

การออกแบบและพัฒนาระบบการนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอล



นาย รั้วเขต หาญชนะ

สถาบันวิทยบริการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมศาสตร์


คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2544

ISBN 974-03-1450-3

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

DESIGN AND DEVELOPMENT OF A REPRESENTATION  
FOR Z FORMAL SPECIFICATION WITH XML



Mr. Rathakate Hanchana

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

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 2001  
ISBN 974-03-1450-3

หัวข้อวิทยานิพนธ์      การออกแบบและพัฒนาระบบการนำเสนอข้อกำหนดรูปนัยภาษาเขต  
   ด้วยเอ็กซ์เอ็มแอล

โดย                              นาย รัฐเขต หาญชนะ

สาขาวิชา                    วิทยาศาสตร์คอมพิวเตอร์

อาจารย์ที่ปรึกษา            ผู้ช่วยศาสตราจารย์ วิวัฒน์ วัฒนาวุฒิ

---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยาลัย  
เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์  
( ศาสตราจารย์ ดร.สมศักดิ์ ปัญญาแก้ว )

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ  
( ผู้ช่วยศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี )

..... อาจารย์ที่ปรึกษา  
( ผู้ช่วยศาสตราจารย์ วิวัฒน์ วัฒนาวุฒิ )

..... กรรมการ  
( อาจารย์ ดร.ธราทิพย์ สุวรรณศาสตร์ )

..... กรรมการ  
( อาจารย์ ดร.ทวิตีย์ เสนีวงศ์ ณ อยุธยา )

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

รัฐเขต ชาญชนะ : การออกแบบและพัฒนาระบบการนำเสนอข้อกำหนดรูปนัยภาษาเซต  
ด้วยเอ็กซ์เอ็มแอล.(DESIGN AND DEVELOPMENT OF A REPRESENTATION FOR Z  
FORMAL SPECIFICATION WITH XML) อ.ที่ปรึกษา:ผู้ช่วยศาสตราจารย์ วิวัฒน์ วัฒนาวุฒิ,  
145 หน้า. ISBN 974-03-1450-3.

วิทยานิพนธ์นี้ได้ออกแบบและพัฒนาระบบการนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอกสาร  
เอ็กซ์เอ็มแอล ซึ่งประกอบด้วยการออกแบบและพัฒนา โปรแกรมประยุกต์ของเอ็กซ์เอ็มแอลในรูปแบบ  
ของเอกสาร 3 ประเภทคือ เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล ทำหน้าที่กำหนดและตรวจสอบโครงสร้าง  
ของข้อกำหนดรูปนัยภาษาเซตที่เขียนด้วยไวยากรณ์เอ็กซ์เอ็มแอลให้อยู่ในรูปแบบที่ถูกต้องสมบูรณ์  
เอกสารเซตเอ็กซ์เอ็มแอลที่ดี ทำหน้าที่ประกาศข้อกำหนด การอ้างอิงถึงสัญลักษณ์ทางคณิตศาสตร์ที่  
มีการใช้ในข้อกำหนดรูปนัยภาษาเซตกับชุดอักขระยูนิโคด และเอกสารเอ็กซ์เอ็มแอลสไตล์ชีท ทำหน้าที่  
แจ้งส่วนไวยากรณ์ภาษาเซต และแปลงข้อกำหนดรูปนัยภาษาเซตในรูปของเอกสารเอ็กซ์เอ็มแอลเป็น  
เอกสารเอชทีเอ็มแอล เพื่อที่จะสามารถแสดงผล ข้อกำหนดรูปนัยภาษาเซตในรูปแบบดังกล่าวบนเว็บ  
เบราว์เซอร์ ได้อย่างถูกต้อง

อนึ่ง ระบบการนำเสนอข้อกำหนดรูปนัยภาษาเซต ด้วยเอ็กซ์เอ็มแอลที่พัฒนาขึ้นนี้ ได้รับการ  
ทดสอบกับกรณีศึกษา และผลลัพธ์ได้รับการยืนยันว่าสามารถแสดงผลข้อกำหนดรูปนัยภาษาเซตอย่าง  
ถูกต้อง

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมคอมพิวเตอร์  
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์  
ปีการศึกษา 2544

ลายมือชื่อนิสิต.....  
ลายมือชื่ออาจารย์ที่ปรึกษา.....  
ลายมือชื่ออาจารย์ที่ปรึกษาร่วม.....

## 4371473221 : MAJOR COMPUTER SCIENCE

KEY WORD: Z NOTATION / FORMAL SPECIFICATION / Z/XML SCHEMA / Z/DTD / Z/XSL

RATHAKATE HANCHANA : DESIGN AND DEVELOPMENT OF A REPRESENTATION  
FOR Z FORMAL SPECIFICATION WITH XML. THESIS ADVISOR : ASSISTANT  
PROFESSOR WIWAT VATANAWOOD,145 pp. ISBN 974-03-1450-3.

This thesis designs and develops a representation for Z specification by using XML document. The system is developed and constructed from three XML application documents as follows: Z/XML schema document - to specify and validate the structure of Z specification in well-formed XML document, Z/XML DTD (Document Type Definitions) - to specify the set of mathematical symbols used within Z specification and to reference Unicode characters, and Z/XSL (XML Stylesheet Language) - to parse the Z syntax and transform the XML document to HTML document for displaying on the World Wide Web browser. The system is tested and the results are correctly verified by using the case study.



Department Computer Engineering  
Field of study Computer Science  
Academic year 2001

Student's signature.....  
Advisor's signature.....  
Co-advisor's signature.....

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ได้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของผู้ช่วยศาสตราจารย์ วิวัฒน์ วัฒนาวุฒิ อาจารย์ที่ปรึกษาที่ให้คำแนะนำ ในระหว่างขั้นตอนการพัฒนางานวิจัยตลอดมา และขอขอบคุณ ผู้ช่วยศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี อาจารย์ ดร.ธราทิพย์ สุวรรณศาสตร์ และ อาจารย์ ดร.ทวีติย์ เสนีวงศ์ ณ อยุธยา กรรมการวิทยานิพนธ์ที่กรุณาเสียสละเวลาให้คำแนะนำ ตรวจสอบและแก้ไขต้นฉบับวิทยานิพนธ์

ขอขอบคุณ เพื่อน ๆ กลุ่ม Formal ในห้องปฏิบัติการวิศวกรรมซอฟต์แวร์ที่เสียสละเวลาในการให้คำปรึกษา ช่วยตรวจสอบผลการวิจัย ตลอดจนให้กำลังใจและข้อเสนอแนะต่าง ๆ และ ขอขอบคุณท่านอื่น ๆ ที่มีส่วนช่วยในการทำวิทยานิพนธ์ที่ไม่ได้กล่าวนามมา ณ โอกาสนี้ด้วย

สุดท้ายนี้ ผู้วิจัยใคร่ขอกราบขอบพระคุณ บิดา มารดาที่สนับสนุนในด้านต่าง ๆ และให้กำลังใจแก่ผู้วิจัยเสมอมา

รัฐเขต หาญชนะ

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## สารบัญ

หน้า

บทคัดย่อภาษาไทย .....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ .....	ช
สารบัญตาราง.....	ฌ
สารบัญรูปภาพ .....	ญ
บทที่ 1 บทนำ.....	1
1.1    ความเป็นมาและความสำคัญของปัญหา .....	1
1.2    วัตถุประสงค์ของการวิจัย .....	2
1.3    ขอบเขตของการวิจัย .....	2
1.4    วิธีดำเนินการวิจัย .....	2
1.5    ประโยชน์ที่จะได้รับ.....	3
บทที่ 2 งานวิจัยและทฤษฎีที่เกี่ยวข้อง .....	4
2.1    งานวิจัยที่เกี่ยวข้อง .....	4
2.2    ทฤษฎีที่เกี่ยวข้อง.....	6
บทที่ 3 กระบวนการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล.....	15
3.1    เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (Z/XML Schema) .....	16
3.2    เอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (Z/XML DTD) .....	23
3.3    การเขียนข้อกำหนดรูปร่างภาษาเซตด้วยวากยสัมพันธ์เอ็กซ์เอ็มแอล .....	26
3.4    การแสดงผลข้อกำหนดรูปร่างภาษาเซตเอ็กซ์เอ็มแอลด้วยเอกสารเซตเอ็กซ์เอสแอล สไตล์ชีท (Z/XSL Style Sheet) .....	33
บทที่ 4 การทดสอบระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล.....	38
4.1    สภาวะที่ใช้ทดสอบระบบ.....	38

## สารบัญ(ต่อ)

หน้า

4.2	หัวข้อการทดสอบระบบ.....	38
4.3	กรณีศึกษาที่ใช้ทดสอบระบบ.....	39
4.4	สรุปผลการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอลจากกรณีทดสอบ	46
บทที่ 5 สรุปผลการวิจัย และข้อเสนอแนะ..... 48		
5.1	สรุปผลการวิจัย.....	48
5.2	ประโยชน์ของระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล.....	49
5.3	ข้อเสนอแนะ.....	49
รายการอ้างอิง.....		50
ภาคผนวก.....		52
ภาคผนวก ก. เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....		53
ภาคผนวก ข. เอกสารเซตเอ็กซ์เอ็มแอลที่ดีที่สุด.....		75
ภาคผนวก ค. เอกสารเซตเอ็กซ์เอ็มแอลสไตลชีท.....		85
ภาคผนวก ง. กรณีศึกษาระบบ BirthdayBook.....		98
ภาคผนวก จ. กรณีศึกษาระบบการรายงานข้อผิดพลาด(Error Handling).....		114
ภาคผนวก ฉ. กรณีศึกษาระบบการนับจำนวนคำ(Word Counting).....		129
ภาคผนวก ช. กรณีศึกษาเพิ่มเติมระบบร้านเช่าวีดีโอ(Video Shop Hiring System).....		143
ประวัติผู้เขียนวิทยานิพนธ์.....		145



## สารบัญตาราง

หน้า

ตารางที่ 2.1 ตัวอย่างการกำหนดชื่อของเอนทิตีที่ใช้ในการเขียนเค้าร่างภาษาเซต.....	11
ตารางที่ 3.1 ตัวอย่างการกำหนดชื่อเอนทิตีกับชุดอักขระยูนิโคด.....	23
ตารางที่ 4.1 สรุปผลการนำเสนอข้อกำหนดรูปนัยภาษาเซตจากกรณีศึกษาด้วย เอ็กซ์เอ็มแอลบนเว็บเบราว์เซอร์.....	46
ตารางที่ ก-1 รายชื่อส่วนย่อยที่นิยามไว้ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	54
ตารางที่ ข-1 กลุ่มสัญลักษณ์การนิยาม (Definitions).....	75
ตารางที่ ข-2 กลุ่มสัญลักษณ์ทางตรรกะ (Logic).....	75
ตารางที่ ข-3 กลุ่มสัญลักษณ์เซตและนิพจน์ (Sets and Expressions).....	76
ตารางที่ ข-4 กลุ่มสัญลักษณ์ความสัมพันธ์ (Relations).....	76
ตารางที่ ข-5 กลุ่มสัญลักษณ์ฟังก์ชัน (Functions).....	77
ตารางที่ ข-6 กลุ่มสัญลักษณ์จำนวน (Number).....	77
ตารางที่ ข-7 กลุ่มสัญลักษณ์แบ็กส์ (Bags).....	78
ตารางที่ ข-8 กลุ่มสัญลักษณ์สัญญนิยม (Conventions).....	78
ตารางที่ ข-9 กลุ่มสัญลักษณ์ลำดับ (Sequence).....	78
ตารางที่ ข-10 กลุ่มสัญลักษณ์ เค้าร่างและทั่วไป (Schemas and Miscellaneous).....	79

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## สารบัญรูปภาพ

หน้า

รูปที่ 2.1	เค้าร่างแบบแนวตั้ง.....	7
รูปที่ 2.2	เค้าร่างแบบสี่พจน์.....	8
รูปที่ 2.3	เค้าร่างแบบเจนเนอริค.....	8
รูปที่ 2.4	การเขียนเค้าร่างแบบตามขวาง.....	8
รูปที่ 2.5	ตัวอย่างเค้าร่างการประกาศตัวแปร.....	9
รูปที่ 2.6	การประกาศข้อกำหนดส่วนย่อย (Element) ในเอกสารดีทีดี.....	10
รูปที่ 2.8	เค้าร่างภาษาเซต Street.....	11
รูปที่ 2.9	การเขียนเค้าร่าง Street ในรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต.....	12
รูปที่ 3.1	องค์ประกอบของระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล.....	15
รูปที่ 3.2	ตัวอย่างการนิยามประเภทของเค้าร่างภาษาเซตในเอกสารดีทีดี.....	17
รูปที่ 3.3	การนิยามประเภทของเค้าร่างภาษาเซตในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	17
รูปที่ 3.4	แผนภาพการนิยามประเภทของเค้าร่างในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	18
รูปที่ 3.5	การนิยามข้อกำหนดส่วนลักษณะประจำของเค้าร่างแบบเหมือนกล่อง.....	19
รูปที่ 3.6	การนิยามข้อกำหนดลำดับโครงสร้างข้อกำหนดรูปร่าง.....	19
รูปที่ 3.7	ข้อกำหนดรูปแบบตำแหน่งการจัดวางข้อกำหนดรูปร่าง.....	20
รูปที่ 3.8	แผนภาพแสดงโครงสร้างของส่วนย่อยในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	21
รูปที่ 3.9	แผนภาพแสดงโครงสร้างของส่วนย่อย decpart ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	22
รูปที่ 3.10	แผนภาพแสดงโครงสร้างของส่วนย่อย axpart ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	22
รูปที่ 3.11	การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคดในเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี.....	24
รูปที่ 3.12	ตัวอย่างการอ้างอิงเอนทิตีในเอกสารเอ็กซ์เอ็มแอล.....	24
รูปที่ 3.13	ตัวอย่างผลลัพธ์การอ้างอิงเอนทิตีบนเว็บเบราว์เซอร์.....	25
รูปที่ 3.14	ข้อกำหนดรูปร่างภาษาเซตเค้าร่าง AddBirthdayBook.....	26
รูปที่ 3.15	ตัวอย่างส่วนประกอบของเอกสารข้อกำหนดรูปร่างภาษาเซตที่ถูกต้อง.....	27
รูปที่ 3.16	รูปแบบการเขียนป้ายระบุเปิดและป้ายระบุปิด.....	28
รูปที่ 3.17	รากของส่วนย่อย Zedml ในการเขียนข้อกำหนดรูปร่างภาษาเซต.....	28
รูปที่ 3.18	ข้อกำหนดรูปร่างภาษาเซตที่มีเค้าร่างแบบสี่พจน์.....	29
รูปที่ 3.19	ลำดับโครงสร้างของ Axiom definition ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	29

## สารบัญรูปภาพ(ต่อ)

หน้า

รูปที่ 3.20 ลำดับโครงสร้างของ decpart ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล.....	30
รูปที่ 3.21 ข้อกำหนดรูปนัยแบบ Axiom schema ที่มีรูปแบบไวยากรณ์ที่ต้องสมบูรณ์ .....	31
รูปที่ 3.22 ขั้นตอนการทำงานของเอกสารเซตเอ็กซ์เอ็มแอลไต้ลชีท .....	33
รูปที่ 3.23 ตัวอย่างลักษณะข้อมูลจากการแปลงเค้าร่าง BirthdayBook ในรูปแบบภูมิต้นไม้.....	34
รูปที่ 3.24 แผ่นแบบการแสดงผลเค้าร่างกีเวนเซตในเอกสารเซตเอ็กซ์เอ็มแอลไต้ลชีท .....	35
รูปที่ 3.25 การเปรียบเทียบเค้าร่าง AddBirthdaybook กับแผ่นแบบและเอกสารเอชทีเอ็มแอล .	37
รูปที่ 3.26 การระบุแหล่งอ้างอิงเอกสารเซตเอ็กซ์เอ็มแอลไต้ลชีทที่ส่วนโปรล็อก.....	37
รูปที่ 4.1 การนิยามข้อมูลแบบกีเวนเซตในระบบ BirthdayBook.....	39
รูปที่ 4.2 เค้าร่างข้อกำหนดรูปนัย BirthdayBook.....	39
รูปที่ 4.3 การแสดงผลเค้าร่างข้อกำหนดรูปนัย BirthdayBook บนเว็บเบราว์เซอร์ .....	40
รูปที่ 4.4 การแจ่งส่วน เค้าร่าง BirthdayBook จากเอกสารเอ็กซ์เอ็มแอล.....	41
รูปที่ 4.5 ข้อกำหนดรูปนัยโครงร่างเหมือนกล่องแบบเจนเนอริค .....	42
รูปที่ 4.6 การแสดงผลเค้าร่างแบบเจนเนอริคบนเว็บเบราว์เซอร์.....	43
รูปที่ 4.7 ข้อกำหนดรูปนัยโครงร่างเหมือนกล่องแบบสัจพจน์.....	43
รูปที่ 4.8 การแสดงผลเค้าร่างแบบสัจพจน์บนเว็บเบราว์เซอร์ .....	44
รูปที่ 4.9 ข้อกำหนดรูปนัยการประกาศตัวแปรในระบบร้านเช่าวีดีโอ.....	45
รูปที่ 4.10 การแสดงผลเค้าร่างการประกาศตัวแปรบนเว็บเบราว์เซอร์ .....	45
รูปที่ ก-1 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย Abbrevdef .....	54
รูปที่ ก-2 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย Application.....	55
รูปที่ ก-3 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย Axdef.....	55
รูปที่ ก-4 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย Aupart .....	56
รูปที่ ก-5 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย body.....	57
รูปที่ ก-6 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย branch.....	57
รูปที่ ก-7 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย declaration .....	58
รูปที่ ก-8 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย declarations .....	58
รูปที่ ก-9 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย declname .....	59
รูปที่ ก-10 โครงสร้างและรหัสต้นฉบับของส่วนย่อยย declpart.....	60

## สารบัญรูปภาพ(ต่อ)

หน้า

รูปที่ ก-11 โครงสร้างและรหัสต้นฉบับของส่วนย่อย formal .....	60
รูปที่ ก-12 โครงสร้างและรหัสต้นฉบับของส่วนย่อย freetypedef.....	61
รูปที่ ก-13 โครงสร้างและรหัสต้นฉบับของส่วนย่อย gendef.....	62
รูปที่ ก-14 โครงสร้างและรหัสต้นฉบับของส่วนย่อย genparam .....	63
รูปที่ ก-15 โครงสร้างและรหัสต้นฉบับของส่วนย่อย givendef .....	63
รูปที่ ก-16 โครงสร้างและรหัสต้นฉบับของส่วนย่อย inclusion.....	64
รูปที่ ก-17 โครงสร้างและรหัสต้นฉบับของส่วนย่อย name .....	65
รูปที่ ก-19 โครงสร้างและรหัสต้นฉบับของส่วนย่อย note .....	66
รูปที่ ก-20 โครงสร้างและรหัสต้นฉบับของส่วนย่อย url .....	67
รูปที่ ก-21 โครงสร้างและรหัสต้นฉบับของส่วนย่อย opdec.....	67
รูปที่ ก-22 โครงสร้างและรหัสต้นฉบับของส่วนย่อย predicate.....	68
รูปที่ ก-23 โครงสร้างและรหัสต้นฉบับของส่วนย่อย schemadef .....	69
รูปที่ ก-24 โครงสร้างและรหัสต้นฉบับของส่วนย่อย sexp .....	71
รูปที่ ก-25 โครงสร้างและรหัสต้นฉบับของส่วนย่อย sub .....	71
รูปที่ ก-26 โครงสร้างและรหัสต้นฉบับของส่วนย่อย sup .....	72
รูปที่ ก-27 โครงสร้างและรหัสต้นฉบับของส่วนย่อย type .....	72
รูปที่ ก-28 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Zedml.....	73
รูปที่ ข-1 การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคดในเอกสารเซดเอ็กซ์เอ็มแอลที่ดี .....	81
รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอ็มแอลสไตล์ซีที .....	85
รูปที่ ง-1 รหัสต้นฉบับเค้าร่างสถานะ.....	98
รูปที่ ง-2 เค้าร่างสถานะ (State Schema) บนเว็บเบราว์เซอร์.....	99
รูปที่ ง-3 รหัสต้นฉบับเค้าร่าง AddBirthdayBook .....	99
รูปที่ ง-4 เค้าร่าง AddBirthdayBook บนเว็บเบราว์เซอร์.....	100
รูปที่ ง-5 รหัสต้นฉบับเค้าร่าง FindBirthdayBook.....	101
รูปที่ ง-6 เค้าร่าง FindBirthdayBook บนเว็บเบราว์เซอร์.....	102
รูปที่ ง-7 รหัสต้นฉบับเค้าร่าง InitBirthdayBook .....	102
รูปที่ ง-8 เค้าร่าง InitBirthdayBook บนเว็บเบราว์เซอร์.....	103

## สารบัญรูปภาพ(ต่อ)

หน้า

รูปที่ ง-9 รหัสต้นฉบับเค้าร่าง AddThenFindBirthdayBook .....	103
รูปที่ ง-10 เค้าร่าง AddThenFindBirthdayBook บนเว็บเบราว์เซอร์ .....	106
รูปที่ ง-11 รหัสต้นฉบับเค้าร่าง BirthdayBook1 .....	107
รูปที่ ง-12 เค้าร่าง BirthdayBook1 บนเว็บเบราว์เซอร์ .....	108
รูปที่ ง-13 รหัสต้นฉบับเค้าร่าง AddBirthdayBook1 .....	108
รูปที่ ง-14 เค้าร่าง BirthdayBook1 บนเว็บเบราว์เซอร์ .....	109
รูปที่ ง-15 รหัสต้นฉบับเค้าร่าง Success .....	110
รูปที่ ง-16 เค้าร่าง Success บนเว็บเบราว์เซอร์ .....	111
รูปที่ ง-17 รหัสต้นฉบับเค้าร่าง Remind .....	112
รูปที่ ง-18 เค้าร่าง Remind บนเว็บเบราว์เซอร์ .....	112
รูปที่ ง-19 รหัสต้นฉบับเค้าร่าง InitBirthdayBook โครงร่างแบบตามขวาง .....	113
รูปที่ ง-20 เค้าร่าง InitBirthdayBook โครงร่างแบบตามขวางบนเว็บเบราว์เซอร์ .....	114
รูปที่ จ-1 รหัสต้นฉบับเค้าร่าง การนิยามข้อมูลชนิดฟรีไทป์ .....	114
รูปที่ จ-2 เค้าร่างการนิยามข้อมูลชนิดฟรีไทป์บนเว็บเบราว์เซอร์ .....	115
รูปที่ จ-3 รหัสต้นฉบับเค้าร่าง BoundedStackMem .....	115
รูปที่ จ-4 เค้าร่าง BoundedStackMem บนเว็บเบราว์เซอร์ .....	116
รูปที่ จ-5 รหัสต้นฉบับเค้าร่าง BoundedStackMemOk .....	117
รูปที่ จ-6 เค้าร่าง BoundedStackMemOk บนเว็บเบราว์เซอร์ .....	118
รูปที่ จ-7 รหัสต้นฉบับเค้าร่าง PushOk .....	118
รูปที่ จ-8 เค้าร่าง PushOk บนเว็บเบราว์เซอร์ .....	119
รูปที่ จ-9 รหัสต้นฉบับเค้าร่าง PopOk .....	120
รูปที่ จ-10 เค้าร่าง PopOk บนเว็บเบราว์เซอร์ .....	121
รูปที่ จ-11 รหัสต้นฉบับเค้าร่าง TopOk .....	121
รูปที่ จ-12 เค้าร่าง TopOk บนเว็บเบราว์เซอร์ .....	122
รูปที่ จ-13 รหัสต้นฉบับเค้าร่าง OverFlow .....	122
รูปที่ จ-14 เค้าร่าง OverFlow บนเว็บเบราว์เซอร์ .....	123
รูปที่ จ-15 รหัสต้นฉบับเค้าร่าง UnderFlow .....	124

## สารบัญรูปภาพ(ต่อ)

หน้า

รูปที่ จ-16	เค้าร่าง UnderFlow บนเว็บเบราว์เซอร์	125
รูปที่ จ-17	รหัสต้นฉบับเค้าร่าง PushMemOk	125
รูปที่ จ-18	เค้าร่าง PushMemOk บนเว็บเบราว์เซอร์	126
รูปที่ จ-19	รหัสต้นฉบับเค้าร่าง Error	127
รูปที่ จ-20	เค้าร่าง Error บนเว็บเบราว์เซอร์	128
รูปที่ ฉ-1	รหัสต้นฉบับเค้าร่างการนิยามข้อมูลชนิดตัวย่อ	129
รูปที่ ฉ-2	เค้าร่างการนิยามข้อมูลชนิดตัวย่อบนเว็บเบราว์เซอร์	130
รูปที่ ฉ-3	รหัสต้นฉบับเค้าร่าง WordCount1	130
รูปที่ ฉ-4	เค้าร่าง WordCount1 บนเว็บเบราว์เซอร์	131
รูปที่ ฉ-5	รหัสต้นฉบับเค้าร่าง wordCount2	131
รูปที่ ฉ-6	เค้าร่าง WordCount2 บนเว็บเบราว์เซอร์	133
รูปที่ ฉ-7	รหัสต้นฉบับเค้าร่าง WordSeq	133
รูปที่ ฉ-8	เค้าร่าง WordSeq บนเว็บเบราว์เซอร์	134
รูปที่ ฉ-9	รหัสต้นฉบับเค้าร่าง wordFreq	134
รูปที่ ฉ-10	เค้าร่าง WordFreq บนเว็บเบราว์เซอร์	135
รูปที่ ฉ-11	รหัสต้นฉบับเค้าร่าง wordCount3	136
รูปที่ ฉ-12	เค้าร่าง WordCount3 บนเว็บเบราว์เซอร์	137
รูปที่ ฉ-13	รหัสต้นฉบับเค้าร่าง numToStr	137
รูปที่ ฉ-14	เค้าร่าง numToStr บนเว็บเบราว์เซอร์	138
รูปที่ ฉ-15	รหัสต้นฉบับเค้าร่าง uhead	138
รูปที่ ฉ-16	เค้าร่าง uhead บนเว็บเบราว์เซอร์	139
รูปที่ ฉ-17	รหัสต้นฉบับเค้าร่าง uhead1 (lambda-expression)	140
รูปที่ ฉ-18	เค้าร่าง uhead1 (lambda-expression) บนเว็บเบราว์เซอร์	140
รูปที่ ฉ-19	รหัสต้นฉบับเค้าร่าง tr_c	141
รูปที่ ฉ-20	เค้าร่าง tr_c บนเว็บเบราว์เซอร์	142
รูปที่ ซ-1	รหัสต้นฉบับเค้าร่างการประกาศตัวแปร	143
รูปที่ ซ-2	เค้าร่างการประกาศตัวแปรบนเว็บเบราว์เซอร์	144

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ภาษาเซต (Z) [1], [2], [4] เป็นภาษาที่ใช้ในการทำข้อกำหนดรูปร่าง (Formal specification) ซึ่งเป็นวิธีการหนึ่งในการทำ ข้อกำหนดความต้องการ (Requirements specification) ที่นำไปใช้ในกระบวนการพัฒนาซอฟต์แวร์ (Software) โดยภาษาเซตจะทำการแยกส่วนของข้อกำหนดรูปร่างที่แทนกลไกของสถานะต่างๆของซอฟต์แวร์ด้วยทฤษฎีทางคณิตศาสตร์ให้เป็นส่วนประกอบย่อยๆที่เรียกว่าเค้าร่าง (Schema) เพื่อที่จะช่วยให้การสร้างข้อกำหนดรูปร่างนั้นทำได้ง่ายขึ้น

ข้อกำหนดรูปร่างภาษาเซตนั้น สามารถทำการสับเปลี่ยน(Interchange) ข้อมูลบนเครือข่ายคอมพิวเตอร์ได้ ด้วยการสร้างเค้าร่างภาษาเซต ตามรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต (ZIF:Z Interchange Format ) [3] ซึ่งมีโครงสร้างของเค้าร่างตามสัญกรณ์เซต (Z Notation) และมีรูปแบบของวากยสัมพันธ์ (Syntax) ตามมาตรฐานของภาษาเอสจีเอ็มแอล (SGML: Standard Generalized Markup Language) [11] แต่เนื่องจากโครงสร้างของภาษาเอสจีเอ็มแอล มีรูปแบบการใช้วากยสัมพันธ์ที่เป็นลักษณะเฉพาะ ทำให้ยากในการเขียนและไม่สามารถแทน (Represent) เค้าร่างภาษาเซตให้อยู่ในรูปแบบที่เข้าใจได้ง่าย ทำให้ไม่ได้รับความนิยมที่จะนำมาใช้งาน จึงมีความพยายามที่จะหาวิธีการนำเสนอเค้าร่างภาษาเซต ด้วยวิธีการอื่นๆ ที่มีความสะดวกในการเขียนและเข้าใจได้ง่ายกว่า เช่นการแปลง(Transform) ข้อกำหนดรูปร่างภาษาเซตให้เป็น ภาษาเอสทีเอ็มแอล (HTML: Hypertext Markup Language) เพื่อแสดงผลบนเว็บเบราว์เซอร์ (Web Browser) แต่วิธีการนี้จะมีข้อจำกัดก็คือ ผลลัพธ์ที่ได้นั้นจะไม่สามารถทำการสับเปลี่ยนข้อมูลตามที่ได้ระบุไว้ในข้อกำหนดของรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต

จากข้อจำกัดดังกล่าวในงานวิทยานิพนธ์นี้จึงต้องการพัฒนา ระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซต ด้วยเอ็กซ์เอ็มแอล (XML: Extensible Markup Language)[10] ซึ่งเป็นโปรแกรมประยุกต์ (Application) ประเภทหนึ่งของภาษาเอสจีเอ็มแอล ประกอบด้วยเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (Z/XML Schema) เอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (Z/XML DTD) และเอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีท (Z/XSL Style Sheet) เพื่อให้ผู้ใช้งานสามารถสร้างข้อกำหนดรูปร่างภาษาเซตด้วยภาษาเอ็กซ์เอ็มแอลได้อย่างถูกต้อง โดยสามารถทำการตรวจสอบโครงสร้างของเค้าร่างภาษาเซต ให้ครอบคลุมตามมาตรฐานของรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต พร้อมทั้งนี้ยังสามารถนำเสนอข้อกำหนดรูปร่างภาษาเซตบนเว็บเบราว์เซอร์ได้อีกด้วย

## 1.2 วัตถุประสงค์ของการวิจัย

- 1.2.1 เพื่อออกแบบและพัฒนาเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (Z/XML Schema)
- 1.2.2 เพื่อออกแบบและพัฒนาเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (Z/XML DTD)
- 1.2.3 เพื่อออกแบบและพัฒนาเอกสารเซตเอ็กซ์เอสแอลสไตล์ชีท (Z/XSL Style Sheet) สำหรับแสดงผลเค้าร่างภาษาเซตบนเว็บเบราว์เซอร์

## 1.3 ขอบเขตของการวิจัย

- 1.3.1 การนำเสนอข้อกำหนดรูปร่างภาษาเซตสามารถนำเสนอด้วยเว็บเบราว์เซอร์ที่สนับสนุนมาตรฐานเอ็กซ์เอ็มแอล 1.0 (XML: Extensible Markup Language 1.0) และได้รับการติดตั้งชุดแบบอักษรยูนิโคด (Unicode font)
- 1.3.2 เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (Z/XML Schema) และเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (Z/XML DTD) จะครอบคลุมข้อกำหนดรูปแบบการสืบเปลี่ยนข้อมูลภาษาเซต (ZIF: Z Interchange Format)
- 1.3.3 เอกสารเอ็กซ์เอ็มแอลสำหรับภาษาเซตที่ได้จะเป็นไปตามมาตรฐานเอ็กซ์เอ็มแอล 1.0
- 1.3.4 เอกสารเซตเอ็กซ์เอสแอลสไตล์ชีท ที่ใช้ในการแจงส่วน (Parse) เค้าร่างเอ็กซ์เอ็มแอลสำหรับภาษาเซต จะเป็นไปตามข้อกำหนดของภาษาเอ็กซ์เอสแอลสไตล์ชีท (XSL: Extensible Style Sheet Language)
- 1.3.5 ใช้กรณีศึกษา 3 กรณี เพื่อทดสอบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล

## 1.4 วิธีดำเนินการวิจัย

- 1.4.1 ศึกษาและวิเคราะห์งานวิจัยต่าง ๆ ที่เกี่ยวข้อง
- 1.4.2 ศึกษาการใช้งานภาษาเซต ได้แก่ รูปแบบการใช้วากยสัมพันธ์ ของภาษาเซต การใช้เครื่องหมายและสัญลักษณ์ ต่างๆ ที่ใช้เขียนข้อกำหนดรูปร่างภาษาเซต
- 1.4.3 ศึกษาวิธีการสร้างข้อกำหนดจากเค้าร่างภาษาเซตที่ระบุไว้ในรูปแบบการสืบเปลี่ยนข้อมูลภาษาเซต



- 1.4.4 ศึกษาวิธีการสร้างเอกสารเค้าร่างเอ็กซ์เอ็มแอลและเอกสารเพิ่มข้อมูลคำอธิบายชนิดของเอกสาร ตามมาตรฐานของ World Wide Web Consortium [W3C]
- 1.4.5 ออกแบบขั้นตอนวิธีการนำเค้าร่างเอ็กซ์เอ็มแอลสำหรับภาษาเซตแสดงผลบนเว็บเบราว์เซอร์ ด้วยเอกสารเอ็กซ์เอสแอลสไตล์ชีท
- 1.4.6 พัฒนาโปรแกรม
- 1.4.7 ทดสอบโปรแกรม
- 1.4.8 สรุปและจัดทำรายงานการวิจัย

## 1.5 ประโยชน์ที่จะได้รับ

- 1.5.1 ได้เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (Z/XML Schema) เอกสารเซตเอ็กซ์เอ็มแอล ดีทีดี (Z/XML DTD) และเอกสารเซตเอ็กซ์เอสแอลสไตล์ชีท (XSL Style Sheet) ที่เป็นองค์ประกอบช่วยให้ระบบสามารถนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอลและแสดงผลบนเว็บเบราว์เซอร์ เป็นไปอย่างถูกต้องสมบูรณ์
- 1.5.2 ได้กระบวนการนำเสนอข้อกำหนดรูปร่างด้วยเอ็กซ์เอ็มแอลที่สามารถปรับเปลี่ยนข้อมูลของข้อกำหนดรูปร่างภาษาเซต ในรูปแบบที่เข้าใจได้ง่ายและใช้งานได้สะดวกกว่ารูปแบบการปรับเปลี่ยนข้อมูลภาษาเซตแบบเดิม โดยที่ยังคงสามารถปรับเปลี่ยนข้อมูลของเค้าร่างภาษาเซตได้ตามข้อกำหนดการปรับเปลี่ยนข้อมูลภาษาเซต

## บทที่ 2

### งานวิจัยและทฤษฎีที่เกี่ยวข้อง

#### 2.1 งานวิจัยที่เกี่ยวข้อง

2.1.1 การทดลองด้วยรูปแบบการสับเปลี่ยนข้อมูลภาษาเซตและ เอสจีเอ็มแอล (Experiments with the Z Interchange Format and SGML) [6] โดย Daniel M. German และ D.D. Cowan

งานวิจัยนี้เป็นการนำเสนอผลการทดลองการสร้างรูปแบบการสับเปลี่ยนข้อมูลข้อกำหนดรูปนัยภาษาเซต (Z Interchange Format) ซึ่งได้ให้คำจำกัดความรูปแบบการสับเปลี่ยนข้อมูลภาษาเซตไว้ว่า เป็นวิธีการแทนข้อกำหนดรูปนัยภาษาเซต ให้สามารถส่งผ่าน (Transmit) ระหว่างเครือข่ายคอมพิวเตอร์ได้ โดยมีรูปแบบของการแทนเค้าร่างภาษาเซตด้วยวากยสัมพันธ์ภาษาเอสจีเอ็มแอล และได้อธิบายถึงความแตกต่างระหว่าง รูปแบบการสับเปลี่ยนข้อมูลภาษาเซต กับรูปแบบมาตรฐานของภาษาเซต ไว้ดังนี้

2.1.1.1 รูปแบบการสับเปลี่ยนข้อมูลภาษาเซตไม่สามารถแสดงไวยากรณ์ (Grammar) ทั้งหมดของภาษาเซต แต่จะทำหน้าที่ในการตรวจสอบลำดับ (Sequence) ของตัวระบุ (Identifier) และตัวดำเนินการ (Operator) ในภาษาเซต โดยใช้วิธีตรวจสอบดังกล่าวจากการนิยามชนิดเอกสารหรือดีทีดี (DTD: Document Type Definition) และใช้วากยสัมพันธ์ภาษาเอสจีเอ็มแอลเป็นมาตรฐานในการสร้างรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต

2.1.1.2 ภาษาเซตมาตรฐานไม่เพียงแต่จะระบุลำดับของตัวระบุและตัวดำเนินการเท่านั้น แต่ยังรวมไปถึงข้อกำหนด (Specification) การเชื่อม (Join) ระหว่างตัวระบุและตัวดำเนินการนั้นด้วย

จากงานวิจัยนี้ช่วยให้เห็นถึงข้อกำหนดพื้นฐานในการสร้างรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต ที่จำเป็น เช่น วิธีการอ้างอิงถึงสัญลักษณ์ทางคณิตศาสตร์จากเอกสารดีทีดี หรือการตรวจสอบลำดับของเค้าร่างภาษาเซตจากเอกสารดีทีดี เป็นต้น

2.1.2 การสร้างภาพนามธรรมสัญกรณ์เซต ในเอกสารเอชทีเอ็มแอล (Visualizing Z notation in HTML document) [5] โดย Paolo Ciancarini, Cecilia Mascolo และ Fabio Vitali

งานวิจัยนี้ได้เสนอวิธีการนำเค้าร่างภาษาเซต ในรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต ที่มีวากยสัมพันธ์เป็นภาษาเอสซีเอ็มแอล มาแสดงผลในรูปแบบของเอกสารเอชทีเอ็มแอล (HTML: Hypertext Markup Language) บนเว็บเบราว์เซอร์ โดยมีขั้นตอนดังต่อไปนี้

2.1.2.1 สร้างโปรแกรมประยุกต์ด้วยจาวาแอปเพลท (JAVA Applet)[7] เพื่อทำหน้าที่ในการตรวจสอบไวยากรณ์และโครงสร้างของเอกสารเอสซีเอ็มแอล ให้ถูกต้องตามข้อกำหนดไวยากรณ์และโครงสร้างของภาษาเซตโดยตรวจสอบจากเอกสารดีทีดี ที่ได้กำหนดโครงสร้างของเค้าร่างภาษาเซต ที่ถูกต้องไว้แล้ว หลังจากตรวจสอบความสมเหตุสมผล (Validate) แล้วจะได้ผลลัพธ์เป็นเอกสารเอสซีเอ็มแอล ที่ถูกต้อง (Valid) ตามหลักไวยากรณ์ของภาษาเซต

2.1.2.2 สร้างโปรแกรมประยุกต์ด้วย จาวาแอปเพลท อีกหนึ่งโปรแกรม เรียกว่า “เซตทูเอชทีเอ็มแอล” (Zed2Html) ที่ทำหน้าที่นำเอกสารเอสซีเอ็มแอล ที่ถูกต้องจากขั้นตอนแรกนำมาแปลง (Transform) รูปแบบจากเอกสารเอสซีเอ็มแอลไปเป็นเอกสารเอชทีเอ็มแอล อีกครั้งหนึ่งเพื่อที่จะนำไปแสดงผลบนเว็บเบราว์เซอร์ ต่อไป

จากงานวิจัยนี้ช่วยให้ทราบถึงขั้นตอนการแสดงผลของเค้าร่างภาษาเซตบนโปรแกรมค้นผ่านเว็บได้อย่างถูกต้องตามรูปแบบของภาษาเซต แต่ยังมีข้อจำกัดในด้านที่จะต้องผ่านขั้นตอนที่ยุ้งยากในวิธีการแปลงเค้าร่างภาษาเซต ในชุดเอกสารเอสซีเอ็มแอล ไปเป็นเอกสารเอชทีเอ็มแอล และในประเด็นสำคัญคือผลลัพธ์ที่ได้จะไม่สามารถตรวจสอบรูปแบบการสับเปลี่ยนข้อมูลได้ตามข้อกำหนดของรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต เพราะโครงสร้างทั้งหมดถูกแปลงให้อยู่ในรูปแบบของเอกสารเอชทีเอ็มแอล

2.1.3 กลุ่มของภาษาเซตบนเว็บด้วยภาษายูเอ็มแอล (Z Family on the Web with Their UML Photos) [7] โดย J. Sun, J.S. Dong, J. Lui และ H. Wang

งานวิจัยนี้แสดงให้เห็นถึงขั้นตอนวิธีการนำกลุ่มของภาษาเซตที่ประกอบด้วย ภาษาเซต และ ภาษาเซตเชิงวัตถุ(Object Z)มาแสดงผลบนเว็บเบราว์เซอร์โดยการสร้างเอกสารเค้าร่าง

เอ็กซ์เอ็มแอล (XML Schema) ของกลุ่มภาษาเซตดังกล่าวสำหรับการตรวจสอบโครงสร้าง และแสดงผลบนเว็บเบราว์เซอร์โดยจะสร้างโปรแกรมประยุกต์ของเอ็กซ์เอ็มแอล ขึ้นมาเพื่อทำหน้าที่เป็นตัวแจงส่วนเอ็กซ์เอ็มแอล (XML Parser) ให้ถูกต้องตามที่กำหนดไว้ในเค้าร่างเอ็กซ์เอ็มแอล และสามารถที่จะเปลี่ยนโครงสร้างของเค้าร่างเอ็กซ์เอ็มแอลสำหรับชุดของกลุ่มภาษาเซตที่ได้ ให้อยู่ในรูปของแผนภาพภาษายูเอ็มแอล (UML Diagram) ทั้งนี้พบว่าโครงสร้างของเค้าร่างเอ็กซ์เอ็มแอลสำหรับชุดของกลุ่มภาษาเซตดังกล่าวไม่ได้ถูกนิยาม (Definition) และให้ความหมายของส่วนย่อย (Element) ตามข้อกำหนดของ รูปแบบการสืบเปลี่ยนข้อมูลภาษาเซต จึงมีข้อผิดพลาดและคลาดเคลื่อนของข้อกำหนดในการใช้สืบเปลี่ยนข้อมูลผ่านเครือข่ายคอมพิวเตอร์

จากงานวิจัยทั้งสามที่กล่าวมาข้างต้นทำให้ทราบถึงแนวคิดพื้นฐานของข้อกำหนดต่างๆที่จำเป็นต่อการออกแบบและพัฒนาระบบนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอล เช่น ขั้นตอนวิธีการอ้างอิงถึงสัญลักษณ์ทางคณิตศาสตร์ที่ใช้ในการเขียนข้อกำหนดรูปนัยภาษาเซตจากเอกสารที่ดี การตรวจสอบลำดับของเค้าร่างภาษาเซตให้ครอบคลุมข้อกำหนด รูปแบบการสืบเปลี่ยนข้อมูล วิธีการแสดงผลข้อกำหนดรูปนัยภาษาเซตบนเว็บเบราว์เซอร์ที่ถูกต้อง และทำให้ทราบถึงสิ่งที่ต้องนิยามเพิ่มเติมเพื่อให้เกิดความสมบูรณ์ของระบบการนำเสนอรูปนัยภาษาเซต ด้วยเอ็กซ์เอ็มแอลต่อไป

## 2.2 ทฤษฎีที่เกี่ยวข้อง

ในงานวิทยานิพนธ์นี้ มีทฤษฎีต่าง ๆ ที่เกี่ยวข้องดังนี้

### 2.2.1 ข้อกำหนดรูปนัย (Formal Specification) [1]

เป็นการแสดงข้อกำหนดความต้องการในรูปของภาษาที่มีการอธิบายประมวลศัพท์ (Vocabulary) วากยสัมพันธ์ (Syntax) และความหมาย (Semantic) ไว้อย่างเป็นระเบียบ คือ มีลักษณะเป็นกลุ่มของตัวอักษรสัญลักษณ์ทางคณิตศาสตร์ ในรูปของตรรกศาสตร์ภาคแสดง ลำดับแรก (First order predicate logic) จึงทำให้ ความหมายที่ได้นั้นมีความชัดเจน และแน่นอน ช่วยลดความผิดพลาด (Error) ของการออกแบบระบบตั้งแต่การออกแบบในระยะเริ่มต้น สามารถพิสูจน์ความสอดคล้อง (Consistency) ของข้อกำหนดรูปนัยโดยใช้เครื่องมือทางคณิตศาสตร์ได้ แต่อย่างไรก็ตามทำข้อกำหนดรูปนัยพบว่ามีข้อจำกัดคือศึกษาให้เข้าใจได้ค่อนข้างยาก เนื่องจากต้องใช้รูปแบบทางคณิตศาสตร์ในการเขียนข้อกำหนดรูปนัย

2.2.2 ภาษาเซต (Z Language) [2], [3],[4]

