TRIBHUVAN UNIVERSITY
SAMRIDDHI COLLEGE
Lokanthali-16, Bhaktapur, Nepal

Bachelor of Science in
Computer Science & Information Technology
(B.Sc. CSIT)
Fourth Semester

Unit 1. *Database and Database Users*

by:

Er. Bikal Adhikari

(*B.E., M.Sc. Engg., M.Sc. ISE*)

Date:- 21st Nov., 2023

UNIT OUTLINE

Unit 1. Database and Database Users

2 hours

Introduction; Characteristics of the Database Approach; Actors on the Scene; Workers behind the Scene; Advantages of Using the DBMS Approach

INTRODUCTION

Data :

→ recording of “something” measured

→ Raw facts and figures, just measured

Information:

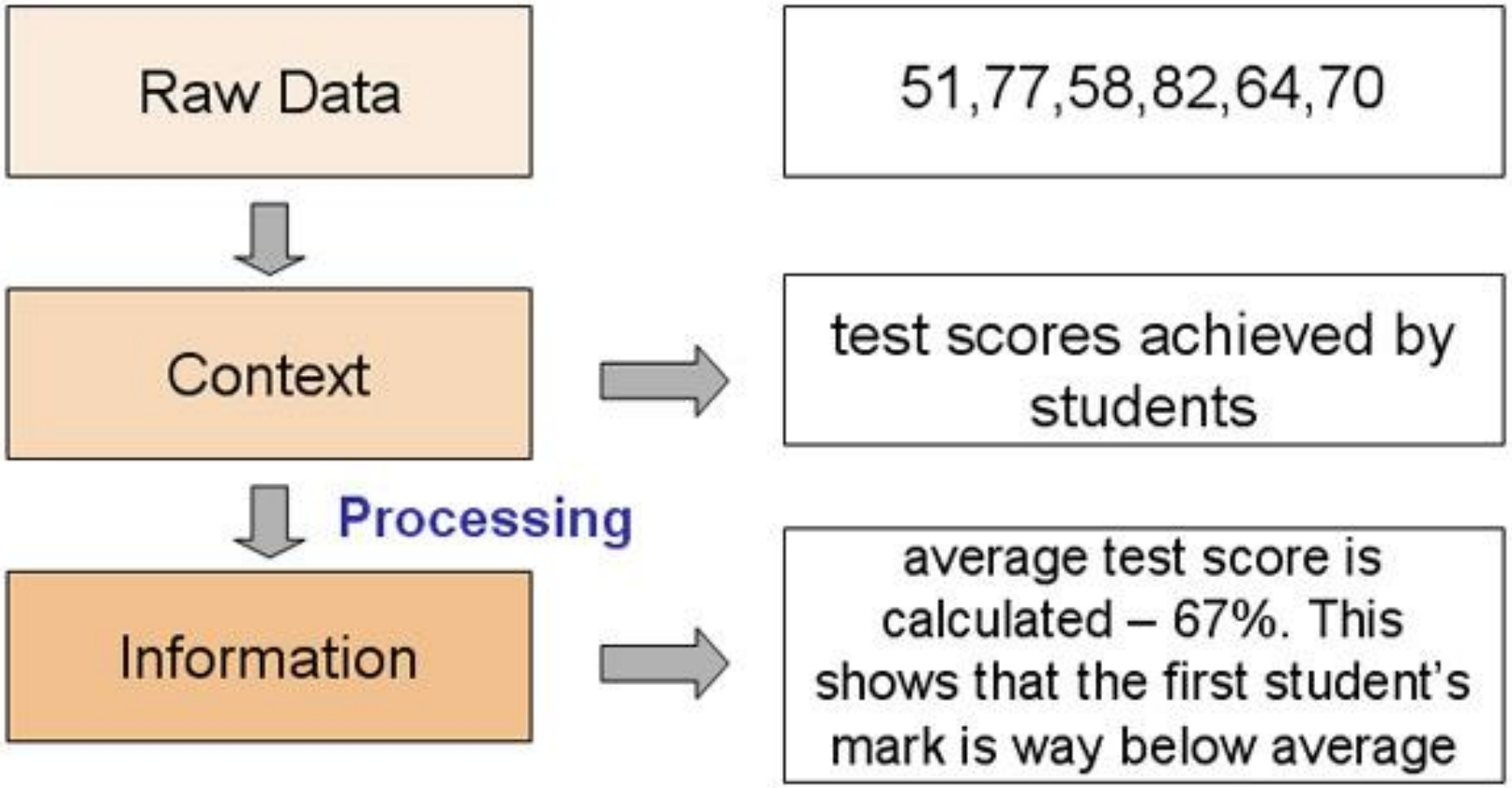
→ Information is the result of processing, manipulating and organizing data in a way that adds to the knowledge of the receiver.

