การจัดโครงสร้างข้อมูลที่อยู่ใหม่เพื่อการเชื่อมโยงฐานข้อมูล ในระบบสารสนเทศภูมิศาสตร์

นายพิชญ์ ติรณสวัสดิ์

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมสำรวจ ภาควิชาวิศวกรรมสำรวจ
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2546

ISBN 974-17-4032-8

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

ADDRESS DATA STRUCTURE REFORM FOR GIS DATABASE LINKAGE

Mr. Pitch Tiranasawasdi

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Survey Engineering

Department of Survey Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2003

ISBN 974-17-4032-8

หัวข้อวิทยานิพนธ์	การจัดโครงสร้างข้อมูลที่อยู่ใหม่เพื่อการเชื่อมโยงฐานข้อมูลในระบบสารสนเทศภูมิศาสตร์
โดย	นายพิชญ์ ติรณสวัสดิ์
สาขาวิชา	วิศวกรรมสำรวจ
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ วิชัย เยี่ยงวีรชน

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.ดิเรก ลาวัณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการสอบ
(รองศาสตราจารย์ ดร.ชูเกียรติ วิเชียรเจริญ)

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ วิชัย เยี่ยงวีรชน)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อิทธิ ตริสิริสัตยวงศ์)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.บรรเจิด พละการ)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

นายพิชญ์ ติรณสวัสดิ์ : การจัดโครงสร้างข้อมูลที่อยู่ใหม่เพื่อการเชื่อมโยงฐานข้อมูลในระบบสารสนเทศภูมิศาสตร์. (ADDRESS DATA STRUCTURE REFORM FOR GIS DATABASE LINKAGE) อ. ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ วิชัย เยี่ยงวีรชน, 109 หน้า. ISBN 974-17-4032-8.

ปัจจุบันหน่วยงานของรัฐมีการจัดทำข้อมูลขึ้นมาใช้ในหน่วยงานของตนเป็นจำนวนมาก ทำให้การเชื่อมโยงข้อมูล เพื่อที่จะให้หน่วยงานต่างๆ ได้ใช้ข้อมูลร่วมกัน เกิดปัญหาขึ้นคือ ไม่อาจเชื่อมโยงข้อมูลด้วยกันได้ กรมสรรพากร เป็นอีกหน่วยงานหนึ่งที่ต้องการเชื่อมข้อมูลของหน่วยงานต่างๆ เช่น ข้อมูลการใช้ไฟฟ้า ข้อมูลประกันสังคม ข้อมูลอาคารและข้อมูลเลขประจำตัวผู้เสียภาษี ของกรมสรรพากร สำหรับขยายฐานภาษี มีการดำเนินโครงการนำร่องเพื่อขยายฐานภาษี ได้มีการเชื่อมโยงข้อมูลฐานข้อมูลจากหน่วยงานเหล่านี้ซึ่งพบว่า มีปัญหาในการพิจารณาเลือกใช้ตัวเชื่อมโยงเข้ากับข้อมูลเชิงตำแหน่งในระบบแผนที่ภาษี การดำเนินงานเชื่อมโยงได้กระทำด้วยมือจากข้อมูล “ที่อยู่” ที่ปรากฏในฐานข้อมูลของแต่ละหน่วยงานและข้อมูล “ที่อยู่” ที่ได้จากการสำรวจ ผลปรากฏว่าสามารถเชื่อมโยงได้ประมาณ 76% โดยต้องสูญเสียกำลังคนในการดำเนินงานทั้งสิ้น 2 เดือน อันเนื่องมาจากความไม่สอดคล้องกันของข้อมูล “ที่อยู่” ของแต่ละหน่วยงาน การกำหนดโครงสร้างการเชื่อมโยงข้อมูลเป็นเรื่องที่มีความจำเป็นอย่างยิ่งเพราะในอนาคตปริมาณข้อมูลที่ทำการจัดเก็บจะมีปริมาณมากขึ้น มีหลายหน่วยงานที่มีแนวคิดที่จะนำข้อมูลมาใช้ร่วมกันโดยกำหนดให้ใช้ รหัสประจำบ้าน (HID) เป็นตัวเชื่อม เพื่อให้การวิเคราะห์ การวางแผนเพื่อตัดสินใจ และนำข้อมูลที่มีอยู่มาใช้ให้เกิดประสิทธิภาพสูงสุด จึงมีการสร้างรหัสจากข้อมูล “ที่อยู่” (NEWID) ขึ้นมาใช้ก่อนที่จะมี HID อยู่ในทุกๆ หน่วยงาน

ในวิทยานิพนธ์ฉบับนี้ทำการจัดโครงสร้างข้อมูล “ที่อยู่” ในแต่ละหน่วยงานให้มีโครงสร้างเดียวกัน เป็นแนวทางหนึ่งในการใช้ประโยชน์ข้อมูล “ที่อยู่” ที่มีการจัดเก็บในทุกหน่วยงาน เป็นส่วนเบื้องต้นเพื่อให้สามารถเชื่อมโยงฐานข้อมูลได้ และช่วยนำไปสู่การใช้รหัสประจำบ้าน เป็นรหัสมาตรฐานสำหรับการเชื่อมโยงฐานข้อมูลต่อไปในระบบสารสนเทศภูมิศาสตร์ โดยใช้โปรแกรมการจัดโครงสร้างข้อมูลที่อยู่ เป็นตัวช่วยในการแก้ไขและปรับปรุงให้ข้อมูลมีความถูกต้องมากขึ้นเพื่อเพิ่มเปอร์เซ็นต์การเชื่อมโยงข้อมูลที่อยู่ให้มีความถูกต้องถึง 85% ในเวลาอันรวดเร็วกว่าการเชื่อมแบบเดิม

ภาควิชา วิศวกรรมสำรวจ ลายมือชื่อนิสิต.....

สาขาวิชา วิศวกรรมสำรวจ ลายมือชื่ออาจารย์ที่ปรึกษา.....

ปีการศึกษา 2546

4370419221 : MAJOR SURVEY ENGINEERING

KEY WORD: DATABASE / LINKAGE / ADDRESS / RESTRUCTURE / ORGANIZATION

MR.PITCH TIRANASAWASDI : ADDRESS DATA STRUCTURE REFORM FOR GIS DATABASE LINKAGE. THESIS ADVISOR : ASST.PROF. VICHAI YIENGVEERACHON, 109 pp. ISBN 974-17-4032-8.

Many government agencies in Thailand have established and exploited the use of databases to link and share data among organizations. The Revenue Department of Thailand is considered as a significant government unit that needs to link its database with other organizations' ; for example, data associated with the consumption of electricity from Metropolitan Electricity Authority, social security number from Ministry of Labor, building data and tax identification number from The Revenue Department of Thailand, etc. for improving taxation. The "Restructure Program" has been created in order to link data from these organizations' database. A problem concerning the selection of the primary key with the spatial data of tax map system was found. Thus, the attempt to transform and restructure "address" data maintained in particular organizations and link it to "address" data obtained from the survey was done manually. The process took 2 months since the data from each organization is not matched. The final result has shown that the correctness of the data linkage is at 76%. The restructuring of data is an essential task that needs to be done due to the amount data stored in the database is increasing rapidly at present. Furthermore, many organizations have planned to utilize the data produced, used, and maintained by other organizations in the most effective and efficient aspect for data sharing, data analysis, and decision making. Therefore, those organizations mentioned above have considered to link their data by using "House Identification" (HID). However, before HID is widely used in those organizations, a specific field derived from "Address" that is used in this research must be created and used in order to link the data instead.

By using "Restructure Program" in this project would increase the correctness of data linkage up to 85% while the time consumption would be decreased. Hopefully, this project would lead to the use of house identification number as a key identifier to link the data in an organization to the others for any GIS database in the government organization.

Department..Survey Engineering..... Student's signature.....

Field of study..Survey Engineering..... Advisor's signature.....

Academic year 2003.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของผู้ช่วยศาสตราจารย์ วิชัย เยี่ยงวีรชน อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำแนะนำและข้อคิดเห็นต่างๆ ในการวิจัยมาด้วยดีตลอด รวมทั้งตรวจแก้วิทยานิพนธ์ฉบับนี้อย่างละเอียด

ขอขอบพระคุณ คุณรังษี สุไลมาน และคุณปิติพันธ์ จีระมงคลพาณิชย์ ที่ได้ให้คำปรึกษาในเรื่องการเขียนโปรแกรม

ขอขอบพระคุณ อาจารย์ณิดา ดรงค์สุวรรณ และคุณตติยะ ชื่นตระกูล ที่ได้ให้คำปรึกษาในเรื่องการใช้ภาษาอังกฤษ

ขอขอบพระคุณ คุณมาลิน ผลบุตร ที่ได้ให้คำปรึกษาในเรื่องการใช้ภาษาไทยและช่วยตรวจแก้คำผิดในวิทยานิพนธ์ฉบับนี้ด้วย

ท้ายนี้ ผู้วิจัยใคร่ขอกราบขอบพระคุณบิดา-มารดา คุณยาย น้องสาว ญาติทั้งหลายและพี่ๆ เพื่อนๆ ที่กรมที่ดินและที่จุฬาฯ ซึ่งได้คอยเกี่ยวเข็ญให้ทำวิทยานิพนธ์ อยู่ตลอด และได้สนับสนุนทุกสิ่งทุกอย่าง ให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

นายพิชญ์ ติรณสวัสดิ์

ธันวาคม 2546

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ฎ

สารบัญภาพ ..... ฎ

1. บทนำ 1

1.1 ความสำคัญและความเป็นมาของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 2

1.3 ขอบเขตของการวิจัย 2

1.4 คำจำกัดความที่ใช้ในการวิจัย 3

1.5 ประโยชน์ที่คาดว่าจะได้รับ 4

1.6 วิธีดำเนินการวิจัย 4

2. ข้อมูลหน่วยงานต่างๆ ที่เกี่ยวข้อง 8

2.1 สถานะปัจจุบันของข้อมูล “ที่อยู่” 8

2.1.1 องค์ประกอบของข้อมูล “ที่อยู่” 8

2.1.2 ระบบข้อมูล “ที่อยู่” ของกรมการปกครอง 9

2.1.3 ระบบข้อมูล “ที่อยู่” ของหน่วยงานภาครัฐ 9

2.2 โครงสร้างข้อมูล “ที่อยู่” ของหน่วยงานที่ใช้ในการวิจัย 12

2.2.1 โครงสร้างฐานข้อมูลการใช้ไฟฟ้า
(ข้อมูลจากการไฟฟ้านครหลวง) 12

2.2.2 โครงสร้างฐานข้อมูลประกันสังคม
(ข้อมูลจากสำนักงานประกันสังคม) 13

2.2.3 โครงสร้างฐานข้อมูลอาคาร
(ข้อมูลจากกรมสรรพากร) 14

2.2.4 โครงสร้างฐานข้อมูลเลขประจำตัวผู้เสียภาษี
(ข้อมูลจากกรมสรรพากร) 15

2.3 สรุปลักษณะโครงสร้างข้อมูล “ที่อยู่” 16

2.3.1 แบบที่อยู่รวมกันภายในฟิลด์เดียว 16

หน้า

2.3.2 แบบที่ถูกแยกเป็นฟิลด์ไว้แล้ว 16
2.4 พฤติกรรมการป้อนข้อมูล “ที่อยู่” 16
2.4.1 พฤติกรรมการป้อนตัวอักษรภาษาไทย 16
2.4.2 รูปแบบการจัดเก็บข้อมูล “ที่อยู่” 17
2.5 ประเด็นปัญหาสำคัญต่างๆ ที่พบของข้อมูล “ที่อยู่” 19
2.5.1 ปัญหาเรื่องของโครงสร้าง 19
2.5.2 ปัญหาเรื่องการเลือกใช้อักษรย่อ 19
2.5.3 ปัญหาเรื่องการนำเข้าข้อมูล 20
2.5.4 ปัญหาเรื่องการแยกข้อมูล “ที่อยู่” 20
2.5.5 ปัญหาเรื่องความทันสมัยของข้อมูล 21
2.6 แนวทางการแก้ปัญหา 21
2.6.1 การกำหนดข้อตกลงร่วมในการจัดเก็บข้อมูล “ที่อยู่” 21
2.6.1.1 การกำหนดโครงสร้างข้อมูล “ที่อยู่” ใหม่ 21
2.6.1.2 การกำหนดการใช้อักษรย่อต่างๆ 21
2.6.1.3 การกำหนดการนำเข้าข้อมูล “ที่อยู่” 21
2.6.1.4 การปรับปรุงข้อมูลให้เป็นปัจจุบัน 22
2.6.2 การพัฒนาโปรแกรมช่วยในการจัดการข้อมูลเดิมที่มีอยู่ ในฐานข้อมูล 22
2.6.2.1 แนวทางการแก้ปัญหาเรื่องลำดับการป้อนอักษรไทย 22
2.6.2.2 แนวทางการปัญหาเรื่องการแยกข้อมูล “ที่อยู่” 22
3. การจัดโครงสร้างข้อมูล “ที่อยู่” 23
3.1 การออกแบบโครงสร้างข้อมูลกลาง 23
3.2 การออกแบบกรรมวิธีการเชื่อมโยงข้อมูล 24
3.3 ขั้นตอนการจัดข้อมูล “ที่อยู่” ของหน่วยงานที่ศึกษา 26
3.3.1 ฐานข้อมูลการใช้ไฟฟ้า ของการไฟฟ้านครหลวง 26
3.3.2 ฐานข้อมูลประกันสังคม ของสำนักงานประกันสังคม 27
3.3.3 ฐานข้อมูลอาคาร ของกรมสรรพากร 28
3.3.4 ฐานข้อมูลเลขประจำตัวผู้เสียภาษี ของกรมสรรพากร 30
3.4 ขั้นตอนการตรวจสอบและแก้ไขข้อมูล “ที่อยู่” 31
3.4.1 เกณฑ์ในการกำหนดเรคคอร์ดที่พิมพ์ผิด 31

หน้า

3.5 สถานภาพข้อมูลหลังจากแยกข้อมูล “ที่อยู่” โดยใช้โปรแกรม 32
3.5.1 กรณีที่ 1 ไม่ถือว่าเรคคอร์ดที่มีไม่ครบทุกฟิลด์เป็นเรคคอร์ดที่พิมพ์ผิด 32
3.5.2 กรณีที่ 2 ถือว่าเรคคอร์ดที่มีไม่ครบทุกฟิลด์เป็นเรคคอร์ดที่พิมพ์ผิด 34
4. การพัฒนาโปรแกรมจัดโครงสร้าง “ที่อยู่” 36
4.1 แนวความคิดในการพัฒนาโปรแกรม 36
4.2 ตารางเปรียบเทียบ (Look Up Table) 38
4.2.1 ผังระบบโปรแกรม 41
4.3 ฟังก์ชันของโปรแกรม 52
4.3.1 ฟังก์ชันสร้างและการเปิดฐานข้อมูล 52
4.3.2 ฟังก์ชันสร้างตารางฐานข้อมูล 53
4.3.3 ฟังก์ชันดึงข้อมูล 54
4.3.4 ฟังก์ชันแยกที่อยู่ 56
4.3.5 ฟังก์ชันแก้ไขข้อมูล 57
4.3.6 ฟังก์ชันแก้ไขข้อมูลระเบียน 60
4.3.7 ฟังก์ชันเข้ารหัส 62
4.3.8 ฟังก์ชันปิดการทำงานของโปรแกรม 63
4.4 การจัดทำ Library 64
4.5 กรณีศึกษาอื่นๆ 65
4.5.1 กรณีที่จัดเก็บข้อมูล “ที่อยู่” ในฟิลด์เดียว 65
4.5.2 กรณีที่จัดเก็บข้อมูล“ที่อยู่” แยกฟิลด์ 66
5. ผลการทดสอบ 67
5.1 การปรับปรุงฐานข้อมูลต่างๆ เพื่อการเชื่อมโยงฐานข้อมูล 67
5.2 ผลของการปรับปรุงความครบถ้วนฐานข้อมูล 69
5.3 การเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลอาคาร โดยใช้โปรแกรม MapInfo 74
5.4 ผลของการเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลอาคาร 79
5.4.1 ผลการเชื่อมโยงฐานข้อมูลก่อนทำการปรับปรุง 79
5.4.2 ผลการเชื่อมโยงฐานข้อมูลหลังทำการปรับปรุง 79
5.5 ผลของการเชื่อมโยงฐานข้อมูลต่างๆ เข้าด้วยกัน 80
6. บทสรุปและข้อเสนอแนะ 82

หน้า

6.1 สรุป 82
6.1.1 สรุปปัญหาสำคัญต่างๆ ที่พบของข้อมูล “ที่อยู่” 82
6.1.2 สรุปปัญหาความผิดพลาดของข้อมูล 82
6.1.3 สรุปผลการเชื่อมโยงฐานข้อมูล 84
6.2 ปัญหาอุปสรรค 84
6.3 ข้อเสนอแนะ 85
6.3.1 ข้อเสนอแนะในการแก้ปัญหาสำคัญต่างๆ ที่พบของข้อมูล “ที่อยู่” 85
6.3.2 แนวทางการแก้ปัญหาความผิดพลาดของข้อมูล 85
6.3.3 การเชื่อมโยงเข้ากับ HID ของสำนักบริหารการทะเบียน 86

รายการอ้างอิง 88
ภาคผนวก 90
ภาคผนวก ก รหัสโปรแกรม ส่วนที่ทำเป็นไฟล์ Library 91
ภาคผนวก ข Look Up Table ของซอยและถนนในเขตสัมพันธวงศ์ 105
ประวัติผู้เขียนวิทยานิพนธ์ 109

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตาราง หน้า

ตารางที่ 2.1 จำนวนหน่วยงานภาครัฐระดับกรมที่มีระบบข้อมูล
เกี่ยวกับที่อยู่โดยตรง จาก 141 แห่งที่สำรวจ 11

ตารางที่ 2.2 โครงสร้างฐานข้อมูลการใช้ไฟฟ้า ของการไฟฟ้า 12

ตารางที่ 2.3 โครงสร้างฐานข้อมูลประกันสังคม ของสำนักงานประกันสังคม 13

ตารางที่ 2.4 โครงสร้างฐานข้อมูลอาคาร ของกรมสรรพากร 14

ตารางที่ 2.5 โครงสร้างฐานข้อมูลเลขประจำตัวผู้เสียภาษี ของกรมสรรพากร 15

ตารางที่ 3.1 โครงสร้างฐานข้อมูลกลาง 24

ตารางที่ 3.2 ความผิดพลาดในการพิมพ์บันทึกข้อมูลของหน่วยงานต่างๆ ในกรณีที่ 1 32

ตารางที่ 3.3 ความผิดพลาดในการพิมพ์บันทึกข้อมูลของหน่วยงานต่างๆ ในกรณีที่ 2 34

ตารางที่ 4.1 โครงสร้างฐานข้อมูล Look Up Table ของซอย ตรอก และถนน 38

ตารางที่ 4.2 โครงสร้างฐานข้อมูล Look Up Table ของตำบล 39

ตารางที่ 4.3 โครงสร้างฐานข้อมูล Look Up Table ของอำเภอ 39

ตารางที่ 4.4 โครงสร้างฐานข้อมูล Look Up Table ของจังหวัด 40

ตารางที่ 5.1 ผลการปรับปรุงข้อมูลการใช้ไฟฟ้า (การไฟฟ้า) จำนวน 11,029 เรคคอร์ด 70

ตารางที่ 5.2 ผลการปรับปรุงข้อมูลประกันสังคม (สำนักงานประกันสังคม)
จำนวน 612 เรคคอร์ด 71

ตารางที่ 5.3 ผลการปรับปรุงข้อมูลอาคาร (กรมสรรพกร) จำนวน 8,944 เรคคอร์ด 72

ตารางที่ 5.4 ผลการปรับปรุงข้อมูลผู้เสียภาษี ของกรมสรรพากร จำนวน 6,765 เรคคอร์ด 72

ตารางที่ 5.5 เปรียบเทียบผลการเชื่อมโยงข้อมูลแต่ละหน่วยงานกับข้อมูลอาคาร 79

ตารางที่ 5.6 ตารางตัวอย่างเปรียบเทียบความไม่เข้ากันของฐานข้อมูล 80

ตารางที่ 5.7 ผลของการเชื่อมโยงฐานข้อมูลระหว่างฐานข้อมูลต่างๆ 80

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

ภาพประกอบ หน้า

รูปที่ 1.1 แผนผังการสร้างข้อกำหนดการนำเข้าข้อมูล(พื้นฐานเบื้องต้น) 4

รูปที่ 1.2 ผังวิธีการดำเนินงานวิจัย 7

รูปที่ 3.1 แผนผังการออกแบบกรรมวิธีการเชื่อมโยงฐานข้อมูล 25

รูปที่ 3.2 แสดงการจัดโครงสร้าง “ที่อยู่” ใหม่จากโครงสร้างฐานข้อมูลการใช้ไฟฟ้า 26

รูปที่ 3.3 แสดงการจัดโครงสร้าง “ที่อยู่” ใหม่จากโครงสร้างฐานข้อมูลกรมประกันสังคม 27

รูปที่ 3.4 แสดงการจัดโครงสร้าง “ที่อยู่” ใหม่จากโครงสร้างฐานข้อมูลอาคาร
ของกรมสรรพากร 29

รูปที่ 3.5 แสดงการจัดโครงสร้าง “ที่อยู่” จากโครงสร้างฐานข้อมูล
เลขประจำตัวผู้เสียภาษีนิติบุคคล 30

รูปที่ 3.6 กราฟแสดงการเปรียบเทียบเปอร์เซ็นต์ความผิดพลาดในการบันทึกข้อมูล
ระหว่างหน่วยงานต่างๆ ในกรณีที่ 1 33

รูปที่ 3.7 กราฟแสดงการเปรียบเทียบเปอร์เซ็นต์ความผิดพลาดในการบันทึกข้อมูล
ระหว่างหน่วยงานต่างๆ ในกรณีที่ 2 34

รูปที่ 4.1 แผนผังการทำงานของโปรแกรมในการสร้างรหัส (NEWID) 37

รูปที่ 4.2 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของซอย ตรอก และถนน 38

รูปที่ 4.3 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของตำบล 39

รูปที่ 4.4 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของอำเภอ 40

รูปที่ 4.5 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของจังหวัด 40

รูปที่ 4.6 แผนผังการทำงานของโปรแกรมในส่วนการแยก “ที่อยู่”
กรณีที่จัดเก็บข้อมูล “ที่อยู่” ในฟิลด์เดียว 42

รูปที่ 4.7 แผนผังการเทียบรหัสอำเภอและจังหวัดของการไฟฟ้า
กับ LUT อำเภอและ LUT จังหวัด 45

รูปที่ 4.8 แผนผังการเทียบรหัสอำเภอและจังหวัดของกรมการประกันสังคม
กับ LUT อำเภอและ LUT จังหวัด 46

รูปที่ 4.9 แผนผังการทำงานของโปรแกรมในส่วนการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล
“ที่อยู่” แยกฟิลด์ ของข้อมูลเลขประจำตัวผู้เสียภาษี (กรมสรรพากร) 49

รูปที่ 4.10 แผนผังการทำงานของโปรแกรมในส่วนของการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล
“ที่อยู่” แยกฟิลด์ ของข้อมูลอาคาร (กรมสรรพากร) 50

หน้า

รูปที่ 4.11 แสดงเมนูสร้างฐานข้อมูล 52
รูปที่ 4.12 แสดงวินโดว์สำหรับเปิดฐานข้อมูล 53
รูปที่ 4.13 แสดงเมนูสร้างตาราง 53
รูปที่ 4.14 แสดงวินโดว์สำหรับสร้างตาราง 54
รูปที่ 4.15 แสดงเมนูดึงข้อมูลเข้าตาราง 55
รูปที่ 4.16 แสดงวินโดว์สำหรับดึงข้อมูลเข้าตาราง 55
รูปที่ 4.17 แสดงเมนูแยกที่อยู่ 56
รูปที่ 4.18 แสดงวินโดว์สำหรับแยกที่อยู่ 57
รูปที่ 4.19 แสดงเมนูแก้ 57
รูปที่ 4.20 แสดงวินโดว์สำหรับแก้ข้อมูล 58
รูปที่ 4.21 แสดงวินโดว์สำหรับแก้ข้อมูลที่ถูกเลือก 59
รูปที่ 4.22 แสดงวินโดว์สำหรับกำหนดเงื่อนไข 59
รูปที่ 4.23 แสดงตัวอย่างการเลือกโดยกำหนดเงื่อนไขเอง 60
รูปที่ 4.24 แสดงเมนูแก้ไขข้อมูลระเบียน 61
รูปที่ 4.25 แสดงวินโดว์สำหรับแก้ไขข้อมูลระเบียน 61
รูปที่ 4.26 แสดงวินโดว์สำหรับลบข้อมูลระเบียน 62
รูปที่ 4.27 แสดงเมนูเข้ารหัส 62
รูปที่ 4.28 แสดงวินโดว์สำหรับเข้ารหัส 63
รูปที่ 4.29 แสดงเมนู References 64
รูปที่ 4.30 การเลือกไฟล์ Library 65
รูปที่ 5.1 ตัวอย่างแสดงการแก้ไขฟิลด์ซอยและถนนโดยใช้ฟิลด์ ADDRESS ช่วย 68
รูปที่ 5.2 ตัวอย่างแสดงการแก้ไขฟิลด์ตำบลให้ครบถ้วนโดยใช้ฟิลด์ SOI ช่วย 69
รูปที่ 5.3 กราฟแสดงการเปรียบเทียบเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นของแต่ละหน่วยงาน 73
รูปที่ 5.4 ตัวอย่างแสดงการเปิดตารางเพื่อทำการเชื่อมข้อมูล 75
รูปที่ 5.5 ตัวอย่างแสดงเมนูการเชื่อมตาราง 75
รูปที่ 5.6 แสดงวินโดว์สำหรับตั้งเงื่อนไขการเชื่อมข้อมูล 76
รูปที่ 5.7 แสดงการเลือกตารางและฟิลด์เพื่อทำการเชื่อมข้อมูล 76
รูปที่ 5.8 แสดงข้อมูลที่สามารถเชื่อมข้อมูลได้ 77
รูปที่ 5.9 ตัวอย่างแสดงเมนูการบันทึกข้อมูลเป็นกราฟิก 77

หน้า

รูปที่ 5.10 ตัวอย่างวินโดว์เลือกไฟล์บันทึกข้อมูลเป็นกราฟิก 78
รูปที่ 5.11 ตัวอย่างวินโดว์บันทึกข้อมูลเป็นกราฟิก 78

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1

บทนำ

1.1 ความสำคัญและความเป็นมาของปัญหา

ภารกิจและบทบาทของกรมสรรพากรคือ การหารายได้ให้รัฐ การจัดเก็บภาษีอากรตามประมวลรัษฎากร และกฎหมายอื่นที่เกี่ยวข้อง มีวัตถุประสงค์หลักคือ จัดเก็บภาษีในระดับที่เหมาะสมและทั่วถึงด้วยค่าใช้จ่ายที่ต่ำของกรมสรรพากรและผู้เสียภาษี มีการเก็บภาษีเงินได้บุคคลธรรมดา ภาษีเงินได้นิติบุคคล ภาษีมูลค่าเพิ่ม ภาษีธุรกิจเฉพาะ อากรแสตมป์ ภาษีเงินได้ปิโตรเลียม ฯลฯ เพื่อให้บรรลุวัตถุประสงค์จึงมีการนำ GIS มาใช้เพื่อขยายฐานภาษี โดยนำข้อมูลจากหน่วยงานอื่นมาใช้ร่วมกัน เช่น ข้อมูลการใช้ไฟฟ้าของการไฟฟ้า เพื่อดูว่าปริมาณการใช้ไฟฟ้าสอดคล้องกับข้อมูลอาคารของกรมสรรพากรหรือไม่ เป็นการตรวจสอบผู้ที่หลีกเลี่ยงการจ่ายภาษี เพื่อให้เก็บภาษีได้ทั่วถึงวิธีหนึ่ง โดยข้อมูลอาคารเป็นข้อมูลที่ได้จากการสำรวจภาคสนามของกรมสรรพากรและเชื่อมโยงกับข้อมูลกราฟิกของตัวอาคาร ข้อมูลประกันสังคมของสำนักงานประกันสังคม เป็นข้อมูลของเจ้าของกิจการที่ได้จดทะเบียนเป็นสมาชิก สามารถขยายฐานภาษีได้จากเจ้าของกิจการที่จดทะเบียนเป็นสมาชิกของสำนักงานประกันสังคม แต่ไม่ได้เสียภาษีให้กรมสรรพากร และ ข้อมูลเลขประจำตัวผู้เสียภาษีของกรมสรรพากร ฯลฯ โดยใช้ข้อมูล “ที่อยู่” เป็นตัวเชื่อมเพราะข้อมูล “ที่อยู่” เป็นข้อมูลที่ถูกจัดเก็บอยู่ในฐานข้อมูลของทุกหน่วยงาน ซึ่งปัญหาที่พบคือโครงสร้างของข้อมูล “ที่อยู่” ในแต่ละหน่วยงานมีความแตกต่างกัน ทำให้เกิดปัญหาการเชื่อมโยงข้อมูลไม่สามารถเชื่อมโยงข้อมูลเข้าด้วยกันได้ และปัญหาความไม่สอดคล้องกันของข้อมูลในแต่ละหน่วยงาน โดยที่ผ่านมามีการกำหนดให้ใช้รหัสประจำบ้าน(HID) เพื่อแก้ปัญหาต่างๆ เหล่านี้ ซึ่งเป็นข้อมูลที่ในแต่ละหน่วยงานควรมีไว้เพื่อประโยชน์ในการใช้ข้อมูลร่วมกันในอนาคต แต่ในปัจจุบันยังไม่มีหน่วยงานใดที่เก็บข้อมูลรหัสประจำบ้าน นอกจากสำนักบริหารการทะเบียน กรมการปกครอง เนื่องจากการนำข้อมูลรหัสประจำบ้านมาเก็บไว้ในฐานข้อมูลนั้นจะต้องใช้งบประมาณ กำลังคน และระยะเวลาในการดำเนินงานมาก เพื่อเป็นการแก้ไขปัญหาในการเชื่อมโยงฐานข้อมูลเข้าด้วยกันก่อนที่ทุกหน่วยงานจะมีการจัดเก็บข้อมูลรหัสประจำบ้าน จึงมีการใช้ประโยชน์จากข้อมูล “ที่อยู่” มาสร้างรหัสที่ใช้ในการเชื่อมโยงฐานข้อมูลของหน่วยงานต่างๆ เข้าด้วยกัน เมื่อนำฐานข้อมูลไปเชื่อมโยงเข้ากับหน่วยงานที่มีข้อมูลรหัสประจำบ้านแล้ว จะทำให้หน่วยงานอื่นที่เชื่อมกับฐานข้อมูลสามารถนำข้อมูลรหัสประจำบ้านนี้ไปใช้ประโยชน์ได้อีกด้วย

งานวิจัยนี้จะศึกษาถึงความผิดพลาดของข้อมูล ความไม่สอดคล้องกันของข้อมูล และแนวทางการแก้ไขปัญหาโครงสร้างที่ต่างกันในแต่ละฐานข้อมูล มีการปรับปรุงฐานข้อมูลให้มีคุณภาพดีขึ้น เพื่อเพิ่มเปอร์เซ็นต์การเชื่อมโยงฐานข้อมูลของกรมสรรพากรกับหน่วยงานต่างๆ ให้มากขึ้น และเกิดประโยชน์สูงสุดในการขยายฐานภาษี

1.2 วัตถุประสงค์ของการวิจัย

1.2.1 จัดรูปแบบโครงสร้างข้อมูล "ที่อยู่" สำหรับใช้ในการเชื่อมโยงฐานข้อมูลต่างๆ ของหน่วยงานทางราชการ เพื่อการขยายฐานภาษีของกรมสรรพากร

1.2.2 เพื่อพัฒนาโปรแกรมในการจัดโครงสร้างฐานข้อมูลเพื่อการเชื่อมโยงฐานข้อมูล โดยทำเป็นฟังก์ชันไว้ใช้งาน

1.2.3 เพื่อเชื่อมฐานข้อมูลแผนที่ในระบบสารสนเทศภูมิศาสตร์ โดยใช้ข้อมูล "ที่อยู่" เป็นตัวเชื่อม

1.2.4 เพื่อทำการเชื่อมโยงฐานข้อมูลต่างๆ เข้าด้วยกัน ก่อนที่ในฐานข้อมูลของแต่ละหน่วยงานจะมีข้อมูล HID

1.3 ขอบเขตของการวิจัย

1.3.1 พื้นที่ในการวิจัยจะใช้เขตสัมพันธวงศ์ ในกรุงเทพฯ โดยใช้ข้อมูลดังนี้

ข้อมูลการใช้ไฟฟ้าของการไฟฟ้า	จำนวน	11,029	เรคคอร์ด
ข้อมูลประกันสังคมของสำนักงานประกันสังคม	จำนวน	612	เรคคอร์ด
ข้อมูลอาคารของกรมสรรพากร	จำนวน	8,944	เรคคอร์ด
ข้อมูลเลขประจำตัวผู้เสียภาษีของกรมสรรพากร	จำนวน	6,765	เรคคอร์ด

โดยข้อมูลที่นำมาใช้เป็นข้อมูลทุติยภูมิ เป็นฐานข้อมูล ปี 2542

1.3.2 ข้อมูลเชิงพื้นที่ (Spatial Data) เป็นข้อมูลมาตราส่วน 1:4,000 เป็นข้อมูลอาคารโดยได้จากการสำรวจข้อมูลที่อยู่ภาคสนามของกรมสรรพากรในปี 2542 ในระบบโปรแกรม MapInfo

1.3.3 โปรแกรมที่ใช้ในการจัดการฐานข้อมูลใช้โปรแกรม MS ACCESS 97

1.3.4 ใช้ Visual Basic ในการพัฒนาโปรแกรมประยุกต์

1.3.5 ใช้ MapInfo เป็นโปรแกรม GIS

1.4 คำจำกัดความที่ใช้ในการวิจัย

การเข้ารหัสภูมิศาสตร์ (Geocoding) เป็นกระบวนการที่นำข้อมูลที่เป็นตารางกับข้อมูลเชิงตำแหน่งเข้าไว้ด้วยกัน เพื่อนำมาใช้ในการวิเคราะห์เชิงตำแหน่ง (U.S. Department of Justice, Office of Community Oriented Policing Services,2000:4)

"Geocoding is the process of bringing tabular and geographic data together based on a common geographic unit of analysis"

ฐานข้อมูล (Database) เป็นการรวมข้อมูลจากหลายแหล่งมาจัดโครงสร้างโดยมีความสัมพันธ์โดยผ่านรหัสสำคัญ (primary key) เพื่อช่วยในการค้นคืนข้อมูล (retrieval)และสนับสนุนการตัดสินใจ (Decision support system)

รหัสประจำบ้าน (House Identification ,HID) คือเลขรหัสที่กำหนดขึ้นโดยสำนักบริหารการทะเบียน กรมการปกครอง เพื่อแทนที่ตั้งหรือที่อยู่ของบ้าน/อาคาร มีทั้งหมด 11 หลัก

หลักที่ 1-2 คือรหัสจังหวัด
หลักที่ 3-4 คือรหัสอำเภอ — รหัสสำนักทะเบียน
หลักที่ 5-10 คือ เลขลำดับที่บ้านเลขที่ของสำนักทะเบียนนั้น
หลักที่ 11 เป็นตัวเลขสำหรับตรวจสอบว่าเลขหลักที่ 1-10 มีความถูกต้องหรือไม่

NEWID เป็นรหัสสำคัญในการเชื่อมโยงข้อมูลมีลักษณะเดียวกับ HID ใช้เป็นตัวเชื่อมฐานข้อมูล โดยฟิลด์นี้จะถูกสร้างจากโครงสร้างกลางที่มาจากข้อมูล "ที่อยู่"

ตารางเปรียบเทียบ (Look Up Table) หรือ (LUT) เป็นตารางที่ใช้ในการเทียบรหัสของซอย ถนน ตำบล อำเภอ และ จังหวัด โดยสำนักบริหารการทะเบียน จะเป็นผู้กำหนดรหัสขึ้นมา โดยจะเทียบชื่อแล้วแสดงออกมาเป็นรหัส

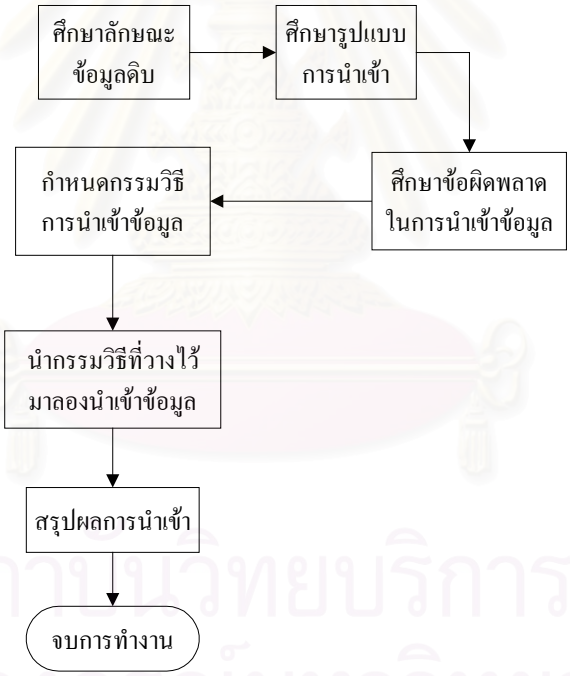
1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 สามารถเชื่อมโยงข้อมูลระหว่างหน่วยงานของการไฟฟ้า กรมสรรพากร และสำนักงานประกันสังคม ใช้ในระบบสารสนเทศภูมิศาสตร์ โดยนำข้อมูลมาวิเคราะห์เพื่อใช้ในการขยายฐานภาษี ของกรมสรรพากร

1.5.2 โปรแกรมในการจัดโครงสร้างฐานข้อมูลเพื่อการเชื่อมโยงฐานข้อมูล โดยทำเป็นฟังก์ชันไว้ใช้งาน

1.6 วิธีดำเนินการวิจัย

1.6.1 ศึกษาการนำเข้าข้อมูล(พื้นฐานเบื้องต้น) โดยมีแผนผังแสดงในรูปที่ 1.1



รูปที่ 1.1 แผนผังการสร้างข้อกำหนดการนำเข้าข้อมูล(พื้นฐานเบื้องต้น)

- การศึกษา ลักษณะข้อมูลดิบ ก่อนการนำเข้าสู่ฐานข้อมูล โดยสังเกตลักษณะข้อมูลโดยรวมว่ามีลักษณะเป็นอย่างไร (จากรูปแบบต่างๆ ก่อนจัดเก็บข้อมูล)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

- การศึกษารูปแบบการนำเข้าข้อมูล ตามลักษณะการใช้ภาษาที่ถูกต้อง โดยกำหนดโครงสร้างการนำเข้าข้อมูลตามหลักการใช้ภาษาไทย เป็นต้นแบบ(รูปแบบการเขียน พยัญชนะ สระ วรรณยุกต์) ที่ถูกต้อง
- ศึกษาข้อผิดพลาดในการนำเข้าข้อมูล ที่แตกต่างจากการนำเข้าตามหลักการการใช้ภาษาไทยที่ถูกต้อง
- กำหนดกรรมวิธีการนำเข้าข้อมูลโดยทั่วไป และ ข้อผิดพลาดที่มักพบเป็นประจำ เป็นส่วนเพิ่มเติมเพื่อช่วยให้ความถูกต้องของข้อมูลเพิ่มขึ้น
- นำรูปแบบกรรมวิธีการนำเข้าข้อมูล (พื้นฐานข้อมูลเบื้องต้น) ที่ได้กำหนด มาทดสอบกับข้อมูลตัวอย่างด้วยการนำเข้าข้อมูลตามรูปแบบใหม่
- นำเข้าข้อมูล (พื้นฐานข้อมูลเบื้องต้น)

1.6.2 ศึกษาอัลกอริธึมที่ใช้ในการเชื่อมฐานข้อมูลในแต่ละหน่วยงาน

- ศึกษาโครงสร้างข้อมูลโดยทั่วไป จากฐานข้อมูลตัวอย่าง ว่ามีลักษณะโครงสร้างข้อมูลโดยทั่วไปเป็นอย่างไร รูปแบบในแต่ละฟิลด์เป็นรูปแบบ ตัวอักษร หรือ ตัวเลข ขนาดความกว้างของฟิลด์ แต่ละฐานข้อมูลมีการกำหนดไว้เป็นอย่างไร
- นำโครงสร้างที่ทำการศึกษา จากหลายฐานข้อมูลมาวิเคราะห์ ความเป็นไปได้ ระหว่างฐานข้อมูลต่างหน่วยงาน โดยการกำหนดเปรียบเทียบ ฟิลด์ที่แสดงอยู่ ว่ามีลักษณะเหมือนหรือแตกต่างกันอย่างไร
- คัดเลือก ฟิลด์ที่มีลักษณะ และความหมาย ที่เหมือนกัน นำมาเปรียบเทียบ ระหว่างฐานข้อมูลต่างหน่วยงาน เพื่อดูรายละเอียดและทำการจัดโครงสร้างข้อมูลกลาง
- จัดลำดับความสำคัญ ของ ฟิลด์รหัสสำคัญ (Primary Key) ที่จำเป็นต้องใช้ในการนำเข้าข้อมูล แต่ละหน่วยงาน โดยการจำแนกเป็น ข้อมูลฟิลด์รหัสสำคัญ และ ข้อมูลฟิลด์รอง โดยดูโครงสร้างตามหลักการพื้นฐาน(โครงสร้างพื้นฐานที่ทุกหน่วยงานต้องมี ถือเป็นข้อมูลฟิลด์หลัก และโครงสร้างย่อยตามรายละเอียดที่แต่ละหน่วยงานต้องการเพิ่มถือเป็น ข้อมูลฟิลด์รอง)

1.6.3 วิเคราะห์ออกแบบเงื่อนไขในการจัดโครงสร้างข้อมูล และออกแบบโครงสร้างการจัดเก็บข้อมูล “ที่อยู่”

1.6.4 พัฒนาโปรแกรม โดยใช้ Library เป็นส่วนประกอบของโปรแกรม

1.6.4.1 การ Key ข้อมูล “ที่อยู่”

- ฟังก์ชันการปรับแก้คำอัตโนมัติ ในการ Key ข้อมูลที่อยู่จะมีตัวตรวจสอบการพิมพ์ตัวอักษรที่ในบางโปรแกรมสามารถพิมพ์ข้อมูลได้หลายรูปแบบ เช่น คำว่าน้ำ บางครั้งสามารถพิมพ์ ไม้โท ก่อน สระอำ หรือ พิมพ์ สระอำก่อนไม้โท ก็ได้ ในโปรแกรมนี้จะทำการเปลี่ยนให้เป็น ไม้โทก่อน สระอำโดยอัตโนมัติ ทำให้ข้อมูลที่ใช้ในการเชื่อมฐานข้อมูลมีความถูกต้องมากยิ่งขึ้น
- ฟังก์ชันการเพิ่ม ลบ และแก้ไขข้อมูล สามารถแก้ไขข้อมูลได้ หรือทำการเพิ่มหรือ ลบข้อมูลได้

1.6.4.2 การจัดรูปแบบโครงสร้างข้อมูล “ที่อยู่”

- ฟังก์ชันการจัดรูปแบบของข้อมูล “ที่อยู่” จะทำการจัดโครงสร้างฟิลด์ “ที่อยู่” ของฐานข้อมูลในหน่วยงานต่างๆ ให้มีโครงสร้างเป็นแบบโครงสร้างข้อมูลกลาง (ในฟังก์ชันนี้จะทำเป็น Library ไว้เรียกใช้)
- ฟังก์ชันการเข้ารหัส จะนำข้อมูลไปเปรียบเทียบกับ Look Up Table แล้วเข้ารหัสเพื่อสร้างฟิลด์ที่จะใช้เป็นตัวเชื่อม

1.6.4.3 การปรับปรุงคุณภาพข้อมูล

- ฟังก์ชันการปรับปรุงคุณภาพข้อมูล จะแสดงข้อมูลทั้งหมดที่ไม่ซ้ำกัน แล้วทำการตรวจสอบดูด้วยตาว่า มีข้อมูลที่ผิดอยู่หรือไม่ จากนั้นก็ทำการแก้ไข โดยมีให้เลือกแก้ไขโดยเลือกแก้ไขเฉพาะที่หรือแก้ไขทั้งหมด

1.6.4.4 การเชื่อมโยงข้อมูลกับข้อมูลกราฟิก

- การเชื่อมโยงข้อมูล จะทำการเชื่อมโยงข้อมูลโดยใช้ฟิลด์ที่สร้างขึ้นมาใหม่จากการเข้ารหัสมาเป็นตัวเชื่อม

1.6.5 สรุปและประเมินผลการทดสอบโปรแกรม และทำการแก้ไขปรับปรุง

1.6.6 สรุปผลการวิจัยและเรียบเรียงวิทยานิพนธ์

โดยขั้นตอนทั้งหมดของการดำเนินการวิจัยจะแสดงในรูปที่ 1.2

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 1.2 ผังวิธีการดำเนินงานวิจัย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ข้อมูลหน่วยงานต่างๆ ที่เกี่ยวข้อง

ในบทนี้จะทำการศึกษาวิเคราะห์ลักษณะและสถานะปัจจุบันของข้อมูล "ที่อยู่" ในหน่วยงานต่างๆ ว่ามีโครงสร้างเป็นอย่างไร มีฟิลด์ใดบ้างที่มีลักษณะเหมือนกันหรือแตกต่างกันอย่างไร สามารถเปลี่ยนรูปแบบโครงสร้างเพื่อนำมาใช้เป็นตัวเชื่อมโยงฐานข้อมูลทั้งหมดเข้าด้วยกันได้หรือไม่ โดยมีจุดประสงค์เพื่อให้ใช้ฐานข้อมูลในระบบสารสนเทศภูมิศาสตร์ (GIS) ให้มีประสิทธิภาพมากยิ่งขึ้น ก่อนที่จะมีการเปลี่ยนมาใช้ระบบรหัสประจำบ้าน (HID) ของสำนักบริหารการทะเบียน โดยนำข้อมูลที่มีอยู่แล้วในฐานข้อมูลของแต่ละหน่วยงานมาทำการวิเคราะห์ศึกษาพฤติกรรมการนำเข้าข้อมูล รวมทั้งจะชี้ให้เห็นประเด็นปัญหาสำคัญต่างๆ ที่พบของข้อมูล "ที่อยู่" พร้อมทั้งเสนอแนวทางการแก้ปัญหาดังกล่าวด้วย

2.1 สถานะปัจจุบันของข้อมูล "ที่อยู่"

ข้อมูล "ที่อยู่" เป็นข้อมูลซึ่งใช้ในระบบสารสนเทศของหน่วยงานภาครัฐทุกแห่ง จากการสำรวจพบว่ามีหลากหลายรูปแบบ ดังรายละเอียดต่อไปนี้ (ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, 2544: 2-1 - 2-2)

2.1.1 องค์ประกอบของข้อมูล "ที่อยู่"

ข้อมูล "ที่อยู่" ซึ่งผนวกอยู่ในระบบสารสนเทศต่างๆ ของหน่วยงานภาครัฐจะประกอบด้วยส่วนต่างๆ ที่สำคัญดังนี้

- เลขที่บ้าน/สำนักงาน
- ตรอก
- ซอย
- ถนน
- ตำบล
- อำเภอ
- จังหวัด
- รหัสไปรษณีย์

2.1.2 ระบบข้อมูล “ที่อยู่” ของกรมการปกครอง

ในระบบทะเบียนราษฎร์ของกรมการปกครอง ซึ่งดำเนินการโดยสำนักบริหารการทะเบียน นั้น มีการจัดเก็บแฟ้มข้อมูล “ที่อยู่” (House) ผนวกไว้เป็นส่วนหนึ่งด้วย ซึ่งมีโครงสร้างที่สำคัญดังนี้

รายการข้อมูล	**ประเภท**	**ความยาว**
เลขรหัสประจำบ้าน (HID)	Char	11
รหัสสำนักทะเบียน	Char	4
รหัสจังหวัด อำเภอ ตำบล หมู่บ้าน	Char	8
รหัสถนน	Char	4
รหัสซอย	Char	4
รหัสตรอก	Char	4
บ้านเลขที่	Char	16
รหัสประเภทของบ้าน	Char	3
รหัสไปรษณีย์	Char	5

2.1.3 ระบบข้อมูล “ที่อยู่” ของหน่วยงานภาครัฐ

ในส่วนของสำนักบริหารการทะเบียน กรมการปกครอง ได้มีการกำหนดเลขรหัสประจำบ้าน (HID) เพื่อแทนที่ตั้งหรือที่อยู่ ของบ้าน/อาคาร ขึ้นมาใช้งานตามโครงสร้างใน 2.1.2 ปรากฏว่าหน่วยงานภาครัฐทุกแห่งไม่ได้มีการนำรหัสดังกล่าวไปใช้งานหรืออ้างอิงแต่อย่างใด ยกเว้นสำนักบริหารการทะเบียน ของกรมการปกครอง เนื่องจากส่วนใหญ่ยังไม่ทราบว่าได้มีการกำหนดเลขรหัสประจำบ้านหรืออาคารสถานที่ต่างๆ ไว้ด้วยแล้ว ซึ่งเป็นตัวเลข 11 หลัก โดยจะปรากฏในเอกสารสำเนาทะเบียนบ้าน ดังนั้น เลขรหัสประจำบ้านจะสามารถแทน “ที่อยู่” ของบ้านได้ และยังสามารถนำไปค้นหาข้อมูลอื่นๆ เกี่ยวกับบ้านดังกล่าวได้ด้วย เพราะในระบบทะเบียนราษฎร์ของกรมการปกครอง ก็จะมีฐานข้อมูลทะเบียนบ้านผนวกอยู่ด้วยแล้ว ซึ่งใช้เลขรหัสประจำบ้านเป็น key หลัก (ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, 2546: 109) ปัญหาเรื่องการกำหนดและใช้งานรหัสข้อมูลในสิ่งเดียวกันที่แตกต่างกันออกไปของหน่วยงานภาครัฐมีมานานแล้ว แต่ยังไม่ได้รับการแก้ไขอย่างจริงจังแต่อย่างใด เพราะเป็นเรื่องที่คาดกันว่าคงทำได้สำเร็จค่อนข้างยาก เนื่องจากต้องตกลงร่วมมือกันจากส่วนราชการหลายฝ่าย และเหตุผลที่สำคัญอีกประการหนึ่งคือ ในอดีตที่ผ่านมาหน่วยงานภาครัฐส่วนใหญ่จะดำเนินการด้านระบบข้อมูลข่าวสารของตนเองอย่างค่อนข้างอิสระ ไม่ได้มีการพึ่งพาแลกเปลี่ยนข้อมูลระหว่างกันมากนัก และส่วนใหญ่ยังไม่พร้อมที่จะให้บริการข้อมูลแก่หน่วยงานอื่น ดังนั้นจึงยังไม่เห็นความสำคัญของการใช้รหัสข้อมูลที่เป็นมาตรฐานเดียวกัน เมื่อหน่วยงานต้องการจะใช้รหัสข้อมูลสิ่งใดและไม่ทราบว่าสิ่งนั้นได้มีการกำหนดรหัส

มาตรฐานไว้ที่ไหนบ้าง ที่สามารถอ้างอิงและขอมาใช้ได้โดยสะดวกแล้ว หน่วยงานก็จะทำการกำหนดรหัสข้อมูลเป็นของตนเองขึ้นมาใหม่ จึงทำให้มีมาตรฐานของรหัสข้อมูลสิ่งเดียวกัน หลากหลายรูปแบบ เช่น รหัสจังหวัด รหัสถนน รหัสคำนำหน้าชื่อ รหัสอาชีพ รหัสประเภทสถานประกอบการ เป็นต้น (ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, 2546: 82-83)

ในข้อเท็จจริงพบว่าบุคคลส่วนหนึ่งจะมีที่อยู่ 2 ประเภท คือ ที่อยู่ตามทะเบียนบ้าน และที่อยู่ตามสภาพที่เป็นจริง เช่น กรณีนิสิตนักศึกษา หรือ ผู้ขายแรงงาน ที่มีภูมิลำเนาในต่างจังหวัดแต่มาเรียนหนังสือหรือทำงานใน กทม. เป็นต้น ซึ่งส่วนหนึ่งของบุคคลเหล่านี้ไม่ได้มีการย้ายที่อยู่จากทะเบียนบ้านเดิมมาเข้ายังทะเบียนบ้านที่อยู่ตามสภาพที่เป็นจริง ดังนั้นการเก็บและใช้งานข้อมูล “ที่อยู่” ของบุคคลในหน่วยงานภาครัฐจึงมีทั้งการเลือกใช้เพียงประเภทใดประเภทหนึ่งและทั้ง 2 ประเภท แตกต่างกันไปตามวัตถุประสงค์ของแต่ละหน่วยงาน ดังตัวอย่างเช่น

สำนักงานบริหารการทะเบียน กรมการปกครอง จะใช้ข้อมูลที่อยู่ตามทะเบียนบ้านเป็นหลักและถือว่าถูกต้องเป็นทางการ เพราะจะต้องใช้ในการเลือกตั้ง และกิจกรรมอื่นๆ อีกมาก

สำนักงานสถิติแห่งชาติ และกระทรวงแรงงานและสวัสดิการสังคม จะใช้ข้อมูล “ที่อยู่” ในสภาพที่เป็นจริง มากกว่าข้อมูล “ที่อยู่” ตามทะเบียนบ้าน เพราะต้องการหาสถิติตัวเลขไปใช้ในการวางแผน และดำเนินงานด้านต่างๆ ที่ถูกต้องกับสภาพที่เป็นจริง โดยจะแสดงหน่วยงานที่มีระบบข้อมูลเกี่ยวกับ “ที่อยู่” ในตารางที่ 2.1

จากการสำรวจหน่วยงานภาครัฐระดับกรม 141 แห่ง พบว่ามีหน่วยงานที่มีระบบข้อมูล “ที่อยู่” โดยตรงจำนวนทั้งสิ้น 53 แห่ง หรือ ประมาณ 39 เปอร์เซ็นต์ ดังแสดงในตารางที่ 2.1 (ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, 2544: 2-3 - 2-4)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 2.1 จำนวนหน่วยงานภาครัฐระดับกรมที่มีระบบข้อมูลเกี่ยวกับ “ที่อยู่” โดยตรง จาก 141 แห่งที่สำรวจ

กระทรวง	จำนวนหน่วยงานระดับกรม	จำนวนหน่วยงานที่มีระบบฐานข้อมูลที่อยู่
สำนักนายกรัฐมนตรี	13	4
กระทรวงกลาโหม	1	0
กระทรวงการคลัง	4	1
กระทรวงการต่างประเทศ	1	0
กระทรวงเกษตรและสหกรณ์	8	2
กระทรวงคมนาคม	8	4
กระทรวงพาณิชย์	5	2
กระทรวงมหาดไทย	6	2
กระทรวงยุติธรรม	3	1
กระทรวงศึกษาธิการ	10	5
กระทรวงสาธารณสุข	6	3
กระทรวงอุตสาหกรรม	4	2
กระทรวงวิทยาศาสตร์และสิ่งแวดล้อม	8	4
ทบวงมหาวิทยาลัย	9	1
กระทรวงแรงงานและสวัสดิการสังคม	6	4
หน่วยงานราชการอิสระ	15	4
รัฐวิสาหกิจ สังกัด สำนักนายกรัฐมนตรี	4	1
รัฐวิสาหกิจ สังกัด กระทรวงกลาโหม	0	0
รัฐวิสาหกิจ สังกัด กระทรวงการคลัง	6	4
รัฐวิสาหกิจ สังกัด กระทรวงเกษตรและสหกรณ์	3	2
รัฐวิสาหกิจ สังกัด กระทรวงคมนาคม	9	5
รัฐวิสาหกิจ สังกัด กระทรวงพาณิชย์	1	0
รัฐวิสาหกิจ สังกัด กระทรวงมหาดไทย	5	2
รัฐวิสาหกิจ สังกัด กระทรวงวิทยาศาสตร์เทคโนโลยี	3	0
รัฐวิสาหกิจ สังกัด กระทรวงศึกษาธิการ	0	0
รัฐวิสาหกิจ สังกัด กระทรวงสาธารณสุข	0	0
รัฐวิสาหกิจ สังกัด กระทรวงอุตสาหกรรม	2	0
องค์การมหาชนอิสระ (กระทรวงวิทยาศาสตร์)	1	0
รวม	141	53

2.2 โครงสร้างข้อมูล “ที่อยู่” ของหน่วยงานที่ใช้ในการวิจัย

ข้อมูลที่นำมาใช้วิเคราะห์โครงสร้างในการทำวิจัย เป็นโครงสร้างที่ในแต่ละหน่วยงานได้ทำการจัดเก็บในฐานข้อมูลอยู่แล้ว โดยจะแสดงให้เห็นถึงลักษณะของโครงสร้างฐานข้อมูลการใช้ไฟฟ้า ของการไฟฟ้านครหลวง โครงสร้างฐานข้อมูลการประกันสังคม ของสำนักงานประกันสังคม โครงสร้างฐานข้อมูลเลขประจำตัวผู้เสียภาษีและโครงสร้างฐานข้อมูลอาคาร ของกรมสรรพากร ดังต่อไปนี้

2.2.1 โครงสร้างฐานข้อมูลการใช้ไฟฟ้า (ข้อมูลจากการไฟฟ้านครหลวง) มีการเก็บข้อมูลของรหัสต่างๆ ที่ใช้ในการไฟฟ้า ข้อมูลเกี่ยวกับผู้ใช้ไฟฟ้า ปริมาณการใช้ไฟฟ้า และจำนวนเงิน

ตารางที่ 2.2 โครงสร้างฐานข้อมูลการใช้ไฟฟ้า ของการไฟฟ้า

Column no.	Data Type	Description	Remark
1-2	Char(2)	ตอน (ตัวอักษร)	
3-5	Numeric(3)	ตอน (ตัวเลข)	
6-9	Numeric(4)	เขตเส้นทาง	
10-15	Numeric(6)	เครื่องวัดฯ เลขที่	
16-75	Char(60)	ชื่อผู้ใช้ไฟฟ้า	
76-135	Char(60)	ที่อยู่ผู้ใช้ไฟฟ้า	
136-137	Char(2)	PY,SP,GV,PL,ML,TL	
138-139	Numeric(2)	ประเภทไฟ	
140	Numeric(1)	อัตรา	
141-145	Numeric(5)	รหัสผู้ใช้ไฟฟ้า/รหัสกิจการ	
146-150	Numeric(5)	AMPERE	
151-158	Numeric(8)	จำนวนหน่วยเฉลี่ย	
159-168	Numeric(3)	จำนวนเงินเฉลี่ย	
169-170	Numeric(2)	รหัสเขต	

โครงสร้างของข้อมูล “ที่อยู่” ที่นำมาใช้ในการสร้างรหัส เพื่อเชื่อมโยงฐานข้อมูลคือ ฟิลด์ที่อยู่ผู้ใช้ไฟฟ้า และฟิลด์รหัสเขต ดังที่ปรากฏในตารางที่ 2.2 โดยลักษณะการจัดเก็บข้อมูล “ที่อยู่” ของการไฟฟ้านครหลวง มีลักษณะในการบันทึกข้อมูล ซึ่งมีรายละเอียดของข้อมูล “ที่อยู่” ทั้งหมดรวมอยู่ภายในฟิลด์เดียวกัน ทำให้ยากต่อการแยกรายละเอียดย่อยภายในฟิลด์ที่มีทั้งข้อมูล บ้านเลขที่

หมู่ ซอย ถนน ตำบล อำเภอ จังหวัด และรหัสไปรษณีย์ ออกจากกันได้ถูกต้องทั้งหมด ทำการแบ่งแยกข้อมูลได้โดยใช้เครื่องหมายเว้นวรรค เป็นตัวคั่นระหว่างข้อมูล

ตัวอย่างของข้อมูลในฟิลด์ “ที่อยู่”

1056/4 ซอยเจริญกรุง28 ถนนเจริญกรุง เขตบางรัก กท.

676 ตรอกฮ่องกงแบงค์ ถ.เจริญกรุง แขวงบางรัก เขตบางรัก กท

127 ซ.สว่าง 6 ถ.มหานคร แขวงมหาพฤฒาราม บางรัก 10500

ข้อมูล “ที่อยู่” ของฐานข้อมูลการใช้ไฟฟ้า ที่นำมาใช้ในการสร้างรหัสเพื่อเชื่อมโยงฐานข้อมูลนั้นก็คือ ข้อมูล บ้านเลขที่ หมู่ ซอย ถนน และตำบล ส่วนข้อมูลอำเภอ และจังหวัดนั้น ใช้ข้อมูลจากฟิลด์รหัสเขต โดยนำรหัสเขต ไปเปรียบเทียบในตาราง LUT ของเขต และ LUT ของจังหวัด

2.2.2 โครงสร้างฐานข้อมูลประกันสังคม (ข้อมูลจากสำนักงานประกันสังคม) มีการเก็บข้อมูลของรหัสต่างๆ ที่ใช้ในสำนักงานประกันสังคม ชื่อสถานประกอบการ ที่อยู่ของสถานประกอบการ และข้อมูลเกี่ยวกับเจ้าของกิจการ

ตารางที่ 2.3 โครงสร้างฐานข้อมูลประกันสังคม ของสำนักงานประกันสังคม

Column no.	Length	Description	remark
1	14	รหัสประกันสังคม	
2	50	ชื่อสถานประกอบการ	
3	62	เลขที่ ซอย ถนน แขวง	
4	4	รหัสเขต	
5	2	รหัสจังหวัด	
6	5	รหัสไปรษณีย์	
7	10	หมายเลขโทรศัพท์	
8	50	ชื่อเจ้าของกิจการ	
9	7	วันที่จดทะเบียนสมาชิก	(mm dd yy)

โครงสร้างของข้อมูล “ที่อยู่” ที่นำมาใช้ในการสร้างรหัสเพื่อเชื่อมโยงฐานข้อมูลคือ ฟิลด์เลขที่ ซอย ถนน แขวง ฟิลด์รหัสเขต และฟิลด์รหัสจังหวัด ดังที่ปรากฏในตารางที่ 2.3 โดยลักษณะการจัดเก็บข้อมูล “ที่อยู่” ของสำนักงานประกันสังคม มีลักษณะในการบันทึกข้อมูล ซึ่งมีรายละเอียดของข้อมูล “ที่อยู่” รวมอยู่ในฟิลด์ เลขที่ ซอย ถนน แขวง ฟิลด์รหัสเขตและฟิลด์รหัสจังหวัด

ข้อมูล “ที่อยู่” ของการประกันสังคม ในส่วนข้อมูลจากฟิลด์รหัสเขต จะนำรหัสเขต ไปเปรียบเทียบในตาราง LUT ของเขต และข้อมูลจังหวัดนั้นจะใช้ข้อมูลจากฟิลด์รหัสจังหวัด ไปเปรียบเทียบในตาราง LUT ของจังหวัด เพื่อให้ได้ชื่อของเขต และจังหวัด

2.2.3 โครงสร้างฐานข้อมูลอาคาร (ข้อมูลจากกรมสรรพากร) มีการเก็บข้อมูลของรหัสต่างๆ เกี่ยวกับ Block อาคาร เป็นข้อมูลที่ได้จากการดำเนินการสำรวจเก็บข้อมูลปฐมภูมิภาคสนามเพื่อให้ได้มาซึ่งรายละเอียดข้อมูลผู้ประกอบการ ซึ่งเป็นแหล่งภาษีอากร และเก็บข้อมูลตำแหน่งที่ตั้งจากพื้นที่จริง โดยฐานข้อมูลนี้เป็นฐานข้อมูลที่เชื่อมเข้ากับข้อมูลกราฟิก Block อาคาร ในระบบ GIS

ตารางที่ 2.4 โครงสร้างฐานข้อมูลอาคาร ของกรมสรรพากร

Field	Data Type	Description	Remark
FIELD_ID	Char(10)	รหัสสำรวจ	FK
TIN	Char(15)	เลขประจำตัวผู้เสียภาษี	FK
TAX_PAYER	Char(50)	ชื่อผู้เสียภาษี	
REG_NAME	Char(50)	ชื่อสถานประกอบการ	
BLDG_NAME	Char(50)	ชื่ออาคาร	
FLOOR	Char(20)	ชั้น	
ADDR_NO	Char(50)	บ้านเลขที่	
MOO	Char(20)	เลขที่หมู่บ้าน	
SOI	Char(30)	ชื่อซอย	
ROAD	Char(30)	ชื่อถนน	
KHWENG	Char(50)	ชื่อแขวง	
ZIP_CODE	Char(6)	รหัสไปรษณีย์	
TEL	Char(20)	หมายเลขโทรศัพท์	
SITE_TYPE	Char(5)	ลักษณะสถานที่	
HOUSES	Integer	จำนวนคูหา	
STORIES	Single	จำนวนชั้น	
BUSINESS	Char(5)	รหัส ISIC	
LOCATION	Char(30)	ทำเลที่ตั้ง	
DECORATION	Char(30)	สภาพของสถานที่	
DENSITY	Char(30)	ความหนาแน่น	
REMARKS	Char(100)	หมายเหตุ	
HOUSE_ID	Char(15)	รหัสประจำบ้าน	FK
ELECTRICITY_ID	Char(2)	รหัสไฟฟ้า	FK
ELECTRICITY_NO	Char(5)	รหัสไฟฟ้า	FK
METER_NO	Char(20)	หมายเลขของ Meter	FK
TIN_INSUR	Char(15)	รหัสประกันสังคม	FK
MAPINFO_ID	Integer	รหัสของ MAPINFO	FK

โครงสร้างของข้อมูล “ที่อยู่” ที่นำมาใช้ในการสร้างรหัสเพื่อเชื่อมโยงฐานข้อมูลคือ ฟิลด์บ้านเลขที่ ฟิลด์เลขที่หมู่บ้าน ฟิลด์ชื่อซอย ฟิลด์ชื่อถนน ฟิลด์ชื่อแขวง ดังที่ปรากฏในตารางที่ 2.4 โดยลักษณะการจัดเก็บข้อมูล “ที่อยู่” ของข้อมูลอาคาร มีลักษณะในการบันทึกข้อมูล ซึ่งมีรายละเอียดของข้อมูล “ที่อยู่” แยกเป็นฟิลด์ เลขที่ หมู่ ซอย ถนน แขวง โดยในส่วนของข้อมูล เขต และจังหวัด จะนำชื่อแขวง ไปเปรียบเทียบในตาราง LUT ของแขวงได้รหัสเขตมาจากนั้นนำไปเปรียบเทียบในตาราง LUT ของเขต เพื่อให้ได้ชื่อเขต และข้อมูลจังหวัดนั้น ใช้ข้อมูลจากชื่อเขต ไปเปรียบเทียบในตาราง LUT ของเขต แล้วไปเปรียบเทียบในตาราง LUT ของจังหวัด เพื่อให้ได้ชื่อจังหวัด

2.2.4 โครงสร้างฐานข้อมูลเลขประจำตัวผู้เสียภาษี (ข้อมูลจากกรมสรรพากร) มีการเก็บข้อมูลของรหัสประจำตัวผู้เสียภาษี ชื่อ และที่อยู่ของสถานประกอบการ

โครงสร้างของข้อมูล “ที่อยู่” ที่นำมาใช้ในการสร้างรหัสเพื่อเชื่อมโยงฐานข้อมูลคือ ฟิลด์เลขที่บ้าน ฟิลด์ซอย/สถานที่ทำงาน ฟิลด์หมู่ ฟิลด์ถนน ฟิลด์ตำบล ฟิลด์อำเภอ และฟิลด์รหัสจังหวัด ดังที่ปรากฏในตารางที่ 2.5 โดยลักษณะการจัดเก็บข้อมูล “ที่อยู่” ของฐานข้อมูลเลขประจำตัวผู้เสียภาษี ซึ่งลักษณะในการบันทึกข้อมูลมีรายละเอียดแยกเป็นฟิลด์บ้านเลขที่ ซอย หมู่ ถนน ตำบล อำเภอ และในส่วนข้อมูลจังหวัดนั้นจะใช้ข้อมูลจากฟิลด์รหัสจังหวัด ไปเปรียบเทียบในตาราง LUT ของจังหวัด เพื่อให้ได้ชื่อจังหวัด

ตารางที่ 2.5 โครงสร้างฐานข้อมูลเลขประจำตัวผู้เสียภาษี ของกรมสรรพากร

Field	หลักที่	จำนวนหลัก	Remark
เลขประจำตัวผู้เสียภาษี	1-10	10	
คำนำหน้าชื่อ	11-35	25	
ชื่อ	36-65	30	
นามสกุล	66-95	30	
เลขที่บ้าน	96-115	20	
ซอย/สถานที่ทำงาน	116-155	20	
หมู่	156-157	2	
ถนน	158-187	30	
ตำบล	188-217	30	
อำเภอ	218-242	25	
จังหวัด	243-267	25	
รหัสไปรษณีย์	268-272	5	
วันเดือนปีเกิด	273-278	6	
เลขประจำตัวประชาชน	279-291	13	
blank	292	1	

2.3 สรุปลักษณะโครงสร้างข้อมูล “ที่อยู่”

ลักษณะโครงสร้างของข้อมูล “ที่อยู่” ของหน่วยงานต่างๆ ที่ได้ศึกษามาแล้วนี้ สามารถสรุปรูปแบบการจัดเก็บข้อมูลออกเป็น 2 ลักษณะ คือ

2.3.1 แบบที่อยู่รวมกันภายในฟิลด์เดียว

มีลักษณะการเก็บข้อมูล “ที่อยู่” ทั้งหมดรวมอยู่ในฟิลด์เดียวกันโดยจะมีทั้งข้อมูลบ้านเลขที่ หมู่ ซอย ถนน ตำบล อำเภอ และจังหวัด ซึ่งการจัดเก็บข้อมูลเช่นนี้ พบในฐานข้อมูลของการใช้ไฟฟ้า และฐานข้อมูลการประกันสังคม

2.3.2 แบบที่ถูกแยกเป็นฟิลด์ไว้แล้ว

มีลักษณะการเก็บข้อมูล“ที่อยู่” โดยแยกเป็นฟิลด์ต่างๆ ซึ่งแต่ละหน่วยงานอาจมีความละเอียดในการแบ่งฟิลด์ไม่เท่ากันสำหรับข้อมูลอาคาร และฐานข้อมูลเลขประจำตัวผู้เสียภาษีของกรมสรรพากรได้แบ่งออกเป็น ข้อมูลบ้านเลขที่ หมู่ ซอย ถนน ตำบล อำเภอ และจังหวัด

2.4 พฤติกรรมการป้อนข้อมูล “ที่อยู่”

การนำเข้าฐานข้อมูล “ที่อยู่” ของหน่วยงานต่างๆ แต่เดิมมีการพัฒนาฐานข้อมูลบนโปรแกรม dBase Foxpro Excel และ Access เป็นหลัก และยังคงมีการใช้อยู่ ถึงแม้จะมีการย้ายฐานข้อมูลขึ้นสู่ระบบใหม่ ก็ยังคงมีลักษณะการจัดเก็บเช่นเดิมไม่ได้เปลี่ยนโครงสร้างรูปแบบการจัดเก็บข้อมูล “ที่อยู่” โดยใช้วิธีการถ่ายโอนฐานข้อมูลโดยตรง จากการศึกษาพบว่า การป้อนข้อมูล “ที่อยู่” มีปัญหาเรื่องลำดับการพิมพ์ก่อนหลังของ พยัญชนะ สระ และวรรณยุกต์ โดยตัวอักษรที่ถ่ายโอนฐานข้อมูลโดยตรงจะมีแถบดำไม่สามารถอ่านข้อมูลได้ พอสรุปได้ดังนี้

2.4.1 พฤติกรรมการป้อนตัวอักษรภาษาไทย

ตัวอย่างคำ	dBase และ Foxpro	Excel และ Access
นุ่น	พยัญชนะ – สระ – วรรณยุกต์	พยัญชนะ – สระ – วรรณยุกต์ พยัญชนะ– วรรณยุกต์ – สระ
ตั้ง	พยัญชนะ – สระ – วรรณยุกต์	พยัญชนะ – สระ – วรรณยุกต์ พยัญชนะ– วรรณยุกต์ – สระ

น้ำ	พยัญชนะ– วรรณยุกต์ – สระ	พยัญชนะ – สระ – วรรณยุกต์ พยัญชนะ– วรรณยุกต์ – สระ
เตี้ย	สระ –พยัญชนะ – สระ – วรรณยุกต์	สระ – พยัญชนะ – สระ – วรรณยุกต์ สระ – พยัญชนะ– วรรณยุกต์ – สระ

จะเห็นได้ว่าโปรแกรม dBase และ Foxpro จะมีข้อจำกัดมากกว่าโปรแกรม Excel และ Access โดยโปรแกรม dBase และ Foxpro จะต้องพิมพ์ตามลำดับดังตัวอย่างข้างต้นเท่านั้น เมื่อพิมพ์ไม่ถูกต้องตามลำดับจะเกิดแถบดำขึ้นแทนที่จะเป็นวรรณยุกต์หรือสระ ส่วนในโปรแกรม Excel และ Access สามารถพิมพ์ข้อมูลได้โดยไม่ต้องคำนึงถึงลำดับก่อนหลัง ซึ่งมีความยืดหยุ่นในการพิมพ์ข้อมูลมากกว่าโปรแกรม dBase และ Foxpro

2.4.2 รูปแบบการจัดเก็บข้อมูล “ที่อยู่”

1. เรียงลำดับความสำคัญของการนำเข้าข้อมูล ก่อน-หลัง โดยเรียงจากซ้ายไปขวา บ้านเลขที่ – หมู่ – ซอย – ถนน – ตำบลหรือแขวง – อำเภอหรือเขต – จังหวัด

ตัวอย่างเช่น

50/1 ม.1 ซ.สุขุมวิท59 ถ.สุขุมวิท แขวงคลองตันเหนือ เขตคลองเตย จ.กรุงเทพฯ

2. โดยทั่วไปข้อมูลบ้านเลขที่จะเป็นข้อมูลที่ผู้บันทึกข้อมูลนำเข้าเป็นอันดับแรก ถ้าตัวอักษรตัวแรกไม่เป็นตัวเลข แสดงว่าข้อมูลนั้นไม่มีบ้านเลขที่

ตัวอย่างเช่น

ข้างร้านเอราวัณ ถ.สุรวงศ์ แขวงสุริวงศ์ เขตบางรัก กท.

3. หมู่เป็นข้อมูลที่ผู้นำเข้าข้อมูลจะใช้ “ม” “ม.” หรือ “หมู่” นำหน้าเลขหมู่

ตัวอย่างเช่น

50/1 *ม.1* ซ.สุขุมวิท59… หรือ

50/1 *ม1* ซ.สุขุมวิท59… หรือ

50/1 *หมู่1* ซ.สุขุมวิท59…

4. ตัวเลขที่ตามหลัง “ม” “ม.” หรือ “หมู่” โดยมีช่องว่างเป็นตัวคั่นหรือไม่มีก็ตาม คือ เลขหมู่

ตัวอย่างเช่น

50/1 *ม. 1* ซ.สุขุมวิท59… หรือ

50/1 *ม.1* ซ.สุขุมวิท59…

5. ซอยเป็นข้อมูลที่ผู้นำเข้าข้อมูลจะใช้ “ซ” “ซ.” “ซอย” หรือ “ตรอก”นำหน้าชื่อซอย โดยไม่มีช่องว่างเป็นตัวคั่น

ตัวอย่างเช่น

50/1 ม.1 *ซ.สุขุมวิท59*… หรือ

50/1 ม.1 *ซสุขุมวิท59*… หรือ

50/1 ม.1 *ซอยสุขุมวิท59*… หรือ

919/423ชั้น35 *ตรอก*เวท ถ.สีลม …

6. ตัวเลขที่ตามหลัง ชื่อซอย โดยมีช่องว่างเป็นตัวคั่นหรือไม่มีก็ตาม คือ เลขที่ซอย

ตัวอย่างเช่น

50/1 ม.1 ซ.*สุขุมวิท 59*… หรือ

50/1 ม.1 ซ.*สุขุมวิท59*…

7. ถนนเป็นข้อมูลที่ผู้นำเข้าข้อมูลจะใช้ “ถ.” หรือ “ถนน” นำหน้าชื่อถนน (บางครั้งก็มีแต่ชื่ออย่างเดียว)

ตัวอย่างเช่น

50/1 ซ.สุขุมวิท59 *ถ.*สุขุมวิท… หรือ

50/1 ซ.สุขุมวิท59 *ถนน*สุขุมวิท… หรือ

50/1 ซ.สุขุมวิท59 *สุขุมวิท*…

8. ตำบลเป็นข้อมูลที่ผู้นำเข้าข้อมูลจะใช้ “ต.” “ตำบล” หรือ แขวง นำหน้าชื่อตำบล (บางครั้งมีแต่ชื่ออย่างเดียว)

ตัวอย่างเช่น

…ถ.สุขุมวิท *แขวง*คลองตันเหนือ…หรือ

…ถ.สุขุมวิท *ต.*คลองตันเหนือ… หรือ

…ถ.สุขุมวิท *ตำบล*คลองตันเหนือ… หรือ

…ถ.สุขุมวิท *คลองตันเหนือ*…

9. อำเภอเป็นข้อมูลที่ผู้นำเข้าข้อมูลจะใช้ “อ.” “อำเภอ” หรือ เขต นำหน้าชื่ออำเภอ (บางครั้งมีแต่ชื่ออย่างเดียว)

ตัวอย่างเช่น

…แขวงคลองตันเหนือ *เขต*คลองเตย จ.กรุงเทพฯ หรือ

…แขวงคลองตันเหนือ *อ.*คลองเตย จ.กรุงเทพฯ หรือ

…แขวงคลองตันเหนือ *อำเภอ*คลองเตย จ.กรุงเทพฯ หรือ

…แขวงคลองตันเหนือ *คลองเตย* จ.กรุงเทพฯ

10. จังหวัดเป็นข้อมูลที่ผู้นำเข้าข้อมูลจะใช้ จ. หรือ จังหวัด นำหน้าชื่อจังหวัด (บางครั้งมีแต่ชื่ออย่างเดียวและบางหน่วยงานใช้เป็นรหัสจังหวัด)

ตัวอย่างเช่น

…แขวงคลองตันเหนือ อ.คลองเตย จ.กรุงเทพฯ หรือ

…แขวงคลองตันเหนือ อำเภอคลองเตย *จังหวัด*กรุงเทพฯ หรือ

…แขวงคลองตันเหนือ อำเภอคลองเตย *กรุงเทพฯ*

11. บางเรคคอร์ดจะมีการอธิบายตำแหน่งที่ตั้งโดยเทียบกับสถานที่ ที่เป็นจุดเด่น

ตัวอย่างเช่น

1210 ใกล้ตรอกวัดม่วงแค ถ.เจริญกรุง

2.5 ประเด็นปัญหาสำคัญต่างๆ ที่พบของข้อมูล “ที่อยู่”

ในแต่ละหน่วยงานจะมีการเก็บข้อมูล “ที่อยู่” ที่แตกต่างกันทำให้เกิดปัญหาต่างๆ ขึ้น ในการศึกษาครั้งนี้ทำให้พบปัญหาสำคัญ ดังจะกล่าวต่อไปนี้

2.5.1 ปัญหาเรื่องของโครงสร้าง

1. การเก็บข้อมูลในแต่ละหน่วยงานมีการเก็บรายละเอียดไม่เท่ากัน

ตัวอย่างเช่น

ในฐานข้อมูลการประกันสังคมไม่มีการเก็บฟิลด์หมู่

2. ข้อมูลในแต่ละหน่วยงานมีโครงสร้างในการจัดเก็บแตกต่างกัน

ตัวอย่างเช่น

ในโครงสร้างฐานข้อมูลการใช้ไฟฟ้ามีการเก็บข้อมูล “ที่อยู่” ในฟิลด์เดียว แต่โครงสร้างฐานข้อมูลเลขประจำตัวผู้เสียภาษี มีการเก็บข้อมูล “ที่อยู่” แยกเป็นหลายฟิลด์

2.5.2 ปัญหาเรื่องการเลือกใช้อักษรย่อ

1. การพิมพ์คำย่อของคำว่า “อาคาร” จะใช้ตัวย่อเป็น อ. ซึ่งจะซ้ำกับตัวย่อของคำว่า “อำเภอ” ทำให้เวลาจำแนกข้อมูลที่อยู่จะแยกชื่ออาคารไปเก็บไว้ในฟิลด์อำเภอ

ตัวอย่างเช่น

0352/3 *อ.*สีลมคอนโดมิเนียมชั้น 3 ซ.ศาลาแดง2…

2. การพิมพ์คำย่อของคำว่า “ตรอก” จะใช้ตัวย่อเป็น ต. ซึ่งจะซ้ำกับตัวย่อของคำว่า “ตำบล” ทำให้เวลาจำแนกข้อมูลที่อยู่จะแยกชื่อตรอกไปเก็บไว้ในฟิลด์ตำบล ทำให้เกิดการจัดเก็บผิดฟิลด์

ตัวอย่างเช่น

330/1 ต.ต้นมะขาม ถ.เจริญกรุง 10500

2.5.3 ปัญหาเรื่องการนำเข้าข้อมูล

1. กรอกข้อมูลไม่ครบถ้วน ทำให้เกิดปัญหาในการให้รหัสเพื่อเชื่อมโยงฐานข้อมูลเป็นผลให้การเชื่อมโยงข้อมูลระหว่างหน่วยงานมีการเชื่อมโยงกันได้น้อย

ตัวอย่างเช่น

บริเวณโรงเรียนอัสสัมชัญ บางรัก กท.

