

Users/Programmers
Database System
Application Programs/Queries
DBMS Software
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition (Meta-Data)
Stored Database

Figure 1.1
A simplified database system environment.

Traditional file systems for storing the data

If we take the example of savings bank enterprise, information about customers and savings accounts etc. need to be stored.

One way to keep the information on computers is to store in files provided by operating systems (OS).

Disadvantages of the above System

- ❖ Difficulty in accessing data (possible operations need to be hard-coded in programs).
- ❖ Redundancy leading to inconsistency.
- ❖ Inconsistent changes made by concurrent users.
- ❖ No recovery on crash.
- ❖ The security provided by OS in the form of password is not sufficient.
- ❖ Data Integrity is not maintained.

Advantages of using DBMS

- Data independence
- Efficient data access
- Data integrity and security
- Data Administration
- Concurrent access and Crash recovery
- Reduced application development time

Disadvantages of DBMS

1. Extra cost due to SW, HW and training.
2. Not suitable or effective for certain applications (Real-time constraints; well-defined limited operations)
3. Data manipulation not supported by Query languages.

Describing and storing data in DBMS

Data model

Is a collection of concepts that can be used to describe the structure of the database. Structure means data types, relationships, constraints etc.

DBMS allows a user to define the data to be stored in terms of a data model.

i) high-level data models (ii) low-level data models, (iii) representational or implementation data models

***High-level data models*:** Use set of concepts to describe the database, where the descriptions are close to user views. High-level data models are also known as conceptual models.
In conceptual data modeling we use concepts like – entity, attributes, relationship etc.

Low-level data models: give details about how the data is stored in a computers (storage level details).

Representational/Implementation data models:

This is in between high-level and low-level data models.

Here we represent the concepts described in conceptual model using a specific structures like, networks, objects, tables, trees etc.

Ex: Relational Model, NW Model, Hierarchical Model, Object Model, Object relational model etc.

Relational Model:

The central data description construct in this model is a relation, which can be thought of as a set of records.

Schema: Description of data in terms of a data model is called a schema. A relation schema specifies the name of the relation, fields, type etc.

Ex.   *Student (sid: string; name: string; age: integer)*

every row follows the schema of the relation.

The following are some important representational data models (DBMS Specific)

1. Hierarchical Model: The records containing data are organized as a collection of trees. Ex.: IBMs IMS (Information Management System), in late 1960s

2. ***Network Model***: Though the basic structure is a record, the relationships are captured using links.
The database can be seen as an arbitrary network of records connected by links.
Ex.: GE's Integrated Data store (IDS), in Early 1960s

3. ***Relational Model***: (early 1970s)Data & relationships are captured as tables & keys.
Ex.: Oracle, IBMs DB2, MySQL, Informix, Sybase, MS Access, Ingress, MySQL etc.
The basic storage structure is a record.

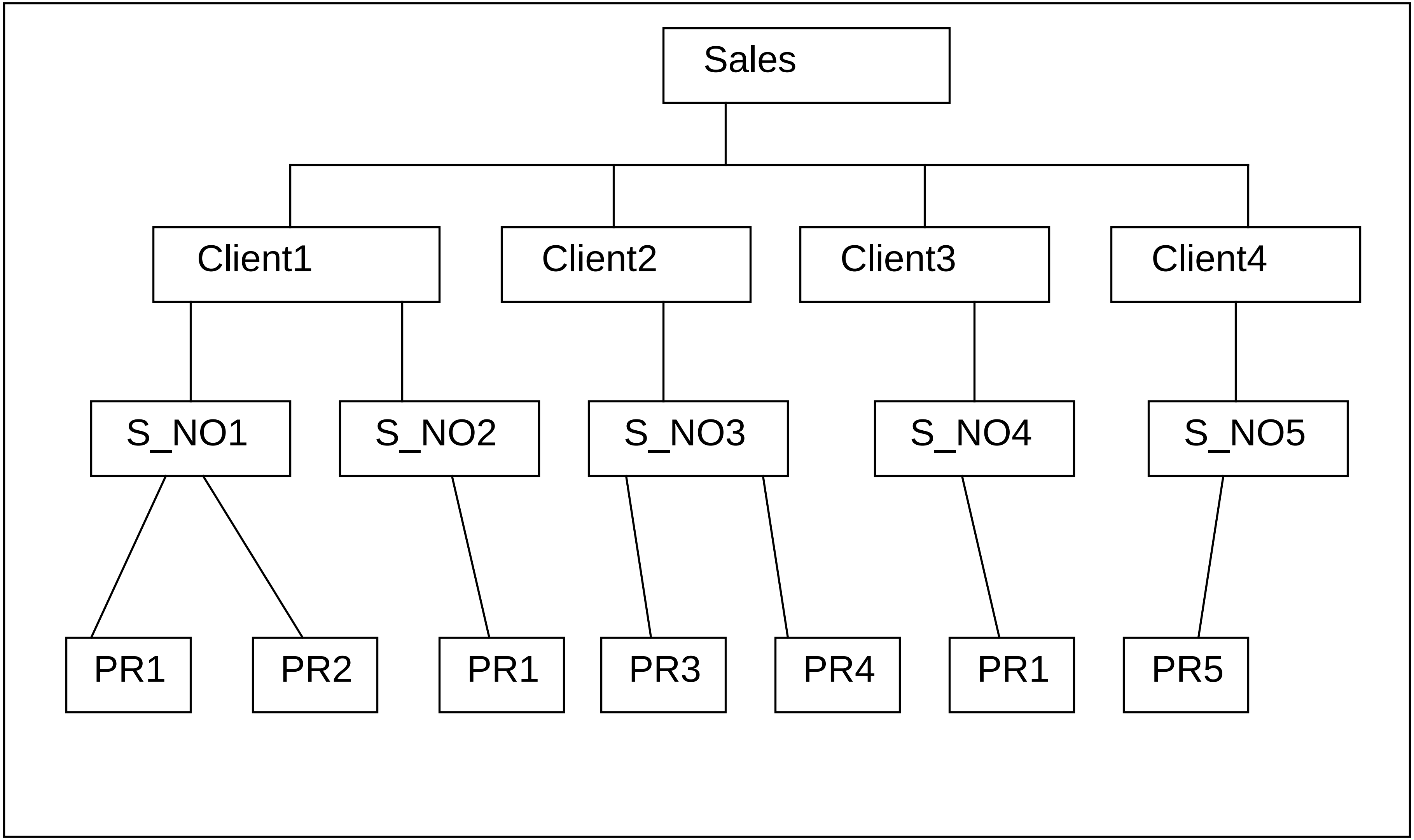
4. ***Object Data Model***: Objects created through object–oriented programs can be stored in database.
Ex.: Object Store

