บทที่ 3

ระบบฐานข้อมูล

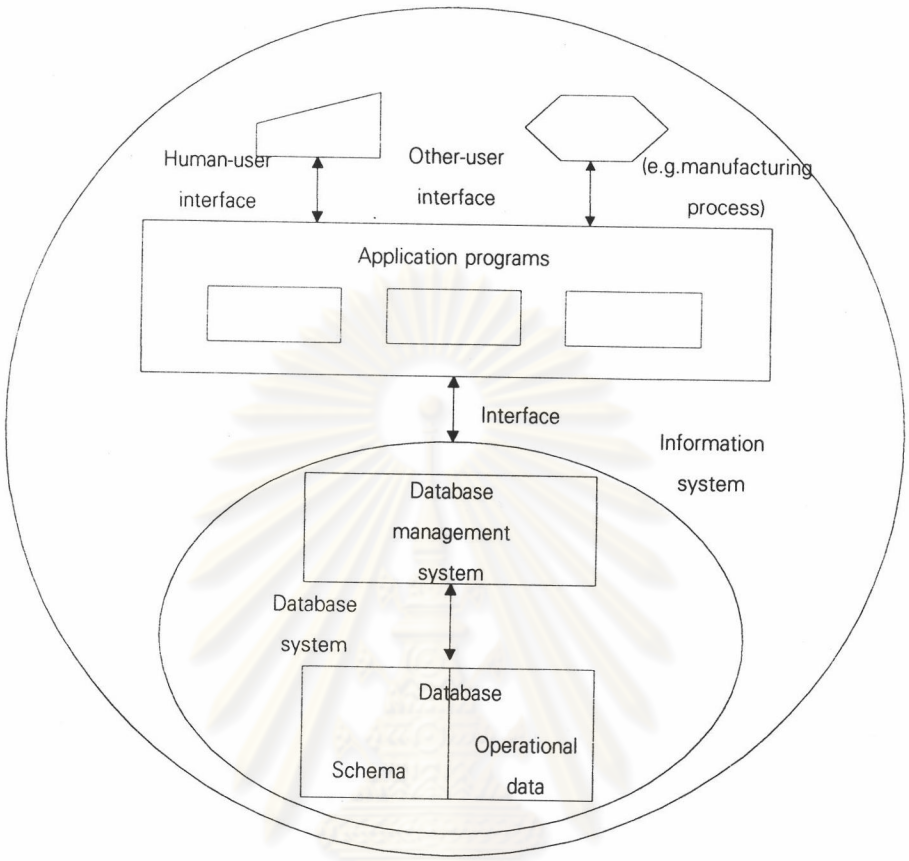
ระบบสารสนเทศ (อ้างถึงในมณฑา พิเชฐสกุล, 2538)

ระบบสารสนเทศ คือ ระบบที่เก็บรวบรวมข้อมูล ซึ่งข้อมูลนั้นอาจใช้เป็นข้อมูลพื้นฐานที่ใช้ในการคาดการณ์ การพยากรณ์ หรือเป็นข้อมูลที่ช่วยในการตัดสินใจ หรืออาจเป็นข้อมูลที่ต้องผ่านขั้นตอนต่าง ๆ เพื่อวิเคราะห์ข้อมูลนั้นในรูปแบบต่าง ๆ แล้วจึงนำผลที่ได้มาช่วยในการวางแผน ช่วยในการจัดการ ตลอดจนช่วยในการตัดสินใจ ในปัจจุบันเทคโนโลยีทางคอมพิวเตอร์ และการสื่อสารมีความเจริญก้าวหน้าอย่างมาก ระบบสารสนเทศโดยทั่วไปจึงหมายถึงระบบที่นำคอมพิวเตอร์มาช่วยประมวลผล การนำข้อมูลเก็บไว้ในระบบคอมพิวเตอร์ หรือการนำข้อมูลมาใช้งานจึงต้องใช้ชุดคำสั่งหรือโปรแกรมที่เป็นตัวเชื่อมระหว่างผู้ใช้ระบบและข้อมูลที่เก็บไว้ในคอมพิวเตอร์ ฉะนั้นระบบสารสนเทศจึงต้องมีส่วนประกอบที่สำคัญ 2 ส่วนดังรูปที่ 3.1 คือ

1. ระบบฐานข้อมูล (Database System)
2. โปรแกรมประยุกต์ (Application Program) ซึ่งทำหน้าที่ประมวลผลข้อมูลตามความต้องการของผู้ใช้ระบบ

ระบบฐานข้อมูล (Whittington, 1988) ประกอบด้วย ฐานข้อมูล (Database) และระบบจัดการฐานข้อมูล (Database Management System)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



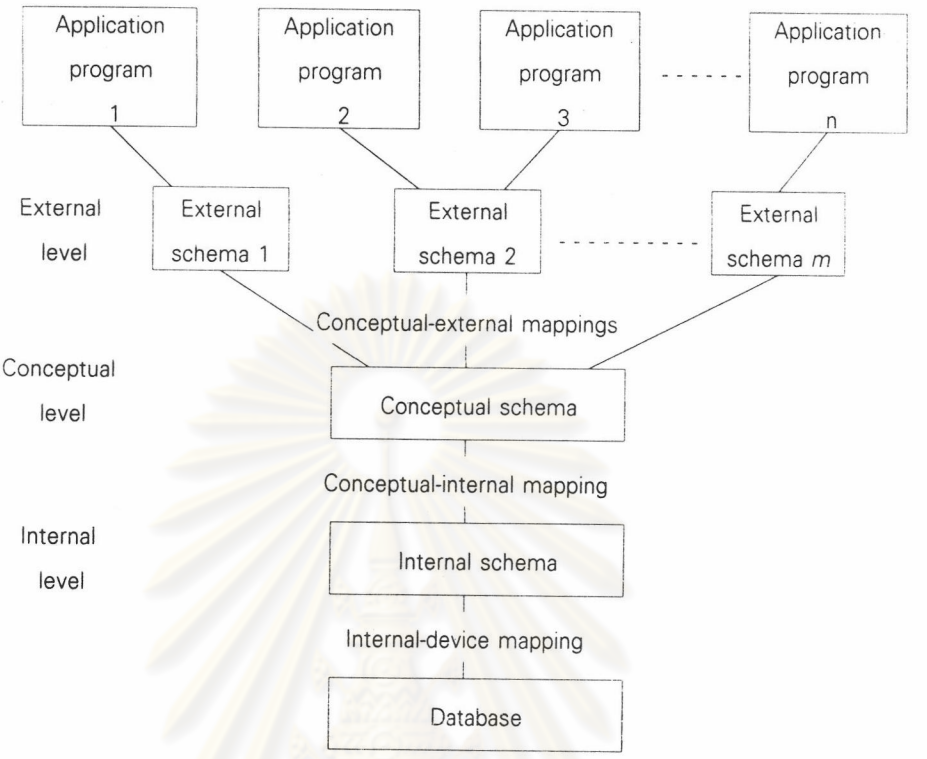
รูปที่ 3.1 ความสัมพันธ์ระหว่างระบบฐานข้อมูล (Database system) และระบบสารสนเทศ (Information system) (Whittington, 1988)

