Database Systems Course Outline

Objective

The world is now driven by data …

We will learn how to use and manage data

We will cover fundamental concepts of Databases with an emphasis on modeling, designing and implementation of Database systems.

© Can Stock Photo

Course Information

- **Instructor: Zareen Alamgir**
  - Email: zareen.alamgir@nu.edu.pk
  - Office Hours
    - Tuesday 1:00-2:00pm
    - Thursday 1:00-2:00pm

- Teaching Assistant
  - Ali Mustafa CS-C
  - M. Ansar Butt CS-D

Course Information

- Course information and updates will be posted on **Google ClassRoom**
  - Assignments
  - Lectures
  - Books
  - Tentative schedule
  - News and announcements

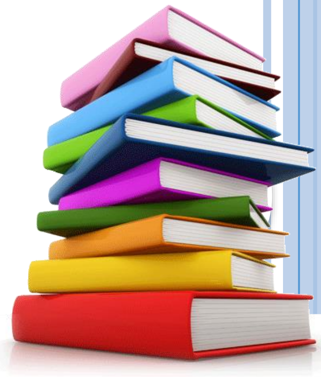
Attendance is Mandatory

Will send invite for google classroom today !!!

Books

- **Textbook:**
  - *Fundamentals of Database Systems* (7th Edition), Ramez Elmasri

- **Reference Books**
  - *Database Systems: The Complete Book*, Hector Garcia-Molina, Jeffrey Ullman, Jennifer Widom
  - *Database Management Systems*, Raghu Ramakrishnan



Pre-Requisites

- The students should have good background in
  - Programming
  - Data Structures

Grading Scheme

Midterms	30%
Quizzes	10%
Assignments / Class Participation	15%
Final	45%

- Minimum eligibility to pass this course is to get **50% marks in both Semester work and Final.**
- Academic integrity is expected of all the students.
    - *Plagiarism or cheating in any assessment will result in at least an* ***F*** *grade in the course*

The Project

- Important component
  - Part of the Lab

- 3 Phases.
  - You build a database application on your own.
  - The domain of the application will be given.
  - The application will have a simple web interface (asp.net).

Course Outline

Topics	Text
Database System Concepts and Architecture	Ch 1, 2
Relational Data Model and Relational Database Constraints	Ch 3
The Relational Algebra	Ch 6
The Database Language SQL	Ch 8, 9
Transaction Processing Concepts	Ch 21
Functional Dependencies and Normalization for Relational Databases	Ch 10
ER Modeling and Enhanced ER Modeling	Ch 3, 4
Relational Database Design by ER to Relational Mapping	Ch 7

Introduction:

Databases and Database Systems

Acknowledgement

Content obtained from many sources: Ramez Elmari Book and Raghu Ramakrishnan *Book*

What is Database ?

- An organized collection of related data.
- Example of Database
  - telephone book
  - T.V. Guide
  - papers in your filing cabinet
  - files on your computer hard drive

Some Representative Applications

- Banking
- Airlines
- Universities
- Hospitals

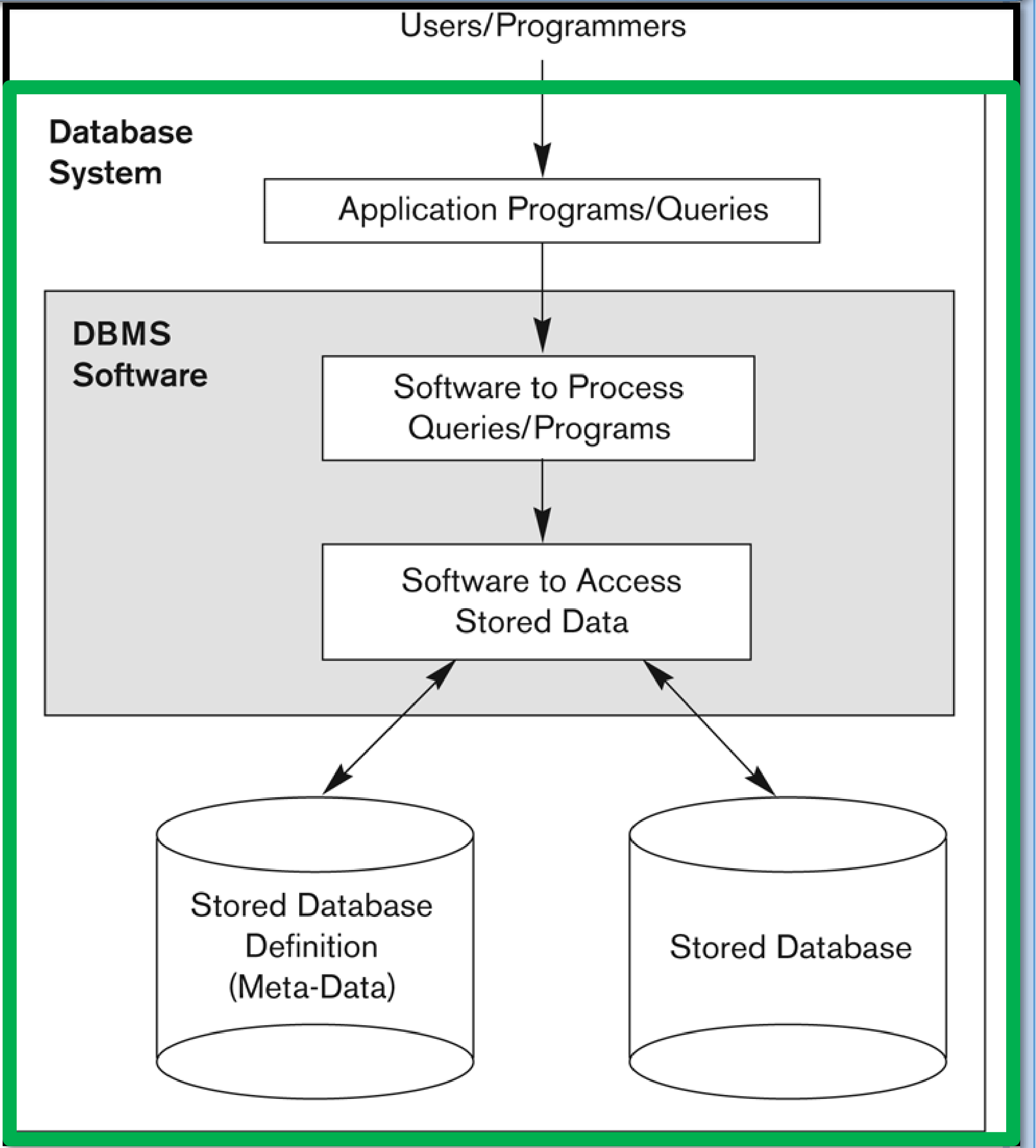
Simplified Database System environment

Program-data independence

<u>**Meta-Data**</u>

the description of DB stored in DBMS catalog

<u>(data types,</u> <u>structure</u> & <u>constraints</u>)

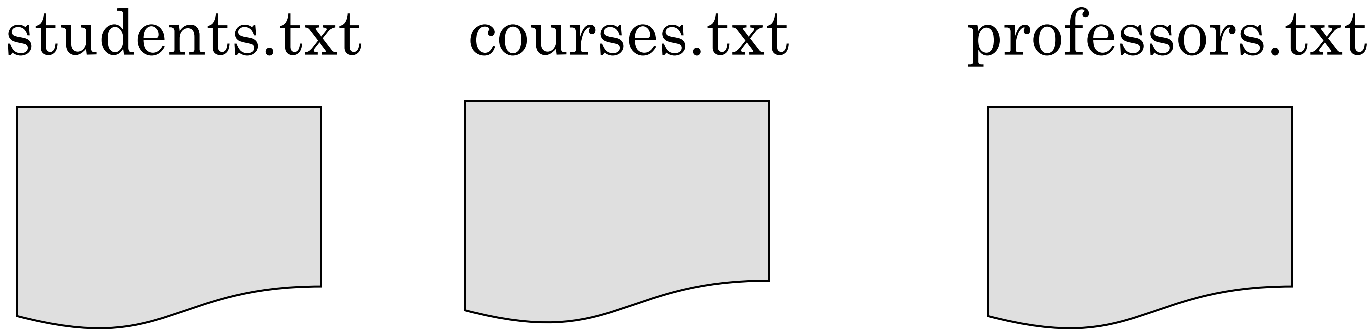

Users/Programmers
Database System
Application Programs/Queries
DBMS Software
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition (Meta-Data)
Stored Database

Database System

- **Database Management System (DBMS):**
  - A software system to store and manage computerized databases.
- **Database System:**
  - A collection of interrelated data and a set of programs to access those data
  - It includes DBMS and the data.
  - Sometimes, the applications are also included.

Can we do it without a DBMS ?

Yes !!! You have been doing it using files:



students.txt courses.txt professors.txt

Now write C++ programs to implement specific tasks

Drawbacks of File system

Data redundancy and inconsistency

- Duplication of information in different files

Difficulty in accessing data

- Need to write a new program to carry out each new task
- Multiple files and formats

Integrity problems

- Integrity constraints (e.g. account balance > 0) become part of program code
- Hard to add new constraints or change existing one

Atomicity of updates

- e.g. transfer of funds from one account to another should either complete OR not happen at all

Drawbacks of file system

Concurrent access by multiple users

- Uncontrolled concurrent accesses can lead to inconsistencies
- e.g. two people reading a balance and updating it at the same time

Security problems

Database systems offer solutions to all the above problems

Database is just a big C++ program written by someone else that accesses and updates those files for you

Situations where DB is not desirable

Simple, well-defined database that are not expected to change at all

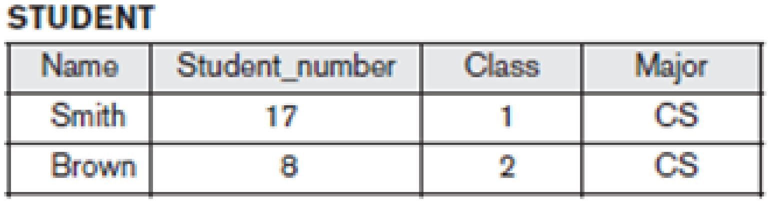
Stringent, real-time requirements that may not be met because of DBMS overhead

Embedded systems with limited storage capacity

No multiple-user access to data

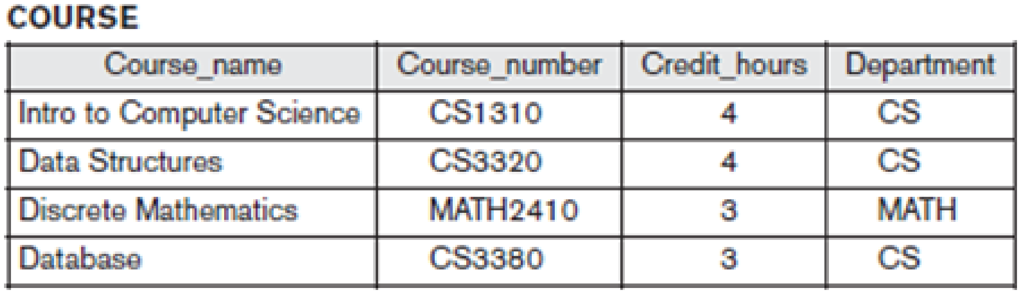
Example of a simple database

Mini-world: UNIVERSITY



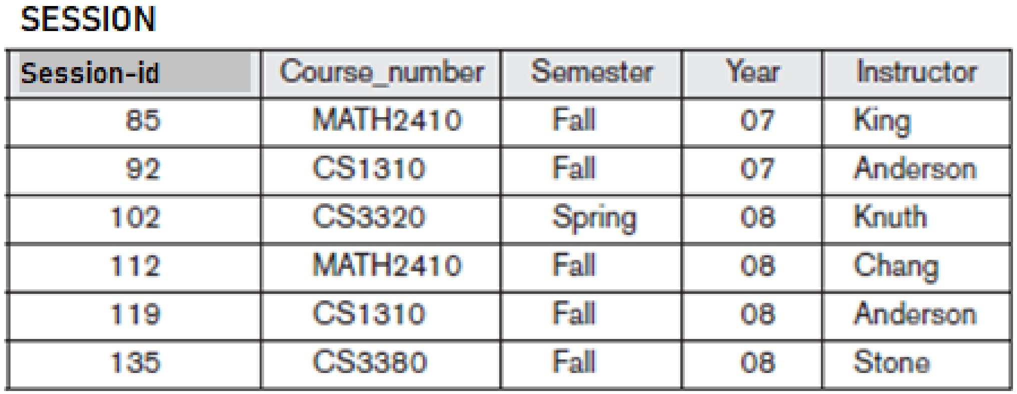
STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS



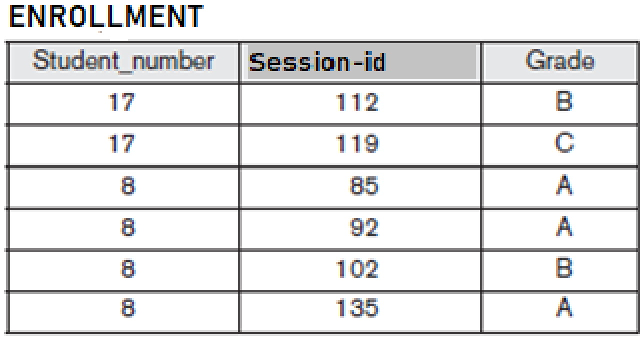
COURSE

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



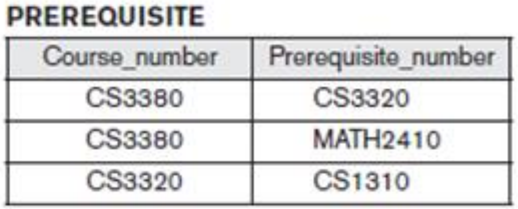
SESSION

Session-id	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone



ENROLLMENT

Student_number	Session-id	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

Entities:

- Students
- Courses
- Offering (Session of Courses)

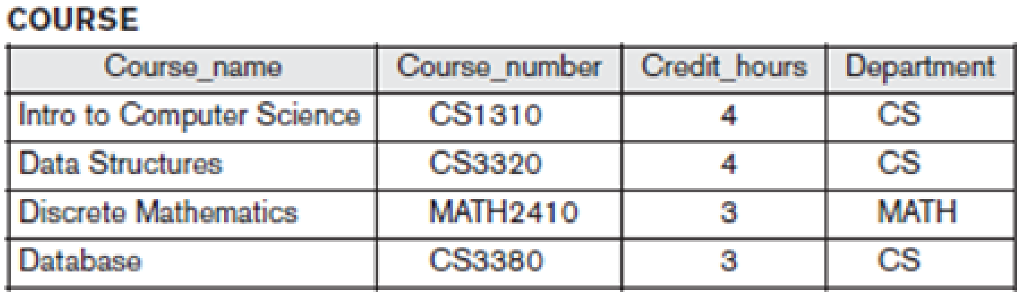
Relationships:

- Session *are of specific* Courses
- Students *enroll in a session of a course*
- Courses *have prerequisite* Courses

DATA MODEL

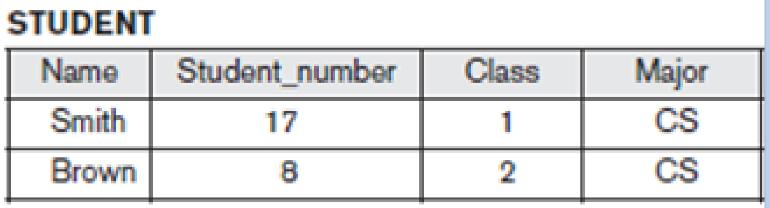
Data Model is a set of concepts to describe the *structure* of a database and achieve data abstraction.

- structure includes data types, relationships and constraints.



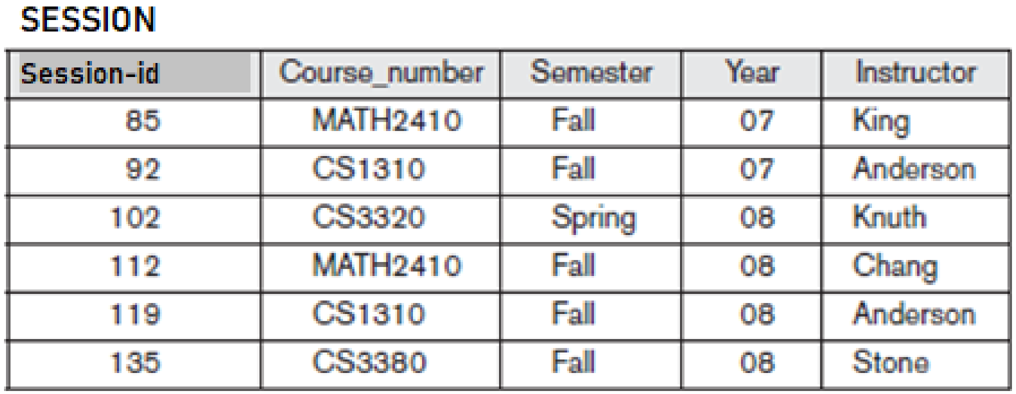
COURSE

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



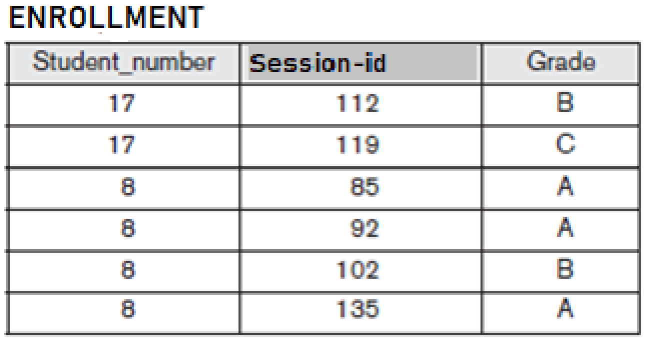
STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS



SESSION

Session-id	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone



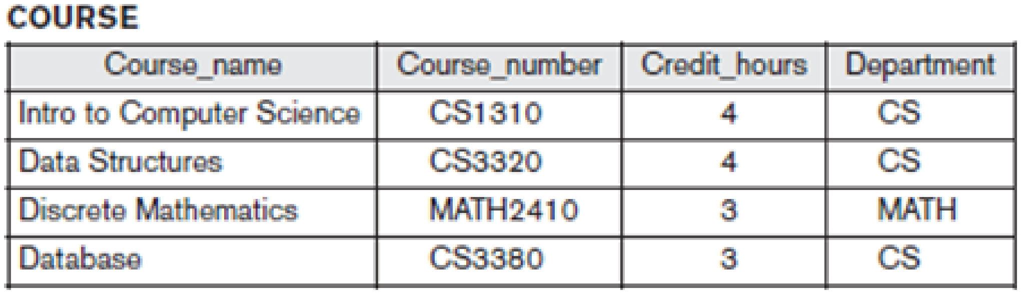
ENROLLMENT

Student_number	Session-id	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

Relational DATA MODEL

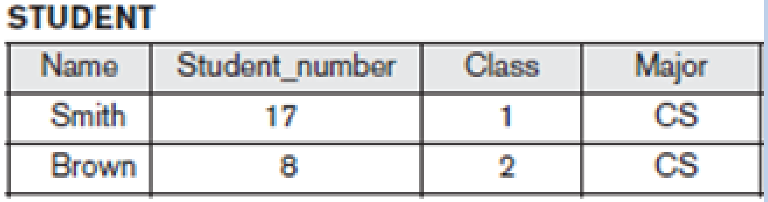
Relational data model is the most widely used model today

- **Main Concept:** the *relation* (table)
- **Data Model Operations**
  - Basic operations (insert, delete, modify and retrieve)
  - User-defined operations (compute GPA)



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



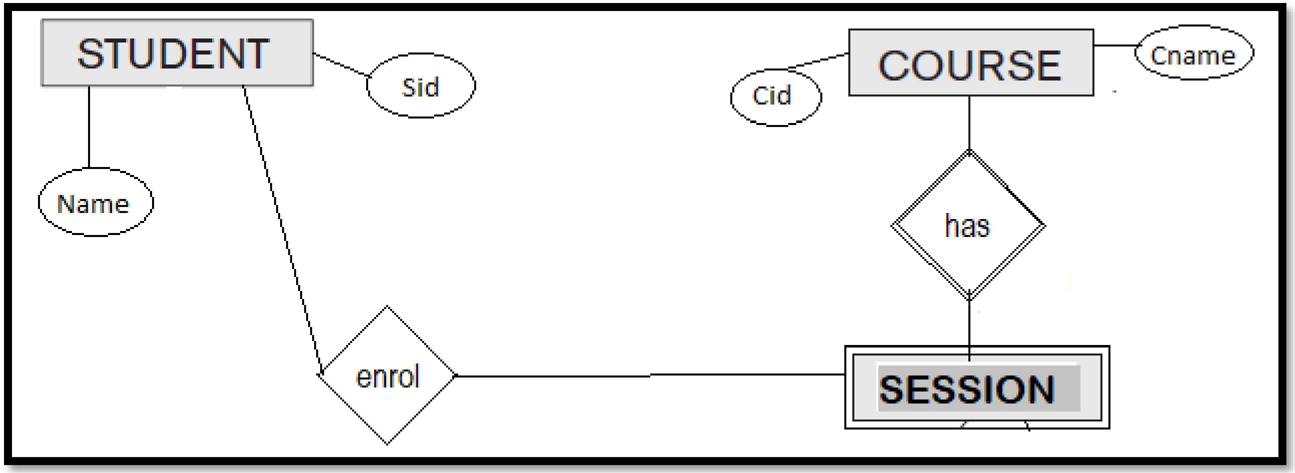
STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

Categories of Data Models

Conceptual (high-level)data model

ER Model (Entity Relationship model)



Categories of Data Models

- **Conceptual (high-level, semantic) data models**
  - Provide concepts that are close to the way many users perceive data. Also called ***entity-based*** data models.

- **Physical (low-level, internal) data models**
  - Provide concepts that describe details of how data is stored in the computer.

- **Implementation (representational) data models**
  - Provide concepts that fall between the above two.
  - Balance user views with some computer storage details.
  - It includes **relational data model.**

Database Users

Database Administrators

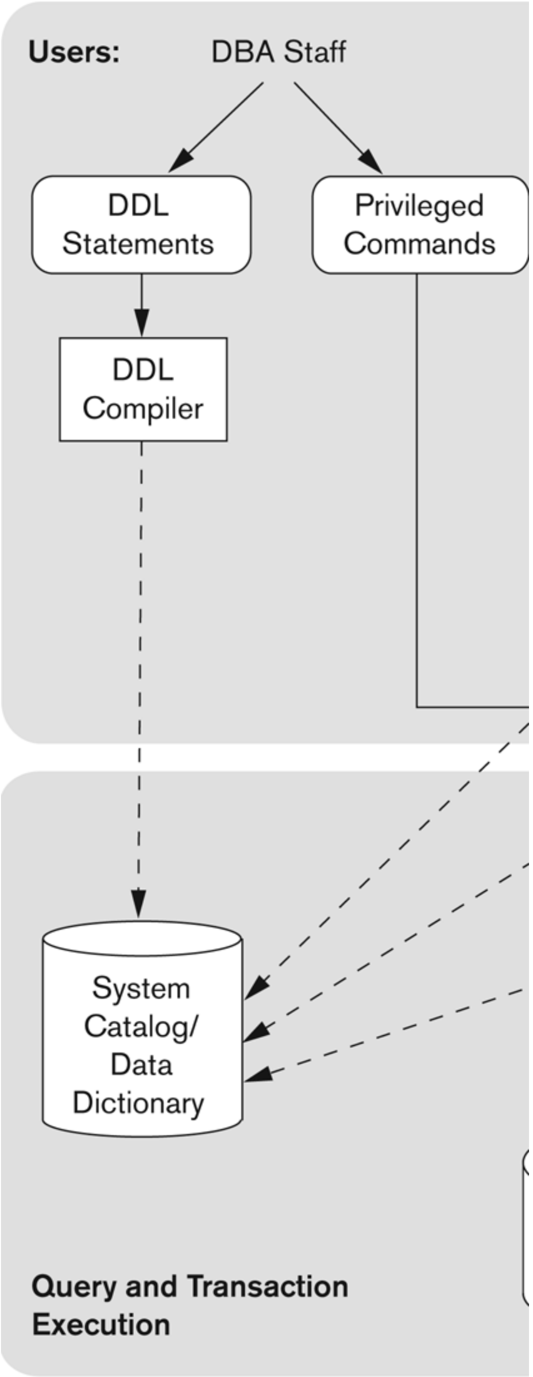
- Responsible for authorizing access
- Coordinating and monitoring its use
- Acquiring software and hardware resources
- Monitoring efficiency of operations.

Database Designers/ Programmers

- Responsible to define the content, the structure, the constraints, and transactions against the database.

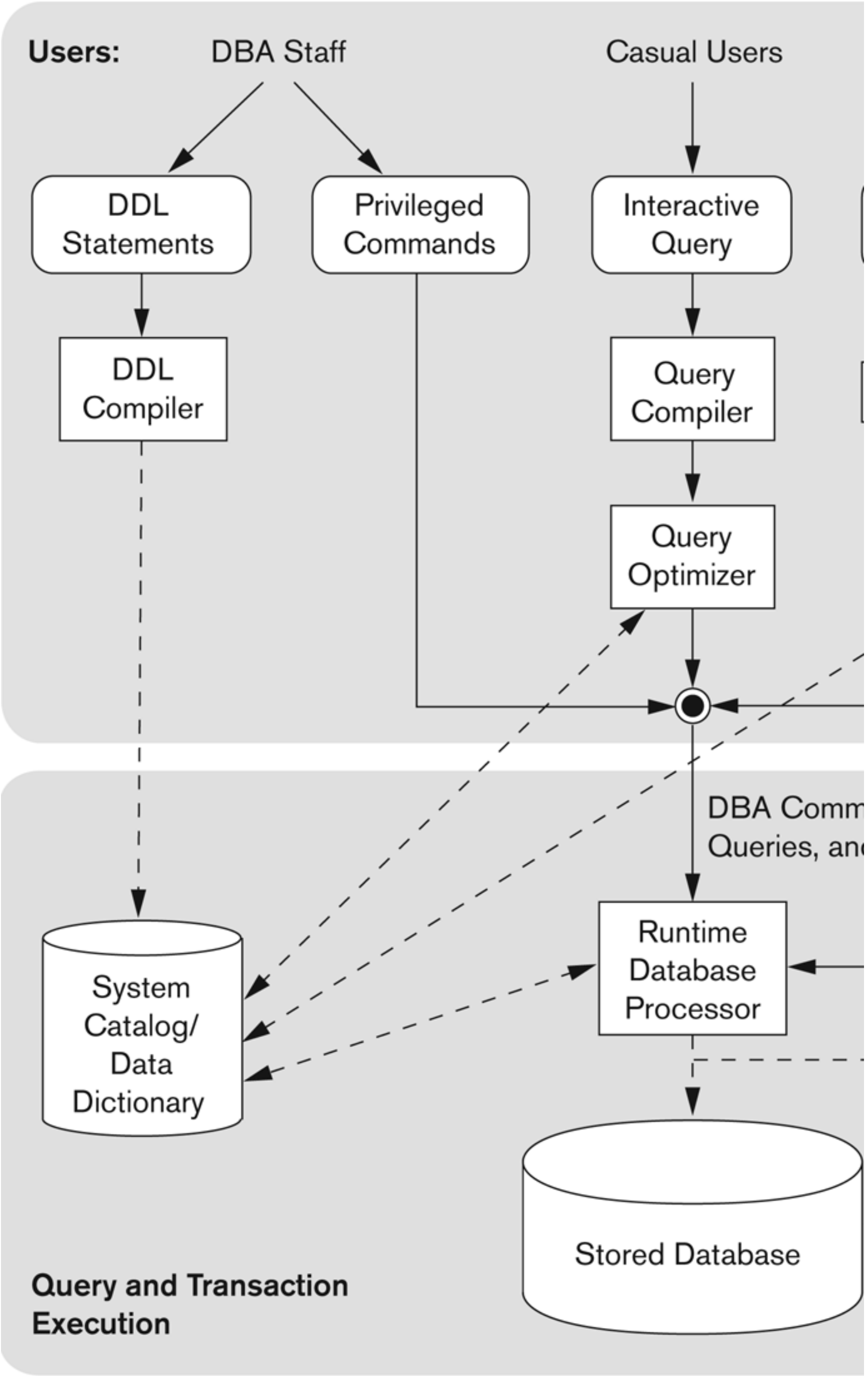
End-users

- They use the data for queries, reports and some of them update the database content.

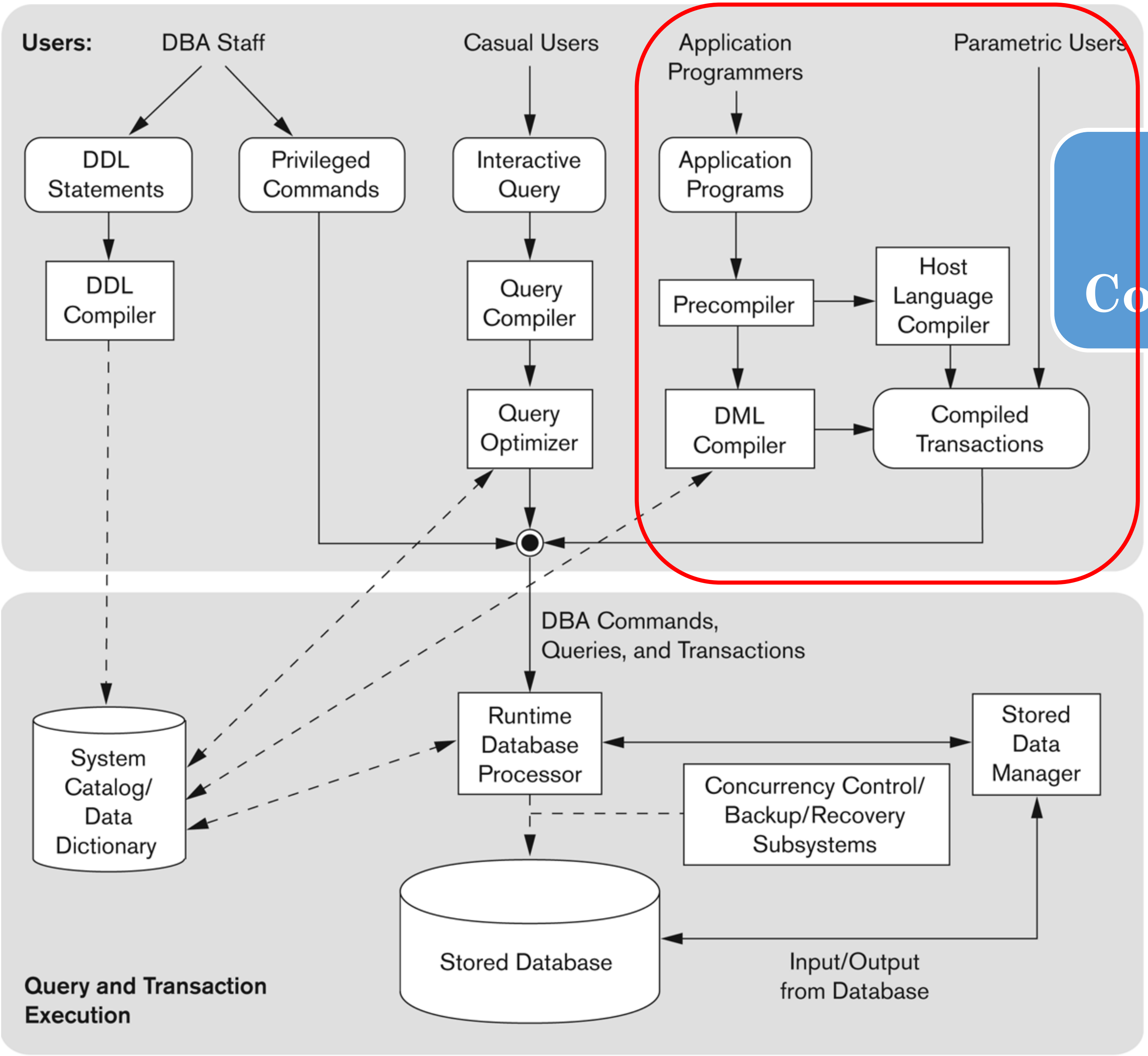

Users:
DBA Staff
DDL Statements
Privileged Commands
DDL Compiler
System Catalog/ Data Dictionary
Query and Transaction Execution

Typical DBMS Components

Typical DBMS Components



Typical DBMS Components



Classification of DBMSs

Users

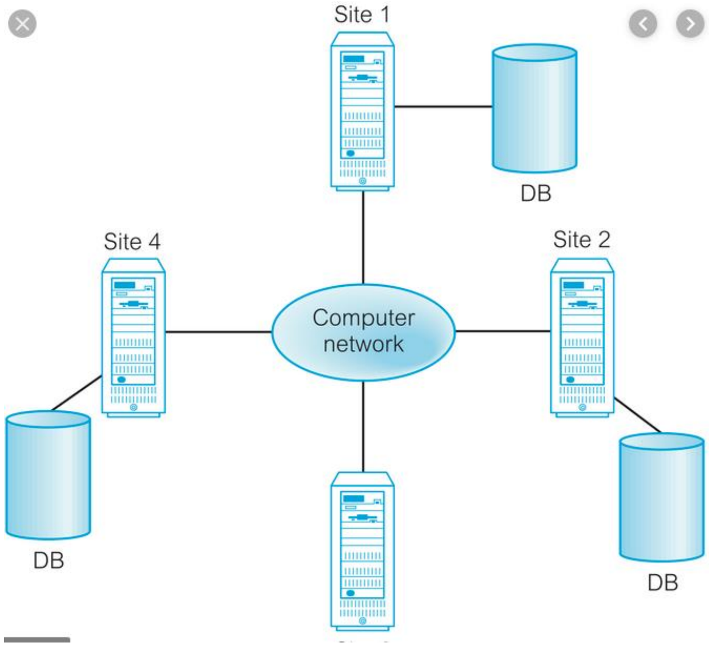
- Single-user (typically used with personal computers)
- Multi-user (most DBMSs).

No of sites

- Centralized (data is stored at a single computer site)
- Distributed (database and DBMS software distributed over many sites)

Distributed DBMSs (DDBMSs)

- Distributed Database Systems can have database and DBMS software distributed over many sites
  - Homogeneous DDBMS
  - Heterogeneous DDBMS



DBMS Architectures

- **Centralized DBMS**
  - Combines everything into single system: DBMS software, hardware, application programs, and user interface.
  - Users can connect through a remote terminal.

