

โครงการพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัตรภาษี
กรมบัญชีกลาง

Information System Development for Tax Card Auditing of
Comptroller General's Department



H001594

โดย

นางรัตนาภรณ์ อัสวานุภาพ

รหัส 40067069

อาจารย์ที่ปรึกษา

อาจารย์อิสระ บุรินทรามาตย์

วัน เดือน ปี.....	21 ธ.ค. 2549
เลขทะเบียน.....	01594
เลขเรียกหนังสือ.....	
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2542
คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	โครงการพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัตรภาษีกรมบัญชีกลาง
นักศึกษา	นางรัตนาภรณ์ อัสวณูภาพ
อาจารย์ที่ปรึกษา	อาจารย์อิสระ บุรินทรามาตย์
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2542

บทคัดย่อ

โครงการนี้เป็นโครงการพัฒนาระบบสารสนเทศขึ้นทดแทนระบบงานเดิม ซึ่งเป็นระบบที่ประมวลผลแบบ Batch Processing บนเครื่อง Mainframe สำหรับตรวจสอบบัตรภาษีของกรมบัญชีกลาง และตรวจสอบบัตรภาษีกับใบนำส่งรายได้แผ่นดิน เพื่อให้ได้ข้อมูลที่ถูกต้องอย่างรวดเร็วก่อนนำบัตรภาษีไปใช้ที่กรมศุลกากรและสามารถสืบค้นข้อมูลที่ต้องการได้รวดเร็วยิ่งขึ้น รวมถึงการจัดทำรายงานตามความต้องการของผู้ใช้ โดยมีการศึกษาถึงความเป็นไปได้ในการพัฒนาระบบ การวิเคราะห์และการออกแบบระบบ ตามขบวนการของ SDLC (System Development Life Cycle) และทำการพัฒนาระบบงานบนเครื่องไมโครคอมพิวเตอร์ โดยใช้ Microsoft Visual Basic

Title Information System Development for Tax Card Auditing of
Comptroller General's Department

Student Mrs. Rattanaporn Ussavanuphap

Advisor Mr. Issra Burintramart

Level of Study Master of Science in Information Technology

Major Information Science

Academic Year 1999



ABSTRACT

The purpose of this project is to develop an existing system replacement of Comptroller General's Department which uses batch processing on mainframe Computer to be more accurate and faster . The system covers Tax Card auditing, checking the items in Tax Card with State Revenue Slip, querying and retrieving the data, and reporting. This project is developed using SDLC (System Development Life Cycle) and Microsoft Visual Basic on PC.

กิตติกรรมประกาศ

ในการจัดทำโครงการพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัตรภาษี กรมบัญชีกลางนี้สำเร็จได้ด้วยความช่วยเหลือและสนับสนุนจากหลายฝ่ายเป็นอย่างดี ผู้จัดทำใคร่ขอขอบพระคุณบุคคลดังต่อไปนี้

1. บิดา มารดา สามี และบุตรที่คอยห่วงใยและให้กำลังใจตลอดระยะเวลาที่ศึกษาเล่าเรียน
2. อาจารย์อิสระ บุรินทรามาตย์ ที่ให้คำแนะนำและให้คำปรึกษา
3. ผู้อำนวยการศูนย์สารสนเทศ กรมบัญชีกลาง ที่สนับสนุนและเปิดโอกาสให้ศึกษา
4. เพื่อนข้าราชการ ในกรมบัญชีกลางที่คอยช่วยเหลือและให้กำลังใจ
5. เพื่อน ๆ ร่วมรุ่น IS4 ทุกคน



สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ.....	IV
สารบัญภาพ.....	VI
สารบัญตาราง	VIII
บทที่	
1. บทนำ.....	1
1.1 ความเป็นมา.....	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขั้นตอนการดำเนินการโครงการ.....	1
1.4 เป้าหมายของโครงการ.....	2
1.5 ขอบเขตของโครงการ.....	2
2. ทฤษฎีและหลักการ.....	3
2.1 วัฏจักรของการพัฒนาระบบ	3
2.2 Structured Methodologies.....	4
2.3 Data Flow Diagram.....	5
2.4 ฐานข้อมูล	7
2.5 การนอร์มัลไลซ์.....	22
3. การวิเคราะห์ระบบ	25
3.1 ศึกษาความเป็นไปได้.....	25
3.2 ระบบงานปัจจุบัน.....	28
3.3 การพัฒนาทางเลือกและประเมินทางเลือก	34
3.4 ระบบงานใหม่	36

4. การสร้าง Application	54
4.1 ฮาร์ดแวร์	54
4.2 ซอฟต์แวร์	54
4.3 การออกแบบหน้าจอ	55
4.4 การพัฒนาโปรแกรม.....	57
5. บทสรุป	73
5.1 ผลการศึกษา	73
5.2 ความสามารถของระบบที่พัฒนาขึ้น	73
5.3 ปัญหาและข้อเสนอแนะ	74
ภาคผนวก ก แนวทางการพัฒนาแอปพลิเคชันบนวินโดวส์ด้วย Visual Basic	75
ก.1 ความรู้เบื้องต้นเกี่ยวกับ Visual Basic	75
ก.2 ดาต้าเบสออบเจกต์ (Database Object).....	76
ก.3 Dynaset Object	78
ก.4 Snapshot Object.....	79
ภาคผนวก ข ปัญหาปี ค.ศ. 2000.....	81
ข.1 สาเหตุของปัญหา	81
ข.2 ปัญหาที่เกี่ยวข้องกับปี ค.ศ. 2000.....	81
ข.3 ปัญหาปี ค.ศ. 2000 กับระบบคอมพิวเตอร์	82
ข.4 แนวทางการแก้ปัญหา Y2K	85
ข.5 การแก้ปัญหาปี ค.ศ. 2000	86
ข.6 แนวทางต่าง ๆ ที่สามารถทำได้.....	87
บรรณานุกรม.....	89
ประวัติผู้เขียน	90

สารบัญภาพ

หน้า

ภาพที่

2.1 แสดงระดับของฐานข้อมูล 3 ระดับ	8
2.2 Schema ในฐานข้อมูล 3 ระดับ.....	10
2.3 โครงสร้างของระบบฐานข้อมูล.....	17
3.1 ตัวอย่างบัตรภาษีมูลค่า 1000 บาท.....	26
3.2 ตัวอย่างบัตรภาษีไม่ระบุราคา	26
3.3 ตัวอย่างด้านหลังบัตรภาษี.....	26
3.4 ตัวอย่างใบนำส่งรายได้แผ่นดิน	27
3.5 Work Flow ของระบบงานปัจจุบัน	29
3.6 Existing System Context Diagram	30
3.7 ขั้นตอนการทำงานของระบบปัจจุบัน	32
3.8 DFD Level 0 Existing System.....	33
3.9 The New System Context Diagram.....	36
3.10 DFD Level 0 of The new System.....	37
3.11 DFD Level 1 –Process 2 of The new System.....	38
3.12 DFD Level 1 –Process 3 of The new System.....	39
3.13 DFD Level 1 –Process 4 of The new System.....	40
3.14 The E-R Diagram of The New System	50
3.15 The Relationship of The New System	51
4.1 แสดง Menu Structure	55
4.2 แสดงหน้าจอการเข้าใช้ระบบ	58
4.3 แสดงรายการในเมนูหลัก	58
4.4 แสดงรายการในเมนูหลัก	59
4.5 แสดงหน้าจอ Import ข้อมูลบัตรภาษี	60
4.6 แสดงหน้าจอ Export ข้อมูลบัตรภาษี	60

ภาพที่	หน้า
4.7 แสดงหน้าจอบันทึกข้อมูลผู้ใช้	61
4.8 แสดงหน้าจอการกำหนดสิทธิ์ผู้ใช้	61
4.9 แสดงหน้าจอการเปลี่ยนรหัสผ่าน	62
4.10 แสดงหน้าจอสำหรับบันทึกข้อมูลเขตท้องที่	62
4.11 แสดงหน้าจอสำหรับบันทึกข้อมูลหน่วยงาน	63
4.12 แสดงหน้าจอสำหรับบันทึกข้อมูลหมวดรายได้	63
4.13 แสดงหน้าจอสำหรับบันทึกข้อมูลชนิดบัตร	64
4.14 แสดงหน้าจอสำหรับเพิ่มใบนำส่ง	64
4.15 แสดงหน้าจอสำหรับแก้ไขใบนำส่ง	65
4.16 แสดงหน้าจอสำหรับลบข้อมูลใบนำส่ง	65
4.17 แสดงหน้าจอตรวจสอบบัตรภาษี	66
4.18 แสดงหน้าจอตรวจสอบบัตรภาษีแล้วถูกต้อง	66
4.19 แสดงหน้าจอตรวจสอบบัตรภาษีแล้วไม่ถูกต้อง	67
4.20 แสดงหน้าจอสำหรับสั่งทำรายงานรายวัน	67
4.21 แสดงหน้าจอสำหรับสั่งทำรายงานรายปี	69
ก. แสดงคอนโทรลพื้นฐานของวิซวลเบสิก	76
ข. ผังแสดงแนวทางแก้ไข Y2K	85

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงสัญลักษณ์ต่าง ๆ ที่ใช้ในการทำ DFD	6
2.2 ตัวอย่างฐานข้อมูลแบบ โมเดลเชิงสัมพันธ์	11
2.3 รีเลชัน R1	13
2.4 รีเลชัน R2	13
2.5 รีเลชัน R	14
2.6 ตัวอย่างความสัมพันธ์ระหว่างนักเรียนกับตู้ปกรอง	20
2.7 ตัวอย่างความสัมพันธ์ระหว่างเลขที่บัตรภาษีกับเลขที่คลังรับ	21
2.8 ตัวอย่างความสัมพันธ์ระหว่างนักเรียนกับตู้ปกรอง	21
2.9 แสดงการนอร์มัลไลซ์ข้อมูลบัตรภาษี	24
3.1 แสดง I/O Description	41
3.2 Data dictionary for The Taxcard Database	51
3.3 Data dictionary for The Taxcard Database	53
4.1 รายงานใบนำส่ง.....	68
4.2 รายงานทะเบียนคุมบัตรภาษี.....	68
4.3 รายงานสรุปการนำส่งรายได้แยกตามหมวดรายได้	69
4.4 รายงานสรุปบัตรภาษีค้างชำระ	70
4.5 รายงานสรุปการใช้บัตรภาษี	70
4.6 รายงานสรุปบัตรภาษีแยกตามผู้ประกอบการ	71
4.7 รายงานสรุปบัตรภาษีแยกตามชนิดบัตรภาษี.....	71
4.8 รายงานเปรียบเทียบการใช้บัตรและบัตรค้างชำระ	72

บทที่ 1

บทนำ

1.1 ความเป็นมา

กรมบัญชีกลาง สังกัดกระทรวงการคลัง มีหน่วยงานกระจายอยู่ทั้งส่วนกลางและส่วนภูมิภาค ซึ่งมีหน้าที่ในการควบคุมการรับจ่ายเงินของประเทศ และได้นำระบบคอมพิวเตอร์มาใช้ในการปฏิบัติงานแทนการทำงานด้วยมือเกือบทั้งหมด และได้จัดเก็บข้อมูลทั้งรายได้และรายจ่ายของประเทศไว้ในฐานข้อมูลของกรม เพื่อนำมาใช้ในการจัดทำบัญชีแผ่นดิน ซึ่งบัตรภาษีในส่วนที่ผู้ประกอบการนำมาชำระแทนการใช้เงินสดถือว่าเป็นรายได้ทางหนึ่ง จำเป็นต้องมีการตรวจสอบให้ถูกต้องก่อนการนำเงินรายได้ชนิดนี้ส่งคลัง

ตั้งแต่อดีตจนถึงปัจจุบันบัตรภาษีมีเป็นจำนวนมาก การตรวจสอบด้วยมือต้องใช้ระยะเวลาและใช้บุคลากรจำนวนมาก ตลอดทั้งมีโอกาสเกิดความผิดพลาดและความซ้ำซ้อน ในการนำบัตรภาษีมาใช้ได้ง่าย เพื่อแก้ปัญหาจึงได้นำระบบคอมพิวเตอร์มาช่วยในการตรวจสอบและจัดทำรายงาน ซึ่งรายงานและข้อมูลที่ได้จากระบบจะถูกจัดส่งไปที่กรมศุลกากร เพื่อใช้อย่างอิงและป้องกันการนำบัตรภาษีกลับมาใช้ซ้ำของผู้ประกอบการส่งสินค้าออก

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัตรภาษีได้รวดเร็วยิ่งขึ้น
- 1.2.2 เพื่อตอบสนองต่อความต้องการของผู้ใช้ได้เร็วยิ่งขึ้น
- 1.2.3 เพื่อแก้ปัญหา Y2K ของระบบปัจจุบัน

1.3 ขั้นตอนการดำเนินการโครงการ

- 1.3.1 ศึกษาระบบงานเดิม
- 1.3.2 ศึกษาถึงความเป็นไปได้ในการพัฒนาระบบ
- 1.3.3 วิเคราะห์และออกแบบระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.4 ศึกษาการพัฒนาระบบงานบนเครื่อง PC โดยใช้ Microsoft Access และ Microsoft Visual Basic

1.3.5 เขียนโปรแกรมและทดสอบระบบ

1.4 เป้าหมายของโครงการ

- 1.4.1 ระบบงานใหม่ต้องมีความรวดเร็วในการตรวจสอบบัญชีและง่ายต่อการปฏิบัติงาน
- 1.4.2 ได้ระบบที่สามารถตอบสนองต่อความต้องการของผู้ใช้ได้อย่างรวดเร็ว
- 1.4.3 สามารถสืบค้นข้อมูลบัญชีได้รวดเร็วยิ่งขึ้น
- 1.4.4 ได้ระบบงานที่ไม่มีปัญหา Y2K

1.5 ขอบเขตของโครงการ

โครงการพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัญชี มีขอบเขตของการศึกษาวิเคราะห์ ออกแบบ และการเขียนโปรแกรม โดยจะครอบคลุมส่วนต่างๆ ดังนี้

- 1.5.1 การตรวจสอบบัญชี
- 1.5.2 การบันทึกและตรวจสอบใบนำส่งรายได้แผ่นดิน
- 1.5.3 การตรวจสอบบัญชีกับใบนำส่งรายได้แผ่นดิน
- 1.5.4 การจัดทำรายงาน

บทที่ 2

ทฤษฎีที่ใช้และหลักการ

ในการวิเคราะห์และออกแบบระบบสารสนเทศสำหรับตรวจสอบบัตรภาษีกรมบัญชีกลางนั้นได้มีการนำทฤษฎีในการออกแบบต่างๆ มาช่วยในการทำงาน ซึ่งรายละเอียดของทฤษฎีต่างๆ มีดังต่อไปนี้

2.1 วัฏจักรของการพัฒนาระบบ (System Development Life Cycle : SDLC)

SDLC เป็น Methodology ที่ใช้ในการวิเคราะห์พัฒนาโครงสร้างของระบบ ซึ่งได้มีการรวมหลักการไว้เป็น 3 Phase ดังนี้

Phase 1 : การวิเคราะห์ระบบ (System Analysis) เป็นระบบปัจจุบัน

Phase 2 : รายละเอียดในการวิเคราะห์และออกแบบ (Detail Analysis and Design)

Phase 3 : การสร้างระบบ (Implementation)

ภายในแต่ละ Phase จะประกอบไปด้วยขั้นตอนต่างๆ ดังนี้

Phase ที่ 1 : System Analysis ใน Phase นี้จะประกอบด้วย 4 ขั้นตอนคือ

ขั้นตอนที่ 1 : การกำหนดปัญหา (Problem Definition) เป็นการกำหนดขอบเขตของปัญหา สถานการณ์ต่างๆ และจุดมุ่งหมาย ซึ่งสิ่งที่จะได้จากขั้นนี้คือ

- การรวบรวมและตรวจสอบ วัตถุประสงค์ และเป้าหมาย
- อธิบายปัญหา
- Context Diagram
- กำหนดขอบเขตของการศึกษา
- เริ่มทำ Data Dictionary

ขั้นตอนที่ 2 : วางแผนการศึกษาระบบ (Plan for Study) ซึ่งเป็นขั้นตอนของการวางแผนงานว่างานที่จะต้องทำมีอะไรบ้าง และเป็นอย่างไร

ขั้นตอนที่ 3 : ข้อมูลทั่วไปภายใต้ขอบเขตที่ศึกษาและความสัมพันธ์ในข้อมูลนั้นเริ่มจากการมองภาพกว้าง ๆ ของสิ่งแวดล้อมที่ต้องการศึกษา และพยายามที่จะวิเคราะห์ข้อมูลที่อยู่ภายในนั้นออกมา (General Information and Its Interactive)

ขั้นตอนที่ 4 : ทำความเข้าใจในระบบปัจจุบัน (Understanding the Existing System)

Phase ที่ 2 : Detail Analysis and Design ใน Phase นี้ประกอบด้วย 4 ขั้นตอน คือ

ขั้นตอนที่ 5 : กำหนดความต้องการของระบบใหม่ (Define New System Requirement) เป็นการวิเคราะห์ความต้องการข้อมูลของ User ที่มีต่องานที่ทำอยู่ การวิเคราะห์จะต้องพยายามที่จะเข้าใจว่า ข้อมูลอะไรที่ User ต้องการที่จะใช้ในการทำงาน โดยข้อมูลทั้งหมดในระบบนี้เราจะเรียกว่า Specification ซึ่งในขั้นตอนนี้จะเป็นการวิเคราะห์โครงสร้างที่ใช้ในการตัดสินใจ จำพวก ข้อจำกัด ข้อกำหนด และกฎต่าง ๆ จะเป็นขั้นตอนการเตรียมเสนอ ข้อเสนอของระบบ

ขั้นตอนที่ 6 : การออกแบบระบบใหม่ (Designing New System) ขั้นตอนการออกแบบนี้เป็นการรวบรวมข้อมูลต่าง ๆ เพื่อที่จะนำข้อมูลเหล่านี้ไปออกแบบเพื่อให้ระบบข้อมูลมีความถูกต้องที่สุด

ขั้นตอนที่ 7 : การออกแบบการควบคุมระบบใหม่ (Design New System Controls)

ขั้นตอนที่ 8 : การเปรียบเทียบค่า (Economic Cost Comparison) จะเป็นการวิเคราะห์ค่าและประโยชน์ของทรัพยากรของระบบ

ขั้นตอนที่ 9 : การขายระบบ (Selling System) ขั้นตอนนี้จะเป็นขั้นตอนของการนำเสนอผลงานที่เราได้ทำการออกแบบและวิเคราะห์

Phase ที่ 3 : Implementation ใน Phase นี้มี 1 ขั้นตอน คือ

ขั้นตอนที่ 10 : การทำระบบใหม่หลังจากที่ได้ทำการวิเคราะห์และออกแบบแล้ว ในขั้นตอนนี้จะมีการทำโปรแกรม (Coding) การทดสอบโปรแกรม การบำรุงรักษาระบบ และการฝึกอบรมการใช้ระบบต่อไป

2.2 Structured Methodologies

Structured Methodologies เป็นวิธีการสร้างระบบสารสนเทศที่แสดงให้เห็นถึงการออกแบบระบบที่มีขั้นตอนตามลำดับอย่างเหมาะสม, ถูกวิธี และมีเอกสารประกอบที่เหมาะสม

- ความจำเป็นในการใช้ Structured Methodologie

การใช้ Structured Methodologies นั้นทำให้เกิดการพัฒนาทางด้านการสื่อสารระหว่างนักวิเคราะห์ด้วยกันเองและระหว่างนักวิเคราะห์กับผู้ใช้ เมื่อมีการนำ Structured Methodologies มาใช้อย่างเหมาะสมจะทำให้เกิดระบบที่มีความผิดพลาดน้อยที่สุดและยังสามารถผลิตเอกสารที่ทำให้เกิดความเข้าใจ, กระชับ, และสามารถเข้าไปหาได้และท้ายที่สุดยังช่วยในการหาคำตอบที่สามารถแก้ปัญหาในระบบให้กับผู้ใช้ได้อย่างมีประสิทธิภาพ

- Top - Down Approach : Functional Decomposition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาด้านนี้ เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




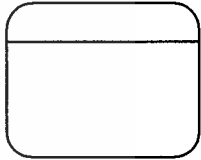


หลักการของ Structured Methodologies คือ แนวความคิด Top - Down ซึ่งกำหนด Hierarchy ขององค์ประกอบภายในระบบโดยที่องค์ประกอบต่างๆสามารถแบ่งย่อยเป็นหน่วยต่างๆ และแต่ละหน่วยก็สามารถแบ่งออกเป็นหน่วยย่อยได้อีก ซึ่งเรียกกระบวนการนี้ว่า Functional Decomposition และเรียกหน่วยที่มีหน้าที่เฉพาะอย่างในระบบว่า Module โดยงานในระดับบนจะครอบคลุมงานที่อยู่ภายใต้ทั้งหมด

2.3 Data Flow Diagram (DFD)

แนวทางการออกแบบระบบงานวิธีหนึ่งที่นิยมกันอย่างแพร่หลายและสามารถแสดงความสัมพันธ์ในรายละเอียดได้ชัดเจนคือ วิธีการออกแบบโดยใช้แผนภาพแสดงทิศทางการไหลของข้อมูล (Data Flow Diagram หรือ DFD) เป็นแผนภาพที่จะแสดงเฉพาะว่ามีงานอะไรบ้างที่ทำอยู่ในระบบงานโดยไม่สนใจว่างานนั้น ๆ มีวิธีการอย่างไร นอกจากนั้นยังแสดงให้เห็นถึงข้อมูลที่อยู่รอบ ๆ ระบบและมีข้อมูลอะไรบ้างที่ถูกเก็บไว้ในระบบ ซึ่งจะใช้สัญลักษณ์ต่าง ๆ ดังแสดงในตาราง 2.1 ส่วนประกอบของ DFD

- Process เป็นส่วนประกอบของ DFD ที่แสดงถึง ส่วนของงานในระบบ ที่มี Input เข้ามาใน Process แล้วมีการเปลี่ยนแปลง Input เป็น Output ส่งออกไปจาก Process สัญลักษณ์ที่ใช้แทน Process มีหลายสัญลักษณ์ด้วยกัน ได้แก่ วงกลม, สี่เหลี่ยม, สี่เหลี่ยมมุมโค้ง
- Flow เป็นส่วนประกอบที่แสดงถึง เส้นทางเข้า หรือออกของข้อมูล หรือเอกสาร ดังนั้น flow จะอธิบายถึงการเคลื่อนไหวของข้อมูลต่าง ๆ สัญลักษณ์ที่ใช้แทน flow คือเส้นตรงมีหัวลูกศร
- Store เป็นส่วนประกอบที่แสดงถึง การเก็บรวบรวมข้อมูล หรือเป็นที่พักของข้อมูล สัญลักษณ์ที่ใช้แทน Store อาจมีได้หลายรูปแบบด้วยกันคือ เส้นตรงสองเส้นขนานกัน, เส้นตรงสองเส้นขนานกันปิดท้ายด้วยเส้นตรงด้านหนึ่ง หรือ เส้นตรงสองเส้นขนานกันปิดท้ายทั้งสองด้านด้วยเส้นโค้ง
- External Entities เป็นส่วนประกอบที่หมายถึง บุคคล, กลุ่มของบุคคล, หน่วยงาน หรือองค์กรต่าง ๆ ที่อยู่ภายนอกระบบที่เรากำลังสนใจวิเคราะห์ และออกแบบ หรือ อยู่ภายนอกการควบคุมของระบบ สัญลักษณ์ที่ใช้แทน External Entities มีหลายสัญลักษณ์ด้วยกัน ได้แก่ วงรี, สี่เหลี่ยม

ตารางที่ 2.1 แสดงสัญลักษณ์ต่างๆ ที่ใช้ในการเขียน DFD

ประเภท DFD	สัญลักษณ์
Source / Destination	
Process	  
Data Store	
Data Flow	

- ข้อดีของการเลือกใช้ DFD

DFD ช่วยให้นักวิเคราะห์สามารถสรุปข้อมูลที่เกี่ยวข้องกับระบบ เข้าใจถึงปัจจัยสำคัญของระบบและระบุส่วนต่าง ๆ ของการทำงานที่ซับซ้อน เข้าใจถึงความสัมพันธ์ระหว่างส่วนต่าง ๆ ของระบบและการประกอบกันเป็นระบบ พัฒนาระบบได้อย่างมีประสิทธิภาพ

DFD เป็นเอกสารร่วมที่ช่วยให้นักวิเคราะห์และผู้ใช้สามารถเข้าใจระบบ และตรวจสอบความถูกต้องได้ทั้ง 2 ฝ่าย

ในการตรวจสอบเรื่องเวลาที่ใช้ในแต่ละขบวนการนั้น นักวิเคราะห์สามารถใช้ DFD เป็นเครื่องมือช่วยในการทราบถึงขอบเขตในการพัฒนารูปแบบระบบ

ลำดับขั้นการจัดการทำแผนภาพการไหลของข้อมูล

- 1) ให้เขียนแผนภาพโดยใช้หลักการเขียนจากบนลงล่าง (Top down) หรือจากใหญ่ไปเล็ก โดยจะต้องรู้ว่าหน่วยงานภายนอกที่เกี่ยวข้องกับระบบทั้งหมดมีอะไรบ้าง ข้อมูลจะเคลื่อนไปในระบบอย่างไร และจะออกจากระบบอย่างไร จะมีเพิ่มข้อมูล

เอกสารนี้เป็นเอกสารที่...
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) เขียนแผนภาพพื้นฐานขึ้นมาก่อน ซึ่งก็คือ แผนภาพระดับ Context Diagram และ ทบทวนว่าได้ครอบคลุมระบบงานที่กำลังทำอยู่หรือไม่ จากนั้นค่อยทำแผนภาพย่อยเพื่อประกอบต่อไป
- 3) บันทึกรายละเอียดในแต่ละขั้นตอน โดยอาจมีหมายเหตุเพิ่มเติม เพื่อให้ผู้อ่านเข้าใจได้ง่าย
- 4) ทบทวนว่าการบันทึกรายละเอียดนั้น ทุกคำได้บันทึกไปมีความหมายชัดเจนดีแล้ว หรือยัง หากว่ายังไม่ชัดเจนดี ควรทำการแก้ไขให้ชัดเจนยิ่งขึ้น

ในการแสดงขั้นตอนการทำงานของระบบงาน และนอกจากนี้ยังมีข้อควรระวังซึ่งก่อให้เกิดในการออกแบบระบบงาน 3 ประการดังมีรายละเอียดคือ

แบลคโฮล (Black Hole) คือ ขั้นตอนการทำงาน (Process) ที่มีแต่ข้อมูลเข้า (Input) แต่ไม่มีผลลัพธ์ (Output) ของการทำงานออกมา

มิราเคิล (Miracle) คือ ขั้นตอนการทำงาน (Process) ที่ไม่มีข้อมูลเข้า (Input) แต่กลับมีผลลัพธ์ (Output) ของการทำงานออกมา

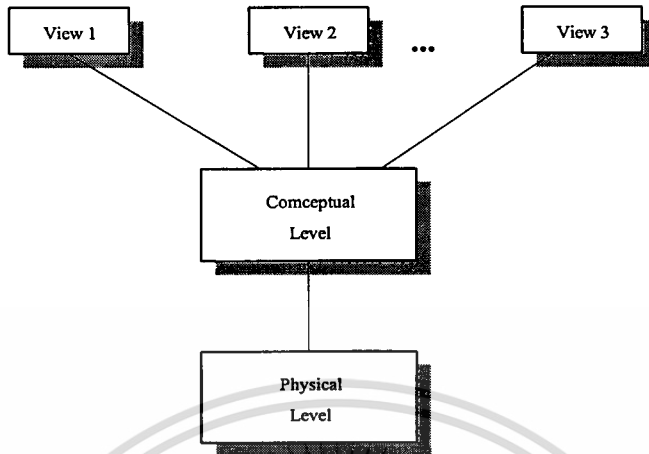
เกรโฮล (Gray Hole) คือ ขั้นตอนการทำงาน (Process) ที่ข้อมูลเข้า (Input) ทั้งหมดไม่เพียงพอที่จะทำให้เกิดผลลัพธ์ (Output) ของการทำงานออกมา

2.4 ฐานข้อมูล

ฐานข้อมูลคือ การรวบรวมเอาข้อมูลที่มีความเกี่ยวข้องโยงกันมาไว้รวมกัน อย่างมีระบบในที่เดียวกัน โดยผู้ใช้ฐานข้อมูลแต่ละคนจะสามารถใช้ข้อมูลได้ตามจุดประสงค์ของการประยุกต์ใช้งาน โดยที่ไม่ต้องสนใจว่าการจัดเก็บข้อมูลที่แท้จริงจะเป็นอย่างไร แต่จะสนใจเพียงว่าจะเรียกข้อมูลขึ้นมาใช้อย่างถูกต้องมีประสิทธิภาพให้มากที่สุด ซึ่งฐานข้อมูลได้จัดแบ่งระดับของข้อมูลออกเป็น 3 ระดับคือ (ดังแสดงในภาพที่ 2.1)

- 1) ระดับภายใน (Internal หรือ Physical Level) ซึ่งเป็นระดับที่ต่ำที่สุดซึ่งได้แก่ ระดับของการจัดเก็บข้อมูลจริง ๆ
- 2) ระดับหลักการ (Conceptual Level) ซึ่งในระดับการมอง Entity และความสัมพันธ์ระหว่าง Entity ที่มีอยู่ในระบบรวมทั้งกฎเกณฑ์ ต่าง ๆ เกี่ยวกับข้อมูลและผู้มีสิทธิ์จะใช้ ฯลฯ ข้อมูลในระดับนี้จะอยู่ในความสนใจของ DBA (Database Administrator) เพราะจะเป็นผู้ออกแบบและควบคุมการใช้งานข้อมูล
- 3) ระดับภายนอก (External หรือ View Level) เป็นระดับที่สูงที่สุดอันเป็นระดับข้อมูลที่จะมองเห็นจากการใช้งานของผู้ใช้แต่ละคน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.1 แสดงระดับของข้อมูล 3 ระดับ

ความหมายของฐานข้อมูลอีกนัยหนึ่งก็คือ ฐานข้อมูลเป็นการเอาไฟล์ข้อมูลที่ใช้ในองค์กรที่อยู่กระจัดกระจายตามที่ตั้งต่าง ๆ ที่อาจจะมีข้อมูลที่ซ้ำกัน ทำให้เปลืองเนื้อที่ในการจัดเก็บข้อมูล และข้อมูลที่จัดเก็บอาจจะเกิดความขัดแย้งกัน กล่าวคือ เมื่อมีการเปลี่ยนแปลงข้อมูลอาจจะเกิดการหลงลืมในการเปลี่ยนแปลงให้ครบทุกไฟล์ จากปัญหาของการประมวลผลแบบไฟล์ทำให้เกิดแนวความคิดของระบบฐานข้อมูลขึ้น ซึ่งการประมวลผลฐานข้อมูล จะสามารถช่วยขจัดปัญหาของการประมวลผลด้วยระบบไฟล์ เช่น ลดความซ้ำซ้อน และความไม่สอดคล้องของข้อมูล ช่วยลดความซ้ำซ้อนในการเข้าถึงข้อมูล สามารถใช้ข้อมูลร่วมกันได้ ข้อมูลมีความปลอดภัย ข้อมูลมีความคงสภาพและข้อมูลที่จัดเก็บมีความเป็นอิสระของข้อมูล