5. Object Relational Model: Objects can be stores in tables.
Ex.: Oracle, Informix

Hierarchical Model

- ✓ It is a single parent-child structure that is similar to an inverted tree.
- ✓ In this model data items are assigned to different levels of a hierarchy.
- ✓ Every data item (except the root node) acts as a node with exactly one parent node and zero or more child nodes.
- ✓ A Parent table can have many child tables,but a child table can have only one parent table.

Hierarchical Model



Problems with Hierarchical Model

1.There is a lot of scope for duplication of data in this model.

2.Making modifications is extremely complex.

3.Extracting information from this model as per requirement will not be easy.

4.The lower records in a tree may be incomplete in meaning without their parents.

5. It is often considered unsuitable for many applications because of its inflexible structure and lack of support for complex relationships.

Network Model

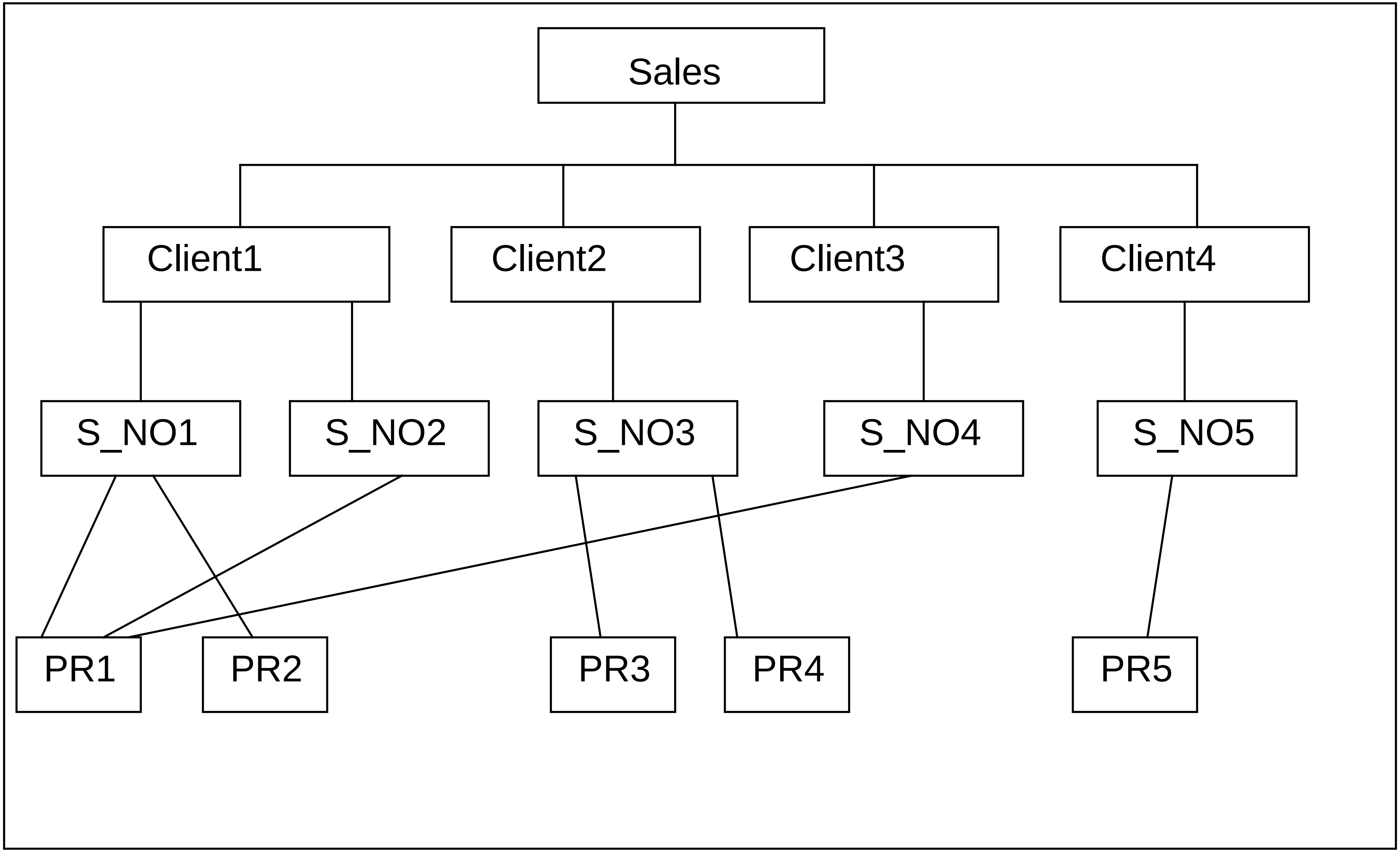
- ✓ Network model is an improvement on the Hierarchical model. Here multiple parent-child relationships are allowed

- ✓ If a child in a data relationship has more than one parent,the relationship cannot be described as a tree or hierarchical structure. Instead it is described as a network or plex structure.

- ✓ Tables are organized into a set structure that relates pairs of tables into owners and members, which supports more complex queries than are possible in the hierarchical model.

- ✓ Eliminates the duplication of data items encountered in the hierarchical model completely. Every data item appears only once as a node.

Network Model



Problems with Network Model

- ✓ The degree of complexity is extremely high. Hence errors are difficult to trace.
- ✓ Not flexible enough to make changes once data is entered.
- ✓ Searching of records becomes difficult.
- ✓ Because of rigid designs, changes in data structures result in rebuilding the database

Relational Model

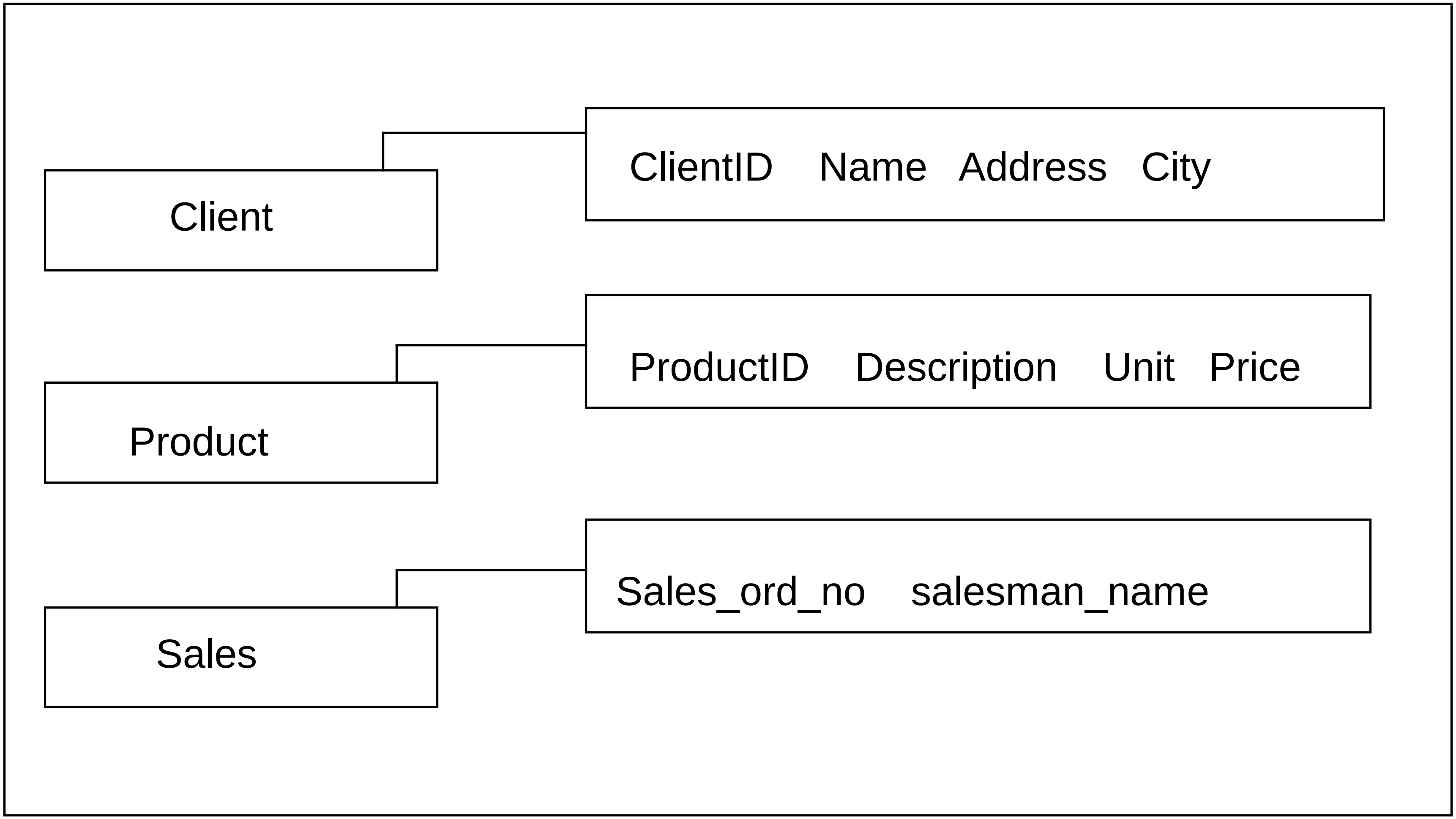
A database based on the relational model developed by E.F.Codd.

- ✓ A relational database allows the definition of data structures, storage and retrieval operations and integrity constraints.
- ✓ The data and relations between them are organised in tables.
- ✓ A table is a collection of records and each record in a table contains the same fields.

Properties of Relational Tables:

- · Values Are Atomic
- · Each Row is Unique
- · Column Values Are of the Same Kind
- · The Sequence of Columns is Insignificant
- · The Sequence of Rows is Insignificant
- · Each Column Has a Unique Name

Relational Model



Client

ClientID Name Address City

Product

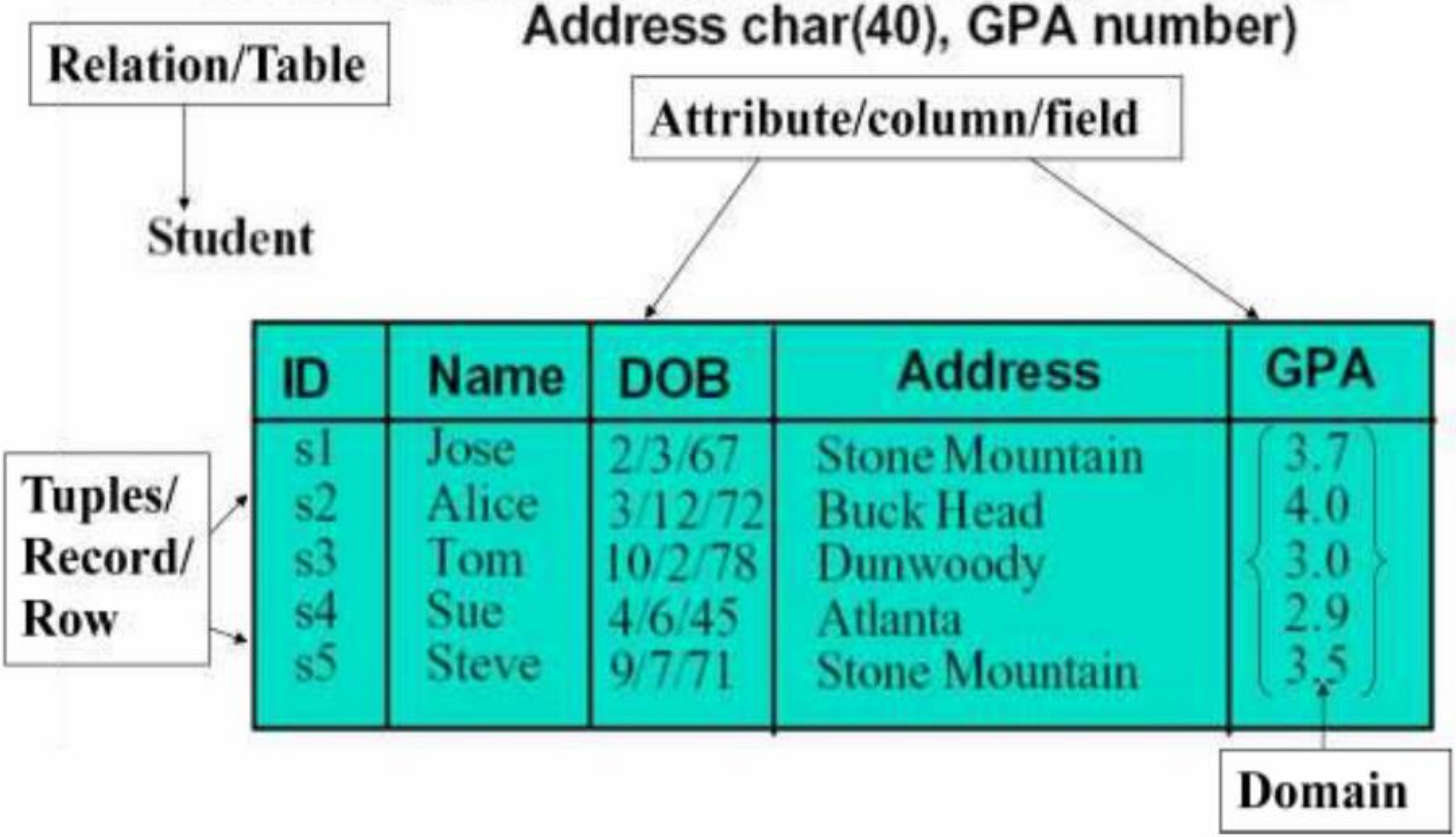
ProductID Description Unit Price

Sales

Sales_ord_no salesman_name

RDBMS terminologies

Student (ID char(30), Name char(30), DOB date
Address char(40), GPA number)



Relation/Table

Student

Attribute/column/field

ID	Name	DOB	Address	GPA
s1	Jose	2/3/67	Stone Mountain	3.7
s2	Alice	3/12/72	Buck Head	4.0
s3	Tom	10/2/78	Dunwoody	3.0
s4	Sue	4/6/45	Atlanta	2.9
s5	Steve	9/7/71	Stone Mountain	3.5

Tuples/
Record/
Row

Domain

Relational Model

Sales

SalesNo	Name	City	Pin code	State
S001	Aman	Mumbai	400054	Maharastra
S002	Omkar	Madras	560008	Tamil Nadu
S003	Raj	Bangalore	789654	Karnataka
S004	Ashish	Mangalore	453211	Karnataka

Product

ProNo	Description	Profit	Unit	Qty
P001	Screw	5	Piece	200
P002	Nuts	3	Grams	150
P003	Bolts	3.5	Grams	180
P004	Handles	2.5	Pieces	100

Relational Model

Client

Client No	Name	City	Pin code	State
C001	Ivan Bayross	Mumbai	400054	Maharastra
C002	Smith	Madras	560008	Tamil Nadu
C003	James	Bangalore	789654	Karnataka
C0004	King	Mangalore	453211	Karnataka

Database Schema

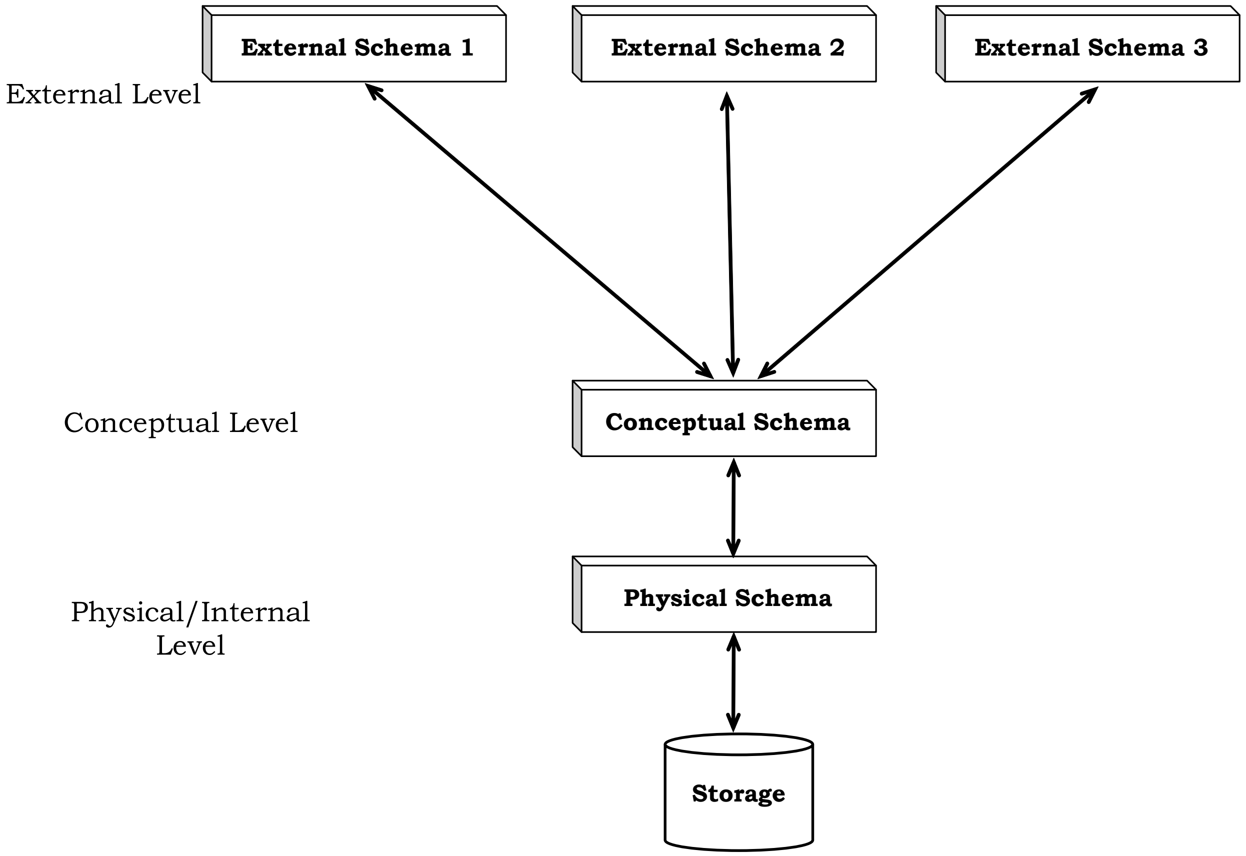
Database Schema: Description of a database is called as *database Schema*

Three-Schema Architecture

A database can be described using three different levels of abstractions. Description at each level can be defined by a schema. For each abstraction we focus on one of the specific issues such as user views, concepts, storage etc.

1. *External schema*: Used to describe the database at external level. Also described in terms of the data model of that DBMS. This allows data access to be customized at the level of individual users/groups/applications. Any external schema has one or more views and relations from the conceptual schema. This schema design is guided by end user requirements.
2. *Conceptual schema* (logical schema) Describes the stored data in terms of the data model specific to that DBMS. In RDBMS conceptual schema describes all relations that are stored in the database. Arriving at good choice of relations, fields and constraints is known as conceptual database design.
3. *Physical schema*: Describes the physical storage strategy for the database.

Three Schema Architecture



External Schema 1

External Schema 2

External Schema 3

External Level

Conceptual Level

Conceptual Schema

Physical/Internal Level

Physical Schema

Storage

Three schema architecture of DBMS

Data Independence

Data Independence:
The three-level architecture which is the result of the three-level abstraction on database, leads to data independence.

