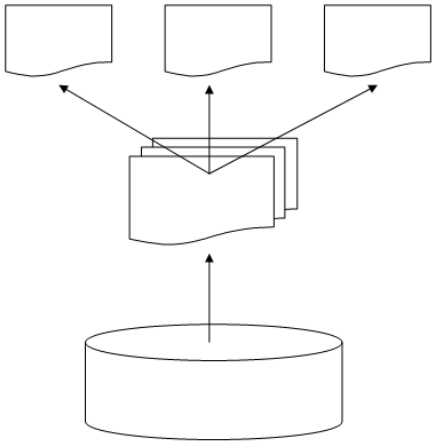


DBMS

Introduction to Database

Overview

- The definition of database systems
- The importance of database systems
- Data abstraction and database models
- Database languages
- Database system components
- Users and creators of database systems
- Database products and applications
- Database research and trends

ลำดับชั้นการเก็บข้อมูล

- บิต (bit) ย่อมาจาก Binary Digit
  - ข้อมูลในคอมพิวเตอร์ 1 บิต จะแสดงได้ 2 สถานะคือ 0 และ 1

- ไบต์ (byte) คือ การเรียงต่อกันของบิต จำนวน 8 บิต เช่น

  01100001 หมายถึง a

  01100010 หมายถึง b

- เขตข้อมูล (field) คือ การเรียงต่อกันของไบต์ เช่น
  - เขตข้อมูลชื่อ (Name)
  - เขตข้อมูลนามสกุล (Last name)

ลำดับชั้นการเก็บข้อมูล (ต่อ)

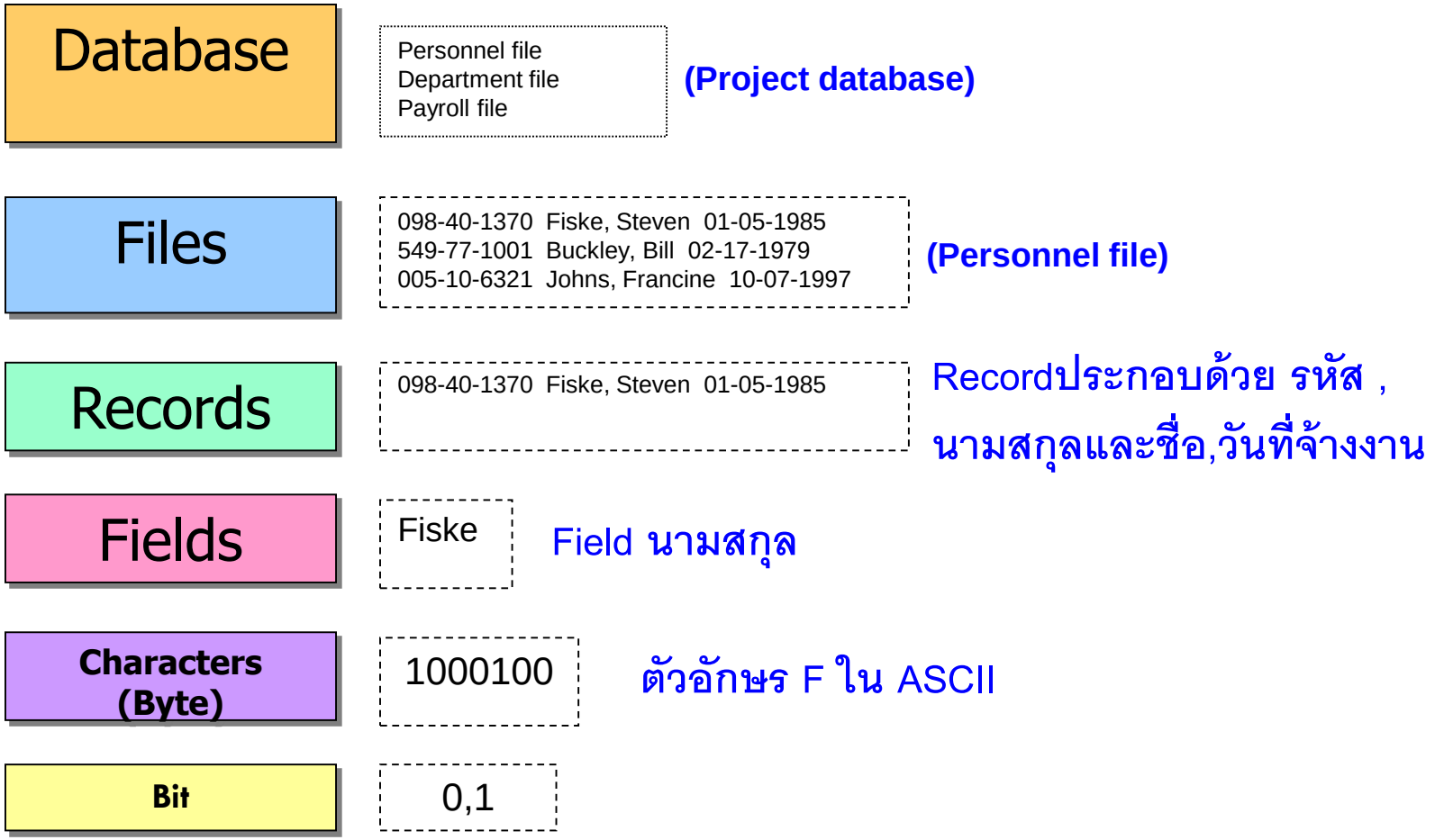
- ระเบียน(Record) คือ การเรียงต่อกันของเขตข้อมูล เช่น
  - ระเบียน ที่ 1 เก็บ ชื่อ นามสกุล วันเดือนปีเกิด ของ นักเรียนคนที่ 1
  - ระเบียน ที่ 2 เก็บ ชื่อ นามสกุล วันเดือนปีเกิด ของ นักเรียนคนที่ 2

- แฟ้มข้อมูล(File) คือ การเก็บระเบียนหลายๆ ระเบียน รวมกัน เช่น
  - แฟ้มข้อมูล นักเรียน จะเก็บ ชื่อ นามสกุล วันเดือนปีเกิด ของนักเรียน จำนวน 500 คน เป็นต้น

ลำดับชั้นการเก็บข้อมูล (ต่อ)

- ฐานข้อมูล(Database) คือ การจัดเก็บ แฟ้มข้อมูล หลายๆ แฟ้มข้อมูล ไว้ภายใต้ระบบเดียวกัน เช่น
  - เก็บแฟ้มข้อมูล นักเรียน อาจารย์ วิชาที่เปิดสอน เป็นต้น

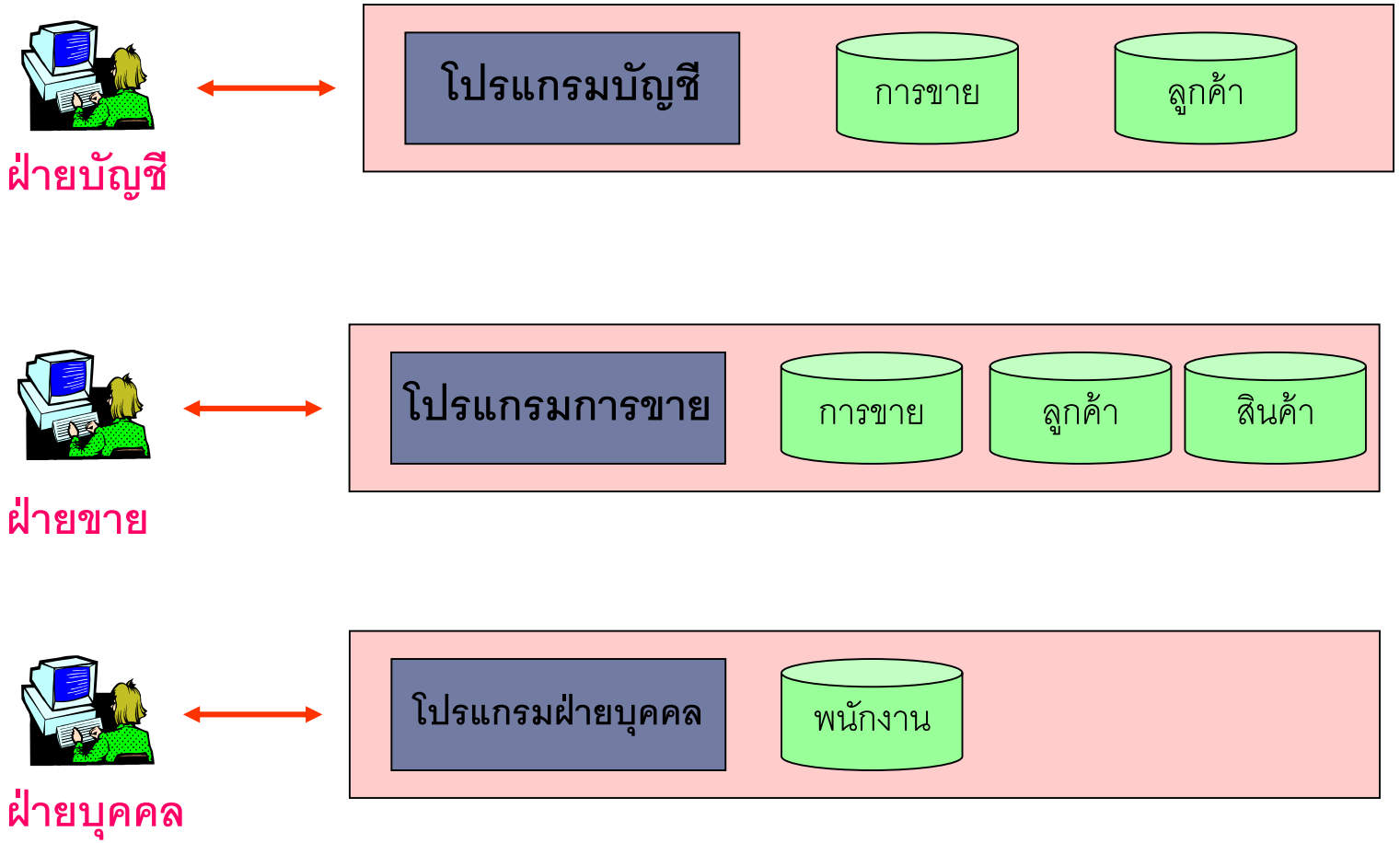
ลำดับชั้นการเก็บข้อมูล (ต่อ)



The Importance of DB Systems

- ก่อนจะมีระบบฐานข้อมูล
  - ระบบแฟ้มข้อมูล(File-based system) คือ ชุดของโปรแกรมประยุกต์ที่ให้ผู้ใช้ใช้เพื่อประมวลผลงานที่ต้องการ โดยแต่ละโปรแกรมก็จะกำหนดและจัดการแฟ้มข้อมูลของตนเอง
  - แฟ้มข้อมูลที่ใช้ในระบบไฟล์จะแยกจากกันเป็นเอกเทศ และอาจไม่มีความสัมพันธ์กัน
  - โดยส่วนใหญ่ข้อมูลและโปรแกรมมักรวมอยู่ด้วยกันเป็นแฟ้มข้อมูล

Fix 1.1 File Systems



Limitation of File-Based System

- ข้อมูลถูกแบ่งและเก็บแยกจากกัน
- ข้อมูลมีความซ้ำซ้อนกัน
- มีความขึ้นต่อกันของข้อมูล
- รูปแบบข้อมูลไม่ตรงกัน
- โปรแกรมที่ใช้งานมีความคงที่ไม่ยืดหยุ่น

Limitation of File-Based System (cont.)

- ข้อมูลถูกเก็บและเก็บแยกจากกัน

  เมื่อข้อมูลต่าง ๆ ถูกเก็บกันไว้คนละไฟล์ หากต้องการนำข้อมูลต่าง ๆ มาสร้างเป็นรายงาน โปรแกรมเมอร์ต้องสร้างไฟล์ชั่วคราว(Temporary file)ขึ้นมา เพื่อดึงข้อมูลต่าง ๆ จากไฟล์ต่าง ๆ มารวมกันก่อน แล้วค่อยสร้างเป็นรายงาน

Limitation of File-Based System (cont.)

- **ข้อมูลมีความซ้ำซ้อน**

  สืบเนื่องจากข้อมูลถูกเก็บแยกจากกัน ทำให้ไม่สามารถควบคุมความซ้ำซ้อนข้อมูลได้ ทำให้สูญเสียพื้นที่ในการจัดเก็บข้อมูลมากขึ้น และก่อให้เกิดความผิดพลาดในการดำเนินการกับข้อมูล 3 ลักษณะ ได้แก่

  - ความผิดพลาดจากการเพิ่มข้อมูล (Insertion anomalies)
  - ความผิดพลาดจากการปรับปรุงข้อมูล (Modification anomalies)
  - ความผิดพลาดจากการลบข้อมูล (Deletion anomalies)

Limitation of File-Based System (cont.)

- **มีความขึ้นต่อกันของข้อมูล**

เนื่องจากโครงสร้างทางกายภาพและการจัดเก็บข้อมูลถูกสร้างโดยการเขียนโปรแกรมประยุกต์(Application program) ดังนั้นหากต้องการเปลี่ยนแปลงโครงสร้างข้อมูล เช่น ชื่อของพนักงาน จากเดิม 20 ตัวอักษร เป็น 30 ตัวอักษร มีขั้นตอนการทำงานดังนี้

1. เปิดไฟล์หลักพนักงานเพื่ออ่านข้อมูล
2. เปิดไฟล์ชั่วคราวที่มีโครงสร้างคล้ายไฟล์หลัก แต่ปรับโครงสร้างของชื่อพนักงานจาก 20 ตัวอักษร เป็น 30 ตัวอักษร
3. อ่านข้อมูลจากไฟล์หลัก และย้ายไปเก็บไว้ในไฟล์ชั่วคราว จนกระทั่งครบทุกรายการ
4. ลบไฟล์หลักทิ้ง
5. เปลี่ยนชื่อไฟล์ชั่วครามให้ชื่อเดียวกับไฟล์หลัก

Limitation of File-Based System (cont.)

- **รูปแบบข้อมูลไม่ตรงกัน**

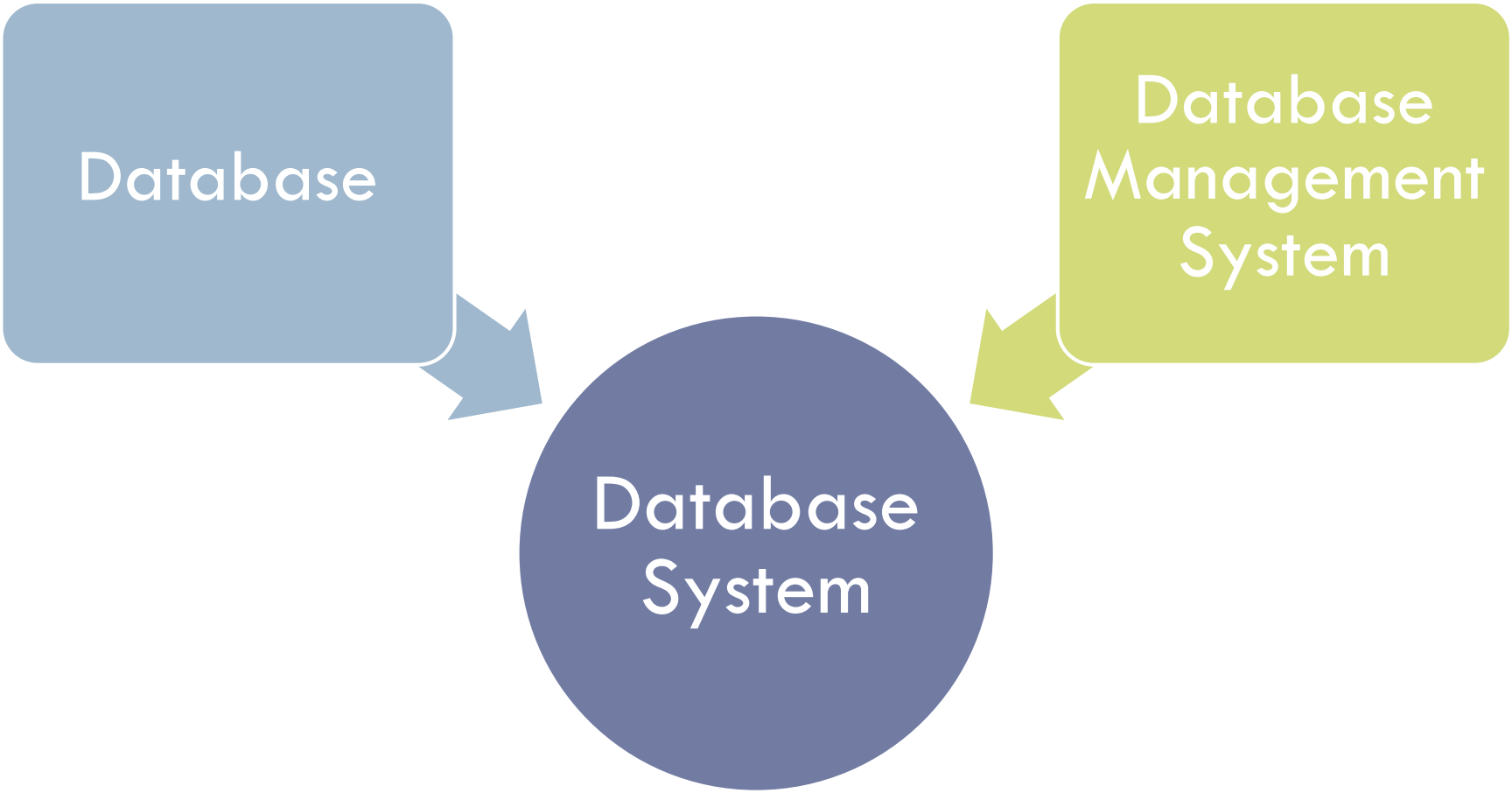
  โครงสร้างข้อมูลจะขึ้นอยู่กับภาษาคอมพิวเตอร์ที่ใช้ในการเขียนโปรแกรมประยุกต์ ถ้าแต่ละฝ่ายใช้ภาษาในการเขียนต่าง ๆ กัน ก็อาจทำให้โครงสร้างข้อมูลของแฟ้มไม่ตรงกัน ทำให้ไม่สามารถนำไฟล์ข้อมูลมาใช้ร่วมกันได้

Limitation of File-Based System (cont.)

- โปรแกรมที่ใช้งานคงที่ไม่ยืดหยุ่น

  ระบบแฟ้มข้อมูล มีความขึ้นกับโปรแกรมประยุกต์ ข้อมูลหรือรายงานต่าง ๆ จะถูกกำหนดรูปแบบตายตัวในโปรแกรมแล้ว ดังนั้นหากต้องการรายงานใหม่ จะต้องให้โปรแกรมเมอร์เขียนโปรแกรมขึ้นมา ทำให้เสียค่าใช้จ่ายเพิ่มขึ้น

Database Systems : Definition



Database Systems : Definition

A Database is

- โครงสร้างของการจัดเก็บข้อมูลที่มีความสัมพันธ์เกี่ยวข้องกันไว้ในที่เดียวกัน เพื่อให้สามารถนำข้อมูลมาประมวลเพื่อช่วยในการตัดสินใจ และสามารถใช้ข้อมูลร่วมกันได้

A Database Management Systems (DBMS) is

- กลุ่มของโปรแกรมที่จัดการข้อมูล

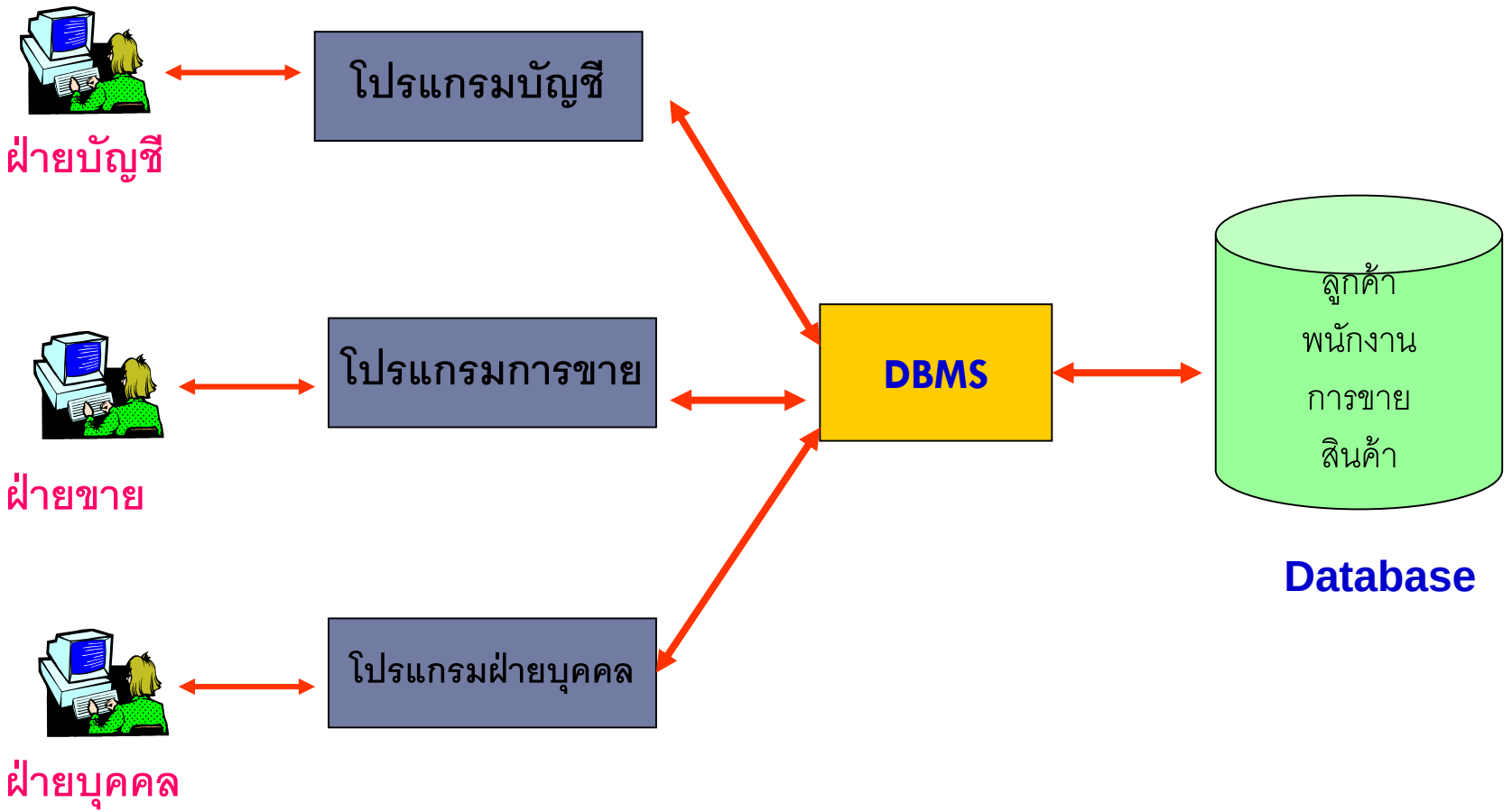
Database Systems

- ฐานข้อมูลมีส่วนที่ทำหน้าที่ในการอธิบายความหมายของรายการข้อมูลที่เก็บอยู่ในฐานข้อมูลด้วย เรียกส่วนนี้ว่า
  - บัญชีระบบ (System catalog)
  - เมตาดาต้า (Meta-data)
    - “Data about data”
  - พจนานุกรมของข้อมูล (Data Dictionary)

Database Systems (cont.)

- โครงสร้างของข้อมูลจะถูกแยกออกจากโปรแกรมประยุกต์และเก็บเอาไว้ในส่วนที่เรียกว่า “ฐานข้อมูล”
- ถ้ามีการเพิ่มหรือปรับปรุงโครงสร้างของข้อมูลก็จะไม่มีผลกระทบกับโปรแกรมประยุกต์

Database Systems (cont.)



Database Management System : DBMS

- ระบบจัดการฐานข้อมูล
- หมายถึง ซอฟต์แวร์ที่ใช้ในการจัดการข้อมูลในฐานข้อมูล
- DBMS จะทำหน้าที่เป็นตัวกลางระหว่างฐานข้อมูลกับโปรแกรมที่มาใช้งานฐานข้อมูลและผู้ใช้งานฐานข้อมูล ที่ติดต่อไปยังฐานข้อมูลเพื่อทำงานที่ผู้ใช้ต้องการให้สำเร็จ
- เช่น การจัดเก็บข้อมูลลงในฐานข้อมูล , การค้นหาข้อมูลที่ต้องการออกมาแสดง หรือ การลบข้อมูล เป็นต้น

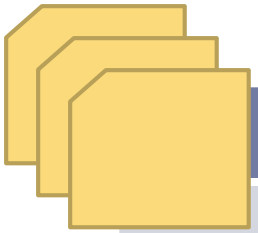
Goal of DBMS

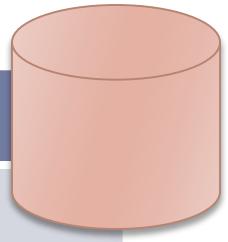
- เพื่อทำให้การจัดเก็บและนำสารสนเทศจากฐานข้อมูลมาใช้งานได้อย่างสะดวกและมีประสิทธิภาพ
- DBMS จะต้องมีความสามารถดังนี้
  - **Data Persistency**
    - ข้อมูลจะต้องมีความคงทน ไม่สูญหาย
  - **System Reliability**
    - ระบบต้องมีความน่าเชื่อถือ
  - **Scalability**
    - ระบบจะต้องมีความสามารถในการจัดการข้อมูลที่มีจำนวนมหาศาล และรองรับผู้ใช้งานจำนวนมาก

Management of Data

- DBMS จะต้องรับผิดชอบในส่วนการจัดการข้อมูล ซึ่งจะรวมถึงสิ่งเหล่านี้ด้วย
  - **DDL :** กำหนดโครงสร้างเพื่อจัดเก็บสารสนเทศ
  - **DML :** มีกลไกเพื่อจัดการสารสนเทศ (แสดง, เพิ่ม, ลบ, ปรับปรุง)
  - **System Reliability :** ต้องสร้างความมั่นใจในด้านความปลอดภัยของข้อมูลที่จัดการ แม้จะเหตุการณ์ระบบล่ม (System crashes)
  - **System Security :** หรือ ความพยายามที่จะเข้าสู่ระบบโดยไม่ได้รับอนุญาต (Unauthorized access)
  - **Concurrency Control :** ป้องกัน ความผิดปกติที่อาจเกิดขึ้นได้ เมื่อมีการแบ่ง หรือมีผู้ที่สามารถเข้าถึงข้อมูลได้พร้อม ๆ กันหลายคน

File Systems vs. Database Systems





File Systems	Database Systems
ข้อมูลซ้ำซ้อน	ข้อมูลสอดคล้อง
เข้าถึงข้อมูลยาก	เข้าถึงข้อมูลง่าย
Integrity problem	Integrity constraint
Atomic problem	Ensure atomicity
Concurrent access anomalies	Concurrency control
ปัญหาด้านความปลอดภัย	กลไกการเข้าถึงข้อมูล
พัฒนานาน	พัฒนาเร็ว
ผูกติดกับตัวโปรแกรมที่ใช้พัฒนา	Data independence

Table 1.1 เปรียบเทียบ File system กับ Database system

การประยุกต์ใช้ระบบงานฐานข้อมูล

- การซื้อของจากซูเปอร์มาเก็ต
- การซื้อของโดยใช้บัตรเครดิต
- การจองตั๋วเครื่องบินผ่านตัวแทนจำหน่าย
- การใช้บริการห้องสมุด
- การใช้งานอินเทอร์เน็ต
- การเรียนในมหาวิทยาลัย
- การบริหารในองค์กร
- ฯลฯ อีกมากมาย

Advantages of DB Systems

- มีความเป็นอิสระต่อกันระหว่างโปรแกรมและข้อมูล
- ลดความซ้ำซ้อนของข้อมูล
- เพิ่มความตรงกันของข้อมูล
- สามารถใช้ข้อมูลร่วมกันได้
- บังคับให้เป็นมาตรฐานเดียวกันได้
- ป้องกันและควบคุมการเข้าถึงข้อมูลได้ง่ายขึ้น
- ลดปัญหาในการบำรุงรักษาโปรแกรม

Disadvantages of DB Systems

- ซับซ้อน(Complexity)
- ขนาดใหญ่(Size)
- ราคาของ DBMS ค่อนข้างมีราคาสูง (Cost of DBMS)
- ราคาของฮาร์ดแวร์แพง (Additional hardware cost)
- ค่าใช้จ่ายในการแปลงระบบ (Cost of conversion)
- ผลกระทบจากความเสียหายสูง (Higher impact of a failure)

Data Abstraction

- จุดประสงค์หลักของ **database system** คือ ให้ผู้สามารถเห็นมุมมองของข้อมูลที่ถูกจัดเก็บได้
- **database system** ทำหน้าที่ซ่อนรายละเอียดการจัดเก็บและการบำรุงรักษาข้อมูล โดยไม่แสดงให้ผู้ใช้เห็น
- เพื่อให้ง่ายต่อการใช้งาน **user interface** ถูกจัดทำให้กับผู้ใช้
- เพื่อประสิทธิภาพ โครงสร้างข้อมูลที่ซับซ้อนจะถูกแทนที่ด้วยข้อมูลภายใน **Database**

View of Data

- ข้อมูลนามธรรม **(data abstraction)** สามารถแสดงได้ในหลายระดับ **(Multi levels)**
- ตัวอย่าง ข้อมูลส่วนตัวบุคคลสามารถมีมุมมอง เช่น
  - **High-level:** ข้อมูลยา, ข้อมูลประวัติการศึกษา
  - **Intermediate-level:** ชื่อ (**alphabetic**), อายุ(**numeric**), เพศ(**enumeration** : Mail/Female)
  - Low-level: sequences of bits, bytes, blocks
- ใน ค.ศ. 1978 **"ANSI/SPARC" architecture** กำหนด ข้อมูลรูปแบบได้ 3 ระดับ
  - External Level (View)
  - Conceptual Level (Logical)
  - Internal Level (Physical)

ANSI/SPARC Architecture

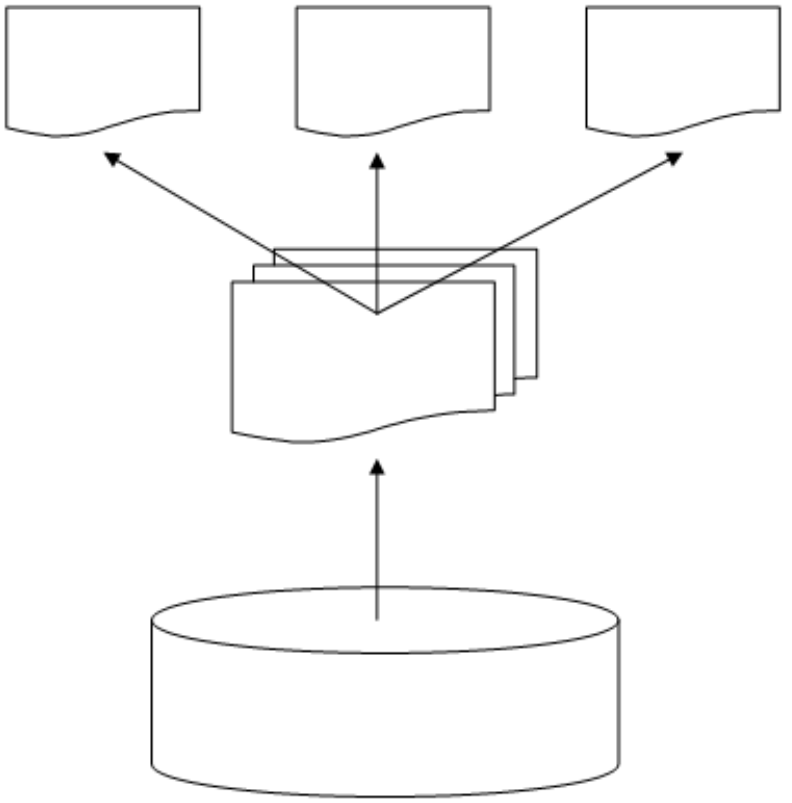
- ใน ANSI/SPARC Architecture , a Database ประกอบด้วย
  - View schema
    - อธิบายมุมมองของ Database (External Level)
  - Logical schema
    - อธิบายการออกแบบที่ระดับตรรกะ (Logical level) (Conceptual Level)
  - Physical schema
    - อธิบายการออกแบบที่ระดับกายภาพ (Physical Level) (Internal Level)

Schema

- A Schema of a database
  - โครงสร้างโดยรวมของข้อมูลในฐานข้อมูลและความสัมพันธ์ของข้อมูลเหล่านั้น
  - Schema นาน ๆ ครั้งจึงมีการเปลี่ยนแปลง ในขณะที่ Data จะมีการเปลี่ยนแปลงอย่างต่อเนื่อง

Levels of Data Abstraction

- View Level
  - ส่วนติดต่อกับผู้ใช้
  - แสดงมุมมองที่แตกต่างกันของข้อมูล
- Logical Level
  - ข้อมูลอะไรที่ถูกเก็บในฐานข้อมูล และ..
  - ความสัมพันธ์ระหว่างข้อมูลเหล่านั้น
- Physical Level
  - ข้อมูลถูกเก็บไว้อย่างไร



View level : ระดับมุมมอง

- ติดต่อกับผู้ใช้
- ผู้ใช้แต่ละคนอาจจะมีมุมมองข้อมูลแตกต่างกันหรือเหมือนกันก็ได้
- รูปแบบข้อมูลที่เห็นในระดับภายนอก เรียกว่า เค้าร่างภายนอก (External schema)หรือ วิว(View) ซึ่งอาจนำเสนอได้หลายรูปแบบ
- แต่ละฐานข้อมูลสามารถมี View ได้หลายรูปแบบ

View level

User A

ชื่อ	โทรศัพท์
กนก	0-4221-1040
ขนิษ	0-4224-4505

User B

รหัส	ชื่อ	ที่อยู่	โทรศัพท์
001	กนก	48/7 ถ.อุดรดุษฎี	0-4221-1040

User C

รหัสวิชา	ชื่อวิชา	หน่วยกิต
DB01	ระบบฐานข้อมูล	3
PR01	หลักการเขียนโปรแกรม	3
NW01	เครือข่ายและโทรคมนาคม	3

View schema

Logical schema

รหัส	ชื่อ	ที่อยู่	โทรศัพท์
001	กนก	48/7 ถ.อุดรดุษฎี	0-4221-1040
002	ขนิษ	64 ถ.ทหาร	0-4224-4505

รหัสวิชา	ชื่อวิชา	หน่วยกิต
DB01	ระบบฐานข้อมูล	3
PR01	หลักการเขียนโปรแกรม	3
NW01	เครือข่ายและโทรคมนาคม	3

Logical Level : ระดับแนวคิด

- เป็นโครงสร้างหลักของฐานข้อมูลโดยรวม
- โครงสร้างข้อมูลในระดับนี้มุ่งเน้นความสัมพันธ์(Relationship) ระหว่างข้อมูลเป็นหลักสำคัญ หรือเรียกว่าแบบจำลองข้อมูล(Data Model)
- เป็นระดับที่อธิบายถึงว่า ข้อมูลอะไร(What) ที่จะจัดเก็บลงในฐานข้อมูล และมีความสัมพันธ์ระหว่างอย่างไร
- ระดับแนวคิดมีความเกี่ยวข้องกับสิ่งต่อไปนี้
  - จำนวนเอนติตี้ทั้งหมด ซึ่งประกอบด้วย แอตทริบิวต์ และความสัมพันธ์ระหว่างเอนติตี้
  - กฎเกณฑ์ของข้อมูล
  - ความปลอดภัย และความคงสภาพของข้อมูล

Logical Level

- ข้อมูลในระดับแนวคิดจะถูกแสดงตามแบบจำลองข้อมูล ที่ฐานข้อมูลนั้นใช้ เรียกว่า เค้าร่างแนวคิด (Conceptual schema)
- ผู้ที่ทำหน้าที่บริหารจัดการโครงสร้างในระดับนี้คือ ผู้บริหารฐานข้อมูล (DBA)

Logical Level

001	กนก	48/7 ถ.อุดรดุษฎี	0-4221-1040
002	ขนิษ	64 ถ.ทหาร	0-4224-4505
003	คนึง	55/2 ถ.ศรีชมชื่น	0-4225-5142

Logical schema

Physical schema

Physical Level : ระดับกายภาพ

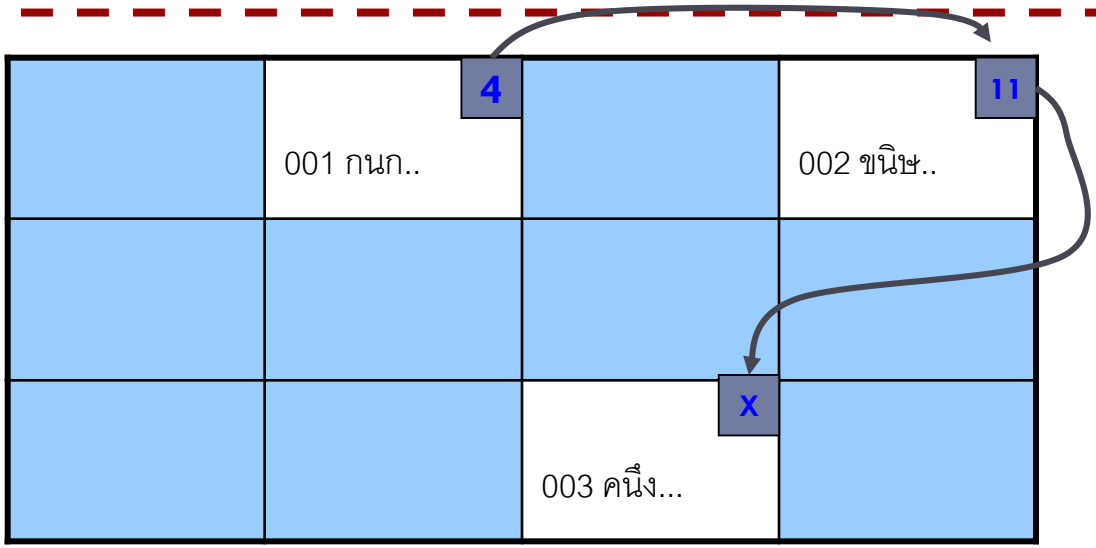
- เป็นระดับที่จัดเก็บข้อมูลด้วยโครงสร้างที่เหมาะสม ซึ่งมีผลต่อความเร็วและประสิทธิภาพในการเข้าถึงข้อมูลที่ต้องการ
- โครงสร้างข้อมูลที่ใช้เก็บ เช่น Tree , B-Tree หรือ Index ขึ้นอยู่กับการกำหนดโดย DBA
- รูปแบบข้อมูลที่เห็นในระดับภายในเรียกว่า เค้าร่างภายใน(Internal schema)
- เป็นระดับที่มีการทำงานประสานกับระบบปฏิบัติการ(OS)
- ข้อมูลในระดับภายในยังไม่ใช่รูปแบบการจัดเก็บข้อมูลจริงๆที่เก็บในดิสก์
- การอ่านและเขียนข้อมูลเป็นหน้าที่ของระบบปฏิบัติการ(OS)

Physical Level

Physical schema

Physical Disk

1	2	3	4
5	6	7	8
9	10	11	12



Data Independence

- Data Independence
  - การเปลี่ยนแปลง Low-level schema จะไม่มีผลกระทบต่อ schema ระดับที่สูงขึ้นไป รวมทั้งแอพลิเคชั่นต่าง ๆ ด้วย
- Physical Data Independence
  - ถ้ามีการเปลี่ยนแปลง physical schema จะไม่ส่งผลกระทบต่อโปรแกรม
  - นั่นคือ โปรแกรมประยุกต์จะไม่ขึ้นอยู่กับ physical schema (แต่จะขึ้นอยู่กับ logical schema หรือ view schema)
- Logical Data
  - ถ้ามีการเปลี่ยนแปลง logical schema จะไม่ส่งผลกระทบต่อโปรแกรม
  - นั่นคือ โปรแกรมประยุกต์จะไม่ขึ้นอยู่กับ logical schema (แต่จะขึ้นอยู่กับ view schema บางตัวเท่านั้น)

Why do we need 3 levels?

- ผู้ใช้แต่ละคนสามารถเข้าถึงข้อมูลเดียวกันได้ แต่อาจจะมีมุมมองในการใช้งานต่างกันได้
- ผู้ใช้จะไม่สามารถเข้าถึงข้อมูลในระดับกายภาพได้โดยตรง
- ผู้บริหารฐานข้อมูลสามารถแก้ไขโครงสร้างในการจัดเก็บฐานข้อมูล โดยไม่ส่งผลกระทบต่อมุมมองของผู้ใช้
- โครงสร้างของระดับภายในของฐานข้อมูลจะไม่ได้รับผลกระทบจากการเปลี่ยนตำแหน่งในการจัดเก็บในระดับกายภาพ
- ผู้บริการฐานข้อมูล(DBA) สามารถที่จะเปลี่ยนโครงสร้างระดับแนวคิดของฐานข้อมูลโดยไม่ส่งผลกระทบกับผู้ใช้ทุกคน

DB Models and History

- 1961, Charles Bachman (1973 Turing Award) designed GE's
- Integrated Data Store: IDS – the predecessor of Network Model.
- Late 1960s, IBM built "Information Management System: IMS"
- Hierarchical Model.
- Late 1960s, CODASYL group defined/standardized Network Model.
- 1969, Relational Model by Edgar Codd [IBM] (1981 Turing Award).
- 1970s, SEQUEL (SQL), QBE; QUEL, System R (DB2), Ingres
- 1976, Entity-Relationship Model by Peter Chen
- 1980s, DB2, Oracle, Sybase, Informix, DBase, Paradox, …

DB Models and History (cont.)

- 1986, SQL Standard
- 1980s, distributed databases
- 1998, James Gray received Turing Award in Transaction Management.
- 1990s-2000s,OODB, ORDB (Postgres); Data Warehouse, OLAP,
- Data Mining, GIS, Mobile DB, Multimedia DB, Web DB, XML DB, …

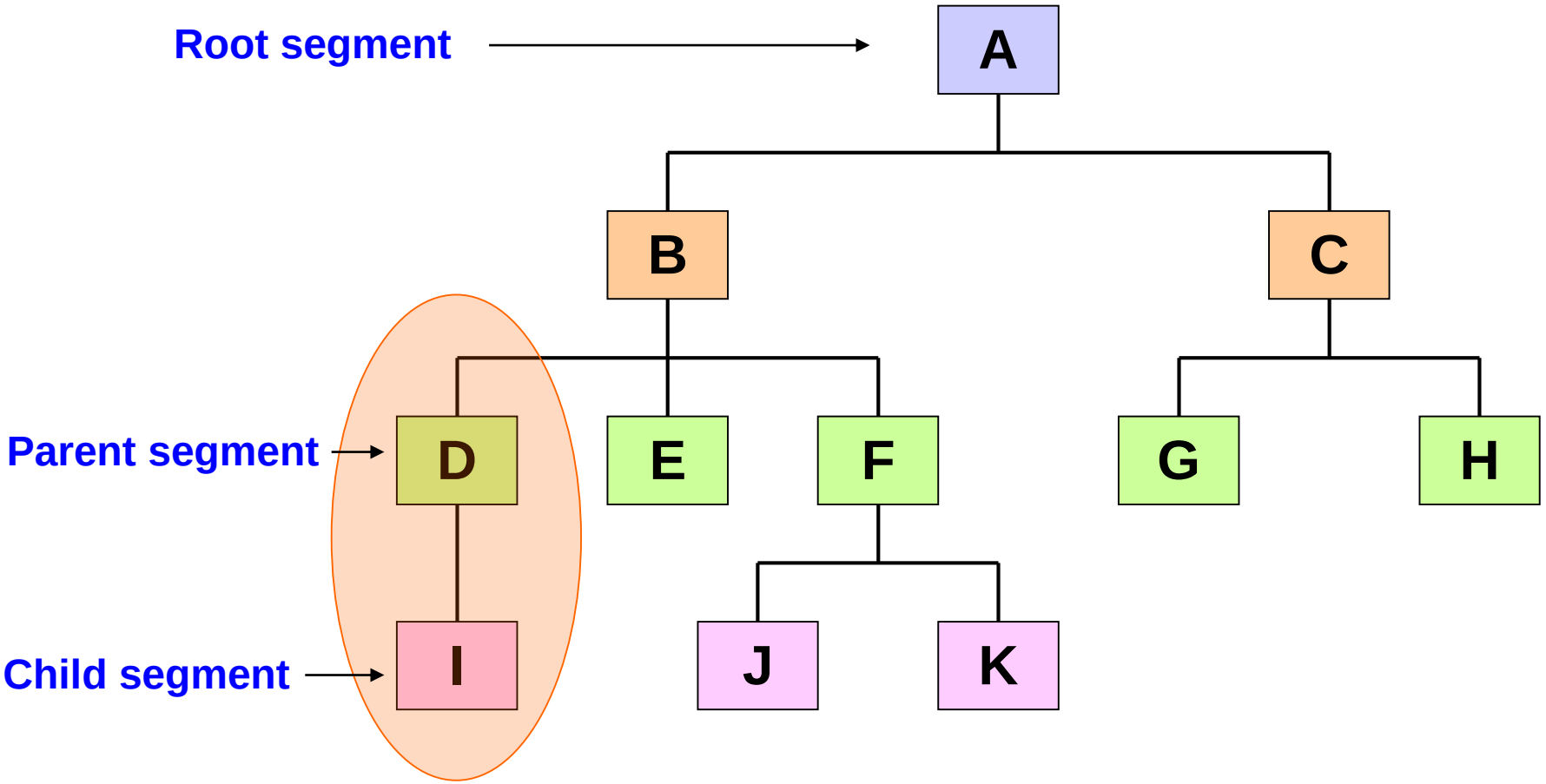
Data Model

- แบบจำลองข้อมูล(Data Model) หมายถึง แบบจำลองที่ใช้อธิบายและจัดการข้อมูล , ความสัมพันธ์ระหว่างข้อมูล และข้อบังคับของข้อมูลในระบบ

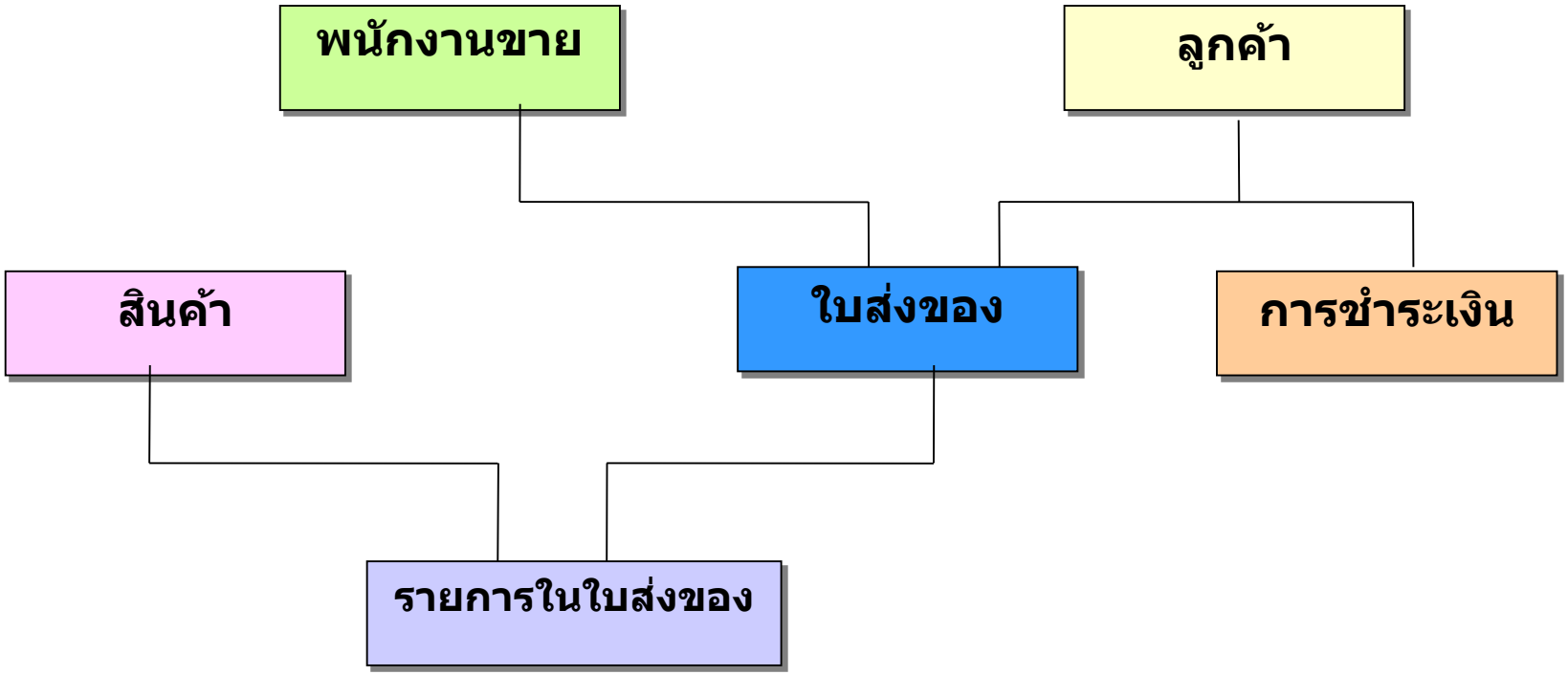
Database Models

- Hierarchical Model
  - the first DB model
  - e.g., IMS, System 2000
- Network Model
  - DBTG standard; after IDS
  - e.g., IDMS, Total, …
- Relational Model
  - the most widely used now
  - e.g., System R, Ingres, ..
- Entity-Relationship Model
  - a conceptual model
- Object-Oriented Model
  - mostly for special purposes
  - e.g., ObjectStore, Gemstone, Ontos, O2, Jasmine, Cache,
- Object-Relational Model
  - extends the relational model
  - e.g., Postgres, Informix, DB2, Oracle, …
- *Semi-structured Data*
  - Focus on data in XML format

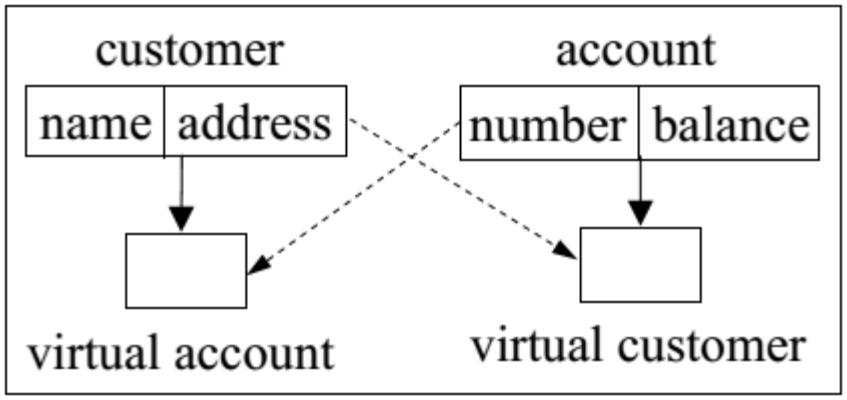
แบบจำลองลำดับชั้น(Hierarchical data model)



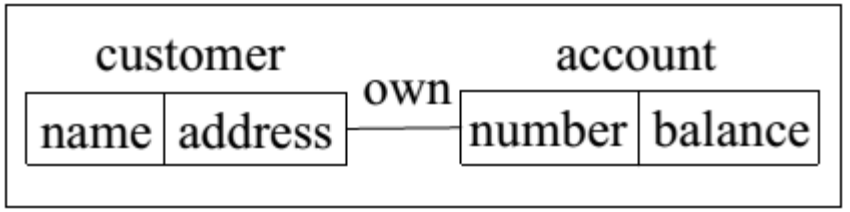
แบบจำลองเครือข่าย(Network data model)



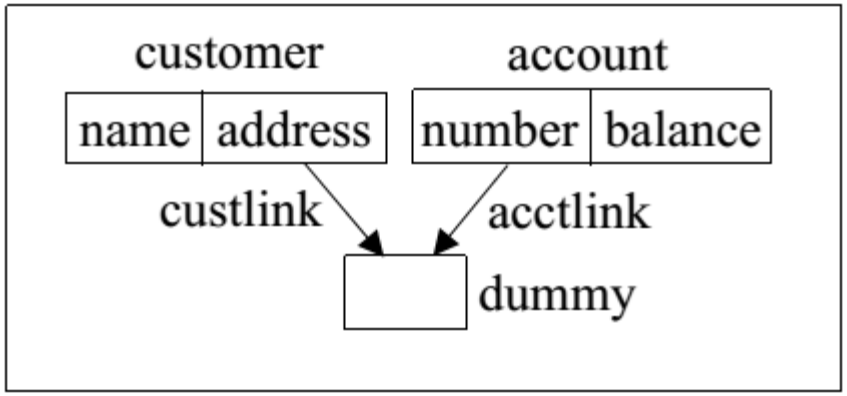
Data Models – Diagrams Hierarchical and Network



Hierarchical tree-structure diagram

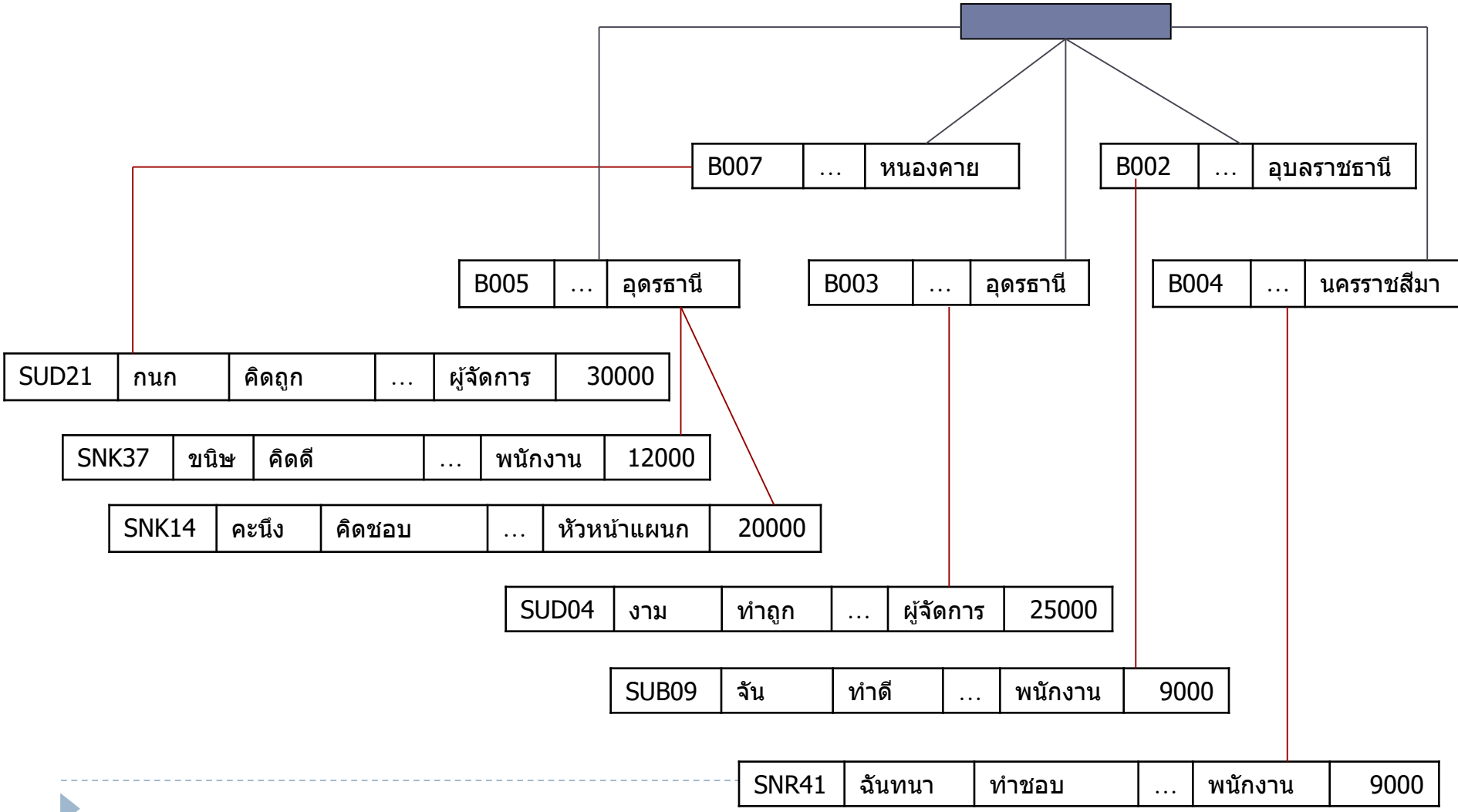


Network data-structure diagram



DBTG data-structure diagram

แบบจำลองลำดับชั้น(Hierarchical data model)



แบบจำลองข้อมูลเชิงสัมพันธ์(Relational data model)

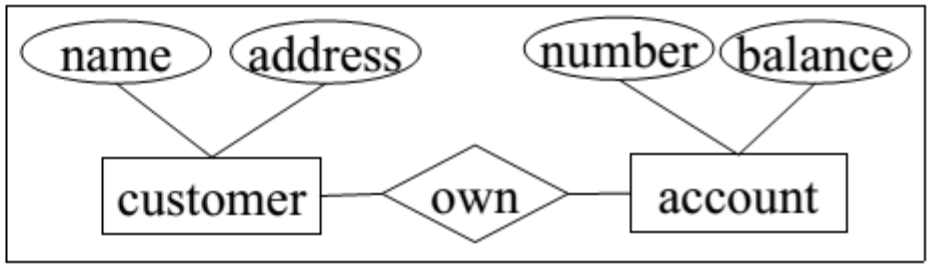
รหัสสาขา	ที่อยู่	จังหวัด	รหัสไปรษณีย์
B003	44/3 ถ.อุดรดุษฎี	อุดรธานี	41000
B005	55/5 ถ.นิตโย	อุดรธานี	41000
B007	16 ถ.โพนพิสัย	หนองคาย	44000
B002	30 ถ.ชยางกูร	อุบลราชธานี	34000
B004	88/10 ถ.ราชสีมา	นครราชสีมา	43000

รหัสพนักงาน	ชื่อ	นามสกุล	ตำแหน่ง	เพศ	วันเกิด	เงินเดือน	รหัสสาขา
SUD21	กนก	คิดถูก	ผู้จัดการ	ช	1 ตค. 2516	30000	B005
SNK37	ขนิษ	คิดดี	พนักงาน	ญ	10 พย. 2519	12000	B007
SNK14	คะนึง	คิดชอบ	หัวหน้าแผนก	ช	24 มีค. 2517	20000	B007
SUB09	งาม	ทำถูก	พนักงาน	ญ	19 กพ. 2521	9000	B002
SUD04	จัน	ทำดี	ผู้จัดการ	ญ	3 กค. 2518	25000	B003
SNR41	ฉันทนา	ทำชอบ	พนักงาน	ญ	13 มิย. 2520	9000	B004

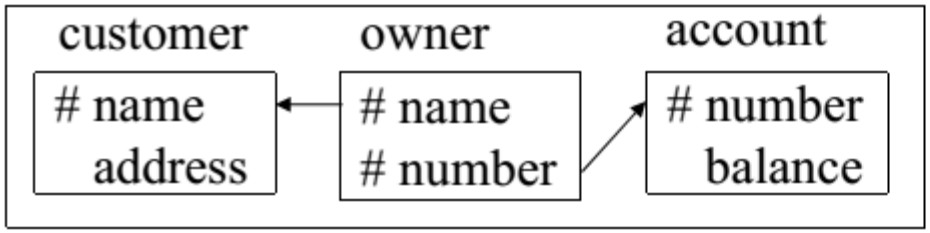
Object data model : แบบจำลองเชิงวัตถุ

- แบบจำลองข้อมูลเชิงวัตถุใช้หลักการเกี่ยวกับ เอนติตี้(Entity) , แอททริบิวท์(Attribute) และความสัมพันธ์(Relationship)
- ตัวอย่างของแบบจำลองนี้ได้แก่ Entity-Relationship , Semantic , Functional , Object-Oriented

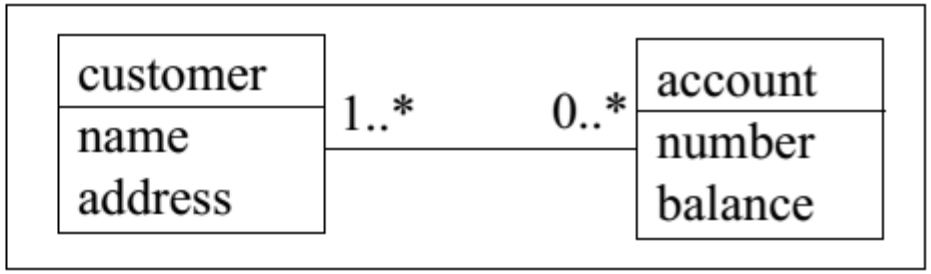
Data Models – Diagrams Relational, ER, OO



Entity-Relationship diagram



Relational schema diagram



UML class diagram (OO Model)

Database Languages

- Data Definition Language (DDL)
  - ใช้เพื่อระบุเจาะจง Database schema
  - Database schemas ถูกเก็บไว้ใน Data dictionary
  - The data dictionary บรรจุ metadata
  - Metadata: data ที่อธิบาย data, e.g. schema

- Data Manipulation Language (DML)
  - เพื่อแสดงผลข้อมูลในฐานข้อมูล (query) และปรับปรุง (เพิ่ม, ลบ, แก้ไข)
  - Procedural DMLs -and- Declarative DMLs

DB Languages : Example

Hierarchical Model (DL/I):

DML:

```
get first customer
where customer.name:="Henry";
```

Network Model (DBTG):

DDL:

```
set name is custlink
   owner is customer
   member is dummy;
```

DML:

```
customer.name := "Newton";
find any customer using name;
get customer;
print(customer.address);
```

Relational Model (SQL):

DDL:

```
create table customer (
   name        char(30),
   address     char(60));
```

DML:

```
select address
from customer
where name likes "Robert";
```

Object-Oriented Model (ODMG):

DDL:

```
class customer:public object {
   string name;
   string address;
   set<ref<account>> accounts;
};
```

Data Manipulation Languages

- Procedural DMLs
  - ผู้ใช้ต้องระบุข้อมูลที่ต้องการ และต้องบอกวิธีการให้ได้มาซึ่งข้อมูลเหล่านั้น
  - e.g., relational algebra; (C, Pascal, Java, etc.)
- Declarative DMLs (nonprocedural DMLs)
  - ผู้ใช้ระบุข้อมูลที่ต้องการเท่านั้น โดยไม่ต้องระบุวิธีการ
  - e.g. relational tuple calculus, relational domain calculus, SQL (core), Quel, QBE, Datalog; (Prolog)
- A Query Language
  - ส่วนหนึ่งใน DML ซึ่งใช้ในการเรียกดูข้อมูลเท่านั้น (ไม่มีการปรับปรุงข้อมูล)

What are the components of DBMSs?

- The functional components of a database system
  - Query Processor Component: เพื่อลดความซับซ้อนและอำนวยความสะดวกในการเข้าถึงข้อมูล (convenient and efficient)
  - Storage Manager Component: ทำหน้าที่จัดการย้ายข้อมูลระหว่างดิสก์และหน่วยความจำให้มีประสิทธิภาพสูงที่สุด (ใช้เวลาน้อยที่สุด)
  - Transaction Manager Component: จัดการความเป็นอันหนึ่งอันเดียวกันของข้อมูล (atomicity) และความสอดคล้อง (concurrency) ของแทรนเซคชัน (transactions) และสม่ำเสมอและคงทนของฐานข้อมูล

Query Processor Component

- **Query Processor:**
  - โปรแกรมโมดูล (program module) ที่เป็นส่วนเชื่อมต่อระหว่างฐานข้อมูลและโปรแกรมประยุกต์
- **Components include:**
  - DDL interpreter
    - ตีความคำสั่ง (interpret) DDL และบันทึกลงใน data dictionary
  - DML compiler
    - แปลคำสั่ง (translate) DML เป็น query evaluation plans
  - Query evaluation engine
    - ประมวลผลคิวรี (queries) ตามแผนที่ได้จาก query evaluation plans

Storage Manager Component

- A Storage Manager is
  - โปรแกรมโมดูล ที่เป็นส่วนเชื่อมต่อระหว่างข้อมูลที่ถูกเก็บไว้ในฐานข้อมูลและ โปรแกรมประยุกต์หรือคิวรีที่ถูกส่งเข้าสู่ระบบ
- Components include:
  - Authorization and integrity manager
  - File manager
  - Buffer manager
- Structures maintained:
  - data files, data dictionary, indices

Transaction Manager Component

- **Transaction:**
  - กลุ่มของการดำเนินงานที่ทำหน้าที่เป็นฟังก์ชั่นเชิงตรรกะเดียวในการประยุกต์ฐานข้อมูล
  - แทรนเซคชั่นต้องมีคุณสมบัติ ACID ชุดของคุณสมบัติต่าง ๆ ที่จะรับประกันว่าการเปลี่ยนแปลงรายการของฐานข้อมูลได้ผ่านการประมวลผลมาอย่างเชื่อถือได้ ในทางฐานข้อมูล
  - การดำเนินการทางตรรกะกับข้อมูลหนึ่งครั้งจะเรียกว่า 1 **transaction** เช่น การโอนเงินจากบัญชีหนึ่งของธนาคารไปยังอีกบัญชีหนึ่ง ถือเป็น 1 **transaction** แม้ว่าจะมีกระบวนการย่อยหลายอย่างใน **transaction** นั้น ๆ ก็ตาม ความหมายของแต่ละคำจาก ACID คือ

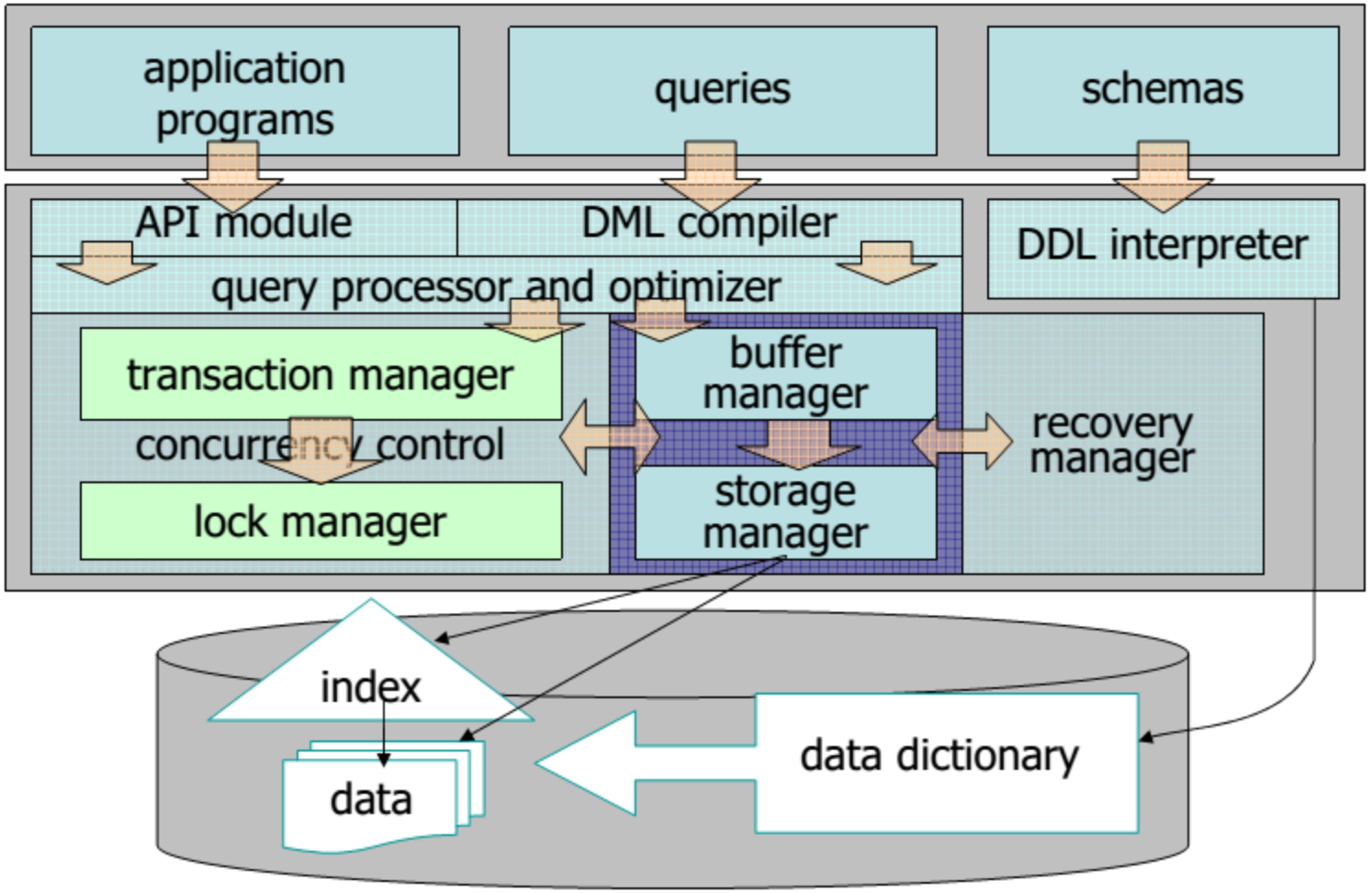
Transaction Manager Component (cont.)

- Transaction Properties: ACID
  - A tomicity: ผลลัพธ์ในการทำรายการเปลี่ยนแปลงหรือบริการใดๆ จะมีความเป็นอันหนึ่งอันเดียวกัน คือ ถูกกระทำและยืนยันผลลัพธ์ทั้งหมด หรือยกเลิกทั้งหมด
  - C onsistency:คุณสมบัติที่ทำให้แน่ใจว่าฐานข้อมูลจะยังคงมีสภาพที่สอดคล้องกันทั้งก่อนหน้าและหลังการทำแทรนเซคชั่น (ไม่ว่าแทรนเซคชันนั้น ๆ จะสำเร็จหรือไม่สำเร็จก็ตาม)
  - I solation: จะไม่มีทรานแซคชั่นใด ๆ มาแทรกกลางระหว่างที่ทรานแซคชั่นหนึ่งกระทำการอยู่
  - D urability: การรับประกันว่าข้อมูลที่เป็นผลจากการกระทำของ **transaction** หนึ่ง ๆ จะยังคงอยู่หลังจากที่ **transaction** นั้น ๆ ทำงานสำเร็จแล้วและไม่มีการยกเลิก แม้ว่าระบบจะเกิดล้มเหลวขึ้นก็ตาม

Transaction Manager Component (cont.)

- Components include
  - Transaction Manager
  - Lock Manager
  - Recovery Manage

DB System Structure



Who interact with DBMSs ?

- Database Administrators (DBAs)
- Database Operators
- Database Designers
- DB Application Developers/Programmers
- End Users
  - Sophisticated End Users
  - Naive End Users
- DB System Designers and Implementers
- Database Tool Developers

DB Administrators (DBAs) and DB Operators

- DBAs are responsible for
  - เปิด/ปิด ฐานข้อมูล
  - บรรจุ, นำเข้า, ส่งออกข้อมูล เข้าสู่/ออกจากฐานข้อมูล
  - ให้สิทธิ์การเข้าใช้ฐานข้อมูลและทรัพยากร
  - ตรวจสอบการใช้ทรัพยากร
  - สำรองฐานข้อมูลตามระยะเวลาและเรียกคืนเมื่อจำเป็น
  - ปรับแต่งประสิทธิภาพ
  - พิจารณาการอัพเกรดซอฟต์แวร์และฮาร์ดแวร์
- DB Operators are responsible for
  - ช่วยเหลือ DBAs ในการบำรุงรักษาการดำเนินงานฐานข้อมูลทั่วไป

DB Designers and DB Developers / Programmers

- Database Designers are responsible for
  - วิเคราะห์ความต้องการในเชิงข้อมูล
  - ออกแบบโครงสร้างฐานข้อมูล
    - View, logical, and physical structures
  - สร้างฐานข้อมูล
- DB Application Developers / Programmers
  - วิเคราะห์ความต้องการในเชิงการใช้งาน
  - พัฒนาโปรแกรม
  - ดำเนินการเกี่ยวกับโปรแกรม

End Users

- Sophisticated End Users
  - เขียนคิวรีเฉพาะกิจ (ad hoc) เพื่อหาคำตอบที่ต้องการ
  - สร้างรายงานแบบเฉพาะกิจเพื่อผู้บริหารระดับสูง
- Naive End Users
  - ใช้งานโปรแกรมประยุกต์ ที่ได้จาก โปรแกรมเมอร์
    - Data entry operators, …
    - Executives, managers, …

Very Important Participants

- ผู้นำในวงการอุตสาหกรรม
  - Oracle
    - Oracle
  - IBM
    - DB2
  - Microsoft
    - SQL Server, Access, FoxPro
    - ODBC, AD
  - Sybase
    - Sybase

Trends of Database Products

- **Past and Present :** รองรับข้อมูลจำนวนมากและผู้ใช้งานพร้อมกันหลายคน **(Multi-user)**
  - การบริหารข้อมูลให้มีโครงสร้างที่ดี
  - On-line transaction (OLTP)
  - Scalability

Trends of Database Products (cont.)

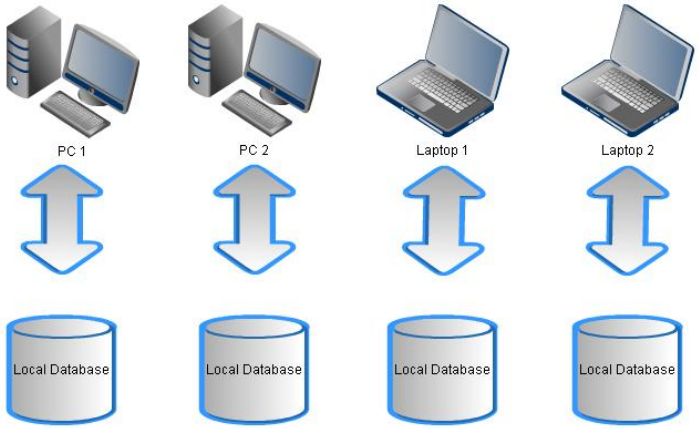
- Present and Future : A Complete System
  - การจัดการข้อมูลที่หลากหลายและซับซ้อน
  - Advanced OLTP: workflow
  - Application development tools
  - Data repository: warehouse, data mart
  - Data analysis tools:
    - Online Analytical Processing (OLAP)
    - Data Mining
  - a well-rounded system (รอบรู้)
    - Network-enable
    - XML-complaint, multi-model

DBMS Architectures

- แบ่งตามจำนวนผู้ใช้
  - Single-user Systems
    - ผู้ใช้งานเพียงคนเดียวใช้ระบบในช่วงเวลาหนึ่ง ๆ
  - Multi-user Systems
    - ผู้ใช้งานหลายคนใช้ระบบในช่วงเวลาเดียวกัน
- แบ่งตามสถานที่ตั้งของฐานข้อมูลและผู้ใช้
  - Stand-alone DBMSs
    - 1 DBMS รองรับผู้ใช้ในเครื่องเดียว
  - Client-Server (Centralized) DBMS
    - 1 ฐานข้อมูลรองรับผู้ใช้หลายลูกข่าย **(multiple clients)**
  - Distributed DBMSs
    - DBMS หลาย ๆ ระบบผสานงานกัน

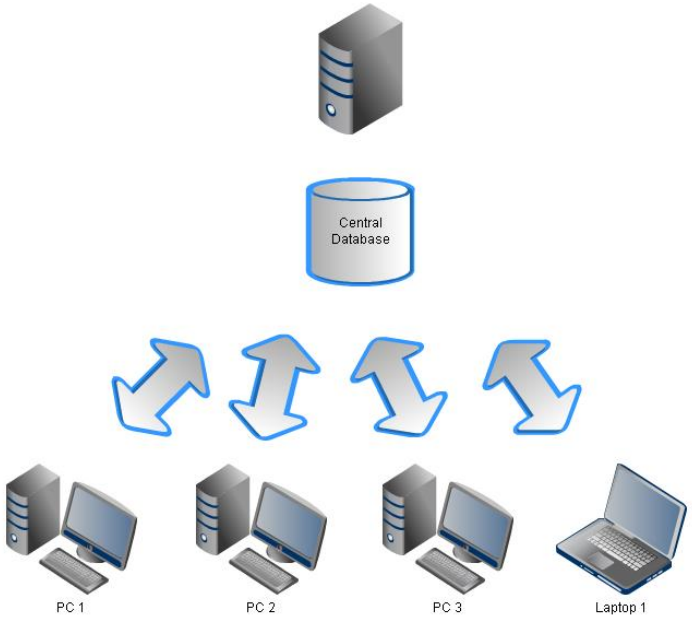
แบ่งตามจำนวนผู้ใช้

- ฐานข้อมูที่มีผู้ใช้คนเดียว (Single-User)
  - บางครั้งเรียกว่า Stand alone database หรือ Desktop database



แบ่งตามจำนวนผู้ใช้(ต่อ)

- ฐานข้อมูลที่มีผู้ใช้ครั้งละหลายคน (Multi-User)
  - ระบบฐานข้อมูลแบบนี้จะสนับสนุนการใช้งานของผู้ใช้หลายคนในเวลาเดียวกัน

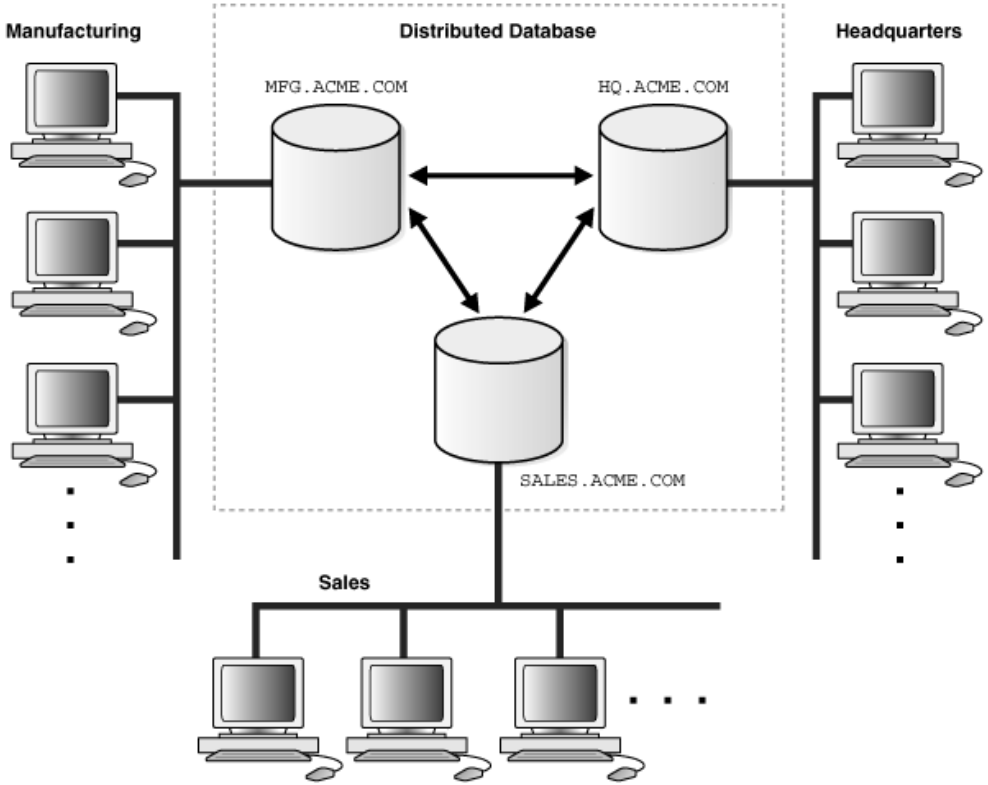


แบ่งตามสถานที่ตั้งของฐานข้อมูล

- ฐานข้อมูลแบบรวมศูนย์ (Centralized Database)

แบ่งตามสถานที่ตั้งของฐานข้อมูล(2)

- ฐานข้อมูลแบบกระจาย (Distributed Database)



DB Application Architectures

- Client-Server(C/S)
  - Client machine: the database users work
  - Server machine: the database system run
- C/S : Two-tier architecture : small systems
  - Client : ผู้ใช้และโปรแกรมประยุกต์ฐานข้อมูล (DB application)
  - Server : Database systems
- C/S : Three-tier architecture : large systems , web
  - Client : ส่งคำขอ/คำสั่งไปที่ Application server
  - Application Server : ดำเนินงานเป็น server สำหรับ client และดำเนินงานเหมือน client สำหรับ database server
  - Database Server : ดำเนินงานของ DBMS

End of Introduction