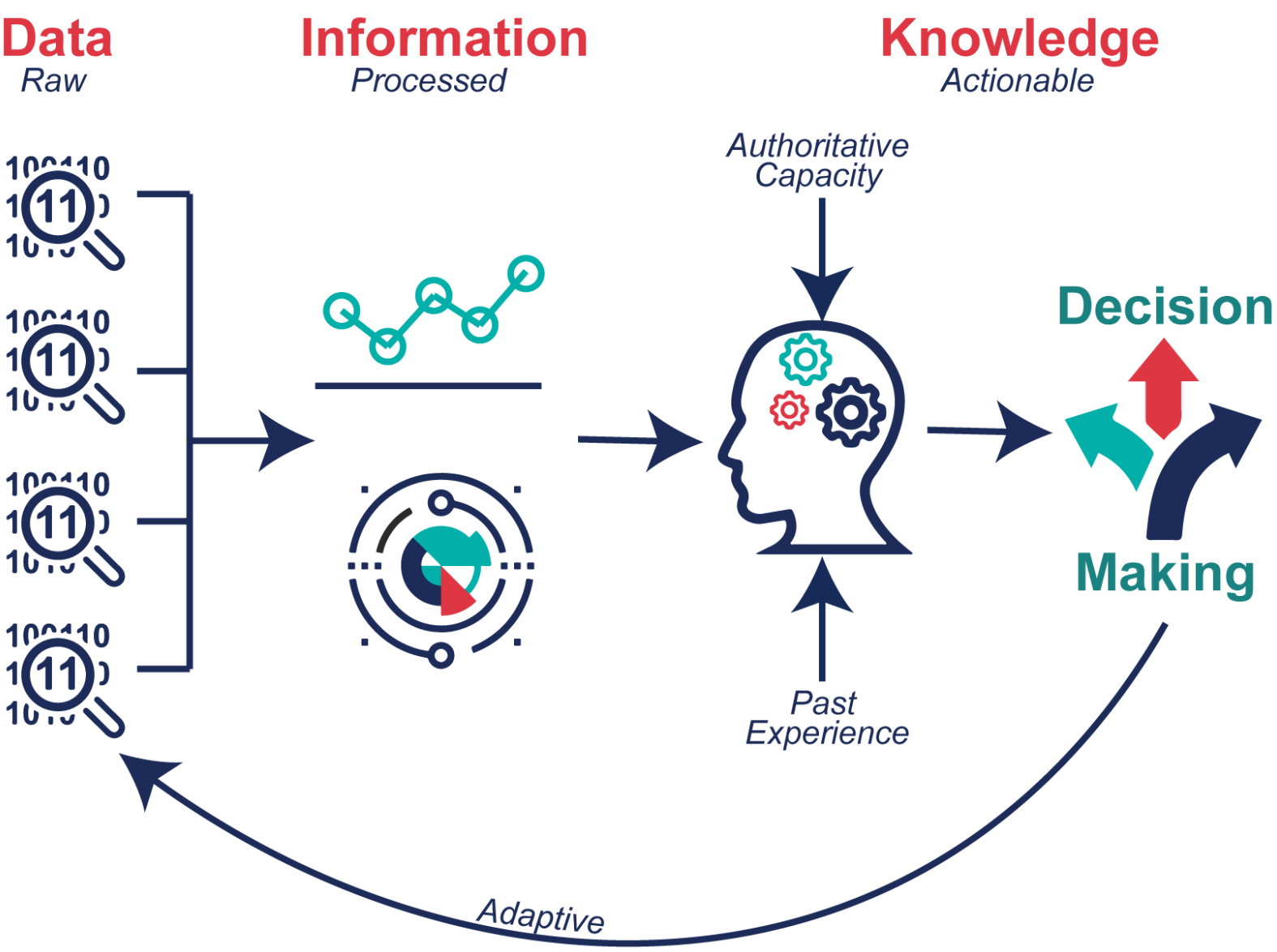
→ Processed data

Knowledge :

→ Knowledge is normally processed by means of structuring, grouping, filtering, organizing or pattern recognition.

→ Highly structured information


Raw Data
51,77,58,82,64,70
Context
test scores achieved by students
Processing
Information
average test score is calculated – 67%. This shows that the first student's mark is way below average


Data
Raw
Information
Processed
Knowledge
Actionable
Authoritative Capacity
Decision Making
Past Experience
Adaptive