2. ผู้นำเข้าไม่ได้เว้นวรรคขั้นระหว่างข้อมูล ทำให้ไม่สามารถแยกข้อมูลออกจากกันได้อย่างถูกต้อง

ตัวอย่างเช่น

942/173ชั้น26ตรงข้ามสถานเสาวภา ถ.พระราม 4 สุริยวงศ์

3. ไม่ใส่ตัวนำหน้าระบุชนิดของข้อมูล ทำให้ไม่สามารถจัดข้อมูลลงในฟิลด์ได้ถูกต้อง

ตัวอย่างเช่น

กรอกแต่ชื่อถนน แต่ไม่ใส่ ถ. นำหน้า

- ตรอกโอเรียนเต็ล เจริญกรุง

4. พิมพ์ข้อมูลผิด ทำให้มีปัญหาในการเข้ารหัสเนื่องจากไม่มีข้อมูลใน LUT

ตัวอย่างเช่น

201/5 ซ.จจอมสมบูรณ์ พระราม4 บางรัก

2.5.4 ปัญหาเรื่องการแยกข้อมูล “ที่อยู่”

เมื่อทำการแยกข้อมูลที่อยู่แล้วจะมีข้อมูลบางส่วนที่ไม่สามารถจำแนกได้ ว่าควรจะเก็บข้อมูลในฟิลด์ใด ซึ่งเป็นปัญหาต่อเนื่องมาจากข้างต้นในเรื่องปัญหาการเลือกใช้อักษรย่อ และปัญหาเรื่องการนำเข้าข้อมูล ทำให้ข้อมูลบางส่วนที่ไม่สามารถจำแนกได้เกิดการสูญหาย

ตัวอย่างเช่น

38/15 *ใกล้ภัตตาคารประตูทอง* ถนนสุรวงศ์ ข้อมูล “ใกล้ภัตตาคารประตูทอง” จะหายไปไม่สามารถจำแนกใส่ฟิลด์ใดได้

2.5.5 ปัญหาเรื่องความทันสมัยของข้อมูล

ข้อมูลที่มีอยู่ในแต่ละหน่วยงานมีการเก็บอยู่ในช่วงเวลาที่แตกต่างกัน เมื่อมีการแบ่งเขตการปกครองใหม่ ในแต่ละหน่วยงานก็ไม่ได้ทำการปรับให้ข้อมูลมีความทันสมัยขึ้น จึงทำให้ข้อมูลมีความแตกต่างกันในเรื่องเขตการปกครอง

2.6 แนวทางการแก้ปัญหา

จากปัญหาดังที่กล่าวมาแล้วนั้น สามารถแก้ไขได้ 2 แนวทางหลัก คือ

2.6.1 การกำหนดข้อตกลงร่วมในการจัดเก็บข้อมูลที่อยู่ สำหรับการนำเข้าข้อมูลชุดใหม่ต่อไป ได้แก่

2.6.1.1 การกำหนดโครงสร้างข้อมูล “ที่อยู่” ใหม่

ก่อนทำการเก็บข้อมูลครั้งต่อไปหน่วยงานต่างๆ ที่จะใช้ข้อมูลร่วมกันควรทำความตกลงในเรื่องความละเอียดของโครงสร้างข้อมูล “ที่อยู่” ว่าจะเก็บข้อมูลอะไรบ้างเพื่อความเป็นอันหนึ่งอันเดียวกันของข้อมูล การเชื่อมโยงฐานข้อมูลระหว่างหน่วยงานเข้าด้วยกันจึงจะมีความถูกต้องมากยิ่งขึ้น

ตัวอย่างเช่น

ใช้โครงสร้างข้อมูลกลาง

2.6.1.2 การกำหนดการใช้อักษรย่อต่างๆ

โดยอาจกำหนดข้อตกลงในการเลือกใช้อักษรย่อ

ตัวอย่างเช่น

1. กำหนดให้ อ. เป็นอักษรย่อของคำว่า “อำเภอ” เท่านั้น ส่วนอาคารให้พิมพ์คำว่า “อาคาร” แล้วตามด้วยชื่ออาคาร
2. กำหนดให้ ต. เป็นอักษรย่อของคำว่า “ตำบล” เท่านั้น ส่วนตรอกให้พิมพ์คำว่า “ตรอก” แล้วตามด้วยชื่อตรอก

2.6.1.3 การกำหนดการนำเข้าข้อมูล “ที่อยู่”

1. กรอกข้อมูลให้ครบโดยมีบ้านเลขที่ หมู่ ซอย ถนน ตำบล อำเภอ และจังหวัด การทำวิจัยครั้งนี้จะทำฟังก์ชันที่ช่วยในการกรอกข้อมูลให้สมบูรณ์ยิ่งขึ้น

2. ให้ผู้นำเข้าเว้นวรรคขั้นระหว่างข้อมูลต่างที่กล่าวมาข้างต้น

ตัวอย่างเช่น

544/1 ซ.แก้วฟ้า ถ.สี่พระยา

3. ใส่ตัวนำหน้าระบุชนิดของข้อมูล เพื่อที่จะได้จำแนกข้อมูลได้อย่างถูกต้อง

ตัวอย่างเช่น

กรอกชื่อถนน โดยใส่ ถ. นำหน้า

4. ในเรื่องการพิมพ์ข้อมูลผิด ในการทำวิจัยครั้งนี้จะทำฟังก์ชันที่ช่วยในการตรวจสอบและแก้ไขคำผิด ให้ถูกต้องด้วย

2.6.1.4 การปรับปรุงข้อมูลให้เป็นปัจจุบัน

ปัญหาเรื่องความทันสมัยของข้อมูลนั้นเป็นเรื่องที่ยุ่งยากมากในการจัดการกับข้อมูลที่ไม่ได้ทำการปรับปรุงมานาน จึงขอเสนอให้จัดเจ้าหน้าที่ โดยให้มีหน้าที่ในการทำข้อมูลให้เป็นปัจจุบัน

2.6.2 การพัฒนาโปรแกรมช่วยในการจัดการข้อมูลเดิมที่มีอยู่ ในฐานข้อมูล ซึ่งรวมทั้งการตรวจสอบและแก้ไขคำผิดให้ถูกต้องเบื้องต้น ซึ่งจะกล่าวในบทที่ 4

2.6.2.1 แนวทางการแก้ปัญหาเรื่องลำดับการป้อนอักษรไทย

ในการทำวิจัยครั้งนี้จะไม่สนใจลำดับก่อนหลังของการนำเข้า เพราะจะมีโปรแกรมทำการตรวจสอบว่าคำๆ เดียวกัน มีลำดับการนำเข้าเหมือนกันหรือไม่ ถ้าไม่เหมือนกันก็จะทำการแยกออกมาเพื่อทำการแก้ไขให้ถูกต้องเป็นแบบเดียวกันโดยใช้ฟังก์ชันแก้คำผิดกึ่งอัตโนมัติ โดยการแทนที่คำที่มีลำดับการนำเข้าที่ไม่เหมือนกันทั้งหมดด้วยคำที่มีลำดับการนำเข้าที่ถูกต้อง

2.6.2.2 แนวทางการแก้ปัญหาเรื่องการแยกข้อมูล “ที่อยู่”

ข้อมูลที่ไม่สามารถจัดลงตารางที่ออกแบบไว้ได้จะนำไปเก็บไว้ที่ฟิลด์หมายเหตุ (REMARK)

บทที่ 3

การจัดโครงสร้างข้อมูล “ที่อยู่”

การจัดโครงสร้างข้อมูล “ที่อยู่” มีวัตถุประสงค์ เพื่อแยกข้อมูลต่างๆ ที่เกี่ยวข้องกับการระบุตำแหน่งที่อยู่ (address) ให้มีความชัดเจน และง่ายต่อการชี้เฉพาะในการเชื่อมโยงข้อมูล ดังนั้น จึงต้องมีการกำหนดโครงสร้างกลางขึ้น เพื่อใช้ในการแก้ปัญหาดังกล่าว ในที่นี้จะเรียกว่า “โครงสร้างข้อมูลกลาง”

3.1 การออกแบบโครงสร้างข้อมูลกลาง

โครงสร้างข้อมูลกลาง จะเป็นโครงสร้างที่จัดเก็บส่วนประกอบต่างๆ ของข้อมูล “ที่อยู่” โดยจะจัดเก็บเป็นชื่อแทนที่จะเป็นรหัสเพื่อแก้ปัญหาในการที่แต่ละหน่วยงานมีการใช้รหัสในการจัดเก็บต่างกัน และนำข้อมูลเหล่านี้ไปเทียบรหัสกับ LUT ต่างๆ ในภายหลัง เพื่อสร้างฟิลด์ที่ใช้สำหรับเป็นตัวเชื่อมระหว่างฐานข้อมูล จากการศึกษาการจัดเก็บข้อมูล “ที่อยู่” ของ การไฟฟ้านครหลวง สำนักงานประกันสังคม กรมสรรพากร และโครงสร้างในระบบทะเบียนราษฎร์ ของกรมการปกครอง พบว่าโครงสร้างข้อมูลกลางควรประกอบด้วย 8 ฟิลด์ คือ บ้านเลขที่ – หมู่ – ชื่อซอย – ชื่อถนน – ชื่อตำบลหรือแขวง – ชื่ออำเภอหรือเขต – ชื่อจังหวัด – หมายเหตุ โดยมีแนวคิดดังนี้

1. ข้อมูลจังหวัดเป็นข้อมูลที่ใหญ่ที่สุด โดยมีส่วนที่ย่อยลงคือ อำเภอหรือเขต ซึ่งชื่ออำเภอหรือเขตอาจซ้ำกันได้ จึงต้องมีฟิลด์จังหวัด ช่วยในการจำแนกชื่ออำเภอหรือเขต ที่ซ้ำกัน
2. ข้อมูลอำเภอหรือเขต เป็นข้อมูลที่ย่อยลงมาจากจังหวัด โดยมีส่วนที่ย่อยลงคือ ตำบลหรือแขวง ซึ่งชื่อตำบลหรือแขวงอาจซ้ำกันได้จึงต้องมีฟิลด์อำเภอ ช่วยในการจำแนกชื่อตำบลหรือแขวงซ้ำกัน
3. ข้อมูลตำบลหรือแขวงเป็นข้อมูลที่ย่อยลงมาจากอำเภอ ซึ่งชื่อตำบลหรือแขวงทั่วประเทศนั้นอาจซ้ำกันได้ จึงต้องมีฟิลด์อำเภอและจังหวัดช่วยในการจำแนกชื่อตำบลหรือแขวงซ้ำกัน
4. ข้อมูลถนน 1 สาย อาจผ่านหลายตำบล โดยมีข้อมูลย่อยลงมาคือ ซอย ซึ่งชื่อถนนทั่วประเทศนั้นอาจซ้ำกันได้ จึงต้องมีฟิลด์ตำบลหรือแขวง และ ฟิลด์ซอยช่วยในการจำแนกชื่อถนนที่ซ้ำกัน

5. ข้อมูลซอยเป็นหน่วยย่อยของถนนและหมู่ ซึ่งชื่อซอยทั่วประเทศอาจซ้ำกันได้ จึงต้องมีฟิลด์อื่นมาช่วยในการจำแนกชื่อที่ซ้ำกัน
6. ข้อมูลหมู่ เป็นข้อมูลที่บางหน่วยงานไม่ได้ทำการจัดเก็บ เป็นหน่วยย่อยของตำบล มีหน่วยย่อยลงไปอีกคือ บ้านเลขที่
7. ข้อมูลบ้านเลขที่เป็นหน่วยย่อยที่เล็กที่สุด มีข้อมูลซ้ำกันจึงต้องมีฟิลด์อื่นมาช่วยในการจำแนกข้อมูลที่ซ้ำกัน
8. ฟิลด์หมายเหตุเป็นฟิลด์ที่เก็บข้อมูลที่ไม่สามารถแยกได้ เพื่อให้ผู้ใช้ทำการแก้ไขโดยนำข้อมูลในช่องหมายเหตุ ไปใส่ในฟิลด์ที่ข้อมูลนั้นควรจะอยู่ โดยข้อมูลที่ไม่สามารถแยกได้นี้จะถูกคั่นด้วยเครื่องหมาย “,”

โครงสร้างฐานข้อมูลกลาง ได้ทำการออกแบบดังรายละเอียดในตารางที่ 3.1

ตารางที่ 3.1 โครงสร้างฐานข้อมูลกลาง

ชื่อ Field	Data Type	Description	Remark
NADD_NO	Char(20)	บ้านเลขที่	
NMOO	Numeric(2)	หมู่	
NSOI	Char(40)	ชื่อซอย	
NROAD	Char(50)	ชื่อถนน	
NTAMBON	Char(30)	ชื่อตำบลหรือแขวง	
NAMPHOE	Char(25)	ชื่ออำเภอหรือเขต	
NPROVINCE	Char(25)	ชื่อจังหวัด	
REMARK	Char(100)	หมายเหตุ	

3.2 การออกแบบกรรมวิธีการเชื่อมโยงข้อมูล

รูปที่ 3.1 แสดงขั้นตอนกรรมวิธีการเชื่อมโยงข้อมูล โดยเริ่มต้นจาก หน่วยงานต่างๆ ที่จะนำฐานข้อมูลมาเชื่อมโยงกันนั้น จะมีข้อมูล “ที่อยู่” ประกอบอยู่ด้วยในทุกๆ ฐานข้อมูล แต่จะอยู่ในรูปแบบโครงสร้างที่แตกต่างกันจึงทำให้ไม่สามารถเชื่อมฐานข้อมูลได้ในทันที โดยจะต้องทำวิธีการดังนี้

(1) จะต้องจัดโครงสร้าง “ที่อยู่” ใหม่ก่อน(ใช้ฟังก์ชันแยก “ที่อยู่”)

(2) ทำการแก้ไขข้อมูลให้ถูกต้อง (ใช้ฟังก์ชันแก้คำผิด)

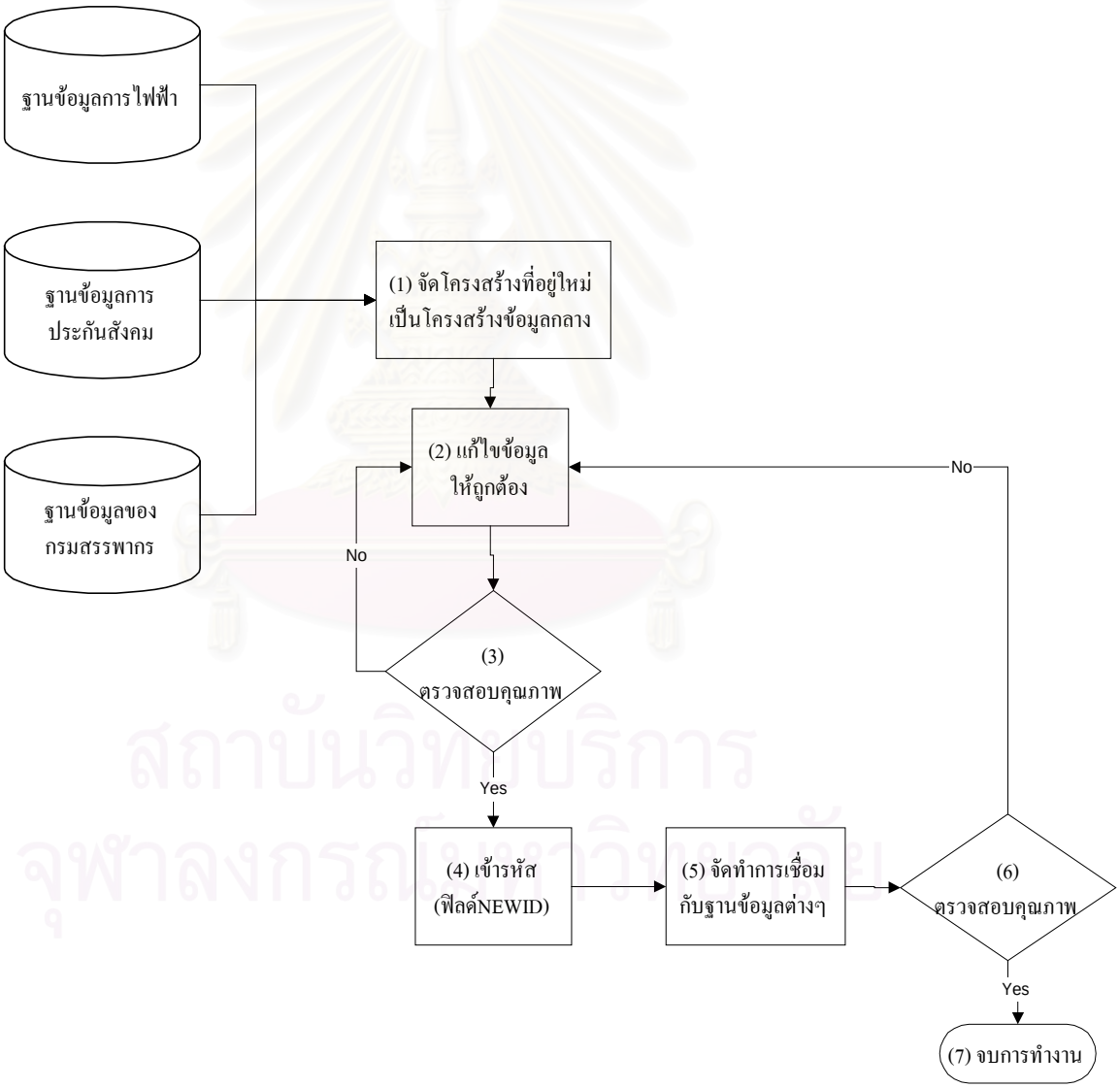
(3) เมื่อแก้ไขเรียบร้อยแล้วทำการตรวจสอบว่ามีที่ต้องแก้ไขอีกหรือไม่

(4) ถ้ามีก็ให้ทำการแก้ไขต่อไป แต่ถ้าไม่มีแล้วก็นำข้อมูลไปเข้ารหัส (ใช้ฟังก์ชันเข้ารหัส) โดยจะนำข้อมูลไปเปรียบเทียบกับตาราง LUT เพื่อเทียบเป็นรหัสในการสร้างฟิลด์ NEWID เมื่อสร้างฟิลด์ NEWID เสร็จแล้ว

(5) ให้นำฟิลด์ NEWID ในแต่ละหน่วยงานมาเป็นตัวเชื่อมฐานข้อมูล

(6) ทำการตรวจสอบผลการเชื่อมโยงฐานข้อมูลว่าถูกต้องหรือไม่ ถ้าไม่ถูกต้องก็กลับไปแก้ไขข้อมูลให้ถูกต้องแล้วเข้ารหัสใหม่

(7) แต่ถ้าผลการเชื่อมโยงฐานข้อมูลให้ความถูกต้องในระดับที่น่าพอใจแล้วก็ถือเป็นการสิ้นสุดการทำงาน



รูปที่ 3.1 แผนผังการออกแบบกรรมวิธีการเชื่อมโยงฐานข้อมูล

3.3 ขั้นตอนการจัดข้อมูล “ที่อยู่” ของหน่วยงานที่ศึกษา

ในขั้นตอนการจัดข้อมูล “ที่อยู่” เป็นวิธีการที่นำมาใช้ในการพัฒนาโปรแกรมในฟังก์ชันแยก “ที่อยู่” โดยมีวิธีการจัดข้อมูล “ที่อยู่” ของแต่ละหน่วยงานที่แตกต่างกัน ให้มาอยู่ในรูปแบบโครงสร้างข้อมูลกลาง โดยมีขั้นตอนในการจัดข้อมูล ในแต่ละหน่วยงานดังนี้

3.3.1 ฐานข้อมูลการใช้ไฟฟ้า ของการไฟฟ้านครหลวง

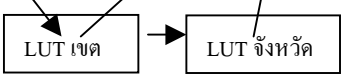
ลักษณะข้อมูล “ที่อยู่” จะเป็นที่อยู่ของผู้ขอจดทะเบียนมิเตอร์ไฟฟ้า (รูปที่ 3.2) ซึ่งที่อยู่ผู้ขอมิเตอร์ไฟฟ้า อาจจะมิใช่ “ที่อยู่” เดียวกับ “ที่อยู่” ที่มีการติดตั้งมิเตอร์ไฟฟ้านั้น ดังนั้นข้อมูลที่อยู่จึงมีบางส่วนที่มิได้มีความสัมพันธ์กับตำแหน่งของอาคาร

ฐานข้อมูลการใช้ไฟฟ้า

Column no.	Data Type	Description
1-2	Char(2)	ตอน (ตัวอักษร)
3-5	Numeric(3)	ตอน (ตัวเลข)
6-9	Numeric(4)	เขตเส้นทาง
10-15	Numeric(6)	เครื่องวัดฯ เลขที่
16-75	Char(60)	ชื่อผู้ใช้ไฟฟ้า
76-135	Char(60)	ที่อยู่ผู้ใช้ไฟฟ้า
136-137	Char(2)	PY,SP,GV,PL,ML,TL
138-139	Numeric(2)	ประเภทไฟ
140	Numeric(1)	อัตรา
141-145	Numeric(5)	รหัสผู้ใช้ไฟฟ้า/รหัสกิจการ
146-150	Numeric(5)	AMPERE
151-158	Numeric(8)	จำนวนหน่วยเฉลี่ย
159-168	Numeric(3)	จำนวนเงินเฉลี่ย
169-170	Numeric(2)	รหัสเขต

โครงสร้างข้อมูลกลาง

ชื่อ Field	Data Type	Description
NADD_NO	Char(20)	บ้านเลขที่
NMOO	Numeric(2)	หมู่
NSOI	Char(40)	ชื่อซอย
NROAD	Char(50)	ชื่อถนน
NTAMBON	Char(30)	ชื่อตำบล/แขวง
NAMPHOE	Char(25)	ชื่ออำเภอ/เขต
NPROVINCE	Char(25)	ชื่อจังหวัด
REMARK	Char(100)	หมายเหตุ



รูปที่ 3.2 แสดงการจัดโครงสร้าง “ที่อยู่” ใหม่จากโครงสร้างฐานข้อมูลการใช้ไฟฟ้า

ข้อมูลจะเป็นข้อมูล**ที่อยู่ผู้ใช้ไฟฟ้า** ในคอลัมน์ที่ 76-135 ดังแสดงในรูปที่ 3.2 โดยข้อมูลที่อยู่ในฟิลด์นี้ มีข้อมูลบ้านเลขที่ เลขหมู่ ซอย ถนน ตำบล อำเภอ จังหวัด การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยกเป็น ข้อมูลบ้านเลขที่ เก็บไว้ในฟิลด์ NADD_NO

ข้อมูลเลขหมู่ เก็บไว้ในฟิลด์ NMOO

ข้อมูลซอย เก็บไว้ในฟิลด์ NSOI

ข้อมูลถนน เก็บไว้ในฟิลด์ NROAD

ข้อมูลตำบล เก็บไว้ในฟิลด์ NTAMBON

ใช้ข้อมูล**รหัสเขต** ในคอลัมน์ที่ 169-170 ดังแสดงในรูปที่ 3.2 โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลเลขรหัสเขต ซึ่งข้อมูลที่จะนำไปเก็บไว้ในโครงสร้างฐานข้อมูลกลางจะต้องเป็นข้อมูลชื่อ ไม่ใช่รหัส ดังนั้นการจัดโครงสร้าง “ที่อยู่” ใหม่ จึงต้องนำเลขรหัสไปตรวจสอบกับข้อมูลใน LUT เขต (ฟิลด์ code_elec) เมื่อรหัสตรงกับชื่ออำเภอใด ก็จะนำชื่ออำเภอ(ฟิลด์ khed) ไปเก็บไว้ในฟิลด์ NAMPHOE จากนั้นก็จะนำรหัสจังหวัดที่อยู่ใน LUT เขต ไปตรวจสอบกับข้อมูลใน LUT จังหวัด (ฟิลด์ province_I) เมื่อรหัสตรงกับชื่อจังหวัดใด ก็จะนำชื่อจังหวัด(ฟิลด์ province_n) ไปเก็บไว้ในฟิลด์ NPROVINCE ดูการเชื่อมโยงได้จากรูปที่ 3.2

3.3.2 ฐานข้อมูลประกันสังคม ของสำนักงานประกันสังคม

ข้อมูล “ที่อยู่” จะเป็นที่อยู่บริษัทที่จดทะเบียน ซึ่งจะสัมพันธ์กับตำแหน่งของอาคาร

ฐานข้อมูลประกันสังคม

Column no.	Length	Description
1	14	รหัสประกันสังคม
2	50	ชื่อสถานประกอบการ
3	62	เลขที่ ซอย ถนน แขวง
4	4	รหัสเขต
5	2	รหัสจังหวัด
6	5	รหัสไปรษณีย์
7	10	หมายเลขโทรศัพท์
8	50	ชื่อเจ้าของกิจการ
9	6	วันที่จดทะเบียนสมาชิก

โครงสร้างข้อมูลกลาง

ชื่อ Field	Data Type	Description
NADD_NO	Char(20)	บ้านเลขที่
NMOO	Numeric(2)	หมู่
NSOI	Char(40)	ชื่อซอย
NROAD	Char(50)	ชื่อถนน
NTAMBON	Char(30)	ชื่อตำบล/แขวง
NAMPHOE	Char(25)	ชื่ออำเภอเขต
NPROVINCE	Char(25)	ชื่อจังหวัด
REMARK	Char(100)	หมายเหตุ

LUT เขต

LUT จังหวัด

รูปที่ 3.3 แสดงการจัดโครงสร้าง “ที่อยู่” ใหม่จากโครงสร้างฐานข้อมูลกรมประกันสังคม

ใช้ข้อมูล**เลขที่ ซอย ถนน แขวง** ในคอลัมน์ที่ 3 ดังแสดงในรูปที่ 3.3 โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลบ้านเลขที่ ซอย ถนน และแขวง การจัดโครงสร้างที่อยู่ใหม่จะแยก

ข้อมูลบ้านเลขที่ เก็บไว้ในฟิลด์ NADD_NO

ข้อมูลซอย เก็บไว้ในฟิลด์ NSOI

ข้อมูลถนน เก็บไว้ในฟิลด์ NROAD

ข้อมูลแขวง เก็บไว้ในฟิลด์ NTAMBON

ใช้ข้อมูล**รหัสเขต** ในคอลัมน์ที่ 4 ดังแสดงในรูปที่ 3.3 โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลเลขรหัสเขต การจัดโครงสร้าง “ที่อยู่” ใหม่จะนำเลขรหัสไปตรวจสอบกับข้อมูลใน LUT เขต (ฟิลด์ code_soc) เมื่อรหัสตรงกับชื่ออำเภอใด (ฟิลด์ khed) ก็จะนำชื่ออำเภอ ไปเก็บไว้ในฟิลด์ NAMPHOE

ใช้ข้อมูล**รหัสจังหวัด** ในคอลัมน์ที่ 5 ดังแสดงในรูปที่ 3.3 โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลเลขรหัสจังหวัด การจัดโครงสร้าง “ที่อยู่” ใหม่ จะนำเลขรหัสไปตรวจสอบกับข้อมูลใน LUT จังหวัด(ฟิลด์ province_I) เมื่อรหัสตรงกับชื่อจังหวัดใด (ฟิลด์ province_n) ก็จะนำชื่อจังหวัด ไปเก็บไว้ในฟิลด์ NPROVINCE ดูการเชื่อมโยงได้จากรูปที่ 3.3

3.3.3 ฐานข้อมูลอาคาร ของกรมสรรพากร

ลักษณะข้อมูล “ที่อยู่” จะเป็นข้อมูลที่ได้จากการดำเนินการสำรวจเก็บข้อมูลปฐมภูมิภาคสนามเพื่อให้ได้มาซึ่งรายละเอียดข้อมูลผู้ประกอบการ ซึ่งเป็นแหล่งภาษีอากร และเก็บข้อมูลตำแหน่งที่ตั้งจากพื้นที่จริง

ใช้ข้อมูล**เลขที่บ้าน** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลบ้านเลขที่ การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยก ข้อมูลบ้านเลขที่ เก็บไว้ในฟิลด์ NADD_NO

ใช้ข้อมูล**หมู่** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลเลขหมู่ การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยกข้อมูลหมู่ เก็บไว้ในฟิลด์ NMOO

ใช้ข้อมูล**ชื่อซอย** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลตรอก/ซอย การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยก ข้อมูลซอย เก็บไว้ในฟิลด์ NSOI

ใช้ข้อมูล**ชื่อถนน** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลถนน การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยกข้อมูลถนน เก็บไว้ในฟิลด์ NROAD

ใช้ข้อมูล**ชื่อแขวง** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลตำบล การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยกข้อมูลตำบล เก็บไว้ในฟิลด์ NTAMBON

จากข้อมูลชื่อแขวงจะนำไปเทียบกับชื่อแขวงใน LUT แขวง(ฟิลด์ Tambol_nam) เพื่อให้ได้รหัสเขต(ฟิลด์ Amphur_id) แล้วนำรหัสเขตไปตรวจสอบกับข้อมูลใน LUT เขต(ฟิลด์ code_soc)

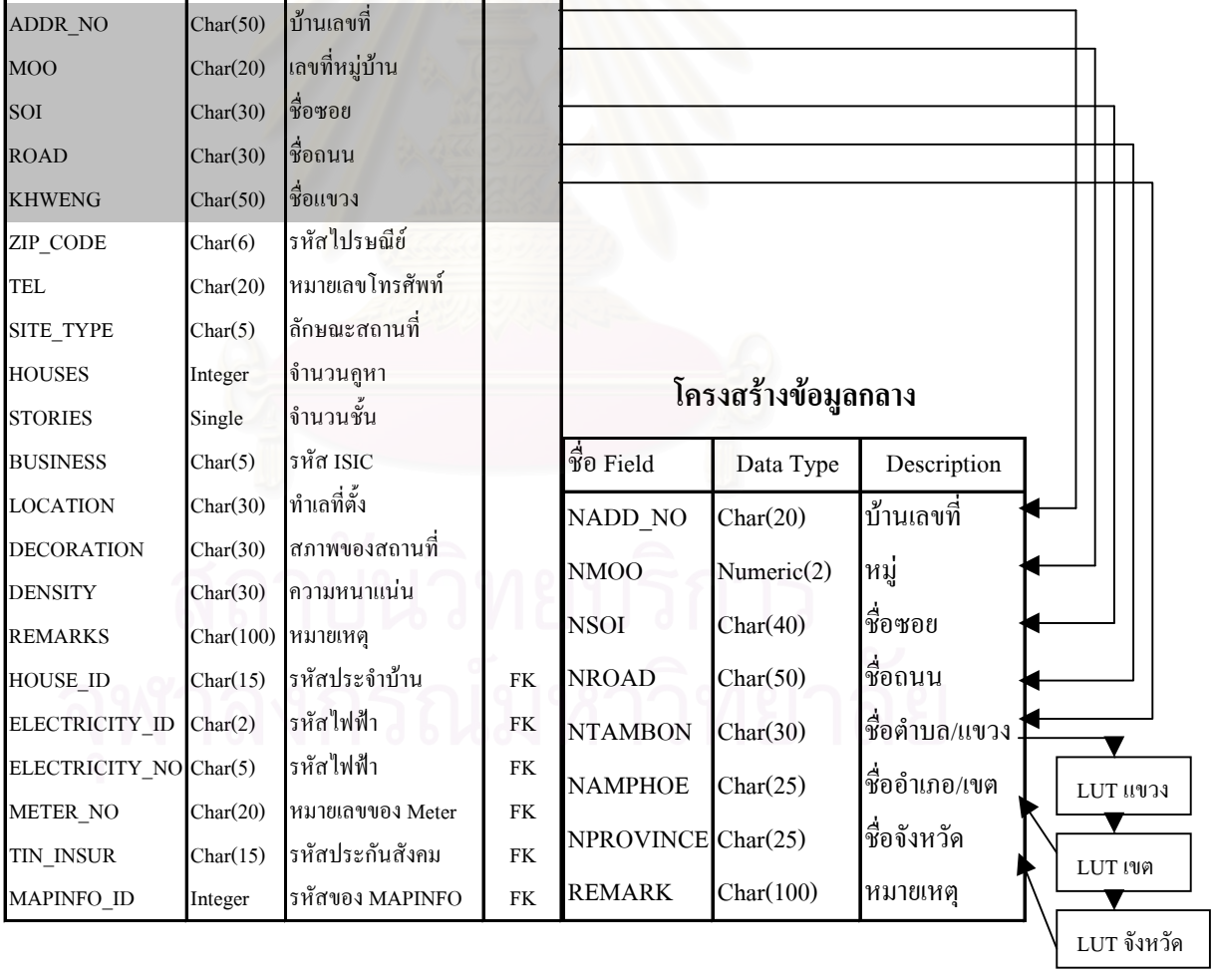
เมื่อรหัสตรงกับชื่ออำเภอใด(ฟิลด์ khed) ก็จะนำชื่ออำเภอ ไปเก็บไว้ในฟิลด์ NAMPHOE จากนั้นก็จะนำรหัสจังหวัดคือรหัสเขต 2 ตัวหน้า ไปตรวจสอบกับข้อมูลใน LUT จังหวัด(ฟิลด์ province_I) เมื่อรหัสตรงกับชื่อจังหวัดใด(ฟิลด์ province_n) ก็จะนำชื่อจังหวัด ไปเก็บไว้ในฟิลด์ NPROVINCE ดูการเชื่อมโยงได้จากรูปที่ 3.4

ฐานข้อมูลอาคาร ของกรมสรรพากร

Field	Data Type	Description	Remark
FIELD_ID	Char(10)	รหัสสำรวจ	FK
TIN	Char(15)	เลขประจำตัวผู้เสียภาษี	FK
TAX_PAYER	Char(50)	ชื่อผู้เสียภาษี	
REG_NAME	Char(50)	ชื่อสถานประกอบการ	
BLDG_NAME	Char(50)	ชื่ออาคาร	
FLOOR	Char(20)	ชั้น	
ADDR_NO	Char(50)	บ้านเลขที่	
MOO	Char(20)	เลขที่หมู่บ้าน	
SOI	Char(30)	ชื่อซอย	
ROAD	Char(30)	ชื่อถนน	
KHWENG	Char(50)	ชื่อแขวง	
ZIP_CODE	Char(6)	รหัสไปรษณีย์	
TEL	Char(20)	หมายเลขโทรศัพท์	
SITE_TYPE	Char(5)	ลักษณะสถานที่	
HOUSES	Integer	จำนวนคูหา	
STORIES	Single	จำนวนชั้น	
BUSINESS	Char(5)	รหัส ISIC	
LOCATION	Char(30)	ทำเลที่ตั้ง	
DECORATION	Char(30)	สภาพของสถานที่	
DENSITY	Char(30)	ความหนาแน่น	
REMARKS	Char(100)	หมายเหตุ	
HOUSE_ID	Char(15)	รหัสประจำบ้าน	FK
ELECTRICITY_ID	Char(2)	รหัสไฟฟ้า	FK
ELECTRICITY_NO	Char(5)	รหัสไฟฟ้า	FK
METER_NO	Char(20)	หมายเลขของ Meter	FK
TIN_INSUR	Char(15)	รหัสประกันสังคม	FK
MAPINFO_ID	Integer	รหัสของ MAPINFO	FK

โครงสร้างข้อมูลกลาง

ชื่อ Field	Data Type	Description
NADD_NO	Char(20)	บ้านเลขที่
NMOO	Numeric(2)	หมู่
NSOI	Char(40)	ชื่อซอย
NROAD	Char(50)	ชื่อถนน
NTAMBON	Char(30)	ชื่อตำบล/แขวง
NAMPHOE	Char(25)	ชื่ออำเภอ/เขต
NPROVINCE	Char(25)	ชื่อจังหวัด
REMARK	Char(100)	หมายเหตุ



รูปที่ 3.4 แสดงการจัดโครงสร้าง “ที่อยู่” ใหม่จากโครงสร้างฐานข้อมูลอาคาร ของกรมสรรพากร

3.3.4 ฐานข้อมูลเลขประจำตัวผู้เสียภาษี ของกรมสรรพากร

ข้อมูลที่นำมาใช้เป็นข้อมูลผู้เสียภาษีนิติบุคคล ซึ่งข้อมูล “ที่อยู่” จะเป็น “ที่อยู่” บริษัทที่จดทะเบียน ซึ่งจะสัมพันธ์กับตำแหน่งของอาคาร

ฐานข้อมูลเลขประจำตัวผู้เสียภาษี ของกรมสรรพากร

Field	หลักที่	จำนวนหลัก
เลขประจำตัวผู้เสียภาษี	1-10	10
คำนำหน้าชื่อ	11-35	25
ชื่อนิติบุคคล	36-95	60
เลขที่บ้าน	96-115	20
ตรอก/ซอย	116-119	4
หมู่	120-121	2
ถนน	122-151	30
ตำบล	152-181	30
รหัสจังหวัด/อำเภอ	182-185	4
รหัสไปรษณีย์	186-190	5

โครงสร้างข้อมูลกลาง

ชื่อ Field	Data Type	Description
NADD_NO	Char(20)	บ้านเลขที่
NMOO	Numeric(2)	หมู่
NSOI	Char(40)	ชื่อซอย
NROAD	Char(50)	ชื่อถนน
NTAMBON	Char(30)	ชื่อตำบล/แขวง
NAMPHOE	Char(25)	ชื่ออำเภอ/เขต
NPROVINCE	Char(25)	ชื่อจังหวัด
REMARK	Char(100)	หมายเหตุ

LUT เขต

LUT จังหวัด

รูปที่ 3.5 แสดงการจัดโครงสร้าง “ที่อยู่” จากโครงสร้างฐานข้อมูลเลขประจำตัวผู้เสียภาษีนิติบุคคล

ใช้ข้อมูล**เลขที่บ้าน** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลบ้านเลขที่ การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยก ข้อมูลบ้านเลขที่ เก็บไว้ในฟิลด์ NADD_NO

ใช้ข้อมูล**ตรอก/ซอย** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลตรอก/ซอย การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยก ข้อมูลซอย เก็บไว้ในฟิลด์ NSOI

ใช้ข้อมูล**หมู่** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลเลขหมู่ การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยกข้อมูลหมู่ เก็บไว้ในฟิลด์ NMOO

ใช้ข้อมูล**ถนน** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลถนน การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยกข้อมูลถนน เก็บไว้ในฟิลด์ NROAD

ใช้ข้อมูล**ตำบล** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลตำบล การจัดโครงสร้าง “ที่อยู่” ใหม่จะแยกข้อมูลตำบล เก็บไว้ในฟิลด์ NTAMBON

ใช้ข้อมูล**รหัสจังหวัด/อำเภอ** โดยข้อมูลที่อยู่ในฟิลด์นี้จะมีข้อมูลเลขรหัสเขต การจัดโครงสร้าง “ที่อยู่” ใหม่จะนำเลขรหัสไปตรวจสอบกับข้อมูลใน LUT เขต(ฟิลด์ code_tin) เมื่อรหัสตรงกับชื่ออำเภอใด(ฟิลด์ khed) ก็จะนำชื่ออำเภอ ไปเก็บไว้ในฟิลด์ NAMPHOE จากนั้นก็จะนำรหัสจังหวัดคือรหัสเขต 2 ตัวหน้า ไปตรวจสอบกับข้อมูลใน LUT จังหวัด(ฟิลด์ province_I) เมื่อรหัสตรงกับชื่อจังหวัดใด(ฟิลด์ province_n) ก็จะนำชื่อจังหวัด ไปเก็บไว้ในฟิลด์ NPROVINCE ดูการเชื่อมโยงได้จากรูปที่ 3.5

3.4 ขั้นตอนการตรวจสอบและแก้ไขข้อมูล “ที่อยู่”

เมื่อทำการแยกข้อมูลจัดเก็บในโครงสร้างข้อมูลกลางแล้ว ก่อนทำการเทียบรหัสเพื่อสร้างฟิลด์ที่จะใช้เชื่อมโยงฐานข้อมูลของหน่วยงานต่างๆ จะต้องทำการประมวลผลวิเคราะห์ความถูกต้องของการนำเข้าและแก้ไขข้อมูล “ที่อยู่” ให้ถูกต้องก่อนเพื่อให้การเชื่อมโยงมีประสิทธิภาพมากยิ่งขึ้น ในการตรวจสอบความถูกต้องของการนำเข้าข้อมูลจำเป็นที่จะต้องหาค่าความผิดพลาดขึ้นเพื่อแสดงให้เห็นว่าข้อมูลที่มีอยู่ มีความผิดพลาดมากเพียงใด ถ้าไม่ทำการแก้ไขข้อมูลให้ถูกต้องก่อนแล้ว การเชื่อมโยงฐานข้อมูลก็จะไม่สามารถทำได้ผลที่น่าพอใจนัก ในวิทยานิพนธ์ฉบับนี้จึงขอเสนอวิธีการวัดผลความผิดพลาดในการนำเข้าข้อมูลของหน่วยงานต่างๆ โดยคิดเป็นเปอร์เซ็นต์ความผิดพลาดของการพิมพ์ผิด คิดจากสูตร

$$\text{เปอร์เซ็นต์ความผิดพลาดของการพิมพ์ข้อมูล} = \frac{\text{จำนวนเรคคอร์ดที่พิมพ์ผิด}}{\text{จำนวนเรคคอร์ดทั้งหมดของพื้นที่ศึกษา}} * 100$$

3.4.1 เกณฑ์ในการกำหนดเรคคอร์ดที่พิมพ์ผิด

1. ฟิลด์บ้านเลขที่

- มีตัวอักษรกำหนดให้เป็นเรคคอร์ดที่พิมพ์ผิด
- ให้มีสัญลักษณ์ได้คือ “-” , “,” และ “/” ถ้ามีสัญลักษณ์เป็นอย่างอื่นนอกจากนี้ให้เป็นเรคคอร์ดที่พิมพ์ผิด

2. ฟิลด์หมู่

- ถ้ามีตัวอักษรกำหนดให้เป็นเรคคอร์ดที่พิมพ์ผิด

3. ฟิลด์ซอย และฟิลด์ถนน

- คำที่พิมพ์ผิด เช่น ศุรศักดิ์ คำที่ถูกต้องถือ สุรศักดิ์ กำหนดให้เป็นเรคคอร์ดที่พิมพ์ผิด

- ข้อมูลที่ไม่ได้เว้นวรรคคั่นระหว่างข้อมูล เช่น สุริวงศ์พลาซ่าถ.สุริวงศ์ กำหนดให้เป็นเรคคอร์ดที่พิมพ์ผิด
- คำที่ไม่มีใน LUT

4. ฟิลด์ตำบล

- คำที่พิมพ์ผิด กำหนดให้เป็นเรคคอร์ดที่พิมพ์ผิด
- ข้อมูลที่ไม่ได้เป็นชื่อตำบลหรือแขวง กำหนดให้เป็นเรคคอร์ดที่พิมพ์ผิด

3.5 สถานภาพข้อมูล หลังจากแยกข้อมูล “ที่อยู่” โดยใช้โปรแกรม

เปอร์เซ็นต์ความผิดพลาดของการพิมพ์ข้อมูล จะคิดเป็น 2 กรณี เพื่อเปรียบเทียบว่าการที่กรอกข้อมูลไม่ครบมีผลต่อเปอร์เซ็นต์ความผิดพลาดของการพิมพ์ข้อมูลมากน้อยเพียงใด

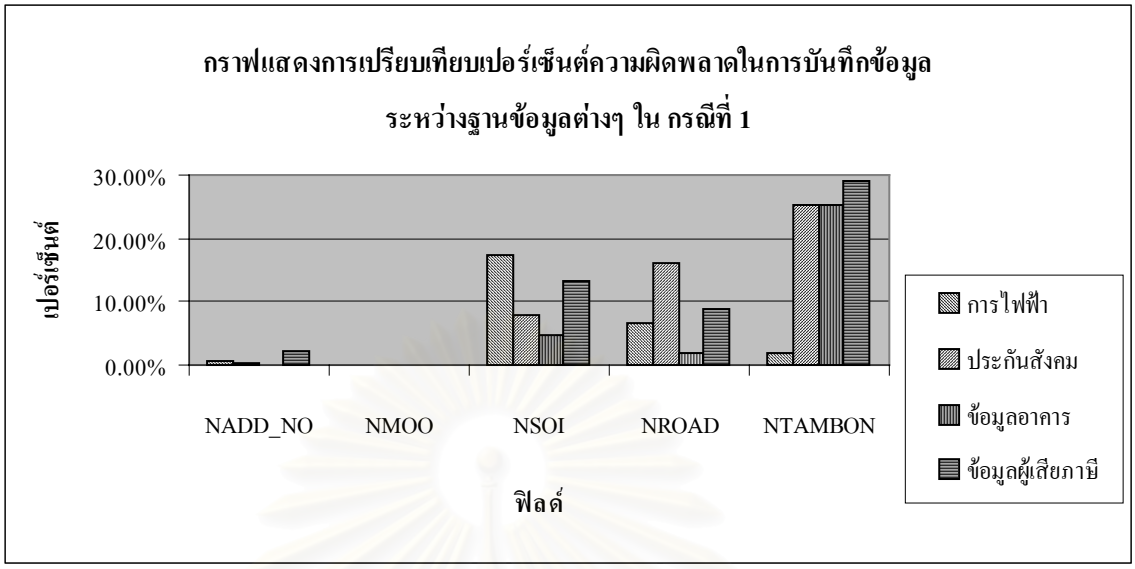
3.5.1 กรณีที่ 1 ไม่ถือว่าเรคคอร์ดที่มีไม่ครบทุกฟิลด์เป็นเรคคอร์ดที่พิมพ์ผิด