เป็นภาษาที่ใช้เขียนข้อกำหนดรูปถ่ายประเภทหนึ่ง ซึ่งได้รับการพัฒนาขึ้นเมื่อประมาณปีคริสต์ศักราช 1970 โดยกลุ่ม พีอาร์จี (PRG: Programming Research Group) ที่มหาวิทยาลัยอ็อกซ์ฟอร์ด (Oxford University) โดยภาษาเซตได้สร้างข้อกำหนดสัญกรณ์เซต (Z Notation) เป็นข้อกำหนดที่อาศัยพื้นฐานของทฤษฎีเกี่ยวกับเซต และ ตรรกศาสตร์ภาคแสดงลำดับแรกเพื่อช่วยในการเขียนข้อกำหนดรูปถ่าย โดยแบ่งข้อกำหนดรูปถ่ายออกเป็น ส่วนประกอบเล็กๆ เรียกว่าสัญกรณ์เค้าร่าง (Schema notation) ทำให้ข้อกำหนดรูปถ่ายมีโครงสร้างที่ให้ง่ายต่อการเข้าใจ

โครงสร้างของเค้าร่างภาษาเซต สามารถแบ่งได้ 2 แบบคือ เค้าร่างสถานะ (State schema) ซึ่งเป็นการประกาศถึง คอมโพเนนต์ (Component) ต่างๆ ในระบบ และ เค้าร่างการปฏิบัติการ (Operation Schema) ซึ่งเป็นการประกาศถึง การปฏิบัติการ (Operation) ต่าง ๆ ในระบบ โดยเค้าร่างสามารถแบ่งออกเป็น 2 ส่วนคือ

2.2.2.1 ส่วนการประกาศ (Declaration Part) เป็นส่วนประกาศตัวแปร และชนิดตัวแปรที่จะใช้

2.2.2.2 ส่วนสัจพจน์ (Axiom Part) เป็นส่วนที่ระบุถึงภาคแสดงซึ่งอ้างอิงถึงตัวแปรที่ได้นิยามไว้ หรือ ตัวแปรอื่นที่อยู่ในขอบเขต

การเขียนเค้าร่างของภาษาเซต สามารถเขียนได้หลายแบบ ได้แก่

2.2.2.3 การเขียนเค้าร่างแบบ โครงร่างเหมือนกล่อง (Box-Like Framework) เป็นเค้าร่างที่มีลักษณะเป็นกล่องโดยแยกส่วน ส่วนการประกาศ และ ส่วนสัจพจน์ เขียนอธิบายแยกส่วนกันในกล่อง การเขียนเค้าร่างแบบโครงร่างเหมือนกล่องนี้ยังมีวิธีการเขียนที่แตกต่างกันอีก 3 วิธี ดังต่อไปนี้

2.2.2.3.1 เค้าร่างแบบแนวตั้ง (Vertical Schema) มีลักษณะดังรูปที่ 2.1

```

ú SchemaName
ú Declaration Part
ú
ú AxiomPart
ü
    
```

รูปที่ 2.1 เค้าร่างแบบแนวตั้ง

### 2.2.2.3.2 คำร่างแบบสัจพจน์ (Axiomatic Schema) มีลักษณะดังรูปที่ 2.2

*ú Declaration Part*  
*ú yyyyy*  
*ú AxiomPart*

รูปที่ 2.2 คำร่างแบบสัจพจน์

### 2.2.2.3.3 คำร่างแบบเจนเนอริค (Generic Schema) มีลักษณะดังรูปที่ 2.3

*ú y [Generic Parameter..] yyyyy*  
*ú Declaration Part*  
*ú yyyyy*  
*ú AxiomPart*  
*ú yyyyy*

รูปที่ 2.3 คำร่างแบบเจนเนอริค

### 2.2.2.4 การเขียนคำร่างแบบการเขียนคำร่างแบบตามขวาง (Horizontal) เป็นคำร่างที่มีลักษณะเป็นข้อความเขียนต่อเนื่องกันไป มีลักษณะดังรูปที่ 2.4

<i>BirthDayBook</i>		<i>[Known : Π</i>		<i>known = dom birthday</i>
		<i>NAME ;</i>		<i>]</i>

รูปที่ 2.4 การเขียนคำร่างแบบตามขวาง

นอกจากนี้ในการเขียนคำร่างภาษาเซตยังมีส่วนประกอบอื่นๆอีกดังต่อไปนี้

#### 2.2.2.5 การตกแต่ง (Decoration)

คือการใส่ตัวอักษรพิเศษท้ายชื่อของอ็อบเจกต์ (Object) เพื่อสื่อความหมายบางอย่างเพื่อให้เกิดความเข้าใจมากขึ้นเท่านั้น ตัวอักษรพิเศษที่ใช้ในมาตรฐานของสัญกรณ์เซต ได้แก่

- ? ใช้ต่อท้ายชื่อของตัวแปรเข้า (Input variable)
- ! ใช้ต่อท้ายชื่อของตัวแปรออก (Output variable)
- ⊃ ใช้ต่อท้ายชื่อของตัวแปรที่เป็นสถานะหลังการเปลี่ยนแปลง (After state)



### 2.2.3 รูปแบบการสับเปลี่ยนข้อมูลภาษาเซต (Z Interchange Format) [3]

เป็นวิธีการแทนข้อกำหนดรูปนัยภาษาเซตให้สามารถส่งผ่าน(Transmit) ระหว่างเครือข่ายคอมพิวเตอร์ได้ โดยมีรูปแบบของการแทนเค้าร่างภาษาเซตด้วย วากยสัมพันธ์ภาษาเอสจีเอ็มแอล รูปแบบการสับเปลี่ยนข้อมูลภาษาเซตนั้น สนับสนุนการสร้างข้อกำหนดมาตรฐานของสัญกรณ์เค้าร่าง (Schema notation) ทั้ง 2 แบบคือ เค้าร่างสถานะ (State schema) และเค้าร่างการปฏิบัติการ (Operation Schema) ซึ่งเป็นลักษณะเฉพาะที่กำหนดไว้ในภาษาเซต โดยใช้วิธีการเข้ารหัส (Encode) โครงสร้างของภาษาเซตในรูปแบบของเอกสาร ซึ่งจะครอบคลุมรายละเอียดต่างๆดังต่อไปนี้

2.2.3.1 ข้อกำหนดส่วนประกอบต่างๆของโครงสร้างสำหรับส่วนย่อย (Element) ในภาษาเซตเพื่อที่จะสามารถทำการตรวจสอบลำดับ (Sequence) ของตัวระบุ (Identifier) และตัวดำเนินการ (Operator) ในข้อกำหนดรูปนัยภาษาเซต โดยการประกาศไว้ในรูปแบบของ เอกสารคำอธิบายชนิดของเอกสารหรือที่เรียกว่าเอกสารดีทีดี ซึ่งเป็นข้อกำหนดมาตรฐานที่ใช้ในภาษาเอสจีเอ็มแอลดังตัวอย่างในรูปที่ 2.6

```
<!ELEMENT Z - -
(opdec | infundec | givendef | axdef | constraint | schemadef | gendef | abbrevdef |
conjecture | structsetdef)* >
<!ELEMENT schemadef (#PCDATA, sub? ,formal?, (sexp | (decpart| axpart?))) >
```

รูปที่ 2.6 การประกาศข้อกำหนดส่วนย่อย (Element) ในเอกสารดีทีดี

2.2.3.2 ข้อกำหนดส่วนที่เป็นลักษณะประจำ (Attribute) ในเอกสารดีทีดี ช่วยในการให้รายละเอียดข้อมูลเพิ่มเติมของส่วนย่อยในภาษาเซต ที่เขียนในรูปแบบการสับเปลี่ยนข้อมูล เช่น อธิบายลักษณะประจำ ของโครงสร้างของเค้าร่างในรูปแบบเหมือนกล่องแนวตั้ง (vert) หรือแบบตามขวาง (horiz) ดังตัวอย่างรูปที่ 2.7

```
<!ATTLIST Schemadef
style (vert | horiz) horiz purpose (state | operation | datatype) #IMPLIED>
```

รูปที่ 2.7 การประกาศข้อกำหนดลักษณะประจำ (Attribute) ในเอกสารดีทีดี



2.2.3.3 ข้อกำหนดส่วนที่เป็นเอนทิตี (Entity) หรือสัญลักษณ์ทางคณิตศาสตร์ต่างๆซึ่งจะถูกอ้างอิงในเอกสารเอสจีเอ็มแอลโดยกำหนดชื่อของเอนทิตีให้กับสัญลักษณ์ทางคณิตศาสตร์ที่ใช้ในเขียนเค้าร่างภาษาเซตดังตัวอย่างใน ตารางที่ 2.1

ตารางที่ 2.1 ตัวอย่างการกำหนดชื่อของเอนทิตีที่ใช้ในการเขียนเค้าร่างภาษาเซต

ชื่ออรูปนัย (Informal name)	ชื่อเอนทิตี (Entity name)	สัญลักษณ์ (Symbol)
Universal quantifier	Forall	$\forall$
Existential quantifier	Exist	$\exists$
Power set	Pset	$\Pi$
Membership	Isin	$\in$
Natural number	Nat	$\mathbb{N}$
Finite set	Fset	$\Phi$

จากรายละเอียดข้างต้นจะพบว่าเราสามารถสร้างข้อกำหนดอรูปนัยภาษาเซตตามข้อกำหนดของรูปแบบการสืบเปลี่ยนข้อมูลภาษาเซตซึ่งมีวากยสัมพันธ์เป็นภาษาเอสจีเอ็มแอลได้ ยกตัวอย่างเช่นผู้ใช้งานต้องการเขียนเค้าร่างภาษาเซต Street ในรูปที่ 2.8 ให้อยู่ในรูปแบบการสืบเปลี่ยนข้อมูลภาษาเซต

```

ú Street
ú inhabits : PERSON x HOUSE
ú house : HOUSE
ú
ú house = ran inhabits
ú

```

รูปที่ 2.8 เค้าร่างภาษาเซต Street

จากตัวอย่างเค้าร่างภาษาเซต Street ในรูปที่ 2.8 เมื่อนำมาเขียนในรูปแบบการสืบเปลี่ยนข้อมูลภาษาเซตที่มีวากยสัมพันธ์แบบภาษาเอสจีเอ็มแอล จะมีรายละเอียดของเค้าร่างแสดงดังตัวอย่างในรูปที่ 2.9

```

<z>
<schemadef style=vert purpose=state>Street
<decpart>
inhabits:PERSON &pfun HOUSE
house: &pset HOUSE
<axpart>
<Predicate>house= &ran inhabits
</schemadef>
</z>

```

รูปที่ 2.9 การเขียนเค้าร่าง Street ในรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต

#### 2.2.4 ภาษาเอ็กซ์เอ็มแอล eXtensible Markup Language (XML) [11]

เอ็กซ์เอ็มแอล เป็นภาษาที่ถูกออกแบบมาเพื่อใช้สำหรับอธิบายข้อมูล การจัดเก็บ และจัดส่งข้อมูลบนเวปไซต์ไวด์เว็บ ซึ่งมีพื้นฐานที่พัฒนามาจากภาษาเอสซีเอ็มแอล โดยทำการลดทอนไวยากรณ์ที่มีความซับซ้อนของภาษาเอสซีเอ็มแอล และสร้างข้อกำหนดของไวยากรณ์ภาษาเอ็กซ์เอ็มแอลที่ง่ายต่อการเขียน และสามารถประมวลผลได้ทั้งจากโปรแกรมประยุกต์และบุคคลทั่วไป เอกสารเอ็กซ์เอ็มแอลจะมีการอธิบายข้อมูลในรูปแบบที่เป็นลำดับชั้น (Hierarchy) เหมือนต้นไม้ (Tree) ทำให้สามารถสร้างโปรแกรมประยุกต์เพื่อทำการประมวลผลเอกสารและต่อประสาน (Binding) กับเอกสารเอชทีเอ็มแอลในการแสดงผลบนเว็บเบราว์เซอร์ได้อย่างอิสระ โดยทั่วไปเอ็กซ์เอ็มแอลจะประกอบด้วยสองส่วนหลักคือ

2.2.4.1 ตัวเอกสารเอ็กซ์เอ็มแอลซึ่งเป็นโครงสร้างทางตรรกะ (Logical Structure) อธิบายคุณลักษณะต่างๆของข้อมูลที่บรรจุอยู่ในเอกสารในรูปแบบลำดับชั้น

2.2.4.2 คำอธิบายชนิดของเอกสารหรือที่เรียกว่าดีทีดี (DTD: Document Type Declaration) ทำหน้าที่ในการกำหนดไวยากรณ์ของเอกสารเอ็กซ์เอ็มแอลที่มีรูปแบบถูกต้อง (Well-formed document) และสามารถทำให้เป็นเอกสารที่ถูกต้องสมบูรณ์ (Valid XML Document) ซึ่งเอกสารทั้งสองแบบนี้จะมีข้อบังคับที่แตกต่างกันคือ

- 2.2.4.2.1 ข้อบังคับของควมมีรูปแบบถูกต้อง (Well-formed Constraints) เป็นการกำหนดรายละเอียดพื้นฐานของไวยากรณ์ที่จำเป็น (Requirement) ซึ่งจะทำให้การฝ่าฝืนข้อบังคับนี้ไม่ได้ หากตัวแ่งส่วนของเอ็กซ์เอ็มแอล (XML Parser) พบการฝ่าฝืนข้อบังคับจะทำให้เกิดข้อผิดพลาดร้ายแรง (Fatal Error) และจะทำการหยุดการประมวลผลเอกสารทันที
- 2.2.4.2.2 ข้อบังคับของควมถูกต้องสมบูรณ์ (Validity Constraints) เป็นส่วนที่เพิ่มขึ้นมาของกฎในข้อกำหนดรายละเอียด (XML Specification) ซึ่งเป็นทางเลือก (Option) สำหรับเอ็กซ์เอ็มแอลการฝ่าฝืนข้อบังคับของควมถูกต้องสมบูรณ์จะทำให้เกิดข้อผิดพลาด (Error) ที่ตรงข้ามกับข้อผิดพลาดร้ายแรง โดยตัววิเคราะห์และประมวลผลในเว็บเบราว์เซอร์จะทำการแจ้งปัญหาให้ผู้ใช้งานทราบและพยายามแก้ปัญหาให้ซึ่งข้อกำหนดแบบนี้จะถูกระบุรายละเอียดไว้ในเอกสารที่ดี

นอกจากนี้เอ็กซ์เอ็มแอลยังมีการเพิ่มข้อกำหนดสำหรับโปรแกรมประยุกต์ เพื่อการพัฒนาเอกสารมีจุดประสงค์เพื่ออนุญาตให้ผู้ใช้เพิ่มการนิยามข้อกำหนดที่ต้องการได้ด้วยตนเองจากโปรแกรมประยุกต์ของเอ็กซ์เอ็มแอลประเภทต่างๆดังต่อไปนี้

#### 2.2.4.3 โปรแกรมประยุกต์ในด้านการแสดงผลเอกสาร เช่น

ภาษาเอ็กซ์เอ็มแอล (XSL: Extensible Style Sheet Language) [13] ถูกนำมาใช้แปลงเอกสารเอ็กซ์เอ็มแอลให้เป็นเอกสารเอชทีเอ็มแอล ด้วยการกำหนดข้อมูลที่ต้องการแสดงผลได้ทุกๆจุดต่อ (Node) ของเอกสารเอ็กซ์เอ็มแอลและจัดรูปแบบของเอกสารเอ็กซ์เอ็มแอลให้แสดงผลบนโปรแกรมค้นผ่านเว็บในรูปแบบของเอกสารเอชทีเอ็มแอล เช่นการกำหนดสีของวัตถุที่ต้องการให้แสดงผล การกำหนดขนาดของตัวอักษร เป็นต้น ซึ่งผู้ใช้ต้องสร้างเอกสารเอ็กซ์เอ็มแอลสไตล์ชีท (XSL Style Sheet) เพื่อทำหน้าที่ตามข้อกำหนดนี้โดยต้องใช้งานร่วมกับเอกสารเอ็กซ์เอ็มแอล ที่ต้องการแสดงผลด้วย

#### 2.2.4.4 โปรแกรมประยุกต์ในส่วนต่อประสานด้วยข้อมูล (Data Binding) เช่น

ข้อกำหนด ดีโอเอ็ม (DOM: Document Object Model) เพื่อการจัดการข้อมูลเอกสารเอ็กซ์เอ็มแอลให้อยู่ในรูปของวัตถุ (Object) ที่มีโครงสร้างเป็นลำดับชั้น

เหมือนต้นไม้ (Tree) ที่สามารถเขียนชุดคำสั่งเพื่อเข้าถึงตัวข้อมูล สามารถจัดการข้อมูล(Data manipulate) จัดเรียง(Sorting) กรองข้อมูล(Filter) ค้นหา(Search) และปรับปรุง (Update) ข้อมูลได้ นอกจากนี้ยังสามารถนำข้อมูลในรูปของวัตถุตั้งกล่าวมาแสดงผลในรูปของส่วนต่อประสานกราฟิกกับผู้ใช้ (Graphical User Interface) ได้อีกด้วย และข้อกำหนดดีไอเอ็มนี้สามารถสร้างด้วยโปรแกรมประยุกต์จากการเขียนสคริปต์(Script) เช่น จาวาสคริปต์(Java Script) หรือวีบีสคริปต์ (VBScript)

#### 2.2.4.5 โปรแกรมประยุกต์ในส่วนิยาม (Definition) เช่น

เค้าร่างเอ็กซ์เอ็มแอล (XML Schema) [12] เป็นส่วนเพิ่มขยายของคำอธิบายชนิดของเอกสารหรือดีทีดี แต่ใช้ไวยากรณ์ตามข้อบังคับของควมมีรูปแบบถูกต้องของเอกสารเอ็กซ์เอ็มแอลซึ่งทำหน้าที่เหมือนเอกสารดีทีดี ซึ่งปัจจุบันมีหลายองค์กรที่พัฒนาข้อกำหนดของเค้าร่างเอ็กซ์เอ็มแอลและประกาศเป็นมาตรฐานแล้วเช่น W3C XML Schema ของ World Wide Web Consortium [W3C] และ MS XML Schema ของ บริษัทไมโครซอฟท์ เป็นต้น ซึ่งสาเหตุที่เค้าร่างเอ็กซ์เอ็มแอลนี้ได้รับความนิยมนำมาใช้ทดแทนเอกสารดีทีดี เพราะมีความสะดวกในการใช้งานได้มากกว่าโดยเฉพาะในด้านการประกาศชนิดของข้อมูล (Data type) ที่ทำได้หลากหลายกว่าและที่สำคัญมีไวยากรณ์ที่ใช้เป็นมาตรฐานเดียวกันกับเอกสารเอ็กซ์เอ็มแอลซึ่งผู้ใช้งานจะมีความสะดวกกว่าการสร้างเอกสารดีทีดี

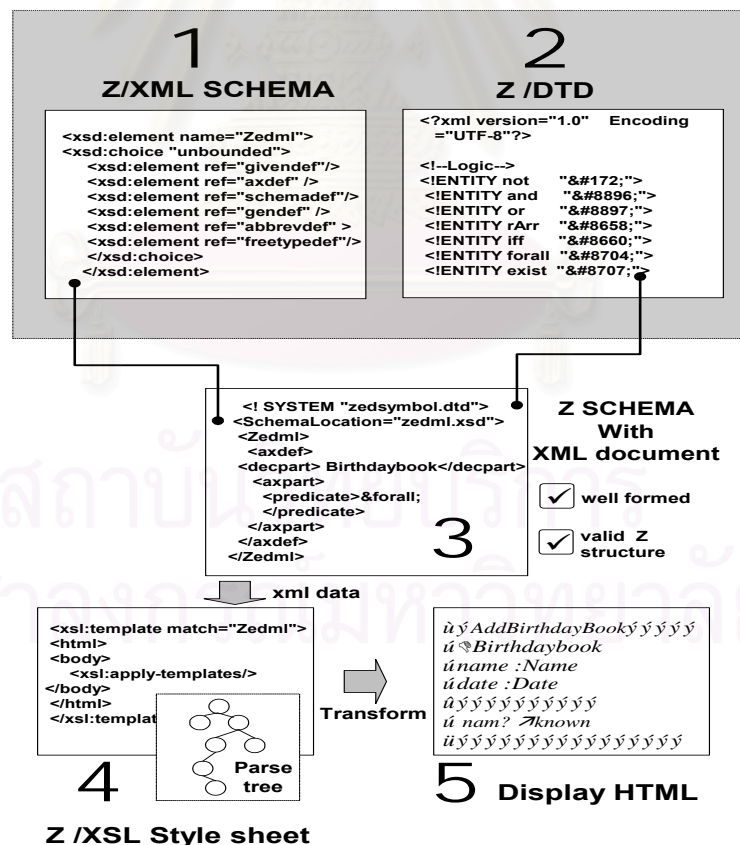
จากองค์ประกอบต่างๆข้างต้นทำให้เอ็กซ์เอ็มแอลนั้นเป็นเอกสารที่มีความเหมาะสมสำหรับนำมาใช้งานกับโปรแกรมประยุกต์ที่มีพื้นฐานบนเว็บ(Web) ด้วยการนำเสนอในรูปแบบข้อความ(Text) จึงไม่ขึ้นกับระบบปฏิบัติการ(Operating System) หรือสถาปัตยกรรมคอมพิวเตอร์ (Computer Architecture) และไวยากรณ์ของเอกสารเอ็กซ์เอ็มแอลนั้นยังสามารถแสดงคำอธิบายเชิงความหมายทำให้เข้าใจได้ง่ายอีกด้วย

ในขณะเดียวกันยังสามารถพัฒนาเอกสารที่อยู่ในวากยสัมพันธ์เอสจีเอ็มแอลให้เป็นเอกสารเอ็กซ์เอ็มแอลได้โดยมีความหมายและโครงสร้างของเอกสารเหมือนเดิมทุกประการ และสามารถแสดงผลเอกสาร บนเว็บเบราว์เซอร์ด้วยโปรแกรมประยุกต์ของเอ็กซ์เอ็มแอลเช่น เอ็กซ์เอสแอลสไตลชีท ตามวัตถุประสงค์ที่ต้องการได้ดังนั้นจึงมีความเป็นไปได้และมีความเหมาะสมที่จะสร้างระบบการนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอล โดยพัฒนาจากข้อกำหนดต่างๆที่ระบุไว้ในรูปแบบการสับเปลี่ยนข้อมูลภาษาเซตที่มีรูปแบบเป็นภาษาเอสจีเอ็มแอล

### บทที่ 3

## กระบวนการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล

ข้อกำหนดรูปร่างภาษาเซตมีลักษณะของโครงสร้างที่ประกอบด้วยเค้าร่างในรูปแบบที่แตกต่างกันหลายรูปแบบ เช่น การวางลำดับของตัวระบุและตัวดำเนินการที่มีการใช้ในเค้าร่าง ตลอดจนมาตรฐานของการกำหนดส่วนประกาศตัวแปรและส่วนสัจพจน์ ที่แยกจากกันอย่างชัดเจน เป็นต้น ดังนั้นในบทนี้จะอธิบายถึงกระบวนการนำเสนอข้อกำหนดรูปร่างด้วยเอ็กซ์เอ็มแอล โดยวิธีการอ้างอิงถึงข้อกำหนดพื้นฐานที่จำเป็นสำหรับการสร้างเค้าร่างภาษาเซต เช่นการตรวจสอบโครงสร้างของเค้าร่างภาษาเซตวิธีการประกาศตัวแปรต่างๆที่ผู้ใช้งานสร้างขึ้น หรือวิธีการอ้างอิงเอนทิตีที่เป็นสัญลักษณ์ทางคณิตศาสตร์ ตลอดจนกระบวนการสุดท้ายคือ การนำเสนอเค้าร่างภาษาเซตบนเว็บเบราว์เซอร์ ซึ่งในงานวิทยานิพนธ์นี้ได้แบ่งองค์ประกอบ ของระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอลออกเป็นส่วนๆ ดังรูปที่ 3.1



รูปที่ 3.1 องค์ประกอบของระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล

จากรูปที่ 3.1 องค์ประกอบของระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซต สามารถแบ่งได้เป็น 4 องค์ประกอบหลัก ซึ่งเป็นโปรแกรมประยุกต์ของเอ็กซ์เอ็มแอลที่อยู่ในรูปแบบเอกสารต่างๆ ดังต่อไปนี้ เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล(หมายเลข 1)ทำหน้าที่ในการนิยามรูปแบบของการเขียนข้อกำหนดรูปร่างด้วยเอ็กซ์เอ็มแอล ตรวจสอบโครงสร้างและลำดับของส่วนประกอบของข้อกำหนดรูปร่างที่ผู้ใช้งานสร้างขึ้น ลำดับต่อมาคือเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (หมายเลข 2) ทำหน้าที่นิยามชื่อของเอนทิตีหรือสัญลักษณ์ทางคณิตศาสตร์กับค่าของชุดอักขระยูนิโคด โดยเอกสารทั้งสองประเภทข้างต้นนี้จะต้องถูกอ้างอิงจากผู้ใช้งานโดยการเขียนระบุแหล่งอ้างอิงเอกสารทั้งสองในส่วนโปรล็อกของเอกสารเอ็กซ์เอ็มแอล(หมายเลข 3) จากการอ้างอิงเอกสารทั้งสองประเภทนั้นจะทำให้ข้อกำหนดรูปร่างที่เขียนในเอกสารเอ็กซ์เอ็มแอลได้รับการตรวจสอบรูปแบบโครงสร้างของข้อกำหนดรูปร่างภาษาเซต ตรวจสอบการอ้างอิงไปยังชื่อเอนทิตีเมื่อต้องการแสดงสัญลักษณ์ทางคณิตศาสตร์ ตลอดจนการตรวจสอบลำดับของส่วนย่อยต่างๆของภาษาเซตให้อยู่ในรูปแบบที่ถูกต้องตามข้อกำหนดของการเขียนเค้าร่างที่ระบุไว้ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล ซึ่งทั้งหมดจะเป็นการตรวจสอบโดยอัตโนมัติจากตัวแจนส่วนของเว็บเบราว์เซอร์

สำหรับการแสดงผลข้อกำหนดรูปร่างบนเว็บเบราว์เซอร์จะเป็นหน้าที่ของเอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีท(หมายเลข 4) ซึ่งเอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีทนี้จะทำหน้าที่แปลงเอกสารเอ็กซ์เอ็มแอลที่ได้แนบให้อยู่ในรูปแบบของเอกสารเอ็กซ์เอ็มแอล(หมายเลข 5) สำหรับรายละเอียดของเอกสารที่เป็นองค์ประกอบของระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอลที่กล่าวมานั้นมีดังต่อไปนี้

### 3.1 เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (Z/XML Schema)

เป็นโปรแกรมประยุกต์ประเภทหนึ่งของเอ็กซ์เอ็มแอลซึ่งในงานวิทยานิพนธ์นี้ ผู้วิจัยได้ออกแบบให้เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลทำหน้าที่ตรวจสอบความถูกต้องสมบูรณ์ (Validity Constraints) ของโครงสร้างข้อกำหนดรูปร่างภาษาเซต และในขณะเดียวกันยังสามารถตรวจสอบไวยากรณ์เอ็กซ์เอ็มแอลที่ใช้เขียนข้อกำหนดรูปร่างภาษาเซตให้มีรูปแบบถูกต้อง (Well-Formed) ตามข้อบังคับของควมมีรูปแบบถูกต้องของเอกสารเอ็กซ์เอ็มแอลที่ทาง Word Wide Web Consortium [W3C] ได้กำหนดไว้ โดยรายละเอียดของเอกสารเซตเอ็กซ์เอ็มแอลมีดังต่อไปนี้

#### 3.1.1 การพัฒนาเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลจากเอกสารดีทีดี

เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลจะมีวากยสัมพันธ์เป็นภาษาเอ็กซ์เอ็มแอลโดยผู้วิจัยพัฒนามาจากรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต ที่กำหนดโครงสร้างของเค้าร่างภาษาเซตไว้ในเอกสารดีทีดีซึ่งมีวากยสัมพันธ์ภาษาเอสซีเอ็มแอล โดยอ้างอิงจากมาตรฐานการสร้างเอกสาร

เค้าร่างเอ็กซ์เอ็มแอลของ W3C ยกตัวอย่างเช่น ใน รูปแบบการสับเปลี่ยนข้อมูลภาษาเซต ได้  
 นิยามประเภทของเค้าร่างที่ใช้ในการเขียนข้อกำหนดรูปร่างภาษาเซต โดยได้ระบุไว้ในเอกสาร  
 ดีทีดี ดังตัวอย่างในรูปที่ 3.2

```
<! ELEMENT Z - -
  ( givendef | axdef | schemadef | gendef | abbrevdef ) * >
```

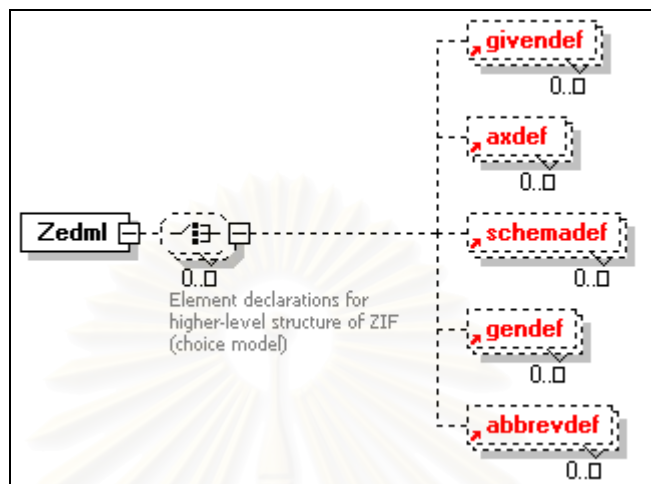
รูปที่ 3.2 ตัวอย่างการนิยามประเภทของเค้าร่างภาษาเซตในเอกสารดีทีดี

จากรูปที่ 3.2 สามารถอธิบายได้ว่าโครงสร้างของภาษาเซตจะประกอบไปด้วยส่วนย่อย  
 ต่างๆดังนี้ ส่วนย่อย givendef ส่วนย่อย axdef ส่วนย่อย schemadef ส่วนย่อย gendef และส่วน  
 ย่อย abbrevdef เมื่อผู้วิจัยได้พัฒนาโครงสร้างของเอกสารดีทีดีดังกล่าว โดยนำมาเขียนให้อยู่ใน  
 รูปของเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลตามข้อกำหนดที่กล่าวไว้ข้างต้นแล้ว จะมีลักษณะการ  
 นิยามประเภทของเค้าร่างภาษาเซตในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล ดังรูปที่ 3.3

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="Zedml">
    <xsd:complexType>
      <xsd:choice minOccurs="0" maxOccurs="unbounded">
        <xsd:element ref="givendef" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="axdef" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="schemadef" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="gendef" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="abbrevdef" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:choice>
    </xsd:complexType>
  </xsd:element>
```

รูปที่ 3.3 การนิยามประเภทของเค้าร่างภาษาเซตในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล

จากข้อกำหนดที่ระบุในรายละเอียดตามรูปที่ 3.3 สามารถแสดงเป็น แผนภาพที่อธิบายถึงลักษณะโครงสร้างของการนิยามประเภทของเค้าร่างภาษาเซต ในเอกสารเค้าร่างเอ็กซ์เอ็มแอล ดังรูปที่ 3.4



รูปที่ 3.4 แผนภาพการนิยามประเภทของเค้าร่างในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล

โครงสร้างข้อกำหนดรูปนัยภาษาเซตที่ผู้วิจัยได้นิยามไว้ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลนั้นจะครอบคลุมข้อกำหนดที่ได้ระบุไว้ในรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต โดยมีรายละเอียด ดังต่อไปนี้

- 3.1.2 ข้อกำหนดสำหรับส่วนย่อย (Element) ของภาษาเซตได้แก่การนิยามชนิดของข้อมูล ที่ใช้ในการเขียนข้อกำหนดรูปนัยภาษาเซต เช่น กิเวณเซต ตัวย่อ และการนิยามข้อมูลแบบฟรีไทป์ ตลอดจนโครงสร้างของเค้าร่างในแบบต่างๆ เช่น โครงสร้างแบบ schemadef และแบบ gendef เป็นต้น ดังตัวอย่างที่ได้แสดงไว้ในรูปที่ 3.3 ข้างต้น
- 3.1.3 ข้อกำหนดส่วนที่เป็นลักษณะประจำ (Attribute) จะบอกถึงรายละเอียดข้อมูลเพิ่มเติมของส่วนย่อยในภาษาเซต เช่น อธิบายลักษณะของโครงสร้างของเค้าร่างในรูปแบบเหมือนกล่อง ที่มีการเขียนได้ทั้งแบบแนวตั้ง (Vertical schema)(1) หรือแบบตามขวาง (Horizontal schema) (2) ดังตัวอย่างการนิยามข้อกำหนดส่วนที่เป็นลักษณะประจำ ในรูปที่ 3.5



```

<xsd:attribute name="style" use="required" default="vert">
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="vert"/> ●.....▶ (1)
      <xsd:enumeration value="horiz"/> ●.....▶ (2)
    </xsd:restriction>
  </xsd:simpleType>
</xsd:attribute>

```

รูปที่ 3.5 การนิยามข้อกำหนดส่วนลักษณะประจำของเค้าร่างแบบเหมือนกล่อง

- 3.1.4 ข้อกำหนดลำดับ(Sequence) โครงสร้างของข้อกำหนดรูปนัยที่ประกอบด้วยส่วนของการประกาศ (Declaration Part) (1) และ ส่วนสัจพจน์ (Axiom Part) (2) จำนวนครั้งที่โครงสร้างส่วนต่างๆสามารถปรากฏซ้ำ (Occur) ได้ ตลอดจนลำดับของตัวแปรชนิดต่างๆที่ต้องใช้ในการเขียนข้อกำหนดรูปนัย ลำดับโครงสร้างของข้อกำหนดรูปนัยดังกล่าวสามารถนิยามได้ดังตัวอย่างในรูปที่ 3.6

```

<xsd:sequence>
  <xsd:annotation>
    <xsd:documentation>
      Sequence model for the axiomatic definition
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element ref="decpart"/> ●.....▶ (1)
  <xsd:element ref="axpart" minOccurs="0"/> ●.....▶ (2)
</xsd:sequence>

```

รูปที่ 3.6 การนิยามข้อกำหนดลำดับโครงสร้างข้อกำหนดรูปนัย

นอกจากนี้ผู้ทำวิจัยยังได้เพิ่มข้อกำหนดวิธีการเขียนคำอธิบายประกอบและรูปแบบการจัดวาง ข้อกำหนดรูปนัย เช่น การจัดขีดซ้าย(1) กึ่งกลาง (2) หรือขีดขวา (3) เพื่อความเหมาะสมในการนำเสนอข้อกำหนดรูปนัยบนเว็บเบราว์เซอร์ดังตัวอย่างในรูปที่ 3.7

```

<xsd:attribute name="align" use="optional" default="left">
  <xsd:simpleType>
    <xsd:enumeration value="left"/> ●————▶ (1)
    <xsd:enumeration value="center"/> ●————▶ (2)
    <xsd:enumeration value="right"/> ●————▶ (3)
  </xsd:simpleType>
</xsd:attribute>

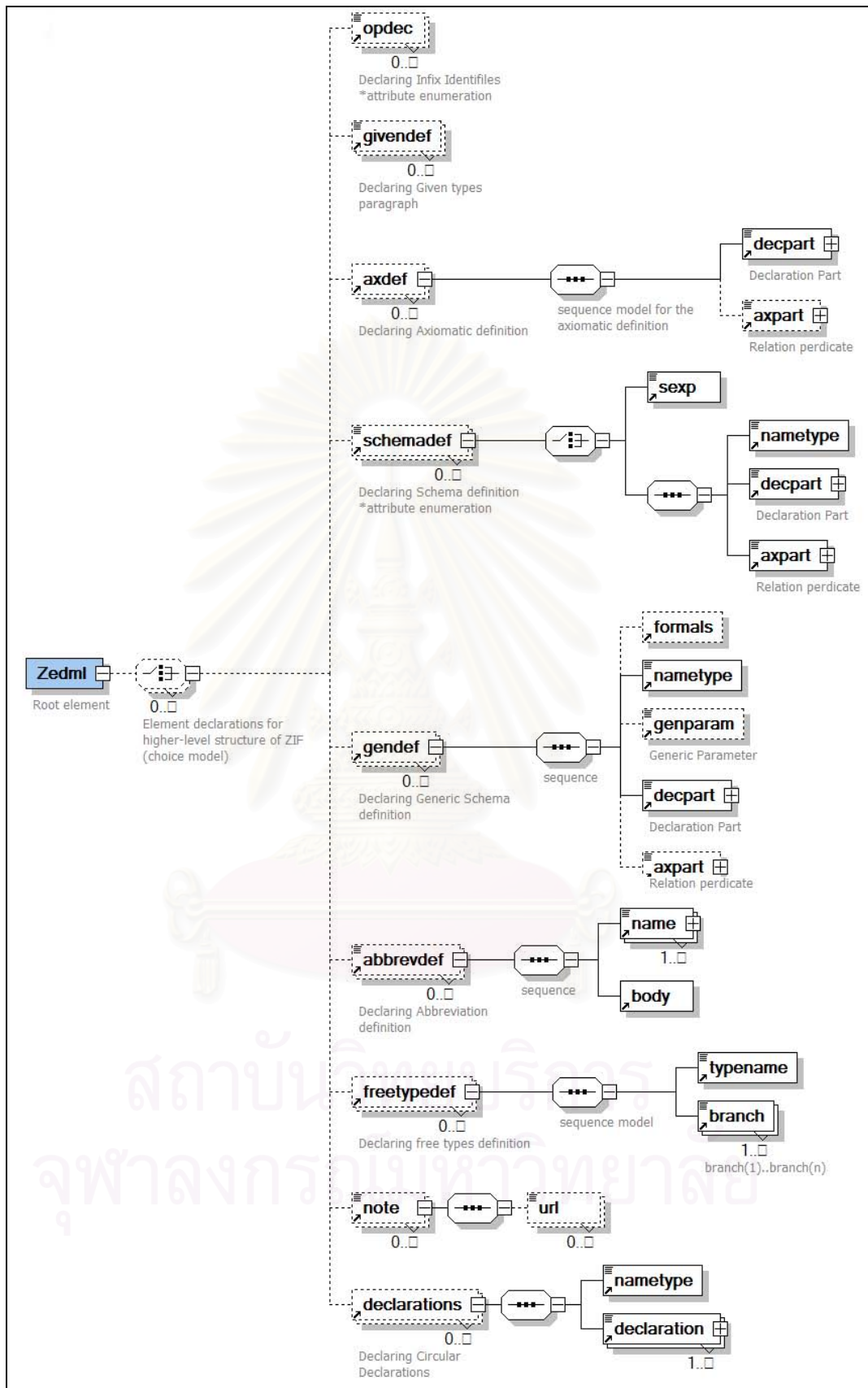
```

รูปที่ 3.7 ข้อกำหนดรูปแบบตำแหน่งการจัดวางข้อกำหนดรูปนัย

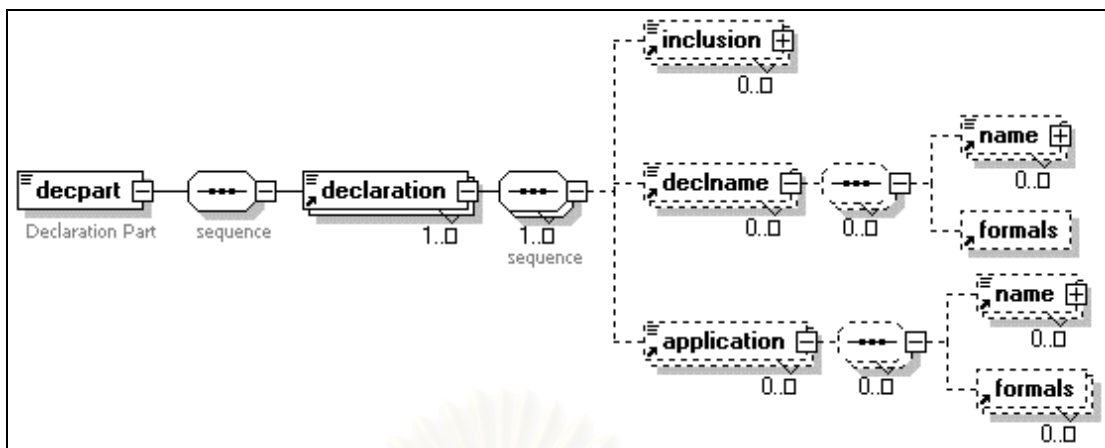
### 3.1.5 บีเอ็นเอฟ (BNF: Backus-Naur form) ของการเขียนข้อกำหนดรูปนัยภาษาเซตด้วย เอ็กซ์เอ็มแอล

รายละเอียดของข้อกำหนดต่างๆที่ผู้วิจัยได้นิยามไว้ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลนั้นถือเป็นรูปแบบบรรทัดฐานในการเขียนข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอล ซึ่งนอกจากจะนิยามองค์ประกอบของส่วนย่อยที่ต้องมีการใช้ในการเขียนข้อกำหนดรูปนัย ส่วนที่เป็นลักษณะประจำ รูปแบบและลักษณะของโครงสร้างข้อกำหนดรูปนัยภาษาเซตที่สามารถเขียนได้ด้วยวากยสัมพันธ์เอ็กซ์เอ็มแอลแล้ว ยังมีการนิยามองค์ประกอบของการนำเสนอข้อกำหนดรูปนัยที่ต้องแสดงผลบนเว็บเบราว์เซอร์ เช่นการจัดวางตำแหน่งของเค้าร่างบนหน้าจอเว็บเบราว์เซอร์ และการเพิ่มข้อความอธิบายรายละเอียดได้ เป็นต้น

เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลนี้สามารถอธิบายถึงภาพรวมของบีเอ็นเอฟในการเขียนข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอล ด้วยแผนภาพแสดงโครงสร้างของส่วนย่อยต่างๆที่ใช้ในการเขียนข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอลในรูปที่ 3.8 ซึ่งจะประกอบด้วย รากของส่วนย่อย Zedml และส่วนย่อยที่แทนรูปแบบการเขียนเค้าร่างข้อกำหนดรูปนัยภาษาเซตในแบบต่างๆ เช่น ส่วนย่อย givendef ส่วนย่อย schemadef ส่วนย่อย abbrevdeef และส่วนย่อย gendef เป็นต้น ส่วนรายละเอียดของส่วนย่อยอื่นๆที่สำคัญเช่นส่วนย่อย decpart ที่แทนรูปแบบการเขียนข้อกำหนดรูปนัยในส่วนการประกาศสามารถแสดงได้ดังแผนภาพในรูปที่ 3.9 และส่วนย่อย axpart ที่แทนรูปแบบการเขียนข้อกำหนดรูปนัยในส่วนสัจพจน์สามารถแสดงได้ดังแผนภาพในรูปที่ 3.10 สำหรับรายละเอียดของส่วนย่อย และรหัสต้นฉบับ(Source code) ของเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลทั้งหมดสามารถดูเพิ่มเติมได้ในภาคผนวก ก

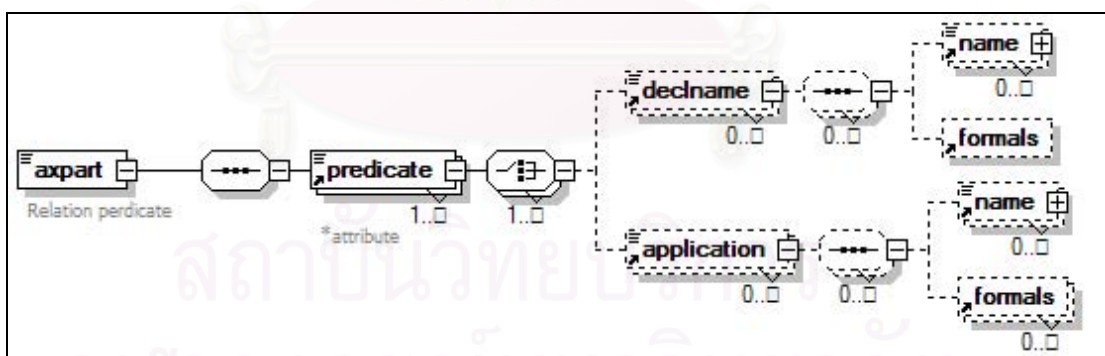


รูปที่ 3.8 แผนภาพแสดงโครงสร้างของส่วนย่อยในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล



รูปที่ 3.9 แผนภาพแสดงโครงสร้างของส่วนย่อย decpart ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล

จากรูปที่ 3.9 จะพบว่าส่วนย่อย decpart ซึ่งแทนข้อกำหนดการเขียนเค้าร่างภาษาเซตในส่วนการประกาศ ประกอบด้วยลำดับของส่วนย่อยอื่นๆอีกเช่น ส่วนย่อย declaration ส่วนย่อย inclusion ส่วนย่อย declname และส่วนย่อย application เป็นต้นซึ่งลำดับของส่วนย่อยเหล่านี้ก็จะเป็นข้อกำหนดของลำดับการเขียนเค้าร่างภาษาเซตที่ผู้ใช้งานต้องปฏิบัติตาม และในรูปที่ 3.10 ส่วนย่อย axpart ซึ่งแทนข้อกำหนดการเขียนเค้าร่างภาษาเซตในส่วนสัจพจน์จะประกอบด้วยส่วนย่อย predicate ส่วนย่อย declname และส่วนย่อย application เรียงกันมาตามลำดับ



รูปที่ 3.10 แผนภาพแสดงโครงสร้างของส่วนย่อย axpart ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล

### 3.2 เอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (Z/XML DTD)

ในงานวิทยานิพนธ์นี้ เอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (DTD: Document Type Declaration) จะทำหน้าที่ นิยามชื่อของเอนทิตี หรือสัญลักษณ์พิเศษทางคณิตศาสตร์ ที่ใช้ในการเขียนข้อกำหนดรูปนัยภาษาเซต ซึ่งมีการกำหนดไว้แล้วในข้อกำหนดรูปแบบการสับเปลี่ยนข้อมูลภาษาเซต โดยผู้ทำวิจัยได้ทำการนิยามเพิ่มเติมลงไปในส่วนของการอ้างอิงชื่อเอนทิตีเหล่านั้น กับชุดอักขระยูนิโคด (Unicode Character Set)[14] ซึ่งเป็นชุดอักขระที่รับรองมาตรฐานสากล โดยองค์การระหว่างประเทศว่าด้วยการมาตรฐานไอโซ (ISO: International Organization for Standardization) โดยค่าของชุดอักขระยูนิโคดแต่ละตัวจะแทนสัญลักษณ์พิเศษทางคณิตศาสตร์ ซึ่งผู้ใช้จะต้องอ้างอิงจากชื่อของเอนทิตีที่ได้นิยามไว้ในเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี

#### 3.2.1 รูปแบบการนิยามชื่อเอนทิตี ในเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี

การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคด ในเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี จะมีรูปแบบและวากยสัมพันธ์ตามข้อกำหนดของการเขียนเอกสารดีทีดี ที่ World Wide Web Consortium [W3C]กำหนดไว้ ดังต่อไปนี้

<!ENTITY ชื่อของเอนทิตี "ค่าของชุดอักขระยูนิโคด">

ในตารางที่ 3.1 จะเป็นการแสดงตัวอย่างการกำหนดชื่อเอนทิตีกับชุดอักขระยูนิโคด ซึ่งมีใช้ในการเขียนข้อกำหนดรูปนัย

ตารางที่ 3.1 ตัวอย่างการกำหนดชื่อเอนทิตีกับชุดอักขระยูนิโคด

ชื่ออรูปนัย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Equivalence	iff	$\Leftrightarrow$	&#8660;
Universal quantifier	forall	$\forall$	&#8704;
Existential quantifier	exist	$\exists$	&#8707;

สำหรับชุดอักขระยูนิโคดดังกล่าว เมื่อนำมานิยามในเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี แล้วจะมีรูปแบบดังรูปที่ 3.11

<!ENTITY iff "&#8660;">	●.....▶	Equivalence
<!ENTITY forall "&#8704;">	●.....▶	Universal quantifier
<!ENTITY exist "&#8707;">	●.....▶	Existential quantifier

รูปที่ 3.11 การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคดในเอกสารเซดเอ็กซ์เอ็มแอลดีทีดี

### 3.2.2 การอ้างอิงเอนทิตี (Entity Reference) กับการแสดงผลบนเว็บเบราว์เซอร์

เอนทิตีนั้นเปรียบเสมือนตัวแปรที่ใช้แทนค่าข้อความหรือสัญลักษณ์ใดๆ ที่ต้องการนำมาแสดงผลในเอกสารเอ็กซ์เอ็มแอล โดยเว็บเบราว์เซอร์จะทำการแจงส่วน (Parse) ผ่านโปรแกรมประยุกต์ที่ทำงานอยู่บนเว็บเบราว์เซอร์ ที่เรียกว่าตัวแจงส่วน (Parser) แล้วแสดงผลข้อความหรือสัญลักษณ์ตัวนั้น ซึ่งตัวแจงส่วนจะใช้อักขระกลุ่มหนึ่งเพื่อประเมินว่าส่วนใดในเอกสารคือเนื้อหา (Content) และส่วนใดคือส่วนมาร์คอัพ (Markup) ดังนั้นการอ้างอิงเอนทิตีก็คือการระบุให้ตัวแจงส่วนสามารถแยกได้ว่า ส่วนที่ทำการอ้างอิงชื่อเอนทิตีนั้นจะต้องแทนค่าข้อความหรือสัญลักษณ์ใดๆ ที่ถูกนิยามไว้ในเอกสารดีทีดี ซึ่งในงานวิทยานิพนธ์นี้ตัวแจงส่วนดังกล่าว คือเอกสารเซดเอ็กซ์เอ็มแอลดีทีดีนั่นเอง และการอ้างอิงเอนทิตีในเอกสารเอ็กซ์เอ็มแอลจะมีรูปแบบดังต่อไปนี้

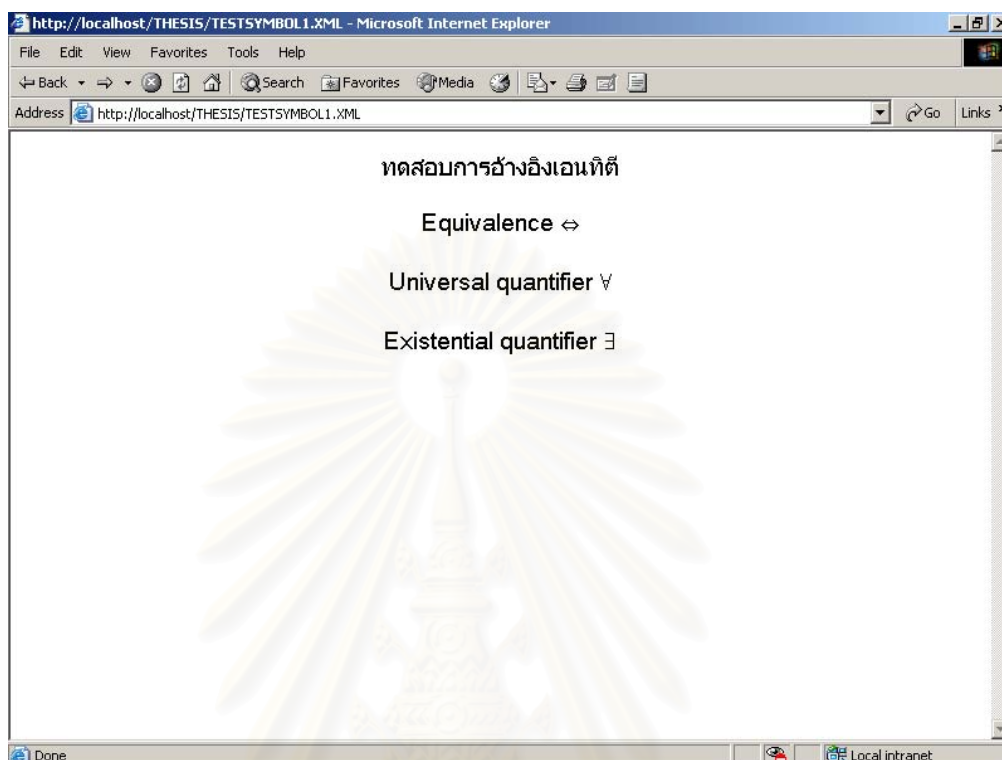
&ชื่อเอนทิตี;

ยกตัวอย่างเช่น ต้องการกำหนดให้เว็บเบราว์เซอร์แสดงผลสัญลักษณ์ทางคณิตศาสตร์ในตารางที่ 3.1 ซึ่งประกอบด้วยสัญลักษณ์  $\Leftrightarrow$  (2), A (3) และ E (4) ในเอกสารเอ็กซ์เอ็มแอลนั้น ต้องมีการระบุแหล่งอ้างอิงเอกสารเซดเอ็กซ์เอ็มแอลดีทีดี ที่ชื่อ zedsymbol.dtd (1) และ อ้างอิงเอนทิตี ตามตำแหน่งที่ต้องการแสดงผลสัญลักษณ์นั้น ดังตัวอย่างในรูปที่ 3.12

<? xml version="1.0" encoding="UTF-8"?>	
<! DOCTYPE Zedml SYSTEM "zedsymbol.dtd">	●.....▶ (1)
<Zedml>	
ทดสอบการอ้างอิงเอนทิตี	
Equivalence &iff;	●.....▶ (2)
Universal quantifier &forall;	●.....▶ (3)
Existential quantifier &exist;	●.....▶ (4)
</Zedml>	

รูปที่ 3.12 ตัวอย่างการอ้างอิงเอนทิตีในเอกสารเอ็กซ์เอ็มแอล

หลังจากเว็บเบราว์เซอร์ได้วิเคราะห์และแจกแจงส่วนเอกสารเอ็กซ์เอ็มแอลในตัวอย่างรูปที่ 3.12 แล้วจะได้ผลลัพธ์แสดงบนเว็บเบราว์เซอร์ดังรูปที่ 3.13



รูปที่ 3.13 ตัวอย่างผลลัพธ์การอ้างอิงเอนทิตีบนเว็บเบราว์เซอร์

จากตัวอย่างดังกล่าวจะพบว่าประโยชน์ของวิธีการอ้างอิงเอนทิตีนั้นนอกจากจะช่วยให้การแสดงผลข้อความหรือสัญลักษณ์พิเศษใดๆที่ต้องการ บนเว็บเบราว์เซอร์ได้อย่างสมบูรณ์แล้วนั้น ยังมีประโยชน์ในด้านที่ช่วยให้ผู้ใช้งานมีความสะดวกสบายมากยิ่งขึ้นเพราะไม่จำเป็นต้องทราบค่าของชุดอักขระยูนิโคดซึ่งเป็นข้อมูลตัวเลขที่จำได้ยาก เพียงทราบชื่อเอนทิตีที่อ้างอิงถึงชุดอักขระยูนิโคดนั้นๆก็จะสามารถสร้างข้อกำหนดรูปแบบภาษาเซตที่มีการใช้สัญลักษณ์ทางคณิตศาสตร์ได้ทันที สำหรับรายละเอียดของเอกสารเซตเอ็กซ์เอ็มแอลดีทีทีที่ใช้ในงานวิจัยนี้ สามารถดูเพิ่มเติมได้ในภาคผนวก ข

### 3.3 การเขียนข้อกำหนดรูปร่างภาษาเซตด้วยวากยสัมพันธ์เอ็กซ์เอ็มแอล

องค์ประกอบที่สำคัญอีกอย่างหนึ่งในระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอลคือการเขียนข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล ที่มีลักษณะเป็นเอกสารที่มีรูปแบบถูกต้อง (Well-formed) ซึ่งสามารถเข้าใจได้ง่าย แต่ก็มีกฎเกณฑ์ที่ต้องปฏิบัติตามอยู่ 2 ข้อคือ ข้อกำหนดด้านเอกสารที่มีรูปแบบถูกต้องของเอ็กซ์เอ็มแอล และข้อกำหนดเรื่องไวยากรณ์ที่ถูกต้องสมบูรณ์ (Valid Document) ที่ผู้วิจัยได้กำหนดขึ้น ซึ่งผู้ใช้งานจะต้องทราบและเข้าใจวิธีการใช้งานตามข้อกำหนดดังกล่าวอย่างถ่องแท้ จึงจะสามารถเขียนข้อกำหนดรูปร่างด้วยวากยสัมพันธ์เอ็กซ์เอ็มแอลอย่างถูกต้อง ดังต่อไปนี้

#### 3.3.1 ข้อกำหนดด้านเอกสารที่มีรูปแบบถูกต้องตามข้อกำหนดของภาษาเอ็กซ์เอ็มแอล

3.3.1.1 การเขียนข้อกำหนดรูปร่างด้วยเอ็กซ์เอ็มแอลที่มีรูปแบบที่ถูกต้องนั้นจะประกอบด้วยองค์ประกอบดังต่อไปนี้ คือ ส่วนที่ 1 เรียกว่าโปรล็อก (Prolog) เป็นส่วนที่จะต้องมีการระบุแหล่งอ้างอิงกระบวนการคำสั่งเครื่อง (Processing instruction) ที่อ้างอิงถึงเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (zedml.xsd) และเอกสารเซตเอ็กซ์เอ็มแอลที่ดี (zedsymbol.dtd) ซึ่งผู้ใช้จะต้องระบุให้ครบทั้งสองเอกสาร สำหรับส่วนที่ 2 คือส่วนของ เอกสารส่วนย่อย (Document Element) หรือ รากของส่วนย่อย (Root Element) จะประกอบด้วยส่วนย่อยของโครงสร้างข้อกำหนดรูปร่างที่ได้ระบุไว้ในเอกสารนอกเค้าร่างเซตเอ็กซ์เอ็มแอลเท่านั้น และในส่วนนี้ผู้ใช้งานยังสามารถอ้างอิงชื่อเอนทิตีเมื่อต้องการเขียนสัญลักษณ์ทางคณิตศาสตร์ ตลอดจนการเขียนหมายเหตุ (Comment) หรือบรรทัดว่างในข้อกำหนดรูปร่างได้ ยกตัวอย่างเช่นข้อกำหนดรูปร่างเค้าร่าง AddBirthdayBook ในรูปที่ 3.14

```

<?xml version="1.0" encoding="UTF-8"?>
<AddBirthdayBook>
  <name>Name</name>
  <date>Date</date>
  <known?>known</known?>
  <birthday? = birthday {name? date?}</birthday?>
</AddBirthdayBook>

```

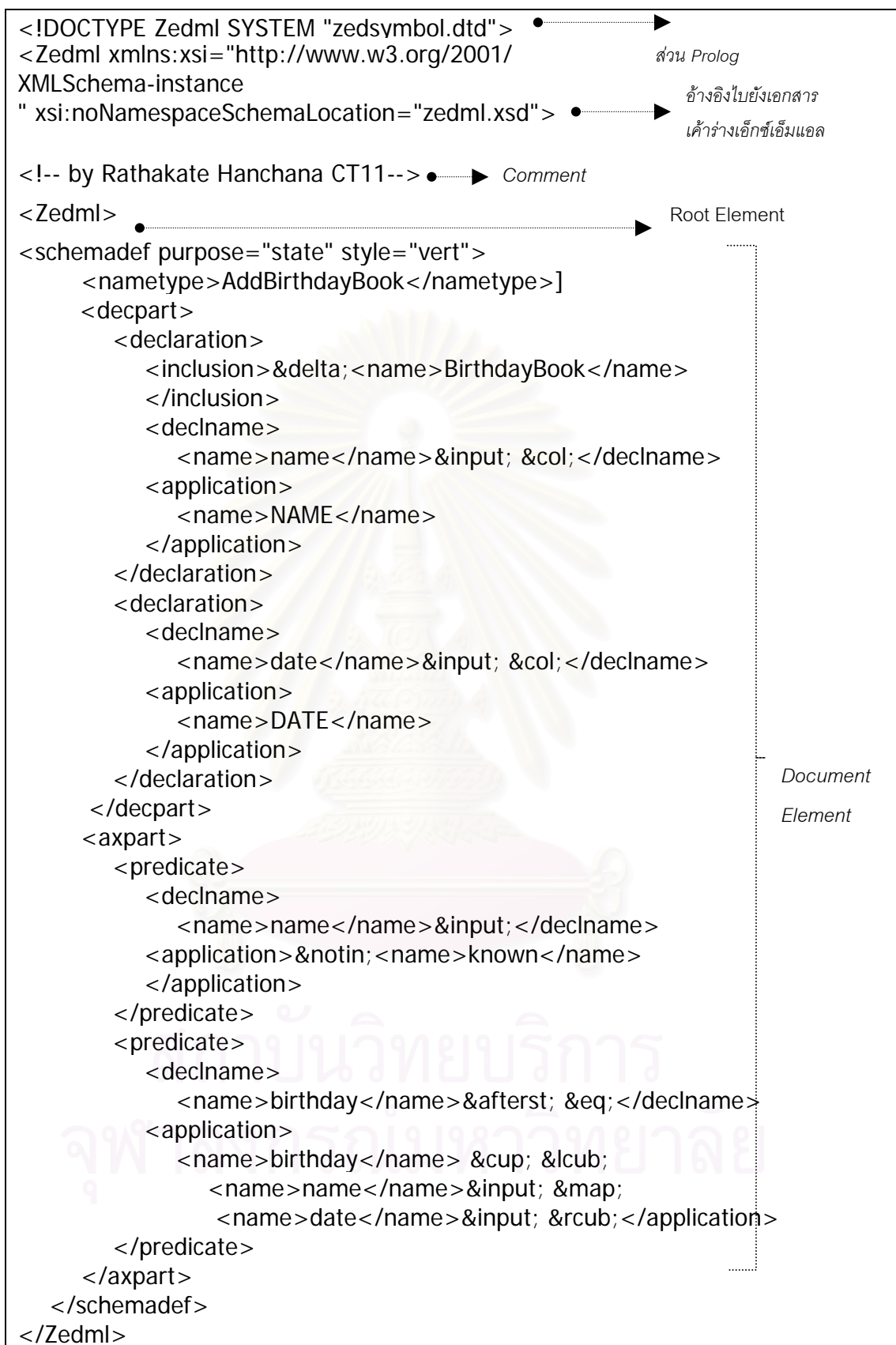
รูปที่ 3.14 ข้อกำหนดรูปร่างภาษาเซตเค้าร่าง AddBirthdayBook

ผู้ใช้สามารถเขียนเป็นเอกสารเอ็กซ์เอ็มแอลในรูปแบบที่ถูกต้องได้ ดังตัวอย่างที่แสดงในรูปที่ 3.15

```
<?xml version="1.0" encoding="UTF-8"?>
```

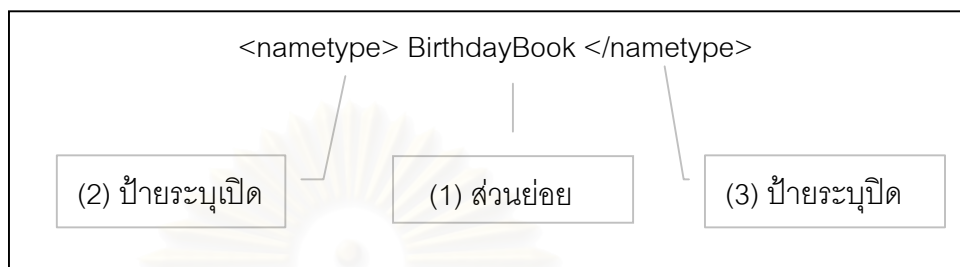
อ้างอิงไปยังเอกสาร ดีทีดี





รูปที่ 3.15 ตัวอย่างส่วนประกอบของเอกสารข้อกำหนดรูปนัยภาษาเซตที่ถูกต้อง

3.3.1.2 ภายในองค์ประกอบส่วนย่อย ของข้อกำหนดรูปร่างในเอกสารเอ็กซ์เอ็มแอลต้องประกอบด้วย การเขียน ป้ายระบุเปิด (Start Tag) และป้ายระบุปิด (End Tag) ดังตัวอย่างในรูปที่ 3.16 ส่วนย่อย BirthdayBook (1) จะประกอบด้วย ป้ายระบุเปิด nametype (2) และป้ายระบุปิด nametype (3)



รูปที่ 3.16 รูปแบบการเขียนป้ายระบุเปิดและป้ายระบุปิด

3.3.1.3 ข้อกำหนดรูปร่างในแต่ละเค้าร่างจะต้องมีรากของส่วนย่อย (Root Element) ที่ชื่อ “Zedml” เท่านั้น ดังตัวอย่างในรูปที่ 3.17



รูปที่ 3.17 รากของส่วนย่อย Zedml ในการเขียนข้อกำหนดรูปร่างภาษาเซต

3.3.2 ข้อกำหนดด้านเอกสารที่ถูกต้องสมบูรณ์ ตามข้อกำหนดในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล

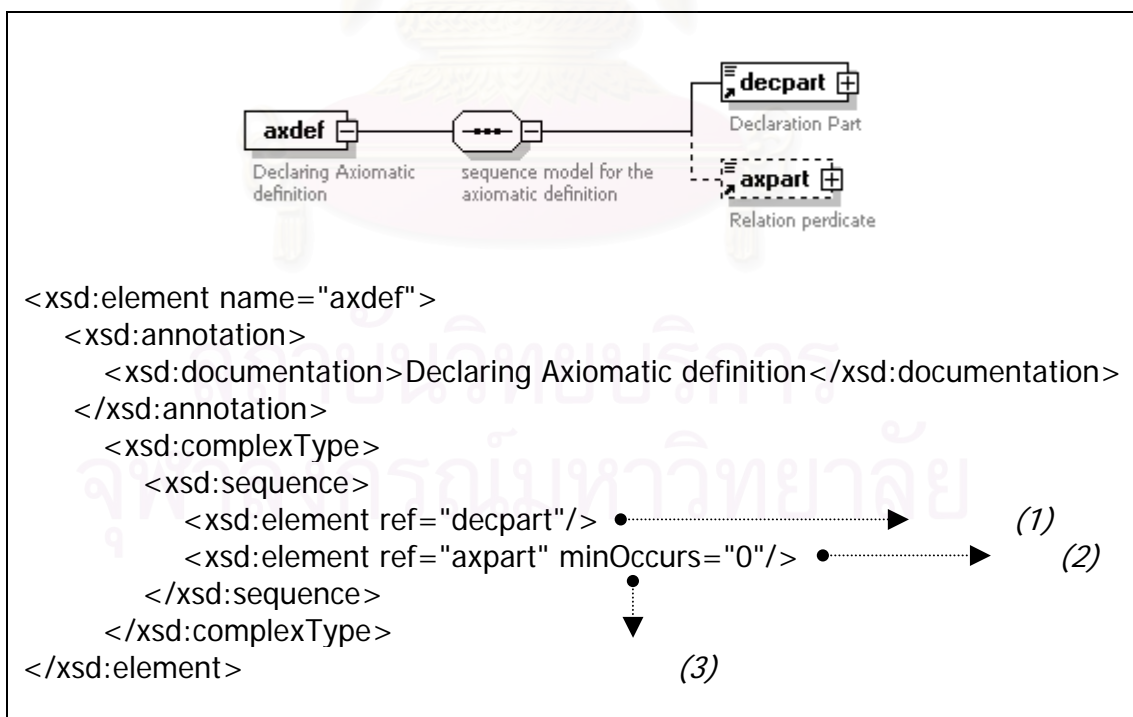
ถึงแม้ว่าข้อกำหนดรูปร่างที่เขียนด้วยเอ็กซ์เอ็มแอลนั้นจะมีรูปแบบที่ถูกต้องตามข้อกำหนดของภาษาเอ็กซ์เอ็มแอล แต่ถ้าหากผู้ใช้งานยังไม่ได้ตรวจสอบข้อกำหนด ด้านความถูกต้องสมบูรณ์ของโครงสร้างข้อกำหนดรูปร่างภาษาเซต ที่ผู้วิจัยระบุไว้ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลตามรายละเอียดในหัวข้อที่ 3.1 ก็จะทำให้ข้อกำหนดรูปร่างที่เขียนมานั้น จะไม่สามารถนำเสนอบนเว็บ

เบราวน์เซอร์ได้ จนกว่าจะได้รับการแก้ไขโครงสร้างของข้อกำหนดรูปนัยให้ตรงตามข้อกำหนดและเงื่อนไขในเอกสารเค้าร่างเซดเอ็กซ์เอ็มแอลยกตัวอย่างเช่น ผู้ใช้งานต้องการเขียนข้อกำหนดรูปนัยภาษาเซดที่มีโครงสร้างของเค้าร่างแบบสัจพจน์ (Axiom definition) ดังรูปที่ 3.18

$$\begin{aligned} &úwordCount :document \times^7 \mathbb{N} \\ &ú\text{ýýýýýýýýýý} \\ &ú \text{ } d:document @wordCount d= sizebag(wordFreq d) \end{aligned}$$

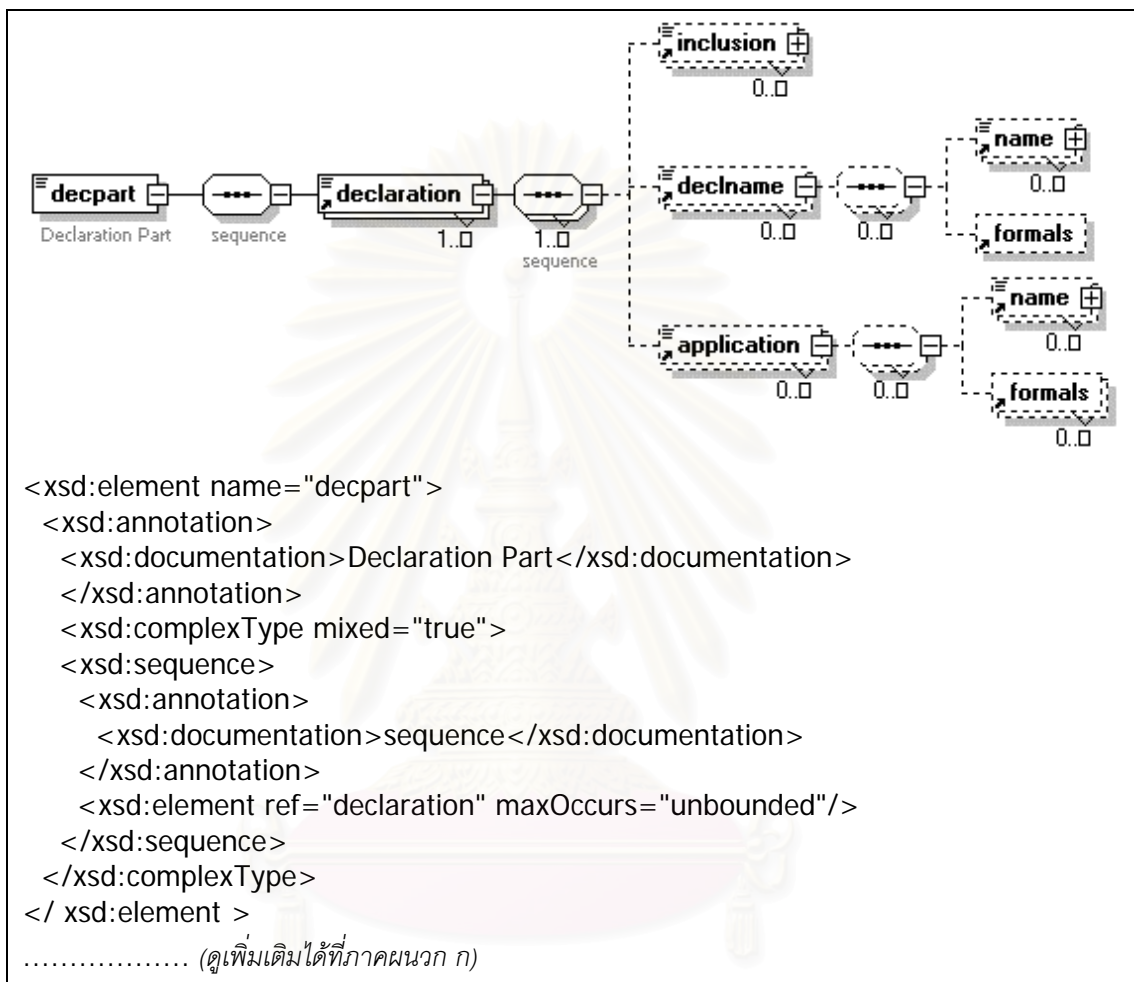
รูปที่ 3.18 ข้อกำหนดรูปนัยภาษาเซดที่มีเค้าร่างแบบสัจพจน์

ในเอกสารเค้าร่างเซดเอ็กซ์เอ็มแอล ผู้วิจัยได้นิยามลำดับส่วนย่อยของโครงสร้างเค้าร่างแบบสัจพจน์ ซึ่งจะประกอบด้วย ส่วนย่อยแบบอ้างอิง (Reference) ชื่อ decpart (Declaration part) (1) และส่วนย่อยแบบอ้างอิงชื่อ axpart (Axiomatic part) (2) เรียงกันมาตามลำดับ โดยที่ส่วนย่อยที่ชื่อ axpart จะมีลักษณะประจำของการปรากฏซ้ำเท่ากับศูนย์ (3) ซึ่งก็หมายความว่าส่วนย่อยที่ชื่อ axpart นั้นผู้ใช้สามารถเลือกได้ว่าจะเขียนหรือไม่เขียนลงไปในการข้อกำหนดรูปนัยแบบสัจพจน์ที่สร้างขึ้น ตัวอย่างการนิยามลำดับของส่วนย่อยแสดงได้ดังรูปที่ 3.19



รูปที่ 3.19 ลำดับโครงสร้างของ Axiom definition ในเอกสารเค้าร่างเซดเอ็กซ์เอ็มแอล

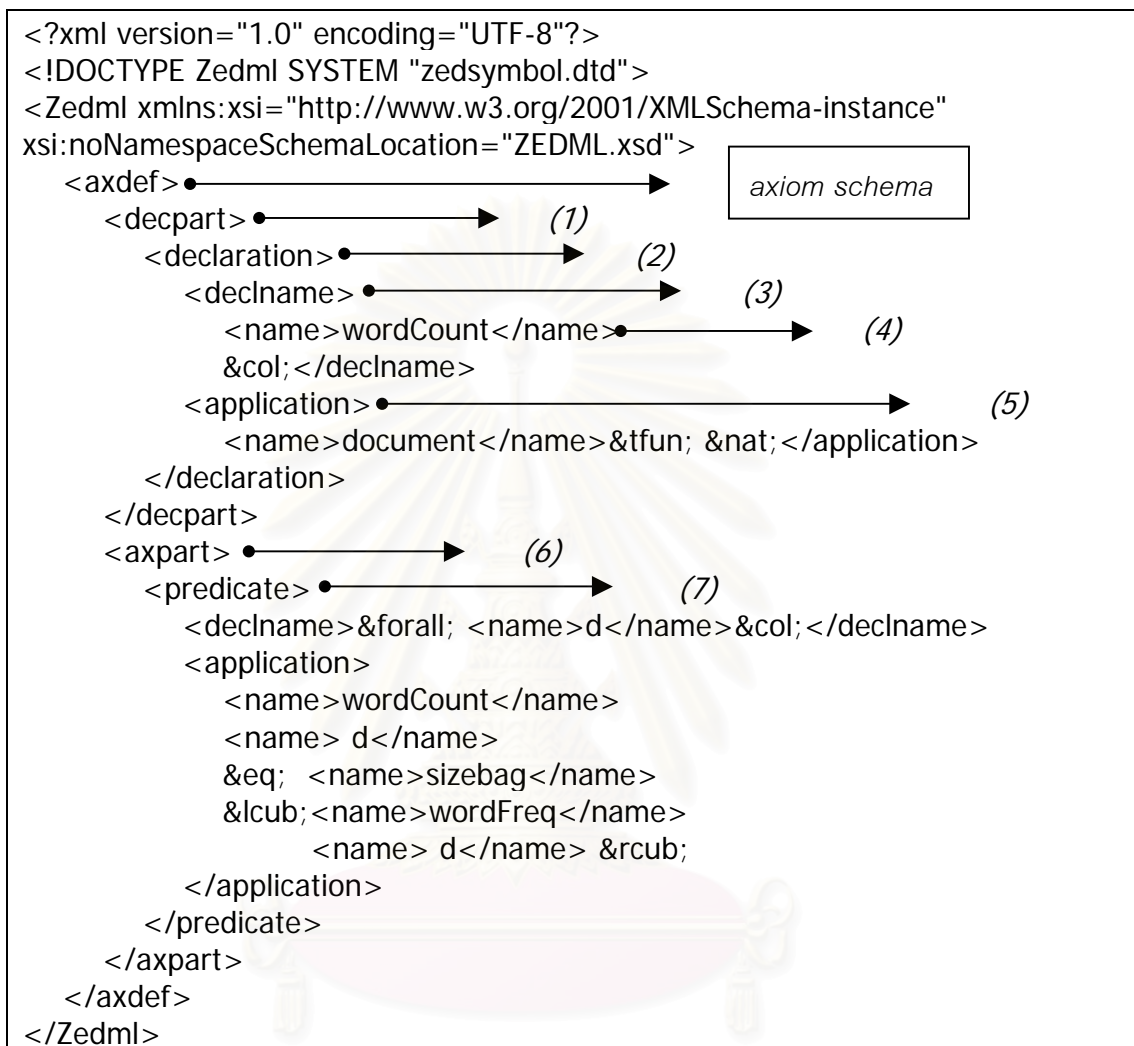
นอกจากนี้รายละเอียดของโครงสร้างส่วนย่อยอ้างอิง decpart และ xpart ก็ถูกนิยามเพิ่มไว้ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลเช่นเดียวกัน เช่น โครงสร้างของส่วนย่อยอ้างอิง decpart จะประกอบด้วยลำดับส่วนย่อยอื่นๆอีก เช่น declaration,inclusion,declname,name ดังตัวอย่างในรูปที่ 13.20



รูปที่ 3.20 ลำดับโครงสร้างของ decpart ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล

ผู้ใช้งานจะต้องมีการเขียนระบุแหล่งอ้างอิงถึงเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอลที่กล่าวมานี้ในส่วนของโปรล็อกในเอกสารเอ็กซ์เอ็มแอลเพื่อให้ตัวแจงส่วนในเว็บเบราว์เซอร์ สามารถทำหน้าที่ตรวจสอบโครงสร้างที่ถูกต้องให้ผู้ใช้งานทันที และต้องเขียนข้อกำหนดรูปนัยดังกล่าวตามโครงสร้างของส่วนย่อยที่ได้ระบุไว้ในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล ดังนั้นจากตัวอย่างข้างต้นนี้ผู้ใช้งานจะต้องเขียนข้อกำหนดรูปนัยตามตัวอย่างในรูปที่ 3.21 จึงจะทำให้เอกสาร เป็นไปตามข้อกำหนดด้านเอกสารที่มีรูปแบบถูกต้องของเอ็กซ์เอ็มแอลและมีไวยากรณ์ที่ถูกต้องสมบูรณ์ที่ผู้วิจัยได้กำหนดขึ้น คือมีลำดับของส่วนย่อย decpart (1) ที่ประกอบด้วยลำดับของส่วนย่อยอื่นๆเช่น

declaration(2), declname (3),name(4), application(5) และ ส่วนย่อย xpart (6) ที่ประกอบ ด้วยส่วนย่อย predicate (7)



รูปที่ 3.21 ข้อกำหนดรูปนัยแบบ Axiom schema ที่มีรูปแบบไวยากรณ์ที่ถูกต้องสมบูรณ์

### 3.3.2.1 ข้อกำหนดการเขียนส่วนย่อย ในส่วนการประกาศและส่วนสัจพจน์

เนื่องจากโครงสร้างของข้อกำหนดรูปนัยที่มีโครงสร้างของเค้าร่างแบบเหมือนกล่อง มีลำดับของส่วนย่อยของเค้าร่างที่เขียนเหมือนกันคือมีลำดับของส่วนการประกาศ และส่วนสัจพจน์ ดังนั้นรูปแบบการเขียนส่วนย่อยของข้อกำหนดรูปนัยแบบต่างๆ ที่นิยามขึ้นในวิทยานิพนธ์นี้ สามารถนำมาใช้ร่วมกันได้โดยมีรายละเอียดของการเขียนส่วนย่อยใน ลำดับของ ส่วนการประกาศ

และส่วนสัจพจน์ เพื่อให้ตัวแจงส่วนสามารถจำแนกประเภทของข้อมูลและพิสูจน์ลำดับของโทเคน (Token) ได้ดังต่อไปนี้

3.3.2.1.1 ข้อกำหนดการเขียนส่วนย่อย ในส่วนการประกาศ จะกำหนดให้เขียนการประกาศตัวแปรทีละชุด โดยส่วนย่อยที่เขียนในลำดับของป้ายระบุ <declaration><declaname>, <name> คือชื่อของตัวแปร

เช่น

```
<declaration>
<declname><name>known</name></declname>
</declaration>
```

และ ส่วนย่อย ที่เขียนในลำดับของป้ายระบุ

```
<application><declname><name> คือส่วนของโทเคนที่จำแนก
การนิยามชนิดของข้อมูลที่กำหนดให้ตัวแปร เช่น
<application><name>DATE</name></application>
```

สำหรับการอ้างอิงเอนทิตีหรือสัญลักษณ์ทางคณิตศาสตร์นั้นให้ทำการอ้างอิงนอกป้ายระบุดังกล่าวเช่น

```
<declname>&pset;<name>NAME</name></declname>
หรือ
<application>&fun;<name>DATE</name></application>
```

3.3.2.1.2 ข้อกำหนดการเขียนส่วนย่อย ในส่วนสัจพจน์ จะกำหนดให้เขียนเพรดิเคต ทีละชุดโดย สำหรับวิธีการเขียนส่วนย่อยในลำดับของป้ายระบุ

```
<predicate><declaname><name>ให้ มีลักษณะการเขียน การอ้างอิง
เอนทิตี และความหมาย เหมือนส่วนประกาศ ข้างต้นทุกประการ
```

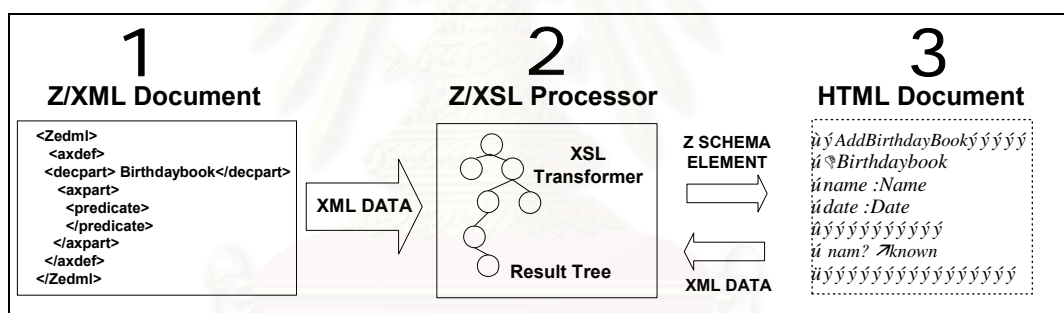
หลังจากมีการสร้างข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอลที่ถูกต้องสมบูรณ์เรียบร้อยแล้ว องค์ประกอบที่สำคัญอีกกระบวนการหนึ่งคือการนำเอาข้อกำหนดรูปนัยในรูปแบบเอกสารเอ็กซ์เอ็มแอลนำเสนอบนเว็บเบราว์เซอร์ ซึ่งในงานวิทยานิพนธ์นี้ได้ใช้โปรแกรมประยุกต์ของเอ็กซ์เอ็มแอลอีกประเภทหนึ่งเพื่อทำหน้าที่แสดงผลข้อกำหนดรูปนัยภาษาเซตบนเว็บเบราว์เซอร์นั้นคือ เอกสารเซตเอ็กซ์เอสแอลสไตล์ชีท (Z/XSL Style Sheet)

### 3.4 การแสดงผลข้อกำหนดรูปร่างภาษาเซตเอ็กซ์เอ็มแอลด้วยเอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีท (Z/XSL Style Sheet)

เอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีทที่ผู้วิจัยนำมาใช้นี้มีหน้าที่สำคัญอยู่ 2 ประการคือ

- 3.4.1 แปลง (Transform) เอกสารข้อกำหนดรูปร่างภาษาเซตที่เขียนด้วยเอ็กซ์เอ็มแอลให้อยู่ในรูปแบบของ เอกสารเอชทีเอ็มแอล
- 3.4.2 จัดรูปแบบ (Format) การแสดงผลเค้าร่างข้อกำหนดรูปร่างภาษาเซตในรูปแบบต่างๆ ให้ถูกต้อง

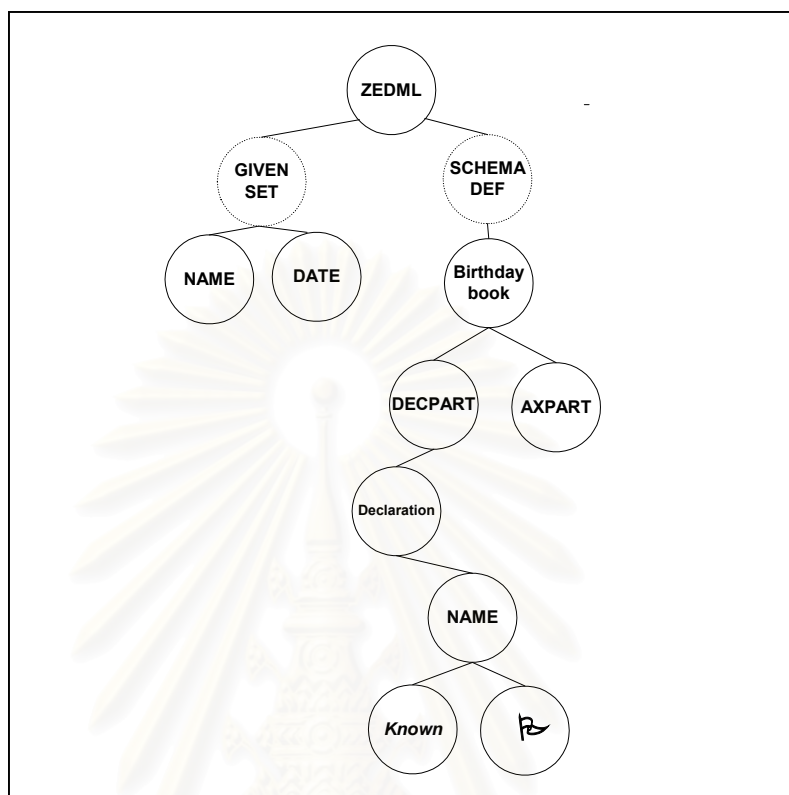
ในรายละเอียดของกระบวนการทำงาน ของเอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีท นี้จะทำงานตามข้อกำหนดของ W3C โดยมีขั้นตอนในการทำงาน ดังรูปที่ 3.22



รูปที่ 3.22 ขั้นตอนการทำงานของเอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีท

ในขั้นตอนที่ 1 ตัวประมวลผลเอ็กซ์เอ็มแอลที่กำหนดไว้ในเอกสารเซตเอ็กซ์เอ็มแอลสไตล์ชีท จะทำหน้าที่อ่านข้อมูลจากข้อกำหนดรูปร่างภาษาเซตที่อยู่ในรูปเอกสารเอ็กซ์เอ็มแอล โดยจะเข้าถึงข้อมูลที่ประกอบด้วยโครงสร้างลำดับของส่วนย่อยต่างๆ ข้อมูลลักษณะประจำของส่วนย่อย ตลอดจนข้อมูลการอ้างอิงเอนทิตี ต่างๆ ที่ผู้ใช้เขียนขึ้นในรูปแบบของข้อมูลเอ็กซ์เอ็มแอล

ขั้นตอนที่ 2 ตัวประมวลผลเอ็กซ์เอ็มแอลจะแปลงข้อมูลเอ็กซ์เอ็มแอลที่ได้จากขั้นตอนที่ 1 เป็น ข้อมูลในรูปแบบต้นไม้ (Tree) ที่ประกอบด้วยข้อมูลของโครงสร้างของข้อกำหนดรูปร่าง และส่วนย่อยต่างๆ สามารถอธิบายลักษณะของข้อมูลที่ได้รับการแปลงจากตัวประมวลผลเอ็กซ์เอ็มแอล ได้ตัวอย่างในรูปที่ 3.23



รูปที่ 3.23 ตัวอย่างลักษณะข้อมูลจากการแปลงเค้าร่าง BirthdayBook ในรูปแบบภูมิต้นไม้

หลังจากที่ได้ข้อมูลที่อยู่ในรูปของแผนภูมิต้นไม้เรียบร้อยแล้วตัวประมวลผลในเอกสารเซด เอ็กซ์เอสแอลสไตล์ซีทจะทำการเปรียบเทียบข้อมูลที่ได้เหล่านั้นกับแผ่นแบบ (Template) ที่ผู้วิจัยได้นิยามไว้ในเอกสารเซดเอ็กซ์เอสแอลสไตล์ซีทโดยจะทำการเปรียบเทียบข้อมูลจากบนลงล่าง (Top-Down) โดยเริ่มจากรากของส่วนย่อย Zedml ไปตามลำดับจนกระทั่งจบข้อมูล ซึ่งแผ่นแบบที่ผู้วิจัยนิยามขึ้นมานั้นประกอบด้วยแผ่นแบบย่อยแบบต่างๆ ที่ทำหน้าที่นิยามและจัดรูปแบบการแสดงผลข้อกำหนดรูปร่างภาษาเซดบนเว็บเบราว์เซอร์ ดังรายละเอียดต่อไปนี้

3.4.3 แผ่นแบบ ที่นิยามโครงสร้างของเค้าร่างข้อกำหนดรูปร่างประเภทต่างๆ

3.4.4 แผ่นแบบ ที่นิยามรูปแบบของชุดแบบอักษร(Font)ที่ต้องใช้เพื่อแสดงผลเอนทิตีที่เป็นสัญลักษณ์ทางคณิตศาสตร์



- 3.4.5 แผ่นแบบทั่วไปที่เพิ่มเติมขึ้นเพื่อความเหมาะสมในการแสดงผลบนเว็บเบราว์เซอร์ เช่น แผ่นแบบที่ควบคุมการจัดวางตำแหน่งซ้าย กลาง และขวาบนจอภาพ เป็นต้น

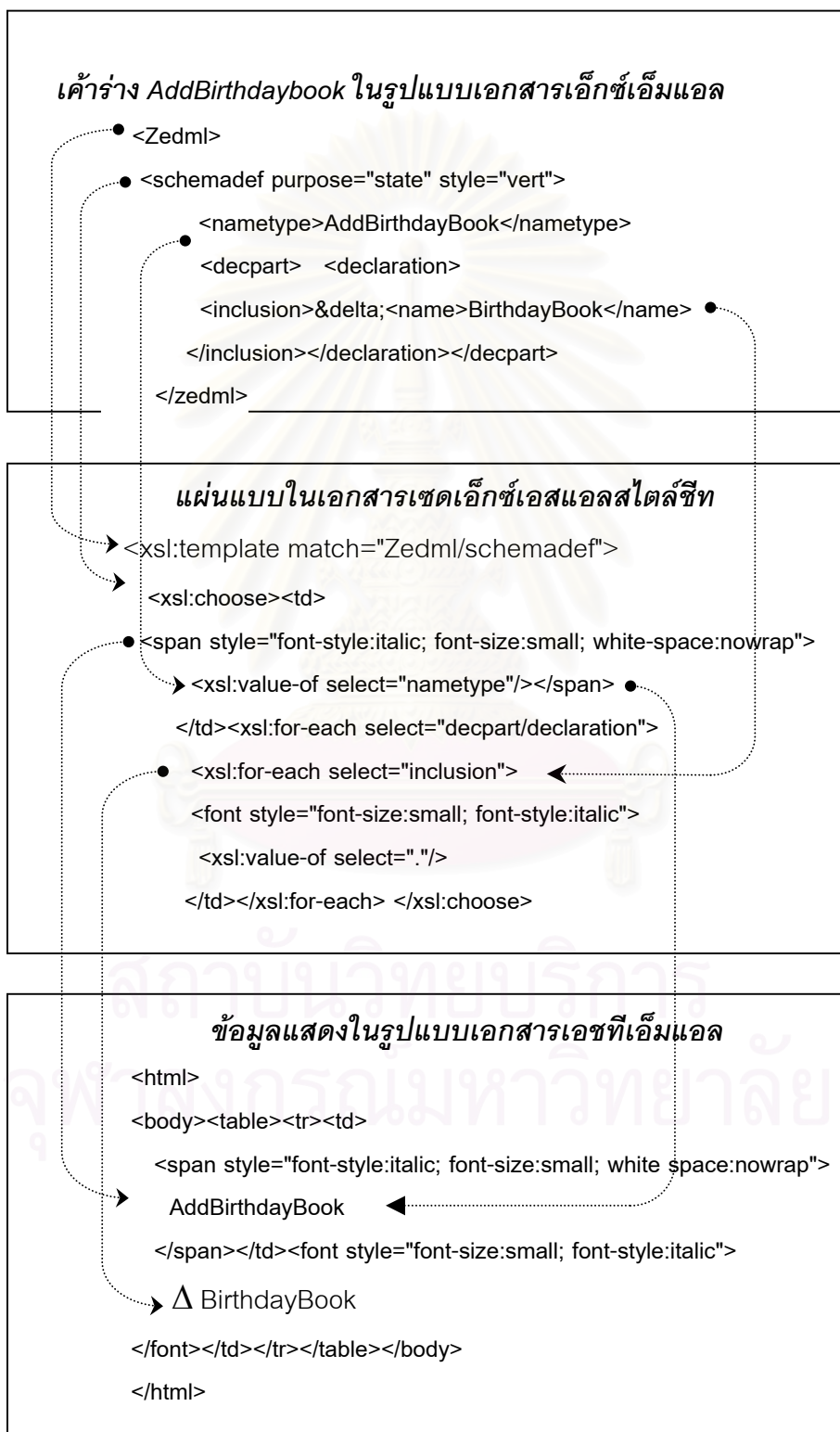
ตัวอย่างของการนิยามแผ่นแบบการแสดงผลของเค้าร่างกีเวนเซต ในเอกสารเซตเอ็กซ์เอสแอล สไคล์ชีทแสดงได้ดังรูปที่ 3.24 และสามารถดูรายละเอียดของเอกสารเพิ่มเติมได้ที่ภาคผนวก ค

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsl:output method="html"/>
  <!-- start givendef template -->
  <xsl:template match="Zedml/givendef">
    <div>
      <p>
        <xsl:attribute name="align"><xsl:value-of
select="@align"/></xsl:attribute>
        <table border="0" cellspacing="0" cellpadding="0">
          <tr>
            <td>[</td>
            <td>
              <span style="font-style:italic; font-size:small">
                <xsl:value-of select="."/>
              </span>
            </td>
          <td>]</td>
        </tr>
        </table>
      </p>
    </div>
  </xsl:template>
  <!--end givendef template-->
```

รูปที่ 3.24 แผ่นแบบการแสดงผลเค้าร่างกีเวนเซตในเอกสารเซตเอ็กซ์เอสแอลสไคล์ชีท

ในขั้นตอนที่ 3 เมื่อข้อมูลข้อกำหนดรูปถ่ายที่ทำการเปรียบเทียบเทียบกับแผ่นแบบตรงกัน เอกสารเซตเอ็กซ์เอสแอลสไคล์ชีท จะทำการแปลงข้อมูลเป็นเอกสารเอชทีเอ็มแอลโดยมีผลลัพธ์ตามที่ระบุไว้ในแผ่นแบบซึ่งจะทำให้สามารถนำเสนอข้อกำหนดรูปถ่ายบนเว็บเบราว์เซอร์ด้วยเอ็กซ์เอ็มแอลเป็นไปอย่างถูกต้อง โดยลักษณะการเปรียบเทียบข้อมูลกับแผ่นแบบจนถึงการแปลงเป็นเอกสารเอชทีเอ็มแอล สามารถอธิบายได้ดังตัวอย่างในรูปที่ 3.25 เป็นตัวอย่างการแปลงข้อมูลของ

เค้าร่าง AddBirthdaybook ที่อยู่ในรูปเอกสารเอ็กซ์เอ็มแอลเป็นเอกสารเอกซที่เอ็มแอล ด้วยเอกสาร เซดเอ็กซ์เอสแอลสไตล์ชีท



รูปที่ 3.25 การเปรียบเทียบเค้าร่าง AddBirthdaybook กับแผ่นแบบและเอกสารเลขที่เอ็มแอล

#### 3.4.6 การอ้างอิงเอกสารเซดเอ็กซ์เอสแอลสไตลชีท ในเอกสารเอ็กซ์เอ็มแอล

วิธีการนำเอกสารเอกสารเซดเอ็กซ์เอสแอลสไตลชีท ไปใช้เพื่อแสดงผลในระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซดด้วยเอ็กซ์เอ็มแอลนั้น ผู้ใช้ต้องทำการระบุแหล่งอ้างอิงเอกสารเซดเอ็กซ์เอสแอลสไตลชีท ที่ส่วนโปรล็อกในเอกสารเอ็กซ์เอ็มแอลที่เขียนข้อกำหนดรูปร่าง ดังตัวอย่างในรูปที่ 3.26

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
.....
</Zedml>
```

● → ระบุแหล่งอ้างอิงเอกสาร

#### รูปที่ 3.26 การระบุแหล่งอ้างอิงเอกสารเซดเอ็กซ์เอสแอลสไตลชีทที่ส่วนโปรล็อก

เมื่อผู้ใช้มีการเรียกดูข้อกำหนดรูปร่างจากเอกสารเอ็กซ์เอ็มแอลบนเว็บเบราว์เซอร์ผลลัพธ์ที่แสดงจะเป็นเอกสารเลขที่เอ็มแอลที่ได้รับการแปลงมาตามแผ่นแบบที่กำหนดไว้นั่นเอง

จากองค์ประกอบทั้ง 4 เอกสารที่กล่าวมาข้างต้นนั้น ล้วนแต่เป็นองค์ประกอบสำคัญในระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซดด้วยเอ็กซ์เอ็มแอล โดยจะขาดเอกสารใดเอกสารหนึ่งไม่ได้ดังนั้นผู้ใช้จะต้องทำความเข้าใจในหน้าที่และความสำคัญของเอกสารแต่ละประเภท จึงจะสามารถนำเสนอข้อกำหนดรูปร่างภาษาเซดด้วยเอ็กซ์เอ็มแอลได้อย่างมีประสิทธิภาพ

## บทที่ 4

### การทดสอบระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล

ในบทนี้จะกล่าวถึงการทดสอบระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล ซึ่งจะเป็นการทดสอบการนำเสนอข้อกำหนดรูปนัยทุกประเภทที่ได้ระบุไว้ใน รูปแบบการสืบเปลี่ยน ข้อมูลภาษาเซตภายใต้ระบบปฏิบัติการวินโดวส์ 2000

#### 4.1 สภาวะที่ใช้ทดสอบระบบ

เครื่องคอมพิวเตอร์และโปรแกรมประยุกต์ที่ใช้ทดสอบระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอลมีรายละเอียดดังต่อไปนี้

- 4.1.1 โน้ตบุ๊กคอมพิวเตอร์ เพนเทียม ทรี 1.2 กิกะเฮิร์ต
- 4.1.2 หน่วยความจำ 512 เมกะไบต์
- 4.1.3 ฮาร์ดดิส 30 กิกะไบต์
- 4.1.4 เว็บเบราว์เซอร์ Microsoft Internet Explorer รุ่น 6
- 4.1.5 โปรแกรมบรรณาธิการ (Edit Program) XML SPY รุ่น 4

#### 4.2 หัวข้อการทดสอบระบบ

การทดสอบระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล จะใช้ข้อกำหนดที่นิยามไว้ใน รูปแบบการสืบเปลี่ยนข้อมูลภาษาเซต เป็นหัวข้อหลักในการทดสอบระบบซึ่งมีรายละเอียดดังต่อไปนี้

- 4.2.1 ข้อกำหนดส่วนประกอบของรูปแบบโครงสร้างข้อกำหนดรูปร่างแบบต่างๆและการลำดับส่วนย่อย และการใช้ลักษณะประจำ ในการเขียนเค้าร่างภาษาเซต
- 4.2.2 การอ้างอิงเอนทิตีที่เป็นสัญลักษณ์ทางคณิตศาสตร์
- 4.2.3 การนิยามข้อมูลชนิดต่างๆและการประกาศตัวแปรที่ใช้ในการเขียนข้อกำหนดรูปร่าง
- 4.2.4 การตกแต่งอักษรพิเศษที่มีในการเขียนข้อกำหนดรูปร่าง

นอกจากนี้ ยังมีการวิเคราะห์เชิงวากยสัมพันธ์ (Syntactical analysis) ของการแจกส่วน (Parser) จากกรณีศึกษาเพื่อตรวจสอบความสมบูรณ์ของข้อกำหนดรูปนัยที่ได้

### 4.3 กรณีศึกษาที่ใช้ทดสอบระบบ

#### 4.3.1 กรณีศึกษา การนำเสนอข้อกำหนดรูปนัยโครงร่างเหมือนกล่องแนวตั้งและตามขวาง ด้วยระบบ BirthdayBook [4]

ระบบ BirthdayBook เป็นที่รู้จักกันอย่างแพร่หลายเนื่องจากเป็นระบบตัวอย่างที่ผู้ศึกษาวิธีการเขียนข้อกำหนดรูปนัยภาษาเซตใช้ศึกษาในชั้นเริ่มต้น เป็นระบบที่ใช้อธิบายการจับเก็บข้อมูลวันเกิดของบุคคลทั่วไปและสามารถทำความเข้าใจได้ง่าย มีลักษณะโครงสร้างของการเขียนข้อกำหนดรูปนัยเป็นรูปเหมือนกล่องแบบแนวตั้งและแบบตามขวาง มีการใช้การนิยามข้อมูลแบบ กี่ เวนเซต และการตกแต่งข้อมูลนำเข้าและนำออกในเค้าร่าง ลักษณะของการนิยามข้อมูลและเค้าร่างแสดงได้ดังรูปที่ 4.1 และรูปที่ 4.2

[NAME, DATE]

รูปที่ 4.1 การนิยามข้อมูลแบบกี่ยเวนเซตในระบบ BirthdayBook

```

ùý BirthdayBookýýýýýýýýýýýýýýýýýý
úknown :{NAME
úbirthday:Name ■ Date
ùýýýýýýýýýýýý
ú known = dom birthday
üýýýýýýýýýýýýýýýýýýýýýýýýýýýýýý

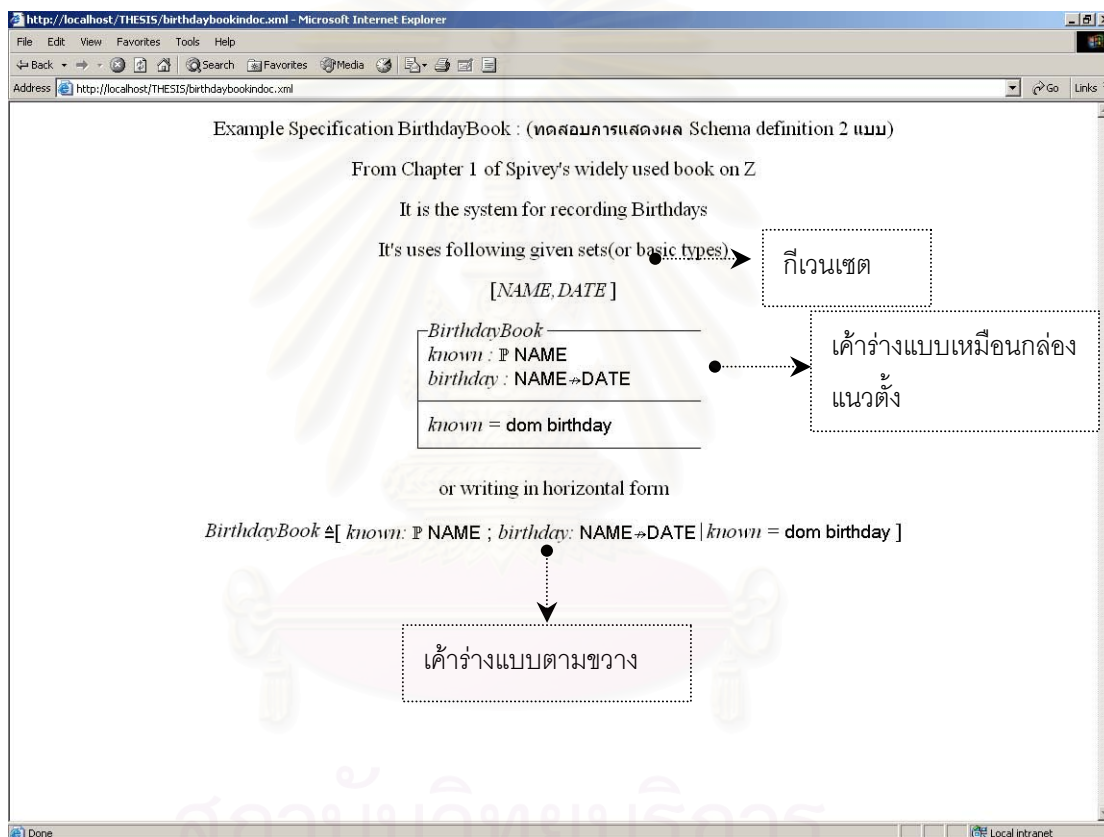
```

รูปที่ 4.2 เค้าร่างข้อกำหนดรูปนัย BirthdayBook

วัตถุประสงค์ของการทดสอบการนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอลในระบบ BirthdayBook นั้น จะทดสอบการนำเสนอรูปแบบโครงสร้างของเค้าร่างทั้งแบบเหมือนกล่องแนวตั้งและแบบตามขวาง และทดสอบการนำเสนอการนิยามข้อมูลแบบกี่ยเวนเซต โดยจะทำการทดสอบการนำเสนอข้อกำหนดรูปนัยภาษาเซตที่เขียนด้วยเอ็กซ์เอ็มแอล ด้วยรูปแบบโครงสร้างของข้อกำหนดรูปนัยดังกล่าวบนเว็บเบราว์เซอร์

#### 4.3.1.1 ผลการทดสอบการนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอลระบบ BirthdayBook

ในระบบ BirthdayBook มีเค้าร่างที่นำมาทดสอบการนำเสนอข้อกำหนดรูปนัยทั้งหมด 10 เค้าร่างมีลักษณะของโครงสร้างทั้งแบบเหมือนกล่องและแบบตามขวาง ซึ่งระบบสามารถนำเสนอข้อกำหนดรูปนัยบนเว็บเบราว์เซอร์ได้ทั้งหมด โดยมีตัวอย่างการแสดงผลบนเว็บเบราว์เซอร์ ดังรูปที่ 4.3

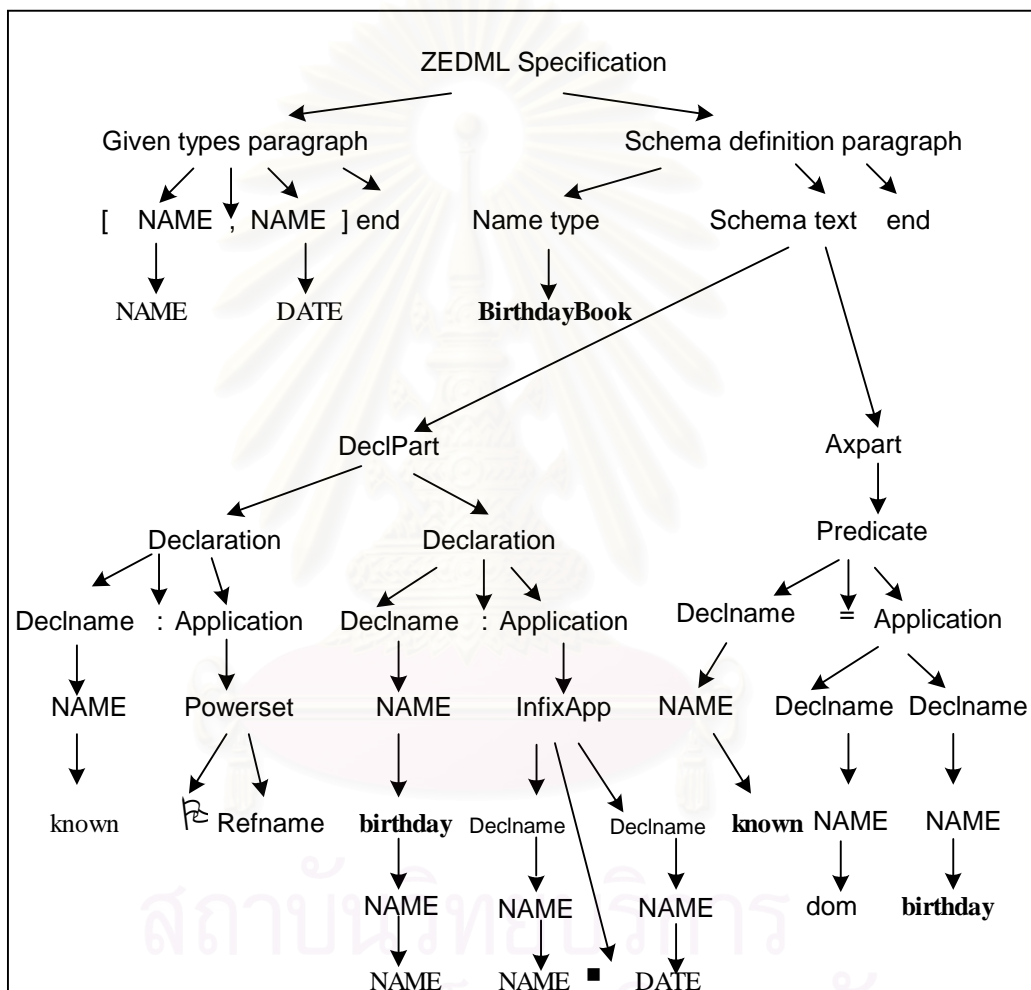


รูปที่ 4.3 การแสดงผลเค้าร่างข้อกำหนดรูปนัย BirthdayBook บนเว็บเบราว์เซอร์

#### 4.3.1.2 ผลการทดสอบการวิเคราะห์เชิงวากยสัมพันธ์

นอกจากนี้ยังมีการทดสอบการวิเคราะห์เชิงวากยสัมพันธ์ของการสร้างข้อกำหนดรูปนัยด้วยระบบ Birthday ด้วยเอ็กซ์เอ็มแอลโดยจะเป็นการตรวจสอบวากยสัมพันธ์ของเอกสารผลลัพธ์กับตัวอย่างการวิเคราะห์กระจายเค้าร่าง BirthdayBook ที่ระบุไว้ในคู่มือความหมายและการ

วิเคราะห์เชิงวากยสัมพันธ์ สัญกรณ์เซต [9] เพื่อตรวจสอบผลลัพธ์ของวากยสัมพันธ์เอ็กซ์เอ็มแอล ที่สร้างขึ้นว่าสามารถพิสูจน์ลำดับของโทเคน (Token) และการวิเคราะห์ศัพท์ (Lexical analysis) มีลักษณะตรงกับที่คู่มือแนะนำไว้หรือไม่ เพื่อประโยชน์ในการแปลงเค้าร่างที่อยู่ในรูปเอกสารเอ็กซ์เอ็มแอลไปยังเอกสารรูปแบบอื่นในรูปแบบที่ถูกต้องเช่น เอกสารเลเท็กซ์ (LaTeX) ซึ่งโครงสร้างการ แจงส่วนข้อกำหนดรูปนัยในเอกสารเอ็กซ์เอ็มแอลที่ได้ สามารถนำมาเขียนในรูปแบบแผนภูมิต้นไม้เพื่อ ให้ง่ายต่อการวิเคราะห์ดังตัวอย่างในรูปที่ 4.4



รูปที่ 4.4 การแจงส่วน เค้าร่าง BirthdayBook จากเอกสารเอ็กซ์เอ็มแอล

สำหรับรายละเอียดการทดสอบการนำเสนอข้อกำหนดรูปนัยระบบ BirthdayBook สามารถดูเพิ่มเติมได้ที่ภาคผนวก ง

4.3.2 กรณีศึกษา การนำเสนอข้อกำหนดรูปถ่ายโครงสร้างเหมือนกล่องแบบเจนเนอริค(Generic form) ด้วย ระบบการรายงานข้อผิดพลาด(Error Handling) [2] กรณี กองซ้อนแบบผลักขึ้น (Pushup stack)

เป็นระบบการรายงานข้อผิดพลาดกรณีการนำข้อมูลเข้ากองซ้อน เช่นตรวจสอบข้อมูลในกองซ้อนว่าว่าง (Empty) อยู่หรือไม่หาก หากพบว่าไม่มีข้อมูลอยู่ระบบจะแสดงรายงานว่า empty เช่นเดียวกันหากข้อมูลในกองซ้อนนั้นเต็ม (Full) ระบบจะแสดงรายงานว่า full ระบบการรายงานข้อผิดพลาดนี้ มีรูปแบบการเขียนข้อกำหนดรูปถ่ายโครงสร้างเหมือนกล่องเป็นแบบเจนเนอริค ดังตัวอย่าง ในรูปที่ 4.5

```

ú PushOK[X]
ú BoundStack[X]
ú x?:X
ú #stak < bound
ú stack' = X? stack

```

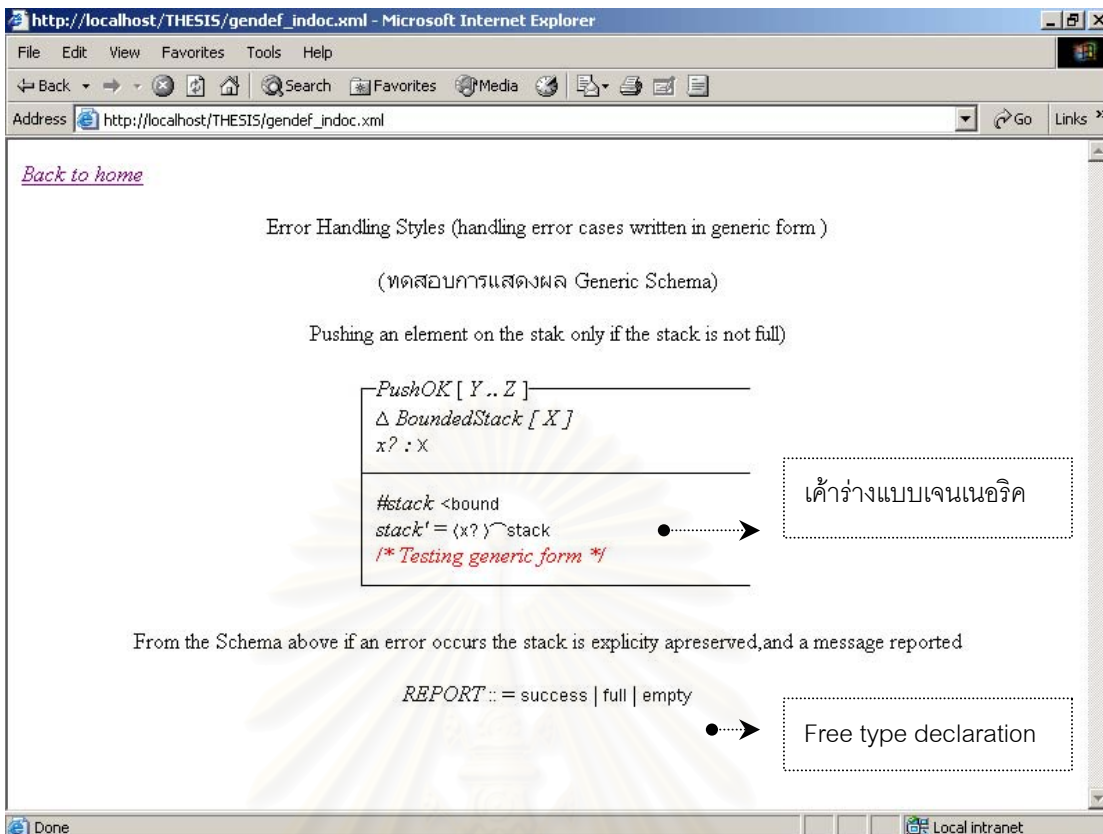
รูปที่ 4.5 ข้อกำหนดรูปถ่ายโครงสร้างเหมือนกล่องแบบเจนเนอริค

วัตถุประสงค์ของการทดสอบการนำเสนอข้อกำหนดรูปถ่ายภาษาเซตด้วยเอ็กซ์เอ็มแอลในระบบการรายงานข้อผิดพลาดนั้น จะทดสอบการนำเสนอรูปแบบโครงสร้างของเค้าร่างแบบเจนเนอริค และทดสอบการนำเสนอการนิยามข้อมูลแบบพีไรไทป์ โดยจะทำการทดสอบการนำเสนอข้อกำหนดรูปถ่ายภาษาเซตที่เขียนด้วยเอ็กซ์เอ็มแอล ด้วยรูปแบบโครงสร้างของข้อกำหนดรูปถ่ายดังกล่าวบนเว็บเบราว์เซอร์

4.3.2.1 ผลการทดสอบการนำเสนอข้อกำหนดรูปถ่ายภาษาเซตด้วยเอ็กซ์เอ็มแอลระบบรายงานข้อผิดพลาด

การทดสอบการนำเสนอระบบการรายงานข้อผิดพลาดกรณีการนำข้อมูลเข้ากองซ้อนประกอบด้วยเค้าร่างของข้อกำหนดรูปถ่ายที่ทดสอบทั้งหมด 10 เค้าร่าง มีการใช้การนิยามข้อมูลแบบพีไรไทป์ และการตกแต่งเค้าร่างด้วยตัวแปรนำเข้าและตัวแปรนำออก โดยสามารถนำเสนอเค้าร่างข้อกำหนดรูปถ่ายบนเว็บเบราว์เซอร์ได้ทั้งหมด ดังตัวอย่างในรูปที่ 4.6 สำหรับรายละเอียดของการแสดงผลเค้าร่างแบบเจนเนอริค สามารถดูเพิ่มเติมได้จากภาคผนวก จ





รูปที่ 4.6 การแสดงผลเค้าร่างแบบเจเนอริคบนเว็บเบราว์เซอร์

4.3.3 กรณีศึกษา การนำเสนอข้อกำหนดรูปนัยโครงสร้างเหมือนกล่องแบบสัจพจน์ (Axiomatic form) ด้วยระบบนับจำนวนคำ (Word counting) [2]

เป็นระบบที่อธิบายการนับจำนวนคำที่ปรากฏอยู่ในเอกสาร โดยในการเขียนข้อกำหนดรูปนัย จะมีการเขียนประกาศตัวแปรในระบบแบบตัวย่อ (Abbreviation) และใช้วิธีการเขียนข้อกำหนดรูปนัยโครงสร้างเหมือนกล่องแบบสัจพจน์ ดังตัวอย่างในรูปที่ 4.7

```

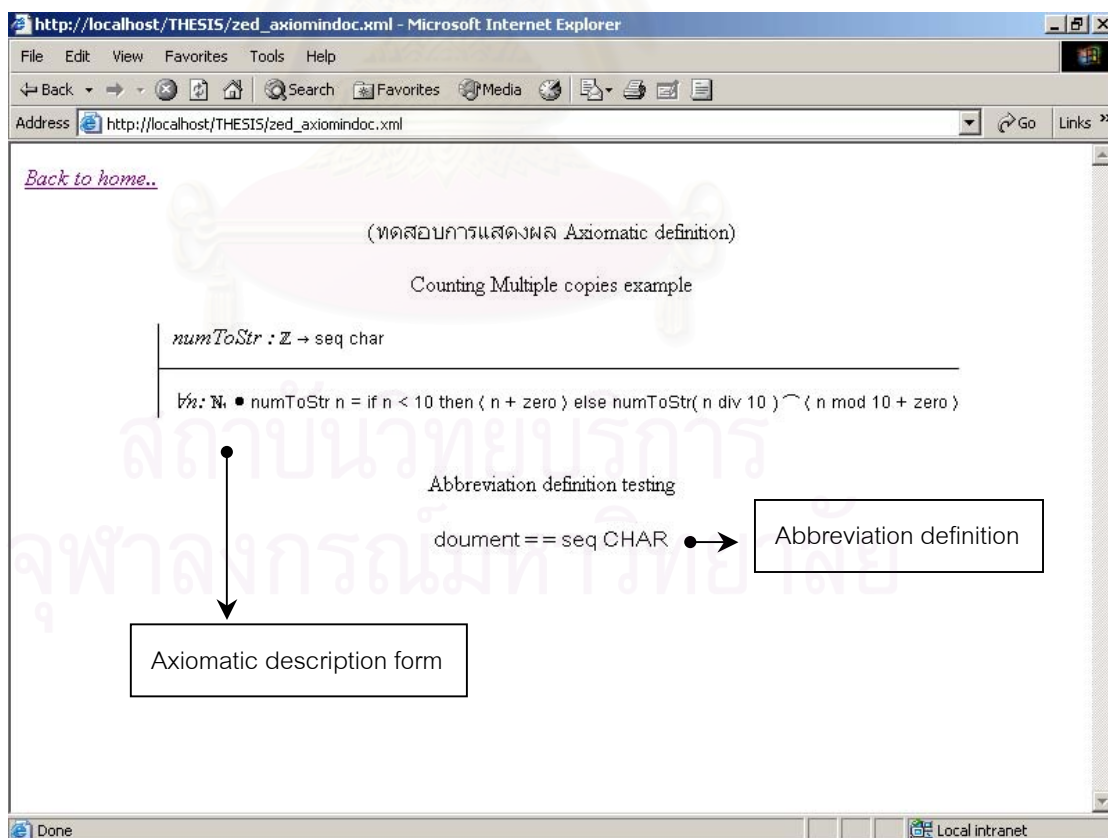
ú numToStr : C ↯ char
ú yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy yyy
ú n ; ๕ ๑ numToStr n = if n < 10 then ๓ +zero ๕ else numToStr (n div 10) ๓ mod 10 +zero ๑
    