1) ลดความซ้ำซ้อนและความไม่สอดคล้องของข้อมูล ข้อมูลที่เป็นข้อมูลชนิดเดียวกันจะถูกเก็บไว้ที่เดียวกันทำให้มีการเปลี่ยนแปลง หรือปรับปรุงข้อมูลก็สามารถทำได้กับข้อมูลชุดเดียวกันทำให้ข้อมูลมีความถูกต้องสมบูรณ์และสอดคล้องอยู่ตลอดเวลา

2) ช่วยลดความซับซ้อนในการเข้าถึงข้อมูล ในระบบฐานข้อมูลจะมีภาษาที่ใช้ในการเข้าถึงข้อมูลที่ทำให้ผู้ใช้สามารถเข้าใจได้ง่ายและในระบบฐานข้อมูลจะมี ซอฟต์แวร์ที่เรียกว่าระบบการจัดการฐานข้อมูล (Database Management System) หรือเรียกสั้น ๆ ว่า DBMS เป็นตัวคอยจัดการให้

3) สามารถใช้ข้อมูลร่วมกันได้ ผู้ใช้สามารถใช้ข้อมูลที่มีอยู่ในฐานข้อมูลร่วมกันได้ ทำให้ไม่เปลืองเนื้อที่ในการจัดเก็บข้อมูลและสามารถแก้ไขข้อมูลได้ง่าย

4) ข้อมูลมีความปลอดภัย ในระบบฐานข้อมูลจะมีการกำหนด View หรือ Sub Schema ให้กับผู้ใช้แต่ละคนได้ ทำให้สามารถกำหนดสิทธิ์ในการเข้าถึงฐานข้อมูลของผู้ใช้แต่ละคนได้ โดย ผู้

บริหารฐานข้อมูล (Database Administrator หรือ DBA) ซึ่งจะเป็นบุคคลเดียวหรือกลุ่มคนก็ได้ และการเข้าถึงข้อมูลจะต้องผ่าน DBMS และข้อมูลยังมีการเข้ารหัสไว้อีกด้วยเพื่อเป็นการป้องกันการเข้าถึงข้อมูลโดยไม่ผ่าน DBMS

5) ข้อมูลมีความคงสภาพ ในระบบฐานข้อมูลจะมีการควบคุมความคงสภาพของข้อมูล (Integrity Constraint) คือเราสามารถที่จะกำหนดค่าของ Attribute แต่ละ Attribute ได้ ซึ่งจะให้มีลักษณะอย่างไร เช่น การที่บริษัทฯ จะนำบัตรภาษีมาชำระแทนเงินสดได้จะต้องได้รับบัตรภาษีจากกรมศุลกากรก่อน หรือ ในการตรวจสอบบัตรภาษีกับใบนำส่งจะต้องมีการบันทึกข้อมูลใบนำส่งก่อน แล้วจึงทำการตรวจสอบได้ เป็นต้น

6) ข้อมูลที่จัดเก็บมีความเป็นอิสระ การจัดเก็บจริงใน Disk จะอยู่ในระดับภายใน Internal หรือ Physical Level เมื่อมีการเปลี่ยนแปลงเทคนิควิธีการจัดเก็บ หรือ การเรียกใช้ข้อมูล ผู้ใช้ไม่จำเป็นต้องเขียนโปรแกรมขึ้นมาใหม่ ซึ่งตัว DBMS จะเป็นตัวที่จัดการเชื่อมข้อมูลระหว่างระดับภายนอกและระดับหลักการ กับเชื่อมระหว่างระดับหลักการกับระดับภายใน ซึ่งการเป็นอิสระของข้อมูลสามารถแบ่งออกเป็น 2 ชนิดคือ

6.1) ความเป็นอิสระแบบกายภาพ (Physical Data Independence) คือมีการเปลี่ยนแปลงวิธีการจัดเก็บข้อมูลระดับภายใน เช่นอาจจะเปลี่ยนเทคนิคการจัดเก็บข้อมูลจาก Index - Sequential เป็นแบบ Direct Access ก็จะไม่มีผลกระทบต่อโครงสร้างของระดับหลักการและระดับภายนอก

6.2) ความเป็นอิสระแบบตรรกะ (Logical Data Independence) คือมีการเปลี่ยนแปลงข้อมูลระดับ Conceptual เช่นอาจจะมีการเพิ่ม Attribute หรือ เพิ่ม Entity เข้าไปในระบบก็จะไม่มีผลกระทบต่อข้อมูลระดับภายนอกหรือระดับ View

2.4.1 Database Schema

Database Schema คือ ส่วนเค้าร่างของฐานข้อมูลที่ได้จากการออกแบบของฐานข้อมูลโดยรวม ซึ่งจะเป็นการกำหนดว่าฐานข้อมูลจะประกอบด้วย Entity อะไรบ้าง และแต่ละ Entity มีความสัมพันธ์กันอย่างไรและจะประกอบด้วย Attribute อะไรบ้าง และจะใช้ Attribute อะไรมาเป็นคีย์หลักของฐานข้อมูล โดยปกติแล้ว Database Schema ไม่ควรมีการเปลี่ยนแปลงหรือให้มีการเปลี่ยนแปลงน้อยที่สุด

Database Schema จะแบ่งออกเป็นระดับ คล้ายกับระดับของข้อมูลคือ

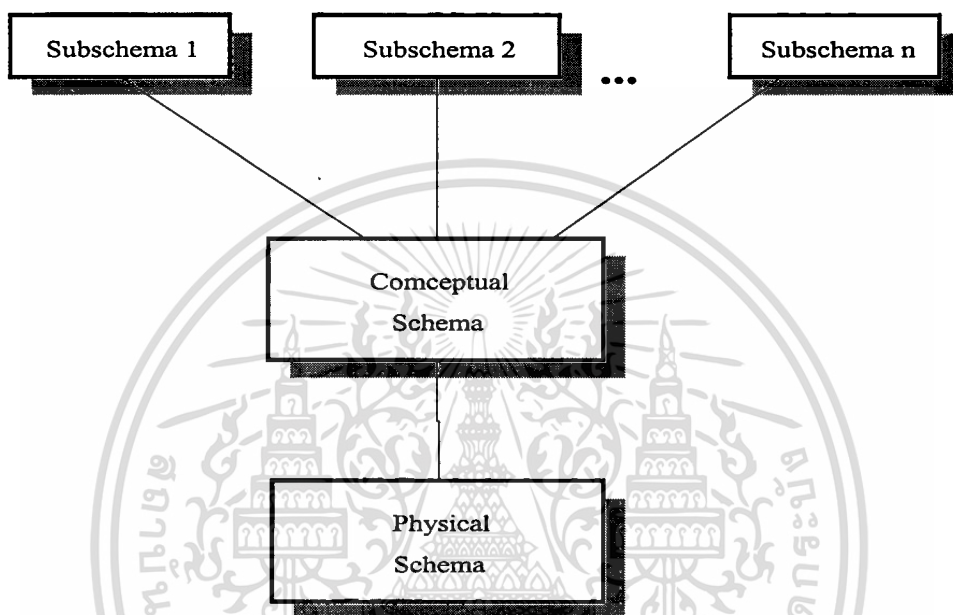
- 1) ระดับภายใน Internal หรือ Physical Schema คือการจัดเก็บข้อมูลจริง ๆ ใน Disk
- 2) ระดับหลักการ (Conceptual Schema) คือระดับของการออกแบบฐานข้อมูลว่าจะจัดเก็บ

ข้อมูลอะไรบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ระดับนอกสุด (Sub Schema) คือระดับในมุมมองของ User

ซึ่งจะเป็นได้ว่า ระบบฐานข้อมูลจะประกอบด้วย Physical Schema และ Conceptual Schema อย่างละ 1 ตัว และจะมี Sub Schema หลาย ๆ ตัว เช่นเดียวกับระดับของข้อมูลที่ตั้งแสดงในภาพที่ 2.2



ภาพที่ 2.2 Schema ในฐานข้อมูล 3 ระดับ

2.4.2 ระบบการจัดการฐานข้อมูล (Database Management Systems)

ระบบการจัดการฐานข้อมูล (Database Management Systems หรือเรียกย่อ ๆ ว่า DBMS) เป็นโปรแกรมที่ทำหน้าที่คอยอำนวยความสะดวกในการใช้งานระบบฐานข้อมูลให้กับผู้ใช้ (User) เพราะผู้ใช้ไม่จำเป็นจะต้องรู้เรื่องเกี่ยวกับรายละเอียดของการจัดเก็บข้อมูลในฐานข้อมูล ซึ่ง DBMS มีหน้าที่ดังต่อไปนี้

1) ทำหน้าที่ติดต่อกับตัวจัดการระบบไฟล์ DBMS จะทำหน้าที่ประสานงานกับ File Manager ในการจัดเก็บ เรียกใช้ และแก้ไขข้อมูล ซึ่ง DBMS จะทำหน้าที่แปลคำสั่ง DML ให้อยู่ในรูปแบบที่ File Manager เข้าใจ

2) การควบคุมความคงสภาพของข้อมูล ซึ่ง DBMS จะต้องทำหน้าที่ในการควบคุมให้ข้อมูลที่จัดเก็บใน Table ให้อยู่บนกรอบของข้อมูลที่กำหนดในส่วนของ Schema เช่น อายุของพนักงานไม่ต่ำกว่า 25

3) การควบคุมระบบความปลอดภัย DBMS จะทำหน้าที่ป้องกันไม่ให้ผู้มีสิทธิใช้ข้อมูล เข้าใช้ข้อมูลได้

4) การสร้างระบบสำรองและฟื้นฟูสภาพ เมื่อระบบหรือไฟล์ได้รับความเสียหาย DBMS จะต้องทำหน้าที่สำรองข้อมูลไว้ และทำการฟื้นฟูสภาพให้ระบบข้อมูลกลับเข้าสู่สภาพที่ถูกต้องสมบูรณ์ได้

5) การควบคุมภาวะพร้อมกัน (Concurrency Control) DBMS จะต้องจัดลำดับการเข้าถึงข้อมูลได้อย่างถูกต้องเมื่อมีการเรียกใช้ข้อมูลพร้อมกัน

2.4.3 โมเดลเชิงสัมพันธ์ (Relational Model) จะแสดงข้อมูลและความสัมพันธ์ระหว่างข้อมูลด้วยการรวบรวมของตาราง (Table) ซึ่งแต่ละตาราง จะประกอบด้วยจำนวนของ Columns ที่มีชื่อไม่ซ้ำกัน (ซึ่งตารางนี้อาจจะเรียกอีกอย่างว่า Relation) ข้อมูลของตาราง จะถูกจัดเก็บเรียงต่อไปเป็น Row ซึ่งจะทำให้ได้ ตาราง ซึ่งมี 2 มิติ คือ Row กับ Column ดังแสดงในตารางที่ 2.2

ตารางที่ 2.2 ตัวอย่างฐานข้อมูลแบบโมเดลเชิงสัมพันธ์

เลขที่รับใบนำส่ง	วันเดือนปีที่นำส่ง	จำนวนเงินที่นำส่ง	รหัสรายได้ที่นำส่ง
00001	11/12/41	40,000,000.00	0001
00002	15/12/41	85,400,000.00	0002
00003	01/01/42	70,258,880.50	0101
00004	10/12/42	1,000,000.30	0001

รหัสรายได้	ชื่อรายได้
0001	ภาษีเงินได้บุคคลธรรมดา
0002	ภาษีเงินได้นิติบุคคล
0101	ภาษีการค้า

ฐานข้อมูลเชิงสัมพันธ์จะแสดงในรูปแบบของตาราง (Table) ที่มี Column และ Row ซึ่ง Column แต่ละ Column จะเก็บข้อมูลชนิดเดียวกัน และ Row แต่ละ Row จะเก็บข้อมูลกลุ่มของ Column ซึ่งเราสามารถเรียก

Table	ว่า	Relation	หรือ	File	ในระบบประมวลผลแบบ File
Column	ว่า	Attribute	หรือ	Field	ในระบบประมวลผลแบบ File
Row	ว่า	Tuple	หรือ	Record	ในระบบประมวลผลแบบ File

ซึ่งเราจะนิยาม Relation คือ ตาราง 2 มิติ ที่

- แต่ละช่องของตารางจะบรรจุข้อมูลเพียงค่าเดียว
- ชื่อหัวข้อแต่ละ Column แตกต่างกันคือค่าของ Attribute
- ค่าข้อมูลที่อยู่ใน Column ได้แก่ ค่าของ Attribute ที่ระบุไว้ในหัวข้อ Column นั้น ๆ
- การเรียงลำดับของ Column ไม่ถือว่ามีค่าสำคัญ
- ข้อมูลแต่ละแถวจะต้องแตกต่างกัน
- การเรียงลำดับแถวไม่ถือว่ามีค่าสำคัญ

ดังนั้น ฐานข้อมูลเชิงสัมพันธ์ ได้แก่ การรวบรวมริเลชันต่าง ๆ ที่มีความสัมพันธ์ (Relationship) ระหว่างกัน

คีย์หลัก (Primary Key) จะประกอบไปด้วยค่าของ Attribute 1 ค่า หรือมากกว่าก็ได้ เพื่อที่สามารถใช้เป็นตัวแยกข้อมูลในแต่ละ Tuple มีความแตกต่างกัน

คีย์นอก (Foreign Key) คือค่าของ Attribute ของ Relation หนึ่งซึ่งมีค่าซ้ำกับคีย์หลัก (Primary Key) ของอีก Relation หนึ่ง

คีย์คู่แข่ง (Candidate Key) คือ Attribute ใน Relation ที่สามารถจะเป็นคีย์หลักของ Relation นั้น แต่เราไม่เลือกที่จะใช้เป็นคีย์หลัก

2.4.4 ทฤษฎีของระบบฐานข้อมูลเชิงสัมพันธ์

1) กฎของความคงสภาพ (Integrity Rule)

กฎของความคงสภาพของ โมเดลเชิงสัมพันธ์ (Relational Model) เป็นทฤษฎีที่ช่วยยืนยันความถูกต้องของความสัมพันธ์ระหว่างข้อมูลว่า ริเลชันใดที่เป็นไปตามกฎของความคงสภาพนี้ แล้วย่อมจะมีความสัมพันธ์ระหว่างข้อมูลอย่างถูกต้องอยู่ตลอดเวลา ไม่ว่าริเลชันนั้นจะมีการเปลี่ยนแปลงแก้ไขข้อมูลไปในรูปแบบใดก็ตาม กฎของความคงสภาพมีความหมายอยู่ 2 ลักษณะ คือ กฎความคงสภาพของเอนติตี้ (Entity Integrity Rule) และกฎความคงสภาพของการอ้างอิง (Referential Integrity Rule) ดังอธิบายได้ดังนี้

1.1) กฎความคงสภาพของเอนติตี้ กล่าวว่า “แอตทริบิวต์ทุกตัวที่เป็นส่วนของคีย์หลักจะไม่อนุญาตให้มีค่าว่าง” หมายความว่า คีย์หลักของทุกรีเลชันจะไม่สามารถเก็บค่าข้อมูลที่เป็นค่าว่างได้ เหตุผลของข้อกำหนดนี้ก็คือ เพื่อให้การเข้าถึงข้อมูลในแถวใดๆ ของรีเลชันมีความเป็นไปได้เสมอเพราะถ้าคีย์หลักของแถวใดมีค่าข้อมูลเป็นค่าว่างแล้ว ก็จะส่งผลให้การเข้าถึงข้อมูลในแถวนั้นไม่สามารถกระทำได้อย่างแน่นอน

1.2) กฎความคงสภาพของการอ้างอิง กล่าวว่า “ถ้าเรามีรีเลชัน R2 ดังตารางที่ 2.4 ซึ่งมี FK (Foreign Key) เป็นคีย์นอกที่อ้างอิงถึงคีย์หลัก PK (Primary Key) ในรีเลชัน R1 ดังตารางที่ 2.3 สำหรับทุกค่าของ FK ใน R2 จะต้อง

ก. มีค่าเท่ากับค่า PK ในแถวใดแถวหนึ่งในรีเลชัน R1

ข. มีค่าของแอตทริบิวต์ทุกตัวใน FK เป็นค่าว่าง”

หมายความว่า แอตทริบิวต์ใดๆ ที่เป็นคีย์หลักของรีเลชันหนึ่ง เมื่อมีการนำแอตทริบิวต์นั้นไปเป็นคีย์นอกของอีกรีเลชันหนึ่ง การเป็นคีย์นอกของแอตทริบิวต์นั้นจะต้องมีโดเมนเป็นโดเมนเดียวกันกับแอตทริบิวต์ที่เป็นคีย์หลัก ทั้งนี้เพื่อให้การนำรีเลชันมาใช้งานร่วมกัน (การนำรีเลชันมา Join กัน) กระทำได้อย่างถูกต้อง คือ ทุกแอตทริบิวต์ที่เป็นคีย์นอกจะต้องมีข้อมูลซ้ำกับข้อมูลของแอตทริบิวต์ที่เป็นคีย์หลักอย่างแน่นอน แต่อาจมีบางค่าข้อมูลของแอตทริบิวต์ที่เป็นคีย์หลักเป็นข้อมูลไม่อยู่ในโดเมนของแอตทริบิวต์ที่เป็นคีย์นอกก็ได้ นั่นคือ โดเมนของคีย์นอกจะต้องเล็กกว่าหรือเท่ากับ โดเมนของคีย์หลักเสมอ

ตารางที่ 2.3 รีเลชัน R1

คีย์หลักของ R1	คีย์อื่นๆของ R1
A	1
B	2
C	3

ตารางที่ 2.4 รีเลชัน R2

คีย์หลักของ R2	คีย์อื่นๆของ R1
A	A
B	B

2) ฟังก์ชันการขึ้นต่อกัน (Functional Dependency)

ฟังก์ชันการขึ้นต่อกันเป็นข้อกำหนดที่ช่วยให้เราเห็นถึงคงความสัมพันธ์ของแอตทริบิวต์ต่างๆ ที่อยู่ในรีเลชัน ทั้งนี้เพราะแอตทริบิวต์ต่างๆ ที่อยู่ในเอนทิตีเดียวกัน จะมีความสัมพันธ์กันเองโดยที่ความสัมพันธ์นี้อาจเกี่ยวข้องหรือไม่เกี่ยวกับความสัมพันธ์ที่มันมีต่อคีย์หลักของเอนทิตีนั้นก็เป็นได้ ซึ่งการที่แอตทริบิวต์เหล่านั้นมีความสัมพันธ์กันเองจะเป็นสิ่งที่เราต้องพิจารณาแยกออกเป็นรีเลชันย่อยๆ เพราะแอตทริบิวต์ของแต่ละรีเลชันก็ควรจะมีความสัมพันธ์กับคีย์หลักของรีเลชันของตนเองเท่านั้น กำหนดรีเลชัน R ดังแสดงตามตารางที่ 2.5 ถ้ามีแอตทริบิวต์ Y ของ R เป็นฟังก์ชันที่ขึ้นต่อแอตทริบิวต์ X ของ รีเลชัน เราสามารถเขียนแทนได้ด้วยสัญลักษณ์ $R.X \rightarrow R.Y$ อ่านว่า R.X มีฟังก์ชันการขึ้นต่อกับ R.Y หรือ R.X มีฟังก์ชันการขึ้นต่อกับ R.Y หรือ R.Y ขึ้นอยู่กับ R.X

นิยาม R.X มีฟังก์ชันการขึ้นต่อกับ R.Y ก็ต่อเมื่อ ทุกค่าข้อมูลของแอตทริบิวต์ X ใน R จะมีค่าข้อมูลของแอตทริบิวต์ Y ใน R ได้เพียงค่าเดียวเสมอ โดยที่แอตทริบิวต์ X และ Y อาจจะมีคีย์แบบรวม (Composite Key) ก็ได้

ตารางที่ 2.5 รีเลชัน R

X	Y
A	1
B	2
A	1
B	3

นิยาม R.X มีฟังก์ชันการขึ้นต่อกับ R.Y อย่างสมบูรณ์ (R.Y Fully Functionally Dependent on R.X) ก็ต่อเมื่อ R.Y มีฟังก์ชันการขึ้นต่อกับ R.X และไม่ขึ้นอยู่กับข้อมูลเพียงบางส่วนของ R.X โดยที่แอตทริบิวต์ X และ Y อาจจะเป็นคีย์แบบรวมก็ได้

ตารางที่ 2.5 รีเลชัน R (ต่อ)

X		Y
A	a	1
b	b	2
a	a	1
B	c	3

2.4.5 ภาษาของฐานข้อมูล

1) ภาษาที่ใช้สำหรับนิยามข้อมูล (Data Definition Language) หรือเรียกย่อ ๆ ว่า DDL เป็นภาษาที่ใช้กำหนด Database Schema ซึ่งเมื่อเราคอมไพล์คำสั่งของ DDL แล้วจะได้กลุ่มของตาราง Table ที่ใช้จัดการเก็บข้อมูล ซึ่งจะอยู่จัดเก็บไว้ในไฟล์พิเศษที่เรียกว่า Data Dictionary (or Directory) ซึ่งจะเก็บข้อมูลเกี่ยวกับโครงสร้างที่ได้จากการออกแบบฐานข้อมูลนั้น ๆ และเมื่อมีการเปลี่ยนแปลงหรือเรียกใช้ข้อมูล DBMS จะต้องอาศัยข้อมูลของโครงสร้างจากไฟล์นี้เสมอ

2) ภาษาที่ใช้สำหรับการจัดการข้อมูล (Data Manipulation Language) การจัดการข้อมูลในที่นี้หมายถึง

- การดึง เรียกดูข้อมูล (Information) ที่อยู่ในฐานข้อมูล
- การ Insert ข้อมูลใหม่ (New Information) เข้าไปในฐานข้อมูล
- การลบ (Delete) ข้อมูลออกจากฐานข้อมูล
- การปรับปรุง (Modify) ข้อมูลในฐานข้อมูล

ในส่วนของผู้ใช้จะเรียกใช้ข้อมูลโดยผ่าน DBMS โดยใช้ภาษาที่เรียกว่าภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language) หรือเรียกย่อ ๆ ว่า DML ซึ่งได้ถูกออกแบบให้ผู้ใช้ได้เข้าถึงข้อมูลหรือจัดการข้อมูลได้ง่าย ซึ่ง DML สามารถแบ่งออกได้ 2 ชนิดคือ

2.1) Procedural DML ซึ่งผู้ใช้จะต้องระบุลงไปที่ต้องการข้อมูลอะไร และจะใช้วิธีการใดซึ่งจะได้ข้อมูลมา ซึ่งจะมีลักษณะคล้าย ๆ กับภาษาชั้นสูง เช่น C, Pascal หรือภาษาเฉพาะของฐานข้อมูลนั้น ๆ เช่น FoxPro, Dbase เป็นต้น

2.2) Non -Procedural DML เป็นภาษาที่ผู้ใช้สามารถเรียนรู้และใช้งานได้ง่าย ถึงแม้ว่าผู้ใช้จะไม่มีความรู้เกี่ยวกับการเรียกใช้ข้อมูลเลย โดยที่ผู้ใช้เพียงแต่ระบุไปว่าต้องการข้อมูลอะไร โดยไม่จำเป็นต้องรู้วิธีการว่าจะทำอย่างไรเลย

2.4.6 โครงสร้างระบบฐานข้อมูล

ประกอบไปด้วยส่วนต่าง ๆ ดังภาพที่ 2.3 โครงสร้างของระบบฐานข้อมูล ซึ่งจะแสดงส่วนประกอบและการติดต่อของระบบฐานข้อมูล

1) File Manager มีหน้าที่ เป็นตัวจัดการ การจองเนื้อที่ใน Disk และโครงสร้างข้อมูลที่ใช้จัดเก็บข้อมูลใน Disk

2) Database Manager มีหน้าที่ติดต่อสื่อสารระหว่าง การจัดเก็บข้อมูลในชั้น Low – Level ฐานข้อมูลและ โปรแกรมประยุกต์ (Application Program) และการ Query

3) Query Processor ทำหน้าที่แปลงคำสั่งของภาษา Query ไปยังชั้น Low - Level เพื่อให้ Database Manager เข้าใจ

4) DML Precompiler ทำหน้าที่แปลงคำสั่งของ DML ที่อยู่ใน โปรแกรมประยุกต์ให้อยู่ในรูปการเรียกใช้ Procedure ปกติในภาษาหลัก (Host Language) ซึ่งตัว Precompiler ได้ตอบกับ Query เพื่อที่จะสร้าง Code ที่ถูกต้อง

5) DDL Compiler ทำหน้าที่ แปลงคำสั่งของ DDL ไปเป็นกลุ่มของตาราง (Table) ซึ่งประกอบด้วย Metadata หรือข้อมูลของข้อมูล

6) Data File ทำหน้าที่ เก็บข้อมูลของตัวเอง

7) Data Dictionary ทำหน้าที่เก็บ Metadata ที่เกี่ยวกับ โครงสร้างของฐานข้อมูลซึ่งจะต้องทำหน้าที่อย่างหนัก ดังนั้นจึงต้องออกแบบและสร้างอย่างดี

8) Indices ทำหน้าที่ช่วยในการเข้าถึงข้อมูลได้เร็วขึ้น

2.4.7 วิธีการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลจะอาศัยแบบจำลองอีอาร์ (Entity Relationship Model หรือ ER Model) โดยการออกแบบฐานข้อมูลด้วยแบบจำลองอีอาร์จะแบ่งออกเป็น 2 ขั้นตอน คือ

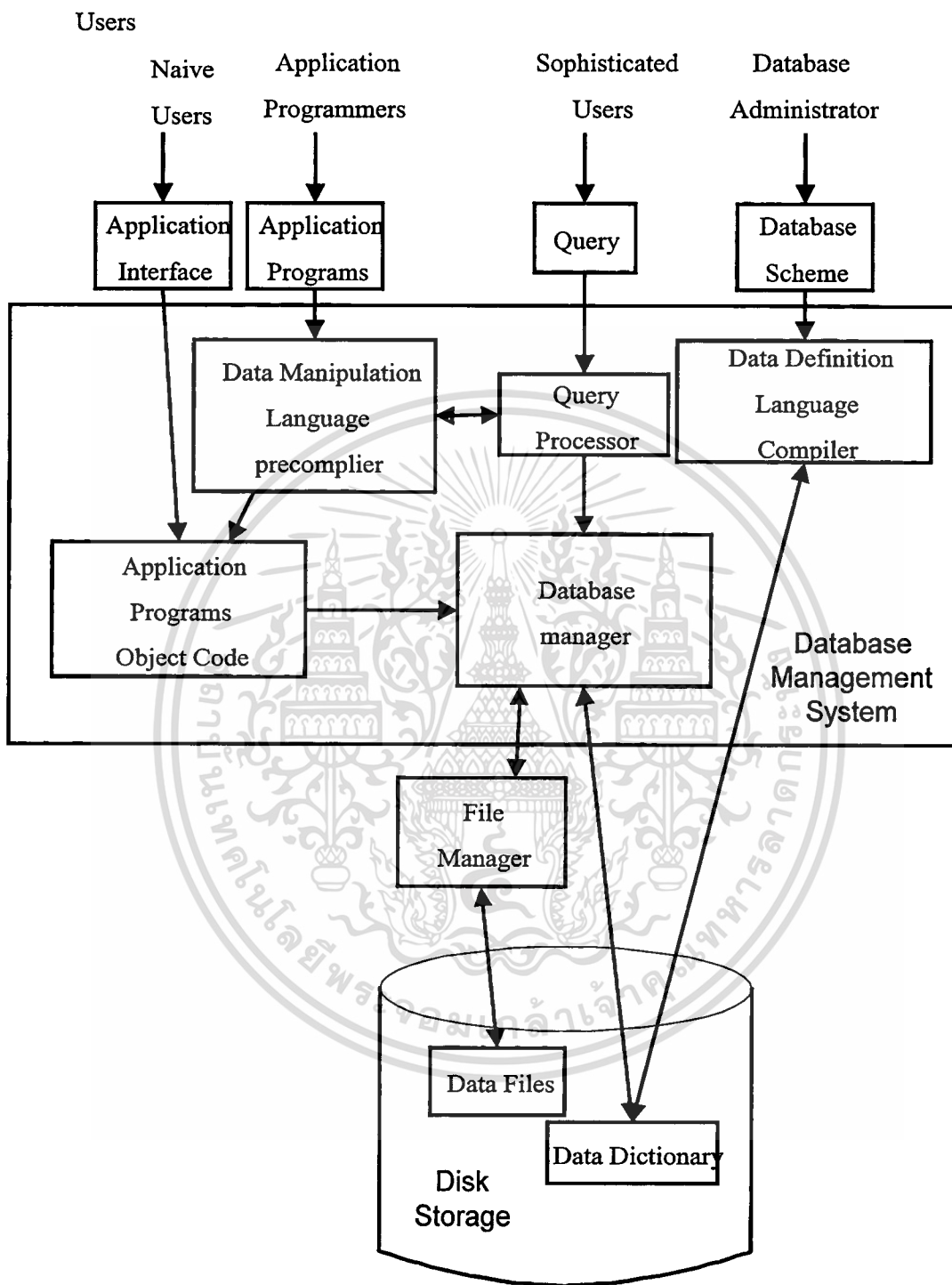
1) เป็นการสร้างแบบจำลองอีอาร์

แบบจำลองอีอาร์ เป็นแผนภาพที่ถูกออกแบบมาเพื่อแสดงความสัมพันธ์ระหว่างเอนทิตี (Entity) ต่าง ๆ ในระบบ ซึ่งจะอยู่ในรูปแบบที่เป็นรูปธรรมมากขึ้น ในแต่ละเอนทิตีก็จะประกอบไปด้วยแอตทริบิวต์ (Attribute)

เอนทิตี หมายถึง สิ่งต่าง ๆ ที่เราให้ความสนใจในระบบ เช่น พนักงาน (EMPLOYEE), หน่วยงาน (DEPARTMENT) หรือโครงการ (PROJECT) เป็นต้น

- Regular Entity Type สัญลักษณ์ที่ใช้ คือรูปสี่เหลี่ยม

- Weak Entity Type สัญลักษณ์ที่ใช้ คือรูปสี่เหลี่ยมซ้อนกัน 2 รูป



ภาพที่ 2.3 โครงสร้างของระบบฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดทริบิวต์ หมายถึง ลักษณะประกอบของแต่ละเอนติตี้ เช่น เอนติตี้พนักงาน (EMPLOYEE) จะประกอบด้วย แอดทริบิวต์ชื่อ, แอดทริบิวต์ที่อยู่, แอดทริบิวต์เงินเดือน หรือ แอดทริบิวต์วันเกิด เป็นต้น สัญลักษณ์ที่ใช้ คือรูปวงรี

ความสัมพันธ์ หมายถึง ความเกี่ยวพันระหว่างเอนติตี้ 2 เอนติตี้ขึ้นไป เช่น เอนติตี้หน่วยงาน (DEPARTMENT) กับเอนติตี้พนักงาน (EMPLOYEE) มีความเกี่ยวพันกันคือ พนักงานทุกคนต้องสังกัดในหน่วยงาน เป็นต้น สัญลักษณ์ที่ใช้ คือรูปสี่เหลี่ยมข้าวหลามตัด

- ความสัมพันธ์แบบหนึ่งต่อหนึ่ง
- ความสัมพันธ์แบบหนึ่งต่อกลุ่ม
- ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

ในความเป็นจริงแล้วสัญลักษณ์ต่าง ๆ ที่ใช้ในแบบจำลองอีอาร์ ยังมีรายละเอียดมากกว่าที่กล่าวมาข้างต้น แต่จะไม่ขอกกล่าวในรายงานฉบับนี้ หัวข้อนี้เพียงแต่กล่าวถึงความหมาย และสัญลักษณ์ที่ใช้ทั่ว ๆ ไปของแบบจำลองอีอาร์

