Chapter 2

Database Environment

Chapter 2 - Objectives

- Purpose of three-level database architecture.
- Contents of external, conceptual, and internal levels.
- Purpose of external/conceptual and conceptual/internal mappings.
- Meaning of logical and physical data independence.
- Distinction between DDL and DML.
- A classification of data models.

Chapter 2 - Objectives

- Purpose/importance of conceptual modeling.
- Typical functions and services a DBMS should provide.
- Function and importance of system catalog.
- Software components of a DBMS.
- Meaning of client–server architecture and advantages of this type of architecture for a DBMS.
- Function and uses of Transaction Processing Monitors.

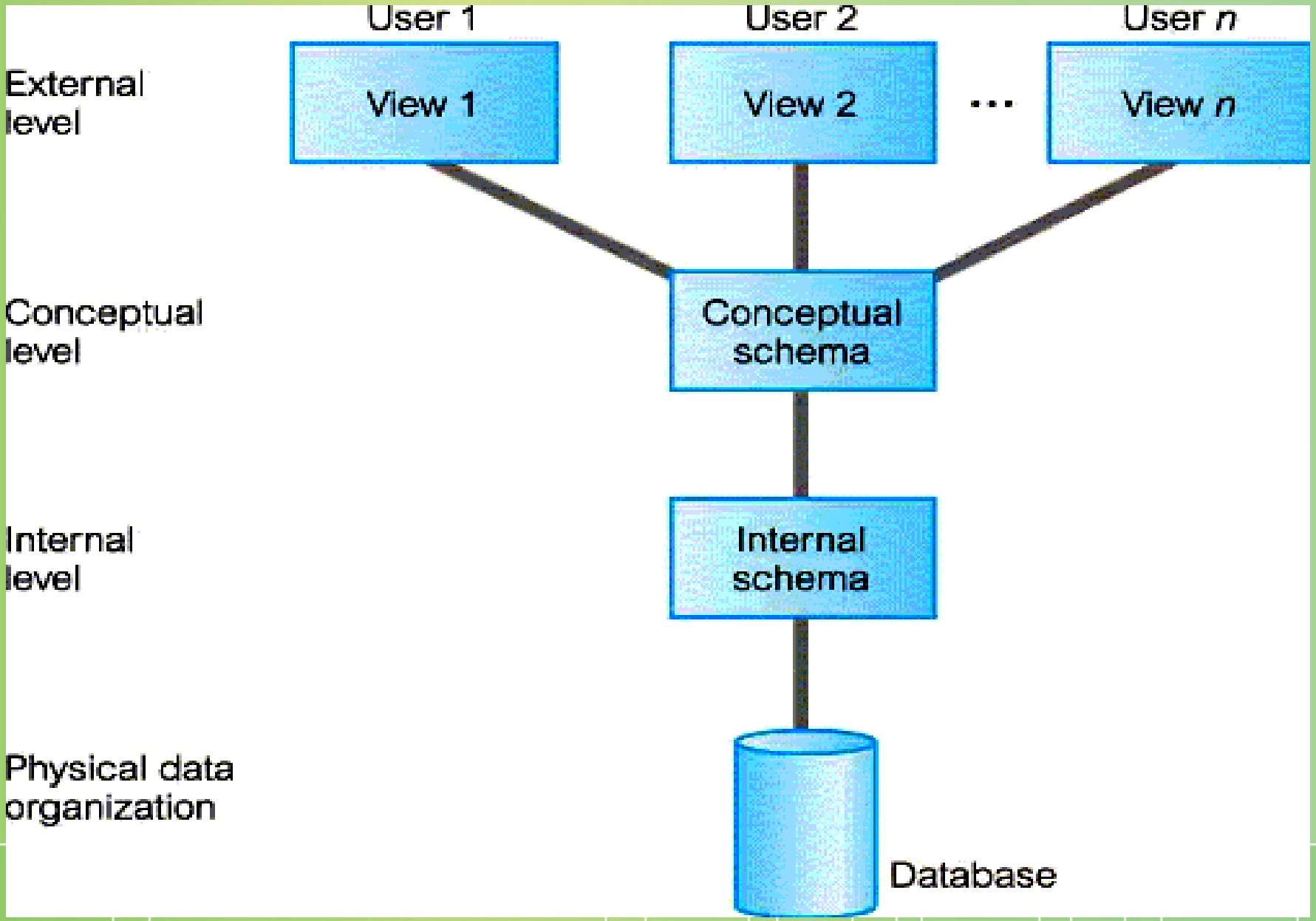
Objectives of Three-Level Architecture

- The objective is to separate each user's view of the database from the way the database is physically represented.
- Purpose 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.

Objectives of Three-Level Architecture

- **Purpose 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.**

ANSI-SPARC Three-Level Architecture



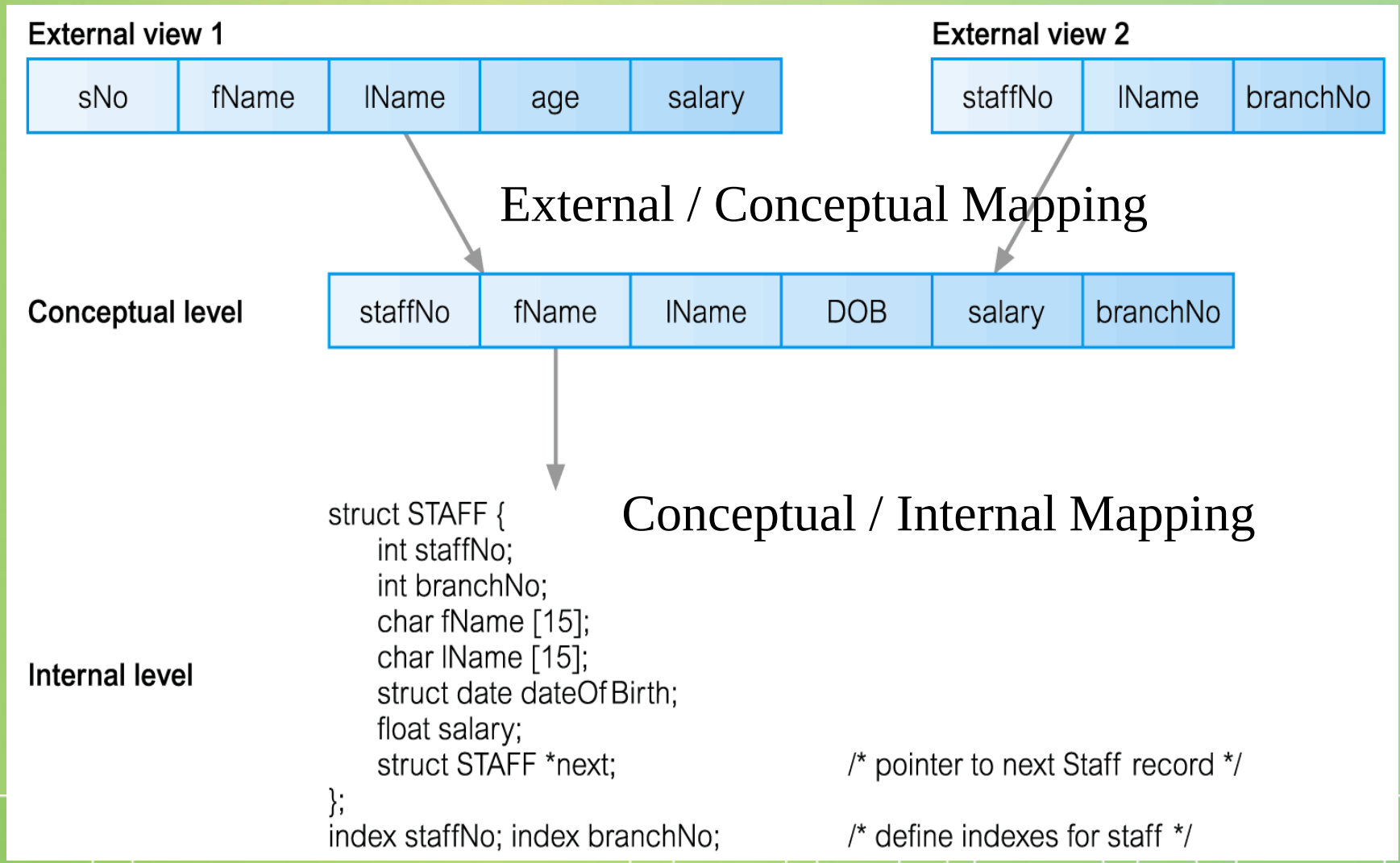
ANSI-SPARC Three-Level Architecture

- **External Level**
  - **Users' view of the database.**
  - **Describes that part of database that is relevant to a particular user.**
- **Conceptual Level**
  - **Contains the logical structure of the database.**
  - **Describes what data is stored in database and relationships among the data.**

ANSI-SPARC Three-Level Architecture

- **Internal Level**
  - **Covers the physical implementation of the database on the computer.**
  - **Describes how the data is stored on the storage devices.**

Differences between Three Levels of ANSI-SPARC Architecture



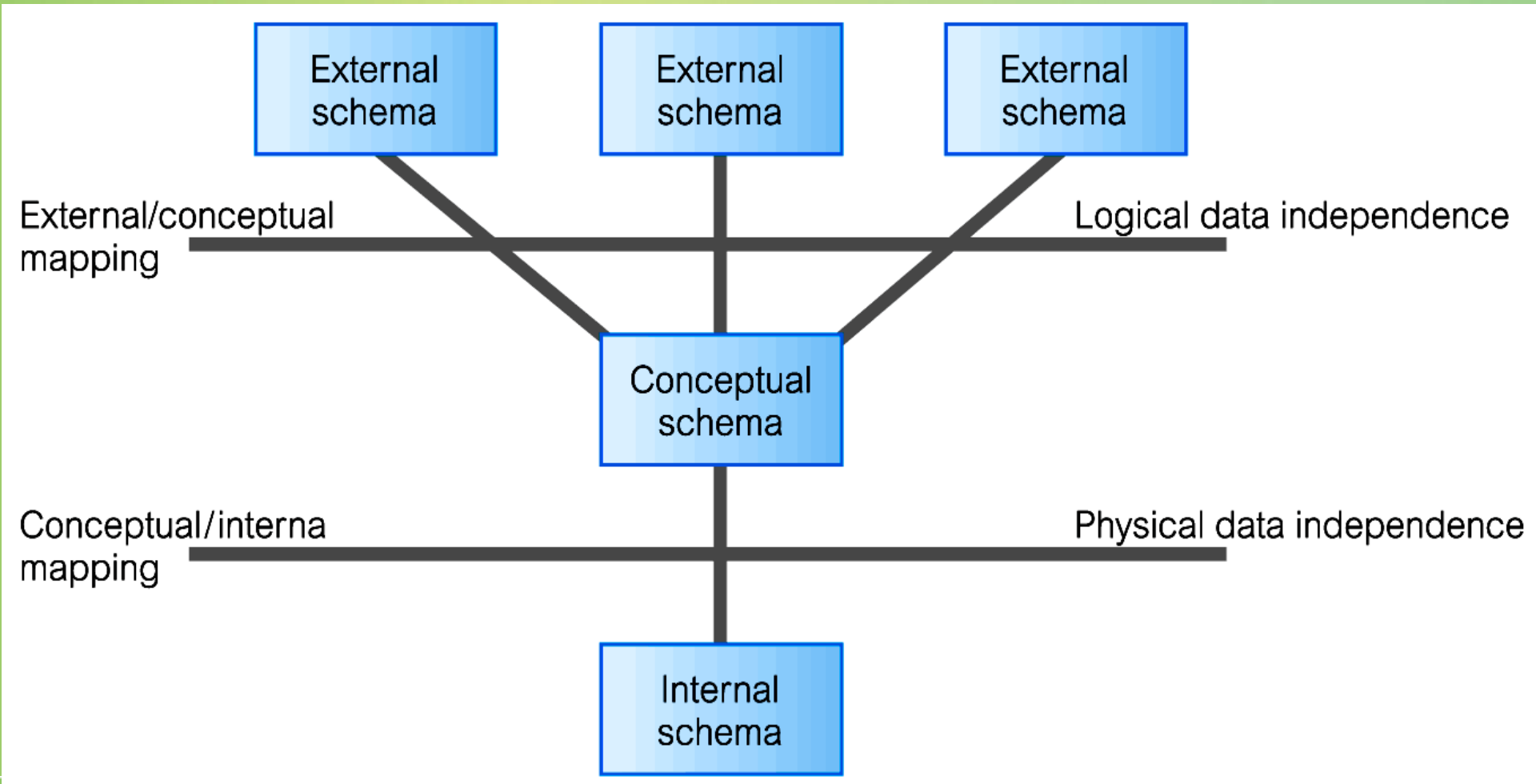
Data Independence

- A major objective for the three-level architecture is to provide data independence – upper levels are unaffected by changes to lower levels.
- Logical Data Independence
  - Refers to immunity of external schemas to changes in conceptual schema.
  - Conceptual schema changes (e.g. addition/removal of entities, attributes, ...)
  - Should not require changes to external schema or rewrites of application programs.

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.**

Data Independence and the ANSI-SPARC Three-Level Architecture



Database Languages

- **Data Definition Language (DDL)**
  - **Allows the DBA or user to describe and name entities, attributes, and relationships required for the application**
  - **plus any associated integrity and security constraints.**
  - **Can’t be used to manipulate data**

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 (hierarchical DBMSs).**
- **Non-Procedural DML**
  - **allows user to state what data is needed rather than how it is to be retrieved (RDBMS).**
- **Fourth Generation Languages (4GLs)**
  - **Non-procedural (SQL, QBE)**

Data Model

A model is a representation of a real-world object (ex. an organization).

It is an abstraction that concentrates on the essential, inherent aspects of an organization.

A data model represents the organization itself. It allows database designers and end-users to communicate unambiguously and accurately their understanding of the organizational data.

Data Model

Data Model comprises:

- a structural part;
  - a set of rules according to which databases can be constructed;
- a manipulative part;
  - defining the types of operation that are allowed on the data
- possibly a set of integrity rules.
  - ensuring the data is accurate.

Data Model

- **Purpose**
  - **To represent data in an understandable way.**
- **Categories of data models include:**
  - **Object-based**
  - **Record-based**
  - **Physical.**

Data Models

- **Object-Based Data Models (External Level)**
  - Use entities, attributes, and relationships.
  - An entity is a person, place, thing, …
  - An attribute is a property of an object
  - A relationship is an association between entities
- **Record-Based Data Models (Conceptual Level)**
  - Relational Data Model
  - Network Data Model
  - Hierarchical Data Model.
- **Physical Data Models (Internal Level)**
  - Describe how data is stored in the computer

Relational Data Model

Branch

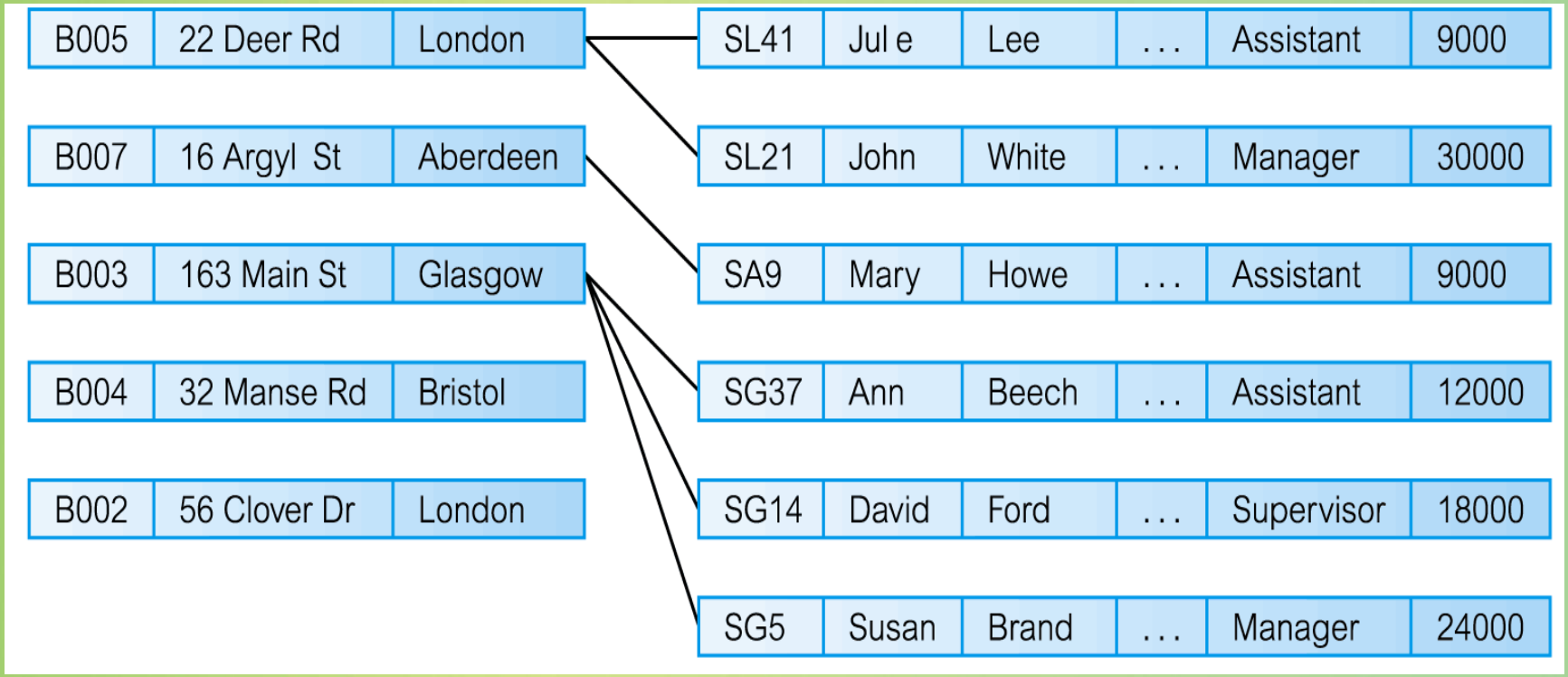
branchNo	street	city	postCode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

