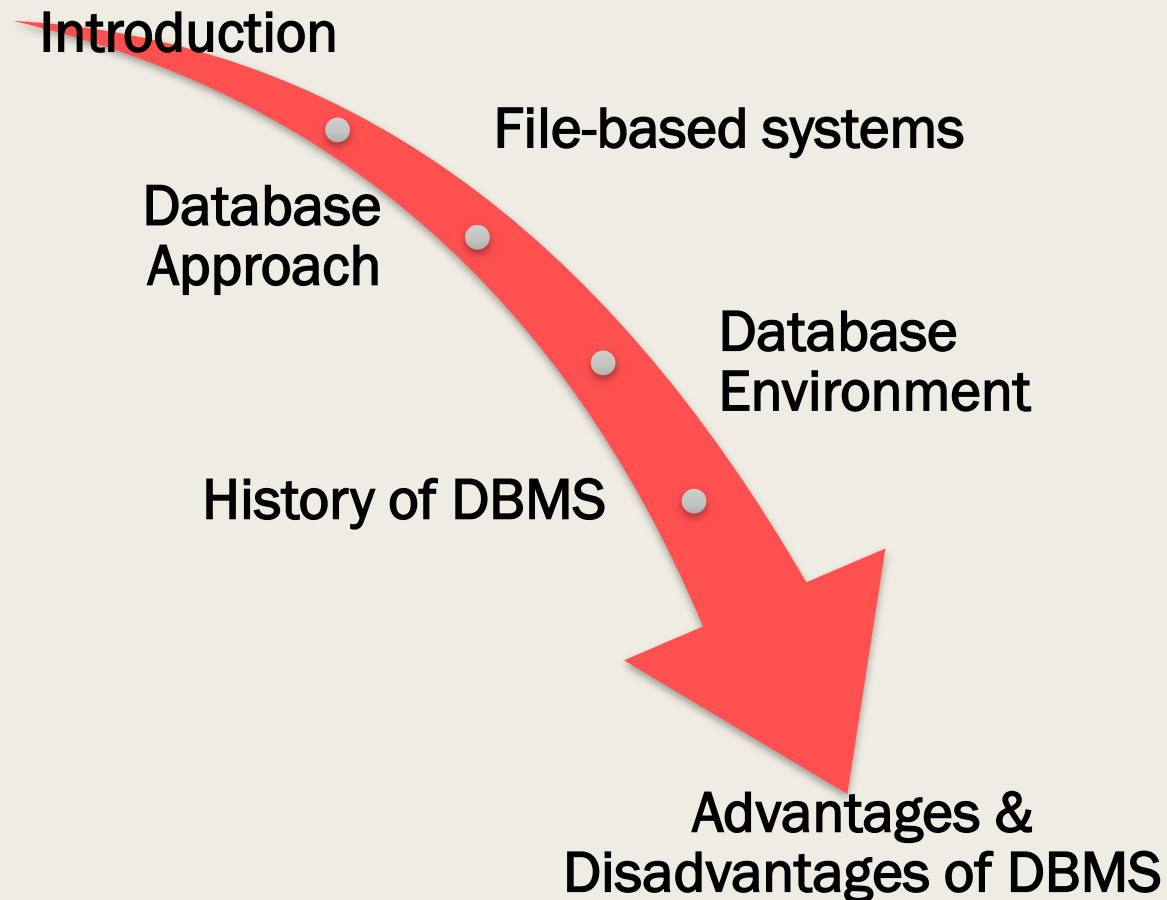


ICT502/ ITS571
LECTURE 1

**INTRODUCTION TO
DATABASE
&
DATABASE ENVIRONMENT**

Faiqah Hafidzah Halim
Muhammad Hamiz Mohd Radzi

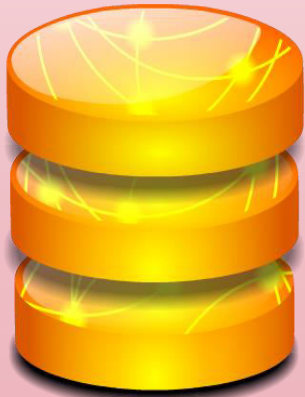
Lecture Content



objectives

- At the end of this lesson, students should be able to:
 - *Describe database approach*
 - *Differentiate database approach and file based system*
 - *Explain the database environment*
 - *Explain the history of DBMS*
 - *Describe the advantages and disadvantages of DBMS*

Introduction to Database



ORACLE

Microsoft®
Access®

MySQL®



Database
Collection of
related data

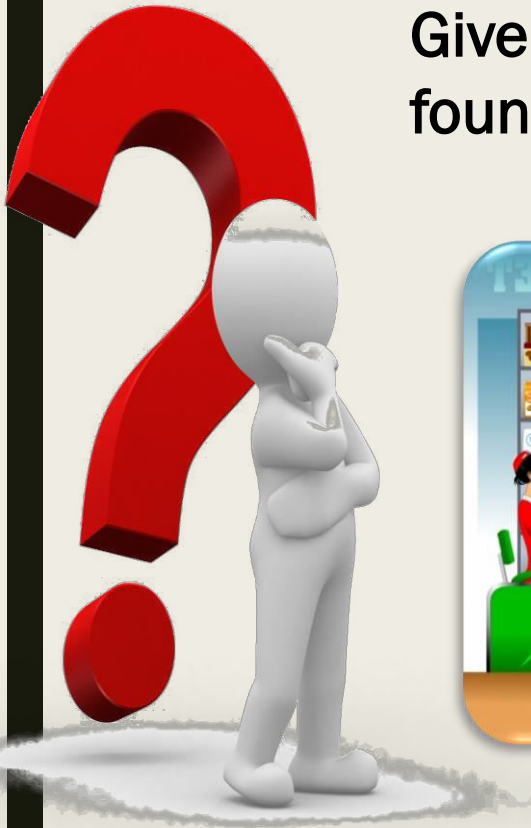
**Database
Management System
(DBMS)**
Software that
manages and controls
access to the
database

**Database
Application**
A program that
interacts with the
database

Database System
A collection of application
programs that interact with
the database along with the
DBMS and the database
itself

Introduction to Database

Give examples of database systems that can be found in everyday life?

