

## บทที่ 5



### การทดสอบและสรุปผล

ในบทนี้จะกล่าวถึงการทดสอบเครื่องมือแปลงแผนภาพคลาสเป็นข้อกำหนดเขต และวิเคราะห์ข้อกำหนดเขตที่ได้โดยการสร้างเงื่อนไขก่อน การสร้างกรณีทดสอบ และการทดสอบทฤษฎีบท

#### 5.1 ขั้นตอนการติดตั้งโปรแกรม

ทำการติดตั้งโปรแกรม CDZ โดยการเรียก SETUP.EXE แล้วปฏิบัติตามขั้นตอนที่โปรแกรมแนะนำเพื่อติดตั้งเครื่องมือ สำหรับวิธีการใช้งานโปรแกรม CDZ สามารถศึกษาได้จากภาคผนวก ก

#### 5.2 สภาพที่ใช้ทดสอบโปรแกรม

เครื่องคอมพิวเตอร์และระบบปฏิบัติการที่ใช้ในการทดสอบ มีรายละเอียดดังนี้

- คอมพิวเตอร์พีซี Pentium 4 1.6 กิกะเฮิร์ต
- หน่วยความจำหลัก 512 เมกะไบต์
- ฮาร์ดดิสก์ความจุ 60 กิกะไบต์
- ระบบปฏิบัติการไมโครซอฟท์วินโดวส์ 2000

#### 5.3 ระบบที่ใช้ทดสอบโปรแกรม

ขั้นตอนการทดสอบเครื่องมือซอฟต์แวร์ มีลำดับดังนี้

- กำหนดระบบที่ใช้ทดสอบ ซึ่งมี 3 ระบบคือ ระบบสารสนเทศของสถานศึกษา (IS for a school) [7] ระบบฐานข้อมูลถังเลี้ยงปลา (Aquarium Database) [15] และระบบเครื่องฝากถอนเงินอัตโนมัติ (Automatic Teller Machine) [16]

- สร้างข้อกำหนดเขตของแต่ละระบบที่ใช้ทดสอบจากแผนภาพคลาสของระบบ
- ตรวจสอบวากยสัมพันธ์ของข้อกำหนดเขตที่ได้โดยใช้ Z/EVES
- ตรวจสอบเค้าร่างของข้อกำหนดเขตที่สร้างขึ้น ว่าเค้าร่างแต่ละเค้าร่างเป็นไปตามกฎการแปลงแผนภาพคลาสเป็นข้อกำหนดเขตในบทที่ 3 หรือไม่ และนับจำนวนเค้าร่างทั้งหมดในแต่ละระบบ
- วิเคราะห์ข้อกำหนดเขตที่ได้โดยใช้ Z/EVES โดยการสร้างเงื่อนไขก่อน การสร้างกรณีทดสอบ และการทดสอบทฤษฎีบท การสร้างเงื่อนไขก่อนเป็นวิธีที่ใช้เพื่อตรวจสอบการดำเนินการพื้นฐานว่าสถานะก่อนของการดำเนินการนั้น ๆ มีความถูกต้อง ไม่มีข้อผิดพลาด ซึ่งถ้าหากเงื่อนไขก่อนเป็น

เท็จแสดงว่าการนิยามมีความขัดแย้งกัน นั่นคือไม่มีสถานะก่อนที่เหมาะสม แต่ถ้าหากเงื่อนไขก่อนเป็นจริงแสดงว่าสถานะก่อนของการดำเนินการพื้นฐานนั้นถูกต้อง ส่วนการสร้างกรณีทดสอบเป็นการตรวจสอบผลลัพธ์ที่ได้เมื่อมีการทำกับการดำเนินการพื้นฐานต่าง ๆ เป็นลำดับกัน และการทดสอบทฤษฎีบทเป็นการตรวจสอบข้อกำหนดโดยการสร้างเป็นทฤษฎีบท โดยในงานวิจัยนี้จะทำการทดสอบค่าของตัวแปรเข้าเมื่อมีการทำกับการดำเนินการพื้นฐานต่าง ๆ เป็นลำดับกัน

สำหรับรายละเอียดในการทดสอบระบบทั้ง 3 ระบบ ดังนี้

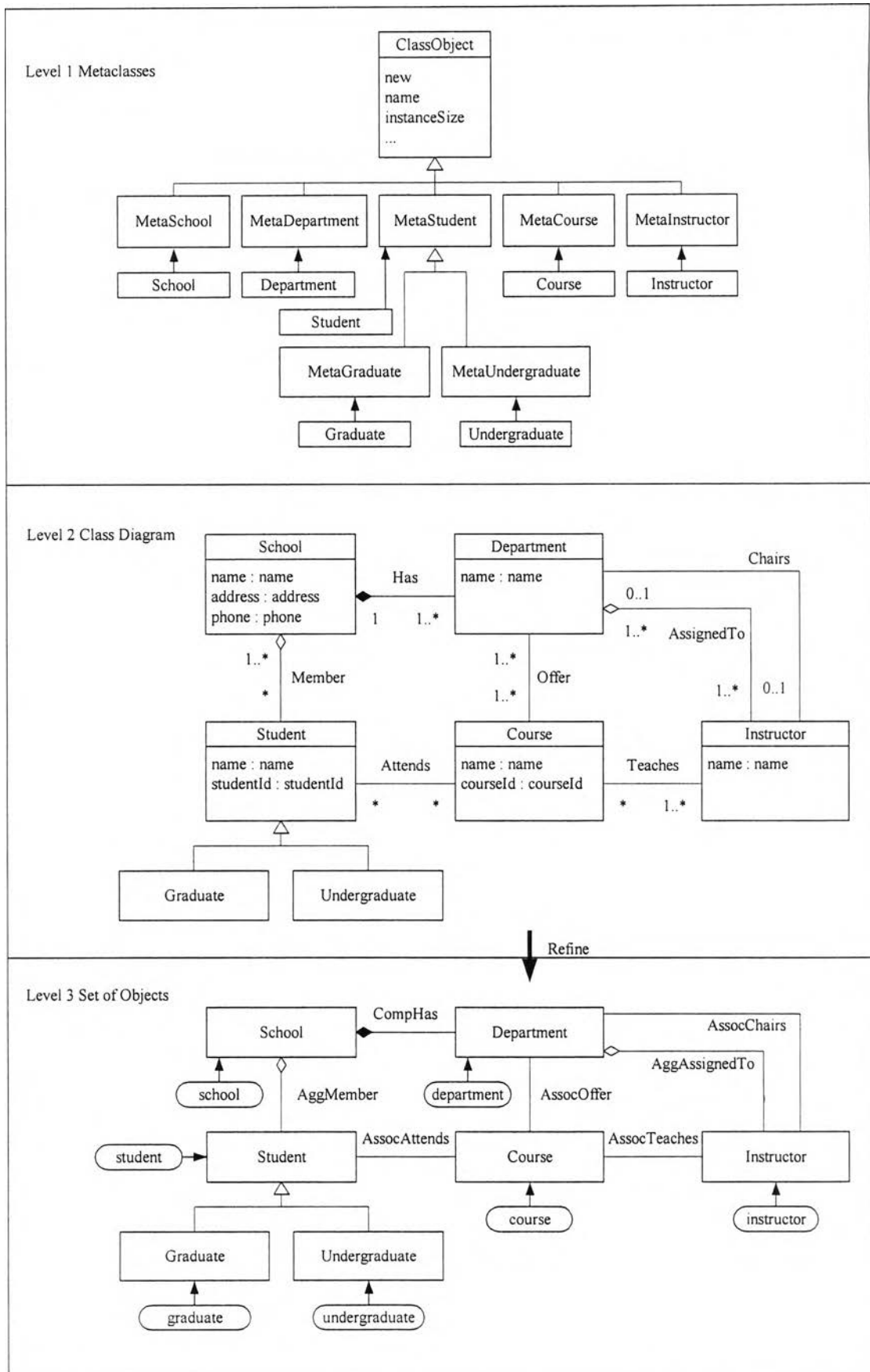
### 5.3.1 ระบบสารสนเทศของสถานศึกษา

ระบบสารสนเทศของสถานศึกษา เป็นระบบที่ใช้ในการบริหารงานเกี่ยวกับสถานศึกษา โดยนิสิตที่ได้ขึ้นทะเบียนเป็นนิสิตของสถานศึกษานั้นจะสามารถลงทะเบียนเรียนได้ โดยวิชาที่เปิดให้ลงทะเบียนเรียนนั้น อาจเปิดโดยแผนกต่าง ๆ หนึ่งหรือหลายแผนกก็ได้ และต้องมีผู้สอนอย่างน้อยหนึ่งคน ซึ่งผู้สอนแต่ละคนจะต้องสังกัดอย่างน้อยหนึ่งแผนก และอาจเป็นหัวหน้าแผนกใด ๆ ได้เพียงหนึ่งแผนกเท่านั้น นอกจากนี้ นิสิตยังแบ่งได้เป็น 2 ชนิดคือ นิสิตหลักสูตรปริญญาบัณฑิต และนิสิตหลักสูตรที่สูงกว่าปริญญาบัณฑิต ดังในรูปที่ 5.1 ในระดับที่ 2 แสดงแผนภาพคลาสของระบบสารสนเทศของสถานศึกษา

จากรูปที่ 5.1 สามารถสร้างเป็นข้อกำหนดเขตของระบบสารสนเทศของสถานศึกษาได้ดังรูปที่ 5.2 โดยข้อมูลนำเข้าและข้อกำหนดเขตที่ได้จากเครื่องมือซอฟต์แวร์ ที่อยู่ในรูปแท่งของลาเท็กซ์แสดงในภาคผนวก ข

การตรวจสอบผลลัพธ์ของข้อกำหนดเขตที่สร้างขึ้น โดยตรวจสอบวากยสัมพันธ์ของข้อกำหนดเขตที่ได้ และวิเคราะห์ข้อกำหนดเขตที่สร้างจากระบบสารสนเทศของสถานศึกษา ได้ดังนี้

- 1) ตรวจสอบวากยสัมพันธ์ของข้อกำหนดเขตที่ได้โดยใช้ Z/EVES พบว่าข้อกำหนดที่ได้นั้นมีความถูกต้องตามวากยสัมพันธ์ของข้อกำหนดเขต
- 2) ระบบสารสนเทศของสถานศึกษามีจำนวนคลาสทั้งหมด 7 คลาส และมีความสัมพันธ์ทั้งหมด 7 ความสัมพันธ์ เมื่อตรวจสอบเค้าร่างของข้อกำหนดเขตที่สร้างขึ้น พบว่าเค้าร่างทุกเค้าร่างในระบบสารสนเทศของสถานศึกษา เป็นไปตามกฎการแปลงแผนภาพคลาสเป็นข้อกำหนดเขตในบทที่ 3 และเมื่อนับจำนวนเค้าร่างทั้งหมดในระบบได้ผลดังนี้
  - จำนวนเค้าร่างคลาสมีทั้งหมด 7 เค้าร่างคือ School Department Student Course Instructor Graduate และ Undergraduate
  - จำนวนเค้าร่างเพิ่มเติมของคลาสมีทั้งหมด 7 เค้าร่างคือ SchoolExt DepartmentExt StudentExt CourseExt InstructorExt GraduateExt และ UndergraduateExt
  - จำนวนเค้าร่างความสัมพันธ์มีทั้งหมด 7 เค้าร่างคือ CompHas AggMember AggAssignTo AssocChairs AssocAttends AssocOffer และ AssocTeaches