```

รูปที่ 4.7 ข้อกำหนดรูปนัยโครงสร้างเหมือนกล่องแบบสัจพจน์

วัตถุประสงค์ของการทดสอบการนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอลในระบบนับจำนวนคำในเอกสารนั้น จะทดสอบการนำเสนอรูปแบบโครงสร้างของเค้าร่างแบบสัจพจน์ และทดสอบการนำเสนอการนิยามข้อมูลแบบตัวย่อ โดยจะทำการทดสอบการนำเสนอข้อกำหนดรูปนัยภาษาเซตที่เขียนด้วยเอ็กซ์เอ็มแอล ด้วยรูปแบบโครงสร้างของข้อกำหนดรูปนัยดังกล่าวบนเว็บเบราว์เซอร์

#### 4.3.3.1 ผลการทดสอบการนำเสนอข้อกำหนดรูปนัยภาษาเซตด้วยเอ็กซ์เอ็มแอล ระบบการนับจำนวนคำในเอกสาร บนเว็บเบราว์เซอร์แสดงได้ดังรูปที่ 4.8

เค้าร่างที่ใช้ทดสอบการนำเสนอระบบการนับจำนวนคำในเอกสาร ประกอบด้วยเค้าร่างของข้อกำหนดรูปนัยทั้งหมด 10 เค้าร่าง มีการใช้การนิยามข้อมูลแบบตัวย่อ และการตกแต่งเค้าร่างด้วยตัวแปรนำเข้าและตัวแปรนำออก โดยสามารถนำเสนอเค้าร่างข้อกำหนดรูปนัยบนเว็บเบราว์เซอร์ได้ทั้งหมด ดังตัวอย่างในรูปที่ 4.8 สำหรับรายละเอียดของการแสดงผลเค้าร่างแบบสัจพจน์สามารถดูเพิ่มเติมได้จากภาคผนวก ข



รูปที่ 4.8 การแสดงผลเค้าร่างแบบสัจพจน์บนเว็บเบราว์เซอร์



สำหรับรายละเอียดของการแสดงผลเค้าร่างในระบบร้านเช่าวิดีโอ สามารถดูเพิ่มเติมได้จาก ภาคผนวก ข

#### 4.4 สรุปผลการนำเสนอข้อกำหนดรูปร่างหน้าตาเว็บไซต์ด้วยเอ็กซ์เอ็มแอลจากกรณีทดสอบ

จากการทดสอบ ด้วยกรณีศึกษาทั้งหมดที่กล่าวมา สามารถสรุปผลการนำเสนอข้อกำหนดรูปร่างหน้าตาเว็บไซต์ด้วยเอ็กซ์เอ็มแอลบนเว็บเบราว์เซอร์แยกตามข้อกำหนดของการเขียนเค้าร่างแบบต่างๆที่ต้องตรวจสอบ ได้ดังตารางที่ 4.1

ตารางที่ 4.1 สรุปผลการนำเสนอข้อกำหนดรูปร่างหน้าตาเว็บไซต์จากกรณีทดสอบด้วยเอ็กซ์เอ็มแอลบนเว็บเบราว์เซอร์

ข้อกำหนดที่ตรวจสอบ การนำเสนอ	กรณีทดสอบ	การแสดงผลข้อกำหนดรูปร่างหน้าตาเว็บไซต์บนเว็บเบราว์เซอร์	
		ถูกต้อง	ไม่ถูกต้อง
เค้าร่างแบบแนวตั้ง	ระบบ BirthdayBook	✓	
เค้าร่างแบบแนวนอน	ระบบ BirthdayBook	✓	
เค้าร่างแบบสี่เหลี่ยม	ระบบการนับจำนวนค่า	✓	
เค้าร่างแบบเส้นแนวนอน	ระบบการรายงานข้อผิดพลาด	✓	
การประกาศตัวแปรในระบบ	ระบบร้านเช่าวิดีโอ	✓	
การตกแต่งอักษรพิเศษ	ทุกกรณีทดสอบ	✓	
การนิยามข้อมูลแบบกวีเมนต์	ระบบ BirthdayBook	✓	
การนิยามข้อมูลแบบฟรีไทป์	ระบบการรายงานข้อผิดพลาด	✓	
การนิยามข้อมูลแบบตัวย่อ	ระบบการนับจำนวนค่า	✓	
สัญลักษณ์ทางคณิตศาสตร์	ทุกกรณีทดสอบ	✓	
ตัวอักษรคำอธิบาย	ทุกกรณีทดสอบ	✓	

จากข้อมูลในตารางการสรุปผลการทดสอบกับกรณีศึกษา พบว่าในข้อกำหนดการนำเสนอที่ได้ระบุไว้ในหัวข้อการทดสอบกับกรณีศึกษานั้น ครอบคลุมข้อกำหนดการเขียนข้อกำหนดรูปร่างหน้าตาในรูปแบบการสืบเปลี่ยนข้อมูลภาษาเว็บ และสามารถแสดงผลบนเว็บเบราว์เซอร์ได้อย่างถูกต้องด้วยการใช้เอกสารทั้งหมดที่ได้ระบุไว้ในวัตถุประสงค์ของงานวิทยานิพนธ์นี้ ซึ่งประกอบด้วย

เอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล (Z/XML Schema) เอกสารเซตเอ็กซ์เอ็มแอลดีทีดี (Z/XML DTD) และเอกสารเซตเอ็กซ์เอสแอลสไตล์ชีท (Z/XSL Style Sheet) ซึ่งเอกสารทั้งหมดเป็นองค์ประกอบที่สำคัญของระบบการนำเสนอข้อมูลกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล

อนึ่ง การที่ระบบการนำเสนอข้อมูลกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอลจะสามารถทำงานได้อย่างถูกต้องและมีประสิทธิภาพนั้นต้องอาศัยองค์ประกอบทั้งหมดที่กล่าวมา ดังนั้นหากในขั้นตอนการเขียนข้อมูลกำหนดรูปร่างภาษาเซตด้วยเอ็กซ์เอ็มแอล ขาดองค์ประกอบใดๆที่เป็นเอกสารดังกล่าว ระบบก็จะไม่สามารถทำงานได้อย่างถูกต้องสมบูรณ์เช่นกัน



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## บทที่ 5

### สรุปผลการวิจัย และข้อเสนอแนะ

งานวิทยานิพนธ์นี้สามารถสรุปผลที่ได้รับจากการวิจัย ประโยชน์ที่ได้รับและข้อเสนอแนะได้ดังต่อไปนี้

#### 5.1 สรุปผลการวิจัย

จากการวิจัย ได้ออกแบบและพัฒนาระบบการนำเสนอข้อกำหนดครุภัณฑ์ด้วยอิเล็กทรอนิกส์ โดยระบบที่ผู้วิจัยได้ออกแบบนั้น ประกอบด้วยองค์ประกอบที่เป็นโปรแกรมประยุกต์ของอิเล็กทรอนิกส์ แอล ได้แก่

- 5.1.1 เอกสารเค้าร่างเซตอิเล็กทรอนิกส์แอล ซึ่งทำหน้าที่ตรวจสอบความถูกต้องสมบูรณ์ของโครงสร้างข้อกำหนดครุภัณฑ์ที่ผู้ใช้สร้างขึ้นด้วยอิเล็กทรอนิกส์ แอล และตรวจสอบรูปแบบที่ถูกต้องตามมาตรฐานของไวยากรณ์และวากยสัมพันธ์อิเล็กทรอนิกส์แอล
- 5.1.2 เอกสารเซตอิเล็กทรอนิกส์แอลดีทีดี ซึ่งทำหน้าที่นิยามชื่อเอนทิตีหรือสัญลักษณ์ทางคณิตศาสตร์เพื่อที่จะนำไปอ้างอิงกับชุดอักขระยูนิโคดในการแสดงผลสัญลักษณ์ทางคณิตศาสตร์เหล่านั้นบนเว็บเบราว์เซอร์
- 5.1.3 เอกสารเซตอิเล็กทรอนิกส์เอสแอลสไตลชีท ทำหน้าที่จัดรูปแบบของโครงสร้างของเค้าร่างข้อกำหนดครุภัณฑ์ภาษาเซตที่สร้างขึ้นให้ถูกต้อง และแปลงข้อมูลข้อกำหนดครุภัณฑ์จากเอกสารอิเล็กทรอนิกส์แอลให้เป็น เอกสารเอชทีเอ็มแอล เพื่อที่จะสามารถแสดงผลบนเว็บเบราว์เซอร์ได้ทันที

นอกจากนี้ยังได้ออกแบบขั้นตอนวิธีการสร้างข้อกำหนดครุภัณฑ์ภาษาเซตด้วยไวยากรณ์ของอิเล็กทรอนิกส์แอล การกำหนดส่วนโปรล็อกของเอกสารอิเล็กทรอนิกส์แอลที่จะต้องมีการระบุแหล่งอ้างอิงที่เป็นเอกสาร องค์ประกอบหลักทั้ง 3 เอกสารที่กล่าวมา และในส่วนของข้อมูลส่วนย่อยในเค้าร่างทั้งหมดต้องสร้างจากโครงร่างที่ได้นิยามไว้ในเอกสารเค้าร่างเซตอิเล็กทรอนิกส์แอล ส่วนการใช้สัญลักษณ์ทางคณิตศาสตร์ให้ใช้ วิธีการอ้างอิงเอนทิตีจากชื่อของเอนทิตีที่ได้นิยามไว้ในเอกสารเซตอิเล็กทรอนิกส์แอลดีทีดี

ในการทดสอบกับกรณีศึกษาทั้ง 3 กรณีได้แก่ ระบบ BirthdayBook ระบบนับจำนวนคำในเอกสาร ระบบการรายงานข้อผิดพลาด และกรณีศึกษาเพิ่มเติมเพื่อทดสอบในส่วนของการ

ประกาศตัวแปรในระบบบ้านเช่าวีดีโอ นั้น สามารถนำเสนอข้อกำหนดรูปถ่ายจากกรณีศึกษาดังกล่าวด้วยเอ็กซ์เอ็มแอลบนเว็บเบราว์เซอร์ โดยไม่ปรากฏข้อผิดพลาดใดๆ

นอกจากนี้ในการออกแบบและพัฒนาระบบผู้วิจัยได้ทำการนิยามข้อกำหนดเพิ่มเติมในเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล ทั้งนี้เพื่อความเหมาะสมกับการแสดงผลบนเว็บเบราว์เซอร์ได้แก่ การจัดวางตำแหน่งการนำเสนอข้อกำหนดรูปถ่าย การเพิ่มข้อความอธิบายข้อกำหนดรูปถ่ายเป็นต้น

## 5.2 ประโยชน์ของระบบการนำเสนอข้อกำหนดรูปถ่ายภาษาเซตด้วยเอ็กซ์เอ็มแอล

- 5.2.1 ได้เอกสารที่เป็นองค์ประกอบสำคัญในการนำเสนอข้อกำหนดรูปถ่ายภาษาเซตด้วยเอ็กซ์เอ็มแอล ทำให้ผู้ใช้สามารถสร้างข้อกำหนดรูปถ่ายภาษาเซตด้วยเอ็กซ์เอ็มแอลได้ทันที โดยที่ไม่ต้องคอยตรวจสอบโครงสร้างของข้อกำหนดรูปถ่ายแบบต่างๆว่าถูกต้องหรือไม่ และไม่ต้องยุ่งยากกับการสร้างวิธีการแสดงผลข้อกำหนดรูปถ่ายจากเอกสารเอ็กซ์เอ็มแอล ทำให้การสร้างข้อกำหนดรูปถ่ายด้วยเอ็กซ์เอ็มแอลทำได้ง่ายและใช้เวลาในการศึกษาน้อย เพราะผู้ใช้สามารถเข้าใจโครงสร้างของข้อกำหนดในรูปแบบของแผนภูมิต้นไม้จากเอกสารเอ็กซ์เอ็มแอลด้วยตนเองได้
- 5.2.2 ข้อกำหนดรูปถ่ายในรูปแบบของไวยากรณ์เอ็กซ์เอ็มแอลที่สร้างขึ้นจากระบบ จะมีความสอดคล้องกับ รูปแบบการสืบเปลี่ยนข้อมูลภาษาเซต เพราะทุกองค์ประกอบของเอกสารเค้าร่างเซตเอ็กซ์เอ็มแอล ได้รับออกแบบมาจากข้อกำหนดเดียวกันทำให้ข้อกำหนดรูปถ่ายที่ได้นั้นยังคงสามารถ ใช้ในรูปแบบการสืบเปลี่ยนข้อมูลภาษาเซตได้เหมือนเดิมและยังสามารถนำเสนอข้อกำหนดรูปถ่ายดังกล่าวบนเว็บเบราว์เซอร์ได้อีกด้วย

## 5.3 ข้อเสนอแนะ

เนื่องจากข้อกำหนดรูปถ่ายที่ได้มีรูปแบบเป็นเอกสารเอ็กซ์เอ็มแอลทำให้นักวิทยานิพนธ์นี้สามารถเพิ่มเติมในส่วนของเอกสารองค์ประกอบต่างๆให้มีความสามารถอื่นๆที่มีประโยชน์เพิ่มเติมได้ ยกตัวอย่างเช่น การแปลงข้อมูลข้อกำหนดรูปถ่ายที่ได้เป็น โปรแกรมประยุกต์ของเอ็กซ์เอ็มแอลประเภทอื่นๆ เช่นแปลงข้อกำหนดรูปถ่ายภาษาเซตเป็นแผนภาพภาษายูเอ็มแอลหรือแปลงข้อกำหนดรูปถ่ายภาษาเซตเป็นข้อมูลในรูปแบบเลเท็กซ์เพื่อใช้ทดสอบไวยากรณ์กับโปรแกรมประยุกต์ Z/EVE ต่อไป ตลอดจนการปรับปรุงเอกสารเอ็กซ์เอ็มแอลสไตร์ซีทีให้สามารถพิสูจน์ความสอดคล้องของชนิดข้อมูล (Type Checking) ของข้อกำหนดรูปถ่ายผ่านระบบอินเทอร์เน็ตได้

## รายการอ้างอิง

- [1] Bowen, J. Formal Specification and Documentation Using Z: A Case Study Approach. (n.p.) : International Thomson Publishing Company, 1996.
- [2] Barden, R., Stepney, S. and Cooper, D. Z IN PRACTICE. : Prentice Hall International (UK) Limited, 1994.
- [3] Brien S. M. and Nicholls J. E. "Z Base Standard". Technical Monograph PRG-107 Oxford University under ISO/IEC JTC1/SC22 (Nov, 1992).
- [4] Spivey, J.M. "An introduction to Z and formal specifications" Software Engineering Journal, (January 1989)
- [5] Ciancarini, P. Mascolo, C. and Vitali, F., "Visualizing Z notation in HTML documents" ZUM'98: The z formal specification Noatation 11<sup>th</sup> International Conference of Z User, Berlin, Germany, (September 1998), pp 81-95.
- [5] German, D. and Cowan, D,"Experiments with the Z Interchange Format and SGML" Ninth Int' Conference. Z Formal Method (ZUM). (1995):224-233.
- [6] Sun, J. Dong, S. Liu, J. and Wang, H. "Z Family on the Web with Their UML Photos". Technical report, School of Computing, National University of Singapore. (TRA1-01, January 2001).
- [7] Bowen, J. and Chippington, D. "Z on the Web Using Java" 11<sup>th</sup> Int' Conference Z Formal Method (ZUM).(1998):pp 66-80.
- [8] Toyn, I. "Formal Specification Z notation Syntax, Type and Semantics". BSI Panel IST/5/-/19/2(Z Notation). (August 24, 2000).
- [9] Mertz, D. Comparing W3C XML Schema and Document Type Definition (DTDs) [Online] Available from: [http://www-106.ibm.com/ developerworks/ library/-x-matter7.html](http://www-106.ibm.com/developerworks/library/-x-matter7.html) [2001, May 29]
- [10] Bray, T. Paoli, J. and Sperberg, M. World Wide Web Consortium, March 1998. Extensible Markup Language Recommendation [Online] Available from: <http://www.3c.org/TR/1998/REC-xml-19980210> [2001, July 9]



- [11] World Wide Web Consortium (W3C). XML Schema [Online] Available from:  
<http://www.3c.org/TR/xmlschema-0> [2001, Nov 3]
- [13] World Wide Web Consortium (W3C). Extensible style sheet language (xsl)  
[Online] Available from: <http://www.3c.org/Style/XSL,2000> [2001, Nov 3]
- [14] Wood A. Unicode Resource [Online] Available from  
<http://www.hclrss.demon.uk/unicode/fonts.html> [2002, Jan 12]



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย



ภาคผนวก

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

### ภาคผนวก ก. เอกสารเค้าร่างเซดเอ็กซ์เอ็มแอล

เอกสารเค้าร่างเซดเอ็กซ์เอ็มแอล เป็นโปรแกรมประยุกต์ประเภทหนึ่งของเอ็กซ์เอ็มแอลจึงมีวากยสัมพันธ์ตามข้อกำหนดของภาษาเอ็กซ์เอ็มแอล ซึ่งในงานวิทยานิพนธ์นี้จะใช้เอกสารเค้าร่างเซดเอ็กซ์เอ็มแอลเป็นโปรแกรมที่คอยตรวจสอบโครงสร้างของข้อกำหนดรูปถ่ายที่ผู้ใช้สร้างขึ้น โดย และเอกสารนี้จะสามารถทำงานได้อย่างสมบูรณ์ก็ต่อเมื่อผู้ใช้งานมีการเขียนระบุแหล่งอ้างอิงเอกสารในส่วนโปรล็อกเรียบร้อยแล้ว

เอกสารเค้าร่างเซดเอ็กซ์เอ็มแอลที่ผู้วิจัยนำมาใช้นี้มีชื่อเอกสารว่า zedml.xsd ในเอกสารจะมีการนิยามโครงสร้างของข้อกำหนดรูปถ่ายภาษาเซดในรูปของแผนภูมิต้นไม้ที่ประกอบด้วยลำดับของส่วนย่อย ข้อมูลลักษณะประจำที่มี เป็นต้น และมีลักษณะการนิยามส่วนย่อยแบบอ้างอิงซึ่งจะทำให้สะดวกในการนำมาใช้งานใหม่(Reuse) ยกตัวอย่างเช่น ลำดับของส่วนย่อย การประกาศ(Decpart) และส่วนย่อยสัจพจน์ (Axpert) จะมีการใช้ในหลายๆโครงสร้างของข้อกำหนดรูปถ่าย การนิยามลำดับของส่วนย่อยทั้งสองนี้แบบอ้างอิงจะทำให้ลดความซ้ำซ้อนของการสร้างเอกสารและง่ายต่อการตรวจสอบข้อผิดพลาด สำหรับรายชื่อของส่วนย่อยที่นิยามไว้ในเอกสารเค้าร่างเซดเอ็กซ์เอ็มแอล เรียงตามตัวอักษรภาษาอังกฤษแสดงได้ดังตารางที่ ก-1

ตารางที่ ก-1 รายชื่อส่วนย่อยที่นิยามไว้ในเอกสารเค้าร่างเซดเอ็กซ์เอ็มแอล

ลำดับ	ส่วนย่อย
1	<u>abbrevdef</u>
2	<u>application</u>
3	<u>axdef</u>
4	<u>axpart</u>
5	<u>body</u>
6	<u>branch</u>
7	<u>declaration</u>
8	<u>declarations</u>
9	<u>declname</u>
10	<u>decpart</u>
11	<u>formals</u>
12	<u>freetypedef</u>
13	<u>gendef</u>
14	<u>genparam</u>
15	<u>givendef</u>
16	<u>inclusion</u>
17	<u>name</u>
18	<u>nametype</u>
19	<u>note</u>
20	<u>opdec</u>
21	<u>predicate</u>
22	<u>schemadef</u>
23	<u>sexp</u>
24	<u>sub</u>
25	<u>sup</u>
26	<u>typename</u>
27	<u>Zedml</u>

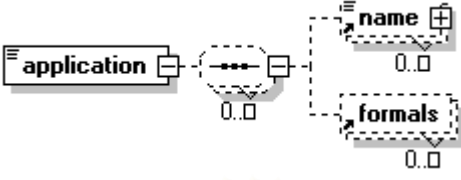
รายละเอียดของส่วนย่อยในเอกสารเค้าร่างเซดเอ็มแอล มีดังต่อไปนี้

1. ส่วนย่อย Abbrevdef หรือข้อกำหนด ตัวย่อ (Abbreviation definition) มีรายละเอียดของโครงสร้างและรหัสต้นฉบับ (Source Code) ดังรูปที่ ก-1

diagram	
children	<b>name body</b>
used by	element <b>Zedml</b>
attributes	Align (left,center,right)
annotation	documentation Declaring Abbreviation definition
source	<pre> &lt;xsd:element name="abbrevdef"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Declaring Abbreviation definition&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence&gt;       &lt;xsd:annotation&gt;         &lt;xsd:documentation&gt;sequence&lt;/xsd:documentation&gt;       &lt;/xsd:annotation&gt;       &lt;xsd:element ref="name" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="body"/&gt;     &lt;/xsd:sequence&gt;     &lt;xsd:attribute name="align" use="optional" default="left"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:NMTOKEN"&gt;           &lt;xsd:enumeration value="left"/&gt;           &lt;xsd:enumeration value="center"/&gt;           &lt;xsd:enumeration value="right"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>

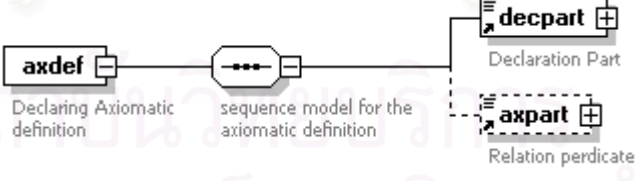
รูปที่ ก-1 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Abbrevdef

2. ส่วนย่อย Application เป็นส่วนย่อยของการแบ่งข้อมูลส่วนที่เป็นเงื่อนไขในระบบมีรายละเอียดของโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-2

diagram	
children	<b>name formals</b>
used by	elements <b>declaration predicate</b>
source	<pre> &lt;xsd:element name="application"&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence minOccurs="0" maxOccurs="unbounded"&gt;       &lt;xsd:element ref="name" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="formals" minOccurs="0" maxOccurs="unbounded"/&gt;     &lt;/xsd:sequence&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>

รูปที่ ก-2 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Application

3. ส่วนย่อย Axdef เป็นส่วนย่อยนิยามโครงสร้างแบบ Axiomatic form มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-3

diagram	
children	<b>decpart axpart</b>
used by	element <b>Zedml</b>
attributes	Align (left,center,right)
annotation	documentation Declaring Axiomatic definition

รูปที่ ก-3 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Axdef

source	<pre> &lt;xsd:element name="axdef"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Declaring Axiomatic definition&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:sequence&gt;       &lt;xsd:annotation&gt;         &lt;xsd:documentation&gt;sequence model for the axiomatic definition&lt;/xsd:documentation&gt;       &lt;/xsd:annotation&gt;       &lt;xsd:element ref="decpart"/&gt;       &lt;xsd:element ref="axpart" minOccurs="0"/&gt;     &lt;/xsd:sequence&gt;     &lt;xsd:attribute name="id" type="xsd:ID" use="optional"/&gt;     &lt;xsd:attribute name="group" type="xsd:NMTOKEN" use="optional"/&gt;     &lt;xsd:attribute name="align" use="optional" default="left"&gt;     &lt;xsd:simpleType&gt;       &lt;xsd:restriction base="xsd:NMTOKEN"&gt;         &lt;xsd:enumeration value="left"/&gt;         &lt;xsd:enumeration value="center"/&gt;         &lt;xsd:enumeration value="right"/&gt;       &lt;/xsd:restriction&gt;     &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>
--------	--

รูปที่ ก-3 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Axdef (ต่อ)

4. ส่วนย่อย Axpart เป็นส่วนย่อยนิยามลำดับของส่วนสัจพจน์ (Axiom part) มีรายละเอียด

โครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-4

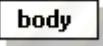
diagram	
children	<b>predicate</b>
used by	elements <b>axdef gendef schemadef</b>
annotation	documentation Relation perdicat

รูปที่ ก-4 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Axpart

source	<pre>&lt;xsd:element name="axpart"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Relation perdicate&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence&gt;       &lt;xsd:element ref="predicate" maxOccurs="unbounded"/&gt;     &lt;/xsd:sequence&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>
--------	--


รูปที่ ก-4 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Axpart (ต่อ)

## 5. ส่วนย่อย body มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-5

diagram	
used by	element <a href="#">abbrevdef</a>
source	<pre>&lt;xsd:element name="body"/&gt;</pre>

รูปที่ ก-5 โครงสร้างและรหัสต้นฉบับของส่วนย่อย body

## 6. ส่วนย่อย branch มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-6

diagram	
type	extension of <code>xsd:string</code>
used by	element <a href="#">freetypedef</a>
annotation	documentation <code>branch(1)..branch(n)</code>
source	<pre>&lt;xsd:element name="branch"&gt;   &lt;xsd:complexType&gt;     &lt;xsd:simpleContent&gt;       &lt;xsd:extension base="xsd:string"/&gt;     &lt;/xsd:simpleContent&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>

รูปที่ ก-6 โครงสร้างและรหัสต้นฉบับของส่วนย่อย branch

## 7. ส่วนย่อย declaration มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-7

diagram	
children	<u>inclusion</u> <u>declname</u> <u>application</u>
used by	elements <u>declarations</u> <u>decpart</u>
source	<pre> &lt;xsd:element name="declaration"&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence maxOccurs="unbounded"&gt;       &lt;xsd:annotation&gt;         &lt;xsd:documentation&gt;sequence&lt;/xsd:documentation&gt;       &lt;/xsd:annotation&gt;       &lt;xsd:element ref="inclusion" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="declname" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="application" minOccurs="0" maxOccurs="unbounded"/&gt;     &lt;/xsd:sequence&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>

รูปที่ ก-7 โครงสร้างและรหัสต้นฉบับของส่วนย่อย declaration

## 8. ส่วนย่อย declarations มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-8

diagram									
children	<u>nametype</u> <u>declaration</u>								
used by	element <u>Zedml</u>								
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> </tr> </thead> <tbody> <tr> <td>align</td> <td>xsd:NMTOKEN</td> <td>optional</td> <td>left</td> </tr> </tbody> </table>	Name	Type	Use	Default	align	xsd:NMTOKEN	optional	left
Name	Type	Use	Default						
align	xsd:NMTOKEN	optional	left						

Fixed

รูปที่ ก-8 โครงสร้างและรหัสต้นฉบับของส่วนย่อย declarations



annotation	documentation Declaring Circular Declarations
source	<pre> &lt;xsd:element name="declarations"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Declaring Circular Declarations&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:sequence&gt;       &lt;xsd:element ref="nametype"/&gt;       &lt;xsd:element ref="declaration" maxOccurs="unbounded"/&gt;     &lt;/xsd:sequence&gt;     &lt;xsd:attribute name="align" use="optional" default="left"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:NMTOKEN"&gt;           &lt;xsd:enumeration value="left"/&gt;           &lt;xsd:enumeration value="center"/&gt;           &lt;xsd:enumeration value="right"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>

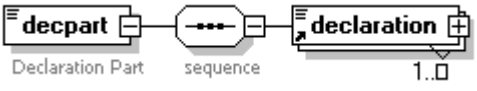
รูปที่ ก-8 โครงสร้างและรหัสต้นฉบับของส่วนย่อย declarations (ต่อ)

## 9. ส่วนย่อย declname มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-9

diagram	
children	<b>name formals</b>
used by	elements <b>declaration predicate</b>
source	<pre> &lt;xsd:element name="declname"&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence minOccurs="0" maxOccurs="unbounded"&gt;       &lt;xsd:element ref="name" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="formals" minOccurs="0"/&gt;     &lt;/xsd:sequence&gt;   &lt;/xsd:complexType&gt;&lt;/xsd:element&gt; </pre>


รูปที่ ก-9 โครงสร้างและรหัสต้นฉบับของส่วนย่อย declname

10. ส่วนย่อย decpart มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-10

diagram	
children	<a href="#">declaration</a>
used by	elements <a href="#">axdef</a> <a href="#">gendef</a> <a href="#">schemadef</a>
annotation	documentation Declaration Part
source	<pre>&lt;xsd:element name="decpart"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Declaration Part&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence&gt;       &lt;xsd:annotation&gt;         &lt;xsd:documentation&gt;sequence&lt;/xsd:documentation&gt;       &lt;/xsd:annotation&gt;       &lt;xsd:element ref="declaration" maxOccurs="unbounded"/&gt;     &lt;/xsd:sequence&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>

รูปที่ ก-10 โครงสร้างและรหัสต้นฉบับของส่วนย่อย declpart

11. ส่วนย่อย formal มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-11

diagram	
used by	elements <a href="#">application</a> <a href="#">declname</a> <a href="#">gendef</a>
source	<pre>&lt;xsd:element name="formals"/&gt;</pre>

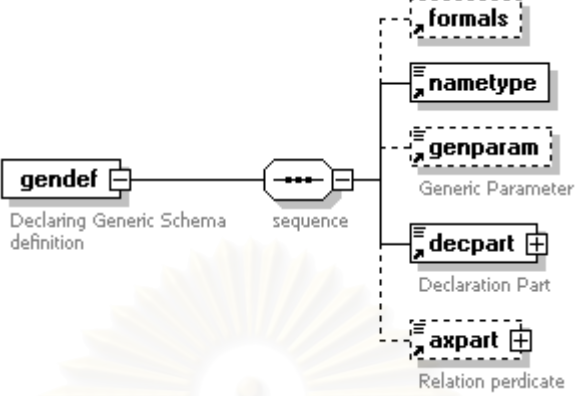
รูปที่ ก-11 โครงสร้างและรหัสต้นฉบับของส่วนย่อย formal

## 12. ส่วนย่อย freetypedef มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-12

diagram	
children	<u>typename</u> <u>branch</u>
used by	element <u>Zedml</u>
attributes	Align (left,center,right)
annotation	documentation Declaring free types definition
source	<pre> &lt;xsd:element name="freetypedef"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Declaring free types definition&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:sequence&gt;       &lt;xsd:annotation&gt;         &lt;xsd:documentation&gt;sequence model&lt;/xsd:documentation&gt;       &lt;/xsd:annotation&gt;       &lt;xsd:element ref="typename"/&gt;       &lt;xsd:element ref="branch" maxOccurs="unbounded"/&gt;     &lt;/xsd:sequence&gt;     &lt;xsd:attribute name="align" use="optional" default="left"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:NMTOKEN"&gt;           &lt;xsd:enumeration value="left"/&gt;           &lt;xsd:enumeration value="center"/&gt;           &lt;xsd:enumeration value="right"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>


รูปที่ ก-12 โครงสร้างและรหัสต้นฉบับของส่วนย่อย freetypedef

## 13. ส่วนย่อย gendef มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-13

diagram	
children	<u>formals</u> <u>nametype</u> <u>genparam</u> <u>decpart</u> <u>axpart</u>
used by	element <u>Zedml</u>
attributes	Align (left,center,right)
annotation	documentation Declaring Generic Schema definition
source	<pre> &lt;xsd:element name="gendef"&gt;   &lt;xsd:complexType&gt;     &lt;xsd:sequence&gt;       &lt;xsd:element ref="formals" minOccurs="0"/&gt;       &lt;xsd:element ref="nametype"/&gt;       &lt;xsd:element ref="genparam" minOccurs="0"/&gt;       &lt;xsd:element ref="decpart"/&gt;       &lt;xsd:element ref="axpart" minOccurs="0"/&gt;     &lt;/xsd:sequence&gt;     &lt;xsd:attribute name="align" use="optional" default="left"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:NMTOKEN"&gt;           &lt;xsd:enumeration value="left"/&gt;           &lt;xsd:enumeration value="center"/&gt;           &lt;xsd:enumeration value="right"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>


รูปที่ ก-13 โครงสร้างและรหัสต้นฉบับของส่วนย่อย gendef

14. ส่วนย่อย genparam มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-14

diagram	 Generic Parameter
type	extension of <code>xsd:string</code>
used by	element <code>gendef</code>
annotation	documentation Generic Parameter
source	<pre>&lt;xsd:element name="genparam"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Generic Parameter&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:simpleContent&gt;       &lt;xsd:extension base="xsd:string"/&gt;     &lt;/xsd:simpleContent&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>

รูปที่ ก-14 โครงสร้างและรหัสต้นฉบับของส่วนย่อย genparam

15. ส่วนย่อย givendef มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-15

diagram	 Declaring Given types paragraph
type	extension of <code>xsd:anySimpleType</code>
used by	element <code>Zedml</code>
attributes	Align (left,center,right)
annotation	documentation Declaring Given types paragraph

รูปที่ ก-15 โครงสร้างและรหัสต้นฉบับของส่วนย่อย givendef

source	<pre> &lt;xsd:element name="givendef"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Declaring Given types paragraph&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:simpleContent&gt;       &lt;xsd:extension base="xsd:anySimpleType"&gt;         &lt;xsd:attribute name="id" type="xsd:anySimpleType" use="optional"/&gt;         &lt;xsd:attribute name="group" type="xsd:anySimpleType" use="optional"/&gt;         &lt;xsd:attribute name="align" use="optional" default="left"&gt;           &lt;xsd:simpleType&gt;             &lt;xsd:restriction base="xsd:NMTOKEN"&gt;               &lt;xsd:enumeration value="left"/&gt;               &lt;xsd:enumeration value="center"/&gt;               &lt;xsd:enumeration value="right"/&gt;             &lt;/xsd:restriction&gt;           &lt;/xsd:simpleType&gt;         &lt;/xsd:attribute&gt;       &lt;/xsd:extension&gt;     &lt;/xsd:simpleContent&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>
--------	--

รูปที่ ก-15 โครงสร้างและรหัสต้นฉบับของส่วนย่อย givendef (ต่อ)

## 16. ส่วนย่อย inclusion มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-16

diagram	
children	<b>name</b>
used by	element <b>declaration</b>
source	<pre> &lt;xsd:element name="inclusion"&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence minOccurs="0" maxOccurs="unbounded"&gt;       &lt;xsd:element ref="name" minOccurs="0" maxOccurs="unbounded"/&gt;     &lt;/xsd:sequence&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>

รูปที่ ก-16 โครงสร้างและรหัสต้นฉบับของส่วนย่อย inclusion

17. ส่วนย่อย name มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-17

diagram	
children	<u>sub</u> <u>sup</u>
used by	elements <a href="#">abbrevdef</a> <a href="#">application</a> <a href="#">declname</a> <a href="#">inclusion</a>
source	<pre>&lt;xsd:element name="name"&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:choice minOccurs="0" maxOccurs="unbounded"&gt;       &lt;xsd:element ref="sub" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="sup" minOccurs="0" maxOccurs="unbounded"/&gt;     &lt;/xsd:choice&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>

รูปที่ ก-17 โครงสร้างและรหัสต้นฉบับของส่วนย่อย name

18. ส่วนย่อย nametype มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-18

diagram	
type	xsd:string
used by	elements <a href="#">declarations</a> <a href="#">gendif</a> <a href="#">schemadef</a>
source	<pre>&lt;xsd:element name="nametype" type="xsd:string"/&gt;</pre>

รูปที่ ก-18 โครงสร้างและรหัสต้นฉบับของส่วนย่อย nametype

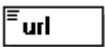
## 19. ส่วนย่อย note มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-19

diagram	
children	<a href="#">url</a>
used by	element <a href="#">Zedml</a>
attributes	Align (left,center,right)
source	<pre> &lt;xsd:element name="note"&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:sequence&gt;       &lt;xsd:element name="url" minOccurs="0" maxOccurs="unbounded"&gt;         &lt;xsd:complexType&gt;           &lt;xsd:simpleContent&gt;             &lt;xsd:extension base="xsd:string"&gt;               &lt;xsd:attribute name="href" type="xsd:anyURI" use="optional"/&gt;               &lt;xsd:attribute name="hostname" type="xsd:string" use="optional"/&gt;             &lt;/xsd:extension&gt;           &lt;/xsd:simpleContent&gt;         &lt;/xsd:complexType&gt;       &lt;/xsd:element&gt;     &lt;/xsd:sequence&gt;     &lt;xsd:attribute name="align" use="optional" default="left"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:NMTOKEN"&gt;           &lt;xsd:enumeration value="left"/&gt;           &lt;xsd:enumeration value="center"/&gt;           &lt;xsd:enumeration value="right"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>

รูปที่ ก-19 โครงสร้างและรหัสต้นฉบับของส่วนย่อย note




20. ส่วนย่อย url มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-20

diagram	
type	extension of <code>xsd:string</code>
attributes	Align (left,center,right)
source	<pre>&lt;xsd:element name="url" minOccurs="0" maxOccurs="unbounded"&gt;   &lt;xsd:complexType&gt;     &lt;xsd:simpleContent&gt;       &lt;xsd:extension base="xsd:string"&gt;         &lt;xsd:attribute name="href" type="xsd:anyURI" use="optional"/&gt;         &lt;xsd:attribute name="hostname" type="xsd:string" use="optional"/&gt;       &lt;/xsd:extension&gt;     &lt;/xsd:simpleContent&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>

รูปที่ ก-20 โครงสร้างและรหัสต้นฉบับของส่วนย่อย url

21. ส่วนย่อย opdec มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-21

diagram	 Declaring Infix Identifiers *attribute enumeration
used by	element <code>Zedml</code>
attributes	Align (left,center,right)
annotation	documentation Declaring Infix Identifiers *attribute enumeration

รูปที่ ก-21 โครงสร้างและรหัสต้นฉบับของส่วนย่อย opdec

source	<pre> &lt;xsd:element name="opdec"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Declaring Infix Identifiles *attribute enumeration&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:attribute name="optype" use="required"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:string"&gt;           &lt;xsd:enumeration value="ingen"/&gt;           &lt;xsd:enumeration value="inrel"/&gt;           &lt;xsd:enumeration value="pregn"/&gt;           &lt;xsd:enumeration value="prerel"/&gt;           &lt;xsd:enumeration value="postfun"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>
--------	---

รูปที่ ก-21 โครงสร้างและรหัสต้นฉบับของส่วนย่อย opdec (ต่อ)

## 22. ส่วนย่อย predicate มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-22

diagram	
children	<b>declname application</b>
used by	element <b>axpart</b>
attributes	Align (left,center,right)
annotation	documentation *attribute

รูปที่ ก-22 โครงสร้างและรหัสต้นฉบับของส่วนย่อย predicate

source	<pre> &lt;xsd:element name="predicate"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;*attribute &lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:choice maxOccurs="unbounded"&gt;       &lt;xsd:element ref="declname" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="application" minOccurs="0" maxOccurs="unbounded"/&gt;     &lt;/xsd:choice&gt;     &lt;xsd:attribute name="label" type="xsd:anySimpleType" use="optional"/&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>
--------	--

รูปที่ ก-22 โครงสร้างและรหัสต้นฉบับของส่วนย่อย predicate (ต่อ)

23. ส่วนย่อย schemadef มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-23

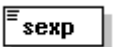
diagram	
children	<b>sexp</b> <b>nametype</b> <b>decpart</b> <b>axpart</b>
used by	element <b>Zedml</b>
attributes	Align (left,center,right)
annotation	documentation Declaring Schema definition *attribute enumeration

รูปที่ ก-23 โครงสร้างและรหัสต้นฉบับของส่วนย่อย schemadef

source	<pre> &lt;xsd:element name="schemadef"&gt;   &lt;xsd:complexType mixed="true"&gt;     &lt;xsd:choice&gt;       &lt;xsd:element ref="sexp"/&gt;       &lt;xsd:sequence&gt;         &lt;xsd:element ref="nametype"/&gt;         &lt;xsd:element ref="decpart"/&gt;         &lt;xsd:element ref="axpart"/&gt;       &lt;/xsd:sequence&gt;     &lt;/xsd:choice&gt;     &lt;xsd:attribute name="style" use="required" default="vert"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:string"&gt;           &lt;xsd:enumeration value="vert"/&gt;           &lt;xsd:enumeration value="horiz"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;     &lt;xsd:attribute name="purpose" use="required"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:string"&gt;           &lt;xsd:enumeration value="state"/&gt;           &lt;xsd:enumeration value="operation"/&gt;           &lt;xsd:enumeration value="datatype"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;     &lt;xsd:attribute name="align" use="optional" default="left"&gt;       &lt;xsd:simpleType&gt;         &lt;xsd:restriction base="xsd:NMTOKEN"&gt;           &lt;xsd:enumeration value="left"/&gt;           &lt;xsd:enumeration value="center"/&gt;           &lt;xsd:enumeration value="right"/&gt;         &lt;/xsd:restriction&gt;       &lt;/xsd:simpleType&gt;     &lt;/xsd:attribute&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>
--------	---


รูปที่ ก-23 โครงสร้างและรหัสต้นฉบับของส่วนย่อย schemadef (ต่อ)

24. ส่วนย่อย sexp มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-24

diagram	
type	extension of <code>xsd:string</code>
used by	element <code>schemadef</code>
source	<pre>&lt;xsd:element name="sexp"&gt;   &lt;xsd:complexType&gt;     &lt;xsd:simpleContent&gt;       &lt;xsd:extension base="xsd:string"/&gt;     &lt;/xsd:simpleContent&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>

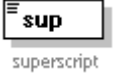
รูปที่ ก-24 โครงสร้างและรหัสต้นฉบับของส่วนย่อย sexp

25. ส่วนย่อย sub มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-25

diagram	 subscript
type	extension of <code>xsd:string</code>
used by	element <code>name</code>
annotation	documentation subscript
source	<pre>&lt;xsd:element name="sub"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;subscript&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:simpleContent&gt;       &lt;xsd:extension base="xsd:string"/&gt;     &lt;/xsd:simpleContent&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>


รูปที่ ก-25 โครงสร้างและรหัสต้นฉบับของส่วนย่อย sub

26. ส่วนย่อย sup มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-26

diagram	
type	extension of <code>xsd:string</code>
used by	element <code>name</code>
annotation	documentation <code>superscript</code>
source	<pre>&lt;xsd:element name="sup"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;superscript&lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:simpleContent&gt;       &lt;xsd:extension base="xsd:string"/&gt;     &lt;/xsd:simpleContent&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt;</pre>

รูปที่ ก-26 โครงสร้างและรหัสต้นฉบับของส่วนย่อย sup

27. ส่วนย่อย typename มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-27

diagram	
type	<code>xsd:string</code>
used by	element <code>freedef</code>
source	<pre>&lt;xsd:element name="typename" type="xsd:string"/&gt;</pre>

รูปที่ ก-27 โครงสร้างและรหัสต้นฉบับของส่วนย่อย type

## 28. ส่วนย่อย Zedmi มีรายละเอียดโครงสร้างและรหัสต้นฉบับ ดังรูปที่ ก-28

diagram	<p>The diagram illustrates the structure of the <b>Zedmi</b> root element. It is a choice model containing the following child elements:</p> <ul style="list-style-type: none"> <li><b>opdec</b> (0..0): Declaring Infix Identifiers *attribute enumeration</li> <li><b>givendef</b> (0..0): Declaring Given types paragraph</li> <li><b>axdef</b> (0..0): Declaring Axiomatic definition</li> <li><b>schemadef</b> (0..0): Declaring Schema definition *attribute enumeration</li> <li><b>gendef</b> (0..0): Declaring Generic Schema definition</li> <li><b>abbrevdef</b> (0..0): Declaring Abbreviation definition</li> <li><b>freetypedef</b> (0..0): Declaring free types definition</li> <li><b>note</b> (0..0): (No description provided)</li> <li><b>declarations</b> (0..0): Declaring Circular Declarations</li> </ul> <p>The <b>Zedmi</b> element itself is described as: Root element, 0..0, Element declarations for higher-level structure of ZIF (choice model).</p>
children	<a href="#">opdec</a> <a href="#">givendef</a> <a href="#">axdef</a> <a href="#">schemadef</a> <a href="#">gendef</a> <a href="#">abbrevdef</a> <a href="#">freetypedef</a> <a href="#">note</a> <a href="#">declarations</a>
annotation	documentation Root element

รูปที่ ก-28 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Zedmi

source	<pre> &lt;xsd:element name="Zedml"&gt;   &lt;xsd:annotation&gt;     &lt;xsd:documentation&gt;Root element &lt;/xsd:documentation&gt;   &lt;/xsd:annotation&gt;   &lt;xsd:complexType&gt;     &lt;xsd:choice minOccurs="0" maxOccurs="unbounded"&gt;       &lt;xsd:annotation&gt;         &lt;xsd:documentation&gt;Element declarations for higher-level structure of ZIF         (choice model) &lt;/xsd:documentation&gt;       &lt;/xsd:annotation&gt;       &lt;xsd:element ref="opdec" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="givendef" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="axdef" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="schemadef" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="gendef" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="abbrevdef" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="freetypedef" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="note" minOccurs="0" maxOccurs="unbounded"/&gt;       &lt;xsd:element ref="declarations" minOccurs="0" maxOccurs="unbounded"/&gt;     &lt;/xsd:choice&gt;   &lt;/xsd:complexType&gt; &lt;/xsd:element&gt; </pre>
--------	--

รูปที่ ก-28 โครงสร้างและรหัสต้นฉบับของส่วนย่อย Zedml (ต่อ)



## ภาคผนวก ข. เอกสารเซตเอ็กซ์เอ็มแอลดีทีดี

เป็นเอกสารที่นิยามชื่อของเอนทิตีที่เป็นสัญลักษณ์ทางคณิตศาสตร์ที่ใช้เขียนข้อกำหนด  
รูปร่างกับค่าของชุดอักขระยูนิโคด เพื่อที่จะใช้ในการอ้างอิงในเอกสารเอ็กซ์เอ็มแอลและแสดงผลรูป  
สัญลักษณ์เหล่านั้นบนเว็บเบราว์เซอร์ต่อไป ซึ่งรายละเอียดของการนิยามชื่อของเอนทิตีสามารถ  
แบ่งได้ตามกลุ่มของสัญลักษณ์ทางคณิตศาสตร์ที่ใช้ในการเขียนข้อกำหนดรูปร่าง ตั้งแต่ตารางที่  
ข-1 ถึงตารางที่ ข-10 ต่อไปนี้

ตารางที่ ข-1 กลุ่มสัญลักษณ์การนิยาม (Definitions)

ชื่อรูปร่าง	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Abbreviated definition	abbdef	$==$	&#61;&#61;
Free type definition	Freedef	$::=$	&#58;&#58;&#61;
Schema definition	sdef		&#8793;

ตารางที่ ข-2 กลุ่มสัญลักษณ์ทางตรรกะ (Logic)

ชื่อรูปร่าง	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Logical true constant	true	$true$	&true;
Logical false constant	Freedef	$false$	&false;
Negation	not	$!$	&#172;
Conjunction	and	$f$	&#8896;
Disjunction	or	$\mathfrak{w}$	&#8897;
Implication	rArr	$\Rightarrow$	&#8658;
Equivalence	iff	$\Leftrightarrow$	&#8660;
Universal quantifier	forall	$A$	&#8704;
Existential quantifier	exist	$E$	&#8707;
Unique existential quantifier	exist1	$E_1$	&#8707;&#x2081;
Not existential quantifier	nexist	$O$	&#8708;
Local definition	let	$let$	&let;

ตารางที่ ข-3 กลุ่มสัญลักษณ์ เซต และนิพจน์ (Sets and Expressions)

ชื่อรูปย่อ	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Equality	eq	=	&#61;
Inequality	ne	$\neq$	&#8800;
Membership	isin	$\in$	&#8712;
Non membership	notin	$\notin$	&#8713;
Empty set	empty	$\emptyset$	&#12299;
Left Set bracket	lcub	{	&#123;
Right Set bracket	rcub	}	&#125;
Proper subset	sub	$\subset$	&#8834;
Subset	sube	$\subseteq$	&#8838;
Lambda	lambda	$\lambda$	&#955;
mu	mu	$\mu$	&#956;
Cartesian product	prod	$\times$	&#9747;
Power set	pset	$\mathcal{P}$	&#8473;
Non empty subset	pset1	$\mathcal{P}_1$	&#8473;&#x2081;
Finite set	fset	$\Phi$	&#70;
None empty finite set	fset1	$\Phi_1$	&#70;&#x2081;
Set union	cup	$\cup$	&#8745;
Set intersection	cap	$\cap$	&#8746;
Set difference	sdif	$\setminus$	&#8726;
Cardinality	num	#	&#35;

ตารางที่ ข-4 กลุ่มสัญลักษณ์ ความสัมพันธ์ (Relations)

ชื่อรูปย่อ	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Binary relation	rel	$\varphi$	&#8596;
Maplet	map	$\square$	&#8614;
Domain	dom	<i>dom</i>	&dom;
Range	ran	<i>ran</i>	&ran;
Identity	id	<i>id</i>	&id;

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Composition (backward)	comp	;	&#59;
Composition (forward)	compfn	o	&#8408;
Domain restriction	dres	p	&#8882;
Range restriction	rres	tau	&#8883;
Domain subtraction	dsub	psi	&#9668;
Range subtraction	rsub	upsilon	&#9010;
Relational inverse	tilde	~	&#732;
Transitive closure	tcl	+	&#726;
Reflexive-transitive closure	rtcl	*	&#42;
Functional override	oplus	±	&#8853;
Left relational image bracket	limg	<	&#12312;
Right relational image bracket	rimg	Ⓜ	&#12313;

ตารางที่ ข-5 กลุ่มสัญลักษณ์ ฟังก์ชัน (Functions)

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Partial functions	pfun	♣	&#8603;
Total function	tfun	phi	&#8594;
Partial injections	pinj	♥	&#8827;&#8603;
Total injections	tinj	⊗	&#8611;
Total surjections	tsur	⊘	&#8608;
bijections	bij	℔	&#8827;&#8608;

ตารางที่ ข-6 กลุ่มสัญลักษณ์ จำนวน (Number)

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Integers	Int	Z	&#8484;
Natural numbers	Nat	N	&#8469;
Strictly positive integers	Nat1	N <sub>1</sub>	&#8469;&#x2081;
Real numbers	real	P	&#8477;

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Less than	lt	<	&#x2039;
Less than or equal to	le	≤	&#8804;
Greater than or equal to	ge	≥	&#8805;
Greater than	gt	>	&#62;
Number range	upto	..	&#8229;
Division	div	<i>div</i>	&div;
Modulo arithmetic	mod	<i>mod</i>	&mod;
Minimum of a set of number	min	<i>min</i>	&min;
Maximum of a set of number	max	<i>max</i>	&max;
Successor function	succ	<i>succ</i>	&succ;

ตารางที่ ข-7 กลุ่มสัญลักษณ์ แบ็กส์ (Bags)

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Left bag bracket	lbag	Ⓛ	&#x301a;
Right bag bracket	rbag	Ⓜ	&#x301b;
Empty bag	embag	Ⓛ Ⓜ	&#x301a;&#x301b;
Bag scaling of multiplicity	bscal	Υ	&#8855;
Bag union	bunion	⊕	&#8846;

ตารางที่ ข-8 กลุ่มสัญลักษณ์ สัญนิยม (Conventions)

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Input to an opration	Input	?	&#63;
Output from operation	output	!	&#33;
State after an operation	afterst	⋅	&#39;

ตารางที่ ข-9 กลุ่มสัญลักษณ์ ลำดับ (Sequence)

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Set of finite Sequence	seq	<i>seq</i>	&seq;
None empty finite sequence	seq <sub>1</sub>	<i>seq1</i>	&seq;&#x2081;

Finite injective sequence	iseq	$iseq$	&iseq;
Empty sequence	emseq	$\emptyset$	&#9001;&#9002;
Left sequence bracket	lseq	$\langle$	&#9001;
Right sequence bracket	rseq	$\rangle$	&#9002;
Filter	filter	$\mathfrak{F}$	&#8638;
Concatenation	cat	$\perp$	&#8256;
Distributed concatenation	dcat	$\perp/$	&#8256;&#8260;
First element of sequence	head	$head$	&head;
All but the head of sequence	tail	$tail$	&tail
Last element of sequence	last	$last$	&last;
All but the last of sequence	front	$front$	&front;
Reverse a sequence	rev	$rev$	&rev;
Compact function	squash	$squash$	&squash;

ตารางที่ ข-10 กลุ่มสัญลักษณ์ เค้ว่างและทั่วไป (Schemas and Miscellaneous)

ชื่อกรุปน้อย	ชื่อเอนทีตี้	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Delta	delta	$\Delta$	&#8710;
Xi	xi	$\Xi$	&#926;
Theta	theta	$\theta$	&#x03b8;
Alpha	alpha	$\alpha$	&#945;
Omega	Omega	$\Omega$	&#936;
Epsilon	epsi	$\epsilon$	&#949;
Zeta	zeta	$\zeta$	&#950;
Bata	beta	$\beta$	&#946;
Eta	eta	$\eta$	&#951;
Phi	phis	$\phi$	&#966;
omega	omega	$\omega$	&#969;
Upsilon	Upsi	$\Upsilon$	&#978;
Psi	Psi	$\Psi$	&#968;

ชื่อกรุปน้อย	ชื่อเอนทิตี	สัญลักษณ์	ค่าของชุดอักขระยูนิโคด
Sigma	Sigma	$\Sigma$	&#8721;
Pi	Pi	$\Pi$	&#8719;
Left square bracket	lsqb	[	&#91;
Right square bracket	rsqb	]	&#93;
Bar	verbar		&#124;
Left chevron bracket	lchev	$\Leftarrow$	&#12298;
Right chevron bracket	rchev	$\Rightarrow$	&#12299;
Fat dot	bull	$\bullet$	&#9679;
Colon	col	:	&#58;
Comma	com	,	&#44;
Left tuple bracket	ltup	(	&#40;
Right tuple bracket	rtup	)	&#41;
Left Set bracket	lcub	{	&#123;
Right Set bracket	rcub	}	&#125;
Binding	rarrw	$\rightsquigarrow$	&#8605;
Schema Projection	proj	$\wp$	&#8638;
Turnstile	turn	$\perp$	&#8866;

การนิยามข้อมูลชื่อเอนทิตีกับค่าของชุดอักขระยูนิโคดในเอกสาร เซดเอ็กซ์เอ็มแอล ดีทีดี  
ที่ชื่อว่า zedsymbol.dtd จะมีรายละเอียดของรหัสต้นฉบับแสดงได้ในรูปที่ ข-1

จุฬาลงกรณ์มหาวิทยาลัย

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!--Glossary of Z notation-->
<!-- Complete repertoire of characters from Unicode -->
<!--Definitions-->
<!ENTITY abbdef "&#61;&#61;,">
<!ENTITY freedef "&#58;&#58;&#61;,">
<!ENTITY verbar "&#124;,">
<!ENTITY sdef "&#8793;,">
<!--Logic-->
<!ENTITY true "true">
<!ENTITY false "false">
<!ENTITY not "&#172;,">
<!ENTITY and "&#8896;,">
<!ENTITY or "&#8897;,">
<!ENTITY rArr "&#8658;    ">
<!ENTITY iff "&#8660;,">
<!ENTITY forall "&#8704;,">
<!ENTITY exist "&#8707;,">
<!ENTITY exist1 "&#8707;&#x2081;,">
<!ENTITY nexist "&#8708;,">
<!ENTITY let "let">
<!-- Sets and Expressions -->
<!ENTITY isin "&#8712;,">
<!ENTITY notin "&#8713;,">
<!ENTITY empty "&#8709;,">
<!ENTITY lcub "&#123;,">
<!ENTITY rcub "&#125;,">
<!ENTITY sub "&#8834;,">
<!ENTITY sube "&#8838;,">
<!ENTITY lambda "&#955;,">
<!ENTITY mu "&#956;,">
<!ENTITY prod "&#9747;,">
<!ENTITY pset "&#8473;,">
<!ENTITY pset1 "&#8473;&#x2081;,">
<!ENTITY fset "&#70;,">
<!ENTITY fset1 "&#70;&#x2081;,">
<!ENTITY cup "&#8746;,">
<!ENTITY cap "&#8745;    ">
<!ENTITY sdiff "&#8726;,">

```

รูปที่ ข-1 การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคดในเอกสารเซตเอ็กซ์เอ็มแอลดีทีดี

```

<!ENTITY num "#35;">
<!ENTITY sup "#8835; ">
<!ENTITY supe "#8839;">
<!-- Relations -->
<!ENTITY rel "#8596;">
<!ENTITY map "#8614;">
<!ENTITY dom "dom">
<!ENTITY ran "ran">
<!ENTITY id "id">
<!ENTITY comp "#59;">
<!ENTITY compfn "#8408;">
<!ENTITY dres "#8882;">
<!ENTITY rres "#8883;">
<!ENTITY dsub "#9668;">
<!ENTITY rsub "#9010;">
<!ENTITY tilde "#732;">
<!ENTITY tcl "#x207a;">
<!ENTITY rtcl "#42;">
<!ENTITY oplus "#8853;">
<!ENTITY limg "#12312;">
<!ENTITY rimg "#12313;">
<!-- Functions -->
<!ENTITY pfun "#8603;">
<!ENTITY tfun "#8594;">
<!ENTITY pinj "#8827;#8603;">
<!ENTITY tinj "#8611;">
<!ENTITY tsur " #8608;">
<!ENTITY bij "#8827;#8608;">
<!-- Numbers -->
<!ENTITY real "#8477;">
<!ENTITY int "#8484;">
<!ENTITY nat "#8469;">
<!ENTITY nat1 "#8469;#x2081;">
<!ENTITY lt "#x2039;">
<!ENTITY le "#8804;">
<!ENTITY ge "#8805;">
<!ENTITY gt "#62;">
<!ENTITY upto "#8229;">
<!ENTITY min "min">

```

รูปที่ ข-1 การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคดในเอกสารเซดเอ็กซ์เอ็มแอลดีทีดี(ต่อ)



```

<!ENTITY max "max">
<!ENTITY succ "succ">
<!--Sequences -->
<!ENTITY seq "seq">
<!ENTITY seq1 "seq&#x2081;">
<!ENTITY iseq "iseq">
<!ENTITY emseq "&#9001;&#9002;">
<!ENTITY lseq "&#9001;">
<!ENTITY rseq "&#9002;">
<!ENTITY filter "&#8638;">
<!ENTITY cat "&#8256;">
<!ENTITY dcat "&#8256/">
<!ENTITY head "head">
<!ENTITY tail "tail">
<!ENTITY last "last">
<!ENTITY front "front">
<!ENTITY rev "rev">
<!ENTITY squash "squash">
<!-- Bags -->
<!ENTITY embag "&#x301a;&#x301b;">
<!ENTITY lbag "&#x301a;">
<!ENTITY rbag "&#x301b;">
<!ENTITY bunion "&#8846;">
<!ENTITY bscal "&#8855;">
<!--Conventions -->
<!ENTITY input "&#63;">
<!ENTITY output "&#33;">
<!ENTITY afterst "&#39;">
<!-- Miscellaneous-->
<!ENTITY delta "&#8710;">
<!ENTITY xi "&#926;">
<!ENTITY theta "&#x03b8;">
<!ENTITY alpha "&#945;">
<!ENTITY Omega "&#937;">
<!ENTITY epsi "&#949;">
<!ENTITY zeta "&#950;">
<!ENTITY beta "&#946;">
<!ENTITY eta "&#951;">
<!ENTITY phis "&#966;">
<!ENTITY omega "&#969;">