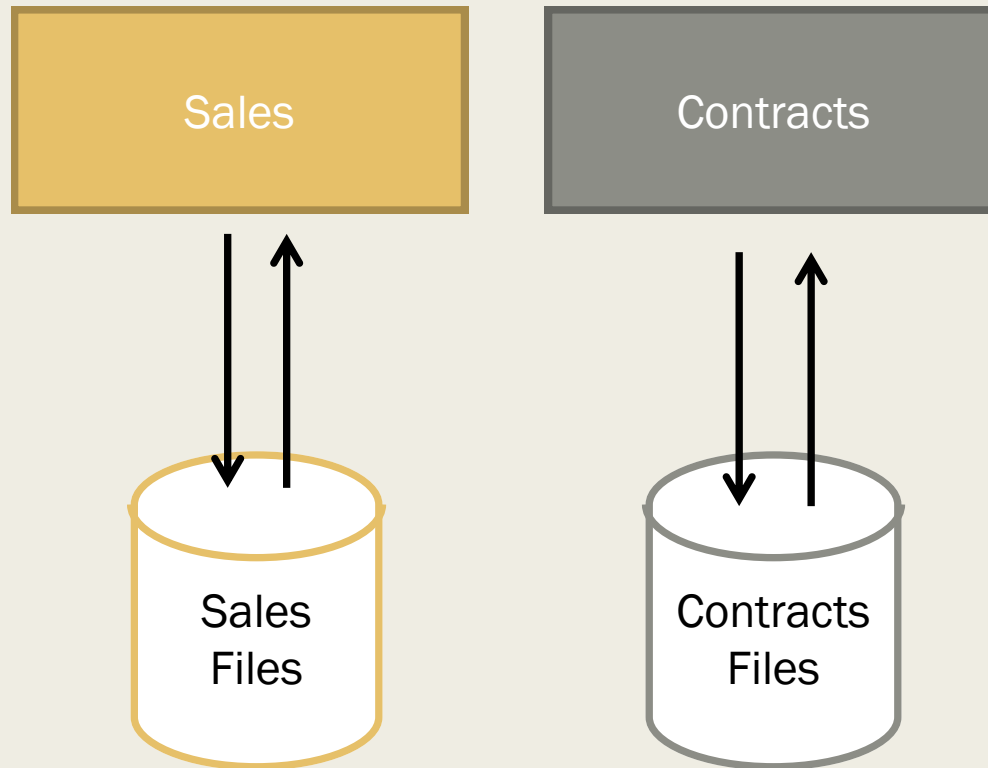


Traditional File-Based Systems

- ❏ Collection of application programs that perform services for the end users (e.g. reports).
- ❏ Each program defines and manages its own data.
- ❏ FBS developed for more efficient data access
- ❏ Decentralized approach for FBS :

Each department with the assistance of Data Processing staff, stored and controlled their own data

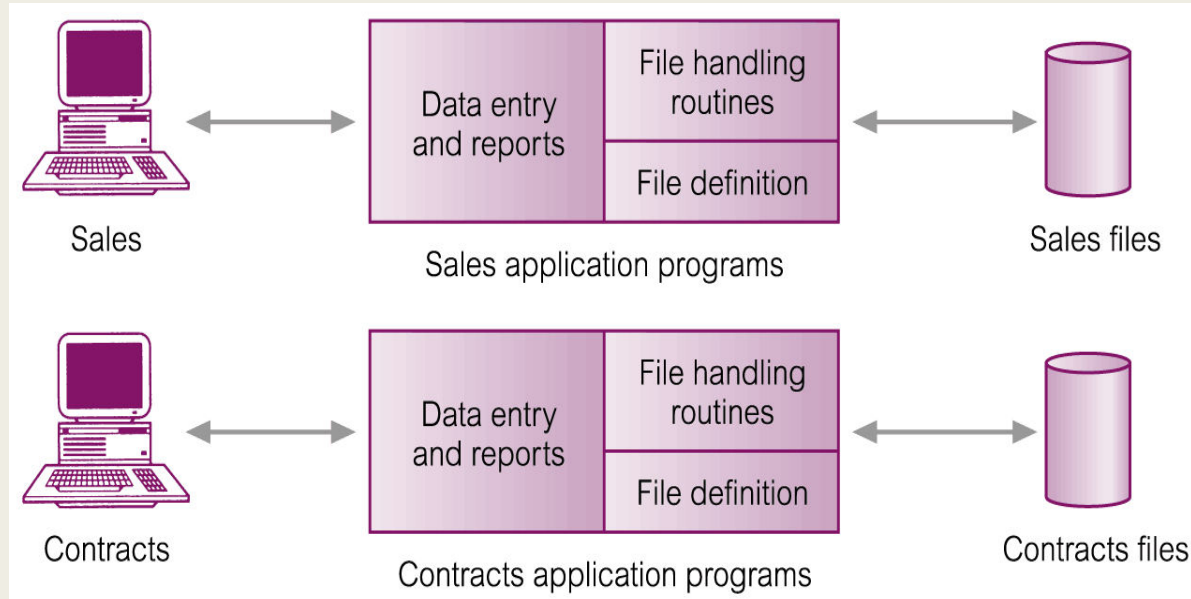
Traditional File-Based Systems



File Based Processing

Figure 1.5

File-based processing.



Similar data

Sales Files

PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)

PrivateOwner (ownerNo, fName, lName, address, telNo)

Client (clientNo, fName, lName, address, telNo, prefType, maxRent)

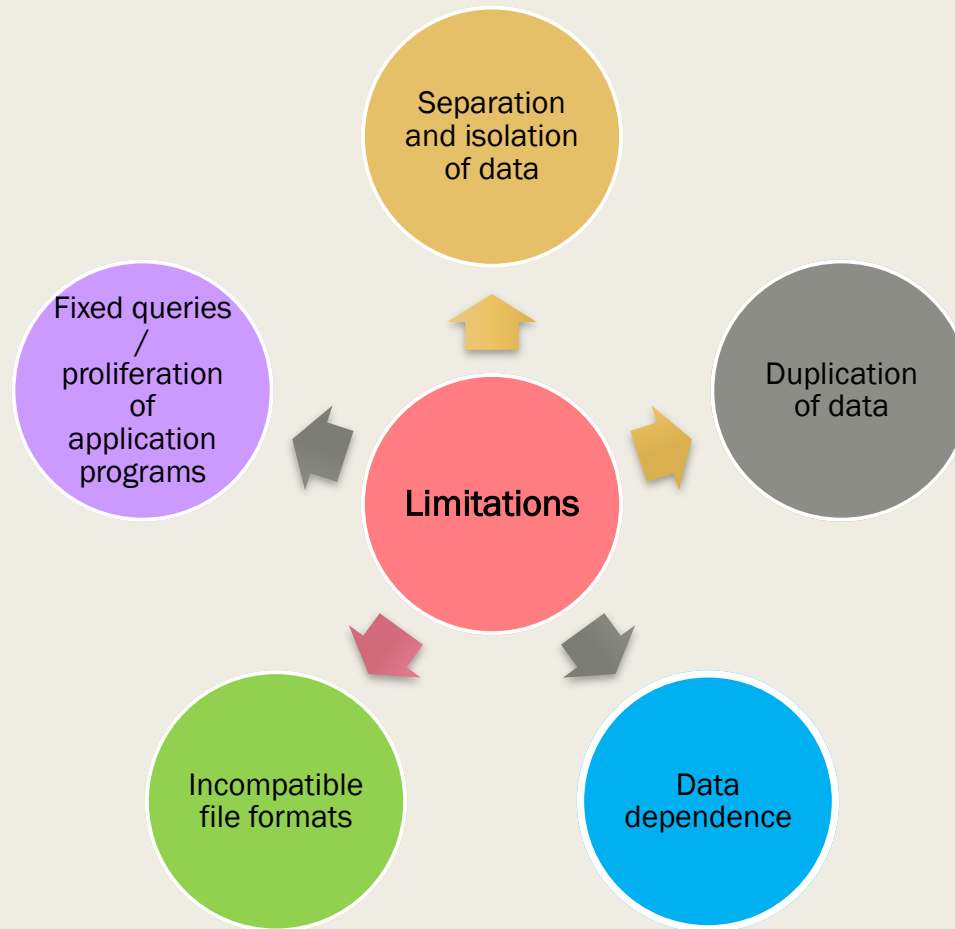
Contracts Files

Lease (leaseNo, propertyNo, clientNo, rent, paymentMethod, deposit, paid, rentStart, rentFinish, duration)

PropertyForRent (propertyNo, street, city, postcode, rent)

Client (clientNo, fName, lName, address, telNo)

Limitations of File Based Approach



Database

1

Definition of data was embedded in application programs, rather than being stored separately and independently.

2

No control over access and manipulation of data beyond that imposed by application programs.



atabase

M

anagement




ystem




The Database

Database

- Shared collection of logically related data (and a description of this data), designed to meet the information needs of an organization.

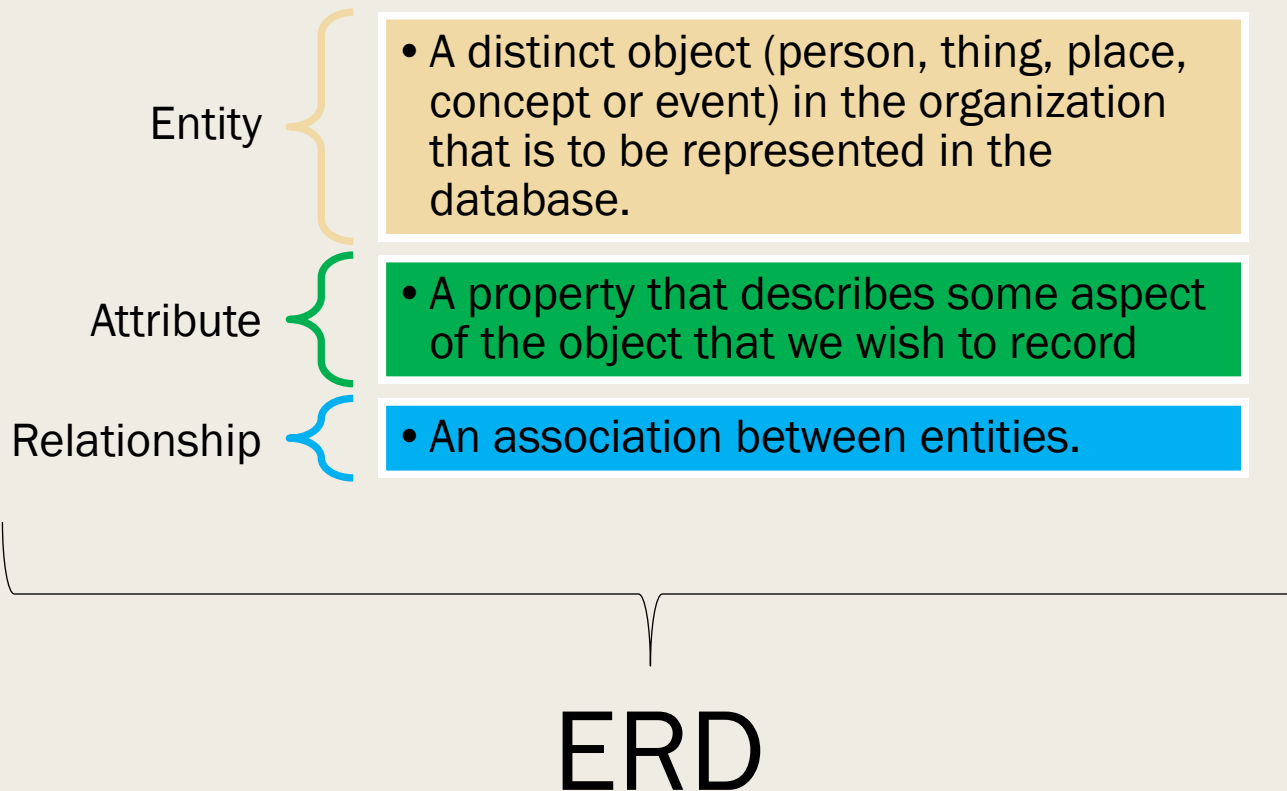
 System catalog (metadata) provides description of data to enable program-data independence.

The Database






-  Definition of data is separated from the application programs.
-  The database approach separates the structure of the data from the application programs and stores it in the database.
-  Applications programs are unaffected if the new data structures are added or existing data being modified.

The Database


- ❏ Logically related data comprises entities, attributes, and relationships of an organization's information.



Database Management System (DBMS)

-  A software system that enables users to define, create, maintain, and control access to the database.
-  Provides facilities:
 -  *Data Definition Language (DDL) allow user to define database.*
 -  *Data Manipulation Language (DML) allows user to insert, update, delete and retrieve data from database*
 -  *Provide control access (security, integrity system, concurrency control, recovery control, user accessible catalog)*

Database Application Program

-  A computer program that interacts with database by issuing an appropriate request (SQL statement) to the DBMS.

Database Application Program

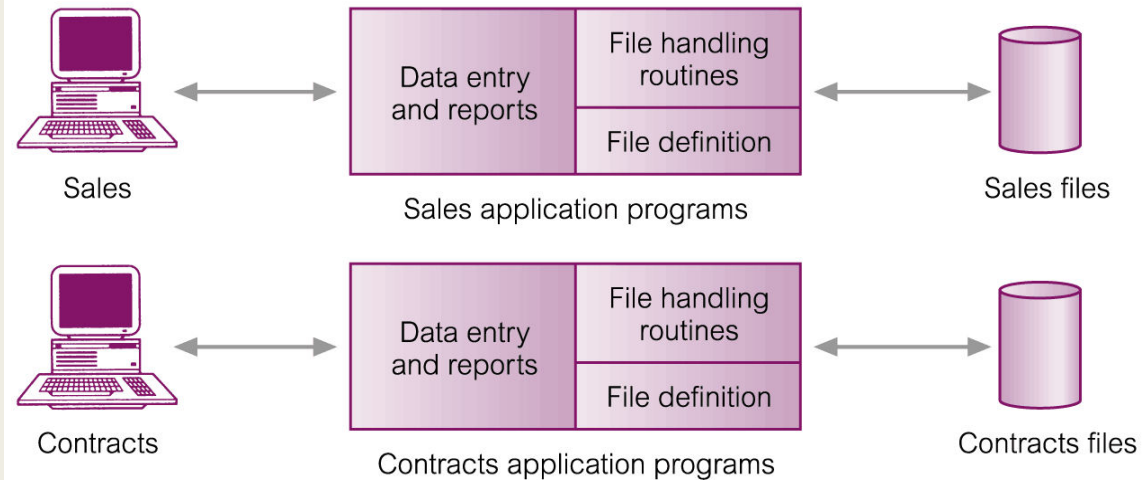
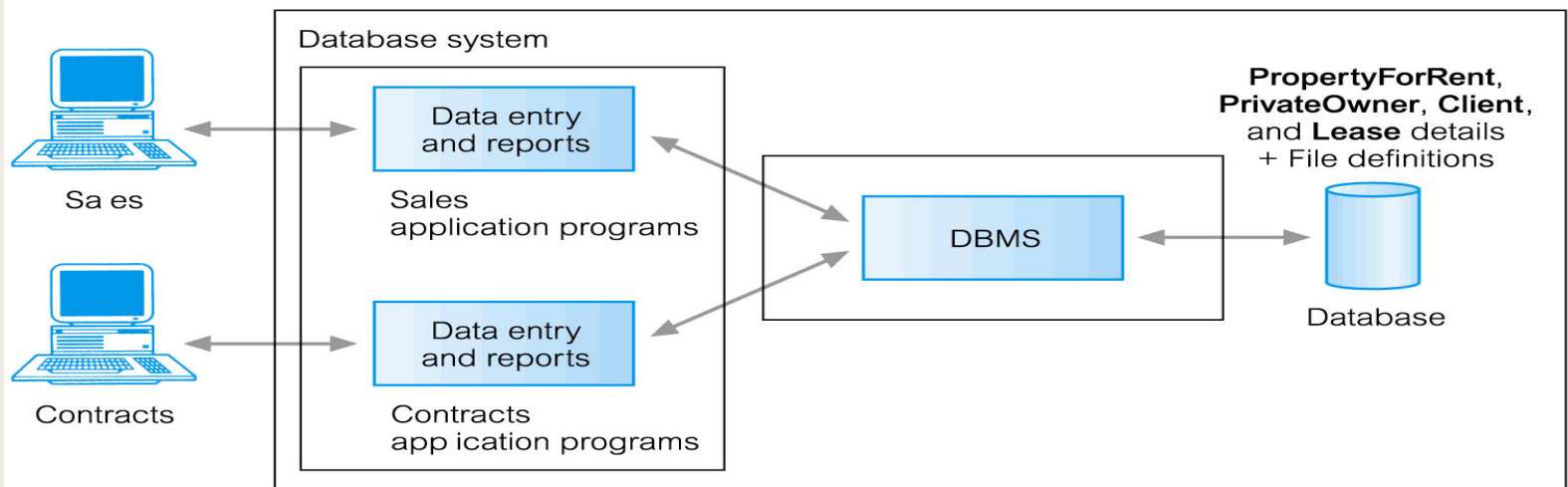





Figure 1.5
File-based processing.



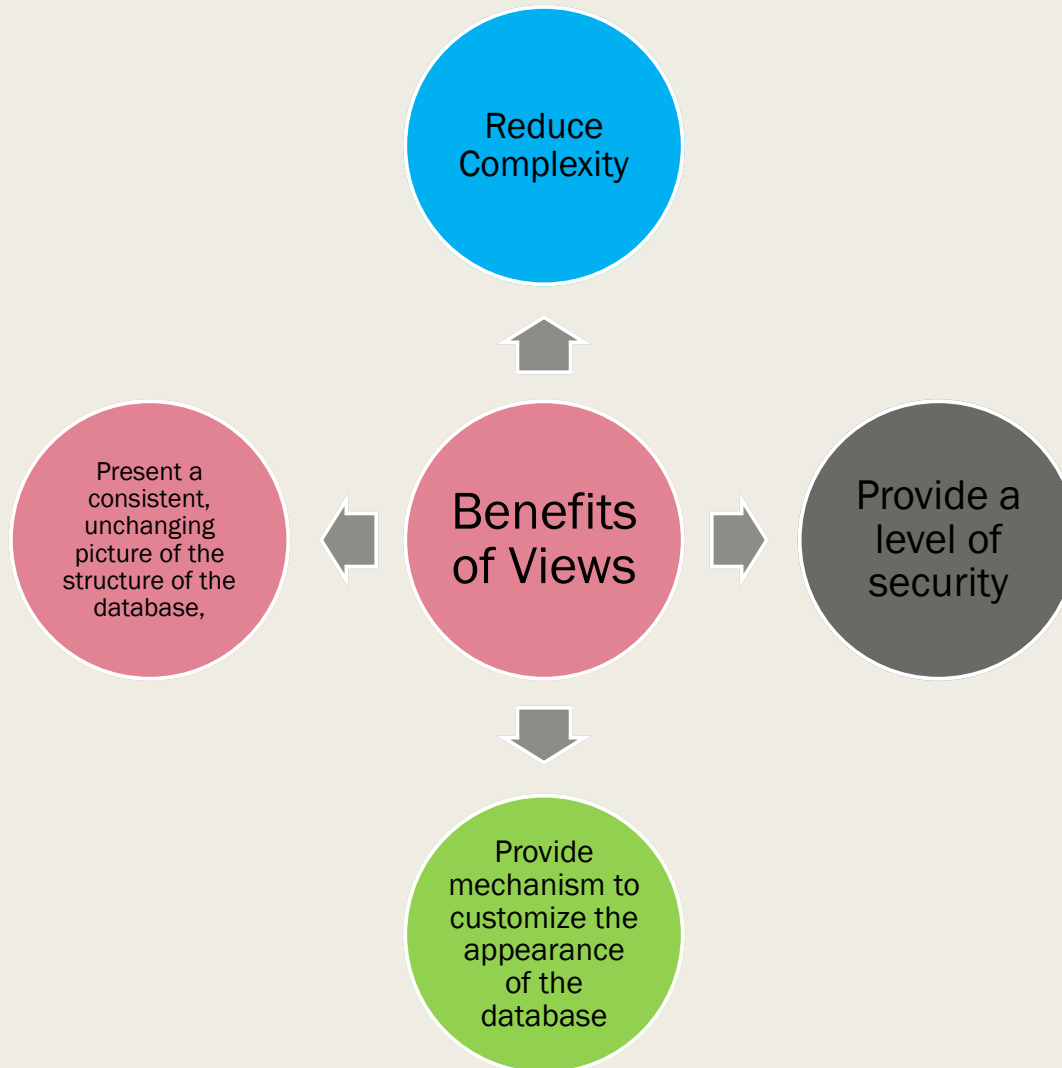
FBS

Database Approach

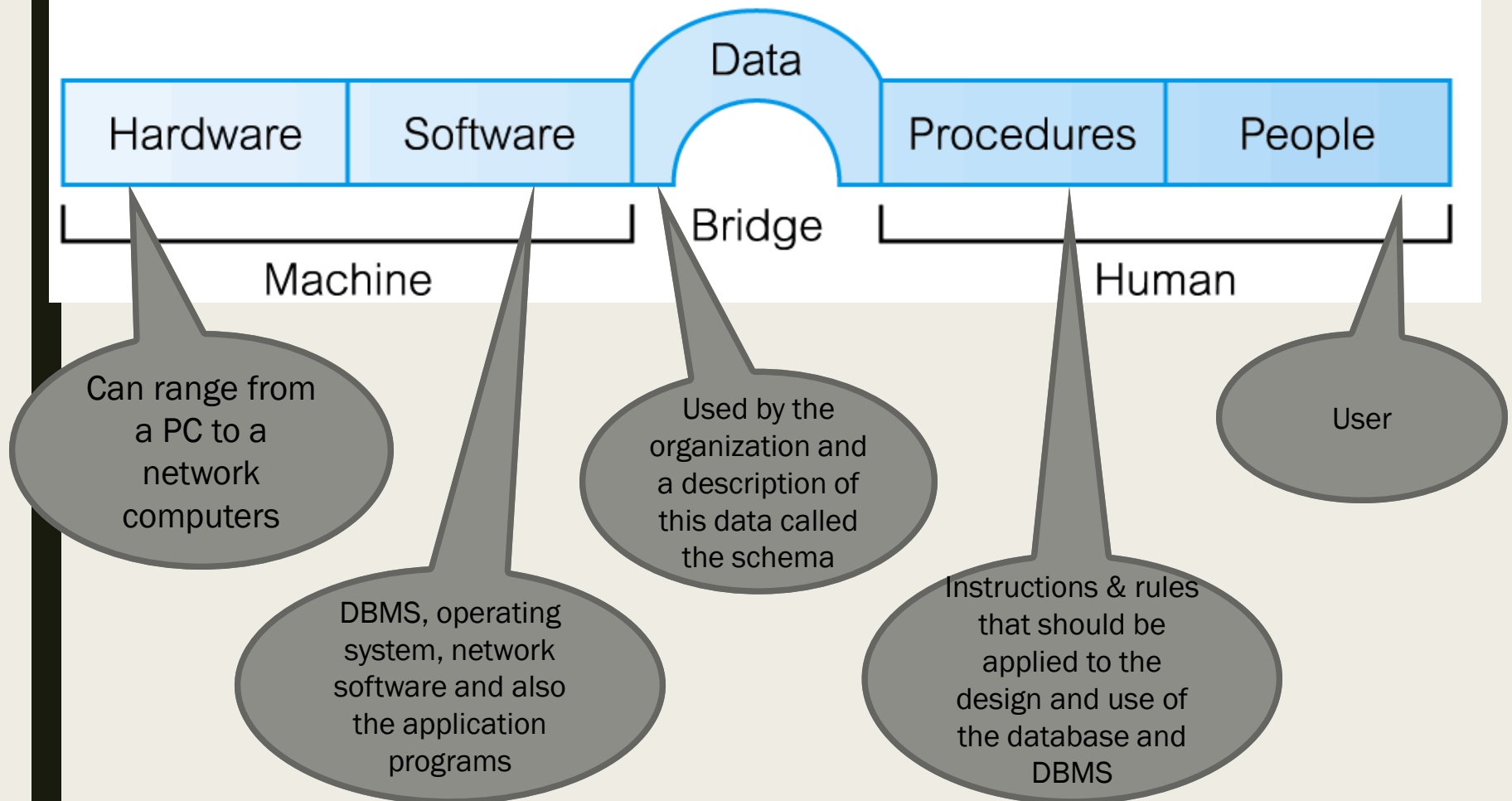
Views

-  DBMS provide view mechanism to avoid complexity.
-  View allows each user to have his or her own view of the database.
-  A view is essentially some subset of the database.






Views



Components of Database Environment



Roles in the Database Environment

-  Data Administrator (DA)
-  Database Administrator (DBA)
-  Database Designers (Logical and Physical)
-  Application Programmers
-  End Users (naïve and sophisticated)

History of Database System



1

First Generation

(Hierarchical and Network)



2

Second Generation

(Relational)

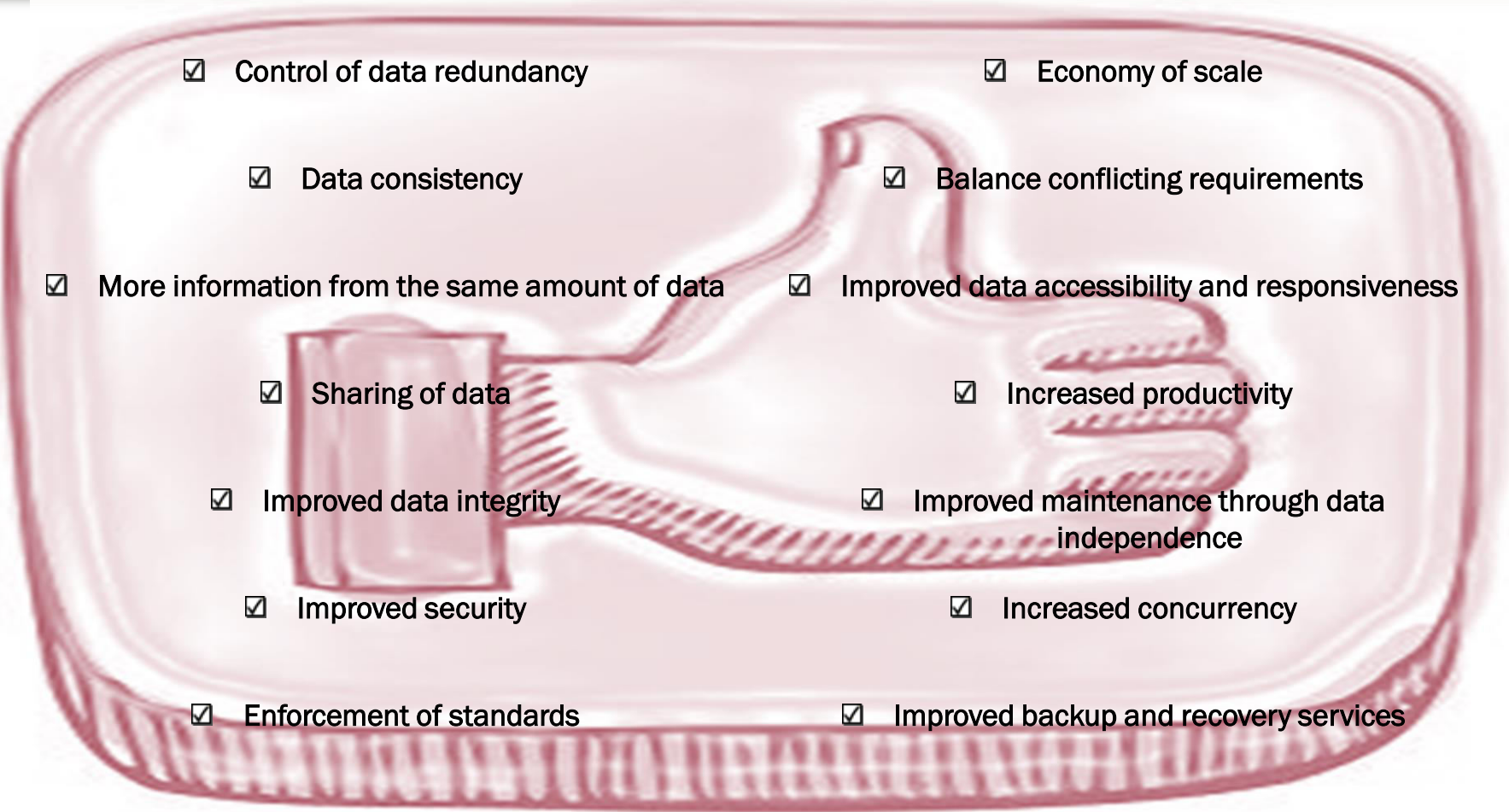


3

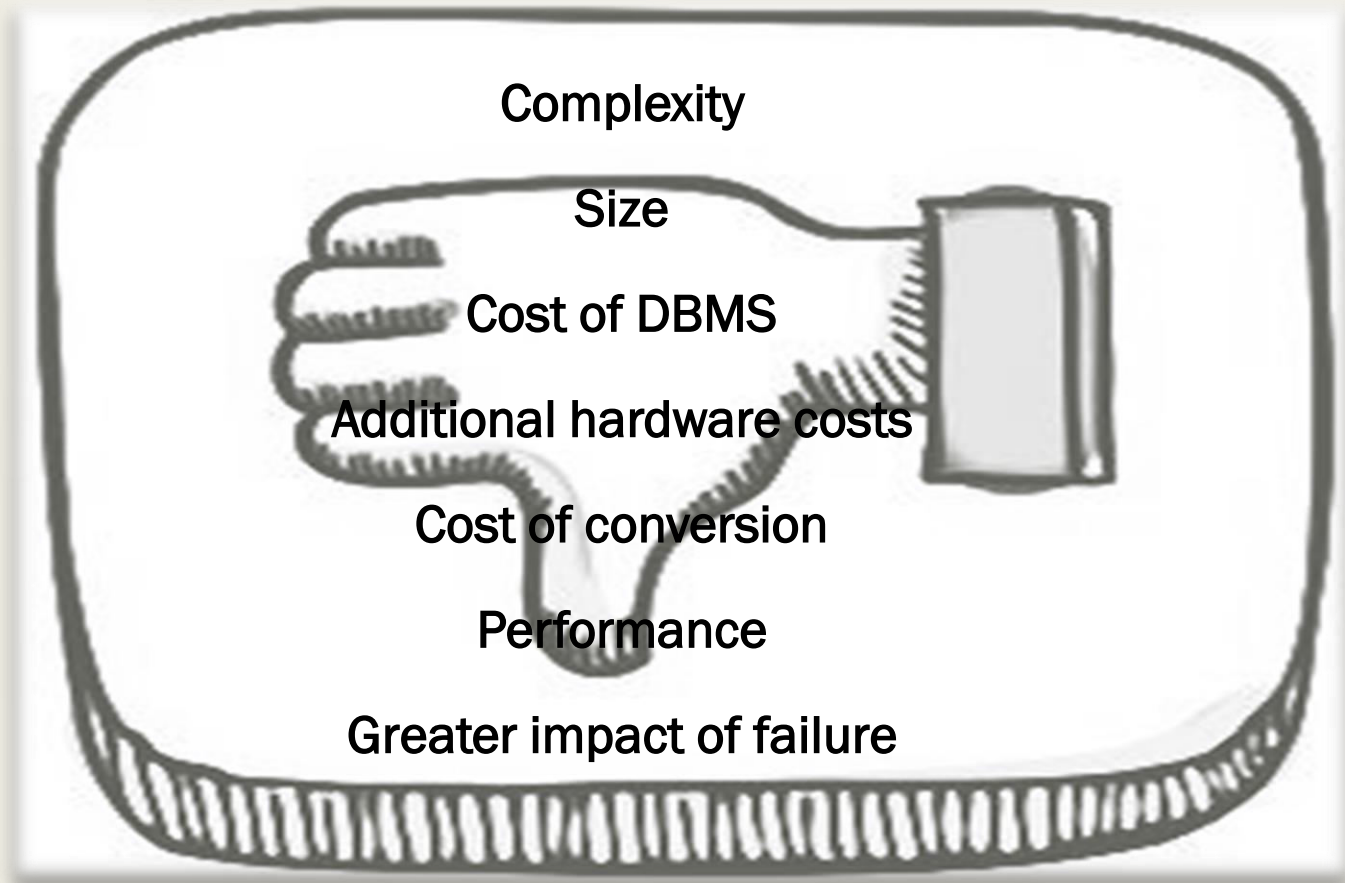
Third Generation

(Object-Relational, Object-Oriented)

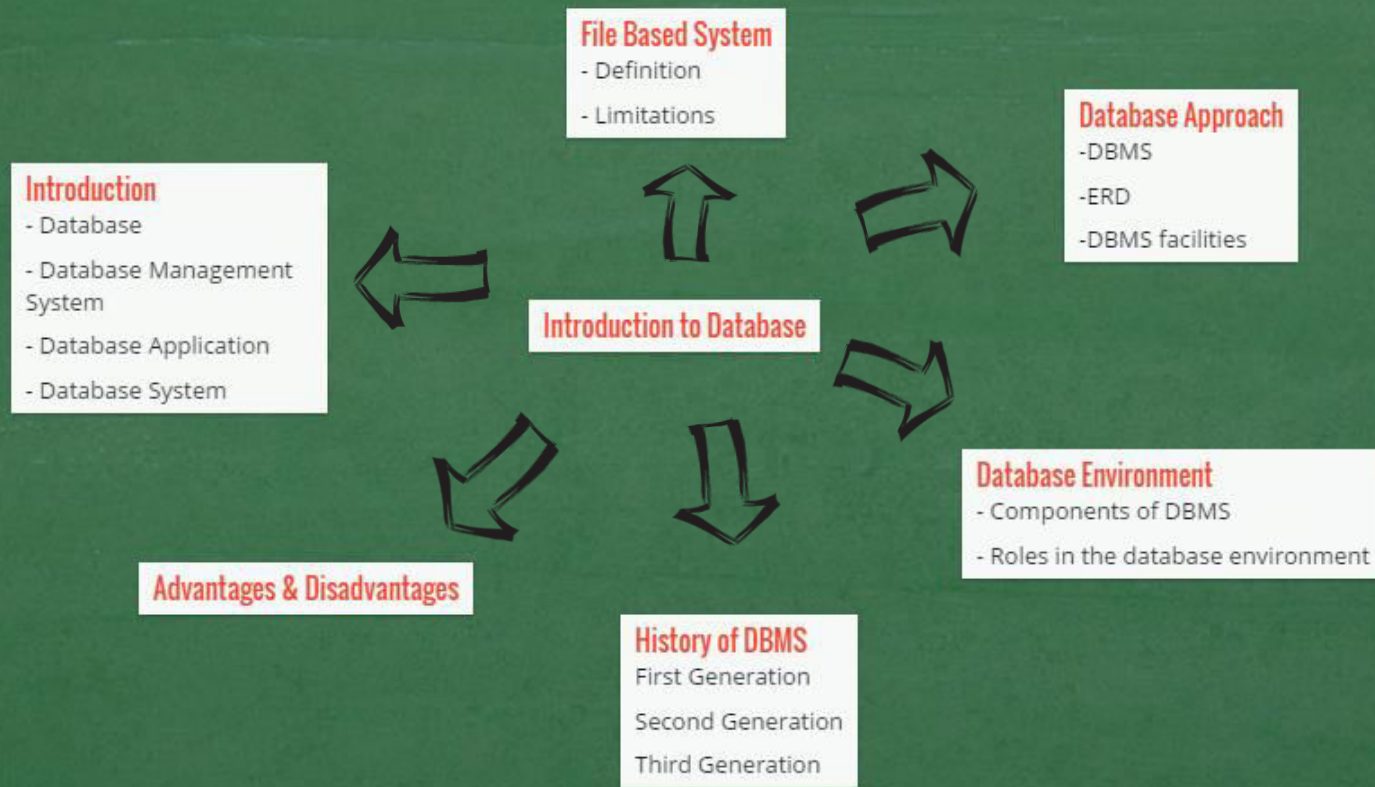
Advantages of DBMS

- 
- ✓ Control of data redundancy
 - ✓ Economy of scale
 - ✓ Data consistency
 - ✓ Balance conflicting requirements
 - ✓ More information from the same amount of data
 - ✓ Improved data accessibility and responsiveness
 - ✓ Sharing of data
 - ✓ Increased productivity
 - ✓ Improved data integrity
 - ✓ Improved maintenance through data independence
 - ✓ Improved security
 - ✓ Increased concurrency
 - ✓ Enforcement of standards
 - ✓ Improved backup and recovery services

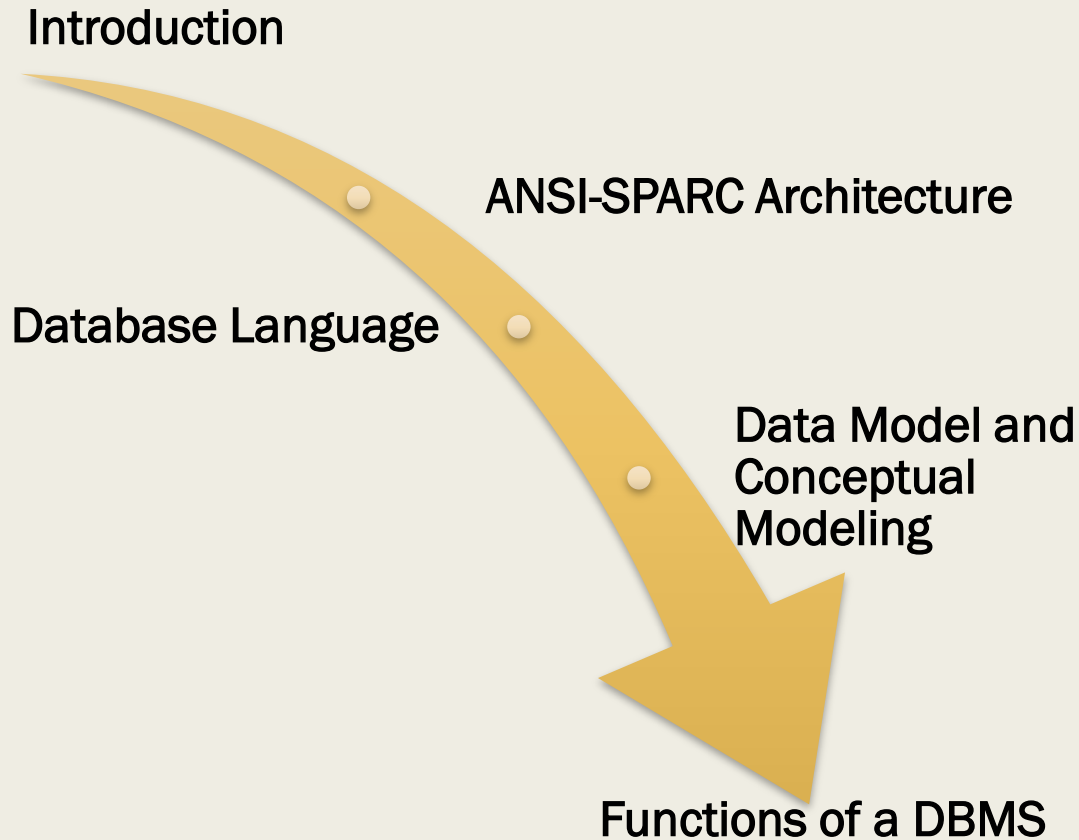
Disadvantages of DBMS



Summary



Lecture Content



objectives

- At the end of this lesson, students should be able to:
 - *Explain three level ANSI-SPARC architecture*
 - *Describe database language*
 - *Explain data model and conceptual modelling*
 - *Describe functions of DBMS*

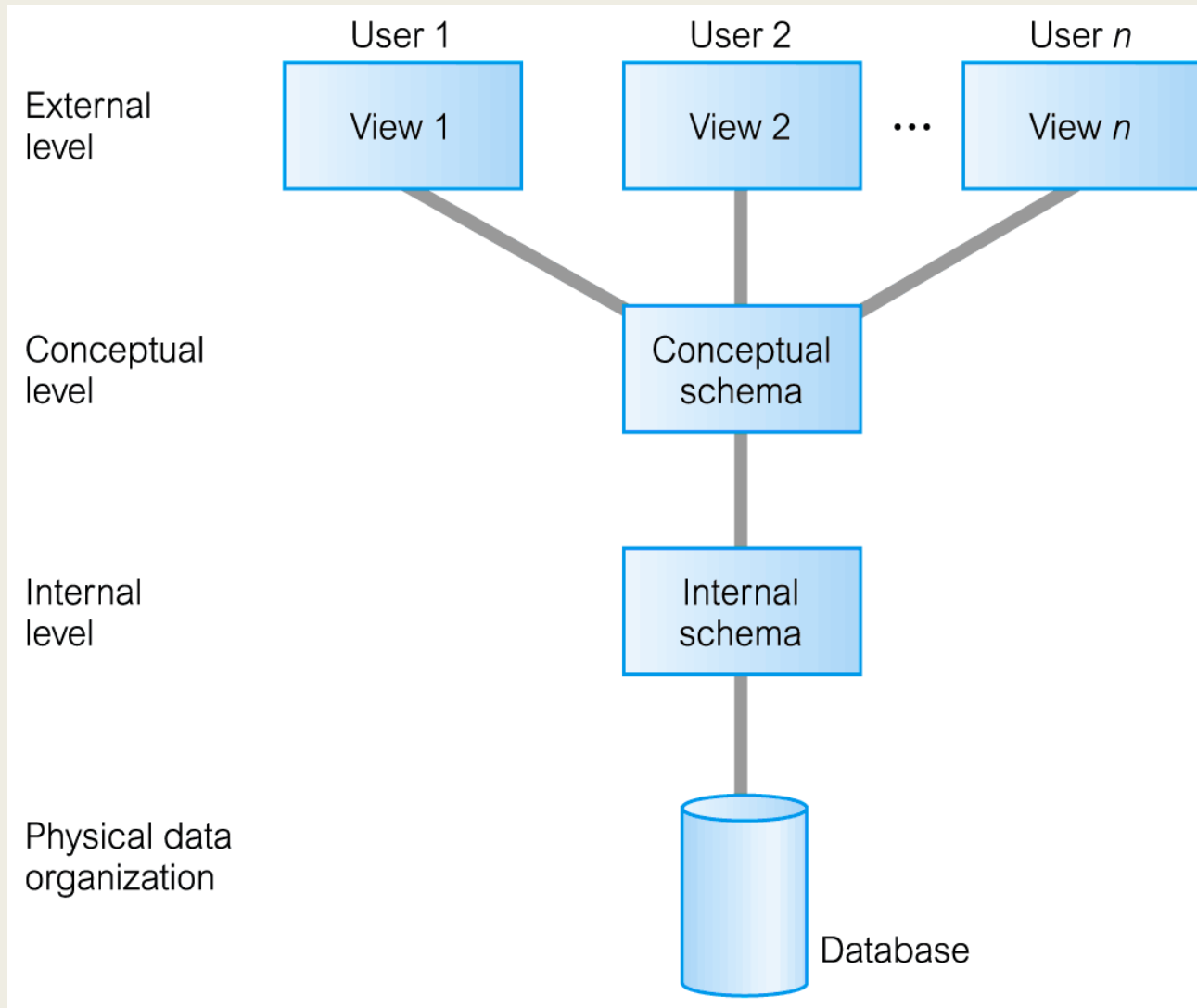
Introduction

- ❏ A major aim of a database system is to provide users with an abstract view of data, hiding certain details of how data is stored and manipulated.
- ❏ Design of a database must be an abstract and general description of the info requirements of the organization that is to be represented in the database.
- ❏ Each user wants a different view of data stored
- ❏ ANSI-SPARC architecture satisfies these needs

ANSI-SPARC Three Level Architecture

- ❏ Consists of three levels of abstraction (3 distinct level) at which data items can be described.
- ❏ **External level:** The way users perceive the data. Describes that part of database that is relevant to a particular user.
- ❏ **Conceptual level:** Provides both mapping and the desired independence between the external and internal levels. Community view of the database. Describes what data is stored in database and relationships among the data.
- ❏ **Internal level:** The way the DBMS and the operating system perceive the data. Physical representation of the database on the computer. Describes how the data is stored in the database.

ANSI-SPARC Three Level Architecture



Objectives of ANSI-SPARC 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.

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.

Differences Between the Three Level ANSI-SPARC Architecture

External view 1

sNo	fName	lName	age	salary
-----	-------	-------	-----	--------

External view 2

staffNo	lName	branchNo
---------	-------	----------

Conceptual level

staffNo	fName	lName	DOB	salary	branchNo
---------	-------	-------	-----	--------	----------

Internal level

```
struct STAFF {  
    int staffNo;  
    int branchNo;  
    char fName [15];  
    char lName [15];  
    struct date dateOf Birth;  
    float salary;  
    struct STAFF *next;  
};  
index staffNo; index branchNo;
```

/* pointer to next Staff record */
/* define indexes for staff */

ANSI-SPARC Three Level Architecture- Data Independence

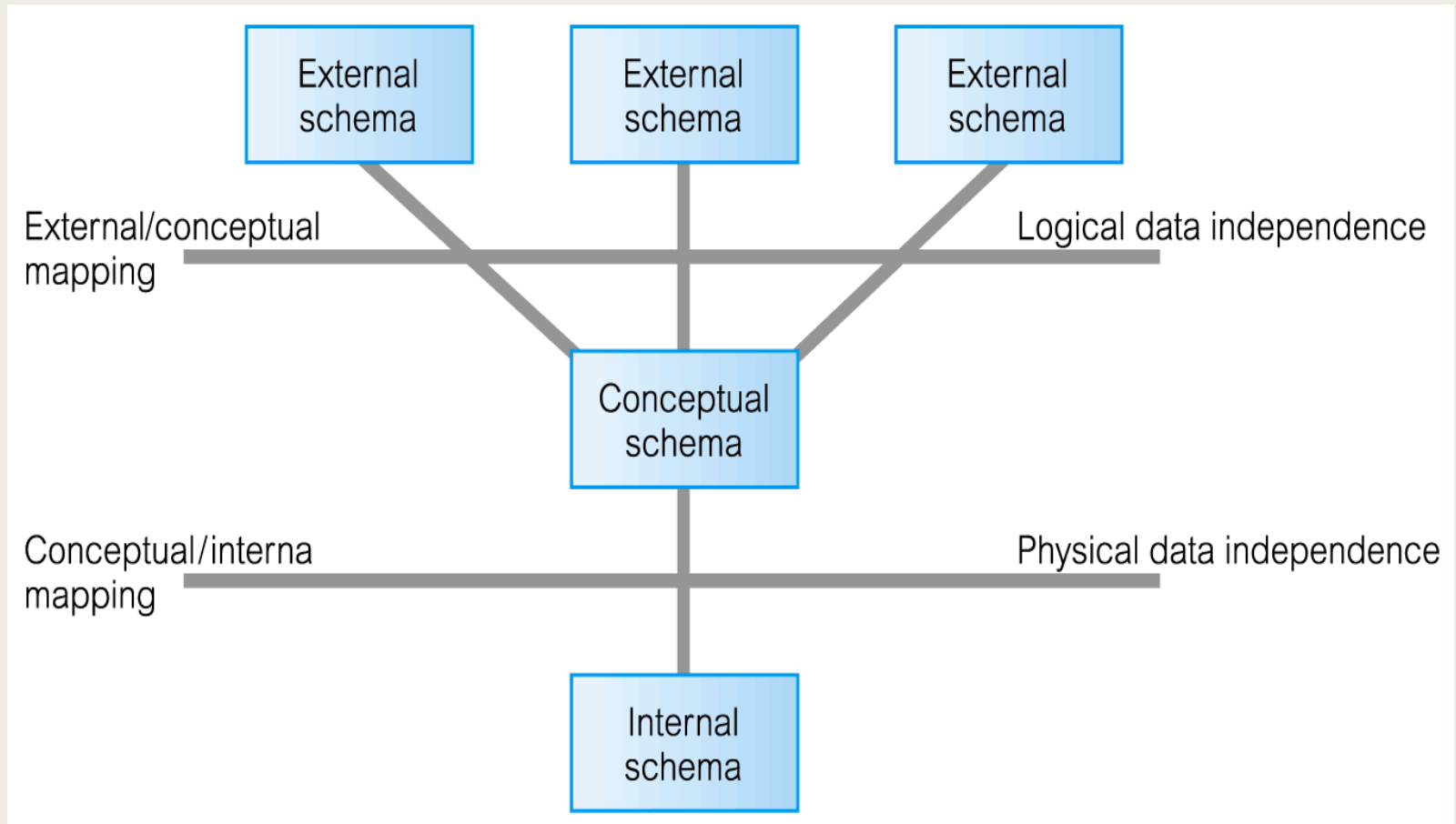
Logical data
independence

- 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.

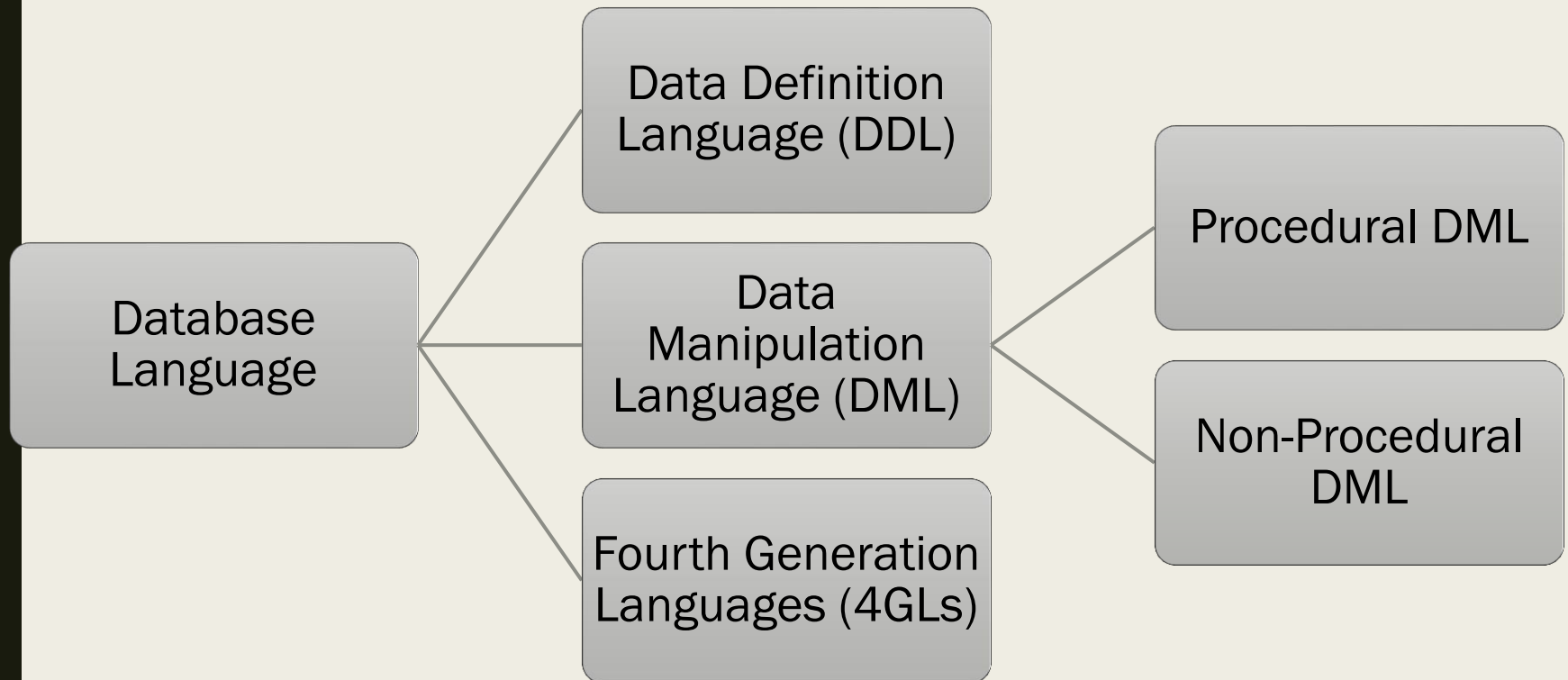
Physical data
independence

- 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.


ANSI-SPARC Three Level Architecture- Data Independence



Database Language



Data Models & Conceptual Modeling

 **Data Model:** Integrated 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*

Data Models & Conceptual Modeling

 **Purpose:** To represent data in an understandable way.

 Categories of data models include:

-  *Object-based (Entity-Relationship, Semantic, Functional, Object-Oriented)*
-  *Record-based (Relational Data Model, Network Data Model, Hierarchical Data Model)*
-  *Physical Data Models*

Data Models & Conceptual Modeling

- ❏ Conceptual schema is the core of a system supporting all user views
- ❏ Should be complete and accurate representation of an organization's data requirements.
- ❏ Conceptual modeling is 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.

Authorization services

A User-Accessible Catalog.

Support for data communication

Transaction Support.

Integrity services

Concurrency control services

Services to promote data
independence

Recovery services

Utility services

Summary

Introduction

ANSI-SPARC Architecture

Database Language

Data Model and Conceptual Modeling

Functions of a DBMS

- DDL
- DML
- 4GLs

Data Storage, Retrieval, and Update.
Authorization services
A User-Accessible Catalog.
Support for data communication
Transaction Support.

Integrity services
Concurrency control services
Services to promote data independence
Recovery services
Utility services

- 3 level
 - External, conceptual, internal
- Objectives
- Differences of level
- Data independence

- Parts
 - Structural, manipulative, integrity rules
- Purpose
- Category
 - Object-based, record-based, physical data models

Reference

- *Database Systems: A Practical Approach to Design, Implementation, and Management*, Thomas Connolly and Carolyn Begg, 5th Edition, 2010, Pearson.