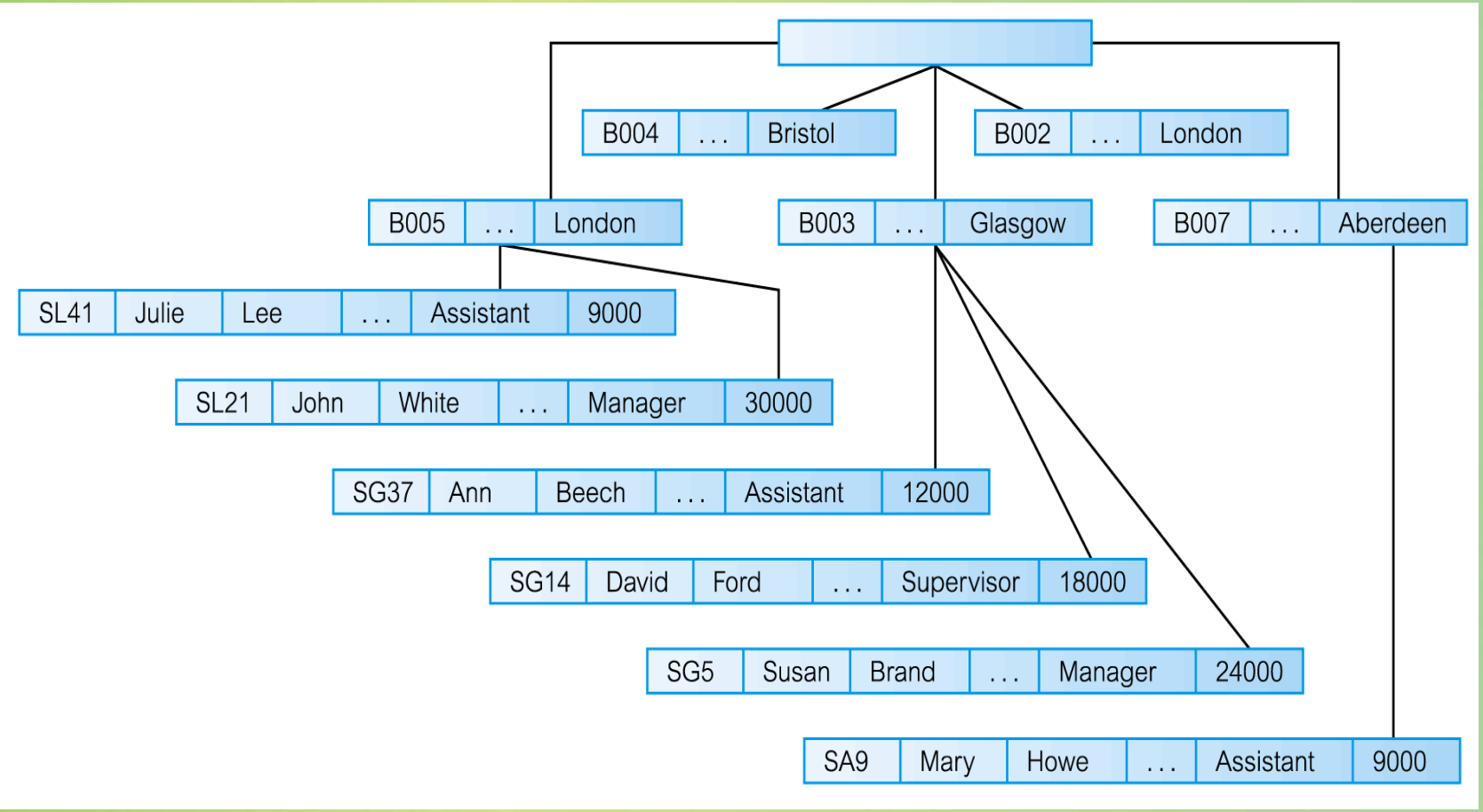
Represented as tables

Network Data Model



Represented as generalized graph.

Hierarchical Data Model



Represented as tree graph.

Conceptual Modeling

- Conceptual schema is the heart of the database supporting all user views.
- Should be complete and accurate representation of an organization's data requirements.
- Conceptual modeling is the process of developing a model of information use that is independent of implementation details.
- Result is a conceptual data model.

Functions of a DBMS

- **Data Storage, Retrieval, and Update.**
  - **Internal physical implementation detail hidden from the user**
- **A User-Accessible Catalog.**
  - **Descriptions of data items (metadata) are stored and accessible to users**
- **Transaction Support.**
  - **Ensures either all the updates corresponding to a certain transaction are made or none of them is made**

Functions of a DBMS

- **Concurrency Control Services.**
  - **Ensure the database is updated correctly when multiple users are updating the database concurrently**

Time	T_1	T_2	bal_x
t_1		read(bal_x)	100
t_2	read(bal_x)	$bal_x = bal_x + 100$	100
t_3	$bal_x = bal_x - 10$	write(bal_x)	200
t_4	write(bal_x)		90
t_5			90

- **Recovery Services.**
  - **Ensure if an event caused the database to be damaged in any way, the database will be restored to a consistent state.**

Functions of a DBMS

- **Authorization Services.**
  - **Ensure that only authorized users can access the database**
- **Support for Data Communication.**
  - **Capable of integrating with communication software so that the database can be accessed through Internet connection**
- **Integrity Services.**
  - **Ensure both the data in the database and changes to the data follow certain rules.**
  - **Database integrity – the correctness and consistency of stored data**

Functions of a DBMS

- **Services to Promote Data Independence.**
  - **A DBMS must include facilities to support the independence of programs from the actual structure of the database**
- **Utility Services.**
  - **Utility programs help the DBA administer the database effectively. They include:**
  - **Import / export facilities**
  - **monitoring facilities**
  - **garbage collection and reallocation**

System Catalog

- Repository of information (metadata) describing the data in the database.
- One of the fundamental components of DBMS.
- Typically stores:
  - names, types, and sizes of data items;
  - constraints on the data;
  - names of authorized users;
  - data items accessible by a user and the type of access;
  - external, conceptual, and internal schemas and the mappings between the schemas
  - usage statistics.