โดยในกรณีที่ 1 จะแสดงถึงความผิดพลาดในการพิมพ์ข้อมูลโดยไม่รวมความผิดพลาดในการกรอกข้อมูลไม่ครบ ดังจะแสดงความผิดพลาดในการพิมพ์บันทึกข้อมูลของหน่วยงานต่างๆ ดังในตารางที่ 3.2

ตารางที่ 3.2 ความผิดพลาดในการพิมพ์บันทึกข้อมูลของฐานข้อมูลต่างๆ ในกรณีที่ 1

หน่วยงาน	ข้อมูล	จำนวนข้อมูลทั้งหมด	NADD_NO		NMOO		NSOI		NROAD		NTAMBON	
			ผิด	%	ผิด	%	ผิด	%	ผิด	%	ผิด	%
การไฟฟ้า	การใช้ไฟฟ้า	11,029	82	0.74%	0	0.00%	1930	17.50%	733	6.65%	196	1.78%
ประกันสังคม	ประกันสังคม	612	1	0.16%	0	0.00%	48	7.84%	99	16.18%	154	25.16%
สรรพากร	ข้อมูลอาคาร	8,944	1	0.01%	0	0.00%	415	4.64%	174	1.95%	2246	25.11%
	ข้อมูลผู้เสียภาษี	6,766	157	2.32%	0	0.00%	889	13.14%	594	8.78%	1968	29.09%

จากตารางที่ 3.2 เมื่อเปรียบเทียบทั้งสี่ฐานข้อมูลแล้วสามารถแสดงเป็นกราฟแท่งดังรูปที่ 3.6



รูปที่ 3.6 กราฟแสดงการเปรียบเทียบเปอร์เซ็นต์ความผิดพลาดในการบันทึกข้อมูลระหว่างฐานข้อมูลต่างๆ ในกรณีที่ 1

ค่าความผิดพลาดในการพิมพ์บันทึกข้อมูลระหว่างหน่วยงานต่างๆ ในกรณีที่ 1 ในรูปที่ 3.6 **ฟิลด์ NADD_NO** จะมีค่าน้อยกว่าฟิลด์อื่นๆ โดยในแต่ละหน่วยงานจะมีเปอร์เซ็นต์ความผิดพลาดใกล้เคียงกัน

ฟิลด์ NMOO ที่มีค่าเป็น 0% เนื่องจากเป็นข้อมูลในบริเวณกรุงเทพฯ ซึ่งโดยส่วนใหญ่จะไม่มีเลขหมู่ โดยในแต่ละหน่วยงานจะมีเปอร์เซ็นต์ความผิดพลาดใกล้เคียงกัน

ฟิลด์ NSOI เมื่อเปรียบเทียบกับฟิลด์อื่นๆ แล้วจะมีค่าเปอร์เซ็นต์ความผิดพลาดอยู่มากที่พอสมควรเมื่อเทียบกับฟิลด์ NADD_NO และ NSOI เพราะชื่อซอยมีทั้งชื่อที่ใช้ในราชการ และชื่อสามัญที่ใช้ในท้องถิ่น ทำให้การนำเข้าข้อมูลมีหลากหลายกว่าข้อมูลในฟิลด์อื่นความผิดพลาดจึงเกิดสูง โดยหน่วยงานการไฟฟ้าจะมีค่าเปอร์เซ็นต์ความผิดพลาดสูงที่สุด

ฟิลด์ NROAD ในแต่ละหน่วยงานจะมีค่าเปอร์เซ็นต์ความผิดพลาดที่แตกต่างกัน ขึ้นอยู่กับคุณภาพของผู้นำเข้าข้อมูลในแต่ละหน่วยงาน

ฟิลด์ NTAMBON เมื่อเปรียบเทียบกับฟิลด์อื่นๆ แล้วจะมีค่าเปอร์เซ็นต์ความผิดพลาดมากที่สุด ยกเว้นหน่วยงานการไฟฟ้าที่มีเปอร์เซ็นต์ความผิดพลาดต่ำที่สุด

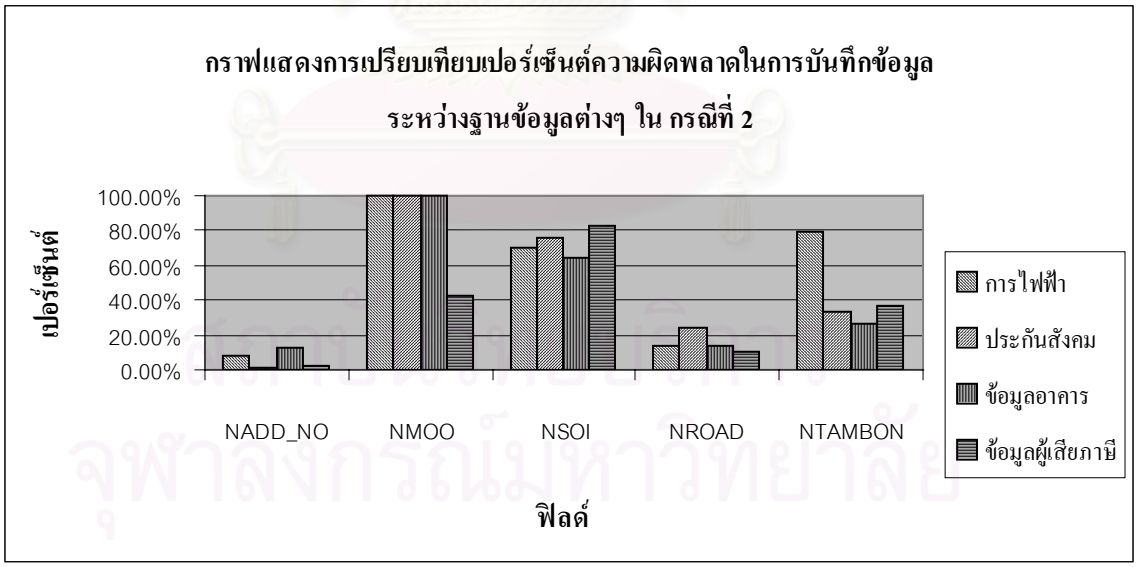
3.5.2 กรณีที่ 2 ถือว่าเรคคอร์ดที่มีไม่ครบทุกฟิลด์เป็นเรคคอร์ดที่พิมพ์ผิด

โดยในกรณีที่ 2 จะแสดงถึงความผิดพลาดในการพิมพ์ข้อมูลโดยรวมความผิดพลาดในการกรอกข้อมูลไม่ครบ ดังจะแสดงความผิดพลาดในการพิมพ์บันทึกข้อมูลของหน่วยงานต่างๆ ดังในตารางที่ 3.3

ตารางที่ 3.3 ความผิดพลาดในการพิมพ์บันทึกข้อมูลของฐานข้อมูลต่างๆ ในกรณีที่ 2

หน่วยงาน	ข้อมูล	จำนวนข้อมูลทั้งหมด	NADD_NO		NMOO		NSOI		NROAD		NTAMBON	
			ผิด	%	ผิด	%	ผิด	%	ผิด	%	ผิด	%
การไฟฟ้า	การใช้ไฟฟ้า	11,029	940	8.52%	11029	100.00%	7715	69.95%	1535	13.92%	8806	79.84%
ประกันสังคม	ประกันสังคม	612	9	1.47%	612	100.00%	467	76.31%	150	24.51%	204	33.33%
สรรพากร	ข้อมูลอาคาร	8,944	1108	12.39%	8944	100.00%	5780	64.62%	1208	13.51%	2331	26.06%
	ข้อมูลผู้เสียภาษี	6,766	177	2.62%	2847	42.08%	5635	83.28%	697	10.30%	2478	36.62%

จากตารางที่ 3.3 เมื่อเปรียบเทียบทั้งสี่ฐานข้อมูลแล้วสามารถแสดงเป็นกราฟแท่งดังรูปที่ 3.7



รูปที่ 3.7 กราฟแสดงการเปรียบเทียบเปอร์เซ็นต์ความผิดพลาดในการบันทึกข้อมูลระหว่างฐานข้อมูลต่างๆ ในกรณีที่ 2

ค่าความผิดพลาดในการพิมพ์บันทึกข้อมูลระหว่างหน่วยงานต่างๆ ในกรณีที่ 2 ในรูปที่ 3.7 **ฟิลด์ NADD_NO** จะมีค่าน้อยกว่าฟิลด์อื่นๆ โดยในแต่ละหน่วยงานจะมีเปอร์เซ็นต์ความผิดพลาดใกล้เคียงกัน เมื่อเทียบกับกรณีที่ 1 ในรูปที่ 3.6 จะมีค่าเพิ่มขึ้นไม่มากนัก

ฟิลด์ NMOO ที่มีค่า 100% เนื่องจากเป็นข้อมูลในบริเวณกรุงเทพฯ ซึ่งโดยส่วนใหญ่จะไม่มีเลขหมู่ ดังนั้นเมื่อกำหนดให้จำนวนเรคคอร์ดที่กรอกข้อมูลไม่ครบเป็นเรคคอร์ดที่ผิดแล้วค่าความผิดพลาดจึงมีค่าสูงมาก ยกเว้นในหน่วยงานกรมสรรพากร ที่มีค่า 42.08% เนื่องจากผู้พิมพ์บันทึกข้อมูลจะกรอกข้อมูลที่ไม่มีเลขหมู่เป็น “0” ดังนั้นเมื่อนำมาคิดเปอร์เซ็นต์ความผิดพลาดในกรณีที่ 2 ข้อมูลที่เป็นฟิลด์ว่างจึงมีค่าต่ำกว่าหน่วยงานอื่น ดังในรูปที่ 3.7

ฟิลด์ NSOI จะมีค่าเปอร์เซ็นต์ความผิดพลาดอยู่มากที่พอสมควร โดยในแต่ละหน่วยงานจะมีค่าใกล้เคียงกัน เมื่อเทียบกับกรณีที่ 1 ในรูปที่ 3.6 จะมีค่าเพิ่มขึ้นมาก

ฟิลด์ NROAD ในแต่ละหน่วยงานจะมีค่าเปอร์เซ็นต์ความผิดพลาดที่ใกล้เคียงกัน เมื่อเทียบกับกรณีที่ 1 ในรูปที่ 3.6 จะมีค่าเพิ่มขึ้นไม่มากนัก

ฟิลด์ NTAMBON ในแต่ละหน่วยงานจะมีค่าเปอร์เซ็นต์ความผิดพลาดที่ใกล้เคียงกัน ยกเว้นหน่วยงานการไฟฟ้าที่มีเปอร์เซ็นต์ความผิดพลาดสูงกว่าหน่วยงานอื่น ซึ่งเมื่อนำไปเปรียบเทียบกับกรณีที่ 1 ในรูปที่ 3.6 จะให้ผลในทางตรงกันข้าม โดยหน่วยงานของการไฟฟ้ามีเรคคอร์ดที่เป็นฟิลด์ว่างอยู่จำนวนมากอาจเนื่องมาจากกรอกข้อมูลไม่ครบ หรือไม่ใส่สัญลักษณ์ที่ระบุว่าเป็นตำบล เช่น ต. เขต หรือ ตำบล นำหน้าชื่อตำบล ในการแยกข้อมูลจึงไม่สามารถดำเนินการได้อย่างถูกต้อง ข้อมูลจึงไม่ได้ถูกนำไปใส่ในฟิลด์ NTAMBON

จากกราฟแสดงการเปรียบเทียบเปอร์เซ็นต์ความผิดพลาดในการบันทึกข้อมูลระหว่างหน่วยงานต่างๆ ในกรณีที่1 และกรณีที่ 2 ข้างต้น แสดงให้เห็นว่าความผิดพลาดในการพิมพ์ข้อมูลและการกรอกข้อมูลไม่ครบทำให้ค่าความผิดพลาดของข้อมูลสูงขึ้นดังในกรณีที่ 2 การที่จะทำให้ข้อมูลมาทำการเชื่อมโยงฐานข้อมูลเข้าด้วยกันนั้นจึงเป็นเรื่องยากที่ทำให้เปอร์เซ็นต์การเชื่อมโยงมีค่าสูง ดังนั้นจึงต้องทำการปรับปรุงแก้ไขข้อมูลให้มีความครบถ้วนมากขึ้นก่อนซึ่งจะกล่าวถึงในบทต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

การพัฒนาโปรแกรมจัดโครงสร้าง “ที่อยู่”

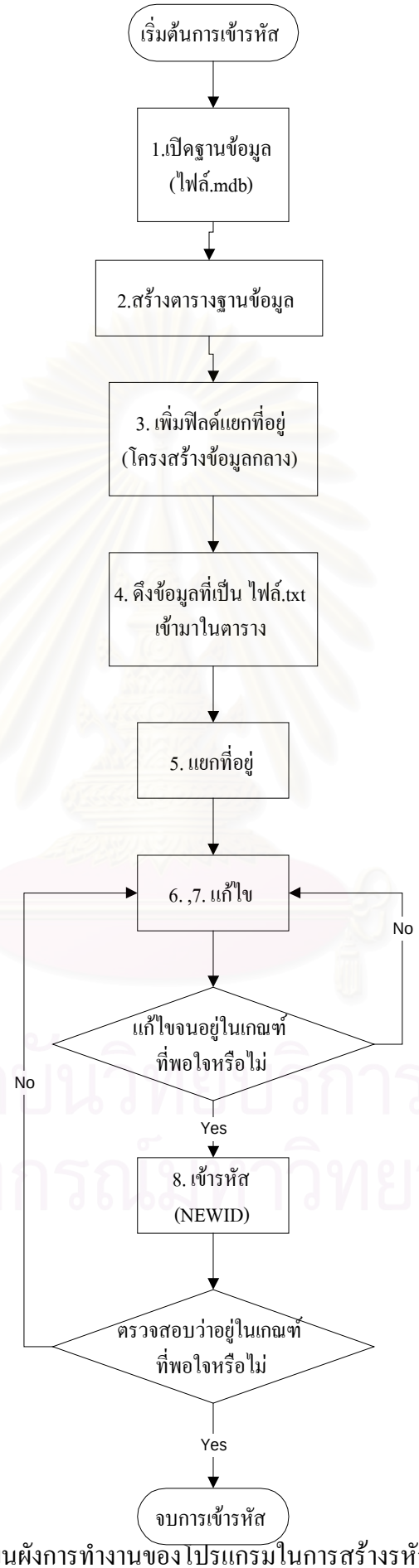
การพัฒนาโปรแกรมจัดโครงสร้าง “ที่อยู่” ใช้ภาษา Visual Basic ในการพัฒนาและใช้ฐานข้อมูลที่จัดเก็บอยู่ในโปรแกรม Access ในการพัฒนาโปรแกรมผู้วิจัยจะคำนึงถึงการที่นักพัฒนาโปรแกรมอื่นๆ จะนำโปรแกรมนี้ไปพัฒนาต่อ จึงจัดบางส่วนของโปรแกรมเป็นรูปแบบ Library เพื่อง่ายต่อการเรียกใช้งานด้วย

4.1 แนวความคิดในการพัฒนาโปรแกรม

เนื่องจากรูปแบบโครงสร้างของข้อมูลในแต่ละหน่วยงานมีความแตกต่างกัน การนำข้อมูลมาเชื่อมโยงกันจึงไม่สามารถทำได้ทันที การใช้ข้อมูล “ที่อยู่” เป็นตัวเชื่อมจึงมีความเหมาะสมที่สุดเพื่อใช้ในการสร้างความเชื่อมโยงเข้าสู่เลขรหัสประจำบ้าน ที่ควรใช้เป็นรหัสมาตรฐานในระบบภูมิสารสนเทศ การพัฒนาโปรแกรมจึงมีวัตถุประสงค์ช่วยในการจัดการโครงสร้างฐานข้อมูล “ที่อยู่” ของแต่ละหน่วยงานให้อยู่ในรูปแบบเดียวกันประกอบด้วยขั้นตอนการทำงานดังต่อไปนี้

1. เปิดฐานข้อมูล ใช้สำหรับเปิดฐานข้อมูลของหน่วยงานต่างๆ
2. การสร้างตารางฐานข้อมูลในรูปแบบของ Access
3. จัดโครงสร้างข้อมูล “ที่อยู่” ใหม่ให้มาอยู่ในรูปแบบโครงสร้างข้อมูลกลาง โดยเพิ่มฟิลด์แยกที่อยู่
4. การดึงข้อมูลในรูปแบบของ Text ไฟล์ ให้มาเก็บอยู่ในรูปแบบตารางฐานข้อมูล
5. แยกข้อมูล “ที่อยู่” ให้มาเก็บในฟิลด์ที่เพิ่มเข้ามาใหม่
6. แก้ไขข้อมูลที่ได้ผ่านการจัดโครงสร้างที่อยู่ใหม่ให้ถูกต้อง
7. แก้ไขข้อมูลระเบียนโดยสามารถเพิ่มหรือลบข้อมูลได้
8. เข้ารหัสสร้างเป็นฟิลด์ใหม่คือ ฟิลด์ NEWID

เพื่อใช้ฟิลด์ NEWID เป็นตัวเชื่อมฐานข้อมูลในแต่ละหน่วยงานต่อไป โดยดูแผนผังการทำงานในรูปที่ 4.1



รูปที่ 4.1 แผนผังการทำงานของโปรแกรมในการสร้างรหัส (NEWID)

4.2 ตารางเปรียบเทียบ (Look Up Table)

ในการเชื่อมโยงข้อมูลมีปัญหาที่สำคัญก็คือปัญหาเกี่ยวกับ ชื่อซอย ซึ่งจะมีทั้งชื่อสามัญและชื่อที่ใช้ในราชการ ในการศึกษาวิจัยครั้งนี้จึงมีแนวความคิดที่จะนำเอา Look Up Table มาช่วยในการแก้ปัญหา โดยในการเขียนโปรแกรมครั้งนี้จะใช้ Look Up Table ทั้งหมด 4 ตารางคือ

1. Look Up Table ของซอย ตรอก และถนน (ตารางที่ 4.1)
2. Look Up Table ของตำบล (ตารางที่ 4.2)
3. Look Up Table ของอำเภอ (ตารางที่ 4.3)
4. Look Up Table ของจังหวัด (ตารางที่ 4.4)

โดยในแต่ละตารางมีรายละเอียดดังนี้

ตารางที่ 4.1 โครงสร้างฐานข้อมูล Look Up Table ของซอย ตรอก และถนน

Column no.	Length	Name	Description	remark
1	7	id_road	รหัสถนน	
2	6	order	ลำดับเรียงตามเขต	
3	4	type	ชนิด ซอย ถนน ตรอก	
4	50	thai_na	ชื่อในราชการ	
5	50	local_tna	ชื่อสามัญ	
6	3	code_khed	รหัสเขต	

ตัวอย่างเช่น

Lutsrd

Id_road	Order	Type	Thai_na	Local_tna	Code_khed
1033	28	ซอย	เจริญกรุง29	พิทยลาภ	07
1034	29	ซอย	เจริญกรุง31	โชฎึก	07
1035	30	ซอย	เจริญกรุง33	โรงน้ำแข็ง	07
1036	31	ซอย	เจริญกรุง35	โรงน้ำแข็ง2	07
1037	32	ซอย	เจริญกรุง8	นครเกษม1	07
1038	33	ซอย	เจริญกรุง10	นครเกษม2	07
1039	34	ซอย	เจริญกรุง12	บำรุงรัฐฯ	07
1040	35	ซอย	เจริญกรุง14	กัมมาตุยาราม	07

รูปที่ 4.2 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของซอย ตรอก และถนน

จากตารางที่ 4.1 เป็นการแก้ปัญหาที่ชื่อซอยมีทั้งชื่อที่ใช้ในราชการและชื่อสามัญ จะมีฟิลด์ชื่อในราชการ (Thai_na) และฟิลด์ชื่อสามัญ (Local_tna) โดยใช้รหัสเดียวกัน (Id_road) การทำงานของโปรแกรมที่พัฒนาขึ้นนี้จะทำการเทียบรหัสชื่อในราชการและฟิลด์ชื่อสามัญ โดยดูจากตัวอย่างในรูปที่ 4.2 เช่น “ซอยเจริญกรุง29” เป็นชื่อในราชการ โดยมี “ซอยพิทยลาภ” เป็นชื่อสามัญ ใช้รหัสเดียวกันคือ “1033”

ตารางที่ 4.2 โครงสร้างฐานข้อมูล Look Up Table ของตำบล

Column no.	Length	Name	Description	remark
1	6	tambol_id	รหัสตำบล	
2	4	amphur_id	รหัสอำเภอ	
3	35	tambol_nam	ชื่อตำบล	

ตัวอย่างเช่น

Tambol

Tambol_id	Amphur_id	Tambol_nam
100101	1001	พระบรมมหาราชวัง
100102	1001	วังบูรพาภิรมย์
100103	1001	วัดราชบพิธ
100104	1001	สำราญราษฎร์
100105	1001	ศาลเจ้าพ่อเสือ

รูปที่ 4.3 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของตำบล

จากรูปที่ 4.3 เป็นตัวอย่างชื่อและรหัสของตำบลในเขตพระนคร ตัวอย่างเช่น แขวงพระบรมมหาราชวัง จะมีรหัสตำบลคือ “100101” โดยโปรแกรมจะนำรหัสนี้ไปเป็นส่วนประกอบของฟิลด์ NEWID ที่ใช้ในการเชื่อมโยงฐานข้อมูล

ตารางที่ 4.3 โครงสร้างฐานข้อมูล Look Up Table ของอำเภอ

Column no.	Length	Name	Description	remark
1	3	code_khed	รหัสเขตของกทม.	
2	20	khed	ชื่อเขต	
3	3	code_elec	รหัสเขตของการไฟฟ้า	
4	4	code_soc	รหัสเขตของประกันสังคม	
5	4	code_tin	รหัสเขตของสรรพากร	

ตัวอย่างเช่น

Lutkhed

Code_khed	Code_elec	Khed	Code_soc	Code_tin
05	1	พระนคร	1001	1047
06	3	ป้อมปราบศัตรูพ่าย	1008	1040
07	2	สัมพันธวงศ์	1013	1070
08	5	บางรัก	1004	1037
09	4	ปทุมวัน	1007	1041

รูปที่ 4.4 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของอำเภอ

จากรูปที่ 4.4 เป็นตัวอย่างชื่อและรหัสของเขต ตัวอย่างเช่น เขตพระนคร จะมีรหัสเขตคือ “1001” โดยโปรแกรมจะนำรหัสนี้ไปเป็นส่วนประกอบของฟิลด์ NEWID ที่ใช้ในการเชื่อมโยงฐานข้อมูล

ตารางที่ 4.4 โครงสร้างฐานข้อมูล Look Up Table ของจังหวัด

Column no.	Length	Name	Description	remark
1	2	province_i	รหัสจังหวัด	
2	35	province_n	ชื่อจังหวัด	
3	2	zone_id	รหัสโซน	

ตัวอย่างเช่น

Province

Province_i	Province_n	Zone_id
10	กรุงเทพมหานคร	0
11	สมุทรปราการ	1
12	นนทบุรี	1
13	ปทุมธานี	1

รูปที่ 4.5 ภาพตัวอย่างโครงสร้างฐานข้อมูล Look Up Table ของจังหวัด

จากรูปที่ 4.5 เป็นตัวอย่างชื่อและรหัสของจังหวัด ตัวอย่างเช่น กรุงเทพมหานคร จะมีรหัสจังหวัด คือ “10” โดยโปรแกรมจะนำรหัสนี้ไปเป็นส่วนประกอบของฟิลด์ NEWID ที่ใช้ในการเชื่อมโยงฐานข้อมูล

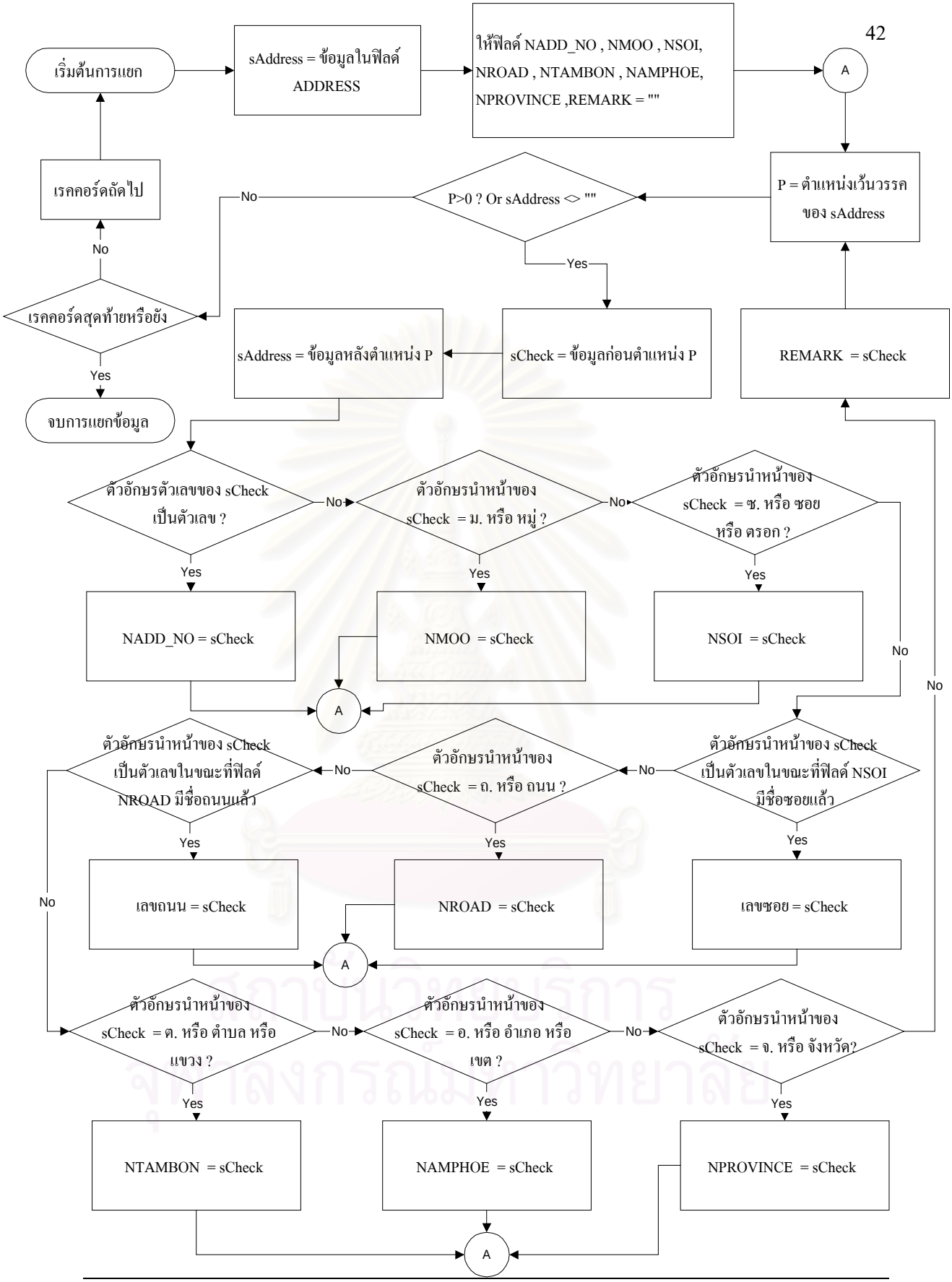
4.2.1 ผังระบบโปรแกรม

แผนผังที่จะนำเสนอในการพัฒนาโปรแกรมนี้จะเป็นแผนผังแสดงขั้นตอนการทำงานของโปรแกรมในส่วนต่างๆ โดยจะแสดงถึงขั้นตอนการทำงานหลักๆ ของโปรแกรม และแสดงแผนผังการทำงานในส่วนแยก "ที่อยู่" ซึ่งเป็นส่วนที่สำคัญที่สุดในการทำวิทยานิพนธ์ครั้งนี้ โดยจะแยกออกเป็น 2 กรณี คือ

1. กรณีที่จัดเก็บข้อมูล "ที่อยู่" ในฟิลด์เดียว

มีลักษณะการเก็บข้อมูล "ที่อยู่" ทั้งหมดรวมอยู่ในฟิลด์เดียวกันโดยจะมีทั้งข้อมูลบ้านเลขที่ หมู่ ซอย ถนน ตำบล อำเภอ และจังหวัด ฐานข้อมูลของการใช้ไฟฟ้า และฐานข้อมูลการประกันสังคม จะเก็บอยู่ในรูปแบบนี้ จึงจำเป็นที่จะต้องศึกษาอัลกอริธึมในการเปลี่ยนรูปแบบโครงสร้างให้เป็นโครงสร้างข้อมูลกลางดังแสดงในรูปที่ 4.6

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.6 แผนผังการทำงานของโปรแกรมในส่วนการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล “ที่อยู่” ในฟิลด์เดียว

จากรูปที่ 4.6 จะแสดงถึงขั้นตอนการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล “ที่อยู่” ในฟิลด์เดียว โดยเริ่มจากนำข้อมูลในฟิลด์ ADDRESS ไปเก็บไว้ในตัวแปร sAddress แล้วหาตำแหน่งของเว้นวรรคเก็บไว้ในตัวแปร P นำข้อมูลก่อนตำแหน่ง P ไปเก็บไว้ในตัวแปร sCheck และ นำข้อมูลหลังตำแหน่ง P ไปเก็บไว้ในตัวแปร sAddress จากนั้นนำตัวแปร sCheck ไปตรวจสอบเงื่อนไขเพื่อนำไปเก็บไว้ในโครงสร้างข้อมูลกลาง โดยเริ่มจาก

ฟิลด์แรก NADD_NO ตรวจสอบ sCheck ว่าตัวอักษรตัวแรกเป็นตัวเลขหรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NADD_NO
- **ไม่ใช่** ตรวจสอบฟิลด์ถัดไป

ฟิลด์ NMOO ตรวจสอบ sCheck ว่ามีตัวอักษรเป็น ม. หรือ หมู่ หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NMOO
- **ไม่ใช่** ตรวจสอบฟิลด์ถัดไป

ฟิลด์ NSOI ตรวจสอบ sCheck ว่ามีตัวอักษรเป็น ซ. หรือ ซอย หรือ ตรอก หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NSOI
- **ไม่ใช่** ตรวจสอบถัดไป

ตรวจสอบ sCheck ว่ามีตัวอักษรตัวแรกเป็นตัวเลข และมีชื่อซอยในฟิลด์ NSOI หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NSOI ต่อจากชื่อซอย
- **ไม่ใช่** ตรวจสอบฟิลด์ถัดไป

ฟิลด์ NROAD ตรวจสอบ sCheck ว่ามีตัวอักษรเป็น ถ. หรือ ถนน หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NROAD
- **ไม่ใช่** ตรวจสอบถัดไป

ตรวจสอบ sCheck ว่ามีตัวอักษรตัวแรกเป็นตัวเลข และมีชื่อซอยในฟิลด์ NROAD หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NROAD ต่อจากชื่อถนน
- **ไม่ใช่** ตรวจสอบฟิลด์ถัดไป

ฟิลด์ NTAMBON ตรวจสอบ sCheck ว่ามีตัวอักษรเป็น ต. หรือ ตำบล หรือ แขวง หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NTAMBON
- **ไม่ใช่** ตรวจสอบฟิลด์ถัดไป

ฟิลด์ NAMPHOE ตรวจสอบ sCheck ว่ามีตัวอักษรเป็น อ. หรือ อำเภอ หรือ เขต หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NAMPHOE
- **ไม่ใช่** ตรวจสอบฟิลด์ถัดไป

ฟิลด์ NPROVINCE ตรวจสอบ sCheck ว่ามีตัวอักษรเป็น จ. หรือ จังหวัด หรือไม่

- **ใช่** เก็บข้อมูลไว้ในฟิลด์ NPROVINCE
- **ไม่ใช่** เก็บข้อมูลไว้ในฟิลด์ REMARK

ตรวจสอบ sAddress ยังมีข้อมูลอยู่หรือไม่

- **ใช่** ตรวจสอบเงื่อนไขเพื่อนำไปเก็บไว้ในโครงสร้างข้อมูลกลาง
- **ไม่ใช่** ตรวจสอบต่อไป

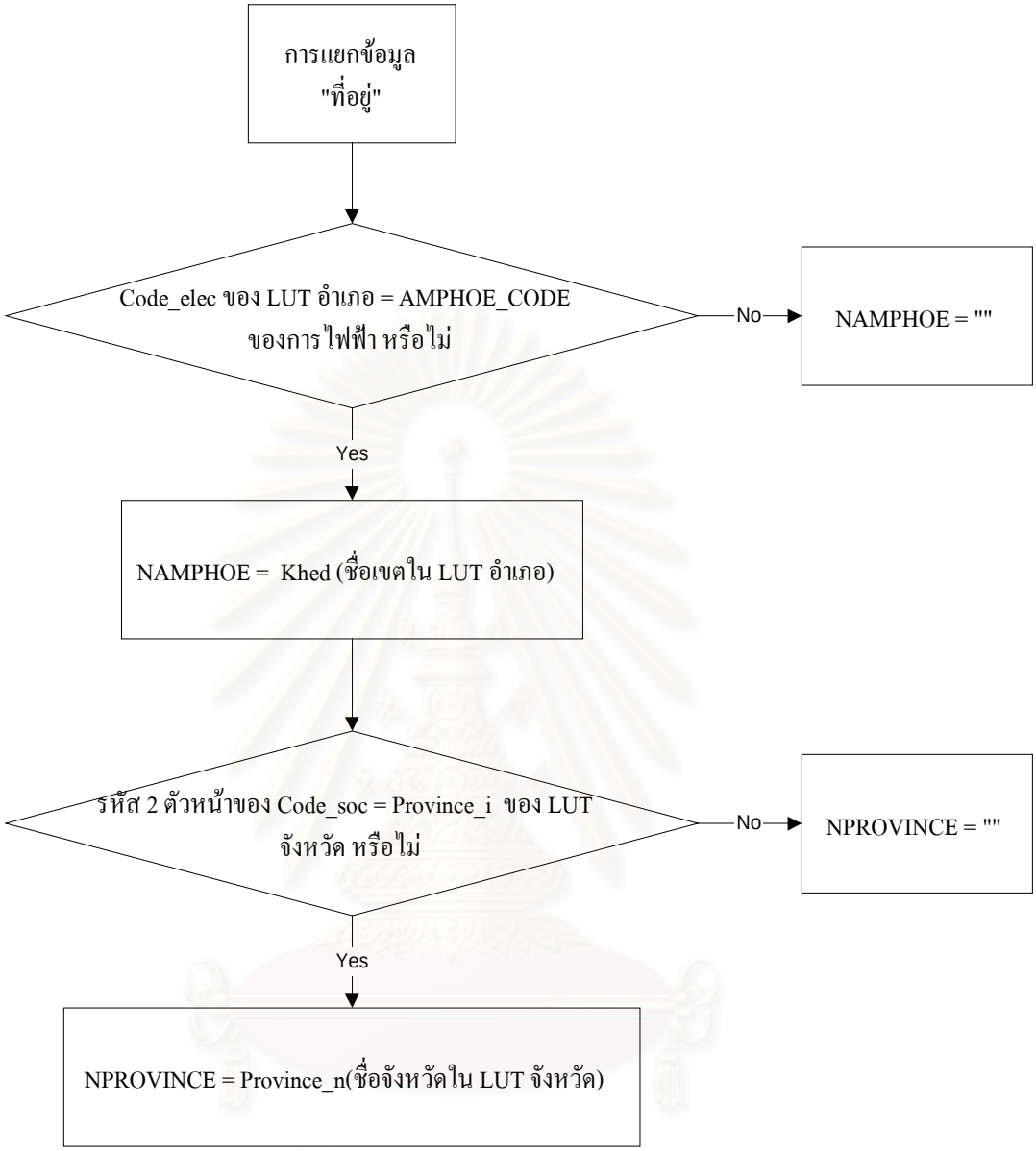
ตรวจสอบว่าเป็นเรคคอร์ดสุดท้ายหรือยัง

- **ใช่** จบการแยกข้อมูล
- **ไม่ใช่** ไปเรคคอร์ดถัดไป แล้วเริ่มต้นการแยกใหม่

ทำจนกว่าจะถึงเรคคอร์ดสุดท้าย

จากแผนผังการทำงานของโปรแกรมในส่วนของกรณีที่จัดเก็บข้อมูล "ที่อยู่" ในฟิลด์เดียว จะมีหน่วยงานที่จัดเก็บข้อมูลอยู่ในรูปแบบนี้คือ การไฟฟ้า และสำนักงานประกันสังคม แต่มีขั้นตอนที่แตกต่างกันในส่วนการทำงานหลังจากนี้ เพราะในแต่ละหน่วยงานจะมีการให้รหัสตำบลและรหัสอำเภอที่แตกต่างกัน จึงต้องมีการเขียนโปรแกรมการเข้ารหัสแตกต่างกันโดยจะแสดงในผังการทำงานในรูปที่ 4.7 และ 4.8

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.7 แผนผังการเทียบรหัสอำเภอและจังหวัดของการไฟฟ้า กับ LUT อำเภอและ LUT จังหวัด

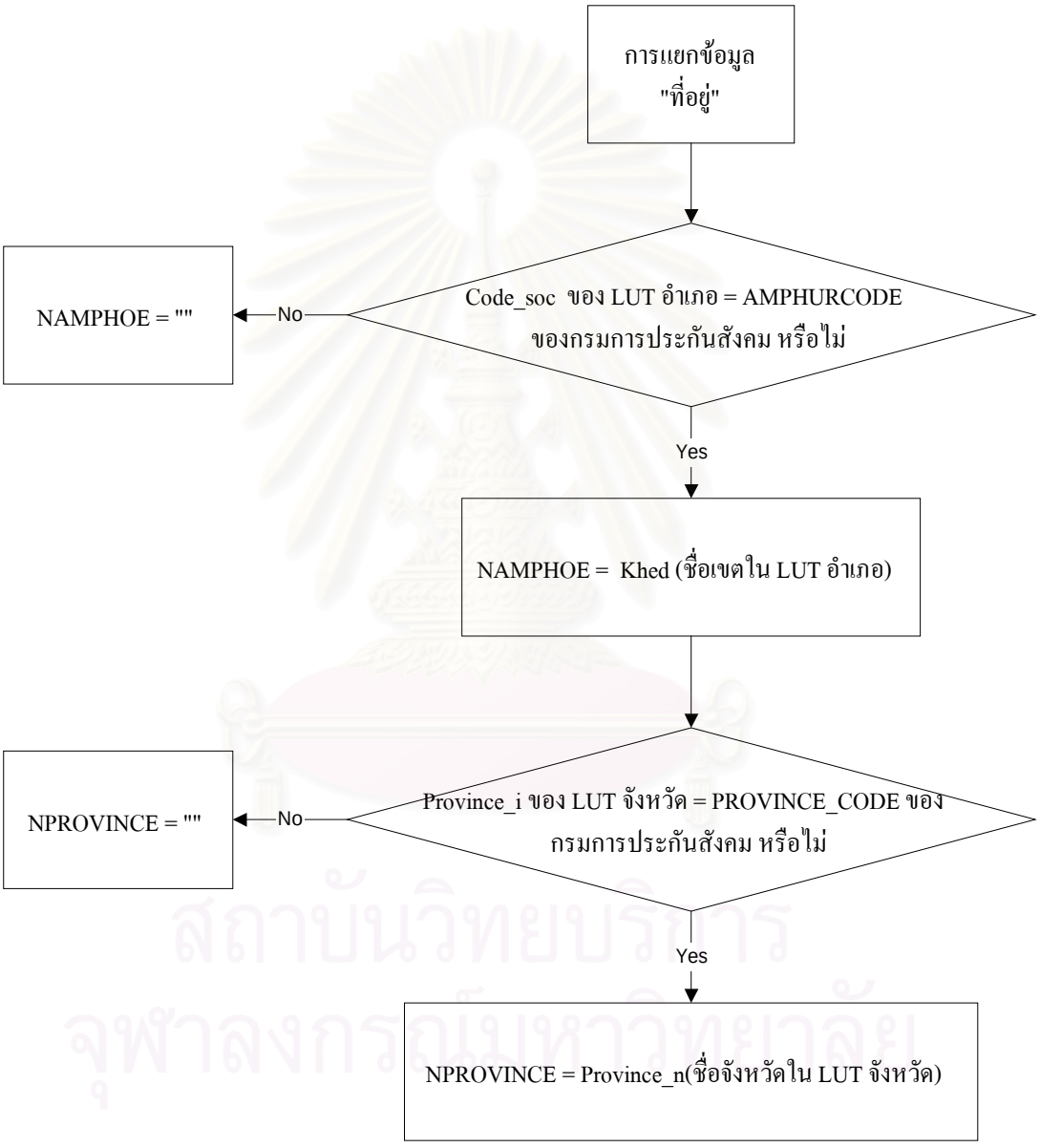
จากรูปที่ 4.7 เป็นการเทียบรหัสอำเภอและรหัสจังหวัด ของการไฟฟ้า จากตารางที่ 4.3 Look Up Table ของอำเภอ และ ตารางที่ 4.4 Look Up Table ของจังหวัด

โดยทำการตรวจสอบ AMPHOE_CODE ของการไฟฟ้า มีข้อมูลอยู่ในฟิลด์ Code_elec ใน Look Up Table ของอำเภอ หรือไม่

- **ใช่** เก็บข้อมูลในฟิลด์ Khed ใน Look Up Table ของอำเภอไว้ในฟิลด์ NAMPHOE แล้วตรวจสอบฟิลด์จังหวัดต่อไป
- **ไม่ใช่** ให้ข้อมูลในฟิลด์ NAMPHOE เท่ากับช่องว่าง

ตรวจสอบรหัส 2 ตัวหน้าของ Code_soc มีข้อมูลอยู่ในฟิลด์ Province_i ใน Look Up Table ของจังหวัด หรือไม่

- **ใช่** เก็บข้อมูลในฟิลด์ Province_n ใน Look Up Table ของจังหวัด ไว้ในฟิลด์ NPROVINCE
- **ไม่ใช่** ให้ข้อมูลในฟิลด์ NPROVINCE เท่ากับช่องว่าง



รูปที่ 4.8 แผนผังการเทียบรหัสอำเภอและจังหวัดของสำนักงานประกันสังคม กับ LUT อำเภอและ LUT จังหวัด

จากรูปที่ 4.8 จะมีลักษณะคล้ายกับการเทียบรหัสในรูปที่ 4.7 แต่จะแตกต่างกันที่มีการเทียบรหัสกันคนละฟิลด์ เนื่องจากเป็นหน่วยงานที่แตกต่างกันจึงมีการให้รหัสอำเภอ และรหัสจังหวัดแตกต่างกัน โดยมีการเทียบรหัสอำเภอและรหัสจังหวัด ของสำนักงานประกันสังคม จากตารางที่ 4.3 Look Up Table ของอำเภอ และ ตารางที่ 4.4 Look Up Table ของจังหวัด

โดยทำการตรวจสอบ AMPHURCODE ของสำนักงานประกันสังคม มีข้อมูลอยู่ในฟิลด์ Code_soc ใน Look Up Table ของอำเภอ หรือไม่

- **ใช่** เก็บข้อมูลในฟิลด์ Khed ใน Look Up Table ของอำเภอไว้ในฟิลด์ NAMPHOE แล้วตรวจสอบฟิลด์จังหวัดต่อไป
- **ไม่ใช่** ให้ข้อมูลในฟิลด์ NAMPHOE เท่ากับช่องว่าง

ตรวจสอบ PROVINCE_CODE ว่ามีข้อมูลอยู่ในฟิลด์ Province_i ใน Look Up Table ของจังหวัด หรือไม่

- **ใช่** เก็บข้อมูลในฟิลด์ Province_n ใน Look Up Table ของจังหวัด ไว้ในฟิลด์ NPROVINCE
- **ไม่ใช่** ให้ข้อมูลในฟิลด์ NPROVINCE เท่ากับช่องว่าง

2. กรณีที่จัดเก็บข้อมูล“ที่อยู่” แยกฟิลด์

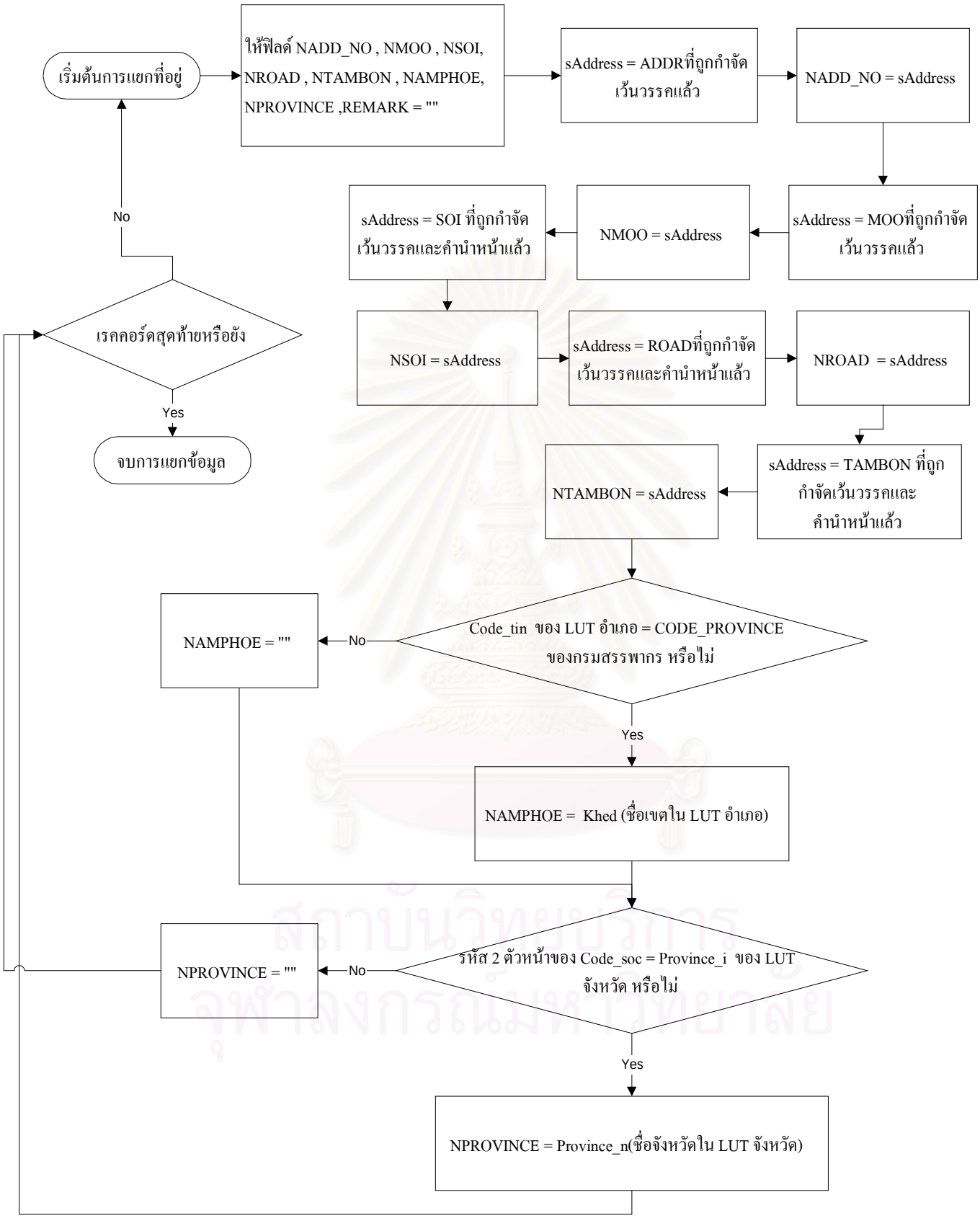
มีลักษณะการเก็บข้อมูล“ที่อยู่” โดยแยกเป็นฟิลด์ข้อมูลบ้านเลขที่ หมู่ ซอย ถนน ตำบล อำเภอ และจังหวัด แต่ก่อนที่จะนำมาข้อมูลมาเข้ารหัสนั้นจะต้องปรับปรุงคุณภาพให้ดีขึ้นก่อน จึงจำเป็นที่จะต้องศึกษาอัลกอริธึมในการเปลี่ยนรูปแบบโครงสร้างให้เป็นโครงสร้างข้อมูลกลางดังแสดงในรูปที่ 4.9 และ รูปที่ 4.10

จากรูปที่ 4.9 จะแสดงถึงขั้นตอนการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล “ที่อยู่” แยกฟิลด์โดยเก็บไว้ในโครงสร้างข้อมูลกลางของข้อมูลเลขประจำตัวผู้เสียภาษี ของกรมสรรพากร โดยเริ่มจาก

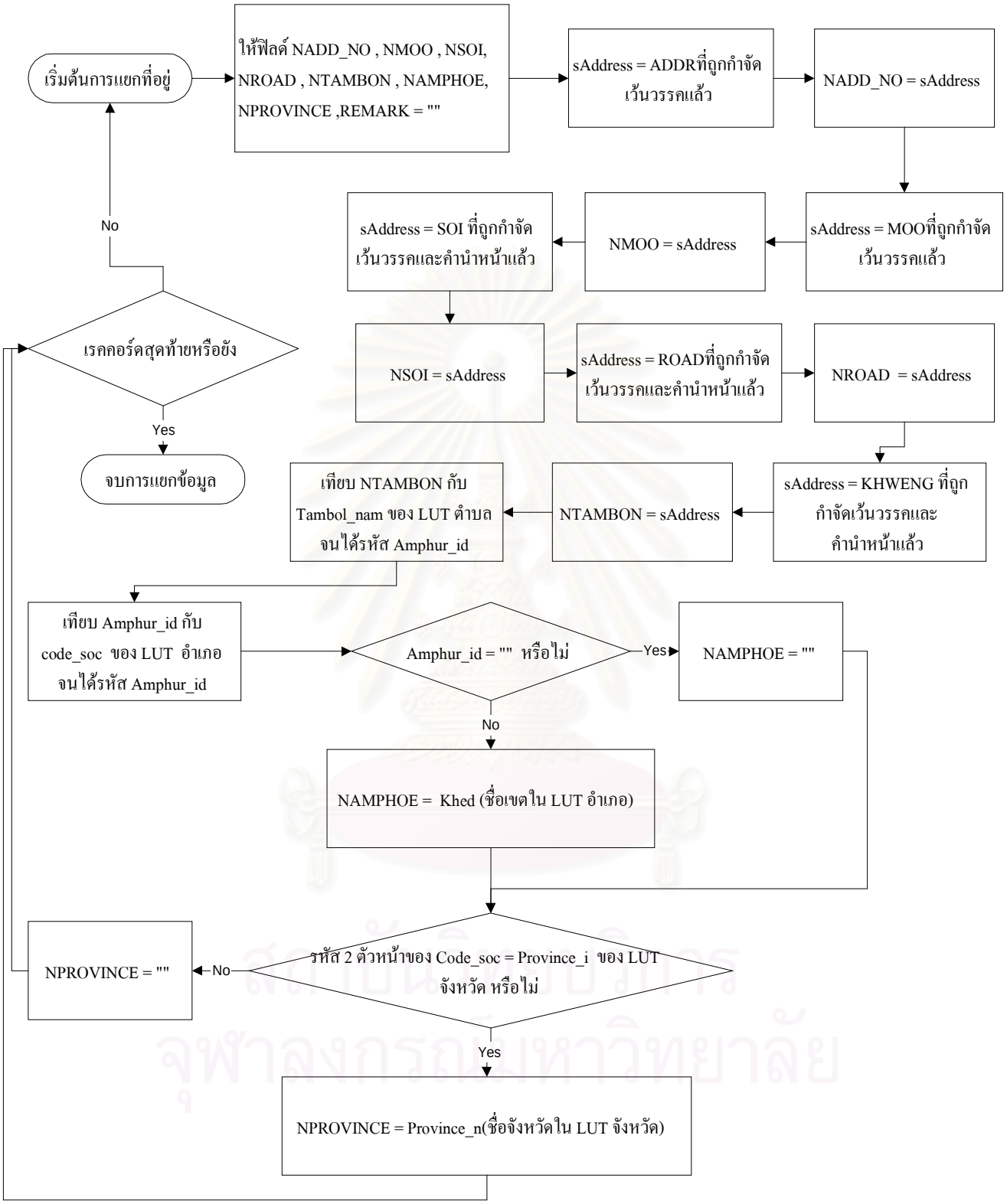
- นำข้อมูลในฟิลด์ ADDR ที่ถูกกำจัดเว้นวรรค แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NADD
- นำข้อมูลในฟิลด์ MOO ที่ถูกกำจัดเว้นวรรค แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NMOO
- นำข้อมูลในฟิลด์ SOI ที่ถูกกำจัดเว้นวรรคและคำนำหน้า แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NSOI
- นำข้อมูลในฟิลด์ ROAD ที่ถูกกำจัดเว้นวรรคและคำนำหน้า แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NROAD

- นำข้อมูลในฟิลด์ TAMBON ที่ถูกกำจัดเว้นวรรคและคำนำหน้า แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NTAMBON
- ทำการตรวจสอบ CODE_PROVINCE ของกรมสรรพากร มีข้อมูลอยู่ในฟิลด์ Code_tin ใน Look Up Table ของอำเภอ หรือไม่
  - **ใช่** เก็บข้อมูลในฟิลด์ Khed ใน Look Up Table ของอำเภอไว้ในฟิลด์ NAMPHOE แล้วตรวจสอบฟิลด์จังหวัดต่อไป
  - **ไม่ใช่** ให้ข้อมูลในฟิลด์ NAMPHOE เท่ากับช่องว่าง
- ตรวจสอบรหัส 2 ตัวหน้าของ Code_soc มีข้อมูลอยู่ในฟิลด์ Province_i ใน Look Up Table ของจังหวัด หรือไม่
  - **ใช่** เก็บข้อมูลในฟิลด์ Province_n ใน Look Up Table ของจังหวัด ไว้ในฟิลด์ NPROVINCE
  - **ไม่ใช่** ให้ข้อมูลในฟิลด์ NPROVINCE เท่ากับช่องว่าง
- ตรวจสอบว่าเป็นเรคคอร์ดสุดท้ายหรือยัง
  - **ใช่** จบการแยกข้อมูล
  - **ไม่ใช่** ไปเรคคอร์ดถัดไป แล้วเริ่มต้นการแยกใหม่
- ทำจนกว่าจะถึงเรคคอร์ดสุดท้าย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.9 แผนผังการทำงานของโปรแกรมในส่วนการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล “ที่อยู่” แยกฟิลด์ ของข้อมูลเลขประจำตัวผู้เสียภาษี (กรมสรรพากร)



รูปที่ 4.10 แผนผังการทำงานของโปรแกรมในส่วนของการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล“ที่อยู่” แยกฟิลด์ ของข้อมูลอาคาร (กรมสรรพากร)

จากรูปที่ 4.10 จะแสดงถึงขั้นตอนการแยก “ที่อยู่” กรณีที่จัดเก็บข้อมูล “ที่อยู่” แยกฟิลด์ โดยเก็บไว้ในโครงสร้างข้อมูลกลางของข้อมูลอาคาร โดยเริ่มจาก

- นำข้อมูลในฟิลด์ ADDR ที่ถูกกำจัดเว้นวรรค แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NADD
- นำข้อมูลในฟิลด์ MOO ที่ถูกกำจัดเว้นวรรค แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NMOO
- นำข้อมูลในฟิลด์ SOI ที่ถูกกำจัดเว้นวรรคและคำนำหน้า แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NSOI
- นำข้อมูลในฟิลด์ ROAD ที่ถูกกำจัดเว้นวรรคและคำนำหน้า แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NROAD
- นำข้อมูลในฟิลด์ KHWENG ที่ถูกกำจัดเว้นวรรคและคำนำหน้า แล้วไปเก็บไว้ในตัวแปร sAddress แล้วเก็บในฟิลด์ NTAMBON
- เทียบ NTAMBON กับ Tambol_nam ใน Look Up Table ของตำบล จนได้รหัส Amphur_id
- เทียบ Amphur_id กับ Code_soc ใน Look Up Table ของอำเภอ จนได้รหัส Amphur_id
- ทำการตรวจสอบ Amphur_id = “” หรือไม่
  - **ใช่** ให้ข้อมูลในฟิลด์ NAMPHOE เท่ากับช่องว่าง
  - **ไม่ใช่** เก็บข้อมูลในฟิลด์ Khed ใน Look Up Table ของอำเภอไว้ในฟิลด์ NAMPHOE แล้วตรวจสอบฟิลด์จังหวัดต่อไป
- ตรวจสอบรหัส 2 ตัวหน้าของ Code_soc มีข้อมูลอยู่ในฟิลด์ Province_i ใน Look Up Table ของจังหวัด หรือไม่
  - **ใช่** เก็บข้อมูลในฟิลด์ Province_n ใน Look Up Table ของจังหวัด ไว้ในฟิลด์ NPROVINCE
  - **ไม่ใช่** ให้ข้อมูลในฟิลด์ NPROVINCE เท่ากับช่องว่าง
- ตรวจสอบว่าเป็นเรคคอร์ดสุดท้ายหรือยัง
  - **ใช่** จบการแยกข้อมูล
  - **ไม่ใช่** ไปเรคคอร์ดถัดไป แล้วเริ่มต้นการแยกใหม่
- ทำจนกระทั่งถึงเรคคอร์ดสุดท้าย

4.3 ฟังก์ชันของโปรแกรม

ฟังก์ชันการทำงานของโปรแกรมจะแบ่งออกเป็น 8 ฟังก์ชัน คือ

4.3.1 ฟังก์ชันสร้างและการเปิดฐานข้อมูล

4.3.2 ฟังก์ชันสร้างตารางฐานข้อมูล

4.3.3 ฟังก์ชันดึงข้อมูล

4.3.4 ฟังก์ชันแยกที่อยู่

4.3.5 ฟังก์ชันแก้ไขข้อมูล

4.3.6 ฟังก์ชันแก้ไขข้อมูลระเบียน

4.3.7 ฟังก์ชันเข้ารหัส

4.3.8 ฟังก์ชันปิดการทำงานของโปรแกรม

4.3.1 ฟังก์ชันสร้างและการเปิดฐานข้อมูล

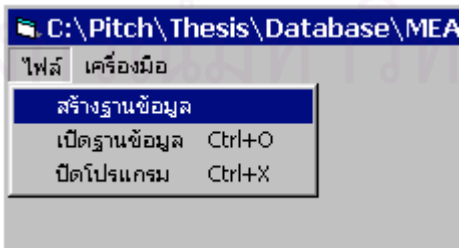
วัตถุประสงค์

ฟังก์ชันสร้างฐานข้อมูล จะทำการสร้างฐานข้อมูล ไฟล์.mdb ในโปรแกรม Access เพื่อเตรียมไว้สำหรับเก็บข้อมูลในกรณีที่ยังไม่เคยมีการจัดเก็บข้อมูลมาก่อน

ฟังก์ชันการเปิดฐานข้อมูล จะทำการเปิดไฟล์.mdb ในโปรแกรม Access ได้เคยมีการจัดเก็บข้อมูลมาแล้ว

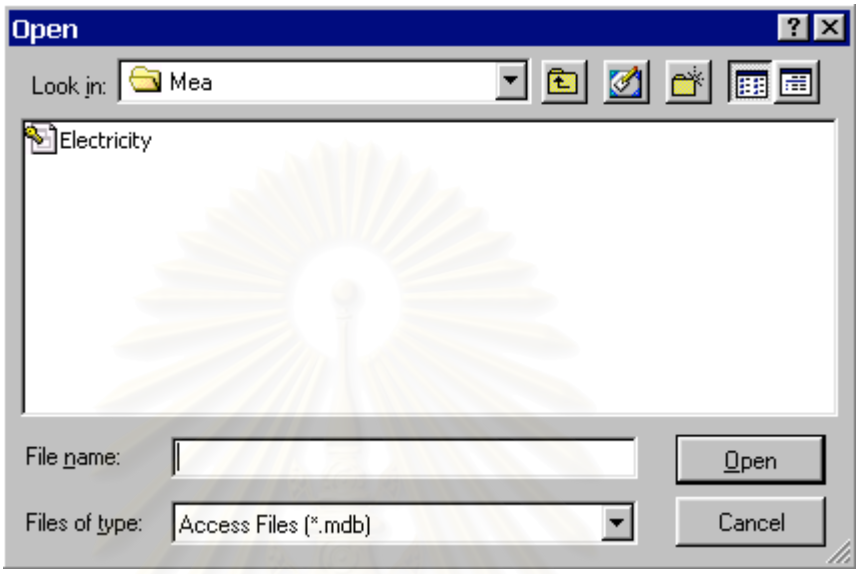
การใช้งาน

1. ทำการสร้างฐานข้อมูล โดยเลือกเมนูไฟล์ - > สร้างฐานข้อมูล หรือกดปุ่ม ในกรณีที่ได้ทำการสร้างฐานข้อมูลแล้ว ให้คลิกเลือกเมนูไฟล์ - > เปิดฐานข้อมูล ดังรูปที่ 4.11



รูปที่ 4.11 แสดงเมนูสร้างฐานข้อมูล

2. เมื่อคลิกเลือกแล้วจะปรากฏวินโดว์ขึ้น เพื่อใส่ชื่อฐานข้อมูลที่ต้องการจะสร้าง ในช่อง File name: แล้วคลิกเลือก Open โดยรูปแบบของไฟล์จะเป็น .mdb ดังรูปที่ 4.12



รูปที่ 4.12 แสดงวินโดว์สำหรับเปิดฐานข้อมูล

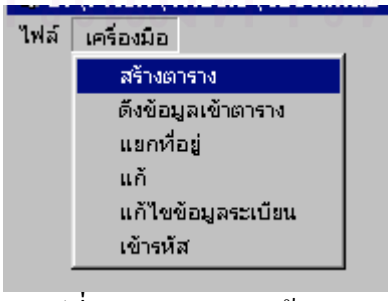
4.3.2 ฟังก์ชันสร้างตารางฐานข้อมูล

วัตถุประสงค์

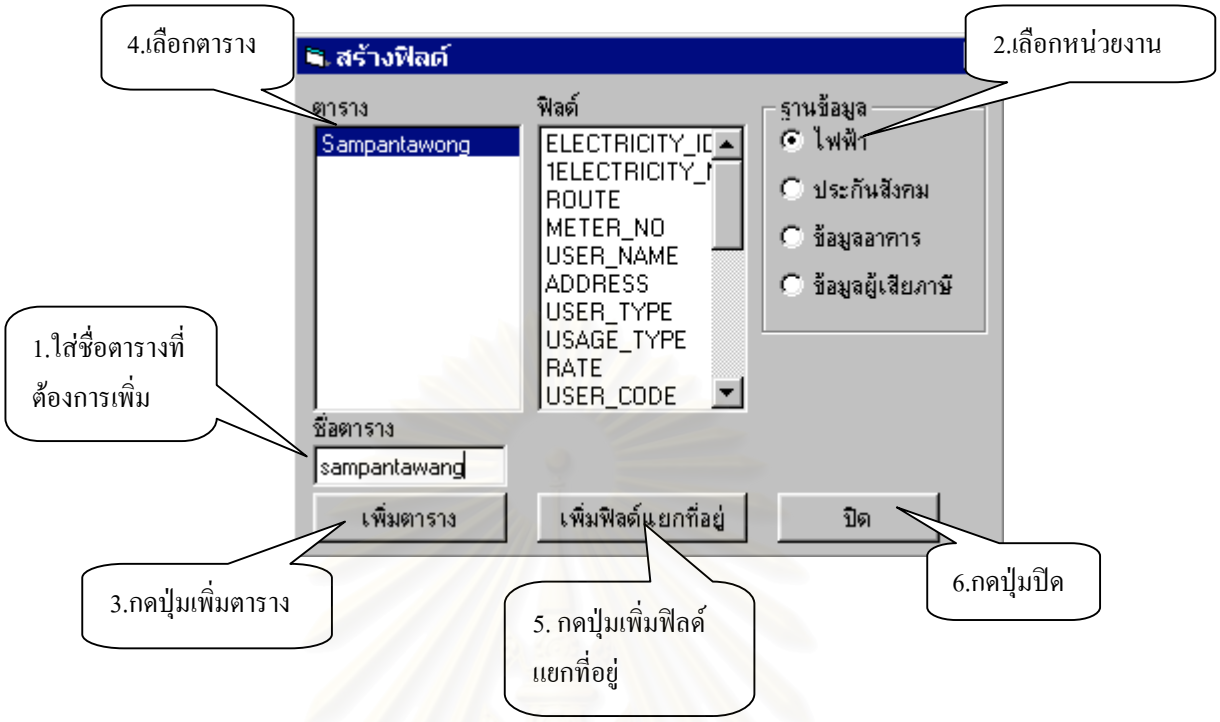
ฟังก์ชันสร้างตารางฐานข้อมูล จะสร้างตารางฐานข้อมูลโดยจะมีโครงสร้างแตกต่างกันไปตามแต่หน่วยงานที่ได้ทำการเลือก เป็นการเตรียมตารางเพื่อไว้สำหรับนำข้อมูลที่เป็น ไฟล์.txt เข้า

การใช้งาน

1. ทำการสร้างตารางในฐานข้อมูล โดยกดปุ่ม สร้างตาราง หรือ เลือกเมนู เครื่องมือ - > สร้างตาราง ดังรูปที่ 4.13



รูปที่ 4.13 แสดงเมนูสร้างตาราง



รูปที่ 4.14 แสดงวินโดว์สำหรับสร้างตาราง

2. เมื่อคลิกเลือกแล้วจะปรากฏวินโดว์ขึ้น ดังรูปที่ 4.14 เพื่อสร้างตาราง โดยใส่ชื่อตารางลงในช่องชื่อตาราง(1) แล้วเลือกหน่วยงานที่ต้องการจะสร้างตาราง(2) เมื่อเลือกแล้วคลิกเลือกปุ่ม เพิ่มตาราง (3)

<u>หมายเหตุ</u> ในแต่ละหน่วยงานจะมีโครงสร้างของฟิลด์ที่แตกต่างกัน โดยชื่อของฟิลด์ทั้งหมดของในแต่ละหน่วยงานจะปรากฏอยู่ในช่องฟิลด์ เมื่อคลิกไปที่ชื่อตารางในช่องตาราง (4)

3. จากนั้นคลิกปุ่ม เพิ่มฟิลด์แยกที่อยู่ (5) เพื่อสร้างฟิลด์ บ้านเลขที่ หมู่ ซอย ตำบล อำเภอ จังหวัด และ หมายเหตุ

4. ถ้าต้องการออกจากฟังก์ชันนี้ ให้คลิกปุ่ม ปิด (6)

4.3.3 ฟังก์ชันดึงข้อมูล

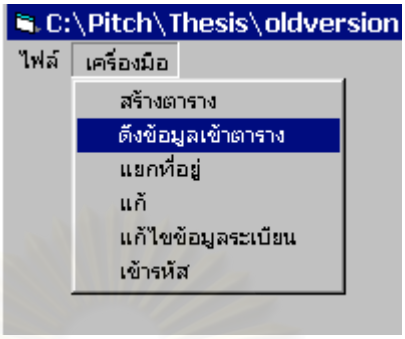
วัตถุประสงค์

ฟังก์ชันดึงข้อมูล จะนำข้อมูลที่อยู่ในรูปแบบของไฟล์ .txt ให้เข้าไปเก็บในตารางฐานข้อมูล .mdb ของ Access ที่ได้ทำการสร้างไว้แล้วก่อนหน้านี้

การใช้งาน

1. นำเข้าข้อมูลที่อยู่ในรูปแบบของไฟล์ .txt ให้เข้าไปอยู่ในตารางฐานข้อมูล .mdb โดยกดปุ่ม

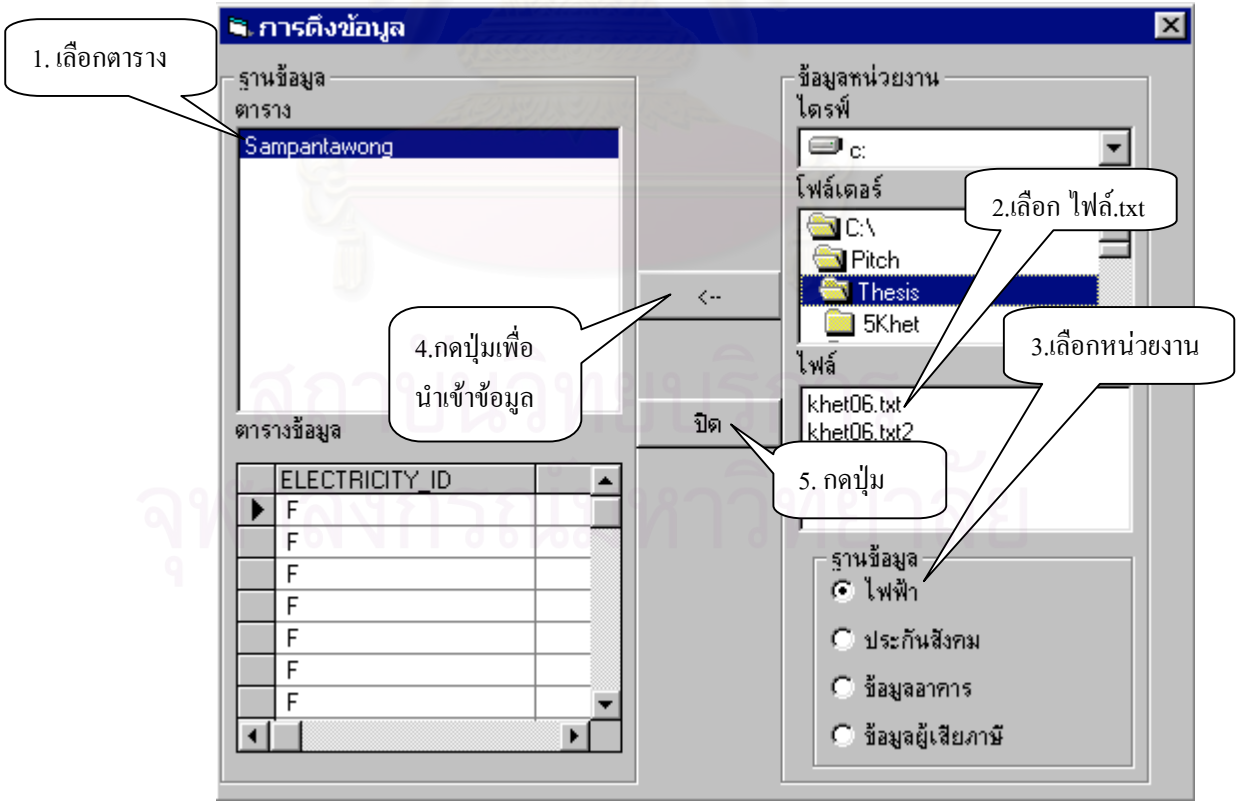
หรือ เลือกเมนูเครื่องมือ -> ดึงข้อมูลเข้าตาราง ดังรูปที่ 4.15



รูปที่ 4.15 แสดงเมนูดึงข้อมูลเข้าตาราง

2. เมื่อคลิกเลือกแล้วจะปรากฏวินโดว์ขึ้น เพื่อเลือกไฟล์ข้อมูล .txt และ เลือกตารางที่จะนำเข้าข้อมูล .txt โดยเลือกชื่อตารางที่ต้องการจะนำข้อมูลเข้าในช่องตาราง(1) และ เลือก ไฟล์ข้อมูล .txt(2) โดยเลือกไดรฟ์ - > โฟล์เดอร์ -> ชื่อไฟล์ จากนั้นทำการเลือกหน่วยงานที่จะนำข้อมูลเข้า(3) แล้วคลิกปุ่ม

(4) แล้วโปรแกรมจะแสดงผลลัพธ์ของการนำเข้าข้อมูล ที่ช่องตารางข้อมูล ดังรูปที่ 4.16



รูปที่ 4.16 แสดงวินโดว์สำหรับดึงข้อมูลเข้าตาราง

3. ถ้าต้องการออกจากฟังก์ชันนี้ ให้คลิกปุ่ม ปิด (5)

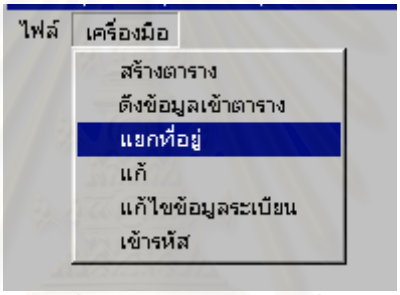
4.3.4 ฟังก์ชันแยกที่อยู่

วัตถุประสงค์

ฟังก์ชันแยกที่อยู่ จะทำการจัดโครงสร้างข้อมูล “ที่อยู่” ใหม่ให้อยู่ในรูปแบบกลาง โดยจะทำการแยกข้อมูลไปเก็บไว้ในฟิลด์ที่สร้างขึ้นมา 8 ฟิลด์ คือ NADD_NO, NMOO, NSOI, NROAD, NTAMBON, NAMPHOE, NPROVINCE และ REMARK

การใช้งาน

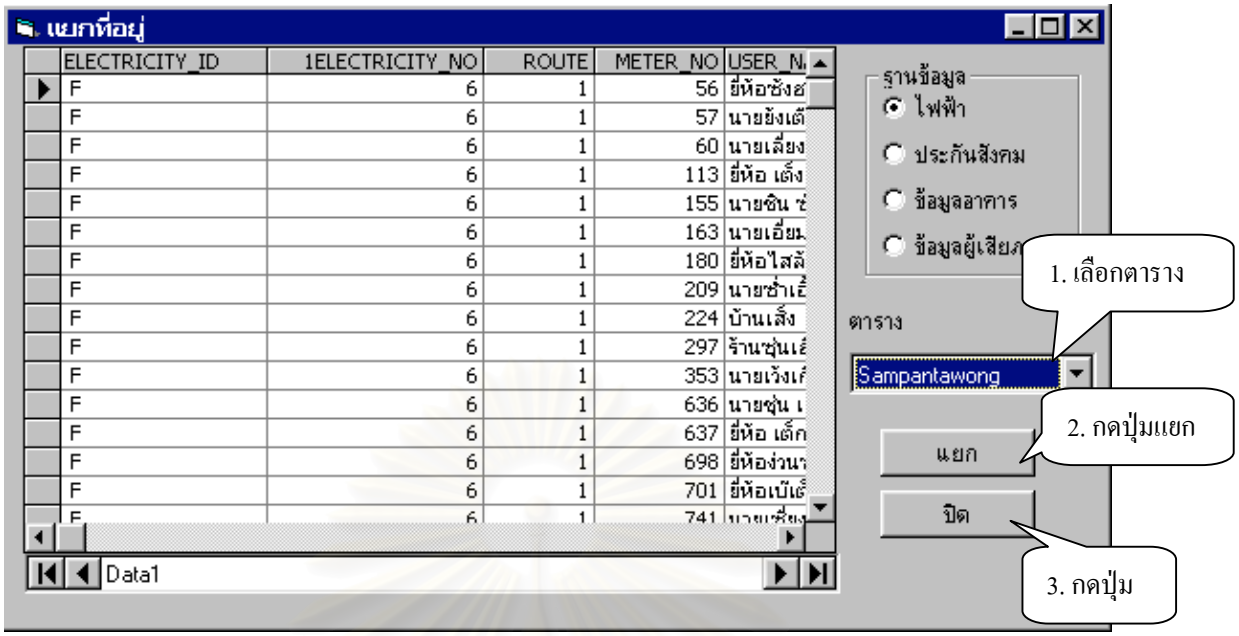
1. แยกข้อมูลที่อยู่ โดยกดปุ่ม แยกที่อยู่ หรือ เลือกเมนูเครื่องมือ -> แยกที่อยู่ ดังรูปที่ 4.17



รูปที่ 4.17 แสดงเมนูแยกที่อยู่

2. เมื่อคลิกเลือกแล้วจะปรากฏวินโดว์ขึ้น แล้วเลือกชื่อตาราง (1)โปรแกรมจะตรวจสอบโดยอัตโนมัติว่าเป็นหน่วยงานใด จากนั้นคลิกปุ่ม แยก (2) ดังรูปที่ 4.18

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.18 แสดงวินโดว์สำหรับแยกที่อยู่

3. ถ้าต้องการออกจากฟังก์ชันนี้ ให้คลิกปุ่ม

(3)

4.3.5 ฟังก์ชันแก้ไขข้อมูล

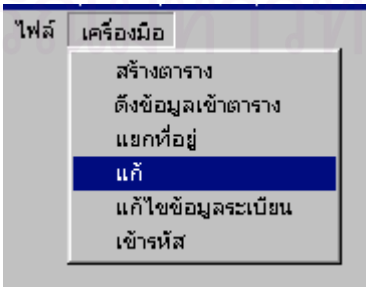
วัตถุประสงค์

ฟังก์ชันแก้ไขข้อมูล จะแก้ไขข้อมูลให้ถูกต้องและครบถ้วนก่อนทำการเข้ารหัส โดยเป็นฟังก์ชันแบบกึ่งอัตโนมัติ ให้ผู้ใช้เป็นผู้ตัดสินใจในการแก้ไขที่พิมพ์ผิดและเพิ่มเติมข้อมูลที่ขาดไปให้ครบถ้วน

การใช้งาน

1. แก้ไขคำผิดของข้อมูล โดยกดปุ่ม

หรือ เลือกเมนูเครื่องมือ -> แก้ ดังรูปที่ 4.19

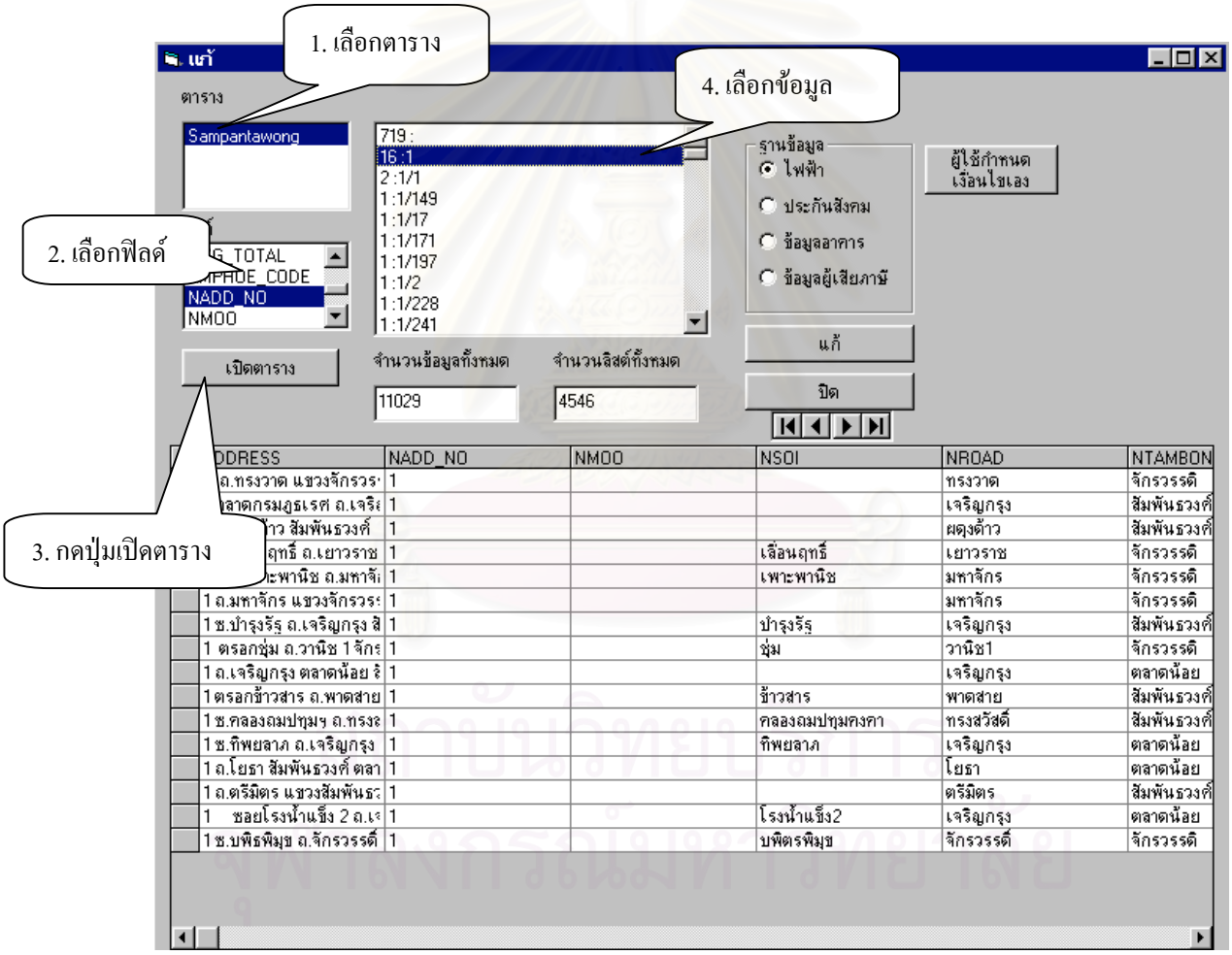


รูปที่ 4.19 แสดงเมนูแก้

2. เมื่อคลิกเลือกแล้วจะปรากฏวินโดว์ขึ้น ในช่องตารางจะปรากฏชื่อตารางที่มีอยู่ทั้งหมดในฐานข้อมูล

3. คลิกเลือกตารางที่ต้องการแก้ไข ในช่องตาราง(1) ในช่องฟิลด์จะปรากฏฟิลด์ทั้งหมดที่อยู่ในตารางที่เลือก

4. คลิกเลือกฟิลด์ที่ต้องการแก้ไข ในช่องฟิลด์(2) แล้วกดปุ่ม เปิดตาราง (3) จะปรากฏข้อมูลในฟิลด์ทั้งหมดที่อยู่ในตาราง ในช่องลิสต์ข้อมูลโดยจะมีตัวเลขจำนวนความถี่ของข้อมูลนั้นอยู่ข้างหน้า คั่นด้วยเครื่องหมาย : แล้วตามด้วยข้อมูล ในช่องจำนวนข้อมูลทั้งหมด จะแสดงจำนวนข้อมูลในฐานข้อมูลทั้งหมด และ ในช่องจำนวนลิสต์ทั้งหมด จะแสดงจำนวนที่แตกต่างกันของข้อมูลในฐานข้อมูล ดังรูปที่ 4.20



รูปที่ 4.20 แสดงวินโดว์สำหรับแก้ข้อมูล

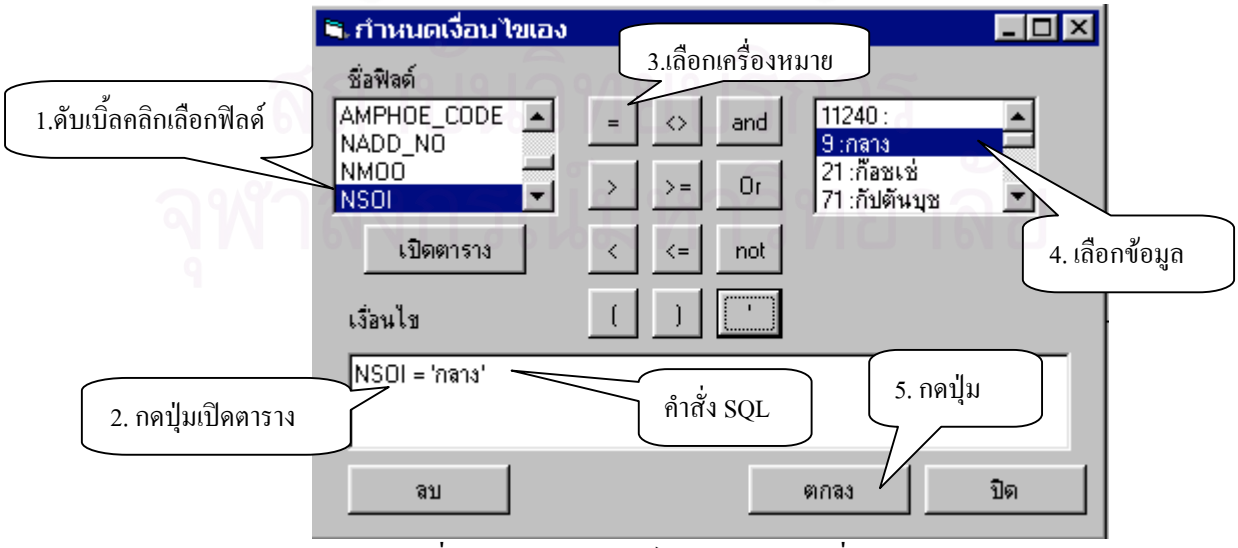
5. คลิกเลือกข้อมูลที่ต้องการแก้ไข (4) ข้อมูล “ที่อยู่” ทั้งหมดที่มีเงื่อนไขตรงกับข้อมูลที่ต้องการแก้ไขจะปรากฏในตารางข้างล่าง โดยดูในฟิลด์ ADDRESS เป็นตัวช่วยในการตัดสินใจ ในการแก้ไขสามารถทำได้ 2 วิธีคือ

- คลิกปุ่ม [แก้] ในกรณีที่ต้องการแก้ไขข้อมูลทั้งหมดในตารางในครั้งเดียว เมื่อคลิกแล้วจะปรากฏหน้าต่างดังรูปที่ 4.21 ใส่ข้อมูลที่ถูกต้องลงไปในช่องว่างแล้วคลิกปุ่ม OK

รูปที่ 4.21 แสดงวินโดว์สำหรับแก้ข้อมูลที่ถูกเลือก

- แก้ข้อมูลในตารางข้างล่าง โดยในตารางจะมีฟิลด์ทั้งหมด 9 ฟิลด์ ที่สามารถแก้ไขโดยพิมพ์ในตารางได้ในทันที

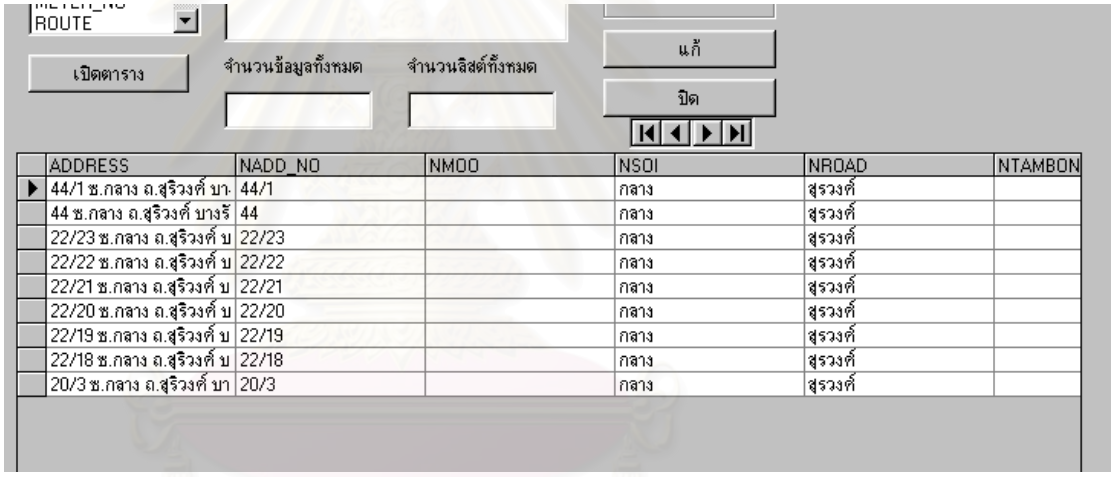
6. ผู้ใช้สามารถกำหนดเงื่อนไขในการเลือกข้อมูลที่ต้องการแก้ไข ได้โดยคลิกเลือกตาราง ก่อนที่จะกดปุ่ม [ผู้ใช้กำหนดเงื่อนไขเอง] จะปรากฏวินโดว์การกำหนดเงื่อนไขขึ้นดังรูปที่ 4.22



รูปที่ 4.22 แสดงวินโดว์สำหรับกำหนดเงื่อนไข

โดยมีวิธีกำหนดเงื่อนไขดังนี้

- ดับเบิ้ลคลิกเลือกชื่อฟิลด์ (1) ชื่อฟิลด์จะปรากฏในช่องเงื่อนไข ตัวอย่าง NSOI ในภาพข้างบน
- คลิกปุ่ม เปิดตาราง (2) ข้อมูลในฟิลด์ที่เลือกจะปรากฏในช่องลิสต์ขวามือ
- คลิกปุ่มเลือกเครื่องหมายเปรียบเทียบ(3) เช่น เครื่องหมาย =
- ถ้าข้อมูลเป็นข้อความให้ใส่เครื่องหมาย ‘ ไว้หน้าและหลังข้อความนั้น
- ดับเบิ้ลคลิกเลือกข้อความ(4) ในช่องลิสต์ ที่ช่องทางขวามือ ตัวอย่างเช่น ‘กลาง’ ในรูปที่ 4.22
- เมื่อกำหนดเงื่อนไขเสร็จแล้วคลิกปุ่ม ตกลง (5) จะปรากฏข้อมูลที่ถูกเลือกตามเงื่อนไขดังรูปที่ 4.23



ADDRESS	NADD_NO	NMOO	NSOI	NROAD	NTAMBON
44/1 ซ.กลาง ถ.สุริวงศ์ บา	44/1		กลาง	สุรวงศ์	
44 ซ.กลาง ถ.สุริวงศ์ บางรั	44		กลาง	สุรวงศ์	
22/23 ซ.กลาง ถ.สุริวงศ์ บ	22/23		กลาง	สุรวงศ์	
22/22 ซ.กลาง ถ.สุริวงศ์ บ	22/22		กลาง	สุรวงศ์	
22/21 ซ.กลาง ถ.สุริวงศ์ บ	22/21		กลาง	สุรวงศ์	
22/20 ซ.กลาง ถ.สุริวงศ์ บ	22/20		กลาง	สุรวงศ์	
22/19 ซ.กลาง ถ.สุริวงศ์ บ	22/19		กลาง	สุรวงศ์	
22/18 ซ.กลาง ถ.สุริวงศ์ บ	22/18		กลาง	สุรวงศ์	
20/3 ซ.กลาง ถ.สุริวงศ์ บา	20/3		กลาง	สุรวงศ์	

รูปที่ 4.23 แสดงตัวอย่างการเลือกโดยกำหนดเงื่อนไขเอง

- ถ้าพิมพ์กำหนดเงื่อนไขผิดให้คลิกปุ่ม ลบ จะลบเงื่อนไขทั้งหมดทิ้ง เพื่อกำหนดเงื่อนไขใหม่

7. ถ้าต้องการออกจากฟังก์ชันนี้ ให้คลิกปุ่ม ปิด

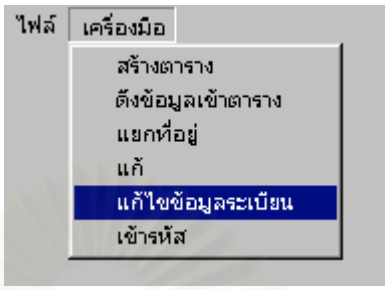
4.3.6 ฟังก์ชันแก้ไขข้อมูลระเบียน

วัตถุประสงค์

ฟังก์ชันแก้ไขข้อมูลระเบียน จะแก้ไขข้อมูลในตารางได้ทั้งหมด สามารถเพิ่มหรือลบข้อมูลระเบียนได้

การใช้งาน

1. แก้ไขข้อมูลในในตาราง โดยกดปุ่ม แก้ไขข้อมูลระเบียน หรือเลือกเมนูเครื่องมือ -> แก้ไขข้อมูลระเบียน ดังรูปที่ 4.24



รูปที่ 4.24 แสดงเมนูแก้ไขข้อมูลระเบียน

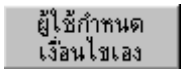
2. เมื่อคลิกเลือกแล้วจะปรากฏวินโดว์ขึ้นมาดังรูปที่ 4.25 แล้วทำการเลือกตาราง

(1) ที่ต้องการแก้ไข ที่มุมด้านซ้ายล่าง จะปรากฏตารางที่เลือกในหน้าต่างด้านบน

รูปที่ 4.25 แสดงวินโดว์สำหรับแก้ไขข้อมูลระเบียน

3. สามารถทำการเพิ่มข้อมูลระเบียนได้โดย พิมพ์ข้อมูลที่บรรทัดล่างสุด และสามารถลบข้อมูลระเบียนได้โดยคลิกที่ช่องสี่เหลี่ยมทางด้านซ้ายมือสุด จะปรากฏเครื่องหมายสามเหลี่ยมขึ้นแล้วกดปุ่ม Delete ที่แป้นพิมพ์ ดังที่แสดงในรูปที่ 4.26

รูปที่ 4.26 แสดงวินโดว์สำหรับลบข้อมูลระเบียน

4. ถ้าต้องการเลือกข้อมูลที่ผู้ใช้กำหนดเงื่อนไขเองมาแก้ไขให้คลิกปุ่ม 

โดยมีวิธีใช้เช่นเดียวกับในฟังก์ชันแก้

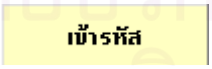
5. ถ้าต้องการออกจากฟังก์ชันนี้ ให้คลิกปุ่ม

4.3.7 ฟังก์ชันเข้ารหัส

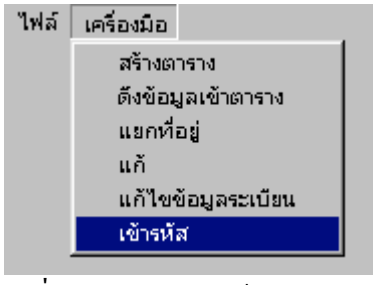
วัตถุประสงค์

ฟังก์ชันเข้ารหัส จะนำข้อมูลในรูปแบบกลางไปเปรียบเทียบใน LUT เพื่อนำรหัสที่ได้มา สร้างเป็นรหัสใหม่ (NEWID) ที่ให้ในการเชื่อมโยง

การใช้งาน

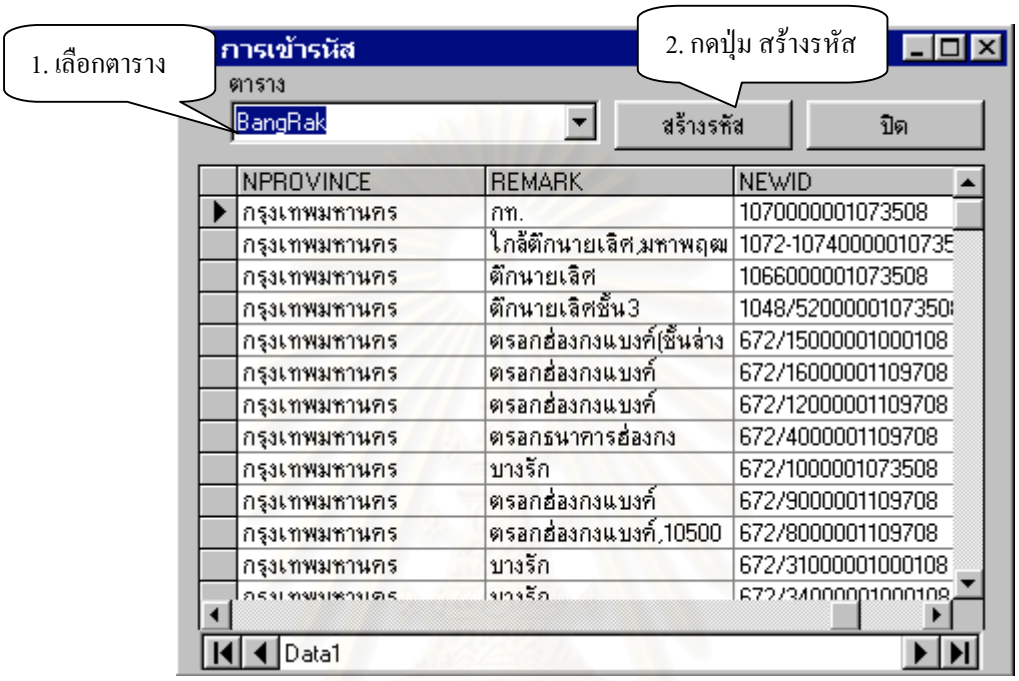
1. เข้ารหัสข้อมูล โดยกดปุ่ม 

หรือ เลือกเมนูเครื่องมือ -> เข้ารหัส ดังรูปที่ 4.27



รูปที่ 4.27 แสดงเมนูเข้ารหัส

2. เมื่อคลิกเลือกแล้วจะปรากฏวินโดว์ขึ้นมาดังรูปที่ 4.28 แล้วทำการเลือกตารางที่ต้องการแก้ไข ที่มุมด้านซ้ายบน จากนั้นคลิกปุ่ม สร้างรหัส เพื่อทำการสร้างรหัสในช่อง IDNEW เพื่อใช้ในการเชื่อมข้อมูลในกระบวนการต่อไป



รูปที่ 4.28 แสดงวินโดว์สำหรับเข้ารหัส

3. ถ้าต้องการออกจากฟังก์ชันนี้ ให้คลิกปุ่ม ปิด

4.3.8 ฟังก์ชันปิดการทำงานของโปรแกรม

วัตถุประสงค์

ฟังก์ชันปิดการทำงานของโปรแกรม จะหยุดการทำงานของโปรแกรมทั้งหมด แล้วออกจากโปรแกรม

การใช้งาน

1. กดปุ่ม ปิดโปรแกรม เมื่อต้องการหยุดการทำงานของโปรแกรมทั้งหมด

4.4 การจัดทำ Library

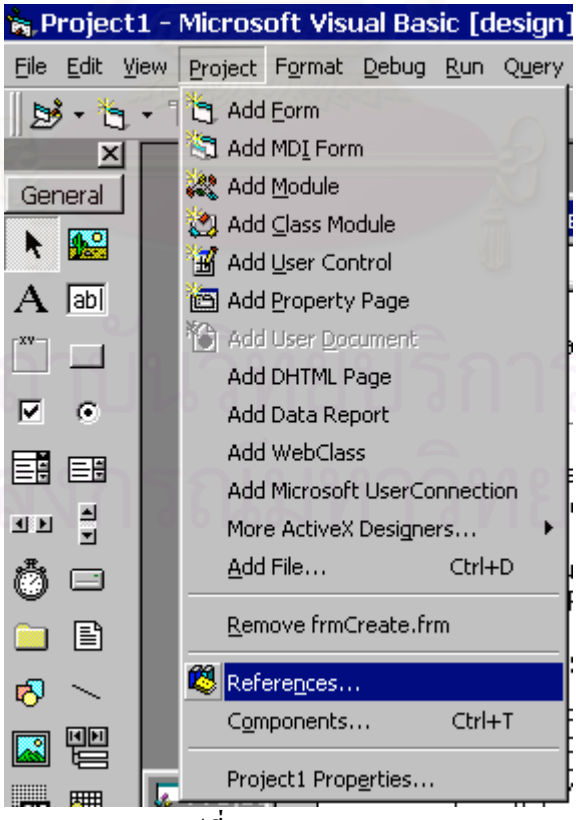
การจัดทำ Library เพื่อวัตถุประสงค์ในการนำไปใช้งาน ดังต่อไปนี้

- สามารถนำ Library ที่สร้างขึ้นมาในเครื่องมือพัฒนาแอพพลิเคชั่นหนึ่งไปใช้ในเครื่องมือพัฒนาแอพพลิเคชั่นอื่นก็ได้ที่สนับสนุนเทคโนโลยี ActiveX เช่น MS Visual C++, Borland Delphi, แอพพลิเคชั่นใน Microsoft Office 97 เป็นต้น
- ทำให้การพัฒนาแอพพลิเคชั่นทำได้ง่าย และรวดเร็วมากยิ่งขึ้น ด้วยการใช้ Library ที่ได้สร้างขึ้นมาเอง หรือหาได้จากนักพัฒนาคนอื่นๆ ที่ได้สร้าง Library ที่ตรงกับความต้องการไว้แล้ว มาร่วมกันสร้างแอพพลิเคชั่นขึ้น
- ง่ายต่อการบำรุงรักษา เนื่องจากในการสร้าง Library มีจำนวนบรรทัดของโปรแกรมน้อยกว่าในโปรแกรมทั่วไป จึงทำความเข้าใจได้ง่ายกว่าในโปรแกรมทั่วไป
- ง่ายต่อการแจกจ่าย โดยที่ผู้ที่นำไปใช้จะไม่เห็นรหัสต้นฉบับที่เราใช้สร้างคอนโทรล

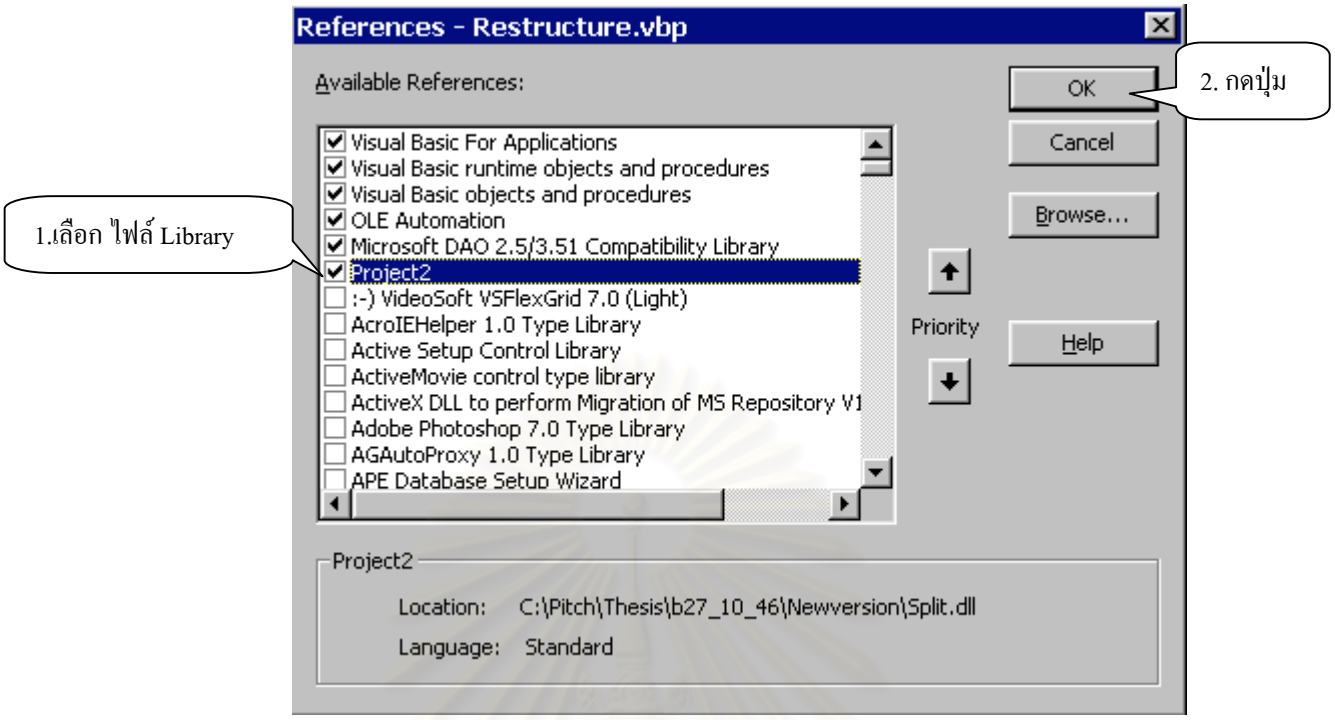
โดยมีตัวอย่าง ในภาคผนวก ก รหัสโปรแกรม ส่วนที่ทำเป็นไฟล์ Library

วิธีการเรียกใช้ไฟล์ Library

1. เลือก เมนู Project -> References ดังรูปที่ 4.29 แล้วจะปรากฏวินโดว์ ในรูปที่ 4.30



รูปที่ 4.29 แสดงเมนู References



รูปที่ 4.30 การเลือกไฟล์ Library

2. กดปุ่ม OK โปรแกรมจะทำการเพิ่ม Library เข้าไปในโปรแกรมเพื่อให้ผู้ใช้สามารถเลือกใช้ได้

4.5 กรณีศึกษาอื่นๆ

การใช้โปรแกรมนี้กับฐานข้อมูลอื่นๆ ที่ไม่ได้มีการศึกษานั้นสามารถทำได้ แต่จะต้องมีการแก้ไขโครงสร้างข้อมูลในฐานข้อมูลนั้นก่อนเพื่อให้สามารถแยกที่อยู่ แก้ไขข้อมูลต่างๆ และเข้ารหัสได้

4.5.1 กรณีที่จัดเก็บข้อมูล “ที่อยู่” ในฟิลด์เดียว

ให้เปลี่ยนชื่อฟิลด์ที่เก็บข้อมูล “ที่อยู่” เป็น ADDRESS แล้วใช้ฟังก์ชันสร้างตารางเพื่อเพิ่มฟิลด์แยกที่อยู่ ก็จะสามารถใช้ฟังก์ชันต่างๆ ของโปรแกรมได้เช่น ฟังก์ชันแยกที่อยู่ แก้ แก้ไขข้อมูลระเบียน และเข้ารหัส

หมายเหตุ เนื่องจากรหัสอำเภอของแต่ละหน่วยงานแตกต่างกัน สำหรับหน่วยงานอื่นๆ ควรตรวจสอบก่อนว่าใช้รหัสตรงกับหน่วยงานของการไฟฟ้า หรือสำนักงานประกันสังคม เมื่อพบว่าตรงกับหน่วยงานใดแล้วก็ให้เลือกหน่วยงานนั้นในการแยกที่อยู่

4.5.2 กรณีที่จัดเก็บข้อมูล“ที่อยู่” แยกฟิลด์

ให้เปลี่ยนชื่อฟิลด์ที่เก็บข้อมูลบ้านเลขที่เป็น ADDR เปลี่ยนชื่อฟิลด์ที่เก็บข้อมูลหมู่เป็น MOO เปลี่ยนชื่อฟิลด์ที่เก็บข้อมูลซอยเป็น SOI เปลี่ยนชื่อฟิลด์ที่เก็บข้อมูลถนนเป็น ROAD เปลี่ยนชื่อฟิลด์ที่เก็บข้อมูลตำบลเป็น TAMBON แล้วใช้ฟังก์ชันสร้างตารางเพื่อเพิ่มฟิลด์แยกที่อยู่ ก็จะสามารถใช้ฟังก์ชันต่างๆ ของโปรแกรมได้เช่น ฟังก์ชันแยกที่อยู่ แก้ แก้ไขข้อมูลระเบียน และเข้ารหัส

หมายเหตุ เนื่องจากรหัสอำเภอของแต่ละหน่วยงานแตกต่าง สำหรับหน่วยงานอื่นๆ ควรตรวจสอบก่อนว่าใช้รหัสตรงกับหน่วยงานของกรมสรรพากรหรือไม่ เมื่อพบว่าตรงกันก็ให้เลือกหน่วยงานนั้นในการแยกที่อยู่

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

ผลการทดสอบ

ในบทนี้จะกล่าวถึงวิธีการและผลการทดสอบขั้นตอนการใช้โปรแกรมทำการปรับปรุงข้อมูลให้มีความถูกต้องมากขึ้น เพื่อเชื่อมโยงฐานข้อมูลจากหน่วยงานต่างๆ เข้าด้วยกัน โดยจะทำการแก้ไขข้อมูลที่มีอยู่แล้วในฐานข้อมูลต่างๆ แล้วทำมาเชื่อมโยงฐานข้อมูลเข้าด้วยกัน

5.1 การปรับปรุงฐานข้อมูลต่างๆ เพื่อการเชื่อมโยงฐานข้อมูล

ก่อนการเชื่อมโยงฐานข้อมูล ต้องทำการแก้ไขข้อมูลที่มีความผิดพลาดและปรับปรุงข้อมูลให้ถูกต้องก่อนเพื่อให้ผลการเชื่อมโยงฐานข้อมูลมีความถูกต้องมากยิ่งขึ้น โดยมีวิธีแก้ไขและปรับปรุงข้อมูลในฟิลด์ต่างๆ ดังนี้

1. การปรับปรุงข้อมูลในฟิลด์บ้านเลขที่(NADD_NO) แก้ไขข้อมูลที่มีตัวอักษร โดยให้ใช้สัญลักษณ์แทน เช่น

1 และ 184	ใช้สัญลักษณ์แก้เป็น	1,184 (โดยสามารถแก้ไขได้ด้วยมือ)
1’2’3	ใช้สัญลักษณ์แก้เป็น	1,2,3 (โปรแกรมจะแก้ไขอัตโนมัติ)
15/10 ถึง 15/12	ใช้สัญลักษณ์แก้เป็น	15/10-12 (โดยสามารถแก้ไขได้ด้วยมือ)
15ชั้น5	ให้ตัดคำ (**Ctrl + x**) “ชั้น5” ไปวาง (**Ctrl + v**) ไว้ในฟิลด์ REMARK (โดยสามารถแก้ไขได้ด้วยมือ)	

2. การปรับปรุงข้อมูลในฟิลด์หมู่(NMOO) แก้ไขข้อมูลให้เป็นตัวเลขทั้งหมด ส่วนที่ไม่ใช่จะใช้ตัดคำไปไว้ในฟิลด์ REMARK

ในส่วนนี้จะไม่พบเพราะโปรแกรมจะแก้ไขอัตโนมัติ

3. การปรับปรุงข้อมูลในฟิลด์ซอย (NSOI) และ ถนน(NROAD) แก้ไขข้อมูลให้ถูกต้องตามที่มีใน LUT ของซอยและถนน (Lutsrd) เช่น

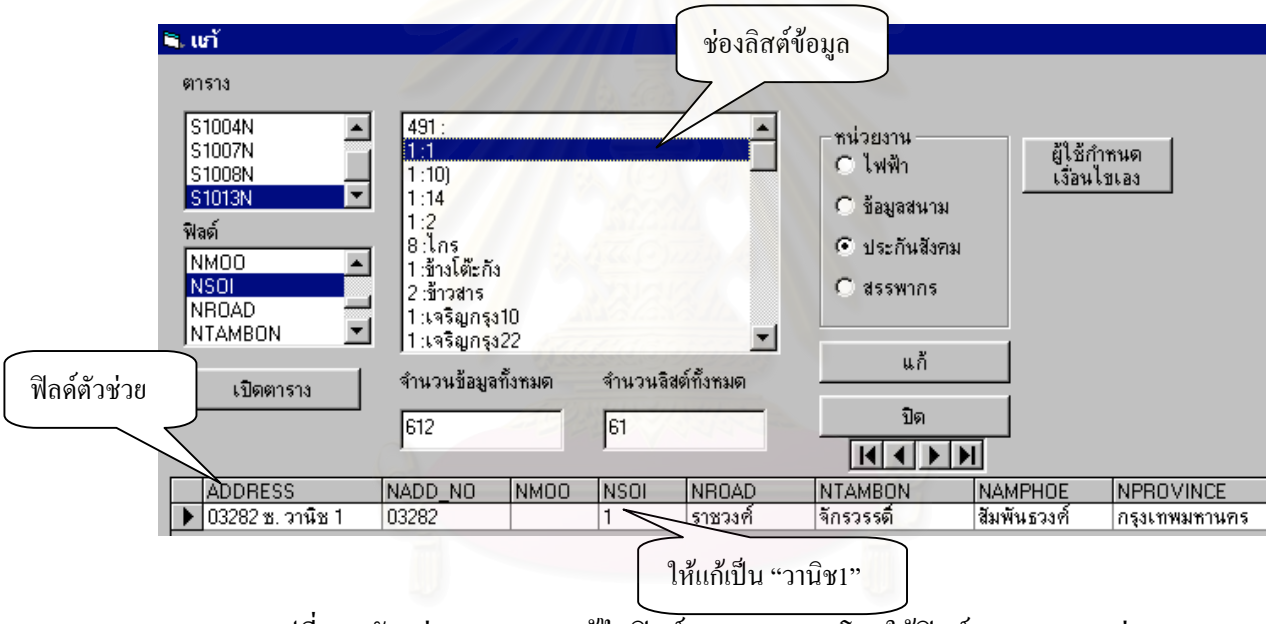
ไกรทามาศ	ให้แก้เป็น	ไกรฑามาตย์
ตรอกชุ่ม	ให้แก้เป็น	ชุ่ม

เยาวพาณิช ให้แก้เป็น เยาวพานิช

เจริญกรุงตรงข้ามตลาดน้อย ให้แก้เป็น เจริญกรุง แล้วตัดคำ “ตรงข้ามตลาดน้อย” ไปไว้ในฟิลด์ REMARK

โดยการพิมพ์ LUT ของซอยและถนน ในเขตที่เป็นพื้นที่ศึกษามาตรวจสอบก่อนเพื่อทราบชื่อซอยและถนนที่ถูกต้อง ซึ่งใน LUT ของซอยและถนน จะมีทั้งชื่อที่ใช้ในราชการและชื่อสามัญที่เรียกกันในท้องถิ่น ดังแสดงในภาคผนวก ข Look Up Table ของซอยและถนนในเขตสัมพันธวงศ์

บางครั้งข้อมูลที่มีอยู่ในฟิลด์ที่ทำการแก้ไขมีข้อมูลที่ไม่เพียงพอให้ทราบได้ว่าข้อมูลที่ถูกต้องเป็นอย่างไรจึงจำเป็นที่ต้องใช้ข้อมูลในฟิลด์อื่นมาช่วยในการแก้ไข ในการแก้ไขข้อมูลหน่วยงานสำนักงานประกันสังคมและการไฟฟ้าจะมีฟิลด์ ADDRESS เป็นตัวช่วยในการแก้ไขข้อมูลดังในรูปที่ 5.1



รูปที่ 5.1 ตัวอย่างแสดงการแก้ไขฟิลด์ซอยและถนนโดยใช้ฟิลด์ ADDRESS ช่วย

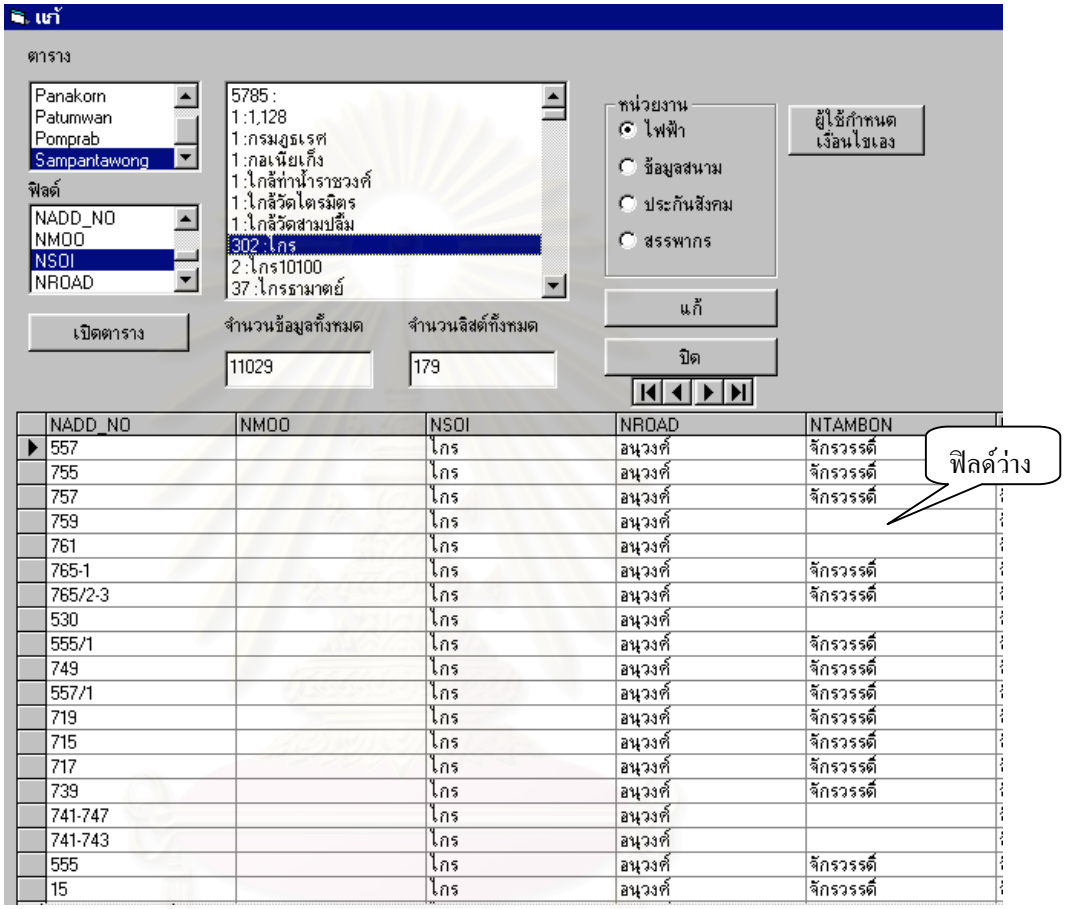
จากรูปที่ 5.1 ข้อมูลในช่องลิสต์ข้อมูลมีค่าเป็น “1:1” หมายความว่า ข้อมูล “1” มีความถี่ของข้อมูลเท่ากับ 1 โดยตัวเลขหน้าเครื่องหมาย “:” คือความถี่ และตัวเลขหลังเครื่องหมาย “:” คือตัวข้อมูล ซึ่งถ้าไม่มีฟิลด์ ADDRESS เป็นตัวช่วยก็จะไม่ทราบได้เลยว่า ข้อมูล “1” เป็นซอยใด มีวิธีการแก้ไขได้โดยใช้ข้อมูลในฟิลด์ ADDRESS มาใส่ในฟิลด์ซอย แก้ “1” เป็น “วานิช1”

4. การปรับปรุงข้อมูลในฟิลด์ตำบล (NTAMBON) และ อำเภอ (NAMPHOE) แก้ไขข้อมูลให้ถูกต้องตามที่มีใน LUT ของแขวง ตัวอย่างเช่น

จักรวรรด ให้แก้เป็น จักรวรรดิ

ตลาดย้อย ให้แก้เป็น ตลาดน้อย

ส่วนข้อมูลที่เป็นฟิลด์ว่าง หาข้อมูลมาเติมให้ครบได้จากการสังเกตข้อมูลฟิลด์ข้างเคียง หรือ เรคคอร์ดข้างเคียง จากรูปที่ 5.2 เมื่อเลือกข้อมูล “ตรอกไกร” ในฟิลด์ชื่อซอย จะมีข้อมูลในฟิลด์ตำบลเป็น “จักรวรรดิ” ให้คัดลอกในเรคคอร์ดข้างเคียงไปวางในเรคคอร์ดที่เป็นช่องว่าง



รูปที่ 5.2 ตัวอย่างแสดงการแก้ไขฟิลด์ตำบลให้ครบถ้วนโดยใช้ฟิลด์ SOI ช่วย

หลังจากการปรับปรุงฐานข้อมูลตามวิธีดังกล่าวแล้ว ทำให้ข้อมูลมีความถูกต้องครบถ้วนมากยิ่งขึ้น

5.2 ผลของการปรับปรุงความครบถ้วนฐานข้อมูล

การปรับปรุงฐานข้อมูลจะทำให้ความครบถ้วนของฐานข้อมูลมีมากขึ้น โดยจะทำการแก้ไขให้จำนวนเรคคอร์ดที่ไม่มีข้อมูล(ฟิลด์ว่าง) ให้น้อยลง ซึ่งข้อมูลที่มีฟิลด์ว่างมากๆ จะทำให้ผลการเชื่อมโยงข้อมูลน้อยลง ในการทำการศึกษาครั้งนี้ มีความคาดหวังว่าเมื่อทำการปรับปรุงความ

ครบถ้วนแล้วจะทำให้ผลลัพธ์การเชื่อมโยงฐานข้อมูลมีมากขึ้น โดยจะเลือกพื้นที่ศึกษาเขตสัมพันธวงศ์ เป็นพื้นที่ในการปรับปรุงโดยทำการปรับปรุงฐานข้อมูลการใช้ไฟฟ้า ของการไฟฟ้า ข้อมูลประกันสังคมของสำนักงานประกันสังคม ข้อมูลอาคารและข้อมูลเลขประจำตัวผู้เสียภาษีของกรมสรรพากร ได้ผลดังนี้

จากตารางที่ 5.1 ผลการปรับปรุงข้อมูลการใช้ไฟฟ้า ของการไฟฟ้า จากจำนวนข้อมูลทั้งหมด 11,029 เรคคอร์ด ในฟิลด์ NADD_NO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 858 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 719 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 1.26% ฟิลด์ NMOO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 11,028 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 11,028 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0% เนื่องจากเป็นข้อมูลในเขตกรุงเทพฯ ส่วนใหญ่แล้วจะไม่มีเลขหมู่ ฟิลด์ NSOI มีจำนวนฟิลด์ว่างก่อนปรับปรุง 5,785 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 5,388 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 3.60% ฟิลด์ NROAD มีจำนวนฟิลด์ว่างก่อนปรับปรุง 802 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 516 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 2.59% ฟิลด์ NTAMBON มีจำนวนฟิลด์ว่างก่อนปรับปรุง 8,610 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 364 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 74.77% ที่มีเปอร์เซ็นต์ในการปรับปรุงสูงเนื่องจากการบันทึกข้อมูลตำบลของการไฟฟ้านั้นผู้นำเข้าไม่ได้ใส่ตัวระบุ เช่น ต. หรือ เขต ทำให้การแยกข้อมูลไม่สามารถจำแนกข้อมูลไว้ในฟิลด์ NTAMBON แต่จะนำไปไว้ใน ฟิลด์ REMARK แทนทำให้ข้อมูลไม่สูญหาย ผู้ใช้สามารถปรับปรุงข้อมูลใน ฟิลด์ REMARK มากรอกข้อมูลให้ครบถ้วนโดยมีเปอร์เซ็นต์การปรับปรุงในอัตราที่สูง

ตารางที่ 5.1 ผลการปรับปรุงข้อมูลการใช้ไฟฟ้า (การไฟฟ้า) จำนวน 11,029 เรคคอร์ด

	NADD_NO	NMOO	NSOI	NROAD	NTAMBON
จำนวนฟิลด์ว่างก่อนปรับปรุง	858	11028	5785	802	8610
จำนวนฟิลด์ว่างหลังปรับปรุง	719	11028	5388	516	364
เปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้น	1.26%	0.00%	3.60%	2.59%	74.77%

จากตารางที่ 5.2 ผลการปรับปรุงข้อมูลประกันสังคม ของสำนักงานประกันสังคม จากจำนวนข้อมูลทั้งหมด 612 เรคคอร์ด ในฟิลด์ NADD_NO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 8 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 2 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0.98% ฟิลด์ NMOO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 612 เรคคอร์ด หลังจาก

ปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 612 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0% เนื่องจากเป็นข้อมูลในเขตกรุงเทพฯ ส่วนใหญ่แล้วจะไม่มีเลขหมู่ ฟิลด์ NSOI มีจำนวนฟิลด์ว่างก่อนปรับปรุง 491 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 405 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 14.05% ฟิลด์ NROAD มีจำนวนฟิลด์ว่างก่อนปรับปรุง 51 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 4 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 7.68% ฟิลด์ NTAMBON มีจำนวนฟิลด์ว่างก่อนปรับปรุง 50 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 4 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 7.52%

ตารางที่ 5.2 ผลการปรับปรุงข้อมูลประกันสังคม (สำนักงานประกันสังคม) จำนวน 612 เรคคอร์ด

	NADD_NO	NMOO	NSOI	NROAD	NTAMBON
จำนวนฟิลด์ว่างก่อนปรับปรุง	8	612	491	51	50
จำนวนฟิลด์ว่างหลังปรับปรุง	2	612	405	4	4
เปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้น	0.98%	0.00%	14.05%	7.68%	7.52%

จากตารางที่ 5.3 ผลการปรับปรุงข้อมูลอาคาร ของกรมสรรพากร จากจำนวนข้อมูลทั้งหมด 8,944 เรคคอร์ด ในฟิลด์ NADD_NO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 1,107 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 1,068 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0.44% ฟิลด์ NMOO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 8,944 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 8,944 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0% เนื่องจากเป็นข้อมูลในเขตกรุงเทพฯ ส่วนใหญ่แล้วจะไม่มีเลขหมู่ ฟิลด์ NSOI มีจำนวนฟิลด์ว่างก่อนปรับปรุง 5,314 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 4,953 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 4.04% ฟิลด์ NROAD มีจำนวนฟิลด์ว่างก่อนปรับปรุง 1,034 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 304 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 8.16% ฟิลด์ NTAMBON มีจำนวนฟิลด์ว่างก่อนปรับปรุง 85 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 39 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0.51%

ตารางที่ 5.3 ผลการปรับปรุงข้อมูลอาคาร (กรมสรรพกร) จำนวน 8,944 เรคคอร์ด

	NADD_NO	NMOO	NSOI	NROAD	NTAMBON
จำนวนฟิลด์ว่างก่อนปรับปรุง	1107	8944	5314	1034	85
จำนวนฟิลด์ว่างหลังปรับปรุง	1068	8944	4953	304	39
เปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้น	0.44%	0.00%	4.04%	8.16%	0.51%

จากตารางที่ 5.4 ผลการปรับปรุงข้อมูลผู้เสียภาษี ของกรมสรรพากร จากจำนวนข้อมูลทั้งหมด 6,765 เรคคอร์ด ในฟิลด์ NADD_NO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 20 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 18 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0.03% ฟิลด์ NMOO มีจำนวนฟิลด์ว่างก่อนปรับปรุง 2,847 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 2,847 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0% เนื่องจากเป็นข้อมูลในเขตกรุงเทพฯ ส่วนใหญ่จะไม่มีเลขหมู่ ฟิลด์ NSOI มีจำนวนฟิลด์ว่างก่อนปรับปรุง 4,746 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 4,257 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 7.23% ฟิลด์ NROAD มีจำนวนฟิลด์ว่างก่อนปรับปรุง 103 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 45 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 0.86% ฟิลด์ NTAMBON มีจำนวนฟิลด์ว่างก่อนปรับปรุง 510 เรคคอร์ด หลังจากปรับปรุงแล้วมีจำนวนฟิลด์ว่างเหลืออยู่ 56 เรคคอร์ด มีเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นคิดเป็น 6.71%

ตารางที่ 5.4 ผลการปรับปรุงข้อมูลผู้เสียภาษี ของกรมสรรพากร จำนวน 6,765 เรคคอร์ด

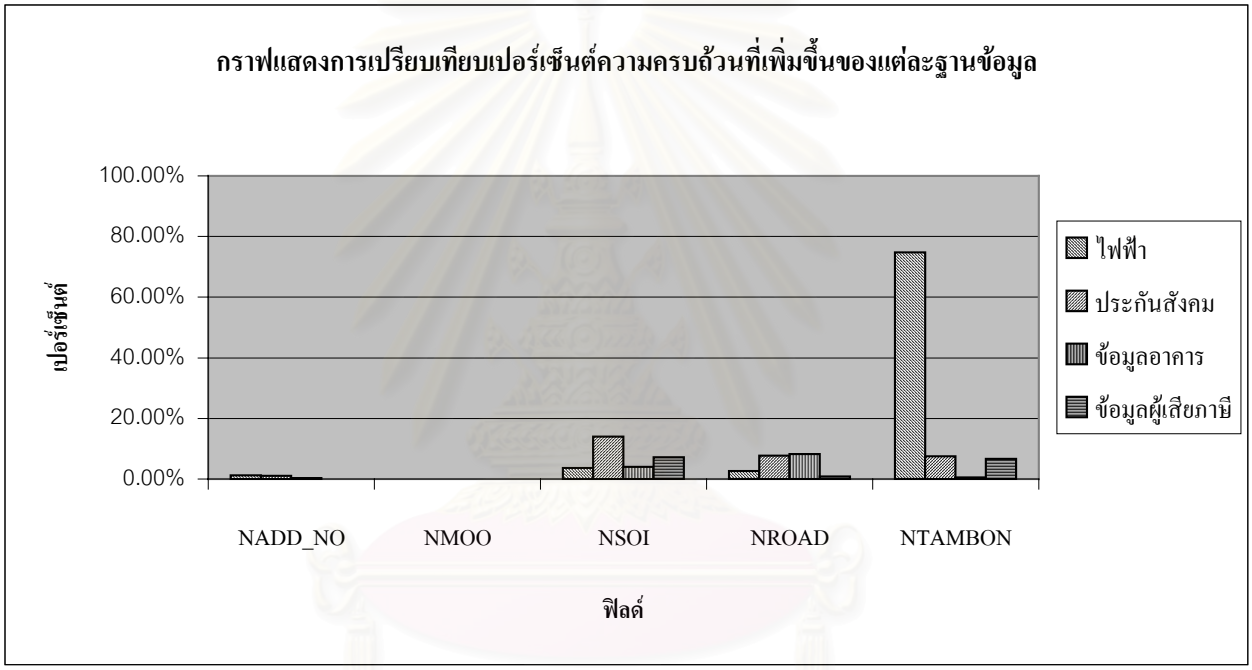
	NADD_NO	NMOO	NSOI	NROAD	NTAMBON
จำนวนฟิลด์ว่างก่อนปรับปรุง	20	2847	4746	103	510
จำนวนฟิลด์ว่างหลังปรับปรุง	18	2847	4257	45	56
เปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้น	0.03%	0.00%	7.23%	0.86%	6.71%

ในตารางที่ 5.1 - 5.4 จะแสดงจำนวนฟิลด์ว่างก่อนและหลังการปรับปรุงข้อมูลของในแต่ละฟิลด์ที่ทำการปรับปรุงข้อมูล และแสดงเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้น การปรับปรุงฟิลด์ต่างๆ สามารถทำได้โดยสังเกตข้อมูลในฟิลด์ที่เก็บชื่อผู้จดทะเบียน หรือ ชื่อบริษัทจดทะเบียน ของหน่วยงานอื่นที่มีชื่อเดียวกันเป็นฟิลด์ช่วย โดยดูว่ามีข้อมูลในฟิลด์ต่างๆ เป็นเท่าใดมีความเหมือนหรือ

ต่างกันอย่างไร แล้วทำการแก้ไขให้สอดคล้องกันเพื่อให้ข้อมูลมีความถูกต้องครบถ้วนมากขึ้น โดยเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นใช้สูตรการคำนวณดังนี้

$$\text{เปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้น} = \frac{\text{จำนวนฟิลด์ว่างก่อนปรับปรุง} - \text{จำนวนฟิลด์ว่างหลังปรับปรุง}}{\text{จำนวนข้อมูลทั้งหมด}} * 100$$

จากตารางที่ 5.1 - 5.4 สามารถทำกราฟเปรียบเทียบเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นในแต่ละหน่วยงานได้ดังรูป 5.3



รูปที่ 5.3 กราฟแสดงการเปรียบเทียบเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นของแต่ละฐานข้อมูล

จากรูปที่ 5.3 จะเห็นว่าใน**ผลการปรับปรุงในฟิลด์ NADD_NO** หลังจากทำการปรับปรุงแล้ว ในแต่ละหน่วยงานจะให้เปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นจะมีค่าใกล้เคียงกัน โดยจะอยู่ในช่วงระหว่าง 0.03% - 1.26% เนื่องจากในฟิลด์ NADD_NO จะเป็นฟิลด์ที่มีความถูกต้องอยู่แล้วจึงทำให้แก้ไข ได้ไม่มากนัก การหาค่าเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้น โดยดูได้จากตาราง 5.3 ข้อมูลของสำนักงานประกันสังคม

ผลการปรับปรุงในฟิลด์ NMOO เมื่อเปรียบเทียบเป็นเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นจะมีค่า 0% ทั้งหมดเนื่องจากข้อมูลที่ใช้ในการศึกษาเป็นข้อมูลในกรุงเทพฯ ซึ่งส่วนใหญ่จะไม่มีข้อมูลหมู่ จึงไม่มีการปรับปรุงข้อมูล

ผลการปรับปรุงในฟิลด์ NSOI เมื่อเปรียบเทียบเป็นเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นจะมีค่าที่ใกล้เคียงกัน การปรับปรุง ฟิลด์ NSOI สามารถทำได้โดยสังเกตข้อมูลในฟิลด์ที่เก็บชื่อผู้จดทะเบียน หรือ ชื่อบริษัทจดทะเบียน ของหน่วยงานอื่นที่มีชื่อเดียวกันว่ามีข้อมูลในฟิลด์ NSOI เป็นเท่าใด

ผลการปรับปรุงในฟิลด์ NROAD เมื่อเปรียบเทียบเป็นเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นจะมีค่าที่แตกต่างกันเนื่องจากจำนวนข้อมูลที่ไม่ครบถ้วนของแต่ละหน่วยงานมีจำนวนต่างกันในลักษณะเดียวกันผลการปรับปรุงในฟิลด์ NADD_NO การปรับปรุงฟิลด์ NROAD สามารถทำได้โดยสังเกตข้อมูลในฟิลด์ที่เก็บชื่อผู้จดทะเบียน หรือ ชื่อบริษัทจดทะเบียน ของหน่วยงานอื่นที่มีชื่อเดียวกันว่ามีข้อมูลในฟิลด์ NROAD เป็นเท่าใด หรือสังเกตจากฟิลด์ชื่อซอย

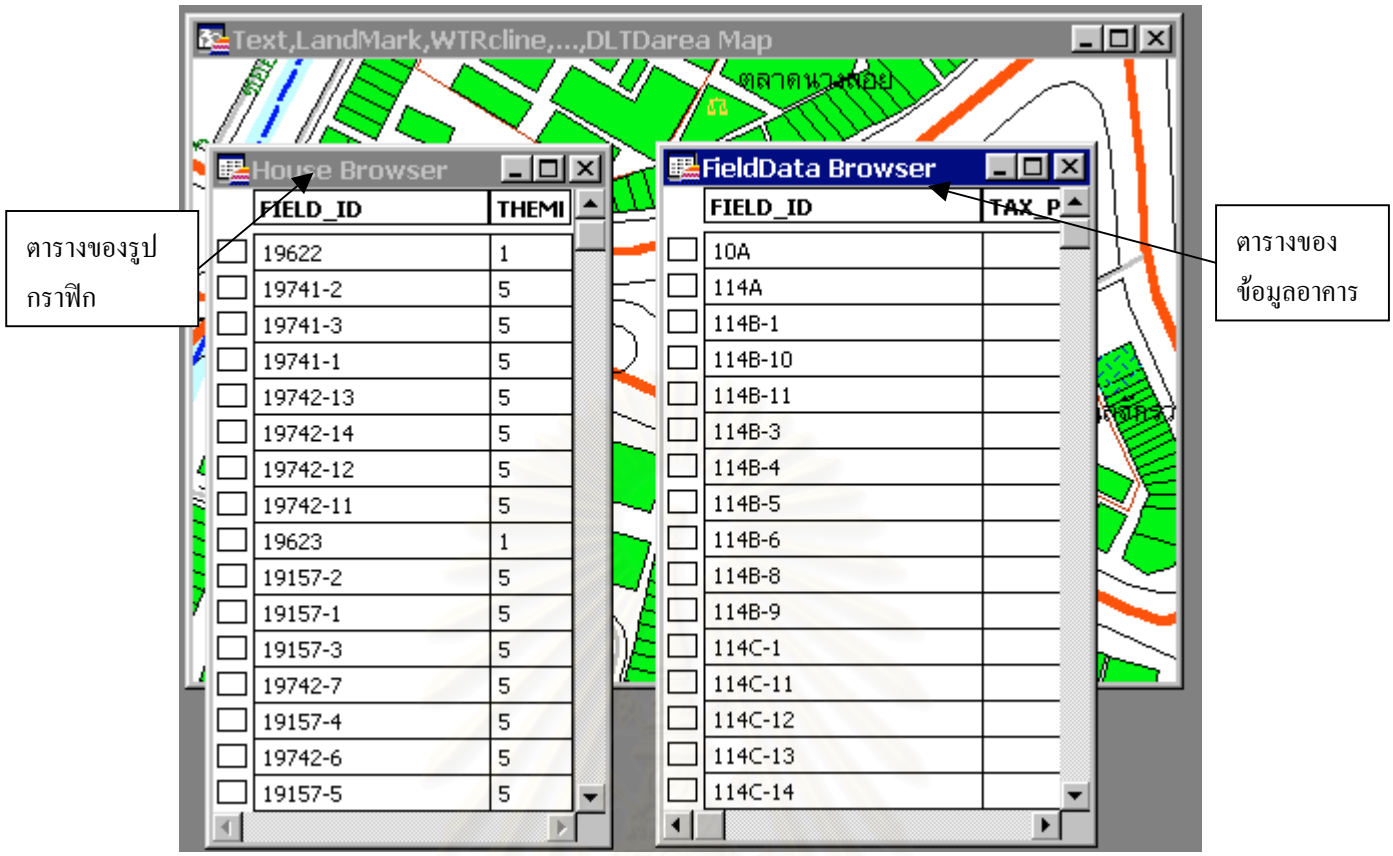
ผลการปรับปรุงในฟิลด์ NTAMBON เมื่อเปรียบเทียบเป็นเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นจะมีค่าที่แตกต่างกันเนื่องจากจำนวนข้อมูลที่ไม่ครบถ้วนของแต่ละหน่วยงานมีจำนวนต่างกัน ในหน่วยงานของการไฟฟ้าสามารถปรับปรุงได้ถึง 74% เนื่องจากการบันทึกข้อมูลของการไฟฟ้าจะบันทึกข้อมูลตำบล โดยข้อมูลส่วนใหญ่จะไม่มีตัวบ่งชี้ว่าเป็นเขต หรือตำบล โปรแกรมจึงนำข้อมูลที่ไม่สามารถจำแนกได้นี้ไปเก็บไว้ในฟิลด์ REMARK ในการปรับปรุงจึงนำข้อมูลที่อยู่ในฟิลด์ REMARK มาบันทึกเก็บไว้ในฟิลด์ NTAMBON ส่วนในหน่วยงานอื่นปรับปรุงโดยใช้ข้อมูลฟิลด์ต่างๆ มาเปรียบเทียบกัน

หลังการปรับปรุงความครบถ้วนของข้อมูล เมื่อสังเกตจำนวนเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นสะท้อนให้เห็นว่า โปรแกรมสามารถทำให้ตรวจพบข้อมูลที่ไม่ครบถ้วนกลับมาครบถ้วนได้โดยเฉพาะฟิลด์ NTAMBON ที่มีจำนวนเปอร์เซ็นต์ความครบถ้วนที่เพิ่มขึ้นที่สูงกว่าฟิลด์อื่นๆ ซึ่งจากการที่ได้ข้อมูลครบถ้วนขึ้นจะทำให้ผลการเชื่อมโยงข้อมูลเพิ่มขึ้นด้วยหรือไม่นั้น จะแสดงในหัวข้อถัดไป

5.3 การเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลอาคาร โดยใช้โปรแกรม MapInfo

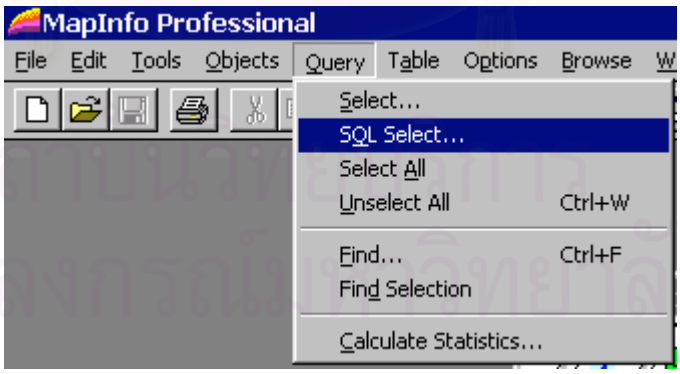
จุดประสงค์ที่สำคัญในการทำการศึกษาครั้งนี้ก็คือการเชื่อมข้อมูลต่างๆ กับข้อมูลอาคาร (ข้อมูลที่เชื่อมกับข้อมูลกราฟิกในระบบ GIS) โดยใช้โปรแกรม MapInfo เป็นตัวช่วยในการเชื่อม มีวิธีการเชื่อมดังต่อไปนี้

1. เปิดตารางที่จะทำการเชื่อมกันก่อนคือ ตาราง House (เป็นรูปกราฟิกของ Block อาคาร) และตาราง FieldData (เป็นข้อมูลอาคารที่ได้จากการสำรวจภาคสนาม) ดังรูปที่ 5.4

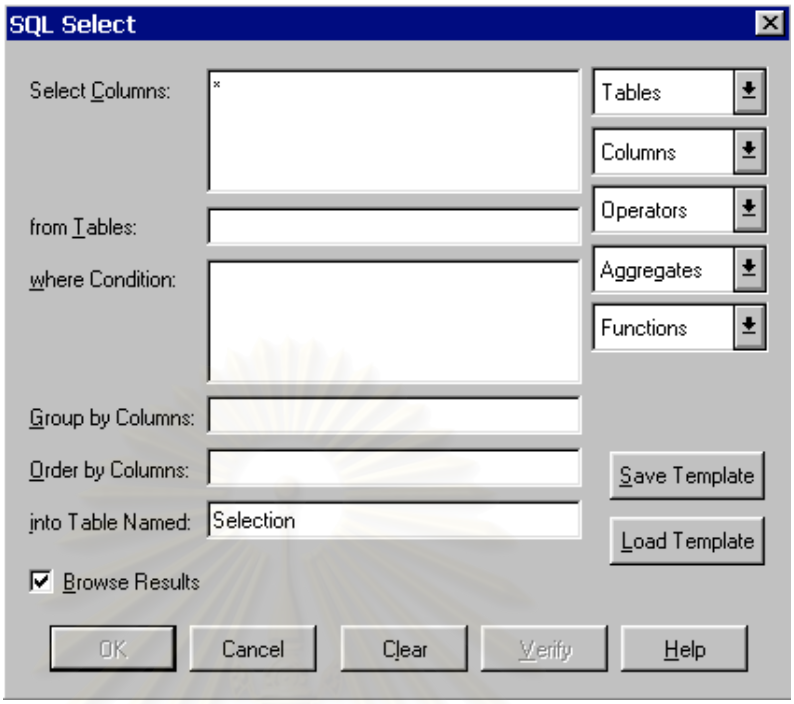


รูปที่ 5.4 ตัวอย่างแสดงการเปิดตารางเพื่อทำการเชื่อมข้อมูล

2. คลิกเลือกเมนู Query -> SQL Select… ดังรูปที่ 5.5 จะปรากฏ วินโดว์ดังรูปที่ 5.6

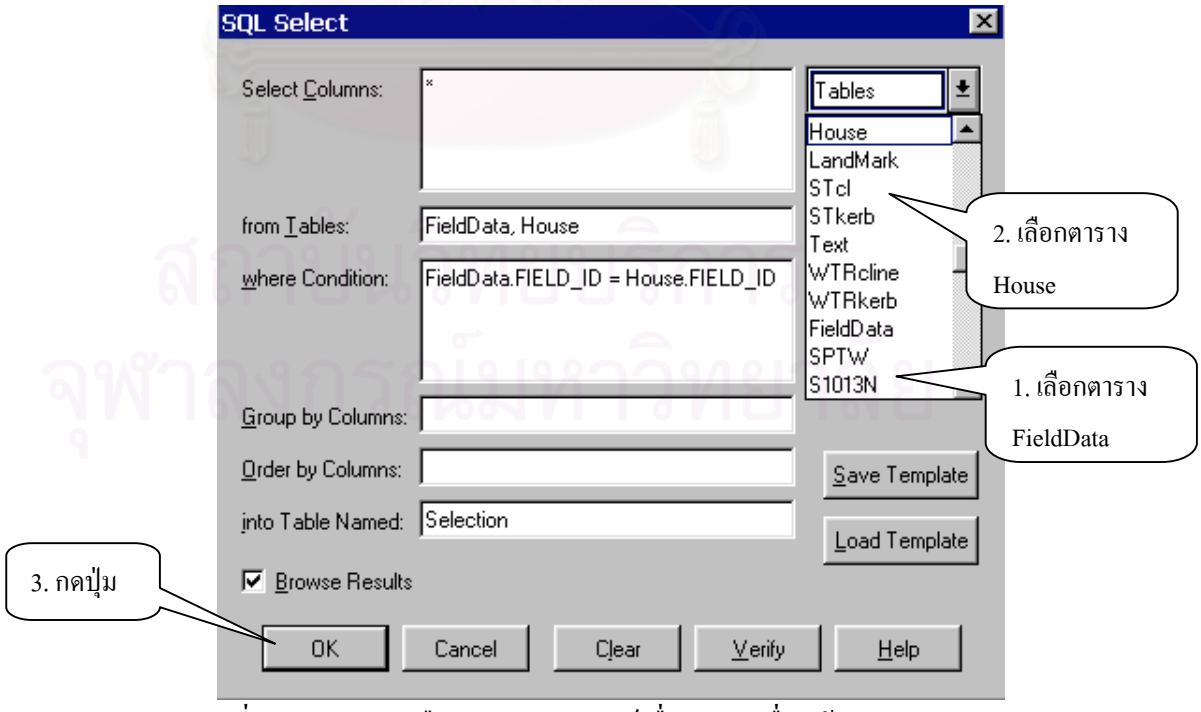


รูปที่ 5.5 ตัวอย่างแสดงเมนูการเชื่อมตาราง



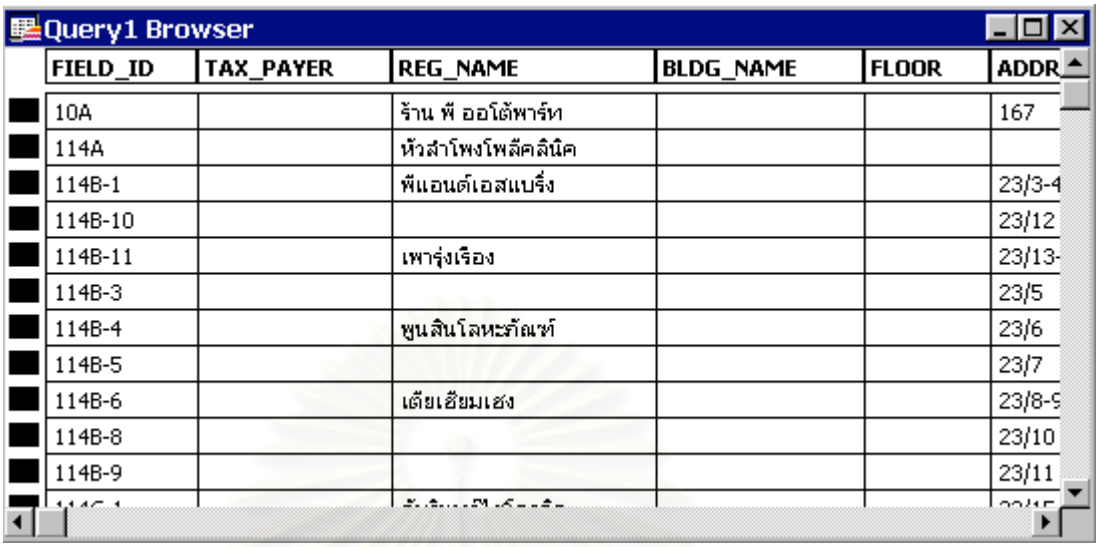
รูปที่ 5.6 แสดงวินโดว์สำหรับตั้งเงื่อนไขการเชื่อมข้อมูล

3. คลิกเลือก Tables เพื่อเลือกตารางที่จะทำการเชื่อมโยงกันโดยเลือกตาราง FieldData และ House จะได้ผลดังรูปที่ 5.7 โปรแกรมจะทำการเลือกฟิลด์ที่จะสามารถเชื่อมกันได้โดยอัตโนมัติ



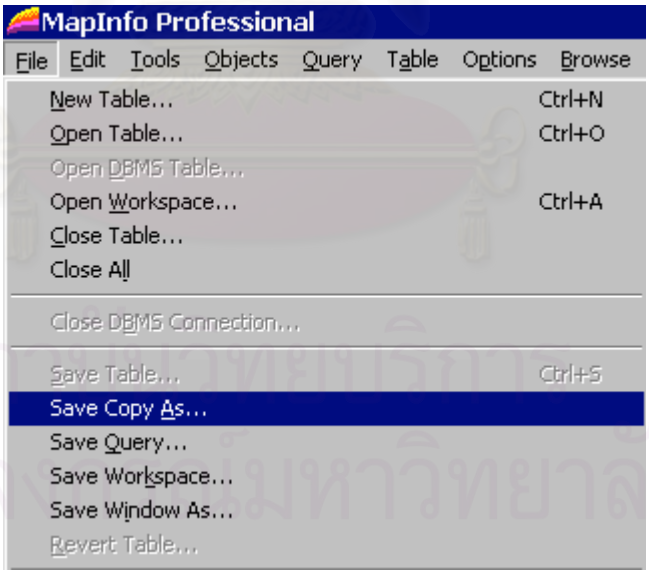
รูปที่ 5.7 แสดงการเลือกตารางและฟิลด์เพื่อทำการเชื่อมข้อมูล

4. จากนั้นกดปุ่ม OK ในรูปที่ 5.7 จะปรากฏวินโดว์ที่เป็นผลการเชื่อมข้อมูลในรูปที่ 5.8



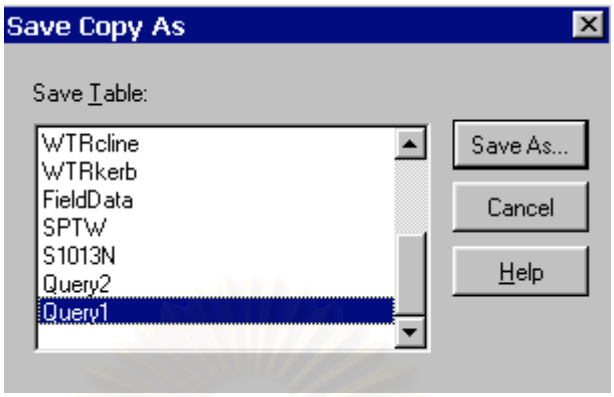
รูปที่ 5.8 แสดงข้อมูลที่สามารถเชื่อมข้อมูลได้

5. บันทึกเป็นข้อมูลกราฟิกโดยคลิกเมนู File -> Save Copy As… ดังรูปที่ 5.9 จะปรากฏวินโดว์ดังรูปที่ 5.10 โดยเลือกไฟล์ที่ต้องการจะบันทึกเป็นข้อมูลกราฟิก



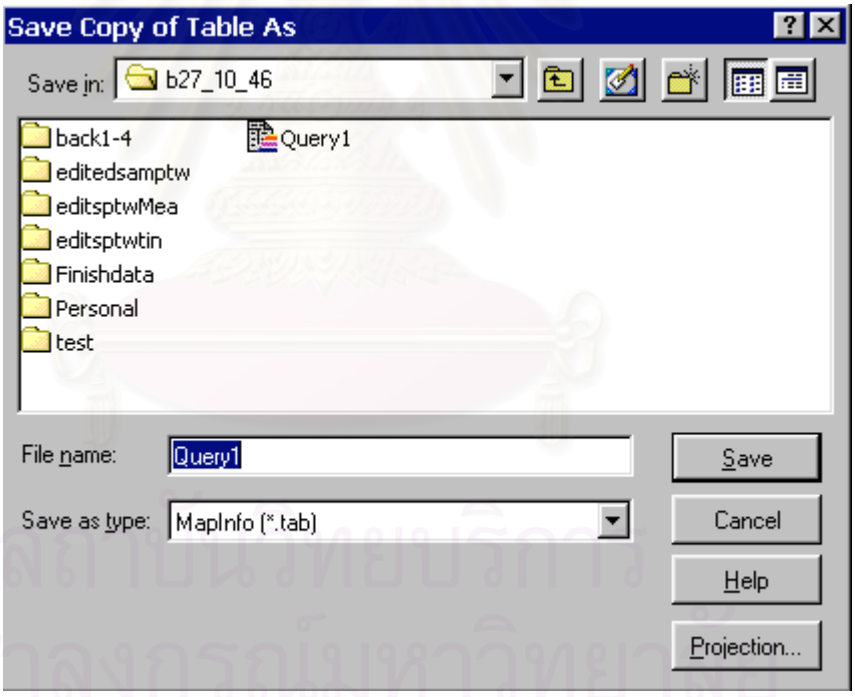
รูปที่ 5.9 ตัวอย่างแสดงเมนูการบันทึกข้อมูลเป็นกราฟิก

6. กดปุ่ม Save As… ในรูปที่ 5.10 จะปรากฏวินโดว์ดังรูปที่ 5.11



รูปที่ 5.10 ตัวอย่างวินโดว์เลือกไฟล์บันทึกข้อมูลเป็นกราฟิก

7. กดปุ่ม Save ในรูปที่ 5.11 โปรแกรมจะทำการบันทึกข้อมูลในรูปแบบ ไฟล์.tab



รูปที่ 5.11 ตัวอย่างวินโดว์บันทึกข้อมูลเป็นกราฟิก

จากกระบวนการข้างต้น จะได้ข้อมูลกราฟิกที่มีข้อมูลบรรยายอธิบาย Block อาคารต่างๆ

5.4 ผลของการเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลอาคาร

ตารางที่ 5.5 เปรียบเทียบผลการเชื่อมโยงข้อมูลแต่ละหน่วยงานกับข้อมูลอาคาร

สถานะของข้อมูล	การไฟฟ้า	ประกันสังคม	ข้อมูลผู้เสียภาษี	ระยะเวลา
ก่อนปรับปรุง				
- ใช้โปรแกรม	32%	21%	22%	2 วัน
หลังปรับปรุง				
- ใช้มือ	32%	59%	76%	2 เดือน
- ใช้โปรแกรม	40%	75%	78%	1 เดือน

5.4.1 ผลการเชื่อมโยงฐานข้อมูลก่อนทำการปรับปรุง

ผลของการเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลกราฟิกก่อนทำการปรับปรุงความครบถ้วนของข้อมูลจะแสดงในตารางที่ 5.5 โดยสามารถเชื่อมกับข้อมูลการใช้ไฟฟ้าได้ 32% ข้อมูลประกันสังคม 21% และข้อมูลผู้เสียภาษี 22% โดยใช้เวลาในการดำเนินการ 2 วัน

5.4.2 ผลการเชื่อมโยงฐานข้อมูลหลังทำการปรับปรุง

หลังจากที่ได้ทำการปรับแก้คำผิดและปรับปรุงข้อมูลให้มีความครบถ้วนมากขึ้นแล้ว การทำงานขั้นต่อไป คือ การนำฐานข้อมูลของหน่วยงานต่างๆ มาเชื่อมโยงกับข้อมูลอาคารโดยการนำฟิลด์ NEWID ที่ได้จากการเข้ารหัสของในแต่ละหน่วยงานมาทำการเชื่อมฐานข้อมูลเข้าด้วยกัน จากตารางที่ 5.5 เปรียบเทียบผลการเชื่อมโยงข้อมูลแต่ละหน่วยงานกับข้อมูลอาคาร โดยวิธีการใช้มือ จะเห็นได้ว่ามีเปอร์เซ็นต์การเชื่อมโยงในอัตราที่สูงคือประมาณ 76% แต่กว่าจะได้ความถูกต้องระดับนี้ต้องใช้เวลาในการจัดการกับข้อมูลเป็นเวลานานถึง 2 เดือนเพื่อเชื่อมโยงข้อมูลเพียงอย่างเดียว ยังไม่ได้ทำการแก้ไขข้อมูลให้ถูกต้อง แต่วิธีการใช้โปรแกรม ก่อนที่จะทำการปรับปรุง ข้อมูลมีความถูกต้อง 22% แต่ใช้เวลาเพียง 2 วันเท่านั้น โดยใช้เวลา 1 วันในการปรับแก้คำผิดและปรับปรุงข้อมูลและใช้เวลาในการเข้ารหัส 5 นาทีโดยการใช้โปรแกรมกับเครื่อง Pentium 4 แต่เมื่อใช้กับ AMD K6 แล้วจะใช้เวลาถึง 2 ชั่วโมง ซึ่งระยะเวลาในการทำงานของโปรแกรมนั้นขึ้นอยู่กับจำนวนของข้อมูลว่ามีมากน้อยเพียงใด

ในการแก้ไข ปรับปรุงและทำการเชื่อมโยงฐานข้อมูล เมื่อทำการวิเคราะห์ผลการเชื่อมโยงข้อมูลในตารางแล้ว พบว่าสาเหตุที่ทำให้การเชื่อมฐานข้อมูลมีเปอร์เซ็นต์ที่ต่ำ มีความผิดพลาดอีกประการหนึ่งนอกเหนือไปจาก ความผิดพลาดในการพิมพ์ และความไม่ครบถ้วนของข้อมูล คือ **ความไม่เข้ากันของข้อมูล**

ตัวอย่างเช่น ที่อยู่ของ ฉั่วฮะเส็ง ฟู้ดโปรดักส์

ตารางที่ 5.6 ตารางตัวอย่างเปรียบเทียบความไม่เข้ากันของฐานข้อมูล

	NADD_NO	NMOO	NSOI	NROAD	NTAMBON
ข้อมูลอาคาร	259			ราชวงศ์	จักรวรรดิ
ประกันสังคม	259		อิสรานุภาพ	เยาวราช	จักรวรรดิ

จากตารางที่ 5.6 เป็นที่อยู่ของ ฉั่วฮะเส็ง ฟู้ดโปรดักส์ ในฐานข้อมูลอาคารจะอยู่บนถนนราชวงศ์ แต่ในฐานข้อมูลของกรมการประกันสังคมจะอยู่บนถนนเยาวราช ข้อมูลจะมีการแย้งกันอยู่ ทำให้เมื่อเข้ารหัสแล้วจะได้รหัสคนละตัวกัน จึงไม่สามารถเชื่อมโยงกันได้ ระยะเวลาในการแก้ไขความไม่เข้ากันของข้อมูลนี้ไม่สามารถคาดเดาได้ขึ้นอยู่กับจำนวนข้อมูลและการความถูกต้องในการเก็บข้อมูลของหน่วยงานที่ใช้ช่วยการปรับปรุงข้อมูล เมื่อใช้โปรแกรมปรับปรุงข้อมูลแล้วจะให้ความถูกต้องสูงถึง 78% โดยใช้เวลาดำเนินการเพียง 1 เดือนเท่านั้น

จะเห็นได้ว่าผลการเชื่อมโยงโดยวิธีการใช้โปรแกรมช่วยในการปรับปรุงคุณภาพข้อมูล ได้ผลลัพธ์ที่ดีกว่า และใช้ระยะเวลาในการดำเนินการน้อยกว่า วิธีการใช้มือ ดังนั้นการใช้โปรแกรมในการปรับปรุงข้อมูลช่วยให้การเชื่อมโยงข้อมูลมีความถูกต้องและรวดเร็วมากขึ้น

5.5 ผลของการเชื่อมโยงฐานข้อมูลต่างๆ เข้าด้วยกัน

จากผลการเชื่อมโยงข้อมูลของหน่วยงานต่างๆ เข้ากับข้อมูลอาคารนั้น ทำให้ทราบว่าข้อมูลหลังจากแก้ไขความผิดพลาดแล้ว มีความไม่เข้ากันของข้อมูลอยู่มาก จึงต้องหาความสอดคล้องกันของข้อมูลในแต่ละหน่วยงาน เพื่อหาหน่วยงานที่มีความสอดคล้องของข้อมูลมากที่สุดเพื่อเป็นหลักในการเชื่อม โดยทดลองเชื่อมฐานข้อมูลในแต่ละหน่วยงานเข้าด้วยกัน ซึ่งได้ผลดังนี้

ตารางที่ 5.7 ผลของการเชื่อมโยงฐานข้อมูลระหว่างฐานข้อมูลต่างๆ

ข้อมูล	การไฟฟ้า	ประกันสังคม	ข้อมูลอาคาร	ข้อมูลผู้เสียภาษี
การไฟฟ้า		47%	40%	45%
ประกันสังคม	47%		75%	85%
ข้อมูลอาคาร	40%	75%		78%
ข้อมูลผู้เสียภาษี	45%	85%	78%	

จากตารางที่ 5.7 จะแสดงผลของการเชื่อมโยงฐานข้อมูลประกันสังคมกับฐานข้อมูลต่างๆ โดยสามารถเชื่อมกับฐานข้อมูลของการไฟฟ้าได้ 47% ข้อมูลอาคาร 75% และข้อมูลผู้เสียภาษี 85%

ผลของการเชื่อมโยงฐานข้อมูลการไฟฟ้ากับฐานข้อมูลต่างๆ โดยสามารถเชื่อมกับฐานข้อมูลประกันสังคมได้ 47% ข้อมูลอาคาร 40% และข้อมูลผู้เสียภาษี 45%

ผลของการเชื่อมโยงฐานข้อมูลผู้เสียภาษี กับฐานข้อมูลต่างๆ โดยสามารถเชื่อมกับฐานข้อมูลของการไฟฟ้า 45% ข้อมูลประกันสังคม 85% และข้อมูลอาคาร 78%

จากผลการเชื่อมโยงในระหว่างฐานข้อมูลต่างๆ จะเห็นได้ว่าข้อมูลที่เชื่อมโยงกับข้อมูลของการไฟฟ้าในตารางที่ 5.7 จะให้ผลเชื่อมโยงเพียง 40% เนื่องจากข้อมูลของการไฟฟ้านั้นเป็นข้อมูลของผู้จดทะเบียนมิเตอร์ไฟ ซึ่งข้อมูลที่ได้บันทึกในฐานข้อมูลอาจจะไม่ใช่ข้อมูลที่ตรงกับตำแหน่งมิเตอร์จริงๆ เมื่อทำการปรับปรุงข้อมูลแล้วพบว่า เหตุที่ทำให้ผลการเชื่อมโยงมีค่าต่ำเช่นนี้อาจเนื่องมาจากเป็นบริเวณที่มีถนนผ่านหลายสาย ทำให้ผู้บันทึกข้อมูลในสนามอาจเกิดความสับสนในการเลือกถนน ทำให้ข้อมูลที่ได้ไม่ตรงกับฐานข้อมูลที่มาจากการจดทะเบียน ดังนั้นจึงมีความไม่เข้ากันของข้อมูลแฝงอยู่เป็นจำนวนมาก

ส่วนการเชื่อมโยงข้อมูลระหว่างฐานข้อมูลผู้เสียภาษี และข้อมูลประกันสังคมมีค่าสูงถึง 85% เนื่องจากเป็นข้อมูลที่ได้จากการจดทะเบียนซึ่งจะเป็นข้อมูลที่ตรงกับความเป็นจริงมากกว่าข้อมูลที่ได้จากสนามและข้อมูลของการไฟฟ้า จึงมีความสอดคล้องกันของข้อมูลอยู่มาก ในขณะที่ผลการเชื่อมโยงระหว่างข้อมูลผู้เสียภาษีกับข้อมูลอาคารเชื่อมได้ 78% ซึ่งมีความสอดคล้องกันของข้อมูลอยู่มาก โดยข้อมูลเหล่านี้สามารถนำไปใช้ในการวิเคราะห์เพื่อขยายฐานภาษีต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6

บทสรุปและข้อเสนอแนะ

จุดประสงค์ในการทำวิจัยครั้งนี้คือ การจัดรูปแบบโครงสร้างข้อมูล “ที่อยู่” สำหรับเชื่อมโยงฐานข้อมูลต่างๆ โดยพัฒนาโปรแกรมการจัดโครงสร้างข้อมูล “ที่อยู่” เพื่อให้ฐานข้อมูลในหน่วยงานต่างๆ ที่มีรูปแบบในการจัดเก็บข้อมูลที่แตกต่างกัน สามารถนำมาใช้งานร่วมกันได้ โดยโปรแกรมนี้ได้ทำการจัดโครงสร้างข้อมูล แก้ไขความผิดพลาดของข้อมูล และเข้ารหัส ทำให้ข้อมูลมีความสมบูรณ์มากยิ่งขึ้น แล้วนำฐานข้อมูลมาเชื่อมกับข้อมูลแผนที่ในระบบสารสนเทศภูมิศาสตร์ เพื่อมาศึกษาวิเคราะห์ในเรื่องการขยายฐานภาษี ของกรมสรรพากร

6.1 สรุป

6.1.1 สรุปปัญหาสำคัญต่างๆ ที่พบของข้อมูล “ที่อยู่”

1. ปัญหาเรื่องของโครงสร้าง
2. ปัญหาเรื่องการเลือกใช้อักษรย่อ
3. ปัญหาเรื่องการกำหนดการนำเข้าข้อมูล “ที่อยู่”
4. ปัญหาเรื่องการแยกข้อมูล “ที่อยู่”
5. ปัญหาเรื่องความทันสมัยของข้อมูล

6.1.2 สรุปปัญหาความผิดพลาดของข้อมูล

จากผลการศึกษาข้อมูลที่มีอยู่แล้วในหน่วยงานต่างๆ ในบทที่ 3 พบว่าเปอร์เซ็นต์ความผิดพลาดในการบันทึกข้อมูลระหว่างหน่วยงานต่างๆ มีค่าสูง ทำให้เกิดปัญหาในการเชื่อมโยงฐานข้อมูลของหน่วยงานต่างๆ เข้าด้วยกัน โดยความผิดพลาดที่พบจะแบ่งออกเป็น 3 ส่วน สามารถสรุปได้ดังนี้

1. **การพิมพ์ข้อมูลผิดพลาด** เป็นความผิดพลาดที่เกิดขึ้นในขั้นตอนการนำเข้าข้อมูล โดยเปอร์เซ็นต์ความผิดพลาด ในกรณีที่ 1 ในฟิลด์

   NADD_NO จะอยู่ในช่วง 0.01% - 2.32%

   NMOO จะอยู่ในช่วง 0%

NSOI	จะอยู่ในช่วง	4.64% - 17.5%
NROAD	จะอยู่ในช่วง	1.95% - 16.18%
NTAMBON	จะอยู่ในช่วง	1.78% - 29.09%

2. **ความไม่ครบถ้วนของข้อมูล** เป็นความผิดพลาดที่ทำให้เกิดฟิลด์ว่างขึ้น คือ

2.1 ในขั้นตอนการนำเข้าข้อมูล เกิดจากการพิมพ์ข้อมูลไม่ครบถ้วน

เช่น พิมพ์ข้อมูลบ้านเลขที่ ซอย ถนน แขวง โดยไม่ใส่เขต หรือ

พิมพ์ข้อมูลบ้านเลขที่ ถนน แขวง เขต โดยไม่ใส่ซอย เป็นต้น

2.2 ในขั้นตอนการแยกข้อมูล เกิดจากเมื่อนำเข้าข้อมูลไม่ได้ใส่ตัวบ่งชี้ว่าข้อมูลนี้เป็นข้อมูลอะไร

เช่น พิมพ์ “เขตสัมพันธวงศ์” พิมพ์เป็น “สัมพันธวงศ์” หรือ

พิมพ์ “ตรอกไกร” พิมพ์เป็น “ไกร” เป็นต้น

โดยเปอร์เซ็นต์ความผิดพลาดในกรณีที่ 2 ในฟิลด์

NADD_NO	จะอยู่ในช่วง	1.47% - 12.39%
NMOO	จะอยู่ในช่วง	42.08% - 100%
NSOI	จะอยู่ในช่วง	64.62% - 83.28%
NROAD	จะอยู่ในช่วง	10.30% - 24.51%
NTAMBON	จะอยู่ในช่วง	26.06% - 79.84%

3. **ความไม่เข้ากันของข้อมูล** เป็นความผิดพลาดที่เกิดขึ้นเมื่อข้อมูลในแต่ละหน่วยงานมีความไม่สอดคล้องกันของข้อมูล

เช่น บริษัท ง่วนเชียงเยาวราชจำกัด จะมีความแย้งกันอยู่ในข้อมูลแขวง กล่าวคือ

“ที่อยู่” ที่ปรากฏในฐานข้อมูลอาคาร ของกรมสรรพากร คือ

273-275 ถ. เยาวราช *แขวงจักรวรรดิ* เขตสัมพันธวงศ์

“ที่อยู่” ที่ปรากฏในฐานข้อมูลของสำนักงานประกันสังคม คือ

273-275 ถ.เพาะพานิช *แขวงสัมพันธวงศ์* เขตสัมพันธวงศ์

โปรแกรมนี้สามารถแก้ไขความผิดพลาดในข้อ 1. และข้อ 2. ได้อย่างรวดเร็วและให้ความถูกต้องในการเชื่อมโยงฐานข้อมูลในแต่ละหน่วยงานเข้าด้วยกันได้ถึง 30 เปอร์เซ็นต์ ส่วนความผิดพลาดในข้อ 3. มีวิธีการแก้ไขได้โดยจะกล่าวต่อไปในข้อเสนอแนะ

6.1.3 สรุปผลการเชื่อมโยงฐานข้อมูล

1. ผลการเชื่อมโยงฐานข้อมูลต่างๆ กับ ข้อมูลอาคาร

ผลของการเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลอาคาร ก่อนทำการปรับปรุงความครบถ้วนของข้อมูล สามารถเชื่อมกับข้อมูลการไฟฟ้าได้ 32% ข้อมูลประกันสังคม 21% และข้อมูลผู้เสียภาษี 22% โดยใช้เวลาการดำเนินการเพียง 2 วัน

ผลของการเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลอาคาร หลังทำการปรับปรุงความครบถ้วนของข้อมูลโดยวิธีการใช้มือ สามารถเชื่อมข้อมูลกับการไฟฟ้าได้ 32% ข้อมูลประกันสังคม 57% และข้อมูลผู้เสียภาษี 76% โดยใช้เวลาการดำเนินการ 2 เดือน

ผลของการเชื่อมโยงฐานข้อมูลต่างๆ กับข้อมูลอาคาร หลังทำการปรับปรุงความครบถ้วนของข้อมูลโดยวิธีการใช้โปรแกรม สามารถเชื่อมกับข้อมูลการไฟฟ้าได้ 40% ข้อมูลประกันสังคม 75% และข้อมูลผู้เสียภาษี 78% โดยใช้เวลาการดำเนินการ 1 เดือน

สามารถสรุปได้ว่าเมื่อทำการปรับปรุงข้อมูลแล้วจะทำให้ผลของการเชื่อมโยงฐานข้อมูลดีขึ้น โดยใช้เวลาในการดำเนินการลดลง

2. ผลการเชื่อมโยงฐานข้อมูลต่างๆ เข้าด้วยกัน

ผลของการเชื่อมโยงฐานข้อมูลการใช้ไฟฟ้ากับฐานข้อมูลต่างๆ สามารถเชื่อมกับฐานข้อมูลของกรมประกันสังคมได้ 47% ข้อมูลอาคาร 40% และข้อมูลผู้เสียภาษี 45%

ผลของการเชื่อมโยงฐานข้อมูลประกันสังคมกับฐานข้อมูลต่างๆ สามารถเชื่อมกับฐานข้อมูลการไฟฟ้าได้ 47% ข้อมูลอาคาร 75% และข้อมูลผู้เสียภาษี 85%

ผลของการเชื่อมโยงฐานข้อมูลอาคารกับฐานข้อมูลต่างๆ สามารถเชื่อมกับฐานข้อมูลของการไฟฟ้า 40% ข้อมูลประกันสังคม 75% และข้อมูลผู้เสียภาษี 78%

ผลของการเชื่อมโยงฐานข้อมูลผู้เสียภาษีกับฐานข้อมูลต่างๆ สามารถเชื่อมกับฐานข้อมูลของการไฟฟ้า 45% ข้อมูลประกันสังคม 85% และข้อมูลอาคาร 78%

เมื่อทำการปรับปรุงข้อมูลแล้วในฐานข้อมูลผู้เสียภาษีของกรมสรรพากร และข้อมูลประกันสังคม ของสำนักงานประกันสังคม มีการเชื่อมโยงฐานข้อมูลได้สูงถึง 85% ใช้เวลาการดำเนินการ 1 เดือน โดยวิธีใช้โปรแกรมช่วยในการปรับปรุงฐานข้อมูล ผลการเชื่อมโยงทำให้สามารถนำข้อมูลจากฐานข้อมูลต่างๆ มาใช้ร่วมกันได้ก่อนที่จะมีการนำ HID มาใช้ในหน่วยงาน

6.2 ปัญหาอุปสรรค

ปัญหาและอุปสรรคที่พบในการทำวิทยานิพนธ์นี้ ได้แก่

1. ปัญหาเรื่องการขอข้อมูลที่ใช้ในการทำวิทยานิพนธ์ เนื่องจากต้องใช้ข้อมูลที่หน่วยงานต่างๆ จัดเก็บอยู่จริง แต่ไม่ค่อยได้รับความร่วมมือเท่าที่ควร เนื่องจากผู้จัดเก็บข้อมูลไม่ใช่ผู้มีอำนาจที่จะเผยแพร่ข้อมูลและไม่มีอำนาจตัดสินใจ แต่เมื่อสอบถามผู้มีอำนาจที่จะให้ข้อมูลได้ พบว่าต้องมีค่าใช้จ่ายในการได้มาซึ่งข้อมูล
2. ข้อมูลที่ได้มาเป็นข้อมูลที่ยังไม่ได้ผ่านการปรับปรุงแก้ไขมาก่อน ข้อมูลจึงมีความผิดพลาดแฝงอยู่มาก ทำให้เสียเวลาในการปรับปรุงข้อมูลเบื้องต้นค่อนข้างมาก

6.3 ข้อเสนอแนะ

6.3.1 ข้อเสนอแนะในการแก้ปัญหาสำคัญต่างๆ ที่พบของข้อมูล “ที่อยู่”

สามารถแก้ไขได้ 2 แนวทางหลัก คือ

1. การกำหนดข้อตกลงร่วมในการจัดเก็บข้อมูลที่อยู่ สำหรับการนำเข้าข้อมูลชุดใหม่ต่อไป ได้แก่

1.1 การกำหนดโครงสร้างข้อมูลที่อยู่ใหม่

1.2 การกำหนดการใช้อักษรย่อต่างๆ

1.3 การกำหนดการนำเข้าข้อมูล “ที่อยู่”

1.4 การปรับปรุงข้อมูลให้เป็นปัจจุบัน

2. การพัฒนาโปรแกรมช่วยในการจัดการข้อมูลเดิมที่มีอยู่ในฐานข้อมูล

2.1 แนวทางการแก้ปัญหาเรื่องลำดับการป้อนอักษรไทย

2.2 แนวทางการแก้ปัญหาเรื่องการแยกข้อมูล “ที่อยู่”

โดยได้กล่าวรายละเอียดไว้ในบทที่ 2 หัวข้อ 2.6

6.3.2 แนวทางการแก้ปัญหาความผิดพลาดของข้อมูล

การแก้ไขข้อมูลให้มีความถูกต้องมากยิ่งขึ้น ควรเพิ่มฟิลด์ชื่อเจ้าของ หรือ ฟิลด์ชื่อบริษัทในโครงสร้างข้อมูลกลางด้วย เพื่อเป็นตัวตรวจสอบว่าเมื่อเชื่อมข้อมูลเข้ากันแล้วมีชื่อเจ้าของหรือชื่อบริษัทตรงกันหรือไม่ เป็นการแก้ไขความผิดพลาดของความไม่เข้ากันของข้อมูล ซึ่งจะทำให้ข้อมูลมีความถูกต้องได้มากขึ้นถึง 80 เปอร์เซ็นต์

วิทยานิพนธ์นี้ได้ทำให้เกิดแนวความคิดในการวิจัยอื่นๆ ต่อไปอีก เช่น

1. ควรพัฒนาโปรแกรมเพื่อแก้ไขความผิดพลาด ในเรื่องการไม่เข้ากันของข้อมูลให้มีการแก้ไขให้รวดเร็วยิ่งขึ้น
2. พัฒนาโปรแกรมในส่วนแยกข้อมูลให้มีความถูกต้องมากยิ่งขึ้น โดยปราศจากตัวบ่งชี้

6.3.3 แนวทางการเชื่อมโยงฐานข้อมูลเข้ากับ HID ของสำนักบริหารการทะเบียน

รหัสประจำบ้าน (HID) คือเลขรหัสที่กำหนดขึ้นโดยสำนักบริหารการทะเบียน กรมการปกครอง ถือเป็นรหัสสำคัญที่เป็นตัวแทนบ้านในประเทศไทย ซึ่งเป็นฐานข้อมูลขนาดใหญ่ จากการศึกษาวิจัยครั้งนี้จะสร้างฟิลด์ NEWID ที่ใช้เป็นตัวเชื่อมข้อมูลระหว่างหน่วยงานต่างๆ เนื่องจากในปัจจุบันระบบ HID ยังไม่มีการเก็บในหน่วยงานต่างๆ การสร้าง HID ขึ้นมาเองไม่สามารถทำได้ เนื่องจากโครงสร้างของ HID จะมีเลขลำดับที่บ้านเลขที่ ที่สร้างโดยสำนักบริหารการทะเบียน และตัวเลขหลักที่ 11 สำหรับตรวจสอบว่าเลขนี้คือ HID หรือไม่ โดยนำตัวเลขทั้ง 10 หลักมาใช้สูตรคำนวณ ซึ่งเป็นข้อมูลที่ไม่สามารถสร้างขึ้นมาได้ จึงมีการใช้ฟิลด์ NEWID ในการศึกษาวิจัยครั้งนี้ เพราะเป็นฟิลด์ที่สามารถสร้างขึ้นโดยใช้ข้อมูล “ที่อยู่” ที่มีอยู่ จึงมีส่วนช่วยให้ข้อมูลมีการเชื่อมโยงกันได้ก่อนที่จะมี HID ถ้าไม่มีฟิลด์ NEWID โอกาสที่จะใช้ข้อมูลร่วมกันมีความเป็นไปได้น้อยมาก แต่ก็ใช่ว่าจะไม่สามารถใช้ข้อมูลร่วมกันได้เลย สามารถทำข้อมูลให้เชื่อมกันได้แต่ต้องใช้เวลามากทีเดียว โดยมีวิธีที่จะเชื่อมโยงข้อมูลเข้ากับ HID ของสำนักบริหารการทะเบียน

วิธีที่ 1 การเพิ่มข้อมูล HID เข้าไปทีละเรคคอร์ด

วิธีการนี้สามารถทำได้โดยประสานงานขอข้อมูลทะเบียนราษฎร์ จากสำนักบริหารการทะเบียน จากนั้นนำเข้าข้อมูลในฐานข้อมูลโดยนำเข้าทีละเรคคอร์ด จะเป็นขั้นตอนที่ต้องใช้เวลาในการดำเนินการมาก เนื่องจากจะต้องใช้เวลาในการหาข้อมูล HID ที่มีข้อมูล “ที่อยู่” ตรงกัน แล้วทำการพิมพ์ค่า HID ซึ่งการนำเข้าโดยการพิมพ์นี้ มีโอกาสที่จะเกิดความผิดพลาดสูง

วิธีที่ 2 การใช้ข้อมูล HID ของฐานข้อมูลอื่น

ในวิธีที่ 2 สามารถแบ่งได้เป็น 2 กรณี คือ

1. นำฐานข้อมูลไปเชื่อมกับฐานข้อมูลที่มี HID อยู่แล้ว

วิธีนี้จะให้ความถูกต้องและรวดเร็วขึ้นกว่าวิธีแรกเนื่องจากไม่ต้องผ่านการนำเข้าข้อมูล HID โดยการพิมพ์อีกครั้งหนึ่ง โดยต้องทำการศึกษาโครงสร้างของหน่วยงานที่จัดเก็บ HID และทราบรหัสที่ใช้ในหน่วยงานนั้นก่อน เพื่อทำการเขียนโปรแกรมช่วยในการจัดโครงสร้างข้อมูลให้

เป็นโครงสร้างกลางเพื่อที่จะใช้ข้อมูล “ที่อยู่” มาสร้างตัวเชื่อม NEWID วิธีการนี้จะช่วยประหยัดเวลาและเพิ่มความถูกต้อง ในการนำเข้า HID

2. ฐานข้อมูลที่มีการเชื่อมโยงกันของหน่วยงานที่ไม่มี HID แต่ในภายหลังมีหน่วยงานใด หน่วยงานหนึ่งจัดทำ HID ขึ้น

วิธีนี้เป็นวิธีที่ผ่านการนำฐานข้อมูลที่ไม่ได้มีการจัดเก็บข้อมูล HID อยู่ในฐานข้อมูลมาก่อนมาเชื่อมโยงฐานข้อมูลเข้าด้วยกันแล้ว ต่อมาผ่านหลัง ได้มีหน่วยงานใดหน่วยงานหนึ่งที่ได้ทำการเชื่อมโยงได้ทำข้อมูล HID ขึ้นมาในฐานข้อมูลของตัวเอง หน่วยงานที่ทำการเชื่อมโยงฐานข้อมูลเข้ากับหน่วยงานนี้อยู่ก่อนแล้วก็สามารถที่จะใช้ประโยชน์ของ HID ที่ถูกสร้างขึ้นมาในภายหลังของหน่วยงานนั้นได้ ซึ่งเป็นวิธีที่ใช้ในวิทยานิพนธ์ครั้งนี้โดยข้อมูล HID จะถูกจัดเก็บอยู่ในฐานข้อมูลอาคารของกรมสรรพากรใน ฟิลด์ HOUSE_ID

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

ภาษาไทย

"การจัดทำแผนที่ภาษี GIS เพื่อเพิ่มประสิทธิภาพในการจัดเก็บภาษีอากร," วารสารภาษี บัญชี และกฎหมายธุรกิจ 51 (ธันวาคม 2541): 12-17.

กรุงเทพมหานคร, Bangkok Land information system project. กรุงเทพมหานคร: 1988.

กิตติ ภักดีวัฒนะกุล และ จำลอง ครูอุตสาหะ. Visual Basic 6 ฉบับโปรแกรมเมอร์. พิมพ์ครั้งที่10. กรุงเทพมหานคร: ไทยเจริญการพิมพ์, 2545.

นฤมล นุชนารถ และ วิชัย เยี่ยงวีรชน. "ระบบสารสนเทศแผนที่เพื่อการบริหารการจัดเก็บภาษี," การประชุมวิชาการ ภูมิสารสนเทศแห่งชาติ ครั้งที่ 1 ณ โรงแรมเซ็นทรัลแกรนด์พลาซ่า กรุงเทพมหานคร (27-28 มิถุนายน 2543): 103-109.

ประยุทธ สุวรรณวิสารท. การเข้ารหัสคำทับศัพท์เพื่อการค้นคืนข้ามภาษาไทย-อังกฤษ. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2541.

วรวัฒน์ วรศิลป์. ขั้นตอนวิธีการจัดทำดัชนีสำหรับข้อความไทยที่มีความผิดพลาด. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2542.

วิสาร กำจรเวทย์. Visual Basic ฉบับ Database. พิมพ์ครั้งที่2. กรุงเทพมหานคร: โปรวิชั่น, 2540.

ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, ข้อเสนอแนะ/มาตรการเกี่ยวกับการดำเนินงานโครงสร้างพื้นฐานข้อมูลภาครัฐ. กรุงเทพมหานคร: กระทรวงวิทยาศาสตร์และเทคโนโลยี, 2546.

ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, รายงานฉบับสมบูรณ์ โครงการวิจัยเรื่องโครงสร้างพื้นฐานข้อมูลภาครัฐ. กรุงเทพมหานคร: กระทรวงวิทยาศาสตร์และเทคโนโลยี, 2544.

ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, สถานภาพและความต้องการโครงสร้างพื้นฐานข้อมูลภาครัฐ. กรุงเทพมหานคร: กระทรวงวิทยาศาสตร์และเทคโนโลยี, 2545.

หน่วยส่งเสริมประสิทธิภาพ. การปรับเปลี่ยนกระบวนการงานสำรวจแหล่งภาษีอากร. กรุงเทพมหานคร: กรมสรรพากร, 2540.

หน่วยส่งเสริมประสิทธิภาพ. โครงการจัดทำระบบสารสนเทศแผนที่ภาษี รายงานฉบับสมบูรณ์สำหรับผู้บริหาร. กรุงเทพมหานคร: กรมสรรพากร, 2542.

รายการอ้างอิง(ต่อ)

ภาษาอังกฤษ

Date,C.J. An introduction to Database Systems. 6th ed. Massachusetts: Addison-Wesley Publishing, 1995.

John Schlossor. Geocoding Issues, Alternatives&Techniques. 2001: pp. 1-7.

U.S. Department of Justice, Office of Community Oriented Policing Services,2000:4.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก

รหัสโปรแกรม ส่วนที่ทำเป็นไฟล์ Library

```
Public Sub SPlit_it(ByRef Elect As Database, aTable As String, aCase As Integer)
Dim Db As Database, RecSet1 As Recordset, RecSet2 As Recordset, RecSet3 As Recordset, RecSet4 As
Recordset
Dim recSet As Recordset
Dim sAddress As String, sTemp As String, sCheck As String, p As Integer, finish As Integer
   On Error Resume Next
   Set recSet = Elect.OpenRecordset(aTable, dbOpenTable)
      Set Db = Workspaces(0).OpenDatabase(App.Path, False, True, "dBASE IV;")
   Set RecSet1 = Db.OpenRecordset("LUTSRD", dbOpenDynaset)
   Set RecSet2 = Db.OpenRecordset("LUTKHED", dbOpenDynaset)
   Set RecSet3 = Db.OpenRecordset("PROVINCE", dbOpenDynaset)
   Set RecSet4 = Db.OpenRecordset("Tambol", dbOpenDynaset)

   'การแยกข้อมูลของหน่วยงานการไฟฟ้า และสำนักงานประกันสังคม
    If aCase = 0 Or aCase = 2 Then
               While Not recSet.EOF
                  If IsNull(recSet!ADDRESS) Then
                     MsgBox "Null"
                  Else
                     sAddress = Trim(recSet!ADDRESS)
                     p = InStr(sAddress, " ")
                     sTemp = ""
                     Dim UseADD_NO As Boolean
                     Dim UseMOO As Boolean
                     Dim UseSOI As Boolean
                     Dim UseROAD As Boolean
                     Dim UseTAMBON As Boolean
                     Dim UseAMPHOE As Boolean
                     Dim UsePROVINCE As Boolean
                     Dim UseREMARK As Boolean
               UseADD_NO = False
```

```
UseMOO = False
UseSOI = False
UseROAD = False
UseTAMBON = False
UseAMPHOE = False
UsePROVINCE = False
UseREMARK = False
recSet.Edit
recSet!NADD_NO = ""
recSet!NMOO = ""
recSet!NSOI = ""
recSet!NROAD = ""
recSet!NTAMBON = ""
recSet!NAMPHOE = ""
recSet!NPROVINCE = ""
recSet!REMARK = ""
recSet.Update

    Do While p > 0 Or Trim(sAddress) <> ""
      If p > 0 Then sCheck = Trim(Left(sAddress, p - 1))
      If p = 0 Then sCheck = sAddress
      If p > 0 Then sAddress = Trim(Mid(sAddress, p + 1))
      If p = 0 Then sAddress = ""
      If IsNumeric(Left(sCheck, 1)) = False Then UseADD_NO = True
      If IsNumeric(Left(sCheck, 1)) = True And UseADD_NO = False And Len(sCheck) < 20
Then
          recSet.Edit
          recSet!NADD_NO = sCheck
          recSet.Update
          UseADD_NO = True
      ElseIf (Left(sCheck, 2) = "ม." Or Left(sCheck, 4) = "หมู่") And _
      UseMOO = False And Len(sCheck) < 6 Then
          If Left(sCheck, 2) = "ม." And IsNumeric(Mid(sCheck, 3, 1)) Then
            recSet.Edit
            recSet!NMOO = Mid(sCheck, 3, Len(sCheck) - 2)
            recSet.Update
```

```
            UseMOO = True
            UseADD_NO = True
        Else
            sTemp = sTemp & sCheck & ","
        End If
        If Left(Trim(sCheck), 4) = "หมู่" And IsNumeric(Mid(sCheck, 3, 1)) Then
            recSet.Edit
            recSet!NMOO = Mid(sCheck, 5, Len(sCheck) - 4)
            recSet.Update
            UseMOO = True
        Else
            If UseMOO = False Then sTemp = sTemp & sCheck & ","
        End If
        If Trim(sCheck) = "หมู่" Then
            p = InStr(sAddress, " ")
            sCheck = Trim(Left(sAddress, p - 1))
            If p <> 0 Then
                If Len(Trim(Left(sAddress, p - 1))) < 6 Then
                    recSet.Edit
                    recSet!NMOO = Trim(sCheck)
                    recSet.Update
                Else
                    sTemp = sTemp & sCheck & ","
                End If
            End If
            sAddress = Trim(Mid(sAddress, p + 1))
            UseMOO = True
        End If
    ElseIf Left(sCheck, 1) = "ซ" Or Left(sCheck, 4) = "ตรอก" And UseSOI = False Then
        If Left(sCheck, 2) = "ซ." Then
            recSet.Edit
            recSet!NSOI = Trim(Mid(sCheck, 3, Len(sCheck) - 2))
            recSet.Update
            UseSOI = True
            UseADD_NO = True
```

```
                    UseMOO = True
                End If
                If Left(sCheck, 3) = "ซอย" Then
                    recSet.Edit
                    recSet!NSOI = Trim(Mid(sCheck, 4, Len(sCheck) - 3))
                    recSet.Update
                    UseSOI = True
                    UseADD_NO = True
                    UseMOO = True
                End If
                If Left(sCheck, 4) = "ตรอก" Then
                    recSet.Edit
                    recSet!NSOI = Trim(Mid(sCheck, 5, Len(sCheck) - 4))
                    recSet.Update
                    UseSOI = True
                    UseADD_NO = True
                    UseMOO = True
                End If

            ElseIf UseSOI = True And UseROAD = False And UseTAMBON = False And _
            IsNumeric(Left(sCheck, 1)) = True And Len(Trim(sCheck)) < 6 Then
                recSet.Edit
                recSet!NSOI = Trim(recSet!NSOI & sCheck)
                recSet.Update
                UseSOI = True
                UseADD_NO = True
                UseMOO = True
            ElseIf (Left(sCheck, 2) = "ถ." Or Left(sCheck, 3) = "ถนน") And UseROAD = False And
Len(sCheck) < 50 Then
                If (Left(sCheck, 2) = "ถ.") Then
                    recSet.Edit
                    recSet!NROAD = Trim(Mid(sCheck, 3, Len(sCheck) - 2))
                    recSet.Update
                    UseROAD = True
                    UseSOI = True
                    UseADD_NO = True
```

```
            UseMOO = True
          End If
          If (Left(sCheck, 3) = "ถนน") Then
            recSet.Edit
            recSet!NROAD = Trim(Mid(sCheck, 4, Len(sCheck) - 3))
            recSet.Update
            UseROAD = True
            UseSOI = True
            UseADD_NO = True
            UseMOO = True
          End If
        ElseIf UseSOI = True And UseROAD = True And UseTAMBON = False And _
        IsNumeric(sCheck) = True And Len(Trim(sCheck)) = 1 Then
          recSet.Edit
          recSet!NROAD = Trim(recSet!NROAD & sCheck)
          recSet.Update
        ElseIf (Left(sCheck, 2) = "ต." Or Left(sCheck, 4) = "ตำบล" Or Left(sCheck, 4) = "แขวง")
And _
        UseTAMBON = False And Len(sCheck) < 31 Then
          If (Left(sCheck, 2) = "ต.") Then
            recSet.Edit
            recSet!NTAMBON = Trim(Mid(sCheck, 3, Len(sCheck) - 2))
            recSet.Update
            UseTAMBON = True
          End If
          If (Left(sCheck, 4) = "ตำบล") Then
            recSet.Edit
            recSet!NTAMBON = Trim(Mid(sCheck, 5, Len(sCheck) - 4))
            recSet.Update
            UseTAMBON = True
          End If
          If (Left(sCheck, 4) = "แขวง") Then
            recSet.Edit
            recSet!NTAMBON = Trim(Mid(sCheck, 5, Len(sCheck) - 4))
            recSet.Update
            UseTAMBON = True
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
            End If
        ElseIf (Left(sCheck, 2) = "อ." Or Left(sCheck, 5) = "อำเภอ" Or Left(sCheck, 3) = "เขต")
And _
        UseAMPHOE = False And Len(sCheck) < 26 Then
            If (Left(sCheck, 2) = "อ.") Then
                recSet.Edit
                recSet!NAMPHOE = Trim(Mid(sCheck, 3, Len(sCheck) - 2))
                recSet.Update
                UseAMPHOE = True
            End If
            If Left(sCheck, 5) = "อำเภอ" Then
                recSet.Edit
                recSet!NAMPHOE = Trim(Mid(sCheck, 6, Len(sCheck) - 5))
                recSet.Update
                UseAMPHOE = True
            End If
            If Left(sCheck, 3) = "เขต" Then
                recSet.Edit
                recSet!NAMPHOE = Trim(Mid(sCheck, 4, Len(sCheck) - 3))
                recSet.Update
                UseAMPHOE = True
            End If
        ElseIf (Left(sCheck, 2) = "จ." Or Left(sCheck, 5) = "จังหวัด") And UsePROVINCE = False
Then
            recSet.Edit
            recSet!NPROVINCE = sCheck
            recSet.Update
            UsePROVINCE = True
        Else
            sTemp = sTemp & sCheck & ","
        End If
        p = InStr(sAddress, " ")
    Loop
    If Len(sTemp) > 0 Then sTemp = Left(sTemp, Len(sTemp) - 1)
    recSet.Edit
    recSet!REMARK = sTemp
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
    recSet.Update
End If
If aCase = 0 Then
    RecSet2.FindFirst "code_elec = '" & Val(recSet!AMPHOE_CODE) & "'"
      If RecSet2.NoMatch Then
        recSet.Edit
        recSet!NAMPHOE = ""
        recSet.Update
      End If

      If Not RecSet2.NoMatch Then
        recSet.Edit
        recSet!NAMPHOE = RecSet2!khed
        recSet.Update
      End If

    RecSet2.FindFirst "khed = '" & recSet!NAMPHOE & "'"
    RecSet3.FindFirst "province_i = '" & Left(RecSet2!code_soc, 2) & "'"
      If Not RecSet3.NoMatch Then
        recSet.Edit
        recSet!NPROVINCE = RecSet3!province_n
        recSet.Update
      End If
End If
If aCase = 2 Then
    RecSet2.FindFirst "code_soc = '" & recSet!AMPHURCODE & "'"
      If Not RecSet2.NoMatch Then
        recSet.Edit
        recSet!NAMPHOE = RecSet2!khed
        recSet.Update
      End If
    RecSet3.FindFirst "province_i = '" & recSet!PROVINCE_CODE & "'"
      If Not RecSet3.NoMatch Then
        recSet.Edit
        recSet!NPROVINCE = RecSet3!province_n
        recSet.Update
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
                End If
            End If
            recSet.MoveNext
        Wend

'การแยกข้อมูลของกรมสรรพากร
ElseIf aCase = 3 Then
        While Not recSet.EOF
        recSet.Edit
        recSet!NADD_NO = ""
        recSet!NMOO = ""
        recSet!NSOI = ""
        recSet!NROAD = ""
        recSet!NTAMBON = ""
        recSet!NAMPHOE = ""
        recSet!NPROVINCE = ""
        recSet!REMARK = ""
        recSet.Update
            If IsNull(recSet!ADDR) Then
            Else
                sAddress = Trim(recSet!ADDR)
                    If IsNumeric(Left(sAddress, 1)) = True Then
                        p = InStr(sAddress, " ")
                        Do While p > 0 And Trim(sAddress) <> ""
                            If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
                            p = InStr(sAddress, " ")
                        Loop
                        p = InStr(sAddress, """)
                        Do While p > 0 And Trim(sAddress) <> ""
                            If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & "," & Trim(Mid(sAddress, p
+ 1))
                            p = InStr(sAddress, """)
                        Loop
                                recSet.Edit
                                recSet!NADD_NO = sAddress
                                recSet.Update
```

```
        Else
                        recSet.Edit
                        recSet!REMARK = sAddress
                        recSet.Update


        End If
End If
If IsNull(recSet!MOO) Then
Else
   sAddress = Trim(recSet!MOO)
   p = InStr(sAddress, " ")
   Do While p > 0 And Trim(sAddress) <> ""
      If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
      p = InStr(sAddress, " ")
   Loop
                recSet.Edit
                recSet!NMOO = sAddress
                recSet.Update
End If
If IsNull(recSet!SOI) Then
Else
   sAddress = Trim(recSet!SOI)
   If Trim(Left(sAddress, 2)) = "ซ." Then sAddress = Trim(Mid(sAddress, 3))
   If Trim(Left(sAddress, 3)) = "ซอย" Then sAddress = Trim(Mid(sAddress, 4))
   If Trim(Left(sAddress, 4)) = "ตรอก" Then sAddress = Trim(Mid(sAddress, 5))
   p = InStr(sAddress, " ")
   Do While p > 0 And Trim(sAddress) <> ""
      If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
      p = InStr(sAddress, " ")
   Loop
                recSet.Edit
                recSet!NSOI = sAddress
                recSet.Update
End If
If IsNull(recSet!ROAD) Then
Else
```

```
    sAddress = Trim(recSet!ROAD)
    If Trim(Left(sAddress, 2)) = "ถ." Then sAddress = Trim(Mid(sAddress, 3))
    If Trim(Left(sAddress, 3)) = "ถนน" Then sAddress = Trim(Mid(sAddress, 4))
    p = InStr(sAddress, " ")
    Do While p > 0 And Trim(sAddress) <> ""
      If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
      p = InStr(sAddress, " ")
    Loop
                recSet.Edit
                recSet!NROAD = sAddress
                recSet.Update
End If
If IsNull(recSet!TAMBON) Then
Else
    sAddress = Trim(recSet!TAMBON)
    If Trim(Left(sAddress, 2)) = "ต." Then sAddress = Trim(Mid(sAddress, 3))
    If Trim(Left(sAddress, 4)) = "ตำบล" Then sAddress = Trim(Mid(sAddress, 5))
    If Trim(Left(sAddress, 4)) = "แขวง" Then sAddress = Trim(Mid(sAddress, 5))
    p = InStr(sAddress, " ")
    Do While p > 0 And Trim(sAddress) <> ""
      If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
      p = InStr(sAddress, " ")
    Loop
                recSet.Edit
                recSet!NTAMBON = sAddress
                recSet.Update
End If
      RecSet2.FindFirst "code_tin = '" & recSet!CODE_PROVINCE & "'"
        If Not RecSet2.NoMatch Then
          recSet.Edit
          recSet!NAMPHOE = RecSet2!khed
          recSet.Update
        End If
      RecSet2.FindFirst "khed = '" & recSet!NAMPHOE & "'"
      RecSet3.FindFirst "province_i = '" & Left(RecSet2!code_soc, 2) & "'"
        If Not RecSet3.NoMatch Then
```

```
                    recSet.Edit
                    recSet!NPROVINCE = RecSet3!province_n
                    recSet.Update
                End If
            recSet.MoveNext
        Wend

'การแยกข้อมูลอาคาร
ElseIf aCase = 1 Then
        While Not recSet.EOF
        recSet.Edit
        recSet!NADD_NO = ""
        recSet!NMOO = ""
        recSet!NSOI = ""
        recSet!NROAD = ""
        recSet!NTAMBON = ""
        recSet!NAMPHOE = ""
        recSet!NPROVINCE = ""
        recSet!REMARK = ""
        recSet.Update
          If IsNull(recSet!ADDR_NO) Then
          Else
            sAddress = Trim(recSet!ADDR_NO)
            p = InStr(sAddress, " ")
            Do While p > 0 And Trim(sAddress) <> ""
              If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
              p = InStr(sAddress, " ")
            Loop
            p = InStr(sAddress, """)
            Do While p > 0 And Trim(sAddress) <> ""
              If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & "," & Trim(Mid(sAddress, p + 1))
              p = InStr(sAddress, """)
            Loop
                    recSet.Edit
                    recSet!NADD_NO = sAddress
                    recSet.Update
```

```
End If
If IsNull(recSet!MOO) Then
Else
   sAddress = Trim(recSet!MOO)
   p = InStr(sAddress, " ")
   Do While p > 0 And Trim(sAddress) <> ""
      If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
      p = InStr(sAddress, " ")
   Loop
            recSet.Edit
            recSet!NMOO = sAddress
            recSet.Update
End If
If IsNull(recSet!SOI) Then
Else
   sAddress = Trim(recSet!SOI)
   If Trim(Left(sAddress, 2)) = "ซ." Then sAddress = Trim(Mid(sAddress, 3))
   If Trim(Left(sAddress, 3)) = "ซอย" Then sAddress = Trim(Mid(sAddress, 4))
   If Trim(Left(sAddress, 4)) = "ตรอก" Then sAddress = Trim(Mid(sAddress, 5))
   p = InStr(sAddress, " ")
   Do While p > 0 And Trim(sAddress) <> ""
      If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
      p = InStr(sAddress, " ")
   Loop
            recSet.Edit
            recSet!NSOI = sAddress
            recSet.Update
End If
If IsNull(recSet!ROAD) Then
Else
   sAddress = Trim(recSet!ROAD)
   If Trim(Left(sAddress, 2)) = "ถ." Then sAddress = Trim(Mid(sAddress, 3))
   If Trim(Left(sAddress, 3)) = "ถนน" Then sAddress = Trim(Mid(sAddress, 4))
   p = InStr(sAddress, " ")
   Do While p > 0 And Trim(sAddress) <> ""
      If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
    p = InStr(sAddress, " ")
  Loop
        recSet.Edit
        recSet!NROAD = sAddress
        recSet.Update
End If
If IsNull(recSet!KHWENG) Then
Else
  sAddress = Trim(recSet!KHWENG)
  If Trim(Left(sAddress, 2)) = "ต." Then sAddress = Trim(Mid(sAddress, 3))
  If Trim(Left(sAddress, 4)) = "ตำบล" Then sAddress = Trim(Mid(sAddress, 5))
  If Trim(Left(sAddress, 4)) = "แขวง" Then sAddress = Trim(Mid(sAddress, 5))
  p = InStr(sAddress, " ")
  Do While p > 0 And Trim(sAddress) <> ""
    If p > 0 Then sAddress = Trim(Left(sAddress, p - 1)) & Trim(Mid(sAddress, p + 1))
    p = InStr(sAddress, " ")
  Loop
        recSet.Edit
        recSet!NTAMBON = sAddress
        recSet.Update
End If
    If Not IsNull(recSet!NTAMBON) Then
      RecSet4.FindFirst "tambol_nam = '" & recSet!NTAMBON & "'"
    End If
    RecSet2.FindFirst "code_soc = '" & RecSet4!Amphur_id & "'"
      If IsNull(RecSet4!Amphur_id) Then
        recSet.Edit
        recSet!NAMPHOE = ""
        recSet.Update
      ElseIf Not recSet.NoMatch Then
        recSet.Edit
        recSet!NAMPHOE = RecSet2!khed
        recSet.Update
      End If
    If recSet!NAMPHOE = "" Then
        recSet.Edit
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
                recSet!NPROVINCE = ""
                recSet.Update
            Else
              RecSet2.FindFirst "khed = '" & recSet!NAMPHOE & "'"
              RecSet3.FindFirst "province_i = '" & Left(RecSet2!code_soc, 2) & "'"
              If Not RecSet3.NoMatch Then
                recSet.Edit
                recSet!NPROVINCE = RecSet3!province_n
                recSet.Update
              End If
            End If
          recSet.MoveNext
        Wend

    End If
End Sub
```

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข

Look Up Table ของซอยและถนนในเขตสัมพันธวงศ์

ID_ROAD	ORDER	TYPE	THAI_NA	LOCAL_TNA	CODE_KHED
1006	1	ถนน	พระรามที่4	พระราม4	07
1007	2	ถนน	เจริญกรุง	-	07
1008	3	ถนน	เยาวราช	-	07
1009	4	ถนน	แปลงนาม	-	07
1010	5	ถนน	วานิช1	สำเพ็ง	07
1011	6	ถนน	จักรวรรดิ์	จักรวรรดิ	07
1012	7	ถนน	ตรีมิตร	-	07
1013	8	ถนน	ทรงสวัสดิ์	-	07
1014	9	ถนน	บริพัตร	-	07
1015	10	ถนน	มังกร	-	07
1016	11	ถนน	มหาจักร	-	07
1017	12	ถนน	นี้จงสวัสดี	-	07
1018	13	ถนน	ทรงเสริม	-	07
1019	14	ถนน	เยาวพานิชย์	เยาวพานิช	07
1020	15	ถนน	ผดุงด้าว	-	07
1021	16	ถนน	ทรงวาด	-	07
1022	17	ถนน	พาดสาย	-	07
1023	18	ถนน	วิวัฒน์เวียง	-	07
1024	19	ถนน	เสริมสินค้า	-	07
1025	20	ถนน	ข้าวหลาม	-	07
1026	21	ถนน	โปริสภา	-	07
1027	22	ถนน	เพาะพานิชย์	เพาะพานิช	07
1028	23	ถนน	ราชวงศ์	-	07
1029	24	ถนน	อนุวงศ์	-	07
1030	25	ถนน	ลำพูนไชย	-	07
1031	26	ถนน	โยธา	-	07
1032	27	ซอย	เจริญกรุง27	-	07
1033	28	ซอย	เจริญกรุง29	พิทยลาภ	07

ID_ROAD	ORDER	TYPE	THAI_NA	LOCAL_TNA	CODE_KHED
1034	29	ซอย	เจริญกรุง31	โชฎึก	07
1035	30	ซอย	เจริญกรุง33	โรงน้ำแข็ง	07
1036	31	ซอย	เจริญกรุง35	โรงน้ำแข็ง2	07
1037	32	ซอย	เจริญกรุง8	นครเกษม1	07
1038	33	ซอย	เจริญกรุง10	นครเกษม2	07
1039	34	ซอย	เจริญกรุง12	บำรุงรัฐ	07
1040	35	ซอย	เจริญกรุง14	กัมมาตุยาราม	07
1041	36	ซอย	เจริญกรุง16	อิสรานุภาพ	07
1042	37	ซอย	เจริญกรุง18	-	07
1043	38	ซอย	เจริญกรุง20	อุภัยราชบำรุง	07
1044	39	ซอย	เจริญกรุง22	ตลาดน้อย	07
1045	40	ซอย	เจริญกรุง24	โรงน้ำแข็ง1	07
1046	41	ตรอก	-	วัดกุศลสมาคร	07
1047	42	ซอย	ไปรษณีย์	-	07
1048	43	ซอย	เลื่อนฤทธิ์	-	07
1049	44	ซอย	อิสรานุภาพ	ตลาดเก่า	07
1050	45	ตรอก	โพธิ์	-	07
1051	46	ตรอก	ไทร	-	07
1052	47	ตรอก	โรงพิมพ์	-	07
1053	48	ตรอก	สิบเบี้ย	-	07
1054	49	ซอย	พิพากษา1	-	07
1055	50	ซอย	พิพากษา2	-	07
1056	51	ซอย	-	วัดสัมพันธวงศ์	07
1057	52	ซอย	บพิตรพิมุข	-	07
1058	53	ตรอก	หัวเม็ด	-	07
1059	54	ซอย	สุกร1	-	07
1060	55	ซอย	สุกร2	-	07
1061	56	ซอย	-	แยกปทุมคงคา	07
1062	57	ซอย	ไพบูลย์สมบัติ	-	07
1063	58	ซอย	เจริญพานิช	เชียงกง	07
1064	59	ซอย	ภาณุรังษี	-	07
1065	60	ซอย	ภิรมย์	-	07

ID_ROAD	ORDER	TYPE	THAI_NA	LOCAL_TNA	CODE_KHED
1066	61	ซอย	อาเนี๊ยเก็ง	อาเนี้ยเก็ง	07
1067	62	ซอย	-	ข้างโรงหนังโอเดียน	07
1068	63	ซอย	วัดปทุมคงคา	ข้างวัดปทุมคงคา	07
1069	64	ซอย	ท่าน้ำศาลเจ้า	-	07
1070	65	ซอย	ไกรฑามาตย์	-	07
1071	66	ตรอก	สะพานญวน	-	07
1072	67	ตรอก	ชัยภูมิ	-	07
1073	68	ตรอก	โรงโคม	-	07
1074	69	ซอย	กลันตัน	-	07
1075	70	ซอย	ไมตรีวานิช	-	07
1076	71	ตรอก	สะพานยาว	-	07
1077	72	ซอย	ผลิตผล	-	07
1078	73	ซอย	-	แยกไมตรีวานิช	07
1079	74	ตรอก	ไกร	-	07
1080	75	ตรอก	ข้าวสาร	-	07
1081	76	ซอย	วานิช2	โรงน้ำแข็ง	07
1082	77	ซอย	ปทุมคงคา	-	07
1083	78	ตรอก	ศาลเจ้าโรงเกือก	-	07
1084	79	ซอย	-	ข้างสหธนาคาร	07
1085	80	ซอย	-	ศาลเจ้าโรงสูบ	07
7511	81	ตรอก	กระเปิด	-	07
7512	82	ซอย	คลองถมปทุมคงคา	-	07
7513	83	ตรอก	ซุ่ม	ซุ่ม	07
7514	84	ซอย	ตลาดสะพานหัน	-	07
7515	85	ซอย	ท่าน้ำจักรวรรดิ์	-	07
7516	86	ซอย	นครเกษม3	-	07
7517	87	ซอย	นครเกษม5	-	07
7518	88	ซอย	แฟริเท็ก	-	07
7519	89	ซอย	วัดชัยชนะสงคราม	วัดตึก	07
7520	90	ซอย	วานิช	-	07
7521	91	ซอย	สมยศ	-	07
7522	92	ซอย	สืบสุข	-	07

ID_ROAD	ORDER	TYPE	THAI_NA	LOCAL_TNA	CODE_KHED
7523	93	ซอย	สุกร	-	07
7524	94	ซอย	ฮั่วเพ่งโกย	-	07
7525	95	ตรอก	เซ่งเฮียม	-	07
7526	96	ถนน	เวิ้งนครเกษม	นครเกษม	07
7527	97	ถนน	สี่แยกวัดตึก	-	07

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายพิชญ์ ติรณสวัสดิ์ เกิดวันที่ 17 กันยายน 2519 ที่กรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขา วิศวกรรมสำรวจ จาก ภาควิชา วิศวกรรมสำรวจ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2540 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต ที่จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ.2543 ปัจจุบันดำรงตำแหน่งวิศวกรรังวัด สำนักเทคโนโลยีทำแผนที่ กรมที่ดิน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย