

การพัฒนาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอด้วย ART  
An Analysis of the VDO Rental Process using ART

โดย

นาย พิชัย กริชกมลกุล

รหัส 43067163



\*H001939\*

อาจารย์ที่ปรึกษา

ผศ. ดร. อาริต ธรรมโน

วัน เดือน ปี.....	23	พ.ค.	2550
เลขทะเบียน.....	01939		
เลขเรียกหนังสือ.....	วิท.	ท/642 ก	2545
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."			

รายงานเป็นส่วนหนึ่งของวิชา โครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2545

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

# ใบรับรองโครงการพัฒนาระบบงาน (System Development Project)

เรื่อง

การพัฒนาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอด้วย ART

An Analysis of the VDO Rental Process using ART

นายพิชัย กริชกมลกุล

รหัส 43067163

รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของ  
การศึกษาวិชาโครงการพัฒนาระบบงาน หลักสูตรวิทยาศาสตรมหาบัณฑิต (เทคโนโลยีสารสนเทศ)  
ภาคเรียนที่ 2 ปีการศึกษา 2545



..... อาจารย์ที่ปรึกษา

(ผศ. ดร. อาริต ธรรมโน)



..... กรรมการสอบ

(ผศ. ดร. วรพจน์ กรีสู่ระเดช)



..... กรรมการสอบ

(ดร. ธนารัตน์ ชลิตาพงศ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอด้วย ART
นักศึกษา	นายพิชัย กริชกมลกุล
อาจารย์ที่ปรึกษา	ผศ. ดร. อาริต ธรรมโน
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

## บทคัดย่อ

โครงการนี้นำเสนอ การพัฒนาบระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอ ด้วยเทคโนโลยี Neural Network โดยนำเอา ART (Adaptive Resonance Theory) มาใช้ในการจำแนกกลุ่มลูกค้า ในการพัฒนาบระบบจะมีการนำเอาภาษา Java มาประยุกต์ใช้กับฐานข้อมูลที่เป็น MS SQL 7.0 และในการพัฒนาโครงการทั้งหมด ได้นำข้อมูลจริงจากร้านเช่าวิดีโอที่มีอยู่ในปัจจุบัน มาพัฒนาให้ใช้งานได้จริง สะดวก รวดเร็ว แม่นยำ และมีประสิทธิภาพ เพื่อใช้ทำ Data Mining กับข้อมูลต่างๆที่บริษัทมีอยู่ เนื่องจากสภาพเศรษฐกิจและธุรกิจทางด้านเช่าวิดีโอ มีการแข่งขันที่สูงมาก ดังนั้นการนำ Information Technology มาใช้เพื่อให้เกิดประโยชน์กับองค์กร เป็นอีกทางเลือกหนึ่งที่จะช่วยให้องค์กรได้ผลประโยชน์มากขึ้น รวมถึงช่องทางต่างๆและโอกาสทางธุรกิจที่เพิ่มมากขึ้นด้วย

<b>Title</b>	An Analysis of The VDO Rental Process Using ART
<b>Student</b>	Mr. Pichai Krichkamolkul
<b>Advisor</b>	Assistant Professor, Dr. Arit Thammano
<b>Level of Study</b>	Master of Science in Information Technology
<b>Major</b>	Information Science
<b>Academic Year</b>	2002

## ABSTRACT

This project developed data analysis system for VDO rental shop by using Neural Network Technology. Customer group can be classified by applying ART (Adaptive Resonance Theory). This development uses Java language with MS SQL 7.0 database and the database on actual current VDO rental shops. By developing system to be practical, convenient, fast and accuracy, thus it can be used for Data Mining with other information of a company. To compete in current economy and high competitive business like VDO rental shop, Information technology is very important to use in the organization. It can help the organization to gain better benefit and also increase business channel and opportunity.

## กิตติกรรมประกาศ

สำหรับการพัฒนาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอ ด้วย ART ในครั้งนี้ ได้รับการสนับสนุนจากหลาย ๆ ฝ่ายเป็นอย่างมาก ในการให้คำปรึกษา เสนอแนะ ตลอดจนข้อมูลและอุปกรณ์ต่าง ๆ ซึ่งเป็นผลให้การพัฒนาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอ ด้วย ART สามารถบรรลุผลตามเป้าหมายที่วางเอาไว้ได้ ในการนี้ผู้จัดทำใคร่ขอขอบพระคุณบุคคลต่าง ๆ ดังนี้

1. ผศ. ดร. อาริต ธรรมโน อาจารย์ที่ปรึกษาที่กรุณาเสียสละเวลาอย่างไม่เห็นแก่เหน็ดเหนื่อย ในการให้คำปรึกษาและคำแนะนำตลอดมา
2. รศ. ดร. วิเชียร เปรมชัยสวัสดิ์, ผศ. ดร. ประจวบ วานิชชัชวาล และคณาจารย์ทุกท่าน ที่กรุณาให้ความอนุเคราะห์เสมอมา
3. เจ้าหน้าที่ประจำคณะเทคโนโลยีสารสนเทศทุกท่าน ที่ช่วยสนับสนุนในด้านต่าง ๆ
4. บริษัท ซีทาคู (ประเทศไทย) ที่อำนวยความสะดวกในด้านข้อมูล
5. คุณ อารักษ์ พรชัยจรศักดิ์ สำหรับคำแนะนำและข้อเสนอแนะต่างๆ เกี่ยวกับ ARTMAP
6. เพื่อน ๆ โดยเฉพาะ เพื่อนร่วมรุ่น IS10 ทุกคน ที่คอยให้กำลังใจและคำแนะนำตลอดมา
7. บิดา มารดา พี่ๆ และน้องชาย ที่คอยให้กำลังใจและส่งเสริมในด้านการศึกษาเสมอ

พิชัย กริชกมลกุล

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VI
สารบัญตาราง	VIII
บทที่	
1. บทนำ	
1.1 ความเป็นมา	1
1.2 วัตถุประสงค์ของการพัฒนาระบบงาน	2
1.3 ขอบเขตและขั้นตอนการพัฒนาระบบงาน	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
2. ทฤษฎีที่เกี่ยวข้อง	
2.1 ระบบจัดการฐานข้อมูล	4
2.2 Microsoft SQL Server	10
2.3 Java	15
2.4 Data Mining	33
2.5 Adaptive Resonance Theory	40
3. ระบบงานและการวิเคราะห์ร้านเช่าวิดีโอ	
3.1 ลักษณะการดำเนินธุรกิจ	48
3.2 การจัดเก็บข้อมูล	49
3.3 ข้อมูลที่นำมาใช้	50
4. การพัฒนาโปรแกรม	
4.1 การเข้าสู่โปรแกรม	55
4.2 การ Train ชุดข้อมูล	56

## สารบัญ (ต่อ)

	หน้า
4.3 การ Test ข้อมูล	60
5. ผลการทดลองและบทสรุป	
5.1 สรุปผลการศึกษาและพัฒนาระบบ	65
5.2 ข้อเสนอแนะ	66
บรรณานุกรม	67
ประวัติผู้เขียน	69



## สารบัญภาพ

ภาพที่	หน้า
2.1 ส่วนประกอบของระบบจัดการฐานข้อมูล	4
2.2 ขั้นตอนการประมวลผลในภาษาสอบดาม	6
2.3 โครงสร้างของตัวประมวลผลสอบดาม	8
2.4 ตัวอย่างโอเปอร์เรชันกราฟ	10
2.5 การสื่อสารระหว่าง Client-Server	11
2.6 Physical File ใน SQL Server 7.0	13
2.7 โครงสร้างระบบฐานข้อมูล SQL Server 7.0	14
2.8 แสดงการแบ่งระบบฐานข้อมูลของ SQL Server	14
2.9 แสดงโครงสร้าง java ไคเรกทอรี	20
2.10 แสดงการ Setup Java Environments	21
2.11 แสดงโปรแกรม JCreator	22
2.12 ขบวนการทำ Data Mining	35
2.13 เทคนิคของ Kohonen Feature Map	39
2.14 ตัวอย่างการจับกลุ่มแบบ K-means Clustering	40
2.15 สถาปัตยกรรมของ ART1	41
2.16 แสดงการเชื่อมต่อภายใน ART	42
2.17 สถาปัตยกรรมพื้นฐานของ ARTMAP	46
2.18 สถาปัตยกรรม ARTMAP	47
3.1 แสดงลักษณะการเชื่อมต่อเครือข่ายในแต่ละสาขา	49
3.2 แสดงลักษณะการเชื่อมต่อเครือข่ายที่ใช้ติดต่อกับแต่ละสาขา	49
3.3 แสดง Table ทั้งหมดในฐานข้อมูล	50
4.1 แสดงหน้าจอโปรแกรมเมื่อเริ่มต้น	55
4.2 แสดงข้อความเตือนว่าต้องการออกจากระบบหรือไม่	56

## สารบัญภาพ (ต่อ)

ภาพที่	หน้า
4.3 แสดงหน้าจอการ Train ชุดข้อมูล	57
4.4 แสดงการเตือนเมื่อป้อนข้อมูลผิดพลาด	57
4.5 แสดงหน้าจอที่ติดต่อกับฐานข้อมูลแล้ว	58
4.6 แสดงหน้าจอเมื่อการ Train เสร็จสิ้น	59
4.7 แสดงการ Save ข้อมูล	60
4.8 แสดงการโหลดข้อมูล	61
4.9 แสดงหน้าจอ Testing Data	61
4.10 แสดงข้อความเตือน เมื่อระบุข้อมูลในการ Test ไม่ครบ	62
4.11 แสดงผลลัพธ์จากการคำนวณ	62

## สารบัญตาราง

	หน้า
ตารางที่	
3.1 แสดงการเปลี่ยนข้อมูล GENDER เป็นข้อมูลไบนารี	52
3.2 แสดงการเปลี่ยนข้อมูล AGE เป็นข้อมูลไบนารี	52
3.3 แสดงการเปลี่ยนข้อมูล JOB เป็นข้อมูลไบนารี	53
3.4 แสดงการเปลี่ยนข้อมูล INCOME เป็นข้อมูลไบนารี	53
3.5 แสดงการเปลี่ยนข้อมูล GENRUCD	54
5.1 แสดงผลการทดลองจากข้อมูลทั้งหมด	63
5.2 แสดงผลการทดลองช่วงวันที่ 01/06/2002 – 30/11/2002	64
5.3 แสดงผลการทดลองช่วงวันที่ 01/01/2002 – 31/03/2002	64
5.4 แสดงผลการทดลองช่วงวันที่ 01/01/2002 – 31/01/2002	65

# บทที่ 1

## บทนำ

การพัฒนาาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอด้วย ART ได้ทำการศึกษาถึงความเป็นมา, ปัญหาต่างๆ ที่เกิดขึ้นกับการจัดการข้อมูลในร้านเช่าวิดีโอ และการนำ Neural Network มาประยุกต์ใช้งาน จากนั้นจึงกำหนดวัตถุประสงค์ ขอบเขต และขั้นตอนการพัฒนากระบวนการ เพื่อหาแนวทางในการพัฒนาาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอด้วย ART ขึ้น

### 1.1 ความเป็นมา

การนำเทคโนโลยีสารสนเทศมาประยุกต์ใช้ร่วมกับภาคธุรกิจ นับวันยังมีความสำคัญและความจำเป็นอย่างยิ่ง เมื่อเทคโนโลยีก้าวไปอย่างรวดเร็ว ธุรกิจใดๆมีเครื่องมือที่ใช้เทคโนโลยีสูงกว่าย่อมเปิดช่องและโอกาสให้กับตัวเองได้มากขึ้น และเมื่อปัจจุบันการแข่งขันทางด้านธุรกิจที่สูงขึ้น ส่งผลให้ “คลังข้อมูล” เป็นส่วนสำคัญต่อความสำเร็จเปรียบทางธุรกิจ จึงได้มีการพัฒนาระบบจัดเก็บข้อมูล, การเรียกค้น และระบบช่วยตัดสินใจที่ฐานข้อมูลทั่วไปไม่สามารถทำได้ ดังนั้นการได้มาของข้อมูลสำคัญต่างๆมาใช้ในการบริหาร, การจัดการ, การวางแผน, การบริการเพื่อสร้างความสัมพันธ์ที่ดีกับลูกค้า และการตัดสินใจในการดำเนินธุรกิจในอนาคต จึงมีความสำคัญอย่างยิ่ง ดังนั้นจึงเกิดเทคโนโลยีที่เรียกว่า “Data Mining” ที่มาช่วยวิเคราะห์, หาความสัมพันธ์ และข้อมูลสำคัญต่างๆที่อาจซ่อนอยู่ภายใต้ฐานข้อมูลขนาดใหญ่

แนวโน้มข้อมูลและสารสนเทศต่างๆ จะเพิ่มสูงมากขึ้น เกือบทุกอย่างองค์กรจะจัดทำเก็บข้อมูลต่างๆไว้ ถึงแม้จะมีการเก็บข้อมูลจำนวนมากแล้ว แต่ไม่สามารถ หาความหมายและสิ่งที่ก่อให้เกิดความรู้ได้ ก็เท่ากับใช้ทรัพยากรที่มีอยู่ไม่คุ้มค่า ดังนั้นจึงมีการคิด ถึงความสามารถของข้อมูลที่มีอยู่มากก่อให้เกิดสารสนเทศที่เป็นประโยชน์แก่ผู้ใช้มากที่สุด ซึ่งหนึ่งในเทคโนโลยีนั้นคือ Data Mining

จากการที่ธุรกิจร้านเช่าวิดีโอมีอัตราการเติบโตขึ้นอย่างรวดเร็ว นอกจากแข่งขันกันระหว่างร้านเช่าวิดีโอด้วยกันแล้ว ยังจะต้องแข่งขันกับธุรกิจที่ถูกและไม่ถูกกฎหมาย เช่น แผ่นวีซีดี (VCD) ทั้งแผ่นเถื่อนและไม่เถื่อนที่มีราคาถูก ส่งผลให้ลูกค้ามีพฤติกรรมการบริโภคที่เปลี่ยนไป จากเดิมซึ่งนิยมการเช่าวิดีโอ ก็หันมานิยมซื้อวีซีดีราคาถูกแทน นอกจากนี้ การที่เทคโนโลยีที่เปลี่ยนไป ทำให้พฤติกรรมของลูกค้าเปลี่ยนแปลงไปด้วย ไม่ว่าจะเป็น การออกแผ่นดีวีดี (DVD), เอส-ดีวีดี (S-DVD)

และการที่เครื่องเล่นทั้งแบบ วีซีดี และ ดีวีดี ที่มีราคาถูกลง ทำให้ผู้บริโภคหันไปหาสิ่งเหล่านี้แทน การดูวิดีโอเพียงอย่างเดียว อีกทั้งปัจจุบันธุรกิจเช่าวิดีโอไม่ได้ให้บริการแก่ การเช่าวิดีโอเท่านั้น ยังมี การขายสินค้าชนิดต่างๆ ไม่ว่าจะ การขาย วีซีดี, ดีวีดี และอื่นๆ ก็เพื่อกระตุ้นยอดขายให้กับองค์กรของตน ดังนั้นจึงทำให้ผู้ประกอบการจะต้องกระตือรือร้นและเร่งที่จะพัฒนาร้านวิดีโอของตนให้ดียิ่งขึ้น ลดค่าใช้จ่ายต่างๆ รวมทั้งต้องทันสมัย ทันต่อเหตุการณ์มากขึ้น เพื่อสร้างความประทับใจและดึงดูดลูกค้าให้มาใช้บริการเช่าวิดีโอ ซึ่งหนึ่งในสิ่งที่จะมาช่วยให้ร้านเช่าวิดีโอวิเคราะห์สภาพต่างๆ ทั้ง การบริหาร, การตลาด, การบริการ และสิ่งต่างๆอีกมากมาย คือการนำ สารสนเทศ ที่มีอยู่มาใช้ให้เกิดประโยชน์สูงสุด และการนำ Neural Network ชนิด ARTMAP มาช่วยในการจำแนก แยกแยะ การเช่าของลูกค้า เป็นอีกวิธีหนึ่งที่จะเพิ่มศักยภาพ การวิเคราะห์ตลาด และ ระบบช่วยตัดสินใจต่างๆ เพื่อ เพิ่มยอดขาย, ลดค่าใช้จ่าย หรือการให้บริการต่างๆ แก่ลูกค้า

## 1.2 วัตถุประสงค์ ของการพัฒนาระบบงาน

วัตถุประสงค์ของการพัฒนาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอด้วย ART ก็เพื่อช่วยวิเคราะห์ลูกค้าในด้านต่างๆ โดยการนำ ARTMAP มาใช้ในการทำนายลักษณะการเช่าวิดีโอของลูกค้า โดยมีรายละเอียดดังต่อไปนี้

1. เพื่อศึกษาแนวทางและค้นหากระบวนการการทำตลาด ให้ตรงกับความต้องการของลูกค้ามากที่สุด นอกเหนือจากวิธีการที่ใช้อยู่ในปัจจุบัน
2. เพื่อสร้างความสัมพันธ์ที่ดีและความประทับใจให้กับลูกค้า
3. เพื่อเพิ่มประสิทธิภาพการดำเนินงานให้กับธุรกิจร้านเช่าวิดีโอ
4. เพื่อลดค่าใช้จ่ายในการสต็อกสินค้า และการส่งสินค้าเข้ามาในร้าน
5. เพื่อให้ได้รับผลตอบแทนที่ดีที่สุดในการประกอบการ

## 1.3 ขอบเขตและขั้นตอนการพัฒนาบบงาน

ระบบงานนี้จะทำการพัฒนาการดำเนินงานภายในร้านเช่าวิดีโอด้วยระบบคอมพิวเตอร์ ภายใต้ขอบเขตและขั้นตอนการทำงานดังนี้

1. ศึกษาและวิเคราะห์ระบบการดำเนินงานภายในร้านเช่าวิดีโอปัจจุบัน
2. ศึกษาถึงปัญหาในด้านต่าง ๆ รวมทั้งความเป็นไปได้ในการพัฒนาระบบงาน
3. ศึกษาโครงสร้างฐานข้อมูลที่ใช้ภายในร้าน รวมถึงการนำมาประยุกต์ใช้งาน
4. จัดเก็บฐานข้อมูลและวิเคราะห์ความเป็นไปได้ ในการพัฒนาระบบ

5. ทำการพัฒนาและทดลองใช้ระบบงานกับตัวอย่างฐานข้อมูล เพื่อหาข้อบกพร่อง และพิจารณาความถูกต้อง เพื่อนำผลที่ได้มาปรับปรุงระบบให้มีประสิทธิภาพสูงสุดต่อไป

#### 1.4 ประโยชน์ที่คาดว่าจะได้รับ

ความสำเร็จในการศึกษาและพัฒนากระบวนการนี้ จะก่อให้เกิดประโยชน์หลายประการ ดังต่อไปนี้

1. ลดค่าใช้จ่ายการสต็อกสินค้าและการส่งสินค้าที่ไม่ตรงตามความต้องการของตลาด
2. ช่วยฝ่ายการตลาดวิเคราะห์กลุ่มลูกค้าได้แม่นยำขึ้น
3. เพิ่มศักยภาพด้านยอดขาย-ขายสินค้าต่างๆ ในแต่ละสาขา
4. เพิ่มภาพพจน์และความพอใจให้กับแฟนโซเชียลมีเดีย ในศักยภาพของฝ่ายการตลาดและบริหารของบริษัท
5. เพื่อนำไปพัฒนาให้ใช้กับเทคนิคต่างๆ ใน Data Mining ด้วยวิธีการอื่นๆ ต่อไปได้ เช่น ID3 หรือ Neural Network ชนิดต่างๆ

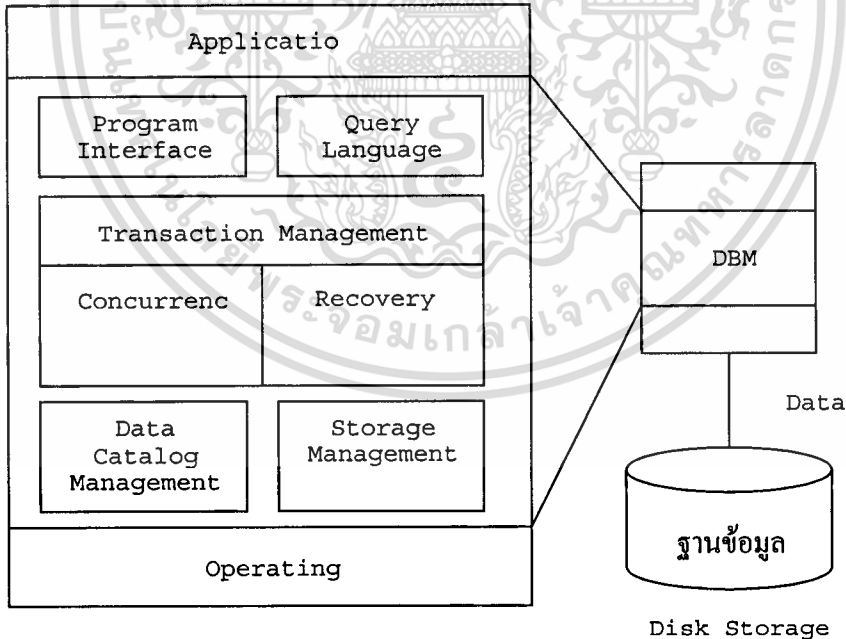
## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

#### 2.1 ระบบจัดการฐานข้อมูล

ฐานข้อมูลมีบทบาทสำคัญมากต่องานด้านต่าง ๆ โดยเฉพาะอย่างยิ่งกับระบบงานที่ใช้คอมพิวเตอร์ เช่น งานด้านธุรกิจ วิศวกรรม การแพทย์ การศึกษาและวิทยาศาสตร์ เป็นต้น กล่าวอย่างง่าย ความหมายของฐานข้อมูลก็คือ การจัดเก็บข้อมูลอย่างมีระบบ ซึ่งผู้ใช้สามารถเรียกใช้ข้อมูลในลักษณะต่าง ๆ ได้ เช่น การเพิ่มเติม เรียกดู แก้ไขหรือลบข้อมูล โดยทั่วไปการจัดเก็บข้อมูลจะมีการนำระบบคอมพิวเตอร์เข้ามาช่วยในการจัดการฐานข้อมูล

ระบบจัดการฐานข้อมูล คือ ซอฟต์แวร์หรือโปรแกรมที่ใช้จัดการฐานข้อมูล ทั้งนี้ระบบจัดการฐานข้อมูล จะต้องทำงานประสานกับระบบปฏิบัติการ (Operating System) และโปรแกรมประยุกต์ที่ใช้งานฐานข้อมูลนั้น ๆ ระบบจัดการฐานข้อมูลมีส่วนประกอบ ดังภาพที่ 2.1



ภาพที่ 2.1 ส่วนประกอบของระบบจัดการฐานข้อมูล

หน้าที่หลักของระบบจัดการฐานข้อมูล มีรายละเอียดดังนี้

1. การจัดการ ทรานแซกชัน (Transaction Management)
2. การฟื้นฟูสภาพ (Recovery)
3. การควบคุมภาวะพร้อมกัน (Concurrency Control)
4. การประมวลผลสอบถาม (Query Processing)
5. การจัดการด้านความปลอดภัยของข้อมูล (Security Management)
6. การให้บริการภาษาฐานข้อมูล (Database Language Service)
7. การให้บริการที่ทนต่อความผิดพลาด (Fault Tolerance)
8. การจัดการแคตาล็อกของข้อมูล (Data Catalog Management)
9. การจัดการด้านการจัดเก็บข้อมูล (Storage Management)

หน้าที่ที่กล่าวมาข้างต้น เป็นหน้าที่ที่ควรมีในระบบจัดการฐานข้อมูล ซึ่งในที่นี้จะกล่าวถึงหน้าที่การประมวลผลสอบถาม ดังรายละเอียดถัดไป

### 2.1.1 การประมวลผลสอบถาม

ในที่นี้จะกล่าวถึงความหมายของการประมวลผลสอบถาม ขั้นตอนการประมวลผลในภาษาสอบถาม และโครงสร้างของตัวประมวลผลสอบถาม

#### 2.1.1.1 ความหมายของการประมวลผลสอบถาม

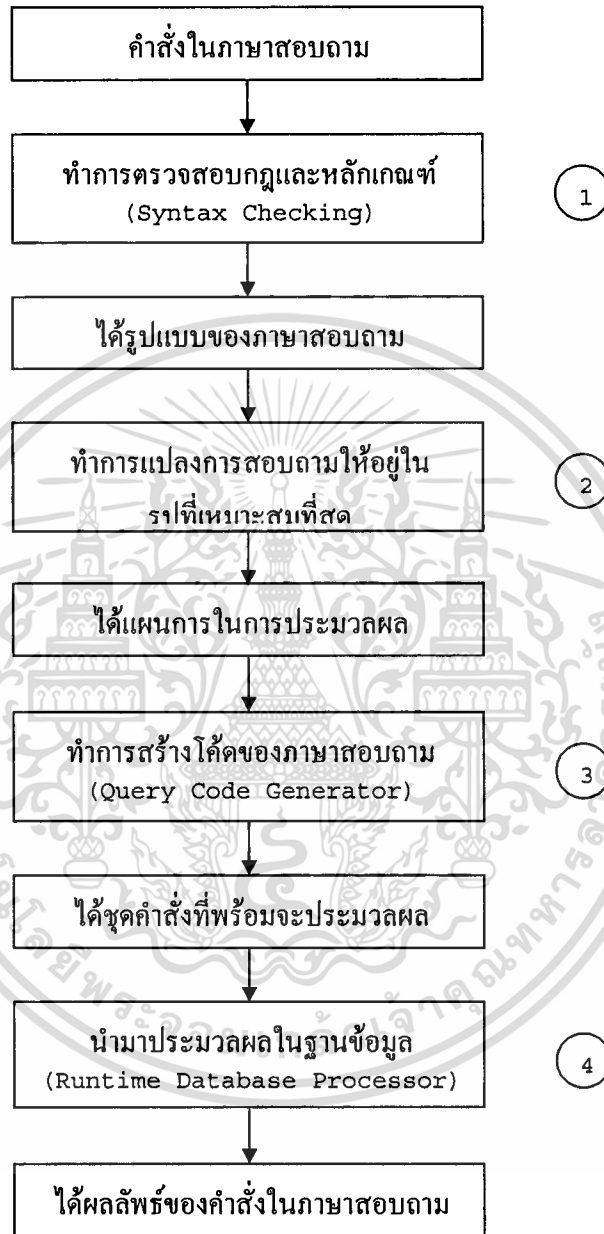
การประมวลผลสอบถาม (Query Processing) เป็นหน้าที่หนึ่งของระบบการจัดการฐานข้อมูล โดยระบบการจัดการฐานข้อมูลจะทำการประมวลผลสอบถามที่เกิดจากภาษาสอบถาม (Query Language) ได้แก่ ภาษา SQL ภาษา QBE เป็นต้น และหาคำตอบให้แก่ผู้ใช้ ได้อย่างเหมาะสมและรวดเร็ว