2) เป็นการแปลงแบบจำลองอีอาร์ให้เป็นตารางฐานข้อมูล มีขั้นตอนดังนี้

2.1) สำหรับแต่ละเอนติตี้ที่เป็นเอนติตี้ปรกติ จะสร้างเป็นรีเลชัน โดยมีทุกแอดทริบิวต์มาประกอบกัน คีย์หลักของรีเลชันอาจเกิดจากการเลือกแอดทริบิวต์ใดแอดทริบิวต์หนึ่งหรือกลุ่มของแอดทริบิวต์ก็ได้

2.2) สำหรับแต่ละเอนติตี้ชนิดอ่อน จะสร้างรีเลชันที่เกิดจากการรวมกันของแอดทริบิวต์ โดยที่รีเลชันนี้จะมีคีย์หลัก หรือคีย์รวม ที่เกิดจากการรวมกันของคีย์หลักของเอนติตี้ชนิดอ่อน กับคีย์หลักของเอนติตี้ที่มันต้องอ้างอิงถึง

- สำหรับความสัมพันธ์ระหว่างเอนติตี้แบบหนึ่งต่อหนึ่ง จะสร้างรีเลชันจากความสัมพันธ์ดังกล่าวได้ คือ เลือกคีย์หลักของเอนติตี้ใดเอนติตี้หนึ่งมาเป็นคีย์หลักของรีเลชันนี้ แล้วให้คีย์หลักของอีกเอนติตี้หนึ่งมาเป็นคีย์นอกของรีเลชันนี้ โดยถ้าความสัมพันธ์นี้มีแอดทริบิวต์ก็ให้นำแอดทริบิวต์เหล่านั้นมารวมอยู่ในรีเลชันนี้ด้วย

- สำหรับความสัมพันธ์ระหว่างเอนติตี้แบบหนึ่งต่อกลุ่ม จะสร้างรีเลชันจากความสัมพันธ์ดังกล่าว โดยนำเอาคีย์หลักของเอนติตี้ฝั่งที่มีความสัมพันธ์แบบกลุ่มมาเป็นคีย์หลักของรีเลชันนี้ แล้วให้นำเอาคีย์หลักของเอนติตี้ฝั่งที่มีความสัมพันธ์แบบหนึ่ง มาเป็นคีย์นอกของรีเลชันนี้ โดยถ้าความสัมพันธ์นี้มีแอดทริบิวต์ ก็ให้นำแอดทริบิวต์เหล่านั้นมารวมอยู่ในรีเลชันนี้ด้วย

- สำหรับความสัมพันธ์ระหว่างเอนทิตีแบบกลุ่มต่อกลุ่ม จะสร้างรีเลชันจากความสัมพันธ์ดังกล่าว โดยนำเอาคีย์หลักของทั้งสองเอนทิตี มาประกอบกันเป็นคีย์หลักของรีเลชันนี้ โดยถ้าความสัมพันธ์นี้มีแอตทริบิวต์ ก็ให้นำแอตทริบิวต์เหล่านั้นมารวมอยู่ในรีเลชันนี้ด้วย

2.3) สำหรับเอนทิตีใดที่มีแอตทริบิวต์ที่มีค่าข้อมูลแบบหลายค่า (Multivalued Attribute) ก็ให้สร้างรีเลชันใหม่ โดยมีคีย์หลักของเอนทิตีนั้นรวมกับแอตทริบิวต์ดังกล่าวเป็นคีย์หลักของรีเลชันนี้

2.4) สำหรับความสัมพันธ์ระหว่างเอนทิตีที่เกิดจากเอนทิตีมากกว่า 2 เอนทิตี ให้สร้างรีเลชันของความสัมพันธ์นี้ โดยนำเอาคีย์หลักของทุกเอนทิตีมาประกอบกันเป็นคีย์หลักของรีเลชันนี้ โดยถ้าความสัมพันธ์นี้มีแอตทริบิวต์ ก็ให้นำแอตทริบิวต์เหล่านั้นมารวมอยู่ในรีเลชันนี้ด้วย

ตารางที่มีลักษณะเป็นรีเลชันจะต้องมีคุณสมบัติดังนี้

- แต่ละช่องของตารางจะบรรจุข้อมูลได้เพียงค่าเดียว
- ชื่อหัวข้อในแต่ละคอลัมน์มีความแตกต่างกัน อันเป็นชื่อของแอตทริบิวต์ของเอนทิตี
- ค่าข้อมูลในแต่ละคอลัมน์ คือ ค่าของแอตทริบิวต์ของเอนทิตี
- การเรียงลำดับคอลัมน์ไม่ถือว่ามีความสำคัญ
- ข้อมูลแต่ละแถวจะต้องแตกต่างกัน
- การเรียงลำดับแถวไม่ถือว่ามีความสำคัญ

ตารางที่มีคุณสมบัติดังกล่าวจะเรียกว่า รีเลชัน ดังนั้นเราจะได้นิยามของฐานข้อมูลเชิงสัมพันธ์ คือ ฐานข้อมูลที่เกิดจากการรวบรวมรีเลชันต่างๆ ที่มีความสัมพันธ์ (Relationship) ระหว่างกัน จะเรียกข้อมูลแต่ละแถวในแนวนอนของรีเลชันว่า ทัพเพิล (Tuple) และเรียกข้อมูลในแต่ละแถวในแนวตั้ง หรือแนวคอลัมน์ว่า แอตทริบิวต์ (Attribute) โดยที่คำว่า คีย์ (Key) จะหมายถึงข้อมูลที่เกิดจากแอตทริบิวต์ 1 ตัว หรือ หลายตัวก็ได้

แต่ละรีเลชันจะต้องมีสิ่งที่เรียกว่า คีย์หลัก (Primary key) คือ ข้อมูลของแอตทริบิวต์ 1 ตัว หรือมากกว่า 1 ตัวก็ได้ ที่สามารถใช้เป็นตัวเจาะจงบอกเราได้ว่ากำลังอ้างอิงถึงข้อมูลทัพเพิลใด

ส่วนคีย์ที่เป็นแอตทริบิวต์ของรีเลชันอื่นที่ซ้ำกับแอตทริบิวต์ที่เป็นคีย์หลักของรีเลชันหนึ่งจะเรียกว่า คีย์นอก (Foreign key) ในกรณีที่มีรีเลชันมีแอตทริบิวต์ หรือ กลุ่มของแอตทริบิวต์ที่มีได้ถูกเลือกให้เป็นคีย์หลัก เรียกว่า คีย์คู่แข่ง (Candidate key) หรือคีย์สำรอง (Alternate key) และแอตทริบิวต์อื่นๆ ที่เหลือที่มีได้เป็นคีย์หลักและไม่ได้เป็นส่วนใดส่วนหนึ่งของคีย์หลัก ก็จะถูกเรียกว่าเป็น Nonkey Attribute

คำว่าโดเมน (Domain) จะหมายถึง กรอบของค่าต่างๆ ที่เป็นไปได้ เช่น โดเมนของ แอตทริบิวต์ วันที่ ก็จะหมายถึงค่าของวันที่ที่เป็นไปได้ คือ มีค่าเท่ากับ 1 ถึง 31 แต่ในการเก็บค่า ข้อมูลลงในรีเลชันนั้น บางกรณีที่เรามีการกำหนดโดเมนให้แอตทริบิวต์แล้ว แต่ข้อมูลที่จะถูกเก็บ เข้าไปอาจถูกบรรจุเข้าไปในภายหลัง ลักษณะนี้จะทำให้เกิด ค่าว่าง (Null Value) ขึ้นชั่วขณะ ก่อนที่จะมีการบรรจุค่าข้อมูลที่อยู่ในโดเมนที่กำหนดไว้เข้าไป ดังนั้น คำว่า "ค่าว่าง" จึงหมายถึง ค่าที่ยังมีทราบชัดว่า แอตทริบิวต์นั้นจะมีค่าเป็นค่าใด หรือ ค่าของข้อมูลที่ไม่อยู่ในโดเมนที่ กำหนด โดยมีข้อบังคับไว้ว่าแอตทริบิวต์ที่ทำหน้าที่เป็น คีย์หลักของ รีเลชันจะมีค่าข้อมูลเป็นค่าว่าง ไม่ได้เสมอ เพราะจะทำให้การเข้าถึงข้อมูลใน ทัพลี (Tuple) นั้นกระทำไม่ได้

เมื่อมีการจัดเก็บข้อมูลในฐานข้อมูลใดๆ แล้ว ข้อมูลจะถูกแยกออกเป็นกลุ่มของข้อมูล เป็นชุดที่ประกอบด้วยแอตทริบิวต์ต่างๆ ที่มีความสัมพันธ์กัน เช่นการเก็บข้อมูลของบุคลากรในโรงเรียน ก็อาจแยกเก็บเป็นกลุ่มของข้อมูลนักเรียน, กลุ่มข้อมูลของครูอาจารย์ และกลุ่มของนักการภารโรง เป็นต้น กลุ่มข้อมูลแต่ละกลุ่มนี้จะเรียกว่า เอนติตี้ (Entity) ซึ่งแต่ละเอนติตี้จะประกอบไปด้วย แอตทริบิวต์ต่างๆ ที่มีความสัมพันธ์กัน เช่น เอนติตี้ของนักเรียนก็จะประกอบไปด้วย ชื่อ, ที่อยู่, ชั้นเรียน เป็นต้น จากการแยกจัดเก็บข้อมูลออกเป็นเอนติตี้นี้ แต่ละเอนติตี้ก็จะมี ความสัมพันธ์กัน ความสัมพันธ์ระหว่างเอนติตี้สามารถแบ่งออกเป็น 3 ชนิด คือ

ก. ความสัมพันธ์แบบหนึ่ง ต่อ หนึ่ง หมายถึง เมื่อเอนติตี้หนึ่งมีข้อมูลของคีย์หลักค่าหนึ่ง ค่าข้อมูลดังกล่าวก็จะมี ความสัมพันธ์กับค่าข้อมูลของคีย์หลักของอีกเอนติตี้หนึ่งเพียงค่าเดียวเท่านั้น เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่าง เอนติตี้นักเรียน กับ เอนติตี้ผู้ปกครอง เป็นแบบหนึ่ง ต่อ หนึ่ง แล้ว หมายความว่า การที่เราจะอ้างอิงถึงนักเรียนคนใดคนหนึ่งก็จะสามารถอ้างอิงถึงผู้ปกครองได้เพียงคนเดียวเท่านั้น และในทางตรงกันข้ามก็ต้องเป็นจริงด้วย คือ เมื่อเราอ้างอิงถึงผู้ปกครองคนใดคนหนึ่งแล้วก็สามารถอ้างอิงถึงนักเรียนได้เพียงคนเดียวเท่านั้น ดังตารางที่ 2.6

ตารางที่ 2.6 ตัวอย่างความสัมพันธ์ระหว่างนักเรียนกับผู้ปกครอง

ชื่อนักเรียน	ชื่อผู้ปกครอง
ก	A
ข	B
ค	C

ข. ความสัมพันธ์แบบหนึ่งต่อกลุ่ม หมายถึง เมื่อเอนติตีหนึ่งมีข้อมูลของคีย์หลักค่าหนึ่ง ค่าข้อมูลดังกล่าวก็จะมีความสัมพันธ์กับข้อมูลของคีย์หลักของอีกเอนติตีหนึ่งได้หลายค่า เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่าง เอนติตีบัตรภาษีกับเอนติตีใบนำส่งเป็นแบบหนึ่งต่อกลุ่มแล้ว จะหมายความว่า การที่เราอ้างอิงถึงเลขที่บัตรภาษีบัตรหนึ่งก็จะสามารถอ้างอิงถึงเลขที่คลังรับได้ เพียงเลขที่เดียวเท่านั้นและในทางตรงกันข้ามก็มีความหมายว่า เมื่อเราอ้างอิงถึงเลขที่คลังรับเลขที่หนึ่งแล้วก็สามารถอ้างอิงถึงเลขที่บัตรภาษีได้หลายค่า ดังตารางที่ 2.7

ตารางที่ 2.7 ตัวอย่างความสัมพันธ์ระหว่างเลขที่บัตรภาษีกับเลขที่คลังรับ

เลขที่บัตรภาษี	เลขที่คลังรับ (ใบนำส่ง)
02w000851	05001
02w000852	05001
01z000010	07002

ค. ความสัมพันธ์แบบกลุ่มต่อกลุ่ม หมายความว่า ค่าข้อมูลของคีย์หลักของเอนติตีหนึ่งที่แตกต่างกันอาจอ้างอิงถึงค่าข้อมูลของคีย์หลักของอีกเอนติตีหนึ่งได้ค่าเดียวหรือหลายค่าก็ได้ เช่น หากเรากำหนดให้ความสัมพันธ์ระหว่างเอนติตีนักเรียนกับเอนติตีผู้ปกครองเป็นแบบกลุ่มต่อกลุ่มแล้วก็จะหมายความว่า การที่เราอ้างอิงถึงนักเรียนคนหนึ่งหรือหลายคนก็จะสามารถอ้างอิงถึงผู้ปกครองคนเดียวกันได้ และในทางกลับกัน การที่เราอ้างอิงถึงผู้ปกครองคนหนึ่งหรือหลายคนก็จะสามารถอ้างอิงถึงนักเรียนคนเดียวกันก็ได้ ดังตารางที่ 2.8

ตารางที่ 2.8 ตัวอย่างความสัมพันธ์ระหว่างนักเรียนกับผู้ปกครอง

ชื่อนักเรียน	ชื่อผู้ปกครอง
ก	A
ข	A
ค	C
ค	D

นอกจากนี้ เอนทิตียังมีเอนทิตีบางประเภทที่การอ้างอิงถึงตัวมันได้อย่างสมบูรณ์นั้น จะต้องอ้างอิงถึงเอนทิตีอื่นเสมอ เช่น หากระบบฐานข้อมูลของบุคคลากรในโรงเรียนมีเอนทิตีของลูกนัการภารโรงด้วยแล้ว การอ้างอิงถึงเอนทิตีนี้ได้จะต้องอ้างอิงถึงเอนทิตีนักการภารโรงด้วยเสมอ เอนทิตีประเภทนี้เราเรียกว่า เอนทิตีชนิดอ่อน (Weak Entity)

2.5 การนอร์มัลไลซ์ (Normalization)

การนอร์มัลไลซ์เป็นการออกแบบฐานข้อมูลแบบที่เป็นมาตรฐานที่สุด ออกแบบโดย Codd โดยมีจุดประสงค์ของการออกแบบก็เพื่อลดความซ้ำซ้อนของความสัมพันธ์ของข้อมูลให้เหลือน้อยที่สุด (Minimum Redundancy) ซึ่งตามมาตรฐานปกติจะมีอยู่ 3 ระดับคือ

- 1NF (First Normal Form)
- 2NF (Second Normal Form)
- 3NF (Third Normal Form)

โดยที่รีเลชันใดที่ยังไม่สอดคล้องตามรูปแบบนอร์มัล (Normal Form) ทั้งสามก็จะต้องมีการแยกรีเลชันนั้น ๆ ออกเป็นรีเลชันย่อย ๆ ต่อไปอีก (Decomposition Method) ต่อมาได้มีการออกแบบเพิ่มเติมขึ้นอีก 2 ระดับ คือ

- 4NF (Forth Normal Form)
- 5NF (Fifth Normal Form)

หากรีเลชันใดมีมาตรฐานถึงรูปแบบนอร์มัลระดับที่ 5 (5NF) แล้วก็จะมั่นใจได้ว่า รีเลชันนั้นจะไม่มี ความซ้ำซ้อนของความสัมพันธ์ของข้อมูลอย่างแน่นอน

นอกจากนี้ยังมีการออกแบบรูปแบบนอร์มัลเพิ่มเติมระหว่างรูปแบบนอร์มัลที่ 3 (3NF) และรูปแบบนอร์มัลที่ 4 (4NF) โดย Boyce และ Codd ซึ่งมีชื่อว่า Boyce Codd Normal Form (BCNF) อีกด้วย

2.5.1 รูปแบบนอร์มัลระดับที่ 1

การปรับรีเลชันให้อยู่ในรูปแบบนอร์มัลระดับที่ 1 คือ การปรับจากรีเลชันที่ไม่นอร์มัล (Unnormalized Relation) เช่นรีเลชันที่มีข้อมูลของแอตทริบิวต์บางช่องมีมากกว่า 1 ค่า (มีแอตทริบิวต์ที่มีข้อมูลเป็น Repeating Group)

นิยามรีเลชันจะอยู่ในรูปแบบนอร์มัลระดับที่ 1(1NF) ก็ต่อเมื่อโดเมนของแต่ละแอตทริบิวต์ ประกอบด้วยข้อมูลที่เป็นหน่วยย่อยที่สุด สิ่งที่ได้จากการที่รีเลชันอยู่ในรูปแบบนอร์มัลระดับที่ 1 ก็คือ รีเลชันยังคงมีความซ้ำซ้อนของความสัมพันธ์ระหว่างข้อมูลอยู่มากมาย เพราะนิยามของรูปแบบนอร์มัลระดับที่ 1 นี้กำหนดเพียงเฉพาะว่า แต่ละแอตทริบิวต์ของรีเลชันจะมีโดเมนที่มี

สมาชิกเป็นหน่วยที่เล็กที่สุดเท่านั้น มิได้เป็นการลดความซ้ำซ้อนของความสัมพันธ์ระหว่างข้อมูล แต่ประการใด

2.5.2 รูปแบบนอร์มัลระดับที่ 2

นิยาม รีเลชันจะอยู่ในรูปแบบนอร์มัลระดับที่ 2 (2NF) ก็ต่อเมื่อรีเลชันนั้นอยู่ในรูปแบบนอร์มัลระดับที่ 1 แล้ว และทุกแอตทริบิวต์ที่ไม่เป็นส่วนใดส่วนหนึ่งของคีย์หลัก (Non-Key attribute) จะต้องขึ้นอยู่กับคีย์หลักของรีเลชันนั้นอย่างสมบูรณ์ สิ่งที่ได้จากการที่รีเลชันอยู่ในรูปแบบนอร์มัลระดับที่ 2 คือ ข้อมูลบางแอตทริบิวต์ที่ไม่ใช่คีย์หลักอาจมีความสัมพันธ์กันเองโดยที่ไม่มีความสัมพันธ์กับคีย์หลักเลย ซึ่งความสัมพันธ์ดังกล่าวนี้ถือว่าเป็นความซ้ำซ้อนประการหนึ่งของรีเลชันนั้น ๆ ที่จะต้องทำการลดด้วยรูปแบบนอร์มัลในระดับต่อไป

2.5.3 รูปแบบนอร์มัลระดับที่ 3

นิยาม รีเลชันจะอยู่ในรูปแบบนอร์มัลระดับที่ 3 (3NF) ก็ต่อเมื่อรีเลชันนั้นอยู่ในรูปแบบนอร์มัลระดับที่ 2 แล้ว และทุกแอตทริบิวต์ที่ไม่เป็นส่วนใดส่วนหนึ่งของคีย์หลักจะต้องไม่เป็นฟังก์ชันที่ขึ้นต่อกันเอง โดยปกติแล้ว สิ่งที่ได้จากการที่รีเลชันอยู่ในรูปแบบนอร์มัลระดับที่ 3 คือ รีเลชันจะไม่มีซ้ำซ้อนอีกต่อไปโดยที่จะสอดคล้องกับรูปแบบนอร์มัลระดับที่ 4 และ 5 ด้วย แต่ก็มีรีเลชันบางลักษณะที่จะต้องทำให้อยู่ในรูปแบบนอร์มัลระดับที่ 4 และ 5 ต่อไป รีเลชันดังกล่าวจะมีลักษณะดังต่อไปนี้

- 1) เป็นรีเลชันที่มีหลายคีย์คู่แข่ง และ
- 2) เป็นคีย์คู่แข่งที่เกิดจากการรวมตัวกันของคีย์ย่อย ๆ (Candidate Key เป็น Combine Key) และ
- 3) เป็นคีย์คู่แข่งที่มีการเหลื่อมซ้อนกัน (Overlap กัน)

2.5.4 รูปแบบบอยส์และคอตต์

นิยาม รีเลชันใด ๆ จะจัดอยู่ในรูปแบบบอยส์และคอตต์ (BCNF) ถ้าทุก ๆ ตัวเลือก (Determinant) เป็นคีย์คู่แข่ง (Candidate Key) หรืออาจกล่าวได้ว่าทุกแอตทริบิวต์ที่เป็นคีย์หลักของรีเลชันจะต้องไม่ขึ้นกับเซตย่อยของคีย์หลักตัวอื่น

2.5.5 รูปแบบนอร์มัลระดับที่ 4

นิยาม รีเลชันใด ๆ จะจัดอยู่ในรูปแบบนอร์มัลระดับที่ 4 (4NF) ถ้ารีเลชันนั้นเป็น BCNF และเป็นรีเลชันไม่มีการขึ้นต่อกันเชิงกลุ่ม (ต้องไม่มี Multi-Valued Dependency (MVD) เกิน 1 ตัว)

2.5.6 รูปแบบนอร์มัลระดับที่ 5

นิยาม รีเลชันใด ๆ จะจัดอยู่ในรูปแบบนอร์มัลระดับที่ 5 (5NF) ถ้ารีเลชันนั้นเป็น 4NF และผลลัพธ์ของการ Join ของทุก ๆ Join Dependency (JD) ต้องเหมือนเดิม ไม่เกินหรือสูญหาย (Lossless Join) ดังตารางที่ 2.9 แสดงการนอร์มัลไลซ์ข้อมูลของระบบที่กำลังพัฒนาในโครงการนี้

ตารางที่ 2.9 แสดงการนอร์มัลไลซ์ข้อมูลบัตรภาษี (TAX CARD)

รายละเอียดบัตรภาษี	UNF	1NF	2NF	3NF	Table Name	
เลขที่บัตรภาษี	TAX-NUM	TAX-NUM	<u>TAX-NUM</u>	<u>TAX-NUM</u>	TAXCARD	
รหัสชนิดบัตร	TAX-KIND	TAX-KIND	TAX-KIND	TAX-KIND		
วันเดือนปี ที่ออกบัตร	TAX-DTE-S	TAX-DTE-S	TAX-DTE-S	TAX-DTE-S		
วันเดือนปี บัตรหมดอายุ	TAX-DTE-E	TAX-DTE-E	TAX-DTE-E	TAX-DTE-E		
มูลค่าของบัตร						
เลขที่ผู้เสียภาษี	TAX-VAL	TAX-VAL	TAX-VAL	TAX-VAL		
ชื่อบริษัท	COMP-NUM	COMP-NUM	COMP-NUM	COMP-NUM		
ผู้ประกอบการ	COMP- NAME	COMP- NAME	COMP- NAME	COMP- NAME		
ชื่อบริษัท ผู้ประกอบการ			<u>COMP-NUM</u> COMP- NAME	<u>COMP-NUM</u> COMP- NAME		COMPANY
รหัสชนิดบัตร ชนิดบัตร			<u>TAX-KIND</u> TAX-NAME	<u>TAX-KIND</u> TAX-NAME		TAXKIND

บทที่ 3

การวิเคราะห์ระบบ

3.1 ศึกษาความเป็นไปได้

การศึกษาความเป็นไปได้ เป็นการศึกษาถึงความเป็นไปได้ในการที่จะพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัตรภาษีของกรมบัญชีกลางขึ้นใหม่ ที่สามารถตอบสนองความต้องการของผู้ใช้และลดขั้นตอนการปฏิบัติงานรวมถึงสามารถสืบค้นข้อมูลที่ต้องการได้รวดเร็วยิ่งขึ้น เพื่อทดแทนระบบงานเดิมที่มีการประมวลผลแบบ Batch Processing บนเครื่อง Mainframe ที่ใช้ภาษาโคบอล โดยมีกระบวนการศึกษาดังนี้

3.1.1 ข้อมูลเกี่ยวกับบัตรภาษี

บัตรภาษีเป็นบัตรที่กรมศุลกากร ออกให้แก่ผู้ประกอบการส่งสินค้าออก เพื่อเป็นการจ่ายชดเชยภาษีอากร ซึ่งมีอยู่ในต้นทุนการผลิตสินค้าส่งออก ตามพระราชบัญญัติชดเชยภาษีอากร สินค้าส่งออกที่ผลิตในราชอาณาจักร พ.ศ. 2524 เป็นการสนับสนุนให้มีการส่งสินค้าที่ผลิตในประเทศไทยออกไปจำหน่ายยังต่างประเทศให้มากขึ้น และมีขั้นตอนดังนี้

1) กรมศุลกากร เป็นผู้ออกบัตรภาษีให้แก่ผู้ประกอบการส่งสินค้าออก ซึ่งบัตรภาษีแต่ละใบจะมีมูลค่าของบัตรกำกับไว้เป็นใบละ หนึ่งแสนบาท, หนึ่งหมื่นบาท, หนึ่งพันบาท และไม่ว่านุราคา (สำหรับบัตรที่มีมูลค่าน้อยกว่าหนึ่งพันบาท) ดังตัวอย่างภาพที่ 3.1, 3.2 และ 3.3 โดยปกติบัตรภาษีจะมีอายุ 3 ปี

2) เมื่อผู้ประกอบการได้รับบัตรภาษีแล้ว สามารถนำมาชำระที่กรมศุลกากร กรมสรรพากรและกรมสรรพสามิต

3) เมื่อหน่วยงานตามข้อ 2 ได้รับการชำระด้วยบัตรภาษีแล้ว จะเก็บรวบรวมบัตรภาษีนำส่งที่ส่วนประมวลบัญชี กรมบัญชีกลางพร้อมกับใบนำส่งดังภาพที่ 3.4 เมื่อส่วนประมวลบัญชีรับบัตรภาษีแล้ว ลงเลขที่คลังรับ และวันที่รับ แล้วรวบรวมบัตรภาษีและตรวจสอบบัตรภาษีคร่าว ๆ ในส่วนของจำนวนบัตรและรายละเอียดบางส่วนครบหรือไม่ เช่น ลายเซ็นต์ เป็นต้น

4) ตรวจสอบข้อมูลในรายละเอียดของบัตรภาษี และตรวจสอบบัตรภาษีกับใบนำส่ง และการจัดทำรายงานด้วยระบบคอมพิวเตอร์เพราะว่าข้อมูลบัตรภาษีมีเป็นจำนวนมาก โดยในแต่ละเดือนมีข้อมูลประมาณ 1 แสนใบ

4) นำข้อมูลบัตรภาษีที่ได้ตรวจสอบความถูกต้องและจัดทำรายงานเสร็จแล้วนำเสนอกรมศุลกากรต่อไป ซึ่งขอบเขตของการพัฒนาระบบจะครอบคลุมข้อ 3) และ 4) โดยจะทำการตรวจสอบบัตรภาษีให้ถูกต้องตรงกัน และตรวจสอบจำนวนบัตรภาษีและจำนวนเงินให้ตรงกับรายได้ที่นำส่งคลังในใบนำส่งที่มาพร้อมกัน

3.1.2 ศึกษาความต้องการ

ในการพัฒนาระบบมีกระบวนการเข้าไปสอบถามผู้ที่เกี่ยวข้องกับระบบทำให้ทราบความต้องการของระบบดังนี้

- 1) ระบบใหม่ต้องเสร็จก่อนปี ค.ศ. 2000 คือ พร้อมจะใช้งานได้ในปี ค.ศ. 2000
- 2) ระบบใหม่ต้องมีความรวดเร็วในการตรวจสอบบัตรภาษีและง่ายต่อการปฏิบัติงาน
- 3) ระบบใหม่ต้องพัฒนาบน Hardware และ Software ที่มีอยู่แล้วในหน่วยงาน

3.1.3 ปัญหาของระบบเดิม

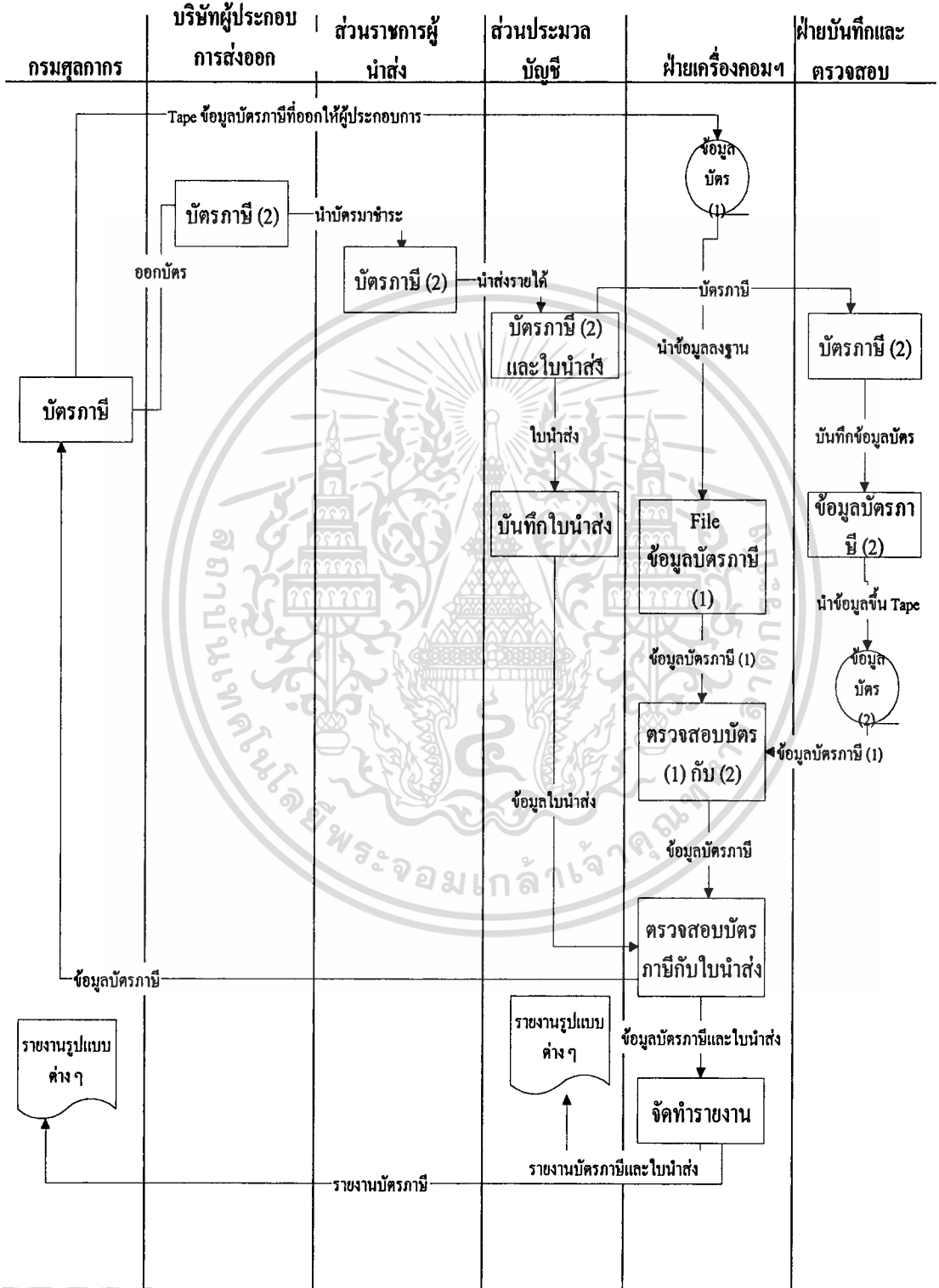
1) ระบบเครื่องคอมพิวเตอร์ที่ใช้ประมวลผลระบบงานปัจจุบัน เป็นเครื่องระดับเมนเฟรม (Mainframe) ยี่ห้อ NEC รุ่น 3400 ซึ่งเป็นระบบปิด (Closed System) เป็นเทคโนโลยีเฉพาะและเป็นรุ่นที่ไม่อยู่ในสายการผลิตของบริษัทผู้ผลิต ทำให้มีความยุ่งยากในการแก้ปัญหา Y2K จำเป็นต้องย้ายระบบงานออกจากระบบเครื่องเดิม โดยระบบเครื่องเดิมในปัจจุบันมีระบบอื่น ๆ ภายในหน่วยงานยังใช้ทำการประมวลผล พอถึงปี ค.ศ. 2000 ถ้าจำเป็นต้องใช้ระบบเครื่องเดิมมีวิธีแก้ไขโดยการตั้งวันที่ของระบบให้ย้อนหลัง แต่ไม่สามารถใช้วันที่ของระบบได้ ซึ่งในลักษณะนี้บริษัทที่เกี่ยวข้องก็ไม่รับประกันว่าจะถูกต้องสมบูรณ์ และท้ายที่สุดทางหน่วยงานมีแผนว่าจะเลิกใช้ระบบเครื่องนี้ไป เพราะขณะนี้ระบบงานอื่น ๆ ที่ประมวลผลบนระบบเครื่องนี้กำลังทำการโอนย้ายระบบและมีการจัดซื้อระบบเครื่องใหม่มาทดแทน