รูปที่ 5.1 ระดับของการสร้างข้อกำหนดเขตของระบบสารสนเทศของสถานศึกษา

<p>[<i>SCHOOL</i>, <i>NAME</i>, <i>ADDRESS</i>, <i>PHONE</i>]  [<i>DEPARTMENT</i>]  [<i>STUDENT</i>, <i>STUDENTID</i>]  [<i>COURSE</i>, <i>COURSEID</i>]  [<i>INSTRUCTOR</i>]</p> <hr/> <p>—<i>School</i>—  <i>name</i> : <i>NAME</i>  <i>address</i> : <i>ADDRESS</i>  <i>phone</i> : <i>PHONE</i></p> <hr/> <p>—<i>SchoolExt</i>—  <i>school</i> : <math>F \text{ SCHOOL}</math>  <i>schoolAttrs</i> : <math>\text{SCHOOL} \rightarrow \text{School}</math></p> <hr/> <p>dom <i>schoolAttrs</i> <math>\subseteq</math> <i>school</i></p> <hr/> <p>—<i>Department</i>—  <i>name</i> : <i>NAME</i></p> <hr/> <p>—<i>DepartmentExt</i>—  <i>department</i> : <math>F \text{ DEPARTMENT}</math>  <i>departmentAttrs</i> : <math>\text{DEPARTMENT} \rightarrow \text{Department}</math></p> <hr/> <p>dom <i>departmentAttrs</i> <math>\subseteq</math> <i>department</i></p> <hr/> <p>—<i>Student</i>—  <i>name</i> : <i>NAME</i>  <i>studentId</i> : <i>STUDENTID</i></p> <hr/> <p>—<i>StudentExt</i>—  <i>student</i> : <math>F \text{ STUDENT}</math>  <i>studentAttrs</i> : <math>\text{STUDENT} \rightarrow \text{Student}</math></p> <hr/> <p>dom <i>studentAttrs</i> <math>\subseteq</math> <i>student</i></p> <hr/> <p>—<i>Course</i>—  <i>name</i> : <i>NAME</i>  <i>courseId</i> : <i>COURSEID</i></p> <hr/> <p>—<i>CourseExt</i>—  <i>course</i> : <math>F \text{ COURSE}</math>  <i>courseAttrs</i> : <math>\text{COURSE} \rightarrow \text{Course}</math></p> <hr/> <p>dom <i>courseAttrs</i> <math>\subseteq</math> <i>course</i></p> <hr/> <p>—<i>Instructor</i>—  <i>name</i> : <i>NAME</i></p> <hr/> <p>—<i>InstructorExt</i>—  <i>instructor</i> : <math>F \text{ INSTRUCTOR}</math>  <i>instructorAttrs</i> : <math>\text{INSTRUCTOR} \rightarrow \text{Instructor}</math></p> <hr/> <p>dom <i>instructorAttrs</i> <math>\subseteq</math> <i>instructor</i></p> <hr/> <p>—<i>Graduate</i>—  <i>studentAttrs</i> : <i>Student</i></p>	<p>—<i>GraduateExt</i>—  <i>StudentExt</i>  <i>graduate</i> : <math>F \text{ STUDENT}</math>  <i>graduateAttrs</i> : <math>\text{STUDENT} \rightarrow \text{Graduate}</math></p> <hr/> <p>dom <i>graduateAttrs</i> <math>\subseteq</math> <i>graduate</i>  <i>graduate</i> <math>\in \{\text{student}\}</math>  <math>\forall a \subseteq \text{dom } \text{graduateAttrs} \mid a \in \text{dom } \text{studentAttrs} \cdot</math>  (<i>graduateAttrs</i> <i>a</i>).<i>studentAttrs</i> = (<i>studentAttrs</i> <i>a</i>)</p> <hr/> <p>—<i>Undergraduate</i>—  <i>studentAttrs</i> : <i>Student</i></p> <hr/> <p>—<i>UndergraduateExt</i>—  <i>StudentExt</i>  <i>undergraduate</i> : <math>F \text{ STUDENT}</math>  <i>undergraduateAttrs</i> : <math>\text{STUDENT} \rightarrow \text{Undergraduate}</math></p> <hr/> <p>dom <i>undergraduateAttrs</i> <math>\subseteq</math> <i>undergraduate</i>  <i>undergraduate</i> <math>\in \{\text{student}\}</math>  <math>\forall a \subseteq \text{dom } \text{undergraduateAttrs} \mid</math>  <i>a</i> <math>\in \text{dom } \text{studentAttrs} \cdot</math>  (<i>undergraduateAttrs</i> <i>a</i>).<i>studentAttrs</i> =  (<i>studentAttrs</i> <i>a</i>)</p> <hr/> <p>—<i>CompHas</i>—  <i>DepartmentExt</i>  <i>SchoolExt</i>  <i>compHas</i> : <math>F (\text{DEPARTMENT} \times \text{SCHOOL})</math></p> <hr/> <p>dom <i>compHas</i> <math>\subseteq</math> <i>department</i>  ran <i>compHas</i> <math>\subseteq</math> <i>school</i>  <math>\forall r1, r2 : \text{ran } \text{compHas} \cdot</math>  <i>compHas</i>~(<i>r1</i>) = <i>compHas</i>~(<i>r2</i>) <math>\Rightarrow r1 = r2</math>  <math>\forall x : \text{department} \cdot \#(\text{compHas}(\{\{x\}\})) = 1</math>  <math>\forall y : \text{school} \cdot \#(\text{compHas}(\{\{y\}\})) &gt; 1</math></p> <hr/> <p>—<i>AggMember</i>—  <i>SchoolExt</i>  <i>StudentExt</i>  <i>aggMember</i> : <math>F (\text{SCHOOL} \times \text{STUDENT})</math></p> <hr/> <p>dom <i>aggMember</i> <math>\subseteq</math> <i>school</i>  ran <i>aggMember</i> <math>\subseteq</math> <i>student</i>  <math>\forall x : \text{school} \cdot \#(\text{aggMember}(\{\{x\}\})) &gt; 0</math>  <math>\forall y : \text{student} \cdot \#(\text{aggMember}(\{\{y\}\})) &gt; 1</math></p> <hr/> <p>—<i>AggAssignTo</i>—  <i>DepartmentExt</i>  <i>InstructorExt</i>  <i>aggAssignTo</i> : <math>F (\text{DEPARTMENT} \times \text{INSTRUCTOR})</math></p> <hr/> <p>dom <i>aggAssignTo</i> <math>\subseteq</math> <i>department</i>  ran <i>aggAssignTo</i> <math>\subseteq</math> <i>instructor</i>  <math>\forall x : \text{department} \cdot \#(\text{aggAssignTo}(\{\{x\}\})) &gt; 1</math>  <math>\forall y : \text{instructor} \cdot \#(\text{aggAssignTo}(\{\{y\}\})) &gt; 1</math></p>
--	---

รูปที่ 5.2 ข้อกำหนดเซตของระบบสารสนเทศของสถานศึกษา

<p><u>AssocChairs</u>  <i>DepartmentExt</i>  <i>InstructorExt</i>  <i>assocChairs</i> : F (DEPARTMENT × INSTRUCTOR)</p> <p>dom <i>assocChairs</i> ⊆ <i>department</i>  ran <i>assocChairs</i> ⊆ <i>instructor</i>  ∀ <i>d1</i>, <i>d2</i> : dom <i>assocChairs</i>;  <i>r1</i>, <i>r2</i> : ran <i>assocChairs</i>;  <i>assocChairs</i>(<i>d1</i>) = <i>assocChairs</i>(<i>d2</i>) ⇒ <i>d1</i> = <i>d2</i> ∧  <i>assocChairs</i>~(<i>r1</i>) = <i>assocChairs</i>~(<i>r2</i>) ⇒ <i>r1</i> = <i>r2</i>  ∀ <i>x</i> : <i>department</i> · #(assocChairs( {x} )) &gt; 0 ∧  #(assocChairs( {x} )) &lt; 1  ∀ <i>y</i> : <i>instructor</i> · #(assocChairs~( {y} )) &gt; 0 ∧  #(assocChairs~( {y} )) &lt; 1</p>	<p><u>AddDepartment</u>  Δ <i>DepartmentExt</i>  <i>department?</i> : DEPARTMENT  <i>d?</i> : Department</p> <p><i>department?</i> ∈ dom <i>departmentAttrs</i>  <i>department'</i> = <i>department</i> ∪ {<i>department?</i>}  <i>departmentAttrs'</i> = <i>departmentAttrs</i> ∪  {<i>department?</i> ↦ <i>d?</i>}</p>
<p><u>AssocAttends</u>  <i>StudentExt</i>  <i>CourseExt</i>  <i>assocAttends</i> : F (STUDENT × COURSE)</p> <p>dom <i>assocAttends</i> ⊆ <i>student</i>  ran <i>assocAttends</i> ⊆ <i>course</i>  ∀ <i>x</i> : <i>student</i> · #(assocAttends( {x} )) &gt; 0  ∀ <i>y</i> : <i>course</i> · #(assocAttends~( {y} )) &gt; 0</p>	<p><u>AddStudent</u>  Δ <i>StudentExt</i>  <i>student?</i> : STUDENT  <i>s?</i> : Student</p> <p><i>student?</i> ∈ dom <i>studentAttrs</i>  <i>student'</i> = <i>student</i> ∪ {<i>student?</i>}  <i>studentAttrs'</i> = <i>studentAttrs</i> ∪ {<i>student?</i> ↦ <i>s?</i>}</p>
<p><u>AssocOffer</u>  <i>DepartmentExt</i>  <i>CourseExt</i>  <i>assocOffer</i> : F (DEPARTMENT × COURSE)</p> <p>dom <i>assocOffer</i> ⊆ <i>department</i>  ran <i>assocOffer</i> ⊆ <i>course</i>  ∀ <i>x</i> : <i>department</i> · #(assocOffer( {x} )) &gt; 1  ∀ <i>y</i> : <i>course</i> · #(assocOffer~( {y} )) &gt; 1</p>	<p><u>AddCourse</u>  Δ <i>CourseExt</i>  <i>course?</i> : COURSE  <i>c?</i> : Course</p> <p><i>course?</i> ∈ dom <i>courseAttrs</i>  <i>course'</i> = <i>course</i> ∪ {<i>course?</i>}  <i>courseAttrs'</i> = <i>courseAttrs</i> ∪ {<i>course?</i> ↦ <i>c?</i>}</p>
<p><u>AssocTeaches</u>  <i>CourseExt</i>  <i>InstructorExt</i>  <i>assocTeaches</i> : F (COURSE × INSTRUCTOR)</p> <p>dom <i>assocTeaches</i> ⊆ <i>course</i>  ran <i>assocTeaches</i> ⊆ <i>instructor</i>  ∀ <i>x</i> : <i>course</i> · #(assocTeaches( {x} )) &gt; 1  ∀ <i>y</i> : <i>instructor</i> · #(assocTeaches~( {y} )) &gt; 0</p>	<p><u>AddInstructor</u>  Δ <i>InstructorExt</i>  <i>instructor?</i> : INSTRUCTOR  <i>i?</i> : Instructor</p> <p><i>instructor?</i> ∈ dom <i>instructorAttrs</i>  <i>instructor'</i> = <i>instructor</i> ∪ {<i>instructor?</i>}  <i>instructorAttrs'</i> = <i>instructorAttrs</i> ∪  {<i>instructor?</i> ↦ <i>i?</i>}</p>
<p><u>AddSchool</u>  Δ <i>SchoolExt</i>  <i>school?</i> : SCHOOL  <i>s?</i> : School</p> <p><i>school?</i> ∈ dom <i>schoolAttrs</i>  <i>school'</i> = <i>school</i> ∪ {<i>school?</i>}  <i>schoolAttrs'</i> = <i>schoolAttrs</i> ∪ {<i>school?</i> ↦ <i>s?</i>}</p>	<p><u>AddGraduate</u>  Δ <i>GraduateExt</i>  <i>graduate?</i> : STUDENT  <i>g?</i> : Graduate</p> <p><i>graduate?</i> ∈ dom <i>graduateAttrs</i>  <i>graduate'</i> = <i>graduate</i> ∪ {<i>graduate?</i>}  <i>graduateAttrs'</i> = <i>graduateAttrs</i> ∪ {<i>graduate?</i> ↦ <i>g?</i>}</p>
<p><u>AddUndergraduate</u>  Δ <i>UndergraduateExt</i>  <i>undergraduate?</i> : STUDENT  <i>u?</i> : Undergraduate</p> <p><i>undergraduate?</i> ∈ dom <i>undergraduateAttrs</i>  <i>undergraduate'</i> = <i>undergraduate</i> ∪  {<i>undergraduate?</i>}  <i>undergraduateAttrs'</i> = <i>undergraduateAttrs</i> ∪  {<i>undergraduate?</i> ↦ <i>u?</i>}</p>	<p><u>AddUndergraduate</u>  Δ <i>UndergraduateExt</i>  <i>undergraduate?</i> : STUDENT  <i>u?</i> : Undergraduate</p> <p><i>undergraduate?</i> ∈ dom <i>undergraduateAttrs</i>  <i>undergraduate'</i> = <i>undergraduate</i> ∪  {<i>undergraduate?</i>}  <i>undergraduateAttrs'</i> = <i>undergraduateAttrs</i> ∪  {<i>undergraduate?</i> ↦ <i>u?</i>}</p>

รูปที่ 5.2 ข้อกำหนดเซตของระบบสารสนเทศของสถานศึกษา (ต่อ)

<hr/> <p><i>RemoveDepartment</i>  <math>\Delta</math> <i>DepartmentExt</i>  <i>department?</i> : <i>DEPARTMENT</i></p> <hr/> <p><i>department?</i> <math>\in</math> dom <i>departmentAttrs</i>  <i>department'</i> = <i>department</i> \ {<i>department?</i>}  <i>departmentAttrs'</i> = {<i>department?</i>} <math>\Leftarrow</math> <i>departmentAttrs</i></p>	<hr/> <p><i>ChangeSchool</i>  <math>\Delta</math> <i>SchoolExt</i>  <i>school?</i> : <i>SCHOOL</i>  <i>s?</i> : <i>School</i></p> <hr/> <p><i>school?</i> <math>\in</math> dom <i>schoolAttrs</i>  <i>school</i> = <i>school</i>  <i>schoolAttrs'</i> = <i>schoolAttrs</i> <math>\oplus</math> {<i>school?</i> <math>\mapsto</math> <i>s?</i>}</p>
<hr/> <p><i>RemoveSchool</i>  <math>\Delta</math> <i>CompHas</i>  <i>RemoveDepartment</i>  <i>school?</i> : <i>SCHOOL</i></p> <hr/> <p><i>school?</i> <math>\in</math> dom <i>schoolAttrs</i>  <i>school</i> = <i>school</i> \ {<i>school?</i>}  <i>schoolAttrs'</i> = {<i>school?</i>} <math>\Leftarrow</math> <i>schoolAttrs</i>  {<i>department?</i>} = {<i>s</i> : <i>department</i>   <i>compHas</i>(<i>s</i>) = <i>school?</i> · <i>s</i>}</p>	<hr/> <p><i>ChangeDepartment</i>  <math>\Delta</math> <i>DepartmentExt</i>  <i>department?</i> : <i>DEPARTMENT</i>  <i>d?</i> : <i>Department</i></p> <hr/> <p><i>department?</i> <math>\in</math> dom <i>departmentAttrs</i>  <i>department'</i> = <i>department</i>  <i>departmentAttrs'</i> = <i>departmentAttrs</i> <math>\oplus</math> {<i>department?</i> <math>\mapsto</math> <i>d?</i>}</p>
<hr/> <p><i>RemoveStudent</i>  <math>\Delta</math> <i>StudentExt</i>  <i>student?</i> : <i>STUDENT</i></p> <hr/> <p><i>student?</i> <math>\in</math> dom <i>studentAttrs</i>  <i>student'</i> = <i>student</i> \ {<i>student?</i>}  <i>studentAttrs'</i> = {<i>student?</i>} <math>\Leftarrow</math> <i>studentAttrs</i></p>	<hr/> <p><i>ChangeStudent</i>  <math>\Delta</math> <i>StudentExt</i>  <i>student?</i> : <i>STUDENT</i>  <i>s?</i> : <i>Student</i></p> <hr/> <p><i>student?</i> <math>\in</math> dom <i>studentAttrs</i>  <i>student'</i> = <i>student</i>  <i>studentAttrs'</i> = <i>studentAttrs</i> <math>\oplus</math> {<i>student?</i> <math>\mapsto</math> <i>s?</i>}</p>
<hr/> <p><i>RemoveCourse</i>  <math>\Delta</math> <i>CourseExt</i>  <i>course?</i> : <i>COURSE</i></p> <hr/> <p><i>course?</i> <math>\in</math> dom <i>courseAttrs</i>  <i>course'</i> = <i>course</i> \ {<i>course?</i>}  <i>courseAttrs'</i> = {<i>course?</i>} <math>\Leftarrow</math> <i>courseAttrs</i></p>	<hr/> <p><i>ChangeCourse</i>  <math>\Delta</math> <i>CourseExt</i>  <i>course?</i> : <i>COURSE</i>  <i>c?</i> : <i>Course</i></p> <hr/> <p><i>course?</i> <math>\in</math> dom <i>courseAttrs</i>  <i>course'</i> = <i>course</i>  <i>courseAttrs'</i> = <i>courseAttrs</i> <math>\oplus</math> {<i>course?</i> <math>\mapsto</math> <i>c?</i>}</p>
<hr/> <p><i>RemoveInstructor</i>  <math>\Delta</math> <i>InstructorExt</i>  <i>instructor?</i> : <i>INSTRUCTOR</i></p> <hr/> <p><i>instructor?</i> <math>\in</math> dom <i>instructorAttrs</i>  <i>instructor'</i> = <i>instructor</i> \ {<i>instructor?</i>}  <i>instructorAttrs'</i> = {<i>instructor?</i>} <math>\Leftarrow</math> <i>instructorAttrs</i></p>	<hr/> <p><i>ChangeInstructor</i>  <math>\Delta</math> <i>InstructorExt</i>  <i>instructor?</i> : <i>INSTRUCTOR</i>  <i>i?</i> : <i>Instructor</i></p> <hr/> <p><i>instructor?</i> <math>\in</math> dom <i>instructorAttrs</i>  <i>instructor'</i> = <i>instructor</i>  <i>instructorAttrs'</i> = <i>instructorAttrs</i> <math>\oplus</math> {<i>instructor?</i> <math>\mapsto</math> <i>i?</i>}</p>
<hr/> <p><i>RemoveGraduate</i>  <math>\Delta</math> <i>GraduateExt</i>  <i>graduate?</i> : <i>STUDENT</i></p> <hr/> <p><i>graduate?</i> <math>\in</math> dom <i>graduateAttrs</i>  <i>graduate'</i> = <i>graduate</i> \ {<i>graduate?</i>}  <i>graduateAttrs'</i> = {<i>graduate?</i>} <math>\Leftarrow</math> <i>graduateAttrs</i></p>	<hr/> <p><i>ChangeGraduate</i>  <math>\Delta</math> <i>GraduateExt</i>  <i>graduate?</i> : <i>STUDENT</i>  <i>g?</i> : <i>Graduate</i></p> <hr/> <p><i>graduate?</i> <math>\in</math> dom <i>graduateAttrs</i>  <i>graduate'</i> = <i>graduate</i>  <i>graduateAttrs'</i> = <i>graduateAttrs</i> <math>\oplus</math> {<i>graduate?</i> <math>\mapsto</math> <i>g?</i>}</p>
<hr/> <p><i>RemoveUndergraduate</i>  <math>\Delta</math> <i>UndergraduateExt</i>  <i>undergraduate?</i> : <i>STUDENT</i></p> <hr/> <p><i>undergraduate?</i> <math>\in</math> dom <i>undergraduateAttrs</i>  <i>undergraduate'</i> = <i>undergraduate</i> \ {<i>undergraduate?</i>}  <i>undergraduateAttrs'</i> = {<i>undergraduate?</i>} <math>\Leftarrow</math> <i>undergraduateAttrs</i></p>	