การประมวลผลสอบถาม เป็นกระบวนการหนึ่งของระบบจัดการฐานข้อมูล ที่ใช้ในการเลือกแผนหรือกลยุทธ์ที่เหมาะสมในการสอบถามข้อมูลในฐานข้อมูล โดยระบบจัดการฐานข้อมูลจะมีตัวประมวลผลที่เรียกว่า ตัวประมวลผลสอบถาม (Query Processor) ทำหน้าที่ในการเลือกแผนหรือกลยุทธ์ที่เหมาะสมในการเข้าถึงข้อมูล

#### 2.1.1.2 ขั้นตอนของการประมวลผลในภาษาสอบถาม

การประมวลผลสอบถาม จะทำการประมวลผลเป็นขั้นตอนต่าง ๆ โดยจะนำคำสั่งภาษาสอบถามที่ผู้ใช้ต้องการ มาตรวจสอบกับกฎและไวยากรณ์ (Rule of grammars) ของภาษาสอบถามนั้น ๆ ระบบการจัดการฐานข้อมูลจะใช้กลยุทธ์การประมวลผล (Execution Strategy)

สำหรับผลลัพธ์ของคำสั่งภาษาสอบถามนั้น ๆ จะทำให้ได้ผลลัพธ์ในเวลาอันรวดเร็ว ขั้นตอนในการประมวลผลสอบถาม ดังภาพที่ 2.2 มีขั้นตอนดังนี้



ภาพที่ 2.2 ขั้นตอนการประมวลผลในภาษาสอบถาม

**ขั้นตอนที่ 1 การตรวจสอบกฎไวยากรณ์ของภาษา (Syntax Checking)** เมื่อผู้ใช้ได้ทำการป้อนคำสั่งภาษาสอบถาม ระบบจัดการฐานข้อมูลโดยตัวประมวลผลสอบถาม จะทำการ

ตรวจสอบกฎและหลักเกณฑ์ของภาษาสอบถามนั้น เมื่อตรวจสอบแล้วหากพบว่ามีคำสั่งภาษาสอบถามที่ป้อนเข้ามายังระบบมีความผิดพลาด ไม่เป็นไปตามกฎเกณฑ์ของภาษาสอบถาม ระบบจัดการฐานข้อมูลโดยตัวประมวลผลสอบถาม ก็จะส่งข้อความไปยังผู้ใช้ให้ผู้ใช้ปรับแก้ให้ถูกต้อง แต่ถ้าตรวจสอบแล้วคำสั่งนั้นถูกต้องตามกฎไวยากรณ์ของภาษาสอบถาม จะแปลทำให้ได้รูปแบบของภาษาสอบถาม (Intermediate form of query) เพื่อให้ระบบจัดการฐานข้อมูลสามารถเข้าใจได้ ซึ่งผลที่แปลจะอยู่ในรูปของสมการพีชคณิตแบบสัมพันธ์ (Relational Algebraic Expression) เพื่อให้ระบบจัดการฐานข้อมูลเข้าใจและทำตามผลที่ได้นั้น เช่น ต้องการหารายชื่อนักศึกษาที่ลงทะเบียนในวิชาฐานข้อมูล ซึ่งสามารถเขียนเป็นภาษา SQL ได้ดังนี้

```
select Std_Name, Course_Name
from STUDENT, REGISTRATION, COURSE
where STUDENT.Std# = REGISTRATION.Course# and
COURSE.Course# = REGISTRATION.Course# and
REGISTRATION.Course# = 'Database';
```

เมื่อป้อนคำสั่งดังกล่าวข้างต้นเข้าสู่ระบบ ระบบจะทำการตรวจสอบ โดยดูว่าคำสั่งที่ผู้ใช้ป้อนเข้าสู่ระบบนั้นมีความผิดพลาดหรือไม่ ถ้าพบว่าผิดพลาดระบบจะส่งข้อความแจ้งให้ผู้ใช้ทราบเพื่อปรับแก้ให้ถูกต้อง แต่ถ้าคำสั่งที่ป้อนเข้าสู่ระบบนั้นถูกต้องแล้ว ระบบจะสร้างสมการพีชคณิตแบบสัมพันธ์ดังนี้

$$\pi_{Std\_Name}(\sigma_{Course\_Name='Database'}(STUDENT \bowtie REGISTRATION \bowtie COURSE))$$

## ขั้นตอนที่ 2 การแปลงการสอบถามให้อยู่ในรูปที่เหมาะสมที่สุด (Query Optimizer)

เมื่อได้สมการพีชคณิตแบบสัมพันธ์แล้ว ระบบจัดการฐานข้อมูลจะทำการประมวลผลสมการพีชคณิตแบบสัมพันธ์นั้น เพื่อให้ได้แผนการเข้าถึงข้อมูล ในการประมวลผลในการเข้าถึงข้อมูล (Access) นั้นจะมีวิธีการเข้าถึงฐานข้อมูลได้หลายวิธี ระบบจัดการฐานข้อมูลจะทำการหาวิธีที่เหมาะสมในการเข้าถึงฐานข้อมูล โดยจะมีเทคนิคต่าง ๆ ในการเข้าถึงข้อมูลที่ใช้เวลาน้อยที่สุดหรือเรียกว่าการทำออฟติไมเซชัน (Optimization) ซึ่งเทคนิคในการเข้าถึงฐานข้อมูลที่เหมาะสมเหล่านี้ เช่น เทคนิคการซอร์ตและการเมอร์จ (Sort and Merge Method) เป็นการเรียงลำดับข้อมูลในตารางข้อมูล โดยในระบบจัดการฐานข้อมูลจะมีโปรแกรมสำหรับจัดเรียงลำดับข้อมูล เพื่อให้ประหยัดเวลาในการประมวลผล เทคนิคดีคอมโพสิชัน (Decomposition Method) เป็นการแตกคำสั่งในภาษาสอบถาม ให้เป็นโมดูลย่อย ๆ เพื่อง่ายต่อการประมวลผล หรือเทคนิคโอเปอเรเตอร์กราฟ

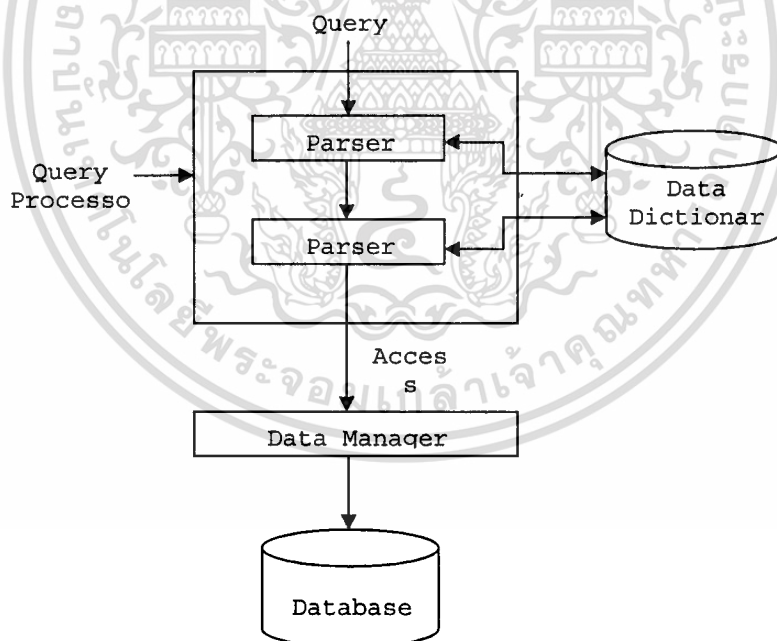
(Operator Graph) เป็นการนำสมการพีชคณิตที่ได้จากการตรวจสอบกฎไวยากรณ์ของภาษา แล้วนำมาหาทางประมวลผลที่ทำให้การประมวลผลนั้นใช้เวลาน้อย เทคนิคต่าง ๆ เหล่านี้จะทำให้ได้แผนการในการประมวลผลที่เหมาะสมว่า ระบบจะต้องประมวลผลอย่างไร เพื่อให้ได้วิธีการเข้าถึงฐานข้อมูลที่มีประสิทธิภาพและใช้เวลาในการประมวลผลน้อยที่สุด

**ขั้นตอนที่ 3 การสร้างชุดคำสั่งของภาษาสอบถาม (Query Code Generation)** เมื่อได้วิธีที่เหมาะสมที่สุดในการเข้าถึงฐานข้อมูลแล้ว ระบบจัดการฐานข้อมูลโดยตัวประมวลผลสอบถาม จะทำการสร้างชุดคำสั่งของภาษาสอบถาม ทำให้ได้ชุดคำสั่ง (Source Code) ที่พร้อมจะประมวลผล

**ขั้นตอนที่ 4 การประมวลผลในฐานข้อมูล (Runtime Database Processor)** เมื่อได้ชุดคำสั่งที่พร้อมจะประมวลผลแล้ว ตัวประมวลผลสอบถามก็จะทำการคอมไพล์ (Compile) หรือแปลคำสั่ง (Interpret) และประมวลผล (Executed) ชุดคำสั่งจนได้ผลลัพธ์

### 2.1.11 โครงสร้างของตัวประมวลผลสอบถาม

ตัวประมวลผลสอบถามเป็นส่วนประกอบหนึ่งของระบบจัดการฐานข้อมูล ที่ทำหน้าที่ในการเลือกแผนหรือกลยุทธ์ที่เหมาะสม ตัวประมวลผลสอบถามมีโครงสร้างดังภาพที่ 2.3 ซึ่งมีส่วนประกอบดังนี้



ภาพที่ 2.3 โครงสร้างของตัวประมวลผลสอบถาม

## 1. ส่วนตรวจสอบภาษา (Parser)

เป็นส่วนหนึ่งของตัวประมวลผลสอบถามที่ทำหน้าที่ตรวจสอบกฎและหลักเกณฑ์ของคำสั่งสอบถามข้อมูลที่กำหนดไว้ในระบบฐานข้อมูล โดยจะทำการตรวจสอบข้อมูลที่อยู่ในพจนานุกรมข้อมูล แล้วทำการแปลคำสั่งให้อยู่ในรูปของสมการพีชคณิตแบบสัมพันธ์ ที่ระบบจัดการฐานข้อมูลสามารถเข้าใจได้

ตัวอย่างของภาษา SQL เพื่อการสอบถามข้อมูลจากราง COURSE โดยทำการค้นหาข้อมูลที่มีรหัสวิชา (Course#) คือ COM353 ที่มี Smith เป็นผู้สอน มีดังนี้

```
select Course#
from COURSE
where Course# = COM353 and
Instructor = 'Smith';
```

จากคำสั่งภาษา SQL ดังกล่าวข้างต้น ส่วนตรวจสอบภาษาจะทำการแปลคำสั่งในการสอบถามจากภาษา SQL ให้อยู่ในรูปของสมการพีชคณิตแบบสัมพันธ์ได้ดังนี้

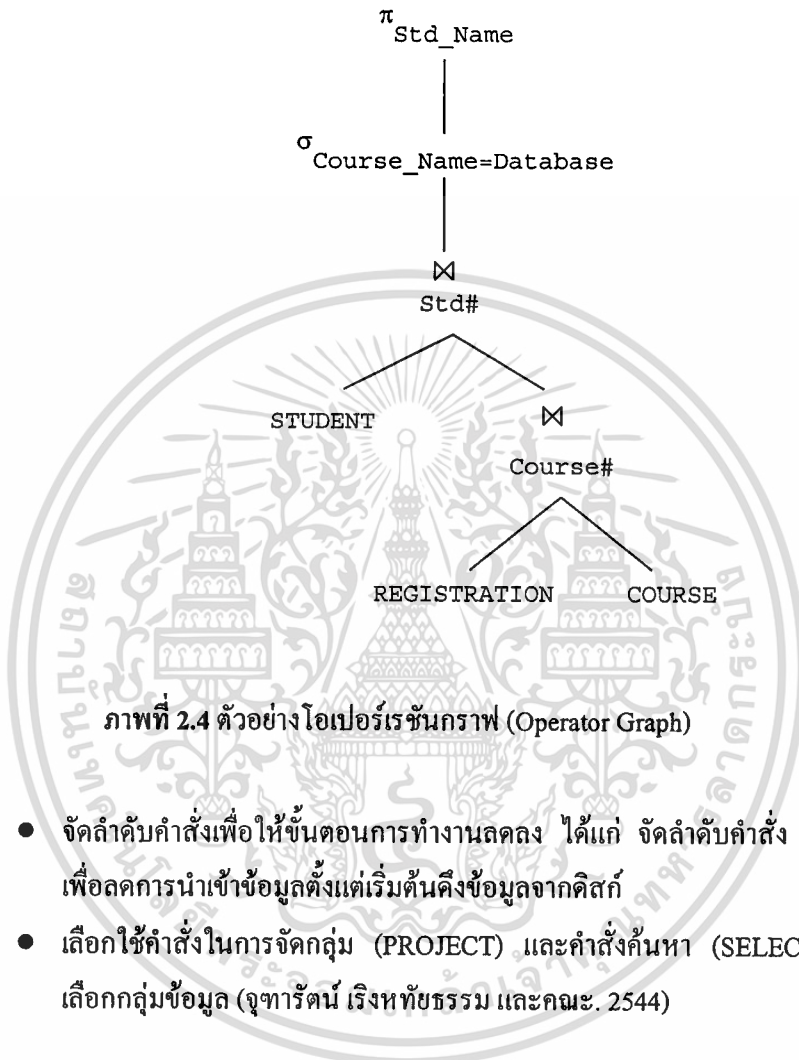
$$C_{\text{Course\#} = \text{COM353} \wedge \text{Instructor} = \text{'Smith'}} (\text{COURSE})$$

## 2. ออฟติไมเซอร์ (Optimizer)

เป็นส่วนหนึ่งของตัวประมวลผลสอบถาม ที่ทำหน้าที่ในการหากลยุทธ์หรือวิธีการที่เหมาะสมเพื่อให้การเข้าถึงข้อมูลประหยัดเวลาที่สุด เมื่อคำสั่งในภาษาสอบถามผ่านตัวตรวจสอบภาษาแล้ว จะได้คำสั่งในการสอบถามที่อยู่ในรูปของสมการพีชคณิตแบบสัมพันธ์ ซึ่งสามารถประมวลผลได้หลายวิธี ด้วยเทคนิคต่าง ๆ ในการหาทางเลือกที่เหมาะสม (Optimization) ออฟติไมเซอร์จะทำการเลือกการเข้าถึงข้อมูล (Access Plan) โดยการเลือกแผนการเข้าถึงข้อมูลซึ่งจะต้องคำนึงถึงประเด็นต่อไปนี้

- การลดปริมาณการดึงหรือเรียกใช้ข้อมูล
- การลดขนาดของโครงสร้างข้อมูล โดยพิจารณาจากการใช้เทคนิคการลดการทำงานระหว่างการดึงข้อมูล
- เพิ่มความสามารถในการดึงข้อมูลด้วยการเรียงลำดับไว้ก่อน การจัดเรียงลำดับใหม่อาจจะช่วยลดการทำงานบางอย่าง เพื่อช่วยให้การประมวลผลสอบถามสามารถทำงานได้ดีขึ้น โดยออฟติไมเซอร์อาจสร้างสมการพีชคณิตแบบสัมพันธ์ให้อยู่ในรูปแบบของโครงสร้างต้นไม้ที่อยู่ในรูปของกราฟ (Operator Graph) ดัง

ภาพที่ 2.4 เป็นโอเปอเรเตอร์กราฟของการค้นหารายชื่อของนักศึกษาที่ลงทะเบียนเรียนวิชาฐานข้อมูล



- จัดลำดับคำสั่งเพื่อให้ขั้นตอนการทำงานลดลง ได้แก่ จัดลำดับคำสั่ง SELECT เพื่อลดการนำเข้าข้อมูลตั้งแต่เริ่มต้นดึงข้อมูลจากดิสก์
- เลือกใช้คำสั่งในการจัดกลุ่ม (PROJECT) และคำสั่งค้นหา (SELECT) ในการเลือกกลุ่มข้อมูล (จุฑารัตน์ เรืองทัญชรรม และคณะ. 2544)

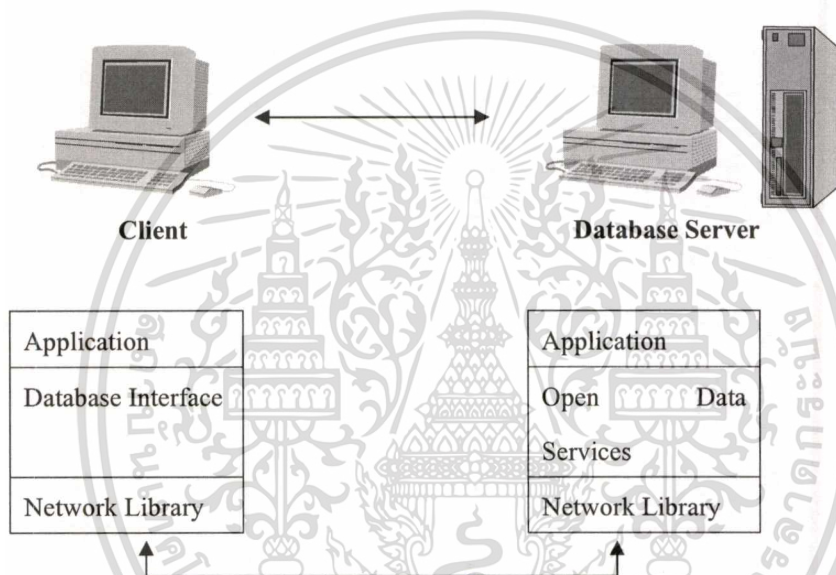
## 2.2 Microsoft SQL Server

เป็นรุ่นของผลิตภัณฑ์ด้านฐานข้อมูลของ Microsoft ที่สร้างขึ้นมาจาก SQL Server version 6.5 ซึ่งเป็นฐานข้อมูลที่ทรงประสิทธิภาพที่สุดสำหรับระบบปฏิบัติการ Windows, SQL Server เป็นอีกทางเลือกหนึ่งของ ระบบฐานข้อมูลแบบความสัมพันธ์ (Relational Database Management system (RDBMS)) สำหรับลูกค้าระดับองค์กร และบริษัทพัฒนาซอฟต์แวร์อิสระ (Independent Software vendors (ISVs)) ที่สร้างโปรแกรมประยุกต์ทางด้านธุรกิจ ความจำเป็นและความต้องการ

ของลูกค้าทำให้ผลิตภัณฑ์ถูกพัฒนาให้ง่ายต่อการใช้ มีความเชื่อถือสูง มีความยืดหยุ่น และมี data warehousing

### 2.2.1 สถาปัตยกรรมของ SQL Server

SQL Server จะมีส่วนที่ทำหน้าที่ติดต่อสื่อสารกับทางฝั่งไคลเอนต์อยู่ด้วย แต่ก็แยกส่วนที่จัดการเน็ตเวิร์กและโปรโตคอลออกจากส่วนที่เป็นแอปพลิเคชัน ทำให้แอปพลิเคชันนี้สามารถทำงานอยู่บนเน็ตเวิร์กแบบใดก็ได้ ดังภาพด้านล่าง



ภาพที่ 2.5 การสื่อสารระหว่าง Client-Server

- **Application** หมายถึง โปรแกรมที่พัฒนาขึ้นเพื่อใช้ข้อมูลจากดาต้าเบสโดยผ่านอินเตอร์เฟซของโปรแกรมที่เรียกว่า API (Application Programming Interface เป็นวิธีการเขียนโปรแกรมให้สามารถติดต่อกับดาต้าเบส)
- **Database Interface** หมายถึง อินเตอร์เฟซที่ใช้โดยแอปพลิเคชันเพื่อติดต่อไปยัง SQL Server เช่น ODBC (Open Database Connectivity เป็น engine หนึ่งในการสร้างทางติดต่อระหว่างแอปพลิเคชันและดาต้าเบส), OLE DB (Object Linking and Embedding Database)
- **Network Library** หรือเรียกว่า Net-Library หมายถึง ซอฟต์แวร์ที่ทำหน้าที่จัดส่งข้อมูลผ่านไปทางเน็ตเวิร์ก โปรแกรมนี้จะต้องติดตั้งทั้งด้านของเซิร์ฟเวอร์และไคลเอนต์ สำหรับ

SQL Server มี Net-Library ที่จะทำงานกับโปรโตคอลได้หลายประเภท เช่น TCP/IP, Name Pipes, IPX/SPX, Banyan VINES/IP และ AppleTalk ADSP

- **Open Database System** เป็นอินเทอร์เฟซระหว่าง Net-Library และแอปพลิเคชันทางฝั่งเซิร์ฟเวอร์ เช่น SQL Server เอง และ Extended stored procedure ต่างก็เป็นแอปพลิเคชันทางฝั่งเซิร์ฟเวอร์ Open Data Service นี้จะทำหน้าที่ในการเชื่อมต่อ และรับ Request จากไคลเอนต์แล้วส่งต่อไปให้กับ SQL Server ทำการประมวลผลรวมทั้งรับผลที่ได้นั้นและส่งกลับไปที่ไคลเอนต์ของ SQL Server

### 2.2.2 การทำงานร่วมกับแอปพลิเคชัน

การพัฒนาโปรแกรมเพื่อให้ติดต่อกับดาต้าเบสทำได้โดยใช้ชุดคำสั่ง หรือฟังก์ชันที่ SQL Server จัดเตรียมไว้ให้ที่เรียกว่า API ซึ่งวิธีการที่จะติดต่อกับดาต้าเบส มีดังนี้คือ

- **OLE DB (Object Linking and Embedding Database)** เป็นอินเทอร์เฟซชนิด Component Object Model ที่สนับสนุนการทำงานของแอปพลิเคชันที่ใช้ OLE DB การทำงานของ OLE DB ได้ถูกออกแบบมาเพื่อใช้กับ Relation Database และได้มีการพัฒนาให้สามารถติดต่อกันระหว่างดาต้าเบสหลาย ๆ แบบ เช่น ACCESS, ORACLE, SYBASE เป็นต้น
- **ODBC (Open Database Connectivity)** เป็นอินเทอร์เฟซที่ช่วยในการติดต่อกับดาต้าเบส หรือ แอปพลิเคชันทั้งหลายที่ใช้ ODBC
- **Data Object Interface** มี 2 ส่วน คือ

**ADO (ActiveX Data Objects)** เป็นออบเจกต์ที่ช่วยให้พัฒนาแอปพลิเคชันด้วย OLE DB1 ทำได้ง่ายขึ้น เพราะ OLE DB เป็น API ระดับ low level มักนำ ADO มาใช้งานในหลาย ๆ ภาษา เช่น Microsoft Visual basic, Microsoft C++, Active Server Page

**RDO (Remote Data Objects)** เป็นออบเจกต์ที่ช่วยให้พัฒนาแอปพลิเคชันด้วย OLE DB ทำได้ง่ายขึ้นเช่นกัน แต่มักใช้ในกรณีที่ใช้ข้อมูลข้ามเครื่อง

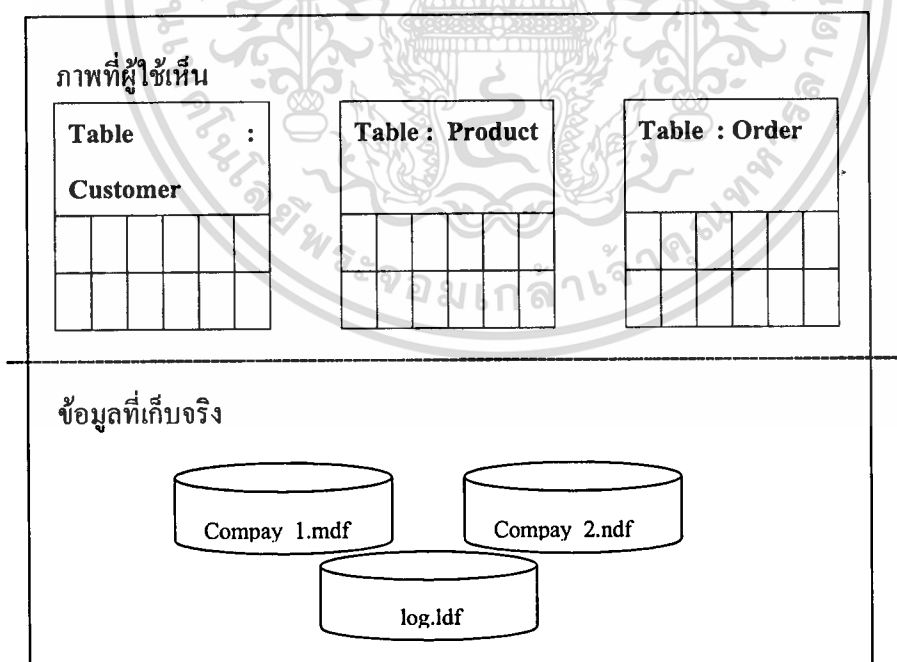
### 2.2.3 สถาปัตยกรรมของดาต้าเบส SQL Server

SQL Server เก็บข้อมูลทั้งหมดไว้ในดาต้าเบส โดยข้อมูลในดาต้าเบสจะถูกจัดแบ่งเป็นส่วนประกอบต่าง ๆ ที่ผู้ใช้ทั่วไปเรียกใช้ได้ เช่น Table, View, StoreProcedure และผู้ใช้ก็จะมองไม่เห็นหรือข้ามส่วนที่เป็น physical ของดาต้าเบสซึ่งเก็บอยู่ในรูปของไฟล์ไป มีแต่เพียงผู้ดูแลระบบเท่านั้นที่ต้องรู้จักและจัดการกับส่วนที่เป็นไฟล์ของดาต้าเบสนี้ ซึ่งไฟล์ของดาต้าเบสจะเก็บอยู่บน

ดิสก์ตามไครว์ที่กำหนดไว้เมื่อครั้งติดตั้งโปรแกรม SQL Server โดยทั่วไปจะกำหนดให้อยู่ในไฟล์ เดอร์ MSSQL7\DATA

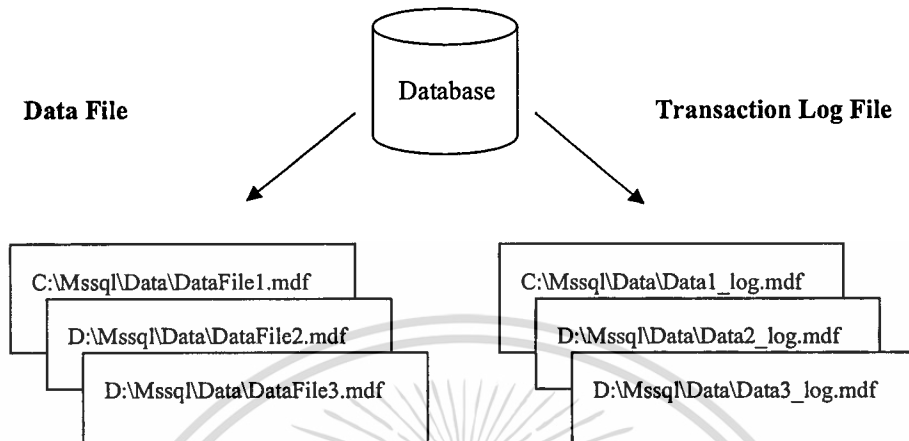
ภาพต่อไปนี้จะแสดงให้เห็นถึงโครงสร้างดาต้าเบสที่ลิงค์ไปยังไฟล์ที่อยู่บนดิสก์ หรือที่ เรียกว่า Physical File

- **Primary File** มีนามสกุลที่เป็น .mdf เก็บข้อมูลที่จำเป็นต้องใช้ในการ start database ตัวนั้น รวมทั้งใช้เก็บข้อมูลด้วย สำหรับดาต้าเบสแต่ละประเภทจะมีเพียง หนึ่ง primary file เท่านั้น
- **Secondary File** มีนามสกุลที่เป็น .ndf เป็นไฟล์ที่ใช้เก็บข้อมูลที่ไม่สามารถเก็บไว้ใน primary file ได้ ในกรณีที่ primary file มีขนาดใหญ่มาก และสามารถเก็บ ข้อมูลได้หมดก็ไม่ต้องสร้าง secondary file แต่ถ้าข้อมูลมาก อาจจะสร้าง secondary file หลาย ๆ ไฟล์เพื่อใช้เก็บข้อมูลเหล่านั้น โดยแยกแต่ละไฟล์ไว้คนละ ไครว์ได้
- **Transaction Log File** จะเก็บไฟล์ที่มีนามสกุลเป็น .ldf ใช้เก็บข้อมูลที่จำเป็นต้อง ใช้ในการกู้ดาต้าเบส เมื่อดาต้าเบสเกิดเสียหายขึ้นมา ซึ่งแต่ละดาต้าเบสจะต้องมี อย่างน้อย 1 transaction log และแต่ละไฟล์มีขนาดไม่ต่ำกว่า 512 MB



ภาพที่ 2.6 Physical File ใน SQL Server 7.0

## 2.2.4 โครงสร้างระบบฐานข้อมูลของ SQL Server



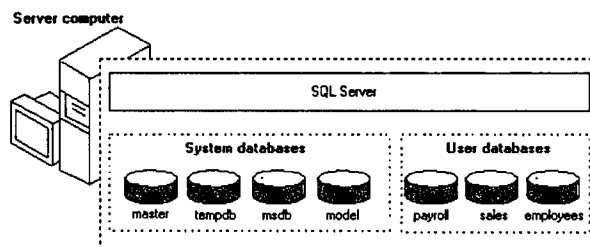
ภาพที่ 2.7 โครงสร้างระบบฐานข้อมูล SQL Server 7.0

ดาต้าเบสของ SQL Server ประกอบด้วยไฟล์อย่างน้อย 2 ไฟล์ คือ

- **Data File** เป็นไฟล์ที่เก็บข้อมูลจริง
- **Transaction Log File** เป็นไฟล์สำหรับเก็บโอเปอเรชั่น หรือข้อมูลล็อก (log) ซึ่งเกิดขึ้นจากการทำงานกับข้อมูลจริง

โดย SQL Server 7.0 สามารถสร้างไฟล์ข้อมูล หรือไฟล์ทรานแซคชันล็อก มากกว่า 1 ไฟล์ได้ ทั้งนี้เพื่อแยกการทำงานที่ชัดเจน และประโยชน์ในแง่ของการบำรุงรักษาอีกด้วย

ระบบฐานข้อมูล SQL Server จะแบ่งเป็น 2 ส่วน คือ System Database และ User Database โดยที่ User Database สามารถมีได้หลาย ๆ ตัว เช่น payroll, Sale และ employees ดังภาพด้านล่าง



ภาพที่ 2.8 แสดงการแบ่งระบบฐานข้อมูลของ SQL Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**System Database** คือ คาด้าเบสหลักของระบบ SQL Server โดยจะมีคาด้าเบส ดังต่อไปนี้

- **Master** เป็นคาด้าเบสที่เก็บข้อมูลซึ่ง SQL Server ใช้ทำงาน ได้แก่ ข้อมูล configuration ของ SQL Server, รหัส login, ข้อมูลหลักของคาด้าเบสทุกตัว เช่น ชื่อ คาด้าเบส, ชื่อของไฟล์บนดิสก์ที่ลิงก์ไปยังคาด้าเบส, storeprocedure เป็นต้น
- **Model** เป็นคาด้าเบสต้นแบบสำหรับการสร้างคาด้าเบสขึ้นใหม่ใน SQL Server รวมถึงใช้เป็นต้นแบบในการสร้าง tempdb ใหม่ถ้า tempdb เดิมเสียหาย ทุกครั้งที่ user สร้างคาด้าเบสขึ้นมาใหม่ ระบบ SQL Server จะก๊อปปี้มาจากคาด้าเบส model ทั้งนี้ เพราะในคาด้าเบส model มี system table ที่จำเป็นต้องมีในแต่ละคาด้าเบส system table นี้จะเก็บออปชันของคาด้าเบส ,ค่า default สิทธิในการใช้ข้อมูลของ user และ rule ต่าง ๆ ของคาด้าเบส
- **Msdb** เป็นคาด้าเบสที่สนับสนุนเซอร์วิสของ SQL ServerAgent โดยเก็บ รายละเอียดของงานที่ต้องทำตามตารางเวลา (Schedule), การแจ้งเตือน (Alerts), Jobs และ Operations
- **Tempdb** เป็นคาด้าเบสที่เก็บข้อมูลชั่วคราวในการทำงานของ SQL Server เช่น เป็น เนื้อที่สำหรับการเรียงลำดับข้อมูล, เนื้อที่สำรองชั่วคราวสำหรับ temporary table หรือ temporary storeprocedure ซึ่ง user ทุกคนมีสิทธิ์ที่จะใช้ได้ ทุกครั้งเมื่อมีการ start service ของ MSSQLServer จะมีการสร้าง tempdb ใหม่ทุกครั้ง ดังนั้นข้อมูลใน tempdb จะมีการเคลียร์ตลอด

**User Database** คือ คาด้าเบสที่ user สร้างขึ้นเพื่อจัดเก็บข้อมูลระบบงานต่าง ๆ ในการ สร้างคาด้าเบสทุกครั้งจะใช้ template จาก คาด้าเบส model นั้น คือขนาดคาด้าเบสเท่ากับค่า default ของคาด้าเบส model ด้วย ยกเว้นจะกำหนดเป็นอย่างอื่น นอกจากนี้เมื่อมีการติดตั้ง โปรแกรม SQL Server ที่เครื่องเซิร์ฟเวอร์จะมีการสร้าง user database ให้ด้วยคือ Northwind และ pubs

## 2.3 Java

ภาษา Java ถูกออกแบบขึ้น โดยมีจุดมุ่งหมายดังนี้

1. เป็นภาษาที่ง่าย (Simple) หมายถึงมีกลไกของภาษาจำนวนไม่มาก และไม่ซับซ้อนเพื่อให้ ผู้ใช้สามารถเรียนรู้ได้อย่างรวดเร็ว ภาษา Java นำไวยากรณ์ของภาษาส่วนใหญ่มาจากภาษา C++ เพื่อให้ผู้ที่คุ้นเคยกับภาษา C หรือ C++ อยู่แล้วไม่ต้องเสียเวลาศึกษาไวยากรณ์ส่วนใหญ่ของ Java แม้ว่าภาษา Java จะนำไวยากรณ์มาจากภาษา C++ แต่ก็คัดกลไกของภาษา C++ ออกไปหลายอย่าง

โดยไปเพิ่มความสามารถให้คอมไพเลอร์จัดการให้แทน เช่น ภาษา Java ไม่มี preprocessor commands, function prototype และ header files รวมทั้งกลไกที่จะทำให้ภาษายุ่งยากเกินไป อย่างเช่น default arguments และ operator overloading ก็ถูกตัดทิ้งไปด้วย และในเมื่อ Java เป็นภาษาเชิงวัตถุอย่างรอบคอบกว่าภาษา C++ ดังนั้นกลไกที่อาจทำให้เกิดความกำกวมอย่างเช่น multiple inheritance และกลไกที่อาจทำลายแบบแผนการเขียนโปรแกรมเชิงวัตถุที่ดี เช่น friend methods และ goto statement ก็ถูกตัดทิ้งด้วย

หากเป็นภาษาอื่น การสร้างโปรแกรมที่เกี่ยวกับ Graphical User Interfaces, multitasking, network programming และ plug-in programs ผู้โปรแกรมจะต้องมีความรู้ระดับเชี่ยวชาญจึงจะทำงานได้ หรือไม่ก็ต้องอาศัยโปรแกรมประเภท wizard อย่างใน Visual C++ สำหรับช่วยสร้างโปรแกรม เนื่องจากรายละเอียดของการสร้างโปรแกรมประเภทนี้มีมากเกินกว่าจะศึกษาให้สามารถใช้งานได้ในเวลาอันสั้น แต่ภาษา Java ประสบความสำเร็จอย่างมากในการนำเทคนิคของการโปรแกรมเชิงวัตถุมาใช้ เพื่อช่วยให้การสร้างโปรแกรมประเภทนี้เป็นไปโดยง่าย

2. โปรแกรมที่ถูกสร้างขึ้นด้วยภาษา Java จะต้องมีความผิดพลาดที่ไม่คาดคิดมาก่อนน้อยที่สุด และโปรแกรมจะต้องไม่ล้มเหลวลงง่ายๆ ด้วยความผิดพลาดเพียงเล็กน้อย คุณสมบัติอย่างนี้เรียกว่าความคงทน (robust) ภาษา Java เป็นภาษาที่เน้นการใช้กลไก exception handling ซึ่งจะช่วยให้โปรแกรมสามารถจัดการกับความผิดพลาดบางอย่างที่อาจเกิดขึ้นในขณะที่โปรแกรมทำงาน โดยโปรแกรมไม่ต้องล้มเหลวและหยุดลง อย่างไรก็ตามภาษา Java ไม่สามารถเป็นภาษาที่คงทนได้ด้วยการมี exception handling เพียงอย่างเดียว ดังนั้นในการออกแบบภาษาจึงตัดกลไกของภาษาบางอย่างที่อาจจะทำให้เกิดความผิดพลาด (หากใช้อย่างไม่ระวัง) ออกไปและเพิ่มการตรวจสอบว่าการทำงานบางอย่างจะก่อให้เกิดความผิดพลาดขึ้นได้หรือไม่

ภาษา Java ตัดการอ้างถึง address ของตัวแปร และการใช้ pointers ในการอ่านหรือเขียนค่าหน่วยความจำโดยตรงออกไปจากภาษา เพื่อให้โปรแกรมไม่มีโอกาสใช้ pointers อ้างถึงหน่วยความจำที่โปรแกรมไม่มีสิทธิ์อ้าง การสร้างโปรแกรมลักษณะนี้จึงเป็นไปได้ นอกจากนี้ ภาษา Java ยังกำหนดให้ผู้โปรแกรมไม่สามารถควบคุมการคืนหน่วยความจำ (memory reallocation) ในขณะที่โปรแกรมทำงาน แต่อาศัย automatic garbage collector ทำหน้าที่เก็บหน่วยความจำที่ไม่อาจอ้างถึงได้แล้ว และนำกลับมาใช้งานได้อีกโดยอัตโนมัติ โปรแกรมจึงไม่มีโอกาสอ้างถึงค่าในหน่วยความจำที่ถูกทำลายไปแล้ว

ภาษา Java ถูกจัดว่าเป็น strongly typed language ซึ่งหมายถึง ภาษาที่เน้นความถูกต้องของชนิดข้อมูล (type) ที่ใช้ในโปรแกรม คอมไพเลอร์ของภาษาประเภทนี้จะทำการตรวจสอบการจัดการกับชนิดข้อมูลของตัวแปรว่าถูกต้องหรือไม่ ซึ่งเรียกว่า type checking ความผิดพลาดเกี่ยวกับ

ชนิดข้อมูลจะถูกปฏิเสธตั้งแต่ตอนคอมไพล์โปรแกรม ทำให้ไม่มีความผิดพลาดเกี่ยวกับชนิดข้อมูลเกิดขึ้นในระหว่างที่โปรแกรมทำงาน ถือเป็นภาษาที่ปลอดภัยในระดับหนึ่ง แต่ก็ยังเป็นภาษาที่ไม่มี ความยืดหยุ่นเกี่ยวกับชนิดข้อมูล ภาษา Java มีคอมไพเลอร์ทำ type checking อย่างเคร่งครัด และยังมี การตรวจสอบอีกหลายอย่างในระหว่างที่โปรแกรมทำงานอีกด้วย เช่นตรวจสอบว่าการเปลี่ยน ชนิดข้อมูล (casting) ว่าถูกต้องหรือไม่ และการอ้างถึงสมาชิกใน array หรือ string อยู่ในขอบเขตที่ ถูกต้องหรือไม่

3. โปรแกรมที่สร้างขึ้นด้วยภาษา Java มักจะถูกส่งผ่านระบบเครือข่ายไปทำงานในเครื่อง คอมพิวเตอร์ของผู้อื่น ดังนั้นภาษา Java ต้องมีหลักประกันแก่ผู้รับโปรแกรมไปทำงานว่า มันจะไม่ ก่อให้เกิดอันตรายต่อเครื่องหรือระบบข้อมูลของผู้ใช้ ในการออกแบบภาษา Java จึงมีข้อกำหนด หลายอย่างที่จะทำให้ไม่สามารถสร้างโปรแกรมเพื่อส่งไปอันตราย (หรือสิ่งที่ไม่สมควรทำ) ต่อ เครื่องที่รับโปรแกรมนั้นไปทำงาน คุณสมบัติอย่างนี้เรียกว่า (secure) อย่างไรก็ตามไม่มีภาษาใดที่ ปลอดภัยร้อยเปอร์เซ็นต์ ภาษา Java มีความปลอดภัยในระดับสูงเพราะถูกออกแบบมาเพื่อความ ปลอดภัยมากกว่าภาษาอื่น โดยแบ่งการป้องกันออกเป็นหลายระดับดังนี้

- ที่ระดับล่าง ภาษา Java ไม่อนุญาตให้อ้างถึงค่าในหน่วยความจำโดยตรงผ่านทาง pointers และทำการตรวจสอบว่าการอ้างถึงสมาชิกใน array อยู่ในขอบเขตของ array นั้น หรือไม่ ทำให้ไม่สามารถเขียนหรืออ่านค่าในหน่วยความจำที่โปรแกรมนั้นไม่มีสิทธิ์ ดังนั้นการพยายามเปลี่ยนแปลงโปรแกรมหรือค่าในหน่วยความจำเพื่อสร้างโปรแกรมที่จะ เป็นอันตรายต่อคนอื่นจึงเป็นไปได้
- ที่ระดับถัดขึ้นมา ในที่โปรแกรมทำงานจะมีโปรแกรมที่เรียกว่า byte-code verifier ทำหน้าที่ตรวจสอบโปรแกรมที่จะถูกทำงานว่ามีรหัสคำสั่งที่ผิดปกติหรือมีการกระทำใดที่ ไม่สมควรหรือไม่ หากพบการกระทำใดที่จะก่อให้เกิดอันตรายต่อระบบก็จะปฏิเสธที่จะ ทำงานโปรแกรมนั้น
- ที่ระดับบน มีระบบรักษาความปลอดภัยที่เรียกว่า sandbox model นั่นคือ โปรแกรมที่ถูก download มาจากเครื่องอื่นจะถูกจัดว่าเป็นโปรแกรมที่ไม่น่าไว้วางใจ (untrusted codes) และถูกเก็บอยู่ใน sandbox โปรแกรมที่อยู่ใน sandbox นี้จะมีข้อจำกัดใน การทำงานหลายอย่างซึ่งถูกควบคุมโดย Security Manager เช่น ไม่สามารถอ่านหรือเขียน ไฟล์ได้ เป็นต้น

อีกจุดมุ่งหมายหนึ่งของการออกแบบภาษา Java คือการทำให้โปรแกรมที่สร้างขึ้นถูกส่งไป ทำงานบนเครื่องต่างชนิดได้ ซึ่งถือเป็นเรื่องสำคัญที่สุดจึงขอแยกกล่าวดังต่อไปนี้

### 2.3.1 Compilation and Interpretation

ภาษาระดับสูงจะต้องใช้ตัวแปลภาษา (Language translator) เพื่อเปลี่ยนโปรแกรมต้นฉบับ (Source code) ให้เป็นโปรแกรมของคำสั่งที่หน่วยประมวลผลทำงานได้ (machine code) โดยทั่วไปวิธีแปลภาษาจะถูกแบ่งออกเป็นสองแบบคือ

1. **Compilation** ตัวแปลภาษา ในกรณีนี้เรียกว่า คอมไพเลอร์ มีหน้าที่วิเคราะห์โปรแกรม source code แล้วสร้างโปรแกรม machine code ออกมาเป็นผลลัพธ์ การกระทำดังกล่าวเรียกว่า การคอมไพล์โปรแกรม ข้อดีของวิธีนี้คือ โปรแกรม machine code ทำงานเร็วมาก เนื่องจากขั้นตอนในการแปลภาษาถูกแยกออกไปทำก่อนโปรแกรมทำงาน และคอมไพเลอร์ก็สามารถวิเคราะห์โปรแกรมได้ทั้งโปรแกรมเสียก่อน จึงสร้าง machine code ที่มีประสิทธิภาพในการทำงาน และโปรแกรมมีขนาดเล็ก ตัวอย่างของภาษาที่ใช้การแปลภาษาแบบ compilation คือ Fortran, Algol, Pascal, C และ C++
2. **Interpretation** ตัวแปลภาษา ในกรณีนี้เรียกว่า interpreter มีหน้าที่อ่านโปรแกรม source code ทีละบรรทัด แล้วแปลโปรแกรมบรรทัดนั้นเป็น machine code และทำงานทันที ต่อจากนั้นก็อ่าน โปรแกรม source code บรรทัดต่อไปเข้ามาทำเช่นเดิม จนกว่าโปรแกรมจะหยุดหรือจบโปรแกรม การกระทำดังกล่าวเรียกว่า การ interpret วิธีนี้จะมีทั้งการแปลภาษา และโปรแกรมทำงาน เกิดขึ้นสลับกันไป โปรแกรมจึงทำงานช้ากว่าวิธีคอมไพล์ แต่ก็มีข้อดีคือ interpreter ถูกสร้างขึ้นได้ง่ายและมีขนาดเล็กกว่า เพราะการวิเคราะห์และแปลโปรแกรมทีละบรรทัดทำได้ง่ายกว่าแปลทั้งโปรแกรม และการแปลภาษาระหว่างโปรแกรมทำงานจะทำให้ได้ภาษาที่มีความยืดหยุ่นในการเขียนโปรแกรมมากกว่า ตัวอย่างของภาษาที่ใช้การแปลภาษาแบบ interpretation คือ Lisp, Basic, Prolog และ Smalltalk

### 2.3.2 Java's Virtual Machine

ภาษา Java นำความคิดในการสร้างเครื่องจักรสมมติมาใช้ เพื่อให้ภาษามีคุณสมบัติ platform independent แต่แยกการแปลภาษาออกไปโดยใช้คอมไพเลอร์ แล้วนำโปรแกรมของเครื่องจักรสมมติที่ได้มาทำงานโดย interpreter วิธีการนี้จะทำให้โปรแกรมทำงานได้เร็วขึ้นกว่าการใช้ interpreter เพียงอย่างเดียว

ไฟล์ของโปรแกรม Source code ภาษา Java จะต้องมี extension เป็น .java คอมไพเลอร์ของภาษา Java จะทำหน้าที่คอมไพล์โปรแกรมภาษา Java ไปเป็นโปรแกรมที่ประกอบด้วยคำสั่งของ Java Virtual Machine (JVM) และจะถูกเก็บในไฟล์ที่มี extension เป็น .class เรียกว่าไฟล์ Java class

ด้วยเหตุที่โปรแกรม Java class ต้องถูกทำงานด้วย JVM ดังนั้นบางที่เราจึงเรียก JVM ว่า Java runtime system

ภายใน JVM มีหน่วยประมวลผลสมมติที่เรียกว่า virtual processor ทำหน้าที่ประมวลผลคำสั่งของ JVM โดยทั่วไป virtual processor ของ JVM ที่จำลองขึ้นบนคอมพิวเตอร์เครื่องหนึ่ง จะแปลคำสั่งของ JVM เป็นคำสั่งของหน่วยประมวลผลในคอมพิวเตอร์เครื่องนั้น เรียกว่า native code แล้วให้หน่วยประมวลผลนั้นทำงานคำสั่งนั้น คำสั่ง (opcode) ของ JVM มีขนาด 1 byte ทุกคำสั่ง (บางครั้งเราจึงเรียกโปรแกรม Java class ว่าโปรแกรม byte code) จำนวนคำสั่งของ JVM จึงมีได้สูงสุดเพียง 256 คำสั่ง เทียบกับหน่วยประมวลผลทั่วไปแล้ว คล้ายกับว่าคำสั่งของ JVM มีมากมายอย่างยิ่ง แต่จริงๆ แล้วคำสั่งของ JVM แบ่งออกได้เป็นไม่กี่ประเภท โดยที่แต่ละประเภทนั้นจะทำหน้าที่คล้ายๆ กัน เพียงแต่ทำกับ operands ต่างชนิดข้อมูลกันเท่านั้น

ชุดคำสั่งของ JVM ถูกออกแบบมาเพื่อสนับสนุนการทำงานของโปรแกรมเชิงวัตถุ จึงมีคำสั่งเกี่ยวกับการสร้าง instances และการอ้างถึงสมาชิกใน instance ซึ่งไม่มีในหน่วยประมวลผลทั่วไป ภาษา Java เป็นภาษาที่เน้นความถูกต้องเกี่ยวกับชนิดข้อมูล (type) จึงมีคำสั่งสำหรับคำนวณชนิดข้อมูลพื้นฐานแต่ละชนิด เช่น คำสั่ง iadd สำหรับบวกเลขจำนวนเต็ม integer และคำสั่ง dadd สำหรับบวกเลขทศนิยม double เป็นต้น

JVM ถูกออกแบบให้สามารถจำลองได้บนเครื่องทั่วไป ไม่ว่าจะใช้หน่วยประมวลผลของบริษัทใด แต่หน่วยประมวลผลต่างรุ่น ต่างบริษัทมีจำนวน registers ไม่เท่ากัน บางรุ่นมี registers หน้าที่พิเศษที่รุ่นอื่นไม่มี ผู้ออกแบบจึงตัดปัญหานี้ โดยให้ JVM ไม่มี register และทำการคำนวณทั้งหมดบน stacks ชุดคำสั่งของ JVM จึงเป็น stacked operations หรือกล่าวได้ว่า JVM เป็น stack machine

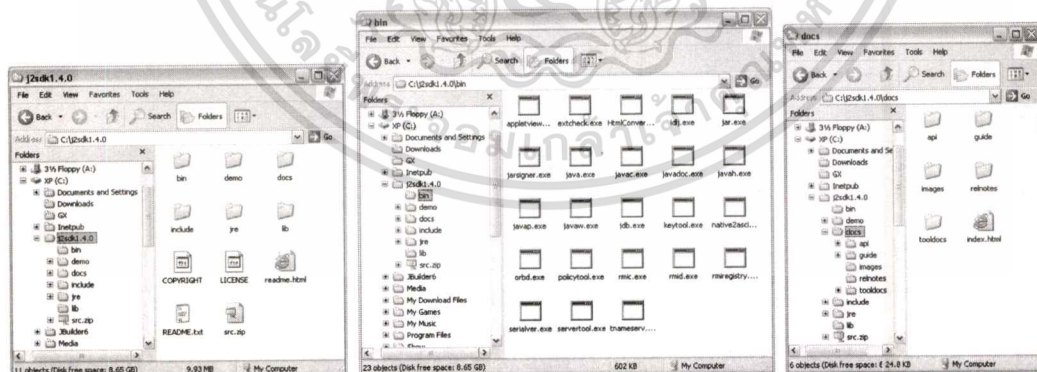
Virtual processor ใน JVM จะทำการ mapping จากคำสั่ง byte code ในไฟล์ Java class ไปเป็นโปรแกรมของ native code ที่ทำหน้าที่ของคำสั่งนั้น (โดยอาจใช้ hash table หรือสร้างขึ้นโดยประโยค switch ในภาษา C) แล้วส่ง native code นั้นให้หน่วยประมวลผลทำงาน สังเกตว่า native code นั้นอาจเป็น Application Program Interfaces (API) ของระบบปฏิบัติการที่ใช้ หรืออาจจะเป็น standard classes ที่สร้างขึ้นสำหรับหน่วยประมวลผลนั้น ทำให้ JVM หนึ่งอาจใช้งานได้ในระบบต่างๆ โดยเปลี่ยนแปลงแค่ standard classes เท่านั้น

### 2.3.3 Java Developer Kit (JDK)

ในการสร้างโปรแกรมภาษา Java อย่างน้อยต้องมี คอมไพเลอร์ และ interpreter ของภาษา Java ซึ่งปัจจุบันมีหลายบริษัทผลิตชุดโปรแกรมสำหรับพัฒนาโปรแกรมภาษา Java ออกมาเช่น

- Java Developer Kit (JDK หรือ J2SE, J2ME, J2EE) และ Forte ของบริษัท Sun
- Visual Cafe ของบริษัท Symantec
- J Builder ของบริษัท Borland
- Visual Age for Java ของบริษัท IBM
- JCreator ของบริษัท Xinox Software

ในการพัฒนาโครงการครั้งนี้ เลือกใช้ JCreator และ JBuilder 6 ของ บริษัท Borland ในการพัฒนา ซึ่ง JCreator จะต้องใช้งานร่วมกับ JDK ของบริษัท Sun ด้วย และใช้ JDK หรือเรียกอีกอย่างว่า J2SDK (Java 2 Software Developer Kit) โดยใช้ J2SDK เวอร์ชัน 1.4 ซึ่งสามารถ download J2SDK ได้จากตามเว็บไซต์ หรือจากเว็บไซต์ของบริษัท Sun โดยตรงที่ <http://java.sun.com/j2se/1.4/> นอกจากนี้ ควรติดตั้ง J2SDK Documents ด้วย ซึ่งมีความจำเป็นและสำคัญต่อการพัฒนาโปรแกรม Java ซึ่ง J2SDK Documents จะช่วยให้นักพัฒนาเข้าใจความหมายและการใช้คำสั่งต่างๆ การพัฒนาโปรแกรม Java ซึ่งโดยปรกติการลง JDK นั้นจะติดตั้งโปรแกรมและไฟล์ลงในไดเรกทอรีที่เราระบุ ซึ่งไฟล์ที่ Download มาจากเว็บไซต์ของบริษัท Sun เราสามารถที่จะเลือกชนิดของ JDK ที่จะติดตั้งลงบน OS อะไร เช่น จะติดตั้ง Java บนเครื่องที่มีระบบปฏิบัติการเป็น Windows 9X ก็ให้เราเลือก Java ที่ใช้สำหรับ Windows ซึ่งการทำโครงการในครั้งนี้ ได้พัฒนาบน Windows XP ดังนั้นจึงใช้ JDK สำหรับ Windows และได้ติดตั้งอยู่ภายใต้ไดเรกทอรี C:\j2sdk1.4.0 ดังภาพ



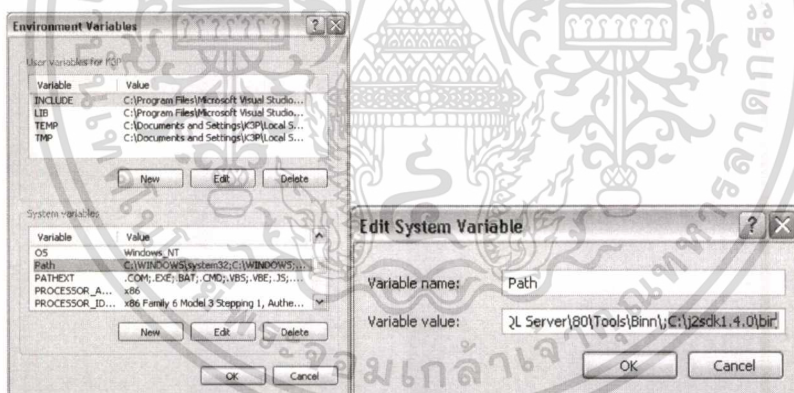
ภาพที่ 2.9 แสดงโครงสร้าง java ไดเรกทอรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่าโปรแกรมและไฟล์ถูกติดตั้งลงไปจำนวนมาก ดังนั้นเพื่อให้ยาวเกินไป จึงขอกล่าวเฉพาะโปรแกรมที่จำเป็นในการสร้างโปรแกรมภาษา Java ขั้นพื้นฐาน ซึ่งในไดเรกทอรี C:\j2sdk1.4.0\bin มีโปรแกรมสำหรับพัฒนาโปรแกรมภาษา Java จำนวนมาก ที่จำเป็นในเบื้องต้นคือ

- javac.exe เป็นโปรแกรม Java Compiler
- java.exe เป็นโปรแกรม Java Interpreter

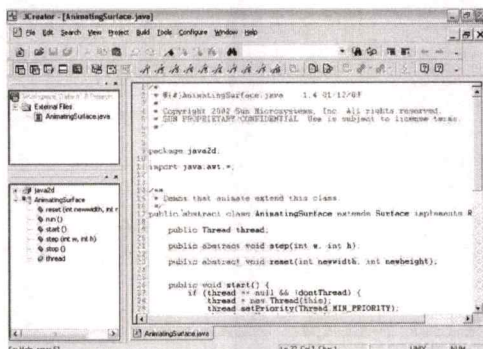
นอกจากนี้ยังได้ทำการติดตั้ง Java Documents ไว้ที่ C:\j2sdk1.4.0\docs ซึ่งตัว Java Documents นี้เราจำเป็นต้อง download เพิ่มเติมจากชุด j2sdk ซึ่งตัว j2sdk นั้นจะไม่มี Documents มาให้ เมื่อติดตั้ง J2SDK แล้ว ตัว javac.exe และ java.exe จะถูกเรียกใช้ได้จากไดเรกทอรี C:\j2sdk1.4.0\bin เท่านั้น ซึ่งโดยปรกติแล้ว เราจะไม่พัฒนาโปรแกรมในไดเรกทอรีที่เก็บคอมพิวเตอร์ ดังนั้นเพื่อให้สามารถเรียกใช้ javac.exe และ java.exe ได้จากไดเรกทอรีใดๆ จึงตั้งค่า directory path ไปที่ C:\j2sdk1.4.0\bin โดยการเพิ่มประโยค C:\j2sdk1.4.0\bin เข้าไปใน Environment Variables ของ Windows XP ดังภาพ



ภาพที่ 2.10 แสดงการ Setup Java Environments

ส่วน Software ที่ใช้ช่วยในการพัฒนาโครงการอื่นอีกตัวหนึ่งคือ JCreator กับ JBuilder6 ในส่วนของ JCreator นั้นเอาไว้ใช้พัฒนาในการเขียน Class ต่างๆที่ใช้ในโครงการครั้งนี้ ส่วน JBuilder6 นั้นใช้สำหรับพัฒนา GUI (Graphic User Interface) ต่างๆ เนื่องจาก JBuilder มีลักษณะคล้าย Visual Basic ของบริษัท Microsoft จึงทำให้การพัฒนาในส่วนของ GUI นั้นทำได้ง่ายและรวดเร็วกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.11 แสดงโปรแกรม JCreator

### 2.3.4 JDBC

Java DataBase Connectivity หรือ JDBC คือมิดเดิลแวร์ที่ถูกใช้บนเทคโนโลยีจาวา ในการเข้าถึงระบบฐานข้อมูล โดยสามารถเรียกใช้งานข้อมูล หรือจัดการระบบฐานข้อมูลผ่าน คำสั่ง SQL-92 หรือภาษา SQL มาตรฐาน มาทำความเข้าใจกันดูว่า หากจะใช้งาน JDBC แล้วจะต้องรู้จักกับอะไรบ้าง และมีวิธีการใช้งานอย่างไร

#### 2.3.4.1 ประเภทของ JDBC

JDBC ซึ่งเป็นมิดเดิลแวร์ตัวหนึ่ง มีรูปแบบและชื่อเรียกที่แบ่งออกเป็น 4 แบบด้วยกันคือ Type 1, Type 2, Type 3 และ Type 4 โดยเป็นข้อกำหนดที่ถูกกำหนดมาโดยบริษัทซัน ไมโครซิสเต็มส์ ซึ่งเป็นผู้กำหนดรายละเอียดของแต่ละประเภท ดังนี้

##### ประเภท 1 (Type1: JDBC-ODBC Bridge)

เรียกกันสั้นๆว่า JDBC-ODBC บริดจ์ หมายถึง JDBC ที่ทำงานอยู่บนชั้นการสื่อสารมิดเดิลแวร์ที่ชื่อ ODBC อีกทีหนึ่ง เนื่องจากการทำงานของคอมพิวเตอร์บนพื้นฐานของระบบปฏิบัติการไมโครซอฟต์วินโดวส์ มีมิดเดิลแวร์ที่ชื่อ ODBC เป็นมาตรฐานในการสื่อสารอยู่แล้ว และเนื่องมาจากระบบที่พัฒนาให้ใช้งานบนวินโดวส์ส่วนใหญ่ก็มักจะถูกพัฒนาโดยอาศัย ODBC มาช่วยในการทำงานอยู่ด้วย ดังนั้นในการสร้างโปรแกรมด้วยภาษาจาวาให้สามารถทำงานอยู่บนพื้นฐาน ODBC เดิม เพื่อหลีกเลี่ยงปัญหาอันเกิดจากความยุ่งยากในการติดตั้ง และต้องแปลงระบบที่ใช้อยู่ไปเป็นระบบใหม่ทั้งหมด

ดังนั้นเมื่อมีการพัฒนาโปรแกรมด้วยภาษาจาวา โดยมีการใช้งานระบบเดิมที่

เรียกใช้ ODBC อยู่แล้ว จึงนิยมที่จะพัฒนาโปรแกรมในรูปแบบการใช้งาน JDBC ประเภทที่ 1 รายละเอียดของการเขียนโปรแกรมผู้เขียนจะยกไปไว้ในส่วนการเขียนโปรแกรมโดยเฉพาะ ส่วนนี้ขอล่าวถึงในเชิงทฤษฎีก่อน

#### ลักษณะของการนำไปใช้งาน

- ถูกใช้สำหรับการศึกษาวิธีการเขียนโปรแกรมจากา เพื่อติดต่อกับ JDBC (ซึ่งในหนังสือจะมีการอ้างถึง JDBC ประเภทนี้ในตัวอย่าง)
- เหมาะกับระบบงานที่มี ODBC ในการใช้งานอยู่แล้ว
- สำหรับระบบงานที่ทั้งหมดทำงานอยู่บนพื้นฐานของแพลตฟอร์มวินโดวส์

#### ข้อดีของการนำไปใช้งาน

- ไม่เหมาะกับการนำไปใช้งานบนระบบงานที่มีขนาดใหญ่ๆ เนื่องจากทำให้เกิดความซ้ำในการทำงาน และประสิทธิภาพในการทำงานที่ไม่ดี
- มีข้อมูลในส่วน โอเวอร์เฮด(Overhead) สูง เนื่องจากต้องมีส่วนในการติดต่อกันระหว่าง JDBC และ ODBC เพิ่มเติม
- ไม่สนับสนุนความสามารถทั้งหมดของมาตรฐาน JDBC เนื่องจากข้อจำกัดของ ODBC ที่มีรูปแบบการทำงานน้อยกว่า JDBC

#### ประเภท 2 (Type2: Partial Java Driver)

สำหรับ JDBC ประเภทนี้ ตัวไดรเวอร์ (มิดเดิลแวร์ของแต่ละผลิตภัณฑ์มักจะใช้คำว่าไดรเวอร์แทนมิดเดิลแวร์ตัวนั้นๆ) จะถูกโหลดไว้ที่ไคลเอนท์หรือที่เียร์ส่วนติดต่อกับผู้ใช้ ดังนั้นรูปแบบในการสร้างโปรแกรมไคลเอนท์จะต้องพยายามโหลดไดรเวอร์ของผลิตภัณฑ์เซิร์ฟเวอร์ที่ต้องการติดต่อกับ เช่น ถ้ามีระบบฐานข้อมูล Oracle โปรแกรมในส่วนไคลเอนท์ต้องสร้างคำสั่งในการโหลดไดรเวอร์ Oracle ขึ้นใช้งาน

#### ลักษณะของการนำไปใช้งาน

- มีประสิทธิภาพในการดีกว่าประเภท 1 เมื่อเปรียบเทียบกัน
- คำสั่งในการติดต่อกับเซิร์ฟเวอร์ จะเป็นคำสั่งที่ดีที่สุดสำหรับเซิร์ฟเวอร์นั้นๆ โดยเฉพาะ ทำการทำงานโดยรวมดีกว่า

### ข้อดีของการนำไปใช้งาน

- ผู้ใช้โปรแกรมในส่วนไคลเอนท์ยังต้องการไดรเวอร์ สำหรับเซิร์ฟเวอร์โดยเฉพาะ
- เมื่อมีการเปลี่ยนแปลงเซิร์ฟเวอร์เป็นผลิตภัณฑ์ตัวอื่น โปรแกรมในส่วนไคลเอนท์ต้องมีการเปลี่ยนแปลงและคอมไพล์ใหม่เสมอ

### ประเภท3 (Type 3: Pure Java Driver)

ในการใช้งานสถาปัตยกรรมแบบหลาย-tier ที่จำเป็นต้องมีแอปพลิเคชันเซิร์ฟเวอร์ทำงานร่วมอยู่ในระบบ ดังที่ผู้อ่านได้ศึกษามาแล้ว แอปพลิเคชันเซิร์ฟเวอร์จะมาลดข้อยุ่งยากในการติดตั้งหรือดูแลระบบในส่วน-tier ไคลเอนท์ ดังนั้นลักษณะของ JDBC ในประเภทที่ 3 คือการนำเอา JDBC ไดรเวอร์ไปวางไว้ที่ตัวแอปพลิเคชันเซิร์ฟเวอร์ เพื่อเป็นตัวเชื่อมโยงการสื่อสารไปหาเซิร์ฟเวอร์

### ลักษณะของการนำไปใช้งาน

- เป็นระบบที่มีประสิทธิภาพในการทำงานที่ดีกว่าประเภทที่ 1 และ 2 เมื่อเปรียบเทียบกัน
- เหมาะสำหรับองค์กรที่มีเซิร์ฟเวอร์ทางด้านฐานข้อมูลที่หลากหลายรูปแบบ
- การทำงานของไคลเอนท์ไม่จำเป็นต้องติดตั้ง JDBC ไดรเวอร์ไว้ในทุกตัว
- การติดตั้งและการดูแลระบบไคลเอนท์ทำได้ง่าย และสะดวก

### ข้อดีของการนำไปใช้งาน

- ยังต้องการไดรเวอร์สำหรับแต่ละผลิตภัณฑ์ เพื่อติดตั้งไว้ที่แอปพลิเคชันเซิร์ฟเวอร์

### ประเภท 4 (Type 4: Direct-to-DB)

ลักษณะการทำงานของ JDBC ในประเภทที่ 4 มีการทำงาน โดยการส่งคำสั่ง SQL จากไคลเอนท์ไปที่เซิร์ฟเวอร์ฐานข้อมูลโดยตรงผ่านระบบเครือข่าย โดยที่รูปแบบคำสั่ง SQL จะต้องเป็นไปตามข้อกำหนดของระบบฐานข้อมูลแต่ละตัวที่เลือกใช้อยู่ ดังนั้นจะเห็นว่าการโต้ตอบของเซิร์ฟเวอร์กับคำสั่ง SQL สามารถทำงานได้รวดเร็ว ทำให้ JDBC ประเภทนี้มีรูปแบบการทำงานที่รวดเร็ว แต่ก็จำเป็นต้องสร้างไคลเอนท์สำหรับฐานข้อมูลแต่ละระบบซึ่งในบางครั้งไคลเอนท์สำหรับฐานข้อมูลตัวหนึ่งจึงไม่สามารถนำไปใช้กับระบบฐานข้อมูลตัวอื่นๆได้

### ลักษณะของการนำไปใช้งาน

- มีประสิทธิภาพในการทำงานดีที่สุด เมื่อเทียบกับทุกประเภทที่กล่าวมา

ข้อดีของการนำไปใช้งาน

- มีความยุ่งยากในการพัฒนาโปรแกรมเพื่อทำงานในฝั่งไคลเอนท์ เพราะผู้พัฒนาต้องเรียนรู้การทำงานของเซิร์ฟเวอร์แต่ละตัวก่อน

#### 2.3.4.2 การใช้งาน JDBC

อย่างที่ทราบกันว่าการเข้าใช้งานฐานข้อมูลด้วยภาษาจาวา มีรูปแบบอยู่ด้วยกันถึง 4 ชนิด การที่โปรแกรมเมอร์จะเลือกใช้วิธีใดก็ขึ้นอยู่กับความเหมาะสมของระบบว่าระบบถูกออกแบบมาให้มีลักษณะการทำงานแบบใด แต่ถึงแม้ว่าจะได้ข้อสรุปอย่างไรก็ตาม วิธีการลงมือเขียน โปรแกรมก็มีขั้นตอนที่คล้ายกัน ดังนี้

- ขั้นตอนเริ่มติดต่อกับฐานข้อมูล (Establish Database)
- ขั้นตอนการส่งคำสั่ง SQL ให้กับฐานข้อมูล (Send Query)
- ขั้นตอนการแสดงผลลัพธ์สู่ผู้ใช้ (Display Result)
- ขั้นตอนการปิดการติดต่อกับฐานข้อมูล (Close Connection)

ขั้นตอนที่ 1 เริ่มติดต่อกับฐานข้อมูล (Establish Database)

แบ่งออกเป็น 2 ขั้นตอนย่อยๆดังนี้ คือ

A) ขั้นตอนการโหลดไดรเวอร์เพื่อใช้งาน

รูปแบบ

```
Class.forName ("Driver_Name")
```

คลาสคักจับข้อผิดพลาด

ClassNotFoundException

เมื่อ

Driver\_Name คือชื่อไดรเวอร์สำหรับใช้งานระบบฐานข้อมูล ที่ต้องการโหลด

ขึ้นมาใช้งาน ซึ่งจะต้องระบุชื่อให้ตรงกับระบบฐานข้อมูลที่ต้องการติดต่อ

กรณีติดต่อกับไดรเวอร์ JDBC-ODBC Bridge ไดรเวอร์ชื่อ

```
"sun.jdbc.odbc.JdbcOdbcDriver"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีติดต่อกับไดร์เวอร์ MySQL ไดร์เวอร์ชื่อ

"org.gjt.mm.mysql.Driver"

ตัวอย่าง

```
try {
    Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");
} catch (ClassNotFoundException e) {
    System.err.print (e.getMessage ());
}
```

## B) ขั้นตอนการติดต่อไปยังระบบฐานข้อมูลปลายทาง

รูปแบบ

Connection Instance\_Name = DriverManager.getConnection (URL, Login, Password)

คลาสที่จับข้อผิดพลาด

SQLException

รูปแบบ URL

jdbc:<subprotocol>:<subname>

เมื่อ

ระบบปลายทาง <subprotocol> คือ โพรโทคอลการสื่อสารข้อมูล เพื่อแจ้งไปยังเครื่องหรือ ซึ่งมีข้อมูลแตกต่างกันไปตามรูปแบบการสื่อสาร เช่น

odbc สำหรับการสื่อสารผ่านระบบ ODBC

dbnet สำหรับการสื่อสารผ่านโพรโทคอล TCP/IP

<subname> คือฐานข้อมูลภายในระบบที่อ้างผ่าน subprotocol หรือ

แหล่งข้อมูล

ตัวอย่าง

```
try {
    String URL = "jdbc:odbc:myData";
    Connection con =
    DriverManager.getConnection(URL,"myLogin","myPassword");
} catch (SQLException e) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
System.err.print (e.getMessage ());
```

```
}
```

## ขั้นตอนที่ 2 การส่งคำสั่ง SQL

เมธอดที่ถูกกำหนดไว้สำหรับส่งคำสั่ง SQL ไปให้กับฐานข้อมูล ถูกแบ่งตามรูปแบบคำสั่ง SQL ได้ 2 แบบ โดยผู้ใช้ต้องเลือกเมธอดที่เหมาะสมกับคำสั่ง SQL ที่ใช้งานคือ

หากคำสั่ง SQL เป็นประเภทดำเนินการ ซึ่งสร้างด้วยคีย์เวิร์ดจำพวก CREATE, ALTER, DROP, INSERT, UPDATE, DELETE, ? การทำงานของคำสั่งลักษณะนี้ เป็นการส่งคำสั่งไปยังฐานข้อมูล เพื่อให้เกิดการเปลี่ยนแปลงใดๆในระบบฐานข้อมูล ดังนั้นผู้ใช้คำสั่งเพียงต้องการทราบแล้วว่าทำงานสำเร็จหรือไม่เท่านั้น การเรียกใช้คำสั่งเหล่านี้ให้ทำงานจึงต้องใช้งานผ่าน เมธอด ExecuteUpdate

หากคำสั่ง SQL เป็นประเภทสืบค้นข้อมูล ซึ่งสร้างด้วยคีย์เวิร์ด SELECT เนื่องจากคำสั่งประเภทสืบค้นข้อมูลมีลักษณะพิเศษที่เมื่อระบบฐานข้อมูลได้รับคำสั่ง SQL แล้วจะต้องส่งข้อมูลผลลัพธ์ที่ต้องการกลับมา ในรูปของกลุ่มข้อมูลเรกอร์ด (Record Set) การเรียกใช้งานคำสั่ง SQL ประเภทนี้จึงต้องเรียกใช้งานผ่านเมธอด ExecuteQuery และต้องมีคลาสมารับผลลัพธ์ที่ได้รับกลับมาในรูปแบบของกลุ่มเรกอร์ดคือคลาส ResultSet

เมธอดทั้งสองรูปแบบดังกล่าว สามารถเรียกใช้งานได้จากคลาส Statement และคลาส PreparedStatement อย่งใดอย่างหนึ่ง แต่ทั้งสองคลาสดังกล่าวใช้สำหรับการสร้างคำสั่ง SQL ที่แตกต่างกันคือ Statement ใช้สำหรับสร้างคำสั่ง SQL ที่เป็นข้อมูลคงที่ (ข้อมูลในคำสั่ง SQL ไม่เปลี่ยนแปลง)

PreparedStatement ใช้สำหรับสร้างคำสั่ง SQL ที่สามารถเปลี่ยนแปลงข้อมูลบางส่วนภายในตามพารามิเตอร์ที่ต้องการใช้งาน

### รูปแบบคลาส Statement

```
int ExecuteUpdate() สำหรับส่งงานคำสั่ง SQL ประเภทจัดการข้อมูล
```

```
ResultSet executeQuery() สำหรับส่งงานคำสั่ง SQL ประเภทสืบค้นข้อมูล
```

### รูปแบบคลาส PreparedStatement

```

void setBoolean(int index, boolean x) กำหนดข้อมูล boolean ให้กับพารามิเตอร์ตำแหน่ง index
void setByte(int index, byte x) กำหนดข้อมูล byte5 ให้กับพารามิเตอร์ตำแหน่ง index
void setShort(int index, short x) กำหนดข้อมูล short ให้กับพารามิเตอร์ตำแหน่ง index
void setInt(int index, int x) กำหนดข้อมูล int ให้กับพารามิเตอร์ตำแหน่ง index
void setLong(int index, long x) กำหนดข้อมูล long ให้กับพารามิเตอร์ตำแหน่ง index
void setFloat(int index, float x) กำหนดข้อมูล float ให้กับพารามิเตอร์ตำแหน่ง index
void setDouble(int index, double x) กำหนดข้อมูล double ให้กับพารามิเตอร์ตำแหน่ง index
void setString(int index, String x) กำหนดข้อมูล String ให้กับพารามิเตอร์ตำแหน่ง index
void setDate(int index, Date x) กำหนดข้อมูล Date ให้กับพารามิเตอร์ตำแหน่ง index
void setTime(int index, Time x) กำหนดข้อมูล Time ให้กับพารามิเตอร์ตำแหน่ง index
void clearParameter() ลบข้อมูลออกจากพารามิเตอร์ทุกตัว
int executeUpdate() สำหรับส่งงานคำสั่ง SQL ประเภทจัดการข้อมูล
ResultSet executeQuery() สำหรับส่งงานคำสั่ง SQL ประเภทสืบค้นข้อมูล
void close() ปิดช่องทางการสื่อสาร

```

การขยับตัวชี้เพื่ออ้างอิงไปยังแถวที่ต้องการสามารถขยับได้โดยใช้เมธอดของ ResultSet ดังนี้

- หากต้องการย้ายไปที่ข้อมูลแถวแรกใช้เมธอด first()
- หากต้องการย้ายไปที่ข้อมูลแถวสุดท้ายใช้เมธอด last()
- หากต้องการย้ายไปที่ข้อมูลแถวถัดไปจากตำแหน่งปัจจุบันใช้เมธอด next()
- หากต้องการย้ายไปที่ข้อมูลแถวก่อนหน้าจากตำแหน่งปัจจุบันใช้เมธอด previous()
- หากต้องการย้ายไปที่ข้อมูลตามแถวตัวเลขที่ต้องการใช้เมธอด absolute(แถวที่ต้องการ)

ในกลุ่มเมธอดที่ใช้ในการตรวจสอบตำแหน่งแถวว่าอยู่แถวใด มีเมธอดในกลุ่มที่เกี่ยวข้องดังนี้

- หากต้องการทราบว่าขณะนี้ตัวชี้อยู่ที่ตำแหน่งแถวใดให้ใช้เมธอด getRow()
- หากต้องการทราบว่าขณะนี้ตัวชี้อยู่ตำแหน่ง Before First หรือไม่ให้ใช้เมธอด isBeforeFirst()
- หากต้องการทราบว่าขณะนี้ตัวชี้อยู่ตำแหน่ง First หรือไม่ให้ใช้เมธอด isFirst()
- หากต้องการทราบว่าขณะนี้ตัวชี้อยู่ตำแหน่ง After Last หรือไม่ให้ใช้เมธอด isAfterLast()
- หากต้องการทราบว่าขณะนี้ตัวชี้อยู่ตำแหน่ง Last หรือไม่ให้ใช้เมธอด isLast()

ตัวอย่างการใช้งานคำสั่ง SQL จัดการฐานข้อมูลด้วยคลาส Statement

```
Statement stmt = con.createStatement();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

stmt.executeUpdate("CREATE TABLE Units (" +
    " UnitID CHAR (2) PRIMARY KEY, " +
    " Description CHAR(30) NOT NULL)");

stmt.executeUpdate("CREATE TABLE Products (" +
    " ProductID CHAR (5) PRIMARY KEY, " +
    " Name CHAR (30) NOT NULL, " +
    " Price NUMBER (8, 2) CHECK > 0, " +
    " UnitID CHAR (2) REFERENCE Units (UnitID))");

```

### ตัวอย่างการใช้งานคำสั่ง SQL เรียกดูข้อมูลในฐานข้อมูลด้วยคลาส Statement

```

Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery(" SELECT * FROM Products ");

```

### ตัวอย่างการใช้งานคำสั่ง SQL จัดการฐานข้อมูลด้วยคลาส PreparedStatement

```

PreparedStatement stmt = con.prepareStatement("INSERT INTO Products(" +
    "ProductID, Name, Price) VALUES ( ?, ?, ?)");

// ใช้งานคำสั่งครั้งที่ 1
stmt.setString(1, "00001"); stmt.setString(2, "Pen"); stmt.setDouble(3, 8.00);
stmt.executeUpdate(stmt);

// ใช้งานคำสั่งครั้งที่ 2
stmt.setString(1, "00101"); stmt.setString(2, "Drawing Book"); stmt.setDouble(3, 32.75);
stmt.executeUpdate(stmt);

// ใช้งานคำสั่งครั้งที่ 3
stmt.setString(1, "00215"); stmt.setString(2, "Blue Ink"); stmt.setDouble(3, 12.50);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
stmt.executeUpdate(stmt);
```

### ขั้นตอนที่ 3 การแสดงผลลัพธ์สู่ผู้ใช้

ผลลัพธ์ในชุด ResultSet มีโครงสร้างคล้ายกับข้อมูลในเทเบิล คือแบบออกเป็นแนวแถวเรียกว่า Row (หรือเรคอร์ด) และแนวคอลัมภ์เรียกว่า Column (หรือฟิลด์) ดังนั้นหลักการดึงข้อมูล ก็ต้องระบุ row ให้ชัดเจนก่อนแล้วจึงค่อยระบุ column ที่ต้องการ

#### รูปแบบคลาส ResultSet

```
boolean getBoolean(String ColumnName) อ่านข้อมูล boolean ในฟิลด์ ColumnName
byte getByte(String ColumnName) อ่านข้อมูล byte ในฟิลด์ ColumnName
short getShort(String ColumnName) อ่านข้อมูล short ในฟิลด์ ColumnName
int getInt(String ColumnName) อ่านข้อมูล int ในฟิลด์ ColumnName
long getLong(String ColumnName) อ่านข้อมูล long ในฟิลด์ ColumnName
float getFloat(String ColumnName) อ่านข้อมูล float ในฟิลด์ ColumnName
double getDouble(String ColumnName) อ่านข้อมูล double ในฟิลด์ ColumnName
String getString(String ColumnName) อ่านข้อมูล String ในฟิลด์ ColumnName
Date getDate(String ColumnName) อ่านข้อมูล Date ในฟิลด์ ColumnName
Time getTime(String ColumnName) อ่านข้อมูล Time ในฟิลด์ ColumnName
```

#### ตัวอย่าง

```
Statement stmt = con.createStatement();
String SQL = "SELECT ProductID, Name, Price FROM Products WHERE Price >
10.00 ";
ResultSet rs = stmt.executeQuery(SQL);
rs.first();
while (!rs.isAfterLast()) {
    System.out.printf("Product ID: " + rs.getString("ProductID"));
    System.out.printf("Name: " + rs.getString("Name"));
    System.out.println("Price: " + rs.getDouble("Price"));
}
```

```

rs.next();
}

```

#### ขั้นตอนที่ 4 การยกเลิกการติดต่อ

สำหรับในขั้นตอนยกเลิกการติดต่อนี้เป็นขั้นตอนที่ง่ายและไม่ยุ่งยากในการทำงาน หากเมื่อพิจารณาจากการดำเนินการติดต่อข้อมูล มีอยู่ 2 ส่วนด้วยกัน ส่วนแรกคือช่องทางการติดต่อที่สร้างมาจากเมธอด `getConnection` และส่วนที่สองคือชุดคำสั่ง `ResultSet` ที่ได้จากเมธอด `executeQuery`

```

Connection con = DriverManager.getConnection ( url, "myLogin", "myPassword");
Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery("SELECT * FROM Products")
?
?
?
rs.close();
con.close();

```

ตัวอย่างการทำงานของระบบฐานข้อมูลผ่าน JDBC โดยทำงานในแบบ JDBC-ODBC Bridge บนฐานข้อมูลระบบ Access โดยก่อนทดสอบ ขอให้สร้าง Data Source Name (DSN) ผ่าน ODBC Administrator ของวินโดวส์ก่อน ในชื่อ myData

```

try {
    Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver"); }

String URL = "jdbc:odbc:myData";

Connection con =
DriverManager.getConnection(URL,"","");

Statement stmt1 = con.createStatement();
stmt1.executeUpdate("CREATE TABLE Units (" +
    " UnitID CHAR (2) PRIMARY KEY, " +
    " Description CHAR(30) NOT NULL)");

```

```

stmt1.executeUpdate("CREATE TABLE Products ("
    " ProductID CHAR (5) PRIMARY KEY, " +
    " Name CHAR (30) NOT NULL, " +
    " Price NUMBER (8, 2) CHECK > 0, " +
    " UnitID CHAR (2) REFERENCE Units
(UnitID) ");

PreparedStatement stmt2 =
con.prepareStatement( "INSERT INTO Products(" +
    "ProductID, Name, Price) VALUES ( ?, ?, ?)");
stmt2.setString(1, "00001"); stmt2.setString(2,
"Pen");
stmt2.setDouble(3, 8.00);
stmt2.executeUpdate(stmt2);
stmt2.setString(1, "00101"); stmt2.setString(2,
"Drawing Book");
stmt2.setDouble(3, 32.75);
stmt2.executeUpdate(stmt2);
stmt2.setString(1, "00215"); stmt2.setString(2, "Blue
Ink");
stmt2.setDouble(3, 12.50);
stmt2.executeUpdate(stmt2);

String SQL = "SELECT ProductID, Name, Price FROM
Products WHERE Price > ? ";

ResultSet rs = stmt1.executeQuery(SQL);
rs.first();
while (!rs.isAfterLast()) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        System.out.printf("Product ID: " +
rs.getString("ProductID");

        System.out.printf("Name: " + rs.getString("Name");
System.out.println("Price: " + rs.getDouble("Price");
        rs.next();
    }
    rs.close();
    con.close();
}
catch (ClassNotFoundException e) {
    System.err.print (e.getMessage ());
} catch (SQLException e) {
    System.err.print (e.getMessage ());
}
}

```

## 2.4 Data Mining

Data Mining นั้นเป็นการอ้างอิงข้อมูลที่ถูกจัดเก็บไว้ในฐานข้อมูล ซึ่งข้อมูลจะถูกจัดเก็บในรูปแบบที่เหมาะสม เช่น ระบบฐานข้อมูล, Data Warehouse หรือ Spreadsheet โดยนำข้อมูลมาเพียงส่วนที่ต้องการและมีประโยชน์ต่อการทำเท่านั้น Data Mining จะมีกระบวนการทำที่มีกฎเกณฑ์ และมีแบบแผน ซึ่งจะนำไปใช้ในการหาข้อมูลจำนวนมาก ให้ออกมาเป็นสารสนเทศที่เป็นประโยชน์ และสามารถนำไปใช้ในงานได้จริง

Data Mining เป็นกระบวนการดึง สิ่งที่ไม่รู้, สิ่งที่มีเหตุมีผล และสามารถนำไปใช้ได้ ของฐานข้อมูล หลังจากได้สาระสำคัญแล้ว ก็จะนำสิ่งที่ได้มาใช้ประกอบในการตัดสินใจทางธุรกิจ ซึ่งสิ่งที่ไม่รู้ในฐานข้อมูลจะถูกซ่อนเอาไว้ภายใต้ข้อมูลขนาดใหญ่ ซึ่งยากต่อการที่จะรู้ได้ว่า สิ่งที่ย่อนอยู่นั้น จะสามารถสืบค้นมาได้อย่างไร รวมถึงไม่ทราบประโยชน์, ความสัมพันธ์ และนำการเตือนภัยต่างๆ จึงจำเป็นต้องทราบถึงกระบวนการและหลักการทฤษฎีต่างๆที่มีให้เห็นในการทำ Data Mining

### 2.4.1 กระบวนการ Data Mining

ตามความเข้าใจของคนทั่วไป เมื่อพูดถึง Data เข้าใจว่าการทำ mining กับข้อมูล แล้วจะทำให้ได้สารสนเทศที่ต้องการ แต่ในความเป็นจริงแล้ว การทำ mining กับข้อมูล เป็นเพียงขั้นตอนหนึ่งของการทำ Data Mining ซึ่งเป็นเพียงแค่ 10% ของขั้นตอนทั้งหมดของ Data Mining เท่านั้น ภาพที่ X แสดงให้เห็นขั้นตอนทั้งหมดของ Data Mining จากภาพในทุกขั้นตอน จะต้องมีข้อมูลเข้า (input data) ซึ่งเป็นข้อมูลดิบ และเมื่อทำขั้นตอนใดเสร็จสิ้นสุดลง ผลที่ได้ (result) จะนำไปเป็นข้อมูลเข้าของขั้นตอนถัดไป ซึ่งจะเห็นว่าในทุกๆขั้นตอน จะมีลูกศรชี้ย้อนกลับไปขั้นตอนที่ได้ผลไม่ถูกต้องหรือไม่สมควร เราก็สามารถไปเริ่มทำใหม่ในขั้นตอนก่อนหน้านี้ได้ เพื่อให้ได้สารสนเทศ ที่เป็นประโยชน์และถูกต้องมากที่สุด โดยสามารถสรุปในแต่ละขั้นตอนดังนี้

#### ขั้นตอนที่ 1: Business Objective Determination

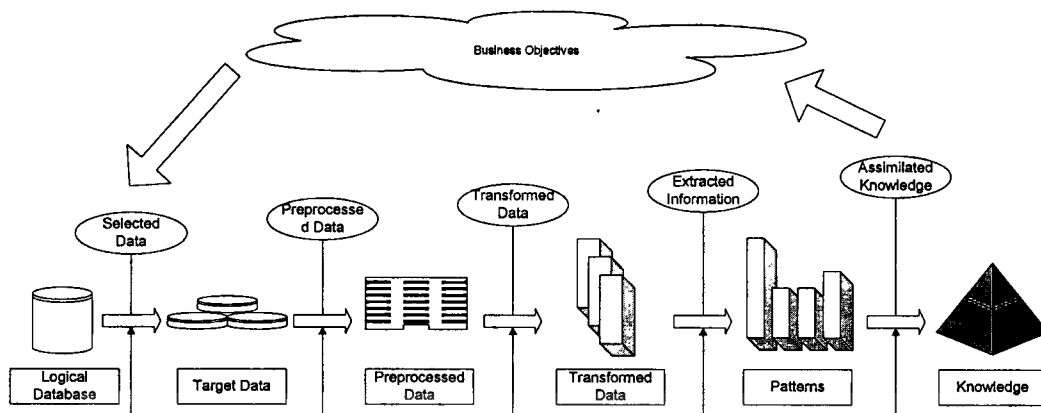
Business Objective เป็นตัวจักรสำคัญที่จะทำให้การทำ Data Mining ในทุกๆ ขั้นตอนขับเคลื่อนต่อไปได้ โดยที่นักวิเคราะห์ธุรกิจ (Business Analyst) จะต้อง ระบุปัญหาที่เกิดขึ้นในการทำธุรกิจให้ครอบคลุมและชัดเจน รวมทั้งต้องบอกถึงวัตถุประสงค์ของการทำ Data Mining ให้ได้ว่า เพื่อวัตถุประสงค์อะไร

#### ขั้นตอนที่ 2 : Data Preparation

เป็นขั้นตอนที่จะต้องใช้ Resource มากที่สุดถึง 60% ของ project ทั้งหมด ในขั้นตอนนี้สามารถแบ่งย่อยออกได้เป็น 3 เฟส ดังนี้

- Data Selection

เป้าหมายของ Data Selection คือการ identify แหล่งข้อมูล (data source) ที่มีอยู่และทำการแยกแยะ (extract) ข้อมูลเพื่อให้ได้ข้อมูลกลุ่มย่อย (subset) จากข้อมูลทั้งหมด ที่จำเป็นสำหรับขบวนการทำ Data Mining ขึ้นต่อไป ซึ่งสิ่งที่เราจะพิจารณาในการเลือกข้อมูล เช่น data type, potential value, original source system, data format และ คุณสมบัติอื่นๆ ของข้อมูลที่นักวิเคราะห์ เห็นว่าจำเป็นสำหรับการทำ Data Mining ซึ่งเราสามารถแบ่ง variable ของ data ได้ 2 ประเภทดังนี้



ภาพที่ 2.12 ขบวนการทำ Data Mining

- Categorical คือค่าที่เป็นกลุ่มเป็นหมู่ที่สามารถ บอกถึงสิ่งนั้น ได้อย่างชัดเจนซึ่งแบ่งย่อยได้เป็น 2 ประเภท คือ nominal และ ordinal

สำหรับ Nominal เป็นการอ้างถึง (refer) สิ่งที่เป็น object ซึ่งลำดับ (order) ของค่าไม่มีความสำคัญ เช่น เพศ (ชาย, หญิง) เป็นต้น

ส่วนแบบ Ordinal variable เป็นค่าที่ต้องเรียงตามลำดับ เช่น ระดับความน่าเชื่อถือของลูกค้า (ดี, ปานกลาง, ไม่ดี) เป็นต้น

- Quantitative เป็นค่าที่บ่งบอกถึงความแตกต่าง (ขนาดหรือปริมาณ) ระหว่างค่าที่เป็นไปได้ แบ่งได้ 2 ชนิด คือ Continuous (real number) เช่น ภาษี, ค่าเฉลี่ยของรายได้แต่ละเดือน และชนิด discrete เช่น จำนวนของลูกค้า เป็นต้น

#### ● Data Preprocessing

เมื่อได้ข้อมูลจากขั้นตอนที่ 2.1 แล้วนักวิเคราะห์จะต้องทำการ Clean ข้อมูลเพื่อให้ได้ข้อมูลที่มีคุณภาพและถูกต้องสมเหตุสมผล ซึ่งข้อมูลจากขั้นตอนนี้จะนำไปสู่การเลือกประเภทของ mining operation ต่อไปซึ่ง ข้อมูลที่จะสร้างปัญหาแก่นักวิเคราะห์ แบ่งได้เป็น 2 ประเภท

- Noise data คือ variables ที่มีค่าแตกต่างไปจาก data ตัวอื่นๆ ที่เรียกว่า outlier ซึ่งค่า outlier ที่เกิดขึ้นอาจจะจริง (actual) หรือเกิดจากความผิดพลาดของคน ดังนั้นนักวิเคราะห์จะต้องใช้ ความละเอียดรอบคอบในการแยกแยะ noise data ที่เกิดขึ้น

- Missing Values อาจเกิดจากความผิดพลาดของคน (human error) หรือจากการเลือกข้อมูลที่ได้มาจาก source ที่มีหน่วยแตกต่างกัน (heterogeneous source) เป็นต้น ซึ่งเทคนิคต่างๆ ที่ใช้ในการจัดการกับ missing value คือการถอยกลับไป (drawback) สุดท้ายข้อมูลผิดพลาด ตัว

missing value จะไม่สร้างปัญหาอะไรถ้ามันอยู่ในกลุ่มของข้อมูลจำนวนมากๆ แต่จะส่งผลกระทบต่อการทำ Data Mining ให้ผิดพลาดได้ถ้ามันอยู่ในกลุ่มข้อมูลเล็กๆ ซึ่งบางครั้งนักวิเคราะห์จะต้องพิจารณาจุดนี้ด้วยว่า คุ่มค่าหรือไม่ที่จะย้อนกลับไปทำให้ missing value นั้นถูกต้อง

- Data Transformation

ในบางครั้งข้อมูลที่ได้อาจจะไม่เหมาะสมที่จะใช้กับโมเดลของการวิเคราะห์ข้อมูลแต่ละแบบ ดังนั้นนักวิเคราะห์ข้อมูลจะต้องทำให้รูปแบบ (Format) ของข้อมูลทั้งหมดอยู่ในหน่วยเดียวกันใน scale ที่เหมาะสม ตัวอย่างเช่น ถ้าข้อมูลที่ได้เป็นแบบ categories variable เช่น ชนิดของรถยนต์ FORD, NISSAN, TOYOTA ซึ่งข้อมูลเหล่านี้มาเหมาะสมกับอัลกอริทึม ดังนั้นจะต้องแปลงข้อมูลอยู่ในรูปอื่น เช่น FORD = 001, NISSAN = 010, TOYOTA = 100 เป็นต้น

### ขั้นตอนที่ 3: Data Mining

เป็นขั้นตอนที่ทำการ mining ข้อมูลจริงๆ โดยที่มี operation ในการทำ Data Mining หลายแบบเช่น Database Segmentation, Predictive Modeling, Link Analysis, Deviation Detection ซึ่งในแต่ละ operation จะมีอัลกอริทึมแต่ละตัวไป ดังนั้นการเลือกอัลกอริทึมสำหรับการ mining นักวิเคราะห์ข้อมูลจะต้องเลือกชนิดของอัลกอริทึม ให้ตรงกับวัตถุประสงค์ของการทำ Data Mining ซึ่งบางครั้งเป็นไปได้ที่จะใช้มากกว่าหนึ่งอัลกอริทึมของ database Segmentation Operation ในการ cluster ข้อมูลออกเป็น segments และต่อจากนั้นก็ใช้อัลกอริทึมของ Association Operation เพื่อช่วยหาความสัมพันธ์ของข้อมูลแต่ละ segment

### ขั้นตอนที่ 4: Analysis of Results

ผลที่ได้จากขั้นตอนที่ 3 นักวิเคราะห์ข้อมูลและนักวิเคราะห์ธุรกิจจะต้องพยายามแปลความหมายและประเมินค่า (Evaluate) ผลที่ได้ ซึ่งการวิเคราะห์ผลจะต้องเป็นไปตามแนวทางของ Data Mining Operation ที่เลือกใช้ในขั้นตอนที่ 3 เพื่อให้สารสนเทศที่ได้ถูกต้องและนำไปใช้ได้ ตัวอย่างเช่น การใช้ Database Segmentation สำหรับแบ่งกลุ่มลูกค้า ซึ่งนักวิเคราะห์จะต้องพยายามกำหนด label สำหรับแต่ละ segment ที่จะต้องบ่งบอกถึงความหมายในธุรกิจ โดยที่แต่ละ segment จะต้องไม่ซ้ำกัน อย่างไรก็ตามถ้ามี segment เพียงเล็กน้อยซึ่งมันไม่เพียงพอที่จะทำให้เราเห็นความแตกต่างระหว่าง segment ได้ในกรณีนี้เราจะต้องแก้ไขให้ดีขึ้น โดยการ removing ตัว common variables จาก segment ที่มีขนาดใหญ่ให้เป็นหน่วยเล็กๆ จำนวนมากแทน ซึ่งจะต้องกลับไปทำใหม่ (rerun) ในขั้นตอนนี้ซ้ำๆจนกว่าจะได้ result เป็นที่พอใจ

### ขั้นตอนที่ 5: Assimilation of knowledge

การนำไปสู่ความรู้ เป็นขั้นตอนสุดท้ายของการทำ Data Mining ซึ่งมี 2 ประเด็นหลักๆ ที่จะชี้ให้เห็นถึงสิ่งที่ได้จากขั้นตอนนี้

- Information ที่ได้จะช่วยเป็นแนวทางในการดำเนินธุรกิจในรูปแบบใหม่ๆ ตัวอย่างเช่น การใช้ Database Segmentation Operation ในการทำ Data Mining จะช่วยให้การทำการตลาดได้ตรงกลุ่มเป้าหมายกลุ่มใหม่ๆ ได้มากขึ้น เช่นการทำประชาสัมพันธ์ โดยวิธี direct mailing จะทำให้ได้กลุ่มเป้าหมายที่ชัดเจนกว่าการส่ง mail แบบลุ่ม หรืออาจจะเป็นกรณีการทำ Data Mining โดยใช้ Association Rules Operation จะช่วยให้เกิดแนวคิดการให้บริการแก่ลูกค้าได้ดียิ่งขึ้น เช่นในกรณีตัวอย่างเรื่อง ผ้าอ้อมและเบียร์ที่กล่าวตอนต้น เมื่อเรารู้ถึงความสัมพันธ์นั้นเราอาจจะจัดวางผ้าอ้อมและเบียร์ไว้ใกล้ๆ กันในทุกๆ วันศุกร์ ก็จะช่วยให้เราขายเบียร์และผ้าอ้อมได้มากขึ้น เป็นต้น
- Information ที่ได้จะช่วยให้เกิดประโยชน์กับการพัฒนาองค์กร เช่นความรู้ที่ได้จะนำไปใช้ในระบบผู้เชี่ยวชาญ (expert system) หรืออาจนำไปใช้ใน database procedure หรือทำให้ได้เป็น application code เป็นต้น

#### 2.4.2 เทคนิคการแบ่งส่วนฐานข้อมูล (Database Segmentation)

จากที่กล่าวมาข้างต้นเราแบ่ง operation ของการทำ Data Mining ได้เป็น 4 ประเภท แต่ในบทความนี้จะกล่าวถึงเฉพาะ database segmentation คือการแบ่งฐานข้อมูลให้กลายเป็น segment ย่อย โดยที่แต่ละ segment จะประกอบด้วย record ที่เหมือนกัน (similar) ซึ่งกล่าวได้ว่าคุณสมบัติของแต่ละ record ใน segment จะเป็น homogeneous (โดยที่คำว่า segmentation และ clustering สามารถใช้เรียก Data Mining operation ที่มีความหมายเหมือนกัน โดยที่คำว่า segment หรือ cluster จะใช้เรียกชื่อกลุ่มของ data record) โดยที่เทคนิคการทำ clustering มีหลายวิธี เช่น Demographic clustering, Neural clustering, K-means clustering หรือ decision tree เป็นต้น

##### 2.4.2.1 Demographic Clustering

แนวความคิดพื้นฐานของการทำ Demographic clustering คือการสร้าง segment แล้วนำข้อมูลใน record มาเปรียบเทียบกับทุกๆ segment ที่สร้างขึ้นขณะ run Data Mining

Demographic clustering เหมาะสมกับข้อมูลที่เป็นแบบ categorical โดยใช้หลักการเทคนิคการวัดระยะห่าง (distance) ระหว่าง record (เป็นการวัดความเหมือนหรือต่างกันของ record) แล้วให้คะแนน ที่เรียกว่าเป็นแบบ Condorset โดยที่จะเป็นการให้คะแนนระหว่าง field แต่ละ field ของแต่ละ record ถ้า field เหมือนกัน จะจัดให้อยู่กลุ่มเดียวกัน ความเหมือน (similar) หรือไม่เหมือน (dissimilar) ตัดสินได้จากจำนวน degree ของ record ที่นำมาเปรียบเทียบกับ โดยที่ field ที่นำมาเปรียบเทียบกับ ถ้าเหมือนกันจะให้ค่าเป็น +1 และถ้าต่างกันจะให้ค่า -1 และสุดท้ายจะนำคะแนนที่

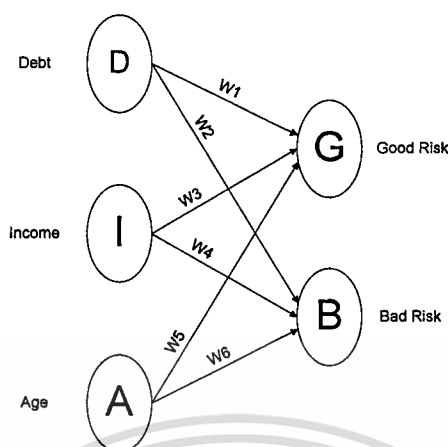
ได้ของแต่ละ record มากำหนดแล้วแบ่งกลุ่มรวบรวมผลที่ได้ และจัดข้อมูลให้อยู่ในแต่ละ segment ที่กำหนด แล้วนำผลไปวิเคราะห์ต่อไป โดยที่อัลกอริทึมวิธีนี้จะเป็นแบบ supervised learning ซึ่งจะแตกต่างจาก Neural network

#### 2.4.2.2 Neural clustering

เทคนิคแบบ neural network เป็นวิธีการเลียนแบบการทำงานของสมองมนุษย์ โดยได้จำลองลักษณะการทำงานให้ใช้ได้กับสถาปัตยกรรมบนคอมพิวเตอร์ โดยที่อัลกอริทึมในลักษณะนี้จะเรียกว่าเป็นแบบ Unsupervised learning ซึ่ง neural clustering มีอัลกอริทึมหลายแบบ แต่ที่นำมากล่าวเป็น Kohonen feature map neural network โดยที่ Kohonen feature map ใช้กระบวนการที่เรียกว่า self-organization ในการปรับแต่งค่าหน่วยของผลลัพท์เข้าสู่ Topological Map การทำงานของ Kohonen feature map นั้นประกอบด้วยหน่วยประมวลผล 2 ชั้น คือ ชั้นข้อมูลเข้า (input layer) และชั้นผลลัพท์ข้อมูล (output layer) โดยที่ระหว่างชั้นทั้งสอง จะมีการเชื่อมต่อกันทุกๆ หน่วย เมื่อมีรูปแบบข้อมูลเข้ามาในชั้นข้อมูลเข้าทุกๆ หน่วยในชั้น ผลลัพท์จะแข่งขันกันทุกๆ หน่วย เพื่อจะได้เป็นผู้ชนะ ผู้ชนะในหน่วยผลลัพท์จะมีใกล้เคียงกับรูปแบบข้อมูลเข้ามามากที่สุด และจะได้รับสิทธิ์ปรับน้ำหนักการเชื่อมต่อ (\*weight connection) จากภาพที่ 2.13 แสดงให้เห็นถึงตัวอย่างของกระบวนการวิเคราะห์ความเสี่ยงของการปล่อยเงินกู้ของธนาคาร โดยชั้นข้อมูลเข้าจะมี 3 หน่วยคือ หนี้สิน (C), รายได้ (Y) และอายุ (A) ของผู้กู้ เมื่อมีรูปแบบหน่วยข้อมูลเข้ามา หน่วยผลลัพท์แต่ละหน่วยก็จะทำการคำนวณ และปรับน้ำหนักให้มีรูปแบบใกล้เคียงกับข้อมูลเข้ามาที่สุด โดยคำนวณได้จาก สูตร

$$G = D(W_1) + I(W_3) + A(W_5)$$

$$G = D(W_2) + I(W_4) + A(W_6)$$



ภาพที่ 2.13 เทคนิคของ Kohonen Feature Map

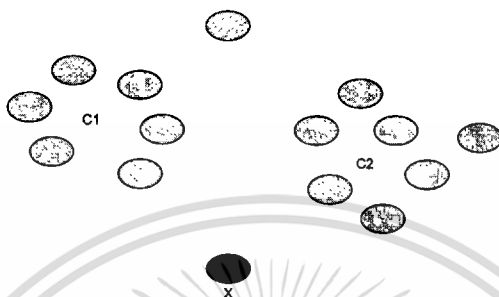
ผลลัพธ์ที่ได้จากการคำนวณถ้าหน่วยใดมีค่ามากกว่า แสดงว่าหน่วยนั้นชนะ จากตัวอย่างหากค่าผลลัพธ์ของลูกค้านั้นมีค่ามากกว่าเป็นผู้ชนะก็แสดงว่า เราควรให้กู้แก่ลูกค้าที่มีคุณสมบัติตามรูปแบบข้อมูลที่ถูกเสนอมา เช่นเดียวกันถ้าหน่วยผลลัพธ์ลูกค้าไม่ดีเป็นผู้ชนะ เราก็ไม่ปล่อยเงินกู้แก่ลูกค้าที่มีรูปแบบข้อมูลตามที่เสนอมา ซึ่งค่า Weight ในแต่ละการเชื่อมต่อจะถูกควบคุมโดยรูปแบบข้อมูลเข้า โดยที่มีอัตราการเรียนรู้ เป็นปัจจัยกำหนด (learning rate parameter) โดยที่วิธีนี้ไม่เหมาะสมกับฐานข้อมูลขนาดใหญ่ ให้ความสำคัญสูงกับกลุ่มข้อมูลขนาดกลางและเล็ก

#### 2.4.2.3 K-Means Clustering

K-Means Clustering เป็นอัลกอริทึมที่ใช้หาศูนย์กลางของเซตจำนวน  $N$  ตัวและให้ข้อมูลกับกลุ่มที่มีระยะห่างของข้อมูลใกล้เคียงกับศูนย์กลางของกลุ่มใดกลุ่มหนึ่งมากที่สุดและตำแหน่งของศูนย์กลางจะถูกปรับใหม่โดยการคำนวณซ้ำไปซ้ำมาทำไปเรื่อยๆ จนไม่มีการเปลี่ยนแปลงอีกจึงจะหยุด โดยที่จุดศูนย์กลาง คือจุดที่เป็นตัวแทนในแต่ละ cluster นั้นๆ จากภาพที่ 3 จะเห็นว่าข้อมูล  $x$  ได้หลุดจากกลุ่ม  $c_1$  และจะถูกจับให้อยู่ในกลุ่ม  $c_2$  เพราะว่ามันมีระยะห่างระหว่างจากจุดศูนย์กลาง  $c_2$  น้อยกว่าระยะห่างจากตัวมันกับจุดศูนย์กลาง  $c_1$

โดยขั้นตอนการทำงานของ K-Means Clustering เริ่มจากเราต้องกำหนดจำนวน cluster ที่ต้องการให้เป็นจำนวน  $K$  cluster ต่อจากนั้นนำเอาข้อมูลมาจำนวน  $K$  ตัว จากกลุ่มของข้อมูลตัวอย่าง (จำนวน  $N$  ตัว) มากำหนดให้เป็นศูนย์กลางของ  $K$  cluster ต่อจากนั้นนำเอาข้อมูลตัวอย่าง

มาจัดให้อยู่ในแต่ละกลุ่ม โดยมีหลักการว่าระยะห่างจากตัวมันกับศูนย์กลางของแต่ละ cluster ใดมีค่าน้อยที่สุดก็ให้สมาชิกเข้าไปอยู่ในกลุ่มนั้น ทำเช่นนี้กับข้อมูลทุกตัวจนครบ  $N$  ตัว จากนั้นทำการคำนวณหาศูนย์กลางใหม่ แล้วนำสมาชิกข้อมูลตัวอย่างแต่ละตัวเทียบหาศูนย์กลางที่คำนวณได้ใหม่ในแต่ละกลุ่มแล้ววัดระยะห่างของข้อมูลตัวอย่างกับศูนย์กลางของกลุ่ม



ภาพที่ 2.14 ตัวอย่างการจัดกลุ่มแบบ K-means Clustering

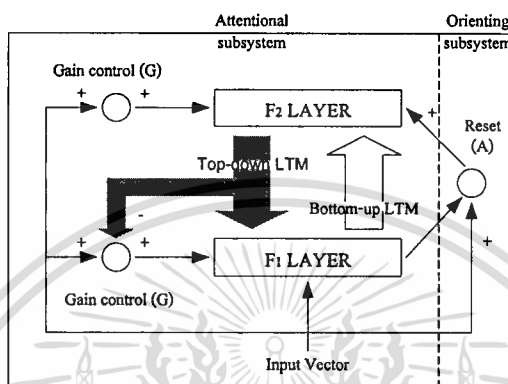
ถ้าหากข้อมูลใดในกลุ่มนั้นมีค่าใกล้เคียงกับศูนย์กลางอื่นก็จับข้อมูลไปอยู่ในกลุ่มนั้น เมื่อทำอย่างนี้จนได้ข้อมูลใหม่ในแต่ละกลุ่มแล้วให้ทำการหาศูนย์กลางใหม่อีกครั้งทำซ้ำขั้นตอนนี้เรื่อยๆ จนไม่มีการเปลี่ยนแปลงของข้อมูลของกลุ่มใดกลุ่มหนึ่งอีก

## 2.5 Adaptive Resonance Theory

Neural Network นั้น เป็นการจำลองหรือเลียนแบบ ลักษณะการทำงานของเครือข่ายประสาท โดยในเครือข่านั้นจะมีการเชื่อมโยงกันของเซลล์ประสาท เพื่อส่งสัญญาณสื่อสารกัน ซึ่งในการพัฒนาระบบครั้งนี้ ได้นำเอา ARTMAP มาประยุกต์ใช้กับระบบงาน โดย ARTMAP นี้เป็นวิธีการแขนงหนึ่งของ Adaptive Resonance Theory (ART) โดยที่ ART นั้นมีหลักการคร่าวๆ เหมือนกับชื่อ คือ จะมีการปรับหรือเปลี่ยนแปลง (Adaptive) ภายใน Network ให้เกิด สถานะที่สอดคล้องและมั่นคง (Resonance) ซึ่งก่อนที่จะอธิบายหลักการของ ARTMAP นั้น จำเป็นต้องเข้าใจหลักการพื้นฐานของ ART1 ก่อน เพราะว่า ARTMAP นั้นเกิดมาจากการนำเอา ART1 สองตัวมาเชื่อมโยงกัน แล้วหาค่า output ที่สอดคล้องกัน (โดยการนำ output ของ ART1 ทั้งสองตัว มา map หาค่าที่เหมาะสม)

### 2.5.1 ART1

ART1 นั้นเป็น Neural Network แบบ Unsupervised เพราะ ART1 จะสามารถให้คำตอบออกมาได้โดยไม่ต้องมีการเรียนรู้รูปแบบของ Output ซึ่งสิ่งที่ ART1 ให้คำตอบออกมานั้น จะอยู่ในรูปของ Cluster ชุดข้อมูลของ Input และ Input (ตัวเลข) ที่เข้าไปใน ART1 นั้นต้องอยู่ในรูปของไบนารี 0 และ 1 เท่านั้น โดยโครงสร้างของ ART1 แสดงดังภาพที่ 2.15



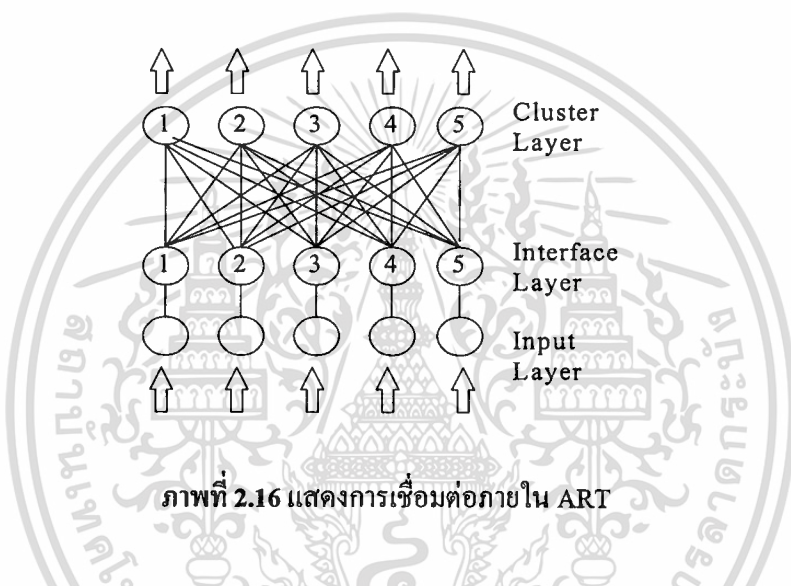
ภาพที่ 2.15 สถาปัตยกรรมของ ART1

จากภาพที่ 2.15 แสดงโครงสร้างของ ART1 โดย ART นั้นประกอบด้วย 2 ส่วน คือ ส่วนที่เรียกว่า Attentional Subsystem ซึ่งจะประกอบด้วย 2 layer คือ F1 กับ F2 และส่วนที่เรียกว่า Orienting Subsystem โดยในส่วน Attentional Subsystem นั้น F1 layer จะเป็นส่วนของการป้อนข้อมูลเข้า หรือ Input ส่วน F2 layer จะเป็น Output ที่อยู่ในรูปการ Cluster ชุดข้อมูล ยกตัวอย่างเช่น เราป้อนค่า input ที่ F1 ด้วยค่า 010 แล้ว F2 จะบอกเราว่าค่า 010 ที่ป้อนเข้ามานั้น อยู่ใน node ไหนของ F2 ซึ่งค่าที่ F2 นั้นจะ Active เพียง node เดียวคือ 1 input และในส่วนของ Orienting Subsystem นั้น จะคอยส่งสัญญาณ Reset ให้กับ F2 ถ้า input นั้นมาอยู่ใน node ที่ไม่มีความสอดคล้องกับชุด Input เดิมที่เคยมีอยู่

ค่า Weight ที่อยู่ในระหว่างการ Train ข้อมูลนั้น จะเรียกว่า Short-Term Memory (STM) ที่เรียก STM เพราะว่า weight ที่อยู่ในระหว่างการ train นั้น มีการเปลี่ยนแปลงอยู่ตลอดเวลา และจะมี Long-Term Memory (LTM) ไว้กรณีที่มีการ train นั้น เกิด resonance แล้ว และนำค่า LTM นั้นมาใช้ในระหว่างการ Test ข้อมูล

ในส่วนการ Control นั้น ในแต่ละ layer จะมี Gain Control เป็นของตัวเอง โดย Gain Control ทั้ง 2 ตัวนั้น จะไม่ยุ่งเกี่ยวกับ แต่ทั้งคู่จะมีความสัมพันธ์กับ Input pattern ที่เข้ามา ซึ่งจะเอาไว้คอยตรวจสอบดูว่า Input pattern นั้น เหมาะสมกับที่มีอยู่หรือไม่ เพื่อให้เข้าใจเกี่ยวกับ Gain

Control มากขึ้นนั้น ทั้ง 2 Gain Control จะทำงานอยู่ในระหว่าง ขั้นตอนที่ 3,5 และ 7 ที่จะกล่าวใน หัวข้อต่อไป และทั้ง F1 และ F2 นั้น จะเชื่อมต่อกันดังภาพที่ 2.16 โดย weight ที่เชื่อมต่อยกหว่าง F1 กับ F2 นั้น จะประกอบด้วย 2 weight คือ Bottom-up กับ Top-down โดย input จาก F1 นั้นจะส่ง ค่าผ่านทาง Bottom-up ไปให้กับ F2 ส่วน Top-down นั้นจะส่งค่าการคำนวณ node ที่ขณะนั้น กลับมาตรวจสอบกับ Input pattern ว่าเหมาะสมหรือไม่ โดยการเปรียบเทียบกับค่า Vigilance ถ้าได้ ค่ามากกว่าที่ตั้งไว้ แสดงว่า node ที่ขณะนั้นเหมาะสมแล้ว ซึ่ง Gain Control ในการพัฒนาโครงงาน ครั้งนี้ ได้ตั้งค่าไว้ที่ 1 หรือเรียกกันว่า Fast Learning ซึ่งจะไม่มีการปรับค่า Learning Rate ใดๆ ทำนองเดียวกัน ถ้า Gain Control สามารถปรับค่าได้ จะเรียกว่า Slow Learning



### 2.5.2 ART1 Algorithm

ART1 นั้นสามารถอธิบายหลักการในช่วงการ Train ได้ดังนี้

ขั้นตอนที่ 1 เริ่มแรกนั้นจะต้องให้ค่าเริ่มต้นกับ Bottom-up weight โดยให้  $b_{ij} = 1 / (n + 1)$  และให้ค่า Top-down weight เท่ากับ  $t_{ji} = 1$ ,  $i$  คือ ลำดับ node input ที่ F1,  $j$  คือลำดับ node output ที่ F2,  $m$  คือ จำนวน output พร้อมกับการตั้งจำนวน output ซึ่งโดยส่วนใหญ่ จะตั้งจำนวน output ไว้ที่  $n-1$  และ  $n$  คือ จำนวน input และสุดท้ายคือการกำหนดค่า  $\rho$  (Vigilance) ที่มีค่าระหว่าง  $0 - 1$  (ซึ่งในการพัฒนาครั้งนี้ กำหนด  $\rho$  ไว้ที่ 0.8)

ขั้นตอนที่ 2 ใส่ค่า input ให้กับ F1 (เลขไบนารี)

ขั้นตอนที่ 3 คำนวณผลรวมของ weight โดยให้  $j = 1, 2, \dots, m$

$$y_j = \sum_{i=1}^n b_{ij}x_i$$

$y_j$  คือ Output node ที่ F2,  $x_i$  คือ ลำดับ  $i$  ของ input และ  $b_{ij}$  คือ bottom-up weight จาก node  $i$  ใน F1 layer ไปที่ node  $j$  ใน F2 layer

ขั้นตอนที่ 4 หาค่า node  $j^*$  ซึ่งเป็นค่ามากที่สุดของ  $y_j$  ใน F2 layer โดยให้  $j = 1, 2, \dots, m$

$$y_j^* = \max \{y_j\}$$

ในกรณีนี้ ค่า  $y_j$  ให้ค่าออกมาเหมือนกัน ให้ทำการเลือกค่าตัวที่อยู่ทางซ้ายสุด

ขั้นตอนที่ 5 นำ pattern ที่ได้และ input pattern มาเปรียบเทียบกับค่า vigilance  $\rho$

$$\sum_{i=1}^n \frac{t_{ij} \cdot x_i}{\|x\|} > \rho$$

โดยที่  $t_{ij}$  นั้นเป็นค่า Top-down weight จาก node  $j^*$  ที่ขณะ  $\|x\|$  คือ ผลรวมของ input ก็คือ  $\sum_{i=1}^n x_i$  ถ้าคำนวณแล้วค่าได้เป็นจริง(คือมากกว่า  $\rho$ ) แสดงว่า input นั้นเหมาะสมที่จะอยู่ใน node  $j^*$  แล้ว ให้ไปขั้นตอนต่อไป แต่ถ้าค่าที่ได้เป็นเท็จ(น้อยกว่า  $\rho$ ) ให้ไปขั้นตอนที่ 8

ขั้นตอนที่ 6 ถ้าไม่มี Input เข้ามาอีกแล้ว ให้เก็บค่า weight แล้วเก็บค่า Input ว่าอยู่ใน node ใด แต่ละ node มีสมาชิกเท่าไรและออกจากโปรแกรม แต่ถ้ายังมี input เข้ามา ให้ไปขั้นตอนต่อไป

ขั้นตอนที่ 7 update weight ของ Bottom-up ( $b_{ij}$ ) และ Top-down ( $t_{ij}$ )

$$t_{ij^*}(k+1) = t_{ij^*}(K)x_i$$

$$b_{ij^*}(k+1) = \frac{t_{ij^*}(k)x_i}{0.5 + \left(\sum_{i=1}^n t_{ij^*}(k)x_i\right)}$$

โดยค่า  $k$  คือขั้นเวลาใดๆ แล้วให้ไปที่ขั้นตอนที่ 2

ขั้นตอนที่ 8 ถ้าค่าที่ได้นั้นไม่เหมาะสมแล้ว ให้ทำการ Reset node  $j^*$  ให้เป็นศูนย์ แล้วจะต้องไม่นำเอา  $j^*$  ที่ได้มาคำนวณอีก จากนั้นไปขั้นตอนที่ 3

เพื่อให้เกิดความเข้าใจมากยิ่งขึ้น จะลองยกตัวอย่างการคำนวณ สมมติว่ามี input pattern อยู่ 3 ตัวคือ (101), (110) และ (001)

ขั้นตอนที่ 1 ให้  $\rho = 0.8$ ,  $n = 3$  และ  $m = 3$  จะได้

$$t_{ij} = 1, b_{ij} = \frac{1}{(1+3)} = 0.25$$

โดย  $i, j = 1, 2, 3$

ขั้นตอนที่ 2 ใส่ input ตัวแรก (101)

ขั้นตอนที่ 3 คำนวณ

$$y_1 = b_{11}x_1 + b_{21}x_2 + b_{31}x_3 = 0.25 \cdot 1 + 0.25 \cdot 0 + 0.25 \cdot 1 = 0.5$$

$$y_2 = b_{12}x_1 + b_{22}x_2 + b_{32}x_3 = 0.25 \cdot 1 + 0.25 \cdot 0 + 0.25 \cdot 1 = 0.5$$

$$y_3 = b_{13}x_1 + b_{23}x_2 + b_{33}x_3 = 0.25 \cdot 1 + 0.25 \cdot 0 + 0.25 \cdot 1 = 0.5$$

ขั้นตอนที่ 4 เมื่อค่า  $y_1 = y_2 = y_3 = 0.5$  ดังนั้นเลือกค่าที่อยู่ซ้ายสุด คือ node 1

ขั้นตอนที่ 5 คำนวณ

$$\|x\| = 1 + 0 + 1 = 2$$

$$t_{11}x_1 + t_{21}x_2 + t_{31}x_3 = 1 \cdot 1 + 1 \cdot 0 + 1 \cdot 1 = 2$$

ดังนั้น  $\frac{2}{2} = 1 > 0.8$  ดังนั้น input (101) จะถูกจัดให้อยู่ใน node 1

ขั้นตอนที่ 6 ยังมี input เหลืออยู่อีก

ขั้นตอนที่ 7 ดังนั้น update ค่า weight ของ  $t_{11} = 1 \cdot 1 = 1$ ,  $t_{21} = 1 \cdot 0 = 0$ , และ  $t_{31} = 1 \cdot 1 = 1$

ส่วนค่า  $t_{ij}$  ที่เหลือจะไม่มีการเปลี่ยนค่า weight แล้ว update ค่า weight ของ

$$b_{11} = \frac{t_{11}x_1}{0.5 + (t_{11}x_1 + t_{21}x_2 + t_{31}x_3)} = \frac{1}{0.5 + 1 \cdot 1 + 1 \cdot 0 + 1 \cdot 1} = 0.4$$

$$b_{21} = \frac{t_{21}x_2}{0.5 + (t_{11}x_1 + t_{21}x_2 + t_{31}x_3)} = \frac{0}{0.5 + 1 \cdot 1 + 1 \cdot 0 + 1 \cdot 1} = 0$$

$$b_{31} = \frac{t_{31}x_3}{0.5 + (t_{11}x_1 + t_{21}x_2 + t_{31}x_3)} = \frac{1}{0.5 + 1 \cdot 1 + 1 \cdot 0 + 1 \cdot 1} = 0.4$$

ส่วนค่า weight ของ  $b_{ij}$  ที่เหลือจะไม่มีการเปลี่ยนแปลง แล้วไปขั้นตอนที่ 2

ขั้นตอนที่ 2 ใส่ input ตัวที่สอง (110)

ขั้นตอนที่ 3 คำนวณ

$$y_1 = b_{11}x_1 + b_{21}x_2 + b_{31}x_3 = 0.4 \cdot 1 + 0 \cdot 1 + 0.4 \cdot 0 = 0.4$$

$$y_2 = b_{12}x_1 + b_{22}x_2 + b_{32}x_3 = 0.25 \cdot 1 + 0.25 \cdot 1 + 0.25 \cdot 0 = 0.5$$

$$y_3 = b_{13}x_1 + b_{23}x_2 + b_{33}x_3 = 0.25 \cdot 1 + 0.25 \cdot 1 + 0.25 \cdot 0 = 0.5$$

ขั้นตอนที่ 4 เลือก  $j^*$  ที่ node 2

ขั้นตอนที่ 5  $\|x\| = 1 + 1 + 0 = 2$  และ  $t_{12}x_1 + t_{22}x_2 + t_{32}x_3 = 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 = 2$

ดังนั้นแล้ว  $\frac{2}{2} = 1 > 0.8$  แสดงว่า input ที่สอง (110) จะอยู่ใน node ที่ 2

ขั้นตอนที่ 6 ยังมี input เหลืออยู่อีก

ขั้นตอนที่ 7 update ค่า weight ของ  $t_{12} = 1 * 1 = 1$ ,  $t_{22} = 1 * 1 = 1$ , และ  $t_{32} = 1 * 0 = 0$  ส่วนค่า  $t_{ij}$  ที่เหลือจะไม่เปลี่ยนแปลง แล้ว update ค่า weight ของ

$$b_{12} = \frac{t_{12}x_1}{0.5 + (t_{12}x_1 + t_{22}x_2 + t_{32}x_3)} = \frac{1}{0.5 + 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 0} = 0.4$$

$$b_{22} = \frac{t_{22}x_2}{0.5 + (t_{12}x_1 + t_{22}x_2 + t_{32}x_3)} = \frac{1}{0.5 + 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 0} = 0.4$$

$$b_{32} = \frac{t_{32}x_3}{0.5 + (t_{12}x_1 + t_{22}x_2 + t_{32}x_3)} = \frac{0}{0.5 + 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 0} = 0$$

ส่วนค่า weight ของ  $b_{ij}$  ที่เหลือจะไม่มีการเปลี่ยนแปลง แล้วไปขั้นตอนที่ 2

ขั้นตอนที่ 2 ได้ input ตัวสุดท้าย (001)

ขั้นตอนที่ 3 คำนวณ

$$y_1 = b_{11}x_1 + b_{21}x_2 + b_{31}x_3 = 0.4 \cdot 0 + 0 \cdot 0 + 0.4 \cdot 1 = 0.4$$

$$y_2 = b_{12}x_1 + b_{22}x_2 + b_{32}x_3 = 0.4 \cdot 0 + 0.4 \cdot 0 + 0 \cdot 1 = 0$$

$$y_3 = b_{13}x_1 + b_{23}x_2 + b_{33}x_3 = 0.25 \cdot 0 + 0.25 \cdot 0 + 0.25 \cdot 1 = 0.25$$

ขั้นตอนที่ 4 เลือก  $j^*$  ที่ node 1

ขั้นตอนที่ 5  $\|x\| = 0 + 0 + 1 = 1$  และ  $t_{11}x_1 + t_{21}x_2 + t_{31}x_3 = 1 \cdot 0 + 0 \cdot 0 + 1 \cdot 1 = 1$

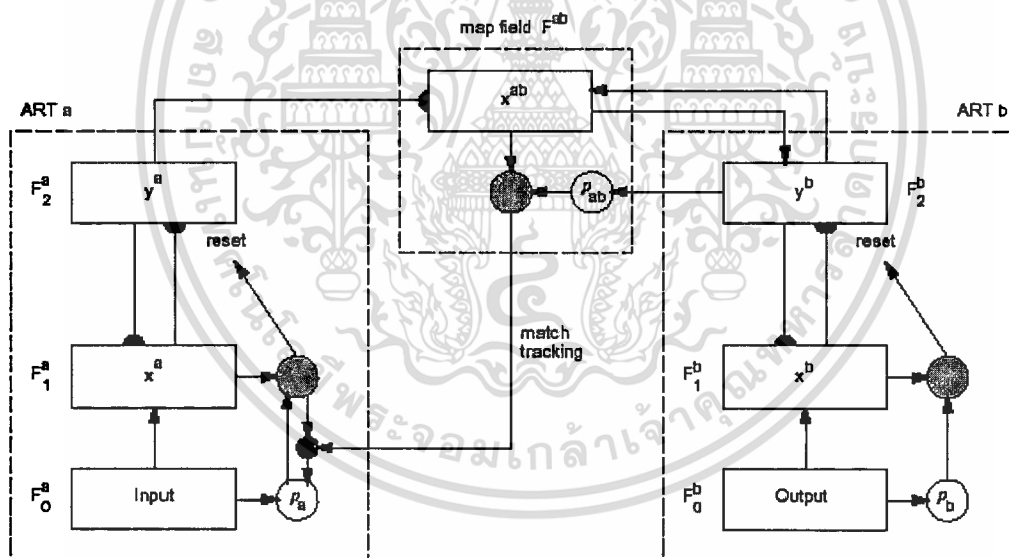
ดังนั้นแล้ว  $\frac{1}{1} = 1 > 0.8$  แสดงว่า input ที่สาม (001) จะอยู่ใน node ที่ 1

ขั้นตอนที่ 6 ไม่มี input เหลือแล้ว ทำการเก็บค่า weight และค่าต่างๆ ออกจากโปรแกรม

ถ้าในกรณีขั้นตอนที่ 5 นั้น คำนวณค่าได้น้อยกว่า Vigilance แล้ว จะทำงานคล้ายกับค้นหาค่าสูงสุดตัวต่อไป มากำหนดในขั้นตอนที่ 5 และสมมติว่าเลือก node ทั้งหมดแล้ว ยังมีค่าน้อยกว่าอีก จะทำการสร้าง node  $j$  ที่ F2 ตัวใหม่ขึ้นมา

### 2.5.3 ARTMAP

หลักการการทำงานของ ARTMAP นั้น จะประกอบด้วย  $ART_a$ ,  $ART_b$  และตัว map ( $F_{ab}$ ) ซึ่งทั้ง  $ART_a$  และ  $ART_b$  นั้นก็คือ  $ART1$  ดังที่กล่าวมาข้างต้น ดังแสดงในภาพที่ 2.17 โดยมีการเพิ่ม Vigilance ( $\rho_{ab}$ ) อีกค่าขึ้นมาสำหรับกรณีที่มี map ระหว่าง  $ART_a$  กับ  $ART_b$  ไม่เจอ และในการพัฒนาระบบงานครั้งนี้ ที่  $ART_b$  นั้น จะไม่ใช่  $ART1$  เพราะว่าการตอบสนองที่ต้องการนั้นได้อยู่ในรูปของ Cluster แล้ว จึงไม่จำเป็นที่  $ART_b$  นั้นต้องทำการ train อีก ดังแสดงในภาพที่ 2.18 ซึ่งเมื่อ  $ART_a$  นั้นทำการ cluster ข้อมูลได้มาแล้ว จะนำเอาผลที่ cluster ไว้แล้วนั้น มา map กับ  $ART_b$  โดยมีลักษณะการ map คล้ายตารางเปรียบเทียบ โดย node ที่ออกจาก  $ART_a$  นั้น จะไม่สามารถจับคู่กับ node ที่ออกจาก  $ART_b$  ได้ แต่ node ที่  $ART_b$  นั้น สามารถ map จับคู่กับ node ที่ออกจาก  $ART_a$  ได้ และเมื่อเกิดเหตุการณ์ไม่สามารถ map ได้ ARTMAP จะทำการเพิ่มค่า  $\rho_{ab}$  ขึ้น โดยค่าที่เพิ่มขึ้นนี้ จะไปเพิ่มให้กับ  $\rho_a$  เพราะถ้าเราไม่เพิ่มค่า  $\rho_a$  แล้ว การ Cluster ที่เกิดจาก  $ART_a$  นั้นจะได้ node เดิมขึ้นมา

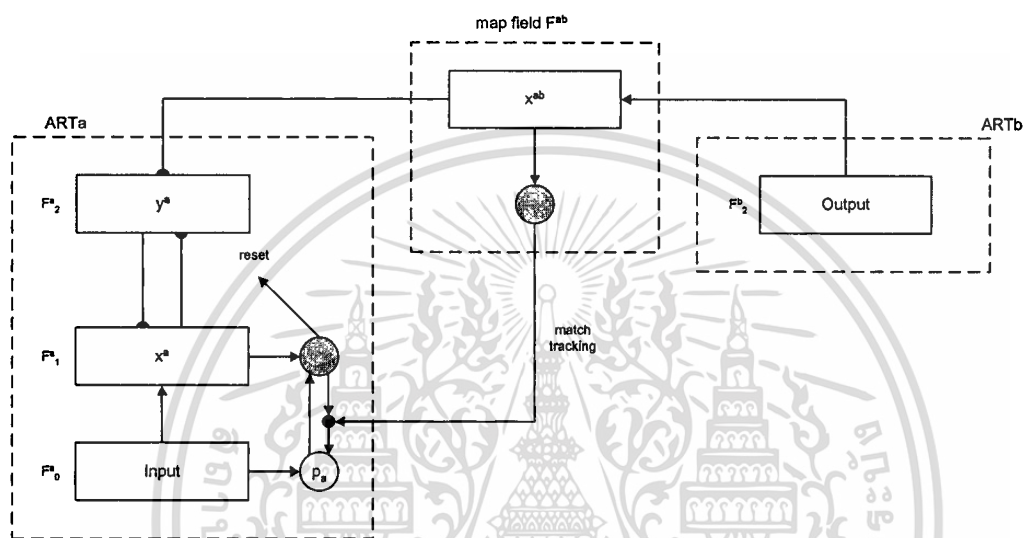


ภาพที่ 2.17 สถาปัตยกรรมพื้นฐานของ ARTMAP

เมื่อ ARTMAP ทำการ Train เสร็จสิ้นแล้ว เมื่อถึงช่วงการ Test ข้อมูล จะทำการถอดเอา  $ART_b$  ออก แล้วนำ input ที่ต้องการ test นั้น เข้าไปใน  $ART_a$  ทำการ Cluster ออกมาว่าไปตกอยู่ใน node ไหน เมื่อทราบ node แล้วจะนำไปเปรียบเทียบกับค่า map ที่เคยได้ทำการ train ไว้ก่อนหน้านี้

แล้วแสดงผลลัพธ์ของการ map ออกมา กรณีที่ นำไปเปรียบเทียบกับค่าที่เคย map เอาไว้ จะแสดงผลลัพธ์ออกมาเป็น Unknow

ซึ่ง ARTMAP จะมีประสิทธิภาพที่สูง และแสดงผลลัพธ์ Unknow ออกมาน้อยๆนั้น จะขึ้นอยู่กับจำนวน input ที่เข้าไป train ดังนั้นยิ่งจำนวน input pattern มากเท่าไร ก็จะแสดงผลลัพธ์ ถูกต้องมากยิ่งขึ้น



ภาพที่ 2.18 สถาปัตยกรรม ARTMAP

### บทที่ 3

#### ระบบงานและการวิเคราะห์ร้านเช่าวิดีโอ

ร้านเช่าวิดีโอซีทาญา เป็นร้านเช่าวิดีโอที่มีถิ่นกำเนิดมาจากประเทศญี่ปุ่นเมื่อปี พ.ศ. 2528 และขยายตัวอย่างรวดเร็วด้วยระบบแฟรนไชส์ ปัจจุบันมีสาขากว่า 1,000 สาขา สมาชิกกว่า 14 ล้านคน และมีรายได้รวมต่อปีมากกว่า 2,000 ล้านบาทหรือร้อยละ 10 ของจีดีพีไทย ด้วยความสำเร็จอย่างมากมาย เครื่องมือซีทาญาจึงได้ขยายธุรกิจสู่ตลาดภายนอกประเทศ และได้จัดตั้ง บริษัท ซีทาญา(ประเทศไทย) จำกัด ขึ้นเมื่อปี พ.ศ. 2540 โดยเปิดดำเนินการธุรกิจให้เช่าวิดีโอภายใต้ชื่อ “ซีทาญา” โดยมีสาขาแรกคือ สาขาเพชรชិเดนท์ ปาร์ค ที่สุขุมวิท 24 ปัจจุบันร้านซีทาญามีสาขามากกว่า 100 สาขาทั่วประเทศไทย

#### 3.1 ลักษณะการดำเนินธุรกิจ

ร้านซีทาญาในประเทศไทย จะเป็นการให้แฟรนไชส์เหมือนที่ประเทศญี่ปุ่น โดยทางบริษัท จะไม่มีการเปิดร้านเช่าเป็นของตัวเอง สาขาที่เปิดนั้นจะเป็นแฟรนไชส์ทั้งหมด เปิดโอกาสให้บุคคลผู้สนใจทั้งที่มีพื้นที่เช่า และไม่มีพื้นที่เช่า สามารถมาเป็นแฟรนไชส์ได้ โดยค่าใช้จ่ายในการเป็นแฟรนไชส์ก็จะแตกต่างกันไปตามขนาดและบริเวณพื้นที่ร้าน ซึ่งลักษณะสินค้าที่ให้บริการนั้น ที่ประเทศไทยมีสินค้าให้บริการน้อยกว่าที่ญี่ปุ่น โดย ณ. ปัจจุบัน ร้านซีทาญาในไทยนั้นจะมี บริการเช่าวิดีโอ, วิดีโอซีดี, ดีวีดี และการขายแผ่นซีดีเพลงและภาพยนตร์, การ์ดโทรศัพท์, นิตยสาร และฟิล์มถ่ายรูป ฯลฯ ในส่วนการบริการให้เช่าวิดีโอ นั้น สมาชิกวิดีโอยังไม่สามารถเช่าหรือคืนต่างร้านที่สมัครเป็นสมาชิกได้ รูปแบบการเช่าวิดีโอจะมีอยู่ 3 แบบ คือ แบบวิดีโอและแบบวิดีโอซีดีนั้น ถ้าเป็นหนังใหม่จะเช่าได้ 1 วัน หนังกลางเก่ากลางใหม่ เช่าได้ 3 วัน และหนังเก่าจะเช่าได้ 7 วัน ส่วนอีกรูปแบบคือ ดีวีดี ถ้าเป็นหนังใหม่เช่าได้ 1 วัน ส่วนหนังเก่าเช่าได้ 3 วัน

ในส่วนการตลาดของบริษัท จะใช้การตลาดในแนวทางเดียวกัน มีการแจก โหมบาย, ใบปิด ภาพยนตร์ และของชำร่วยต่างๆ เพื่อไว้ประดับตามสาขา รวมทั้งการใช้โปรโมชันต่างๆ เข้ามาพร้อมด้วย โดยค่าใช้จ่ายในด้านการตลาดให้ร้าน ทางบริษัทจะหักจากรายได้ 1% ของร้านสาขา และจะมีการตลาดให้เฉพาะแต่ละสาขา ในกรณีที่ สาขานั้นมีรายได้ต่ำ เพื่อให้สาขานั้นอยู่รอดได้ เพราะรายได้ของบริษัทส่วนใหญ่จะได้จากการอยลดี (Royalty) ที่ได้รับจากแฟรนไชส์ สาขาต่างๆ ดังนั้น ถ้าแฟรนไชส์ทำกำไรได้น้อย บริษัทก็จะมียาได้น้อยตามไปด้วย เพื่อให้บริษัทมีรายได้

เพิ่มขึ้น ก็คือการทำให้แฟรนไชส์แต่ละสาขามีรายรับให้มากที่สุด ในขณะที่เดียวกันก็ควรมีรายจ่ายที่น้อยที่สุดด้วย

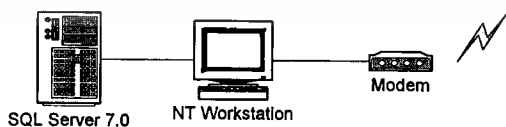
### 3.2 การจัดเก็บข้อมูล

การจัดเก็บข้อมูลต่างๆ ไม่ว่าจะเป็นข้อมูล ลูกค้า, สินค้า หรือ ยอดขาย ต่างๆนั้น แต่ละสาขาจะเป็นผู้จัดเก็บไว้ในฐานข้อมูลของตัวเอง โดยถ้าเป็นข้อมูลของสมาชิก แต่ละสาขาจะต้องทำการป้อนข้อมูล ผ่านทางเครื่อง NT Workstation แล้วจะถูกจัดเก็บไว้ในเครื่อง Server ที่มีระบบจัดการฐานข้อมูลชนิด MS SQL Server 6.5 ดังแสดงในภาพที่ 3.1 ส่วนยอดขายต่างๆนั้น จะผ่านทางเครื่องบาร์โค้ดที่เป็น POS (Point Of Sale) ที่เชื่อมต่อกับเครื่อง Server เดียวกัน แล้วข้อมูลทั้งหมดนั้นก็จะถูกจัดเก็บในเครื่อง Server และในส่วนเครื่อง NT Workstation นั้น ก็จะมี Modem เอาไว้เชื่อมต่อโดยใช้โปรแกรม PC Anywhere



ภาพที่ 3.1 แสดงลักษณะการเชื่อมต่อเครือข่ายในแต่ละสาขา

ในส่วนที่จัดการดูแลข้อมูลสาขาย่อยต่างๆของบริษัทช้ทานุ่ำนั้น มีลักษณะดังภาพที่ 3.2 คือ ข้อมูลจากทุกสาขาย่อยนั้น จะถูกทำการ backup ผ่านโปรแกรม PC Anywhere และใช้ระบบจัดการฐานข้อมูล MS SQL 7.0 การ backup ข้อมูล ส่วนใหญ่จะทำในช่วงปิดร้าน แต่ก็มืบางสาขาที่อาจจะไม่ทำช่วงปิดร้าน เนื่องจาก การเสียค่าโทรศัพท์ทางไกลที่แพง ดังนั้น อย่างน้อยที่สุด บริษัทช้ทานุ่ำน จะทำการ backup ข้อมูลของสาขาย่อยลงในฐานข้อมูลของตัวเองในทุกวันที่ 1 ของเดือน



ภาพที่ 3.2 แสดงลักษณะการเชื่อมต่อเครือข่ายที่ใช้ติดต่อกับแต่ละสาขา

### 3.3 ข้อมูลที่นำมาใช้

ระบบในการจัดเก็บข้อมูลต่าง ๆ นั้น ซืหาญามีการใช้ Table ทั้งหมดจำนวน 26 table ที่ได้ถูกออกแบบเอาไว้แล้วดังภาพที่ 3.3

Create table in Design view	SELCPRIE
AREACD	SHOP
ERRMSG	STAFF
GENRUCD	SUBAREACD
ITEMC0	SUPPLIER
MEMBER	TRANSACTIONDAY
MEMBERDETAIL	TRANSACTIONSUM
MEMBERPRICE	
MEMBERSUM	
NAME	
POSKEY	
PROBLEM	
REJCINT	
REJCNTM	
RENTAL	
RENTALPRICE	
REPORT	
REPORTM	
RETAIL	
SELL	

ภาพที่ 3.3 แสดง Table ทั้งหมดในฐานข้อมูล

แต่ที่เราจะนำมาใช้ในโครงงานส่วนใหญ่จะเป็นการ Select เอาแต่ Attribute ที่มีประโยชน์ในการทำ Data Mining เท่านั้น ซึ่ง Attribute ที่จะใช้ส่วนใหญ่ จะเป็นประวัติสมาชิก และข้อมูลสินค้าต่างๆ ที่ได้ถูกเข้าไปจากสมาชิก ซึ่ง Attribute ที่จะนำมาใช้โครงงานมีดังนี้

- **GENRUCD** เป็นประเภทของหนังซึ่งเก็บไว้เป็นรหัสแต่ละประเภท ซึ่งมีจำนวนมากถึง 107 ประเภท ถึงทำการจัดหมวดหมู่ประเภทให้เหลือ 18 ประเภท ดังนี้ ACTION, DRAMA, COMEDY, HOROR, SCI-FI, THAI, INDIAN, CHINESE, CARTOON, VARIETY, JOKE, SPORT, KARAOKE, KOREAN, MUSIC, DOCUMENTARY, EDUCATION และ JAPAN
- **AGE** แสดงอายุของผู้เช่า โดยจัดเก็บในรูปแบบ yyymmdd และใช้เฉพาะช่วงอายุผู้เช่าตั้งแต่ 10 – 70 ปี
- **GENDER** ระบุเพศของผู้เช่า ชายระบุเป็น 1 หญิงระบุ 2
- **INCOME** ระบุรายได้เฉลี่ยของสมาชิก โดยแบ่งออกเป็น 4 ระดับ คือ

- ระบุ 1 หมายถึง มีรายได้ระหว่าง 5000 -15,000 บาท
  - ระบุ 2 หมายถึง มีรายได้ระหว่าง 15,001 – 30,000 บาท
  - ระบุ 3 หมายถึง มีรายได้ระหว่าง 30,001 – 60,000 บาท
  - ระบุ 4 หมายถึง มีรายได้มากกว่า 60,000 บาท
- **JOB** ระบุอาชีพของสมาชิก แบ่งออกเป็น 5 ประเภท ดังนี้
- ระบุ 1 หมายถึง นักเรียน/นักศึกษา
  - ระบุ 2 หมายถึง พนักงานบริษัท
  - ระบุ 3 หมายถึง ข้าราชการ
  - ระบุ 4 หมายถึง กิจการส่วนตัว
  - ระบุ 5 หมายถึง อื่นๆ

การจัด Pattern ของ input ที่ใช้ใน ART เนื่องจาก ART ต้องการ input ที่อยู่ในรูปไบนารี ดังนั้นจึงมีการทำ input pattern ดังนี้

- **GENRUCD** นั้นเป็น target output ที่อยู่ใน ARTb ดังนั้นจึงไม่มีการแปลงให้เป็นไบนารีใดๆทั้งสิ้น โดยแต่ละประเภทจะแทนด้วยหมายเลขของกลุ่มประเภทรูปนั้นๆ
- **AGE** ใช้บิตจำนวน 3 บิต โดยการแบ่งกลุ่มช่วงอายุคนออกเป็น 5 กลุ่มช่วงอายุ ดังนี้
  - ช่วงอายุ 10 – 20 ปี แทนด้วย 000
  - ช่วงอายุ 21 – 30 ปี แทนด้วย 001
  - ช่วงอายุ 31 – 40 ปี แทนด้วย 010
  - ช่วงอายุ 41 – 50 ปี แทนด้วย 011
  - ช่วงอายุที่มากกว่า 51 ปี แทนด้วย 100
- **GENDER** ใช้ 1 บิต โดยที่ 0 แทน เพศชาย และ 1 แทนเพศหญิง
- **INCOME** รายได้สมาชิกนั้น ได้แบ่งเป็น 4 หมวดแล้ว จึงใช้ 2 บิต ดังนี้
  - รายได้ 5000 – 15,000 บาท แทนด้วย 00
  - รายได้ 15,001 – 30,000 บาท แทนด้วย 01
  - รายได้ 30,001 – 60,000 บาท แทนด้วย 10
  - รายได้มากกว่า 60,001 บาท แทนด้วย 11
- **JOB** ใช้ 3 บิต ดังนี้
  - นักเรียน/นักศึกษา แทนด้วย 000

- พนักงานบริษัท แทนด้วย 001
- ข้าราชการ แทนด้วย 010
- กิจการส่วนตัว แทนด้วย 011
- อื่นๆ แทนด้วย 100

เมื่อได้รูปแบบทั้งหมดแล้ว ก็จะเข้าไปคำนวณใน ART ต่อไป ค่า Vigilance ที่ใช้ปรับ Neural Network ได้กำหนดค่าเริ่มต้นไว้ที่ 0.8 รวมถึง สเตปที่ใช้เพิ่มค่า Vigilance ในกรณีที่มี mismatch ก็กำหนดไว้ที่ 0.01

ซึ่งจะสรุปค่าต่างๆ และรูปแบบที่ใช้ดังตารางต่อไปนี้

ตารางที่ 3.1 แสดงการเปลี่ยนข้อมูล GENDER เป็นข้อมูลไบนารี

เพศ	ข้อมูลใน Database	ไบนารี
ชาย	1	0
หญิง	2	1

ตารางที่ 3.2 แสดงการเปลี่ยนข้อมูล AGE เป็นข้อมูลไบนารี

ช่วงอายุ (ปี)	ไบนารี
10 - 20	000
21 - 30	001
31 - 40	010
41 - 50	011
มากกว่า 51 ปีขึ้นไป	100

ซึ่งการจัดเก็บข้อมูลอายุของสมาชิกนั้นเป็นการจัดเก็บจากวันเดือนปีที่เกิด ซึ่งอายุที่ได้มานั้น ได้มาจากการใช้ฟังก์ชันทางคณิตศาสตร์ที่อยู่ใน SQL Server ทำการแปลงให้เป็นอายุปัจจุบัน

ตารางที่ 3.3 แสดงการเปลี่ยนข้อมูล JOB เป็นข้อมูลไบนารี

อาชีพ	ข้อมูลใน Database	ไบนารี
นักเรียน/นักศึกษา	1	000
พนักงานบริษัท	2	001
ข้าราชการ	3	010
กิจการส่วนตัว	4	011
อื่นๆ	5	100

ตารางที่ 3.4 แสดงการเปลี่ยนข้อมูล INCOME เป็นข้อมูลไบนารี

รายได้	ข้อมูลใน Database	ไบนารี
5,000 – 15,000	1	00
15,001 – 30,000	2	01
30,001 – 60,000	3	10
มากกว่า 60,000	4	11

จากชุดไบนารี ที่เป็น Input pattern สำหรับ ARTa จะมีขนาด 9 บิต ยกตัวอย่างเช่น ข้อมูลสมาชิก ที่เป็นชาย มีอายุ 20 ปี เป็นนักศึกษา และมีรายได้ 5,000 บาท ดังนั้น Input pattern จะได้ 000000000 เป็นต้น

นอกจากนี้แล้ว สังเกตว่าชุด Input pattern บางตัวที่ได้ นั้น มีค่า 0 อยู่จำนวนมาก และบางครั้ง Input pattern ก็มีค่า 1 อยู่มาก ซึ่งค่าแตกต่างกันอย่างมากของชุด Input อาจส่งผลให้ ARTMAP นั้น มีประสิทธิภาพที่ไม่ดีพอ จึงนำหลักการ Complement มาใช้กับชุด Input ด้วย เพื่อให้เฉลี่ยค่ากัน จะส่งผลให้ค่า Input ที่เข้าไปนั้น มีอัตราเฉลี่ยเท่าๆกัน โดยนำเอาชุด Input ที่ได้ นั้น มาต่อกับ Complement ของตัวเอง เช่น

ถ้าค่า Input ที่ได้เป็น 000000000 ก็นำมาทำ 1's Complement ผลที่ได้คือ 111111111 ดังนั้นชุด Input จริงๆ ที่เข้าไปใน ARTa นั้นจะมีขนาด 18 บิต ดังนี้

000000000111111111

ตารางที่ 3.5 แสดงการเปลี่ยนข้อมูล GENRUCD

ประเภท	ข้อมูลใน Database	Target Output
ACTION	0025	1
DRAMA	0028	2
COMEDY	0028L	3
HOROR	0030	4
SCI-FI	0033	5
THAI	0026	6
INDIAN	0027	7
CHINESE	0029	8
CARTOON	060Z	9
VARIETY	0062, 0064	10
JOKE	0063	11
SPORT	0065	12
KARAOKE	0066	13
MUSIC	0067	14
DOCUMENTARY	0069	15
EDUCATION	0061	16
JAPAN	0038	17
KOREAN	0031	18

ซึ่ง Input pattern สำหรับป้อนเข้า ARTb นั้น เป็น Target Output ที่เราต้องการอยู่แล้ว ดังนั้นจึงไม่จำเป็นต้องทำการแปลงข้อมูลให้เป็น ไบนารี เพียงแปลงให้เป็นหมายเลขแทนกลุ่มต่างๆก็พอ

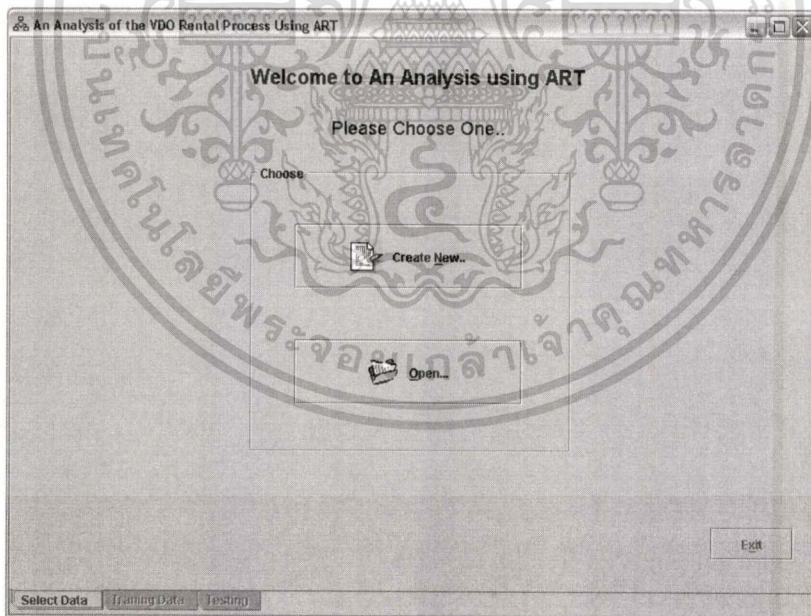
## บทที่ 4

### การพัฒนาโปรแกรม

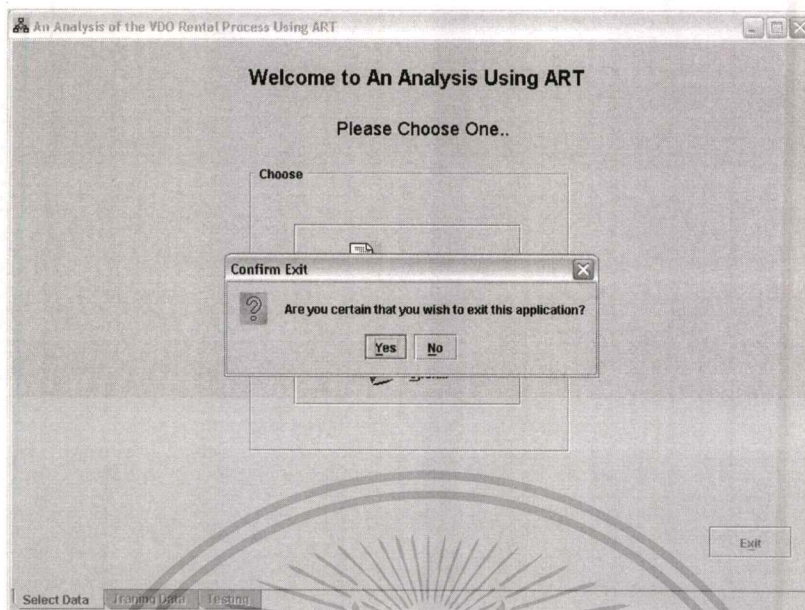
ในบทนี้จะกล่าวถึงการพัฒนาโปรแกรมประยุกต์ ซึ่งประกอบด้วยกระบวนการดังต่อไปนี้  
คือ การเข้าสู่โปรแกรม, การ Train ชุดข้อมูล และการ Test ข้อมูล โดยมีรายละเอียดดังต่อไปนี้

#### 4.1 การเข้าสู่โปรแกรม

เมื่อเริ่มต้นทำงาน โปรแกรมจะเข้าสู่หน้าจอแรก เพื่อให้ผู้ใช้เลือกได้ว่า จะ Train ชุดข้อมูลใหม่ โดยการคลิกที่ปุ่ม “Create New...” หรือ จะทำงาน Test ข้อมูลโดยการเปิดไฟล์ที่เคยได้ save เอาไว้ได้โดยการคลิกที่ปุ่ม “Open...” ดังภาพที่ 4.1 และถ้าต้องการออกจากโปรแกรม ก็สามารถคลิกที่ปุ่ม “Exit” พร้อมกับ คำเตือนว่า ต้องการออกจากโปรแกรมใช่หรือไม่ ดังภาพที่ 4.2



ภาพที่ 4.1 แสดงหน้าจอโปรแกรมเมื่อเริ่มต้น

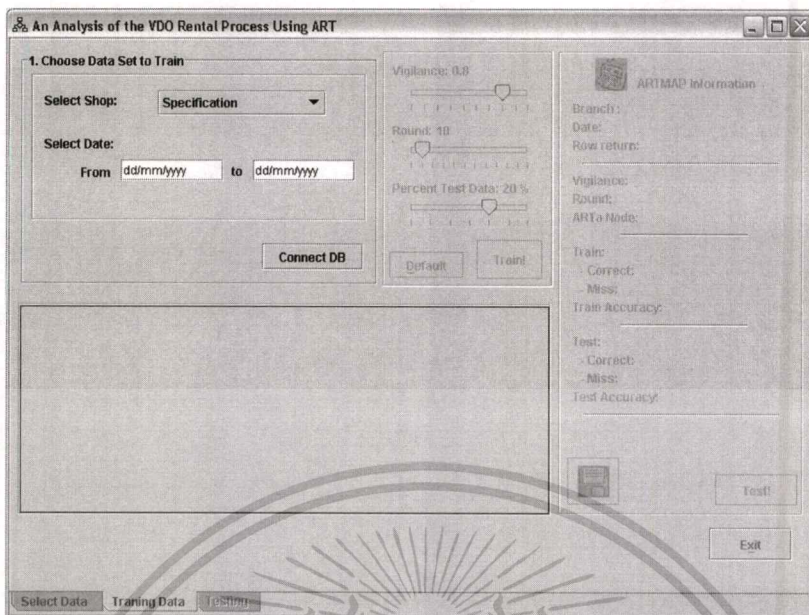


ภาพที่ 4.2 แสดงข้อความเตือนว่าต้องการออกจากระบบหรือไม่

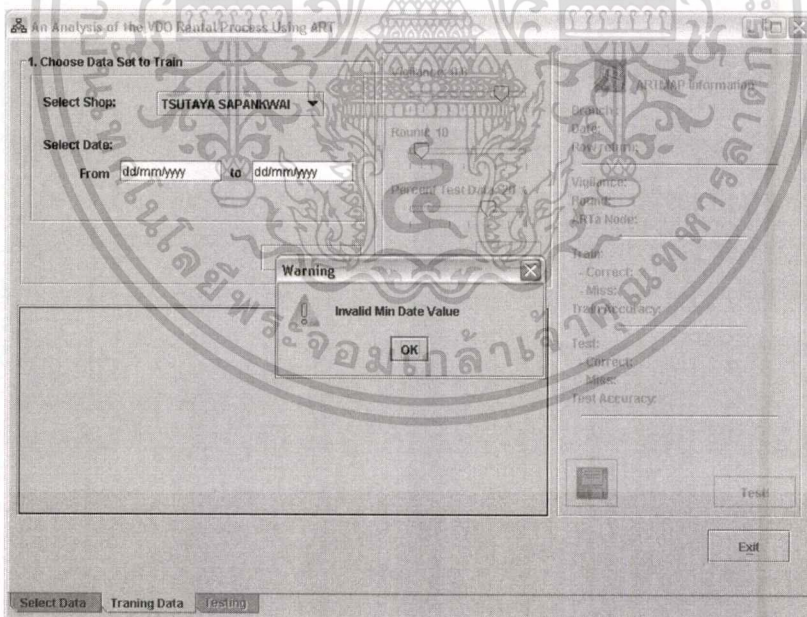
เมื่อผู้ใช้คลิกปุ่ม “Create New...” โปรแกรมจะเข้าสู่หน้าจอ Training Data ซึ่งจะกล่าวในหัวข้อต่อไป

#### 4.2 การ Train ชุดข้อมูล

เมื่อเข้าสู่หน้าจอนี้ ระบบจะทำการรอผู้ใช้ป้อนข้อมูล สาขาของร้านที่ต้องการจะนำมา Train และช่วงเวลาที่ต้องการ ดังแสดงในภาพที่ 4.3 และผู้ใช้จำเป็นต้องระบุสาขาและวันให้ถูกต้อง ซึ่งวันที่จะเริ่มทำการ Train จะต้องน้อยกว่าวันสิ้นสุด ถ้าผู้ใช้กรอกผิด หรือไม่ได้ระบุ แล้วกดปุ่ม Connect ระบบจะมีคำเตือน พร้อมทั้ง เปลี่ยน cursor ไปที่ยังฟิลด์ที่ไม่ได้ระบุ ดังภาพที่ 4.4 เมื่อผู้ใช้กรอกข้อมูลครบ และทำการกดปุ่ม Connect โปรแกรมจะทำการติดต่อกับฐานข้อมูล SQL Server และแสดงผลออกมาดังภาพที่ 4.5 พร้อมทั้งการ Enable ปุ่ม Train เพื่อเตรียมที่จะเข้าสู่การ Train

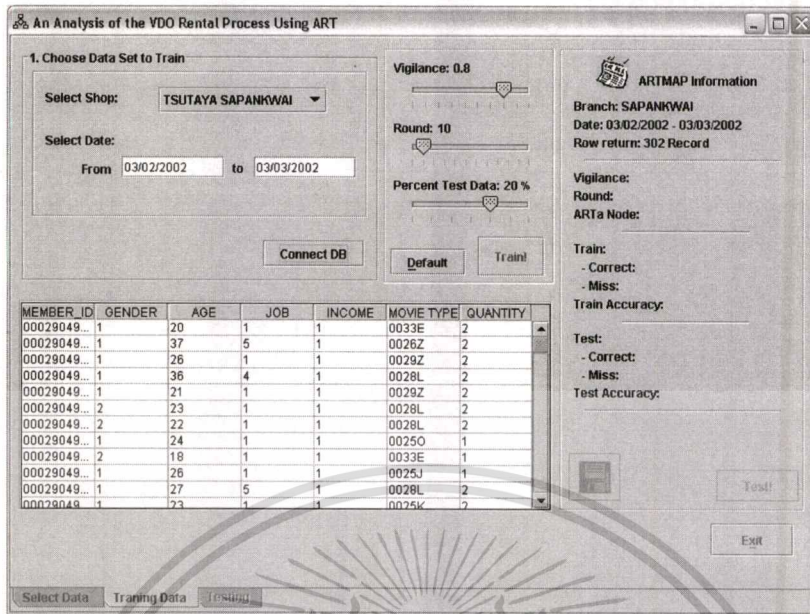


ภาพที่ 4.3 แสดงหน้าจอการ Train ชุดข้อมูล



ภาพที่ 4.4 แสดงการเตือนเมื่อป้อนข้อมูลผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.5 แสดงหน้าจอที่ติดต่อกับฐานข้อมูลแล้ว

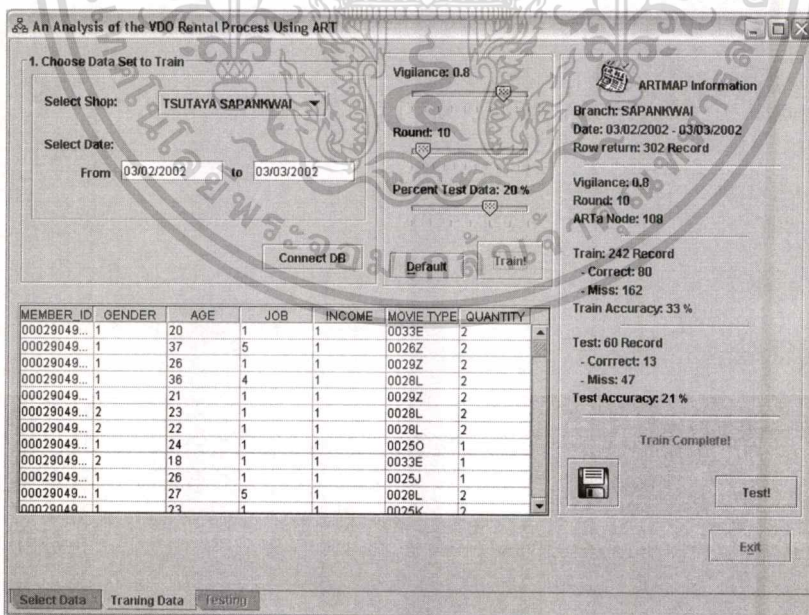
สังเกตว่าปุ่ม Train! จะ enable ขึ้นมาเมื่อเราทำการ connect database เรียบร้อยแล้ว ซึ่งภายในกรอบของการ train นั้น จะมี Slider อยู่ 3 ตัวคือ อันแรกนั้นเป็นการปรับค่า Vigilance ที่ต้องการ โดยสามารถปรับค่าได้ตั้งแต่ค่า 0.1 – 1.0 Slider ที่สองคือ Round เอาไว้ให้เลือกจำนวนรอบในการ train ว่าต้องการทำการ train จำนวนกี่รอบ โดยสามารถที่จะปรับค่าได้ตั้งแต่ 1 – 100 รอบ ส่วน Slider ตัวสุดท้ายคือ Percent Test Data เอาไว้ให้สำหรับปรับเลือกข้อมูลที่จะนำมา test ว่าต้องการจะ test จำนวนกี่เปอร์เซ็นต์จากข้อมูลที่ได้ทำการ connect มา สมมติว่า เมื่อเราทำการ connect ฐานข้อมูลแล้ว และมีจำนวน record ที่ return กลับมา 100 record ถ้าเราเลือก Percent Test Data ไว้ที่ 20 % ก็หมายความว่า ข้อมูล 100 record ที่ได้ นั้น จะเอาแค่ 80% หรือ 80 record เข้าไป train ส่วนอีก 20 % หรือ 20 record นั้น จะเก็บเอาไว้สำหรับทำการ test

จากภาพที่ 4.6 เป็นการแสดงข้อมูลของ ART เมื่อทำการ train เสร็จสิ้น เมื่อการ train นั้น เสร็จสิ้นแล้ว ข้อมูลต่างๆใน ART จะนำมาแสดงผลดังนี้

- Branch: คือ สาขาที่ทำการ train
- Date: คือ ช่วงระยะเวลาที่ได้ทำการเลือกมา
- Row Return: คือจำนวน record ที่ได้
- Vigilance: คือ ค่า Vigilance ที่ได้ตั้งเอาไว้
- Round: คือจำนวนรอบที่ train

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

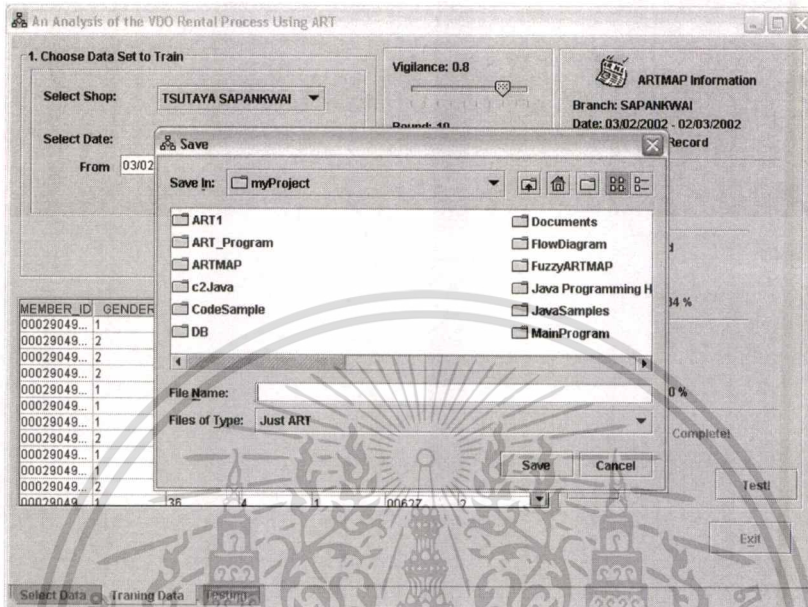
- ARTa Node: คือ จำนวน Cluster ที่ได้จาก ARTa
- Train 242 Record คือเป็นการบอกว่าข้อมูลจำนวน 302 Record นั้น แบ่งสำหรับการ Train 80% จะได้ 242 Record
  - Correct: เป็นการบอกว่าชุดข้อมูลที่นำมา train นั้น เมื่อนำกลับไป test แล้ว ถูกต้อง ก็ record
  - Miss: เป็นการบอกว่าชุดข้อมูลที่นำมา train นั้น เมื่อนำกลับไป test แล้วผิดพลาดก็ record
- Train Accuracy: เป็นการบอกว่าชุดข้อมูลที่ทำการ train แล้วนำกลับมา test นั้นมีความแม่นยำกี่เปอร์เซ็นต์ (ข้อมูลที่ถูกต้อง/ข้อมูลที่นำไป train)
- Test 60 Record คือเป็นการบอกว่าข้อมูลจำนวน 302 Record นั้น แบ่งสำหรับการ Test 20% จะได้ 60 Record
  - Correct: เป็นการบอกว่าชุดข้อมูลที่นำมา test นั้น ถูกต้องก็ record
  - Miss: เป็นการบอกว่าชุดข้อมูลที่นำมา test นั้น ผิดพลาดก็ record
- Test Accuracy: เป็นการบอกว่าชุดข้อมูลที่นำมา test นั้นมีความแม่นยำกี่เปอร์เซ็นต์ (ข้อมูลที่ถูกต้อง/ข้อมูลที่นำไป train)



ภาพที่ 4.6 แสดงข้อมูลเมื่อทำการ Train เสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 4.7 แสดงการ Save ข้อมูล เมื่อต้องการจะเก็บค่าต่างๆ ใน ART เพื่อนำค่า weight นั้นกลับมาใช้ใหม่

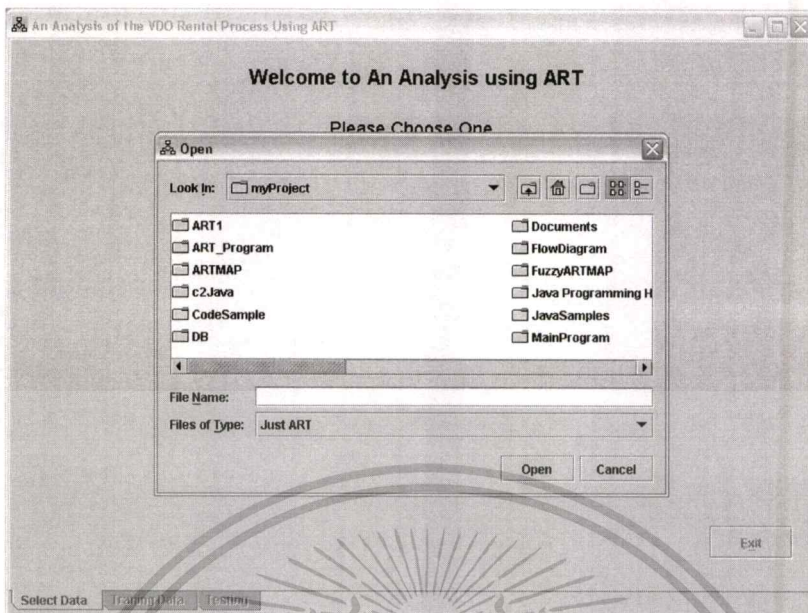


ภาพที่ 4.7 แสดงการ Save ข้อมูล

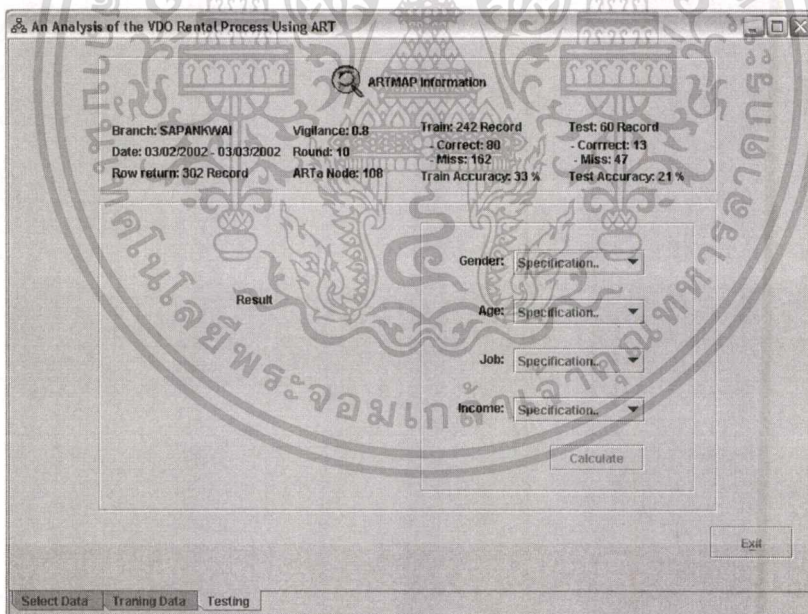
### 4.3 การ Test ข้อมูล

จากภาพที่ 4.1 เมื่อผู้ใช้คลิกปุ่ม Open จะเป็นการให้ผู้ใช้สามารถโหลดค่า weight ของ Neural Network ที่ได้เคยทำการ Train เอาไว้แล้ว เพื่อมาใช้ในการ Classification ข้อมูล ดังภาพที่ 4.8 เมื่อผู้ใช้โหลดค่าที่ได้เคย Train ไว้แล้ว ระบบจะเข้าสู่หน้าจอ Testing Data ดังภาพที่ 4.9 ผู้ใช้ต้องระบุข้อมูลที่จะทำการ Classifies ว่าข้อมูลที่จะให้ Classifies นั้น มีลักษณะอย่างไร ลูกค้ำที่ต้องการแยกแยะนั้นเป็น เพศอะไร อายุเท่าไร มีรายได้เท่าไร และประกอบอาชีพอะไร ถ้าผู้ใช้ไม่ระบุให้ครบ ก็จะมีข้อความเตือนขึ้นมา ดังภาพที่ 4.10 เมื่อผู้ใช้ระบุครบแล้ว แล้วทำการกดปุ่ม Calculate แล้ว โปรแกรมจะทำการคำนวณผลลัพธ์ของประเภทหนึ่งที่มีลักษณะใกล้เคียงกับ input ที่ได้ระบุไป แล้วโปรแกรมจะทำการแสดงผลลัพธ์จากการคำนวณดังภาพที่ 4.11

ซึ่งผลลัพธ์อาจจะแสดง UNKNOWN ขึ้นมาได้ กรณีที่ชุด Input นั้นไม่ครอบคลุมกับข้อมูล Test ดังนั้นจำนวนข้อมูลที่เข้าไป Train ควรมีขนาดที่มากพอ นอกจากเป็นการไม่ให้ ARTMAP ตอบ UNKNOWN แล้ว ยังเป็นการเพิ่มประสิทธิภาพการ Classifies ด้วย

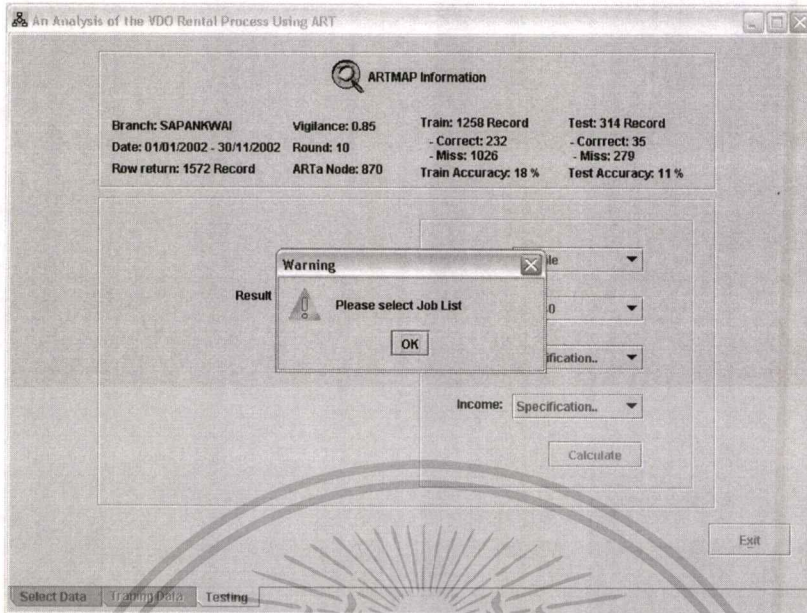


ภาพที่ 4.8 แสดงการ โหลดข้อมูล

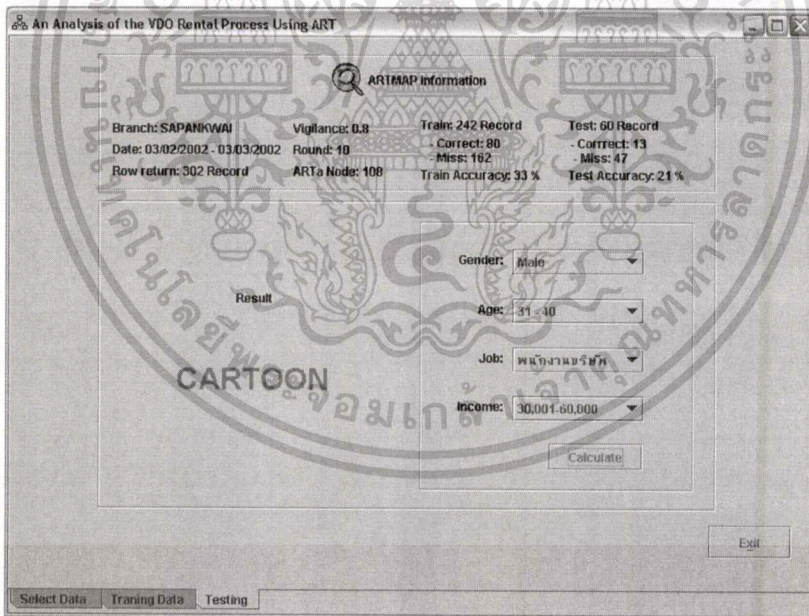


ภาพที่ 4.9 แสดงหน้าจอ Testing Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.10 แสดงข้อความเตือน เมื่อระบุข้อมูลในการ Test ไม่ครบ



ภาพที่ 4.11 แสดงผลลัพธ์จากการคำนวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### ผลการทดลองและบทสรุป

การพัฒนาโครงการนี้ได้ทำการทดลองโปรแกรม ARTMAP ที่ได้พัฒนาขึ้นมา โดยมีผลการทดลองดังนี้

- นำข้อมูลจริงที่มีอยู่ทั้งหมด คือ ตั้งแต่ 01/01/2002 ถึง 30/11/2002 และมีผลการทดลองดังตารางที่ 5.1
  - ข้อมูลจำนวน 1,360 Record
  - Train 20 รอบ
  - Test จำนวน 20%

ตารางที่ 5.1 แสดงผลการทดลองจากข้อมูลทั้งหมด

Vigilance	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85
Yj	129	120	177	166	217	245	271	391
Train Accuracy	10%	11%	12%	14%	12%	14%	13%	18%
Test Accuracy	8%	4%	9%	9%	5%	10%	6%	14%

- นำข้อมูลช่วง 6 เดือน ตั้งแต่ 01/06/2002 ถึง 30/11/2002 มีผลการทดลองดังตารางที่ 5.2
  - ข้อมูลจำนวน 1050 Record
  - Train 20 รอบ
  - Test จำนวน 20%

ตารางที่ 5.2 แสดงผลการทดลองช่วงวันที่ 01/06/2002 – 30/11/2002

Vigilance	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85
Yj	114	110	138	142	181	210	249	339
Train Accuracy	9%	14%	14%	16%	15%	17%	15%	18%
Test Accuracy	5%	6%	8%	5%	8%	9%	8%	11%

- นำข้อมูลช่วง 3 เดือน ตั้งแต่ 01/01/2002 ถึง 31/03/2002 มีผลการทดลองดังตารางที่ 5.3
  - ข้อมูลจำนวน 615 Record
  - Train 20 รอบ
  - Test จำนวน 20%

ตารางที่ 5.3 แสดงผลการทดลองช่วงวันที่ 01/01/2002 – 31/03/2002

Vigilance	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85
Yj	78	79	94	100	117	148	146	212
Train Accuracy	17%	15%	17%	20%	20%	21%	22%	24%
Test Accuracy	5%	7%	16%	12%	7%	9%	14%	14%

- นำข้อมูลช่วง 1 เดือน ตั้งแต่ 01/01/2002 ถึง 31/01/2002 มีผลการทดลองดังตารางที่ 5.4
  - ข้อมูลจำนวน 239 Record
  - Train 20 รอบ
  - Test จำนวน 20%

ตารางที่ 5.4 แสดงผลการทดลองช่วงวันที่ 01/01/2002 – 31/01/2002

Vigilance	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85
Yj	49	47	60	62	69	84	84	117
Train Accuracy	25%	26%	30%	31%	34%	33%	37%	36%
Test Accuracy	17%	14%	10%	4%	8%	8%	6%	8%

จากผลการทดลองนั้น เห็นได้ว่า ค่าความถูกต้องของการ Train และ Test นั้นมีค่าที่ต่ำมาก ซึ่งจะอยู่ในช่วง 25-40% เท่านั้น เนื่องมาจากตัวข้อมูลจากฐานข้อมูลนั้นมีความแตกต่างกันอย่างมาก จึงทำให้ความแม่นยำลดลง ยกตัวอย่างเช่น ข้อมูลลูกค้า 2 คน เป็นเพศชาย ทั้งคู่ต่างมีอายุในช่วงกลุ่มเดียวกัน ประกอบอาชีพเดียวกัน ข้อมูลค่านต่างๆนั้นเหมือนกันหมด แต่เข้าหนึ่งคนละประเภท ด้วยเหตุนี้ ARTMAP จึงให้ผลที่ค่อนข้างผิดพลาด เพราะถ้า pattern ที่ ARTa นั้นเหมือนกัน target output ที่ ARTb ก็ควรก็ให้คำตอบเหมือนกันด้วย ด้วยเหตุนี้เองจึงทำให้ ผลการทดลองที่ได้ในการทำโครงการครั้งนี้จึงมีผลการทดลองที่มีความถูกต้องน้อย ดังนั้นจึงควรที่จะมีการวิจัยและการพัฒนาในขั้นต่อไป เพื่อให้เกิดการทำนายที่แม่นยำและถูกต้องมากขึ้น

และในการพัฒนาระบบวิเคราะห์ข้อมูลร้านเช่าวิดีโอ โดยนำเอา ARTMAP มาประยุกต์ใช้งานทางการทำนายความต้องการของลูกค้ากลุ่มต่างๆนั้น มีความมุ่งหวังให้เกิดประโยชน์ทางด้านการจัดการและการตลาด ซึ่งจากการวิเคราะห์ ออกแบบและดำเนินการปฏิบัติ สามารถสรุปได้ดังนี้

### 5.1 สรุปผลการศึกษาและพัฒนาระบบ

จากการที่นำ Neural Network ชนิด ARTMAP มาประยุกต์ในการทำนายความต้องการของลูกค้า นั้น จะมีประสิทธิภาพสูง และมีความแม่นยำในการทำนายนั้น จำเป็นต้องใช้ ข้อมูลในการ Train ในจำนวนที่มากพอเพียง ซึ่งการทำนายที่มีความถูกต้องมากเท่าไร ก็จะยิ่งช่วยให้ความสามารถในด้านการตลาดของบริษัทสูงมากเท่านั้น และนอกจากนั้น การเจาะกลุ่มลูกค้า กลุ่มต่างๆ ก็จะสามารถเข้าถึงแต่ละกลุ่มได้อย่างถูกต้องและแม่นยำมากขึ้น ซึ่งนอกจากจะเป็นการช่วยให้เกิดความประหยัดกับบริษัทแล้ว ยังช่วยให้บริษัทมีค่าใช้จ่ายที่ลดลง รวมถึงผลตอบแทนที่มากขึ้นด้วย

## 5.2 ข้อเสนอแนะ

สำหรับการพัฒนาระบบงานในทุกๆขั้นตอน จำเป็นต้องศึกษาในรายละเอียดของระบบฐานข้อมูลที่ใช้, วิธีการจัดเก็บ รวมถึงการศึกษา ARTMAP ให้เข้าใจอย่างถ่องแท้ ซึ่งประสิทธิภาพของระบบงานที่พัฒนาขึ้นมา นี้ จะมีประสิทธิภาพที่สูงขึ้น นอกจากวิธีการเขียนโปรแกรมที่มีประสิทธิภาพแล้ว ยังขึ้นอยู่กับฮาร์ดแวร์ที่ใช้ รวมถึงระบบฐานข้อมูลด้วย เนื่องจากตัวระบบงานเป็นการวิเคราะห์ โดยใช้หลักการคำนวณทางคณิตศาสตร์ ดังนั้นยิ่งตัวประมวลผล (ไมโครโปรเซสเซอร์) มีความเร็วมากเท่าไร เวลาในการ train ชุดข้อมูลก็จะใช้เวลาน้อยลงเท่านั้น นอกจากนี้แล้ว การเก็บข้อมูลของลูกค้าให้ถูกต้อง และครบถ้วน ก็จะช่วยเพิ่มประสิทธิภาพต่อระบบงานนี้ด้วย รวมถึงการวิจัยและพัฒนาขั้นต่อไป ในด้านการจัดการข้อมูล การเตรียมข้อมูลที่ถูกต้องและแม่นยำมากขึ้น เพื่อการทำนายผลที่ถูกต้องมากยิ่งขึ้นไปด้วย



## บรรณานุกรม

- กิตติ ภัคดีวัฒนะกุล. 2544. **JAVA ฉบับโปรแกรมเมอร์**. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.  
กรรณิกา สุดกั้งวล. 2542. การแบ่งส่วนฐานข้อมูลด้วย Data mining บทควมวิชาสัมมนา 1  
ภาคเรียนที่ 1. คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีเจ้าคุณทหารลาดกระบัง  
กรุงเทพฯ.
- จุฬารัตน์ เริงหทัยธรรม และคณะ. 2544. เอกสารการสอนชุดวิชาการจัดการระบบฐานข้อมูล.  
นนทบุรี : สำนักพิมพ์มหาวิทยาลัยสุโขทัยธรรมมาธิราช.
- พิชัย กริชกมลกุล. 2544. การวิเคราะห์ฐานเข้าวิดีโอ ด้วยเทคโนโลยี Data Mining  
บทความวิชาสัมมนา 2 ภาคเรียนที่ 1. คณะเทคโนโลยีสารสนเทศ  
สถาบันเทคโนโลยีเจ้าคุณทหารลาดกระบัง กรุงเทพฯ.
- เลียง กุญรัตน์. 2542. **The Adaptive Resonance Theory (ART) Neural Network FAQ site**.  
[Online]. Available: [http://www.ee.psu.ac.th/leang/neural/art\\_faq.htm](http://www.ee.psu.ac.th/leang/neural/art_faq.htm).
- วีระศักดิ์ ชิงถาวร. 2541. **Fundamental of JAVA Programming Volume 1**. กรุงเทพฯ :  
ซัม ซิสเต็ม.
- สมพร จิวรสกุล. 2543. **คู่มือการติดตั้งและใช้งาน Microsoft SQL Server 7.0 ฉบับสมบูรณ์**.  
นนทบุรี : อินโฟเพรส.
- Assadi, H. et al.1992. "Application of the ARTMAP neural network to power system stability  
studies." 1080-1085. IEEE International Conference on Systems, Man and Cybernetics.  
New York: IEEE
- Bartfai, G. 1998. "An adaptive resonance theory-based neural network capable of learning via  
representational redescription." 1137-1142. IEEE International Joint Conference on Neural  
Networks. New York: IEEE Press.
- Bartfai, G. 1996. "On the match tracking anomaly of the ARTMAP neural network."  
**Neural Network**. 9(2): 295-308.
- Cabena, P. et al. 1998. **Discovering Data Mining from Concept to Implement**. New Jersey:  
Prentice Hall.

- Carpenter, A.G. et al. 1991. "ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network." *Neural Network*. 4: 565-588.
- Dagli, C.H. 1994. **Artificial Neural Network for Intelligent Manufacturing**. London: Chapman & Hall.
- Goldstein M. 2001. **Hardcore JFC conquering the swing architecture**. New York: Cambridge University Press.
- Han, J. and Kamber, M. 2001. **Data Mining: Concepts and Techniques**. California: Morgan Kaufmann.
- Kusian, A. 2000. **Computational Intelligence in Design and Manufacturing**. New York: Wiley & Sons.
- Liang, D.Y. 2002. **Rapid Java application development using JBuilder 4/5/6**. New Jersey: Prentice Hall.
- Spell, B. 2000. **Professional Java Programming**. Birmingham: Wrox Press.



## ประวัติผู้เขียน

ชื่อผู้เขียน

นายพิชัย กริชกมลกุล

สถานที่เกิด

กรุงเทพฯ

### ประวัติการศึกษา

วุฒิการศึกษา

อสบ. (อิเล็กทรอนิกส์)

สถานที่สำเร็จการศึกษา

มหาวิทยาลัยสยาม

ปีการศึกษาที่สำเร็จการศึกษา

2538

วุฒิการศึกษา

ปวส. (อิเล็กทรอนิกส์)

สถานที่สำเร็จการศึกษา

มหาวิทยาลัยสยาม

ปีการศึกษาที่สำเร็จ

2535

