เครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด

นางสาววัชรวรรณ ศรีรัตนาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2542

ISBN 974-333-634-6

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

I 19065383

A TOOL FOR TRANSLATING THE ENTITY RELATIONSHIP MODEL TO FORMAL SPECIFICATION IN Z NOTATION

MISS VACHARAWAN SRIRATANALAI

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science in Computer Science

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 1999

ISBN 974-333-634-6

หัวข้อวิทยานิพนธ์	เครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด
โดย	นางสาววัชรวรรณ ศรีรัตนาลัย
ภาควิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ วิวัฒน์ วัฒนาวุฒิ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์

(รองศาสตราจารย์ ดร.ธัชชัย สุมิตร.)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์)

.....อาจารย์ที่ปรึกษา

(อาจารย์ วิวัฒน์ วัฒนาวุฒิ)

.....กรรมการ

(อาจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์)

.....กรรมการ

(อาจารย์ ดร.พรศิริ หมื่นไชยศรี)

วัชรวรรณ ศรีรัตนาลัย : เครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด (A TOOL FOR TRANSLATING THE ENTITY RELATIONSHIP MODEL TO FORMAL SPECIFICATION IN Z NOTATION) อ.ที่ปรึกษา : อ.วิวัฒน์ วัฒนาวุฒิ, 86 หน้า. ISBN 974-333-634-6.

วิทยานิพนธ์นี้ได้ออกแบบขั้นตอนวิธีและพัฒนาเครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด โดยใช้ข้อมูลจากพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ของแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อมูลนำเข้า ขั้นตอนวิธีที่ใช้ในการแปลงแบ่งได้เป็น 2 กลุ่มคือ กลุ่มขั้นตอนวิธีการสังเคราะห์คุณสมบัติโครงสร้าง และ กลุ่มขั้นตอนวิธีการสังเคราะห์คุณสมบัติพฤติกรรม โดยผลของขั้นตอนวิธีการสังเคราะห์คุณสมบัติโครงสร้างจะได้เป็นสัญกรณ์เซด 3 เค้าร่าง คือ เค้าร่างเอนทิตี เค้าร่างเพิ่มเติม และ เค้าร่างความสัมพันธ์ ในขณะที่ผลของขั้นตอนวิธีการสังเคราะห์คุณสมบัติพฤติกรรมจะได้เป็นสัญกรณ์เซด 3 เค้าร่างเช่นกัน คือ เค้าร่างการเพิ่มระเบียน เค้าร่างการแก้ไขระเบียน และเค้าร่างการลบระเบียน ซึ่งคำนึงถึงบูรณภาพของการอ้างอิง

นอกจากนี้ ยังได้มีการนิยามชนิดข้อมูลเพิ่มเติม คือ ชนิดข้อมูลตัวอักษร ชนิดข้อมูลสายอักษร และชนิดข้อมูลบูลีน เพื่อให้สัญกรณ์เซดซึ่งเป็นผลของการแปลงที่ได้มีความสมบูรณ์ยิ่งขึ้น

โปรแกรมที่พัฒนาได้รับการทดสอบ และผลลัพธ์ได้รับการยืนยันว่ามีไวยากรณ์ถูกต้องจากโปรแกรมพิสูจน์ Z/EVES

ภาควิชา วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต วัชรวรรณ ศรีรัตนาลัย
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์	ลายมือชื่ออาจารย์ที่ปรึกษา วิวัฒน์ วัฒนาวุฒิ
ปีการศึกษา 2542	ลายมือชื่ออาจารย์ที่ปรึกษาร่วม

##4170509921 : MAJOR COMPUTER SCIENCE
KEYWORD : FORMAL SPECIFICATION / Z NOTATION / ENTITY RELATIONSHIP MODEL
VACHARAWAN SRIRATANALAI : A TOOL FOR TRANSLATING THE ENTITY RELATIONSHIP MODEL TO FORMAL SPECIFICATION IN Z NOTATION.
THESIS ADVISOR : WIWAT VATANAWOOD, 86 pp. ISBN 974-333-634-6.

This thesis designs algorithms and develops a tool for translating entity relationship model to formal specification in Z notation by using data dictionary and relationship dictionary as inputs. The algorithms are divided into 2 categories: structure property synthesizing algorithms and behavior property synthesizing algorithms while structure property synthesizing algorithms define entity schema, entity extension schema, and relationship schema. Behavior property synthesizing algorithms define insert schema, update schema, and delete schema with referential integrity.

Moreover, additional data types are defined such as character, string, and boolean which make the results of translation scheme, Z notation, completed.

The translation tool is tested and the results are syntactically verified using Z/EVES checker program.

ภาควิชา วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต วัชรวรรณ ศรีรัตนาลัย
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์	ลายมือชื่ออาจารย์ที่ปรึกษา วิวัฒน์ วัฒนวูด
ปีการศึกษา 2542	ลายมือชื่ออาจารย์ที่ปรึกษาร่วม

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ได้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของ อ.วิวัฒน์ วัฒนาวุฒิ อาจารย์ที่ปรึกษา และขอขอบคุณ รองศาสตราจารย์ ดร. วันชัย ริ้วไพบูลย์ อ.ดร. ธาราทิพย์ สุวรรณศาสตร์ และ อ.ดร. พรศิริ หมื่นไชยศรี กรรมการวิทยานิพนธ์ที่กรุณาเสียสละเวลาให้คำแนะนำ ตรวจและแก้ไขต้นฉบับวิทยานิพนธ์

ขอขอบคุณ เพื่อน ๆ กลุ่ม Formal ในห้องปฏิบัติการวิศวกรรมซอฟต์แวร์ที่เสียสละเวลาในการให้คำปรึกษา และช่วยตรวจสอบผลการวิจัยที่ได้ ขอขอบคุณเพื่อน ๆ ที่ให้กำลังใจและข้อเสนอแนะต่าง ๆ และ ขอขอบคุณท่านอื่น ๆ ที่มีส่วนช่วยในการทำวิทยานิพนธ์ที่ไม่ได้กล่าวนามมา ณ โอกาสนี้ด้วย

สุดท้ายนี้ ผู้วิจัยใคร่ขอกราบขอบพระคุณ บิดา มารดาที่สนับสนุนในด้านต่าง ๆ และให้กำลังใจแก่ผู้วิจัยเสมอมา

วัชรวรรณ ศรีรัตนาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ฌ

สารบัญรูป ..... ญ

บทที่ 1 บทนำ 1

1.1. ความเป็นมาและความสำคัญของปัญหา 1

1.2. วัตถุประสงค์ 1

1.3. ขอบเขตการวิจัย 2

1.4. ขั้นตอนการวิจัย 2

1.5. ประโยชน์ที่จะได้รับ 2

บทที่ 2 งานวิจัยและทฤษฎีที่เกี่ยวข้อง 4

2.1. งานวิจัยที่เกี่ยวข้อง 4

2.2. ทฤษฎีที่เกี่ยวข้อง 5

บทที่ 3 ขั้นตอนวิธีแปลงแบบจำลองเอนทิตี และความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด .. 16

3.1. ส่วนการสังเคราะห์คุณสมบัติโครงสร้าง 17

3.2. ส่วนการสังเคราะห์คุณสมบัติพฤติกรรม 23

บทที่ 4 การพัฒนาเครื่องมือซอฟต์แวร์แปลงแบบจำลองเอนทิตีเป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด. 27

4.1. ส่วนรับข้อมูล 27

4.2. ส่วนสังเคราะห์สัญกรณ์เซด 29

4.3. ส่วนบันทึกข้อมูล 29

4.4. ผังโครงสร้างของระบบ 32

4.5 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือซอฟต์แวร์ 35

บทที่ 5 การทดสอบ และสรุปผล 36

5.1. ขั้นตอนการติดตั้ง 36

5.2. สภาวะที่ใช้ทดสอบโปรแกรม 36
5.3. กรณีทดสอบที่ใช้ทดสอบโปรแกรม 36
5.4. ขั้นตอนการทดสอบ 38
5.5. ผลการทดสอบโปรแกรม 38
5.6. ผลการทดสอบข้อกำหนดรูปนัยด้วย Z/EVES 41
5.7. การตรวจสอบความหมายของผลลัพธ์สัญกรณ์เซต 42
5.8. การทดสอบกับแบบจำลองเอนทิตีและความสัมพันธ์ที่ไม่เป็นนอร์มอลฟอร์มลำดับที่สาม 46

บทที่ 6 สรุปผลการวิจัย 49
6.1. สรุปผลการวิจัย 49
6.2. ประโยชน์ของเครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซต 50
6.3. ปัญหา และข้อจำกัดที่ได้พบจากการวิจัย 50

รายการอ้างอิง 51
ภาคผนวก 52
ภาคผนวก ก. ระบบให้เช่ารถแวน 53
ภาคผนวก ข. ระบบการขายของผ่านโทรศัพท์ 63
ประวัติผู้เขียน 86

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงเครื่องหมายสำหรับประพจน์.....13
ตารางที่ 4.1 แสดงรูปแบบแฟ้มข้อความสำหรับพจนานุกรมข้อมูล.....27
ตารางที่ 4.2 แสดงรูปแบบแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์.....28
ตารางที่ 5.1 แสดงสรุปผลการตรวจสอบผลลัพธ์สัญกรณ์เซตของระบบให้เช่ารถแวน.....44
ตารางที่ 5.2 แสดงสรุปผลการตรวจสอบผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์.....46

สารบัญรูป

หน้า

รูปที่ 2.1 รูปแสดงวัฏจักรการพัฒนาซอฟต์แวร์ 6
รูปที่ 2.2 รูปแสดงสัญลักษณ์ที่ใช้ในแผนภาพเอนทิตีและความสัมพันธ์ 7
รูปที่ 2.3 รูปแสดงโครงร่างเหมือนกล่อง 8
รูปที่ 2.4 รูปแสดงตัวอย่างโครงร่างเหมือนกล่อง 8
รูปที่ 2.5 รูปแสดงการรวมเค้าร่าง 9
รูปที่ 2.6 รูปแสดงตัวอย่างผลที่ได้จากการรวมเค้าร่าง 9
รูปที่ 2.7 รูปแสดงการแมพปิงของฟังก์ชัน 14
รูปที่ 3.1 แผนภาพแสดงขั้นตอนการทำงานของโปรแกรม 16
รูปที่ 3.2 รูปแบบเค้าร่างเอนทิตี 20
รูปที่ 3.3 รูปแบบเค้าร่างเพิ่มเติม 21
รูปที่ 3.4 รูปแบบเค้าร่างความสัมพันธ์ 22
รูปที่ 3.5 รูปแบบเค้าร่างการเพิ่ม 23
รูปที่ 3.6 รูปแบบเค้าร่างการแก้ไข 25
รูปที่ 3.7 รูปแบบเค้าร่างการลบ (ส่วนเอนทิตีลูก) 26
รูปที่ 3.8 รูปแบบเค้าร่างการลบ (ส่วนเอนทิตีพ่อ) 26
รูปที่ 4.1 รูปแสดงส่วนหนึ่งของแผนภาพเอนทิตีและความสัมพันธ์ของระบบให้เช่ารถแวน 28
รูปที่ 4.2 ตัวอย่างแฟ้มข้อความสำหรับพจนานุกรมข้อมูล 28
รูปที่ 4.3 ตัวอย่างแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ 29
รูปที่ 4.4 รูปแสดงผังงานการสังเคราะห์สัญกรณ์เซต 31
รูปที่ 4.5 รูปแสดงสัญกรณ์เซตที่เขียนแบบโครงร่างเหมือนกล่อง 30
รูปที่ 4.6 ผังโครงสร้างแสดงส่วนการทำงานต่าง ๆ ของระบบ 33
รูปที่ 5.1 แผนภาพเอนทิตี และความสัมพันธ์ของระบบให้เช่ารถแวน 37
รูปที่ 5.2 แผนภาพเอนทิตี และความสัมพันธ์ของระบบการขายของผ่านโทรศัพท์ 37
รูปที่ 5.3 หน้าจอเครื่องมือ 38
รูปที่ 5.4 หน้าจอเครื่องมือเมื่อทำการสร้างสัญกรณ์เซตเสร็จสมบูรณ์ 39
รูปที่ 5.5 หน้าจอเครื่องมือเมื่อข้อมูลนำเข้าไม่สมบูรณ์ 40
รูปที่ 5.6 แสดงเค้าร่าง van 41
รูปที่ 5.7 แสดงเค้าร่าง vanExt 41
รูปที่ 5.8 แสดงเค้าร่าง customer 42
รูปที่ 5.9 แสดงเค้าร่าง customerExt 42
รูปที่ 5.10 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูล 46

รูปที่ 5.11 รูปแสดงผลลัพธ์สัญกรณ์เซดของตารางความสัมพันธ์ที่ไม่ได้เป็นนอร์มอลฟอร์มลำดับที่สาม.47
รูปที่ ก-1 แผนภาพเอนทิตี และความสัมพันธ์ของระบบให้เช่ารถแวน 53
รูปที่ ก-2 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลของระบบให้เช่ารถแวน 53
รูปที่ ก-3 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ของระบบให้เช่ารถแวน 54
รูปที่ ก-4 ผลลัพธ์สัญกรณ์เซดของระบบให้เช่ารถแวน 54
รูปที่ ก-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบให้เช่ารถแวนด้วย Z/EVES 59
รูปที่ ข-1 แผนภาพเอนทิตี และความสัมพันธ์ของระบบการขายของผ่านโทรศัพท์ 63
รูปที่ ข-2 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลของระบบการขายของผ่านโทรศัพท์ 66
รูปที่ ข-3 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ของระบบการขายของผ่านโทรศัพท์66
รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซดของระบบการขายของผ่านโทรศัพท์ 67
รูปที่ ข-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบการขายของผ่านโทรศัพท์ด้วย Z/EVES 79

บทที่ 1
บทนำ

1.1. ความเป็นมาและความสำคัญของปัญหา

ข้อกำหนดรูปนัย (Formal specification) คือการทำข้อกำหนดแบบหนึ่งโดยข้อกำหนดจะอยู่ในรูปแบบทางคณิตศาสตร์ ซึ่งทำให้สามารถอนุมาน (Infer) ถ้อยแถลง (Statement) ใหม่จาก ถ้อยแถลงเดิมได้ มีความเที่ยงตรง ไม่กำกวม และสามารถพิสูจน์ความสอดคล้องของข้อกำหนดได้ อย่างไรก็ตาม ผู้ที่ไม่มีความรู้ในด้านนี้ และทางคณิตศาสตร์จะพัฒนาและเข้าใจความหมายที่แสดงได้ยาก ทั้งนี้วิธีในการทำข้อกำหนดที่ใช้กันมากคือข้อกำหนดกึ่งรูปนัย (Semi-formal specification) ซึ่งมีลักษณะเป็นแผนภาพ (Diagram) ทำให้สามารถเข้าใจความหมายของข้อกำหนดได้ง่าย เช่น แผนภาพเอนทิตีและความสัมพันธ์ (Entity Relationship Diagram) โดยในการทำแผนภาพเอนทิตีและความสัมพันธ์จะอยู่ในขั้นตอนของการวิเคราะห์และออกแบบระบบ เพื่อพิจารณาว่าระบบมีเอนทิตีใดเกี่ยวข้องบ้าง และเกี่ยวข้องกันอย่างไร อย่างไรก็ดี ข้อกำหนดแบบกึ่งรูปนัยก็มีข้อเสียคือ ในบางส่วนของข้อกำหนดอาจอธิบายในรูปของภาษาธรรมชาติ (Natural language) ซึ่งเป็นเหตุให้เกิดความกำกวมในข้อกำหนด และเกิดความคลาดเคลื่อนในการสื่อสารระหว่างบุคคลได้ เพราะข้อความหนึ่ง ๆ อาจแปลความหมายได้แตกต่างกัน

ในปัจจุบันมีหลายองค์กรเริ่มให้ความสนใจกับการทำข้อกำหนดรูปนัยและเริ่มหันมาใช้การทำข้อกำหนดความต้องการแบบนี้ในการทำข้อกำหนดความต้องการ [1] เช่น บริษัท ไอบีเอ็ม (IBM) ในโครงการซีไอซีเอส (CICS: Customer Information Control System) ซึ่งจากโครงการนี้ทำให้สามารถลดจำนวนบรรทัดของโปรแกรม (Line of code) จากเดิม 90,000 บรรทัด เหลือเพียง 2,000 บรรทัด นอกจากนี้ บริษัท บริทิชเทเลคอมพ์ (British Telecom)และ บริษัท เพลสซี่ (Plessey) ก็เริ่มหันมาใช้ การทำข้อกำหนดรูปนัยแล้ว

จากที่ได้กล่าวมาแสดงให้เห็นว่า ข้อกำหนดรูปนัยเป็นข้อกำหนดที่เหมาะสมในการทำข้อกำหนดความต้องการของระบบ หากแต่ว่า ยากในการเขียน และทำความเข้าใจ ส่วนข้อกำหนดกึ่งรูปนัยเป็นข้อกำหนดที่ใช้กันมากในปัจจุบัน ถ้าหากจะต้องปรับเปลี่ยนไปใช้การทำข้อกำหนดรูปนัยในทันที อาจทำให้ต้องเสียเวลาในการเรียนรู้ และเกิดความท้อในการศึกษา ดังนั้นในวิทยานิพนธ์นี้จึงต้องการนำเสนอการแปลงข้อมูลจากแบบจำลองเอนทิตีและความสัมพันธ์ ซึ่งเป็นข้อกำหนดกึ่งรูปนัยแบบหนึ่งเป็นสัญกรณ์เซด (Z notation) ซึ่งเป็นข้อกำหนดรูปนัย พร้อมกันนี้ได้เสนอวิธีในการสร้างชุดคำสั่งพื้นฐาน การเพิ่ม การแก้ไข และการลบ ที่มักใช้กันสำหรับการทำงานกับเอนทิตีด้วย

1.2. วัตถุประสงค์

1) เพื่อออกแบบขั้นตอนวิธีในการสร้างข้อกำหนดรูปนัยของโครงสร้างและพฤติกรรมพื้นฐานของระบบจากแบบจำลองเอนทิตีและความสัมพันธ์

2) เพื่อออกแบบ และพัฒนาเครื่องมือซอฟต์แวร์สนับสนุนการสร้างข้อกำหนดรูปนัยสัญกรณ์เซด

1.3. ขอบเขตการวิจัย

1) ข้อมูลนำเข้าอยู่ในรูปของตารางพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ โดยที่ตารางพจนานุกรมข้อมูลจะต้องมีรายการข้อมูล ชื่อเอนทิตี ชื่อลักษณะประจำ ชนิดของลักษณะประจำ คีย์ของเอนทิตี และเงื่อนไขอื่น ๆ ซึ่งอยู่ในรูปที่อธิบายด้วยแทคของลาเท็กซ์(LATEX) และตารางพจนานุกรมความสับพันธ์จะต้องมีรายการข้อมูล ชื่อความสัมพันธ์ เอนทิตีที่เกี่ยวข้องในความสัมพันธ์ ลักษณะความสัมพันธ์ และเงื่อนไขอื่น ๆ ในรูปที่อธิบายด้วยแทคของลาเท็กซ์
2) ความสัมพันธ์ของเอนทิตีต้องอยู่ในรูปนอร์มอลฟอร์มลำดับที่สาม (Third normal form) และเป็นความสัมพันธ์ทวิภาค (Binary relation) เฉพาะความสัมพันธ์แบบหลายต่อหนึ่ง (many to one) หนึ่งต่อหลาย (one to many) และ หนึ่งต่อหนึ่ง (one to one)
3) ภาษาข้อกำหนดที่ใช้คือสัญกรณ์เซด โดยแทนค่าสัญกรณ์เซดด้วยแทคในลาเท็กซ์
4) ข้อกำหนดรูปนัยที่ได้จะประกอบด้วยส่วนโครงสร้าง และคำสั่งพื้นฐาน คือ การเพิ่ม การลบ และการแก้ไข
5) เครื่องมือซอฟต์แวร์ทำงานบนระบบปฏิบัติการวินโดว์ส (Windows)

1.4. ขั้นตอนการวิจัย

1) ศึกษา และวิเคราะห์งานวิจัยต่าง ๆ ที่เกี่ยวข้อง
2) ศึกษาสัญกรณ์เซด ได้แก่ การศึกษาถึงลักษณะ วิธีการเขียน และเครื่องหมายต่างๆ ที่มีในสัญกรณ์เซด
3) ออกแบบวิธีการสร้างข้อกำหนดจากแบบจำลองเอนทิตีและความสัมพันธ์เป็นสัญกรณ์เซด
4) ออกแบบข้อกำหนดรูปนัยของชุดคำสั่งพื้นฐาน ได้แก่คำสั่งสำหรับการเพิ่ม ลบ และ แก้ไข สำหรับแต่ละเอนทิตีที่เกี่ยวข้อง
5) ออกแบบรูปแบบข้อมูลนำเข้า คือการศึกษาถึงลักษณะของข้อมูลนำเข้า รวมทั้งวิธีการนำเข้า
6) พัฒนาโปรแกรม
7) ทดสอบโปรแกรม
8) สรุปงานวิจัย และจัดทำเอกสาร

1.5. ประโยชน์ที่จะได้รับ

1) ได้แนวทางในการสร้างข้อมูลจากแบบจำลองเอนทิตีและความสัมพันธ์ในรูปของพจนานุกรมข้อมูล และ พจนานุกรมความสัมพันธ์ให้อยู่ในรูปของแทคสัญลักษณ์เซด

2) ได้แนวทางในการนิยามชุดคำสั่งพื้นฐาน เพิ่ม ลบ และ แก้ไข สำหรับเอนทิตีต่าง ๆ ให้อยู่ในรูปแทคของสัญกรณ์เซด

บทที่ 2
งานวิจัยและทฤษฎีที่เกี่ยวข้อง

2.1. งานวิจัยที่เกี่ยวข้อง

1) การสร้างข้อกำหนดรูปนัยจากนิยามความต้องการโดยอัตโนมัติ (Automatic Generation of Formal Specification From Requirements Definitions) [2] โดย Lingzi Jin และ Hong Zhu

งานวิจัยนี้เป็นการนำเสนอระบบเอ็นดีอาร์เอเอสเอส(NDRASS)ซึ่งเป็นระบบที่สร้าง (Generate) ข้อกำหนดรูปนัยในรูปสัญกรณ์เซดจากนิยามความต้องการ(Requirements Definition) ซึ่งอยู่ในรูปของเอ็นดีอาร์ดีแอล (NDRDL) ซึ่งเป็นภาษาสำหรับนิยามความต้องการภาษาหนึ่ง โดยเอ็นดีอาร์ดีแอลสามารถเก็บนิยามความต้องการในรูปแบบต่าง ๆ ได้ดังนี้

- คำอธิบายในแบบอรูปนัย (Informal Description)
- แผนภาพ ได้แก่ แผนภาพเอนทิตีและความสัมพันธ์ (Entity Relationship Diagram) แผนภาพแสดงการไหลของข้อมูล (Data Flow Diagram) และ แผนภาพแสดงการควบคุม (Control Flow Diagram)
- พจนานุกรม (Dictionary) ได้แก่ พจนานุกรมข้อมูล (Data Dictionary) พจนานุกรมความสัมพันธ์(Relationship Dictionary) และ พจนานุกรมการปฏิบัติการ (Operation Dictionary)

เอ็นดีอาร์เอเอสเอสได้เสนอวิธีในการแปลงแบบจำลองเอนทิตีและความสัมพันธ์ให้อยู่ในรูปของสัญกรณ์เซด โดยอาศัยข้อมูลจากพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ โดยได้แบ่งโครงสร้างได้เป็น 2 ส่วน คือ ส่วนนิยามเค้าร่างเอนทิตี (Entity schema) และ ส่วนนิยามเค้าร่างความสัมพันธ์ (Relationship schema)

ส่วนนิยามเค้าร่างเอนทิตี คือส่วนที่นิยามถึงโครงสร้างของเอนทิตี ว่าประกอบไปด้วยลักษณะประจำอะไรบ้าง และ มีเงื่อนไขของลักษณะประจำอย่างไร โดยในส่วนนี้ไม่ได้มีการกล่าวถึงเงื่อนไขอื่น ๆ สำหรับเอนทิตี เช่น คีย์หลัก (Primary key) และ คีย์นอก (Foreign key) เป็นต้น ซึ่งเป็นส่วนสำคัญสำหรับงานทางด้านฐานข้อมูล ในส่วนของการนิยามเค้าร่างความสัมพันธ์นั้นเป็นการนิยามถึงความสัมพันธ์ต่าง ๆ ระหว่างเอนทิตีใด ๆ ซึ่งในส่วนนี้ได้นิยามเค้าร่างเป็นแบบครอบคลุม (Global) คือการนิยามเค้าร่างเพื่อให้เค้าร่างอื่นสามารถเรียกใช้ตัวแปรที่ประกาศไว้ในเค้าร่างนี้ได้

2) การทำข้อกำหนดกึ่งรูปนัยให้สมบูรณ์ด้วยเซด (Complementing Semi-Formal Specification with Z) [3] โดย Yves Ledru

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

งานวิจัยนี้เป็นงานวิจัยเกี่ยวกับการทำข้อกำหนดกึ่งรูปนัยให้สมบูรณ์ด้วยเซตเนื่องจากข้อกำหนดกึ่งรูปนัยอาจไม่สามารถแสดงรายละเอียดบางอย่างได้อย่างสมบูรณ์ในตัวเอง แต่จะแสดงโดยการให้รายละเอียดในรูปของการบรรยายเป็นภาษาธรรมชาติ ดังนั้นในงานวิจัยนี้จึงเสนอสิ่งที่ควรนิยามเพิ่มเติมเพื่อให้แผนภาพเอนทิตีและความสัมพันธ์มีความสมบูรณ์ยิ่งขึ้นโดยได้ยกตัวอย่างส่วนที่ควรนิยามเป็นสัญกรณ์เซตดังนี้

ก) ชนิดของข้อมูล (Data type) โดยการกำหนดชนิดของข้อมูลอย่างง่าย ๆ เป็น กีเวนเซต (Given set) โดยตั้งเป็นชื่อที่มีความหมาย

ข) ลักษณะประจำแยกส่วน และลักษณะประจำหลายค่า (Partial and multi-valuated attributes) คือการกำหนดค่าในแต่ละลักษณะประจำว่า ณ เวลาหนึ่ง ๆ ลักษณะประจำใดจำเป็นต้องมีการระบุค่า หรือลักษณะประจำใดไม่จำเป็นต้องมีการระบุค่า

ค) เงื่อนไขต่าง ๆ ที่ควรนิยามในสัญกรณ์เซต ซึ่งแบ่งได้เป็น

- เงื่อนไขสำหรับชนิดของเอนทิตี (Constraint on entity types) เป็นเงื่อนไขเกี่ยวกับค่าที่เป็นไปได้ของลักษณะประจำของชนิดของเอนทิตี เช่น อายุต้องมีค่ามากกว่า 0
- เงื่อนไขสำหรับส่วนเพิ่มเติม (Constraint on extension) เป็นเงื่อนไขซึ่งนิยามถึงสิ่งที่ไม่สามารถนิยามได้ในชนิดของเอนทิตี เนื่องจากมีการอ้างอิงถึงตัวชนิดของเอนทิตี เซตของเอนทิตีในฐานข้อมูลก็เป็นการนิยามเพิ่มเติมจากชนิดของเอนทิตีที่ได้มีการนิยามไว้แล้ว นอกจากนี้ การนิยามคีย์หลักก็ถือเป็นการนิยามเพิ่มเติมเช่นเดียวกัน
- เงื่อนไขสำหรับความสัมพันธ์ (Constraint on relation) เป็นการนิยามเงื่อนไขสำหรับความสัมพันธ์หนึ่ง ๆ
- เงื่อนไขสำหรับฐานข้อมูล (Constraint on whole database) เป็นการนิยามเงื่อนไขที่เกี่ยวข้องกับหลาย ๆ ความสัมพันธ์ในฐานข้อมูล

จากงานวิจัยทั้งสองที่ได้กล่าวมา งานวิจัยของ Lingzi Jin และ Hong Zhu [2] ช่วยให้เห็นแนวคิดพื้นฐานในการแปลงข้อมูลในพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ให้อยู่ในรูปของ สัญกรณ์เซต และงานวิจัยของ Yves Ledru [3] ทำให้รู้ถึงสิ่งที่ควรจะนิยามเพิ่มเติมเพื่อให้เกิดความสมบูรณ์ต่อแผนภาพเอนทิตีและความสัมพันธ์

2.2. ทฤษฎีที่เกี่ยวข้อง

ในงานวิทยานิพนธ์นี้ มีทฤษฎีต่าง ๆ ที่เกี่ยวข้องดังนี้

1) วัฏจักรการพัฒนาซอฟต์แวร์ (Software Development Life-cycle) [4]

เป็นขั้นตอนในการพัฒนาซอฟต์แวร์ ซึ่งสามารถแสดงแผนภาพได้ ดังรูปที่ 2.1 ในที่นี้แสดงโดยแบบจำลองน้ำตก (Waterfall model) ซึ่งมีขั้นตอนดังนี้

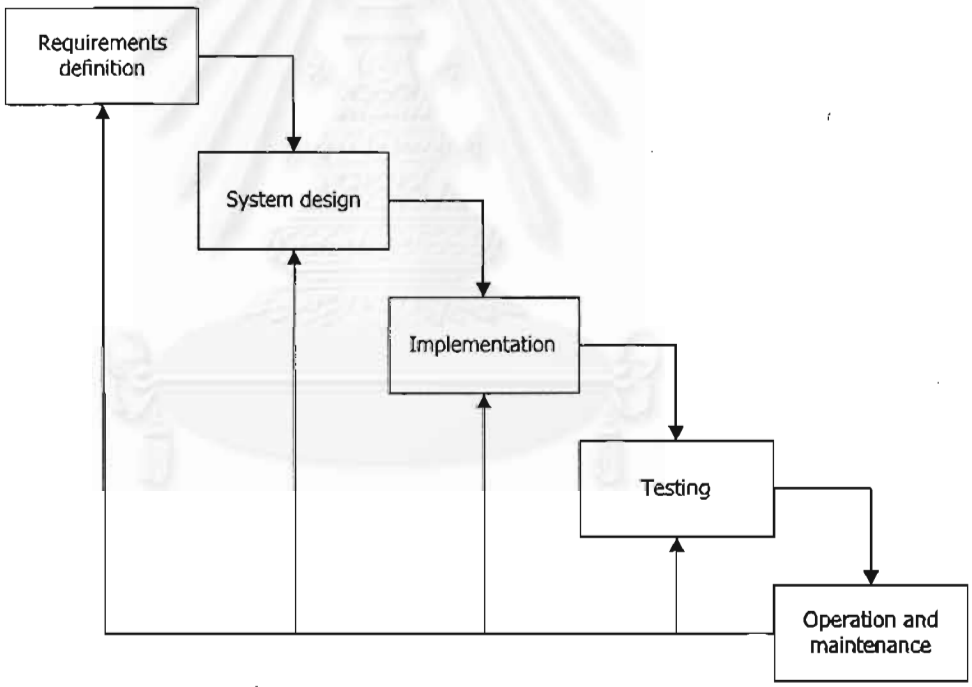
1.1) การนิยามความต้องการ เป็นส่วนที่ทำความเข้าใจกับระบบว่าระบบจะต้องสามารถทำอะไรได้บ้าง เงื่อนไขต่าง ๆ และจุดประสงค์ของระบบ ซึ่งผลที่ได้จากขั้นตอนนี้คือ ความต้องการต่าง ๆ ของระบบตามที่ลูกค้าต้องการ