2) ได้รับความผิดพลาดและรายงานช้า เนื่องจากข้อมูลบัตรภาษีมีเป็นจำนวนมากต้องใช้เวลาในการตรวจสอบข้อมูลนาน เช่น กรณีที่กรมศุลกากรต้องการสอบถามข้อมูลบัตรภาษีที่มีการแจ้งบัตรหายว่าหายจริงหรือไม่ ก็ใช้เวลานานกว่าจะทราบคำตอบทำให้ไม่ทันกับความต้องการ เนื่องจากระบบปัจจุบันใช้ภาษาโคบอล และการประมวลผลเป็นแบบ Batch Processing จึงมีความจำเป็นต้องเปลี่ยนระบบเป็นลักษณะ On-Line

3.2 ระบบงานปัจจุบัน

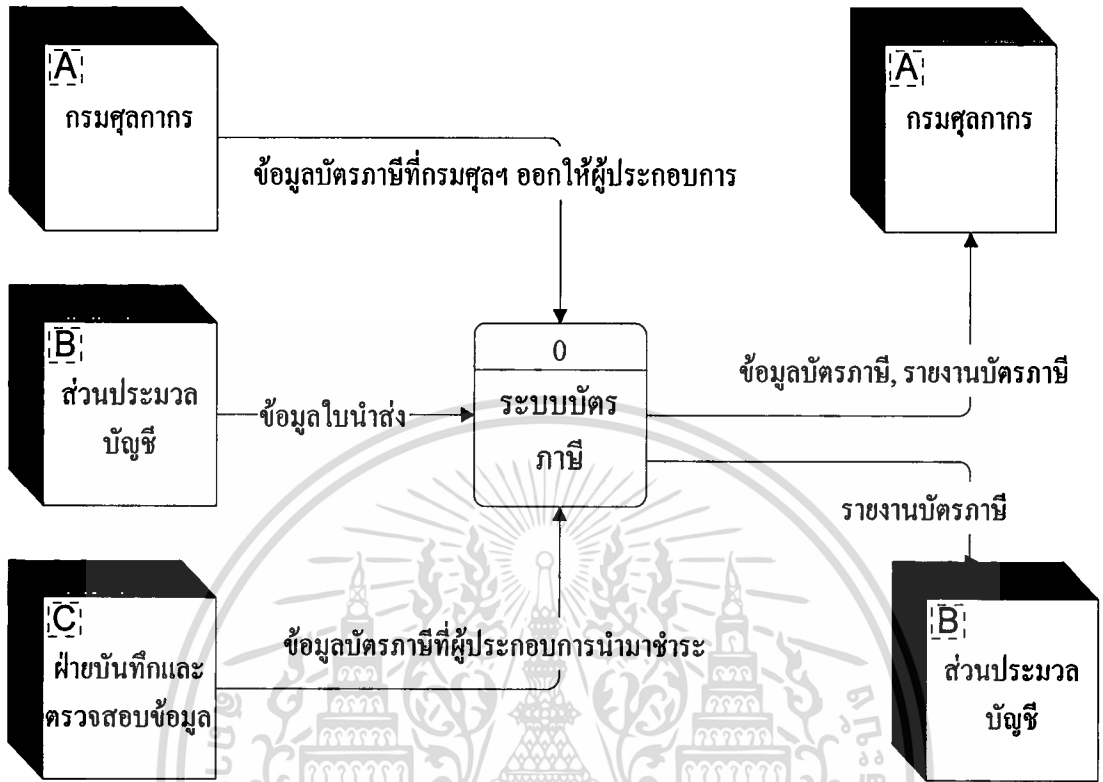
ระบบงานเดิมเรียกว่า “ระบบบัตรภาษี” โดยครอบคลุมถึงทั้งส่วนกลางและส่วนภูมิภาค ซึ่งภาพรวมของระบบ ดังภาพที่ 3.5 Work Flow ของระบบ, 3.6 Existing System Context Diagram, ขั้นตอนการทำงานของระบบปัจจุบัน ตามภาพที่ 3.7 และ ภาพที่ 3.8 DFD Level 0 Existing System

Work Flow ของระบบงาน



ภาพที่ 3.5 Work Flow ของระบบงานปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.6 Existing System Context Diagram

3.2.1 ข้อมูลนำเข้าระบบ

จากภาพที่ 3.6 จะเห็นว่าข้อมูลนำเข้าได้มาจาก 3 แหล่งคือ

1) กรมศุลกากร ข้อมูลที่นำเข้าระบบ เป็นข้อมูลบัตรภาษีที่กรมศุลกากร ออกให้แก่ผู้ประกอบการส่งสินค้าออก โดยจะมีการนำเฉพาะข้อมูลส่วนที่เพิ่มและส่วนที่มีการเปลี่ยนแปลงเข้าสู่ระบบสัปดาห์ละ 2 ครั้ง ซึ่งเป็น "Master File" ของระบบ เหตุผลของการนำข้อมูลจากกรมศุลกากรมาใช้ในระบบ เพื่อลดความซ้ำซ้อนในการบันทึกและตรวจสอบข้อมูลบัตรภาษีในส่วนที่กรมศุลกากรออกให้แก่ผู้ประกอบการ ซึ่งได้มีการจัดเก็บข้อมูลไว้ในระบบเครื่องคอมพิวเตอร์อยู่แล้ว

2) ฝ่ายบันทึกและตรวจสอบข้อมูล กรมบัญชีกลาง ข้อมูลที่นำเข้าระบบ จะเป็นข้อมูลบัตรภาษี ที่ผู้ประกอบการส่งสินค้าออกนำมาชำระแทนเงินสดที่กรมศุลกากรหรือกรมสรรพสามิตหรือกรมสรรพากร โดยมีส่วนประมวลบัญชี กรมบัญชีกลาง เป็นผู้รวบรวมก่อนที่จะนำบัตรภาษี ส่งให้ฝ่ายบันทึกและตรวจสอบข้อมูล ทำการบันทึกข้อมูล

3) ส่วนประมวลบัญชี กรมบัญชีกลาง ข้อมูลที่นำเข้าระบบเป็นข้อมูลโอนนำส่ง ซึ่งโอนนำส่งจะมาพร้อมกับบัตรภาษี ที่ส่วนราชการได้รวบรวม แล้วนำส่งที่ส่วนประมวลบัญชี

3.2.2 ข้อมูลนำออกจากระบบ

จากภาพที่ 3.6 มีข้อมูลออกจากระบบดังนี้

1) รายงานต่างๆ ตามความต้องการของผู้ใช้ได้แก่ ส่วนประมวลบัญชี กรมบัญชีกลาง และกรมศุลกากร

2) ข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาชำระแทนเงินสด ที่ได้ตรวจสอบความถูกต้องแล้ว

3.2.3 ขั้นตอนการทำงานขอระบบปัจจุบัน

จากภาพที่ 3.7 แสดงถึงขั้นตอนการทำงานขอระบบปัจจุบันดังนี้

- 1) โหลดข้อมูลบัตรภาษีที่นำมาจากกรมศุลกากรในรูปแบบแม่เหล็กจัดเก็บลงไฟล์
- 2) บันทึกข้อมูลโอนนำส่งรายได้แผ่นดินแล้วจัดเก็บขึ้นเทปเพื่อนำไปตรวจสอบกับข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาชำระแทนเงินสดที่ส่งมาพร้อมกัน
- 3) บันทึกข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาชำระแทนเงินสดแล้วจัดเก็บขึ้นเทปแม่เหล็กเพื่อนำไปเปรียบเทียบกับข้อมูลบัตรภาษีของกรมศุลกากร
- 4) หลังจากเปรียบเทียบข้อมูลบัตรภาษีที่ได้จากทั้ง 2 แหล่งแล้ว
 - 4.1) ถ้ามีข้อมูลไม่ตรงกัน จัดทำรายงานข้อผิดพลาดเพื่อนำมาตรวจสอบให้ถูกต้องก่อนทำขั้นตอนต่อไป
 - 4.2) ถ้าไม่มีข้อมูลที่เกิดจากการบันทึกผิดพลาด ไปทำขั้นตอนต่อไป
- 5) ทำการประมวลผลเพื่อคำนวณจำนวนเงินและจำนวนบัตรภาษี
- 6) เปรียบเทียบข้อมูลบัตรภาษีที่ถูกต้องที่ได้จากขั้นตอนที่ 5 กับข้อมูลโอนนำส่งรายได้

แผ่นดิน

6.1) ถ้ามีข้อมูลผิดพลาด จัดทำรายงานข้อมูลที่ไม่ถูกต้องเพื่อนำไปตรวจสอบให้ถูกต้อง

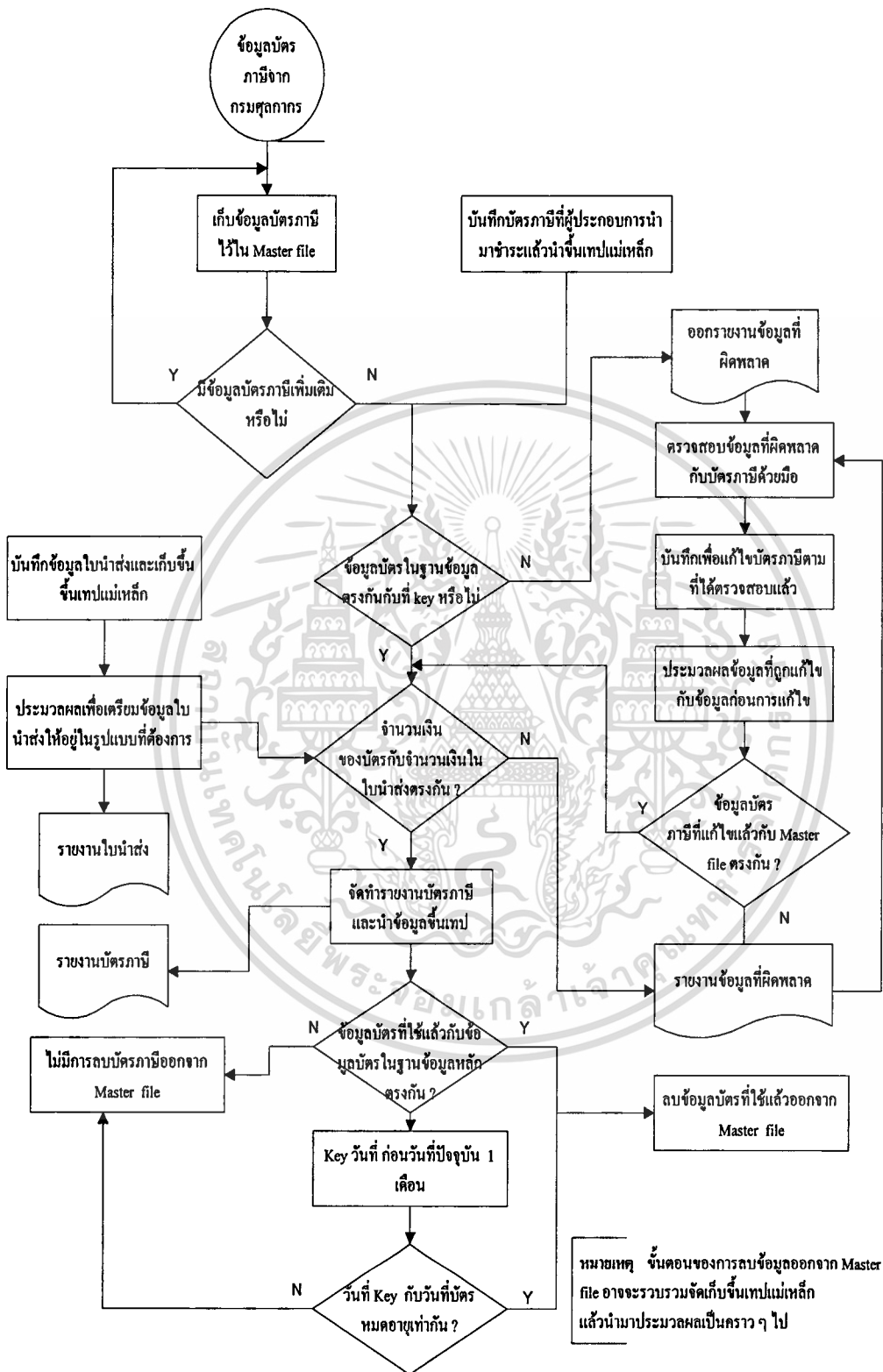
6.2) ถ้าข้อมูลถูกต้องไปทำขั้นตอนต่อไป

7) จัดทำรายงานบัตรภาษี

8) นำข้อมูลที่ได้ตรวจสอบความถูกต้องแล้วขึ้นเทปแม่เหล็ก เพื่อนำส่งกรมศุลกากร

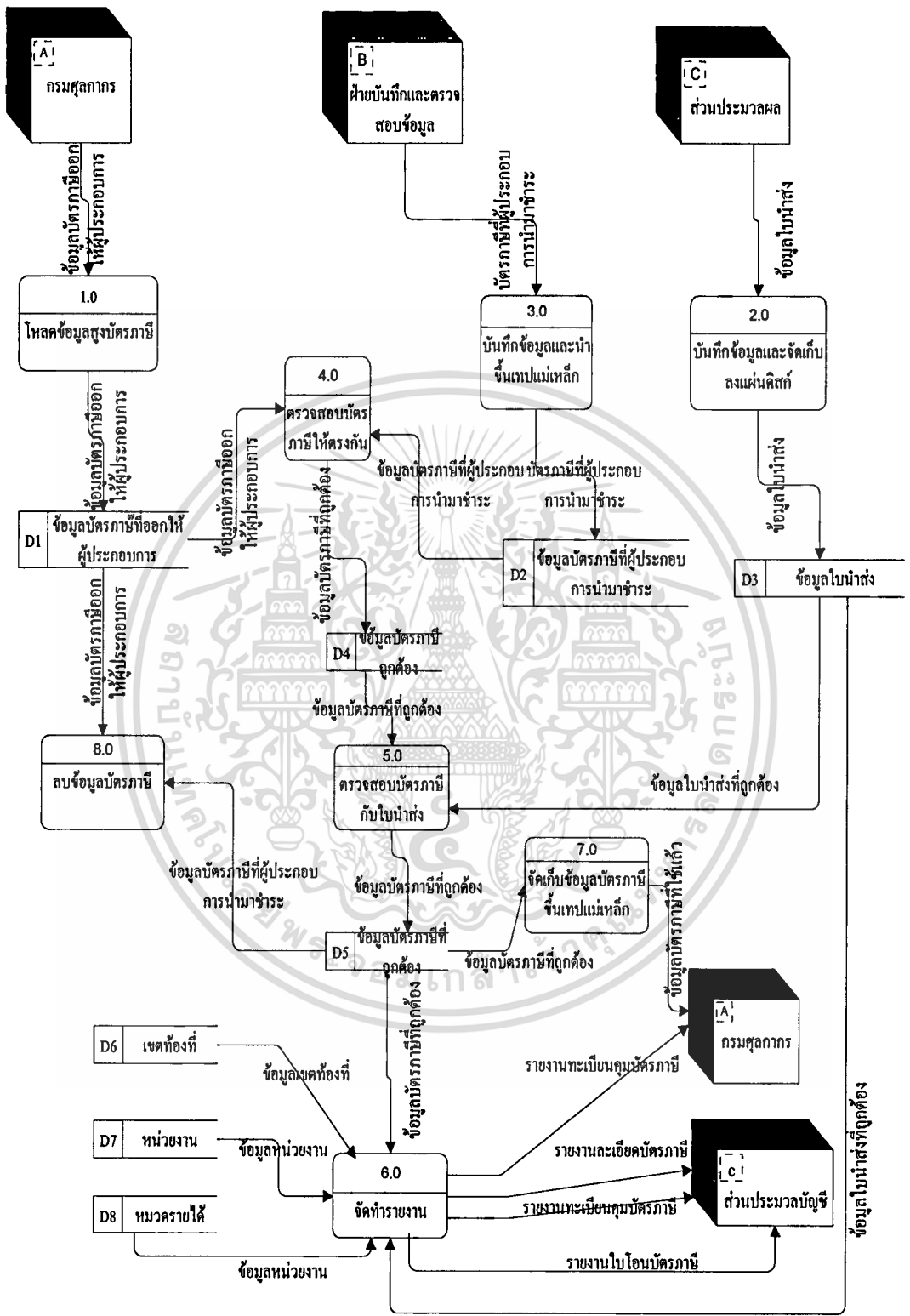
9) ลบข้อมูลที่ใช้แล้วออกจากไฟล์หลักด้วยข้อมูลที่ได้จากข้อ 6

10) ลบข้อมูลบัตรภาษีที่หมดอายุออกจากไฟล์หลัก



ภาพที่ 3.7 ขั้นตอนการทำงานของระบบปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.8 DFD Level 0 Existing System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การพัฒนาทางเลือกและประเมินทางเลือก

จากการศึกษาระบบงานปัจจุบัน สามารถพัฒนาทางเลือก สำหรับแก้ปัญหาได้ 2 ทางเลือก คือ

3.3.1 ทางเลือกที่ 1

1) ทำการเชื่อมต่อฐานข้อมูลบัตรภาษี ที่ศูนย์สารสนเทศ กรมบัญชีกลาง กับฐานข้อมูลบัตรภาษี กรมศุลกากรเข้าด้วยกัน โดยผ่านเครือข่ายขององค์การโทรศัพท์ แทนระบบปัจจุบันที่ใช้เทปแม่เหล็ก เป็นสื่อกลางในการนำข้อมูลเข้าสู่ระบบ

2) ลดขั้นตอนการปฏิบัติงาน ในส่วนของการตรวจสอบการบันทึกบัตรภาษีที่ผู้ประกอบการนำมาชำระแทนเงินสด โดยบันทึกกับเครื่องที่เชื่อมต่อฐานข้อมูลบัตรภาษีโดยตรง ซึ่งจะบันทึกเฉพาะเลขที่บัตรภาษี ซึ่งเป็นรหัสที่ Unique คือรหัสที่ไม่ซ้ำกัน พร้อมกับเปรียบเทียบข้อมูลที่บันทึกขณะนั้นกับข้อมูลในฐานข้อมูลหลัก ก็จะสมารถนำข้อมูลที่มี รหัสเลขที่บัตรภาษี จากฐานข้อมูลหลักมาใช้เลย โดยไม่ต้องมีการตรวจสอบก่อน เนื่องจากข้อมูลในฐานข้อมูลบัตรภาษีหลักได้ผ่านการตรวจสอบ จากกรมศุลกากรเรียบร้อยแล้ว ซึ่งระบบปัจจุบัน ต้องบันทึกบัตรภาษีที่ส่วนราชการนำมาชำระแทนเงินสด (นำบัตรภาษีมาใช้แทนเงินสด) ที่เครื่องคอมพิวเตอร์สำหรับบันทึกข้อมูล (ไม่ได้เชื่อมต่อกับฐานข้อมูล) แล้วจึงจัดเก็บขึ้นเทปแม่เหล็ก เป็นข้อมูลนำเข้าสู่ระบบงาน

3) ลดขั้นตอนการปฏิบัติงาน ในส่วนของการตรวจสอบ จำนวนเงินของบัตรภาษีกับจำนวนเงินใบนำส่ง ซึ่งระบบงานใหม่ จำนวนเงินในบัตรภาษีจะมีการผิดพลาดน้อยมาก เนื่องจากใช้ข้อมูลจากฐานข้อมูลหลักของระบบ (ในข้อ 1) ส่วนจำนวนเงินในใบนำส่ง จะผิดพลาดน้อยเช่นกัน ถ้าผิดพลาดก็เกิดจากการบันทึกจำนวนเงินของใบนำส่ง ซึ่งสามารถตรวจสอบได้ทันที เนื่องจากมีการโชว์ข้อมูลทางหน้าจอ ขณะที่กำลังบันทึกโดยที่มีการบันทึก กับเครื่องที่ต่อเชื่อมกับระบบ ซึ่งระบบปัจจุบัน มีการบันทึกใบนำส่งที่ส่วนประมวลบัญชี แล้วจัดเก็บขึ้นเทปแม่เหล็ก เป็นข้อมูลนำเข้าสู่ระบบ

3.3.2 ทางเลือกที่ 2

1) ใช้แผ่นดิสเกตต์เป็นสื่อกลาง ในการนำข้อมูลบัตรภาษีจากกรมศุลกากร เพื่อเป็นข้อมูลนำเข้าสู่ระบบเหมือนระบบปัจจุบัน

2) ลดขั้นตอนการปฏิบัติงาน ในส่วนของการตรวจสอบการบันทึกบัตรภาษีที่ผู้ประกอบการนำมาชำระ โดยบันทึกกับเครื่องที่เชื่อมต่อฐานข้อมูลบัตรภาษีโดยตรง ซึ่งจะบันทึกเฉพาะเลขที่บัตรภาษี เป็นรหัสที่ Unique พร้อมกับเปรียบเทียบข้อมูลที่บันทึกขณะนั้นกับข้อมูลในฐานข้อมูลหลัก ก็จะสมารถนำข้อมูลที่มี รหัสเลขที่บัตรภาษี จากฐานข้อมูลหลักมาใช้เลย โดยไม่

ต้องมีการตรวจสอบ เนื่องจากข้อมูลในฐานข้อมูลบัตรภาษีหลักได้ผ่านการตรวจสอบ จากกรม-ศุลกากรเรียบร้อยแล้ว ซึ่งระบบปัจจุบันต้องบันทึกบัตรภาษีที่ส่วนราชการนำมาชำระ ที่เครื่องคอมพิวเตอร์สำหรับบันทึกข้อมูล (ไม่ได้เชื่อมต่อกับฐานข้อมูล) แล้วจัดเก็บขึ้นเทปแม่เหล็ก เป็นข้อมูลนำเข้าระบบงาน

3) ลดขั้นตอนการปฏิบัติงาน ในส่วนของการตรวจสอบ จำนวนเงินของบัตรภาษีกับจำนวนเงินใบนำส่ง ซึ่งระบบงานใหม่ จำนวนเงินในบัตรภาษีจะมีการผิดพลาดน้อยมาก เนื่องจากใช้ข้อมูลจากฐานข้อมูลหลักของระบบ (ในข้อ 1)) ส่วนจำนวนเงินในใบนำส่ง จะผิดพลาดน้อยเช่นกัน ถ้าผิดพลาดก็เกิดจากการบันทึกจำนวนเงินของใบนำส่ง ซึ่งสามารถตรวจสอบได้ทันที เนื่องจากมีการโชว์ข้อมูลทางหน้าจอ ขณะที่กำลังบันทึกโดยที่มีการบันทึก กับเครื่องที่ต่อเชื่อมกับระบบ ซึ่งระบบปัจจุบัน ต้องบันทึกใบนำส่งที่ส่วนประมวลบัญชี แล้วจัดเก็บขึ้นเทปแม่เหล็ก เป็นข้อมูลนำเข้าระบบ

3.3.3 การประเมินทางเลือก

ทางเลือกที่ 1

- ข้อดี

เมื่อข้อมูลบัตรภาษี ของกรมศุลกากร มีการเปลี่ยนแปลงสามารถนำข้อมูลมาจัดเก็บข้อมูลลงในฐานข้อมูลหลักของระบบงาน ได้ทันที

- ข้อเสีย

1) สิ้นเปลืองค่าใช้จ่าย ในการจัดหาอุปกรณ์ต่างๆ เพื่อที่จะเชื่อมต่อฐานข้อมูลทั้ง 2 หน่วยงานเข้าด้วยกัน เนื่องจากปัจจุบันยังไม่มีเชื่อมต่อ

2) ต้องหาทำการศึกษาเพื่อหาวิธีการต่างๆ ในการที่จะทำให้ ฐานข้อมูลทั้งสอง ส่งข้อมูลกันได้อย่างถูกต้อง

ทางเลือกที่ 2

- ข้อดี

1) ประหยัดเงินงบประมาณ ไม่ต้องสิ้นเปลืองงบประมาณในการจัดหาอุปกรณ์ เชื่อมต่อระบบงาน ของทั้ง 2 หน่วยงาน

2) ปัญหาความล่าช้าของการปฏิบัติงานของระบบงาน ไม่ได้เกิดจากข้อมูลที่ได้จากกรมศุลกากร เพราะปัจจุบัน มีการนำข้อมูลมาจัดเก็บลงฐานข้อมูลบัตรภาษีหลัก สัปดาห์ละ 2 ครั้ง

- ข้อเสีย

ใช้เวลาในการบันทึกรายละเอียดของ ข้อมูลบัตรภาษี ที่ส่วนราชการนำมาชำระแทนเงินสดเพิ่มขึ้น เนื่องจากข้อมูลกรมศุลกากรนำเข้าฐานข้อมูลหลัก ของระบบไม่ทัน

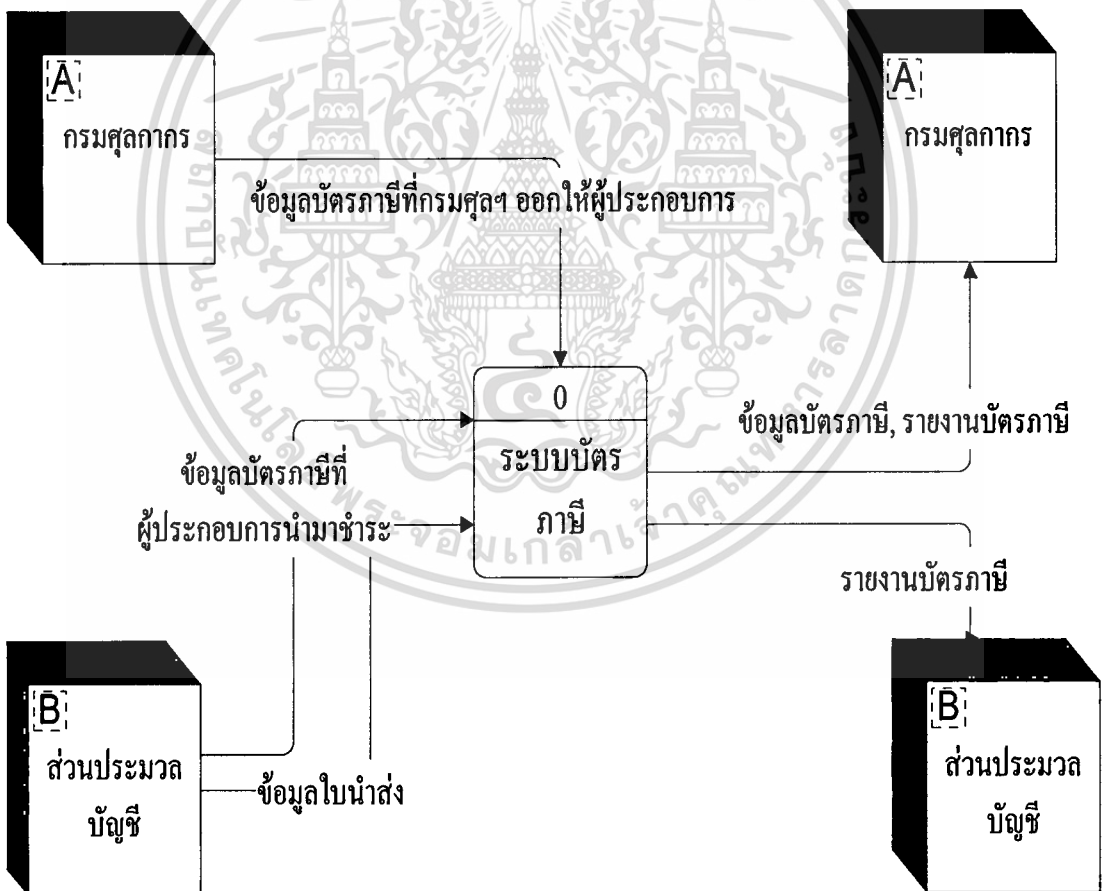
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากเปรียบเทียบข้อดีข้อเสียของแต่ละทางเลือก และพิจารณาถึงข้อจำกัดต่าง ๆ ได้แก่ เวลา บุคลากรและงบประมาณที่มีอย่างจำกัดแล้ว ผู้พัฒนาระบบ เลือกทางเลือกที่ 2

3.4 ระบบงานใหม่

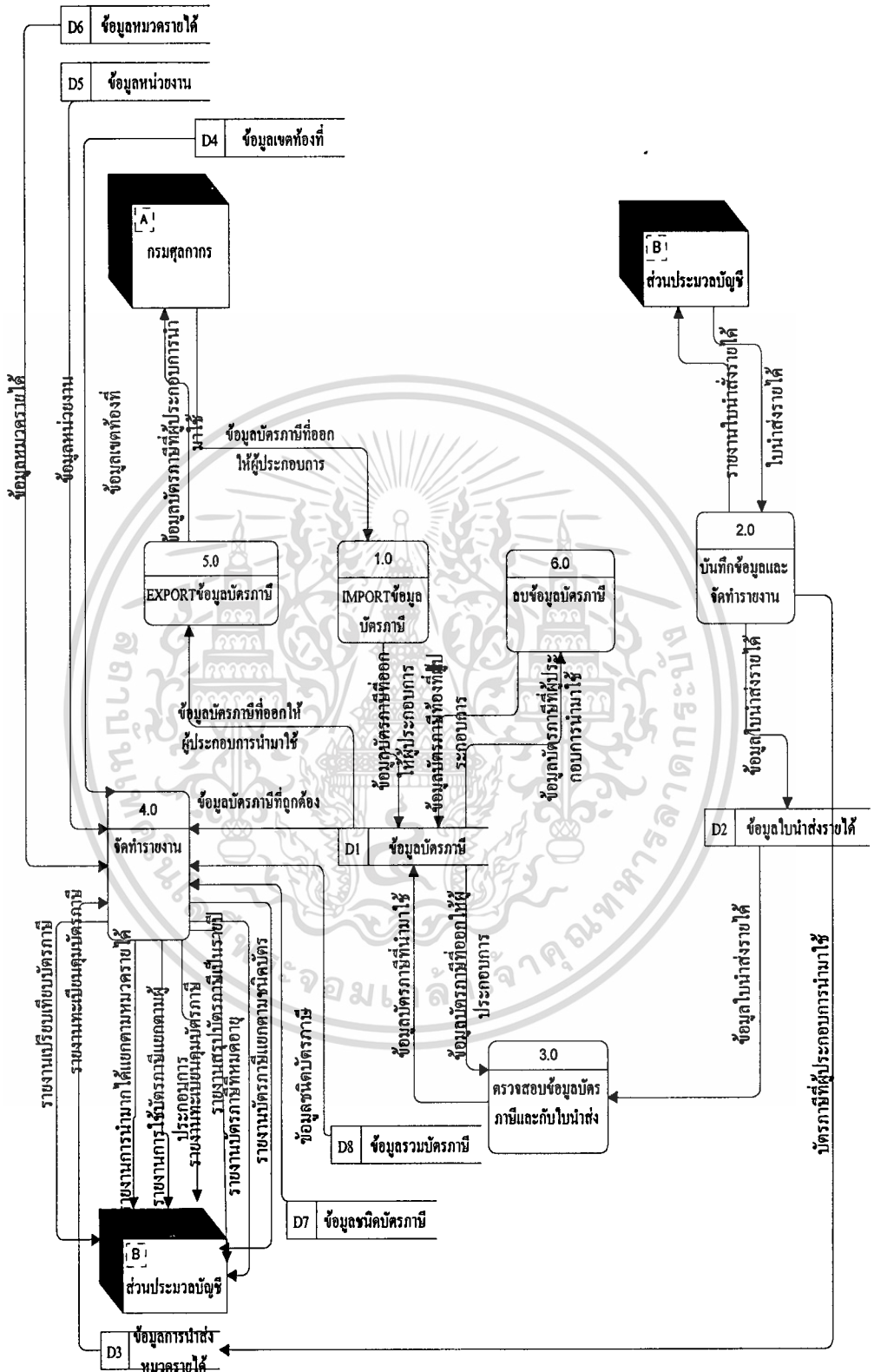
3.4.1 Data Flow Diagram ของระบบงานใหม่

Data Flow Diagram ของระบบใหม่ เป็น DFD ที่สร้างมาจาก DFD ของระบบงานปัจจุบัน และปรับเปลี่ยนตามความต้องการและความเหมาะสมของระบบใหม่เข้าไป จะประกอบไปด้วย Context Diagram, DFD Level 0 ซึ่งจะแสดงรายละเอียดต่าง ๆ ดังภาพที่ 3.9, ภาพที่ 3.10, ภาพที่ 3.11, ภาพที่ 3.12 และภาพที่ 3.13 และตาราง 3.1 แสดง I/o ของระบบ



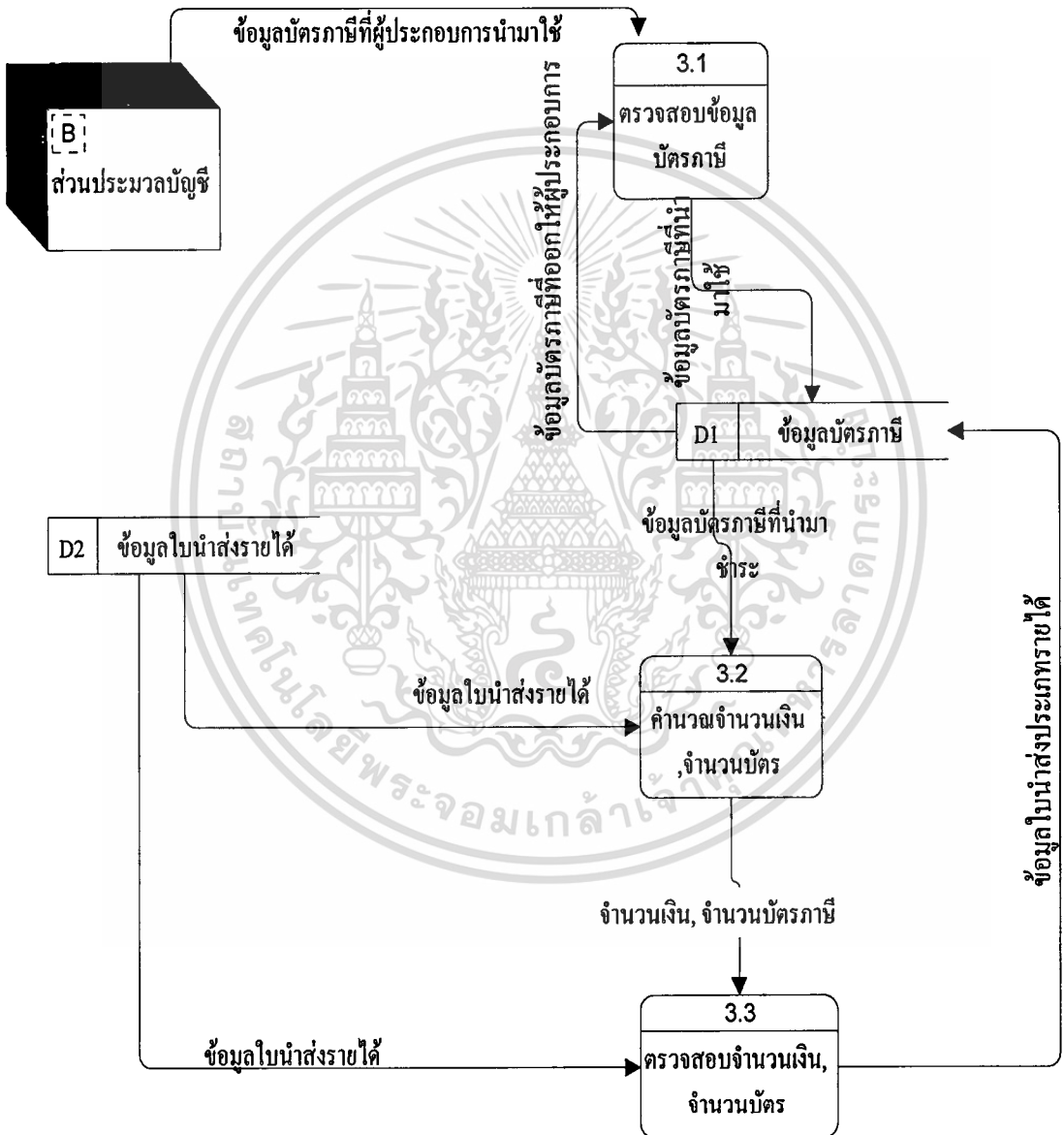
ภาพที่ 3.9 The New System Context Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



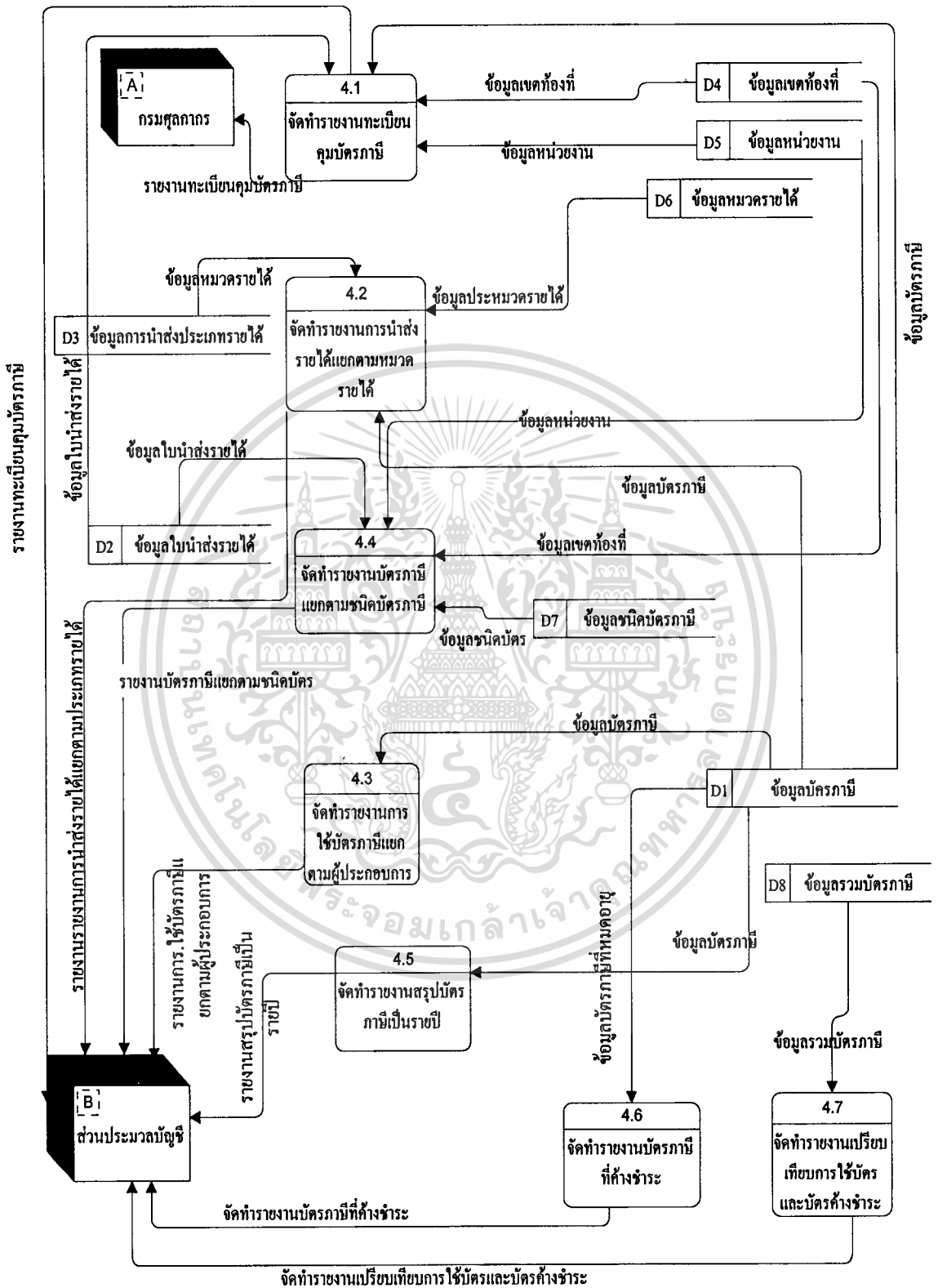
ภาพที่ 3.10 DFD Level 0 of The New System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.12 DFD Level 1 –Process 3 of The New System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.13 DFD Level 1 –Process 4 of The New System

เอกสารนี้เป็นเอกสารที่สงวนไว้ใช้กับโครงการนี้เท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 แสดง I/O Description ของระบบ

From	To	Data Flow Name	Data Content
กรมศุลกากร	Process 1.1	ข้อมูลบัตรภาษีที่ออกให้ผู้ประกอบการ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวเสียภาษี - ชื่อบริษัท
Process 1.1	D1	ข้อมูลบัตรภาษีที่ออกให้ผู้ประกอบการ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวเสียภาษี - ชื่อบริษัท
D1	Process 1.2	ข้อมูลบัตรภาษีที่ออกให้ผู้ประกอบการ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี
D1	Process 1.3	ข้อมูลบัตรภาษีที่ออกให้ผู้ประกอบการ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวเสียภาษี - ชื่อบริษัท
Process 1.2		ข้อมูลบัตรภาษีที่ออกให้ผู้ประกอบการ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี
Process 1.3	D1	ข้อมูลบัตรภาษีที่ออกให้ผู้ประกอบการ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านนอกรั้ว

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
		ประกอบการ	<ul style="list-style-type: none"> - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวเสียภาษี - ชื่อบริษัท
ส่วน ประมวล บัญชี	Process 2.1	ข้อมูลใบนำส่งรายได้	<ul style="list-style-type: none"> - ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ - ว.ค.ป.ที่นำส่ง - ว.ค.ป.ที่รับ - จำนวนบัตรภาษี - จำนวนเงิน - รหัสเขตท้องที่ - รหัสหน่วยงาน - รหัสหมวดรายได้
Process 2.1	D2	ข้อมูลใบนำส่งรายได้	<ul style="list-style-type: none"> - ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ - ว.ค.ป.ที่นำส่ง - ว.ค.ป.ที่รับ - จำนวนบัตรภาษี - จำนวนเงิน - รหัสเขตท้องที่ - รหัสหน่วยงาน - รหัสหมวดรายได้
Process 2.1	D3	ข้อมูลนำส่งหมวดรายได้	<ul style="list-style-type: none"> - ปี พ.ศ.ที่นำส่ง - เลขที่คลังรับ - รหัสหมวดรายได้ - จำนวนเงิน
D2	Process 2.2	ข้อมูลใบนำส่ง	<ul style="list-style-type: none"> - ปี พ.ศ. ที่นำส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะในพิธีการที่เกี่ยวเนื่องเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้เพื่อวัตถุประสงค์อื่นใด

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
			<ul style="list-style-type: none"> - เลขที่คลังรับ - ว.ค.ป.ที่นำส่ง - ว.ค.ป.ที่รับ - จำนวนบัตรภาษี - จำนวนเงิน - รหัสเขตท้องที่ - รหัสหน่วยงาน - รหัสหมวดรายได้
D3	Process 4.2	ข้อมูลนำส่งหมวดรายได้	<ul style="list-style-type: none"> - ปี พ.ศ.ที่นำส่ง - รหัสหมวดรายได้ - จำนวนเงิน
D2	Process 2.3	ข้อมูลใบนำส่ง	<ul style="list-style-type: none"> - ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ - ว.ค.ป.ที่นำส่ง - ว.ค.ป.ที่รับ - จำนวนบัตรภาษี - จำนวนเงิน - รหัสเขตท้องที่ - รหัสหน่วยงาน - รหัสหมวดรายได้
D4	Process 2.3	ข้อมูลเขตท้องที่	<ul style="list-style-type: none"> - รหัสเขตท้องที่ - ชื่อเขตท้องที่
D5	Process 2.3	ข้อมูลหน่วยงาน	<ul style="list-style-type: none"> - รหัสหน่วยงาน - ชื่อหน่วยงาน
D6	Process 2.3	ข้อมูลหมวดรายได้	<ul style="list-style-type: none"> - รหัสหมวดรายได้ - ชื่อหมวดรายได้
D3	Process 2.3	ข้อมูลการนำส่งหมวดรายได้	<ul style="list-style-type: none"> - ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ - รหัสหมวดรายได้

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของไอซีทีประเทศไทยและใช้เฉพาะเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
			- จำนวนเงิน
ส่วน ประมวล บัญชี	Process 3.1	ข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาใช้	- เลขที่บัตรภาษี
Process 3.1	D1	ข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาใช้	- เลขที่บัตรภาษี
D1	Process 3.1	ข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาใช้	- เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวเสียภาษี - ชื่อบริษัท
D1	Process 3.2	ข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาใช้	- เลขที่บัตรภาษี - มูลค่าของบัตร
Process 3.2	D1	ข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาใช้	- เลขที่บัตรภาษี - มูลค่าของบัตร
D2	Process 3.2	ข้อมูลใบนำส่งรายได้	- ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ
D2	Process 3.3	ข้อมูลใบนำส่งรายได้	- ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ - จำนวนเงิน - จำนวนบัตร
D1	Process 3.3	ข้อมูลบัตรภาษีที่นำมาใช้	- จำนวนบัตร - จำนวนเงิน
Process 3.3	D1	ข้อมูลใบนำส่งรายได้	- ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ
D1	Process 4.1	ข้อมูลบัตรภาษีที่นำมาใช้ แทนเงินสด	- เลขที่บัตรภาษี - ชนิดบัตรภาษี

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ได้รับอนุญาตจากทางผู้จัดทำ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
			<ul style="list-style-type: none"> - ว.ค.ป ที่ออกบัตร - ว.ค.ป บัตรหมดอายุ - มูลค่าบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
D4	Process 4.1	ข้อมูลเขตท้องที่	<ul style="list-style-type: none"> - รหัสเขตท้องที่ - ชื่อเขตท้องที่
D5	Process 4.1	ข้อมูลหน่วยงาน	<ul style="list-style-type: none"> - รหัสหน่วยงาน - ชื่อหน่วยงาน
D2	Process 4.1	ข้อมูลใบนำส่งรายได้	<ul style="list-style-type: none"> - ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ
Process 4.1	ส่วน ประมวล บัญชี	รายงานทะเบียนคุมบัตรภาษี	
Process 4.1	กรม ศุลกากร	รายงานทะเบียนคุมบัตรภาษี	
D2	Process 4.2	ข้อมูลหมวดรายได้	<ul style="list-style-type: none"> - รหัสหมวดรายได้ - ชื่อหมวดรายได้
D3	Process 4.2	ข้อมูลนำส่งหมวดรายได้	<ul style="list-style-type: none"> - ปี พ.ศ. นำส่ง - เลขที่คลังรับ - รหัสหมวดรายได้ - จำนวนเงิน
Process 4.2	ส่วน ประมวล บัญชี	รายงานสรุปการนำส่งรายได้ แยกตามหมวดรายได้	
D1	Process 4.3	ข้อมูลบัตรภาษีที่นำมาใช้ แทนเงินสด	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. บัตรหมดอายุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในวงเพื่อวัตถุประสงค์เฉพาะเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่หรือใช้เพื่อวัตถุประสงค์อื่นใด

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
			<ul style="list-style-type: none"> - มูลค่าบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
Process 4.3	ส่วน ประมวล บัญชี	รายงานสรุปการชำระบัตร ภาษี แยกตามผู้ประกอบการ	
D2	Process 4.4	ข้อมูลใบนำส่งรายได้	<ul style="list-style-type: none"> - ปี พ.ศ. ที่นำส่ง - เลขที่คลังรับ - จำนวนบัตรที่นำส่ง - จำนวนเงินที่นำส่ง - รหัสเขตท้องที่ - รหัสหน่วยงาน
D4	Process 4.4	ข้อมูลเขตท้องที่	<ul style="list-style-type: none"> - รหัสเขตท้องที่ - ชื่อเขตท้องที่
D5	Process 4.4	ข้อมูลหน่วยงาน	<ul style="list-style-type: none"> - รหัสหน่วยงาน - ชื่อหน่วยงาน
Process 4.4	ส่วน ประมวล บัญชี	รายงานบัตรภาษีแยกตาม ชนิดบัตรภาษี	
D1	Process 4.5	ข้อมูลบัตรภาษี (ที่นำมาใช้ แทนเงินสด)	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - จำนวนเงิน - ปี พ.ศ. ที่นำส่ง
Process 4.5	ส่วน ประมวล บัญชี	รายงานสรุปบัตรภาษีเป็นราย ปี	
D1	Process 4.6	ข้อมูลบัตรภาษีที่หมดอายุ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - วัตถุประสงค์ - วัตถุประสงค์หมดอายุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในหน่วยงานที่ออกเอกสารนี้ ไม่สามารถเผยแพร่หรือใช้เพื่อวัตถุประสงค์อื่นได้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
			<ul style="list-style-type: none"> - มูลค่าของบัตร - เลขประจำตัวเสียภาษี - ชื่อบริษัท
Process 4.6	ส่วน ประมวล บัญชี	รายงานสรุปบัตรภาษีที่ค้าง ชำระเป็นรายปี	
D1	Process 5.1	ข้อมูลบัตรภาษี (ที่ผู้ประกอบ การนำมาใช้แทนเงินสด)	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
Process 5.1	D1	ข้อมูลบัตรภาษี (ที่ผู้ประกอบ การนำมาใช้แทนเงินสด)	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
D1	Process 5.2	ข้อมูลบัตรภาษี (ที่ผู้ประกอบ การนำมาใช้แทนเงินสด)	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
Process 5.2	กรม ศุลกากร	ข้อมูลบัตรภาษี (ที่ผู้ประกอบ การนำมาใช้แทนเงินสด)	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้ภายในหน่วยงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
			<ul style="list-style-type: none"> - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
D1	Process 6.1	ข้อมูลบัตรภาษี	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
Process 6.1	D1	ข้อมูลบัตรภาษี	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
D1	Process 6.2	ข้อมูลบัตรภาษี	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
Process 6.2	D1		
D1	Process 6.3	ข้อมูลบัตรภาษี	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้วยการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

From	To	Data Flow Name	Data Content
			<ul style="list-style-type: none"> - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
Process 6.3	D1	ข้อมูลบัตรภาษี	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
D1	Process 6.4	ข้อมูลบัตรภาษีที่หมดอายุ	<ul style="list-style-type: none"> - เลขที่บัตรภาษี - ชนิดบัตรภาษี - ว.ค.ป. ที่ออกบัตร - ว.ค.ป. ที่บัตรหมดอายุ - มูลค่าของบัตร - เลขประจำตัวผู้เสียภาษี - ชื่อบริษัท
Process 6.4	D1		

3.4.2 Entity Relationship Model (E-R MODEL) ของระบบงานใหม่

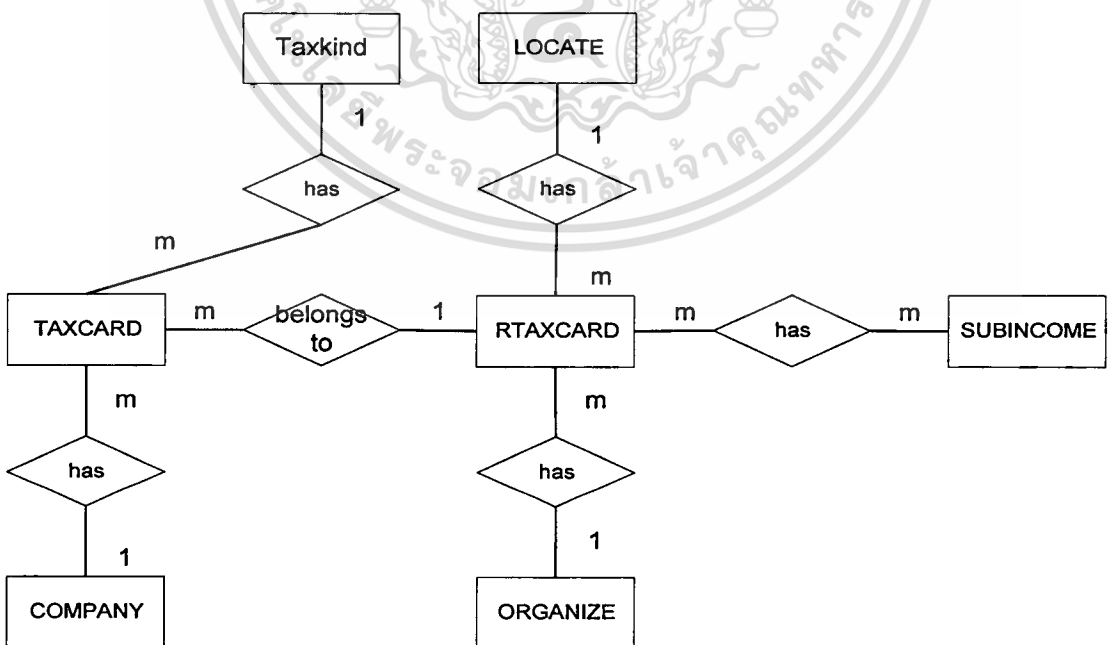
ในการพัฒนาระบบจะใช้ E-R Diagram แสดงถึงความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่เกี่ยวข้องและทำให้ทราบถึง Entity Type หรือเรียกว่า Entity และ Entity Instance ของระบบ จากภาพที่ 3.14 The E-R Diagram for Taxcard ประกอบด้วย 7 Entity Type คือ COMPANY, TAXCARD, TAXKIND, RTAXCARD, SUBINCOME, ORGANIZE และ LOCATE และแสดง Relationship ได้ดังภาพที่ 3.15 โดยแต่ละ Entity มีความสัมพันธ์กันดังนี้

- ระหว่างบริษัทที่ประกอบการส่งสินค้าออก (COMPANY) กับ บัตรภาษี (TAXCARD) มีความสัมพันธ์กันในลักษณะ One-To-Many

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

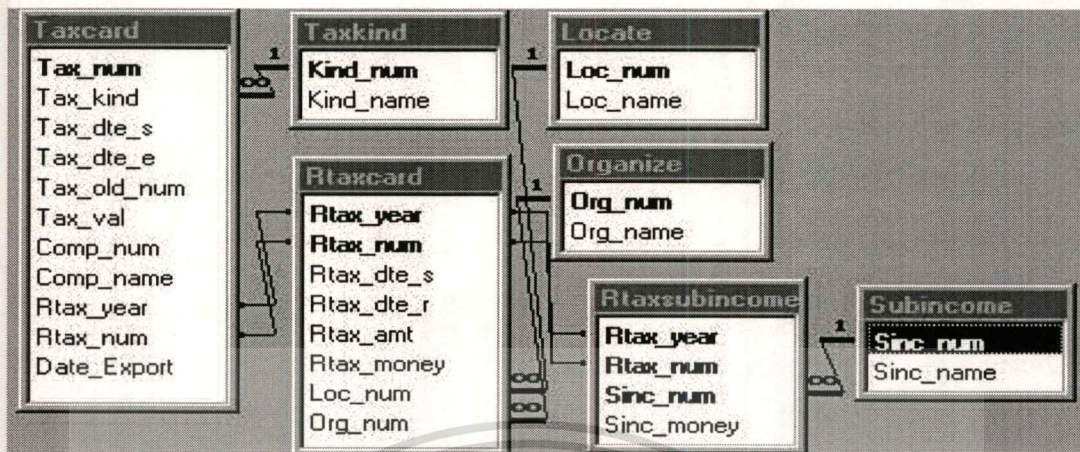
- ระหว่างใบนำส่งเงินรายได้แผ่นดิน (RTAXCARD) กับ บัตรภาษี (TAXCARD) มีความสัมพันธ์กันในลักษณะ One-To-Many
- ระหว่างชนิดบัตร (TAXKIND) กับ บัตรภาษี (TAXCARD) มีความสัมพันธ์กันในลักษณะ One-To-Many
- ระหว่างหน่วยงานที่มีหน้าที่จัดเก็บรายได้ (ORGANIZE) กับ ใบนำส่งเงินรายได้แผ่นดิน (RTAXCARD) มีความสัมพันธ์กันในลักษณะ One-To-Many
- ระหว่างเขตท้องที่มีหน้าที่จัดเก็บรายได้ (LOCATE) กับใบนำส่งเงินรายได้แผ่นดิน (RTAXCARD) มีความสัมพันธ์กันในลักษณะ One-To-Many
- ระหว่างใบนำส่งเงินรายได้แผ่นดิน (RTAXBILLS) กับ หมวครายได้ (SUBINCOME) มีความสัมพันธ์กันในลักษณะ One-To-Many

หลังจากได้ E-R Diagram ของระบบแล้วนำมา Normalize จะได้ตารางต่าง ๆ ของฐานข้อมูลในระบบพร้อมทั้งรายละเอียดดังตารางที่ 3.2 Data Dictionary for The Taxcard Database แต่หลังจากที่ทำ Normalize และได้ศึกษาข้อมูลในระบบจะพบว่าข้อมูลในส่วนของชื่อบริษัท (COMPANY) จะมีกรมศุลกากรเป็นหน่วยงานที่ดูแลอยู่ ดังนั้นจึงยอมให้มีการซ้ำซ้อนของข้อมูลเพื่อความสะดวก รวดเร็วในการประมวลผลและไม่ต้องดูแลในการเพิ่ม แก้ไขข้อมูลในส่วนนี้ ในการพัฒนาระบบจึงใช้ข้อมูลตามตารางที่ 3.3 แทน



ภาพที่ 3.14 The E-R Diagram of the New System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- Relationship ในส่วนที่เกี่ยวข้องกับข้อมูลในระบบ



- Relationship ในส่วนของผู้ใช้ระบบ

ภาพที่ 3.15 The Relationship of the New System

ตารางที่ 3.2 Data Dictionary for the Taxcard Database

TABLE NAME	ATTRIBUTE NAME	CONTENTS	TYPE	LENGTH	KEY
TAXCAED	TAX_NUM	เลขที่บัตรภาษี	CHAR	9	PK
	KIND_NUM	รหัสชนิดบัตรภาษี	INT	2	FK
	TAX_DTE_S	ว.ด.ป. ที่ออกบัตร	DATE	8	
	TAX_DTE_E	ว.ด.ป. บัตรหมดอายุ	DATE	8	
	TAX_OLD_N	เลขที่บัตรเดิม	CHAR	9	
	TAX_VAL	มูลค่าของบัตร	NUM	8(6.2)	
	COMP_NUM	เลขประจำตัวผู้เสียภาษี	INT	10	FK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านกรรค่า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 Data Dictionary for the Taxcard Database (ต่อ)

TABLE NAME	ATTRIBUTE NAME	CONTENTS	TYPE	LENGTH	KEY
COMPANY	COMP_NUM	เลขประจำตัวผู้เสียภาษี	INT	10	PK
	COMP_NAME	ชื่อบริษัท	CHAR	59	
RTAXCARD	RTAX_YEAR	ปี พ.ศ.ที่นำส่ง	CHAR	4	PK
	RTAX_NUM	เลขที่คั้งรับ	CHAR	5	PK
	RTAX_DTE_S	ว.ค.ป ที่นำส่ง	DATE	8	
	RTAX_DTE_R	ว.ค.ป ที่รับใบนำส่ง	DATE	8	
	RTAX_AMT	จำนวนบัตรภาษีที่ส่ง	INT	4	
	RTAX_MNEY	จำนวนเงินที่นำส่ง	NUM	15(13.2)	
	LOC_NUM	รหัสเขตท้องที่	INT	6	FK
	ORG_NUM	รหัสหน่วยงาน	INT	5	FK
	SINC_NUM	รหัสหมวดรายได้	INT	4	FK
SUBINCOME	SINC_NUM	รหัสหมวดรายได้	INT	4	PK
	SINC_NAME	ชื่อหมวดรายได้	CHAR	76	
ORGANIZE	ORG_NUM	รหัสหน่วยงาน	INT	5	PK
	ORG_NAME	ชื่อหน่วยงาน	CHAR	75	
LOCATE	LOC_NUM	รหัสเขตท้องที่	INT	6	PK
	LOC_NAME	ชื่อเขตท้องที่	CHAR	74	
TAXKIND	KIND_NUM	รหัสชนิดบัตร	INT	2	PK
	KIND_NAME	ชนิดบัตร	CHAR	20	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 Data Dictionary for the Taxcard Database

TABLE NAME	ATTRIBUTE NAME	CONTENTS	TYPE	LENGTH	KEY
TAXCAED	TAX_NUM	เลขที่บัตรภาษี	CHAR	9	PK
	KIND_NUM	ชนิดบัตรภาษี	INT	2	FK
	TAX_DTE_S	ว.ค.ป ที่ออกกบัตร	DATE	8	
	TAX_DTE_E	ว.ค.ป บัตรหมดอายุ	DATE	8	
	TAX_OLD_N	เลขที่บัตรเดิม	CHAR	9	
	TAX_VAL	มูลค่าของบัตร	NUM	8(6.2)	
	COMP_NUM	เลขประจำตัวผู้เสียภาษี	INT	10	
	COMP_NAME	ชื่อบริษัท	CHAR	59	
RTAXCARD	RTAX_YEAR	ปี พ.ศ.ที่นำส่ง	CHAR	4	PK
	RTAX_NUM	เลขที่คลังรับ	CHAR	5	PK
	RTAX_DTE_S	ว.ค.ป ที่นำส่ง	DATE	8	
	RTAX_DTE_R	ว.ค.ป ที่รับใบนำส่ง	DATE	8	
	RTAX_AMT	จำนวนบัตรภาษีที่ส่ง	INT	4	
	RTAX_MNEY	จำนวนเงินที่นำส่ง	NUM	15(13.2)	
	LOC_NUM	รหัสเขตท้องที่	INT	6	FK
	ORG_NUM	รหัสหน่วยงาน	INT	5	FK
SUBINCOME	SINC_NUM	รหัสหมวดรายได้	INT	4	PK
	SINC_NAME	ชื่อหมวดรายได้	CHAR	76	
ORGANIZE	ORG_NUM	รหัสหน่วยงาน	INT	5	PK
	ORG_NAME	ชื่อหน่วยงาน	CHAR	75	
LOCATE	LOC_NUM	รหัสเขตท้องที่	INT	6	PK
	LOC_NAME	ชื่อเขตท้องที่	CHAR	74	
TAXKIND	KIND_NUM	รหัสชนิดบัตรภาษี	INT	2	PK
	KIND_NAME	ชนิดบัตรภาษี	CHAR	20	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การสร้าง Application

องค์ประกอบหลักที่ใช้ในการพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัตรภาษี แบ่งออกได้เป็น 2 ส่วน คือ

4.1 ฮาร์ดแวร์ (Hardware)

อุปกรณ์ฮาร์ดแวร์ที่ใช้ในการพัฒนาระบบ และใช้งานโปรแกรม จะใช้เครื่องไมโครคอมพิวเตอร์ ซึ่งมีคุณสมบัติ ดังนี้

- หน่วยประมวลผลกลาง (CPU) ชนิด Pentium III 330 ขึ้นไป
- หน่วยความจำหลัก (RAM) 32 MB หรือมากกว่า
- หน่วยความจำสำรอง (HARDDISK) ขนาด 4.2 GB หรือมากกว่า
- จอแสดงผล (Monitor) ชนิด SVGA ขนาด 800 x 600 จุด 256 สี หรือมากกว่า

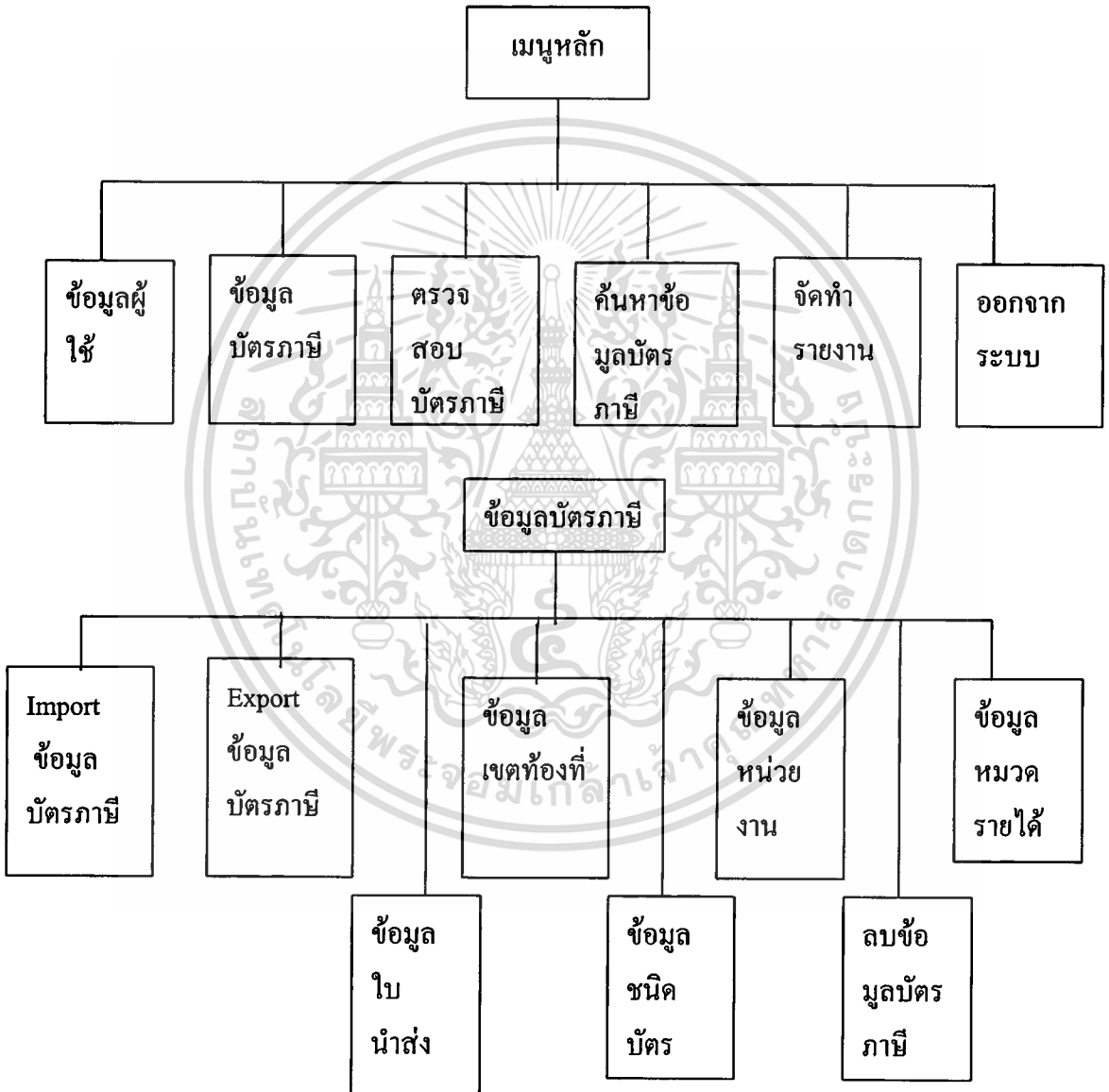
4.2 ซอฟต์แวร์ (Software)

ระบบซอฟต์แวร์ที่ใช้ในการพัฒนาระบบจะประกอบด้วยส่วนต่างๆ ได้แก่ ทั้งระบบปฏิบัติการ (OS) โปรแกรมที่ใช้ในการพัฒนาระบบ (Application Generator) และเครื่องมือ (Tools) ที่ช่วยในการวิเคราะห์และออกแบบรายงาน โดยมีรายละเอียด ดังนี้

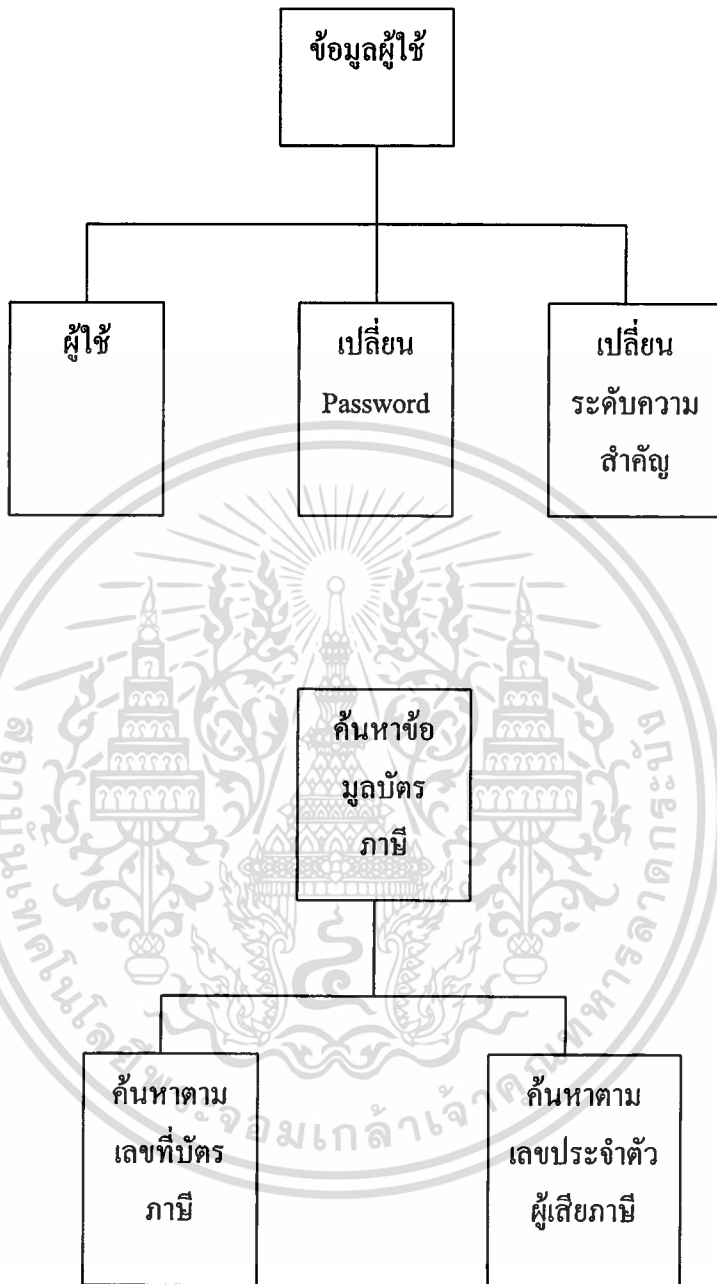
- ระบบปฏิบัติการ (OS) ใช้ระบบปฏิบัติการ WINDOWS 95
- โปรแกรมที่ใช้ในการจัดทำดาต้าเบส (Database) คือ Microsoft Access 97
- โปรแกรมที่ใช้ในการจัดทำพัฒนาระบบ คือ Microsoft Visual Basic 6.0 Enterprise
- เครื่องมือ (Tools) ที่ใช้ในการวิเคราะห์และออกแบบระบบ เพื่อให้เป็นไปตามข้อกำหนด โดยใช้ Visio Version 5 สำหรับสร้าง แผนภาพรวม (Context Diagram) และสร้างการไหลของข้อมูล (Data Flow Diagram) ตลอดจนทำแผนภาพ E-R Diagram

4.3 การออกแบบหน้าจอ

การออกแบบหน้าจอจะใช้ตารางข้อมูลที่ได้จากการออกแบบฐานข้อมูล โดยหน้าจอที่ได้จะครอบคลุมการทำงานของทั้งระบบ ดังแสดงดังภาพที่ 4.1

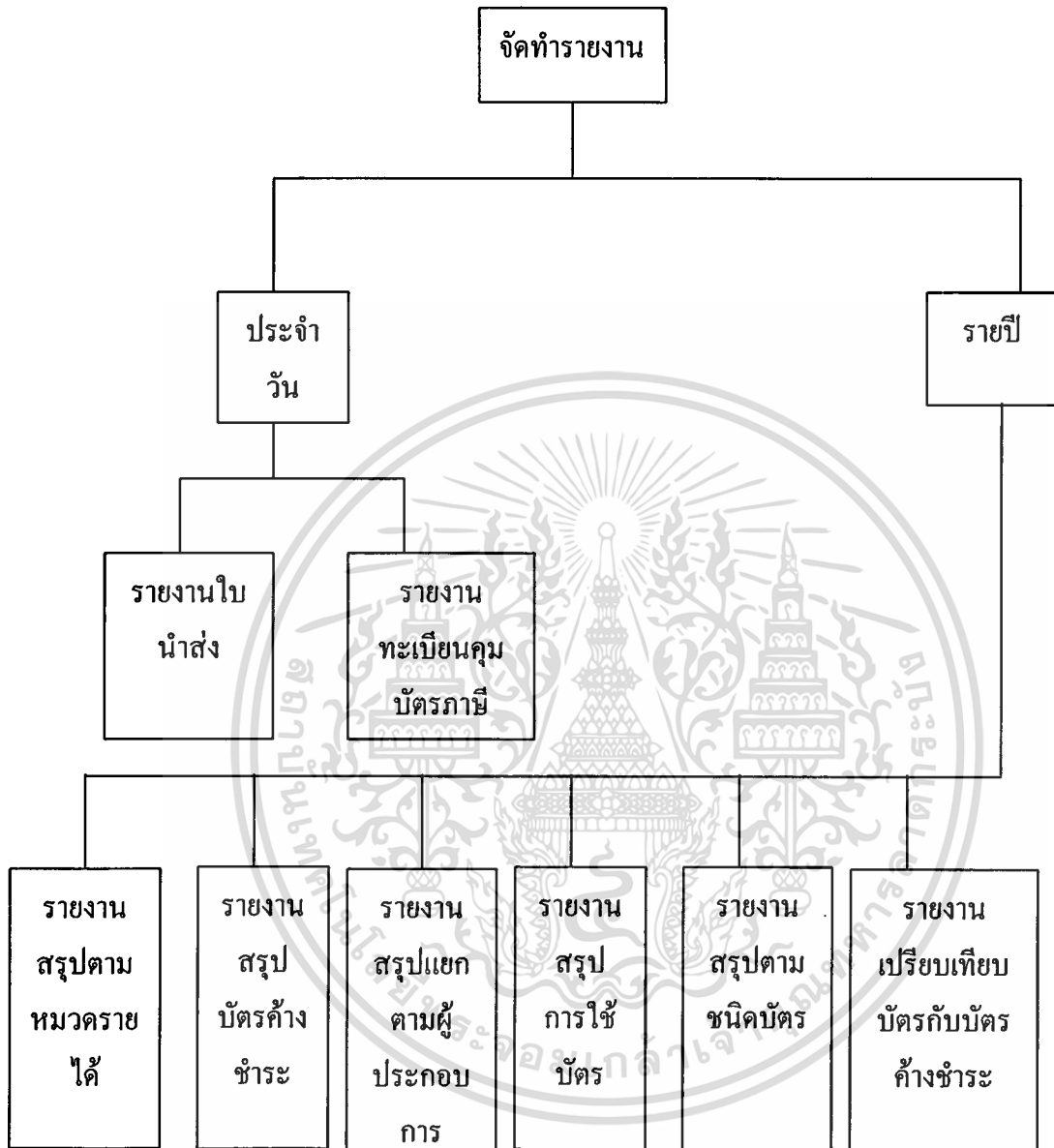


ภาพที่ 4.1 แสดง Menu Structure



ภาพที่ 4.1 แสดง Menu Structure (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.1 แสดง Menu Structure (ต่อ)

4.4 การพัฒนาโปรแกรม

หลังจากที่ได้มีการออกแบบ Menu Structure แล้ว จึงได้ทำการพัฒนาโปรแกรมโดยใช้ Microsoft Visual Basic 6.0 และจัดเก็บข้อมูลของระบบโดยใช้ Microsoft Access ซึ่งประกอบด้วยส่วนต่างๆ ดังนี้

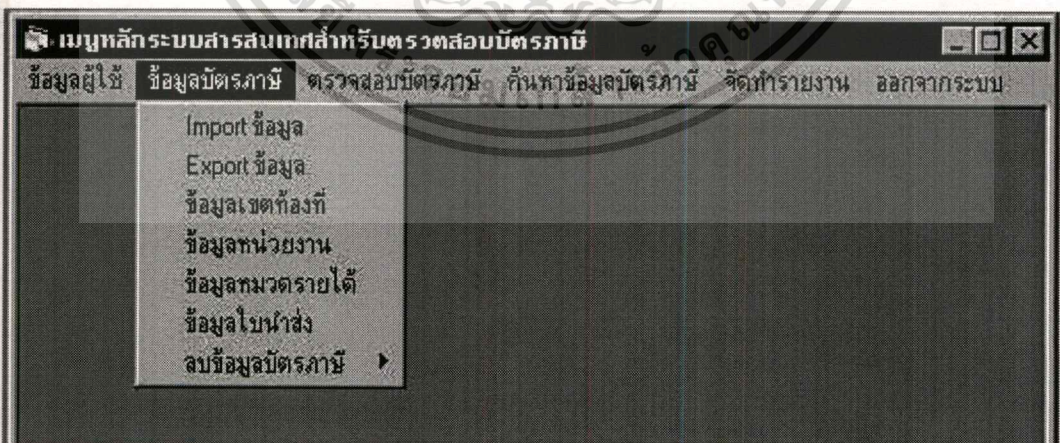
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) การเข้าสู่ระบบ ดังภาพที่ 4.2



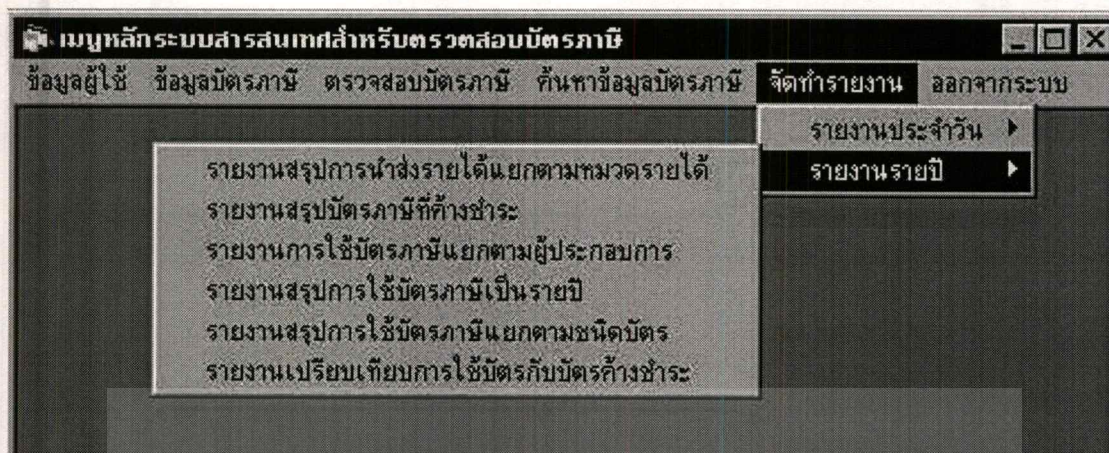
ภาพที่ 4.2 แสดงหน้าจอการเข้าไปใช้ระบบ

2) เมนูหลักของระบบ ซึ่งประกอบด้วยรายการต่าง ๆ สำหรับการใช้งานในระบบดังภาพที่ 4.3 และ 4.4



ภาพที่ 4.3 แสดงรายการในเมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.4 แสดงรายการในเมนูหลัก

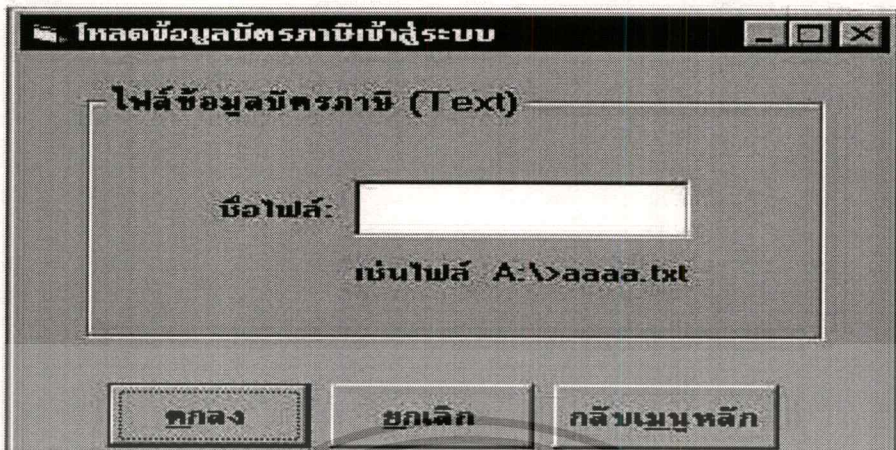
4.4.1 หน้าจอสำหรับการบันทึกข้อมูล

หลังจากผู้ใช้งานเข้ามาใช้งานแล้วและเลือกรายการเมนูย่อยจากเมนูหลักสำหรับการบันทึกข้อมูลซึ่งแต่ละหน้าจอจะมีฟังก์ชันของการทำงาน (ปุ่มกด) ได้แก่

- ปุ่มเพิ่ม สำหรับเพิ่มข้อมูลเพื่อจัดเก็บลงฐานข้อมูล
- ปุ่มค้นหา สำหรับค้นหาข้อมูลขึ้นมาโชว์ที่หน้าจอ
- ปุ่มแก้ไข สำหรับแก้ไขข้อมูล
- ปุ่มลบ สำหรับลบข้อมูลออกจากฐานข้อมูล
- ปุ่มยกเลิก สำหรับยกเลิกการทำงาน
- ปุ่มกลับเมนูหลัก สำหรับกลับสู่เมนูหลัก

4.4.1.1 หน้าจอสำหรับ Import ซึ่งใช้สำหรับโหลดข้อมูลบัตรภาษีที่ออกโดยกรมศุลกากรให้ผู้ประกอบการส่งสินค้าออกเข้าสู่ฐานข้อมูล ดังภาพที่ 4.5

4.4.1.2 หน้าจอสำหรับ Export เป็นการนำข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาชำระแทนเงินสดหลังจากที่ได้ผ่านการตรวจสอบข้อมูลบัตรภาษีในระบบแล้วจัดเก็บลงแผ่นดิสเกตต์ เพื่อนำส่งกรมศุลกากร ดังภาพที่ 4.6



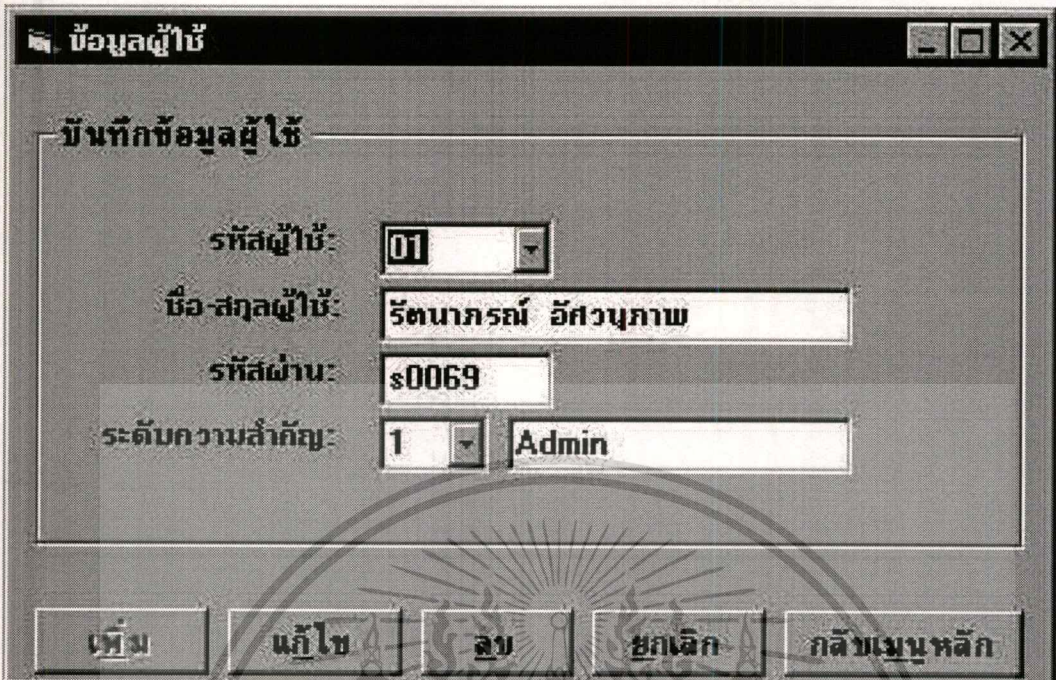
ภาพที่ 4.5 แสดงหน้าจอ Import ข้อมูลบัตรภาษี



ภาพที่ 4.6 แสดงหน้าจอ Export ข้อมูลบัตรภาษี

4.4.1.3 หน้าจอสำหรับการบันทึกข้อมูลผู้ใช้ ซึ่งจะมีรายละเอียดเกี่ยวกับผู้ใช้ ดังภาพที่ 4.7 สำหรับการกำหนดสิทธิการเข้าไปใช้ ดังภาพที่ 4.8 และสำหรับเปลี่ยนแปลงรหัสผ่านดังภาพที่ 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ข้อมูลผู้ใช้

บันทึกข้อมูลผู้ใช้

รหัสผู้ใช้: 01

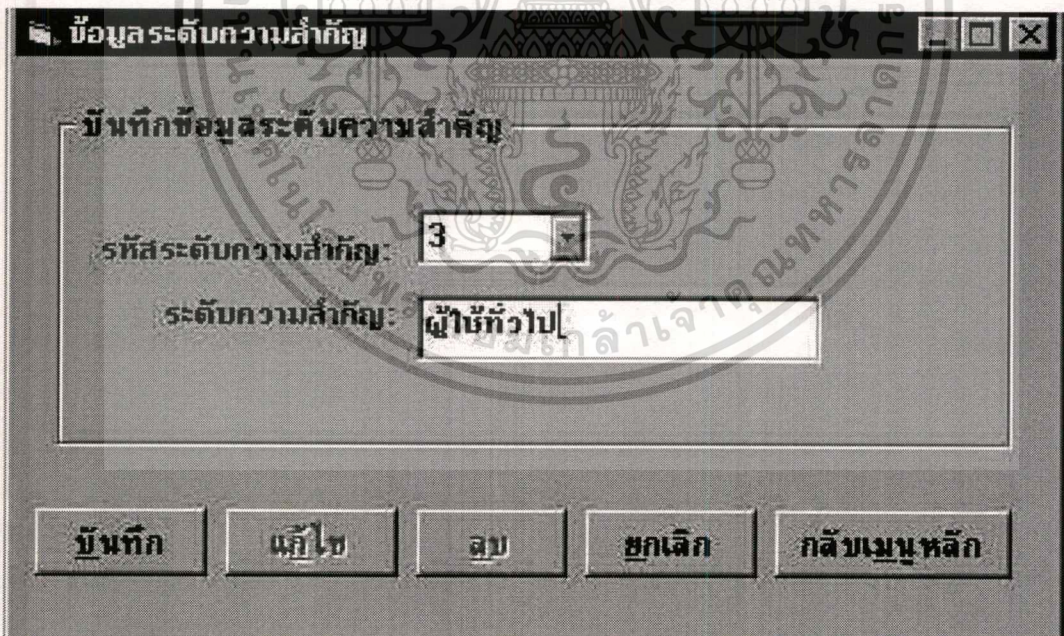
ชื่อ-สกุลผู้ใช้: รัตนภรณ์ อัสวบุญภาพ

รหัสผ่าน: ๙0069

ระดับความสำคัญ: 1 Admin

เพิ่ม แก้ไข ลบ ยกเลิก กลับเมนูหลัก

ภาพที่ 4.7 แสดงหน้าจอการบันทึกข้อมูลผู้ใช้



ข้อมูลระดับความสำคัญ

บันทึกข้อมูลระดับความสำคัญ

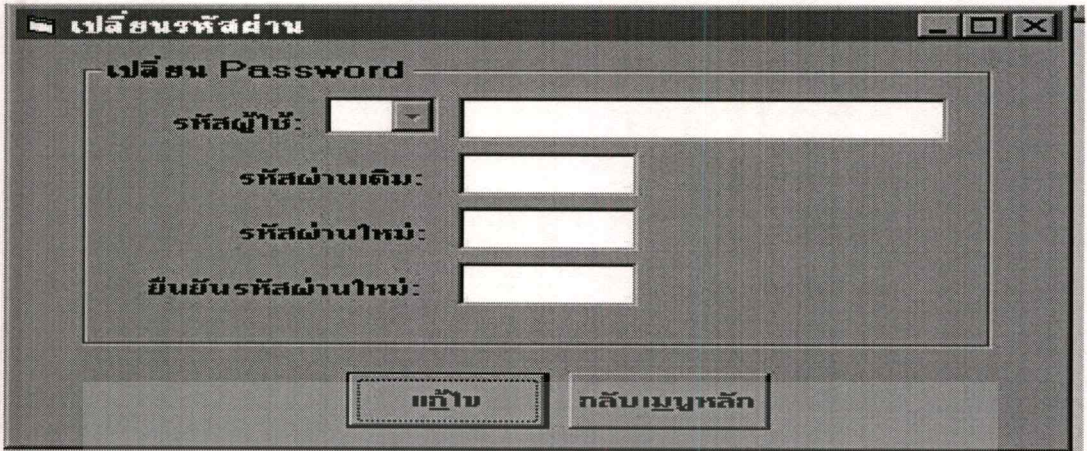
รหัสระดับความสำคัญ: 3

ระดับความสำคัญ: ผู้ใช้ทั่วไป

บันทึก แก้ไข ลบ ยกเลิก กลับเมนูหลัก

ภาพที่ 4.8 แสดงหน้าจอการกำหนดสิทธิ์ผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.9 แสดงหน้าจอการเปลี่ยนรหัสผ่าน

4.4.1.4 หน้าจอสำหรับการบันทึกข้อมูลพื้นฐานของระบบ ประกอบด้วยหน้าจอต่าง ๆ ดังนี้
คือ

1) ข้อมูลเขตท้องที่ จะบันทึกข้อมูลสำหรับเขตท้องที่ ซึ่งเป็นหน่วยงานผู้นำส่งราย
ได้จะมีรายละเอียด ดังภาพที่ 4.10



ภาพที่ 4.10 แสดงหน้าจอสำหรับบันทึกข้อมูลเขตท้องที่

2) ข้อมูลหน่วยงาน สำหรับบันทึกข้อมูลหน่วยงาน ซึ่งเป็นหน่วยงานผู้นำส่งราย
ได้จะมีรายละเอียด ดังภาพที่ 4.11 เป็นการเพิ่มข้อมูลหน่วยงานในรหัส 03080 ยังไม่ได้บันทึกชื่อ
หน่วยงานแต่ไปกดปุ่มบันทึกเพื่อเพิ่มข้อมูลก่อน จะมีข้อความเตือนให้บันทึกชื่อหน่วยงานให้ถูก
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้อง เพราะว่าการเพิ่มหรือแก้ไขข้อมูลระบบจะบังคับให้มีการบันทึกข้อมูลให้ครบทุกช่องก่อนเสมอ ซึ่งจะเป็นในลักษณะนี้เหมือน ๆ กันทุกหน้าจอ

ภาพที่ 4.11 แสดงหน้าจอสำหรับบันทึกข้อมูลหน่วยงาน

3) ข้อมูลหมวดรายได้ สำหรับบันทึกข้อมูลหมวดรายได้ที่นำส่งรายได้แผ่นดิน ซึ่งมีรายละเอียด ดังภาพที่ 4.12 เป็นการเรียกข้อมูลหมวดรายได้ขึ้นมาแก้ไข เมื่อมีการกดปุ่มแก้ไขแล้วจะไม่ยอมให้แก้ไขรหัสหมวดรายได้ จะให้แก้ไขได้เฉพาะหมวดรายได้เท่านั้น

ภาพที่ 4.12 แสดงหน้าจอสำหรับบันทึกข้อมูลหมวดรายได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ข้อมูลชนิดบัตรภาษี สำหรับบันทึกข้อมูลชนิดของบัตรภาษี ดังมีรายละเอียด ดังภาพที่ 4.13 เป็นการเรียกข้อมูลขึ้นมาโชว์หน้าจอ เพื่อทำการแก้ไข ลบ หรือยกเลิกแต่ปุ่มเพิ่มจะไม่ยอมให้กดหรือทำอะไรได้

ภาพที่ 4.13 แสดงหน้าจอสำหรับบันทึกข้อมูลชนิดบัตรภาษี

5) ข้อมูลใบนำส่ง จะบันทึกข้อมูลใบนำส่งรายได้ ดังภาพที่ 4.14, 4.15 และ 4.16 โดยจะแยกการบันทึกข้อมูลออกเป็น 2 ส่วน คือส่วนแรกจะบันทึกข้อมูลใบนำส่ง (ส่วนนอกเฟรม) และตามด้วยการบันทึกหมวดรายได้ (ในเฟรม) จนหมดตามใบนำส่งนั้น ๆ

ภาพที่ 4.14 แสดงหน้าจอสำหรับเพิ่มใบนำส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลใบนำส่ง

วันที่: 03/10/99 เวลา: 11:09:45

นำส่งปี พ.ศ.: 2541 เลขที่คลังรับ: 03371

วันที่นำส่ง: 8/9/41 วันที่รับใบนำส่ง: 8/9/42

จำนวนบัตรภาษี: 4 ฉบับ จำนวนเงิน: 111733.78 บาท

รหัสเขตท้องที่: 900117

รหัสหน่วยงาน: 03070

นำส่งหมวดรายได้

รหัสหมวดรายได้: 0001

จำนวนเงิน: 10000 บาท

ภาพที่ 4.15 แสดงหน้าจอสำหรับแก้ไขใบนำส่ง

ข้อมูลใบนำส่ง

วันที่: 03/10/99 เวลา: 11:09:45

นำส่งปี พ.ศ.: 2541 เลขที่คลังรับ: 03371

วันที่นำส่ง: 8/9/41 วันที่รับใบนำส่ง: 8/9/42

จำนวนบัตรภาษี: 4 จำนวนเงิน: 11733.78 บาท

รหัสเขตท้องที่: 900117

รหัสหน่วยงาน: 03070

นำส่งหมวดรายได้

รหัสหมวดรายได้: 0001

จำนวนเงิน: 10000 บาท

Protax

? ต้องการลบข้อมูล

ภาพที่ 4.16 แสดงหน้าจอสำหรับลบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2 การตรวจสอบบัตรภาษี

ก่อนที่จะทำการตรวจสอบบัตรภาษีได้ ต้องบันทึกข้อมูลใบนำส่งและการ Import ข้อมูลบัตรภาษีที่นำมาจากกรมศุลกากรเก็บลงฐานข้อมูลก่อน ซึ่งการตรวจสอบบัตรภาษีจะครอบคลุม ในส่วนของการตรวจสอบข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาชำระแทนเงินสดกับบัตรที่กรมศุลกากรออกให้ผู้ประกอบการ และตรวจสอบจำนวนเงินของบัตรภาษีว่าตรงกับจำนวนเงินในใบนำส่งที่ส่งมาพร้อมกันหรือไม่ดังภาพที่ 4.17, 4.18, และ 4.19

ภาพที่ 4.17 แสดงหน้าจอการตรวจสอบบัตรภาษี

ภาพที่ 4.18 แสดงหน้าจอการตรวจสอบบัตรภาษีแล้วถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ซึ่งไม่เพื่อแจกจ่ายให้ผู้อื่น เมื่อผู้ใดเห็นไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วันที่: 03/10/99 เวลา: 11:26:59

นำส่งปี พ.ศ.: 2541 เลขที่คลังรับ: 03371

จำนวนบัตรภาษี: 4 ฉบับ จำนวนเงิน: 111733.78 บาท

ชื่อผู้สมัครภาษี: []

เลขที่บัตรภาษี: []

ชนิดบัตรภาษี: []

วันที่บัตรหมดอายุ: []

เลขที่ผู้เสียภาษี: []

จำนวนบัตรรวม: 3 ฉบับ จำนวนเงินรวม: 111000 บาท

ปุ่ม: ตกลง ยกเลิก ค้นหาข้อมูล

Dialog Box: Protax ตรวจสอบแล้ว ไม่ถูกต้อง OK

ภาพที่ 4.19 แสดงหน้าจอผลการตรวจสอบบัตรภาษีที่ไม่ถูกต้อง

4.4.3 รายงาน

การจัดทำรายงานของระบบจะแบ่งเป็น 2 ช่วง คือ

- 1) จัดทำรายงานประจำวัน ซึ่งจะมีการจัดทำรายงานทุกวันในส่วนของการรายงานใบนำส่งและรายงานทะเบียนคุมบัตรภาษี ดังภาพที่ 4.20 และตามตารางที่ 4.1 และ 4.2

รายงานใบนำส่ง

นำส่งปี พ.ศ.: 2542

เลขที่คลังรับ ตั้งแต่: [] ถึง: []

ปุ่ม: ตกลง ยกเลิก กลับเมนูหลัก

ภาพที่ 4.20 แสดงหน้าจอสำหรับสั่งทำรายงานรายวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 รายงานใบนำส่ง

หน่วยงานผู้นำส่ง	ว/ด/ป ที่รับ	ว/ด/ป ที่นำส่ง	หมวดรายได้	จำนวนเงิน (บาท)
สรรพากรเขตบึงกุ่ม	10/10/42	13/10/42	ภาษีเงินได้บุคคลธรรมดา	120,500.50
สรรพากรเขตบึงกุ่ม	10/10/42	13/10/42	ภาษีการค้า	31,000.00
สรรพากรเขตบางรัก	20/10/42	21/10/42	ภาษีเงินได้บุคคลธรรมดา	1,120,000.50
สรรพากรเขตบางรัก	20/10/42	21/10/42	ภาษีมูลค่าเพิ่ม	11,000.00
สรรพากรเขตบางรัก	20/10/42	21/10/42	อากรแสตมป์	21,000.00
			รวมบัตรภาษี	18 ฉบับ
			รวมเงินทั้งสิ้น	1,263,501.00 บาท

ตารางที่ 4.2 รายงานทะเบียนคุมบัตรภาษี

ทะเบียนคุมบัตรภาษี

ส่วนราชการ สรรพากรเขตบึงกุ่ม ปี พ.ศ. 2542 เลขที่คลังรับ 00001 ลงวันที่ 10/04/42
ฝ่ายบัญชีทั่วไป สาขาทะเบียนการเงินและบัตรภาษี

ออกให้บริษัท	เลขที่บัตรภาษี	วันที่ออกบัตร	วันที่บัตรหมดอายุ	จำนวนเงิน (บาท)
บ.ฐิติเลิศวัฒนา จำกัด	02C009979	10/06/42	10/06/45	10,000.00
	02K014264	21/08/42	21/08/45	733.78
	02G088738	21/08/42	21/08/45	1,000.00
บ.เทชิน (ประเทศไทย) จำกัด	02G020010	01/09/42	01/09/45	1,000.00
	02C0000901	01/05/42	01/05/45	10,000.00
			รวมบัตรภาษี 5 ฉบับ	
			เป็นเงิน 22,733.38 บาท	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) จัดทำรายงานรายปี จะมีหน้าจอสำหรับใช้สั่งทำรายงาน ดังภาพที่ 4.21 และมีรายงานต่าง ๆ ดังตารางที่ 4.3, 4.4, 4.5, 4.6, 4.7 และ 4.8 พร้อมกันนั้นก็มีการจัดเก็บข้อมูลสรุปรายปีลงฐานข้อมูล เพื่อนำไปใช้ในการจัดทำรายงานเปรียบเทียบการใช้บัตรกับปีอื่น ๆ ต่อไป



ภาพที่ 4.21 แสดงหน้าจอสำหรับสั่งทำรายงานรายปี

ตารางที่ 4.3 รายงานสรุปการนำส่งรายได้แยกตามหมวดรายได้

รายงานสรุปการนำส่งรายได้แยกตามหมวดรายได้
ประจำปี พ.ศ. 2542

หมวดรายได้	จำนวนเงิน (บาท)
ภาษีเงินได้บุคคลธรรมดา	1,000,000.00
ภาษีเงินได้นิติบุคคล	500,000.00
ภาษีการค้า	45,000.00
ภาษีมูลค่าเพิ่ม	400,000.50
ภาษีธุรกิจเฉพาะ	151,000.00
อากรแสตมป์	200,000.00
รวมทั้งสิ้น	2,296,000.50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 รายงานสรุปบัญชีค้างชำระ

รายงานสรุปบัญชีค้างชำระ
ประจำปี พ.ศ. 2542

ชนิดบัตร	จำนวนบัตร (ฉบับ)	จำนวนเงิน (บาท)
หนึ่งแสนบาท	1	100,000.00
หนึ่งหมื่นบาท	5	50,000.00
หนึ่งพันบาท	10	10,000.00
ไม่กำหนดราคา	18	110,005.30
รวมทั้งสิ้น	34	270,005.30

ตารางที่ 4.5 รายงานสรุปการใช้บัตรภาษี

รายงานสรุปการใช้บัตรภาษี

ปี พ.ศ.	ชนิดบัตรภาษี	จำนวนเงิน (บาท)
2541	หนึ่งแสนบาท	2,100,000.00
	หนึ่งหมื่นบาท	4,150,000.00
	หนึ่งพันบาท	11,010,000.00
	ไม่กำหนดราคา	5,001,005.30
	รวมปี 2541	22,261,005.30
2542	หนึ่งแสนบาท	1,100,000.00
	หนึ่งหมื่นบาท	55,000.00
	หนึ่งพันบาท	1,010,000.00
	ไม่กำหนดราคา	131,000.50
รวมปี 2542	2,296,000.50	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 รายงานสรุบบัตรภาษีแยกตามผู้ประกอบการ

รายงานสรุบบัตรภาษีแยกตามผู้ประกอบการ
ประจำปี พ.ศ. 2542

ชื่อบริษัท	จำนวนเงิน (บาท)
บ. อูทีเลศวิวัฒนาจำกัด	1,090,000.00
บ. เทยิน (ประเทศไทย) จำกัด	1,000,000.00
บ. รัตน จำกัด	206,000.50
รวมทั้งสิ้น	2,296,000.50

ตารางที่ 4.7 รายงานสรุบบัตรภาษีแยกตามชนิดบัตร

รายงานสรุบบัตรภาษีแยกตามชนิดบัตร
ประจำปี พ.ศ. 2542

ชนิดบัตร	จำนวนบัตร (ฉบับ)	จำนวนเงิน (บาท)
หนึ่งแสนบาท	11	1,100,000.00
หนึ่งหมื่นบาท	10	100,000.00
หนึ่งพันบาท	100	1,000,000.00
ไม่กำหนดราคา	105	96,000.50
รวมทั้งสิ้น	226	2,296,000.50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 รายงานเปรียบเทียบการใช้บัตรและบัตรค้างชำระ

รายงานเปรียบเทียบการใช้บัตรและบัตรค้างชำระ

ประจำปี พ.ศ. 2542

ชนิดบัตร	จำนวนบัตร (ฉบับ) ที่ชำระ	จำนวนบัตร (ฉบับ) ค้างชำระ
หนึ่งแสนบาท	11	1
หนึ่งหมื่นบาท	10	5
หนึ่งพันบาท	100	10
ไม่กำหนดราคา	105	18
รวมทั้งสิ้น	226	34

4.4.4 การลบข้อมูลบัตรภาษีออกจากรฐานข้อมูล

การลบข้อมูลบัตรภาษีออกจากรฐานข้อมูลมี 2 ลักษณะ คือ ลบบัตรที่นำมาใช้แล้ว และบัตรที่หมดอายุ ซึ่งจะมีการลบปีละครั้ง โดยจะมีการลบหลังจากมีการจัดทำรายงานรายปีแล้ว

บทที่ 5

บทสรุป

5.1 ผลการศึกษา

การศึกษาและพัฒนาระบบสารสนเทศสำหรับตรวจสอบบัตรภาษี สรุปได้ดังนี้

5.1.1 การศึกษารวบรวมข้อมูล

จากการศึกษารวบรวมข้อมูล พบว่าการตรวจสอบบัตรภาษี แบ่งเป็น 2 ส่วน คือ การตรวจสอบบัตรที่ผู้ประกอบการนำมาชำระแทนเงินสดกับบัตรที่กรมศุลกากรออกให้ผู้ประกอบการ ให้ตรงกัน และตรวจสอบระหว่างบัตรภาษีกับใบนำส่งรายได้แผ่นดิน

5.1.2 การออกแบบฐานข้อมูล

หลังจากที่เก็บรวบรวมข้อมูลเรียบร้อยแล้ว จะนำข้อมูลที่ได้มาเขียนแผนภาพการไหลของข้อมูล (Data Flow Diagram) ของระบบปัจจุบัน และระบบใหม่ เพื่อที่จะใช้ในการออกแบบฐานข้อมูล ซึ่งจะใช้วิธีการสร้างความสัมพันธ์ของข้อมูล ให้อยู่ในรูปแบบของ E-R Diagram แล้วทำการแปลงให้อยู่ในรูปแบบของตาราง และทำการนอร์มัลไลซ์ให้อยู่ในแบบ บรรทัดฐานที่ห้า (SNF : Fifth Normal Form) ดังตารางในบทที่ 3

5.1.3 การออกแบบหน้าจอ และรายงาน

จากตารางข้อมูลที่ได้จากการออกแบบ นำไปสร้างต้นแบบ (Prototype) และรายงาน ได้ดังนี้คือ

จอภาพการทำงานหลัก	2	จอภาพ
จอภาพสนับสนุนการทำงาน	19	จอภาพ
รายงาน	8	รายงาน

5.2 ความสามารถของระบบที่พัฒนาขึ้น

ในการออกแบบระบบใหม่ พยายามให้มีการตรวจสอบบัตรภาษีให้รวดเร็วยิ่งขึ้นเมื่อเปรียบเทียบกับระบบปัจจุบันได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) สามารถตรวจสอบบัตรภาษีได้รวดเร็วกว่าระบบปัจจุบัน เนื่องจากระบบปัจจุบันมีการประมวลผลแบบ Batch Processing บนเครื่องเมนเฟรม และข้อมูลที่ได้มาจากคนละเครื่อง เช่น ข้อมูลใบนำส่งรายได้ได้มาจากส่วนประมวลบัญชี และข้อมูลบัตรภาษีที่ผู้ประกอบการนำมาชำระก็มาจากฝ่ายบันทึกและตรวจสอบข้อมูล ซึ่งก็มาจากอีกเครื่องหนึ่งทำให้การประมวลผลสำหรับตรวจสอบบัตรภาษีโดยฝ่ายเครื่องคอมพิวเตอร์ได้ช้า แต่ระบบใหม่ที่ได้พัฒนาขึ้นจะทำงานอยู่ที่เครื่องเดียวกัน และมีหน่วยงานที่ทำหน้าที่โดยตรงรับผิดชอบ คือส่วนประมวลบัญชี ทำให้การตรวจสอบข้อมูลบัตรภาษีได้ทันที

2) สามารถค้นหาข้อมูลบัตรภาษีได้รวดเร็ว เมื่อมีการสอบถามจากผู้บริหารหรือจากกรมศุลกากร เพียงแต่บันทึกเลขที่บัตรภาษีที่ต้องการไป ก็จะทราบได้ทันทีว่าบัตรภาษีที่ต้องการค้นหาถูกใช้ไปหรือยัง แต่ถ้าเป็นระบบปัจจุบันจะต้องส่งข้อมูลบัตรภาษีไปให้ ฝ่ายเครื่องคอมพิวเตอร์ ทำการประมวลผลเพื่อค้นหาข้อมูลให้ ทำให้ได้ข้อมูลช้าไม่ได้ในทันที

3) สามารถจัดทำรายงานสำหรับผู้บริหารได้ โดยจะออกเป็นรายปี แต่ระบบปัจจุบันไม่มีการทำในส่วนนี้

5.3 ปัญหา และข้อเสนอนะ

ในการพัฒนาระบบ จะมีจุดด้อยอยู่บ้าง เนื่องจากมีเวลาน้อยมากสำหรับการศึกษาและพัฒนาโปรแกรม ซึ่งทำให้การใช้งานยังเป็นลักษณะ Stand alone บนเครื่อง PC และข้อเสนอนะในการพัฒนาต่อไป เมื่อข้อมูลมีปริมาณมากขึ้น และผู้ใช้เพิ่มขึ้น ควรจะพัฒนาบนเครื่องที่มีลักษณะเชื่อมต่อกัน และ Server คอยให้บริการและจัดการฐานข้อมูล ซึ่งจะทำให้มีการ Share การใช้งานได้พร้อม ๆ กัน เช่น SQL Server, Oracle เป็นต้น จะทำให้การปฏิบัติงานเป็นไปด้วยความรวดเร็วยิ่งขึ้น

ภาคผนวก ก

แนวทางการพัฒนาแอปพลิเคชันบนวินโดวส์โดยใช้ Visual Basic

ก.1 ความรู้เบื้องต้นเกี่ยวกับ Visual Basic

เนื่องจากปัจจุบันความต้องการในการนำเอาระบบคอมพิวเตอร์มาใช้ในการทำงานต่างๆมีมากขึ้น ซึ่งงานในแต่ละระบบย่อมประกอบไปด้วย ผู้ใช้ (User) ที่แตกต่างกันออกไปทั้งในแง่ของประสิทธิภาพในการใช้คอมพิวเตอร์และความชำนาญ ด้วยเหตุนี้เองทำให้ผู้พัฒนาระบบพยายามที่จะออกแบบระบบให้สามารถติดต่อสื่อสารกับผู้ใช้ได้ที่แตกต่างกันนั้นได้ โดยมองที่ความสามารถทางด้าน User Interface ที่ดี เข้าใจง่าย ใช้งานสะดวกเป็นสำคัญ User Interface แบ่งเป็น 2 ประเภทใหญ่ๆ คือ

- 1) Command User Interface
- 2) Graphics User Interface

ซึ่งมีความเหมาะสมในการนำไปใช้งานที่แตกต่างกัน กล่าวคือ Command User Interface เหมาะกับระบบงานที่ผู้ใช้งานมีความชำนาญในการใช้ Keyboard และ คอมพิวเตอร์เป็นอย่างดี

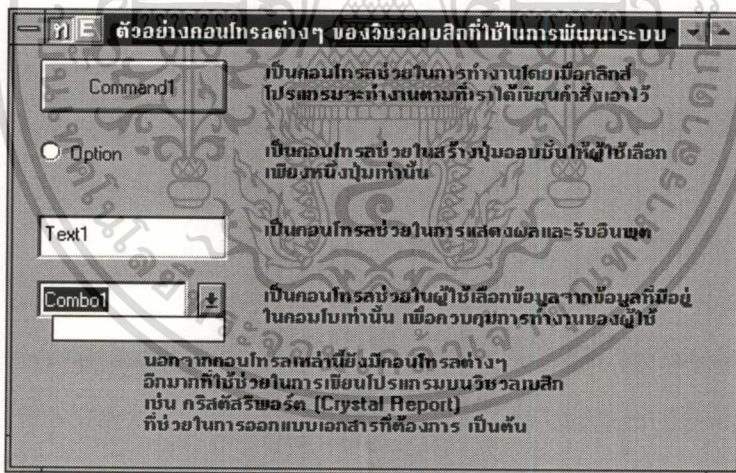
ส่วน Graphics User Interface นั้นเหมาะกับระบบงานที่ ผู้ใช้ไม่มีความชำนาญในการใช้คอมพิวเตอร์นัก หรือระบบงานที่ต้องการความสะดวกรวมทั้งความสวยงามของหน้าจอต่างๆ

ด้วยเหตุนี้เองจึงทำให้ผู้พัฒนาระบบส่วนใหญ่ในปัจจุบันหันมาใช้จัดทำ User Interface แบบ Graphics กันมากขึ้น จึงเกิดการพัฒนาซอฟต์แวร์ที่ช่วยในการเขียนโปรแกรมด้านนี้มากขึ้น

จากการพัฒนาระบบปฏิบัติการจากดอส (DOS) ไปสู่วิโดวส์ (Windows) การพัฒนาแอปพลิเคชันต่างๆซึ่งเดิมที่เคยใช้เครื่องมือในการพัฒนาเพียงแค่ แอสเซมเบลอร์, ซี-คอมไพเลอร์, ปาสคาล-คอมไพเลอร์ ฯลฯ กับโปรแกรมเอดิเตอร์ ก็เพียงพอแล้วสำหรับแอปพลิเคชันบนดอสซึ่งมีสถานะแวดล้อมที่ไม่ซับซ้อนอะไรมากนัก แต่ในวินโดวส์ซึ่งมีสถานะแวดล้อมแบบกราฟิก, เป็นระบบมัลติทาสกิ้ง(Multitasking) ฯลฯ เครื่องมือ (Tools) ในการพัฒนาที่เคยใช้กันอยู่คงจะไม่เหมาะสมแล้วหากนำมาใช้ในการพัฒนาแอปพลิเคชันบนวินโดวส์ (Windows Application) เพราะจะทำให้ใช้ทรัพยากรทั้งกำลังคนและเวลาเป็นอย่างมาก จากจุดนี้เองจึงเป็นจุดเริ่มต้นของแนวความคิดในการสร้างเครื่องมือต่างๆ เพื่อนำมาช่วยสำหรับการพัฒนาแอปพลิเคชันบนวินโดวส์ให้สะดวกและง่ายยิ่งขึ้น ระบบช่วยเขียนโปรแกรม Microsoft Visual for Windows ถือได้ว่าเป็นความก้าว

หน้าที่สำคัญที่เกี่ยวข้องกับการเขียนแอปพลิเคชันที่ทำงานบน Windows ใน Visual Basic ประกอบไปด้วยเอนจิน (Engine) ที่ทำงานด้วย อีเวนต์และเครื่องมือช่วยออกแบบที่ใช้งานได้ง่าย ซึ่งด้วยลักษณะเหล่านี้ทำให้สามารถได้รับประโยชน์จากสภาพแวดล้อมกราฟิกของ Windows ได้อย่างเต็มที่ เพื่อให้สามารถสร้างแอปพลิเคชันที่มีสมรรถนะสูงได้อย่างรวดเร็ว โดยเฉพาะอย่างยิ่งสำหรับงานทางด้าน Graphic User Interface

Visual Basic ยังเป็นหนึ่งในภาษาคอมพิวเตอร์ภาษาแรกที่สนับสนุน Event-Driven programming ซึ่งเป็นรูปแบบการเขียนโปรแกรมที่เหมาะสมกับ Graphic User Interface เป็นอย่างยิ่ง ใน Event-driven programming แทนที่จะเขียนโปรแกรมที่แสดงขั้นตอนทุกๆ ขั้นตอนออกมาเรียงตามลำดับ ผู้เขียนโปรแกรมกลับเขียนโปรแกรมที่ตอบสนองต่อกิจกรรมที่ผู้ใช้ทำเช่น การเลือกคำสั่ง การคลิกหรือกดปุ่มเมาส์ในวินโดว์ การเคลื่อนย้ายเมาส์ แทนที่จะเขียนโปรแกรมขนาดใหญ่เพียงโปรแกรมเดียว ผู้เขียนโปรแกรมจะสามารถสร้างโปรแกรมเป็นกลุ่มของโปรแกรมย่อยๆ ซึ่งจะเกิดการทำงานโดยอีเวนต์ที่ผู้ใช้เริ่มขึ้น และด้วย Visual Basic ทำให้สามารถเขียนแอปพลิเคชันเช่นนั้นได้ด้วยความรวดเร็วและง่ายขึ้นดังแสดงตามภาพ ก.



ภาพ ก. แสดงคอนโทรลพื้นฐานของวิซวลเบสิก

นอกจากนี้ยังมีออบเจกต์ต่างๆ ที่ช่วยในการพัฒนาระบบด้านฐานข้อมูล เช่น

4.2 ดาต้าเบสออบเจกต์ (Database Object)

เป็นฐานข้อมูลที่รวบรวมข้อมูลที่มีความสัมพันธ์กันในด้านใดด้านหนึ่ง ในฐานข้อมูล (Database) จะสามารถประกอบด้วยตาราง (Table), Queries และ Indexes ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในออบเจ็กต์ฐานข้อมูล (Database Object.) นี้จะมีคุณสมบัติ (Properties) และกลวิธี (Method) ต่างๆ สำหรับใช้ในการจัดการกับข้อมูลให้ใช้

เราสามารถสร้างตัวแปรออบเจ็กต์ (Object variables) ชนิดฐานข้อมูล (Database) โดยการ ใช้คำสั่ง Dim และ Set ดังตัวอย่างต่อไปนี้ที่สร้างและเปิดออบเจ็กต์ฐานข้อมูล

```
Dim MyDb As Database
```

```
Set MyDb = OpenDatabase ("C:\VB\BIBLIO.MDB")
```

ซึ่งมีคุณสมบัติและการใช้งานที่สำคัญดังนี้

- คุณสมบัติ Connect (Connect Property) ใช้สำหรับกำหนดข่าวสาร (Information) ที่ใช้สำหรับการเปิดฐานข้อมูลภายนอก (External Database) หรือใช้สำหรับการเชื่อมต่อกับฐานข้อมูลจากภายนอกนั่นเอง

```
ไวยากรณ์ database.Connect { = connectinfo }
```

สำหรับฐานข้อมูลภายนอกชนิด ODBC (ODBC external database) เช่น SQL Server หรือ Oracle (ซึ่งต้องลงทะเบียน (Register) ในแฟ้ม ODBC.INI ไว้แล้ว) ค่าของคุณสมบัติ Connect สำหรับฐานข้อมูลจะเป็นดังต่อไปนี้

```
ODBC ; DSN = ourdbsvr, UID = guest ; PWD = guest ;
```

DSN คือชื่อของแหล่งข้อมูล(data source name), UID คือรหัสผู้ใช้(user ID), และ PWD คือรหัสผ่าน(password)

ถ้าไม่ได้ให้ค่าสำหรับชื่อฐานข้อมูล(Database Name)และ คุณสมบัติConnect จะมีรายชื่อของแหล่งข้อมูล ODBC ที่ลงทะเบียนไว้ปรากฏขึ้นมาให้เลือก

ถ้าต้องใช้รหัสผ่านแต่ได้ให้ค่าไว้ในคุณสมบัติConnect จะปรากฏ Login Dialog Box ขึ้นมาทุกครั้งที่เข้าถึงตาราง

- OpenDatabase Function ใช้สำหรับเปิดฐานข้อมูลที่มีอยู่แล้วและส่งกลับเป็นค่าออบเจ็กต์ฐานข้อมูล

```
ไวยากรณ์ OpenDatabase(dbname [,exclusive [,readonly [,connect ]]])
```

dbname คือ ชื่อฐานข้อมูลที่มีอยู่แล้ว

exclusive เป็นชนิด Boolean ถ้าเป็น True ฐานข้อมูลจะเปิดให้ผู้ใช้คนเดียว, ถ้าเป็น False ฐานข้อมูล จะเปิดให้ผู้ใช้ใช้ได้หลายคนพร้อมๆกัน ค่าปกติกำหนดเป็น False

readonly เป็นชนิด Boolean ถ้าเป็น True ฐานข้อมูลจะเปิดให้อ่าน

ได้อย่างเดียว, ถ้าเป็น False ฐานข้อมูลจะเปิดให้อ่านและเขียนได้ ค่าปกติกำหนดเป็น False

ชนด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

connect เป็นข้อความที่ใช้สำหรับการเปิดฐานข้อมูล

ถ้า dbname ที่อ้างถึงในฐานข้อมูลเปิดอยู่แล้วสำหรับการเข้าถึงชนิด exclusive สำหรับผู้อื่น, หรือ dbname นี้ไม่มี จะเกิดข้อผิดพลาดขึ้น และถ้าจะเปิดฐานข้อมูลให้เป็นชนิด exclusive แต่มีผู้อื่นเปิดใช้อยู่แล้วจะเกิดข้อผิดพลาดขึ้น

ถ้า dbname เป็น empty string("") และ connect กับ ODBC แล้วจะมี Dialog Box รายชื่อของแหล่งข้อมูล ODBC ที่ลงทะเบียนไว้ปรากฏขึ้นมาให้เลือก

- Close Method ใช้ปิดฐานข้อมูล

ไวยากรณ์ `database.Close`

database คือตัวแปรออบเจกต์ที่ได้จากการเปิดฐานข้อมูล

ถ้าออบเจกต์ปิดไปแล้วการ Close จะเกิดข้อผิดพลาด ควรจะปิด recordsets ก่อนที่จะปิดฐานข้อมูล มิเช่นนั้นอาจจะสูญเสียข้อมูลที่เปลี่ยนแปลงจากการเขียนหรือเพิ่มเติม

ก.3 Dynaset Object

เป็น recordset ที่ได้จากการ query และสามารถแก้ไขได้ Dynaset นี้จะเหมือนกับ Table(ตาราง) คือมี records และ field Dynaset นี้จะเป็นกลุ่มของ record ที่สามารถเพิ่ม,เปลี่ยนแปลง และลบ record จากตาราง ได้ Dynaset นี้สามารถจะเก็บ field จากตารางต่างๆได้มากกว่าหนึ่งตาราง

เราสามารถกำหนดออบเจกต์ตัวแปรเป็นชนิด Dynaset ได้โดยใช้กลวิธี CreateDynaset ในฐานข้อมูล(Database) หรือ Dynaset อื่น สามารถสร้างออบเจกต์ตัวแปรได้มากกว่าหนึ่งตัวที่อ้างอิงถึง record เดียวกัน

- คุณสมบัติ Transaction (Transactions Property) ใช้กับ Dynaset เพื่อสนับสนุนให้ record ที่มีการเปลี่ยนแปลงสามารถที่จะ roll back(ยกเลิก) หรือ committed (บันทึก)

ไวยากรณ์ `recordset.Transactions`

กำหนดค่าเป็น True ถ้าให้สนับสนุน transactions หรือเป็น False ถ้าไม่สนับสนุน transactions

- CreateDynaset Method ใช้สร้าง ออบเจกต์ชนิด Dynaset จากตารางที่กำหนด,Queryที่กำหนด,หรือคำสั่ง SQL

ไวยากรณ์ `Set dynasetobject = database.CreateDynaset(source [,options])`

`Set dynasetobject = {recordset | querydef}.CreateDynaset ([Options])`

dynasetobject คือออบเจกต์ตัวแปรชนิด Dynaset

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

database	คือออบเจกต์ตัวแปรที่จากการเปิดฐานข้อมูล
recordset	คือออบเจกต์ตัวแปรของออบเจกต์ที่เป็น recordset
querydef	เป็นออบเจกต์ตัวแปรของออบเจกต์ QueryDef
source	เป็นชื่อของ recordset หรือ QueryDef ของคำสั่ง SQL ที่มีอยู่
options	เป็นตัวเลขที่แทนตัวเลือกต่างๆค่าของตัวเลขสามารถนำมารวมกัน โดยการเพิ่มเพื่อเป็นการใช้ควบคู่กันได้

ก.4 Snapshot Object

เป็น recordset ที่ได้จากการ query ซึ่งไม่สามารถแก้ไขได้ Snapshot นี้จะเหมือนกับ Table (ตาราง) คือมี records และ field Snapshot นี้จะเป็นภาพของกลุ่ม record ที่ใช้สำหรับหาข้อมูลหรือทำรายงาน Snapshot นี้สามารถจะเก็บ field จากตารางต่างๆได้มากกว่าหนึ่งตาราง

เราสามารถกำหนดออบเจกต์ตัวแปรเป็นชนิด Snapshot ได้โดยใช้กลวิธี CreateSnapshot ในฐานข้อมูล(Database) หรือ Dynaset หรือ Snapshot อื่น สามารถสร้างออบเจกต์ตัวแปรได้มากกว่าหนึ่งตัวที่อ้างอิงถึง record เดียวกัน

จะเหมือนกับ recordset ที่เมื่อสร้าง Snapshot ขึ้น record แรกจะเป็น record ปัจจุบัน และสามารถใช้ Move method เพื่อเลื่อนตำแหน่งของ record ปัจจุบัน หรือเลื่อนไปยังตำแหน่งที่ต้องการค้นหาได้

- CreateSnapshot Method ใช้สร้าง ออบเจกต์ชนิดSnapshot จากตารางที่กำหนด,Queryที่กำหนด,หรือคำสั่ง SQL

ไวยากรณ์ Set Snapshot = database.CreateSnapshot(source [,options])
Set Snapshot = {recordset | querydef}.CreateSnapshot ([Options])

Snapshot	คือออบเจกต์ตัวแปรชนิดSnapshot
database	คือออบเจกต์ตัวแปรที่จากการเปิดฐานข้อมูล
recordset	คือออบเจกต์ตัวแปรของออบเจกต์ที่เป็น recordset
querydef	เป็นออบเจกต์ตัวแปรของออบเจกต์ QueryDef
source	เป็นชื่อของ recordset หรือ QueryDef ของคำสั่ง SQL ที่มีอยู่
options	เป็นตัวเลขที่แทนตัวเลือกต่างๆค่าของตัวเลขสามารถนำมารวมกัน โดยการเพิ่มเพื่อเป็นการใช้ควบคู่กันได้

ไม่เหมือนกับข้อมูลใน Dynaset ข้อมูลใน Snapshot จะไม่เชื่อมโยงกับข้อมูลใน table(ตาราง) ข้อมูลใน Snapshot จะไม่เปลี่ยนแม้ว่าข้อมูลในตารางจะมีการเปลี่ยนแปลง ยิ่งไปกว่านั้นข้อมูลใน Snapshot จะไม่สามารถเปลี่ยนแปลงได้

ด้วยความสามารถต่างๆของ Visual Basic ที่กล่าวมานี้เองจึงทำให้ถูกนำมาใช้ในการพัฒนาแอปพลิเคชันต่างๆอย่างมากมายโดยเฉพาะการนำมาช่วยในด้าน User Interface



ภาคผนวก ข

ปัญหาปี ค.ศ. 2000

ข.1 สาเหตุ

ปัญหาปี ค.ศ. 2000 คือ ปัญหาที่ระบบคอมพิวเตอร์ไม่สามารถคำนวณปีที่แทนด้วยตัวเลขสองหลัก หลังปี ค.ศ.1999 ได้อย่างถูกต้อง ซึ่งสาเหตุเกิดจากการจัดเก็บตัวเลข 2 หลัก แทน 4 หลัก ดังนี้

1) Hardware โดยเฉพาะในส่วนของหน่วยความจำ สมัยก่อนมีราคาแพง เพื่อเป็นการประหยัดเนื้อที่ในการจัดเก็บข้อมูลในระบบคอมพิวเตอร์ จึงมีการจัดเก็บข้อมูลเกี่ยวกับวันที่ (Date) เพียง 2 หลักท้ายของปี เช่น “97” หมายถึง “1997” (หมายถึง ค.ศ.1997) และภาษาโคบอลก็มีฟอร์แมตของวันที่เป็นแบบ YYMMDD

2) ในอดีตประสิทธิภาพการประมวลผลของคอมพิวเตอร์ช้ากว่าปัจจุบันมาก จึงต้องใช้ตัวเลขอ้างอิงให้น้อยที่สุด เพื่อให้การประมวลผลของระบบคอมพิวเตอร์เร็วขึ้น การอ้างอิงตัวเลขเพียง 2 หลัก จึงทำได้เร็วกว่า 4 หลัก

3) เพื่อให้การบันทึกข้อมูล (Data Entry) รวดเร็วขึ้น ซึ่งก็มีเหตุผลทำนองเดียวกับข้อ 2) เพราะทุกครั้งจะป้อนข้อมูลเลขปีแค่เพียง 2 หลัก เท่านั้น

ข.2 ปัญหาที่เกี่ยวข้องกับปี ค.ศ. 2000

1) ปัญหาเกี่ยวกับการคำนวณวันในสัปดาห์

โปรแกรมส่วนมากใช้วิธีการคำนวณวัน ในแต่ละสัปดาห์ด้วยการใช้ปี ค.ศ. แบบสองหลัก ซึ่งทำให้ คอมพิวเตอร์ไม่ทราบข้อเท็จจริง ที่ว่า วันที่ 1 มกราคม ค.ศ. 2000 เป็นวันเสาร์ แต่กลับคำนวณว่าเป็นวันที่ 1 มกราคม ค.ศ. 1900 ซึ่งเป็นวันจันทร์ และแจ้งผลผิดพลาดไม่เพียงแต่ในวันที่ 1 มกราคม ค.ศ. 2000 เท่านั้น ในวันอื่นๆ หลังจากวันนี้ ก็จะเกิดผิดพลาดไปหมด การคำนวณด้วยเลขสองหลักแบบนี้ จะถูกต้องก็ต่อเมื่อนั้นอยู่ในกรอบ 1 มกราคม 1900 ถึง 31 ธันวาคม 1999

2) ปัญหาที่เกิดปี ค.ศ. 1999

จากรูปแบบวันที่ที่ใช้ YYMMDD จึงมีผู้ทำโปรแกรมกำหนดวันหมดอายุของการใช้งาน เช่น ซอฟต์แวร์ตัวหนึ่ง มีการกำหนดวันหมดอายุให้นานเท่าที่จะนานได้ จึงมีการกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น 99/9/9 หรือวันที่ 9 กันยายน ค.ศ. 1999 ซอฟต์แวร์ที่หมดอายุนี้จะเกิดขึ้น โดยที่ผู้ใช้อาจจะไม่รู้
ว่าเพราะอะไร ทั้งนี้ผู้เขียนโปรแกรมอาจเขียนไว้เมื่อสิบกว่าปีที่แล้ว และคิดว่าการใช้ระยะเวลาเป็น
99/9/9 นี้ น่าจะนานพอแล้ว จึงไม่ได้บอกใคร

ข.3 ปัญหาปี ค.ศ. 2000 กับระบบคอมพิวเตอร์

ปัญหาที่เกิดขึ้นอาจแบ่งได้เป็น 2 ส่วน คือ
ส่วนแรกเป็นปัญหาจากระบบคอมพิวเตอร์ และส่วนที่สองก็จะเป็นปัญหาของความยากที่จะแก้ไข

1) ปัญหาจากระบบคอมพิวเตอร์

ระบบคอมพิวเตอร์ในที่นี้ หมายถึง อุปกรณ์เครื่อง (Hardware) โปรแกรมระบบ (System Software) โปรแกรมประยุกต์ (Application Software) ระบบเครือข่ายและการสื่อสารระยะไกล (Network & Telecommunication) และข้อมูล (Data)

- อุปกรณ์เครื่อง (Hardware)

อุปกรณ์เครื่องจะหมายถึงคอมพิวเตอร์ขนาดต่างๆ ได้แก่ ซุปเปอร์คอมพิวเตอร์ (Supercomputer) คอมพิวเตอร์เมนเฟรม (Computer Mainframe) มินิคอมพิวเตอร์ (Minicomputer) เวิร์กสเตชัน (Workstation) และเครื่องคอมพิวเตอร์ส่วนบุคคล (Personal Computer : PC) หรือไมโครคอมพิวเตอร์ (Microcomputer) รวมทั้งระบบไคลเอนต์เซิร์ฟเวอร์ (Client/Server) ปัญหาที่จะพบในระบบของอุปกรณ์เครื่อง ก็คือ ระบบฐานเวลาหรือซิสเต็มคล็อก (System Clock) ใช้ตัวเลข 2 ตัว กำหนดปีในซิสเต็มคล็อก

ในไมโครคอมพิวเตอร์ ข้อมูลวันที่ (Date) จะเก็บไว้ในเรียลไทม์คล็อก (CMOS RTC : Real Time Clock) ซึ่งเป็นชิป (Chip) ตัวหนึ่งอยู่บนเมนบอร์ด (Mainboard) มีหน้าที่ดูแลและจัดการในส่วนของวันที่ (Date) และเวลา (Time) ของระบบ จะทำงานตลอดเวลาไม่หยุดแม้ในเวลาปิดเครื่องก็ตาม ซึ่งชิป RTC ในเครื่องไมโครคอมพิวเตอร์ จะเก็บตัวเลขปีเพียง 2 หลัก โดยเก็บไว้ใน CMOS ที่ Address 9 และตัวเลขปีที่เป็นคริสตศักราช ซึ่งหมายถึง 2 หลักแรก ก็จะเก็บไว้ใน CMOS เช่นกัน ที่ Address 5d ค่าตัวเลข ที่เป็น

คริสตศักราชจะไม่ถูกเพิ่ม โดยอัตโนมัติ แม้ตัวเลขปีจะเปลี่ยนจาก 99 ไปเป็น 00 ผลลัพธ์ของการทำงานที่เกิดขึ้น คือ ข้อมูลวันที่ 2000-01-01 (YYYY-MM-DD) เวลา 00:00:00 จะกลายเป็น 1900-01-01 เวลา 00:00:00 ในชิป RTC โปรแกรมไบออส (BIOS : Basic Input/Output System) จะกำหนดและอ่านค่าของคริสตศักราชจาก RTC แต่จะไม่เก็บไว้

- โปรแกรมระบบ (System Software)

โปรแกรมระบบอย่างหนึ่งก็คือ OS (Operating System) บนไมโครคอมพิวเตอร์ เช่น DOS, Windows โดยที่ข้อมูลวันที่ถูกเก็บไว้ 2 แห่ง คือ ใน CMOS RTC ของฮาร์ดแวร์แห่งหนึ่งและอีกแห่งหนึ่งเก็บไว้ใน OS คือ ทั้ง OS DOS และ OS Windows เมื่อไมโครคอมพิวเตอร์ถูก Reboot หลังจากวันที่ 2000-01-01 เวลา 00:00:00 เนื่องจากวันที่ ที่ชิป RTC ส่งให้กับ DOS เปลี่ยนไปใช้ค่า Default (ค่าที่เครื่องคอมพิวเตอร์จะกำหนดให้โดยอัตโนมัติ ถ้าผู้ใช้งานไม่ได้กำหนดให้) ตามวันที่ในตัวมันแทน คือ 1980-01-01 หรือ 1980-04-01 ในบางกรณี พบว่าซอฟต์แวร์ที่ใช้กันอยู่ในปัจจุบัน ส่วนใหญ่จะอ่านวันที่จาก OS รับค่ามาไม่ถูกต้อง ยังผลให้มีการประมวลผลผิดพลาด หรือไม่ประมวลผล เพราะเกิดความสับสนใน LOGIC

- โปรแกรมประยุกต์ (Application Software)

เป็นโปรแกรมที่เขียนขึ้น เพื่อใช้งานเฉพาะงานใดงานหนึ่ง ด้วยภาษาคอมพิวเตอร์ ซึ่งมีอยู่ด้วยกันหลายภาษา และพบว่าโปรแกรมประยุกต์ที่เขียนด้วยภาษาบนเมนเฟรม จะมีปัญหาปี ค.ศ. 2000 ภาษาที่เขียนใช้งานบนเมนเฟรมมากที่สุด คือ ภาษาโคบอล (COBOL) มีการเขียนและพัฒนาใช้ต่อเนื่องกันมานาน และพบว่าโปรแกรมที่เขียนด้วยภาษาโคบอลในอดีตก่อนปี ค.ศ. 1990 นับตั้งแต่ ค.ศ. 1960 เป็นต้นมา ส่วนใหญ่จะมีปัญหาทั้งหมด งานธุรกิจที่ใช้โปรแกรมที่เขียนด้วยภาษาโคบอลบนเมนเฟรม ส่วนใหญ่ยังใช้โปรแกรมโคบอลอยู่ บางบริษัทมีการพัฒนาต่อเนื่องกันมาจนยากที่จะหา Source Code ได้ (Source Code คือ โปรแกรมที่ผู้เขียนโปรแกรมเขียนขึ้น ก่อนส่งให้ Compiler แปลเป็นภาษาเครื่อง) เพราะมีการพัฒนารับช่วงต่อกันมาหลายระดับและหลายคน จึงมีเพียงข้อมูลขั้นต้น ที่แสดงว่าผู้เขียนภาษาโคบอลในยุคที่ใช้เมนเฟรมช่วงแรก มีกว่า 900,000 ราย เขียนผลงานเป็นโปรแกรมมากกว่า 180 พันล้านบรรทัด ซึ่งโปรแกรมเหล่านี้ยังมีใช้งานกระจายในงานประยุกต์ต่างๆ มากมาย และเมื่อรวมกับภาษาอื่นๆ แล้วประมาณกันว่า ปัจจุบันมีซอฟต์แวร์ประมาณ 500 พันล้านบรรทัดใช้งานอยู่ทั่วโลก โดยประมาณ 85% จะถูกกระทบจากการเปลี่ยนศตวรรษ ซึ่งเป็นจำนวนมหาศาล จะต้องถูกตรวจสอบคำสั่งที่เกี่ยวข้องกับ วัน เวลา ในทุกบรรทัดของแต่ละซอฟต์แวร์

- โปรแกรมสำเร็จรูป (Package)

เป็นซอฟต์แวร์ที่ผู้ผลิตผลิตเพื่อขายให้แก่ลูกค้านำไปใช้งานตามลักษณะงาน เช่น ระบบงานบัญชี ระบบงานบุคลากร ระบบงานสินค้าคงคลัง ระบบงานเงินเดือน ซึ่งส่วนหนึ่งของโปรแกรมพัฒนามาจากโปรแกรมที่เรียกว่า Spreadsheet เช่น Lotus, Excel โปรแกรมจัดการฐานข้อมูล เช่น dBase, FoxPro, Oracle, Sybase เป็นต้น หากมีระบบจัดเก็บตัวเลขปีเพียง 2 หลัก ก็จะมีปัญหาปี ค.ศ. 2000 เช่นเดียวกัน แต่ปัญหาจะน้อยกว่าโปรแกรมที่เขียนด้วยภาษาที่กล่าวมาแล้ว

เช่น ภาษาโคบอล เนื่องจากเป็นโปรแกรมที่มีการพัฒนาใช้ในระยะเวลายาวนานมาแล้ว และมีผลิตภัณฑ์ที่ผู้ผลิตยืนยันว่า ไม่กระทบกับปี ค.ศ. 2000 เช่น dBase IV , Oracle , Lotus เป็นต้น

- ข้อมูล (Data)

ฐานข้อมูลที่มีการจัดเก็บข้อมูลเกี่ยวกับวันที่ด้วยตัวเลขเพียงสองหลักท้าย ปัญหาปี ค.ศ. 2000 จะมีผลกระทบอย่างแน่นอน เช่น การเรียงลำดับข้อมูล (Sorting) การค้นหาข้อมูลและแทนที่ (Searching and Replacing) การปรับปรุงแก้ไขข้อมูล (Updating/Modifying) เป็นสิ่งที่องค์กรจะต้องตระหนักและหาทางแก้ไข โดยที่ข้อมูลมีเป็นจำนวนมาก ดังนั้นองค์กรจะต้องมีกลยุทธ์ในการแก้ปัญหาที่จะเกิดขึ้น โดยทำอย่างไรให้ระบบคอมพิวเตอร์ค้นหาข้อมูลที่ต้องการพบแล้วประมวลผลได้ถูกต้อง

- ระบบเครือข่าย (Network System)

ภาพรวมหลักๆ ของระบบเครือข่ายประกอบด้วย ฮาร์ดแวร์ ซอฟต์แวร์ระบบ (System Software) ซอฟต์แวร์ประยุกต์ (Application) ข้อมูล (Data) และการสื่อสารข้อมูล (Data-communication) ผ่านระบบการสื่อสารต่างๆ เช่น สายสื่อสาร (โทรศัพท์ , Fibre Optic) ดาวเทียม (Satellite) ไมโครเวฟ (Microwave) เมื่อพิจารณาแต่ละส่วนประกอบแล้ว ได้รับผลกระทบจากปัญหาปี ค.ศ. 2000 แน่แน่นอน ระบบโปรแกรมที่ฝังในฮาร์ดแวร์ (Embedded System) ระบบปฏิบัติการของเครื่องแต่ละเครื่องแต่ละขนาด นับตั้งแต่ Supercomputer , Mainframe , Minicomputer , Server , Workstation , Terminal ต้องได้รับการตรวจสอบแล้ว ยังมีซอฟต์แวร์ประยุกต์ในแต่ละงาน (Application) เช่น โปรแกรมจัดการฐานข้อมูล หากมีรูปแบบการจัดเก็บตัวเลขปีเพียง 2 หลักท้าย ก็จะต้องกระทบด้วยเช่นกัน และยังมีระบบการสื่อสารอีกที่จะต้องพิจารณา ซึ่งอาจจะทำให้งานต่างๆ ที่อยู่บนเครือข่ายทำงานผิดพลาด หรือหยุดทำงาน เช่น Electronic-Mail , EDI , Office Automation , Multimedia on LANs , CAD/CAM , Computer-Aided Instruction , Video-based conferencing system

2) ปัญหาของความยากที่จะแก้ไข

จากการวิจัยของ DATA SOFTWARE PRODUCTIVITY RESEACH ในอเมริกาพบว่า ภาษาโปรแกรมที่จะโดนผลกระทบของปี 2000 มากที่สุด คือ ภาษาโคบอล เพราะปริมาณของโปรแกรมมีเป็นจำนวนมาก และภาษาอื่นๆ ก็มีผลกระทบลดหลั่นกันตามจำนวนแอปพลิเคชัน

เมื่อปริมาณของโปรแกรม (Code) มีเป็นจำนวนมาก ปัญหาที่พบก็คือ การตรวจสอบโปรแกรมแต่ละส่วน แต่ละฟังก์ชันว่า มีฟังก์ชันวันที่ (Date Function) ตรงไหนบ้าง ต้องดูทีละ

บรรทัด เพื่อจะได้แก้ไขให้ถูกต้อง ซึ่งพิจารณาแล้ว ไม่มีอะไรสำหรับการแก้ไข แต่มีข้อจำกัดที่เป็นอุปสรรคในการดำเนินการ ก็คือ

- เวลาเมื่อวิเคราะห์แล้ว จะเห็นว่าใช้เวลาสำหรับการแก้ไขน้อยมาก เมื่อเทียบกับปริมาณโปรแกรมที่มีอยู่ เมื่อนับจากปัจจุบันจนถึง 31 ธันวาคม 1999 ดังนั้นเวลาจึงเป็นปัจจัยสำคัญสำหรับระบบขององค์กรทุกองค์กร โดยที่จะต้องแก้ไขให้เสร็จภายในวันที่ 31 ธันวาคม 1998 เพื่อจะได้มีเวลาสำหรับการทดสอบระบบ (Testing) อย่างน้อยๆ ควรให้เวลาในการทดสอบไม่น้อยกว่า 1 ปี

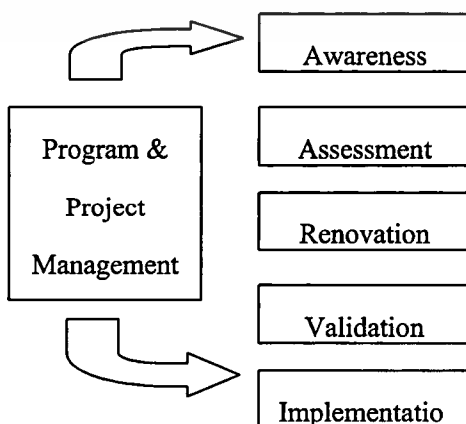
- ระบบบางระบบ ได้มีการพัฒนาใช้ต่อเนื่องมานาน เอกสารต่างๆ และ Source Code เป็นสิ่งสำคัญมาก หากไม่มีสิ่งเหล่านี้ เป็นการยากยิ่งในการแก้ไข ยิ่งเป็นระบบใหญ่เท่าใด ยิ่งมีความซับซ้อนของโปรแกรมมากเท่านั้น และปัญหาอีกประการหนึ่งขององค์กรก็คือ ระบบงานที่เกี่ยวข้องกับระบบงานขององค์กรอื่นๆ เช่น EDI (Electronic Data Interchange) ระบบการซื้อขายของตลาดหุ้น ระบบการจองตั๋วเครื่องบิน ฯลฯ

ความยากในการแก้ไขโปรแกรมก็คือ การค้นหาคำสั่งที่เกี่ยวข้องกับวัน เวลา ที่จะต้องดูกันอย่างละเอียด เช่น การกำหนดตัวแปรกำหนดอย่างไร ตัวแปรไหนถูกใครเรียกไปใช้ การเรียกใช้ไฟล์ (File) อื่น การเรียกใช้ไลบรารี (Library) รวมถึงการเรียกใช้ Routine , Subroutine ต่างๆ

- เมื่อได้แก้ไขแล้ว จะสามารถแน่ใจได้อย่างไรว่าทุกคำสั่งที่เกี่ยวข้องกับวันที่ (Date) และเวลา (Time) ได้รับการแก้ไขหมดแล้ว หากมีส่วนใดส่วนหนึ่งยังไม่ได้รับการแก้ไข จะส่งผลกระทบต่อการทำงานของโปรแกรมตัวอื่นๆ ที่เกี่ยวข้อง ซึ่งจำนวนโปรแกรมมีเป็นจำนวนมากหลายพันล้านบรรทัด นับว่าเป็นงานหนักมากทีเดียวในการตามแก้ไข

ข.4 แนวทางการแก้ปัญหา Y2K

แนวทางการแก้ไขปัญหา Y2K แบ่งได้เป็น 5 ขั้นตอน ดังแสดงในภาพ ข.



ภาพที่ ข. แสดงแนวทางแก้ไข Y2K

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) ความตื่นตัว (Awareness) โดยทำความเข้าใจในปัญหา Y2K สาเหตุ และผลกระทบทั้งในกลุ่มผู้เกี่ยวข้อง และระดับผู้บริหาร จัดตั้งคณะทำงาน และพัฒนากลยุทธ์ในการแก้ปัญหา
- 2) การประเมิน (Assessment) ประเมินผลกระทบของปัญหา Y2K ต่อองค์กรโดยรวม ระบุแผนธุรกิจและกระบวนการที่เกี่ยวข้อง จัดทำรายการอุปสรรค และซอฟต์แวร์ ทั้งที่มีปัญหาและไม่มีปัญหา จัดลำดับความสำคัญของการเปลี่ยนแปลงระบบ หรือทดแทนระบบเดิม จัดทำแผนงานเตรียมบุคลากรและงบประมาณ พัฒนาแผนงานฉุกเฉิน เพื่อแก้ปัญหา การแลกเปลี่ยนข้อมูล การขาดข้อมูล หรือข้อมูลผิดพลาด ระบุและจัดหาทรัพยากรต่างๆ ที่จำเป็น
- 3) การปรับปรุงแก้ไข (Renovation) โดยทำการปรับเปลี่ยน (Re-engineer , Re-develop) แก้ไข (Repair) ทดแทน (Replace) หรือเลิกใช้ระบบงานเดิม (Retire)
- 4) การทดสอบ ตรวจสอบและยืนยันความถูกต้อง (Validation) ของระบบที่ถูกเปลี่ยน หรือถูกทดแทน รวมถึงการทดสอบความสามารถในการทำงานกับระบบอื่นๆ ด้วย
- 5) การติดตั้งใช้งานระบบใหม่ (Implementation) การนำระบบใหม่ที่พร้อมใช้งานในปี 2000 มาปฏิบัติงานจริง

ข.5 การแก้ปัญหาปี ค.ศ. 2000

แนวทางการแก้ปัญหาปี ค.ศ. 2000 สามารถแยกออกมาเป็นขั้นๆ เพื่อวางแผนการดำเนินการ ซึ่งประกอบด้วย การเตรียมการ การดำเนินการ การทดสอบและนำไปใช้งาน

1) ในการเตรียมการ แต่ละองค์กรมีงานที่แตกต่างกัน สิ่งแรกคือ การสำรวจจำนวนโปรแกรมที่ใช้ในองค์กร แยกแยะโปรแกรมต่างๆ ทำรายการเหล่านั้น ตรวจสอบดูว่า ปัญหาที่เกิดขึ้นจากปี 2000 ที่จะมีต่อโปรแกรมเป็นอย่างไรบ้าง ถ้าเป็นไปได้ทำความเข้าใจกับปัญหา ซึ่งให้ดูทั้งพีซี ฮาร์ดแวร์ ซอฟต์แวร์ พยายามดูว่ามีปัญหาที่ส่วนใดบ้าง ศึกษาในรายละเอียด และดูผลกระทบว่า โปรแกรมแต่ละโปรแกรมอาจมีอะไรผิดพลาดบ้าง

2) การดำเนินการ หากเป็นโปรแกรมที่ซื้อใหม่ ตรวจสอบให้แน่ใจว่า ปัญหาเกี่ยวกับวันที่ และวันหลังจากปี ค.ศ. 2000 ไม่มีปัญหาในการเก็บข้อมูลชุดใหม่ ให้เก็บข้อมูล ค.ศ. ครบทั้ง 4 ตัว หากเป็นโปรแกรมที่เคยพัฒนามาแล้ว ถ้าผู้พัฒนาอยู่ ให้เตรียมทีมงานตรวจสอบและแก้ไข โดยอาจจะแก้ไขที่ซอร์สโค้ด หากเป็นโปรแกรมเก่า ตรวจสอบดูแน่แล้ว หาวิธีการแก้ไขปัญหา ซึ่งอาจจะต้องมีการแก้ไขข้อมูลตัวเลขใหม่ โดยป้อนหลัก ค.ศ. ให้มี 4 ตัว หรือเปลี่ยนข้อมูลในฐานข้อมูล ในการคำนวณก็ใช้รูปแบบตัวเลขเต็มครบทั้ง ค.ศ.

3) การทดสอบ เป็นการทดสอบแต่ละโปรแกรมของคนที่พัฒนาขึ้น โดยดูว่าการแก้ไขแต่ละโปรแกรมนั้นใช้งานได้ การทดสอบอาจต้องมีเงื่อนไขทดสอบที่ต่างกัน การทดสอบยังต้องสร้างเงื่อนไขทดลองเฉพาะหลายอย่าง หากเราเข้าใจปัญหาได้ดี ก็จะทำให้การทดสอบได้ผลแน่นอน

ข.6 แนวทางต่างๆ ที่สามารถกระทำได้

ข.6.1 แนวทางที่ทำได้กับข้อมูล

1) ขยายตัวเลข ค.ศ.สองหลักให้เป็นสี่หลัก โดยเฉพาะในการเก็บข้อมูลในฐานะข้อมูล ถึงแม้ว่าจะต้องเก็บด้วยที่เก็บมากขึ้น แต่ก็ไม่มีปัญหาในเรื่องที่เก็บแล้ว เพราะปัจจุบันที่เก็บมีราคาถูกลงมาก

2) อาจกำหนดเงื่อนไขใหม่ เช่น ถ้าตัวเลข 60-99 หมายถึง 1960-1999 แต่ถ้าตัวเลข 00-59 คือ 2000 ถึง 2059 ในกรณีนี้ การคำนวณจะต้องคำนวณใหม่ และใช้สูตรใหม่ เรียกการกำหนดแบบนี้ว่า การเลื่อนจอของข้อมูล (Slide Window)

ข.6.2 เปลี่ยนข้อมูลหรือเปลี่ยนโปรแกรม

ในการแก้ปัญหาที่เกิดขึ้นนี้อาจทำได้ โดยการเปลี่ยนที่ข้อมูล คือ ให้ข้อมูลทุกตัวเป็นปี ค.ศ. สี่หลัก หรือจะแก้ที่โปรแกรม คือ ให้โปรแกรมฉลาดพอที่จะรู้ว่า ให้เติม 19 หรือ เติม 20 ข้างหน้า การแก้ไขที่ข้อมูลหรือ โปรแกรม มีข้อดีข้อเสียที่ต้องพิจารณาทั้งคู่ ดังนี้

ข.6.2.1 เปลี่ยนที่เก็บข้อมูล ข้อดีสำหรับกรณีนี้มีหลายประการ เช่น

- 1) การแก้ไขข้อมูล อาจทำได้ตลอดเวลา
ทำได้แม้แต่ในปี ค.ศ. 1999 เพื่อเตรียมการเข้าสู่ปี ค.ศ. 2000
- 2) แก้เฉพาะข้อมูล จึงทำได้ลดมาตรฐานข้อมูล โดยไม่เกี่ยวข้องกับอัลกอริทึม
- 3) การแก้ไขข้อมูล ทำได้ละเอียด เทคนิคที่ใช้จึงเป็นเทคนิคเดียว
- 4) การแก้ไขที่โปรแกรม อาจเกิดปัญหาอื่น และต้องรีคอมไพล์โปรแกรมใหม่ แต่การแก้ไขที่ข้อมูล อาจไม่ต้องแก้ไขโปรแกรม
- 5) การแก้ไขโปรแกรมมาก จะเกี่ยวข้องกับการรีคอมไพล์ และต้องเสียเวลามาก

ข.6.2.2 เปลี่ยนที่โปรแกรม มีข้อดี คือ

- 1) ไม่ต้องเปลี่ยนข้อมูลจำนวนมาก โดยไม่ต้องให้มีการชิงโครไนซ์ในเรื่องข้อมูล โปรแกรมจะตรวจสอบ และดำเนินการให้
- 2) การแก้ไขในบางโปรแกรม อาจแก้ไขได้เพียงเล็กน้อย บางโปรแกรมอาจแก้ตรงจุดที่ทราบได้ บางโปรแกรมอาจไม่ต้องแก้ไข

3) อาจทำงานได้เร็ว และแก้ไขให้ตรงเป้าหมายในอนาคตได้ดี แต่จุดอ่อนของการแก้ไขโปรแกรม อยู่ที่บางโปรแกรมเมื่อคอมไพล์แล้ว และไม่มีซอร์สโค้ด หรือผู้เขียนโปรแกรมไม่อยู่ อาจทำให้แก้ไขได้ยากหรือทำไม่ได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- กิตติ ภัคดีวัฒนกุล และจำลอง กระจูตสาหะ. 2541. **Visual Basic 5 ฉบับ โปรแกรมเมอร์**. กรุงเทพฯ: ดวงกมลสมัย.
- กรมบัญชีกลาง. 2524. หนังสือค่วนที่สุด ที่ กค. 0502/43146. **ระเบียบกระทรวงการคลังว่าด้วยวิธีปฏิบัติเกี่ยวกับเงินชดเชยการส่งสินค้าออก พ.ศ. 2524.**
- ธาริน สิทธิธรรมชารี. 2542. **คู่มือการเขียนโปรแกรม Microsoft Visual Basic Version 6.0 ฉบับเพื่อใช้งานจริง**. กรุงเทพฯ: ชัคเชส มีเดีย.
- ธาริน สิทธิธรรมชารี และสุรสิทธิ์ ธิวประสพศักดิ์. 2542. **คู่มือการเขียนโปรแกรม Advanced Visual Basic Version 6.0**. กรุงเทพฯ: ชัคเชส มีเดีย.
- Date, C.J. 1993. **An Introduction to Database System**. 6th ed. New York: Addison-Wesley.
- Eliason, Alan L. 1990. **System Development: analysis, design, and Implementation**. 2nd ed. New York: HarperCollinsPublishers
- Hawryszkiewicz, I.T. 1990. **Relational Database Design : An Introduction**. Australia : Prentice Hall.
- Pressman, Roger S. 1997. **Software Engineering A Practitioner's Approach**. 4th ed. New York: McGraw-Hill.
- Rob, Peter and Coronel Carlos. 1997. **Database System: design, Implementation and Management**. 3rd ed. New York: International Thomson.

ประวัติผู้เขียน

ชื่อผู้เขียน	นางรัตนภรณ์ อัสวนุภาพ
วันเดือนปีเกิด	10 ตุลาคม 2511
สถานที่เกิด	จังหวัดขอนแก่น
วุฒิการศึกษาระดับปริญญาตรี	วท.บ.(คณิตศาสตร์)
สถานที่สำเร็จการศึกษาระดับปริญญาตรี	คณะวิทยาศาสตร์ มหาวิทยาลัยนเรศวร
ปีที่สำเร็จการศึกษา	ปีการศึกษา 2533
การทำงาน	ตำแหน่งนักวิชาการคอมพิวเตอร์ 6ว. ศูนย์สารสนเทศ กรมบัญชีกลาง

