

Chapter 2

Database Environment Transparencies

1

Chapter 2 - Objectives

- ◆ **The purpose and origin of the three-level database architecture.**
- ◆ **The contents of the external, conceptual and internal levels.**
- ◆ **The purpose of the external/conceptual and the conceptual/internal mappings.**

2

Chapter 2 - Objectives

- ◆ **The meaning of logical and physical data independence.**
- ◆ **The distinction between a Data Definition Language (DDL) and a Data Manipulation Language (DML).**
- ◆ **A classification of data models.**

3

Chapter 2 - Objectives

- ◆ **The purpose and importance of conceptual modeling.**
- ◆ **The meaning of the client–server architecture and the advantages of this type of architecture for a DBMS.**

4

Objectives of Three-Level Architecture

- ◆ **All users should be able to access same data.**
- ◆ **A user's view is immune to changes made in other views.**
- ◆ **Users should not need to know physical database storage details.**

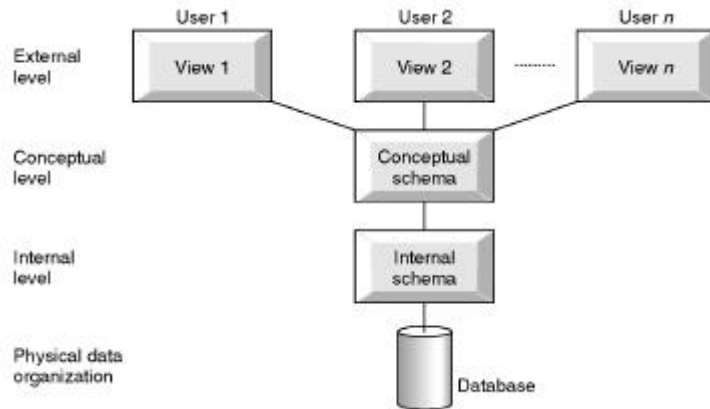
6

Objectives of Three-Level Architecture

- ◆ **DBA should be able to change database storage structures without affecting the users' views.**
- ◆ **Internal structure of database should be unaffected by changes to physical aspects of storage.**
- ◆ **DBA should be able to change conceptual structure of database without affecting all users.**

7

ANSI-SPARC Three-level Architecture



8

ANSI-SPARC Three-level Architecture

◆ External Level

- Users' view of the database. Describes that part of database that is relevant to a particular user.

◆ Conceptual Level

- Community view of the database. Describes what data is stored in database and relationships among the data.

9

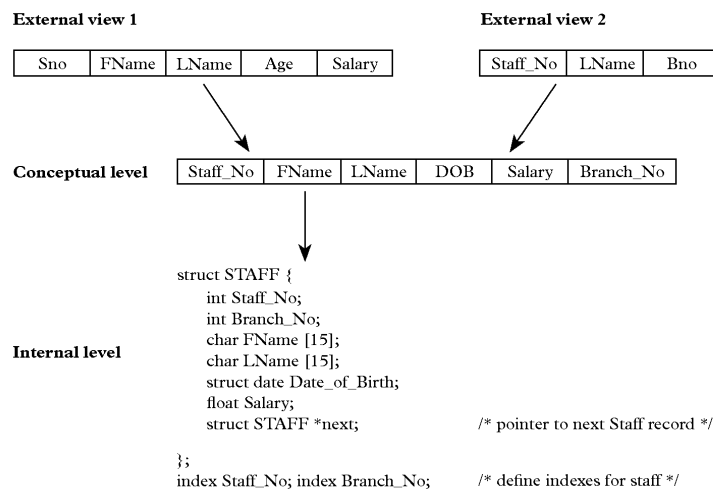
ANSI-SPARC Three-level Architecture

◆ Internal Level

- **Physical representation of the database on the computer. Describes how the data is stored in the database.**

10

Differences between Three Levels of ANSI-SPARC Architecture



11

Data Independence

◆ Logical Data Independence

- Refers to immunity of external schemas to changes in conceptual schema.
- Conceptual schema changes e.g. addition/removal of entities.
- Should not require changes to external schema or rewrites of application programs.

12

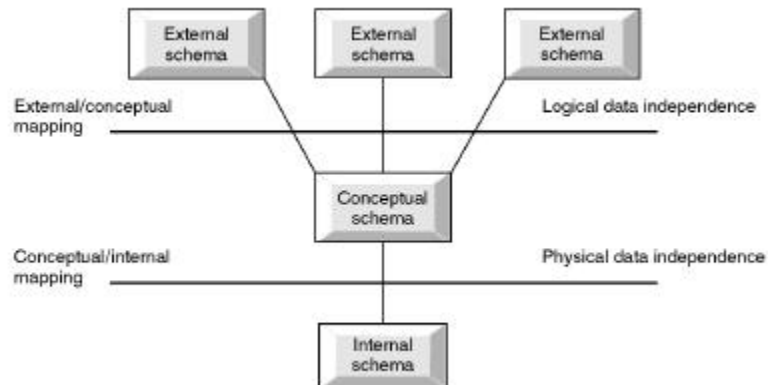
Data Independence

◆ Physical Data Independence

- Refers to immunity of conceptual schema to changes in the internal schema.
- Internal schema changes e.g. using different file organizations, storage structures/devices.
- Should not require change to conceptual or external schemas.

13

Data Independence and the ANSI-SPARC Three-level Architecture



14

Database Languages

- ◆ **Data Definition Language (DDL)**
 - Allows DBA or user to describe and name entitles, attributes and relationships required for the application.

15

Database Languages

- ◆ **Data Manipulation Language (DML)**
 - **Provides basic data manipulation operations on data held in the database.**
 - **Procedural DML - allows user to tell system exactly how to manipulate data.**
 - **Non-Procedural DML - allows user to state what data is needed rather than how it is to be retrieved.**

16

Data Model

- ◆ **Collection of concepts for describing data, relationships between data and constraints on the data in an organization.**
- ◆ **Data Model comprises:**
 - **A structural part**
 - **A manipulative part**
 - **Possibly a set of integrity rules**

18

Data Model

◆ Object-based Data Models

- Entity-Relationship**
- Semantic**
- Functional**
- Object-Oriented**

20

Data Model

◆ Record-based Data Models

- Relational Data Model**
- Network Data Model**
- Hierarchical Data Model**

◆ Physical Data Models

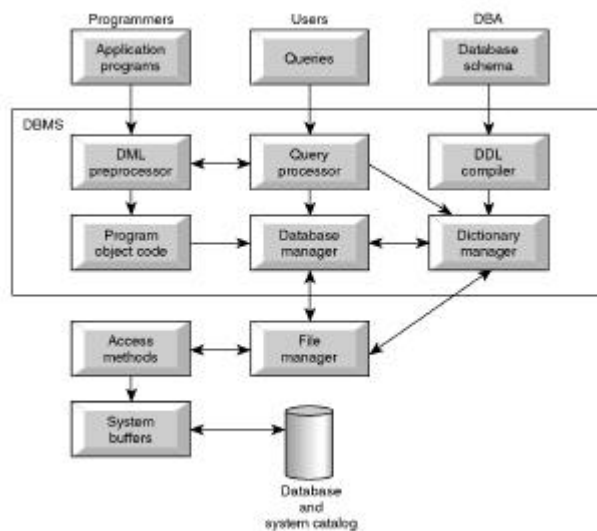
21

Conceptual modeling

- ◆ The process of developing a conceptual data model that is
 - a complete and accurate representation of an organization's data requirements.
 - independent of implementation details.

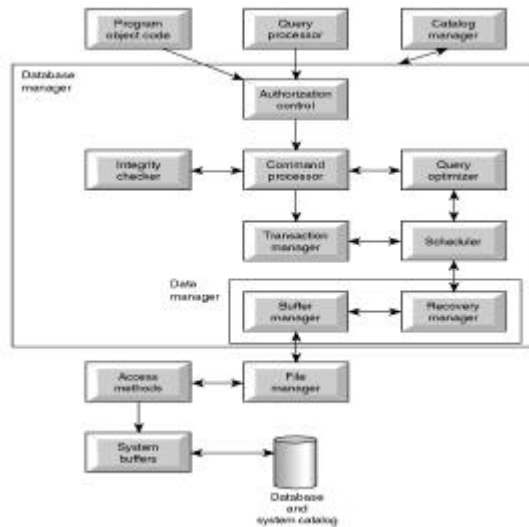
22

Components of a DBMS



28

Components of Database Manager (DM)