1.2) การออกแบบ (System design) เป็นส่วนของการออกแบบระบบจากความต้องการต่าง ๆ ของระบบที่ได้จากขั้นตอนแรก เพื่อนำไปพัฒนาให้เป็นโปรแกรมต่อไป โดยผลที่ได้จากการออกแบบจะจัดทำเป็นเอกสารเพื่อใช้เป็นส่วนหนึ่งในการทำโครงแบบซอฟต์แวร์

1.3) การเขียนโปรแกรม (Implementation) เป็นส่วนที่เอาผลที่ได้จากขั้นการออกแบบมาใช้ในการเขียนเป็นโปรแกรม

1.4) การทดสอบ (Testing) เป็นการตรวจสอบการทำงานของโปรแกรมว่าทำงานถูกต้องตามความต้องการหรือไม่

1.5) การใช้งาน และบำรุงรักษา (Operation and maintenance) เป็นส่วนที่ทำการติดตั้งระบบเพื่อใช้งานจริง และแก้ไขข้อผิดพลาดต่าง ๆ ที่เกิดขึ้น รวมทั้งปรับปรุงระบบเมื่อมีความต้องการใหม่เพิ่มขึ้น

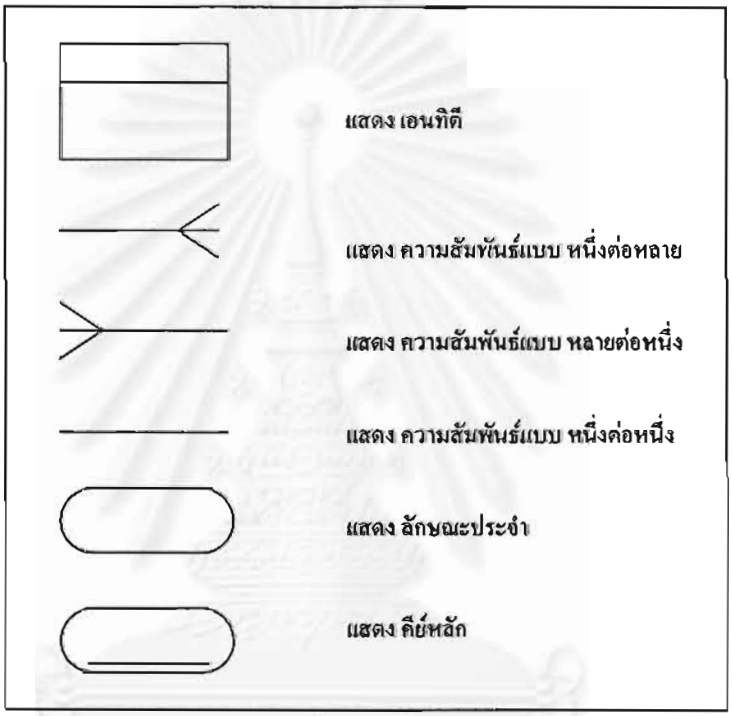


รูปที่ 2.1 รูปแสดงวัฏจักรการพัฒนาซอฟต์แวร์

2) แผนภาพเอนทิตีและความสัมพันธ์ [5],[6]

แผนภาพเอนทิตีและความสัมพันธ์เป็นข้อกำหนดกึ่งรูปนัยแบบหนึ่ง โดยมีลักษณะเป็นแผนภาพที่อธิบายข้อมูลในรูปของเอนทิตี (Entity) ความสัมพันธ์ (Relationship) และลักษณะประจำ (Attribute) โดย สิ่งที่แผนภาพเอนทิตีและความสัมพันธ์ต้องการแสดงถึงคือ เอนทิตีซึ่งแสดงถึงสิ่งใดสิ่งหนึ่งในโลกที่กำลังสนใจอยู่ ทั้งนี้แต่ละเอนทิตีต่างประกอบไปด้วยลักษณะประจำ ซึ่งเป็นตัว

อธิบายเอนทิตีนั้นๆ และแต่ละเอนทิตีก็มีค่าสำหรับแต่ละลักษณะประจำด้วย โดยข้อมูลต่างๆ เกี่ยวกับแผนภาพเอนทิตีและความสัมพันธ์จะถูกอธิบายใน พจนานุกรมข้อมูล และ พจนานุกรมความสัมพันธ์ นอกจากนี้แผนภาพเอนทิตีและความสัมพันธ์ยังแสดงถึงความสัมพันธ์ระหว่างเอนทิตี ซึ่งเป็นการมองภาพโดยรวมของระบบ ว่าระบบประกอบด้วยส่วนใดบ้าง และ แต่ละส่วนมีความสัมพันธ์กันอย่างไร ทำให้เราสามารถมองภาพกว้าง ๆ ของระบบออก ทั้งนี้ขั้นตอนของการทำแผนภาพเอนทิตีและความสัมพันธ์เป็นส่วนหนึ่งของการนิยามความต้องการและการออกแบบระบบ โดยสามารถแสดงสัญลักษณ์ที่ใช้ในแผนภาพเอนทิตีและความสัมพันธ์ได้ดังรูปที่ 2.2



รูปที่ 2.2 รูปแสดงสัญลักษณ์ที่ใช้ในแผนภาพเอนทิตีและความสัมพันธ์

3) การนอร์มอลไรซ์ (Normalization) [7] คือ การจำแนกโครงสร้างของข้อมูลเป็นโครงสร้างใหม่เพื่อให้เกิดความสอดคล้องกันภายใน (Internal consistency) ลดความซ้ำซ้อนของข้อมูล และให้เกิดความมั่นคงต่อข้อมูล ในการทำนอร์มอลไรซ์สามารถทำได้หลายระดับ โดยในที่นี้จะกล่าวถึงเฉพาะ 3 ระดับแรก ได้แก่

- นอร์มอลฟอร์มลำดับที่หนึ่ง (First normal form) ตารางความสัมพันธ์ (Relation) จะเป็นนอร์มอลฟอร์มลำดับที่หนึ่งก็ต่อเมื่อในทุกๆ ลักษณะประจำของแต่จะเถาวมีค่าเพียงค่าเดียว
- นอร์มอลฟอร์มลำดับที่สอง(Second normal form)ตารางความสัมพันธ์จะเป็นนอร์มอลฟอร์มลำดับที่สองก็ต่อเมื่อตารางความสัมพันธ์นั้นเป็นนอร์มอลฟอร์มลำดับที่หนึ่งและลักษณะประจำที่ไม่ได้เป็นคีย์ขึ้นอยู่ที่ลักษณะประจำที่เป็นคีย์ทั้งหมด

- นอร์มอลฟอร์มลำดับที่สาม ตารางความสัมพันธ์จะเป็นนอร์มอลฟอร์มลำดับที่สามก็ต่อเมื่อตารางความสัมพันธ์นั้นเป็นนอร์มอลฟอร์มลำดับที่สองและลักษณะประจำที่ไม่ได้เป็นคีย์ต้องไม่ขึ้นกับลักษณะประจำอื่นที่ไม่ได้เป็นคีย์เช่นเดียวกัน

4) ข้อกำหนดรูปนัย [8]

ข้อกำหนดรูปนัยเป็นการแสดงข้อกำหนดความต้องการในรูปของภาษาที่ได้มีการอธิบายประมวลศัพท์ (Vocabulary) วากยสัมพันธ์ (Syntax) และ ความหมาย (Semantic) ไว้อย่างเป็นระเบียบ คือ มีลักษณะเป็นกลุ่มของตัวอักษรที่แสดงสัญลักษณ์ทางคณิตศาสตร์ ในรูปของตรรกศาสตร์ภาคแสดงลำดับแรก (First order predicate logic) จึงทำให้ ความหมายที่ได้นั้นชัดเจน และแน่นอน และจากลักษณะของภาษาที่อยู่ในรูปแบบทางคณิตศาสตร์จึงทำให้สามารถพิสูจน์ความสอดคล้อง (Consistency) ของข้อกำหนดโดยใช้เครื่องมือทางคณิตศาสตร์ได้

ข้อดีของการทำข้อกำหนดรูปนัย [1] มีดังนี้ คือ

- ไม่กำกวม
- สามารถพิสูจน์ความสอดคล้องของข้อกำหนดได้
- ช่วยให้เข้าใจในการทำงานของระบบได้ดียิ่งขึ้นตั้งแต่ในช่วงเริ่มต้นของการออกแบบ
- ช่วยให้มีแนวคิดในการออกแบบระบบได้มากขึ้น
- ช่วยลดความเป็นไปได้ที่จะเกิดความผิดพลาด (Error) ของการออกแบบระบบได้
- ช่วยลดค่าใช้จ่ายในการพัฒนาระบบ เนื่องจากความผิดพลาดต่าง ๆ สามารถตรวจพบได้ในขั้นตอนของการออกแบบ ซึ่งทำให้สามารถแก้ไขได้โดยเสียค่าใช้จ่ายไม่มาก ถ้าเทียบกับการตรวจพบความผิดพลาดในขั้นตอนท้าย ๆ ของการพัฒนาระบบ

ข้อเสียของการทำข้อกำหนดรูปนัย คือ

- เข้าใจยาก เนื่องจากใช้รูปแบบทางคณิตศาสตร์ในการเขียนข้อกำหนด
- เขียนยากสำหรับผู้ที่ไม่มีความรู้ในด้านนี้มาก่อน และผู้ที่ไม่มีความรู้ทางคณิตศาสตร์

5) สัญกรณ์เซด [8], [9]

สัญกรณ์เซดเป็นการแสดงข้อกำหนดรูปนัยรูปแบบหนึ่งซึ่งได้รับการพัฒนาขึ้นโดยกลุ่มพีอาร์จี (PRG: Programming Research Group) ที่มหาวิทยาลัยอ๊อกซ์ฟอร์ด (Oxford University) เมื่อประมาณปีคริสตศักราช 1970 สัญกรณ์เซดเป็นข้อกำหนดที่อาศัยพื้นฐานของทฤษฎีเกี่ยวกับเซต และ ตรรกศาสตร์ภาคแสดงลำดับแรก โดยใช้สัญลักษณ์ทางคณิตศาสตร์ที่มีอยู่และเพิ่มสัญลักษณ์บางตัวเพื่อให้เหมาะกับการใช้งาน แต่ปัญหาของการใช้สัญลักษณ์ทางคณิตศาสตร์ในการเขียนข้อกำหนดคือ ยากต่อการจัดการโครงสร้าง และ อ่านเข้าใจได้ยาก ดังนั้นเซดจึงเพิ่มสัญกรณ์เค้าร่าง (Schema notation) ซึ่งเป็นโมดูล (Module) เพื่อให้ข้อกำหนดมีความเป็นโครงสร้างมากขึ้น โดยเราสามารถแบ่งประเภทของเค้าร่างได้เป็น 2 แบบคือ เค้าร่างสถานะ (State schema) ซึ่งเป็นการประกาศถึงคอมโพเนนท์

(Component) ต่างๆ ในระบบ และ เค้าร่างการปฏิบัติการ (Operation Schema) ซึ่งเป็นการประกาศถึงการปฏิบัติการ (Operation) ต่าง ๆ ในระบบ

โดยทั่วไปเค้าร่างจะประกอบด้วย 2 ส่วนคือ

- ส่วนการประกาศ (Declaration Part) เป็นส่วนประกาศตัวแปร และชนิดตัวแปรที่จะใช้
- ส่วนสัจพจน์ (Axiom Part) เป็นส่วนที่ระบุถึงภาคแสดงซึ่งอ้างอิงถึงตัวแปรที่ได้นิยามไว้ หรือ ตัวแปรอื่นที่อยู่ในขอบเขต

ในกรณีที่เป็นเค้าร่างสถานะ ส่วนสัจพจน์จะแสดงถึงเงื่อนไขของตัวแปร และในกรณีที่เป็นเค้าร่างการปฏิบัติการ ส่วนสัจพจน์จะระบุถึงพฤติกรรมการปฏิบัติการ (Operation behavior) ว่ามี การเปลี่ยนสถานะ (State transition) อย่างไร

การเขียนเค้าร่างสามารถเขียนได้หลายแบบ ได้แก่

- การเขียนเค้าร่างแบบ โครงร่างเหมือนกล่อง (Box-Like Framework)

  การเขียนเค้าร่างแบบโครงร่างเหมือนกล่องเป็นการเขียนเค้าร่างที่มีลักษณะเป็นกล่อง ซึ่งมีลักษณะดังรูปที่ 2.3 และในรูปที่ 2.4 แสดงตัวอย่างโครงร่างเหมือนกล่อง

รูปที่ 2.3 รูปแสดงโครงร่างเหมือนกล่อง

รูปที่ 2.4 รูปแสดงตัวอย่างโครงร่างเหมือนกล่อง

- การเขียนเค้าร่างแบบ ตามขวาง (Horizontal)

  การเขียนเค้าร่างแบบตามขวางเป็นการเขียนเค้าร่างที่มีลักษณะเป็นข้อความเขียนต่อเนื่องกันไป

  SchName ≙ Schema Expr

  เช่น

  TimeHMNat ≙ [hrs : ℕ; mins : ℕ | hrs < 23 ∧ mins ⩾ 59]

ในการเขียนเค้าร่างสามารถใช้วิธีการเขียนแบบใดก็ได้ไม่ต่างกัน อย่างไรก็ดีในการเขียนข้อกำหนดรูปนัยก็ควรจะมีข้อความเพื่ออธิบายความหมายของสัญลักษณ์ทางคณิตศาสตร์เหล่านั้นด้วย นอกจากนี้เซดยังประกอบด้วยส่วนอื่น ๆ ดังนี้

5.1) การตกแต่ง (Decoration)

การตกแต่ง คือการใส่ตัวอักษรพิเศษท้ายชื่อของวัตถุ (Object) เพื่อสื่อความหมายบางอย่าง การใช้การตกแต่งไม่ได้เป็นกฎที่ต้องปฏิบัติตาม แต่เป็นเพียงทางเลือกในการเขียนข้อกำหนดเพื่อให้เกิดความเข้าใจมากขึ้น ตัวอักษรพิเศษที่มักใช้ในมาตรฐานของสัญกรณ์เซด ได้แก่

? ใช้ต่อท้ายชื่อของตัวแปรเข้า (Input variable)

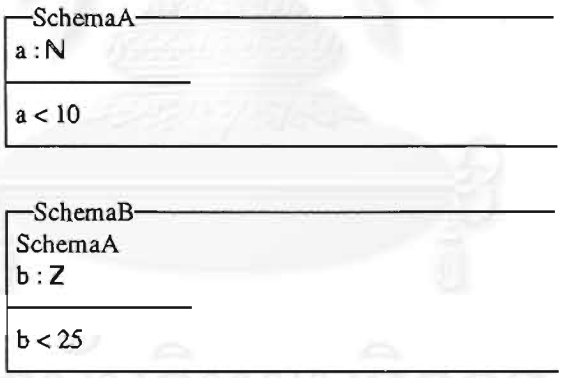
! ใช้ต่อท้ายชื่อของตัวแปรออก (Output variable)

′ ใช้ต่อท้ายชื่อของตัวแปรที่เป็นสถานะหลังการเปลี่ยนแปลง (After state variable)

5.2) การรวมเค้าร่าง (Schema Inclusion)

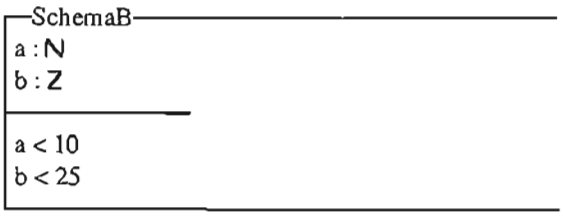
การรวมเค้าร่าง เป็นการนำข้อความของเค้าร่างอื่นรวมเข้ามากับเค้าร่างนี้ โดยการรวมจะใช้การอ้างถึงชื่อเค้าร่างที่ต้องการรวมเข้ามาในส่วนของการประกาศ ผลจากการรวมเค้าร่างคือส่วนของการประกาศก็จะรวมกับส่วนการประกาศและส่วนสัจพจน์ก็จะรวมกับส่วนสัจพจน์ตามลำดับเช่น

กำหนดเค้าร่าง SchemaA และ SchemaB ดังรูปที่ 2.5



รูปที่ 2.5 รูปแสดงการรวมเค้าร่าง

จากเค้าร่างที่กำหนดจะเห็นว่า SchemaB ได้รวมเอา SchemaA เข้าไปด้วยจากการอ้างถึงชื่อเค้าร่าง SchemaA ในส่วนการประกาศของ SchemaB ซึ่ง SchemaB อาจเขียนได้ใหม่โดยมีความหมายเดิมได้ดังรูปที่ 2.6



รูปที่ 2.6 รูปแสดงผลที่ได้จากการรวมเค้าร่าง

5.3) กีเวนเซต (Given Set)

กีเวนเซตจะเป็นการนิยามให้ชนิดข้อมูลนามธรรม (Abstract data type) ใด ๆ เป็นชนิดข้อมูล ซึ่งสามารถถูกใช้ได้ในการประกาศชนิดของตัวแปร เช่น

กำหนดกีเวนเซต

[NAME]

สามารถนิยามได้ว่า

a : NAME

b : NAME

โดยที่ a และ b เป็นตัวแปรที่มีชนิดเป็น NAME

5.4) เครื่องหมาย Δ และ Ξ

Δ เป็นการระบุว่าอาจมีการเปลี่ยนแปลงค่าของข้อมูลหลังจากทำงานตามคำสั่งแล้วกล่าวคือ สถานะก่อน (Before state) ไม่เท่ากับสถานะหลัง (After state)

$$\Delta State \mathrel{\widehat{=}} [State; State']$$

Ξ เป็นการระบุว่าไม่มีการเปลี่ยนแปลงค่าของข้อมูลหลังจากทำงานตามคำสั่งแล้วกล่าวคือ สถานะก่อนเท่ากับสถานะหลัง

$$\Xi State \mathrel{\widehat{=}} [\Delta State \mid v1' = v1 \wedge v2' = v2 \wedge \ldots vk' = vk]$$

ทั้งนี้ สถานะของข้อมูลสามารถแบ่งได้เป็น 2 สถานะ คือ

- สถานะก่อน คือสถานะหรือค่าของข้อมูลก่อนการทำงานตามคำสั่ง
- สถานะหลัง คือสถานะหรือค่าของข้อมูลหลังการทำงานตามคำสั่ง

5.5) ฟังก์ชันโพรเจคชัน (Projection function)

ฟังก์ชันโพรเจคชัน เป็นฟังก์ชันที่ทำการตัดส่วนใดส่วนหนึ่งของคู่ลำดับ ในสัญกรณ์เซดมีฟังก์ชันสำหรับการทำตัดคู่ลำดับ 2 ฟังก์ชัน คือ เฟิร์ส (first) และ เซคคันด์ (second)

- เฟิร์ส เป็นการตัดเพื่อเอาส่วนแรกของคู่ลำดับ
- เซคคันด์ เป็นการตัดเพื่อเอาส่วนที่สองของคู่ลำดับ

5.6) ชนิดข้อมูลฟรีไทป์ (Free type)

ชนิดข้อมูลฟรีไทป์ คือการกำหนดชนิดของข้อมูลใหม่ขึ้นมา พร้อมทั้งระบุถึงค่าที่เป็นไปได้ของชนิดข้อมูลนั้น ๆ เช่น

Boolean ::= True | False

เป็นการกำหนด Boolean ให้เป็นชนิดข้อมูลฟรีไทป์ โดยที่มีค่าที่เป็นไปได้คือ True หรือ False ซึ่ง True และ False เป็นค่าที่แตกต่างกัน อย่างไรก็ตาม ข้อจำกัดของการกำหนดชนิดข้อมูลฟรีไทปคือ ค่าของชนิดข้อมูลฟรีไทปจะไม่สามารถนำไปใช้เป็นชื่อตัวแปรได้อีก เช่น จากตัวอย่าง True และ False จะไม่สามารถถูกนำไปใช้เป็นชื่อตัวแปรได้อีก เป็นต้น

5.7) ตัวย่อ (Abbreviation)

ตัวย่อ คือการกำหนดตัวย่อสำหรับเรียกแทนค่าข้อมูลความหมายทางขวาของเครื่องหมาย $\widehat{=}$ เช่น

$X \widehat{=} 0..127$

เป็นการกำหนดให้ X แทนค่าตั้งแต่ 0 ถึง127 และ

$Y \widehat{=} a \longleftrightarrow b$

เป็นการกำหนดให้ Y แทนความสัมพันธ์ระหว่าง a และ b ในการกำหนดตัวย่อ ตัวแปรที่ใช้กำหนดเป็นตัวย่อแล้ว จะไม่สามารถนำไปใช้เป็นตัวแปรได้อีก เช่น จากตัวอย่าง X และ Y จะไม่สามารถถูกนำไปกำหนดเป็นตัวแปรอื่นได้อีก

6) ทฤษฎีทางคณิตศาสตร์ที่เกี่ยวข้อง [12]

เนื่องจากสัญกรณ์เซดเป็นสัญกรณ์ซึ่งตั้งอยู่บนพื้นฐานทางคณิตศาสตร์จึงมีทฤษฎีทางคณิตศาสตร์ ต่างๆ ที่เกี่ยวข้องดังนี้

6.1) ตัวบ่งปริมาณ (Quantifiers)

ตัวบ่งปริมาณ เป็นการบอกจำนวนของสิ่งที่เรากำลังสนใจ ได้แก่

- ตัวบ่งปริมาณทั้งหมด (Universal quantifier) หรือที่เรียกว่า ฟอร์ออล (For all) ใช้สัญลักษณ์ คือ $\forall$ เป็นการบ่งบอกว่าเรากำลังสนใจจำนวนทั้งหมด
- ตัวบ่งปริมาณบางส่วน (Existential quantifier) หรือที่เรียกว่า ฟอร์ซัม (For some) ใช้สัญลักษณ์คือ $\exists$ เป็นการบ่งบอกว่าเราสนใจข้อมูลเพียงบางส่วนเท่านั้น

6.2) ตรรกศาสตร์ (Logic)

ตรรกศาสตร์เป็นการกำหนดกฎและเทคนิคในการพิจารณาว่าอาร์กิวเมนต์ (Argument) นั้นถูกต้อง ในทางคณิตศาสตร์ ตรรกศาสตร์ถูกใช้ในการพิสูจน์ทฤษฎีต่างๆ ส่วนในทางคอมพิวเตอร์ ตรรกศาสตร์ถูกใช้ในการตรวจสอบความถูกต้องของโปรแกรมและใช้ในการพิสูจน์ทฤษฎี ตรรกศาสตร์สามารถแบ่งได้เป็น 2 ชนิดคือ

- ตรรกศาสตร์ประพจน์(Propositional logic) เป็นตรรกะซึ่งไม่มีตัวแปร เช่น P ∨ Q โดยที่ P และ Q เป็นประพจน์ เป็นต้น ทั้งนี้ เครื่องหมายสำหรับประพจน์มีดังตารางที่ 2.1

ตารางที่ 2.1 แสดงเครื่องหมายสำหรับประพจน์

สัญลักษณ์	ชื่อสัญลักษณ์	ความหมาย
¬	'not'	¬Pจะมีค่าความจริงเป็นจริง (True) เมื่อ P มีค่าเป็นผิด (False)
∧	'and'	P ∧ Q จะมีค่าความจริงเป็นจริง เมื่อ P และ Q มีค่าเป็นจริง
∨	'or'	P ∨ Q จะมีค่าความจริงเป็นจริง เมื่อ P หรือ Q มีค่าเป็นจริง
⇒	'implies'	P ⇒ Q จะมีค่าความจริงเป็นจริง เมื่อ P มีค่าเป็นเท็จ หรือ Q มีค่าเป็นจริง หรือ ทั้งสองอย่าง
⇔	'if and only if'	P ⇔ Q จะมีค่าความจริงเป็นจริง เมื่อ P และ Q มีค่าเหมือนกัน

- ตรรกศาสตร์ภาคแสดง (Predicate logic) เป็นตรรกะซึ่งมีตัวแปร ทำให้ไม่สามารถหาค่าความจริงได้ถ้าหากไม่รู้ค่าของตัวแปรนั้น เช่น x > 5 เป็นต้น ซึ่งเราสามารถเขียนภาคแสดงให้เป็นประพจน์ได้โดยการเพิ่มตัวบ่งปริมาณทั้งหมด หรือตัวบ่งปริมาณบางส่วนเข้าไปข้างหน้าของนิพจน์ (Expression) เช่น

$$\exists x : \mathbb{N} \bullet x > 5$$

เป็นการระบุว่ามี 'บางค่าของ x ซึ่งมีชนิดเป็นจำนวนเต็มที่มีค่ามากกว่า 5 ' ทั้งนี้จากการเพิ่มตัวบ่งปริมาณเข้าไปทำให้สามารถหาค่าความจริงของนิพจน์ได้

6.3) เซต (Set)

เซตคือกลุ่มของข้อมูลใด ๆ ซึ่งมีชนิดเดียวกันโดยที่สมาชิกแต่ละตัวในเซตต้องมีค่าแตกต่างกัน ทั้งนี้ลำดับของสมาชิกไม่มีความสำคัญ เช่น เซตของจำนวนเต็ม ก็คือกลุ่มของข้อมูลที่มีสมาชิกเป็นเลขจำนวนเต็ม เช่น {1,2,3} และ {3,2,1} ทั้ง 2 เซตถือว่าเป็นเซตเดียวกันเนื่องจากลำดับของสมาชิกไม่มีความสำคัญ

6.4) ความสัมพันธ์ (Relation)

ความสัมพันธ์ คือ ความเกี่ยวข้องกันระหว่างเซต 2 เซตใด ๆ โดยเซตต้นทาง (Source set) เรียกว่า โดเมน (Domain) และเซตเป้าหมายเรียกว่าพิสัย (Range) เช่น

กำหนดให้

A= {1,2,3,4} = B

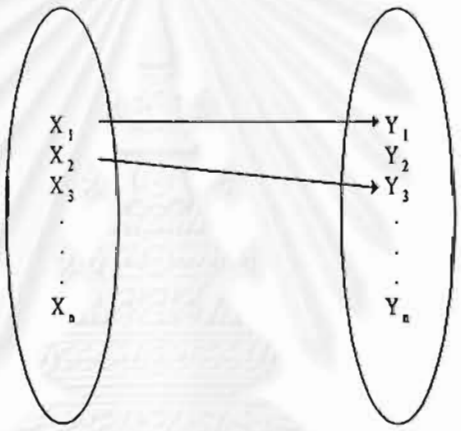
นิยามความสัมพันธ์ R สำหรับ A ว่า

$a \, R \, b$ เมื่อ $a < b$

ดังนั้น R = {(1,2),(1,3),(1,4),(2,3),(2,4),(3,4)}

6.5) ฟังก์ชัน (Function)

ฟังก์ชัน เป็นชนิดหนึ่งของความสัมพันธ์ โดยที่โดเมนหนึ่งตัวสามารถมีพิสัยได้เพียงหนึ่งตัวเท่านั้น บางครั้ง ฟังก์ชันอาจเรียกได้ว่าเป็นการแมพปิง (Mapping) หรือ การเปลี่ยนรูป (Transformation) ดังรูป 2.7 เพราะเสมือนกับการกำหนดเป็นกฎในการเปลี่ยนค่า คือ เมื่อให้ค่าเข้าไปแล้วค่าที่ให้เข้าไปนั้นสามารถทำการแมพปิงกับค่าที่อยู่ในโดเมนได้ จะให้ผลลัพธ์กลับออกมาค่าหนึ่งเสมอ เช่น ให้ f เป็นฟังก์ชันจาก A ไป B โดยที่ $a \in dom(f)$ แล้ว f (a) จะเท่ากับสมาชิกตัวหนึ่งของ B



