BITS Pilani
Hyderabad Campus

Database Systems (CS F212)

Prof.R.Gururaj
CS&IS Dept.

Introduction DBMS Concepts and Architecture(Ch.1 & 2 of T1)

Content

- ❑ *Database Systems and Significance*
- ❑ *Database System Applications*
- ❑ *DBMS*
- ❑ *Database system environment*
- ❑ *Advantages of DBMS over traditional file systems*
- ❑ *Data model*
- ❑ *Data abstraction and three-schema architecture*
- ❑ *Data independence*
- ❑ *Database users*

Database Systems

Databases and Systems to manage them have become significant components of any present day business of any nature.
These databases help businesses to perform their day-to-day activities in an efficient and effective manner.

- Banking and Finance
- E-business
- Science and Engg
- Sports
- Education and so on….

Here some program access the database.

In past few years advances in technology has given raise to new concepts-

- ❑ Multimedia databases
- ❑ GIS
- ❑ Web data
- ❑ Data warehousing and mining
- ❑ Big Data
- ❑ No-SQL Databases
- ❑ Distributed databases
- ❑ In memory databases

etc.

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. Not a random assortment of data.

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.

Descriptions are stored in the form of Database catalog or Data dictionary and are 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, *updating* some data items, and *deleting* certain items from the database.

Database System

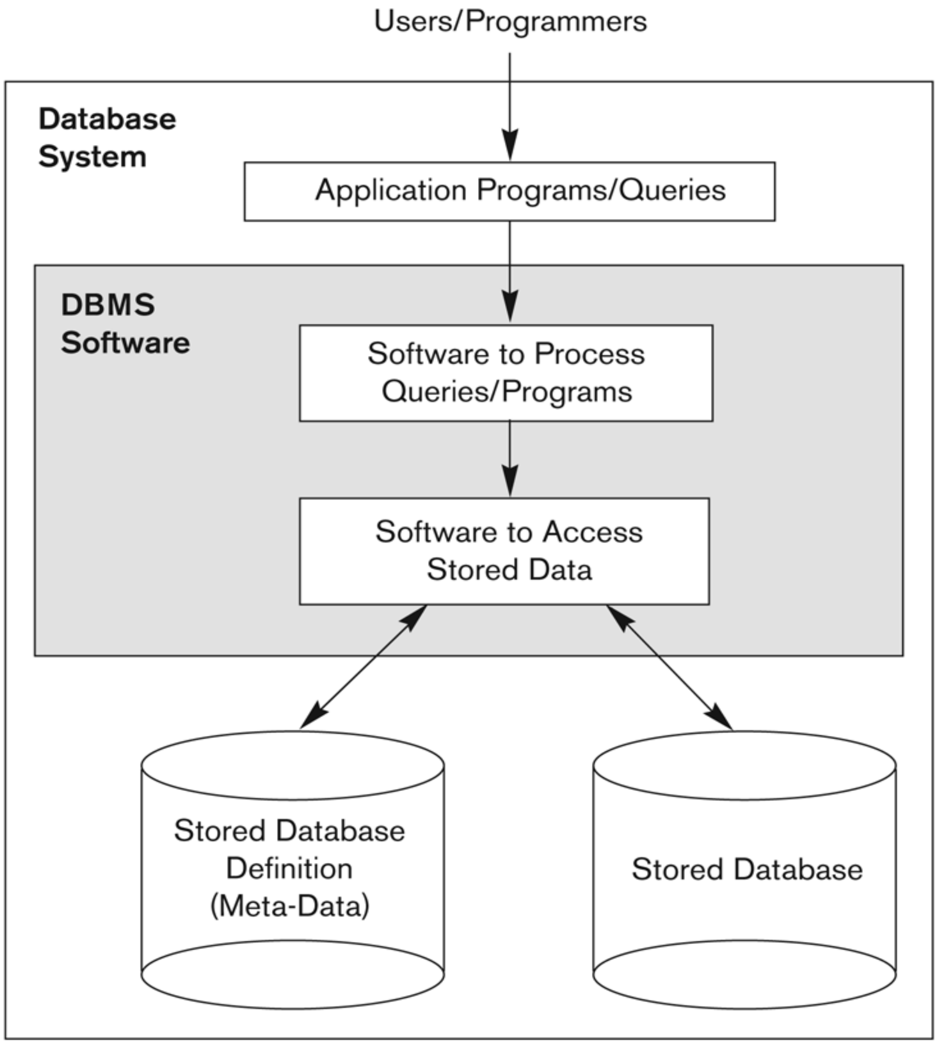

Users/Programmers
Database System
Application Programs/Queries
DBMS Software
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition (Meta-Data)
Stored Database

Figure 1.1
A simplified database system environment.

Traditional file Systems as Databases

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

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

Disadvantages of the above System

- ❑ Data redundancy leading to inconsistency.
- ❑ Difficulty in accessing data (possible operations need to be hard- coded in programs).
- ❑ Data is scattered across different files.
- ❑ Integrity problems.
- ❑ Atomicity problems due to non availability of transaction management.
- ❑ Concurrency control is not there.
- ❑ Security provided by the OS is not adequate.
- ❑ The above shortcomings prompted the development of DBMS.

Main characteristics of database approach versus file-processing approach

- ❑ *Self-describing nature of database systems*
- ❑ *Insulation between programs and data (program-data independence)*
- ❑ *Support multiple views of data*
- ❑ *Sharing of data and multiuser transaction processing*

Advantages of using DBMS

- ❑ We can control the degree of redundancy thus avoiding inconsistency.
- ❑ Access control privileges to different user categories/groups can be specified.

  (Security and Authorization subsystem)
- ❑ Persistent storage for program objects.
- ❑ DBMS provides multiuser interface to the underlying database.

  > Through SQL commands at SQL prompt.

  > Through programming languages like JAVA, C, C++, PHP etc.

  > Through form-based interface.
- ❑ Integrity constraints like- unique key, foreign key, primary key, NOT NULL etc. can be specified on the data.
- ❑ Supports efficient backup and recovery mechanisms.
- ❑ Supports transaction management.
- ❑ Provides data independence.

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.

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.

