Database Management Systems (4^{th} Semester)

21CIC43

B.Tech (CSE)

Module 1.1
Databases and Database Users

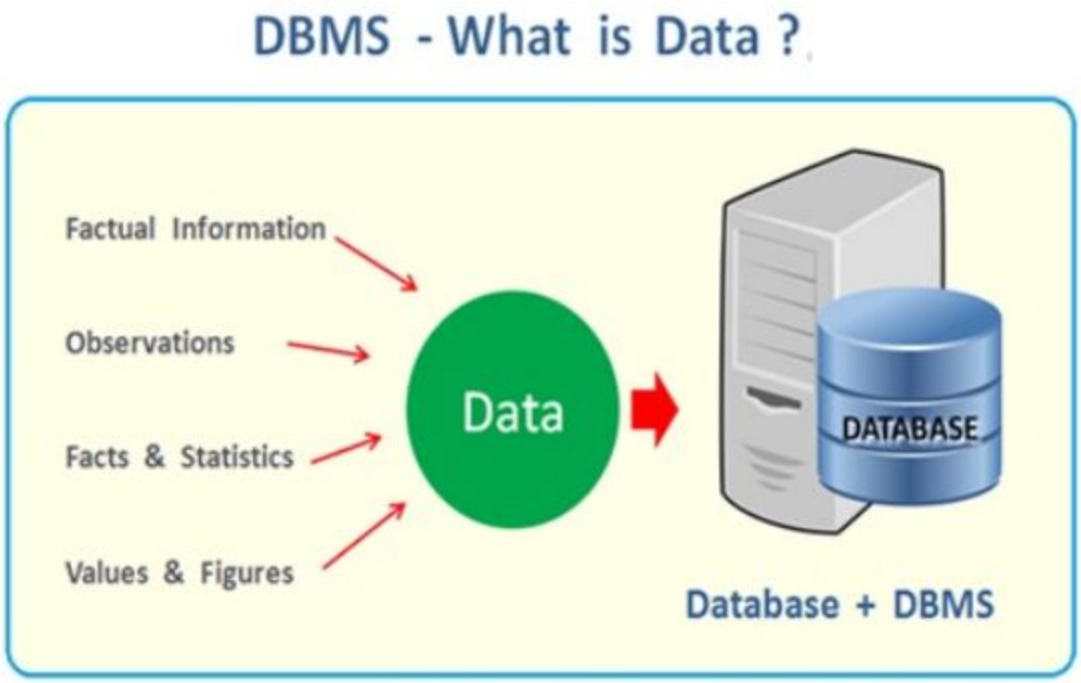
Reference: Fundamentals of Database Systems, RamezElmasri and Shamkant B. Navathe. 7th Edition, 2017, Pearson

OUTLINE

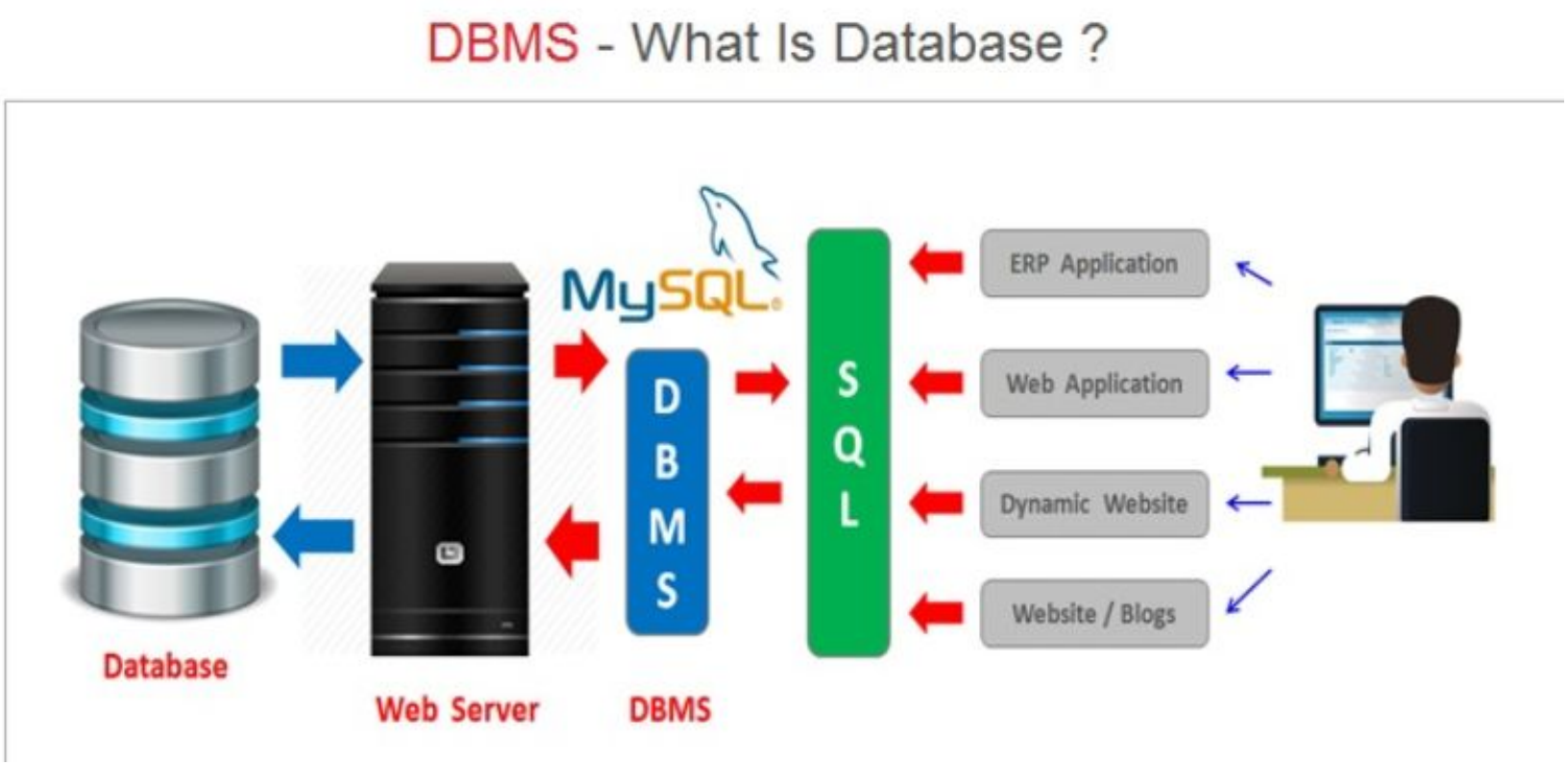
- Databases and Database Users
  - Introduction
  - Characteristics of the Database Approach
  - Actors on the scene
  - Workers behind the scene
  - Advantages of Using the DBMS Approach
  - A Brief History of Database Applications

Introduction

- **Data:**
  - Known facts that can be recorded and have an implicit meaning.
- **Database:**
  - A collection of related data.
- **Mini-world:**
  - Some part of the real world about which data is stored in a database. For example, student grades at a university.



INTRODUCTION



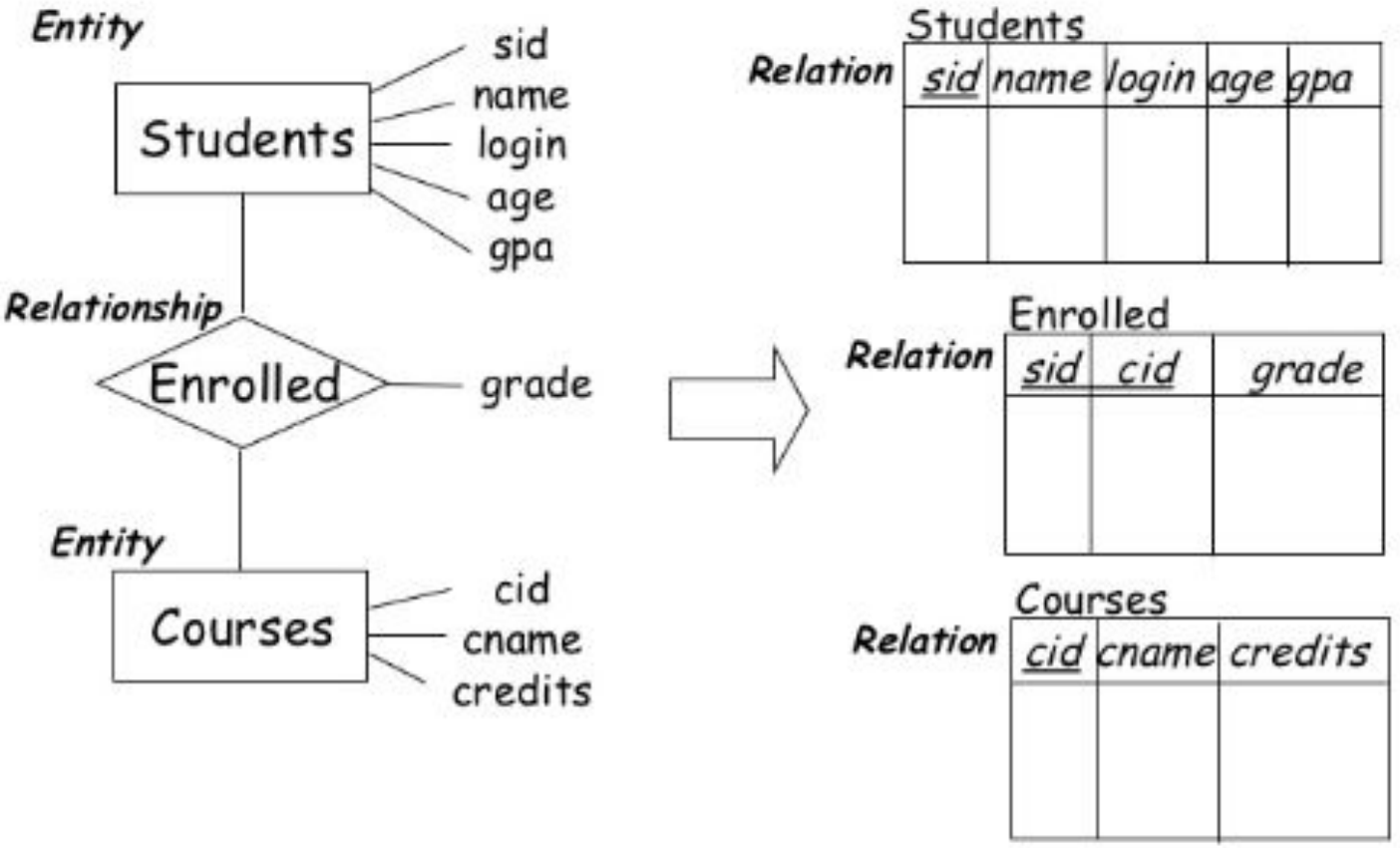
INTRODUCTION

DATA	
RECORDABLE	FACTS
UNPROCESSED	OBSERVATIONS
UNRELATED	FIGURES
DISORGANIZED	STATISTICS

DATABASE
ORGANIZED COLLECTION OF DATA
INTERRELATED DATA
STORE , MAINTAIN , RETRIVE DATA