รูปที่ 5.2 ข้อกำหนดเขตของระบบสารสนเทศของสถานศึกษา (ต่อ)

$\text{ChangeUndergraduate}$ $\Delta \text{UndergraduateExt}$ $\text{undergraduate?} : \text{STUDENT}$ $u? : \text{Graduate}$
$\text{undergraduate?} \in \text{dom } \text{undergraduateAttrs}$ $\text{undergraduate}' = \text{undergraduate}$ $\text{undergraduateAttrs}' = \text{undergraduateAttrs} \oplus$ $\{\text{undergraduate?} \mapsto u?\}$

รูปที่ 5.2 ข้อกำหนดเซตของระบบสารสนเทศของสถานศึกษา (ต่อ)

- จำนวนเค้าร่างการเพิ่มมีทั้งหมด 7 เค้าร่างคือ AddSchool AddDepartment AddStudent AddCourse AddInstructor AddGraduate และ AddUndergraduate
  - จำนวนเค้าร่างการลบมีทั้งหมด 7 เค้าร่างคือ RemoveDepartment RemoveSchool RemoveStudent RemoveCourse RemoveInstructor RemoveGraduate และ RemoveUndergraduate
  - จำนวนเค้าร่างการแก้ไขมีทั้งหมด 7 เค้าร่างคือ ChangeSchool ChangeDepartment ChangeStudent ChangeCourse ChangeInstructor ChangeGraduate และ ChangeUndergraduate
- 3) วิเคราะห์ข้อกำหนดเซตที่ได้โดยใช้ Z/EVES โดยการสร้างเงื่อนไขก่อน การสร้างกรณีทดสอบ และการทดสอบทฤษฎีบท ซึ่งในที่นี้จะยกตัวอย่างการวิเคราะห์การดำเนินการพื้นฐาน AddStudent RemoveStudent และ ChangeStudent ดังนี้

$\text{AddStudent}$ $\Delta \text{StudentExt}$ $\text{student?} : \text{STUDENT}$ $s? : \text{Student}$
$\text{student?} \notin \text{dom } \text{studentAttrs}$ $\text{student}' = \text{student} \cup \{\text{student?}\}$ $\text{studentAttrs}' = \text{studentAttrs} \cup \{\text{student?} \mapsto s?\}$

$\text{RemoveStudent}$ $\Delta \text{StudentExt}$ $\text{student?} : \text{STUDENT}$
$\text{student?} \in \text{dom } \text{studentAttrs}$ $\text{student}' = \text{student} \setminus \{\text{student?}\}$ $\text{studentAttrs}' = \{\text{student?}\} \triangleleft \text{studentAttrs}$

$\text{ChangeStudent}$ $\Delta \text{StudentExt}$ $student? : \text{STUDENT}$ $s? : \text{Student}$
$student? \in \text{dom studentAttrs}$ $student' = student$ $studentAttrs' = studentAttrs \oplus \{student? \mapsto s?\}$

- การสร้างเงื่อนไขก่อนของ AddStudent

Theorem preAddStudent

$\forall \text{StudentExt}; student? : \text{STUDENT}; s? : \text{Student} \bullet$

$student? \notin \text{dom studentAttrs} \Rightarrow \text{pre AddStudent}$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true

- การสร้างเงื่อนไขก่อนของ RemoveStudent

Theorem preRemoveStudent

$\forall \text{StudentExt}; student? : \text{STUDENT} \bullet$

$student? \in \text{dom studentAttrs} \wedge$

$\text{dom studentAttrs} \setminus \{student?\} \in \mathbb{P}(\text{student} \setminus \{student?\}) \Rightarrow$

$\text{pre RemoveStudent}$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true

- การสร้างเงื่อนไขก่อนของ ChangeStudent

Theorem preChangeStudent

$\forall \text{StudentExt}; student? : \text{STUDENT}; s? : \text{Student} \bullet$

$student? \in \text{dom studentAttrs} \Rightarrow \text{pre ChangeStudent}$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true

เงื่อนไขก่อนทั้งหมดในระบบสารสนเทศของสถานศึกษา แสดงในตารางที่ 5.1



ตารางที่ 5.1 เงื่อนไขก่อนในระบบสารสนเทศของสถานศึกษา

การดำเนินการ	เงื่อนไขก่อน
AddSchool	$school? \notin \text{dom schoolAttrs}$
AddDepartment	$department? \notin \text{dom departmentAttrs}$
AddStudent	$student? \notin \text{dom studentAttrs}$
AddCourse	$course? \notin \text{dom courseAttrs}$
AddInstructor	$instructor? \notin \text{dom instructorAttrs}$
AddGraduate	$graduate? \notin \text{dom graduateAttrs} \wedge$ $graduate? \notin \text{graduate} \wedge$ $graduate? \in \text{student}$
AddUndergraduate	$undergraduate? \notin \text{dom undergraduateAttrs} \wedge$ $undergraduate? \notin \text{undergraduate} \wedge$ $undergraduate? \in \text{student}$
RemoveDepartment	$department? \in \text{dom departmentAttrs} \wedge$ $\text{dom departmentAttrs} \setminus \{department?\} \in$ $\mathbb{P}(\text{department} \setminus \{department?\})$
RemoveSchool	$school? \in \text{dom schoolAttrs} \wedge$ $school? \in \text{school}$
RemoveStudent	$student? \in \text{dom studentAttrs} \wedge$ $\text{dom studentAttrs} \setminus \{student?\} \in \mathbb{P}(\text{student} \setminus \{student?\})$
RemoveCourse	$course? \in \text{dom courseAttrs} \wedge$ $\text{dom courseAttrs} \setminus \{course?\} \in \mathbb{P}(\text{course} \setminus \{course?\})$
RemoveInstructor	$instructor? \in \text{dom instructorAttrs} \wedge$ $\text{dom instructorAttrs} \setminus \{instructor?\} \in \mathbb{P}(\text{instructor} \setminus \{instructor?\})$
RemoveGraduate	$graduate? \in \text{dom graduateAttrs} \wedge$ $graduate? \in \text{graduate} \wedge$ $graduate? \notin \text{student}$
RemoveUndergraduate	$undergraduate? \in \text{dom undergraduateAttrs} \wedge$ $undergraduate? \in \text{undergraduate} \wedge$ $undergraduate? \notin \text{student}$
ChangeSchool	$school? \in \text{dom schoolAttrs}$
ChangeDepartment	$Department? \in \text{dom departmentAttrs}$

ตารางที่ 5.1 เงื่อนไขก่อนในระบบสารสนเทศของสถานศึกษา (ต่อ)

การดำเนินการ	เงื่อนไขก่อน
ChangeStudent	$student? \in \text{dom studentAttrs}$
ChangeCourse	$course? \in \text{dom courseAttrs}$
ChangeInstructor	$instructor? \in \text{dom instructorAttrs}$
ChangeGraduate	$graduate? \in \text{dom graduateAttrs} \wedge$ $graduate? \in \text{graduate} \wedge$ $graduate? \notin \text{student}$
ChangeUndergraduate	$undergraduate? \in \text{dom undergraduateAttrs} \wedge$ $undergraduate? \in \text{undergraduate} \wedge$ $undergraduate? \notin \text{student}$

- การสร้างกรณีทดสอบของ AddStudent ซึ่งต้องมีการกำหนดสถานะเริ่มต้น (Initial state) ของเซต student และเซต studentAttrs ให้เป็นเซตว่าง ดังนี้

$\text{InitStudent}$ $\text{StudentExt}'$
$student' = \emptyset$ $studentAttrs' = \emptyset$

กรณีทดสอบที่ 1

$\text{TestAdd2Students} \equiv \text{InitStudent} \ ; \ \text{AddStudent} \ ;$   
 $\text{AddStudent}[student2?/student?, s2?/s?]$

จากข้างต้นเป็นการใช้สัญกรณ์ ; (Schema composition) เพื่อเป็นการระบุถึงลำดับของการดำเนินการ จากนั้นใช้คำสั่ง try ดังนี้

$\text{try TestAdd2Students}[student? := \text{OID1}, s? := \text{john},$   
 $student2? := \text{OID2}, s2? := \text{jack}];$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะเป็นการเพิ่มสมาชิกในเซตของ student และ studentAttrs ดังนี้

$$\begin{aligned} \text{student}' &= \{\text{OID1}\} \cup \{\text{OID2}\} \wedge \\ \text{studentAttrs}' &= \{(\text{OID1}, \text{john})\} \cup \{(\text{OID2}, \text{jack})\} \wedge \\ &\neg \text{OID2} = \text{OID1} \end{aligned}$$

try TestAdd2Students[student? := OID1, s? := john,  
student2? := OID1, s2? := jack];

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น false เนื่องจากเงื่อนไขก่อนของการเพิ่มสมาชิกในเซตคือ student?  $\notin$  dom studentAttrs

- การสร้างกรณีทดสอบของ RemoveStudent

กรณีทดสอบที่ 2

TestAddThenRemoveStudent  $\equiv$  InitStudent § AddStudent §  
RemoveStudent[student2?/student?]

try TestAddThenRemoveStudent[student? := OID1, s? := john,  
student2? := OID1];

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้คือ เซตว่างของเซต student และ studentAttrs ดังนี้

$$\text{student}' = \emptyset \wedge \text{studentAttrs}' = \emptyset$$

- การสร้างกรณีทดสอบของ ChangeStudent

กรณีทดสอบที่ 3

TestChangeStudent  $\equiv$  InitStudent § AddStudent §  
ChangeStudent[student2?/student?, s2?/s?]

try TestChangeStudent[student? := OID1, s? := john,  
student2? := OID1, s2? := jack];

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้คือ เป็นการลบสมาชิกในเซตของ studentAttrs ที่มีสมาชิกลำดับที่หนึ่งของคู่ลำดับใน studentAttrs มีค่าเท่ากับตัวแปรเข้า student2? แล้วเพิ่มสมาชิกในเซต studentAttrs เป็นค่าของคู่ลำดับที่ต้องการแก้ไขค่า ดังนี้

$$\text{student}' = \{\text{OID1}\} \wedge \text{studentAttrs}' = \{(\text{OID1}, \text{john})\} \oplus \{(\text{OID1}, \text{jack})\}$$

นั่นก็คือ

$$\text{student}' = \{\text{OID1}\} \wedge \text{studentAttrs}' = \{(\text{OID1}, \text{jack})\}$$

- การทดสอบทฤษฎีบทของ AddStudent

Theorem TwoStudentsOfAddingAreNotSame

$$\forall \text{TestAdd2Students} \bullet \text{student2?} \neq \text{student?}$$

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ AddStudent คือ  $\text{student?} \notin \text{dom studentAttrs}$  นั่นคือตัวแปรเข้า student? และ student2? ต้องไม่เท่ากัน

- การทดสอบทฤษฎีบทของ RemoveStudent

Theorem TwoStudentsOfRemovingAreSame

$$\forall \text{TestAddThenRemoveStudent} \bullet \text{student2?} = \text{student?}$$

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ RemoveStudent คือ  $\text{student?} \in \text{dom studentAttrs} \wedge \text{dom studentAttrs} \setminus \{\text{student?}\} \in \mathcal{P}(\text{student} \setminus \{\text{student?}\})$  นั่นคือตัวแปรเข้า student? และ student2? ต้องเท่ากัน

- การทดสอบทฤษฎีบทของ ChangeStudent

Theorem TwoStudentsOfChangingAreSame

$$\forall \text{TestChangeStudent} \bullet \text{student2?} = \text{student?}$$

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ ChangeStudent คือ  $\text{student?} \in \text{dom studentAttrs}$  นั่นคือตัวแปรเข้า student? และ student2? ต้องเท่ากัน

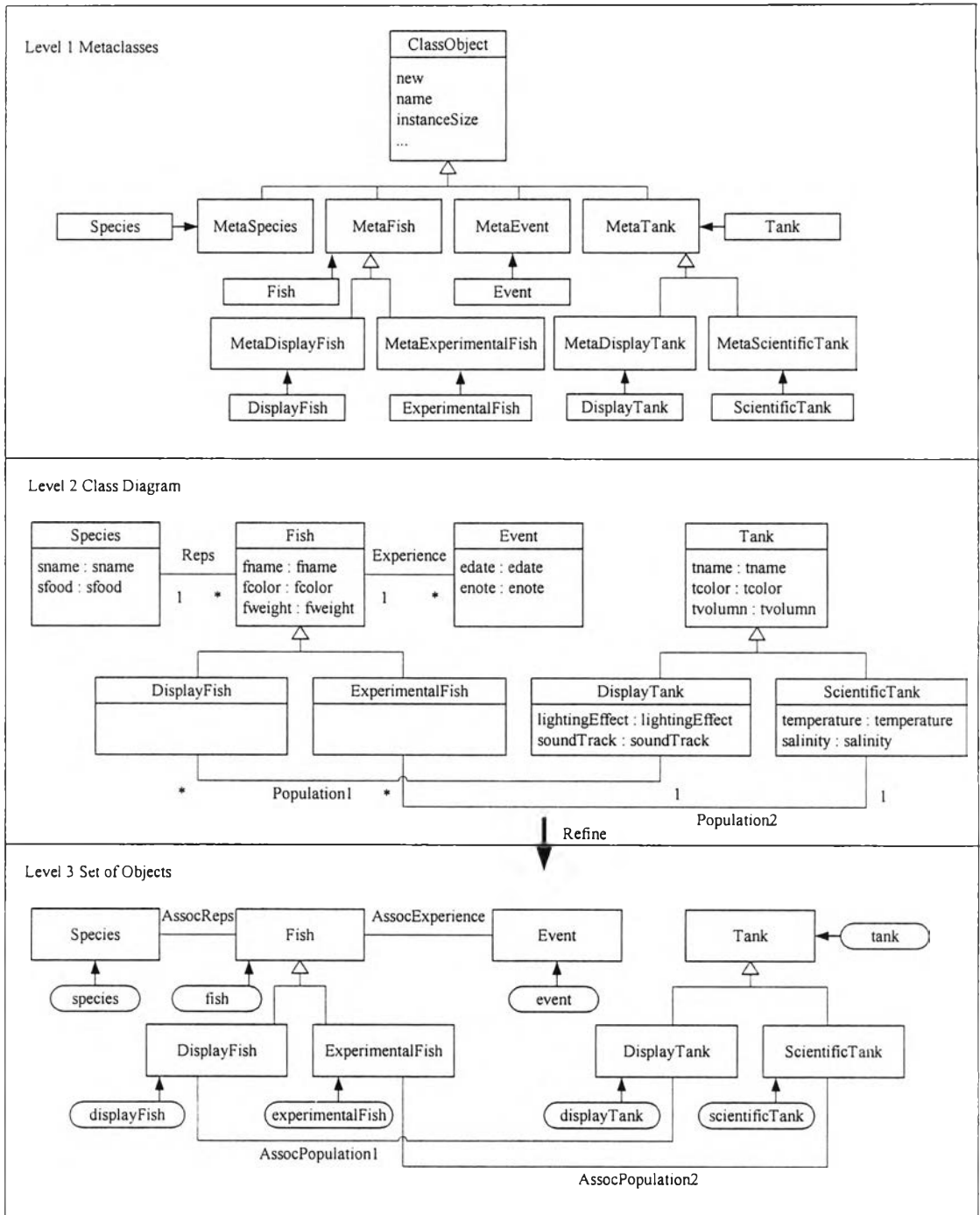
### 5.3.2 ระบบฐานข้อมูลถึงเลี้ยงปลา

ระบบฐานข้อมูลถึงเลี้ยงปลา เป็นระบบที่ใช้ในการจัดการเกี่ยวกับการเลี้ยง การขยายพันธุ์ และการจัดแสดงปลา โดยจะแบ่งปลาออกเป็น 2 ชนิดคือ ปลาสวยงามและปลาทดลอง ดังในรูปที่ 5.3 ในระดับที่ 2 แสดงแผนภาพคลาสของระบบฐานข้อมูลถึงเลี้ยงปลา

จากรูปที่ 5.3 สามารถสร้างเป็นข้อกำหนดเขตของระบบฐานข้อมูลถึงเลี้ยงปลาได้ดังรูปที่ 5.4 โดยข้อมูลนำเข้าและข้อกำหนดเขตที่ได้จากเครื่องมือซอฟต์แวร์ ที่อยู่ในรูปเทคนิคของลาเท็กซ์แสดงในภาคผนวก ค

การตรวจสอบผลลัพธ์ของข้อกำหนดเขตที่สร้างขึ้น โดยตรวจสอบวากยสัมพันธ์ของข้อกำหนดเขตที่ได้ และวิเคราะห์ข้อกำหนดเขตที่สร้างจากระบบฐานข้อมูลถึงเลี้ยงปลา ได้ดังนี้

- 1) ตรวจสอบวากยสัมพันธ์ของข้อกำหนดเขตที่ได้โดยใช้ Z/EVES พบว่าข้อกำหนดที่ได้นั้นมี ความถูกต้องตามวากยสัมพันธ์ของข้อกำหนดเขต
- 2) ระบบฐานข้อมูลถึงเลี้ยงปลามีจำนวนคลาสทั้งหมด 8 คลาส และมีความสัมพันธ์ทั้งหมด 4 ความสัมพันธ์ เมื่อตรวจสอบเค้าร่างของข้อกำหนดเขตที่สร้างขึ้น พบว่าเค้าร่างทุกเค้าร่าง ในระบบฐานข้อมูลถึงเลี้ยงปลา เป็นไปตามกฎการแปลงแผนภาพคลาสเป็นข้อกำหนดเขต ในบทที่ 3 และเมื่อนับจำนวนเค้าร่างทั้งหมดในระบบได้ผลดังนี้
  - จำนวนเค้าร่างคลาสมีทั้งหมด 8 เค้าร่างคือ Species Fish Event Tank DisplayFish ExperimentalFish DisplayTank และ ScientificTank
  - จำนวนเค้าร่างเพิ่มเติมของคลาสมีทั้งหมด 8 เค้าร่างคือ SpeciesExt FishExt EventExt TankExt DisplayFishExt ExperimentalFishExt DisplayTankExt และ ScientificTankExt
  - จำนวนเค้าร่างความสัมพันธ์มีทั้งหมด 4 เค้าร่างคือ AssocReps AssocExperience AssocPopulation1 และ AssocPopulation2
  - จำนวนเค้าร่างการเพิ่มมีทั้งหมด 8 เค้าร่างคือ AddSpecies AddFish AddEvent AddTank AddDisplayFish AddExperimentalFish AddDisplayTank และ AddScientificTank
  - จำนวนเค้าร่างการลบมีทั้งหมด 8 เค้าร่างคือ RemoveSpecies RemoveFish RemoveEvent RemoveTank RemoveDisplayFish RemoveExperimentalFish RemoveDisplayTank และ RemoveScientificTank
  - จำนวนเค้าร่างการแก้ไขมีทั้งหมด 8 เค้าร่างคือ ChangeSpecies ChangeFish ChangeEvent ChangeTank ChangeDisplayFish ChangeExperimentalFish ChangeDisplayTank และ ChangeScientificTank



รูปที่ 5.3 ระดับของการสร้างข้อกำหนดเขตของระบบฐานข้อมูลถึงเลี้ยงปลา

<p>[SPECIES, SNAME, SFOOD]  [FISH, FNAME, FCOLOR, FWEIGHT]  [EVENT, EDATE, ENOTE]  [TANK, TNAME, TCOLOR, TVOLUMN]  [LIGHTINGEFFECT, SOUNDTRACK]  [TEMPERATURE, SALINITY]</p> <hr/> <p>Species  sname : SNAME  sfood : SFOOD</p> <hr/> <p>SpeciesExt  species : F SPECIES  speciesAttrs : SPECIES → Species</p> <hr/> <p>dom speciesAttrs ⊆ species</p> <hr/> <p>Fish  fname : FNAME  fcolor : FCOLOR  fweight : FWEIGHT</p> <hr/> <p>FishExt  fish : F FISH  fishAttrs : FISH → Fish</p> <hr/> <p>dom fishAttrs ⊆ fish</p> <hr/> <p>Event  edate : EDATE  enote : ENOTE</p> <hr/> <p>EventExt  event : F EVENT  eventAttrs : EVENT → Event</p> <hr/> <p>dom eventAttrs ⊆ event</p> <hr/> <p>Tank  tname : TNAME  tcolor : TCOLOR  tvolumn : TVOLUMN</p> <hr/> <p>TankExt  tank : F TANK  tankAttrs : TANK → Tank</p> <hr/> <p>dom tankAttrs ⊆ tank</p> <hr/> <p>DisplayFish  fishattrs : Fish</p> <hr/> <p>DisplayFishExt  FishExt  displayFish : F FISH  displayFishAttrs : FISH → DisplayFish</p> <hr/> <p>dom displayFishAttrs ⊆ displayFish  displayFish ∈ {fish}  ∀ a ⊆ dom displayFishAttrs   a ∈ dom fishAttrs ·  (displayFishAttrs a).fishattrs = (fishAttrs a)</p>	<hr/> <p>ExperimentalFish  fishattrs : Fish</p> <hr/> <p>ExperimentalFishExt  FishExt  experimentalFish : F FISH  experimentalFishAttrs : FISH → ExperimentalFish</p> <hr/> <p>dom experimentalFishAttrs ⊆ experimentalFish  experimentalFish ∈ {fish}  ∀ a ⊆ dom experimentalFishAttrs    a ∈ dom fishAttrs ·  (experimentalFishAttrs a).fishattrs = (fishAttrs a)</p> <hr/> <p>DisplayTank  tankattrs : Tank  lightingEffect : LIGHTINGEFFECT  soundTrack : SOUNDTRACK</p> <hr/> <p>DisplayTankExt  TankExt  displayTank : F TANK  displayTankAttrs : TANK → DisplayTank</p> <hr/> <p>dom displayTankAttrs ⊆ displayTank  displayTank ∈ {tank}  ∀ a ⊆ dom displayTankAttrs   a ∈ dom tankAttrs ·  (displayTankAttrs a).tankattrs = (tankAttrs a)</p> <hr/> <p>ScientificTank  tankattrs : Tank  temperature : TEMPERATURE  salinity : SALINITY</p> <hr/> <p>ScientificTankExt  TankExt  scientificTank : F TANK  scientificTankAttrs : TANK → ScientificTank</p> <hr/> <p>dom scientificTankAttrs ⊆ scientificTank  scientificTank ∈ {tank}  ∀ a ⊆ dom scientificTankAttrs   a ∈ dom tankAttrs ·  (scientificTankAttrs a).tankattrs = (tankAttrs a)</p> <hr/> <p>AssocReps  FishExt  SpeciesExt  assocReps : F (FISH × SPECIES)</p> <hr/> <p>dom assocReps ⊆ fish  ran assocReps ⊆ species  ∀ r1, r2 : ran assocReps ·  assocReps~(r1) = assocReps~(r2) ⇒ r1 = r2  ∀ x : fish · #(assocReps~( {x} )) = 1  ∀ y : species · #(assocReps~( {y} )) ≥ 0</p>
---	--

รูปที่ 5.4 ข้อกำหนดเขตของระบบฐานข้อมูลถึงเลี้ยงปลา

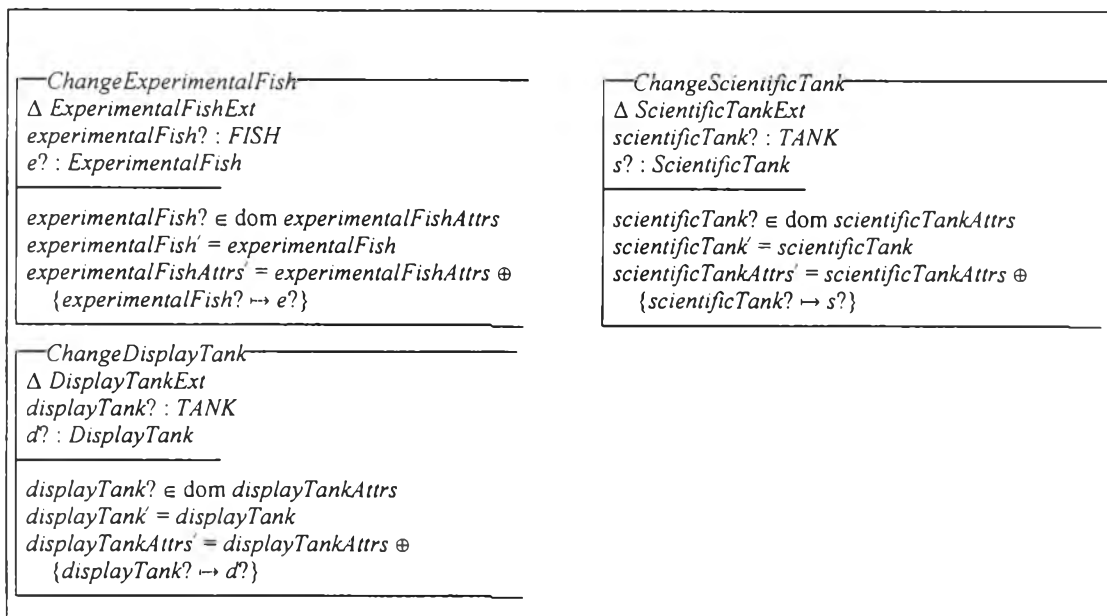
<p><b>AssocExperience</b>  <i>EventExt</i>  <i>FishExt</i>  <i>assocExperience</i> : F (EVENT × FISH)</p> <hr/> <p>dom <i>assocExperience</i> ⊆ <i>event</i>  ran <i>assocExperience</i> ⊆ <i>fish</i>  <math>\forall r1, r2 : \text{ran } \textit{assocExperience} \cdot</math>  <math>\textit{assocExperience} \sim (r1) = \textit{assocExperience} \sim (r2) \Rightarrow r1 = r2</math>  <math>\forall x : \textit{event} \cdot \#(\textit{assocExperience}(\{\{x\}\})) = 1</math>  <math>\forall y : \textit{fish} \cdot \#(\textit{assocExperience} \sim (\{\{y\}\})) &gt; 0</math></p>	<p><b>AddEvent</b>  <math>\Delta \textit{EventExt}</math>  <i>event?</i> : EVENT  <i>e?</i> : Event</p> <hr/> <p><i>event?</i> ∉ dom <i>eventAttrs</i>  <i>event'</i> = <i>event</i> ∪ {<i>event?</i>}  <i>eventAttrs'</i> = <i>eventAttrs</i> ∪ {<i>event?</i> ↦ <i>e?</i>}</p>
<p><b>AssocPopulation1</b>  <i>DisplayFishExt</i>  <i>DisplayTankExt</i>  <i>assocPopulation1</i> : F (FISH × TANK)</p> <hr/> <p>dom <i>assocPopulation1</i> ⊆ <i>displayFish</i>  ran <i>assocPopulation1</i> ⊆ <i>displayTank</i>  <math>\forall r1, r2 : \text{ran } \textit{assocPopulation1} \cdot</math>  <math>\textit{assocPopulation1} \sim (r1) = \textit{assocPopulation1} \sim (r2) \Rightarrow r1 = r2</math>  <math>\forall x : \textit{displayFish} \cdot \#(\textit{assocPopulation1}(\{\{x\}\})) = 1</math>  <math>\forall y : \textit{displayTank} \cdot \#(\textit{assocPopulation1} \sim (\{\{y\}\})) &gt; 0</math></p>	<p><b>AddTank</b>  <math>\Delta \textit{TankExt}</math>  <i>tank?</i> : TANK  <i>t?</i> : Tank</p> <hr/> <p><i>tank?</i> ∉ dom <i>tankAttrs</i>  <i>tank'</i> = <i>tank</i> ∪ {<i>tank?</i>}  <i>tankAttrs'</i> = <i>tankAttrs</i> ∪ {<i>tank?</i> ↦ <i>t?</i>}</p>
<p><b>AssocPopulation2</b>  <i>ExperimentalFishExt</i>  <i>ScientificTankExt</i>  <i>assocPopulation2</i> : F (FISH × TANK)</p> <hr/> <p>dom <i>assocPopulation2</i> ⊆ <i>experimentalFish</i>  ran <i>assocPopulation2</i> ⊆ <i>scientificTank</i>  <math>\forall r1, r2 : \text{ran } \textit{assocPopulation2} \cdot</math>  <math>\textit{assocPopulation2} \sim (r1) = \textit{assocPopulation2} \sim (r2) \Rightarrow r1 = r2</math>  <math>\forall x : \textit{experimentalFish} \cdot</math>  <math>\#(\textit{assocPopulation2}(\{\{x\}\})) = 1</math>  <math>\forall y : \textit{scientificTank} \cdot</math>  <math>\#(\textit{assocPopulation2} \sim (\{\{y\}\})) &gt; 0</math></p>	<p><b>AddDisplayFish</b>  <math>\Delta \textit{DisplayFishExt}</math>  <i>displayFish?</i> : FISH  <i>d?</i> : DisplayFish</p> <hr/> <p><i>displayFish?</i> ∉ dom <i>displayFishAttrs</i>  <i>displayFish'</i> = <i>displayFish</i> ∪ {<i>displayFish?</i>}  <i>displayFishAttrs'</i> = <i>displayFishAttrs</i> ∪ {<i>displayFish?</i> ↦ <i>d?</i>}</p>
<p><b>AddSpecies</b>  <math>\Delta \textit{SpeciesExt}</math>  <i>species?</i> : SPECIES  <i>s?</i> : Species</p> <hr/> <p><i>species?</i> ∉ dom <i>speciesAttrs</i>  <i>species'</i> = <i>species</i> ∪ {<i>species?</i>}  <i>speciesAttrs'</i> = <i>speciesAttrs</i> ∪ {<i>species?</i> ↦ <i>s?</i>}</p>	<p><b>AddExperimentalFish</b>  <math>\Delta \textit{ExperimentalFishExt}</math>  <i>experimentalFish?</i> : FISH  <i>e?</i> : ExperimentalFish</p> <hr/> <p><i>experimentalFish?</i> ∉ dom <i>experimentalFishAttrs</i>  <i>experimentalFish'</i> = <i>experimentalFish</i> ∪ {<i>experimentalFish?</i>}  <i>experimentalFishAttrs'</i> = <i>experimentalFishAttrs</i> ∪ {<i>experimentalFish?</i> ↦ <i>e?</i>}</p>
<p><b>AddFish</b>  <math>\Delta \textit{FishExt}</math>  <i>fish?</i> : FISH  <i>f?</i> : Fish</p> <hr/> <p><i>fish?</i> ∉ dom <i>fishAttrs</i>  <i>fish'</i> = <i>fish</i> ∪ {<i>fish?</i>}  <i>fishAttrs'</i> = <i>fishAttrs</i> ∪ {<i>fish?</i> ↦ <i>f?</i>}</p>	<p><b>AddDisplayTank</b>  <math>\Delta \textit{DisplayTankExt}</math>  <i>displayTank?</i> : TANK  <i>d?</i> : DisplayTank</p> <hr/> <p><i>displayTank?</i> ∉ dom <i>displayTankAttrs</i>  <i>displayTank'</i> = <i>displayTank</i> ∪ {<i>displayTank?</i>}  <i>displayTankAttrs'</i> = <i>displayTankAttrs</i> ∪ {<i>displayTank?</i> ↦ <i>d?</i>}</p>
	<p><b>AddScientificTank</b>  <math>\Delta \textit{ScientificTankExt}</math>  <i>scientificTank?</i> : TANK  <i>s?</i> : ScientificTank</p> <hr/> <p><i>scientificTank?</i> ∉ dom <i>scientificTankAttrs</i>  <i>scientificTank'</i> = <i>scientificTank</i> ∪ {<i>scientificTank?</i>}  <i>scientificTankAttrs'</i> = <i>scientificTankAttrs</i> ∪ {<i>scientificTank?</i> ↦ <i>s?</i>}</p>

รูปที่ 5.4 ข้อกำหนดเขตของระบบฐานข้อมูลถึงเลี้ยงปลา (ต่อ)



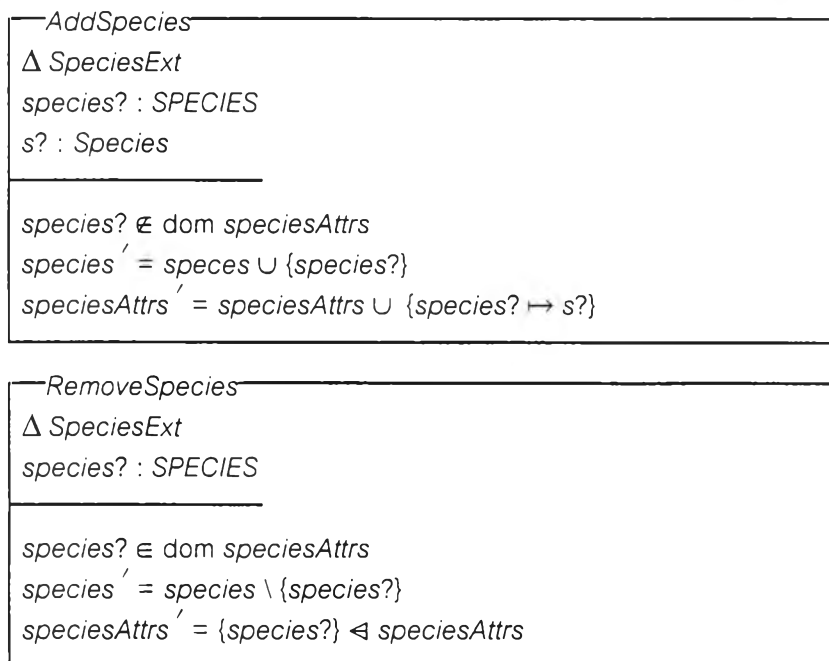
<hr/> <i>RemoveSpecies</i> $\Delta$ <i>SpeciesExt</i> <i>species?</i> : <i>SPECIES</i> <hr/> <i>species?</i> $\in$ dom <i>speciesAttrs</i> <i>species'</i> = <i>species</i> \ { <i>species?</i> } <i>speciesAttrs'</i> = { <i>species?</i> } $\Leftarrow$ <i>speciesAttrs</i> <hr/>	<hr/> <i>RemoveScientificTank</i> $\Delta$ <i>ScientificTankExt</i> <i>scientificTank?</i> : <i>TANK</i> <hr/> <i>scientificTank?</i> $\in$ dom <i>scientificTankAttrs</i> <i>scientificTank'</i> = <i>scientificTank</i> \ { <i>scientificTank?</i> } <i>scientificTankAttrs'</i> = { <i>scientificTank?</i> } $\Leftarrow$ <i>scientificTankAttrs</i> <hr/>
<hr/> <i>RemoveFish</i> $\Delta$ <i>FishExt</i> <i>fish?</i> : <i>FISH</i> <hr/> <i>fish?</i> $\in$ dom <i>fishAttrs</i> <i>fish'</i> = <i>fish</i> \ { <i>fish?</i> } <i>fishAttrs'</i> = { <i>fish?</i> } $\Leftarrow$ <i>fishAttrs</i> <hr/>	<hr/> <i>ChangeSpecies</i> $\Delta$ <i>SpeciesExt</i> <i>species?</i> : <i>SPECIES</i> <i>s?</i> : <i>Species</i> <hr/> <i>species?</i> $\in$ dom <i>speciesAttrs</i> <i>species'</i> = <i>species</i> <i>speciesAttrs'</i> = <i>speciesAttrs</i> $\oplus$ { <i>species?</i> $\mapsto$ <i>s?</i> } <hr/>
<hr/> <i>RemoveEvent</i> $\Delta$ <i>EventExt</i> <i>event?</i> : <i>EVENT</i> <hr/> <i>event?</i> $\in$ dom <i>eventAttrs</i> <i>event'</i> = <i>event</i> \ { <i>event?</i> } <i>eventAttrs'</i> = { <i>event?</i> } $\Leftarrow$ <i>eventAttrs</i> <hr/>	<hr/> <i>ChangeFish</i> $\Delta$ <i>FishExt</i> <i>fish?</i> : <i>FISH</i> <i>f?</i> : <i>Fish</i> <hr/> <i>fish?</i> $\in$ dom <i>fishAttrs</i> <i>fish'</i> = <i>fish</i> <i>fishAttrs'</i> = <i>fishAttrs</i> $\oplus$ { <i>fish?</i> $\mapsto$ <i>f?</i> } <hr/>
<hr/> <i>RemoveTank</i> $\Delta$ <i>TankExt</i> <i>tank?</i> : <i>TANK</i> <hr/> <i>tank?</i> $\in$ dom <i>tankAttrs</i> <i>tank'</i> = <i>tank</i> \ { <i>tank?</i> } <i>tankAttrs'</i> = { <i>tank?</i> } $\Leftarrow$ <i>tankAttrs</i> <hr/>	<hr/> <i>ChangeEvent</i> $\Delta$ <i>EventExt</i> <i>event?</i> : <i>EVENT</i> <i>e?</i> : <i>Event</i> <hr/> <i>event?</i> $\in$ dom <i>eventAttrs</i> <i>event'</i> = <i>event</i> <i>eventAttrs'</i> = <i>eventAttrs</i> $\oplus$ { <i>event?</i> $\mapsto$ <i>e?</i> } <hr/>
<hr/> <i>RemoveDisplayFish</i> $\Delta$ <i>DisplayFishExt</i> <i>displayFish?</i> : <i>FISH</i> <hr/> <i>displayFish?</i> $\in$ dom <i>displayFishAttrs</i> <i>displayFish'</i> = <i>displayFish</i> \ { <i>displayFish?</i> } <i>displayFishAttrs'</i> = { <i>displayFish?</i> } $\Leftarrow$ <i>displayFishAttrs</i> <hr/>	<hr/> <i>ChangeTank</i> $\Delta$ <i>TankExt</i> <i>tank?</i> : <i>TANK</i> <i>t?</i> : <i>Tank</i> <hr/> <i>tank?</i> $\in$ dom <i>tankAttrs</i> <i>tank'</i> = <i>tank</i> <i>tankAttrs'</i> = <i>tankAttrs</i> $\oplus$ { <i>tank?</i> $\mapsto$ <i>t?</i> } <hr/>
<hr/> <i>RemoveExperimentalFish</i> $\Delta$ <i>ExperimentalFishExt</i> <i>experimentalFish?</i> : <i>FISH</i> <hr/> <i>experimentalFish?</i> $\in$ dom <i>experimentalFishAttrs</i> <i>experimentalFish'</i> = <i>experimentalFish</i> \ { <i>experimentalFish?</i> } <i>experimentalFishAttrs'</i> = { <i>experimentalFish?</i> } $\Leftarrow$ <i>experimentalFishAttrs</i> <hr/>	<hr/> <i>ChangeDisplayFish</i> $\Delta$ <i>DisplayFishExt</i> <i>displayFish?</i> : <i>FISH</i> <i>d?</i> : <i>DisplayFish</i> <hr/> <i>displayFish?</i> $\in$ dom <i>displayFishAttrs</i> <i>displayFish'</i> = <i>displayFish</i> <i>displayFishAttrs'</i> = <i>displayFishAttrs</i> $\oplus$ { <i>displayFish?</i> $\mapsto$ <i>d?</i> } <hr/>
<hr/> <i>RemoveDisplayTank</i> $\Delta$ <i>DisplayTankExt</i> <i>displayTank?</i> : <i>TANK</i> <hr/> <i>displayTank?</i> $\in$ dom <i>displayTankAttrs</i> <i>displayTank'</i> = <i>displayTank</i> \ { <i>displayTank?</i> } <i>displayTankAttrs'</i> = { <i>displayTank?</i> } $\Leftarrow$ <i>displayTankAttrs</i> <hr/>	

รูปที่ 5.4 ข้อกำหนดเขตของระบบฐานข้อมูลถึงเลี้ยงปลา (ต่อ)



รูปที่ 5.4 ข้อกำหนดเขตของระบบฐานข้อมูลถึงเลี้ยงปลา (ต่อ)

- 3) วิเคราะห์ข้อกำหนดเขตที่ได้โดยใช้ Z/EVES โดยการสร้างเงื่อนไขก่อน การสร้างกรณีทดสอบ และการทดสอบทฤษฎีบท ซึ่งในที่นี้จะยกตัวอย่างการวิเคราะห์การดำเนินการพื้นฐาน AddSpecies RemoveSpecies และ ChangeSpecies ดังนี้



$\text{ChangeSpecies}$ $\Delta \text{SpeciesExt}$ $\text{species?} : \text{SPECIES}$ $s? : \text{Species}$
$\text{species?} \in \text{dom speciesAttrs}$ $\text{species}' = \text{species}$ $\text{speciesAttrs}' = \text{speciesAttrs} \oplus \{\text{species?} \mapsto s?\}$

- การสร้างเงื่อนไขก่อนของ AddSpecies  
 Theorem preAddSpecies  
 $\forall \text{SpeciesExt}; \text{species?} : \text{SPECIES}; s? : \text{Species} \bullet$   
 $\text{species?} \notin \text{dom speciesAttrs} \Rightarrow \text{pre AddSpecies}$   
 ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true
- การสร้างเงื่อนไขก่อนของ RemoveSpecies  
 Theorem preRemoveSpecies  
 $\forall \text{SpeciesExt}; \text{species?} : \text{SPECIES} \bullet$   
 $\text{species?} \in \text{dom speciesAttrs} \wedge$   
 $\text{dom speciesAttrs} \setminus \{\text{species?}\} \in \mathbb{P}(\text{species} \setminus \{\text{species?}\}) \Rightarrow$   
 $\text{pre RemoveSpecies}$   
 ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true
- การสร้างเงื่อนไขก่อนของ ChangeSpecies  
 Theorem preChangeSpecies  
 $\forall \text{SpeciesExt}; \text{species?} : \text{SPECIES}; s? : \text{Species} \bullet$   
 $\text{species?} \in \text{dom speciesAttrs} \Rightarrow \text{pre ChangeSpecies}$   
 ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true

เงื่อนไขก่อนทั้งหมดในระบบฐานข้อมูลดังกล่าวแสดงในตารางที่ 5.2

ตารางที่ 5.2 เงื่อนไขก่อนในระบบฐานข้อมูลกึ่งเลี้ยงปลา

การดำเนินการ	เงื่อนไขก่อน
AddSpecies	$species? \notin dom\ speciesAttrs$
AddFish	$fish? \notin dom\ fishAttrs$
AddEvent	$event? \notin dom\ eventAttrs$
AddTank	$tank? \notin dom\ tankAttrs$
AddDisplayFish	$displayFish? \notin dom\ displayFishAttrs \wedge$ $displayFish? \notin displayFish \wedge$ $displayFish? \in fish$
AddExperimentalFish	$experimentalFish? \notin dom\ experimentalFishAttrs \wedge$ $experimentalFish? \notin experimentalFish \wedge$ $experimentalFish? \in fish$
AddDisplayTank	$displayTank? \notin dom\ displayTankAttrs \wedge$ $displayTank? \notin displayTank \wedge$ $displayTank? \in tank$
AddScientificTank	$scientificTank? \notin dom\ scientificTankAttrs \wedge$ $scientificTank? \notin scientificTank \wedge$ $scientificTank? \in tank$
RemoveSpecies	$species? \in dom\ speciesAttrs \wedge$ $dom\ speciesAttrs \setminus \{species?\} \in \mathbb{P}(species \setminus \{species?\})$
RemoveFish	$fish? \in dom\ fishAttrs \wedge$ $dom\ fishAttrs \setminus \{fish?\} \in \mathbb{P}(fish \setminus \{fish?\})$
RemoveEvent	$event? \in dom\ eventAttrs \wedge$ $dom\ eventAttrs \setminus \{event?\} \in \mathbb{P}(event \setminus \{event?\})$
RemoveTank	$tank? \in dom\ tankAttrs \wedge$ $dom\ tankAttrs \setminus \{tank?\} \in \mathbb{P}(tank \setminus \{tank?\})$
RemoveDisplayFish	$displayFish? \in dom\ displayFishAttrs \wedge$ $displayFish? \in displayFish \wedge$ $displayFish? \notin fish$
RemoveExperimentalFish	$experimentalFish? \in dom\ experimentalFishAttrs \wedge$ $experimentalFish? \in experimentalFish \wedge$ $experimentalFish? \notin fish$

ตารางที่ 5.2 เงื่อนไขก่อนในระบบฐานข้อมูลถังเลี้ยงปลา (ต่อ)

การดำเนินการ	เงื่อนไขก่อน
RemoveDisplayTank	$displayTank? \in \text{dom } displayTankAttrs \wedge$ $displayTank? \in displayTank \wedge$ $displayTank? \notin tank$
RemoveScientificTank	$scientificTank? \in \text{dom } scientificTankAttrs \wedge$ $scientificTank? \in scientificTank \wedge$ $scientificTank? \notin tank$
ChangeSpecies	$species? \in \text{dom } speciesAttrs$
ChangeFish	$fish? \in \text{dom } fishAttrs$
ChangeEvent	$event? \in \text{dom } eventAttrs$
ChangeTank	$tank? \in \text{dom } tankAttrs$
ChangeDisplayFish	$displayFish? \in \text{dom } displayFishAttrs \wedge$ $displayFish? \in displayFish \wedge$ $displayFish? \notin fish$
ChangeExperimentalFish	$experimentalFish? \in \text{dom } experimentalFishAttrs \wedge$ $experimentalFish? \in experimentalFish \wedge$ $experimentalFish? \notin fish$
ChangeDisplayTank	$displayTank? \in \text{dom } displayTankAttrs \wedge$ $displayTank? \in displayTank \wedge$ $displayTank? \notin tank$
ChangeScientificTank	$scientificTank? \in \text{dom } scientificTankAttrs \wedge$ $scientificTank? \in scientificTank \wedge$ $scientificTank? \notin tank$

- การสร้างกรณีทดสอบของ AddSpecies ซึ่งต้องมีการกำหนดสถานะเริ่มต้น ของเซต species และเซต speciesAttrs ให้เป็นเซตว่าง ดังนี้

$InitSpecies$ $SpeciesExt'$
$species' = \emptyset$ $speciesAttrs' = \emptyset$

กรณีทดสอบที่ 4

TestAdd2Species  $\equiv$  InitSpecies ; AddSpecies ;  
 AddSpecies[species2?/species?, s2?/s?]

จากนั้นใช้คำสั่ง try ดังนี้

```
try TestAdd2Species[species? := OID1, s? := dolphin,
  speies2? := OID2, s2? := shark];
```

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะเป็นการเพิ่มสมาชิกในเซตของ species และ speciesAttrs ดังนี้

$$\begin{aligned} \text{species}' &= \{OID1\} \cup \{OID2\} \wedge \\ \text{speicesAttrs}' &= \{(OID1, dolphin)\} \cup \{(OID2, shark)\} \wedge \\ &\neg OID2 = OID1 \end{aligned}$$

```
try TestAdd2Species[species? := OID1, s? := dolphin,
  species2? := OID1, s2? := shark];
```

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น false เนื่องจากเงื่อนไขก่อนของการเพิ่มสมาชิกในเซตคือ species?  $\notin$  dom speciesAttrs

- การสร้างกรณีทดสอบของ RemoveSpecies

กรณีทดสอบที่ 5

TestAddThenRemoveSpecies  $\equiv$  InitSpeices ; AddSpecies ;  
 RemoveSpecies[species2?/species?]

```
try TestAddThenRemoveSpecies[species? := OID1, s? := dolphin,
  species2? := OID1];
```

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้คือ เซตว่างของเซต species และ speciesAttrs ดังนี้

$$\text{species}' = \emptyset \wedge \text{speciesAttrs}' = \emptyset$$

- การสร้างกรณีทดสอบของ ChangeSpecies

กรณีทดสอบที่ 6

TestChangeSpecies  $\equiv$  InitSpecies ; AddSpecies ;

ChangeSpecies[species2?/species?, s2?/s?]

try TestChangeSpecies[species? := OID1, s? := dolphin,

species2? := OID1, s2? := shark];

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้คือ เป็นการลบสมาชิกในเซตของ speciesAttrs ที่มีสมาชิกลำดับที่หนึ่งของคู่ลำดับใน speciesAttrs มีค่าเท่ากับตัวแปรเข้า species2? แล้วเพิ่มสมาชิกในเซต speciesAttrs เป็นค่าของคู่ลำดับที่ต้องการแก้ไขค่า ดังนี้

species' = {OID1}  $\wedge$  speciesAttrs' = {(OID1, dolphin)}  $\oplus$  {(OID1, shark)}

นั่นก็คือ

species' = {OID1}  $\wedge$  speciesAttrs' = {(OID1, shark)}

- การทดสอบทฤษฎีบทของ AddSpecies

Theorem TwoSpeciesOfAddingAreNotSame

$\forall$  TestAdd2Species • species2?  $\neq$  species?

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ AddSpecies คือ species?  $\notin$  dom speciesAttrs นั่นคือตัวแปรเข้า species? และ species2? ต้องไม่เท่ากัน

- การทดสอบทฤษฎีบทของ RemoveSpecies

Theorem TwoSpeciesOfRemovingAreSame

$\forall$  TestAddThenRemoveSpecies • species2? = species?

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ RemoveSpecies คือ species?  $\in$  dom speciesAttrs  $\wedge$  dom speciesAttrs  $\setminus$  {species?}  $\in$  P(species  $\setminus$  {species?}) นั่นคือตัวแปรเข้า species? และ species2? ต้องเท่ากัน

- การทดสอบทฤษฎีบทของ ChangeSpecies  
Theorem TwoSpeciesOfChangingAreSame  
 $\forall \text{TestChangeSpecies} \cdot \text{species2?} = \text{species?}$

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ ChangeSpecies คือ  $\text{species?} \in \text{dom speciesAttrs}$  นั่นคือตัวแปรเข้า species? และ species2? ต้องเท่ากัน

### 5.3.3 ระบบเครื่องฝากถอนเงินอัตโนมัติ

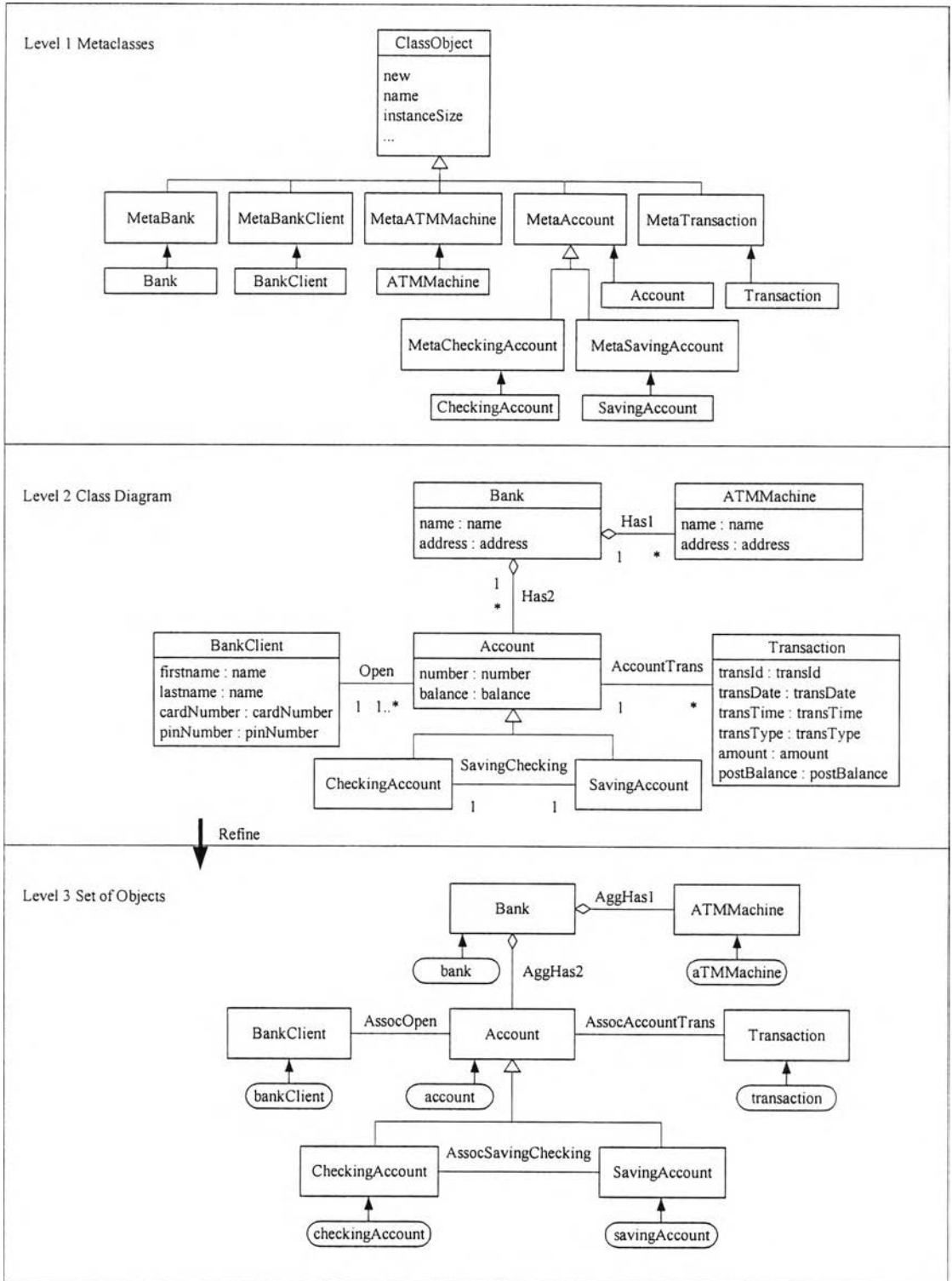
ระบบเครื่องฝากถอนเงินอัตโนมัติ เป็นระบบที่ใช้ในการจัดการเกี่ยวกับการทำรายการทางการเงินของธนาคาร ซึ่งลูกค้าของธนาคารจะต้องทำรายการต่าง ๆ ผ่านระบบเครื่องฝากถอนเงินอัตโนมัติ ดังในรูปที่ 5.5 ในระดับที่ 2 แสดงแผนภาพคลาสของระบบเครื่องฝากถอนเงินอัตโนมัติ

จากรูปที่ 5.5 สามารถสร้างเป็นข้อกำหนดเขตของระบบเครื่องฝากถอนเงินอัตโนมัติได้ดังรูปที่ 5.6 โดยข้อมูลนำเข้าและข้อกำหนดเขตที่ได้จากเครื่องมือซอฟต์แวร์ ที่อยู่ในรูปแท่งของลาเท็กซ์แสดงในภาคผนวก ง

การตรวจสอบผลลัพธ์ของข้อกำหนดเขตที่สร้างขึ้น โดยตรวจสอบวากยสัมพันธ์ของข้อกำหนดเขตที่ได้ และวิเคราะห์ข้อกำหนดเขตที่สร้างจากระบบเครื่องฝากถอนเงินอัตโนมัติ ได้ดังนี้

- 1) ตรวจสอบวากยสัมพันธ์ของข้อกำหนดเขตที่ได้โดยใช้ Z/EVES พบว่าข้อกำหนดที่ได้นั้นมี ความถูกต้องตามวากยสัมพันธ์ของข้อกำหนดเขต
- 2) ระบบเครื่องฝากถอนเงินอัตโนมัติมีจำนวนคลาสทั้งหมด 7 คลาส และมีความสัมพันธ์ทั้งหมด 5 ความสัมพันธ์ เมื่อตรวจสอบเค้าร่างของข้อกำหนดเขตที่สร้างขึ้น พบว่าเค้าร่างทุกเค้าร่างในระบบเครื่องฝากถอนเงินอัตโนมัติ เป็นไปตามกฎการแปลงแผนภาพคลาสเป็นข้อกำหนดเขตในบทที่ 3 และเมื่อนับจำนวนเค้าร่างทั้งหมดในระบบได้ผลดังนี้
  - จำนวนเค้าร่างคลาสมีทั้งหมด 7 เค้าร่างคือ Bank ATMMachine BankClient Account Transaction CheckingAccount และ SavingAccount
  - จำนวนเค้าร่างเพิ่มเติมของคลาสมีทั้งหมด 7 เค้าร่างคือ BankExt ATMMachineExt BankClientExt AccountExt TransactionExt CheckingAccountExt และ SavingAccountExt
  - จำนวนเค้าร่างความสัมพันธ์มีทั้งหมด 5 เค้าร่างคือ AggHas1 AggHas2 AssocOpen AssocAccountTrans และ AssocSavingChecking





รูปที่ 5.5 ระดับของการสร้างข้อกำหนดเขตของระบบเครื่องฝากถอนเงินอัตโนมัติ

<p>[BANK, NAME, ADDRESS] [ATMMACHINE] [BANKCLIENT, FIRSTNAME, LASTNAME, CARDNUMBER, PINNUMBER] [ACCOUNT, NUMBER, BALANCE] [TRANSACTION, TRANSID, TRANSDATE, TRANSTIME, TRANSTYPE, AMOUNT, POSTBALANCE]</p> <hr/> <p>—Bank— name : NAME address : ADDRESS</p> <hr/> <p>—BankExt— bank : F BANK bankAttrs : BANK → Bank</p> <hr/> <p>dom bankAttrs ⊆ bank</p> <hr/> <p>—ATMMachine— name : NAME address : ADDRESS</p> <hr/> <p>—ATMMachineExt— aTMMachine : F ATMMACHINE aTMMachineAttrs : ATMMACHINE → ATMMachine</p> <hr/> <p>dom aTMMachineAttrs ⊆ aTMMachine</p> <hr/> <p>—BankClient— firstname : FIRSTNAME lastname : LASTNAME cardNumber : CARDNUMBER pinNumber : PINNUMBER</p> <hr/> <p>—BankClientExt— bankClient : F BANKCLIENT bankClientAttrs : BANKCLIENT → BankClient</p> <hr/> <p>dom bankClientAttrs ⊆ bankClient</p> <hr/> <p>—Account— number : NUMBER balance : BALANCE</p> <hr/> <p>—AccountExt— account : F ACCOUNT accountAttrs : ACCOUNT → Account</p> <hr/> <p>dom accountAttrs ⊆ account</p> <hr/> <p>—Transaction— transId : TRANSID transDate : TRANSDATE transTime : TRANSTIME transType : TRANSTYPE amount : AMOUNT postBalance : POSTBALANCE</p>	<hr/> <p>—TransactionExt— transaction : F TRANSACTION transactionAttrs : TRANSACTION → Transaction</p> <hr/> <p>dom transactionAttrs ⊆ transaction</p> <hr/> <p>—CheckingAccount— accountAttrs : Account</p> <hr/> <p>—CheckingAccountExt— AccountExt checkingAccount : F ACCOUNT checkingAccountAttrs : ACCOUNT → CheckingAccount</p> <hr/> <p>dom checkingAccountAttrs ⊆ checkingAccount checkingAccount ∈ {account} ∀ a ⊆ dom checkingAccountAttrs   a ∈ dom accountAttrs · (checkingAccountAttrs a).accountAttrs = (accountAttrs a)</p> <hr/> <p>—SavingAccount— accountAttrs : Account</p> <hr/> <p>—SavingAccountExt— AccountExt savingAccount : F ACCOUNT savingAccountAttrs : ACCOUNT → SavingAccount</p> <hr/> <p>dom savingAccountAttrs ⊆ savingAccount savingAccount ∈ {account} ∀ a ⊆ dom savingAccountAttrs   a ∈ dom accountAttrs · (savingAccountAttrs a).accountAttrs = (accountAttrs a)</p> <hr/> <p>—AggHas1— ATMMachineExt BankExt aggHas1 : F (ATMMACHINE × BANK)</p> <hr/> <p>dom aggHas1 ⊆ aTMMachine ran aggHas1 ⊆ bank ∀ r1, r2 : ran aggHas1 · aggHas1~(r1) = aggHas1~(r2) ⇒ r1 = r2 ∀ x : aTMMachine · #(aggHas1({x})) = 1 ∀ y : bank · #(aggHas1~({y})) &gt; 0</p> <hr/> <p>—AggHas2— AccountExt; BankExt aggHas2 : F (ACCOUNT × BANK)</p> <hr/> <p>dom aggHas2 ⊆ account ran aggHas2 ⊆ bank ∀ r1, r2 : ran aggHas2 · aggHas2~(r1) = aggHas2~(r2) ⇒ r1 = r2 ∀ x : account · #(aggHas2({x})) = 1 ∀ y : bank · #(aggHas2~({y})) &gt; 0</p>
--	---

รูปที่ 5.6 ข้อกำหนดเซตของระบบเครื่องฝากถอนเงินอัตโนมัติ

<p><u>AssocOpen</u>  <i>BankClientExt; AccountExt</i>  <i>assocOpen</i> : <math>\mathbb{F} (BANKCLIENT \times ACCOUNT)</math></p> <p><math>\text{dom } assocOpen \subseteq bankClient</math>  <math>\text{ran } assocOpen \subseteq account</math>  <math>\forall r1, r2 : \text{dom } assocOpen \cdot</math>  <math>assocOpen(r1) = assocOpen(r2) \Rightarrow r1 = r2</math>  <math>\forall x : bankClient \cdot \#(assocOpen(\{x\})) &gt; 1</math>  <math>\forall y : account \cdot \#(assocOpen^{-1}(\{y\})) = 1</math></p>	<p><u>AddBankClient</u>  <math>\Delta BankClientExt</math>  <i>bankClient?</i> : <i>BANKCLIENT</i>  <i>b?</i> : <i>BankClient</i></p> <p><math>bankClient? \notin \text{dom } bankClientAttrs</math>  <math>bankClient' = bankClient \cup \{bankClient?\}</math>  <math>bankClientAttrs' = bankClientAttrs \cup</math>  <math>\{bankClient? \mapsto b?\}</math></p>
<p><u>AssocAccountTrans</u>  <i>AccountExt</i>  <i>TransactionExt</i>  <i>assocAccountTrans</i> :  <math>\mathbb{F} (ACCOUNT \times TRANSACTION)</math></p> <p><math>\text{dom } assocAccountTrans \subseteq account</math>  <math>\text{ran } assocAccountTrans \subseteq transaction</math>  <math>\forall r1, r2 : \text{dom } assocAccountTrans \cdot</math>  <math>assocAccountTrans(r1) = assocAccountTrans(r2)</math>  <math>\Rightarrow r1 = r2</math>  <math>\forall x : account \cdot \#(assocAccountTrans(\{x\})) &gt; 0</math>  <math>\forall y : transaction \cdot \#(assocAccountTrans^{-1}(\{y\})) = 1</math></p>	<p><u>AddAccount</u>  <math>\Delta AccountExt</math>  <i>account?</i> : <i>ACCOUNT</i>  <i>a?</i> : <i>Account</i></p> <p><math>account? \notin \text{dom } accountAttrs</math>  <math>account' = account \cup \{account?\}</math>  <math>accountAttrs' = accountAttrs \cup \{account? \mapsto a?\}</math></p>
<p><u>AssocSavingChecking</u>  <i>CheckingAccountExt</i>  <i>SavingAccountExt</i>  <i>assocSavingChecking</i> :  <math>\mathbb{F} (ACCOUNT \times ACCOUNT)</math></p> <p><math>\text{dom } assocSavingChecking \subseteq checkingAccount</math>  <math>\text{ran } assocSavingChecking \subseteq savingAccount</math>  <math>\forall d1, d2 : \text{dom } assocSavingChecking;</math>  <math>r1, r2 : \text{ran } assocSavingChecking;</math>  <math>assocSavingChecking(d1) =</math>  <math>assocSavingChecking(d2) \Rightarrow d1 = d2 \wedge</math>  <math>assocSavingChecking^{-1}(r1) =</math>  <math>assocSavingChecking^{-1}(r2) \Rightarrow r1 = r2</math>  <math>\forall x : checkingAccount \cdot</math>  <math>\#(assocSavingChecking(\{x\})) = 1</math>  <math>\forall y : savingAccount \cdot</math>  <math>\#(assocSavingChecking^{-1}(\{y\})) = 1</math></p>	<p><u>AddTransaction</u>  <math>\Delta TransactionExt</math>  <i>transaction?</i> : <i>TRANSACTION</i>  <i>t?</i> : <i>Transaction</i></p> <p><math>transaction? \notin \text{dom } transactionAttrs</math>  <math>transaction' = transaction \cup \{transaction?\}</math>  <math>transactionAttrs' = transactionAttrs \cup</math>  <math>\{transaction? \mapsto t?\}</math></p>
<p><u>AddBank</u>  <math>\Delta BankExt</math>  <i>bank?</i> : <i>BANK</i>  <i>b?</i> : <i>Bank</i></p> <p><math>bank? \notin \text{dom } bankAttrs</math>  <math>bank' = bank \cup \{bank?\}</math>  <math>bankAttrs' = bankAttrs \cup \{bank? \mapsto b?\}</math></p>	<p><u>AddCheckingAccount</u>  <math>\Delta CheckingAccountExt</math>  <i>checkingAccount?</i> : <i>ACCOUNT</i>  <i>c?</i> : <i>CheckingAccount</i></p> <p><math>checkingAccount? \notin \text{dom } checkingAccountAttrs</math>  <math>checkingAccount' = checkingAccount \cup</math>  <math>\{checkingAccount?\}</math>  <math>checkingAccountAttrs' = checkingAccountAttrs \cup</math>  <math>\{checkingAccount? \mapsto c?\}</math></p>
<p><u>AddATMMachine</u>  <math>\Delta ATMMachineExt</math>  <i>aTMMachine?</i> : <i>ATMMACHINE</i>  <i>a?</i> : <i>ATMMachine</i></p> <p><math>aTMMachine? \notin \text{dom } aTMMachineAttrs</math>  <math>aTMMachine' = aTMMachine \cup \{aTMMachine?\}</math>  <math>aTMMachineAttrs' = aTMMachineAttrs \cup</math>  <math>\{aTMMachine? \mapsto a?\}</math></p>	<p><u>AddSavingAccount</u>  <math>\Delta SavingAccountExt</math>  <i>savingAccount?</i> : <i>ACCOUNT</i>  <i>s?</i> : <i>SavingAccount</i></p> <p><math>savingAccount? \notin \text{dom } savingAccountAttrs</math>  <math>savingAccount' = savingAccount \cup</math>  <math>\{savingAccount?\}</math>  <math>savingAccountAttrs' = savingAccountAttrs \cup</math>  <math>\{savingAccount? \mapsto s?\}</math></p>
	<p><u>RemoveBank</u>  <math>\Delta BankExt</math>  <i>bank?</i> : <i>BANK</i></p> <p><math>bank? \in \text{dom } bankAttrs</math>  <math>bank' = bank \setminus \{bank?\}</math>  <math>bankAttrs' = \{bank?\} \triangleleft bankAttrs</math></p>

รูปที่ 5.6 ข้อกำหนดเซตของระบบเครื่องฝากถอนเงินอัตโนมัติ (ต่อ)

<p>—RemoveATMMachine—  <math>\Delta</math> ATMMachineExt  <math>aTMMachine? : ATMMACHINE</math></p> <hr/> <p><math>aTMMachine? \in \text{dom } aTMMachineAttrs</math>  <math>aTMMachine' = aTMMachine \setminus \{aTMMachine?\}</math>  <math>aTMMachineAttrs' = \{aTMMachine?\} \triangleleft aTMMachineAttrs</math></p>	<p>—ChangeATMMachine—  <math>\Delta</math> ATMMachineExt  <math>aTMMachine? : ATMMACHINE</math>  <math>a? : ATMMachine</math></p> <hr/> <p><math>aTMMachine? \in \text{dom } aTMMachineAttrs</math>  <math>aTMMachine' = aTMMachine</math>  <math>aTMMachineAttrs' = aTMMachineAttrs \oplus \{aTMMachine? \rightarrow a?\}</math></p>
<p>—RemoveBankClient—  <math>\Delta</math> BankClientExt  <math>bankClient? : BANKCLIENT</math></p> <hr/> <p><math>bankClient? \in \text{dom } bankClientAttrs</math>  <math>bankClient' = bankClient \setminus \{bankClient?\}</math>  <math>bankClientAttrs' = \{bankClient?\} \triangleleft bankClientAttrs</math></p>	<p>—ChangeBankClient—  <math>\Delta</math> BankClientExt  <math>bankClient? : BANKCLIENT</math>  <math>b? : BankClient</math></p> <hr/> <p><math>bankClient? \in \text{dom } bankClientAttrs</math>  <math>bankClient' = bankClient</math>  <math>bankClientAttrs' = bankClientAttrs \oplus \{bankClient? \rightarrow b?\}</math></p>
<p>—RemoveAccount—  <math>\Delta</math> AccountExt  <math>account? : ACCOUNT</math></p> <hr/> <p><math>account? \in \text{dom } accountAttrs</math>  <math>account' = account \setminus \{account?\}</math>  <math>accountAttrs' = \{account?\} \triangleleft accountAttrs</math></p>	<p>—ChangeAccount—  <math>\Delta</math> AccountExt  <math>account? : ACCOUNT</math>  <math>a? : Account</math></p> <hr/> <p><math>account? \in \text{dom } accountAttrs</math>  <math>account' = account</math>  <math>accountAttrs' = accountAttrs \oplus \{account? \rightarrow a?\}</math></p>
<p>—RemoveTransaction—  <math>\Delta</math> TransactionExt  <math>transaction? : TRANSACTION</math></p> <hr/> <p><math>transaction? \in \text{dom } transactionAttrs</math>  <math>transaction' = transaction \setminus \{transaction?\}</math>  <math>transactionAttrs' = \{transaction?\} \triangleleft transactionAttrs</math></p>	<p>—ChangeTransaction—  <math>\Delta</math> TransactionExt  <math>transaction? : TRANSACTION</math>  <math>t? : Transaction</math></p> <hr/> <p><math>transaction? \in \text{dom } transactionAttrs</math>  <math>transaction' = transaction</math>  <math>transactionAttrs' = transactionAttrs \oplus \{transaction? \rightarrow t?\}</math></p>
<p>—RemoveCheckingAccount—  <math>\Delta</math> CheckingAccountExt  <math>checkingAccount? : ACCOUNT</math></p> <hr/> <p><math>checkingAccount? \in \text{dom } checkingAccountAttrs</math>  <math>checkingAccount' = checkingAccount \setminus \{checkingAccount?\}</math>  <math>checkingAccountAttrs' = \{checkingAccount?\} \triangleleft checkingAccountAttrs</math></p>	<p>—ChangeCheckingAccount—  <math>\Delta</math> CheckingAccountExt  <math>checkingAccount? : ACCOUNT</math>  <math>c? : CheckingAccount</math></p> <hr/> <p><math>checkingAccount? \in \text{dom } checkingAccountAttrs</math>  <math>checkingAccount' = checkingAccount</math>  <math>checkingAccountAttrs' = checkingAccountAttrs \oplus \{checkingAccount? \rightarrow c?\}</math></p>
<p>—RemoveSavingAccount—  <math>\Delta</math> SavingAccountExt  <math>savingAccount? : ACCOUNT</math></p> <hr/> <p><math>savingAccount? \in \text{dom } savingAccountAttrs</math>  <math>savingAccount' = savingAccount \setminus \{savingAccount?\}</math>  <math>savingAccountAttrs' = \{savingAccount?\} \triangleleft savingAccountAttrs</math></p>	<p>—ChangeSavingAccount—  <math>\Delta</math> SavingAccountExt  <math>savingAccount? : ACCOUNT</math>  <math>s? : SavingAccount</math></p> <hr/> <p><math>savingAccount? \in \text{dom } savingAccountAttrs</math>  <math>savingAccount' = savingAccount</math>  <math>savingAccountAttrs' = savingAccountAttrs \oplus \{savingAccount? \rightarrow s?\}</math></p>
<p>—ChangeBank—  <math>\Delta</math> BankExt  <math>bank? : BANK</math>  <math>b? : Bank</math></p> <hr/> <p><math>bank? \in \text{dom } bankAttrs</math>  <math>bank' = bank</math>  <math>bankAttrs' = bankAttrs \oplus \{bank? \rightarrow b?\}</math></p>	

รูปที่ 5.6 ข้อกำหนดเซตของระบบเครื่องฝากถอนเงินอัตโนมัติ (ต่อ)

- จำนวนเค้าร่างการเพิ่มมีทั้งหมด 7 เค้าร่างคือ AddBank AddATMMachine AddBankClient AddAccount AddTransaction AddCheckingAccount และ AddSavingAccount
  - จำนวนเค้าร่างการลบมีทั้งหมด 7 เค้าร่างคือ RemoveBank RemoveATMMachine RemoveBankClient RemoveAccount RemoveTransaction RemoveCheckingAccount และ RemoveSavingAccount
  - จำนวนเค้าร่างการแก้ไขมีทั้งหมด 7 เค้าร่างคือ ChangeBank ChangeATMMachine ChangeBankClient ChangeAccount ChangeTransaction ChangeCheckingAccount และ ChangeSavingAccount
- 3) วิเคราะห์ข้อกำหนดเขตที่ได้โดยใช้ Z/EVES โดยการสร้างเงื่อนไขก่อน การสร้างกรณีทดสอบ และการทดสอบทฤษฎีบท ซึ่งในที่นี้จะยกตัวอย่างการวิเคราะห์การดำเนินการพื้นฐาน AddAccount RemoveAccount และ ChangeAccount ดังนี้

<p>—AddAccount—</p> <p><math>\Delta</math> AccountExt</p> <p><math>account? : ACCOUNT</math></p> <p><math>a? : Account</math></p> <hr/> <p><math>account? \notin \text{dom } accountAttrs</math></p> <p><math>account' = account \cup \{account?\}</math></p> <p><math>accountAttrs' = accountAttrs \cup \{account? \mapsto a?\}</math></p>
---

<p>—RemoveAccount—</p> <p><math>\Delta</math> AccountExt</p> <p><math>account? : ACCOUNT</math></p> <hr/> <p><math>account? \in \text{dom } accountAttrs</math></p> <p><math>account' = account \setminus \{account?\}</math></p> <p><math>accountAttrs' = \{account?\} \triangleleft accountAttrs</math></p>
---

<p>—ChangeAccount—</p> <p><math>\Delta</math> AccountExt</p> <p><math>account? : ACCOUNT</math></p> <p><math>a? : Account</math></p> <hr/> <p><math>account? \in \text{dom } accountAttrs</math></p> <p><math>account' = account</math></p> <p><math>accountAttrs' = accountAttrs \oplus \{account? \mapsto a?\}</math></p>
---

- การสร้างเงื่อนไขก่อนของ AddAccount

Theorem preAddAccount

$\forall \text{AccountExt}; \text{account?} : \text{ACCOUNT}; a? : \text{Account} \bullet$

$\text{account?} \notin \text{dom accountAttrs} \Rightarrow \text{pre AddAccount}$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true

- การสร้างเงื่อนไขก่อนของ RemoveAccount

Theorem preRemoveAccount

$\forall \text{AccountExt}; \text{account?} : \text{ACCOUNT} \bullet$

$\text{account?} \in \text{dom accountAttrs} \wedge$

$\text{dom accountAttrs} \setminus \{\text{account?}\} \in \mathbb{P}(\text{account} \setminus \{\text{account?}\}) \Rightarrow$

$\text{pre RemoveAccount}$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true

- การสร้างเงื่อนไขก่อนของ ChangeAccount

Theorem preChangeAccount

$\forall \text{AccountExt}; \text{account?} : \text{ACCOUNT}; a? : \text{Account} \bullet$

$\text{account?} \in \text{dom accountAttrs} \Rightarrow \text{pre ChangeAccount}$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true

เงื่อนไขก่อนทั้งหมดในระบบเครื่องฝากถอนเงินอัตโนมัติ แสดงในตารางที่ 5.3

ตารางที่ 5.3 เงื่อนไขก่อนในระบบเครื่องฝากถอนเงินอัตโนมัติ

การดำเนินการ	เงื่อนไขก่อน
AddBank	$\text{bank?} \notin \text{dom bankAttrs}$
AddATMMachine	$\text{aTMMachine?} \notin \text{dom aTMMachineAttrs}$
AddBankClient	$\text{bankClient?} \notin \text{dom bankClientAttrs}$
AddAccount	$\text{account?} \notin \text{dom accountAttrs}$
AddTransaction	$\text{transaction?} \notin \text{dom transactionAttrs}$
AddCheckingAccount	$\text{checkingAccount?} \notin \text{dom checkingAccountAttrs} \wedge$ $\text{checkingAccount?} \notin \text{checkingAccount} \wedge$ $\text{checkingAccount?} \in \text{account}$

ตารางที่ 5.3 เงื่อนไขก่อนในระบบเครื่องฝากถอนเงินอัตโนมัติ (ต่อ)

การดำเนินการ	เงื่อนไขก่อน
AddSavingAccount	savingAccount? $\notin$ dom savingAccountAttrs $\wedge$ savingAccount? $\notin$ savingAccount $\wedge$ savingAccount? $\in$ account
RemoveBank	bank? $\in$ dom bankAttrs $\wedge$ dom bankAttrs $\setminus$ {bank?} $\in$ $\mathbb{P}$ (bank $\setminus$ {bank?})
RemoveATMMachine	aTMMachine? $\in$ dom aTMMachineAttrs $\wedge$ dom aTMMachineAttrs $\setminus$ {aTMMachine?} $\in$ $\mathbb{P}$ (aTMMachine $\setminus$ {aTMMachine?})
RemoveBankClient	bankClient? $\in$ dom bankClientAttrs $\wedge$ dom bankClientAttrs $\setminus$ {bankClient?} $\in$ $\mathbb{P}$ (bankClient $\setminus$ {bankClient?})
RemoveAccount	account? $\in$ dom accountAttrs $\wedge$ dom accountAttrs $\setminus$ {account?} $\in$ $\mathbb{P}$ (account $\setminus$ {account?})
RemoveTransaction	transaction? $\in$ dom transactionAttrs $\wedge$ dom transactionAttrs $\setminus$ {transaction?} $\in$ $\mathbb{P}$ (transaction $\setminus$ {transaction?})
RemoveCheckingAccount	checkingAccount? $\in$ dom checkingAccountAttrs $\wedge$ checkingAccount? $\in$ checkingAccount $\wedge$ checkingAccount? $\notin$ checkingAccount
RemoveSavingAccount	savingAccount? $\in$ dom savingAccountAttrs $\wedge$ savingAccount? $\in$ savingAccount $\wedge$ savingAccount? $\notin$ savingAccount
ChangeBank	bank? $\in$ dom bankAttrs
ChangeATMMachine	aTMMachine? $\in$ dom aTMMachineAttrs
ChangeBankClient	bankClient? $\in$ dom bankClientAttrs
ChangeAccount	account? $\in$ dom accountAttrs
ChangeTransaction	transaction? $\in$ dom transactionAttrs
ChangeCheckingAccount	checkingAccount? $\in$ dom checkingAccountAttrs $\wedge$ checkingAccount? $\in$ checkingAccount $\wedge$ checkingAccount? $\notin$ account
ChangeSavingAccount	savingAccount? $\in$ dom savingAccountAttrs $\wedge$ savingAccount? $\in$ savingAccount $\wedge$ savingAccount? $\notin$ account

- การสร้างกรณีทดสอบของ AddAccount ซึ่งต้องมีการกำหนดสถานะเริ่มต้นของเซต account และเซต accountAttrs ให้เป็นเซตว่าง ดังนี้

$\text{InitAccount}$ $\text{AccountExt}'$
$\text{account}' = \emptyset$ $\text{accountAttrs}' = \emptyset$

กรณีทดสอบที่ 7

TestAdd2Accounts  $\equiv$  InitAccount ; AddAccount ;  
 AddAccount[account2?/account?, a2?/a?]

จากนั้นใช้คำสั่ง try ดังนี้

try TestAdd2Accounts[account? := OID1, a? := account101,  
 account2? := OID2, a2? := account102];

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะเป็นการเพิ่มสมาชิกในเซตของ account และ accountAttrs ดังนี้

$$\begin{aligned} \text{account}' &= \{\text{OID1}\} \cup \{\text{OID2}\} \wedge \\ \text{accountAttrs}' &= \{(\text{OID1}, \text{account101})\} \cup \{(\text{OID2}, \text{account102})\} \\ &\wedge \neg \text{OID2} = \text{OID1} \end{aligned}$$

try TestAdd2Accounts[account? := OID1, a? := account101,  
 account2? := OID1, a2? := account102];

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น false เนื่องจากเงื่อนไขก่อนของการเพิ่มสมาชิกในเซตคือ account?  $\notin$  dom accountAttrs

- การสร้างกรณีทดสอบของ RemoveAccount

กรณีทดสอบที่ 8

TestAddThenRemoveAccount  $\equiv$  InitAccount ; AddAccount ;  
 RemoveAccount[account2?/account?]

try TestAddThenRemoveAccount[account? := OID1, a? := account101,  
 account2? := OID1];



ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้คือ เซตว่างของเซต account และ accountAttrs ดังนี้

$$\text{account}' = \emptyset \wedge \text{accountAttrs}' = \emptyset$$

- การสร้างกรณีทดสอบของ ChangeAccount  
กรณีทดสอบที่ 9

TestChangeAccount  $\equiv$  InitAccount  $\wp$  AddAccount  $\wp$

$$\text{ChangeAccount}[\text{account2?}/\text{account?}, \text{a2?}/\text{a?}]$$

try TestChangeAccount[account? := OID1, a? := account101,

$$\text{account2?} := \text{OID1}, \text{a2?} := \text{account102}];$$

ซึ่งเมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้คือ เป็นการลบสมาชิกในเซตของ accountAttrs ที่มีสมาชิกลำดับที่หนึ่งของคู่ลำดับใน accountAttrs มีค่าเท่ากับตัวแปรเข้า account2? แล้วเพิ่มสมาชิกในเซต accountAttrs เป็นค่าของคู่ลำดับที่ต้องการแก้ไขค่า ดังนี้

$$\text{account}' = \{\text{OID1}\} \wedge$$

$$\text{accountAttrs}' = \{(\text{OID1}, \text{account101})\} \oplus \{(\text{OID1}, \text{account102})\}$$

นั่นก็คือ

$$\text{account}' = \{\text{OID1}\} \wedge \text{accountAttrs}' = \{(\text{OID1}, \text{account102})\}$$

- การทดสอบทฤษฎีบทของ AddAccount

Theorem TwoAccountsOfAddingAreNotSame

$$\forall \text{TestAdd2Account} \bullet \text{account2?} \neq \text{account?}$$

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ AddAccount คือ account?  $\notin$  dom accountAttrs นั่นคือตัวแปรเข้า account? และ account2? ต้องไม่เท่ากัน

- การทดสอบทฤษฎีบทของ RemoveAccount

Theorem TwoAccountOfRemovingAreSame

$$\forall \text{TestAddThenRemoveAccount} \bullet \text{account2?} = \text{account?}$$

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ RemoveAccount คือ  $account? \in \text{dom accountAttrs} \wedge \text{dom accountAttrs} \setminus \{account?\} \in \mathbb{P}(\text{account} \setminus \{account?\})$  นั่นคือตัวแปรเข้า account? และ account2? ต้องเท่ากัน

- การทดสอบทฤษฎีบทของ ChangeAccount

Theorem TwoAccountsOfChangingAreSame

$$\forall \text{TestChangeAccount} \bullet \text{account2?} = \text{account?}$$

เมื่อใช้คำสั่ง prove by reduce ใน Z/EVES ผลลัพธ์ที่ได้จะมีค่าเป็น true ทั้งนี้เป็นผลมาจากเงื่อนไขก่อนของ ChangeAccount คือ  $account? \in \text{dom accountAttrs}$  นั่นคือตัวแปรเข้า account? และ account2? ต้องเท่ากัน