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

1. ***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

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

Database Schema

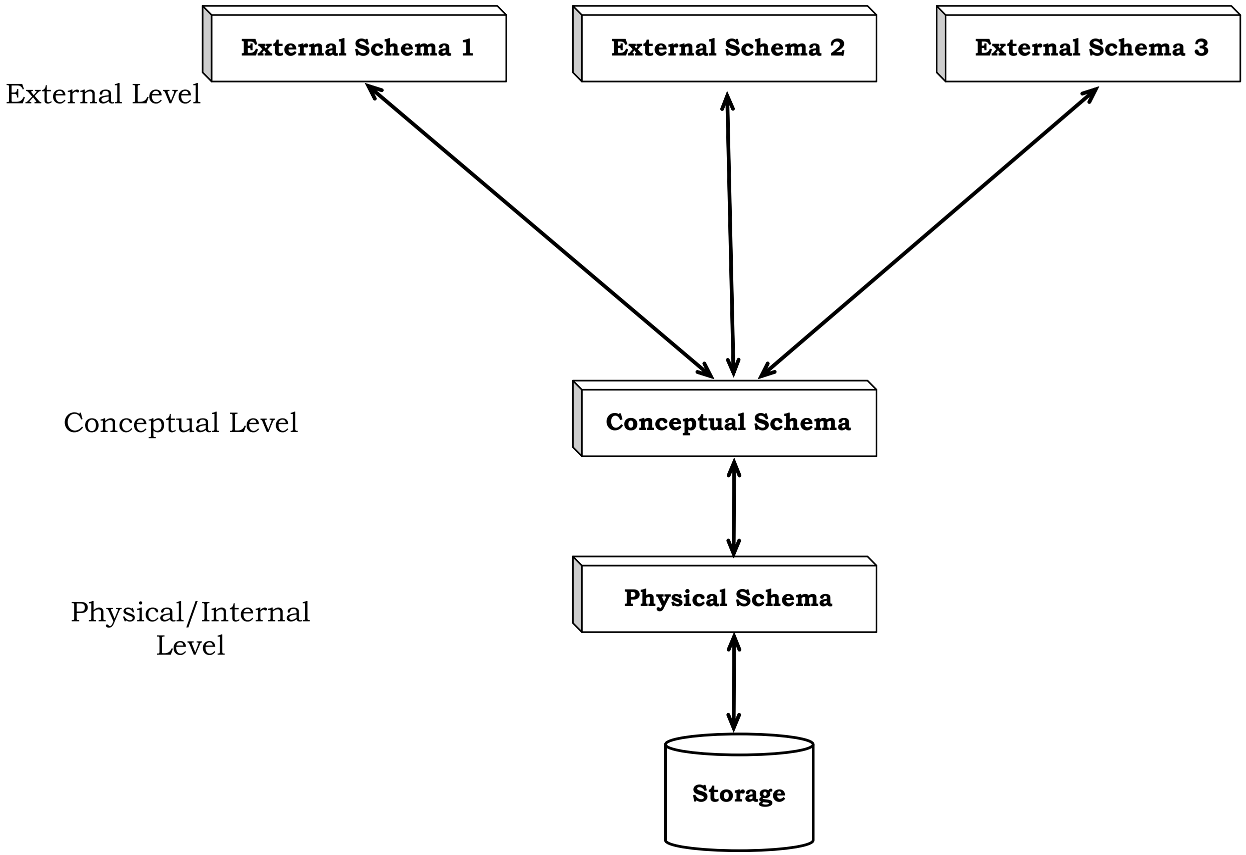
Database Schema: Description of a database is called as *database Schema*

