

บทที่ 2

แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

การวิจัยเรื่อง การพัฒนาระบบช่วยสืบค้นรหัสโรค หัตถการและการผ่าตัดผ่านอินเทอร์เน็ต ผู้วิจัยได้ค้นคว้าจากเอกสาร ตำราที่เป็นทฤษฎี หลักการ แนวความคิดที่เกี่ยวข้อง เพื่อเป็นกรอบในการวิเคราะห์ วิจัย และสนับสนุนการวิจัย ดังนี้

1. ระบบรหัสโรค ICD-10 และรหัสหัตถการ ICD-9-CM
2. ระบบฐานข้อมูล (database system) และระบบการจัดการฐานข้อมูล (Data Base Management System--DBMS)
3. การวิเคราะห์และออกแบบระบบ (system analysis and design)
4. การออกแบบฐานข้อมูลโดยวิธีแบบจำลองข้อมูล (Entity-Relationship Model--E-R Model)
5. ภาษา UML (Unified Modeling Language--UML)
6. การพัฒนาโปรแกรมด้วยภาษา PHP ร่วมกับ Asynchronous JavaScript and XML (AJAX)
7. ภาษา Java และภาษา Java script
8. การเขียนโปรแกรมด้วย NetBeans IDE และ Compiler Java
9. การใช้ Library MySQL-Connector-Java เชื่อมต่อระหว่างฐานข้อมูล (database) กับ NetBeans

ระบบรหัสโรค ICD-10 และรหัสหัตถการ ICD-9-CM

ระบบรหัสโรค ICD-10

ICD-10 ย่อมาจาก International Classification of Diseases and Related Health Problem 10th Revision ศัพท์ภาษาไทยโดยกระทรวงสาธารณสุขเรียกว่า บัญชีจำแนก

โรคระหว่างประเทศ ฉบับแก้ไขครั้งที่ 10 เป็นระบบที่มีองค์ประกอบสำคัญ 2 ส่วนดังนี้ (มหาวิทยาลัยธรรมศาสตร์, คณะแพทยศาสตร์, 2548, หน้า 3-4)

1. ระบบการจัดหมวดหมู่ของโรคและปัญหาสุขภาพต่าง ๆ ที่พบในมนุษย์
2. ระบบรหัสโรคและรหัสปัญหาสุขภาพ

ระบบการจัดหมวดหมู่ของโรคใช้หลักการของศาสตร์แห่งการจัดหมวดหมู่โรค (Nosology) โดยการจับกลุ่มโรคที่มีลักษณะใกล้เคียงกัน มาอยู่ในหมวดหมู่เดียวกัน ส่วนระบบรหัสโรคและรหัสปัญหาสุขภาพ ใช้การกำหนดรหัสเป็นสัญลักษณ์แทนโรคหรือปัญหาทางสุขภาพ

ลักษณะรหัสของ ICD-10

ICD-10 เป็นรหัสตัวอักษรผสมตัวเลข (alphanumeric code) โดยรหัสแต่ละตัวจะขึ้นต้นด้วยตัวอักษรภาษาอังกฤษ A-Z แล้วตามด้วยตัวเลขอารบิก 0-9 อีก 2 ถึง 4 ตัว จึงเป็นรหัสที่มีความยาว 3, 4 หรือ 5 ตำแหน่ง ตัวอย่างรหัส ICD-10 ได้แก่ (มหาวิทยาลัยธรรมศาสตร์, คณะแพทยศาสตร์, 2548, หน้า 4)

1. I10 เป็นรหัสแทนโรค Hypertension
2. J18.9 เป็นรหัสแทนโรค Pneumonia
3. M00.91 เป็นรหัสแทนโรค Pyogenic Arthritis at Shoulder

จะเห็นได้ว่า ในรหัส ICD-10 ที่มีความยาว 4 หรือ 5 ตำแหน่ง จะมีเครื่องหมายจุด (.) คั่นกลางระหว่างรหัสตำแหน่งที่ 3 กับรหัสตำแหน่งที่ 4 เพื่อให้ให้อ่านได้ง่ายขึ้น

ในอดีต ICD-10 ไม่มีการใช้รหัสที่ขึ้นต้นด้วยตัวอักษร U แต่เมื่อปี พ.ศ. 2550 กระทรวงสาธารณสุขได้มีการเพิ่มรหัสตัว U (Chapter XXII) ซึ่งใช้แทนรหัสโรคที่เกิดขึ้นใหม่ในช่วงนั้น เช่น U04.9 เป็นรหัสแทนโรค Severe Acute Respiratory Syndrome (SARS)

ประเภทการจัดกลุ่มของรหัสโรค

การจัดกลุ่มโรคที่พบในมนุษย์มีหลายแบบ ได้แก่ การจัดกลุ่มโรคตามอวัยวะที่เป็นโรค การจัดกลุ่มโรคโดยแบ่งลักษณะของโรคว่าเป็นประเภทโรคติดต่อ หรือเป็น

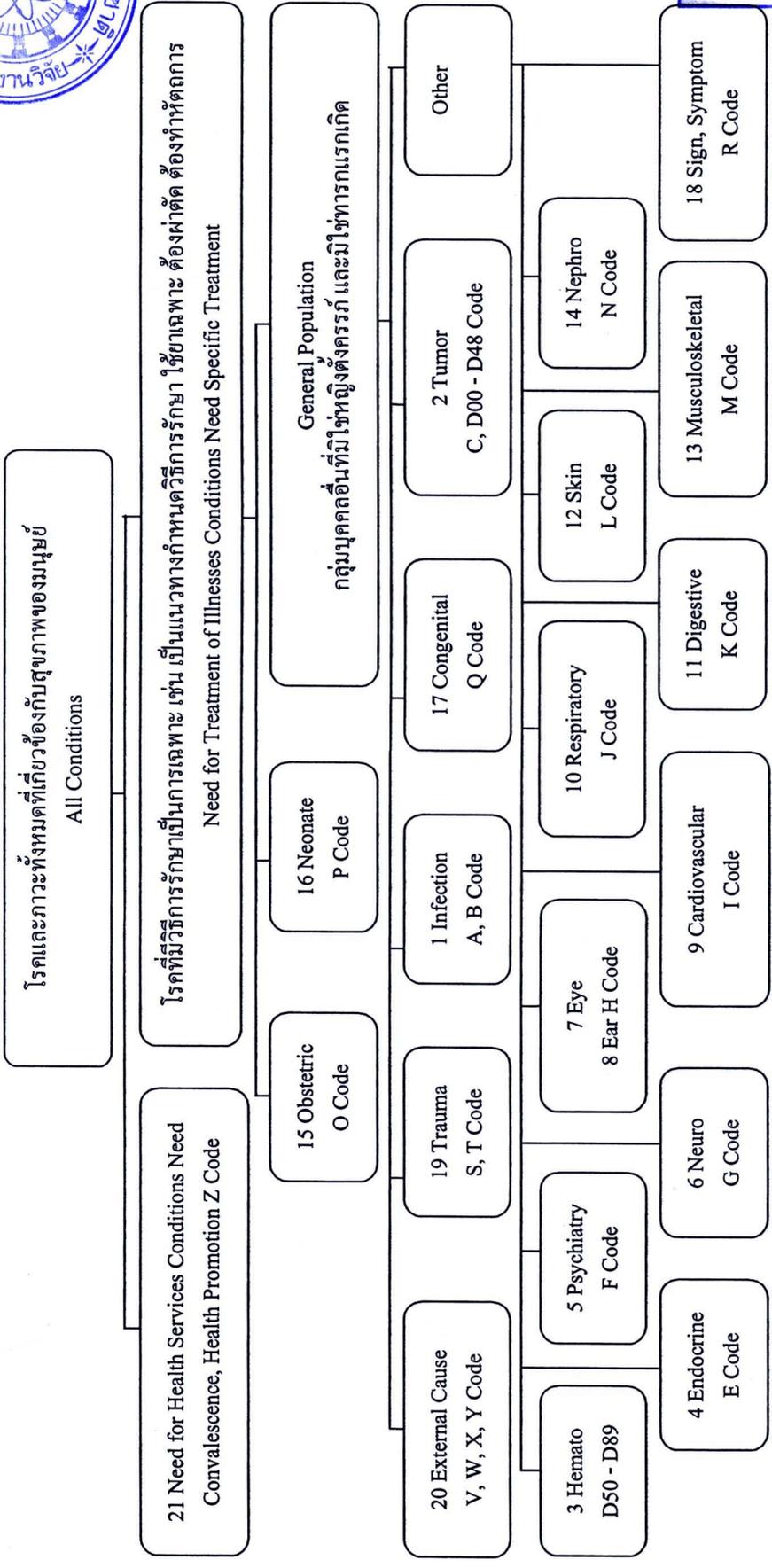
โรคที่ไม่ติดต่อกัน การจัดกลุ่มตามสาเหตุของโรค เป็นต้น ระบบการจัดกลุ่มโรคใน ICD-10 มีลักษณะพิเศษเฉพาะตัว คือ มีการจัดกลุ่มโรคหลายระบบรวมอยู่ในรหัสชุดเดียวกัน การจัดกลุ่มโรคแบ่งออกเป็น 4 กลุ่ม ดังนี้ (มหาวิทยาลัยธรรมศาสตร์, คณะแพทยศาสตร์, 2548, หน้า 7)

1. ระบบการจัดกลุ่มโรคตามลักษณะความต้องการใช้บริการ
 - 1.1 ภาวะที่ต้องการบริการป้องกันหรือส่งเสริมสุขภาพ
 - 1.2 ภาวะที่ต้องการบริการรักษาโรค
2. ระบบการจัดกลุ่มโรคตามลักษณะผู้ป่วย
 - 2.1 หญิงตั้งครรภ์
 - 2.2 ทารกแรกเกิด
 - 2.3 ประชาชนกลุ่มอื่น ๆ
3. ระบบการจัดกลุ่มโรคตามสาเหตุที่ทำให้เกิดโรค
 - 3.1 บาดเจ็บ-เหตุที่ทำให้เกิด
 - 3.2 บาดเจ็บ-ตำแหน่ง
 - 3.3 พิกการแต่กำเนิด
 - 3.4 โรคติดเชื้อ
 - 3.5 เนื้องอก มะเร็ง
 - 3.6 อื่น ๆ
4. ระบบการจัดกลุ่มโรคตามตำแหน่งอวัยวะหรือระบบที่เป็นโรค
 - 4.1 โรคอื่น ๆ อันมิได้มีสาเหตุมาจากการบาดเจ็บ การติดเชื้อ พิกการแต่กำเนิด เนื้องอก มะเร็ง ได้แก่
 - 4.1.1 โรคเลือดและระบบภูมิคุ้มกัน
 - 4.1.2 โรคต่อมไร้ท่อ
 - 4.1.3 โรคจิต โรคประสาท
 - 4.1.4 โรคสมอง ไขสันหลัง เส้นประสาท
 - 4.1.5 โรคตา
 - 4.1.6 โรคหู

- 4.1.7 โรคหัวใจและหลอดเลือด
 - 4.1.8 โรคระบบหายใจ
 - 4.1.9 โรคระบบย่อยอาหาร
 - 4.1.10 โรคผิวหนังและเนื้อเยื่อ
 - 4.1.11 โรคกล้ามเนื้อและกระดูก
 - 4.1.12 โรคระบบประสาท ระบบสืบพันธุ์
- 4.2 ภาวะอื่น ๆ ที่มีสามารถจัดเข้ากลุ่มโรคใด ๆ ได้หรือวินิจฉัยโรคไม่ได้



สำนักงานคณะกรรมการวิจัยแห่งชาติ
 ห้องสมุดงานวิจัย
 วันที่: ๑๙ ก.ย. ๒๕๕๕
 เลขทะเบียน: ๒๔๙๖๑๐
 เลขเรียกหนังสือ:



ภาพ 1 หมวดของรหัสโรคในระบบ ICD-10

ที่มา. จาก คู่มือการให้รหัสโรคและหัตถการขั้นต้น: บทเรียนอิเล็กทรอนิกส์การให้รหัสโรคและหัตถการ (หน้า 16), โดย มหาวิทยาลัย-ธรรมศาสตร์, คณะแพทยศาสตร์, 2548, กรุงเทพมหานคร: ผู้แต่ง.

ลักษณะรหัสของ ICD-9-CM

จากการศึกษาของมหาวิทยาลัยธรรมศาสตร์, คณะแพทยศาสตร์ (2548, หน้า 17) พบว่า รหัส ICD-9-CM ย่อมาจาก International Statistical Classification of Disease Ninth Revision Clinical Modification เป็นระบบรหัสหัตถการและการผ่าตัดของประเทศสหรัฐอเมริกา สร้างขึ้นในปี ค.ศ. 1979 ภายใต้การควบคุมดูแลของ Health Care Financial Administration (HCFA) และ National Committee on Vital and Health Statistics (NCVHS) โดยหนังสือเล่มที่ 3 ของ ICD-9-CM เป็นเล่มที่มีรายการรหัสหัตถการและการผ่าตัดรวมอยู่กับครรชนี

รหัส ICD-9-CM เป็นรหัสที่ใช้สำหรับเก็บข้อมูลในการทำหัตถการและการผ่าตัดกรณีผู้ป่วยในของโรงพยาบาลต่าง ๆ เป็นรหัสที่ใช้คำนวณและจัดกลุ่ม Diagnosis Related Group (DRG) ของ HCFA จึงต้องมีการปรับปรุงให้ทันสมัยอยู่เสมอ ภายหลังจากได้มีการพิมพ์ ICD-9-CM รุ่นใหม่ ๆ ออกมาจำหน่ายทุกปี โดยล่าสุดที่ใช้ คือ ICD-9-CM ฉบับปี ค.ศ. 2000

รหัส ICD-9-CM นั้น ได้มีการพัฒนาอย่างต่อเนื่อง โดยการเพิ่มรหัส การตัดรหัส การเพิ่ม Include และ Exclude และการลบ Include และ Exclude ซึ่งในขณะนี้ได้พัฒนาและใช้งานจนถึงฉบับปี ค.ศ. 2010

รหัส ICD-9-CM เป็นรหัสตัวเลข มีความยาว 3-4 ช่องตัวอักษร โดยใช้เครื่องหมายจุด (.) คั่นระหว่างรหัสตัวเลขตำแหน่งที่ 2 กับตำแหน่งที่ 3

ตัวอย่างรหัส ICD-9-CM ได้แก่ (1) 01.31 Incision of Cerebral Meninges (2) 37.31 Pericardiectomy (3) 47.09 Other Appendectomy และ (4) 76.76 Open Reduction of Mandibular Fracture

ระบบฐานข้อมูลและระบบการจัดการฐานข้อมูล

ความหมายของระบบฐานข้อมูล

โอภาส เอี่ยมสิริวงศ์ (2546, หน้า 28) กล่าวว่า ระบบฐานข้อมูล (database system) เป็นแหล่งหรือศูนย์รวมข้อมูลที่มีความสัมพันธ์กัน มีกระบวนการจัดหมวดหมู่ของข้อมูล

ที่มีแบบแผน ซึ่งก่อให้เกิดฐานข้อมูลที่เป็นแหล่งรวมข้อมูลจากแผนกต่าง ๆ และถูกจัดเก็บไว้อย่างเป็นระบบภายในฐานข้อมูลชุดเดียว ซึ่งผู้ใช้งานต่าง ๆ ในแต่ละแผนก สามารถใช้ข้อมูลส่วนกลางนี้เพื่อนำไปประมวลผลร่วมกันได้

สุรัสวดี วงศ์จันทร์สุข และสังจะ จรัสรุ่งรวิวรร (2549, หน้า 30) กล่าวว่า ฐานข้อมูลคือ วิธีการจัดการกับข้อมูลที่มีอยู่อย่างเป็นระบบ การจัดการในที่นี้ ไม่ได้หมายความว่าเพียงแค่การจัดเก็บข้อมูลเท่านั้น แต่ยังรวมไปถึงการควบคุมการเปลี่ยนแปลงที่จะเกิดขึ้นกับข้อมูล การดูแลให้ข้อมูลนั้นยังคงถูกต้องและปลอดภัยอยู่เสมอ นอกจากนี้ยังรวมไปถึงการเรียกเอาข้อมูลไปใช้งาน ไม่ว่าจะเป็นการนำเสนอรายงานต่าง ๆ หรือการวิเคราะห์ตัวข้อมูลที่เกี่ยวข้องด้วย

ทบวงมหาวิทยาลัย, โครงการเครือข่ายสารสนเทศเพื่อพัฒนาการศึกษา (2544, หน้า 17) อธิบายว่า ฐานข้อมูล (database) หมายถึง กลุ่มของข้อมูลที่ถูกเก็บรวบรวมไว้โดยมีความสัมพันธ์ซึ่งกันและกัน และไม่ได้บังคับว่าข้อมูลทั้งหมดนี้จะต้องเก็บไว้ในแฟ้มข้อมูลเดียวกัน หรือจะแยกเก็บเป็นหลาย ๆ แฟ้มข้อมูล นั่นคือ การเก็บข้อมูลในฐานข้อมูลนั้น อาจจะเก็บทั้งฐานข้อมูลโดยใช้แฟ้มข้อมูลเพียงแฟ้มเดียวกันได้ หรือจะเก็บไว้ในหลาย ๆ แฟ้มข้อมูล ที่สำคัญคือ จะต้องสร้างความสัมพันธ์ระหว่างระเบียบและเรียกใช้ความสัมพันธ์นั้นด้วย โดยจะต้องกำจัดความซ้ำซ้อนของข้อมูลออกและเก็บแฟ้มข้อมูลเหล่านี้ไว้ที่ศูนย์กลางเพื่อที่จะนำข้อมูลเหล่านี้มาใช้ร่วมกัน ควบคุมดูแลรักษาเมื่อผู้ต้องการใช้งานและผู้มีสิทธิ์จะใช้ข้อมูลนั้นสามารถดึงข้อมูลที่ต้องการออกไปใช้ได้ ข้อมูลบางส่วนอาจใช้ร่วมกับผู้อื่นได้ แต่บางส่วนผู้มีสิทธิ์เท่านั้นจึงจะสามารถใช้ได้

กล่าวโดยสรุป ฐานข้อมูล คือ ข้อมูลที่ถูกเก็บรวบรวมไว้ และมีการจัดการข้อมูลอย่างเป็นระบบ ข้อมูลมีความสัมพันธ์ซึ่งกันและกัน โดยเก็บในแฟ้มข้อมูลเดียวกัน หรือแยกเก็บไว้หลาย ๆ แฟ้มข้อมูลก็ได้ นั่นคือ การเก็บข้อมูลในฐานข้อมูลนั้น อาจจะเก็บทั้งฐานข้อมูล โดยใช้แฟ้มข้อมูลเพียงแฟ้มข้อมูลเดียวกันหรือจะเก็บไว้ในหลาย ๆ แฟ้มข้อมูล และต้องสร้างความสัมพันธ์ระหว่างระเบียบและเรียกใช้ความสัมพันธ์นั้นด้วย โดยจะเก็บไว้ที่ศูนย์กลางเพื่อที่จะนำข้อมูลเหล่านี้มาใช้ร่วมกัน มีการควบคุมดูแลรักษาความปลอดภัยของข้อมูล เมื่อผู้ต้องการใช้งานและผู้มีสิทธิ์จะใช้ข้อมูลนั้นสามารถดึงข้อมูล

ที่ต้องการออกไปใช้ได้ ข้อมูลบางส่วนอาจใช้ร่วมกับผู้อื่นได้ แต่บางส่วนผู้มีสิทธิ์เท่านั้น จึงจะสามารถใช้ได้

ระบบฐานข้อมูล ประกอบด้วยส่วนประกอบ 2 ส่วน ดังนี้ (วิธาน เจริญผล, ม.ป.ป.)

1. ระบบจัดการฐานข้อมูล (Database Management System--DBMS) เป็น โปรแกรม หรือกลุ่มของซอฟต์แวร์โปรแกรม ที่ทำหน้าที่เข้าถึงและจัดการกับข้อมูลจำนวนมาก ที่มีความสัมพันธ์กัน เปรียบเสมือนตัวกลางระหว่างผู้ใช้งานกับฐานข้อมูลและระบบปฏิบัติการของเครื่องคอมพิวเตอร์

2. Database Application เป็น โปรแกรมที่ช่วยดูแลและแก้ไขข้อมูลที่ถูกเก็บ อยู่ในระบบจัดการฐานข้อมูล

โดยปกติทั้งระบบจัดการฐานข้อมูล (Database Management System--DBMS) และ Database Application จะทำงานอยู่บนคอมพิวเตอร์เครื่องเดียวกัน ส่วนใหญ่ ทั้งสองส่วนจะถูกรวมอยู่ในโปรแกรมเดียวกัน แต่ขณะนี้ความสนใจส่วนใหญ่ได้มุ่งเน้น เทคโนโลยีในการปฏิวัติระบบจัดการฐานข้อมูล คือ เทคโนโลยีเครือข่ายแบบผู้ใช้/ ผู้ให้บริการ (client/server)

ความสำคัญของระบบฐานข้อมูล

การจัดข้อมูลให้เป็นระบบฐานข้อมูล ทำให้ข้อมูลมีส่วนดีกว่าการเก็บข้อมูล ในรูปของแฟ้มข้อมูล เพราะการจัดเก็บข้อมูลในระบบฐานข้อมูลจะมีส่วนที่สำคัญกว่า การจัดเก็บข้อมูลในรูปของแฟ้มข้อมูล ดังนี้ (ทบวงมหาวิทยาลัย, โครงการเครือข่าย- สารสนเทศเพื่อพัฒนาการศึกษา, 2544, หน้า 17-18)

1. ลดการเก็บข้อมูลที่ซ้ำซ้อน ข้อมูลบางชุดที่อยู่ในรูปของแฟ้มข้อมูลอาจปรากฏ อยู่หลาย ๆ แห่ง เพราะมีผู้ใช้ข้อมูลชุดนี้หลายคน เมื่อใช้ระบบฐานข้อมูลแล้วจะช่วยให้ ความซ้ำซ้อนของข้อมูลลดน้อยลง โดยจัดเก็บในฐานข้อมูลไว้ในที่เดียวกัน ผู้ใช้ทุกคน ที่ต้องการใช้ข้อมูลชุดนี้จะใช้โดยผ่านระบบฐานข้อมูล ทำให้ไม่สิ้นเปลืองเนื้อที่ ในการเก็บข้อมูลและลดความซ้ำซ้อนลงได้

2. รักษาความถูกต้องของข้อมูล เนื่องจากฐานข้อมูลมีเพียงฐานข้อมูลเดียว ในกรณีที่มีข้อมูลชุดเดียวกันปรากฏอยู่หลายแห่งในฐานข้อมูล ข้อมูลเหล่านี้จะต้องตรงกัน

หากมีการแก้ไขข้อมูล ทุก ๆ แห่งที่ข้อมูลปรากฏอยู่จะแก้ไขให้ถูกต้องตามกันหมด โดยอัตโนมัติด้วยระบบจัดการฐานข้อมูล

3. การป้องกันและรักษาความปลอดภัยให้กับข้อมูล ทำได้โดยให้ใช้เฉพาะผู้ที่เกี่ยวข้องเท่านั้นที่จะมีสิทธิ์เข้าไปใช้ฐานข้อมูลได้ เรียกว่า สิทธิส่วนบุคคล (privacy) ซึ่งก่อให้เกิดความปลอดภัย (security) ของข้อมูลด้วย ฉะนั้น ผู้ใดจะมีสิทธิ์ที่จะเข้าถึงข้อมูลได้จะต้องมีการกำหนดสิทธิ์กันไว้ก่อน และเมื่อเข้าไปใช้ข้อมูลนั้น ๆ ผู้ใช้จะเห็นข้อมูลที่ถูกเก็บไว้ในฐานข้อมูลในรูปแบบที่ผู้ใช้ออกแบบไว้ ตัวอย่างเช่น ผู้ใช้สร้างตารางข้อมูลขึ้นมาและเก็บลงในระบบฐานข้อมูล ระบบจัดการฐานข้อมูลจะเก็บข้อมูลเหล่านี้ลงในอุปกรณ์เก็บข้อมูลในรูปแบบของระบบจัดการฐานข้อมูล ซึ่งอาจเก็บข้อมูลเหล่านี้ลงในแผ่นจานบันทึกแม่เหล็กเป็นระเบียบน บล็อก หรืออื่น ๆ ผู้ใช้ไม่จำเป็นต้องรับรู้ว่าโครงสร้างของแฟ้มข้อมูลนั้นเป็นอย่างไร ปล่อยให้เป็นที่หน้าทีของระบบจัดการฐานข้อมูล

ดังนั้น ถ้าผู้ใช้เปลี่ยนแปลงลักษณะการเก็บข้อมูล เช่น เปลี่ยนแปลงรูปแบบของตารางเสียใหม่ ผู้ใช้ก็ไม่ต้องกังวลว่าข้อมูลจะถูกเก็บลงในแผ่นจานบันทึกแม่เหล็กในลักษณะใด ระบบการจัดการฐานข้อมูลจะจัดการให้ทั้งหมด ในทำนองเดียวกัน ถ้าผู้ออกแบบระบบฐานข้อมูลเปลี่ยนวิธีการเก็บข้อมูลลงบนอุปกรณ์จัดเก็บข้อมูล ผู้ใช้ไม่ต้องแก้ไขฐานข้อมูลที่ได้ออกแบบไว้แล้ว ระบบการจัดการฐานข้อมูลจะจัดการให้ลักษณะเช่นนี้เรียกว่า ความไม่เกี่ยวข้องกันของข้อมูล (data independent)

4. สามารถใช้ข้อมูลร่วมกันได้ เนื่องจากในระบบฐานข้อมูลจะเป็นที่เก็บรวบรวมข้อมูลทุกอย่างไว้ ผู้ใช้แต่ละคนจึงสามารถที่จะใช้ข้อมูลในระบบได้ทุกข้อมูล ถ้าข้อมูลไม่ได้ถูกจัดให้เป็นระบบฐานข้อมูลแล้ว ผู้ใช้ก็จะใช้ได้เพียงข้อมูลของตนเองเท่านั้น

5. มีความเป็นอิสระของข้อมูล เมื่อผู้ใช้ต้องการเปลี่ยนแปลงข้อมูลหรือนำข้อมูลมาประยุกต์ใช้ให้เหมาะสมกับโปรแกรมที่เขียนขึ้นมา จะสามารถสร้างข้อมูลนั้นขึ้นมาใช้ใหม่ได้ โดยไม่มีผลกระทบต่อระบบฐานข้อมูล เพราะข้อมูลที่ผู้ใช้นำมาประยุกต์ใช้ใหม่นั้นจะไม่กระทบต่อโครงสร้างที่แท้จริงของการจัดเก็บข้อมูล นั่นคือ การใช้ระบบฐานข้อมูลจะทำให้เกิดความเป็นอิสระระหว่างการจัดเก็บข้อมูลและการประยุกต์ใช้

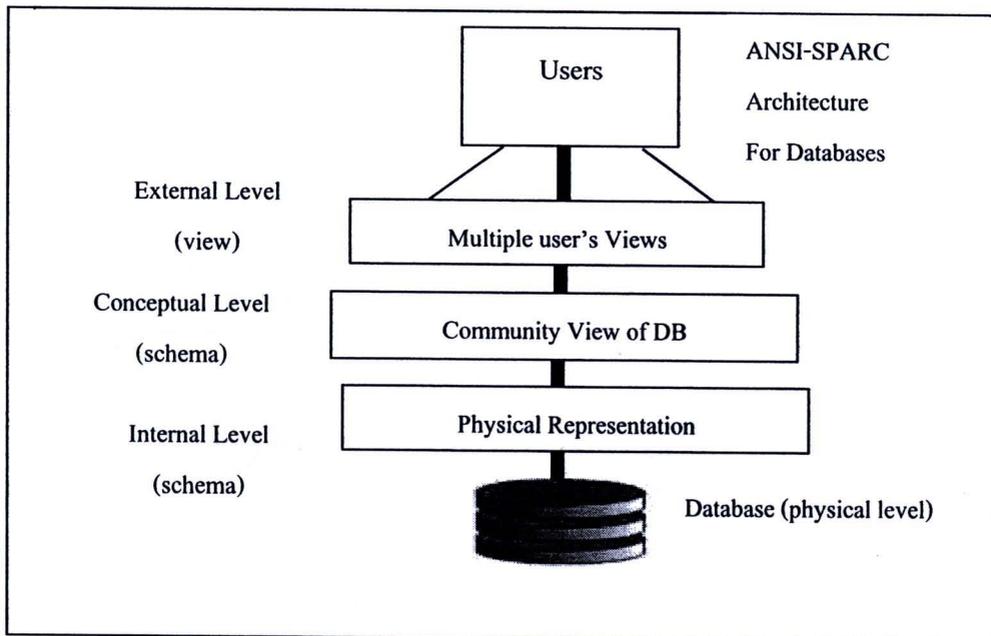
6. สามารถขยายงานได้ง่าย เมื่อต้องการจัดเพิ่มเติมข้อมูลที่เกี่ยวข้องจะสามารถเพิ่มได้อย่างง่ายไม่ซับซ้อน เนื่องจากมีความเป็นอิสระของข้อมูล จึงไม่มีผลกระทบต่อข้อมูลเดิมที่มีอยู่

7. ทำให้ข้อมูลบูรณะกลับสู่สภาพปกติได้เร็วและมีมาตรฐาน เนื่องจากการจัดพิมพ์ข้อมูลในระบบที่ไม่ได้ใช้ฐานข้อมูล ผู้เขียนโปรแกรมแต่ละคนมีเพิ่มข้อมูลของตนเอง โดยเฉพาะ ฉะนั้น แต่ละคนจึงต่างสร้างระบบการบูรณะข้อมูลให้กลับสู่สภาพปกติในกรณีที่ข้อมูลเสียหายด้วยตนเองและด้วยวิธีการของตนเอง จึงขาดประสิทธิภาพและมาตรฐาน แต่เมื่อมาเป็นระบบฐานข้อมูลแล้ว การบูรณะข้อมูลให้กลับคืนสู่สภาพปกติจะมีโปรแกรมชุดเดียวและมีผู้ดูแลเพียงคนเดียวที่ดูแลทั้งระบบ ซึ่งย่อมต้องมีประสิทธิภาพและเป็นมาตรฐานเดียวกันแน่นอน

ระดับของข้อมูลในระบบฐานข้อมูล

ระบบฐานข้อมูลเป็นการนำข้อมูลขององค์กรที่มีความเกี่ยวข้องกันมารวมไว้ อย่างเป็นระบบในที่เดียวกัน โดยผู้ใช้งานข้อมูลจะมองข้อมูลนี้ในแง่มุมหรือวิวที่แตกต่างกันไปตามจุดประสงค์ของการประยุกต์ใช้งาน โดยผู้ใช้ไม่จำเป็นต้องสนใจว่า ลักษณะการจัดเก็บข้อมูล โดยแท้จริงแล้วเป็นเช่นไร ระบบฐานข้อมูลจะซ่อนรายละเอียดไว้โดยจัดแบ่งระดับของข้อมูลออกเป็นระดับชั้น (มหาวิทยาลัยราชภัฏกาฬสินธุ์, 2547)

จากการศึกษาข้อมูลของทบวงมหาวิทยาลัย, โครงการเครือข่ายสารสนเทศเพื่อพัฒนาการศึกษา (2544, หน้า 17-18) พบว่า ระดับชั้นของข้อมูลถูกพัฒนาขึ้นโดย The Standards Planning and Requirements Committee (SPARC) ของ American National Standards Institute (ANSI) ดังแสดงในภาพ 2



ภาพ 2 ระดับการมองที่ฐานข้อมูล

ที่มา. จาก เอกสารประกอบการสอน: การจัดการระบบฐานข้อมูลทางธุรกิจ, โดย พัฒนพงษ์ โพธิ์ปัสสา, ม.ป.ป., ค้นเมื่อ 19 มกราคม 2554, จาก <http://www.kana.sskru.ac.th/compsheet/4121205/ch4.doc>

ระดับชั้นของข้อมูลถูกแบ่งออกเป็น 3 ระดับ ดังนี้ (มหาวิทยาลัยราชภัฏกาฬสินธุ์, 2547)

1. ระดับภายใน คือ ระดับที่แสดงลักษณะทางกายภาพของฐานข้อมูลที่อยู่ในคอมพิวเตอร์ ระดับนี้จะรวม โครงสร้างข้อมูลและการจัดการแฟ้ม (file organization) ซึ่งบอกถึงวิธีการที่ข้อมูลถูกเก็บในแหล่งเก็บข้อมูลทางกายภาพของฐานข้อมูลนั้น โดยระบบบริหารฐานข้อมูล (database management system) ทำงานร่วมกับระบบปฏิบัติการ ในวิธีการที่จะวางข้อมูลในอุปกรณ์จัดเก็บ (storage device) รวมถึงการสร้างดัชนี (index) หรือกำหนดตัวชี้ (pointer) ที่ใช้ในการดึงข้อมูลจากฐานข้อมูล

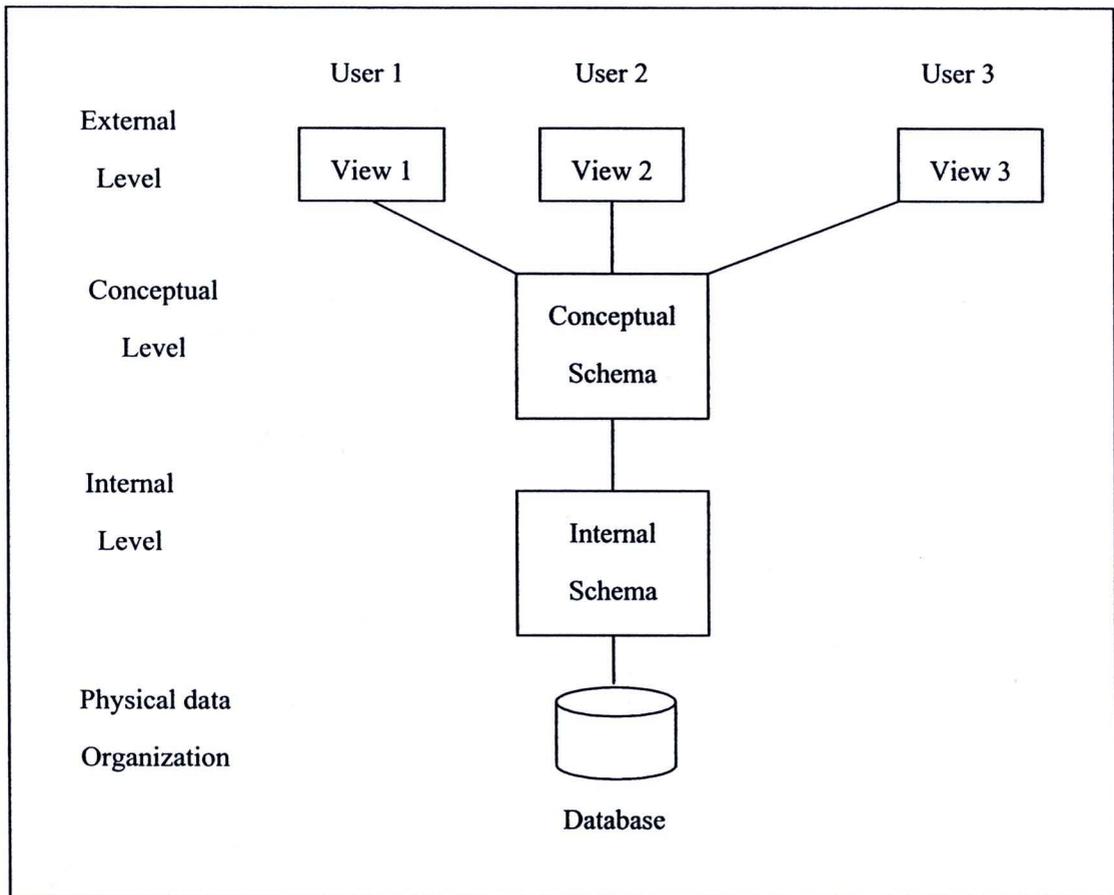
2. ระดับแนวคิด คือ มุมมองข้อมูลที่มีลักษณะร่วมกันของฐานข้อมูล ในระดับนี้อธิบายถึงโครงสร้างของฐานข้อมูลทั้งระบบในลักษณะของแนวคิด โดยบอกถึงสิ่งที่ข้อมูลเก็บอยู่ในฐานข้อมูลและความสัมพันธ์ระหว่างข้อมูลเหล่านั้น เช่น ชนิดของข้อมูลที่อาจ

เป็นทั้งจำนวนเต็ม (integer) ตัวอักษร (character) ขนาดของ Field มุมมองเหล่านี้จะอิสระจากแหล่งเก็บข้อมูลจริง คือ จะไม่มองถึงจำนวน Byte ของข้อมูลที่เก็บในฐานข้อมูลนั้น ในมุมมองระดับความคิดจะบ่งบอกถึง Entity ทั้งหมดที่อยู่ในแฟ้มข้อมูล ลักษณะของข้อมูลและความสัมพันธ์ระหว่างแฟ้มข้อมูล เงื่อนไขต่าง ๆ ที่มีมากับข้อมูล สารสนเทศ ที่เกี่ยวข้องกับการรักษาความปลอดภัย และการทำให้ข้อมูลสอดคล้องและเป็นหนึ่งเดียว

3. ระดับภายนอก คือ มุมมองฐานข้อมูลของผู้ใช้ในระบับนี้จะประกอบด้วย มุมมองในฐานข้อมูลที่แตกต่างกันของผู้ใช้จำนวนมาก ผู้ใช้แต่ละคนจะมีการมองข้อมูล ในลักษณะที่ผู้ใช้สนใจ ซึ่งจะแสดงออกมาในรูปแบบที่แตกต่างกัน ทั้งที่มาจากข้อมูล ชุดเดียวกัน จึงขึ้นอยู่กับความต้องการของผู้ใช้แต่ละคนว่า ต้องการให้แสดงข้อมูลออกมา ในลักษณะใดตามที่ตนเองต้องการ

ระบบการจัดการฐานข้อมูล

ระบบการจัดการฐานข้อมูล (Database Management System--DBMS) เป็นซอฟต์แวร์ ที่เหมือนกับสื่อกลางระหว่างผู้ใช้และ โปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้ฐานข้อมูล โดยที่ผู้ใช้ไม่จำเป็นต้องทราบเกี่ยวกับรายละเอียดภายใน โครงสร้างของฐานข้อมูล เช่น ทำหน้าที่ในการควบคุม ดูแล การสร้างตาราง และเรียกใช้ฐานข้อมูล ดังแสดงในภาพ 3



ภาพ 3 ระบบการจัดการฐานข้อมูล DBMF

ที่มา. จาก โปรแกรมจัดการฐานข้อมูล (DBMS), โดย ศูนย์กลางความรู้แห่งชาติ, 2553, ค้นเมื่อ 19 มกราคม 2554, จาก <http://www.tkc.go.th/wiki/show/%E0%B9%82%E0%B8%9B%E0%B8%A3%E0%B9%81%E0%B8%81%E0%B8%A3%E0%B8%A1%E0%B8%88%E0%B8%B1%E0%B8%94%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%90%E0%B8%B2%E0%B8%99%E0%B8%82%E0%B9%89%E0%B8%AD%E0%B8%A1%E0%B8%B9%E0%B8%A5+DBMS>

หน้าที่ของระบบการจัดการฐานข้อมูล

ระบบฐานการจัดการฐานข้อมูลที่ดีจะต้องทำหน้าที่แก้ปัญหาคความไม่สมบูรณ์ ไม่คงเส้นคงวาของข้อมูล และทำให้ข้อมูลมีความถูกต้อง ไม่ขัดแย้งกัน จึงต้องมีหน้าที่ครอบคลุมในหลาย ๆ ด้าน ดังนี้ (ัชวาล ชุมรักษา, ม.ป.ป.)

1. ช่วยกำหนดและเก็บโครงสร้างฐานข้อมูล (define and store database structure)
2. ช่วยดึงข้อมูลจากฐานข้อมูล (load database) ข้อมูลที่นำมาประมวลผลด้วยโปรแกรมคอมพิวเตอร์ ระบบจัดการฐานข้อมูลจะรับและเก็บข้อมูลไว้ในข้อมูลเพื่อใช้ในการประมวลผล
3. ช่วยเก็บและดูแลข้อมูล (store and maintain data) ข้อมูลที่เก็บในฐานข้อมูลจะถูกเก็บรวบรวมได้ด้วยกัน โดยมีระบบจัดการฐานข้อมูลเป็นผู้ดูแลรักษาข้อมูลเหล่านั้น
4. ช่วยประสานงานกับระบบปฏิบัติการ (operating system) เนื่องจากคอมพิวเตอร์ต้องพึ่งระบบปฏิบัติการช่วยให้ทำงานได้ ดังนั้น ระบบปฏิบัติการจะคอยควบคุมการทำงานของอุปกรณ์คอมพิวเตอร์หรือโปรแกรมต่าง ๆ เป็นต้น ระบบจัดการฐานข้อมูลจะประสานงานกับระบบปฏิบัติการในการเรียกใช้ แก้ไขข้อมูล ลบข้อมูล ออกรายงาน
5. ช่วยควบคุมความปลอดภัย (security control) ในระบบการจัดการฐานข้อมูลจะมีวิธีควบคุมการเรียกใช้ข้อมูล หรือแก้ไขเปลี่ยนแปลงข้อมูลของผู้ใช้ในระบบซึ่งสามารถเรียกข้อมูลมาแก้ไขได้แตกต่างกัน เพื่อป้องกันความเสียหายที่จะเกิดขึ้นกับฐานข้อมูล
6. ช่วยจัดทำสำรองข้อมูลและการกู้คืน (backup and recover) ในระบบจัดการฐานข้อมูลจะสำรองข้อมูลของฐานข้อมูล และเมื่อเกิดปัญหาขึ้นกับฐานข้อมูล เช่น เพิ่มข้อมูลเสียหายเนื่องจากคิส์ก์เสีย ระบบจัดการฐานข้อมูลจะใช้ระบบข้อมูลสำรองนี้ในการฟื้นฟูสภาพการทำงานของระบบให้สู่สภาวะปกติ
7. ช่วยควบคุมการใช้ข้อมูลพร้อมกันของผู้ใช้ในระบบ (concurrency control) ในระบบคอมพิวเตอร์ ระบบฐานข้อมูลที่มีคุณสมบัติผู้ใช้หลายคนจะสามารถเรียกข้อมูลได้พร้อมกัน ระบบฐานข้อมูลที่มีคุณสมบัตินี้ จะควบคุมการใช้ข้อมูลพร้อมกันของผู้ใช้หลายคนในเวลาเดียวกัน โดยมีการควบคุมอย่างถูกต้องเหมาะสม
8. ช่วยควบคุมความสมบูรณ์ของข้อมูล (integrity control) ระบบจัดการฐานข้อมูลจะควบคุมค่าของข้อมูลในระบบให้ถูกต้องตามที่ควรจะเป็น เช่น รหัสของนักศึกษาที่ลงทะเบียน จะต้องตรงกับรหัสนักศึกษาในข้อมูลประวัติ

9. ช่วยทำหน้าที่จัดทำบรรณานุกรม (data directory) ระบบจัดการฐานข้อมูล จะสร้างบรรณานุกรมเมื่อมีการกำหนดโครงสร้างของฐานข้อมูลขึ้นมาเพื่อเก็บรายละเอียดต่าง ๆ ที่เกี่ยวข้องกับฐานข้อมูล เช่น ชื่อตาราง ชื่อ Field บรรณานุกรมต่าง ๆ เป็นต้น

ภาษา SQL

ภาษา SQL หรือ Sequel ย่อมาจาก Structured Query Language หรือภาษาในการสอบถามข้อมูล เป็นภาษาทางด้านฐานข้อมูลที่สามารถสร้างและปฏิบัติการกับฐานข้อมูลแบบสัมพันธ์ (relational database) โดยเฉพาะ (ทบวงมหาวิทยาลัย, โครงการเครือข่ายสารสนเทศเพื่อพัฒนาการศึกษา, 2544, หน้า 173)

ประเภทของคำสั่งของภาษา SQL

ภาษา SQL เป็นภาษาที่ใช้งาน ได้ตั้งแต่ระดับเครื่องคอมพิวเตอร์ส่วนบุคคล (Personal computer--PC) จนถึงระดับเครื่องคอมพิวเตอร์แบบ Mainframe ประเภทของคำสั่งในภาษา SQL (the subdivision of SQL) แบ่งออกเป็น 3 ประเภท ดังนี้ (ทบวงมหาวิทยาลัย, โครงการเครือข่ายสารสนเทศเพื่อพัฒนาการศึกษา, 2544)

1. ภาษาสำหรับการนิยามข้อมูล (Data Definition Language--DDL) ประกอบด้วย คำสั่งที่ใช้ในการกำหนดโครงสร้างข้อมูลว่ามีคอลัมน์อะไร แต่ละคอลัมน์เก็บข้อมูลประเภทใด รวมถึงการเพิ่มคอลัมน์ การกำหนดดัชนี การกำหนดวิวหรือตารางเสมือนของผู้ใช้ เป็นต้น

2. ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language--DML) ประกอบด้วย คำสั่งที่ใช้ในการเรียกใช้ข้อมูล การเปลี่ยนแปลงข้อมูล การเพิ่มหรือลบข้อมูล เป็นต้น

3. ภาษาควบคุม (Data Control Language--DCL) ประกอบด้วย คำสั่งที่ใช้ในการควบคุม การเกิดภาวะพร้อมกัน หรือการป้องกันการเกิดเหตุการณ์ที่ผู้ใช้หลายคนเรียกใช้ข้อมูลพร้อมกัน และคำสั่งที่เกี่ยวข้องกับการควบคุมความปลอดภัยของข้อมูล ด้วยการกำหนดสิทธิ์ของผู้ใช้ที่แตกต่างกัน เป็นต้น

ระบบ MySQL

จากการศึกษาของบุรินทร์ รุจจนพันธุ์ (ม.ป.ป.-ง) กล่าวว่า MySQL คือ โปรแกรมระบบจัดการฐานข้อมูล มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL (structured query language) และเป็นเครื่องมือสำหรับเก็บข้อมูลที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับความต้องการของผู้ใช้ เช่น ทำงานร่วมกับเครื่องบริการเว็บ (web server) เพื่อให้บริการแก่ภาษา Script ที่ทำงานฝั่งเครื่องบริการ (server-side script) เช่น ภาษา PHP ภาษา ASP หรือภาษา JSP เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (application program) เช่น ภาษา Visual Basic ภาษา Java หรือภาษา C เป็นต้น

MySQL เป็นระบบฐานข้อมูลแบบ Open Source Database สำหรับจัดการระบบ Database ผ่าน SQL โปรแกรมนี้ถูกพัฒนาโดยบริษัท MySQL AB ในประเทศสวีเดน มีทั้งแบบใช้ฟรีและเชิงธุรกิจ

การวิเคราะห์และออกแบบระบบ

การวิเคราะห์และออกแบบระบบ (system analysis and design) คือ วิธีการที่ใช้ในการสร้างระบบสารสนเทศขึ้นมาใหม่ในธุรกิจใดธุรกิจหนึ่ง หรือระบบย่อยของธุรกิจ นอกจากการสร้างระบบสารสนเทศใหม่แล้ว การวิเคราะห์ระบบช่วยในการแก้ไขระบบสารสนเทศเดิมที่มีอยู่แล้วให้ดีขึ้นด้วยก็ได้ การวิเคราะห์ระบบ คือ การหาความต้องการ (requirements) ของระบบสารสนเทศว่าคืออะไร หรือต้องการเพิ่มเติมอะไรเข้ามาในระบบ ส่วนการออกแบบ คือ การนำเอาความต้องการของระบบมาเป็นแบบแผน หรือเรียกว่า พิมพ์เขียว ในการสร้างระบบสารสนเทศเพื่อให้ใช้ในการปฏิบัติงานได้จริง ผู้ที่ทำหน้าที่นี้คือ นักวิเคราะห์และออกแบบระบบ (System Analysis--SA)

วงจรการพัฒนาระบบ

วงจรการพัฒนาระบบ (System Development Life Cycle--SDLC) คือ ระบบสารสนเทศ มีขั้นตอนที่เป็นลำดับและเป็นระบบที่ใช้งานได้ ซึ่งนักวิเคราะห์ระบบต้องทำความเข้าใจให้ได้ว่าในแต่ละขั้นตอนจะต้องทำอะไร และทำอย่างไร ขั้นตอนการพัฒนาระบบมีอยู่ด้วยกัน 7 ขั้นตอน ดังนี้ (อำไพ วรรณสินธุ์, ม.ป.ป.)

1. เข้าใจปัญหา (problem recognition) ระบบสารสนเทศจะเกิดขึ้นได้ก็ต่อเมื่อผู้บริหารหรือผู้ใช้ตระหนักว่าต้องการระบบสารสนเทศหรือระบบจัดการเดิม ได้แก่ ระบบเอกสารในตู้เอกสารไม่มีประสิทธิภาพเพียงพอที่ตอบสนองความต้องการในปัจจุบัน การแก้ไขระบบเดิมที่มีอยู่แล้วไม่ใช่เรื่องที่ยั่งยืน หรือแม้แต่การสร้างระบบใหม่ ดังนั้น ควรจะมีการศึกษาเสียก่อนว่าความต้องการของตนเองเพียงพอที่เป็นไปได้หรือไม่ ได้แก่ การศึกษาความเป็นไปได้ (feasibility study)
2. ศึกษาความเป็นไปได้ (feasibility study) จุดประสงค์ของการศึกษาความเป็นไปได้นี้คือการกำหนดว่าปัญหาคืออะไร และตัดสินใจว่าการพัฒนาสร้างระบบสารสนเทศหรือการแก้ไขระบบสารสนเทศเดิมมีความเป็นไปได้หรือไม่ โดยเสียค่าใช้จ่ายและเวลาน้อยที่สุด และได้ผลเป็นที่น่าพอใจ นักวิเคราะห์ระบบจะต้องกำหนดให้ได้ว่าการแก้ไขปัญหาดังกล่าวมีความเป็นไปได้ทางเทคนิคและบุคลากร ปัญหาทางเทคนิคจะเกี่ยวข้องกับเรื่องคอมพิวเตอร์ และเครื่องมือต่าง ๆ รวมทั้งเครื่องคอมพิวเตอร์ซอฟต์แวร์ด้วย
3. การวิเคราะห์ระบบ (analysis system) เริ่มตั้งแต่การศึกษาระบบการทำงานกรณีที่ระบบนั้นเป็นระบบสารสนเทศเดิมอยู่แล้วจะต้องศึกษาว่าทำงานอย่างไร เพราะเป็นการยากที่จะออกแบบระบบใหม่โดยที่ไม่ทราบว่าจะระบบเดิมทำงานอย่างไร หรือธุรกิจดำเนินการอย่างไร หลังจากนั้นจึงกำหนดความต้องการของระบบใหม่ ซึ่งนักวิเคราะห์ระบบจะต้องใช้เทคนิคในการเก็บข้อมูล (fact-gathering techniques) ได้แก่ ศึกษาเอกสารที่มีอยู่ ตรวจสอบวิธีการทำงานในปัจจุบัน สัมภาษณ์ผู้ใช้และผู้จัดการที่มีส่วนเกี่ยวข้องกับระบบเอกสารที่มีอยู่ ได้แก่ คู่มือการใช้งาน แผนผังใช้งานขององค์กร รายงานต่าง ๆ ที่หมุนเวียนในระบบการศึกษา วิธีการทำงานในปัจจุบัน ทำให้นักวิเคราะห์ระบบรู้ว่าระบบจริง ๆ ทำงานอย่างไร เมื่อเก็บรวบรวมข้อมูลแล้วจะนำมาเขียนรวมเป็น

รายงานการทำงานของระบบ ซึ่งควรแสดงหรือเขียนออกมาเป็นแผนภาพ ทำให้เข้าใจ ได้ดีและง่ายขึ้น หลังจากนั้นนักวิเคราะห์ระบบอาจจะนำข้อมูลที่รวบรวมได้นำมาเขียน เป็นแบบทดลอง (prototype) หรือตัวต้นแบบ แบบทดลองจะเขียนขึ้นด้วยภาษาคอมพิวเตอร์ ต่าง ๆ ที่ช่วยให้ทำงานง่ายขึ้น เมื่อจบขั้นตอนการวิเคราะห์ นักวิเคราะห์ระบบจะต้องเขียน รายงานสรุปออกมาเป็นข้อมูลเฉพาะของปัญหา (problem specification)

4. การออกแบบ (design) นักวิเคราะห์ระบบจะนำการตัดสินใจของฝ่ายบริหาร ที่ได้จากขั้นตอนการวิเคราะห์การเลือกซื้อคอมพิวเตอร์ ฮาร์ดแวร์ และซอฟต์แวร์ด้วย (ถ้ามีหรือเป็นไปได้) หลังจากนั้นนักวิเคราะห์ระบบจะนำแผนภาพต่าง ๆ ที่เขียนขึ้น ในขั้นตอนการวิเคราะห์มาแปลงเป็นแผนภาพลำดับขั้น เพื่อให้มองเห็นภาพลักษณะ ที่แน่นอนของโปรแกรมว่ามีความสัมพันธ์กันอย่างไร และโปรแกรมอะไรบ้างที่จะต้อง เขียนในระบบ จากนั้นจึงตัดสินใจว่าควรจัด โครงสร้างจากโปรแกรมอย่างไร การเชื่อม ระหว่างโปรแกรมควรจะทำอย่างไร ในขั้นตอนการวิเคราะห์ นักวิเคราะห์ระบบต้องหาว่า จะต้องทำอะไร (what) ขั้นตอนการออกแบบต้องรู้อะไร (how) การออกแบบ โปรแกรมต้องคำนึงถึงความปลอดภัย (security) ของระบบด้วย

นักวิเคราะห์ระบบจะต้องออกแบบฟอร์มสำหรับข้อมูลขาเข้า (input format) ออกแบบรายงาน (report format) และการแสดงผลบนจอภาพ (screen format) หลักการออกแบบฟอร์มข้อมูลขาเข้า คือ ง่ายต่อการใช้งานและป้องกันข้อผิดพลาดที่อาจจะ เกิดขึ้น ถัดมาระบบจะต้องออกแบบวิธีการใช้งาน สิ่งนี้นักวิเคราะห์ระบบออกแบบ มาทั้งหมดจะนำมาเขียนรวมเป็นเอกสารชุดหนึ่งเรียกว่า ข้อมูลเฉพาะของการออกแบบ ระบบ (system design specification) เมื่อสำเร็จแล้ว โปรแกรมเมอร์สามารถใช้เป็นแบบ ในการเขียนโปรแกรมได้ทันที ก่อนที่จะเริ่มเขียน โปรแกรมควรตรวจสอบความพึงพอใจ ของผู้ใช้ และตรวจสอบความถูกต้องสมบูรณ์ด้วย

5. การพัฒนาระบบ (construction) ขั้นตอนนี้โปรแกรมเมอร์จะเริ่มเขียน และทดสอบโปรแกรมว่าทำงานถูกต้องหรือไม่ ต้องมีการทดสอบกับข้อมูลจริงที่เลือกแล้ว ถ้าระบบทำงานได้ถูกต้องจะได้นำโปรแกรมไปใช้งานจริง หลังจากนั้นต้องเตรียมคู่มือ การใช้และการฝึกอบรมผู้ใช้งานจริงของระบบ

6. การปรับเปลี่ยน (construction) ขั้นตอนนี้เป็นกรนำระบบใหม่มาใช้แทนของเก่า ภายใต้การดูแลของนักวิเคราะห์ระบบ การป้อนข้อมูลต้องทำให้เรียบร้อย การเริ่มต้นใช้งานระบบใหม่ควรจะทำอย่างค่อยเป็นค่อยไปที่ละน้อย วิธีที่ดีที่สุดคือการใช้ระบบใหม่ควบคู่กับระบบเก่าไปสักระยะหนึ่ง โดยจะต้องใช้ข้อมูลชุดเดียวกันแล้วเปรียบเทียบผลลัพธ์ว่าตรงกันหรือไม่ ถ้าไม่พบปัญหาจึงเอาระบบเก่าออกได้ แล้วใช้ระบบใหม่ต่อไป

7. การบำรุงรักษา (maintenance) ได้แก่ การแก้ไขโปรแกรมหลังจากการใช้งานแล้ว สาเหตุที่ต้องแก้ไขโปรแกรมหลังจากใช้งานส่วนใหญ่มี 2 ข้อ ดังนี้

7.1 มีปัญหาในโปรแกรม (bug)

7.2 การดำเนินงานในองค์กรหรือธุรกิจเปลี่ยนไป

ผังงาน

ผังงาน (flow chart) คือ รูปภาพ (image) หรือสัญลักษณ์ (symbol) ที่ใช้เขียนแทนขั้นตอน คำอธิบาย ข้อความ หรือคำพูดที่ใช้ใน Algorithm เพราะการนำเสนอขั้นตอนของงานให้เข้าใจตรงกัน ระหว่างผู้เกี่ยวข้องด้วยคำพูดหรือข้อความทำได้ยากกว่าเมื่อใช้รูปภาพหรือสัญลักษณ์ ผังงานแบ่งได้ 2 ประเภท ดังนี้

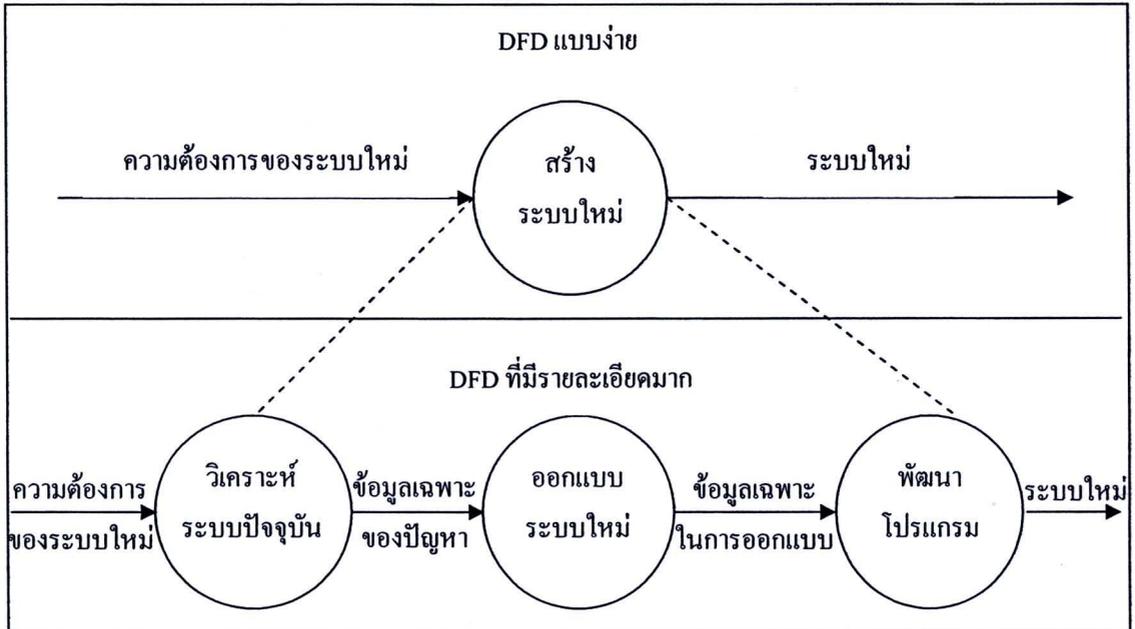
1. ผังงานระบบ (system flowchart) คือ ผังงานที่แสดงขั้นตอนการทำงานในระบบอย่างกว้าง ๆ แต่ไม่เจาะลงในระบบงานย่อย

2. ผังงานโปรแกรม (program flowchart) คือ ผังงานที่แสดงถึงขั้นตอนในการทำงานของโปรแกรม ตั้งแต่รับข้อมูล คำนวณ จนถึงแสดงผล

แผนภาพกระแสข้อมูล

จากการศึกษาของกิตติมา เจริญหิรัญ (2550, หน้า 68) ได้อธิบายไว้ว่า แผนภาพกระแสข้อมูล (Data Flow Diagram--DFD) เป็นเครื่องมือที่แสดงถึงทิศทางการส่งผ่านของข้อมูลในระบบ แสดงความสัมพันธ์ระหว่างกระบวนการกับข้อมูลที่เกี่ยวข้องภายในระบบ แสดงการส่งผ่านข้อมูลนำเข้าและข้อมูลนำออก และขั้นตอนการทำงานของระบบ

เป็นการแสดงการทำงานของระบบ โดยไม่อธิบายถึงขั้นตอนและวิธีการประมวลผล โดยแสดงว่าระบบทำงานอะไร แต่ไม่แสดงว่าระบบทำงานอย่างไร ดังแสดงในภาพ 4



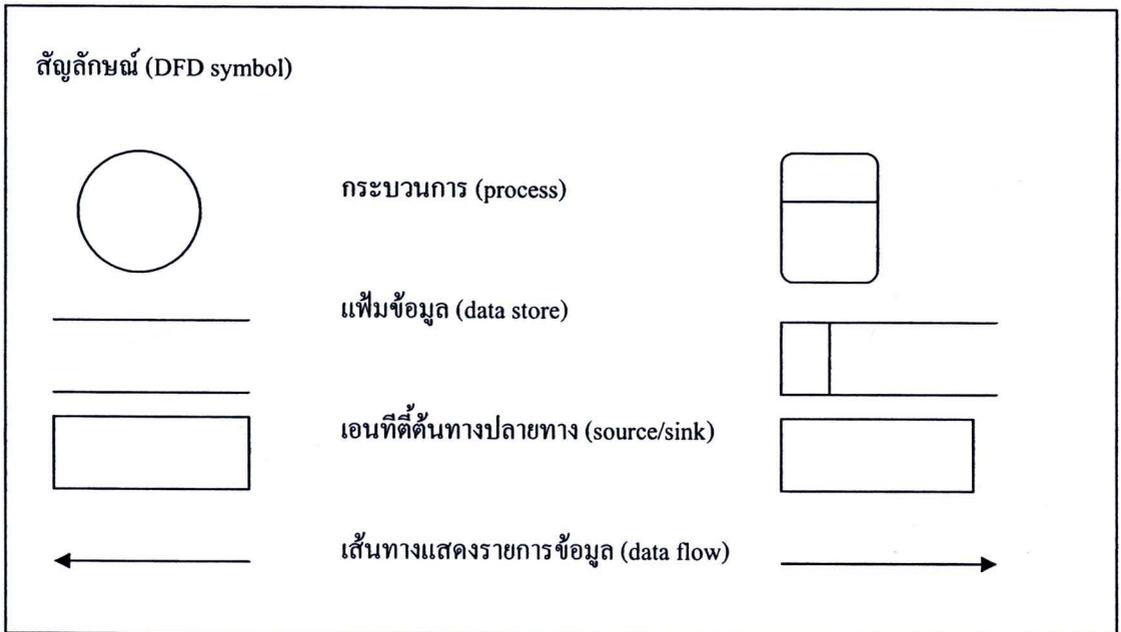
ภาพ 4 รูป DRD แบบต่าง ๆ

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

ส่วนประกอบของ DFD

แผนภาพกระแสข้อมูล มีองค์ประกอบ 4 อย่าง (ดูภาพ 5 ประกอบ) ซึ่งใช้สัญลักษณ์ต่าง ๆ แทน ดังต่อไปนี้ (ปรัชญา ศิริภูรี, ม.ป.ป.)

1. สัญลักษณ์แทนการประมวลผล (process) เป็นวงกลม
2. สัญลักษณ์แทนกระแสข้อมูลเป็นลูกศร
3. สัญลักษณ์แทนแหล่งเก็บข้อมูลเป็นเส้นขนาน 2 เส้น โดยมีชื่อกำกับ
4. สี่เหลี่ยมผืนผ้าเป็นสัญลักษณ์แทนสิ่งที่อยู่นอกระบบ



ภาพ 5 สัญลักษณ์ของ DFD

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย
 ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

พจนานุกรมข้อมูล

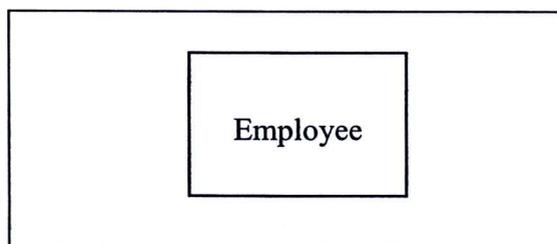
พจนานุกรมข้อมูล (data dictionary) คือ แฟ้มที่เก็บบันทึกรายละเอียดต่าง ๆ เกี่ยวกับข้อมูลที่จัดเก็บอยู่ในฐานข้อมูล ตัวอย่างเช่น โครงสร้างข้อมูล โครงสร้างตาราง โครงสร้างดัชนีกฎที่ใช้เพื่อควบคุมความบูรณาภาพของข้อมูล (integrity rule) กฎที่ใช้เพื่อรักษาความปลอดภัยของข้อมูล (security rule) และรายละเอียดอื่น ๆ ที่เกี่ยวข้องกับการบริหารฐานข้อมูล เป็นต้น ซึ่งข้อมูลดังกล่าวเป็นข้อมูลที่มีความสำคัญสำหรับระบบจัดการฐานข้อมูลในการตัดสินใจ เพื่อดำเนินการเรื่องต่าง ๆ ในฐานข้อมูล ตัวอย่างเช่น ข้อมูลเกี่ยวกับกฎที่ใช้เพื่อรักษาความปลอดภัยของข้อมูล จะถูกนำมาใช้ในการพิจารณากำหนดสิทธิของผู้ใช้ในระบบฐานข้อมูล (ทบทวมหาวิทยาลัย, โครงการเครือข่ายสารสนเทศเพื่อพัฒนาการศึกษา, 2544)

การออกแบบฐานข้อมูลโดยวิธีแบบจำลองข้อมูล

Entity-Relationship Model หรือ E-R Model คือ โมเดลการออกแบบฐานข้อมูล หรือเรียกกัน โดยทั่วไปว่า ER-Diagram นั้น ประกอบด้วยองค์ประกอบพื้นฐาน ได้แก่ (1) Entity เป็นวัตถุหรือสิ่งของที่สนใจในระบบงานนั้น ๆ (2) Attribute เป็นคุณสมบัติของวัตถุที่สนใจ (3) Relationship คือ ความสัมพันธ์ระหว่าง Entity (ปรัชญา ศิริภูรี, ม.ป.ป.)

Entity

Entity หมายถึง สิ่งของหรือวัตถุที่สนใจ ซึ่งอาจจับต้องได้และเป็นได้ทั้งนามธรรม โดยทั่วไป Entity จะมีลักษณะที่แยกออกจากกันไป Entity จะมีกลุ่มที่บอกคุณสมบัติที่บอกลักษณะของ Entity โดยจะมีค่าของคุณสมบัติบางกลุ่มที่ทำให้สามารถแยก Entity ออกจาก Entity อื่นได้ เรียกค่าของคุณสมบัติกลุ่มนี้ว่าเป็นคีย์ของ Entity รูปสัญลักษณ์ของ Entity คือ รูปสี่เหลี่ยมผืนผ้า ดังแสดงในภาพ 6

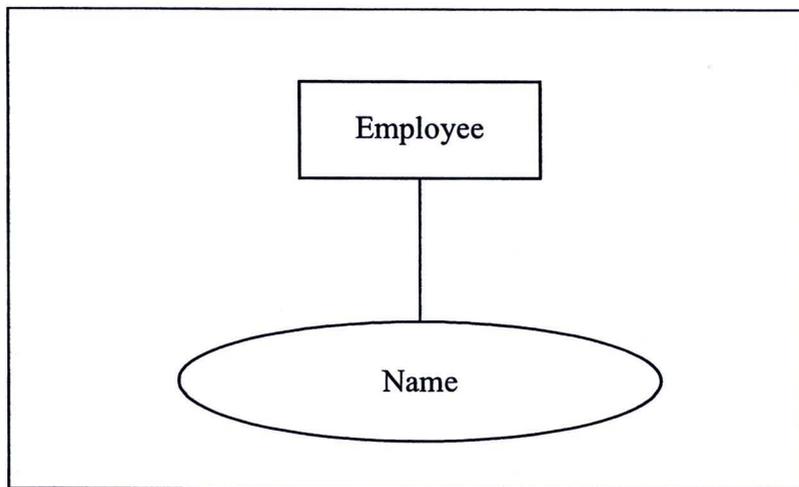


ภาพ 6 สัญลักษณ์ของ Entity

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

Attribute

Attribute คือ คุณสมบัติของวัตถุหรือสิ่งของที่สนใจ โดยอธิบายรายละเอียดต่าง ๆ ที่เกี่ยวข้องกับลักษณะของ Entity โดยคุณสมบัตินี้มีอยู่ในทุก Entity โดยทั่วไปแล้ว โมเดลข้อมูลจะพบว่า Attribute มีลักษณะข้อมูลพื้นฐานอยู่โดยที่ไม่ต้องมีคำอธิบายมากมาย และ Attribute ก็ไม่สามารถอยู่แบบโดด ๆ โดยไม่มี Entity หรือความสัมพันธ์ รูปสัญลักษณ์ของ Attribute คือ รูปวงรี โดยที่จะมีเส้นเชื่อมต่อกับ Entity ดังแสดงในภาพ 7



ภาพ 7 สัญลักษณ์ของ Attribute

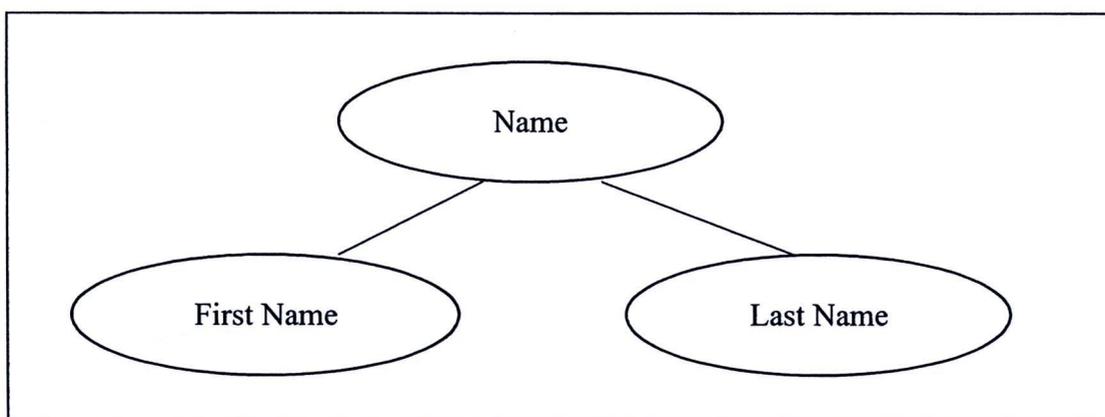
ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

ชนิดของ Attribute สามารถแบ่งออกได้หลายลักษณะดังนี้ (ปรัชญา ศิริภูรี, ม.ป.ป.)

1. Simple และ Composite

1.1 Simple Attribute คือ Attribute ที่ไม่สามารถแยกออกเป็นส่วนย่อยได้
เช่น รหัส

1.2 Composite Attribute คือ Attribute ที่สามารถแยกออกเป็นส่วนย่อยได้
เช่น ชื่อ อาจจะประกอบด้วยชื่อต้นและชื่อสกุล เป็นต้น ดังแสดงในภาพ 8



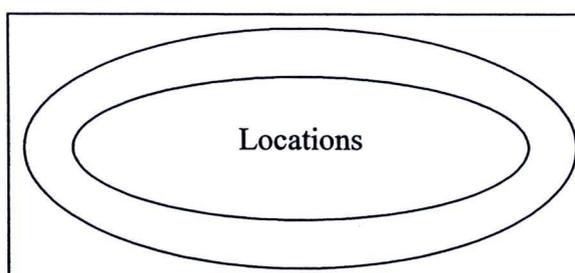
ภาพ 8 สัญลักษณ์ของ Composite attribute

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย
 ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

2. Single-valued และ Multi-valued attribute

2.1 Single-valued คือ ค่าของ Entity ที่สามารถมีได้แค่ค่าเดียว เช่น วันเกิด
 พนักงานซึ่งมีได้เพียงค่าเดียว จึงให้สัญลักษณ์ของ Attribute ปกติ

2.2 Multi-valued คือ ค่าที่เป็นไปได้มากกว่า 1 ค่า เช่น ทำเลที่ตั้งโรงงาน
 ซึ่งมีได้มากกว่า 1 แห่ง รูปสัญลักษณ์ที่ใช้จะเป็นรูปวงรีซ้อนกัน 2 รูป ดังแสดงในภาพ 9



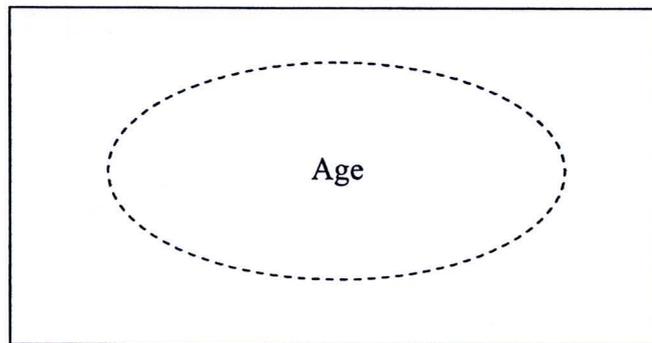
ภาพ 9 สัญลักษณ์ของ Multi-valued

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย
 ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

3. Stored Attribute และ Derived Attribute

3.1 Stored Attribute จะเป็น Attribute ที่เก็บอยู่ในฐานข้อมูล เช่น วันเกิด ใช้สัญลักษณ์ปกติ

3.2 Derived Attribute เป็น Attribute ที่เกิดจากการคำนวณ รูปสัญลักษณ์ คือ รูปวงรีมีเส้นประรอบๆ ดังแสดงในภาพ 10

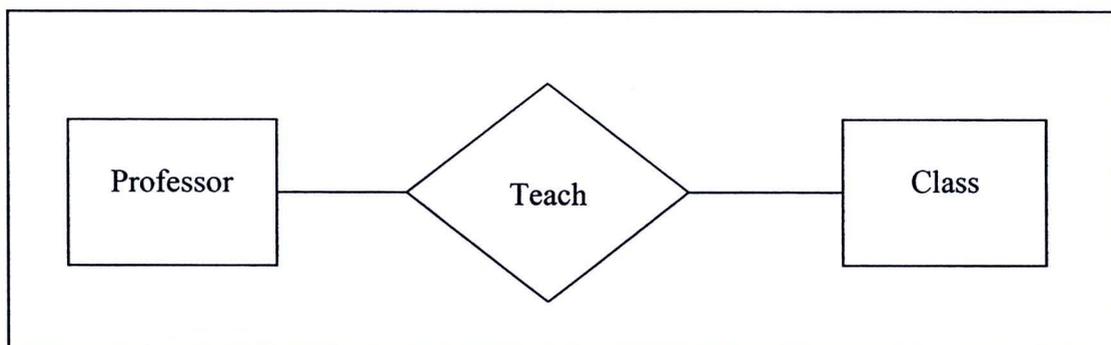


ภาพ 10 สัญลักษณ์ของ Derived attribute

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

ความสัมพันธ์

จากการศึกษาของปรัชญา ศิริภูรี (ม.ป.ป.) ได้กล่าวว่า Entity แต่ละ Entity จะต้องมีความสัมพันธ์ (relationship) ร่วมกัน โดยจะมีชื่อแสดงความสัมพันธ์ร่วมกัน ซึ่งจะใช้รูปภาพสัญลักษณ์ที่เหลื่อมรูปว่าว แสดงความสัมพันธ์ระหว่าง Entity และระบุชื่อความสัมพันธ์ลงในสี่เหลี่ยม ดังแสดงในภาพ 11



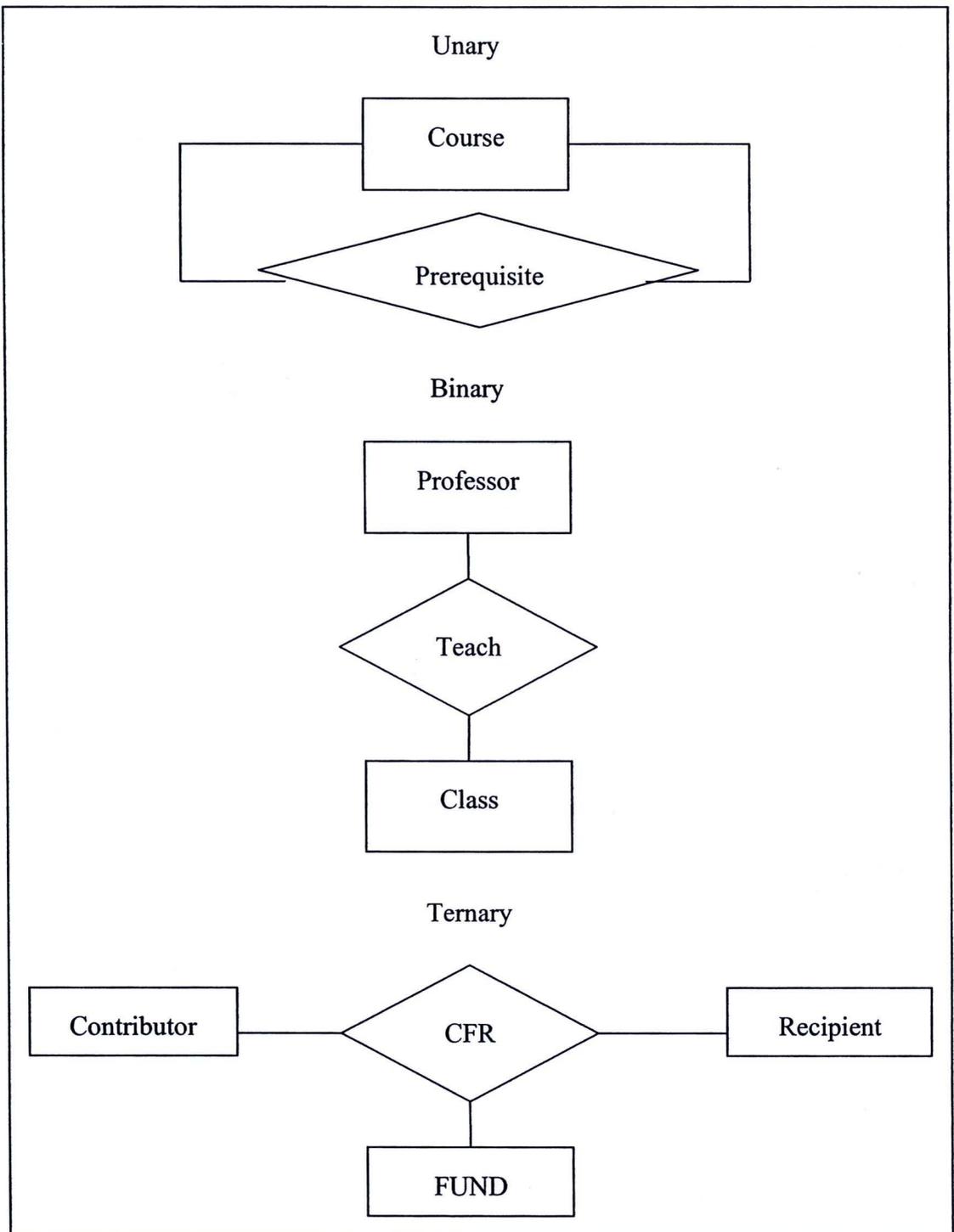
ภาพ 11 ความสัมพันธ์ระหว่าง Entity อาจารย์กับกลุ่มเรียน

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย
 ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

Entity นั้นจะต้องมีความสัมพันธ์ร่วมกัน โดยจะมีชื่อแสดงความสัมพันธ์ร่วมกัน ซึ่งจะใช้รูปภาพสัญลักษณ์ที่เปลี่ยนรูปว่า แสดงความสัมพันธ์ระหว่าง Entity ระดับชั้นของความสัมพันธ์ (relationships degree) จะบอกถึงความสัมพันธ์ระหว่าง Entity ดังนี้ (ดูภาพ 12) (ปรัชญา ศิริภูรี, ม.ป.ป.)

1. ระดับชั้นของความสัมพันธ์ (relationships degree) จะบอกถึงความสัมพันธ์ระหว่าง Entity ดังนี้

- 1.1 Unary Relationships หมายถึง Entity หนึ่ง ๆ จะมีความสัมพันธ์กับตัวมันเอง
- 1.2 Binary Relationships หมายถึง Entity สอง Entity จะมีความสัมพันธ์กัน
- 1.3 Ternary Relationships หมายถึง Entity สาม Entity มีความสัมพันธ์กัน

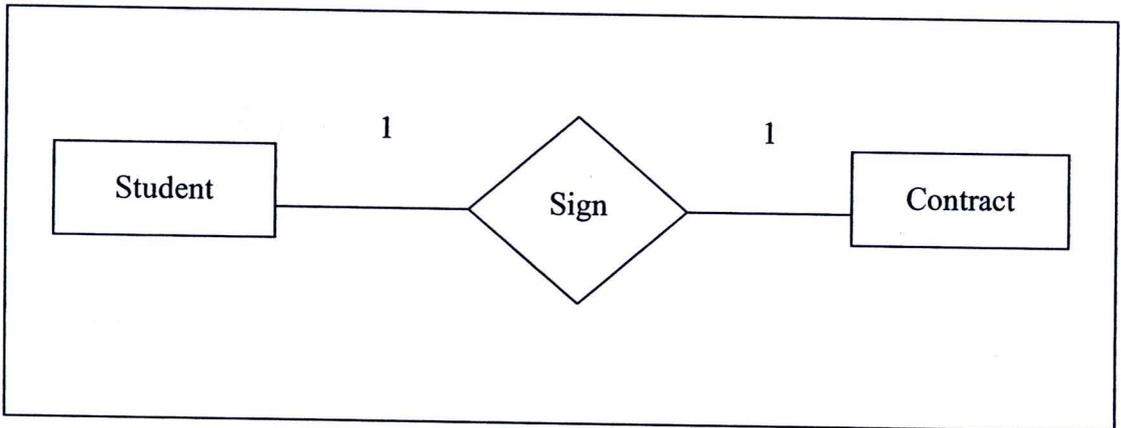


ภาพ 12 ตัวอย่างระดับชั้นของข้อความ

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย
 ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก [http://itd.htc.ac.th/st_it50/it5016/
 nidz/Web_Analyse/unit9.html](http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html)



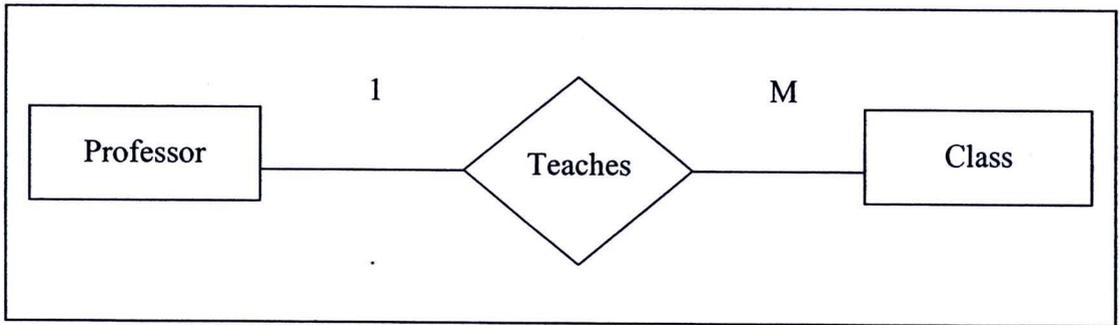
2. การระบุตำแหน่งความสัมพันธ์ระหว่าง Entity Connectivity ว่าเป็นแบบหนึ่งต่อหนึ่ง (one to one relationships) แบบหนึ่งต่อกลุ่ม (one to many relationships) หรือแบบกลุ่มต่อกลุ่ม (many to many relationships) โดยจะใช้ Connectivity เพื่อระบุตำแหน่ง 1, M หรือ N ไว้ข้างใดของ Entity ดังแสดงในภาพ 13



ภาพ 13 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดยปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

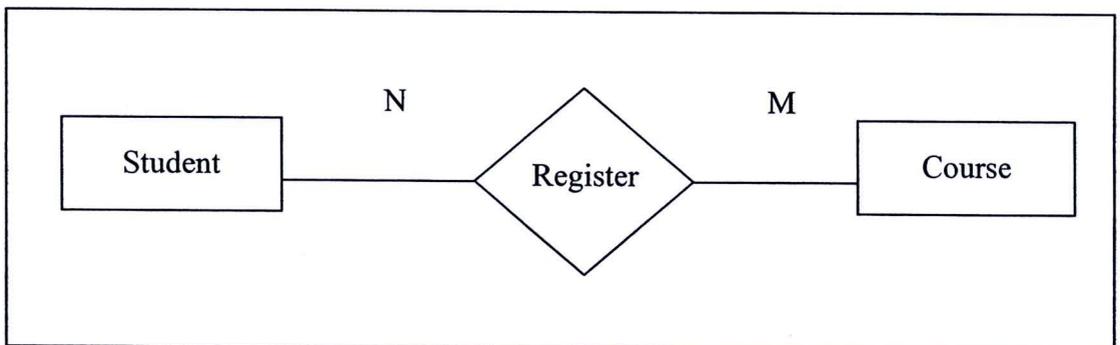
จากภาพ 13 แสดงความสัมพันธ์ระหว่างนักศึกษา กับสัญญาเงินกู้ โดยที่นักศึกษาหนึ่งคนทำสัญญาเงินกู้ได้เพียงครั้งเดียว สัญญาการกู้เงินแต่ละฉบับจะถูกลงชื่อได้จากนักศึกษาเพียงคนเดียวเท่านั้น ความสัมพันธ์การกู้เงินที่เชื่อมระหว่างนักศึกษาและสัญญาเงินกู้จึงเป็นแบบหนึ่งต่อหนึ่ง



ภาพ 14 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย
 ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

จากภาพ 14 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม จะประกอบด้วย Entity อาจารย์ กับ Entity กลุ่มเรียน หมายความว่า อาจารย์จะสอนได้หลายกลุ่มเรียน แต่ละกลุ่มเรียน จะมีอาจารย์สอนได้เพียงคนเดียว ไว้ด้าน Entity อาจารย์ และตัวอักษร M ไว้ด้าน Entity กลุ่มเรียน



ภาพ 15 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย
 ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

จากภาพ 15 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม ประกอบด้วย Entity นักเรียน กับ Entity วิชาเรียน โดยที่นักศึกษาแต่ละคนลงทะเบียนเรียนวิชาได้มากกว่า 1 วิชา แต่ละวิชามีนักศึกษาได้มากกว่า 1 คน ความสัมพันธ์ของการลงทะเบียนของนักศึกษา กับวิชาเป็นแบบ N : M

คีย์

คุณสมบัติหนึ่งที่สำคัญของความสัมพันธ์ก็คือ ความเป็นเอกลักษณ์ (uniqueness property) สิ่งที่ใช้กำหนดความเป็นเอกลักษณ์ของแถวในความสัมพันธ์เรียกว่า คีย์ (key) สามารถแบ่งออกเป็น 6 คีย์ ดังนี้ (ทบวงมหาวิทยาลัย, โครงการเครือข่ายสารสนเทศเพื่อพัฒนาการศึกษา, 2544)

1. คีย์หลัก (primary key) คือ คีย์ที่ใช้ในการอ้างอิงถึง Entity ในฐานข้อมูล การเลือกคีย์หลักสามารถเลือกได้จาก Record ใด ๆ ก็ได้ ที่ไม่มีโอกาสซ้ำซ้อนกันบนฐานข้อมูลนั้น

2. คีย์รอง (secondary key) คือ คีย์เดี่ยวหรือคีย์ผสม (single or composite key) ซึ่งเมื่อใช้ในการค้นหาข้อมูลจากความสัมพันธ์จะได้มากกว่าหนึ่งเรคคอร์ด ต่างจากคีย์หลักที่ทำให้ข้อมูลในตารางไม่ซ้ำกัน ดังนั้น คีย์รองจึงไม่จำเป็นจะต้องเป็นเอกลักษณ์

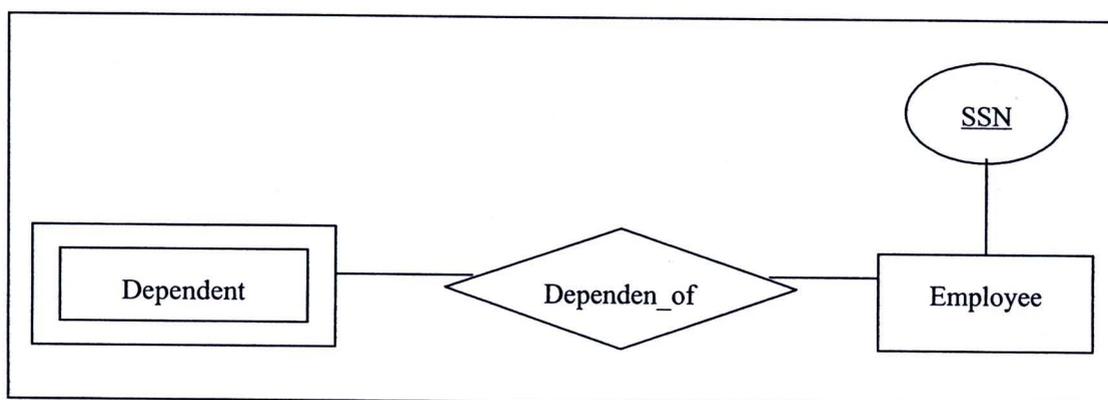
3. คีย์นอก (foreign key) คือ คีย์เดี่ยวหรือคีย์ผสม (single or composite key) ซึ่งปรากฏเป็นคีย์ทั่วไปของความสัมพันธ์หนึ่ง แต่ไปปรากฏเป็นอีกคีย์หลักในอีกความสัมพันธ์หนึ่ง คีย์นอกเป็นอีกคีย์หนึ่งที่มีความสำคัญมากในฐานข้อมูลเชิงสัมพันธ์ เนื่องจากเป็นตัวที่ใช้สร้างการเชื่อมต่อระหว่างความสัมพันธ์การเปลี่ยนแปลงค่าของคีย์นอกจะต้องอาศัยความระมัดระวังเป็นอย่างมาก เนื่องจากจะมีผลกระทบโดยตรงต่อข้อมูลในความสัมพันธ์อื่นที่มีการอ้างอิงถึงคีย์นอกตัวนี้ จึงมีกฎและเงื่อนไขที่บังคับใช้เพื่อให้ข้อมูลมีความถูกต้องอยู่เสมอ

การกำหนดค่าให้กับคีย์นอกของความสัมพันธ์ที่อ้างอิงถึงจะต้องกำหนดค่าของคีย์ให้อยู่ใน Domain เดียวกับความสัมพันธ์ที่คีย์นอกนั้นเป็นคีย์หลัก แต่คีย์นอกนั้นไม่จำเป็นจะต้องเป็นส่วนหนึ่งในคีย์หลักของความสัมพันธ์อื่น

4. ซุปเปอร์คีย์ (super key) คือ กลุ่มของ Attribute ที่สามารถนำไปใช้ในการค้นหาข้อมูลที่เป็นเอกลักษณ์ได้

5. คีย์แข่งขัน (candidate key) คือ ซุปเปอร์คีย์ (super key) และไม่มีกลุ่มย่อยของคีย์ใดในคีย์แข่งขันที่จะสามารถเป็นซุปเปอร์คีย์ได้อีก

บางครั้งอาจพบว่า Entity ที่มี Primary Key ประกอบจาก Primary Key ของ Entity Set อื่น ๆ นั้น คือ Entity ไม่มี Primary Key หรือ Attribute เพียงพอในการสร้าง Primary Key ได้ด้วยตนเอง เรียก Entity นี้ว่า Weak Entity Set ดังนั้น หากจะระบุถึง Entity นี้ได้ จะต้องผูกสัมพันธ์กับบาง Entity ผ่าน Primary Key เป็นของ Entity ที่สัมพันธ์กับ Weak Entity ที่มี Primary Key ว่าเป็น Strong Entity Set พบว่า Weak Entity นั้น จะต้องสัมพันธ์เกี่ยวข้องแบบ Total Participate กับ Strong Entity เสมอ ดังแสดงในภาพ 16



ภาพ 16 ความสัมพันธ์แบบกลุ่มต่อหนึ่ง

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย ปรัชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html

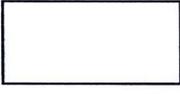
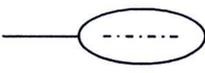
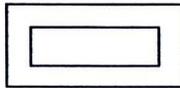
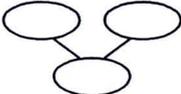
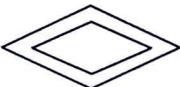
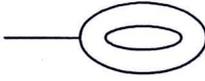
การแปลง E-R Model ให้อยู่ในรูปแบบโครงสร้างฐานข้อมูล

การแปลง E-R Model ให้อยู่ในรูปแบบโครงสร้างฐานข้อมูลหรือตารางของข้อมูล มีกฎดังนี้ (ปรัชญา ศิริภูรี, ม.ป.ป.)

1. การแปลง Entity ที่มีความสัมพันธ์แบบหนึ่งต่อหนึ่ง (one to one relationships) ไปเป็นตาราง โดยแทนที่หนึ่ง Entity เป็นหนึ่งตาราง Attribute แต่ละ Entity เป็น Field หรือคอลัมน์แต่ละตาราง

2. การแปลง Entity ที่มีความสัมพันธ์แบบหนึ่งต่อกลุ่ม (one to many relationships) ไปเป็นตาราง โดยด้าน Entity ที่เป็นตัวเลข 1 นั้น สามารถแปลงเป็นตารางได้ทันที Attribute ของ Entity นั้นจะเป็น Field ของตารางทันที ส่วนด้าน Entity ที่เป็นตัวอักษร M ให้แปลง Entity เป็นตาราง โดยมี Attribute ของ Entity ตัวมันเอง และนำคีย์หลักของ Entity ที่เป็นเลข 1 มาใส่ Field ของตารางนั้นด้วย

3. การแปลง Entity ที่มีความสัมพันธ์แบบกลุ่มต่อกลุ่ม (many to many relationships) ไปเป็นตาราง โดยสร้าง Entity กลาง (composite entity) และนำคีย์หลักของทั้งสองตาราง มาเป็นคีย์หลักของ Entity กลางด้วย ส่วน Entity ทั้งสองที่อยู่ระหว่าง Entity กลาง ก็แปลงเป็นตารางได้ โดยนำเอา Attribute ของแต่ละ Entity ไปเป็น Field (ทำตามกฎของความสัมพันธ์แบบหนึ่งต่อหนึ่ง)

สัญลักษณ์	ความหมาย	สัญลักษณ์	ความหมาย
	Entity Set		Discriminator Key Attribute
	Weak Entity Set		Composite Attribute
	Relationship Set		Derived Attribute
	Identifying Relationship set		Key Attribute
	Attribute		Multi Valued Attribute

ภาพ 17 สัญลักษณ์ของ E-R model

ที่มา. จาก เว็บไซต์เสนอผลการเรียนการสอนวิชาการวิเคราะห์และออกแบบระบบ, โดย ปรชญา ศิริภูรี, ม.ป.ป., ค้นเมื่อ 19 สิงหาคม 2552, จาก http://itd.htc.ac.th/st_it50/it5016/nidz/Web_Analyse/unit9.html



ภาษา UML

ภาษา UML ย่อมาจากคำว่า Unified Modeling Language คือ สัญลักษณ์อันเป็นหนึ่งเดียวกัน ที่ใช้อธิบาย แสดงรายละเอียด จำลองการสร้าง และจัดการกับเอกสารต่าง ๆ ในระบบการทำงานจริง เพื่อให้การออกแบบซอฟต์แวร์ที่แทนระบบการทำงานจริงนั้นทำได้โดยง่าย และปรับปรุงวิธีการทำงานที่มีอยู่เดิมให้ดียิ่งขึ้น UML มักใช้เป็นการอธิบาย และนำเสนอแนวความคิดของการเขียนโปรแกรมเชิงวัตถุ ก่อนนำไปเขียนโปรแกรมจริง (วิกิพีเดีย สารานุกรมเสรี, 2553ข)

บุรินทร์ รุจจนพันธุ์ (ม.ป.ป.-ข) กล่าวว่า UML เป็นเครื่องมือที่มีความหลากหลาย ในการแสดงแบบซอฟต์แวร์ เป็นโมเดลมาตรฐานที่ใช้หลักการออกแบบ Object Oriented Programming (OOP) รูปแบบของภาษามี Notation เป็นสัญลักษณ์สำหรับสื่อความหมาย มีกฎระเบียบที่มีความหมายต่อการเขียนโปรแกรม (coding) ดังนั้น การใช้ UML จะต้องทราบความหมายของ Notation เช่น Generalize, Association, Dependency, Class และ Package สิ่งเหล่านี้มีความจำเป็นต่อการตีความการออกแบบก่อนนำไป Implement ระบบงานจริง

ปัจจุบันมีเครื่องมือมากมายที่สามารถแปลง UML Model เป็น Code ภาษาต่าง ๆ เช่น ภาษา Java, Power Builder และ VB เป็นต้น UML เป็น Visual Modeling ซึ่งจะทำให้หน้าที่ในการแสดงโครงสร้าง การทำงานของซอฟต์แวร์ให้ออกมาในโมเดลที่สามารถมองเห็นได้ โดยการสื่อให้ออกมาในรูปของ Diagram รูปภาพ ด้วยวิธีการแบบนี้จะทำให้ Model, Implement และ Coding มีความสอดคล้องกันไปไปในแนวทางเดียวกัน (ไอทีเมโลดิดอทคอม, ม.ป.ป.-ข)

ประโยชน์ของ UML

UML ถูกนำมาใช้ในการพัฒนาซอฟต์แวร์อย่างกว้างขวางและทำให้การทำงานมีคุณภาพ ดังนี้ (ไอทีเมโลดิดอทคอม, ม.ป.ป.-ข)

1. ช่วยลดระยะเวลาในการพัฒนาระบบงาน (shortest development life cycle)
2. ช่วยเพิ่มความสามารถในการทำงาน (increase productivity)
3. ช่วยเพิ่มคุณภาพของระบบงาน (improve software quality)
4. รองรับระบบงานเดิม (support legacy system)
5. ช่วยในการสื่อสารระหว่างทีมผู้พัฒนาระบบงาน (improve team connectivity)

กระบวนการในการพัฒนาโครงการ มีวัตถุประสงค์เพื่อสนับสนุนการทำงาน หรือความต้องการของผู้ใช้เป็นหลัก และผู้ใช้ไม่ได้หมายถึงคนเพียงอย่างเดียว อาจหมายถึงระบบอื่นที่อยู่ภายนอกระบบ (some one or something such as other system out side the system)

ผู้ใช้อจะมีการกระทำหรือกิจกรรมต่อระบบ เช่น การสอดบัตรพลาสติกลงในเครื่องอ่านบัตร การกดปุ่มที่ตอบสนองทางหน้าจอของเครื่อง ATM (automatic teller machine) และการรับเงินที่เครื่องส่งออกมา

UML ประกอบด้วยโมเดลจำนวนหนึ่งที่น่าเอามาใช้ร่วมกัน เพื่อการดำเนินงานโครงการซอฟต์แวร์ โมเดลดังกล่าวมีทั้งหมด 9 โมเดล ดังนี้ (บรรจง หะรังษี และญาณวรรณ สันธุภิญโญ, 2542)

1. Use Case Diagram
2. Sequence Diagram
3. Class Diagram
4. Activity Diagram
5. Collaboration Diagram
6. Component Diagram
7. Deployment Diagram
8. Object Diagram
9. State Chart Diagram

จากโมเดลต่าง ๆ ของ UML ที่แสดง มีโมเดล 3 คือ (1) Use Case Diagram (2) Sequence Diagram และ (3) Class Diagram เป็นโมเดลพื้นฐานที่ใช้ในระบบงานทั่วไป

Use Case Diagram

Use Case Diagram คือ การอธิบายฟังก์ชันการทำงานต่าง ๆ ของระบบ ในกรณีของการใช้งานที่เกิดจากมุมมองของผู้ใช้ระบบ การสร้าง Use Case Diagram จะต้องมีการกลุ่มคนเป้าหมาย ซึ่งเรียกว่า Actor หรือผู้ใช้ระบบ โดย Actor จะอยู่ภายนอกระบบ Actor อาจจะขอให้ระบบทำอะไรสักอย่างหนึ่ง หรือในทางกลับกัน ระบบอาจจะขอให้ Actor นั้นทำอะไรให้สักอย่างหนึ่ง ดังนั้น Actor ยังหมายความรวมถึงสิ่งอื่น ๆ ที่อยู่นอกระบบ ซึ่งสามารถติดต่อกับระบบได้ เช่น ระบบสินค้าคงคลัง ระบบบัญชี ระบบพัสดุ เป็นต้น (บรรจง หะรังษี และญาณวรรณ สินธุภิญโญ, 2542)

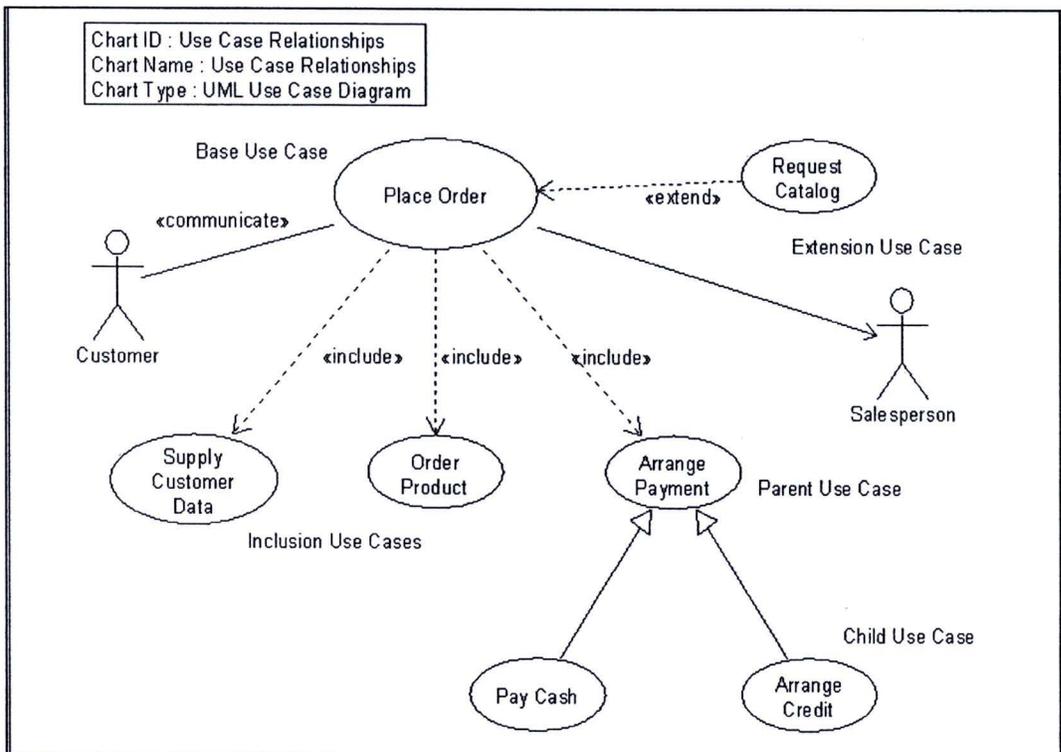
องค์ประกอบ Use Case Diagram

องค์ประกอบในการออกแบบ Use Case Diagram แบ่งออกเป็น 9 ข้อ ดังนี้ (บรรจง หะรังษี และญาณวรรณ สินธุภิญโญ, 2542)

1. ชื่อของ Use Case การกำหนดชื่อของ Use Case นั้น ชื่อที่ใช้ควรสั้นและกระชับ แต่สื่อความหมายที่ดีของ Use Case ตัวนั้น ๆ
2. ภาพรวมของการทำงาน (overview) เป็นการอธิบายภาพการทำงานของ Use Case โดยรวมว่าทำอะไร ส่วนนี้ควรมีความยาวไม่เกิน 5 บรรทัด
3. Actor หลัก (primary actor) เป็นการอธิบายว่าใครคือผู้ใช้ของ Use Case Diagram ตัวนี้
4. Actor รอง (secondary actor) เป็นการอธิบายว่าใครคือผู้ใช้อื่น ๆ นอกเหนือจาก Actor หลัก ที่เป็นผู้ใช้ Use Case ตัวนี้
5. จุดเริ่มต้น (starting point) เป็นการอธิบายเงื่อนไขหรือข้อกำหนดต่าง ๆ ณ จุดเริ่มต้นของ Use Case โดยเงื่อนไขเหล่านี้ โดยปกติแล้วจะต้อง “มี” หรือ “เป็น” อย่างนั้นก่อนการทำงานของ Use Case นี้
6. จุดสิ้นสุด (end point) คล้ายกับจุดเริ่มต้นเป็นการอธิบายเงื่อนไขหรือข้อกำหนดต่าง ๆ ณ จุดสิ้นสุดการทำงานของ Use Case Diagram ตัวนี้
7. การทำงานของ Use Case Flow of Events อธิบายขั้นตอนการทำงานของ Use Case นี้

8. การทำงานของ Use Case Alternative flow of Events เมื่อมีปัญหาเกิดขึ้น การทำงานของ Use Case ในกรณีเมื่อมีปัญหาใด ๆ ก็ตาม ที่ทำให้การทำงานตามปกติของ Use Case ไม่สามารถดำเนินต่อไปได้ จะมีการส่งต่อการทำงานของ Use Case จากขั้นตอนการทำงานปกติ (ในส่วนการทำงานของ Use Case) มาที่ส่วนนี้ของ Use Case

9. ผลของการทำงานของ Use Case Measurable Result ผลการทำงานของ Use Case แสดงให้เห็นว่า เมื่อเสร็จสิ้นการทำงานของ Use Case นี้แล้ว จะมีผลพวงอะไรเกิดขึ้นตามมา ดังแสดงในภาพ 18



ภาพ 18 ตัวอย่าง Use case diagram และความสัมพันธ์

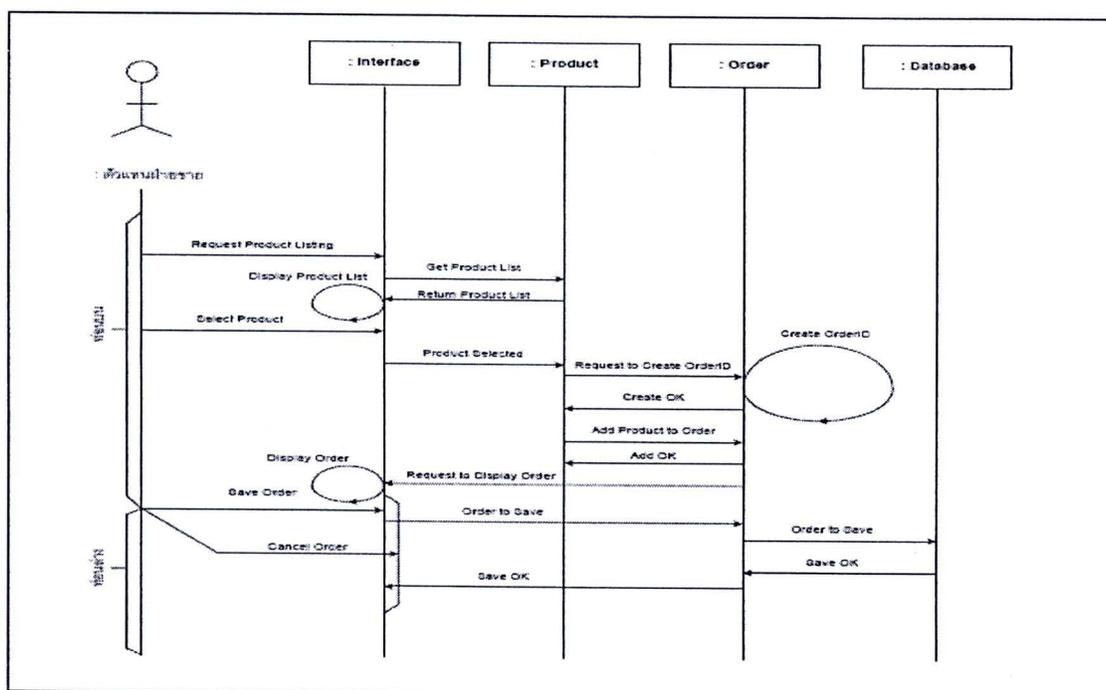
ที่มา. จาก *Unified Modeling Language*, โดย ไอทีเมโลดี้คอม, ม.ป.ป.-ข, ค้นเมื่อ 11 มกราคม 2553, จาก <http://www.itmelody.com/tu/uml1.htm#usecasedrive>

Sequence Diagram

บรรจง หารังษี และญาณวรรณ สิริสุทธิโย (2542) อธิบายว่า Sequence Diagram คือ Diagram ที่แสดงลำดับขั้นตอน (sequence) การทำงานภายในของ Use Case ตัวหนึ่ง ปกติแล้ว Use Case จะไม่สามารถมองเห็นลำดับขั้นตอนการทำงานภายในของ Use Case ตัวนั้นได้ ตัว Sequence Diagram เท่านั้นที่สามารถมองเห็นลำดับขั้นตอนการทำงานภายในของ Use Case ตัวนั้นได้

นอกจากนี้ Sequence Diagram ยังแสดงให้เห็นถึงการติดต่อกันระหว่าง Object ต่าง ๆ ของ Use Case นั้น ส่วน Object และ Actor ของ Use Case การติดต่อกันนั้น จะทำให้มีข้อความ (message) วิ่งไปมาใน Diagram

บรรจง หารังษี และญาณวรรณ สิริสุทธิโย (2542) กล่าวว่า Sequence Diagram ถูกสร้างขึ้นมาจาก Use Case ซึ่งตัว Use Case นั้นไม่ได้บอกรายละเอียดของการทำงานภายในว่าเป็นอย่างไร Sequence Diagram จะเป็นตัวบอกว่ามีรายละเอียดการทำงานภายในอย่างไร และรายละเอียดการทำงานภายในเหล่านั้นก็คือ ข้อความที่วิ่งไปมาใน Diagram นั้น ดังแสดงในภาพ 19



ภาพ 19 Sequence diagram ของ Create order

ที่มา. จาก *Unified Modeling Language*, โดย ไอทีเมโลดี้คอม, ม.ป.ป.-ข, ค้นเมื่อ 11 มกราคม 2553, จาก <http://www.itmelody.com/tu/uml1.htm#usecase-drive>

Class Diagram

คำว่า Class คือ Set ของ Object ที่มีคุณสมบัติ (properties) และพฤติกรรม (methods) ส่วนคำว่า Class Diagram คือ ภาพที่ประกอบไปด้วย Class ต่าง ๆ ที่มาประกอบหรือรวมตัวกันกลายเป็นระบบหรือซอฟต์แวร์ตัวหนึ่ง เป็นต้น ยกตัวอย่างเช่น Class: Interface, Product, Order, and Database เป็นต้น (บรรจง หารังษี และญาณวรรณ สินธุภิญโญ, 2542)

รูปแบบของ Class Diagram

ขั้นตอนการออกแบบ Class Diagram ภายหลังจากการออกแบบ Use Case Diagram และ Sequence Diagram มีดังนี้ (บรรจง หารังษี และญาณวรรณ สินธุภิญโญ, 2542)

1. Class Properties ในส่วนของ Class Properties คุณสมบัติที่เกี่ยวข้องกับ Class Order (ดูภาพ 20 ประกอบ)
2. Class Methods สิ่งสุดท้ายที่ต้องการ คือ Class Methods ซึ่งจะได้มาจาก Sequence Diagram สิ่งที่ต้องทราบเบื้องต้นก็คือ ข้อความแต่ละข้อความใน Sequence Diagram สามารถเทียบเท่ากับ Method 1 Method
3. Method ของ Class Order จาก Sequence Diagram อื่น ๆ
4. Class อื่น ๆ นอกจาก Class Order

Order
Order ID
Order Date
Required Date
Ship Via
Ship Date
Ship Address
Unit Price
Quantity
วางเปล่า

ภาพ 20 ตัวอย่าง Class order

ที่มา. จาก *Unified Modeling Language*, โดย ไอทีเมโลดี้คอตคอม, ม.ป.ป.-ข, กันยายน 11 มกราคม 2553, จาก <http://www.itmelody.com/tu/uml1.htm#usecasedrive>

จากภาพ 20 จะเห็นได้ว่า จากความสัมพันธ์ของ Use Case, Sequence Diagram และ Class Diagram สามารถสรุปภาพโดยรวมของกระบวนการในการทำงานทั้งหมดของ UML ดังนี้

1. การหา Use Case ทั้งหมดในระบบ
2. การสร้าง Sequence Diagram ที่สอดคล้องกับ Use Case ตัวหนึ่ง ๆ

3. การสร้าง Class Diagram Class Diagram จะประกอบไปด้วย Class ทั้งหมดในระบบ ส่วน Class หนึ่ง Class ใน Diagram นั้น โดยปกติจะถูกสร้างขึ้นมาจากหลาย Use Case และหลาย Sequence Diagram โดย Use Case และ Sequence Diagram ดังกล่าวจะต้องมีความเกี่ยวข้องกับ Class ตัวนั้น

การพัฒนาโปรแกรมด้วยภาษา PHP ร่วมกับ AJAX

ภาษา PHP

บุรินทร์ รุจจนพันธุ์ (ม.ป.ป.-ก) ได้ให้ความหมายของคำว่า ภาษา PHP หมายถึง ภาษาคอมพิวเตอร์ประเภท Open Source เพื่อใช้สำหรับพัฒนาเว็บเพจแบบ Dynamic เมื่อเครื่องบริการได้รับคำร้องจากผู้ใช้ ก็จะส่งให้กับตัวแปลภาษาทำหน้าที่ประมวลผล และส่งข้อมูลกลับไปยังเครื่องของผู้ใช้ที่ร้องขอในรูปแบบของ HTML ภาพ หรือแฟ้มดิจิทัลอื่นใด ลักษณะของภาษามีรากฐานคำสั่งมาจากภาษา C เป็นภาษาที่สามารถพัฒนาให้ใช้งานแบบโต้ตอบกับผู้ใช้ได้

ภาษา PHP มีการทำงานแบบ Server-side Script จึงต้องมีเครื่องบริการ (server) ที่ทำหน้าที่บริการแปลภาษา และส่งผลให้กับเครื่องผู้ใช้ (client) ที่ร้องขอ ด้วยการส่งคำร้องเข้ามายังเครื่องบริการ คำว่า PHP ย่อมาจาก Personal Home Page แต่พัฒนาให้มีประสิทธิภาพยิ่งขึ้น จึงเปลี่ยนเป็น Professional Home Page

PHP เป็นภาษาหนึ่งในภาษา Script สำหรับการประมวลผล Web Server ที่มีผู้ใช้งานมากที่สุด คำสั่งต่าง ๆ นั้นถูกเก็บอยู่ในไฟล์ที่เรียกว่า Script เวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษา Script ได้แก่ JavaScript และ PERL เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษา Script แบบอื่น ๆ คือ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้น จึงกล่าวได้ว่า PHP เป็นภาษาคอมพิวเตอร์ในลักษณะ HTML Embedded Scripting Language และจัดเป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้สามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพ และมีลูกเล่นมากขึ้น (ไทยครีเอท, 2551) มีการตอบโต้ได้อย่างรวดเร็ว เนื่องจาก PHP เป็นภาษาที่เรียนรู้ได้ง่าย

และไม่มีหลักเกณฑ์ที่ซับซ้อน จึงทำให้ PHP เป็นภาษาที่น่าสนใจมากกว่าภาษาหนึ่ง ทั้งนี้ PHP เป็นภาษาที่ถูกออกแบบและสร้างขึ้นด้วยจุดมุ่งหมายหลัก คือ ทำให้คนทั่วไปสามารถสร้างเว็บไซต์ได้โดยที่ไม่มีความรู้เรื่องคอมพิวเตอร์มากนัก และสามารถทำได้ในระยะเวลาอันสั้น เพราะเป็นภาษาที่ไม่ซับซ้อน

เนื่องจาก PHP ไม่ได้เป็นส่วนหนึ่งของตัวเครื่อง Web Server ดังนั้น การใช้ PHP ต้องดูก่อนว่าเครื่อง Web Server นั้นสามารถใช้ Script PHP ได้หรือไม่ ตัวอย่างเช่น PHP สามารถใช้ได้กับ Apache Web Server และ Personal Web Server (PWP) สำหรับระบบปฏิบัติการ Windows 95/98/NT

กรณีของ Apache Web Server นี้ สามารถใช้ PHP ได้ 2 รูปแบบ คือ ในลักษณะของ Common Gateway Interface (CGI) และ Apache Module ความแตกต่างอยู่ที่ว่า ถ้าใช้ PHP เป็นแบบ Module PHP จะเป็นส่วนหนึ่งของ Apache หรือเป็นส่วนขยายในการทำงานนั่นเอง ซึ่งจะทำงานได้เร็วกว่าแบบที่เป็น CGI เพราะว่าถ้าเป็น CGI แล้วตัวแปลชุดคำสั่งของ PHP ถือว่าเป็นแค่โปรแกรมภายนอก ซึ่ง Apache จะต้องเรียกขึ้นมาทำงานทุกครั้งที่ต้องการใช้ PHP ดังนั้น ถ้ามองถึงเรื่องของประสิทธิภาพในการทำงาน การใช้ PHP แบบที่เป็น Module หนึ่งของ Apache จะทำงานได้มีประสิทธิภาพมากกว่า (บันทึกของคนรัก, 2553)

ลักษณะเด่นของภาษา PHP

ภาษาที่ใช้ในการพัฒนาเว็บเพจนั้นมีหลากหลาย สำหรับภาษา PHP มีลักษณะเด่นที่ทำให้ผู้ใช้เลือกใช้ภาษา PHP คือ (ไทยครีเอท, 2551)

1. เป็น Freeware หรือใช้ได้ฟรี
2. ภาษา PHP เป็น โปรแกรมที่ทำงานฝั่งเครื่องแม่ข่าย (server) ดังนั้น

ขีดความสามารถจึงไม่จำกัด

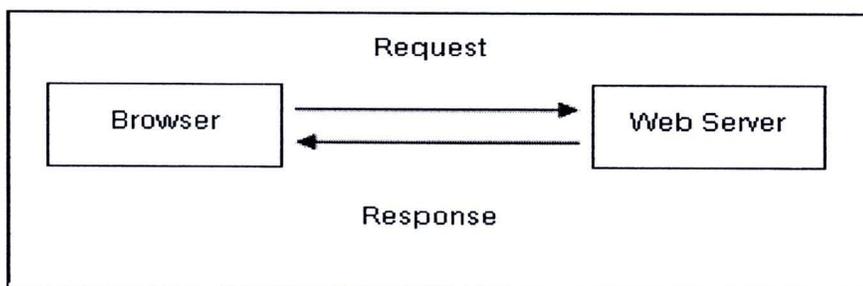
3. สามารถทำงานได้ทุกระบบปฏิบัติการ อาทิ UNIX, Linux และ Windows เป็นต้น

4. เรียนรู้ได้ง่าย เนื่องจาก PHP ถูกฝังเข้าไปในภาษา HTML ซึ่งใช้โครงสร้างและไวยากรณ์ภาษาแบบง่าย

5. สามารถทำงานได้อย่างรวดเร็วและมีประสิทธิภาพ โดยเฉพาะเมื่อใช้กับ Apache Web Server เพราะไม่จำเป็นต้องอาศัย โปรแกรมจากภายนอก
6. สามารถใช้งานร่วมกับ XML ได้ทันที
7. ใช้กับระบบเพิ่มข้อมูลได้
8. ใช้กับข้อมูลตัวอักษรได้อย่างมีประสิทธิภาพ
9. ใช้กับโครงสร้างข้อมูลแบบ Scalar, Array และ Associative Array ได้
10. ใช้กับการประมวลผลภาพได้

สถาปัตยกรรมฐานข้อมูลเว็บ

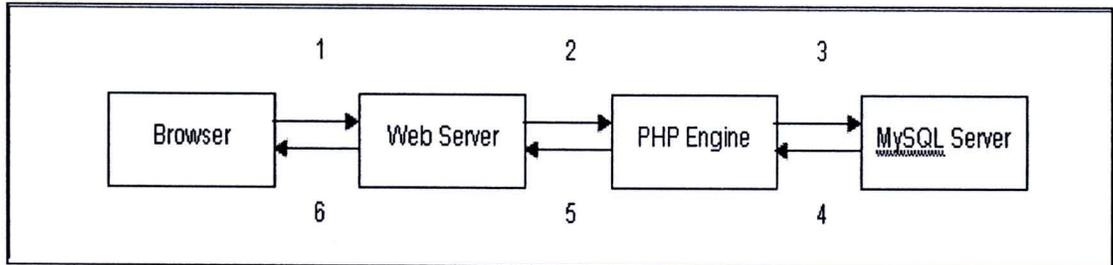
ปฏิบัติการพื้นฐานของแม่ข่ายเว็บตามการแสดง ประกอบด้วย 2 Objects คือ (1) Web Browser และ (2) Web Server การเชื่อมต่อด้านคมนาคมมีความต้องการระหว่าง Objects ซึ่ง Web Browser จะทำคำขอไปยังแม่ข่าย แล้วแม่ข่ายส่งกลับการตอบสนอง สถาปัตยกรรมนี้เหมาะสมกับแม่ข่ายส่งผ่าน Static Page สถาปัตยกรรมส่งผ่านฐานข้อมูลจากเว็บมีความซับซ้อนมากกว่าเล็กน้อย (โลกกว้างแห่งเทคโนโลยีสารสนเทศ, ม.ป.ป.)



ภาพ 21 การสื่อสารระหว่าง Browser กับ Web server

ที่มา. จาก การออกแบบฐานข้อมูลเว็บ, โดย โลกกว้างแห่งเทคโนโลยีสารสนเทศ, ม.ป.ป.,
 ค้นเมื่อ 28 กันยายน 2552, จาก [http://www.widebase.net/database/mysql/mysqltutorial/
 mysqltutor0103.shtml](http://www.widebase.net/database/mysql/mysqltutorial/mysqltutor0103.shtml)

โปรแกรมประยุกต์ฐานข้อมูลเว็บสามารถทำตามโครงสร้างฐานข้อมูลเว็บ
 ดังแสดงในภาพ 22



ภาพ 22 สถาปัตยกรรมพื้นฐานประกอบด้วย Browser, Web server, PHP engine
 และ MySQL server

ที่มา. จาก การออกแบบฐานข้อมูลเว็บ, โดย โลกกว้างแห่งเทคโนโลยีสารสนเทศ, ม.ป.ป.,
 ค้นเมื่อ 28 กันยายน 2552, จาก [http://www.widebase.net/database/mysql/mysqltutorial/
 mysqltutor0103.shtml](http://www.widebase.net/database/mysql/mysqltutorial/mysqltutor0103.shtml)

จากภาพ 22 มีการเรียงหมายเลขตามแบบแผน Transaction ของฐานข้อมูลเว็บ
 โดยมีขั้นตอนในสำรวจ ดังนี้ (โลกกว้างแห่งเทคโนโลยีสารสนเทศ, ม.ป.ป.)

1. ผู้ใช้ Web Browser ทำคำขอ HTTP สำหรับเว็บเพจที่ต้องการ ด้วยการใช้ฟอร์ม HTML ผลลัพธ์การค้นหา คือ การเรียก results.php
2. Web Server รับคำขอสำหรับ results.php ดึงไฟล์และส่งผ่านไปยัง PHP Engine สำหรับการประมวลผล
3. PHP Engine เริ่มการกระจาย Script ภายใน Script คือ คำสั่งการเชื่อมต่อกับฐานข้อมูลและการประมวลผล Query PHP เปิดการเชื่อมต่อกับ MySQL Server และส่ง Query ที่เหมาะสม
4. MySQL Server รับ Query สำหรับฐานข้อมูล และประมวลผล จากนั้นส่งผลลัพธ์กลับไปยัง PHP Engine ในกรณีนี้ คือ รายชื่อหนังสือ

5. PHP Engine เสร็จสิ้นการเรียกใช้ Script ซึ่งจะเกี่ยวข้องกับการจัดรูปแบบผลลัพธ์กลับไปยัง PHP Engine จากนั้นส่งออกผลลัพธ์ HTML ไปยัง Web Server

6. Web Server ส่งผ่าน HTML กลับไปยัง Browser ที่ผู้ใช้สามารถดูรายการหนังสือตามคำขอ

กระบวนการมีพื้นฐานเหมือนกับ Scripting Engine และฐานแม่ข่ายทั่วไป ส่วนมากซอฟต์แวร์ Web Server ในที่นี้คือ PHP Engine และฐานข้อมูลแม่ข่ายทำงานบนเครื่องเดียวกัน อย่างไรก็ตาม ฐานข้อมูลแม่ข่ายอาจทำงานบนคนละเครื่อง การทำเช่นนี้ด้วยเหตุผลด้านความปลอดภัย การเพิ่มสมรรถนะ หรือการกระจายภาระจากมุมมองการพัฒนาที่จะมีการทำงานเหมือนกัน แต่อาจจะให้สมรรถนะดีขึ้น

ภาษา Java และภาษา JavaScript

จากการศึกษาของบรูนิทส์ รุจจนพันธุ์ (ม.ป.ป.-ก) ให้ความหมายของคำว่า ภาษา Java คือ ภาษาคอมพิวเตอร์ที่ถูกพัฒนาขึ้นโดยบริษัท ซัน ไมโครซิสเต็มส์ จำกัด เป็นภาษาสำหรับเขียน โปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming--OOP) โปรแกรมที่เขียนขึ้นถูกสร้างภายใน Class ดังนั้น Class คือ ที่เก็บ Method หรือพฤติกรรม (behavior) ซึ่งมีสถานะ (state) รูปพรรณ (identity) และประจำพฤติกรรม (behavior)

ภาษา JavaScript คือ ภาษาโปรแกรมคล้ายภาษา C ถูกใช้ร่วมกับภาษา HTML ในการพัฒนาเว็บเพจและประมวลผลในเครื่องของผู้ใช้ ช่วยให้การนำเสนอเป็นแบบโต้ตอบกับผู้ใช้ได้ในระดับหนึ่ง

ภาษา JavaScript คือ ภาษาโปรแกรมที่มีโครงสร้างคล้ายภาษา C ทำหน้าที่แปลความหมาย และดำเนินการทีละคำสั่ง ภาษานี้มีชื่อเดิมว่า LiveScript ถูกพัฒนาโดย Netscape Navigator เพื่อช่วยให้เว็บเพจสามารถแสดงเนื้อหาที่มีการเปลี่ยนแปลงได้ตามเงื่อนไขหรือสภาพแวดล้อมที่แตกต่างกัน หรือโต้ตอบกับผู้ใช้ได้มากขึ้น เพราะภาษา HTML ที่เป็นภาษาพื้นฐานของเว็บเพจ ทำได้เพียงแสดงข้อมูลแบบคงที่ (static display)

JavaScript ถูกออกแบบมาเพื่อใช้เป็นส่วนเพิ่มขยายในภาษา HTML โดยเฉพาะช่วยให้สามารถควบคุมเว็บเพจได้อย่างง่ายดาย เหมาะกับการทำงานอย่างรวดเร็ว และเน้นที่ความถูกต้องเป็นสำคัญ (คลังความรู้จาวาภาษาไทย, ม.ป.ป.)

การเขียน JavaScript อาจเขียนรวมอยู่ในไฟล์เดียวกับ HTML ได้ ซึ่งแตกต่างจากการเขียนโปรแกรมภาษา Java ที่ต้องเขียนแยกออกเป็นไฟล์ต่างหาก ไม่สามารถเขียนรวมอยู่ในไฟล์เดียวกับ HTML ได้ วิธีการเขียน JavaScript เพื่อสั่งให้เว็บเพจทำงานมีอยู่ด้วยกัน 2 วิธี ดังนี้ (วิทยาลัยเทคนิคแพร่, แผนกเทคโนโลยีสารสนเทศ, ม.ป.ป.)

1. เขียนด้วยชุดคำสั่งและฟังก์ชันของ JavaScript
2. เขียนตามเหตุการณ์ที่เกิดขึ้น การใช้งานจากชุดคำสั่งของ HTML เมื่อเริ่มใช้งาน

โปรแกรม Navigator จะอ่านข้อมูลจากส่วนบนของเว็บเพจ HTML และทำงานไปตามลำดับจากบนลงล่าง (top-down) โดยเริ่มที่ส่วน < HEAD > . . . < /HEAD > ก่อน จากนั้นจึงทำงานในส่วน < BODY > . . . < /BODY > เป็นลำดับต่อมา การทำงานของ JavaScript คู่มือไม่แตกต่างไปจาก HTML เท่าใดนัก แต่ HTML จะวาง Lay Out โครงสร้างของ Object ภายใน และส่วนเชื่อมโยงกับเว็บเพจเท่านั้น ในขณะที่ JavaScript สามารถเพิ่มเติมในส่วนของการเขียนโปรแกรม และ Logic เข้าไปเริ่มที่ส่วน Tract บนก่อน จากนั้นจึงทำงานในส่วน Tract ล่าง เป็นลำดับต่อมา นอกเหนือจากช่วยลดภาระการทำงานของ Server แล้ว JavaScript ยังมีความสามารถที่เหนือกว่าภาษาสำหรับการตรวจสอบความถูกต้องในฟอร์มทั่วไป JavaScript เป็นเสมือนกาวที่ทาง Client ใช้ในการเก็บส่วนประกอบของเว็บเพจเข้าด้วยกัน โดยการใช้ Logic ขึ้นพื้นฐาน ที่ตัดสินใจได้ว่า จะใช้ Browser หรือ Plug In ตัวใด

JavaScript กลายเป็นภาษากลางสำหรับควบคุมและติดต่อสื่อสารระหว่าง Object อย่าง Plug In, Layer, Frame, From, Style Sheet และ Applet ซึ่งได้รับการยอมรับมากขึ้น ยิ่งกว่าภาษา HTML เสียอีก

การทำงานของ JavaScript คู่มือไม่แตกต่างไปจาก HTML เท่าใดนัก แต่ HTML จะวาง Lay Out โครงสร้างของ Object ภายในและส่วนเชื่อมโยงกับเว็บเพจเท่านั้น ในขณะที่ JavaScript สามารถเพิ่มเติมในส่วนของการเขียนโปรแกรมและ Logic เข้าไปได้

ลักษณะการทำงานของ JavaScript

JavaScript เป็นภาษา Script เชิงวัตถุ หรือเรียกว่า Object Oriented Programming ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนเอกสารด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ ทำงานร่วมกับภาษา HTML และภาษา Java ได้ทั้งทางฝั่ง Client และทางฝั่ง Server โดยมีลักษณะการทำงาน ดังนี้ (นาโนซอฟต์แวร์, 2548)

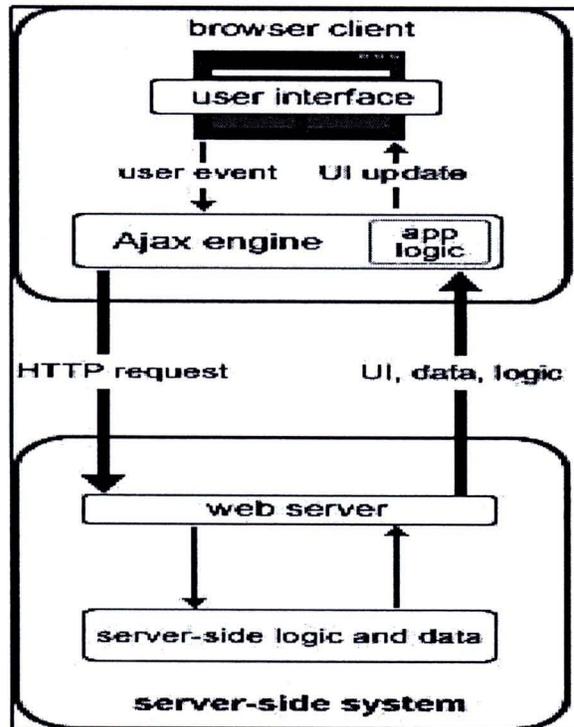
1. Navigator JavaScript เป็น Client-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่ง Client (หมายถึง ฝั่งเครื่องคอมพิวเตอร์ของผู้ใช้ ไม่ว่าจะเป็นเครื่อง PC เครื่อง Macintosh หรืออื่น ๆ) จึงมีความเหมาะสมต่อการใช้งานของผู้ใช้ทั่วไปส่วนใหญ่

2. Live Wire JavaScript เป็น Server-Side JavaScript ซึ่งหมายถึง JavaScript ที่ถูกแปลทางฝั่ง Server (หมายถึง ฝั่งเครื่องคอมพิวเตอร์ของผู้ให้บริการเว็บ โดยอาจจะ เป็นเครื่องของชั้น ซิลิคอมกราฟิกส์ หรืออื่น ๆ) สามารถใช้ได้เฉพาะกับ LiveWire ของ เน็ตสเคป โดยตรง

เทคโนโลยี AJAX

AJAX เป็นชุดของเทคโนโลยีต่าง ๆ ย่อมาจากคำว่า Asynchronous JavaScript and XML หมายถึง การทำงานร่วมกันของ JavaScript และ XML แบบ Asynchronous โดยมีหลักการการทำงาน 2 ประเด็น คือ การปรับปรุงหน้าจอ (update) แบบบางส่วน และการติดต่อสื่อสารกับ Server โดยใช้หลักการ Asynchronous ทำให้ผู้ใช้ไม่ต้องหยุดการทำงานเพื่อรอการประมวลผลจาก Server รวมถึงการ Load และการ Reface หน้าจอของ Browser ทางฝั่ง Client มีการใช้ AJAX โดยการเพิ่ม Layer ระหว่าง User Browser กับ Server ทำให้ผู้ใช้สามารถทำงานได้โดยไม่ต้องรอให้ Client ติดต่อไปยัง Server รวมถึงการ Load และการ Reface หน้าจอทั้งหมดด้วย ดังนั้น ผู้ใช้สามารถใช้งาน Application ได้อย่างมีประสิทธิภาพมากขึ้น ทำให้ผู้ใช้สามารถทำงานได้โดยไม่ต้องหยุดรอการประมวลผลของ Server ในส่วน Browser ทางฝั่ง Client มีการเพิ่มชั้นระหว่าง ผู้ใช้ Browser ของผู้ใช้กับ Server ทำให้ผู้ใช้งานทำงานได้โดยไม่ต้องรอให้ Browser ทางฝั่ง Client ติดต่อไปยัง Server (ชรินทร์ญา กล้าแข็ง, 2551)

AJAX จึงไม่ใช่เทคโนโลยีในตัวของมันเอง แต่เป็นการนำเทคโนโลยีหลาย ๆ ตัวมารวมกัน เช่น JavaScript, DHTML, XML, CSS, DOM และ XMLHttpRequest



ภาพ 23 โครงสร้างการทำงานของ AJAX

ที่มา. จาก *AJAX (Asynchronous JavaScript and XML)*, โดย ชรินทร์ญา กล้าแข็ง, 2551, <http://wiki.nectec.or.th/giti/Knowledge/Ajax>

เทคนิคของ AJAX

เทคนิค AJAX เป็นการผสมผสานกันของเทคโนโลยีในปัจจุบันหลาย ๆ ตัวเข้ามารวมกัน ดังนี้ (วิฑวัฒน์ วะสุรี, 2550)

1. XHTML (หรือ HTML) และ CSS สำหรับตกแต่งและจัดระเบียบข้อมูลในส่วนแสดงผล
2. DOM และ JavaScript หรือ J-Script เป็น Client-side Scripting Language เอาไว้แสดงผลแบบ Dynamic และจัดการตอบสนองกับการแสดงผล

3. XMLHTTP Request เป็น Object ที่ใช้สำหรับแลกเปลี่ยนข้อมูลกับ Web Server แบบไม่ต่อเนื่องกัน (asynchronous) ในบางสถานการณ์ Object ประเภท I-Frame จะถูกใช้แทนการใช้ XMLHTTP Request

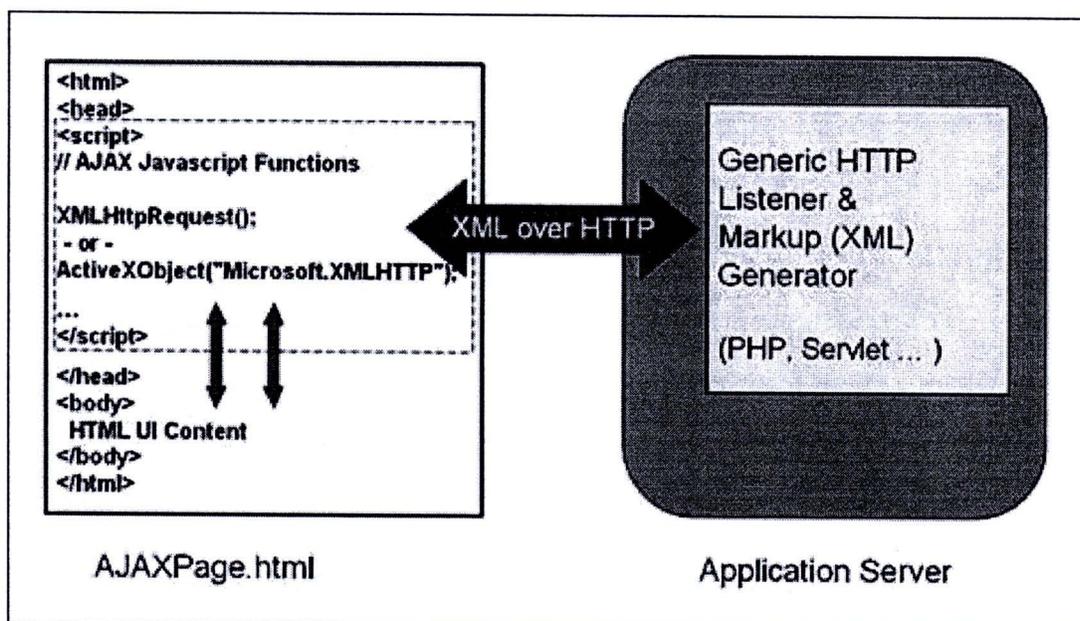
4. XML ภาษา XML ถูกออกแบบมาเพื่อที่จะใช้เป็นสื่อกลางในการรับข้อมูลมาจาก Server

การทำงานของ AJAX กับ PHP

จาก มาย์พีเอชพี (2549) อธิบายไว้ว่า ก่อนที่จะใช้งานและทดลองเขียน AJAX ได้ นั้น ต้องมี 2 เทคโนโลยี ดังนี้

1. Browser ที่สนับสนุน XMLHTTP หรือ XMLHTTP Request
2. Objects HTTP Server ที่ตอบสนอง XML ซึ่งทั้ง 2 ตัวนี้ โดยทั่วไปใช้ได้อยู่แล้ว เช่น Internet Explorer หรือ Firefox ก็สนับสนุน XMLHTTP หรือ XMLHTTP Request อยู่แล้ว และ Server โดยทั่วไปก็สามารถสนับสนุน XML ด้วยการจะเขียนหรือใช้งาน AJAX จึงไม่ต้องคิดตั้งอะไรเพิ่มเติม

หลักการของ AJAX คือ HTML JavaScript และ XML และควรมีความรู้ทางด้านภาษาที่ทำงานบน Server เช่น PHP, JSP และ ASP ใดๆอย่างหนึ่งด้วย เพื่อเพิ่มประสิทธิภาพและความสามารถของ AJAX ยิ่งขึ้น ดังแสดงในภาพ 24



ภาพ 24 การทำงานของ AJAX กับ PHP

ที่มา. จาก *มาเรียน AJAX กับ PHP*, โดย มาย์พีเอชพี, 2549, ค้นเมื่อ 28 กันยายน 2553, จาก <http://www.mindphp.com/modules.php?name=News&file=article&sid=14>

การเขียนโปรแกรมด้วย NetBeans IDE และ Compiler Java

โปรแกรม NetBeans

วิกิพีเดีย สารานุกรมเสรี (2553ก) อธิบายว่า NetBeans เป็นเครื่องมือสำหรับนักโปรแกรมเมอร์ที่จะใช้พัฒนา Application ด้วยภาษา Java และเป็นซอฟต์แวร์แบบ Open Source ด้วย

นอกจากนี้ ยังสามารถพัฒนาได้อีกหลากหลาย โดยติดตั้งโปรแกรมเสริม (add-on) ได้จากเว็บไซต์หรือผ่านตัว Update Center ของ NetBeans เช่น ภาษา C หรือ C++, Ruby, UML, SOA, Web Application, Java EE, Mobility (Java ME), Java FX, Java Script และ PHP NetBeans ใน Version 6.0 เป็นต้นไป มีการรวมโปรแกรมเสริมต่าง ๆ ที่สำคัญเข้าไปในตัวติดตั้งของ NetBeans โดยสามารถเลือกติดตั้งได้ภายหลัง

โปรแกรม NetBeans IDE

สิ่งแวดล้อมสำหรับการพัฒนาแบบเบ็ดเสร็จ (Integrated Development Environment--IDE) คือ โปรแกรมประยุกต์ซอฟต์แวร์ที่คอยอำนวยความสะดวกให้แก่ นักเขียน โปรแกรมคอมพิวเตอร์ในการพัฒนาซอฟต์แวร์ ซึ่งประกอบด้วย บรรณาธิกรรหัสต้นทาง (source code editor) โปรแกรมแปลโปรแกรม (compiler) หรือโปรแกรมแปลคำสั่ง (interpreter) หรือทั้งสอง เครื่องมือวางระบบอัตโนมัติ (build automation tools) และ โปรแกรมตรวจแก้จุดบกพร่อง (debugger) (วิกิพีเดีย สารานุกรมเสรี, 2554)

โปรแกรม Compiler Java

ภาษา Java เป็นภาษาที่ใช้ตัวแปลภาษาชนิด Compiler ดังนั้น จึงต้องทำ Compile โปรแกรมก่อน

การ Compile โปรแกรมภาษา Java นั้น จะถูกนำไปสร้างโปรแกรมตามหลักการ และไวยากรณ์ของการเขียน จะได้ไฟล์นามสกุล .java จากนั้นจึงนำไป Compile โดยใช้ Java Compiler ให้เป็น Byte Codes ซึ่งจะมีนามสกุลเป็น .class แล้วนำโปรแกรมหรือไฟล์ .class นั้นมาทำงานด้วยเครื่องจักรเสมือน (Java Virtual Machine--JVM) ที่จำลองขึ้นโดย Java Interpreter (ไอทีเมโลดี้คอตคอม, ม.ป.ป.-ก)

โปรแกรม JVM

Java Virtual Machine--JVM คือ Engine ในระบบคอมพิวเตอร์ที่เข้าใจ โปรแกรมของภาษา Java ทำหน้าที่จัดการและตีความแต่ละคำสั่งใน โปรแกรม Java ที่ถูกเรียกว่า Byte Codes แปลงให้เป็นคำสั่ง เพื่อประมวลผลของแต่ละระบบปฏิบัติการที่ใช้งาน (ไอทีเมโลดี้คอตคอม, ม.ป.ป.-ก)

การใช้ Library MySQL-Connector-Java เชื่อมต่อระหว่าง Database กับ NetBeans

การใช้ Library MySQL-Connector-Java เชื่อมโยงกับ Database

การเชื่อมโยง MySQL กับ Database มีขั้นตอนดังนี้ (มหาวิทยาลัยพะเยา, คณะเทคโนโลยีสารสนเทศและการสื่อสาร, ม.ป.ป.)

1. การติดตั้ง Database Driver ในโปรแกรม NetBeans สามารถใช้ NetBeans เพื่อเชื่อมต่อ Database โดยใช้ JDBC Driver ซึ่งเป็น Driver ที่พัฒนาโดยใช้ภาษา Java และติดต่อโดยตรงกับ Database โดยใช้ Protocol MySQL สำหรับการติดตั้ง Driver
 2. การสร้าง Database Connection โดยใช้ Driver ที่พัฒนาโดยใช้ภาษา Java และติดต่อโดยตรงกับ Database โดยใช้ Protocol MySQL และเป็น Driver ซึ่งโปรแกรม NetBeans ได้ติดตั้งมาไว้แล้ว สามารถที่จะเชื่อมต่อกับ Database โดยใช้ Driver ดังกล่าวได้
 3. การสร้างฐานข้อมูล กำหนดให้สร้าง Table ให้อยู่ภายใต้ Schema โดย Table นี้ กำหนดให้มี Column ต่าง ๆ และจะใช้โปรแกรม NetBeans ในการที่จะสร้างฐานข้อมูล
 4. การใช้คำสั่ง SQL ใน NetBeans ภายหลังจากที่มีการสร้าง Table สามารถที่จะใช้โปรแกรม NetBeans สร้างคำสั่ง SQL เพื่อที่จะติดต่อกับฐานข้อมูล ในที่นี้จะแสดงการเพิ่มข้อมูลลงใน Table
- นอกจากนี้ สามารถที่จะกำหนดการใช้คำสั่ง SQL ค้นหาข้อมูลใน Table ได้ โดยเลือกคำสั่ง Design Query โปรแกรม NetBeans จะแสดงหน้าต่างเพื่อให้สามารถกำหนดคำสั่ง Select ได้