LOGICAL AND PHYSICAL STRUCTURE

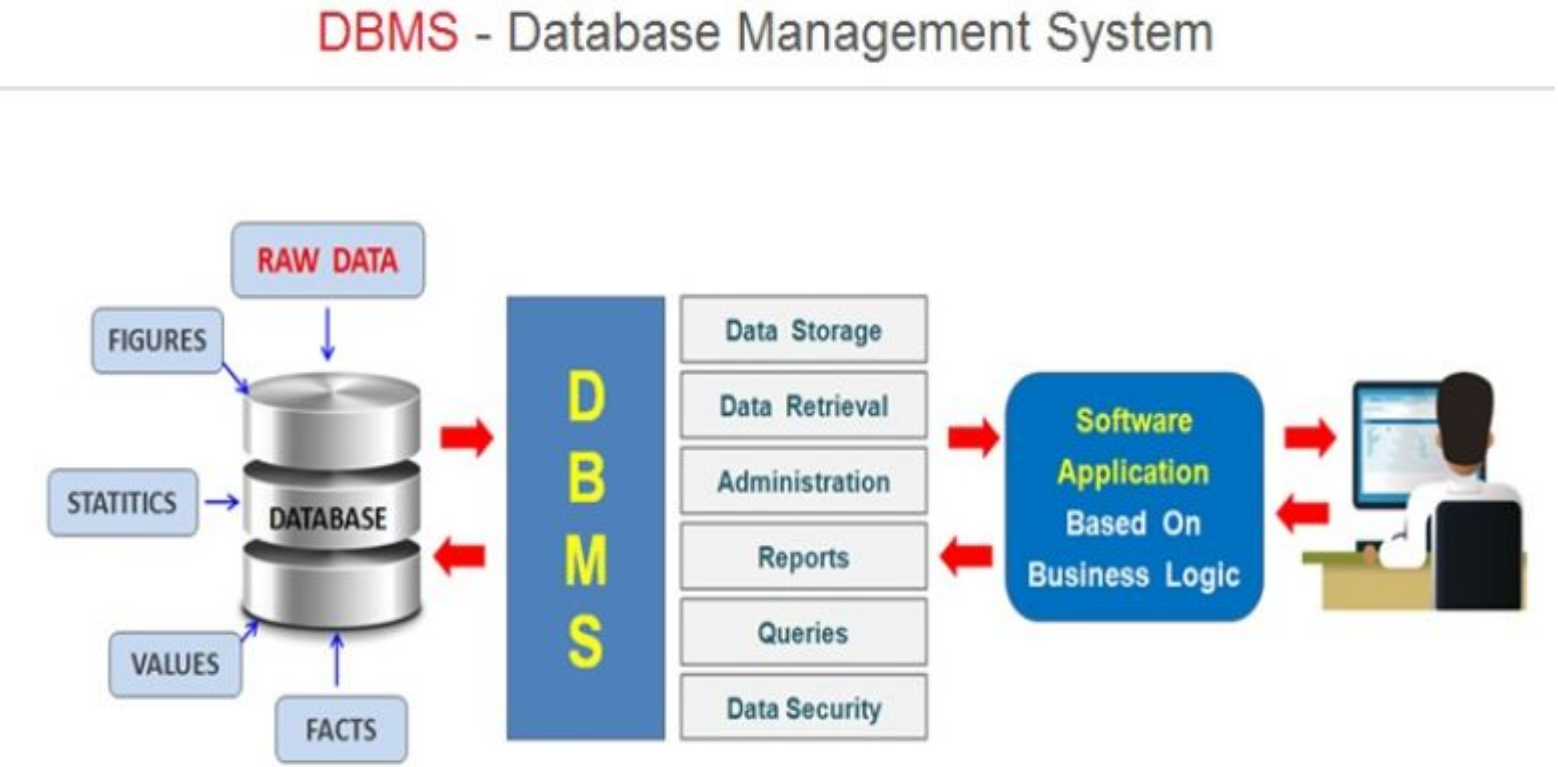
Introduction

University Database



Introduction

- **Database Management System (DBMS):**
  - A software package/ system to facilitate the creation and maintenance of a computerized database.



Introduction

- **Different DBMS Software**

Introduction

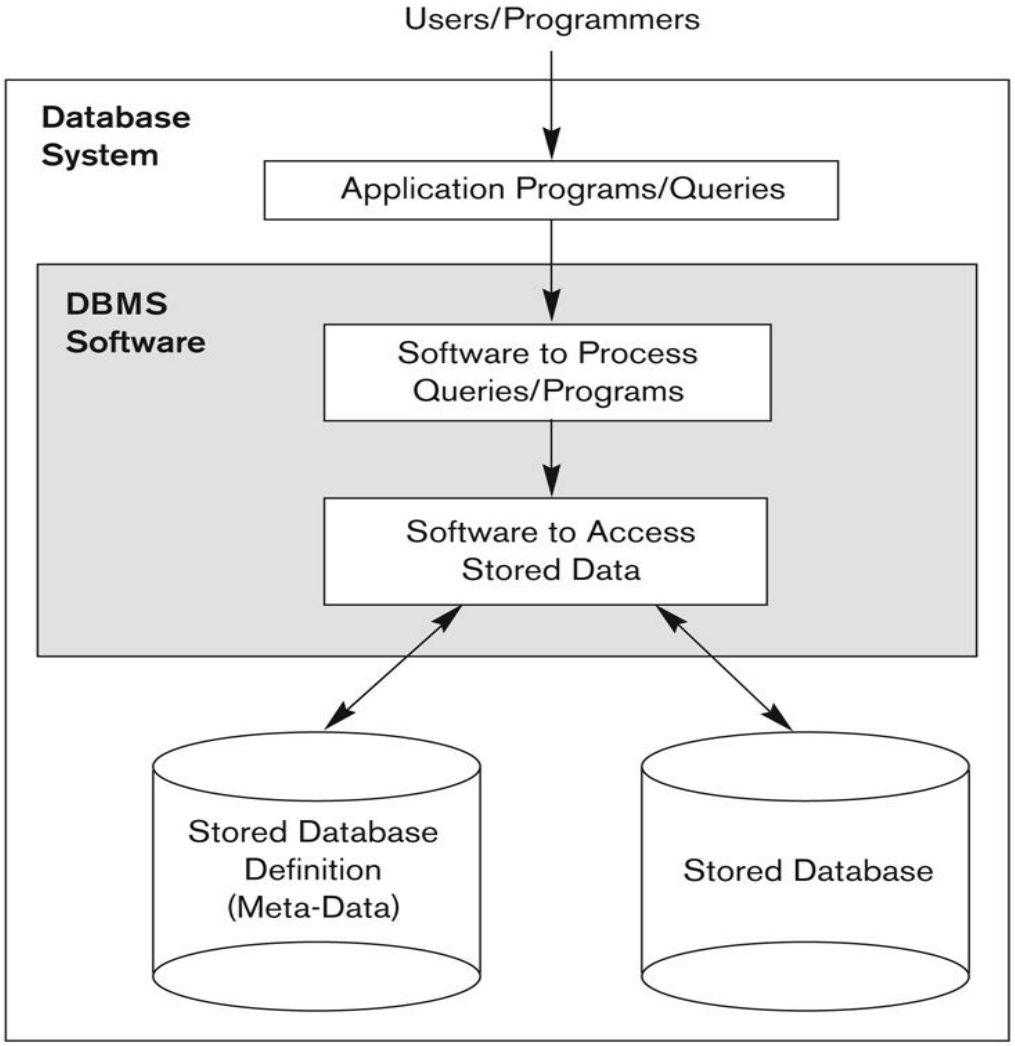


Figure 1.1
A simplified database system environment.

Introduction

Mini-world for the example:

Part of a UNIVERSITY environment.

Some mini-world entities:

STUDENTs
COURSEs
SECTIONs (of COURSEs)
(academic) DEPARTMENTs
INSTRUCTORs

Some mini-world *relationships*:

SECTIONs *are of specific* COURSEs
STUDENTs *take* SECTIONs
COURSEs *have prerequisite* COURSEs
INSTRUCTORs *teach* SECTIONs
COURSEs *are offered by* DEPARTMENTs
STUDENTs *major in* DEPARTMENTs

Introduction

Example of a simplified Database

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Main Characteristics of the Database Approach (versus file processing approach)

- **Self-describing nature of a database system:**
  - A DBMS **catalog** stores the description of a particular database (e.g. data structures, types, and constraints)
  - The description is called **meta-data**.
  - This allows the DBMS software to work with different database applications.

- **Insulation between programs and data and Data Abstraction:**
  - Structure of the data files is stored in the DBMS catalog separately from the access programs, this property is also Called as **program-data independence**.
  - The characteristic that allows program-data independence and program-operation independence is called **data abstraction**. A DBMS provides users with a **conceptual representation** of data that does not include many of the details of how the data is stored or how the operations are implemented

Main Characteristics of the Database Approach (versus file processing approach)

Database Catalog

RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
....	....	.....
....	....	.....
....	....	.....
Prerequisite_number	XXXXNNNN	PREREQUISITE

Main Characteristics of the Database Approach (versus file processing approach)

- **Support of multiple views of the data:**
  - Each user may see a different view of the database, which describes **only** the data of interest to that user.
- **Sharing of data and multi-user transaction processing:**
  - Allowing a set of **concurrent users** to retrieve from and to update the database.
  - *Concurrency control* within the DBMS guarantees that each **transaction** is correctly executed or aborted
  - *Recovery* subsystem ensures each completed transaction has its effect permanently recorded in the database
  - **OLTP** (Online Transaction Processing) is a major part of database applications. This allows hundreds of concurrent transactions to execute per second.

Actors on the scene

- **Database administrators:**
  - Responsible for authorizing access to the database, for coordinating and monitoring its use, acquiring software and hardware resources, controlling its use and monitoring efficiency of operations.
- **Database Designers:**
  - Responsible to define the content, the structure, the constraints, and functions or transactions against the database. They must communicate with the end-users and understand their needs.

Actors on the scene

- **End-users:** They use the data for queries, reports and some of them update the database content. End-users can be categorized into:
  - **Casual**: access database occasionally when needed
  - **Naive** or Parametric: they make up a large section of the end-user population.
    - They use previously well-defined functions in the form of "canned transactions" against the database.
    - Examples are bank-tellers or reservation clerks who do this activity for an entire shift of operations.
  - **Sophisticated:**
    - These include business analysts, scientists, engineers, others thoroughly familiar with the system capabilities.
    - Many use tools in the form of software packages that work closely with the stored database.
  - **Stand-alone:**
    - Mostly maintain personal databases using ready-to-use packaged applications.
    - An example is a tax program user that creates its own internal database.

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.*

- **DBMS system designers and implementers** design and implement the DBMS modules and interfaces as a software package.
- **Tool developers** design and implement **tools**—the software packages that facilitate database modeling and design, database system design, and improved performance.
- **Operators and maintenance personnel** (system administration personnel) are responsible for the actual running and maintenance of the hardware and software environment for the database system.

Advantages of Using the Database Approach

- **Controlling redundancy.**

  This **redundancy** in storing the same data multiple times leads to several problems.

  First, there is the need to perform a single logical update—such as entering data on a new student—multiple times: once for each file where student data is recorded. This leads to *duplication of effort.*

  Second, *storage space is wasted* when the same

  data is stored repeatedly, and this problem may be serious for large databases.

  Third, files that represent the same data may become *inconsistent*

  Data normalization technique in DBMS ensures

  consistency and saves storage space

Advantages of Using the Database Approach

- Restricting unauthorized access to data.

  When multiple users share a large database, it is likely that most users will not be authorized to access all information in the database. For example, financial data such as salaries and bonuses is often considered confidential, and only authorized persons are allowed to access such data. DBMS should provide a **security and authorization subsystem**

- Providing persistent storage for program Objects

  Databases can be used to provide **persistent storage** for program objects and data structures. This is one of the main reasons for **object-oriented database systems**

Advantages of Using the Database Approach

- **Providing Storage Structures and Search Techniques for Efficient Quercy Processing**

  Database systems must provide capabilities for *efficiently executing queries and updates*. Because the database is typically stored on disk, the DBMS must provide specialized data structures and search techniques to speed up disk search for the desired records. Auxiliary files called **indexes** are often used for this purpose.

- **Providing Backup and Recovery**

  A DBMS must provide facilities for recovering from hardware or software failures. The **backup and recovery subsystem** of the DBMS is responsible for recovery.

Advantages of Using the Database Approach

- **Providing Multiple User Interfaces**

  Because many types of users with varying levels of technical knowledge use a database, a DBMS should provide a variety of user interfaces. These include apps for mobile users, query languages for casual users, programming language interfaces for application programmers, forms and command codes for parametric users.

- **Representing Complex Relationships among Data**

  A database may include numerous varieties of data that are interrelated in many ways. A DBMS must have the capability to represent a variety of complex relationships among the data, to define new relationships as they arise, and to retrieve and update related data easily and efficiently.

Advantages of Using the Database Approach

- **Enforcing Integrity Constraints**

  Most database applications have certain **integrity constraints** that must hold for the data. A DBMS should provide capabilities for defining and enforcing these constraints. The simplest type of integrity constraint involves specifying a data type for each data item.

- **Permitting Inferencing and Actions Using Rules and Triggers**

  Some database systems provide capabilities for defining *deduction rules* for *inferencing* new information from the stored database facts. Such systems are called **deductive database systems**.

  For example, there may be complex rules in the miniworld application for determining when a student is on probation. These can be specified d*eclaratively* as **rules**, which when compiled and maintained by the DBMS can determine all students on probation.

Summary

- Definitions of Different Terminologies
- Characteristics and Advantages of Database Systems over Traditional File Systems
- People on the scene and behind the scene
- History of Database Systems