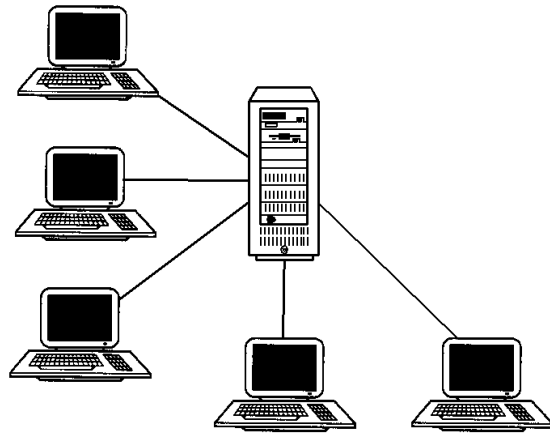
30

Teleprocessing

- ◆ Traditional architecture.
- ◆ Single mainframe with a number of terminals attached.
- ◆ Trend is now towards downsizing.

33

Teleprocessing Topology



34

File-server

- ◆ File-server is connected to several workstations across a network.
- ◆ Database resides on file-server.
- ◆ DBMS and applications run on each workstation.

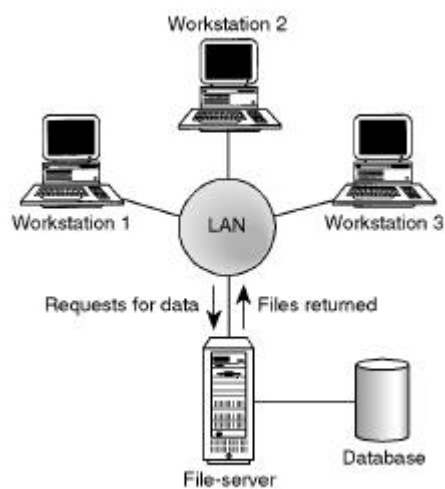
35

File-server

- ◆ **Disadvantages include:**
 - **Significant network traffic.**
 - **Copy of DBMS on each workstation.**
 - **Concurrency, recovery and integrity control more complex.**

36

File-server Architecture



37

Client-server

- ◆ **Server holds the database and the DBMS.**
- ◆ **Client manages the user interface and runs applications.**

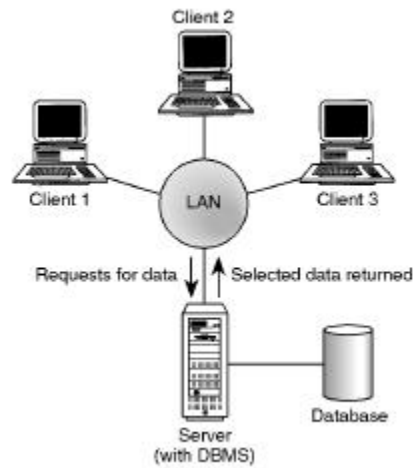
38

Client-server

- ◆ **Advantages include:**
 - **Wider access to existing databases.**
 - **Increased performance.**
 - **Possible reduction in hardware costs.**
 - **Reduction in communication costs.**
 - **Increased consistency.**

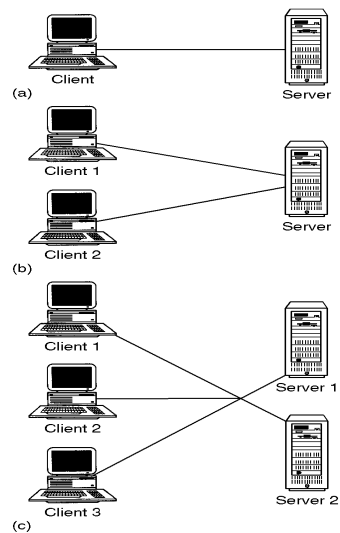
39

Client-server Architecture



40

Alternative Client-server Topologies



41

Summary of Client-server Functions

<i>Client</i>	<i>Server</i>
Manages the user interface	Accepts and processes database requests from clients
Accepts and checks syntax of user input	Checks authorization
Processes application	Ensures integrity constraints not violated
Generates database requests and transmits to server	Performs query/update processing and transmits response to client
Passes response back to user	Maintains system catalog
	Provides concurrent database access
	Provides recovery control