รูปที่ 2.7 รูปแสดงการแมพปิงของฟังก์ชัน

ฟังก์ชันสามารถแบ่งได้เป็น

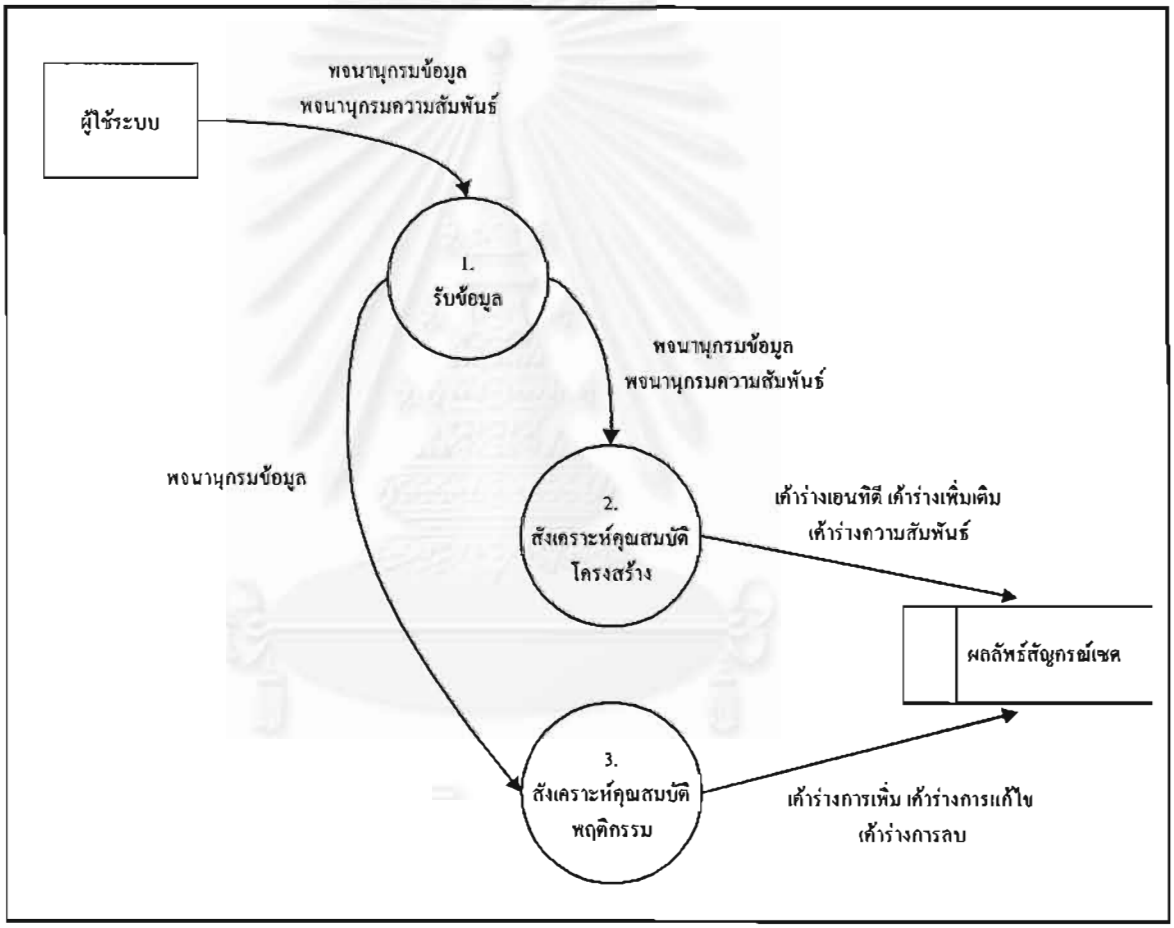
- ฟังก์ชันบางส่วน (Partial function) คือ ฟังก์ชันที่โดเมนของฟังก์ชันเป็นเซตย่อย (Subset) ของเซตต้นทาง
- ฟังก์ชันทั้งหมด (Total function) คือฟังก์ชันที่โดเมนของฟังก์ชันเท่ากับเซตต้นทาง
- ฟังก์ชันหนึ่งต่อหนึ่ง (Injection) คือฟังก์ชันที่โดเมนตัวหนึ่งจะมีพิสัยได้หนึ่งพิสัย และพิสัยตัวหนึ่งจะมีโดเมนได้หนึ่งโดเมน
- ฟังก์ชันทั่วถึง (Surjection) คือฟังก์ชันที่พิสัยของฟังก์ชันเท่ากับเซตเป้าหมาย (Target set)
- ฟังก์ชันหนึ่งต่อหนึ่งทั่วถึง (Bijection) คือฟังก์ชันที่เป็นทั้งฟังก์ชันหนึ่งต่อหนึ่ง และฟังก์ชันทั่วถึง

6.6) ลำดับ (Sequence)

ลำดับ คือการที่ข้อมูลเรียงต่อกันไป โดยข้อมูลเหล่านั้นจะต้องมีชนิดเดียวกัน และลำดับของข้อมูลมีความสำคัญ เช่น <u,p,c> และ <p,u,c> ไม่ถือว่าเป็นข้อมูลเดียวกัน เพราะมีลำดับของข้อมูลแตกต่างกัน เป็นต้น

บทที่ 3
ขั้นตอนวิธีแปลงแบบจำลองเอนทิตี และความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เชด

ในบทนี้ จะกล่าวถึงขั้นตอนและวิธีการแปลงข้อมูลของแบบจำลองเอนทิตีและความสัมพันธ์ โดยกำหนดรูปแบบข้อมูลนำเข้าเป็นพจนานุกรมข้อมูลซึ่งเป็นส่วนที่แสดงถึงโครงสร้างของแต่ละเอนทิตี ว่าประกอบไปด้วยลักษณะประจำใดบ้าง รวมทั้งดีย์ของเอนทิตี และพจนานุกรมความสัมพันธ์ซึ่งเป็นส่วนที่แสดงถึงความสัมพันธ์ของเอนทิตีใด ๆ และลักษณะของความสัมพันธ์ของเอนทิตีเหล่านั้นให้เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เชด โดยได้แบ่งขั้นตอนในการทำงานออกเป็นส่วน ๆ ดังรูปที่ 3.1



รูปที่ 3.1 แผนภาพแสดงขั้นตอนการทำงานของโปรแกรม

จากรูปที่ 3.1 สามารถแบ่งได้เป็น 3 ส่วน ได้แก่ ส่วนรับข้อมูล ส่วนสังเคราะห์คุณสมบัติโครงสร้าง (Synthesize structure property) และส่วนสังเคราะห์คุณสมบัติพฤติกรรม (Synthesize behavior property) โดยในที่นี้จะกล่าวถึงเฉพาะส่วนสังเคราะห์คุณสมบัติโครงสร้าง และส่วนสังเคราะห์คุณสมบัติพฤติกรรม โดยส่วนรับข้อมูลจะกล่าวถึงในบทที่ 4

จากพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ ที่ใช้ในการทำข้อกำหนดความต้องการทั่ว ๆไป ซึ่งรูปแบบของพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ที่ใช้ในงานวิจัยนี้จะกล่าวถึงในบทที่ 4 สามารถนำมานิยามเป็นข้อกำหนดรูปนัยในรูปของสัญกรณ์เซดได้เป็นโมดูล ๆ โดยในส่วนสังเคราะห์คุณสมบัติโครงสร้างเป็นส่วนที่นิยามลักษณะโครงสร้างของเอนทิตีว่าประกอบไปด้วยลักษณะประจำใดบ้าง และมีความสัมพันธ์ต่อกันแบบใด ซึ่งสามารถนิยามเค้าร่างได้ 3 เค้าร่างได้แก่ เค้าร่างเอนทิตี เพื่อแสดงถึงโครงสร้างข้อมูลของแต่ละเอนทิตี เค้าร่างเพิ่มเติม (Entity Extension Schema) เพื่อแสดงถึงคุณสมบัติเพิ่มเติมของเค้าร่างเอนทิตี และเค้าร่างความสัมพันธ์ เพื่อแสดงถึงความสัมพันธ์ของแต่ละเอนทิตีว่ามีความสัมพันธ์กันแบบใด และในส่วนสังเคราะห์คุณสมบัติพฤติกรรมเป็นการนิยามชุดคำสั่งพื้นฐานที่ใช้กับแต่ละเอนทิตี โดยสามารถนิยามเป็นเค้าร่างได้ 3 เค้าร่างได้แก่ เค้าร่างการเพิ่ม (Insert schema) เค้าร่างการแก้ไข (Update schema) และเค้าร่างการลบ (Delete schema) โดยในการสังเคราะห์คุณสมบัติพฤติกรรมได้คำนึงถึงบูรณภาพของการอ้างอิง (Referential integrity) ด้วย

3.1. ส่วนการสังเคราะห์คุณสมบัติโครงสร้าง

ส่วนการสังเคราะห์คุณสมบัติโครงสร้างเป็นการนำข้อมูลต่าง ๆ ในพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ มาสังเคราะห์คุณสมบัติโครงสร้าง โดยแบ่งเป็นการสังเคราะห์โครงสร้างของเอนทิตี และ การสังเคราะห์ความสัมพันธ์ระหว่างเอนทิตี

ในการสังเคราะห์โครงสร้างของเอนทิตี ผลของการสังเคราะห์จะอยู่ในรูปของเค้าร่างเอนทิตี และเค้าร่างเพิ่มเติม โดยในส่วนของเค้าร่างเอนทิตี จะแสดงถึงส่วนประกอบของแต่ละเอนทิตีว่าประกอบไปด้วยลักษณะประจำใดบ้าง แต่ละลักษณะประจำมีชนิดข้อมูลเป็นอะไร และในส่วนของเค้าร่างเพิ่มเติม จะเป็นส่วนประกาศ คุณสมบัติเพิ่มเติมสำหรับแต่ละเค้าร่างเอนทิตีนั้น เพื่อให้เกิดความยืดหยุ่นในการใช้ข้อมูล เนื่องจากการประกาศเค้าร่างเอนทิตีจะเสมือนการประกาศชนิดข้อมูลใหม่ซึ่งมีลักษณะเป็นระเบียน และการประกาศเค้าร่างเพิ่มเติมก็เป็นการนำชนิดข้อมูลนั้นมาใช้

ในการสังเคราะห์ความสัมพันธ์ระหว่างเอนทิตี ผลของการสังเคราะห์จะอยู่ในรูปของเค้าร่างความสัมพันธ์ ซึ่งจะเป็นการแสดงว่าเอนทิตีใด มีความสัมพันธ์คาร์ดินัลริตีแบบใด เช่น หนึ่งต่อหนึ่ง หนึ่งต่อหลาย หรือ หลายต่อหนึ่ง เป็นต้น

นอกจากนี้ ผู้วิจัยได้กำหนดชนิดของข้อมูลพื้นฐานที่จำเป็นเพิ่มเติมในกระบวนการสังเคราะเค้าร่างเอนทิตีด้วยเนื่องจากในสัญกรณ์เซดมีชนิดข้อมูลเพียงชนิดเดียว คือ ชนิดจำนวนเต็ม โดยขั้นตอนวิธีการทำงานในส่วนนี้จะแบ่งการนิยามได้เป็น 4 ส่วน คือ ส่วนนิยามชนิดของข้อมูล ส่วนการนิยามเค้าร่างเอนทิตี ส่วนนิยามเค้าร่างเพิ่มเติม และส่วนนิยามเค้าร่างความสัมพันธ์

3.1.1) ส่วนนิยามชนิดของข้อมูล

เนื่องจากในสัญกรณ์เซดมีชนิดของข้อมูลพื้นฐานอยู่เพียงชนิดเดียวคือ ชนิดจำนวนเต็ม ดังนั้น ถ้าหากต้องการใช้ชนิดของข้อมูลเป็นชนิดอื่นก็จำเป็นจะต้องทำการนิยามชนิดของข้อมูลนั้น ๆ ขึ้นมาใหม่ ซึ่งเซด

ก็ได้คำนึงถึงจุดนี้จึงให้ผู้ใช้สามารถนิยามชนิดของข้อมูลใด ๆ ขึ้นมาก็ได้ตามต้องการโดยใช้ก็เวนเซต อย่างไรก็ดี การนิยามเป็นก็เวนเซตนั้น ในตัวสัญกรณ์เซด ไม่สามารถรู้ได้ว่าข้อมูลที่เป็นไปได้สำหรับชนิดข้อมูลนั้น ๆ คืออะไร และ คำสั่งใดที่สามารถใช้ได้ ซึ่งจะต้องทำการนิยามใหม่ด้วยเช่นกัน

ในภาษาโปรแกรมทั่ว ๆ ไป ชนิดของข้อมูลที่นิยมใช้กัน ได้แก่ จำนวนเต็ม ตัวอักษร สายอักษร (String) และบูลีน (Boolean) ซึ่งในสัญกรณ์เซดมีชนิดข้อมูลชนิดจำนวนเต็มอยู่แล้ว ดังนั้นในงานวิจัยนี้จะทำการนิยามชนิดของข้อมูล ตัวอักษร สายอักษร และบูลีน เพิ่มเติมเพื่อให้เหมาะสมกับการใช้งานทั่ว ๆ ไป และเพื่อสามารถใช้การพิสูจน์ข้อกำหนดด้วย Z/EVES ได้ เนื่องจากข้อมูลพื้นฐานชนิดเดียวของ สัญกรณ์เซด คือ จำนวนเต็ม ดังนั้น การนิยามชนิดข้อมูลอื่นเพิ่มเติม จะเป็นการนิยามโดยตั้งอยู่บนพื้นฐานจำนวนเต็ม เพื่อให้สามารถใช้คำสั่งต่าง ๆ ที่มีอยู่แล้วของชนิดข้อมูลจำนวนเต็มได้ ซึ่งจะมีการนิยามชนิดของข้อมูลดังนี้

1) การนิยามชนิดข้อมูลตัวอักษร สามารถกำหนดได้โดย

$$Char == 0..127$$

จากการนิยาม แสดงถึง Char เป็นตัวย่อของ 0-127 หมายความว่า Char เป็นชนิดข้อมูลซึ่งมีค่าที่เป็นไปได้คือ 0 ถึง 127 ซึ่งในความหมายของ Char ที่ใช้กันทั่ว ๆ ไป จะหมายถึงตัวอักษร A-Z และ a-z ถ้าหากทำการนิยามตามข้างต้น สัญกรณ์เซดจะรู้แค่ว่าค่าที่เป็นไปได้สำหรับข้อมูลชนิด Char คือ 0 ถึง 127 ดังนั้นจึงทำการนิยามตัวอักษรเพิ่มเติม เพื่อให้สามารถใช้ได้ใกล้เคียงกับข้อมูลชนิด Char ทั่ว ๆ ไป ซึ่งจะนิยามตัวอักษรแต่ละตัวให้เท่ากับตัวเลขตามรหัสแอสกี (ASCII) โดย

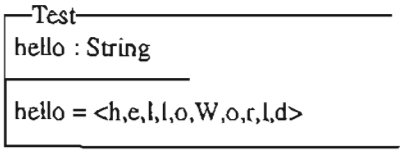
$$A == 65$$
$$B == 66$$
$$\vdots$$
$$z == 122$$

จากการนิยามดังนี้ จะทำให้ได้ข้อมูลชนิด Char ที่ใกล้เคียงกัน แต่ด้วยข้อจำกัดของสัญกรณ์เซดว่าถ้าหากมีการนิยามเป็นตัวย่อ หรือชนิดข้อมูลฟรีไทปไว้แล้ว จะไม่สามารถนิยามตัวย่อ หรือชนิดข้อมูลฟรีไทป นั้นเป็นตัวแปรได้อีก ดังนั้น ภายในข้อกำหนดจะไม่สามารถมีตัวแปรซึ่งมีชื่อเดียวกับตัวที่เราได้กำหนดเป็นตัวย่อไว้ เช่น ในที่นี้ไม่สามารถมีตัวแปรชื่อ A หรือ B เป็นต้น

2) การนิยามชนิดข้อมูลสายอักษร สามารถกำหนดได้โดย

$$String == seq\ Char$$

จากการนิยาม เป็นการกำหนดให้ String เป็น ลำดับ (Sequence) ของข้อมูลชนิด Char ลักษณะของข้อมูลแบบลำดับในสัญกรณ์เซตจะอยู่ในรูป <w,h,a,t> ซึ่งถ้าหากเรากำหนดให้ hello เป็นชนิด String แล้วต้องการให้ hello มีค่าเท่ากับ helloWorld จะต้องเขียนดังนี้



3) การนิยามชนิดข้อมูลบูลีน สามารถกำหนดได้โดย

Boolean ::= True | False

จากการนิยามเป็นการนิยาม Boolean เป็นชนิดข้อมูลฟรีไทปซึ่งมีค่าที่เป็นไปได้ คือ True และ False ซึ่งด้วยข้อจำกัดของสัญกรณ์เซต ว่าถ้าหากมีการนิยามเป็นตัวย่อ หรือ ชนิดข้อมูลฟรีไทปไว้แล้ว จะไม่สามารถนิยามตัวย่อ หรือ ชนิดข้อมูลฟรีไทปนั้นเป็นตัวแปรได้อีก ดังนั้น ภายในข้อกำหนด จะไม่สามารถมีตัวแปรซึ่งมีชื่อเดียวกับตัวที่เราได้กำหนดเป็นค่าในชนิดข้อมูลฟรีไทปไว้ เช่น ในที่นี้จะไม่สามารถมีตัวแปรชื่อ True และ False

ในการนิยามชนิดของข้อมูลเป็นชนิดอื่น ๆ นอกเหนือจากนี้จะนิยามให้เป็นกีเวนเซต ซึ่งใช้สำหรับการนิยามชนิดข้อมูลนามธรรมใด ๆ ซึ่งจะทำให้ไม่สามารถรู้ได้ถึงค่าที่เป็นไปได้ของชนิดข้อมูลนั้น ๆ แต่ในการสร้างสัญกรณ์เซตนี้ เป็นเพียงการสร้างข้อกำหนดที่ใช้สำหรับการสร้างระบบเท่านั้น จึงอาจยังไม่มีการลงรายละเอียดเกี่ยวกับชนิดของข้อมูลมากนัก ดังนั้นการกำหนดชนิดของข้อมูลจึงอาจกำหนดเป็นชนิดข้อมูลนามธรรม ซึ่งมีความหมายอยู่ในตัวเองอยู่แล้ว

3.1.2) ส่วนนิยามเค้าร่างเอนทิตี

ส่วนนิยามเค้าร่างเอนทิตีเป็นส่วนที่นิยามถึงโครงสร้างเอนทิตี คือการกำหนดว่าเอนทิตีประกอบด้วยลักษณะประจำอะไรบ้าง มีชนิดเป็นอะไร รวมถึงเงื่อนไขต่าง ๆ ของลักษณะประจำโดยอาศัยข้อมูลจากพจนานุกรมข้อมูล ซึ่งมีขั้นตอนการนิยามดังต่อไปนี้

<u>ขั้นตอนวิธีที่ 1</u> สร้างเค้าร่างเอนทิตี

ให้ EN แทนชื่อเอนทิตี

AN แทนลักษณะประจำ

AT แทนชนิดของลักษณะประจำ

CS แทนเงื่อนไขของลักษณะประจำ

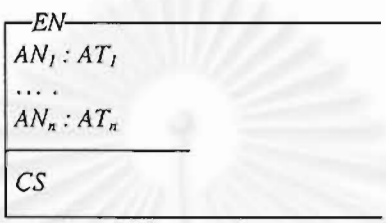
ในการสร้างเค้าร่างเอนทิตีจะมีขั้นตอนในการสร้างดังนี้

ก) ใช้ชื่อของเอนทิตี EN เป็นชื่อของเค้าร่างเอนทิตี

ข) นำลักษณะประจำ AN ของเอนทิตีมาใส่ในส่วนการประกาศ พร้อมทั้งกำหนดชนิดของลักษณะประจำ AT

ค) ลักษณะประจำ AN ที่เป็นคีย์หลัก และคีย์นอก ก็สามารถนิยามได้เช่นเดียวกับลักษณะประจำทั่ว ๆ ไป

ง) ถ้าหากมีภาคแสดง ที่แสดงถึงเงื่อนไขของลักษณะประจำ CS ให้นำมาใส่ไว้ในส่วนสัจพจน์



รูปที่ 3.2 รูปแบบเค้าร่างเอนทิตี

จากรูปที่ 3.2 แสดงให้เห็นว่าเอนทิตีประกอบไปด้วยลักษณะประจำ AN_1 ถึง AN_n ซึ่งมีชนิดเป็น AT_1 ถึง AT_n ตามลำดับ นอกจากนั้น ยังบ่งบอกถึงเงื่อนไข CS สำหรับลักษณะประจำด้วย

3.1.3) ส่วนนิยามเค้าร่างเพิ่มเติม

ส่วนนิยามเค้าร่างเพิ่มเติมเป็นการนิยามข้อกำหนดต่าง ๆ ที่ไม่สามารถนิยามไว้ภายในเค้าร่างเอนทิตีได้ เนื่องจากข้อกำหนดเหล่านี้มีการอ้างถึงตัวเค้าร่างเอนทิตีของตัวเอง เช่น ข้อกำหนดคีย์หลัก และคีย์นอก เป็นต้น ซึ่งมีขั้นตอนการนิยามดังต่อไปนี้

<u>ขั้นตอนวิธีที่ 2</u> สร้างเค้าร่างเพิ่มเติม

ให้ EN แทนเค้าร่างเอนทิตี

enSet แทนเซตจำกัดของเค้าร่างเอนทิตี EN

key แทนคีย์หลักของเอนทิตี EN

P แทนเค้าร่างเอนทิตีพ่อ

FK แทนคีย์นอก

PK แทนคีย์หลัก

e_1,e_2 แทนตัวแปรซึ่งมีชนิดเป็น EN

enVar แทนตัวแปรซึ่งมีชนิดเป็น EN

pVar แทนตัวแปรซึ่งมีชนิเป็น P

ในการสร้างเค้าร่างเพิ่มเติมจะมีขั้นตอนในการสร้างดังนี้

ก) นำชื่อของเอนทิตี EN รวมกับ Ext (ตัวย่อของ Extension) เป็นชื่อของเค้าร่าง

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ข) นิยามตัวแปร enSet เป็นเซตจำกัดของเอนทิตี EN

ค) สำหรับการนิยามคีย์หลัก ให้ใส่ภาคแสดงตามรูปแบบต่อไปนี้ในส่วนของสัจพจน์

$$\forall e_1, e_2 : EN \mid e_1 \in enSet \land e_2 \in enSet \land e_1 \neq e_2 \bullet e_1.key \neq e_2.key$$

ง) สำหรับการนิยามคีย์นอก ให้ใส่ภาคแสดงตามรูปแบบต่อไปนี้ในส่วนของสัจพจน์

$$\forall enVar : EN \bullet (\exists pVar : P \bullet enVar.FK = pVar.PK)$$

ENEx

$enSet : \mathbb{F} EN$ ------(1)

$\forall e_1, e_2 : EN \mid e_1 \in enSet \land e_2 \in enSet \land e_1 \neq e_2 \bullet e_1.key \neq e_2.key$ ------(2)

$\forall enVar : EN \bullet (\exists pVar : P \bullet enVar.FK = pVar.PK)$ -----(3)

รูปที่ 3.3 รูปแบบเค้าร่างเพิ่มเติม

จากรูปที่ 3.3 (1) แสดงถึง การนิยาม enSet เป็นเซตจำกัดของ EN ซึ่งเปรียบเสมือนการนิยามเป็นตารางของข้อมูล EN ส่วน (2) เป็นการนิยามเพื่อแสดงถึงความเป็นคีย์หลัก โดยการเปรียบเทียบระเบียบคู่ใด ๆ ในตารางว่า "ถ้าระเบียนคู่ใด ๆ ไม่เท่ากันแล้วค่าข้อมูลที่เป็นคีย์หลักของทั้งสองระเบียนนี้ก็ต้องไม่เท่ากัน" และ (3) เป็นการนิยามถึงความเป็นคีย์นอก โดยนิยามว่า "ทุก ๆ ค่าของคีย์นอก จะต้องเท่ากับค่าใดค่าหนึ่งที่เป็นคีย์หลักในตารางที่เป็นเอนทิตีพ่อ"

3.1.4) ส่วนนิยามเค้าร่างความสัมพันธ์

ส่วนนิยามเค้าร่างความสัมพันธ์เป็นส่วนนิยามถึงความสัมพันธ์ต่าง ๆ ระหว่างเอนทิตี ว่ามีความสัมพันธ์ต่อเอนทิตีใดบ้าง และมีความสัมพันธ์ต่อกันแบบใด ในส่วนนี้ได้ทำการนิยามรูปแบบสำหรับความสัมพันธ์ต่าง ๆ ได้แก่ หลายต่อหนึ่ง หนึ่งต่อหลาย และ หนึ่งต่อหนึ่ง เป็นต้น โดยอาศัยข้อมูลจากพจนานุกรมความสัมพันธ์ ซึ่งมีขั้นตอนการนิยามความสัมพันธ์ระหว่าง 2 เอนทิตี ใด ๆ เป็นเค้าร่างความสัมพันธ์ดังต่อไปนี้

<u>ขั้นตอนวิธีที่ 3</u> สร้างเค้าร่างความสัมพันธ์

ให้ R แทนความสัมพันธ์ใด ๆ ระหว่าง 2 เอนทิตีใด ๆ

EN_1, EN_2 แทนเอนทิตีใด ๆ

Rel แทนลักษณะความสัมพันธ์

CS แทนเงื่อนไขของความสัมพันธ์

r_1,r_2 แทนตัวแปรซึ่งเป็นความสัมพันธ์ R

first, second เป็นโพรเจคชันฟังก์ชันในการตัดส่วนใดส่วนหนึ่งของคู่ลำดับ

ในการสร้างเค้าร่างความสัมพันธ์จะมีขั้นตอนในการสร้างดังนี้

ก) กำหนดความสัมพันธ์ R เป็นเซตจำกัดของผลคูณคาร์ทีเซียนระหว่าง 2 เอนทิตีใด ๆ

$$R : F\ (EN_1 \times EN_2)$$

ข) จากนิยามข้างบนเป็นการระบุว่า EN_1 และ EN_2 มีความสัมพันธ์ต่อกันแบบหลายต่อหลาย ดังนั้นจึงมีส่วนระบุความสัมพันธ์เพิ่มเติมสำหรับความสัมพันธ์แบบอื่น ๆ คือ

- หากความสัมพันธ์ระหว่าง 2 เอนทิตีเป็นแบบหนึ่งต่อหลายให้ระบุภาคแสดงดังต่อไปนี้ในส่วนสัจพจน์

$$\forall r_1, r_2 : R \bullet second(r_1) = second(r_2) \Rightarrow first(r_1) = first(r_2)$$

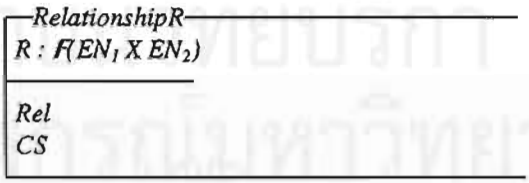
- หากความสัมพันธ์ระหว่าง 2 เอนทิตีเป็นแบบหลายต่อหนึ่ง ให้ระบุภาคแสดงดังต่อไปนี้ในส่วนสัจพจน์

$$\forall r_1, r_2 : R \bullet first(r_1) = first(r_2) \Rightarrow second(r_1) = second(r_2)$$

- หากความสัมพันธ์ระหว่าง 2 เอนทิตีเป็นแบบหนึ่งต่อหนึ่ง ให้ระบุภาคแสดงดังต่อไปนี้ในส่วนสัจพจน์

$$\forall r_1, r_2 : R \bullet second(r_1) = second(r_2) \Rightarrow first(r_1) = first(r_2) \wedge$$
$$first(r_1) = first(r_2) \Rightarrow second(r_1) = second(r_2)$$

ค) ถ้าหากมีภาคแสดงที่แสดงถึงเงื่อนไข CS ของความสัมพันธ์ให้นำมาใส่ไว้ในส่วนสัจพจน์



รูปที่ 3.4 รูปแบบเค้าร่างความสัมพันธ์

จากรูปที่ 3.4 เป็นการแสดงถึงความสัมพันธ์ R ซึ่งเป็นความสัมพันธ์ระหว่าง เอนทิตี EN_1 และเอนทิตี EN_2 โดยมีความสัมพันธ์กันแบบ Rel ซึ่งมีเงื่อนไขของความสัมพันธ์นี้คือ CS

3.2. ส่วนการสังเคราะห์คุณสมบัติพฤติกรรม

ส่วนการสังเคราะห์คุณสมบัติพฤติกรรมเป็นส่วนนิยามถึงชุดคำสั่งพื้นฐานที่สามารถสร้างได้จากพจนานุกรมข้อมูล โดยคำสั่งพื้นฐานนี้ประกอบไปด้วยคำสั่ง เพิ่ม ลบ และ แก้ไข ซึ่งคำสั่งเหล่านี้เป็นคำสั่งที่ใช้กับเอนทิตี โดยในการทำคำสั่งใดๆ กับเอนทิตีจะคำนึงถึงเงื่อนไขต่าง ๆ ที่ได้ระบุไว้ และบูรณภาพของการอ้างอิงด้วย เช่น ในการทำการเพิ่มระเบียน จะอนุญาตให้ทำการเพิ่มได้เมื่อข้อมูลนั้นไม่ผิดต่อเงื่อนไขต่างๆ ของลักษณะประจำ และคีย์หลักยังไม่เคยมีอยู่ เป็นต้น

3.2.1) การเพิ่มระเบียน

ส่วนการเพิ่มระเบียนเป็นการรับข้อมูลใหม่แล้วนำไปเพิ่มในฐานข้อมูล โดยจะคำนึงถึงบูรณภาพของการอ้างอิง และเงื่อนไขของลักษณะประจำด้วย มีขั้นตอนการนิยามดังต่อไปนี้

<u>ขั้นตอนวิธีที่ 4</u> สร้างเค้าร่างการเพิ่ม

ให้ EN แทนเค้าร่างเอนทิตี

ENExt แทนเค้าร่างเพิ่มเติมของเอนทิตี EN

newValue? แทนตัวแปรนำเข้าที่รับค่าที่ต้องการทำการเพิ่ม

enSet, enSet′ แทนเซตจำกัดของเอนทิตี EN

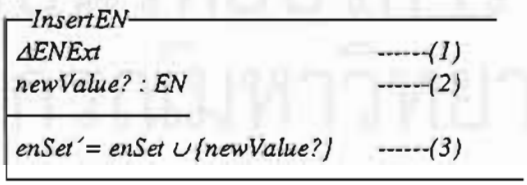
ในการสร้างเค้าร่างการเพิ่มจะมีขั้นตอนในการสร้างดังนี้

ก) นิยาม ΔENExt ในส่วนการประกาศ

ข) นิยาม newValue? เป็นตัวแปรนำเข้าซึ่งมีชนิดเป็น EN ในส่วนการประกาศ

ค) ใส่ภาคแสดงดังต่อไปนี้ในส่วนของสัจพจน์ ซึ่งภาคแสดงนี้แสดงถึงการเพิ่มระเบียนใหม่เข้าไปในเซตของข้อมูลเดิมที่มีอยู่แล้ว

$$enSet' = enSet \cup \{ newValue? \}$$



รูปที่ 3.5 รูปแบบเค้าร่างการเพิ่ม

จากรูปที่ 3.5 (1) เป็นการรวมเค้าร่างโดย Δ เป็นการบอกว่าข้อมูลที่รวมเข้ามานั้นอาจเกิดการเปลี่ยนแปลงค่าได้หลังจากที่ได้ทำตามคำสั่งแล้ว ส่วน (2) เป็นการนิยาม newValue? ให้เป็นตัวแปร

สำหรับรับค่าข้อมูลที่ต้องการจะเพิ่มเข้ามาจากภายนอก และ (3) เป็นการเพิ่มระเบียนใหม่เข้ากับเซตข้อมูลที่มีอยู่เดิม

3.2.2) การแก้ไขระเบียน

ส่วนการแก้ไขระเบียนเป็นการแก้ไขค่าของลักษณะประจำที่มีอยู่ โดยไม่สามารถแก้ไขค่าของลักษณะประจำที่เป็นคีย์หลักของเอนทิตีได้ เช่นเดียวกับการเพิ่มระเบียน การแก้ไขระเบียนจะต้องคำนึงถึงเงื่อนไขต่าง ๆ และบูรณภาพของการอ้างอิง คือ ไม่สามารถทำการแก้ไขข้อมูลในส่วนของคีย์ และค่าของข้อมูลที่ถูกแก้ไขแล้วต้องยังคงถูกต้องตามเงื่อนไขของลักษณะประจำ มีขั้นตอนการนิยามดังต่อไปนี้