สถาปัตยกรรมฐานข้อมูล

The ANSI/SPARC Study Group on Database Management Systems (ANSI, 1975) ได้นำเสนอหลักการของฐานข้อมูล ซึ่งได้ใช้เป็นแนวทางในการกำหนดกรอบงาน (Framework) ของระบบจัดการฐานข้อมูล ในขณะเดียวกันหลักการดังกล่าวก็สามารถใช้เป็นข้อมูลในการเปรียบเทียบ และประเมินคุณสมบัติของระบบจัดการฐานข้อมูลทั่ว ๆ ไปของแต่ละบริษัท หรือแต่ละยี่ห้อว่ามีความแตกต่าง หรือเหมือนกันอย่างไร

ในรายงานดังกล่าวได้บรรยายและเน้นถึงระบบจัดการฐานข้อมูลในลักษณะของความสัมพันธ์ระหว่างบทบาทของคน หน้าที่การประมวลผล (Processing Functions) และการไหลไปของข้อมูล (Information Flow) และเสนอลักษณะสำคัญของการย่อข้อมูล (Data Abstraction) เป็น 3 ระดับคือ คือ เค้าร่างภายใน (Internal Schema) เค้าร่างมโนภาพ (Conceptual Schema) และเค้าร่างภายนอก (External Schema) ดังแสดงในรูปที่ 3.2

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.2 ระบบการจัดการฐานข้อมูลแบบ 3 ระดับ (Whittington, 1988)

1. เค้าร่างภายใน (Internal Schema) อธิบายการจัดการข้อมูลในระดับต่ำ แต่มิใช่การเก็บข้อมูลระดับกายภาพ เพราะมิได้อธิบายถึงการเก็บข้อมูลเป็นทรงกระบอก (Cylinder) หรือเป็นวง (Track) (Date, 1986) ลักษณะการเก็บข้อมูลกายภาพจะต้องพึ่งพิงโครงสร้างของหน่วยเก็บ เพราะหน่วยเก็บแต่ละชนิดจะมีโครงสร้างแตกต่างกันไป เค้าร่างภายในเป็นการอธิบายถึงข้อมูลในระดับที่อยู่เหนือจากกายภาพ และไม่พึ่งพิงโครงสร้างของหน่วยเก็บ โดยจะอธิบายถึงประเภทระเบียน (Record) เขตข้อมูล (Field) มีอะไรบ้าง เก็บอย่างไร ใช้อะไรเป็นดรรชนี มีการเรียงลำดับกายภาพ (Physical Sequence) อย่างไร ความยาวของระเบียน

2. เค้าร่างมโนภาพ (Conceptual Schema) หรือการมองภาพรวมของข้อมูลธุรกิจทั้งหมด (Fleming and Halle, 1989) ภาพที่เห็นในระดับนี้จะเป็นตัวแทนของเนื้อหาทั้งหมดของฐานข้อมูล ซึ่งเป็นการมองข้อมูลที่แท้จริงที่ควรจะเป็น โดยไม่สนใจว่าจะใช้ภาษาอะไรในการจัดการหรือใช้ฮาร์ดแวร์อะไรในการเก็บข้อมูล ผู้บริหารฐานข้อมูลที่เรียกว่า Database Administrator หรือ DBA ทำหน้าที่รวบรวม (Integrate) ความต้องการของผู้ใช้ในองค์กรทั้งหมด และกำหนดเนื้อหาของฐานข้อมูลควรจะเก็บข้อมูลอะไรบ้าง อย่างไร และสร้างเป็นเค้าร่างมโนภาพ โดยไม่สนใจโครงสร้างของหน่วยเก็บ หรือกลยุทธ์ในการเข้าถึงข้อมูล สิ่งที่จะกำหนดในเค้าร่างมโนภาพมีดังนี้

2.1 กำหนดลักษณะของข้อมูล (Data Definition) อธิบายประเภทข้อมูล ความยาวของเขตข้อมูลที่เล็กที่สุดในฐานข้อมูล

2.2 ความสัมพันธ์ของข้อมูล (Data Relationship) กำหนดการเชื่อมโยงของข้อมูลระหว่างเอนทิตีที่มีความสัมพันธ์กัน

3. เค้าร่างภายนอก (External Schema) เค้าร่างภายนอกนี้เป็นการมองข้อมูลของผู้ใช้แต่ละคนซึ่งอาจเป็นผู้ใช้ขั้นสุดท้าย (End User) ต้องการใช้ข้อมูลในลักษณะเชื่อมตรง (On-Line) หรือเป็นนักเขียนโปรแกรมจะใช้ข้อมูลในแต่ละโปรแกรมเพื่อประมวลผลตามต้องการ เค้าร่างภายนอกจะมองภาพข้อมูลจากฐานข้อมูลเป็นส่วน ๆ อธิบายมุมมองของผู้ใช้แต่ละคน

ระบบฐานข้อมูลที่มีโครงสร้างตามแนวทางที่ ANSI/SPARC กำหนดไว้ และนิยมใช้เป็นแบบจำลองข้อมูลในการพัฒนาระบบจัดการฐานข้อมูลคือฐานข้อมูลแบบสัมพันธ์ (Relational Database) ฐานข้อมูลชนิดนี้จะมองข้อมูลในลักษณะของตาราง (Table) 2 มิติ แต่ละตารางจะแยกจากกัน ไม่เป็นลำดับชั้น หรือโครงสร้างที่ซับซ้อน การสืบค้นข้อมูลจึงสามารถเข้าถึงแต่ละตารางได้โดยตรงโดยไม่ต้องกำหนดเส้นทาง (Path) ไว้ล่วงหน้า และสามารถเชื่อมโยงตาราง 2 ตารางด้วยเขตข้อมูลที่มีความสัมพันธ์กัน ในระบบฐานข้อมูลแบบสัมพันธ์มีส่วนประกอบต่าง ๆ และคำศัพท์ (Terminology) ที่ใช้เรียกกันดังนี้ (Tsai, 1988)

1. รีเลชัน (Relation) หมายถึง ตาราง ซึ่งเป็นที่ที่เก็บข้อมูลที่มีความสัมพันธ์กัน ตารางประกอบด้วยรายการ (Occurrences) และแต่ละรายการจะประกอบด้วยเขตข้อมูล (Field)

2. แอตตริบิว (Attribute) แต่ละสดมภ์ของรีเลชันจะเรียกว่า แอตตริบิว หรือเขตข้อมูล (Field) นั่นเอง

3. โดเมน (Domain) คือ เซทของข้อมูลที่แต่ละแอตตริบิวสามารถมีค่าได้ เช่น โดเมนของแอตตริบิว ฤดู (Season) คือ ฤดูใบไม้ร่วง ฤดูร้อน ฤดูใบไม้ผลิ ฤดูฝน เป็นต้น

4. ทัพเพิล (Tuple) คือแถว (Row) ของตาราง (Table) แต่ละทัพเพิลจะแทนข้อมูล 1 รายการ (Record Occurrence) และ 1 รีเลชัน จะประกอบด้วยเซทของทัพเพิล

5. กุญแจหลัก หรือคีย์หลัก (Primary Key) และแคนดิเดทคีย์ (Candidate Key) เขตข้อมูลที่สามารถกำหนดข้อมูลของทัพเพิลได้ชัดเจน หรือยูนีค (Uniquely Identify) เรียกว่า แคนดิเดทคีย์ (Candidate Key) แต่ละรีเลชันมีได้หลายแคนดิเดทคีย์ แต่หนึ่งในหลายแคนดิเดทคีย์จะถูกกำหนดให้ทำหน้าที่เป็นคีย์หลักซึ่งอาจประกอบด้วยหนึ่งหรือหลายแอตตริบิว

6. กุญแจภายนอก หรือฟอเรนคีย์ (Foreign Key) เขตข้อมูลที่มีค่าตรงกับค่าของคีย์หลักในอีกรีเลชัน

ระบบจัดการฐานข้อมูล (Database Management System - DBMS)

เพื่อให้การทำงานระบบฐานข้อมูลเป็นไปตามสถาปัตยกรรมดังที่กล่าวมาแล้วข้างต้น จะต้องมีตัวจัดการฐานข้อมูลที่สอดคล้องกับโครงสร้างนั้น ระบบฐานข้อมูลจึงมีส่วนประกอบที่สำคัญอีกส่วนคือ ระบบจัดการฐานข้อมูล (DBMS)

ระบบจัดการฐานข้อมูลประกอบด้วยซอฟต์แวร์รูทีน (Software Routine) จำนวนมากที่มีความสัมพันธ์กัน โดยแต่ละรูทีนจะทำงานเฉพาะอย่าง (Tsai. 1988) แต่หน้าที่สำคัญของระบบจัดการฐานข้อมูล (Whittington. 1988) คือ

1. การกำหนดรายละเอียดข้อมูล (Data Definition) การสร้าง และการปรับปรุงโครงสร้างของฐานข้อมูล ตลอดจนทำหน้าที่บรรจุข้อมูลดำเนินการเข้าสู่ฐานข้อมูล

2. การจัดการข้อมูล (Data Manipulation) ทำการปรับปรุง (Update) หรือค้นสืบ (Retrieval) ข้อมูลจากฐานข้อมูลทันทีพร้อม ๆ กันหลายทาง และตอบสนองตามความต้องการของผู้ใช้ โดยมิให้เกิดความขัดแย้ง

3. การควบคุมข้อมูล (Data Control) กำหนดและควบคุมการใช้ข้อมูลให้เป็นไปตามเอกสิทธิ (Privileges) เพื่อรักษาความปลอดภัย (Security) และบูรณภาพของข้อมูล (Data Integrity)และป้องกันมิให้เกิดความเสียหายต่อข้อมูล รวมถึงการสำรองข้อมูล และสามารถกู้กลับ (Recovery) เมื่อระบบเกิดปัญหาด้วย

บูรณภาพของข้อมูล (Data Integrity) (Fleming and Halle, 1989) หมายถึง กฎธุรกิจ (Business Rules) ที่เป็นข้อกำหนดเกี่ยวกับค่าของข้อมูล หรือความสัมพันธ์ของข้อมูลควรเป็นอย่างไร บูรณภาพของข้อมูลสามารถแบ่งเป็น 3 อย่างคือ

3.1 บูรณภาพเอนทิตี (Entity Integrity) กฎนี้กล่าวไว้ว่า สดมภ์หรือแอตตริบิวที่เป็นส่วนหนึ่งของคีย์หลัก (Primary Key - คีย์หลักอาจประกอบด้วยหนึ่งหรือหลายแอตตริบิว) จะมีค่าว่าง (Null Value) หรือค่าที่ไม่รู้จักไม่ได้ ทั้งนี้เพราะคีย์หลักนี้จะเป็นตัวชี้เฉพาะข้อมูลแต่ละแถวหรือระเบียน จึงไม่ควรมีค่าว่างหรือค่าที่ไม่รู้จัก

3.2 บูรณภาพอ้างอิง (Referential Integrity) ข้อกำหนดเกี่ยวกับฟอเรนคีย์ (Foreign Key)

ฟอเรนคีย์ หมายถึง แอตตริบิวหรือกลุ่มของแอตตริบิวของตารางหนึ่ง มีค่าตรงกับคีย์หลักของอีกตารางหนึ่ง เนื่องจากบูรณภาพอ้างอิงเป็นการเชื่อมโยงกันระหว่าง 2 ตาราง เพื่อใช้อ้างอิงข้อมูลที่มีความสัมพันธ์กัน ดังนั้นบูรณภาพอ้างอิงจึงกล่าวไว้ว่า ตารางใดที่มีฟอเรนคีย์ ค่าทุกค่าของฟอเรนคีย์ไม่ว่าจะเป็นค่าว่างหรือไม่จะต้องมีค่าตรงกับคีย์หลักของอีกตารางหนึ่ง

การเพิ่ม การปรับปรุง หรือลบข้อมูลของทั้ง 2 ตารางย่อมมีผลกระทบต่อกัน แต่จะมีผลกระทบอย่างไรขึ้นอยู่กับข้อมูลทางธุรกิจของแต่ละฐานข้อมูล

3.3 บูรณภาพโดเมน (Domain Integrity) โดเมน คือ ค่าที่เป็นไปได้ของแต่ละแอตตริบิว ไม่ว่าแอตตริบิวนั้นจะเป็นคีย์หลัก ฟอเรนคีย์ หรือแอตตริบิวอื่นที่ไม่ใช่คีย์ในตาราง ผู้ใช้สามารถกำหนดค่าที่เป็นไปได้ของแต่ละแอตตริบิวไว้ เพื่อใช้ตรวจสอบข้อมูลที่จะบรรจุในแต่ละแอตตริบิว

4. รวบรวมสถิติการใช้งานของระบบ เพื่อให้สามารถควบคุมการทำงานของระบบ และปรับปรุงผลการดำเนินการ (Performance) ให้ดีขึ้นตามที่คาดหวังไว้

5. ใช้ภาษารุ่นที่ 4 (Fourth-Generation Languages) เช่น แบบฟอร์ม (Forms) โปรแกรมก่อกำเนิดข้อมูลนำเข้า (Input-Program Generators) โปรแกรมก่อกำเนิดรายงาน (Report-Program Generators) เป็นต้น เพื่อเพิ่มความสามารถในการประยุกต์ใช้งาน

6. มีพจนานุกรมข้อมูล (Data Dictionary) เพื่อใช้ช่วยพัฒนาระบบฐานข้อมูล

7. สามารถเชื่อมโยงกับระบบประมวลผลคำ (Word Processing) แผ่นตารางทำการ (Spreadsheet) และระบบงานอื่น ๆ

หน้าที่ของระบบจัดการฐานข้อมูล 3 ประการหลังนี้ เป็นส่วนที่ระบบในปัจจุบันได้พยายามจัดเตรียมให้มีขึ้น เพื่อให้ระบบสามารถเชื่อมโยงกับระบบงานอื่น ๆ และสามารถทำงานได้สมบูรณ์มากยิ่งขึ้น

<u>การเลือกระบบฐานข้อมูล</u>

แต่เดิมการพัฒนาระบบงานต่าง ๆ จะใช้ภาษายุคที่สามและเป็นการประมวลผลของแฟ้มสัญนิยม (Conventional File) การเขียนโปรแกรมประยุกต์จะยุ่งยาก ใช้เวลานาน และอิงกับโครงสร้างของข้อมูล เมื่อมีการเปลี่ยนแปลงข้อมูล หรือต้องการเพิ่มข้อมูลใหม่ ๆ เข้าไปในระบบจะต้องแก้ไขปรับปรุงโปรแกรม หรือเขียนโปรแกรมใหม่ พร้อมทั้งทดสอบโปรแกรมนั้น ๆ ซึ่งเป็นเรื่องที่ต้องใช้ค่าใช้จ่าย และใช้เวลามาก ภาษาที่ใช้มักจะเป็นภาษาที่ใช้เฉพาะแต่ละระบบ ถ้าเปลี่ยนระบบจะต้องเรียนรู้ภาษาใหม่ การเก็บข้อมูลก็ซ้ำซ้อน ทำให้ต้องใช้เนื้อที่ในการเก็บข้อมูลมากกว่าที่ควร ไม่มีระบบรักษาความปลอดภัยที่เหมาะสม และที่สำคัญทำให้ยากต่อการบำรุงรักษาให้ข้อมูลทั้งหมดมีความสอดคล้องกัน ถูกต้องตรงกัน ประกอบกับในปัจจุบันธุรกิจมีการแข่งขันสูงขึ้น ผู้บริหารแต่ละบริษัทมักมีคำถามหรือความต้องการใช้ข้อมูลในลักษณะต่าง ๆ ที่เปลี่ยนแปลงอยู่เสมอ และแต่ละครั้งมักต้องการคำตอบในเวลารวดเร็วหรือทันทีเพื่อให้สามารถตัดสินใจ

ได้ฉับไวด้วยข้อมูลที่ถูกต้องรวดเร็วทันต่อเหตุการณ์เสมอ จากปัญหาและลักษณะการใช้ข้อมูลที่เปลี่ยนไป ระบบฐานข้อมูลซึ่งเป็นระบบที่ได้รับการออกแบบมาเพื่อแก้ไขปัญหาดังกล่าว พร้อมทั้งการเพิ่มคุณลักษณะของการใช้ภาษาข้อคำถาม (Query Language) ที่เรียนรู้ง่าย จึงได้รับความสนใจมากขึ้น โดยเฉพาะระบบฐานข้อมูลแบบสัมพันธ์ซึ่งมีโครงสร้างที่เข้าใจง่าย ไม่ยุ่งยากซับซ้อน สามารถทำงานแบบข้อคำถาม หรือคิวรี (Query) ได้ดี บริษัทผู้ผลิตระบบฐานข้อมูลจึงให้ความสนใจฐานข้อมูลแบบสัมพันธ์ และพยายามปรับปรุงให้มีประสิทธิภาพมากขึ้น ปัจจุบันจึงมีระบบฐานข้อมูลแบบสัมพันธ์จำหน่ายในท้องตลาดหลายระบบ

จากคุณสมบัติที่ดีของระบบฐานข้อมูลแบบสัมพันธ์ ในการพัฒนาระบบสืบค้นหนังสือของห้องสมุด จึงเลือกใช้ระบบฐานข้อมูลแบบสัมพันธ์ โดยเลือกใช้ระบบฐานข้อมูลชื่อ ไมโครซอฟต์แอคเซส (Microsoft Access) ซึ่งพัฒนาภายใต้โปรแกรมไมโครซอฟต์วินโดวส์ (Microsoft Windows) ผลิตโดยบริษัทไมโครซอฟต์ คอร์ปอเรชัน (Microsoft Corporation) ทั้งนี้เพราะแอคเซสเป็นระบบฐานข้อมูลแบบสัมพันธ์ที่ประกอบด้วยระบบจัดการฐานข้อมูลแบบสัมพันธ์ (Relational Database Management System หรือ RDBMS) ที่มีคุณสมบัติสำคัญ 3 ประการ (Viescas, 1994) คือ