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



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.

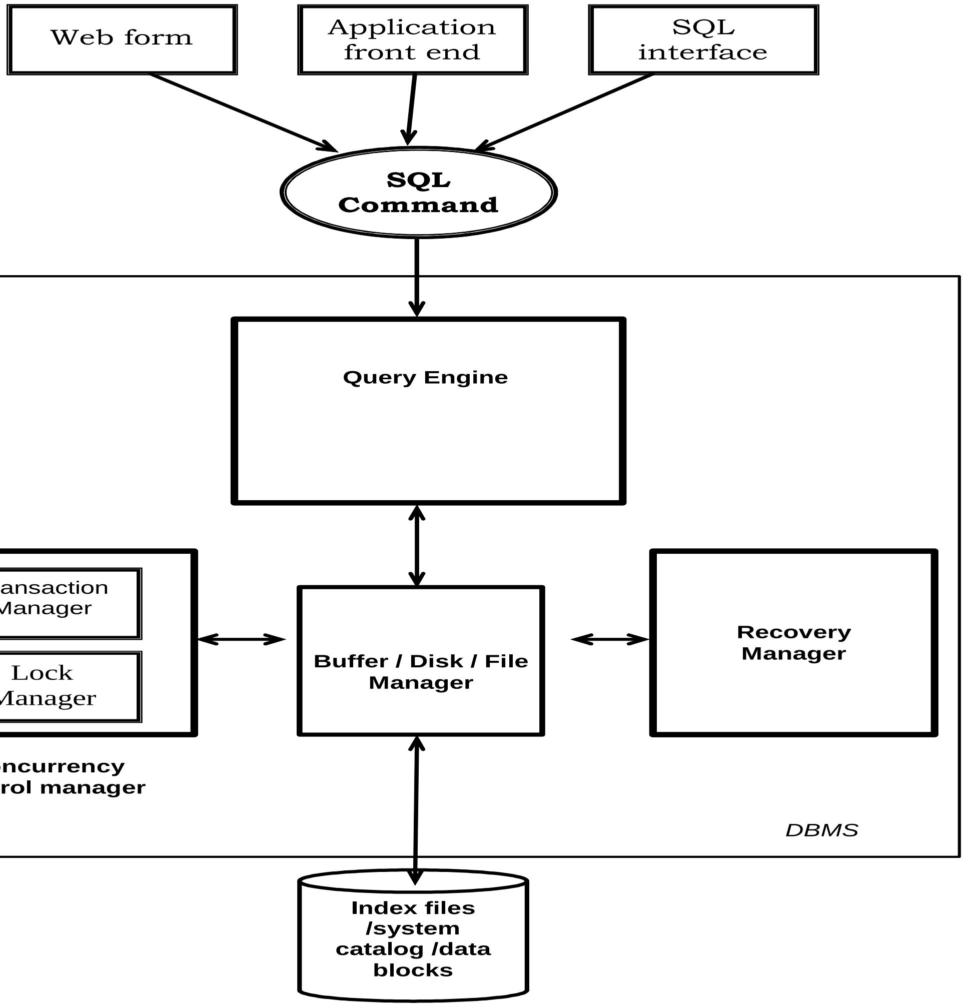
2. *Physical data independence:* The changes in physical features of storage, i.e., changes to the physical storage format should not affect schema at conceptual level.