ขั้นตอนวิธีที่ 5 สร้างเค้าร่างการแก้ไข

ให้ EN แทนเค้าร่างเอนทิตี

ENExt แทนเค้าร่างเพิ่มเติมของเอนทิตี EN

uEN? แทนตัวแปรนำเข้าที่รับค่าที่ต้องการทำการแก้ไข

enSet แทนเซตจำกัดของเอนทิตี EN

key แทนคีย์หลัก

AN แทนลักษณะประจำ

tEN, tEN′ แทนตัวแปร

ในการสร้างเค้าร่างการแก้ไขจะมีขั้นตอนในการสร้างดังนี้

ก) นิยามเค้าร่างเพิ่มเติม ENExt ในส่วนการประกาศ เพื่อเป็นการประกาศให้รู้จักส่วนต่าง ๆ ของ ENExt

ข) นิยาม uEN? เป็นตัวแปรสำหรับรับค่าที่ต้องการเปลี่ยนแปลงเป็น โดยมีชนิด EN ในส่วนการประกาศ

ค) นิยาม tEN และ tEN′ ซึ่งมีชนิดเป็น EN ในส่วนการประกาศ เพื่อเป็นการประกาศถึงสถานะก่อนและ สถานะหลังจากทำพฤติกรรม ของ tEN

ง) ใส่ภาคแสดงดังต่อไปนี้ในส่วนสัจพจน์

$$
\begin{aligned}
&tEN \in enSet\\
&tEN.key = uEN?.key\\
&tEN'.AN_1 = uEN?.AN_1\\
&\vdots\\
&tEN'.AN_n = uEN?.AN_n
\end{aligned}
$$

UpdateEN

$ENExt$	------(1)
$uEN? : EN$	------(2)
$tEN, tEN' : EN$	------(3)
$tEN \in enSet$	------(4)
$tEN.key = uEN?.key$	------(5)
$tEN'.AN_1 = uEN?.AN_1$	------(6)
:	
$tEN'.AN_n = uEN?.AN_n$	------(7)

รูปที่ 3.6 รูปแบบเค้าร่างการแก้ไข

จากรูปที่ 3.6 (1) เป็นการรวมเค้าร่าง คือเป็นการรวมคุณสมบัติต่าง ๆ ที่ระบุในเค้าร่างเพิ่มเติม ENExt เข้าไว้กับเค้าร่างการแก้ไขนี้ (2) เป็นการกำหนดตัวแปรชื่อ uEN? เพื่อใช้เป็นตัวแปรในการรับค่าของข้อมูลที่ต้องการเปลี่ยน (3) เป็นการประกาศตัวแปรเพิ่มอีก 2 ตัวคือ tEN ซึ่งจะเก็บค่าก่อนที่จะมีการกระทำใด ๆ และ tEN′ ซึ่งจะเก็บค่าหลังจากที่เกิดการกระทำใด ๆ แล้ว (4) และ (5) เป็นการหาระเบียนที่ต้องการแก้ไข (6) และ (7) เป็นการแก้ไขข้อมูลในระเบียนนั้น โดยการแทนที่ด้วยค่าข้อมูลเดิมด้วยค่าข้อมูลใหม่ที่ได้รับมา

3.2.3) การลบระเบียน

ส่วนการลบระเบียนเป็นลบข้อมูลออกจากเซตของข้อมูลทั้งหมด ทั้งนี้ต้องมีการพิจารณาถึงบูรณภาพของการอ้างอิง เช่นการลบข้อมูลที่ถูกข้อมูลอื่นอ้างอิงอยู่ โดยในที่นี้ได้แก้ปัญหานี้โดยการใช้วิธีลบแบบลดหลั่น มีขั้นตอนการนิยามดังนี้

<u>ขั้นตอนวิธีที่ 6</u> สร้างเค้าร่างการลบ

ให้ EN แทนเค้าร่างสถานะ

ENExt แทนเค้าร่างเพิ่มเติมของเอนทิตี EN

enSet, enSet′ แทนเซตจำกัดของเอนทิตี

dEN? แทนข้อมูลที่ต้องการจะลบ โดยมีชนิดเป็น EN

C แทนเค้าร่างสถานะของเอนทิตีลูก

DelC แทนเค้าร่างการลบของเอนทิตีลูก C

dC? แทนข้อมูลที่ต้องการจะลบ ของเค้าร่าง DelC โดยมีชนิดเป็น C

FK แทนคีย์นอก

PK แทนคีย์หลัก

ในการสร้างเค้าร่างการลบจะมีขั้นตอนในการสร้างดังนี้

ก) ใช้ Del ตามด้วย EN เป็นชื่อเค้าร่าง โดยที่ EN คือชื่อของเค้าร่างสถานะที่ต้องการจะลบ

ข) นิยาม dEN? ในส่วนการประกาศ เพื่อรับค่าข้อมูลที่ต้องการจะลบ

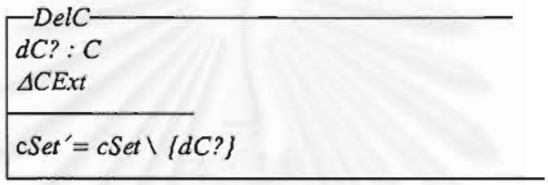
ค) นิยาม ΔENExt ในส่วนการประกาศ

ง) ในกรณีที่มีเอนทิตีอื่นซึ่งขึ้นกับเอนทิตีที่จะลบ dEN? ให้ใส่ DelC ในส่วนการประกาศด้วย เพื่อเป็นการเรียกคำสั่งการลบของเอนทิตีลูกให้ทำงาน โดยที่ C เป็นชื่อของเอนทีตีลูกที่มีการอ้างอิงถึงข้อมูลที่จะลบ และใส่ภาคแสดงดังต่อไปนี้ในส่วนสัจพจน์

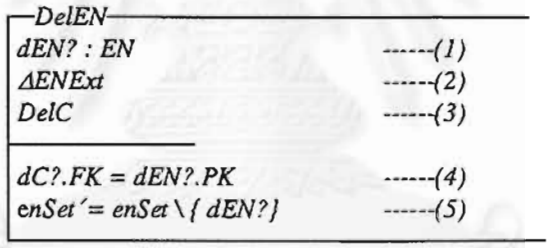
dC?.FK = dEN?.PK

จ) ใส่ภาคแสดงดังต่อไปนี้ในส่วนสัจพจน์

enSet′ = enSet \ {dEN?}



รูปที่ 3.7 รูปแบบเค้าร่างการลบ (ส่วนเอนทิตีลูก)



รูปที่ 3.8 รูปแบบเค้าร่างการลบ (ส่วนเอนทิตีพ่อ)

จากรูปที่ 3.7 DelC เป็นเค้าร่างการลบของเอนทิตีลูก C และ จากรูปที่ 3.8 (1) เป็นการกำหนดตัวแปร dEN? เพื่อใช้ในการรับค่าข้อมูลที่ต้องการจะลบ (2) เป็นการรวมเค้าร่างโดย Δ เป็นการบอกว่าข้อมูลที่รวมเข้ามานั้นอาจเกิดการเลี่ยนแปลงค่าได้หลังจากที่ได้ทำตามคำสั่งแล้ว เนื่องจากในการลบข้อมูลจะต้องคำนึงถึงบูรณะภาพการอ้างอิง ดังนั้นส่วน (3) จึงเป็นการรวมเค้าร่างที่ใช้ในการลบเอนทิตีที่เป็นเอนทิตีลูกเพื่อลบข้อมูลในเอนทิตีลูกที่อ้างอิงถึงข้อมูลที่ข้อมูลในเอนทิตีพ่อที่จะทำการลบนี้ (4) เป็นการส่งค่าที่จะทำการลบของเอนทิตีพ่อไปให้กับตัวแปรของเอนทิตีลูกเพื่อให้คำสั่งในการลบของเอนทิตีลูกสามารถทำงานได้ และ (5) เป็นการลบข้อมูลออกจากเซตของข้อมูลเดิม ทั้งนี้ถ้าหากว่า เอนทิตีที่จะทำการลบไม่มีเอนทิตีลูกก็จะไม่มีส่วน (3) และ (4)

บทที่ 4
การพัฒนาเครื่องมือซอฟต์แวร์แปลงแบบจำลองเอนทิตีเป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เชด

ในการทำงานของระบบสามารถแบ่งได้เป็น 3 ส่วน ได้แก่ ส่วนรับข้อมูล ส่วนสังเคราะห์ สัญกรณ์เชด และส่วนบันทึกข้อมูล

4.1. ส่วนรับข้อมูล

ส่วนรับข้อมูลเป็นส่วนเริ่มต้นการทำงานโดย การรับชื่อของแฟ้มข้อมูลซึ่งจะใช้ในการทำงานโดยลักษณะของข้อมูลนำเข้าจะอยู่ในรูปของแฟ้มข้อความ (Text file) สามารถแยกได้เป็น 2 แฟ้ม คือ แฟ้มข้อความสำหรับพจนานุกรมข้อมูล และแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์

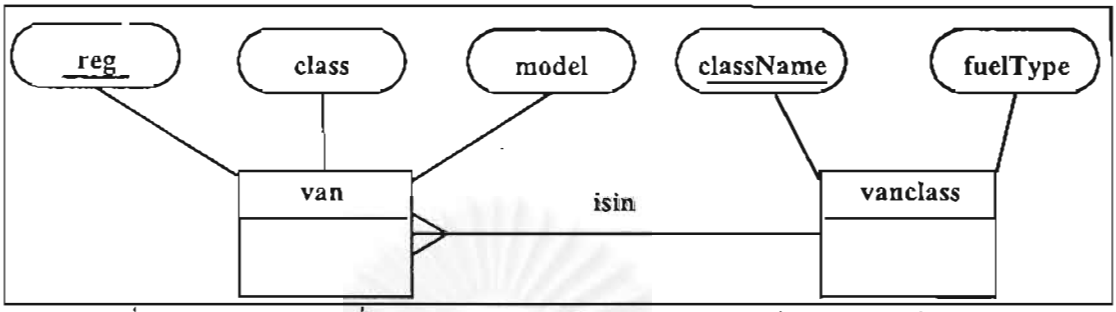
1) รูปแบบแฟ้มข้อความสำหรับพจนานุกรมข้อมูลจะมีลักษณะคือ ในแต่ละบรรทัดจะแทนข้อมูล 1 ระเบียน (Record) โดยแต่ละระเบียนประกอบไปด้วยข้อมูล 5 เขตข้อมูล (Field) แสดงในตารางที่ 4.1

ตารางที่ 4.1 แสดงรูปแบบแฟ้มข้อความสำหรับพจนานุกรมข้อมูล

ชื่อเอนทิตี	เป็นการระบุชื่อเอนทิตีของลักษณะประจำนั้น
ชื่อลักษณะประจำ	เป็นการระบุชื่อลักษณะประจำ
ชนิดลักษณะประจำ	เป็นการระบุถึงชนิดข้อมูลของลักษณะประจำ โดยในที่นี้มีชนิดของข้อมูลพื้นฐานที่ให้ใช้คือ ชนิดจำนวนเต็ม ซึ่งจะต้องใส่ชนิดเป็น Int ชนิดข้อมูลตัวอักษรต้องใส่ชนิดเป็น Char ชนิดข้อมูลสายอักษรต้องใส่ชนิดเป็น String และชนิดข้อมูลบูลีนต้องใส่ชนิดเป็น Boolean ส่วนชนิดของข้อมูลชนิดอื่นก็สามารถนิยามได้ โดยใส่ชื่อชนิดของข้อมูลที่ต้องการ
คีย์	เป็นการระบุถึงความเป็นคีย์ของลักษณะประจำ ไม่ว่าจะเป็นคีย์หลัก หรือ คีย์นอก โดยถ้าหากเป็นคีย์หลัก จะต้องระบุค่า PK ในเขตข้อมูลนี้ และถ้าหากเป็นคีย์นอก จะอยู่ในรูป FK(*ENT-ATTR*) โดย *ENT* เป็นเอนทิตีที่อ้างอิงถึง และ *ATTR* เป็น ลักษณะประจำภายในเอนทิตี *ENT* ที่อ้างอิงถึง และถ้าหากลักษณะประจำนั้นเป็นทั้งคีย์หลัก และคีย์นอก จะต้องระบุค่าเป็นรูป PK,FK(*ENT-ATTR*)
เงื่อนไข	เงื่อนไขอื่น ๆ ที่เกี่ยวข้องกับลักษณะประจำ โดยอยู่ในรูปแทคของลาเท็กซ์ของ Z/EVES

ข้อมูลในแต่ละเขตข้อมูลจะอยู่ในเครื่องหมายอัญประกาศ เพื่อเป็นการแบ่งแยกข้อมูลในแต่ละเขตข้อมูล ทั้งนี้ถ้าหากข้อมูลที่อยู่ภายในเครื่องหมายอัญประกาศประกอบไปด้วยช่องว่าง ช่องว่างนั้นจะถือว่าไม่

มีความหมายโดยจะทำการตัดทิ้ง ทั้งนี้แฟ้มข้อความนี้จะต้องมีนามสกุลเป็น .DAZ จากรูปที่ 4.1 ซึ่งแสดงส่วนหนึ่งของแผนภาพเอนทิตีและความสัมพันธ์ของระบบให้เช่ารถแวนสามารถแสดงตัวอย่างแฟ้มข้อความสำหรับพจนานุกรมข้อมูลได้ดังรูปที่ 4.2



รูปที่ 4.1 รูปแสดงส่วนหนึ่งของแผนภาพเอนทิตีและความสัมพันธ์ของระบบให้เช่ารถแวน

ชื่อเอนทิตี	ชื่อลักษณะประจำ	ชนิดลักษณะประจำ	คีย์	เงื่อนไข
"van"	"reg"	"int"	"PK"	""
"van"	"model"	"string"	""	""
"van"	"class"	"string"	"FK(vanclass-classname)"	""
"vanclass"	"classname"	"string"	"PK"	""
"vanclass"	"fueltype"	"Fueltype"	""	""

รูปที่ 4.2 ตัวอย่างแฟ้มข้อความสำหรับพจนานุกรมข้อมูล

2) รูปแบบแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ จะมีลักษณะคือ ในแต่ละบรรทัดจะแทนข้อมูล 1 ระเบียน โดยแต่ละระเบียนประกอบไปด้วยข้อมูล 5 เขตข้อมูล แสดงในตารางที่ 4.2

ตารางที่ 4.2 แสดงรูปแบบแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์

ชื่อความสัมพันธ์	ชื่อของความสัมพันธ์ที่เกิดระหว่าง 2 เอนทิตีใด ๆ
เอนทิตี1	ชื่อเอนทิตีตัวที่ 1 ที่เกี่ยวข้องในความสัมพันธ์
เอนทิตี2	ชื่อเอนทิตีตัวที่ 2 ที่เกี่ยวข้องในความสัมพันธ์
ลักษณะความสัมพันธ์	ลักษณะของความสัมพันธ์ระหว่าง 2 เอนทิตี ซึ่งในที่นี้ มีความสัมพันธ์ที่เป็นไปได้ 3 ลักษณะ คือ m-1 1-m และ 1-1
เงื่อนไข	เงื่อนไขอื่น ๆ ที่เกี่ยวข้องกับความสัมพันธ์ โดยอยู่ในรูปแทคของลาเท็กซ์ของ Z/EVES

ข้อมูลในแต่ละเขตข้อมูลจะอยู่ในเครื่องหมายอัญประกาศ เพื่อเป็นการแบ่งแยกข้อมูลในแต่ละเขตข้อมูล ทั้งนี้ถ้าหากข้อมูลที่อยู่ภายในเครื่องหมายอัญประกาศประกอบไปด้วยช่องว่าง ช่องว่างนั้นจะถือว่าไม่

มีความหมายโดยจะทำการตัดทิ้ง ทั้งนี้แฟ้มข้อความนี้จะต้องมีนามสกุลเป็น .REZ จากรูปที่ 4.1 ซึ่งแสดงส่วนหนึ่งของแผนภาพเอนทิตีและความสัมพันธ์ของระบบให้เช่ารถแวนสามารถแสดงตัวอย่างแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ได้ดังรูปที่ 4.4

ชื่อความสัมพันธ์	เอนทิตี1	เอนทิตี2	ลักษณะความสัมพันธ์	เงื่อนไข
"isin"	"van"	"vanclass"	"m-1"	""

รูปที่ 4.3 ตัวอย่างแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์

การสร้างสัญกรณ์เซตจะสร้างตามลำดับข้อมูลที่ได้รับ กล่าวคือ เอนทิตีใดที่คาดว่าจะถูกอ้างถึงโดยเอนทิตีอื่น จะต้องถูกสร้างขึ้นก่อน ดังนั้นข้อมูลของเอนทิตีนั้นจะต้องอยู่ในลำดับก่อนเอนทิตีที่ทำการอ้างถึง

4.2. ส่วนสังเคราะห์สัญกรณ์เซต

ส่วนสังเคราะห์สัญกรณ์เซตเป็นส่วนที่นำข้อมูลที่ได้จากพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ มาแปลงให้อยู่ในรูปของสัญกรณ์เซต ดังแสดงในรูปที่ 4.4 ซึ่งมีขั้นตอนการแปลงดังต่อไปนี้

1) รับชื่อแฟ้มข้อมูลพจนานุกรมข้อมูล พจนานุกรมความสัมพันธ์ และ แฟ้มที่จะเก็บผลลัพธ์
2) อ่านข้อมูลจากแฟ้มข้อความสำหรับพจนานุกรมข้อมูล และแฟ้มข้อความสำหรับพจนานุกรมข้อมูล
3) นิยามชนิดข้อมูลพื้นฐาน และกีเวนเซต และเก็บผลลัพธ์ลงแฟ้มข้อมูลผลลัพธ์
4) นิยามส่วนเค้าร่างเอนทิตี โดยนิยามตามขั้นตอนวิธีที่ 1 และเก็บผลลัพธ์ลงแฟ้มข้อมูลผลลัพธ์
5) นิยามส่วนเค้าร่างเพิ่มเติม โดยนิยามตามขั้นตอนวิธีที่ 2 และเก็บผลลัพธ์ลงแฟ้มข้อมูลผลลัพธ์
6) ถ้าต้องการเค้าร่างความสัมพันธ์ด้วย จะนิยามเค้าร่างความสัมพันธ์ โดยนิยามตามขั้นตอนวิธีที่ 3 และเก็บผลลัพธ์ลงแฟ้มข้อมูลผลลัพธ์
7) ทำการย้อนลำดับของข้อมูลเพื่อทำการสร้างเค้าร่างคำสั่งต่าง ๆ เนื่องจากในการลบจะมีการเรียกใช้เค้าร่างการลบของเอนทิตีลูกดังนั้นจึงต้องสร้างเค้าร่างการลบของเอนทิตีลูกก่อน
8) นิยามส่วนเค้าร่างการเพิ่ม โดยนิยามตามขั้นตอนวิธีที่ 4 และเก็บผลลัพธ์ลงแฟ้มข้อมูลผลลัพธ์
9) นิยามส่วนเค้าร่างการแก้ไข โดยนิยามตามขั้นตอนวิธีที่ 5 และเก็บผลลัพธ์ลงแฟ้มข้อมูลผลลัพธ์
10) นิยามส่วนเค้าร่างการลบ โดยนิยามตามขั้นตอนวิธีที่ 6 และเก็บผลลัพธ์ลงแฟ้มข้อมูลผลลัพธ์

4.3. ส่วนบันทึกข้อมูล

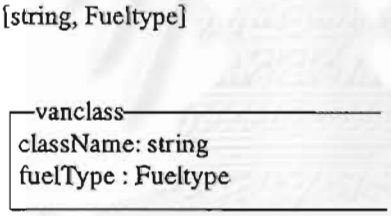
ส่วนบันทึกข้อมูลเป็นส่วนที่ทำการบันทึกข้อมูลที่ได้จากส่วนการสร้างสัญกรณ์เซตลงในแฟ้มข้อมูลที่ต้องการ โดยจะอยู่ในรูปแฟ้มตัวอักษรแทตของลาเท็กซ์ของ Z/EVES ซึ่งแบ่งเป็นเค้าร่างต่าง ๆ ได้แก่เค้า

ร่างซึ่งนิยามชนิดของข้อมูล เค้าร่างเอนทิตี เค้าร่างเพิ่มเติม เค้าร่างความสัมพันธ์ เค้าร่างการเพิ่ม เค้าร่างการแก้ไข และเค้าร่างการลบ ซึ่งแฟ้มข้อมูลสำหรับบันทึกข้อมูลนี้ จะมีนามสกุลเป็น .ZED

ตัวอย่างแทคของลาเท็กซ์

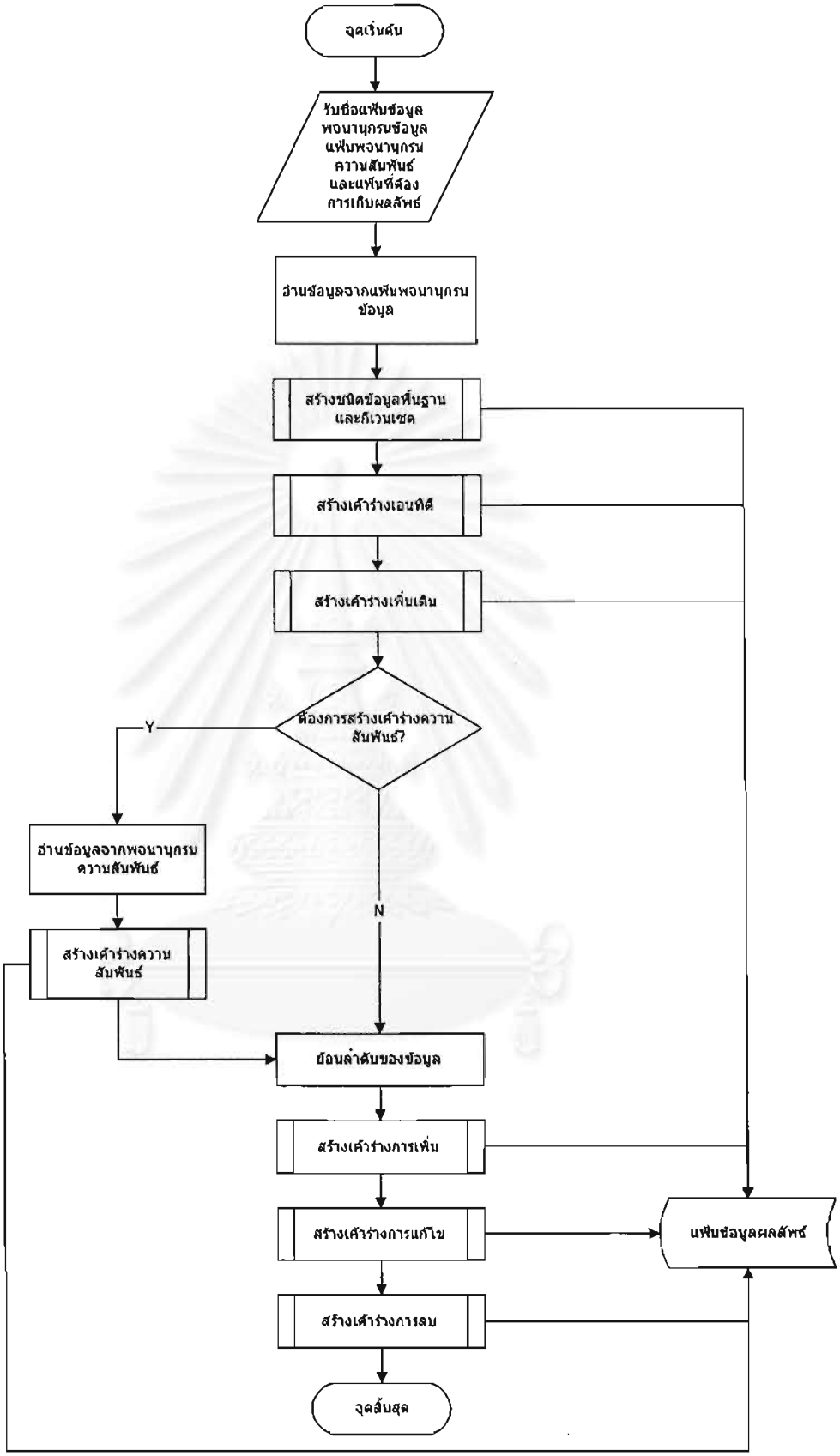
```
\begin{zed}
[string, Fueltype]
\end{zed}

\begin{schema}{vanclass}
className : string \also
fuleType : Fueltype
\end{schema}
```

จากแทคของลาเท็กซ์ที่แสดงสามารถเทียบได้กับข้อกำหนดรูปนัยในรูปสัญกรณ์เซดที่เขียนแบบโครงร่างเหมือนกล่องได้ดังรูปที่ 4.5



รูปที่ 4.5 รูปแสดงสัญกรณ์เซดที่เขียนแบบโครงร่างเหมือนกล่อง

นอกจากนี้ ยังมีแฟ้มข้อมูลซึ่งเก็บผลซึ่งไม่สามารถสร้างเป็นสัญกรณ์เซดได้ เนื่องจากข้อมูลที่ให้ไม่มีความสมบูรณ์ เช่น ไม่มีการระบุชื่อลักษณะประจำ ไม่มีการระบุชนิดของลักษณะประจำ หรือไม่มีการระบุชื่อเอนทิตีของคีย์นอก เป็นต้น โดยแฟ้มที่เก็บข้อมูลเหล่านี้จะมีชื่อว่า ERZ.LOG ซึ่งข้อมูลในแฟ้มข้อมูลนี้จะถูกแสดงเมื่อมีการสั่งให้สร้างสัญกรณ์เซดแล้วมีข้อมูลบางส่วนไม่สามารถสร้างเป็นสัญกรณ์เซดได้



รูปที่ 4.4 รูปแสดงผังงานการสังเคราะห์สัญกรณ์เซต

4.4. ผังโครงสร้างของระบบ

จากที่ได้กล่าวมา จะสามารถแสดงให้เห็นถึงภาพการทำงานของระบบโดยรวม เริ่มตั้งแต่การรับชื่อแฟ้มข้อมูล ไปจนถึงการแสดงผลลัพธ์สัญกรณ์เซตโดยแบ่งการทำงานต่าง ๆ ออกเป็นโมดูล ซึ่งสามารถแสดงเป็นผังโครงสร้างได้ดังรูปที่ 4.6 ซึ่งสามารถอธิบายการทำงานในส่วนต่าง ๆ ของระบบ ได้ดังนี้

โมดูล 1 รับชื่อแฟ้มข้อมูล

โมดูลรับชื่อแฟ้มข้อมูล คือการรับชื่อแฟ้มข้อมูลพจนานุกรมข้อมูล แฟ้มข้อมูลพจนานุกรมความสัมพันธ์ และ ชื่อแฟ้มที่ต้องการเก็บผลลัพธ์

โมดูล 1.1 รับชื่อแฟ้มข้อมูลพจนานุกรมข้อมูล

โมดูลรับชื่อแฟ้มข้อมูลพจนานุกรมข้อมูล คือ การรับชื่อแฟ้มพจนานุกรมข้อมูล แล้วทำการตรวจสอบว่าชื่อแฟ้มนั้นเป็นแฟ้มที่มีอยู่แล้วหรือไม่

โมดูล 1.2 รับชื่อแฟ้มข้อมูลพจนานุกรมความสัมพันธ์

โมดูลรับชื่อแฟ้มข้อมูลพจนานุกรมความสัมพันธ์ คือ การรับชื่อแฟ้มพจนานุกรมความสัมพันธ์ แล้วทำการตรวจสอบว่าชื่อแฟ้มนั้นเป็นแฟ้มมีอยู่แล้วหรือไม่

โมดูล 1.3 รับชื่อแฟ้มข้อมูลผลลัพธ์

โมดูลรับชื่อแฟ้มข้อมูลผลลัพธ์ คือ การรับชื่อแฟ้มข้อมูลที่จะใช้เก็บผลลัพธ์สัญกรณ์เซตที่ได้จากการสร้างด้วยเครื่องมือนี้ แล้วทำการสร้างแฟ้มขึ้นมาใหม่เพื่อบันทึกผลลัพธ์

โมดูล 2 อ่านข้อมูล

โมดูลอ่านข้อมูล คือ การอ่านข้อมูลจากแฟ้มข้อมูลพจนานุกรมข้อมูล และแฟ้มข้อมูลพจนานุกรมความสัมพันธ์เข้ามาไว้ในระบบ

โมดูล 3 สังเคราะห์คุณสมบัติโครงสร้าง

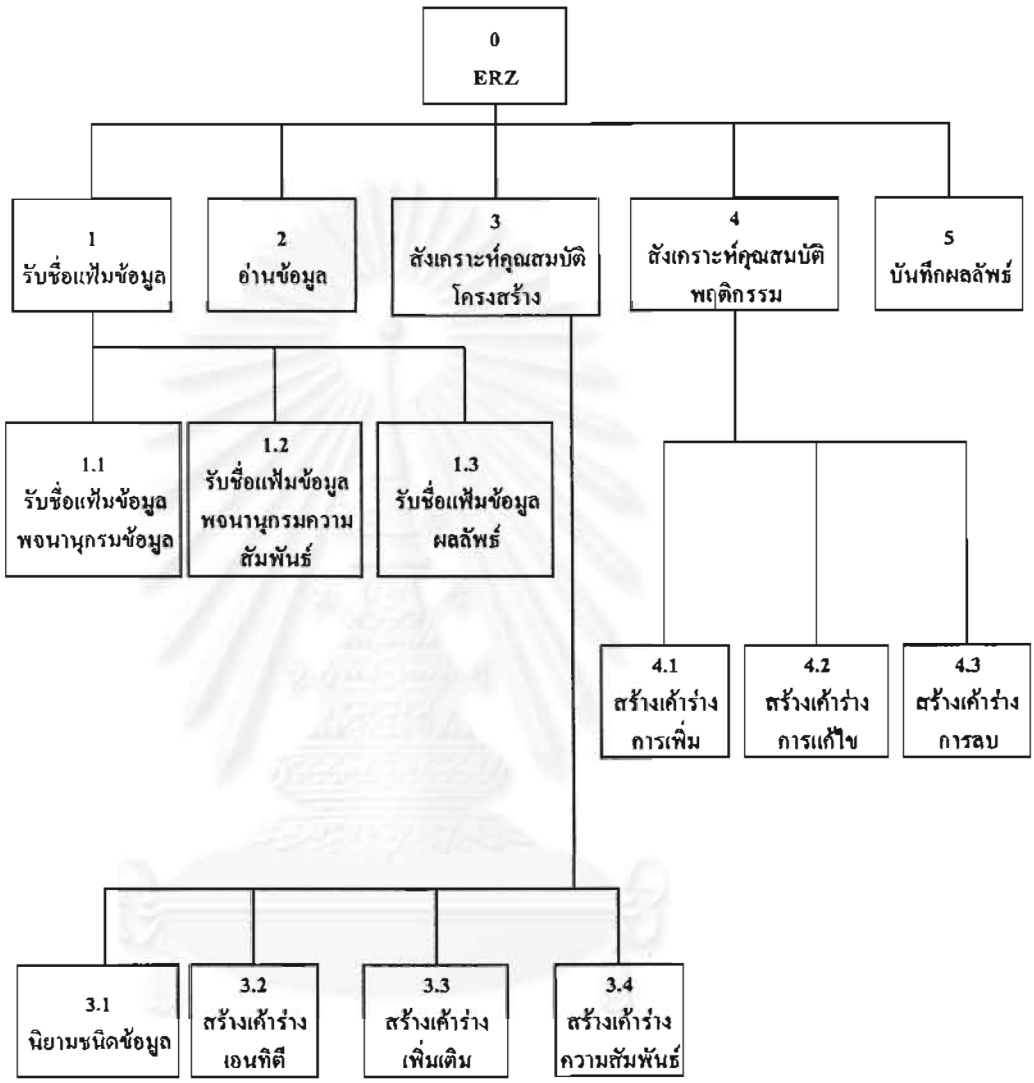
โมดูลสังเคราะห์คุณสมบัติโครงสร้าง คือ ส่วนสังเคราะห์สัญกรณ์เซต โดยจะสร้างในส่วนของโครงสร้างของข้อมูลซึ่งแบ่งเป็นโมดูลต่าง ๆ ได้เป็น เค้าร่างเอนทิตี เค้าร่างเพิ่มเติม และเค้าร่างความสัมพันธ์

โมดูล 3.1 นิยามชนิดข้อมูล

โมดูลนิยามชนิดข้อมูล คือส่วนที่ทำการนิยามชนิดข้อมูลพื้นฐานต่าง ๆ ได้แก่ ชนิดตัวอักษร ชนิดสายอักษร ชนิดบูลีน และชนิดข้อมูลนามธรรมอื่น ๆ ซึ่งจะนิยามเป็นกีเวนเซต

โมดูล 3.2 สร้างเค้าร่างเอนทิตี

โมดูลสร้างเค้าร่างเอนทิตี คือ การสร้างเค้าร่างเอนทิตี ซึ่งสามารถแบ่งได้เป็น 2 ส่วน คือการสร้างส่วนการประกาศ ซึ่งจะเป็นการประกาศตัวแปรของเค้าร่าง โดยการนำลักษณะประจำของเอนทิตีมาเป็นตัวแปรของเค้าร่าง และการสร้างส่วนสัจพจน์ ซึ่งเป็นการประกาศเงื่อนไขต่าง ๆ ของตัวแปรที่ประกาศเพิ่มเติม



รูปที่ 4.6 ผังโครงสร้างแสดงส่วนการทำงานต่าง ๆ ของระบบ

โมดูล 3.3 สร้างเค้าร่างเพิ่มเติม

โมดูลสร้างเค้าร่างเพิ่มเติม คือ การสร้างเค้าร่างเพิ่มเติม ซึ่งสามารถแบ่งได้เป็น 2 ส่วน คือ การสร้างส่วนการประกาศ ซึ่งจะเป็นการประกาศข้อมูลเป็นชนิดเซตของข้อมูลที่ได้ประกาศเป็นเค้าร่างเอนทิตี เพื่อให้ข้อมูลมีลักษณะคล้ายกับตารางในระบบฐานข้อมูล และการสร้างส่วนสัจพจน์ ซึ่งเป็นการประกาศถึงคุณสมบัติของเอนทิตี ว่ามีลักษณะประจำใดเป็นคีย์หลัก และคีย์นอก

โมดูล 3.4 สร้างเค้าร่างความสัมพันธ์

โมดูลสร้างเค้าร่างความสัมพันธ์ คือ การสร้างเค้าร่างความสัมพันธ์ ซึ่งสามารถแบ่งได้เป็น 2 ส่วน คือ การสร้างส่วนการประกาศ ซึ่งเป็นการประกาศว่าเอนทิตีใดที่มีความสัมพันธ์แก่กัน และการสร้างส่วนสัจพจน์ ซึ่งเป็นการประกาศว่าเอนทิตีที่มีความสัมพันธ์แก่กันนั้น มีความสัมพันธ์กันแบบใด

โมดูล 4 สังเคราะห์คุณสมบัติพฤติกรรม

โมดูลสังเคราะห์คุณสมบัติพฤติกรรม คือ ส่วนสังเคราะห์คำสั่งพื้นฐานสำหรับเอนทิตี ซึ่งสร้างได้เป็น เค้าร่างการเพิ่ม เค้าร่างการแก้ไข และเค้าร่างการลบ

โมดูล 4.1 สร้างเค้าร่างการเพิ่ม

โมดูลสร้างเค้าร่างการเพิ่ม คือ การสร้างเค้าร่างการเพิ่ม ซึ่งสามารถแบ่งได้เป็น 2 ส่วนคือ การสร้างส่วนการประกาศ ซึ่งจะเป็นการประกาศตัวแปรของเค้าร่าง ที่จะใช้ในการนำเข้าค่าของข้อมูลที่จะทำการเพิ่ม และการสร้างส่วนสัจพจน์ ซึ่งเป็นการนำค่าที่ได้รับเข้ามา เพิ่มเข้าไปกับข้อมูลที่มีอยู่

โมดูล 4.2 สร้างเค้าร่างการแก้ไข

โมดูลสร้างเค้าร่างการแก้ไข คือ การสร้างเค้าร่างการแก้ไข ซึ่งสามารถแบ่งได้เป็น 2 ส่วนคือ การสร้างส่วนการประกาศ ซึ่งจะเป็นการประกาศตัวแปรของเค้าร่างที่ใช้ในการนำเข้าข้อมูลที่ต้องการแก้ไข และการสร้างส่วนสัจพจน์ ซึ่งเป็นการนำค่าที่ได้รับเข้ามาไปค้นหาข้อมูลที่ต้องการแก้ไขตามคีย์หลัก แล้วทำการแก้ไขข้อมูลตามค่าที่ได้รับเข้ามาใหม่

โมดูล 4.3 สร้างเค้าร่างการลบ

โมดูลสร้างเค้าร่างการลบคือ การสร้างเค้าร่างการลบ ซึ่งสามารถแบ่งได้เป็น 2 ส่วนคือ การสร้างส่วนการประกาศ ซึ่งจะเป็นการประกาศตัวแปรของเค้าร่างที่จะนำเข้าข้อมูลที่ต้องการลบ และประกาศส่วนการลบสำหรับเอนทิตีลูก เพื่อรวมเอาส่วนการลบของเอนทิตีลูกเข้าไว้ด้วยกัน และการสร้างส่วนสัจพจน์ ซึ่งเป็นการประกาศส่วนการส่งค่าที่จะลบให้เอนทิตีลูก คือการประกาศการส่งค่าที่จะลบให้กับตัวแปรของเอนทิตีลูก เพื่อให้ใช้ในการลบสำหรับข้อมูลในเอนทิตีลูกที่อ้างอิงถึงข้อมูลในเอนทิตีพ่อที่กำลังจะลบ และประกาศส่วนการลบสำหรับเอนทิตีพ่อ คือการลบข้อมูลที่ต้องการจะลบออกจากเอนทิตีพ่อ

โมดูล 5 บันทึกผลลัพธ์

โมดูลบันทึกผลลัพธ์ คือ ส่วนที่บันทึกผลลัพธ์สัญกรณ์เชตที่ได้จากการสังเคราะห์ ลงในแฟ้มข้อมูล

4.5 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือซอฟต์แวร์

เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนามีรายละเอียดดังนี้

- คอมพิวเตอร์แบบพีซี PentiumIII 450 เมกะเฮิร์ส
- หน่วยความจำ 64 เมกะไบต์
- ฮาร์ดดิสต์ 6 กิกะไบต์

ซอฟต์แวร์ที่ใช้ในการพัฒนามีรายละเอียดดังนี้

- ระบบปฏิบัติการวินโดว์ส98
- เดลไฟล์ 4.0

I 19066383

บทที่ 5
การทดสอบ และสรุปผล

ในบทนี้จะกล่าวถึงการทดสอบเครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด ภายใต้ระบบปฏิบัติการวินโดว์ส 98

5.1. ขั้นตอนการติดตั้ง

ทำการติดตั้งเครื่องมือนี้โดยการเรียก SETUP.EXE แล้วปฏิบัติตามขั้นตอนที่โปรแกรมแนะนำเพื่อเป็นการติดตั้งเครื่องมือ

5.2. สภาวะที่ใช้ทดสอบโปรแกรม

เครื่องคอมพิวเตอร์ที่ใช้ในการทดสอบมีรายละเอียดดังนี้

- คอมพิวเตอร์แบบพีซี Pentium III 450 เมกะเฮิร์ส
- หน่วยความจำ 64 เมกะไบต์
- ฮาร์ดดิสต์ 6 กิกะไบต์

5.3. กรณีทดสอบที่ใช้ทดสอบโปรแกรม

1) ระบบให้เช่ารถแวน [8]

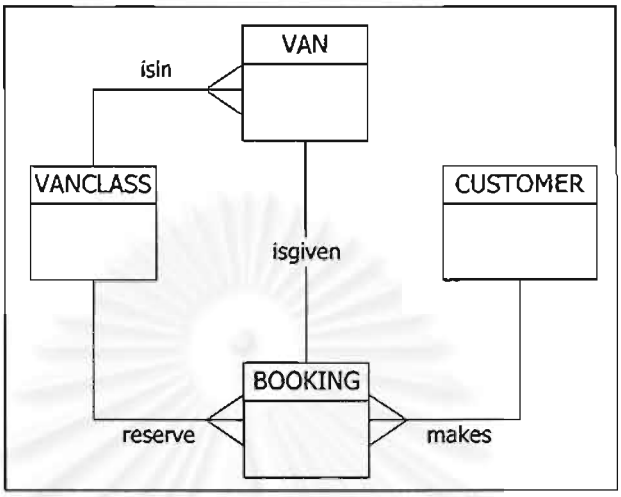
ระบบให้เช่ารถแวนเป็นระบบสำหรับอู่ที่ให้บริการเช่ารถแวน จะคล้ายกับระบบให้เช่ารถยนต์ทั่วไป แต่จะแตกต่างกันตรงที่รถที่ให้เช่ามีเฉพาะรถแวนโดยสามารถแบ่งประเภทของรถแวนได้เป็นประเภทต่าง ๆ เช่น มินิแวน แวนสำหรับขนส่ง และแวนสำหรับงานหนัก เป็นต้น โดยในการเช่ารถแวน ลูกค้าจะทำการจองโดยการระบุชนิดของรถแวนที่ต้องการเช่าล่วงหน้า และเมื่อมารับรถทางอู่จะจัดหารถแวนตามชนิดที่จองไว้ให้บริการ ดังแสดงเป็นแผนภาพเอนทิตีและความสัมพันธ์ในรูปที่ 5.1

จากรูป 5.1 แสดงความสัมพันธ์ของเอนทิตีต่าง ๆ ในระบบ ซึ่งรายละเอียดของลักษณะประจำความสัมพันธ์ และคีย์ ของแต่ละเอนทิตี สามารถดูได้ในภาคผนวก ก.

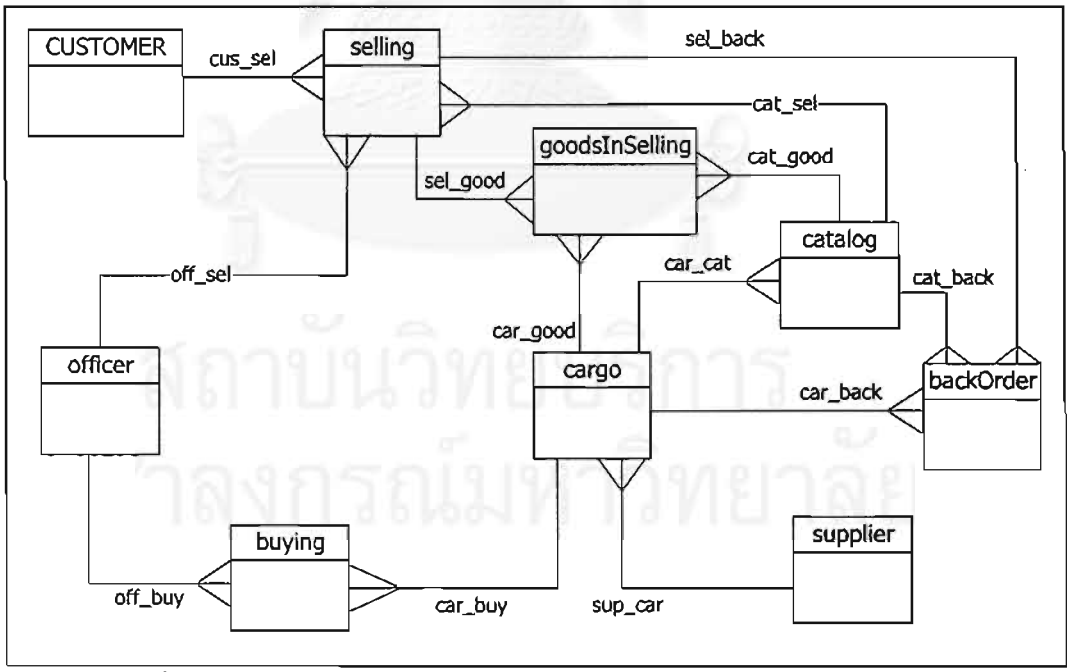
2) ระบบขายของผ่านโทรศัพท์

ระบบขายของผ่านโทรศัพท์เป็นระบบการให้บริการซื้อของผ่านทางโทรศัพท์ โดยลูกค้าที่จะซื้อได้จะต้องถือบัตรเครดิตของธนาคาร ABC ซึ่งเมื่อมีการใช้จ่ายผ่านบัตรเครดิต ลูกค้าจะได้รับคะแนนสะสม ซึ่งสามารถใช้คะแนนสะสมนั้นมาใช้ในการแลกซื้อสินค้าที่ธนาคารทำการกำหนดไว้ โดยสินค้าที่

จะแลกซื้อได้นั้นจะจัดทำเป็นรายการสินค้าสำหรับแต่ละเดือน นอกจากนี้ ถ้าหากสินค้าที่ลูกค้าต้องการซื้อหมด ทางบริษัทจะรับการสั่งซื้อสินค้านั้นๆไว้ และจะทำการจัดส่งสินค้าให้ภายหลัง จากระบบการขายผ่านโทรศัพท์นี้ ดังแสดงเป็นแผนภาพเอนทิตีและความสัมพันธ์ในรูปที่ 5.2



รูปที่ 5.1 แผนภาพเอนทิตี และความสัมพันธ์ของระบบให้เช่ารถแวน

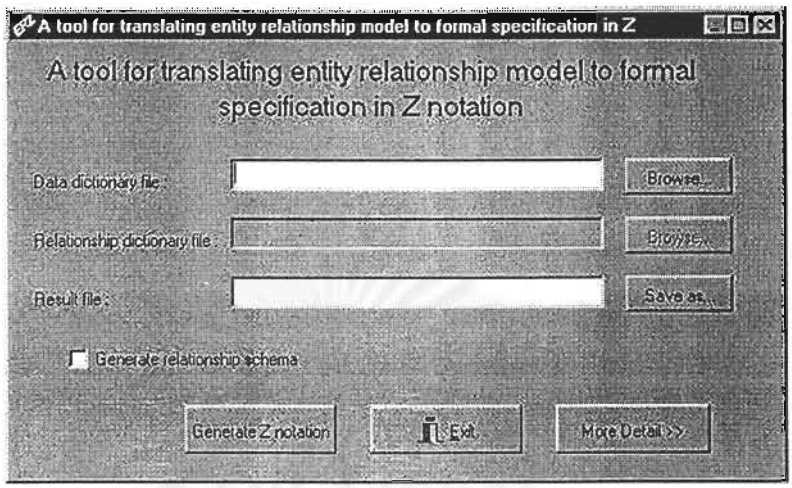


รูปที่ 5.2 แผนภาพเอนทิตี และความสัมพันธ์ของระบบการขายของผ่านโทรศัพท์

จากรูปที่ 5.2 แสดงความสัมพันธ์ของเอนทิตีต่าง ๆ ในระบบ ซึ่งมีรายละเอียดของลักษณะประจำความสัมพันธ์ และคีย์ ของแต่ละเอนทิตี สามารถดูได้ในภาคผนวก ข.

5.4. ขั้นตอนการทดสอบ

1) เรียกโปรแกรม ERZ.EXE เพื่อทำการทดสอบ ดังรูป 5.3



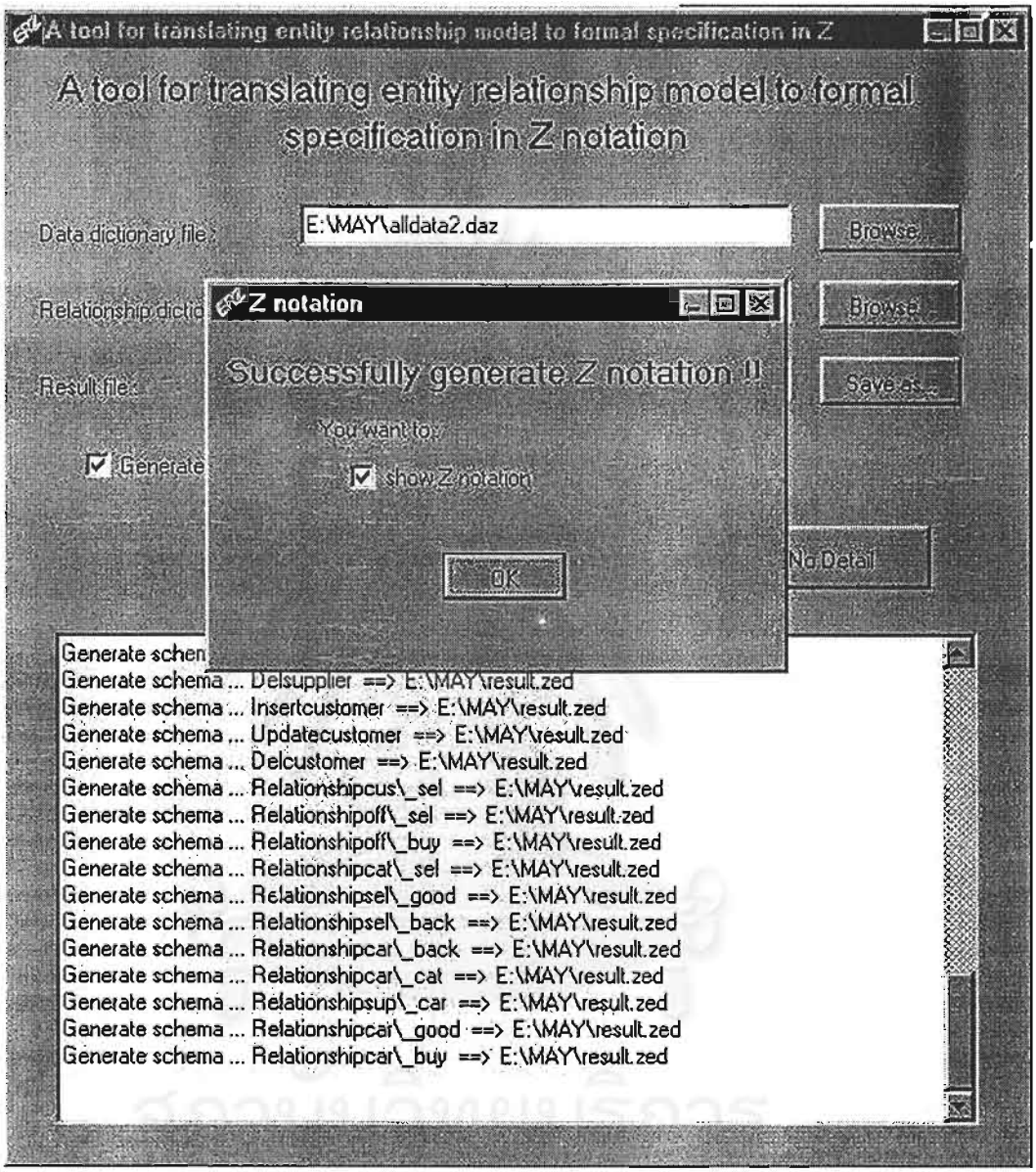
รูปที่ 5.3 หน้าจอเครื่องมือ

2) ใส่ชื่อแฟ้มที่เป็นแฟ้มข้อความสำหรับพจนานุกรมข้อมูล (*.DAZ)
3) ใส่ชื่อแฟ้มที่เป็นแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ (*.REZ)
4) ใส่ชื่อแฟ้มที่จะใช้สำหรับบันทึกผลลัพธ์
5) เลือกสร้างเค้าร่างความสัมพันธ์เพื่อทำการสร้างเค้าร่างความสัมพันธ์ด้วย
6) กดปุ่ม

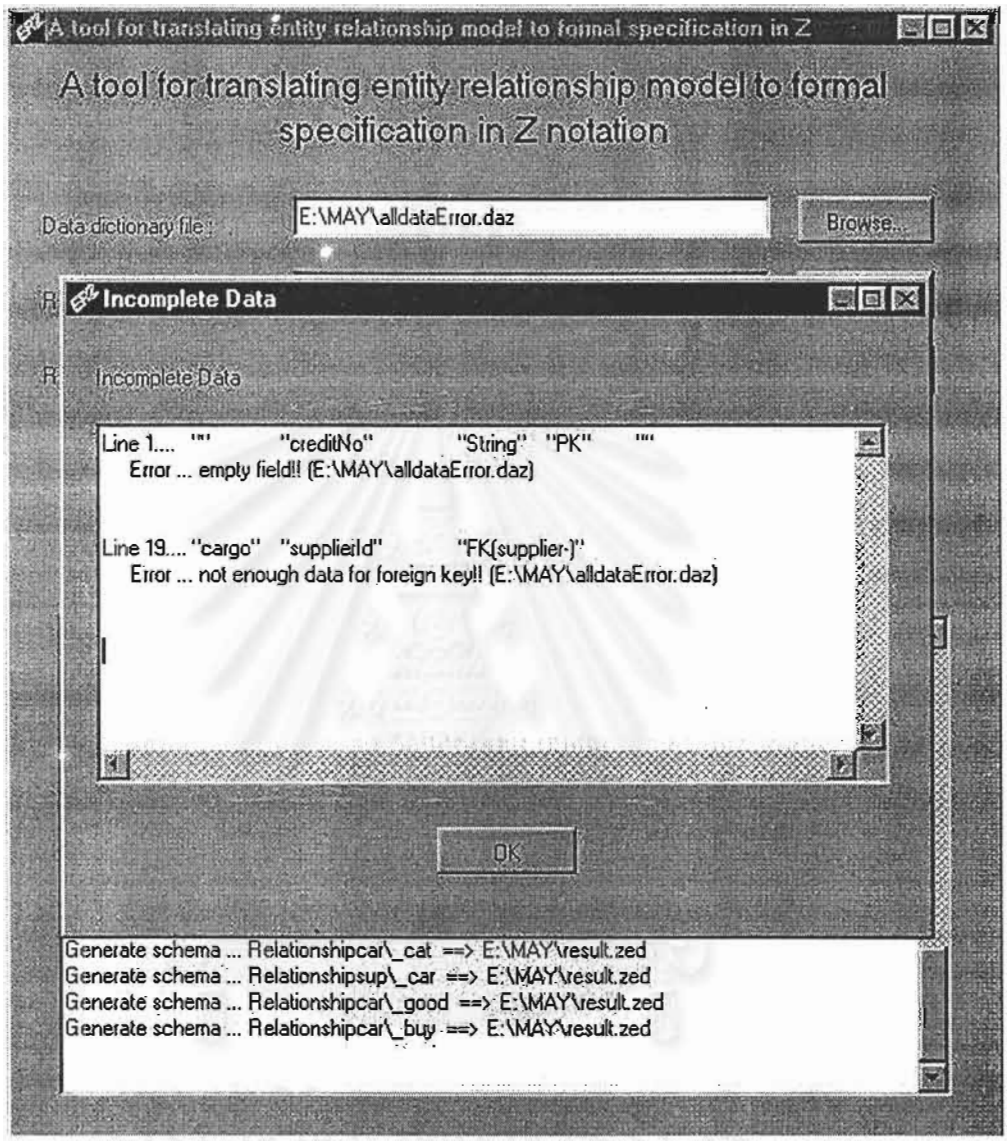
เพื่อสร้างสัญกรณ์เซด
   - ถ้าหากข้อมูลที่ได้รับมีความสมบูรณ์ สามารถสร้างสัญกรณ์เซดได้อย่างสมบูรณ์ จะได้ผลดังรูปที่ 5.4
   - ถ้าหากข้อมูลที่ได้รับมีบางส่วนไม่สมบูรณ์ทำให้ไม่สามารถสร้างสัญกรณ์เซดได้อย่างสมบูรณ์ จะได้ผลดังรูปที่ 5.5

5.5. ผลการทดสอบโปรแกรม

จากการทดสอบพบว่าเครื่องมือนี้สามารถทำการแปลงข้อมูลจากแบบจำลองเอนทิตี และ ความสัมพันธ์เป็นข้อกำหนดรูปนัย ในรูปสัญกรณ์เซดได้ โดยอยู่ในรูปแทคของลาเท็กซ์ของ Z/EVES ทั้งนี้ จากการทดสอบกับระบบให้เช่ารถแวนสามารถสร้างเป็นเค้าร่างต่าง ๆ ได้ทั้งหมด 24 เค้าร่าง โดยแบ่งเป็น เค้าร่างเอนทิตี 4 เค้าร่าง เค้าร่างเพิ่มเติม 4 เค้าร่าง เค้าร่างความสัมพันธ์ 4 เค้าร่าง เค้าร่างการเพิ่ม 4 เค้าร่าง เค้าร่างการแก้ไข 4 เค้าร่าง และ เค้าร่างการลบ 4 เค้าร่าง ดังแสดงตัวอย่างในรูป 5.6 และ 5.7 ซึ่งรายละเอียดของเค้าร่างอื่น ๆ แสดงอยู่ในภาคผนวก ก.



รูปที่ 5.4 หน้าจอเครื่องมือเมื่อทำการสร้างสัญกรณ์เซดเสร็จสมบูรณ์



รูปที่ 5.5 หน้าจอเครื่องมือเมื่อข้อมูลนำเข้าไม่สมบูรณ์

จากการทดสอบกับกรณีทดสอบที่ 2 ระบบการขายของผ่านโทรศัพท์สามารถสร้างเป็นเค้าร่างต่าง ๆ ได้ทั้งหมด 58 เค้าร่าง โดยแบ่งเป็น เค้าร่างเอนทิตี 9 เค้าร่าง เค้าร่างเพิ่มเติม 9 เค้าร่าง เค้าร่างความสัมพันธ์ 13 เค้าร่าง เค้าร่างการเพิ่ม 9 เค้าร่าง เค้าร่างการแก้ไข 9 เค้าร่าง และ เค้าร่างการลบ 9 เค้าร่าง ดังแสดงตัวอย่างในรูป 5.8 และ 5.9 ซึ่งรายละเอียดของเค้าร่างอื่น ๆ แสดงอยู่ในภาคผนวก ข.

5.6. ผลการทดสอบข้อกำหนดรูปนัยด้วย Z/EVES

ในการตรวจสอบความถูกต้องทางด้านไวยากรณ์ได้ใช้ Z/EVES ในการทดสอบโดย Z/EVES จะพิจารณาความถูกต้องทางด้านไวยากรณ์ทีละเค้าร่าง ถ้าหากว่าเค้าร่างนั้นถูกต้อง ก็จะทำการสร้างเป็นทฤษฎีบท (Theorem)ไว้ แต่ถ้าหากว่าเค้าร่างนั้นไม่ถูกต้อง ก็จะแสดงข้อความซึ่งแสดงถึงความผิดพลาดของไวยากรณ์

จากการนำสัญกรณ์เซดที่ได้จากเครื่องมือแปลง มาทำการทดสอบความถูกต้องทางด้านไวยากรณ์ พบว่าข้อมูลที่ได้จากเครื่องมือนี้มีความถูกต้องทางด้านไวยากรณ์ของสัญกรณ์เซด คือ Z/EVES สามารถสร้างทฤษฎีบทจากเค้าร่างได้ โดยไม่มีการแสดงข้อผิดพลาดใด ๆ ในกรณีที่ข้อมูลนำเข้ามีความถูกต้อง

```
\begin{schema} {van}
reg : string \also
model : string \also
class : string \also
\end{schema}
```

รูปที่ 5.6 แสดงเค้าร่าง van

```
\begin{schema} {vanExt}
vanSet : \finset van \also
\where
\forall van1,van2 : van | van1 \in vanSet \land van2 \in vanSet \land van1 \neq van2 @ van1.reg \neq
van2.reg \also
\forall vanVar : van @( \exists vanclassVar : vanclass @ vanVar.class = vanclassVar.classname) \also
\end{schema}
```

รูปที่ 5.7 แสดงเค้าร่าง vanExt

```
\begin{schema} {customer}
creditNo : String \also
customer : String \also
customAdd : String \also
creditExpire : String \also
point : \num \also
paid : float \also
customTel : String \also
customTax : String \also
customFlag : Boolean \also
\end{schema}
```

รูปที่ 5.8 แสดงเค้าร่าง customer

```
\begin{schema} {customerExt}
customerSet : \finset customer \also
\where
\forall customer1,customer2 : customer | customer1 \in customerSet \land customer2 \in customerSet
\land customer1 \neq customer2 @ customer1.creditNo \neq customer2.creditNo \also
\end{schema}
```

รูปที่ 5.9 แสดงเค้าร่าง customerExt

5.7. การตรวจสอบความหมายของผลลัพธ์สัญกรณ์เชด

ในการตรวจสอบความหมายของผลลัพธ์สัญกรณ์เชด ได้ทำการตรวจสอบโดยการเปรียบเทียบความหมายของแบบจำลองเอนทิตีและความสัมพันธ์กับผลลัพธ์สัญกรณ์เชด การตรวจสอบจะแบ่งเป็นขั้นตอนย่อย ๆ ดังนี้

1) ขั้นตอนตรวจสอบโครงสร้างของเอนทิตี เป็นการตรวจสอบว่ามีส่วนที่ได้ทำการนิยามถึงโครงสร้างของเอนทิตีแต่ละเอนทิตีว่าประกอบไปด้วยลักษณะประจำใด โดยจำนวนเค้าร่างเอนทิตีจะต้องเท่ากับจำนวนเอนทิตี และตัวแปรที่ประกาศในเค้าร่างเอนทิตีจะต้องเป็นลักษณะประจำของเอนทิตี
2) ขั้นตอนตรวจสอบการกำหนดคีย์ให้แก่เอนทิตี เป็นการตรวจสอบว่ามีการนิยามความเป็นคีย์หลักและคีย์นอกสำหรับแต่ละเอนทิตี โดยจำนวนเค้าร่างเพิ่มเติมจะต้องเท่ากับจำนวนเอนทิตี และมีการนิยามคีย์หลัก และคีย์นอกได้อย่างถูกต้อง

3) ขั้นตอนตรวจสอบความสัมพันธ์ระหว่างเอนทิตี เป็นการตรวจสอบว่ามีการนิยามลักษณะความสัมพันธ์ระหว่างเอนทิตีอย่างครบถ้วน โดยจำนวนเค้าร่างความสัมพันธ์จะต้องเท่ากับจำนวนความสัมพันธ์ที่เกิดขึ้น และมีการนิยามลักษณะความสัมพันธ์ระหว่างเอนทิตีได้อย่างถูกต้อง
4) ขั้นตอนตรวจสอบคำสั่งการเพิ่มระเบียนข้อมูล เป็นการตรวจสอบว่ามีการนิยามคำสั่งสำหรับการเพิ่มระเบียนให้กับเอนทิตี โดยจำนวนเค้าร่างการเพิ่มระเบียนข้อมูลต้องเท่ากับจำนวนเอนทิตี และเนื่องจากการเพิ่มระเบียนจะต้องคำนึงว่าคีย์หลักของระเบียนที่เพิ่มเข้าไปจะต้องไม่ซ้ำกับคีย์หลักของระเบียนที่มีอยู่เดิม และคีย์นอกของระเบียนที่เพิ่มเข้าไปจะต้องมีค่าอยู่ในเอนทิตีที่อ้างอิงถึง ซึ่งในเค้าร่างการเพิ่มได้มีการรวมเค้าร่างเพิ่มเติมของเอนทิตีจึงเสมือนกับการใส่เงื่อนไขของคีย์หลักและคีย์นอกภายในเค้าร่างการเพิ่มเอง
5) ขั้นตอนตรวจสอบคำสั่งการแก้ไขระเบียน เป็นการตรวจสอบว่ามีการนิยามคำสั่งสำหรับการแก้ไขระเบียนให้กับเอนทิตี โดยจำนวนเค้าร่างการแก้ไขระเบียนข้อมูลจะต้องเท่ากับจำนวนเอนทิตี และเนื่องจากการแก้ไขระเบียนจะต้องคำนึงว่าคีย์หลักของระเบียนที่แก้ไข จะต้องไม่ซ้ำกับคีย์หลักของระเบียนที่มีอยู่เดิม และคีย์นอกของระเบียนที่แก้ไขจะต้องมีค่าอยู่ในเอนทิตีที่อ้างอิงถึง ซึ่งในเค้าร่างการแก้ไขได้มีการรวมเค้าร่างเพิ่มเติมของเอนทิตีจึงเสมือนกับการใส่เงื่อนไขของคีย์หลักและคีย์นอกภายในเค้าร่างการแก้ไข เอง
6) ขั้นตอนตรวจสอบคำสั่งการลบระเบียน เป็นการตรวจสอบว่ามีการนิยามคำสั่งสำหรับการลบระเบียน โดยจำนวนเค้าร่างการลบระเบียนข้อมูลจะต้องเท่ากับจำนวนเอนทิตี และถ้าหากระเบียนที่จะลบถูก อ้างอิงจากระเบียนข้อมูลอื่นก็จะต้องทำการลบระบียนข้อมูลนั้น ๆ ด้วย โดยจะต้องมีการรวมเค้าร่างสำหรับเค้าร่างการลบของเอนทิตีที่มาอ้างอิงถึงเข้ากับเค้าร่างการลบของเอนทิตีที่จะลบ

5.7.1) การตรวจสอบความหมายของผลลัพธ์สัญกรณ์เซตของระบบให้เช่ารถแวน

ระบบให้เช่ารถแวนประกอบด้วยเอนทิตีทั้งหมด 4 เอนทิตี และความสัมพันธ์ทั้งหมด 4 ความสัมพันธ์ได้ทำการตรวจสอบตามขั้นตอนในการตรวจสอบ 6 ขั้นตอนซึ่งได้ผลดังนี้

1) เมื่อตรวจสอบตามขั้นตอนตรวจสอบโครงสร้างของเอนทิตี พบว่าเครื่องมือซอฟต์แวร์สามารถสร้างเค้าร่างเอนทิตีได้ 4 เค้าร่างคือ เค้าร่างvan เค้าร่างvanclass เค้าร่างbooking และ เค้าร่าง customer
2) เมื่อตรวจสอบตามขั้นตอนการกำหนดคีย์ให้แก่เอนทิตี พบว่าเครื่องมือซอฟต์แวร์สามารถสร้างเค้าร่างเพิ่มเติมได้ 4 เค้าร่าง คือ เค้าร่างvanExt เค้าร่างvanclassExt เค้าร่างbookingExt และ เค้าร่างcustomerExt ซึ่งทั้ง 4 เค้าร่างมีการนิยามคีย์หลัก และคีย์นอกตามพจนานุกรมข้อมูลของแต่ละเอนทิตี
3) เมื่อตรวจสอบตามขั้นตอนตรวจสอบความสัมพันธ์ระหว่างเอนทิตี พบว่าเครื่องมือซอฟต์แวร์สามารถสร้างเค้าร่างความสัมพันธ์ได้ 4 เค้าร่าง คือ เค้าร่างRelationshipisin เค้าร่าง

Relationshipisgiven เค้าร่างRelationshipreserve และ เค้าร่างRelationshipmakes ซึ่งทั้ง 4 เค้าร่างมีการนิยามความสัมพันธ์ตามพจนานุกรมความสัมพันธ์

4) เมื่อตรวจสอบตามขั้นตอนตรวจสอบคำสั่งการเพิ่มระเบียนข้อมูล พบว่าเครื่องมือซอฟต์แวร์สามารถสร้างเค้าร่างการเพิ่มได้ 4 เค้าร่าง คือ เค้าร่างInsertvan เค้าร่างInsertvanclass เค้าร่างInsertbooking และ เค้าร่างInsertcustomer
5) เมื่อตรวจสอบตามขั้นตอนตรวจสอบคำสั่งการแก้ไขระเบียน พบว่าเครื่องมือซอฟต์แวร์สามารถสร้างเค้าร่างการแก้ไขได้ 4 เค้าร่างคือ เค้าร่างUpdatevan เค้าร่างUpdatevanclass เค้าร่างUpdatebooking และ เค้าร่างUpdatecustomer
6) เมื่อตรวจสอบตามขั้นตอนตรวจสอบคำสั่งการลบระเบียน พบว่าเครื่องมือซอฟต์แวร์สามารถสร้างเค้าร่างการลบได้ 4 เค้าร่างคือ เค้าร่างDelvan เค้าร่างDelvanclass เค้าร่างDelbooking และ เค้าร่างDelcustomer ซึ่งทั้ง 4 เค้าร่างมีการนิยามให้มีการลบข้อมูลของเอนทิตีอื่นที่มาอ้างอิงถึงข้อมูลของเอนทิตีที่จะลบ

จากการตรวจสอบสามารถสรุปการตรวจสอบผลลัพธ์สัญกรณ์เซดของระบบให้เช่ารถแวนได้ดังตารางที่ 5.1

ตารางที่ 5.1 แสดงสรุปผลการตรวจสอบผลลัพธ์สัญกรณ์เซดของระบบให้เช่ารถแวน

สิ่งที่ตรวจสอบ	ผลการทดสอบ	
	ถูกต้อง	ไม่ถูกต้อง
ตรวจสอบเอนทิตี		
จำนวนเอนทิตี	✔	
ลักษณะประจำในเอนทิตี	✔	
คีย์ของเอนทิตี	✔	
จำนวนความสัมพันธ์ระหว่างเอนทิตี	✔	
ลักษณะความสัมพันธ์ระหว่างเอนทีตี	✔	
ตรวจสอบคำสั่งพื้นฐานของเอนทิตี		
การเพิ่มระเบียน	✔	
การแก้ไขระเบียน	✔	
การลบระเบียนที่ไม่มีระเบียนอื่นอ้างอิง	✔	
การลบระเบียนที่ถูกระเบียนอื่นอ้างถึง	✔	

5.7.2) การตรวจสอบความหมายของผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์

ระบบการขายของผ่านโทรศัพท์ประกอบด้วยเอนทิตีทั้งหมด 9 เอนทิตี และความสัมพันธ์ทั้งหมด 13 ความสัมพันธ์ได้ทำการตรวจสอบตามขั้นตอนในการตรวจสอบ 6 ขั้นตอนซึ่งได้ผลดังนี้

1) เมื่อตรวจสอบตามขั้นตอนตรวจสอบโครงสร้างของเอนทิตี พบว่าเครื่องมือสามารถสร้างเค้าร่างเอนทิตีได้ 9 เค้าร่างคือ เค้าร่างcustomer เค้าร่างselling เค้าร่างgoodsInSelling เค้าร่างcatalog เค้าร่างofficer เค้าร่างcargo เค้าร่างbuying เค้าร่างbackOrder และ เค้าร่างsupplier
2) เมื่อตรวจสอบตามขั้นตอนการกำหนดคีย์ให้แก่เอนทิตี พบว่าเครื่องมือสามารถสร้างเค้าร่างเพิ่มเติมได้ 9 เค้าร่าง คือ เค้าร่างcustomerExt เค้าร่างsellingExt เค้าร่างgoodsInSellingExt เค้าร่างcatalogExt เค้าร่างofficerExt เค้าร่างcargoExt เค้าร่างbuyingExt เค้าร่างbackOrderExt และเค้าร่างsupplierExt ซึ่งทั้ง 9 เค้าร่างมีการนิยามคีย์หลัก และคีย์นอกตามพจนานุกรมข้อมูลของแต่ละเอนทิตี
3) เมื่อตรวจสอบตามขั้นตอนตรวจสอบความสัมพันธ์ระหว่างเอนทิตี พบว่าเครื่องมือสามารถสร้างเค้าร่างความสัมพันธ์ได้ 13 เค้าร่าง คือ เค้าร่างRelationshipcus_sel เค้าร่างRelationshipoff_sel เค้าร่างRelationshipoff_buy เค้าร่างRelationshipcat_sel เค้าร่างRelationshipcat_good เค้าร่างRelationshipcat_back เค้าร่างRelationshipsel_good เค้าร่างRelationshipsel_back เค้าร่างRelationshipcar_back เค้าร่างRelationshipcar_cat เค้าร่างRelationshipsup_car เค้าร่างRelationshipcar_good และ เค้าร่างRelationshipcar_buy ซึ่งทั้ง 13 เค้าร่างมีการนิยามความสัมพันธ์ตามพจนานุกรมความสัมพันธ์
4) เมื่อตรวจสอบตามขั้นตอนตรวจสอบคำสั่งการเพิ่มระเบียนข้อมูล พบว่าเครื่องมือสามารถสร้างเค้าร่างการเพิ่มได้ 9 เค้าร่าง คือ เค้าร่างInsertcustomer เค้าร่างInsertselling เค้าร่างInsertgoodsInSelling เค้าร่างInsertofficer เค้าร่างInsertcatalog เค้าร่างInsertcargo เค้าร่างInsertbuying เค้าร่างInsertbackOrder และ เค้าร่างInsertsupplier
5) เมื่อตรวจสอบตามขั้นตอนตรวจสอบคำสั่งการแก้ไขระเบียน พบว่าเครื่องมือสามารถสร้างเค้าร่างการแก้ไขได้ 9 เค้าร่างคือ เค้าร่างUpdatecustomer เค้าร่างUpdateselling เค้าร่างUpdategoodsInSelling เค้าร่างUpdateofficer เค้าร่างUpdatecatalog เค้าร่างUpdatecargo เค้าร่างUpdatebuying เค้าร่างUpdatebackOrder และ เค้าร่างUpdatesupplier
6) เมื่อตรวจสอบตามขั้นตอนตรวจสอบคำสั่งการลบระเบียน พบว่าเครื่องมือสามารถสร้างเค้าร่างการลบได้ 9 เค้าร่างคือ เค้าร่างDelcustomer เค้าร่างDelselling เค้าร่างDelgoodsInSelling เค้าร่างDelofficer เค้าร่างDelcatalog เค้าร่างDelcargo เค้าร่างDelbuying เค้าร่างDelbackOrder และ เค้าร่างDelsupplier ซึ่งทั้ง 9 เค้าร่างมีการนิยามให้มีการลบข้อมูลของเอนทิตีอื่นที่มาอ้างอิงถึงข้อมูลของเอนทิตีที่จะลบ

จากการตรวจสอบสามารถสรุปการตรวจสอบผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ได้ดังตารางที่ 5.2

ตารางที่ 5.2 แสดงสรุปผลการตรวจสอบผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์

สิ่งที่ตรวจสอบ	ผลการทดสอบ	
	ถูกต้อง	ไม่ถูกต้อง
ตรวจสอบเอนทิตี		
จำนวนเอนทิตี	✔	
ลักษณะประจำในเอนทิตี	✔	
คีย์ของเอนทิตี	✔	
จำนวนความสัมพันธ์ระหว่างเอนทิตี	✔	
ลักษณะความสัมพันธ์ระหว่างเอนทีตี	✔	
ตรวจสอบคำสั่งพื้นฐานของเอนทิตี		
การเพิ่มระเบียน	✔	
การแก้ไขระเบียน	✔	
การลบระเบียนที่ไม่มีระเบียนอื่นอ้างอิง	✔	
การลบระเบียนที่ถูกระเบียนอื่นอ้างถึง	✔	

5.8. การทดสอบกับแบบจำลองเอนทิตีและความสัมพันธ์ที่ไม่เป็นนอร์มอลฟอร์มลำดับที่สาม

ในการทดสอบการทำงานของเครื่องมือซอฟต์แวร์กับข้อมูลตารางความสัมพันธ์ที่ไม่ได้เป็นนอร์มอลฟอร์มลำดับที่สามแต่เป็นนอร์มอลฟอร์มลำดับที่หนึ่งเป็นอย่างน้อย พบว่าเครื่องมือซอฟต์แวร์สามารถสร้างสัญกรณ์เซตในส่วนของโครงสร้างของเอนทิตี ได้แก่เค้าร่างเอนทิตี เค้าร่างเพิ่มเติม และเค้าร่างความสัมพันธ์ และส่วนคำสั่งพื้นฐานสำหรับเอนทิตีได้แก่เค้าร่างการเพิ่ม เค้าร่างการแก้ไข และเค้าร่างการลบได้ถูกต้อง ทั้งนี้ได้ทดสอบกับตัวอย่างข้อมูลที่ไม่เป็นนอร์มอลฟอร์มลำดับที่สามที่ซึ่งมีพจนานุกรมข้อมูลดังรูปที่ 5.10

ชื่อเอนทิตี	ชื่อลักษณะประจำ	ชนิดของลักษณะประจำ	คีย์	เงื่อนไข
"class"	"course"	"String"	"PK"	""
"class"	"stuid"	"String"	"PK"	""
"class"	"stuname"	"String"	""	""
"class"	"facId"	"String"	""	""
"class"	"sched"	"String"	""	""
"class"	"room"	"String"	""	""
"class"	"grade"	"String"	""	""

รูปที่ 5.10 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูล

จากตัวอย่างเป็นพจนานุกรมข้อมูลของเอนทิตี class ซึ่งมี course และ stuid เป็นคีย์หลัก โดยที่ stuname ขึ้นกับบางส่วนของคีย์หลักคือ stuid จึงทำให้ไม่เป็นนอร์มอลฟอร์มลำดับที่สาม เมื่อนำข้อมูลนี้มาทำการสร้างสัญกรณ์เซดด้วยเครื่องมือซอฟต์แวร์ที่ได้พัฒนาขึ้น ได้ผลลัพธ์สัญกรณ์เซดดังรูปที่ 5.11

```
\begin{schema} {class}
course : String \also
stuid : String \also
stuname : String \also
facId : String \also
sched : String \also
room : String \also
grade : String \also
\end{schema}

\begin{schema} {classExt}
classSet : \finset class \also
\where
\forall class1,class2 : class | class1 \in classSet \land class2 \in classSet \land class1 \neq class2 @
class1.stuid \neq class2.stuid
 \lor class1.course \neq class2.course \also
\end{schema}

\begin{schema} {Insertclass}
\Delta classExt \also
newValue? : class
\where
classSet' = classSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updateclass}
classExt \also
uclass? : class \also
tclass, tclass' : class \also
\where
tclass \in classSet \also
tclass.course = uclass?.course \also
tclass.stuid = uclass?.stuid \also
tclass'.grade = uclass?.grade \also
tclass'.room = uclass?.room \also
tclass'.sched = uclass?.sched \also
tclass'.facId = uclass?.facId \also
tclass'.stuname = uclass?.stuname \also
\end{schema}

\begin{schema} {Delclass}
dclass? : class \also
\Delta classExt \also
\where
classSet' = classSet \setminus \{ dclass? \} \also
\end{schema}
```

รูปที่ 5.11 รูปแสดงผลลัพธ์สัญกรณ์เซดของตารางความสัมพันธ์ที่ไม่ได้เป็นนอร์มอลฟอร์มลำดับที่สาม

อย่างไรก็ดีการใช้ตารางความสัมพันธ์ที่ไม่ได้เป็นนอร์มอลฟอร์มลำดับที่สามจะทำให้เกิดความซ้ำซ้อนของข้อมูล นำมาซึ่งความไม่สอดคล้องกันของข้อมูลในกรณีที่มีการแก้ไขข้อมูล การสูญเสียข้อมูลบางอย่างจากการลบข้อมูล และการไม่สามารถเพิ่มเติมข้อมูลแต่เพียงบางส่วนได้

บทที่ 6
สรุปผลการวิจัย

จากการศึกษา และ วิจัยสามารถสรุปผลที่ได้รับจากการวิจัย ปัญหา และข้อจำกัดที่พบ ได้ดังนี้

6.1. สรุปผลการวิจัย

จากการวิจัย ได้ออกแบบขั้นตอนวิธีที่ใช้ในการแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซด โดยข้อมูลจากแบบจำลองเอนทิตีและความสัมพันธ์จะอยู่ในรูปของพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ ทั้งนี้สามารถออกแบบขั้นตอนวิธีได้ 6 ขั้นตอนวิธี ได้แก่

1) ขั้นตอนวิธีสร้างเค้าร่างเอนทิตี เป็นส่วนการนิยามโครงสร้างของเอนทิตีว่าประกอบไปด้วยลักษณะประจำใดบ้าง
2) ขั้นตอนวิธีสร้างเค้าร่างเพิ่มเติม เป็นส่วนนิยามคุณสมบัติต่าง ๆ เพิ่มเติมจากคุณสมบัติที่มีอยู่ในเค้าร่างเอนทิตี เช่นการนิยามคีย์ เป็นต้น
3) ขั้นตอนวิธีสร้างเค้าร่างความสัมพันธ์ เป็นส่วนนิยามถึงความสัมพันธ์ระหว่าง 2 เอนทิตีใด ๆ ว่ามีความสัมพันธ์ต่อกันแบบ หนึ่งต่อหนึ่ง หนึ่งต่อหลาย หรือ หลายต่อหนึ่ง
4) ขั้นตอนวิธีสร้างเค้าร่างการเพิ่ม เป็นส่วนนิยามคำสั่งในการเพิ่มระเบียนข้อมูลเข้าไปในข้อมูลที่มีอยู่เดิม
5) ขั้นตอนวิธีสร้างเค้าร่างการแก้ไข เป็นส่วนนิยามคำสั่งในการแก้ไขข้อมูลในระเบียนใดระเบียนหนึ่งของข้อมูลที่มีอยู่
6) ขั้นตอนวิธีสร้างเค้าร่างการลบ เป็นส่วนนิยามคำสั่งในการลบระเบียนออกจากข้อมูลทั้งหมด โดยในที่นี้ได้นิยามเป็นการลบแบบลดหลั่นในกรณีที่ระเบียนที่ต้องการจะลบถูกอ้างอิงจากระเบียนอื่น

นอกจากนี้ เนื่องจากสัญกรณ์เซดมีชนิดข้อมูลเพียงชนิดเดียวคือ ชนิดจำนวนเต็ม ผู้วิจัยจึงได้นิยามชนิดของข้อมูลที่ต้องใช้ในการนิยามเค้าร่างเอนทิตีเพิ่มเติม เพื่อให้สัญกรณ์เซดที่ได้มีความสมบูรณ์ เอื้อต่อการทดสอบด้วย Z/EVES โดยชนิดข้อมูลที่ได้นิยามเพิ่มเติม คือ ชนิดข้อมูลตัวอักษร ชนิดข้อมูลสายอักษร และชนิดข้อมูลบูลีน

ในการทดสอบ ได้ใช้ Z/EVES ในการทดสอบทางด้านไวยากรณ์ของสัญกรณ์เซดที่ได้จากการใช้เครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซดในการแปลงข้อมูลจากกรณีทดสอบ 2 กรณี คือระบบให้เช่ารถแวน และระบบการขายของผ่านโทรศัพท์ โดยเครื่องมือนี้สามารถแปลงข้อมูลจากพจนานุกรมข้อมูล และพจนานุกรมความสัมพันธ์ที่ได้จากทั้ง 2 ระบบนี้ ให้เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซดได้ และผลลัพธ์สัญกรณ์เซดที่ได้จากการแปลงเมื่อนำไปทดสอบด้วย

Z/EVES ปรากฏว่า Z/EVES สามารถใช้สัญกรณ์เซตที่ได้สร้างเป็นทฤษฎีบทได้ โดยที่ไม่ปรากฏข้อผิดพลาดใด ๆ

ทั้งนี้วิธีการแปลงที่ใช้ในวิทยานิพนธ์นี้ เป็นวิธีที่ออกแบบเพื่อใช้กับงานทางด้านฐานข้อมูลโดยเฉพาะ กล่าวคือมีการคำนึงถึงคีย์หลัก คีย์นอก และบูรณภาพการอ้างอิง ซึ่งทำให้สัญกรณ์เซตที่ได้มีความสมบูรณ์ขึ้น

6.2. ประโยชน์ของเครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์เป็นข้อกำหนดรูปนัยในรูปสัญกรณ์เซต

1) เป็นเครื่องมือสำหรับช่วยในการทำข้อกำหนดรูปนัยในรูปสัญกรณ์เซตสำหรับผู้เริ่มต้นเนื่องจากสัญกรณ์เซตจะเขียนโดยใช้รูปแบบทางคณิตศาสตร์ทำให้ผู้ที่ไม่เคยเขียน หรือ มีความรู้ในด้านนี้มาก่อนต้องทำการศึกษาใหม่หมด ซึ่งต้องใช้เวลาในการศึกษามาก ดังนั้นเครื่องมือนี้จะเป็นการช่วยในการเขียนเบื้องต้น ทำให้ผู้ที่ริเริ่มในการเขียนสามารถศึกษาสัญกรณ์เซตอย่างค่อยเป็นค่อยไป
2) ช่วยลดเวลาในการเขียนสัญกรณ์เซต เนื่องจากสามารถนิยามชุดคำสั่งพื้นฐาน เพิ่ม ลบ และแก้ไขสำหรับเอนทิตีต่าง ๆ ให้ จึงทำให้ผู้พัฒนาไม่ต้องเสียเวลาในการเขียนข้อมูลส่วนนี้เอง แต่อาจต้องมีการเปลี่ยนแปลง หรือเพิ่มเติมบางส่วนเพื่อให้เหมาะสมกับแต่ละระบบให้โดยอัตโนมัติ
3) สามารถนำแบบจำลองเอนทิตีและความสัมพันธ์ที่เคยทำไว้มาสร้างให้อยู่ในรูปของสัญกรณ์เซตได้ เพื่อสามารถนำไปแปลงเป็นภาษาอื่นทำให้เกิดการนำกลับมาใช้ใหม่ (Reusable)

6.3. ปัญหา และข้อจำกัดที่ได้พบจากการวิจัย

1) ข้อมูลนำเข้าจะต้องอยู่ในรูปแบบตามที่กำหนด จึงจะสามารถสร้างสัญกรณ์เซตได้โดยสมบูรณ์
2) ข้อมูลนำเข้าจะต้องอยู่ในลำดับที่เหมาะสมกับการอ้างอิง คือ ถ้าหากเอนทิตีใดถูกอ้างอิงโดยเอนทิตีอื่น เอนทิตีนั้น ก็จะต้องถูกสร้างก่อนที่จะถูกอ้างอิง
3) ความสัมพันธ์ของเอนทิตีต้องอยู่ในรูปนอร์มอลฟอร์มลำดับที่สาม และเป็นความสัมพันธ์ทวิภาค เฉพาะความสัมพันธ์แบบหลายต่อหนึ่ง หนึ่งต่อหลาย และ หนึ่งต่อหนึ่ง
4) ในการทดสอบ จำเป็นจะต้องใช้ข้อมูลแบบจำลองเอนทิตี และความสัมพันธ์มาทำการทดสอบ ซึ่งหาข้อมูลแบบจำลองเอนทิตี และความสัมพันธ์ที่มีจำนวนเอนทิตีมากกว่า 20 เอนทิตีได้ยาก
5) สัญกรณ์เซตมีชนิดข้อมูลพื้นฐานเพียงชนิดเดียว คือ จำนวนเต็ม ดังนั้น ถ้าหากต้องการชนิดข้อมูลอื่นเช่น ชนิดตัวอักษร จะต้องทำการนิยามชนิดข้อมูลนั้นขึ้นมาใหม่

รายการอ้างอิง

[1] Bowen, J. P. "Formal Specification in Z as a Design and Documentation Tool" Second Software Engineering IEEE/BCS Conference. (1988) : 164-168.

[2] Jin, L. and Zhu, H. "Automatic Generation of Formal Specification from Requirements Definition" First IEEE International Conference. (1997) : 243-251.

[3] Ledru, Y. "Complementing Semi-Formal Specifications with Z" Proceeding of KBSE. (1996) : 52-61.

[4] Sommerville, I. Software Engineering. Fourth Edition. London :Addison Wesley, 1992, pp. 7-9.

[5] Elmasri, R. and Navathe, S. B. Fundamentals of Database Systems. Second Edition California : Benjamin/Cummings 1994, pp. 39-68.

[6] Hawryszkiewycz, I. T. Database Analysis and Design. Second Edition. New York : Maxwel Macmillan, 1991, pp. 111-142.

[7] Fleming C.C. Handbook of Relational Database Design. (n.p.) : Addison-Wesley 1989, pp.179-189

[8] Ratcliff, B. Introducing Specification Using Z : A Practical Case Study Approach. England : McGraw-Hill, 1994.

[9] Bowen, J. Formal Specification and Documentation Using Z: A Case Study Approach. (n.p.) : International Thomson Publishing Company 1996.

[10] Fraser, M. D., Kumar, K. and Vaishnavi, V. K. "Informal and Formal Requirements Specification Languages: Bridging the Gap" TSE. Vol17 No. 5 (MAY 1991) : 454-466.

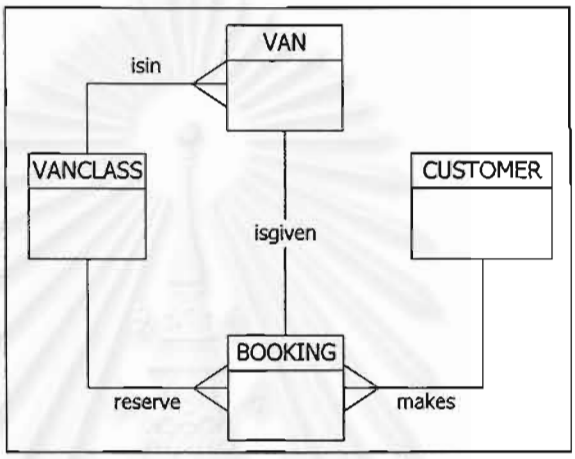
[11] Alexander, P. "Combining formal and semi-formal methods in software engineering" Software Engineering. (December'95/January'96) : 29-32.

[12] Kolman, B. and Busby, R. C. Discrete Mathematical Structures For Computer Science. Second Edition Englewood Cliffs N.J. : Prentice Hall, 1987.

ภาคผนวก

ภาคผนวก ก. ระบบให้เช่ารถแวน

ระบบให้เช่ารถแวน เป็นระบบสำหรับอู่ที่ให้บริการเช่ารถแวน จะคล้ายกับระบบให้เช่ารถยนต์ทั่วไป แต่จะแตกต่างกันตรงที่รถที่ให้เช่ามีเฉพาะรถแวนโดยสามารถแบ่งประเภทของรถแวนได้เป็นประเภทต่าง ๆ เช่น มินิแวน แวนสำหรับขนส่ง และแวนสำหรับงานหนัก เป็นต้น โดยในการเช่ารถแวนลูกค้าจะทำการจองโดยการระบุชนิดของรถแวนที่ต้องการเช่าล่วงหน้า และเมื่อมารับรถทางอู่จะจัดหารถแวนตามชนิดที่จองไว้ให้บริการ จากระบบรถแวน นี้ สามารถเขียนเป็นแผนภาพเอนทิตี และความสัมพันธ์ได้ดังรูปที่ ก-1



รูปที่ ก-1 แผนภาพเอนทิตี และความสัมพันธ์ของระบบให้เช่ารถแวน

ซึ่งข้อมูลของพจนานุกรมข้อมูลอยู่ในรูปของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลดังรูป ก-2

ชื่อเอนทิตี	ชื่อลักษณะประจำ	ชนิดลักษณะประจำ	คีย์	เงื่อนไข
"vanclass"	"classname"	"String"	"PK"	""
"vanclass"	"fueltype"	"Fueltype"	""	""
"van"	"reg"	"String"	"PK"	""
"van"	"model"	"String"	""	""
"van"	"class"	"String"	"FK(vanclass-classname)"	""
"customer"	"custId"	"String"	"PK"	""
"customer"	"custName"	"String"	""	""
"customer"	"custAddr"	"String"	""	""
"booking"	"bookId"	"String"	"PK"	""
"booking"	"startDate"	"Date"	""	""
"booking"	"endDate"	"Date"	""	""
"booking"	"custId"	"String"	"FK(customer-custId)"	""
"booking"	"class"	"String"	"FK(vanclass-classname)"	""
"booking"	"vanNo"	"String"	"FK(van-reg)"	""

รูปที่ ก-2 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลของระบบให้เช่ารถแวน

และข้อมูลของพจนานุกรมความสัมพันธ์อยู่ในรูปของแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ดังรูปที่ ก-3

ชื่อความสัมพันธ์	เอนทิตี1	เอนทิตี2	ลักษณะความสัมพันธ์	เงื่อนไข
"isin"	"van"	"vanclass"	"m-1"	""
"reserve"	"booking"	"vanclass"	"m-1"	""
"makes"	"customer"	"booking"	"1-m"	""
"isgiven"	"booking"	"van"	"1-1"	""

รูปที่ ก-3 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ของระบบให้เช่ารถแวน

ผลจากการใช้เครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์ซึ่งมีจำนวนเอนทิตี 4 เอนทิตี สามารถสร้างเป็นสัญกรณ์เซดโดยอยู่ในรูปของเค้าร่างต่าง ๆ ได้ทั้งหมด 24 เค้าร่าง โดยแบ่งเป็น เค้าร่างเอนทิตี 4 เค้าร่าง เค้าร่างเพิ่มเติม 4 เค้าร่าง เค้าร่างความสัมพันธ์ 4 เค้าร่าง เค้าร่างการเพิ่ม 4 เค้าร่าง เค้าร่างการแก้ไข 4 เค้าร่าง และ เค้าร่างการลบ 4 เค้าร่าง ซึ่งมีผลลัพธ์เป็นดังรูปที่ ก-4

```
\begin{zed}
Char == 0 \upto 127 \also
A == 65 \also
B == 66 \also
C == 67 \also
D == 68 \also
E == 69 \also
F == 70 \also
G == 71 \also
H == 72 \also
I == 73 \also
J == 74 \also
K == 75 \also
L == 76 \also
M == 77 \also
N == 78 \also
O == 79 \also
P == 80 \also
Q == 81 \also
R == 82 \also
S == 83 \also
T == 84 \also
U == 85 \also
V == 86 \also
W == 87 \also
X == 88 \also
Y == 89 \also
Z == 90 \also
a == 97 \also
```

รูปที่ ก-4 ผลลัพธ์สัญกรณ์เซดของระบบให้เช่ารถแวน

```
b == 98 \also
c == 99 \also
d == 100 \also
e == 101 \also
f == 102 \also
g == 103 \also
h == 104 \also
i == 105 \also
j == 106 \also
k == 107 \also
l == 108 \also
m == 109 \also
n == 110 \also
o == 111 \also
p == 112 \also
q == 113 \also
r == 114 \also
s == 115 \also
t == 116 \also
u == 117 \also
v == 118 \also
w == 119 \also
x == 120 \also
y == 121 \also
z == 122 \also
\end{zed}

\begin{zed}
String == \seq Char \also
\end{zed}

\begin{zed}
Boolean ::= True|False \also
\end{zed}

\begin{zed}
[Date,Fueltype,string]
\end{zed}

\begin{schema}{vanclass}
classname : string \also
fueltype : Fueltype \also
\end{schema}

\begin{schema}{van}
reg : string \also
model : string \also
class : string \also
\end{schema}

\begin{schema}{customer}
custId : string \also
custName : string \also
custAddr : string \also
\end{schema}
\begin{schema}{booking}
```

รูปที่ ก-4 ผลลัพธ์สัญกรณ์เซดของระบบให้เช่ารถแวน (ต่อ)

```
bookId : string \also
startDate : Date \also
endDate : Date \also
custId : string \also
class : string \also
vanNo : string \also
\end{schema}

\begin{schema} {vanclassExt}
vanclassSet : \finset vanclass \also
\where
\forall vanclass1,vanclass2 : vanclass | vanclass1 \in vanclassSet \land vanclass2 \in vanclassSet \land
vanclass1 \neq vanclass2 @ vanclass1.classname \neq vanclass2.classname \also
\end{schema}

\begin{schema} {vanExt}
vanSet : \finset van \also
\where
\forall van1,van2 : van | van1 \in vanSet \land van2 \in vanSet \land van1 \neq van2 @ van1.reg \neq
van2.reg \also
\forall vanVar : van @( \exists vanclassVar : vanclass @ vanVar.class = vanclassVar.classname) \also
\end{schema}

\begin{schema} {customerExt}
customerSet : \finset customer \also
\where
\forall customer1,customer2 : customer | customer1 \in customerSet \land customer2 \in customerSet
\land customer1 \neq customer2 @ customer1.custId \neq customer2.custId \also
\end{schema}

\begin{schema} {bookingExt}
bookingSet : \finset booking \also
\where
\forall booking1,booking2 : booking | booking1 \in bookingSet \land booking2 \in bookingSet \land
booking1 \neq booking2 @ booking1.bookId \neq booking2.bookId \also
\forall bookingVar : booking @( \exists vanVar : van @ bookingVar.vanNo = vanVar.reg) \also
\forall bookingVar : booking @( \exists vanclassVar : vanclass @ bookingVar.class =
vanclassVar.classname) \also
\forall bookingVar : booking @( \exists customerVar : customer @ bookingVar.custId =
customerVar.custId) \also
\end{schema}

\begin{schema} {Insertbooking}
\Delta bookingExt \also
newValue? : booking
\where
bookingSet' = bookingSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatebooking}
bookingExt \also
ubooking? : booking \also
tbooking, tbooking' : booking \also
\where
tbooking \in bookingSet \also
tbooking.bookId = ubooking?.bookId \also
```

รูปที่ ก-4 ผลลัพธ์สัญกรณ์เซดของระบบให้เช่ารถแวน (ต่อ)

```
tbooking'.vanNo = ubooking?.vanNo \also
tbooking'.class = ubooking?.class \also
tbooking'.custId = ubooking?.custId \also
tbooking'.endDate = ubooking?.endDate \also
tbooking'.startDate = ubooking?.startDate \also
\end{schema}

\begin{schema} {Delbooking}
dbooking? : booking \also
\Delta bookingExt \also
\where
bookingSet' = bookingSet \setminus \{ dbooking? \} \also
\end{schema}

\begin{schema} {Insertcustomer}
\Delta customerExt \also
newValue? : customer
\where
customerSet' = customerSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatecustomer}
customerExt \also
ucustomer? : customer \also
tcustomer, tcustomer' : customer \also
\where
tcustomer \in customerSet \also
tcustomer.custId = ucustomer?.custId \also
tcustomer'.custAddr = ucustomer?.custAddr \also
tcustomer'.custName = ucustomer?.custName \also
\end{schema}

\begin{schema} {Delcustomer}
dcustomer? : customer \also
\Delta customerExt \also
Delbooking \also
\where
dbooking?.custId = dcustomer?.custId \also
customerSet' = customerSet \setminus \{ dcustomer? \} \also
\end{schema}

\begin{schema} {Insertvan}
\Delta vanExt \also
newValue? : van
\where
vanSet' = vanSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatevan}
vanExt \also
uvan? : van \also
tvan, tvan' : van \also
\where
tvan \in vanSet \also
tvan.reg = uvan?.reg \also
tvan'.class = uvan?.class \also
```

รูปที่ ก-4 ผลลัพธ์สัญกรณ์เซตของระบบให้เช่ารถแวน (ต่อ)

```
tvan'.model = uvan?.model \also
\end{schema}

\begin{schema} {Delvan}
dvan? : van \also
\Delta vanExt \also
Delbooking \also
\where
dbooking?.vanNo = dvan?.reg \also
vanSet' = vanSet \setminus \{ dvan? \} \also
\end{schema}

\begin{schema} {Insertvanclass}
\Delta vanclassExt \also
newValue? : vanclass
\where
vanclassSet' = vanclassSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatevanclass}
vanclassExt \also
uvanclass? : vanclass \also
tvanclass, tvanclass' : vanclass \also
\where
tvanclass \in vanclassSet \also
tvanclass.classname = uvanclass?.classname \also
tvanclass'.fueltype = uvanclass?.fueltype \also
\end{schema}

\begin{schema} {Delvanclass}
dvanclass? : vanclass \also
\Delta vanclassExt \also
Delbooking \also
Delvan \also
\where
dbooking?.class = dvanclass?.classname \also
dvan?.class = dvanclass?.classname \also
vanclassSet' = vanclassSet \setminus \{ dvanclass? \} \also
\end{schema}

\begin{schema} {Relationshipisin}
isin : \finset(van \cross vanclass)
\where
\forall isin1,isin2 : isin @ first~isin1 = first~isin2 \implies second~isin1 = second~isin2 \also
\end{schema}

\begin{schema} {Relationshipreserve}
reserve : \finset(booking \cross vanclass)
\where
\forall reserve1,reserve2 : reserve @ first~reserve1 = first~reserve2 \implies second~reserve1 =
second~reserve2 \also
\end{schema}
\begin{schema} {Relationshipmakes}
makes : \finset(customer \cross booking)
\where
```

รูปที่ ก-4 ผลลัพธ์สัญกรณ์เซดของระบบให้เช่ารถแวน (ต่อ)

```
\forall makes1,makes2 : makes @ second~makes1 = second~makes2 \implies first~makes1 =
first~makes2 \also
\end{schema}

\begin{schema} {Relationshipisgiven}
isgiven : \finset(booking \cross van)
\where
\forall isgiven1,isgiven2 : isgiven @ second~isgiven1 = second~isgiven2 \implies first~isgiven1 =
first~isgiven2 \land first~isgiven1 = first~isgiven2 \implies second~isgiven1 = second~isgiven2 \also
\end{schema}
```

รูปที่ ก-4 ผลลัพธ์สัญกรณ์เซตของระบบให้เช่ารถแวน (ต่อ)

และจากการนำข้อกำหนดสัญกรณ์เซตที่เป็นผลลัพธ์มาทำการตรวจสอบความถูกต้องทางด้านไวยากรณ์ด้วย Z/EVES ได้ผลลัพธ์ดังรูปที่ ก-5

```
Allegro CL for Windows Copyright (C) 1992-1997, Franz Inc., Berkeley, CA, USA.
All rights reserved.
=>
Reading file E:\MAY\test.zed
definition of Char
definition of A
definition of B
definition of C
definition of D
definition of E
definition of F
definition of G
definition of H
definition of I
definition of J
definition of K
definition of L
definition of M
definition of N
definition of O
definition of P
definition of Q
definition of R
definition of S
definition of T
definition of U
definition of V
definition of W
definition of X
definition of Y
definition of Z
definition of a
definition of b
definition of c
definition of d
definition of e
definition of f
```

รูปที่ ก-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบให้เช่ารถแวนด้วย Z/EVES

```
definition of g
definition of h
definition of i
definition of j
definition of k
definition of l
definition of m
definition of n
definition of o
definition of p
definition of q
definition of r
definition of s
definition of t
definition of u
definition of v
definition of w
definition of x
definition of y
definition of z
definition of String
free type Boolean
... theorem True\$declaration
... theorem False\$declaration
... theorem Boolean\$expansion
... theorem Boolean\$member
... theorem Boolean\$induction
given sets Date, Fueltype, string
schema vanclass
... theorem vanclass\$declarationPart
schema van
... theorem van\$declarationPart
schema customer
... theorem customer\$declarationPart
schema booking
... theorem booking\$declarationPart
schema vanclassExt
... theorem vanclass\$thetasEqual
... theorem vanclass\$inSet
... theorem vanclass\$thetaInSet
... theorem vanclass\$setInPowerSet
... theorem vanclass\$member
... theorem vanclass\$thetaMember
... theorem vanclass\$declaration
... theorem vanclass\$select\$classname
... theorem vanclass\$select\$fueltype
... theorem vanclassExt\$declarationPart
schema vanExt
... theorem van\$thetasEqual
... theorem van\$inSet
... theorem van\$thetaInSet
... theorem van\$setInPowerSet
... theorem van\$member
... theorem van\$thetaMember
... theorem van\$declaration
... theorem van\$select\$class
```

รูปที่ ก-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบให้เช่ารถแวนด้วย Z/EVES (ต่อ)

```
... theorem van\$select\$model
... theorem van\$select\$reg
... theorem vanExt\$declarationPart
schema customerExt
... theorem customer\$thetasEqual
... theorem customer\$inSet
... theorem customer\$thetaInSet
... theorem customer\$setInPowerSet
... theorem customer\$member
... theorem customer\$thetaMember
... theorem customer\$declaration
... theorem customer\$select\$custAddr
... theorem customer\$select\$custId
... theorem customer\$select\$custName
... theorem customerExt\$declarationPart
schema bookingExt
... theorem booking\$thetasEqual
... theorem booking\$inSet
... theorem booking\$thetaInSet
... theorem booking\$setInPowerSet
... theorem booking\$member
... theorem booking\$thetaMember
... theorem booking\$declaration
... theorem booking\$select\$bookId
... theorem booking\$select\$class
... theorem booking\$select\$custId
... theorem booking\$select\$endDate
... theorem booking\$select\$startDate
... theorem booking\$select\$vanNo
... theorem bookingExt\$declarationPart
schema Insertbooking
... schema \Delta bookingExt
... theorem Delta\$bookingExt\$declarationPart
... theorem Insertbooking\$declarationPart
schema Updatebooking
... theorem Updatebooking\$declarationPart
schema Delbooking
... theorem Delbooking\$declarationPart
schema Insertcustomer
... schema \Delta customerExt
... theorem Delta\$customerExt\$declarationPart
... theorem Insertcustomer\$declarationPart
schema Updatecustomer
... theorem Updatecustomer\$declarationPart
schema Delcustomer
... theorem Delcustomer\$declarationPart
schema Insertvan
... schema \Delta vanExt
... theorem Delta\$vanExt\$declarationPart
... theorem Insertvan\$declarationPart
schema Updatevan
... theorem Updatevan\$declarationPart
schema Delvan
... theorem Delvan\$declarationPart
schema Insertvanclass
... schema \Delta vanclassExt
```

รูปที่ ก-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบให้เช่ารถแวนด้วย Z/EVES (ต่อ)

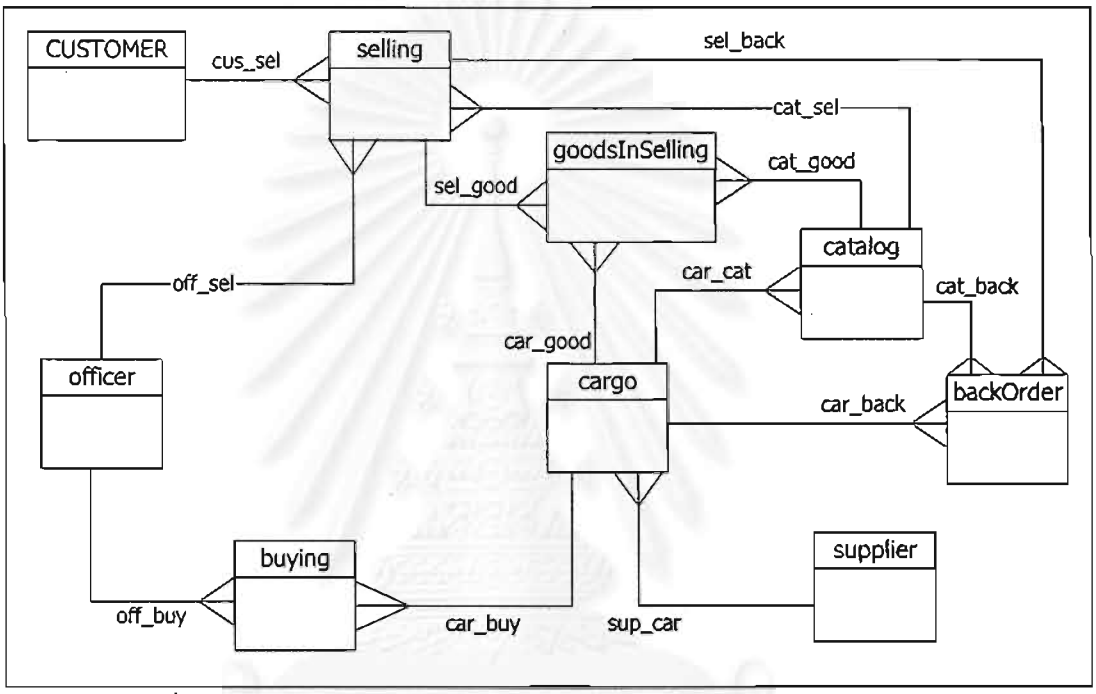
```
... theorem Delta\$vanclassExt\$declarationPart
... theorem Insertvanclass\$declarationPart
schema Updatevanclass
... theorem Updatevanclass\$declarationPart
schema Delvanclass
... theorem Delvanclass\$declarationPart
schema Relationshipisin
... theorem Relationshipisin\$declarationPart
schema Relationshipreserve
... theorem Relationshipreserve\$declarationPart
schema Relationshipmakes
... theorem Relationshipmakes\$declarationPart
schema Relationshipisgiven
... theorem Relationshipisgiven\$declarationPart
Done.
=>
```

รูปที่ ก-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบให้เช่ารถแวนด้วย Z/EVES (ต่อ)

ในการตรวจสอบไวยากรณ์ของ Z/EVES จะทำการตรวจสอบความถูกต้องของไวยากรณ์และถ้าหากไวยากรณ์มีความถูกต้อง Z/EVES จะสร้างทฤษฎีบทเกี่ยวกับเค้าร่างนั้นขึ้นมา จากผลลัพธ์ แสดงว่าสัญกรณ์เซตที่ได้มีความถูกต้องทางด้านไวยากรณ์ เนื่องจากสามารถสร้างเป็นทฤษฎีบทได้หมด ไม่มีการแสดงข้อผิดพลาดใด ๆ

ภาคผนวก ข. ระบบการขายของผ่านโทรศัพท์

ระบบการขายของผ่านโทรศัพท์ เป็นระบบการให้บริการซื้อของผ่านทางโทรศัพท์ โดยลูกค้าที่จะซื้อได้จะต้องถือบัตรเครดิตของธนาคาร ABC ซึ่งเมื่อมีการใช้จ่ายผ่านบัตรเครดิต ลูกค้าจะได้รับคะแนนสะสม ซึ่งสามารถใช้คะแนนสะสมนั้นมาใช้ในการแลกซื้อสินค้าที่ธนาคารทำการกำหนดไว้ โดยสินค้าที่จะแลกซื้อได้นั้นจะจัดทำเป็นรายการสินค้าสำหรับแต่ละเดือน นอกจากนี้ ถ้าหากสินค้าที่ลูกค้าต้องการซื้อหมด ทางบริษัทจะรับการสั่งซื้อสินค้านั้นๆไว้ และจะทำการจัดส่งสินค้าให้ภายหลัง จากระบบการขายผ่านโทรศัพท์นี้ สามารถเขียนเป็นแผนภาพเอนทิตี และความสัมพันธ์ได้ดังรูป ข-1



รูปที่ ข-1 แผนภาพเอนทิตี และความสัมพันธ์ของระบบการขายของผ่านโทรศัพท์

ซึ่งข้อมูลของพจนานุกรมข้อมูลอยู่ในรูปของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลดังรูป ข-2

ชื่อเอนทิตี	ชื่อลักษณะประจำ	ชนิดลักษณะประจำ	คีย์	เงื่อนไข
"customer"	"creditNo"	"String"	"PK"	""
"customer"	"customerName"	"String"	""	""
"customer"	"customAdd"	"String"	""	""
"customer"	"creditExpire"	"String"	""	""
"customer"	"point"	"Int"	""	"point \geq 0"
"customer"	"paid"	"float"	""	""
"customer"	"customTel"	"String"	""	""

รูปที่ ข-2 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลของระบบการขายของผ่านโทรศัพท์

"customer"	"customTax"	"String"	""	""
"customer"	"customFlag"	"Boolean"	""	""
"supplier"	"supplierId"	"String"	"PK"	""
"supplier"	"supplierName"	"String"	""	""
"supplier"	"supplierAdd"	"String"	""	""
"supplier"	"supplierTel"	"String"	""	""
"supplier"	"supplierFax"	"String"	""	""
"supplier"	"supplierJoin"	"String"	""	""
"cargo"	"goodsId"	"String"	"PK"	""
"cargo"	"goodsName"	"String"	""	""
"cargo"	"goodsRemain"	"String"	""	""
"cargo"	"supplierId"	"String"	"FK(supplier-supplierId)"	""
"cargo"	"saftyStock"	"String"	""	""
"cargo"	"goodsStatus"	"String"	""	""
"cargo"	"goodsComment"	"String"	""	""
"cargo"	"onHandGoods"	"Int"	""	"onHandGoods \geq 0"
"catalog"	"catalogId"	"String"	"PK"	""
"catalog"	"goodsId"	"String"	"PK,FK(cargo-goodsId)"	""
"catalog"	"point"	"Int"	""	"point \geq 0"
"catalog"	"price"	"float"	""	""
"catalog"	"pointRef"	"Int"	""	"pointRef \geq 0"
"catalog"	"priceRef"	"float"	""	""
"officer"	"officerId"	"String"	"PK"	""
"officer"	"officerName"	"String"	""	""
"officer"	"officerAdd"	"String"	""	""
"officer"	"officerTel"	"String"	""	""
"officer"	"secialNo"	"String"	""	""
"selling"	"sellId"	"String"	"PK"	""
"selling"	"credit"	"String"	"FK(customer-creditNo)"	""

รูปที่ ข-2 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลของระบบการขายของผ่านโทรศัพท์ (ต่อ)

"selling"	"receiveName"	"String"	""	""
"selling"	"receiveAdd"	"String"	""	""
"selling"	"receiveTel"	"String"	""	""
"selling"	"orderDate"	"String"	""	""
"selling"	"officerId"	"String"	"FK(officer-officerId)"	""
"selling"	"togetherFlag"	"Boolean"	""	""
"selling"	"catalogId"	"String"	"FK(catalog-catalogId)"	""
"goodsInSelling"	"sellId"	"String"	"PK,FK(selling-sellId)"	""
"goodsInSelling"	"goodsId"	"String"	"PK,FK(cargo-goodsId)"	""
"goodsInSelling"	"sellType"	"String"	""	""
"goodsInSelling"	"quantity"	"Int"	""	""
"goodsInSelling"	"sellPoint"	"Int"	""	""
"goodsInSelling"	"sellMoney"	"float"	""	""
"goodsInSelling"	"catalogId"	"String"	"FK(catalog-catalogId)"	""
"goodsInSelling"	"sentFlag"	"Boolean"	""	""
"goodsInSelling"	"sentDate"	"date"	""	""
"backOrder"	"sellId"	"String"	"PK,FK(selling-sellId)"	""
"backOrder"	"goodsId"	"String"	"PK,FK(cargo-goodsId)"	""
"backOrder"	"sellType"	"String"	""	""
"backOrder"	"quantity"	"Int"	""	"quantity \geq 0"
"backOrder"	"sellPoint"	"Int"	""	"sellPoint \geq 0"
"backOrder"	"sellMoney"	"float"	""	""
"backOrder"	"catalogId"	"String"	"FK(catalog-catalogId)"	""
"buying"	"orderId"	"String"	"PK"	""

รูปที่ ช-2 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลของระบบการขายของผ่านโทรศัพท์ (ต่อ)

"buying"	"goodsId"	"String"	"PK,FK(cargo-goodsId)"	""
"buying"	"buyQuantity"	"Int"	""	"buyQuantity \geq 0"
"buying"	"pricePerUnit"	"float"	""	""
"buying"	"orderDate"	"date"	""	""
"buying"	"sendDate"	"date"	""	""
"buying"	"orderOfficer"	"String"	""	""
"buying"	"receiveOfficer"	"String"	"FK(officer-officerId)"	""
"buying"	"slipNo"	"String"	""	""
"buying"	"orderComment"	"String"	""	""
"buying"	"receiveFlag"	"Boolean"	""	""

รูปที่ ข-2 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมข้อมูลของระบบการขายของผ่านโทรศัพท์ (ต่อ)

และข้อมูลของพจนานุกรมความสัมพันธ์อยู่ในรูปของแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ดังรูปที่ ข-3

ชื่อความสัมพันธ์	เอนทิตี1	เอนทิตี2	ลักษณะความสัมพันธ์	เงื่อนไข
"cus_sel"	"customer"	"selling"	"1-m"	""
"off_sel"	"officer"	"selling"	"1-m"	""
"off_buy"	"officer"	"buying"	"1-m"	""
"cat_sel"	"catalog"	"selling"	"1-m"	""
"cat_good"	"catalog"	"goodsInSelling"	"1-m"	""
"cat_back"	"catalog"	"backOrder"	"1-m"	""
"sel_good"	"goodsInSelling"	"selling"	"m-1"	""
"sel_back"	"backOrder"	"selling"	"m-1"	""
"car_back"	"backOrder"	"cargo"	"m-1"	""
"car_cat"	"catalog"	"cargo"	"m-1"	""
"sup_car"	"cargo"	"supplier"	"m-1"	""
"car_good"	"cargo"	"goodsInSelling"	"1-m"	""
"car_buy"	"cargo"	"buying"	"1-m"	""

รูปที่ ข-3 ข้อมูลของแฟ้มข้อความสำหรับพจนานุกรมความสัมพันธ์ของระบบการขายของผ่านโทรศัพท์