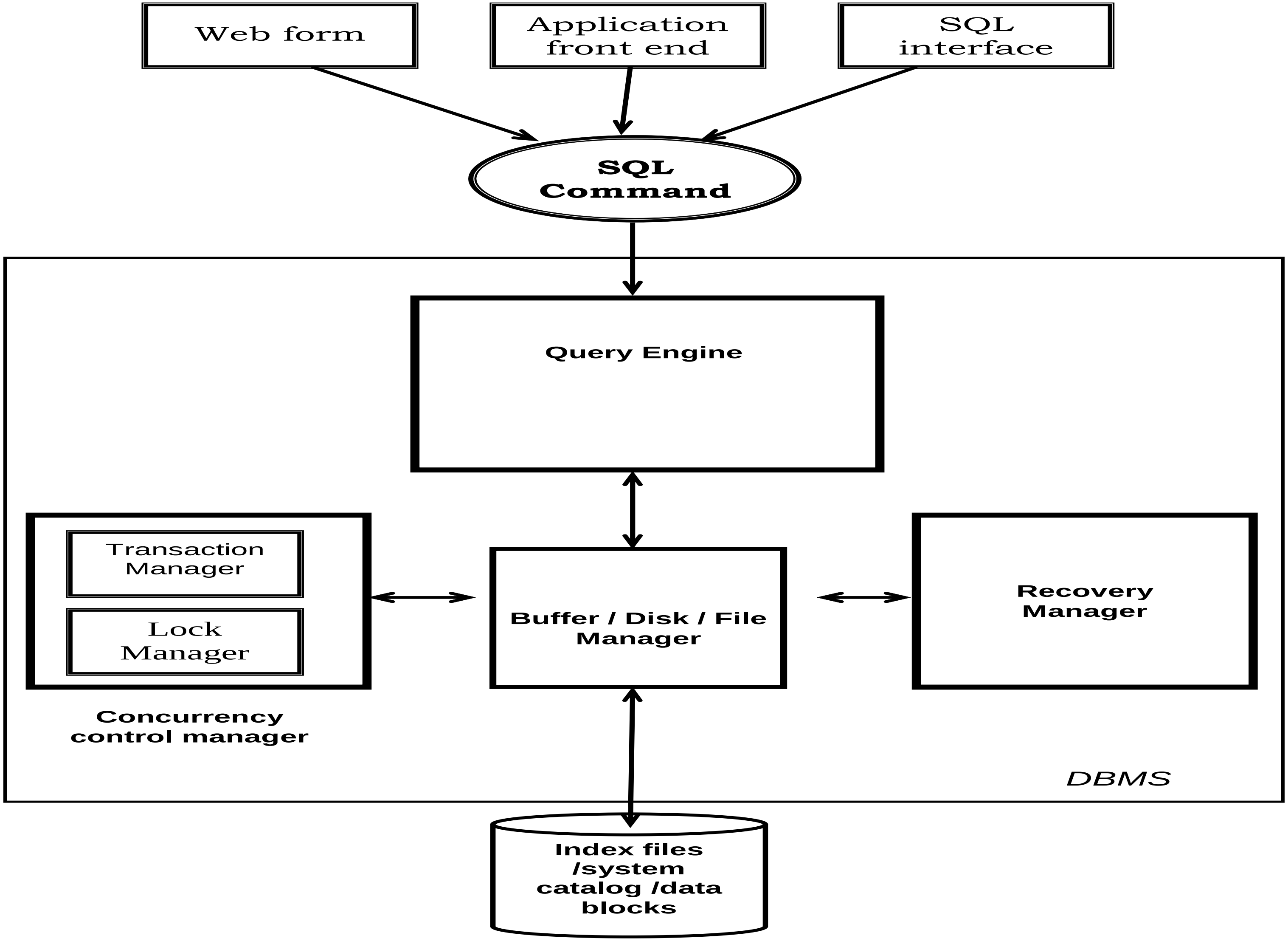
1. *Logical data independence:* changes in conceptual level schema should not affect the application level or external level schemas.

2. *Physical data independence:* The changes in physical features of storage, i.e., changes to the physical storage format should not affect schema at conceptual level.

The above data independence is one of the important advantages of DBMS.

The DBMS stores the description of schemas as System catalog.

DBMS Structure



RDBMS

- RDBMS stores data in the form of related tables.

- Relational databases are powerful because they require few assumptions about how data is related or how it will be extracted from the database.

- The same database can be viewed in many different ways.

- RDBMS, a software package which manages a relational database, optimized for rapid and flexible retrieval of data; also called a database engine.

- An important feature of relational systems is that a single database can be spread across several tables. This differs from flat-file databases, in which each database is self-contained in a single table.

Relational Database

- A relational database is a collection of relations or two-dimensional tables.

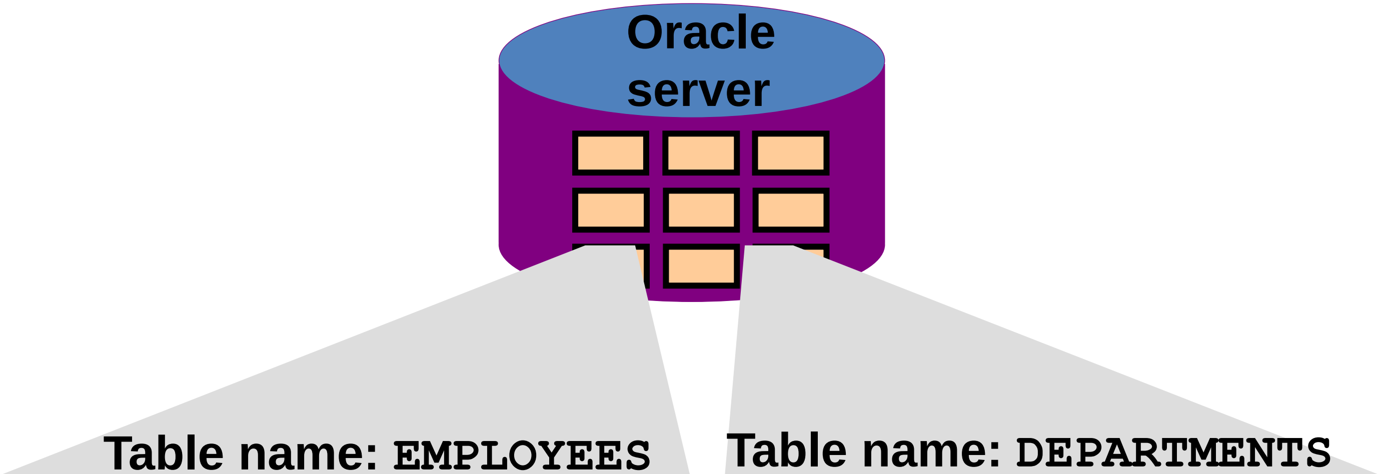
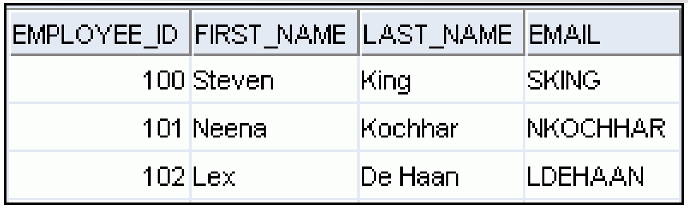


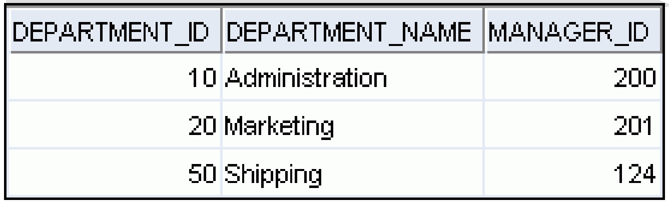
Table name: **`EMPLOYEES`**



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL
100	Steven	King	SKING
101	Neena	Kochhar	NKOCHHAR
102	Lex	De Haan	LDEHAAN

…

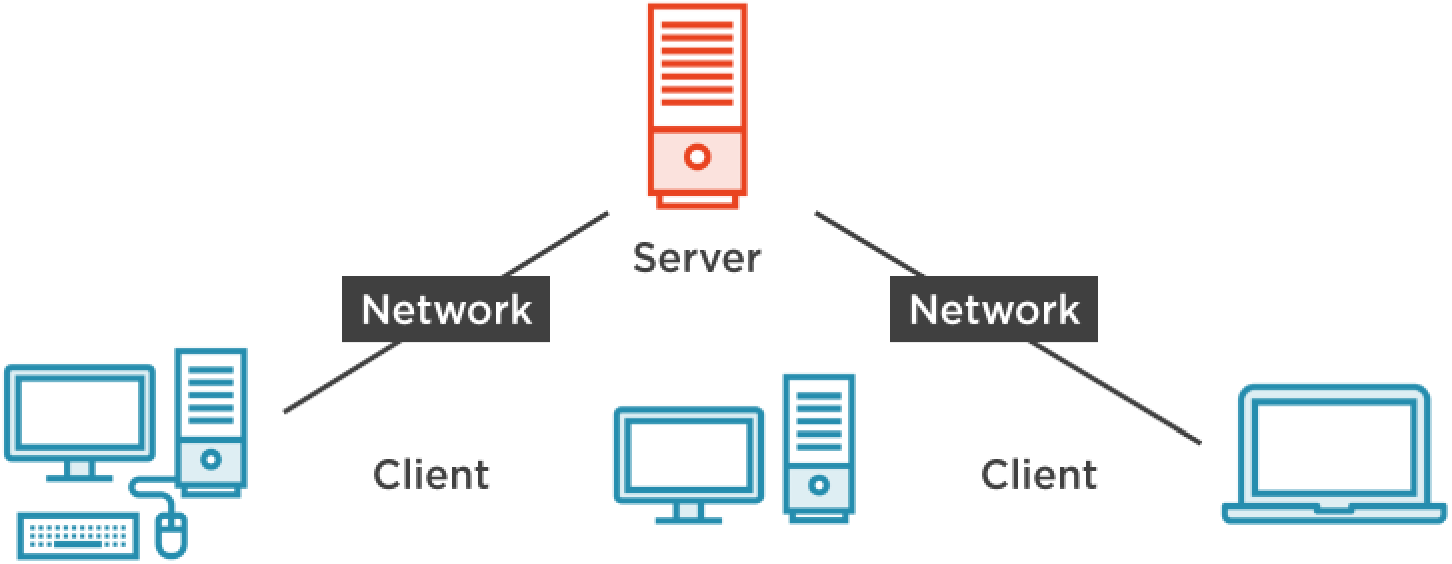
Table name: **`DEPARTMENTS`**



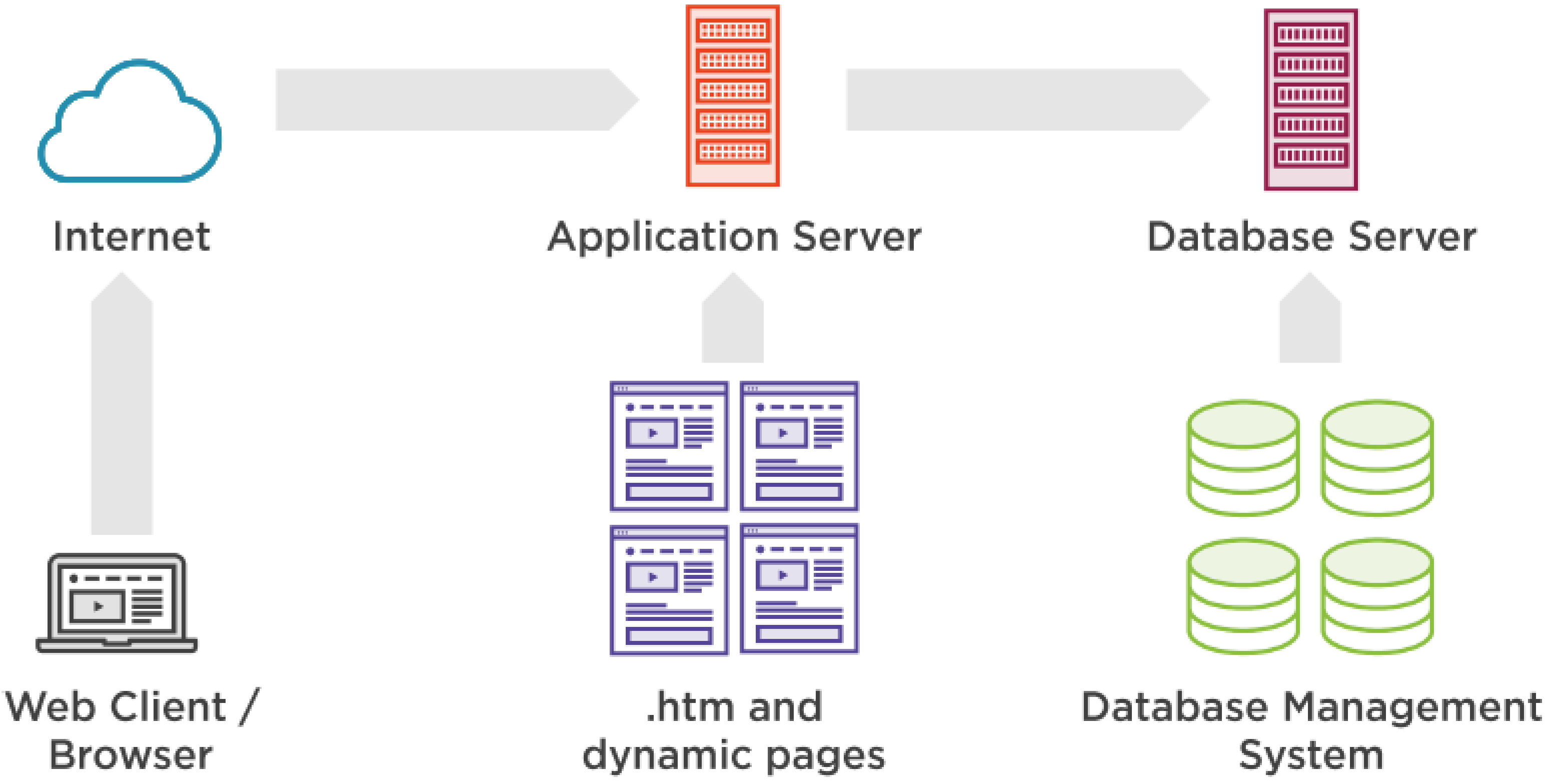
DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID
10	Administration	200
20	Marketing	201
50	Shipping	124

…

Client / server system



Web application server



Relational Database Model concepts

- Tables
- Columns(fields)
  - Attribute of the entity
  - Eg Street #, City, Pincode
- Rows (Records)
  - Set of values for a single instance of entity
  - Eg A single addres
- Cells
  - Intersection of a row and a column

Relational Database Model Keys

- Primary Key
  - Unique identifier of row
  - One per table
  - Does not allow NULL
  - Single or multiple columns(composite columns)

- Foreign Key
  - Columns in a table that refer to a Primary Key of another table
  - Enforces referential integrity
  - One-to-one
    - Eg one person has one address
  - One-to-many
    - Eg one person has residential address and office address
    - There is one address and many people are staying at the same address
  - Many-to-many
    - One address has multiple people staying there
    - And people having individual address too

Column Definition

- Data type determines the type of information
  - – String – CHAR,VARCHAR
  - – Integer- INT
  - – Float – FLOAT
  - – Date and time - DATE
- Default value
- Column containing NULL values
- Auto increment column