```

รูปที่ ข-1 การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคดในเอกสารเซดเอ็กซ์เอ็มแอลดีทีดี (ต่อ)

```

<!ENTITY Sigma "&#8721;">
<!ENTITY Pi "&#8719;">
<!ENTITY proj "&#8638;">
<!ENTITY Upsi "&#978;">
<!ENTITY Psi "&#968;">
<!ENTITY rarrw "&#8605;">
<!ENTITY lsqb "&#91;">
<!ENTITY rsqb "&#93;">
<!ENTITY lchev "&#12298;">
<!ENTITY rchev "&#12299;">
<!ENTITY bull "&#9679;">
<!ENTITY col "&#58;">
<!ENTITY com "&#44;">
<!ENTITY ltup "&#40;">
<!ENTITY rtup "&#41;">
<!ENTITY eq "&#61;">
<!ENTITY ne "&#8800;">
<!ENTITY turn "&#8866;">

```

รูปที่ ข-1 การนิยามชื่อเอนทิตีกับชุดอักขระยูนิโคดในเอกสารเซดเอ็กซ์เอ็มแอลที่ดี (ต่อ)

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## ภาคผนวก ค. เอกสารเซดเอ็กซ์เอสแอลสไต้ลส์ซีท

เอกสารเค้าร่างเซดเอ็กซ์เอสแอลสไต้ลส์ซีท เป็นโปรแกรมประยุกต์ประเภทหนึ่งของเอ็กซ์เอ็มแอลจึงมีวากยสัมพันธ์ตามข้อกำหนดของภาษาเอ็กซ์เอ็มแอล ซึ่งในงานวิทยานิพนธ์นี้ทำหน้าที่แปลงเอกสารข้อกำหนดรूपนัยภาษาเซดจากเอกสารเอ็กซ์เอ็มแอลให้เป็นเอกสาร เอชทีเอ็มแอล เพื่อแสดงผลบนเว็บเบราว์เซอร์ ทำหน้าที่จัดรูปแบบการแสดงผลโครงสร้างของเค้าร่างข้อกำหนด रूपนัยแบบต่างๆ และจัดรูปแบบของตัวอักษร การจัดตำแหน่งการแสดงผลโดยจะเลือกโครงสร้างแบบดังกล่าวจากแผ่นแบบ(Template) ที่ได้นิยามไว้ในเอกสารที่ชื่อว่า sedmlxslt.xslt ที่มีรหัสต้นฉบับ แสดงรายละเอียดของแผ่นแบบ แสดงได้ในรูปที่ ค-1

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsl:output method="html"/>
  <!-- start note template -->
  <xsl:template match="Zedml/note">
    <p>
      <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
      <xsl:for-each select="text()|url">
        <xsl:apply-templates select="."/>
      </xsl:for-each>
    </p>
  </xsl:template>
  <xsl:template match="url">
    <span style="font-style:italic; font-size:small">
      <a>
        <xsl:attribute name="href"><xsl:value-of select="@href"/></xsl:attribute>
        <xsl:value-of select="@hostname"/>
      </a>
    </span>
  </xsl:template>
  <!-- opdec template-->
  <xsl:template match="Zedml/opdec">
    <span style="font-style:italic; font-size:small">
      <xsl:value-of select="."/>
    </span>
  </xsl:template>
  <!--end notetemplate-->
```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลส์ซีท

```

<xsl:template match="Zedml">
  <html>
    <body>
      <xsl:apply-templates/>
    </body>
  </html>
</xsl:template>
<!-- start givendef template -->
<xsl:template match="Zedml/givendef">
  <div>
    <p>
      <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
      <table border="0" cellSpacing="0" cellPadding="0">
        <tr>
          <td></td>
          <td>
            <span style="font-style:italic; font-size:small">
              <xsl:value-of select="."/>
            </span>
          </td>
        </tr>
      </table>
    </p>
  </div>
</xsl:template>
<!--end givendef template-->
<!-- start Schemadef template-->
<xsl:template match="Zedml/schemadef">
  <xsl:choose>
    <xsl:when test="@style='vert'">
      <div>
        <p>
          <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
          <table border="0" cellSpacing="0" cellPadding="0">
            <tr>
              <td width="22" background="Leftop.jpg"/>
              <td>
                <span style="font-style:italic; font-size:small; white-space:nowrap">
                  <xsl:value-of select="nametype"/>
                </span>
              </td>
            </tr>
          </table>
        </p>
      </div>
    </xsl:when>
  </xsl:choose>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

        <td background="horiz.jpg" width="160"/>
    </tr>
    <xsl:for-each select="decpart/declaration">
        <xsl:for-each select="inclusion">
            <tr>
                <td background="vert.jpg"/>
                <td colspan="2">
                    <font style="font-size:small; font-style:italic">
                        <xsl:value-of select="."/>
                    </font>
                </td>
            </tr>
        </xsl:for-each>
    <tr>
        <td background="vert.jpg"/>
        <td colspan="2">
            <span style="font-style:italic; font-size:small">
                <xsl:value-of select="declname"/>
            </span>&#9;
            <xsl:variable name="app" select="application"/>
            <span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
                <!-- <font face="Arial Unicode MS" style="font-size:small; font-style:italic">-->
                    <xsl:value-of select="$app"/>
                </span>
            </td>
        </tr>
    </xsl:for-each>
    <!--template for st line-->
    <tr>
        <td width="20" height="23" background="middle.jpg"/>
        <td colspan="2" background="horiz.jpg"/>
    </tr>
    <!--end template-->
    <xsl:for-each select="axpart/predicate">
        <tr>
            <td background="vert.jpg"/>
            <td colspan="2">
                <span style="font-style:italic; font-size:small">
                    <xsl:value-of select="declname"/>
                </span>&#9;
            </td>
        </tr>
    </xsl:for-each>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท(ต่อ)

```

<xsl:variable name="app2" select="application"/>
<span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
  <xsl:value-of select="$app2"/>
</span>
</td>
</tr>
</xsl:for-each>
<!--template for labeling-->
<xsl:for-each select="axpart/predicate">
  <tr>
    <td background="vert.jpg"/>
    <td colspan="2">
      <xsl:if test="@label !=' '">
        <span style="font-style:italic; font-size:small;color:red">
          <xsl:text> /* </xsl:text>
          <xsl:value-of select="@label"/>
          <xsl:text> */</xsl:text>
        </span>
      </xsl:if>
    </td>
  </tr>
</xsl:for-each>
<!--template for end box-->
<tr>
  <td width="20" height="23" background="Leftbot.jpg"/>
  <td colspan="2" background="horiz.jpg"/>
</tr>
<!--end template-->
</table>
</p>
</div>
</xsl:when>
<!--template for written schema in horizontal form-->
<xsl:otherwise>
  <div>
    <p>
      <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
      <table border="0" cellSpacing="0" cellPadding="0">
        <tr>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

<td>
  <span style="font-style:italic; font-size:small; white-space:nowrap">
    <xsl:value-of select="nametype"/>
  </span>
</td>
<td width="10" align="center">
<span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
  &#x2259;
</span>
</td>
<td align="center">[
  <xsl:for-each select="decpart/declaration/inclusion">
    <font style="font-size:small; font-style:italic">
      <xsl:value-of select="."/>
    </font>
<span style="font-family:Arial Unicode MS; font-size:small; font-style:normal; font-variant:normal; font-weight:normal">
  &#59;&#9;</span>
</xsl:for-each>
  <xsl:for-each select="decpart/declaration[not(position()=last())]">
    <span style="font-style:italic; font-size:small">
      <xsl:value-of select="declname"/>
    </span>&#9;
    <xsl:variable name="app1" select="application"/>
  <span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
    <xsl:value-of select="$app1"/>
  </span>
  <span style="font-family:Arial Unicode MS; font-size:small; font-style:normal; font-variant:normal; font-weight:normal">
    &#59;&#9;</span>
  </xsl:for-each>
  <!--plus the last declaration-->
  <xsl:for-each select="decpart/declaration[last()]">
    <span style="font-style:italic; font-size:small">
      <xsl:value-of select="declname"/>
    </span>&#9;
    <xsl:variable name="app1" select="application"/>
  <span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
    <xsl:value-of select="$app1"/>
  </span>
  </xsl:for-each>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

</td>
<td align="center" width="10">&#9;&#9;</td>
<td>
  <xsl:for-each select="axpart/predicate[not(position()=last())]">
    <span style="font-style:italic; font-size:small">
      <xsl:value-of select="declname"/>
    </span>&#9;
    <xsl:variable name="app2" select="application"/>
<span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
  <xsl:value-of select="$app2"/>
</span>
<span style="font-family:Arial Unicode MS; font-size:small; font-style:normal; font-variant:normal; font-weight:normal">
  &#9;&#x22c0;&#9;</span>
</xsl:for-each>
<!-- plus the last predicate-->
<xsl:for-each select="axpart/predicate[last()]">
  <span style="font-style:italic; font-size:small">
    <xsl:value-of select="declname"/>
  </span>&#9;
  <xsl:variable name="app2" select="application"/>
<span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
  <xsl:value-of select="$app2"/>
</span>
</xsl:for-each>]
</td>
</tr>
</table>
</p>
</div>
</xsl:otherwise>
</xsl:choose>
</xsl:template>
<!-- end schemadef template-->
<!-- start axiomatic definition -->
<xsl:template match="Zedml/axdef">
  <div>
    <p>
      <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
      <table border="0" cellSpacing="0" cellPadding="0">
        <xsl:for-each select="decpart/declaration">

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)



```

<xsl:for-each select="inclusion">
  <tr>
    <td background="vert.jpg"/>
    <td colspan="2">
      <font style="font-size:small; font-style:italic">
        <xsl:value-of select="."/>
      </font>
    </td>
  </tr>
</xsl:for-each>
<tr>
  <td width="22" background="vert.jpg"/>
  <td colspan="2">
    <span style="font-style:italic; font-size:small">
      <xsl:value-of select="declname"/>
    </span>&#9;
    <xsl:variable name="app" select="application"/>
    <span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
      <xsl:value-of select="$app"/>
    </span>
  </td>
</tr>
</xsl:for-each>
<!--template for st line-->
<tr>
  <td width="22" height="23" background="middle.jpg"/>
  <td colspan="2" background="horiz.jpg"/>
</tr>
<!--end template-->
<xsl:for-each select="axpart/predicate">
  <tr>
    <td background="vert.jpg"/>
    <td colspan="2">
      <span style="font-style:italic; font-size:small">
        <xsl:value-of select="declname"/>
      </span>&#9;
      <xsl:variable name="app2" select="application"/>
      <span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
        <xsl:value-of select="$app2"/>
      </span>
    </td>
  </tr>
</xsl:for-each>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

        </td></tr>
    </xsl:for-each>
    <!--template for labeling-->
    <xsl:for-each select="axpart/predicate">
        <tr>
            <td background="vert.jpg"/>
            <td colspan="2">
                <xsl:if test="@label != ' '">
                    <span style="font-style:italic; font-size:small;color:red">
                        <xsl:text> /* </xsl:text>
                        <xsl:value-of select="@label"/>
                        <xsl:text> */</xsl:text>
                    </span>
                </xsl:if>
            </td>
        </tr>
    </xsl:for-each>
</table>
</p>
<br/>
</div>
</xsl:template>
<!-- end axiom definition template-->
<!-- start gendef template -->
<xsl:template match="Zedml/gendef">
    <div>
        <p>
            <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
            <table border="0" cellSpacing="0" cellPadding="0">
                <tr>
                    <td width="22" background="Leftop.jpg">
                    <td>
                        <span style="font-style:italic; font-size:small; white-space:nowrap">
                            <xsl:value-of select="nametype"/>
                        </span>
                        <xsl:variable name="param" select="genparam"/>
                        <xsl:if test="$param != "">
                            [
                                <span style="font-style:italic; font-size:small; white-space:nowrap">
                                    <xsl:value-of select="$param"/>
                                </span>
                            </xsl:if>
                        ]
                    </td>
                </tr>
            </table>
        </p>
    </div>
</xsl:template>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

        </td>
        <td background="horiz.jpg" width="160"/>
    </tr>
    <xsl:for-each select="decpart/declaration">
        <xsl:for-each select="inclusion">
            <tr>
                <td background="vert.jpg"/>
                <td colspan="2">
                    <font style="font-size:small; font-style:italic">
                        <xsl:value-of select="."/>
                    </font>
                </td></tr></xsl:for-each>
            <tr>
                <td background="vert.jpg"/>
                <td colspan="2">
                    <span style="font-style:italic; font-size:small">
                        <xsl:value-of select="declname"/>
                    </span>&#9;
                    <xsl:variable name="app" select="application"/>
                    <span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
                        <!-- <font face="Arial Unicode MS" style="font-size:small; font-style:italic"-->
                        <xsl:value-of select="$app"/>
                    </span>
                </td>
            </tr>
        </xsl:for-each>
        <!--template for st line-->
        <tr>
            <td width="20" height="23" background="middle.jpg"/>
            <td colspan="2" background="horiz.jpg"/>
        </tr>
        <!--end template-->
        <xsl:for-each select="axpart/predicate">
            <tr>
                <td background="vert.jpg"/>
                <td colspan="2">
                    <span style="font-style:italic; font-size:small">
                        <xsl:value-of select="declname"/>
                    </span>&#9;
                    <xsl:variable name="app2" select="application"/>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

<span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
    <xsl:value-of select="$app2"/>
</span>
</td>
</tr>
</xsl:for-each>
<!--template for labeling-->
<xsl:for-each select="axpart/predicate">
    <tr>
        <td background="vert.jpg"/>
        <td colspan="2">
            <xsl:if test="@label != "">
                <span style="font-style:italic; font-size:small;color:red">
                    <xsl:text> /* </xsl:text>
                    <xsl:value-of select="@label"/>
                    <xsl:text> */</xsl:text>
                </span>
            </xsl:if></td></tr>
        </xsl:for-each>
        <!--template for end box-->
        <tr>
            <td width="20" height="23" background="Leftbot.jpg"/>
            <td colspan="2" background="horiz.jpg"/>
        </tr>
        <!--end template-->
    </table>
</p>
</div>
</xsl:template>
<!-- end gendef template-->
<!-- start abbreviation definition template-->
<xsl:template match="Zedml/abbrevdef">
    <div>
        <p>
            <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
            <table border="0" cellSpacing="0" cellPadding="0">
                <tr>
                    <td>
                        <xsl:variable name="strbody" select="body"/>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

        <xsl:choose>
            <xsl:when test="$strbody="">
                <span style="font-family:Arial Unicode MS; font-size:x-small; font-size:small">
                    <xsl:value-of select="name"/>
                </span>&#9;
            </xsl:when>
            <xsl:otherwise>
                <span style="font-family:Arial Unicode MS; font-size:x-small;font-size:small">
                    <xsl:value-of select="name"/>
                </span>&#9;
            </xsl:otherwise>
        </xsl:choose>
        <strtext>= </strtext>&#9;
    <span style="font-family:Arial Unicode MS; font-size:small; font-style:normal; font-variant:normal; font-weight:normal">
        <xsl:value-of select="body"/>
    </span>
</xsl:otherwise>
</xsl:choose>
</td>
</tr>
</table>
</p>
</div>
</xsl:template>
<!-- start free type definition template-->
<xsl:template match="Zedml/freotypedef">
    <div>
        <p>
            <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
            <table border="0" cellSpacing="0" cellPadding="0">
                <tr>
                    <td>
                        <xsl:variable name="strbranch" select="branch"/>
                        <xsl:choose>
                            <xsl:when test="$strbranch="">
                                <span style="font-style:italic; font-size:small">
                                    <xsl:value-of select="typename"/>
                                </span>&#9;
                            </xsl:when>
                            <xsl:otherwise>
                                <span style="font-style:italic; font-size:small">
                                    <xsl:value-of select="typename"/>
                                </span>&#9;
                            </xsl:otherwise>
                        </xsl:choose>
                    </td>
                </tr>
            </table>
        </p>
    </div>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

        <xsl:for-each select="branch[not(position()=last())]">
<span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
            <xsl:value-of select="."/>
        </span> |
    </xsl:for-each>
    <xsl:for-each select="branch[position()=last()]">
<span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
            <xsl:value-of select="."/>
        </span>
    </xsl:for-each>
</xsl:otherwise>
</xsl:choose>
</td>
</tr>
</table>
</p>
</div>
</xsl:template>
<!--start for declarations circular form templaet-->
<xsl:template match="Zedml/declarations">
    <div>
        <p>
            <xsl:attribute name="align"><xsl:value-of select="@align"/></xsl:attribute>
            <table border="0" cellSpacing="0" cellPadding="0">
                <tr>
                    <td width="22" background="Leftop.jpg"/>
                    <td>
                        <span style="font-style:italic; font-size:small; white-space:nowrap">
                            <xsl:value-of select="nametype"/>
                        </span>
                    </td>
                    <td background="horiz.jpg" width="160"/>
                </tr>
                <xsl:for-each select="declaration">
                    <xsl:for-each select="inclusion">
                        <tr>
                            <td background="vert.jpg">
                            <td colspan="2">
                                <font style="font-size:small; font-style:italic">
                                    <xsl:value-of select="."/>
                                </font>
                            </td>
                        </tr>
                    </xsl:for-each>
                </xsl:for-each>
            </table>
        </p>
    </div>
</xsl:template>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

```

        </font>
    </td>
</tr>
</xsl:for-each>
<tr>
    <td background="vert.jpg"/>
    <td colspan="2">
        <span style="font-style:italic; font-size:small">
            <xsl:value-of select="declname"/>
        </span>&#9;
        <xsl:variable name="app" select="application"/>
    <span style="font-family:Arial Unicode MS; font-size:x-small; font-style:normal; font-variant:normal; font-weight:normal">
        <!-- <font face="Arial Unicode MS" style="font-size:small; font-style:italic"-->
        <xsl:value-of select="$app"/>
    </span>
    </td>
</tr>
</xsl:for-each>
<!--template for end box-->
<tr>
    <td width="20" height="23" background="Leftbot.jpg"/>
    <td colspan="2" background="horiz.jpg"/>
</tr>
<!--end template-->
</table>
</p>
</div>
</xsl:template>
</xsl:stylesheet>

```

รูปที่ ค-1 การนิยามรายละเอียดของแผ่นแบบในเอกสารเซดเอ็กซ์เอสแอลสไต้ลชีท (ต่อ)

จุฬาลงกรณ์มหาวิทยาลัย

## ภาคผนวก ง. กรณีศึกษาระบบ BirthdayBook

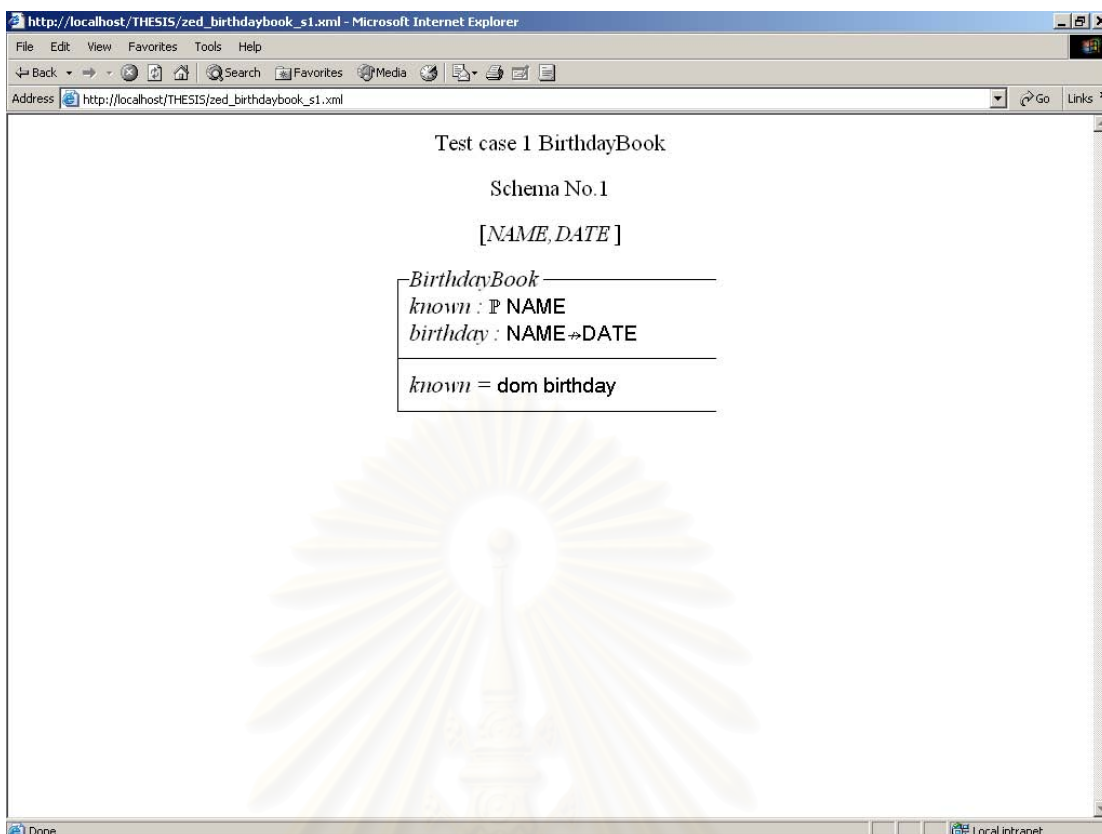
ระบบ BirthdayBook เป็นระบบที่ได้รับความนิยมที่จะนำมาศึกษาการเขียนข้อกำหนดรูปร่างภาษาเซต เป็นข้อกำหนดรูปร่างที่อธิบายการจัดเก็บข้อมูลวันเกิดของบุคคลทั่วไป ในระบบนี้มีการทดสอบการนำเสนอข้อกำหนดรูปร่างทั้งหมด 10 เค้าร่าง และจากการทดสอบระบบการนำเสนอข้อกำหนดรูปร่างภาษาเซต ด้วยเอ็กซ์เอ็มแอล บนเว็บเบราว์เซอร์สามารถแสดงได้ทุกเค้าร่างดังต่อไปนี้

1. เค้าร่างสถานะ (State Schema) เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-1 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-2

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
<note align="center">Test case 1 BirthdayBook</note>
<note align="center">Schema No.1</note>
<givendef align="center"> NAME,DATE</givendef>
<schemadef purpose="state" style="vert" align="center">
<nametype>BirthdayBook</nametype>
<decpart>
<declaration>
<declname>
<name>known</name> &col; </declname>
<application>&pset;
<name> NAME</name>
</application>
</declaration>
<declaration>
<declname>
<name>birthday</name> &col; </declname>
<application>
<name>NAME</name>&pfun; <name>DATE</name>
</application>
</declaration>
</decpart>
<axpart>
<predicate>
<declname>
<name>known </name>&eq; </declname>
<application>&dom; <name>birthday</name>
</application>
</predicate>
</axpart>
</schemadef>
</Zedml>
```

รูปที่ ง-1 รหัสต้นฉบับเค้าร่างสถานะ





รูปที่ ง-2 คำร่างสถานะ (State Schema) บนเว็บเบราว์เซอร์

2. คำร่าง AddBirthday เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-3 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-4

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
<note align="center">Test case 1 BirthdayBook</note>
<note align="center">Schema No.2</note>
<schemadef purpose="state" style="vert" align="center">
  <nametype>AddBirthdayBook</nametype>]
  <decpart>
    <declaration>
      <inclusion>&delta;</inclusion>
      <name>BirthdayBook</name>
    </inclusion>
    <declname>
      <name>name</name>&input; &col;</declname>
    </declname>
    <application>
      <name>NAME</name>
    </application>
  </declaration>
  <declaration>
    <declname>
      <name>date</name>&input; &col;</declname>
    </declname>
  </declaration>
</schemadef>
</Zedml>
</xml>
```

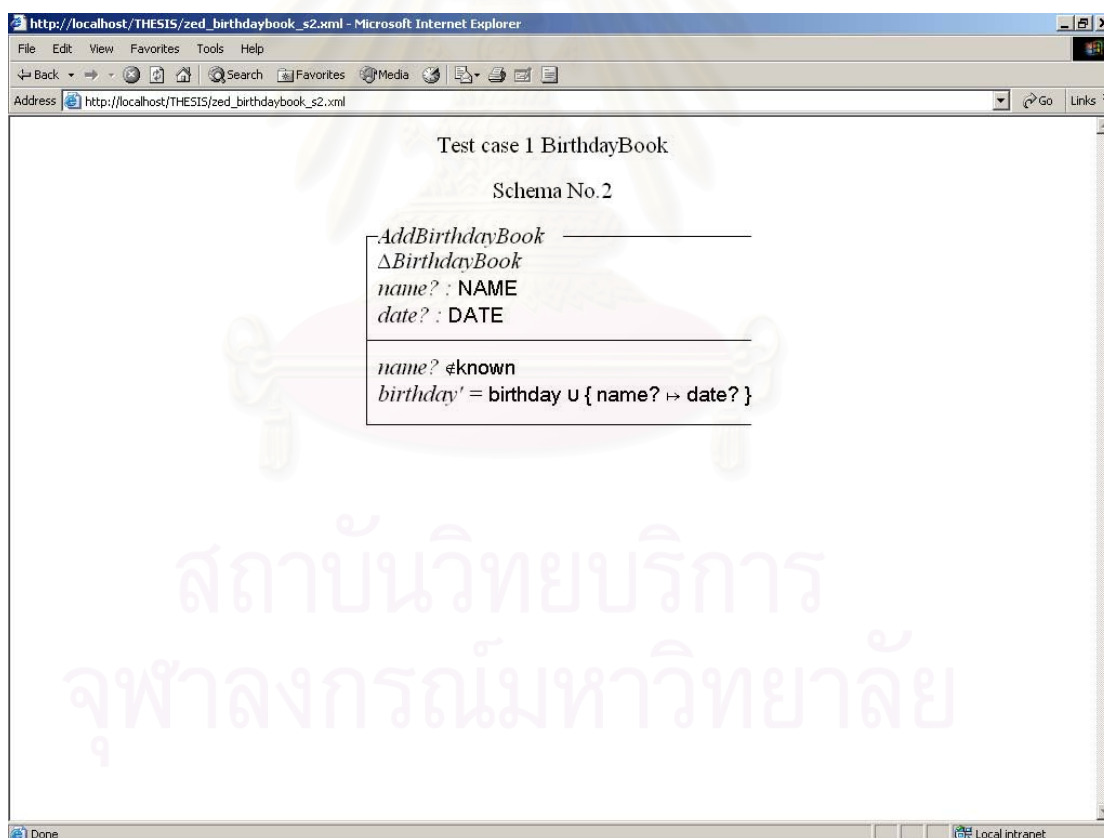
รูปที่ ง-3 รหัสต้นฉบับคำร่าง AddBirthdayBook

```

    <application>
      <name>DATE</name>
    </application>
  </declaration>
</decpart>
<axpart>
  <predicate>
    <declname>
      <name>name</name>&input;</declname>
    <application>&notin;<name>known</name>
    </application>
  </predicate>
  <predicate>
    <declname>
      <name>birthday</name>&afterst; &eq;</declname>
    <application>
      <name>birthday</name> &cup; &lcub; <name>name</name>&input; &map;
    <name>date</name>&input; &rcub;</application>
  </predicate>
</axpart>
</schemadef>
</Zedml>

```

รูปที่ ง-3 รหัสต้นฉบับเค้าร่าง AddBirthdayBook (ต่อ)



รูปที่ ง-4 เค้าร่าง AddBirthdayBook บนเว็บเบราว์เซอร์

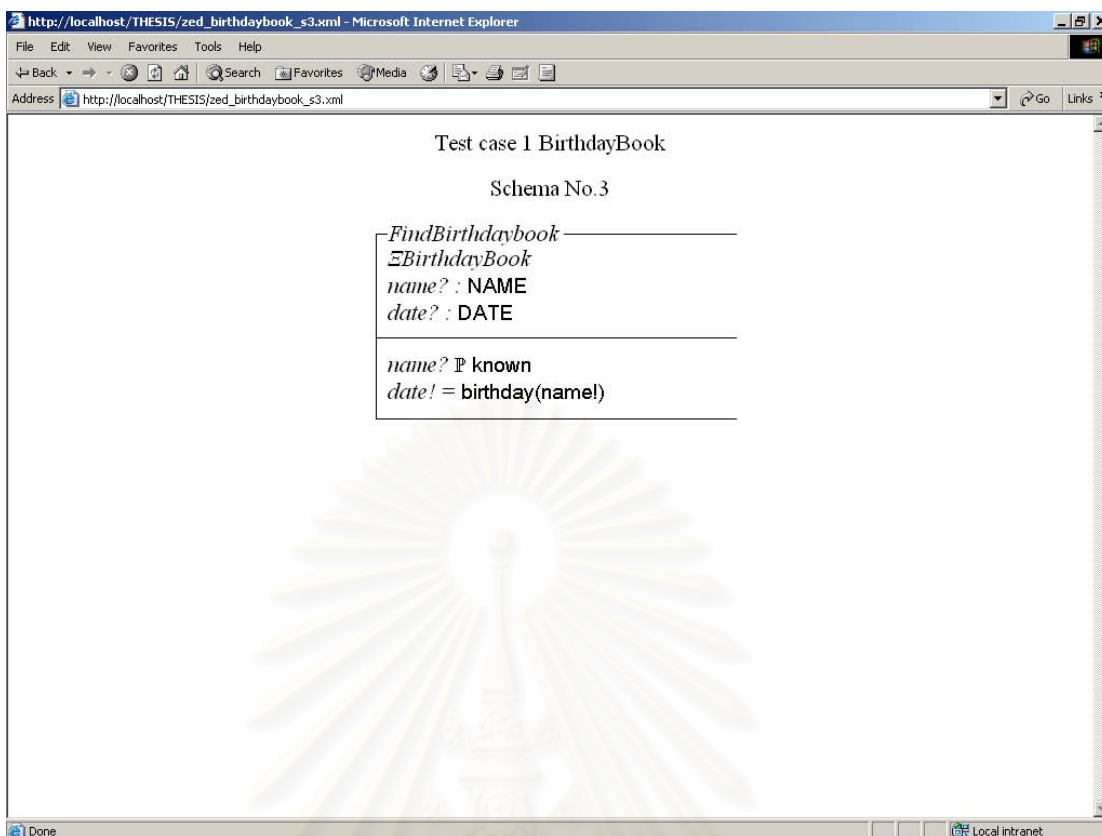
3. คำร่าง FindBirthday เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-5 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-6

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
noNamespaceSchemaLocation="ZEDML.xsd">
<note align="center">Test case 1 BirthdayBook</note>
<note align="center">Schema No.3</note>
<schemadef purpose="state" style="vert" align="center">
  <nametype>FindBirthdaybook</nametype>
  <decpart>
    <declaration>
      <inclusion>&xsi;<name>BirthdayBook</name>
      </inclusion>
      <declname>
        <name>name</name>&input; &col;</declname>
      <application>
        <name>NAME</name>
      </application>
    </declaration>
    <declaration>
      <declname>
        <name>date</name>&input; &col;</declname>
      <application>
        <name>DATE</name>
      </application>
    </declaration>
  </decpart>
  <axpart>
    <predicate>
      <declname>
        <name>name</name>&input;</declname>
      <application>&pset; <name>known</name>
      </application>
    </predicate>
    <predicate>
      <declname>
        <name>date</name>&output; &eq;</declname>
      <application>
        <name>birthday</name>&ltup; <name>name</name>&output;&rtup;</application>
      </predicate>
    </axpart>
  </schemadef>
</Zedml>

```

รูปที่ ง-5 รหัสต้นฉบับคำร่าง FindBirthdayBook

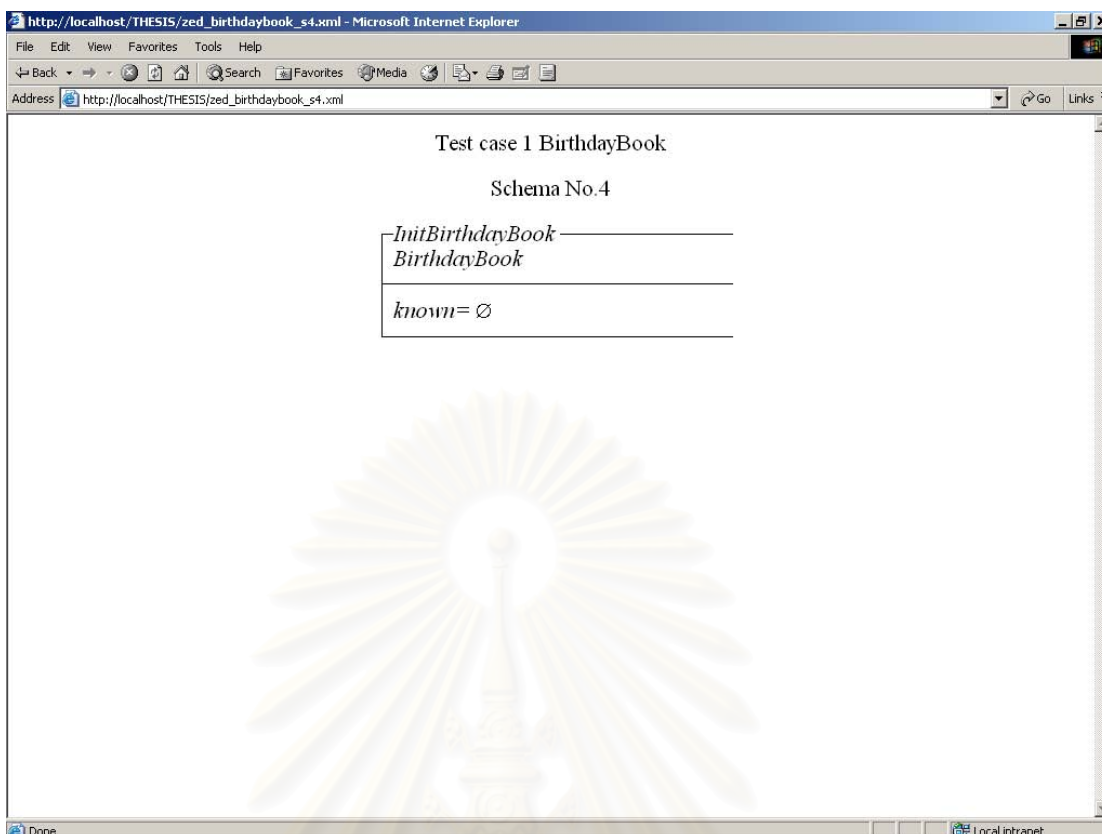


รูปที่ ง-6 คำร่าง FindBirthdayBook บนเว็บเบราว์เซอร์

4. คำร่าง InitBirthday เมื่อนำมาเขียนด้วยเอกซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-7 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-8

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
<note align="center">Test case 1 BirthdayBook</note>
<note align="center">Schema No.4</note>
<schemadef purpose="state" style="vert" align="center">
  <nametype>InitBirthdayBook</nametype>
  <decpart>
    <declaration>
      <declname>
        <name>BirthdayBook</name>
      </declname>
    </declaration>
  </decpart>
  <axpart>
    <predicate>
      <declname>
        <name>known</name>&eq;
      </declname>
      <application>&empty;</application>
    </predicate>
  </axpart>
</schemadef>
</Zedml>
```

รูปที่ ง-7 รหัสต้นฉบับคำร่าง InitBirthdayBook



รูปที่ ง-8 เค้ร่ร่ง InitBirthdayBook บนเว็บเบรเวร์เซออร์

5. เค้ร่ร่ง AddThenFindBirthday เมื่อนำมาเขียนด้วยเอ็ทซ์เอ็มแอลที่นียามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-9 และแสดงผลบนเว็บเบรเวร์เซออร์ได้ดังรูปที่ ง-10

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
<note align="center">Test case 1 BirthdayBook</note>
<note align="center">Schema No.5</note>
<schemadef style="vert" purpose="state" align="center">
  <nametype>AddThenFindBirthday</nametype>
  <decpart>
    <declaration>
      <declname>
        <name>known</name>&col; </declname>
        <application>&pset; <name>NAME</name>
      </application>
    </declaration>
    <declaration>
      <declname>
        <name>birthday</name>&col; </declname>
        <application>
          <name>NAME</name>&pfun; <name>DATE</name>
        </application>
      </declaration>
    </decpart>
  </schemadef>
</Zedml>
```

รูปที่ ง-9 รหัสต้นฉบับเค้ร่ร่ง AddThenFindBirthdayBook

```

<declaration>
  <declname>
    <name>known&afterst;</name> &col;</declname>
    <application>&pset;<name>NAME</name>
  </application>
</declaration>
<declaration>
  <declname>
    <name>birthday&afterst;</name> &col;</declname>
    <application>
      <name>NAME</name> &pfun; <name>DATE</name>
    </application>
</declaration>
<declaration>
  <declname>
    <name>name&input;</name>&col;</declname>
    <application>
      <name>NAME</name>
    </application>
</declaration>
<declaration>
  <declname>
    <name>date &input;</name>&col;</declname>
    <application>
      <name>DATE</name>
    </application>
</declaration>
</decpart>
<xpart>
  <predicate>
    <declname>
      &exist;<name> known&afterst;&afterst; </name>&col;
    </declname>
    <application>&pset; <name>NAME</name>
  </application>
</predicate>
<predicate>
  <declname>
    <name> birthday&afterst;&afterst; </name>&col;
  </declname>
  <application>
    <name>NAME</name>&pfun;<name>DATE</name>&bull;
  </application>
</predicate>
<predicate>
  <declname>
    <name>known</name>&eq;</declname>
    <application>&dom;<name> birthday</name> &and;
  </application>
</predicate>
<predicate>
  <declname>
    <name>known &afterst;&afterst;</name> &eq;</declname>
    <application>&dom;<name> birthday &afterst;&afterst;</name> &and;
  </application>
</predicate>
<predicate>
  <declname>
    <name>name</name> &input;</declname>
    <application>&notin;<name> known</name> &and;
  </application>
</predicate>

```

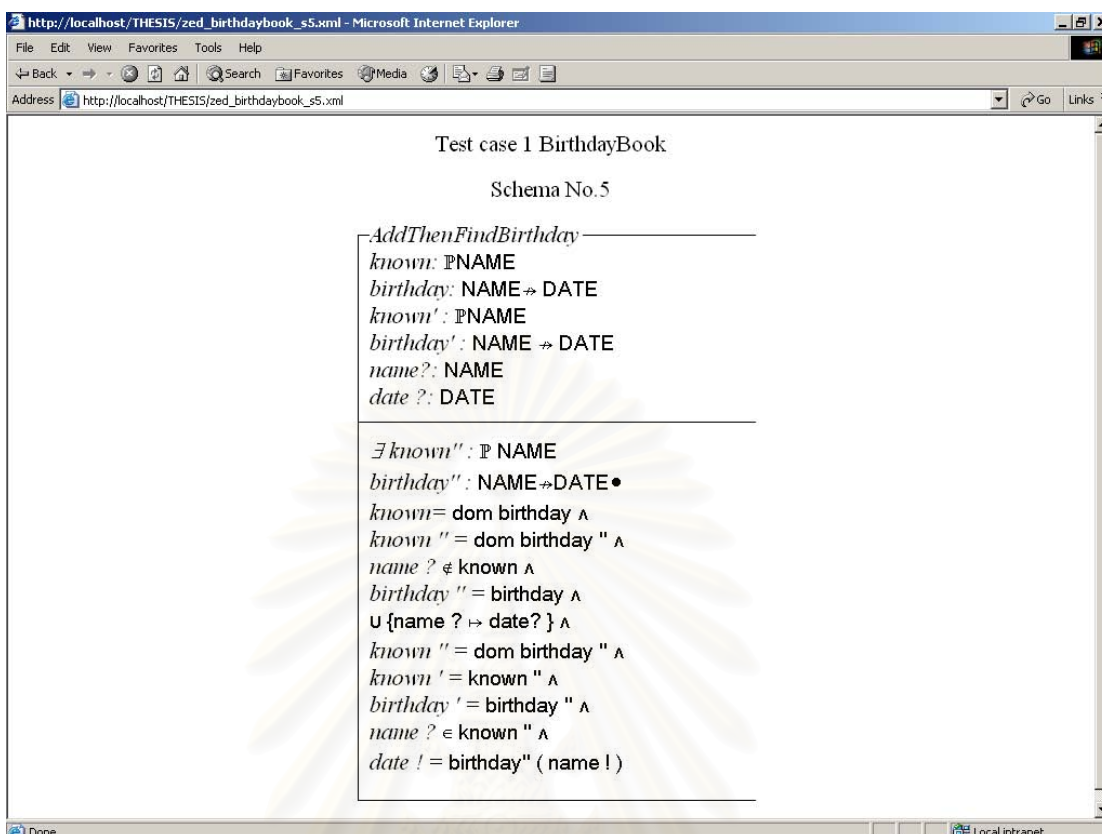
รูปที่ ง-9 รหัสต้นฉบับเค้าร่าง AddThenFindBirthdayBook (ต่อ)

```

<predicate>
  <declname>
    <name>birthday &afterst;&afterst;</name> &eq;</declname>
  <application>
    <name> birthday</name> &and;
  </application>
</predicate>
<predicate>
  <application>
    &cup; &lcub;<name>name</name> &input; &map; <name>date</name>&input;
&rcub;
    &and;
  </application>
</predicate>
<predicate>
  <declname>
    <name>known &afterst;&afterst;</name> &eq;</declname>
  <application>&dom;<name> birthday &afterst;&afterst;</name> &and;
  </application>
</predicate>
<predicate>
  <declname>
    <name>known &afterst;</name> &eq;</declname>
  <application>
    <name>known &afterst;&afterst;</name> &and;
  </application>
</predicate>
<predicate>
  <declname>
    <name>birthday &afterst;</name> &eq;</declname>
  <application>
    <name>birthday &afterst;&afterst;</name> &and;
  </application>
</predicate>
<predicate>
  <declname>
    <name>name &input;</name>
  </declname>
  <application>&isin;<name> known &afterst;&afterst;</name> &and;
  </application>
</predicate>
<predicate>
  <declname>
    <name>date &output;</name> &eq;</declname>
  <application>
    <name>birthday&afterst;&afterst; </name>&ltup; <name>name</name> &output;
    &rtup;
  </application>
</predicate>
</xpart>
</schemadef>
</Zedml>

```

รูปที่ ง-9 รหัสต้นฉบับเค้าร่าง AddThenFindBirthdayBook (ต่อ)



รูปที่ ง-10 เค้กร่าง AddThenFindBirthdayBook บนเว็บเบราว์เซอร์

6. เค้กร่าง BirthdayBook1(Implementing the Birthday Book) เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-11และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-12

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 1 BirthdayBook</note>
  <note align="center">Schema No.6</note>
  <schemadef purpose="state" style="vert" align="center">
    <nametype>BirthdayBook1</nametype>
    <decpart>
      <declaration>
        <declname>
          <name>names</name> &col; </declname>
          <application>&nat1; &fun;</application>
          <name> NAME</name>
        </application>
      </declaration>
```



รูปที่ ง-11 รหัสต้นฉบับเค้าร่าง BirthdayBook1

```

<declaration>
<declname>
  <name>date</name> &col;</declname>
  <application>&nat1; &tfun;
  <name>DATE</name>
</application>
</declaration>
<declaration>
<declname>
  <name>hwm</name> &col;</declname>
  <application>&nat;
  </application>
</declaration>
</decpart>
<xpart>
<predicate>
  <declname>&forall;
  <name> i </name>&com; <name> j </name>1 &upto;<name>hwm</name>&bull;
  </declname>
  <application><name> i </name>&ne;<name> j </name> &RArr;
<name>names(i)</name>
  &ne;
  <name>nemes(j)</name>
  </application>
</predicate>
</xpart>
</schemadef>
</Zedml>

```

รูปที่ ง-11 รหัสต้นฉบับเค้าร่าง BirthdayBook1 (ต่อ)

Test case 1 BirthdayBook

Schema No.6

*BirthdayBook1*

*names* :  $N_1 \rightarrow \text{NAME}$

*date* :  $N_1 \rightarrow \text{DATE}$

*hwm* :  $N$

$\forall i, j 1..hwm \bullet i \neq j \Rightarrow \text{names}(i) \neq \text{nemes}(j)$

## รูปที่ ง-12 เค้าร่าง BirthdayBook1 บนเว็บเบราว์เซอร์

7. เค้าร่าง AddBirthdayBook1(Implementing the Birthday Book) เมื่อนำมาเขียนด้วย  
 เอ็กซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-13 และแสดงผลบนเว็บเบราว์เซอร์ได้  
 ดังรูปที่ ง-14

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 1 BirthdayBook</note>
  <note align="center">Schema No.7</note>
  <schemadef purpose="state" style="vert" align="center">
    <nametype>AddBirthdayBook1</nametype>
    <decpart>
      <declaration>
        <inclusion>&delta; BirthdayBook1</inclusion>
        <declname>
          <name>name</name> &output; &col; </declname>
        <application>
          <name> NAME</name>
        </application>
      </declaration>
      <declaration>
        <declname>
          <name>date</name>&output; &col;</declname>
        <application>
          <name>DATE</name>
        </application>
      </declaration>
    </decpart>
    <axpart>
      <predicate>
        <declname>&forall;
          <name> i </name>&com; <name> j </name>>1
          &upto;<name>hwm</name>&bull;
        </declname>
        <application>
          <name> i </name>&ne; <name> j </name> &rArr;
          <name>names(i)</name> &ne;
          <name>nemes(j)</name>
        </application>
      </predicate>
      <predicate>
        <declname>
          <name>hwm</name>&afterst; &eq;
        </declname>
        <application>
          <name>hwm</name> + 1
        </application>
      </predicate>
      <predicate>
        <declname>
          <name>name</name>&afterst; &eq;
        </declname>

```

รูปที่ ง-13 รหัสต้นฉบับเค้าร่าง AddBirthdayBook1

```

<application>
  <name>names</name> &oplus; &lcub;<name>hwm</name>&afterst;
  &map;<name>name</name>&output; &rcub;
</application>
</predicate>
</predicate>
<declname>
  <name>dates</name>&afterst; &eq;
</declname>
<application>
  <name>dates</name> &oplus; &lcub;<name>hwm</name>&afterst;
  &map;<name>date</name>&output; &rcub;
</application>
</predicate>
</axpart>
</schemadef>
</Zedml>

```

รูปที่ ง-13 รหัสต้นฉบับเค้าร่าง AddBirthdayBook1 (ต่อ)

รูปที่ ง-14 เค้าร่าง BirthdayBook1 บนเว็บเบราว์เซอร์

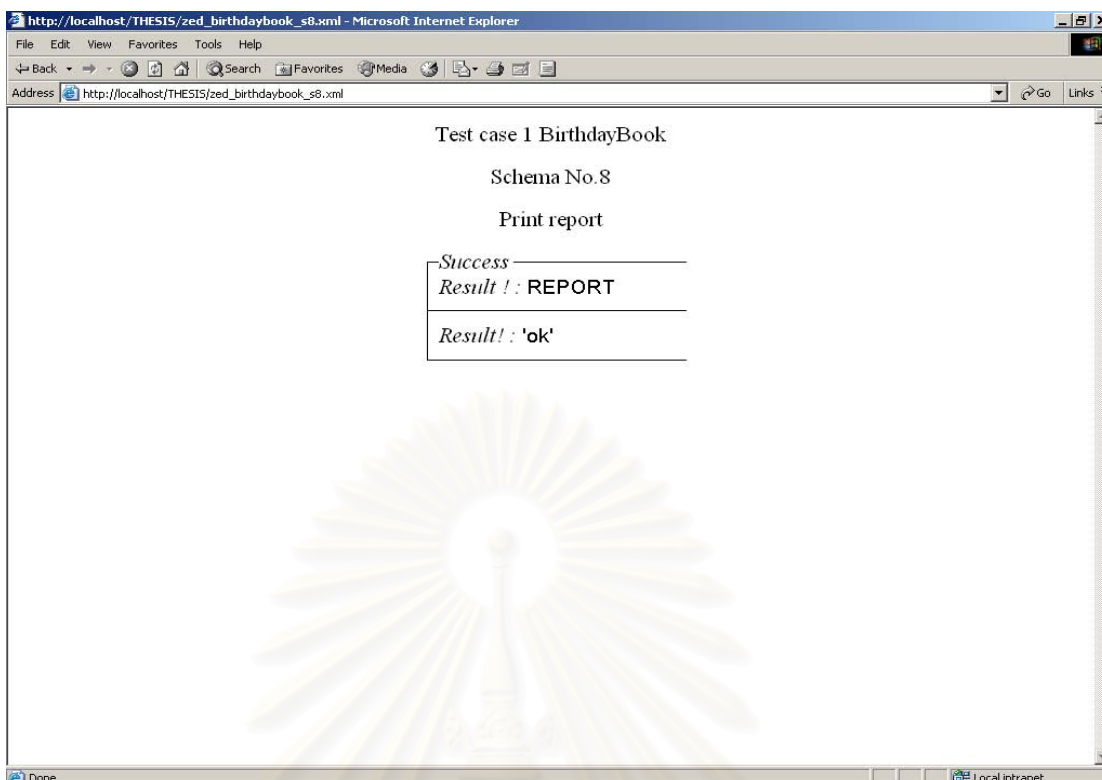
8. เค้าร่าง Success เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่

ง-15 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-16

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 1 BirthdayBook</note>
  <note align="center">Schema No.8</note>
  <note align="center">Print report</note>
  <schemadef purpose="state" style="vert" align="center">
    <nametype>Success</nametype>
    <decpart>
      <declaration>
        <declname>
          <name>Result </name>&output; &col;</declname>
          <application>
            <name>REPORT</name>
          </application>
        </declaration>
      </decpart>
      <axpart>
        <predicate>
          <declname>
            <name>Result</name>&output; &col;</declname>
            <application>
              <name>'ok'</name>
            </application>
          </predicate>
        </axpart>
      </schemadef>
    </Zedml>
```

รูปที่ ง-15 รหัสต้นฉบับเค้าร่าง Success

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ ง-16 คำร่าง Success บนเว็บเบราว์เซอร์

9. คำร่าง Remind เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-17 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-18

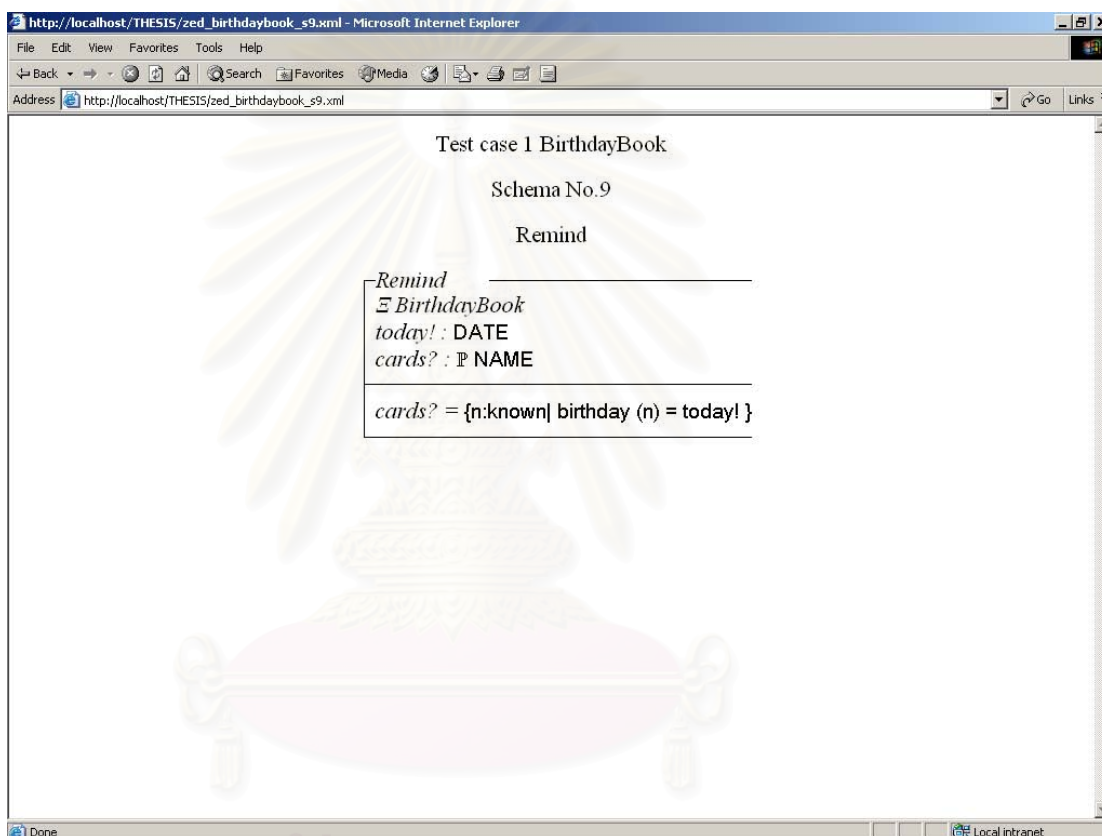
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 1 BirthdayBook</note>
  <note align="center">Schema No.9</note>
  <note align="center">Remind</note>
  <schemadef purpose="state" style="vert" align="center">
    <nametype>Remind</nametype>
    <decpart>
      <declaration>
        <inclusion>&xi; <name>BirthdayBook</name></inclusion>
        <declname>
          <name>today</name>&output; &col;</declname>
        <application>
          <name>DATE</name>
        </application>
      </declaration>
      <declaration>
        <declname>
          <name>cards</name>&input; &col;</declname>
        <application>&pset;
          <name> NAME</name>
        </application>
      </declaration>
    </decpart>
    <axpart>
      <predicate>
        <declname>
```

```

        <name>cards</name>&input; &eq;</declname>
    <application>
        &lcub;<name>n</name>&col;<name>known</name>&verbar;
<name>birthday</name>
        &lt;up;<name>n</name>&rtup; &eq; <name>today</name>&output; &rcub;
    </application>
</predicate>
</xpart>
</schemadef>
</Zedml>

```

รูปที่ ง-17 รหัสต้นฉบับเค้าร่าง Remind



รูปที่ ง-18 เค้าร่าง Remind บนเว็บเบราว์เซอร์

10. เค้าร่าง InitBirthdayBook เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลโดยกำหนดโครงร่างแบบตาม ขวาง ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ง-19 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ง-20

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
    <note align="center">Test case 1 BirthdayBook</note>

```

```

<note align="center">Schema No.10</note>
<schemadef purpose="state" style="horiz" align="center">
  <nametype>InitBirthdayBook</nametype>
  <decpart>
    <declaration>
      <declname>
        <name>BirthdayBook</name>
      </declname>
    </declaration>
  </decpart>

```

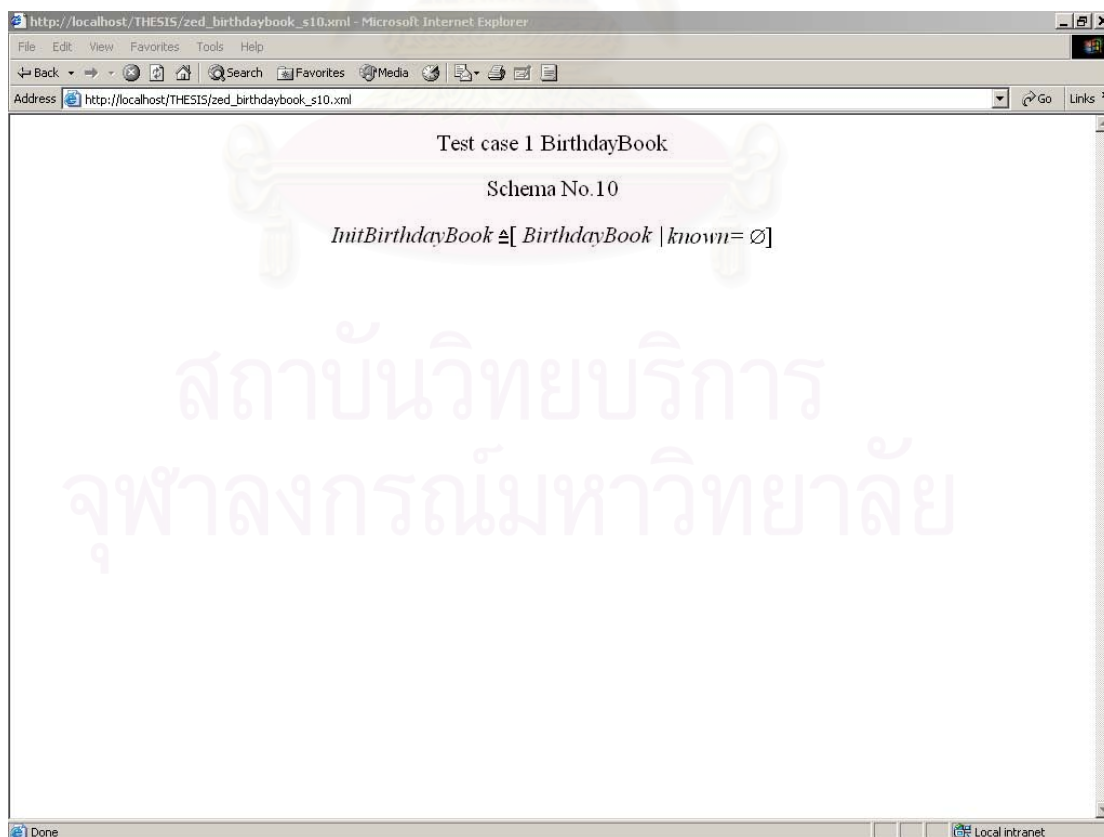
รูปที่ ง-19 รหัสต้นฉบับเค้าร่าง InitBirthdayBook โครงร่างแบบตามขวาง

```

<xpart>
  <predicate>
    <declname>
      <name>known</name>&eq;
    </declname>
    <application>&empty;</application>
  </predicate>
</xpart>
</schemadef>
</Zedml>

```

รูปที่ ง-19 รหัสต้นฉบับเค้าร่าง InitBirthdayBook โครงร่างแบบตามขวาง (ต่อ)



รูปที่ ง-20 เค้าร่าง InitBirthdayBook โครงร่างแบบตามขวางบนเว็บเบราว์เซอร์



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย



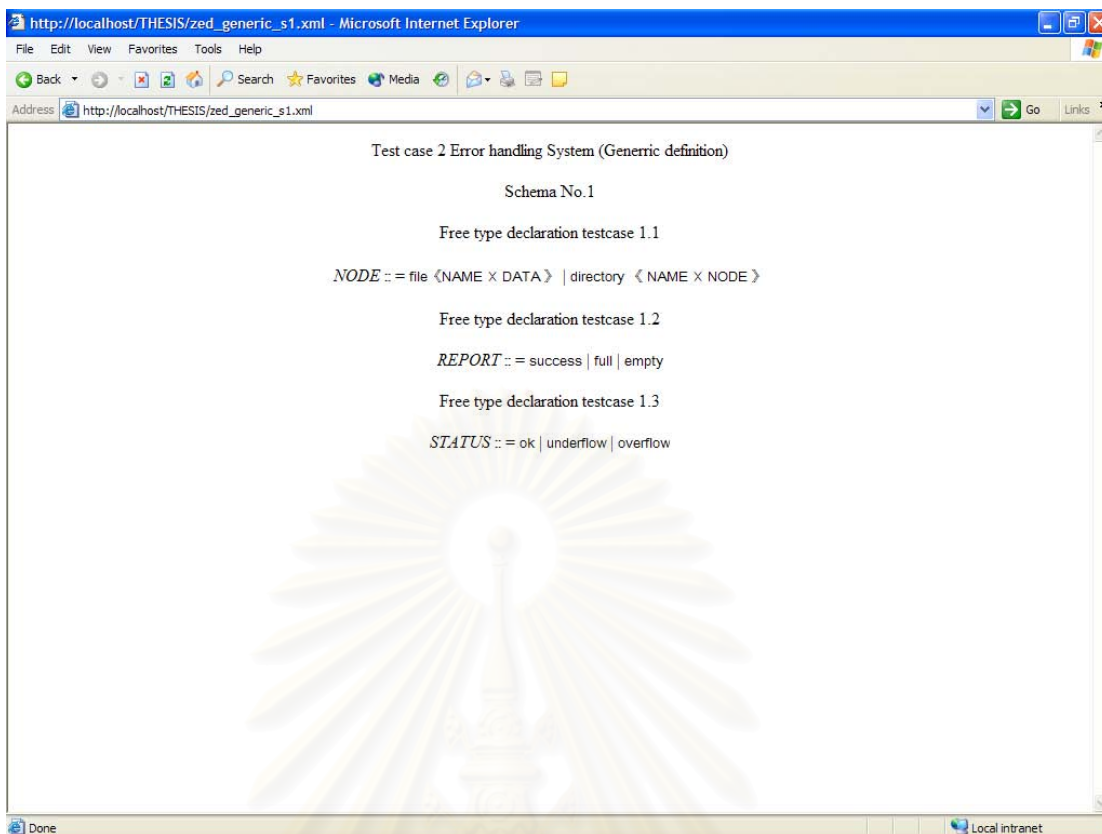
## ภาคผนวก จ. กรณีศึกษาระบบการรายงานข้อผิดพลาด(Error Handling)

ระบบการรายงานข้อผิดพลาดเป็นระบบที่เขียนข้อกำหนดรูปร่างหน้าตาเซต ด้วยโครงสร้างแบบเจนเนอริก(Generic Form) และมีการนิยามข้อมูลชนิดฟรีไทป์ (Free type) การทดสอบการนำเสนอข้อกำหนดรูปร่างด้วยเอ็กซ์เอ็มแอล ในระบบการรายงานข้อผิดพลาดจะทดสอบทั้งหมด 10 เค้าร่าง และจากการทดสอบระบบการนำเสนอ บนเว็บเบราว์เซอร์สามารถแสดงได้ทุกเค้าร่างดังต่อไปนี้

- 1 เค้าร่างการนิยามข้อมูลชนิดฟรีไทป์ เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ จ-1 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ จ-2

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Generic definition)</note>
  <note align="center">Schema No.1</note>
  <note align="center">Free type declaration testcase 1.1 </note>
  <freetypedef align="center">
    <typename>NODE</typename>
    <branch>file&lchev;NAME &prod; DATA &rchev;</branch>
    <branch>directory &lchev; NAME &prod; NODE &rchev;</branch>
  </freetypedef>
  <note align="center">Free type declaration testcase 1.2 </note>
  <freetypedef align="center">
    <typename>REPORT</typename>
    <branch>success</branch>
    <branch>full</branch>
    <branch>empty</branch>
  </freetypedef>
  <note align="center">Free type declaration testcase 1.3 </note>
  <freetypedef align="center">
    <typename>STATUS</typename>
    <branch>ok</branch>
    <branch>underflow</branch>
    <branch>overflow</branch>
  </freetypedef>
</Zedml>
```

รูปที่ จ-1 รหัสต้นฉบับเค้าร่าง การนิยามข้อมูลชนิดฟรีไทป์



รูปที่ จ-2 คำร่างการนิยามข้อมูลชนิดฟรีไทป์บนเว็บเบราว์เซอร์

- 2 คำร่าง BoundedStackMem เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ จ-3 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ จ-4

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd" >
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Generic definition)</note>
  <note align="center">Schema No.2</note>
  <gendef align="center">
    <nametype>BoundedStackMem</nametype>
    <genparam>X</genparam>
    <decpart>
      <declaration>
        <declname>
          <name>stack</name> &col; &seq;
        </declname>
        <application>
          <name>X</name>
        </application>
      </declaration>
    </decpart>
  </gendef>
</Zedml>
```

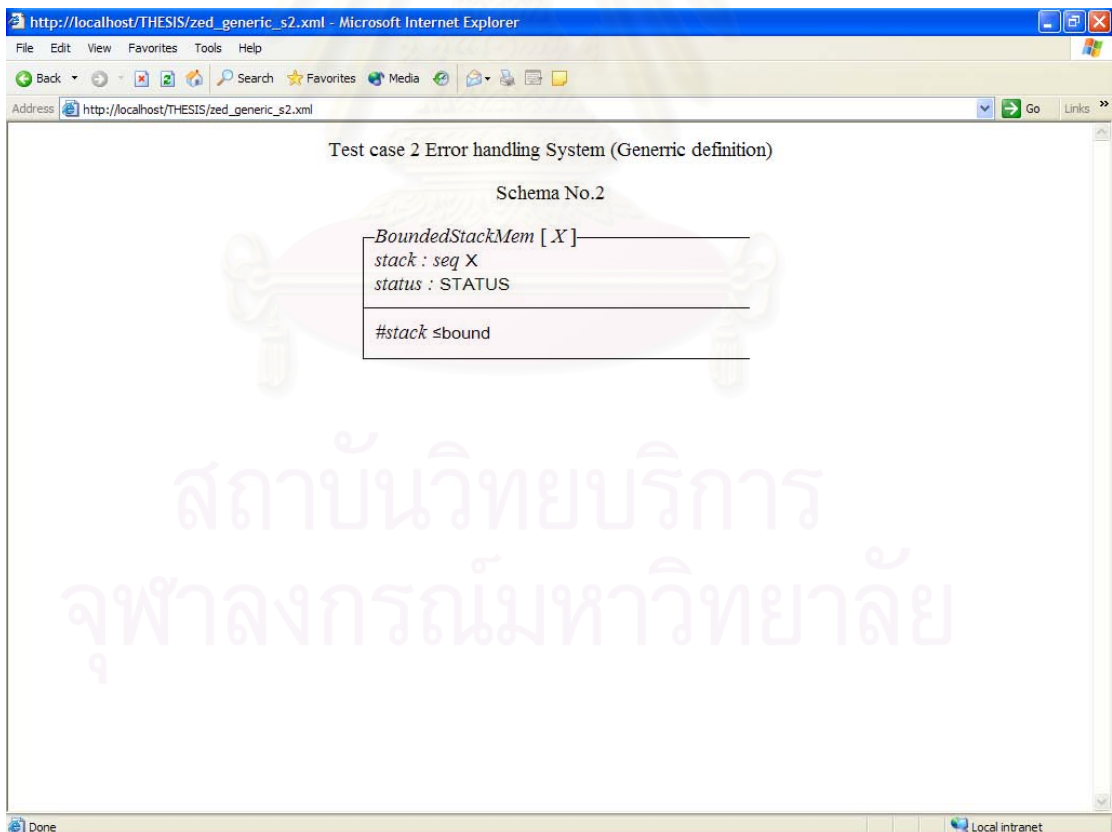
รูปที่ จ-3 รหัสต้นฉบับคำร่าง BoundedStackMem

```

<declaration>
  <declname>
    <name>status</name> &col;
  </declname>
  <application>
    <name>STATUS</name>
  </application>
</declaration>
</decpart>
<axpart>
  <predicate>
    <declname>
      <name>&num;stack</name>
    </declname>
    <application>&le;<name>bound</name>
  </application>
  </predicate>
</axpart>
</gendef>
</Zedml>

```

รูปที่ จ-3 รหัสต้นฉบับเค้าร่าง BoundedStackMem (ต่อ)



รูปที่ จ-4 เค้าร่าง BoundedStackMem บนเว็บเบราว์เซอร์

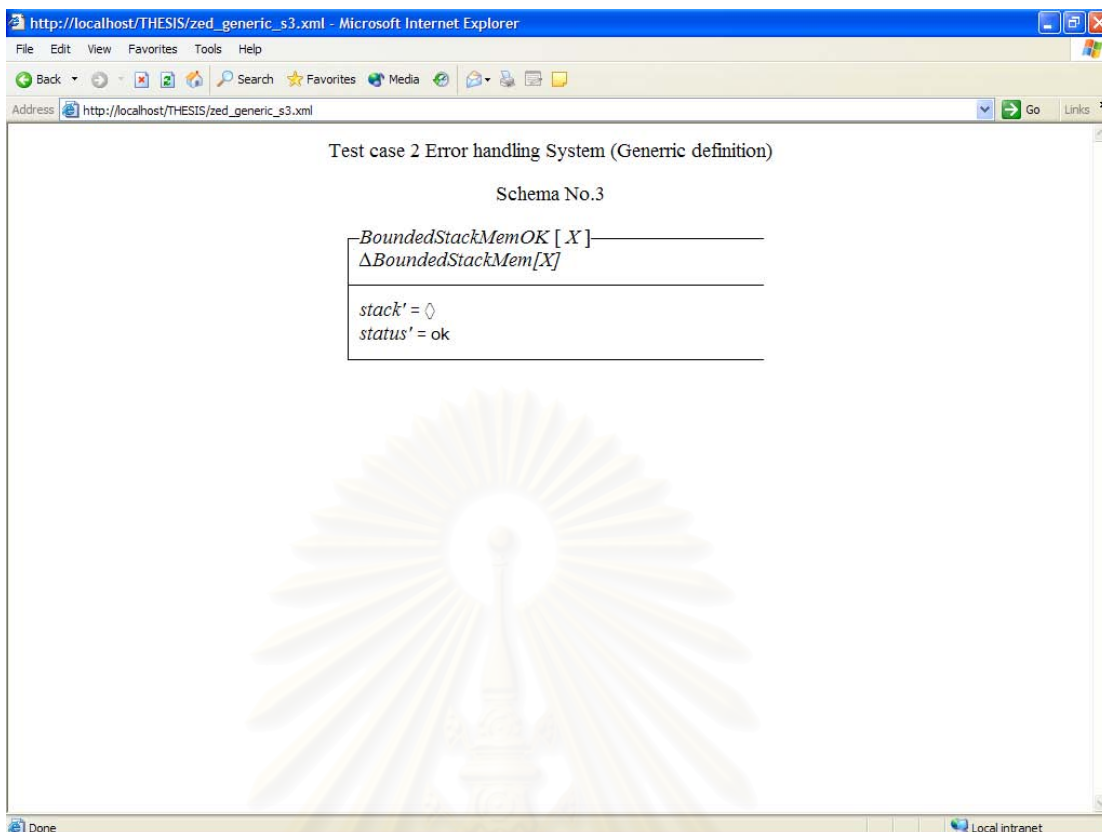
- 3 คำร่าง BoundedStackMemOk เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมี รหัสต้นฉบับดังรูปที่ ๑-5 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-6

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
<note align="center">Test case 2 Error Handling System (Generic definition)</note>
<note align="center">Schema No.3</note>
<gendef align="center">
  <nametype>BoundedStackMemOK</nametype>
  <genparam>X</genparam>
  <decpart>
    <declaration>
      <inclusion>&delta;BoundedStackMem[X]</inclusion>
    </declaration>
  </decpart>
  <axpart>
    <predicate>
      <declname>
        <name>stack</name>&afterst;
      </declname>
      <application>&eq; &emseq;
      </application>
    </predicate>
    <predicate>
      <declname>
        <name>status</name>&afterst;
      </declname>
      <application>&eq;
      <name>ok</name>
      </application>
    </predicate>
  </axpart>
</gendef>
</Zedml>

```

รูปที่ ๑-5 รหัสต้นฉบับคำร่าง BoundedStackMemOk



รูปที่ ๑-6 คำร่าง BoundedStackMemOk บนเว็บเบราว์เซอร์

- 4 คำร่าง PushOk เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-7 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-8

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Generic definition)</note>
  <note align="center">Schema No.4</note>
  <gendef align="center">
    <nametype>PushOK</nametype>
    <genparam>X</genparam>
    <decpart>
      <declaration>
        <inclusion>&delta; BoundedStack &lsqb; X &rsqb;</inclusion>
        <declname>
          <name>x</name>&input; &col;
        </declname>
        <application>
          <name>X</name>
        </application>
      </declaration>
    </decpart>
  </gendef>
</Zedml>
```

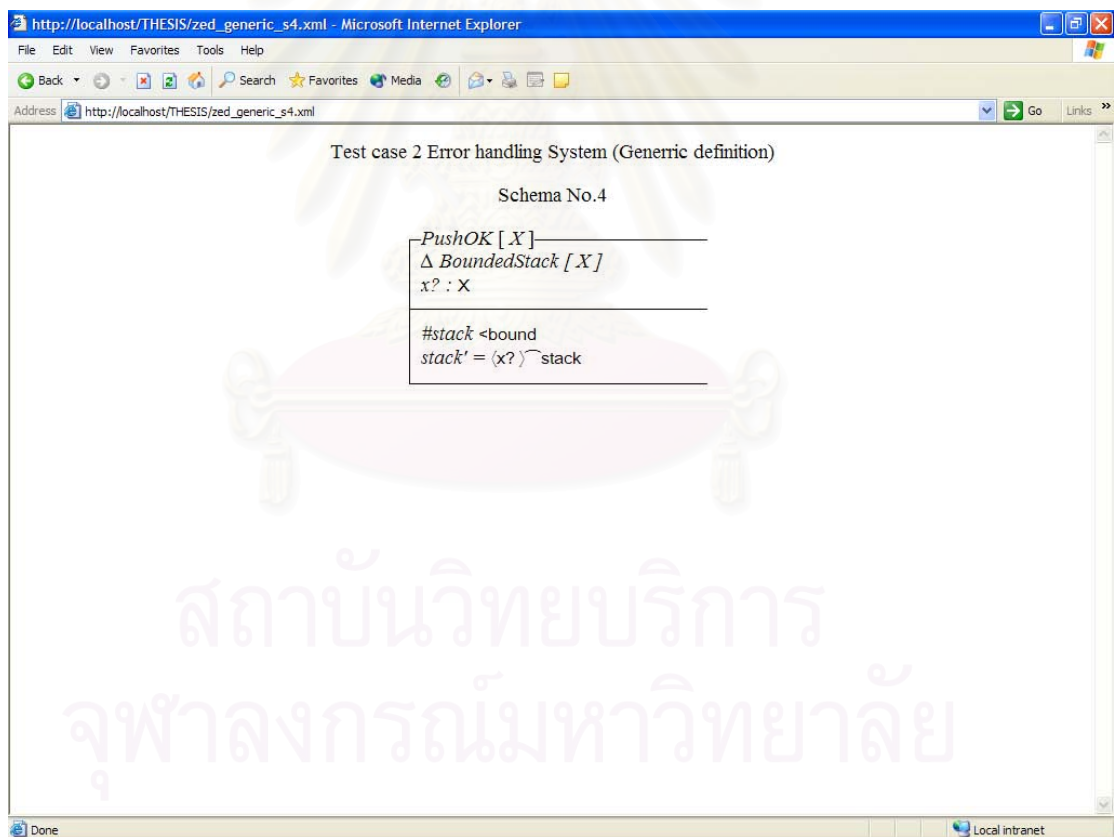
รูปที่ ๑-7 รหัสต้นฉบับคำร่าง PushOk

```

<axpart>
  <predicate>
    <declname>
      <name>&num;stack</name>
    </declname>
    <application>&lt;<name>bound</name>
    </application>
  </predicate>
  <predicate>
    <declname>
      <name>stack</name>&afterst; &eq;</declname>
    <application>
      &lseq;<name>x</name>&input; &rseq;&cat;<name>stack</name>
    </application>
  </predicate>
</axpart>
</gendef>
</Zedml>

```

รูปที่ ๑-7 รหัสต้นฉบับเค้าร่าง PushOk (ต่อ)

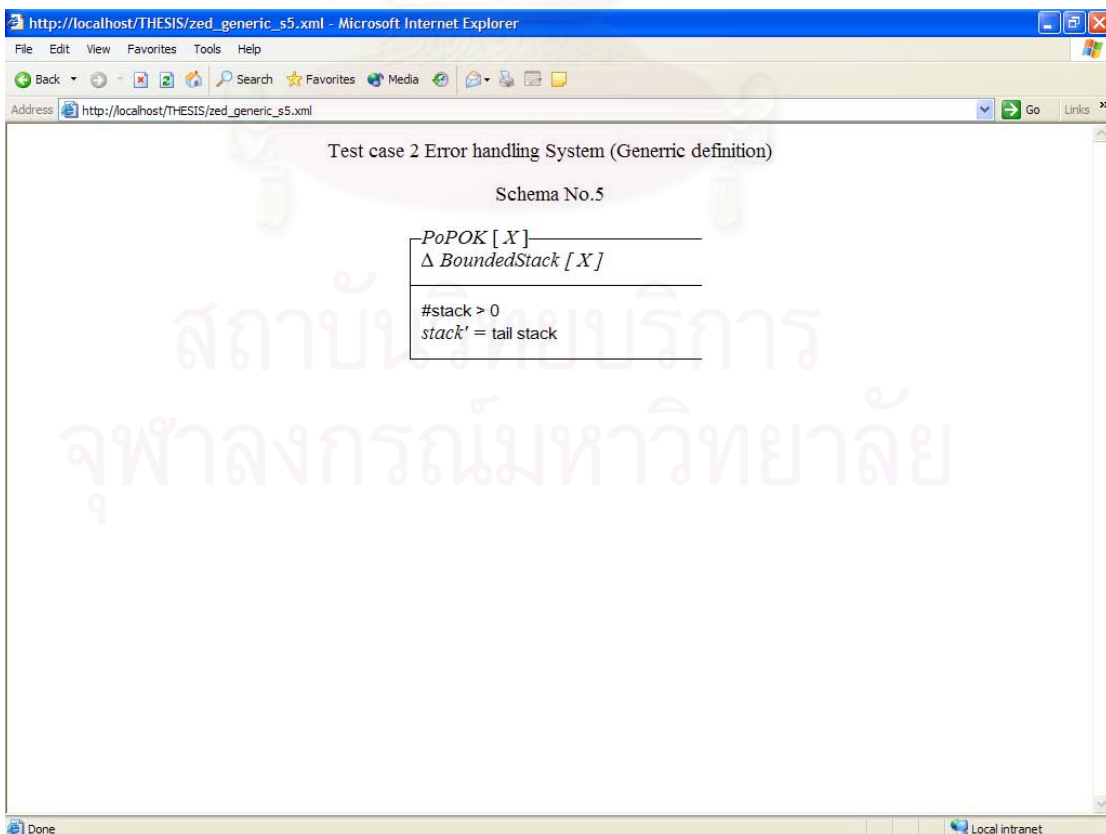


รูปที่ ๑-8 เค้าร่าง PushOk บนเว็บเบราว์เซอร์

- 5 คำร่าง PopOk เมื่อนำมาเขียนด้วยเจ็ทเอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-9 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-10

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
<note align="center">Test case 2 Error Handling System (Generic definition)</note>
<note align="center">Schema No.5</note>
<gendef align="center">
<nametype>PoPOK</nametype>
<genparam>X</genparam>
<decpart>
<declaration>
<inclusion>&delta; BoundedStack &lsqb; X &rsqb;</inclusion>
</declaration>
</decpart>
<xpart>
<predicate>
<application>&num;<name>stack</name> &gt; 0</application>
</predicate>
<predicate>
<declname>
<name>stack&afterst;</name> &eq; </declname>
<application>tail stack</application>
</predicate>
</xpart>
</gendef>
</Zedml>
```

รูปที่ ๑-9 รหัสต้นฉบับคำร่าง PopOk



## รูปที่ จ-10 คำร่าง PopOk บนเว็บเบราว์เซอร์

- 6 คำร่าง TopOk เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ จ-11 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ จ-12

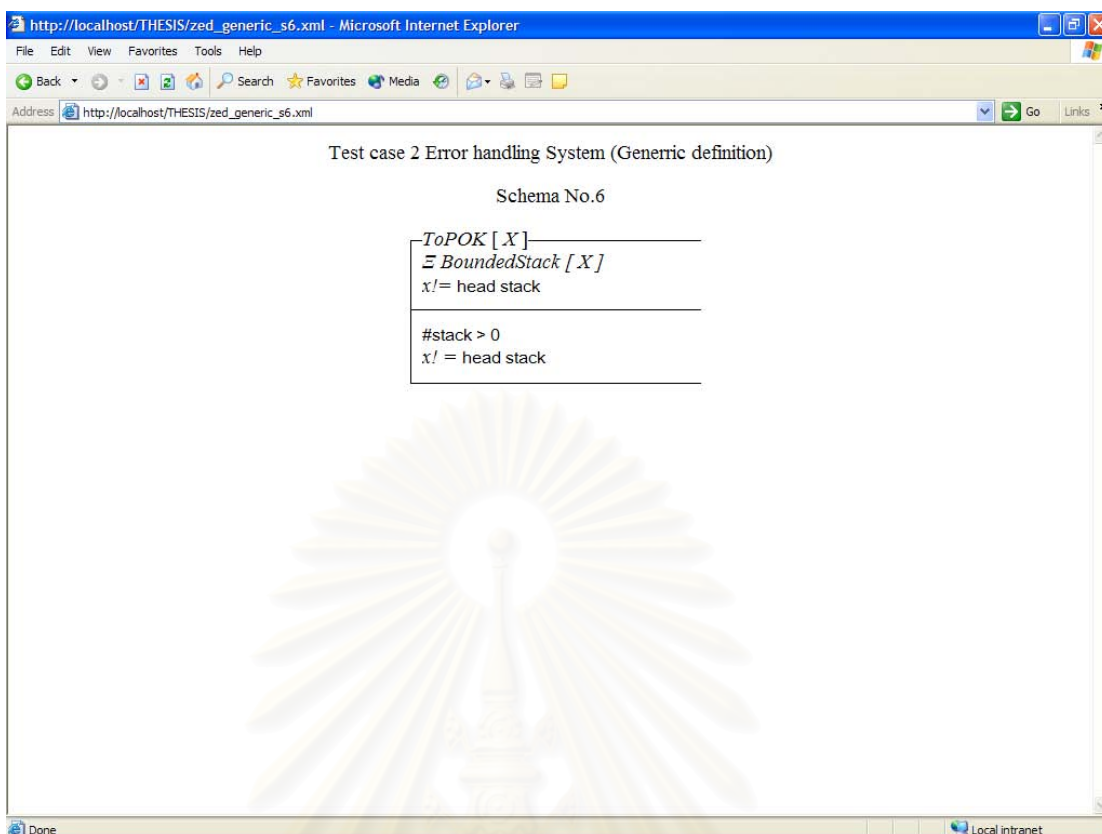
```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Generric definition)</note>
  <note align="center">Schema No.6</note>
  <gendef align="center">
    <nametype>ToPOK</nametype>
    <genparam>X</genparam>
    <decpart>
      <declaration>
        <inclusion>&xi; BoundedStack &lsqb; X &rsqb;</inclusion>
        <declname>
          <name>x&output;</name>&eq;</declname>
        <application>
          <name>head stack</name>
        </application>
      </declaration>
    </decpart>
    <axpart>
      <predicate>
        <application>&num;<name>stack</name> &gt; 0</application>
      </predicate>
      <predicate>
        <declname>
          <name>x&output;</name> &eq;</declname>
        <application>head stack</application>
      </predicate>
    </axpart>
  </gendef>
</Zedml>

```

รูปที่ จ-11 รหัสต้นฉบับคำร่าง TopOk





รูปที่ ๑-12 คำร่าง TopOk บนเว็บเบราว์เซอร์

- 7 คำร่าง Overflow เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-13 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-14

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Generic definition)</note>
  <note align="center">Schema No.7</note>
  <gendef align="center">
    <nametype>Overflow</nametype>
    <genparam>X</genparam>
    <decpart>
      <declaration>
        <inclusion>&xi; BoundedStack &lsqb; X &rsqb;</inclusion>
        <declname>
          <name>r&output;</name>&col;</declname>
        <application>
          <name>REPORT</name>
        </application>
      </declaration>
    </decpart>
  </gendef>
</Zedml>
```

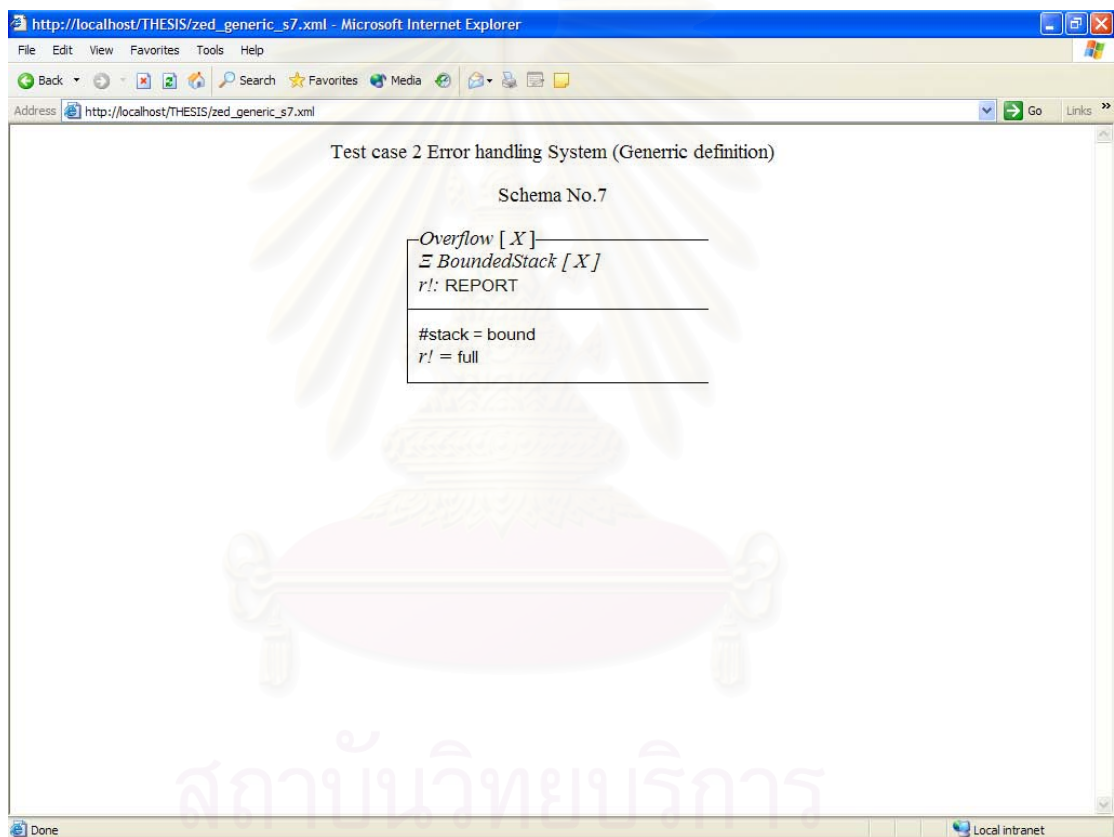
รูปที่ ๑-13 รหัสต้นฉบับคำร่าง Overflow

```

<axpart>
  <predicate>
    <application>&num;<name>stack</name> &eq; <name>bound</name>
  </application>
</predicate>
<predicate>
  <declname>
    <name>r&output;</name> &eq; </declname>
  <application>full</application>
</predicate>
</axpart>
</gendef>
</Zedml>

```

รูปที่ ๑-13 รหัสต้นฉบับเค้าร่าง OverFlow (ต่อ)



รูปที่ ๑-14 เค้าร่าง OverFlow บนเว็บเบราว์เซอร์

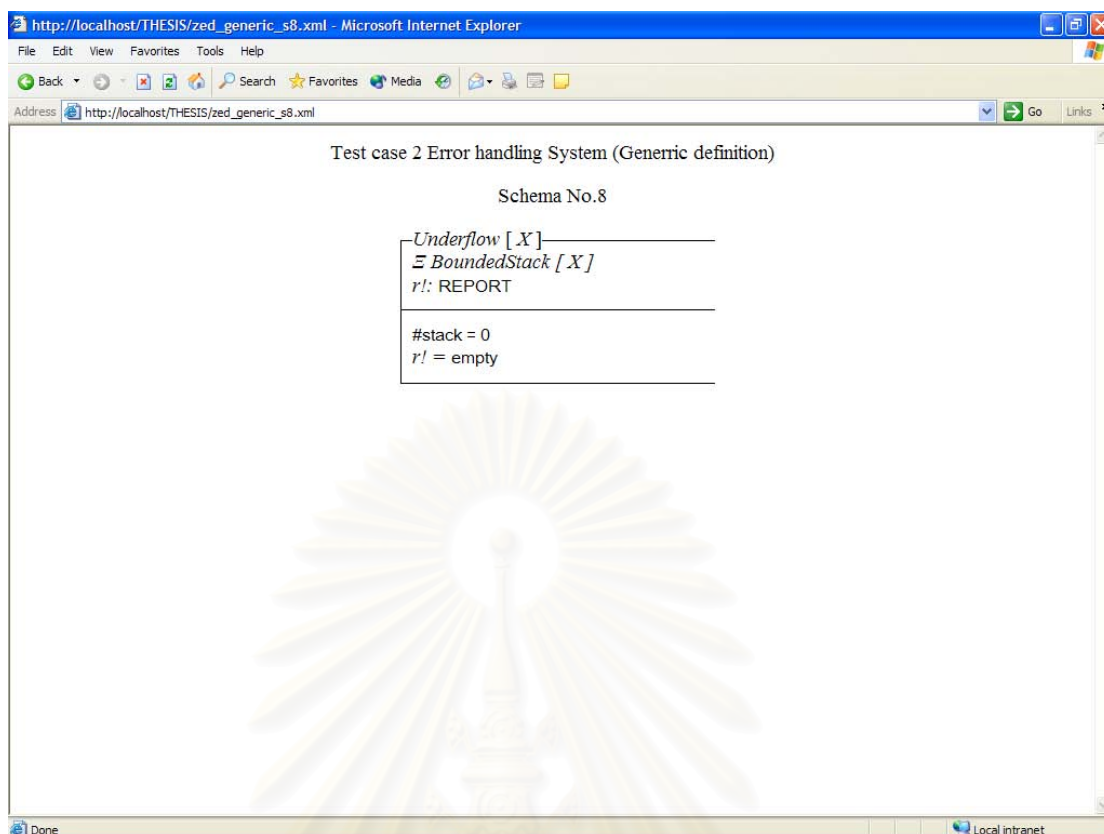
- 8 คำร่าง UnderFlow เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับ ดังรูปที่ จ-15 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ จ-16

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Genereric definition)</note>
  <note align="center">Schema No.8</note>
  <gendif align="center">
    <nametype>Underflow</nametype>
    <genparam>X</genparam>
    <decpart>
      <declaration>
        <inclusion>&xi; BoundedStack &lsqb; X &rsqb;</inclusion>
        <declname>
          <name>r&output;</name> &col;</declname>
        <application>
          <name>REPORT</name>
        </application>
      </declaration>
    </decpart>
    <xpart>
      <predicate>
        <application>&num;<name>stack</name> &eq; <name>0</name>
      </application>
    </predicate>
    <predicate>
      <declname>
        <name>r&output;</name> &eq; </declname>
      <application>empty</application>
    </predicate>
  </xpart>
</gendif>
</Zedml>

```

รูปที่ จ-15 รหัสต้นฉบับคำร่าง UnderFlow



รูปที่ ๑-16 คำร่าง UnderFlow บนเว็บเบราว์เซอร์

- 9 คำร่าง PushMemOk เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-17 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-18

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Generic definition)</note>
  <note align="center">Schema No.9</note>
  <gendef align="center">
    <nametype>PushMemOk</nametype>
    <genparam>X</genparam>
    <decpart>
      <declaration>
        <inclusion>&delta; BoundedStackMemOk &lsqb; X &rsqb;</inclusion>
        <declname>
          <name>x</name>&output; &col;</declname>
        <application>
          <name>X</name>
        </application>
      </declaration>
    </decpart>
  </gendef>
</Zedml>
```

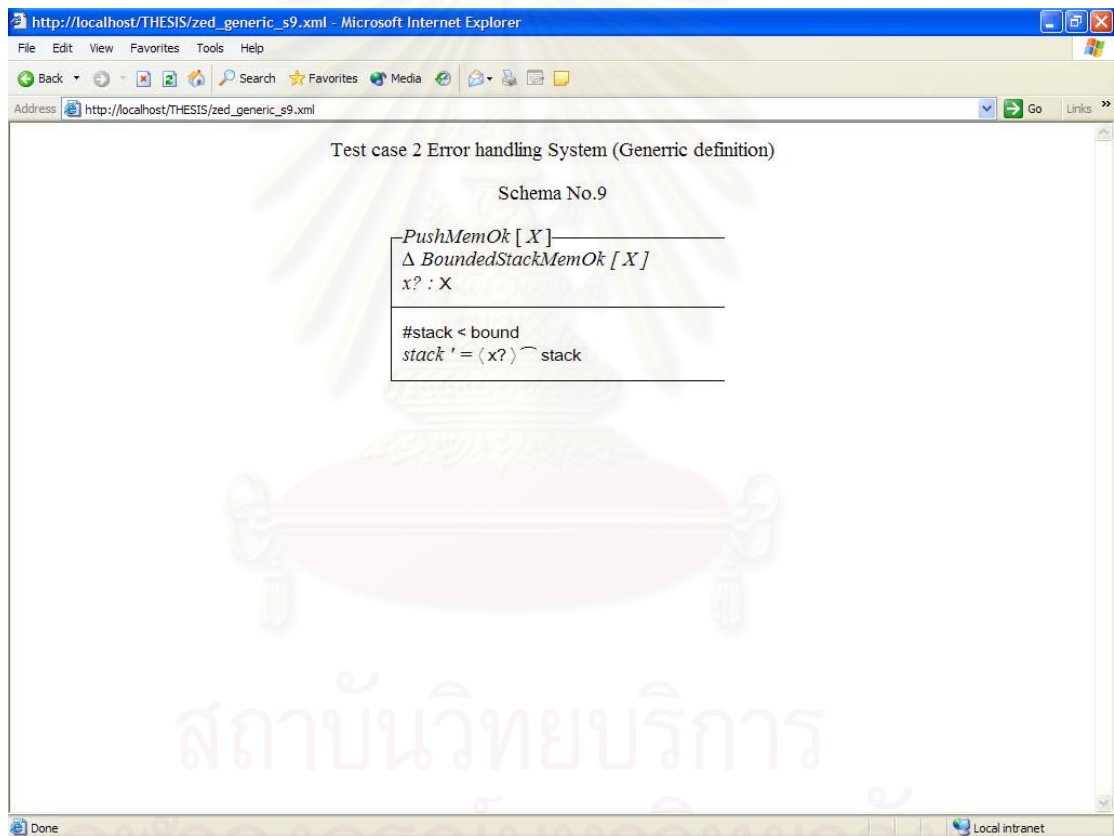
รูปที่ ๑-17 รหัสต้นฉบับคำร่าง PushMemOk

```

<axpart>
  <predicate>
    <application>&num;<name>stack</name> &lt; <name>bound</name>
  </application>
</predicate>
<predicate>
  <declname>
    <name>stack</name> &afterst; &eq;</declname>
  <application>&lseq; <name>x</name>&output; &rseq;
  &cat; <name>stack</name>
  </application>
</predicate>
</axpart>
</gendef>
</Zedml>

```

รูปที่ จ-17 รหัสต้นฉบับเค้าร่าง PushMemOk (ต่อ)



รูปที่ จ-18 เค้าร่าง PushMemOk บนเว็บเบราว์เซอร์

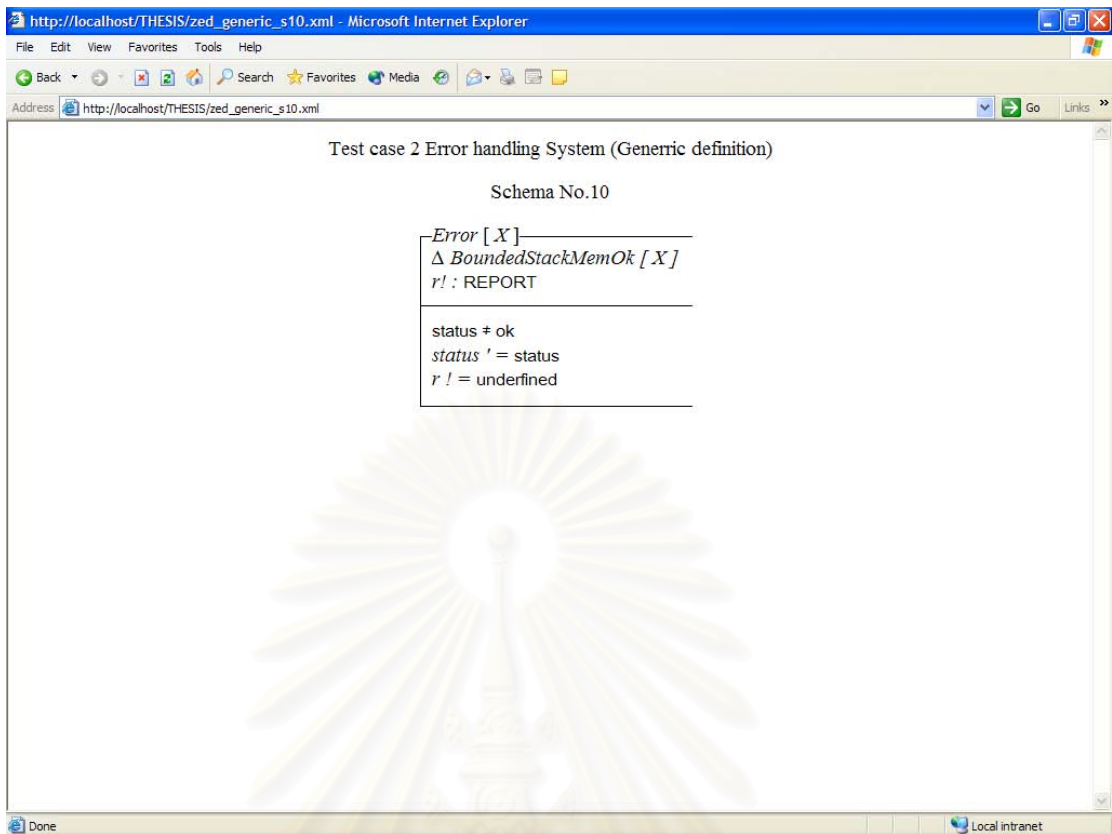
- 10 คำร่าง Error เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ จ-19 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ จ-20

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Zedml SYSTEM "ZEDSYMBOL.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ZEDML.xsd">
  <note align="center">Test case 2 Error Handling System (Generric definition)</note>
  <note align="center">Schema No.10</note>
  <gendef align="center">
    <nametype>Error</nametype>
    <genparam>X</genparam>
    <decpart>
      <declaration>
        <inclusion>&delta; BoundedStackMemOk &lsqb; X &rsqb;</inclusion>
        <declname>
          <name>r</name>&output; &col;</declname>
        <application>
          <name>REPORT</name>
        </application>
      </declaration>
    </decpart>
    <axpart>
      <predicate>
        <application><name>status</name> &ne; <name>ok</name>
        </application>
      </predicate>
      <predicate>
        <declname>
          <name>status</name> &afterst; &eq;</declname>
        <application><name>status</name>
        </application>
      </predicate>
      <predicate>
        <declname>
          <name>r</name> &output; &eq;</declname>
        <application><name>underfined</name>
        </application>
      </predicate>
    </axpart>
  </gendef>
</Zedml>

```

รูปที่ จ-19 รหัสต้นฉบับคำร่าง Error



รูปที่ ๑-20 คำร่าง Error บนเว็บเบราว์เซอร์

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## ภาคผนวก จ. กรณีศึกษาระบบการนับจำนวนคำ(Word Counting)

ระบบการนับจำนวนคำ เป็นระบบที่เขียนข้อกำหนดรูปนัยภาษาเซต ด้วยโครงสร้างแบบสัจพจน์ (Axiom Definition) และมีการนิยามข้อมูลชนิดตัวย่อ (Abbreviation) การทดสอบการนำเสนอข้อกำหนดรูปนัยด้วยเอ็กซ์เอ็มแอล ในระบบการนับจำนวนคำจะทดสอบทั้งหมด 10 เค้าร่าง และจากการทดสอบระบบการนำเสนอ บนเว็บเบราว์เซอร์สามารถแสดงได้ทุกเค้าร่างดังต่อไปนี้

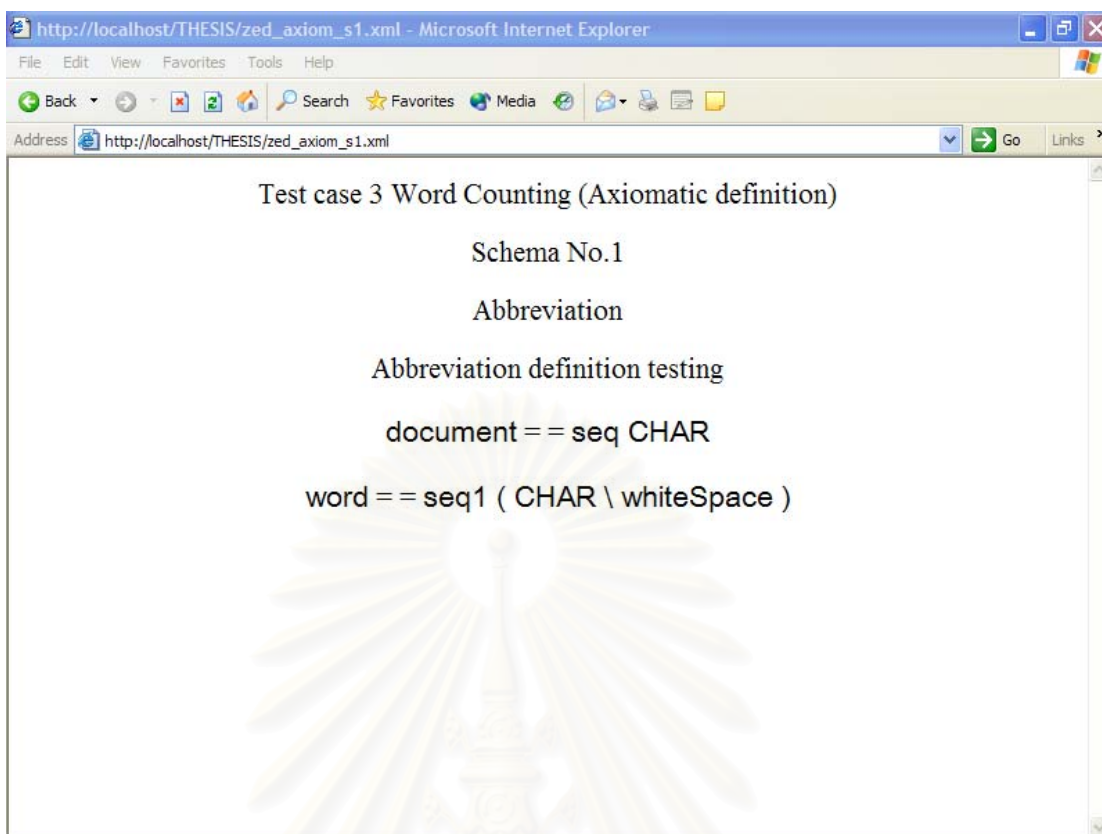
- 1 เค้าร่าง การนิยามข้อมูลชนิดตัวย่อ เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ จ-1 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ จ-2

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 3 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.1</note>
  <note align="center">Abbreviation</note>
  <note align="center">Abbreviation definition testing</note>
  <abbrevdef align="center">
    <name>doument</name>
    <body>seq CHAR</body>
  </abbrevdef>
  <abbrevdef align="center">
    <name>word</name>
    <body>seq1 &ltup; CHAR &sdiff; whiteSpace &rtup;</body>
  </abbrevdef>
</Zedml>
```

รูปที่ จ-1 รหัสต้นฉบับเค้าร่างการนิยามข้อมูลชนิดตัวย่อ

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย



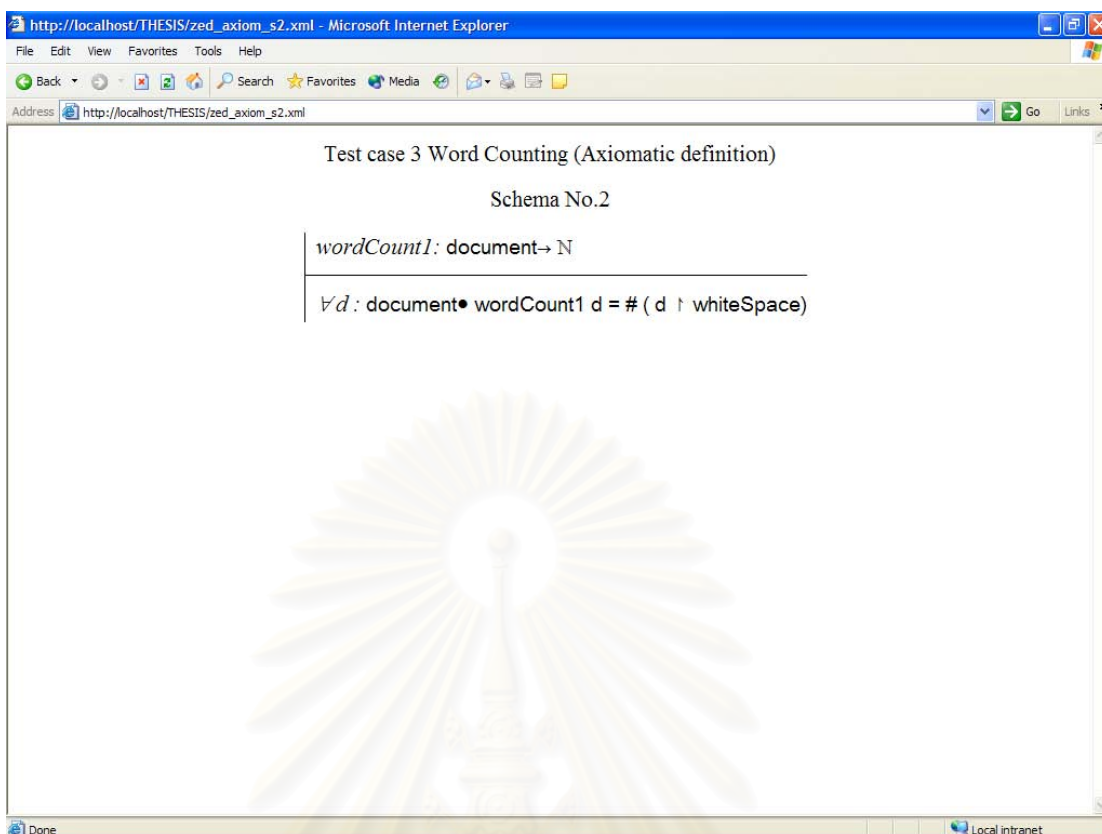


รูปที่ ๑-2 คำร่างการนิยามข้อมูลชนิดตัวอย่อบนเว็บเบราว์เซอร์

- 2 คำร่าง WordCount1 เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-3 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-4

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 3 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.2</note>
  <axdef align="center">
    <decpart><declaration> <declname>
      <name>wordCount1</name>&col;</declname>
      <application><name>document</name>&fun; &nat;</application>
    </declaration></decpart>
    <axpart>
      <predicate><declname>&forall; d &col;</declname>
      <application><name>document</name>&bull; <name>wordCount1</name>
      <name> d</name>&eq; &num; &ltup; <name> d</name>&proj;
      <name>whiteSpace</name>&rtup;
      </application>
    </predicate></axpart></axdef>
</Zedml>
```

รูปที่ ๑-3 รหัสต้นฉบับคำร่าง WordCount1



รูปที่ ๓-4 คำร่าง WordCount1 บนเว็บเบราว์เซอร์

- 3 คำร่าง WordCount2 เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๓-5 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๓-6

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 3 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.3</note>
  <axdef align="center">
    <decpart>
      <declaration>
        <declname>
          <name>wordCount2</name>&col;</declname>
          <application>
            <name>document</name>&fun; &nat;</application>
          </declaration>
        </decpart>
      <axpart>
        <predicate>
          <declname><name>wordCount2 </name>
          </declname>
          <application>&emseq; &eq; 0</application>
        </predicate>
      </axpart>
    </axdef>
  </Zedml>
</xml>
```

รูปที่ ๓-5 รหัสต้นฉบับคำร่าง wordCount2

```

<predicate>
  <declname>&forall; <name>sp</name>&col;
  <name>whiteSpace</name>&comp;
  <name>l</name> &com;
  <name>r</name> &col;
</declname>
</application>
&seq;<name>CHAR</name>&bull;
<name>wordCount2</name>&ltup;
<name>l</name>&cat; &lseq;
<name>sp</name>&rseq; &cat;
<name>r</name>&rtup; &eq;
<name>wordCount2</name>
  <name>l</name>+
  <name>wordCount2</name>
  <name>r</name>
</application>
</predicate>
<predicate>
  <declname>&forall;<name>w</name>&col;</declname>
  <application>
    <name>word</name>&bull;
    <name>wordCount2</name>
    <name>w</name>&eq; 1
  </application>
</predicate>
</axpart>
</axdef></Zedml>

```

รูปที่ ๕-5 รหัสต้นฉบับเค้าร่าง wordCount2 (ต่อ)

Test case 3 Word Counting (Axiomatic definition)

Schema No.3

$wordCount2: document \rightarrow N$
$wordCount2 \ 0 = 0$ $\forall sp: whiteSpace; l, r: seqCHAR \bullet wordCount2(l \wedge (sp) \wedge r) = wordCount2 \ l + wordCount2 \ r$ $\forall w: word \bullet wordCount2 \ w = 1$

## รูปที่ ๑-6 คำร่าง WordCount2 บนเว็บเบราว์เซอร์

- 4 คำร่าง WordSeq เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-7 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-8

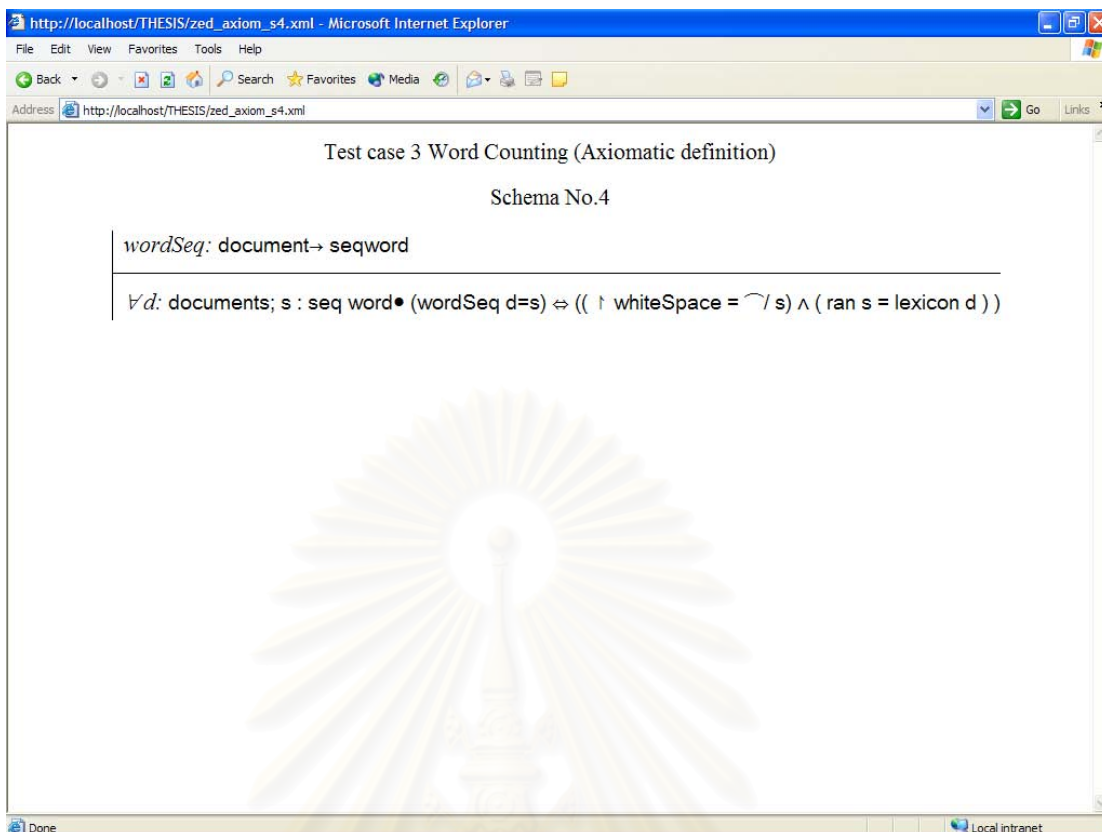
```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 3 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.4</note>
  <axdef align="center">
    <decpart>
      <declaration>
        <declname>
          <name>wordSeq</name>&col;</declname>
          <application>
            <name>document</name>&tfun;
            &seq;<name>word</name></application>
          </declaration>
        </decpart>
        <axpart>
          <predicate>
            <declname>&forall;
              <name>d</name>&col;
            </declname>
            <application><name>documents</name>&comp;
              <name> s </name>&col; &seq; <name>word</name>&bull;

            &ltup;<name>wordSeq</name><name>d</name>&eq;<name>s</name>&rtup;
              &iff;
              &ltup;&ltup; &filter;<name> whiteSpace </name>&eq; &dcat;
              <name>s</name>&rtup; &and; &ltup; &ran; <name>s</name> &eq;
              <name>lexicon</name><name> d</name> &rtup; &rtup;
            </application>
          </predicate>
        </axpart>
      </axdef>
    </Zedml>
  
```

รูปที่ ๑-7 รหัสต้นฉบับคำร่าง WordSeq

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ ๘-8 คำร่าง WordSeq บนเว็บเบราว์เซอร์

- 5 คำร่าง WordFreq เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๘-9 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๘-10

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd" >
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
<note align="center">Test case 3 Word Counting (Axiomatic definition)</note>
<note align="center">Schema No.5</note>
<axdef align="center">
<decpart>
<declaration>
<declname>
<name>wordFreq</name>&col;</declname>
<application>
<name>document</name>&tfun; <name>bag</name>
<name>word</name>
</application>
</declaration>
</decpart>
<axpart>
```

รูปที่ ๘-9 รหัสต้นฉบับคำร่าง wordFreq

```

<predicate>
  <declname>&forall;
  <name>d</name>&col;
</declname>
<application>
  <name>documents</name>&bull;
  <name> wordFreq </name>
  <name>d</name>&eq;
  <name>items</name>&ltup;<name> wordFreq </name>
  <name> d </name>&rtup;
</application>
</predicate>
</axpart>
</axdef>
</Zedml>

```

รูปที่ ๑-9 รหัสต้นฉบับเค้าร่าง wordFreq (ต่อ)

Test case 3 Word Counting (Axiomatic definition)

Schema No.5

$wordFreq: document \rightarrow bag\ word$

$\forall d: documents \bullet wordFreq\ d = items( wordFreq\ d)$

Done Local intranet

รูปที่ ๑-10 เค้าร่าง WordFreq บนเว็บเบราว์เซอร์

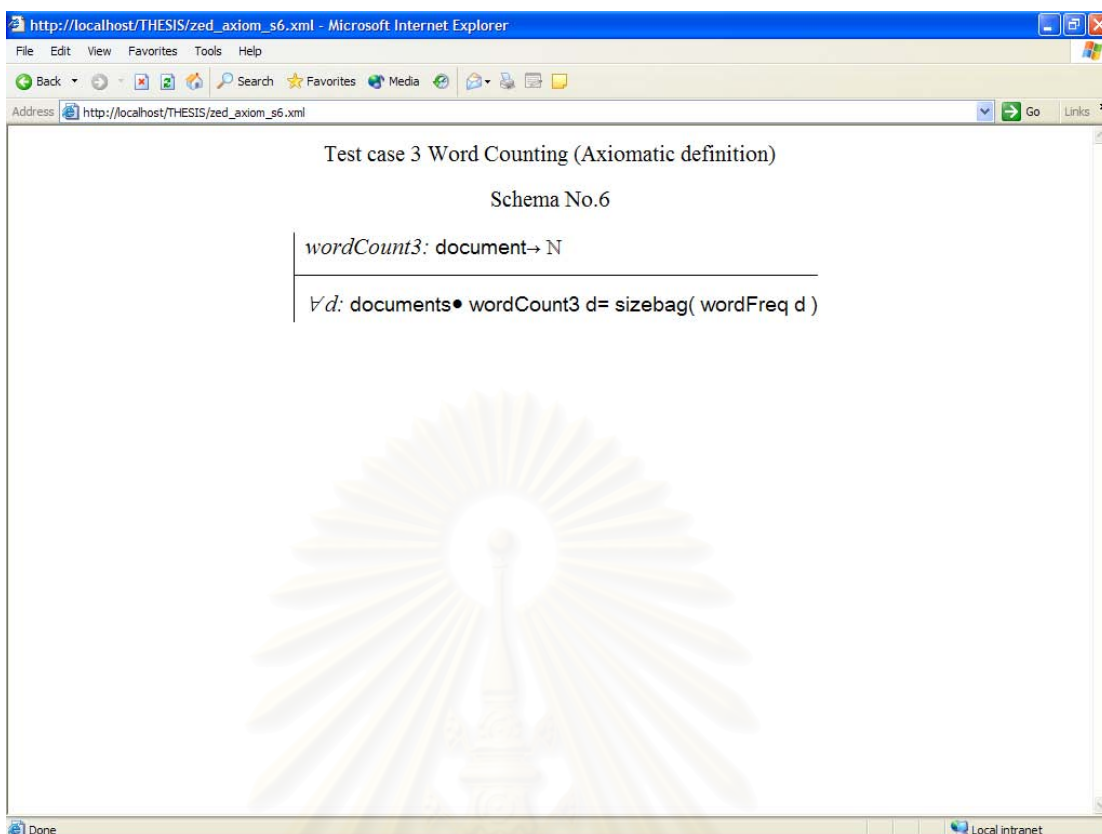
- 6 คำร่าง WordCount3 เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่  
 ฉ-11 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ฉ-12

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 3 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.6</note>
  <axdef align="center">
    <decpart>
      <declaration>
        <declname>
          <name>wordCount3</name>&col;</declname>
          <application>
            <name>document</name>&tfun; <name>&nat;</name>
          </application>
        </declaration>
      </decpart>
      <axpart>
        <predicate>
          <declname>&forall;</name>&col;</declname>
          <application>
            <name>documents</name>&bull;</name>
            <name> wordCount3 </name>
            <name>d</name>&eq;</name>
            <name>sizebag</name>&ltup;</name> <name> wordFreq </name>
            <name> d </name>&rtup;</name>
          </application>
        </predicate>
      </axpart>
    </axdef>
  </Zedml>

```

รูปที่ ฉ-11 รหัสต้นฉบับคำร่าง wordCount3



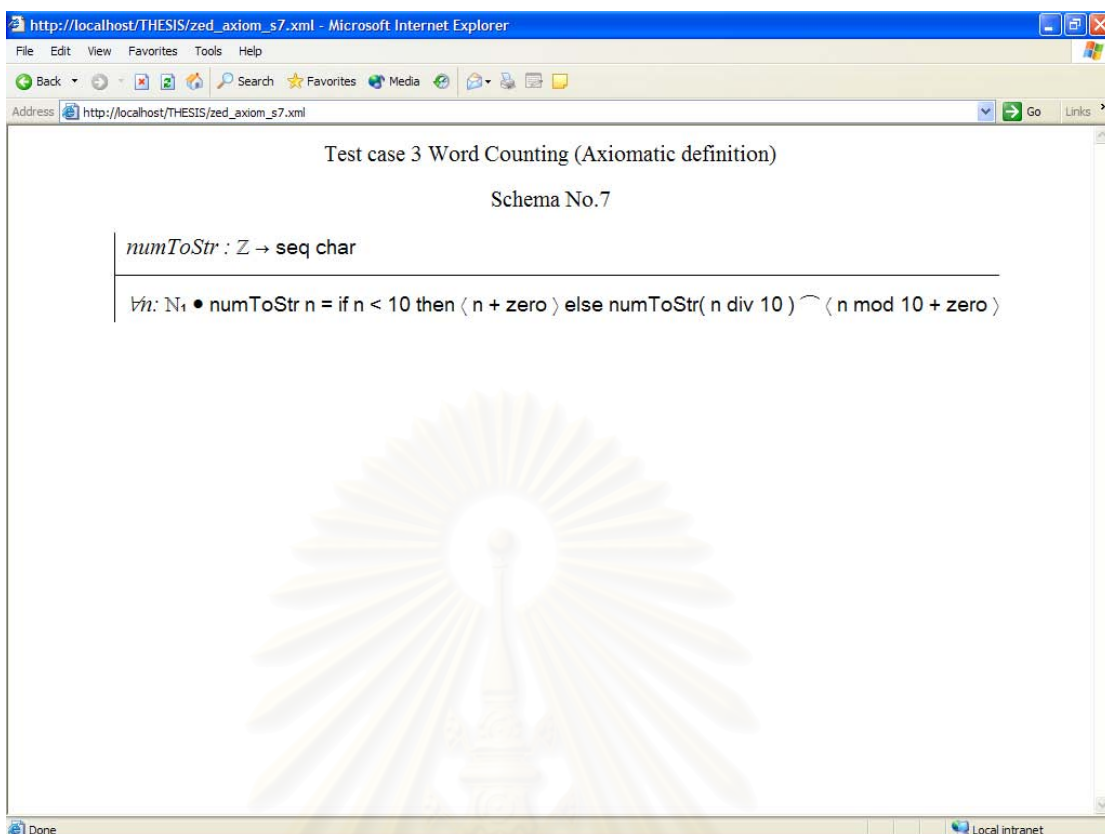
รูปที่ ๑-12 คำร่าง WordCount3 บนเว็บเบราว์เซอร์

- 7 คำร่าง numToStr เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-13 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-14

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 3 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.7</note>
  <axdef align="center">
    <decpart>
      <declaration>
        <declname>
          <name>numToStr </name>&col;</declname>
          <application>&int; &tfun; seq char</application>
        </declaration>
      </decpart>
      <axpart><predicate><declname>&forall;n: </declname>
      <application>&nat1; &bull; <name>numToStr</name> n &eq; if n &lt; 10 then
      &lseq; n + zero &rseq; else <name>numToStr</name>&ltup; n div 10 &rtup; &cat; &lseq; n
      mod 10 + zero &rseq;</application></predicate>
    </axpart></axdef></Zedml>
```

รูปที่ ๑-13 รหัสต้นฉบับคำร่าง numToStr





รูปที่ ๑-14 คำร่าง numToStr บนเว็บเบราว์เซอร์

- 8 คำร่าง uhead เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอล ที่นิยามขึ้นจะมีรหัสต้นฉบับดังรูปที่ ๑-15 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-16

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd" >
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 2 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.8</note>
  <axdef align="center">
    <decpart>
      <declaration>
        <declname>
          <name>uhead</name>&col;</declname>
          <application>&nat; &tfun; seq <name> line </name>&tfun;
        <name>line</name>
        </application>
      </declaration>
    </decpart>
  </axpart>
```

รูปที่ ๑-15 รหัสต้นฉบับคำร่าง uhead

```

<predicate>
  <declname>&forall; <name> l </name>&col; </declname>
  <application>&seq; <name> line </name>&comp; <name>k</name>&col;
    &nat;&bull;
  <name>uhead</name>
    <name> k </name>
    <name> l </name>&eq;
    &ltup; <name> 1</name> &upto; <name>k </name>&rtup; &dres; <name>
      l</name>
  </application>
</predicate>
</axpart>
</axdef>
</Zedml>

```

รูปที่ ๑-15 รหัสต้นฉบับเค้าร่าง uhead (ต่อ)

The screenshot shows a web browser window with the address bar displaying `http://localhost/THESIS/zed_axiom_s8.xml`. The main content area displays the following text:

Test case 3 Word Counting (Axiomatic definition)

Schema No.8

$uhead: N \rightarrow \text{seq line} \rightarrow \text{line}$
$\forall l: \text{seq line}; k: N \bullet uhead\ k\ l = (1..k) \triangleleft l$

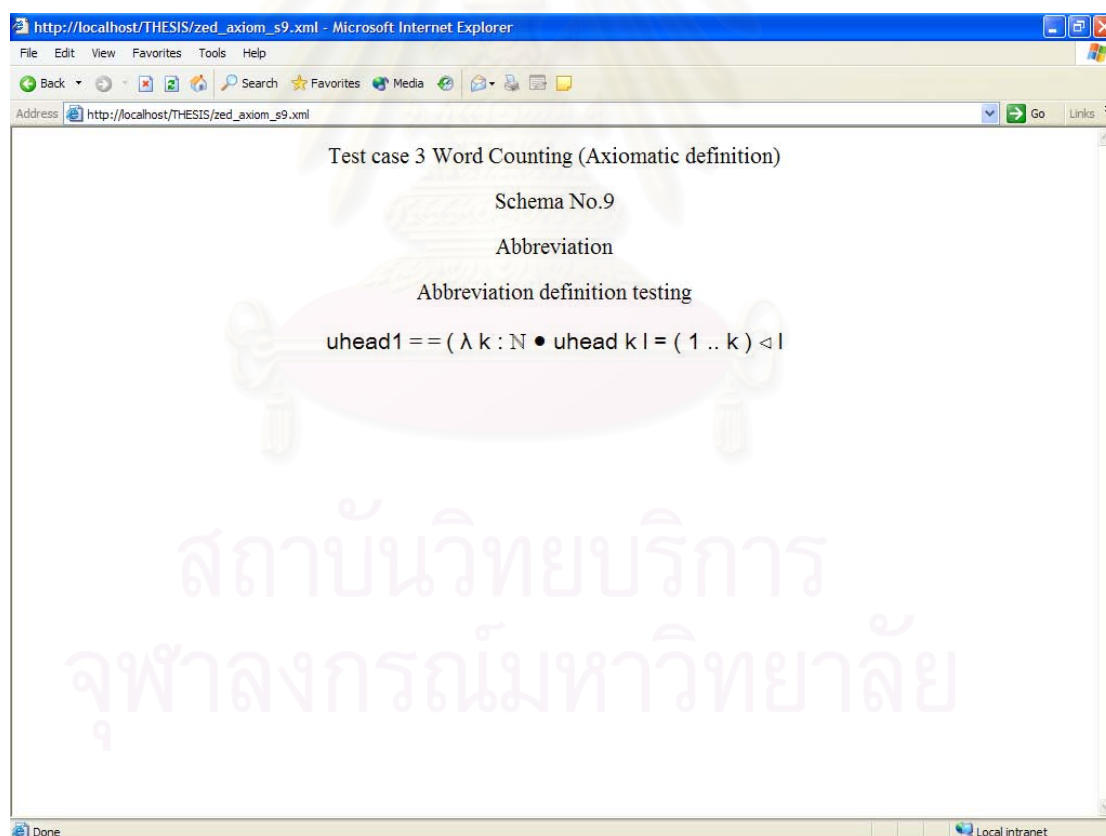
The browser interface includes a menu bar (File, Edit, View, Favorites, Tools, Help), a toolbar with navigation buttons, and a status bar at the bottom showing "Done" and "Local intranet".

รูปที่ ๑-16 เค้าร่าง uhead บนเว็บเบราว์เซอร์

- 9 คำร่าง uhead1 (lambda-expression) เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้น จะมีรหัสต้นฉบับดังรูปที่ ๑-17 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ๑-18

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 2 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.9</note>
  <note align="center">Abbreviation</note>
  <note align="center">Abbreviation definition testing</note>
  <abbrevdef align="center">
    <name>uhead1</name>
    <body>&lt;math>\lambda k : \mathbb{N} \bullet \text{uhead } k \mid = (1 \dots k) \mid </math>
&dres; |</body>
  </abbrevdef>
</Zedml>
```

รูปที่ ๑-17 รหัสต้นฉบับคำร่าง uhead1 (lambda-expression)



รูปที่ ๑-18 คำร่าง uhead1 (lambda-expression) บนเว็บเบราว์เซอร์

- 10 คำร่าง tr\_c เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้น จะมีรหัสต้นฉบับดังรูปที่ ฉ-19 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ฉ-20

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Test case 2 Word Counting (Axiomatic definition)</note>
  <note align="center">Schema No.10</note>
  <axdef align="center">
    <decpart>
      <declaration>
        <declname>
          <name>tr_c</name>&col;</declname>
          <application>&ltup; &pset; <name>char</name>&prod;
<name>char</name>&rtup;
          </application>
        </declaration>
      </decpart>
      <xpart>
        <predicate>
          <declname>&forall;<name> patt </name>&col; </declname>
          <application>&pset; <name> char </name>&comp;
          <name>c</name>&col;<name>char</name>&comp;
          <name>s</name>&col;<name>stream</name>&comp;
          <name>i</name>&col; &nat; &verbar;<name> i </name> &isin;
          <name> 1 </name>&upto; &num;<name>s</name>&bull;
          </application>
        </predicate>
        <predicate>
          <declname>
            <name>tr_c</name>
            &ltup;<name> patt</name>&com;<name>c</name>&rtup;
            <name> s </name><name> i </name> &eq;
          </declname>
          <application>
            if <name> s </name><name> i </name>&notin;
            <name> patt </name>then<name> c </name>
            else
            <name> s </name><name> i </name>
          </application>
        </predicate>
      </xpart>
    </axdef>
  </Zedml>

```

รูปที่ ฉ-19 รหัสต้นฉบับคำร่าง tr\_c

http://localhost/THESIS/zed\_axiom\_s10.xml - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print

Address http://localhost/THESIS/zed\_axiom\_s10.xml Go Links

Test case 3 Word Counting (Axiomatic definition)

Schema No.10

$$tr\_c : (\mathbb{P} \text{ char} \times \text{char})$$


---


$$\forall patt : \mathbb{P} \text{ char} ; c : \text{char} ; s : \text{stream} ; i : \mathbb{N} \mid i \in 1 \dots \#s \bullet$$

$$tr\_c (patt, c) s i = \text{if } s i \notin patt \text{ then } c \text{ else } s i$$

Done Local intranet

รูปที่ ๑-20 คำร่าง tr\_c บนเว็บเบราว์เซอร์

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## ภาคผนวก ข. กรณีศึกษาเพิ่มเติมระบบร้านเช่าวิดีโอ(Video Shop Hiring System)

ระบบร้านเช่าวิดีโอเป็นระบบที่เขียนข้อกำหนดรูปร่างภาษาเซต ซึ่งมีการใช้โครงสร้างการประกาศตัวแปร(Declaration) ที่ใช้ในระบบ การทดสอบการนำเสนอข้อกำหนดรูปร่างด้วยเอ็กซ์เอ็มแอล ในระบบ ร้านเช่าวิดีโอจะทดสอบทั้งหมด 2 เค้าร่าง และผลการทดสอบด้วยระบบการนำเสนอ บนเว็บเบราว์เซอร์สามารถแสดงผลได้อย่างถูกต้องดังต่อไปนี้

- 1 เค้าร่าง การประกาศตัวแปรในระบบ Video Shop Hiring ซึ่งมีอยู่ 2 ตัวแปรคือ ตัวแปร User กับ ตัวแปร Video เมื่อนำมาเขียนด้วยเอ็กซ์เอ็มแอลที่นิยามขึ้น จะมีรหัสต้นฉบับดังรูปที่ ข-1 และแสดงผลบนเว็บเบราว์เซอร์ได้ดังรูปที่ ข-2

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Rathakate (Hanchana) -->
<?xml-stylesheet type="text/xsl" href="zedmlxslt.xslt"?>
<!DOCTYPE Zedml SYSTEM "zedsymbol.dtd">
<Zedml xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="zedml.xsd">
  <note align="center">Declaration variables in Video shop system(Source: VIDEO hiring
system)</note>
  <declarations align="center">
    <nametype>Video</nametype>
    <declaration>
      <declname>
        <name>title</name>&col;
      </declname>
      <application>
        <name>TITLE</name>
      </application>
    </declaration>
    <declaration>
      <declname>
        <name>subject</name>&col;
      </declname>
      <application>
        <name>SUBJECT</name>
      </application>
    </declaration>
    <declaration>
      <declname>
        <name>cert</name>&col;
      </declname>
      <application>
        <name>CERT</name>
      </application>
    </declaration>
  </declarations>
```

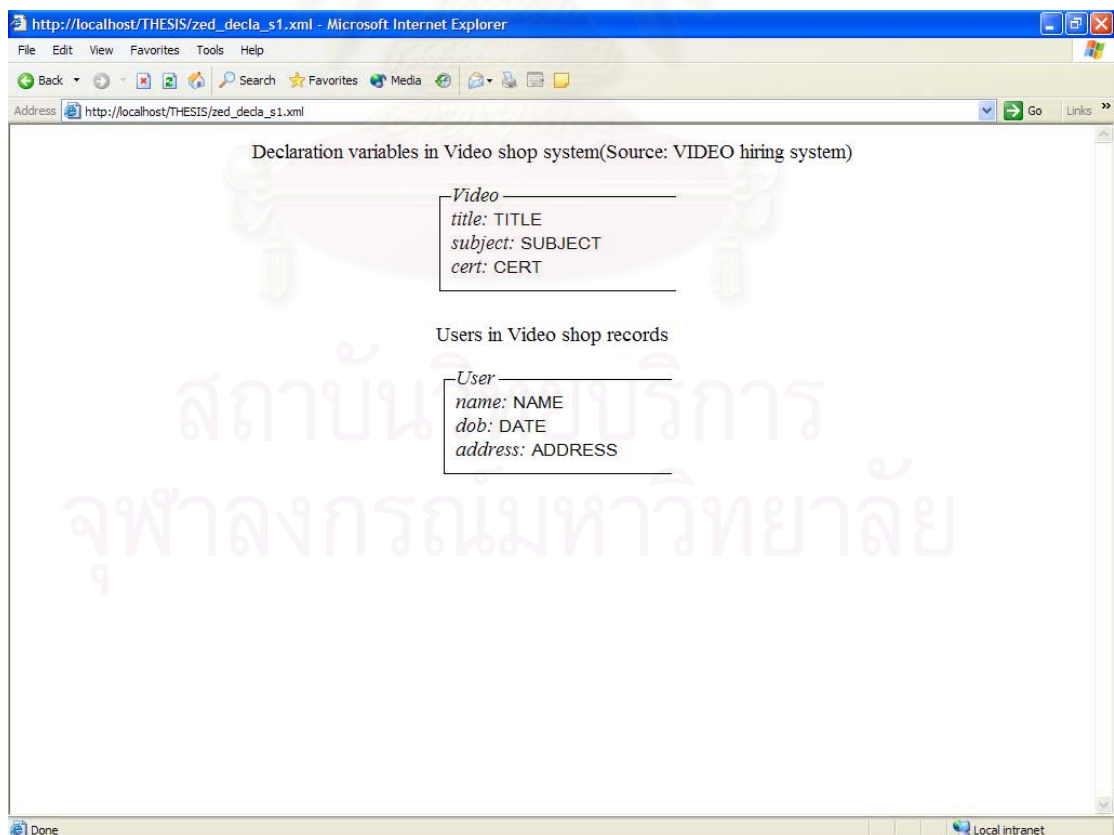
รูปที่ ข-1 รหัสต้นฉบับเค้าร่างการประกาศตัวแปร

```

<note align="center">Users in Video shop records</note>
<declarations align="center">
  <nametype>User</nametype>
  <declaration>
    <declname>
      <name>name</name>&col;</declname>
    <application>
      <name>NAME</name>
    </application>
  </declaration>
  <declaration>
    <declname>
      <name>dob</name>&col;</declname>
    <application>
      <name>DATE</name>
    </application>
  </declaration>
  <declaration>
    <declname>
      <name>address</name>&col;</declname>
    <application>
      <name>ADDRESS</name>
    </application>
  </declaration>
</declarations>
</Zedml>

```

รูปที่ ๗-1 รหัสต้นฉบับเค้าร่างการประกาศตัวแปร (ต่อ)



รูปที่ ๗-2 เค้าร่างการประกาศตัวแปรบนเว็บเบราว์เซอร์

## ประวัติผู้เขียนวิทยานิพนธ์

นายรัฐเขต หาญชนะ เกิดเมื่อวันที่ 22 พฤษภาคม พ.ศ. 2517 ที่อำเภอเชียงของ จังหวัดเชียงราย สำเร็จการศึกษาหลักสูตรบริหารธุรกิจบัณฑิต (วธ.บ.) สาขาคอมพิวเตอร์ธุรกิจ คณะบริหารธุรกิจ วิทยาลัยโยนง เมื่อปีการศึกษา 2539 และเข้าศึกษาต่อหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิตมหาบัณฑิต (วท.ม.) ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2543



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย