

204222 - Fundamentals of Database Systems

Chapter 1

Databases and Database Users

Adapted for 204222

by Areerat Trongratsameethong

Addison-Wesley
is an imprint of

PEARSON

Copyright © 2011 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

Chapter 1 Outline

- Introduction
- An Example
- Characteristics of the Database Approach
- Actors on the Scene
- Workers behind the Scene
- Advantages of Using the DBMS Approach
- A Brief History of Database Applications
- When Not to Use a DBMS

Overview

- **Traditional database applications**

- Store textual or numeric information

- ฐานข้อมูลแบบดั้งเดิมใช้จัดเก็บข้อมูลที่อยู่ในรูปของอักขระ และตัวเลข

- **Multimedia databases**

- Store images, audio clips, and video streams digitally

- ฐานข้อมูลสื่อผสม (ฐานข้อมูลมัลติมีเดีย) ใช้จัดเก็บข้อมูลที่อยู่ในรูปของ รูปภาพ เสียง และวิดีโอ

- **Geographic information systems (GIS)**

- Store and analyze maps, weather data, and satellite images

- ระบบสารสนเทศภูมิศาสตร์ ใช้จัดเก็บและวิเคราะห์แผนที่ ข้อมูลเกี่ยวกับอากาศ และภาพถ่ายจากดาวเทียม

Overview (cont'd.)

- **Data warehouses and online analytical processing (OLAP) systems**
 - Extract and analyze useful business information from very large databases
 - Support decision making
 - **Data Warehouses และ OLAP** ใช้วิเคราะห์ข้อมูลจากฐานข้อมูลขนาดใหญ่ เพื่อเป็นประโยชน์ทางธุรกิจ และช่วยในการตัดสินใจในการดำเนินธุรกิจ
- **Real-time and active database technology**
 - Control industrial and manufacturing processes
 - ฐานข้อมูลที่ใช้ควบคุมเครื่องจักร และขบวนการต่างๆ ในโรงงานแบบ **Real-time** (แบบ ทันทีทันใด)

Introduction

- **Database**

- ฐานข้อมูลคือ ที่ที่ใช้สำหรับจัดเก็บข้อมูลที่มีความสัมพันธ์กัน
- ข้อมูลคือ ข้อเท็จจริงที่มีความหมายชัดเจนที่ถูกรับบันทึก หรือจัดเก็บ
- **Database** มีคุณสมบัติดังต่อไปนี้
 - นำเสนอบางมุมมองของความเป็นจริงที่เกิดขึ้น บางครั้งเรียกว่า **Miniworld** หรือ **Universe of Discourse (UoD)**
 - ข้อมูลที่จัดเก็บมีความหมายที่สอดคล้องกันอย่างเหนียวแน่น และไปในทิศทางเดียวกัน
 - สร้างขึ้นเพื่อวัตถุประสงค์ที่เฉพาะเจาะจง

- **Example of a large commercial database**

- ตัวอย่างฐานข้อมูลขนาดใหญ่ในวงการธุรกิจคือ ฐานข้อมูลของ **Amazon.com**

Introduction (cont'd.)

- **Database Management System (DBMS)**

- DBMS คือที่รวมของโปรแกรมที่สร้างขึ้นเพื่อให้ผู้ใช้สามารถสร้างและจัดการฐานข้อมูล
- ในการสร้างฐานข้อมูล จะต้องกำหนดชนิดของข้อมูล (Data Type) โครงสร้างของข้อมูล (Structure) และข้อจำกัด (Constraint) ของข้อมูลที่จะจัดเก็บในฐานข้อมูล

- **Meta-data**

- Meta-data คือ คำนิยาม หรือคำอธิบายข้อมูลที่อยู่ในฐานข้อมูล ซึ่งถูกจัดเก็บโดย DBMS ในรูปแบบของ Database Catalog หรือ Data Dictionary

Introduction (cont'd.)

- **Manipulating a database**

- การจัดการข้อมูลในฐานข้อมูล เช่น การสืบค้นข้อมูล การปรับปรุงข้อมูล การสร้างรายงานต่างๆ

- **Sharing a database**

- DBMS อนุญาตให้ผู้ใช้งานหลายคน หรือโปรแกรมหลายโปรแกรม เข้าใช้งานฐานข้อมูลพร้อมๆกันได้

- **Application program**

- โปรแกรมประยุกต์สามารถเข้าใช้งานฐานข้อมูลโดยการส่งคำสั่งสืบค้นข้อมูล (query) ไปยัง DBMS

Introduction (cont'd.)

- **Transaction**

- **Transaction** ประกอบด้วยส่วนของการดำเนินการ เพื่อบริหารจัดการฐานข้อมูล ที่อาจทำให้ข้อมูลในฐานข้อมูลบางส่วนถูกอ่าน หรือถูกเขียน

- **Protection includes:** การป้องกันฐานข้อมูล ประกอบด้วย

- **System protection:** ป้องกันจากความผิดปกติ หรือความเสียหาย เช่น ดิสก์เกิดความเสียหาย เป็นต้น

- **Security protection:** ป้องกันจากผู้ใช้ที่ไม่ได้รับอนุญาต หรือจากการบุกรุกเพื่อทำให้ข้อมูลเสียหาย

- **Maintain the database system**

- ระบบฐานข้อมูลเื่ออำนวยความสะดวกให้มีการบำรุงรักษาระบบ และปรับเปลี่ยนข้อมูลได้ตลอดเวลา เนื่องจากความต้องการของผู้ใช้ระบบมีการเปลี่ยนแปลงได้ตลอดเวลา

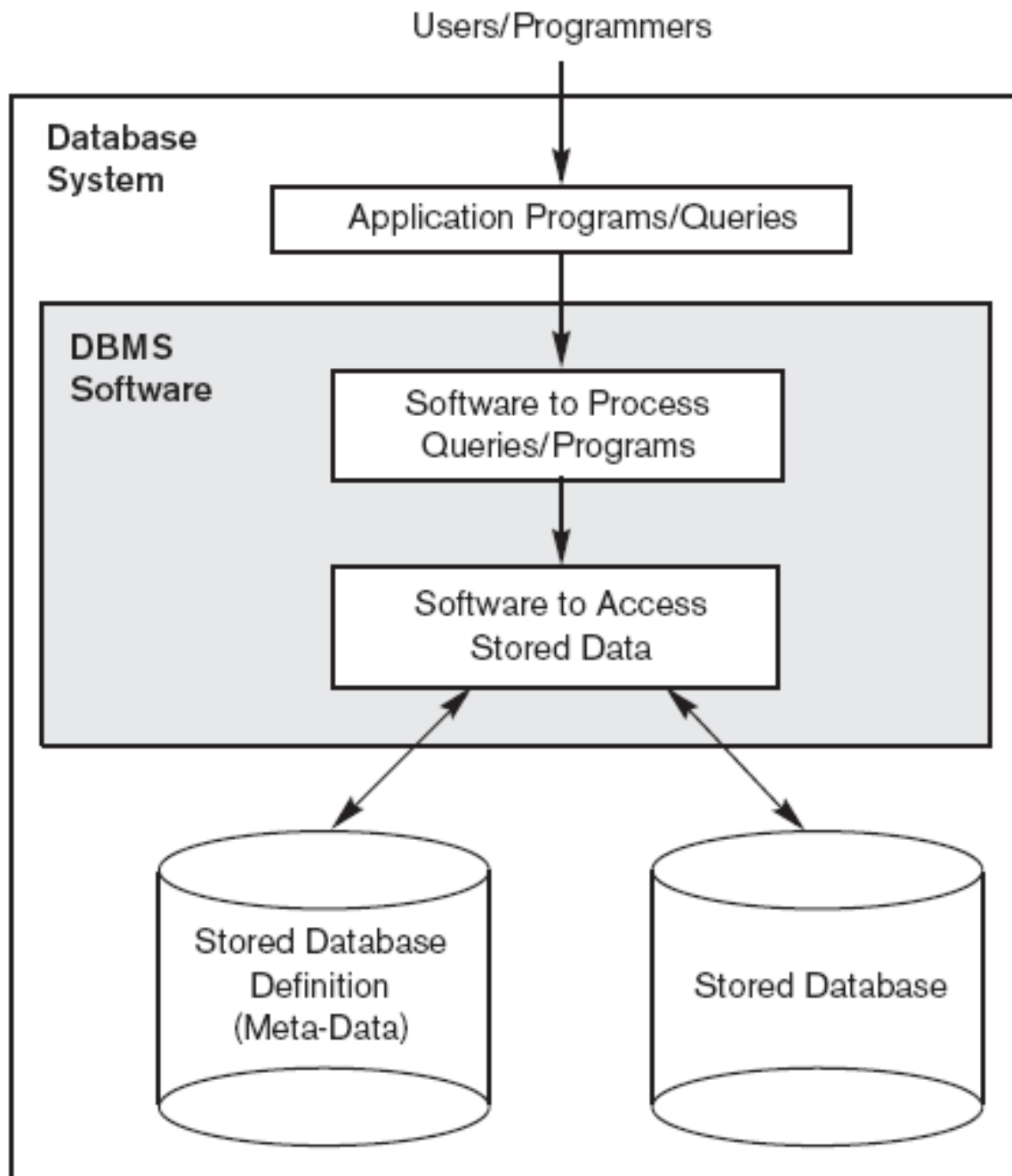


Figure 1.1
A simplified database system environment.

An Example

- UNIVERSITY database
 - Information concerning (เกี่ยวข้องกับ) students, courses, and grades in a university environment
- **Data records**
 - STUDENT
 - COURSE
 - SECTION
 - GRADE_REPORT
 - PREREQUISITE

An Example (cont'd.)

- Specify structure of records of each file by specifying **data type** for each **data element**
- กำหนดโครงสร้างของ **records** ของแต่ละ **file** โดยการ กำหนด **data type** สำหรับข้อมูลแต่ละ **element**
 - String of alphabetic characters
 - Integer
 - Etc.

An Example (cont'd.)

- Construct UNIVERSITY database
 - Store data to represent each student, course, section, grade report, and prerequisite as a record in appropriate file
- Relationships among the records: ความสัมพันธ์ระหว่าง records
- Manipulation involves querying and updating: การจัดการข้อมูลเกี่ยวข้องกับการสอบถามข้อมูล หรือสืบค้นข้อมูล (query) และการปรับปรุงข้อมูล

An Example (cont'd.)

- Examples of queries:
 - Retrieve the transcript
 - List the names of students who took the section of the 'Database' course offered in fall 2015 and their grades in that section
 - List the prerequisites of the 'Database' course
- Examples of updates:
 - Change the class of 'Smith' to sophomore
 - Create a new section for the 'Database' course for this semester
 - Enter a grade of 'A' for 'Smith' in the 'Database' section of last semester

An Example (cont'd.)

- Phases for designing a database:
 - **Requirements specification and analysis**
 - **Conceptual design**
 - **Logical design**
 - **Physical design**
- จะได้เรียนในบทถัดๆไป

STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Figure 1.2

A database that stores student and course information.

RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

Figure 1.3

An example of a database catalog for the database in Figure 1.2.

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
....
....
....
Prerequisite_number	XXXXNNNN	PREREQUISITE

Note: Major_type is defined as an enumerated type with all known majors.

XXXXNNNN is used to define a type with four alpha characters followed by four digits.

Characteristics of the Database Approach

วิวัฒนาการของการจัดเก็บข้อมูล

- **Traditional file processing**

- การจัดเก็บข้อมูลแบบดั้งเดิม จัดเก็บในรูปแบบของ **file** โดยที่ผู้ใช้แต่ละคน กำหนดข้อมูล และจัดเก็บข้อมูลของตัวเองสำหรับใช้ใน **software application** ที่เฉพาะเจาะจงของแต่ละคน

- **Database approach**

- การจัดเก็บข้อมูลในรูปแบบของฐานข้อมูล โดยที่จัดเก็บข้อมูลไว้ที่เดียวกัน และผู้ใช้หลายๆคนสามารถเข้าถึง และใช้ข้อมูลร่วมกันได้

Characteristics of the Database Approach (cont'd.)

- Main characteristics of database approach:

ลักษณะสำคัญของการจัดเก็บข้อมูลในรูปแบบของฐานข้อมูล

- **Meta-data: Data about data** คือ คำนิยาม หรือคำอธิบาย ข้อมูลที่จัดเก็บอยู่ในฐานข้อมูล ถูกจัดเก็บไว้ในฐานข้อมูล เช่นเดียวกัน (ข้อมูลที่ใช้อธิบายเกี่ยวกับตัวข้อมูลเอง)
- ข้อมูล และโปรแกรม ถูกจัดเก็บแยกออกจากกัน
- สนับสนุนข้อมูลในหลายมุมมอง คือ ข้อมูลสำหรับผู้ใช้ต่างแผนก สามารถเห็นข้อมูลต่างมุมมองกันได้
- ผู้ใช้หลายคน สามารถใช้ข้อมูลร่วมกัน และข้อมูลสามารถถูกประมวลผลโดยผู้ใช้หลายๆคนได้พร้อมกัน

Self-Describing Nature of a Database System

- Database system contains complete definition of structure and constraints: ระบบฐานข้อมูล ประกอบด้วย การกำหนดโครงสร้าง และข้อจำกัดของข้อมูล
- **Meta-data (Data about data):** ใช้อธิบายโครงสร้างของฐานข้อมูล
- Database catalog used by: Database catalog ถูกใช้โดย
 - DBMS software
 - Database users

Insulation Between Programs and Data

- **Program-data independence:** โปรแกรม และข้อมูลแยกอิสระจากกัน
 - โครงสร้างของข้อมูลถูกจัดเก็บไว้ใน **DBMS Catalog** ซึ่งแยกออกจากตัวโปรแกรมที่ต้องการเข้าถึงข้อมูล ทำให้การเปลี่ยนแปลงโครงสร้างข้อมูลไม่กระทบกับโปรแกรม (โปรแกรมไม่จำเป็นต้อง **compile** ใหม่)
 - **Operations specified in two parts:** การดำเนินการกับข้อมูลสามารถทำผ่าน
 - **Interface:** โดยการระบุชื่อ **operation, arguments,** และ **data types** ของ **arguments**
 - **Operation** สามารถถูกแก้ไขได้โดยไม่มีผลกระทบต่อ **interface**

Data Abstraction

- **Data abstraction**

- โปรแกรม และข้อมูล เป็นอิสระจากกัน
- โปรแกรม และ **operation** เป็นอิสระจากกัน

- **Conceptual representation of data**

- ข้อมูลถูกนำเสนอในรูปของ **conceptual** คือ ข้อมูลไม่ได้แสดงรายละเอียดของการจัดเก็บ หรือไม่ได้แสดงรายละเอียดของ **operations** ที่ใช้ดำเนินการกับข้อมูลว่าติดตั้งอย่างไร

- **Data model**

- **Data Model** คือรูปแบบของข้อมูลที่ใช้ในการนำเสนอข้อมูลในรูปแบบ **conceptual**

Support of Multiple Views of the Data

- **View:** ผู้ใช้รับรู้ข้อมูลในมุมมองที่แตกต่างกันได้
 - เป็น Subset ของ database
 - ประกอบด้วยข้อมูลเสมือนจริงที่ได้มาจากข้อมูลที่จัดเก็บอยู่ในฐานข้อมูล แต่ข้อมูลที่อยู่ในรูปแบบของ **view** ไม่ได้จัดเก็บจริงในฐานข้อมูล
- **Multiuser DBMS**
 - ผู้ใช้สามารถเข้าถึงข้อมูลจาก **application** ที่แตกต่างกันได้
 - **DBMS** มีสิ่งอำนวยความสะดวกสำหรับการกำหนด **views** ที่แตกต่างกันให้ **users** ที่หลากหลายได้

Sharing of Data and Multiuser Transaction Processing

- Allow multiple users to access the database at the same time: อนุญาตให้ user หลายคนเข้าถึงข้อมูลในฐานข้อมูลในเวลาเดียวกันได้
- **Concurrency control software: software ที่ใช้ควบคุมการทำงานพร้อม ๆ กัน**
 - Concurrency control software ช่วยทำให้มั่นใจได้ว่า ผู้ใช้หลายคนที่พยายามที่จะปรับปรุงข้อมูลในเวลาเดียวกัน สามารถควบคุมการปรับปรุงข้อมูลให้ถูกต้องตามความต้องการได้ ส่งผลให้การปรับปรุงข้อมูลมีความถูกต้อง
- **Online transaction processing (OLTP) application: สนับสนุนโปรแกรมที่ประมวลผลข้อมูลแบบ online**

Sharing of Data and Multiuser Transaction Processing (cont'd.)

- **Transaction**

- เป็นหัวใจสำคัญของ **database applications**

- ขบวนการที่สามารถดำเนินการกับข้อมูลที่จัดเก็บอยู่ในหนึ่งฐานข้อมูล หรือหลายฐานข้อมูลได้

- **Isolation property (คุณสมบัติ):** แต่ละ **transaction** จะถูกดำเนินการแยกจาก **transaction** อื่นๆ

- **Atomicity property:** หากมีการดำเนินการกับข้อมูล ต้องดำเนินการให้เสร็จสมบูรณ์ หากดำเนินการไม่สมบูรณ์ ต้องคืนสภาพเดิมของข้อมูล

Transaction ประกอบด้วยส่วนของการดำเนินการ เพื่อบริหารจัดการฐานข้อมูล ที่อาจทำให้ข้อมูลในฐานข้อมูลบางส่วนถูกอ่าน หรือถูกเขียน

Actors on the Scene

คนทำงานเบื้องหน้า

- **Database administrators (DBA)** are responsible for: DBA ทำหน้าที่
 - กำหนดสิทธิในการเข้าถึงข้อมูลในฐานข้อมูล
 - ผู้ประสานงาน และผู้ดูแลการใช้ฐานข้อมูลของผู้ใช้
 - จัดหาทรัพยากรด้าน **software** และ **hardware** ที่สนับสนุนการใช้งานฐานข้อมูล
- **Database designers** are responsible for: ทำหน้าที่
 - กำหนดข้อมูลที่จะจัดเก็บในฐานข้อมูล
 - เลือกโครงสร้างข้อมูลที่เหมาะสมที่จะใช้สำหรับนำเสนอข้อมูล และจัดเก็บข้อมูล

Actors on the Scene (cont'd.)

- **End users:** บุคคลที่เข้าใช้งานฐานข้อมูล
 - ประเภทของ **End Users**
 - **Casual end users: user** ที่ใช้งานปกติทั่วไป
 - **Naive or parametric end users: user** เฉพาะ เช่น **Bank Teller, Reservation Agency**
 - **Sophisticated end users: user** ที่ต้องการ การประมวลผลที่ซับซ้อน เช่น นักคณิตศาสตร์ นักวิเคราะห์ธุรกิจ
 - **Standalone users: user** ที่ใช้งานคนเดียว เช่น โปรแกรมสำเร็จรูปต่างๆ เช่น โปรแกรมบัญชี เป็นต้น
- **System analysts**
 - เก็บรวบรวม และวิเคราะห์ความต้องการของผู้ใช้
- **Application programmers**
 - สร้างโปรแกรมตามความต้องการของผู้ใช้

Workers behind the Scene

คนทำงานเบื้องหลัง

- **DBMS system designers and implementers**
 - ออกแบบ และติดตั้ง DBMS software package
- **Tool developers**
 - ออกแบบ และติดตั้งเครื่องมือ (tools) ต่างๆ ที่ช่วยสนับสนุนการใช้งานฐานข้อมูล
- **Operators and maintenance personnel**
 - Run โปรแกรม และบำรุงรักษา hardware and software สำหรับระบบฐานข้อมูล

Advantages of Using the DBMS Approach

- **Controlling redundancy:** สามารถควบคุมความซ้ำซ้อนของข้อมูลได้ โดยการทำ
 - **Data normalization**
 - **Denormalization:** ในบางครั้งก็จำเป็นที่จะต้องทำ Denormalization (จัดเก็บข้อมูลซ้ำซ้อน) เพื่อเพิ่มความเร็วในการสืบค้นข้อมูล
- **Restricting unauthorized access:** สามารถกำหนดสิทธิ์ในการเข้าถึงข้อมูล เพื่อป้องกันความเสียหายที่จะเกิดขึ้นจากผู้ที่ไม่มีความรู้สิทธิ์
 - มี subsystem ที่ใช้สำหรับกำหนดความปลอดภัยของข้อมูล และกำหนดสิทธิ์ในการเข้าใช้ข้อมูลของผู้ใช้
 - **Privileged software:** มี software ที่ใช้จัดการสิทธิ์ (authority) ต่างๆ

Advantages of Using the DBMS Approach (cont'd.)

- Providing **persistent** storage for program objects: สามารถจัดเก็บ object แบบถาวรได้
 - Object ที่มีความซับซ้อนใน C++ ที่อยู่ใน memory ขณะที่ run โปรแกรม สามารถถูกจัดเก็บอย่างถาวรใน object-oriented Database
 - **Impedance mismatch problem**
 - Object-oriented database systems typically offer data structure **compatibility**: OODB คือ สัญลักษณ์ของการเสนอโครงสร้างข้อมูลที่สามารถเข้าถึงได้ (ซึ่งมีโครงสร้างที่แตกต่างจาก Relational Database)

Impedance mismatch problem: ปัญหาความเข้ากันไม่ได้

Advantages of Using the DBMS Approach (cont'd.)

- Providing storage structures and search techniques for efficient query processing: มีโครงสร้างของที่ใช้จัดเก็บข้อมูล และเทคนิคในการค้นหาข้อมูล เพื่อประสิทธิภาพในการค้นหาข้อมูลได้เร็วขึ้น โดยมีเทคนิคต่างๆดังต่อไปนี้
 - **Indexes**
 - **Buffering and caching**
 - **Query processing and optimization**

Advantages of Using the DBMS Approach (cont'd.)

- Providing backup and recovery
 - **Backup and recovery** (การกู้ข้อมูล หรือคืนสภาพข้อมูล) **subsystem** of the DBMS is responsible for recovery
- Providing multiple user interfaces
 - **Graphical user interfaces (GUIs)**
- Representing complex relationships among data
 - May include numerous varieties of data that are interrelated in many ways: สามารถนำเสนอความสัมพันธ์ของข้อมูลที่มีความหลากหลาย และเกี่ยวของสัมพันธ์กันหลายๆทาง

Advantages of Using the DBMS Approach (cont'd.)

- Enforcing **integrity constraints**
 - **Referential integrity** constraint
 - Every section record must be related to a course record
 - **Key or uniqueness** constraint
 - Every course record must have a unique value for Course_number
 - **Business rules**
 - **Inherent rules** of the data model: กฎที่สืบทอดมาจาก data model เช่น ความสัมพันธ์ระหว่างสอง entities

Advantages of Using the DBMS Approach (cont'd.)

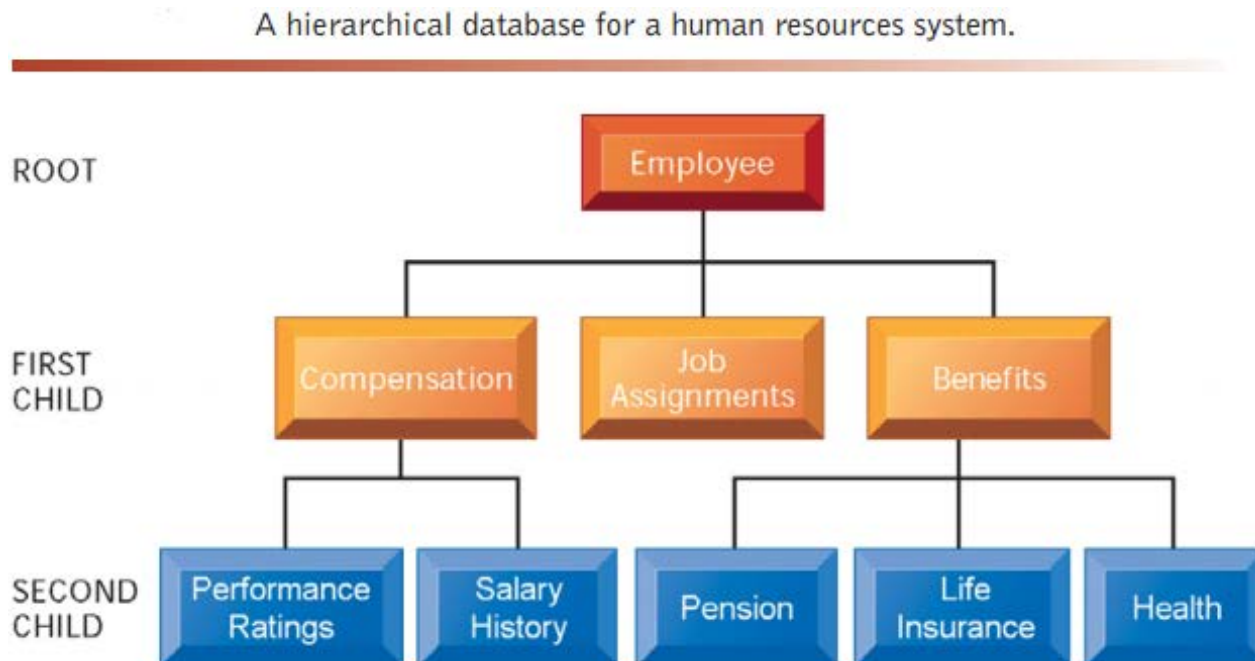
- Permitting inferences (การอนุมาน การสรุป) and actions using rules
 - **Deductive database systems**
 - Provide capabilities for defining deduction **rules**
 - Inferencing new information from the stored database facts
 - e.g. Create new student must under the defined rules
 - **Trigger**
 - Rule activated (กระตุ้น) by updates to the table
 - **Stored procedures**
 - More involved procedures to enforce rules

Advantages of Using the DBMS Approach (cont'd.)

- Additional implications of using the database approach
 - Reduced application development time: ช่วยลดเวลาในการพัฒนาโปรแกรม
 - Flexibility: มีความยืดหยุ่น
 - Availability of up-to-date information: ทำให้ข้อมูลเป็นปัจจุบัน
 - Economies of scale: ประหยัดค่าใช้จ่าย

A Brief History of Database Applications

- Early database applications using **hierarchical and network systems**
 - A data model in which the data is organized into a tree-like structure. The data is stored as records which are connected to one another through links.



A Brief History of Database Applications

- Providing data abstraction and application flexibility with **relational databases**
 - Separates physical storage of data from its conceptual representation
 - Provides a mathematical foundation (พื้นฐานในการคำนวณทางคณิตศาสตร์) for data representation and querying

A Brief History of Database Applications (cont'd.)

- Object-oriented applications and the need for more complex databases
 - Used in specialized applications: engineering design, multimedia publishing, and manufacturing systems
- Interchanging data on the Web for e-commerce using XML: การแลกเปลี่ยนข้อมูลบน Web สำหรับ e-commerce โดยใช้ eXtensible Markup Language (XML)
 - XML primary standard for interchanging data among various types of databases and Web pages

A Brief History of Database Applications (cont'd.)

- Extending database capabilities for new applications
 - **Enterprise resource planning (ERP)**
 - **Customer relationship management (CRM)**
- Databases versus information retrieval
 - **Information retrieval (IR)**
 - Deals with books, manuscripts, and various forms of library-based articles

When Not to Use a DBMS

- More desirable to use regular files for: เมื่อใดที่ไม่ต้องใช้ Database โดยที่ใช้ file ข้อมูลปกติแทน
 - Simple, well-defined database applications not expected to change at all: ข้อมูลมีโครงสร้างที่ไม่ซับซ้อน และไม่มี การเปลี่ยนแปลง
 - Stringent, real-time requirements that may not be met because of DBMS overhead: ระบบที่ต้องการความเร็วที่ไม่สามารถรอได้จากการเสียเวลาที่เกิดจาก overhead ของ DBMS
 - Embedded systems with limited storage capacity: ระบบฝังตัวที่มีข้อจำกัดของพื้นที่ใช้จัดเก็บข้อมูล
 - No multiple-user access to data: มีผู้ใช้ข้อมูลเพียงคนเดียว

Summary

- Database
 - Collection of related data (recorded facts)
- DBMS
 - Generalized software package for implementing and maintaining a computerized database
- Several categories of database users
- Database applications have evolved
 - Current trends: IR, Web