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

Metadata:

The DBMS stores the description of schema constructs, the constraints, and other related info about the data in the DB. These descriptions are called as *metadata*. Otherwise it can be seen as data about data.

DBMS Structure



Users of Database Systems

- ❑ *Casual users (occasional users)*
- ❑ *Naïve users (bank tellers, reservation agents)*
- ❑ *Application Programmers (SW developers)*
- ❑ *Sophisticated users (Engineers, scientists having complex requirements and have good knowledge)*
- ❑ *DBA*

Responsibilities of DBA:

1. Design of physical & Conceptual schemas
2. Security and authorization
3. Data availability , recovery and backup
4. Monitoring space
5. Monitoring jobs
4. Database tuning- modifying the schemas to meet the requirements

DBMS Client-Server Architecture

DBMS and Client-server Architectre:

1. *Centralized:*

2. *Client server:*

   - ❖two-tier
   - ❖three-tier

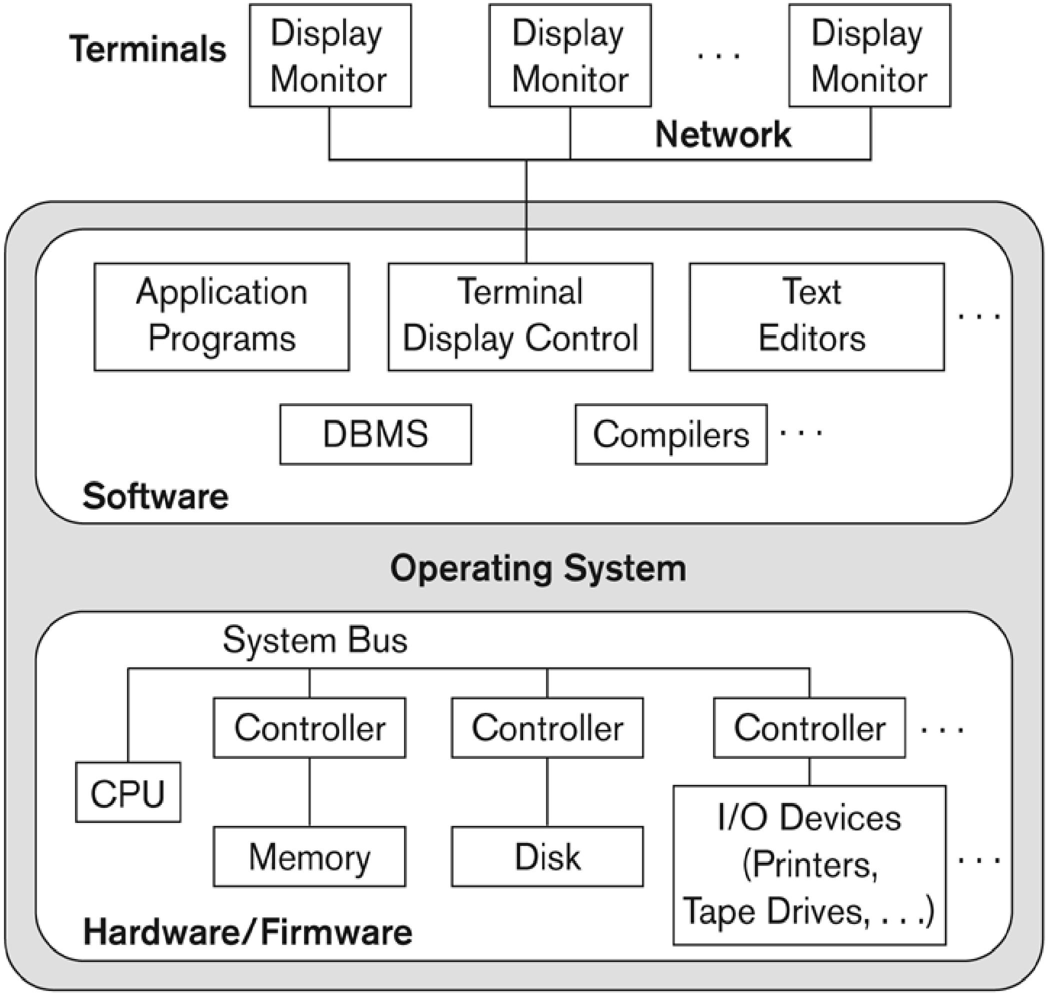


Figure 2.4
A physical centralized architecture.

Figure 2.5
Logical two-tier client/server architecture.

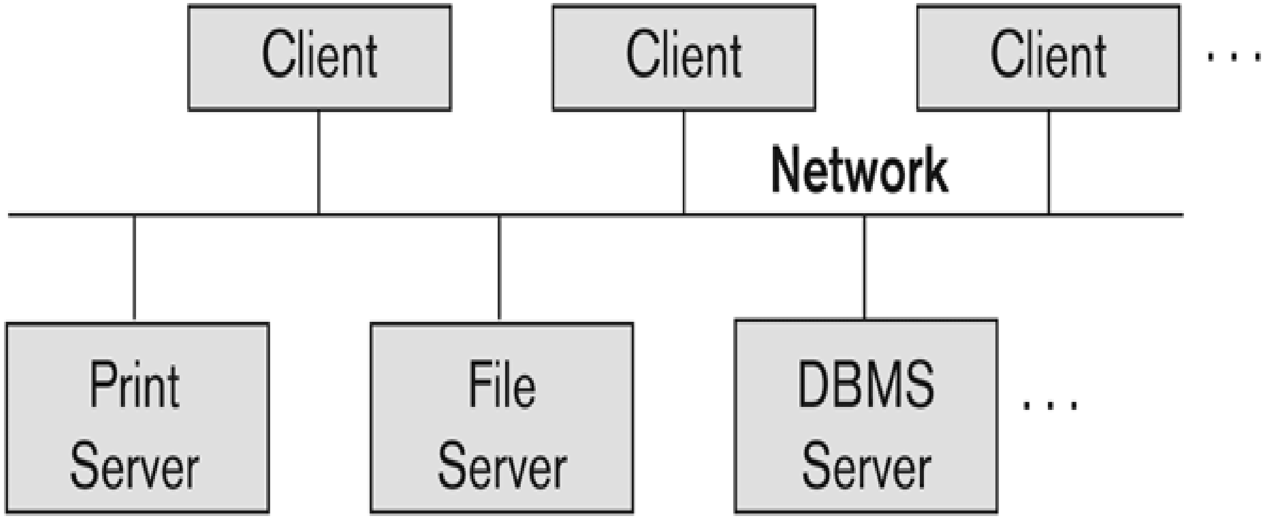
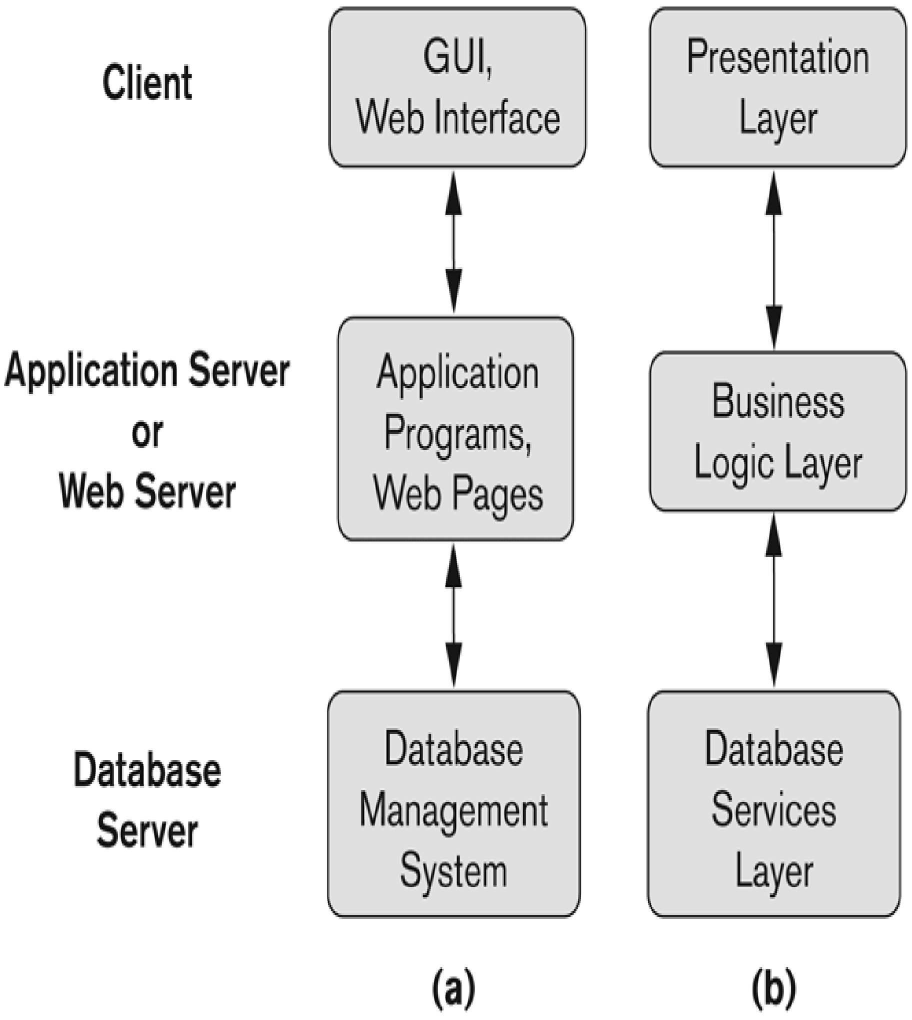


Figure 2.7
Logical three-tier client/server architecture, with a couple of commonly used nomenclatures.



Summary

- ✓ *What is Data, Database, and DBMS*
- ✓ *Importance of DBMS*
- ✓ *Advantages of DBMS over traditional file systems*
- ✓ *What is a data model*
- ✓ *What is abstraction*
- ✓ *What is three schema architecture of a DBMS*
- ✓ *What is data independence*
- ✓ *Components of a DBMS*
- ✓ *Users of DBS*
- ✓ *Database Client server Architectures*