Master of Computer Applications

MCA Sem : 1

RDBMS 05MC0105

Marwadi University
Marwadi Chandarana Group

NAAC A+

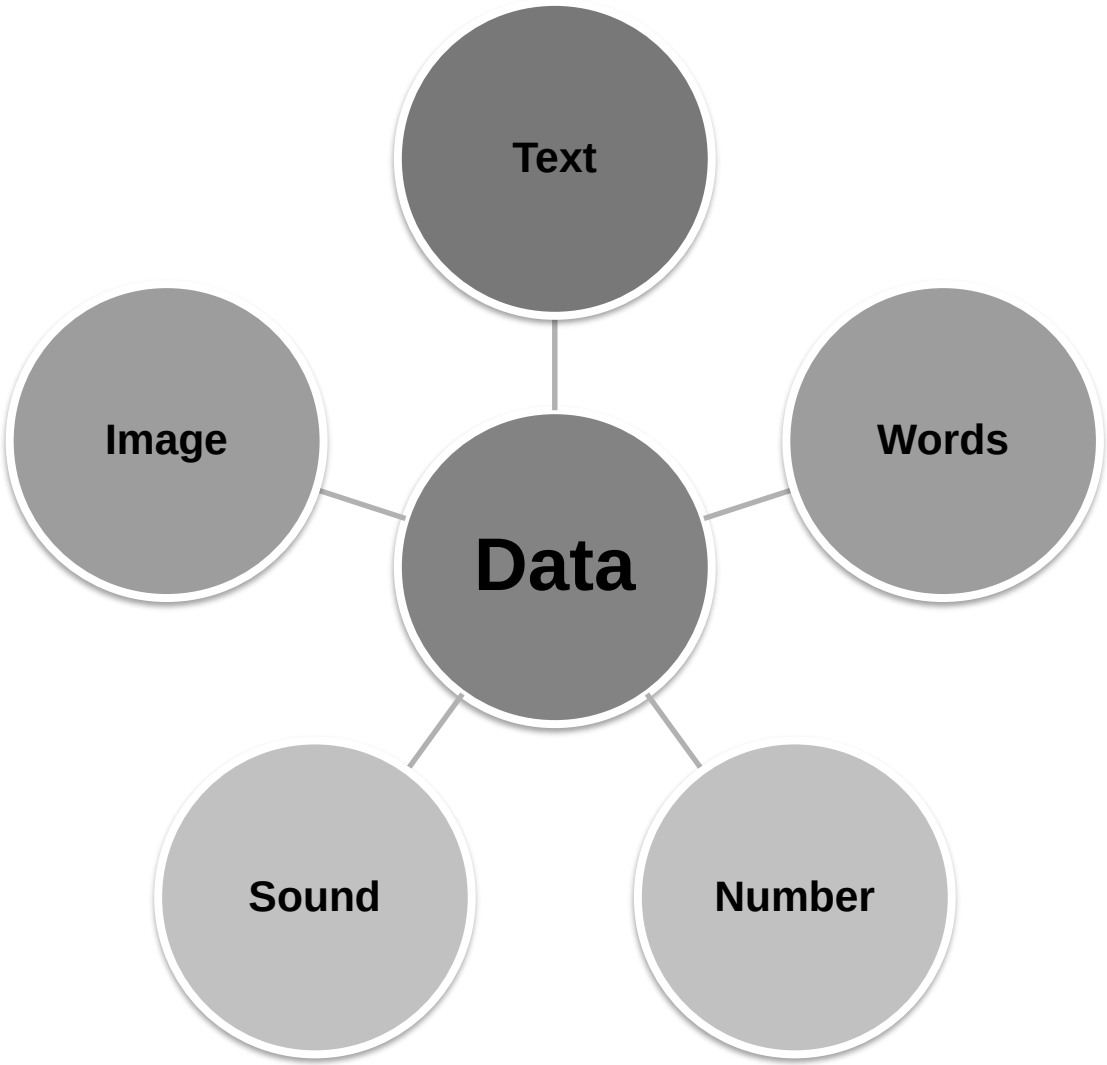
UNIT 1: Fundamentals of DBMS

TOPIC TO BE DISCUSSED.....

Marwadi University
Marwadi Chandarana Group
NAAC A+

1) Overview of Database Management systems
2) RDBMS
3) DBMS vs RDBMS
4) Characteristics of Database
5) Benefits of DBMS
6) Applications of DBMS
7) Database Users (Actors & Workers)
8) Schema vs Instance
9) Data Independence
10) Brief overview of Database Models
11) Database System Architecture.

Data



Data

- Data is a collection of facts, such as text, numbers, words, images , sounds measurements etc.
- Data is raw, unorganized facts that need to be processed.
- Example:
  - Marks of students
  - Name of employees
  - Building photos

Information

- Processed data is called information.
- A process includes:
  - Calculations
  - Comparisons
  - Organization
  - Sorting
  - Summarization

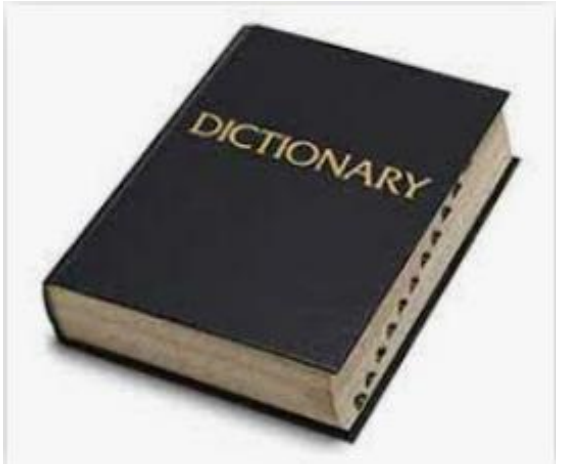
When data is processed, organized, structured or presented in a given context so as to make it useful, it is called information.

What Is Database?

- Database is an organized collection of meaningful data that is designed for a specific purpose.
- It’s a place where data is stored in such a way that we can retrieve data in a way that we want.
- Database stores each & every information in an organized way such as tables, views, reports etc.
- Database is collection of logically related data.

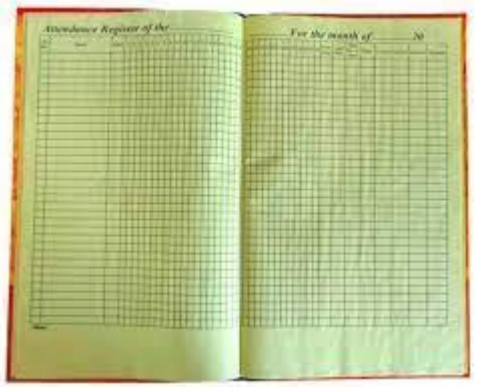
Some Real Word Example Of Database

Telephone Directory

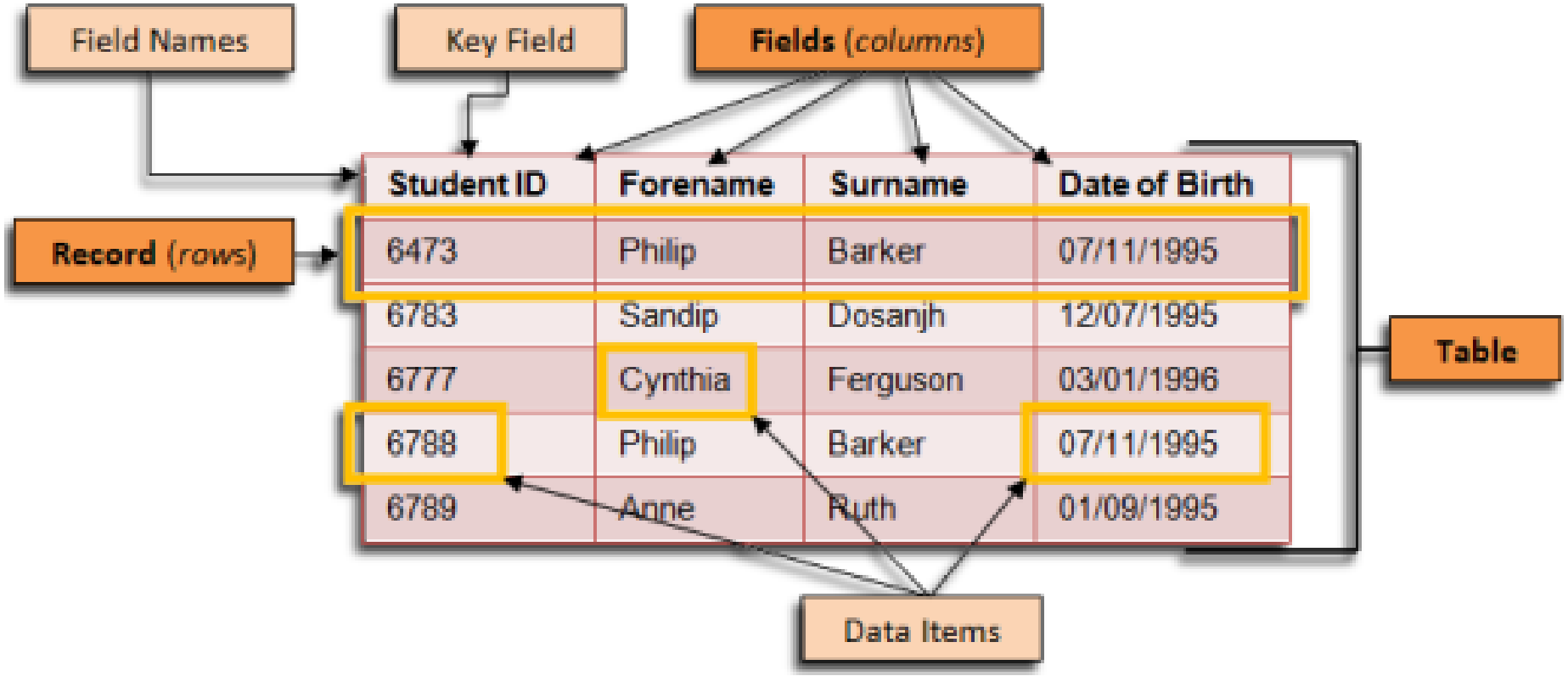


Dictionary

Employee Records



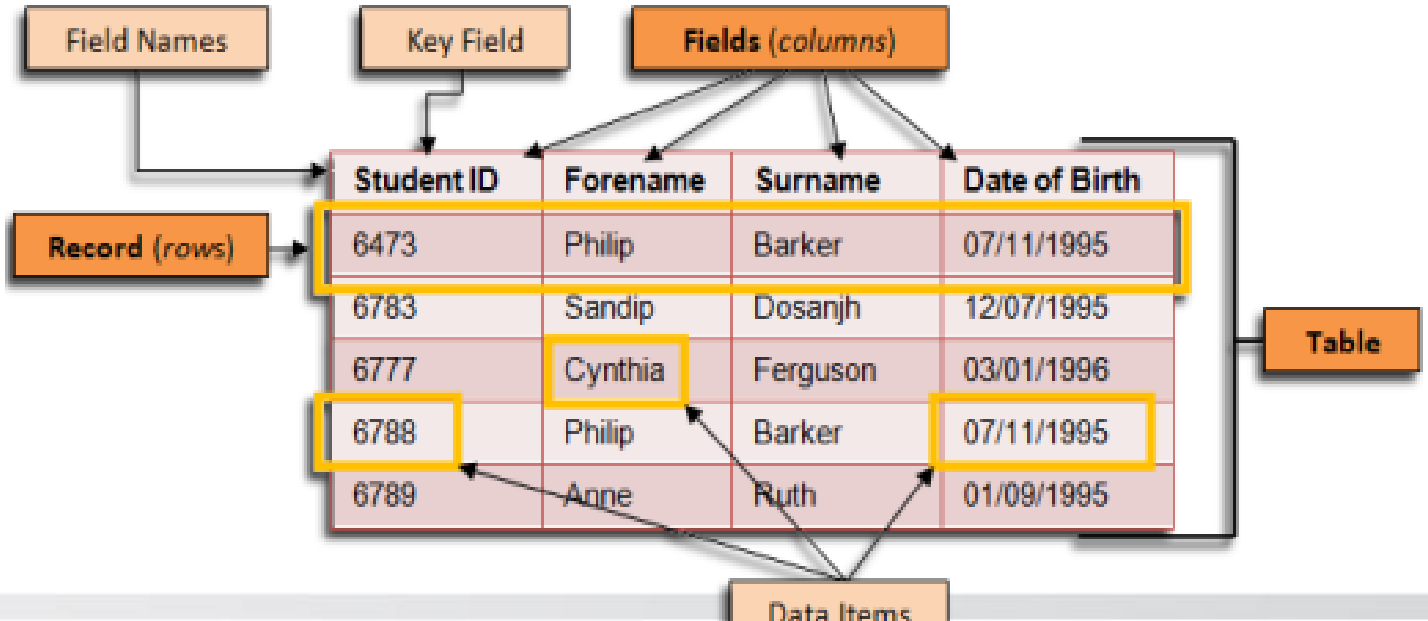
Attendance Register



Student ID	Forename	Surname	Date of Birth
6473	Philip	Barker	07/11/1995
6783	Sandip	Dosanjh	12/07/1995
6777	Cynthia	Ferguson	03/01/1996
6788	Philip	Barker	07/11/1995
6789	Anne	Ruth	01/09/1995

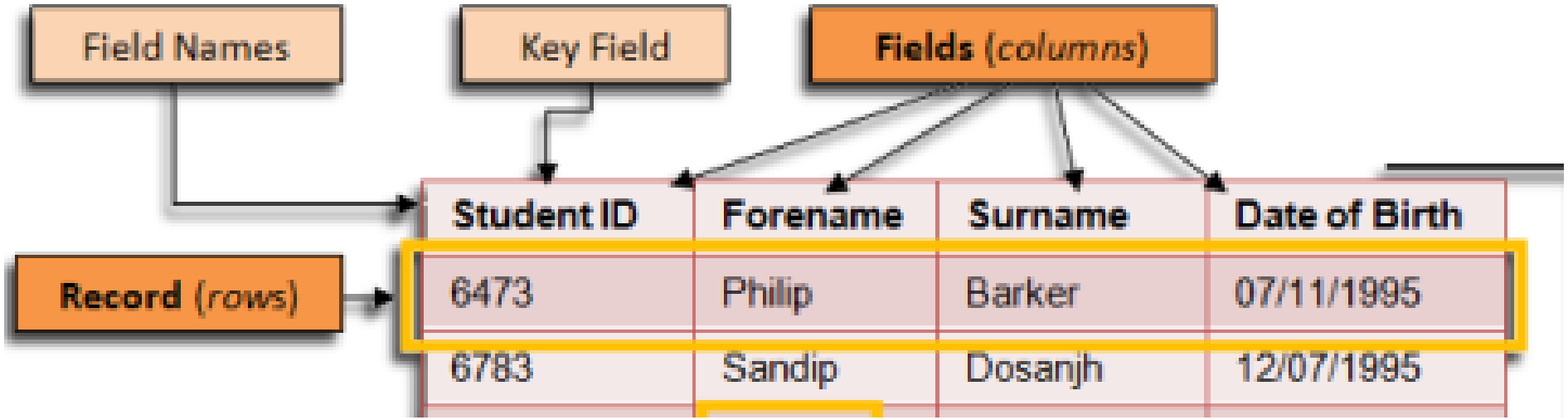
What Is Fields?

- A field is a **character** or group of characters that have a specific meaning.
- A **field /column** is a single piece of information.
- A field contain similar types of data.
- E.g:
  - Student Id, Forename Surname , Date Of Birth



What Is Records?

- Known as tuple /row / entity .
- A record is a **collection of logically related fields**.



What Is Metadata?

❖Data about data is called metadata That contain list of user along with tables, fields & datatype.

Student

Roll_No	Stud_Name	Sem
1	Ekta	1
2	Nihar	2
3	Mansi	3

Course

Sub_Code	Sub_Name	C_C
CS01	Oracle	10
CS02	Python	10
CS03	Java	10
CS04	Asp .Net	10

Meta data

tables	Fields
student	3
course	3

Column_name	Data_type	Belongs to
Roll_No	Number	Student
Stud_Name	Varchar	Student
Sem	Number	Student
Sub_code	Character	Course
Sub_name	Varchar	Course
C_C	Number	Course

RDBMS

- A **database management system (DBMS) is a computerized system that enables** users to create and maintain a database.
- The DBMS is a general-purpose software system that facilitates the processes of defining, constructing, manipulating, and sharing databases among various users and applications.
- **Defining a database** involves specifying the data types, structures, and constraints of the data to be stored in the database.

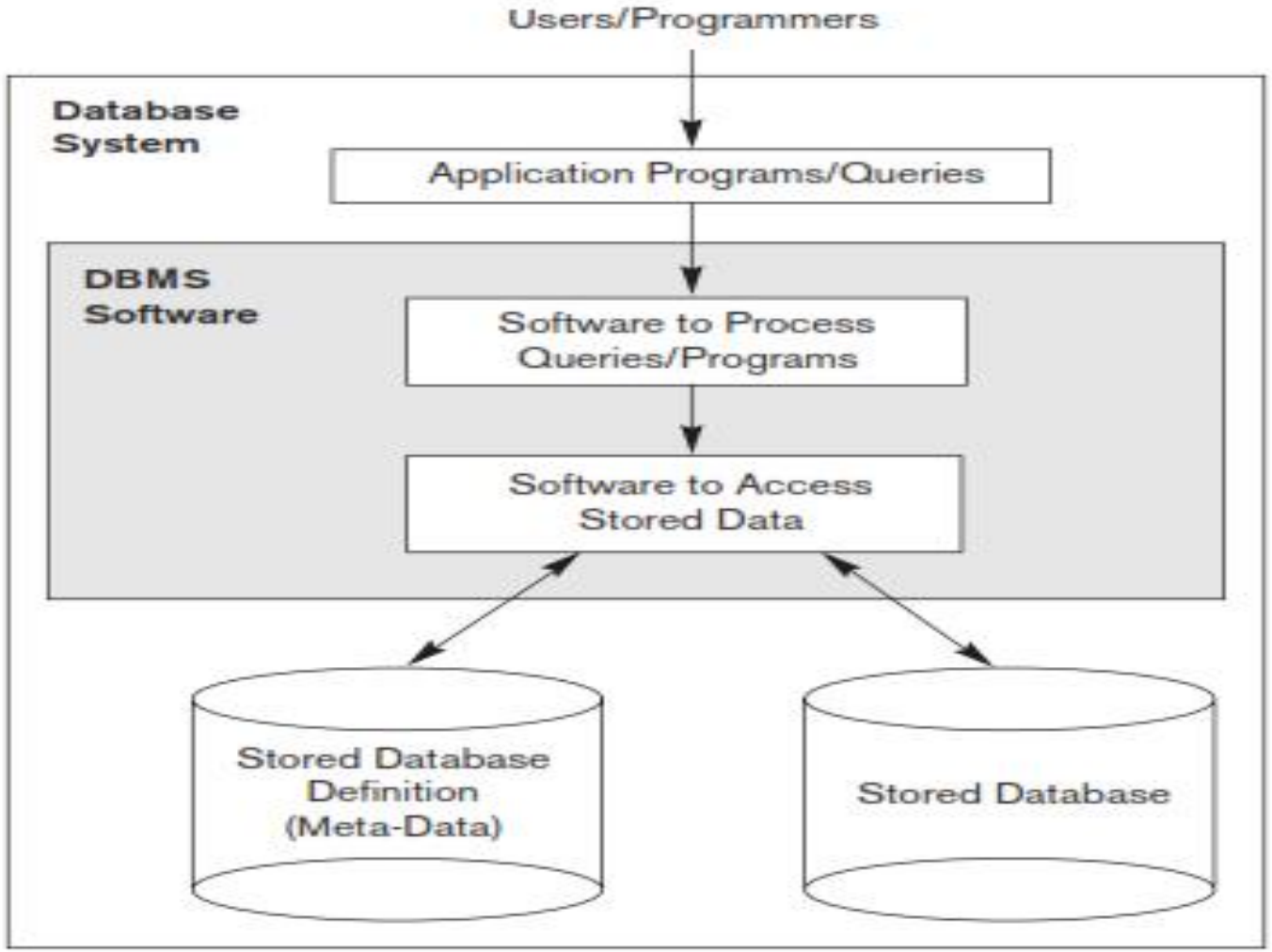
RDBMS

- For Example :
- **Traditional database applications, in which most of the information that is stored and accessed is either** textual or numeric.
- The majority of social media Web sites, such as Facebook, Twitter, and Flipcart, among many others, has required the creation of huge databases **that store nontraditional data**, such as posts, tweets, images, and video clips.
- New types of database systems, often referred to as **big data** storage systems, or **NOSQL systems,** have been created to manage data for social media applications.

- The database definition or descriptive information is also stored by the DBMS in the form of a database catalog or dictionary; it is called **meta-data.**
- A **query** typically causes some data to be retrieved;
- A **transaction may cause some data to be read and some data to be written into the database.**

DBMS SYSTEM ENVIRONMENT

A simplified database system environment.



RDBMS

- **RDBMS** stands for **R**elational **D**atabase **M**anagement **S**ystem. DBMS is the basis for SQL, and for all modern database systems like MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.
- A **Relational database management system (RDBMS)** is a database management system (DBMS) that is based on the relational model as introduced by E. F. Codd.
- The RDBMS is the most popular database system among organizations across the world. It provides a dependable method of storing and retrieving large amounts of data while offering a combination of system performance and ease of implementation.

DBMS vs RDBMS

DBMS vs RDBMS

DBMS	RDBMS
DBMS stands for Database Management System.	RDBMS stands for Relational Database Management System.
DBMS is a set of related records that used for accessing, storing and managing a data.	RDBMS is a set of software programs that used for creating, maintaining, modifying & determining relational database.
A General-purpose software provides the users with the process of defining, constructing & manipulating the database for various applications.	It can used to create the application that a user will require for interacting with the data that stored within the database.

DBMS vs RDBMS

DBMS	RDBMS
DBMS stands for Database Management System.	RDBMS stands for Relational Database Management System.
DBMS is a set of related records that used for accessing, storing and managing a data.	RDBMS is a set of software programs that used for creating, maintaining, modifying & determining relational database.
A General-purpose software provides the users with the process of defining, constructing & manipulating the database for various applications.	It can used to create the application that a user will require for interacting with the data that stored within the database.

DBMS vs RDBMS

DBMS	RDBMS
The performance of DBMS operation is not good	The RDBMS operation gives better performance
It is not based on the concept of relationships	It is based on the concept of relationships
The speed of DBMS operation is very slow	The speed of RDBMS operation is very faster
Hardware and Software requirements are less	Hardware and Software requirements are high
Facilities and Utilities are limited.	Provide High Facilities and Utilities.
It uses the concept of file	It uses the concept of table

DBMS vs RDBMS

DBMS	RDBMS
DBMS uses a 3GL (Third Graphical Language)	RDBMS uses a 4GL (Fourth Graphical Language)
Provides less security as compared to RDBMS	Provides high security as compared to DBMS.
Normalization is not present	Normalization is present
It deals with small quantity of data.	It can work with large amount of data.
It support single user.	Multiple user can access data.

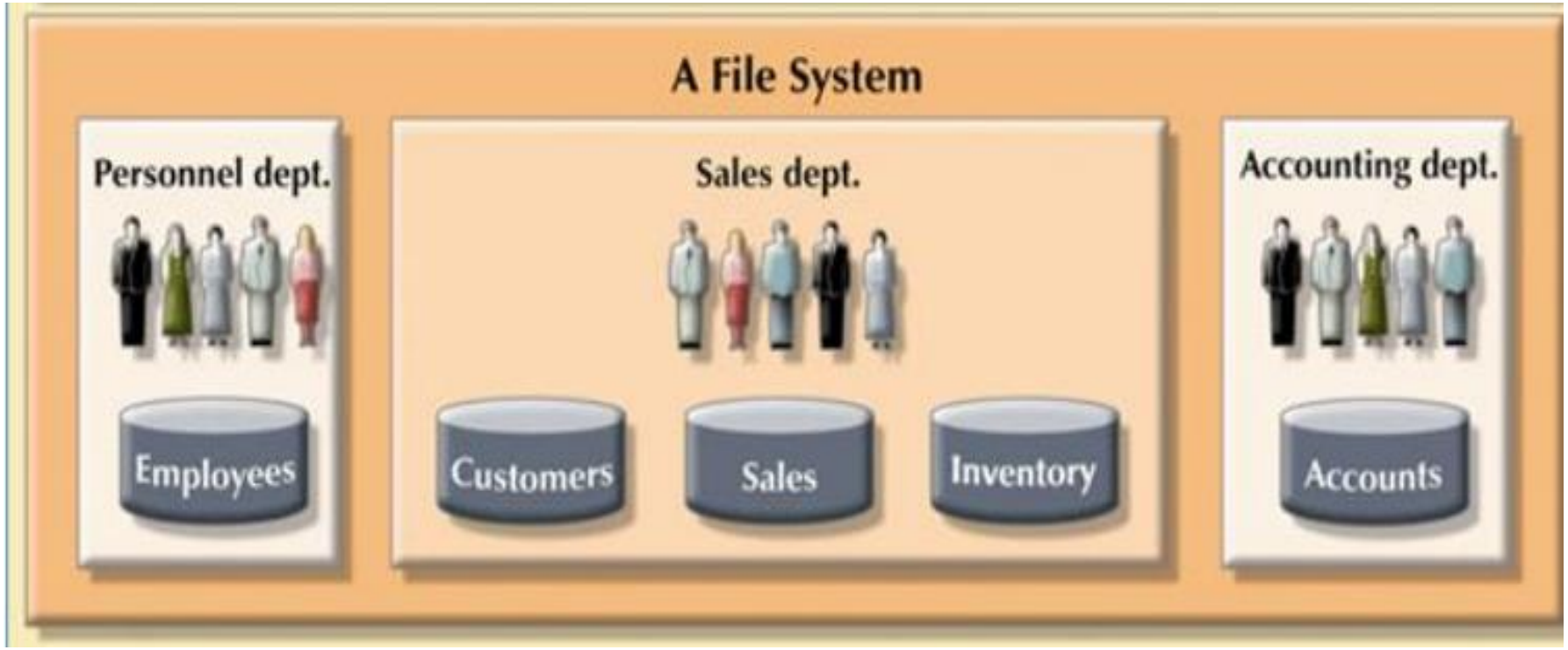
DBMS vs RDBMS

DBMS	RDBMS
Data redundancy is the issue.	Data redundancy is reduced.
Examples of DBMS are : DBASE DBASE-II FOXBASE FOXPRO	Examples of RDBMS are: ORACLE SQL SERVER MYSQL INGERS

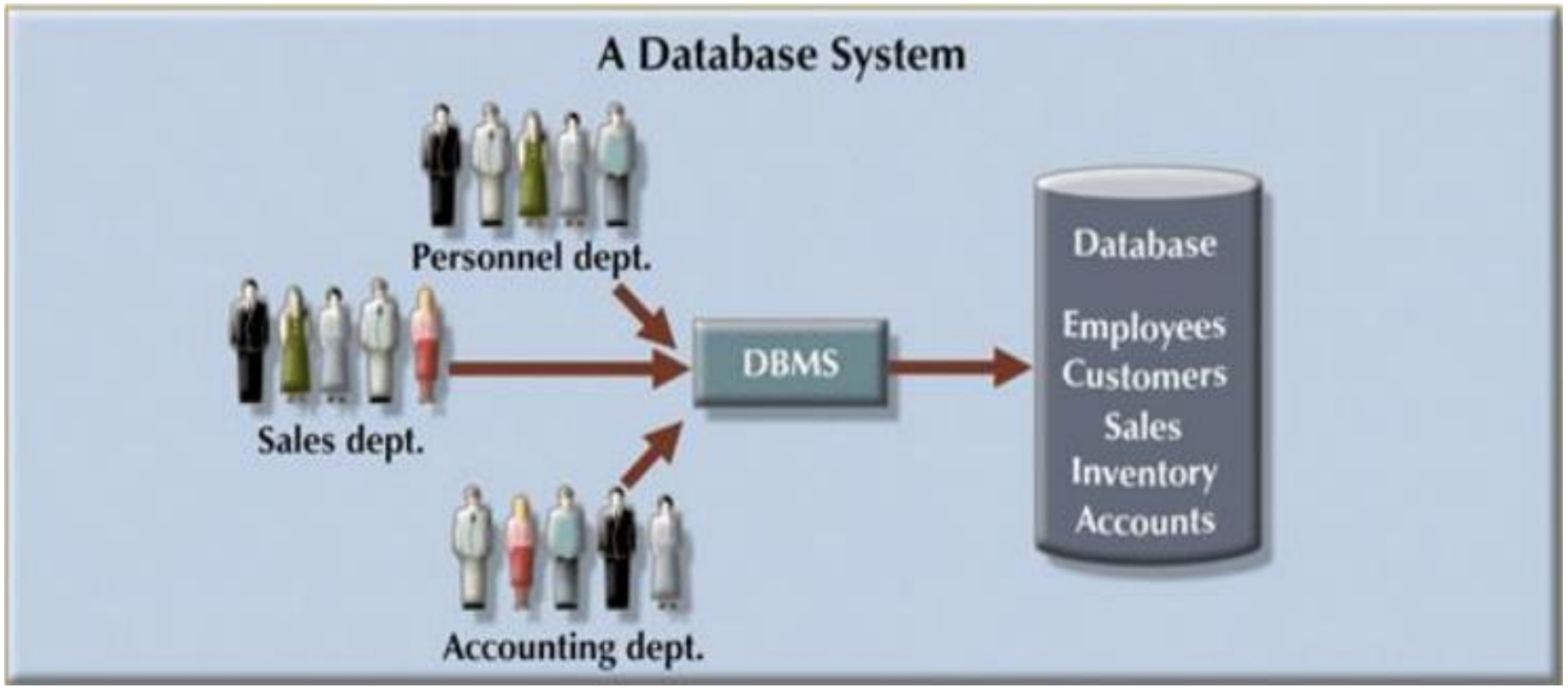
Advantages of DBMS

- Data Sharing
- Reduce Data Redundancy
- Remove Data Inconsistency
- Data Validation
- Data Security
- Backup And Recovery
- Faster Data Access
- Garneted atomicity

Data Sharing



Data Sharing



Data Sharing is possible between multiple users.

Reduce Data Redundancy (Duplication)

Marwadi University
Marwadi Chandarana Group

NAAC A+

Computer

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	1234	CPU

Civil

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	1234	CPU

Database management system **can remove such data redundancy** by storing data centrally.

Same data is stored at **four different places.**

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	1234	CPU

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	1234	CPU

Electrical

Mechanical

Remove Data Inconsistency

Marwadi University
Marwadi Chandarana Group
NAAC A+

Computer

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	6789	CPU

Civil

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	1234	CPU

Database management system can keep data in **consistent state**.

Same data having **different state (values)**

Mobile no is changed

Electrical

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	6789	CPU

Mechanical

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	1234	CPU

Data Validation

Roll_No	Stud_Name	Sem	Elective_sub	Percentage
1	Ekta	1	Cyber security	80.25
2.5	Nihar	2		75
3	Mansi	30	Cloud computing	65

Roll number should not be fractional

Semester must be between 1 to 6

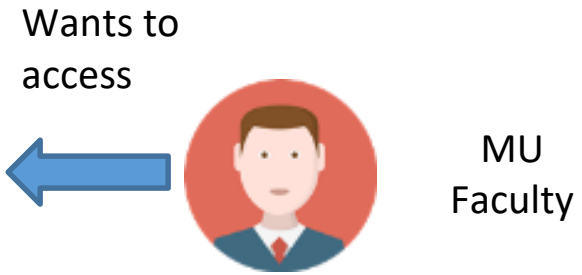
Elective subject must not be blanked

Data Security

Emp_Name	Address	Mob	Subject
Prof. Ajay Shah	Rajkot	1234	CPU

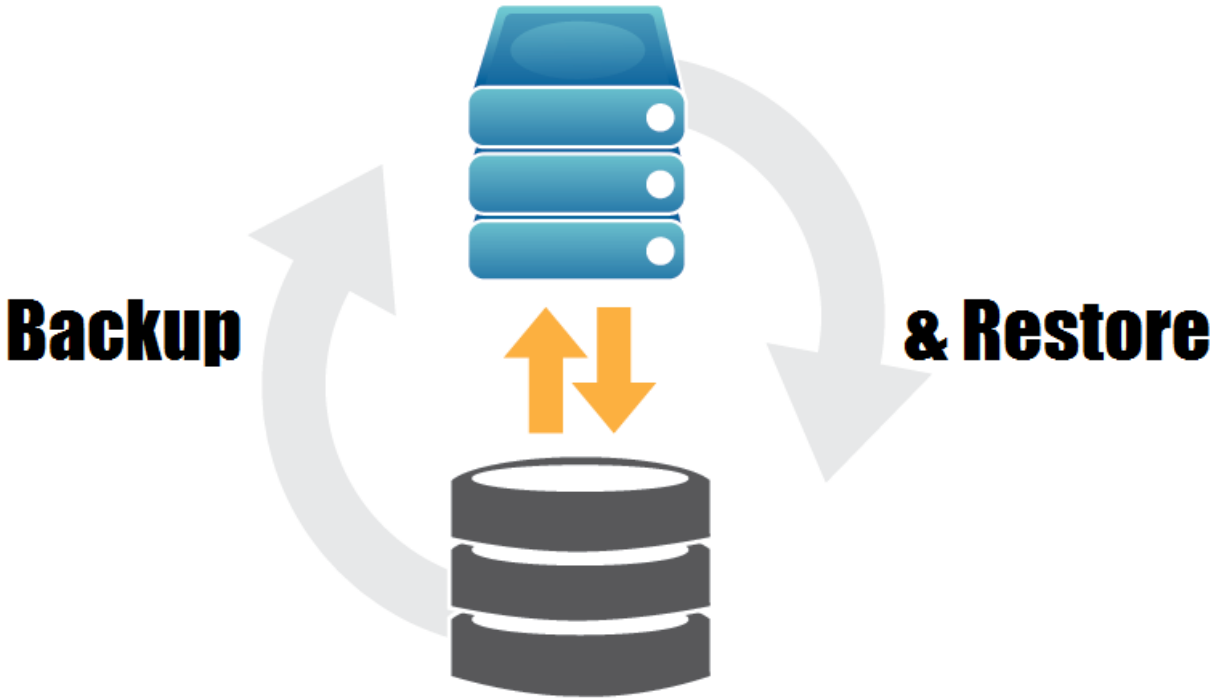
Emp_Name	Salary	Load
Prof. Ajay Shah	50000	20

Emp_Name	Teaching	Knowledge	Rating
Prof. Ajay Shah	Good	V. Good	9



DBMS **prevents unauthorized user** to access data.

Backup & Recovery



Faster data access

ProgrammeID	Programmename	Duration	Department	Fee
1	Bsc.in IT	4	Computing	3000
2	Bsc Software Development	3	Software	3500
3	Bsc Applied Computingg	4	Computing	4000
4	Business in IT	3	Software	3000
5	Bsc. Acturial science	3	Business	4000
6	Bachelor of commerce	3	Business	3000

STUDENT

StudentID	Firstname	Lastname	Age	ProgrammeID
1	Henry	Mutua	21	1
2	Henry	Kamau	21	2
3	Mary	Otieno	20	5
4	Susan	Kamau	19	2
5	Ahmed	Hassan	20	2
6	Joan	Kamau	18	3
7	Peter	Wekesa	23	4
8	Peter	Muriuki	22	3
9	Leonard	Mwang'ombe	21	6
10	Mary	Ratemo	20	2

DBMS allows user to **access data all or any piece of data** from database.

Guaranteed Atomicity

- Either transaction execute 0% or 100%.

Person A
Account A
Bal : 2000

Sum of both account before transfer is **3000**

Person B
Account B
Bal : 1000

Transfer **500**

Step 1 : Debit 500 from Account A
Step 2 : Credit 500 into Account B

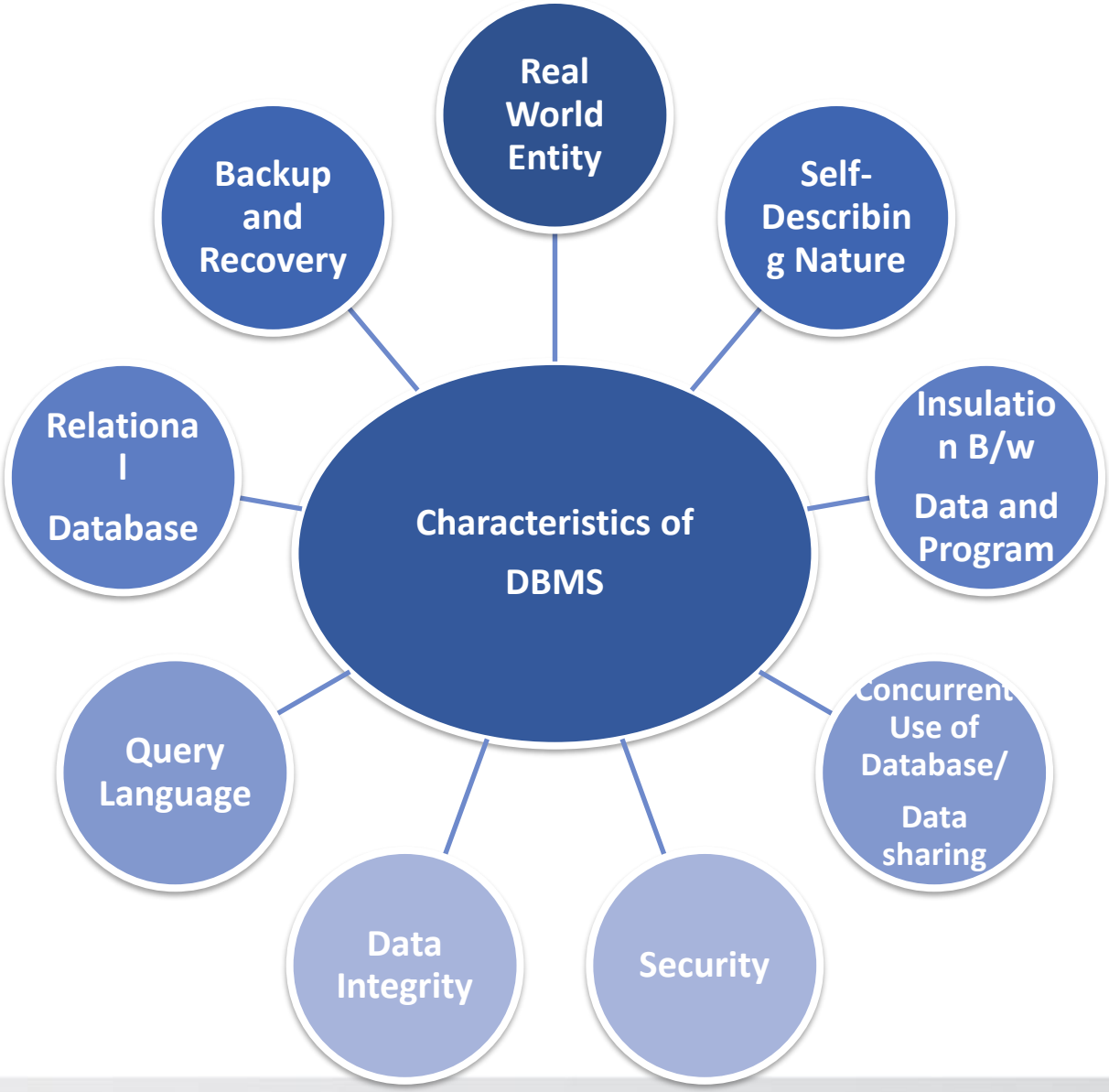
Transaction is failed

Sum of both account after transfer is **3000**

Sum of both account is 2500 so **inconsistent**

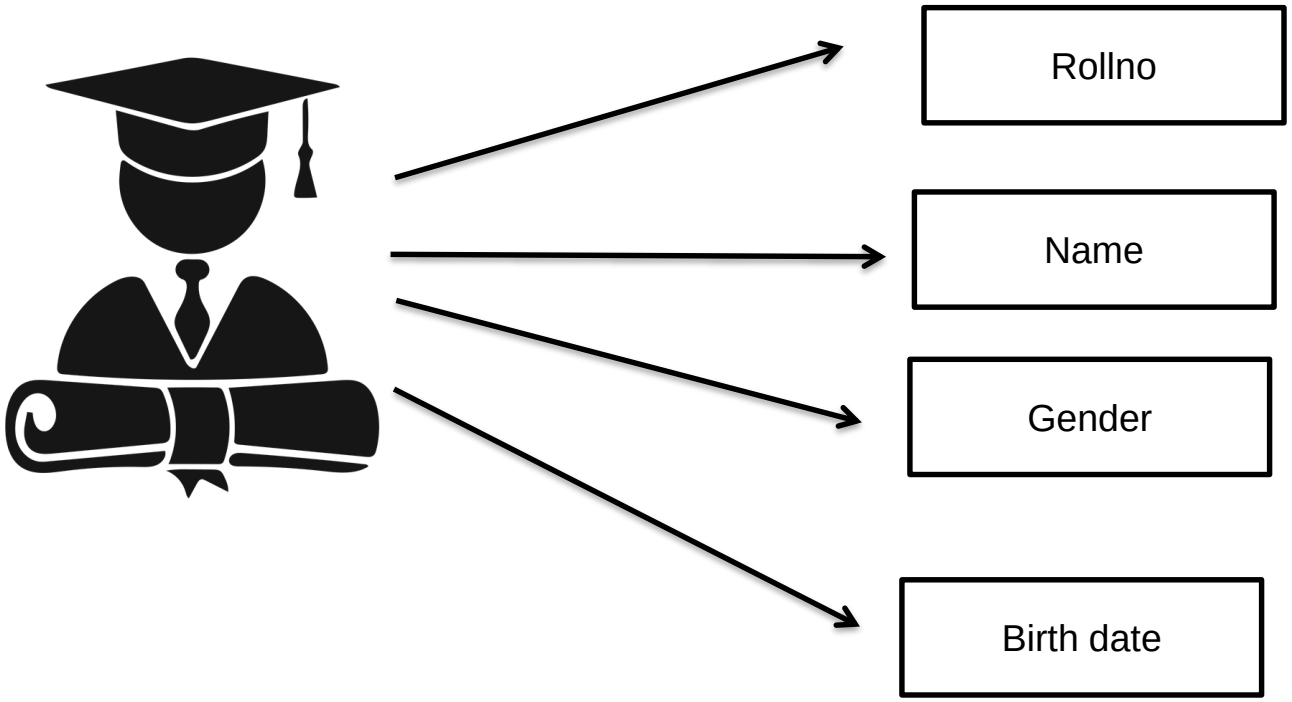
Characteristics Of DBMS

Characteristics of DBMS



Real World Entity

- Use a real-world entities to design its architecture.
- Also, behavior and attributes are used by DBMS.
- For example



Self Describing Nature

- Database system consists data & meta data.
- Metadata maintains automatically.
- The meta data is stored in the catalog which is used by DBMS software & Users.

emlployee_id	first_name	last_name	nin	department_id
44	Simon	Martinez	HH 45 09 73 D	1
45	Thomas	Goldstein	SA 75 35 42 B	2
46	Eugene	Comelsen	NE 22 63 82	2
47	Andrew	Petculescu	XY 29 87 61 A	1
48	Ruth	Stadick	MA 12 89 36 A	15
49	Barry	Scardelis	AT 20 73 18	2
50	Sidney	Hunter	HW 12 94 21 C	6
51	Jeffrey	Evans	LX 13 26 39 B	6
52	Doris	Berndt	YA 49 88 11 A	3
53	Diane	Eaton	BE 08 74 68 A	1
54	Bonnie	Hall	WW 53 77 68 A	15
55	Taylor	Li	ZE 55 22 80 B	1

Data

Meta data

Column	Data Type	Description
emlployee_id	int	Primary key of a table
first_name	nvarchar(50)	Employee first name
last_name	nvarchar(50)	Employee last name
nin	nvarchar(15)	National Identification Number
position	nvarchar(50)	Current postion title, e.g. Secretary
department_id	int	Employee deparmtnet. Ref: Departmetns
gender	char(1)	M = Male, F = Female, Null = unknown
employment_start_date	date	Start date of employment in organization.
employment_end_date	date	Employment end date. Null if employee sti

Insulation Between Data & Program

- In DBMS structure of database stored in the catalog & programs to access data stored in different locations.
- So that any changes got to structure of database it will not affected to program .

Concurrent Use of Database

- Multiple users are allowed to access data simultaneously.
- Concurrent access to centralized data can be allowed under some supervision.
- It enhance the performance.
- For Example
  - Railway reservation

Security

- The database should be accessible to the users in a limited way.
- The access to make changes to a database by the user should be limited and the users must not be given complete access to the entire database.

Data integrity

- Data in database must be correct and consistent.
- So, data stored in database must satisfy certain types of constraints (rules).
- DBMS provides different ways to implement such type of constraints (rules).
- This improves data integrity in a database.

Data integrity

Student

Roll_No	Stud_Name	Elle_Sub_Code
1	Ekta	CS01
2	Nihar	CS02
3	Mansi	CS01
4	Zarmar	CS05
6	Aditya	CS04

Data integrity Issue

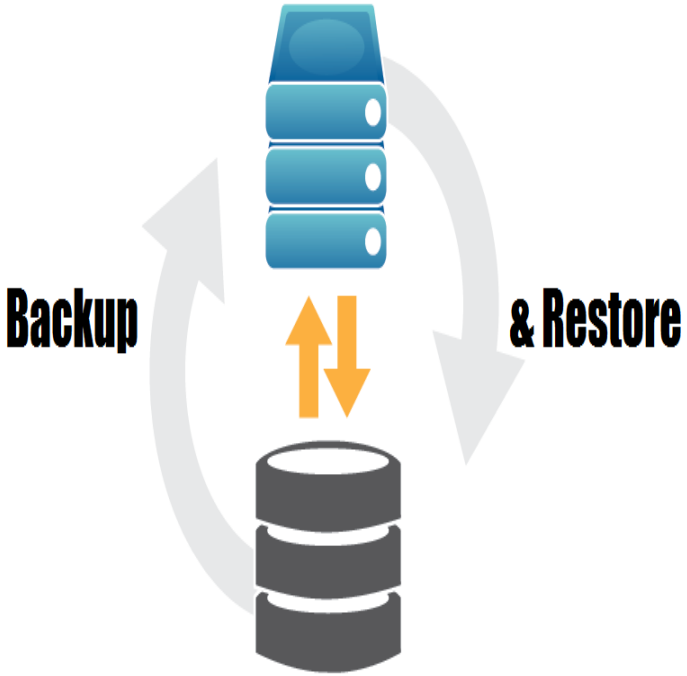
Course

Elle_Sub_Code	Sub_Name	C_C
CS01	Oracle	10
CS02	Python	10
CS03	Java	10
CS04	Asp .Net	10

Query Language

- Queries are used to retrieve and manipulate data but DBMS is armed by a strong query language that makes it more effective and efficient.
- By using query language users can store & retrieve data.
- Users have the power to retrieve any kind of data they want from the database by applying different sets of queries.

Backup & recovery

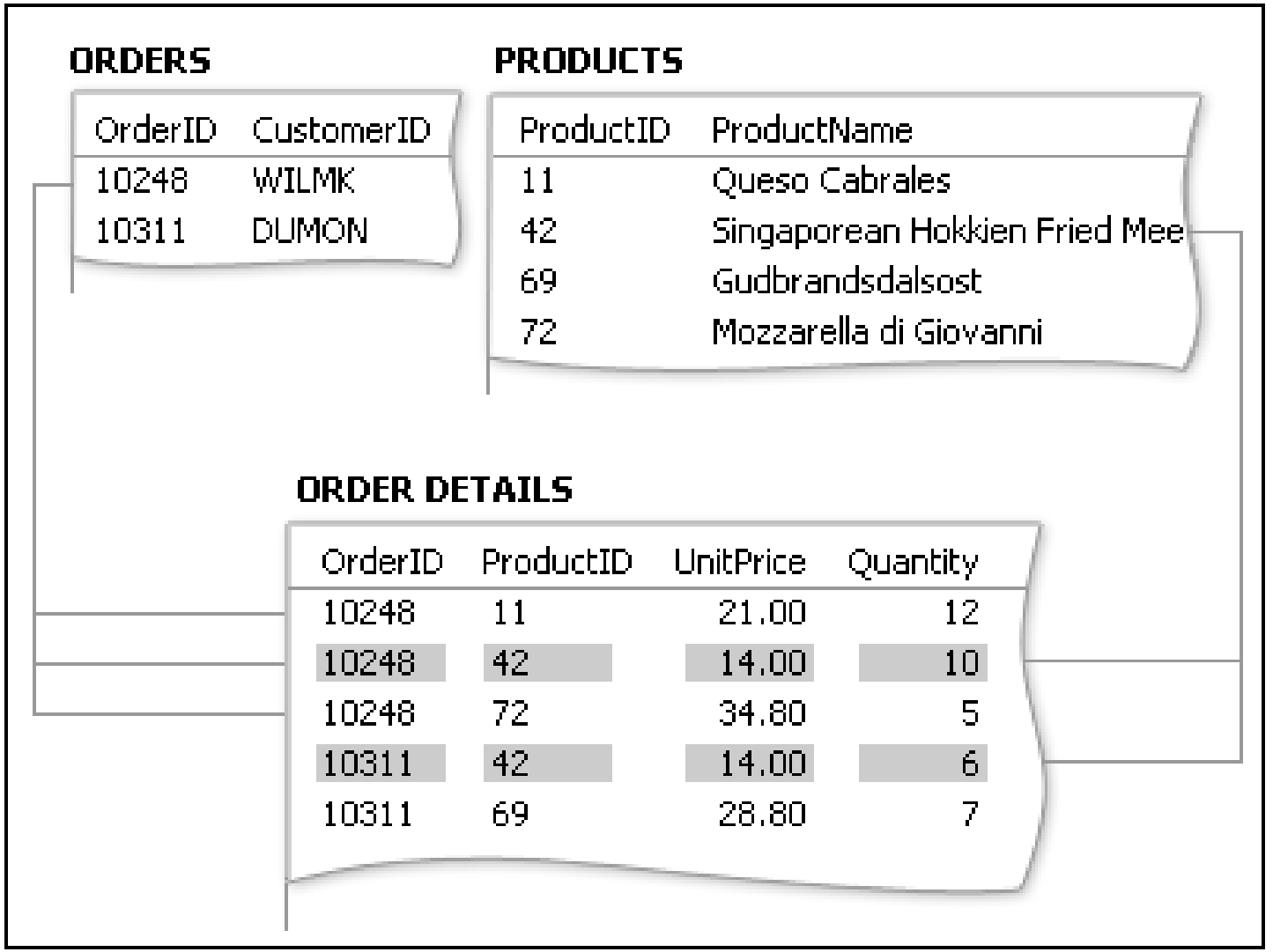


- There are many chances of failure of the whole database.
- At that time no one will be able to get the database back and for sure company will be in a big loss.
- The only solution is to take backup of database and whenever it is needed, it can be stored back.
- A database must have this characteristic to enable more effectiveness.

Relational databases

- Relational databases organize data into tables consisting of rows and columns.
- Each row represents a single record while each column holds a value associated with a specific field.
- Tables may be linked together to form relationships between them.

Relational databases



ORDERS

OrderID	CustomerID
10248	WILMK
10311	DUMON

PRODUCTS

ProductID	ProductName
11	Queso Cabrales
42	Singaporean Hokkien Fried Mee
69	Gudbrandsdalsost
72	Mozzarella di Giovanni

ORDER DETAILS

OrderID	ProductID	UnitPrice	Quantity
10248	11	21.00	12
10248	42	14.00	10
10248	72	34.80	5
10311	42	14.00	6
10311	69	28.80	7

- A number of characteristics distinguish the database approach from the much older approach of writing customized programs to access data stored in files.

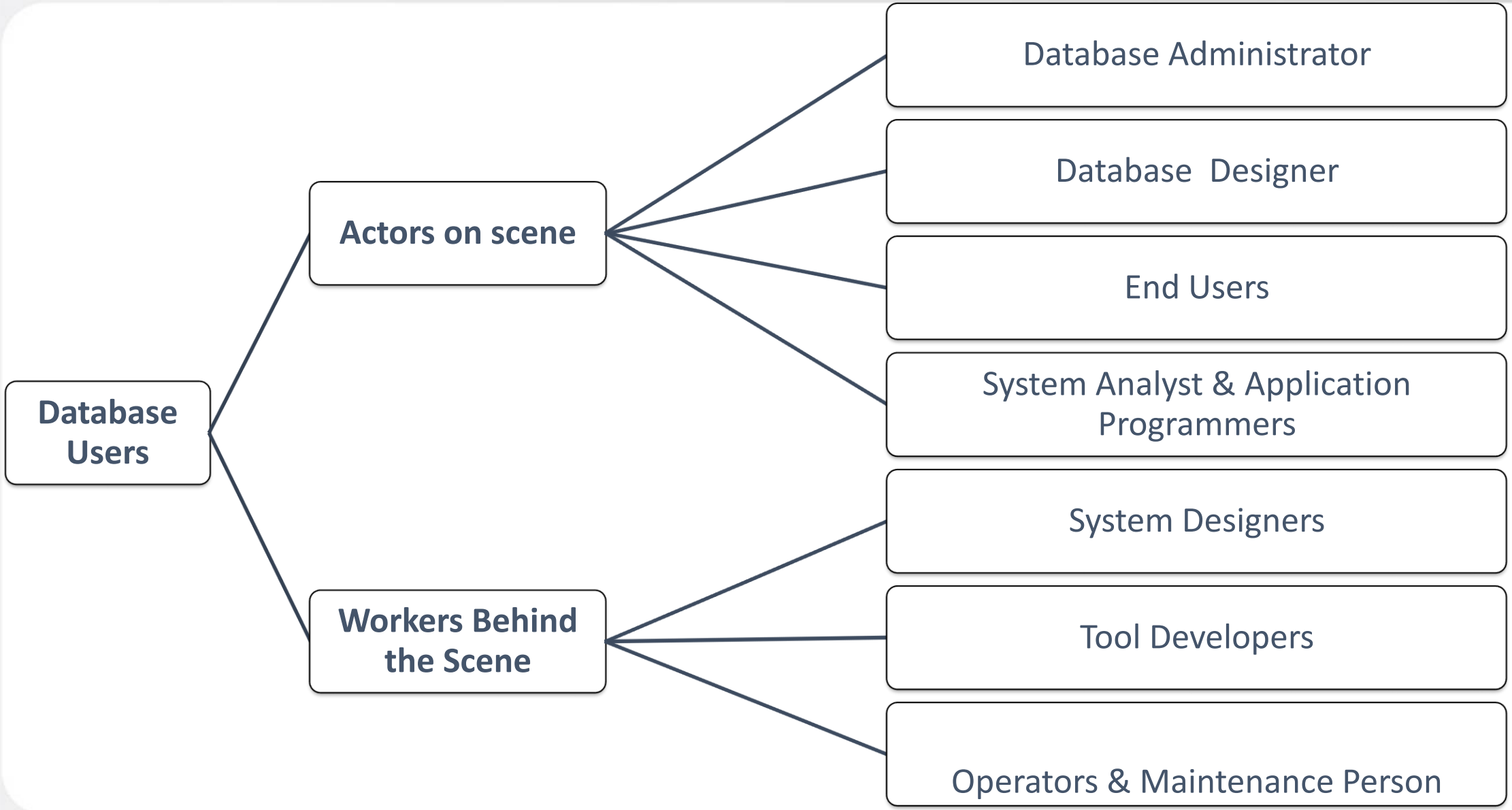
- In traditional **file processing, each user defines and implements the files needed for a** specific software application as part of programming.

- This redundancy in defining and storing data results in wasted storage space and in redundant efforts to maintain common up-to-date data.g the application.

Actors on Scene

- Users may be divided into two categories:
  - Those who actually use and control the content (called “Actors on the Scene”)
  - Those who enable the database to be **developed** and the DBMS software to be designed and implemented (called “Workers Behind the Scene”).

Actors on Scene



Actors on Scene: Database administrators

- **Database administrators** responsible for :
  - Access to the database
  - Coordinating and monitoring its use
  - Acquiring software/hardware resources
  - Controlling its use and monitoring run-time performance.

Actors on Scene: Database Designers

- Database Designers responsible to define the content, structure, constraints, and functions or transactions against the database.
- They communicate with the end-users and understand their needs.
- It is the responsibility of database designers to communicate with all prospective database users in order to understand their requirements and to create a design that meets these requirements.

Actors on Scene: END USERS

- End users are the people whose jobs require access to the database for querying, updating, and generating reports; the database primarily exists for their use.
- There are several categories of end users:

  (1) Casual End users

  (2) Naive or parametric end users

  (3) Sophisticated end users

  (4) Standalone users

Actors on Scene: CAUSAL USER

- This user occasionally access the database, but they may need different information each time.
- They use a sophisticated database query interface to specify their requests and are typically middle- or high-level managers or other occasional browsers.

Actors on Scene: Naive or parametric end users

- Their main job function is around constantly querying and updating the database, using standard types of queries and updates called canned transactions—that have been carefully programmed and tested.
- Many of these tasks are now available as mobile apps for use with mobile devices.
- Few examples are:
  - Bank customers and tellers check account balances and post withdrawals and deposits.
  - Reservation agents or customers for airlines, hotels, and car rental companies check availability.
  - Social media users post and read items on social media Web sites.

Actors on scene: Sophisticated end users

- Include engineers, scientists, business analysts, and others who thoroughly familiarize themselves with the facilities of the DBMS in order to implement their own applications to meet their complex requirements.

Actors on scene: standalone user

- They maintain personal databases by using ready-made program packages that provide easy-to-use menu-based or graphics-based interfaces.

- An example is the user of a financial software package that stores a variety of personal financial data.

Workers behind the Scene

- In addition to those who design, use, and administer a database, others are associated with the design, development, and operation of the DBMS *software and system environment.*
- These persons are typically not interested in the database content itself. **We call them the *workers behind the scene*,** and they include the following categories:

(1) DBMS system designers and implementers

(2) Tool developers

(3) Operators and maintenance personnel

Workers behind the Scene: DBMS system designers and implementers

- Design and implement the DBMS modules and interfaces as a software package.
- A DBMS is a very complex software system that consists of many components, or modules, including modules for

  - Implementing the catalog,
  - Query language processing,
  - Interface processing,
  - Accessing and buffering data,
  - Controlling concurrency, and handling data recovery and security.

Workers behind the Scene: Tool developers

- Design and implement tools—the software packages that facilitate database modeling and design, database system design, and improved performance.
- Tools are optional packages that are often purchased separately.

Workers behind the Scene: Operators and maintenance personnel

- They are responsible for the actual running and maintenance of the hardware and software environment for the database system.
- Although these categories of workers behind the scene are instrumental in making the database system available to end users, they typically do not use the database contents for their own purposes.

Schema vs Instance

- In a data model, it is important to distinguish between the ***description of the*** **database** and the ***database itself.***
- *The description of a database is called the* **database schema, which is specified during database design and is not expected** to change frequently.
- A displayed schema is called a **schema diagram.**

Schema vs Instance

Figure 2.1
Schema diagram for the database in Figure 1.2.

STUDENT

Name	Student_number	Class	Major

COURSE

Course_name	Course_number	Credit_hours	Department

PREREQUISITE

Course_number	Prerequisite_number

SECTION

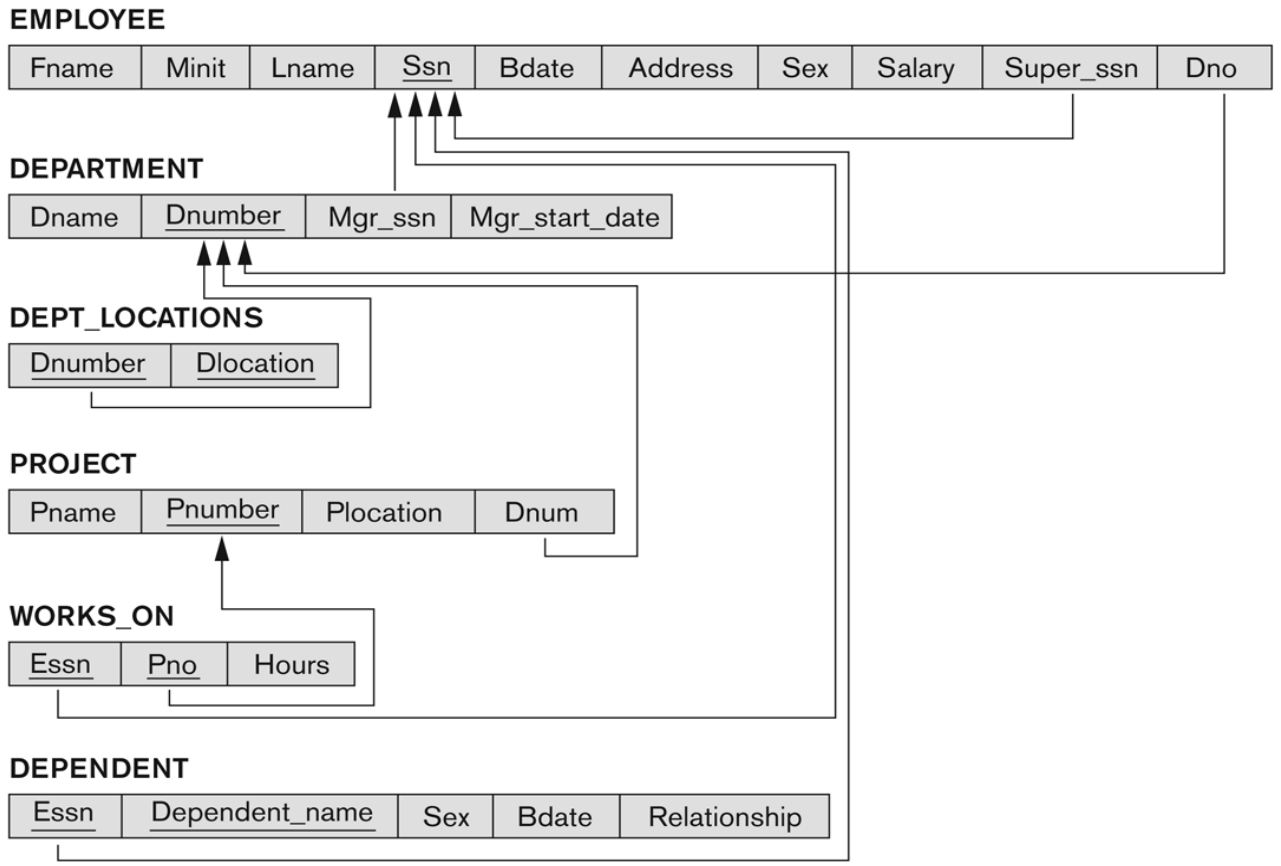
Section_identifier	Course_number	Semester	Year	Instructor

GRADE_REPORT

Student_number	Section_identifier	Grade

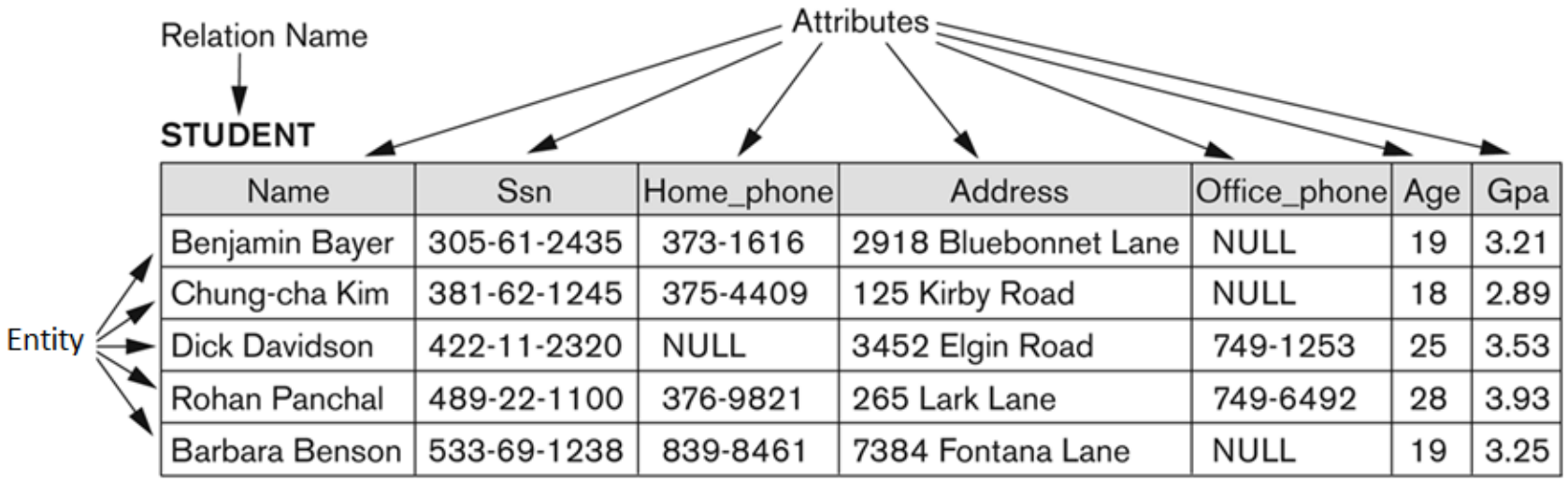
Schema vs Instance

Figure 5.7
Referential integrity constraints displayed on the COMPANY relational database schema.



Instance

- Instance of any table represents a row (record). A table contains set of instances.
- *for example, the* STUDENT construct (Table) will contain the set of individual student entities (records) as its **instances**.



Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	749-1253	25	3.53
Rohan Panchal	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	NULL	19	3.25

Figure 5.1
The attributes and tuples of a relation STUDENT.

Instance

❖ **Concept of Database State :**

❖ When every any operation is occurred on database , at that time it changes database state.

❖ **Empty state**

❖ When a new database is defined, it specify only database schema to the DBMS

❖ At this point, the corresponding database state is **the *empty state*** with no data.

❖ **Initial State :**

❖ When the database is first **loaded or populated then it database is in initial state.**

Current State

- An update operation is applied to the database
- At any point in time, the database has a ***current state.***

Valid State:

- A state that satisfies the structure and constrains specified in the schema.
- The ***database schema*** changes very infrequently.
- The ***database state*** changes every time the database is updated.

- **Schema** is also called **intension**.
- **State** is also called **extension**.

Schema vs Instance

Schema	Instance
The overall design of database is called schema	Instances are the collection of information stored at a particular moment.
The schema remains same for whole database	Data in instance will change during the updatation, deletion or addition of data.
Schema changed rarely	Instance changed frequently
Define the basic structure of database. i.e. how the data will be stored in the database.	It is a set of information stored in particular time.

Centralized DBMSs Architecture

- Gradually, DBMS systems started to exploit the available processing power at the user side, which led to client/server DBMS architectures.

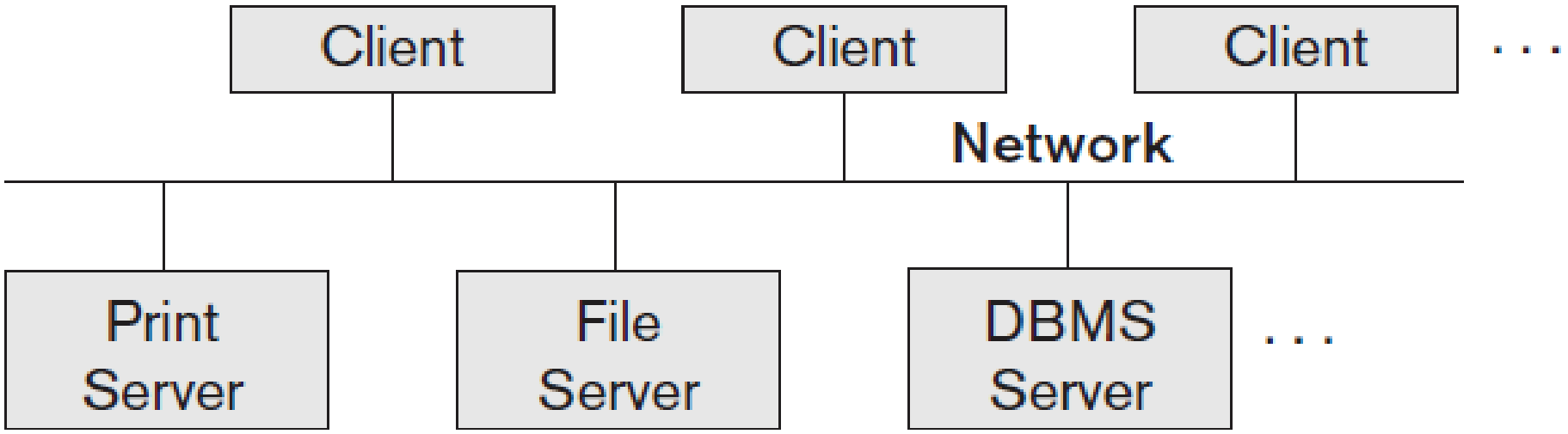
Client/Server Architectures

- The **client/server architecture** was developed to deal with computing environments in which a large number of
  - *PCs / workstations,*
  - *file servers,*
  - *printers,*
  - *database servers,*
  - *Web servers,*
  - *e-mail servers*, and other software and *equipment are connected via a network.*

Client/Server Architectures: Logical two-tier client/server architecture

- The idea is to define specialized servers with specific functionalities.
- For example,
- it is possible to connect a number of PCs or small workstations as clients to a file server that maintains the files of the client machines.
- printer server : All print requests by the clients are forwarded to this machine.
- Web servers / email server etc.

Client/Server Architectures



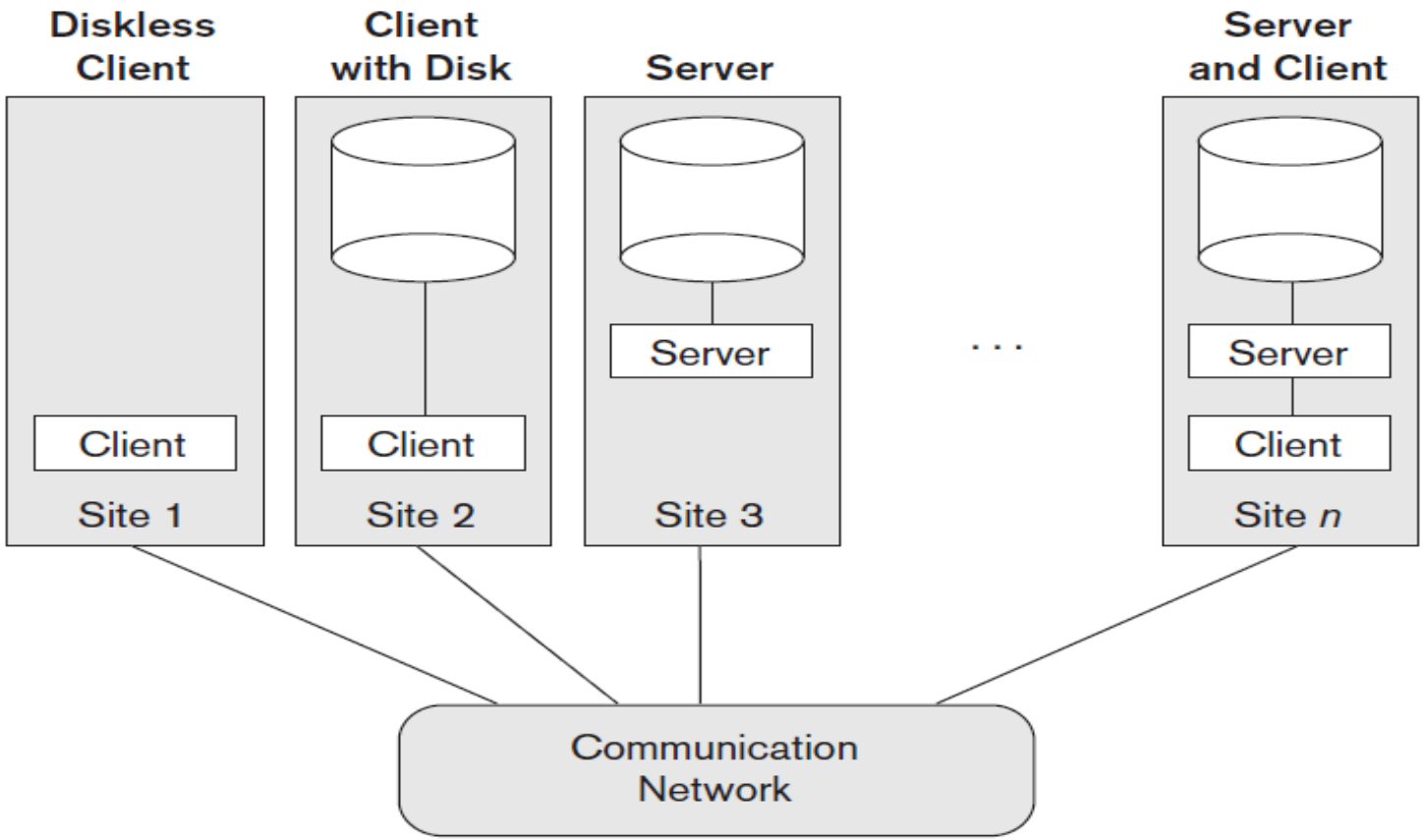
Client/Server Architectures

- **A client** is typically a user machine that provides user interface capabilities and local processing.
- **A server** is a system containing both hardware and software that can provide services to the client machines.

Client/Server Architectures: Two-Tier Client/Server Architectures for DBMSs

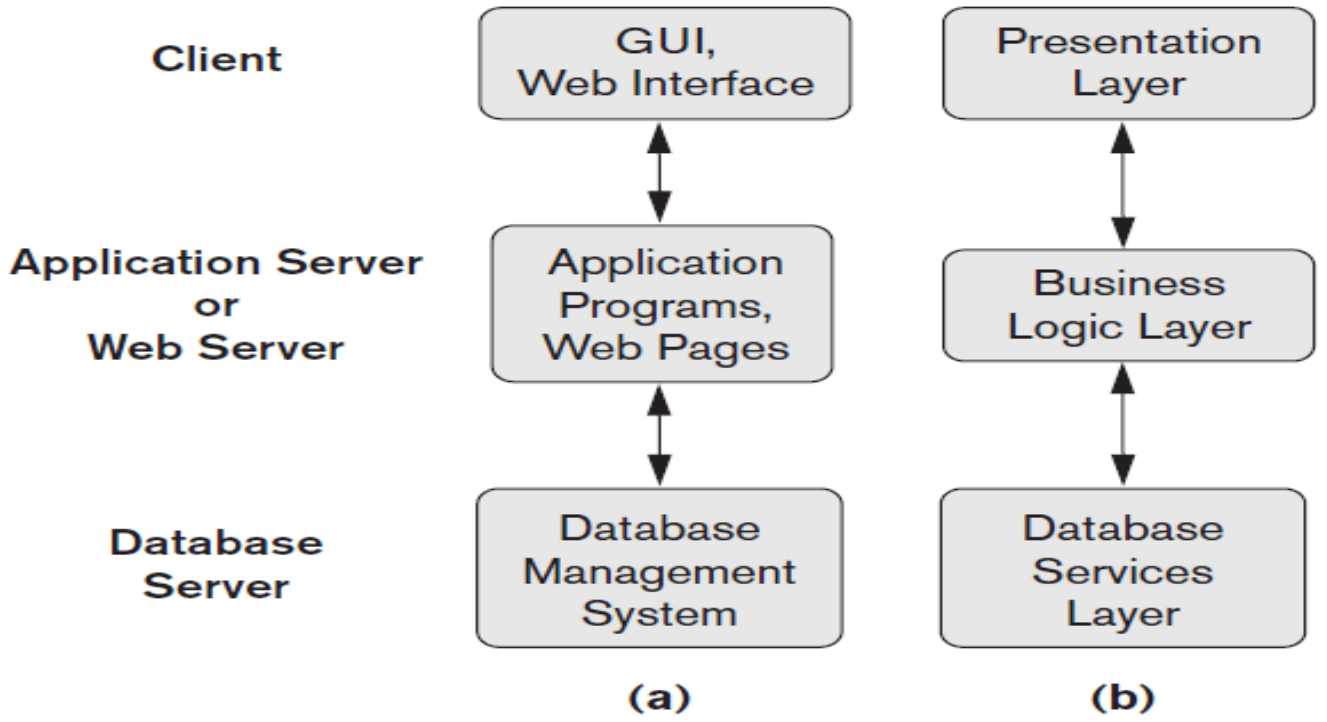
- In two-tier architectures, the software components are distributed over two systems: client and server.
- The advantages of this architecture are its simplicity and compatibility with existing systems.

Client/Server Architectures



Client/Server Architectures: Three-Tier and n-Tier Architectures for Web Applications

❖ Many Web applications use an architecture called the **three-tier architecture**, which adds an intermediate layer between the client and the database server, as shown in the following figure:



Client/Server Architectures

- The intermediate layer or **middle layer** is sometimes called the **application server** or **Web server**
- This server is running application programs and storing business rules (procedures or constraints) that are used to access data from the database server.
- **Three-tier Architecture Can Enhance Security:**
  - Database server only accessible via middle tier.
  - Clients cannot directly access database server.

Client/Server Architectures

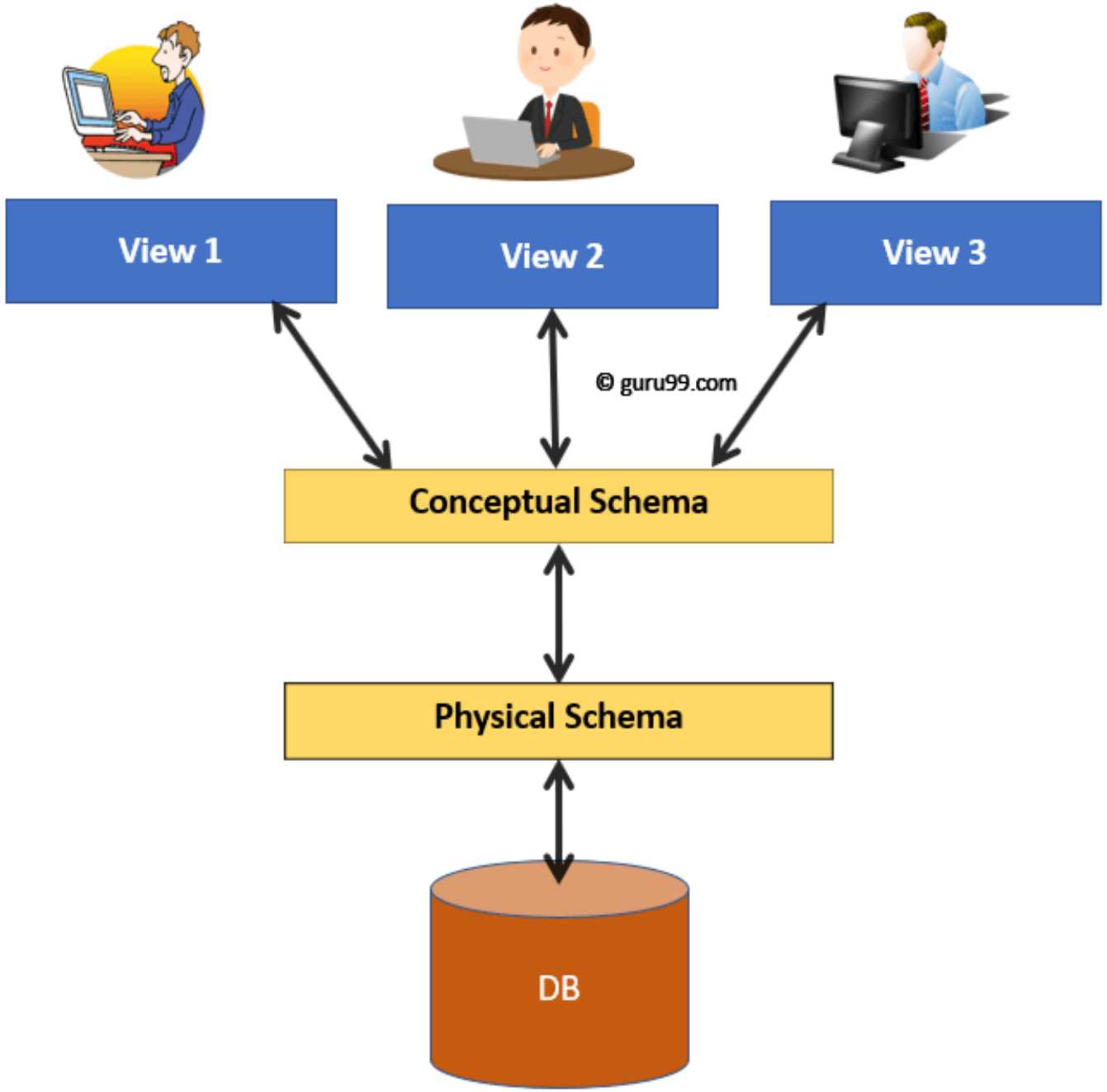
- **The presentation layer** displays information to the user.
- **The business logic layer** handles intermediate rules and constrains / validation before data is passed up to the user or down to the DBMS.
- **The Database Services Layer** : Which manage the actual database and send respond.

3 Levels ANSI SPARC Database System / Data Abstraction

External Layout View
End-Users

Conceptual Layer

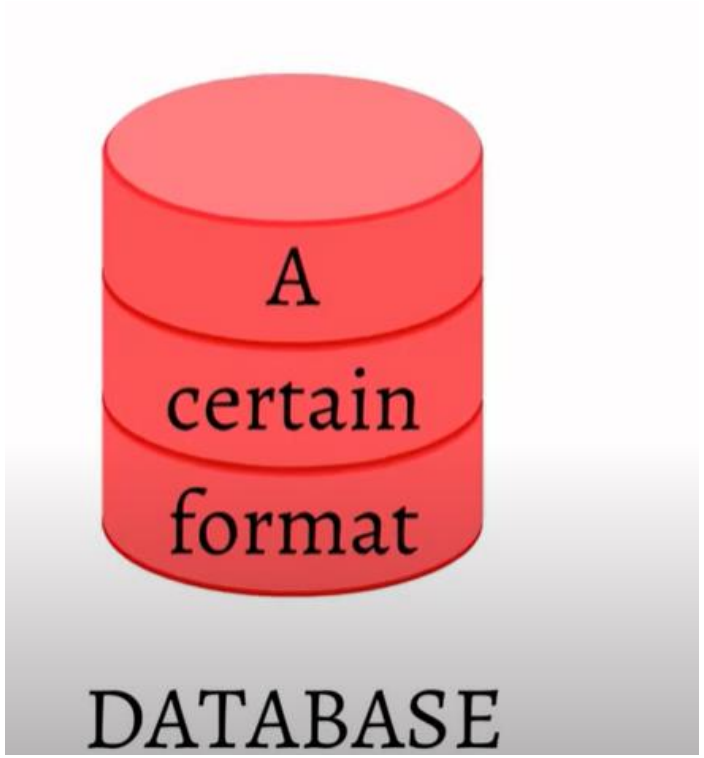
Physical Layer



Data Models

What Is Data Model?

- A **Data Model** Decides The Method Of Storing Data In Database.
- **Defines the logical structure of a database.**
- When The Data Is Stored In The Database It Needs To Be Stored In A Particular Format.
- So Data Model Decide The **Structure To Store Data**



What Is Data Model?

Decide required layouts

Architect will design house

Contractor implement design

Required attributes

Database designer design

DBA implement design using DBMS software

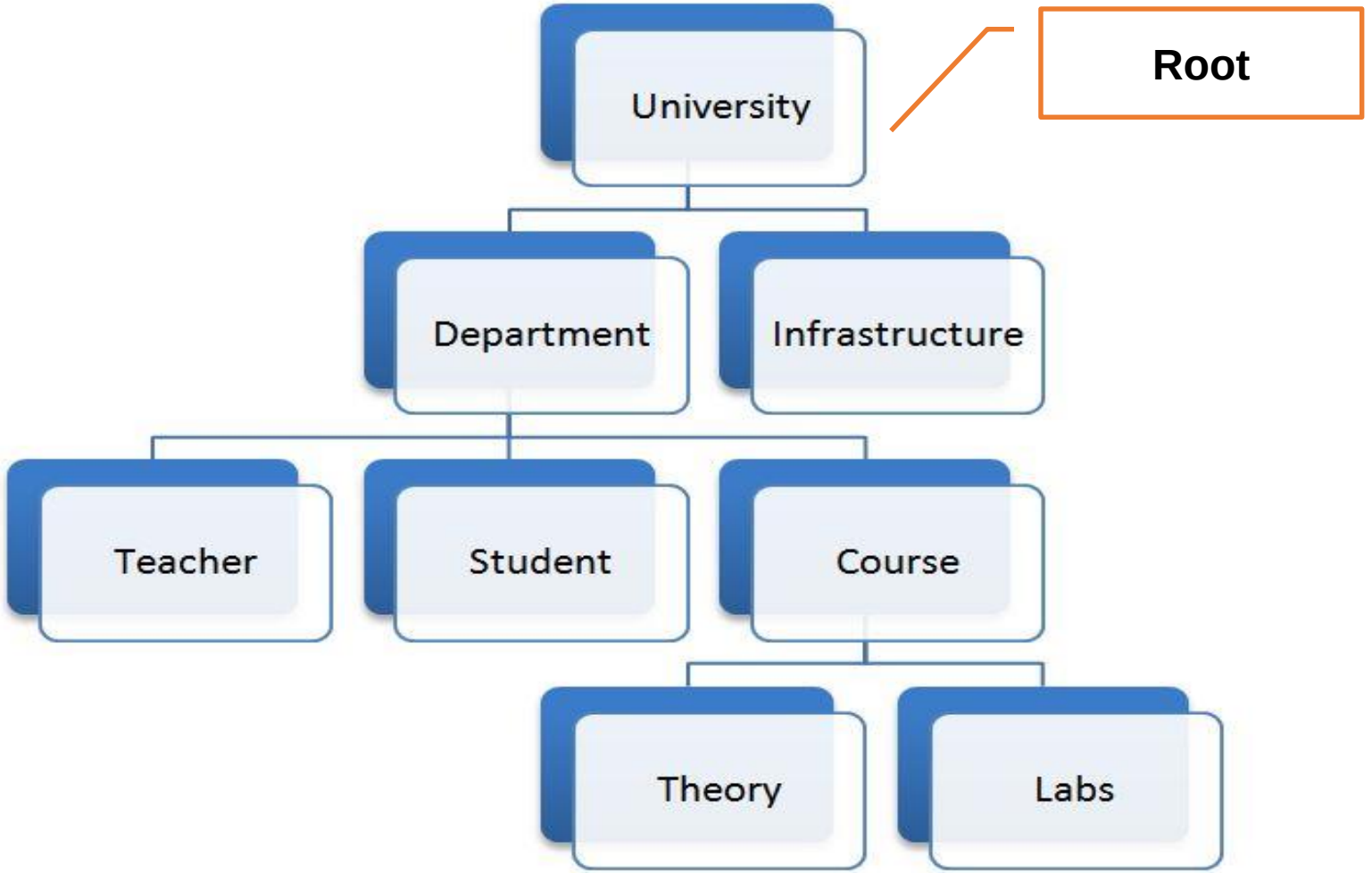
Types Of Data Models

- Hierarchical Model
- Network Model
- Relational Model
- Entity-Relationship Model

Hierarchical Model

- It was developed by IBM, in the 1960s.
- Used tree structure to represent data.
- The hierarchy starts from the Root data, and expands like a tree, adding child nodes to the parent nodes.
- A parent node contain one or more child node.
- Data are viewed as a collection of tables.
- Support one to one or one to many relationship.
- Upside down approach.

Hierarchical model



Advantages

- Simple & Easy to understand
- Data security
- Maintain data integrity

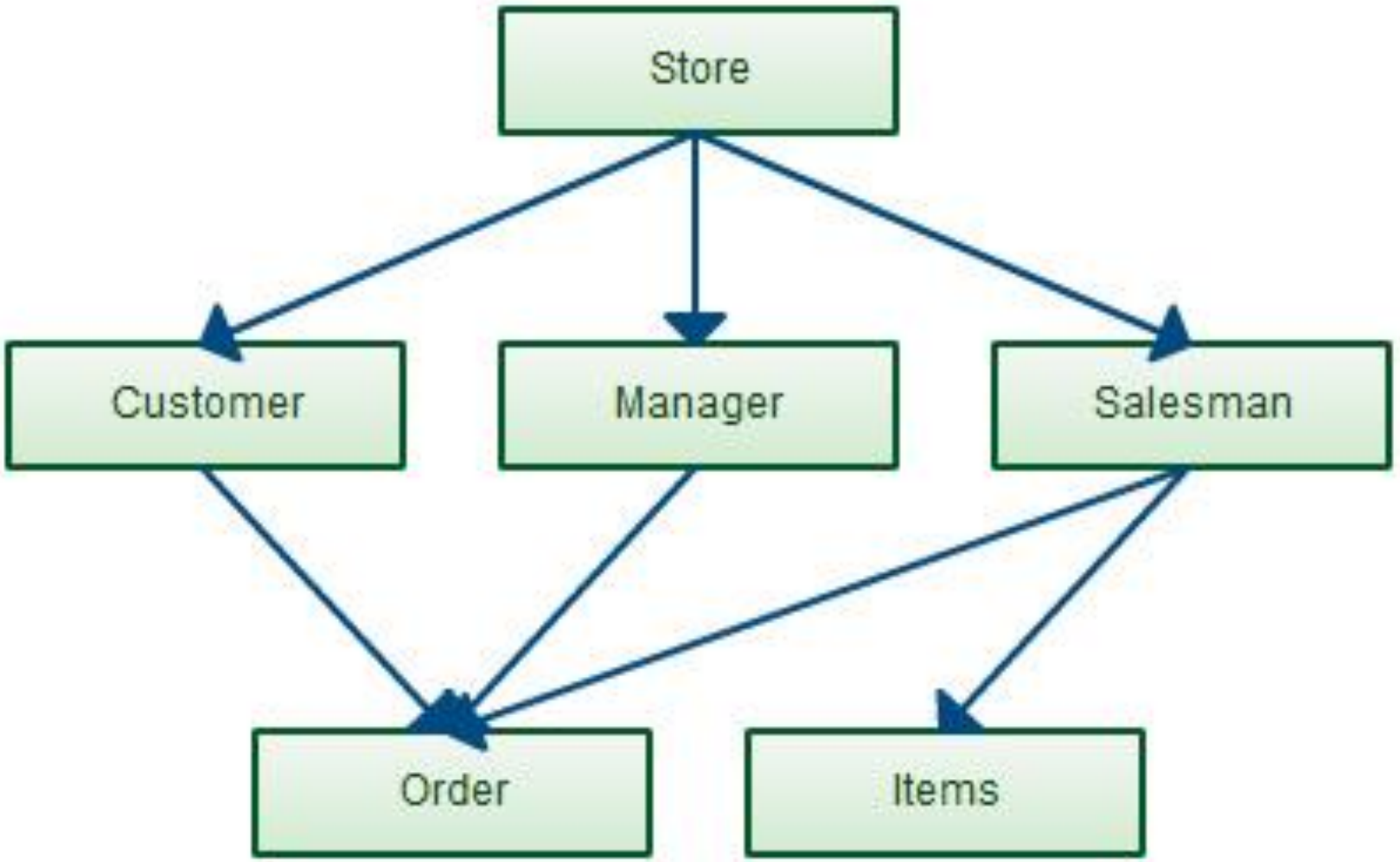
Disadvantages

- Top to down traversal approach
- Complex model
- One parent per child is allowed in hierarchical model.
- Does not support Many too many relationships.

Network Model

- This is an extension of the Hierarchical model.
- In this model data is organized more like a graph.
- It allowed to have more than one parent node.
- It is designed to represent objects and their relationships flexibly.
- Each set contains one owner or parent record as well as one or more child or member records.

Network Model Example



Advantages

- Conceptually simple and easy to design.
- Data security
- Maintain data integrity
- Support many too many relationships.
- Data access is easier and flexible
- Reduce data redundancy

Disadvantages

- System complexity
- Lack of structural independence

ER Model

- It is based on real-world entities & their relationships.
- Used to design a conceptual view of database.
- In terms of DBMS, an entity is a table or attribute of a table in database.
- By showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.
- relationships are created by dividing object of interest into entity and its characteristics into attributes.

ER Model Example

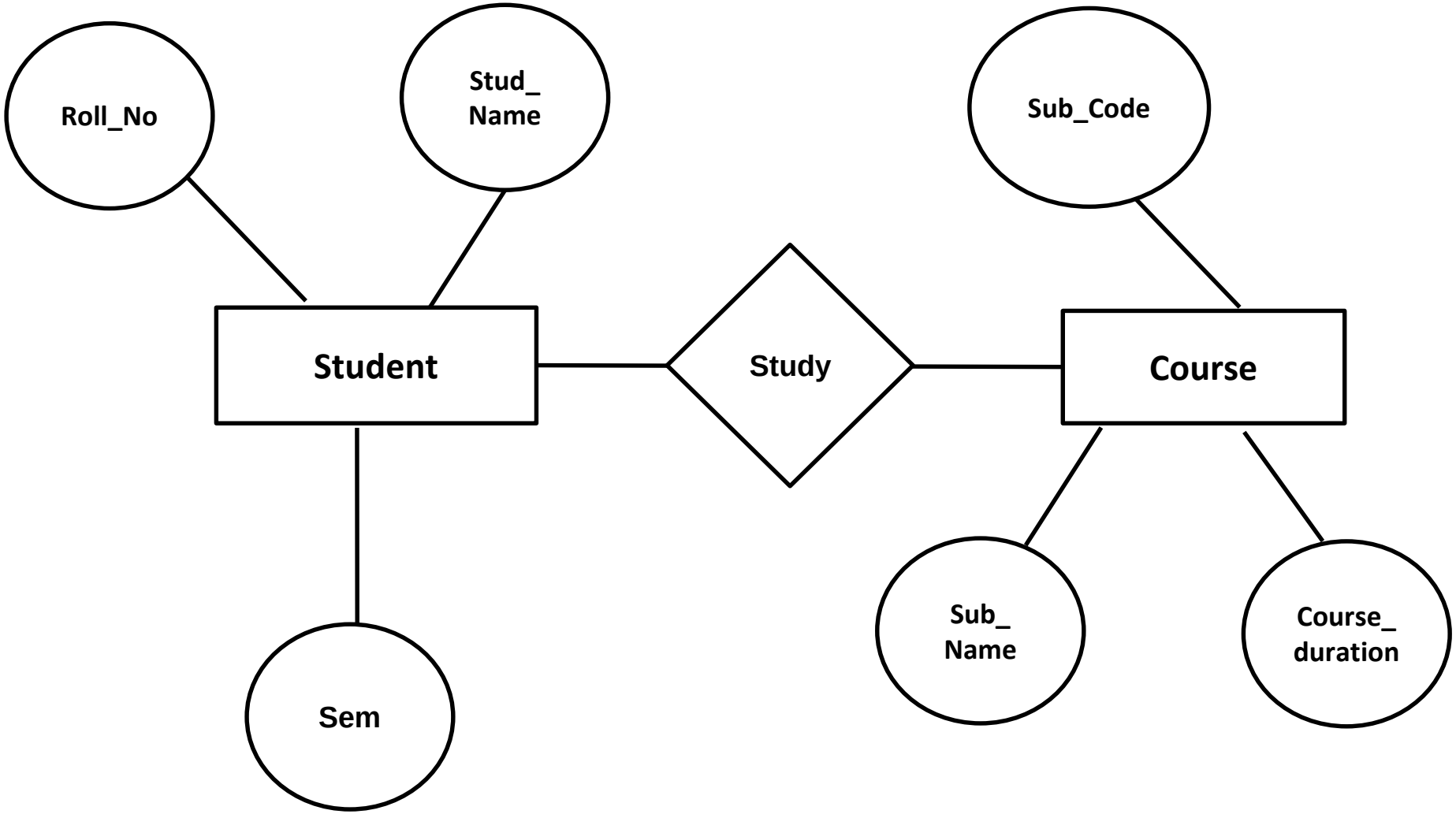
Student

Roll_No	Stud_Name	Sem
1	Ekta	1
2	Nihar	2
3	Mansi	3

Course

Sub_Code	Sub_Name	Course duration
CS01	Oracle	3 months
CS02	Python	6 months
CS03	Java	3 months
CS04	Asp .Net	3 months

ER model Example



Advantage

- Easy to understand
- Simple
- More specific to relational database modelling
- Good DBMS support
- Visual representation

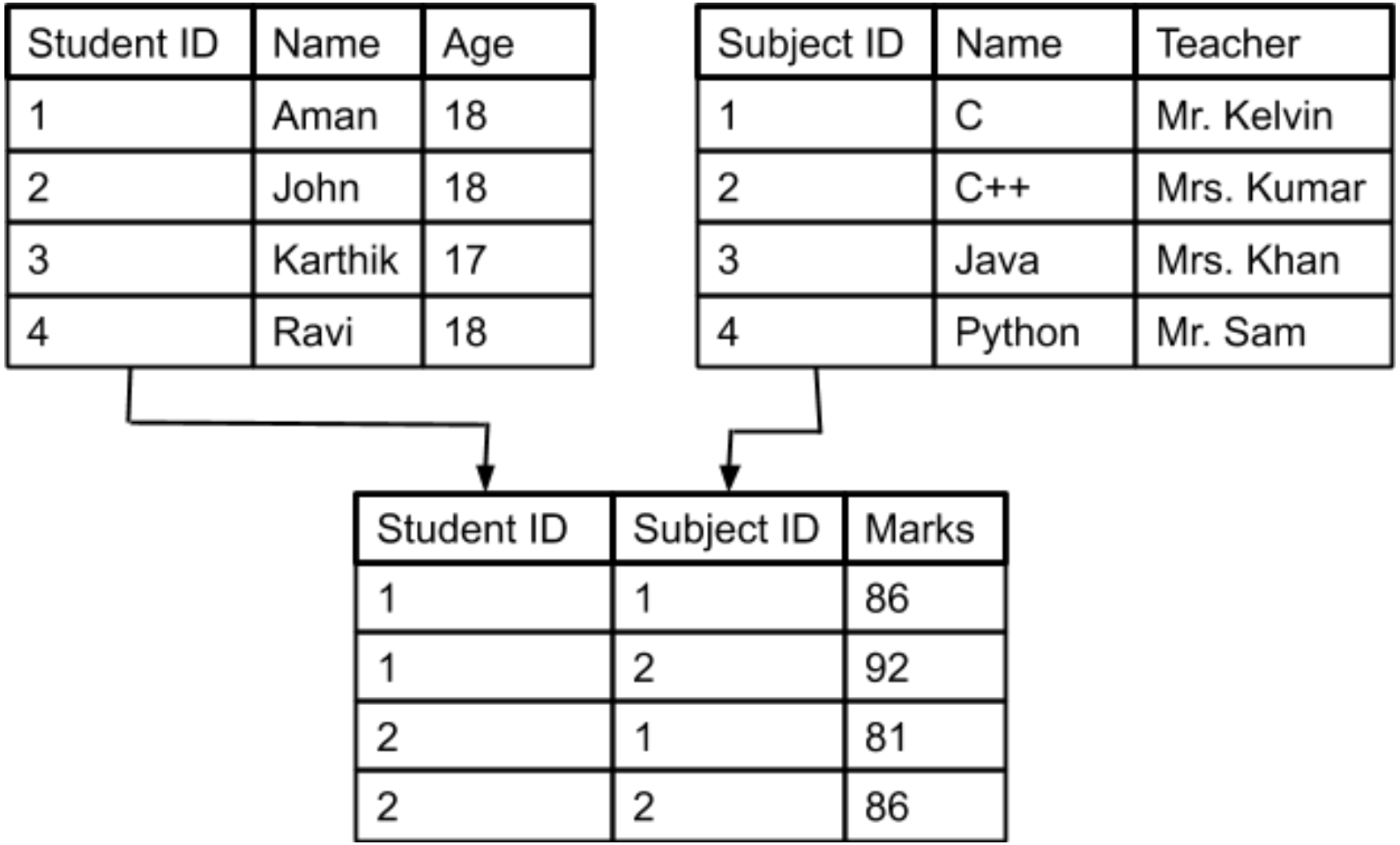
Disadvantages

- Limited expressiveness
- Can be ambiguous
- Mostly for relational database only
- Limited constraint representation
- No data manipulation language

Relational model

- This data model is introduced by C.F.Codd in 1970.
- Currently, it is considered as the most widely used data model.
- It represents the database as a collection of relations.
- A relation is nothing but a table of values.
- Every row in the table represents a collection of related data values.
- The table name and column names are helpful to interpret the meaning of values in each row.
- The data are represented as a set of relations.

Relational model



Student ID	Name	Age
1	Aman	18
2	John	18
3	Karthik	17
4	Ravi	18

Subject ID	Name	Teacher
1	C	Mr. Kelvin
2	C++	Mrs. Kumar
3	Java	Mrs. Khan
4	Python	Mr. Sam

Student ID	Subject ID	Marks
1	1	86
1	2	92
2	1	81
2	2	86

Advantages:

- Structural independence
- Conceptual simplicity
- Design , implementation , maintenance and usage ease
- Query capability
- Limits redundancy
- Flexible
- Offers better data integrity

Disadvantages:

- Performance Issue
- Complex to understand when there is more number of tables.