DATABASE SYSTEM

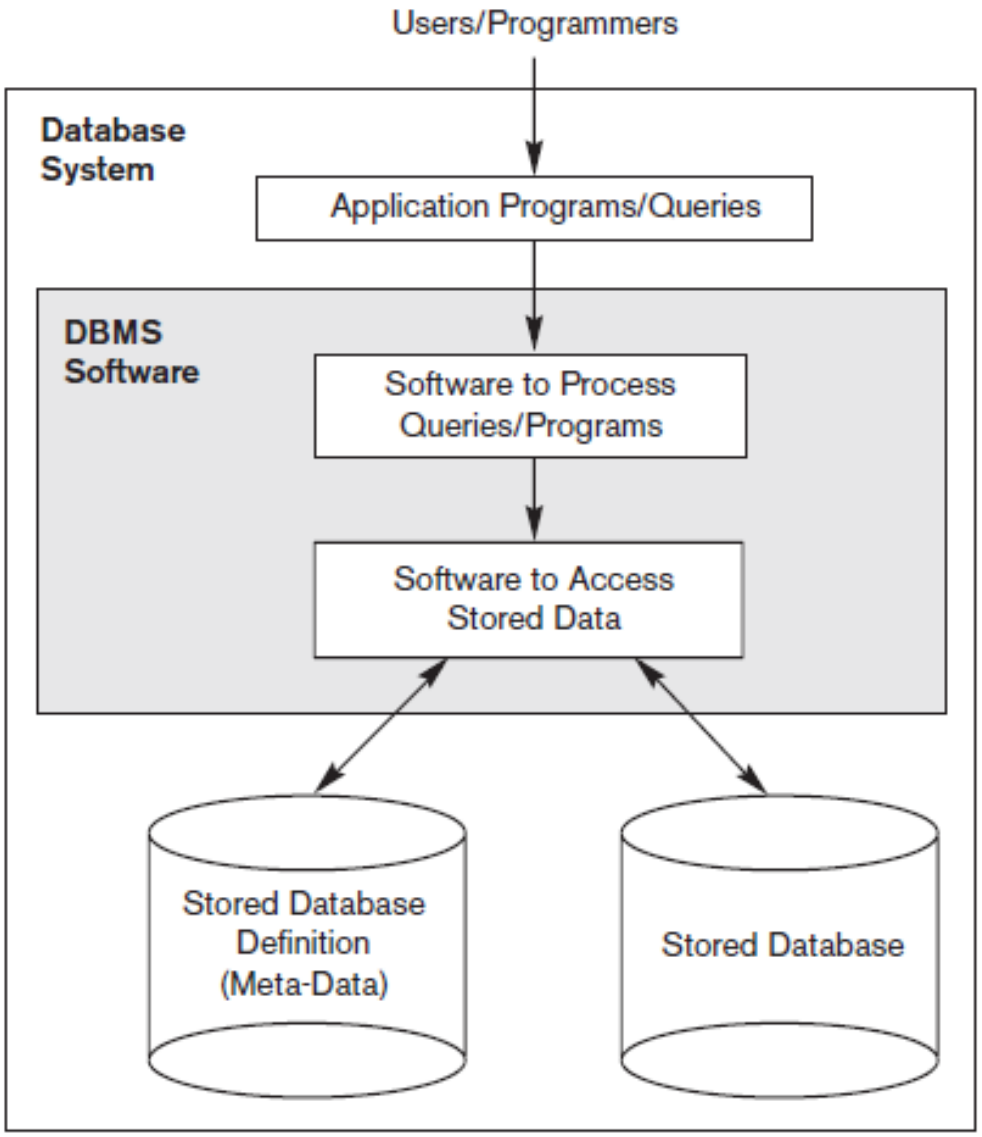


Figure: ***A simplified Database System Environment***

- **It is a computerized record-keeping system**
- **Supports operations**
  - Add or delete files to the database
  - Insert, retrieve, remove, or change data in database
- **Components**
  - Data, hardware, software, users

Components of a Database System

- **Data**
  - May support single or many users
  - Many users in organizations
    - Data is integrated
    - Data is shared
  - Different users will require different views

- **Hardware**
  - Data is stored on Disk
  - Data operated on in main memory

Software

- Database server
- Database management system (DBMS)
- DBMS provided by specific vendor
- DBMS is not (but may come with)
  - Application Development Tools
  - Application Software
  - Report Writer
  - System utilities

Users

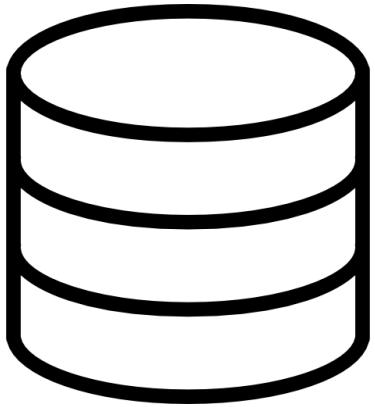
- Application Programmers
- End Users
- Database Administrators

DATABASE

→ Database is an organized collection of data which may be classified according to their organizational approach such as distributed database, object oriented database, relational database, etc.

→ Database consists of following four components:

- ❑ Data items
- ❑ Relationships
- ❑ Constraints (i.e. criteria)
- ❑ Schema (plan of data)



Data Models for Data Base

→ A collection of tools for describing
- ❑ Data
- ❑ Data relationships
- ❑ Data semantics
- ❑ Data constraints

→ Relational model

→ Entity-Relationship data model (mainly for database design)

→ Object-based data models (Object-oriented and Object-relational)

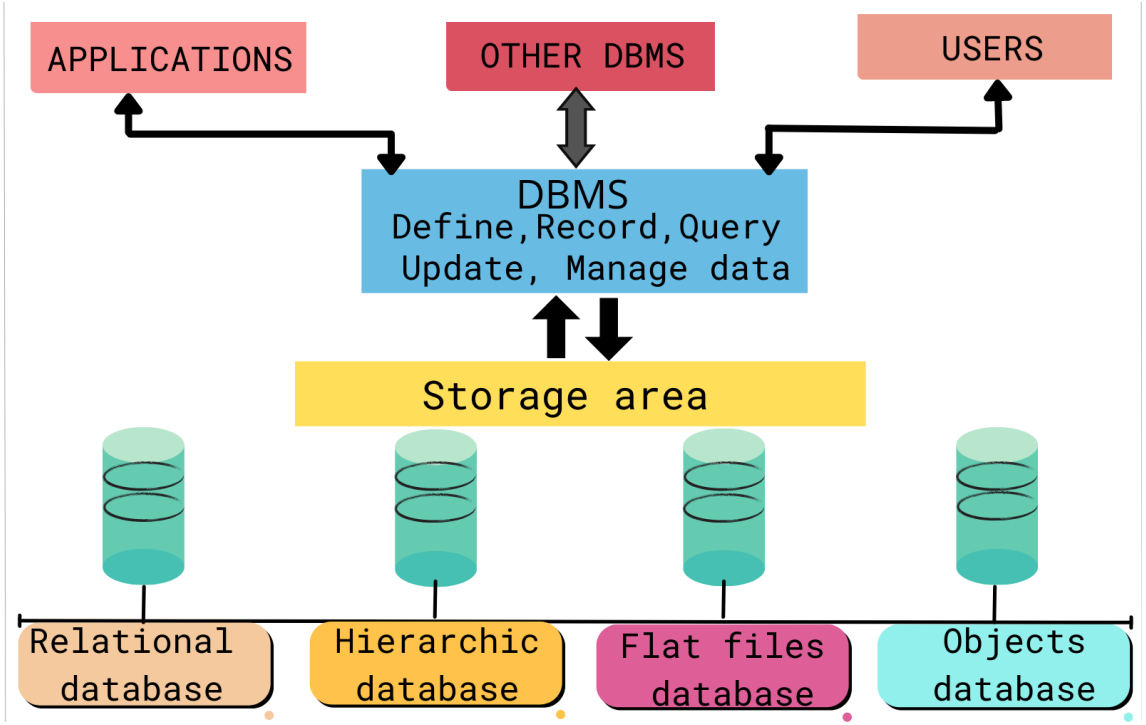
→ Semi-structured data model (XML)

→ Other older models:
- ❑ Network model
- ❑ Hierarchical model

DATABASE MANAGEMENT SYSTEM(DBMS)

→ It is a computer software program that allows a user to perform a database function of storing, retrieving, deleting and modifying data.

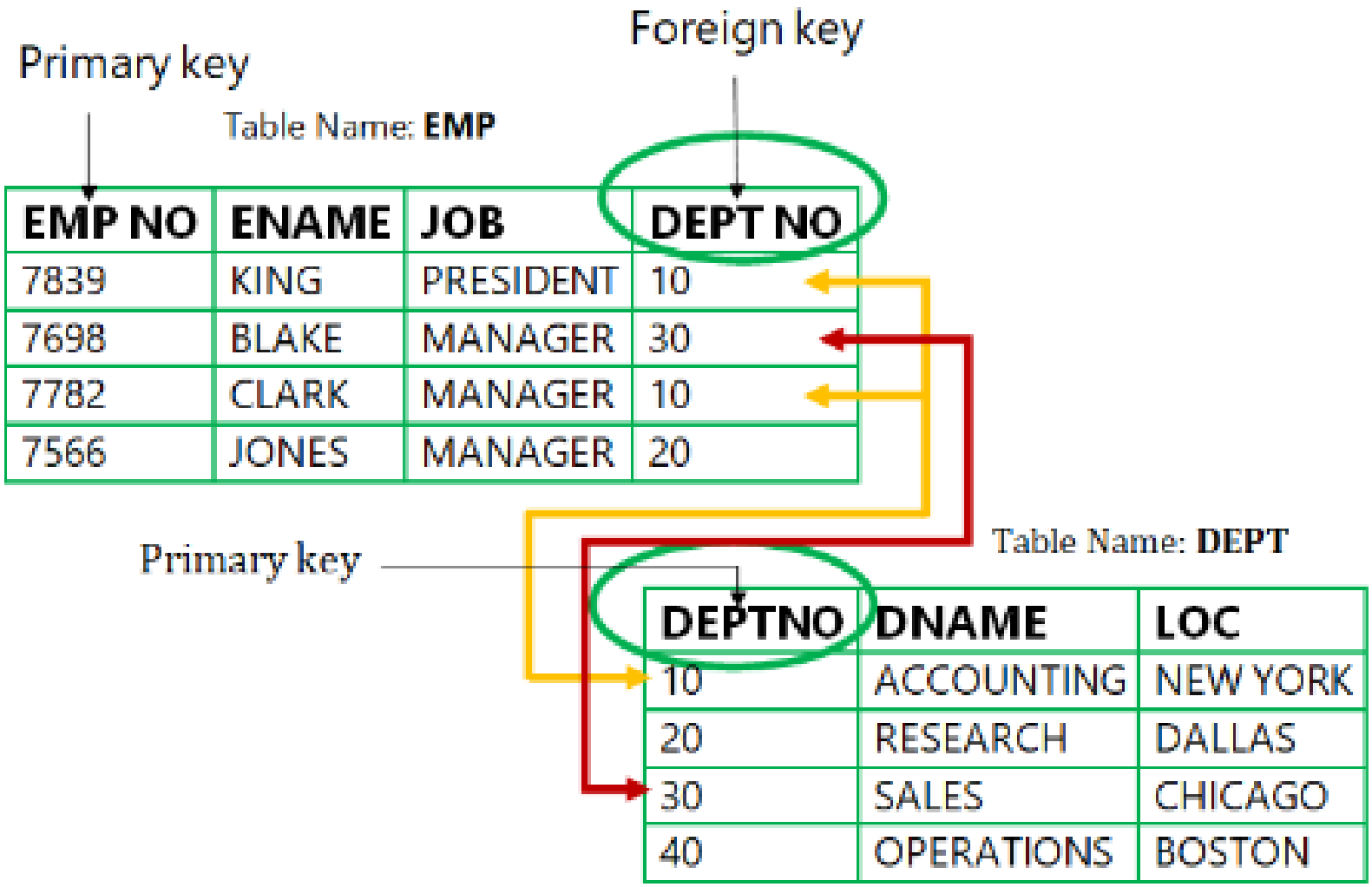
→ DBMS is the interface between the users and the database. Example of DBMS is MYSQL, Oracle, MS-SQL, IBM DB2, Microsoft Access, Microsoft Excel, SAP Sybase ASE etc.



Relational Database Management System (RDBMS)

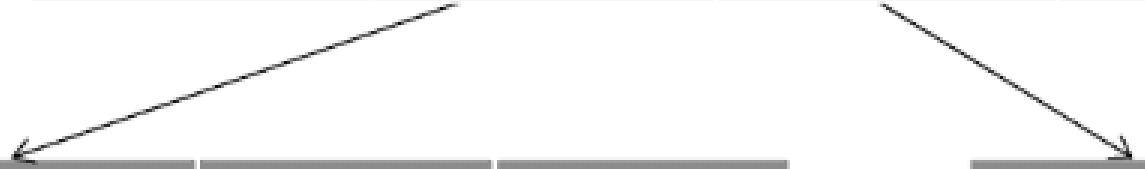
- → RDBMS stands for Relational Database Management System.
- → All modern database management systems like SQL, MS SQL Server, IBM DB2, ORACLE, My-SQL, and Microsoft Access are based on RDBMS.
- → It is called Relational Database Management System (RDBMS) because it is based on the relational model introduced by E.F. Codd.
- → Data is represented in terms of tuples (rows) in RDBMS.
- → A relational database is the most commonly used database. It contains several tables, and each table has its primary key.
- → Due to a collection of an organized set of tables, data can be accessed easily in RDBMS.

Relating Multiple Table in RDBMS



Purchase table

Transaction ID	Customer ID	Product ID	Purchange date
1112	24221	8977	03-22-2010
1113	24222	8978	03-22-2010
1114	24223	8979	03-22-2010



Customer ID	Customer	Address
24221	Bob	123 East street
24222	Alice	223 Main street
24223	Martha	465 North street

Customer table

Product ID	Name	Price
8977	Banana	.79
8978	TV	400
8979	Watch	50

Product table

Database Applications

- ***Banking***: transactions
- ***Airlines***: reservations, schedules
- ***Universities***: registration, grades
- ***Sales***: customers, products, purchases
- ***Online retailers***: order tracking, customized recommendations
- ***Manufacturing***: production, inventory, orders, supply chain
- ***Human resources***: employee records, salaries, tax deductions

PERSONS BEHIND THE SCENES

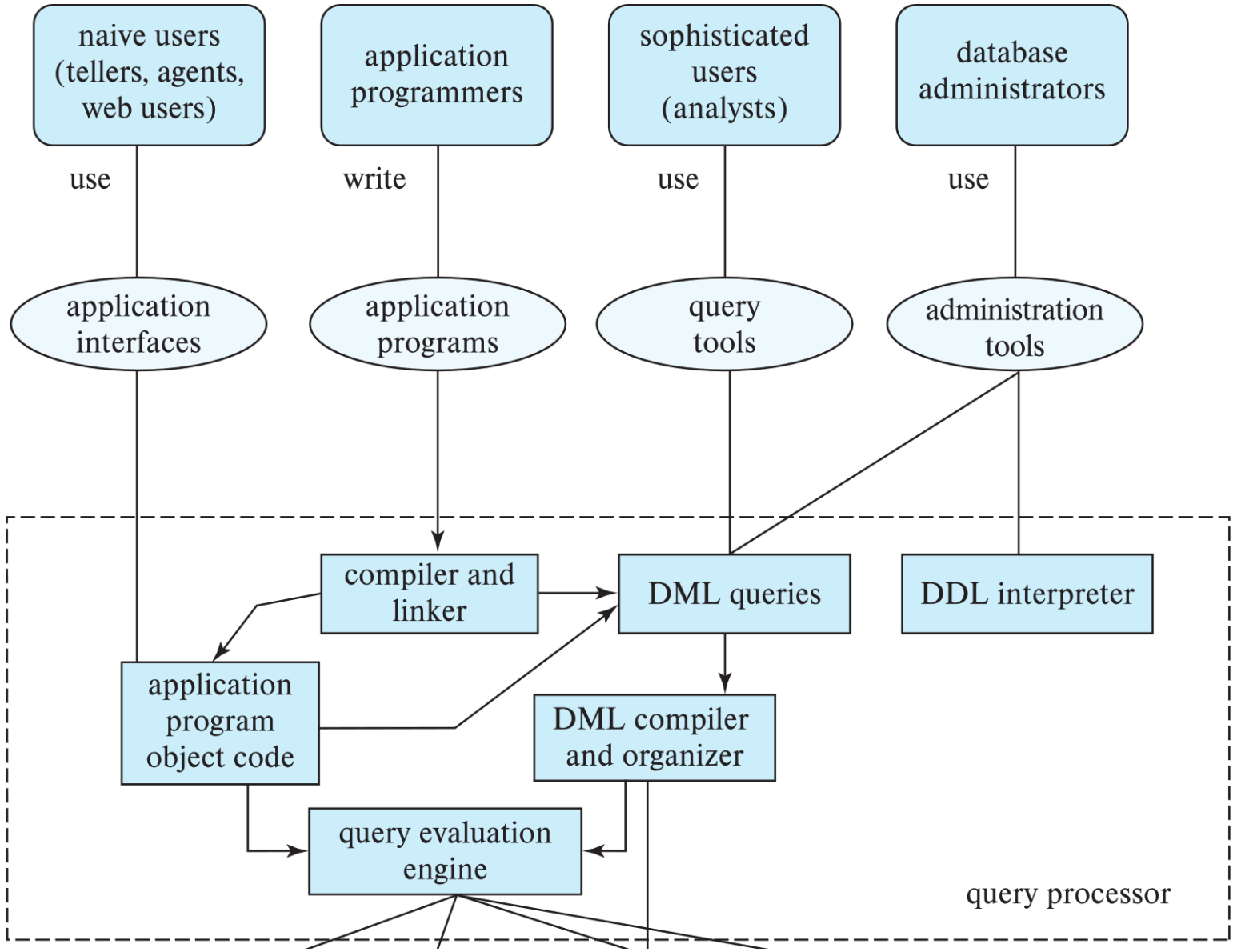
- Persons involved in design, development, operation, and maintenance of the DBMS software and system environment.
  - **DBMS designers and implementers:**

    Design and implement the DBMS software package itself.
  - **Tool developers**:

    Design and implement tools that facilitate the use of the DBMS software. Tools include design tools, performance tools, special interfaces, etc.
  - **Operators and maintenance personnel**:

    Work on running and maintaining the hardware and software environment for the database system.

Database Users and Administrators



→ The people who are accessing or working with the database are called database users and administrators.

Database Users:

→ Users are differentiated by the way they expect to interact with the system

- ***Application programmers*** – interact with system through DML calls
- ***Sophisticated users*** – form requests in a database query language
- ***Specialized users*** – write specialized database applications that do not fit into the traditional data processing framework
- ***Naïve users*** – invoke one of the permanent application programs that have been written previously. Examples: people accessing database over the web, bank tellers, clerical staff