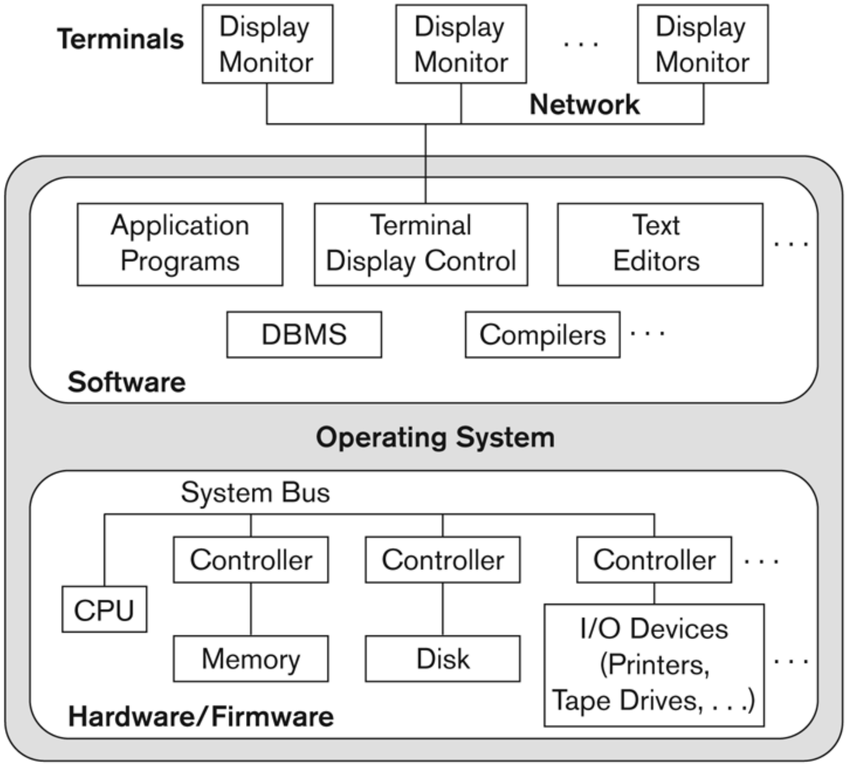
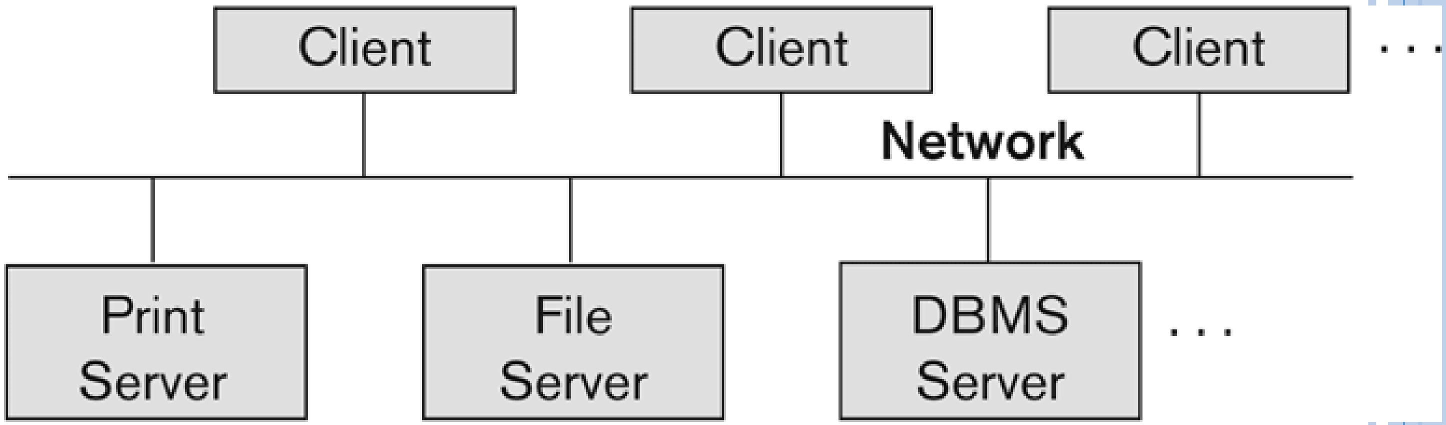


Figure 2.4
A physical centralized architecture.

Basic 2-tier Client-Server Architectures

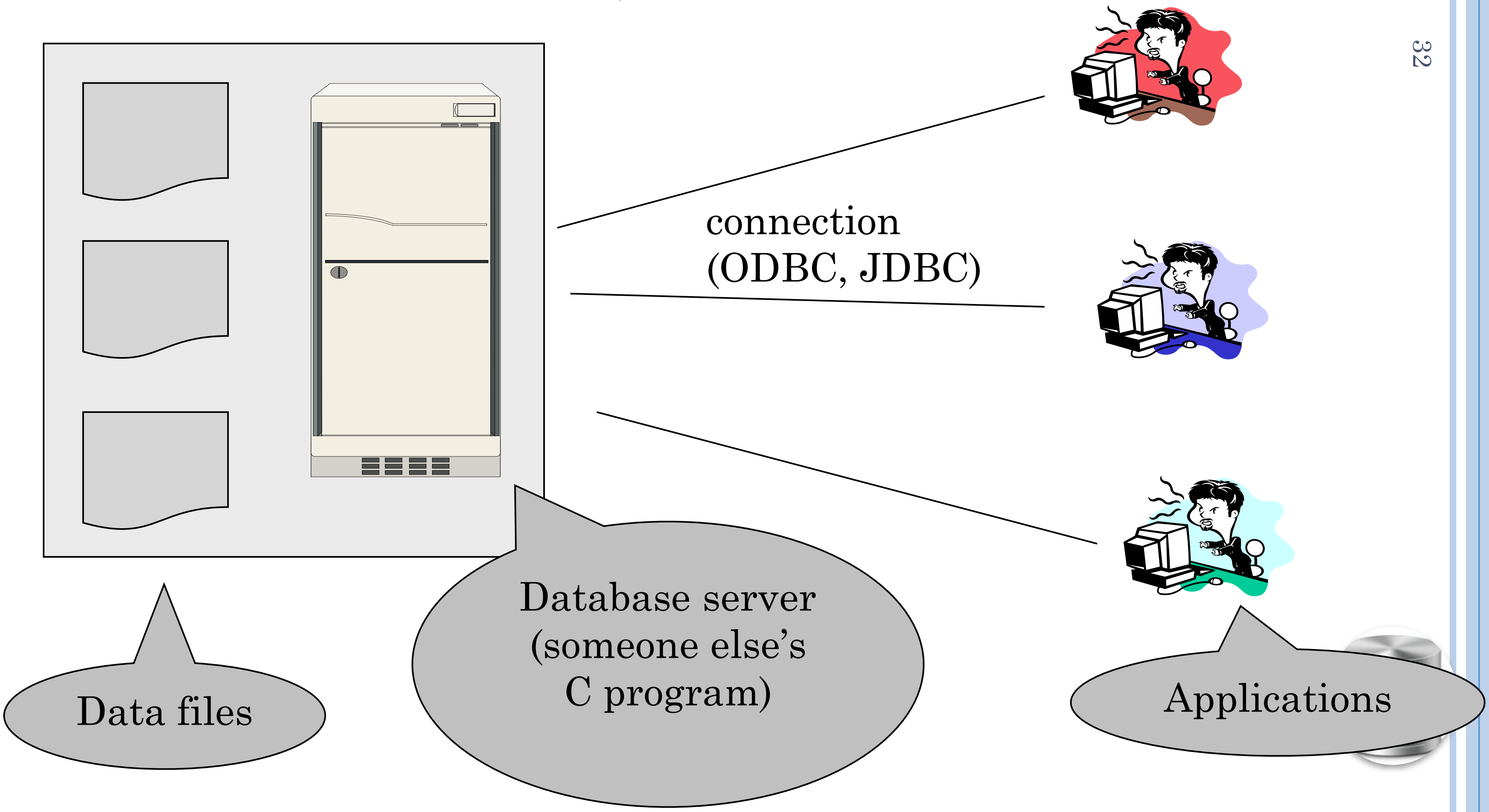
- **Client**
  - a user machine with user interface capabilities and local processing.
  - It can access the specialized servers as needed
- **Server**
  - contains both hardware and software that provide services to clients, such as printing, file access, or database access

Figure 2.5
Logical two-tier client/server architecture.



Enters a DMBS

"Two tier database system"



connection (ODBC, JDBC)

Data files

Database server (someone else's C program)

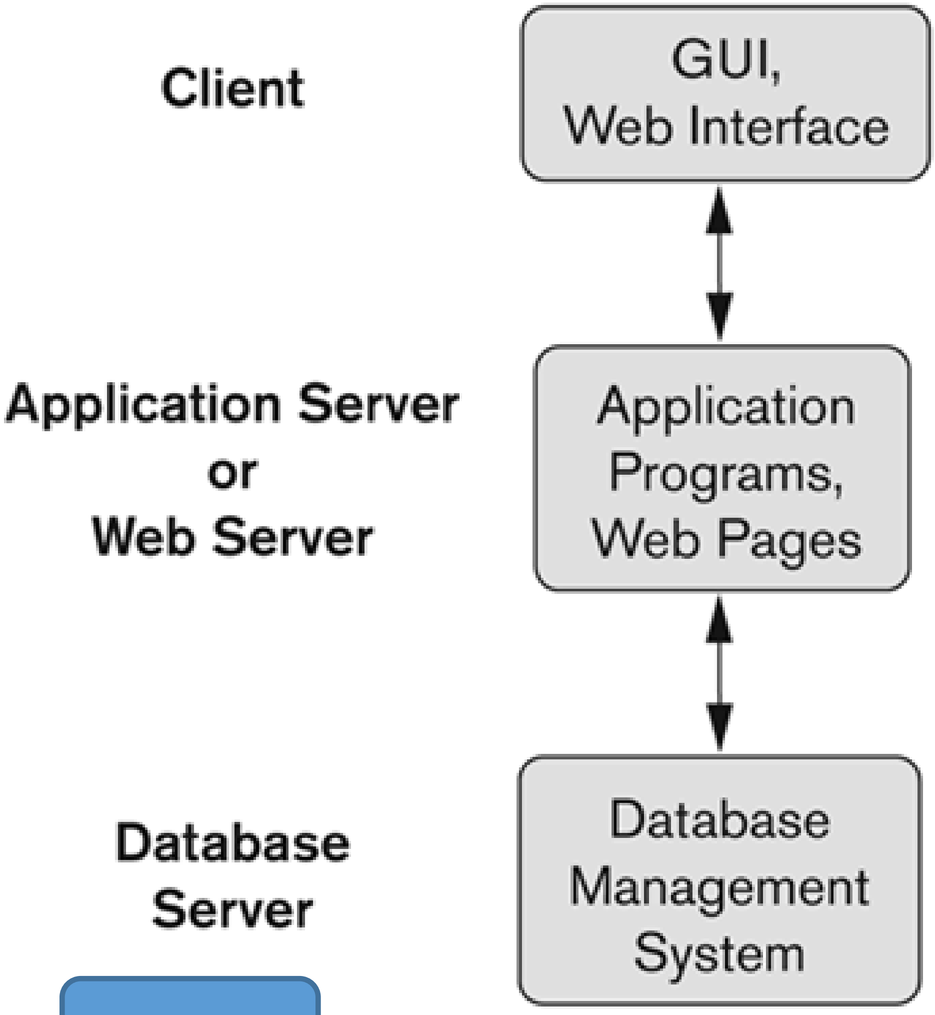
Applications

Three Tier Client-Server Architecture

Adds Intermediate layer called Application Server or Web Server.

Middle Tier stores business logic that is use to access data from the database server

Enhance security as DB server is only accessible via middle tier


Client
GUI, Web Interface
Application Server or Web Server
Application Programs, Web Pages
Database Server
Database Management System

RDBMS