ผลจากการใช้เครื่องมือแปลงแบบจำลองเอนทิตีและความสัมพันธ์ซึ่งมีจำนวนเอนทิตี 9 เอนทิตี สามารถสร้างเป็นสัญกรณ์เซดโดยอยู่ในรูปของเค้าร่างต่าง ๆ ได้ทั้งหมด 58 เค้าร่าง โดยแบ่งเป็น เค้าร่างเอนทิตี 9 เค้าร่าง เค้าร่างเพิ่มเติม 9 เค้าร่าง เค้าร่างความสัมพันธ์ 13 เค้าร่าง เค้าร่างการเพิ่ม 9 เค้าร่าง เค้าร่างการแก้ไข 9 เค้าร่าง และ เค้าร่างการลบ 9 เค้าร่าง ซึ่งมีผลลัพธ์เป็นดังรูปที่ ข-4

```
\begin{zed}
Char == 0 \upto 127 \also
A == 65 \also
B == 66 \also
C == 67 \also
D == 68 \also
E == 69 \also
F == 70 \also
G == 71 \also
H == 72 \also
I == 73 \also
J == 74 \also
K == 75 \also
L == 76 \also
M == 77 \also
N == 78 \also
O == 79 \also
P == 80 \also
Q == 81 \also
R == 82 \also
S == 83 \also
T == 84 \also
U == 85 \also
V == 86 \also
W == 87 \also
X == 88 \also
Y == 89 \also
Z == 90 \also
a == 97 \also
b == 98 \also
c == 99 \also
d == 100 \also
e == 101 \also
f == 102 \also
g == 103 \also
h == 104 \also
i == 105 \also
j == 106 \also
k == 107 \also
l == 108 \also
m == 109 \also
n == 110 \also
o == 111 \also
p == 112 \also
q == 113 \also
r == 114 \also
s == 115 \also
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซดของระบบการขายของผ่านโทรศัพท์

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
t == 116 \also
u == 117 \also
v == 118 \also
w == 119 \also
x == 120 \also
y == 121 \also
z == 122 \also
\end{zed}

\begin{zed}
String == \seq Char \also
\end{zed}

\begin{zed}
Boolean ::= True|False \also
\end{zed}

\begin{zed}
[date,float]
\end{zed}

\begin{schema}{customer}
creditNo : String \also
customerName : String \also
customAdd : String \also
creditExpire : String \also
point : \num \also
paid : float \also
customTel : String \also
customTax : String \also
customFlag : Boolean \also
\where
point \geq 0 \also
\end{schema}

\begin{schema}{supplier}
supplierId : String \also
supplierName : String \also
supplierAdd : String \also
supplierTel : String \also
supplierFax : String \also
supplierJoin : String \also
\end{schema}

\begin{schema}{cargo}
goodsId : String \also
goodsName : String \also
goodsRemain : String \also
supplierId : String \also
saftyStock : String \also
goodsStatus : String \also
goodsComment : String \also
onHandGoods : \num \also
\where
onHandGoods \geq 0 \also
\end{schema}
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซดของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
\begin{schema} {catalog}
catalogId : String \also
goodsId : String \also
point : \num \also
price : float \also
pointRef : \num \also
priceRef : float \also
\where
point \geq 0 \also
pointRef \geq 0 \also
\end{schema}

\begin{schema} {officer}
officerId : String \also
officerName : String \also
officerAdd : String \also
officerTel : String \also
secialNo : String \also
\end{schema}

\begin{schema} {selling}
sellId : String \also
credit : String \also
receiveName : String \also
receiveAdd : String \also
receiveTel : String \also
orderDate : String \also
officerId : String \also
togetherFlag : Boolean \also
catalogId : String \also
\end{schema}

\begin{schema} {goodsInSelling}
sellId : String \also
goodsId : String \also
sellType : String \also
quantity : \num \also
sellPoint : \num \also
sellMoney : float \also
catalogId : String \also
sentFlag : Boolean \also
sentDate : date \also
\end{schema}

\begin{schema} {backOrder}
sellId : String \also
goodsId : String \also
sellType : String \also
quantity : \num \also
sellPoint : \num \also
sellMoney : float \also
catalogId : String \also
\where
quantity \geq 0 \also
sellPoint \geq 0 \also
\end{schema}
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซดของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
\begin{schema} {buying}
orderId : String \also
goodsId : String \also
buyQuantity : \num \also
pricePerUnit : float \also
orderDate : date \also
sendDate : date \also
orderOfficer : String \also
receiveOfficer : String \also
slirpNo : String \also
orderComment : String \also
receiveFlag : Boolean \also
\where
buyQuantity \geq 0 \also
\end{schema}

\begin{schema} {customerExt}
customerSet : \finset customer \also
\where
\forall customer1,customer2 : customer | customer1 \in customerSet \land customer2 \in customerSet
\land customer1 \neq customer2 @ customer1.creditNo \neq customer2.creditNo \also
\end{schema}

\begin{schema} {supplierExt}
supplierSet : \finset supplier \also
\where
\forall supplier1,supplier2 : supplier | supplier1 \in supplierSet \land supplier2 \in supplierSet \land
supplier1 \neq supplier2 @ supplier1.supplierId \neq supplier2.supplierId \also
\end{schema}

\begin{schema} {cargoExt}
cargoSet : \finset cargo \also
\where
\forall cargo1,cargo2 : cargo | cargo1 \in cargoSet \land cargo2 \in cargoSet \land cargo1 \neq cargo2 @
cargo1.goodsId \neq cargo2.goodsId \also
\forall cargoVar2 : cargo @( \exists supplierVar1 : supplier @ cargoVar2.supplierId =
supplierVar1.supplierId) \also
\end{schema}

\begin{schema} {catalogExt}
catalogSet : \finset catalog \also
\where
\forall catalog1,catalog2 : catalog | catalog1 \in catalogSet \land catalog2 \in catalogSet \land catalog1
\neq catalog2 @ catalog1.goodsId \neq catalog2.goodsId
 \lor catalog1.catalogId \neq catalog2.catalogId \also
\forall catalogVar2 : catalog @( \exists cargoVar1 : cargo @ catalogVar2.goodsId = cargoVar1.goodsId)
\also
\end{schema}

\begin{schema} {officerExt}
officerSet : \finset officer \also
\where
\forall officer1,officer2 : officer | officer1 \in officerSet \land officer2 \in officerSet \land officer1 \neq
officer2 @ officer1.officerId \neq officer2.officerId \also
\end{schema}
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซดของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
\begin{schema} {sellingExt}
sellingSet : \finset selling \also
\where
\forall selling1,selling2 : selling | selling1 \in sellingSet \land selling2 \in sellingSet \land selling1 \neq
selling2 @ selling1.sellId \neq selling2.sellId \also
\forall sellingVar2 : selling @( \exists catalogVar1 : catalog @ sellingVar2.catalogId =
catalogVar1.catalogId) \also
\forall sellingVar2 : selling @( \exists officerVar1 : officer @ sellingVar2.officerId =
officerVar1.officerId) \also
\forall sellingVar2 : selling @( \exists customerVar1 : customer @ sellingVar2.credit =
customerVar1.creditNo) \also
\end{schema}

\begin{schema} {goodsInSellingExt}
goodsInSellingSet : \finset goodsInSelling \also
\where
\forall goodsInSelling1,goodsInSelling2 : goodsInSelling | goodsInSelling1 \in goodsInSellingSet \land
goodsInSelling2 \in goodsInSellingSet \land goodsInSelling1 \neq goodsInSelling2 @
goodsInSelling1.goodsId \neq goodsInSelling2.goodsId
 \lor goodsInSelling1.sellId \neq goodsInSelling2.sellId \also
\forall goodsInSellingVar2 : goodsInSelling @( \exists catalogVar1 : catalog @
goodsInSellingVar2.catalogId = catalogVar1.catalogId) \also
\forall goodsInSellingVar2 : goodsInSelling @( \exists cargoVar1 : cargo @
goodsInSellingVar2.goodsId = cargoVar1.goodsId) \also
\forall goodsInSellingVar2 : goodsInSelling @( \exists sellingVar1 : selling @
goodsInSellingVar2.sellId = sellingVar1.sellId) \also
\end{schema}

\begin{schema} {backOrderExt}
backOrderSet : \finset backOrder \also
\where
\forall backOrder1,backOrder2 : backOrder | backOrder1 \in backOrderSet \land backOrder2 \in
backOrderSet \land backOrder1 \neq backOrder2 @ backOrder1.goodsId \neq backOrder2.goodsId
 \lor backOrder1.sellId \neq backOrder2.sellId \also
\forall backOrderVar2 : backOrder @( \exists catalogVar1 : catalog @ backOrderVar2.catalogId =
catalogVar1.catalogId) \also
\forall backOrderVar2 : backOrder @( \exists cargoVar1 : cargo @ backOrderVar2.goodsId =
cargoVar1.goodsId) \also
\forall backOrderVar2 : backOrder @( \exists sellingVar1 : selling @ backOrderVar2.sellId =
sellingVar1.sellId) \also
\end{schema}

\begin{schema} {buyingExt}
buyingSet : \finset buying \also
\where
\forall buying1,buying2 : buying | buying1 \in buyingSet \land buying2 \in buyingSet \land buying1 \neq
buying2 @ buying1.goodsId \neq buying2.goodsId
 \lor buying1.orderId \neq buying2.orderId \also
\forall buyingVar2 : buying @( \exists officerVar1 : officer @ buyingVar2.receiveOfficer =
officerVar1.officerId) \also
\forall buyingVar2 : buying @( \exists cargoVar1 : cargo @ buyingVar2.goodsId = cargoVar1.goodsId)
\also
\end{schema}

\begin{schema} {Insertbuying}
\Delta buyingExt \also
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
newValue? : buying
\where
buyingSet' = buyingSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatebuying}
buyingExt \also
ubuying? : buying \also
tbuying, tbuying' : buying \also
\where
tbuying \in buyingSet \also
tbuying.orderId = ubuying?.orderId \also
tbuying.goodsId = ubuying?.goodsId \also
tbuying'.receiveFlag = ubuying?.receiveFlag \also
tbuying'.orderComment = ubuying?.orderComment \also
tbuying'.slirpNo = ubuying?.slirpNo \also
tbuying'.receiveOfficer = ubuying?.receiveOfficer \also
tbuying'.orderOfficer = ubuying?.orderOfficer \also
tbuying'.sendDate = ubuying?.sendDate \also
tbuying'.orderDate = ubuying?.orderDate \also
tbuying'.pricePerUnit = ubuying?.pricePerUnit \also
tbuying'.buyQuantity = ubuying?.buyQuantity \also
\end{schema}

\begin{schema} {Delbuying}
dbuying? : buying \also
\Delta buyingExt \also
\where
buyingSet' = buyingSet \setminus \{ dbuying? \} \also
\end{schema}

\begin{schema} {InsertbackOrder}
\Delta backOrderExt \also
newValue? : backOrder
\where
backOrderSet' = backOrderSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {UpdatebackOrder}
backOrderExt \also
ubackOrder? : backOrder \also
tbackOrder, tbackOrder' : backOrder \also
\where
tbackOrder \in backOrderSet \also
tbackOrder.sellId = ubackOrder?.sellId \also
tbackOrder.goodsId = ubackOrder?.goodsId \also
tbackOrder'.catalogId = ubackOrder?.catalogId \also
tbackOrder'.sellMoney = ubackOrder?.sellMoney \also
tbackOrder'.sellPoint = ubackOrder?.sellPoint \also
tbackOrder'.quantity = ubackOrder?.quantity \also
tbackOrder'.sellType = ubackOrder?.sellType \also
\end{schema}

\begin{schema} {DelbackOrder}
dbackOrder? : backOrder \also
\Delta backOrderExt \also
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
\where
backOrderSet' = backOrderSet \setminus \{ dbackOrder? \} \also
\end{schema}

\begin{schema} {InsertgoodsInSelling}
\Delta goodsInSellingExt \also
newValue? : goodsInSelling
\where
goodsInSellingSet' = goodsInSellingSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {UpdategoodsInSelling}
goodsInSellingExt \also
ugoodsInSelling? : goodsInSelling \also
tgoodsInSelling, tgoodsInSelling' : goodsInSelling \also
\where
tgoodsInSelling \in goodsInSellingSet \also
tgoodsInSelling.sellId = ugoodsInSelling?.sellId \also
tgoodsInSelling.goodsId = ugoodsInSelling?.goodsId \also
tgoodsInSelling'.sentDate = ugoodsInSelling?.sentDate \also
tgoodsInSelling'.sentFlag = ugoodsInSelling?.sentFlag \also
tgoodsInSelling'.catalogId = ugoodsInSelling?.catalogId \also
tgoodsInSelling'.sellMoney = ugoodsInSelling?.sellMoney \also
tgoodsInSelling'.sellPoint = ugoodsInSelling?.sellPoint \also
tgoodsInSelling'.quantity = ugoodsInSelling?.quantity \also
tgoodsInSelling'.sellType = ugoodsInSelling?.sellType \also
\end{schema}

\begin{schema} {DelgoodsInSelling}
dgoodsInSelling? : goodsInSelling \also
\Delta goodsInSellingExt \also
\where
goodsInSellingSet' = goodsInSellingSet \setminus \{ dgoodsInSelling? \} \also
\end{schema}

\begin{schema} {Insertselling}
\Delta sellingExt \also
newValue? : selling
\where
sellingSet' = sellingSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updateselling}
sellingExt \also
uselling? : selling \also
tselling, tselling' : selling \also
\where
tselling \in sellingSet \also
tselling.sellId = uselling?.sellId \also
tselling'.catalogId = uselling?.catalogId \also
tselling'.togetherFlag = uselling?.togetherFlag \also
tselling'.officerId = uselling?.officerId \also
tselling'.orderDate = uselling?.orderDate \also
tselling'.receiveTel = uselling?.receiveTel \also
tselling'.receiveAdd = uselling?.receiveAdd \also
tselling'.receiveName = uselling?.receiveName \also
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
tselling'.credit = uselling?.credit \also
\end{schema}

\begin{schema} {Delselling}
dselling? : selling \also
\Delta sellingExt \also
DelbackOrder \also
DelgoodsInSelling \also
\where
dbackOrder?.sellId = dselling?.sellId \also
dgoodsInSelling?.sellId = dselling?.sellId \also
sellingSet' = sellingSet \setminus \{ dselling? \} \also
\end{schema}

\begin{schema} {Insertofficer}
\Delta officerExt \also
newValue? : officer
\where
officerSet' = officerSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updateofficer}
officerExt \also
uofficer? : officer \also
tofficer, tofficer' : officer \also
\where
tofficer \in officerSet \also
tofficer.officerId = uofficer?.officerId \also
tofficer'.secialNo = uofficer?.secialNo \also
tofficer'.officerTel = uofficer?.officerTel \also
tofficer'.officerAdd = uofficer?.officerAdd \also
tofficer'.officerName = uofficer?.officerName \also
\end{schema}

\begin{schema} {Delofficer}
dofficer? : officer \also
\Delta officerExt \also
Delbuying \also
Delselling \also
\where
dbuying?.receiveOfficer = dofficer?.officerId \also
dselling?.officerId = dofficer?.officerId \also
officerSet' = officerSet \setminus \{ dofficer? \} \also
\end{schema}

\begin{schema} {Insertcatalog}
\Delta catalogExt \also
newValue? : catalog
\where
catalogSet' = catalogSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatecatalog}
catalogExt \also
ucatalog? : catalog \also
tcatalog, tcatalog' : catalog \also
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
\where
tcatalog \in catalogSet \also
tcatalog.catalogId = ucatalog?.catalogId \also
tcatalog.goodsId = ucatalog?.goodsId \also
tcatalog'.priceRef = ucatalog?.priceRef \also
tcatalog'.pointRef = ucatalog?.pointRef \also
tcatalog'.price = ucatalog?.price \also
tcatalog'.point = ucatalog?.point \also
\end{schema}

\begin{schema} {Delcatalog}
dcatalog? : catalog \also
\Delta catalogExt \also
DelbackOrder \also
DelgoodsInSelling \also
Delselling \also
\where
dbackOrder?.catalogId = dcatalog?.catalogId \also
dgoodsInSelling?.catalogId = dcatalog?.catalogId \also
dselling?.catalogId = dcatalog?.catalogId \also
catalogSet' = catalogSet \setminus \{ dcatalog? \} \also
\end{schema}

\begin{schema} {Insertcargo}
\Delta cargoExt \also
newValue? : cargo
\where
cargoSet' = cargoSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatecargo}
cargoExt \also
ucargo? : cargo \also
tcargo, tcargo' : cargo \also
\where
tcargo \in cargoSet \also
tcargo.goodsId = ucargo?.goodsId \also
tcargo'.onHandGoods = ucargo?.onHandGoods \also
tcargo'.goodsComment = ucargo?.goodsComment \also
tcargo'.goodsStatus = ucargo?.goodsStatus \also
tcargo'.saftyStock = ucargo?.saftyStock \also
tcargo'.supplierId = ucargo?.supplierId \also
tcargo'.goodsRemain = ucargo?.goodsRemain \also
tcargo'.goodsName = ucargo?.goodsName \also
\end{schema}

\begin{schema} {Delcargo}
dcargo? : cargo \also
\Delta cargoExt \also
Delbuying \also
DelbackOrder \also
DelgoodsInSelling \also
Delcatalog \also
\where
dbuying?.goodsId = dcargo?.goodsId \also
dbackOrder?.goodsId = dcargo?.goodsId \also
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
dgoodsInSelling?.goodsId = dcargo?.goodsId \also
dcatalog?.goodsId = dcargo?.goodsId \also
cargoSet' = cargoSet \setminus \{ dcargo? \} \also
\end{schema}

\begin{schema} {Insertsupplier}
\Delta supplierExt \also
newValue? : supplier
\where
supplierSet' = supplierSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatesupplier}
supplierExt \also
usupplier? : supplier \also
tsupplier, tsupplier' : supplier \also
\where
tsupplier \in supplierSet \also
tsupplier.supplierId = usupplier?.supplierId \also
tsupplier'.supplierJoin = usupplier?.supplierJoin \also
tsupplier'.supplierFax = usupplier?.supplierFax \also
tsupplier'.supplierTel = usupplier?.supplierTel \also
tsupplier'.supplierAdd = usupplier?.supplierAdd \also
tsupplier'.supplierName = usupplier?.supplierName \also
\end{schema}

\begin{schema} {Delsupplier}
dsupplier? : supplier \also
\Delta supplierExt \also
Delcargo \also
\where
dcargo?.supplierId = dsupplier?.supplierId \also
supplierSet' = supplierSet \setminus \{ dsupplier? \} \also
\end{schema}

\begin{schema} {Insertcustomer}
\Delta customerExt \also
newValue? : customer
\where
customerSet' = customerSet \cup \{ newValue? \}
\end{schema}

\begin{schema} {Updatecustomer}
customerExt \also
ucustomer? : customer \also
tcustomer, tcustomer' : customer \also
\where
tcustomer \in customerSet \also
tcustomer.creditNo = ucustomer?.creditNo \also
tcustomer'.customFlag = ucustomer?.customFlag \also
tcustomer'.customTax = ucustomer?.customTax \also
tcustomer'.customTel = ucustomer?.customTel \also
tcustomer'.paid = ucustomer?.paid \also
tcustomer'.point = ucustomer?.point \also
tcustomer'.creditExpire = ucustomer?.creditExpire \also
tcustomer'.customAdd = ucustomer?.customAdd \also
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซดของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
tcustomer'.customerName = ucustomer?.customerName \also
\end{schema}

\begin{schema} {Delcustomer}
dcustomer? : customer \also
\Delta customerExt \also
Delselling \also
\where
dselling?.credit = dcustomer?.creditNo \also
customerSet' = customerSet \setminus \{ dcustomer? \} \also
\end{schema}

\begin{schema} {Relationshipcus\_sel}
cus\_sel : \finset(customer \cross selling)
\where
\forall cus\_sel1,cus\_sel2 : cus\_sel @ second~cus\_sel1 = second~cus\_sel2 \implies first~cus\_sel1 =
first~cus\_sel2 \also
\end{schema}

\begin{schema} {Relationshipoff\_sel}
off\_sel : \finset(officer \cross selling)
\where
\forall off\_sel1,off\_sel2 : off\_sel @ second~off\_sel1 = second~off\_sel2 \implies first~off\_sel1 =
first~off\_sel2 \also
\end{schema}

\begin{schema} {Relationshipoff\_buy}
off\_buy : \finset(officer \cross buying)
\where
\forall off\_buy1,off\_buy2 : off\_buy @ second~off\_buy1 = second~off\_buy2 \implies first~off\_buy1
= first~off\_buy2 \also
\end{schema}

\begin{schema} {Relationshipcat\_sel}
cat\_sel : \finset(catalog \cross selling)
\where
\forall cat\_sel1,cat\_sel2 : cat\_sel @ second~cat\_sel1 = second~cat\_sel2 \implies first~cat\_sel1 =
first~cat\_sel2 \also
\end{schema}

\begin{schema} {Relationshipcat\_good}
cat\_good : \finset(catalog \cross goodsInSelling)
\where
\forall cat\_good1,cat\_good2 : cat\_good @ second~cat\_good1 = second~cat\_good2 \implies
first~cat\_good1 = first~cat\_good2 \also
\end{schema}

\begin{schema} {Relationshipcat\_back}
cat\_back : \finset(catalog \cross backOrder)
\where
\forall cat\_back1,cat\_back2 : cat\_back @ second~cat\_back1 = second~cat\_back2 \implies
first~cat\_back1 = first~cat\_back2 \also
\end{schema}

\begin{schema} {Relationshipsel\_good}
sel\_good : \finset(goodsInSelling \cross selling)
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ (ต่อ)

```
\where
\forall sel\_good1,sel\_good2 : sel\_good @ first~sel\_good1 = first~sel\_good2 \implies
second~sel\_good1 = second~sel\_good2 \also
\end{schema}

\begin{schema} {Relationshipsel\_back}
sel\_back : \finset(backOrder \cross selling)
\where
\forall sel\_back1,sel\_back2 : sel\_back @ first~sel\_back1 = first~sel\_back2 \implies
second~sel\_back1 = second~sel\_back2 \also
\end{schema}

\begin{schema} {Relationshipcar\_back}
car\_back : \finset(backOrder \cross cargo)
\where
\forall car\_back1,car\_back2 : car\_back @ first~car\_back1 = first~car\_back2 \implies
second~car\_back1 = second~car\_back2 \also
\end{schema}

\begin{schema} {Relationshipcar\_cat}
car\_cat : \finset(catalog \cross cargo)
\where
\forall car\_cat1,car\_cat2 : car\_cat @ first~car\_cat1 = first~car\_cat2 \implies second~car\_cat1 =
second~car\_cat2 \also
\end{schema}

\begin{schema} {Relationshipsup\_car}
sup\_car : \finset(cargo \cross supplier)
\where
\forall sup\_car1,sup\_car2 : sup\_car @ first~sup\_car1 = first~sup\_car2 \implies second~sup\_car1 =
second~sup\_car2 \also
\end{schema}

\begin{schema} {Relationshipcar\_good}
car\_good : \finset(cargo \cross goodsInSelling)
\where
\forall car\_good1,car\_good2 : car\_good @ second~car\_good1 = second~car\_good2 \implies
first~car\_good1 = first~car\_good2 \also
\end{schema}

\begin{schema} {Relationshipcar\_buy}
car\_buy : \finset(cargo \cross buying)
\where
\forall car\_buy1,car\_buy2 : car\_buy @ second~car\_buy1 = second~car\_buy2 \implies
first~car\_buy1 = first~car\_buy2 \also
\end{schema}
```

รูปที่ ข-4 ผลลัพธ์สัญกรณ์เซตของระบบการขายของผ่านโทรศัพท์ (ต่อ)

และจากการนำข้อกำหนดสัญกรณ์เซตที่เป็นผลลัพธ์มาทำการตรวจสอบความถูกต้องทางด้านไวยากรณ์ด้วย Z/EVES ได้ผลลัพธ์ดังรูปที่ ข-5

```
Allegro CL for Windows Copyright (C) 1992-1997, Franz Inc., Berkeley, CA, USA.
All rights reserved.
=>
Reading file E:\MAY\test2.zed
definition of Char
definition of A
definition of B
definition of C
definition of D
definition of E
definition of F
definition of G
definition of H
definition of I
definition of J
definition of K
definition of L
definition of M
definition of N
definition of O
definition of P
definition of Q
definition of R
definition of S
definition of T
definition of U
definition of V
definition of W
definition of X
definition of Y
definition of Z
definition of a
definition of b
definition of c
definition of d
definition of e
definition of f
definition of g
definition of h
definition of i
definition of j
definition of k
definition of l
definition of m
definition of n
definition of o
definition of p
definition of q
definition of r
definition of s
definition of t
definition of u
definition of v
definition of w
definition of x
definition of y
```

รูปที่ ข-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบการขายของผ่านโทรศัพท์ด้วย Z/EVES

```
definition of z
definition of String
free type Boolean
... theorem True\$declaration
... theorem False\$declaration
... theorem Boolean\$expansion
... theorem Boolean\$member
... theorem Boolean\$induction
given sets date, float
schema customer
... theorem customer\$declarationPart
schema supplier
... theorem supplier\$declarationPart
schema cargo
... theorem cargo\$declarationPart
schema catalog
... theorem catalog\$declarationPart
schema officer
... theorem officer\$declarationPart
schema selling
... theorem selling\$declarationPart
schema goodsInSelling
... theorem goodsInSelling\$declarationPart
schema backOrder
... theorem backOrder\$declarationPart
schema buying
... theorem buying\$declarationPart
schema customerExt
... theorem customer\$thetasEqual
... theorem customer\$inSet
... theorem customer\$thetaInSet
... theorem customer\$setInPowerSet
... theorem customer\$member
... theorem customer\$thetaMember
... theorem customer\$declaration
... theorem customer\$select\$creditExpire
... theorem customer\$select\$creditNo
... theorem customer\$select\$customAdd
... theorem customer\$select\$customFlag
... theorem customer\$select\$customTax
... theorem customer\$select\$customTel
... theorem customer\$select\$customer
... theorem customer\$select\$paid
... theorem customer\$select\$point
... theorem customerExt\$declarationPart
schema supplierExt
... theorem supplier\$thetasEqual
... theorem supplier\$inSet
... theorem supplier\$thetaInSet
... theorem supplier\$setInPowerSet
... theorem supplier\$member
... theorem supplier\$thetaMember
... theorem supplier\$declaration
... theorem supplier\$select\$supplierAdd
... theorem supplier\$select\$supplierFax
... theorem supplier\$select\$supplierId
```

รูปที่ ช-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบการขายของผ่านโทรศัพท์ด้วย Z/EVES (ต่อ)

```
... theorem supplier\$select\$supplierJoin
... theorem supplier\$select\$supplierName
... theorem supplier\$select\$supplierTel
... theorem supplierExt\$declarationPart
schema cargoExt
... theorem cargo\$thetasEqual
... theorem cargo\$inSet
... theorem cargo\$thetaInSet
... theorem cargo\$setInPowerSet
... theorem cargo\$member
... theorem cargo\$thetaMember
... theorem cargo\$declaration
... theorem cargo\$select\SgoodsComment
... theorem cargo\$select\SgoodsId
... theorem cargo\$select\SgoodsName
... theorem cargo\$select\SgoodsRemain
... theorem cargo\$select\SgoodsStatus
... theorem cargo\$select\SonHandGoods
... theorem cargo\$select\SsaftyStock
... theorem cargo\$select\SsupplierId
... theorem cargoExt\$declarationPart
schema catalogExt
... theorem catalog\$thetasEqual
... theorem catalog\$inSet
... theorem catalog\$thetaInSet
... theorem catalog\$setInPowerSet
... theorem catalog\$member
... theorem catalog\$thetaMember
... theorem catalog\$declaration
... theorem catalog\$select\ScatalogId
... theorem catalog\$select\SgoodsId
... theorem catalog\$select\Spoint
... theorem catalog\$select\SpointRef
... theorem catalog\$select\Sprice
... theorem catalog\$select\SpriceRef
... theorem catalogExt\$declarationPart
schema officerExt
... theorem officer\$thetasEqual
... theorem officer\$inSet
... theorem officer\$thetaInSet
... theorem officer\$setInPowerSet
... theorem officer\$member
... theorem officer\$thetaMember
... theorem officer\$declaration
... theorem officer\$select\SofficerAdd
... theorem officer\$select\SofficerId
... theorem officer\$select\SofficerName
... theorem officer\$select\SofficerTel
... theorem officer\$select\SsecialNo
... theorem officerExt\$declarationPart
schema sellingExt
... theorem selling\$thetasEqual
... theorem selling\$inSet
... theorem selling\$thetaInSet
... theorem selling\$setInPowerSet
... theorem selling\$member
```

รูปที่ ข-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบการขายของผ่านโทรศัพท์ด้วย Z/EVES (ต่อ)

```
... theorem selling\$thetaMember
... theorem selling\$declaration
... theorem selling\$select\$catalogId
... theorem selling\$select\$credit
... theorem selling\$select\$officerId
... theorem selling\$select\$orderDate
... theorem selling\$select\$receiveAdd
... theorem selling\$select\$receiveName
... theorem selling\$select\$receiveTel
... theorem selling\$select\$sellId
... theorem selling\$select\$togetherFlag
... theorem sellingExt\$declarationPart
schema goodsInSellingExt
... theorem goodsInSelling\$thetasEqual
... theorem goodsInSelling\$inSet
... theorem goodsInSelling\$thetaInSet
... theorem goodsInSelling\$setInPowerSet
... theorem goodsInSelling\$member
... theorem goodsInSelling\$thetaMember
... theorem goodsInSelling\$declaration
... theorem goodsInSelling\$select\$catalogId
... theorem goodsInSelling\$select\$goodsId
... theorem goodsInSelling\$select\$quantity
... theorem goodsInSelling\$select\$sellId
... theorem goodsInSelling\$select\$sellMoney
... theorem goodsInSelling\$select\$sellPoint
... theorem goodsInSelling\$select\$sellType
... theorem goodsInSelling\$select\$sentDate
... theorem goodsInSelling\$select\$sentFlag
... theorem goodsInSellingExt\$declarationPart
schema backOrderExt
... theorem backOrder\$thetasEqual
... theorem backOrder\$inSet
... theorem backOrder\$thetaInSet
... theorem backOrder\$setInPowerSet
... theorem backOrder\$member
... theorem backOrder\$thetaMember
... theorem backOrder\$declaration
... theorem backOrder\$select\$catalogId
... theorem backOrder\$select\$goodsId
... theorem backOrder\$select\$quantity
... theorem backOrder\$select\$sellId
... theorem backOrder\$select\$sellMoney
... theorem backOrder\$select\$sellPoint
... theorem backOrder\$select\$sellType
... theorem backOrderExt\$declarationPart
schema buyingExt
... theorem buying\$thetasEqual
... theorem buying\$inSet
... theorem buying\$thetaInSet
... theorem buying\$setInPowerSet
... theorem buying\$member
... theorem buying\$thetaMember
... theorem buying\$declaration
... theorem buying\$select\$buyQuantity
... theorem buying\$select\$goodsId
```

รูปที่ ข-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบการขายของผ่านโทรศัพท์ด้วย Z/EVES (ต่อ)

```
... theorem buying\$select\$orderComment
... theorem buying\$select\$orderDate
... theorem buying\$select\$orderId
... theorem buying\$select\$orderOfficer
... theorem buying\$select\$pricePerUnit
... theorem buying\$select\$receiveFlag
... theorem buying\$select\$receiveOfficer
... theorem buying\$select\$sendDate
... theorem buying\$select\$slirpNo
... theorem buyingExt\$declarationPart
schema Insertbuying
... schema \Delta buyingExt
... theorem Delta\$buyingExt\$declarationPart
... theorem Insertbuying\$declarationPart
schema Updatebuying
... theorem Updatebuying\$declarationPart
schema Delbuying
... theorem Delbuying\$declarationPart
schema InsertbackOrder
... schema \Delta backOrderExt
... theorem Delta\$backOrderExt\$declarationPart
... theorem InsertbackOrder\$declarationPart
schema UpdatebackOrder
... theorem UpdatebackOrder\$declarationPart
schema DelbackOrder
... theorem DelbackOrder\$declarationPart
schema InsertgoodsInSelling
... schema \Delta goodsInSellingExt
... theorem Delta\$goodsInSellingExt\$declarationPart
... theorem InsertgoodsInSelling\$declarationPart
schema UpdategoodsInSelling
... theorem UpdategoodsInSelling\$declarationPart
schema DelgoodsInSelling
... theorem DelgoodsInSelling\$declarationPart
schema Insertselling
... schema \Delta sellingExt
... theorem Delta\$sellingExt\$declarationPart
... theorem Insertselling\$declarationPart
schema Updateselling
... theorem Updateselling\$declarationPart
schema Delselling
... theorem Delselling\$declarationPart
schema Insertofficer
... schema \Delta officerExt
... theorem Delta\$officerExt\$declarationPart
... theorem Insertofficer\$declarationPart
schema Updateofficer
... theorem Updateofficer\$declarationPart
schema Delofficer
... theorem Delofficer\$declarationPart
schema Insertcatalog
... schema \Delta catalogExt
... theorem Delta\$catalogExt\$declarationPart
... theorem Insertcatalog\$declarationPart
schema Updatecatalog
... theorem Updatecatalog\$declarationPart
```

รูปที่ ข-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบการขายของผ่านโทรศัพท์ด้วย Z/EVES (ต่อ)

```
schema Delcatalog
... theorem Delcatalog\$declarationPart
schema Insertcargo
... schema \Delta cargoExt
... theorem Delta\$cargoExt\$declarationPart
... theorem Insertcargo\$declarationPart
schema Updatecargo
... theorem Updatecargo\$declarationPart
schema Delcargo
... theorem Delcargo\$declarationPart
schema Insertsupplier
... schema \Delta supplierExt
... theorem Delta\$supplierExt\$declarationPart
... theorem Insertsupplier\$declarationPart
schema Updatesupplier
... theorem Updatesupplier\$declarationPart
schema Delsupplier
... theorem Delsupplier\$declarationPart
schema Insertcustomer
... schema \Delta customerExt
... theorem Delta\$customerExt\$declarationPart
... theorem Insertcustomer\$declarationPart
schema Updatecustomer
... theorem Updatecustomer\$declarationPart
schema Delcustomer
... theorem Delcustomer\$declarationPart
schema Relationshipcus\_sel
... theorem Relationshipcus\_sel\$declarationPart
schema Relationshipoff\_sel
... theorem Relationshipoff\_sel\$declarationPart
schema Relationshipoff\_buy
... theorem Relationshipoff\_buy\$declarationPart
schema Relationshipcat\_sel
... theorem Relationshipcat\_sel\$declarationPart
schema Relationshipcat\_good
... theorem Relationshipcat\_good\$declarationPart
schema Relationshipcat\_back
... theorem Relationshipcat\_back\$declarationPart
schema Relationshipsel\_good
... theorem Relationshipsel\_good\$declarationPart
schema Relationshipsel\_back
... theorem Relationshipsel\_back\$declarationPart
schema Relationshipcar\_back
... theorem Relationshipcar\_back\$declarationPart
schema Relationshipcar\_cat
... theorem Relationshipcar\_cat\$declarationPart
schema Relationshipsup\_car
... theorem Relationshipsup\_car\$declarationPart
schema Relationshipcar\_good
... theorem Relationshipcar\_good\$declarationPart
schema Relationshipcar\_buy
... theorem Relationshipcar\_buy\$declarationPart
Done.
=>
```

รูปที่ ข-5 ผลลัพธ์การตรวจสอบไวยากรณ์ของระบบการขายของผ่านโทรศัพท์ด้วย Z/EVES (ต่อ)

จากผลลัพธ์ แสดงว่าสัญกรณ์เซตที่ได้มีความถูกต้องทางด้านไวยากรณ์ เนื่องจากสามารถสร้างเป็นทฤษฎีบทได้หมด ไม่มีการแสดงข้อผิดพลาดใด ๆ

ประวัติผู้เขียน

นางสาววัชรวรรณ ศรีรัตนาลัย เกิดเมื่อวันที่ 2 พฤษภาคม พ.ศ. 2520 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาหลักสูตรวิทยาศาสตร์บัณฑิต (วท.บ.) สาขาวิทยาศาสตร์คอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์ เมื่อปีการศึกษา 2540 และเข้าศึกษาต่อหลักสูตรวิทยาศาสตร์มหาบัณฑิต (วท.ม.) ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2541