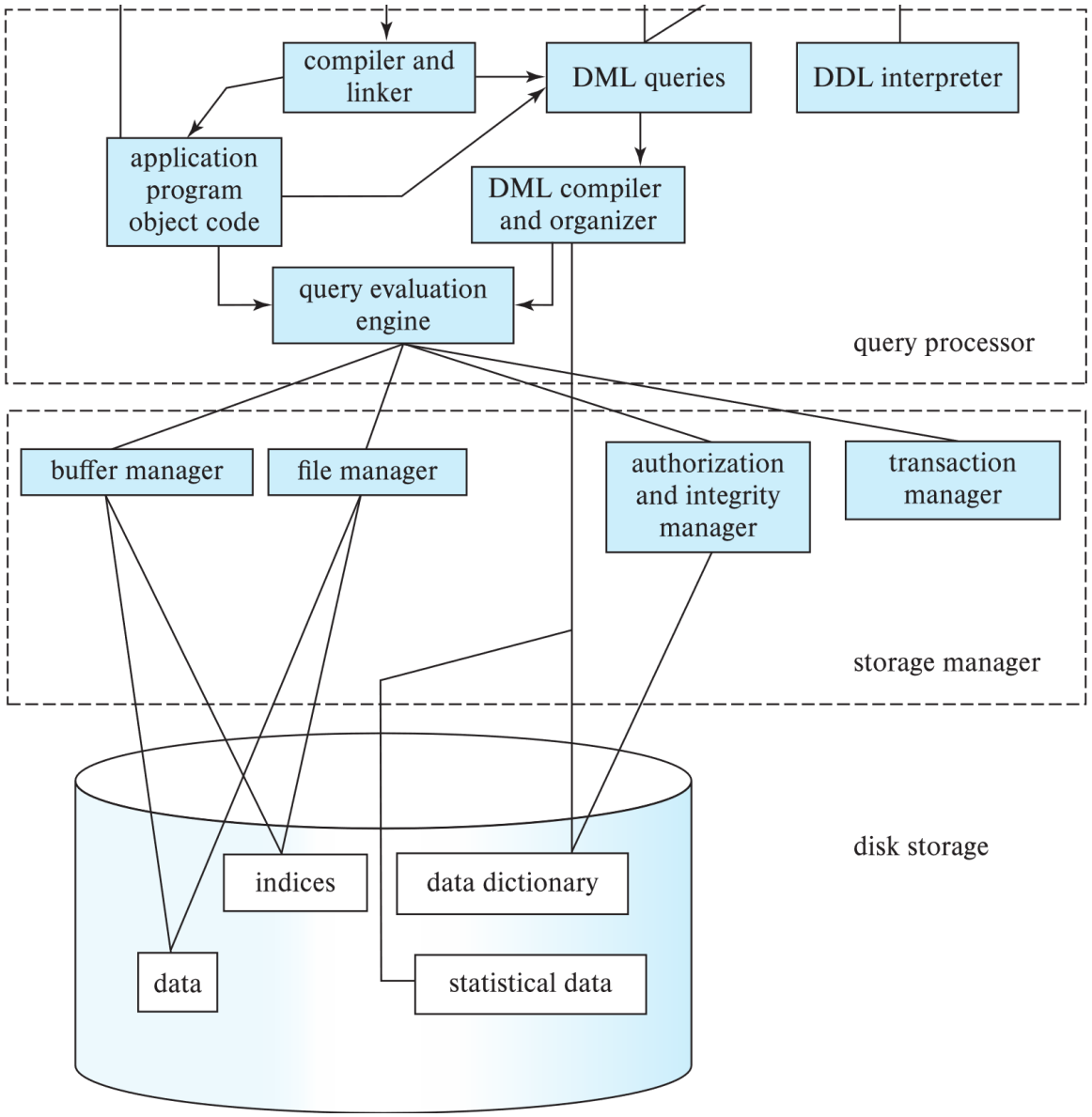
Database Administrators:

→ A person who has central control over the system is called a database administrator (DBA).

→ Functions of a DBA include:

- Schema definition(Schema-logical view of database)
- Storage structure and access-method definition
- Schema and physical-organization modification
- Granting of authorization for data access
- Routine maintenance
- Periodically backing up the database
- Ensuring that enough free disk space is available for normal operations, and upgrading disk space as required
- Monitoring jobs running on the database

Database Architecture



→ Centralized databases
  - One to a few cores, shared memory

→ Client-server,
  - One server machine executes work on behalf of multiple client machines.

→ Parallel databases
  - Many core shared memory
  - Shared disk
  - Shared nothing

→ Distributed databases
  - Geographical distribution
  - Schema/data heterogeneity

DATABASE APPLICATIONS

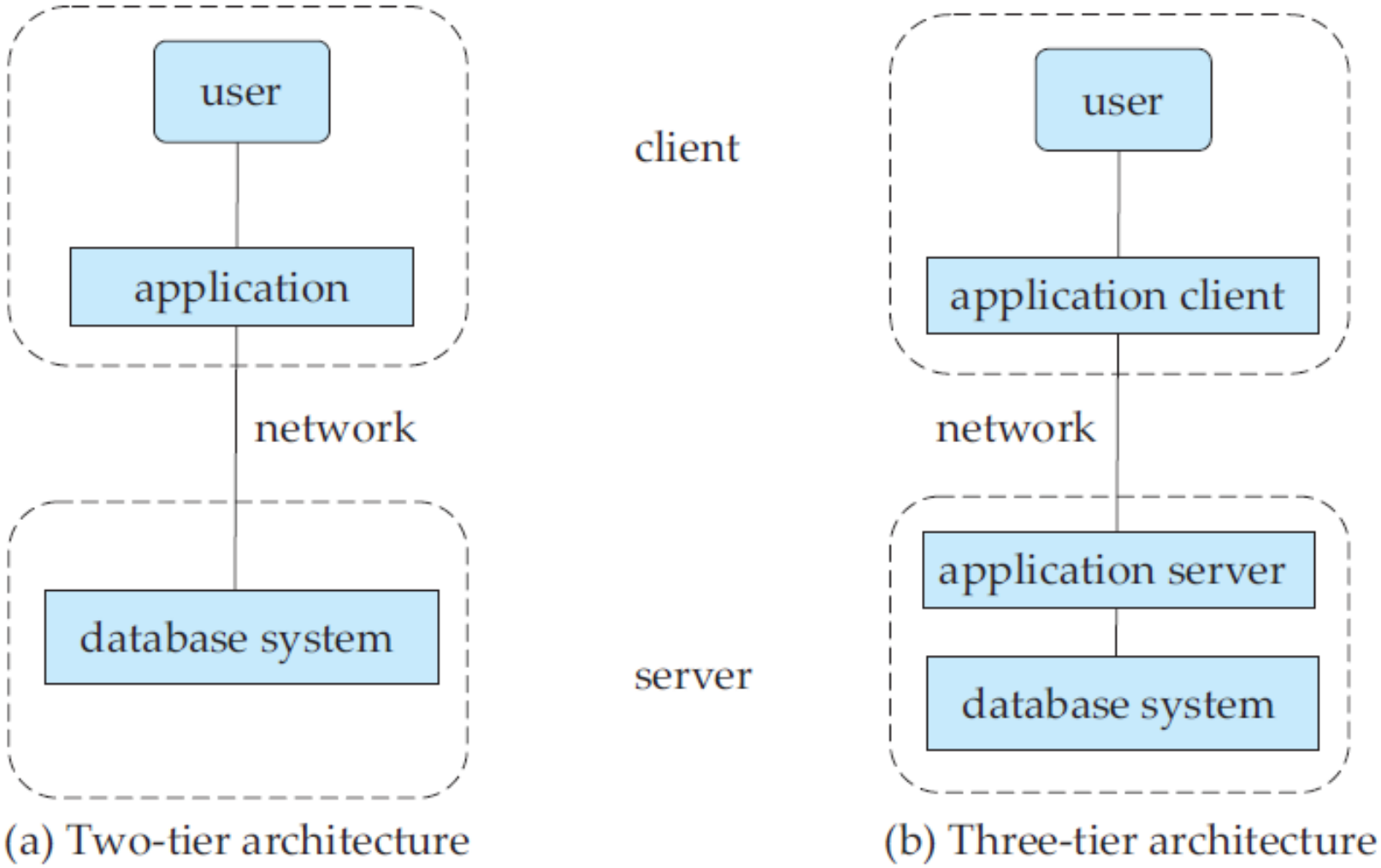
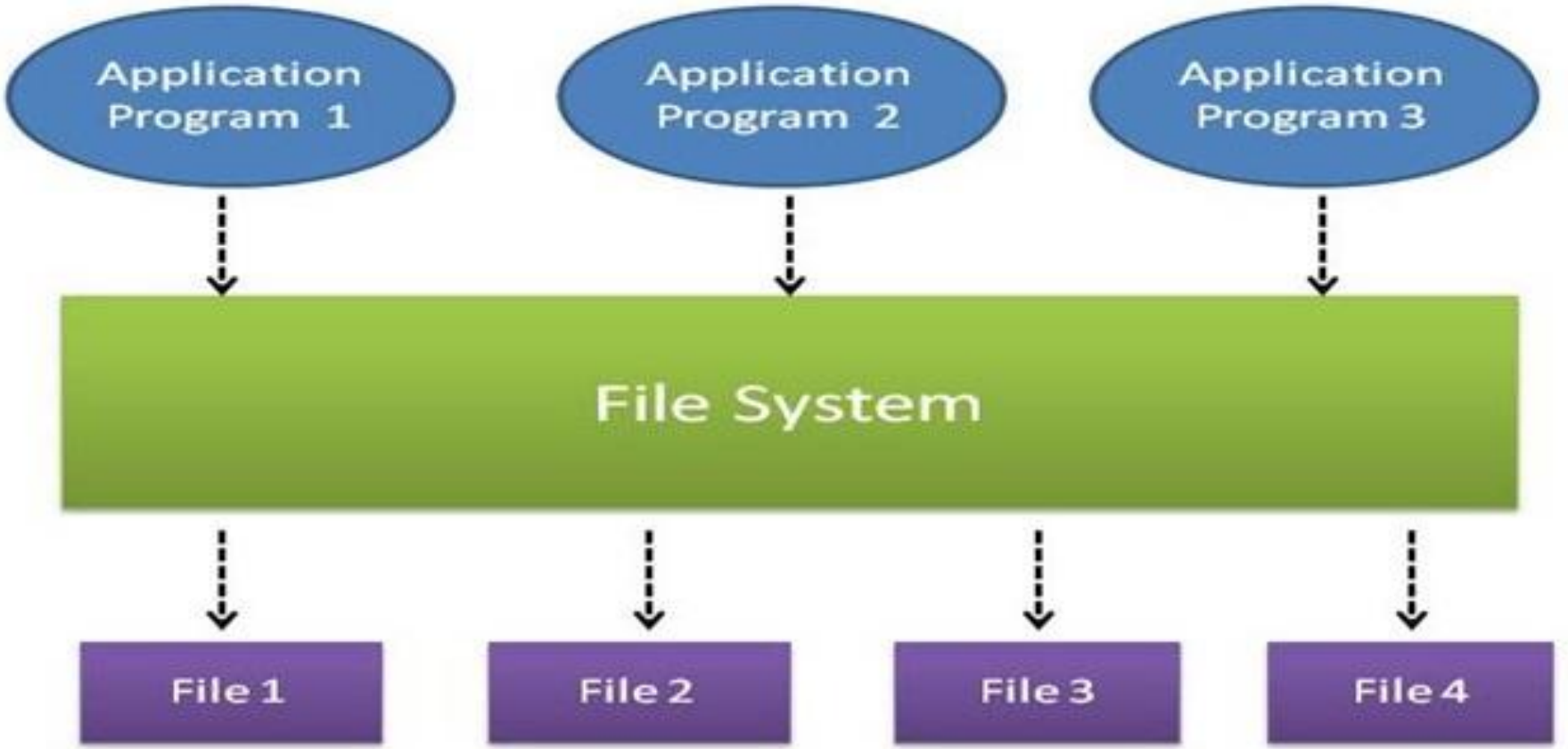


Figure: ***Two-Tier and Three-Tier Architecture for Database Applications***

File Management System



Secondary Storage

A:\>

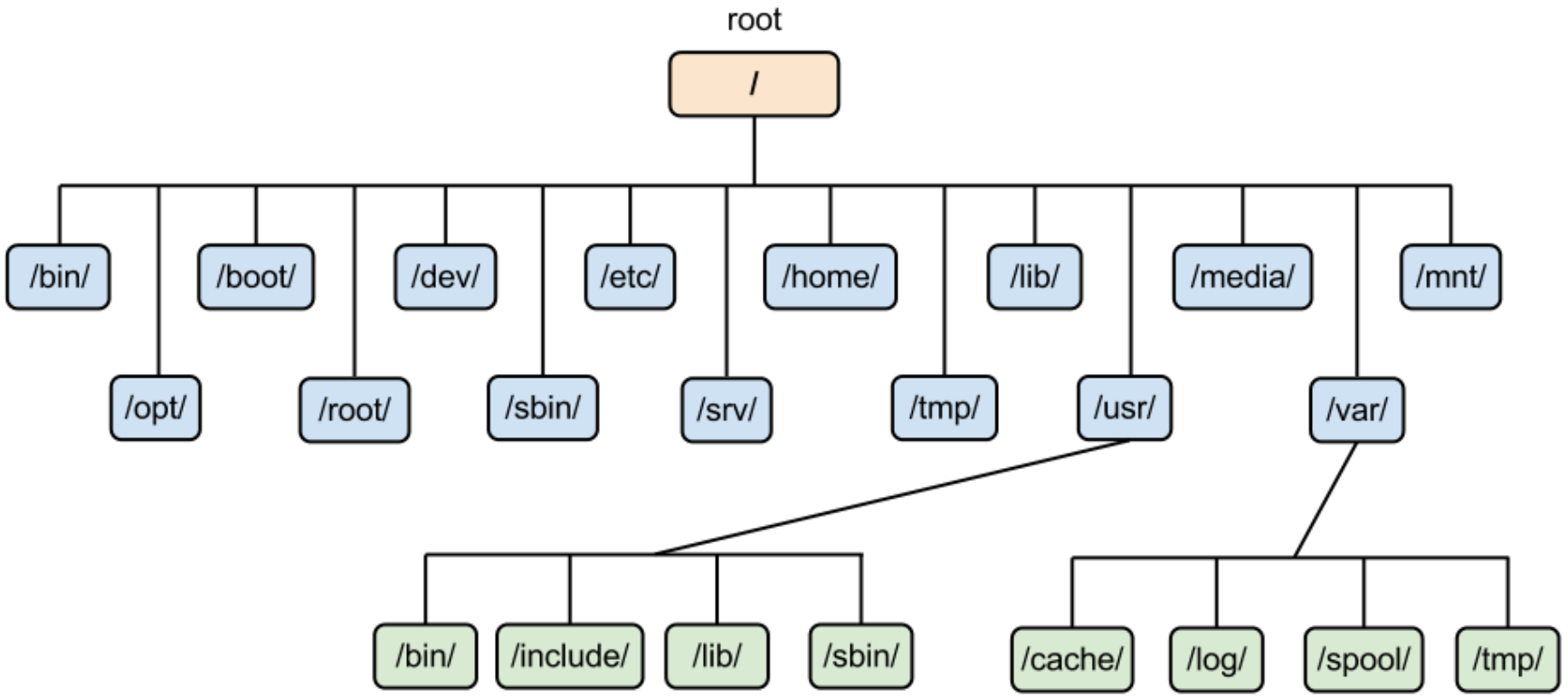
B:\>

C:\>

D:\>

Directories

Files


root
/
/bin/
/boot/
/dev/
/etc/
/home/
/lib/
/media/
/mnt/
/opt/
/root/
/sbin/
/srv/
/tmp/
/usr/
/var/
/bin/
/include/
/lib/
/sbin/
/cache/
/log/
/spool/
/tmp/

Drawbacks of Using File System to Store Data

In the early days, database applications were built directly on top of file systems, which leads to:

- ***Data redundancy and inconsistency***: data is stored in multiple file formats resulting in duplication of information in different files
- ***Difficulty in accessing data***
  - Need to write a new program to carry out each new task
- ***Data isolation***
  - Multiple files and formats
- ***Integrity problems***
  - Integrity constraints (e.g., account balance > 0) become "buried" in program code rather than being stated explicitly
  - Hard to add new constraints or change existing ones

- ***Atomicity of updates***
  - Failures may leave database in an inconsistent state with partial updates carried out
  - Example: Transfer of funds from one account to another should either complete or not happen at all
- ***Concurrent access by multiple users***
  - Concurrent access needed for performance
  - Uncontrolled concurrent accesses can lead to inconsistencies
    - Ex: Two people reading a balance (say 100) and updating it by withdrawing money (say 50 each) at the same time
- ***Security problems***
  - Hard to provide user access to some, but not all, data

Database systems offer solutions to all the above problems

Advantages of Using DBMS

→ To reduce redundancy
→ To avoid inconsistency
→ To share data
→ To provide support for transactions
→ To maintain integrity
→ To enforce security
→ To provide efficient backup and recovery
→ To concurrently access database

Disadvantages of Using DBMS

→ Problems associated with centralization
→ Increased Costs
→ Complexity of backup and recovery etc.

History of Database Systems

- 1950s and early 1960s:
  - Data processing using magnetic tapes for storage
    - Tapes provided only sequential access
  - Punched cards for input
- Late 1960s and 1970s:
  - Hard disks allowed direct access to data
  - Network and hierarchical data models in widespread use
  - Ted Codd defines the relational data model
    - Would win the ACM Turing Award for this work
    - IBM Research begins System R prototype
    - UC Berkeley (Michael Stonebraker) begins Ingres prototype
    - Oracle releases first commercial relational database
  - High-performance (for the era) transaction processing

- 1980s:
  - Research relational prototypes evolve into commercial systems
    - SQL becomes industrial standard
  - Parallel and distributed database systems
    - Wisconsin, IBM, Teradata
  - Object-oriented database systems
- 1990s:
  - Large decision support and data-mining applications
  - Large multi-terabyte data warehouses
  - Emergence of Web commerce

- 2000s
  - Big data storage systems
    - Google BigTable, Yahoo PNuts, Amazon,
    - “NoSQL” systems.
  - Big data analysis: beyond SQL
    - Map reduce and friends
- 2010s
  - SQL reloaded
    - SQL front end to Map Reduce systems
    - Massively parallel database systems
    - Multi-core main-memory databases

Thank You !!!