UTS:ENGINEERING AND INFORMATION TECHNOLOGY

UTS
UNIVERSITY OF TECHNOLOGY, SYDNEY

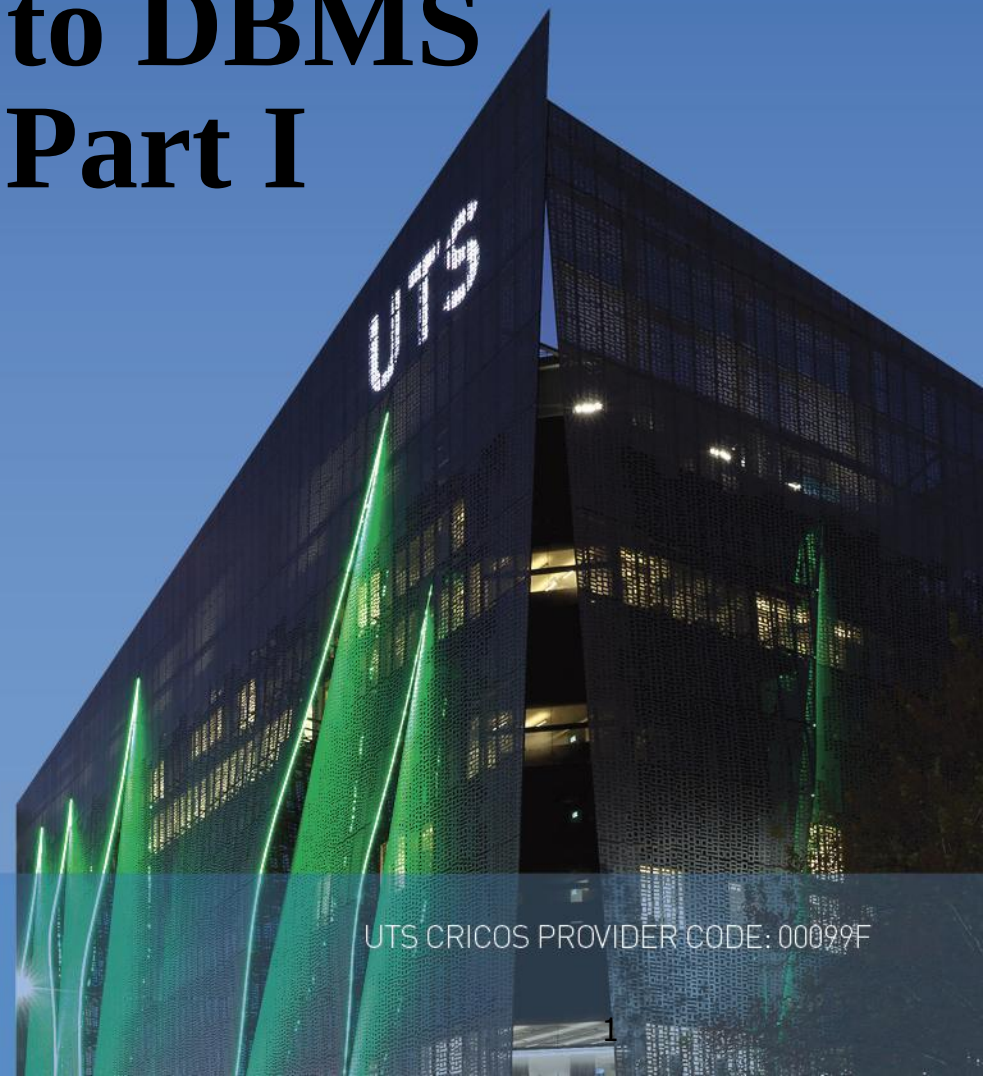
lecture 1: Introduction to DBMS and Data Modeling Part I

Modern Database Management

11th Edition, International Edition

Chapter 1 & 2: Modeling Data in the Organization

Jeffrey A. Hoffer, V. Ramesh, Heikki Topi



Innovation in practice

eng.uts.edu.au • it.uts.edu.au

Module 1: Self study
Why Databases & Introduction to DBMS

Note: This section is designed to provide you with extra information and give you a perspective to the subject. Information in this section will not be examined.

Traditional File Processing system Creates Duplicate Data

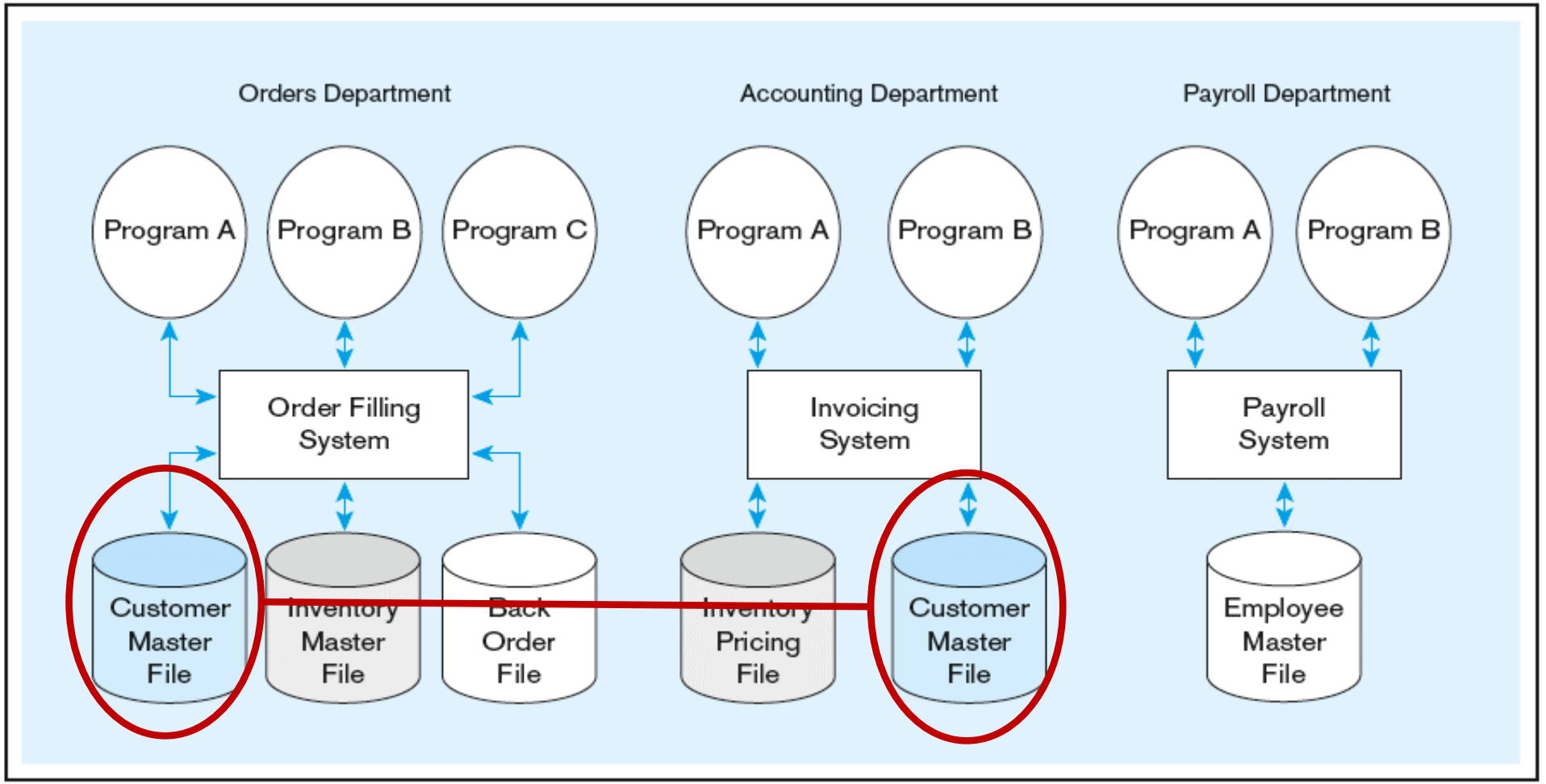
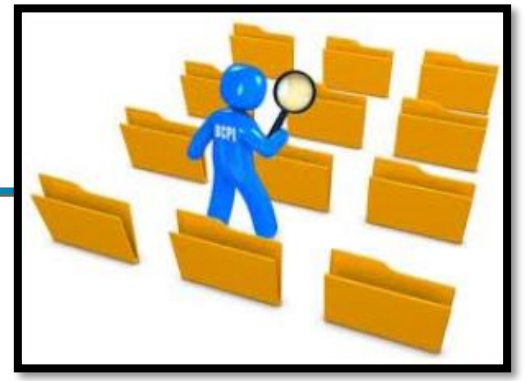


FIGURE 1-2 Old file processing systems at Pine Valley Furniture Company

Disadvantages of File Processing



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

Problems with Data Dependency

- Each application programmer must maintain his/her 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 (Duplication of Data)

- Waste of space to have duplicate data
- Causes more maintenance headaches
- The biggest problem:
  - **Data changes in one file could cause inconsistencies**
  - Compromises in ***data integrity***

Problem with Spreadsheets: Redundancy

- In a spreadsheet, each row is intended to stand on its own.
  As a result, the same information may be entered several times

- E.g. The BoyGirl

  Compare the BoyGirl spreadsheet to BoyGirl Relational database …

BoyGirl Database: Spreadsheet

- **NOTE:** Not a good design!

BoyGirlNo	BoyName	BoyMobile	GirlName	GirlMobile
1	Adam	0414 1236	Alice	0414 1234
2	Bob	0414 1237	Bonnie	0414 1235
3	Charlie	0414 1238	Bonnie	0414 1235
4	Dennis	<null>	<null>	<null>

- **One** girl can contact **many** boys, so …
- .. we store **redundant** data about Bonnie.

A BoyGirl Relational Database

BoyNo	BoyName	BoyMobile	GirlNo
1	Adam	0414 1236	1
2	Bob	0414 1237	2
3	Charlie	0414 1238	2
4	Dennis	<null>	<null>

GirlNo	GirlName	GirlMobile
1	Alice	0414 1234
2	Bonnie	0414 1235

Primary keys

Foreign keys

Primary keys

- No redundant data about Bonnie
- Two tables with a **One-to-Many** relationship
- … linked by a **Foreign Key**

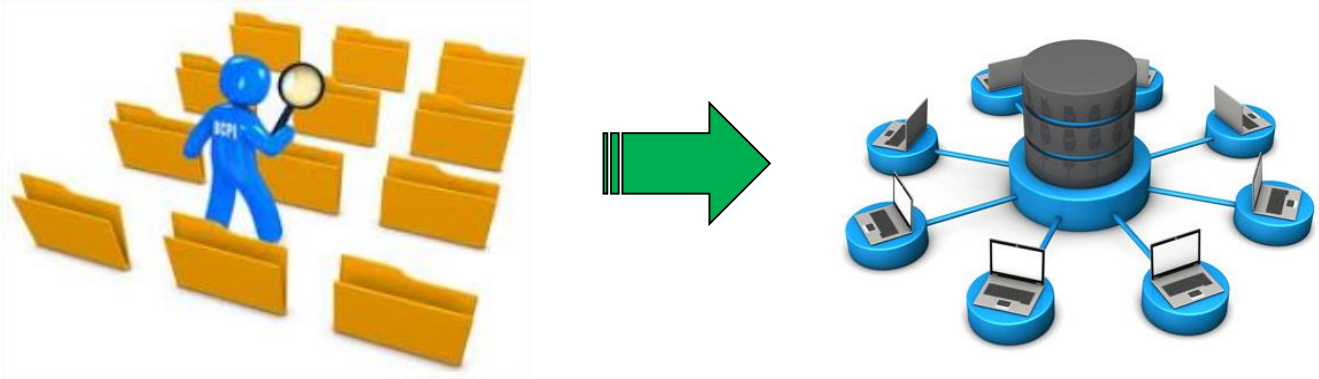
The Problem of Storing Redundant Data

- **Delete:** some but not all instances of data
- **Update:** some but not all instances of data
- **Insert:** multiple data entry can introduce inconsistency

- Also – very important! – multiple data entry is expensive

SOLUTION: The DATABASE Approach

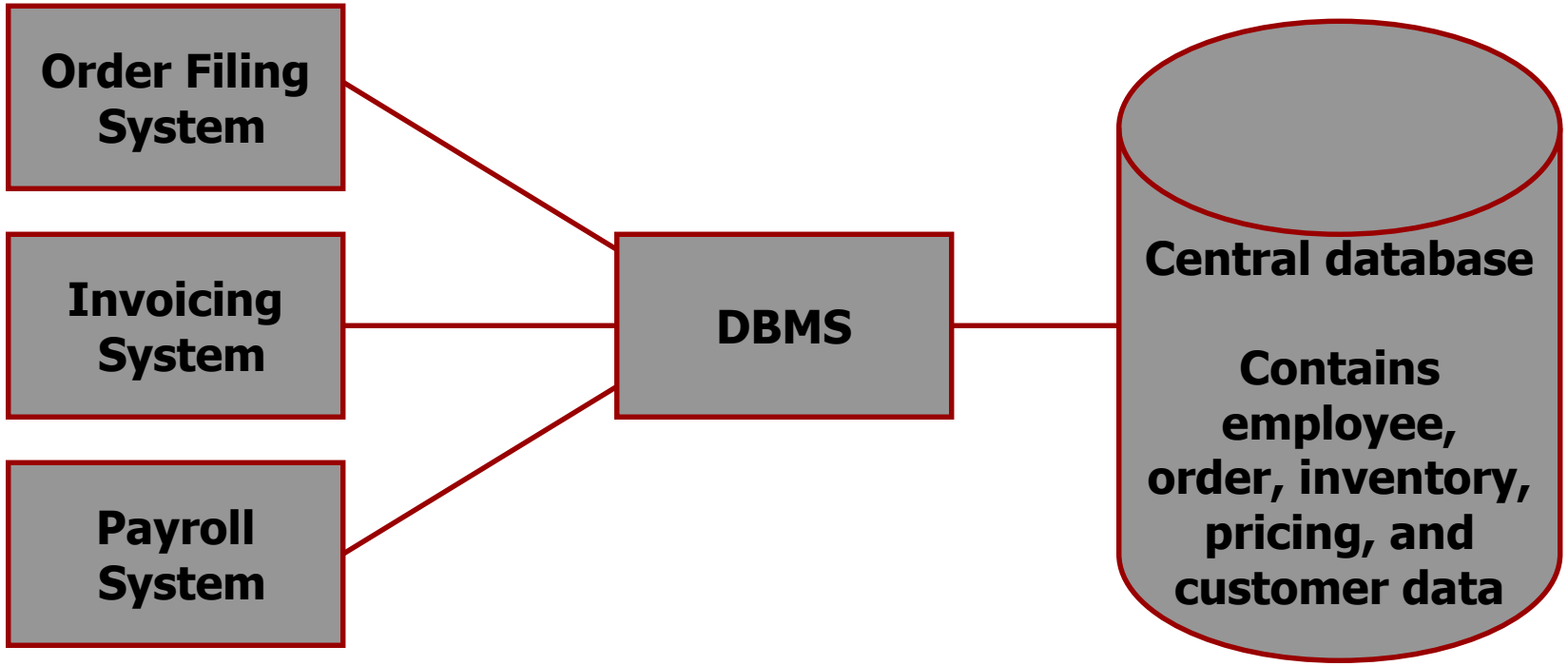
- Central repository of shared data
- Data is managed by a controlling agent
- Stored in a standardized, convenient form



Requires a Database Management System (DBMS)

Database Management System

- **A software system that is used to define, create, maintain a database, and provide controlled access to user databases**



DBMS manages data resources like an operating system manages hardware resources

Advantages of the Database Approach

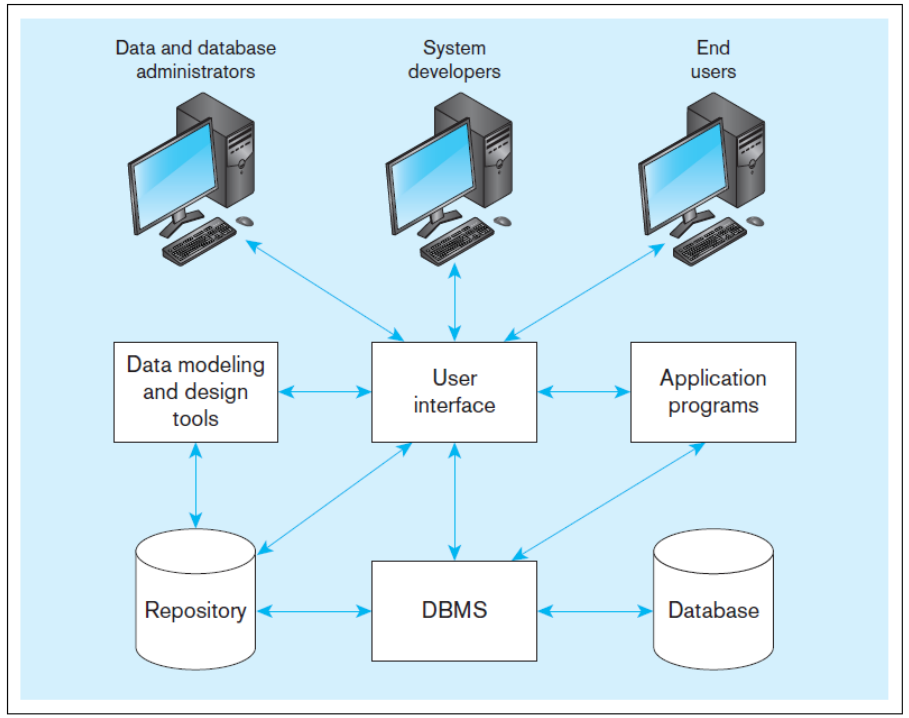
- Program-data independence
- Planned data redundancy
- Improved data consistency
- Improved data sharing
- Increased application development productivity
- Enforcement of standards
- Improved data quality
- Improved data accessibility and responsiveness
- Reduced program maintenance
- Improved decision support

Costs and Risks of the Database Approach

- New, specialized personnel
- Installation and management cost and complexity
- Conversion costs
- Need for explicit backup and recovery
- Organizational conflict

Components of the Database Environment (figure 1-5)

- **Data modeling and design tools** –automated tools used to design databases and application programs
- **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, graphical displays, menus, etc. for user
- **Data/Database 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



Database Schema based on "American National Standards Institute-Standards Planning And Requirements Committee (ANSI-SPARC)"

- **External Schema**
  - User Views
  - Subsets of Conceptual Schema

  Different people have different views of the database these are the external schema

- **Conceptual Schema**
  - E-R models (Lectures 1, 2 and 3)

- **Internal Schema**
  - Logical structures (Lecture 4)
  - Physical structures

  The internal schema is the underlying design and implementation

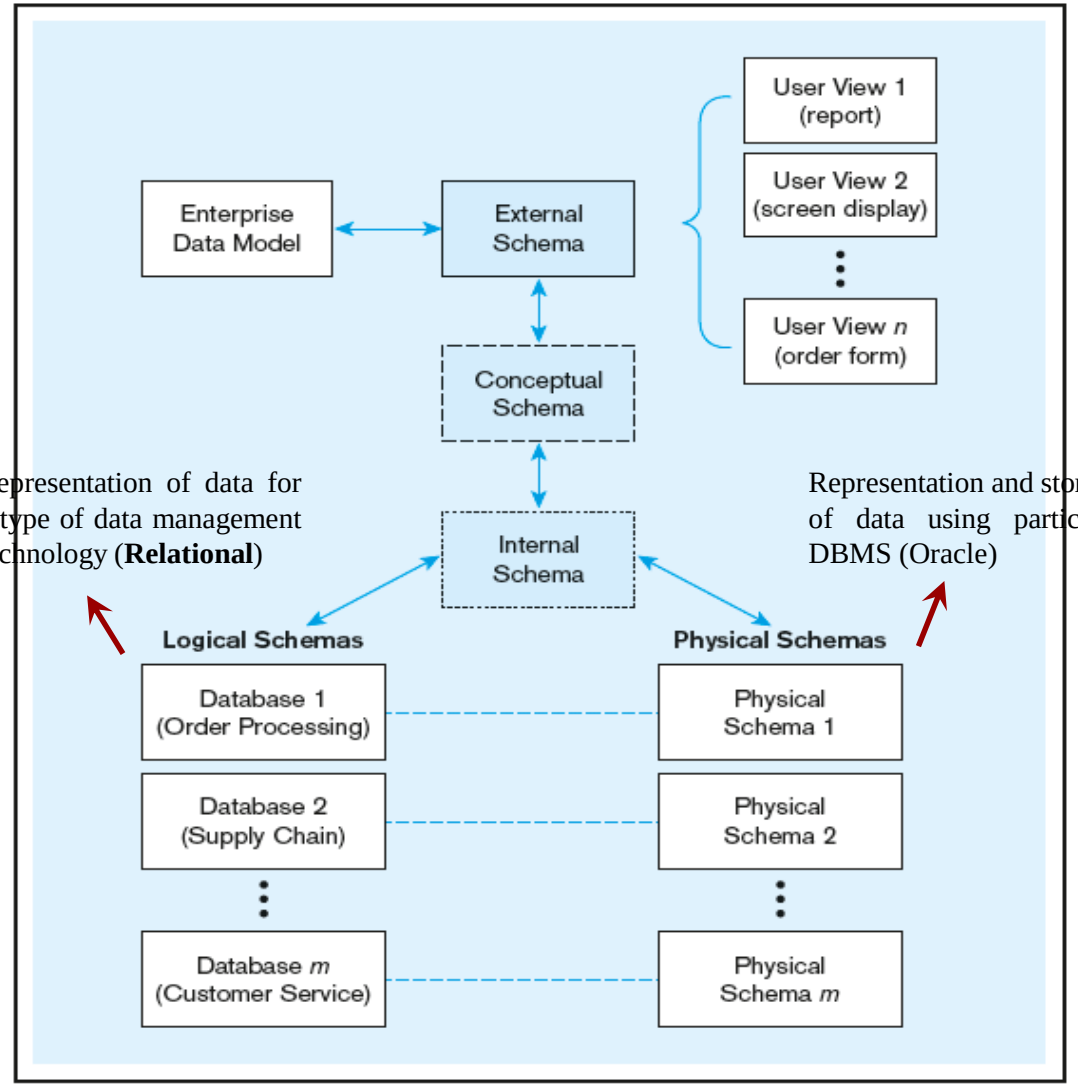


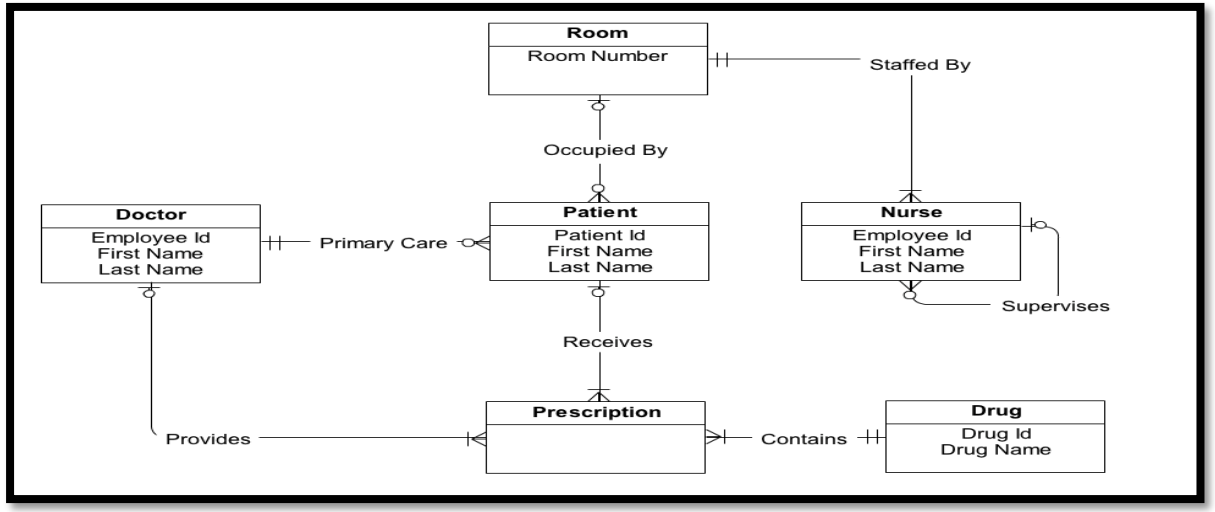
Figure 1-9 Three-schema architecture

ANSI-SPARC Architecture

External Level (User Views): A user's view of the database describes a part of the database that is relevant to a particular user. It excludes irrelevant data as well as data which the user is not authorized to access.

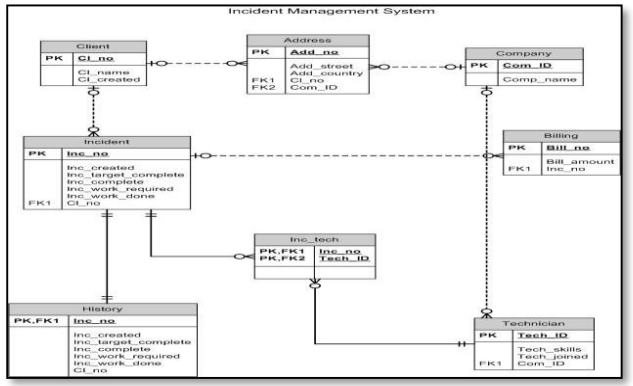
ANSI-SPARC Architecture

- **Conceptual Level:** The conceptual level is a way of describing what data is stored within the whole database and how the data is inter-related. The conceptual level does not specify how the data is physically stored.
- It is a detailed, technology-independent specification of the overall structure of organizational data (could be represented by an entity-relationship diagram (ERD)).



ANSI-SPARC Architecture

- **Internal Level:** The internal level involves how the database is physically represented on the computer system. It describes how the data is actually stored in the database and on the computer hardware.
  - **Logical data model (or schema):** Data model specific to a particular database approach, for example the relational data model. In the case of a relational data model, elements of the logical model include tables, columns, rows, primary and foreign keys, as well as constraints.



- **Physical data model (or schema):** A set of specifications that detail how data from a logical data model are stored in a computer's secondary memory for a specific database management system.