บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

ในการพัฒนาระบบสารสนเทศเพื่อการจัดการสำหรับแผนกจัดการสำนักงาน มหาวิทยาลัยฟาร์อีสเทอร์น จังหวัดเชียงใหม่ ผู้ศึกษาได้ค้นคว้าเอกสารและงานวิจัยที่เกี่ยวข้องโดยมีรายละเอียดตามหัวข้อที่กำหนดตามลำดับดังนี้

2.1 แนวความคิดเกี่ยวกับระบบฐานข้อมูล

1) ความหมายของระบบฐานข้อมูล

จรณิต แก้วกังวาล (2536) ได้ให้ความหมายของฐานข้อมูล (Database) หมายถึง กลุ่มของข้อมูล (Data Group) ที่ถูกรวบรวมเข้าไว้ด้วยกัน โดยครอบคลุมรายละเอียดต่างๆ เช่นในสำนักงานก็จะรวบรวมข้อมูลตั้งแต่หมายเลขโทรศัพท์ของผู้ที่มาติดต่อ จนถึงการเก็บเอกสารทุกอย่างของสำนักงาน ซึ่งข้อมูลจะมีส่วนที่สัมพันธ์กันและเป็นที่ต้องการออกมาใช้ให้เป็นประโยชน์ ข้อมูลนั้นอาจจะเกี่ยวกับบุคคล สิ่งของ สถานที่ หรือเหตุการณ์ใด ๆ ที่เราสนใจศึกษา ซึ่งข้อมูล (Data) อาจจะได้มาจากการสังเกต การนับหรือการวัด และข้อมูลอาจเป็นได้ทั้งตัวเลขหรือเป็นข้อความก็ได้ ที่สำคัญคือข้อมูลจะต้องเป็นสิ่งที่เป็นความจริง

รายละเอียดของข้อมูลต่างๆ ต้องนำมาเชื่อมโยงสัมพันธ์กันให้ตรงตามที่ต้องการ เพื่อสะดวกในการค้นหาและกรอกข้อมูลเพิ่มเติม

2) องค์ประกอบของระบบฐานข้อมูล

องค์ประกอบของระบบฐานข้อมูลในที่นี้ หมายถึง โครงสร้างสารสนเทศ (Information) ที่ประกอบด้วย Entity หลาย ๆ ตัว ซึ่งบรรดา Entity เหล่านี้จะต้องมีความสัมพันธ์กัน

Entity ในที่นี้หมายถึง สิ่งที่มีอยู่จริง อันได้แก่ บุคคล สถานที่ และสิ่งของ

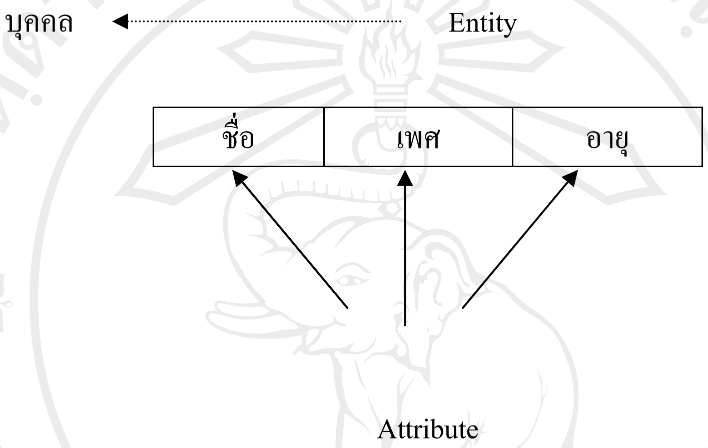
ตัวอย่าง Entity ในระบบฐานข้อมูลของโรงเรียนอาจได้แก่ อาจารย์ แผนกวิชา ประวัติการทำงาน หรือถ้าเป็น Entity ของบริษัทก็อาจจะได้แก่ พนักงานขาย ลูกค้า การสั่งซื้อ และประเภทสินค้า เป็นต้น

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

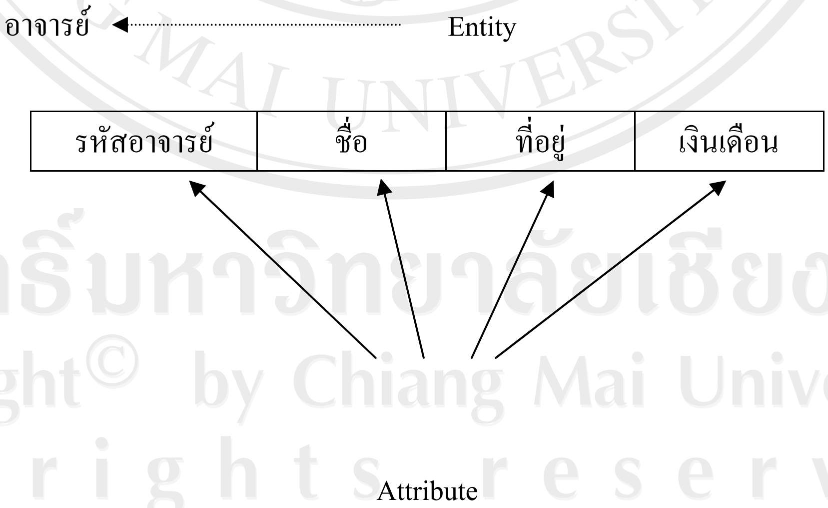
ส่วนของข้อมูลที่แสดงลักษณะและคุณสมบัติของ Entity เรียกว่า Attribute เช่น

- Attribute ของบุคคลได้แก่ ชื่อ เพศ อายุ ฯลฯ
- Attribute ของอาจารย์ได้แก่ รหัสอาจารย์ ชื่อ ที่อยู่ เงินเดือน ฯลฯ
- Attribute ของประเภทสินค้า ได้แก่ รหัสสินค้า สี ราคา ฯลฯ

ซึ่งพอจะแจกแจงได้ดังนี้

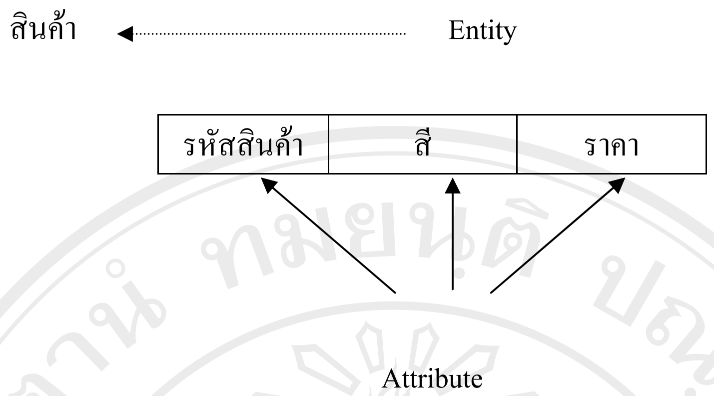


รูป 2.1 แสดง Entity ของบุคคล



รูป 2.2 แสดง Entity ของอาจารย์

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



รูป 2.3 แสดง Entity ประเภทของสินค้า

ดังนั้น ถ้าจะเปรียบไปแล้วจะเห็นว่า Entity นั้นเหมือนกับแฟ้มข้อมูล ส่วนAttributeนั้นเหมือนกับเขตข้อมูลนั่นเอง

เขตข้อมูล หรือ ฟิลด์ (Field) หมายถึง ตัวอักขระ (Character) แต่ละตัว ถูกนำมาประกอบกันเป็นกลุ่มคำที่มีความหมายขึ้น เช่น กลุ่มตัวอักษรที่ประกอบกันเป็นชื่อหรือนามสกุล กลุ่มตัวเลขที่ประกอบกันเป็นรหัสประจำตัว ราคา โดยเรียกกลุ่มข้อมูลนี้ว่า เขตข้อมูล เช่น เขตข้อมูลชื่อ เขตข้อมูลนามสกุล เขตข้อมูลรหัสประจำตัว เป็นต้น

ตัวอักขระ คือข้อมูลพื้นฐานที่เล็กที่สุดภายในแฟ้มข้อมูลคือ บิท (Bit : Binary Digit) ซึ่งเป็นหน่วยข้อมูลพื้นที่เก็บอยู่ในหน่วยความจำภายในคอมพิวเตอร์ บิทนี้จะแทนด้วยตัวเลข 1 ตัว ได้แก่ 1 หรือ 0 อย่างใดอย่างหนึ่ง เรียกตัวเลข 1 หรือ 0 นี้ว่า เป็นบิท 1 บิท ข้อมูลซึ่งได้แก่ ตัวอักขระแต่ละตัวเช่น A, B, ……Z, 0, 1, 2, ……., 9 และสัญลักษณ์พิเศษอื่น ๆ เช่น $, &, +, -, *, / ฯลฯ เมื่อจะถูกนำไปเก็บไว้ในคอมพิวเตอร์จะต้องถูกแปลงให้อยู่ในรูปของบิทหลายบิทที่มาประกอบกัน โดยตัวอักขระ 1 ตัวจะแทนด้วยบิท 7 หรือ 8 บิท ตัวอักขระแต่ละตัวจะเรียกได้อีกอย่างว่า ไบท์ (Byte) ตัวอย่างเช่น ตัวอักขระ A เมื่อเก็บอยู่ในคอมพิวเตอร์จะเก็บเป็น 1000001 ตัวอักขระ B จะเก็บเป็น 100010 เป็นต้น

เมื่อนำข้อมูลในหลายเขตข้อมูลหรือฟิลด์มารวมกันซึ่งมีลักษณะเป็นแถว ๆ จะเรียกว่าระเบียนหรือเรคคอร์ด (Record) ระเบียนแต่ละระเบียนของข้อมูลชนิดเดียวกัน จะสามารถนำมารวมกันเป็นแฟ้มข้อมูลหรือไฟล์ (File) ดังตัวอย่างต่อไปนี้

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

เขตข้อมูล (4)

แฟ้มข้อมูล (1) | ระเบียน (5)

รหัสอาจารย์	ชื่อ	ที่อยู่	เงินเดือน
1-350	นาวิน	กรุงเทพฯ	9,500
1-361	สุชาดา	ยโสธร	8,500
2-372	สมศักดิ์	เชียงใหม่	10,000
2-390	มยุรี	ตรัง	7,200
3-452	นิรมล	กรุงเทพฯ	8,900

รูป 2.4 แสดงรายละเอียดของแฟ้มข้อมูล

ถ้านำแฟ้มข้อมูลที่มีความสัมพันธ์กัน ซึ่งแต่ละแฟ้มข้อมูลจะประกอบด้วยเขตข้อมูลมารวมกันนั้นเรียกว่า ฐานข้อมูล

รูป 2.5 แสดงความสัมพันธ์ของตาราง

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

ฉะนั้นอาจจะกล่าวได้ว่า ถ้านำแฟ้มข้อมูลหลายแฟ้มข้อมูลมารวมกันก็จะกลายเป็นฐานข้อมูลได้ แต่ฐานข้อมูลที่สมบูรณ์จะต้องมีการเก็บคำอธิบายเกี่ยวกับโครงสร้างของฐานข้อมูลที่เรียกว่าพจนานุกรมข้อมูล (Data Dictionary) หรืออาจเรียกอีกอย่างหนึ่งว่าเมตาดาต้า (Meta Data) ด้วย

พจนานุกรมข้อมูล หมายถึง ส่วนที่มีหน้าที่อธิบายลักษณะของข้อมูลที่เก็บอยู่ในฐานข้อมูลรวมทั้งความสัมพันธ์ของข้อมูล

ระบบฐานข้อมูล (Database Management System หรือ DBMS) ถ้ามีหลายตัวจะมีการรวมพจนานุกรมข้อมูลเป็นส่วนหนึ่งของ DBMS ด้วย ซึ่งพจนานุกรมข้อมูลนี้จะเป็นองค์ประกอบทางซอฟต์แวร์ ทำหน้าที่เก็บรายละเอียดเกี่ยวกับข้อมูลภายในฐานข้อมูล เช่น โครงสร้างของแต่ละตาราง ใครเป็นผู้สร้าง สร้างเมื่อใด และแต่ละตารางประกอบด้วยเขตข้อมูลใดบ้าง คุณลักษณะของแต่ละเขตข้อมูลเป็นอย่างไร มีการเรียกใช้อยู่ในโปรแกรมประยุกต์ใดบ้าง และมีตารางใดที่มีความสัมพันธ์กันบ้าง มีเขตข้อมูลใดเป็นคีย์บ้าง เป็นต้น

พจนานุกรมข้อมูลยังมีส่วนที่ทำหน้าที่สำคัญดังต่อไปนี้

- ควบคุมระบบรักษาความปลอดภัยของข้อมูล และความคงสภาพของข้อมูล (Data Security and Data Integrity)
- ควบคุมเกี่ยวกับการใช้งานฐานข้อมูลพร้อมกัน (Concurrency Control)

ถ้าเปรียบเทียบฐานข้อมูลเหมือนกับห้องสมุดของโรงเรียนที่มีการเก็บหนังสือเล่มต่าง ๆ หนังสือเหล่านั้นจะเปรียบเสมือนกับข้อมูลที่เก็บอยู่ภายในฐานข้อมูล และในห้องสมุดจะต้องมีการทำบัญชีรายชื่อหนังสือต่างๆ เก็บไว้ เพื่อใช้บอกรายละเอียดเกี่ยวกับหนังสือแต่ละเล่มว่าใครเป็นผู้แต่งเก็บอยู่ที่ใดในห้องสมุด บัญชีรายชื่อหนังสือนี้จะเปรียบได้กับพจนานุกรมข้อมูล เช่น ระหว่างระเบียนของแต่ละแฟ้มข้อมูล จะมีพจนานุกรมข้อมูลซึ่งถูกเก็บและจะถูกเรียกใช้งานในระหว่างที่มีการประมวลผลฐานข้อมูลและข้อมูลที่มีอยู่ในฐานข้อมูลจะต้องมีความสัมพันธ์กันด้วย

3) ประโยชน์จากการประมวลผลด้วยฐานข้อมูล

การประมวลผลด้วยระบบฐานข้อมูล มีประโยชน์ดังนี้

1) ลดความซ้ำซ้อนของข้อมูลในกรณีที่มีข้อมูลชนิดเดียวกันถูกเก็บไว้หลายๆ แห่ง หรือที่เรียกว่า ความซ้ำซ้อนการนำข้อมูลทั้งหมดมาเก็บไว้ที่เดียวกันในฐานข้อมูล จึงถือว่าเป็นการ “ลด” ความซ้ำซ้อนลงไปได้ ทั้งนี้มิใช่หมายความว่าให้ขจัดข้อมูลออกไปเพื่อให้เหลือน้อยลง

2) สามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ในระดับหนึ่งเนื่องจากบางครั้งจะต้องมีการแก้ไขข้อมูลจึงอาจจะก่อให้เกิดปัญหาในการแก้ไขคือเมื่อเราแก้ไขข้อมูลที่เหมือนกันแต่แก้ไขไม่หมดหรือแก้ไขไม่ครบทุกข้อมูลที่มีอยู่ในแต่ละแห่งจึงทำให้ข้อมูลชุดเดียวกันอาจมีค่าในแต่ละแห่ง

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright © by Chiang Mai University
All rights reserved

ไม่ตรงกันดังนั้นถ้าการใช้ระบบฐานข้อมูลทำให้เราสามารถลดความซ้ำซ้อนลงไปได้ ซึ่งถ้าใช้ระบบฐานข้อมูลเมื่อเกิดการแก้ไขข้อมูลขึ้นเมื่อใดก็จะต้องแก้ให้เหมือนกันครบทุกแห่ง

3) สามารถใช้ข้อมูลร่วมกันได้การใช้ข้อมูลร่วมกันได้นี้ไม่ได้จำกัดอยู่เฉพาะโปรแกรมที่ใช้ข้อมูลอยู่ในปัจจุบันเท่านั้นแต่โปรแกรมประยุกต์ที่มีการพัฒนาขึ้นมาใหม่ ก็สามารถที่จะใช้ข้อมูลที่มีอยู่ได้เลยโดยไม่จำเป็นต้องเพิ่มเติมข้อมูลเข้าไปในระบบอีก

4) สามารถควบคุมความเป็นมาตรฐานได้ เมื่อมีการนำข้อมูลมาเก็บรวบรวมกันไว้ในฐานข้อมูลเช่นนี้ ทำให้ผู้ที่มีหน้าที่ควบคุมดูแลการใช้ระบบฐานข้อมูลสามารถกำหนดมาตรฐานของข้อมูลขึ้นมาได้ทำให้การบริหารหรือแลกเปลี่ยนข้อมูลระหว่างระบบเป็นไปอย่างสะดวกและถูกต้อง

5) สามารถจัดหาระบบความปลอดภัยที่รัดกุมได้ระบบฐานข้อมูลสามารถที่จะกำหนดสิทธิการใช้ให้แก่ผู้ใช้คนใดๆ ก็ได้ตามความเหมาะสมและผู้ใช้แต่ละคนก็อาจจะใช้ข้อมูลได้ในระดับที่ต่างกัน หรือพูดอีกนัยหนึ่งก็คือ ผู้ใช้แต่ละคนจะมองฐานข้อมูลด้วยวิวที่ต้องการจึงทำให้มีความปลอดภัยในการใช้ข้อมูลร่วมกัน

6) สามารถควบคุมความคงสภาพของข้อมูลได้โดยมีการกำหนดค่าต่างๆ หรือจำกัดช่วงของข้อมูลไว้เพื่อป้องกันการพิมพ์ข้อมูลผิดพลาดเช่นกำหนดช่วงของข้อมูลในการกรอกหมายเลขโทรศัพท์ไว้ 7 ตัว เมื่อพิมพ์ครบ 7 ตัวแล้ว ก็กำหนดให้เลื่อนไปข้อมูลถัดไป ฉะนั้น ถ้าพิมพ์ไม่ครบโปรแกรมก็จะไม่เลื่อนให้หรือพิมพ์เกิน 7 ตัวก็ไม่ได้

7) สามารถสร้างสมดุลในความขัดแย้งของความต้องการได้ การที่ผู้ใช้ทั้งหมดขององค์กรใช้ข้อมูลจากฐานข้อมูลร่วมกัน ทำให้ทราบถึงความต้องการและความสำคัญของผู้ใช้งานทั้งหมด จึงสามารถกำหนดโครงสร้างของฐานข้อมูลเพื่อให้บริการที่ดีที่สุดได้ เช่น เลือกเก็บข้อมูลที่จะต้องใช้บ่อยๆ ไว้ในสื่อข้อมูลที่มีความเร็วเป็นพิเศษ เป็นต้น เป็นการสร้างสมดุลของความต้องการไม่ให้เกิดความขัดแย้งในหมู่ผู้ใช้ เพราะการออกแบบนั้นกระทำบนแนวทางที่มุ่งจะให้ประโยชน์ส่วนรวมดีที่สุดแล้ว

8) เกิดความเป็นอิสระของข้อมูลโปรแกรมประยุกต์ที่เขียนขึ้นจะไม่ขึ้นกับโครงสร้างของตารางที่มีการเปลี่ยนแปลงไปเนื่องจากโครงสร้างของตารางต่างๆ และตัวข้อมูลในแต่ละตารางจะถูกเก็บอยู่ในฐานข้อมูลทั้งหมดโปรแกรมประยุกต์ไม่จำเป็นต้องเก็บโครงสร้างของตารางที่จะใช้ไว้ ซึ่งต่างกับระบบการประมวลผลแฟ้มข้อมูล ดังนั้น การเปลี่ยนแปลงโครงสร้างของตาราง เช่น การเปลี่ยนแปลงขนาดของเขตข้อมูลในตารางใดภายในฐานข้อมูล ก็ไม่จำเป็นต้องไปทำการแก้ไขโปรแกรมประยุกต์ที่มีการเรียกใช้เขตข้อมูลนั้น

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

4) ความสัมพันธ์และรูปแบบฐานข้อมูล

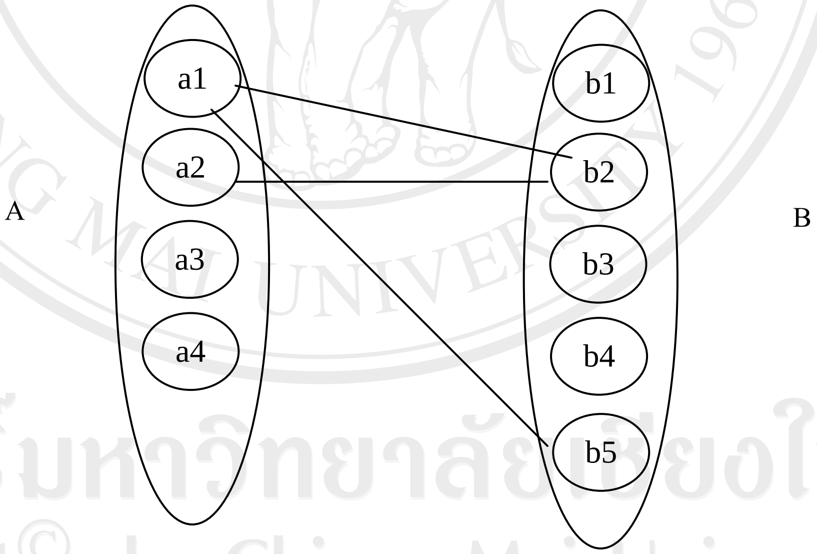
4.1 ความสัมพันธ์ (Relationship)

ฐานข้อมูลที่ใช้ในงานหนึ่ง ๆ นั้น ปกติมักจะประกอบด้วยแฟ้มข้อมูลหลายแฟ้ม แต่ละแฟ้มเก็บข้อมูลแตกต่างกันออกไป ฉะนั้นจึงต้องมีการกำหนดความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่ต้องการเก็บ ซึ่งความสัมพันธ์นี้อาจจำแนกได้เป็น 3 ชนิด คือ

- ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-One Relationship)
- ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-Many Relationship)
- ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-Many Relationship)

(1) ความสัมพันธ์แบบหนึ่งต่อหนึ่ง

เป็นลักษณะความสัมพันธ์ที่มีระเบียนเพียง 1 ระเบียนใน Entity A ที่มีความสัมพันธ์กับระเบียนเพียง 1 ระเบียนใน Entity B และในทางกลับกัน ระเบียนเพียง 1 ระเบียนใน Entity B จะมีความสัมพันธ์กับระเบียนเพียง 1 ระเบียนใน Entity A



รูป 2.6 แสดงความสัมพันธ์ในลักษณะหนึ่งต่อหนึ่ง

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

สมาชิก

รหัสสมาชิก	ชื่อสมาชิก	แผนก
1540	อรนุช	การขาย
1645	กิ่งแก้ว	เลขานุการ
1748	ภาวนา	การบัญชี
1985	สมนึก	คอมพิวเตอร์

การยืมหนังสือ

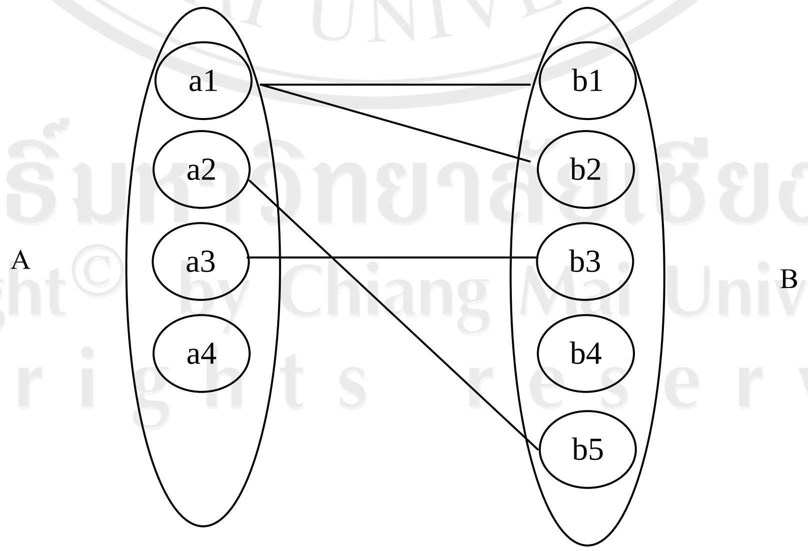
รหัสหนังสือ	รหัสสมาชิก
LB 435	1748
SC 593	1645
GM 185	1540
CO 556	2001
SC 602	1645

รูป 2.7 อธิบายความสัมพันธ์แบบหนึ่งต่อหนึ่งด้วยตารางความสัมพันธ์

จากตัวอย่าง ความสัมพันธ์ดังกล่าว หมายถึง สมาชิกแต่ละคน สามารถที่จะยืมหนังสือได้เพียงคนละ 1 เล่มเท่านั้น และหนังสือเล่มดังกล่าวก็ไม่สามารถให้ใครยืมได้อีก เพราะมีเพียงเล่มเดียว ในที่นี้จะสังเกตไว้ว่าทั้ง 2 Entity จะมีคีย์หรือข้อมูลหลักคือ รหัสสมาชิก

(2) ความสัมพันธ์แบบหนึ่งต่อกลุ่ม

เป็นลักษณะความสัมพันธ์ที่ระเบียนหนึ่งระเบียนใน Entity A มีความสัมพันธ์กับระเบียนหลายระเบียนใน Entity B และในทางกลับกันหลายระเบียนใน Entity B จะมีความสัมพันธ์กับระเบียนเพียงหนึ่งระเบียนใน Entity A



รูป 2.8 แสดงความสัมพันธ์ในลักษณะหนึ่งต่อกลุ่ม

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

สมาชิก

รหัสสมาชิก	ชื่อสมาชิก	แผนก
1540	อรนุช	การขาย
1645	กิ่งแก้ว	เลขานุการ
1748	ภาวนา	การบัญชี
1985	สมนึก	คอมพิวเตอร์

การยืมหนังสือ

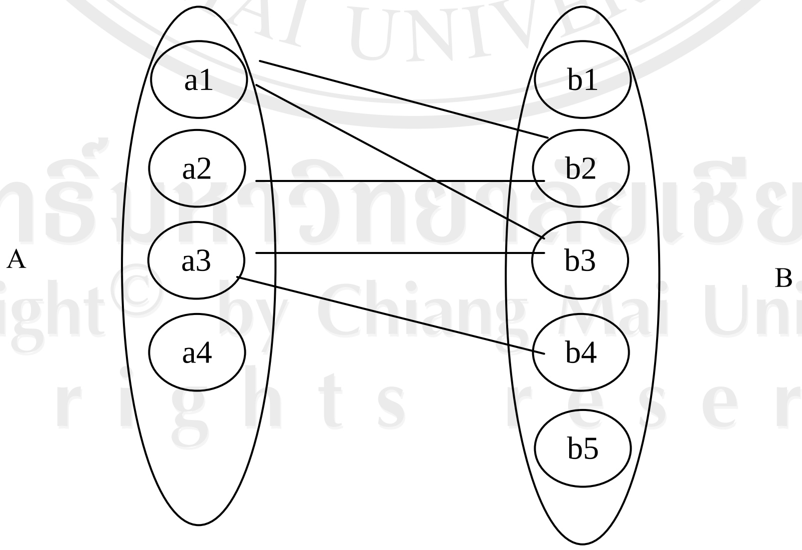
รหัสหนังสือ	รหัสสมาชิก
LB 435	1748
SC 593	1645
GM 185	1540
CO 556	2001
SC 602	1645

รูป 2.9 อธิบายความสัมพันธ์แบบหนึ่งต่อกลุ่มด้วยตารางความสัมพันธ์

จากตัวอย่าง สมาชิกที่มีรหัสสมาชิก 1540 สามารถยืมหนังสือได้ 2 เล่ม คือ รหัสหนังสือ LB 435 และ SC 593 แต่หนังสือทั้ง 2 เล่ม จะถูกยืมด้วยสมาชิกได้เพียงคนเดียว

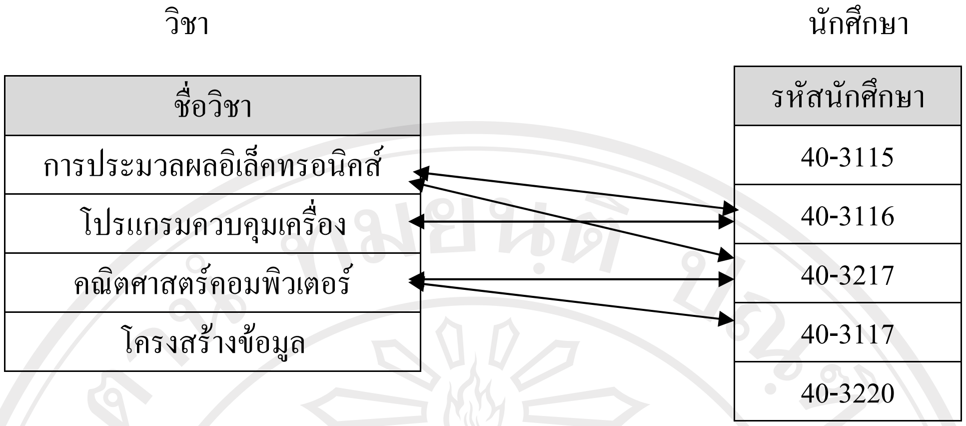
(3) ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

เป็นความสัมพันธ์ที่แต่ละระเบียนใน Entity A จะมีความสัมพันธ์กับระเบียนหลายระเบียนใน Entity B และในทางกลับกันแต่ละระเบียนใน Entity B จะมีความสัมพันธ์กับระเบียนที่อยู่ใน Entity A



รูป 2.10 แสดงความสัมพันธ์ในลักษณะกลุ่มต่อกลุ่ม

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



รูป 2.11 อธิบายความสัมพันธ์แบบกลุ่มต่อกลุ่มด้วยตารางความสัมพันธ์

จากตัวอย่าง ในแต่ละรายวิชา จะสามารถมีนักศึกษาเรียนได้มากกว่า 1 คน และในทางกลับกันนักศึกษาแต่ละคนก็สามารถเลือกเรียนวิชาได้มากกว่าหนึ่งรายวิชา

4.2 รูปแบบของฐานข้อมูล

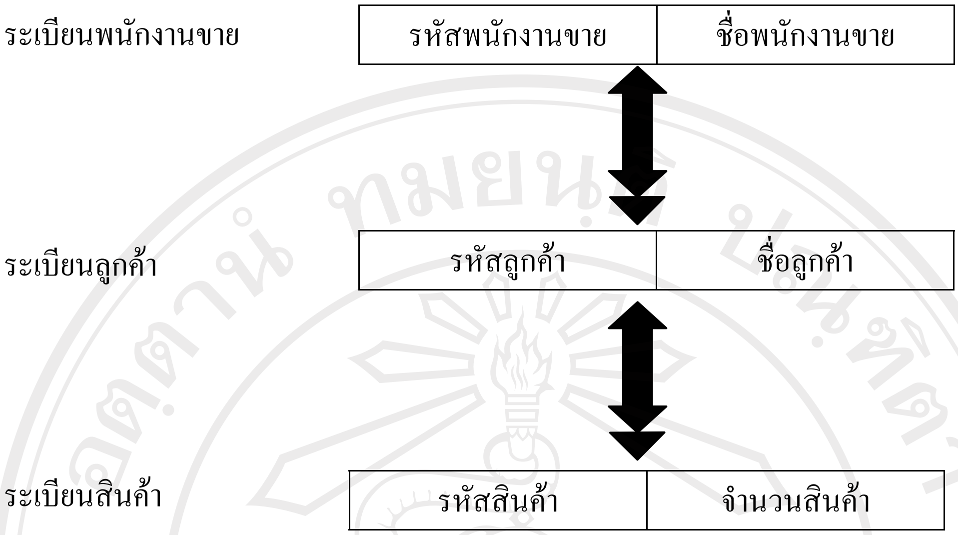
โครงสร้างของข้อมูลโดยทั่วไปจะมี 3 แบบด้วยกัน คือ

- ฐานข้อมูลแบบลำดับขั้น (Hierarchical Database)
- ฐานข้อมูลแบบเครือข่าย (Network Database)
- ฐานข้อมูลแบบเชิงสัมพันธ์ (Relational Database)

(1) ฐานข้อมูลแบบลำดับขั้น

ลักษณะของโครงสร้างของฐานข้อมูลแบบลำดับขั้นนี้ จะมีลักษณะคล้ายต้นไม้ที่คว่ำหัวลงจึงอาจเรียกโครงสร้างข้อมูลแบบนี้ได้อีกแบบว่าเป็น โครงสร้างแบบต้นไม้ (Tree Structure) โดยจะมีระเบียนที่อยู่แถวบนซึ่งจะเรียกว่าเป็น ระเบียนพ่อแม่ (Parent Record) ระเบียนในแถวถัดลงมาจะเรียกว่าระเบียนลูก (Child Record) ซึ่งระเบียนพ่อแม่จะสามารถมีระเบียนลูกได้มากกว่าหนึ่งระเบียน แต่ระเบียนลูกแต่ละระเบียนจะมีพ่อแม่เพียงหนึ่งระเบียนเท่านั้น

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



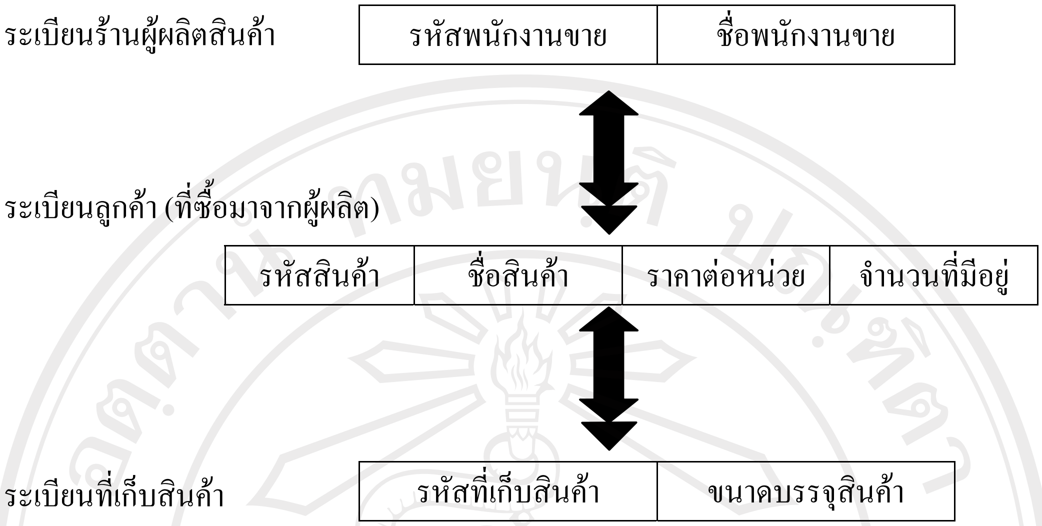
รูป 2.12 แสดงโครงสร้างฐานข้อมูลแบบลำดับขั้น

จากตัวอย่างข้างต้นจะเห็นว่า ลูกค้าแต่ละคนจะไม่สามารถได้รับบริการจากพนักงานขายมากกว่าหนึ่งคนได้ เนื่องจากลูกค้าแต่ละคนถือว่าเป็นระเบียนลูก และพนักงานขายจะถือว่าเป็นระเบียนพ่อแม่ของลูกค้า สินค้าแต่ละชนิดก็จะถูกซื้อโดยลูกค้าเพียงคนเดียวเท่านั้น เนื่องจากสินค้าแต่ละชนิด จะเป็นระเบียนลูกของระเบียนลูกค้า เป็นต้น

(2) ฐานข้อมูลแบบเครือข่าย

ข้อมูลภายในฐานข้อมูลแบบนี้สามารถมีความสัมพันธ์กันแบบใดก็ได้ เช่น อาจเป็นแบบหนึ่งต่อหนึ่ง หนึ่งต่อกลุ่ม ตัวอย่างของฐานข้อมูลแบบนี้ เช่น การสั่งซื้อสินค้าจากร้านผู้ผลิตสินค้า และการนำสินค้าไปเก็บในคลังสินค้า ซึ่งจะแสดงความสัมพันธ์ระหว่างระเบียนร้านผู้ผลิตสินค้าและระเบียนสินค้า และความสัมพันธ์ระหว่างระเบียนสินค้าและระเบียนที่เก็บสินค้าได้โดยการใช้ลูกศรเชื่อมโยงเช่นกัน ดังรูป 2.13

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



รูป 2.13 แสดงโครงสร้างฐานข้อมูลแบบเครือข่าย

(3) ฐานข้อมูลแบบเชิงสัมพันธ์

ฐานข้อมูลเชิงสัมพันธ์ จะมีโครงสร้างข้อมูลต่างจากฐานข้อมูลสองแบบแรก คือ ข้อมูลจะถูกเก็บอยู่ในรูปแบบของตาราง (Table) ซึ่งภายในตารางก็จะแบ่งออกเป็นแถว (Row) และคอลัมน์ (Column) แต่ละตารางจะมีจำนวนแถวได้หลายแถว และจำนวนคอลัมน์ได้หลายคอลัมน์ แถวแต่ละแถวจะสามารถเรียกได้อีกอย่างว่าระเบียนหรือเรคคอร์ด (Record) คอลัมน์แต่ละคอลัมน์สามารถเรียกได้อีกอย่างว่า เขตข้อมูลหรือฟิลด์ (Field)

2.2 แนวความคิดเกี่ยวกับสถาปัตยกรรมการพัฒนาระบบ SDLC

โอภาส เอี่ยมสิริวงศ์ (2546) ได้ให้ความหมายว่า เป็นวงจรที่แสดงถึงกิจกรรมต่างๆ ในแต่ละขั้นตอน ตั้งแต่ริเริ่มจนกระทั่งสำเร็จ วงจรการพัฒนาระบบนี้ จะทำให้เข้าใจถึงกิจกรรมพื้นฐานและรายละเอียดต่างๆ ในการพัฒนาระบบ โดยมีอยู่ 7 ขั้นตอนด้วยกัน

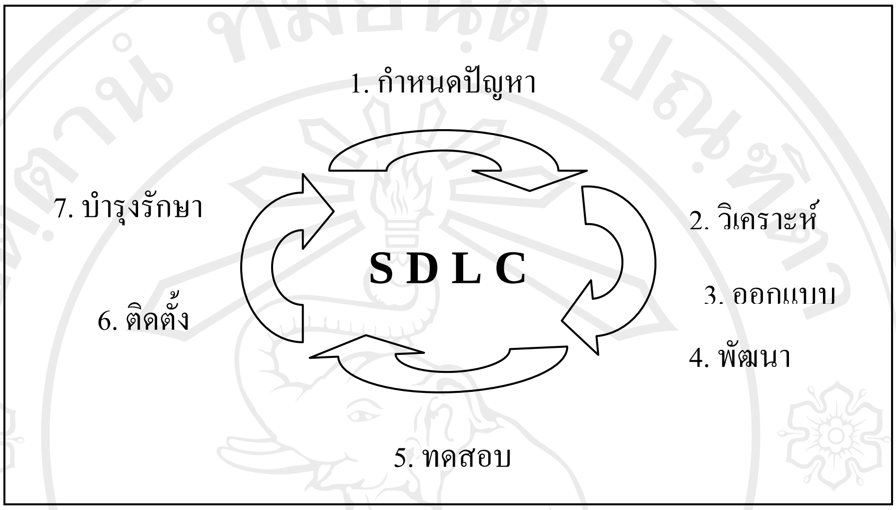
1) กำหนดปัญหา (Problem Definition)
2) วิเคราะห์ (Analysis)
3) ออกแบบ (Design)
4) พัฒนา (Development)

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

5) ทดสอบ (Testing)

6) ติดตั้ง (Implementation)

7) บำรุงรักษา (Maintenance)



รูปที่ 2.14 วงจรการพัฒนาระบบ

(1) การกำหนดปัญหา (Problem Definition)

เป็นขั้นตอนของการกำหนดขอบเขตของปัญหา สาเหตุของปัญหาจากการดำเนินงานในปัจจุบัน ความเป็นไปได้กับการสร้างระบบใหม่ การกำหนดความต้องการระหว่างนักวิเคราะห์ระบบ และผู้ใช้งาน โดยข้อมูลเหล่านี้ได้จากการสัมภาษณ์ การรวบรวมข้อมูลจากการดำเนินงานต่างๆ เพื่อทำการสรุปเป็นข้อกำหนด ที่ชัดเจน

(2) การวิเคราะห์ (Analysis)

การวิเคราะห์เป็นขั้นตอนของการวิเคราะห์การดำเนินงานของระบบปัจจุบัน โดยการนำ Requirements Specification ที่ได้มาจากขั้นตอนแรกมาวิเคราะห์รายละเอียด เพื่อทำการพัฒนาเป็นแบบจำลองลอจิคัล (Logical Model) ซึ่งประกอบด้วยแผนภาพ กระแสข้อมูล (Data flow Diagram) และแบบจำลองข้อมูล (ER–Diagram) ทำให้ทราบถึงรายละเอียดขั้นตอนการดำเนินงานในระบบว่าประกอบด้วยอะไรบ้าง มีความเกี่ยวข้องหรือมีความสัมพันธ์กับสิ่งใด

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

(3) ออกแบบ (Design)

การออกแบบเป็นขั้นตอนของการนำผลลัพธ์ที่ได้จากการวิเคราะห์ทาง Logical Model มาพัฒนาเป็น Physical Model ให้สอดคล้องกัน โดยการออกแบบจะเริ่มจากส่วนของอุปกรณ์และเทคโนโลยีต่าง ๆ และโปรแกรมคอมพิวเตอร์นำมาพัฒนา การออกแบบจำลองข้อมูล การออกแบบรายงาน และการออกแบบหน้าจอภาพการติดต่อกับผู้ใช้งาน (User Interface) การจัดทำพจนานุกรมข้อมูล (Data Dictionary)

(4) พัฒนา (Development)

การพัฒนาโปรแกรมเป็นขั้นตอนของการพัฒนาโปรแกรม ด้วยการสร้างชุดคำสั่งหรือเขียนโปรแกรมเพื่อการสร้างระบบงานโดยโปรแกรมที่ใช้ในการพัฒนาจะต้องพิจารณาถึงความเหมาะสมกับเทคโนโลยีที่ใช้งานอยู่ ซึ่งปัจจุบันมีการพัฒนาในรูปแบบของเชิงวัตถุ ซึ่งช่วยอำนวยความสะดวกต่อการพัฒนารวมทั้งการมีเครื่องมือต่างๆ ให้เลือกใช้มากมายตามความเหมาะสม

(5) ทดสอบ (Testing)

การทดสอบระบบ เป็นขั้นตอนของการทดสอบระบบก่อนที่จะนำไปปฏิบัติใช้งานจริง ทีมงานจะทำการทดสอบข้อมูลเบื้องต้นก่อน ด้วยการสร้างข้อมูลจำลองเพื่อตรวจสอบการทำงานของระบบ หากมีข้อผิดพลาดเกิดขึ้นก็จะต้องย้อนกลับไปทำขั้นตอนของการพัฒนาโปรแกรมใหม่ โดยการทดสอบนี้ จะมีการทดสอบแบบ 2 อย่างคือ ตรวจสอบด้านภาษาเขียน และการตรวจสอบวัตถุประสงค์ตรงกับความต้องการหรือไม่

(6) ติดตั้ง (Implementation)

ขั้นตอนต่อมาหลังจากที่ได้ทำการทดสอบ จนมีความมั่นใจแล้วระบบสามารถทำงานได้จริงและตรงกับความต้องการของผู้ใช้ระบบ จากนั้นจึงดำเนินการติดตั้งระบบ

(7) บำรุงรักษา (Maintenance)

เป็นขั้นตอนของการปรับปรุงแก้ไขระบบหลังจากที่ได้มีการติดตั้งและใช้งานแล้ว ในขั้นตอนนี้อาจเกิดจากปัญหาของโปรแกรม ซึ่งโปรแกรมเมอร์จะต้องรีบแก้ไขข้อมูลให้ถูกต้อง หรือเกิดจากความต้องการของผู้ใช้งานที่ต้องการเพิ่มโมดูลในการทำงานอื่น ๆ ซึ่งทั้งนี้ทั้งนั้นก็จะเกี่ยวข้องกับ Requirements Specification ที่เคยตกลงกันไว้ก่อนหน้าด้วย ดังนั้น ในส่วนงานนี้จะคิดค่าใช้จ่ายเพิ่มหรืออย่างไร เป็นเรื่องของรายละเอียดที่ผู้พัฒนาหรือนักวิเคราะห์ระบบจะต้องดำเนินการกับผู้ว่าจ้างต่อไป

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

2.3 แนวความคิดเกี่ยวกับสถาปัตยกรรม .Net Framework และ Visual Basic .NET

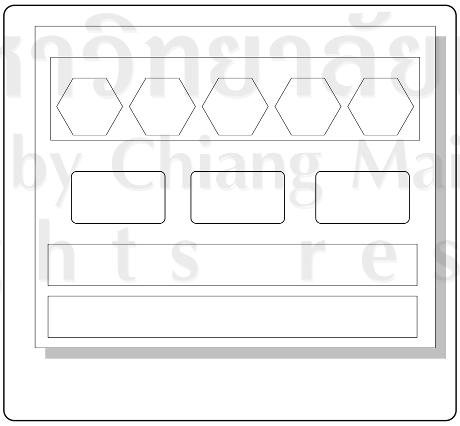
สัจจะ จรัสรุ่งรวีวร (2545) ได้ให้ความหมายของ Visual Basic ว่าเป็นเครื่องมือพัฒนาโปรแกรมสำหรับระบบปฏิบัติการ Windows ที่ได้รับความนิยมอย่างแพร่หลาย โดยเฉพาะอย่างยิ่งในเมืองไทยแล้วอาจกล่าวได้ว่าได้รับความนิยมสูงสุด ตั้งแต่นักเรียนนักศึกษาจนกระทั่งนักพัฒนาซอฟต์แวร์ระดับมืออาชีพ เนื่องจากการเรียนรู้และใช้งาน VB ทำได้ง่าย สามารถใช้ในการพัฒนาโปรแกรมระดับเบื้องต้นไปจนกระทั่งโปรแกรมที่มีความสลับซับซ้อน นอกจาก VB ยังได้รับการพัฒนาเพิ่มเติมขีดความสามารถมาโดยตลอด จนกระทั่งเวอร์ชั่นล่าสุดคือ Visual Basic .NET หรือ VB .NET ถือได้ว่าเป็นการเปลี่ยนแปลงครั้งสำคัญของ VB ถึงขึ้นยกเครื่องภาษา VB เลยก็ว่าได้ VB .NET ได้ถูกพัฒนาให้สอดคล้องและรับเอาความสามารถต่างๆ ที่อยู่ภายใต้เทคโนโลยี .NET ซึ่งเป็นวัตรรมการพัฒนาแอพลิเคชั่นยุคใหม่ที่ไมโครซอฟต์ชูโรงอยู่ในขณะนี้

1) Visual Basic .NET เป็นเครื่องมือที่ใช้ในการพัฒนาโปรแกรมแบบ Visual Programming บนระบบปฏิบัติการ Windows ซึ่งได้รับการพัฒนามาจากภาษา BASIC (Beginners All Purpose Symbolic Instruction Code) ซึ่งเป็นภาษาโปรแกรมที่ได้รับความนิยมอย่างแพร่หลายสำหรับผู้เริ่มต้นหัดเขียนโปรแกรมคอมพิวเตอร์ เนื่องจาก Basic เป็นภาษาโปรแกรมที่สามารถทำความเข้าใจได้ง่าย

หนึ่งสิ่งที่สำคัญที่ทำให้ VB .NET ต่างจาก VB เวอร์ชั่นอื่นๆ ก็คือ การเป็นภาษาโปรแกรมแบบออบเจ็คโอเรียนเต็ด (Object-Oriented Programming หรือ OOP) เต็มตัว เช่นเดียวกับภาษาสมัยใหม่อื่นๆ ที่ได้รับความนิยมอย่างกว้างขวาง เช่น C++, C#, Delphi และ JAVA เป็นต้น

2) นวัตกรรม .Net Framework

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved



รูปที่ 2.15 สถาปัตยกรรม .NET Framework

.NET Framework ของไมโครซอฟต์เป็นแพลตฟอร์มแบบใหม่สำหรับสร้างแอพพลิเคชั่นชุด และแอพพลิเคชันบริการเพื่อสนองความต้องการของธุรกิจอินเตอร์เน็ต .NET เฟรมเวิร์กเป็นแอพพลิเคชันที่ทำหน้าที่รวบรวมข่าวสาร และการทำงานร่วมกันระหว่างซอร์สที่หลากหลายโดยไม่คำนึงถึงแพลตฟอร์มหรือภาษาที่ใช้

การบริการต่างๆ ดังเช่นที่กล่าวมานี้ยังไม่มีอยู่เนื่องจากว่ายังไม่มีมาตรฐานสำหรับการนำข้อมูลข่าวสารทั้งหลายมาประมวลผล ผู้ขายแต่ละรายจะมีวิธีการของตนเองในการอธิบายสินค้าต่างๆ ที่ขายอยู่นอกจากนั้นยังเป็นเรื่องยากลำบากที่จะพัฒนาโค้ดที่จำเป็นเพื่อการให้บริการเช่นนี้

ฉะนั้น จึงไม่ประหลาดใจที่ทางบริษัทไมโครซอฟต์ตระหนักถึงความต้องการในการให้บริการต่างๆ ไมโครซอฟต์จึงได้นำเอามาตรฐาน XML มาใช้สำหรับอธิบายข้อมูลและจัดหาโครงสร้างพื้นฐานของแพลตฟอร์มการพัฒนาแบบใหม่ซึ่งช่วยให้นักพัฒนาสามารถสร้างและใช้ประโยชน์แอพพลิเคชันที่แจกจ่ายออกไปได้ในยุคของอินเตอร์เน็ต

แพลตฟอร์มนี้เรียกว่า Microsoft .NET Framework หรือเรียกสั้นๆ ว่า .NET โดย .NET จะอนุญาตให้นักพัฒนาใช้ความก้าวหน้าทางเทคโนโลยีที่ดีกว่าแพลตฟอร์มก่อนๆ ของไมโครซอฟต์และความสามารถในการนำโค้ดมาใช้อีก ความพิเศษของโค้ด การจัดการทรัพยากร การพัฒนาหลายภาษา การรักษาความปลอดภัย การใช้ประโยชน์อย่างได้ผลและการบริหาร ในขณะที่ออกแบบแพลตฟอร์มใหม่นี้ ไมโครซอฟท์ยังได้ปรับปรุงความสามารถบางอย่างของแพลตฟอร์มวินโดวส์ปัจจุบันด้วย

อย่างที่ได้กล่าวไปแล้วในตอนต้นว่าไมโครซอฟท์ต้องการที่จะสร้างอะไรที่เป็นมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่างสามารถติดต่อสื่อสารกันได้หมด โดยคิดค้นระบบซึ่งหมายมั่นปั้นมือว่าจะให้เป็นระบบมาตรฐาน ระบบนี้คือ .NET Framework ซึ่งระบบนี้ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือนโปรแกรมหนึ่งที่จะสามารถสร้างสภาวะแวดล้อมหนึ่ง ซึ่งสามารถทำงานในระบบ .NET นี้ได้

ในอนาคตไมโครซอฟท์ก็หวังที่จะนำเอาระบบนี้ไปติดตั้งลงบนอุปกรณ์ทุกชนิด เพื่อทำให้อุปกรณ์ทุกอย่างมีระบบๆหนึ่งที่เหมือนกันหมด โดย .NET Framework นั้นมี ส่วนประกอบภายในแบ่งออกเป็น 3 ชั้นใหญ่ๆ คือ

(1) Programming Language : เป็นรูปแบบของภาษาที่ออกแบบมาเพื่อให้สามารถทำงานในสภาวะที่เป็น .NET ได้โดยที่ทางไมโครซอฟท์ได้เปิดตัวภาษาหลักๆที่จะใช้พัฒนาบน .NET นี้ 3 ภาษา

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright © by Chiang Mai University
All rights reserved

- C# เป็นภาษาใหม่ที่ไมโครซอฟท์พัฒนามาจาก C++ กับ JAVA เป็นหลัก
- VB.NET เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชั่น 6.0
- JScript.net เป็นภาษาที่พัฒนามาจาก JScript ซึ่งเป็น JavaScript ในเวอร์ชั่นของไมโครซอฟท์

(2) Base Classes Library : Library นั้นเปรียบเสมือนชุดคำสั่งสำเร็จรูปย่อยๆที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้น จึงมีผู้คิดค้นเพื่ออำนวยความสะดวกในการเขียนโปรแกรม ซึ่ง Library ในภาษาต่างๆ ส่วนใหญ่จะอยู่ในรูปแบบไฟล์ incould แต่ถ้าเป็น ASP สิ่งที่เป็น library ก็คือ คอมโพเนนท์ต่างๆ นั่นเอง ซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น Library พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้ Library ที่เป็นตัวเดียวกันได้หมด

(3) Common Language Runtime (CLR) : นับเป็นสิ่งสำคัญแทบจะที่สุดของระบบ .NET นี้ก็ว่าได้ เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆกันกลายเป็นภาษารูปแบบมาตรฐานเดียวกัน ทั้งหมด เราเรียกภาษาที่ว่านี้ว่า Intermediate Language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้จะตรวจสอบเครื่องที่รันว่ามีสภาวะแวดล้อมการทำงานเช่นใดหลังจากนั้นก็จะคอมไพล์เป็นโปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้เราสามารถใช้งานโปรแกรม ต่างๆได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่อง

3) ประโยชน์และข้อดีของ .NET Framework

o เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน เนื่องจากมีไลบรารีที่เป็นมาตรฐานเดียวกันทั้งหมดทำให้เราไม่ต้องกังวล ว่าภาษาที่ใช้เขียนนั้นมีไลบรารีตัวนั้นตัวนี้หรือไม่ รวมทั้งไม่ต้องคอยกังวลว่าถ้าใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะไม่มีไลบรารีตัวนั้น

o ไม่ขึ้นกับระบบปฏิบัติการ (OS) เนื่องจากระบบปฏิบัติการ ที่แต่ละบุคคลหรือองค์กรใช้นั้นย่อมไม่เหมือนกัน แต่ภายใน .NET Framework จะไม่มีปัญหานี้ของเพียงแค่มีระบบ .NET Framework ก็จะทำให้สามารถใช้งานโปรแกรมต่างๆได้ ซึ่งเป็นข้อดีตรงที่เราจะสามารถใช้โปรแกรมต่างๆ ได้ทุกระบบปฏิบัติการ

o ใช้ในการพัฒนาได้ทุกภาษา ทำให้เราไม่ต้องคอยมาศึกษาภาษาใหม่ๆ เมื่อต้องการสร้างโปรแกรมในแต่ละครั้ง นอกจากนั้น เรายังสามารถเลือกใช้ภาษาที่เราถนัดที่สุดในการพัฒนาโปรแกรมต่างๆได้ด้วย

o มีการควบคุมสิ่งแวดล้อมในการทำงานเป็นอย่างดี เนื่องจากเป็นระบบที่เป็นมาตรฐานทำให้การควบคุมจัดสรรระบบต่างๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำด้านการใช้งานเครื่องก็มีความรวดเร็วมากขึ้น ลดโอกาสที่เครื่องจะค้างได้เป็นอย่างดี

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

o ความปลอดภัยที่มีมากขึ้น .NET Framework สามรถกำหนดสิทธิ์การใช้งานหรือ Permission ของผู้ใช้งานได้มากขึ้นทำให้สมารถกำหนดว่า จะให้โปรแกรมในส่วนใดใช้งานได้หรือไม่ได้ แล้วแต่เฉพาะบุคคล

ทั้งหมดนี้เป็นเพียงประโยชน์ส่วนหนึ่งในแนวคิดของไมโครซอฟท์ที่กำลังจะพัฒนาให้สำเร็จเท่านั้น บางข้อทำได้สำเร็จแล้ว แต่บางข้อก็ยังไม่สำเร็จดังนั้นจึงต้องคอยรอดูว่า ฝันของไมโครซอฟท์จะเป็นจริงและสำเร็จได้เมื่อไร

4) Object-Oriented Programming (OOP)

(1) ออบเจ็ค (Object)

ข้อมูลรูปแบบหนึ่งซึ่งประกอบไปด้วย ข้อมูล และคำสั่ง ซึ่งใช้จัดการกับข้อมูลที่มีอยู่ โดยที่คำสั่ง และข้อมูลจะถูกเก็บแยกเป็นส่วนออกกันต่างหาก แต่ยังรวมอยู่ในตัวหลักตัวเดียวกัน ให้ลองนึกถึงเวลาที่ซื้อเครื่องคอมพิวเตอร์มาเครื่องหนึ่ง ซึ่งประกอบไปด้วยอะไหล่ และ คู่มือในการประกอบเครื่อง โดยที่อะไหล่ก็คือ ข้อมูล และ คู่มือในการประกอบก็คือ ชุดของคำสั่งรวมดังนั้นเมื่อต้องการที่จะประกอบเครื่องคอมพิวเตอร์เข้าด้วยกันก็ต้องดูคู่มือ (เรียกใช้คำสั่ง) เพื่อสามารถนำอะไหล่มาประกอบได้ถูกต้อง เมื่อประกอบมันเข้าด้วยกันเสร็จด้วยการอาศัยคู่มือ และ อะไหล่ ก็จะได้ Object ซึ่งก็คือ เครื่องคอมพิวเตอร์นั่นเอง จะเข้าใจ Object ได้ดีถ้าเข้าใจคำว่า Record โดยทั่วไปแล้วใน Pascal จะหมายถึง พื้นที่ที่ใช้สำหรับเก็บข้อมูลซึ่งจะแสดงลักษณะเฉพาะของแต่ละ Record จากข้อมูลที่มันเก็บอยู่ ซึ่ง Object ก็จะคล้ายกับ Record คือ มีส่วนเก็บข้อมูล แต่จะมีพื้นที่ส่วนหนึ่งเพิ่มขึ้นมาคือ Procedure และ Function ที่ใช้สำหรับจัดการกับ Record นั้น

(2) คลาส (Class)

ชนิดข้อมูลที่ประกอบไปด้วย Field Method และ Properties ซึ่งตัวอย่างของ Class ก็ได้แก่ Object ส่วน Filed, Method และ Properties จะเรียกว่า Members หรือ สมาชิกของ Object

- Field เป็นตัวแปรที่สำคัญของ Object ซึ่งมีความหมายคล้ายกับ Field ของ Record นั่นเอง แต่จะแสดงข้อมูลของแต่ละ Item ที่อยู่ภายใน Class
- Method เป็นการเรียกรวมกันของ Procedure และ Function ซึ่งมีหน้าที่หลักคือจัดการกับ Object ที่ได้ทำการสืบทอดสมบัติจาก Class และ Method บางตัว สามารถจัดการกับตัว Class เองได้ด้วย (เรียกว่า Class Method)
- Properties คือ ส่วนที่จัดการกับข้อมูลของ Object (ซึ่งปรกติจะเก็บไว้ใน field) ซึ่งตัว Properties จะเป็นตัวกำหนดว่าควรจะอ่านหรือ แก้ไขค่าของข้อมูลอย่างไร ให้กับส่วนอื่นๆ ของ Program หรือ Object อื่น ที่สืบทอดมาจาก Class ตัวนี้จะเรียกใช้ได้จากค่าที่มีอยู่ใน Properties เท่านั้น

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

- Objects จะทำการจองความจำแบบ Dynamic ซึ่งโครงสร้างข้อมูล (Structure) จะนำมาจาก Class ที่เป็นต้นแบบของมัน Objectทุกตัวที่สืบทอดมาจาก Class ต้นแบบตัวเดียวกันจะทำการคัดลอกทุกๆ Field ที่มีอยู่ใน Class มาทั้งหมด (โดยการจองพื้นที่หน่วยความจำใหม่) แต่ จะใช้ Method ร่วมกัน (คือไม่มีการจองความจำสำหรับ Method แต่จะเรียก Method จากตัว Class ต้นแบบเลย) แต่ Object ทุกตัวจะต้องเรียกใช้ Method พิเศษอยู่ 2 ตัวที่ใช้สำหรับสร้าง และทำลายตัว Object เอง คือ Constructors และ Destructors

จะเห็นได้ว่า Object จะมีความสมบูรณ์ในตัวเอง ซึ่งถูกสร้างขึ้นโดยอาศัยคลาสหรือต้นแบบ และเราสามารถนำเอาต้นแบบนั้นไปดัดแปลงแก้ไขหรือพัฒนาเพิ่มเติมต่อยอดให้กลายเป็นต้นแบบใหม่ได้ ทั้งหมดนี้เป็นการจำลองสิ่งที่มีอยู่ตามธรรมชาติให้อยู่ในรูปของซอฟต์แวร์ ซึ่งเป็นที่มาของแนวคิดในการเขียนโปรแกรมแบบ Object-Oriented Programming หรือ OOP นั่นเอง

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved