

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา  
เพื่อสนับสนุนระบบการดำเนินงานข้อมูล

TEMPORAL DATABASE APPLICATION DEVELOPMENT FOR  
THE DATABASE REPLICATION



จพ. โน  
ร ๑๑๗  
๒๖๕๕

เลขหมู่...  
เลขทะเบียน... 131409  
วัน,เดือน,ปี... 2 ส.ย. 2557

b. 126๐๙๓๘๙  
i. ....

รายงานนี้เป็นส่วนหนึ่งของวิชาการศึกษาระดับ 2

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ  
คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**PROPOSAL OF TEMPORAL DATABASE APPLICATION  
DEVELOPMENT FOR THE DATABASE REPLICATION**



**A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS OF THE COURSE INDEPENDENT STUDY 2  
MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY  
FACULTY OF INFORMATION TECHNOLOGY  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
2/2012



**COPY RIGHT 2013**

**FACULTY OF INFORMATION TECHNOLOGY**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
**KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKARBANG**  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา  
เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล

นักศึกษา นางสาวรัตนภรณ์ นพรัตน์

รหัสนักศึกษา 54660504

ปริญญา วิทยาศาสตร์มหาบัณฑิต

สาขาวิชา เทคโนโลยีสารสนเทศ

แขนงวิชา เทคโนโลยีระบบสารสนเทศ

ปีการศึกษา 2555

อาจารย์ที่ปรึกษา ดร. สิงหะ ฉวีสุข

### บทคัดย่อ

ปัจจุบันข้อมูลนั้นถือว่าเป็นสิ่งสำคัญต่อการพัฒนาองค์กรต่างๆ การทำซ้ำข้อมูลของตารางสัมพันธ์ไว้ในหลายๆ จุด เพื่อประสิทธิภาพในการทำงานเพิ่มขึ้น เนื่องจากสามารถทำงานได้ ณ จุดย่อยที่เรียกใช้ได้เลย และเป็นการประหยัดการติดต่อสื่อสารระหว่างจุดด้วย นอกจากนี้ ยังเป็นการเพิ่มสภาพความพร้อมในการใช้งานด้วย เมื่อฐานข้อมูลหลักเกิดขัดข้อง ข้อมูลชุดเดียวกัน ณ จุดย่อยอื่นก็ยังสามารถนำมาใช้งานแทนกันได้ การสำเนาข้อมูลในการใช้งานจำนวนมากที่ต้องการเก็บข้อมูลที่เปลี่ยนแปลงโดยอ้างอิงกับเวลา ฐานข้อมูลนั้นสามารถเก็บข้อมูลได้ทั้งที่เป็นอดีต ปัจจุบัน และ อนาคต ซึ่งผู้ดูแลฐานข้อมูลใช้ในการวางแผนจัดการ การตรวจสอบครบถ้วนและถูกต้องของข้อมูลที่เก็บสำเนาไว้ การพัฒนาระบบนี้มีวัตถุประสงค์เพื่อนำเสนอแนวคิดเกี่ยวกับฐานข้อมูลเชิงเวลา และได้นำเสนอขั้นตอนในการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา ซึ่งเป็นการนำคุณสมบัติของฐานข้อมูลเชิงเวลามาช่วยในการเก็บข้อมูลการปรับปรุงตารางสัมพันธ์ โดยใช้ระบบจัดการฐานข้อมูลไมโครซอฟท์ เอสคิวเอลเซิร์ฟเวอร์มาพัฒนาระบบสนับสนุนฐานข้อมูลเชิงเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Title** Proposal of Temporal Database Application Development for  
The Database Replication

**Student** Ms. Rattanaporn Nopparat

**Student ID.** 54660504

**Degree** Master of Science

**Program** Information Technology

**Major** Information System Technology

**Academic Year** 2012

**Advisor** Dr.Singha Chaveesuk

## ABSTRACT

The data is important for organization development. Replication of multiple tables can be related to the performance increase. The backup site can operate at run time and save the contact point between the state and also increase the use of them. In case of the current production database crash due to Operating System or Primary disk failure. At the same data set, it can be used interchangeably. The many copies of real-time application data need to recover in a short time reference. Database that can store all the past, present and future, the database administrator in the management schedule plan to determine completeness and accuracy of data stored copy. This system aims to present Temporal Database concept, step of Temporal Database application development.. The feature of database is to assist in data collection to improve the relation which is replicated to backup site. Using a database management system is Microsoft SQL Server database system.

# สารบัญ

บทคัดย่อภาษาไทย .....	I
บทคัดย่อภาษาอังกฤษ .....	II
กิตติกรรมประกาศ.....	III
สารบัญ .....	IV
สารบัญตาราง .....	VI
สารบัญรูป .....	VII
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ .....	2
1.3 เป้าหมายและขอบเขตของโครงการ.....	3
1.4 ขั้นตอนการพัฒนาระบบ .....	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	4
1.6 ระยะเวลาในการดำเนินงาน.....	5
บทที่ 2 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง.....	6
2.1 การพัฒนาระบบเชิงวัตถุ.....	6
2.2 การออกแบบฐานข้อมูล.....	12
2.3 การพัฒนาเว็บแอปพลิเคชัน .....	14
2.4 ฐานข้อมูลเชิงเวลา .....	18
บทที่ 3 การวิเคราะห์ระบบงานปัจจุบัน .....	24
3.1 สถาปัตยกรรมระบบฐานข้อมูลแบบกระจาย.....	24
3.2 ชนิดของระบบฐานข้อมูลแบบกระจาย .....	28
3.3 รูปแบบการจัดเก็บข้อมูลในระบบฐานข้อมูลแบบกระจาย.....	29
3.4 กระบวนการสืบค้นข้อมูลแบบกระจาย .....	35
3.5 การทำสำเนาฐานข้อมูลในไมโครซอฟท์ เอสคิวเอลเซิร์ฟเวอร์ .....	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศีกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

3.6 ปัญหาของระบบงานปัจจุบัน.....	43
3.7 คุณลักษณะของเครื่องมือตรวจสอบการสำเนาฐานข้อมูล .....	43
3.8 แนวทางการแก้ไขปัญหา.....	44
บทที่ 4 การวิเคราะห์และออกแบบระบบใหม่.....	48
4.1 การวิเคราะห์ความต้องการของระบบ.....	48
4.2 ขอบเขตการทำงานของระบบใหม่ .....	49
4.3 การวิเคราะห์และออกแบบระบบงาน .....	49
4.3.1 ยูสเคสไดอะแกรม .....	49
4.3.2 แอกทิวิตีไดอะแกรม.....	59
4.3.3 อีอาร์ไดอะแกรม.....	63
4.3.4 พจนานุกรมข้อมูล .....	65
4.3.5 คลาสไดอะแกรม.....	69
บทที่ 5 สรุปผลการพัฒนาและข้อเสนอแนะ.....	70
5.1 สรุปผลการพัฒนา.....	70
5.2 ปัญหาและอุปสรรค.....	70
5.3 ข้อเสนอแนะ .....	71
บรรณานุกรม.....	72
ภาคผนวก .....	73
ภาคผนวก ก การติดตั้งเครื่องมือในการพัฒนา .....	73
ประวัติผู้เขียน .....	84

## สารบัญตาราง

ตารางที่	หน้า
1.1 แสดงระยะเวลาในการดำเนินงาน.....	5
3.1 ความสัมพันธ์ของช่วงเวลา.....	20
3.1 เงื่อนไขการแบ่งแยกข้อมูลตามแนวนอน.....	31
3.2 ผลลัพธ์การแบ่งแยกข้อมูลตามแนวนอน.....	31
3.3 เงื่อนไขการแบ่งแยกข้อมูลตามแนวตั้ง.....	32
3.4 ผลลัพธ์การแบ่งแยกข้อมูลตามแนวตั้ง.....	33
3.5 เงื่อนไขการแบ่งแยกข้อมูลแบบผสม.....	34
3.6 ผลลัพธ์การแบ่งแยกข้อมูลแบบผสม.....	34
3.7 การจอยแบบขนาน.....	39
3.8 ตารางแสดงแนวทางการแก้ไขปัญหา.....	46
4.1 คำอธิบายยูสเคสจัดการรายละเอียดการสำเนาข้อมูล.....	51
4.2 คำอธิบายยูสเคสคูสติชองกิจกรรมการใช้งาน.....	50
4.3 คำอธิบายยูสเคสเปรียบเทียบฐานข้อมูลที่สำเนา.....	55
4.4 คำอธิบายยูสเคสนำเข้าข้อมูล โครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ใช้ในการสำเนา.....	56
4.5 คำอธิบายยูสเคสนำเข้าข้อมูลการใช้งานฐานข้อมูลที่สำเนา.....	58
4.6 คำอธิบายยูสเคสบันทึกข้อมูลการใช้งานฐานข้อมูลที่สำเนา.....	59
4.7 แสดงรายละเอียดของฐานข้อมูลของตาราง CHANGELOG.....	65
4.8 แสดงรายละเอียดของฐานข้อมูลของตาราง TBDISTRIBUTION_AGENTS.....	65
4.9 แสดงรายละเอียดของฐานข้อมูลของตาราง TBDISTRIBUTION_HISTORY.....	66
4.10 แสดงรายละเอียดของฐานข้อมูลของตาราง TBPUBLICATION_INFO.....	66
4.11 แสดงรายละเอียดของฐานข้อมูลของตาราง TBPUBLICATIONS.....	67
4.12 แสดงรายละเอียดของฐานข้อมูลของตาราง TBREPLICATION_MONITORDATA.....	67
4.13 แสดงรายละเอียดของฐานข้อมูลของตาราง TBSUBSCRIBER_INFO.....	68
4.14 แสดงรายละเอียดของฐานข้อมูลของตาราง TBSUBSCRIBER_SCHEDULE.....	68
4.15 แสดงรายละเอียดของฐานข้อมูลของตาราง TBSUBSCRIPTIONS.....	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศรัทธาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

รูปที่	หน้า
2.1	แผนภาพเอกทวิติแบบทั่วไป..... 9
2.2	แผนภาพเอกทวิติแบบมีทางเลือกให้ตัดสินใจ ..... 9
2.3	แผนภาพเอกทวิติแบบที่มีการทำงานพร้อมกันหลายงาน ..... 9
2.4	แผนภาพเอกทวิติแบบแสดงการส่งสัญญาณ ..... 10
2.5	แผนภาพสัญลักษณ์ของอ็อบเจกต์ในซีควเอนซ์ไดอะแกรม ..... 11
2.6	แผนภาพสัญลักษณ์ของสเตทชาร์ต ไดอะแกรม ..... 12
2.7	แผนภาพการจัดเก็บข้อมูลของระบบฐานข้อมูลปกติ ..... 18
2.8	แผนภาพการจัดเก็บข้อมูลของระบบฐานข้อมูลเชิงเวลา ..... 18
2.9	แผนภาพ Entity – Relationship Diagram ฐานข้อมูลตัวอย่าง ..... 21
3.1	ฐานข้อมูลแบบกระจาย ..... 25
3.2	แพลตฟอร์ม ระบบการจัดการฐานข้อมูล ระบบปฏิบัติการและระบบเครือข่าย ..... 29
3.3	การจัดเก็บข้อมูลด้วยวิธีการทำสำเนา..... 30
3.4	ที่อยู่ของตารางข้อมูล ..... 37
3.5	การสำเนาข้อมูลไปไว้ที่สาขา Si ..... 37
3.6	การสำเนาข้อมูลที่ละสาขา ..... 38
3.7	แสดงบทบาทของเครื่องบริการในการสำเนาฐานข้อมูล ..... 40
3.8	ขั้นตอนการทำสำเนาฐานข้อมูลในปัจจุบัน ..... 45
4.1	ยูสเคสไดอะแกรมของระบบ ..... 50
4.2	แสดงหน้าจอการทำงานการจัดการรายละเอียดการสำเนาข้อมูล..... 53
4.3	แสดงหน้าจออ็อบเจกต์บนฐานข้อมูลที่ถูกปรับปรุง ..... 57
4.4	แสดงหน้าจอการเปรียบเทียบฐานข้อมูลที่ทำสำเนา..... 57
4.5	แสดงเอกทวิติไดอะแกรมของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล ..... 60
4.6	แสดงเอกทวิติไดอะแกรมของการจัดการรายละเอียดการสำเนาข้อมูล ..... 60
4.7	แสดงเอกทวิติไดอะแกรมของการนำเข้าสู่ข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา..... 61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

รูปที่	หน้า
4.8	แสดงเอกทิวทัศน์โดยะแกรมของการเปรียบเทียบฐานข้อมูลที่ทำสำเนา..... 62
4.9	แสดงเอกทิวทัศน์โดยะแกรมของการดูสถิติของกิจกรรมการใช้งาน..... 62
4.10	แสดงเอกทิวทัศน์โดยะแกรมของการนำเข้าข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา..... 63
4.11	แสดงคลาสโดยะแกรมของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล..... 63



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการ VIII เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันข้อมูลนั้นถือว่าเป็นสิ่งสำคัญต่อการพัฒนาองค์กรต่างๆ จึงมีโปรแกรมประยุกต์จำนวนมากที่ต้องการเก็บข้อมูลอ้างอิงกับเวลา (ยกตัวอย่างเช่น ระบบสนับสนุนการตัดสินใจของผู้บริหาร, ระบบคลังข้อมูล เป็นต้น) โดยที่คำว่าข้อมูลอ้างอิงกับเวลานั้นหมายถึง ฐานข้อมูลนั้นสามารถเก็บข้อมูลได้ทั้งที่เป็น อดีต ปัจจุบัน และ อนาคต เช่น ข้อมูลกำหนดการหรือข้อมูลที่มีการวางแผนไว้ล่วงหน้า เทคโนโลยีการสื่อสารและเครือข่ายคอมพิวเตอร์ที่ก้าวหน้าและแพร่หลายในทุกองค์กร ทำให้การเรียกใช้ฐานข้อมูลจากหลาย ๆ ที่ตั้งสามารถทำได้

การทำงานของระบบฐานข้อมูลบนระบบเครือข่ายสามารถที่จะเชื่อมต่อและกระจายข้อมูลระหว่างเครื่องให้บริการฐานข้อมูลได้ โดยวิธีการที่เรียกว่า การทำสำเนาฐานข้อมูลหรือการทำสำเนาฐานข้อมูล(Database Replication) ไปยังเครื่องให้บริการฐานข้อมูลเครื่องอื่นๆที่ตั้งอยู่ห่างไกล ด้วยความสามารถที่จะสำเนาฐานข้อมูล ทำให้มีความเป็นอิสระ การสำเนาฐานข้อมูลจะต้องมีองค์ประกอบคอยทำงานรวมกัน

1. ฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) เป็นฐานข้อมูลหลักที่ใช้เป็นแหล่งสำเนาฐานข้อมูลเพื่อส่งไปยังตัวกลาง (Distributor) ซึ่งอยู่บนเครื่องให้บริการฐานข้อมูลเดียวกันหรือต่างกันได้ โดยที่ฐานข้อมูลที่ใช้ในการทำสำเนาฐานข้อมูลหนึ่ง สามารถจะสำเนาฐานข้อมูลให้กับหลายฐานข้อมูลที่รับการสำเนาฐานข้อมูล(Subscriber)ได้

2. ตัวกลาง (Distributor) เป็นตัวที่ทำหน้าที่คอยรับข้อมูลที่มีการเปลี่ยนแปลงจากฐานข้อมูลที่ใช้ในการทำสำเนาฐานข้อมูล เพื่อส่งต่อไปให้กับฐานข้อมูลที่รับการสำเนาฐานข้อมูลอีกทีหนึ่ง ซึ่งบนระบบขนาดเล็กนั้น ฐานข้อมูลที่รับการสำเนาฐานข้อมูลจะอยู่บนเซิร์ฟเวอร์เครื่องเดียวกันกับฐานข้อมูลที่ใช้ในการทำสำเนาฐานข้อมูล แต่ถ้าเป็นระบบใหญ่ ตัวกลางจำทำงานจะทำงานอยู่บนเครื่องของตัวเอง

3. ฐานข้อมูลที่รับการสำเนาฐานข้อมูล (Subscriber) เป็นตัวที่รับข้อมูลจากการสำเนาจากฐานข้อมูลที่ใช้ในการทำสำเนาฐานข้อมูล ซึ่งตัวฐานข้อมูลที่รับการสำเนาจะมีวิธีการรับสำเนาข้อมูลได้ 2 แบบคือ แบบผลัก (Push) และแบบดึง (Pull) กล่าวคือ ถ้าเป็นแบบดึงนั้น ตัวฐานข้อมูลที่ใช้ในการทำสำเนาฐานข้อมูล จะส่งข้อมูลที่สำเนาข้อมูลไปยังตัวฐานข้อมูลที่รับการสำเนาฐานข้อมูล โดยไม่ต้องมีการร้องขอ ส่วนในแบบดั่งนั้น ตัวฐานข้อมูลที่รับการสำเนาฐานข้อมูลจะเป็นตัวดึงข้อมูล

เอกสารนี้จากตัวฐานข้อมูลที่ใช้ในการทำสำเนาฐานข้อมูลมาได้ โดยผ่านการร้องขอก่อน ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. โครงสร้างข้อมูลที่จะทำการสำเนา (Articles) เป็นเสมือนตารางเดี่ยวๆ หรือส่วนหนึ่งของข้อมูลในตาราง (Table) โดยปกติแล้วโครงสร้างข้อมูลที่จะทำสำเนาจะเก็บอยู่ในส่วนเก็บโครงสร้างข้อมูลที่จะทำสำเนา (Publication) การเผยแพร่ส่วนโครงสร้างข้อมูลที่จะทำสำเนาจะต้องใช้การกรองข้อมูล (Filtering Data) ภายในตาราง เพื่อส่งให้ฐานข้อมูลที่รับการสำเนาฐานข้อมูลต่อไป

5. ส่วนเก็บโครงสร้างข้อมูลที่จะทำสำเนา (Publication) เป็นส่วนที่ใช้เก็บโครงสร้างข้อมูลที่จะทำสำเนาหรือกลุ่มของข้อมูลที่ต้องการทำสำเนา รวมทั้ง กระบวนการคำสั่งที่เตรียมไว้ (Stored procedures) ซึ่งบรรจุข้อมูลในการทำสำเนาฐานข้อมูลจากฐานข้อมูลที่ใช้ในการทำสำเนาฐานข้อมูลไปยังฐานข้อมูลที่รับการสำเนาฐานข้อมูล

การพัฒนาแบบนี้มีจุดประสงค์ที่จะพัฒนาระบบพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล โดยการเก็บสถิติแต่ละครั้งการเปลี่ยนแปลงตารางความสัมพันธ์และโครงสร้างข้อมูลที่ต้องการทำสำเนาในฐานข้อมูล เมื่อนำข้อมูลสถิติที่มีอยู่ไปใช้ในการวางแผน ระบบจะเป็นส่วนที่ช่วยตรวจสอบและแสดงการปรับปรุงข้อมูลจะเลือกเป็นช่วงเวลาที่ไม่มีผู้ใช้งานฐานข้อมูลนั้น เพื่อไม่ให้ผู้อื่นได้รับผลกระทบในระหว่างที่ทำการเก็บสถิติการเปลี่ยนแปลง ในการวางแผนแผนงาน เป็นการเลือกวิธีที่ใช้ในการเข้าถึงข้อมูลอย่างมีประสิทธิภาพมากที่สุด นอกจากจะใช้ข้อมูลสถิติในฐานข้อมูลมาใช้เป็นส่วนตัดสินใจเลือกวิธีแล้ว สิ่งสำคัญที่ต้องคำนึงถึงคือปริมาณข้อมูลที่ต้องส่งผ่านเครือข่าย เนื่องจากความเร็วของระบบเครือข่ายมีค่าน้อยกว่าความเร็วของดิสก์หลายเท่า ดังนั้นก่อนที่จะส่งข้อมูลจากฐานข้อมูลใดๆ ควรที่จะมีการคัดกรองให้เหลือเฉพาะข้อมูลที่ต้องการจริงๆ เท่านั้น ในการทำสำเนาฐานข้อมูลอาจจะส่งผลกระทบต่อประสิทธิภาพของระบบในระหว่างการทำงานได้ จึงจำเป็นต้องวางแผนและออกแบบระบบที่เหมาะสม

## 1.2 วัตถุประสงค์ของการพัฒนาระบบ

การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล เป็นโครงการที่จัดทำขึ้นเพื่อตรวจสอบและแสดงการเปลี่ยนแปลงโครงสร้างข้อมูลที่ใช้ในการทำสำเนาข้อมูลขององค์กร ช่วยการปรับปรุงข้อมูลให้มีความทันสมัยอยู่เสมอ ลดความขัดแย้งของข้อมูลระหว่างเครื่องให้บริการฐานข้อมูล โครงการพัฒนาระบบขึ้นเพื่อวัตถุประสงค์ดังนี้

1. เพื่อศึกษาถึงลักษณะ วิธีการทำงาน ข้อดีและข้อจำกัดของการทำสำเนาฐานข้อมูล เพื่อหาแนวทางในการนำมาพัฒนา โปรแกรมประยุกต์กับระบบการสำเนาฐานข้อมูล
2. เพื่อนำฐานข้อมูลเชิงเวลามาใช้ในการเก็บสถิติในระบบการสำเนาฐานข้อมูล โดยให้ตรวจสอบโครงสร้างความสัมพันธ์และแสดงเวลาที่เหมาะสมในการเก็บสถิติ
3. เพื่อนำระบบมาใช้ในการวางแผนบริหารการทำสำเนาฐานข้อมูล โดยให้โปรแกรมประยุกต์ทำหน้าที่ที่วิเคราะห์คำถามย่อยจากฐานข้อมูลต่างๆ ที่เครื่องฐานข้อมูลแต่ละแห่ง เพื่อเปรียบเทียบความสมบูรณ์ของข้อมูลระหว่างฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) กับฐานข้อมูลที่รับการทำสำเนาฐานข้อมูล (Subscriber)
4. เพื่อวัดประสิทธิภาพการทำงานของระบบสำเนาฐานข้อมูล ในการใช้ตรวจสอบและแสดงการทำงานด้วยฐานข้อมูลเชิงเวลา มีส่วนช่วยเพิ่มประสิทธิภาพการทำงานได้

### 1.3 ขอบเขตของโครงการ

เพื่อศึกษาและทดลองใช้ฐานข้อมูลเชิงเวลาในการเก็บสถิติของระบบฐานข้อมูลเชิงเวลาและวางแผนแสดงระบบสำเนาฐานข้อมูลแบบกระจาย โดยที่

1. ฐานข้อมูลที่ใช้ในการทดลองเป็นฐานข้อมูล SQL Server เวอร์ชัน 2008
2. สามารถปรับปรุงและจัดเก็บข้อมูลของเครื่องที่ให้บริการสำเนาฐานข้อมูลบนระบบเครือข่ายได้
3. สามารถตรวจสอบและเปรียบเทียบความถูกต้องขององค์ประกอบบนฐานข้อมูลที่ทำสำเนาฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) กับฐานข้อมูลที่รับการทำสำเนาฐานข้อมูล (Subscriber)
4. สามารถจัดเก็บและแสดงข้อมูลการใช้งานฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) เพื่อใช้ในการวางแผนและจัดทำการสำเนาฐานข้อมูลที่เหมาะสม
5. การตรวจสอบและแสดงข้อมูลการเปลี่ยนแปลงของโครงสร้างองค์ประกอบภายในฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) เมื่อการเริ่มทำสำเนาฐานข้อมูลแล้ว (Reinitialize Subscription)

## 1.4 ขั้นตอนการพัฒนาระบบ

การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลดั่งนั้นในการพัฒนาระบบ ผู้พัฒนาได้ศึกษารายละเอียดและขั้นตอนปฏิบัติ เพื่อให้ได้มาซึ่งข้อมูลที่จะนำมาใช้ในการออกแบบและพัฒนาระบบ โดยมีขั้นตอนการทำงาน ดังนี้

1. ศึกษาลักษณะและการทำงานของระบบสำเนาฐานข้อมูล
2. ศึกษาถึงวิธีการทำงานของระบบจัดการบริหารข้อมูล โดยเน้นในจุดที่สามารถพัฒนาโปรแกรมประยุกต์ที่ช่วยประสิทธิภาพการทำงานให้กับระบบการจัดการบริหารข้อมูล
3. ศึกษาความต้องการจากผู้ที่เกี่ยวข้องเพื่อเป็นข้อมูลไปใช้ในการพัฒนาระบบ
4. ศึกษาเครื่องมือต่างๆ ที่ใช้ในแบบจำลองและโปรแกรมประยุกต์
5. วิเคราะห์ปัญหาและออกแบบระบบงานใหม่ โดยใช้แบบจำลองเชิงวัตถุด้วยยูเอ็มแอลมาเป็นเครื่องมือ
6. ออกแบบฐานข้อมูลโดยการใช้แบบจำลองความสัมพันธ์ระหว่างเอนทิตี และจัดทำพจนานุกรมข้อมูล
7. ออกแบบหน้าจอส่วนต่อประสานกับผู้ใช้และรายงานของระบบ
8. พัฒนาโปรแกรมที่ได้ออกแบบไว้
9. ปรับปรุงและแก้ไขผิดพลาด
10. จัดทำเอกสารประกอบ
11. สรุปข้อเสนอแนะต่างๆ

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลาด้วยวิธีการ UML เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล เป็นระบบที่นำเสนอการเปรียบเทียบฐานข้อมูลและช่วยการดำเนินงานงานของผู้บริหารฐานข้อมูล ซึ่งจะช่วยอำนวยความสะดวกในการดำเนินงานวางแผนงานให้ผู้ใช้ในการจัดการบริหารฐานข้อมูลและการติดต่อประสานงานกับเพื่อใช้ในการสำเนาฐานข้อมูลได้

## 1.6 ระยะเวลาในการดำเนินงาน

ตารางที่ 1.1 แสดงระยะเวลาในการดำเนินงาน

รายละเอียดการทำงาน	เดือนที่											
	1	2	3	4	5	6	7	8	9	10	11	12
1) การศึกษาระบบงานเดิม รวบรวมข้อมูลและขั้นตอนการจัดทำสำเนาฐานข้อมูล												
2) ศึกษาทฤษฎีและเครื่องมือที่ใช้ในการพัฒนาระบบ												
3) การวิเคราะห์และออกแบบระบบ												
4) พัฒนาระบบ												
5) ทดสอบการทำงานและปรับปรุงแก้ไขระบบ												
6) เรียบเรียงและจัดทำเอกสารการพัฒนาระบบ												

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

สำหรับการออกแบบและพัฒนา โปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลนั้น ได้ใช้แนวคิดในการพัฒนาเป็นแบบเว็บแอปพลิเคชัน ซึ่งเป็นการแสดงผลบนหน้าเว็บเพจ และสามารถติดต่อรับส่งข้อมูลระหว่างผู้ใช้กับระบบฐานข้อมูลได้ โดยทฤษฎีและเทคโนโลยีสำคัญที่เกี่ยวข้องกับการออกแบบระบบ มีดังต่อไปนี้

1. การพัฒนาระบบเชิงวัตถุ
2. การออกแบบฐานข้อมูล
3. การพัฒนาเว็บแอปพลิเคชัน
4. ฐานข้อมูลเชิงเวลา

### 2.1 การพัฒนาระบบเชิงวัตถุ

การพัฒนาระบบเชิงวัตถุ เป็นการพัฒนาระบบที่นำแนวคิดเชิงวัตถุมาปรับใช้กับการวิเคราะห์ การออกแบบ และการเขียน โปรแกรมของระบบ ซึ่งการใช้แนวคิดเกี่ยวกับวัตถุ เป็นแนวทางในการพิจารณาความเป็นจริงต่างๆที่เกิดขึ้นในโลก(สุนทริน วงศ์ศิริกุล และชัยวัฒน์ สิทธิกร โอฬารกุล 2550 : 3-12)

#### 2.1.1 คุณสมบัติของระบบเชิงวัตถุ

สำหรับระบบที่มีความสามารถทางด้านเชิงวัตถุ จะต้องสร้างโดยใช้แนวคิดหลักเชิงวัตถุ หรือมีความสามารถตามที่แนวคิดหลักกำหนดไว้ ดังนี้

1. การกำหนดสาระสำคัญ(Abstraction) เป็นการล้อมกรอบความคิดในการมองสิ่งใดสิ่งหนึ่ง เพื่อเน้นเฉพาะส่วนที่เกี่ยวข้องกับปัญหาที่ต้องการแก้ไข และตัดส่วนที่ไม่เกี่ยวข้องออกไป

2. การห่อหุ้ม (Encapsulation) เป็นการมองเฉพาะพฤติกรรมและข้อมูลที่ต้องการให้เห็น โดยไม่สามารถเปลี่ยนแปลงหรือเข้าถึงข้อมูลส่วนอื่นได้
3. ข้อความปฏิบัติการ(Message) การติดต่อระหว่างอ็อบเจกต์หนึ่งกับอ็อบเจกต์หนึ่ง เพื่อให้อ็อบเจกต์นั้นทำงานตามต้องการ
4. การจำแนกชั้น(Classification) การนำคุณสมบัติและพฤติกรรมที่เหมือนกันมารวมกลุ่มให้อยู่ในคลาสเดียวกัน โดยที่ในคลาสนั้นๆ มีคุณสมบัติร่วมกัน
5. ซุปเปอร์คลาส/ซับคลาส(Superclass/Subclass) ซุปเปอร์คลาสทำหน้าที่เป็นต้นแบบหรือแม่แบบให้กับซับคลาส และถ่ายทอดคุณสมบัติทุกอย่างไปให้กับซับคลาส
6. การรับทอด(Inheritance) เป็นการรับมรดกจากซุปเปอร์คลาสไปยังซับคลาส ซึ่งหมายถึง การที่ซับคลาสจะเหมือนกับซุปเปอร์คลาส และเพิ่มคุณสมบัติพิเศษบางอย่างที่ไม่มีในซูปเปอร์คลาส
7. โพลีมอร์ฟิซึม(Polymorphism) การที่คลาสแต่ละคลาสสามารถตอบสนองที่ต่างกันต่อเมสเสจเดียวกัน

### 2.1.2 ยูเอ็มแอล

ในการพัฒนาซอฟต์แวร์ จำเป็นต้องมีวิธีที่สามารถอธิบายสถาปัตยกรรมของระบบในมุมมองต่างๆ ซึ่งยูเอ็มแอล ได้นำเสนอการสร้างโมเดลเชิงวัตถุ และแสดงภาพรวมของระบบทั้งหมดในรูปแบบของแผนภาพ เพื่อให้เกิดความเข้าใจที่ตรงกันระหว่างผู้ออกแบบระบบ โปรแกรมเมอร์ และผู้ใช้งานยูเอ็มแอล มีองค์ประกอบ 3 ส่วนคือ

1. สัญลักษณ์ทั่วไป คือ สัญลักษณ์พื้นฐานที่ใช้ในการสร้างไดอะแกรมยูเอ็มแอลต่างๆ
2. ความสัมพันธ์ ประกอบด้วย ความสัมพันธ์แบบพึ่งพา ความสัมพันธ์แบบเกี่ยวพันและความสัมพันธ์แบบทั่วไป
3. ไดอะแกรมต่างๆ ซึ่งมีทั้งหมด 9 ไดอะแกรม ดังนี้

3.1 ยูสเคสไดอะแกรม(Use Case Diagram)

3.2 คลาสไดอะแกรม(Class Diagram)

3.3 อ็อบเจกต์ไดอะแกรม(Object Diagram)

3.4 สเตทชาร์ทไดอะแกรม(State Chart Diagram)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ซีควเอนซ์ไดอะแกรม(Sequence Diagram)

3.6 คอลลาบอเรชันไดอะแกรม (Collaboration Diagram)

3.7 แอกทิวิตีไดอะแกรม(Activity Diagram)

3.8 คอมโพเนนต์ไดอะแกรม(Component Diagram)

3.9 ดีพลอยเมนต์ไดอะแกรม(Deployment Diagram)

สำหรับการวิเคราะห์และออกแบบโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลได้ใช้ไดอะแกรม 5 ประเภทด้วยกัน ดังต่อไปนี้

### 1. ยูสเคสไดอะแกรม

ในการพัฒนาระบบงานใดๆนั้น การเก็บรวบรวมความต้องการของผู้ใช้มีความสำคัญเป็นอย่างยิ่ง ซึ่งยูสเคสไดอะแกรมนี้ จะทำหน้าที่บันทึกความต้องการ โดยมีองค์ประกอบ ดังนี้

1.1 แอกเตอร์ ใช้แทนคน กลุ่มคน หรือระบบอื่นที่ระบบสนใจปฏิสัมพันธ์ด้วย

1.2 ยูสเคส คือ กิจกรรมการทำงานหลักของระบบที่มองจากมุมมองของผู้ใช้ และยูสเคสจะทำการโต้ตอบกับผู้ใช้ของระบบ

1.3 เส้นแสดงความสัมพันธ์ คือ เส้นแสดงความสัมพันธ์ระหว่างแอกเตอร์กับยูสเคส

### 2. คลาสไดอะแกรม

สำหรับคลาสไดอะแกรม จะใช้สำหรับการแสดงถึงเอนทิตีต่างๆในระบบหรือภายในโดเมนหนึ่งๆ และอธิบายความสัมพันธ์ระหว่างเอนทิตีนั้นๆ ซึ่งคลาสจะประกอบด้วย 3 ส่วนคือ

2.1 ชื่อของคลาส

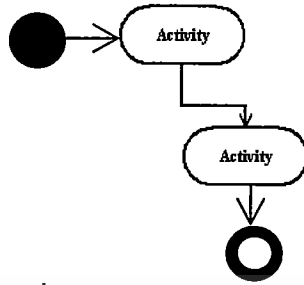
2.2 แอตทริบิวต์ของคลาส เป็นการกำหนดคุณลักษณะของคลาส

2.3 โอเปอเรชันของคลาส คือ กิจกรรมที่สามารถกระทำกับอ็อบเจกต์นั้นๆ ได้

### 3. แอกทิวิตีไดอะแกรม

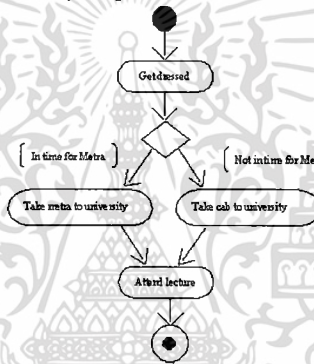
แอกทิวิตีไดอะแกรม เป็นไดอะแกรมที่ใช้แสดงขั้นตอนการทำงานของระบบ และแสดงทางเลือกที่เกิดขึ้นได้ โดยขั้นตอนการทำงานแต่ละขั้นตอนจะเรียกว่า แอกทิวิตี ซึ่งจะมีจุดเริ่มต้นและสิ้นสุดเพียงจุดเดียว รูปแบบของแอกทิวิตีไดอะแกรม แบ่งออกเป็น 4 รูปแบบ ดังนี้

3.1 แบบทั่วไป จะมีจุดเริ่มต้นกับจุดสิ้นสุดเพียงจุดเดียว และในระหว่างจุดเริ่มต้นกับจุดสิ้นสุดจะมีขั้นตอนต่างๆ ของระบบ ดังรูปที่ 2.1



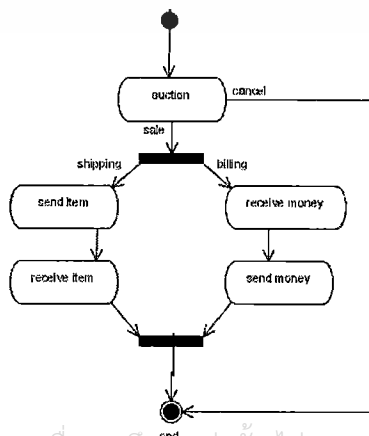
รูปที่ 2.1 แอกทิวิตีแบบทั่วไป

3.2 แบบมีทางเลือกให้ตัดสินใจ เป็นการกำหนดทางเลือกให้แก่แอกทิวิตีว่าจะเลือกทำกิจกรรมใดตามเงื่อนไขที่ระบุ ดังรูปที่ 2.3



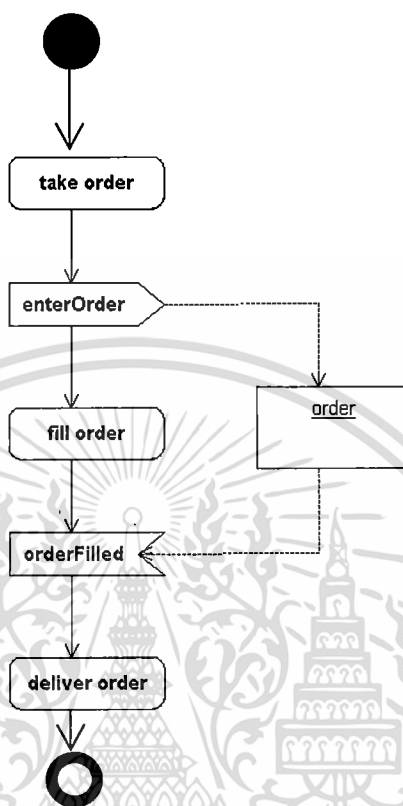
รูปที่ 2.2 แอกทิวิตีแบบมีทางเลือกให้ตัดสินใจ

3.3 แบบที่มีการทำงานพร้อมๆ กันหลายงาน กรณีที่มีกิจกรรมหลายกิจกรรมทำงานพร้อมๆ กัน เมื่อเขียนเป็นแอกทิวิตีไดอะแกรม จะใช้เส้นแนวนอนเส้นหนา เป็นสัญลักษณ์จัดกลุ่มงานที่มีการทำงานพร้อมกัน หรือการทำกิจกรรมในลักษณะคู่ขนาน ดังรูปที่ 2.3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้เผยแพร่ข้อมูลใดๆ และต้องรับผิดชอบต่อสิ่งที่มีการนำไปใช้

3.4 แบบแสดงการส่งสัญญาณ ในกระบวนการทำงาน อาจจะมีการส่งสัญญาณบางอย่างในระหว่างการทำงาน เมื่อเกิดการส่งและรับสัญญาณ จะเกิดเป็นแอกทิวิตีได้เช่นกัน ดังรูปที่ 2.4



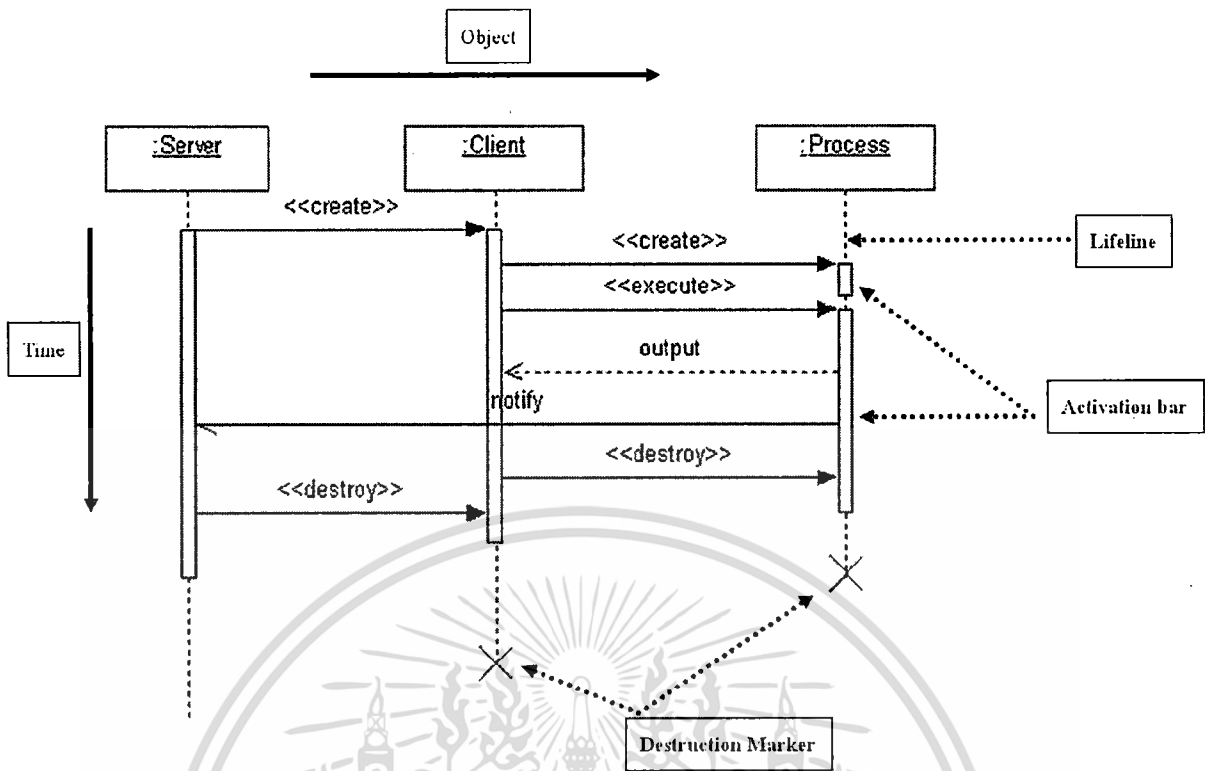
รูปที่ 2.4 แอกทิวิตีแบบแสดงการส่งสัญญาณ

#### 4. ซีควเอนซ์ไดอะแกรม

ซีควเอนซ์ไดอะแกรม เป็นแผนภาพที่อธิบายการติดต่อกันระหว่างวัตถุ ณ ช่วงเวลาต่างๆ ซึ่งจะแสดงลำดับการส่งข่าวสารที่เน้นช่วงเวลาการเกิดปฏิสัมพันธ์ ทำให้เห็นว่า ณ ช่วงเวลาหนึ่งมีออบเจกต์ต่างๆ ในระบบงานที่มีการติดต่อสื่อสารกันอย่างไร โดยองค์ประกอบสำคัญของไดอะแกรมนี้มีด้วยกัน 3 ส่วน คือ

4.1 อ็อบเจกต์ ประกอบด้วยอ็อบเจกต์ ไลน์ และแอกติเวชัน ดังรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 สัญลักษณ์ของอ็อบเจกต์ในซีควเอนซ์ไดอะแกรม

4.2 เมสเสจ เป็นการส่งข้อความระหว่างอ็อบเจกต์หนึ่งกับอ็อบเจกต์หนึ่ง หรืออาจจะส่งกลับมาตัวเองก็ได้ โดยจะแบ่งออกเป็น

4.2.1 การส่งข้อความจากผู้ให้บริการไปยังผู้ให้บริการ โดยเป็นการรอคำตอบ หรือการตอบกลับมาก่อนที่จะทำงานอื่นต่อไป

4.2.2 การติดต่อแบบไม่ต้องรอคำตอบที่จะตอบกลับ

4.2.3 การติดต่อกลับ ในกรณีที่ต้นทางเริ่มการติดต่อ แล้วปลายทางต้องมีการติดต่อกลับในการส่งข้อความ จะเขียนกำกับไว้ด้วยว่าเมสเสจนั้นคืออะไร หากเมสเสจมีเงื่อนไข จะมีการส่งกลับก็ต่อเมื่อเงื่อนไขนั้นเป็นจริง

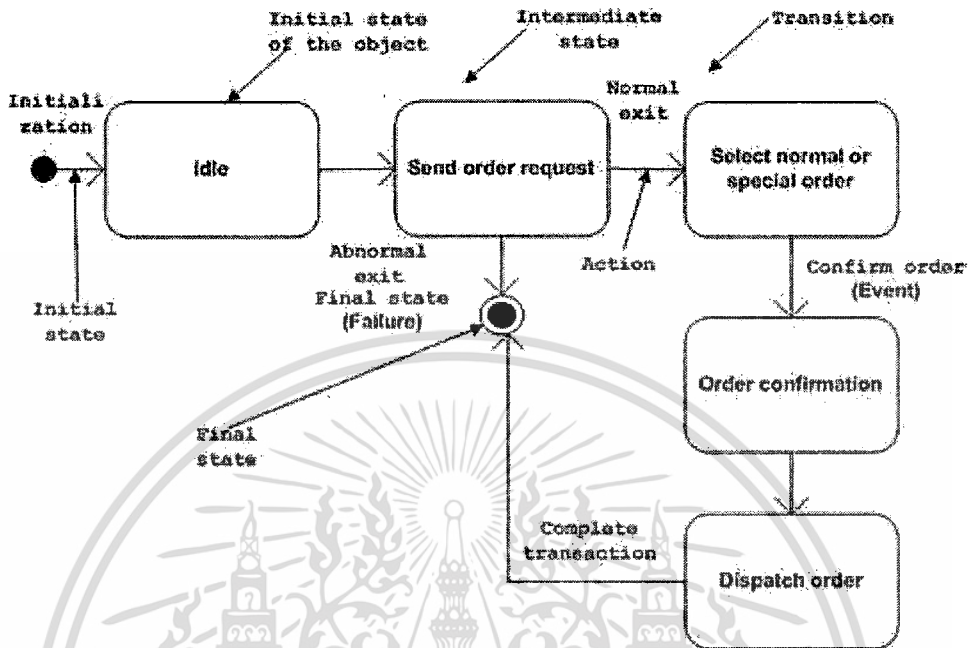
4.3 ช่วงเวลา เป็นการแสดงในลักษณะแนวตั้ง หรือจากบนลงล่าง โดยมีเมสเสจที่อยู่ด้านบนจะเป็นส่วนที่เกิดขึ้นก่อนเมสเสจที่อยู่ด้านล่าง และจากซ้ายไปขวา

5. สเตทชาร์ตไดอะแกรม

สเตทชาร์ตไดอะแกรม เป็นแผนภาพที่แสดงให้เห็นพฤติกรรมของอ็อบเจกต์ ซึ่งจะเน้นแสดงสถานะ การเปลี่ยนสถานะที่มีต่อเหตุการณ์ที่เกิดขึ้นในช่วงชีวิตของอ็อบเจกต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับสัญลักษณ์ที่ใช้ในไดอะแกรมนี้ ประกอบด้วย สถานะเริ่มต้น (Initial State) สถานะสุดท้าย (Final State) การเปลี่ยนสถานะ (Transition) และสถานะ(State) ดังรูปที่ 2.6



รูปที่ 2.6 สัญลักษณ์ของสเตทชาร์ตไดอะแกรม

## 2.2 การออกแบบฐานข้อมูล

สำหรับ โปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล ได้เลือกใช้ฐานข้อมูลเชิงสัมพันธ์ เนื่องจากเป็นระบบการจัดเก็บข้อมูลที่มีประสิทธิภาพ ข้อมูลและ โครงสร้างของข้อมูลไม่ขึ้นอยู่กับ โปรแกรมที่ใช้งาน สามารถลดความซ้ำซ้อนของข้อมูล ประหยัดเนื้อที่ในการจัดเก็บข้อมูลการแก้ไขข้อมูลจะถูกควบคุมให้ข้อมูลที่แก้ไขถูกต้องตามเกณฑ์ที่กำหนดไว้ นอกจากนี้ ยังสามารถใช้งานร่วมกันได้ ตลอดจนป้องกันไม่ให้ผู้ที่ไม่มีสิทธิเข้ามาใช้ระบบได้อีกด้วย(ศิริลักษณ์ โรจนกิจอำนวย. 2552:135)

อย่างไรก็ตาม ฐานข้อมูลเชิงสัมพันธ์ มีข้อเสียด้านฮาร์ดแวร์และระบบปฏิบัติการ ที่จะต้องมีความสามารถสูงกว่าฐานข้อมูลที่มีโครงสร้างข้อมูลรูปแบบอื่น เนื่องจากโปรแกรมดีบีเอ็ม แอสของฐานข้อมูลนี้ จะทำหน้าที่จัดการกับโครงสร้างของข้อมูลภายในฐานข้อมูลแทนผู้ใช้ ส่งผลให้การทำงานของโปรแกรมมีความซับซ้อน และต้องใช้ทรัพยากรของฮาร์ดแวร์ที่สูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.1 ขั้นตอนการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลแบ่งออกเป็น 3 ระดับ คือ

1. การออกแบบระดับแนวคิด เป็นการออกแบบในส่วนของหลักการเชิงข้อมูลและความสัมพันธ์ ซึ่งเกิดจากการออกแบบหลังจากที่ได้ทำการวิเคราะห์ระบบมาแล้ว ผลลัพธ์ที่ได้จะอยู่ในรูปแบบของอีอาร์ไออะแกรม

2. การออกแบบระดับตรรกะ เป็นการสร้างโครงสร้างฐานข้อมูลจริงบนกระดาษ หรือพิมพ์เขียวฐานข้อมูล โดยสร้างจากผังอีอาร์ไออะแกรม ที่ผ่านการนอร์มัลไลเซชันแล้ว จนเกิดเป็นโครงสร้างฐานข้อมูลที่ลดความซ้ำซ้อน และสามารถนำไปใช้ได้จริง ผลลัพธ์ในขั้นตอนนี้จะได้เป็นพจนานุกรมข้อมูล

3. การออกแบบระดับกายภาพ เป็นการสร้างโครงสร้างฐานข้อมูลจริง โดยออกแบบฐานข้อมูลบนระบบจัดการฐานข้อมูล ที่ต้องการนำมาใช้จริง เช่น กำหนดประเภทและขนาดของฟิลด์ข้อมูลที่สอดคล้องกับการทำงานจริง และความสามารถของระบบฐานข้อมูลที่เลือกใช้ อีกทั้งยังต้องสร้างให้เป็นไปตามโครงสร้างฐานข้อมูลที่ได้จากการออกแบบระดับตรรกะด้วย ซึ่งระบบหรือโปรแกรมที่ใช้ในการจัดการสำหรับระบบนี้คือ ไมโครซอฟท์ เอสคิวเอลเซิร์ฟเวอร์

### 2.2.2 อีอาร์ไออะแกรม

การออกแบบฐานข้อมูลขึ้นมาใช้งานนั้น จะต้องอาศัยแบบจำลองของข้อมูล เพื่อนำเสนอรายละเอียดต่างๆ ที่เกี่ยวข้องกับข้อมูลในฐานข้อมูลที่ออกแบบ สำหรับการออกแบบฐานข้อมูลของระบบนี้ ได้นำแบบจำลองอีอาร์ไอมาใช้ในการอธิบายข้อมูลต่างๆ ในรูปแบบ Crow's Foot ซึ่งจะประกอบด้วย

1. เอนทิตี คือ บุคคล สถานที่ หรือเหตุการณ์ที่ทำให้เกิดกลุ่มของข้อมูลที่ต้องการจัดเก็บ รวมถึงสามารถบ่งชี้ถึงความเป็นเอกลักษณ์เฉพาะตัวได้

2. แอตทริบิวต์ คือ คุณลักษณะเฉพาะของเอนทิตีหรือความสัมพันธ์

3. ความสัมพันธ์ เป็นความสัมพันธ์ระหว่างเอนทิตีต่างๆ ในระบบ โดยแบ่งออกเป็น ความสัมพันธ์แบบหนึ่งต่อหนึ่ง ความสัมพันธ์หนึ่งต่อกลุ่ม ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

4. ดีกรีของความสัมพันธ์ คือ จำนวนเอนทิตีในการมีส่วนร่วมของความสัมพันธ์

### 2.2.3 พจนานุกรมข้อมูล

พจนานุกรมข้อมูล คือ พจนานุกรมชนิดหนึ่งที่ทำขึ้นมา เฉพาะสำหรับการบริหาร

จัดการข้อมูลในระบบ โดยพจนานุกรมนี้จะอธิบายรายละเอียดของข้อมูลที่เกี่ยวข้องกับการทำงานกับเอกสารไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูล ได้แก่ โครงสร้างข้อมูล ลักษณะข้อมูลที่จัดเก็บ กฎเกณฑ์รักษาความถูกต้องของการจัดเก็บข้อมูล เป็นต้น

#### 2.2.4 ระบบฐานข้อมูลไมโครซอฟท์ เอสคิวแอลเซิร์ฟเวอร์

เอสคิวแอลเซิร์ฟเวอร์ คือระบบจัดการฐานข้อมูลที่พัฒนาโดยบริษัทไมโครซอฟท์ ถูกออกแบบขึ้นมาเพื่อช่วยให้แก้ไขปัญหาขององค์กรในปัจจุบันเมื่อข้อมูลมีจำนวนมากขึ้น เช่น ความต้องการเรียกใช้ข้อมูลให้เร็วขึ้น และการตัดสินใจที่อิงกับข้อมูลมากขึ้นกว่าเดิม สามารถใช้งานกับระบบหลายๆ รูปแบบ เพื่อนักพัฒนาใช้สำหรับการพัฒนาระบบ หรือคนที่ต้องการศึกษาการทำงาน ของซอฟต์แวร์นำไปศึกษาต่อ เพราะมีวิธีการใช้งานที่ค่อนข้างง่าย และมีเครื่องมือต่างๆช่วยในการทำงานค่อนข้างมาก

สำหรับคุณสมบัติหลักของไมโครซอฟท์ เอสคิวแอลเซิร์ฟเวอร์ สามารถกล่าวโดยสังเขปได้ ดังนี้

1. สามารถรองรับผู้ใช้งานที่เข้ามาใช้งานฐานข้อมูลได้หลายๆคนในเวลาเดียวกัน
2. สามารถเพิ่มประสิทธิภาพการทำงานให้กับระบบฐานข้อมูล เช่น การสร้างวิว (View) การสร้างดัชนี (Index) และการสร้างฟังก์ชันเพิ่มเติมนอกเหนือจากฟังก์ชันที่มีไว้ให้แล้ว เป็นต้น
3. สามารถจัดการความปลอดภัยของระบบได้อย่างมีประสิทธิภาพ และมีความยืดหยุ่น เนื่องจากระบบความปลอดภัยแบ่งออกเป็นหลายระดับชั้นตั้งแต่ระดับการล็อกอิน (Login) ใช้งานระบบฐานข้อมูล จนถึงระดับการตรวจสอบสิทธิ์การใช้งานออบเจกต์ต่างๆ ที่อยู่ในระบบฐานข้อมูล
4. มีความสามารถในการสำรองข้อมูล และยังสามารถคืนสภาพฐานข้อมูลได้อย่างอัตโนมัติ เมื่อระบบทำงานล้มเหลว เนื่องจากมีการเก็บข้อมูลที่เกี่ยวข้องกับการแก้ไขข้อมูลนั้นไว้ ทำให้ข้อมูลในระบบฐานข้อมูลมีเสถียรภาพน่าเชื่อถือมากยิ่งขึ้น

### 2.3 การพัฒนาเว็บแอปพลิเคชัน

การพัฒนาระบบการกล่าวถึง โปรแกรม Microsoft SQL Server 2008 ซึ่งใช้สำหรับสร้าง และจัดการฐานข้อมูล การกล่าวถึงโปรแกรม Visual Studio ซึ่งใช้สำหรับสร้างหน้าจอ และภาษา C# ซึ่งเป็นภาษาที่ใช้ในการสร้างหน้าจอบนโปรแกรม Visual Studio และนอกจากนี้ยังกล่าวถึง โปรแกรม .NET Framework ผู้จัดทำได้ดำเนินการศึกษา ค้นคว้าหาข้อมูล เพื่อทราบถึงขั้นตอน เอกสารวิธีการในการทำงานซึ่งจำเป็นต้องศึกษาที่ถุกฎที่เกี่ยวข้อง ดังต่อไปนี้

ให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.1 โปรแกรม Microsoft Visual Studio 2008

Microsoft Visual Studio 2008 เป็นเครื่องมือที่ใช้พัฒนาโปรแกรมแบบ Visual Programming บนระบบปฏิบัติการ Windows ซึ่งได้รับการพัฒนามาจากภาษา BASIC (Beginners All Purpose Symbolic Instruction Code) ซึ่งเป็นภาษาโปรแกรมที่ได้รับความนิยมอย่างแพร่หลายสำหรับผู้เริ่มหัดเขียนโปรแกรมคอมพิวเตอร์ เนื่องจาก BASIC เป็นภาษาโปรแกรมที่สามารถทำความเข้าใจได้ง่าย

Microsoft Visual Studio 2008 เป็นโปรแกรมที่บริษัทไมโครซอฟท์ได้พัฒนาอย่างต่อเนื่อง (เวอร์ชันก่อนหน้านี้ได้แก่เวอร์ชัน Microsoft Visual Studio 2005) ไมโครซอฟท์ได้เพิ่มขีดความสามารถขึ้นมาอีกมากมายใน Visual Basic.NET สิ่งที่น่าสนใจก็คือการปรับเปลี่ยนภาษาเป็นลักษณะ OOP (Object Oriented Programming) เต็มตัวเหมือนกับภาษาโปรแกรมสมัยใหม่ เช่น C++ , C# , Delphi และ Java เป็นต้น และด้วยความที่ Microsoft Visual Studio 2008 อยู่ในตระกูล Visual Basic.NET จึงซึมซับเอาความสามารถอื่น ๆ ใน Visual Basic.NET เข้ามาด้วยเช่นกัน นอกจากนี้แล้ว Visual Basic.NET ยังเป็นภาษาที่ถูกผนวกเข้ากับโปรแกรมอื่น ๆ ของไมโครซอฟท์ เช่น Microsoft Access, Excel, Word เป็นต้น เพื่อใช้เขียนโปรแกรมลักษณะสคริปต์ (Script) หรือมาโคร (macro) การเรียนรู้ Visual Basic จึงนับว่าคุ้มค่าเป็นอย่างยิ่งโดยทั่วไป Microsoft Visual Studio 2008 ถือเป็นส่วนหนึ่งของเครื่องมือพัฒนาในชุดโปรแกรม Visual Studio.NET แต่ก็มีเวอร์ชัน Standard ที่ไม่ได้รวมอยู่ในชุด Microsoft Visual Studio 2008 อีกด้วย

### 2.3.2 โปรแกรมดอตเน็ตเฟรมเวิร์ก .NET Framework

ดอตเน็ตเฟรมเวิร์ก (.NET Framework) คือแพลตฟอร์มสำหรับพัฒนาซอฟต์แวร์สร้างขึ้นโดยบริษัทไมโครซอฟท์ เพื่อใช้สำหรับการพัฒนาโปรแกรมประยุกต์ .NET Framework ถูกออกแบบมาเพื่อให้สามารถถูกเรียกใช้จากตัวแปลภาษาใดๆ ก็ได้ เช่น C#, C++, Visual Basic, JScript, Delphi เป็นต้น เพื่อให้ผู้ปรกณ์สามารถติดต่อสื่อสารกันได้หมด .NET Framework ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือน โปรแกรมหนึ่งที่จะสร้าง สภาวะแวดล้อม หรือไลบรารี (Library) เป็นจำนวนมากสำหรับการเขียนโปรแกรม เพื่อที่จะส่งเสริมความสามารถ ในการสื่อสารสัมพันธ์ระหว่างภาษา ที่ถูกเรียกว่า Common Type System (CTS) การใช้ดอตเน็ตเฟรมเวิร์กทำให้เกิดกฎเกณฑ์ทั้งหลายต่างๆ ขึ้น จึงทำให้มี .Net Common Language Runtime (CLR) เป็นตัวกลางที่จะรับผิดชอบในการจัดการกับระบบปฏิบัติการ ของโปรแกรมประยุกต์ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในอนาคต Microsoft ยังหวังที่จะนำเอา ระบบ .net นี้ไปติดตั้งลงบน อุปกรณ์ ทุกชนิด เพื่อให้ อุปกรณ์ ทุกอย่างมีระบบฯหนึ่งที่เหมือนกันหมด .NET Framework จะมี ส่วนประกอบ ภายในแบ่งออกเป็น 3 ชั้นใหญ่ๆ คือ

- Programming Language เป็นรูปแบบของภาษาที่ออกแบบมาเพื่อให้สามารถทำงานในสถานะที่เป็น .NET โดยทางไมโครซอฟท์ ได้ใช้ภาษาหลักที่จะใช้ในการพัฒนา บน .NET นี้ 3 ภาษา คือ

- C# เป็น ภาษา ใหม่ที่ Microsoft พัฒนา มาจาก C++ กับ JAVA เป็นหลัก
- VB.NET เป็น ภาษา ที่ พัฒนา มาจาก Visual Basic ในเวอร์ชัน 6.0
- JScript.net เป็น ภาษา ที่ พัฒนา มาจาก J Script ซึ่งเป็น JavaScript ใน เวอร์ชันของ Microsoft

- Base Classes Library เปรียบเสมือน ชุดคำสั่งสำเร็จรูปย่อยๆ ที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่ง ที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเครื่องอำนวยความสะดวกในการเขียนโปรแกรม ซึ่งไลบรารีในภาษาต่างๆ ส่วนใหญ่จะอยู่ใน รูปแบบไฟล์ include แต่ถ้าเป็น ASP สิ่งที่เป็น ไลบรารีก็คือ คอมโพเนนต์ต่างๆนั่นเอง ซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น ไลบรารีพื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ ภาษาใดในการพัฒนาก็สามารถที่จะเรียกใช้ไลบรารีที่เป็นตัวเดียวกันได้หมด

- Common Language Runtime (CLR) มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆ กัน กลายเป็นภาษา ในรูปแบบมาตรฐานเดียวกันทั้งหมด โดยจะเรียกภาษานี้ว่า Intermediate language (IL) เมื่อต้องการที่จะรัน โปรแกรมใดๆ CLR จะทำการตรวจสอบเครื่องว่ามีสถานะแวดล้อม การทำงาน เช่นใดหลังจากนั้นก็ทำการคอมไพล์เป็น โปรแกรมที่เหมาะสมต่อ การทำงานของเครื่องนั้น ทำให้สามารถใช้งาน โปรแกรมต่างๆ ได้อย่างมีประสิทธิภาพสูงสุด ในแต่ละเครื่อง

การที่โปรแกรมที่เขียนบนคอมพิวเตอร์เฟรมเวิร์กจะทำงานบนสภาพแวดล้อมที่บริหารโดย Common Language Runtime (CLR) จึงทำให้ผู้พัฒนาไม่จำเป็นต้องคำนึงถึงความสามารถที่แตกต่างกันระหว่างหน่วยประมวลผลต่างๆ CLR ยังให้บริการในด้านกลไกระบบความปลอดภัย การบริหารหน่วยความจำ และ Exception Handling การเขียนโปรแกรมภายใต้คอทเน็ทเฟรมเวิร์คจึงมีประโยชน์ และข้อดีกว่าการเขียนโปรแกรมโดยทั่วไป ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีระบบไลบรารีที่เป็นมาตรฐานเดียวกันทำให้ไม่ต้องคอยกังวลว่าภาษาที่จะใช้เขียนนั้นมีไลบรารีครบตามที่ต้องการหรือไม่รวมทั้งไม่ต้องคอยระวังว่าจะใช้ไลบรารีของภาษาใดภาษาหนึ่งแล้วอีกภาษาจะไม่มีไลบรารีตัวนี้

- ไม่ขึ้นกับระบบปฏิบัติการเนื่องจากระบบปฏิบัติการที่แต่ละบุคคลหรือองค์กรใช้นั้น อาจจะไม่เหมือนกันแต่การเขียนโปรแกรมที่รันภายใต้คอปเน็ตนั้นหากมีคอปเน็ตเฟรมเวิร์คติดตั้งอยู่ โปรแกรมที่เขียนก็สามารถทำงานบนระบบปฏิบัติการนั้นได้

- ใช้ทุกภาษาในการพัฒนาได้ทำให้ผู้พัฒนาโปรแกรมไม่ต้องศึกษาภาษาใหม่ๆ สามารถใช้ภาษาที่ตนเองถนัดที่สุดในการพัฒนาโปรแกรมได้

- มีการควบคุมสภาพแวดล้อมในการทำงานเป็นอย่างดี เนื่องจากเป็นระบบมาตรฐานทำให้การควบคุมและจัดสรรระบบต่างๆสามารถทำได้ง่าย ทั้งการจัดสรรหน่วยความจำ ทำให้เครื่องสามารถทำงานได้รวดเร็วขึ้น

- มีความปลอดภัยที่สูงขึ้นเนื่องจากคอปเน็ตเฟรมเวิร์คสามารถกำหนดสิทธิ์ในการใช้งาน (Permission) ของผู้ใช้งานได้มากขึ้น โดยสามารถกำหนดได้ว่าจะให้โปรแกรมในส่วนใดทำงานได้บ้าง แล้วแต่สิทธิ์การเข้าใช้งานของผู้ใช้งาน อีกทั้งยังมีฟังก์ชันไว้สำหรับถอดรหัส และเข้ารหัสข้อมูลอีกด้วย

### 2.3.3 ภาษา C#

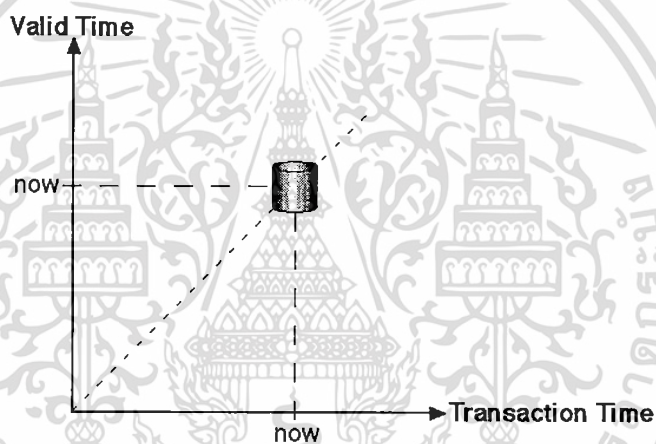
ภาษา C# เป็นภาษาโปรแกรมเชิงวัตถุทำงานบนคอตเน็ตเฟรมเวิร์ค พัฒนาโดยบริษัท ไมโครซอฟท์เป็นหนึ่งในภาษาที่สามารถนำมาสร้าง Application ที่ Run อยู่ใน .Net ได้ เป็นอีกวิวัฒนาการหนึ่งของภาษา C และ C++ ซึ่งถูกสร้างขึ้นมาให้ทำงานกับ Platform .Net ภาษา C# ถูกออกแบบมาอย่างมีพื้นฐานมาจาก C และ C++ ให้รับเอาฟีเจอร์ที่ดีที่สุดจากภาษาอื่น ๆ และลบสิ่งปัญหาที่มีอยู่ในภาษาเหล่านั้นให้หมดไป ภาษา C# เป็นภาษาโครงสร้างบล็อก (block structured) หมายความว่า ทุก ๆ statement ถือเป็น ส่วนหนึ่งของ block code และ block เหล่านี้ถูกจำกัดด้วยเครื่องหมาย { } เหมือนกับ C++ ทุกประการ รูปแบบจึงเหมือนกับ C++ และมีโครงสร้างเหมือน C++ และภาษา C# ยังไม่มีการแจ้งเตือนเกี่ยวกับช่องว่างที่อยู่ใน code ซึ่งเป็นอักขระที่รู้จักกันในชื่อว่า white space นั่น ก็คือ การมีอิสระในการจัดรูปแบบ source code ถึงแม้ว่าการทำตามกฎที่แน่นอนจะช่วยให้ทำสิ่งต่าง ๆ ให้อ่านได้ง่ายขึ้นก็ตาม code C# นั้นสร้างขึ้นจาก statement ชุดหนึ่งแต่ละ Statement จะจบด้วย semicolon (;) เนื่องจาก white space ถูกมองข้ามไป จึงสามารถมีหลาย ๆ statement ในบรรทัดเดียวกันได้ แต่เพื่อความง่ายในการอ่าน จึงเพิ่ม carriage return เข้าไปหลัง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ประโยชน์ภายในเท่านั้น ไม่อนุญาตให้เผยแพร่หรือแจกจ่ายโดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

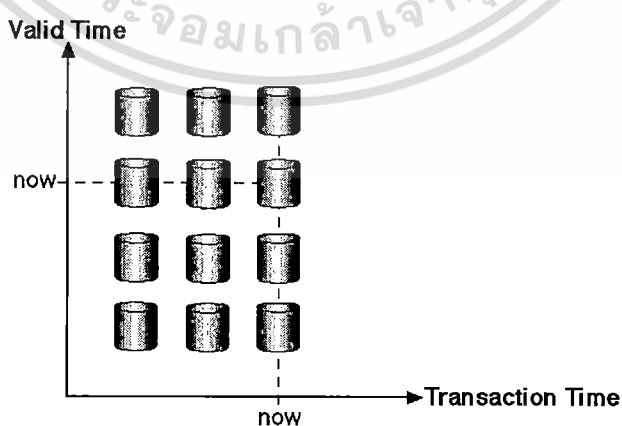
semicolon (;) ดังนั้นจะไม่มีหลาย ๆ Statement ในบรรทัดเดียวกัน สำหรับตัวแปรใน ภาษา C# การที่จะใช้ตัวแปร จะต้องประกาศตัวแปรก่อน ก็จะต้องกำหนดชื่อ และชนิดของตัวแปรนั้น หลังจากประกาศชนิดของตัวแปรเสร็จสิ้น ก็สามารถใช้เป็นตัวจัดเก็บข้อมูลตามชนิดที่ประกาศไว้ได้ทันที

## 2.4 ฐานข้อมูลเชิงเวลา

ระบบฐานข้อมูลเชิงเวลา(Temporal Database) คือ ระบบฐานข้อมูลสามารถเรียกค้นข้อมูลที่แปรผันตามเวลาได้ ระบบฐานข้อมูลที่เก็บข้อมูลหลายสถานะ(Snodgrass. 1999 : 36-44) ดังรูปที่ 2.6 และรูปที่ 2.7



รูปที่ 2.6 การจัดเก็บข้อมูลของระบบฐานข้อมูลปกติ



รูปที่ 2.7 การจัดเก็บข้อมูลของระบบฐานข้อมูลเชิงเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.6 และรูปที่ 2.7 สามารถเห็นความแตกต่างของระบบฐานข้อมูลทั้งสองได้ ข้อมูลในฐานข้อมูลเชิงเวลา แตกต่างจากข้อมูลที่ถูกเก็บอยู่ในฐานข้อมูลปกติ ตรงที่ระบบฐานข้อมูลเชิงเวลาเก็บช่วงเวลาไว้กับข้อมูล(Timestamp) เพื่อแสดงว่าข้อมูลถูกเก็บในฐานข้อมูลเมื่อใดหรือข้อมูลเป็นจริงในช่วงเวลาใด โดยเวลาที่เก็บใน Temporal Database มี 3 แบบคือ

1. Valid Time คือ เวลาที่ข้อมูลเป็นจริง ซึ่งจะแบ่งแยกย่อยออกเป็น
  - Valid Time Start คือ เวลาเริ่มต้นที่ข้อมูลนั้นเป็นจริง
  - Valid Time End คือ เวลาสุดท้ายที่ข้อมูลนั้นเป็นจริง
2. Transaction Time คือ เวลาที่ข้อมูลถูกเก็บในฐานข้อมูล โดยสามารถแบ่งเป็น
  - Transaction Time Start คือ เวลาที่เริ่มจัดเก็บข้อมูลลงฐานข้อมูล
  - Transaction Time End คือ เวลาที่แก้ไขหรือลบข้อมูล
3. User Defined Time คือ เวลาที่ผู้ใช้งานสร้างขึ้นเพื่อใช้งาน ซึ่งผู้ใช้งานต้องจัดการและเรียกคั่นเอง ระบบฐานข้อมูลจะไม่จัดการให้อย่างอัตโนมัติ ซึ่งเวลาชนิดนี้จะเป็นเวลาที่จริงเสมอ เช่น วันเกิด วันเข้าทำงาน และวันที่เกษียณ เป็นต้น

#### 2.4.1 คำสั่งเชิงเวลา

ในการใช้งานฐานข้อมูลเชิงเวลา สามารถสั่งได้ 3 ลักษณะ คือ

- คำสั่งปัจจุบัน (Current Query) คือ การสั่งให้กระทำกับข้อมูล ณ ปัจจุบัน หรือ ขณะนี้
- คำสั่ง Sequenced (Sequenced Query) คือ คำสั่งที่กระทำ โดยคำนึงถึงตราเวลา (Timestamp) ที่ประทับไว้กับข้อมูล เช่น หากเรามีตารางข้อมูลเชิงเวลาชนิด Valid time 2 ตารางคือ ข้อมูลตำแหน่งพนักงานและข้อมูลเงินเดือนของพนักงาน คำสั่งสืบค้นว่าพนักงานคนหนึ่งเคยมีตำแหน่งอะไร และได้รับเงินเดือนเท่าไร เนื่องจากการเปลี่ยนตำแหน่งและการเปลี่ยนเงินเดือน ไม่จำเป็นที่ต้องทำพร้อม ๆ กัน
- คำสั่ง Nonsequenced (Nonsequenced Query) คือ คำสั่งที่มองว่า ตราเวลาที่ประทับไว้ก็เป็นข้อมูลเช่นกัน แตกต่างกับ Sequenced Query ที่ตรงตราเวลาเป็นอีกส่วนหนึ่งต่างจากนั้น สามารถนำข้อมูลตราเวลามาใช้ประมวลได้

## 2.4.2 ช่วงเวลา

เป็นระยะเวลาที่มีการกำหนดเริ่มต้นและกำหนดสิ้นสุด การนำเสนอช่วงเวลาสามารถทำได้แบบปลายเปิด ซึ่งแสดงด้วยสัญลักษณ์ '[' ]' หมายความว่า ช่วงเวลานั้น รวมจุดเวลาเริ่มต้นหรือสิ้นสุดที่ระบุไว้ด้วยหรือแบบปลายเปิด แสดงด้วยสัญลักษณ์ '( )' หมายความว่า ช่วงเวลาไม่รวมจุดเวลาเริ่มต้นหรือสิ้นสุด (Snodgrass. 2000 : 90)

การนำเสนอ 2 แบบสามารถผสมกันเป็น ช่วงเวลา 4 ลักษณะ หากให้ a และ b เป็นจุดเวลา การนำเสนอช่วงเวลามีได้ ดังนี้ คือ

- PERIOD [a,b] เป็นช่วงเวลาแบบปลายปิด – ปิด (Closed – Closed Representation)
- PERIOD [a, b) เป็นช่วงเวลาแบบปลายปิด – เปิด (Closed – Open Representation)
- PERIOD (a, b] เป็นช่วงเวลาแบบปลายเปิด – ปิด (Open – Closed Representation)
- PERIOD (a,b) เป็นช่วงเวลาแบบปลายเปิด – เปิด (Open – Open Representation)

ในฐานะข้อมูลเชิงเวลา การเปรียบเทียบช่วงเวลาควรเปรียบเทียบช่วงเวลาที่นำเสนอแบบเดียวกัน การเปรียบเทียบช่วงเวลาที่นำเสนอต่างกัน ต้องนำความหยาบละเอียดของจุดเวลาเข้ามาเกี่ยวข้องด้วยว่ามีหน่วยเป็นวัน เป็นชั่วโมง หรือ เป็นนาที มิฉะนั้นความหมายจะเปลี่ยนแปลงช่วงเวลาที่เหมาะสม และเมื่อใช้ ต้องคำนึงถึงความหยาบละเอียดของจุดเวลาน้อยคือช่วงเวลาแบบปลายปิด – เปิด ดังนั้น งานวิจัยฉบับนี้จะยึดช่วงเวลาแบบปลายปิด – เปิด เป็นหลัก

ช่วงเวลาลายปิด – เปิด ..... พบว่า a ต้องน้อยกว่า b เสมอจึงจะเป็นช่วงเวลาที่แท้จริง  
**ความสัมพันธ์ของช่วงเวลา**

หาก A = PERIOD [a1, a2) , B = PERIOD [b1, b2) (Snodgrass. 2000 : 91)

### ตารางที่ 2.1 ความสัมพันธ์ของช่วงเวลา

ความสัมพันธ์	ความหมาย	ความสัมพันธ์เปรียบเทียบเท่า
A equals B	$a1 = b1 \text{ AND } a2 = b2$	B equals A
A before B	$a2 < b1$	B before A
A meets B	$a2 = b1$	B meet A
A overlaps B	$a1 < b1 \text{ AND } a2 < b2 \text{ AND } b1 < a2$	B overlaps A
A during B	$b1 < a1 \text{ AND } a2 < b2$	B during A
A starts B	$a1 = b1 \text{ AND } a2 < b2$	B starts A
A finishes B	$b1 < a1 \text{ AND } b2 = a2$	B finishes A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณีใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านอื่นๆ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

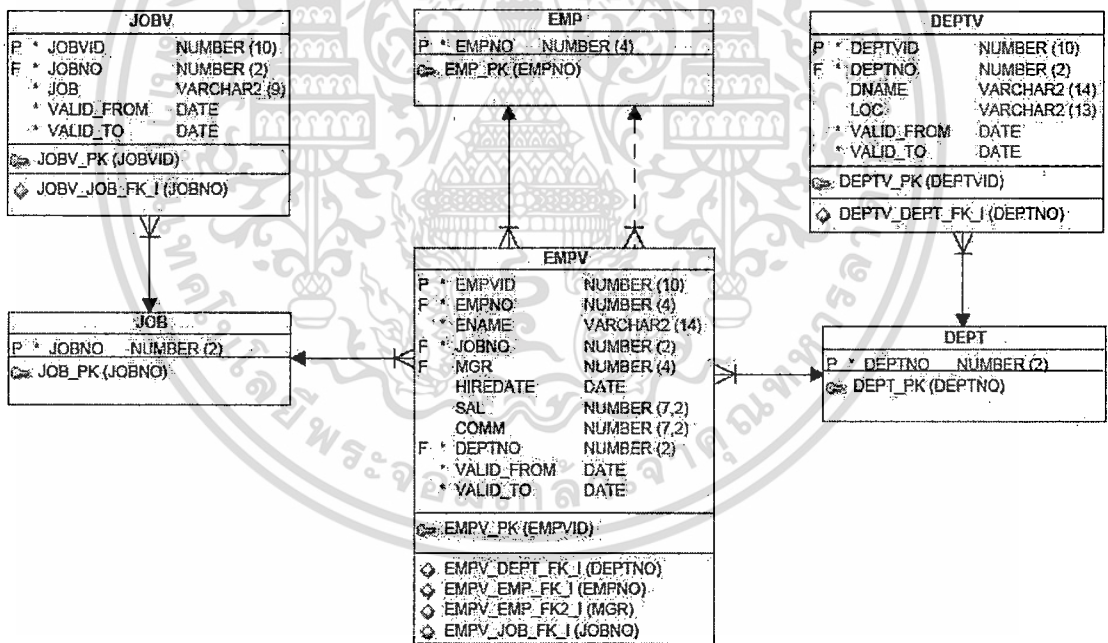
2.4.3 ฐานข้อมูล temporaldb

ในงานวิจัยฉบับนี้ จะใช้ฐานข้อมูล temporaldb เป็นตัวอย่างในการอธิบายหลักการต่างๆ ของฐานข้อมูลเชิงเวลา โดยฐานข้อมูล temporaldb มีตารางข้อมูลดังรูป 2.1

ตารางข้อมูล DEPT EMP และ JOB เป็นส่วนหนึ่งของฐานข้อมูลปกติ โดยมี DEPTV EMPV กับ JOBV เป็นตารางข้อมูลเชิงเวลา ซึ่งขอให้อ่านใจเพียงตารางข้อมูล DEPTV EMPV กับ JOBV เท่านั้น เพราะต้องการใช้ในการอธิบายหลักการฐานข้อมูลเชิงเวลา

นอกจาก DEPTVID EMPVID กับ JOBVID เป็น Primary Key ของตารางข้อมูล DEPTV EMPV และ JOBV ตามลำดับ ในความหมายของฐานข้อมูลเชิงสัมพันธ์ ไม่ใช่ Primary Key ในความหมายของฐานข้อมูลเชิงเวลา

ตารางข้อมูล DEPT EMP และ JOB จะใช้เพื่ออธิบายแนวทางประยุกต์ใช้ฐานข้อมูลเชิงเวลาร่วมกับฐานข้อมูลปกติ



รูปที่ 2.9 Entity – Relationship Diagram ฐานข้อมูลตัวอย่าง

## 2.4.4 ประโยชน์ของฐานข้อมูลเชิงเวลา

การใช้งานฐานข้อมูลเชิงเวลาในระบบ Valid Time Period ให้ประโยชน์กับผู้ใช้นี้

### 1. การสืบค้นลักษณะ Time – slice ในตารางเดียว

เป็นการสืบค้นว่า ข้อมูล ณ ช่วงเวลาที่กำหนดนั้นเป็นอย่างไร (Snodgrass. 2000 : 145) หากไม่ได้นำ Valid Time Period มาใช้ เช่น ตารางข้อมูลที่มีเพียงคอลัมภ์ที่เป็น User – defined Time การสืบค้นลักษณะนี้

หากไม่มีการเก็บประวัติไว้ในรูปแบบใดๆ ไม่สามารถสืบค้นได้

หากเก็บประวัติในลักษณะจุดเวลา (วันที่เปลี่ยนแปลง) จะต้องตั้งสมมติฐานก่อนว่า จะให้ข้อมูลเป็นช่วงเวลาอย่างไร ทำให้ไม่สามารถนำเสนอข้อมูลที่เป็นจริง

### ตัวอย่าง

ณ วันที่ 1/1/2546 พนักงานรหัส 10000 ชื่อ นามสกุลอะไร

ณ ช่วงเวลาตั้งแต่ ปี 2510 ถึง ปี 2548 บุคคล รหัส 10000 ใช้ชื่อ นามสกุลอะไร

### 2. การสืบค้นข้อมูลในอดีตเชื่อมโยงหลายตาราง

ข้อมูลที่เราต้องการอาจถูกแยกออกเก็บไว้ในหลายตาราง แต่ละตาราง หากเราต้องการสืบค้นหาความสัมพันธ์เชิงเวลาของข้อมูลเหล่านั้น

หากไม่มีการเก็บประวัติในรูปแบบใด ๆ ไม่สามารถสืบค้นได้

หากเก็บประวัติไว้ในลักษณะจุดเวลา การเชื่อมตารางโดยใช้ช่วงเวลาเพื่อหาข้อมูลที่สัมพันธ์กัน ไม่น่าเป็นไปได้ (เพราะไม่มีช่วงเวลา) การเชื่อมโดยอาศัยจุดเวลาที่ไม่ให้ผลลัพธ์ตามต้องการ

การสืบค้นเชิงเวลาต้องอาศัย Sequenced Join เพื่อให้ได้ผลลัพธ์ตามต้องการ (Snodgrass. 2000 : 146)

### 3. การสืบค้นหาความสัมพันธ์เชิงเวลาระหว่างแถวข้อมูล

การติดตามความสัมพันธ์ของแถวข้อมูล โดยกำหนดว่า ต้องการความสัมพันธ์บนคณันใด (ที่ไม่ใช่ Primary Key) และช่วงเวลาใด

เช่นเดียวกับกรณีข้อ 2.2.1 และ 2.2.2 หากไม่มีการเก็บ Valid Time Period ไว้ก็ไม่สามารถสืบค้นได้ เพราะกรณีนี้จำเป็นต้องใช้ Sequenced Join หรือการหาช่วงเวลาที่เหลื่อมซ้อนกัน (Snodgrass. 2000 : 146)

#### ตัวอย่าง

นาย ก เคยมีโอกาสรู้จักนาย ข บ้างไหม เช่น เรียนอยู่ที่เดียวกัน ในช่วงเวลาเดียวกัน หรืออยู่ตำบลเดียวกัน ในช่วงเวลาเดียวกัน

กำหนดช่วงเวลา นาย ก เคยมีโอกาสรู้จักนาย ข บ้างไหม เช่น เรียนอยู่ที่เดียวกัน ในช่วงระหว่างปี 2000 และปี 2004 หรือ อยู่ในตำบลเดียวกัน ในช่วงเวลาที่กำหนด

#### 2.4.5 ตารางข้อมูลเชิงเวลา

ตารางข้อมูลเชิงเวลาชนิด Valid Time คือตารางข้อมูลที่มีตาราง ประทับอยู่เพื่อบอกให้ทราบว่าแถวข้อมูลหนึ่งๆ เป็นจริงเฉพาะในช่วงเวลาใด

เราใช้คอลัมภ์ ชื่อว่า validfrom และ validto เป็นการนำเสนอช่วงเวลาแบบปลายปิด – เปิด โดยให้ช่วงเวลา Valid Time นั้นเป็นคุณสมบัติของแต่ละแถวข้อมูล

การสร้างตารางในระดับแถวข้อมูลก่อให้เกิด Redundancy ขึ้น เนื่องจากแต่ละคอลัมภ์ในแถวข้อมูลต่างมีช่วงเวลาของตนเอง ดังนั้น ตารางควรผูกติดอยู่กับระดับคอลัมภ์ ซึ่งนำไปสู่แนวความคิดว่า ฐานข้อมูลเชิงวัตถุจะเหมาะสมกับการนำฐานข้อมูลเชิงเวลาไปใช้มากกว่าฐานข้อมูลเชิงสัมพันธ์

ในงานวิจัยฉบับนี้ เนื่องจากต้องการนำเสนอฐานข้อมูลเชิงเวลาไปใช้กับฐานข้อมูลที่มีอยู่ในปัจจุบัน ซึ่งเป็นฐานข้อมูลเชิงสัมพันธ์จึงเลือกใช้ตารางในระดับแถวข้อมูล ซึ่งอธิบายถึงแนวทางประยุกต์ จะพบว่า สามารถสร้างตารางข้อมูลเชิงเวลาแบบนำตารางเวลาไปไว้ในระดับคอลัมภ์ก็ได้ หากในฐานข้อมูลเชิงเวลานั้นมีคอลัมภ์ข้อมูลเพียงคอลัมภ์เดียว

## บทที่ 3

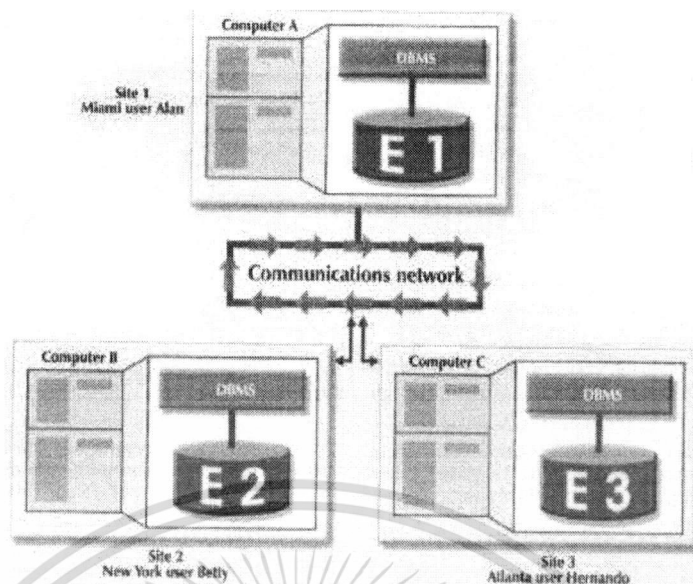
# การวิเคราะห์ระบบงานปัจจุบัน

ระบบการทำสำเนาฐานข้อมูล(Database Replication) เป็นเทคโนโลยีฐานข้อมูลขั้นสูงที่มีวัตถุประสงค์เพื่อประยุกต์ใช้กับฐานข้อมูล ที่มีฐานข้อมูลเก็บอยู่ในระบบคอมพิวเตอร์ตามสถานที่ต่างๆ การมองข้อมูลในฐานข้อมูลแบบกระจาย สถาปัตยกรรมระบบฐานข้อมูลแบบกระจาย ชนิดของฐานข้อมูลแบบกระจาย รูปแบบการจัดเก็บข้อมูลในฐานข้อมูลแบบกระจาย ตลอดจนกระบวนการสืบค้นข้อมูลแบบกระจาย ดังต่อไปนี้

1. สถาปัตยกรรมระบบฐานข้อมูลแบบกระจาย
2. ชนิดของระบบฐานข้อมูลแบบกระจาย
3. รูปแบบการจัดเก็บข้อมูลในระบบฐานข้อมูลแบบกระจาย
4. กระบวนการสืบค้นข้อมูลแบบกระจาย
5. การทำสำเนาฐานข้อมูลในไมโครซอฟท์ เอสคิวแอลเซิร์ฟเวอร์
6. ปัญหาของระบบงานปัจจุบัน
7. คุณลักษณะของเครื่องมือตรวจสอบการสำเนาฐานข้อมูล
8. แนวทางการแก้ไขปัญหา

### 3.1 สถาปัตยกรรมระบบฐานข้อมูลแบบกระจาย

ระบบฐานข้อมูลที่มีฐานข้อมูลเก็บอยู่ในระบบคอมพิวเตอร์หลายๆ เครื่องที่ติดตั้งอยู่ตามที่ตั้งต่างๆ โดยที่ระบบคอมพิวเตอร์เหล่านี้มีการสื่อสารกันได้ แต่ผู้ใช้จะมองการใช้งานเสมือนหนึ่งว่ากำลังใช้ระบบฐานข้อมูลแบบรวมอยู่ตามปกติ ตัวอย่างเช่น ระบบการฝากถอนเงินจากธนาคารสมมติว่าเราเปิดบัญชีไว้ที่กรุงเทพฯ และทำการฝากถอนเงินในจังหวัดนครปฐม จำเป็นต้องใช้ข้อมูลบัญชีของเราจากฐานข้อมูลในกรุงเทพฯ ดังนั้น เราจึงต้องเสียค่าธรรมเนียม ในส่วนของการสื่อสารเชื่อมต่อฐานข้อมูลระหว่างจังหวัดนครปฐมกับกรุงเทพฯ แต่ถ้าเรามีบัญชีอยู่ที่จังหวัดนครปฐมหรือในเขตพื้นที่เดียวกัน การฝากถอนเงินก็จะไม่เสียค่าใช้จ่ายใดๆ ดังรูปที่ 3.1 แสดงฐานข้อมูลแบบกระจาย



รูปที่ 3.1 ฐานข้อมูลแบบกระจาย

จากภาพอธิบายได้ว่า ฐานข้อมูลจะถูกแบ่งออกเป็น 3 ส่วน คือ E1 E2 และ E3 และติดตั้งในที่ที่แตกต่างกันซึ่งอาจจะเป็นคนละจังหวัดหรือคนละประเทศก็ได้ โดยการทำงานของคอมพิวเตอร์จะเชื่อมโยงผ่านระบบเครือข่าย ซึ่งผู้ใช้แต่ละคนไม่จำเป็นต้องทราบชื่อ หรือตำแหน่งที่ตั้งของฐานข้อมูลที่เก็บกักใช้งานอยู่เลย นั่นคือไม่ว่าผู้ใช้จะอยู่ที่ไมอามี นิวยอร์ก หรือ แอตแลนต้า ผู้ใช้ยังสามารถใช้งานฐานข้อมูลเดียวกันได้ ซึ่งการมองข้อมูลในฐานข้อมูลแบบกระจายมี 3 แบบคือ

1. การมองผ่านสถานที่ตั้ง

การมองผ่านสถานที่ตั้ง (location transparency) เป็นการกระจายฐานข้อมูลไปเก็บตามเครื่องคอมพิวเตอร์ ที่ติดตั้งตามสถานที่ต่างๆ

2. การมองผ่านการเก็บซ้ำ

การมองผ่านการเก็บซ้ำ (replication transparency) เป็นการเก็บข้อมูลชุดเดียวกันไว้หลายๆ เครื่องตามสถานที่ที่ติดตั้ง ทำให้สะดวกในการเรียกใช้งาน แต่ทุกครั้งที่แก้ไขข้อมูล จะต้องแก้ไขทุกเครื่องให้ถูกต้องตรงกัน

3. การมองผ่านการแตกกระจาย

การมองผ่านการแตกกระจาย (fragmentation transparency) เป็นการจัดแยกข้อมูลออกเก็บตามจุดต่างๆ เพื่อให้ได้ประโยชน์ใช้งานสูงสุด มักจะเก็บในจุดที่มีการเรียกใช้งานบ่อย ๆ

ปัจจุบันการจัดเก็บข้อมูล มีแนวโน้มที่จะกระจายข้อมูลไปเก็บไว้หลายๆ เครื่อง เพื่อเพิ่มประสิทธิภาพในการประมวลผล เพิ่มความน่าเชื่อถือของข้อมูล เพื่อกระจายการใช้งานข้อมูลอย่างเหมาะสม ดังนั้นจึงเกิดแนวความคิดของระบบจัดการฐานข้อมูลแบบกระจาย เพื่อตอบสนองต่อเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความต้องการเหล่านั้น ในระบบฐานข้อมูลแบบกระจาย ข้อมูลจะถูกจัดเก็บไว้ในหลายๆ เครื่อง และจะเชื่อมต่อเข้าด้วยกันผ่านระบบเครือข่าย โดยแต่ละที่จะมีระบบจัดการฐานข้อมูลเป็นของตนเอง และสามารถที่จะทำงานได้ด้วยตนเอง หรือร่วมกันทำงานก็ได้ โดยที่คอมพิวเตอร์แต่ละเครื่อง จะเรียกว่าไซด์ (site) หรือ โหนด (node) สถาปัตยกรรมระบบฐานข้อมูลแบบกระจาย มีดังนี้คือ

### 1. ระบบรับ-ให้บริการ

ระบบรับ-ให้บริการ (client-server system) เป็นระบบคอมพิวเตอร์ที่ประกอบไปด้วย เครื่องคอมพิวเตอร์จำนวนหลายๆ เครื่อง ทั้งคอมพิวเตอร์ส่วนบุคคล เครื่องให้บริการเพิ่มข้อมูล เครื่องให้บริการการพิมพ์ โดยมีการเชื่อมต่อกันผ่านระบบเครือข่าย แบ่งออกเป็นสองส่วนคือเครื่องที่ขอรับบริการ (client) และเครื่องให้บริการ (server) ในเรื่องของระบบฐานข้อมูลจะแบ่งโปรแกรมระบบจัดการฐานข้อมูลออกเป็นสองระดับ คือ รับบริการ และให้บริการ ซึ่งบางไซด์จะทำงานเฉพาะโปรแกรมรับบริการ และบางไซด์จะทำงานเฉพาะโปรแกรมให้บริการเท่านั้น ตัวอย่างเช่น ในการประมวลผลคำสั่งเอสคิวแอล คำเนิการระหว่างรับ และให้บริการ มีขั้นตอนดังนี้

- 1.1 คอมพิวเตอร์ลูกข่ายวิเคราะห์คำสั่ง และแยกคำสั่งออกเป็นหลายๆ คำสั่งตามจำนวนของแม่ข่ายในแต่ละไซด์ที่จะทำการสืบค้นข้อมูล จากนั้นก็จะส่งคำสั่งไปยังแม่ข่ายแต่ละไซด์นั้น
- 1.2 คอมพิวเตอร์แม่ข่ายแต่ละตัวจะประมวลผลคำสั่งในเครื่องตนเอง และส่งผลลัพธ์กลับไปยังลูกข่าย
- 1.3 คอมพิวเตอร์ลูกข่ายจะรวบรวมผลลัพธ์ที่ได้รับกลับมาจากไซด์ ต่าง ๆ แล้วสร้างเป็นผลลัพธ์สุดท้าย

จากแนวทางนี้เครื่องที่ทำหน้าที่ประมวลผลคำสั่งจะเรียกว่าดีพี (database processor : DP หรือเครื่อง back-end) และคอมพิวเตอร์ลูกข่ายจะเรียกว่าแอปพี (application processor : AP หรือเครื่อง front-end) ในระบบจัดการฐานข้อมูลแบบกระจาย (distributed database management system : DDBMS) จะแบ่งออกเป็นสามระดับคือ โปรแกรมส่วนแม่ข่าย รับผิดชอบเกี่ยวกับการจัดการข้อมูล ซึ่งจะเหมือนกับโปรแกรมการจัดการข้อมูลแบบรวมศูนย์ โปรแกรมส่วนลูกข่าย จะรับผิดชอบเกี่ยวกับการเข้าถึงข้อมูลที่อยู่ใน สารบัญเพิ่มเติมของระบบการจัดการฐานข้อมูลแบบกระจาย (distributed database management system catalog : DDBMS catalog) และทำการร้องขอการใช้บริการข้อมูลไปที่ไซด์อื่น และโปรแกรมส่วนของการติดต่อสื่อสาร (communication) จะสนับสนุนการสื่อสารข้อมูลบนระบบเครือข่าย ซึ่งจะถูกใช้งานโดยโปรแกรมส่วนลูกข่าย เพื่อทำการส่งคำสั่งและข้อมูลไปยังไซด์ที่ต้องการ

ลูกข่ายจะทำหน้าที่ในการสร้างแผนการสืบค้นข้อมูลแบบกระจาย ควบคุมให้การทำการรายการเปลี่ยนแปลง (transaction) มีคุณสมบัติแอซิด (analysis console for Intrusion databases : ACID) อยู่เสมอ และหน้าที่ที่สำคัญอีกอย่างหนึ่งคือมีความสามารถที่จะซ่อนรายละเอียดของการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระจายของข้อมูลจากผู้ใช้ได้ นั่นคือผู้ใช้งานในไซต์ต่างๆ ก็สามารถที่จะเข้าถึงข้อมูลได้โดยไม่จำเป็นต้องรู้ว่าข้อมูลเก็บอยู่ที่ใด เราเรียกคุณสมบัติ นี้ว่าการมองเห็นแบบกระจาย (distribution transparency)

## 2. ระบบผู้ช่วยแม่ข่าย

สถาปัตยกรรมเดิม ไม่ยอมให้การสอบถามข้อมูลกระจายไปหาแม่ข่ายหลายๆ ชุด เพราะว่า การประมวลผลของลูกค้าจะต้องมีความสามารถในการแตกการสอบถามไปหาแม่ข่ายหลายๆ ที่ได้ ในขณะที่เดียวกันเมื่อได้รับคำตอบกลับมา ก็จะต้องนำมารวมตอบให้กับผู้ถามได้ในตำแหน่งเดิม แต่ผู้ช่วยแม่ข่ายทำการทำการประมวลผลการสอบถามของผู้ใช้ที่ยังออกไปนั้นค่อนข้างจะซับซ้อน และความสามารถของมันจะต้องเริ่มที่จะทำการทับซ้อนกับแม่ข่ายได้ การกระจายการสอบถามระหว่างผู้ใช้กับแม่ข่ายจะเป็นเรื่องที่ยากขึ้น การที่จะขจัดปัญหาของการแบ่งแยก ได้นำเราไปสู่ระบบผู้ใช้ แม่ข่ายที่เรียกว่า ผู้ช่วยแม่ข่าย เราสามารถที่จะสะสมฐานข้อมูลของแม่ข่ายหลายๆ ตัว สามารถที่จะทำการเปลี่ยนแปลงตามฐานข้อมูลท้องถิ่นแต่ละอัน ซึ่งจะเกิดความร่วมมือในการประมวลผลรายการเปลี่ยนแปลงในแต่ละแม่ข่าย

เมื่อแม่ข่ายได้รับการสอบถามที่ไม่ใช่ของมัน มันจะทำการสร้างการสอบถามส่งไปยังแม่ข่ายอื่น และรอรับคำตอบกลับมาส่งไปให้ผู้ถาม แนวคิดคือการกระจายของคำสั่งถูกทำให้เกิดต้นทุนที่เหมาะสม ถ้าพูดถึงในแง่ของค่าใช้จ่ายแล้วจะได้ว่าต้นทุนของการสื่อสารข้อมูล จะเท่ากับต้นทุนของการประมวลผลท้องถิ่นนั่นเอง

## 3. ระบบมิดเดิลแวร์

ระบบมิดเดิลแวร์ (middleware systems) เป็นสถาปัตยกรรมที่ถูกออกแบบ มาเพื่อยินยอมให้การสอบถามสามารถกระจายไปสู่แม่ข่ายหลายๆ ตัวได้ โดยปราศจากความต้องการของฐานข้อมูลแม่ข่ายหลายๆ ตัวนั้น ซึ่งสามารถจัดการกับการประมวลผลหลายๆ ไซต์ได้ เป็นเรื่องที่น่าสนใจมากเมื่อพยายามที่จะนำเอาระบบแม่ข่ายเก่าๆ หลายตัวที่ไม่สามารถทำการขยายการใช้งานได้แล้ว

แนวความคิดก็คือ เราแค่มีหนึ่งฐานข้อมูลแม่ข่ายที่สามารถจัดการกับคำสั่ง รายการเปลี่ยนแปลง และกระจายไปยังแม่ข่ายหลายๆ ตัว ซึ่งแม่ข่ายเก่าๆ เพียงแต่ประมวลผลคำสั่งของตามความสามารถเดิมๆ เราอาจจะคิดว่าแม่ข่ายพิเศษตัวนี้เป็นเพียงชั้นหนึ่งของซอฟต์แวร์ที่จะร่วมการทำงานรายการเปลี่ยนแปลง ส่งต่อไปยังแม่ข่ายเก่าหลายๆ ตัวได้เอง ซอฟต์แวร์ตัวนี้ถูกเรียกว่า มิดเดิลแวร์ ชั้นของมิดเดิลแวร์ สามารถรวบรวมการกลับมาของฐานข้อมูล จากแม่ข่ายหลายๆ ตัว โดยที่ตัวมันเองไม่ต้องมีฐานข้อมูล

## 3.2 ชนิดของระบบฐานข้อมูลแบบกระจาย

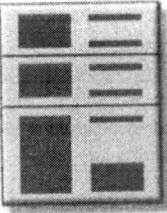
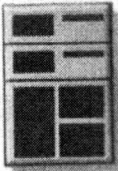
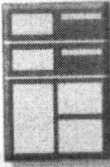
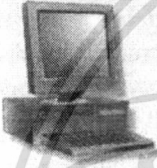

ระบบฐานข้อมูลแบบกระจาย สามารถจำแนกได้เป็น 2 ชนิด คือ

### 3.2.1 ระบบการกระจายฐานข้อมูลเดียวกัน

ระบบการกระจายฐานข้อมูลเดียวกัน (homogeneous distributed database system) เป็นระบบการจัดการฐานข้อมูลแบบกระจายที่ทุกๆ แห่งจะใช้ระบบการจัดการฐานข้อมูล ระบบปฏิบัติการ และแบบจำลองข้อมูล เดียวกัน อาจอยู่บนเครื่องต่างระบบกันก็ได้ รูปแบบของข้อมูลในฐานข้อมูลแต่ละแห่งมีรูปแบบเดียวกัน ส่งผลให้การออกแบบและการจัดการมีความสะดวกและง่าย เช่น สมมติว่าส่วนกลางมีการใช้ออราเคิล ดังนั้นทุกๆ สาขาที่จะใช้ออราเคิลเหมือนกันหมด

### 3.2.2 ระบบการกระจายฐานข้อมูลต่างกัน

ระบบการกระจายฐานข้อมูลต่างกัน (heterogeneous distributed database system หรือ multidatabase system) เป็นระบบการจัดการฐานข้อมูลแบบกระจายที่แต่ละสาขามีการใช้งานระบบการจัดการฐานข้อมูล ระบบปฏิบัติการหรือแบบจำลองข้อมูลที่ต่างกัน ซึ่งเป็นไปได้ว่าในแต่ละสาขานั้นได้มีการพัฒนาฐานข้อมูลไว้ใช้งานในส่วนงานของตนเอง โดยวิธีนี้อาจทำให้มีความแตกต่างกัน ทั้งในด้านอุปกรณ์ทางคอมพิวเตอร์และระบบการจัดการฐานข้อมูล ดังนั้น จึงส่งผลให้แต่ละสาขาต่างก็มีฐานข้อมูลที่พัฒนาขึ้นในรูปแบบที่แตกต่างกัน ก่อให้เกิดปัญหาในด้านของการนำข้อมูลมารวมกัน จึงจำเป็นต้องใช้อุปกรณ์ที่ทำหน้าที่เป็นตัวกลางในการแปลงข้อมูลต่างๆ ที่เรียกว่าเกตเวย์ เพื่อให้สามารถนำฐานข้อมูลในรูปแบบหนึ่งมาใช้งานกับฐานข้อมูลของตนได้ แต่อย่างไรก็ตาม เราสามารถหลีกเลี่ยงการใช้อุปกรณ์เกตเวย์ได้ เนื่องจากมีโปรแกรมที่ทำหน้าที่เชื่อมต่อระหว่างระบบการจัดการฐานข้อมูลและโปรแกรมต่างๆ เช่น โอดีบีซี (open database connectivity : ODBC) โดยผ่านเอพีไอ (application programming interface : API) ของโอดีบีซี ทำให้สามารถเชื่อมโยงกับฐานข้อมูลที่มาจากต่างผลิตภัณฑ์ได้ ดังรูปที่ 3.2 แสดงแพลตฟอร์มระบบการจัดการฐานข้อมูล ระบบปฏิบัติการและระบบเครือข่าย

	Platform	DBMS	Operating System	Network Communications Protocol
	IBM 3090	DB2	MVS	APPC LU 6.2
	DEC/VAX	VAX rdb	MVS	DECnet
	IBM AS/400	AS/400	OS/400	3270
	RISC computer	Informix	UNIX	TCP/IP
	Pentium CPU	Oracle	Windows XP	NetBIOS

รูปที่ 3.2 แพลตฟอร์ม ระบบการจัดการฐานข้อมูล ระบบปฏิบัติการและระบบเครือข่าย

จากรูปจะเห็นว่า ถ้าแต่ละสาขามีการใช้แพลตฟอร์ม IBM 3090 ระบบการจัดการฐานข้อมูล DB2 ระบบปฏิบัติการ MVS และระบบเครือข่าย APPC LU 6.2 เหมือนกันทุกสาขา ก็จะเรียกว่า ระบบการกระจายฐานข้อมูลเดียวกัน แต่ถ้ามีบางอย่างไม่เหมือนกัน จะเรียกว่า ระบบการกระจายฐานข้อมูลต่างกัน

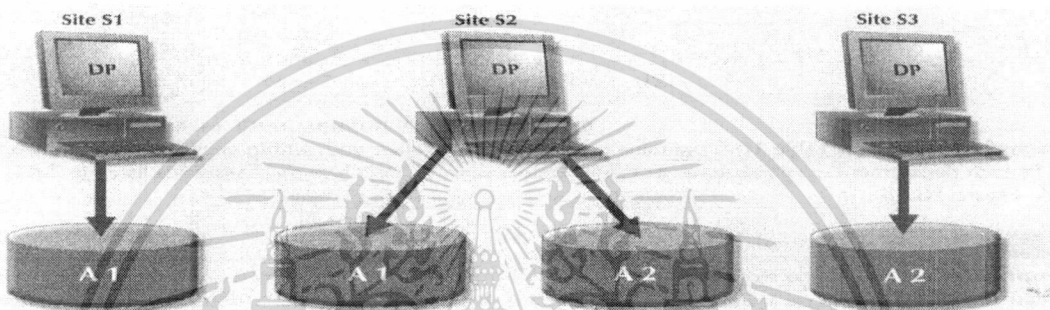
### 3.3 รูปแบบการจัดเก็บข้อมูลในระบบฐานข้อมูลแบบกระจาย

การจัดเก็บข้อมูลในระบบฐานข้อมูลแบบกระจาย จะทำการจัดเก็บข้อมูลกระจายไปยังเครื่องต่างๆ ซึ่งมีรูปแบบการจัดเก็บดังนี้

#### 3.3.1 วิธีการทำสำเนา

วิธีการทำสำเนา (replication) เป็นการทำสำเนาด้วยการนำตารางข้อมูลต่างๆ มาทำสำเนาและนำไปจัดเก็บในแต่ละสาขา ซึ่งโดยปกติจะทำสำเนาแบบเต็มรูปแบบ นั่นคือ มีข้อมูลเหมือนกันทุกประการ ซึ่งมีข้อดีคือการทำสำเนาข้อมูลไว้ที่แต่ละสาขา เมื่อสาขาใดสาขาหนึ่งเกิดปัญหา ระบบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็สามารถดำเนินต่อไปได้ด้วยการนำข้อมูลจากสาขาอื่นๆ มาใช้งาน นอกจากนี้การทำสำเนาไว้ทุกสาขายังทำให้การทำงานมีความรวดเร็ว กล่าวคือ การเรียกใช้ข้อมูลสามารถเรียกใช้ข้อมูลได้จากสาขาของตนเอง ไม่ต้องเสียเวลาติดต่อเรียกใช้ข้อมูลผ่านระบบเครือข่าย ส่วนข้อเสียของการทำสำเนาไว้ที่แต่ละสาขา ก็คือ กระบวนการทำให้ข้อมูลแต่ละสาขาถูกต้องตรงกัน นั่นคือหากมีการเปลี่ยนแปลง แก้ไข ข้อมูลที่สาขาย่อยใด ต้องทำการเปลี่ยนแปลง แก้ไข ข้อมูลที่สาขาอื่นๆ ด้วย นอกจากนี้การทำสำเนาข้อมูลไว้ทุกสาขา ทำให้สิ้นเปลืองเนื้อที่ในการจัดเก็บข้อมูลด้วย ดังรูปที่ 3.3 แสดงการจัดเก็บข้อมูลด้วยวิธีการทำสำเนา



รูปที่ 3.3 การจัดเก็บข้อมูลด้วยวิธีการทำสำเนา

### 3.3.2. วิธีการแบ่งแยกข้อมูล

วิธีการการแบ่งแยกข้อมูล (fragmentation หรือ partition) เป็นการจัดเก็บข้อมูลด้วยการแบ่งแยกข้อมูลออกเป็นส่วนๆ กล่าวคือ จะมีตารางใดๆ ที่จะถูกแบ่งออกเป็นตารางย่อยๆ ซึ่งตารางย่อยๆ เหล่านี้สามารถนำกลับมารวมกันให้เป็นตารางเดิมได้ การจัดเก็บข้อมูลด้วยวิธีแบ่งแยกข้อมูล จะมีอยู่ 3 ชนิด คือ

**3.3.2.1 การแบ่งแยกข้อมูลตามแนวนอน** การแบ่งแยกข้อมูลตามแนวนอน (horizontal fragmentation) หมายถึงการแบ่งแถวของข้อมูลในตารางหนึ่ง ไปไว้ในตารางย่อยๆ ซึ่งจะต้องประกอบด้วยแถวที่ไม่ซ้ำกัน หรือกล่าวได้ว่าแถวในตารางย่อยจะประกอบด้วยแถวที่ตรงตามเงื่อนไขของคำสั่ง select นั่นเอง ดังตารางที่ 3.1 แสดงเงื่อนไขการแบ่งแยกข้อมูลตามแนวนอน และตารางที่ 3.2 แสดงผลลัพธ์การแบ่งแยกข้อมูลตามแนวนอน

### ตารางที่ 3.1 เงื่อนไขการแบ่งแยกข้อมูลตามแนวนอน

ตาราง CUSTOMER							
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
10	Sinex, Inc.	12 Main St.	TN	\$3,500.00	\$2,700.00	3	\$1,245.00
11	Martin Corp.	321 Sunset Blvd.	FL	\$6,000.00	\$1,200.00	1	\$0.00
12	Mynux Corp.	910 Eagle St.	TN	\$4,000.00	\$3,500.00	3	\$3,400.00
13	BTBC, Inc.	Rue du Monde	FL	\$6,000.00	\$5,890.00	3	\$1,090.00
14	Victory, Inc.	123 Maple St.	FL	\$1,200.00	\$550.00	1	\$0.00
15	NBCC Corp.	909 High Ave.	GA	\$2,000.00	\$350.00	2	\$50.00

เงื่อนไข

FRAGMENT NAME	CONDITION	CUS_NUM	NUMBER OF ROWS
CUST_H1	CUST_STATE='TN'	10,12	2
CUST_H2	CUST_STATE='GA'	15	1
CUST_H3	CUST_STATE='FL'	11,13,14	3

### ตารางที่ 3.2 ผลลัพธ์การแบ่งแยกข้อมูลตามแนวนอน

ตาราง CUST_H1							
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
10	Sinex, Inc.	12 Main St.	TN	\$3,500.00	\$2,700.00	3	\$1,245.00
12	Mynux Corp.	910 Eagle St.	TN	\$4,000.00	\$3,500.00	3	\$3,400.00

ตาราง CUST\_H2

CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
15	NBCC Corp.	909 High Ave.	GA	\$2,000.00	\$350.00	2	\$50.00

ตาราง CUST\_H3

CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
11	Martin Corp.	321 Sunset Blvd.	FL	\$6,000.00	\$1,200.00	1	\$0.00
13	BTBC, Inc.	Rue du Monde	FL	\$6,000.00	\$5,890.00	3	\$1,090.00
14	Victory, Inc.	123 Maple St.	FL	\$1,200.00	\$550.00	1	\$0.00

จากตารางจะเห็นว่า การแบ่งแยกข้อมูลตามแนวนอนจากตาราง CUSTOMER ซึ่งประกอบด้วยลักษณะประจำ CUS\_NUM CUS\_NAME CUS\_ADDRESS CUS\_STATE เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CUS\_LIMIT CUS\_RATING CUS\_DUE ได้เป็น 3 ตาราง คือ CUST\_H1 CUST\_H2 และ CUST\_H3 ซึ่งทั้งสามตารางประกอบด้วยลักษณะประจำเหมือนกันกับตาราง CUSTOMER และแต่ละตารางเป็นการเลือกแถวด้วยเงื่อนไขที่ต้องการ นั่นคือ ตาราง CUST\_H1 เป็นการเลือกแถวจากตาราง CUSTOMER ด้วยเงื่อนไข CUST\_STATE='TN' ซึ่งจะได้ข้อมูล 2 แถว คือ CUS\_NUM มีค่าเท่ากับ 10 และ 12 ตาราง CUST\_H2 เป็นการเลือกแถวจากตาราง CUSTOMER ด้วยเงื่อนไข CUST\_STATE='GA' ซึ่งจะได้ข้อมูล 1 แถว คือ CUS\_NUM มีค่าเท่ากับ 15 ตาราง CUST\_H3 เป็นการเลือกแถวจากตาราง CUSTOMER ด้วยเงื่อนไข CUST\_STATE='FL' ซึ่งจะได้ข้อมูล 3 แถว คือ CUS\_NUM มีค่าเท่ากับ 11 13 และ 14

**3.3.2.2 การแบ่งแยกข้อมูลตามแนวตั้ง** การแบ่งแยกข้อมูลตามแนวตั้ง (vertical fragmentation) หมายถึงการแบ่งลักษณะประจำของข้อมูลในตารางหนึ่ง ไปไว้ในตารางย่อยๆ ซึ่งจะประกอบด้วยลักษณะประจำที่มีค่าไม่ซ้ำกัน หรือกล่าวได้ว่า เป็นการเลือกลักษณะประจำที่ต้องการไว้ในตารางย่อยด้วยคำสั่ง projection นั่นเอง ดังตารางที่ 3.3 แสดงเงื่อนไขการแบ่งแยกข้อมูลตามแนวตั้ง และตารางที่ 3.4 แสดงผลลัพธ์การแบ่งแยกข้อมูลตามแนวตั้ง

**ตารางที่ 3.3** เงื่อนไขการแบ่งแยกข้อมูลตามแนวตั้ง

ตาราง CUSTOMER							
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
10	Sinex, Inc.	12 Main St.	TN	\$3,500.00	\$2,700.00	3	\$1,245.00
11	Martin Corp.	321 Sunset Blvd.	FL	\$6,000.00	\$1,200.00	1	\$0.00
12	Mynux Corp.	910 Eagle St.	TN	\$4,000.00	\$3,500.00	3	\$3,400.00
13	BTBC, Inc.	Rue du Monde	FL	\$6,000.00	\$5,890.00	3	\$1,090.00
14	Victory, Inc.	123 Maple St.	FL	\$1,200.00	\$550.00	1	\$0.00
15	NBCC Corp.	909 High Ave.	GA	\$2,000.00	\$350.00	2	\$50.00

เงื่อนไข

FRAGMENT NAME	ATTRIBUTE NAMES
CUST_V1	CUS_NUM,CUS_NAME,CUS_ADDRESS,CUS_STATE
CUST_V2	CUS_NUM,CUS_LIMIT,CUS_BAL,CUS_RATING,CUS_DUE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.4 ผลลัพธ์การแบ่งแยกข้อมูลตามแนวตั้ง

ตาราง CUST_V1			
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE
10	Sinex, Inc.	12 Main St.	TN
11	Martin Corp.	321 Sunset Blvd.	FL
12	Mynux Corp.	910 Eagle St.	TN
13	BTBC, Inc.	Rue du Monde	FL
14	Victory, Inc.	123 Maple St.	FL
15	NBCC Corp.	909 High Ave.	GA

ตาราง CUST_V2				
CUS_NUM	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
10	\$3,500.00	\$2,700.00	3	\$1,245.00
11	\$6,000.00	\$1,200.00	1	\$0.00
12	\$4,000.00	\$3,500.00	3	\$3,400.00
13	\$6,000.00	\$5,890.00	3	\$1,090.00
14	\$1,200.00	\$550.00	1	\$0.00
15	\$2,000.00	\$350.00	2	\$50.00

จากตารางจะเห็นว่า การแบ่งแยกข้อมูลตามแนวตั้ง จากตาราง CUSTOMER ซึ่งประกอบด้วยลักษณะประจำ CUS\_NUM CUS\_NAME CUS\_ADDRESS CUS\_STATE CUS\_LIMIT CUS\_RATING CUS\_DUE ได้เป็น 2 ตาราง คือ CUST\_V1 และ CUST\_V2 ซึ่งทั้งสองตารางประกอบด้วยข้อมูลทุกแถวเฉพาะลักษณะประจำที่ถูกระบุ เท่านั้น นั่นคือ ตาราง CUST\_V1 เป็นการเลือกทุกแถวเฉพาะลักษณะประจำ CUS\_NUM CUS\_NAME CUS\_ADDRESS CUS\_STATE จากตาราง CUSTOMER ส่วนตาราง CUST\_V2 เป็นการเลือกทุกแถวเฉพาะลักษณะประจำ CUS\_NUM CUS\_LIMIT CUS\_RATING CUS\_DUE จากตาราง CUSTOMER

**3.3.2.3 การแบ่งแยกข้อมูลแบบผสม** การแบ่งแยกข้อมูลแบบผสม (mixed fragmentation) หมายถึงการแบ่งแถวของข้อมูลโดยเลือกมาเฉพาะบางลักษณะประจำ ดังตารางที่ 3.5 แสดงเงื่อนไขการแบ่งแยกข้อมูลแบบผสม และตารางที่ 3.6 แสดงผลลัพธ์การแบ่งแยกข้อมูลแบบผสม

### ตารางที่ 3.5 เงื่อนไขการแบ่งแยกข้อมูลแบบผสม

ตาราง CUSTOMER							
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
10	Sinex, Inc.	12 Main St.	TN	\$3,500.00	\$2,700.00	3	\$1,245.00
11	Martin Corp.	321 Sunset Blvd.	FL	\$6,000.00	\$1,200.00	1	\$0.00
12	Mynux Corp.	910 Eagle St.	TN	\$4,000.00	\$3,500.00	3	\$3,400.00
13	BTBC, Inc.	Rue du Monde	FL	\$6,000.00	\$5,890.00	3	\$1,090.00
14	Victory, Inc.	123 Maple St.	FL	\$1,200.00	\$550.00	1	\$0.00
15	NBCC Corp.	909 High Ave.	GA	\$2,000.00	\$350.00	2	\$50.00

เงื่อนไข

FRAGMENT NAME	HORIZONTAL CRITERIA	RESULTING ROWS	ATTRIBUTE NAMES
CUST_M1	CUS_STATE = 'TN'	10,12	CUS_NUM,CUS_NAME,CUS_ADDRESS,CUS_STATE
CUST_M2	CUS_STATE = 'TN'	10,12	CUS_NUM,CUS_LIMIT,CUS_BAL,CUS_RATING,CUSDUE
CUST_M3	CUS_STATE = 'GA'	15	CUS_NUM,CUS_NAME,CUS_ADDRESS,CUS_STATE
CUST_M4	CUS_STATE = 'GA'	15	CUS_NUM,CUS_LIMIT,CUS_BAL,CUS_RATING,CUSDUE
CUST_M5	CUS_STATE = 'FL'	11,13,14	CUS_NUM,CUS_NAME,CUS_ADDRESS,CUS_STATE
CUST_M6	CUS_STATE = 'FL'	11,13,14	CUS_NUM,CUS_LIMIT,CUS_BAL,CUS_RATING,CUSDUE

### ตารางที่ 3.6 ผลลัพธ์การแบ่งแยกข้อมูลแบบผสม

ตาราง CUST_M1			
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE
10	Sinex, Inc.	12 Main St.	TN
12	Mynux Corp.	910 Eagle St.	TN

ตาราง CUST_M2				
CUS_NUM	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
10	\$3,500.00	\$2,700.00	3	\$1,245.00
12	\$4,000.00	\$3,500.00	3	\$3,400.00

ตาราง CUST_M3			
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE
15	NBCC Corp.	909 High Ave.	GA

ตาราง CUST_M4				
CUS_NUM	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
15	\$2,000.00	\$350.00	2	\$50.00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.6 (ต่อ)

ตาราง CUST_M5			
CUS_NUM	CUS_NAME	CUS_ADDRESS	CUS_STATE
11	Martin Corp.	321 Sunset Blvd.	FL
13	BTBC, Inc.	Rue du Monde	FL
14	Victory, Inc.	123 Maple St.	FL

ตาราง CUST_M6				
CUS_NUM	CUS_LIMIT	CUS_BAL	CUS_RATING	CUS_DUE
11	\$6,000.00	\$1,200.00	1	\$0.00
13	\$6,000.00	\$5,890.00	3	\$1,090.00
14	\$1,200.00	\$550.00	1	\$0.00

จากตารางจะเห็นว่า การแบ่งแยกข้อมูลแบบผสม จากตาราง CUSTOMER ซึ่งประกอบด้วยลักษณะประจำ CUS\_NUM CUS\_NAME CUS\_ADDRESS CUS\_STATE CUS\_LIMIT CUS\_RATING CUS\_DUE ได้เป็น 6 ตาราง คือ CUST\_M1 CUST\_M2 CUST\_M3 CUST\_M4 CUST\_M5 และ CUST\_M6 ซึ่งทั้งหกตารางประกอบด้วยข้อมูลบางแถวที่ตรงตามเงื่อนไข เฉพาะลักษณะประจำที่ถูกระบุ เท่านั้น นั่นคือ ตาราง CUST\_M1 เป็นการเลือกแถวที่มี CUS\_STATE = 'TN' เฉพาะลักษณะประจำ CUS\_NUM CUS\_NAME CUS\_ADDRESS CUS\_STATE จากตาราง CUSTOMER ตาราง CUST\_M2 เป็นการเลือกแถวที่มี CUS\_STATE = 'TN' เฉพาะลักษณะประจำ CUS\_NUM CUS\_LIMIT CUS\_RATING CUS\_DUE จากตาราง CUSTOMER

ตาราง CUST\_M3 เป็นการเลือกแถวที่มี CUS\_STATE = 'GA' เฉพาะลักษณะประจำ CUS\_NUM CUS\_NAME CUS\_ADDRESS CUS\_STATE จากตาราง CUSTOMER ตาราง CUST\_M4 เป็นการเลือกแถวที่มี CUS\_STATE = 'GA' เฉพาะลักษณะประจำ CUS\_NUM CUS\_LIMIT CUS\_RATING CUS\_DUE จากตาราง CUSTOMER

ตาราง CUST\_M5 เป็นการเลือกแถวที่มี CUS\_STATE = 'FL' เฉพาะลักษณะประจำ CUS\_NUM CUS\_NAME CUS\_ADDRESS CUS\_STATE จากตาราง CUSTOMER ตาราง CUST\_M6 เป็นการเลือกแถวที่มี CUS\_STATE = 'FL' เฉพาะลักษณะประจำ CUS\_NUM CUS\_LIMIT CUS\_RATING CUS\_DUE จากตาราง CUSTOMER

### 3.4 กระบวนการสืบค้นข้อมูลแบบกระจาย

ในระบบฐานข้อมูลแบบรวมศูนย์ ประสิทธิภาพของการสืบค้นข้อมูลจะวัดจากปริมาณของการเข้าถึงข้อมูลในสื่อบันทึกข้อมูล แต่ในระบบฐานข้อมูลแบบกระจายจะต้องพิจารณาเพิ่มเติม เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกคือ ค่าใช้จ่ายในการส่งข้อมูลผ่านระบบเครือข่าย และ ประสิทธิภาพของแต่ละแห่งที่ประมวลผลคำสั่งแบบขนาน ความสัมพันธ์ของค่าใช้จ่ายระหว่างการส่งข้อมูลผ่านระบบเครือข่ายและการส่งข้อมูลจากสื่อบันทึกข้อมูล จะขึ้นอยู่กับประเภทของระบบเครือข่าย และความเร็วของสื่อบันทึกข้อมูล ดังนั้นเราไม่สามารถที่จะระบุลงไปได้เลยว่าจะต้องเสียค่าใช้จ่ายไปกับสื่อบันทึกข้อมูล หรือระบบเครือข่ายมากกว่ากัน กระบวนการสืบค้นข้อมูลแบบกระจายมีดังนี้

### 3.4.1 การสืบค้นข้อมูลด้วยภาษาสอบถาม

เมื่อต้องการแสดงข้อมูลทุกแถวของตาราง CUSTOMER จะเห็นว่าในระบบฐานข้อมูลแบบรวมศูนย์จะเป็นแบบสอบถามที่ง่าย แต่ในระบบฐานข้อมูลแบบกระจาย การประมวลผลแบบสอบถามนี้จะค่อนข้างยุ่งยากในการประมวลผล เนื่องจากตาราง CUSTOMER อาจจะถูกทำสำเนาไว้หลายๆ แห่งหรือถูกแบ่งออกเป็นหลายๆ ตารางย่อย หรือถูกทำทั้งสองอย่าง ถ้าตาราง CUSTOMER ถูกทำสำเนาและไม่ได้ถูกแบ่งออกเป็นตารางย่อย เราก็จะเลือกสำเนาที่มีค่าใช้จ่ายในการส่งข้อมูลน้อยที่สุด อย่างไรก็ตามถ้าสำเนาของตารางมีการแบ่งออกเป็นตารางย่อยด้วย การเลือกก็จะมี ความยุ่งยากเพิ่มขึ้น เนื่องจากเราต้องทำการรวมตารางด้วยการจอย (join) หรือ ยูเนียน (union) เพื่อสร้างตาราง CUSTOMER ดังตัวอย่าง

ตัวอย่าง การเรียกดูข้อมูลจากตาราง CUSTOMER ที่มี CUST\_STATE='FL' ผู้ใช้อาจจะเขียนแบบสอบถามเป็น

$$\delta_{\text{CUST\_STATE='FL'}}(\text{CUSTOMER})$$

และเนื่องจาก ตาราง CUSTOMER ถูกกำหนดดังนี้ CUST\_H1 U CUST\_H2 U CUST\_H3 สามารถเขียนได้ดังนี้

$$\delta_{\text{CUST\_STATE='FL'}}(\text{CUST\_H1 U CUST\_H2 U CUST\_H3})$$

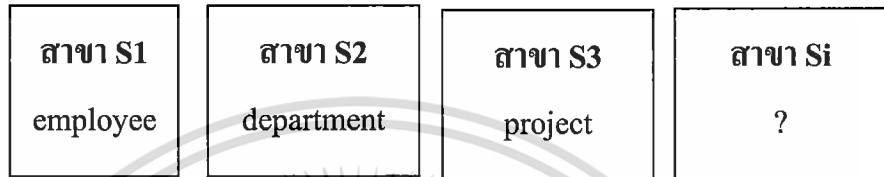
หรือ

$$\delta_{\text{CUST\_STATE='FL'}}(\text{CUST\_H1}) \cup \delta_{\text{CUST\_STATE='FL'}}(\text{CUST\_H2}) \cup \delta_{\text{CUST\_STATE='FL'}}(\text{CUST\_H3})$$

จะเห็นว่าข้อมูลที่ต้องการนั้นมีในตาราง CUST\_H1 เพียงตารางเดียว

### 3.4.2 การรวมตารางด้วยการจอย

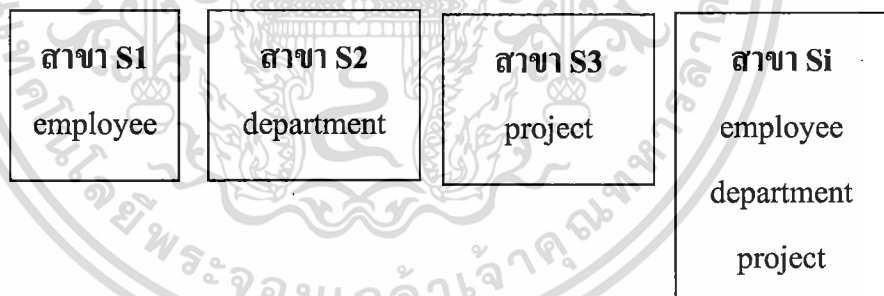
การรวมตารางด้วยการจอย (simple join processing) เมื่อพิจารณาจากนิพจน์ employee  $\bowtie$  department  $\bowtie$  project สมมติว่าทั้งสามตารางไม่ได้ถูกทำสำเนาและไม่ได้ถูกแบ่งเป็นตารางย่อย และตาราง employee ถูกเก็บไว้ที่สาขา  $S_1$ , ตาราง department เก็บไว้ที่สาขา  $S_2$  และตาราง project เก็บไว้ที่สาขา  $S_3$  และกำหนด  $S_i$  เป็นสาขาที่จะส่งผลลัพธ์ของแบบสอบถามไปให้ ดังรูปที่ 3.4 แสดงที่อยู่ของตารางข้อมูล



รูปที่ 3.4 ที่อยู่ของตารางข้อมูล

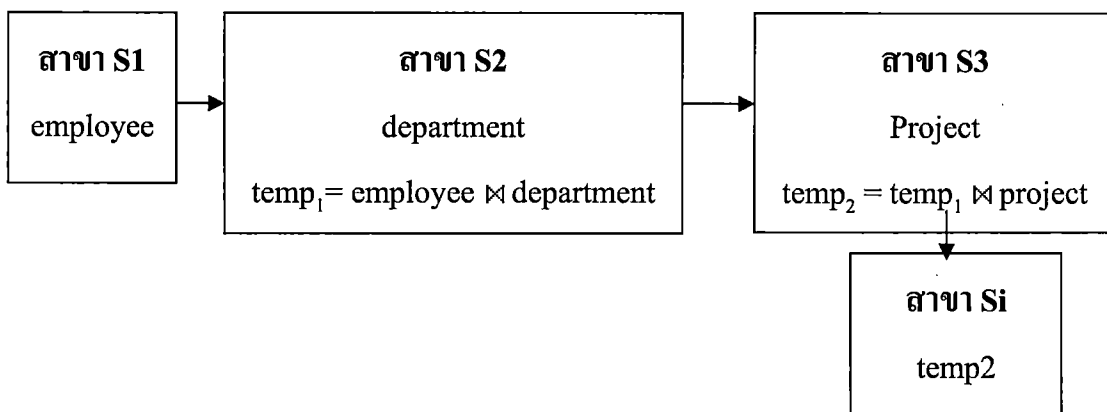
ดังนั้นวิธีการที่เป็นไปได้สำหรับประมวลผลแบบสอบถามนี้คือ

3.4.2.1 ส่งสำเนาของทั้งสามตารางไปที่สาขา  $S_i$  และใช้เทคนิคต่าง ๆ ในการในการสืบค้นข้อมูลที่สาขา  $S_i$  ดังรูปที่ 3.5 แสดงการสำเนาข้อมูลไปไว้ที่สาขา  $S_i$



รูปที่ 3.5 การสำเนาข้อมูลไปไว้ที่สาขา  $S_i$

3.4.2.2 ส่งสำเนาของตาราง employee ไปที่สาขา  $S_2$  และทำการประมวลผล  $temp_1 = employee \bowtie department$  ที่สาขา  $S_2$  จากนั้นส่ง  $temp_1$  จากสาขา  $S_2$  ไปยังสาขา  $S_3$  และประมวลผล  $temp_2 = temp_1 \bowtie project$  และส่ง  $temp_2$  ไปยังสาขา  $S_i$  ดังรูปที่ 3.6 แสดงการสำเนาข้อมูลที่ละสาขา



รูปที่ 3.6 การสำเนาข้อมูลทีละสาขา

### 3.4.2.3 ทำในลักษณะคล้ายๆ กับวิธีการที่สอง แต่สลับสาขาในการส่งข้อมูล

เมื่อพิจารณาแล้วจะพบว่า ต้องพิจารณาระหว่างปริมาณของข้อมูลที่จะต้องส่งระหว่างสาขา ค่าใช้จ่ายในการส่งผ่านข้อมูลระหว่างสองสาขา และความเร็วในการประมวลผลของแต่ละสาขา ซึ่งในวิธีการแรก ถ้าส่งข้อมูลทั้งหมดไปที่สาขา  $S_1$  โดยที่ตารางเหล่านั้นมีการสร้างดัชนีจำเป็นที่จะต้องสร้างดัชนีเหล่านั้นที่สาขา  $S_1$  ด้วย ซึ่งการสร้างดัชนีทำให้มีการประมวลผลเพิ่มขึ้นมา และยังมีการใช้สื่อบันทึกข้อมูลเพิ่มขึ้นอีก อย่างไรก็ตาม วิธีการที่สองก็มี ข้อเสียคือตารางที่ได้จากการทำประมวลผลจะมีขนาดใหญ่ ( $employee \bowtie department$ ) ซึ่งต้องส่งข้อมูลจากสาขา  $S_2$  ไปที่สาขา  $S_3$  ซึ่งวิธีการที่สองจะทำให้มีการส่งข้อมูลบนระบบเครือข่ายมากกว่า เมื่อเทียบกับวิธีการที่หนึ่ง

### 3.4.3 ยุทธศาสตร์การเซมิจอย

ยุทธศาสตร์การเซมิจอย (semijoin strategy) แนวคิดการทำแบบสอบถามแบบกระจาย โดยใช้วิธีการทำเซมิจอย มีจุดประสงค์เพื่อลดจำนวนของแถวใน ตารางก่อนที่จะทำการส่งให้สาขาอื่น สมมติว่าเราต้องการประมวลผลนิพจน์  $r_1 \bowtie r_2$  ซึ่ง  $r_1$  และ  $r_2$  เก็บอยู่ที่สาขา  $S_1$  และ  $S_2$  ตามลำดับ กำหนดให้  $R_1$  และ  $R_2$  แทนเค้าร่างของ  $r_1$  และ  $r_2$  ตามลำดับ สมมติว่าเราต้องการผลลัพธ์ที่  $S_1$  ถ้ามีแถวหลาย ๆ แถวใน  $r_2$  ที่ไม่ได้ join กับแถวใด ๆ ใน  $r_1$  ดังนั้นการส่งตาราง  $r_2$  ทั้งหมดไปที่สาขา  $S_1$  ก็จะเป็นการส่งแถวที่ทำให้เกิดการ join เกิดผลลัพธ์ที่เกินมาได้ ดังนั้น ก่อนที่เราจะส่งข้อมูลจาก  $r_2$  ไปที่สาขา  $S_1$  ก็น่าจะส่งเฉพาะแถวสามารถ join กับ ตาราง  $r_1$  ที่สาขา  $S_1$  เท่านั้น เราสามารถดำเนินการดังกล่าวได้ดังนี้

3.1 ทหา  $temp_1 \beta P_{R_1 \cap R_2}(r_1)$  ที่สาขา  $S_1$

3.2 ส่ง  $temp_1$  จากสาขา  $S_1$  ไป  $S_2$

3.3 ทหา  $temp_2 \beta r_2 \bowtie temp_1$  ที่สาขา  $S_2$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ส่ง  $temp_2$  จากสาขา  $S_2$  ไป  $S_1$

3.5 ประมวลผล  $r_1 \bowtie temp_2$  ที่สาขา  $S_1$

ในขั้นตอนที่ 3.3  $temp_2$  สามารถหาได้จาก  $r_2 \bowtie P_{R_1 \cap R_2}(r_1)$  และในขั้นตอนที่ 3.5 ทำ  $r_1 \bowtie r_2 \bowtie P_{R_1 \cap R_2}(r_1)$  เราสามารถเขียนนิพจน์ใหม่ได้ดังนี้

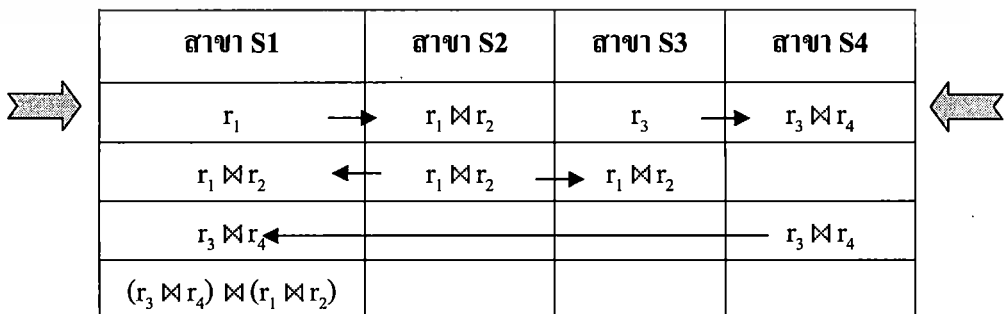
$$(r_1 \bowtie P_{R_1 \cap R_2}(r_1)) \bowtie r_2$$

เนื่องจาก  $r_1 \bowtie P_{R_1 \cap R_2}(r_1) = r_1$  ดังนั้นนิพจน์นี้จะเท่ากับ  $r_1 \bowtie r_2$  วิธีการนี้จะมีเหมาะสมในกรณีที่จำนวนของแถวของ  $r_2$  มีจำนวนน้อย วิธีการดำเนินการแบบ เซมิจอยแทนด้วยสัญลักษณ์  $\bowtie$  ดังนั้นเซมิจอยของ  $r_1$  และ  $r_2$  เขียนแทนด้วย  $r_1 \bowtie r_2$  คือ  $P_{R_1}(r_1 \bowtie r_2)$  ดังนั้น  $r_1 \bowtie r_2$  จะเป็นการเลือกแถวของ  $r_1$  เพื่อทำการ  $r_1 \bowtie r_2$  ซึ่งในขั้นตอนที่ 3.3 จะสามารถเขียนได้ใหม่ดังนี้  $temp_2 = r_2 \bowtie r_1$

**3.4.4 ยุทธศาสตร์การจอยแบบขนาน**

ยุทธศาสตร์การจอยแบบขนาน (join strategies that exploit parallelism) พิจารณาการจอยของตาราง 4 ตาราง  $r_1 \bowtie r_2 \bowtie r_3 \bowtie r_4$  ซึ่งตาราง  $r_1$  จะถูกเก็บไว้ที่สาขา  $S_1$  สมมติว่าเราต้องการผลลัพธ์ที่สาขา  $S_1$  ซึ่งก็มีอยู่หลายวิธีการที่จะดำเนินการแบบขนาน ยกตัวอย่างเช่น  $r_1$  ถูกส่งไปที่สาขา  $S_2$  และทำ  $r_1 \bowtie r_2$  ที่สาขา  $S_2$  ในขณะเดียวกัน  $r_3$  ก็ถูกส่งไปที่สาขา  $r_4$  และทำ  $r_3 \bowtie r_4$  ที่สาขา  $S_4$  สาขา  $S_2$  จะส่งผลลัพธ์  $(r_1 \bowtie r_2)$  กลับไปที่สาขา  $S_1$  และสาขา  $S_3$  สาขา  $S_4$  จะส่งผลลัพธ์  $(r_3 \bowtie r_4)$  กลับไปที่สาขา  $S_1$  เมื่อ  $S_1$  ได้รับผลลัพธ์กลับจาก  $S_2$  และ  $S_4$  แล้ว ก็จะทำการ join อีกครั้ง  $(r_3 \bowtie r_4) \bowtie (r_1 \bowtie r_2)$  ดังนั้นจะเห็นว่าการประมวลผลการ join ที่  $S_1$  สามารถดำเนินการแบบขนานโดยมีการประมวลผลที่สาขา  $S_2$  และสาขา  $S_4$  ไปพร้อมๆ กัน ดังตารางที่ 3.7 แสดงการจอยแบบขนาน

ตารางที่ 3.7 การจอยแบบขนาน



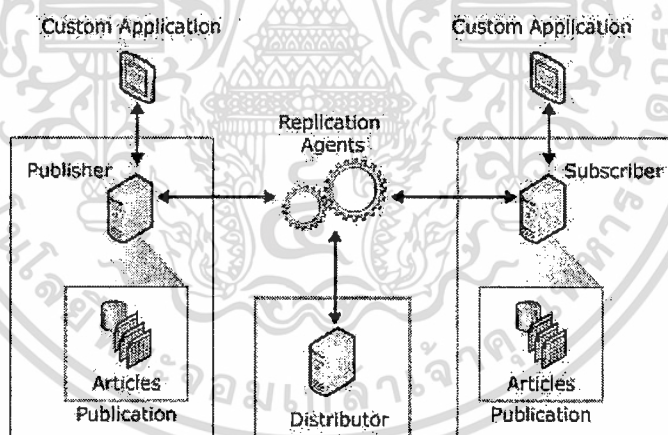
### 3.5 การทำสำเนาฐานข้อมูลในไมโครซอฟท์ เอสคิวแอลเซิร์ฟเวอร์(Microsoft SQL Server Replication)

การทำ Replication มีจุดประสงค์เพื่อกระจายข้อมูลไปยังเครื่องให้บริการอื่นๆ อย่างมีประสิทธิภาพนั้น มีสิ่งที่จะต้องคำนึงถึง ดังนี้

1. ความถูกต้องของข้อมูลในแต่ละเครื่องให้บริการ ในกรณีที่มีการกระจายข้อมูลไปยังเครื่องให้บริการอื่นๆ แล้วเกิดปัญหาข้อมูลขัดแย้งกันระหว่างต้นทางกับปลายทาง เราขอรับได้เพียงใด เช่น ในเครื่องบริการปลายทางมีการแก้ไขข้อมูลในตารางลูกค้า เมื่อเครื่องบริการปลายทางมีการแก้ไขข้อมูลในตารางลูกค้า เมื่อเครื่องบริการต้นทางกระจายข้อมูลมาที่เครื่องบริการปลายทาง ก็จะเกิดปัญหาข้อมูลขัดแย้งกันเกิดขึ้น เป็นต้น ซึ่งปัญหาดังกล่าวสามารถบริหารจัดการ โดยการกำหนดลำดับความสำคัญของข้อมูล

2. ระดับความเป็นอิสระของเครื่องบริการปลายทางต่อการกระจายข้อมูลจากเครื่องบริการต้นทางว่า ต้องทำบ่อยครั้งเพียงใด

3. ช่วงเวลาในการกระจายข้อมูลจากเครื่องบริการต้นทางว่าจะทำทันทีที่มีการเปลี่ยนแปลงข้อมูลในเครื่องบริการต้นทางหรือตามเวลาที่กำหนด



รูปที่ 3.7 แสดงบทบาทของเครื่องบริการในการสำเนาฐานข้อมูล

#### 3.5.1 บทบาทของเครื่องบริการในการทำสำเนาฐานข้อมูล

ในการทำการทำสำเนาฐานข้อมูลใน SQL Server 2008 จะต้องกำหนดบทบาทให้กับเครื่องบริการซึ่งต้องกำหนดบทบาทให้กับเครื่องบริการซึ่งมีตามที่แสดงรูปที่ 3.4 ดังนี้

3.5.1.1 Publisher จะเป็นบทบาทสำหรับเครื่องบริการที่เป็นต้นทาง มีหน้าที่ในการจัดเตรียมข้อมูลที่จะกระจายในการสำเนาข้อมูล ซึ่งเรียกข้อมูลนั้นว่า Published Data และทำหน้าที่ในการตรวจสอบเงื่อนไขในการสำเนาฐานข้อมูล รวมถึงส่ง Published Data ไปยัง

3.5.1.2 Distributor จะอยู่ในเครื่องบริการที่มีฐานข้อมูล distribution ซึ่ง Distributor สามารถอยู่ในเครื่องเดียวกับ Publisher ได้ หรืออาจจะอยู่ต่างเครื่องกันก็ได้ Distributor ทำหน้าที่ในการเก็บ Metadata ประวัติของข้อมูลและรายการเปลี่ยนแปลงในการสำเนาฐานข้อมูล รวมถึงยังทำหน้าที่รับ Published Data จาก Publisher แล้วส่งต่อไปยัง Subscriber บทบาทของ Distributor นั้นจะขึ้นอยู่กับประเภทของการทำสำเนาข้อมูลซึ่ง Distributor จะมีบทบาทมากใน Snapshot Replication และ Transaction Replication

3.5.1.3 Subscriber จะอยู่ในเครื่องบริการปลายทาง มีหน้าที่รับ Published Data จาก Distributor มาทำการแก้ไขข้อมูลในเครื่องบริการปลายทาง ซึ่ง Subscriber อาจจะทำการแก้ไขข้อมูลที่เครื่องบริการปลายทางแล้วทำสำเนากลับไปยัง Publisher เพื่อให้ Publisher ทำการสำเนาข้อมูลต่อไปยัง Subscriber อื่นๆได้ ซึ่งเรียกว่า Updating Subscriber

### 3.5.2 ปัจจัยที่เกี่ยวข้องกับการทำสำเนาฐานข้อมูล

ในการทำสำเนาฐานข้อมูล มีปัจจัยเกี่ยวข้องหลายประการ ประกอบด้วย

3.5.2.1 Article หน่วยของข้อมูลที่ย่อยที่สุดที่ใช้ในการสำเนาฐานข้อมูล สามารถเป็นไปได้ทั้งส่วนใดส่วนหนึ่งของตาราง ตารางออบเจกต์ในฐานข้อมูล รวมถึง Stored Procedure, View และ User-defined Function

3.5.2.2 Filtering Data (Partition) ใน Article สามารถเป็นข้อมูลส่วนใดส่วนหนึ่งของตารางได้ เรียกว่า Filtering Data ใช้เพื่อหลีกเลี่ยงปัญหาความขัดแย้งของข้อมูลในการสำเนาฐานข้อมูล สามารถกรองข้อมูลที่เรากำลังต้องการได้ 2 แนวคือ แนวตั้ง (Vertical Filtering) เลือกเฉพาะข้อมูลในสดมภ์ที่ต้องการเท่านั้น และแนวนอน (Horizontal Filtering) เลือกเอาเฉพาะข้อมูลในแถวที่ต้องการเท่านั้น

3.5.2.3 Publication ประกอบด้วย Article หนึ่งหรือหลายตัว Publication นั้นสามารถแบ่งประเภทตามลักษณะการทำงาน แต่ละประเภทมีรายละเอียด ดังนี้

1. Snapshot Publication โดย Publisher จะทำการส่งข้อมูลทั้งหมดของออบเจกต์ทั้งข้อมูลที่ถูกรับแก้ไขและไม่ถูกรับแก้ไข ไปแทนที่ข้อมูลใน Subscriber

2. Transactional Publication โดย Publisher จะส่งข้อมูลเฉพาะรายการที่เปลี่ยนแปลงของออบเจกต์ที่ข้อมูลถูกรับแก้ไขเท่านั้น ไปยัง Subscriber

3. Transactional Publication with updateable Subscription มีลักษณะคล้ายกับ Transactional Publication แต่แตกต่างกันที่ Subscriber มีการแก้ไขข้อมูลก็สามารถส่งข้อมูลรายการที่เปลี่ยนแปลงนั้นกลับมาแก้ไขข้อมูลใน Publisher ได้

4. Merge Publication ทั้ง Publisher และ Subscriber สามารถแก้ไขข้อมูลได้อย่างอิสระ เมื่อข้อมูลเกิดความขัดแย้งกัน จะใช้ลำดับความสำคัญ(Priority) ในการผสานข้อมูลของ Publisher กับ Subscriber เข้าด้วยกัน

ในฐานะข้อมูลหนึ่งๆสามารถมีหลาย Publication ได้ แต่ Publication มีข้อจำกัดหลายประการ ดังนี้

- Stored Procedure, View, Trigger และ User-defined Function ที่มีการใช้เงื่อนไข WITH ENCRYPTION จะไม่สามารถเป็น Article ใน Publication ได้

- Transactional Replication ตารางที่เป็น Article ใน publication จะต้องเป็น Primary Key และจะไม่สามารถเป็น Disable Index ของสคีมที่เป็น Primary Key ได้ เนื่องจากในการทำสำเนาฐานข้อมูลจะต้องมีการใช้งาน index ด้วย

- Merge Replication ใน publication หนึ่งๆ จะประกอบด้วย Article ได้สูงสุดเพียง 256 Article เท่านั้น

- Snapshot Replication และ Transactional Replication ตารางที่จะทำสำเนา จะมีจำนวนสคีมสูงสุดได้ 1,000 สคีม ส่วน Merge Replication จะมีจำนวนสคีมสูงสุดได้ 246 สคีม

- ฐานข้อมูล master msdb model tempdb และ distribution ไม่สามารถทำสำเนาข้อมูลได้

3.5.2.4 Subscription เป็นการใช้งาน Publication ที่สร้างไว้ในการทำสำเนาข้อมูลไปยังเครื่องบริการปลายทาง ซึ่งประกอบด้วย 2 วิธี ดังนี้

1. Push Subscription เป็นการกำหนดให้เครื่องบริการต้นทางส่งข้อมูลไปยังเครื่องบริการปลายทาง เมื่อถึงเวลาที่กำหนด โดยกำหนด Push Subscription ที่ Publisher เพื่อใช้เป็นศูนย์กลางในการกระจายข้อมูล ซึ่งจะง่ายในการบริหารจัดการ แต่ถ้าข้อมูลมีจำนวนมากและกระจายไปยังเครื่องบริการปลายทางหลายเครื่อง ก็จะเป็นภาระของเครื่องบริการ ที่อาจจะกระทบกับประสิทธิภาพโดยรวมของระบบในฝั่งของเครื่องบริการต้นทางได้

2. Pull Subscription เป็นการกำหนดให้เครื่องบริการต้นทางมาดึงข้อมูลจากเครื่องต้นทาง เมื่อถึงเวลาที่กำหนด คือกำหนด Pull Subscription ที่ Subscriber ซึ่งเราสามารถกำหนดให้ Subscriber ดึงข้อมูลเมื่อถึงเวลาที่ต้องการได้ ทำให้ไม่มีผลกระทบกับประสิทธิภาพโดยรวมของเครื่องบริการต้นทางและเครื่องบริการต้นทางสามารถทำงานรองรับกับ Subscriber จำนวนมากได้ แต่ Pull Subscription เหมาะกับงานที่ต้องการความปลอดภัยของข้อมูลในระดับต่ำ (Low Security)

### 3.6 ปัญหาของระบบงานปัจจุบัน

สามารถสรุป ปัญหาของระบบสำเนาฐานข้อมูลแบบกระจาย ได้ดังนี้

- 3.2.1 การแก้ไขข้อมูลที่เก็บซ้ำกัน ต้องคอยแก้ไขตามจุดของเครื่องต่าง ๆ
- 3.2.2 ความซับซ้อนในการประมวลผลการเรียกใช้ข้อมูล การใช้คำสั่ง ยุ่งยาก ลำช้า และค่าใช้จ่ายในการบำรุงรักษาสูง
- 3.2.3 ความซับซ้อนในการควบคุมภาวะพร้อมกัน การขอใช้ และแก้ไขมีขั้นตอน มาก ยุ่งยาก เสียเวลา
- 3.2.4 ความซับซ้อนในการฟื้นฟูสภาพ เมื่อมีการเปลี่ยนแปลงข้อมูล จำเป็นต้อง แก้ไขข้อมูลตามจุดต่าง ๆ
- 3.2.5 ความยุ่งยากในการดูแลจัดการพจนานุกรมข้อมูล เช่น การเก็บ พจนานุกรมฉบับสมบูรณ์ไว้ที่สาขาเดียว การเก็บพจนานุกรมฉบับสมบูรณ์ไว้ทุกสาขา และการเก็บ พจนานุกรมบางส่วนกระจายอยู่ตามสาขาต่างๆ เกิดความลำบากในการเรียก ใช้พจนานุกรมจาก สาขาอื่น เพราะต้องสิ้นเปลืองเวลาในการสื่อสารข้อมูล
- 3.2.6 การควบคุมความถูกต้องตรงกัน ในข้อมูลเป็นไป ได้ค่อนข้างยาก
- 3.2.7 ยังไม่มีหลักการหรือมาตรฐานการจัดการที่แน่นอนและชัดเจน
- 3.2.8 ขาดแคลนผู้เชี่ยวชาญหรือผู้ที่มีประสบการณ์
- 3.2.9 การออกแบบฐานข้อมูลมีความซับซ้อนสูง

### 3.7 คุณลักษณะของเครื่องมือตรวจสอบการสำเนาฐานข้อมูล

โปรแกรมตรวจสอบการสำเนาฐานข้อมูลของ Microsoft SQL Server 2008 เป็นเครื่องมือ แบบกราฟิกที่ช่วยให้สามารถตรวจสอบการทำงาน โดยรวมของโครงสร้างการสำเนาข้อมูล การ ตรวจสอบรายละเอียดเกี่ยวกับสถานะและประสิทธิภาพการทำงานของฐานข้อมูลที่ใช้ในการสำเนา ข้อมูล(Publisher) และฐานข้อมูลที่รับสำเนาข้อมูล (Subscriber) มีคุณลักษณะดังต่อไปนี้

- 3.6.1 การแสดงภาพรวมของการสำเนาฐานข้อมูล สามารถอธิบายแต่ละหน้าต่างและแท็บ เพื่อการตรวจสอบการสำเนาข้อมูลได้
- 3.6.2 แสดงสถานะฐานข้อมูลที่ใช้ในการสำเนาข้อมูล(Publisher) และฐานข้อมูลที่รับ สำเนาข้อมูล (Subscriber) เพื่อการตรวจสอบการสำเนาข้อมูล นำข้อมูลของสถานะจะปรากฏขึ้นใน โปรแกรมการตรวจสอบการสำเนาข้อมูล
- 3.6.3 การกำหนดเกณฑ์การตั้งค่าและค่าเตือนในการตรวจสอบการสำเนาข้อมูล อธิบาย ค่าเตือนและเกณฑ์และการแจ้งเตือนคุณสามารถตั้งค่าในการตรวจสอบการสำเนาข้อมูล ค่าแนะนำ

ในการใช้งาน คำเตือนสำหรับโครงสร้างเครือข่ายเพื่อให้ข้อมูลเกี่ยวกับสถานะและประสิทธิภาพการทำงาน

3.6.4 การตรวจสอบประสิทธิภาพการสำเนาข้อมูล วิธีการตรวจสอบประสิทธิภาพการสำเนาข้อมูล การกำหนดค่าข้อมูลเกี่ยวข้องกับประสิทธิภาพ ข้อมูลเกี่ยวกับสถิติของวิธีการสำเนาข้อมูลแบบพาส แสดงรายละเอียดของการประมวลผล

3.6.5 การวัดและตรวจสอบการเชื่อมต่อสำหรับการสำเนาข้อมูลแบบทรานแซกชัน อธิบายการติดตามการสำเนาข้อมูล โครงสร้างการทรานแซกชัน

3.6.6 การตรวจสอบการสำเนาข้อมูลเอเจนท์ สามารถอธิบายถึงวิธีการหาข้อมูลเกี่ยวกับเอเจนท์แต่ละ replication

3.6.7 แคมเปญการตรวจสอบประสิทธิภาพการสำเนาข้อมูล อธิบายวิธีการสำเนาข้อมูล ข้อมูลและการตรวจสอบและการคำนวณเพื่อปรับปรุงประสิทธิภาพ อธิบายถึงวิธีการรีเฟรชของส่วนติดต่อผู้ใช้ที่เกี่ยวข้องกับการฟื้นฟูของแคม

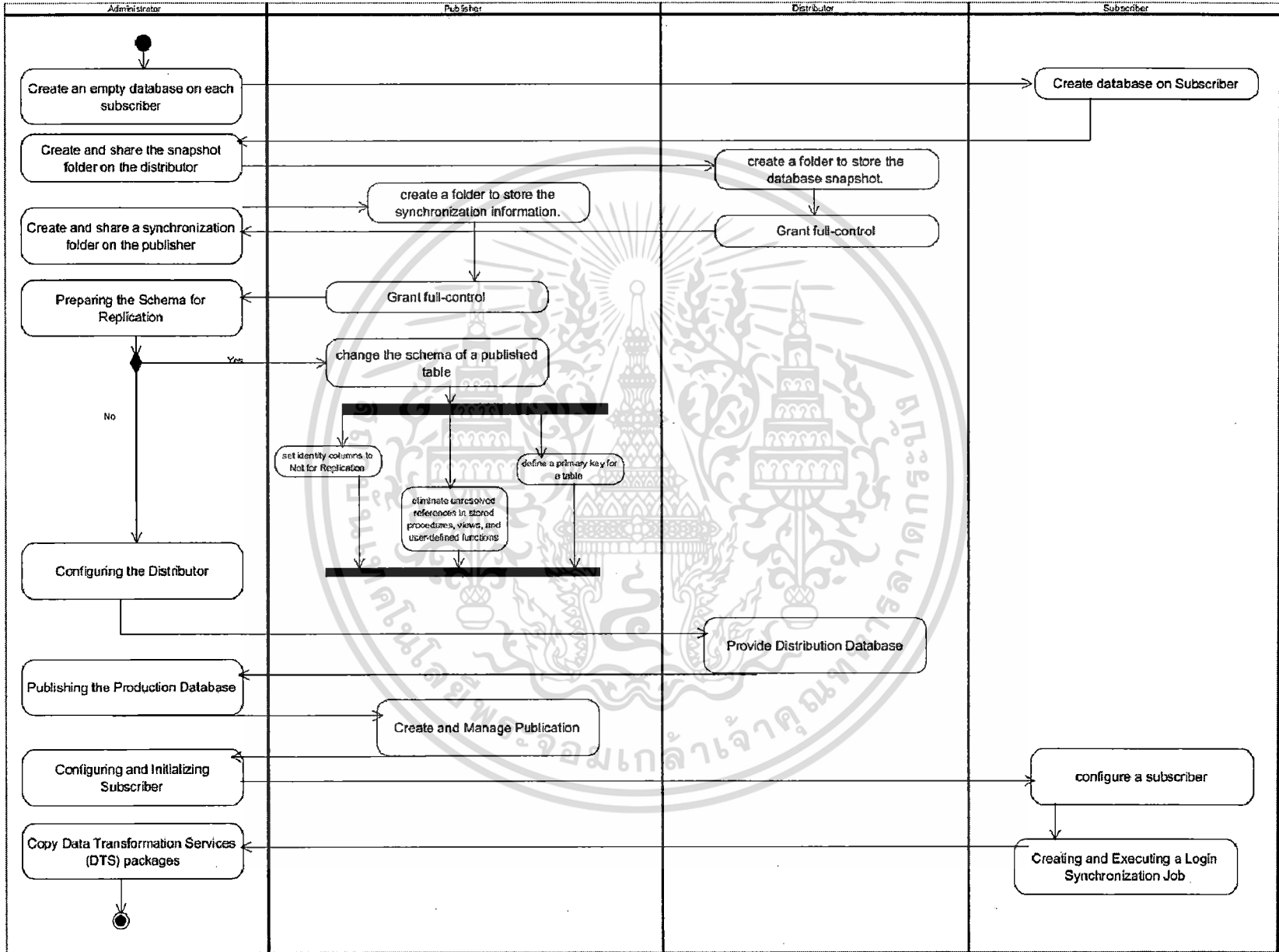
จากการศึกษาคุณลักษณะของโปรแกรมตรวจสอบการสำเนาฐานข้อมูลของ Microsoft SQL Server 2008 สามารถประยุกต์ใช้งานและลักษณะซอฟต์แวร์ที่สามารถแสดงข้อมูลสำเนาข้อมูลได้ แต่ยังคงขาดเครื่องมือที่สนับสนุนการพิจารณาโครงสร้างฐานข้อมูลสำหรับการทำสำเนาและการแสดงข้อมูลการเปลี่ยนข้อมูลหน่วยของข้อมูล (ARTICLE) ของฐานข้อมูลที่ทำสำเนา หากละเว้นรายละเอียดส่วนนี้ ทำให้มีความเสี่ยงต่อความถูกต้องของข้อมูล

### 3.8 แนวทางการแก้ไขปัญหา

การพัฒนากระบวนการข้อมูลเชิงเวลา เพื่อเก็บการใช้สถิติเพื่อการตรวจสอบและช่วยการจัดการในการทำสำเนาฐานข้อมูล เมื่อมีการเรียกหรือจัดการข้อมูลในระดับ โกลบอล

เนื่องจากระบบต้องการความสมบูรณ์ของฐานข้อมูลที่ซ่อนแยกข้อมูลทางกายภาพ การแยกข้อมูลทางลอจิกและคุณสมบัติเฉพาะของระบบสำเนาฐานข้อมูลไว้จากผู้ใช้งาน ระบบสามารถจัดการ การปรับปรุง โครงสร้างตารางสัมพันธ์และการอัปเดตซึ่งจะมีการเคลื่อนย้ายข้อมูลในระดับการภาพโดยที่อยู่ต่างที่กัน

รูปที่ 3.8 ขั้นตอนการทำสำเนาฐานข้อมูลในปัจจุบัน



ภาพรวมของขั้นตอนการทำสำเนาฐานข้อมูลในปัจจุบัน สามารถแสดงได้ดังรูปที่ 3.8

จากปัญหาที่พบดังข้างต้นสามารถสรุปแนวทางการแก้ไขปัญหาได้ ดังตารางที่ 3.8

ตารางที่ 3.8 แนวทางการแก้ไขปัญหา

ปัญหา	แนวทางการแก้ไขปัญหา
3.8.1 การแก้ไขข้อมูลที่เก็บซ้ำกัน ต้องคอยแก้ไขตามจุดของเครื่องต่าง ๆ	ใช้ระบบสารสนเทศที่สามารถเปรียบเทียบความถูกต้องขององค์ประกอบบนฐานข้อมูล (ดูหัวข้อ 4.3 ในส่วนของคำอธิบายยูสเคสเปรียบเทียบฐานข้อมูลที่ทำสำเนา หน้า 46)
3.8.2 ความซับซ้อนในการประมวลผลการเรียกใช้ข้อมูล การใช้คำสั่ง ยุ่งยาก ลำบาก และค่าใช้จ่ายในการบำรุงรักษาสูง	ระบบสามารถแสดงข้อมูลกิจกรรมของการใช้งานบนระบบสำเนาฐานข้อมูลตามช่วงเวลาต่างๆ (ดูหัวข้อ 4.2 ในส่วนของคำอธิบายยูสเคสดูสถิติของกิจกรรมการใช้งาน หน้า 46)
3.8.3 ความซับซ้อนในการควบคุมภาวะพร้อมกัน การขอใช้ และแก้ไขมีขั้นตอนมาก ยุ่งยาก เสียเวลา	ระบบสามารถแสดงข้อมูลกิจกรรมของการใช้งานบนระบบสำเนาฐานข้อมูลตามช่วงเวลาต่างๆ (ดูหัวข้อ 4.2 ในส่วนของคำอธิบายยูสเคสดูสถิติของกิจกรรมการใช้งาน หน้า 46)
3.8.4 ความซับซ้อนในการฟื้นฟูสภาพ เมื่อมีการเปลี่ยนแปลงข้อมูล จำเป็นต้องแก้ไขข้อมูลตามจุดต่าง ๆ	ใช้ระบบสารสนเทศที่ระบุโครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ถูกรับปรุงจากฐานข้อมูลที่ใช้ในการสำเนา (ดูหัวข้อ 4.4 ในส่วนของคำอธิบายยูสเคสนำเข้าข้อมูลโครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ใช้ในการสำเนา หน้า 47)
3.8.5 ความยุ่งยากในการดูแลจัดการพจนานุกรมข้อมูล เช่น การเก็บพจนานุกรมฉบับสมบูรณ์ไว้ที่สาขาเดียว การเก็บพจนานุกรมฉบับสมบูรณ์ไว้ทุกสาขา และการเก็บพจนานุกรมบางส่วนกระจายอยู่ตามสาขาต่างๆ เกิดความลำบากในการเรียก ใช้พจนานุกรมจากสาขาอื่น เพราะต้องสิ้นเปลืองเวลาในการสื่อสารข้อมูล	ใช้ระบบสารสนเทศที่แก้ไขรายการสำเนาฐานข้อมูล รวมถึงแสดงรายละเอียดข้อมูลของการทำสำเนาฐานข้อมูล (ดูหัวข้อ 4.1 ในส่วนของคำอธิบายยูสเคสจัดการรายละเอียดการสำเนาข้อมูล หน้า 42)

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 3.8 (ต่อ)

ปัญหา	แนวทางการแก้ไขปัญหา
3.8.6 การควบคุมความถูกต้องตรงกันในข้อมูล เป็นไปได้ค่อนข้างยาก	ใช้ระบบสารสนเทศที่สามารถเปรียบเทียบความถูกต้องขององค์ประกอบบนฐานข้อมูล (ดูหัวข้อ 4.3 ในส่วนของคำอธิบายยูสเคสเปรียบเทียบฐานข้อมูลที่ทำสำเนา หน้า 46)
3.8.7 ยังไม่มีหลักการหรือมาตรฐานการจัดการที่แน่นอนและชัดเจน	เครื่องมือช่วยให้การปรับเปลี่ยนฐานข้อมูลธรรมดาเป็นฐานข้อมูลเชิงเวลาได้ โดยมีผลกระทบกับแอปพลิเคชันน้อยที่สุด (ดูหัวข้อ 4.5 ในส่วนของคำอธิบายยูสเคสนำเข้าข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา หน้า 49)
3.8.8 ขาดแคลนผู้เชี่ยวชาญหรือผู้ที่มีประสบการณ์	ใช้เครื่องมือและแนวทางฐานข้อมูลเชิงเวลาที่ช่วยรักษาความถูกต้องของข้อมูลปกติเสมือนเป็นมุมมองปัจจุบันของตารางข้อมูลเชิงเวลา
3.8.9 การออกแบบฐานข้อมูลมีความซับซ้อนสูง	ระบบใช้งานง่าย โดยเครื่องมือและแนวทางฐานข้อมูลเชิงเวลาที่แยกฐานข้อมูลธรรมดากับฐานข้อมูลเชิงเวลาออกจากกัน (ดูหัวข้อ 4.6 ในส่วนของคำอธิบายยูสเคสบันทึกข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา หน้า 50)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การวิเคราะห์และออกแบบระบบ

การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลเป็นระบบที่ช่วยให้ผู้ดูแลระบบสามารถบริหารจัดการเกี่ยวกับ การตรวจสอบการเปลี่ยนแปลงขององค์ประกอบภายในฐานข้อมูลหลักและแสดงข้อมูลเปรียบเทียบความสมบูรณ์ของข้อมูลระหว่างฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูลกับฐานข้อมูลที่รับการสำเนาฐานข้อมูล เพื่อช่วยประกอบการพิจารณาในสำหรับการส่งผ่านข้อมูลระหว่างแหล่งข้อมูลจะส่งเฉพาะข้อมูลที่จำเป็นหรือข้อมูลที่มีการปรับปรุงเท่านั้น โดยจะแสดงการออกแบบระบบงานใหม่ในภาพไดอะแกรมแบบต่างๆ รวมไปถึงการออกแบบพจนานุกรมข้อมูล ของข้อมูลภายในฐานข้อมูลที่พัฒนาขึ้น ดังที่จะแสดงต่อไปนี้

1. การวิเคราะห์ความต้องการของระบบงาน
2. ขอบเขตการทำงานของระบบใหม่
3. การวิเคราะห์และออกแบบระบบงาน

#### 4.1 การวิเคราะห์ความต้องการของระบบงาน

การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลที่จะพัฒนาขึ้นนั้น จะมีการออกแบบให้ตรงกับความต้องการของผู้ใช้ ซึ่งมีรายละเอียดดังต่อไปนี้

4.1.1 ระบบสามารถปรับปรุงและจัดเก็บข้อมูลของเครื่องที่ให้บริการสำเนาฐานข้อมูลบนระบบเครือข่ายได้ โดยอาศัยคุณสมบัติการทำงานของฐานข้อมูลเชิงเวลา

4.1.2 ระบบสามารถตรวจสอบและเปรียบเทียบความถูกต้องขององค์ประกอบบนฐานข้อมูลที่ทำสำเนาฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) กับฐานข้อมูลที่รับการสำเนาฐานข้อมูล (Subscriber) เพื่อช่วยให้ผู้ดูแลฐานข้อมูลการพิจารณาการสำเนาฐานข้อมูลของรายการออบเจกต์ที่เปลี่ยนแปลงไปได้ถูกต้องและครบถ้วน

4.1.3 ระบบสามารถจัดเก็บและแสดงข้อมูลการใช้งานฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) เพื่อใช้ในการวางแผนและจัดทำการสำเนาฐานข้อมูลที่เหมาะสม

4.1.4 การตรวจสอบและแสดงข้อมูลการเปลี่ยนแปลงของโครงสร้างองค์ประกอบภายในฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) เมื่อการเริ่มทำสำเนาฐานข้อมูลแล้ว (Reinitialize Subscription) เพื่อหลีกเลี่ยงช่วงเวลาทำสำเนาข้อซึ่งทำให้เกิดผลกระทบกับการใช้งานเครื่องบริการสำเนาฐานข้อมูล (Publisher)

## 4.2 ขอบเขตการทำงานของระบบใหม่

ระบบงานใหม่ที่จะทำการพัฒนาขึ้นนั้น เป็นการพัฒนาโดยมุ่งเน้นให้สอดคล้องกับความต้องการของผู้ใช้ แต่เนื่องจากความต้องการของข้อมูลเพื่อใช้ประกอบการพิจารณาของผู้ใช้ต่อระบบใหม่นั้นมีปริมาณมาก ดังนั้น จึงต้องมีการกำหนดขอบเขตระบบงานใหม่ โดยในขั้นต้นแรกจะพัฒนาในส่วนที่จำเป็นเท่านั้น หากพัฒนาระบบในขั้นต้นแรกเสร็จสิ้นแล้ว ก็จะค่อยพัฒนาระบบในส่วนถัดไป ดังนั้น ระบบใหม่จึงมีขอบเขตดังนี้

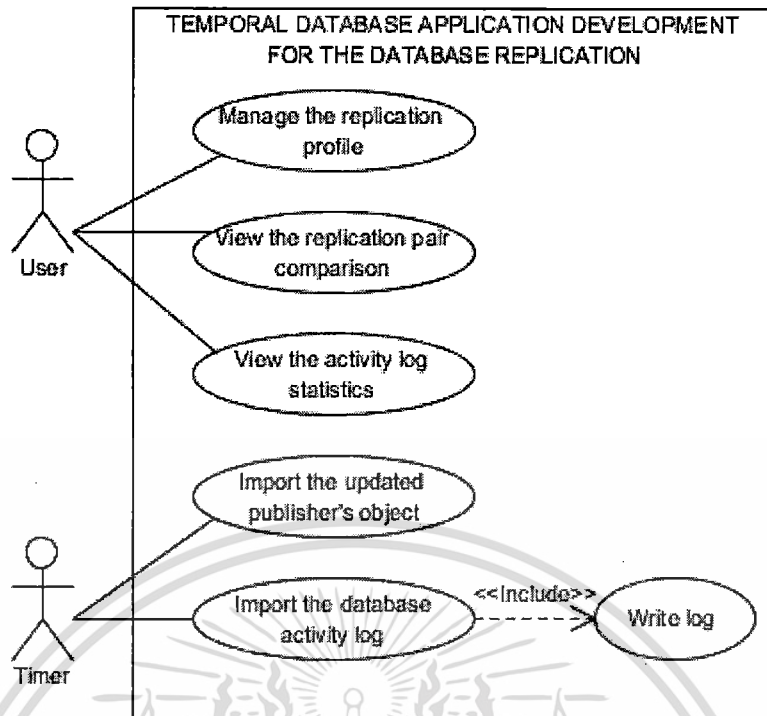
1. ระบบต้องใช้งานง่าย
2. ผู้ใช้สามารถเข้าสู่ระบบได้จาก โดยใช้สิทธิ์ที่กำหนดเป็นผู้บริหารฐานข้อมูลขององค์กร
3. สามารถทำการบันทึกข้อมูลข้อมูลรายละเอียดของเครื่องบริการทำสำเนาข้อมูลและแนะนำวิธีการฐานข้อมูลที่มีการกำหนดชนิดของสำเนาข้อมูลไว้
4. ผู้ใช้สามารถตรวจสอบสถานะของการและสถิติการใช้งานฐานข้อมูลได้
5. ผู้ใช้สามารถตรวจสอบและเปรียบเทียบความถูกต้องขององค์ประกอบบนฐานข้อมูล เพื่อช่วยให้ผู้ดูแลฐานข้อมูลการพิจารณาการสำเนาฐานข้อมูลของรายการออบเจกต์ที่เปลี่ยนแปลงไปได้ถูกต้องและครบถ้วน
6. ระบบมีการรักษาความปลอดภัยที่ดีพอสมควร โดยจะต้องมีการล็อกอินด้วย ชื่อผู้ใช้และรหัสผู้ใช้ทุกครั้งเพื่อเข้าสู่ระบบ

## 4.3 การวิเคราะห์และออกแบบระบบงาน

### 4.3.1 ยูสเคสไดอะแกรม

ยูสเคสไดอะแกรม เป็นแผนภาพที่ช่วยอธิบายกิจกรรมหลักและขอบเขตของระบบเพื่อให้เห็นภาพรวมของการทำงานของระบบ จากการศึกษาและออกแบบการพัฒนาบบเพื่อสนับสนุนระบบการสำเนาฐานข้อมูลจะประกอบด้วยแอกเตอร์และยูสเคสต่างๆ ดังแสดงในรูปที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 ยูสเคสโคดอะแกรมของระบบ

สำหรับแอกเตอร์ คือ ผู้ใช้งานยูสเคส หรือกระทำกับยูสเคสมีทั้งหมด 2 แอกเตอร์ คือ

1. User คือ ผู้ใช้งานระบบ ได้แก่ ผู้บริหารและจัดการฐานข้อมูล
2. Timer คือ เวลาของระบบจัดการฐานข้อมูลที่ทำหน้าที่ตรวจสอบและบันทึกกิจกรรม

ฐานข้อมูล

สำหรับหน้าที่และการทำงานหลักของระบบ จะประกอบด้วยทั้งหมด 6 ยูสเคส ดังต่อไปนี้

1. ยูสเคสจัดการรายละเอียดการสำเนาข้อมูล(Manage the replication profile) คือ การเพิ่มและแก้ไขรายการสำเนาฐานข้อมูล รวมถึงแสดงรายละเอียดข้อมูลของการทำสำเนาฐานข้อมูล

2. ยูสเคสเปรียบเทียบฐานข้อมูลที่ทำสำเนา (View the replication pair comparison) คือ การแสดงจำนวนและความถูกต้องขององค์ประกอบของฐานข้อมูลที่ทำสำเนาระบบ โดยเปรียบเทียบระหว่างฐานข้อมูลที่ใช้ในการสำเนา กับฐานข้อมูลที่รับการสำเนา

3. ยูสเคสดูสถิติของกิจกรรมการใช้งาน(View the activity log statistics) คือ การแสดงข้อมูลกิจกรรมของการใช้งานบนระบบสำเนาฐานข้อมูลตามช่วงเวลาต่างๆ

4. ยูสเคสนำเข้าข้อมูล โครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ใช้ในการสำเนา(Import the updated publisher's object) คือ การนำเข้าข้อมูล โครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ถูกปรับปรุง

จากฐานข้อมูลที่ใช้ในการสำเนาเข้าสู่ระบบ เพื่อใช้ในการตรวจสอบย้อนหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ยูสเคสนำเข้าข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา (Import the database activity log) คือ การจัดเก็บข้อมูลล็อกการใช้งานบนฐานข้อมูลที่ทำสำเนาเข้าสู่ระบบ เพื่อการตรวจสอบข้อมูล ช่วงเวลาที่มีปริมาณการใช้งานตามช่วงเวลาต่างๆ รวมถึงประกอบการพิจารณาของผู้บริหาร ฐานข้อมูลในการเลือกช่วงเวลาที่เหมาะสมในการเริ่มทำสำเนาฐานข้อมูลใหม่ (Reinitialize Subscription)

6. ยูสเคสบันทึกข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา (Write log) คือ การบันทึกข้อมูล เริ่มต้นและสิ้นสุดกระบวนการใช้งานฐานข้อมูลที่ทำสำเนา

จากยูสเคสไดอะแกรม ดังรูปที่ 4.1 สามารถเขียนคำบรรยายยูสเคส เพื่อแสดง รายละเอียดของการทำงานแต่ละยูสเคส ดังตารางที่ 4.1 ถึง 4.6 ตามลำดับ

ตารางที่ 4.1 คำอธิบายยูสเคสจัดการรายละเอียดการสำเนาข้อมูล

Use-Case Description	
Use Case Name:	Manage the replication profile
Brief Description:	ผู้ใช้เพิ่มและแก้ไขรายการสำเนาฐานข้อมูล รวมถึงแสดงรายละเอียด ข้อมูลของการทำสำเนาฐานข้อมูล
Primary actor:	User
Stakeholders:	-
Precondition:	พนักงานสามารถเพิ่มและแก้ไขรายละเอียดการสำเนาฐานข้อมูลของ ระบบตามที่สร้างบนระบบจัดการฐานข้อมูลไว้ล่วงหน้า
Postcondition:	ระบบบันทึกข้อมูลของการสำเนาข้อมูลลงฐานข้อมูลของระบบ และ แจ้งให้ผู้ใช้พนักงานทราบว่าปรับปรุงข้อมูลตามที่ต้องการแล้ว
Trigger:	ผู้ใช้ต้องการเพิ่มและแก้ไขรายการสำเนาฐานข้อมูลเข้าสู่ระบบ
Actor Action	System Action
<p>Typical Cause of Event:</p> <p>Step 1: ผู้ใช้กรอกชื่อเครื่องของฐานข้อมูลที่ใช้ ในการสำเนาฐานข้อมูล (Publisher) ชื่อผู้ใช้และ รหัสผ่าน โดยผู้ใช้ต้องถูกกำหนดสิทธิ์การทำงานให้ระดับผู้สร้างฐานข้อมูล (DB_OWNER) เป็นอย่างน้อย</p>	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1(ต่อ)

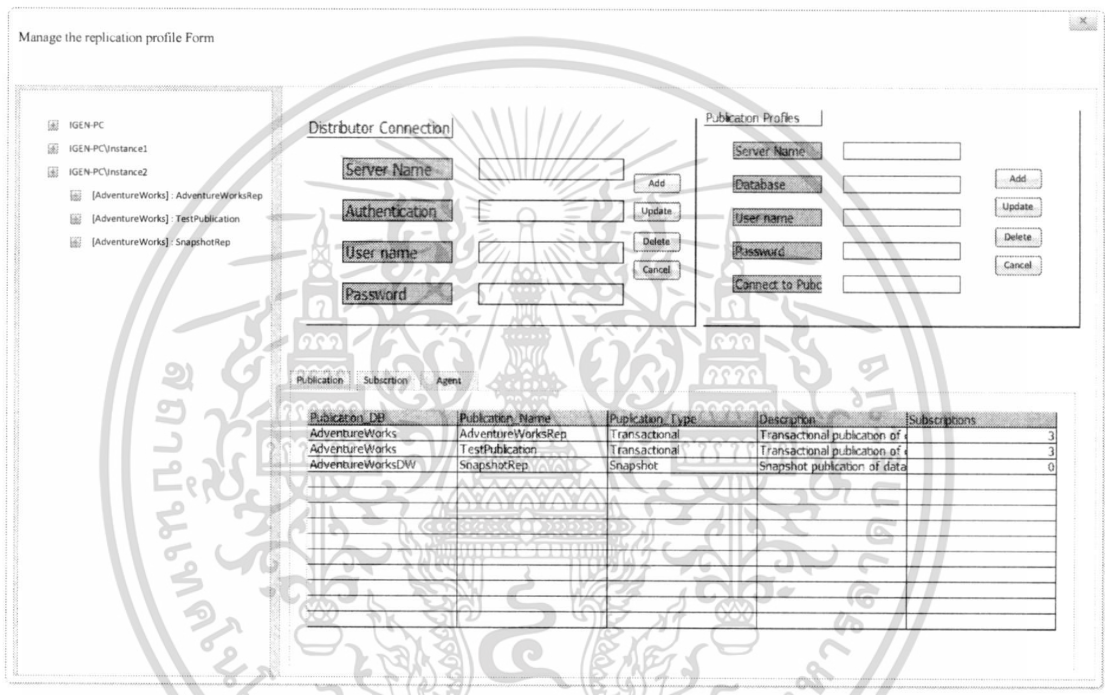
<p>Step 3: ผู้ใช้กรอกชื่อเครื่องตัวกลาง (Distributor) ชื่อผู้ใช้และรหัสผ่าน โดยผู้ใช้ต้องถูกกำหนดสิทธิ์การทำงานให้ระดับผู้สร้างฐานข้อมูล(DB_OWNER)เป็นอย่างน้อย</p> <p>Step 5: ผู้ใช้กรอกชื่อเครื่องฐานข้อมูลที่รับการสำเนาฐานข้อมูล (Subscriber) ชื่อผู้ใช้และรหัสผ่าน โดยผู้ใช้ต้องถูกกำหนดสิทธิ์การทำงานให้ระดับผู้สร้างฐานข้อมูล (DB_OWNER)เป็นอย่างน้อย</p> <p>Step 7: ผู้ใช้งานกดปุ่มยืนยันการบันทึกข้อมูล</p>	<p>Step 2: ระบบจะตรวจสอบข้อมูลชื่อเครื่องชื่อผู้ใช้และรหัสผ่าน ว่าข้อมูลที่ได้รับครบและถูกต้อง</p> <p>Step 4: ระบบจะตรวจสอบข้อมูลชื่อเครื่องชื่อผู้ใช้และรหัสผ่าน ว่าข้อมูลที่ได้รับครบและถูกต้อง</p> <p>Step 6: ระบบจะตรวจสอบข้อมูลชื่อเครื่องชื่อผู้ใช้และรหัสผ่าน ว่าข้อมูลที่ได้รับครบและถูกต้อง</p> <p>Step 8: ถ้าการสำเนาฐานข้อมูลที่กรอกไม่เกิดความผิดพลาด ระบบบันทึกข้อมูลลงฐานข้อมูล</p> <p>Step 9: ระบบแสดงผลการแก้ไขข้อมูลและแสดงข้อความยืนยันการบันทึก</p>
<p>Alternate Cause of Event:</p> <p>Step 2a: ถ้าข้อมูลที่ผู้ใช้กรอกเกิดความผิดพลาด เช่น รหัสผ่านเครื่องฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (Publisher) ไม่ถูกต้อง ระบบจะแจ้งเตือนให้ระบบแก้ไขข้อมูลให้ถูกต้องเหมาะสม หรือถ้าไม่ต้องการแก้ไขต่อยุสเคสนี้จะสิ้นสุดการทำงาน</p> <p>Step 4a: ถ้าข้อมูลที่ผู้ใช้กรอกเกิดความผิดพลาด เช่น รหัสผ่านเครื่องตัวกลาง (Distributor) ไม่ถูกต้อง ระบบจะแจ้งเตือนให้ระบบแก้ไขข้อมูลให้ถูกต้องเหมาะสม หรือถ้าไม่ต้องการแก้ไขต่อ</p>	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 4.1(ต่อ)

Alternate Cause of Event:

Step 6a: ถ้าข้อมูลที่ใช้กรอกเกิดความผิดพลาด เช่น รหัสผ่านเครื่องฐานข้อมูลที่รับการสำเนาฐานข้อมูล (Subscriber) ไม่ถูกต้อง ระบบจะแจ้งเตือนให้ระบบแก้ไขข้อมูลให้ถูกต้องเหมาะสม หรือถ้าไม่ต้องการแก้ไขต่อยูสเคสนี้จะสิ้นสุดการทำงาน



รูปที่ 4.2 แสดงหน้าจการทำงานการจัดการรายละเอียดการสำเนาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 คำอธิบายยูสเคสคูสติของกิจกรรมการใช้งาน

Use-Case Description	
Use Case Name:	View the activity log statistics
Brief Description:	เป็นการแสดงข้อมูลกิจกรรมของการใช้งานบนระบบสำเนาฐานข้อมูลตามช่วงเวลาต่างๆ
Primary actor:	User
Stakeholders:	-
Precondition:	-
Postcondition:	ข้อมูลกิจกรรมของการใช้งานฐานข้อมูลถูกแสดงออกมา
Trigger:	-
Actor Action	System Action
Typical Cause of Event: Step 1: ผู้ใช้เลือกฐานข้อมูลที่ใช้ในการสำเนา	Step 2: ระบบดึงข้อมูลกิจกรรมของการใช้งานตามเงื่อนไขของเวลาที่ผู้ใช้กำหนด Step 3: ระบบแสดงข้อมูลการใช้งานบนหน้าจอ
Alternate Cause of Event:	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

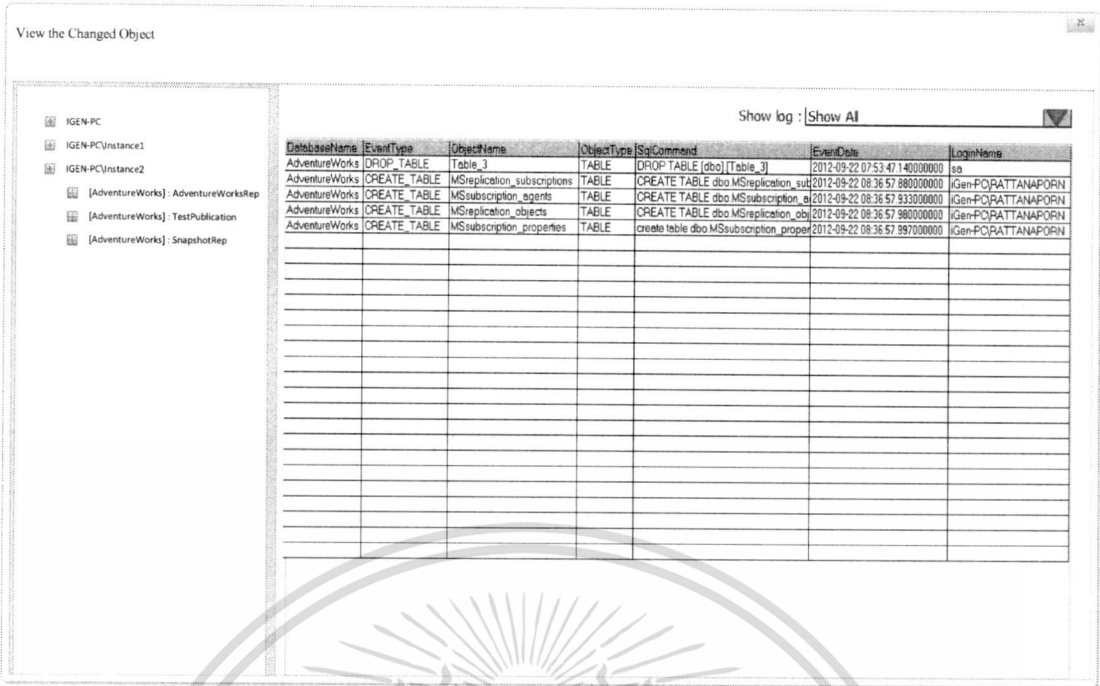
### ตารางที่ 4.3 คำอธิบายยูสเคสเปรียบเทียบฐานข้อมูลที่ทำสำเนา

Use-Case Description	
Use Case Name:	View the replication pair comparison
Brief Description:	แสดงจำนวนและความถูกต้องของโครงสร้างอ็อบเจกต์ของฐานข้อมูลที่ทำสำเนาระบบ โดยเปรียบเทียบระหว่างฐานข้อมูลที่ใช้ในการสำเนา กับฐานข้อมูลที่รับการสำเนา
Primary actor:	User
Stakeholders:	-
Precondition:	ต้องมีรายการฐานข้อมูลที่ใช้ในการสำเนา กับฐานข้อมูลที่รับการสำเนา อยู่ในฐานข้อมูลของระบบก่อน
Postcondition:	ข้อมูลผลการเปรียบเทียบถูกแสดงออกมาตามที่กำหนดไว้
Trigger:	ผู้ใช้ต้องการทราบตามแตกต่างของโครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ทำสำเนาระบบ
Actor Action	System Action
<p>Typical Cause of Event:</p> <p>Step 1: ผู้ใช้เลือกรายการสำเนา กับฐานข้อมูลที่ต้องการตรวจสอบ</p> <p>Step 3: ผู้ใช้ยืนยันเริ่มการเปรียบเทียบฐานข้อมูลที่รับการสำเนา</p> <p>Step 5: ผู้ใช้ดูข้อมูลผลการเปรียบเทียบ</p>	<p>Step 2: ระบบตรวจสอบการเข้าสู่เครื่องของระบบสำเนาฐานข้อมูลตามที่กำหนดไว้</p> <p>Step 4: ระบบแสดงผลการเปรียบเทียบตามระบบสำเนา กับฐานข้อมูลที่เลือกไว้</p>
<p>Alternate Cause of Event:</p> <p>Step 2a: ถ้าระบบไม่สามารถติดต่อไปยังเครื่องของระบบสำเนาฐานข้อมูลตามที่กำหนด ยูสเคสนี้จะสิ้นสุดลง</p>	

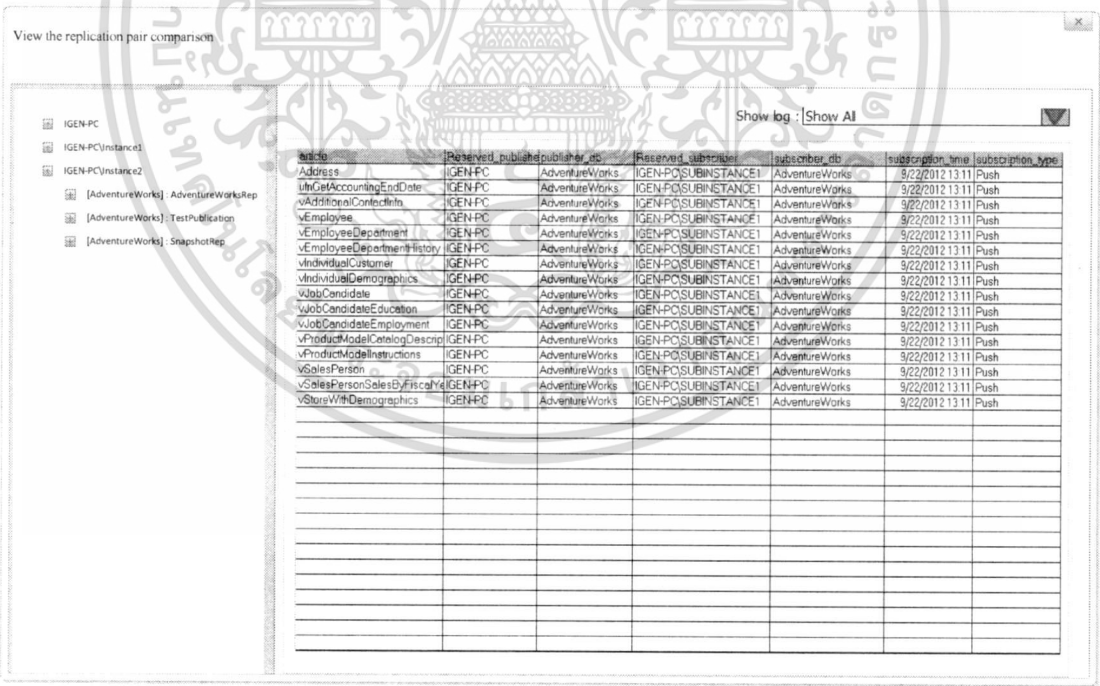
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 คำอธิบายยูสเคสนำเข้าข้อมูลโครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ใช้ในการสำเนา

Use-Case Description	
Use Case Name:	Import the updated publisher's object
Brief Description:	การนำเข้าข้อมูล โครงสร้างอ็อบเจกต์บนฐานข้อมูลที่ถูกปรับปรุงจากฐานข้อมูลที่ใช้ในการสำเนาเข้าสู่ระบบ เพื่อใช้ในการตรวจสอบย้อนหลัง
Primary actor:	Timer
Stakeholders:	-
Precondition:	อ็อบเจกต์บนฐานข้อมูลที่ถูกปรับปรุง โครงสร้างบนฐานข้อมูลที่ใช้ในการสำเนาในไม่เป็นสถานะถูกปิดกั้น(lock)
Postcondition:	เพิ่มรายการข้อมูล โครงสร้างอ็อบเจกต์ที่เปลี่ยนแปลงใหม่
Trigger:	ครบรอบระยะเวลา การตรวจสอบตามที่ผู้ใช้กำหนด
Actor Action	System Action
<p>Typical Cause of Event:</p> <p>Step 1: ครบกำหนดเวลาการตรวจสอบโครงสร้างอ็อบเจกต์ที่เปลี่ยนแปลง</p>	<p>Step 2: ระบบติดต่อเข้าสู่ระบบฐานข้อมูลที่ใช้ในการสำเนาตามที่กำหนด</p> <p>Step 3: ระบบตรวจสอบเวลาที่โครงสร้างอ็อบเจกต์และเลือกอ็อบเจกต์ที่เวลาปรับปรุงโครงสร้างล่าสุดกว่าที่ระบบได้บันทึกไว้ก่อนหน้านี้</p> <p>Step 4: ตรวจสอบแล้วพบว่าข้อมูลที่จำเป็นครบถ้วนและบันทึกข้อมูลเข้าสู่ระบบ</p>
<p>Alternate Cause of Event:</p> <p>Step 2a: ถ้าระบบไม่สามารถติดต่อไปยังเครื่องของระบบสำเนาฐานข้อมูลตามที่กำหนด ยูสเคสนี้จะสิ้นสุดลง</p>	



รูปที่ 4.3 แสดงหน้าจออ็อบเจกต์บนฐานข้อมูลที่ถูกปรับปรุง



รูปที่ 4.4 แสดงหน้าจอการเปรียบเทียบฐานข้อมูลที่ทำสำเนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 คำอธิบายยูสเคสนำเข้าข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา

Use-Case Description	
Use Case Name:	Import the database activity log
Brief Description:	การจัดเก็บข้อมูลล็อกการใช้งานบนฐานข้อมูลที่ทำสำเนาเข้าสู่ระบบ เพื่อการตรวจสอบข้อมูลช่วงเวลาที่มึปริมาณการใช้งานตามช่วงเวลาต่างๆ รวมถึงประกอบการพิจารณาของผู้บริหารฐานข้อมูลในการเลือกช่วงเวลาที่เหมาะสมในการเริ่มทำสำเนาฐานข้อมูลใหม่ (Reinitialize Subscription)
Primary actor:	Timer
Stakeholders:	-
Precondition:	-
Postcondition:	-
Trigger:	ครบกำหนดรอบตามระยะเวลาที่ผู้ดูแลระบบกำหนด
Actor Action	System Action
Typical Cause of Event:	
Step 1: ครบกำหนดระยะเวลาการตรวจสอบและเก็บข้อมูลการใช้งานของผู้ใช้	Step 2: : ระบบติดต่อเข้าสู่ระบบฐานข้อมูลที่ใช้ในการสำเนาตามที่กำหนด Step 3: ระบบเริ่มการรวบรวมข้อมูลการใช้งาน Step 6: บันทึกข้อมูลเข้าสู่ระบบ
Step 5: กำหนดเวลาในการทำงานครั้งถัดไป	
Alternate Cause of Event:	
Step 2a: ถ้าระบบไม่สามารถติดต่อไปยังเครื่องของระบบสำเนาฐานข้อมูลตามที่กำหนด ยูสเคสนี้จะสิ้นสุดลง	

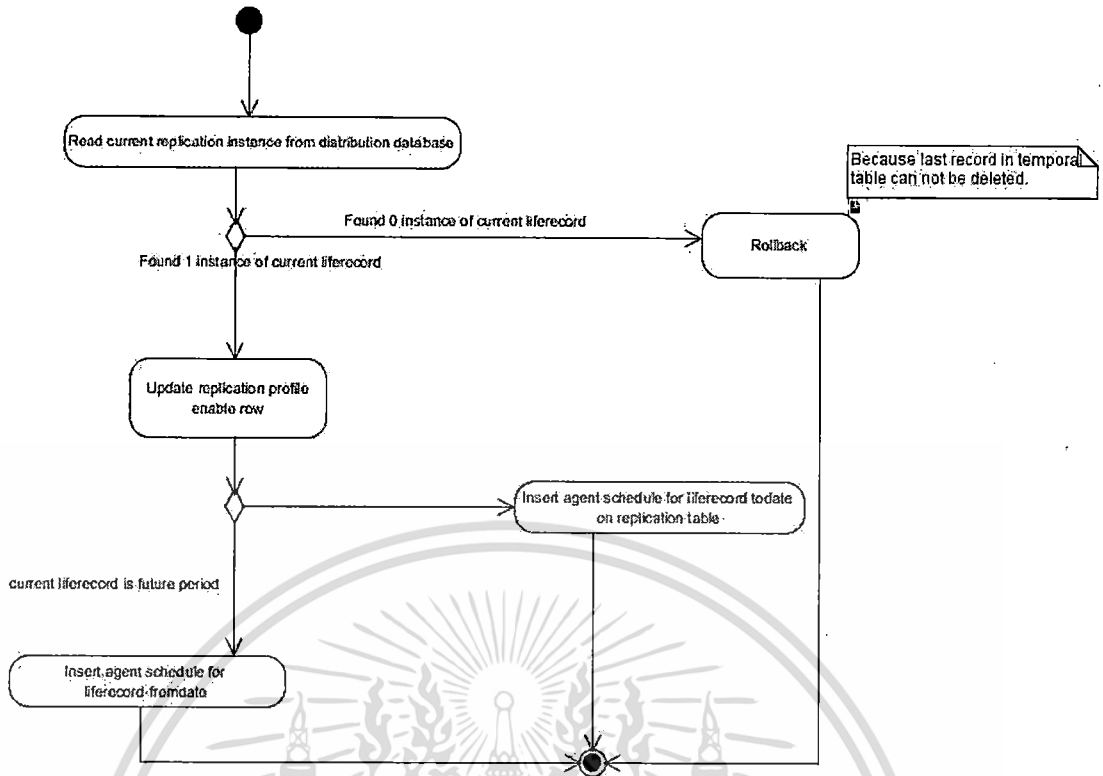
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 คำอธิบายยูสเคสบันทึกข้อมูลการใช้งานฐานข้อมูลที่สำคัญ

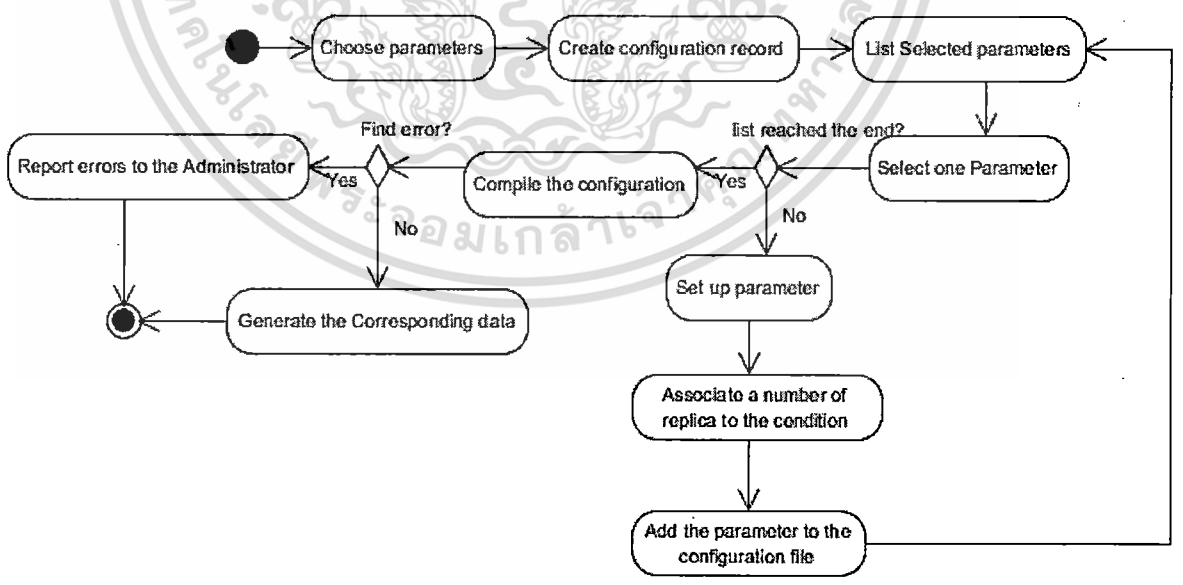
Use-Case Description	
Use Case Name:	Write log
Brief Description:	เป็นการบันทึกข้อมูลเริ่มต้นและสิ้นสุดกระบวนการใช้งานฐานข้อมูลที่สำคัญ
Primary actor:	-
Stakeholders:	-
Precondition:	-
Postcondition:	-
Trigger:	ข้อมูลการใช้งานฐานข้อมูลที่สำคัญ
Actor Action	System Action
	Step 1: ระบบตรวจสอบและปรับปรุงรูปแบบข้อมูลที่ได้รับ Step 2: บันทึกข้อมูลกิจกรรมการใช้งานฐานข้อมูลที่สำคัญลงฐานข้อมูลของระบบ
Alternate Cause of Event:	-

#### 4.3.2 แอคทิวิตีไดอะแกรม

แอคทิวิตีไดอะแกรม เป็นแผนภาพที่แสดงพฤติกรรมหรือลำดับขั้นตอนของกิจกรรมต่างๆ ที่เกิดขึ้นในผังงาน เพื่อเป็นการแสดงขั้นตอนการทำงานของระบบ ซึ่งแอคทิวิตีไดอะแกรมแสดงภาพรวมของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลสามารถแสดงได้ดังรูปที่ 4.1

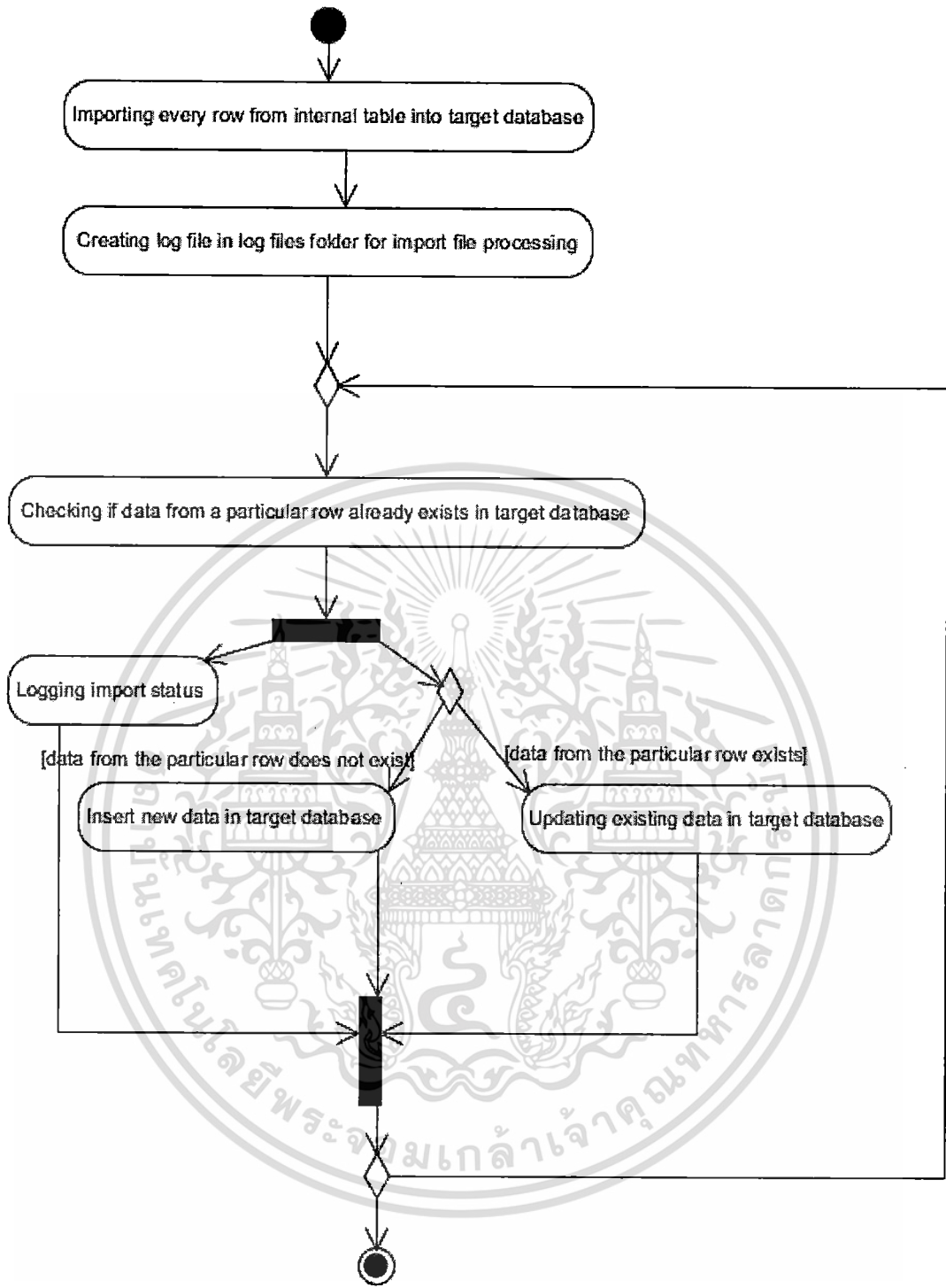


รูปที่ 4.5 แอคทิวิตีไดอะแกรมของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล



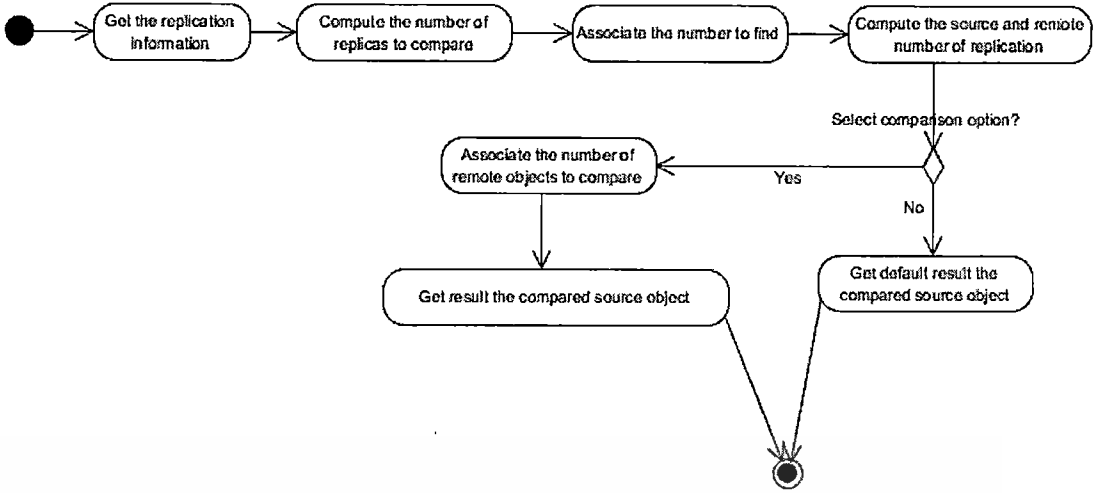
รูปที่ 4.6 แอคทิวิตีไดอะแกรมของการจัดการรายละเอียดการสำเนาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

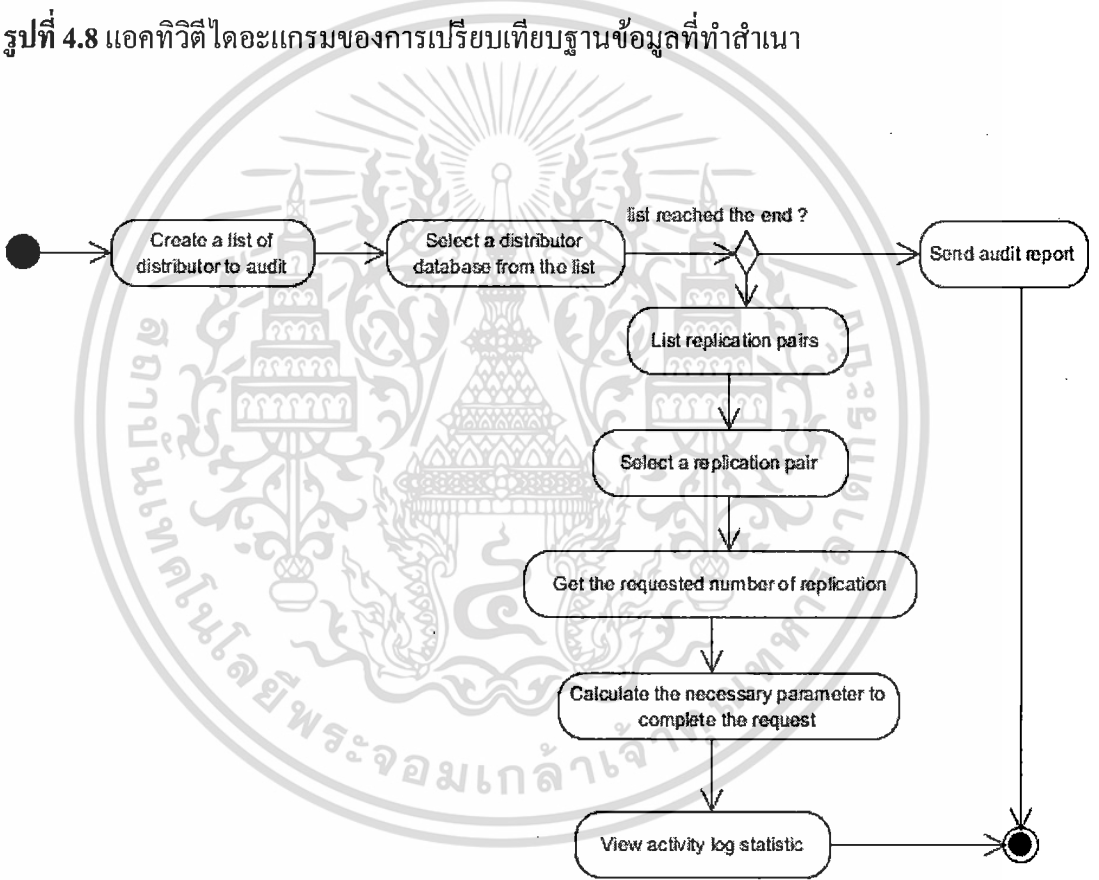


รูปที่ 4.7 แอคทิวิตีไดอะแกรมของการนำเข้าข้อมูลการใช้งานฐานข้อมูลที่ทำสำเนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

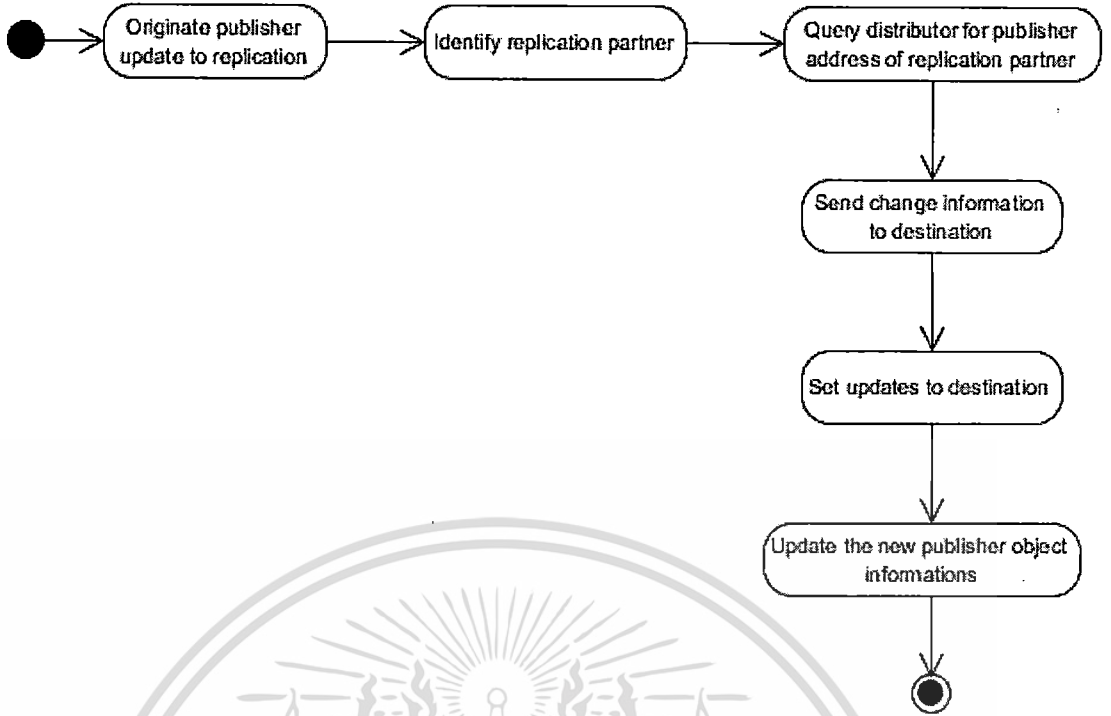


รูปที่ 4.8 แอคทิวิตีไดอะแกรมของการเปรียบเทียบฐานข้อมูลที่ทำสำเนา



รูปที่ 4.9 แอคทิวิตีไดอะแกรมของการดูสถิติของกิจกรรมการใช้งาน

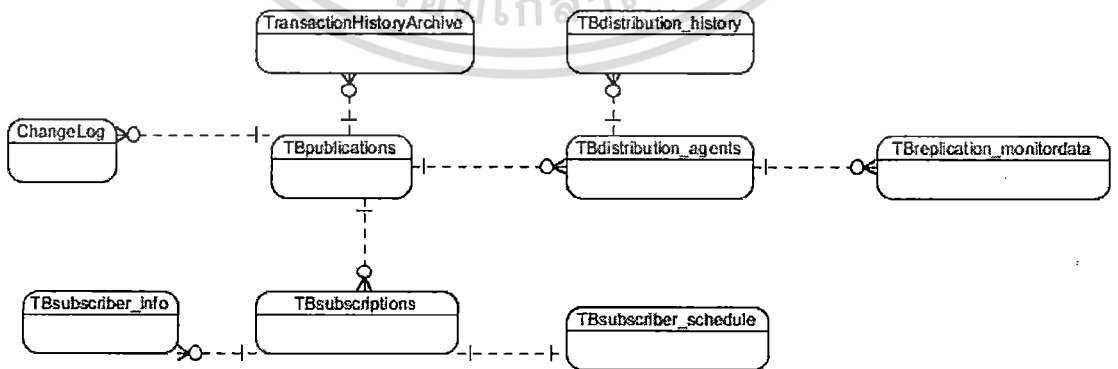
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 แอคทีวิตีไดอะแกรมของการนำเข้าข้อมูลการใช้งานฐานข้อมูลที่สำคัญ

### 4.3.3 อีอาร์ไดอะแกรม

จากความสัมพันธ์ทั้งหมด สามารถแปลงค่าเป็นส่วนของ Entity Type โดยมีส่วนประกอบที่สำคัญที่เป็น Attribute ต่างๆ ดังนี้



รูปที่ 4.11 อีอาร์ไดอะแกรมของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลาเพื่อสนับสนุนระบบการสำเนาฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีอาร์ไออะแกรมของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลสามารถอธิบายรายละเอียดต่างๆ ได้ดังนี้

1. ตาราง CHANGELOG เป็นตารางเก็บข้อมูลการเปลี่ยนแปลง โครงสร้างอ็อบเจ็กต์ที่ถูกระบุในการทำสำเนา
2. ตาราง TBDISTRIBUTION\_AGENTS เป็นตารางเก็บข้อมูลฐานข้อมูลตัวกลาง (DISTRIBUTOR)
3. ตาราง TBDISTRIBUTION\_HISTORY เป็นตารางเก็บข้อมูลการเปลี่ยนแปลงข้อมูลตัวกลาง
4. ตาราง TBPUBLICATION\_INFO เป็นตารางเก็บข้อมูลหน่วยของข้อมูล (ARTICLE) ของฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล
5. ตาราง TBPUBLICATIONS เป็นตารางเก็บข้อมูลฐานข้อมูลที่ใช้ในการสำเนาฐานข้อมูล (PUBLISHER)
6. ตาราง TBREPLICATION\_MONITORDATA เป็นตารางเก็บข้อมูลการเปรียบเทียบระหว่างฐานข้อมูลที่ใช้ในการสำเนา กับฐานข้อมูลที่รับการสำเนา
7. ตาราง TBSUBSCRIBER\_INFO เป็นตารางเก็บข้อมูลหน่วยของข้อมูล (ARTICLE) ของฐานข้อมูลที่รับการสำเนา
8. ตาราง TBSUBSCRIBER\_SCHEDULE เป็นตารางเก็บข้อมูลกำหนดการตรวจสอบและใช้งานฐานข้อมูลที่รับการสำเนา
9. ตาราง TBSUBSCRIPTIONS เป็นตารางเก็บข้อมูลฐานข้อมูลฐานข้อมูลที่รับการสำเนาฐานข้อมูล (SUBSCRIBER)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.3.4 พจนานุกรมข้อมูล

ฐานข้อมูลของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูล สามารถอธิบายรายละเอียดด้วยพจนานุกรมข้อมูลได้ดังนี้

ตารางที่ 4.7 รายละเอียดของฐานข้อมูลของตาราง CHANGELOG

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
LOGID	หมายเลขล็อก	Integer	PK	
EVENTTYPE	ประเภทล็อก	Varchar(256)		
OBJECTNAME	ชื่ออ็อบเจกต์	Varchar(256)		
OBJECTTYPE	ชื่ออ็อบเจกต์	Varchar(256)		
SQLCOMMAND	คำสั่งเอสคิวแอล	Varchar(256)		
LOGINNAME	ชื่อล็อกอิน	Varchar(256)		
DATABASEID	หมายเลขฐานข้อมูล	Integer	FK	TBPUBLICATIO NS
TTSLOG	วัน-เวลาที่เริ่มมีการแก้ไข	Date		
TTELOG	วัน-เวลาที่จบแก้ไข	Date		

ตารางที่ 4.8 รายละเอียดของฐานข้อมูลของตาราง TBDISTRIBUTION\_AGENTS

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
ID	หมายเลขตัวกลาง	Integer	PK	
SERVERNAME	ฐานข้อมูลตัวกลาง	Varchar(256)		
PUBLISHER_DATABASE_ID	หมายเลขฐานข้อมูลสำเนา	Integer	FK	TBPUBLICATIO NS
SUBSCRIBER_ID	หมายเลขฐานข้อมูลรับสำเนา	Integer	FK	TBSUBSCRIPTIO NS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.9 รายละเอียดของฐานข้อมูลของตาราง TBDISTRIBUTION\_HISTORY

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
AGENT_ID	หมายเลข ตัวกลาง	INT	PK,FK	TBDISTRIBUTION_AGE NTS
RUNSTATUS	ประเภทล๊อค	INT		
DURATION	คำสั่งเอสคิว แอล	INT		
CURRENT_DELIVERY_ RATE	อัตราจัดส่ง ปัจจุบัน	FLOAT		
CURRENT_DELIVERY_L ATENCY	เวลาการจัดส่ง ล่าช้าปัจจุบัน	INT		
AVERAGE_COMMANDS	ค่าเฉลี่ยคำสั่ง	INT		
DELIVERED_COMMAN DS_DATE	วัน-เวลาเก็บ ข้อมูล	DATE	PK	

ตารางที่ 4.10 รายละเอียดของฐานข้อมูลของตาราง TBPUBLICATION\_INFO

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
PUBLISHER	หมายเลขการ สำเนา	INTEGER	PK,FK	TBPUBLICATION
OBJECT_NAME	ชื่ออ็อบเจกต์	NVARCHAR(128)	PK	
OBJECT_TYPE	ชนิดอ็อบเจกต์	CHAR(2)		
ARTICLE	ชื่อข้อมูล หน่วยของ ข้อมูล	NVARCHAR(128)	PK	
TTARTICLE	วัน-เวลาที่มี การแก้ไข	DATE		

ตารางที่ 4.11 รายละเอียดของฐานข้อมูลของตาราง TBPUBLICATIONS

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
PUBLISHER_ID	หมายเลขฐานข้อมูล สำเนา	INTEGER	PK	
PUBLISHER_DB	ชื่อฐานข้อมูลสำเนา	NVARCHAR(256)		
INSTANCE	ชื่ออินสแตนซ์	NVARCHAR(128)		
SERVERNAME	ชื่อเครื่องเซิร์ฟเวอร์	NVARCHAR(128)		

ตารางที่ 4.12 รายละเอียดของฐานข้อมูลของตาราง TBREPLICATION\_MONITORDATA

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
PUBLICATION_ID	หมายเลขการ สำเนา	INTEGER	PK,FK	TBPUBLICATIONS
AGENT_ID	หมายเลข ตัวกลาง	INTEGER	PK,FK	TBDISTRIBUTION_AGEN TS
WARNING	ความถี่	INTEGER		
LAST_DISTSYNC	วัน-เวลาการ ตรวจสอบ	DATETIME		
AGENTSTOPTIME	วัน-เวลา ยกเลิกการ ตรวจสอบ	DATETIME		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.13 รายละเอียดของฐานข้อมูลของตาราง TBSUBSCRIBER\_INFO

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
SUBSCRIBER	หมายเลขการรับ สำเนา	INTEGER	PK,FK	TBSUBSCRIPTIONS
OBJECT_NAME	ชื่ออ็อบเจกต์	NVARCHAR(1 28)	PK	
OBJECT_TYPE	ชนิดอ็อบเจกต์	CHAR(2)		
ARTICLE	ชื่อข้อมูลหน่วย ของข้อมูล	NVARCHAR(1 28)	PK	
TTARTICLE	วัน-เวลาที่มีการ แก้ไข	DATE		

ตารางที่ 4.14 รายละเอียดของฐานข้อมูลของตาราง TBSUBSCRIBER\_SCHEDULE

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
PUBLISHER	หมายเลขการ ทำสำเนา	INTEGER	PK,FK	TBPUBLICATIONS
SUBSCRIBER	หมายเลขการ รับสำเนา	INTEGER	PK,FK	TBSUBSCRIPTIONS
FREQUENCY_TYPE	ประเภทรอบ	Integer		
FREQUENCY_INTERVAL	จำนวนรอบ	Integer		
TSACTIVE_START_DATE	วัน-เวลาเริ่ม ทำงาน	DATE		
TSACTIVE_END_DATE	วัน-เวลาที่หยุด ทำงาน	DATE		

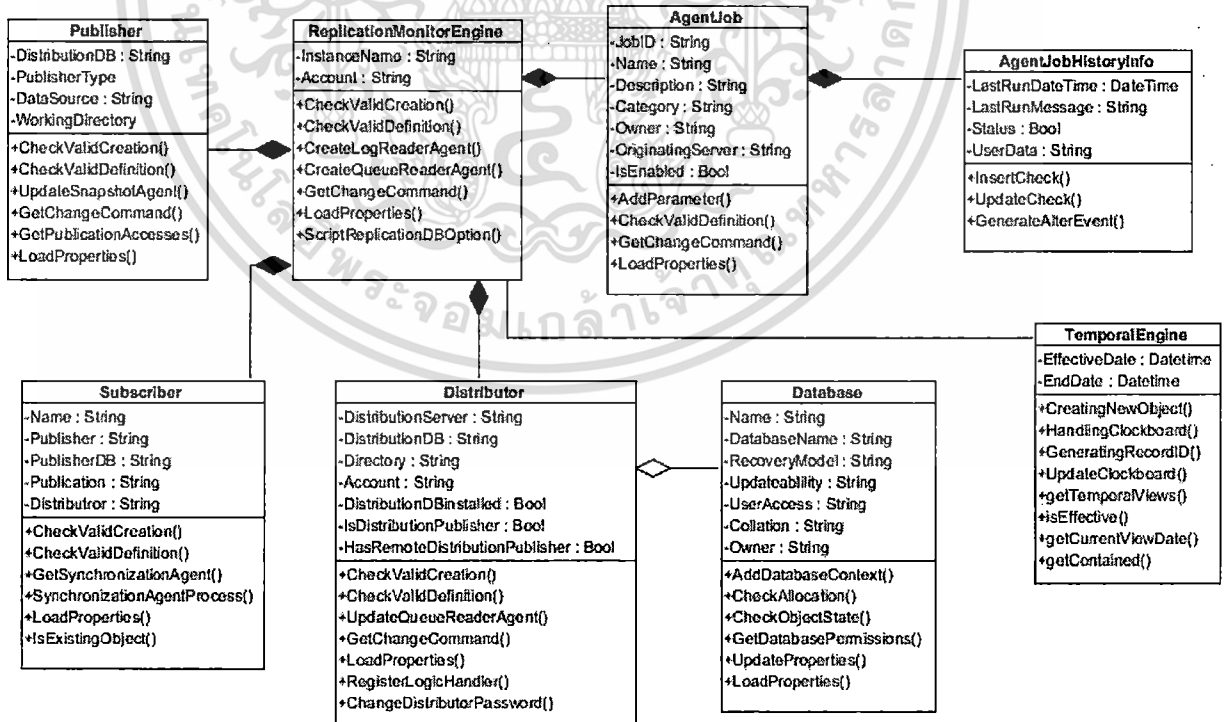
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.15 รายละเอียดของฐานข้อมูลของตาราง TBSUBSCRIPTIONS

แอตทริบิวต์	รายละเอียด	ประเภท	ชนิดคีย์	ตารางที่สัมพันธ์
SUBSCRIBER_ID	หมายเลขฐานข้อมูล	INTEGER	PK	
SUBSCRIBER_DB	ชื่อฐานข้อมูลรับสำเนา	NVARCHAR(256)		
INSTANCE	ชื่ออินสแตนส์	NVARCHAR(128)		
SERVERNAME	ชื่อเครื่องเซิร์ฟเวอร์	NVARCHAR(128)		

### 4.3.5 คลาสไดอะแกรม

การออกแบบระบบได้ทำการปรับปรุงคลาสไดอะแกรม ในขั้นตอนของการวิเคราะห์ระบบ ให้มีความสมบูรณ์มากขึ้น โดยเพิ่มเติมคุณสมบัติและพฤติกรรมของคลาส ซึ่งแสดงด้วยแผนภาพดังต่อไปนี้



รูปที่ 4.10 คลาสไดอะแกรมของการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุน

ระบบการสำเนาฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# สรุปผลการพัฒนาและข้อเสนอแนะ

การพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบการสำเนาฐานข้อมูลเป็นระบบที่ช่วยให้ผู้ดูแลระบบสามารถประยุกต์ใช้กับข้อมูลเชิงเวลา รวมทั้งได้สร้างส่วนที่ใช้จัดการตรวจสอบสถานะการทำงานของการทำงานของการสำเนาฐานข้อมูล ดังสรุปต่อไปนี้

1. สรุปผล
2. ปัญหาและอุปสรรค
3. ข้อเสนอแนะในการพัฒนาระบบต่อ

### 5.1 สรุปผล

โปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลาจะช่วยผู้ใช้ในการเรียกใช้งานและสนับสนุนระบบการสำเนาฐานข้อมูลได้สะดวกขึ้น โดยที่ผู้ใช้สามารถกำหนดคุณสมบัติต่างๆ ที่มีมาให้กำหนดการเชื่อมต่อกับเครื่องให้บริการอื่น ซึ่งหลักการของการเขียนโปรแกรมแบบวิซวล ทำให้ผู้ใช้ไม่จำเป็นต้องมีความรู้ด้านการเขียน โปรแกรมก็สามารถใช้โปรแกรมได้ ผู้ดูแลฐานข้อมูลจะได้รับความสะดวกสบายด้วยเพราะไม่จำเป็นต้องเขียนคำสั่งขึ้นมาตรวจสอบวิเคราะห์หรือเรียกใช้งานการสำเนาฐานข้อมูลก็สามารถทำการตรวจสอบคุณสมบัติหรือความถูกต้องของข้อมูลและความถูกต้องได้ซึ่งทำให้การพัฒนาหรือเรียกใช้เป็นไปได้อย่างรวดเร็วแล้วสะดวกสบายมากยิ่งขึ้น (การพัฒนาโครงการนี้จะประสบผลสำเร็จหากมีการนำไปใช้งานได้จริงและตัวโปรแกรมสามารถทำงานได้ตามคุณสมบัติที่ได้ออกแบบไว้แล้วในข้างต้น ซึ่งในขณะนี้กำลังนำไปทดลองใช้งาน)

การไหลคข้อมูลแต่ผู้พัฒนา ก็ได้ศึกษาข้อมูลต่างๆ เพิ่มเติม และคิดหาวิธีการต่างๆ ในการแก้ไขปัญหา จนข้อมูลระบบการจัดการบริหารเครื่องคอมพิวเตอร์นี้ เสร็จลุล่วงมาได้ โดยสามารถสร้างความพึงพอใจต่อผู้ใช้งานได้เป็นอย่างดี และได้รับข้อมูลที่ต้องการรวดเร็วมากขึ้น

### 5.2 ปัญหาและอุปสรรค

เนื่องจากข้อมูลที่ต้องการแปลงมีความหลากหลายมาก จึงมีปัญหและแนวทางแก้ไขระหว่างการพัฒนา โดยเฉพาะ ปัญหาของระบบฐานข้อมูลที่มีความซับซ้อนมาก ทั้งยังมีปัญหาในเรื่องของข้อมูลที่มีปริมาณมาก และปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เว็บเซอร์วิสเป็นเทคโนโลยีใหม่ คู่มือในการพัฒนาหรืออุปกรณ์ยังไม่แพร่หลายทำให้ การศึกษาค้นคว้าหาความรู้ยังไม่เพียงพอกับความจำเป็นที่ต้องใช้ในการพัฒนาโปรแกรม การแก้ไข เมื่อไม่เข้าใจหรือสงสัยจะสอบถามผู้รู้หรือค้นคว้าเพิ่มเติมในเว็บไซต์

2. การพัฒนาโปรแกรมแบบวิซวล (Visual Programming) นั้นเป็นการออกแบบหรือพัฒนา เพื่อให้ผู้ใช้สามารถใช้งานโปรแกรมได้ง่าย ซึ่งต้องอาศัยประสบการณ์และความรู้เพื่อที่จะออกแบบ ให้ผู้ใช้สามารถใช้งานได้ง่าย ซึ่งผู้พัฒนาเองยังมีประสบการณ์น้อยและความรู้ยังไม่เพียงพอทำให้ การพัฒนาหรือออกแบบทำให้ยังใช้งานไม่ง่ายเท่าที่ควร

การแก้ไข เมื่อจะทำการออกแบบโปรแกรมแบบวิซวล (Visual Programming) จะสอบถาม ผู้ใช้ว่าโปรแกรมแบบไหนที่ชอบ และใช้งานได้ง่ายรวมไปถึงการศึกษาหาความรู้เพื่อออกแบบ โปรแกรมให้สามารถใช้งานได้ง่าย

3.การพัฒนาเพื่อให้สามารถทำการเรียกข้อมูลที่ซับซ้อน (Complex Type) นั้นทำได้ยาก เนื่องจากเครื่องมือหรือ อุปกรณ์ที่ใช้ในการพัฒนายังไม่สามารถสนับสนุนการใช้ข้อมูลที่ซับซ้อน (Complex Type) มากนักซึ่งทำให้การพัฒนาเป็นไปด้วยความยากลำบาก ทำให้การพัฒนาเพื่อ รองรับข้อมูลที่ซับซ้อน (Complex Type) รวมไปถึงการแสดงผลเป็นไปด้วยความยากลำบาก

การแก้ไข การแก้ปัญหาข้อมูลที่ซับซ้อน (Complex Type) จะทำการสร้างเมทอดหรือ คลาสที่เป็นคลาสมกลางเพื่อเรียกใช้ข้อมูลที่ซับซ้อน

### 5.3 ข้อเสนอแนะในการพัฒนาระบบต่อ

จากการศึกษาการพัฒนาโปรแกรมประยุกต์บนฐานข้อมูลเชิงเวลา เพื่อสนับสนุนระบบ การสำเนาฐานข้อมูล มีข้อเสนอแนะสำหรับผู้ที่ต้องการศึกษาโครงการในลักษณะเดียวกัน ดังต่อไปนี้

1. ระบบควรมีการเพิ่มการเชื่อมต่อและการทำสำเนาฐานข้อมูล ร่วมกับระบบการจัดการ ฐานข้อมูลเชิงสัมพันธ์อื่น เช่น โปรแกรมจัดการฐานข้อมูลออรากเคิล เพื่อที่จะสามารถอำนวยความสะดวกในการใช้งานข้อมูลให้ครอบคลุม

2. ควรปรับปรุงกระบวนการตรวจสอบความถูกต้องขององค์ประกอบของวัตถุที่ถูกทำ สำเนาบนฐานข้อมูล ควบคู่กับการเฝ้าดูและการแจ้งเตือนงานอัตโนมัติ (Jobs) และตัวแจ้งเตือน (Alerts) ผู้ดูแลฐานข้อมูล ซึ่งมีข้อความเตือนว่าองค์ประกอบการทำสำเนาฐานข้อมูลมีปัญหา

## บรรณานุกรม

- ธาริน สิทธิธรรมชารี. 2554. **บริหารและจัดการฐานข้อมูลระดับมืออาชีพ SQL Server 2008 R2.** กรุงเทพฯ: ซิมพลิฟาย.
- ผุสดี พรพล. 2546. "การออกแบบและพัฒนาฐานข้อมูลเชิงเวลาด้วยวิธีโนแอม." **วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.**
- พงษ์พันธ์ ศิวาลัย. 2552. **SQL server 2008 ฉบับสมบูรณ์.** กรุงเทพฯ: ซีเอ็ดยูเคชั่น.
- ศิริลักษณ์ โรจนกิจอำนวย. 2552. **การออกแบบและบริหารฐานข้อมูล.** กรุงเทพฯ: ดวงกลมสมัย.
- สุนทริน วงศ์ศิริกุล และชัยวัฒน์ สิทธิกร โอพารกุล. 2550. **การพัฒนาโมเดลสำหรับการเขียนโปรแกรมเชิงวัตถุด้วย UML 2.0 = Unified Modeling Language.** กรุงเทพฯ: ชัสเชสมิเดีย.
- Azhar P. Taj. 2012. "Achieving Excellence in Designing and Maintaining SQL Server Transactional Replication Environments" **Technical White Paper.** 26-33.
- Christian S. Jensen and Richard T. Snodgrass. 1999. "Temporal Data Management" **IEEE Transactions on Knowledge and Data Engineering.** 36-44.
- Fuller A. 2007. **Database Design: A Point in Time Architecture.** [Online] Available: <http://www.simple-talk.com/sql/database-administration/database-design-a-point-in-time-architecture/>.
- Gregersen, H. Jensen, C. S. and Mark, L. 2000. "From Temporal ER Models to Relations." **Temporal Database Management.** 1065-1077.

## ภาคผนวก ก

# การติดตั้งเครื่องมือในการพัฒนา

สำหรับการติดตั้งเว็บเซอร์วิสจำเป็นต้องติดตั้งลงในเครื่องเซิร์ฟเวอร์การติดตั้งโปรแกรมที่จำเป็นต่างๆในเครื่อง Server โดยส่วนสำคัญที่ต้องติดตั้ง คือ

- Microsoft .NET Framework 3.5 Service Pack1
- Microsoft SQL Server 2008
- Microsoft SQL server 2008 management studio

### 1. การติดตั้ง .NET Framework 3.5 Service Pack1

.NET Framework 3.5 SP1 เป็นตัวสำหรับเตรียมการติดตั้ง SQL Server 2008 ซึ่งสำหรับ Windows Server 2008 R2 นี้วิธีการลง .NET Framework 3.5 SP1 จะแตกต่างจากเวอร์ชันเก่า โดยมีวิธีติดตั้ง ดังนี้

#### 1.2.1 การเปิดใช้งานเมื่อเคยติดตั้ง .NET Framework 3.5 SP1 แล้ว

- คลิกปุ่ม Start ที่มุมซ้ายของหน้าจอ
- เลือก Administrative Tools และเลือก Server Manager
- ในหน้า Server Manager ให้กดปุ่ม Features เพื่อดูโครงสร้างทั้งหมด (Features) ให้คลิก

เลือก .NET Framework 3.5 SP1

1.2.2 ถ้า .NET Framework 3.5 SP1 ไม่มีอยู่ในรายการที่เลือก หรือยังไม่เคยติดตั้งมาก่อน ให้ติดตั้งดังวิธีการดังนี้

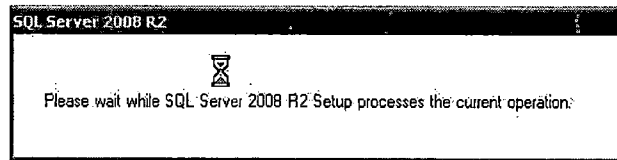
- ในหน้า Server Manager interface เลือก Add Features เพื่อดูรายการของ features ที่มี
- ในส่วน Select Features interface เลือก .NET Framework 3.5.1 Features.
- เมื่อเลือก .NET Framework 3.5.1 Features แล้วจะเห็น check box “.NET Framework 3.5.1”และ “WCF Activation” ให้เลือก .NET Framework 3.5.1 และคลิก Next.
- ในหน้า Confirm Installation Selections ให้คลิก Install
- รอจนกว่าจะติดตั้งเสร็จแล้วคลิก close

### 2. การติดตั้ง Microsoft SQL Server 2008

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นระบบการติดต่อกับฐานข้อมูล (Database Operating System หรือ Database engine)

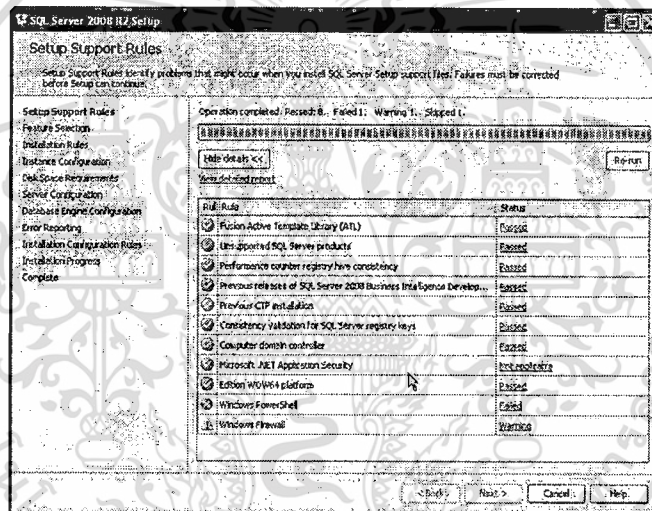
2.1 เริ่มติดตั้ง Microsoft SQL Server 2008 ดังรูปที่ ก.1



รูปที่ ก.1 การตรวจสอบความพร้อมของเครื่อง

2.2 โปรแกรมจะตรวจสอบว่าเครื่องไม่มีโปรแกรมใด ถ้าไม่ขาดแล้วจะดำเนินการต่อ ดังรูป

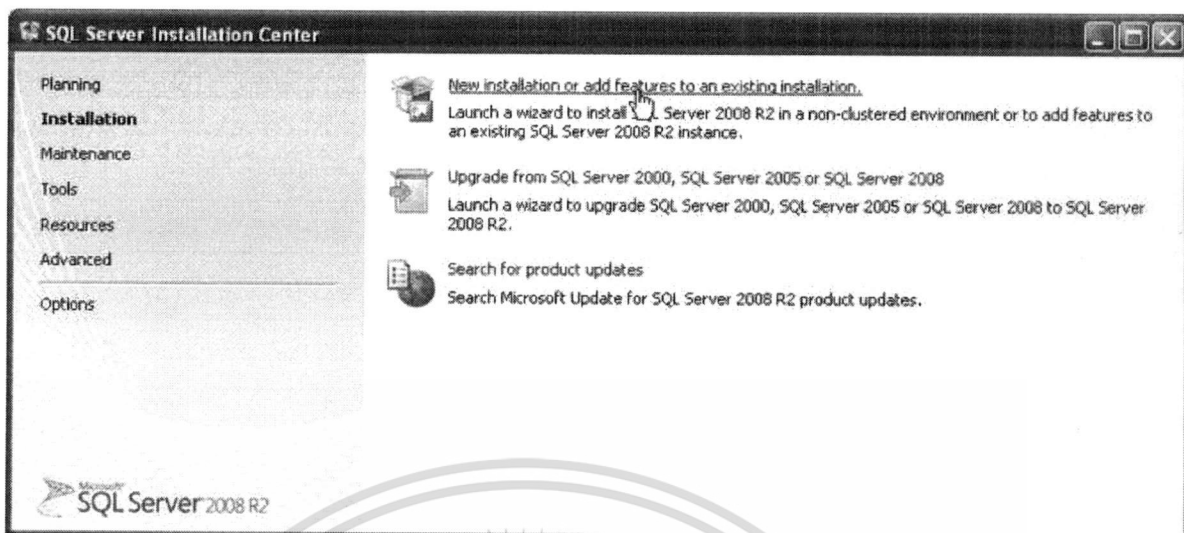
ที่ ก.7



รูปที่ ก.2 การตรวจสอบ Setup Support Rules

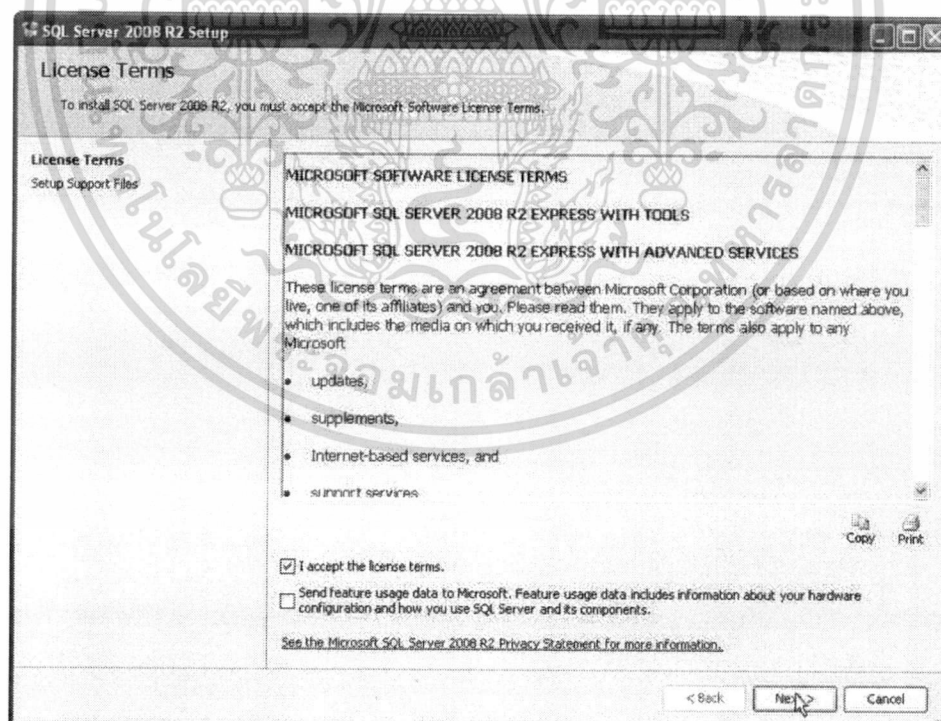
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 คลิกเลือกติดตั้งโปรแกรมดังรูปที่ ก.3



รูปที่ ก.3 การเลือกติดตั้งโปรแกรม

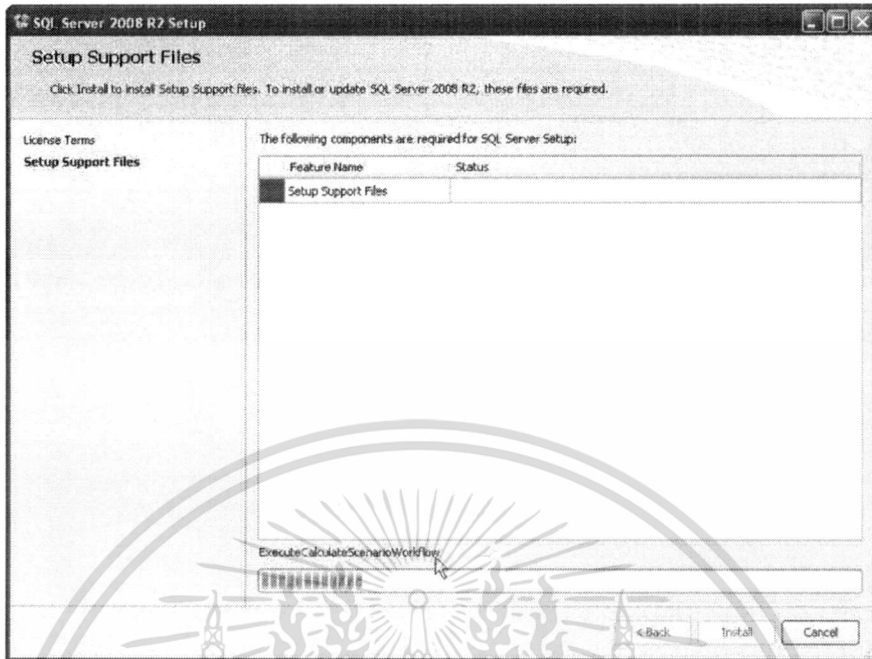
2.4 แสดงข้อตกลงของสิทธิ์ในการใช้โปรแกรม คลิก  I accept the license terms. เพื่อยอมรับ และคลิก Next ดังรูปที่ ก.4



รูปที่ ก.4 ข้อตกลงของสิทธิ์ในการใช้โปรแกรม

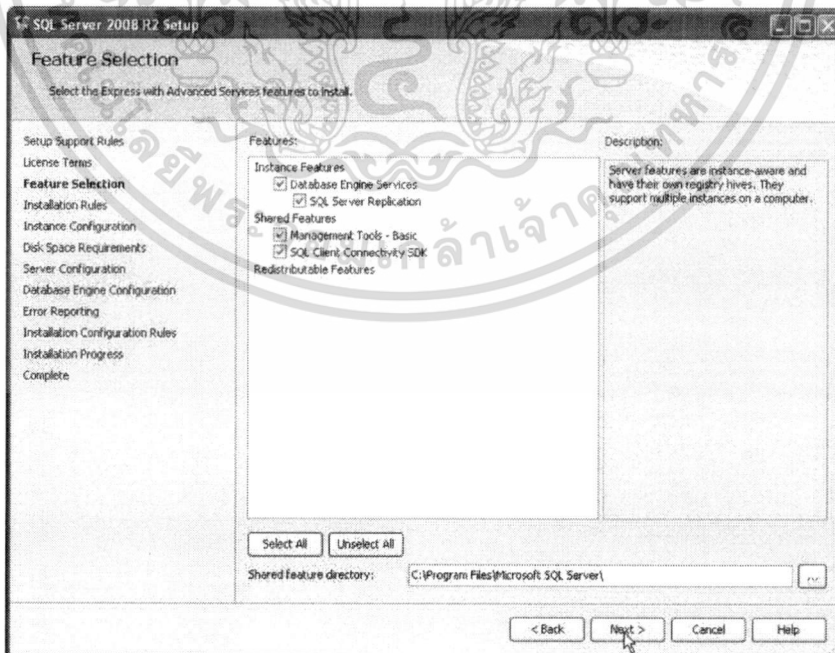
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 เลือก Install และรอกะหว่างการติดตั้งโปรแกรม อาจใช้เวลาสักครู่ดังรูปที่ ก.5



รูปที่ ก.5 การ Setup Support Files

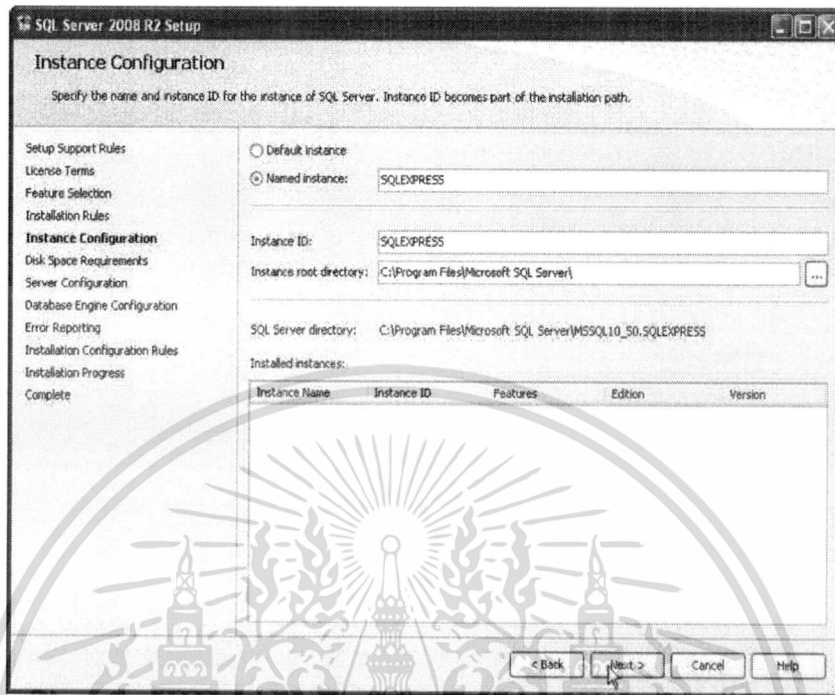
## 2.6 เลือก Features ที่ต้องการติดตั้ง และคลิก Next เพื่อดำเนินการต่อดังรูปที่ ก.6



รูปที่ ก.6 การเลือก Feature

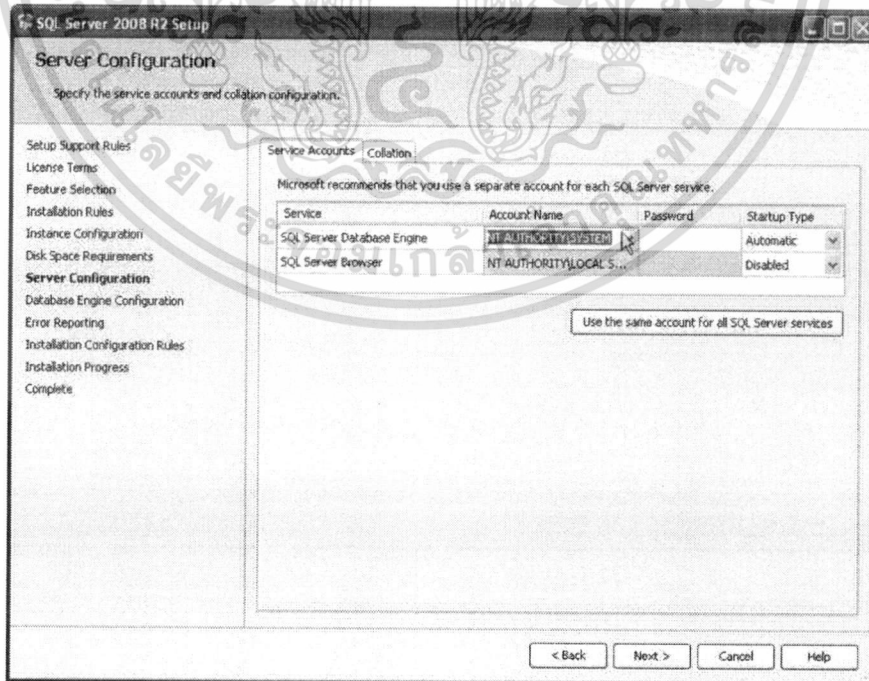
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ใส่ชื่อ Named instance, Instance ID และที่ตั้งของ Instance root directory แล้วคลิก Next ดังรูปที่ ก.7



รูปที่ ก.7 Instance Configuration

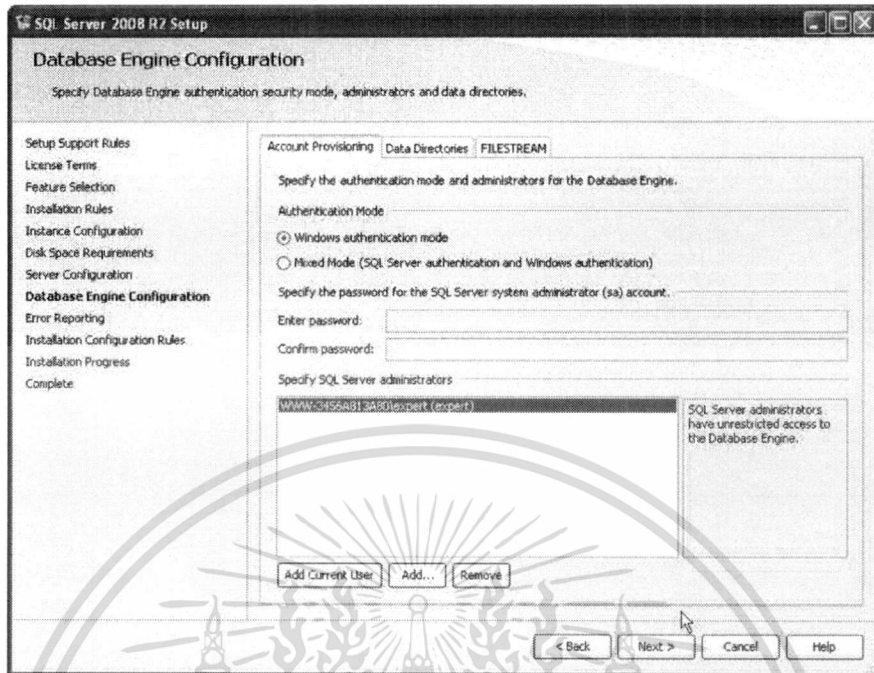
2.8 เลือกบัญชีผู้ใช้ แล้วคลิก Next เพื่อดำเนินการต่อ ดังรูปที่ ก.8



รูปที่ ก.8 Server Configuration และเลือก Account

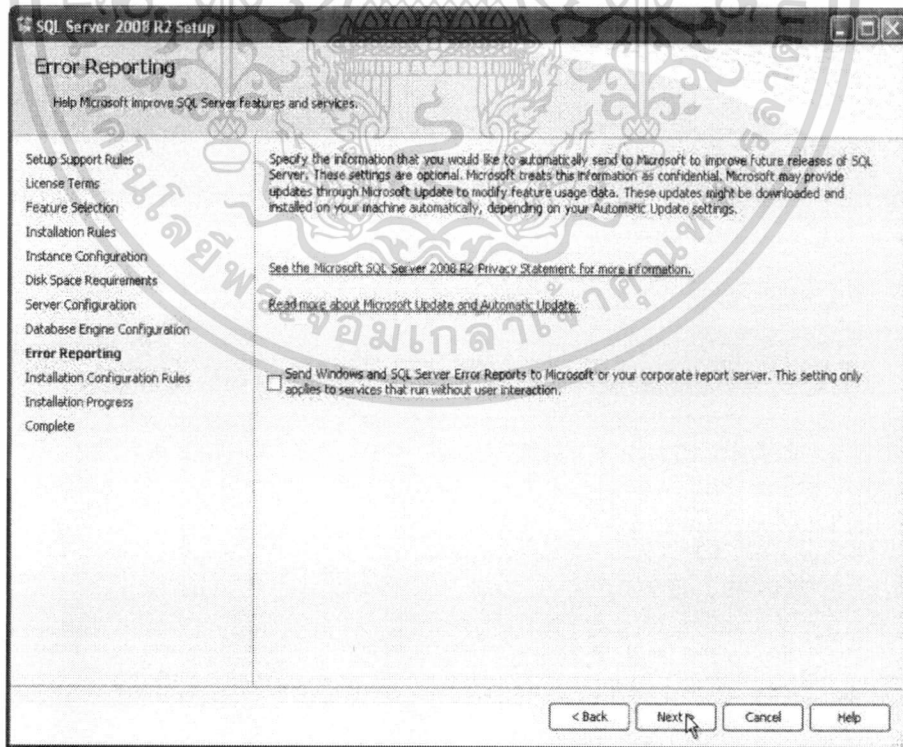
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.9 เพิ่มผู้ใช้งานโดยเลือกจากผู้ใช้งานของ Windows ดังรูปที่ ก.9



รูปที่ ก.9 Database Engine Configuration

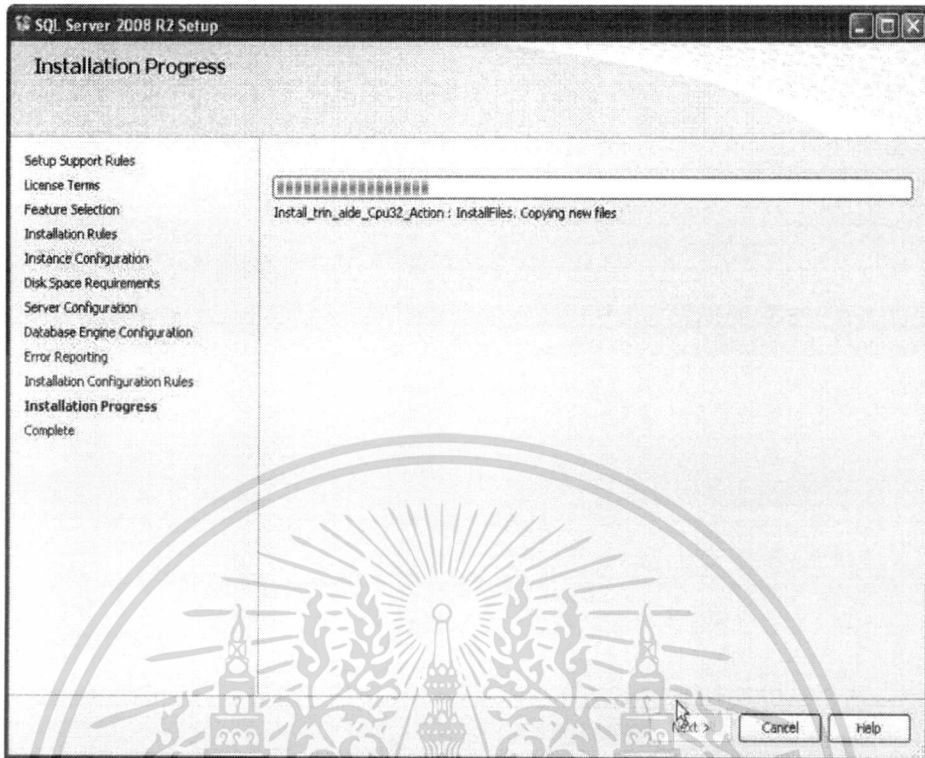
## 2.10 ขั้นตอนการกำหนดรายงานข้อผิดพลาดดังรูปที่ ก.10



รูปที่ ก.10 Error Reporting

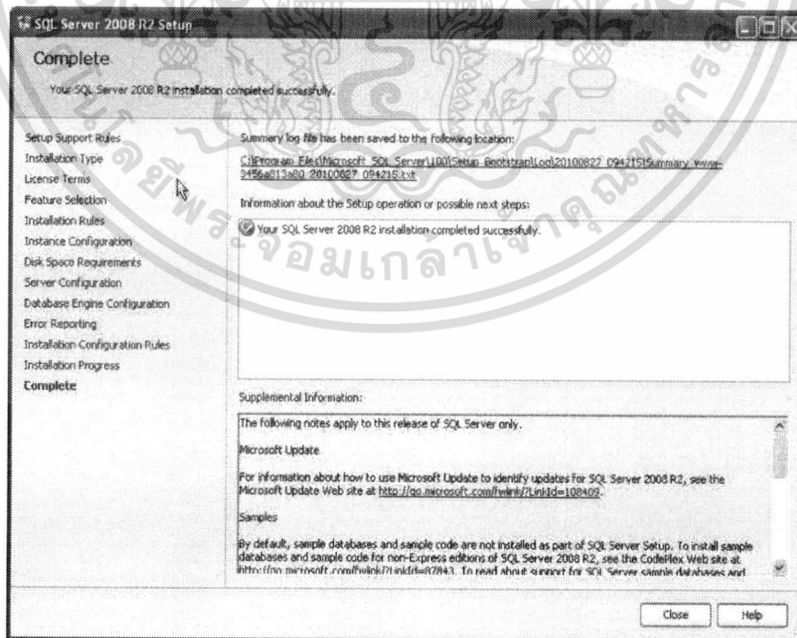
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.11 ขั้นตอนการติดตั้ง รอสักครู่ดังภาพที่ ก.11



รูปที่ ก.11 Installing Progress.

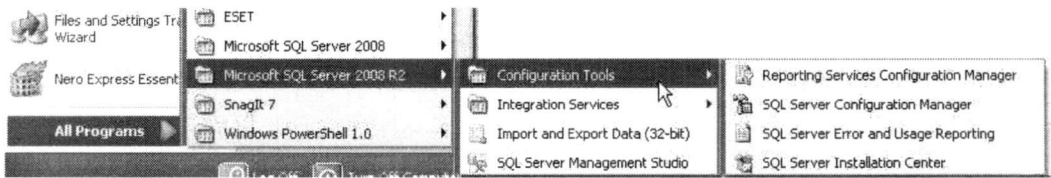
## 2.12 การติดตั้งเสร็จเรียบร้อยดังรูปที่ ก.12



รูปที่ ก.12 หน้าการติดตั้งเสร็จเรียบร้อย

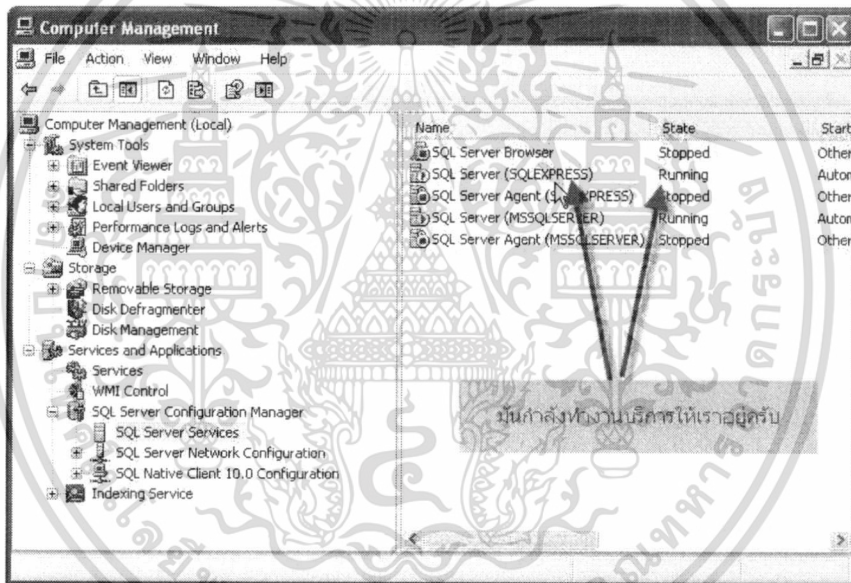
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 เมื่อโปรแกรมติดตั้งเสร็จเรียบร้อยแล้วจะปรากฏไอคอนเข้าโปรแกรมที่เมนู All Programs ดังรูปที่ ก.13



รูปที่ ก.13 Short cut เข้าสู่โปรแกรม Microsoft SQL Server 2008

2.14 สำหรับการตรวจสอบการทำงานของ SQL Service นั้นให้คลิกขวาที่ My Computer แล้วเลือก Manage จากนั้นดูที่เมนู SQL Server Services จะเห็นสถานะ SQL Server ดังรูปที่ ก.14



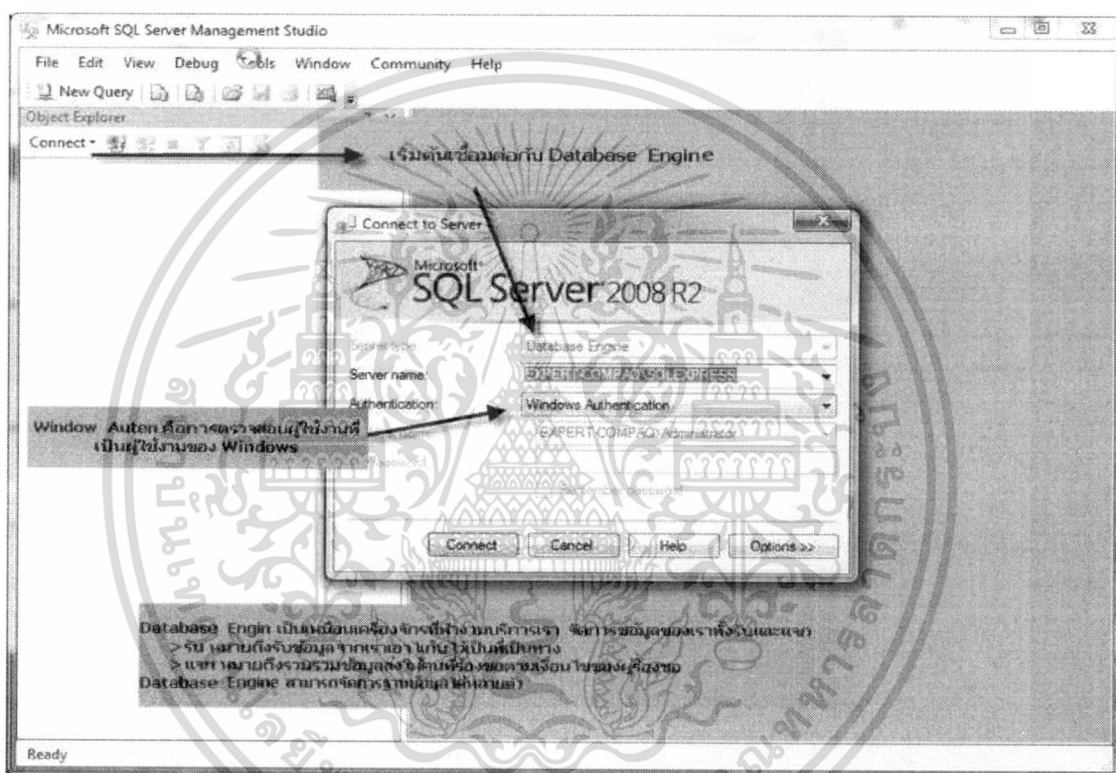
รูปที่ ก.14 การเข้าเมนู Manage เพื่อตรวจสอบการทำงานของ SQL Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. การติดตั้ง SQL server 2008 management studio

SQL Server Management Studio เป็นตัวช่วยจัดการฐานข้อมูล เช่น สร้าง, เพิ่มฐานข้อมูลใหม่ ลบ Query, Backup, Restore และ Attach ฐานข้อมูลที่มีอยู่แล้ว แต่ทั้งนี้ทั้งนั้น ฐานข้อมูลที่สร้างจาก SQL express เวอร์ชันอื่นที่สูงกว่าไม่สามารถนำมา Attach หรือ Restore ได้และฐานข้อมูลที่สร้างจากเวอร์ชันนี้ก็ไม่สามารถใช้กับ SQL เวอร์ชันที่ต่ำกว่าได้

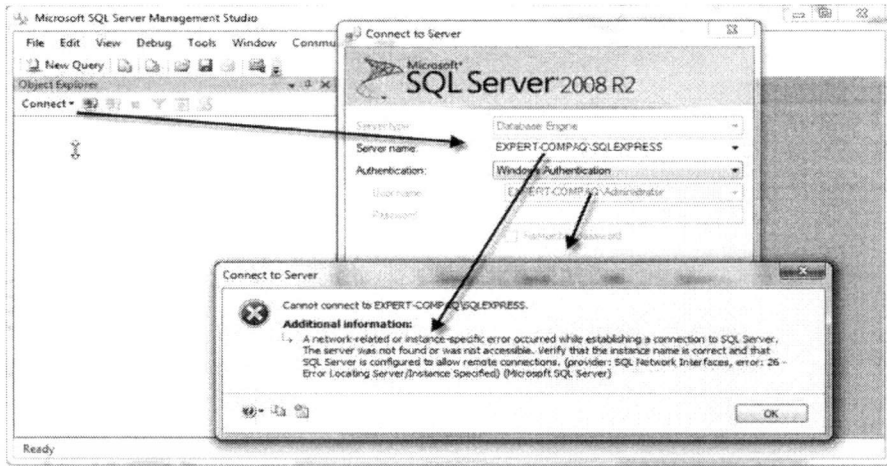
#### 3.1 เริ่มต้นเชื่อมต่อกับฐานข้อมูล โดยคลิก Connect ดังรูปที่ ก.15



รูปที่ ก.15 การเริ่มต้นเชื่อมต่อกับฐานข้อมูล

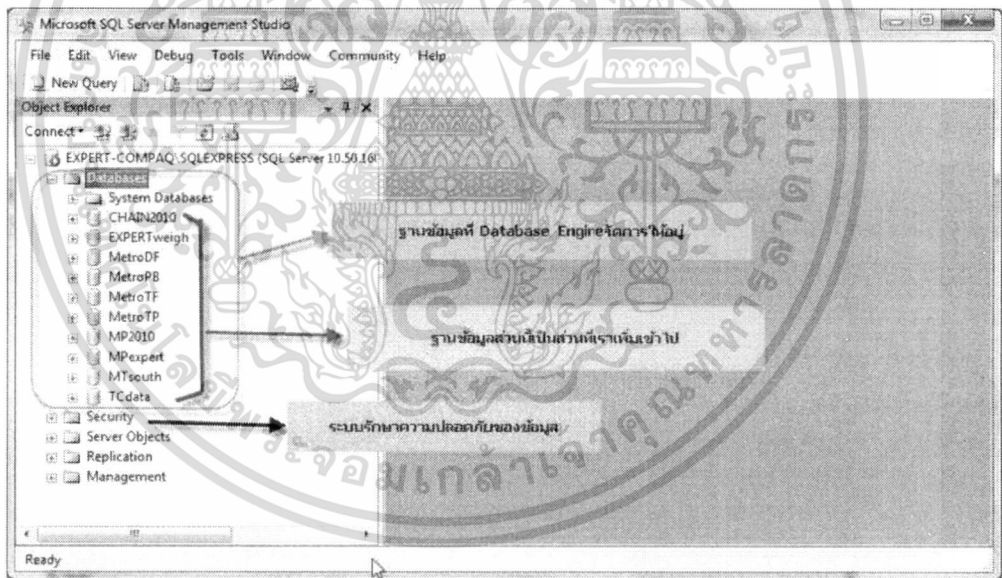
3.2 ถ้าเกิดข้อความแจ้งเตือนดังภาพนี้แสดงว่าเข้าจัดการไม่ได้เนื่องจาก SQL Database Engine ยังไม่ได้เปิดบริการ หรือชื่อ Server ไม่ถูกต้อง ดังรูปที่ ก.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.16 ตัวอย่างข้อความแจ้งเตือน

3.3 วิธีตรวจสอบว่า SQL Database Engine เครื่องติดอยู่หรือไม่ สังเกตได้ว่าถ้า SQL server Database Engine ทำงานอยู่ SQL Management Studio จึงจะเข้าไปจัดการได้ ดังรูปที่ ก.17



รูปที่ ก.17 การเข้าจัดการฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้เขียน

ชื่อผู้เขียน	นางสาวรัตนภรณ์ นพรัตน์
วันเดือนปีเกิด	21 มกราคม 2525
สถานที่เกิด	จ. สระบุรี
วุฒิการศึกษา	วิทยาศาสตรบัณฑิต สาขาระบบสารสนเทศคอมพิวเตอร์
สถานที่สำเร็จการศึกษา	มหาวิทยาลัยบูรพา
ปีที่สำเร็จการศึกษา	2547
ปัจจุบันทำงานที่	ธนาคารซูมิโตโม มิตซูบิ แวงกิ่ง คอร์ปอเรชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้