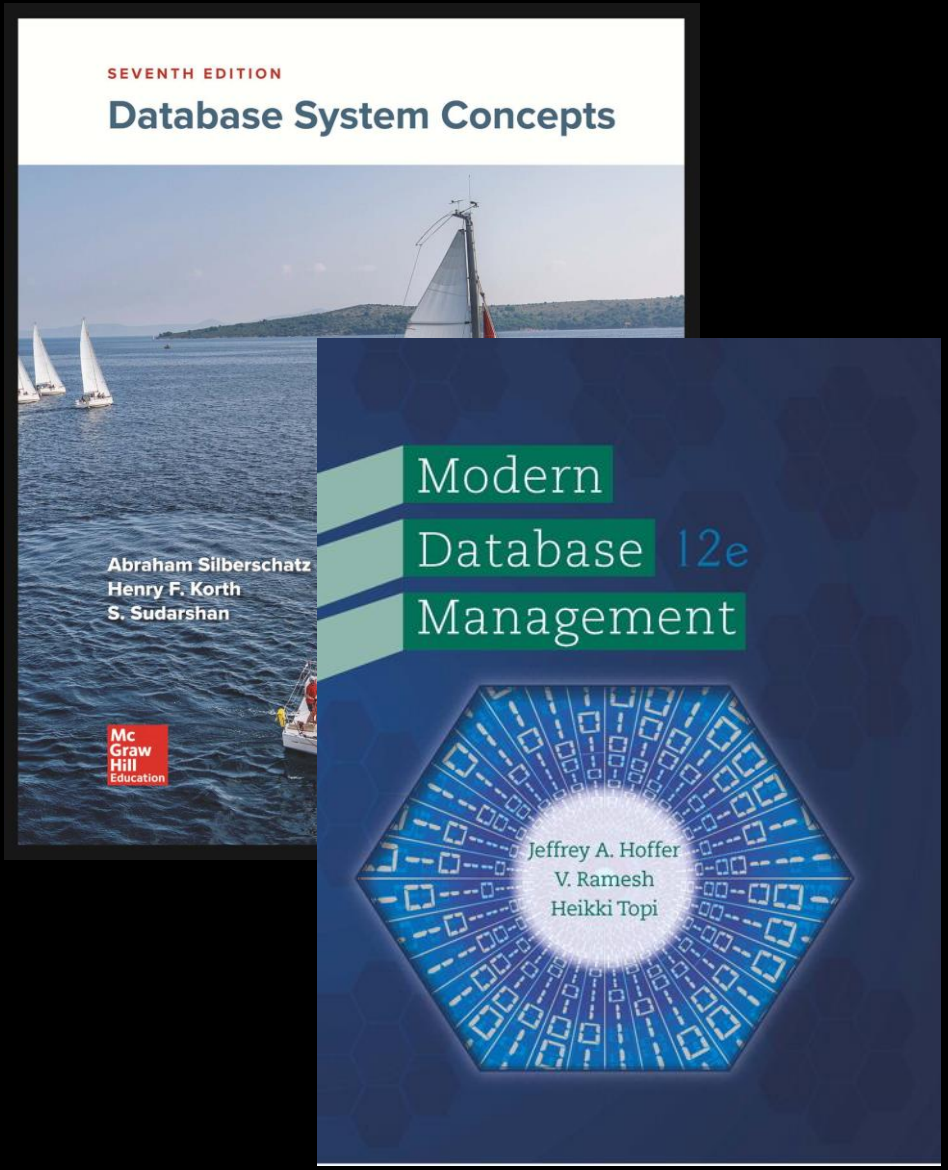
IF2240 – Basis Data

Introduction

KNOWLEDGE & SOFTWARE ENGINEERING



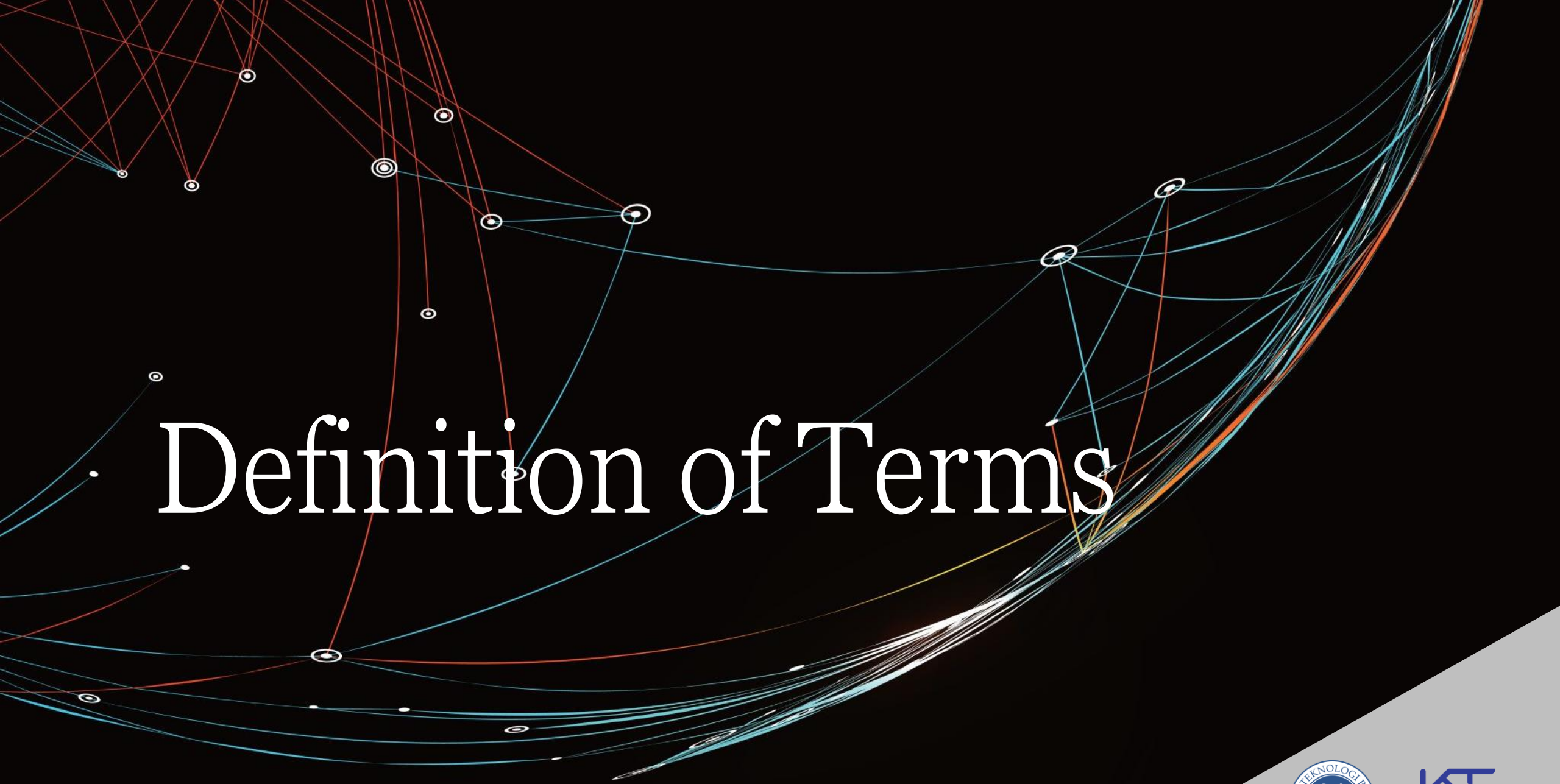
References

Abraham Silberschatz, Henry F. Korth, S. Sudarshan : "**Database System Concepts**", 7th Edition

- Chapter 1: Introduction

Jeffrey A. Hoffer, Mary B. Prescott, Heikki Topi : "**Modern Database Management**", 12th Edition

- Chapter 1: The Database Environment and Development Process



Definition of Terms

KNOWLEDGE & SOFTWARE ENGINEERING

Definition (1/3)

Database	organized collection of logically related data
Data	stored representations of meaningful objects and events Structured: numbers, text, dates Unstructured: images, video, documents

Example:

Baker, Kenneth D.	324917628
Doyle, Joan E.	476193248
Finkle, Clive R.	548429344
Lewis, John C.	551742186
McFerran, Debra R.	409723145

Definition (2/3)

Information	data processed to increase knowledge in the person using the data

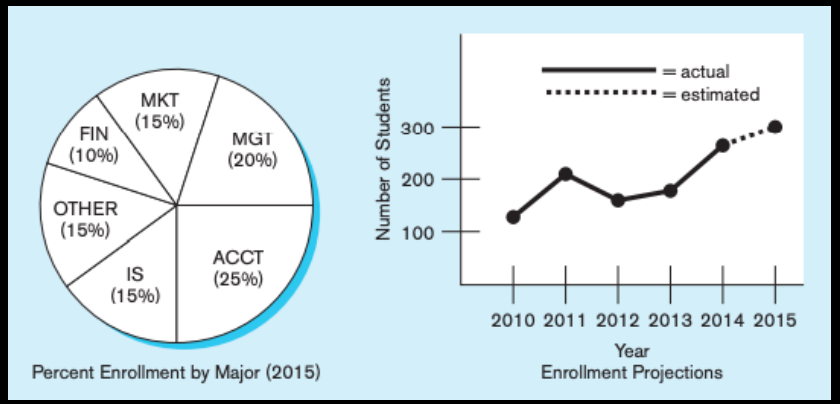
Example:

Class Roster

Course: MGT 500 Business Policy Semester: Spring 2015

Section: 2

Name	ID	Major	GPA
Baker, Kenneth D.	324917628	MGT	2.9
Doyle, Joan E.	476193248	MKT	3.4
Finkle, Clive R.	548429344	PRM	2.8
Lewis, John C.	551742186	MGT	3.7
McFerran, Debra R.	409723145	IS	2.9
Sisneros, Michael	392416582	ACCT	3.3



Summarized data - Graphical displays turn data into useful information that managers can use for decision making and interpretation

Data in context - Context helps users understand data

Definition (3/3)

Metadata	data that describes the properties and context of user data

Example:

TABLE 1-1 Example Metadata for Class Roster

Data Item	Metadata					
Name	**Type**	**Length**	**Min**	**Max**	**Description**	**Source**
Course	Alphanumeric	30			Course ID and name	Academic Unit
Section	Integer	1	1	9	Section number	Registrar
Semester	Alphanumeric	10			Semester and year	Registrar
Name	Alphanumeric	30			Student name	Student IS
ID	Integer	9			Student ID (SSN)	Student IS
Major	Alphanumeric	4			Student major	Student IS
GPA	Decimal	3	0.0	4.0	Student grade point average	Academic Unit

Descriptions of the properties or characteristics of the data, including data types, field sizes, allowable values, and data context

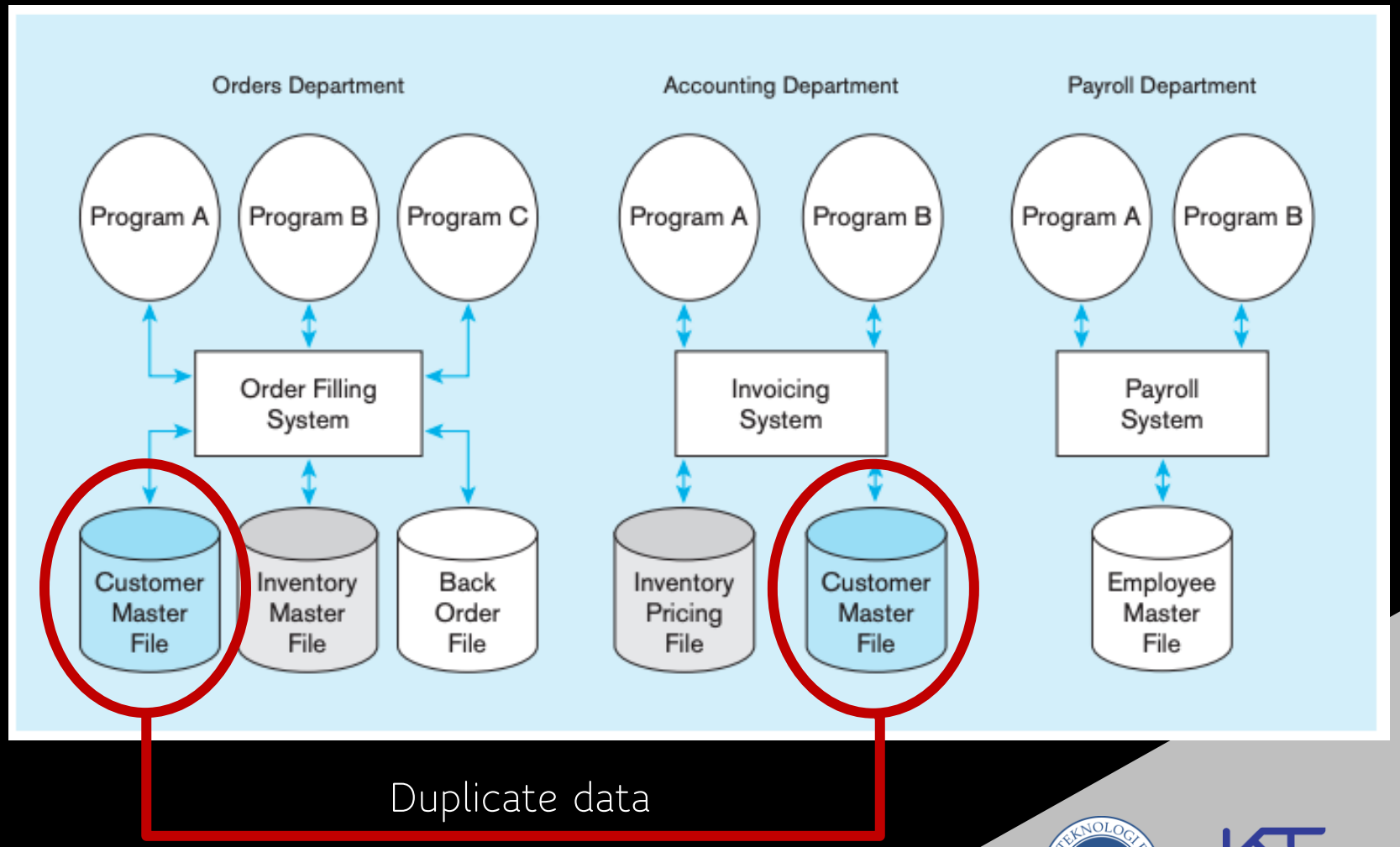
File Processing System vs The Database Approach

KNOWLEDGE & SOFTWARE ENGINEERING

Traditional File Processing System

When computer-based data processing was first available, there were no databases. To be useful for business applications, computers had to store, manipulate, and retrieve large files of data.

File processing systems at Pine Valley Furniture Company



Disadvantages of File Processing (1)

Program-Data Dependence	Data Redundancy (Duplication of data)	Limited Data Sharing	Lengthy Development Times	Excessive Program Maintenance
· All programs maintain metadata for each file they use	· Different systems/ programs have separate copies of the same data	· No centralized control of data	· Programmers must design their own file formats	· 80% of information systems budget

Disadvantages of File Processing (2)

Difficulty in accessing data	Data isolation	Integrity problems	Atomicity of updates	Concurrent access by multiple users	Security problems
· Need to write a new program to carry out each new task	· Multiple files and formats	· 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	· Failures may leave database in an inconsistent state with partial updates carried out	· Concurrent access needed for performance · Uncontrolled concurrent accesses can lead to inconsistencies	· Hard to provide user access to some, but not all, data

Problems with **Data Dependency**

Each application programmer must maintain their own data

Each application program needs to include code for the metadata of each file

Each application program must have its own processing routines for reading, inserting, updating and deleting data

Lack of coordination and central control

Non-standard file formats

Problems with **Data Redundancy**

Waste of space to have duplicate data

Causes more maintenance headaches

The biggest Problem:

- When data changes in one file, could cause inconsistencies
- Compromises *data integrity*

Solution : The Database Approach

CENTRAL REPOSITORY OF SHARED DATA

DATA IS MANAGED BY A CONTROLLING AGENT

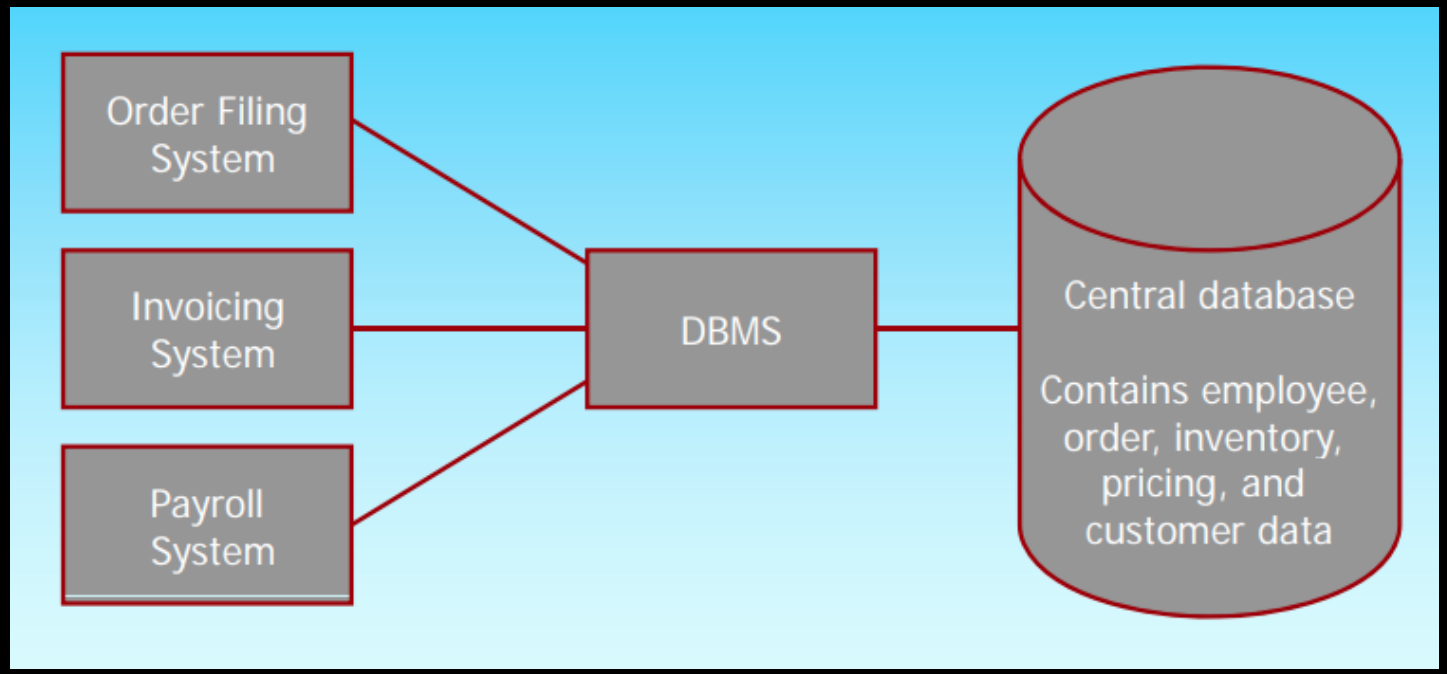
STORED IN A STANDARDIZED, CONVENIENT FORM

Requires a **Database Management System (DBMS)**

Database Management System (DBMS)

A software system that is used to create, maintain, and provide controlled access to user databases

DBMS manages data resources like an operating system manages hardware resources



Advantages of Database Approach

Improved Data Sharing	Enforcement of Standards	Improved Data Quality	Better Data Accessibility/ Responsiveness	Security, Backup/Recovery, Concurrency
· Different users get different views of the data	· All data access is done in the same way	· Constraints, data validation rules	· Use of standard data query language (SQL)	· Disaster recovery is easier

Costs and Risks of the Database Approach

Up-front costs

- Installation Management Cost and Complexity
- Conversion Costs

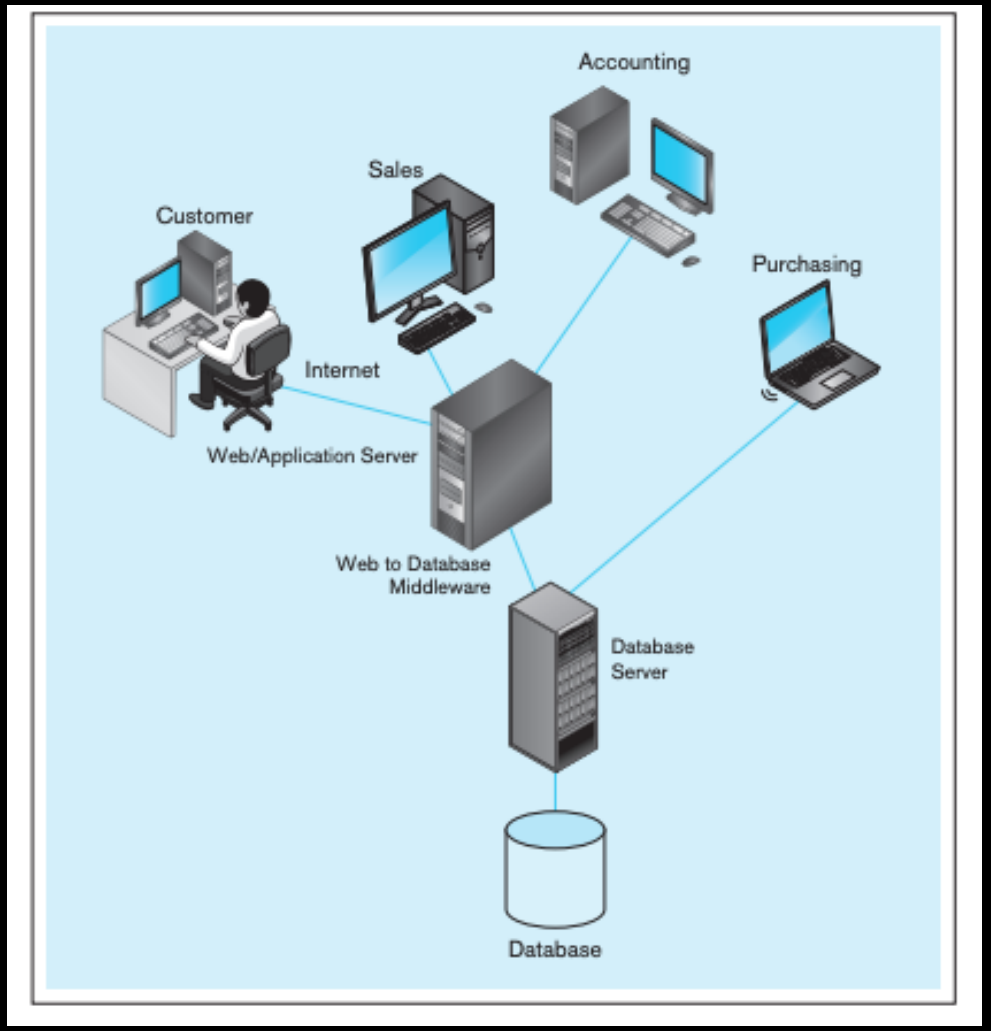
Ongoing Costs

- Requires New, Specialized Personnel
- Need for Explicit Backup and Recovery

Organizational Conflict

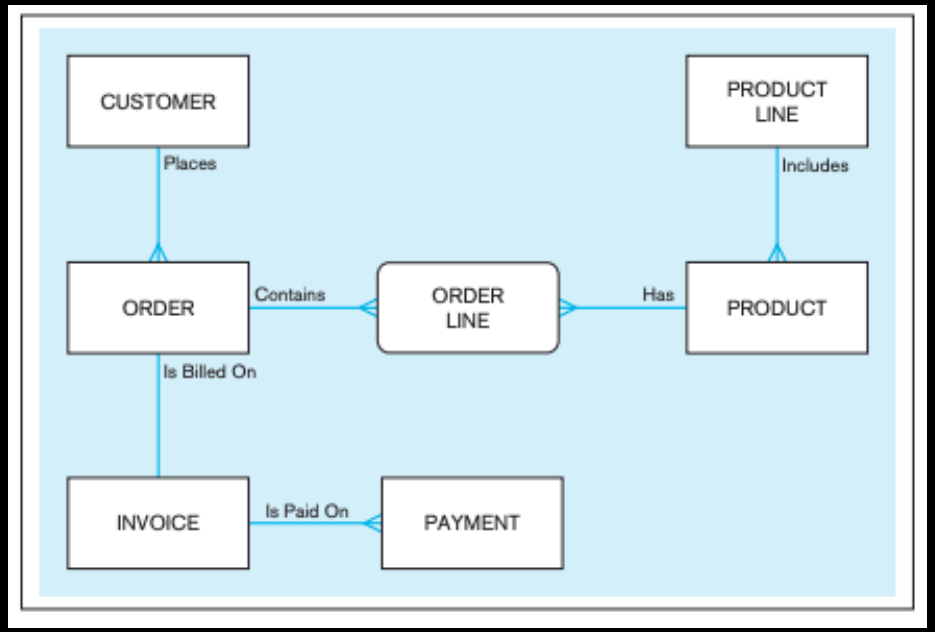
- Old habits die hard

Developing a **database** application for Pine Valley Furniture Company (1/4)

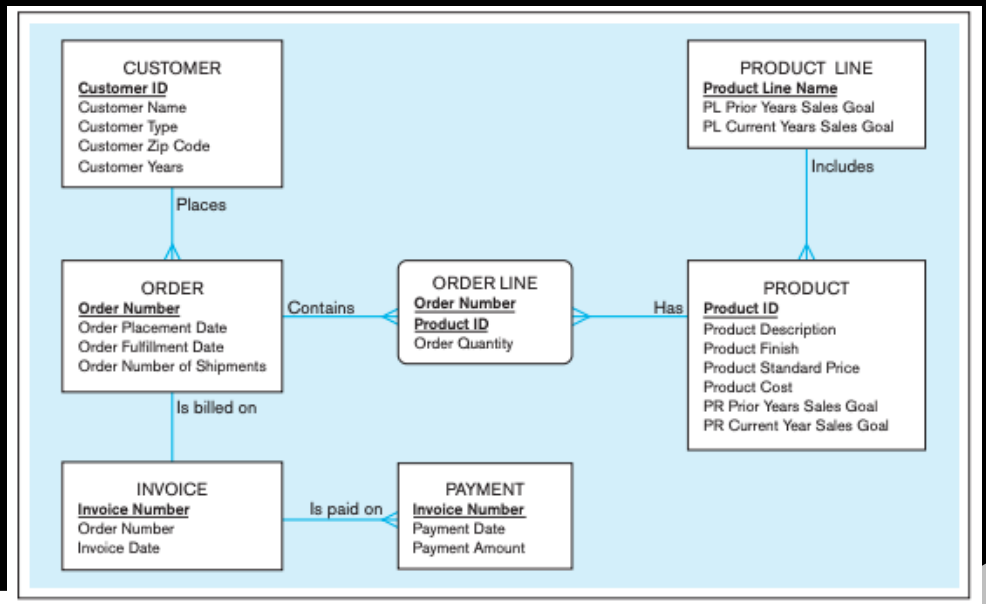


Computer System for Pine Valley Furniture Company

Developing a **database** application for Pine Valley Furniture Company (2/4)



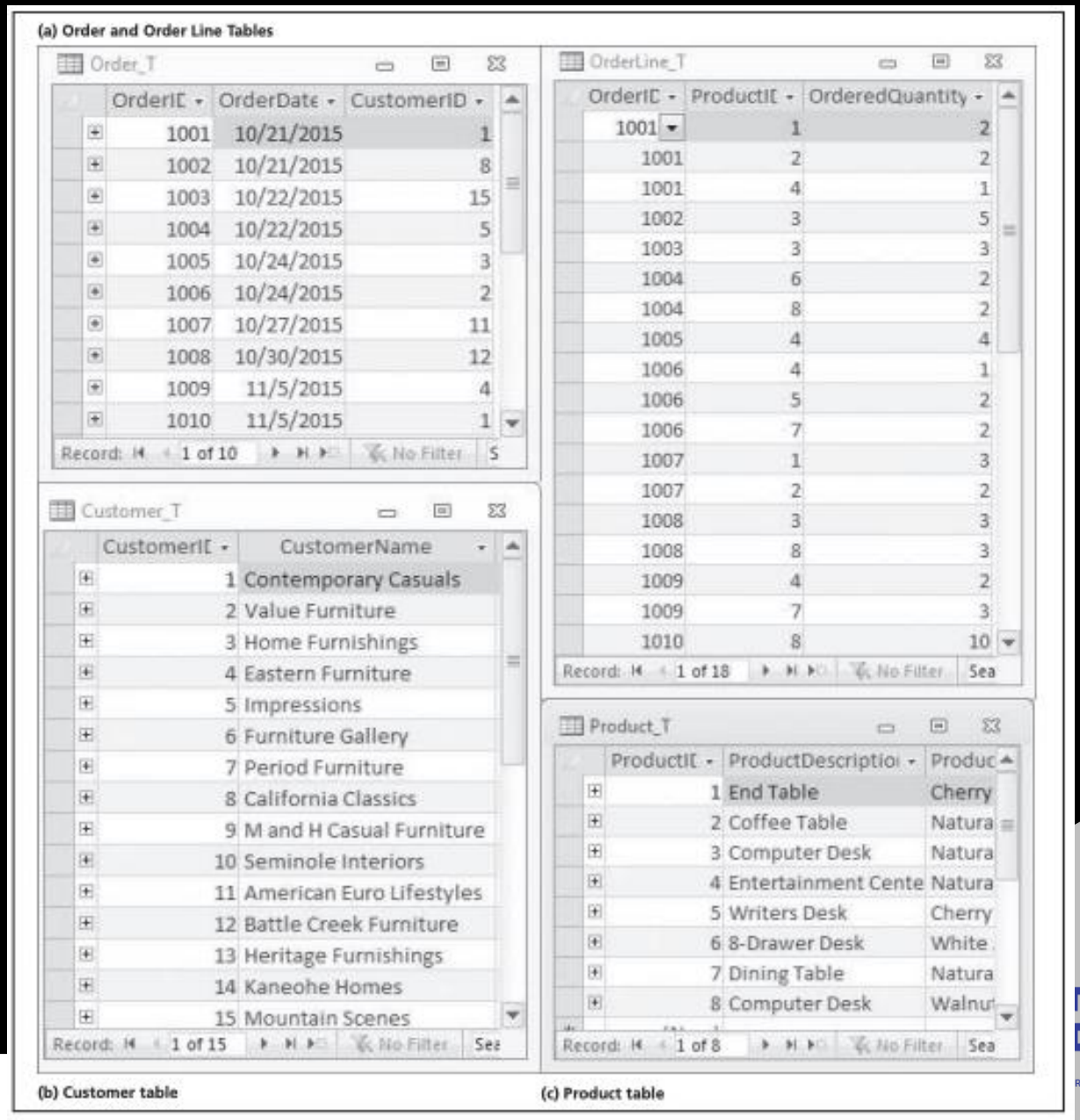
Preliminary data model for Home Office product line marketing support system



Project data model for Home Office product line marketing support system

Developing a **database** application for Pine Valley Furniture Company (3/4)

Four **relations** (Pine Valley Furniture Company)



Developing a **database** application for Pine Valley Furniture Company (4/4)

```
SELECT Product.ProductID, Product.ProductDescription, Product.PRCurrentYearSalesGoal,
       (OrderQuantity * ProductPrice) AS SalesToDate
FROM Order.OrderLine, Product.ProductLine
WHERE Order.OrderNumber = OrderLine.OrderNumber
AND Product.ProductID = OrderedProduct.ProductID
AND Product.ProductID = ProductLine.ProductID
AND Product.ProductLineName = "Home Office";
```

SQL query for Home Office sales-to-goal comparison

Home Office Sales to Date : Select Query

Product ID	Product Description	PR Current Year Sales Goal	Sales to Date
3	Computer Desk	$23,500.00	5625
10	96" Bookcase	$22,500.00	4400
5	Writer's Desk	$26,500.00	650
3	Computer Desk	$23,500.00	3750
7	48" Bookcase	$17,000.00	2250
5	Writer's Desk	$26,500.00	3900

Home Office product line sales comparison

Customer invoice (Pine Valley Furniture Company)

PVFC Customer Invoice

Customer ID	2	Order Number	1006
Customer Name	Value Furniture	Order Date	10/24/2000
Address	15145 S.W. 17th St. Plano, TX 75094		

Product_ID	Product Description	Finish	Quantity	Unit Price	Extended Price:
7	Dining Table	Natural As	2	$800.00	$1,600.00
5	Writer's Desk	Cherry	2	$325.00	$650.00
4	Entertainment Center	Natural M	1	$650.00	$650.00
				Total	$2,900.00

Database Applications

KNOWLEDGE & SOFTWARE ENGINEERING

Components of the Database Environment

CASE Tools – computer-aided software engineering

Repository – centralized storehouse of metadata

Database Management System (DBMS) – software for managing the database

Database – storehouse of the data

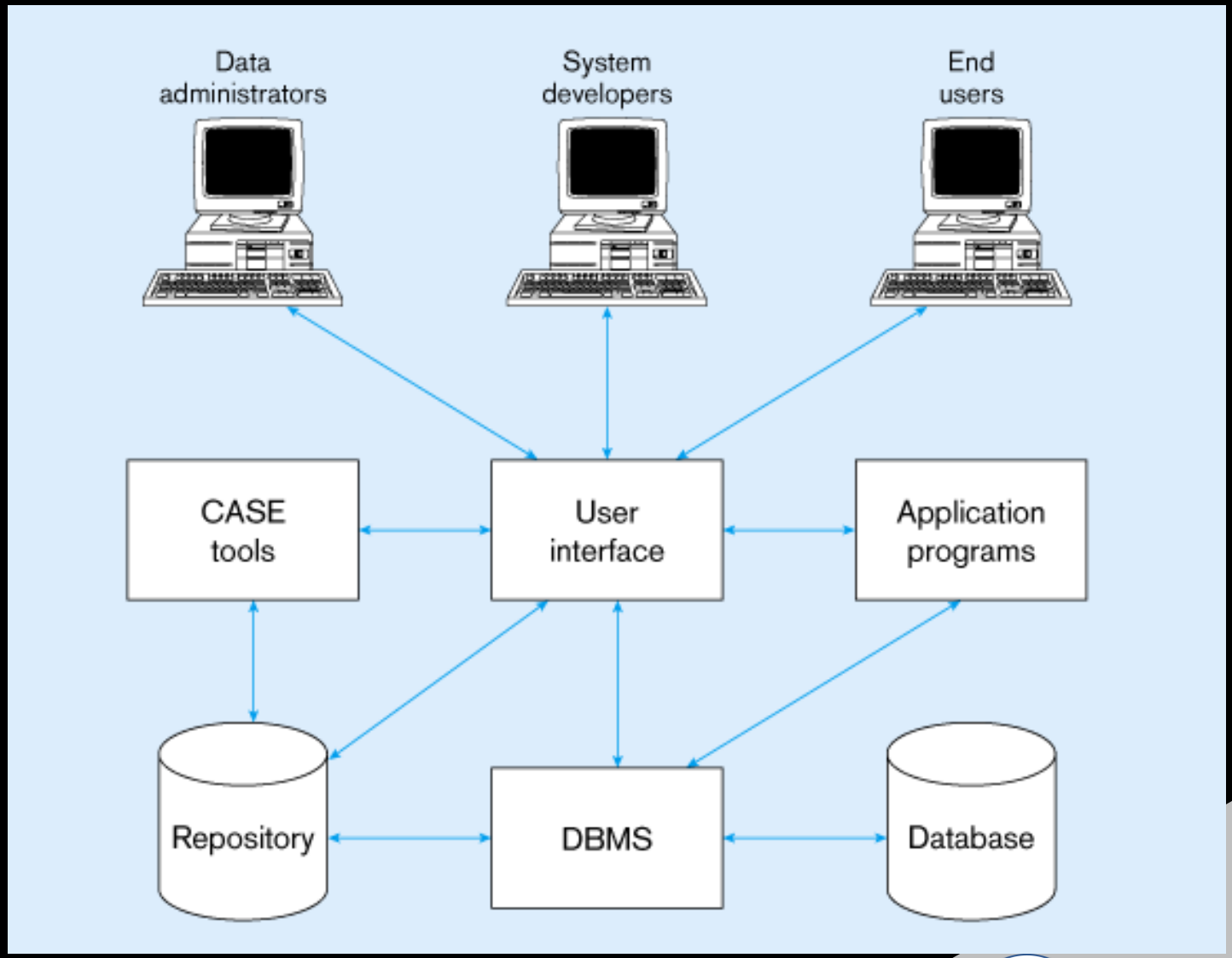
Application Programs – software using the data

User Interface – text and graphical displays to users

Data Administrators – personnel responsible for maintaining the database

System Developers – personnel responsible for designing databases and software

End Users – people who use the applications and databases



The Range of Database Applications

Personal databases

Workgroup databases

Departmental/ divisional databases

Enterprise database

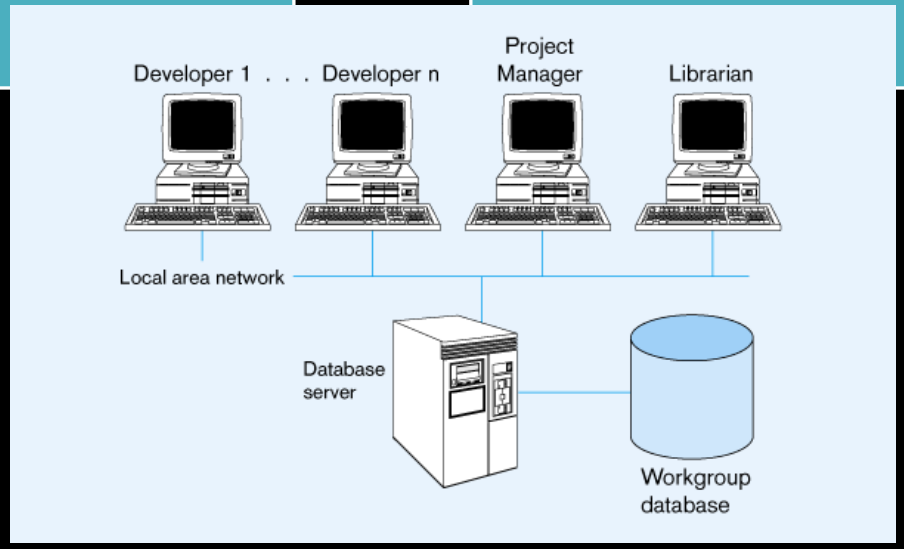
- Enterprise resource planning (ERP) systems
- Data warehousing implementations

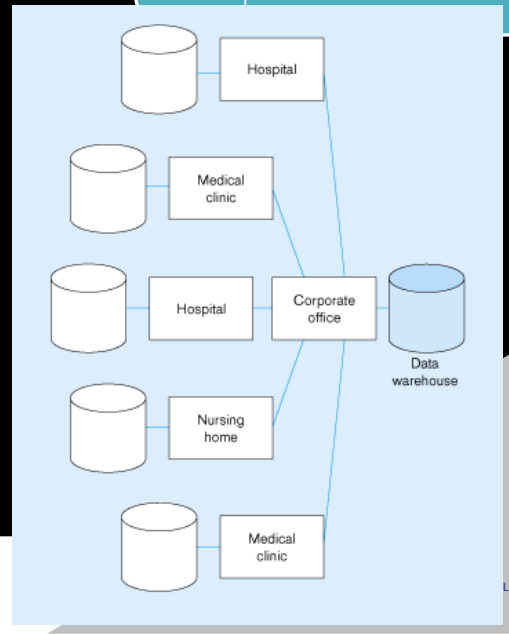
Customer

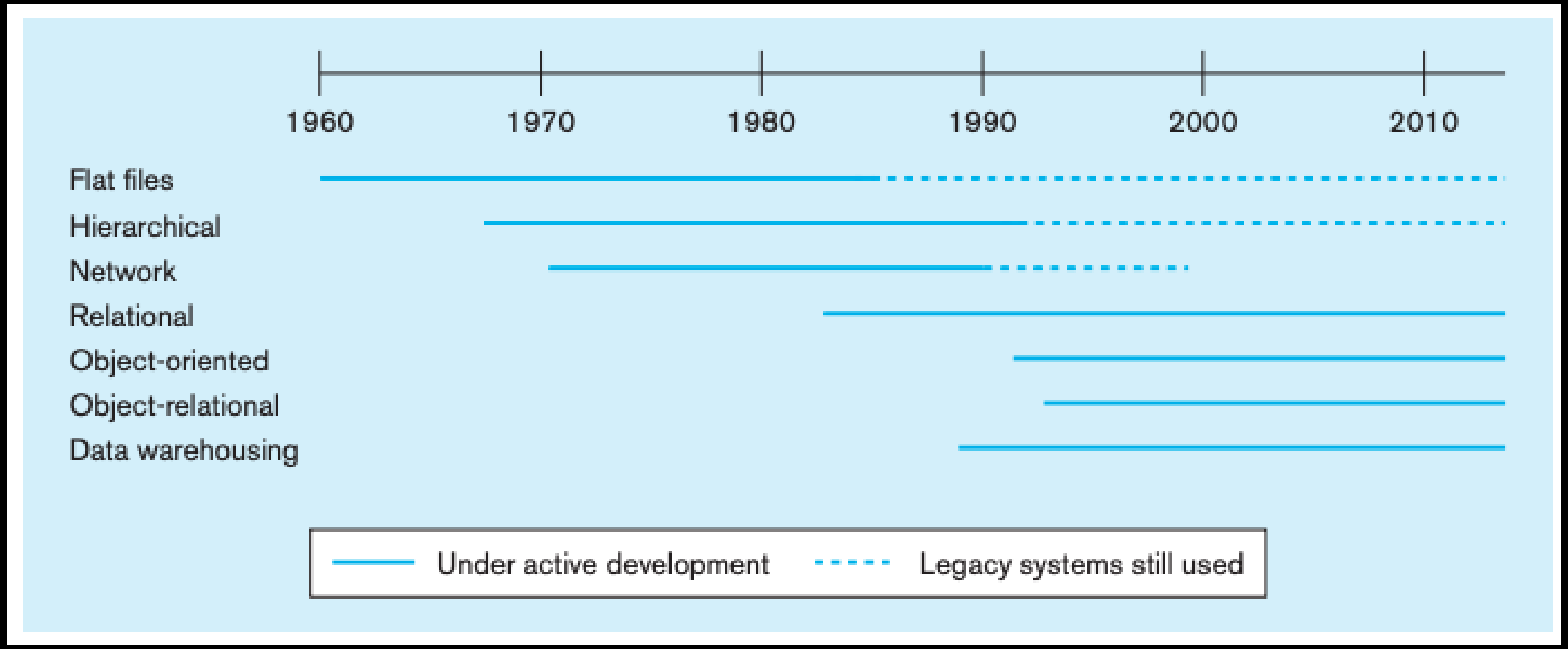
Customer Name: **Multi Media, Inc.**
Address: **1000 River Road**
City: **San Antonio**
State: **TX**
Zip: **76235**
Phone: **(219) 864-2000**
Next Contact Date: **10/17/2000** / Time: **10:30 AM**

Contact History for Customer

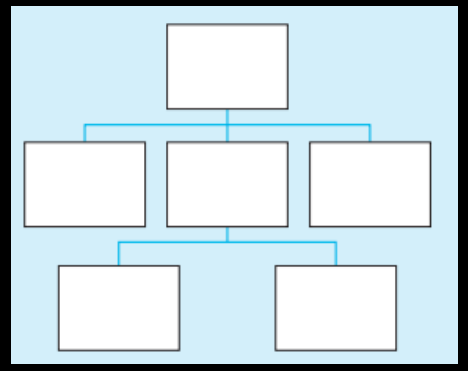
Date	Time	Contact	Comments
08/04/2000	10:00 AM	Roberts	Review proposal
08/19/2000	08:00 AM	Roberts	Revise schedule
09/10/2000	09:00 AM	Pearson	Sign contract
09/21/2000	02:00 PM	Roberts	Follow up





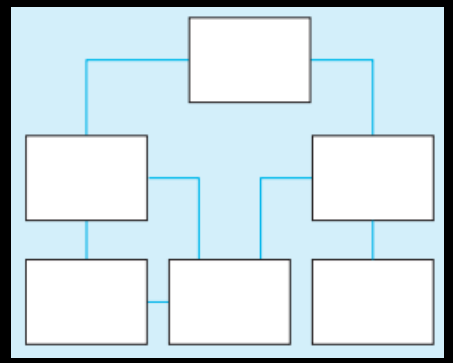


Evolution of Database Technologies

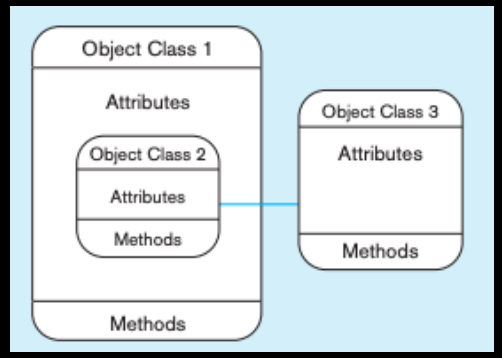


Hierarchical database model

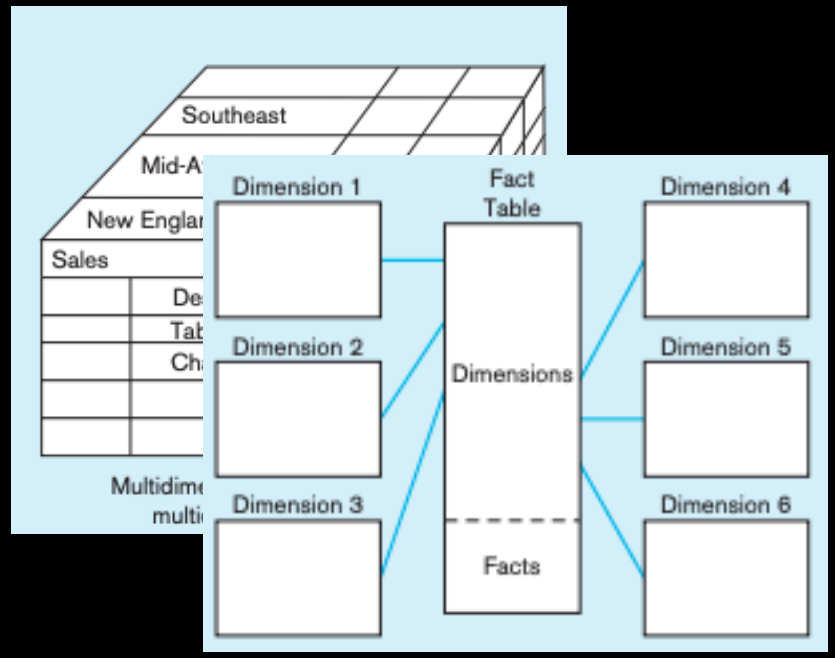
Relational database model



Network database model



Object-oriented database model



Multidimensional database model

Database Architecture

More on Database Approach

Modified form Silberschatz's slides, Database System Concept, 7th edition

KNOWLEDGE & SOFTWARE ENGINEERING

Levels of Abstraction

Physical level: describes how a record is stored.

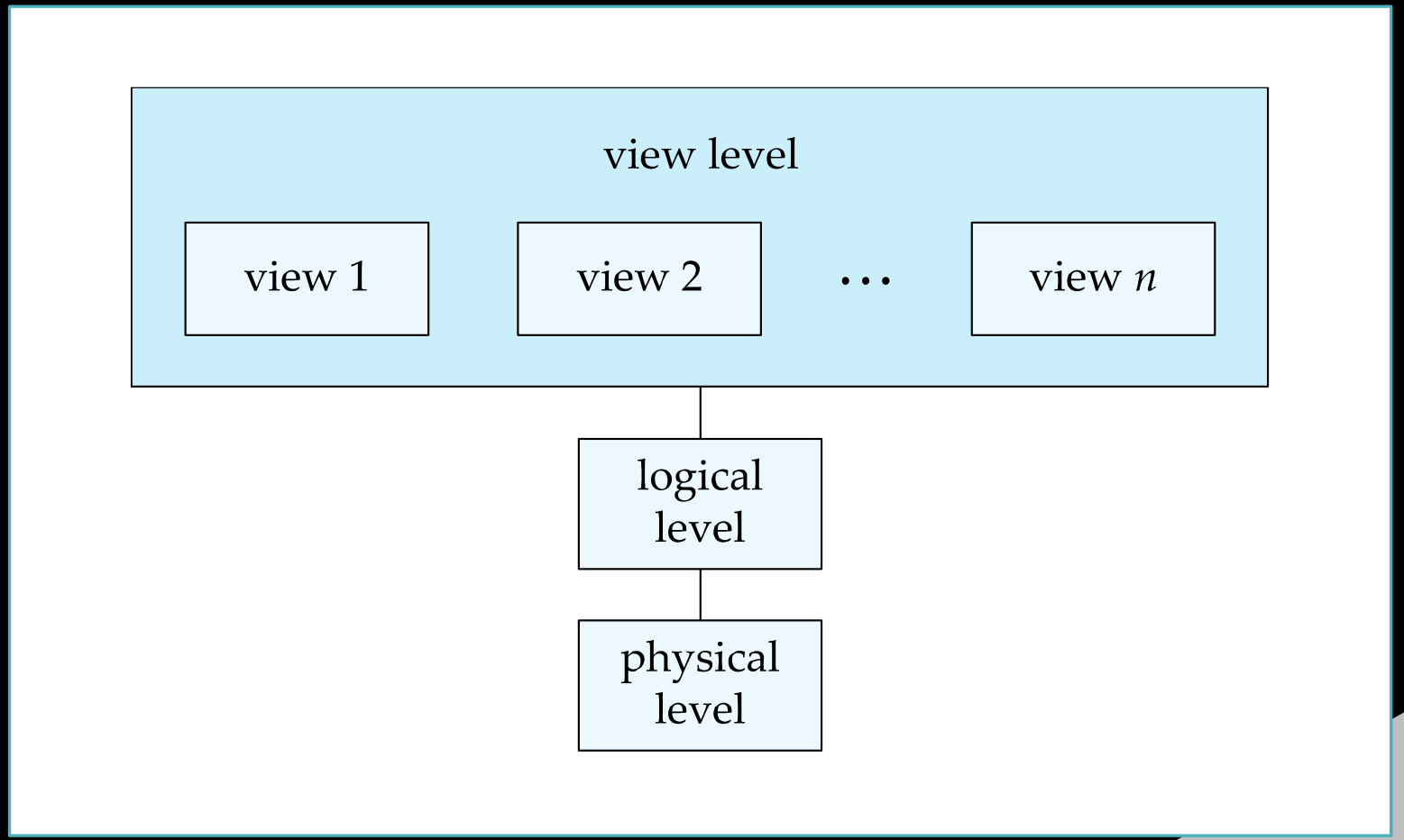
Logical level: describes data stored in database, and the relationships among the data.

```
type instructor = record
        ID : string;
        name : string;
        dept_name : string;
        salary : integer;
    end;
```

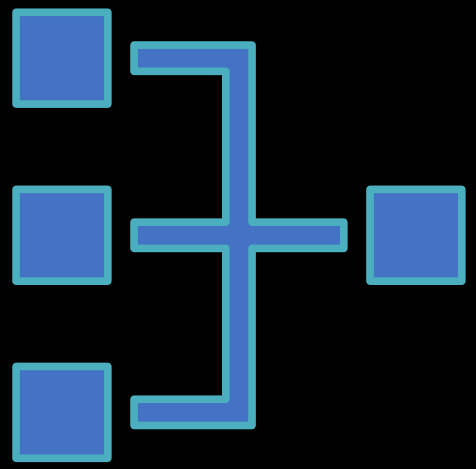
View level: application programs hide details of data types. Views can also hide information (such as an employee's salary) for security purposes.

View of Data

An architecture for a database system (ANSI/SPARC Architecture)



Physical Data Independence



Physical Data Independence – the ability to modify the physical schema without changing the logical schema

- Applications depend on the logical schema
- In general, the interfaces between the various levels and components should be well defined so that changes in some parts do not seriously influence others.

Instances and Schemas

Schema : structure of the database

Logical Schema – the overall logical structure of the database

Physical Schema – the overall physical structure of the database

Instance : the actual content of the database at a particular point in time

Similar to types and variables in programming languages

Data Definition Language (DDL)

Specification notation for defining the database schema

Example:
```
create table instructor (
    ID          char(5),
    name        varchar(20),
    dept_name   varchar(20),
    salary      numeric(8,2))
```

DDL compiler generates a set of table templates stored in a ***data dictionary*** - contains metadata (i.e., data about data)

- Database schema
- Integrity constraints
  - Primary key (ID uniquely identifies instructors)
- Authorization
  - Who can access what

Data Manipulation Language (DML)

Language for accessing and updating the data organized by the appropriate data model

There are basically two types of data-manipulation language

- **Procedural DML** -- require a user to specify what data are needed and how to get those data.
- **Declarative DML** -- require a user to specify what data are needed without specifying how to get those data.

SQL Query Language

SQL query language is nonprocedural. A query takes as input several tables (possibly only one) and always returns a single table.

Example to find all instructors in Comp. Sci. dept

```
select name
from instructor
where dept_name = 'Comp. Sci.'
```

Application programs generally access databases through one of

- Language extensions to allow embedded SQL
- Application program interface (e.g., ODBC/JDBC) which allow SQL queries to be sent to a database

KNOWLEDGE & SOFTWARE ENGINEERING

Database Design

Logical Design – Deciding on the database schema. Database design requires that we find a "good" collection of relation schemas.

Business decision – What attributes should we record?

Computer Science decision – What relation schemas should we have?

Physical Design – Deciding on the physical layout of the database

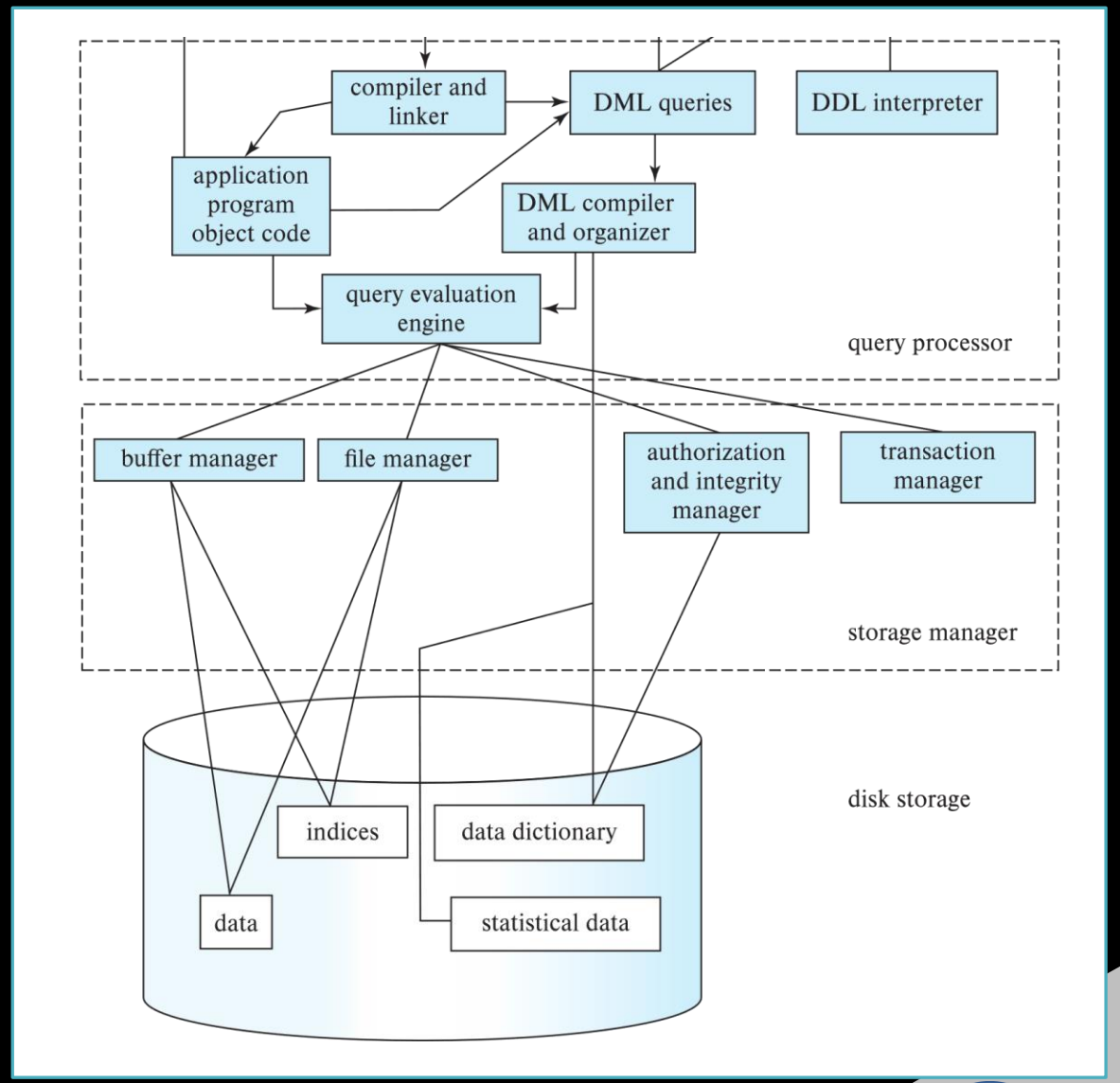
Database Engine

- A database system is partitioned into modules that deal with each of the responsibilities of the overall system.
- The **functional components** of a database system can be divided into:

The storage manager	The query processor component	The transaction management component

A Database System Architecture

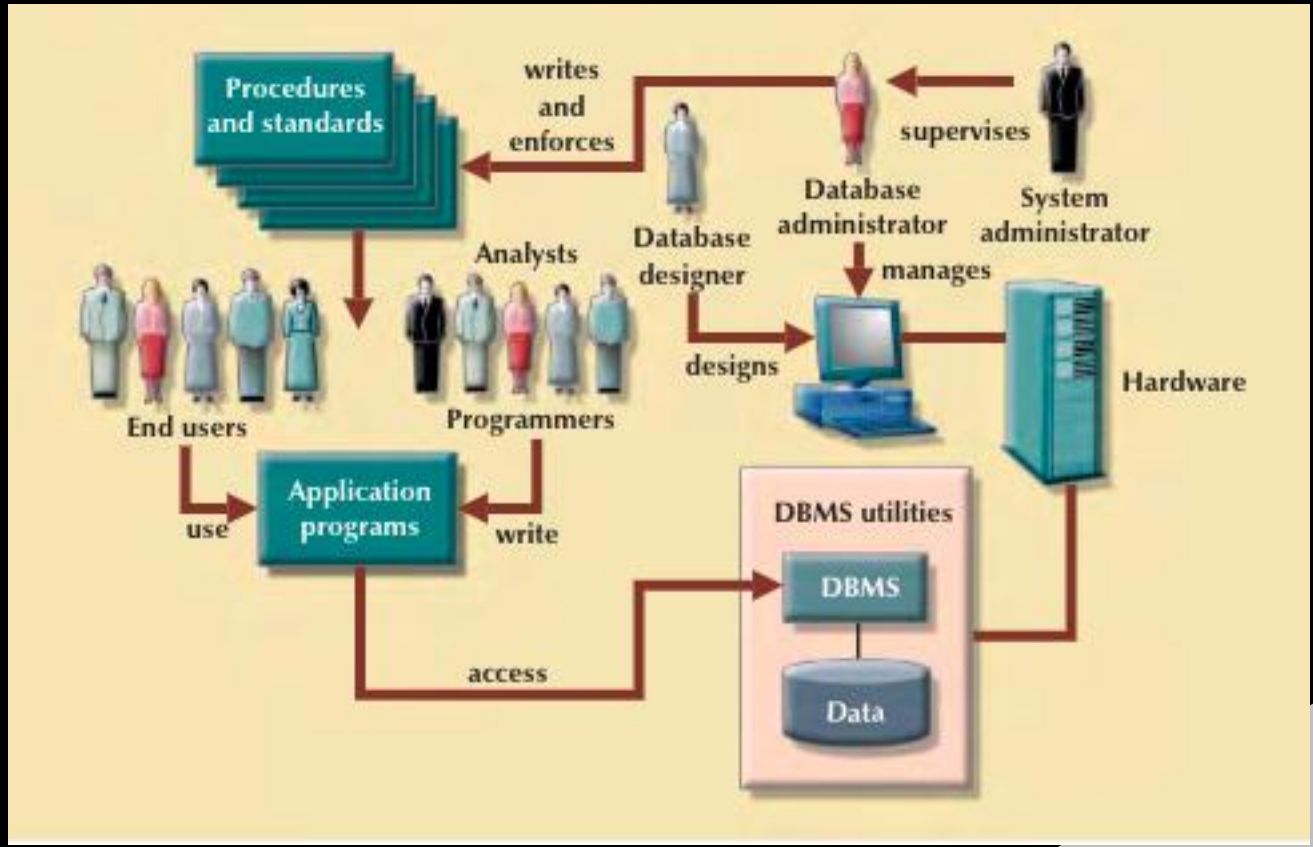
on a centralized server machine



The Database System Environment

Database System refers to an organization of components that define and regulate the collection, storage, management, and use of data within a database environment.

The database system is composed of the five major parts: hardware, software, people, procedures, and data.

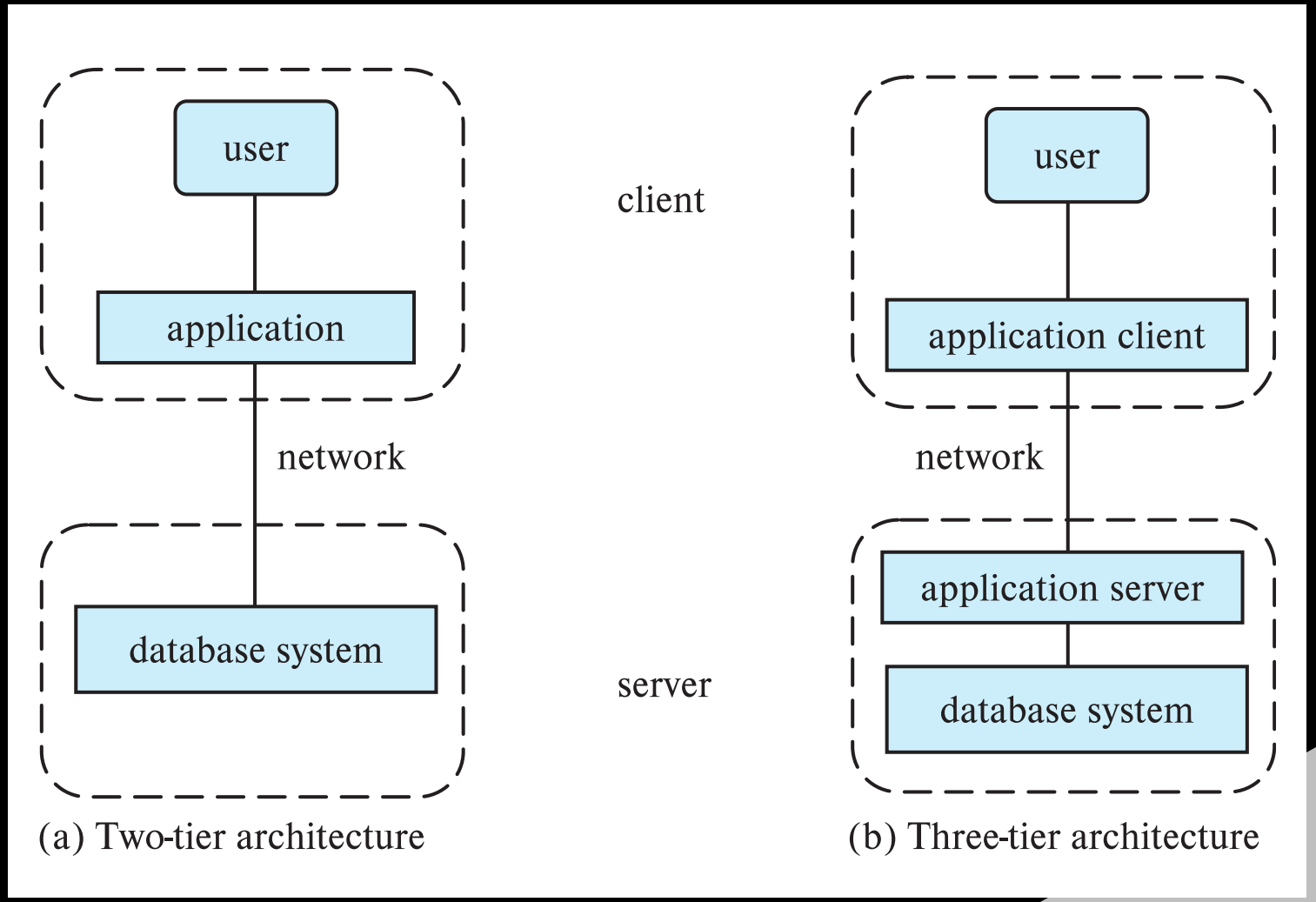


C. Coronel, S. Morris, P. Rob, 'Database System: Design, Implementation, and Management', Ninth Edition, Cengage Learning

Database Applications Architecture

Two-tier architecture -- the application resides at the client machine, where it invokes database system functionality at the server machine

Three-tier architecture -- the client machine acts as a front end and does not contain any direct database calls.



(a) Two-tier architecture

(b) Three-tier architecture

Database Users