1. การกำหนดรายละเอียดของข้อมูล (Data Definition) คือการกำหนดข้อมูลอะไรบ้างที่ควรจะเก็บในฐานข้อมูล กำหนดประเภทข้อมูล (ตัวเลข อักขระ บันทึกข้อความ ฯลฯ) รูปแบบควรเป็นอย่างไร มีความสัมพันธ์แบบไหน จะตรวจสอบข้อมูลนั้น ๆ อย่างไร

2. การจัดการข้อมูล (Data Manipulation) สามารถเลือกข้อมูลเฉพาะที่สนใจ เรียงลำดับข้อมูลตามต้องการ หรือเปลี่ยนแปลงข้อมูล

3. การควบคุมข้อมูล (Data Control) สามารถกำหนดสิทธิของผู้ใช้ระบบในลักษณะต่าง ๆ เพื่อให้ผู้มีสิทธิเท่านั้นที่จะใช้ข้อมูลได้ในขอบเขตที่ได้รับอนุญาต เช่น อ่านได้อย่างเดียว แก้ไขข้อมูลไม่ได้ เป็นต้น

แอคเซสนั้นนอกจากจะเป็นระบบฐานข้อมูลที่มีคุณสมบัติตามข้อกำหนดที่ดีของระบบฐานข้อมูลแบบสัมพันธ์แล้ว แอคเซสสามารถเชื่อมโยงการทำงานกับโปรแกรมสำเร็จรูปอื่น ๆ ที่มีประสิทธิภาพ และได้รับความนิยมอยู่ในขณะนี้ เช่น โปรแกรมประมวลผลคำชื่อ Microsoft Word โปรแกรมไมโครซอฟต์เอคเซล (Microsoft Excel) โปรแกรมวิชวลเบสิก (Visual Basic) เป็นต้น นอกจากนี้แอคเซสสามารถทำงานร่วมกับระบบฐานข้อมูลชนิดอื่นที่มีจำหน่ายในท้องตลาดและมีผู้นิยมใช้ เช่น Dbase Paradox Foxpro และ Betrieve รวมทั้งภาษาเอสคิวแอล (SQL-Structured Query Language) เป็นต้น และที่สำคัญแอคเซสสามารถเก็บข้อมูลที่มีเขตข้อมูลขนาดใหญ่ถึง 64 กิโลไบต์

เรียกเขตข้อมูลประเภทนี้ว่า Memo และสามารถสืบค้นข้อมูลได้ด้วยการใช้วายคาร์ด (Wildcard) จึงเหมาะสมที่จะใช้เก็บบทคัดย่อของงานวิจัยและหนังสือ

ดังได้กล่าวแล้วข้างต้นว่าไมโครซอฟต์แอคเซสเป็นระบบที่สร้างขึ้นสำหรับการประยุกต์ใช้งานภายใต้ไมโครซอฟต์วินโดวส์โดยเฉพาะ ดังนั้นแอคเซสจึงสามารถใช้สิ่งอำนวยความสะดวกของวินโดวส์ได้แก่ การแลกเปลี่ยนข้อมูลด้วย Dynamic Data Exchange (DDE) และการเชื่อมโยงข้อมูลด้วย Object Linking and Embedding (OLE) ทั้ง DDE และ OLE ทำให้แอคเซสทำงานและใช้ข้อมูลร่วมกับโปรแกรมประยุกต์อื่น ๆ ภายใต้วินโดวส์ที่มีคุณสมบัติทั้ง 2 นี้ได้เป็นอย่างดี ซึ่งทำให้เกิดการใช้ข้อมูลร่วมกันอย่างมีประสิทธิภาพ ลดความซ้ำซ้อนของข้อมูล และทำให้สามารถทำงานหลายระบบได้พร้อม ๆ กัน

สถาปัตยกรรมของแอคเซส

ลักษณะฐานข้อมูลของแอคเซสจะเก็บข้อมูลทุกอย่างไว้ในแฟ้มหนึ่งแฟ้มซึ่งมีนามสกุลเป็น MDB แฟ้มข้อมูลนี้จะประกอบด้วยสิ่งต่าง ๆ ที่เรียกว่า ออบเจ็กต์ (Object) ซึ่งมีอยู่ 6 ชนิดคือ ตาราง (Tables) ข้อคำถามหรือคิวรี (Queries) ฟอร์ม (Forms) รายงาน (Reports) มาโคร (Macros) และโมดูล (Modules) ในที่นี้จะขอกล่าวถึงตารางของไมโครซอฟต์แอคเซส ซึ่งใช้ในการวิจัยครั้งนี้

ตาราง (Tables) ในไมโครซอฟต์แอคเซส คือ ออบเจ็กต์ที่เก็บข้อมูล และในฐานข้อมูลหนึ่งจะประกอบด้วยตารางหลายตาราง โดยแต่ละตารางจะเก็บข้อมูลที่มีความสัมพันธ์ในเรื่องเดียวกัน เช่น ตารางนักศึกษา ข้อมูลแต่ละระเบียน (Record) หรือข้อมูล 1 คน จะประกอบด้วยเขตข้อมูลชนิดต่าง ๆ (Data Type) ซึ่งอาจเป็นตัวเลข อักขระ หรือแม้แต่ออบเจ็กต์แบบ OLE (Object Linking and Embedding) ที่ใช้เก็บข้อมูลที่เป็นรูปภาพ หรือกราฟ ซึ่งอาจเป็นข้อมูลที่เชื่อมโยงมาจากโปรแกรมประยุกต์อื่นภายใต้วินโดวส์ หรือเป็นการฝังตัวอยู่ในแอคเซสก็ได้ สำหรับเขตข้อมูลแต่ละเขตสามารถกำหนดวิธีหรือกฎเกณฑ์ต่างๆ เพื่อตรวจสอบความถูกต้องของข้อมูล (Validation Rule) หรือค่าที่เป็นไปได้ของข้อมูลแต่ละเขต

ตารางแต่ละตารางจะมีคีย์หลัก (Primary Key) เพื่อใช้กำหนดรายละเอียดของข้อมูลแต่ละระเบียน และใช้เป็นข้อมูลในการสืบค้นซึ่งจะได้ข้อมูลที่เป็นหนึ่งเดียว (Unique) ไม่ซ้ำ คือชี้เฉพาะรายการนั้น เช่น รหัสนักศึกษา สามารถใช้เป็นคีย์หลัก เพราะรหัสนักศึกษาจะไม่ซ้ำกัน อย่างไรก็ตามผู้ใช้ยังสามารถกำหนดเขตข้อมูลอื่น ๆ (ยกเว้นข้อมูลบางประเภท เช่น OLE, Memo) เป็นดรรชนีเพื่อประโยชน์ในการเข้าถึง หรือสืบค้นข้อมูลได้รวดเร็วขึ้น ผู้ใช้สามารถสร้างความสัมพันธ์ระหว่างตารางเพื่อเชื่อมโยงข้อมูลได้ ความสัมพันธ์ในฐานข้อมูลมีได้ 2 แบบคือ

1. ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-To-One, 1:1 Relationship)
2. ความสัมพันธ์แบบหนึ่งต่อหลาย (One-To-Many, 1:N Relationship)

นอกจากนี้ผู้ใช้อาจสร้างความสัมพันธ์ระหว่างตารางและกำหนดให้แอคเซสตรวจสอบความสัมพันธ์ของข้อมูล 2 ตารางนั้นให้อัตโนมัติด้วย เพื่อให้เกิดบูรณภาพอ้างอิง (Referential Integrity) ของข้อมูล กล่าวคือจะเพิ่มข้อมูลในตารางรองได้ต่อเมื่อได้มีข้อมูลที่สัมพันธ์กันในตารางหลักแล้วเท่านั้น หรือจะลบข้อมูลในตารางหลักก่อนลบข้อมูลในตารางรองไม่ได้ ดังเช่น ในระบบฐานข้อมูลหนังสือ ความสัมพันธ์ระหว่างตารางนักศึกษาที่เป็นสมาชิกห้องสมุด (ตารางหลัก) และตารางยืม-คืนหนังสือ จะต้องมีชื่อนักศึกษาในตารางหลักก่อนจึงสามารถเพิ่มการยืม-คืนหนังสือของนักศึกษานั้นในตารางการยืม-คืนหนังสือได้ และในทางกลับกันจะลบชื่อนักศึกษานั้นจากตารางหลักได้เมื่อได้ลบระเบียนในตารางการยืม-คืนหนังสือก่อน เป็นต้น แอคเซสจะดูแลข้อมูลให้เป็นไปตามกฎบูรณภาพอ้างอิงนี้ เมื่อผู้ใช้ทำงานผิดขั้นตอน ระบบจะมีข้อความเตือนให้ทราบโดยอัตโนมัติ

<u>สถาปัตยกรรมของวิชวลเบสิก</u>

วิชวลเบสิก (Visual Basic) มีสภาพแวดล้อมสำหรับการพัฒนาโปรแกรมบนวินโดวส์ประกอบด้วยเครื่องมือต่าง ๆ ครบถ้วน เช่น ส่วนของการออกแบบเพื่อการติดต่อกับผู้ใช้ (User Interface) ส่วนออกแบบเมนู (Menu Designer) การสร้างรายงาน (Report Writer) เอดิเตอร์สำหรับป้อนโปรแกรม และส่วนการตรวจหาข้อผิดพลาดในโปรแกรม (Debugger) องค์ประกอบเหล่านี้เอื้ออำนวยต่อการทำงานในการพัฒนาโปรแกรมเป็นอย่างมาก

ในด้านของตัวภาษาวิชวลเบสิก ได้นำไวยกรณ์ของเบสิกา (BASICA) และจีดับบลิวเบสิก (GW-BASIC) มาใช้ โดยสนับสนุนความสามารถเดิมเกือบทั้งหมด นอกจากนี้ยังได้เพิ่มโปรแกรมแบบมีโครงสร้างของควิกเบสิก (QuickBasic) ซึ่งคล้ายกับภาษาที่มีโครงสร้าง เช่น ปาสคาล (Pascal) หรือภาษาซี (C Language) เข้าไปด้วย

ในวิชวลเบสิกนั้น การพัฒนาและเขียนโปรแกรมจะเป็นไปในอีกรูปแบบหนึ่ง กล่าวคือ ในการเขียนโปรแกรมแบบเดิมนั้น ผู้พัฒนาโปรแกรมจะต้องออกแบบหน้าจอ ระบุตำแหน่งการแสดงผล คิดหาขั้นตอนและการทำงานอื่น ๆ ก่อน จากนั้นจึงจะเขียนโปรแกรม โปรแกรมที่ได้จะอธิบายและสั่งงานคอมพิวเตอร์เป็นลำดับไป แต่ในวิชวลเบสิกจะใช้หลักของภาพและการมองเห็นโดยเริ่มจากการออกแบบวินโดว์ย่อยหรือที่ในวิชวลเบสิกเรียกว่า "ฟอร์ม" ในฟอร์มจะประกอบด้วยสิ่งต่าง ๆ เรียกว่า ออบเจ็กต์ (Object) เช่น ข้อความ ช่องรับข้อความ ปุ่ม (Button) เมื่อกำหนดสิ่ง

เหล่านี้ครบตามความต้องการแล้ว จึงระบุว่าองค์ประกอบแต่ละอย่างจะทำอย่างไร โดยเขียนโปรแกรมย่อยๆ ปะเข้าไปกับออบเจ็กต์เหล่านี้ การทำงานบนวินโดวส์ในลักษณะนี้เป็นแบบ Event-Driven คือ ขึ้นกับเหตุการณ์ (Event)

การทำงานตามแนวทางแบบ Event-Driven จะกำหนดรูปแบบของจอภาพหรือส่วนติดต่อกับผู้ใช้ และระบุว่าถ้าเกิดเหตุการณ์อย่างใดอย่างหนึ่งกับสิ่งนี้จะต้องทำอย่างไร จะต้องเขียนโปรแกรมเพื่อระบุว่าจะต้องทำอย่างไรกับสิ่งนี้

นอกจากออบเจ็กต์จะมีการตอบสนองต่อเหตุการณ์ต่าง ๆ ที่กำหนดแล้ว ทุกออบเจ็กต์จะมีลักษณะหรือคุณสมบัติของตัวเอง (Property) เช่น ช่องรับข้อความ (Text Box) จะมีชื่อ, ข้อความในนั้น, ความกว้าง, ความสูง, สี และอื่น ๆ โดยผู้พัฒนาโปรแกรมสามารถอ้างถึงหรือเปลี่ยนคุณสมบัติเหล่านี้ได้ในขณะที่โปรแกรมทำงานอยู่

ในการที่จะกระทำอย่างใดอย่างหนึ่งกับออบเจ็กต์นั้นจะมีสิ่งที่เรียกว่ากระบวนการ (Method) ซึ่งเป็นกระบวนการทำงานของออบเจ็กต์ โดยออบเจ็กต์แต่ละแบบก็อาจจะมีกระบวนการที่แตกต่างกันออกไป เช่น ถ้าต้องการสั่งให้เลื่อนตำแหน่งของข้อความ (Label) ก็จะมีกระบวนการหรือ Method ชื่อ Move ของข้อความ เพื่อทำงานนี้

โดยสรุปแล้ว รูปแบบของหลักการในวิชวลเบสิกคือ การออกแบบจอภาพและเขียนโปรแกรมสำหรับแต่ละเหตุการณ์ปะเข้าไปยังออบเจ็กต์ต่าง ๆ ให้ทำงานตามเหตุการณ์ที่เกิดขึ้น โดยทุกออบเจ็กต์จะมีคุณสมบัติเฉพาะที่สามารถเปลี่ยนแปลงได้ของตัวเอง

ในวิชวลเบสิกจะประกอบไปด้วยวินโดวส์หลักในการทำงานดังนี้

1. วินโดว์ฟอร์ม Form) เป็นวินโดว์ว่าง ๆ สำหรับสร้างองค์ประกอบของแอพพลิเคชันโดยนำออบเจ็กต์มาใส่ในฟอร์ม เมื่อเริ่มทำงานโปรแกรมวิชวลเบสิกจะมีฟอร์มเปล่านี้ขึ้นมาเสมอ

2. วินโดว์เครื่องมือ (ToolBox) เป็นที่รวมของออบเจ็กต์ต่าง ๆ ที่จะนำมาประกอบในแอพพลิเคชัน

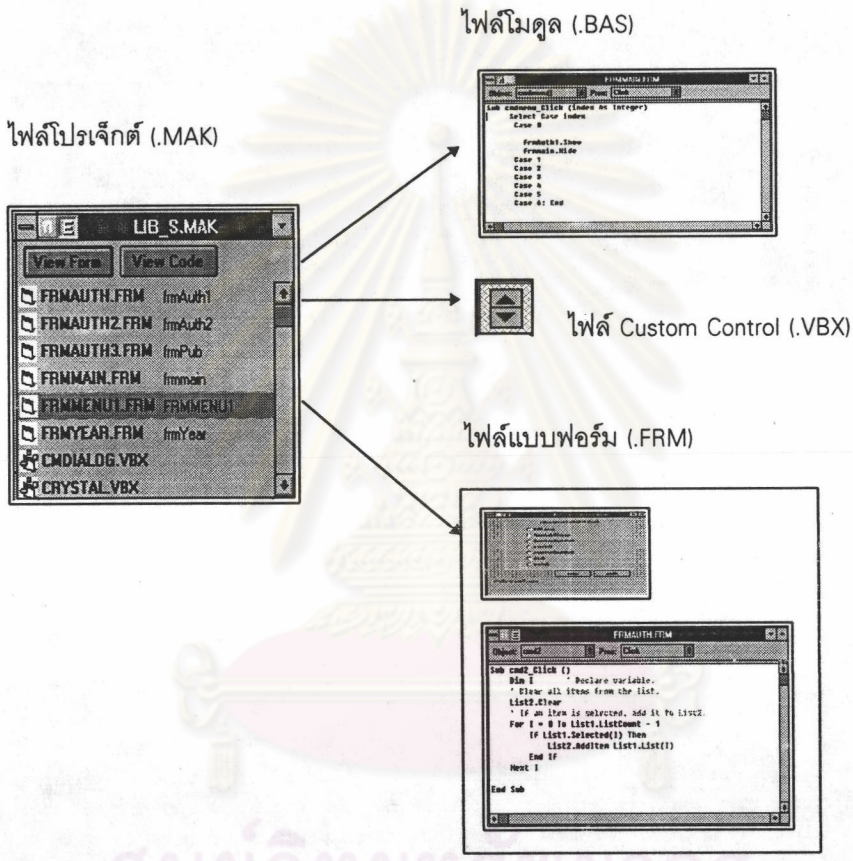
3. วินโดว์คุณสมบัติ (Properties) เป็นหน้าต่างที่แสดงคุณสมบัติทั้งหมดของออบเจ็กต์ที่ถูกเลือก ซึ่งสามารถจะเลือกคุณสมบัติเฉพาะของแต่ละออบเจ็กต์ได้โดยระบุหรือเลือกคุณสมบัติตามรายการที่กำหนดให้ในตาราง

4. วินโดว์ตรวจสอบ (Debug) เป็นหน้าต่างที่ใช้ในการตรวจสอบค่าของตัวแปรหรือนิพจน์ และสั่งงานในการทดสอบการทำงานทีละคำสั่ง (Single Step)

5. วินโดว์ป้อนโปรแกรม (Editor) เป็นวินโดว์สำหรับป้อนและแก้ไขโปรแกรม โดยจะมีช่องสำหรับแสดงออบเจ็กต์ และโพรซีเยอร์ (Procedure)

6. วินโดว์โครงงาน (Project) ใช้ในการควบคุมไฟล์ต่าง ๆ ทั้งหมดที่จะประกอบกันขึ้นเป็นแอพพลิเคชัน ซึ่งไฟล์ต่าง ๆ ที่ถูกรวมอยู่ในโปรเจ็กต์ แบ่งออกเป็น 3 ประเภทคือ ไฟล์โมดูล ไฟล์ควบคุม และไฟล์ฟอร์ม

ความสัมพันธ์ของไฟล์ต่าง ๆ ในโปรเจ็กต์แสดงดังรูปที่ 3.3



รูปที่ 3.3 โครงสร้างของโปรเจ็กต์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ขั้นตอนการออกแบบฐานข้อมูล (Phase of Database Design) (Mannila and Raiha, 1992)

1. วิเคราะห์ความต้องการของผู้ใช้ระบบ (Requirement Analysis)

ผู้เชี่ยวชาญการออกแบบฐานข้อมูลจะต้องรวบรวมความต้องการของผู้ใช้ระบบจากหน่วยงานต่างๆ และวิเคราะห์ความต้องการและปัญหาของระบบและของผู้ใช้ในปัจจุบัน รวมทั้งความต้องการขยายงานในอนาคต เพื่อให้แน่ใจว่าฐานข้อมูลเก็บข้อมูลที่จำเป็นในการประยุกต์ใช้งานอย่างครบถ้วน ผลที่ได้จากขั้นตอนนี้คือ ทราบการกระทำ (Actions) ในระบบ และการปฏิบัติ

งานอย่างครบถ้วน ผลที่ได้จากขั้นตอนนี้คือ ทราบการกระทำ (Actions) ในระบบ และการปฏิบัติการ (Operations) ที่ใช้ในฐานข้อมูลนี้

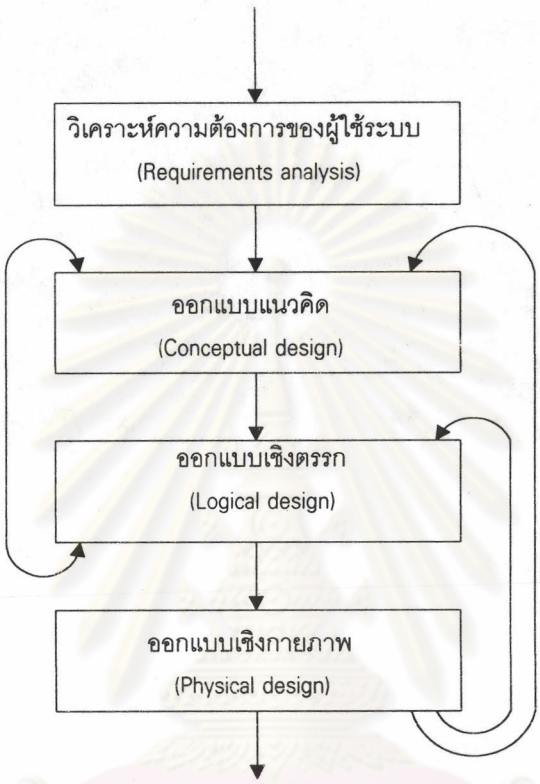
2. ออกแบบแนวคิด (Conceptual Design) ผลของขั้นตอนนี้เป็นการอธิบายโครงสร้างของข้อมูลว่าควรมีข้อมูลอะไรบ้างในฐานข้อมูลซึ่งเป็นโครงสร้างในระดับบน (High-Level Term) เรียกว่า เค้าร่างฐานข้อมูล (Database Schema)

ในขั้นตอนนี้ต้องศึกษาแนวคิดต่างๆ รวมทั้งแนวคิดของผู้ใช้ระบบแต่ละท่าน เพื่อหาแนวคิดที่จำเป็นซึ่งได้จากคำอธิบายขั้นตอนการปฏิบัติงานที่ได้จากการวิเคราะห์ความต้องการในขั้นตอนแรก เพราะเป้าหมายที่จำเป็นของการออกแบบฐานข้อมูลคือ การเก็บข้อเท็จจริงที่ไม่ซ้ำซ้อน ในขั้นตอนนี้อาจใช้แบบจำลองข้อมูล (Data Model) เพื่อใช้อธิบายแนวคิดต่าง ๆ และการทำทำวิจัยครั้งนี้ก็เลือกใช้แบบจำลองข้อมูลเชิงตรรก (Logical Data Model) มาจำลองแนวคิด เพราะเป็นแบบจำลองที่เข้าใจง่าย ไม่อิงกับโปรแกรมประยุกต์ หรือเทคโนโลยีใด ๆ สามารถขยายขอบเขตงานโดยมีผลกระทบต่องานที่มีอยู่แล้วน้อยมากหรืออาจไม่มีเลย ใช้แสดงโครงสร้างและการใช้ข้อมูลได้สอดคล้องกับแนวทางทางธุรกิจ

3. ออกแบบเชิงตรรก (Logical Design) เพื่อให้ได้ข้อมูลที่ไม่ซ้ำซ้อน (Redundancy) และผลจากการที่เก็บข้อมูลไม่ซ้ำซ้อนจึงเป็นการลดความผิดพลาดในการปรับปรุงข้อมูล ในขั้นตอนนี้จะใช้กฎนอร์มัลไลเซชัน (Normalization Rules) และดีนอร์มัลไลเซชัน (Denormalization) ผลที่ได้จากขั้นตอนนี้คือ คำอธิบายโครงสร้างที่จะใช้เก็บข้อมูลในฐานข้อมูล รวมทั้งการกำหนดประเภท ความยาว และค่า (Values) ของข้อมูล

4. ออกแบบเชิงกายภาพ (Physical Design) สร้างตารางให้มีโครงสร้างตามที่ได้กำหนดไว้ในขั้นตอนที่แล้ว กล่าวคือขั้นตอนนี้เป็นการเปลี่ยน (Transform) แบบจำลองข้อมูลที่ได้จากขั้นตอนที่ 3 ให้มาอยู่ในรูปแบบทางกายภาพ ในขั้นตอนนี้จะต้องทราบถึงรูปแบบต่างๆ ในมุมมองของผู้ใช้ระบบหรือคิวรี (Query) ความถี่ในการเรียกใช้งาน หรือความถี่ในการประมวลผล เพื่อใช้ปรับโครงสร้างให้เหมาะสมกับการใช้งาน เพราะผลการทำงาน (Performance) ขึ้นอยู่กับโครงสร้างของตารางหรือแฟ้มข้อมูล ปริมาณข้อมูล และความถี่ของการใช้งาน ดังนั้นบางระบบอาจต้องใช้เวลาและต้องทำการประเมินผลอย่างสม่ำเสมอ เพราะปริมาณข้อมูลน้อยอาจมองไม่เห็นผลการทำงานที่ถูกต้อง เมื่อมีการเปลี่ยนแปลงจำนวนข้อมูลมากขึ้น จึงจะเห็นผลการทำงานที่ถูกต้อง ซึ่งอาจทำให้ต้องปรับโครงสร้างฐานข้อมูลใหม่

ในการทำงานแต่ละขั้นตอนควรมีการตรวจสอบผลป้อนกลับ (Feedback) เพื่อให้แน่ใจว่าผลที่ได้ในแต่ละขั้นตอนถูกต้องตามที่ผู้ออกแบบต้องการ ดังรูปที่ 3.4



รูปที่ 3.4 ขั้นตอนการออกแบบฐานข้อมูล (Mannila and Raiha, 1992)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย