

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบฐานข้อมูลจราจรเมืองหลวงบนเว็ลด์ไวด์เว็บ

The capitalized-transportation database system on World Wide Web



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

เลขหมู่.....
เลขทะเบียน..... 34128
วัน, เดือน, ปี - 5 ต.ค. 2542

ระบบฐานข้อมูลจราจรเมืองหลวงบนเว็ลด์ไวด์เว็บ

The capitalized-transportation database system on World Wide Web



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทบริหารศึกษาศึกษา 2541

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

เรื่อง ระบบฐานข้อมูลจราจรเมืองหลวงบน เวิลด์ไวด์เว็บ

ผู้จัดทำ

1. นาย นเรศ สุระถิตย์ เลขประจำตัว 39013238
2. นาย สุเทพ จันทร์เพ็ญ เลขประจำตัว 39013255



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบฐานข้อมูลจราจรเมืองหลวงบน เวิลด์ไวด์เว็บ

นาย นเรศ สุระถิตย์ 39013238

นาย สุเทพ จันทร์เพ็ง 39013255

อ.นวพร วรรณวิมลศรี อาจารย์ที่ปรึกษา

ปีการศึกษา 2541

บทคัดย่อ

โครงการนี้เป็น การสร้างระบบผู้ให้บริการเวิลด์ไวด์เว็บด้านการให้บริการฐานข้อมูลจราจรใน กรุงเทพมหานครและเขตปริมณฑลบนเวิลด์ไวด์เว็บ ซึ่งจะให้บริการในเรื่องของข้อมูลการเดินทางใน กรุงเทพมหานครและเขตปริมณฑล ด้วยยานพาหนะประเภทต่างๆ อันได้แก่ รถขสมก. , เรือ , รถไฟ , รถตู้ เป็นต้น พร้อมทั้งมีบริการอื่นๆ ที่มีความจำเป็นในการเดินทางอีกด้วย โดยเราอาศัย Microsoft SQL Server Version 6.5 ซึ่งเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์(Relational Database Management) เป็นระบบจัดการฐานข้อมูล(DBMS: Database Management system) ส่วนการติดต่อกับฐานข้อมูลจราจรเมืองหลวงนั้นเราอาศัยบริการของ IIS(Internet Information Server) ซึ่งได้เตรียมบริการสำหรับการติดต่อกับ ODBC(Open Database Connectivity) ซึ่งจะทำให้เราติดต่อกับระบบจัดการฐานข้อมูลหลายๆ ประเภทได้รวมทั้ง MS SQL โดยใช้หลักการของ IDC(Internet Database Connector) ส่วนในเรื่องของการติดต่อกับผู้ใช้บริการนั้นเราอาศัยภาษาเอชทีเอ็มแอล เขียนโฮมเพจขึ้นมา เพื่อให้ผู้ใช้บริการสามารถใช้บริการที่มีอยู่ได้ และได้มีการใช้ภาษาจาวา เข้ามาช่วยทำให้โฮมเพจที่เราสร้างขึ้นมีความน่าสนใจ มากขึ้นสำหรับการทำงานนั้นเราจะเริ่มจากการศึกษารายละเอียดของส่วนประกอบต่างๆ ของการทำงานเพื่อนำมาใช้ในการทำโครงการนี้ซึ่งเราก็คือว่าจะเป็นแนวทางในการพัฒนาในอนาคตต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The capitalized-transpiration database system on World Wide Web

Mr.Nares Surathit

Mr.Suthep Janpheng

Miss.Navaphorn Vanvimonsi Advisor

ABSTRACT

This project is the database system, which arrange information about transportation in and around a city on the World Wide Web. The details, that guiding the paths to you in order to reach your destination by public bus, train and van, are the service of this system. Tool to manager DBMS (Database management system) is Microsoft SQL Server version 6.5, which is the relational database management. Method to connect to database, is done by IIS (Internet information Server), that is providing service for connecting to ODBC (Open database connectivity) with the base on IDC (Internet database Connector). HTML, which a language for creating home page is the way to contract to users. Besides of HTML, JAVA is the once which supporting home page to interest work. The operation of this project has start which critical and analytical information of process in system as to use in. We hope that it is developed in the further

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการนี้นับตั้งแต่เริ่มต้นจนกระทั่งสำเร็จเป็นโครงการที่สมบูรณ์ได้นั้นส่วนหนึ่งเป็นเพราะอาจารย์ที่ปรึกษา อ.นวพร วรรณวิมลศรี ที่ท่านได้สละเวลาอันมีค่ามาให้คำปรึกษาตลอดจนแนะนำและช่วยแก้ปัญหาต่างๆ ที่เกิดขึ้นในระหว่างการทำงาน

และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ที่ให้การเลี้ยงดูอบรมสั่งสอนจนทำให้ผู้เขียนมีวันนี้ ข้าพเจ้าขอระลึกในพระคุณอันสุดประมาณและขอกราบขอบพระคุณมา ณ ที่นี้

นาย นเรศ สุระถิตย์

นาย สุเทพ จันทรเพ็ญ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญภาพ	VII
บทที่ 1 บทนำ	1
1.1.ความสำคัญและที่มา	1
1.2.วัตถุประสงค์ของงานวิจัย	2
1.3.ขอบเขตของงานวิจัย	2
1.4.วิธีการดำเนินงาน	2
บทที่ 2 สถาปัตยกรรมฐานข้อมูลเชิงสัมพันธ์	4
2.1.สถาปัตยกรรมของระบบฐานข้อมูล	4
2.2.ฐานข้อมูลเชิงสัมพันธ์	9
2.3.แนวคิดเกี่ยวกับการออกแบบฐานข้อมูล	19
2.4.การสร้างโมเดลจำลองความสัมพันธ์ระหว่างข้อมูลแบบ E-R โมเดล	38
บทที่ 3 การใช้งาน Microsoft SQL Server 6.5 เบื้องต้น	43
3.1.สถาปัตยกรรมระบบฐานข้อมูล	43
3.2.ภาษาสอบถามเชิงโครงสร้าง (Structure Query Language)	69
บทที่ 4 เทคนิคการติดต่อกับฐานข้อมูล	84
4.1.หลักการทำงานของ Internet Database Connector	84
4.2.ODBC (Open Database Connectivity)	85
4.3.การติดตั้ง ODBC	87
4.4.การสร้าง System Data Sources	87
บทที่ 5 ภาษาเอชทีเอ็มแอล (HTML) เบื้องต้น	91
5.1.บทนำ	91
5.2. เอชทีเอ็มแอล (HTML) เบื้องต้น	95
5.3.การนำเว็บเพจที่เราสร้างใส่ในอินเทอร์เน็ต	101
บทที่ 6 การติดตั้งและบริหารเว็บเซิร์ฟเวอร์	105
6.1. โครงสร้างและองค์ประกอบของระบบปฏิบัติการวินโดวส์เอ็นทีเซิร์ฟเวอร์	105
6.2. การติดตั้งและจัดการ โพรโตคอลเครือข่ายแบบ TCP/IP	107
6.3. การติดตั้งและบริหารอินเทอร์เน็ตเซิร์ฟเวอร์	120

6.4.การจัดการ FTP Service	129
บทที่ 7 การดำเนินการปฏิบัติและพัฒนาโครงการ	131
7.1.คำนำ	131
7.2.รายละเอียดของโครงการ	132
7.3.การดำเนินการพัฒนาโครงการ	132
บทที่ 8 การทดสอบและผลการทดสอบ	149
8.1.บทนำ	149
8.2.การทดสอบและผลการทดสอบ	149
บทที่ 9 บทสรุปและวิจารณ์	156
บรรณานุกรม	157



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้าที่
ตารางที่ 2.1 แสดงศัพท์เทคนิคที่เกี่ยวข้อง	9
ตารางที่ 2.2 รีเลชั่นที่แสดงประวัติพนักงาน	10
ตารางที่ 2.3 แสดงข้อมูลในแต่ละแถวจะมีซ้ำกัน	12
ตารางที่ 2.4 แสดงการเกิด Repeating Group	13
ตารางที่ 2.5 รีเลชั่น PROJWORK	14
ตารางที่ 2.6 รีเลชั่นข้อมูลพนักงานและแผนกของบริษัท	15
ตารางที่ 2.7 รีเลชั่น Supplier	20
ตารางที่ 2.8 รีเลชั่น Supplier1	22
ตารางที่ 2.9 รีเลชั่น SPJC	23
ตารางที่ 2.10 ก. รีเลชั่นที่ไม่อยู่ในรูปแบบบรรทัดฐานที่ 1	25
ตารางที่ 2.10 ข. รีเลชั่นที่อยู่ในรูปแบบบรรทัดฐานที่ 1	25
ตารางที่ 2.11 รีเลชั่น ORDER1	27
ตารางที่ 2.12 รีเลชั่น SUPPLIER1	28
ตารางที่ 2.13 แสดงการแตกรีเลชั่น	30
ตารางที่ 2.14 รีเลชั่น SUPPLIER3	31
ตารางที่ 2.15 รีเลชั่น SPP	35
ตารางที่ 3.1 แสดงตาราง System table ของ Master Database	45
ตารางที่ 3.2 แสดงตาราง System table ของ Model Database	46
ตารางที่ 3.3 แสดงตาราง System table ของ MSDB Database	47
ตารางที่ 3.4 แสดง SQL Server Object	59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้าที่

รูปที่ 2.1 แสดงเค้าร่างฐานข้อมูลบริษัท	4
รูปที่ 2.2 โครงสร้างข้อมูลระดับภายนอกหรือวิว	5
รูปที่ 2.3 ภาพรวมความสัมพันธ์ข้อมูลทั้งสามระดับ	7
รูปที่ 2.5 แสดงการสร้างวิวของผู้ใช้	18
รูปที่ 2.7 รีเลชัน SPJC ที่ถูกทำให้อยู่ในรูปแบบบรรทัดฐานขั้นที่ 4	21
รูปที่ 2.8 แสดงการแตกรีเลชัน SPP	31
รูปที่ 2.9 ความสัมพันธ์แบบ Total Participation	31
รูปที่ 2.10 ความสัมพันธ์แบบ Partial Participation	33
รูปที่ 3.1 แสดงโครงสร้างฐานข้อมูล SQL Server	36
รูปที่ 3.2 แสดงโครงสร้างของ Master Database	39
รูปที่ 3.3 แสดงลักษณะของ System Database	40
รูปที่ 3.4 แสดงโครงสร้างของ Model Database	43
รูปที่ 3.5 แสดงโครงสร้างของ Msdb	44
รูปที่ 3.6 แสดงโครงสร้างของ Tempdb Database	44
รูปที่ 3.7 แสดงการใช้คำสั่ง Sp_ (Stored Procedures)	45
รูปที่ 3.8 แสดงการใช้คำสั่ง Sp_datatype_info	47
รูปที่ 3.9 แสดงการปรับเปลี่ยค่า Configuration	48
รูปที่ 3.10 แสดง Manafer Database Devices	49
รูปที่ 3.11 แสดงการสร้าง Device ให้ Transaction log	50
รูปที่ 3.12 แสดงการใช้สแตทเมนต์ DISK INIT ในการสร้าง Device	51
รูปที่ 3.13 แสดงการใช้ Sp_helpdevice	53
รูปที่ 3.14 แสดงรายละเอียดของ Database Device	53
รูปที่ 3.14 (ต่อ) แสดงรายละเอียดของ Database Device	54
รูปที่ 3.14 (ต่อ) แสดงรายละเอียดของ Database Device	55
รูปที่ 3.15 แสดงการจัดพื้นที่ของฐานข้อมูล	55
รูปที่ 3.16 แสดงการทำงานของ Transaction log	56
รูปที่ 3.17 แสดง Device และ Database แต่ละแบบ	57
รูปที่ 3.18 แสดงการการสร้างฐานข้อมูลชื่อ DATA_T	58
รูปที่ 3.19 แสดงการ Edit Database และ Database option	60
รูปที่ 3.19 (ต่อ) แสดงการ Edit Database และ Database option	60
รูปที่ 3.20 แสดงสถานะภาพของ Virtud Device number (VDEVNO)	61
รูปที่ 3.21 แสดงการใช้คำสั่ง Sp_helpdb เพื่อดูรายละเอียดของฐานข้อมูล	62

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นสำหรับงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านอื่น
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.22 แสดงข้อความผิดพลาดในการสร้าง Device	63
รูปที่ 3.23 แสดงการขยายฐานข้อมูล	64
รูปที่ 3.23(ต่อ) แสดงการขยายฐานข้อมูล	65
รูปที่ 3.24 แสดงการลดขนาดฐานข้อมูลด้วย Shrink	66
รูปที่ 3.25 แสดงการขยาย Transaction log	67
รูปที่ 4.1 การติดต่อ Database ผ่านทาง Browser	67
รูปที่ 4.2 การเชื่อมต่อไปยังฐานข้อมูลของ Internet Information Server	68
รูปที่ 4.3 แสดงการเชื่อมต่อและเรียกใช้ข้อมูลหลากหลายด้วย ODBC	84
รูปที่ 4.4 แสดงขั้นตอนการเรียกใช้ข้อมูลผ่าน ODBC Driver Manager	85
รูปที่ 4.5 แสดงการติดตั้ง ODBC	85
รูปที่ 4.6 แสดงขั้นตอนการทำงานของ การ Access Database จนกระทั่งได้ผลลัพธ์	86
รูปที่ 5.1 แสดงลักษณะของโฮมเพจ	87
รูปที่ 5.2 แสดงลักษณะของเว็บเพจ	88
รูปที่ 5.3 แสดงลักษณะของเว็บไซต์	91
รูปที่ 6.1 โครงสร้างของระบบปฏิบัติการวินโดวส์เอ็นที	92
รูปที่ 6.2 หลักการทำงานของ VMM	93
รูปที่ 6.3 โครงสร้างการทำงาน I/O Manager	105
รูปที่ 6.4 มาตรฐานของ ISO (International Organization for Standardization)	106
รูปที่ 6.5 การเปรียบเทียบโปรโตคอลแบบต่าง ๆ กับมาตรฐานของ ISO	107
รูปที่ 6.6 แสดงถึงการติดต่อสื่อสารระหว่างคอมพิวเตอร์ที่ใช้โปรโตคอล TCP/IP	108
รูปที่ 6.7 การแบ่งเชิงชั้นของการระบุดำเนินงานแบบ Hierarchy	109
รูปที่ 6.8 วิธีการติดตั้งโปรโตคอล TCP/IP	109
รูปที่ 6.9 แสดงรายชื่อของโปรโตคอล	113
รูปที่ 6.10 จอภาพ Microsoft TCP/IP Properties	114
รูปที่ 6.11 แสดงการใส่ IP Address, SUBNET Mask และ Default Gateway	115
รูปที่ 6.12 แสดงการทำงานของโปรแกรม FTP Client	117
รูปที่ 6.13 แสดงการทำงานของคำสั่ง Ping	118
รูปที่ 6.14 แสดงเส้นทางที่จะเข้าถึง web site ของศูนย์คอมพิวเตอร์และอิเล็กทรอนิกส์แห่งชาติ	119
รูปที่ 6.15 ลักษณะของโปรแกรม ftp client	120
รูปที่ 6.16 การทำงานของคำสั่ง Telnet	120
รูปที่ 6.17 จอภาพ Select Network Services	121
รูปที่ 6.18 แสดงจอภาพ Microsoft Internet Information Server Setup	122
รูปที่ 6.19 แสดง ODBC Drivers and Administration	123
รูปที่ 6.20 จอภาพเพื่อติดตั้ง ITS เรียบร้อย	124

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.21 แสดงโปรแกรม Internet Service Manager	124
รูปที่ 6.22 จอภาพ WWW Service Properties	125
รูปที่ 6.23 ลักษณะของ Directories Tab	127
รูปที่ 6.24 แสดง Directory Browsing Allow	128
รูปที่ 6.25 แสดง FTP Services Properties	129
รูปที่ 6.26 แสดงการจัดการ Gopher Service	130
รูปที่ 7.1 Flow Chart แสดงการทำงานพัฒนาโครงการ	131
รูปที่ 7.2 E-R โมเดล	137
รูปที่ 7.3 E-R โมเดล	138
รูปที่ 7.4 E-R โมเดลที่ผ่านการทำให้รูปในรูปบรรทัดฐาน	139
รูปที่ 7.5 เคำร่างข้อมูลของแต่ละริเลชั่น	140
รูปที่ 7.6 วิวของริเลชั่น Index_Goal	141
รูปที่ 7.7 วิวสำหรับการสืบค้น 1 ต่อถึงปลายทาง	141
รูปที่ 7.8 วิวเพื่อการสืบค้นเส้นทางแบบ 2 ต่อถึงปลายทาง	142
รูปที่ 7.9 วิวเพื่อการสืบค้นเส้นทางแบบ 3 ต่อถึงปลายทาง	142
รูปที่ 7.10 หน้าต่าง Microsoft SQL Enterprise Manager	143
รูปที่ 7.11 หน้าต่างการสร้าง Database Device	143
รูปที่ 7.12 หน้าต่างการสร้าง Log Device	144
รูปที่ 7.13 แสดงการสร้างฐานข้อมูลชื่อ Traffic_DBNOW	144
รูปที่ 7.14 ฐานข้อมูลที่จะทำการ Upsize จาก Access ไปที่ SQL Server	144
รูปที่ 7.15 การติดตั้ง ODBC	145
รูปที่ 7.16 แสดงตารางและวิวที่เราสร้างขึ้นบน DBMS	145
รูปที่ 7.17 โสมเพจแสดงบริการทั้งหมดของโครงการ	146
รูปที่ 7.18 แสดงเว็บเพจการสืบค้นเส้นทางจราจร	146
รูปที่ 7.19 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลรถ ขสมก	146
รูปที่ 7.20 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลรถไฟ	147
รูปที่ 7.21 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลรถตู้	147
รูปที่ 7.22 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลเรือ	148
รูปที่ 7.23 แสดงเว็บเพจการสืบค้นรายละเอียดของบริการสอบถามและรายงานสภาพจราจร	148
รูปที่ 7.24 แสดงเว็บเพจการสืบค้นรายละเอียดของบริการเสนอแนะการให้บริการ	148
รูปที่ 7.25 แสดงเว็บเพจการสืบค้นสถานที่ท่องเที่ยว	148
รูปที่ 7.25 แสดงเว็บเพจบริการสอบถามตารางรถไฟทั่วประเทศ	148
รูปที่ 8.1 การทดสอบสืบค้น โดยกำหนดจุดค้นทางปลายทาง	149
รูปที่ 8.2 แสดงผลการทดสอบความเป็นไปได้	149

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ
 ไร่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 8.3 ตัวอย่างการสืบค้นแบบ 1 ต่อถึงปลายทาง	150
รูปที่ 8.4 การทดสอบสืบค้นรายละเอียดรถ ขสมก	150
รูปที่ 8.5 ผลการทดสอบจากการสืบค้นจากหมายเลขรถ ขสมก.	150
รูปที่ 8.6 ผลการทดสอบการสืบค้นจากชื่อสถานที่	151
รูปที่ 8.7 การสืบค้นรายละเอียดรถไฟ	151
รูปที่ 8.8 ผลการทดสอบการสืบค้นจากชื่อสถานที่รถไฟ	151
รูปที่ 8.9 ผลการทดสอบการสืบค้นจากหมายเลขรถไฟ	152
รูปที่ 8.10 การทดสอบสืบค้นรายละเอียดตู้	152
รูปที่ 8.11 ผลการทดสอบการสืบค้นจากชื่อรถตู้	152
รูปที่ 8.12 การทดสอบสืบค้นรายละเอียดเรือ	153
รูปที่ 8.13 ผลการทดสอบการสืบค้นจากชื่อเรือ	153
รูปที่ 8.14 ผลการทดสอบการสืบค้นจากชื่อท่าเรือ	153
รูปที่ 8.15 การทดสอบการรายงานสภาพจราจร	154
รูปที่ 8.16 ผลการทดสอบแสดงการเก็บข้อมูลลงฐานข้อมูลแล้ว	154
รูปที่ 8.17 การทดสอบการสอบถามสภาพจราจร	154
รูปที่ 8.18 ผลการทดสอบการสอบถาม	154
รูปที่ 8.19 แสดงความช่วยเหลือกรณีไม่ทราบจุดที่จะสอบถามสภาพจราจร	155
รูปที่ 8.20 การสืบค้นเส้นทางจากแผนที่	155

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1. ความสำคัญและที่มา

จากการที่อินเทอร์เน็ต(Internet) ได้รับความนิยมอย่างมากในปัจจุบัน และมีอัตราการเจริญเติบโตสูงมาก เราทราบว่าอินเทอร์เน็ตในวันนี้ คือเครือข่ายคอมพิวเตอร์สาธารณะที่มีขนาดใหญ่ และสำคัญที่สุดในโลก เราถือว่าการเชื่อมต่อคอมพิวเตอร์และเครือข่ายด้วยโพรโทคอล TCP/IP โดยคอมพิวเตอร์ที่เข้ามาเป็นส่วนหนึ่งของอินเทอร์เน็ตจะต้องมี IP Address ไว้เป็นสิ่งที่อ้างอิงเมื่อเราจะติดต่อกับคอมพิวเตอร์เครื่องนั้น

1.1.1. สำหรับบริการต่างๆ ที่มีในเครือข่ายอินเทอร์เน็ต มีดังนี้คือ

- 1.E-MAIL (Electronic Mail)
- 2.FTP (File Transfer Protocol)
- 3.IRC (Internet Relay Chat)
- 4.NewsGroup
- 5.Gopher
6. WWW (World Wide Web)

สำหรับบริการที่ได้รับความนิยมสูงสุด และเป็นสิ่งที่ทำให้อินเทอร์เน็ตน่าใช้งานคือ WWW หรือที่เราเรียกว่าสั้นๆ ว่าเว็บ(Web) ซึ่งเป็นการขยายความสามารถในเชิงธุรกิจมากที่สุด เราจึงมุ่งเน้นให้ความสนใจกับเว็ลด์ไวด์เว็บมากกว่าบริการอื่นๆ ที่มีอยู่

1.1.2. องค์ประกอบของเว็ลด์ไวด์เว็บ

- เว็บเบราว์เซอร์(Web Browser) คือแอปพลิเคชันที่นำผู้ใช้งานอินเทอร์เน็ต ไปสู่แหล่งข้อมูลต่างๆ โดยเบราว์เซอร์จะทำหน้าที่แสดงเอกสารตามที่ผู้ใช้ต้องการ
- เว็บเซิร์ฟเวอร์(Web Server) คือแอปพลิเคชันที่คอยรองรับการร้องขอจากเบราว์เซอร์ ซึ่งการร้องขอจากเบราว์เซอร์ อาจจะต้องการดูเอกสาร, เรียกค้นข้อมูลจากฐานข้อมูลหรือทำการคำนวณ ซึ่งเว็บเซิร์ฟเวอร์จะดำเนินการตามที่ต้องการแล้วส่งผลลัพธ์ไปแสดงที่เบราว์เซอร์
- โพรโทคอล TCP/IP เป็นโพรโทคอลสำหรับการสื่อสารระหว่างเครื่องคอมพิวเตอร์ที่เชื่อมต่อกันในเครือข่ายอินเทอร์เน็ต
- ไฮเปอร์ลิงก์ (Hyperlink) เป็นการเชื่อมจากแหล่งข้อมูลหนึ่งไปอีกแหล่งข้อมูลหนึ่ง
- เอชทีเอ็มแอล (HTML: Hypertext Markup Language)
- ISP (Internet Service Provider)
- ผู้ให้บริการฐานข้อมูล (Database Server)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นเราจะทำการศึกษาการทำงานและการสร้างเว็บแอปพลิเคชัน โดยจะมุ่งเน้นไปที่เว็บแอปพลิเคชันที่มีการติดต่อกับฐานข้อมูล โดยจะยกตัวอย่างการสร้างเว็บแอปพลิเคชันที่มีการติดต่อกับฐานข้อมูลที่ให้บริการการสืบค้นเส้นทางจราจรเมืองหลวง

1.2 วัตถุประสงค์ของงานวิจัย

- 1.2.1. เพื่อศึกษาการออกแบบและสร้างฐานข้อมูลเชิงสัมพันธ์(Relational Database)
- 1.2.2. เพื่อศึกษาการทำงานของเว็บแอปพลิเคชันในเรื่องของการติดต่อกับฐานข้อมูลบนเว็ลด์ไวด์เว็บ
- 1.2.3. เพื่อเป็นการเสนอแนวคิดในการสร้างเว็บแอปพลิเคชันฐานข้อมูลจราจรเมืองหลวง ที่สามารถนำไปใช้งานได้และเป็นแนวทางในการพัฒนาต่อไปสำหรับผู้ที่มีความสนใจ

1.3 ขอบเขตของงานวิจัย

เราสามารถมองภาพกว้างๆของการให้บริการเป็น 2 ส่วน คือ

- บริการสืบค้นข้อมูลจากฐานข้อมูลจราจรเมืองหลวง
- บริการเสริมที่ทำให้เว็บไซต์ดูมีประโยชน์และน่าสนใจมากขึ้น

1.3.1. บริการสืบค้นข้อมูลจากฐานข้อมูลจราจรเมืองหลวง

รายละเอียดการให้บริการค้นหาเส้นทางจราจรในกรุงเทพมหานครและเขตปริมณฑลผ่านระบบเครือข่ายอินเทอร์เน็ต โดยเราจะแยกการสืบค้นเส้นทางที่เป็นไปได้ออกเป็น 2 กรณี คือ

1. สืบค้นด้วยพาหนะส่วนบุคคล
2. สืบค้นด้วยพาหนะอื่นๆ คือ รถประจำทาง, รถไฟ, รถตู้และเรือ

1.3.2. บริการเสริมที่ทำให้เว็บไซต์ดูมีประโยชน์และน่าสนใจมากขึ้น

ดังนั้นเราจึงแยกการพัฒนาและออกแบบการติดต่อกับผู้ใช้บริการ ออกเป็น 2 กรณี และเพื่อเป็นการเพิ่มความสามารถของเว็บไซต์และทำให้ เว็บไซต์ดูมีประโยชน์ที่หลากหลายมากขึ้นเราจะเพิ่มบริการอื่นๆ ที่เราเห็นว่า มีประโยชน์ต่อผู้ที่เข้ามาใช้บริการดังนี้ คือ

1. บริการแนะนำสถานที่ท่องเที่ยวที่น่าสนใจในกรุงเทพมหานครและเขตปริมณฑล
2. บริการสอบถามและรายงานสภาพการจราจรจุดต่างๆ ในกรุงเทพมหานครและเขตปริมณฑล
3. บริการรับข้อเสนอแนะจากผู้เข้ามาใช้บริการ
4. บริการข่าวสารที่น่าสนใจและมีประโยชน์ในการเดินทาง

1.4 วิธีการดำเนินงาน

เราสามารถแบ่งการทำงานออกเป็น 3 ขั้นตอน คือ

- 1.4.1. ศึกษาทฤษฎีที่จำเป็นในการออกแบบฐานข้อมูลเชิงสัมพันธ์และเทคนิคการติดต่อกับฐานข้อมูล โดยเราออกสารนี้เป็นเอกสารที่วางไว้สำหรับนำไปใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า สามารถแจกจ่ายออกเป็นขั้นตอนได้ดังนี้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 เรื่องความรู้เบื้องต้นของฐานข้อมูลเชิงสัมพันธ์

บทที่ 3 เรื่องการใช้งาน Microsoft SQL Server version 6.5 เบื้องต้นเพื่อนำมาใช้ในการออกแบบและสร้างฐานข้อมูลเชิงสัมพันธ์

บทที่ 4 เรื่องเทคนิคต่างๆ ที่จำเป็นในการติดต่อกับฐานข้อมูล

บทที่ 5 เรื่องภาษาเอชทีเอ็มแอลเบื้องต้นเพื่อนำมาใช้ในการเขียนโฮมเพจ

บทที่ 6 เรื่องการติดตั้งและบริหารเว็บเซิร์ฟเวอร์

1.4.2.การออกแบบและพัฒนาโครงการ โดยอาศัยทฤษฎีที่เราได้ศึกษามาในขั้นตอนแรกโดยเราสามารถจำแนกออกเป็น 3 ส่วนคือ

- การออกแบบและสร้างฐานข้อมูลจราจรเมืองหลวง
- การออกแบบและสร้างโฮมเพจ
- การออกแบบส่วนที่ใช้ในการติดต่อกับฐานข้อมูล

1.4.3.เป็นขั้นตอนการทดสอบการทำงานของส่วนต่างๆ ที่เราสร้างขึ้นในทุกกรณีที่เป็นไปได้ และสุดท้ายเป็นการสรุปผลการทำงานและนำเสนอแนวทางในการพัฒนาโครงการของผู้ที่มีความสนใจ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

สถาปัตยกรรมฐานข้อมูลเชิงสัมพันธ์

2.1. สถาปัตยกรรมของระบบฐานข้อมูล

คุณลักษณะเด่นอย่างหนึ่งของระบบฐานข้อมูล คือ การอำนวยความสะดวกในการให้ข้อมูลแก่ผู้ใช้ซึ่งมีวัตถุประสงค์ที่แตกต่างกันออกไป เช่น ผู้ใช้บางคนต้องการเรียกใช้ข้อมูลทั้งเพิ่มข้อมูล หรือผู้ใช้บางคนต้องการเรียกใช้ข้อมูลบางส่วนเท่านั้น ในขณะที่ผู้ใช้บางคนต้องการทราบถึงลักษณะการจัดเก็บข้อมูล เป็นต้น สถาปัตยกรรมของฐานข้อมูลเป็นการกล่าวถึงระดับต่างๆ ของข้อมูลในระบบฐานข้อมูล ซึ่งข้อมูลในระบบฐานข้อมูลมักจะถูกกำหนดขึ้นหรือถูกมองในรูปแบบที่แตกต่างกันตามวัตถุประสงค์ของการใช้ข้อมูลของผู้ใช้

ในหัวข้อนี้จะกล่าวถึงเค้าร่างของฐานข้อมูลสถาปัตยกรรมของระบบฐานข้อมูล ความเป็นอิสระของข้อมูล และภาษาที่ใช้ในระบบจัดการฐานข้อมูล

2.1.1. เค้าร่างของฐานข้อมูล (Database Schema)

ในการออกแบบฐานข้อมูลใด จะต้องระบุถึงเอนทิตีชื่ออะไรบ้าง และในแต่ละเอนทิตีประกอบด้วยแอททริบิวต์อะไร รวมถึงความสัมพันธ์ของข้อมูลในเอนทิตี ที่มีอยู่ว่ามีความสัมพันธ์ในลักษณะใด รายละเอียดของโครงสร้างของฐานข้อมูลนี้เรียกว่า เค้าร่างของฐานข้อมูล โดยทั่วไปเค้าร่างในฐานข้อมูลมักจะไม่เปลี่ยนแปลงบ่อยนัก แต่อาจมีการเปลี่ยนแปลงบ้างในบางครั้ง ถ้าการเปลี่ยนแปลงทำให้ฐานข้อมูลเหมาะสมมากยิ่งขึ้น ข้อมูลที่ถูกบันทึกลงในฐานข้อมูลในขณะใดขณะหนึ่ง เรียกว่า อินสแตนซ์ (Instance หรือ Occurrence) กล่าวอีกนัยหนึ่ง คือ เมื่อเริ่มกำหนดเค้าร่างของฐานข้อมูลจะยังไม่มีข้อมูลอยู่ คือ เป็นอินสแตนซ์ว่าง (Empty Instance) จนกว่าจะบันทึกข้อมูลลงในฐานข้อมูล ซึ่งอาจมีการปรับเปลี่ยนข้อมูลให้ถูกต้องตามที่ควรจะเป็นในขณะใดขณะหนึ่ง ตัวอย่างของเค้าร่างฐานข้อมูลบริษัทประกอบด้วย เอนทิตีพนักงาน (EMPLOYEE) เอนทิตีแผนก (DEPT) เอนทิตีโครงการ (PROJECT) และเอนทิตีงานที่มอบหมาย (PROJWORK) ดังแสดงตามรูปที่ 2.1

EMPLOYEE (พนักงาน)

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPTNO	MGRNO
--------	---------	----------	--------	----------	--------	-------

DEPT (แผนก)

DEPTNO	DEPTNAME	LOCATION
--------	----------	----------

PROJECT (โครงการ)

PROJNO	PROJDESC	STARTDATE	ENDDATE	BUDGET
--------	----------	-----------	---------	--------

PROJWORK (งานที่มอบหมาย)

PROJNO	EMPNUM	HOURS
--------	--------	-------

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 2.1 แสดงเค้าร่างฐานข้อมูลบริษัท อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2. สถาปัตยกรรมของฐานข้อมูล

จากการที่ระบบฐานข้อมูลประกอบด้วยโครงสร้างของข้อมูล ที่ผู้ใช้หลายกลุ่มสามารถเรียกใช้ข้อมูลได้ การแบ่งระดับของข้อมูลออกเป็นระดับต่างๆ มีวัตถุประสงค์เพื่อให้การใช้ข้อมูลของผู้ใช้เป็นไปอย่างเหมาะสม นั่นคือ ถ้าเป็นผู้ใช้ทั่วไป (End Users) ผู้ใช้อาจไม่จำเป็นต้องทราบถึงรายละเอียดในการเก็บข้อมูลหรือข้อมูลที่ไม่ต้องการเรียกใช้ ระดับของข้อมูล แบ่งออกเป็น 3 ระดับ ดังนี้ คือ

2.1.2.1. ระดับภายนอกหรือวิว (External Level หรือ View)

เป็นระดับของข้อมูลที่ประกอบด้วยภาพที่ผู้ใช้แต่ละคนมองข้อมูล เคี้ยวร่างของข้อมูลระดับนี้เกิดจากภาพ และความต้องการข้อมูลของผู้ใช้ ตัวอย่างเช่น ฐานข้อมูลบริษัทจากรูปที่ 1 ผู้ใช้ทั่วไปคนแรก ต้องการใช้ข้อมูลรหัสพนักงาน และเงินเดือน ในขณะที่โปรแกรมเมอร์ ซึ่งเป็นผู้ใช้ข้อมูลอีกผู้หนึ่ง ต้องการใช้ข้อมูลรหัสพนักงาน เงินเดือน และรหัสแผนก ดังรูปที่ 2.2

ระดับภายนอก	
ผู้ใช้คนที่ 1 (PASCAL)	
VAR	
	EMPNUM : STRING (4);
	SALARY : REAL;
END	
ผู้ใช้คนที่ 2 (COBOL)	
01	EMP
02	EMPNO PIC X(4)
02	SALARY PIC 99999
03	DEPNO PIC X(2)
ระดับแนวคิด	
EMPLOYEE	
	EMPNUM CHARACTER (4)
	EMPNAME CHARACTER (15)
	HIREDATE DATE (10)
	SALARY NUMERIC (6)
	POSITION CHARACTER (10)
	DEPNO CHARACTER (2)
	MGRNO CHARACTER (4)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 2.2 โครงสร้างของข้อมูลระดับภายนอกหรือวิว นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.2. ระดับแนวคิด (Conceptual Level)

ประกอบด้วยเค้าร่างที่อธิบายถึงฐานข้อมูลรวมว่ามีเอนทิตี โครงสร้างของข้อมูล ความสัมพันธ์ของข้อมูล กฎเกณฑ์และข้อจำกัดต่างๆ อย่างไรก็ตาม ข้อมูลในระดับนี้เป็นข้อมูลที่ผ่านมาการวิเคราะห์และออกแบบโดย ผู้บริหารฐานข้อมูล(Database Administrator) หรือนักวิเคราะห์และออกแบบฐานข้อมูลเป็นระดับของข้อมูลที่ถูกออกแบบ เพื่อให้ผู้ใช้ข้อมูลต่างๆ ในระดับภายนอกสามารถเรียกใช้ข้อมูลได้ ซึ่งผู้ใช้ทั่วไปในระดับภายนอกอาจต้องการใช้ข้อมูลที่แตกต่างกัน

2.1.2.3. ระดับภายใน (Internal หรือ Physical Level)

ประกอบด้วยเค้าร่างที่เกี่ยวกับการจัดเก็บข้อมูลจริงๆ ว่ามีโครงสร้างการจัดเก็บรูปแบบใด รวมถึงวิธีการเข้าถึงข้อมูลต่างๆ ในฐานข้อมูล เพื่อดึงข้อมูลที่ต้องการ เช่น การอินเด็กซ์ (Indexing) เป็นต้น

ความสัมพันธ์ระหว่างข้อมูลระดับต่างๆ จะถูกจัดการโดยระบบการจัดการฐานข้อมูล ซึ่งเป็นการแปลงความหมายของข้อมูลจากระดับหนึ่งไปยังอีกระดับหนึ่งนี้เรียกว่า การแปลงส่ง (Mapping) โดยทั่วไปการแปลงส่งระหว่างระดับต่างๆ ของข้อมูล ระบบจัดการฐานข้อมูลจะจัดการให้โดยอัตโนมัติ การแปลงส่งนี้แบ่งออกเป็น 2 ลักษณะ คือ

ก. การแปลงส่งระหว่างระดับแนวคิดและระดับภายใน (Conceptual/Internal Mapping)

เป็นการแปลงส่งข้อมูลระหว่างระดับแนวคิด และระดับภายใน เช่น ข้อมูลในระดับแนวคิดถูกแสดงไว้อย่างไรในระดับภายใน

ข. การแปลงส่งระหว่างระดับภายนอกและระดับแนวคิด (External/Conceptual Mapping)

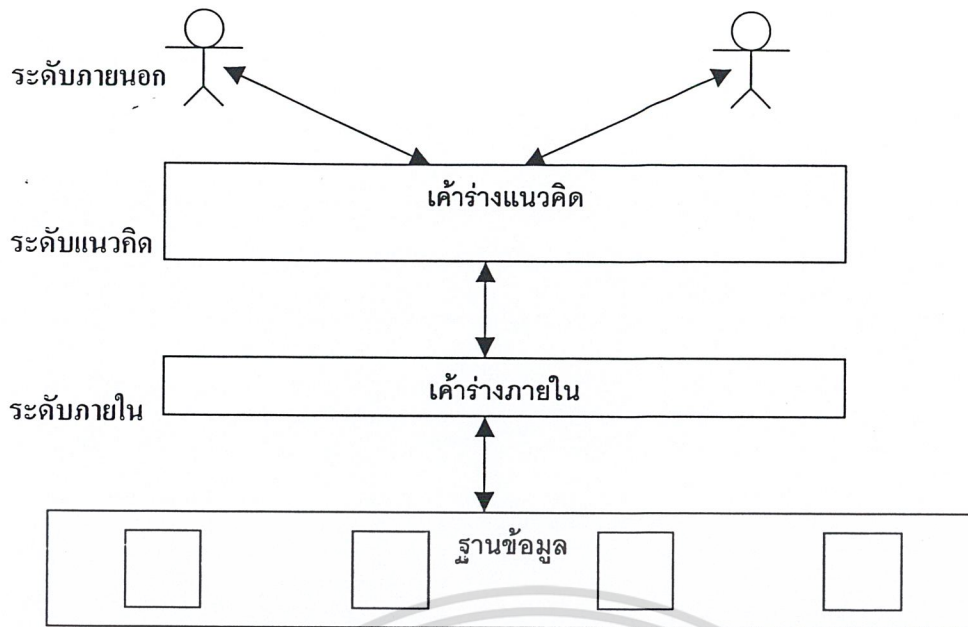
เป็นการแปลงส่งระหว่างระดับแนวคิดกับระดับภายนอก เช่น การใช้ภาษาสำหรับนิยามข้อมูล (Data Definition Language) ในการแปลงข้อมูลระดับภายนอกเป็นตารางหรือแอททริบิวต์ต่างๆ เป็นต้น

ตามที่ได้กล่าวมาข้างต้นว่า ระดับของข้อมูลแบ่งออกเป็น 3 ระดับ ดังนั้น ข้อมูลในระดับดังกล่าวจะประกอบด้วยเค้าร่าง 3 ประเภท ดังนี้ คือ

1. เค้าร่างภายใน (Internal Schema) เป็นเค้าร่างที่แสดงรายละเอียดถึงลักษณะการจัดเก็บโครงสร้างจริงๆ

2. เค้าร่างแนวคิด (Conceptual Schema) เป็นเค้าร่างที่แสดงรายละเอียดของฐานข้อมูลทั้งหมด ไม่ว่าจะเป็นชื่อเอนทิตี โครงสร้างข้อมูล ความสัมพันธ์ และข้อจำกัดต่างๆ

3. เค้าร่างภายนอก (External Schema หรือ Subschema หรือ View) เป็นเค้าร่างในระดับภายนอกที่แสดงถึงรายละเอียดของข้อมูลที่ใช้ต่างๆ ต้องการ



รูปที่ 2.3 ภาพรวมความสัมพันธ์ของข้อมูลทั้งสามระดับ

2.1.3. ความเป็นอิสระของข้อมูล (Data Independence)

การแบ่งระดับของข้อมูลรวมถึงการเชื่อมโยงของข้อมูลในสองลักษณะที่กล่าวมาข้างต้น เป็นจุดเด่นของฐานข้อมูลในด้านความเป็นอิสระของข้อมูล ซึ่งความเป็นอิสระของข้อมูล แบ่งออกเป็น 2 ประเภท คือ

2.1.3.1. ความเป็นอิสระของข้อมูลในเชิงตรรกะ (Logical Data Independence)

เป็นความอิสระของข้อมูล ในระดับแนวคิด (Conceptual Level) กับระบบภายนอก (External Level) นั่นคือ หากมีการเปลี่ยนแปลงของข้อมูลในระดับแนวคิด (Conceptual Level) จะไม่มีผลต่อเค้าร่างในระดับภายนอก (External Level) หรือโปรแกรมประยุกต์ใช้งานไม่จำเป็นที่จะเป็นการปรับโครงสร้าง เช่น การเพิ่มแอททริบิวต์ หรือการเพิ่มเอนทิตีใหม่เข้าไปในฐานข้อมูล

2.1.3.2. ความเป็นอิสระในเชิงกายภาพ (Physical Data Independence)

เป็นความอิสระของข้อมูลในระดับภายใน (Internal Level) กับระดับแนวคิด (Conceptual Level) หรือระดับภายนอก (External Level) เช่น การเพิ่มประสิทธิภาพการเรียกดูข้อมูลให้เร็วขึ้น โดยการปรับปรุงเค้าร่างภายใน (Internal Schema) โดยไม่กระทบถึงเค้าร่างแนวคิดหรือเค้าร่างภายนอก (Conceptual หรือ External Schema)

2.1.4. ภาษาที่ใช้ในระบบจัดการฐานข้อมูล

ในการจัดการฐานข้อมูลสามารถเขียนโปรแกรมโดยใช้ภาษาต่างๆ เช่น COBOL, PASCAL, C ฯลฯ นอกจากนี้ยังมีภาษาเฉพาะ ที่สามารถนำมาใช้ในการจัดการฐานข้อมูล ซึ่งผู้ใช้ทั่วไป โปรแกรมเมอร์ หรือผู้จัดการฐานข้อมูลสามารถนำมาใช้ เป็นคำสั่งง่ายๆ ไม่สลับซับซ้อน ภาษาเฉพาะดังกล่าวสามารถเรียกใช้ผ่านระบบจัดการฐานข้อมูล แบ่งออกเป็น 3 ประเภท คือ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.1. ภาษาสำหรับนิยามข้อมูล (Data Definition Language: DDL)

ประกอบด้วยคำสั่งที่ใช้ในการกำหนดโครงสร้างข้อมูลว่ามีแอททริบิวต์อะไรเก็บข้อมูลประเภทใด การเพิ่มแอททริบิวต์ การกำหนดดัชนีที่ใช้ในการจัดเก็บข้อมูล การกำหนดวิวของผู้ใช้ เป็นต้น จากกรณีที่พนักงาน สามารถใช้ภาษาที่ใช้ในการจัดการฐานข้อมูล กำหนดโครงสร้างของตารางพนักงาน ว่าประกอบด้วยแอททริบิวต์อะไรบ้าง ภาษาที่จะยกตัวอย่างในที่นี้คือ SQL (Structure Query Language) ซึ่งจะกล่าวอย่างละเอียดต่อไป คำสั่งของ SQL ที่ใช้ในการกำหนดโครงสร้างเป็นดังนี้ คือ

```
CREATE TABLE EMPLOYEE
    (EMPNUM CHAR (4),
    EMPNAME CHAR (15),
    HIREDATE DATE,
    SALARY NUMERIC (6),
    POSITION CHAR (10),
    DEPNO CHAR (2),
    MGRNO CHAR (4))
```

2.1.4.2. ภาษาสำหรับจัดการข้อมูล (Data Manipulation Language: DML)

ประกอบด้วยคำสั่งที่ใช้ในการเรียกใช้ข้อมูล การเปลี่ยนแปลงข้อมูล การเพิ่มหรือลบข้อมูล เป็นต้น จากตารางพนักงาน สามารถใช้ SQL ในการเรียกข้อมูลต่างๆ มาดู เช่น ต้องการดูข้อมูล ชื่อพนักงาน และตำแหน่งของพนักงานที่มีเงินเดือนมากกว่า 30,000 คำสั่งที่ใช้คือ

```
SELECT EMPNAME, POSTION
FROM EMPLOYEE
WHERE SALARY > 30000;
```

2.1.4.3. ภาษาที่ใช้ในการควบคุมข้อมูล (Data Control Language: DCL)

ประกอบด้วยคำสั่งที่ใช้ในการควบคุมความถูกต้องของข้อมูล หรือป้องกันการเกิดเหตุการณ์ที่ผู้ใช้หลายคนเรียกใช้ข้อมูลพร้อมกัน ในขณะที่ข้อมูลนั้นๆ กำลังถูกปรับปรุงแก้ไขอยู่ ซึ่งเป็นเวลาเดียวกับที่ผู้ใช้อีกคนหนึ่งก็เรียกใช้ข้อมูลนี้ โดยข้อมูลที่ผู้ใช้คนที่สองได้ไปเป็นค่าที่ไม่ถูกต้อง ทั้งนี้เพราะผู้ใช้งานแรกยังปรับปรุงแก้ไขข้อมูลไม่เสร็จ นอกจากนี้ ยังประกอบด้วยคำสั่งที่เกี่ยวข้องกับการควบคุมความปลอดภัยของข้อมูล คือ การให้สิทธิ์ผู้ใช้ในการเรียกดูหรือแก้ไขปรับปรุงข้อมูล ตัวอย่างเช่น ให้สิทธิ์ผู้ใช้ A ในการเรียกดูข้อมูลจากราย EMPLOYEE คำสั่ง SQL ที่ใช้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
GRANT SELECT ON EMPLOYEE TO A;
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2. ฐานข้อมูลเชิงสัมพันธ์

ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เป็นรูปแบบของฐานข้อมูลที่นิยมใช้ในปัจจุบัน รูปแบบนี้ได้ถูกคิดค้นและพัฒนาโดย อี เอฟ คอดด์ (E.F.Codd) ฐานข้อมูลเชิงสัมพันธ์เป็นรูปแบบที่เข้าใจง่าย สำหรับผู้ใช้ ไม่ซับซ้อนรวมถึงเป็นรูปแบบที่มีเครื่องมือที่ช่วยในการเรียกใช้ข้อมูลโดยใช้คำสั่งง่ายๆ ดังเช่น SQL ส่วนใหญ่ระบบจัดการฐานข้อมูลที่ใช้กันอยู่ในปัจจุบัน เป็นระบบที่ใช้กับฐานข้อมูลเชิงสัมพันธ์ เป็นส่วนใหญ่ ตัวอย่างเช่น DB2, ORACLE, INGRES, dBase IV, FoxPro หรือ ACCESS เป็นต้น

ในหัวข้อนี้จะกล่าวถึง ศัพท์เทคนิคที่เกี่ยวข้องกับฐานข้อมูลเชิงสัมพันธ์ ความหมายและข้อดีของ ฐานข้อมูลเชิงสัมพันธ์ คุณลักษณะในการจัดเก็บข้อมูลของรีเลชัน ประเภทของคีย์ กฎที่เกี่ยวข้องกับคีย์ ในฐานข้อมูลเชิงสัมพันธ์ และประเภทของรีเลชัน

2.2.1. ศัพท์เทคนิคที่เกี่ยวข้อง

ในการประมวลผลแบบเพิ่มข้อมูล คำศัพท์ที่นิยมใช้คือเพิ่มข้อมูล (File) เรคคอร์ด (Record) ฟิลด์ (Field) เมื่อเปรียบเทียบกับศัพท์ที่ใช้ในฐานข้อมูลเชิงสัมพันธ์ คือ ตาราง (Table) แถว (Row) และคอลัมน์ (Column) ตามลำดับ อันที่จริงแล้ว ศัพท์ต่างๆ มักจะถูกเรียกสลับกันไป เช่น ใช้คำว่าไฟล์ แทนตาราง เป็นต้น อย่างไรก็ตามเพื่อให้เข้าใจถึงศัพท์ต่างๆ ทั้งที่เป็นศัพท์เทคนิคที่ใช้เรียกเฉพาะ และ ศัพท์ที่ใช้เรียกทั่วไปในฐานข้อมูลเชิงสัมพันธ์มากขึ้น คำศัพท์ต่างๆ ที่ควรทราบมีดังตารางที่ 2.1 คือ

ศัพท์เทคนิค	ศัพท์ทั่วไป
รีเลชัน (Relation)	ตาราง (Table)
ทิวเปิล (Tuple)	แถว (Row) หรือ เรคคอร์ด (Record)
แอททริบิวต์ (Attribute)	คอลัมน์ (Column) หรือฟิลด์ (Field)
คาร์ดินาลิตี (Cardinality)	จำนวนแถว (Number of Rows)
คีย์หลัก (Primary key)	ค่าเอกลักษณ์ (Unique Identifier)
โดเมน (Domain)	ขอบเขตค่าของข้อมูล

ตารางที่ 2.1 แสดงศัพท์เทคนิคที่เกี่ยวข้อง

จากตารางที่ 2.1 เป็นตารางที่แสดงให้เห็นถึงคำศัพท์ต่างๆ ที่ใช้เรียกในฐานข้อมูลเชิงสัมพันธ์ ซึ่งมักจะใช้เรียกแทนกันในบางครั้ง สรุปได้ดังนี้คือ

- ความหมายของรีเลชัน (Relation) มีคำเรียกทั่วไปว่าตาราง (Table) เป็นคำนามที่แทนข้อมูลของเรื่องใดเรื่องหนึ่ง เช่น รีเลชันของข้อมูลพนักงาน เป็นต้น ในตำราเล่มนี้จะใช้คำว่า รีเลชัน เป็นการแทนความหมายของตารางในฐานข้อมูลเชิงสัมพันธ์

- ความหมายของทิวเปิล (Tuple) คือ ค่าของข้อมูลในแต่ละแถว (Row) หรือที่เรียกว่าเรคคอร์ด (Record)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเนื้อหาและข้อมูลอ้างอิงเชิงวิชาการที่สงวนไว้สำหรับการนำไปใช้

- ความหมายของคาร์ดินาลิตี (Cardinality) คือ จำนวนแถวของข้อมูลในแต่ละรีเลชัน

- ความหมายของแอททริบิวต์ (Attribute) คือ รายละเอียดของข้อมูลในแต่ละคอลัมน์ ซึ่งเป็นสิ่งที่แสดงให้รู้ว่ามีอะไรบ้าง ประกอบด้วยรายละเอียดข้อมูลอะไรบ้าง หรือที่รู้จักกันในชื่อของคอลัมน์ (Column) หรือฟิลด์ (Field)

- ความหมายของคีย์หลัก (Primary Key) คือ แอททริบิวต์ที่มีค่าของข้อมูลเป็นเอกลักษณ์หรือเฉพาะเจาะจงและจะเป็นค่าที่ไม่ซ้ำกันในแต่ละทูเปิล

- ความหมายของโดเมน หมายถึง ขอบเขตของค่าของข้อมูลที่ควรจะเป็นในแต่ละแอททริบิวต์ เพื่อให้เกิดความเข้าใจถึงศัพท์เทคนิคดังกล่าว จะใช้ตัวอย่างในตารางที่ 2.2 อธิบายเพิ่มเติมดังนี้

EMPLOYEE (พนักงาน)

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPNO	MGRNO
1001	SIRIWAN	06/13/93	9000	CLERK	10	1002
3001	ARLEE	08/18/93	17000	SALESMAN	30	3004
4001	WICHAI	12/26/93	33000	MANAGER	40	2002
1002	JINTANA	10/31/93	30000	CONTROLLER	10	1003
3002	MITREE	12/05/93	13000	SALESMAN	30	3004
3003	BENJAWAN	06/11/94	29000	MANAGER	30	2002
2001	CHAI	05/14/93	14000	CLERK	20	2003
1003	SURASIT	03/15/94	30000	MANAGER	10	2002
2002	KANJANA	07/10/94	50000	DIRECTOR	20	
3001	TANACHO	06/14/94	25000	SUPERVISOR	30	3003
1004	AMPORN	06/04/94	12000	CLERK	10	1002
3005	TAWATCHAI	07/13/94	10000	SALESMAN	30	3004
4002	THIDARAT	12/01/94	9000	CLERK	40	4001
2003	TERNJAI	11/01/94	24000	MANAGER	20	2002

ตารางที่ 2.2 รีเลชันที่แสดงข้อมูลของประวัติพนักงาน

จากตารางที่ 2.2 เป็นรีเลชันที่แสดงข้อมูลของประวัติพนักงาน (EMPLOYEE) ซึ่งประกอบด้วย 7 แอททริบิวต์ หรือ รายละเอียด 7 อย่าง ดังนี้ คือ

EMPNUM หมายถึง รหัสของพนักงาน

EMPNAME หมายถึง ชื่อของพนักงาน

HIREDATE หมายถึง วันที่เริ่มทำงาน

SALARY หมายถึง เงินเดือน

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEPNO	หมายถึง รหัสแผนกที่พนักงานทำงานอยู่
MGRNO	หมายถึง รหัสผู้บังคับบัญชา

รีเลชันนี้ประกอบด้วย 7 แอททริบิวต์ที่เป็นเค้าร่างของรีเลชัน (Relation Schema) ซึ่งรีเลชันนี้จะมีคาร์ดินาลลิตี้ เท่ากับ 14 ทูเพิล

สำหรับคีย์หลัก (Primary Key) เป็นค่าของแอททริบิวต์ที่มีค่าของข้อมูลที่เป็นเอกลักษณ์ ไม่มีความซ้ำซ้อนของข้อมูลในแต่ละทูเพิล นั่นคือ EMPNUM ซึ่งเป็นรหัสของพนักงาน

ความหมายของโดเมน เป็นขอบเขตหรือค่าที่ควรจะเป็นของข้อมูลในแอททริบิวต์หนึ่ง เช่น วันที่เริ่มทำงาน (HIREDATE) ควรจะเป็นค่าในวันที่ในปฏิทิน หรือข้อมูล MGRNO เป็นรหัสผู้บังคับบัญชา ซึ่งควรจะเป็นรหัสพนักงานบางคนที่อยู่ในตำแหน่งบริหาร และอาจจะมีกรณีของค่าว่างในแอททริบิวต์นี้ สำหรับพนักงานที่เป็นผู้บริหารสูงสุดจึงไม่มีผู้บังคับบัญชา เพราะฉะนั้น ค่าของ MGRNO อาจเป็นค่าว่างได้ (NULL) สำหรับพนักงานที่เป็นผู้บริหารสูงสุดจะเป็นค่าว่าง จากตารางที่ 2.2 พนักงานรหัส 2002 คือ DIRECTOR จึงไม่มีค่าของรหัสผู้บังคับบัญชา

2.2.2. ฐานข้อมูลเชิงสัมพันธ์

เป็นการจัดเก็บรวบรวมข้อมูลเป็นแถวและคอลัมน์ในลักษณะตารางสองมิติ ที่ประกอบด้วยแอททริบิวต์ที่แสดงคุณสมบัติของรีเลชันหนึ่งๆ โดยที่รีเลชันต่างๆ ได้ผ่านกระบวนการทำรีเลชันให้เป็นบรรทัดฐาน (Normalized) ในระหว่างการออกแบบเพื่อลดความซ้ำซ้อน และเพื่อให้การจัดการฐานข้อมูลเป็นไปอย่างมีประสิทธิภาพ

ฐานข้อมูลเชิงสัมพันธ์เป็นฐานข้อมูลซึ่งให้ภาพของข้อมูลในระดับภายนอก (External Level) และระดับแนวคิด (Conceptual Level) แก่ผู้ใช้ข้อมูลได้เป็นอย่างดี รีเลชันต่างๆ ในฐานข้อมูลจะประกอบด้วยแอททริบิวต์ต่างๆ ที่ถูกออกแบบเพื่อลดความซ้ำซ้อนของการเก็บข้อมูล และสามารถเรียกใช้ข้อมูลอย่างมีประสิทธิภาพ โดยมีระบบจัดการฐานข้อมูลเป็นผู้จัดการฐานข้อมูลตามที่ฐานข้อมูลได้ถูกออกแบบไว้

2.2.2.1. ข้อดีของฐานข้อมูลเชิงสัมพันธ์

ฐานข้อมูลเชิงสัมพันธ์เป็นรูปแบบของฐานข้อมูลที่เข้าใจง่ายสำหรับผู้ใช้งานทั่วไป โดยเฉพาะผู้ใช้ซึ่งไม่ใช่นักวิเคราะห์และออกแบบระบบ โปรแกรมเมอร์หรือผู้จัดการฐานข้อมูล เป็นต้น ข้อดีของฐานข้อมูลเชิงสัมพันธ์ที่ทำให้ผู้ใช้ทั่วไปรู้สึกฐานข้อมูลชนิดนี้เข้าใจง่าย มีดังนี้ คือ

1. ฐานข้อมูลเชิงสัมพันธ์เป็นกลุ่มของรีเลชันหรือตารางที่ข้อมูลถูกจัดเก็บเป็นแถวและคอลัมน์ ซึ่งทำให้ผู้ใช้เห็นภาพของข้อมูลได้ง่าย

2. ผู้ใช้ไม่ต้องรู้ว่าข้อมูลถูกจัดเก็บจริงอย่างไร รวมถึงวิธีการเรียกใช้ข้อมูล (Access Approach)

3. ภาษาที่ใช้ในการเรียกดูข้อมูล เป็นลักษณะคล้ายภาษาอังกฤษ และไม่จำเป็นต้องเขียนเป็นลำดับขั้นตอน (Procedural) เช่น SQL

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การเรียกใช้หรือเชื่อมโยงข้อมูลทำได้ง่ายโดยใช้โอเปอเรเตอร์ทางคณิตศาสตร์ เช่น SELECT PROJECTION เป็นต้น

2.2.2.2. คุณลักษณะในการจัดเก็บข้อมูลของรีเลชัน

ในแต่ละรีเลชันประกอบด้วยข้อมูลของแอททริบิวต์ต่างๆ ที่ถูกจัดเก็บอยู่ในรูปตารางสองมิติ คือ แถว (Row) และคอลัมน์ (Column) คุณลักษณะในการจัดเก็บข้อมูลของรีเลชันเป็นดังนี้ คือ

2.2.2.2.1. ข้อมูลในแต่ละแถวจะไม่ซ้ำกัน

ข้อมูลที่จัดเก็บในแต่ละแถวจะไม่มีข้อมูลที่ซ้ำกัน ทั้งนี้เพราะการจัดการฐานข้อมูลที่ดีไม่ควรจะมีข้อมูลที่ซ้ำซ้อนปรากฏอยู่ในรีเลชัน โดยระบบจัดการฐานข้อมูลจะมีกลไกที่ใช้ในการควบคุมไม่ให้ความซ้ำซ้อนเกิดขึ้น ตัวอย่างเช่น จากตารางที่ 2.3 มีข้อมูลของพนักงานรหัส 1001 ซ้ำกัน ระบบจะมีความเตือนว่า รหัสพนักงานซ้ำซ้อนกัน เป็นต้น

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPNO	MGHNO
1001	SIRIWAN	06/13/93	9000	CLERK	10	1002
3001	ARLEE	08/15/93	17000	SALESMAN	30	3004
4001	WICHAI	12/26/93	33000	NABAGER	40	2002
1001	SIRIWAN	06/13/93	9000	CLERK	10	1002

ตารางที่ 2.3 ข้อมูลในแต่ละแถวจะมีซ้ำกัน

2.2.2.2.2. การเรียงลำดับของข้อมูลในแต่ละแถวไม่เป็นสาระสำคัญ

การจัดเก็บของข้อมูลในรีเลชัน จะถูกจัดเรียงตามลำดับลงบนสื่อที่เก็บข้อมูล แต่การเรียกใช้ข้อมูลในรีเลชันสามารถเรียกใช้ตามที่ใช้ต้องการ

2.2.2.2.3. การเรียงลำดับของแอททริบิวต์จะเรียงลำดับก่อนหลังอย่างไรก็ได้

การเรียงลำดับของแอททริบิวต์แต่ละแอททริบิวต์ของรีเลชันจะเรียงลำดับอย่างไรก็ได้ ไม่มีการระบุว่าแอททริบิวต์ซ้ายสุดคือ แอททริบิวต์แรก หรือแอททริบิวต์ขวาสุดคือ แอททริบิวต์สุดท้าย ทั้งนี้เพราะการอ้างอิงถึงแอททริบิวต์ใดจะใช้ชื่อของแอททริบิวต์นั้นๆ ในการอ้างอิง ไม่ใช่ลำดับที่แอททริบิวต์นั้นปรากฏอยู่

2.2.2.2.4. ค่าของข้อมูลในแต่ละแอททริบิวต์ของทูเพิลหนึ่งๆ จะบรรจุข้อมูลได้เพียงค่าเดียว

ข้อมูลในแต่ละแอททริบิวต์ของทูเพิลหนึ่งๆ จะต้องบรรจุข้อมูลเพียงค่าเดียวไม่ใช่กลุ่มของข้อมูลที่แสดงค่าที่มากกว่าหนึ่งแถว (Repeating Group)
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPNO	MGRNO
1001	SIRIWAN	06/13/93	9000	CLERK	10 20	1002 2001
3001	ARLEE	08/15/93	17000	SALESMAN	30	3004
4001	WICHAI	12/26/93	33000	MANAGER	40 30	2002 3004

ตารางที่ 2.4 แสดงการเกิด Repeating Group

จากตารางที่ 2.4 ข้อมูลของพนักงานรหัส 1001 และ 4001 มีค่าของ DEPNO ที่มากกว่าหนึ่งค่า (Repeating Group) ซึ่งไม่ใช่คุณลักษณะในการจัดเก็บข้อมูลของรีเลชัน

2.2.2.2.5. ค่าของข้อมูลในแต่ละแอททริบิวต์จะบรรจุค่าของข้อมูลประเภทเดียวกัน

ข้อมูลในแต่ละแอททริบิวต์จะเป็นค่าของข้อมูลประเภทเดียวกัน ซึ่งถูกกำหนดค่าให้เป็นโดเมน ตัวอย่างเช่น แอททริบิวต์วันเกิด จะมีค่าของข้อมูลเป็นวันเดือนปีเกิดต่างๆ ในปีปฏิทิน ไม่ใช่เงินเดือนหรือข้อมูลที่อยู่หรือแอททริบิวต์สถานภาพจะมีค่าของข้อมูลที่แสดงว่าเป็นโสดหรือแต่งงาน โดยขอบเขตของค่าที่ถูกกำหนดไว้ คือ S แทนค่าโสด และ M แทนค่าว่าแต่งงานแล้ว เป็นต้น

2.2.2.3. ประเภทของคีย์

เค้าร่างของรีเลชัน (Relation Schema) ประกอบด้วยรายละเอียดของแอททริบิวต์ต่างๆ ซึ่งแอททริบิวต์ใดแอททริบิวต์หนึ่งในรีเลชันจะมีคุณสมบัติเป็นคีย์ โดยคีย์นั้นๆ สามารถใช้ในการแสดงหรือบ่งบอกถึงค่าของแอททริบิวต์อื่นๆ ในทูเปิลใดทูเปิลหนึ่งของรีเลชัน หรือใช้ในการเชื่อมโยงข้อมูลกับข้อมูลอื่นในอีกรีเลชันหนึ่ง ประเภทของคีย์ที่จะกล่าวในที่นี้ แบ่งเป็น 2 ประเภทหลัก ดังนี้ คือ

2.2.2.3.1. คีย์หลัก (Primary Key)

เป็นแอททริบิวต์ที่มีคุณสมบัติของข้อมูล ที่เป็นค่าเอกลักษณ์หรือมีค่าที่ไม่ซ้ำซ้อนกัน คุณสมบัตินี้จะทำให้สามารถระบุค่าข้อมูลนั้นเป็นข้อมูลของทูเปิลใด เช่น รหัสของพนักงาน เลขที่ 2001 สามารถระบุว่าเป็นรหัสของพนักงานชื่อ CHAI และค่ารหัสพนักงานแต่ละคนจะมีค่าไม่ซ้ำกัน เป็นต้น แอททริบิวต์ที่มีคุณสมบัติเป็นคีย์หลักอาจประกอบด้วยหลายแอททริบิวต์รวมกัน เพื่อที่จะให้ค่าที่เป็นเอกลักษณ์ได้ คีย์หลักที่ประกอบด้วยหลายแอททริบิวต์นี้เรียกว่าคีย์ผสม (Composite Key) ตัวอย่างเช่น จากตาราง 2.5 รีเลชัน PROJWORK เป็นรีเลชันที่แสดงข้อมูลของพนักงานที่ได้รับมอบหมายให้ทำงานในโครงการต่างๆ แอททริบิวต์ที่เป็นคีย์หลักของรีเลชันนี้ คือ แอททริบิวต์ PROJNO (รหัสโครงการ) และแอททริบิวต์ EMPNUM (รหัสพนักงาน) ประกอบกันจึงให้ค่าที่เป็นเอกลักษณ์ได้หรือไม่ซ้ำซ้อนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีเลชัน PROJWORK

PHOJNO	EMPNUM	HOURS
01	3001	25
01	3003	30
02	4002	40
03	2003	20
03	2001	30

ตารางที่ 2.5 ตาราง PROJWORK

นอกจากนี้ ในรีเลชันหนึ่งๆ อาจจะมีแอททริบิวต์ที่มีคุณสมบัติเป็นคีย์หลักได้มากกว่าหนึ่งแอททริบิวต์ แอททริบิวต์เหล่านี้ เรียกว่า คีย์คู่แข่ง (Candidate Key) ถ้าแอททริบิวต์หนึ่งถูกกำหนดให้เป็นคีย์หลัก อีกแอททริบิวต์หนึ่งที่มีคุณสมบัติเป็นคีย์หลัก แต่ไม่ได้ถูกเลือกให้เป็นคีย์หลัก จะเรียกว่าคีย์สำรอง (Alternate Key)

2.2.2.3.2. คีย์นอก (Foreign Key)

เป็นแอททริบิวต์ในรีเลชันหนึ่ง ที่ใช้ในการอ้างอิงถึงแอททริบิวต์เดียวกันนี้ในอีกรีเลชันหนึ่ง โดยที่แอททริบิวต์นี้มีคุณสมบัติเป็นคีย์หลักในรีเลชันที่ถูกอ้างอิงถึง การที่มีแอททริบิวต์นี้ปรากฏอยู่ในรีเลชันทั้งสองก็เพื่อประโยชน์ในการเชื่อมโยงข้อมูลซึ่งกันและกัน

ตัวอย่างเช่น จากตารางที่ 2.6 ในรีเลชัน EMPLOYEE แอททริบิวต์ที่มีคุณสมบัติเป็นคีย์หลัก คือ EMPNUM และรีเลชัน DEP มีแอททริบิวต์ DEPNO เป็นคีย์หลัก ถ้าต้องการดูข้อมูลว่าพนักงานรหัส 2002 อยู่แผนกชื่ออะไร จะใช้แอททริบิวต์ DEPNO ในรีเลชัน EMPLOYEE ไปตรวจสอบกับข้อมูลรหัสแผนก DEPNO ซึ่งเป็นคีย์หลักในรีเลชัน DEP คุณสมบัติของแอททริบิวต์ DEPNO ในรีเลชัน EMPLOYEE คือ เป็นคีย์นอก นั่นเอง

EMPLOYEE (พนักงาน)

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPNO	MGRNO
1001	SIRIWAN	06/13/93	9000	CLERK	10	1002
3001	ARLEE	08/18/93	17000	SALESMAN	30	3004
4001	WICHAI	12/26/93	33000	MANAGER	40	2002
1002	JINTANA	10/31/93	30000	CONTROLLER	10	1003
3002	MITREE	12/05/93	13000	SALESMAN	30	3004
3003	BENJAWAN	06/11/94	29000	MANAGER	30	2002
2001	CHAI	05/14/93	14000	CLERK	20	2003
1003	SURASIT	03/15/94	30000	MANAGER	10	2002

เอกสารนี้เป็นเอกสารสำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่ให้ผู้อื่นได้รับทราบ หากมีข้อผิดพลาดประการใด ขออภัยเป็นอย่างสูงและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EMPLOYEE (พนักงาน) (ต่อ)

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPNO	MGRNO
2002	KANJANA	07/10/94	50000	DIRECTOR	20	
3001	TANACHO	06/14/94	25000	SUPERVISOR	30	3003
1004	AMPORN	06/04/94	12000	CLERK	10	1002
3005	TAWATCHAI	07/13/94	10000	SALESMAN	30	3004
4002	THIDARAT	12/01/94	9000	CLERK	40	4001
2003	TERNJAI	11/01/94	24000	MANAGER	20	2002

DEP (แผนก)

DEPNO	DEPNAME	LOCATION
10	ACCOUNTING	SILOM
20	ADMINISTRATION	SUKUMVIT
30	MARKETING	RATCHADA
40	FINANCE	SILOM
50	RESEARCH	SUKUMVIT

ตารางที่ 2.6 ตารางข้อมูลพนักงานและแผนกของบริษัท

2.2.2.4. กฎที่เกี่ยวข้องกับคีย์ในฐานะข้อมูลเชิงสัมพันธ์

เนื่องจากรีเลชันต่างๆ ในฐานะข้อมูลเชิงสัมพันธ์จะอ้างอิงความสัมพันธ์ของข้อมูลโดยใช้คีย์ กฎที่เกี่ยวข้องกับคีย์ในฐานะข้อมูลเชิงสัมพันธ์มีดังต่อไปนี้

2.2.2.4.1. กฎความบูรณาภาพของเอนทิตี (The Entity Integrity Rule)

กฎนี้ระบุไว้ว่า แอททริบิวต์ใดที่จะเป็นคีย์หลัก ข้อมูลในแอททริบิวต์นั้นจะเป็นค่าเอกลักษณ์ (Unique) และเป็นค่าว่าง (Null) ไม่ได้ ความหมายของการเป็นค่าว่างไม่ได้ (Not Null) ในที่นี้หมายความว่า ข้อมูลของแอททริบิวต์ที่เป็นคีย์หลัก จะไม่มีค่าไม่ได้และเป็นค่าเอกลักษณ์ในการที่จะระบุค่าของแอททริบิวต์อื่นๆ ในทูเปิลหนึ่งๆ ได้

2.2.2.4.2. กฎความบูรณาภาพของการอ้างอิง (The Referential Integrity Rule)

การอ้างอิงข้อมูลระหว่างรีเลชันในฐานะข้อมูลเชิงสัมพันธ์ จะใช้คีย์นอกของรีเลชันหนึ่งไปตรวจสอบกับค่าของแอททริบิวต์ที่เป็นคีย์หลักของอีกรีเลชันหนึ่ง เพื่อเรียกดูข้อมูลอื่นๆ ที่เกี่ยวข้อง กล่าวอีกนัยหนึ่งคือค่าของคีย์นอกจะต้องสามารถอ้างอิงให้ตรงกับค่าของคีย์หลักได้ จึงจะเชื่อมโยงหรืออ้างอิงข้อมูลระหว่างสองรีเลชันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับคีย์นอกจะมีค่าว่างได้หรือไม่ขึ้นอยู่กับกฎเกณฑ์การออกแบบฐานข้อมูลว่า ข้อจำกัดในการออกแบบข้อมูลตามนโยบายของหน่วยงานนั้นๆ เป็นอย่างไร เช่น ในกรณีรีเลชั่น พนักงาน ซึ่งมี DEPNO เป็นคีย์นอก อาจจะถูกระบุว่าต้องทราบค่าในกรณีที่พนักงานทุกคนจะถูกบรรจุเข้าแผนกต่างๆ หรือบางหน่วยงานอาจจะให้ทำการทดลองงานก่อนแล้วค่อยบรรจุเข้าแผนกต่างๆ ในกรณีหลังนี้ พนักงานนั้นก็จะมีค่ารหัสแผนกปรากฏอยู่ เป็นต้น

ในกรณีที่มีการลบข้อมูลหรือแก้ไขข้อมูลของแอททริบิวต์ ที่เป็นคีย์หลักในรีเลชั่นหนึ่ง ซึ่งมีคีย์นอกจากอีกรีเลชั่นหนึ่งอ้างอิงถึง จะทำการลบหรือแก้ไขข้อมูลได้หรือไม่นั้น ขึ้นอยู่กับการออกแบบฐานข้อมูลว่าได้ระบุให้แอททริบิวต์นั้นๆ มีคุณสมบัติอย่างไร ซึ่งจะกล่าวในที่นี้เพียง 4 ทางเลือก คือ

1. การลบหรือแก้ไขข้อมูลแบบมีข้อจำกัด (Restrict)

การลบหรือแก้ไขข้อมูลจะกระทำไม่ได้ เมื่อข้อมูลของคีย์หลักในรีเลชั่นหนึ่งไม่มีข้อมูลที่ถูอ้างอิงโดยคีย์ นอกจากรีเลชั่นหนึ่ง เช่น รหัสแผนก DEPNO ในรีเลชั่น DEP จะถูกแก้ไขหรือลบทิ้งต่อเมื่อไม่มีพนักงานคนใดสังกัดอยู่ในแผนกนั้น เป็นต้น

2. การลบหรือแก้ไขข้อมูลแบบต่อเรียง (Cascade)

การลบหรือแก้ไขข้อมูลจะทำแบบเป็นลูกโซ่ คือ หากมีการแก้ไขหรือลบข้อมูลของคีย์หลักในรีเลชั่นหนึ่งระบบจะทำการลบหรือแก้ไขข้อมูลของคีย์นอกในอีกรีเลชั่นหนึ่งที่อ้างอิงถึงข้อมูลของคีย์หลักที่ถูกลบให้ด้วย เช่น ในกรณีที่ยกเลิกแผนก 10 ในรีเลชั่น DEP ข้อมูลของพนักงานที่อยู่แผนก 10 ในรีเลชั่น EMPLOYEE จะถูกลบหรือแก้ไขไปด้วย

3. การลบหรือแก้ไขข้อมูลโดยเปลี่ยนเป็นค่าว่าง (Nullify)

การลบหรือแก้ไขข้อมูลจะทำได้เมื่อมีการเปลี่ยนค่าของคีย์นอกที่ถูกอ้างอิงให้เป็นค่าว่างเสียก่อน ตัวอย่างเช่น พนักงานที่อยู่แผนก 10 จะถูกเปลี่ยนค่าเป็นค่าว่างก่อน หลังจากนั้น การลบข้อมูลของแผนกที่มีรหัส 10 ในรีเลชั่น DEP จึงจะถูกลบทิ้งหรือแก้ไข เป็นต้น

4. การลบหรือแก้ไขข้อมูลแบบใช้ค่าโดยปริยาย (Default)

การลบหรือแก้ไขข้อมูลของคีย์หลักสามารถทำได้ โดยถ้าหากมีคีย์นอกที่อ้างอิงถึงคีย์หลักที่ถูกลบหรือแก้ไข ก็จะทำให้การปรับค่าของคีย์นอกนั้นเป็นค่าโดยปริยาย (Default Value) ที่ถูกกำหนดขึ้น ตัวอย่างเช่น ในกรณีที่ยกเลิกแผนก 10 ในรีเลชั่น DEP ข้อมูลของพนักงานที่อยู่แผนก 10 ในรีเลชั่น EMPLOYEE จะถูกเปลี่ยนค่าเป็น "00" ซึ่งเป็นค่าโดยปริยายที่หมายความว่าไม่ได้สังกัดแผนกใด เป็นต้น

ดังนั้น ผู้จัดการฐานข้อมูล หรือผู้ที่ออกแบบฐานข้อมูลจะเป็นผู้กำหนดคุณสมบัติของการกระทำดังกล่าวให้เหมาะสม เพื่อไม่ให้ข้อมูลที่จัดเก็บเสียหายหรือเกิดความขัดแย้งกันเองโดยกฎเกณฑ์ดังกล่าวจะถูกกำหนดไว้ในขณะที่กำหนดโครงสร้างของรีเลชันว่ารีเลชันนั้นมีแอททริบิวต์อะไรบ้าง มีแอททริบิวต์ใดเป็นคีย์หลัก และคีย์นอก คีย์นอกจะมีค่าว่างได้หรือไม่ การลบหรือแก้ไขข้อมูลของคีย์หลักที่ถูกอ้างอิง

ไม่อาจกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยคีย์นอกจะกระทำภายใต้วิธีใด ซึ่งการลบและแก้ไขข้อมูลดังกล่าวอาจจะอนุญาตให้ทำภายใต้วิธีการที่ต่างกันก็ได้ ตัวอย่างเช่น ถ้าต้องการสร้างรีเลชัน EMPLOYEE โดยมีรหัสแผนกเป็นคีย์นอก และมีค่าว่างไม่ได้ ต้องการให้ลบข้อมูลอย่างมีข้อจำกัด และแก้ไขข้อมูลแบบต่อเรียง ตัวอย่างคำสั่ง SQL ที่ใช้เป็นดังนี้ คือ

```
CREATE TABLE EMPLOYEE
(EMPNUM CHAR (4) PRIMARY KEY,
EMPNAME CHAR (15),
HIREDATE DATE,
SALARY NUMERIC (6),
POSITION CHAR (10),
DEPNO CHAR (2),
MGRNO CHAR (4),
FOREIGN KEY (DEPNO) REFERENCES DEP (DEPNO),
NULLS NOT ALLOWED,
DELETE OF DEP RESTRICTED,
UPDATE OF DEP CASCADES);
```

2.2.2.5. ประเภทของรีเลชัน

ประเภทของรีเลชัน อาจจำแนกออกเป็นหลายประเภท ประเภทของรีเลชันที่จะกล่าวถึงในที่นี้จะเป็นรีเลชันที่มักจะกล่าวถึงในระบบจัดการฐานข้อมูลทั่วไป ดังนี้คือ

2.2.2.5.1. รีเลชันหลัก (Base Relation)

เป็นรีเลชันหลักที่ถูกกำหนดขึ้น เพื่อเก็บข้อมูลเพื่อนำข้อมูลไปใช้ เมื่อมีการสร้างรีเลชันในภาษาสำหรับนิยามข้อมูล (DDL) เช่นใน SQL คำสั่ง CREATE TABLE เป็นการสร้างรีเลชันหลัก หลังจากนั้นก็จะทำการเก็บข้อมูลเพื่อการเรียกใช้ข้อมูลในภายหลัง รีเลชันหลักจะเป็นตารางที่มีการจัดเก็บข้อมูลจริงไว้

2.2.2.5.2. วิว (View)

เป็นรีเลชันที่ถูกสร้างขึ้น ตามความต้องการใช้ข้อมูลของผู้ใช้แต่ละคน ทั้งนี้เพราะผู้ใช้แต่ละคนในฐานข้อมูลอาจต้องการใช้ข้อมูลในลักษณะที่แตกต่างกัน จึงทำการกำหนดคิวของตนเองขึ้นจากรีเลชันหลักขึ้นมาต่างหาก เพื่อความสะดวกในการใช้ข้อมูลและช่วยให้การรักษาความปลอดภัยของฐานข้อมูลได้ง่ายขึ้น รีเลชันที่ถูกสมมุติขึ้นจะไม่มีการเก็บข้อมูลจริงๆ ในระบบ ซึ่งจะแตกต่างจากรีเลชันหลักที่มีการเก็บข้อมูลจริงๆ ไว้ จึงเป็นตารางสมมุติหรือตารางที่แปลค่ามา (Virtual Table หรือ Derived Table) ที่ถูกกำหนดโดยผู้ใช้แต่ละคน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.4 รีเลชัน EMPLOYEE และ DEP เป็นรีเลชันหลัก สมมุติว่าผู้ใช้คนที่ 1 ต้องการสร้างวิวขึ้นมา โดยเรียกดูข้อมูลชื่อตำแหน่งและเงินเดือนของพนักงานที่มีเงินเดือนมากกว่า 30000 และผู้ใช้คนที่ 2 ต้องการดูข้อมูลรหัสพนักงาน ชื่อของพนักงานที่อยู่แผนก ACCOUNTING

EMPLOYEE

EMPNUM	EMPNAME	HIREDATE	SALARY	POSITION	DEPNO	MGRNO
1001	SIRIWAN	06/13/93	9000	CLERK	10	1002
3001	ARLEE	08/15/93	17000	SALESMAN	30	3004
4001	WICHAJ	12/26/93	33000	MANAGER	40	2002
1002	JINTANA	10/31/93	30000	CONTROLLER	10	1003
3002	MITREE	12/05/93	13000	SALESMAN	30	3004
3003	BENJAWAN	06/11/94	29000	MANAGER	30	2002
2001	CHAI	05/14/93	14000	CLERK	20	2003
1003	SURASIT	03/15/94	30000	MANAGER	10	2002
2002	KANJANA	07/10/94	50000	DIRECTOR	20	
3004	TANACHOTE	06/14/94	25000	SUPERVISOR	30	3003
1004	AMPORN	06/04/94	12000	CLERK	10	1002
3005	TAWATCHAI	07/03/94	10000	SALESMAN	30	3004
4002	THIDARAT	12/01/94	9000	CLERK	40	4001
2003	TERNJAI	11/01/94	24000	MANAGER	20	2002

DEP

DEPNO	DEPNAME	LOCATION
10	ACCOUNTING	SILOM
20	ADMINISTRATION	SUKUMVIT
30	MARKETING	RATCHADA
40	FINANCE	SILOM
50	RESEARCH	SUKUMVIT

วิว : ผู้ใช้คนที่ 1

EMPNUM	POSITION	SALARY
4001	MANAGER	33000
2002	DIRECTOR	50000

วิว : ผู้ใช้คนที่ 2

EMPNUM	EMPNAME
1001	SIRIWAN
1002	JINTANA
1003	SURASIT
1004	AMPORN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
รูปที่ 2.4 แสดงการสร้างวิวของผู้ใช้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3. แนวคิดเกี่ยวกับการออกแบบฐานข้อมูล

เมื่อพูดถึงการออกแบบฐานข้อมูลจะหมายถึง การออกแบบฐานข้อมูลในระดับแนวคิด (Conceptual Level) และการออกแบบฐานข้อมูลในระดับภายในหรือเชิงกายภาพ (Internal หรือ Physical Level) การออกแบบฐานข้อมูลในระดับแนวคิดเป็นการออกแบบเค้าร่างของรีเลชันว่า รีเลชันนั้นๆ ประกอบด้วยแอททริบิวต์อะไร การออกแบบฐานข้อมูลในระดับนี้จะช่วยให้ผู้ใช้ทั่วไปสามารถเข้าใจถึงข้อมูลที่เก็บในแอททริบิวต์ต่างๆ รวมถึงการเรียกใช้ข้อมูลด้วย ส่วนการออกแบบฐานข้อมูลในระดับภายใน เป็นการออกแบบที่เน้นในเรื่องของการจัดเก็บข้อมูลว่าควรจะมีการจัดเก็บอย่างไร

การออกแบบเค้าร่างของข้อมูลเป็นการกำหนดรีเลชันต่างๆ รวมถึง แอททริบิวต์ในแต่ละรีเลชัน เพื่อให้ได้ข้อมูลตามที่ต้องการ แต่การกำหนดหรือออกแบบอย่างไรจึงจะได้เค้าร่างของข้อมูลที่ดี กล่าวอีกนัยหนึ่งคือ เค้าร่างของรีเลชันประกอบด้วย แอททริบิวต์ที่เหมาะสมหรือมากเกินไปหรือไม่ ข้อมูลที่เก็บในรีเลชันนั้นๆ มีความซ้ำซ้อนเกิดขึ้นหรือไม่ เป็นต้น ดังนั้นในการออกแบบเค้าร่างของรีเลชันที่ดี ควรจะเป็นการออกแบบที่สามารถลดปัญหาต่างๆ ที่จะเกิดขึ้นกับฐานข้อมูลให้มากที่สุด เช่น ปัญหาข้อมูลซ้ำซ้อน ปัญหาข้อมูลไม่ถูกต้อง ปัญหาการรักษาความปลอดภัยของข้อมูล เป็นต้น

ก่อนที่จะกล่าวถึงขั้นตอนการออกแบบฐานข้อมูล จะกล่าวถึงแนวคิดที่สำคัญที่ใช้เป็นแนวทางในการออกแบบฐานข้อมูลในระดับแนวคิด แนวคิดสำคัญดังกล่าวประกอบด้วย แนวคิดเรื่องความสัมพันธ์ระหว่างค่าของแอททริบิวต์ในแต่ละรีเลชัน (Functional Dependency) และการทำรีเลชันให้อยู่ในรูปแบบบรรทัดฐานต่างๆ (Normalization) ในส่วนของเรื่องการทำให้ความสัมพันธ์ระหว่างข้อมูล (Entity-Relationship Mode) จะกล่าวแยกในหัวข้อต่อไป

2.3.1. ความสัมพันธ์ระหว่างค่าของแอททริบิวต์ในแต่ละรีเลชัน (Dependency)

เนื่องจากค่าของแอททริบิวต์ในแต่ละรีเลชัน อาจมีความสัมพันธ์กันในลักษณะที่เมื่อทราบค่าของแอททริบิวต์หนึ่งๆ จะสามารถทราบถึงค่าของแอททริบิวต์อื่นๆ ของทูเพิลหนึ่งๆ ในรีเลชันได้ ลักษณะของความสัมพันธ์ระหว่างค่าของแอททริบิวต์ในแต่ละรีเลชันที่จะกล่าวถึงมีดังต่อไปนี้ คือ

2.3.1.1. ความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบฟังก์ชัน (Functional Dependency)

ความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบฟังก์ชัน คือ การที่แอททริบิวต์หนึ่งหรืออาจมากกว่าหนึ่งแอททริบิวต์ประกอบกันสามารถระบุค่าของแอททริบิวต์อื่นๆ ในทูเพิลหนึ่งได้ชัดเจน เมื่อพูดถึงถึงความสัมพันธ์ในการระบุค่าแอททริบิวต์จะเกี่ยวข้องกับคีย์หลัก ทั้งนี้เพราะว่าคุณสมบัติของคีย์หลักจะเป็นแอททริบิวต์ที่มีค่าเอกลักษณ์ (Unique) ที่สามารถระบุค่าของแอททริบิวต์อื่น ในทูเพิลหนึ่งๆ ได้

เพื่อให้เข้าใจถึงความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบฟังก์ชันได้ดียิ่งขึ้น จะใช้รีเลชัน SUPPLIER เป็นตัวอย่างในการอธิบาย รีเลชันนี้ประกอบด้วยรหัสผู้ผลิต (SNO) ชื่อผู้ผลิต (SNAME) จังหวัดที่ผู้ผลิตอยู่ (CITY) ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SUPPLIER

SNO	SNAME	CITY
S1	SERI	BANGKOK
S2	WANIDA	RAYONG
S3	SOMCHAI	RAYONG
S4	ORAPIN	BANGKOK
S5	TANACHOTE	PATUMTANE

ตารางที่ 2.7 รีเลชัน SUPPLIER

จากตารางที่ 2.7 จะเห็นว่าเมื่อทราบค่าแอททริบิวต์รหัสผู้ผลิตจะสามารถทราบค่าของแอททริบิวต์ตัวอื่นๆ ได้อย่างชัดเจนนั่นหมายความว่า แอททริบิวต์รหัสผู้ผลิต มีความสัมพันธ์ในการระบุค่าของแอททริบิวต์แบบฟังก์ชันกับแอททริบิวต์ตัวอื่นๆ ซึ่งสามารถเขียนเป็นสัญลักษณ์ ดังนี้

SUPPLIER.SNO -> SUPPLIER.SNAME

SUPPLIER.SNO -> 74747474FLOAT X, Y ประเภทของตัวเลขที่กำหนดใน

IER. (SNAME, CITY)

ในทางตรงกันข้าม หากทราบว่าค่าของแอททริบิวต์จังหวัดที่ผู้ผลิตอยู่ จะสามารถทราบค่าของรหัสของผู้ผลิต หรือชื่อของผู้ผลิตได้ชัดเจนหรือไม่ คำตอบคือ ไม่ชัดเจน ตัวอย่างเช่น หากทราบค่าจังหวัด BANGKOK ค่านี้ไม่สามารถระบุได้ชัดเจนว่าเป็นรหัสผู้ผลิต S1 หรือ S4 ทั้งนี้เพราะแอททริบิวต์ชื่อจังหวัดผู้ผลิตมีชื่อซ้ำกัน กล่าวอีกนัยหนึ่งแอททริบิวต์ต่างๆ ในรีเลชัน SUPPLIER ไม่สามารถระบุค่าของแอททริบิวต์ จังหวัดที่ผู้ผลิตอยู่ได้ชัดเจนนั่นเอง

ดังนั้นถ้าแอททริบิวต์หนึ่งมีความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบฟังก์ชันกับแอททริบิวต์อื่นๆ แสดงว่า แอททริบิวต์นั้นเป็นตัวระบุค่า (Determinant) ของแอททริบิวต์ตัวอื่นๆ

2.3.1.2. ความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบบางส่วน (Partial Dependency)

ความสัมพันธ์แบบนี้จะเกิดขึ้นได้เมื่อรีเลชันหนึ่งๆ มีคีย์หลักเป็นคีย์ผสม (Composite Key) นั่นคือ คีย์หลักของรีเลชันนั้นๆ ประกอบด้วย แอททริบิวต์หลายแอททริบิวต์ ความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบบางส่วนเกิดขึ้นเมื่อ แอททริบิวต์บางส่วนของคีย์หลัก สามารถระบุค่าของแอททริบิวต์อื่นๆ ที่ไม่ใช่คีย์หลักของรีเลชันได้ (Non-Key Attribute)

ตัวอย่างเช่น จากรูปที่ 2.5 รีเลชัน ORDER1 มีแอททริบิวต์ SNO และ PNO เป็นคีย์หลักและประกอบด้วย แอททริบิวต์อื่นๆ ดังต่อไปนี้

SNO หมายถึง รหัสผู้ผลิต

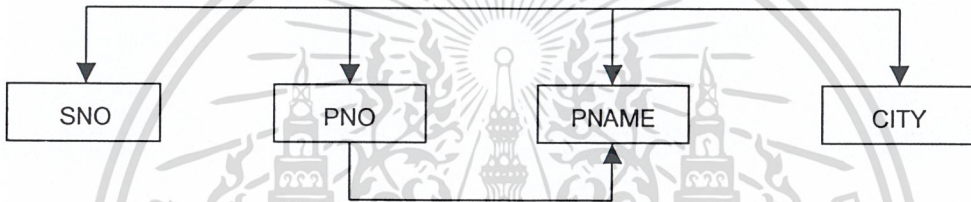
PNO หมายถึง รหัสสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเฉพาะเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PNAME หมายถึง ชื่อสินค้า
 QTY หมายถึง จำนวนสินค้าที่ถูกจัดส่ง

ORDER1

SNO	PNO	PNAME	QTY
S1	P1	PEN	100
S1	P2	TABLE	200
S1	P3	DESK	300
S2	P1	PEN	300
S2	P3	DESK	400
S3	P4	BAG	100



รูปที่ 2.5 แสดงความสัมพันธ์แบบบางส่วน

จากรูปที่ 2.5 จะเห็นว่ารีเลชัน ORDER1 มีแอททริบิวต์ SNO และ PNO เป็นคีย์หลักที่สามารถระบุค่าของชื่อสินค้า (PNAME) และจำนวนสินค้าที่ถูกจัดส่ง(QTY) ขณะเดียวกันมีความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบบางส่วนเกิดขึ้น โดยแอททริบิวต์ PNO ซึ่งเป็นส่วนประกอบตัวหนึ่งของคีย์หลัก (SNO และ PNO) สามารถระบุค่าของชื่อสินค้าได้ (PNAME) ความสัมพันธ์ลักษณะนี้ก่อให้เกิดปัญหาในเรื่องของความซ้ำซ้อน และการปรับปรุงข้อมูล

ดังนั้น ความสัมพันธ์ลักษณะนี้จะไม่เกิดขึ้นกับรีเลชันที่มีแอททริบิวต์เดียวเป็นคีย์หลัก แต่เป็นกับคีย์หลัก และแอททริบิวต์ตัวใดตัวหนึ่งที่ประกอบเป็นคีย์หลักสามารถระบุค่าของแอททริบิวต์อื่นๆ ที่ไม่ใช่คีย์หลักได้ ดังตัวอย่างนี้

(SNO, PNO) -> (PNAME, QTY)

(PNO) -> PNAME

2.3.1.3. ความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบทรานซิทีฟ (Transitive Dependency)

แอททริบิวต์ที่มีคุณสมบัติเป็นคีย์หลักจะสามารถระบุค่าของทุกแอททริบิวต์ในแต่ละทูเปิลได้ อย่างไรก็ตาม ในบางรีเลชัน (ที่ออกแบบไม่เหมาะสม) อาจจะมีกรณีแอททริบิวต์อื่น (Nonkey Attribute) ารค่าไม่ซ้ำกันใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่สามารถระบุค่าของแอททริบิวต์อื่นๆ ในทูเพิลได้ ลักษณะของความสัมพันธ์ในการระบุค่าแอททริบิวต์แบบนี้ เรียกว่า ความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบทรานซิทีฟ (Transitive Dependency)

ตัวอย่างเช่น จากตารางที่ 2.8 รีเลชัน SUPPLIER1 ประกอบด้วย แอททริบิวต์ ดังต่อไปนี้

SNO	หมายถึง รหัสผู้ผลิต
SNAME	หมายถึง ชื่อผู้ผลิต
CITY	หมายถึง ชื่อจังหวัดที่ผู้ผลิตอยู่
RATING	หมายถึง การจัดอันดับความสะดวกในการขนส่งสินค้ามาให้ โดยใช้จังหวัดที่ผู้ผลิตอยู่เป็นตัวกำหนดความใกล้เคียงในการขนส่ง

SUPPLIER1

SNO	SNAME	CITY	RATING
S1	SERI	BANGKOK	2
S2	WANIDA	RAYONG	3
S3	SOMCHAI	RAYONG	3
S4	ORAPIN	BANGKOK	2
S5	TANACHOTE	PATUMTANE	1

ตารางที่ 2.8 รีเลชัน SUPPLIER1

รีเลชันนี้มีรหัสผู้ผลิตเป็นคีย์หลักที่สามารถระบุค่าของแอททริบิวต์อื่นๆ ในแต่ละทูเพิลได้ นอกจากนี้ยังพบว่า แอททริบิวต์ CITY ซึ่งไม่ได้เป็นคีย์หลักสามารถกำหนดค่าของแอททริบิวต์ RATING ว่าถูกจัดอันดับไว้ในอันดับใดบ้าง

SNO → (SNAME, CITY, RATING)

CITY → RATING

2.3.1.4. ความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบหลายค่า (Multivalued Dependency)

จากความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบฟังก์ชัน ที่กล่าวมาข้างต้น เป็นลักษณะที่แอททริบิวต์หนึ่งมีคุณสมบัติในการระบุค่าของแอททริบิวต์อื่นๆ ในแต่ละทูเพิลได้เพียงหนึ่งค่า ซึ่งคุณสมบัติของแอททริบิวต์ที่สามารถระบุค่าของแอททริบิวต์อื่นๆ ได้ คือ แอททริบิวต์ที่เป็นคีย์หลักหรือคีย์คู่แข่ง อย่างไรก็ตาม ในบางรีเลชันอาจจะมีกรณีของความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบหลายค่าเกิดขึ้นได้ โดยความสัมพันธ์นี้จะเกิดกับรีเลชันที่ประกอบด้วย แอททริบิวต์อย่างน้อยสามแอททริบิวต์และเป็นรีเลชันที่แอททริบิวต์หนึ่ง สามารถระบุค่าของแอททริบิวต์อื่นๆ (เช่น B หรือ C) ในรีเลชันได้มากกว่าหนึ่งค่า กรณีเช่นนี้เรียกว่ารีเลชันนั้นๆ มีความสัมพันธ์ในการระบุค่าของแอททริบิวต์แบบหลายค่า (Multivalued Dependency)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเช่น จากตารางที่ 2.9 รีเลชัน SPJC ประกอบด้วย แอททริบิวต์ ดังต่อไปนี้ คือ

SNO	หมายถึง รหัสผู้ผลิต
PJNO	หมายถึง รหัสโครงการที่ส่งผลิต
CITY	หมายถึง ชื่อจังหวัดที่ผู้ผลิตตั้งอยู่

ข้อสมมุติฐานของความสัมพันธ์ของข้อมูล คือ

- ให้ผู้ผลิตแต่ละรายสามารถรับงานได้หลายโครงการ และ
- ผู้ผลิตหนึ่งๆ มีโรงงานตั้งอยู่ที่หลายจังหวัด

SNO	PJNO	CITY
S1	PJ01	BANGKOK
S1	PJ01	SAMUTHPRAKARN
S1	PJ02	BANGKOK
S1	PJ02	SAMUTHPRAKARN
S2	PJ03	RAYONG
S2	PJ05	RAYONG
S3	PJ04	CHONBURE

ตารางที่ 2.9 รีเลชัน SPJC

รีเลชัน SPJC เป็นรีเลชันที่มีความสัมพันธ์ของแอททริบิวต์ในการระบุค่าได้หลายค่า นั่นคือ

SNO ระบุค่า PJNO ได้หลายค่า

SNO -> -> PJNO

SNO ระบุค่า CITY ได้หลายค่า

SNO -> -> CITY

ในรีเลชันนี้ เกิดความสัมพันธ์ในการระบุค่าของแอททริบิวต์แบบหลายค่าระหว่างแอททริบิวต์ SNO กับ PJNO และ SNO กับ CITY โดยที่แอททริบิวต์ PJNO และ CITY ไม่มีความสัมพันธ์ต่อกันเลย แต่มาปรากฏอยู่ในรีเลชันเดียวกัน (Independently Multivalued Dependency) ทำให้ค่าของ CITY ปรากฏซ้ำซ้อนอยู่ในหลายทิวเพิลเพื่อให้ข้อมูลของรีเลชันครบถ้วน ซึ่งก่อให้เกิดปัญหาในการเพิ่ม ลบ หรือปรับปรุงข้อมูลได้

รีเลชันที่มีความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบทรานซิทีฟ (Transitive Dependency)

และแบบหลายค่า (Multivalued Dependency) จะต้องผ่านกระบวนการทำรีเลชันให้อยู่ในรูปแบบเอกสารเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรทัดฐาน (Normalization) ซึ่งจะกล่าวต่อไป เพื่อไม่ให้เกิดการออกแบบฐานข้อมูลนั้นๆ มีปัญหาในด้านการเพิ่ม ลบ หรือปรับปรุงข้อมูลได้

2.3.2. การทำรีเลชันให้อยู่ในรูปแบบบรรทัดฐาน (Normalization)

แนวคิดในการทำรีเลชันให้อยู่ในรูปแบบบรรทัดฐาน (Normalization Process) ถูกคิดค้นโดย อี.เอฟ.คอดด์ (E.F.Codd) เป็นกระบวนการที่นำเค้าร่างของรีเลชันมาทำให้อยู่ในรูปแบบที่เป็นบรรทัดฐาน (Normal Form) เพื่อให้แน่ใจว่าการออกแบบเค้าร่างของรีเลชันเป็นการออกแบบที่เหมาะสม วัตถุประสงค์ของการทำให้เป็นบรรทัดฐาน มีดังนี้ คือ

2.3.2.1. เพื่อลดเนื้อหาในการจัดเก็บข้อมูล

การทำให้เป็นบรรทัดฐานเป็นการลดความซ้ำซ้อนของข้อมูลในรีเลชัน ซึ่งเป็นการลดเนื้อหาในการจัดเก็บข้อมูลได้

2.3.2.2. เพื่อลดปัญหาที่ข้อมูลไม่ถูกต้อง (Inconsistency)

เนื่องจากข้อมูลในรีเลชันหนึ่ง จะมีข้อมูลไม่ซ้ำกัน เมื่อมีการปรับปรุงข้อมูลก็จะปรับปรุงทีละครั้งเดียว ไม่ต้องปรับปรุงหลายแห่ง โอกาสที่จะเกิดความผิดพลาดในการปรับปรุงไม่ครบถ้วนก็จะไม่เกิดขึ้น

2.3.2.3. เป็นการลดปัญหาที่เกิดจากการเพิ่ม ปรับปรุงและลบข้อมูล

ช่วยแก้ปัญหาที่อาจจะเกิดขึ้นจากการปรับปรุงข้อมูลไม่ครบ หรือข้อมูลหายไปจากฐานข้อมูล หรือการเพิ่มข้อมูล

2.3.3. รูปแบบบรรทัดฐาน (Normal Form)

รูปแบบบรรทัดฐาน ที่ใช้ในการกำหนดแอททริบิวต์ที่เหมาะสมในรีเลชันแบ่งออกเป็น ดังนี้ คือ

2.3.3.1. รูปแบบบรรทัดฐานขั้นที่ 1 (First Normal Form : 1NF)

รีเลชันหนึ่งๆ จะอยู่ในรูปแบบบรรทัดฐานขั้นที่ 1 ก็ต่อเมื่อ

“ค่าของแอททริบิวต์หนึ่งในแต่ละทูเปิลจะมีค่าของข้อมูลเพียงค่าเดียว”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EXAMPLE

SNO	SNAME	CITY	PNO	QTY
S1	SERI	BANGKOK	P1	300
			P2	200
			P3	400
			P4	200
			P5	100
			P6	100
S2	WANIDA	RAYONG	P1	300
			P2	400
S3	SOMCHAI	RAYONG	P2	200
S4	ORAPIN	BANGKOK	P2	200
			P4	300
			P5	400

ตารางที่ 2.10 ก. รีเลชันที่ไม่อยู่ในรูปแบบบรรทัดฐานขั้นที่ 1

SNO	SNAME	CITY	PNO	QTY
S1	SERI	BANGKOK	P1	300
S1	SERI	BANGKOK	P2	200
S1	SERI	BANGKOK	P3	400
S1	SERI	BANGKOK	P4	200
S1	SERI	BANGKOK	P5	100
S1	SERI	BANGKOK	P6	100
S2	WANIDA	RAYONG	P1	300
S2	WANIDA	RAYONG	P2	400
S3	SOMCHAI	RAYONG	P2	200
S4	ORAPIN	BANGKOK	P2	200
S4	ORAPIN	BANGKOK	P4	300
S4	ORAPIN	BANGKOK	P5	400

ตารางที่ 2.10 ข. รีเลชันที่อยู่ในรูปแบบบรรทัดฐานขั้นที่ 1

จากตารางที่ 2.10 ก. สมมุติให้รีเลชัน EXAMPLE ประกอบด้วยแอททริบิวต์รหัสของผู้ผลิต S1 S2 และ S4 ที่มีข้อมูลของรหัสสินค้า (PNO) หลายค่า วิธีแก้ไขปัญหานี้คือ การใส่ข้อมูลของ S1 S2 และ S4 ลงไปในทุกรหัสสินค้า (PNO) ที่ผู้ผลิตขาย ดังตารางที่ 2.10 ข. โดยผลจากการใส่ข้อมูลดังกล่าวนี้จะเอกสารเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติเห็นไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้รหัสผู้ผลิต (SNO) ไม่ใช่คีย์หลัก แต่คีย์หลักจะประกอบด้วยรหัสผู้ผลิตและรหัสสินค้า (SNO และ PNO)

อย่างไรก็ตามเมื่อมีการกำหนดให้คีย์หลักเป็นรหัสผู้ผลิต และรหัสสินค้าเพื่อให้ข้อมูลในรีเลชันนี้อยู่ในรูปแบบบรรทัดฐานขั้นที่หนึ่งแล้วก็ตาม ความผิดพลาดบางอย่างที่อาจจะเกิดขึ้นกับข้อมูลในรีเลชันนี้ก็ยังมียู่ ดังนี้คือ

ก. ความผิดพลาดที่เกิดจากการเพิ่มข้อมูล (Insert Anomaly) จากรีเลชันนี้ จะเห็นว่ากรณีที่เพิ่มข้อมูลของผู้ผลิตจะทำให้ได้ต่อเมื่อผู้ผลิตรายนั้นได้มีการส่งสินค้าไปให้ผู้ซื้อ รีเลชันนี้จะไม่แสดงรายละเอียดของสินค้าของผู้ผลิต S5 ที่อยู่ทุหมุณนี้ถ้ายังไม่มีมีการส่งสินค้า ปัญหาก็คือ การกำหนดแอททริบิวต์ที่เป็นคีย์หลักของรีเลชัน เป็นการกำหนดที่ยังไม่เหมาะสม จากกฎความบูรณาภาพของเอนทิตี (The Entity Integrity Rule) ที่กล่าวในบทที่ 4 ได้ระบุไว้ว่า แอททริบิวต์ที่เป็นส่วนของคีย์หลักจะไม่มีค่าไม่ได้ แต่กรณีนี้รหัสสินค้า (PNO) จะเป็นค่าว่าง หากมีเพียงข้อมูลของผู้ผลิตแต่ไม่เคยได้รับการส่งสินค้า ก็จะเพิ่มข้อมูลได้ไม่ครบถ้วน

ข. ความผิดพลาดที่เกิดจากการลบข้อมูล (Delete Anomaly) ในการลบข้อมูลบางทูเฟิลทิ้งไปจะลบทั้งข้อมูลผู้ผลิตและข้อมูลการส่งสินค้า เช่น ข้อมูลของผู้ผลิตที่มีรหัส S3 และส่งสินค้ารหัส P2 การลบข้อมูลของทูเฟิลนี้ จะทำให้ข้อมูลของผู้ผลิตรหัส S3 หายไปจากฐานข้อมูล ปัญหาของรีเลชันนี้ คือรีเลชันนี้ประกอบด้วยแอททริบิวต์ที่เกินเงื่อนไขในการใช้งาน กล่าวอีกนัยหนึ่ง คือมีแอททริบิวต์มากเกินไปโดยที่ข้อมูลของบางแอททริบิวต์อาจจะไม่จำเป็นต้องใช้งานแต่มาผูกติดกับแอททริบิวต์ที่ต้องการใช้งาน

ค. ความผิดพลาดที่เกิดจากการปรับปรุงข้อมูล (Update Anomaly) การปรับปรุงข้อมูลของรีเลชันจะทำให้เกิดความยุ่งยากและเสียเวลา รวมถึงอาจให้เกิดความผิดพลาดที่ข้อมูลไม่เหมือนกัน ตัวอย่างเช่นการปรับปรุงข้อมูลของผู้ผลิต S1 โดยต้องการเปลี่ยนชื่อจังหวัดจาก BANGKOK เป็น AYUDTAYA ซึ่งการปรับเปลี่ยนข้อมูลนี้จะต้องค้นหารหัสผู้ผลิต S1 ทุกทูเฟิลมาปรับปรุง ทำให้เสียเวลาและอาจมีโอกาสดเกิดความผิดพลาดที่ผู้ผลิตรหัส S1 บางทูเฟิลไม่ได้ถูกเปลี่ยนเป็นชื่อจังหวัดใหม่ ปัญหาที่เกิดขึ้นของรีเลชัน EXAMPLE ที่กล่าวมาข้างต้น สามารถแก้ไขโดยแตกรีเลชัน (Decomposition) นี้เป็นสองรีเลชัน คือ รีเลชัน SUPPLIER และรีเลชัน ORDER ดังนี้

SUPPLIER ประกอบด้วย แอททริบิวต์ (SNO , SNAME, CITY) โดยมีแอททริบิวต์ SNO เป็นคีย์หลัก

ORDER ประกอบด้วยแอททริบิวต์ (SNO , PNO QTY) โดยมีแอททริบิวต์ SNO และ PNO เป็นคีย์หลักร่วมกัน

การแตกรีเลชันดังกล่าวนี้ จะแก้ปัญหาคความผิดพลาดทั้งสามประเภทที่กล่าวมาข้างต้นได้ เช่น การเพิ่มข้อมูลของผู้ผลิตได้โดยที่ผู้ผลิตนั้นยังไม่มีมีการส่งสินค้าหรือการลบข้อมูลการส่งสินค้าก็ไม่ทำให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลของผู้ผลิตหายไปจากฐานข้อมูล หรือการปรับปรุงข้อมูลของผู้ผลิตก็ทำเพียงครั้งเดียวเพราะไม่มีความซ้ำซ้อนของข้อมูล

2.3.3.2. รูปแบบบรรทัดฐานขั้นที่ 2 (Second Normal Form : 2NF)

รีเลชันหนึ่ง ๆ จะอยู่ในรูปแบบบรรทัดฐานขั้นที่ 2 ก็ต่อเมื่อ

"รีเลชันนั้น ๆ อยู่ในรูปแบบบรรทัดฐานขั้นที่ 1 และมีคุณสมบัติอีกประการหนึ่งคือ แอททริบิวต์ทุกแอททริบิวต์ที่ไม่ได้เป็นคีย์หลัก จะต้องมีความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบฟังก์ชันกับคีย์หลัก(Fully Functional Dependency) กล่าวอีกนัยหนึ่งก็คือค่าของแอททริบิวต์ที่ไม่ได้เป็นคีย์หลักจะสามารถระบุค่าโดยแอททริบิวต์ที่เป็นคีย์หลัก หรือโดยแอททริบิวต์ทั้งหมดที่ประกอบกันเป็นคีย์หลักในกรณีที่คีย์หลักเป็นคีย์ผสม" (ไม่มี Partial Dependency เกิดขึ้น)

จากรีเลชัน SUPPLIER และ ORDER รีเลชันทั้งสองอยู่ในรูปแบบบรรทัดฐานขั้นที่ 2 แล้ว ในรีเลชัน SUPPLIER มีรหัสผู้ผลิตเป็นคีย์หลัก เช่น เมื่อทราบค่า รหัสผลิต S1 ก็สามารถทราบค่าของชื่อและจังหวัดของผู้ผลิตคือ SERI อยู่ที่จังหวัด BANGKOK หรือในรีเลชัน ORDER จำนวนการส่งสินค้า (QTY) จะถูกระบุโดยรหัสผู้ผลิตและรหัสสินค้า ดังนั้น ค่าของแอททริบิวต์อื่น ๆ ที่ไม่ได้เป็นคีย์หลักของรีเลชัน SUPPLIER และ ORDER สามารถระบุโดยค่าของแอททริบิวต์ที่เป็นคีย์หลัก

แต่ถ้าหากสมมุติรีเลชัน ORDER1 ถูกออกแบบโดยประกอบด้วยแอททริบิวต์ ดังนี้

ORDER1

SNO	PNO	PNAME	QTY
S1	P1	PEN	100
S1	P2	TABLE	200
S1	P3	DESK	300
S2	P1	PEN	300
S2	P3	DESK	400
S3	P4	BAG	100

ตารางที่ 2.11 รีเลชัน ORDER1

จากตารางที่ 2.11 รีเลชัน ORDER1 มีแอททริบิวต์รหัสผู้ผลิต (SNO) และรหัสสินค้า (PNO) ประกอบกันเป็นคีย์หลัก ที่สามารถระบุค่าของชื่อสินค้า (PNAME) และจำนวนสินค้าที่ถูกจัดส่ง (QTY) และยังเกิดความสัมพันธ์ที่แอททริบิวต์ชื่อสินค้า (PNAME) ที่ไม่ได้เป็นคีย์หลัก สามารถระบุค่าโดยคีย์หลักที่เป็นคีย์ผสม (SNO และ PNO) และขณะเดียวกัน ยังสามารถระบุค่าโดยแอททริบิวต์ที่เป็นส่วนหนึ่งของคีย์หลักคือ รหัสสินค้า (PNO)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรุงเทพ ชื่อของผู้ผลิตอาจเป็น SERI หรือ ORAPIN ก็ได้ ซึ่งไม่สามารถกำหนดชัดเจนว่าชื่ออะไร
 แอททริบิวต์ CITY จึงไม่มีคุณสมบัติเป็นคีย์หลักหรือคีย์คู่แข่ง

นอกจากนี้ จะพบว่ามีความสัมพันธ์ระหว่างแอททริบิวต์ที่ไม่ได้เป็นคีย์หลักอยู่นั้นคือ แอททริบิวต์
 ชื่อจังหวัด (CITY) และแอททริบิวต์การจัดอันดับของผู้ผลิต (RATING) โดยแอททริบิวต์ชื่อจังหวัด
 สามารถระบุค่า การจัดอันดับความสะดวกในการขนส่งสินค้า ซึ่งเป็นความสัมพันธ์ระหว่างค่าของ
 แอททริบิวต์แบบทรานซิทีฟ (Transitive Dependency) หากรีเลชันใดมีคุณสมบัติดังกล่าว จะก่อให้เกิด
 ความผิดพลาดที่เกิดจากการเพิ่ม หรือลบหรือปรับปรุงข้อมูลได้ เช่น

ก. ความผิดพลาดที่เกิดจากการปรับปรุงข้อมูล (Update Anomaly) หากมีการแก้ไขการจัด
 อันดับของจังหวัดของผู้ผลิต เนื่องจากความสะดวกในการขนส่งที่เปลี่ยนแปลงไป จะต้องการแก้ไขข้อมูล
 หลายทิวเพิล ซึ่งอาจจะเกิดปัญหาการแก้ไขไม่ครบถ้วน อันจะก่อให้เกิดความไม่ตรงกันของข้อมูลได้

ข. ความผิดพลาดที่เกิดจากการลบข้อมูล (Delete Anomaly) หากมีการลบข้อมูลการจัดอันดับ
 ของจังหวัดของผู้ผลิต จะทำให้ข้อมูลการจัดอันดับของจังหวัดนั้นหายไปจากฐานข้อมูล เช่น หากลบทิวเพิล
 ของรหัสผู้ผลิต S5 จะไม่มีข้อมูลของผู้ผลิตที่อยู่จังหวัดปทุมธานีว่า ถูกจัดอันดับอยู่อันดับที่เท่าไร

ด้วยเหตุผลที่แอททริบิวต์ชื่อจังหวัด ไม่มีคุณสมบัติเป็นคีย์หลักในรีเลชัน SUPPLIER1 แต่
 สามารถกำหนดค่าของแอททริบิวต์การจัดอันดับได้ ซึ่งอาจก่อให้เกิดปัญหาความผิดพลาดจากการเพิ่ม
 ปรับปรุงหรือลบข้อมูล การจัดการปัญหาดังกล่าวจะทำได้โดยการแตกรีเลชัน SUPPLIER1 เป็นสองรีเลชัน
 (Decomposition) โดยแยกแอททริบิวต์ที่ถูกกำหนดค่ากับแอททริบิวต์ที่เป็นตัวกำหนดค่า (Determinant)
 ออกเป็นรีเลชันใหม่และกำหนดให้แอททริบิวต์ที่เป็นตัวกำหนดค่า เป็นคีย์หลักของรีเลชันใหม่ดังนี้

SUPPLIER (SNO , SNAME , CITY)
 CITY (CITY , RATING)

รีเลชัน SUPPLIER ประกอบด้วยแอททริบิวต์ (SNO , SNAME , CITY) มี SNO เป็นคีย์หลัก
 โดยมี CITY เป็นคีย์นอกที่อ้างอิงถึงแอททริบิวต์ CITY ที่อยู่ในรีเลชัน CITY ส่วนรีเลชัน CITY ประกอบด้วย
 แอททริบิวต์ (CITY , RATING) และมี CITY เป็นคีย์หลัก

วิธีการแตกรีเลชัน (Decomposition) ที่กล่าวมาข้างต้น จะต้องทำตามขั้นตอนดังนี้ คือ

ประการแรก ให้นำแอททริบิวต์ที่ถูกกำหนดค่ากับแอททริบิวต์ที่เป็นตัวกำหนดค่า แยกออกเป็น
 รีเลชันใหม่

ประการที่สอง คือกำหนดให้แอททริบิวต์ที่เป็นตัวกำหนดค่าเป็นคีย์หลักของรีเลชันใหม่นั้น

อย่างไรก็ตามในบางครั้ง ผู้ออกแบบเค้าร่างของฐานข้อมูลพยายามแตกรีเลชันที่มีแอททริบิวต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาก ๆ ออกเป็นหลาย ๆ รีเลชั่น โดยไม่ได้ปฏิบัติตามวิธีการที่กล่าวมาแล้วข้างต้น ซึ่งจะทำให้เกิดปัญหาการแตกฐานข้อมูลที่ไม่เหมาะสมได้ (Bad Decomposition) ตัวอย่างเช่นหากแตกรีเลชั่นของ SUPPLIER1 เป็นดังนี้

รีเลชั่น SUPPLIER (SNO SNAME CITY) มี SNO เป็นคีย์หลัก

รีเลชั่น SUPPLIER2 (SNO , RATING) มี SNO เป็นคีย์หลัก

SUPLIER

SNO	SNAME	CITY
S1	SERI	BANGKOK
S2	WANIDA	RAYONG
S3	SOMCHAI	RAYONG
S4	ORAPIN	BANGKOK
S5	TANACHOTE	PATUMTANE

SUPPLIER2

SNO	RATING
S1	2
S2	3
S3	3
S4	2
S5	1

ตารางที่ 2.13 แสดงการแตกรีเลชั่น

จากตารางที่ 2.13 การแตกรีเลชั่นจากตัวอย่างนี้ดูเหมือนจะแก้ปัญหาค่าความผิดพลาดที่อาจจะเกิดขึ้นจากการเพิ่ม ลบ หรือปรับปรุงข้อมูลได้ เพราะได้แตกรีเลชั่นออกเป็นสองรีเลชั่น และอยู่ในรูปแบบบรรทัดฐานขั้นที่ 3 แล้วแต่การแตกรีเลชั่นทั้งสองอาจจะก่อให้เกิดความผิดพลาดขึ้นได้อีกเช่นกัน จากตัวอย่างนี้ จะเห็นว่า การเพิ่มชื่อจังหวัด และการจัดอันดับของจังหวัดใหม่จะทำไม่ได้ จนกว่าจะมีผู้ผลิตรายใดอยู่ที่จังหวัดที่เพิ่มชื่อและจัดอันดับนี้ รีเลชั่น SUPPLIER และ รีเลชั่น SUPPLIER2 เป็นรีเลชั่นที่ไม่มีความเป็นอิสระจากกัน ทั้งนี้เพราะจังหวัดเป็นตัวกำหนดการจัดอันดับนั่นเอง

หากเปรียบเทียบการแตกรีเลชั่นในกรณีหลังกับกรณีแรก ดังรูปที่ 2.6 ก. คือ รีเลชั่น SUPPLIER และ CITY จะเห็นว่า การแตกรีเลชั่นทั้งสองจะเป็นอิสระจากกัน ทั้งนี้เพราะแอททริบิวต์รหัสผู้ผลิตเป็นตัวกำหนดแอททริบิวต์ชื่อจังหวัด และขณะเดียวกันในรีเลชั่น CITY แอททริบิวต์ชื่อจังหวัดเป็นคีย์หลัก และเป็นตัวกำหนดค่าการจัดอันดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.6 ก.	SNO -> CITY -> RATING
รูปที่ 2.6 ข.	SNO -> CITY -> RATING

ในทางตรงกันข้าม การแตกรีเลชันในกรณีหลังดังรูปที่ 2.6 ข. เป็นการแตกที่ไม่เป็นอิสระจากกัน ทั้งนี้เพราะแอททริบิวต์ CITY ยังเป็นตัวกำหนดค่าของแอททริบิวต์ CITY และ RATING อยู่นั่นเอง ดังนั้น หากต้องการแตกรีเลชันที่ไม่ก่อให้เกิดปัญหาดังกล่าวข้างต้น จะต้องคำนึงถึงขั้นตอนสองข้อที่กล่าวมาข้างต้น

2.3.3.4. รูปแบบบรรทัดฐานของบอยส์และคอดด์ (Boyce/Codd Normal Form : BCNF)

รีเลชันหนึ่ง ๆ จะอยู่ในรูปแบบบรรทัดฐานของบอยส์ และคอดด์ ก็ต่อเมื่อ

"รีเลชันนั้น ๆ อยู่ในรูปแบบบรรทัดฐานขั้นที่ 3 และไม่มีแอททริบิวต์อื่นในรีเลชันที่สามารถระบุค่าของแอททริบิวต์ที่เป็นคีย์หลักหรือส่วนหนึ่งส่วนใดของคีย์หลักในกรณีที่คีย์หลักเป็นคีย์ผสม"

โดยทั่วไป รูปแบบบรรทัดฐานของบอยส์และคอดด์จะอยู่ในรูปแบบบรรทัดฐานขั้นที่ 3 แต่ไม่จำเป็นเสมอไปว่ารูปแบบบรรทัดฐานขั้นที่ 3 จะอยู่ในรูปแบบของ BCNF ทั้งนี้เนื่องจากรูปแบบนี้เป็นรูปแบบที่ขยายขอบเขตของรูปแบบบรรทัดฐานขั้นที่ 3 ให้เหมาะสมยิ่งขึ้น โดยรูปแบบของรีเลชันที่จะต้องผ่านการทำให้เป็นบรรทัดฐาน BCNF มักจะมีคุณสมบัติดังนี้ คือ

"เป็นรีเลชันที่มีคีย์คู่แข่งหลายคีย์ (Multiple Candidate Key) โดยที่คีย์คู่แข่งเป็นคีย์ผสม (Composite Key) และคีย์คู่แข่งนั้นมีบางส่วนซ้ำกัน (Overlapped)"

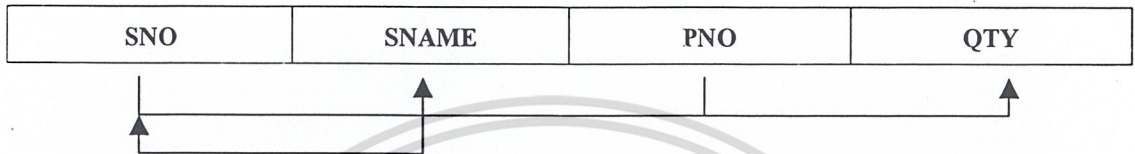
เพื่อให้ง่ายต่อการอธิบาย จะใช้รีเลชัน SUPPLIER3 เป็นตัวอย่างในการอธิบายโดยสมมุติว่าชื่อของผู้ผลิต (SNAME) เป็นค่าไม่ซ้ำกัน และมีคุณสมบัติเป็นคีย์หลักได้เช่นกัน ดังตารางที่ 2.14

SUPPLIER3

SNO	SNAME	PNO	QTY
S1	SERI	P1	300
S1	SERI	P2	200
S1	SERI	P3	400
S1	SERI	P4	200

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษายเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องยกย่องเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 2.14 เป็นรีเลชันที่มีปัญหาเพราะมีคีย์คู่แข่งเป็นคีย์ผสมและมีซ้ำซ้อนกัน คีย์คู่แข่งที่มีคุณสมบัติเป็นคีย์หลักของรีเลชัน อาจจะเป็นแอททริบิวต์รหัสผู้ผลิต และรหัสสินค้า (SNO และ PNO) หรือแอททริบิวต์ชื่อผู้ผลิตและรหัสสินค้า (SNAME และ PNO) รีเลชันนี้ไม่ได้อยู่ในรูปแบบบรรทัดฐาน BCNF เพราะเมื่อเลือกคีย์คู่แข่งใดเป็นคีย์หลักแล้ว คีย์คู่แข่งที่ไม่ได้ถูกเลือกจะยังปรากฏซ้ำซ้อนอยู่ ในรีเลชันนี้ และมีคุณสมบัติในการระบุค่าของแอททริบิวต์ที่เป็นคีย์หลัก ตัวอย่างเช่น หากเลือกแอททริบิวต์ SNO และ PNO เป็นคีย์หลักแล้วจะเกิดปัญหาในรีเลชันนี้คือ แอททริบิวต์ SNAME จะมีคุณสมบัติการระบุค่าของแอททริบิวต์ SNO ดังนี้

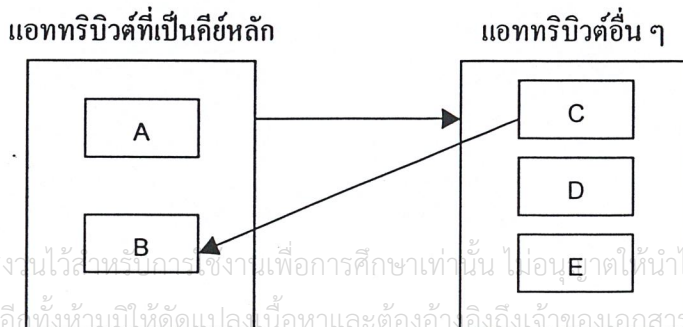


นอกจากนี้ รีเลชันยังเป็นรีเลชันที่ประกอบด้วยข้อมูลที่ซ้ำซ้อนกัน อันจะก่อให้เกิดความผิดพลาดที่เกิดจากการเพิ่ม แก้ไขหรือปรับปรุงข้อมูล ตัวอย่างเช่น การเปลี่ยนชื่อของผู้ผลิต S1 จะต้องแก้ไขหลายทิวเปิล ซึ่งอาจเกิดความผิดพลาดที่แก้ไขข้อมูลไม่ครบถ้วนได้

ดังนั้น รีเลชัน SUPPLIER3 จึงต้องทำการแตกรีเลชันออก โดยแยกแอททริบิวต์ที่สามารถระบุค่าของคีย์คู่แข่งแยกเป็นอีกรีเลชันหนึ่งซึ่งอาจทำได้สองกรณี คือ

- กรณีที่หนึ่ง SUPPLIER4 (SNO, SNAME)
 - มี SNO และ SNAME เป็นคีย์หลัก
 - ORDER2 (SNO, PNO, QTY)
 - มี SNO และ PNO เป็นคีย์หลัก
- หรือ กรณีที่สอง SUPPLIER4 (SNO, SNAME)
 - มี SNO และ SNAME เป็นคีย์หลัก
 - ORDER2 (SNAME, PNO, QTY)
 - มี SNAME และ PNO เป็นคีย์หลัก

กล่าวโดยสรุปรูปแบบ BCNF เป็นรูปแบบที่ขยายขอบเขตของรูปแบบบรรทัดฐานขั้นที่ 3 โดยมี การตรวจสอบรีเลชันว่ามีแอททริบิวต์อื่นที่ไม่ใช่คีย์หลัก แต่สามารถระบุค่าของคีย์หลักได้ ดังนี้



2.3.3.5. รูปแบบบรรทัดฐานขั้นที่ 4 (Fourth Normal Form : 4NF)

รีเลชันจะอยู่ในรูปแบบบรรทัดฐานขั้นที่ 4 ก็ต่อเมื่อ

"รีเลชันนั้น ๆ อยู่ในรูปแบบ BCNF และเป็นรีเลชันที่ไม่มีความสัมพันธ์ในการระบุค่าของแอททริบิวต์แบบหลายค่าโดยที่แอททริบิวต์ที่ถูกระบุค่าหลายค่าเหล่านี้ไม่มีความสัมพันธ์กัน (Independently Multivalued Dependency)"

รีเลชันที่ต้องผ่านการทำให้อยู่ในรูปแบบบรรทัดฐานขั้นที่ 4 นี้ จะต้องแอททริบิวต์อย่างน้อยสามแอททริบิวต์ ดังรูปที่ 1.21 SPJC เป็นรีเลชันที่ประกอบด้วยแอททริบิวต์รหัสผู้ผลิต (SNO) รหัสโครงการที่ส่งผลิต(PJNO) และชื่อจังหวัดที่โรงงานของผู้ผลิตตั้งอยู่ (CITY) โดยกำหนดให้ผู้ผลิตหนึ่งผลิตให้หลายโครงการ และผู้ผลิตหนึ่ง ๆ มีโรงงานตั้งอยู่ที่หลายจังหวัด (CITY)

SPJC

SNO	PJNO	CITY
S1	PJ01	BANGKOK
	PJ02	SAMUTHPRAKARN
S2	PJ03	RAYONG
	PJ04	CHONBURE
	PJ05	



ทำให้อยู่ในรูปแบบบรรทัดฐาน BCNF

SPJC

SNO	PJNO	CITY
S1	PJ01	BANGKOK
S1	PJ01	SAMUTHPRAKARN
S1	PJ02	BANGKOK
S1	PJ02	SAMUTHPRAKARN
S2	PJ03	RAYONG
S2	PJ03	CHONBURE
S2	PJ04	RAYONG
S2	PJ04	CHONBURE
S2	PJ05	RAYONG
S2	PJ05	CHONBURE

เอกสารนี้เป็นเอกสารที่สงวนรูปที่ 2.7 รีเลชัน SPJC ที่ถูกทำให้อยู่ในรูปแบบบรรทัดฐานขั้นที่ 4 ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากจากรูปที่ 2.7 รีเลชั่น SPJC ประกอบด้วยแอททริบิวต์รหัสผู้ผลิต (SNO) ซึ่งมีความสัมพันธ์ในการระบุแอททริบิวต์รหัสโครงการที่ส่งผลิต (PJNO) และแอททริบิวต์จังหวัดที่โรงงานของผู้ผลิตตั้งอยู่ (CITY) ได้ หลายค่า รีเลชั่น SPJC อยู่ในรูปแบบ BCNF เมื่อมีแอททริบิวต์ทุกแอททริบิวต์เป็นคีย์ นอกจากนี้ รหัสผู้ผลิต (SNO) มีความสัมพันธ์ในการระบุค่าของรหัสโครงการที่ส่งผลิต (PJNO) แบบหลายค่า (SNO->-> PJNO) ในขณะที่เดียวกัน รหัสผู้ผลิต (SNO) ก็มีความสัมพันธ์ในการระบุค่าแบบหลายค่ากับชื่อจังหวัดที่ผู้ผลิตตั้งอยู่ (CITY) การที่แอททริบิวต์ทั้งสองไม่มีความสัมพันธ์กันแต่มาอยู่ในรีเลชั่นเดียวกันจะก่อให้เกิดความซ้ำซ้อนของข้อมูลเกิดขึ้น เพราะค่าของชื่อจังหวัดจะต้องปรากฏในรีเลชั่นซ้ำกันเพื่อให้ข้อมูลในแต่ละทูเปิลมีข้อมูลครบถ้วนถึงแม้ว่ารีเลชั่น SPJC จะอยู่ในรูปแบบของ BCNF โดยมีแอททริบิวต์ทุกแอททริบิวต์ประกอบกันเป็นคีย์หลักก็ตาม รีเลชั่นนี้ก็ยังมีปัญหาแฝงอยู่เช่นกัน

ปัญหาความผิดพลาดที่เกิดขึ้นจากการเพิ่ม ลบ หรือปรับปรุงข้อมูลในรีเลชั่น SPJC มีโอกาสที่เกิดขึ้นมากมาย เช่น หากผู้ผลิตรหัส S2 ย้ายโรงงาน RAYONG ไปอยู่จังหวัดอื่นก็ตามทำการแก้ไขข้อมูลหลายทูเปิลหรือความผิดพลาดที่เกิดจากการเพิ่มข้อมูลอาจจะทำไม่ได้ เช่น การเก็บข้อมูลชื่อจังหวัดที่ตั้งของผู้ผลิตคนใหม่ที่ยังไม่เคยได้รับงาน โครงการใดจะทำไม่ได้ เป็นต้น การแก้ไขปัญหาทำได้โดยการแตกรีเลชั่น SPJC ออกเป็นสองรีเลชั่น ดังนี้

รีเลชั่น SPJ ประกอบด้วย แอททริบิวต์ SNO และ PJNO และมีแอททริบิวต์ทั้งสองประกอบกับคีย์หลัก ดังนี้ คือ

SPJ

SNO	PNO
S1	PJ01
S1	PJ02
S2	PJ03
S2	PJ04
S2	PJ05

รีเลชั่น SC ประกอบด้วย แอททริบิวต์ SNO และ CITY และมีแอททริบิวต์ทั้งสองประกอบกันเป็นคีย์หลัก ดังนี้ คือ

SC

SNO	CITY
S1	BANGKOK
S1	SUMUTHPRAKARN
S2	RAYONG
S2	CHONBURE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3.6. รูปแบบบรรทัดฐานขั้นที่ 5 (Fifth Normal Form : 5NF)

รีเลชันหนึ่ง ๆ จะอยู่ในรูปแบบบรรทัดฐานขั้นที่ 5 หรือที่เรียกว่า Project-Join Normal Form (PJ/NF) ก็ต่อเมื่อ

"รีเลชันนั้นอยู่ในรูปแบบบรรทัดฐานขั้นที่ 4 และเป็นรีเลชันที่มีคีย์หลักเป็นคีย์ผสมที่ประกอบด้วยแอททริบิวต์ตั้งแต่สามแอททริบิวต์เป็นต้นไป หากมีการแตกรีเลชันออกเป็นรีเลชันย่อยสามรีเลชันย่อย (หรือมากกว่า) (Projection) ซึ่งเกิดจากการจับคู่แอททริบิวต์แต่ละคู่ของรีเลชันเดิมเป็นคีย์ผสม และเมื่อทำการเชื่อมโยงรีเลชันย่อยทั้งหมด (Join) จะไม่ก่อให้เกิดข้อมูลใหม่ที่ไมเหมือนรีเลชันเดิม"

จากตารางที่ 2.15 รีเลชัน SPP ประกอบด้วยแอททริบิวต์ชื่อผู้ผลิต (SNAME) ชื่อสินค้า (PNAME) และชื่อ โครงการ (PJNAME) โดยมีแอททริบิวต์ทั้งสามเป็นคีย์หลัก

SPP

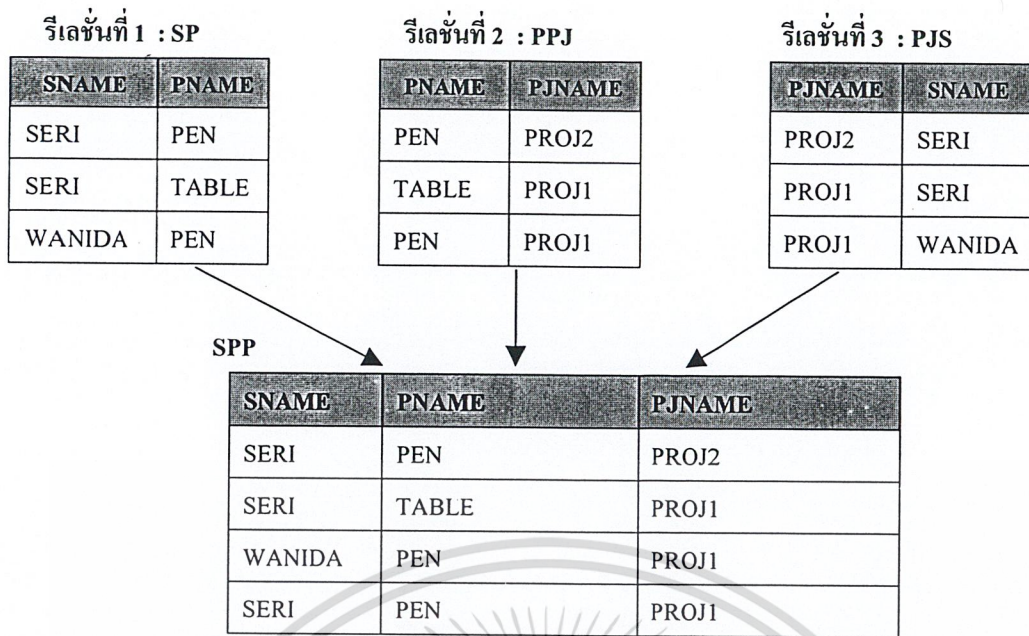
SNAME	PNAME	PJNAME
SERI	PEN	PROJ2
SERI	TABLE	PROJ1
WANIDA	PEN	PROJ1
SERI	PEN	PROJ1

ตารางที่ 2.15 รีเลชัน SPP

รีเลชัน SPP อยู่ในรูปแบบบรรทัดฐานขั้นที่ 4 แต่ยังคงประสบกับปัญหาในการปรับปรุงแก้ไขข้อมูล เช่น การลบข้อมูลของทูเพิลที่ 3 (WANIDA, PEN, PROJ1) ข้อมูลของชื่อผู้ผลิตชื่อ WANIDA ก็จะหายไปจากฐานข้อมูล หรือหากมีการเปลี่ยนชื่อสินค้าใดสินค้าหนึ่ง ก็จะต้องทำการปรับปรุงข้อมูลหลายทูเพิลเป็นต้น

เพื่อหลีกเลี่ยงปัญหาดังกล่าว จึงทำการแตกรีเลชัน SPP เป็นสามรีเลชันย่อย (PROJECTION) โดยใน แต่ละรีเลชันย่อยจะประกอบด้วยแอททริบิวต์แต่ละคู่เป็นคีย์ผสมของรีเลชันย่อย ดังรูปที่ 2.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงการแตกรีเลชัน SPP

เมื่อแตกรีเลชัน SPP ออกเป็นรีเลชัน SP และ PPJ แล้ว หากนำรีเลชันทั้งสามมาเชื่อมโยงกัน (JOIN) จะมีข้อมูลเหมือนในรีเลชันเดิมคือ SPP ไม่มีลักษณะของทูปเล็ตที่เกิดขึ้นที่เรียกว่า Spurious Tuple ดังรูปที่ 2.8

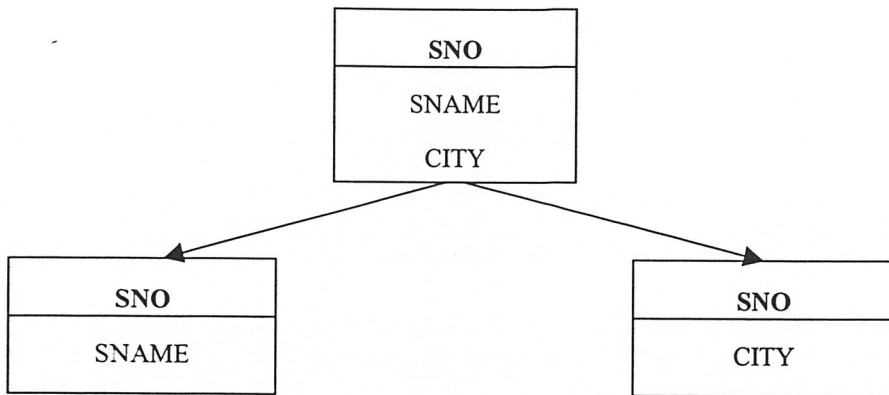
ดังนั้น เมื่อมีการแตกรีเลชันออกมา แล้วทำการเชื่อมโยงรีเลชันย่อยนั้นใหม่ หากไม่มีข้อมูลที่แตกต่างไปจากรีเลชันเดิม ก็สามารถจะแตกรีเลชันนั้นได้ แต่ถ้าหากแตกเป็นรีเลชันย่อยแล้วเกิดข้อมูลไม่เหมือนกับรีเลชันเดิม ก็ไม่ควรแตกรีเลชัน

2.3.4. ประเด็นที่ควรคำนึงถึงในการทำให้รูปแบบบรรทัดฐาน

2.3.4.1. การแตกรีเลชันมากเกินไป (Over normalization)

วัตถุประสงค์หนึ่งของการทำให้รูปแบบบรรทัดฐานก็คือ เพื่อลดปัญหาในด้านความซ้ำซ้อนของข้อมูล และปัญหาในเรื่องการเพิ่ม การปรับปรุง หรือลบข้อมูล โดยทั่วไปแล้วออกแบบในระดับแนวคิดผู้ออกแบบจะพยายามวิเคราะห์รีเลชันให้อยู่ในรูปแบบขั้นที่ 3 แต่ถ้ากรณีของปัญหาที่จำเป็นต้องทำต่อไปถึงรูปแบบบรรทัดฐานของบอส์และคอตต์ หรือขั้นที่ 4 หรือขั้นที่ 5 (ซึ่งเกิดขึ้นน้อยมากในทางปฏิบัติ) แต่อย่าพยายามแตกรีเลชันให้มากเกินไปจนเกิดความจำเป็น (Over normalization) เพราะการแตกรีเลชัน ออกเป็นรีเลชันย่อยมากเกินไป มีผลต่อประสิทธิภาพในการทำงานของฐานข้อมูลนั้น ๆ เช่น ในการค้นหา ข้อมูล จะใช้เวลามาก เป็นต้น

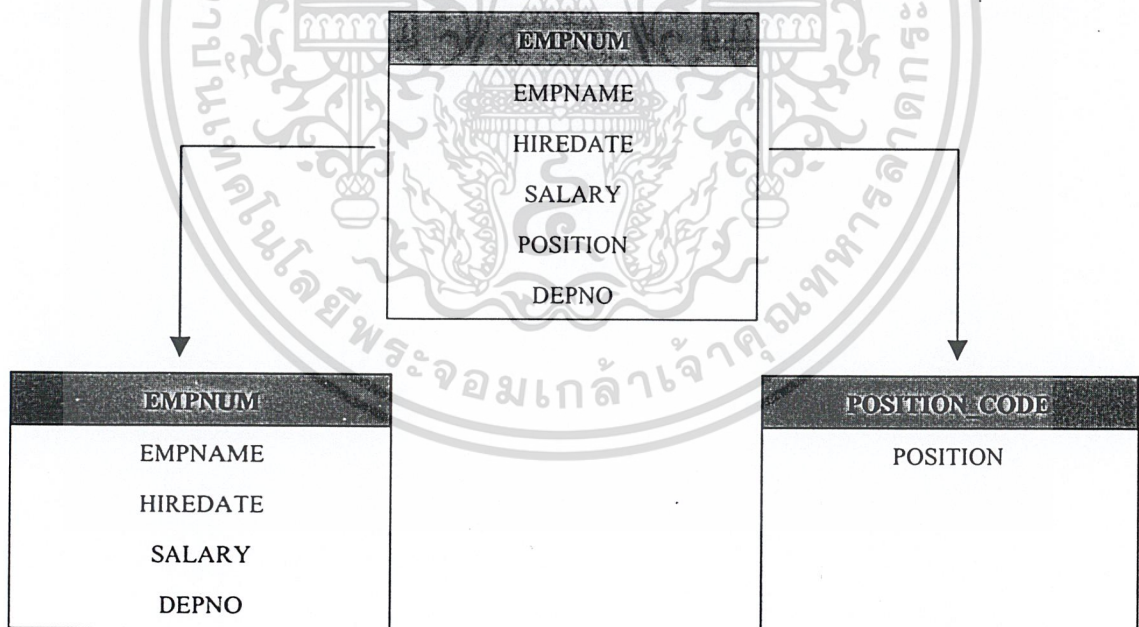
ตัวอย่างเช่น รีเลชัน SUPPLIER อยู่ในรูปแบบบรรทัดฐานขั้นที่ 3 และไม่มีความจำเป็นต้องทำให้อยู่ในรูปแบบบรรทัดฐานขั้นต่อไป สมมุติว่า มีการแตกรีเลชัน SUPPLIER ย่อยออกเป็นสองรีเลชัน การกระทำดังกล่าวไม่ผิด แต่ไม่มีความจำเป็นและไม่ควรทำอย่างยิ่ง เพราะจะทำให้ฐานข้อมูลมีรีเลชันมากเกินไป การค้นหาหรือเรียกดูข้อมูลจะต้องเชื่อมโยงข้อมูลจากหลายรีเลชัน จะเสียเวลามาก ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



2.3.4.2. การคืนอร์มอลไลเซชัน (Denormalization)

ในกรณีที่บางรีเลชันถูกออกแบบโดยการไม่ทำให้อยู่ในรูปแบบบรรทัดฐานที่เป็นไปตามกฎเกณฑ์ที่กำหนดไว้ เช่น รีเลชันนั้น ๆ ควรจะปรับให้อยู่ในรูปแบบบรรทัดฐานขั้นที่สาม แต่หยุดอยู่เพียงรูปแบบบรรทัดฐานขั้นที่ 2 เป็นต้น ทั้งนี้ อาจเป็นเพราะเหตุผลในเรื่องของประสิทธิภาพในการเรียกดูหรือค้นหาข้อมูล และยอมให้เกิดความซ้ำซ้อนของข้อมูลได้

ตัวอย่างเช่นรีเลชัน EMPLOYEE ได้รวมแอททริบิวต์ POSITION ไว้ แทนที่จะแยกเป็นรีเลชันหนึ่งทีแสดงถึงรหัสตำแหน่ง และชื่อตำแหน่ง ทั้งนี้เพราะตำแหน่งของพนักงานมีไม่กี่ประเภทเท่านั้น



ด้วยเหตุที่การคืนอร์มอลไลเซชัน อาจก่อให้เกิดปัญหาความซ้ำซ้อนของข้อมูลเกิดขึ้นได้ จึงควรมีการระบุนสาเหตุ และวิธีการในการปรับปรุงข้อมูลในโปรแกรมประยุกต์ใช้งาน เพื่อป้องกันไม่ให้เกิดปัญหาข้อมูลไม่ถูกต้อง อีกประเด็นซึ่งอาจใช้เป็นแนวทางว่าจะยอมให้มีการคืนอร์มอลไลเซชันหรือไม่ก็คือ ถ้าข้อมูลในรีเลชันนั้น ๆ ส่วนใหญ่จะเป็นการเรียกดูข้อมูล (Select) มากกว่าการเพิ่ม ปรับปรุง หรือลบข้อมูลซ้ำ ไม่ว่าจะเป็นกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็อาจจะคืนฟอร์มอลไลเซชัน ได้ถ้าคิดว่าการออกแบบลักษณะนี้จะเพิ่มประสิทธิภาพในการทำงานของฐานข้อมูล และไม่มีปัญหาด้านความไม่ถูกต้องของข้อมูลที่ซ้ำซ้อนกันได้

2.4. การสร้างโมเดลจำลองความสัมพันธ์ระหว่างข้อมูล : E-R โมเดล

โมเดลที่ใช้เป็นเครื่องในการจำลองความสัมพันธ์ระหว่างข้อมูลมีหลายโมเดล เช่น E-R โมเดล (Entity Relationship Model) คิดค้นโดยคอดด์ (E.F. Codd) หรือ NI-AM โมเดล (Nijssen's Information Analysis Methodology) คิดค้นโดย Prof. G.M. Nijssen โมเดลที่จะกล่าวถึงในบทนี้จะยกตัวอย่างของ E-R โมเดล เพื่ออธิบายถึงแนวคิดของโมเดลนี้ที่ช่วยในการออกแบบข้อมูลเพื่อประโยชน์ในการออกแบบระดับแนวคิด (Conceptual Level) ซึ่งโมเดลนี้เป็นที่รู้จักกันอย่างแพร่หลายในเมืองไทย และเป็นโมเดลที่ระบบจัดการฐานข้อมูลหลายชนิดมีเครื่องมือช่วยในการออกแบบ

โมเดลนี้เป็นแนวคิดที่ใช้เป็นเครื่องมืออย่างหนึ่งในการออกแบบข้อมูลในระดับแนวคิด (Conceptual Data Modeling) โดยแสดงถึงรายละเอียดและความสัมพันธ์ของข้อมูลต่าง ๆ ในระบบลักษณะที่เป็นภาพรวม ซึ่งเป็นประโยชน์ในด้านการรวบรวมและวิเคราะห์ข้อมูลว่ารายละเอียดและความสัมพันธ์อะไรบ้าง

2.4.1. แนวคิดเกี่ยวกับ E-R โมเดล

ในการสร้างโมเดลจำลองความสัมพันธ์ระหว่างข้อมูลมีแนวคิดที่เกี่ยวข้องคือ ความหมายของเอนทิตี คุณลักษณะของเอนทิตี ความสัมพันธ์ระหว่างเอนทิตี และสัญลักษณ์ที่ใช้ใน E-R โมเดล

2.4.1.1. เอนทิตี (Entity)

ความหมายของเอนทิตีได้เคยกล่าวอย่างละเอียดมาแล้วในบทที่ 2 ซึ่งผู้ที่จะสร้างโมเดลจำลองความสัมพันธ์ระหว่างข้อมูล จะต้องกำหนดว่าประเภทของเอนทิตีต่าง ๆ ที่ควรจะต้องมีในระบบงานที่กำลังออกแบบอยู่ว่ามีอะไรบ้าง เช่น ในระบบฐานข้อมูลบริษัทประกอบด้วยเอนทิตีประวัติพนักงาน เอนทิตีแผนก เอนทิตีโครงการ เอนทิตีงานที่ได้รับมอบหมาย และเอนทิตีประวัติครอบครัวของพนักงาน

นอกจากนี้ จะต้องพิจารณาว่าเอนทิตีใดมีคุณลักษณะที่จัดเป็น Supertype หรือ Subtype เป็นกรณีเฉพาะลงไปด้วย

2.4.1.2. คุณลักษณะของเอนทิตี (Property)

คุณลักษณะของเอนทิตี คือ รายละเอียดข้อมูลในเอนทิตีหนึ่ง ๆ ตัวอย่างเช่น เอนทิตีประวัติพนักงาน ประกอบด้วยรายละเอียดของรหัสพนักงาน ชื่อพนักงาน เงินเดือน เป็นต้น กล่าวอีกนัยหนึ่งคุณลักษณะของเอนทิตี ก็คือ แอททริบิวต์ที่ใช้ในฐานข้อมูลเชิงสัมพันธ์นั่นเอง ดังนั้น คุณลักษณะของเอนทิตีอาจประกอบด้วยรายละเอียดของข้อมูลที่อาจมีคุณสมบัติคล้ายคลึงกับแอททริบิวต์ เช่น เป็นข้อมูลผสม(Composite) เป็นคีย์ (Key) หรือเป็นข้อมูลที่แปลค่ามา (Derived) เป็นต้น

ในการสร้างโมเดลจำลองความสัมพันธ์ระหว่างข้อมูลจะต้องกำหนดคุณลักษณะของเอนทิตีและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า
 ทรานซาคชันของรายละเอียดยังคงกล่าวว่ามีคุณสมบัติอย่างไรบ้าง
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.3. ความสัมพันธ์ระหว่างเอนทิตี

ในฐานะข้อมูลหนึ่ง ๆ อาจประกอบด้วยหลายเอนทิตี ซึ่งข้อมูลของแอททริบิวต์เหล่านี้มีความสัมพันธ์กันในการสร้างโมเดลจำลองความสัมพันธ์ระหว่างข้อมูล จะต้องระบุว่าเอนทิตีต่าง ๆ มีความสัมพันธ์กันอย่างไรบ้าง เพื่อประโยชน์ในการกำหนดโครงสร้างการเชื่อมโยงข้อมูล และลดความซ้ำซ้อนของข้อมูล

ลักษณะของความสัมพันธ์ระหว่างเอนทิตี อาจเป็นแบบหนึ่งต่อหนึ่ง (1:1) หนึ่งต่อกลุ่ม (1:N) หรือ กลุ่มต่อกลุ่ม (M:M) หรือแม้แต่ความสัมพันธ์กับเอนทิตีของตัวเอง (Recursive หรือ Self Relationships) และความสัมพันธ์ระหว่าง Super type กับ Subtype ที่เคยกล่าวมาแล้ว

นอกจากนี้ อาจคำนึงถึงความสัมพันธ์ของข้อมูลระหว่างเอนทิตีโดยคำนึงถึงการที่ข้อมูลทุกข้อมูลของคอนทิตีหนึ่งมีความสัมพันธ์กับข้อมูลอีกเอนทิตีหนึ่ง หรือการที่ข้อมูลเพียงบางข้อมูลของเอนทิตีหนึ่งมีความสัมพันธ์กับข้อมูลของอีเอนทิตีหนึ่ง (Participation Constraint) ข้อจำกัดของความสัมพันธระหว่างเอนทิตีในลักษณะนี้แบ่งออกได้เป็น 2 ประเภท คือ

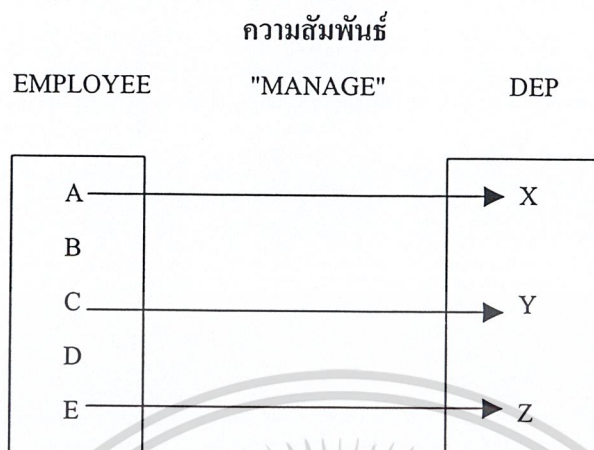
ก. แบบ Total Participation หมายถึง ข้อมูลทุกข้อมูลของเอนทิตีหนึ่งจะต้องมีความสัมพันธ์กับข้อมูลของอีกเอนทิตีหนึ่งตามประเภทของความสัมพันธ์ที่กำหนดขึ้นเสมอ เช่น จากรูปที่ 2.9 ถ้าบริษัทกำหนดว่าพนักงานแต่ละคนจะต้องสังกัดในแผนกหนึ่งแผนกใดของบริษัท นั่นหมายความว่า พนักงานแต่ละคนจะต้องมีข้อมูลการทำงานสังกัดในแผนกใดแผนกหนึ่งปรากฏอยู่ ความสัมพันธ์แบบนี้เรียกว่า Existence Dependency ในกรณีของเอนทิตีประเภทอ่อนแอ เอนทิตีอ่อนแอจะมีความสัมพันธ์กับอีกเอนทิตีหนึ่งที่มีความหมายกับมัน (Owner Entity หรือ Characteristic Entity) ในลักษณะนี้เช่นเดียวกัน ความสัมพันธ์ระหว่างเอนทิตีอ่อนแอกับเอนทิตีที่มีความหมายกับมันมีชื่อเรียกเฉพาะว่า Identifying Relationship Type



รูปที่ 2.9 ความสัมพันธ์แบบ Total Participation

ข. แบบ Partial Participation หมายถึง ข้อมูลเพียงบางข้อมูลของเอนทิตีหนึ่งมีความสัมพันธ์กับข้อมูลของอีกเอนทิตีหนึ่ง ตามประเภทของความสัมพันธ์ที่กำหนดขึ้น เช่นจากรูป 2.10 หากกำหนดความเอกสารสัมพันธ์ระหว่างพนักงานบางคนเท่านั้นที่เป็นระดับผู้บริหารของแผนก ดังนั้นข้อมูลของพนักงานบางคนไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่านั้นที่เป็นผู้บริหารจะสัมพันธ์กับข้อมูลของแผนกว่าใครทำหน้าที่เป็นผู้จัดการแผนกอยู่ในขณะนั้นไม่ใช้ข้อมูลของพนักงานทุกคนที่จะสัมพันธ์กับข้อมูลแผนก



รูปที่ 2.10 ความสัมพันธ์แบบ *Partial Participation*

2.4.2. ขั้นตอนในการออกแบบฐานข้อมูลโดยใช้ E-R โมเดล

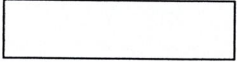
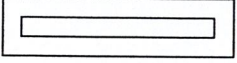








ขั้นตอนในการออกแบบข้อมูลโดยใช้ E-R โมเดล ประกอบด้วยขั้นตอนดังนี้คือ

1. ศึกษาถึงลักษณะหน้าที่งานของระบบ (Business Function) ว่า มีรายละเอียดของการทำงาน และข้อมูลที่เกี่ยวข้องอะไรบ้าง มีข้อสมมุติฐาน (Business Rule) ของงานต่าง ๆ อะไรบ้าง
2. กำหนดเอนทิตีที่จะอยู่ในฐานข้อมูล ฐานข้อมูลหนึ่ง ๆ ประกอบด้วยหลายเอนทิตี ในการกำหนดเอนทิตีที่จะอยู่ในฐานข้อมูลหนึ่ง ๆ จะต้องคำนึงรวมไปถึงว่าเอนทิตีนั้น ๆ เป็นเอนทิตีประเภทอ่อนแอ(Weak entity) หรือเป็นเอนทิตีประเภทที่ควรแบ่งเป็น Supertype หรือ Subtype หรือไม่
3. กำหนดประเภทของความสัมพันธ์ระหว่างเอนทิตีว่ามีความสัมพันธ์อย่างไรบ้าง โดยพิจารณาจากข้อสมมุติฐานของความสัมพันธ์ระหว่างเอนทิตีที่ได้ศึกษามา
4. กำหนดคุณสมบัติของเอนทิตีว่าควรจะมีรายละเอียดอะไรบ้าง ซึ่งการกำหนดคุณลักษณะของเอนทิตี จะพิจารณาว่ารายละเอียดต่าง ๆ เป็นรายละเอียดที่มีคุณสมบัติเป็นคีย์ หรือเป็นรายละเอียดที่แปลค่ามาหรือเป็นรายละเอียดที่ประกอบด้วยรายละเอียดที่เป็นข้อมูลผสม เช่น ที่อยู่ ประกอบด้วย บ้านเลขที่ ถนน เขตตำบล จังหวัด รหัสไปรษณีย์ เป็นต้น
5. กำหนดคีย์ของแต่ละเอนทิตีว่า จะใช้รายละเอียดของข้อมูลใดเป็นคีย์หลักของเอนทิตีนั้น ๆ ซึ่งจะต้องเป็นรายละเอียดของข้อมูลที่มีค่าเป็นเอกลักษณ์ หรือค่าเฉพาะไม่ซ้ำซ้อนในเอนทิตีนั้น ๆ
6. นำรายละเอียดตั้งแต่ขั้นตอนที่ 2 ถึง 5 มาพิจารณาทบทวนอีกครั้ง หลังจากนั้นก็เขียน E-R โมเดลโดยใช้สัญลักษณ์ที่กล่าวมาข้างต้นเป็นการอธิบายความสัมพันธ์ระหว่างข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3. สัญลักษณ์ที่ใช้ใน E-R โมเดล

ในการสร้าง E-R โมเดล เครื่องหมายและความหมายของสัญลักษณ์ที่ใช้ในโมเดลนี้มีดังต่อไปนี้

เครื่องหมาย	ความหมาย
	เอนทิตี
	เอนทิตีชนิดอ่อนแอ
	ประเภทของความสัมพันธ์ระหว่างเอนทิตี
	ประเภทของความสัมพันธ์ระหว่างเอนทิตีหนึ่งกับ เอนทิตีอ่อนแอ (Weak Entity) (Identifying Relationship Type)
	แอททริบิวต์
	แอททริบิวต์คีย์หลัก
	แอททริบิวต์ผสม
	แอททริบิวต์ที่แปลค่ามา
	การแสดงความสัมพันธ์ระหว่างเอนทิตี (Cardinality Ratio)
	การแสดงความสัมพันธ์ของเอนทิตี E2 ที่มีต่อ แบบ Total Participation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4. การแปลง E-R โมเดลให้อยู่ในรูปแบบฐานข้อมูลเชิงสัมพันธ์

เนื่องจาก E-R โมเดล เป็นการจำลองความสัมพันธ์ระหว่างข้อมูลโดยผลที่ได้จะทราบว่า ในฐานะข้อมูลนั้น ๆ ประกอบด้วย เอนทิตี รายละเอียดของเอนทิตีอะไรบ้าง การแปลงความสัมพันธ์และ E-R โมเดลให้อยู่ในรูปแบบฐานข้อมูลเชิงสัมพันธ์ โดยการวิเคราะห์นี่จะเป็นผลเพื่อนำไปใช้ภาษากฎนิยามฐานข้อมูลในการสร้างรีเลชัน โดยใช้ภาษาในระดับการฐานข้อมูลนั้น ๆ การแปลง ประกอบด้วยขั้นตอนดังต่อไปนี้ คือ

แปลงเอนทิตีต่าง ๆ ใน E-R โมเดลให้เป็นรีเลชัน และแปลงประเภทของความสัมพันธ์ของเอนทิตีเป็นความสัมพันธ์ของรีเลชัน

ในกรณีที่มีความสัมพันธ์ระหว่างเอนทิตีเป็นแบบ M:N จะนิยามกำหนดรีเลชันใหม่ขึ้น เพื่อแปลงความสัมพันธ์ให้อยู่ในรูป 1:N

กล่าวโดยสรุป E-R โมเดล เป็นการสร้างโมเดลจำลองความสัมพันธ์ระหว่างข้อมูลเบื้องต้น โดยที่ผู้สร้างโมเดล จะต้องนำแนวคิดในการออกแบบฐานข้อมูลมาใช้ร่วมกัน เพื่อให้รายละเอียดของข้อมูลที่จะแปลงไปเป็นฐานข้อมูลรูปแบบต่าง ๆ ประกอบด้วยข้อมูลที่เหมาะสม เช่น อยู่ในรูปแบบบรรทัดฐานขั้นที่สาม เป็นอย่างน้อย เป็นต้น ในปัจจุบัน ระบบจัดการฐานข้อมูลบางระบบมีเครื่องมือที่ช่วยในการสร้าง E-R โมเดลนี้และมี CASE TOOL(Computer Aided Software Engineering) ที่ช่วยในการแปลง E-R โมเดลให้อยู่ในรูปแบบฐานข้อมูลที่ใช้ให้โดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

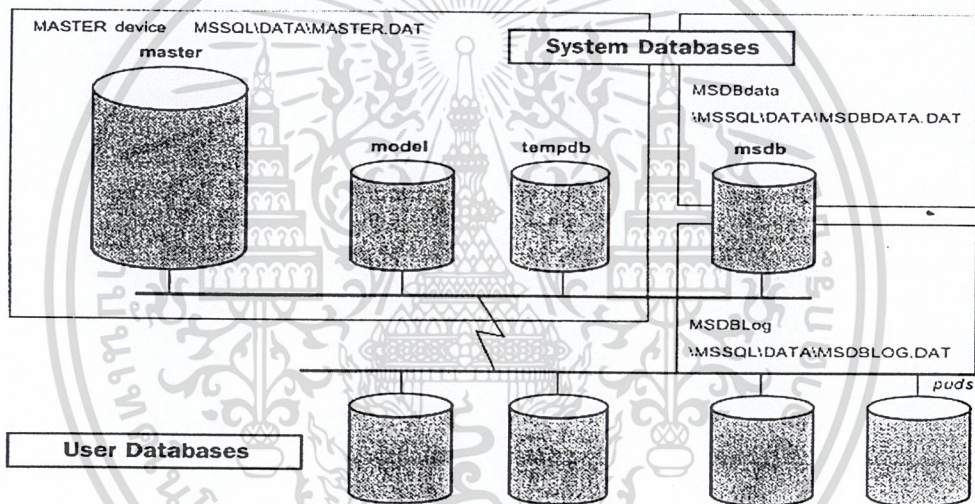
บทที่ 3

การใช้งาน Microsoft SQL Server 6.5 เบื้องต้น

3.1.สถาปัตยกรรมระบบฐานข้อมูล

ระบบฐานข้อมูลของ Microsoft SQL Server จะมีสถาปัตยกรรมฐานข้อมูลอยู่ 2 แบบคือ System Database (ประกอบด้วย master, model, tempdb, msdb) และ User Database SQL Server จะใช้ System Database ในการปฏิบัติการและจัดการระบบ ผู้บริหารระบบสามารถจะจัดการกับ System Database และเรียกดูข้อมูลใน System table ซึ่งจะเก็บรายละเอียดฐานข้อมูลอ็อบเจ็กต์ต่าง ๆ อินฟอร์มเมชันของผู้ใช้ ฯลฯ

Microsoft SQL Server จะติดตั้งระบบฐานข้อมูล 4 ระบบ คือ master model tempdb และ msdb ระบบ toured procedure จะใช้วิธีข่าวสารเกี่ยวกับข้อมูลที่อยู่ในตาราง ซึ่งตารางจะเก็บบันทึกข่าวสารอินฟอร์มเมชันเกี่ยวกับ SQL Server ทั้งหมดและข้อมูลของผู้ใช้งาน



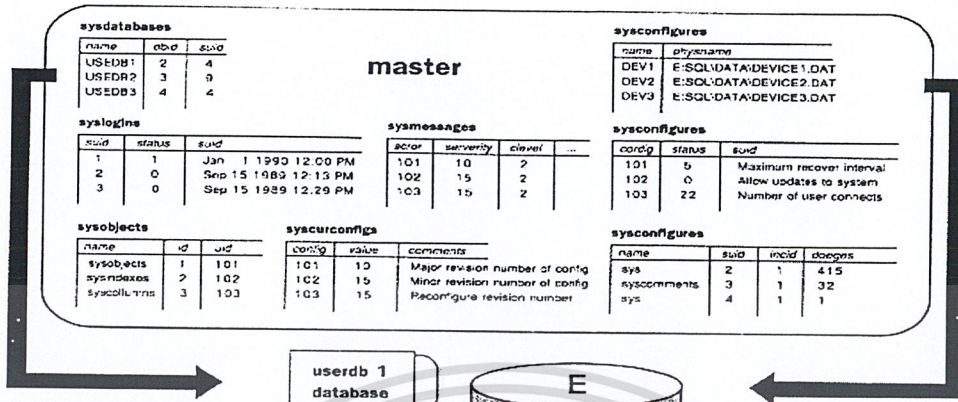
รูปที่ 3.1 แสดงโครงสร้างฐานข้อมูลของ SQL Server

จากรูปที่ 3.1 จะเห็นว่า มี devices ของฐานข้อมูลอยู่ 3 ตัว คือ MASTER, MSDBDATA และ MSDBLOG โดยที่ใน MASTER device จะมีฐานข้อมูล master, model, tempdb และ pubs อยู่ภายใน ส่วนฐานข้อมูล msdb จะอยู่ใน MSDBDATA device และ transaction log ของ msdb จะเก็บอยู่บน MSDBLOG

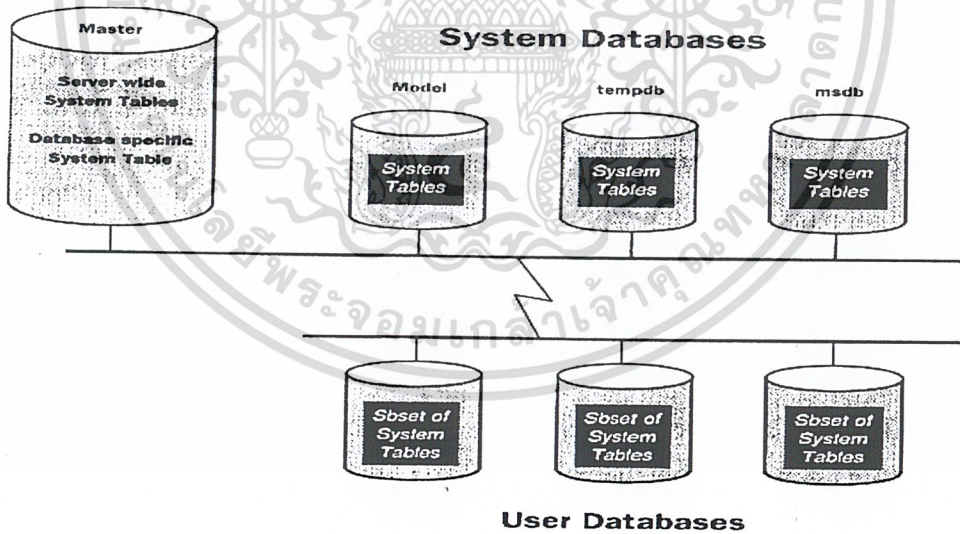
Master Database

master Database จะบรรจุตาราง (system Tables) ซึ่งเก็บ อินฟอร์มเมชัน เกี่ยวกับ SQL Server ทั้งหมด และข้อมูลเกี่ยวกับ User ตารางจะประกอบด้วยไดเรกทอรีข้อมูล หรือระบบแคตตาล็อกของ SQL Server เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า master Database จะควบคุมฐานข้อมูลของผู้ใช้งาน และปฏิบัติการกับ SQL Server นอกจากนี้ยังเก็บ User ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

accounts, system error message, active locks, database-backup device, system-stored procedures ฯลฯ (จะใช้พื้นที่ 25 เมกะไบต์ เป็นค่าโดยประมาณสำหรับ master database)



รูปที่ 3.2 แสดงโครงสร้างของ master Database



รูปที่ 3.3 แสดงลักษณะของ System Tables

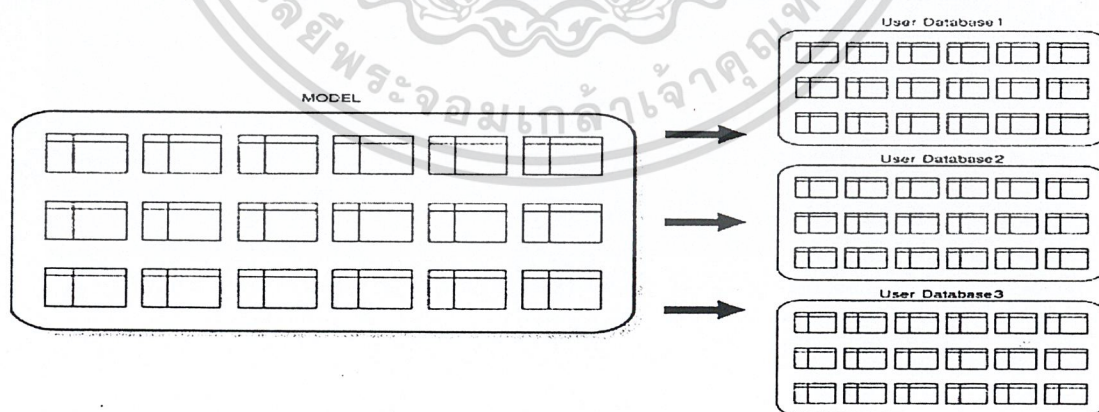
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

System Tables	หน้าที่
Syscharsets	เก็บ Character set หรือ sort order
Sysconfigures/syscurconfigs	คอนฟิกสภาพแวดล้อมที่เปลี่ยนไป
Sysdatabases	เก็บฐานข้อมูลที่มีอยู่บน SQL Server
Sysdevice	เก็บฐานข้อมูลของdeviceและdisk device
Sydlanguages	ภาษาในการติดตั้ง SQL
Syslocks	แ็็คทีฟล็อก (ไคนามิก)
Syslogins	เก็บ User Accounts-SQL Server login ID
Sysmessages	เก็บ system error message
Sysprocesses	เก็บโปรเซสปัจจุบัน (ไคนามิก)
Sysremotelogins	เก็บรีโมต Userf Account
Sysservers	เก็บรีโมตเซิร์ฟเวอร์และ โลกอลเซิร์ฟเวอร์
Sysusages	พื้นที่ว่างสำหรับฐานข้อมูล

ตารางที่ 3.1 แสดงตาราง system Tables ของ master Database

Model Database

model Database จะบรรจุตารางเก็บฐานข้อมูลของผู้ใช้งานทุกคน ทุกครั้งที่มีการสร้างฐานข้อมูล SQL Server จะใช้ model database เป็นเทมเพลตเราสามารถจะปรับเปลี่ยน เพื่อ customize โครงสร้างของฐานข้อมูลที่เราสร้างใหม่ และทุกสิ่งที่ได้ทำบน model จะมีผลกับฐานข้อมูลใหม่ รูปแบบในการเปลี่ยนแปลงบน model database จะรวมถึงการเซตสิทธิ์ในการเข้าถึงอย่างแน่นนอน การเปลี่ยนแปลงจะรวมถึงการเพิ่มผู้ใช้งาน กำหนดชนิดข้อมูล, กฎ(rule), defaults หรือ stored procedure และ database configuration options ก็จะสามารถจะเซตใน model database ได้เช่นกัน (ในพื้นที่ 1 เมกะไบต์ สำหรับ model database)



รูปที่ 3.4 แสดงโครงสร้างของ model Database

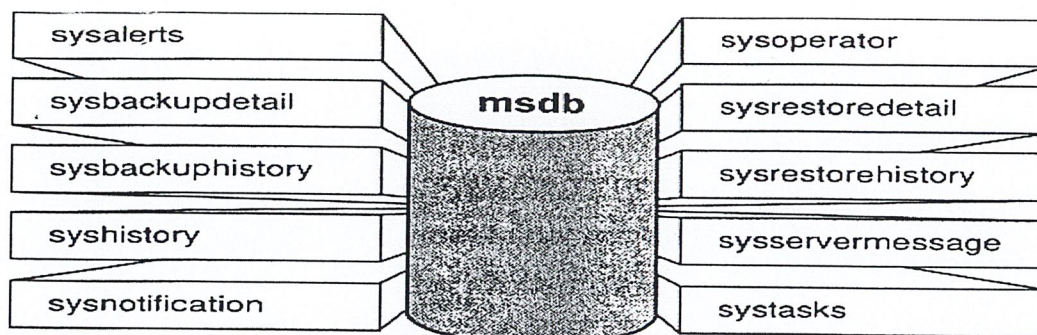
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

System Tables	หน้าที่
sysaltermates	จะมี 1 แถวสำหรับแต่ละ SQL Server User แม้จะไปพื้นฐานข้อมูล
sysarticles	บรรจุกุ artical อินฟอร์มเมชันสำหรับแต่ละ artical ที่สร้างขึ้นสำหรับการเรพพลิเคชัน
syscolumns	จะมี 1 แถวสำหรับแต่ละคอลัมในตารางหรือวิว และสำหรับแต่ละพารามิเตอร์ใน Stored Procedure
syscomments	จะมี 1 แถวหรือมากกว่า สำหรับแต่ละ View, rule, default, trigger และ Store Procedure กับ SQL เพื่อกำหนดคสเททเมนต์
sysconstraints	บรรจุกุ 1 แถวสำหรับแต่ละ PRIMARY KEY, FOREIGN KEY, CHECK,DEFAULT และ UNIQUE
Sysindexes	จะมี 1 แถวสำหรับแต่ละ clustered index, nonclustered index และตารางที่ไม่มี index เพิ่มแถวพิเศษสำหรับแต่ละตารางกับเท็กซ์ หรือข้อมูลแบบอิมเมจ
syskeys	จะมี 1 แถวสำหรับแต่ละ Foreign, Primary หรือ Common Key
syslogs	บรรจุกุ Transaction log
sysobjects	จะมี 1 แถวสำหรับแต่ละตาราง, View, Stored Procedure, rule, trigger,default, log และ temporary object (เฉพาะใน tempdb)
sysprocedures	มี 1 แถวสำหรับแต่ละ View,rule, default, trigger และ Stored Procedure
Sysprotects	บรรจุกุอินฟอร์มเมชันของ User Permissions
Syspublications	บรรจุกุ 1 แถวสำหรับแต่ละ publication ที่สร้างขึ้น
Sysreferences	บรรจุกุ 1 แถวสำหรับแต่ละ FOREIGN KEY ที่บังคับ
Syssegments	มี 1 แถวสำหรับแต่ละ Segment
Syssubscriptions	บรรจุกุ 1 แถวสำหรับแต่ละ Subscription จาก Subscribing server (เป็นการรวม ID ของแต่ publication ที่เกิดขึ้นกับ ID ของ subscription Server ทั้งหมดในการรับข้อมูล)
sysypes	มี 1 แถวสำหรับแต่ละคิฟอลท์ System-supplied และ User difined datatype
sysusers	มี 1 แถว สำหรับแต่ละ User และ Group ที่กำหนดในฐานข้อมูล

ตารางที่ 3.2 แสดงตาราง System Tables ของ model Database

MsdB Database

msdb database จะสนับสนุนบริการของ SQL Executive, Task scheduling และ Alerts management ฐานข้อมูลตัวนี้จะใช้ในการแก้ปัญหาเกี่ยวกับ scheduled tasks msdbdatabase จะถูกสร้างขึ้นอัตโนมัติโดยเอ็กส์แอรันเป็นเอ็กส์แอรันที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษานี้ เหมือนญาติให้เขาไปเซิร์ฟเวอร์เซชันด้านการค้า โปรแกรม Setup (จะใช้พื้นที่ 6 เมกะไบต์ สำหรับ msdb database และ 2 เมกะไบต์ สำหรับ Transaction Log) ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอ็กส์แอรันทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 แสดงโครงสร้างของ msdb

System Tables	หน้าที่
Sysalerts	เก็บอินฟอร์เมชันเกี่ยวกับ User ทุกคน-กำหนดการเตือน SQL Executive engine จะตรวจเช็คเพื่อดูว่ามีการเข้ามาในตารางนี้ ก็จะร่วมบันทึกเหตุการณ์ลงใน Event Viewer ของ Windows NT และตัดสินใจในการเตือน (alert)
Sysbackupdetail	รายการผลสรุปของ device ที่ใช้ในการแบ็คอัพ
sysbackuphistory	รายการผลสรุปของทุกการทำแบ็คอัพ
sysoperators	เก็บอินฟอร์เมชันเกี่ยวกับการจัดการ-ดำเนินการทั้งหมดเช่น work schedules, e-msil names และ Pager number
systaks	เก็บอินฟอร์เมชันเกี่ยวกับ User ทั้งหมดกำหนดงาน SQL Executive engine จะคอยตรวจดูว่ามีการเข้ามาในตารางนี้เพื่อกำหนดงาน
syshistory	เก็บอินฟอร์เมชันในอดีตที่เกี่ยวกับ alert และ Task ที่ได้ทำไปแล้ว (ทั้งสำเร็จและไม่สำเร็จ) เก็บหลักฐานในการส่ง e-mail, จำนวนครั้ง ในการพยายามสำหรับงาน (task) วัน เวลาในการทำ ฯลฯ
sysnotification	เก็บ alert ที่เกิดขึ้นและวิธีการที่ใช้ส่ง (e-mail หรือ pager)
sysrestoredetail	รายการผลสรุปของ device ที่ใช้ในการ โหลด(restore)
sysrestorehistory	รายการผลสรุปของการ โหลด (restore) ทุกครั้ง

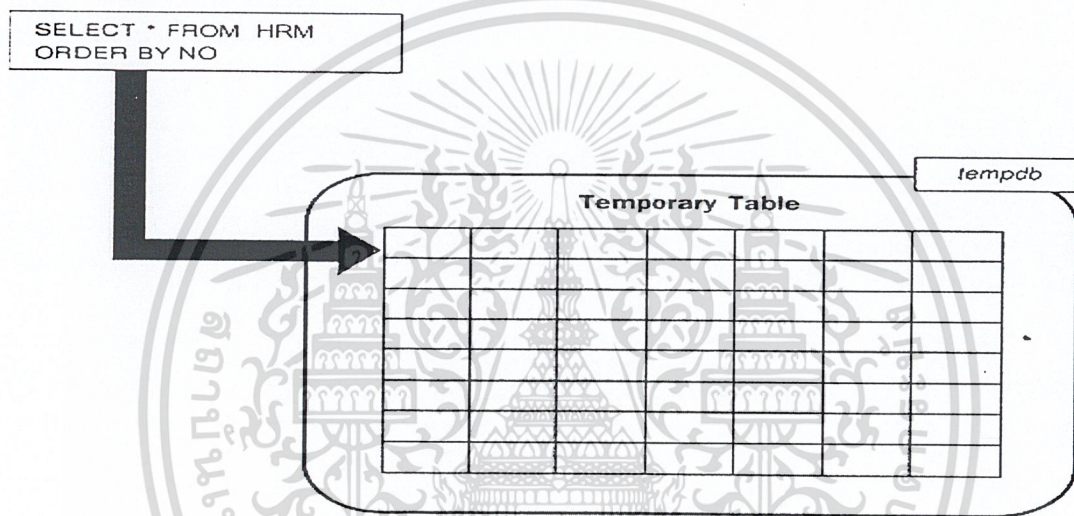
ตารางที่ 3.3 แสดงตาราง System Tables ของ msdb Database

tempdb Database

tempdb database จะจัดเตรียมที่เก็บสำหรับตารางชั่วคราว และงานชั่วคราวที่ต้องการที่เก็บ (storage) tempdb database จะทำการแชร์พื้นที่ว่างในการทำงานซึ่งจะใช้โดยฐานข้อมูลทั้งหมดบน SQL Server ตารางชั่วคราวทั้งหมดจะเก็บอยู่ใน tempdb โดยจะไม่คำนึงถึงฐานข้อมูลที่ User กำหนดให้อยู่ และเมื่อใดก็ตามที่ User ไม่ได้เชื่อมต่อกับทาง SQL Server แล้ว User's tempdb ที่อยู่ภายในจะถูกเกลียดด้วย และถ้าระบบเกิดไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หยุดทำงาน (stop) สิ่งที่อยู่ภายใน tempdb ทั้งหมดจะถูกเคลียร์ทันที tempdb database ยังใช้ในการจัดเรียงอินฟอร์เมชันได้ เช่น เมื่อ query จัดการ (executer) บนตารางที่ปราศจาก index ก็จะมีการสร้าง index ชั่วคราวขึ้นใน tempdb บางระบบ stored procedures จะใช้ tempdb Queries จะจัดเรียงหรือกรอง อินฟอร์เมชันก็จะใช้ tempdb ในการเก็บผลลัพธ์ระดับกลาง

tempdb database สามารถจะอยู่ในหน่วยความจำ RAM โดยการเซต SQL Server Configuration option : tempdb in RAM (MB) ซึ่งจะมีผลกระทบต่อครอบคลุมถึง Server performance เพราะ queries จะจัดเรียงโดยใช้ tempdb ในหน่วยความจำ RAM จัดการอย่างรวดเร็วกว่าบนฮาร์ดดิสก์ ตัวเลือกนี้ไม่ควรใช้เว้นแต่เซิร์ฟเวอร์จะมีหน่วยความจำ RAM อย่างน้อย 64 เมกะไบต์ (จะใช้พื้นที่สำหรับ tempdb database เพียง 2 เมกะไบต์ เป็นค่าดีฟอลท์ สามารถปรับเปลี่ยนได้โดย SA)



รูปที่ 3.6 แสดงโครงสร้างของ tempdb Database

3.1.1. System Stored Procedures

System Stored Procedure จะเป็นตัวเก็บ SQL สเตทเมนต์ ซึ่งจะช่วยให้เราสามารถเรียกดูข้อมูลจาก System table ได้ง่ายขึ้น ปกติแล้ว System Stored Procedure จะอยู่ในฐานข้อมูล master ผู้ที่เป็นผู้บริหารระบบ สามารถจะเรียกดูได้ System Stored Procedure จะขึ้นด้วยตัวอักษร Sp_ การทำงานเพียงแค่พิมพ์ชื่อลงไปที่สามารถจะเรียกดูอินฟอร์เมชันต่างๆ ใน System Stored Procedure ได้ตั้งขั้นตอนต่อไปนี้

* เปิดหน้าต่างของ query (ISQL_W) ที่ครอบห่อง DB ให้เลือก pubs database แล้วพิมพ์คำสั่ง `Sp_tables` เสร็จแล้วกดคีย์ `Ctrl+E` เพื่อจัดการกับคำสั่งนี้ จะได้ผลลัพธ์ดังรูปที่ 3.7 ก.

* พิมพ์คำสั่ง `sp_helpdb` จะแสดงขนาด (db_size) ของฐานข้อมูลทั้ง 5 แบบ พร้อมทั้ง รายละเอียดอื่นๆ ดังรูปที่ 3.7 ข.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

* พิมพ์คำสั่ง `sp_spaceused` แต่ให้ใช้ `master` database จะแสดงรายละเอียดของฐาน ข้อมูล `master` ดังรูปที่ 3.7 ก.

* ต่อไปเราจะเรียกดูรายละเอียดชนิดของข้อมูลใน ระบบฐานข้อมูล SQL Server โดยใช้ Stored Procedure เกี่ยวกับชนิดของข้อมูล (ให้ใช้ฐานข้อมูล `pubs`) ให้พิมพ์คำสั่ง `sp_datatype_info` จะแสดง รายละเอียดชนิดของข้อมูลดังรูปที่ 3.8

TABLE_QUALIFIER	TABLE_OWNER	TABLE_NAME
pubs	dbo	sysalternates
pubs	dbo	sysarticles
pubs	dbo	syscolumns
pubs	dbo	syscomments
pubs	dbo	sysconstraints
pubs	dbo	sysdepends
pubs	dbo	sysindexes
pubs	dbo	syskeys
pubs	dbo	syslogs
pubs	dbo	sysobjects
pubs	dbo	sysprocedures
pubs	dbo	sysprotects
pubs	dbo	syspublications
pubs	dbo	sysreferences

ก.

name	db_size	owner	dbid	created	status
master	17.00 MB	sa	1	Apr 3 1995	trunc log on chkpt
model	1.00 MB	sa	3	Apr 3 1995	no options set
msdb	8.00 MB	sa	5	Feb 4 1998	trunc log on chkpt
pubs	3.00 MB	sa	4	Apr 3 1995	trunc log on chkpt
tempdb	2.00 MB	sa	2	Feb 5 1998	select into/bulkcopy

ข.

database_name	database_size	unallocated space
master	17.00 MB	3.48 MB

reserved	data	index_size	unused
13844 KB	12254 KB	435 KB	1144 KB

ค.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในภาคการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.7 แสดงการใช้คำสั่ง `Sp_ (Stored Procedures)`
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPE_NAME	DATA_TYPE	PRECISION	LITERAL_PREFIX	LITERAL_SUFFIX
bit	-7	1	{null}	{null}
tinyint	-6	3	{null}	{null}
tinyint identity	-6	3	{null}	{null}
image	-4	2147483647	0x	{null}
varbinary	-3	255	0x	{null}
binary	-2	255	0x	{null}
timestamp	-2	8	0x	{null}
text	-1	2147483647	.	{null}
char	1	255	.	.
varchar	1	9	.	.
numeric	2	28	{null}	{null}
numeric() identity	2	28	{null}	{null}
decimal	3	28	{null}	{null}
money	3	19	\$	{null}

รูปที่ 3.8 แสดงการใช้คำสั่ง Sp_datatype_info

3.1.2. การปรับเปลี่ยนค่าของ Configuration

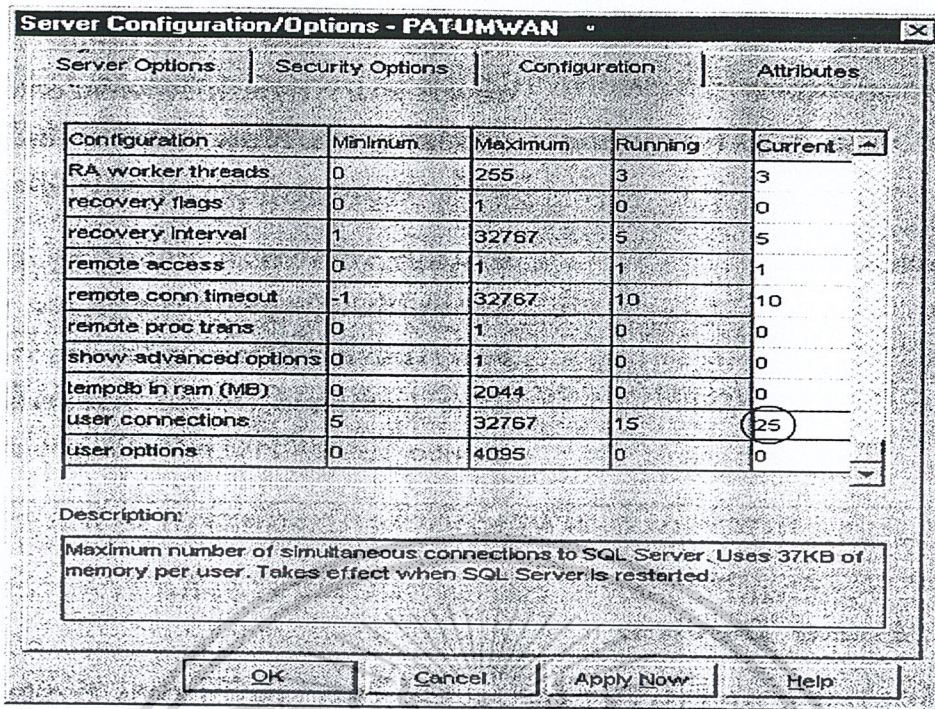
ในการเชื่อมต่อเข้ามาของผู้ใช้แต่ละคนใช้พื้นที่ในหน่วยความจำประมาณ 40 กิโลไบต์ ผู้บริหารระบบ (System administrator) จะต้องพิจารณาว่า หน่วยความจำของเซิร์ฟเวอร์ฐานข้อมูลเพียงพอหรือไม่ ผู้บริหารระบบสามารถจะปรับเปลี่ยนจำนวนการเชื่อมต่อของผู้ใช้ให้เหมาะสมค่าดีฟอลท์ในการเชื่อมต่อของผู้ใช้คือ 15 เราสามารถจะปรับเปลี่ยนค่า Configuration ได้ตามขั้นตอนต่อไปนี้

* ให้เปิด SQL Enterprise Manager แล้วเลือกเซิร์ฟเวอร์ SQL ที่เมนูคำสั่ง Server เลือกคำสั่ง SQL Server/Configure(หรือเมาส์ไปที่เซิร์ฟเวอร์ SQL คลิกเมาส์ปุ่มของจะแสดงเมนูคำสั่งให้เลือก Configure)

* จะปรากฏหน้าต่างของ Server Configuration/Options ให้คลิกที่แถบ Configuration เดือนสกอลบาร์ ลงมาล่างสุดจะพบ User Connections ที่ช่อง Current ให้เปลี่ยนตัวเลขจาก 15 เป็น 25 ดังรูปที่ 3.9 แล้วคลิกปุ่ม OK จะกลับมาที่หน้าต่างของ SQL Enterprise Manager

* ให้ทำการ Stop และ Start เครื่องเซิร์ฟเวอร์ SQL เพื่อทดสอบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แสดงการปรับเปลี่ยนค่า Configuration

3.1.3. การจัดการฐานข้อมูล

ผู้จัดการระบบ จะมีหน้าที่ในการจัดการกับ Database และ Device รวมทั้งการจัดสรรพื้นที่ว่าง สำหรับ modifying creating expanding dropping mirroring และ Viewing information เกี่ยวกับ Database และ Device SQL Server จะเก็บข้อมูลบน device device จะจัดแบ่งระบบปฏิบัติการในการใช้เก็บ database, transaction logs และ backup device จะมีอยู่ 2 แบบด้วยกันคือ

- * Database Device
- * Backup Device

Database Device คือ disk file ซึ่งจะเก็บข้อมูลและ transaction log ใน 1 device สามารถจะเก็บได้ หลาย Database และ database ยังสามารถเก็บไว้บน device ตัวอื่นได้เช่นกัน (ฮาร์ดดิสก์ตัวอื่นบน device) ผู้บริหารระบบเท่านั้นมีสิทธิ์ในการสร้าง device

Backup Device เป็นอุปกรณ์จำพวก ฮาร์ดดิสก์ เทปแบ็คอัพ ดัมคิสต์จะคอยเก็บแบ็คอัพของข้อมูล และ transaction log เราสามารถจะสร้าง Backup Device ในเวลาเดียวกันกับการสร้าง Database Device หรือจะเลื่อนการสร้าง Backup Device ไปจนกว่าจะพร้อมในการแบ็คอัพฐานข้อมูลก็ได้

3.1.4. การสร้าง Database Device

ผู้บริหารระบบจะต้องประมาณจำนวนของ device ในการสนับสนุน System database และ User Database ซึ่ง System database ประกอบไปด้วย master, model, msdb และ tempdb ส่วน User database จะประกอบไปด้วย pubs ขนาดต่ำสุดของ Database Device จะต้องเป็น 1 เมกะไบต์ เราสามารถจะทำให้ device

มีขนาดใหญ่กว่าเดิมเพื่อความเหมาะสมในงานแต่ไม่สามารถจะทำให้ขนาดเล็กลงได้ ในกรณีที่เกิดปัญหาพื้นที่ขนาดของ device ไม่พอสามารถจะขยายขนาดได้ในภายหลัง Database device สามารถจะสร้างบนฮาร์ดดิสก์ที่อยู่บนเครื่องเซิร์ฟเวอร์ เพราะว่า SQL Server ไม่สนับสนุนให้ใช้งาน Database device ผ่านระบบเน็ตเวิร์ก

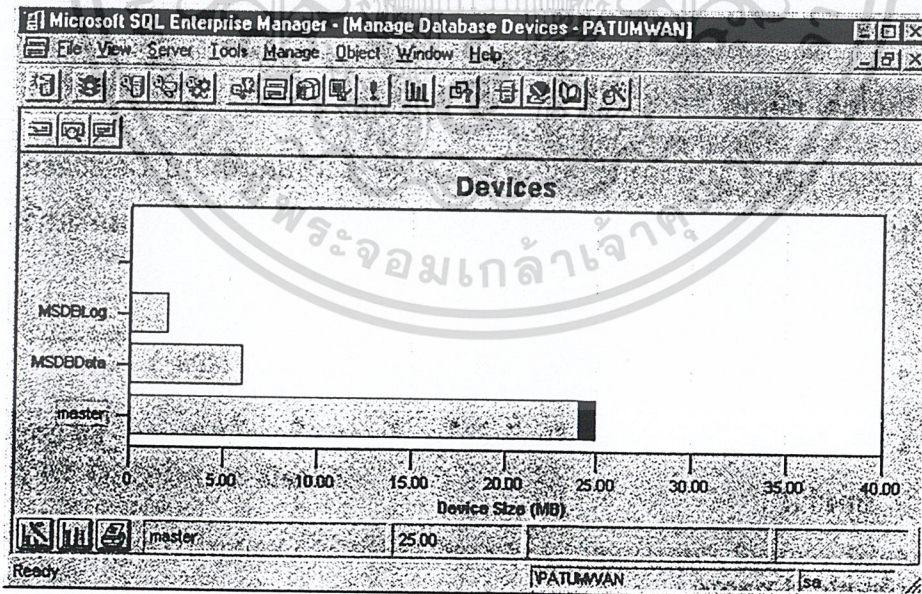
ขั้นตอนในการสร้าง Device มีอยู่ 2 วิธีด้วยกัน คือ

1. การใช้ SQL Enterprise Manager
2. การใช้สแตทเมนต์ DISK INIT

3.1.4.1. การใช้ SQL Enterprise Manager

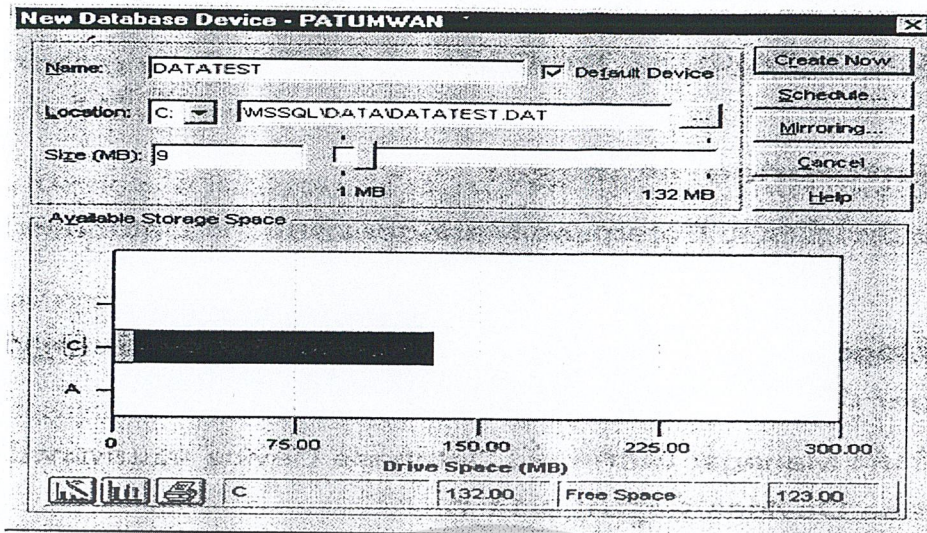
เราสามารถจะใช้ SQL Enterprise Manager เพื่อสร้าง Database Device โดยทำตามขั้นตอนต่อไปนี้

- * ทำการ logon เข้าสู่เซิร์ฟเวอร์ด้วย Administrator แล้วเริ่มสตาร์ท SQL Enterprise Manager เลือกเซิร์ฟเวอร์ SQL
- * ที่เมนูคำสั่ง Manager เลือกคำสั่ง Database Device จะแสดงหน้าต่างของ Manager Database Devices ดังรูปที่ 3.10 ก.
- * ที่เมนูคำสั่ง File เลือกคำสั่ง New Device หรือจะคลิกเมาส์ที่ทูลบาร์ (New Device) ก็ได้ จะปรากฏหน้าต่างของ New Database Device
- * ให้พิมพ์ชื่อของ Database ลงไปในช่อง Name คือ DATATEST แล้วคลิกเมาส์ที่เช็ทบ็อก Default Device ส่วนของ Location เลือกไดรฟ์ C จะเห็นว่าในกรอบจะเป็นชื่อ physical ซึ่งอ้างอิงถึง logical ดังรูปที่ 3.10 ข.



ก.

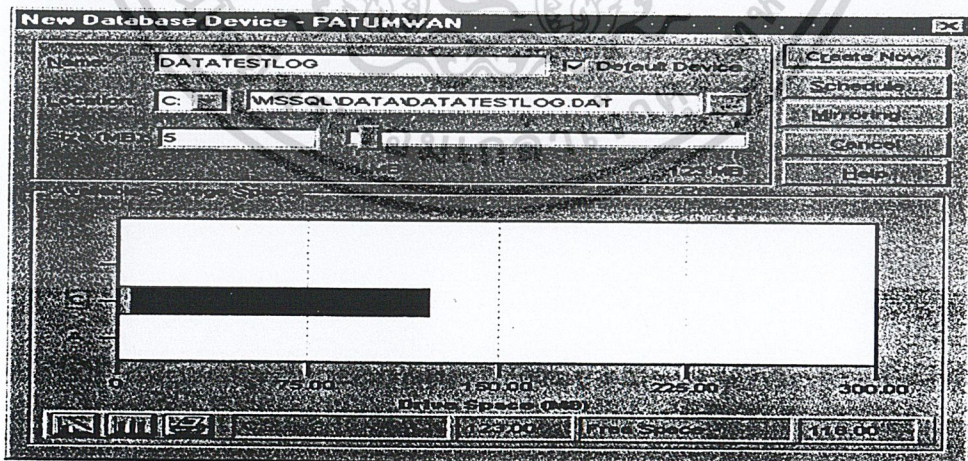
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ข.

รูปที่ 3.10 แสดง Manager Database Devices

- * ที่ช่อง Size (MB) ให้ใส่ขนาดของ Database Device ลงไปเช่น 9 จะสังเกตเห็นว่าเราไม่สามารถจะใส่ขนาดที่มีมากกว่าพื้นที่ว่างของฮาร์ดดิสก์ได้เสร็จแล้วคลิกปุ่ม Create Now
- * จะกลับมาที่หน้าจอของ Manager Database Devices ก็จะมี Device ตัวใหม่คือ DATATEST ปรากฏอยู่ด้วย
- * ต่อไปจะสร้าง Device สำหรับ Transaction log คลิกที่ทูลบาร์ New Device จะเข้าสู่การสร้าง Device
- * ที่ Location เลือกไดรฟ์ C จะเห็นว่าข้อความของพาทจะอ้างถึง logical ดังรูปที่ 3.11 เพิ่มขนาดเป็น 5 MB แล้วคลิกที่ปุ่ม Create Now



รูปที่ 3.11 แสดงการสร้าง Device ให้ Transaction log

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

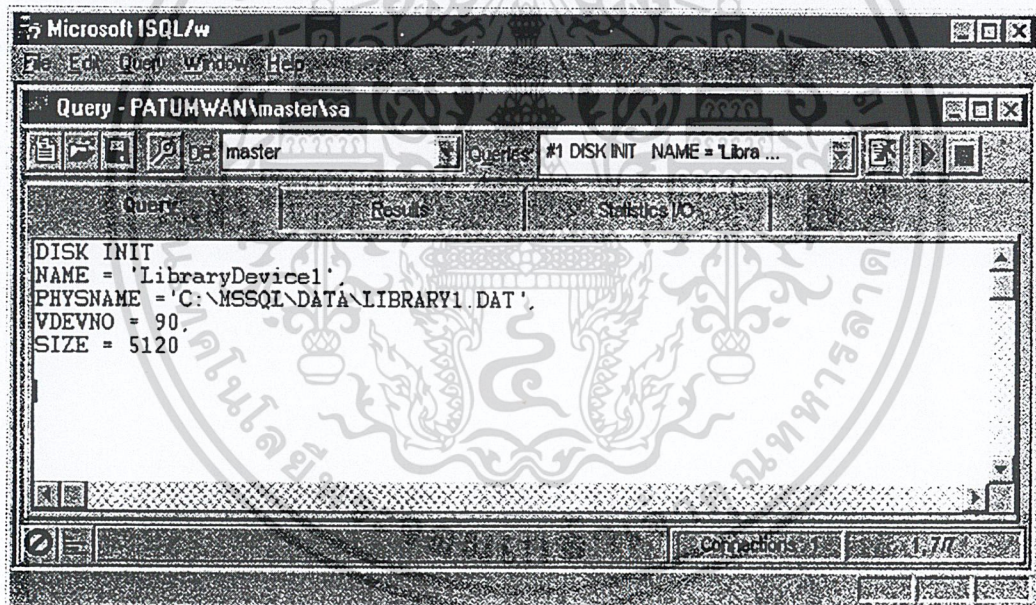
3.1.4.2. การใช้สแตทเมนต์ DISK INIT

วิธีสองในการสร้าง Database Device คือการใช้สแตทเมนต์ DISK INIT สามารถทำได้ดังนี้

* เข้าสู่ ISQL/w เพื่อเปิด Query แล้วใช้ฐานข้อมูล master เลือกแถบ Query แล้วพิมพ์ ข้อความดังนี้

```
DISK INIT
NAME='LibraryDevice1',
PHYSNAME='C:\MSSQL\DATA\LIBRARY1.DAT',
VDENO = 90,
SIZE = 5120
```

เสร็จแล้วให้กดคีย์ Ctrl+E เพื่อรันถ้าการสร้างสำเร็จด้วยคีย์จะมีความ “This Command did not return data, and it did not return any rows” (ข้อระวังในการพิมพ์คำสั่งขอให้มีความละเอียดเรื่องเครื่องหมายต่าง ๆ ด้วย)



รูปที่ 3.12 แสดงการใช้สแตทเมนต์ DISK INIT ในการสร้าง Device

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

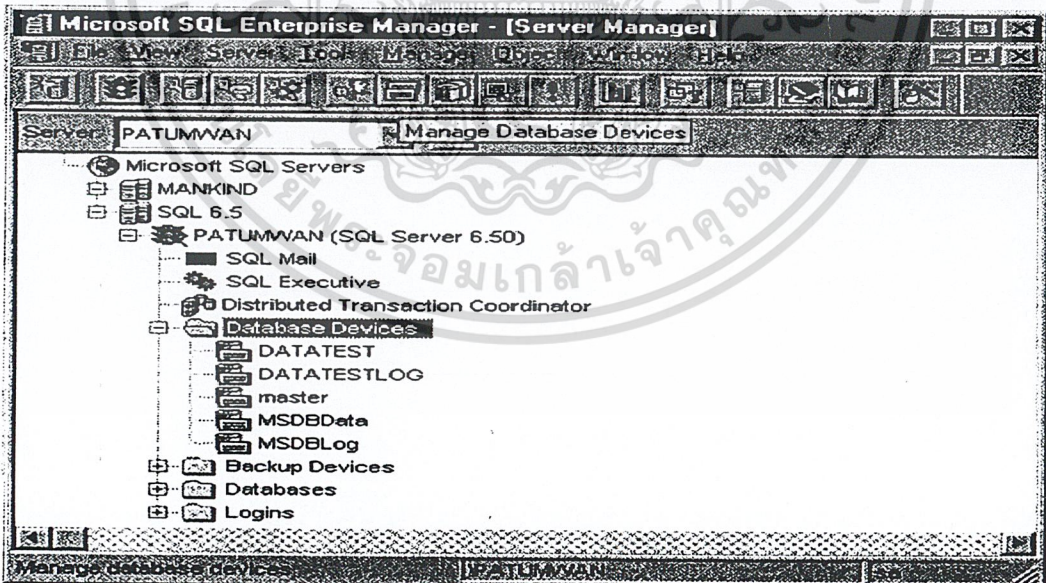
device_name	physical_name	description
DATATEST	C:\MSSQL\DATA\DATATEST.DAT	special, default disk, physical disk, 9 MB
DATATESTLOG	C:\MSSQL\DATA\DATATESTLOG.DAT	special, default disk, physical disk, 5 MB
diskdump	nul	disk, dump device
diskettedumpa	a: sqltable.dat	diskette, 1.2 MB, dump device
diskettedumpb	b: sqltable.dat	diskette, 1.2 MB, dump device
LibraryDevice1	C:\MSSQL\DATA\LIBRARY1.DAT	special, physical disk, 10 MB
master	C:\MSSQL\DATA\MASTER.DAT	special, default disk, physical disk, 25 MB
MSDBData	C:\MSSQL\DATA\MSDB.DAT	special, physical disk, 6 MB
MSDBLog	C:\MSSQL\DATA\MSDBLOG.DAT	special, physical disk, 2 MB

(1 row(s) affected)

รูปที่ 3.13 แสดงการใช้ Sp_helpdevice

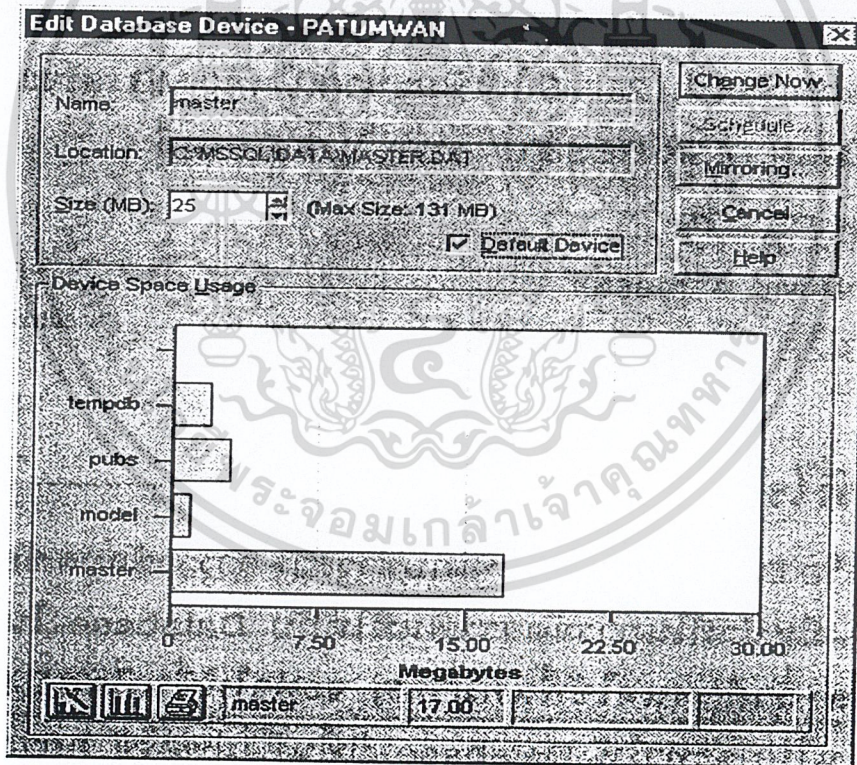
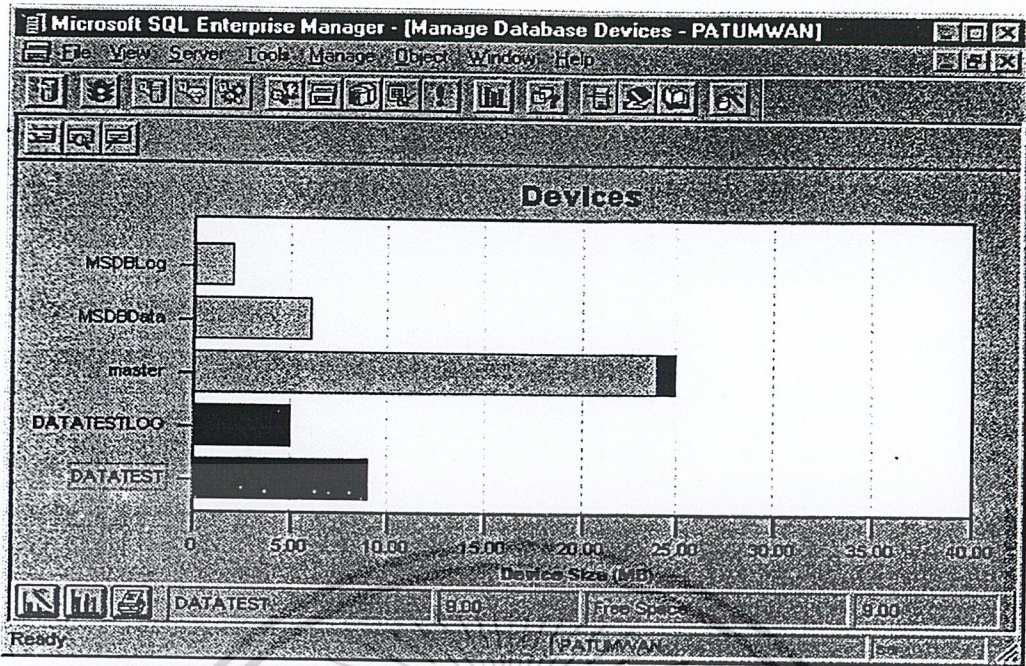
เราสามารถจะใช้ stored procedure ในการตรวจดู Device ต่าง ๆ ที่มีอยู่ ด้วยคำสั่ง sp_helpdevice ดังรูปที่ 3.13 นอกจากนี้ยังใช้ SQL Enterprise Manager ในการตรวจดู Device ต่าง ๆ ได้ตามขั้นตอนต่อไปนี้

1. เข้าสู่ SQL Enterprise Manager เลือกเซิร์ฟเวอร์ SQL ดังรูปที่ 3.14 ก. ตรงโฟลเดอร์ของ Database Devices จะมีเพิ่มขึ้นมา 2 Device คือ DATATEST และ DATATESTLOG
2. ให้คลิกที่ทูลบาร์ Manager Database Device ตัวที่ 7 หรือเมนูคำสั่ง Manager \ Database Device จะแสดงหน้าต่างของ Manager Database Devices ดังรูปที่ 2.14 ข. จะเห็นว่ามีการแสดงเป็นกราฟแท่งแฉวนอน
3. ขั้นตอนถัดไปถ้าต้องการทราบรายละเอียดของ Device ก็ให้ดับเบิลคลิกเมาส์ที่รูปกราฟของ Device นั้นก็จะแสดงรายละเอียดของชื่อทั้ง logical, physical ขนาดความจุ ดังรูปที่ 3.14 ค,



ก.

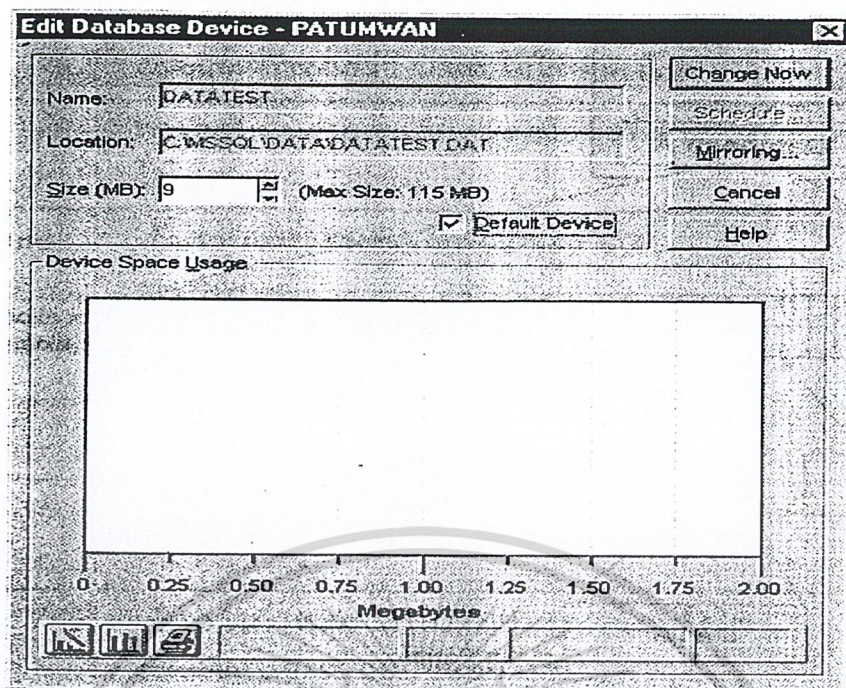
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.14 แสดงรายละเอียดของ Database Device
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก.

รูปที่ 3.14(ต่อ) แสดงรายละเอียดของ Database Device

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14(ต่อ) แสดงรายละเอียดของ Database Device

เราสามารถจะเปลี่ยนขนาดของ Device ได้ตามต้องการเช่น Master Device ดังรูปที่ 2.15 ค. จะเห็นว่าขนาดเดิมเป็น 25 MB ให้เพิ่มขนาดเป็น 40 MB เสร็จแล้วคลิกที่ปุ่ม Change Now หรือจะใช้สเตทเมนต์ DISK RESIZE โดยจะมีรูปแบบดังนี้

```
DISK RESIZE
NAME = 'master'
SIZE = 20480
```

3.1.5. การตั้งชื่อ Database Device

เมื่อได้สร้าง Database Device จะต้องมีการกำหนด physical name และ logical name และเมื่อมีการใช้ SQL Enterprise Manager หรือ Transaction SQL statement ก็จะต้องอ้างถึง Database Device โดยใช้ logical name

3.1.5.1. Logical name

logical name เป็นชื่อที่ใช้โดย SQL Server ซึ่งเหมือนกันกับ Database Device และเป็นไปตามกฎของ SQL Server สามารถจะมีตัวอักษรได้ถึง 30 ตัว รวมทั้งเครื่องหมายและตัวเลข โดยจะมีกฎของการตั้งชื่อดังนี้

- * ตัวอักษรตัวแรกจะต้องเป็นตัวอักษรหรือเครื่องหมาย (@#_)
- * ตัวอักษรตัวที่สองอาจจะเป็นตัวอักษร, ตัวเลข หรือเครื่องหมาย_#\$ และห้ามเว้น

ช่องว่าง

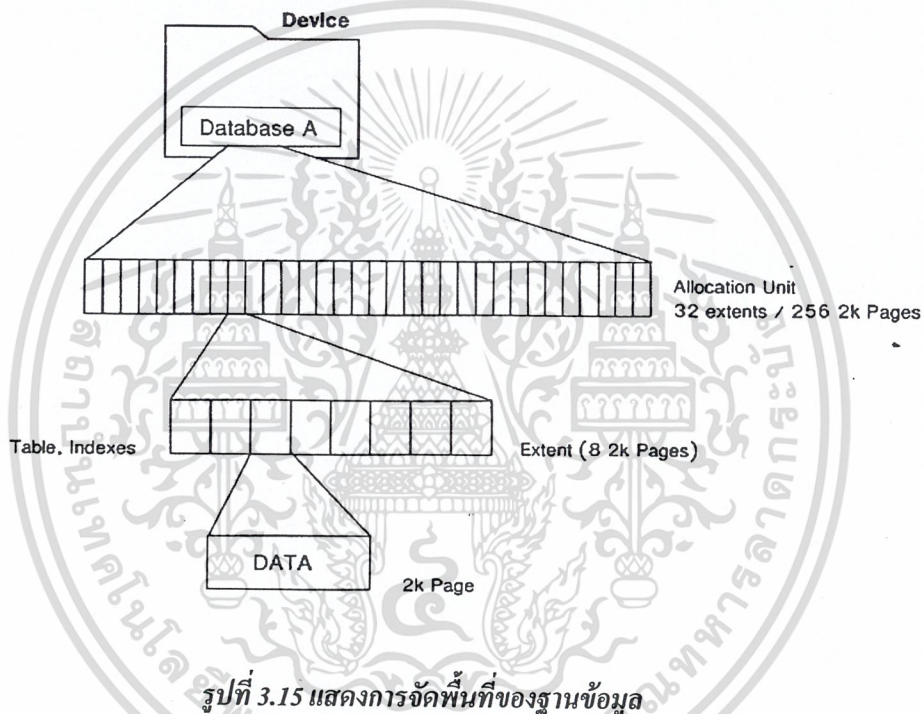
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5.2. Physical name

physical name จะเป็นชื่อตามกฎของระบบปฏิบัติการ ตลอดทั้งชื่อไฟล์ พาธใดเรกทอรี โดยจะใช้ชื่อให้ใกล้เคียงกันเช่นถ้า logical name มีชื่อ CUSTOM_LIST physical name จะเป็นชื่อ C:\MSSQL\DATA\CUSTOM.DAT

3.1.6. การจัดการพื้นที่ให้ฐานข้อมูล

ฐานข้อมูล หรือ Database จะเป็นที่เก็บข่าวสาร ตารางข้อมูล ข้อมูลต่าง ๆ ไว้มากมายผู้บริหารระบบมีหน้าที่ในการจัดสรรพื้นที่ให้กับ Database ในการออกแบบควรจะให้เสียพื้นที่น้อยที่สุด ถ้ากำหนดใช้พื้นที่มาก ๆ จะทำให้พื้นที่ของ Device ไม่เพียงพอแต่ถ้ากำหนดน้อยเกินไปก็จะทำให้ Database มีพื้นที่ไม่พอใช้ อย่างไรก็ตามเราสามารถจะขยายพื้นที่(Expand) หรือลดขนาด (Shrink) ของ Database ได้ตามความเหมาะสม



รูปที่ 3.15 แสดงการจัดการพื้นที่ของฐานข้อมูล

page	มีขนาด 2KB (2048 ไบต์)
Extents	มีจำนวน 8 pages ติดต่อกัน ขนาด 16 KB SQL Server จะเก็บ Table หรือ Index ที่สร้างขึ้นไว้ใน Extent ทุกอ็อบเจกต์จะมี Extent เป็นของตัวเองเพราะไม่มีการแชร์ให้ใช้งาน
Allocation Units	มีจำนวน 32 Extents หรือ 256 2K Page เมื่อมีการสร้าง Database จะมีการจัดสรรพื้นที่เพิ่มขึ้น 500 ไบต์ (256 2K Page ติดต่อกัน) (แต่ละ page จะถูกลบออกไป 32 ไบต์ สำหรับ overhead)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

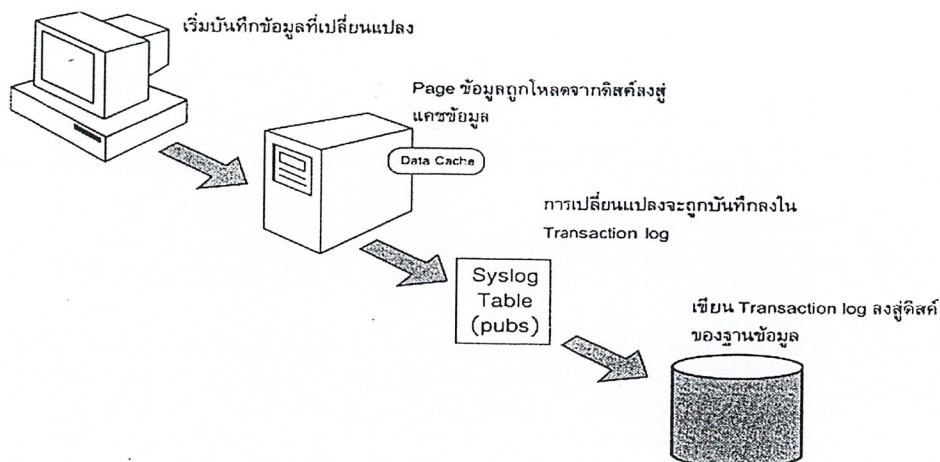
อ็อบเจ็กต์	หน้าที่	System Table
Table	เก็บรวบรวม rows และ columns	Sysobjects
View	สำหรับคู่มือข้อมูลในตารางอื่น ๆ	sysobjects
Default	เป็นค่า insert ของ SQL Server ใน Column ถ้าหากว่า User ไม่ได้ค่าลงไป	Sysobjects
Rule	การควบคุมรายละเอียดของข้อมูลที่สามารถใส่ในตาราง (Table)	Sysobjects
Store procedure	เก็บรวบรวม Precompiled SQL สเตทเมนต์	Sysobjects
Trigger	แบบฟอร์มพิเศษของ store procedure ซึ่งมี การเอ็กซีคิวต์แบบอัตโนมัติ เมื่อผู้ใช้ทำการเปลี่ยนแปลงข้อมูลในตาราง	Sysobjects
Index	จัดเรียงค่าของ rows ในตารางเพื่อช่วยให้ ค้นหาได้เร็ว	Sysindexes
Datatype	เป็นการกำหนดว่าจะเก็บข่าวสารแบบใดใน Column และเก็บข้อมูลอย่างไร	Systypes
Constraint	เป็นส่วนประกอบที่จะช่วยบำรุงรักษาข้อมูล อย่างถูกต้อง ซึ่งอยู่บนตารางหรือ column	Sysconstraints Sysreferences

ตารางที่ 3.4 แสดง SQL Server อ็อบเจ็กต์

3.1.7. Transaction Log

ในการสร้างระบบฐานข้อมูลก็จะมีการสร้าง transaction log ขึ้นมาด้วย โดยจะมีการจองพื้นที่เป็นตารางระบบ (syslogs) ทุกฐานข้อมูลจะมี transaction log เป็นของตัวเอง transaction log สามารถจะเก็บบน Device เดียวกับฐานข้อมูลก็ได้ หรือต่าง Device ก็ได้ แต่ควรจะมี transaction log ไว้ต่าง Device กับฐานข้อมูล เพื่อเวลาแบ็คอัพจะไม่มีผลกระทบต่อฐานข้อมูล และไม่สิ้นเปลืองพื้นที่ เพราะจะทำการแบ็คอัพแยกจากฐานข้อมูล ถ้าไม่ได้กำหนดให้ transaction log เก็บต่าง Device แล้วก็จะเก็บอยู่บน Device เดียวกับฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

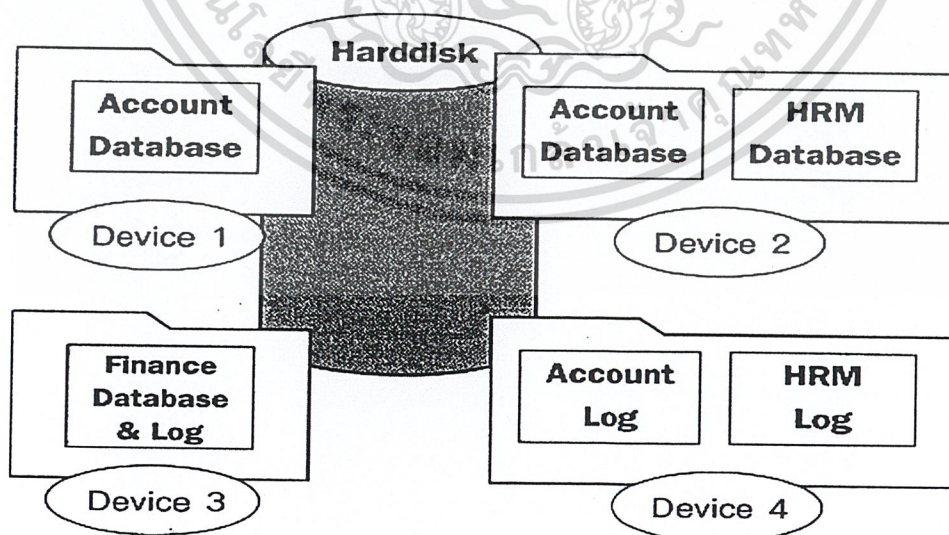


รูปที่ 3.16 แสดงการทำงานของ Transaction Log

Transaction log จะเก็บบันทึกข้อมูลที่มีการเปลี่ยนแปลง (insert, update, delete statements) จากรูปที่ 3.16 จะเห็นว่า เมื่อเริ่มบันทึกข้อมูลที่เปลี่ยนแปลง มีผลให้ Page ข้อมูลถูกโหลดจากดิสก์สู่แคชข้อมูล (ทุกการเปลี่ยนแปลงข้อมูลที่เกิดขึ้นจะถูกบันทึกลงใน log (syslogs) การเปลี่ยนแปลงจะถูกบันทึกลงใน log ก่อนที่การเปลี่ยนแปลงจะเกิดขึ้นบนฐานข้อมูล เรียก log แบบนี้ว่า write-ahead log จากนั้น page ข้อมูลที่อยู่ในหน่วยความจำแคชและ log page จะถูกบันทึกลงบนดิสก์ในตาราง syslogs สุดท้ายจึงทำการเขียน Transaction log ที่สมบูรณ์ ลงสู่ดิสก์ของฐานข้อมูล

(transaction log ของฐานข้อมูล จะบรรจุอยู่ในตารางระบบ syslogs อย่างไม่สามารถปรับเปลี่ยนแก้ไขได้ขนาด เพราะถูกใช้งานโดย SQL Server เท่านั้น)

3.18. รูปแบบของ Device และ Database



รูปที่ 3.17 แสดง Device และ Database แต่ละแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตามปกติแล้ว Database Device สามารถจะเก็บฐานข้อมูลและ transaction log แล้วใน 1 ฐานข้อมูลยังเก็บต่าง Device ได้ด้วย หรือใน 1 Device จะเก็บหลายฐานข้อมูลก็ได้จากรูปที่ 3.17 จะเห็นว่ารูปแบบการเก็บฐานข้อมูลหลายแบบด้วยกัน ดังนี้

* ฐานข้อมูล และ transaction log บน 1 Device ฐานข้อมูลของ Finance และ transaction log อยู่บน Device 3

* 1 ฐานข้อมูลกระจายอยู่บนหลาย Device ฐานข้อมูลของ Account กระจายอยู่บน Device 1, 2 และ 4 ส่วน transaction log จะอยู่บน Device 4 สำหรับฐานข้อมูล HRM จะอยู่บน Device 2, 4 และ log อยู่บน Device 4

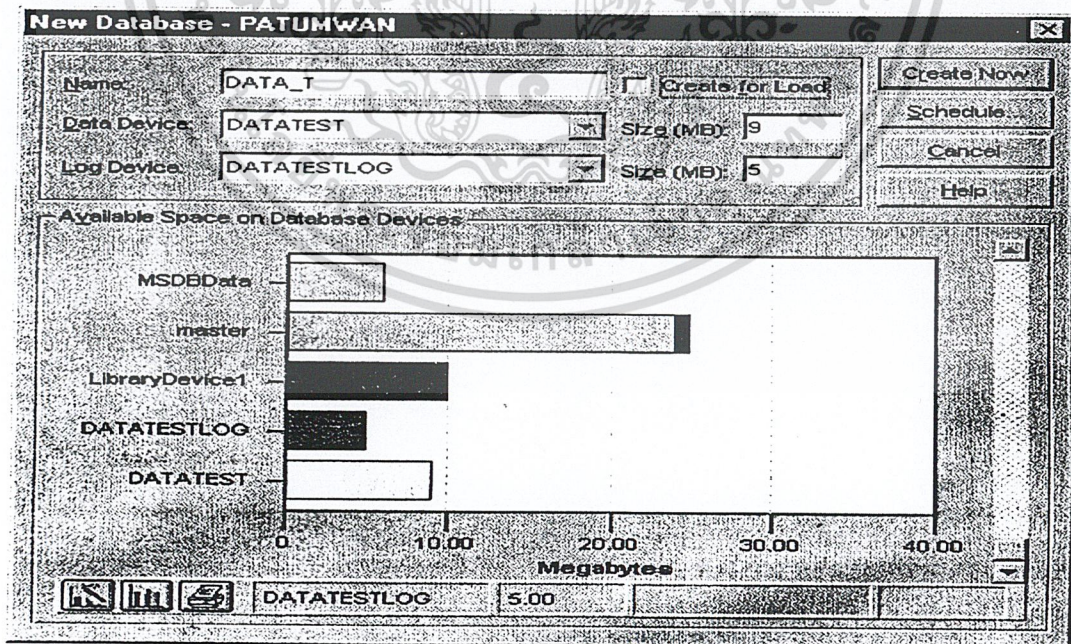
* หลายฐานข้อมูลอยู่บน 1 Device ฐานข้อมูล Account และ HRM จะอยู่บน Device 2

3.1.8.1. การสร้างฐานข้อมูลโดยใช้ SQL Enterprise Manager

ผู้บริหารระบบ สามารถจะสร้างฐานข้อมูลขึ้นมาใหม่เพื่อใช้งาน ตามความต้องการได้จะสร้างโดยใช้ SQL Enterprise Manager หรือคำสั่ง SQL สตatement ก็ได้ขอให้ทำการสร้างฐานข้อมูลตามขั้นตอนต่อไปนี้ เพราะจะใช้การทำงานของบต่อไป

* เข้าสู่ SQL Enterprise Manager ให้เลือกเซิร์ฟเวอร์ SQL แล้วคลิกเมาส์ที่ทูลบาร์ตัวที่ 8 หรือเมนูคำสั่ง Manager/Database จะปรากฏหน้าต่างของ Manager Database

* ให้คลิกที่ทูลบาร์ New Database จะปรากฏหน้าต่างของ New Database ที่ช่อง Name พิมพ์ชื่อฐานข้อมูลลงไปคือ DATA_T แล้วตรงช่อง Data Device ให้เลือก DATATEST (เป็น Database Device ที่ได้สร้างไว้แล้ว)

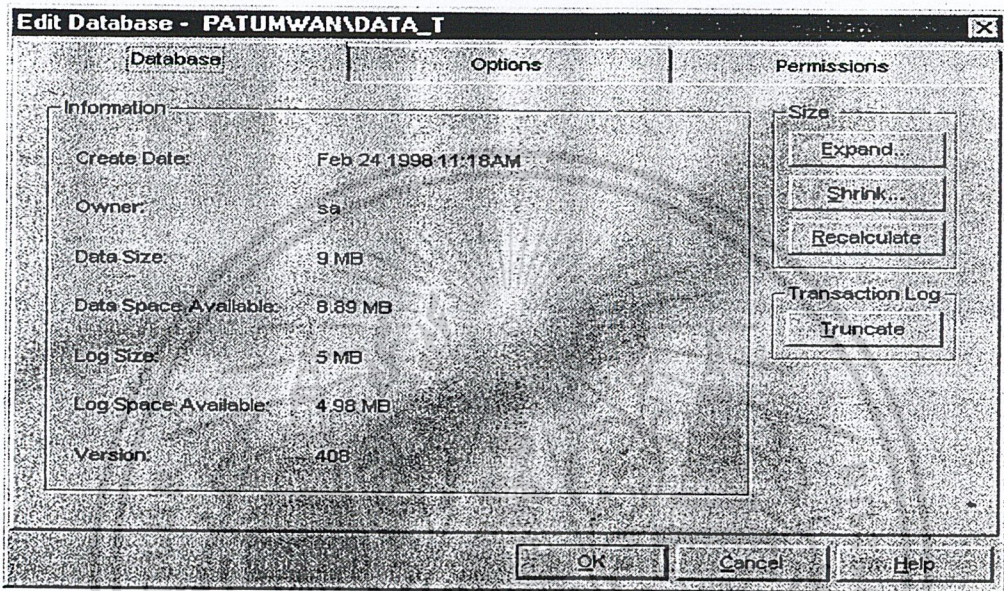


เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 3.18 แสดงการสร้างฐานข้อมูลชื่อ DATA_T ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

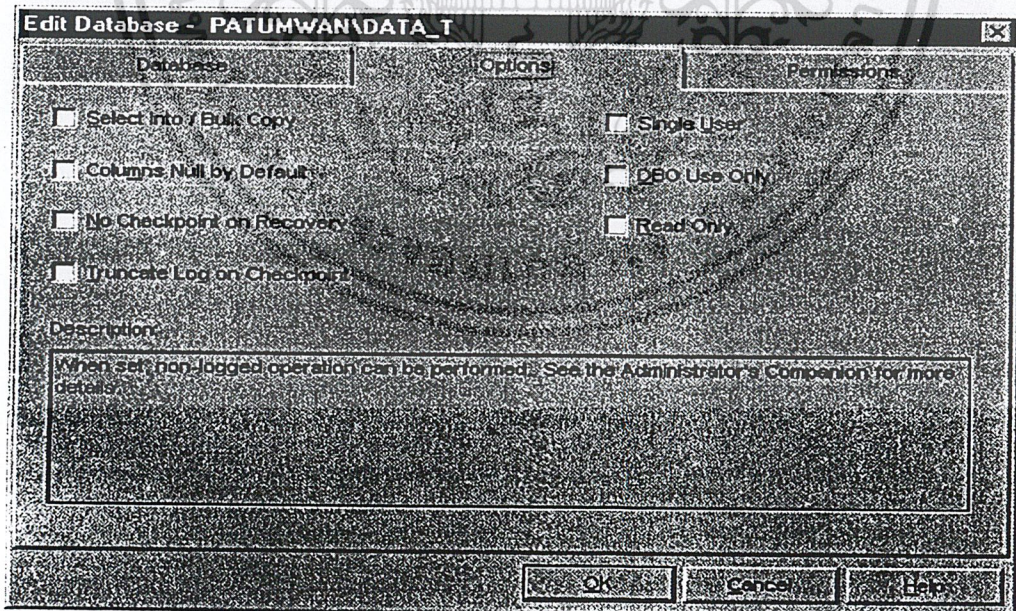
* จากรูปที่ 3.18 จะเห็นว่าที่กราฟแท่งแวนอนของ DATAEST เป็นสี่เหลี่ยม พร้อมทั้งแสดงขนาดในช่อง size (MB) คือ 9 เมกะไบต์ (ขนาดของ DATA Device จะได้ตั้งแต่ 1 เมกะไบต์จนถึง 32 GB

* ในช่อง Log Device ให้เลือก DATATESTLOG เสร็จแล้วคลิกที่ปุ่ม Create Now แล้วจะกลับมาที่หน้าจอของ Manage Database จะเห็นว่าฐานข้อมูล DATA_T ปรากฏอยู่ให้ดับเบิลคลิกเมาส์ที่ DATA_T หรือคลิกเมาส์ที่ทูลบาร์ Edit Database จะแสดงหน้าต่างของ Edit Database ดังรูปที่ 3.19 ก.

* จะแสดงรายละเอียดต่าง ๆ ของฐานข้อมูล เช่น วัน-เดือน-ปี/เวลา ในการสร้างขนาด Data size พื้นที่ว่าง และขนาดของ Log ที่ปุ่มทางขวามือคือ Expand และ shrink จะกล่าวภายหลัง



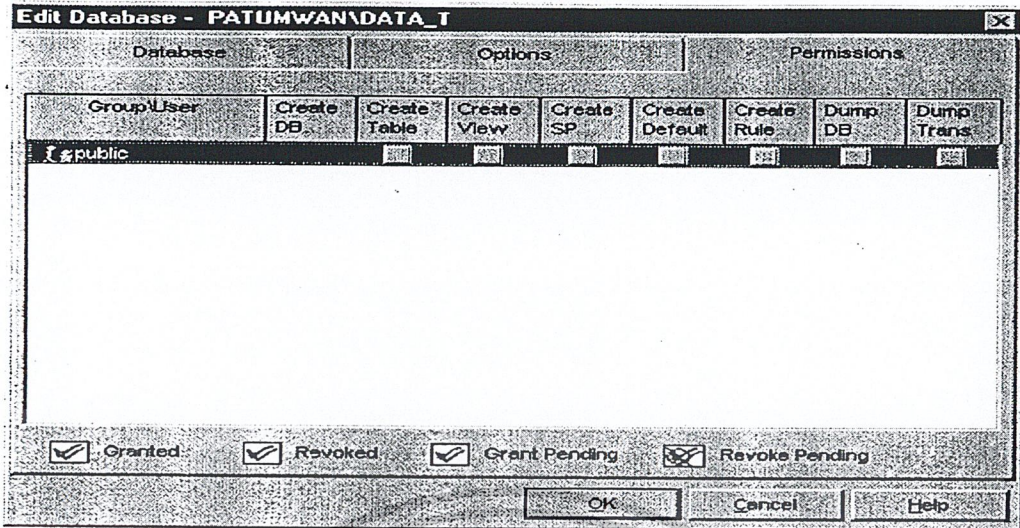
ก.



ข.

รูปที่ 3.19 แสดงการ Edit Database และ Database option

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก.

รูปที่ 3.19 (ต่อ) แสดงการ Edit Database และ Database option

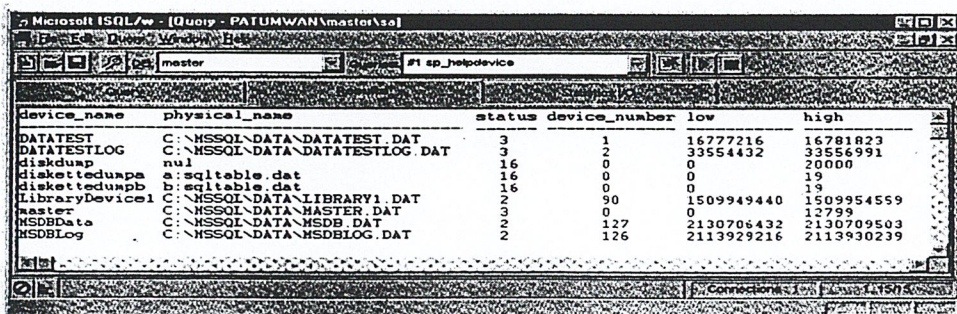
3.1.8.2. การสร้างฐานข้อมูลโดยใช้คำสั่ง CREATE DATABASE

การสร้างฐานข้อมูลด้วยคำสั่ง CREATE DATABASE นอกจากจะสร้างฐานข้อมูลแล้วยังสามารถจะสร้างทรานส์แอคชั่นล็อก (Transaction log) บนไดรฟ์ที่ต่าง ๆ ได้เช่นกัน

รูปแบบ

```
CREATE DATABASE database_name
[ON {DEFAULT : database_device} [=size]
[, database_device [=size]]....]
[LOG ON database_device [=size]
[, database_device [=size]]....]
[FOR LOAD]
```

ก่อนอื่นต้องสร้าง Log Device ชื่อ LibraryLogDev2 ให้มีขนาด 2048 2K หรือ 4 เมกะไบต์ และให้ Virtual Device number เป็น 92 เราสามารถจะใช้คำสั่ง Sp_helpdevice เพื่อช่วยตรวจสอบสถานะและ VDEVNO ได้ดังรูปที่ 3.20



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของ บริษัท ไมโครซอฟท์ (ประเทศไทย) จำกัด ผู้ดูแลระบบและใช้ประโยชน์ด้านการค้า
 รูปที่ 3.20 แสดงสถานะของ Virtual Device number (VDEVNO)
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.20 จะเห็นว่า LibraryDevice1 มี Device_number เป็น 90 หมายเลข VDEVNO หรือ Virtual Device number จะมีได้ตั้งแต่ 0-255 เราสามารถจะกำหนดเองได้ แต่อย่าให้ซ้ำกันกับที่มีอยู่ ส่วน VDEVNO หมายเลข 0 จะเป็นของ master

* เข้าสู่ ISQL/w เปิด Query ใช้ฐานข้อมูล master ที่แถบ Query พิมพ์คำสั่งเหล่านี้ ลงไป

```
DIS INT
NAME = 'Library Log Dev2',
PHYSNAME = 'C:\MSSQL\DATA\LIBRARYLOG.DAT',
VDEVNO = 92,
SIZE = 2048
```

เสร็จแล้วกดคีย์ Ctrl+E ถ้าไม่มีข้อผิดพลาดจะแสดงข้อความ “This Command did not return data, and it did not return and rows”

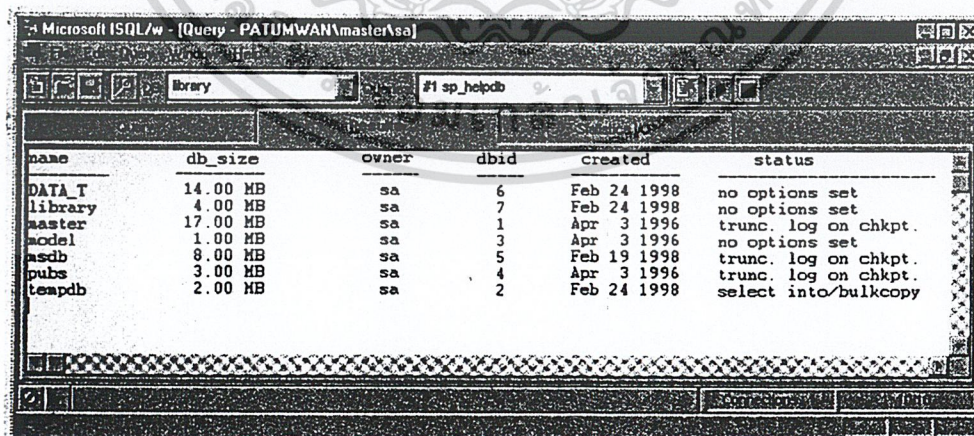
* ขั้นต่อไปจะสร้าง Library ฐานข้อมูลด้วยคำสั่ง CREATE DATABASE ที่หน้าต่าง Query ใช้ฐานข้อมูล master ที่แถบ Query พิมพ์คำสั่ง

```
USE master
CREATE DATABASE library
ON Library Device1
LOG ON Library Log Dev2
```

แล้วกดคีย์ Ctrl+E เพื่อเอ็กซิกิว ถ้าสร้าง Library เสร็จแล้วจะแสดงข้อความ

“CREATE DATABASE : allocating 1024 Page on disk ‘LibraryDevice1’

“CREATE DATABASE : allocating 1024 Page on disk ‘LibraryLogDev2’



name	db_size	owner	dbid	created	status
DATA_T	14.00 MB	sa	6	Feb 24 1998	no options set
library	4.00 MB	sa	7	Feb 24 1998	no options set
master	17.00 MB	sa	1	Apr 3 1996	trunc. log on chkpt.
model	1.00 MB	sa	3	Apr 3 1996	no options set
msdb	8.00 MB	sa	5	Feb 19 1998	trunc. log on chkpt.
pubs	3.00 MB	sa	4	Apr 3 1996	trunc. log on chkpt.
tempdb	2.00 MB	sa	2	Feb 24 1998	select into/bulkcopy

รูปที่ 3.21 แสดงการใช้คำสั่ง Sp_helpdb เพื่อดูรายละเอียดของฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.9. การ Dropping a Device

การสร้างดีไวท์หรือฐานข้อมูลขึ้นมาในระบบย่อมจะต้องการพื้นที่ในหน่วยความจำถ้ามีดีไวท์ไหนที่ไม่ต้องการใช้ควรลบทิ้งไว้ เพื่อเป็นการเพิ่มพื้นที่ในหน่วยความจำ ของระบบสามารถทำได้ตามขั้นตอนต่อไปนี้

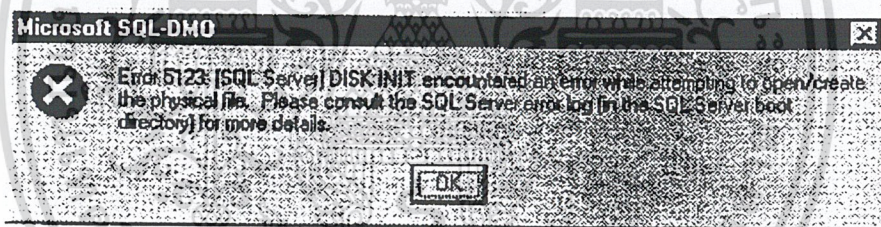
* ก่อนอื่นจะให้สร้าง Device ขึ้นมา 1 ตัวเพื่อใช้ทดสอบเข้าสู่ SQL Enterprise Manager เลือกเซิร์ฟเวอร์ SQL แล้วคลิกเมาส์ปุ่มขวาที่ Database Devices จะปรากฏเมนูคำสั่งให้เลือก New Device

* จะปรากฏหน้าต่างของ New Database Device ให้พิมพ์ชื่อในช่อง Name ว่า DATA_007 คลิกเมาส์ที่เช็คบ็อก Default Device ในช่อง Location เลือก C: ส่วนขนาดเป็น 2 เมกะไบต์ เสร็จแล้วคลิกที่ปุ่ม Create Now จะเห็นว่ามี Device ตัวใหม่คือ DATA_007 ปรากฏให้เห็น

* ต่อไปเราจะทำการ Drop device ของ DATA_007 ที่ เพื่อเพิ่มพื้นที่ให้ฮาร์ดิสต์ คลิกเมาส์ปุ่มขวาที่ device ของ DATA_007 เลือกคำสั่ง Delete/Yes ตามลำดับ

* ตอนนี้เราจะมาสร้าง Device ใหม่ในชื่อเดิม (DATA_007) แล้วคลิกที่ปุ่ม Create Now จะแสดงข้อความ error ดังรูปที่ 3.22 มันผิด พรบ. SQL เป็นเพราะไฟล์ operationg system ของ Device ยังไม่ได้ถูกลบทิ้ง

* ให้เปิด windows Explorer แล้วลบไฟล์ดังกล่าวในไดเรกทอรี C:\MSSQLDATA\DATA_007.DAT หลังจากนั้นให้สร้าง Device ตัวนี้อีกครั้ง นอกจากนี้ยังสามารถจะใช้ Stored Procedure ในการลบดีไวท์ ด้วยคำสั่ง Sp_dropdevice logical_name



รูปที่ 3.22 แสดงข้อความผิดพลาดในการสร้าง Device

3.1.10. การ Dropping Database

การลบฐานข้อมูล (Database) ที่ไม่ต้องการ ออกจากระบบสามารถทำได้เพียง 2 คน คือ SA และ dbo (Database owner) หรือเจ้าของฐานข้อมูลเท่านั้น สิทธิ์ในการลบฐานข้อมูลไม่สามารถโอนให้ผู้ใช้คนอื่นได้ ในการลบฐานข้อมูลสามารถทำได้ 2 วิธีคือการใช้ SQL Enterprise Manager และการใช้สแตทเมนต์ DROP DATABASE database_name การ drop ฐานข้อมูลจะเป็นการลบฐานข้อมูลนั้นออกจากตาราง sysdatabase และ sysusage จะมีฐานข้อมูลอยู่ 3 ตัว คือ master, model และ tempdb ที่ไม่สามารถจะลบได้

3.1.11. การ Expanding Database

เมื่อใช้งานฐานข้อมูลไปนาน ๆ ข้อมูลก็จะเพิ่มขึ้น ทำให้ต้องมีการขยาย (Expanding) ฐานข้อมูลให้มีขนาดใหญ่ขึ้นตามไปด้วย ในการขยายฐานข้อมูลสามารถทำได้ 2 วิธีคือ การใช้ SQL Enterprise Manager และการใช้สแตทเมนต์ ALTER DATABASE ซึ่งมีรูปแบบดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

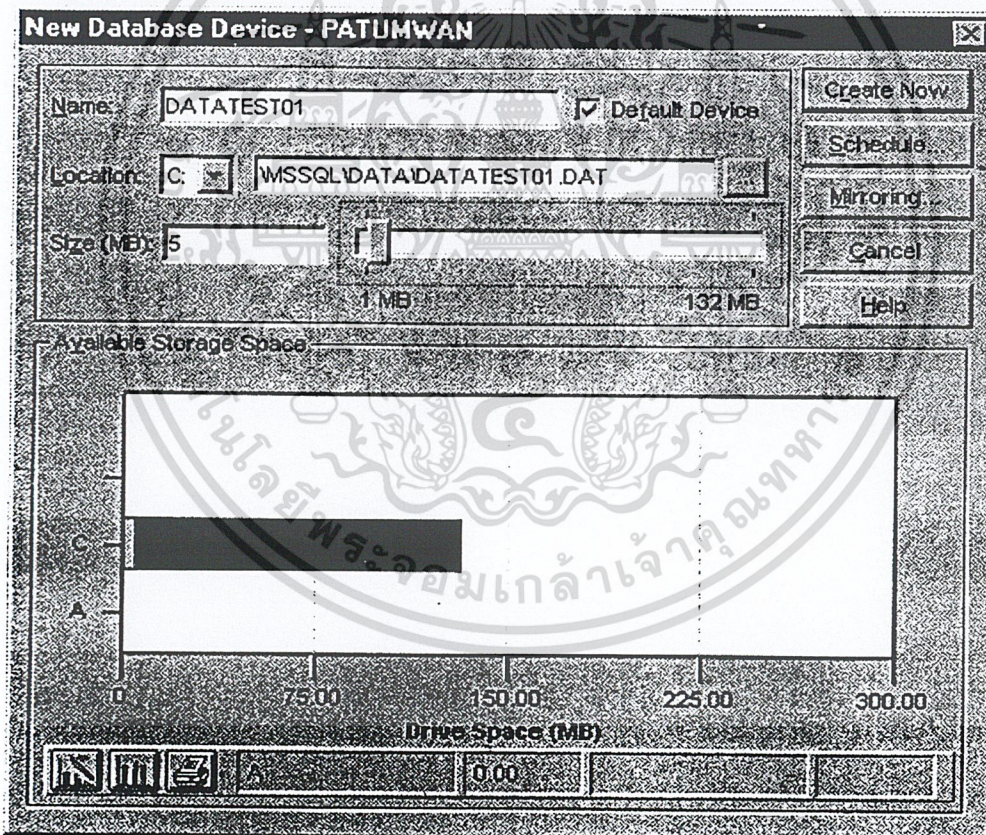
```
[ON {DEFAULT : database_device}[=size]
[,database_device [=size]]....]
[FORLOAD]
```

* เข้าสู่ SQL Enterprise Manager เลือกเซิร์ฟเวอร์ SQL แล้วคลิกที่โฟลเดอร์ของ Database จะเห็นว่า มีฐานข้อมูลอยู่หลายตัวด้วยกัน ให้ดับเบิลคลิกที่ DATA_T จะแสดงหน้าต่างของ Edit Database

* โดยจะแสดงขนาด (Data size,Log size,Data space Available) ของฐานข้อมูล DATA_T ให้คลิกที่ปุ่ม Expand จะแสดงหน้าต่างของ Expand Database

* ที่ช่อง Data Device เลือก New จะแสดงหน้าต่างของ New Database Device ตรงช่อง Name ให้พิมพ์ชื่อ DATAEST01 กำหนดขนาด 5 เมกะไบต์ ดังรูปที่ 3.23 ก. เสร็จแล้วคลิกที่ปุ่ม Create Now

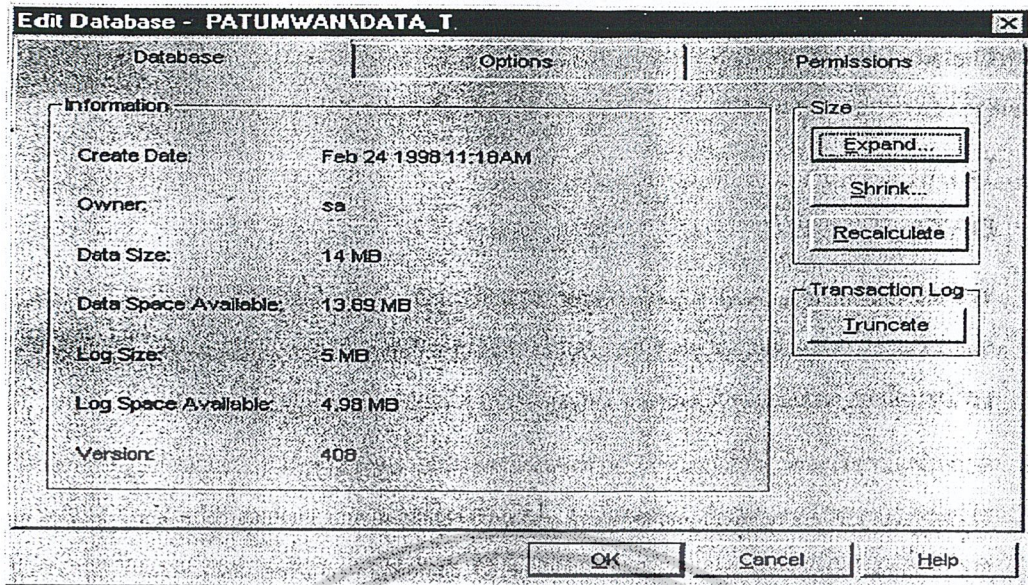
* จะกลับมาที่หน้าต่างของ Expand Database จะเห็นว่า มี DATATEST01 ปรากฏอยู่เป็นรูปกราฟแนวนอนสีเหลือง ให้คลิกที่ปุ่ม Expand Now จะกลับมาที่หน้าต่างของ Edit Database ดังรูปที่ 3.23 ข. จะเห็นว่าขนาด Data size, DATA space Available เพิ่มขึ้นจากเดิม



ก.

รูปที่ 3.23 แสดงการขยายฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ข.

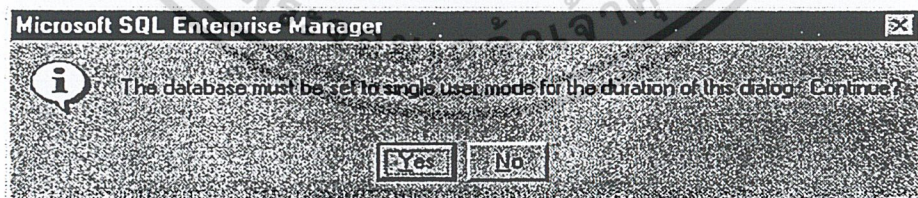
รูปที่ 3.23 (ต่อ) แสดงการขยายฐานข้อมูล

3.1.12. การลดขนาดข้อมูลด้วย Shrink

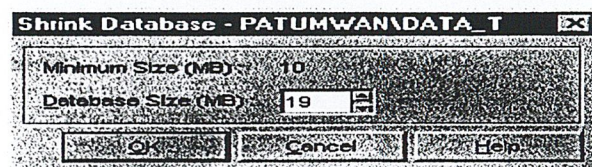
การลดขนาดฐานข้อมูลจะเป็นการเพิ่มพื้นที่ในหน่วยความจำของระบบ แต่ก่อนจะทำการลดหรือ shrink ฐานข้อมูล ควรจะกำหนดอ็อบชั่นของฐานข้อมูลให้เป็น Single User โหมดก่อน ดังรูปที่ 3.20 ข. เพื่อป้องกันผู้ใช้อื่นเข้ามาใช้ข้อมูลในขณะนั้น การลดขนาดฐานข้อมูลสามารถทำได้ 2 วิธีคือ การใช้ SQL Enterprise Manager และ สเตทเมนต์ DBCC SHRINKDDB โดยทำตามขั้นตอนต่อไปนี้

* เข้าสู่ SQL Enterprise Manager เลือกเซิร์ฟเวอร์ SQLI คลิกที่โฟลเดอร์ Database แล้วดับเบิลคลิกที่ฐานข้อมูล DATA_T จะแสดงหน้าต่างของ Edit Database

* ให้คลิกที่ปุ่ม shrink จะแสดงข้อความ “ฐานข้อมูลจะต้องเซตในโหมด single user ในระหว่างการทำ Shrink” ดังรูปที่ 3.24 ก. ให้ตอบ Yes จะปรากฏหน้าต่างของ Shrink Database ดังรูปที่ 3.24 ข. จะเห็นว่ามีขนาด Minimum size กำหนดไว้คือ 10 ห้ามลดขนาดต่ำกว่านี้



ก.



ข.

เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 3.24 แสดงการลดขนาดฐานข้อมูลด้วย shrink ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การลดขนาดฐานข้อมูลด้วย สเตทเมนต์ DBCC SHRINKDB มีรูปแบบดังนี้

DBCC {SHRINKDB (database_name [, new_size[, 'MASTEROVERRIDE']])} เช่นถ้าต้องการจะลดขนาดฐานข้อมูล ชื่อ BASIC_DATA ให้พิมพ์คำสั่ง

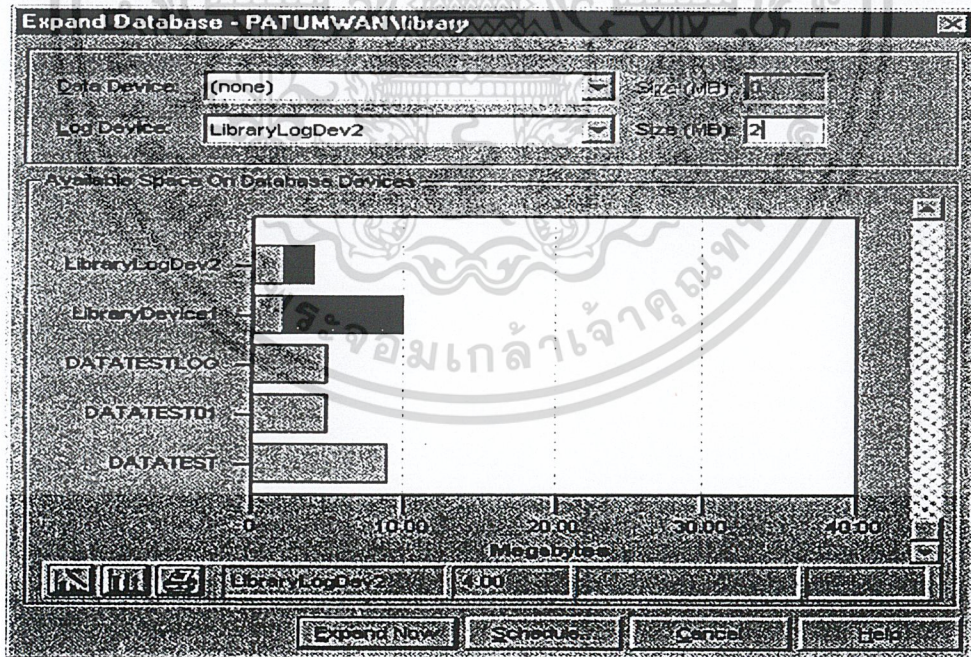
DBCC SHRINKDB (BASIC_DATA, ใส่ขนาดหน่วยความจำ

3.1.13. การขยาย Transaction Log

Transaction Log จะเก็บและบันทึกข้อมูลที่เปลี่ยนแปลงในการทำรายการ ไม่ว่าจะเป็นการเพิ่ม ลบ อัปเดตข้อมูลในตาราง จะถูกเก็บไว้ใน Transaction Log ทั้งสิ้น ถ้าหาก Transaction Log เกิดเต็มจะต้องมีการขยายพื้นที่เพิ่มขึ้น การขยาย Transaction Log สามารถทำได้ 3 วิธี ด้วยกันคือ ใช้ SQL Enterprise Manager, Stored Procedure และสเตทเมนต์ ALTER DATABASE ในการใช้ SQL Enterprise Manager ทำได้ตามขั้นตอนต่อไปนี้

* เข้าสู่ SQL Enterprise Manager เลือกเซิร์ฟเวอร์ SQL คลิกที่ทูลบาร์ตัวที่ 8 หรือเมนูคำสั่ง Manager/Database จะแสดงหน้าต่างของ Manager Databases ให้ดับเบิลคลิกที่ฐานข้อมูล library ก็จะปรากฏหน้าต่างของ Edit Database

* คลิกที่ปุ่ม Expand จะแสดงหน้าต่างของ Expand Database ที่ช่อง Log Device ให้เลือก Library Log Dev2 ดังรูปที่ 3.25 ตรงช่อง size MB ใส่เลข 2 เสร็จแล้วคลิกที่ปุ่ม Expand Now/OK ตามลำดับ



รูปที่ 3.25 แสดงการขยาย Transaction Log

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้สแตทเมนต์ ALTER DATABASE มีรูปแบบดังนี้

```
ALTER DATABASE database_name
[ON{DEFAULT : database_device}[=size]
[, database_device[=size]]....]
[FOR LOAD]
```

การใช้ stored Procedures มีรูปแบบดังนี้

```
Sp_logdevice dbname,database_device
```

3.2. ภาษาสอบถามเชิงโครงสร้าง (Structure Query Language)

ภาษาสอบถามเชิงโครงสร้าง (Structured Query Language) หรือที่เรียกว่าสั้น ๆ ว่าเอสคิวแอล (SQL) หรือซีคิวแอล SE-QUEL นั้นเป็นภาษาที่ใช้ในการจัดการข้อมูลของฐานข้อมูลเชิงสัมพันธ์ ผู้ผลิตซอฟต์แวร์ด้านระบบจัดการฐานข้อมูลที่สนับสนุน SQL มากขึ้น เช่น DE2 ORAGESS FoxPro dBase IV ฯลฯ รูปแบบของ SQL ที่ใช้ในระบบจัดการฐานข้อมูลแต่ละชนิดจะแตกต่างกันไปบ้าง ทั้งนี้เพื่อเป็นการแสดงความสามารถและจุดเด่นของระบบจัดการฐานข้อมูลของแต่ละบริษัท อย่างไรก็ตาม American National Standards Institute (ANSI) ได้กำหนดรูปแบบ SQL มาตรฐานขึ้นเพื่อให้เป็นมาตรฐานทั่วไปของ SQL

ในหัวข้อนี้จะกล่าวถึงรูปแบบของการใช้คำสั่ง SQL และรายละเอียดคำสั่งของ SQL โดยจะแบ่งคำสั่งของ SQL ตามเกณฑ์ภาษาที่ใช้ในระบบจัดการฐานข้อมูล คือ SQLที่ใช้สำหรับนิยามข้อมูล (DDL) และ SQL ที่ใช้สำหรับจัดการข้อมูล (DML) ส่วน SQL ที่ใช้การควบคุมข้อมูล (DCT)

3.2.1. รูปแบบการใช้คำสั่ง SQL

รูปแบบของการใช้คำสั่ง SQL ที่สามารถใช้ได้เป็น 2 รูปแบบ ดังนี้ คือ

3.2.1.1. คำสั่ง SQL ที่ใช้เรียกดูข้อมูลได้ทันที (Interactive SQL)

เป็นการใช้คำสั่ง SQL สั่งงานบนจอภาพ เพื่อเรียกดูข้อมูลในขณะที่ทำงานได้ทันที เช่น

```
SELECT CITY
FROM SUPPLIER
WHERE SNO = 'S4';
```

ผลที่ได้ คือ

```
CITY
BANGKOK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.2. คำสั่ง SQL ที่ใช้เขียนร่วมกับโปรแกรมอื่น ๆ (Embedded SQL)

เป็นคำสั่ง SQL ที่ใช้ร่วมกับคำสั่งของโปรแกรมภาษาต่าง ๆ เช่น PL/1 PASCAL ฯลฯ หรือแม้แต่กับคำสั่งในโปรแกรมที่ระบบจัดการฐานข้อมูลนั้นมิใช่เฉพาะ เช่น ORACLE มี PL/SQL (Procedural Language /SQL) ที่สามารถเขียนโปรแกรม และนำคำสั่ง SQL มาเขียนร่วมด้วย เป็นต้น

ตัวอย่างการใช้คำสั่ง SQL ในภาษา PL/1

```
EXEC SQL SELECT SELECT CITY
INTO : XCITY
FROM SUPPLIER
WHERE SNO = 'S4';
```

3.2.2. คำสั่ง SQL ที่ใช้สำหรับนิยามข้อมูล

3.2.2.1. CREATE TABLE

เป็นคำสั่ง SQL ที่ใช้ในการสร้างรีเลชัน โดยการระบุชื่อแอททริบิวต์ต่าง ๆ ขนาดความกว้างของแอททริบิวต์ ประเภทของข้อมูลที่จัดเก็บในแอททริบิวต์นั้น รวมถึง การระบุว่าแอททริบิวต์ใดเป็นคีย์หลักหรือคีย์นอก

รูปแบบ

```
CREATE TABLE <TABLE NAME >
(<COLUMN NAME ><DATA TYPE > [<WIDTH >
[,<COLUMN NAME > <DATA TYPE > [<WIDTH >],.....])
[,<PRIMARY KEY DEFINITION >]
[,<FOREIGN KEY DEFINITION >];
```

COLUMN NAME เป็นชื่อของแอททริบิวต์ของรีเลชัน
WIDTH คือขนาดความกว้างของข้อมูล
DATA TYPE คือ ประเภทค่าของข้อมูล ซึ่งประเภทของข้อมูลอาจเป็นได้ ดังนี้ คือ

ประเภท	ความหมาย
SMALLINT	เป็นประเภทของข้อมูลที่เป็นเลขจำนวนเต็ม
INTEGER	เป็นประเภทของข้อมูลที่เป็นเลขจำนวนเต็มที่ขนาดกว้างกว่า SMALLINT
DECIMAL (X,Y)	เป็นประเภทของตัวเลขทศนิยม จำนวน X หลัก โดยมีตัวเลขหลังจุดทศนิยม Y จำนวน
FLOAT (X, Y)	เป็นประเภทของตัวเลขที่กำหนดในรูปแบบยกกำลัง
CHARACTER (N)	เป็นสตริงที่มีขนาดคงที่ตามที่ระบุ
VARCHAR	สตริงขนาดไม่คงที่
DATE	เป็นข้อมูลประเภท เดือน/วัน/ปี
LOGICAL	เป็นข้อมูลที่มีค่าเป็นตรรกที่เป็น "T"(True) หรือ "F" (False)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และห้ามทำซ้ำโดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อแหล่งอื่นและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 1 ให้สร้างรีเลชัน SUPPLIER PRODUCT และ ORDER
รีเลชัน SUPPLIER คำสั่งที่ใช้คือ

```
CREATE TABLE SUPPLIER
(SNO CHAR(5),
SNAME CHAR (20),
CITY CHAR(15),
PRIMARY KEY (SNO));
```

รีเลชัน PRODUCT คำสั่งที่ใช้คือ

```
CREATE TABLE PRODUCT
(PNO CHAR(5),
PNAME CHAR(20),
COLOR CHAR (10),
CITY CHAR (15),
PRIMARY KEY (PNO));
```

รีเลชัน ORDER คำสั่งที่ใช้คือ

```
CREATE TABLE ORDER
(SNO CHAR (5),
PNO CHAR (5),
QTY INTEGER,
PRIMARY KEY (SNO, PNO),
FOREIGN KEY (SNO) REFERENCES SUPPLIER (SNO),
FOREIGN KEY (PNO) REFERENCES PRODUCT (PNO));
```

3.2.2.2. ALTER TAVLE เป็นคำสั่งที่ใช้ในการเพิ่มแอททริบิวต์ในรีเลชันที่มีอยู่เดิม
รูปแบบ

```
ALTER TABLE < TABLE NAME >
ADD ( < COLUMN NAME > < DATA TYPE > < WIDTH > [, < COLUMN NAME >
< DATA TYPE > < WIDTH > ,.....]);
```

ตัวอย่างที่ 2 ต้องการเพิ่มแอททริบิวต์ชื่อ COLOR เพิ่มอีก 1 แอททริบิวต์ลงใน รีเลชัน PRODUCT โดยให้มี
ลักษณะเป็นตัวอย่างที่มีความยาว 6 ตัวอักษร คำสั่งที่ใช้คือ

```
ALTER TABLE PRODUCT
ADD (COLOR CHAR(6) );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.3. DROP TABLE คำสั่งที่ใช้ในการลบรีเลชั่น

รูปแบบ

```
DROP TABLE <TABLE NAME >;
```

3.2.2.4. CREATE INDEX เป็นคำสั่งที่ใช้ในการสร้างอินเด็กซ์

รูปแบบ

```
CREATE [ UNIQUE ] INDEX <INDEX NAME >  
ON THABLE NAME (COLUMN NAME [,COLUMN NAME]);
```

ตัวอย่างที่ 3 ต้องการสร้างอินเด็กซ์โดยใช้แอททริบิวต์ QTY ในรีเลชั่น ORDER

```
CREATE INDEX QTYNDX  
ON ORDER (QTY);
```

3.2.2.5. DROP INDEX เป็นคำสั่งที่ใช้ในการลบ INDEX

รูปแบบ

```
DROP INDEX <INDEX NAME >;
```

3.2.2.6. CREATE VIEW เป็นคำสั่งที่ใช้ในการสร้าง VIEW

รูปแบบ

```
CRATE VIEW <VIEW NAME >  
AS SELECT Statement;
```

ตัวอย่างที่ 4 ต้องการสร้างวิวจากรีเลชั่น ORDER โดยแสดงข้อมูลเฉพาะที่มี QTY มากกว่า 200

```
CREATE VIEW ORDERVIEW  
AS SELECT SNO, PNO, QTY  
FROM ORDER  
WHERE QTY > 200;
```

3.2.2.7. DROP VIEW เป็นคำสั่งที่ใช้ในการลบ VIEW

รูปแบบ

```
DROP VIEW <VIEW NAME >;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3. คำสั่ง SQL ที่ใช้สำหรับจัดการข้อมูล

หลังจากเมื่อการสร้างรีเลชันต่าง ๆ ในฐานข้อมูลเชิงสัมพันธ์แล้ว เมื่อต้องการเรียกข้อมูลมาใช้หรือมีการเพิ่มเติมข้อมูลต่าง ๆ คำสั่งที่ใช้มีดังต่อไปนี้ คือ

3.2.3.1. INSERT เป็นคำสั่งที่ใช้ในการใส่ข้อมูลทุเปลลงในรีเลชันหนึ่ง โดยการเพิ่มข้อมูลจะเพิ่มได้ทีละหนึ่งทุเปลต่อครั้ง

รูปแบบ

```
INSERT INTO <TABLE NAME >
[ <COLUMN LIST > ] VALUES ( <VALUE LIST > );
```

TABLE NAME คือ ชื่อของรีเลชัน ซึ่งจะมีการเพิ่มข้อมูล

COLUMN LIST คือ ชื่อของแอททริบิวต์ที่ต้องการเพิ่มข้อมูลลงไป ถ้ารีเลชันมีมากกว่า 1 แอททริบิวต์ ให้ระบุโดยมีเครื่องหมาย , คั่น

VALUE LIST คือ ค่าของแอททริบิวต์ ที่ต้องการเพิ่มเข้าไปในรีเลชันและถ้าเป็นข้อมูลประเภทตัวอักษรจะต้องมีเครื่องหมาย ' ' กำกับ

ตัวอย่างที่ 5 ให้ใส่ข้อมูลในรีเลชัน PRODUCT

```
INSERT INTO PRODUCT (PNO, PNAME, COLOR, CITY)
VALUES ('P', 'PEN', 'BANKOK');
```

3.2.3.2. UPDATE เป็นคำสั่งที่ใช้ในการเปลี่ยนแปลงข้อมูลในแต่ละรีเลชัน

รูปแบบ

```
UPDATE <TABLE >
SET <COLUMN NAME 1 > = <VALUE1 >
[ , <COLUMN NAME 2 > = <VALUE2 >
, <COLUMN NAME N > = <VALUE N > }
[WHERE <CONDITIONS >];
```

TABLE หมายถึง รีเลชันที่ข้อมูลต้องการเปลี่ยนแปลงค่าใหม่

COLUMN NAME หมายถึง ชื่อของแอททริบิวต์ ที่ต้องการเปลี่ยนค่าใหม่

[WHERE < conditions > หมายถึง เงื่อนไขที่ใช้ในการเลือกข้อมูลที่จะนำมาเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ตัวอย่างที่ 6 ให้ปรับค่า P2 ให้มีสีเป็น YELLOW

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเหตุผลแบ่ส่งเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
UPDATE PRODUCT
SET COLOR = 'YELLOW'
WHERE PNO = 'P';
```

3.2.3.3. DELETE เป็นคำสั่งที่ใช้ในการลบข้อมูลในแต่ละทิวเพิล
รูปแบบ

```
DELETE FORM < TABLE NAME > [ WHERE < CONDITION > ];
```

ตัวอย่างที่ 7 ให้ลบผู้ผลิตที่มีรหัส S5

```
DELETE FROM SUPPLIER
WHERE SNO = 'S5';
```

3.2.3.4. SELECT เป็นคำสั่งที่ใช้ในการเรียกข้อมูลจากฐานข้อมูลตามที่ใช้ต้องการ
รูปแบบ

```
SELECT [ ALL/DISTINCT ] < TARGET LIST >
FROM < TABLE LIST > [ < ALLASES > ]
[ WHERE PREDICATE ]
[ GROUP BY < COLUMN LIST > [ HAVING PREDICATE ] ]
[ CRDER BY < COLUMN LIST > ];
```

ALL/DISTNCT เป็นการสั่งให้เลือกข้อมูลที่ต้องการ ถ้าใช้ คำว่า ALL เครื่องก็จะเลือกข้อมูลที่เป็นไปตามเงื่อนไขออกมาให้หมด ซึ่งสามารถใช้เครื่องหมาย * แทนได้ แต่ถ้าใช้ DISTINCT เครื่องจะเลือกข้อมูลที่เป็นไปตามเงื่อนไขเฉพาะทิวเพิลที่มีข้อมูลไม่ซ้ำกัน

TARGET LIST เป็นชื่อของแอททริบิวต์ และนิพจน์ที่ต้องการให้แสดงค่าออกมาตามที่ต้องการ

FROM <TABLE LIST [ALLASES]> เป็นการระบุถึงชื่อรีเลชัน ที่ต้องการให้ดึงข้อมูลออกมาส่วน
ALLASES คือ ชื่อของรีเลชันที่ตั้งขึ้นใหม่

[WHERE < PREDICATE >] หมายถึง เงื่อนไขที่ระบุขึ้นเพื่อดึงข้อมูลของทิวเพิลตามที่ต้องการ ในส่วนของเงื่อนไขนี้จะกล่าวในรายละเอียดต่อไป

[GROUP BY < COLUMN LIST > [HAVING PREDICATE]] เป็นการจัดกลุ่มตามเงื่อนไขที่ระบุและภายใต้การจัดกลุ่มก็จะมีเงื่อนไขที่จะให้แสดงผลที่เกิดจากจัดกลุ่ม โดยใช้ HAVING

[ORDER BY < COLUMN LIST >] เป็นการระบุให้เรียงลำดับข้อมูลตามแอททริบิวต์ที่ต้องการถ้ามี ASC ต่อท้ายชื่อของแอททริบิวต์เป็นการเรียงจากน้อยไปหามาก และถ้าเป็น DESC เป็นการเรียงจากมากไปหาน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเรียกดูข้อมูล ข้อมูลที่ผู้ใช้เรียกดูจากกริเลขันต่าง ๆ อาจจะไม่มีเงื่อนไขหรือมีเงื่อนไขก็ได้ โดยเงื่อนไขที่ระบุหลัง WHERE สามารถทำได้หลายรูปแบบ ไม่ว่าจะเป็นเพียงเงื่อนไขเดียว หรือหลายเงื่อนไข นอกจากนี้ ยังสามารถระบุเงื่อนไขโดยเขียนเป็นข้อคำถามย่อย (Subquery) ได้เช่นกัน รายละเอียดที่เกี่ยวข้องกับการเรียกดูข้อมูลมีดังนี้ คือ

3.2.3.4. 1 การเรียกดูข้อมูลแบบไม่มีเงื่อนไข

การเรียกดูข้อมูลแบบไม่มีเงื่อนไข อาจจะเป็นการเรียกดูข้อมูลทั้งริเลขัน หรือเรียกดูเพียงบางแอททริบิวต์หรือการเรียกดูข้อมูลโดยการให้จัดเรียงข้อมูล

ตัวอย่างที่ 8 ให้แสดงข้อมูลทั้งหมดริเลขัน SUPPLIER

```
SELECT *
FROM SUPPLIER;
```

ผลที่ได้ คือ SUPPLIER

SND	SNAME	CITY
S1	SERI	BANGKOK
S2	WANIDA	RAYONG
S3	SOMCHAI	RAYONG
S4	ORAPIN	BANGKOK

ตัวอย่าง ที่ 9 ให้แสดงรหัสสินค้า และชื่อสินค้า

```
SELECT PNO , PNAME
FROM PRODUCT;
```

ผลที่ได้ คือ

PNO	PNAME
P1	PEN
P2	TABLE
P3	DESK
P4	BAG
P5	CHAIR
P6	TELEPHONE

3.2.3.4.2 การเรียกดูข้อมูลแบบมีเงื่อนไข

การเรียกดูข้อมูลมีเงื่อนไข เป็นการระบุชื่อแอททริบิวต์ที่ต้องการระบุเป็นเงื่อนไข และข้อมูลเฉพาะในอนุประโยค WHERE โดยการระบุเงื่อนไขจะนำโอเปอเรเตอร์ต่าง ๆ เข้ามาใช้ในการแสดงเงื่อนไข โอเปอเรเตอร์ดังกล่าวมีดังต่อไปนี้ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.4.2.1. การเรียกดูข้อมูลแบบมีเงื่อนไข

โดยใช้โอเปอเรเตอร์ทางตรรกะและโอเปอเรเตอร์บูลีนโอเปอเรเตอร์ทางตรรกะ (Logical Operator) ใช้แสดงการเปรียบเทียบค่าของข้อมูล ซึ่งจะแสดงอยู่ระหว่างชื่อคอลัมน์ และข้อมูลเฉพาะที่ต้องการแสดงเป็นเงื่อนไข โอเปอเรเตอร์ดังกล่าว ประกอบด้วย

โอเปอเรเตอร์	ความหมาย
>	มากกว่า
<	น้อยกว่า
=	เท่ากับ
>=	มากกว่า เท่ากับ
<=	น้อยกว่า เท่ากับ
<>	ไม่เท่ากับ

ส่วนโอเปอเรเตอร์บูลีน (Boolean Operator) ใช้สำหรับการเรียกดูข้อมูลที่มีเงื่อนไขมากกว่าหนึ่งเงื่อนไข เช่น AND หรือ OR หรือใช้ในการปฏิเสธเงื่อนไข เช่น NOT

ตัวอย่างที่ 10 ให้แสดงรหัสผู้ผลิต สำหรับผู้ผลิตที่อยู่ ที่ RAYONG หรือ BANGKOK

```
SELECT SNO
FROM SUPPLIER
WHERE CITY = 'RAYONG' OR 'BANGKOK';
```

ผลที่ได้ คือ

SNO
S1
S2
S3
S4

3.2.3.4.2.2 การเรียกดูข้อมูลแบบมีเงื่อนไขโดยใช้โอเปอเรเตอร์ของ SQL

โอเปอเรเตอร์ของ SQL ที่ใช้ในการกำหนดเงื่อนไขของข้อมูลในอนุประโยค WHERE มีดังนี้คือ

ก. โอเปอเรเตอร์ BETWEEN.....AND.....

เป็นโอเปอเรเตอร์ที่กำหนดเงื่อนไขของแอททริบิวต์เป็นค่าระหว่างสองค่า โดยโอเปอเรเตอร์

BETWEEN จะแสดงต่อจากแอททริบิวต์ที่ถูกกำหนดให้เป็นเงื่อนไข ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 11 ให้แสดงข้อมูลแอททริบิวต์ SNO และ QTY จากกริเลขัน ORDER ที่ส่งสินค้าระหว่าง 300 และ 500 หน่วย

```
SELECT SNO, PNO, QTY
FROM ORDER
WHERE QTY BETWEEN 300 AND 500;
```

ผลที่ได้ คือ

SNO	PNO	QTY
S1	P3	300
S2	P1	300
S2	P2	400
S4	P2	500
S4	P4	300
S4	P5	400

ข. โอเปอเรเตอร์ IN

เป็นโอเปอเรเตอร์ที่ใช้กับเงื่อนไขของคอลัมน์ที่ต้องการระบุเงื่อนไขเป็นกลุ่มของข้อมูลโดยโอเปอเรเตอร์ IN จะแสดงต่อท้ายชื่อแอททริบิวต์ที่ถูกระบุเป็นเงื่อนไขนี้จะอยู่ในวงเล็บ () และมีเครื่องหมาย, คั่น

ตัวอย่างที่ 12 ให้แสดงรายละเอียดของ รหัส และชื่อผู้ผลิต ที่ อยู่ BANGKOK หรือ RAYONG

```
SELECT SNO, SNAME
FROM SUPPLIER
WHERE CITY IN ('BANGKOK','RAYONG');
```

ผลที่ได้ คือ

SNO	SNAME
S1	SERI
S2	WANIDA
S3	SOMCHAI
S4	ORAPIN

ค. โอเปอเรเตอร์ LIKE

เป็นโอเปอเรเตอร์ที่ใช้ในการค้นหาข้อมูลของคอลัมน์ที่เก็บข้อมูลประเภทตัวอักษรเท่านั้น (CHARACTER) โดยที่ยังไม่ทราบค่าที่แน่นอนทั้งหมดของข้อมูลที่จะค้นหา หรือรู้เพียงบางตัวอักษรเท่านั้น โอเปอเรเตอร์ LIKE จะระบุต่อท้ายชื่อคอลัมน์ที่เป็นเงื่อนไข โดยจะใช้สัญลักษณ์ที่ช่วยในการค้นหาข้อมูลเป็นตัวช่วยในการค้นหาข้อมูลที่เรียกว่า “Wild Card” สัญลักษณ์ดังกล่าวประกอบด้วย % และ _ เครื่องหมายขีดเส้นใต้ โดยข้อมูลบางส่วนที่ใช้ในการค้นหาพร้อมกับสัญลักษณ์ทั้งสองนี้ จะต้องมีเครื่องหมาย ‘_’ กำกับเสมอ ความหมายของสัญลักษณ์ทั้งสองเป็นดังนี้ คือ

สัญลักษณ์

%

ความหมาย

ใช้แทนจำนวนตัวอักษรได้หลายตัวเช่น พนักงานที่มีชื่อขึ้นต้นด้วย T จะเขียนเงื่อนไข 'T%' ใช้แทนตัวอักษรที่ไม่ทราบค่า 1 ตัว เช่น พนักงานที่มีชื่อขึ้นต้นด้วย S และมีความหมาย 7 ตัวอักษร เช่น

```
WHERE EMPNAME LIKE 'S_____'
```

ตัวอย่างที่ 13 ให้แสดง PNAME ที่ขึ้นต้นด้วยอักษร C

```
SELECT *
FROM PRODUCT
WHERE PNAME LIKE 'C%';
```

ผลที่ได้ คือ

PNO	PNAME	COLOR	CITY
P5	CHAIR	BLUE	RAYONG

3.2.3.4. 3 การเรียกดูข้อมูลจากรีเลชันมากกว่าหนึ่งรีเลชัน

คำสั่ง SELECT ยังสามารถใช้ในการเรียกดูข้อมูลหรือเชื่อมโยงจากข้อมูลจากรีเลชันมากกว่าหนึ่งรีเลชัน หรือที่เรียกว่า JOIN การเชื่อมโยงข้อมูลจากหลายรีเลชัน อาจจะเรียกดูข้อมูลแบบมีเงื่อนไขหรือไม่มีเงื่อนไขก็ได้

3.2.3.4.3.1 Equi Join

เป็นการเชื่อมโยงจากรีเลชันหลายรีเลชันเข้าด้วยกัน โดยแต่ละรีเลชันมีข้อมูลของแอททริบิวต์ที่ตรงกันตามเงื่อนไขที่ระบุไว้ โดยข้อมูลที่แสดงออกมามีข้อมูลของแอททริบิวต์ที่ตรงกันตามเงื่อนไข แสดงออกมาเป็นผลลัพธ์ซ้ำกัน กล่าวอีกนัยหนึ่งคือ เงื่อนไขที่ระบุจะเป็นเครื่องหมายเท่ากับ (=)

ตัวอย่างที่ 14 ให้แสดงข้อมูลของรีเลชัน SUPPLIER และ PRODUCT ที่อยู่จังหวัดเดียวกัน

```
SELECT S.*,P.*
FROM SUPPLIER S, PRODUCT P
WHERE S.CITY = P.CITY;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลที่ได้คือ

SNO	SNAME	CITY	PNO	PNAME	COLOR	CITY
S1	SERI	BANGKOK	P1	PEN	RED	BANGKOK
S1	SERI	BANGKOK	P4	BAG	RED	BANGKOK
S1	SERI	BANGKOK	P6	TELEPHONE	RED	BANGKOK
S2	WANIDA	RAYONG	P2	TABLE	GREEN	RAYONG
S2	WANIDA	RAYONG	P5	CHAIR	BLUE	RAYONG
S3	SOMCHAI	RAYONG	P2	TABLE	GREEN	RAYONG
S3	SOMCHAI	RAYONG	P5	CHAIR	BLUE	RAYONG
S4	ORAPIN	BANGKOK	P1	PEN	RED	BANGKOK
S4	ORAPIN	BANGKOK	P4	BAG	RED	BANGKOK
S4	ORAPIN	BANGKOK	P6	TELEPHONE	RED	BANGKOK

3.2.3.4.3.2. Natural Join

เป็นการเชื่อมโยงข้อมูลในลักษณะเช่นเดียวกับ Equijion เพียงแต่ผลลัพธ์ที่ได้จากการเชื่อมโยงข้อมูลนี้จะแสดงข้อมูลแถวที่เหมือนกันออกซ้ำกัน หรือเป็นการเชื่อมโยงโดยเงื่อนไขที่แสดงไม่ใช่เครื่องหมาย =

ตัวอย่างที่ 15 ให้แสดงข้อมูลของผู้ผลิตและสินค้าที่อยู่จังหวัดเดียวกัน

```
SELECT SNO, SNAME, PNO, PNAME, COLOR, PRODUCT.CITY
FROM SUPPLIER, PRODUCT
WHERE SUPPLIER.CITY = PRODUCT.CITY;
```

ผลที่ได้คือ

SNO	SNAME	PNO	PNAME	COLOR	CITY
S1	SERI	P1	PEN	RED	BANGKOK
S1	SERI	P4	BAG	RED	BANGKOK
S1	SERI	P6	TELEPHONE	RED	BANGKOK
S2	WANIDA	P2	TABLE	GREEN	RAYONG
S2	WANIDA	P5	CHAIR	BLUE	RAYONG
S3	SOMCHAI	P2	TABLE	GREEN	RAYONG

3.2.3.4.3.3. Self Join

เป็นการเชื่อมโยงข้อมูลโดยใช้รหัสเดียวกันและตั้งชื่อให้รหัสเป็นชื่ออีกชื่อรหัสหนึ่ง ทั้งนี้เพื่อประโยชน์ในการเชื่อมโยงข้อมูลใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 16 ให้แสดงรหัสสินค้าของผู้ขายที่ตั้งอยู่ที่จังหวัดเดียวกัน

```
SELECT S1.SNO, S2. SNO
FROM SUPPLIER S1, SUPPLIER S2
WHERE S1. CITY = S2.CITY
AND S1. SNO <> S2.SNO;
```

ผลที่ได้ คือ

S1.SNO	S2.SNO
S3	S2
S4	S1

3.2.3.4.3.4. Outer Join

เป็นการเชื่อมโยงข้อมูลในลักษณะที่แสดงข้อมูลที่ไม่เป็นตามเงื่อนไขที่กำหนดไว้ออกมาด้วยทั้งนี้เพื่อประโยชน์ในการดูข้อมูลที่ครบถ้วนมากขึ้น ซึ่งระบบจัดการฐานข้อมูลบางชนิดมีคำสั่ง SQL ที่สามารถทำการเชื่อมโยงเพื่อเรียกข้อมูลในลักษณะนี้ได้เช่น ORACLE ACCESS เป็นต้น

ตัวอย่างที่ 17 ให้แสดงรหัสผู้ขาย รหัสสินค้า จำนวนสินค้าที่ถูกจัดส่ง รหัส และชื่อผู้ขายที่ไม่เคยจัดส่งสินค้า โดยสมมุติให้รีเลชั่น SUPPLIER เป็นดังนี้

SUPPLIER

SNO	SNAME	CITY
S1	SERI	BANGKOK
S2	WANIDA	RAYONG
S3	SOMCHAI	RAYONG
S4	ORAPIN	BANGKOK
S5	TANACHOTE	PATUMTANE
S6	VARUMPORN	PATUMTANEE

ORDER

SNO	PNO	QTY
S1	P1	100
S1	P2	200
S2	P1	200
S2	P2	100
S3	P2	500
S4	P3	200
S5	P5	300
S5	P6	100

คำสั่ง SQL ใน ORACLE ที่ทำการเชื่อมโยงเพื่อเรียกดูข้อมูลในลักษณะนี้ จะใช้โอเปอเรเตอร์ที่เป็นเครื่องหมาย + และมีวงเล็บเปิด ปิด กำกับ (+) โดยจะแสดงต่อท้ายคอลัมน์ที่ใช้เป็นเงื่อนไขในการเชื่อมโยงรีเลชั่นที่ไม่มีข้อมูล

```
SELECT SNO, SNAME,PNO, QTY
FROM SUPPLIER S, ORDER O
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ WHERE O.SNO (+) =S.SNO ก็หาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์ที่ได้ คือ

SNO	SNAME	PNO	CITY
S1	SERI	P1	100
S1	SERI	P2	200
S2	WANIDA	P1	200
S2	WANIDA	P2	100
S3	SOMCHAI	P2	500
S4	ORAPIN	P3	200
S5	TANACHOTE	P5	300
S4	TANACHOTE	P6	100
S6	VARUMPORN		

3.2.3.4.4. การเรียกข้อมูลโดยใช้ฟังก์ชันที่เกี่ยวกับการรวม (Built-in Functions)

ฟังก์ชันที่ใช้ใน SQL ประกอบด้วย AVG (ค่าเฉลี่ย) COUNT (นับจำนวน) MAX (ค่าสูงสุด) MIN (ค่าต่ำสุด) และ SUM (ผลรวม) ฟังก์ชันทั้งห้านี้สามารถใช้คำสั่ง SELECT หรืออนุประโยค HAVING

ฟังก์ชันที่เกี่ยวกับการรวม

ฟังก์ชัน	ผลที่ได้จากการใช้ฟังก์ชัน
AVG [* / DISTINCT] <column name> หรือ AVG [* / DISTINCT] <column expression>	เป็นฟังก์ชันที่ใช้ในการหาค่าเฉลี่ยของคอลัมน์หนึ่ง ๆ ที่เก็บข้อมูลประเภทตัวเลข
COUNT [* / DISTINGT] <column name>	เป็นฟังก์ชันที่ใช้ในการนับจำนวนแถว
MAX [* / DISTINCT] <column name> หรือ MAX [* / DISTINCT] <column expression>	เป็นฟังก์ชันที่ใช้ในการคำนวณหาค่าสูงสุดของ คอลัมน์ที่กำหนดให้หาค่า
MIN [* / DISTINCT] <column name> หรือ MIN [* / DISTINCT] <column expression>	เป็นฟังก์ชันที่ใช้ในการคำนวณหาค่าต่ำสุดของ คอลัมน์ที่กำหนดให้หาค่า
SUM [* / DISTINCT] <column name>) หรือ SUM [* / DISTNCT] <column expression>	เป็นฟังก์ชันที่ใช้ในการหาค่ารวมของคอลัมน์หนึ่ง ๆ ที่เก็บข้อมูลประเภทตัวเลข

นอกเหนือจากการใช้ฟังก์ชันที่เกี่ยวกับการที่กล่าวมาข้างต้น การแสดงผลในลักษณะของการจัดกลุ่มข้อมูลยังสามารถใช้ GROUP BY เพื่อสั่งให้จัดกลุ่มตามแอททริบิวต์ที่ต้องการให้จัดกลุ่มเฉพาะลงไป เช่น การให้แสดงยอดรวมของการส่งสินค้าตามรหัสผู้ผลิตรายใดรายหนึ่ง เป็นต้น ในกรณีที่ใช้ GROUP BY การระบุชื่อแอททริบิวต์ที่จะเรียกข้อมูลออกมาจะต้องเป็นข้อมูลของแอททริบิวต์ที่ถูกระบุให้จัดกลุ่มในอนุประโยค GROUP BY เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้เงื่อนไข GROUP BY อาจจะใช้ร่วมกับอนุประโยค HAVING เพื่อให้เห็นข้อมูลที่ได้ผ่านการจัดกลุ่มโดย GROUP BY เพื่อดึงบางข้อมูลที่เป็นตามเงื่อนไขที่ระบุในอนุประโยค HAVING

ตัวอย่างที่ 18 ให้แสดงผลรวมของจำนวนที่ขายของสินค้ารหัส P2

```
SELECT SUM(QTY)
FROM ORDER
WHERE PNO = 'P2';
```

ผลที่ได้คือ

:

:

:

1200

ตัวอย่างที่ 19 ให้แสดงรหัสสินค้าที่มีผู้ขายมากกว่า 1 คน

```
SELECT PNO
FROM ORDER
GROUP BY PNO
HAVING COUNT(*) > 1;
```

ผลที่ได้ คือ

PNO

P1

P2

P3

P4

P5

3.2.3.4.5. การเรียกดูข้อมูลโดยมีข้อความย่อ

เงื่อนไขในอนุประโยค WHERE สามารถจะใช้คำสั่ง SELECT เป็นข้อความย่อ (SUBQUERY) เพื่อเรียกดูข้อมูลได้ ในกรณีที่ผลของข้อมูลเรียกจากข้อความย้อมีค่ามากกว่าหนึ่งค่า ในการระบุเงื่อนไขอาจใช้โอเปอเรเตอร์ดังต่อไปนี้ในการแสดงเงื่อนไข

3.2.3.4.5.1. โอเปอเรเตอร์ ANY

เป็นโอเปอเรเตอร์ที่ใช้ในการเปรียบเทียบค่าของแอททริบิวต์หนึ่ง ๆ ว่ามีค่าตรงกับค่าใดค่าหนึ่งของผลลัพธ์แต่ละค่าที่ได้จากข้อความย่อที่ระบุเป็นเงื่อนไขในอนุประโยค WHERE หรือไม่ และจะใช้ร่วมกับโอเปอเรเตอร์ทางตรรกะ เช่น $= > < = < >$ เป็นต้น

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปเผยแพร่ขึ้นด้านการค้า
โอเปอเรเตอร์ใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.4.5.2. โอเปอเรเตอร์ ALL

เป็นโอเปอเรเตอร์ที่ใช้ในการเปรียบเทียบค่าของแอททริบิวต์หนึ่งว่า มีค่าตรงกับทุกค่าของผลลัพธ์ที่ได้จากข้อคำถามย่อย และจะใช้ร่วมกับโอเปอเรเตอร์ทางตรรกะ เช่น = <> > = < = <> เป็นต้น

3.2.3.4.5.3. โอเปอเรเตอร์ EXISTS

เป็นโอเปอเรเตอร์ที่ใช้เพื่อให้เห็นค่าจริง (True) หรือเท็จ (False) ออกมา หากข้อคำถามย่อยในอนุประโยค WHERE มีค่าตรงกับค่าที่อ่านได้จากข้อมูลคำถามของ SELECT ที่อยู่ด้านนอก ถ้าเป็นจริงก็แสดงข้อมูลออกมา ถ้าเป็นเท็จก็จะไม่แสดงค่าของข้อมูลนั้นออกมา

ตัวอย่างที่ 20 ให้แสดงชื่อผู้ผลิตที่ขายสินค้ารหัส P2

```
SELECT SNAME
FROM SUPPLIER
WHERE SNO IN
(SELECT SNO
FROM ORDER
WHERE PNO = 'P2');
```

ผลที่ได้คือ

```
SNAME
SERI
WANIDA
SOMCHAI
ORAPIN
```

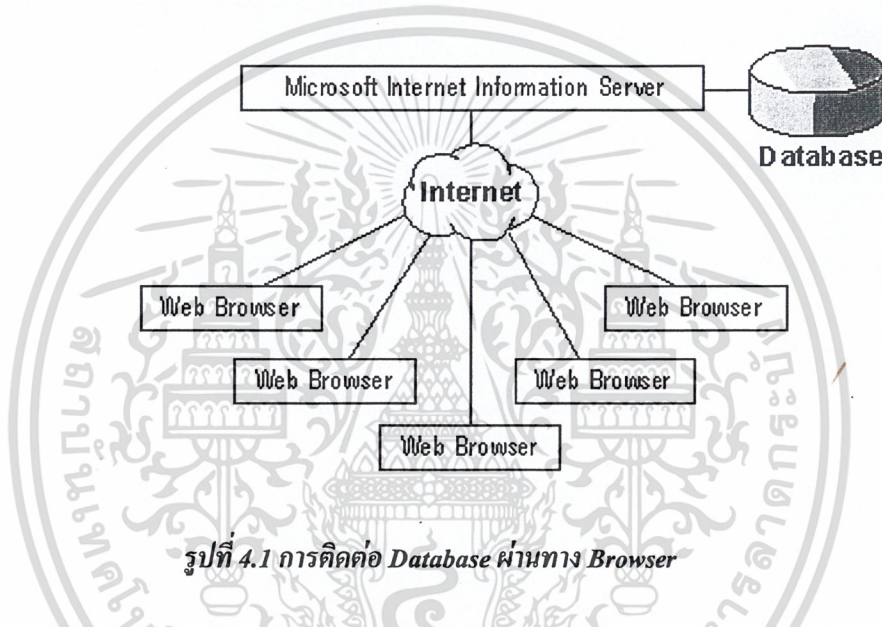
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

เทคนิคการติดต่อกับฐานข้อมูล

สำหรับแอปพลิเคชันชื่อ IIS (Internet Information Server) ของบริษัท Microsoft ที่มาพร้อมกับระบบปฏิบัติการ Microsoft Windows NT Server ที่เราใช้เป็นเว็บเซิร์ฟเวอร์ ได้เตรียมบริการสำหรับการติดต่อกับ ODBC (Open Database Connectivity) ซึ่งจะให้เราติดต่อกับระบบจัดการฐานข้อมูล หลายๆ ประเภทได้ซึ่งทำได้โดยการใช้ หลักการของ IDC (Internet Database Connector)

4.1. หลักการทำงานของ Internet Database Connector



รูปที่ 4.1 การติดต่อ Database ผ่านทาง Browser

เว็บเบราว์เซอร์ (เช่น Internet Explorer, Netscape) จะทำการส่งคำร้องขอไปยังผู้ให้บริการอินเทอร์เน็ต โดยการใช้ HTTP และเมื่อผู้ให้บริการอินเทอร์เน็ต ได้รับคำร้องขอในรูปแบบภาษาเอชทีเอ็มแอล ก็จะทำการเข้าถึงฐานข้อมูล โดยการใช้องค์ประกอบของ Internet Information Server ที่เรียกว่า Internet Database Connector (IDC) ซึ่ง Internet Database Connector จะบรรจุอยู่ใน Httpodbc.dll ซึ่งเป็น ISAPI DLL ที่ติดต่อกับ ODBC Driver เพื่อทำการเข้าถึงฐานข้อมูล

IDC จะใช้ไฟล์ 2 ชนิดสำหรับควบคุมวิธีการที่จะเข้าถึงฐานข้อมูลและวิธีการแสดงผลพท์ที่ได้ไปยังเว็บเพจซึ่งไฟล์ 2 ชนิดนี้คือ

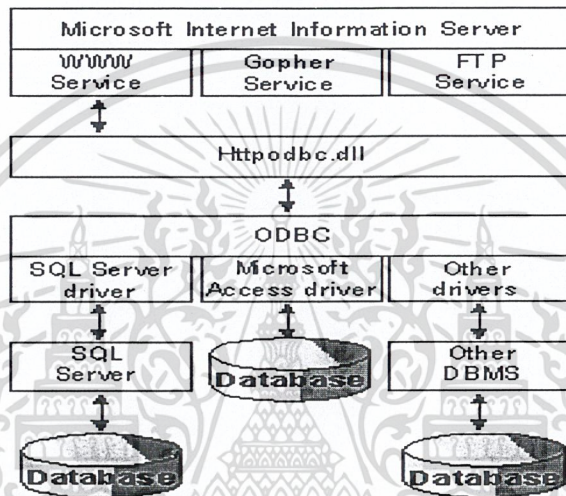
1. ไฟล์ Internet Database Connector (<ชื่อไฟล์>.idc)
2. ไฟล์ HTML extension (<ชื่อไฟล์>.htx)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2. ไฟล์ HTML extension

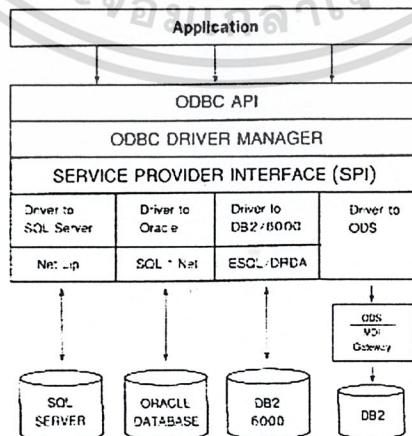
สำหรับ เอกสารเอชทีเอ็มแอล จะส่งกลับไปยังเว็บเบราว์เซอร์ หลังจากทำการรวม (merge) กับ ข้อมูลฐานข้อมูล (database information) ซึ่งกระทำโดย IDC ภาพข้างล่างเป็นการแสดงองค์ประกอบต่างๆ สำหรับการเชื่อมต่อไปยังฐานข้อมูลของ Internet Information Server.

รูปที่ 4.2 การเชื่อมต่อไปยังฐานข้อมูลของ Internet Information Server



4.2. ODBC (Open Database Connectivity)

ODBC เป็นเทคโนโลยีในการเชื่อมต่อและเรียกใช้ข้อมูลจากข้อมูลที่ต่างกันทำให้สามารถเรียกใช้ฐานข้อมูลได้หลายตระกูล ODBC เป็นผลิตภัณฑ์ของไมโครซอฟท์ซึ่งใช้สถาปัตยกรรมการเชื่อมต่อแบบ Common Interface ไมโครซอฟท์ได้พัฒนาออกแบบสถาปัตยกรรม WOSA (Windows Open Service Architecture) ซึ่งเป็นต้นแบบในการเชื่อมต่อระหว่างโปรแกรมประยุกต์ผ่าน API จนถึงฐานข้อมูลดังรูปที่ 4.3



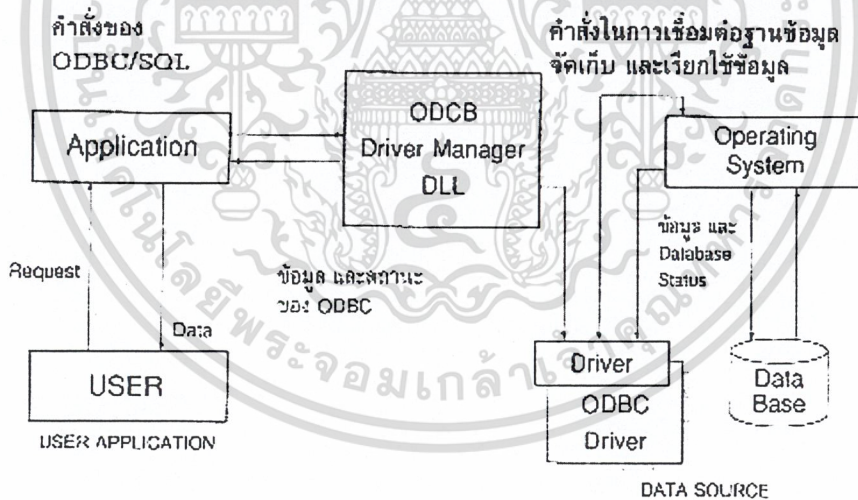
รูปที่ 4.3 แสดงการเชื่อมต่อและเรียกใช้ข้อมูลหลากหลายด้วย ODBC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.3 จะเห็นว่าโปรแกรมประยุกต์ต่าง ๆ (Application) จะติดต่อชั้นของ ODBC API ผ่านมาที่ ODBC DRIVER MANAGER ซึ่ง เป็น DLL (Dynamic Link Library) ของไมโครซอฟท์เอง มีหน้าที่ตรวจสอบความถูกต้องในการเรียกใช้ข้อมูล กำหนดการเรียกใช้ฟังก์ชัน ODBC ให้ไคร์ฟเวอร์ทั่วไป และเก็บข้อมูลกับไคร์ฟเวอร์นามสกุล DLL ทั้งหมด จากนั้นก็เชื่อมต่อผ่านมายังส่วนของ SPI (Service Provider Interface) แล้วเข้าสู่ไคร์ฟเวอร์ของฐานข้อมูลแต่ละตระกูล

ประมาณปี ค.ศ. 1989 ได้มีหน่วยงานหนึ่งก่อตั้งขึ้นซึ่งเป็นการรวมผู้ผลิตซอฟต์แวร์และฮาร์ดแวร์ที่มีชื่อเสียง เช่น DEC, HP, Ingres, Informix, ORACLE, Sun, Tandem ฯลฯ เพื่อกำหนดรูปแบบในการโอนย้ายฐานข้อมูลให้เป็นมาตรฐานชื่อหน่วยงาน SAG (SQL Access Group) ได้กำหนดรูปแบบของ SQL CLI (SQL call Level Interface) ซึ่งเป็นฟังก์ชันในการเรียกใช้ภาษา SQL ได้โดยตรงจากแอปพลิเคชันทำให้ไม่ต้องฝังส่วนภาษา SQL ลงในภาษาอื่น(Embedded-SQL) ฟังก์ชันของ SQL CLI มาตรฐานมีอยู่ 22 ฟังก์ชัน ส่วน ODBC จะมีอยู่ 51 ฟังก์ชันด้วยกันเช่น ฟังก์ชัน SQL AllocEnv SQL AllocConnect SQL Procedures

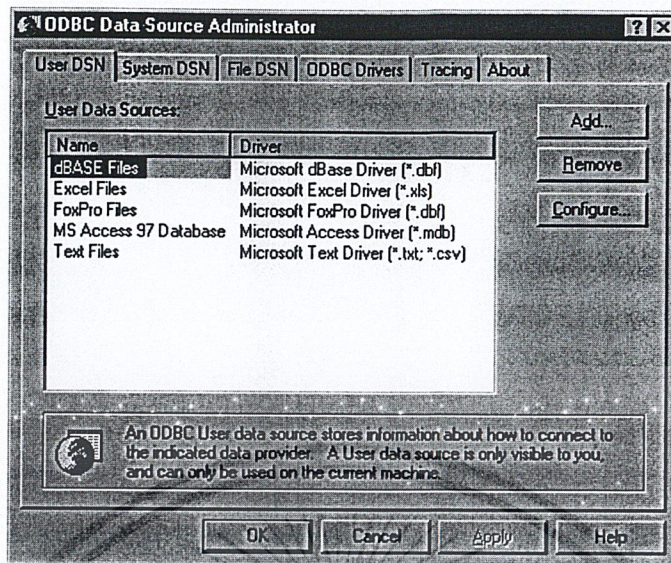
ODBC เป็นเสมือน API ที่มีหน้าที่ในการเชื่อมต่อและเรียกใช้ข้อมูลจากหลายตระกูล โดยจะใช้แอปพลิเคชันตัวเดียวกัน แต่จะใช้ไคร์ฟเวอร์ในการเชื่อมต่อต่างกัน ทำให้ไม่ต้องไปแก้ไขแอปพลิเคชันเพียงแต่เปลี่ยนไคร์ฟเวอร์ให้ตรงกับฐานข้อมูลที่จะเชื่อมต่อเท่านั้น (ปัจจุบันได้มีการพัฒนา IDAPI (Integrated Database-Application Programming Interface) จากกลุ่มบริษัท Borland, IBM, Novell เพื่อเพิ่มประสิทธิภาพในการเชื่อมต่อและเรียกใช้ข้อมูลจากหลายตระกูลนับเป็นคู่แข่งที่มาแรงของ ODBC จากไมโครซอฟท์ทีเดียว)



รูปที่ 4.4 แสดงขั้นตอนการเรียกใช้ข้อมูลผ่าน ODBC Driver Manager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3. การติดตั้ง ODBC และ การสร้าง System Data Sources



รูปที่ 4.5 แสดงการติดตั้ง ODBC

ในขณะที่ติดตั้ง IIS นั้นจะให้เลือกรูปประกอบ ต่างๆ ให้ทำการเลือก ODBC option ODBC version 2.5 องค์กรประกอบก็จะถูกติดตั้งซึ่งในรุ่นนี้ support System DSNs (Data Source Names) และสามารถถูกเรียกใช้ได้โดย Microsoft Internet Information Server ได้

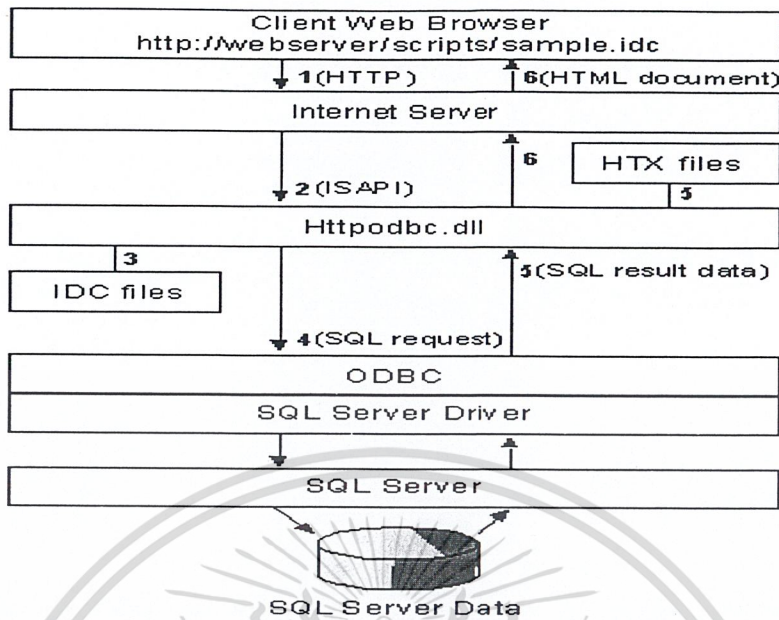
System DSN คือ ระบบที่ทำให้ ODBC ยินยอมให้ Windows NT services การใช้งานได้

4.4. การสร้าง System Data Sources

1. Double-click ที่ control Panel icon ใน Main program group ของ Program Manager.
2. Double-click ที่ ODBC icon.
3. Click ที่ ปุ่ม System DSN
4. Click ที่ ปุ่ม Add.
5. เลือก ODBC driver ใน list box และ click OK
6. ป้อน name ของ data source ชื่อนี้จะถูกนำไปอ้างอิงใน ไฟล์ .idc และกำหนด Option ต่างๆ เช่น Path และชื่อของ Source
7. กดปุ่ม OK
8. Click ที่ปุ่ม OK เพื่อที่จะ complete the ODBC และการติดตั้ง DSN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงานของ การทำงานของการ Access Database จนกระทั่งได้ผลลัพธ์



รูปที่ 4.6 แสดงขั้นตอนการทำงานของ การ Access Database จนกระทั่งได้ผลลัพธ์

สามารถอธิบายขั้นตอนการทำงานเป็นดังนี้

1. Internet Information Server ได้รับ URL จากการส่งมาของ browser
2. Internet Information Server ทำการโหลด Httpodbc.dll และ information ใน URL ที่ได้รับ ไฟล์ที่มีส่วนขยายเป็น IDC ถูกโหลดขึ้นมาโดยพิจารณาจากชื่อที่ส่งมาให้ Internet Information Server ชื่อนั้นคือชื่อไฟล์ Internet Database Connector
3. Httpodbc.dll อ่าน ไฟล์ Internet Database Connector ขึ้นมาไฟล์ Internet Database Connector ประกอบไปด้วย

Datasource : ชื่อที่กำหนดตอนสร้าง System Data Sources

Template : ชื่อไฟล์ HTML extension (. htx)

SQLStatement

/// คำสั่ง SQL ///

ตัวอย่าง

Datasource: Web SQL

Username: sa

Template: sample.htx

SQLStatement:

+ SELECT au_lname, ytd_sales

+ from pubs.dbo.titleview

+ where ytd_sales > 5000

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. IDC จะทำการ connect ไปยัง ODBC data source และ ODBC execute SQL statement ไปยัง Database ที่ได้กำหนดตอนการสร้าง System Data Sources โดยคำสั่ง SQL ไปยัง SQL Server
4. IDC นำผลลัพธ์ที่ได้จาก database ไป merge กับ ไฟล์ HTML extension
5. IDC ส่ง document ที่ได้จากการ merge กลับไปยัง Internet Information Server และส่งกลับไปยังเครื่องผู้ให้บริการ

ตัวอย่างของไฟล์ HTML extension

```

<HTML>
<BODY>
<HEAD><TITLE>Authors and YTD Sales</TITLE></HEAD>
<%if idc.sales eq ""%>
<H2>Authors with sales greater than <I>5000</I></H2>
<%else%>

<H2>Authors with sales greater than <I><%idc.sales%></I></H2>
<%endif%>

<P>
<%begindetail%>
<%if CurrentRecord EQ 0 %>

Query results:
<B>Author YTD Sales<BR></B>
<%endif%>
<%au_lname%><%ytd_sales%>
<%enddetail%>
<P>
<%if CurrentRecord EQ 0 %>
<I><B>Sorry, no authors had YTD sales greater than </I><%idc.sales%>.</B>

<P>
<%else%>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 <HR>
 ไม่วารณใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

</I>

The Web page you see here was created by merging the results of the SQL query with the template file Sample.htx.

<P>

The merge was done by the Microsoft Internet Database Connector and the results were returned to this Web browser by the Microsoft Internet Information Server.

</I>

<%endif%>

</BODY>

</HTML>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ภาษา เอชทีเอ็มแอล (HTML) เบื้องต้น

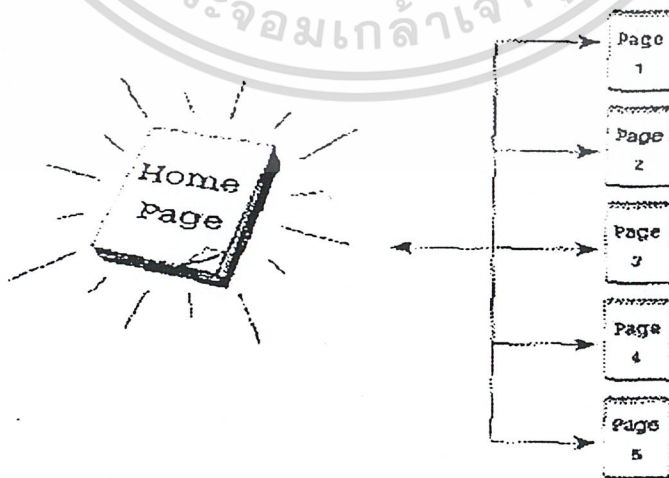
5.1. บทนำ

ปัจจุบัน เทคโนโลยีข่าวสารข้อมูล ได้มีการพัฒนาไปอย่างรวดเร็ว จากสิ่งต่าง ๆ ที่ผู้คนเคยคิดฝันในอดีต กลายเป็นความจริงได้ในโลกปัจจุบัน ในเรื่องของการสื่อสารข้อมูลที่ผู้คนเคยคิดว่า หากต้องการส่งข้อมูลจากเครื่องคอมพิวเตอร์เครื่องหนึ่งไปยังอีกเครื่องคอมพิวเตอร์อีกเครื่องหนึ่งซึ่งอยู่ห่างไกลกันคนละทวีป จะมีความเป็นไปได้มากน้อยแค่ไหน ด้วยวิธีใดที่จะใช้เวลาให้น้อยที่สุด และประหยัดค่าใช้จ่ายด้วย

ในยุคที่ถูกเรียกว่า ยุคแห่งเทคโนโลยีสารสนเทศนี้ มีระบบอินเทอร์เน็ตที่ช่วยให้ความฝันของผู้คนเป็นความจริงขึ้นมาได้ สามารถรับ-ส่งข้อมูลจากเครื่องคอมพิวเตอร์ที่อยู่ห่างกันคนละทวีปได้ในเวลาไม่กี่นาที โดยมีการกำหนดมาตรฐานต่าง ๆ ที่ทำให้เครื่องคอมพิวเตอร์ สามารถติดต่อสื่อสารกันได้ ซึ่งมาตรฐานต่าง ๆ นั้นมีองค์ประกอบด้วยกันอยู่หลายส่วน เช่น มาตรฐานของอุปกรณ์เชื่อมต่อมาตรฐานของช่องทางการสื่อสารต่าง ๆ เป็นต้น แต่ในที่นี้จะกล่าวถึง มาตรฐานของการติดต่อสื่อสารบนระบบอินเทอร์เน็ต โดยใช้เอกสาร เอชทีเอ็มแอล เป็นตัวกลาง หรือตัวแสดงผลบนจอภาพระหว่างการติดต่อสื่อสาร ให้ผู้ใช้ในระบบอินเทอร์เน็ต ใช้สื่อสารกันได้ และก่อนที่จะเข้าสู่การเรียนรู้การใช้คำสั่งต่าง ๆ อิงเอกสาร เอชทีเอ็มแอล สร้างเอกสารบนอินเทอร์เน็ต เราจะมาดูองค์ประกอบส่วนต่าง ๆ ที่เกี่ยวกับเอกสาร เอชทีเอ็มแอล และระบบอินเทอร์เน็ต

5.1.1. โฮมเพจ (Home Page)

เมื่อคุณต้องการจะไปทีใดในระบบอินเทอร์เน็ตก็แล้วแต่ คุณเพียงแต่พิมพ์ที่อยู่ที่คุณต้องการเข้าไป (กรณีที่มีการต่อเข้าระบบอินเทอร์เน็ตแล้ว และกำลังใช้โปรแกรมบราวเซอร์อยู่) แล้วรอสักพัก ก็จะได้เห็นสิ่งต่าง ๆ ปรากฏบนจอภาพ สิ่งที่คุณเห็นหน้าแรกจะเรียกว่า โฮมเพจ ซึ่งเปรียบได้กับเอกสารหน้าแรกที่ใช้ใช้ อินเทอร์เน็ตทุกคนจะเห็นเมื่อไปยังเว็บไซต์ต่าง ๆ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ **รูปที่ 5.1 แสดงลักษณะของโฮมเพจ** อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

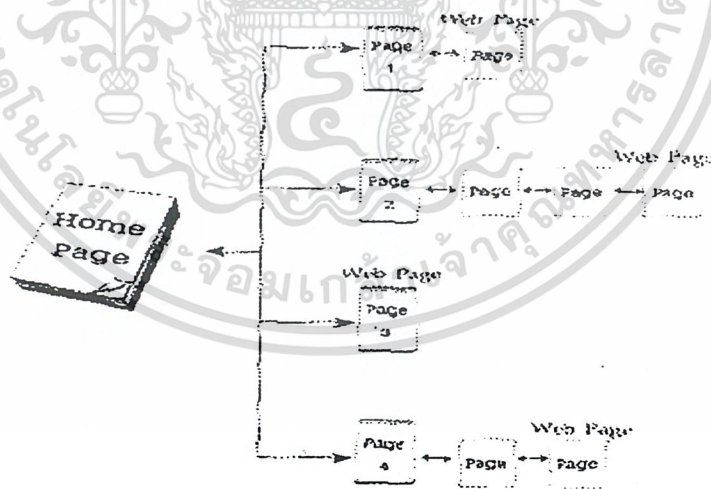
ในปัจจุบันนี้มีเว็บไซต์ต่างๆ เกิดขึ้นมากมาย แต่ละแห่งพยายามที่จะแสดงข้อมูลข่าวสาร เพื่อการโฆษณา การประชาสัมพันธ์ ฯลฯ หน่วยงาน หรือองค์กรของตนเอง ให้มีความสวยงาม แปลกตา ดึงดูดให้ผู้ใช้ อยากเข้ามาดูอีก ถึงขนาดที่ว่า มีการจัดการแข่งขัน ประกวดโฮมเพจกัน ทั้งในและต่างประเทศ เพื่อดูความสวยงาม และแนวความคิดสร้างสรรค์ผลงานที่ผู้ใช้อินเทอร์เน็ตส่วนใหญ่พัฒนาขึ้นขึ้นมา โดยมีลูกเล่นให้กับโฮมเพจของตนเองไม่ว่าในการตกแต่งด้วยรูปภาพ (Graphic), ภาพเคลื่อนไหว (Animation), สี (Color) หรือเสียง (Sound) ช่วยในการตกแต่งพื้นฉากหลัง (Background) แต่หากโฮมเพจมีลูกเล่นมากเท่าไรก็ตาม การโหลด (Load) หน้าเว็บเพจก็จะช้าลงไปด้วย

5.1.2. เว็บเพจ (Web Page)

ในการเข้าไปยังเว็บไซต์ต่าง ๆ บนอินเทอร์เน็ตนั้น เราจะได้เห็นโฮมเพจของแต่ละไซต์ ซึ่งโฮมเพจของแต่ละแห่งนั้น จะมีการสร้างจุดเชื่อมโยง หรือที่เรียกว่า ลิงค์ (Link) ไปยังที่ต่าง ๆ ได้ ไม่ว่าจะเป็น หน้าอื่นๆ ของไซต์นั้น, เว็บไซต์อื่น ๆ หรือลิงค์ไปยังบริการอื่น ๆ ในระบบอินเทอร์เน็ตได้

สำหรับเอกสารหน้าใด ๆ ในเว็บไซต์แต่ละแห่งนั้น ถูกเรียกว่า เว็บเพจ มักถูกใช้ในการกล่าวถึงหน้า (Page) ต่าง ๆ โดยรวมของเว็บไซต์ใด ๆ ซึ่งสามารถเข้าไปในแต่ละหน้านั้น โดยการลิงค์จากโฮมเพจ หรือลิงค์จากหน้าอื่นในไซต์เดียวกัน

การออกแบบเว็บเพจที่ดีนั้น ไม่ควรมีเว็บเพจมากเกินไป เนื่องจากจะทำให้ผู้ที่เข้าชม สับสนในการไป-มาระหว่างหน้ากันได้ จำนวนหน้าที่มีไม่ควรเกิน 10 หน้า หรือหากมากกว่านั้น ควรมีการวางโครงสร้างทั้งหมด เพื่อความสะดวกในการปรับปรุงข้อมูลได้อย่างทั่วถึง



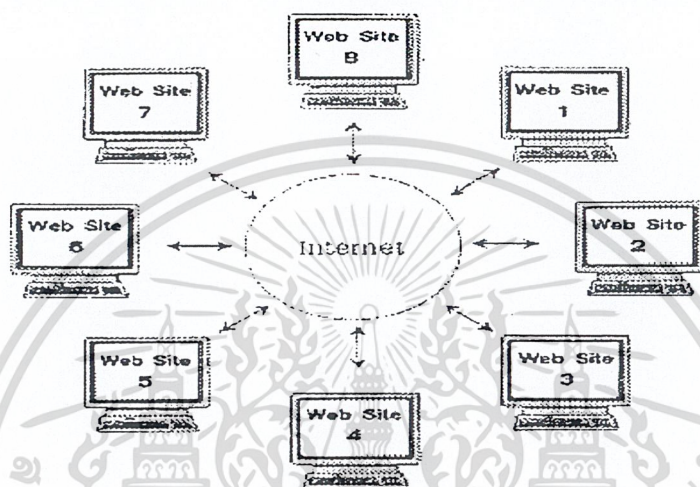
รูปที่ 5.2 แสดงลักษณะของเว็บเพจ

5.1.3. เว็บไซต์ (Web Site)

เรามักได้ยินคำพูดว่า “เว็บไซต์” หรือ “ไซต์” อยู่บ่อย ๆ เช่น “ไปไซต์นั้นมาหรือยัง?” เป็นต้นว่า “เว็บไซต์” ถูกเรียกเป็นตำแหน่งที่อยู่ของผู้ที่มีเว็บเพจของตัวเองบนระบบอินเทอร์เน็ต ซึ่งได้จากการลงทะเบียนกับผู้ให้บริการเช่าพื้นที่บนระบบอินเทอร์เน็ตเมื่อได้ลงทะเบียนในชื่อที่ต้องการแล้ว ก็สามารถจัดทำไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเพจ แล้วส่งให้ศูนย์บริการนำขึ้นไปไว้บนระบบอินเทอร์เน็ต (UpLoad) ถือได้ว่ามีเว็บไซต์เป็นของตนเองแล้ว ซึ่งคราวนี้จากที่เคยเข้าไปดูข้อมูลอื่น ก็จะให้ผู้อื่น เข้ามาดูข้อมูลของเราได้บ้าง

โดยส่วนใหญ่แล้ว หน่วยงานต่าง ๆ ทั้งภาครัฐบาลและเอกชนมักนิยมจัดทำเว็บไซต์เป็นของตัวเอง เนื่องจากสามารถประชาสัมพันธ์ข้อมูลข่าวสารได้ในวงกว้าง มีค่าใช้จ่ายในการเข้าเว็บไซต์เทียบกับการโฆษณาด้วยสื่ออื่น ๆ แล้วแตกต่างกันอย่างเห็นได้ชัดเจนนอกจากนี้ในระบบอินเทอร์เน็ต ยังมีบริการเสริมในอีกหลายด้านเช่น บริการรับฝากข้อความ (E-mail) , บริการรับส่งข้อมูลผ่านช่องทางการสื่อสาร (File Transfer Protocol : FTP) เป็นต้น จะเห็นว่า การใช้งานผ่านระบบอินเทอร์เน็ต มีประโยชน์ และคุ้มค่ามาก



รูปที่ 5.3 แสดงลักษณะของเว็บไซต์

5.1.4. อินเทอร์เน็ต (Internet)

คำว่า “อินเทอร์เน็ต” คงเป็นที่คุ้นเคยกันเป็นอย่างดี กับบุคคลทั่วไป ซึ่งบางท่านอาจจะเคยสัมผัสอย่างจริงจังก็ตาม แต่หลายท่านก็พอทราบถึงลักษณะของอินเทอร์เน็ต ว่าเป็นระบบเครือข่ายคอมพิวเตอร์รูปแบบหนึ่ง ที่สามารถเชื่อมต่อกันได้ในเครื่องคอมพิวเตอร์ต่างชนิดกัน (Platform) โดยผ่านทางสายโทรศัพท์ และอุปกรณ์แปลงสัญญาณ ที่เรียกว่า “โมเด็ม (Modem)”

จากการที่อินเทอร์เน็ต ค่อยผ่านสายโทรศัพท์ และสามารถเข้ากับเครื่องคอมพิวเตอร์ใด ๆ ได้ ทำให้การติดต่อสื่อสาร เกิดการกระจายออกเป็นวงกว้างมากขึ้นเนื่องจากปัจจุบันแทบทุกหลังคาเรือน ก็ต้องมีโทรศัพท์อย่างน้อย 1 สาย ซึ่งหากเรามีเครื่องคอมพิวเตอร์พร้อมโมเด็ม และหมายเลขสมาชิกของอินเทอร์เน็ต (Internet Account) เราก็สามารถต่อเครื่องคอมพิวเตอร์เข้าสู่ระบบอินเทอร์เน็ต ได้ทันที

ในการติดต่อสื่อสารกันบนระบบอินเทอร์เน็ตนั้น จะใช้เอกสารที่เป็นตัวแสดงข้อมูลบนจอภาพของเครื่องคอมพิวเตอร์แต่ละเครื่อง เอกสารดังกล่าว คือ เอกสาร เอชทีเอ็มแอล (Hypertext Markup Language) ที่เราจะได้ศึกษาถึงวิธีการสร้างเอกสาร เอชทีเอ็มแอล ภายในหนังสือเล่มนี้ ซึ่งการสร้างเอกสาร เอชทีเอ็มแอล ไม่ใช่เรื่องที่ยากเย็น แต่อย่างใดเลย ถึงแม้ผู้ที่ไม่มีความรู้ในการเขียน โปรแกรมมาก่อน ก็สามารถศึกษาเข้าใจได้ไม่ยาก ทำให้สามารถพัฒนาโฮมเพจของตนเอง หรือของหน่วยงานของตนขึ้นมาได้ตามความต้องการ ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.5. อินทราเน็ต (Intranet)

ในอดีตที่ผ่านมา การสื่อสารข้อมูลระหว่างเครื่องคอมพิวเตอร์ เริ่มจากระบบเครือข่ายที่เป็นแบบภายในพื้นที่หรือระบบ LAN (Local Area Network) พัฒนามาเป็นระบบเครือข่ายวงกว้าง หรือระบบ WAN (Wide Area Network) ที่ใช้ในต่างพื้นที่กัน แต่ยังคงมีข้อจำกัดในเรื่องของสายสัญญาณที่ใช้เชื่อมต่อในแต่ละจุด ยิ่งจุดเชื่อมต่อไกลมากเท่าใด ค่าใช้จ่ายในการเดินสายก็จะมากเท่านั้น จนกระทั่งมีการพัฒนามาถึงระบบอินเทอร์เน็ตที่สามารถเชื่อมต่อกันได้ทางสายโทรศัพท์ ไม่ว่าจะเป็นที่ใดก็ตาม หากมีสายโทรศัพท์ที่สามารถต่อเข้าสู่ระบบอินเทอร์เน็ตได้ นับว่าเป็น ค่าใช้จ่ายในการลงทุนที่ต่ำมากที่สุดในการติดต่อสื่อสารข้อมูล

ปัจจุบันจากระบบอินเทอร์เน็ตก็ได้มีการประยุกต์รูปแบบของอินเทอร์เน็ตมาใช้กันภายในองค์กร เรียกว่า “ระบบอินเทอร์เน็ต” คือสามารถสื่อสารกันได้เหมือนกับใช้งานในระบบอินเทอร์เน็ต เพียงแต่ใช้สื่อสารกันภายในองค์กร รูปแบบการใช้งานจะคล้ายกับการใช้งานในระบบอินเทอร์เน็ต คือ ใช้โปรแกรมบราวเซอร์ในการเข้าถึงข้อมูลต่าง ๆ ของผู้ที่อยู่ภายในเครือข่ายอินทราเน็ต บริการรับฝากข้อความ, บริการรับส่งข้อมูลผ่านช่องทางการสื่อสาร และบริการต่าง ๆ ในระบบอินเทอร์เน็ต สามารถนำมาใช้กับระบบอินทราเน็ตได้

นอกเหนือจากนี้ หากเราอยู่ในระหว่างการใช้งานในระบบอินทราเน็ต และได้มีการเชื่อมต่อเข้าสู่ระบบอินเทอร์เน็ตแล้ว จะช่วยให้เราสามารถติดต่อสื่อสารกับบุคคลได้ทั้งภายใน และภายนอกองค์กร อย่างไรก็ตาม ในระบบอินทราเน็ตขนาดไม่ใหญ่นัก ยังคงต้องพึ่งระบบแลน เป็นตัวกลางในการสื่อสารข้อมูล ถึงแม้ว่าระบบอินเทอร์เน็ตจะใช้สายสัญญาณโทรศัพท์ก็ตาม

5.1.6. เวิลด์ไวด์เว็บ (World Wide Web)

บริการต่าง ๆ ในระบบอินเทอร์เน็ตในปัจจุบันนี้ ไม่มีบริการใดที่ทันสมัย และมีผู้ใช้บริการมากกว่าบริการอื่น ๆ บนอินเทอร์เน็ต นั่นก็คือ เวิลด์ไวด์เว็บ ด้วยเหตุที่ว่า บริการนี้เป็นบริการที่ใช้งานง่าย มีการติดต่อกับผู้ใช้แบบกราฟฟิก (Graphic User Interface) สามารถแสดงรูปภาพได้ ทั้งที่เป็นภาพนิ่ง, ภาพเคลื่อนไหว และแม้กระทั่งเสียง หากเครื่องคอมพิวเตอร์ที่ใช้ มีระบบมัลติมีเดีย

ในการทำงานของ เวิลด์ไวด์เว็บ นั้น ใช้การติดต่อแบบเครือข่ายที่โยงใยกันทั่วถึงทุกเครื่องทั่วโลกที่อยู่ในระบบอินเทอร์เน็ต ลักษณะของการเชื่อมโยงกันนี้ ถูกเปรียบเทียบกับการโยงใยของแมงมุม จึงใช้คำว่าเว็บ ส่วนการเชื่อมโยงเป็นเครือข่ายขนาดใหญ่ ติดต่อกันได้ไม่จำกัดระยะทาง ก็มีการกำหนด คำว่า เวิลด์ไวด์ ขึ้นมารวมกันเป็นเวิลด์ไวด์เว็บหรืออาจเรียกสั้น ๆ ว่า เว็บ

ระบบเว็บนี้ จะมีการทำงานหลัก 2 ส่วน คือ ส่วนให้บริการ หรือเว็บเซิร์ฟเวอร์ ได้แก่เครื่องคอมพิวเตอร์ของศูนย์บริการ ที่เราสมัครเป็นสมาชิกอินเทอร์เน็ต และส่วนขอใช้บริการ (Web Client) ก็คือเครื่องที่เราใช้ติดต่อเข้าสู่ระบบอินเทอร์เน็ตซึ่งทั้ง 2 ส่วนจะติดต่อสื่อสารกันผ่านช่องทาง ที่เรียกว่าโพรโตคอล โดยใช้เอกสารไฮเปอร์เท็ก เป็นข้อมูลในการสื่อสารระหว่างกัน จึงตั้งชื่อโพรโตคอลนี้ว่า HTTP ย่อมาจาก Hyper Text Transfer Protocol สังเกตได้ว่า เวลาที่เราป้อนชื่อเว็บไซต์ที่ต้องการเข้าดูข้อมูล มักใช้เป็น HTTP นำหน้า เช่น <http://www.topspace.com> เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.17. เอกสาร เอชทีเอ็มแอล

รูปแบบไฟล์ต่าง ๆ ที่เราเห็นจอภาพในระบบอินเทอร์เน็ตนั้น ถูกจัดเก็บในรูปแบบ HTML (Hyper Text Markup Language) ซึ่งเป็นเอกสารแบบไฮเปอร์เท็กซ์ มีความสามารถในการเชื่อมโยงข้อมูลไปยังเอกสารอื่นได้ เป็นเอกสารที่มีความสามารถสูงกว่าเอกสารธรรมดาทั่วไปจากที่บอกว่าเป็นเอกสารแบบไฮเปอร์เท็กซ์นั้น ก็เนื่องจากสามารถเปิดดูข้อความภายในได้ โดยใช้โปรแกรมเอดิเตอร์ใด ๆ เปิดดูได้ ส่วนความสามารถในการเชื่อมโยงข้อมูลไปยังเอกสารอื่นนั้น ทำได้โดยการใส่สัญลักษณ์พิเศษเข้าไปในเอกสาร (Markup) หรือที่เรียกว่า แท็ก (Tag) นั่นเอง แท็ก หรือคำสั่งต่าง ๆ ถูกอ่านและกระทำตามแต่ละคำสั่งโดยโปรแกรมบราวเซอร์ต่าง ๆ เช่น Netscape Navigator, Microsoft Internet Explorer หรือ NCSA Mosaic เป็นต้น

ปัจจุบันนี้ เอชทีเอ็มแอล ถูกถือว่าเป็นภาษาคอมพิวเตอร์ ที่ใช้ในการสร้างหน้าเอกสารเพื่อแสดงบนจอภาพในระบบอินเทอร์เน็ต เพราะมีรากฐานมาจากภาษา SGML (Standard General Markup Language) ซึ่งเป็นอีกภาษาหนึ่ง ที่ใช้ในการสร้างหน้าเอกสารในระบบอินเทอร์เน็ตในระยะแรก ๆ ของการใช้งานอินเทอร์เน็ต และได้มีการพัฒนา เอชทีเอ็มแอล กันตลอดระยะเวลาที่ผ่านมา จนกระทั่งถึงเวอร์ชันล่าสุดนี้ เป็นเวอร์ชัน 3.2 แล้ว สามารถกำหนดแท็กคำสั่งรองรับบริการต่าง ๆ บนระบบอินเทอร์เน็ตได้มาก ไม่ว่าจะเป็น FTP, TELNET,

E-mail เป็นต้น ทั้งยังสามารถที่จะเชื่อมโยงข้อมูลกับรูปภาพ, เสียง หรือภาพเคลื่อนไหวได้อีกด้วย

ในภาษา เอชทีเอ็มแอล นี้ ยังมีข้อดีกว่าภาษาคอมพิวเตอร์อื่น ๆ คือ เมื่อเราสร้างงานเสร็จแล้ว สามารถนำไปเรียกดูผลการทำงานด้วยโปรแกรมบราวเซอร์ได้เลย โดยไม่ต้องผ่านการคอมไพล์ ก่อนและโอกาสที่จะเกิดข้อผิดพลาดจากการเขียนคำสั่งก็ค่อนข้างน้อยมาก เพราะหากคำสั่งใดที่พิมพ์ผิด เมื่อโปรแกรมบราวเซอร์อ่านแล้วไม่รู้จักก็จะไม่สนใจคำสั่งนั้นจะแสดงผลเอกสารตามปกติ

สำหรับการใช้งานเอกสาร เอชทีเอ็มแอล นี้จะมีการทำงานเปรียบได้กับการทำงานของโปรแกรม Help บน Windows คือ เวลาที่เราเลื่อนตัวชี้เมาส์ไปยังคำที่มีสีเขียว และขีดเส้นใต้ ตัวชี้เมาส์จะเปลี่ยนจากรูปลูกศรเป็นรูปมือ ในจุดนี้เอง คือ จุดที่มีการเชื่อมโยงข้อมูลไปยังเอกสารอื่น เช่นเดียวกับการเชื่อมโยงข้อมูลในเอกสาร เอชทีเอ็มแอล เราเรียกจุดที่มีการเชื่อมโยงข้อมูลนี้ว่า ลิงค์ ลิงค์ของเอกสารไฮเปอร์เท็กซ์ คือ จุดที่ใช้เชื่อมโยงข้อมูลไปยังเอกสารอื่น อาจเป็นเอกสาร เอชทีเอ็มแอล อื่น, ไฟล์รูปภาพทั้งภาพนิ่ง และภาพเคลื่อนไหว, ไฟล์เสียง หรือแม้กระทั่งเว็บไซต์ใด ๆ ในระบบอินเทอร์เน็ต นอกจากนี้หากมีการติดต่อกับระบบอินเทอร์เน็ตอยู่ เราสามารถลิงค์ไปยังบริการต่าง ๆ ในระบบอินเทอร์เน็ตได้

5.2. เอชทีเอ็มแอล เบื้องต้น

5.2.1. รูปแบบของ เอชทีเอ็มแอล

มีศัพท์ 2 คำที่ควรทราบเกี่ยวกับการสร้างเว็บเพจ ด้วย เอชทีเอ็มแอล ได้แก่

Text หมายถึง ข้อความทั่ว ๆ ไป

Tag หมายถึง คำสั่งที่ใช้ในการกำหนดแบบของ Text โดยคำสั่งของ เอชทีเอ็มแอล จะอยู่ในเครื่องหมาย < และ >

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับใช้เรียนเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นใจเสียใจขอสงวนลิขสิทธิ์ไว้
แม้ว่าใครเห็นได้ ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This is a Bold word

ตัวอักษร a, b, c,.....คือ text

 และ คือ Tag โดยทั่วไปจะมี Tag เริ่มต้นและ Tag จบท้ายโดยจะใช้ตัวเล็กหรือตัวใหญ่ ก็มีผลเหมือนกัน จากตัวอย่างหมายถึงการทำข้อความ Bold ให้เป็นตัวเข้ม ผลลัพธ์ที่ได้คือ

This is a **bold** word

5.2.2. กฎในการใช้ เอชทีเอ็มแอล

- แต่ละ Tag จะต้องอยู่ในเครื่องหมาย < > เท่านั้น
- จะใช้ตัวพิมพ์เล็กหรือใหญ่ก็ได้ มีความหมายเดียวกัน เช่น <title>, <Title> หรือ <TTITLE> ต่างก็ให้ความหมายเดียวกัน
- แทบทุก Tag จะมี Tag เปิด และ Tag ปิด โดย Tag ปิด จะมียุบรวมอยู่ด้วย แต่ก็มีบาง Tag ที่ไม่มีตัวปิด เช่น
, <P> เป็นต้น
- ในการพิมพ์เอกสาร เอชทีเอ็มแอล นั้นจะพิมพ์ติดต่อกันทั้งข้อความและคำสั่งหรือพิมพ์แยกบรรทัดหรือเคาะช่องว่างก็แล้วแต่ผู้พิมพ์ แต่การพิมพ์ให้ ดูสวยงามจะทำให้ง่ายต่อการอ่านและแก้ไขในภายหลัง

5.2.3. การทำงานของ เอชทีเอ็มแอล กับ Browser

ปกติถ้าเรานำข้อมูลที่สร้างจากโปรแกรมทั่ว ๆ ไปที่ไม่ใช่ เอชทีเอ็มแอล มาแสดงในเว็บเบราว์เซอร์ ผลลัพธ์ที่ได้จะไม่ถูกต้องตามเอกสารต้นฉบับ เนื่องจากเบราว์เซอร์ ไม่เข้าใจในโครงสร้างของข้อมูล แต่ก็แสดงผลให้ โดยนำข้อความทั้งหมดมาต่อกันคงรูป

ตัวอย่างที่ 5.1

Dve's Desk
West Lafayette,Indiana
3 November, 1996

Dear Reader,

Thank you for connection to my Web server ,but I reget to tell you
that things aren't up and running yet!. They will be_soon_,but they aren't today.

Sincerely,

Dave Tayor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลที่ได้ คือ

Dve's Desk West Lafayette, Indiana 3 November, 1996 Dear Reader, Thank you for connecting to my web server', but I regret to tell you that aren't up and running yetl. They will be _soon_,but they aren'to day. Sincerely, Dave Tayor.

บราวเซอร์ เป็นตัวแปลภาษา เอชทีเอ็มแอล ที่แปลได้อย่างตรงไปตรงง ถ้าโครงสร้างผิดๆ ก็จะแสดงแบบผิดๆ ถ้าถูกก็จะแสดงผลให้ตรงตามที่ต้องการ ไม่มีการบอกว่าผิดพลาด (Error) ที่ตำแหน่งใด เหมือนกับตัวแปลภาษาอื่นๆ แต่จะแสดงผลลัพธ์ให้เห็นเลยไม่ว่าจะผิดหรือถูก ดังนั้นผู้เขียนจึงจำเป็นต้องค้นหาส่วนที่ผิดให้ได้

5.2.4. การจัดรูปแบบให้เป็นย่อหน้า (Paragraph) และเป็นบรรทัด

จะใช้ <HTML>วางไว้ที่ต้นเอกสารและ </HTML>ไว้ที่ท้ายเอกสารเพื่อให้บราวเซอร์ทราบว่าเป็นเอกสาร เอชทีเอ็มแอล

จะใช้ <P> เป็นตัวจัดให้ข้อความที่ตามหลังมาขึ้นย่อหน้าใหม่เพื่อความเข้าใจ ไม่มีคำสั่งปิดท้าย

จะใช้
 เพื่อให้ข้อความขึ้นบรรทัดใหม่

ตัวอย่างที่ 5.2

<HTML>

Dve's Desk

3 November, 1996

<P>

Dear Reader,

<P>

Thank you for connecting to my Web server ,but I regret to tell you that things aren't up and running yet!

They will be _soon_,but they aren't today.

<P>

Sincerely,

<P>

Dave Tayor

</HTML>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลที่ได้ คือ

Dve's Desk West Lafayette, Indiana 3 November, 1996

Dear Reader,

Thank you for connecting to my Web server ,but I regret tell you that things aren't up and running yet! They will be _soon_,but they aren't today.

Sincerely,

Dave Tayor

ไม่มีคำสั่งปิดท้าย ถึงแม้จะเว้นบรรทัดว่างไว้ให้ในเอดิเตอร์บราวเซอร์ ก็จะไม่เว้นให้ เพราะไม่เข้าใจ
ตัวอย่างที่ 5.3

<HTML>

Dve's Desk

West Lafayette,Indiana

3 November, 1996

<P>

Dear Reaber,

<P>

Thank you for connecting to my Web server ,but I regret to tell you that things aren't up and running yet! they will be _soon_,but they aren't today.

<P>

Sincerely,

<P>

Dave Tayor

</HTML>

ผลที่ได้ คือ

Dve's Desk

West Lafayette, Indiana

3 November, 1996

Dear Reader.

Thank you for connecting to my Web server ,but I regret to tell you that things aren't up and running yet! They will be _soon_,but they aren't today.

Sincerely.

Dave Tayor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
จะต้องใส่
 ลงไป เพื่อจัดให้ข้อความขึ้น
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.5. การจัดรูปแบบข้อความให้ตรงตามเอกสารเดิม

ในบางครั้งเราจะได้ข้อมูลที่โอนมาจากโปรแกรมอื่น เช่น Excel ซึ่งจัดรูปแบบไว้เรียบร้อยแล้วจึงเป็นการเสียเวลาถ้าจะต้องมาใช้คำสั่ง เอชทีเอ็มแอล จัดซ้ำใหม่ หรือบางครั้งเราอาจต้องการจัดให้เสร็จเรียบร้อยใน Editor เลยก็ได้ ลักษณะนี้จะใช้คำสั่ง <PRE และ </PRE> มาช่วย โดยนำ <PRE> วางไว้หน้าข้อมูลและ </PRE> ไว้ท้ายข้อมูล

ตัวอย่างที่ 5.4

```
<PRE> Once upon a time I was a young man,
      but then I
      ate some walnuts
      Sold some pretzels
      and juggled kittens. Or not.. (/PRE>
```

ผลที่ได้ คือ

```
once upon a time I was a young man,
      but then I
      ate some walnuts
      Sold some pretzels
      and juggled kittens. Or not.
```

5.2.6. การแบ่งเอกสารออกเป็น ส่วน ๆ (Section)

กรณีทีเอกสาร เอชทีเอ็มแอล ที่สร้างมามีขนาดใหญ่ เราอาจแบ่งเอกสารเป็นส่วน ๆ เพื่อให้การค้นหาหรือแก้ไขทำได้ง่ายขึ้น (เอกสารขนาดเล็ก ก็นิยมทำเช่นเดียวกัน) การใส่ชื่อเรื่อง และขอบเขตของเนื้อหาลงไป

```
<HEAD>, </HEAD> ใช้กำหนดชื่อเรื่องให้กับส่วนเอกสาร แต่จะไม่แสดงให้เห็นในบราวเซอร์
<BODY>, </BODY> ใช้กำหนดขอบเขตของเนื้อหาในเอกสารส่วนนั้น (เนื้อหาที่จะแสดงใน
บราวเซอร์
```

ตัวอย่างที่ 5.5

```
<HTML>
<HEAD></HEAD>
<BODY>
Dve's Desk <BR>
```

```
West Lafayette, Indiana <BR>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 <P> ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dear Reader,

<P>

Thank you for connecting to my Web server ,but I reger to tell you
that things aren't up and running yet!

They will be _soon_,but they aren't today.

<P>

Sincerely,

<P>

Dave Tayor

</BODY>

</HTML>

เป็นการกำหนดว่าเอกสาร เอชทีเอ็มแอล นี้ชื่อว่าอะไร (Head) และมีขอบเขตของเนื้อหาที่จะแสดงใน
Browser อยู่ระหว่างช่วงใด (Body)

5.2.7. การกำหนดชื่อเอกสารให้แสดงบนบราวเซอร์

เราสามารถแสดงเนื้อหาเรื่องไว้ในบราวเซอร์ได้ด้วยคำสั่ง <TITLE> และ </TITLE> โดยจะต้องวาง
ไว้ระหว่างคำสั่ง <HEAD> และ </HEAD> เสมอ

ตัวอย่างที่ 5.6

```
<HTML>
```

```
<HEAD>
```

```
<TITLE> This is the tile </TITLE>
```

```
</HEAD>
```

```
</HTML>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.8. การกำหนดขนาดของตัวอักษรในแต่ละเซกชัน (Headline)

กรณีทีเอกสารถูกแบ่งออกเป็นหลาย ๆ ส่วน และเราต้องการกำหนดขนาดของตัวอักษรในแต่ละ Section ให้แตกต่างกันออกไป ก็ทำได้ง่าย ๆ โดยใช้คำสั่ง <Hn> และ </Hn> โดย n แทนตัวเลข 1 ถึง 6 6 คือ ขนาดเล็กสุด และ 1 คือ ขนาดใหญ่สุด แก้ไขขนาดเองไม่ได้ เนื่องจากถูกกำหนดมาไว้แล้วโดยผู้ออกแบบข้อความที่กำหนดด้วยเฮดไลน์ จะเริ่มต้นที่บรรทัดใหม่ทุกครั้ง

ตัวอย่างที่ 7

```
<H1> Monday </H1>
<H2> Tuesday </H2>
<H3> Wednesday </H3>
<H4> Thursday </H4>
<H5> Friday </H5>
<H6> Saturday </H6>
```

ผลที่ได้ คือ

Monday

Tuesday

Wednesday

Thursday

Friday

Saturday

5.2.9. การกำหนด Footer

เป็นข้อความที่นิยมใส่ไว้ด้านล่างเอกสาร เช่น ที่อยู่ที่เราจะติดต่อได้ ใช้คำสั่ง <ADDRESS> และ </ADDRESS> ข้อความที่อยู่ระหว่าง Tags นี้จะถูกกำหนดให้เป็นตัวเอน (Italic) โดยอัตโนมัติ

5.3. การนำเว็บเพจที่เราสร้างใส่ในอินเทอร์เน็ต

5.3.1. การวางแผนเกี่ยวกับเว็บเพจ

ในการวางแผนเกี่ยวกับเว็บเพจ มีข้อที่ควรพิจารณาหลายอย่างด้วยกัน เช่น

5.3.1.1. สถานที่ติดตั้งเว็บเพจ

ผู้ใช้อินเทอร์เน็ตมีอยู่ด้วยกันหลายระดับ ผู้ที่มีเซิร์ฟเวอร์เป็นของตนเองคงจะไม่มีปัญหาในเรื่องสถานที่ติดตั้งเว็บเพจ แต่สำหรับผู้ใช้อินเทอร์เน็ตโดยทั่วไปที่เชื่อมต่อกับอินเทอร์เน็ต มีพื้นที่ที่ได้รับการจัดสรรจำกัด ใช้งานโดยธรรมดาก็ค่อนข้างจะไม่เพียงพออยู่แล้ว ถ้าสร้าง Web Page แล้วนำไปเก็บไว้ในพื้นที่นี้ก็จะทำงานอย่างอื่นได้ยาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประการสำคัญอีกอย่างหนึ่งที่ควรพิจารณาคือ ศูนย์บริการอินเทอร์เน็ตที่เราเป็นสมาชิกอยู่นั้นเป็นแบบที่อนุญาตให้บุคคลภายนอกเข้ามาใช้ข้อมูลได้หรือไม่ (เช่น Anonymous FTP) ถ้าไม่อนุญาตก็คงไม่มีผู้เห็นโฮมเพจของเราได้

5.3.1.2. การเก็บเอกสารในแต่ละวิธี

วิธีที่ใช้เก็บเอกสาร เอชทีเอ็มแอล มีอยู่ด้วยกัน 2 วิธีคือ

5.3.1.2.1. FTP Anonymous วิธีนี้ถ้าเซิร์ฟเวอร์เป็น WWW เซิร์ฟเวอร์แล้วก็ไม่จำเป็นต้องใช้ เนื่องจากว่าเว็บเบราว์เซอร์ต่างๆ ที่มีอยู่สามารถทำการโอนย้ายข้อมูลด้วย FTP ได้อยู่แล้ว การดึงข้อมูลเอกสารประเภทนี้ใช้การกำหนดที่อยู่ด้วย ftp:// หรือ file:// (การเก็บเอกสารของวิธีนี้เป็นแบบ File)

5.3.1.2.2. HTTP วิธีนี้เอกสาร เอชทีเอ็มแอล จะถูกเก็บบนเซิร์ฟเวอร์ การดึงข้อมูลและการส่งเอกสารบนเซิร์ฟเวอร์ เหล่านี้ ผู้ใช้จะแสดงความต้องการด้วยการกำหนด URL เป็น http://

ความแตกต่างของทั้งสองวิธีก็คือ การดึงข้อมูลเอกสารต่าง ๆ ในวิธีของ FTP Anonymous นั้นไม่ว่าจะเป็นแท็กเสียง รูปภาพ และการเชื่อมโยงไปยังเอกสารอยู่ในระดับที่น่าพอใจ แต่จะไม่สามารถเข้าถึงข้อมูลฟอรัมหรือระบบที่เป็นการโต้ตอบ (Interactive) ได้

ส่วนวิธี HTTP ตัวเว็บเบราว์เซอร์ เช่น Netscape, Mosaic หรืออื่น ๆ สามารถเข้าถึงข้อมูลของ FTP Anonymous ได้โดยเชื่อมโยงไฟล์ข้อมูลกับเอกสาร เอชทีเอ็มแอล จัดการประมวลผลฟอรัมและทำงานกับเอกสารที่เป็นอินเทอร์เน็ตแอ็กทีฟ (Interactive) ได้

หมายเหตุ “ใช้แท็กข้ออย่างเดี่ยวหรือกราฟฟิก”

สิ่งหนึ่งที่ต้องคำนึงถึงก็คือเบราว์เซอร์บางตัว เช่น Lynx จะแสดงข้อมูลได้เฉพาะแบบแท็กเท่านั้น หรือผู้ใช้ที่สมัครสมาชิกประเภทที่เป็นแท็กจะไม่สามารถเห็นหรือเข้าถึงเอกสารที่รูปภาพได้ การออกแบบเว็บเพจที่ดีจึงจะเน้นหนักอยู่ในส่วนที่แสดงผลทางหน้าจอมากกว่าส่วนอื่น ๆ

5.3.2. ลักษณะของเว็บเพจที่ดี

เว็บเพจเป็นการแสดงข้อมูลที่สามารถมองเห็นได้ โดยที่ผู้เข้ามาดูไม่จำเป็นต้องรู้เกี่ยวกับการจัดการที่เป็นตัวกำหนดการทำงาน (Logical Way) หรือการจัดการทางฮาร์ดแวร์ ดังนั้นการสร้างเว็บเพจ ที่ดีจึงจะเน้นหนักอยู่ในส่วนที่แสดงผลทางหน้าจอมากกว่าส่วนอื่น

5.3.3. ข้อเสนอแนะในการสร้างเว็บเพจที่ดี

- ทำให้มีข้อมูลที่ใช้ประโยชน์ได้มากแต่ได้มากแต่ไม่แน่นจนเกินไป จัดที่วางให้เหมาะสมแต่ละย่อหน้าไม่ ควรใกล้กันหรือห่างจนเกินไป

- ให้อพยายามแสดงข้อมูล โดยทำเป็นตารางหรือรายการที่สามารถกำหนดหรือเลือกใช้ได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ไม่สร้างเว็บเพจที่มีลักษณะภาพอยู่ในภาพ ให้ใช้พื้นที่ว่างให้เป็นประโยชน์ เพราะว่ามันหน้าจอมืดที่ว่างมากพอจึงควรใช้มากกว่าที่จะประหยัดเนื้อที่จนไม่น่าดู
- พยายามแสดงข้อมูลแต่ละส่วนให้มีรูปแบบคล้ายกัน แต่ละย่อหน้าไม่ควรมีความยาวมากเกินไป หรือถ้ายาวมากก็ให้แบ่งมาย่อหน้าใหม่
- ถ้าเอกสารยาวมากควรใช้ลิงค์เข้ามาช่วยโดยแบ่งเอกสารไปสร้างเพจใหม่ที่มีลิงค์ เชื่อมโยงไปหาได้
- ใช้รูปภาพ หรือลักษณะทางกราฟฟิกเข้ามาช่วยเพิ่มความน่าสนใจ
- ข้อความที่เป็นหัวเรื่องหรือลิงค์ ควรเป็นคำหรือวลีที่น่าสนใจ แต่ต้องไม่เกินความจริงเพราะจะมีผลเสียได้ ภายหลัง

5.3.4. หาตัวอย่างเว็บเพจดี ๆ ได้จากไหน

มีคณกรู่มหนึ่งที่จะตอบตรวจสอบว่า มีเว็บเพจใดที่ดีที่สุด (โดยทั้งไปจะเรียกว่า “Cool Web Page” คือดูแล้วสบายใจ) และเราสามารถศึกษาการใช้งาน และออกแบบจากไซต์ เหล่านี้ได้ไซต์ ต่าง ๆ เหล่านี้จะมี การจัดอันดับ URL ดี ๆ ไว้ให้ เช่น

- * <http://home.netscape.com/home/whats-cool.html> เป็นบริการจาก Netscape
- * http://www.yahoo.com/Entertainmebt/Cool_Links เป็นบริการจาก Yahoo
- * <http://www.pointcom.com> เป็นบริการ Pointcom ที่จะแสดง Web Site ที่ได้คะแนนสูงสุด 5 อันดับแรกมาให้

5.3.5. ข้อกำหนดเกี่ยวกับชื่อของ URL

จากที่ทราบกันแล้วว่า URL เป็นเครื่องมือที่เว็ลด์ไวด์เว็บใช้ค้นหาเพจ ที่กำหนดไม่ว่าจะเป็นภาพหรือเสียงก็ตามแบ่งได้เป็น 2 ประเภทคือ

- * **Absolute URL** : ใช้กำหนดที่อยู่ที่มีสมบูรณ์ไม่ต้องการข้อมูลอื่นเพิ่มเติมอีก
- * **Relative URL** : ใช้ที่อยู่ปัจจุบันมาอ้างอิงแล้วเพิ่มข้อมูลที่อยู่ที่ต้องการมาประกอบกัน

เอชทีเอ็มแอล ก็ใช้ URL เป็นเครื่องมือในการกำหนดเอกสารที่ค้นหาเช่นเดียวกัน

รูปแบบ

Service://Host:Port/Part/File.Ext

file:// ใช้ FTP ดึงไฟล์ข้อมูล	gopher:// ใช้เครื่องมือ Gopher
fip:// ใช้ FTP ดึงไฟล์ข้อมูลเช่นเดียวกับ File://	telnet:// ใช้ Telnet เข้าถึงเครื่องคอมพิวเตอร์ควบคุม
http:// ใช้ WWW เซอร์ฟเวอร์ดึงไฟล์ข้อมูล	news. อ่านข่าวจากเครื่องคอมพิวเตอร์ควบคุม

เอกสารนี้เป็น **หมายเหตุ** News เป็นบริการที่ไม่ต้องใช้ // ในการกำหนดที่อยู่ URL นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 5.8 ถ้าต้องการกำหนดให้เข้าถึงไฟล์ P1.gif ซึ่งอยู่ในไดเรกทอรี CW บนเครื่องคอมพิวเตอร์ชื่อ www.iguest ด้วย เลขที่เอ็มแอล จะทำได้ดังนี้

ABSOLUTE : < Img Src = "http://www.iguest.net/cw/P1.gif">

RELATIVE : < Img Src = "P1.gif">

5.3.6. REGISTER

Register ก็คือการนำเว็บเพจของเราไปใส่ในเว็บไซด์ที่เหมาะสม การหาไซด์ ที่เหมาะสมนั้น ขึ้นอยู่กับวิจารณ์ญาณของเราเองด้วย แต่วิธีการง่าย ๆ ก็คือ การค้นหาจากเพจอื่น ๆ ที่มีอยู่แล้วว่าเพจ ที่มีเนื้อหาคคล้าย ๆ กับเราอยู่ในไซด์ ไตบ้าง เราก็นำเพจ ของเราใส่เอาไว้บ้าง เพราะคนที่ต้องการค้นหาข้อมูลอยู่ที่คล้าย ๆ กับเราก็มักจะใช้วิธีการค้นหาเช่นเดียวกับเรา การค้นหาจะใช้เครื่องมือที่เรียกว่า "Search Engine" ซึ่งมีอยู่หลายตัว เช่น Yahoo, Magellan, ฯลฯ เป็นต้น เช่น เพจของเราเป็นเรื่องเกี่ยวกับบ้าน เราอาจ Search หาคำว่า "House, Home, Furniture" เมื่อหาพบจะแสดงรายชื่อ URL ที่มีค่าเหล่านี้อยู่จากนั้นจึงเลือกไซด์ ที่ต้องการ Register ได้

5.3.7. เว็บไซต์ที่เราจะ Register ได้ (แบบฟรี) มี 3 ชนิดคือ

1. Directory Site เป็นไซด์ที่เราต้องการกรอกรบข้อมูล, Keyword ต่าง ๆ ลงไปเอง แล้วไซด์ เหล่านี้จะนำข้อมูลที่เรกรอกไปใช้ในการค้นหา เช่น Yahoo และ Trade Wave Galaxy
2. Reboot Site หรือ Crawler System เป็นไซด์ ที่จะค้นหาข้อมูลจากเว็บเพจของเราเอง ดังนั้นข้อมูลจริงมีอะไรก็จะใช้ข้อมูลเหล่านั้นในการค้นหา เช่น Alta Vista, Web Crawler และ Lycos
3. Announcement Service จะนำหน้าที่ประกาศโฆษณาเว็บเพจ ของเราไปให้กับ Search System อื่น ๆ มากมายมีทั้งแบบฟรีและเสียเงิน) ได้แก่ Submit IT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การติดตั้งและบริหารเว็บเซิร์ฟเวอร์

6.1. โครงสร้างและองค์ประกอบของระบบปฏิบัติการวินโดวส์เอ็นทีเซิร์ฟเวอร์

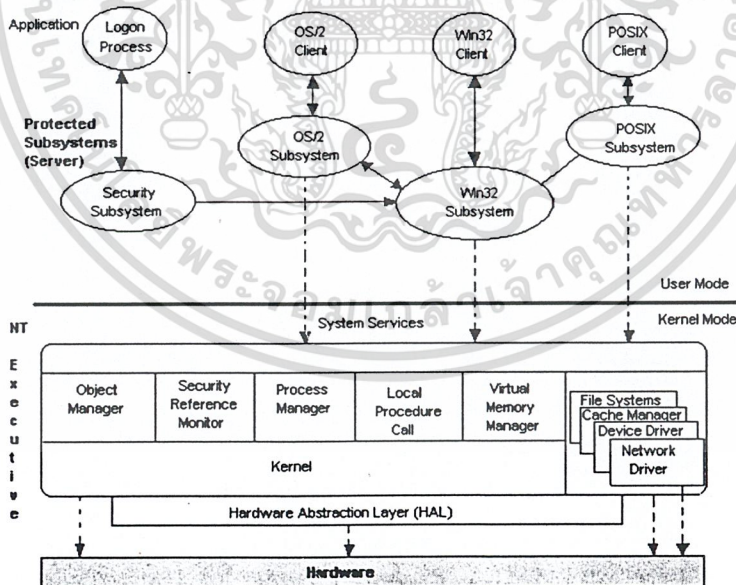
วินโดวส์เอ็นทีจัดเป็นระบบปฏิบัติการแบบเครือข่ายที่มีประสิทธิภาพสูง ราคาถูก จัดการง่ายทำให้หน่วยงานต่าง ๆ นำเอาไปใช้อย่างแพร่หลาย วินโดวส์เอ็นทีถูกออกแบบมาโดยมีคุณสมบัติเด่นดังนี้

- * **Extensibility** สามารถรองรับการขยายขีดความสามารถได้ตามความต้องการที่เพิ่มขึ้นของผู้ใช้
- * **Portability** สามารถนำไปดัดแปลงแก้ไขเพื่อให้สามารถทำงานบนซีพียูอื่นได้ง่าย
- * **Reliability and Robustness** สามารถป้องกันหรือเตือนความผิดปกติในการทำงานที่เกิดขึ้นทั้งสาเหตุที่เกิดสิ่งภายใน และภายนอก รวมถึงการตรวจสอบระบบได้ตลอดเวลา

* **Compatibility** ระบบการติดต่อกับผู้ใช้ (Interface) และ API (Application Program Interface) ทำงานเข้ากันได้กับโปรแกรมต่าง ๆ ของไมโครซอฟต์ที่มีอยู่เดิมได้

- * **Performance** มีความเร็วในการทำงานสูงและตอบสนองผู้ใช้เร็ว ถึงแม้ว่าจะมีข้อจำกัดต่าง ๆ

โครงสร้างของระบบปฏิบัติการวินโดวส์เอ็นที แบ่งออกเป็น 2 ส่วนใหญ่ ๆ คือ User Mode กับ Kernel Mode ดังรูปที่ 6.1



รูปที่ 6.1 โครงสร้างของระบบปฏิบัติการวินโดวส์เอ็นที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ระบบย่อย (Sub system) ใน Kernel Mode สามารถเข้าใช้ทรัพยากรต่าง ๆ ได้ไม่จำกัด ในขณะที่ระบบย่อยใน User Mode จะเข้าถึงทรัพยากรได้จำกัด

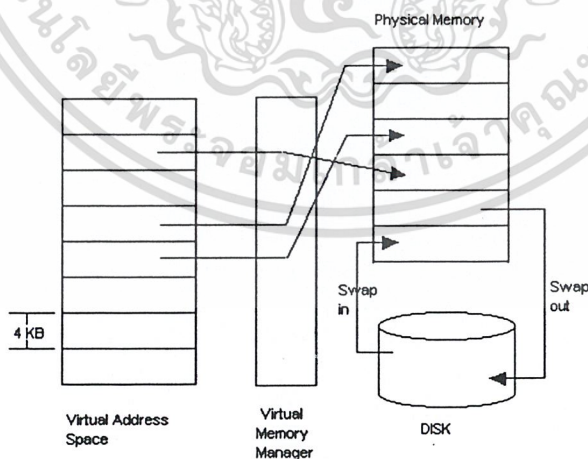
Hardware Abstraction Layer หรือ HAL เป็นสิ่งที่เชื่อมระบบปฏิบัติการเข้ากับ ฮาร์ดแวร์ของเครื่องคอมพิวเตอร์อีกทีหนึ่ง ทำให้วินโดวส์เอ็นทีสามารถทำงานได้บนเครื่องคอมพิวเตอร์หลายชนิดได้แก่ อินเทล 486, อินเทลเพนเทียม, RISC (Reduced Instruction Set Computer) หรือ MIPS

Object Manager เป็นระบบย่อยที่จัดการเกี่ยวกับอ็อบเจกต์ต่าง ๆ เนื่องจากวินโดวส์เอ็นทีจะมองทรัพยากรต่าง ๆ ในระบบเป็นอ็อบเจกต์ การสร้าง,การใช้,ทำลาย จะผ่านระบบย่อยส่วนนี้ตัวอย่างของอ็อบเจกต์ที่เด่นชัด ได้แก่ ไคลเร็กทอรี, แฟ้มข้อมูล, โพรเซส (Process), เทรด (Thread)

Security Reference Monitor หรือ SRM ใช้ในการรักษาความปลอดภัยภายในระบบโดยมีระดับความปลอดภัยเท่ากับ ซีทู(C2) ตามมาตรฐานความปลอดภัยของสหรัฐอเมริกา วิธีการของระบบความปลอดภัยแบบนี้จะทำการกำหนดให้ทรัพยากรต่าง ๆ ในระบบจะมีข้อมูลเกี่ยวกับความปลอดภัยซึ่งบอกถึงการอนุญาตให้ใช้งานบอกถึงการอนุญาตให้ใช้งาน ได้แก่ผู้ใช้ต่าง ๆ ข้อมูลเหล่านี้เรียกว่า Access Control List(ACL) ก่อนที่ผู้ใช้จะสามารถใช้ทรัพยากรใช้ทรัพยากรนั้นได้ จะต้องผ่านการตรวจสอบความปลอดภัยโดย SRM เสียก่อน

Process Manager ใช้ในการบริหารโปรเซสและเทรดโดยที่โปรเซสหนึ่ง ๆ ของโปรแกรมที่ทำงานบนวินโดวส์เอ็นทีนั้นสามารถที่เทรดที่ทำงานขนาดกันไปได้มากกว่าหนึ่งเทรด (Multi-Thread) สิ่งที่ Process Manager จะรับผิดชอบได้แก่การสร้างโปรเซสขึ้นเมื่อมีการร้องขอ หรือการทำลายเมื่อใช้งานเสร็จสิ้น

Virtual Memory Manager หรือ VMM ใช้ในการบริหารหน่วยความจำเสมือนโดยนำเอาพื้นที่ในจานแม่เหล็กมาทำเป็นหน่วยความจำแล้ว VMM จะเป็นผู้ที่สลับชิ้นส่วนของหน่วยความจำที่อยู่ในจานแม่เหล็กกับหน่วยความจำหลัก วัตถุประสงค์ของ VMM เพื่อให้โปรแกรมสามารถมีขนาดใหญ่กว่าหน่วยความจำที่มีอยู่จริงของเครื่องคอมพิวเตอร์ได้ หลักการทำงานของ VMM เป็นดังรูปที่ 6.2



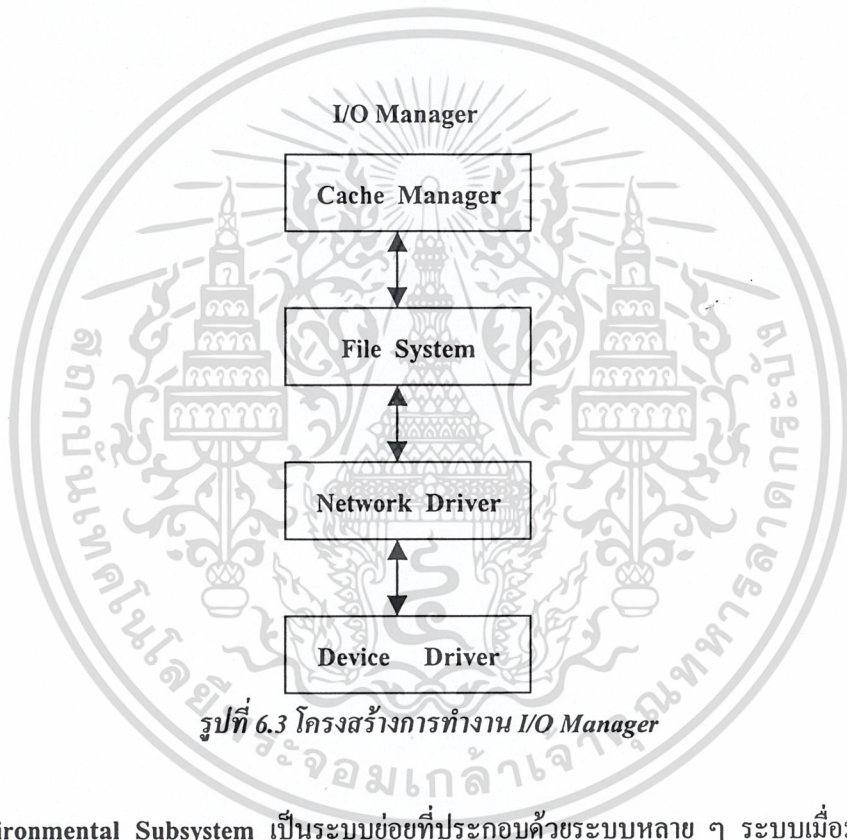
รูปที่ 6.2 หลักการทำงานของ VMM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Local Procedure Call หรือ LPC เป็นระบบย่อยของระบบ Environment ซึ่งได้แก่ DOS/Win 16, O/2S, POSIX ทำหน้าที่เป็นผู้ให้บริการสำหรับ โปรแกรมลูกข่ายที่อยู่ในเครื่องเดียวกันเรียกใช้งาน เพื่อจำลองระบบการทำงานแบบ Client/Server ถึงแม้ว่าจะอยู่บนเครื่องเดียวกัน

Remote Procedure Call หรือ RPC มีลักษณะการทำงานคล้ายกับ LPC มาก เพียงแต่มีความแตกต่างกันที่เครื่องลูกข่ายจะเรียกใช้บริการผ่านเครือข่ายไปยังเครื่องเซิร์ฟเวอร์ที่เป็นเครื่องคอมพิวเตอร์คนละเครื่องกัน

I/O Manager รับผิดชอบกระบวนการอ่าน/เขียนข้อมูลของระบบ ซึ่งจะทำงานโดยติดต่อกับไดรเวอร์ของอุปกรณ์ต่างโครงสร้างการทำงานจะเรียกใช้ต่อกันเป็นทอดๆ ดังรูปที่ 6.3 ลักษณะเช่นนี้ทำให้การเปลี่ยนอุปกรณ์ สามารถทำได้ง่าย โดยไม่กระทบกับโปรแกรมส่วนบนที่ไม่เกี่ยวข้อง



รูปที่ 6.3 โครงสร้างการทำงาน I/O Manager

Environmental Subsystem เป็นระบบย่อยที่ประกอบด้วยระบบหลาย ๆ ระบบเมื่อมาประกอบกันแล้วทำให้วินโดวส์เอ็นทีสามารถรองรับการทำงานโปรแกรมที่มาจากระบบปฏิบัติการหลาย ๆ ได้แก่ DOS, Win16, Win32, OS/2, POSIX (POSIX คือมาตรฐานการทำงานแบบยูนิกซ์) โดยที่ Environmental Sub system จะจำลองสภาพแวดล้อมของระบบปฏิบัติการสำหรับ โปรแกรมที่มาจากระบบปฏิบัติการต่างๆ จัดให้อยู่ใน Address Space ที่ต่างกันและทำงานโดยไม่รบกวนกันซึ่งมีประโยชน์ในกรณีที่โปรแกรมใดโปรแกรมหนึ่งหยุดทำงานก็จะไม่ทำให้ระบบทั้งระบบหยุดทำงาน

6.2.การติดตั้งและจัดการโปรโตคอลเครือข่ายแบบ TCP/IP

เอกสารนี้เป็น วินโดวส์เอ็นทีสนับสนุนโปรโตคอลเครือข่ายหลายประเภท โปรโตคอลที่สำคัญมีดังนี้ โยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- NetBEUI (NetBIOS Extended Use Interface) เป็นโปรโตคอลที่ใช้กับเครือข่ายแบบไมโครซอฟต์ (Microsoft Network) ซึ่งเป็นเครือข่ายที่บริษัทไมโครซอฟต์คิดค้นขึ้นและใช้กันเรื่อยมา โปรโตคอลชนิดนี้ใช้ในการติดต่อสื่อสารระหว่างเครื่องที่ใช้ระบบปฏิบัติการของไมโครซอฟต์ ได้แก่ MSDOS, Windows, Windows for Workgroup, Windows 95 และ Windows NT โปรโตคอลชนิดนี้เหมาะในการใช้งานเครือข่ายขนาดเล็ก (จำนวนเครื่องถูกข่ายไม่เกิน 200 เครื่อง) และโปรโตคอลชนิดนี้ไม่สามารถ Route ผ่านอุปกรณ์หาเส้นทาง (Router) ได้ ทำให้ไม่สามารถใช้งานในเครือข่ายระยะไกล เช่น อินเทอร์เน็ต ได้

- IPX/SPX (Internetwork Packet Exchange/Sequenced Packet Exchange) เป็นโปรโตคอลที่พัฒนาโดยบริษัทโนเวล เพื่อใช้กับระบบปฏิบัติการเครือข่ายเน็ตแวร์โปรโตคอลชนิดนี้สามารถ Route ผ่าน Router ได้ แต่ประสิทธิภาพต่ำซึ่งต่อมาโนเวลได้ทำการปรับปรุงประสิทธิภาพและได้เพิ่มเข้าไปในเน็ตแวร์เวอร์ชัน 4 ส่วนเน็ตแวร์เวอร์ชัน 3 ต้องติดตั้งโปรแกรมเพิ่มเติม

- TCP/IP (Transmission Control Protocol/Internet Protocol) เป็นโปรโตคอลมาตรฐานที่ใช้กันอยู่ในระบบปฏิบัติการแบบยูนิกซ์เริ่มพัฒนาโดยกระทรวงกลาโหมของสหรัฐ ในปี ค.ศ. 1969 เพื่อเชื่อมต่อเครื่องคอมพิวเตอร์หลายชนิดที่อยู่ห่างไกลกัน เครือข่ายที่จัดตั้งในระยะแรกชื่อว่า ARPANET ต่อมาได้พัฒนาเป็นเครือข่ายอินเทอร์เน็ต โปรโตคอลนี้เหมาะสำหรับเชื่อมต่อคอมพิวเตอร์ทั้งใกล้เข้าด้วยกัน และมีมาตรฐานรองรับทำให้ผู้ผลิตฮาร์ดแวร์และซอฟต์แวร์สามารถสร้างอุปกรณ์และโปรแกรมที่จะรองรับ ทำให้เครื่องคอมพิวเตอร์สามารถรับส่งข้อมูลกันได้ไม่ว่าจะเป็นเครื่องขนาดเล็กหรือขนาดใหญ่หรือจะใช้ระบบปฏิบัติการอะไรก็ตาม TCP/IP เป็นชุดโปรโตคอลที่ประกอบด้วยโปรโตคอลต่างๆ หลายโปรโตคอล แต่ละโปรโตคอลมีคุณลักษณะและมีความสามารถในการทำงานแตกต่างกัน

6.2.1. มาตรฐานระบบการสื่อสาร

กระบวนการติดต่อสื่อสารระหว่างอุปกรณ์สื่อสารหรือคอมพิวเตอร์นั้น สามารถแบ่งเป็นหน้าที่ย่อย ๆ ได้เป็น 7 ระดับตามมาตรฐานของ ISO (International Organization for Standardization) ดังรูปที่ 6.4

Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

รูปที่ 6.4 มาตรฐานของ ISO (International Organization for Standardization)

โดยที่ในแต่ละระดับมีหน้าที่แตกต่างกันซึ่งทำงานร่วมกันสาเหตุที่ต้องแบ่งระดับการสื่อสารออกเป็นหลายระดับก็เพื่อให้สินค้าของผู้ผลิตที่ต่างกันสามารถนำมาเชื่อมต่อกันเช่นผู้ผลิตการ์ดเครือข่ายสามารถเขียนโปรแกรมในระดับ Data link Layer เพื่อรับข้อมูลจากผู้ผลิตที่เขียนโปรแกรมในระดับ Network Layer ได้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

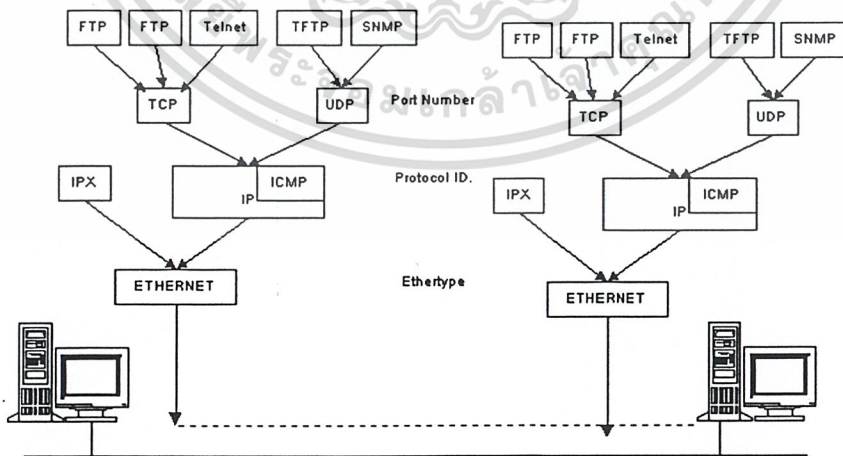
การติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์ในแต่ละระดับ จะติดต่อกันเองในลักษณะที่เรียกว่า Peer-to-peer คือแต่ละ Layer จะมองเฟรมข้อมูลที่มาจก Layer ข้างบนเป็นข้อมูลและจะนำเอาข้อมูลที่ใช้สำหรับการทำงานใน Layer ของตนเองครอบข้อมูลไว้ที่ส่วนหัว/ส่วนท้าย แล้วส่งต่อไปยัง Layer ข้างล่าง และทำเช่นนี้ไปเรื่อยๆ จนกระทั่งส่งออกไปยังสายการสื่อสาร เมื่อเครื่องผู้รับได้รับเฟรมข้อมูลก็จะถอดรหัสส่วนหัว/ส่วนท้ายออกไปแล้วนำส่งข้อมูลให้กับ Layer ข้างบนต่อไป เป็นเช่นนี้ไปเรื่อย ๆ จนถึงโปรแกรม Application

รูปที่ 6.5 เป็นการเปรียบเทียบโปรโตคอลแบบต่าง ๆ กับมาตรฐานของ ISO

OSI	NetWare	UNIX	Apple	LAN Manager
Application	NetWare Core Protocol		AppleShare	
Presentation			AppleTalk Filing Protocol (AFP)	
Session	Named Pipes NetBIOS	SNMP FTP SMTP Telnet	ASP ADSP ZIP PAP	NetBIOS Named Pipe
Transport	SPX		ATP NBP AEP RTMP	NetBEUI
Network	IP		Datagram Delivery Protocol (DDP)	
Data Link	LAN Drivers		LAN Drivers	
Physical	ODI NDIS	Media Access Control	Local-Talk Ether-Talk Token-Talk	NDIS
Physical	Physical		Physical	

รูปที่ 6.5 การเปรียบเทียบโปรโตคอลแบบต่าง ๆ กับมาตรฐานของ ISO

รูปต่อไปนี้แสดงถึงการติดต่อสื่อสารระหว่างคอมพิวเตอร์ที่ใช้โปรโตคอล TCP/IP



รูปที่ 6.6 แสดงถึงการติดต่อสื่อสารระหว่างคอมพิวเตอร์ที่ใช้โปรโตคอล TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 6.6 โพรโทคอล TCP/IP ประกอบด้วยโปรโตคอลย่อยหลายโปรโตคอลซึ่งทำงานในระดับต่างๆ กันโดยที่โปรโตคอลย่อยที่ทำงานในระดับ Application Layer ได้แก่ FTP (File Transfer Protocol), TELNET (Terminal Emulation), SNMP (Simple Network Management Protocol) โพรโตคอลเหล่านี้จะเรียกใช้โปรโตคอลย่อยในระดับ Transport ด้านล่างแบบใดแบบหนึ่งระหว่าง TCP (Transmission Control Protocol) หรือ UDP (User Datagram Protocol) และโปรโตคอลย่อยทั้งสองจะเรียกใช้โปรโตคอลระดับ Network ต่อซึ่งได้แก่ IP หรือ ICMP (Internet Communication Message Protocol)

การทำงานในระดับ Transport ที่มีโปรโตคอลย่อย TCP หรือ UDP นั้น โปรแกรมที่อยู่ด้านบนจะเรียกใช้ผ่านช่องทางที่เป็นตัวเลขที่เรียกว่า Port Number โดยหมายเลขนี้เป็นเลขที่มาตรฐานในโปรโตคอลแบบ TCP/IP มาตรฐานของ Port Number ทั้ง TCP และ UDP ได้แสดงไว้ในตารางต่อไปนี้

TCP PORT NUMBER	Application Layer Service
5	Remote Job Entry
7	Echo
20	FTP Data
21	FTP Control
23	TELNET
25	SMTP
37	Time
53	Domain Name Server (DNS)
66	Oracle SQL*Net
80	World Wide Web HTTP
110	Post Office Protocol (POP3)

UDP PORT NUMBER	Application Layer Interface
7	Echo
13	Day time
37	Time
69	Trivial File Transfer Protocol (TFTP)
70	Gopher
107	Remote Telnet
161	SNMP (Simple Network Management Protocol)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.2. การ Implement โพรโทคอล TCP/IP ของระบบปฏิบัติการต่าง ๆ

- * ในระบบปฏิบัติการยูนิกซ์ โพรโทคอล TCP/IP จะเป็นส่วนหนึ่งของระบบปฏิบัติการ
- * ในระบบปฏิบัติการวินโดวส์เอ็นที, VMS, OS/2, Windows จะอยู่ในรูปของ Device Driver เพื่อให้โปรแกรมเรียกใช้ ซึ่งต้องการทำการติดตั้ง Device Driver ก่อน
- * ในระบบปฏิบัติการ MS -DOS มักจะสร้างเป็นโปรแกรมแบบ TSR (Terminate Stay Resident) เมื่อเรียกโปรแกรมแล้วจะฝังตัวอยู่ในหน่วยความจำเพื่อให้โปรแกรมสามารถเรียกใช้งานได้
- * กรณีของระบบปฏิบัติการแบบ Windows โพรโทคอล TCP/IP จะถูกสร้างอยู่ในรูปของ DLL (Dynamic Link Library) เพื่อให้โปรแกรมในวินโดวส์สามารถเรียกใช้งานได้ โดยมีมาตรฐาน เรียกว่า WinSock (ย่อมาจาก Window Socket) ซึ่งมีบริษัทต่าง ๆ ที่ได้คิดค้นโพรโทคอลแบบ TCP/IP ที่ทำงานในลักษณะนี้หลาย ๆ บริษัท เช่น บริษัทโมโครซอฟต์, บริษัทโนเวล, FTP Software Corp, NetManage Corp รวมทั้งโปรแกรมที่เป็นประเภท Shareware ได้แก่ Trumper Winsock

โพรโทคอลแบบ TCP/IP ในระดับ Application ที่สำคัญมีดังนี้

- * FTP
- * TELNET
- * SMTP
- * HTTP
- * DNS

6.2.2.1. FTP (File Transfer Protocol)

FTP ใช้ในการรับ-ส่งแฟ้มข้อมูลระหว่างเครื่องลูกข่ายและเครื่องเซิร์ฟเวอร์โดยที่ เครื่องเซิร์ฟเวอร์จะต้องมีโปรแกรมให้บริการ FTP (FTP Server) ติดตั้งและทำงานอยู่เพื่อให้เครื่องลูกข่ายที่รันโปรแกรม FTP Client สามารถเข้ามาขอใช้บริการได้นอกจากรับส่งแฟ้มข้อมูลแล้ว FTP ยังมีคำสั่งที่ใช้ในการให้บริการต่อไปนี้

- * แสดงชื่อแฟ้มข้อมูลบนเครื่องเซิร์ฟเวอร์
- * เปลี่ยนชื่อแฟ้มหรือลบแฟ้มข้อมูล

FTP จะใช้โพรโทคอลย่อยชั้น Transport คือ TCP โดยที่ TCP จะมีลักษณะการทำงานแบบ Connection oriented คือ เครื่องเซิร์ฟเวอร์และเครื่องลูกข่ายจะต่อเชื่อมกันตลอดเวลาเสมือนมีสายเชื่อมโยงกันโดยตรง (Virtual Circuit) เพื่อทำการรับส่งข้อมูลระหว่างกันโดยมีกระบวนการตอบรับ (Acknowledgement) ว่าได้รับข้อมูลถูกต้องหรือไม่

6.2.2.2. TELNET

TELNET เป็นบริการที่ให้เครื่องลูกข่ายสามารถเข้าไปใช้เครื่องเซิร์ฟเวอร์โดยการจำลองตัวเองให้ทำงานเป็นเทอร์มินัล ผู้ใช้งานจะต้องใส่รหัสใช้และรหัสผ่านเพื่อแจ้งการเข้าใช้เครื่องเมื่อเข้าไปได้แล้วการทำงานต่าง ๆ จะเหมือนกับการเข้าไปทำงานที่หน้าจอของเครื่องเซิร์ฟเวอร์การทำงานแบบนี้เครื่องเซิร์ฟเวอร์จะ

ต้องติดตั้งโปรแกรมให้บริการ TELNET ซึ่งโดยปกติในระบบปฏิบัติการยูนิกซ์จะมีบริการนี้ติดตั้งไว้แล้วเป็นมาตรฐานมีศัพท์เรียกโปรแกรมให้บริการบนเครื่องยูนิกซ์ว่า daemon เช่น FTP daemon, TELNET daemon เป็นต้น โพรโทคอล TELNET จะใช้ TCP Port เลขที่ 23

6.2.2.3. SMTP

SMTP เป็นการให้บริการเพื่อรับส่งจดหมายอิเล็กทรอนิกส์ (e-mail) โดยที่ SMTP จะมีตู้ไปรษณีย์เพื่อทำหน้าที่รับจดหมายจากผู้อื่นที่ต้องการส่งให้และเก็บจดหมายของผู้ใช้ที่ต้องการส่งไปยังผู้อื่น เมื่อถึงกำหนดเวลาที่ตั้งไว้โปรแกรมจะทำการส่งจดหมายออกและรับจดหมายเข้ามาผู้ใช้ก็สามารถจะเปิดอ่านได้เมื่อต้องการ

SMTP จะใช้โปรโตคอล TCP Port เลขที่ 25 ส่วนการรับส่งจดหมายระหว่างเครื่องลูกข่ายกับ SMTP Server ในลักษณะที่เป็น Client/Server จะใช้โปรโตคอลที่ชื่อว่า POP3 (Port Office Protocol)

6.2.2.4. HTTP

HTTP ใช้ในการติดต่อรับส่งข้อมูลชนิดไฮเปอร์เท็กซ์ (Hypertext) ระหว่างเครื่องลูกข่ายกับ WWW Server (World Wide Web) โดยที่เอกสารนี้จะอยู่ในรูปแบบที่เขียนในภาษา HTML (HyperText Markup Language) เอกสารแต่ละชิ้นจะสามารถเชื่อมโยงไปยังเอกสารชิ้นอื่นได้ ซึ่งเอกสารที่ถูกเชื่อมโยงนี้อาจจะอยู่บนเครื่องคอมพิวเตอร์เดียวกันหรือต่างเครื่องกันก็ได้

บริการแบบ WWW เกิดขึ้นภายในเครือข่ายอินเทอร์เน็ตในปี 1990 เมื่อนักฟิสิกส์ที่ CERN ซึ่งเป็นหน่วยงานวิจัยในประเภทสวิสเซอร์แลนด์ ต้องการแลกเปลี่ยนกันโดยใช้ FTP จึงคิดค้นภาษา HTML ขึ้นมาเพื่อจัดทำเอกสารวิจัยที่สามารถเชื่อมโยงกับเอกสารวิจัยที่เกี่ยวข้องกันได้โดยในระยะแรกโปรแกรมที่ใช้ในการแสดงข้อมูลแบบนี้มีการติดต่อกับผู้ใช้แบบ Text Mode ต่อมามีนักศึกษาปริญญาโทชื่อ Marc Andressen ที่ NCSA ได้ทำโปรแกรมที่สามารถแสดงผลข้อมูล Hypertext ให้อยู่ในรูปแบบที่เป็นกราฟิกโปรแกรมที่สร้างขึ้นมีชื่อว่า MOSAIC และในขณะเดียวกันมีการพัฒนาให้ HTML สามารถรับข้อมูลที่เป็นรูปภาพ และมัลติมีเดียเพิ่มขึ้นไป ทำให้เกิดการขยายตัวในการใช้เครือข่ายอินเทอร์เน็ตกันอย่างแพร่หลาย และเกิดโปรแกรมที่ทำงานด้านนี้มากขึ้น เช่น Internet Explorer ของไมโครซอฟต์, Netscape Navigator ของเน็ตสเคปคอมมูนิตี้ชัน โปรแกรมเหล่านี้เป็นโปรแกรมประเภท WWW Client ซึ่งมีชื่อเรียกอีกอย่างหนึ่งว่า WWW Browser

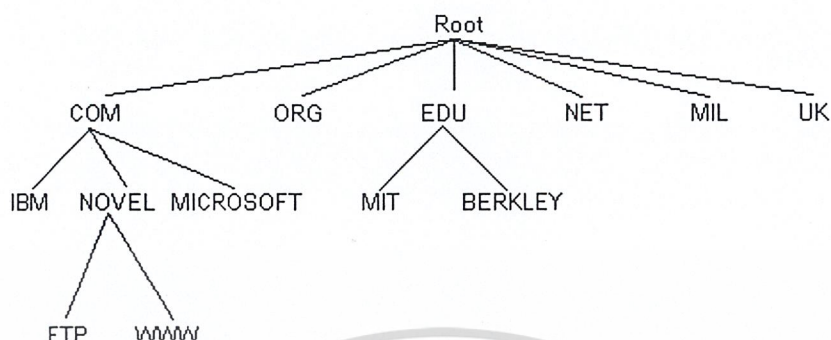
การทำงานของ WWW จะใช้โปรโตคอล HTTP ซึ่งจะเรียกผ่าน TCP Port เลขที่ 80

6.2.2.5. DNS (Domain Name Server)

ในการเชื่อมโยงเครือข่ายแบบใช้โปรโตคอล TCP/IP นั้นเครื่องเซิร์ฟเวอร์และเครื่องลูกข่ายทุกตัวจะต้องมีหมายเลขที่ใช้ในการระบุตัวเองคล้ายกับชื่อ-นามสกุลของหมายเลขที่กล่าวมานี้เรียกว่า IP Address ซึ่งมีความยาว 4 ไบต์หรือ 32 บิต โดยเขียนในลักษณะนี้ 203.154.126.134. การจดจำ IP Address เป็นสิ่งที่ทำได้ยากกว่าการจำชื่อของเครื่องคอมพิวเตอร์ดังนั้นจึงเกิดการสร้างเซิร์ฟเวอร์ที่จะให้บริการการสอบถามชื่อเครื่องและ IP Address ในเครือข่ายอินเทอร์เน็ตขึ้นมา ซึ่งเรียกว่า Domain Name Services ในการใช้งานนั้นผู้ใช้เพียงแต่ละบุ IP Address ของเครื่องที่ให้บริการนี้แล้วเมื่อต้องการจะติดต่อกับเครื่องคอมพิวเตอร์เครื่องใดในเครือข่ายอินเทอร์เน็ต DNS จะช่วยค้นหา IP Address ของเครื่องที่ต้องการให้เพื่อให้โปรแกรมสามารถใช้ IP Address ที่ได้ในกาติดต่อ

ไม่มีการแจ้งเตือนใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบ DNS นั้นเป็นลักษณะการประมวลแบบกระจายหมายถึงในเครือข่ายอินเทอร์เน็ตนั้นจะมีเครื่องที่ให้บริการ DNS ทำการควบคุมโดเมนจำนวนมากซึ่งจะช่วยกันในการทำงาน โดยมีการแบ่งเชิงชั้นของการระบุตำแหน่งแบบ Hierarchy ดังรูปที่ 6.7



รูปที่ 6.7 การแบ่งเชิงชั้นของการระบุตำแหน่งแบบ Hierarchy

โดยที่ระดับแรกจะเป็นระดับของหน่วยงานหรือประเทศระดับต่อไปจะเป็นหน่วยงานระดับล่างจะเป็นชื่อเครื่องเซิร์ฟเวอร์

ตัวอย่างของชื่อย่อในโดเมน

COM	บริษัท
ORG	หน่วยงานที่ไม่แสวงหากำไร
EDU	มหาวิทยาลัย/สถาบันการศึกษา
MIL	หน่วยงานทางทหาร
TH	ประเทศไทย
UK	ประเทศอังกฤษ
IBM	บริษัทไอบีเอ็ม
NOVELL	บริษัทโนเวล
MIT	มหาวิทยาลัย MIT

การอ้างชื่อเครื่องจะต้องระบุในรูปแบบดังนี้

ชื่อเครื่อง.หน่วยงาน.ประเภทหน่วยงาน

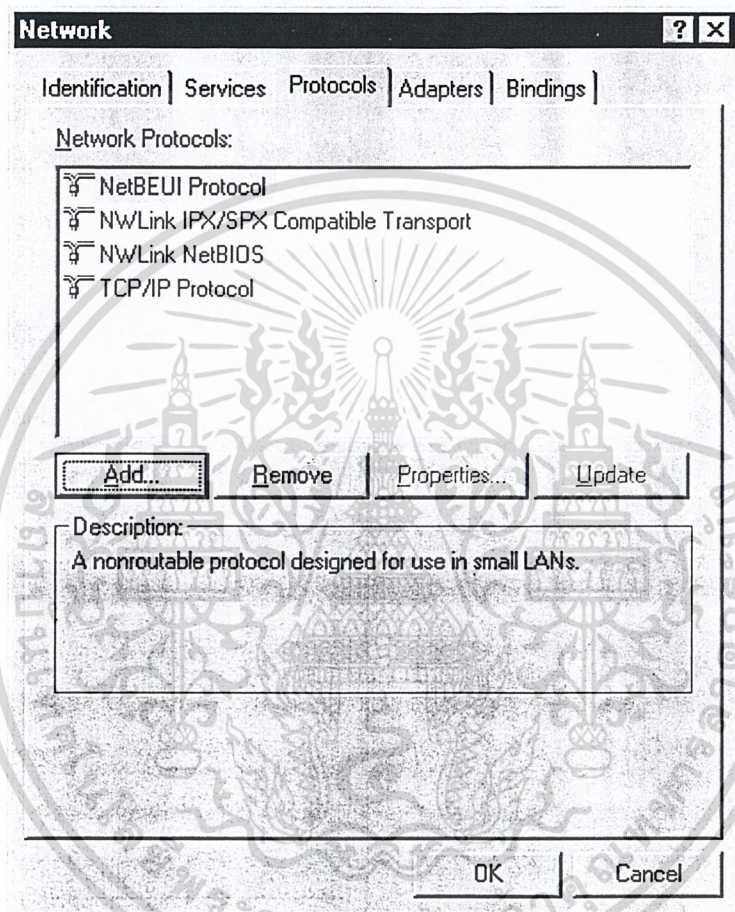
เช่น WWW.IBM, COM, ISECWWW.NIDA.AC.TH เครื่องเซิร์ฟเวอร์ที่เป็น DNS Server ในระดับต่าง ๆ จะช่วยกันทำงานเพื่อตอบคำถามในเรื่องการเปลี่ยนชื่อให้เป็น IP Address ในองค์กรขนาดใหญ่เราอาจจะติดตั้ง DNS เพื่อให้บริการภายในองค์กรเพิ่มได้ หากไม่มีการติดตั้ง DNS ไว้ก็ต้องใช้ DNS ของผู้ให้บริการอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2.3. วิธีการติดตั้งโปรโตคอล TCP/IP

ในขณะที่ติดตั้งวินโดวส์เอ็นที ถ้าได้เลือกว่าต้องการติดตั้ง Internet Information Server ด้วย โปรแกรมติดตั้งจะทำการเพิ่มโปรโตคอล TCP/IP เข้าไปด้วย หากไม่ได้ติดตั้งไว้ก่อนหรือ โปรโตคอล TCP/IP หายไป เราสามารถติดตั้งเพิ่มได้โดยวิธีการต่อไปนี้

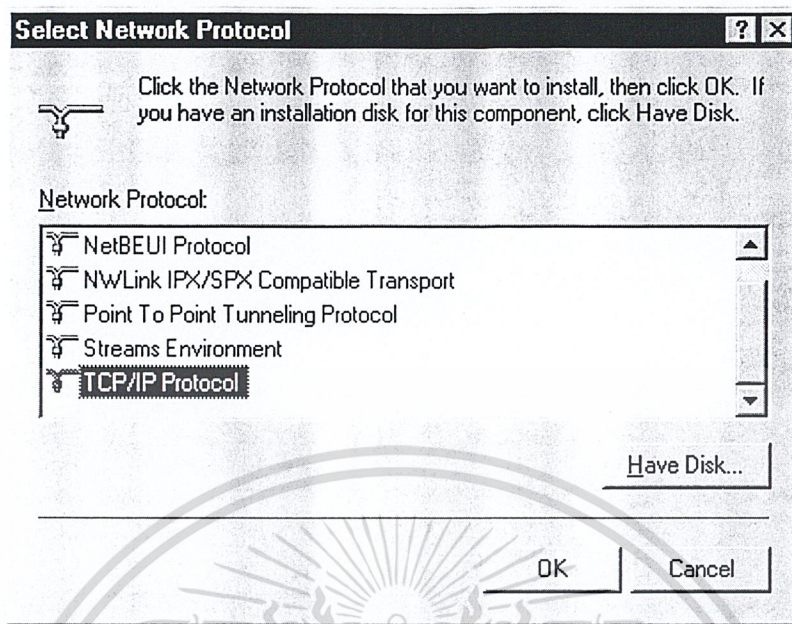
1. Logon เป็น Administrator
2. ดับเบิลคลิกไอคอน Network ที่คอนโทรลพาเนล จะปรากฏจอภาพดังรูปที่ 6.8



รูปที่ 6.8 วิธีการติดตั้งโปรโตคอล TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. คลิกเลือก Protocol Tab แล้วกดปุ่ม Add... จะปรากฏรายชื่อของโปรโตคอลขึ้นมาให้เลือกดังรูปที่ 6.9



รูปที่ 6.9 แสดงรายชื่อของโปรโตคอล

4. คลิกที่ TCP/IP Protocol แล้วกดปุ่ม OK จะปรากฏจอภาพขึ้นมาถามว่าต้องการให้โปรแกรม DHCP (Dynamic Host Configuration Protocol) Server เป็นผู้กำหนด IP Address แทนการกำหนดด้วย manual หรือไม่ (DHCP Server เป็นโปรแกรมที่ทำหน้าที่แจก IP Address ให้กับเครื่องลูกข่ายโดยอัตโนมัติ) ถ้าต้องการใช้ความสามารถนี้จะต้องระบุเครื่องที่ทำหน้าที่เป็น DHCP Server ให้กับเครื่องลูกข่ายไว้

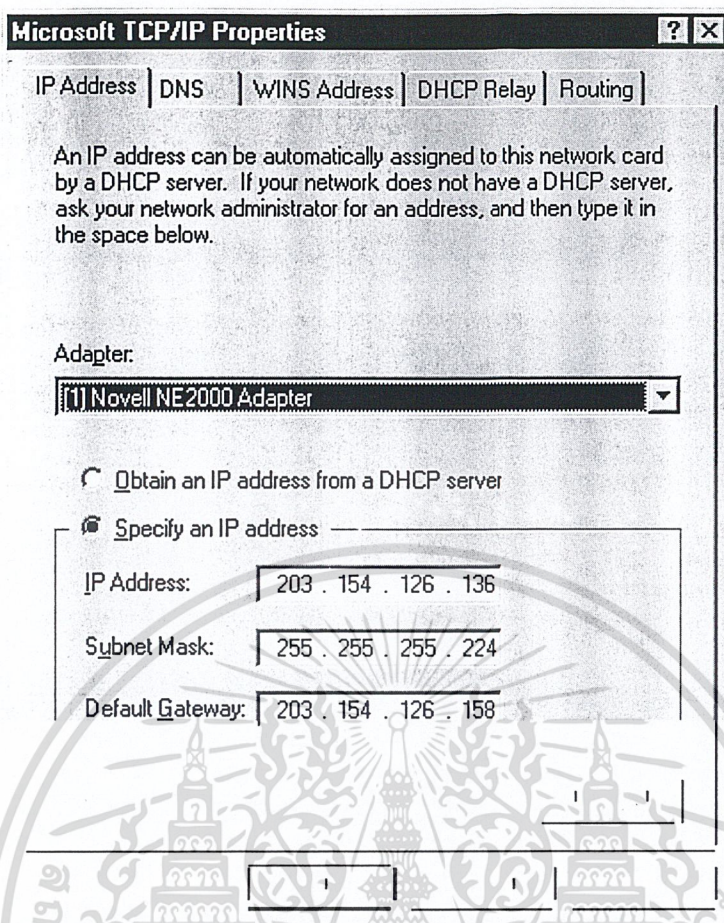
5. ให้ตอบ Yes ในกรณีที่มี DHCP Server หรือตอบ No ถ้าต้องการกำหนด IP Address ด้วย manual วินโดวส์เอ็นทีจะถามตำแหน่งของ CD-ROM ของวินโดวส์เอ็นที ให้ระบุตำแหน่งของ CD-ROM แล้วกดปุ่ม Continue โปรแกรมจะอ่าน CD-ROM เพื่อนำข้อมูลที่ต้องการเข้ามาใช้

6.2.3.1. การปรับแต่ง IP Address

ถ้าไม่ได้เลือกใช้ DHCP ไว้เราจะต้องกำหนดข้อมูลที่เกี่ยวข้องสำหรับโปรโตคอล TCP/IP ได้แก่ IP Address, Subnet Mask, Default Gateway และข้อมูลของ DNS

1. คลิกที่ Binding Tab เพื่อโปรโตคอลที่เพิ่มเข้าไปทำการ Bind กับ Services ต่าง ๆ ที่มี แล้วคลิกที่ Protocol Tab คลิกเลือก TCP/IP ในรายการแล้วกดปุ่ม Properties จะปรากฏจอภาพ Microsoft TCP/IP Properties ดังรูปที่ 6.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.10 จอภาพ Microsoft TCP/IP Properties

ถาม DHCP Server ในเครือข่ายให้เลือก Obtain an IP Address from a DHCP Server แต่อาจจะกำหนด IP Address เองให้เลือก Specify an IP Address

2. ใส่ IP Address, Subnet Mask และ Default Gateway

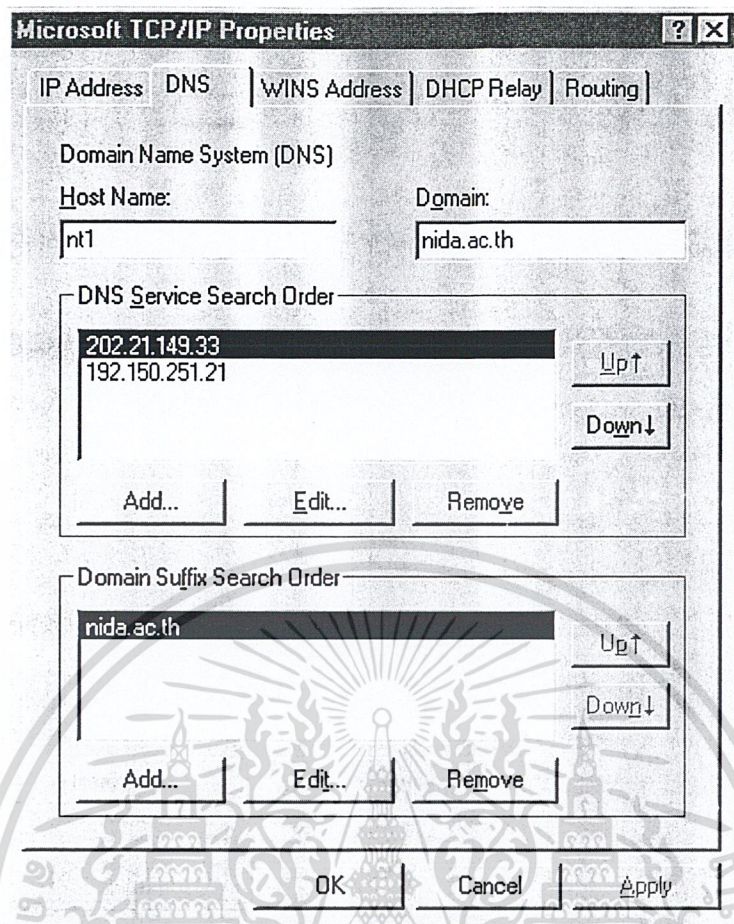
* Subnet Mask : ใช้ในการทำงานของ TCP/IP เพื่อกำหนดเครื่องคอมพิวเตอร์ที่ต้องการโดยใช้ขอบเขตของ Subnet Mask ในการคำนวณ

* Default Gateway : เป็น IP Address ของ LAN Segment ที่ Router ซึ่งเครื่องคอมพิวเตอร์ของเราเป็นสมาชิกขอเพื่อให้ออกไปยัง subnet ภายนอก IP Address ได้

* ในกรณีที่เครื่องคอมพิวเตอร์ภายในสำนักงานไม่ต้องติดต่อกับเครื่องคอมพิวเตอร์ที่เราสามารถเลือกใส่ IP Address ใดๆก็ได้ แต่จะเป็นเครื่องที่ติดต่อกับอินเทอร์เน็ตแล้ว เราจำเป็นต้องขอ IP Address เป็นกลุ่ม ๆ มาจากฝ่ายบริการอินเทอร์เน็ตที่เราเป็นสมาชิก

3. กดที่ DNS Tab จะปรากฏจอภาพดังรูปที่ 6.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.11 แสดงการใส่ IP Address, Subnet Mask และ Default Gateway

ใส่ข้อมูลของ

- * Host Name : ชื่อเครื่องคอมพิวเตอร์
- * Domain : โดเมนซึ่งเราเป็นสมาชิก
- * DNS Service Search Order : IP Address ของเครื่องที่เป็น DNS Server ใส่ได้ 3 ซึ่งเพื่อ

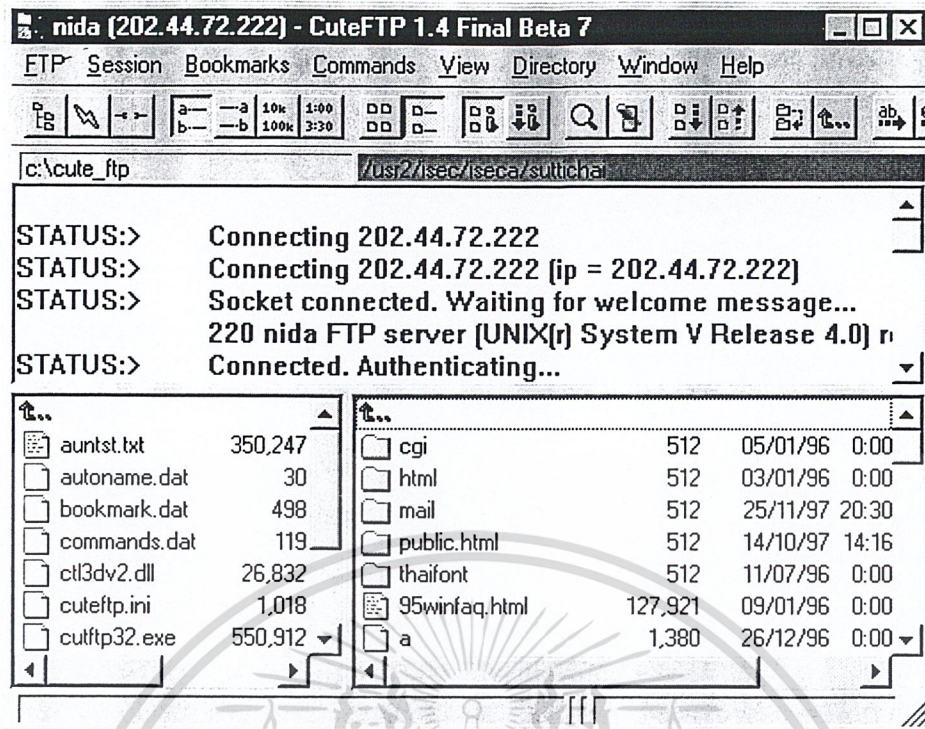
เป็นสำรองในกรณีที่เครื่องใดเสียหายจะได้ใช้เครื่องที่เหลือเป็นเซิร์ฟเวอร์แทน

4. เมื่อสำเร็จแล้วตอบ OK และตอบ Close จากนั้นทำการ Restart เครื่องใหม่เพื่อให้การเปลี่ยนแปลงมีผล

6.2.3.2. การใช้โปรแกรมที่เกี่ยวข้องกับโปรโตคอล TCP/IP

เมื่อติดตั้งโปรโตคอล TCP/IP เสร็จแล้ว เราสามารถนำเอาโปรแกรมสำหรับเครื่องลูกข่ายมาทำงานกับโปรโตคอลนี้ได้ เช่น FTP Client, WWW Browser, Telnet เป็นต้น โปรแกรมที่กล่าวมานี้ถือเป็นโปรแกรมที่เขียนขึ้นมาโดย Third Party ต่าง ๆ ในรูปที่ 6.12 แสดงการทำงานของโปรแกรม FTP Client ซึ่งทำงานในโหมดกราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.12 แสดงการทำงานของโปรแกรม FTP Client

ในวินโดวส์เอ็นทีเมื่อติดตั้งโปรโตคอล TCP/IP แล้วจะมีการ Copy โปรแกรม Utility ต่าง ๆ เพื่อช่วยในการจัดการบริหารโปรโตคอล TCP/IP มาด้วย โปรแกรมเหล่านี้เป็นโปรแกรมที่ทำงานในลักษณะที่เป็น Command Line ได้แก่

- * **arp** : ใช้สำหรับแสดงและปรับปรุงข้อมูลของชื่อเครื่องคอมพิวเตอร์กับ IP Address คำสั่งนี้ข้อมาจาก address resolution protocol
- * **hostname** : ใช้สำหรับแสดงชื่อเครื่องคอมพิวเตอร์ของเรา
- * **ipconfig** : ใช้สำหรับแสดงข้อมูลการติดตั้งของ TCP/IP ได้แก่ IP Address, Subnet Mask, Default Gateway, ข้อมูลของ DNS
- * **nbstat** : แสดงสถิติต่าง ๆ ของโปรโตคอลแบบ NetBIOS ที่วิ่งอยู่ข้างบนโปรโตคอล TCP/IP
- * **netstat** : แสดงสถิติต่าง ๆ ของโปรโตคอล TCP/IP และการเชื่อมต่อโดย TCP/IP ขณะนั้น
- * **ping** : เป็นเครื่องมือที่สำคัญที่ใช้ในการตรวจสอบเครื่องคอมพิวเตอร์ที่อยู่ในเครือข่ายว่าเครื่องคอมพิวเตอร์ดังกล่าวยังทำงานอยู่หรือไม่ คำสั่ง ping จะอาศัยโปรโตคอล ICMP (Internet Control Message Protocol) การทำงานของคำสั่ง Ping เป็นดังรูปที่ 6.13
- * **route** : ใช้ในการจัดการหรือแสดงข้อมูล Routing table ของเครื่องคอมพิวเตอร์ เช่น route print ใช้ในการแสดงข้อมูลของ Routing table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

C:\users\default>ping www.nida.ac.th

Pinging www.nida.ac.th [202.44.72.33] with 32 bytes of data:

Reply from 202.44.72.33: bytes=32 time=10ms TTL=254
Reply from 202.44.72.33: bytes=32 time<10ms TTL=254
Reply from 202.44.72.33: bytes=32 time<10ms TTL=254
Reply from 202.44.72.33: bytes=32 time<10ms TTL=254

C:\users\default>

```

รูปที่ 6.13 แสดงการทำงานของคำสั่ง Ping

* **tracert** : ย่อมาจาก Trace Route ใช้สำหรับการวิเคราะห์เพื่อหาเส้นทางในการส่งข้อมูลจากเครื่องคอมพิวเตอร์ของเราไปยังเครื่องที่เราต้องการได้ผ่านเครือข่ายใดบ้างและใช้เวลานานเท่าไรจึงจะถึงปลายทาง ตัวอย่าง **tracert www.nectec.or.th** ใช้ดูเส้นทางที่จะเข้าถึง web site ของศูนย์คอมพิวเตอร์และอิเล็กทรอนิกส์แห่งชาติ ดังรูปที่ 6.14

```

C:\users\default>tracert www.nectec.or.th

Tracing route to www.nectec.or.th [202.44.204.33]
over a maximum of 30 hops:

  0  <10 ms    <10 ms    <10 ms    203.154.126.158
  1  <10 ms    10 ms     <10 ms    202.44.72.220
  2  <10 ms    10 ms     <10 ms    202.44.72.219
  3  *          <10 ms    10 ms     203.154.127.120
  4  1201 ms   1102 ms   1011 ms   gw-NIDA.inet-th.net [203.151.176.113]
  5  1773 ms   1131 ms   972 ms    april.inet.co.th [203.150.11.25]
  6  1652 ms   2424 ms   2533 ms   PubNet-gw.inet-th.net [203.151.176.238]
  7  3084 ms   *         2854 ms   www.nectec.or.th [202.44.204.33]

Trace complete.

C:\users\default>

```

รูปที่ 6.14 แสดงเส้นทางที่จะเข้าถึง web site ของศูนย์คอมพิวเตอร์และอิเล็กทรอนิกส์แห่งชาติ

* **finger** : ใช้สำหรับตรวจสอบว่ามีผู้ใช้ที่กำลังทำงานอยู่กับเครื่องคอมพิวเตอร์ ตัวอย่าง **finger @nida.nida.ac.th** ตรวจสอบเครื่องคอมพิวเตอร์ nida ใน โดเมน nida.ac.th อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

* **ftp** : ใช้สำหรับทำงานเป็น ftp client เครื่องที่จะรับบริการนี้จะต้องรันโปรแกรม FTP Server และกำหนดรหัสผู้ใช้ที่ต้องการให้เข้ามาใช้ได้ และอาจกำหนดรหัสผู้ใช้พิเศษที่ไม่เจาะจงว่าเป็นผู้ใดที่เรียกว่า Anonymous เพื่อให้ผู้ใช้ใด ๆ ในเครือข่ายสามารถ login เข้ามาใช้บริการได้ ปกติรหัสผ่านของ Anonymous คือ email address ของผู้ใช้นั้น ๆ ลักษณะของโปรแกรม ftp client เป็นดังรูปที่ 6.15

```

C:\>ftp ftp.nectec.or.th
Connected to ftp.nectec.or.th.
220-
220- Welcome to ThaiSarn FTP server at NECTEC.
220-
220- National Electronics and Computer Technology Center
220- National Science and Technology Development Agency
220- Bangkok, Thailand
220-
220- The local time is Mon Dec 29 14:28:06 1997.
220-
220- INTERNATIONAL USERS PLEASE USE THIS SERVER AS YOUR LAST RESOURCE.
220- Thousands of users in Thailand are relying on the congested link that
220- you are using.
220-
220- All file transfers to/from this host are logged.
220-
220- Direct all administrative concerns to ftpadmin@nectec.or.th
220-
220 ftp.nectec.or.th FTP server (Version wu-2.4(22) Thu Feb 13 18:02:26 GMT+0700
1997) ready.
User (ftp.nectec.or.th:(none)): anonymous
  
```

รูปที่ 6.15 ลักษณะของโปรแกรม ftp client

* **telnet** : ใช้สำหรับทำ remote login ไปยังเครื่องคอมพิวเตอร์ที่ให้บริการ TELNET เพื่อให้เครื่องลูกข่ายสามารถจำลองตัวเองเป็นเทอร์มินัลเข้ามาใช้ทรัพยากรในเซิร์ฟเวอร์ลักษณะการทำงานของคำสั่ง Telnet เป็นดังรูปที่ 6.16

```

Telnet - nida
Connect Edit Terminal Help

UNIX(r) System V Release 4.0 (nida)
login: sutticha
Password:
Last login: Mon Dec 29 14:28:56 from sutticha
-----
                Welcome to
-      National Institute of Development Administration (NID
-      Sukapiban-2 Rd, Bangkapi, Bangkok, THAILAND
-----
-      SysAdmin: suttichai suttitossatam <sutticha@nida.nid
-      Host name: nida.nida.ac.th <202.44.72.222>
-      Machine: SUN SPARCServer10 with 128 MBRAM 6 GBDiskAR
-      Operating System: Solaris System Rev 2.3

You have new mail.
$ █
  
```

รูปที่ 6.16 การทำงานของคำสั่ง Telnet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

6.3. การติดตั้งและบริหารอินเทอร์เน็ตเซิร์ฟเวอร์ (Installation and administration of IIS Server)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเหตุผลเบื้องหลังเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอฟต์แวร์ด้านอินเทอร์เน็ตเซิร์ฟเวอร์มีหลายชนิดๆ ที่สำคัญและมีผู้นิยมใช้มากได้แก่ Netscape Enterprise Server, Web Site ของ O'Reilly และ Internet Information Server(IIS) ของไมโครซอฟต์ แต่ละชนิดก็มีข้อดี ข้อเสียแตกต่างกันไป ข้อดีหนึ่งของ IIS ก็เป็นซอฟต์แวร์ที่แจกฟรีมากับวินโดวส์เอ็นทีเซิร์ฟเวอร์ ซึ่งสามารถติดตั้งพร้อมกับการติดตั้งวินโดวส์เอ็นทีเซิร์ฟเวอร์ หรือติดตั้งเพิ่มเติมในภายหลังก็ได้ ในบทนี้จะกล่าวถึงการติดตั้ง IIS ภายหลังจากการติดตั้งวินโดวส์เอ็นทีเซิร์ฟเวอร์เสร็จแล้ว อย่างไรก็ตามจรรยาบรรณการติดตั้งจะไม่แตกต่างกัน

6.3.1. การติดตั้ง IIS ต้องมีทรัพยากรต่อไปนี้

- * เครื่องคอมพิวเตอร์ ที่มีหน่วยความจำเพิ่มเติมจากปกติ ประมาณ 8 เมกะไบต์
- * ที่ว่างในจานแม่เหล็ก ประมาณ 30 เมกะไบต์
- * โพรโทคอล TCP/IP

ถ้าต้องการใช้ IIS ภายในองค์กร จะต้องมีเครื่องปลายทางที่เชื่อมต่อกับ IIS โดยจะต้องเตรียมการดังต่อไปนี้

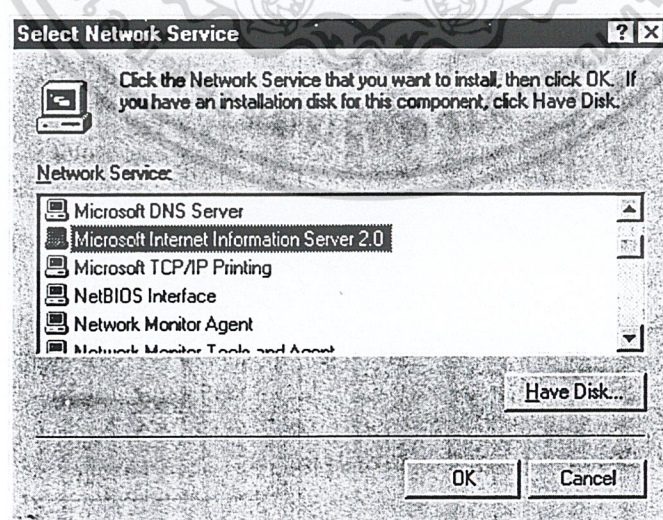
- * โพรโทคอล TCP/IP ที่เครื่องลูกข่าย
- * การจัดส่ง DNS Server หรือ WINS Server ในองค์กร

ในกรณีที่ต้องการเชื่อมต่อกับอินเทอร์เน็ต จะต้องเพิ่มเติมสิ่งต่อไปนี้

- * สายเชื่อมโยงไปยัง ISP (Internet Services Provider)
- * กลุ่ม IP Address และชื่อโดเมนซึ่งขอลงทะเบียนได้จาก ISP

6.3.2. การติดตั้ง IIS

1. Logon ด้วย Administrator
2. เปิดไอคอน network ในคอนโทรลพานเนล
3. คลิกที่ Services Tab แล้วคลิกปุ่ม Add...จะปรากฏจอภาพ Select Network Services ดังรูปที่ 6.17

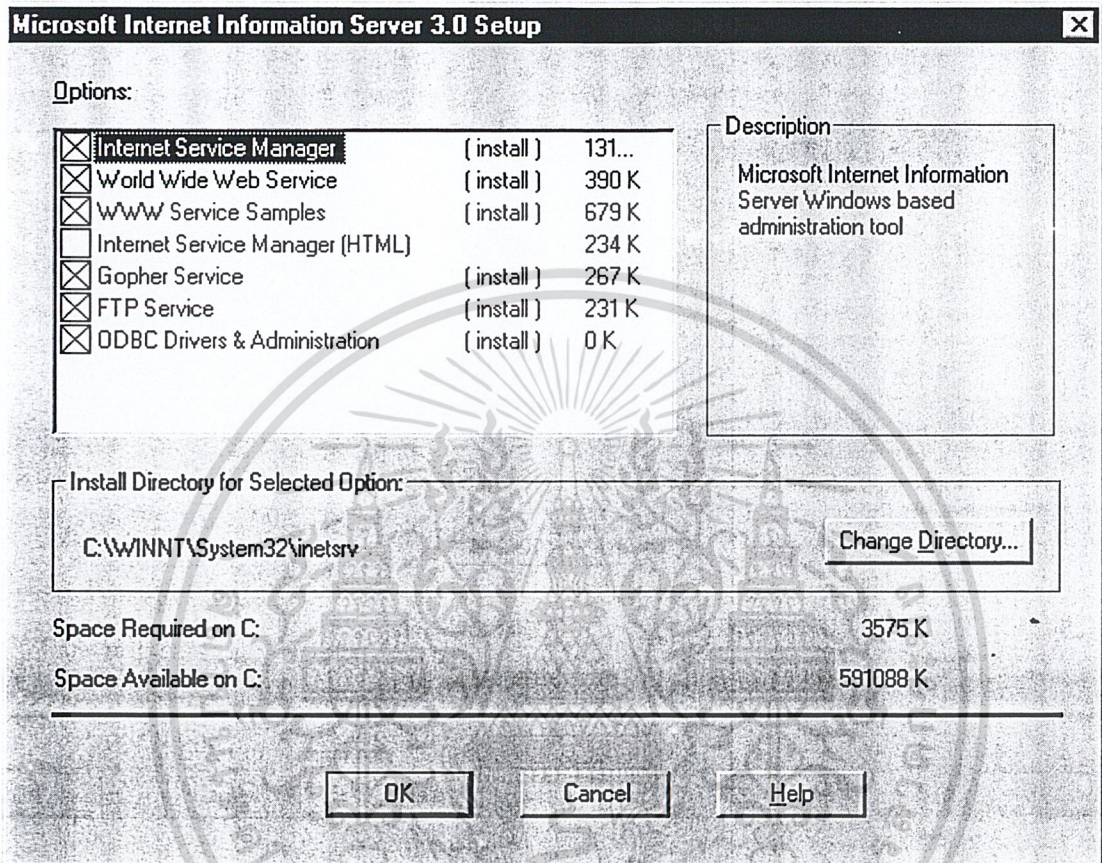


รูปที่ 6.17 จอภาพ Select Network Services

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เลือกรายการ Microsoft Internet Information Server แล้วตอบ OK โปรแกรมติดตั้งจะให้ระบุตำแหน่งที่เก็บข้อมูล CD-ROM ของวินโดวส์เอ็นทีเซิร์ฟเวอร์

5. เมื่อได้ตอบคำถามเกี่ยวกับการ Close โปรแกรมต่าง ๆ ที่กำลังทำงานอยู่แล้ว จะปรากฏจอภาพ Microsoft Internet Information Server Setup ขึ้นมาดังรูปที่ 6.18



รูปที่ 6.18 แสดงจอภาพ Microsoft Internet Information Server Setup

รายการแต่ละรายการที่แสดงขึ้นมาให้เลือกมีความหมายดังนี้

- * Internet Service Manager : โปรแกรมที่ใช้สำหรับการบริหารจัดการ IIS
- * World Wide Web Service : โปรแกรม WWW Server
- * WWW Service Sample : โปรแกรมและข้อมูลตัวอย่างสำหรับทำงานใน IIS
- * Internet Service manager(HTML) : คล้ายกับโปรแกรม Internet Service Manager แต่ทำงานโดยใช้

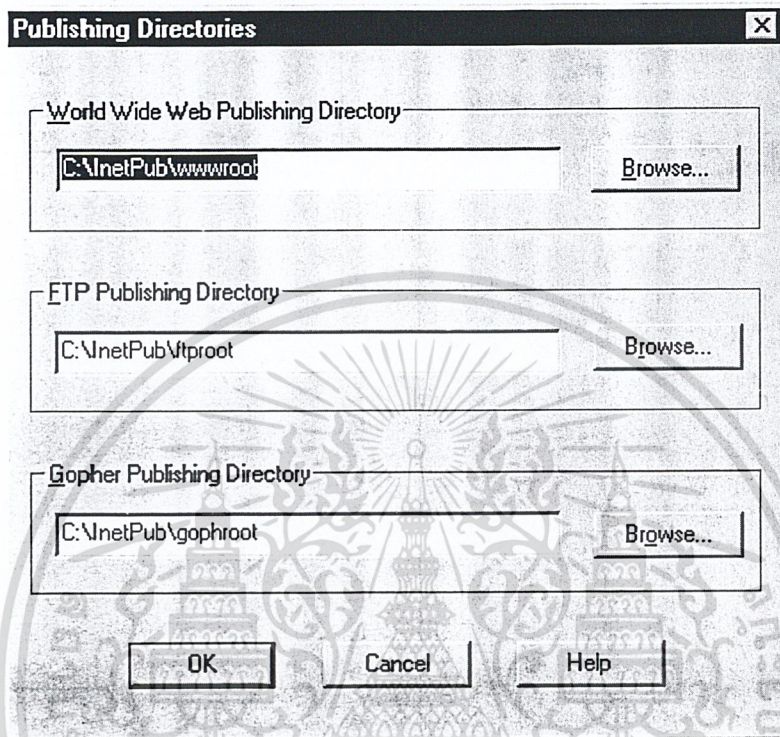
WWW Browser

- * Gopher Service : ติดตั้งโปรแกรม Gopher
- * FTP Service : ติดตั้งโปรแกรม FTP
- * ODBC Drivers and Administration : ติดตั้ง ODBC(Open Database Connectivity) ไดรฟ์และ

โปรแกรมที่ใช้ในการบริหาร ODBC (ODBC) ใช้สำหรับการเชื่อมต่อระหว่าง โปรแกรมต่าง ๆ กับระบบฐาน
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้โดยไม่ได้รับอนุญาต
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูล โดยมี ODBC เป็นตัวกลางเพื่อให้โปรแกรมที่เขียนขึ้นไม่ยึดติดกับฐานข้อมูลใด ๆ เพียงแต่ฐานข้อมูลที่จะนำมาใช้จะต้องสนับสนุน การเชื่อมต่อแบบ ODBC ด้วย

6. เมื่อเลือกรายการที่ต้องการติดตั้งแล้ว กดปุ่ม OK โปรแกรมติดตั้งจะแสดงจอภาพเพื่อให้ระบุ Path ที่ต้องการติดตั้งโปรแกรม WWW, FTP และ Gopher Server ที่เลือกไว้ดังรูปที่ 6.19



รูปที่ 6.19 แสดง ODBC Drivers and Administration

7. เมื่อระบุ Path เสร็จแล้ว ตอบ OK โปรแกรมติดตั้งจะทำการ Copy ข้อมูลจะแผ่น CD-ROM ของวินโดวส์เอ็นทีเซิร์ฟเวอร์ไปยังฮาร์ดดิสก์

6.3.3. การทดสอบผลการติดตั้ง IIS

ทดสอบโดยใช้โปรแกรม WWW Browser ซึ่งมีอยู่หลายยี่ห้อ เช่น Netscape navigator หรือ Internet Explorer โดยเรียกโปรแกรมนี้ขึ้นมา แล้วระบุข้อความใน URL โดยระบุ IP Address ของเซิร์ฟเวอร์ที่ติดตั้ง IIS (การทดสอบนี้สามารถทำได้ทั้งเครื่องลูกข่ายอื่นหรือทดสอบที่เครื่องติดตั้ง IIS) ตัวอย่าง URL

<http://203.154.126.136>

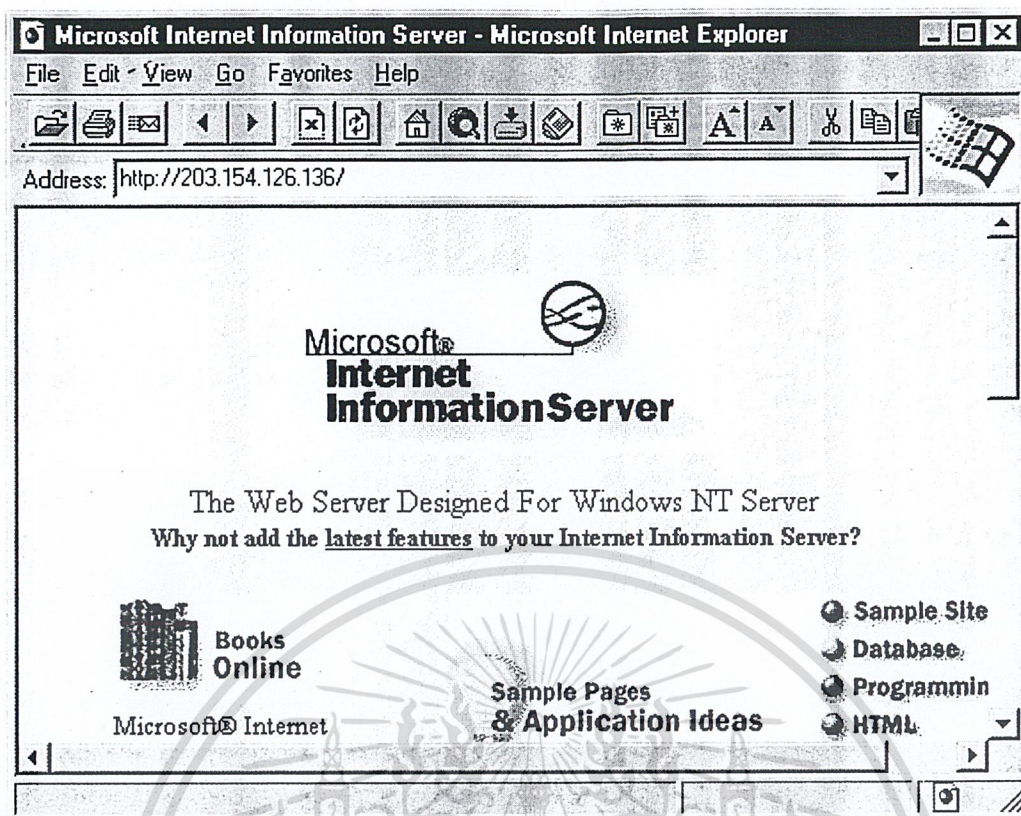
ในกรณีที IP Address ได้ลงทะเบียนเป็นชื่อแล้วก็สามารถใช้ชื่อนั้นใน URL ได้ เช่น

<http://isecwww.nida.ac.th>

ถ้าการติดตั้ง IIS เสร็จเรียบร้อยจะปรากฏจอภาพแสดงข้อมูลดังรูปที่ 6.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาระดับบัณฑิตศึกษา ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

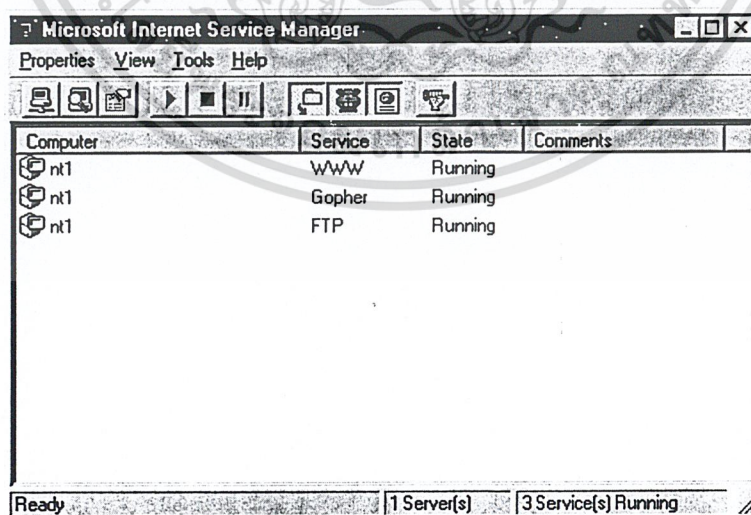
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.20 จอภาพเมื่อติดตั้ง IIS เรียบร้อย

6.3.4. การบริหารจัดการ IIS

การบริหารและจัดการ IIS ทำได้โดยใช้โปรแกรม Internet Service Manager หรือ โปรแกรม Internet Service Manager (HTML) ก็ได้ โปรแกรมทั้งสองจะอยู่ในเมนู Microsoft Internet Server (Common) เมื่อเรียกโปรแกรม Internet Service Manager จะปรากฏจอภาพดังรูปที่ 6.21



รูปที่ 6.21 แสดงโปรแกรม Internet Service Manager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะการแสดงรายการของบริการต่าง ๆ มี 3 ลักษณะคือ แสดงรายการเรียงตาม Server(Server View) แสดงรายการเรียงตาม Server โดยแสดงเป็นรายละเอียด (Report View) และเรียงตาม Service (Services View) การเปลี่ยนลักษณะการแสดงผลทำได้โดยใช้เมนู View

การแสดงผลรายการแบบ Report View จะแสดงรายชื่อ Service (Running/Paused/Stopped) โดยมีเมนู Popup Menu เพื่อใช้สำหรับการ Start, Stop, Pause บริการ

โปรแกรมนี้ยังสามารถจัดการ IIS ที่รันบนเซิร์ฟเวอร์อื่นได้ โดยใช้เมนู Properties/Connect to แล้วระบุเครื่องที่ต้องการบริหาร

6.3.5. การจัดการ WWW Service

ทำโดยการเรียก Properties ของ WWW Service ขึ้นมาโดยคลิกปุ่ม Properties บนทูลบาร์หรือใช้เมนู Properties/Service Properties จะปรากฏจอภาพ WWW Service Properties ดังรูปที่ 6.22

WWW Service Properties for nt1

Service | Directories | Logging | Advanced

ICP Port: 80

Connection Timeout: 900 seconds

Maximum Connections: 100000

Anonymous Logon

Username: IUSR_NT1

Password: *****

Password Authentication

Allow Anonymous

Basic (Clear Text)

Windows NT Challenge/Response

Comment:

OK Cancel Apply Help

รูปที่ 6.22 จอภาพ WWW Service Properties

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน WWW Service Properties จะมี Tab 4 Tab ด้วยกันดังนี้

1. **Service Tab** : ใช้ในการปรับแต่งข้อมูลต่าง ๆ ของ Service รายละเอียดการปรับแต่งได้แก่

* **TCP Port** : ระบุหมายเลข Port ที่ให้บริการ WWW Service

* **Connection Time Out** : ระยะเวลาที่ WWW จะลบข้อมูลของเครื่องลูกข่ายที่ติดต่อเข้ามาแล้วและขาดการติดต่อมานานเท่าใด

* **Maximum Connection** : จำนวนเครื่องลูกข่ายสูงสุดที่ติดต่อเข้ามาพร้อมกันได้

* **Anonymous Logon, User Name** : กำหนดชื่อผู้ใช้ซึ่งเป็นตัวแทนของการเข้ามาใช้บริการ WWW ของ IIS บนวินโดวส์เอ็นทีเซิร์ฟเวอร์ โดยที่ผู้ใช้ไม่ต้องมีรหัสลับผู้ใช้บนวินโดวส์เอ็นทีเซิร์ฟเวอร์มาก่อนก็สามารถจะแยกข้อมูลไปเมื่อติดตั้ง IIS โปรแกรมติดตั้งจะสร้างรหัสลับผู้ใช้ขึ้นด้วย IUSR แล้วตามด้วยชื่อเครื่องคอมพิวเตอร์และจะกำหนดรหัสผู้ใช้นี้เป็นตัวแทนการใช้งานสำหรับการขอรับบริการด้าน WWW

* **Password Authentication**

Allow Anonymous : ยอมให้ผู้ใช้ที่ไม่มีรหัสลับผู้ใช้และใช้รหัสผู้ใช้ใน Anonymous Logon เพื่อเข้ามาใช้บริการ ได้หรือไม่

Basic (Clear Text) : วิธีการรับส่งรหัสผู้ใช้/รหัสผ่านระหว่างเครื่องลูกข่ายกับเซิร์ฟเวอร์ เป็นแบบไม่เข้ารหัส

Windows NT Challenge/Response : วิธีการรับส่งรหัสผู้ใช้/รหัสผ่านระหว่างเครื่องลูกข่ายกับเซิร์ฟเวอร์ เป็นแบบเข้ารหัส

2. **Directories** : ใช้ในการกำหนดชื่อแทน (alias) เพื่อใช้ในการอ้างถึงเอกสารแต่ละชุดกันคล้ายกับการตั้งชื่อเอกสารชุดต่าง ๆ เพื่อให้สามารถเรียกใช้ได้ง่ายและรวดเร็ว เช่น

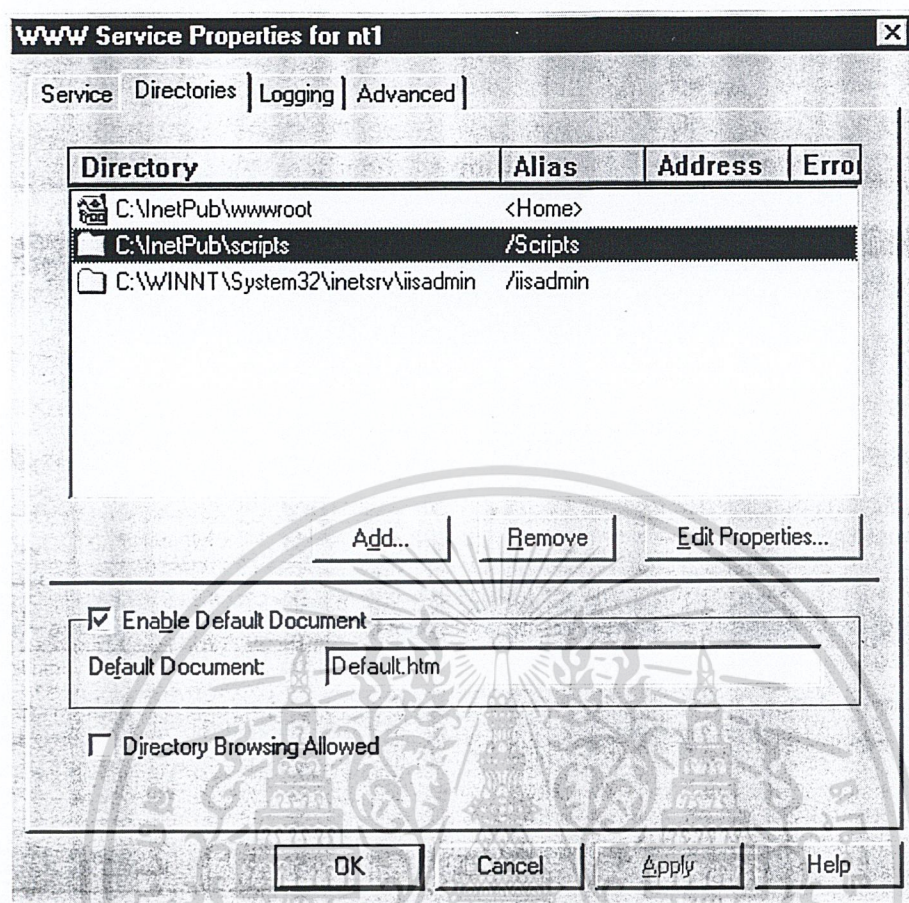
sample → c:\inetpub\wwwroot\project\sample

เป็นการตั้งชื่อ sample เป็นเอกสารชุดหนึ่งที่เก็บไว้ในโฟลเดอร์ c:\inetpub\wwwroot\project\sample และเมื่อต้องการเอกสารชุดนี้เพียงแต่ระบุ sample ไว้ต่อท้าย URL เดิม ดังนี้

<http://203.154.126.136/sample>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของ directories tab เป็นดังรูปที่ 6.23



รูปที่ 6.23 ลักษณะของ directories tab

- * พื้นที่ส่วนบน แสดงรายการของ directories และชื่อแทน (alias)
- * Enable Default Document ขอมให้เครื่องลูกค้าระบุแต่เพียงชื่อ alias เข้ามา แล้วโปรแกรมจะนำเอาเอกสารเริ่มต้นในโฟลเดอร์มาให้ เช่น <http://203.154.126.136/sample> โปรแกรมจะนำเอาเอกสารชื่อ default.htm มาจาก sample ให้ ซึ่งมีค่าเท่ากับการป้อน URL เป็น

<http://203.154.126.136/sample/default.htm>

- * Directory Browsing Allow ขอมให้สามารถแสดงรายชื่อแฟ้มและโฟลเดอร์ขึ้นมาได้ ถ้าไม่มีการระบุชื่อเอกสาร โดยแสดงรายการคล้ายกับการเรียกคำสั่ง DIR ใน MS-DOS ถ้าต้องการเพิ่มชื่อ alias หรือต้องการเปลี่ยนแปลงก็สามารถทำได้โดย กดปุ่ม Add....หรือ Fdit properties ตามลำดับ ซึ่งจะปรากฏภาพดังรูปที่ 6.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Directory Properties

Directory:

Home Directory

Virtual Directory

Alias:

Account Information

User Name:

Password:

Virtual Server

Virtual Server IP Address:

Access

Read Execute

Require secure SSL channel (Not Installed)

Enable Client Certificates Require Client Certificates

รูปที่ 6.24 แสดง Directory Browsing Allow

- * **Directory** : ระบุ Path ที่ต้องการเก็บข้อมูล
- * **Home Directory** : กำหนดว่าเป็นจุดเก็บเอกสาร โดยที่ผู้ใช้เรียกเข้ามาโดยใช้ชื่อเครื่องหรือ IP Address เท่านั้นจะทำให้ได้เอกสารใน Home Directories ไป
- * **Alias** : ชื่อแทนไดเรกทอรี
- * **Virtual Server** : กำหนดให้ NT Server มีลักษณะคล้ายกับมี Server ให้บริการอยู่หลาย ๆ Server การกำหนดแบบนี้จะต้องกำหนดให้เครื่องนี้มี IP Address หลาย ๆ อันก่อน โดยกำหนดที่ไอคอนของ Network ในคอลโทรพาเนล แล้วจึงกำหนดว่ามี Virtual Server แล้วระบุ IP Address ให้
- * **Access** : กำหนดสิทธิในการใช้งานไดเรกทอรีว่า
 - # **Read** : สามารถอ่านข้อมูลได้
 - # **Execute** : สามารถเรียกโปรแกรมมารันทำงานได้ เช่น ในกรณีที่เขียนโปรแกรมโดยใช้ CGI (Common Gateway Interface) หรือเขียนโดย ASP (Active Server Page) จะต้องระบุว่า Execute ได้

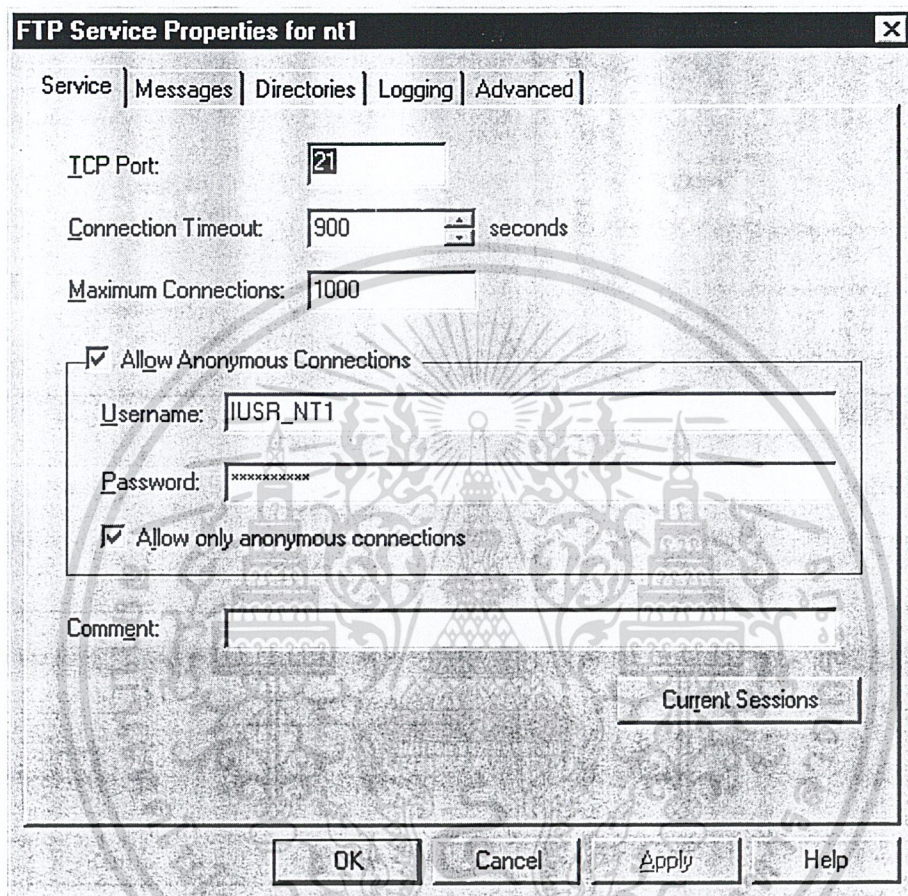
3. **Logging Tab** : กำหนดว่าจะตรวจจับการทำงานของ WWW Server พร้อมทั้งบันทึกรายการละเอียด การปฏิบัติงานไว้ในแฟ้มข้อมูลหรือไว้ในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Advance Tab : เป็นส่วนหนึ่งของการกำหนดระดับความปลอดภัย โดยการระบุ IP Address ของ เครื่องข่ายที่สามารถเข้ามาใช้บริการได้ หรือ ห้าม IP Address ของเครือข่ายที่ไม่ต้องให้เข้ามาใช้บริการ

6.4. การจัดการ FTP Service

เมื่อเรียก FTP Services Properties ขึ้นมาจะปรากฏภาพดังรูปที่ 6.25



รูปที่ 6.25 แสดง FTP Services Properties

จากรูปที่ 6.25 จะเห็นว่า มี Tab 5 Tab โดยแต่ละ Tab มีรายละเอียดการใช้งานดังนี้

1. **Service Tab** : ใช้กำหนดรายละเอียดการเข้าใช้บริการ FTP Server การกำหนดคล้ายกับของ WWW Service
2. **Message Tab** : ใช้กำหนด ข้อความสื่อสารกับเครื่องลูกข่ายเมื่อเข้ามาใช้งาน (Welcome Message) เมื่อออกจากการใช้บริการ (Exit Message)
3. **Directories Tab** : กำหนดชื่อแทนของไดเรกทอรีที่ยอมให้เข้ามาใช้งานได้ การกำหนดคล้ายกับ Service
4. **Logging Tab** : การกำหนดคล้ายกับ WWW Service

5. **Advance Tab** : การกำหนดคล้ายกับ WWW Service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาก็เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.5. การจัดการ Gopher Service

ลักษณะการจัดการคล้ายกับ WWW Service ลักษณะจอภาพเป็นดังรูปที่ 6.26

Gopher Service Properties for ntl [X]

Service | Directories | **Logging** | Advanced

ICP Port:

Connection Timeout: seconds

Maximum Connections:

Service Administrator

Name:

Email:

Anonymous Logon

Username:

Password:

Comment:

OK Cancel Apply Help

รูปที่ 6.26 แสดงการจัดการจัดการ Gopher Service

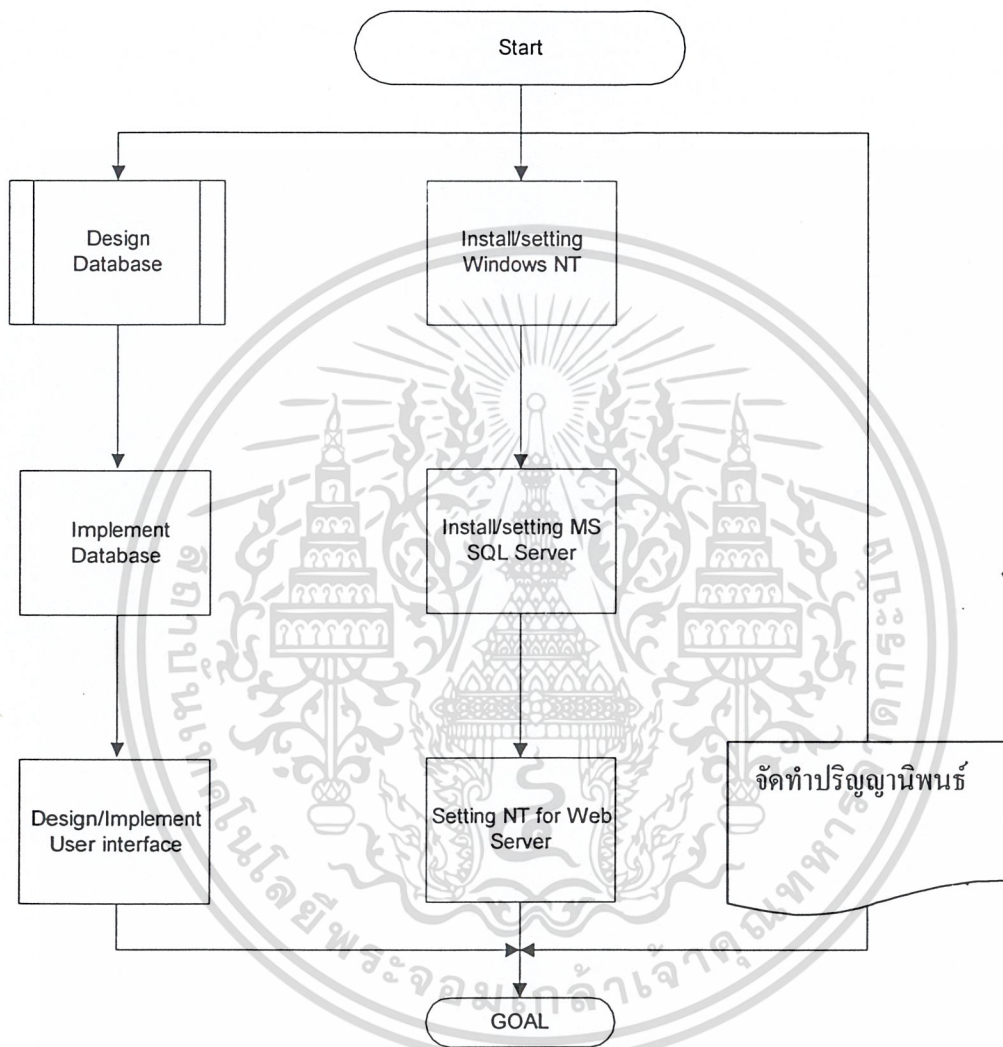
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

การดำเนินการปฏิบัติและพัฒนาโครงการ

6.1. คำนำ

ขั้นตอนในการดำเนินงานพัฒนาโครงการเราสามารถแสดงขั้นตอนคร่าวๆ ตาม Flow Chart ข้างล่างนี้



รูปที่ 7.1 Flow Chart แสดงการทำงานพัฒนาโครงการ

จากการที่อินเทอร์เน็ต(Internet) ที่ได้รับความนิยมอย่างมาก ได้แก่ การสร้างเป็นผู้ให้บริการเว็บ โดยเราจะต้องสร้างให้ผู้ให้บริการเว็บ นั้นมีรูปแบบการติดต่อกับผู้ใช้บริการ ที่น่าสนใจและมีประสิทธิภาพ และตัวอย่างงานเกี่ยวกับงานให้บริการเว็บ ที่ดีและมีประโยชน์ต่อบุคคลทั่วไปก็คือ “การให้บริการค้นหาเส้นทางจราจรในกรุงเทพมหานครและเขตปริมณฑล” เราสามารถสรุปงานของเราออกเป็น 2 ส่วนใหญ่ๆ คือ

1.การพัฒนาฐานข้อมูลจราจรเมืองหลวง

2.การพัฒนาการติดต่อกับผู้ใช้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2. รายละเอียดของโครงการงาน

รายละเอียดของระบบฐานข้อมูลจราจรเมื่อหลวงเราสามารถจำแนกได้ โดยเราจะแยกการสืบค้นออกเป็น 2 กรณี คือ

1. สืบค้นด้วยรถยนต์ส่วนบุคคล
2. สืบค้นด้วยพาหนะอื่นๆ คือ รถประจำทาง รถไฟ รถคู่ และเรือ

เราจึงแยกการติดต่อกับผู้ใช้บริการ ออกเป็น 2 กรณี และสิ่งที่เราเพิ่มเติมเข้ามาเพื่อให้เว็บไซต์มีประโยชน์ที่หลากหลายมากขึ้น คือ

1. บริการแนะนำสถานที่ท่องเที่ยวที่น่าสนใจในกรุงเทพมหานครและเขตปริมณฑล
2. บริการสอบถามและรายงานสภาพการจราจรจุดต่างๆ ในกรุงเทพมหานครและเขตปริมณฑล
3. บริการรับข้อเสนอแนะจากผู้เข้ามาใช้บริการ
4. บริการข่าวสารที่น่าสนใจ

โดยในเบื้องต้นของการทำงานเราจะเริ่มจากการหาเครื่องมือ (Tools) ที่ใช้ในการทำงาน ซึ่งจากการศึกษาเราสรุปได้ว่าสิ่งที่เราต้องใช้ประกอบการทำงานคือ

1. Microsoft Windows NT 4.0 Server : ทำหน้าที่เป็น Network Operating System
2. Microsoft SQL 6.5 Server : ทำหน้าที่เป็นระบบจัดการฐานข้อมูล
3. Microsoft Access 97 + Upsizing : ทำหน้าที่เป็นตัวสร้าง table และ Form
4. JAVA Script : เพื่อทำให้การติดต่อกับผู้ใช้บริการ ดูน่าสนใจขึ้น
5. IIS(Internet Information Server) : ทำหน้าที่ ผู้ให้บริการเว็บ

6.3. การดำเนินการพัฒนาโครงการงาน

เราจะจำแนกการแสดงผลการพัฒนาออกเป็น 2 ส่วนคือ

1. การออกแบบฐานข้อมูล
2. ออกแบบส่วนติดต่อกับผู้ใช้งาน (User Interface)

6.3.1. การออกแบบฐานข้อมูลจราจรเมื่อหลวง

เราจะจำแนกออกเป็นส่วนย่อยๆ คือ

1. การออกแบบฐานข้อมูล
2. การออกแบบคำร้องขอ (Query) ที่ใช้ในการสืบค้นเส้นทางเดินทางที่เป็นไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.1.1. การออกแบบฐานข้อมูล

เราจะอาศัยโมเดลจำลองความสัมพันธ์ระหว่างข้อมูล (E-R Model) ซึ่งมีขั้นตอนการทำงานดังนี้

1. ออกแบบฐานข้อมูลโดยใช้ E-R Model
2. แปลง E-R Model ให้อยู่ในรูปแบบฐานข้อมูลเชิงสัมพันธ์

6.3.1.1.1. ออกแบบฐานข้อมูลโดยใช้ E-R Model ประกอบด้วยขั้นตอนดังนี้

1. ศึกษาถึงลักษณะหน้าทำงานของระบบว่ามีรายละเอียดของการทำงานและข้อมูลที่เกี่ยวข้องอะไรบ้าง มีข้อสมมุติฐาน (Business Rule) ของงานต่าง ๆ อะไรบ้าง

เราสามารถจำแนกเอนทิตี (Entity) ออกตามบริการที่เรากำหนดคือ

- เอนทิตีคุณสมบัติของยานพาหนะ เราให้ชื่อว่า ' Vehicle_property '
- เอนทิตีคุณสมบัติของสถานที่ เราให้ชื่อว่า ' Place_property '
- เอนทิตีการสอบถามและรายงานสภาพการจราจร เราให้ชื่อว่า ' Traffic_service '
- เอนทิตีรับข้อเสนอแนะจากผู้ใช้บริการ เราให้ชื่อว่า ' Introduce_service '
- เอนทิตีข่าวสารที่น่าสนใจ เราให้ชื่อว่า ' News '

2. กำหนดเอนทิตีที่ควรอยู่ในฐานข้อมูล ฐานข้อมูลหนึ่ง ๆ ประกอบด้วยหลายเอนทิตี ในการกำหนดเอนทิตีที่ควรอยู่ในฐานข้อมูลหนึ่ง ๆ จะต้องคำนึงรวมไปถึงว่าเอนทิตีนั้น ๆ เป็นเอนทิตีประเภทอ่อนแอ (Weak entity) หรือเป็นเอนทิตีประเภทที่ควรแบ่งเป็น Supertype หรือ Subtype หรือไม่

เราสามารถกำหนดข้อสมมุติฐานของข้อมูลได้คือ

- สถานที่หนึ่งอาจมีพาหนะมากกว่า 1 สายหรือ 1 ชนิดผ่าน
- พาหนะ 1 สาย อาจจะผ่านสถานที่มากกว่า 1 สถานที่

3. กำหนดประเภทของความสัมพันธ์ระหว่างเอนทิตีว่ามีความสัมพันธ์อย่างไรบ้าง โดยพิจารณาจากข้อสมมุติฐานของความสัมพันธ์ระหว่างเอนทิตีที่ได้ศึกษามา

เรากำหนดให้ความสัมพันธ์ประเภท " Index_goal " เป็นความสัมพันธ์แบบ M:N ระหว่างเอนทิตี Place_property กับ Vehicle_property



4. กำหนดคุณสมบัติของเอนทิตีว่าควรมีรายละเอียดอะไรบ้าง ซึ่งการกำหนดคุณลักษณะของเอนทิตี จะพิจารณาว่ารายละเอียดต่าง ๆ เป็นรายละเอียดที่มีคุณสมบัติเป็นคีย์ หรือเป็นรายละเอียดที่แปลค่ามา เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือเป็นรายละเอียดที่ประกอบด้วยรายละเอียดที่เป็นข้อมูลผสม เช่น ที่อยู่ ประกอบด้วย บ้านเลขที่ ถนน เขต ตำบล จังหวัด รหัสไปรษณีย์ เป็นต้น

4.1.Entity Place_property ประกอบด้วยรายละเอียดดังนี้

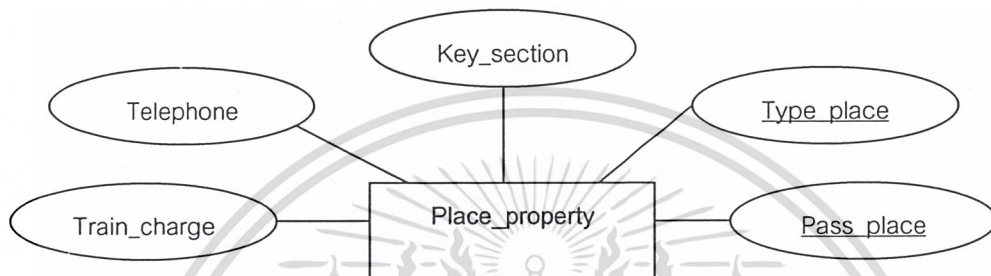
Pass_place หมายถึง ชื่อสถานที่

Type_place หมายถึง ชนิดสถานที่

Key_section หมายถึง เขต

Telephone หมายถึง หมายเลขโทรศัพท์สถานที่นั้นๆ

Train_Charge หมายถึง ค่าบริการจากกรุงเทพถึงสถานีนี้นั้นๆ



4.2.Entity Vehicle_property ประกอบด้วยรายละเอียดดังนี้

ID_vehicle หมายถึง หมายเลขพาหนะ

Type_vehicle หมายถึง ชนิดพาหนะ

Service_time หมายถึง ช่วงเวลาที่ให้บริการ

Detail_vehicle หมายถึง ชนิดของพาหนะประเภทเรือ

Charge หมายถึง ค่าโดยสารรถตู้, ขสมก.

Start_station หมายถึง สถานีต้นทาง

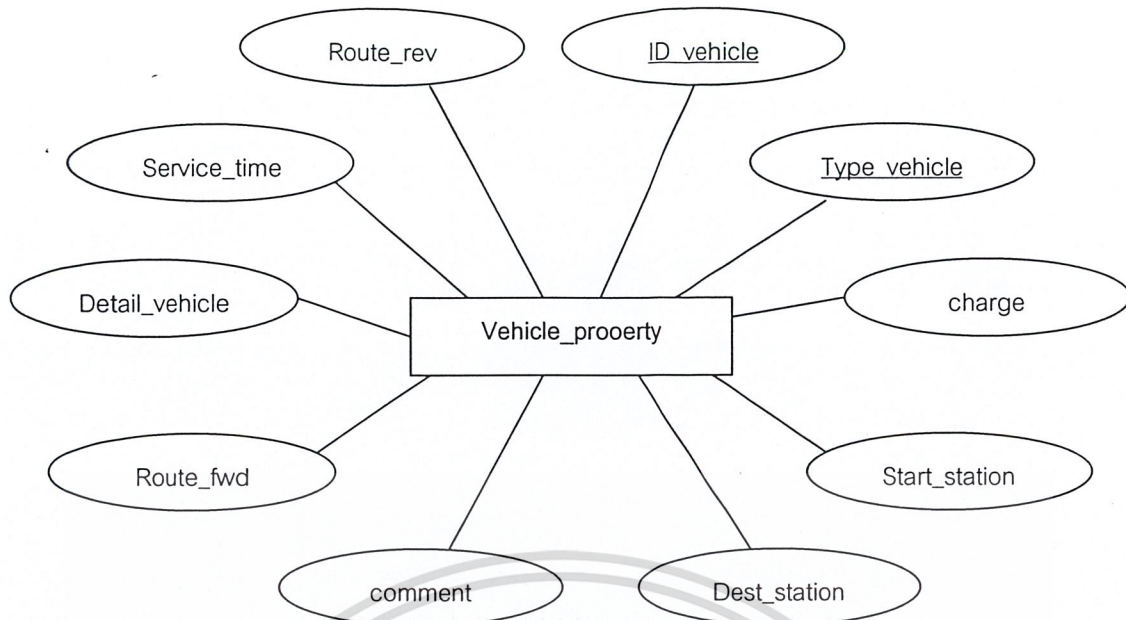
Dest_station หมายถึง สถานีปลายทาง

Route_fwd หมายถึง รายละเอียดเส้นทางเดินรถ ขสมก. ที่ข่วไป

Route_rev หมายถึง รายละเอียดเส้นทางเดินรถ ขสมก. ที่ข่วกลับ

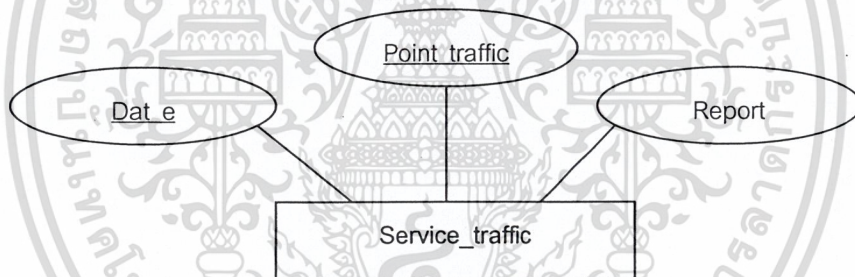
Comment หมายถึง หมายเหตุสำหรับพาหนะต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



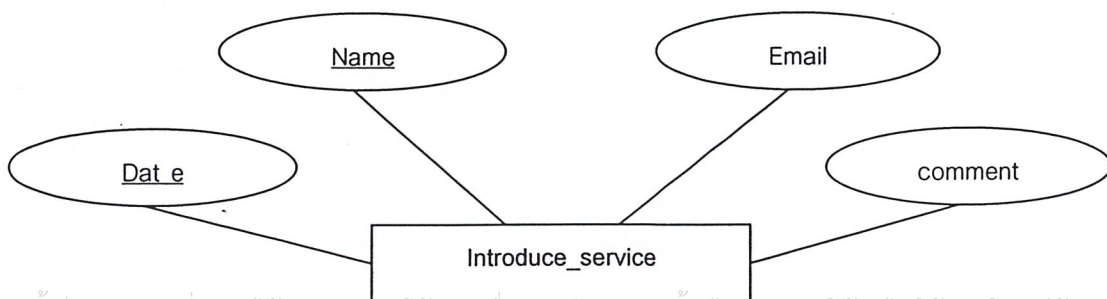
4.3.Entity Traffic_service ประกอบด้วยรายละเอียดดังนี้

- date หมายถึงวันที่รายงานสภาพจราจร
- point_traffic หมายถึง จุดที่รายงานสภาพจราจร
- report หมายถึง รายละเอียดการรายงานสภาพจราจร



4.4.Entity Introduce_service ประกอบด้วยรายละเอียดดังนี้

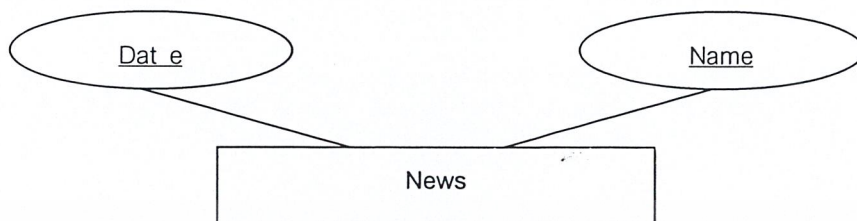
- Date หมายถึง วัน เดือน ปี ที่แนะนำ
- Name หมายถึง ชื่อผู้ที่เข้ามาให้คำแนะนำ
- Email หมายถึง Email Address ผู้ที่เข้ามาให้คำแนะนำ
- Comment หมายถึง รายละเอียดการให้คำแนะนำ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.Entity news ประกอบด้วยรายละเอียดดังนี้

Dat_e	หมายถึง วัน เดือน ปี ที่เอาข่าวสารมาแสดง
News	หมายถึง รายละเอียดข่าวสาร



4.6.ความสัมพันธ์ระหว่างเอนทิตี Place_property และ Vehicle_property ประกอบด้วยรายละเอียด ดังนี้

Time_in	หมายถึง เวลาที่รถไฟเข้าจากสถานี
Time_out	หมายถึง เวลาที่รถไฟออกจากสถานี



5. กำหนดคีย์ของแต่ละเอนทิตีว่า จะใช้รายละเอียดของข้อมูลใดเป็นคีย์หลักของเอนทิตีนั้น ๆ ซึ่งจะ
ต้องเป็นรายละเอียดของข้อมูลที่มีค่าเป็นเอกลักษณ์ หรือค่าเฉพาะไม่ซ้ำซ้อนในเอนทิตีนั้น ๆ

5.1.Entity Place_property มี Pass_place และ Type_place เป็นคีย์หลัก

5.2.Entity Vehicle_property มี ID_vehicle และ Type_vehicle เป็นคีย์หลัก

5.3.Entity Introduce_service มี Dat_e และ Name เป็นคีย์หลัก

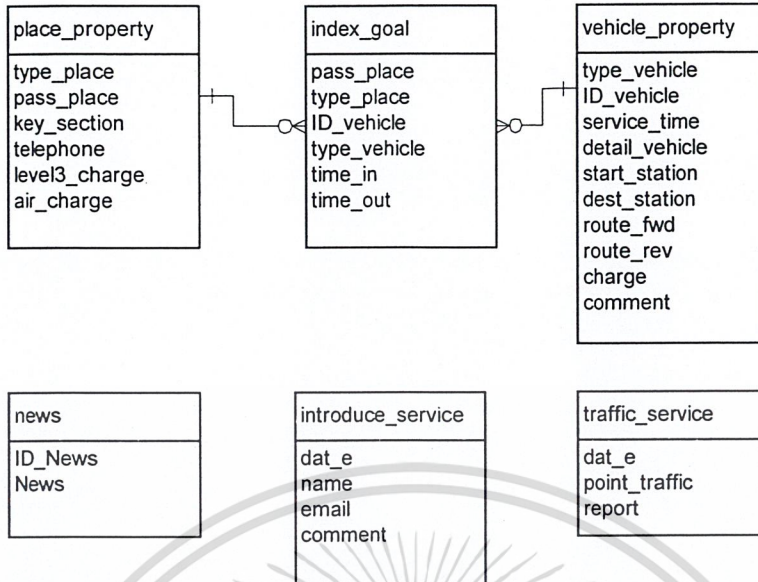
5.4.Entity Traffic_service มี Dat_e และ Point_traffic เป็นคีย์หลัก

5.5.Entity News มี Dat_e และ News เป็นคีย์หลัก

5.6.ความสัมพันธ์ระหว่างเอนทิตี Place_property และ Vehicle_property มี Pass_place และ
Type_place ของ Place_property และมี ID_vehicle และ Type_vehicle ของ Vehicle_property เป็นคีย์หลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. นำรายละเอียดตั้งแต่ขั้นตอนที่ 2 ถึง 5 มาพิจารณาทบทวนอีกครั้ง หลังจากนั้นก็เขียน E-R โมเดลได้ดังนี้



จากรีเลชันที่เราได้จาก E-R โมเดลใหม่ นั้นเราจะมาทำการตรวจสอบและปรับปรุงให้อยู่ในรูปบรรทัดฐาน(Normalization) เพื่อเหตุผล 3 ประการ คือ

1. เพื่อลดเนื้อที่ในการจัดเก็บข้อมูล
2. เพื่อลดปัญหาที่ข้อมูลไม่ถูกต้อง(Inconsistency)
3. เพื่อลดปัญหาที่เกิดจากการเพิ่ม ปรับปรุงและลบข้อมูล (Insert, update and delete anomalies)

โดยมีขั้นตอนดังนี้

1.First Normal Form: 1NF

นิยาม

“ค่าของแอททริบิวต์หนึ่ง ในแต่ละทูเปิลจะมีค่าของข้อมูลเพียงค่าเดียว ”

เราพบว่าฐานข้อมูลที่เราสร้างขึ้นเป็น First Normal Form

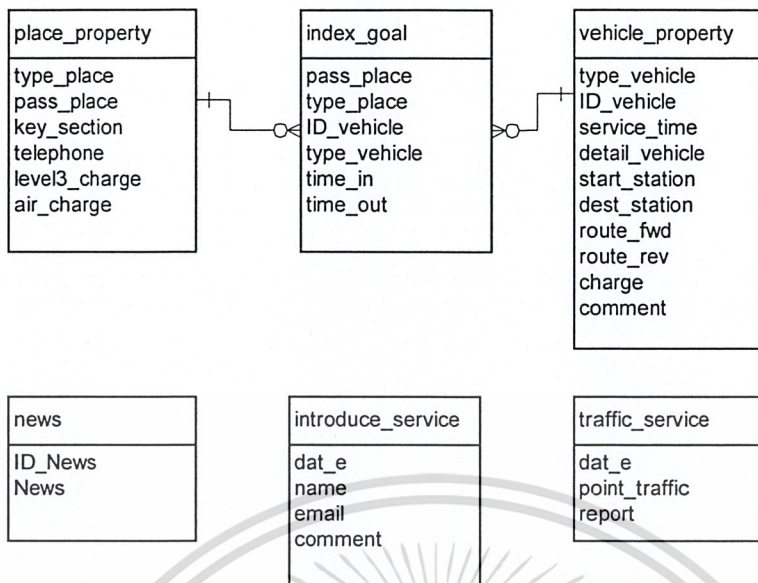
2.Second Normal Form: 2NF

นิยาม

“รีเลชันนั้น ๆ อยู่ในรูปแบบบรรทัดฐานขั้นที่ 1 และมีคุณสมบัติอีกประการหนึ่งคือ แอททริบิวต์ทุกแอททริบิวต์ที่ไม่ได้เป็นคีย์หลัก จะต้องมีความสัมพันธ์ระหว่างค่าของแอททริบิวต์แบบฟังก์ชันกับคีย์หลัก (Fully Functional Dependency) กล่าวอีกนัยหนึ่งคือค่าของแอททริบิวต์ที่ไม่ได้เป็นคีย์หลักจะสามารถระบุค่าโดยแอททริบิวต์ที่เป็นคีย์หลัก หรือโดยแอททริบิวต์ทั้งหมดที่ประกอบกันเป็นคีย์หลักในกรณีที่คีย์หลักเป็นคีย์ผสม” (ไม่มี Partial Dependency เกิดขึ้น)”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากฐานข้อมูลที่เราออกแบบข้างล่างนี้



เราพบว่าฐานข้อมูลที่เราสร้างขึ้นยังไม่เป็น Second Normal Form: 2NF เพราะว่าเราเห็นว่าในตาราง Introduce_service เกิด Partial Dependency ระหว่าง Attribute NAME กับ Email ขึ้นมาเราจึงจัดการแตกให้อยู่ในรูปที่เป็น 2NF ดังนี้

Introduce_service

Dat_e	Name	Comment
-------	------	---------

Introduce_email

Name	Email
------	-------

3.Third Normal Form: 3NF

นิยาม

"รีเลชันนั้น ๆ อยู่ในรูปแบบบรรทัดฐานขั้นที่ 2 และคุณสมบัติอีกประการหนึ่งคือ แอททริบิวต์ที่ไม่ได้เป็นคีย์หลักไม่มีคุณสมบัติในการกำหนดค่าของแอททริบิวต์อื่นที่ไม่ใช่คีย์หลัก (ไม่มี Transitive Dependency เกิดขึ้น)"

เราพบว่าฐานข้อมูลที่เราสร้างขึ้นเป็น Third Normal Form

4. Boyce/Codd Normal Form: BCNF)

นิยาม

"รีเลชันนั้น ๆ อยู่ในรูปแบบบรรทัดฐานขั้นที่ 3 และไม่มีแอททริบิวต์อื่นในรีเลชันที่สามารถระบุค่าของแอททริบิวต์ที่เป็นคีย์หลักหรือส่วนหนึ่งของคีย์หลักในกรณีคีย์หลักเป็นคีย์ผสม"

เราพบว่าฐานข้อมูลที่เราสร้างขึ้นเป็น BCNF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Fourth Normal Form: 4NF

นิยาม

"รีเลชันนั้น ๆ อยู่ในรูปแบบ BCNF และเป็นรีเลชันที่ไม่มีความสัมพันธ์ในการระบุค่าของแอททริบิวต์แบบหลายค่าโดยที่แอททริบิวต์ที่ถูกระบุค่าหลายค่าเหล่านี้ไม่มีความสัมพันธ์กัน (Independently Multivalued Dependency)"

เราพบว่าฐานข้อมูลที่เราสร้างขึ้นเป็น Fourth Normal Form

6. Fifth Normal Form: 5NF

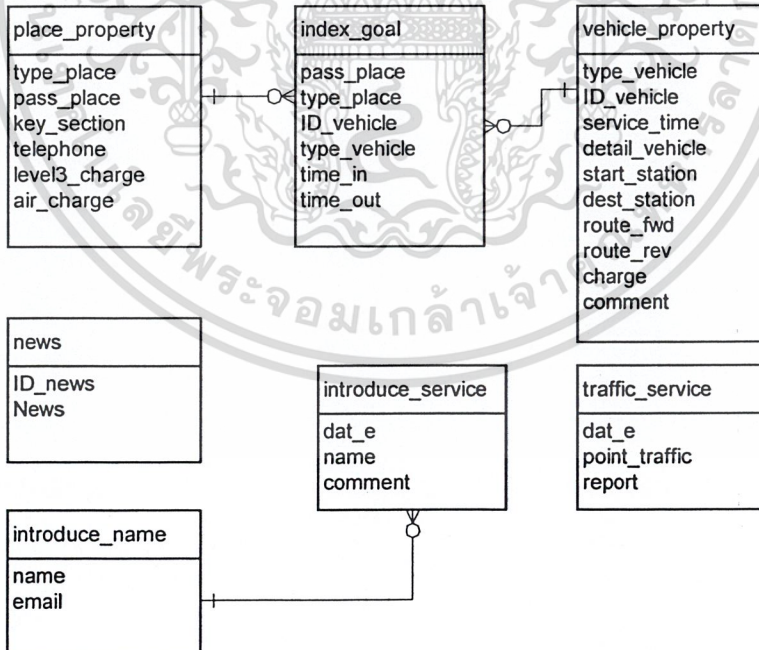
นิยาม

"รีเลชันนั้นอยู่ในรูปแบบบรรทัดฐานขั้นที่ 4 และเป็นรีเลชันที่มีคีย์หลักเป็นคีย์ผสมที่ประกอบด้วยแอททริบิวต์ตั้งแต่สามแอททริบิวต์เป็นต้นไป หากมีการแตกรีเลชันออกเป็นรีเลชันย่อยสามรีเลชันย่อย (หรือมากกว่า) (Projection) ซึ่งเกิดจากการจับคู่แอททริบิวต์แต่ละคู่ของรีเลชันเดิมเป็นคีย์ผสม และเมื่อทำการเชื่อมโยงรีเลชันย่อยทั้งหมด (Join) จะไม่ก่อให้เกิดข้อมูลใหม่ที่ไม่เหมือนรีเลชันเดิม"

เราพบว่าฐานข้อมูลที่เราทำการออกแบบนั้นเป็น 5NF แล้วเพราะ Table ที่ Split ไม่ได้แล้วเราจัดเป็น 5NF

ฐานข้อมูลที่ผ่านขั้นตอน Normalization

ซึ่งเรากลัวว่าเป็นฐานข้อมูลที่สมบูรณ์และเราจะนำมาใช้เป็นฐานข้อมูลของโครงการ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.1.1.2. แปลง E-R Model ให้อยู่ในรูปแบบฐานข้อมูลเชิงสัมพันธ์

เราสามารถแปลงเป็นฐานข้อมูลเชิงสัมพันธ์ในรูปแบบเค้าร่างข้อมูลของแต่ละรีเลชัน ได้ดังนี้

ตาราง Place_property

Type_place	Pass_place	Key_section	Telephone	Level3_charge	Air_charge
------------	------------	-------------	-----------	---------------	------------

ตาราง Vehicle_Property

ID_vehicle	Type_Vehicle	Service_time	Detail_vehicle	Start_station
Dest_station	Route_fwd	Route_rev	Charge	Comment

ตาราง Index_Goal

Pass_place	Type_place	ID_vehicle	Type_Vehicle	Time_in	Time_out
------------	------------	------------	--------------	---------	----------

ตาราง Traffic_service

Dat_e	Point_traffic	Report
-------	---------------	--------

ตาราง Introduce_Service

Dat_e	Name	Comment
-------	------	---------

ตาราง Introduce_name

Name	email
------	-------

ตาราง News

ID_news	news
---------	------

6.3.1.2. การออกแบบคำร้องขอ (Query) ที่ใช้ในการสืบค้นเส้นทางเดินทางที่เป็นไปได้

เราทราบว่า ในการสร้าง Application หรือ Script เพื่อการติดต่อกับ User แบบ Dynamic Page นั้นเราสามารถทำได้ 2 วิธี คือ

- 1.อาศัย ISAPI (Internet Server Application Programming Interface)
- 2.อาศัย CGI (Common Gateway Interface)

ซึ่งเราอาศัยความสามารถในการเข้าถึงฐานข้อมูลของ ISAPI ดังนั้นเราจึงทราบว่าในการที่เราใช้ ISAPI ส่วนที่เราเรียกว่า IDC (Internet Database Connector) ที่มีความสามารถในการติดต่อกับฐานข้อมูลได้นั้นเราจะอาศัยองค์ประกอบ 2 ส่วนก็คือ

1. Internet Database Connector File (.IDC)

2. HTML Extension file (.htx)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของ IDC file นั้นเราจะทำการบรรจุ SQL Statement เพื่อใช้ในการติดต่อกับฐานข้อมูลผ่านตัวจัดการฐานข้อมูล เราจะเริ่มจากการเขียน SQL Statement ที่ต้องใช้ใน IDC file ทั้งหมดก่อนโดยอาศัยรายละเอียดจากการออกแบบรายละเอียดเว็บเพจ ที่เราใช้ในการโต้ตอบกับผู้ใช้บริการ ในการออกแบบคำร้องขอ นั้นเราสามารถจำแนกออกเป็นหัวข้อได้ดังนี้

1. คำร้องขอเพื่อการค้นหาเส้นทางจราจรในกรณีของการใช้พาหนะสาธารณะ

- 1.1. การสืบค้นแบบนั่งรถต่อเดียว
- 1.2. การสืบค้นแบบนั่งรถ 2 ต่อถึงปลายทาง
- 1.3. การสืบค้นแบบนั่งรถ 3 ต่อถึงปลายทาง
- 1.4. การสืบค้นแบบกำหนดจุดหรือสถานที่ที่ต้องการจะเดินทาง

2. คำร้องขอเพื่อการสอบถามและรายงานสภาพจราจร

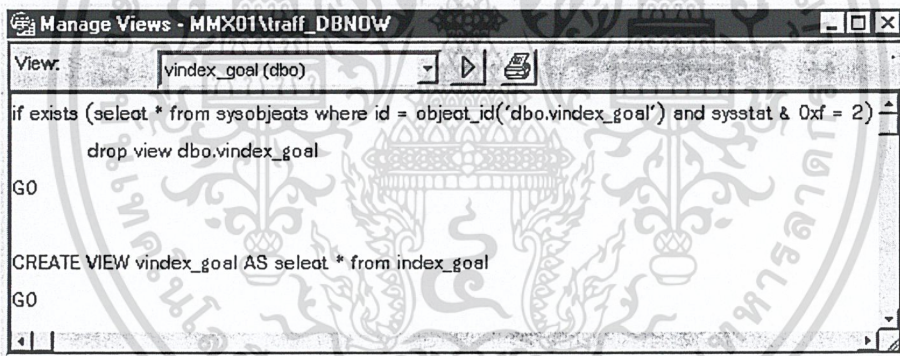
- 2.1. การสอบถามสภาพจราจร
- 2.2. การรายงานสภาพจราจร

3. คำร้องขอเพื่อการเสนอแนะการให้บริการ

สำหรับคำร้องขอเพื่อการค้นหาเส้นทางจราจรในกรณีของการใช้พาหนะสาธารณะนั้นจะอาศัยวิวต่อไปนี้

1. วิวเพื่อการสืบค้นเส้นทางแบบ 1 ต่อถึงปลายทาง จะมี 2 ขั้นตอนคือ

1.1. การสร้างวิวของรีเลชัน Index_Goal ดังนี้



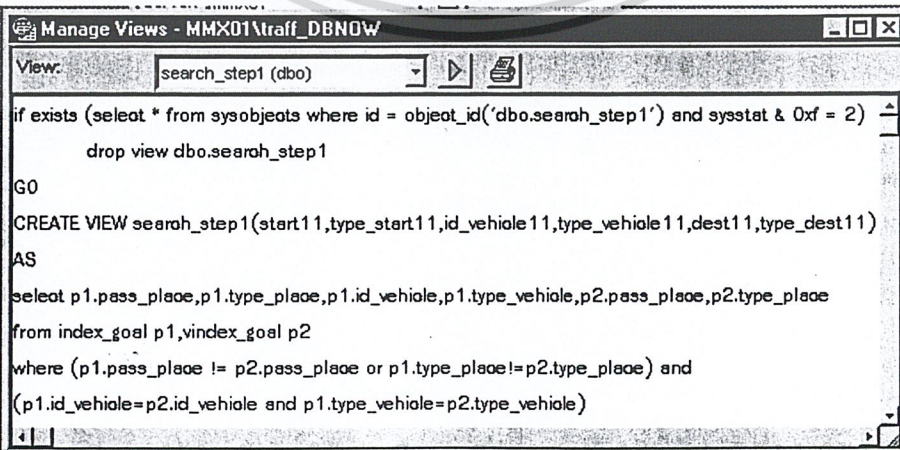
```

View: vindex_goal (dbo)
if exists (select * from sysobjects where id = object_id('dbo.vindex_goal') and sysstat & 0xf = 2)
    drop view dbo.vindex_goal
GO

CREATE VIEW vindex_goal AS select * from index_goal
GO
  
```

รูปที่ 7.6 วิวของรีเลชัน Index_Goal

1.2. สร้างวิวสำหรับการสืบค้น 1 ต่อถึงปลายทาง



```

View: search_step1 (dbo)
if exists (select * from sysobjects where id = object_id('dbo.search_step1') and sysstat & 0xf = 2)
    drop view dbo.search_step1
GO

CREATE VIEW search_step1(start11,type_start11,id_vehiole11,type_vehiole11,dest11,type_dest11)
AS
select p1.pass_place,p1.type_place,p1.id_vehiole,p1.type_vehiole,p2.pass_place,p2.type_place
from index_goal p1,vindex_goal p2
where (p1.pass_place != p2.pass_place or p1.type_place!=p2.type_place) and
(p1.id_vehiole=p2.id_vehiole and p1.type_vehiole=p2.type_vehiole)
  
```

รูปที่ 7.7 วิวสำหรับการสืบค้น 1 ต่อถึงปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูงและขอสงวนสิทธิ์ในการนำข้อมูลไปใช้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. วิิวเพื่อการสืบค้นเส้นทางแบบ 2 ต่อถึงปลายทาง

```

Manage Views - MMX01\traff_DBNOW
View: search_step2 (dbo)
CREATE VIEW search_step2 (start21,type_start21,id_vehiole21,type_vehiole21,pass21,
type_pass21,id_vehiole22,type_vehiole22,dest21,type_dest21)
AS
select p1.start11,p1.type_start11,p1.id_vehiole11,p1.type_vehiole11,p2.start11,p2.type_start11,
p2.id_vehiole11,p2.type_vehiole11,p2.dest11,p2.type_dest11
from search_step1 p1,v_search_step1 p2
where (p1.dest11=p2.start11 and p1.type_dest11=p2.type_start11) and
(p1.id_vehiole11!=p2.id_vehiole11 or p1.type_vehiole11!=p2.type_vehiole11) and
(p1.start11!=p2.dest11 or p1.type_start11!=p2.type_dest11)
GO
  
```

รูปที่ 7.8 วิิวเพื่อการสืบค้นเส้นทางแบบ 2 ต่อถึงปลายทาง

3. วิิวเพื่อการสืบค้นเส้นทางแบบ 3 ต่อถึงปลายทาง

```

Manage Views - MMX01\traff_DBNOW
View: search_step3 (dbo)
CREATE VIEW search_step3(start31,type_start31,id_vehiole31,type_vehiole31,pass31,
type_pass31,id_vehiole32,type_vehiole32,pass32,type_pass32,id_vehiole33,
type_vehiole33,dest31,type_dest31)
AS
select p1.start21,p1.type_start21,p1.id_vehiole21,p1.type_vehiole21,p1.pass21,
p1.type_pass21,p1.id_vehiole22,p1.type_vehiole22,p1.dest21,p1.type_dest21,
p2.id_vehiole11,p2.type_vehiole11,p2.dest11,p2.type_dest11
from search_step2 p1,search_step1 p2
where (p1.start21!=p2.dest11 or p1.type_start21!=p2.type_dest11) and
(p1.pass21!=p2.dest11 or p1.type_start21!=p2.type_dest11) and
(p1.dest21=p2.start11 and p1.type_dest21=p2.type_start11) and
(p1.id_vehiole21!=p2.id_vehiole11 or p1.type_vehiole21!=p2.type_vehiole11) and
(p1.id_vehiole22!=p2.id_vehiole11 or p1.type_vehiole22!=p2.type_vehiole11)
  
```

รูปที่ 7.9 วิิวเพื่อการสืบค้นเส้นทางแบบ 3 ต่อถึงปลายทาง

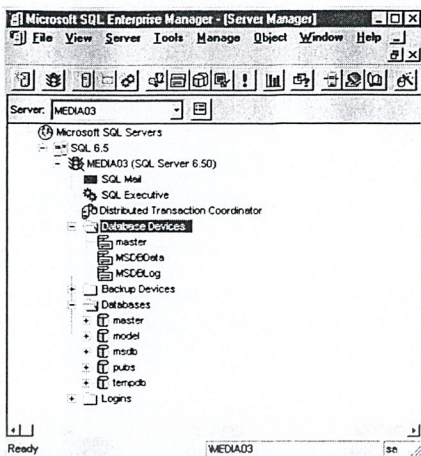
จากนั้นเราจะใช้วิิวที่สร้างขึ้นในการสืบค้นเส้นทาง โดยเราสามารถดูรายละเอียดได้จากไฟล์ IDC ที่เราสร้างขึ้นในการติดต่อกับฐานข้อมูล นอกจากนั้นคำร้องขอสำหรับบริการอื่นๆ เราก็สามารถดูได้จากไฟล์ IDC ที่เราสร้างขึ้น ซึ่งเราแสดงไว้ในภาคผนวกโดยละเอียด

6.3.2. การสร้างฐานข้อมูล

เราเริ่มจากการใช้เครื่องมือตัวแรกคือ Microsoft SQL Server 6.5 มาทำหน้าที่ในการจัดการฐานข้อมูล (Database Management System: DBMS) โดยเราจะเริ่มจากการสร้าง Database Device และ Database ที่อยู่ใน การควบคุมดูแลของ SQL Server ดังนี้

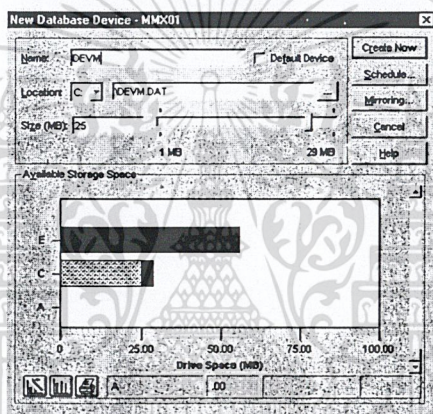
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.เรียก Microsoft SQL Enterprise Manager เพื่อใช้ในการ Create Database Device และ Database



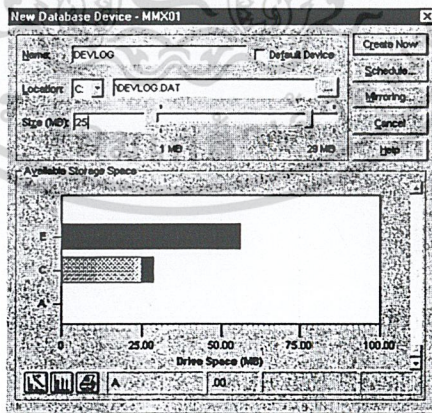
รูปที่ 7.10 หน้าต่าง Microsoft SQL Enterprise Manager

1.1 สร้าง Database Device ชื่อ DEVM.dat ขนาด 25 MB



รูปที่ 7.11 หน้าต่างการสร้าง Database Device

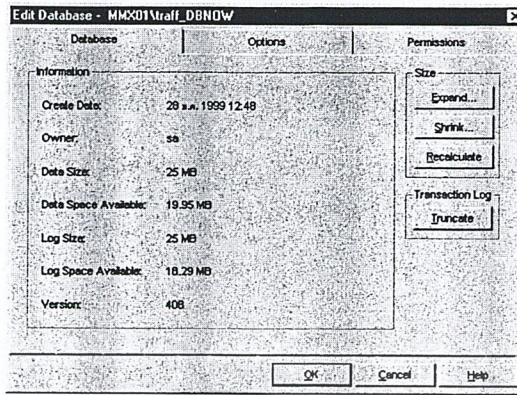
1.2 สร้าง Log Device ชื่อ DEVLOG.dat ขนาด 25 MB



รูปที่ 7.12 หน้าต่างการสร้าง Log Device

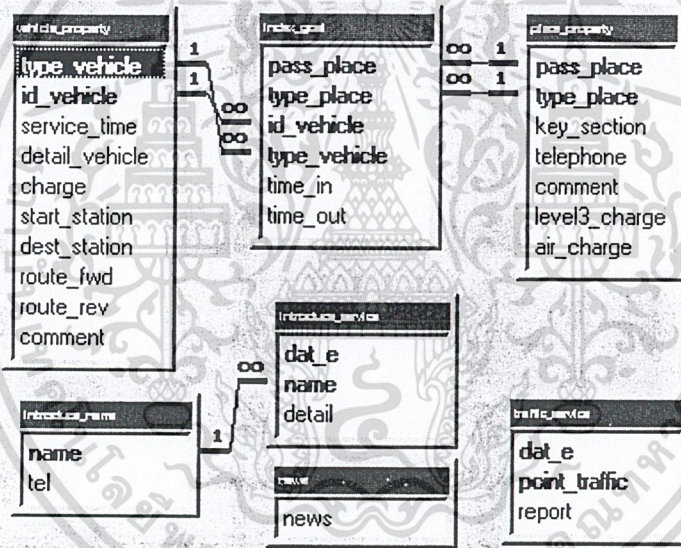
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.สร้าง Database ชื่อ traff_DBNOW ใน DEVLOG ที่เราสร้างขึ้นและทำการเก็บ Log file ใน DEVLOG ดังนี้



รูป 7.13 แสดงการสร้างฐานข้อมูลชื่อ Traffic_DBNOW

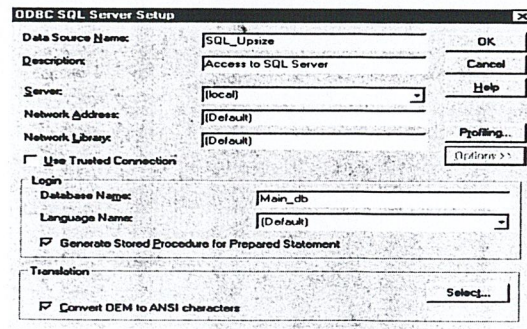
1.4.ทำการสร้าง Table ที่ใช้ในการเก็บข้อมูลต่างๆ ในการสืบค้นเส้นทางจราจรและบริการอื่นโดยอาศัย Microsoft Access 97 ที่มีการเพิ่มโปรแกรม Upsizing Tools ดังนี้



รูปที่ 7.14 ฐานข้อมูลที่จะทำการ Upsize จาก Access ไปที่ SQL Server

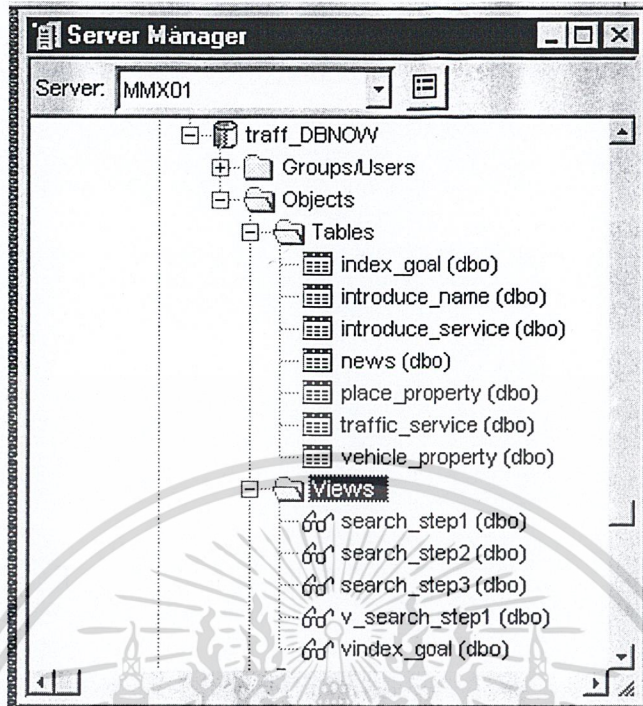
1.5.จากนั้นเราจะทำการ Upsize to SQL Server โดยมีลำดับขั้นตอนดังนี้

1.ทำการ Set ODBC เพื่อเป็นตัวกลางการติดต่อระหว่าง Access กับ SQL Server



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการ **รูปที่ 7.15 การติดตั้ง ODBC** ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. จะได้ฐานข้อมูลที่เราทำการออกแบบดังนี้



รูปที่ 7.16 แสดงตารางและวิวที่เราสร้างขึ้นบน DBMS

6.3.3. สร้างส่วนการติดต่อกับฐานข้อมูลและผู้ใช้บริการ

โฮมเพจของการให้บริการทั้งหมดของโครงการ แสดงดังรูปที่ 7.17

รูปที่ 7.17 โฮมเพจแสดงบริการทั้งหมดของโครงการ

จากรูป เราสามารถแบ่งงานบริการทั้งหมดของเว็บไซต์ออกเป็น 2 ประเภทใหญ่ๆ คือ

1. บริการที่มีการติดต่อกับฐานข้อมูล
2. บริการที่ไม่มีการติดต่อกับฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.3.1. บริการที่มีการติดต่อกับฐานข้อมูล มีรายละเอียดดังนี้

1.บริการสืบค้นเส้นทางจราจรที่เป็นไปได้โดยการกำหนดจุดที่จะเดินทางและข้อมูลอื่นๆ แสดงดังรูปที่ 7.18

รูปที่ 7.18 แสดงเว็บเพจการสืบค้นเส้นทางจราจร

2.บริการสืบค้นรายละเอียดของข้อมูลรถ ขสมก. มีรายละเอียดการให้บริการแสดงดังรูปที่ 7.19

รูปที่ 7.19 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลรถ ขสมก.

3.บริการสืบค้นรายละเอียดของข้อมูลรถไฟ มีรายละเอียดการให้บริการแสดงดังรูปที่ 7.20

รูปที่ 7.20 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลรถไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. บริการสืบค้นรายละเอียดของข้อมูลรถตู้ มีรายละเอียดการให้บริการแสดงดังรูปที่ 7.21

ศูนย์สารสนเทศการจราจร
กรุงเทพมหานคร

กรุณาป้อนรายละเอียด

รายชื่อ: ปากเกร็ด-สะพานใหม่

รูปที่ 7.21 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลรถตู้

5. บริการสืบค้นรายละเอียดของข้อมูลเรือ มีรายละเอียดการให้บริการแสดงดังรูป

ศูนย์สารสนเทศการจราจร
กรุงเทพมหานคร

กรุณาป้อนรายละเอียด

ชื่อ: กรมต หมายเลข 001

ชื่อ: ท่ากรมเจ้าฟ้า

รูปที่ 7.22 แสดงเว็บเพจการสืบค้นรายละเอียดของข้อมูลเรือ

5. รายละเอียดของบริการสอบถามและรายงานสภาพจราจรมีรูปแบบดังรูปที่

สอบถามสภาพการจราจร
กรุงเทพมหานคร

กรุณาป้อนรายละเอียด

สอบถาม จราจร

รายงาน จราจร

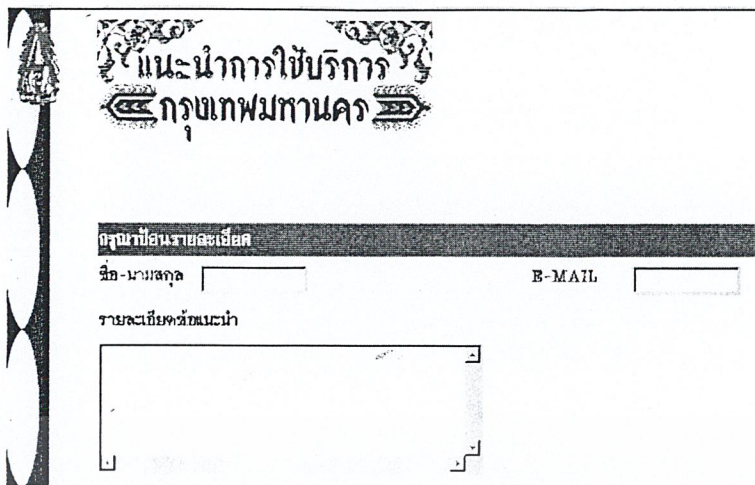
จุดที่มีการรายงาน

สภาพจราจร

รูปที่ 7.23 แสดงเว็บเพจการสืบค้นรายละเอียดของบริการสอบถามและรายงานสภาพจราจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

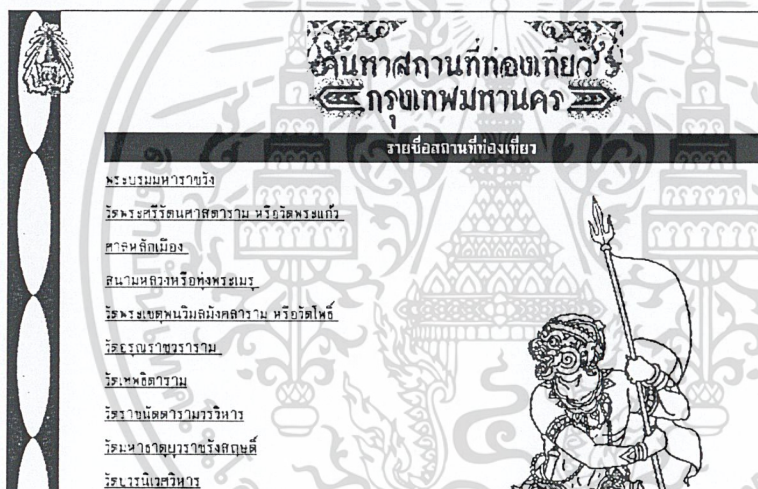
6. รายละเอียดของบริการเสนอแนะการให้บริการของผู้ใช้บริการมีดังรูปที่ 7.25



รูปที่ 7.24 แสดงเว็บเพจการสืบค้นรายละเอียดของบริการเสนอแนะการให้บริการ

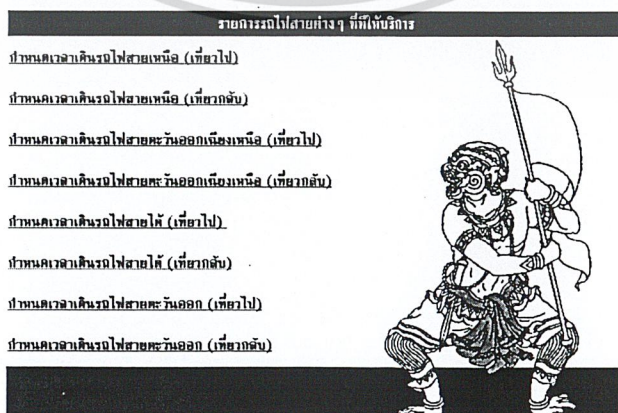
6.3.3.2. บริการที่ไม่มีการติดต่อกับฐานข้อมูล มีรายละเอียดดังนี้

1. บริการสืบค้นสถานที่ท่องเที่ยว มีรายละเอียดการให้บริการแสดงดังรูปที่ 7.25



รูปที่ 7.25 แสดงเว็บเพจการสืบค้นสถานที่ท่องเที่ยว

2. บริการสอบถามตารางรถไฟทั่วประเทศ แสดงรายละเอียดการให้บริการดังรูปที่ 7.26



เอกสารนี้เป็นเอกสารที่รูปที่ 7.25 แสดงเว็บเพจบริการสอบถามตารางรถไฟทั่วประเทศ ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

การทดสอบและผลการทดสอบ

8.1. บทนำ

ในการทดสอบนั้นเราจะทำหลังจากที่ทำการออกแบบ และสร้างฐานข้อมูลที่เราใช้ในการเก็บข้อมูลต่างๆ ในการสืบค้นและบริการต่างๆ ที่สามารถใช้งานได้ ดังที่ได้กล่าวไปแล้วในบทที่ 7 และทำการสร้างส่วนที่ใช้ในการติดต่อระหว่างผู้ใช้บริการอินเตอร์เน็ตกับฐานข้อมูลที่เราสร้างขึ้น

โดยการทดสอบนั้นเราจะทำเพื่อที่จะตรวจสอบว่าสิ่งที่เราทำการออกแบบและตั้งสมมุติฐานไว้ในขั้นตอนของการออกแบบนั้นตรงกับผลการทดสอบหรือไม่ โดยจากการทดสอบเราพบว่าผลลัพธ์ตรงตามที่เรารตั้งไว้ดังจะแสดงในหัวข้อถัดไป

8.2. การทดสอบและผลการทดสอบ




เราสามารถจำแนกการทดสอบตามหัวข้อการให้บริการดังแสดงในบทที่ 7 ดังนี้

8.2.1. ทดสอบสืบค้น โดยการกำหนดจุดค้นหาปลายทาง แสดงดังรูปที่ 8.1

จุดค้นหา	ปลายทาง
คลองตัน	เกษตรศาสตร์
ประเภทของสถานที่	ประเภทของสถานที่
สถานีรถไฟ	มหาวิทยาลัย
รายละเอียดการค้นหา	
<input checked="" type="checkbox"/> ทหารเรือ	ทหารส่วนหัว
<input checked="" type="checkbox"/> รถผสม.	
<input checked="" type="checkbox"/> รถไฟ	
<input checked="" type="checkbox"/> เรือ	
<input checked="" type="checkbox"/> รถตู้	

รูปที่ 8.1 การทดสอบสืบค้นโดยกำหนดจุดค้นหาปลายทาง

ผลการทดสอบและแสดงความเป็นไปได้ของการเดินทางถึงปลายทางแบบ 1,2 และ 3 ต่อ ดังรูปที่ 8.2

-  กรุณา Click ที่ รูปภาพ สำหรับการเดินทางจาก คลองตัน ไปยัง เกษตรศาสตร์ ใช้ 1 ต่อการเดินทาง
-  กรุณา Click ที่ รูปภาพ สำหรับการเดินทางจาก คลองตัน ไปยัง เกษตรศาสตร์ ใช้ 2 ต่อการเดินทาง
-  กรุณา Click ที่ รูปภาพ สำหรับการเดินทางจาก คลองตัน ไปยัง เกษตรศาสตร์ ใช้ 3 ต่อการเดินทาง

ค้นหาทั่วไป

รูปที่ 8.2 แสดงผลการทดสอบความเป็นไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากเราทำการเลือกรายการที่เป็นไปได้ในการเดินทางจะแสดงผลัพท์ ดังรูปที่ 8.3

รายละเอียดการค้นพบ	
สถานที่ต้นทาง	คลองตัน
ประเภทของสถานที่ต้นทาง	สถานีรถไฟ
สถานที่ปลายทาง	เกษตรศาสตร์
ประเภทของสถานที่ปลายทาง	มหาวิทยาลัย

รายละเอียดของยานพาหนะ	
หมายเลขยานพาหนะ	ชนิดของพาหนะ
206	ขสมก
สิ้นสุดข้อมูล	

รูปที่ 8.3 ตัวอย่างการสืบค้นแบบ 1 ต่อถึงปลายทาง

2.การทดสอบการสืบค้นรายละเอียดรถ ขสมก. แสดงดังรูปที่ 8.4

กรุณาป้อนรายละเอียด

หมายเลขรถ

ชื่อสถานที่ ประเภทสถานที่

รูปที่ 8.4 การทดสอบการสืบค้นรายละเอียดรถ ขสมก.

เราจะสังเกตเห็นว่ามีการสืบค้นอยู่ 2 แบบ คือ สืบค้นจากหมายเลขรถประจำทางและชื่อสถานที่ที่เราจะเดินทางไป โดยเราจะแสดงผลการทดสอบทั้งสองแบบดังนี้

-ผลการทดสอบการสืบค้นจากหมายเลขรถ ขสมก. รูปที่ 8.5

รายละเอียด	
หมายเลขรถ	104
เวลาที่ให้บริการ	04.30-23.40

รายละเอียดเส้นทาง ที่ยาวไป	รายละเอียดเส้นทาง ที่ยาวกลับ
เที่ยวไป เริ่มต้นที่ท่าเมืองทองธานี	เที่ยวกลับ
ถนนแจ้งวัฒนะ แยกปากเกร็ด	เริ่มต้นสถานีขนส่งหมอชิตแห่งใหม่
ตรงเข้าท่าหน้าปากเกร็ด วนกลับทางเดิม	ถนนกำแพงเพชร 2 ถนนกำแพงเพชร ถนน
ถนนติวานนท์ ถนนงามวงศ์วาน แยกเกษตร	พหลโยธิน ถนนงามวงศ์วาน แยกคลอง
เลี้ยวขวาถนนพหลโยธิน ถนนกำแพงเพชร	ถนนติวานนท์ เลี้ยวซ้ายเข้าท่าหน้าปากเกร็ด

เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 8.5 ผลการทดสอบการสืบค้นจากหมายเลขรถ ขสมก. ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ผลการทดสอบการสืบค้นจากชื่อสถานที่ รูปที่ 8.6

รายละเอียด	
ชื่อสถานที่	กรเกษม1
ชนิดสถานที่	ชอย

หมายเลขรถ	เวลาให้บริการ	เส้นทาง	ปลายทาง
05	05.30-20.30	คลองกร	ลาดกระบัง
1	04.10-22.00	เขตประเวศ	นาบุญครอง
133	04.00-22.00	การเคหะบางเขน	วัดธาตุทอง
92	04.30-22.00	พหลโยธิน-คลองจั่น	รพ.สงฆ์

สิ้นสุดข้อมูล

ค้นหาทั่วไป

รูปที่ 8.6 ผลการทดสอบการสืบค้นจากชื่อสถานที่

2. การทดสอบสืบค้นรายละเอียดรถไฟ แสดงดังรูปที่ 8.7

กรุณาป้อนรายละเอียด

ชื่อสถานีรถไฟ

หมายเลขขบวนรถไฟ

รูปที่ 8.7 การสืบค้นรายละเอียดรถไฟ

เราจะสังเกตเห็นว่ามีผลการสืบค้นอยู่ 2 แบบคือการสืบค้นจากชื่อสถานีรถไฟและหมายเลขขบวนรถไฟ โดยผลการทดสอบทั้ง 2 แบบ แสดงดังนี้

- ผลการทดสอบการสืบค้นจากชื่อสถานีรถไฟ ดังรูปที่ 8.8

รายละเอียดรถไฟ ขาเข้า/ขาออก	
สถานีที่รถไป	คลองตัน

หมายเลขรถไฟ	ชนิดรถไฟ	เที่ยวเข้า	เที่ยวออก
154	ดีเซลราง	11.32น.	
155	ขบวนเมือง	07.01น.	
156	ขบวนเมือง		08.49น.
180	ขบวนรถ	15.48น.	
181	ดีเซลราง		17.04น.
182	ดีเซลราง	09.39น.	
183	ดีเซลราง		18.00น.
184	ดีเซลราง	20.02น.	
185	ดีเซลราง		18.00น.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเฉพาะเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต
รูปที่ 8.8 ผลการทดสอบการสืบค้นจากชื่อสถานีรถไฟ
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุใดเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ผลการทดสอบการสืบค้นจากหมายเลขรถไฟ ดังรูปที่ 8.9

รายละเอียดรถไฟ	
รถไฟหมายเลข	101
ชนิดรถไฟ	ธรรมดา
เส้นทาง	กรุงเทพ
ปลายทาง	พิษณุโลก

สถานีที่ผ่าน	เวลาออก	ค่าบริการ	จากสถานีกรุงเทพ(หัวลำโพง)
ดอนเมือง	07.51		5
บางซื่อ	07.24		2
บางเขน	07.35		3
รังสิต	08.03		6
หลักสี่	07.43		4
หัวลำโพง			1
ฉะพาน	07.17		2

สิ้นสุดข้อมูล

รูปที่ 8.9 ผลการทดสอบการสืบค้นจากหมายเลขรถไฟ

3. การทดสอบสืบค้นรายละเอียดครูดู แสดงดังรูปที่ 8.10

กรุณาป้อนรายละเอียด

รายชื่อครู

รูปที่ 8.10 การทดสอบสืบค้นรายละเอียดครูดู

ผลการทดสอบการสืบค้นจากชื่อครูดู แสดงดังรูปที่ 8.11

รายละเอียดครูดู	
เส้นทาง	ปากเกร็ด
ปลายทาง	สะพานใหม่
เวลาที่ให้บริการ	
อัตราค่าบริการ	20

สถานีที่ผ่าน	ชนิดของสถานี
หัวลำโพง, บางเขน, บางซื่อ	ศูนย์การค้า
บางพลี, หลักสี่	ท่าเรือ

สิ้นสุดข้อมูล

รูปที่ 8.11 ผลการทดสอบการสืบค้นจากชื่อครูดู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การทดสอบสืบค้นรายละเอียดเรือ แสดงดังรูปที่ 8.12

กรุณาป้อนรายละเอียด

ชื่อ เรือ หมายเลข เรือ

ชื่อ ท่าเรือ

รูปที่ 8.12 การทดสอบสืบค้นรายละเอียดเรือ

เราจะสังเกตเห็นว่ามีการสืบค้นอยู่ 2 แบบคือการสืบค้นจากชื่อเรือและชื่อท่าเรือ โดยผลการทดสอบทั้ง 2 แบบ แสดงดังนี้

- ผลการทดสอบการสืบค้นจากชื่อเรือ ดังรูปที่ 8.13

รายละเอียดเรือ

ต้นทาง	ท่าพิบูลสงคราม(นนทบุรี)
ปลายทาง	ท่าราชวรปุระ
เวลาที่ให้บริการ	

สถานที่ผ่าน	ชนิดของสถานที่
ท่าพิบูลสงคราม	ท่าเรือ
ท่าราชวรปุระ	ท่าเรือ
ท่าเรือ	ท่าเรือ
ท่าเรือ	ท่าเรือ
ท่าเรือ	ท่าเรือ

รูปที่ 8.13 ผลการทดสอบการสืบค้นจากชื่อเรือ

- ผลการทดสอบการสืบค้นจากชื่อท่าเรือ ดังรูปที่ 8.14

รายละเอียดท่าเรือ

ชื่อท่าเรือ	ท่ากรมเจ้าท่า
-------------	---------------

หมายเลขของยานพาหนะ	ชนิดของยานพาหนะ	ต้นทาง	ปลายทาง	เวลาที่ให้บริการ
001	เรือ	ท่าพิบูลสงคราม(นนทบุรี)	ท่าราชวรปุระ	
สิ้นสุดข้อมูล				

รูปที่ 8.14 ผลการทดสอบการสืบค้นจากชื่อท่าเรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การทดสอบการรายงานสภาพจราจรจุดต่างๆ เข้ามาเก็บไว้ในฐานข้อมูลดังรูปที่ 8.15

กรุณาป้อนรายละเอียด

สมทบการจราจร [[นึกไม่ออก]]

รายงานจราจร

จุดที่มีการรายงาน

สภาพจราจร

รูปที่ 8.15 การทดสอบการรายงานสภาพจราจร

ผลการทดสอบแสดงดังรูปที่ 8.16 (ซึ่งหมายถึง ได้ทำการเก็บลงฐานข้อมูลแล้ว)

การทดสอบเรียบร้อยแล้ว

ค้นหาทั่วไป

รูปที่ 8.16 ผลการทดสอบแสดงการเก็บข้อมูลลงฐานข้อมูลแล้ว

6. การทดสอบการสอบถามสภาพจราจรจุดต่างๆ ที่มีการรายงานเข้ามาเก็บไว้ในฐานข้อมูลดังรูปที่ 8.17

กรุณาป้อนรายละเอียด

สมทบการจราจร [นึกไม่ออก]]

รายงานจราจร

จุดที่มีการรายงาน

รูปที่ 8.17 การทดสอบการสอบถามสภาพจราจร

ผลการทดสอบการสอบถามแสดงดังรูปที่ 8.18

จุดที่มีการรายงาน	สภาพการจราจร	เวลาที่รายงานการจราจร
Ladkabang	Traffic jam.....	22:21:42-27:40:42
สิ้นสุดข้อมูล		

ค้นหาทั่วไป

รูปที่ 8.18 ผลการทดสอบการสอบถาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่เราไม่สามารถบอกจุดที่จะสอบถามได้ เราสามารถเรียกดูได้จากฐานข้อมูลดังรูปที่ 8.19

ตารางแสดงจุดที่มีการรายงานสภาพจราจรเข้ามาเก็บในฐานข้อมูล	
วัน/เดือน/ปี	จุดที่มีการรายงานสภาพจราจร
10:2:10:18:1-420	ลาดกระบัง
12:1:5:43:58-420	คูบอน
2:2:19:55:19-420	หนองคาย

รูปที่ 8.19 แสดงความช่วยเหลือกรณีไม่ทราบจุดที่จะสอบถาม

7. อีกข้อหนึ่งที่เราทำการทดสอบก็คือการสืบค้นเส้นทางจากแผนที่ซึ่งเราไม่มีการเก็บลงฐานข้อมูลแสดงดังรูปที่ 8.20

ภาพแผนที่ กลุ่มเทพมหานคร

ในการค้นหาโดยใช้แผนที่นั้น จะต้องใช้ปุ่มต่าง ๆ ดังต่อไปนี้เข้าช่วย

Left	ในการเลื่อนแผนที่ของเขต ทางด้านทิศตะวันตกมาแสดง
Top	ในการเลื่อนแผนที่ของเขต ทางด้านทิศเหนือมาแสดง
Bottom	ในการเลื่อนแผนที่ของเขต ทางด้านทิศใต้มาแสดง
Right	ในการเลื่อนแผนที่ของ เขต ทางด้านทิศตะวันออกมาแสดง

R
2

Left

Top

Bottom

Right

รูปที่ 8.20 การสืบค้นเส้นทางจากแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9

บทสรุปและบทวิจารณ์

จากการดำเนินการพัฒนาโครงการมาตั้งแต่ต้นจนกระทั่งงานเสร็จสมบูรณ์ตามขอบเขตที่เราตั้งขึ้นมานั้น เมื่อนำมาทดสอบการทำงานซึ่งก็ได้ผลตรงกับที่เราต้องการ แต่หากจะกล่าวว่าเป็นงานที่สมบูรณ์และสามารถนำมาใช้งานได้จริงเป็นอย่างคืบหน้า เรายังไม่สามารถยืนยันเช่นนั้นได้ เหตุเพราะโครงการดังกล่าวต้องอาศัยข้อมูลปัจจุบันที่ทันสมัยและละเอียดมาก ซึ่งโครงการที่สร้างขึ้นนั้นยังมีจุดบกพร่องตรงนี้อยู่เนื่องจากข้อมูลที่จะมาสนับสนุนการทำงานของเรานั้นหาข้อมูลที่สมบูรณ์ได้ยากมาก เพราะบางหน่วยงานที่เขามีข้อมูลที่สมบูรณ์ ก็จะสงวนไว้ไม่ยอมเปิดเผยอันเนื่องมาจากผลประโยชน์ทางการค้า

หากเราจะมองหาจุดบกพร่องที่ควรจะทำการพัฒนาให้ดีขึ้นสำหรับผู้สนใจจะพัฒนาต่อไปในอนาคต โดยจุดที่ทางผู้เขียนเห็นว่าควรปรับปรุงมีดังนี้

9.1. ข้อมูลที่เราใช้ในการสืบค้นควรจะมี ความทันสมัยตลอดเวลาเพื่อที่จะให้การสืบค้นมีประสิทธิภาพมากที่สุด เท่าที่จะเป็นไปได้ โดยอาศัยผู้ที่ดูแลระบบเป็นคนคอยปรับปรุงข้อมูลในฐานข้อมูลเมื่อมีการเปลี่ยนแปลง หรืออาจจะอาศัยการตั้งทีมงานขึ้นมารับผิดชอบการปรับปรุงฐานข้อมูล

9.2. ข้อมูลที่เราใช้ในการค้นหาควรจะพยายามหาข้อมูลที่บอกถึงตำแหน่งก่อนหลังของสถานที่ เช่น ระหว่างวัดโพธิ์กับสนามหลวงเราควรจะทราบว่าเมื่อเดินทางด้วยรถประจำทางแล้วจะผ่านสถานที่ใดก่อน และระยะทางระหว่างสถานที่ ซึ่งหากทราบข้อมูลดังกล่าวจะทำให้เราสามารถเขียน Query ที่ใช้ในการสืบค้นเส้นทางที่ยืนยันได้ว่าเป็นเส้นทางที่ดีที่สุด(ทั้งระยะทางและเวลาในการเดินทาง) ซึ่งในโครงการที่เราทำขึ้นนั้นเป็นเพียงการสืบค้นเส้นทางที่เป็นไปได้ระหว่างจุดสองจุดเท่านั้น

9.3. วิธีการติดต่อกับฐานข้อมูลนั้นมี 2 แบบ คือ

- อาศัย ISAPI (Internet Server Application Programming Interface)
- อาศัย CGI (Common Gateway Interface)

สำหรับโครงการที่เราทำขึ้นนั้นเราอาศัย ISAPI ซึ่งมีข้อจำกัดในเรื่องของความยืดหยุ่นในการเข้าถึงฐานข้อมูลและการนำเอาข้อมูลออกมาแสดงผลบนโฮมเพจ หากผู้ที่จะทำการพัฒนาต่อในเรื่องนี้อาจจะลองทำการทดสอบว่าการเขียน CGI นั้นมีการทำงานที่ทำให้โครงการมีประสิทธิภาพมากขึ้นหรือไม่ โดยทำการเปรียบเทียบจุดดีจุดเสียของแต่ละวิธี เพราะการอาศัย ISAPI ก็มีจุดดีที่เหมาะสมกับการทำโครงการนี้ เพราะมันมีความสามารถที่จะรองรับการทำงานของการทำงานของการเข้าถึงฐานข้อมูลได้ระดับหนึ่ง แต่หากเห็นว่าเมื่อมีการปรับปรุงฐานข้อมูลและต้องการเพิ่มความสามารถบางอย่างที่ CGI สามารถทำงานได้ดีกว่าก็อาจเปลี่ยนมาใช้ CGI แทนเพื่อการทำงานที่มีประสิทธิภาพมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.4. นอกจากนี้ส่วนที่เราควรรีความสนใจอีกประการหนึ่งก็คือการบริหารเว็บไซต์ให้มีความทำงานที่มีประสิทธิภาพเพื่อรองรับการเข้ามาใช้บริการบนเครือข่ายอินเทอร์เน็ต

9.5. รูปแบบการติดต่อกับผู้ใช้บริการควรปรับปรุงให้ดูน่าใช้งานตามสมัยนิยมให้มากที่สุด ดังนั้นควรมีการสำรวจความต้องการของผู้ใช้บริการว่ามีความต้องการอะไรเพิ่มเติมหรือต้องการเปลี่ยนแปลงรูปแบบการสืบค้นที่ผู้ใช้ต้องการมากที่สุด ซึ่งจะทำให้เว็บไซต์ดูน่าสนใจตลอดเวลา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. อ.บัณฑิต จามรภูติ, การใช้งาน Microsoft SQL Server .-กรุงเทพฯ : ว.พีเชรสกุล, 2541, 384 หน้า
2. อรรถพ ขันธิกุล, กรภัทร์ สุทธิคารา, ตังจะ จรัสรุ่งรวิวรร, Windows NT 4.0 Server & Workstation สำหรับผู้
บริหารระบบ, บริษัท ดวงกมลสมย์ จำกัด, 460 หน้า
3. อ.กิตติ ภัคดีวัฒน์กุล, สร้าง Web Page แบบมืออาชีพด้วย HTML, บริษัท ดวงกมลสมย์ จำกัด, 321 หน้า
4. รองศาสตราจารย์ ศิริลักษณ์ โจรนกิจอำนวย, SQL, บริษัทสามัคคีสาร(ดอกหญ้า) จำกัด (มหาชน), 152 หน้า
5. Mark Swank, Drew Kittel, Mark Spenik, World Wide Web Database Developer's Guide, Copyright 1997
by Sams.net Publishing First Edition
6. พ.อ.เจนวิทย์ เหลืองอร่าม, การเขียนโปรแกรมสำหรับ Application และ Applets สำหรับ JAVA, บริษัท ซี
เอ็ดยูนิเทค จำกัด (มหาชน)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้