การใช้แอคทีฟเรพลิเคชันร่วมกับความเสถียรของระบบเลกาซีสำหรับการย้ายระบบ

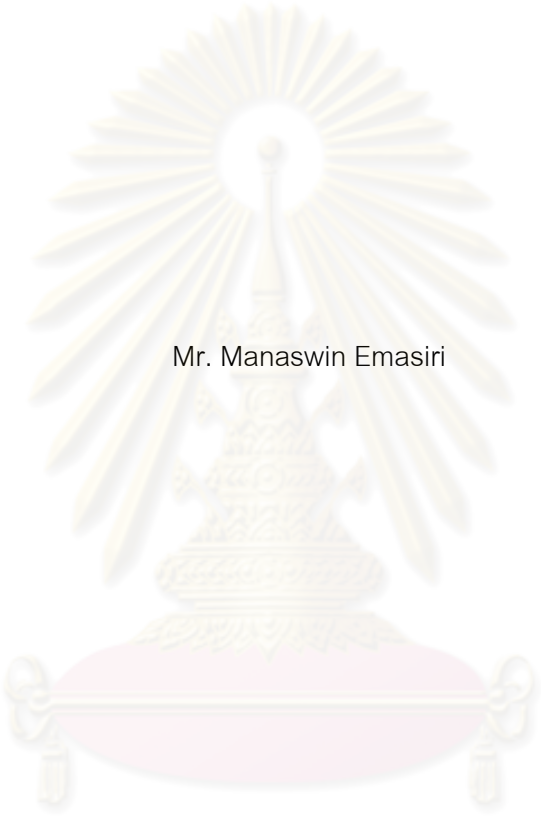
นายมนัสวิน เอมะศิริ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2551
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

UTILIZING ACTIVE REPLICATION AND STABILITY OF LEGACY SYSTEM FOR SYSTEM MIGRATION

Mr. Manaswin Emasiri



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Computer Science
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2008

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การใช้แอคทีฟเรพลิเคชันร่วมกับความเสถียรของระบบเลกาซีสำหรับการย้ายระบบ
โดย	นายมนัสวิน เอมะศิริ
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	อาจารย์ ดร. ยรรยง เต็งอำนวย

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(อาจารย์ จารุมาตร ปิ่นทอง)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(อาจารย์ ดร.ยรรยง เต็งอำนวย)

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

..... กรรมการภายนอกมหาวิทยาลัย
(อาจารย์ ดร.เบญจพร ลิ้มธรรมาภรณ์)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

มนัสวิน เอมะศิริ : การใช้แอคทีฟเรพลิเคชันร่วมกับความเสถียรของระบบเลกาซีสำหรับการย้ายระบบ . (UTILIZING ACTIVE REPLICATION AND STABILITY OF LEGACY SYSTEM FOR SYSTEM MIGRATION) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: อ.ดร.ยรรยง เต็งอำนวย, 75หน้า.

แม้ว่าในปัจจุบันเทคโนโลยีสารสนเทศได้ถูกพัฒนาอย่างต่อเนื่องเพื่อเพิ่มประสิทธิภาพการทำงานของภาคธุรกิจ แต่องค์กรส่วนใหญ่ยังใช้งานระบบเลกาซีอยู่ เนื่องจากไม่สามารถรับความเสี่ยงได้หากระบบที่พัฒนาด้วยเทคโนโลยีใหม่นั้นให้ผลการทำงานที่แตกต่างจากระบบเลกาซี งานวิจัยนี้จึงได้นำเสนอโมเดลที่ให้ระบบเลกาซีมาทำงานคู่ขนานกับระบบใหม่แบบแอคทีฟเรพลิเคชันและสรุปผลลัพธ์กลับไปยังผู้เรียกใช้ โดยการสรุปผลลัพธ์นั้นยึดตามแบบจำลองนโยบายซึ่งทำให้สามารถควบคุมการสรุปผลลัพธ์ได้ตามสถานการณ์ โดยสิ่งที่ได้จากงานวิจัยนี้นอกจากผลลัพธ์การทำงานจะน่าเชื่อถือมากขึ้นแล้ว ยังเสริมการทนทานต่อความเสียหายของระบบโดยรวมด้วย และสามารถใช้เป็นเครื่องมือช่วยทดสอบหาข้อบกพร่องให้กับระบบใหม่ได้อย่างมีประสิทธิภาพ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต มนัสวิน เอมะศิริ
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออ.ที่ปรึกษาวิทยานิพนธ์ ยรรยง
ปีการศึกษา 2551

4971457821 : MAJOR COMPUTER SCIENCE

KEYWORDS : SYSTEM MIGRATION / ACTIVE-REPLICATION / FAULT-TOLERANCE

MANASWIN EMASIRI : UTILIZING ACTIVE REPLICATION AND STABILITY OF LEGACY SYSTEM FOR SYSTEM MIGRATION. ADVISOR : YUNYONG TENG-AMNUAY, Ph.D., 75 pp.

Eventhough technology is continuously developed to increase business efficiency, most organization still use legacy system because of the risk of differing results of the new system. This research offers model that uses legacy system in parallel to the new system, in active-replication form, and generates result based on voting. Policy model is used to customize the voting process. This research provides more reliable results, higher system stability, and helps find defects of the new system.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Department : Computer Engineering Student's Signature มนัสวิน เอมะสิริ

Field of Study : Computer Science Advisor's Signature ยันยง ตรงงาม

Academic Year : 2008

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความอนุเคราะห์อย่างดียิ่งของท่านอาจารย์ ดร. ยรรยง เต็งอำนวย อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้สละเวลาให้ความรู้ ให้คำปรึกษา ตรวจสอบ ให้คำแนะนำแนวทางการวิจัย และสนับสนุนเป็นอย่างดี จนทำให้การวิจัยในครั้งนี้สำเร็จออกมาด้วยดี

ขอขอบพระคุณท่านอาจารย์ จารุมาตร ปิ่นทอง ท่านรองศาสตราจารย์ ดร.ทวิตีย์ เสนีวงศ์ ณ อยุธยา และท่านอาจารย์ ดร.เบญจพร ลิ้มธรรมาภรณ์ กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลา ให้คำแนะนำ ตรวจสอบ และแก้ไขวิทยานิพนธ์ฉบับนี้

ขอกราบขอบพระคุณ คุณพ่อ-คุณแม่ ที่ให้การสนับสนุนและเป็นกำลังใจที่ดีเสมอมา ขอขอบคุณพี่ๆ ธนาคารกรุงเทพ จำกัด (มหาชน) ที่ให้โอกาส ให้คำปรึกษา และให้การสนับสนุนเป็นอย่างดี

ท้ายที่สุด ผู้วิจัยขอขอบคุณเพื่อนๆ ทุกๆ คน ที่คอยติดตามและให้กำลังใจ รวมถึงท่านอื่นๆ ที่มิได้กล่าวชื่อไว้ ณ ที่นี้ที่มีส่วนทำให้วิทยานิพนธ์สำเร็จได้ด้วยดี

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ง

บทคัดย่อภาษาอังกฤษ..... จ

กิตติกรรมประกาศ..... ฉ

สารบัญ..... ช

สารบัญตาราง..... ฌ

สารบัญภาพ..... ญ

บทที่ 1 บทนำ..... 1

1.1 ความเป็นมาและความสำคัญของปัญหา..... 1

1.2 วัตถุประสงค์ของการวิจัย..... 2

1.3 ขอบเขตของการวิจัย..... 2

1.4 ประโยชน์ที่คาดว่าจะได้รับ..... 3

1.5 วิธีดำเนินการวิจัย..... 3

1.6 ลำดับขั้นตอนในการเสนอผลการวิจัย..... 4

1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์..... 4

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง..... 5

2.1 แอคทีฟ เรพลิเคชัน 5

2.2 แบบจำลองนโยบาย 5

2.3 ตัวห่อ..... 6

2.4 ดับบลิวเอสมัลติคาสต์ 6

2.5 การลงคะแนนเสียง 7

2.6 โอเลดีบี 8

2.7 ระบบเลกาซี 9

2.8 การย้ายระบบเลกาซี 9

บทที่ 3 วิธีดำเนินการวิจัย..... 11

3.1 แนวคิดในการแก้ปัญหา..... 11

3.2 ส่วนประกอบโมเดลสำหรับการย้ายระบบเลกาซี..... 15

3.3 การทำงานของโมเดลการย้ายระบบ..... 34

บทที่ 4 การพัฒนาและทดสอบระบบสนับสนุน..... 39

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4.1 การออกแบบระบบสนับสนุน..... 39
4.2 ความต้องการด้านหน้าที่ของระบบสนับสนุน 40
4.3 การพัฒนาระบบสนับสนุน..... 42
4.4 การทดสอบงานวิจัย..... 55
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ..... 68
5.1 สรุปผลการวิจัย..... 68
5.2 ปัญหาที่พบจากงานวิจัย..... 70
5.3 แนวทางการวิจัยต่อ..... 71
รายการอ้างอิง..... 72
ประวัติผู้เขียนวิทยานิพนธ์..... 75

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญตาราง

ตารางที่		หน้า
3.1	ข้อมูลที่ต้องใช้ในการลงทะเบียนระบบ.....	17
3.2	ตัวอย่างการกำหนดนโยบายให้เหมาะสมกับสถานการณ์.....	26
3.3	ผลลัพธ์และสถานะฐานข้อมูลของแต่ละระบบ.....	26
3.4	ผลลัพธ์และสถานะฐานข้อมูลของแต่ละระบบ.....	32
4.1	ความต้องการด้านหน้าที่ของระบบสนับสนุน.....	40
4.2	ฟังก์ชันของระบบเลกาซี.....	56
4.3	จำนวนระเบียนแต่ละตารางภายในฐานข้อมูลของระบบเลกาซี.....	57
4.4	ฟังก์ชันของระบบรีเอมพลิเมนท์.....	58
4.5	ผลการทดสอบด้านความถูกต้องของผลลัพธ์การทำงาน.....	64
4.6	ผลการทดสอบด้านความคงทนต่อความเสียหาย.....	65
4.7	ผลการทดสอบด้านความเร็วในการตอบกลับ.....	66

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

ภาพที่		หน้า
2.1	แผนภาพการทำงานของแอคทีฟเรพลิชัน.....	5
2.2	ตัวอย่างการใช้คำสั่ง send ของดับบลิวเอสมัลติคาสต์.....	7
2.3	การเข้าถึงข้อมูลผ่านทางโอเลดีบี.....	8
3.1	โมเดลการย้ายระบบ.....	14
3.2	ขั้นตอนการห่อระบบเลกาซี.....	15
3.3	การเรียกใช้งานฟังก์ชันของเลกาซีที่ผ่านการห่อด้วยชุดเครื่องมือโซ๊ฟ.....	16
3.4	การรีอิมพลิเมนท์ฟังก์ชันของระบบเลกาซีด้วยเทคโนโลยีเว็บเซอร์วิซ.....	17
3.5	เส้นทางการส่งข้อความร้องขอเมื่อเปลี่ยนไปใช้งานพรอกซี.....	19
3.6	การรับการร้องขอเข้าสู่ตัวกลางด้วยพรอกซี.....	20
3.7	นโยบายการคัดเลือกผลลัพธ์สำหรับระบบค้นหาอุณหภูมิ.....	25
3.8	การกระจายการร้องขอ.....	27
3.9	การรับการตอบกลับ.....	28
3.10	การตรวจสอบสถานะฐานข้อมูลด้วยคำค้นข้อมูลจากตารางติดตาม.....	29
3.11	การตรวจสอบสถานะฐานข้อมูลด้วยวิธีการห่อฐานข้อมูล.....	30
3.12	การตรวจสอบและคัดเลือกผลลัพธ์.....	31
3.13	ขั้นตอนคัดเลือกผลลัพธ์.....	32
3.14	ขั้นตอนการคัดเลือกผลลัพธ์เมื่อน้ำหนักรวมเท่ากัน.....	33
3.15	การทำงานของโมเดลการย้ายระบบ.....	37
4.1	แผนภาพยูสเคสของระบบสนับสนุน.....	39
4.2	แผนภาพอีอาร์ของฐานข้อมูล.....	40
4.3	แสดงส่วนต่อประสานผู้ใช้สำหรับลงทะเบียนระบบ.....	44
4.4	แสดงการลงทะเบียนเว็บเซอร์วิซ.....	45
4.5	แสดงการลงทะเบียนระบบเสร็จสมบูรณ์.....	45
4.6	แสดงการแก้ไขระบบเสร็จสมบูรณ์.....	46
4.7	แสดงการลบระบบเสร็จสมบูรณ์.....	47
4.8	การแนะนำคำสั่งด้วยอินเทลลิเซนท์	48
4.9	แสดงการตรวจสอบไวยากรณ์ของนโยบาย.....	49
4.10	แสดงการแปลความหมายจากนโยบาย.....	49

4.11	ส่วนต่อประสานผู้ใช้สำหรับควบคุมระบบสนับสนุน.....	51
4.12	ขั้นตอนทำงานทั้งหมดของระบบสนับสนุน.....	52
4.13	ภาพรวมการทดสอบงานวิจัย.....	55
4.14	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบชื่อผู้ใช้และรหัสผ่าน	60
4.15	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบระดับของผู้ใช้.....	60
4.16	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันเปลี่ยนรหัสผ่านของผู้ใช้.....	60
4.17	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันเพิ่มผู้ใช้เข้าสู่ระบบ.....	61
4.18	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันแก้ไขข้อมูลผู้ใช้.....	61
4.19	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันลบข้อมูลผู้ใช้.....	61
4.20	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบการซ้ำของชื่อผู้ใช้.....	62
4.21	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันส่งเช็คเดินทางไปยังสาขา.....	62
4.22	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันรับเช็คเดินทางมาจากตัวแทน.....	62
4.23	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบเช็คเดินทางภายในคลังกลาง.....	63
4.24	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบการซ้ำของเช็คเดินทาง.....	63
4.25	นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันค้นหาเช็คเดินทาง.....	63
4.26	กราฟแสดงความเร็วในการตอบกลับ.....	67

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันเว็บเซอร์วิซ (web service) เป็นที่ยอมรับและใช้งานอย่างแพร่หลายทำให้องค์กรธุรกิจต่างๆ ได้หันมาพัฒนาระบบเว็บเซอร์วิซเพื่อให้ทำงานแทนระบบดั้งเดิมที่ใช้งานอยู่ที่เรียกว่าระบบเลกาซี (legacy system) แต่ปัญหาสำหรับการเปลี่ยนไปใช้งานเว็บเซอร์วิซเพื่อทดแทนระบบเลกาซียังคงมีอยู่ เช่น ระบบที่ต้องเกี่ยวข้องกับความปลอดภัยของชีวิตและทรัพย์สิน หรือระบบที่สามารถยอมรับข้อผิดพลาดได้น้อยมาก เช่น ธุรกิจด้านการบิน หรือ ตลาดหลักทรัพย์ การเปลี่ยนไปใช้งานเว็บเซอร์วิซเพื่อทดแทนระบบเลกาซีนั้นจะมีความเสี่ยงสูง [1] ดังนั้นจึงต้องมีสิ่งที่รับประกันว่าเว็บเซอร์วิซที่นำมาใช้แทนนั้นสามารถทำงานได้อย่างถูกต้องและคงทนต่อความเสียหาย

ในงานวิจัยนี้ได้นิยามระบบดั้งเดิมหรือระบบเลกาซีไว้ว่าเป็น "โปรแกรมประยุกต์ (application) ที่พัฒนาขึ้นด้วยเทคโนโลยีที่ตกรุ่นส่งผลให้การบำรุงรักษาทำได้ยากเนื่องจากขาดแคลนบุคลากรที่มีความรู้ความเข้าใจและฮาร์ดแวร์ที่เข้ากันได้กับตัวโปรแกรมประยุกต์" [1] ทั้งนี้จะเห็นได้ว่าระบบดั้งเดิมไม่จำเป็นต้องเป็นเทคโนโลยีสมัยเก่ามาก เทคโนโลยีอาจเป็นแบบที่ใช้งานกันอยู่อย่างแพร่หลายแต่มีเทคโนโลยีใหม่ที่เริ่มได้รับความนิยมจนกระทั่งระบบสนับสนุนและบุคลากรของเทคโนโลยีดั้งเดิมเริ่มขาดแคลนซึ่งส่งผลให้ต้องเริ่มคำนึงถึงการย้ายไปสู่เทคโนโลยีใหม่

งานวิจัยนี้มีแนวคิดที่ทำให้เว็บเซอร์วิซที่นำมาทดแทนระบบเลกาซีมีความน่าเชื่อถือมากขึ้น ทั้งในด้านความเสถียรในการทำงานและผลของการทำงาน โดยการนำเอาระบบเลกาซีที่มีความน่าเชื่อถือในการทำงานสูงเนื่องจากใช้งานมาเป็นเวลานานมาทำงานคู่ขนาน (parallel) กับเว็บเซอร์วิซที่พัฒนาใหม่ในลักษณะของแอคทีฟเรพลิเคชัน (active replication) [2] และนำผลลัพธ์ที่ได้จากการทำงานทั้งสองระบบมาลงคะแนน (vote) และสรุปผล โดยประโยชน์ที่ได้รับจากงานวิจัยนี้ คือ การทำงานของระบบมีความเสถียรมากขึ้นและให้ผลการทำงานที่เชื่อถือได้

1.2 วัตถุประสงค์ของการวิจัย

งานวิจัยชิ้นนี้มีจุดประสงค์เพื่อกำหนดโมเดลที่นำเอาระบบเลกาซีมาทำงานคู่ขนานกับระบบใหม่ในลักษณะของแอคทีฟเรพลิเคชันและใช้การลงคะแนนเสียงสำหรับทดสอบความน่าเชื่อถือของระบบใหม่เพื่อสนับสนุนการย้ายระบบ

1.3 ขอบเขตของการวิจัย

1. ตรวจสอบผลลัพธ์ระหว่างระบบเลกาซีและระบบที่พัฒนาขึ้นใหม่และสรุปผลลัพธ์เพียงหนึ่งผลลัพธ์ส่งให้กับผู้ขอบริการโดยการสรุปผลลัพธ์นั้นยึดตามนโยบาย

2. การทำแอคทีฟเรพลิเคชันระหว่างระบบเลกาซีและระบบที่ได้รับการพัฒนาใหม่เพื่อรองรับในกรณีที่ระบบใดระบบหนึ่งได้รับความเสียหายจะไม่ส่งผลกระทบกับอีกระบบ

3. ระบบที่ได้รับการพัฒนาใหม่ต้องอยู่ในรูปของเว็บเซอร์วิซโดยอาจจะมีการปฏิบัติงานที่เหมือนหรือไม่เหมือนกับระบบเลกาซีก็ได้ ซึ่งการกำหนดเปรียบเทียบการปฏิบัติงานเป็นหน้าที่ของผู้ติดตั้งระบบ (deployer) ที่จะกำหนดไว้ที่แบบจำลองนโยบาย

4. เว็บเซอร์วิซของระบบที่พัฒนาใหม่ต้องเป็นแบบไร้สถานะ (stateless) การควบคุมธุรกรรม (transaction control) เป็นหน้าที่ของโปรแกรมประยุกต์ของระบบที่พัฒนาใหม่

5. ระบบเลกาซีที่นำมาทำงานคู่ขนานกันกับระบบที่พัฒนาใหม่ต้องผ่านการห่อ (wrap) ให้กลายเป็นเว็บเซอร์วิซที่เป็นแบบไร้สถานะ

6. การตรวจสอบผลลัพธ์การทำงานเซอร์วิซตัวกลางจะตรวจสอบผลลัพธ์แบบการปฏิบัติงานต่อการปฏิบัติงานและตรวจสอบเฉพาะการปฏิบัติงานที่ปรากฏอยู่ในเอกสารดับบลิวเอสดีแอลของแต่ละระบบ โดยการปฏิบัติงานนั้นต้องมีจุดประสงค์การทำงานที่เหมือนกัน

7. ในกรณีที่ระบบได้รับความเสียหายหรือให้ผลลัพธ์ที่ผิดพลาดเว็บเซอร์วิซตัวกลางจะรายงานผลลัพธ์ที่ผิดพลาดให้กับผู้ดูแลระบบได้ทราบแต่จะไม่กู้ให้เว็บเซอร์วิซนั้นกลับมาทำงานหรือ ให้ผลลัพธ์การทำงานที่ถูกต้องได้

8. ระบบเลกาซีและระบบที่พัฒนาใหม่ต้องใช้ฐานข้อมูลแบบสัมพันธ์ (relational database) ที่มีโครงสร้างฐานข้อมูล (database schema) เหมือนกัน

9. เว็บเซอร์วิซตัวกลางจะไม่ตรวจสอบการทำงานที่เกิดจากคำสั่งจัดการโครงสร้างข้อมูล (data definition language - DDL)

10.ในกรณีที่มีคำสั่งในการจัดการฐานข้อมูล (data manipulation language - DML) เว็บเซอร์วิซตัวกลางสามารถตรวจสอบสถานะฐานข้อมูลของทั้งระบบเลกาซีและระบบที่ได้รับการพัฒนาใหม่ หากมีความแตกต่างเกิดขึ้นระบบจะรายงานให้กับผู้ดูแลระบบได้ทราบ

11.ในกรณีที่มีการใช้คำสั่งจัดการฐานข้อมูล ระบบที่พัฒนาขึ้นมาใหม่จะต้องส่งคำสั่งจัดการฐานข้อมูลนั้นมายังเซอร์วิซตัวกลาง

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถสร้างความเชื่อมั่นในการเปลี่ยนจากระบบเก่าไปเป็นระบบใหม่ให้กับองค์กรแม้ระบบใหม่จะทำงานผิดพลาดจะไม่ ส่งผลกระทบต่อการทำธุรกรรม

2. สามารถใช้เป็นเครื่องมือในการทำการทดสอบ (system test) ให้กับระบบใหม่ได้โดยใช้สภาพแวดล้อมที่เหมือนจริง

3. สามารถทำให้ธุรกรรมขององค์กรดำเนินต่อไปได้ แม้ระบบใดระบบหนึ่งจะได้รับภัยพิบัติจากเหตุการณ์ต่างๆ

1.5 วิธีดำเนินการวิจัย

1. ศึกษาทฤษฎีพื้นฐานของการทำเว็บเซอร์วิซ, การทำเรพลิเคชันและวิธีการย้ายระบบเลกาซี

2. สำรวจงานวิจัยที่เกี่ยวข้องกับการย้ายระบบที่เป็นแบบออนไลน์

3. ตั้งปัญหาในการย้ายระบบเลกาซีซึ่งไม่มีงานวิจัยรองรับ

4. สร้างแนวคิดโดยการออกแบบขั้นตอนวิธีการทำงาน

5. สร้างต้นแบบเว็บเซอร์วิซตัวกลางเพื่อใช้ในการทดสอบการทำงาน

6. ห่อระบบเลกาซีและสร้างต้นแบบระบบใหม่ซึ่งมีการทำงานเหมือนกับระบบเลกาซี

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

7. ใช้ต้นแบบทดสอบวิธีการที่ได้เสนอโดยการนำเอาระบบทั้งสองระบบมาเชื่อมต่อกับเซอร์วิซตัวกลางโดยให้ทำงานพร้อมกัน

8. สรุปผลและเรียบเรียงวิทยานิพนธ์

1.6 ลำดับขั้นตอนในการเสนอผลการวิจัย

วิทยานิพนธ์นี้แบ่งเนื้อหาออกเป็น 5 บทดังต่อไปนี้ บทที่ 1 เป็นบทนำซึ่งกล่าวถึงความเป็นมาและความสำคัญของปัญหา รวมถึงวัตถุประสงค์ของการวิจัย บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานและงานวิจัยที่เกี่ยวข้องในงานวิจัยนี้ บทที่ 3 กล่าวถึงการออกแบบขั้นตอนดำเนินงาน บทที่ 4 กล่าวถึงการพัฒนาและทดสอบระบบสนับสนุน บทที่ 5 กล่าวถึงการสรุปผลการวิจัยและข้อเสนอแนะ

1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตีพิมพ์เป็นบทความทางวิชาการในหัวข้อเรื่อง “การใช้แอคทีฟเรพลิเคชันร่วมกับระบบเลกาซีสำหรับการย้ายระบบ” โดย มนัสวิน เอมะศิริ และ ดร. ยรรยง เต็งอำนวย, ในงานประชุมวิชาการ “The 12th National Computer Science and Engineering Conference (NCSEC 2008)” ณ Long Beach Garden Hotel & Spa, พัทยา ระหว่างวันที่ 20-21 พฤศจิกายน 2551

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

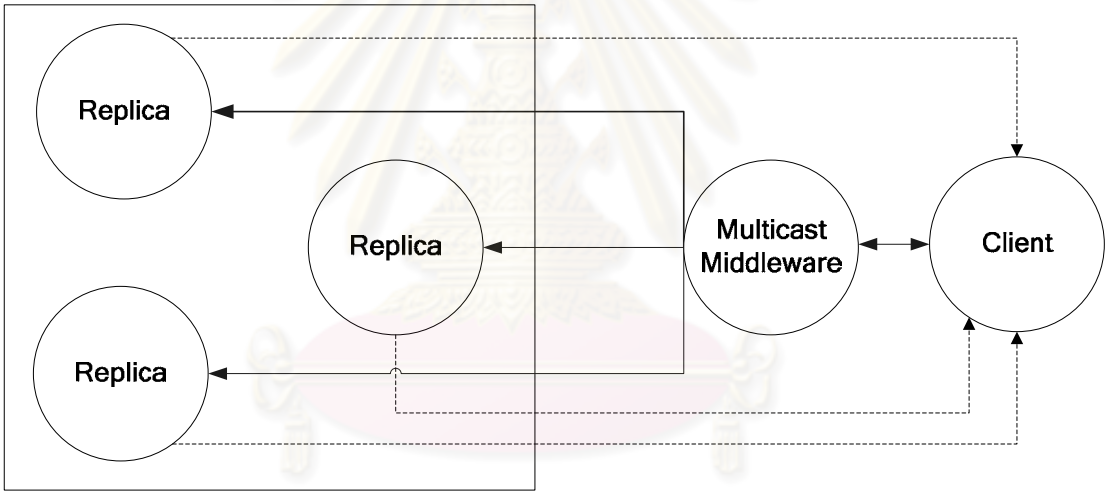
บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้กล่าวถึงประเด็นทางทฤษฎีซึ่งผู้วิจัยนำมาใช้ในงานวิจัย รวมทั้งงานวิจัยที่เกี่ยวข้องกัน ดังต่อไปนี้

2.1 แอคทีฟ เรพลิเคชัน

แนวทางแอคทีฟเรพลิเคชัน (active replication) [2] เป็นวิธีการที่ทำให้ทุกสำเนา (replica) ทำงานพร้อมกัน เมื่อมีการใช้งานทุกสำเนาจะทำงานไปพร้อมกัน เมื่อประมวลผลเสร็จแต่ละสำเนาจะส่งผลลัพธ์กลับมาเพื่อสรุปผลส่งให้กับผู้ร้องขอ ซึ่งผลลัพธ์ที่ได้จากทุกสำเนาควรเหมือนกัน ในกรณีที่บางสำเนาไม่สามารถทำงานได้จะไม่ส่งผลกระทบต่อผู้ใช้เนื่องจากสามารถใช้ผลลัพธ์จากสำเนาอื่นทดแทนได้



รูปที่ 2.1 แผนภาพการทำงานของแอคทีฟเรพลิชัน

2.2 แบบจำลองนโยบาย

หลักการของแบบจำลองนโยบาย (policy model) เป็นวิธีการที่นำเสนอโดย ดอบสัน [3] โดยการนำเอากระบวนการคงทนต่อความเสียหายต่างๆ ยกตัวอย่าง เช่น การพยายามทำซ้ำ (retry) การกู้คืน (recovery) การทำซ้ำ (redundancy) มาจัดให้เป็นรูปแบบที่สามารถปรับเปลี่ยนได้โดยการสื่อสารกับระบบผ่านทางภาษาเอ็กซ์เอ็มแอล และบังคับใช้กับเว็บเซอร์วิซทำให้เว็บเซอร์วิซที่ไม่ได้ถูกออกแบบให้มีความสามารถทางด้านความคงทนต่อความเสียหายมีความสามารถทางด้านนี้ โดยแบบจำลองนโยบายนั้นประกอบด้วย ส่วนย่อยนโยบาย (policy element) โดยส่วนย่อยนโยบายเป็นที่รวบรวมนโยบายทั้งหมดของแบบจำลองนโยบาย โดย

ภายในส่วนย่อยนโยบายประกอบด้วย กระบวนคำสั่ง (procedure) ย่อยๆ จำนวนมาก โดยแต่ละกระบวนคำสั่งสามารถส่งผลการทำงานไป-มาระหว่างกันได้โดยใช้ ส่วนย่อยการเชื่อมโยง (connection element) ในการเชื่อมต่อกันระหว่างกระบวนคำสั่ง ยกตัวอย่าง เช่น หลังจากที่กระบวนคำสั่งแม่ (parent procedure) ได้รับการร้องขอ กระบวนคำสั่งแม่จะส่งสารไปยังกระบวนคำสั่งลูก (child procedure) ตามที่ได้ระบุไว้ในส่วนการเชื่อมต่อ เมื่อมีการใช้งานส่วนนโยบายและส่วนกระบวนคำสั่งต่างๆ จะถูกเปลี่ยนให้อยู่ในรูปของคลาส (class) โดยแต่ละคลาสจะต้องมีส่วนต่อประสาน (interface) เพื่อที่จะสามารถเรียกใช้งานได้

2.3 ตัวห่อ

หลักการของตัวห่อ (wrapper) [4] คือ ส่วนโปรแกรม (component) ที่ทำให้โปรแกรมที่มีโครงสร้างพื้นฐานหรือสภาพแวดล้อมที่ต่างกันสามารถทำงานร่วมกันได้ โดยคลุมหรือครอบการทำงานของโปรแกรมใดโปรแกรมหนึ่งไว้ และจัดหาหรือสร้างส่วนต่อประสานที่เหมาะสมกับอีกโปรแกรมหนึ่ง ในกรณีที่ระบบเลกาซีทำงานกับระบบ ผ่านทางส่วนต่อประสานโปรแกรมประยุกต์ (API: application program interface) ตัวห่อจะทำหน้าที่เป็นตัวกลางประสานการทำงานระหว่างระบบเลกาซีและระบบใหม่ โดยจัดหาส่วนต่อประสานโปรแกรมประยุกต์ที่เหมาะสมให้

ในกรณีที่ระบบเลกาซีทำงานกับผู้ใช้โดยใช้ส่วนต่อประสานผู้ใช้แบบเต็มจอ หรือแบบกราฟฟิกส์ (full-screen or graphic user interface) ตัวห่อจะอยู่ในรูปของตัวขูดหน้าจอ (screen scraping) [5] ซึ่งทำหน้าที่ในการเปลี่ยนหน้าจอของระบบเลกาซีให้อยู่ในรูปของการเรียกใช้งานฟังก์ชัน (function call) เมื่อผู้ใช้ทำการร้องขอ ตัวขูดหน้าจอจะนำรายละเอียดการร้องขอเข้าไปใส่ในส่วนต่อประสานผู้ใช้ของระบบเลกาซีที่เหมาะสม เมื่อระบบเลกาซีประมวลผลเสร็จ ตัวขูดหน้าจอจะนำเอาผลลัพธ์ที่ได้เปลี่ยนเป็นรูปแบบที่เหมาะสมส่งให้กับผู้ใช้

2.4 ดับบลิวเอสมัลติคาสต์

หลักการของดับบลิวเอสมัลติคาสต์ (WS-multicast) ถูกนำเสนอโดย จอร์จ ซาลาส [6] โดยออกแบบให้อยู่ในรูปของเว็บเซอร์วิซที่ทำหน้าที่สำหรับส่งข้อมูลแบบมัลติคาสต์ (multicast) ไปยังเว็บเซอร์วิซอื่นๆ โดยมีขั้นตอนการทำงาน ดังนี้คือ

1. ดับบลิวเอสมัลติคาสต์สร้างช่องทางการติดต่อสื่อสารในการติดต่อกับชื่อกลุ่ม (group name)

2. ดับบลิวเอสมัลติคาสต์เริ่มส่งข้อมูลโดยสามารถส่งได้ทั้งแบบกระจาย และแบบเจาะจง (unicast) โดยใช้คำสั่ง *Send* ในการส่งสาร (message) ไปยังเว็บเซอร์วิซที่ต้องการ ดังตัวอย่างในรูปที่ 2.3

Send (Message Destination, Source Address, Message)	
ปลายทาง (message destination)	คือ ที่อยู่ของเว็บเซอร์วิซปลายทางที่ต้องการจะส่งในกรณีที่ระบุ ถือว่าเป็นการส่งแบบเจาะจงแต่ถ้าไม่ระบุถือว่าเป็นการส่งแบบกระจาย
ที่อยู่ต้นทาง (source address)	คือ ที่อยู่ของเว็บเซอร์วิซต้นทางเพื่อใช้ในการตอบกลับ
สาร (message)	คือ สารที่ต้องการส่ง

รูปที่ 2.2 ตัวอย่างการใช้คำสั่ง *send* ของดับบลิวเอสมัลติคาสต์

3. สารที่ส่งมาจากเว็บเซอร์วิซอื่นๆ จะถูกเก็บไว้ในแถวคอย (queue) ของดับบลิวเอสมัลติคาสต์โดยดับบลิวเอสมัลติคาสต์จะใช้คำสั่ง *Receive* ดึงสารจากแถวคอยไปประมวลผล

4. เมื่อต้องการยกเลิกการติดต่อเว็บเซอร์วิซจะใช้คำสั่ง *Disconnect* เพื่อใช้ลบการร้องขอทั้งหมดออกจากแถวคอย และใช้คำสั่ง *Close* เพื่อยกเลิกการติดต่อ

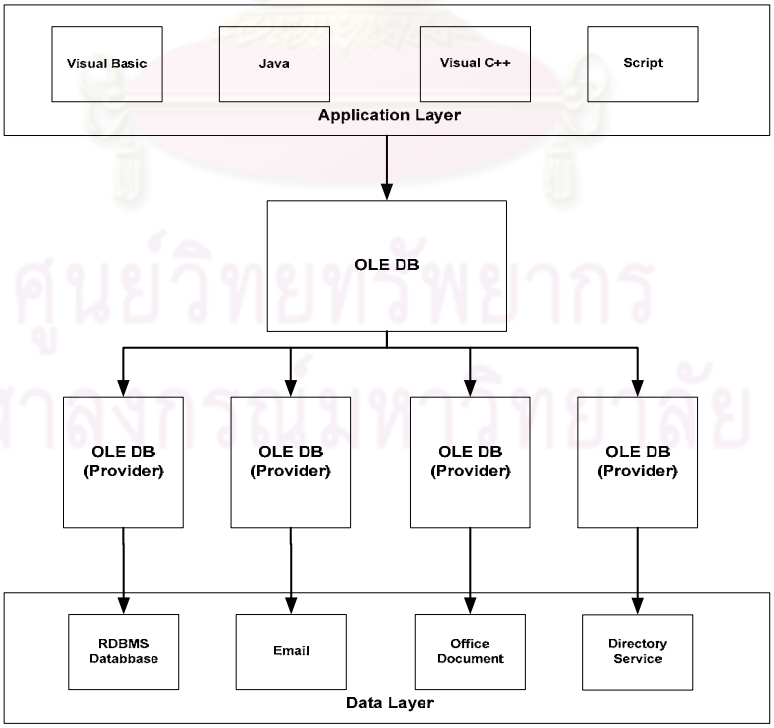
2.5 การลงคะแนนเสียง

การลงคะแนนเสียง (voting) คือการนำเอาหลายผลลัพธ์มาสรุปผลให้ได้ผลลัพธ์เดียวซึ่งน่าเชื่อถือมากที่สุด [7] ซึ่งแต่เดิมนั้นการลงคะแนนเสียงได้ถูกใช้เฉพาะในฮาร์ดแวร์ [8] แต่ปัจจุบันการลงคะแนนเสียงได้ถูกนำมาใช้เพื่อสรุปผลให้กับซอฟต์แวร์ที่ทำงานแบบคู่ขนานและงานที่ต้องการความคงทนต่อความเสียหายที่สูง [9]

การลงคะแนนเสียงมีหลายวิธี เช่น แบบเสียงส่วนใหญ่ (majority voting) ออกเสียงหลายเสียง (plurality voting) ถัวเฉลี่ยถ่วงน้ำหนัก (weighted average voting) หรือค่าเฉลี่ย (median voting) แต่ละวิธีมีขั้นตอนการโหวตที่แตกต่างกันและเหมาะสมกับสถานการณ์แตกต่างกันออกไป ดังแจกแจงไว้ในงานวิจัยของ ล็อคแซค [10]

2.6 โอเลดีบี

เนื่องจากปัจจุบันข้อมูลจะอยู่ในรูปแบบที่หลากหลายมากกว่าในอดีตส่งผลให้เกิดปัญหาการเรียกใช้ข้อมูลของโปรแกรมประยุกต์ที่ต้องออกแบบโปรแกรมให้ตรงกับรูปแบบข้อมูลที่ต้องการเรียกใช้ ดังนั้นจึงได้เกิดแนวคิดใหม่ที่เรียกว่า ยูดีเอ (UDA: universal data access) ขึ้น [11] แนวคิดนี้มีจุดมุ่งหมายเพื่อให้เกิดศูนย์กลางการติดต่อระหว่างชั้นโปรแกรมประยุกต์ (application layer) กับชั้นข้อมูล (data layer) เพื่อให้โปรแกรมประยุกต์สามารถเรียกใช้ข้อมูลได้โดยไม่ต้องเปลี่ยนโครงสร้างของโปรแกรมประยุกต์ ซึ่งแนวคิดนี้คล้ายกับ โอดีบีซี (ODBC: open database connectivity) แต่แตกต่างที่โอดีบีซีใช้ได้กับฐานข้อมูลแบบสัมพันธ์เท่านั้น เพื่อให้สามารถนำแนวคิดยูดีเอไปใช้งานทางไมโครซอฟท์จึงได้พัฒนาเทคโนโลยีที่มีชื่อว่า โอเลดีบี (OLE DB: object linking and embedding database) [12] ขึ้นมาทดแทนโอดีบีซี โดยโอเลดีบีได้แบ่งออกเป็นสองส่วนคือ ส่วนของผู้ให้บริการข้อมูล (data provider) และส่วนของผู้บริโภคข้อมูล (data consumer) โดยส่วนผู้ให้บริการข้อมูลจะทำหน้าที่จัดหาข้อมูลให้กับส่วนผู้บริโภคข้อมูล ในการทำให้ส่วนผู้ให้บริการข้อมูลสามารถติดต่อกับข้อมูลได้หลายรูปแบบ จำเป็นต้องมีการเตรียมผู้ให้บริการ (provider) ซึ่งเปรียบเสมือนตัวขับ (driver) ที่ใช้ในการติดต่อกับข้อมูลแต่ละรูปแบบไว้ ดังรูปที่ 2.3



รูปที่ 2.3 การเข้าถึงข้อมูลผ่านทางโอเลดีบี

2.7 ระบบเลกาซี

ระบบเลกาซี (legacy system) [1] คือ ระบบคอมพิวเตอร์หรือโปรแกรมประยุกต์ที่ล้าสมัยในเชิงเทคโนโลยีที่ใช้ในการพัฒนา แต่ยังคงใช้งานอยู่เนื่องจากผู้ใช้หรือองค์กรไม่สามารถเปลี่ยนไปใช้งานระบบที่มีเทคโนโลยีใหม่ได้เนื่องจากปัญหาการย้ายระบบ ระบบเลกาซีนั้นยากในการบำรุงรักษาและการปรับปรุง เนื่องจากขาดบุคลากรที่มีความรู้ความเข้าใจในระบบรวมถึงการทำงานร่วมกันกับระบบใหม่ไม่สามารถทำได้เนื่องจากถูกพัฒนาด้วยเทคโนโลยีที่แตกต่างกัน แม้ว่าระบบเลกาซีจะมีข้อเสียต่างๆ มากมายแต่องค์กรยังคงใช้ระบบเลกาซีในการทำงานอยู่ เพราะปัจจัยต่างๆ ดังนี้

1. ระบบเลกาซีส่วนใหญ่เป็นระบบที่ใหญ่และซับซ้อนเป็นหัวใจหลักในการทำธุรกรรมขององค์กร

2. ระบบเลกาซีส่วนใหญ่มีความสามารถในการใช้งานได้ของระบบที่สูง (100% availability) เนื่องจากใช้งานมาเป็นเวลานาน

3. การที่จะเข้าใจการทำงานของระบบเลกาซีเพื่อมาออกแบบระบบใหม่นั้นทำได้ยากเนื่องจากระบบเลกาซีถูกพัฒนาด้วยเทคโนโลยีที่ล้าสมัย เช่น ภาษาโคบอล (COBOL) หรือฟอร์แทรน (FORTRAN) ทำให้ขาดแคลนบุคลากรที่มีความรู้ความเข้าใจในรหัสต้นฉบับ (Source Code) ของภาษานั้นๆ

2.8 การย้ายระบบเลกาซี

ในหลายปีที่ผ่านมาได้มีการนำเสนอ วิธีและเครื่องมือสำหรับการย้ายระบบเลกาซี (legacy migration) โดยการพัฒนาระบบเป้าหมาย (target system) ให้มีฟังก์ชันการทำงาน และมีข้อมูลต้นฉบับเหมือนในระบบเดิม แต่ต้องใช้ประโยชน์เทคโนโลยีใหม่ได้และง่ายสำหรับการดูแลรักษารวมทั้งสามารถปรับแต่งให้ตรงตามความต้องการของธุรกิจที่มีการปรับเปลี่ยนอยู่สม่ำเสมอได้

ราห์โกซา และ โอรอบเชียน [13] ได้นำเสนอการจำแนกแนวทางรวมทั้งวิธีการที่เกี่ยวกับการย้ายระบบเลกาซี ทั้งนี้ได้นำเสนอเครื่องมือและวิธีการเพื่อช่วยลดเวลา แรงงาน ต้นทุน และความเสี่ยงในกระบวนการย้ายระบบโดยเน้นรีเอ็นจิเนียร์ริ่ง (re-engineering) ส่วนต่อประสานผู้ใช้ ทั้งนี้แนวทางและวิธีการ ได้ใช้การวิเคราะห์ดาต้าโฟลว์ (data-flow analysis)

มัวร์ และ มอชคินา [14] ได้อธิบายถึงการย้ายจากส่วนติดต่อกับผู้ใช้ไปยังหน้าจอเว็บ รวมทั้งนำเสนอวิธีการและชุดเครื่องมือที่ใช้ในการออกแบบตัวห่อ (wrapper) เพื่อสนับสนุนการย้ายระบบ ทั้งนี้ยังมีแนวทางเพิ่มเติม คือ นำสถาปัตยกรรมเอสโอเอ (SOA: service oriented architecture) มาใช้

โดยทั่วไปการห่อและรีเวิร์สเอ็นจิเนียริ่งมีบทบาทสำคัญในการย้ายระบบเลกาซี ทั้งนี้สนีด [15] ได้นำเสนอเครื่องมือและเทคนิคการห่อ โดยแปลงโปรแกรมออนไลน์ไปเป็นอีกโปรแกรมหนึ่งที่มีการสร้างเอกสารเอ็กซ์เอ็มแอล (XML: extensible markup language)

มีหลายแนวทางที่ย้ายระบบเลกาซีไปยังระบบแบบเว็บซึ่งถูกนำเสนอโดยงานวิจัยของอาเวอร์ซาโน [16] [17] โดยนำเสนอผลลัพธ์ของการย้ายระบบเลกาซีมุ่งรวมระบบที่พัฒนาด้วยภาษาโคบอลให้เข้ากับโครงสร้างพื้นฐานแบบเว็บ

หยิง [18] ได้นำเสนอวิธีการย้ายระบบที่รวมระบบเลกาซีเข้าเป็นส่วนหนึ่งของระบบใหม่ (integration) โดยการใช้ระบบเลกาซีที่ผ่านการห่อมาทำงานร่วมกัน และค้นหาระบบเลกาซีที่เครื่องไคลเอนต้องการใช้งานโดยอัตโนมัติ ซึ่งงานวิจัยชิ้นนี้ ก่อนจะเริ่มการย้ายระบบจะต้องหยุดระบบทั้งหมดก่อน แล้วจึงนำส่วนประกอบระบบเลกาซี (legacy component) ที่ได้จากการห่อเข้าไปแทนที่ซึ่งต้องเสียเวลาทำการทดสอบระบบโดยรวมอีกครั้ง

นอกจากนี้ยังมีงานวิจัยด้านการทำงานคู่ขนาน (parallel running) โดยมีข้อมูลเข้าและการประมวลผลที่เหมือนกันและนำผลลัพธ์ที่ได้มาเปรียบเทียบกัน ทั้งนี้เพื่อพิสูจน์ความน่าเชื่อถือของระบบใหม่ ซึ่งเมื่อระบบใหม่ถูกยอมรับ ระบบเก่าก็จะถูกยกเลิกไป [19] โดยงานวิจัยที่ผ่านมามุ่งเน้นหาข้อผิดพลาดที่เกิดขึ้น ซึ่งได้พัฒนาเครื่องมือที่ช่วยในการตรวจสอบพฤติกรรมของโปรแกรมรวมทั้งโครงสร้างข้อมูลภายใน โดยมีแนวโน้มมุ่งไปที่การลดเวลาในการตรวจสอบลง

จากที่กล่าวมาแล้วนั้นทำให้ทราบถึงทฤษฎีที่นำมาใช้ในงานวิจัย นอกจากนี้ยังกล่าวถึงผลงานวิจัยที่เกี่ยวข้องกันด้วย จากนั้นในบทถัดไปจะกล่าวถึงผลงานวิจัยชิ้นนี้ ว่ามีแนวทางการดำเนินงานรวมทั้งผลที่ได้เป็นอย่างไร

บทที่ 3

วิธีดำเนินการวิจัย

บทที่ผ่านมาได้กล่าวให้ทราบถึงทฤษฎีต่างๆ รวมทั้งผลงานวิจัยที่เกี่ยวข้อง ในบทนี้จะกล่าวถึงการนำเอาทฤษฎีต่างๆ มาประยุกต์ใช้เพื่อออกแบบโมเดลสำหรับย้ายระบบเลกาซีซึ่งอาศัยความเสถียรของระบบเลกาซีร่วมกับแอคทีฟเรพลิเคชัน โดยเริ่มตั้งแต่เหตุผลในการออกแบบส่วนประกอบโมเดลส่วนต่างๆ เพื่อใช้เป็นข้อมูลให้กับการสร้างระบบสนับสนุนทดสอบ ในบทต่อไปด้วย

3.1 แนวคิดในการแก้ปัญหา

จากบทที่ 1 การย้ายระบบเลกาซีนั้นมีเป้าหมายเพื่อให้ได้มาซึ่งระบบที่สามารถใช้ประโยชน์จากเทคโนโลยีสมัยใหม่ในขณะที่ยังบรรลุเป้าประสงค์ขององค์กรได้ ดังนั้นในการย้ายระบบเลกาซีจึงจำเป็นต้องสร้างระบบซึ่งใช้เทคโนโลยีสมัยใหม่และมีการทำงานเช่นเดียวกับระบบเลกาซี

โดยผู้วิจัยได้นำรหัสต้นฉบับของระบบเลกาซีไปสร้างเลียนแบบด้วยเทคโนโลยีสมัยใหม่ ซึ่งในปัจจุบันเทคโนโลยีที่ได้รับความนิยมอย่างแพร่หลายคือ เว็บเซอร์วิซ ซึ่งวิธีการแก้ปัญหาการย้ายระบบเลกาซีภายใต้เงื่อนไขที่ว่าระบบใหม่นั้นต้องคงไว้ซึ่งความเสถียรของผลลัพธ์และการทำงานเช่นเดียวกับระบบเลกาซี คือ การให้ระบบเลกาซีเป็นต้นแบบให้กับระบบใหม่

ในงานวิจัยนี้จึงมีแนวคิดที่จะให้ทั้งสองระบบทำงานคู่ขนานกันในลักษณะของแอคทีฟเรพลิเคชันและลงคะแนนเพื่อเลือกผลลัพธ์ที่ได้จากทั้งสองระบบ แต่ในการทำให้ทั้งสองระบบสามารถทำงานคู่ขนานกันได้นั้นต้องให้ทั้งสองระบบใช้เทคโนโลยีเดียวกันก่อน โดยผู้วิจัยได้เปลี่ยนระบบเลกาซีให้ทำงานอยู่บนเทคโนโลยีเดียวกับระบบใหม่ด้วยวิธีการห่อ เมื่อทั้งระบบเลกาซีและระบบใหม่อยู่บนเทคโนโลยีเดียวกันแล้วก็จะสามารถทำงานร่วมกันได้

โดยในการทำงานนั้นระบบเลกาซีและระบบใหม่จะทำงานพร้อมกันตามหลักการของแอคทีฟเรพลิเคชัน ซึ่งระบบเลกาซีและระบบใหม่จำเป็นต้องได้รับการร้องขอจากไคลเอนท์พร้อมกัน ผู้วิจัยจึงได้นำกระบวนการกระจายส่ง (multicast) เพื่อกระจายการร้องขอไปยังระบบเลกาซีและระบบใหม่พร้อมกัน เมื่อได้รับการร้องขอแล้วระบบเลกาซีและระบบใหม่จะประมวลผลในลำดับการร้องขอที่เหมือนกัน เมื่อแต่ละระบบประมวลผลเสร็จจะส่งผลลัพธ์กลับมาเข้ากระบวนการลงคะแนนเพื่อหาคำตอบที่น่าเชื่อถือมากที่สุด ซึ่งผู้วิจัยได้เลือกการลงคะแนนแบบถ่วง

น้ำหนักมาประยุกต์ใช้ เนื่องจากเหมาะสมสำหรับการย้ายระบบที่มีการใช้ระบบเลกาซีมาเป็นต้นแบบ โดยกำหนดแนวทางการลงคะแนนโดยอิงนโยบาย เมื่อสรุปคำตอบได้จะส่งให้กับไคลเอนท์ ในกรณีที่มีความแตกต่างจะแจ้งให้ผู้ดูแลระบบได้ทราบเพื่อทำการแก้ไข ซึ่งจากแนวคิดนี้ผู้วิจัยตั้งสมมติฐานว่าการทำงานของระบบใหม่จะมีความเชื่อถือเพิ่มขึ้นเรื่อยๆ เมื่อเวลาผ่านไปจนเทียบเคียงระบบเลกาซี

อย่างไรก็ตามการย้ายระบบเลกาซีโดยการตรวจสอบผลลัพธ์เพียงอย่างเดียวนั้นไม่เพียงพอ เนื่องจากระบบเลกาซีส่วนใหญ่เป็นระบบที่เกี่ยวข้องกับองค์กรธุรกิจซึ่งมักมีการทำงานที่เกี่ยวข้องกับข้อมูลเป็นหลัก ดังนั้นในการย้ายระบบจึงจำเป็นต้องคำนึงถึงสถานะของฐานข้อมูลร่วมด้วย โดยใช้การตรวจสอบสถานะฐานข้อมูลของระบบใหม่เปรียบเทียบกับระบบเลกาซี

ในปัจจุบันระบบเลกาซีมีการใช้ฐานข้อมูลที่หลากหลาย ทำให้การตรวจสอบข้อมูลจำเป็นต้องมีตัวกลางที่ทำหน้าที่ติดต่อกับฐานข้อมูลที่หลากหลายได้ โดยผู้วิจัยได้นำเอาหลักการของโอเลดีบี ซึ่งได้กล่าวรายละเอียดไว้ในบทที่ 2 มาประยุกต์ใช้ จากภาพรวมการย้ายระบบที่กล่าวมาข้างต้น จำเป็นต้องนำกระบวนการแนวคิดและทฤษฎีต่างๆ เข้ามาประกอบกันโดยมีรายละเอียดดังต่อไปนี้

1. <u>การสร้างระบบเลียนแบบระบบเลกาซี</u> เนื่องจากการย้ายระบบเลกาซีคือ การสร้างระบบที่ใช้เทคโนโลยีสมัยใหม่แต่ยังคงการทำงานเดิมของระบบเลกาซีไว้ ดังนั้นผู้วิจัยจึงนำระบบเลกาซีมาสร้างเลียนแบบด้วยเทคโนโลยีเว็บเซอร์วิซ โดยให้หนึ่งฟังก์ชัน (function) ของระบบเลกาซีแทนด้วยหนึ่งเซอร์วิซ (service) ของระบบใหม่

2. <u>การห่อระบบเลกาซี</u> การทำให้ระบบเลกาซีซึ่งเป็นต้นแบบสามารถทำงานคู่ขนานกับระบบที่ใช้เทคโนโลยีเว็บเซอร์วิซได้นั้นจำเป็นต้องทำให้ระบบเลกาซีอยู่ในรูปแบบของเว็บเซอร์วิซด้วย โดยผู้วิจัยได้เลือกใช้วิธีการห่อด้วยชุดเครื่องมือโซปของไมโครซอฟท์ รุ่น 2 (Microsoft SOAP Toolkit version 2)

3. <u>กระบวนการรับการร้องขอ</u> โดยทั่วไปการร้องของานในลักษณะของเว็บเซอร์วิซนั้น ไคลเอนท์ (client) จะส่งการร้องขอ (request) ไปยังเว็บเซอร์วิซของระบบใหม่โดยตรงเพื่อขอใช้บริการ แต่ในกรณีที่ต้องการคำตอบที่น่าเชื่อถือนั้นจำเป็นต้องกระจายการร้องขอนั้นไปยังเว็บเซอร์วิซของระบบเลกาซีซึ่งห่อแล้วที่เป็นต้นแบบด้วย ดังนั้นจึงต้องมีตัวกลาง (middleware) เพื่อรับการร้องขอนั้นมากระจายส่งให้กับแต่ละระบบ ซึ่งวิธีการส่งการร้องขอมาที่ตัวกลางแทนการส่งไปที่เว็บเซอร์วิซเดิมโดยให้มีผลกระทบต่อวิธีการร้องขอของไคลเอนท์น้อยที่สุด คือ การนำพรอก

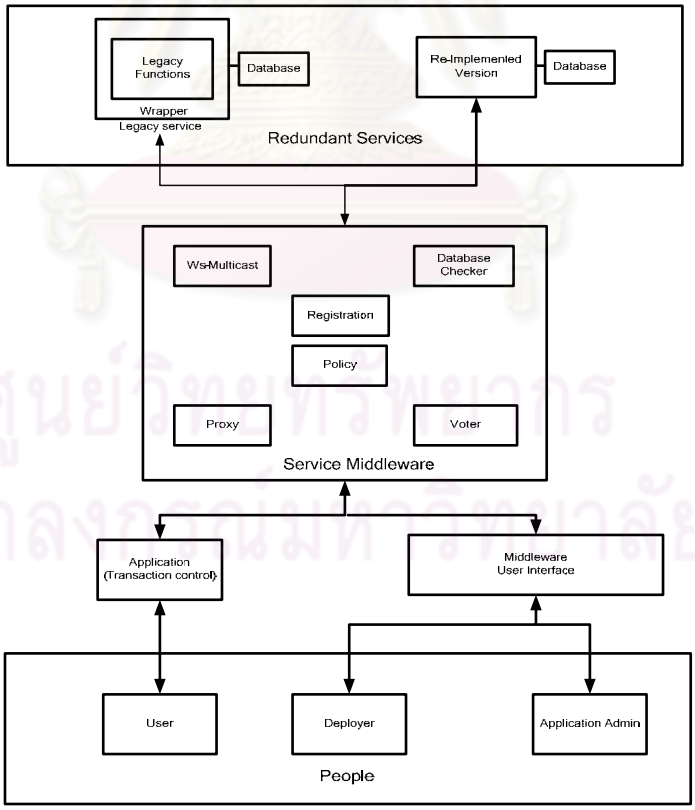
ซี (proxy) มารับการร้องขอนั้น โดยพรอกซีจะทำหน้าที่เสมือนเว็บเซอร์วิซจำลองซึ่งมีวิธีการร้องขอเหมือนเว็บเซอร์วิซที่พรอกซีเข้าไปแทนที่ทุกประการ แต่หน้าที่การทำงานภายในของพรอกซีจะเป็นการรับการร้องขอนั้นมาส่งให้กับตัวกลาง โดยตัวกลางจะเรียกใช้ดับบลิวเอสมัลติคาสต์(ในหัวข้อถัดไป)เพื่อกระจายส่งการร้องขอนั้นต่อไป

4. กระบวนการกระจายการร้องขอ เมื่อมีการร้องขอจากไคลเอนท์เพื่อใช้งานเว็บเซอร์วิซของระบบใหม่ผ่านทางพรอกซี เมื่อพรอกซีส่งการร้องขอดังกล่าวให้ตัวกลาง ตัวกลางจะเรียกใช้ดับบลิวเอสมัลติคัสต์เพื่อกระจายการร้องขอนั้นไปยังเว็บเซอร์วิซของระบบใหม่และเว็บเซอร์วิซของระบบเลกาซีที่เทียบเคียงตามที่ระบุไว้ในนโยบาย เมื่อระบบเลกาซีและระบบใหม่ตอบกลับมายังตัวกลาง ตัวกลางจะบันทึกแหล่งและคำตอบนั้นไว้และนำเอาผลลัพธ์ของแต่ละระบบที่ตอบกลับมาเข้าขั้นตอนการลงคะแนนเพื่อหาผลลัพธ์ที่น่าเชื่อถือมากที่สุดต่อไป

5. กระบวนการเปรียบเทียบสถานะฐานข้อมูล ในการเปรียบเทียบสถานะฐานข้อมูลได้นั้น จำเป็นต้องทำให้ระบบจัดการฐานข้อมูลของระบบเลกาซีและระบบใหม่สามารถบันทึกคำสั่งจัดการฐานข้อมูลต่างๆลงในฐานข้อมูลในรูปของตารางติดตาม (trace table) ซึ่งทำให้ตัวกลางสามารถตรวจสอบคำสั่งจัดการฐานข้อมูลได้ในภายหลัง อย่างไรก็ตามฐานข้อมูลที่ระบบเลกาซีและระบบใหม่สามารถใช้ได้นั้นมีอยู่ด้วยกันหลายชนิดทำให้ไม่สามารถออกแบบส่วนการต่อประสานที่เหมาะสมเพื่อใช้ในการเข้าถึงได้ ดังนั้นจึงจำเป็นต้องมีกระบวนการที่สามารถติดต่อกับฐานข้อมูลได้ทุกรูปแบบ โดยงานวิจัยนี้ได้นำเทคโนโลยีโอเลดีบี (OLE-DB) มาประยุกต์ใช้ ให้สามารถเชื่อมต่อไปยังฐานข้อมูลของระบบเลกาซีและระบบใหม่ได้ จากนั้นส่วนการเปรียบเทียบสถานะฐานข้อมูลจะค้นหาคำสั่งจัดการฐานข้อมูลจากตารางติดตามของระบบเลกาซีและระบบใหม่เมื่อได้คำสั่งจัดการฐานข้อมูลของระบบเลกาซีและระบบใหม่แล้ว ส่วนการเปรียบเทียบสถานะฐานข้อมูลจะแปลงคำสั่งจัดการฐานข้อมูลดังกล่าวให้เป็นคำค้นข้อมูลที่เหมาะสมเพื่อเข้าไปตรวจสอบสถานะฐานข้อมูลภายในฐานข้อมูลของทั้งระบบเลกาซีและระบบใหม่ เมื่อได้รับสถานะฐานข้อมูลของระบบเลกาซีและระบบใหม่แล้ว ตัวกลางจะนำสถานะฐานข้อมูลดังกล่าวเข้าสู่กระบวนการคัดเลือกผลลัพธ์ต่อไป แต่มีบางกรณีที่ระบบจัดการฐานข้อมูลที่ระบบเลกาซีใช้นั้นไม่มีฟังก์ชันในการสร้างตารางติดตาม ผู้วิจัยจึงนำวิธีตรวจสอบสถานะฐานข้อมูลด้วยการห่อฐานข้อมูล (database wrapping) เข้ามาประยุกต์ใช้ร่วม ซึ่งเป็นการนำข้อมูลจากตารางภายในฐานข้อมูลของระบบเลกาซีและระบบใหม่มาแปลงให้อยู่ในโครงสร้างเอกซ์เอ็มแอล ซึ่งตัวกลางจะนำสถานะฐานข้อมูลดังกล่าวเข้าสู่กระบวนการคัดเลือกผลลัพธ์ เพื่อค้นหาความแตกต่างระหว่างสถานะฐานข้อมูลของทั้งสองระบบต่อไป

6. <u>กระบวนการคัดเลือกผลลัพธ์</u> การย้ายระบบโดยอาศัยหลักการของแอคทีฟเรพลิเคชันนั้น ทั้งระบบเลกาซีและระบบใหม่จะทำงานด้วยลำดับคำสั่งเดียวกัน เมื่อระบบเลกาซีและระบบใหม่ประมวลผลเสร็จจะส่งผลลัพธ์กลับมายังตัวกลาง ดังนั้นจึงจำเป็นต้องใช้กระบวนการลงคะแนนเพื่อเลือกผลลัพธ์ที่น่าเชื่อถือมากที่สุด เนื่องจากงานวิจัยนี้ใช้เสถียรภาพของระบบเลกาซีเพื่อมาสนับสนุนการย้ายระบบซึ่งโดยปกติจะกำหนดให้ระบบเลกาซีมีน้ำหนักความน่าเชื่อถือของผลลัพธ์มากกว่าระบบใหม่ แต่บางกรณีที่จำเป็นต้องปรับเปลี่ยนน้ำหนักของแต่ละระบบและแนวทางการคัดเลือกผลลัพธ์ให้เหมาะสมกับสถานการณ์ เช่น อาจมีรีเอมพลิเมนท์เซอร์วิซของระบบเลกาซีหลายตัว ทำให้ต้องมีกระบวนการที่สามารถปรับเปลี่ยนวิธีการลงคะแนนที่คล่องตัว ซึ่งผู้วิจัยได้นำเอาแนวทางการกำหนดนโยบายมาปรับใช้ซึ่งเป็นเครื่องมือที่ผู้ติดตั้งระบบใช้สื่อสารกับตัวกลางเพื่อให้ตัวกลางทราบถึงแนวทางการคัดเลือกผลลัพธ์ โดยการกำหนดวิธีการเลือกผลลัพธ์นั้นผู้ติดตั้งระบบสามารถกำหนดผ่านส่วนต่อประสานผู้ใช้ของตัวกลางด้วยภาษาเอ็กซ์เอ็มแอล (XML : extensible markup language)

จากกระบวนการข้างต้นสามารถนำมาสรุปเป็นโมเดลเพื่อสนับสนุนการย้ายระบบได้ ดังรูปที่ 3.1



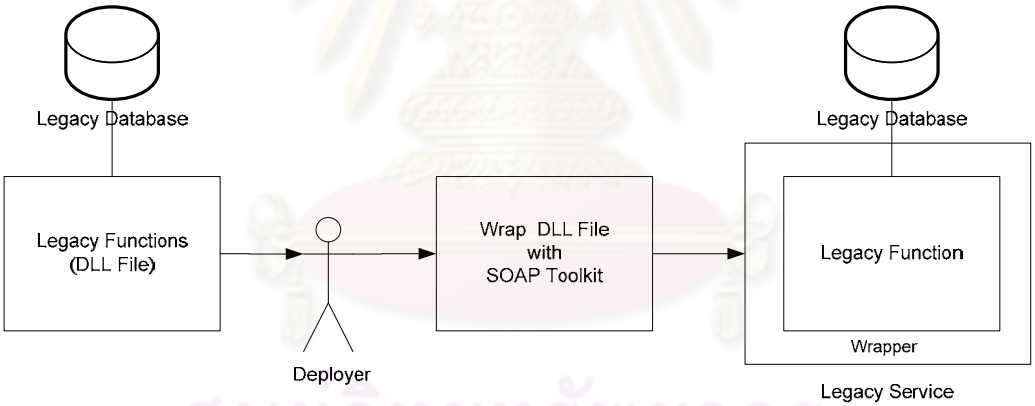
รูป 3.1 โมเดลการย้ายระบบ

3.2 ส่วนประกอบโมเดลสำหรับการย้ายระบบเลกาซี

จากรูป 3.1 แสดงให้เห็นถึงส่วนประกอบของโมเดล ซึ่งมีการแบ่งส่วนประกอบของโมเดลออกเป็นสามส่วนหลัก คือ

3.2.1 **เว็บเซอร์วิซซ้ำซ้อน** (Redundant services) ประกอบด้วยเว็บเซอร์วิซของระบบใหม่และเว็บเซอร์วิซของระบบเลกาซีที่มีการทำงานเทียบเคียงกัน โดยระบบเผื่อเสีย แบ่งออกเป็นสองส่วน ดังนี้

3.2.1.1 **เลกาซีเซอร์วิซ** (Legacy Service) ส่วนของคลังโปรแกรมหรือฟังก์ชันการทำงานของระบบเลกาซีที่ผ่านการห่อด้วยชุดเครื่องมือโซป เพื่อให้ฟังก์ชันของระบบเลกาซีดังกล่าวสามารถทำงานอยู่บนมาตรฐานของเว็บเซอร์วิซได้ โดยการห่อนั้นทำได้โดยการนำคลังโปรแกรมซึ่งบรรจุฟังก์ชันของระบบเลกาซีมาห่อด้วยชุดเครื่องมือโซป ซึ่งผู้วิจัยได้แสดงขั้นตอนการห่อระบบเลกาซีไว้ดังรูปที่ 3.2 โดยมีรายละเอียด ดังนี้



รูป 3.2 ขั้นตอนการห่อระบบเลกาซี

กระบวนการห่อทำให้ฟังก์ชันของระบบเลกาซีสามารถทำงานอยู่บนมาตรฐานของเว็บเซอร์วิซโดยอาศัยชุดเครื่องมือโซปเป็นตัวกลาง โดยผู้วิจัยได้แสดงการเรียกใช้งานฟังก์ชันของระบบเลกาซีผ่านชุดเครื่องมือโซป ไว้ดังรูปที่ 3.3 โดยมีรายละเอียด ดังนี้

1. เมื่อไคลเอนท์เรียกใช้งานฟังก์ชันของระบบเลกาซี ไคลเอนท์จะส่งการร้องขอดังกล่าวมายังลิสเซินเนอร์

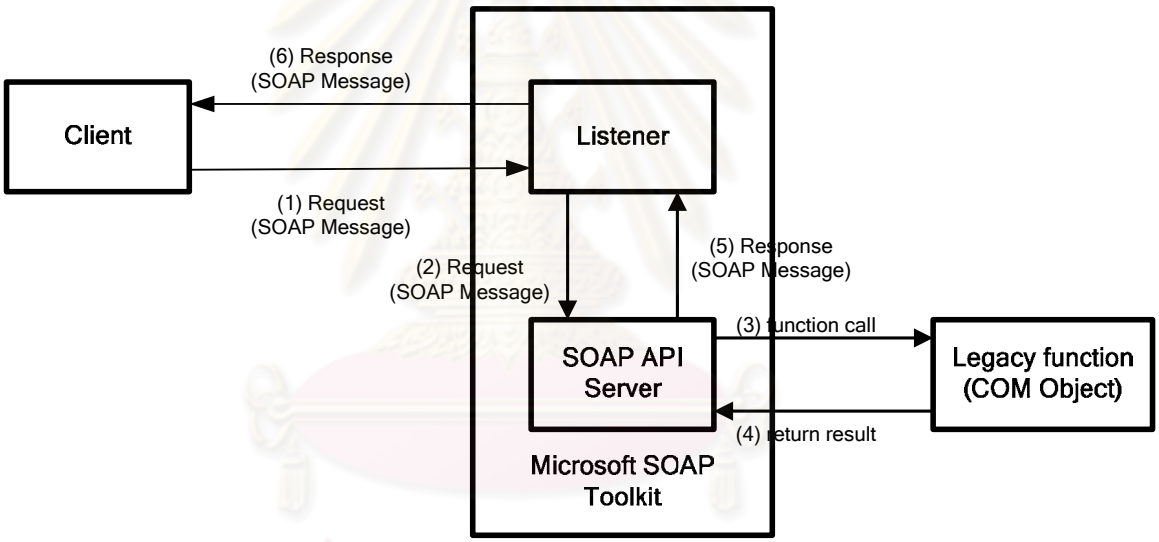
2. ลิสเซินเนอร์จะนำการร้องขอดังกล่าวส่งให้กับส่วนต่อประสานโปรแกรมโซปเซิฟเวอร์ที่เชื่อมต่อกับฟังก์ชันของระบบเลกาซีที่ไคลเอนท์ต้องการเรียกใช้

3. ส่วนต่อประสานโปรแกรมโซปเซิฟเวอร์จะเปลี่ยนการร้องขอดังกล่าวให้อยู่ในรูปแบบที่ฟังก์ชันของระบบเลกาซีเข้าใจ

4. เมื่อฟังก์ชันของระบบเลกาซีประมวลผลเสร็จส่งผลลัพธ์กลับมา

5. ส่วนต่อประสานโปรแกรมโซปเซิฟเวอร์ซึ่งเชื่อมต่อกับฟังก์ชันของระบบเลกาซีจะนำผลลัพธ์ดังกล่าวส่งให้กับลิสเซินเนอร์ในรูปของข้อความสนองโซป (SOAP response message)

6. ลิสเซินเนอร์นำข้อความสนองโซป ส่งให้กับไคลเอนท์

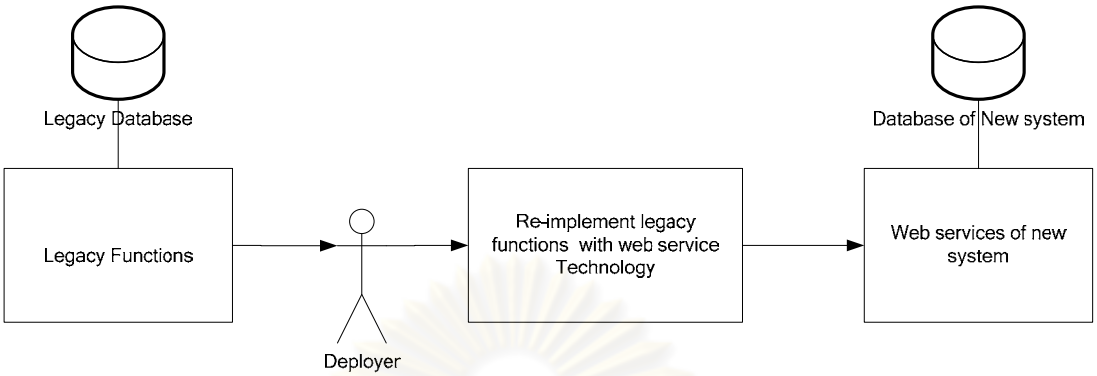


รูป 3.3 การเรียกใช้งานฟังก์ชันของเลกาซีที่ผ่านการห่อด้วยชุดเครื่องมือโซป

3.2.1.2 **รีเอมพลิเมนท์เซอร์วิซ** (re-implemented service) เป็นส่วนของเว็บเซอร์วิซซึ่งมีการทำงานที่เทียบเคียงกับฟังก์ชันของระบบเลกาซี ด้วยการนำฟังก์ชันของระบบเลกาซีมาสร้างเลียนแบบด้วยเทคโนโลยีเว็บเซอร์วิซโดยให้หนึ่งฟังก์ชันของระบบเลกาซีแทนด้วยหนึ่งเว็บเซอร์วิซของระบบใหม่ ซึ่งผู้วิจัยได้แสดงขั้นตอนการรีเอมพลิเมนท์ฟังก์ชันของระบบเลกาซี ไว้ดังรูปที่ 3.4 โดยมีรายละเอียด ดังนี้

1. ผู้ติดตั้งนำรหัสต้นฉบับในส่วนฟังก์ชันของระบบเลกาซีมาสร้างเลียนแบบด้วยเทคโนโลยีเว็บเซอร์วิซ

2. เมื่อผ่านกระบวนการรีเอมพลิเมนท์ จะได้เว็บเซอร์วิซซึ่งให้ผลลัพธ์การทำงานที่คาดว่าจะเป็นเช่นเดียวกับฟังก์ชันของระบบเลกาซี



รูป 3.4 การรีเอมพลิเมนท์ฟังก์ชันของระบบเลกาซีด้วยเทคโนโลยีเว็บเซอร์วิซ

3.2.2 **เซอร์วิซตัวกลาง** (service middleware) ประกอบไปด้วยส่วนต่างๆ คือ

3.2.2.1 **การลงทะเบียน** (registration) ในการตรวจสอบผลลัพธ์การทำงานของระบบเลกาซีและระบบใหม่ได้นั้น จำเป็นต้องทราบถึงข้อมูลรายละเอียดต่างๆ ทั้งของระบบเลกาซีและระบบใหม่ ดังนั้นผู้วิจัยจึงได้นำเอาแนวคิดของ ดับบลิวเอสเดปพลอยเยอร์ (WS-Deployer) ในงานวิจัยของซาลาส [6] มาประยุกต์ใช้ โดยทั้งระบบเลกาซีและระบบใหม่ต้องให้ข้อมูลที่จำเป็นผ่านการลงทะเบียน ดังนี้

ตาราง 3.1 ข้อมูลที่ต้องใช้ในการลงทะเบียนระบบ

ชื่อข้อมูล	คำอธิบาย
ข้อมูลของเว็บเซอร์วิซในระบบ	ชื่อและตำแหน่งของเว็บเซอร์วิซและข้อมูลที่จำเป็นในการร้องขอ
ข้อมูลของฐานข้อมูลในระบบ	ข้อมูลที่จำเป็นในการเข้าถึงและวิธีการตรวจสอบสถานะฐานข้อมูล

3.2.2.2 **พรอกซี** (proxy) พรอกซีนั้นถูกสร้างขึ้นเมื่อการลงทะเบียนกับตัวกลางเสร็จสมบูรณ์ โดยพรอกซีจะถูกสร้างตามจำนวนรีเอมพลิเมนท์เซอร์วิซที่มาลงทะเบียนในกรณีที่รีเอมพลิเมนท์เซอร์วิซดังกล่าวยังไม่มีพรอกซีรองรับ โดยพรอกซีจะทำหน้าที่เสมือนเว็บเซอร์วิซจำลองซึ่งมีวิธีการร้องขอเหมือนเว็บเซอร์วิซที่มาลงทะเบียนทุกประการ แต่หน้าที่การทำงานภายในของพรอกซีจะเป็นการรับการร้องขอนั้นเข้าสู่ตัวกลาง ผู้วิจัยยึดแนวคิดการสร้างพรอกซีตามงานวิจัยของซาลาส [6] การนำพรอกซีมาใช้งานนั้นไคลเอนท์ต้องเปลี่ยนเส้นทางการร้องขอจากเครื่องบริการเว็บเซอร์วิซไว้มาเป็นเครื่องให้บริการพรอกซี ดังนี้

การเรียกใช้งานเว็บเซอร์วิซจากเครื่องบริการเซอร์วิซ	การเรียกใช้งานเว็บเซอร์วิซจากเครื่องบริการพรอกซี
Http:// [Service Server]/[Service]	Http:// [Proxy Server]/[Service]

โดยที่

[Service Server] หมายถึง ที่อยู่เครื่องบริการเว็บเซอร์วิซของระบบใหม่

[Proxy Server] หมายถึง ที่อยู่เครื่องบริการเว็บพรอกซี

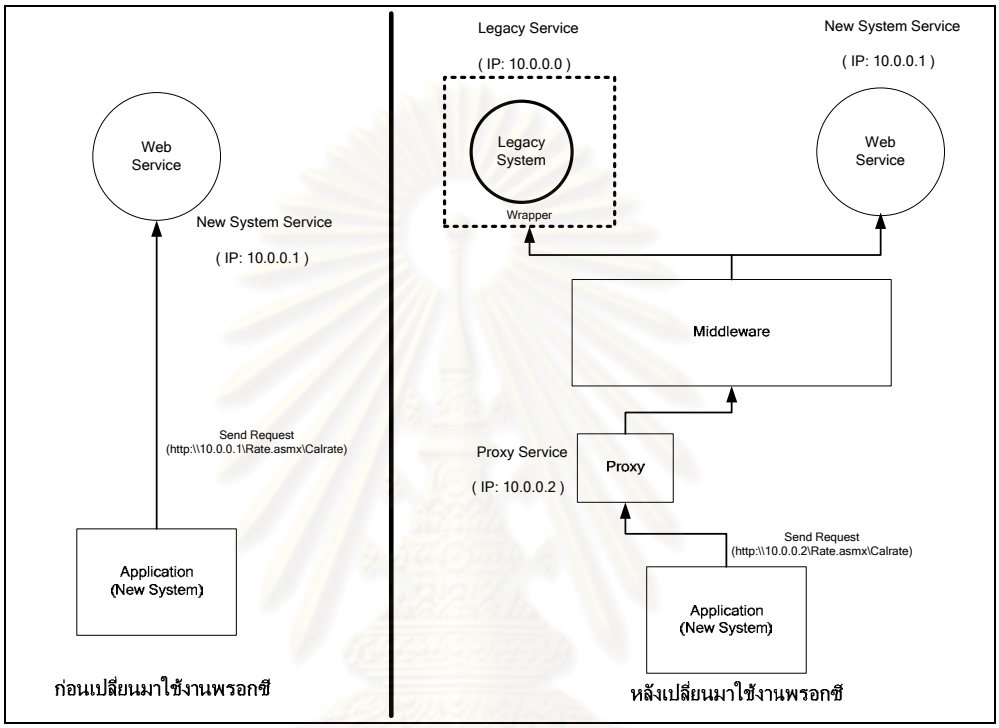
[Service] หมายถึง เว็บเซอร์วิซที่ต้องการร้องขอ

ผู้วิจัยได้แสดงตัวอย่างการเปลี่ยนไปใช้งานพรอกซีไว้ดังนี้ คือ ผู้วิจัยได้สมมุติเหตุการณ์ว่าองค์กรได้พัฒนาระบบใหม่ขึ้นเพื่อใช้ทดแทนระบบเลกาซีเดิมซึ่งเป็นระบบที่ใช้คำนวณอัตราแลกเปลี่ยนเงินตราต่างประเทศ โดยให้ระบบใหม่ประกอบด้วยเว็บเซอร์วิซชื่อ *Rate* ภายในมีเว็บเมท็อดชื่อ *Calrate* ใช้สำหรับคำนวณอัตราแลกเปลี่ยนเงินตราต่างประเทศ ซึ่งมีที่อยู่สำหรับเรียกใช้งานเว็บเซอร์วิซดังกล่าวที่ *http://10.0.0.1/Rate.asmx/calrate* โดยได้พัฒนาโปรแกรมประยุกต์สำหรับควบคุมการเรียกใช้งานเว็บเซอร์วิซดังกล่าว ซึ่งหากเป็นการใช้งานปกติ โปรแกรมประยุกต์จะเรียกใช้งานเว็บเซอร์วิซโดยตรงที่ *http://10.0.0.1/Rate.asmx/calrate* แต่หากต้องการให้ระบบเลกาซีช่วยยืนยันคำตอบ จึงจำเป็นต้องทำให้ระบบเลกาซีได้รับการร้องขอดังกล่าวด้วย ด้วยเหตุนี้จึงต้องอาศัยตัวกลางเพื่อรับการการร้องขอ มากระจายส่งให้กับเว็บเซอร์วิซของระบบเลกาซีและระบบใหม่ อย่างไรก็ตามในการรับการร้องขอนั้น ตัวกลางต้องมีส่วนสำหรับรับการร้องขอจากโปรแกรมประยุกต์เข้าสู่ตัวกลาง ซึ่งส่วนดังกล่าวเรียกว่าพรอกซี โดยขั้นตอนการใช้งานพรอกซีมีดังนี้

1. ผู้ติดตั้งระบบนำเว็บเซอร์วิซ *Rate* ของระบบใหม่มาลงทะเบียน
2. ตัวกลางจะสร้างเว็บเซอร์วิซชื่อ *Rate* สำหรับทำหน้าที่เป็นพรอกซี ซึ่งภายในเว็บเซอร์วิซดังกล่าวมีเว็บเมท็อดชื่อ *Calrate* โดยมีที่อยู่ในการเรียกใช้งานที่ *http://10.0.0.2/Rate.asmx/calrate* อย่างไรก็ตามเว็บเมท็อด *Calrate* ของพรอกซีนั้นไม่ได้ทำหน้าที่คำนวณอัตราแลกเปลี่ยนเงินตราต่างประเทศเหมือนกับเว็บเมท็อด *Calrate* ของระบบใหม่ แต่ทำหน้าที่ในการรับการร้องขอเข้าสู่ตัวกลาง
3. ผู้ติดตั้งระบบเปลี่ยนที่อยู่การเรียกใช้งานเว็บเซอร์วิซจากเดิม *http://10.0.0.1/Rate.asmx/calrate* มาเป็น *http://10.0.0.2/Rate.asmx/calrate*

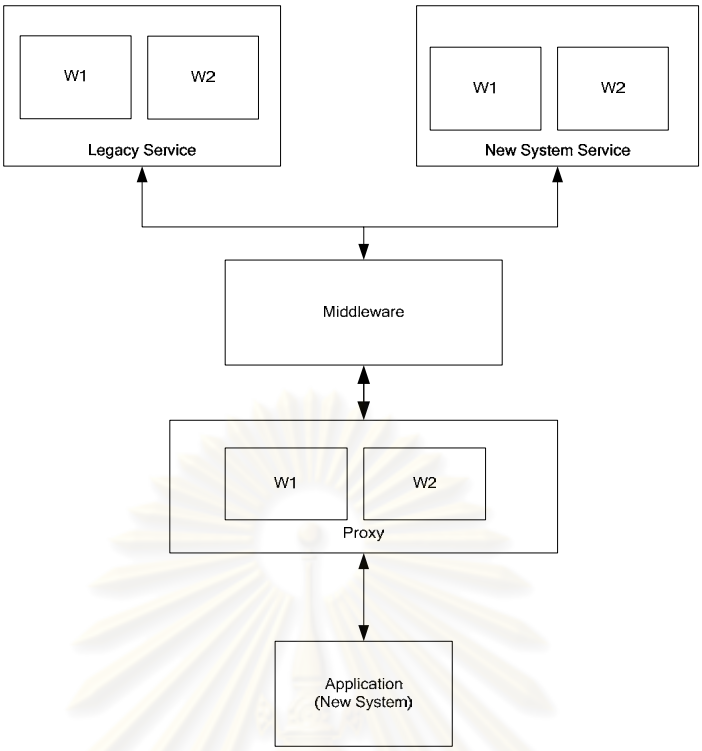
4. เมื่อเมื่อมีการใช้งานโปรแกรมประยุกต์ การร้องขอดังกล่าวจะถูกส่งไปยังเว็บพรอกซี *Rate* แทนการส่งไปยังเว็บเซอร์วิซ *Rate* ของระบบใหม่ ดังรูปที่ 3.5

5. พรอกซีนำการร้องขอดังกล่าวเข้าสู่ตัวกลางเพื่อนำไปกระจายส่งยังเว็บเซอร์วิซของระบบเลกาซีและระบบใหม่



รูป 3.5 เส้นทางการส่งข้อความร้องขอเมื่อเปลี่ยนไปใช้งานพรอกซี

โดยการสร้างพรอกซีนั้น หากเว็บเซอร์วิซที่นำมาลงทะเบียนกับตัวกลางมีพรอกซีรองรับอยู่แล้ว ตัวกลางก็จะไม่สร้างพรอกซีซ้ำซ้อนขึ้นอีก โดยผู้วิจัยได้สมมุติเหตุการณ์ว่าระบบเลกาซีและระบบใหม่ โดยระบบเลกาซีประกอบด้วยเว็บเซอร์วิซชื่อ W1 และ W2 ส่วนระบบใหม่ประกอบด้วยเว็บเซอร์วิซชื่อ W1 W2 เช่นเดียวกับระบบเลกาซี เนื่องจากต้องการให้มีการทำงานที่เหมือนกับระบบเลกาซีเดิม โดยในการสร้างพรอกซีนั้น เมื่อนำเว็บเซอร์วิซชื่อ W1และ W2 ของระบบใหม่มาลงทะเบียนกับตัวกลาง ตัวกลางจะสร้างพรอกซีชื่อ W1และ W2 ขึ้น เมื่อนำเว็บเซอร์วิซ W1 และ W2 ของระบบเลกาซีมาลงทะเบียน ตัวกลางจะไม่สร้างพรอกซีชื่อ W1 และ W2 ขึ้นอีก เนื่องจากตัวกลางมีพรอกซีสำหรับเว็บเซอร์วิซดังกล่าวรองรับแล้ว ดังรูปที่ 3.6



รูป 3.6 การรับการร้องขอเข้าสู่ตัวกลางด้วยพรอกซี

3.2.2.3 **ส่วนนโยบาย** (Policy Module) ส่วนของการกำหนดแนวทางสำหรับการคัดเลือกผลลัพธ์ โดยส่วนนโยบายนั้นยึดแนวคิดการสร้างนโยบายมาจากงานวิจัยของ ดอบสัน [3] โดยสิ่งที่กำหนดไว้ในนโยบายประกอบด้วยสี่ส่วน ดังนี้

1. ส่วนการกำหนดเว็บเซอร์วิซที่นำมาเปรียบเทียบ ใช้ระบุเว็บเซอร์วิซที่นำมาเปรียบเทียบเพื่อหาผลลัพธ์รวมทั้งการกำหนดน้ำหนักให้กับแต่ละเว็บเซอร์วิซ โดยเว็บเซอร์วิซดังกล่าวต้องเป็นเว็บเซอร์วิซที่ได้ลงทะเบียนกับเซอร์วิซตัวกลางแล้ว ซึ่งการกำหนดเว็บเซอร์วิซที่นำมาเปรียบเทียบนั้นมีรูปแบบไวยากรณ์ดังนี้

```
<Connection>
    <Operation>
        < [SystemName]. [Service Name]. [Method]  Weight = [Weight Values]>
    </Operation>
</Connection>
```

โดยที่

ตัวหนาเป็น tag และ option name

[System Name] หมายถึง ชื่อระบบ

[Service Name] หมายถึง ชื่อเว็บเซอร์วิซของระบบ

[Method] หมายถึง ฟังก์ชันการทำงานภายในเซอร์วิซ

[Weight Value] หมายถึง น้ำหนักที่กำหนดให้กับฟังก์ชันการทำงาน

ตัวอย่างเช่น

<Connection>
<Operation>
<System2000.temp.temperature_TH **Weight**=2>
</Operation>
<Operation>
<System2008.temperature.temperature_Thailand **Weight**=1>
</Operation>
</Connection>

จากตัวอย่างนี้ ผู้วิจัยต้องการหาอุณหภูมิของประเทศไทยโดยนำผลลัพธ์จากเว็บเซอร์วิซของระบบเลกาซีและผลลัพธ์จากเว็บเซอร์วิซของระบบใหม่มาเปรียบเทียบกัน ซึ่งก่อนการเปรียบเทียบนั้นผู้วิจัยได้ลงทะเบียนระบบเลกาซีโดยใช้ชื่อ *System2000* และลงทะเบียนระบบใหม่ด้วยชื่อ *System2008* รวมทั้งลงทะเบียนเว็บเซอร์วิซของแต่ละระบบไว้กับเซอร์วิซตัวกลาง โดยในตัวอย่างผู้วิจัยนำเมท็อด (method) ชื่อ *temperature_TH* จากเว็บเซอร์วิซชื่อ *temp* ของระบบเลกาซี ซึ่งกำหนดน้ำหนักความ น่าเชื่อถือเท่ากับ 2 มาเปรียบเทียบกับเมท็อดชื่อ *temperature_Thailand* จากเว็บเซอร์วิซชื่อ temperature ของระบบใหม่ ซึ่งกำหนดน้ำหนักความน่าเชื่อถือเท่ากับ 1

2. <u>ความต้องการจำนวนผู้ร่วมลงคะแนน</u> ใช้ระบุจำนวนคำตอบที่ต้องการก่อนการคัดเลือกผลลัพธ์ โดยมีรูปแบบไวยากรณ์ ดังนี้

<Cardinality> *[Number of Response]* **</Cardinality>**

โดยที่

ตัวหนาเป็น tag และ option name

[Number of Response] หมายถึง จำนวนคำตอบที่ต้องการ

ตัวอย่างเช่น

<Cardinality>2</Cardinality>

จากตัวอย่างข้างต้น ผู้วิจัยต้องการคำตอบจำนวน 2 คำตอบในการคัดเลือกผลลัพธ์ ดังนั้นเมื่อกำหนดเป็น 1 เซอร์วิซตัวกลางจะส่งผลลัพธ์ที่ตอบกลับมาเร็วที่สุดส่งให้กับไคลเอนท์

3. ส่วนระยะเวลาการรอคอย (Timeout) ใช้กำหนดระยะเวลาการรอคอย การตอบกลับ ในกรณีที่จำนวนคำตอบยังไม่เท่ากับจำนวนคำตอบที่กำหนดไว้ในส่วนความต้องการจำนวนผู้ร่วมลงคะแนน โดยมีรูปแบบไวยากรณ์ ดังนี้

<Response_waiting_time> *[Seconds]* </ Response_waiting_time >

โดยที่

ตัวหนาเป็น tag และ option name

[Seconds] หมายถึง จำนวนเวลาที่ใช้ในการรอคอยการตอบกลับ มีหน่วยเป็นวินาที

ตัวอย่างเช่น

<Response_waiting_time>30</Response_waiting_time>

จากตัวอย่างข้างต้น ผู้วิจัยได้กำหนดให้เซอร์วิซตัวกลางรอคำตอบเป็นเวลา 30 วินาที ในกรณีที่คำตอบที่ตอบกลับมานั้นยังไม่เท่ากับที่กำหนดไว้ในส่วนความต้องการจำนวนผู้ร่วมลงคะแนน เมื่อครบ 30 วินาทีและจำนวนคำตอบที่ตอบกลับมานั้นยังไม่เท่ากับที่กำหนดไว้ในส่วนความต้องการจำนวนผู้ร่วมลงคะแนน เซอร์วิซตัวกลางจะทำในส่วนยกเว้นซึ่งจะกล่าวในหัวข้อถัดไป

4. ส่วนยกเว้น ใช้กำหนดการกระทำ เมื่อครบกำหนดระยะเวลารอคอยและคำตอบที่ตอบกลับมานั้นยังไม่เท่ากับที่กำหนดไว้ในส่วนความต้องการจำนวนผู้ร่วมลงคะแนน โดยมีไวยากรณ์ ดังนี้

<Exception> *[Action]* </Exception>

โดยที่

ตัวหนาเป็น tag และ option name

[Action] คือ การกระทำที่ต้องการให้ปฏิบัติเพื่อให้สามารถสรุปผลลัพธ์ได้ โดยมีสองการกระทำ คือ

Error: ส่งข้อผิดพลาดไปยังไคลเอนท์

Ignore_Cardinality: ส่งคำตอบที่มีน้ำหนักมากที่สุดที่เซอร์วิซตัวกลางมีอยู่ไปยังไคลเอนท์ ในกรณีที่เซอร์วิซตัวกลางไม่มีคำตอบตัวกลางจะส่งข้อผิดพลาดไปยังไคลเอนท์

ตัวอย่างเช่น

<Exception> *Ignore_Cardinality*</Exception>

จากตัวอย่างข้างต้น ในกรณีที่กำหนดส่วนยกเว้นเป็น *Ignore_Cardinality* เมื่อครบกำหนดระยะเวลาการรอคอยและจำนวนคำตอบที่ตอบกลับมานั้นยังไม่เท่ากับที่กำหนดไว้ในส่วนความต้องการจำนวนผู้ร่วมลงคะแนน เซอร์วิซตัวกลางจะเลือกคำตอบที่มีน้ำหนักมากที่สุดจากคำตอบทั้งหมดที่เซอร์วิซตัวกลางได้รับในขณะนั้น หากเซอร์วิซตัวกลางไม่มีคำตอบ เซอร์วิซตัวกลางจะส่งข้อผิดพลาดให้กับไคลเอนท์ ในกรณีที่กำหนดส่วนยกเว้นเป็น Error เซอร์วิซตัวกลางจะส่งข้อผิดพลาดไปยังไคลเอนท์ทันที โดยไม่สนใจคำตอบของระบบใดๆ ที่เซอร์วิซตัวกลางมีอยู่

จากส่วนต่างๆของแบบนโยบายที่กล่าวมาข้างต้นนั้น สามารถนำมารวมเป็นนโยบายสำหรับย้ายระบบ ซึ่งผู้วิจัยได้นำเสนอตัวอย่างนโยบายการย้ายระบบสำหรับระบบค้นหาอุณหภูมิไว้ ดังรูปที่ 3.7 โดยภายในนโยบายนั้น ผู้วิจัยได้กำหนดให้มีการเปรียบเทียบฟังก์ชันของระบบเลกาซีและระบบใหม่สองฟังก์ชันด้วยกันนั่นคือ การเปรียบเทียบฟังก์ชันการหาอุณหภูมิประเทศไทยระหว่างระบบเลกาซีกับระบบใหม่ และการเปรียบเทียบฟังก์ชันการหาอุณหภูมิประเทศสหรัฐอเมริการะหว่างระบบเลกาซีและระบบใหม่ โดยในส่วนของฟังก์ชันในการหาอุณหภูมิประเทศไทยของระบบเลกาซีนั้น ผู้วิจัยได้กำหนดน้ำหนักความน่าเชื่อถือให้เท่ากับ 2 ส่วนฟังก์ชันการหาอุณหภูมิประเทศไทยของระบบใหม่นั้นผู้วิจัยได้กำหนดน้ำหนักความน่าเชื่อถือให้เท่ากับ 1 โดยกำหนดความต้องการจำนวนคำตอบไว้เท่ากับ 2 และกำหนดระยะเวลาการรอคอยเมื่อจำนวนคำตอบยังไม่ครบไว้เท่ากับ 3 วินาที นอกจากนี้ยังกำหนดส่วนยกเว้นเป็น Ignore_Cardinality ในกรณีที่ตัวกลางได้รับคำตอบ 2 คำตอบภายระยะเวลา 3 วินาที ตัวกลางจะนำคำตอบดังกล่าวไปคัดเลือกผลลัพธ์ ซึ่งผู้วิจัยได้กล่าวถึงวิธีการคัดเลือกผลลัพธ์ไว้ในหัวข้อ

3.2.2.6 การลงคะแนนเสียง ในกรณีที่จำนวนคำตอบน้อยกว่า 2 คำตอบ เซอร์วิซตัวกลางจะรอจนกว่าจะได้รับคำตอบครบ 2 คำตอบภายในระยะเวลา 3 วินาที หากครบ 3 วินาทีและเซอร์วิซตัวกลางยังไม่ได้รับคำตอบ เซอร์วิซ ตัวกลางจะนำคำตอบที่มีอยู่ส่งให้กับไคลเอนท์โดยเลือกคำตอบที่มีน้ำหนักมากที่สุด หากเซอร์วิซตัวกลางไม่มีคำตอบอยู่เซอร์วิซตัวกลางจะส่งข้อผิดพลาดให้กับไคลเอนท์ ในกรณีที่เซอร์วิซตัวกลางได้รับคำตอบครบ 2 คำตอบภายในเวลา 3 วินาที เซอร์วิซตัวกลางจะส่งผลลัพธ์ของฟังก์ชันการหาอุณหภูมิประเทศไทยของระบบเลกาซีซึ่งมีน้ำหนักมากที่สุดให้กับไคลเอนท์

หากฟังก์ชันการหาอุณหภูมิประเทศไทยของระบบใหม่ให้ผลลัพธ์ที่แตกต่างจากระบบเลกาซี เซอร์วิซตัวกลางจะถือว่าฟังก์ชันการหาอุณหภูมิประเทศไทยของระบบใหม่ผิดปกติและรายงานไปให้กับผู้ดูแลระบบได้ทราบ

ส่วนการเปรียบเทียบฟังก์ชันการหาอุณหภูมิประเทศสหรัฐอเมริกานั้น ผู้วิจัยได้กำหนดน้ำหนักความน่าเชื่อถือให้กับฟังก์ชันการหาอุณหภูมิ ประเทศสหรัฐอเมริกาของระบบเลกาซีเท่ากับ 2 และกำหนดน้ำหนักความน่าเชื่อถือให้กับฟังก์ชันในการหาอุณหภูมิประเทศสหรัฐอเมริกาของระบบใหม่ให้เท่ากับ 1 โดยกำหนดความต้องการจำนวนคำตอบเท่ากับ 1 และกำหนดระยะเวลาการรอคอยเมื่อจำนวนคำตอบยังไม่ครบไว้เท่ากับ 2 วินาที และกำหนดส่วนยกเว้นเป็น error ในกรณีที่ตัวกลางได้รับคำตอบ 1 คำตอบภายในระยะเวลา 2 วินาที ตัวกลางจะส่งคำตอบดังกล่าวให้กับไคลเอนท์ทันที ในกรณีที่เซอร์วิซตัวกลางยังไม่ได้รับคำตอบ เซอร์วิซตัวกลางจะรอคอยภายในระยะเวลา 2 วินาที หากครบ 2 วินาทีและเซอร์วิซตัวกลางยังไม่ได้รับคำตอบ เซอร์วิซตัวกลางจะส่งข้อผิดพลาดให้กับไคลเอนท์

จากนโยบายที่ได้ยกตัวอย่างไปนั้นเห็นได้ว่า ผู้ติดตั้งระบบสามารถปรับเปลี่ยนพฤติกรรมการคัดเลือกผลลัพธ์ของโมเดลให้เหมาะสมกับสถานการณ์ต่างๆ ผ่านทางนโยบายได้ โดยผู้วิจัยได้ยกตัวอย่างการกำหนดนโยบายให้เหมาะสมกับสถานการณ์ โดยได้กำหนดระบบที่นำมาเปรียบเทียบกัน มีเว็บเซอร์วิซสำหรับคำนวนเงินตราต่างประเทศชื่อ Rate ซึ่งประกอบด้วยเว็บเมท็อดชื่อ CalRate ดังตารางที่ 3.2

```
<PolicyModel >

 <Vote>
< Name >TempTH</ Name >
    <Connection>
        <operation>system2000.temp.temperature_TH weight=2</operation>
        <operation>system2008.temperature.temperature_Thailand weight=1</operation>
    </Connection>
    <Cardinality>2</Cardinality>
    <Response_waiting_time>3</Response_waiting_time>
    <Exception>Ignore_Cardinality</Exception>
 </Vote>

<Vote>
<Name>TempUS</Name>
    <Connection>
        <operation>system2000.temp.temperature_US weight=1</operation>
        <operation>system2008.temperature.temperature_USA weight=1</operation>
    </Connection>
    <Cardinality>1</Cardinality>
    <Response_waiting_time >2</Response_waiting_time>
    <Exception>Error</Exception>
 </Vote>

</PolicyModel>
```

รูป 3.7 นโยบายการคัดเลือกผลลัพธ์สำหรับระบบค้นหาอุณหภูมิ

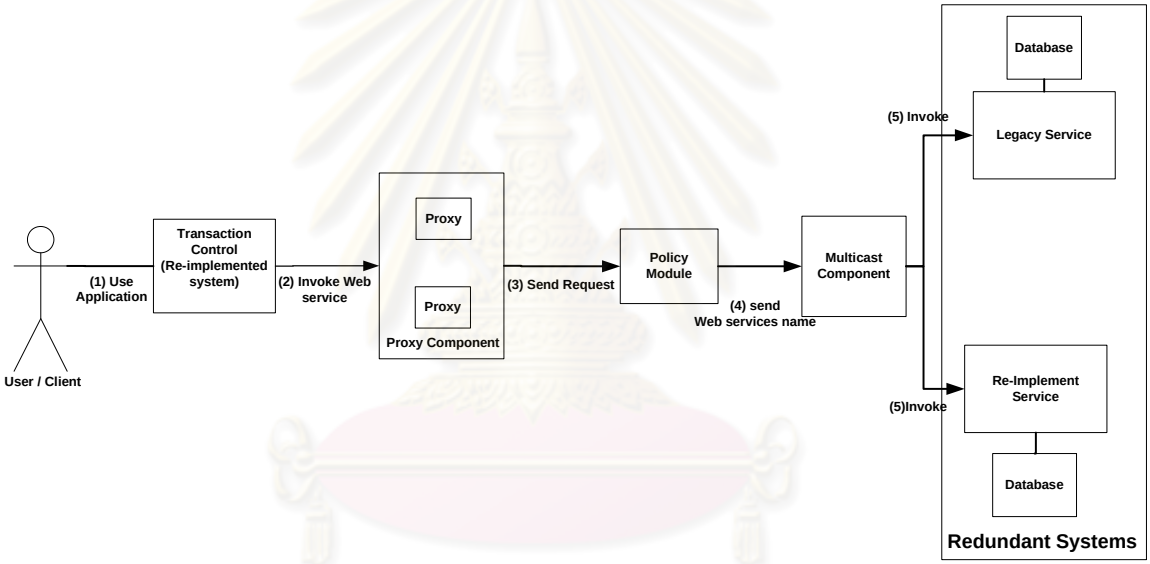
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 3.2 ตัวอย่างการกำหนดนโยบายให้เหมาะสมกับสถานการณ์

System	Service	Method	Weight	Cardinality	Response Waiting time	Exception
1.กรณีที่เชื่อถือระบบเลกาซีมากที่สุดเพียงระบบเดียว เหมาะสมกับสถานการณ์ที่ใช้งานระบบเลกาซีเป็นเวลานาน ต่อมาได้มีการรีเอมพลิเมนท์ระบบใหม่ขึ้น แต่องค์กรยังคงต้องการการทำงานของระบบเลกาซีเพื่อมาสนับสนุนการทำงานของระบบใหม่ ซึ่งหากระบบใหม่ทำงานผิดพลาดก็จะไม่ส่งผลกระทบต่อธุรกรรมขององค์กร						
Legacy	Rate	CalRate	2	2	10	Error
New system	Rate	CalRate	1			
2.กรณีที่เชื่อถือระบบเลกาซี แต่หากระบบเลกาซีตอบแตกต่างจากระบบทั้งหมดจะถือว่าระบบเลกาซีผิดปกติ เหมาะสมกับสถานการณ์ที่องค์กรมีระบบรีเอมพลิเมนท์หลายรุ่น ซึ่งทุกรุ่นได้ผ่านการทำงานคู่ขนานกับระบบเลกาซีมาแล้ว ซึ่งโดยปกติการทำงานของระบบทั้งหมดกับระบบเลกาซีต้องไม่แตกต่างกัน แต่หากสภาพแวดล้อมของระบบเลกาซีมีช่องโหว่ด้านความปลอดภัยอาจทำให้ระบบเลกาซีทำงานผิดพลาดได้ซึ่งจะส่งผลกระทบต่อธุรกรรมขององค์กร หากเชื่อถือระบบเลกาซีเพียงระบบเดียว						
Legacy	Rate	CalRate	6	3	10	Error
New system version 1	Rate	CalRate	4			
New system version 2	Rate	CalRate	2			
New system version 3	Rate	CalRate	1			
3.กรณีที่ให้ความสำคัญกับความเร็วในการตอบกลับมากกว่าความถูกต้องของผลลัพธ์ เหมาะสมกับสถานการณ์ที่องค์กรใช้งานระบบรีเอมพลิเมนท์คู่ขนานกับระบบเลกาซีมานาน จนระบบรีเอมพลิเมนท์มีการทำงานเหมือนกับระบบเลกาซี อย่างไรก็ตามฟังก์ชันที่กำหนดนั้นควรเป็นฟังก์ชันที่ไม่ส่งผลกระทบต่อองค์กร						
Legacy	Rate	CalRate	6	1	10	Ignore cardinality
New system version 1	Rate	CalRate	4			
New system version 2	Rate	CalRate	2			
New system version 3	Rate	CalRate	1			
4.กรณีที่ให้ความสำคัญกับความเร็วในการตอบกลับมากกว่าความถูกต้องของผลลัพธ์แต่ยังคงการคัดเลือกผลลัพธ์อยู่ เหมาะสมกับสถานการณ์ที่องค์กรใช้งานระบบรีเอมพลิเมนท์คู่ขนานกับระบบเลกาซีมานาน จนระบบรีเอมพลิเมนท์มีการทำงานเหมือนกับระบบเลกาซี และฟังก์ชันดังกล่าวเป็นฟังก์ชันที่ส่งผลกระทบต่อองค์กร						
Legacy	Rate	CalRate	3	2	10	Error
New system version 1	Rate	CalRate	2			
New system version 2	Rate	CalRate	1			
New system version 3	Rate	CalRate	1			
5. กรณีที่เชื่อถือระบบเลกาซีมากที่สุดเพียงระบบเดียว แต่หากระบบเลกาซีไม่สามารถตอบได้สามารถใช้คำตอบของระบบรีเอมพลิเมนท์ตอบแทนได้ เหมาะสมกับสถานการณ์ที่ใช้งานระบบเลกาซีเป็นเวลานาน ต่อมาได้มีการรีเอมพลิเมนท์ระบบใหม่ขึ้น แต่องค์กรยังต้องการการทำงานของระบบเลกาซีสนับสนุนการทำงานของระบบใหม่ซึ่งหากระบบใหม่ทำงานผิดพลาดก็จะไม่ส่งผลกระทบต่อธุรกรรมขององค์กร และกฟังก์ชันที่นำมาเปรียบเทียบนั้นไม่ใช่ฟังก์ชันที่ส่งผลกระทบต่อธุรกรรมขององค์กร เช่น การตรวจระดับผู้ใช้ ในกรณีที่ระบบเลกาซีไม่สามารถทำงานได้ ระบบยังสามารถใช้ผลลัพธ์ของระบบรีเอมพลิเมนท์ส่งให้กับไคลเอนท์เพื่อให้ธุรกรรมขององค์กรสามารถดำเนินต่อไปได้						
Legacy	Rate	CalRate	2	2	10	Ignore cardinality
New system	Rate	CalRate	1			

3.2.2.4 **ดับบลิวเอสมัลติคาสต์** (WS-multicast) ส่วนของการมัลติคาสต์การร้องขอไปยังเว็บเซอร์วิซทั้งของระบบเลกาซีและระบบใหม่ รวมทั้งรอคอยการตอบกลับ โดยผู้วิจัยได้นำเอาดับบลิวเอสมัลติคาสต์ในงานวิจัยของซาลาส [6] ในส่วนของดับบลิวเอสมัลติคาสต์มาประยุกต์ใช้ โดยการทำงานของดับบลิวเอสมัลติคาสต์ แบ่งเป็นสองส่วน ดังนี้ คือ

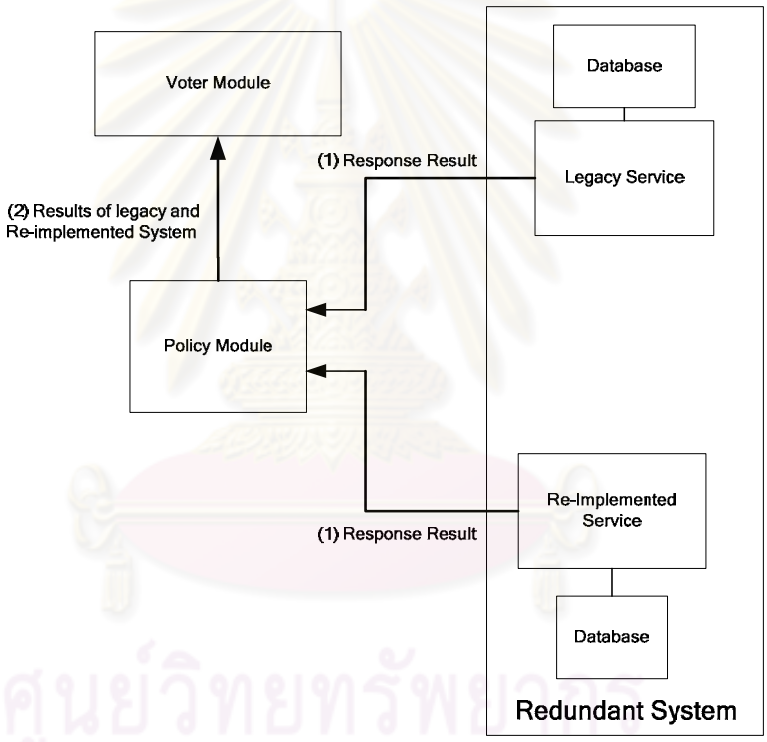
1. <u>การกระจายการร้องขอ</u> ตามรูปที่ 3.8 เมื่อไคลเอนท์ (1) ร้องขอเว็บเซอร์วิซของระบบใหม่ การร้องขอดังกล่าว (2) จะถูกส่งมายังพรอกซี พรอกซีจะนำการร้องขอนั้น (3) ส่งให้กับส่วนนโยบาย เมื่อทราบชื่อเว็บเซอร์วิซของทั้งสองระบบที่จะนำมาเปรียบเทียบกันแล้วก็จะนำชื่อดังกล่าว (4) ส่งให้กับดับบลิวเอสมัลติคาสต์ ดับบลิวเอสมัลติคาสต์จะสร้างข้อความการร้องขอ (request message) ให้กับเว็บเซอร์วิซทั้งของระบบเลกาซีและระบบใหม่และ (5) กระจายการร้องขอนั้นไปยังเว็บเซอร์วิซของทั้งสองระบบ



รูป 3.8 การกระจายการร้องขอ

2. <u>การรับการตอบกลับ</u> ตามรูปที่ 3.9 นั้น เมื่อเว็บเซอร์วิซของระบบเลกาซีและระบบใหม่ประมวลผลเสร็จจะส่งผลลัพธ์ (1) คืนกลับมาในรูปของข้อความตอบกลับ (response message) ส่วนนโยบายจะตรวจสอบข้อความตอบกลับเพื่อบันทึกแหล่งและคำตอบนั้นไว้ เมื่อจำนวนคำตอบครบตามที่ได้กำหนดไว้ในส่วนความต้องการจำนวนผู้ร่วมลงคะแนนแล้วแบบนโยบายจะตรวจสอบกับส่วนลงทะเบียนว่าระบบที่ตอบกลับมานั้นมีการใช้งานฐานข้อมูลหรือไม่ ถ้าไม่มีการใช้งานฐานข้อมูล (2) ส่วนนโยบายจะนำผลลัพธ์ที่ได้จากการตอบกลับส่งให้กับส่วนของการลงคะแนน ในกรณีที่จำนวนคำตอบน้อยกว่าที่กำหนดไว้ในส่วนความต้องการจำนวนผู้ร่วมลงคะแนนส่วนนโยบายจะรอคอยภายในระยะเวลาที่กำหนดไว้ในส่วนระยะเวลารอคอย หาก

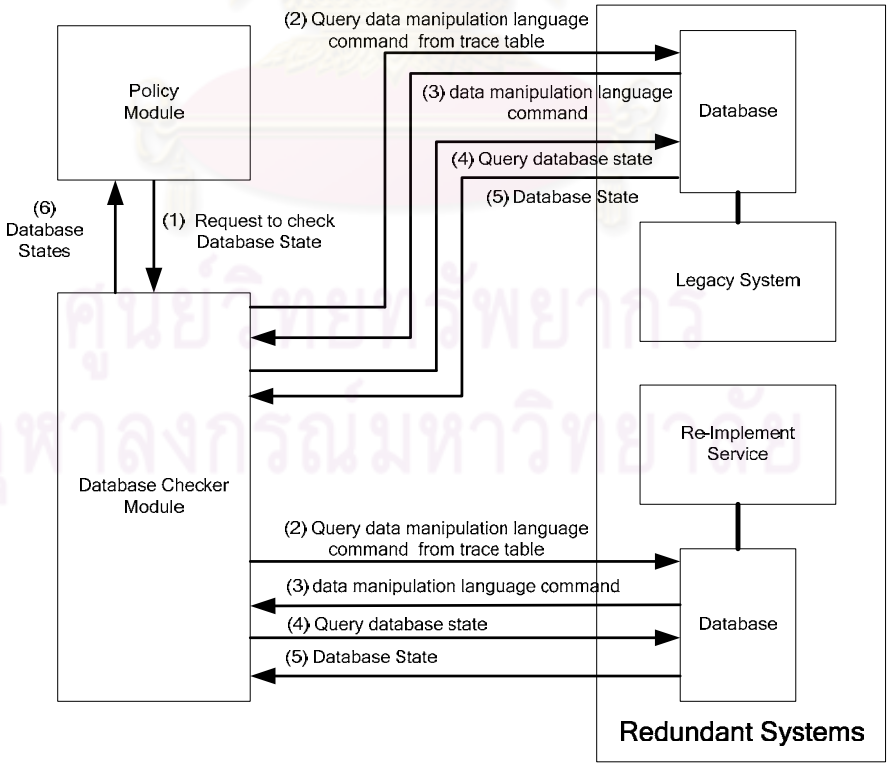
หมดระยะเวลารอคอยส่วนนโยบายจะกระทำในส่วนยกเว้นภายในนโยบาย หากส่วนยกเว้นกำหนดเป็น *Ignore_cardinality* และส่วนนโยบายมีคำตอบ ส่วนนโยบายจะนำคำตอบที่มีน้ำหนักมากที่สุดที่ส่วนนโยบายมีอยู่ส่งให้กับไคลเอนท์ผ่านทางพรอกซี หากไม่มีคำตอบส่วนนโยบายจะส่งข้อผิดพลาดไปยังไคลเอนท์ผ่านทางพรอกซี หากส่วนยกเว้นกำหนดเป็น *error* ส่วนนโยบายจะส่งข้อผิดพลาดไปไคลเอนท์ผ่านทางพรอกซีทันที แต่ในกรณีที่มีการใช้งานฐานข้อมูล ส่วนนโยบายจะเรียกใช้ส่วนตรวจสอบสถานะฐานข้อมูลเพื่อร้องขอสถานะฐานข้อมูล ซึ่งผู้วิจัยได้กล่าวถึงขั้นตอนการร้องขอสถานะฐานข้อมูลไว้ในหัวข้อ 3.2.2.5 ส่วนตรวจสอบสถานะฐานข้อมูล เมื่อได้รับสถานะฐานข้อมูลแล้ว แบบนโยบายจึงจะส่งผลลัพธ์และสถานะฐานข้อมูลไปให้กับส่วนลงคะแนนเสียง



รูป 3.9 การรับการตอบกลับ

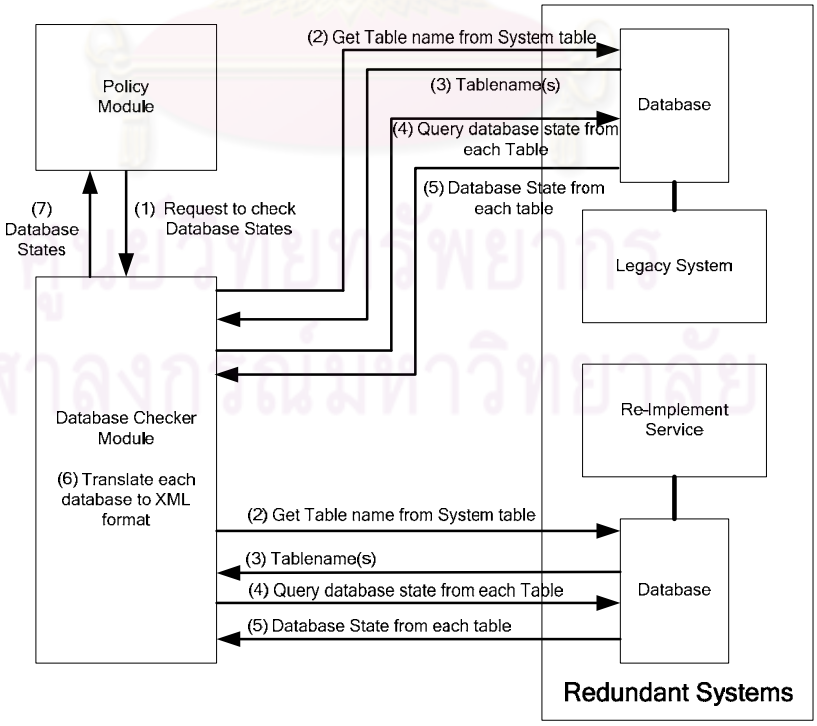
3.2.2.5 **ส่วนตรวจสอบสถานะฐานข้อมูล** (database checker module) ส่วนนโยบายจะตรวจสอบว่าระบบเลกาซีและระบบใหม่ที่ตอบผลลัพธ์กลับมานั้นมีการใช้งานฐานข้อมูลหรือไม่ หากระบบดังกล่าวมีการใช้งานฐานข้อมูล ส่วนนโยบายจะเรียกใช้งานส่วนตรวจสอบสถานะฐานข้อมูล (database checker module) โดยในการตรวจสอบสถานะฐานข้อมูลนั้นมีวิธีการตรวจสอบอยู่ 2 วิธี ดังนี้

1.<u>การตรวจสอบด้วยคำค้นข้อมูลจากตารางติดตาม</u> การตรวจสอบด้วยคำค้นข้อมูลจากตารางติดตามนั้น สามารถใช้ได้ในกรณีที่ระบบจัดการฐานข้อมูลมีฟังก์ชันการสร้างตารางติดตาม ซึ่งวิธีนี้สามารถทำงานได้รวดเร็วกว่าวิธีการห่อฐานข้อมูล (กล่าวถึงในหัวข้อถัดไป) โดยในการตรวจสอบด้วยคำค้นข้อมูลจากตารางติดตามนั้นมีขั้นตอนการตรวจสอบ ตามรูปที่ 3.10 โดย เมื่อระบบที่นำมาทดสอบมีการใช้งานฐานข้อมูล ส่วนนโยบายจะ(1) ร้องขอการตรวจสอบสถานะข้อมูล ส่วนตรวจสอบสถานะจะนำรายละเอียดฐานข้อมูลของทั้งระบบเลกาซีและระบบใหม่มาสร้างเส้นทางการเชื่อมต่อและเชื่อมต่อไปยังฐานข้อมูลของระบบเลกาซีและระบบใหม่ผ่านโอเลดีบีซึ่งได้กล่าวรายละเอียดไว้ในบทที่ 2 เมื่อเชื่อมต่อไปยังฐานข้อมูลของระบบเลกาซีและระบบใหม่แล้ว ส่วนตรวจสอบสถานะฐานข้อมูลจะสร้างคำค้นข้อมูล (query) เพื่อไป (2) ตรวจสอบยังตารางติดตาม (trace table) ซึ่งเป็นตารางที่เก็บคำสั่งจัดการฐานข้อมูลในกรณีที่มีการใช้คำสั่งจัดการกับข้อมูลภายในฐานข้อมูล เมื่อส่วนตรวจสอบสถานะฐานข้อมูล (3) ทราบถึงคำสั่งจัดการฐานข้อมูลของระบบเลกาซีและระบบใหม่แล้ว ส่วนตรวจสอบสถานะฐานข้อมูลจะนำคำสั่งดังกล่าวมาแปลงให้เป็นคำค้นข้อมูลและนำคำค้นข้อมูลดังกล่าวไป (4) ค้นสถานะฐานข้อมูลของระบบเลกาซีและระบบใหม่ เมื่อ (5) ได้รับสถานะฐานข้อมูลของระบบเลกาซีและระบบใหม่ ส่วนตรวจสอบสถานะฐานข้อมูลจะ (6) ส่งสถานะฐานข้อมูลของระบบเลกาซีและระบบใหม่ ไปยังส่วนนโยบาย



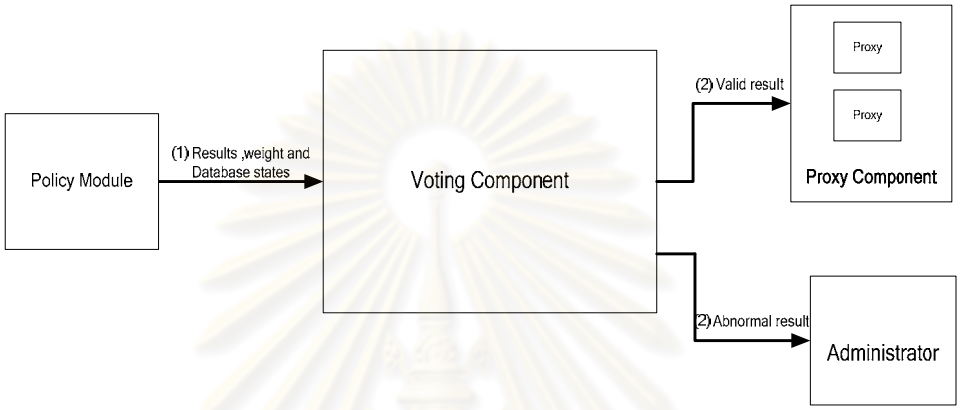
รูป 3.10 การตรวจสอบสถานะฐานข้อมูลด้วยคำค้นข้อมูลจากตารางติดตาม

2.<u>การตรวจสอบด้วยวิธีการห่อฐานข้อมูล</u> การตรวจสอบด้วยวิธีการห่อฐานข้อมูลนั้นใช้ในกรณีที่ระบบจัดการฐานข้อมูลที่นำมาใช้นั้นไม่มีฟังก์ชันในการสร้างตารางติดตาม ซึ่งมักเกิดกับฐานข้อมูลระดับผู้ใช้ โดยการตรวจสอบด้วยวิธีการห่อฐานข้อมูลนั้นสามารถตรวจสอบสถานะฐานข้อมูลแบบสัมพันธ์ได้ทุกระบบ เนื่องจากนำข้อมูลในตารางมาแปลงให้อยู่ในรูปของโครงสร้าง XML และนำมาเปรียบเทียบกับอีกระบบทีละตาราง อย่างไรก็ตามวิธีดังกล่าวทำงานได้ค่อนข้างล่าช้าเมื่อเทียบกับวิธีการตรวจสอบด้วยคำค้นข้อมูลจากตารางติดตาม โดยในการตรวจสอบด้วยวิธีการห่อฐานข้อมูลนั้นมีขั้นตอนการตรวจสอบ ตามรูปที่ 3.11 โดย เมื่อระบบที่นำมาทดสอบมีการใช้งานฐานข้อมูล ส่วนนโยบายจะ (1) ร้องขอการตรวจสอบสถานะข้อมูล ส่วนตรวจสอบสถานะฐานข้อมูลจะนำรายละเอียดฐานข้อมูลของทั้งระบบเลกาซีและระบบใหม่มาสร้างเส้นทางการเชื่อมต่อและเชื่อมต่อไปยังฐานข้อมูลของระบบเลกาซีและระบบใหม่ผ่านโอเลดีบีซึ่งได้กล่าวรายละเอียดไว้ในบทที่ 2 เมื่อเชื่อมต่อไปยังฐานข้อมูลของระบบเลกาซีและระบบใหม่แล้ว ส่วนตรวจสอบสถานะฐานข้อมูลจะสร้างคำค้นข้อมูล (query) เพื่อไป (2) ค้นหายังตารางระบบ (Syetem table) เพื่อค้นหาชื่อตารางผู้ใช้ (user table) ฐานข้อมูลแต่ละระบบจะ(3) ส่งชื่อตารางผู้ใช้กลับมา ส่วนตรวจสอบสถานะฐานข้อมูลจะสร้างคำค้นข้อมูลเพื่อ (4) ค้นสถานะฐานข้อมูลจากแต่ละตารางผู้ใช้ เมื่อ (5) ได้สถานะฐานข้อมูลทุกตารางของแต่ละระบบแล้ว ส่วนตรวจสอบสถานะฐานข้อมูลดังกล่าวมาแปลงให้อยู่ในรูปแบบของ XML และ (6) ส่งสถานะฐานข้อมูลดังกล่าว ไปยังส่วนนโยบาย



รูป 3.11 การตรวจสอบสถานะฐานข้อมูลด้วยวิธีการห่อฐานข้อมูล

3.2.2.6 **การลงคะแนนเสียง** (voting) ตามรูปที่ 3.12 เมื่อส่วนนโยบายส่งผลลัพธ์ น้ำหนักและสถานะฐานข้อมูล (1) ของระบบที่ตอบกลับมาให้กับส่วนคะแนนเสียงแล้ว จึงเริ่มการลงคะแนนเสียง (2) ส่วนการลงคะแนนเสียงจะส่งผลลัพธ์ดังกล่าวให้กับไคลเอนท์ผ่านทางพรอกซี ระบบอื่นๆที่ให้ผลลัพธ์หรือสถานะฐานข้อมูลแตกต่างจากระบบที่น่าเชื่อถือดังกล่าวส่วนการลงคะแนนจะถือว่าผิดปกติ โดยส่วนการลงคะแนนเสียงจะแจ้งให้กับผู้ดูแลระบบ เพื่อนำไปปรับปรุงแก้ไขข้อผิดพลาดต่อไป



รูป 3.12 การตรวจสอบและคัดเลือกผลลัพธ์

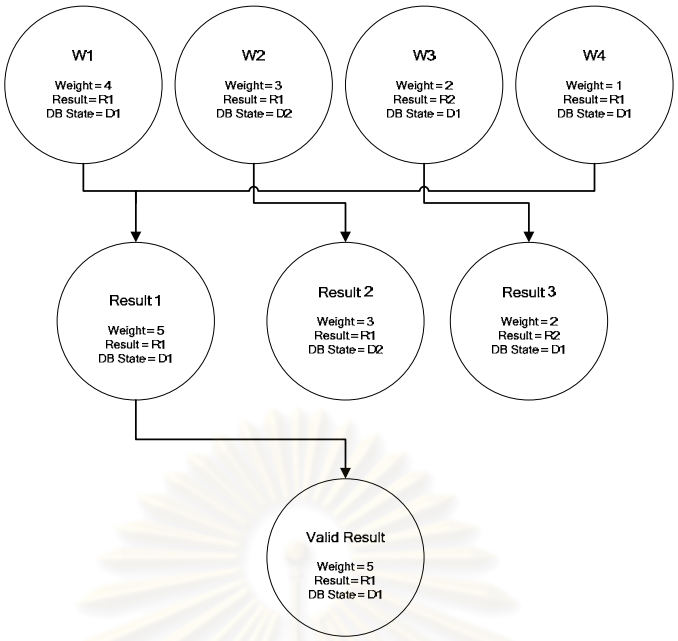
โดยผู้วิจัยได้กำหนดกฎเกณฑ์การลงคะแนนเสียงไว้ ดังนี้

1. น้ำหนักที่ให้กับเว็บเซอร์วิซที่นำมาเปรียบเทียบต้องไม่ซ้ำกัน

2. ส่วนการลงคะแนนเสียงจะรวมน้ำหนักเว็บเซอร์วิซที่มีผลลัพธ์และสถานะฐานข้อมูลที่เหมือนกันเข้าไว้ด้วยกัน

3. ส่วนการลงคะแนนเสียงจะแต่งตั้ง (promote) ผลลัพธ์และสถานะฐานที่มีผลรวมของน้ำหนักมากที่สุดเป็นผลลัพธ์การทำงานที่น่าเชื่อถือ ดังรูปที่ 3.13

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.13 ขั้นตอนคัดเลือกผลลัพธ์

จากรูปที่ 3.13 ผู้วิจัยกำหนดให้มีเว็บเซอร์วิซของแต่ละระบบเปรียบเทียบกัน 4 เว็บเซอร์วิซ ซึ่งตอบผลลัพธ์และมีสถานะฐานข้อมูล ดังแสดงในตารางที่ 3.3

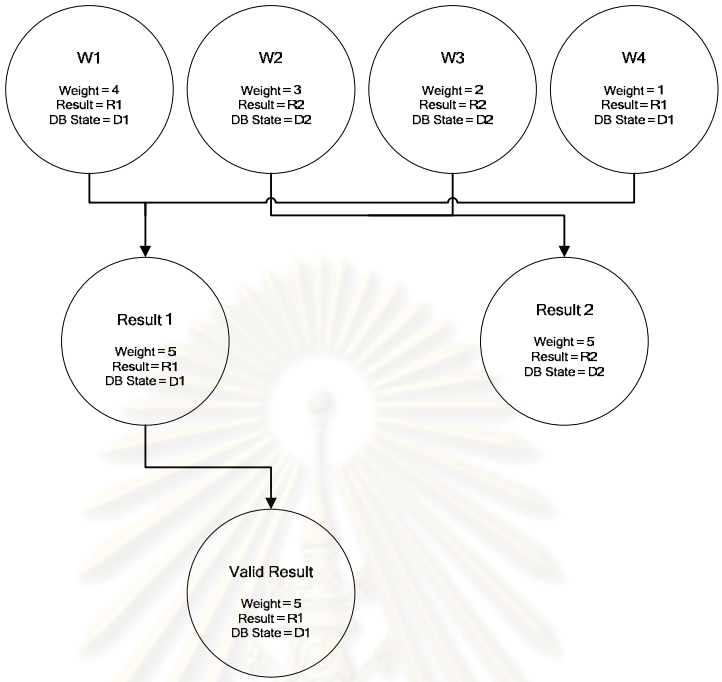
ตารางที่ 3.3 ผลลัพธ์และสถานะฐานข้อมูลของแต่ละระบบ

เว็บเซอร์วิซ	น้ำหนัก	ผลลัพธ์	สถานะฐานข้อมูล
W1	4	R1	D1
W2	3	R1	D2
W3	2	R2	D1
W4	1	R1	D1

จากตารางที่ 3.3 เว็บเซอร์วิซชื่อ W1 กับเว็บเซอร์วิซชื่อ W4 มีผลลัพธ์และสถานะฐานข้อมูลเหมือนกันจึงสามารถนำน้ำหนักของเว็บเซอร์วิซดังกล่าวมา รวมกัน เท่ากับ 5 ส่วนเว็บเซอร์วิซชื่อ W2 นั้นแม้ว่าจะให้ผลลัพธ์เหมือนกันกับเว็บเซอร์วิซชื่อ W3 แต่สถานะฐานข้อมูลของระบบทั้งสองแตกต่างกัน จึงถือว่าเป็นคนละผลลัพธ์การทำงานกัน เมื่อเทียบน้ำหนักระหว่างผลลัพธ์การทำงาน Result1, Result2 และ Result3 ตามลำดับแล้ว ปรากฏว่าผลลัพธ์การทำงาน Result1 มีน้ำหนักมากที่สุด ระบบอื่นๆที่มีผลลัพธ์หรือสถานะฐานข้อมูลแตกต่างจาก Result1 จะถือว่าผิดปกติ

ในกรณีที่มีผลรวมน้ำหนักของเว็บเซอร์วิซที่มีผลลัพธ์กับสถานะฐานข้อมูลเหมือนกัน เข้าไว้ด้วยกัน และปรากฏว่าน้ำหนักรวมของแต่ละผลลัพธ์การทำงานนั้นเท่ากัน ส่วน

การลงคะแนนเสียงจะแต่งตั้งผลลัพธ์การทำงานของเว็บเซอร์วิซที่มีน้ำหนักมากที่สุดเป็น ผลลัพธ์การทำงานที่น่าเชื่อถือ ดังรูปที่ 3.14



รูปที่ 3.14 ขั้นตอนการคัดเลือกผลลัพธ์เมื่อน้ำหนักรวมเท่ากัน

จากรูปที่ 3.14 ผู้วิจัยกำหนดให้มีเว็บเซอร์วิซของแต่ละระบบเปรียบเทียบกัน 4 เว็บเซอร์วิซ ซึ่งตอบผลลัพธ์และมีสถานะฐานข้อมูล ดังแสดงในตารางที่ 3.4

ตารางที่ 3.4 ผลลัพธ์และสถานะฐานข้อมูลของแต่ละระบบ

เว็บเซอร์วิซ	น้ำหนัก	ผลลัพธ์	สถานะฐานข้อมูล
W1	4	R1	D1
W2	3	R2	D2
W3	2	R2	D2
W4	1	R1	D1

จากตารางที่ 3.4 เว็บเซอร์วิซชื่อ W1 กับเว็บเซอร์วิซชื่อ W4 มีผลลัพธ์และสถานะฐานข้อมูลเหมือนกันจึงสามารถนำน้ำหนักของเว็บเซอร์วิซดังกล่าวมารวมกัน เท่ากับ 5 โดยเว็บเซอร์วิซชื่อ W2 กับเว็บเซอร์วิซชื่อ W3 มีผลลัพธ์และสถานะฐานข้อมูลเหมือนกันจึงสามารถนำน้ำหนักของเว็บเซอร์วิซดังกล่าวมารวมกันได้ซึ่งเท่ากับ 5 เช่นเดียวกัน จึงทำให้ไม่สามารถเลือกผลลัพธ์ได้ ดังนั้นผู้วิจัยจึงยึดผลลัพธ์และสถานะฐานข้อมูลจากระบบที่มีน้ำหนักมากที่สุดที่ตอบระหว่าง Result1 หรือ Result2 ซึ่งในตัวอย่างนี้คือ ระบบ W1 ซึ่งตอบ Result1 ดังนั้น

ผลลัพธ์การทำงานที่น่าเชื่อถือคือ Result1 ระบบใดที่ให้ผลการทำงานแตกต่างจาก Result1 จะถือว่าผิดปกติ

3.2.3 บุคลากร (people) โมเดลการย้ายระบบนี้ต้องประกอบด้วยบุคลากร ดังนี้

1. ผู้ติดตั้งระบบ (deployer) ทำหน้าที่เปลี่ยนเส้นทางการร้องขอจากเครื่องให้บริการเว็บเซอร์วิซของระบบใหม่มายังเครื่องให้บริการพรอกซีโดยอ้างอิงการเปลี่ยนไปใช้งานพรอกซีจากข้อ 3.2.2.2 พรอกซี รวมทั้งลงทะเบียนระบบและกำหนดนโยบายเพื่อใช้เป็นแนวทางให้กับเซอร์วิซตัวกลางในการคัดเลือกผลลัพธ์

2. ผู้ดูแลระบบ (administrator) ทำหน้าที่ตรวจสอบความแตกต่างระหว่างระบบเลกาซีและระบบใหม่ผ่านส่วนต่อประสานผู้ใช้ของเซอร์วิซตัวกลาง หากพบความแตกต่างก็จะรายงานให้กับบุคลากรที่เกี่ยวข้องได้ทราบ

3. ผู้ใช้ (user) ผู้ใช้งานโปรแกรมประยุกต์ผ่านทางส่วนต่อประสานผู้ใช้ของระบบใหม่

3.3 การทำงานของโมเดลการย้ายระบบ

ก่อนเริ่มกระบวนการทำงานของโมเดลการย้ายระบบตามรูปที่ 3.15 นั้น ต้องมีขั้นตอนการเตรียมระบบเลกาซีและระบบใหม่ โดยฟังก์ชันของระบบใหม่นั้นผู้วิจัยได้สร้างเลียนแบบฟังก์ชันเดิมของระบบเลกาซีด้วยเทคโนโลยีเว็บเซอร์วิซ และนำฟังก์ชันของระบบเลกาซีมาห่อด้วยชุดเครื่องมือโซป เพื่อให้อยู่บนมาตรฐานเว็บเซอร์วิซเช่นเดียวกับระบบใหม่เพื่อให้สามารถทำงานร่วมกันกับเซอร์วิซตัวกลางได้ เมื่อทำให้ระบบเลกาซีและระบบใหม่อยู่บนมาตรฐานเดียวกันแล้ว จึงเริ่มกระบวนการย้ายระบบ โดยมีขั้นตอนดังนี้

1. ผู้ติดตั้งระบบนำเว็บเซอร์วิซของระบบเก่าและระบบใหม่ที่ต้องการจะเปรียบเทียบมาลงทะเบียนกับส่วนลงทะเบียน ซึ่งส่วนลงทะเบียนจะสร้างเอกสารดับบลิวเอสดีแอลและวิเคราะห์เพื่อทราบถึงวิธีการเรียกใช้งานเว็บเซอร์วิซดังกล่าว

2. ในกรณีที่เว็บเซอร์วิซดังกล่าวยังไม่มีพรอกซีมารองรับ ตัวกลางจะสร้างพรอกซีซึ่งมีส่วนต่อประสานเช่นเดียวกับเว็บเซอร์วิซ โดยพรอกซีจะรับการร้องขอจากไคลเอนท์แทนเว็บเซอร์วิซของระบบใหม่ โดยการเปลี่ยนมาใช้งานพรอกซีนั้นผู้วิจัยได้แสดงไว้ในข้อ 3.2.2.2 พรอกซี

3. ผู้ติดตั้งระบบต้องสื่อสารกับตัวกลางในการกำหนดวิธีการตัดสินใจเลือกคำตอบที่เหมาะสมตามสถานการณ์ต่างๆ โดยการกำหนดแนวทางดังกล่าวผ่านทางนโยบาย

4. ไคลเอนท์เรียกใช้งานโปรแกรมประยุกต์ผ่านส่วนควบคุมรายการ (transaction control) ของระบบใหม่

5. การร้องขอดังกล่าวจะถูกส่งไปที่พรอกซี

6. พรอกซีส่งการร้องขอนั้นให้กับส่วนนโยบาย

7. ส่วนนโยบายจะตรวจสอบชื่อเว็บเซอร์วิซจากข้อความการร้องขอและนำชื่อดังกล่าวไปค้นหาภายในนโยบายเพื่อให้ทราบถึงชื่อเว็บเซอร์วิซของระบบเลกาซีที่ถูกกำหนดให้เปรียบเทียบ,จำนวนผู้ร่วมลงคะแนน,ระยะเวลารอคอยและการกระทำเมื่อจำนวนผู้ลงคะแนนไม่ครบ เมื่อทราบชื่อเว็บเซอร์วิซของทั้งสองระบบที่จะนำมาเปรียบเทียบกันแล้วก็จะนำชื่อดังกล่าวส่งให้กับดับบลิวเอสมัลติคาสต์

8. ดับบลิวเอสมัลติคาสต์จะสร้างข้อความการร้องขอสำหรับเรียกใช้งานเว็บเซอร์วิซของระบบเลกาซีและระบบใหม่และกระจายส่งการร้องขอนั้นไปยังเว็บเซอร์วิซของระบบเลกาซีและระบบใหม่

9. ทั้งระบบเลกาซีและระบบใหม่จะประมวลผลการร้องขอนั้นในลักษณะที่เป็นอิสระต่อกัน เมื่อระบบใดประมวลผลเสร็จก็จะส่งผลลัพธ์กลับมายังส่วนนโยบาย เมื่อจำนวนผู้ตอบครบตามที่กำหนด ส่วนนโยบายจะตรวจสอบกับส่วนลงทะเบียนว่าระบบที่ตอบกลับมานั้นมีการใช้งานฐานข้อมูลหรือไม่ ถ้าไม่มีการใช้งานฐานข้อมูล ตัวกลางจะนำผลลัพธ์ของระบบเลกาซีและระบบใหม่ไปเข้ากระบวนการลงคะแนนเสียง ในกรณีที่มีจำนวนผู้ตอบน้อยกว่าที่กำหนดไว้ในนโยบาย ส่วนนโยบายจะรอจนกว่าจะครบระยะเวลารอคอย เมื่อครบระยะเวลารอคอยและจำนวนผู้ตอบยังน้อยกว่าที่กำหนดไว้ในนโยบาย ส่วนนโยบายจะกระทำในส่วนยกเว้นภายในนโยบาย หากส่วนยกเว้นกำหนดเป็น Ignore_cardinality และเซอร์วิซตัวกลางมีคำตอบ ส่วนนโยบายจะส่งคำตอบที่มีน้ำหนักมากที่สุดดังกล่าวให้กับพรอกซีเพื่อที่พรอกซีจะนำคำตอบนั้นส่งให้กับไคลเอนท์ หากไม่มีคำตอบส่วนนโยบายจะส่งข้อผิดพลาดไปยังไคลเอนท์ผ่านทางพรอกซี หากส่วนยกเว้นกำหนดเป็น error ส่วนนโยบายจะส่งข้อผิดพลาดไปยังไคลเอนท์ผ่านทางพรอกซีทันที

10. ในกรณีที่ระบบที่ตอบกลับมานั้นมีการใช้งานฐานข้อมูล ส่วนนโยบายจะเรียกใช้งานส่วนตรวจสอบสถานะฐานข้อมูล

11. ส่วนตรวจสอบสถานะฐานข้อมูลจะนำรายละเอียดฐานข้อมูลของทั้งระบบเลกาซีและระบบใหม่มาสร้างเส้นทางการเชื่อมต่อและเชื่อมต่อไปยังฐานข้อมูลของระบบเลกาซีและระบบใหม่ผ่านโอเลดีบีซึ่งได้กล่าวรายละเอียดไว้ในบทที่ 2 เมื่อเชื่อมต่อไปยังฐานข้อมูลของระบบเลกาซีและระบบใหม่แล้ว ส่วนตรวจสอบสถานะฐานข้อมูลจะสร้างคำค้นข้อมูล (query) เพื่อไปตรวจสอบตามวิธีการตรวจสอบที่ผู้วางระบบเลือกไว้ในขั้นตอนลงทะเบียน โดยรายละเอียดของแต่ละวิธีนั้นผู้วิจัยได้กล่าวไว้ในหัวข้อ 3.2.2.5 ส่วนตรวจสอบสถานะฐานข้อมูล

12. ฐานข้อมูลของระบบเลกาซีและระบบใหม่จะส่งสถานะฐานข้อมูลกลับมา

13. เมื่อได้รับสถานะฐานข้อมูลของระบบเลกาซีและระบบใหม่แล้วก็จะนำสถานะฐานข้อมูลดังกล่าว ส่งให้กับส่วนนโยบาย

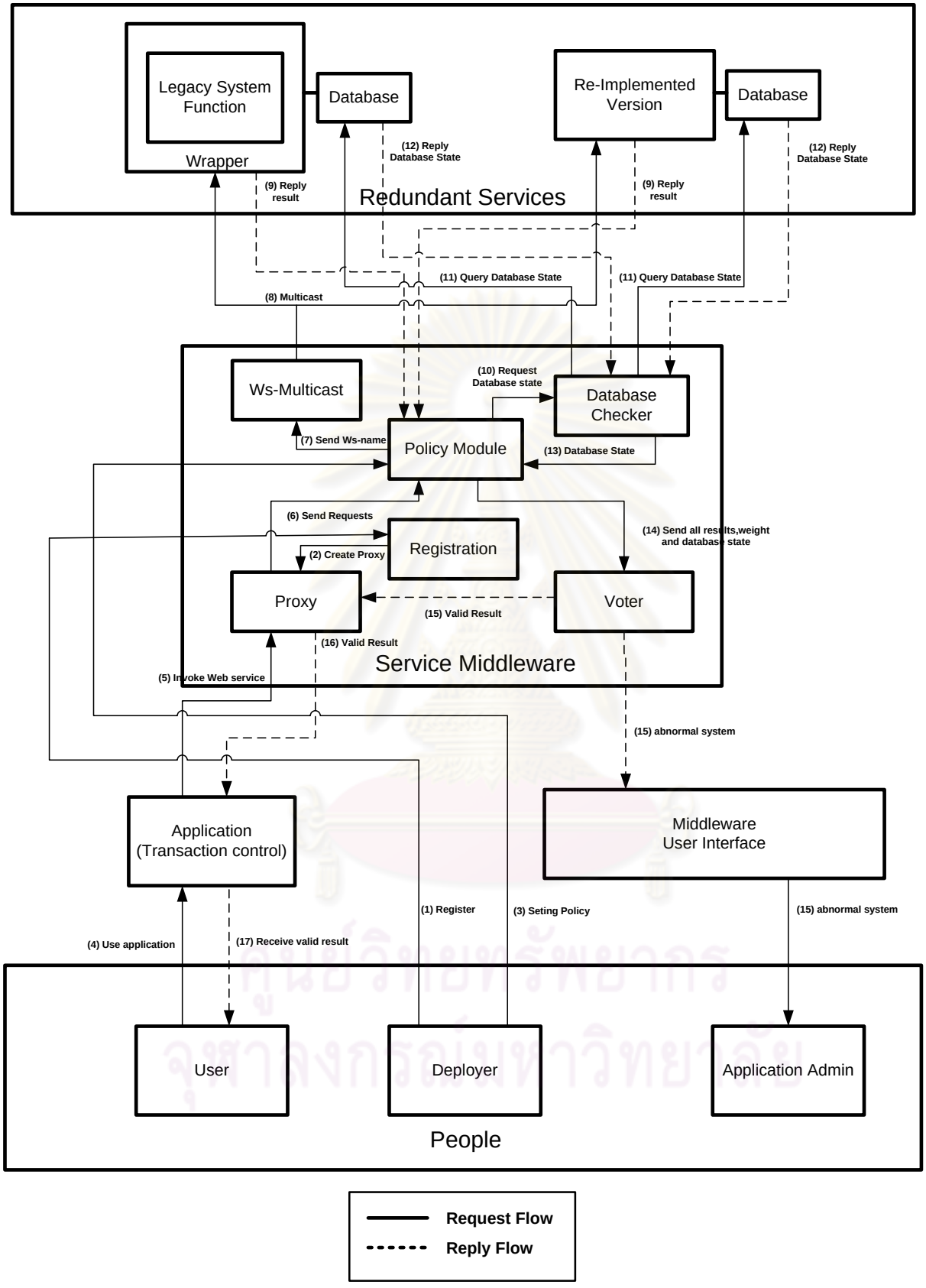
14. ส่วนนโยบายจะนำผลลัพธ์, น้ำหนักของระบบที่ตอบและสถานะฐานข้อมูลของระบบเลกาซีและระบบใหม่มาลงคะแนนเสียง โดยการลงคะแนนเสียงนั้นจะถือว่าผลลัพธ์และสถานะฐานข้อมูลของระบบที่มีน้ำหนักมากที่สุดเป็นระบบที่น่าเชื่อถือ

15. ส่วนลงคะแนนเสียงจะส่งผลลัพธ์ดังกล่าวให้กับไคลเอนท์ผ่านทางพรอกซี ระบบอื่นๆที่ให้ผลลัพธ์หรือสถานะฐานข้อมูลแตกต่างจากระบบที่น่าเชื่อถือดังกล่าวจะถือว่าผิดปกติ ส่วนลงคะแนนเสียงจะรายงานความแตกต่างให้กับผู้ดูแลระบบผ่านทางส่วนต่อประสานผู้ใช้ของตัวกลาง

16. พรอกซีจะส่งผลลัพธ์ดังกล่าวให้กับส่วนควบคุมธุรกรรมของไคลเอนท์

17. ไคลเอนท์จะได้รับทราบผลลัพธ์ผลลัพธ์จากตัวควบคุมธุรกรรมของระบบใหม่

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูป 3.15 การทำงานของโมเดลการย้ายระบบ

ในบทนี้ได้กล่าวถึงการออกแบบโมเดลสำหรับสนับสนุนการย้ายระบบ โดยได้นำทฤษฎีแนวคิดจากบทที่ 2 มาใช้ในการออกแบบส่วนประกอบ ภายในโมเดล รวมทั้งแสดงการทำงานร่วมกันระหว่างส่วนประกอบเหล่านั้น ในบทต่อไปนั้นผู้วิจัยจะนำโมเดลที่ได้ออกแบบไปสร้างเป็นระบบสนับสนุนเพื่อใช้ทดสอบงานวิจัยชิ้นนี้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

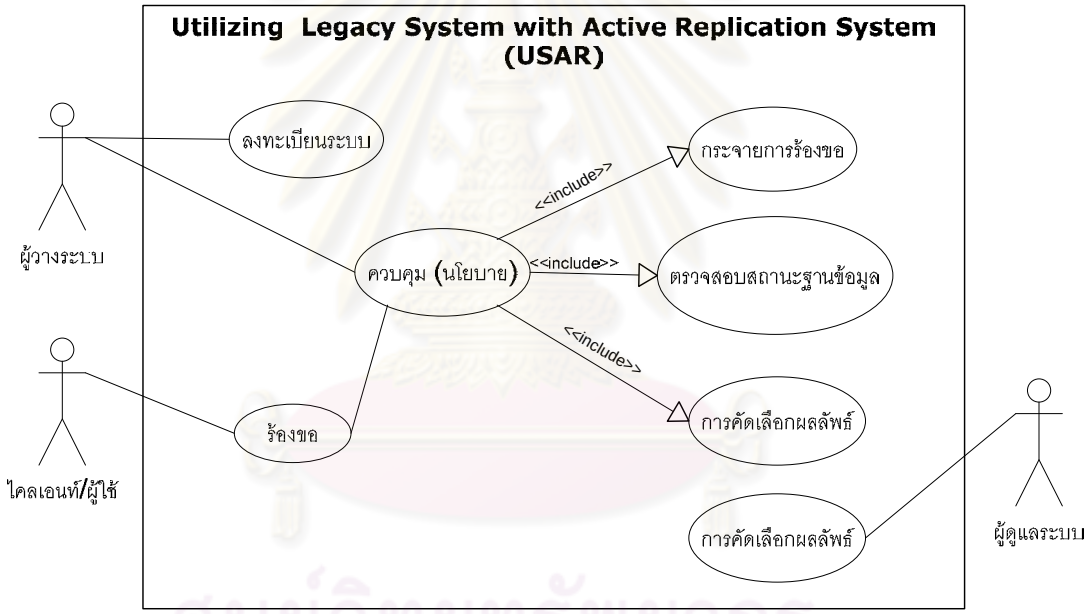
การพัฒนาและทดสอบระบบสนับสนุน

จากการศึกษาและออกแบบขั้นตอนการดำเนินงานกระบวนการย้ายระบบตามที่ได้นำเสนอไปในบทที่ 3 แล้วนั้น ผู้วิจัยได้ทำออกแบบระบบสนับสนุนและทำการพัฒนาระบบสนับสนุนตามขั้นตอนการดำเนินงานดังกล่าว

4.1 การออกแบบระบบสนับสนุน

4.1.1 แผนภาพยูสเคส (Use Case Diagram)

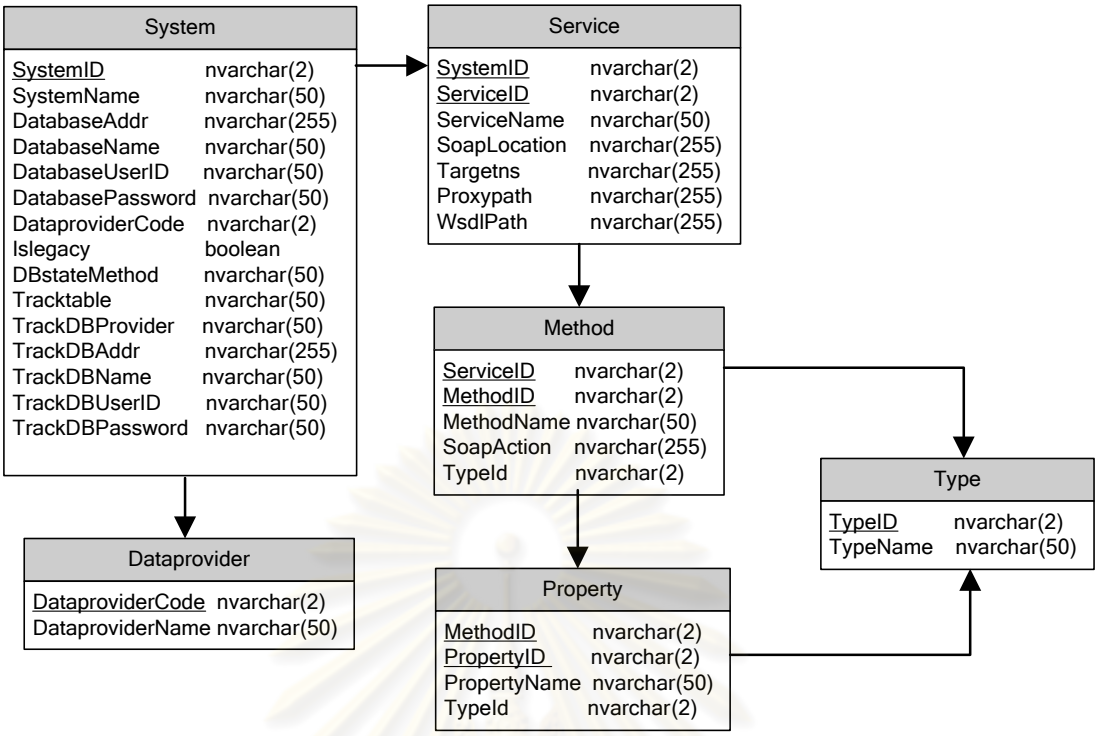
แผนภาพยูเคสนำมาใช้อธิบายฟังก์ชันการทำงานหลักของระบบสนับสนุนดังแสดงใน รูปที่ 4.1



รูปที่ 4.1 แผนภาพยูสเคสของระบบสนับสนุน

4.1.2 แผนภาพอีอาร์ (E-R Diagram)

แผนภาพอีอาร์ เป็นแผนภาพที่ใช้ในการอธิบายถึงโครงสร้างและความสัมพันธ์ระหว่างข้อมูลภายในฐานข้อมูล ดังแสดงในรูปที่ 4.2



รูปที่ 4.2 แผนภาพอีอาร์ของฐานข้อมูล

4.2 ความต้องการด้านหน้าที่ของระบบสนับสนุน

ความต้องการด้านหน้าที่ของระบบสนับสนุนแนวคิดการย้ายระบบซึ่งใช้ความเสถียรของระบบเลกาซีร่วมกับแอคทีฟเรพลิเคชันโดยอ้างอิงแผนภาพยูสเคส มีดังตารางที่ 4.1

ตารางที่ 4.1 ความต้องการด้านหน้าที่ของระบบสนับสนุน

รหัส	ชื่อ	คำอธิบาย
F01	ลงทะเบียนระบบ	ส่วนการลงทะเบียนระบบ มีหน้าที่ดังนี้ 1. บันทึกชื่อระบบ เว็บเซอร์วิซ และข้อมูลฐานข้อมูลของระบบที่นำมาลงทะเบียน 2. สร้างเว็บพรอกซีสำหรับรับการร้องขอแทนที่เว็บเซอร์วิซของระบบที่นำมาลงทะเบียน 3.แก้ไขและลบข้อมูลระบบที่ได้ลงทะเบียนไว้กับระบบสนับสนุน

ตารางที่ 4.1 ความต้องการด้านหน้าที่ของระบบสนับสนุน (ต่อ)

รหัส	ชื่อ	คำอธิบาย
F02	รับการร้องขอ	ส่วนรับการร้องขอมีหน้าที่ ดังนี้ 1. รับการร้องขอจากไคลเอนท์เข้าสู่ระบบสนับสนุน 2. นำผลลัพธ์ที่ได้จากการลงคะแนนเสียงส่งให้กับไคลเอนท์
F03	กำหนดนโยบาย	ส่วนกำหนดนโยบาย มีหน้าที่ดังนี้ 1. ตรวจสอบไวยากรณ์ของนโยบาย 2. บันทึกนโยบายในรูปของเอกสารเอ็กซ์เอ็มแอล 3. แปลความหมายจากนโยบาย 4. เรียกใช้ส่วนกระจายการร้องขอเพื่อกระจายการร้องขอไปยังเว็บเซอร์วิซที่กำหนดไว้ในนโยบาย 5. รอคอยการตอบกลับจากแต่ละเว็บเซอร์วิซ 6. ตรวจสอบว่าแต่ละระบบมีการใช้งานฐานข้อมูลหรือไม่ หากมีการใช้งานฐานข้อมูล จะร้องขอสถานะฐานข้อมูลจากส่วนตรวจสอบสถานะฐานข้อมูล 7. บันทึกแหล่งและคำตอบรวมทั้งสถานะฐานข้อมูล เมื่อจำนวนคำตอบครบตามที่ระบุในนโยบายแล้ว จะส่งแหล่งคำตอบ สถานะฐานข้อมูล รวมทั้งน้ำหนักของเว็บเซอร์วิซดังกล่าวไปยังส่วนคัดเลือกผลลัพธ์ 8. กระทำตามการกระทำที่กำหนดไว้ในส่วนยกเว้นของนโยบาย ในกรณีที่ไม่สามารถคัดเลือกผลลัพธ์ได้พร้อมกับรายงานความผิดปกติดังกล่าวให้กับผู้ดูแลระบบได้ทราบ ผ่านทางส่วนต่อประสานผู้ใช้ของระบบสนับสนุน
F04	กระจายการร้องขอ	ส่วนการกระจายการร้องขอ มีหน้าที่ดังนี้ 1. สร้างข้อความร้องขอสำหรับเรียกใช้งานเว็บเซอร์วิซตามที่ส่วนกำหนดนโยบายระบุ 2. กระจายข้อความร้องขอไปยังเว็บเซอร์วิซตามที่ส่วนกำหนดนโยบายระบุ
F05	ตรวจสอบสถานะฐานข้อมูล	ส่วนของการตรวจสอบสถานะฐานข้อมูล มีหน้าที่ดังนี้ 1. สอบถามสถานะฐานข้อมูลจากฐานข้อมูลของแต่ละระบบ

ตารางที่ 4.1 ความต้องการด้านหน้าที่ของระบบสนับสนุน (ต่อ)

		2. ส่งสถานะข้อมูลของแต่ละระบบให้กับส่วนกำหนดนโยบาย
F06	การคัดเลือกผลลัพธ์	ส่วนของการคัดเลือกผลลัพธ์มีหน้าที่ ดังนี้ 1. ลงคะแนนตามวิธีที่ได้กำหนดไว้ในบทที่ 3 ข้อ 3.2.2.6 2. รายงานความผิดปกติสำหรับระบบที่ให้ผลลัพธ์แตกต่างจากระบบที่มีน้ำหนักมากที่สุดให้กับผู้ดูแลระบบได้ทราบ ผ่านทางส่วนต่อประสานผู้ใช้ของระบบสนับสนุน 3. ส่งผลลัพธ์ของระบบที่มีน้ำหนักมากที่สุดให้กับพรอกซี
F07	รายงานสถานะ	ส่วนของการรายงานสถานะมีหน้าที่ ดังนี้ 1. รายงานผลลัพธ์ที่ตอบกลับไปยังผู้ใช้/ไคลเอนท์ 2. รายงานระบบที่ให้ผลลัพธ์หรือสถานะฐานข้อมูลที่ผิดปกติ

4.3 การพัฒนาระบบสนับสนุน

4.3.1 สภาพแวดล้อมของระบบสนับสนุน

ฮาร์ดแวร์และซอฟต์แวร์ของสภาพแวดล้อมที่ใช้ในการพัฒนาระบบมีสภาพแวดล้อมทางด้านฮาร์ดแวร์และซอฟต์แวร์ดังต่อไปนี้

ฮาร์ดแวร์

1. หน่วยประมวลผลกลาง รุ่น Intel Centrino 1.4 GHz.
2. หน่วยความจำเข้าถึงโดยสุ่มชนิด DDR ขนาด 1 GB
3. จานบันทึกแบบแข็ง (Hard disk) 100 กิกะไบต์ (100 GB)

ซอฟต์แวร์

1. ระบบปฏิบัติการวินโดวส์เอ็กซ์พี โปรเฟสชันนอล (Windows XP Professional operating system)
2. เครื่องมือพัฒนาโปรแกรมไมโครซอฟต์วิชวลสตูดิโอ รุ่น 2005 (Microsoft visual studio 2005)
3. ระบบจัดการฐานข้อมูลไมโครซอฟต์เอสคิวแอล 2005 เซิฟเวอร์ (Microsoft SQL 2005 server database management system)
4. เว็บให้บริการ ไอไอเอส 6.0 (IIS web server 6.0)

4.3.2 ขั้นตอนการพัฒนาระบบสนับสนุน

การติดตั้งซอฟต์แวร์ในการพัฒนาระบบ เมื่อเตรียมเครื่องมือสำหรับการพัฒนาระบบเรียบร้อยแล้ว ขั้นต่อมาจึงเป็นส่วนของการติดตั้งเครื่องมือทั้งหมดลงในเครื่องคอมพิวเตอร์ที่ใช้พัฒนาระบบ โดยมีลำดับการติดตั้งเครื่องมือเป็นไปตามขั้นตอนต่อไปนี้

1. ติดตั้งระบบปฏิบัติการวินโดว์เอ็กซ์พี โปรเฟสชันนอล และเว็บให้บริการ ไอไอเอส 6.0
2. ติดตั้งระบบจัดการฐานข้อมูลไมโครซอฟต์เอสคิวแอล 2005 เซิฟเวอร์
3. ติดตั้งเครื่องมือพัฒนาโปรแกรมไมโครซอฟต์วิชวลสตูดิโอ 2005

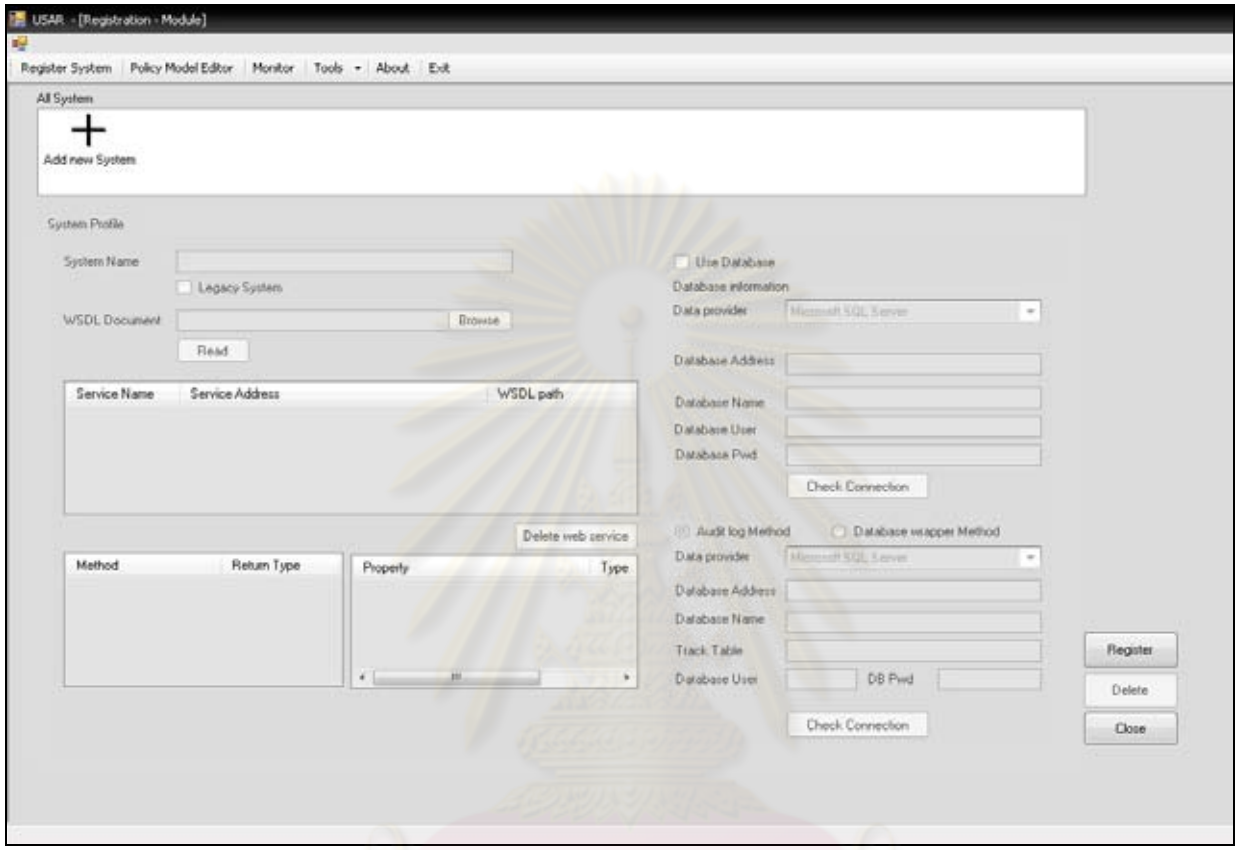
การพัฒนาส่วนต่อประสานผู้ใช้ การพัฒนาส่วนต่อประสานผู้ใช้ของระบบสนับสนุนการย้ายระบบนี้ ผู้วิจัยได้พัฒนาให้สอดคล้องกับขอบเขตของระบบที่ได้กล่าวไว้ในบทที่ 1 และมีการทำงานตรงกับความต้องการด้านหน้าที่ ดังแสดงไว้ในตารางที่ 4.1 โดยสามารถดูตัวอย่างหน้าจอได้ตามรูปที่ 4.3 ถึงรูปที่ 4.11 โดยระบบสนับสนุนที่ผู้วิจัยพัฒนาขึ้นนั้นประกอบด้วยส่วนต่อประสานผู้ใช้ ดังนี้

4.3.2.1 **ส่วนต่อประสานผู้ใช้สำหรับลงทะเบียนระบบ** (user interface for registration) ดังรูปที่ 4.3 เป็นส่วนต่อประสานผู้ใช้สำหรับลงทะเบียนระบบ จุดประสงค์เพื่อให้ระบบสนับสนุนสามารถทราบข้อมูลของระบบที่นำมาเปรียบเทียบ โดยทำให้ระบบสนับสนุนสามารถทำงานร่วมกันกับระบบที่นำมาทดสอบได้ โดยส่วนต่อประสานผู้ใช้สำหรับการลงทะเบียนนั้นประกอบด้วยฟังก์ชันการทำงานย่อยดังนี้

4.3.2.1.1. การเพิ่มข้อมูลระบบ เป็นการให้ข้อมูลที่จำเป็นแก่ระบบสนับสนุน โดยมีขั้นตอนการใช้งาน ดังนี้

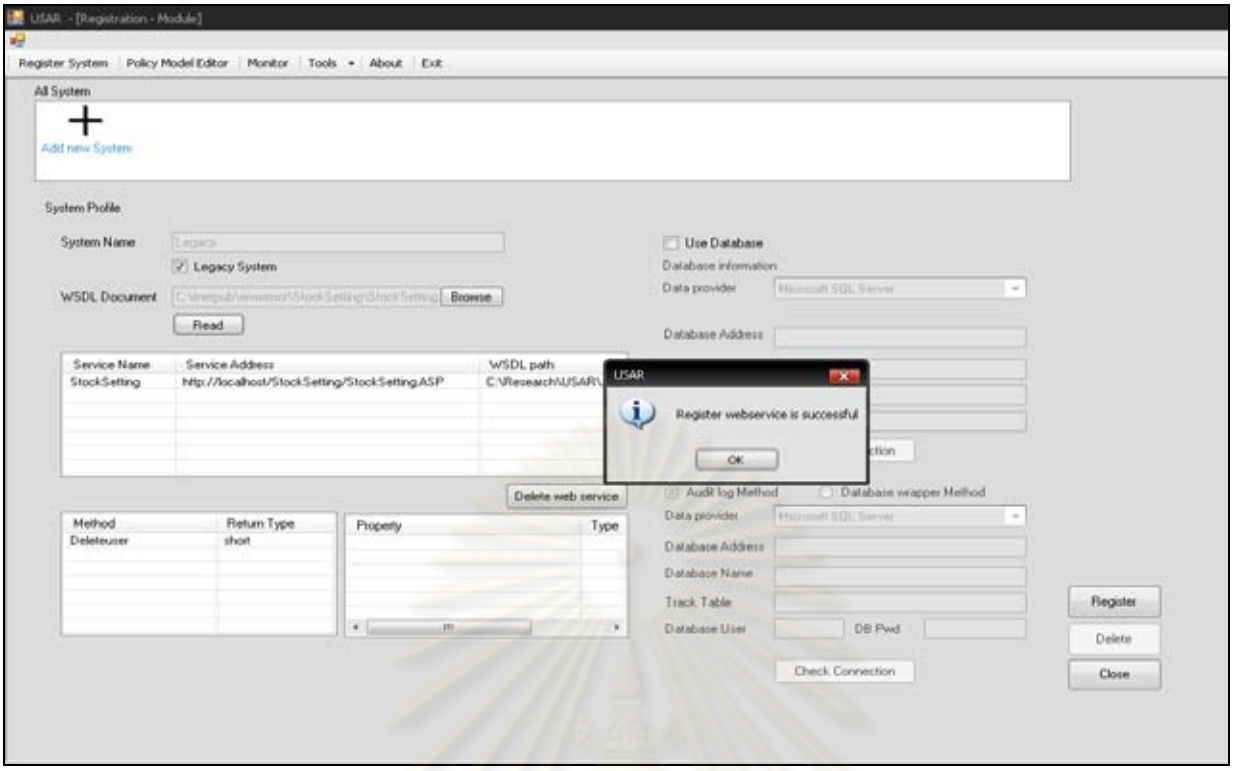
1. ผู้ติดตั้งระบบกดที่ปุ่ม”Add new system” ภายในส่วน All System
2. ผู้ติดตั้งระบบกรอกชื่อระบบซึ่งต้องเป็นชื่อที่ไม่ซ้ำกับชื่อระบบที่เคยลงทะเบียนไว้แล้ว
3. ในกรณีที่มีการใช้งานฐานข้อมูล ให้ผู้ติดตั้งเลือกชนิดระบบจัดการฐานข้อมูล ที่อยู่ฐานข้อมูล ชื่อผู้ใช้ฐานข้อมูล รหัสผ่านฐานข้อมูล วิธีการตรวจสอบสถานะฐานข้อมูล รวมทั้งข้อมูลสำหรับตรวจสอบสถานะฐานข้อมูล
4. ผู้ติดตั้งระบุเว็บเซอร์วิซของระบบดังกล่าวด้วยการนำเอกสารดับบลิวเอสดีแอลให้ระบบสนับสนุนวิเคราะห์เพื่อให้ทราบถึงวิธีการเรียกใช้งานเว็บเซอร์วิซดังกล่าว ดังรูปที่ 4.4 ในกรณีที่เว็บเซอร์วิซนั้นยังไม่มีพรอกซีมารองรับระบบสนับสนุนจะสร้างพรอกซีให้

5. เมื่อผู้ติดตั้งกรอกข้อมูลครบถ้วนแล้ว ให้กดปุ่ม “Register” เมื่อระบบสนับสนุนตรวจสอบข้อมูลเสร็จสิ้น ระบบสนับสนุนจะนำข้อมูลไปบันทึกยังฐานข้อมูลของระบบสนับสนุนและจะแสดงกล่องข้อความ”Registration is successful” ดังรูปที่ 4.5

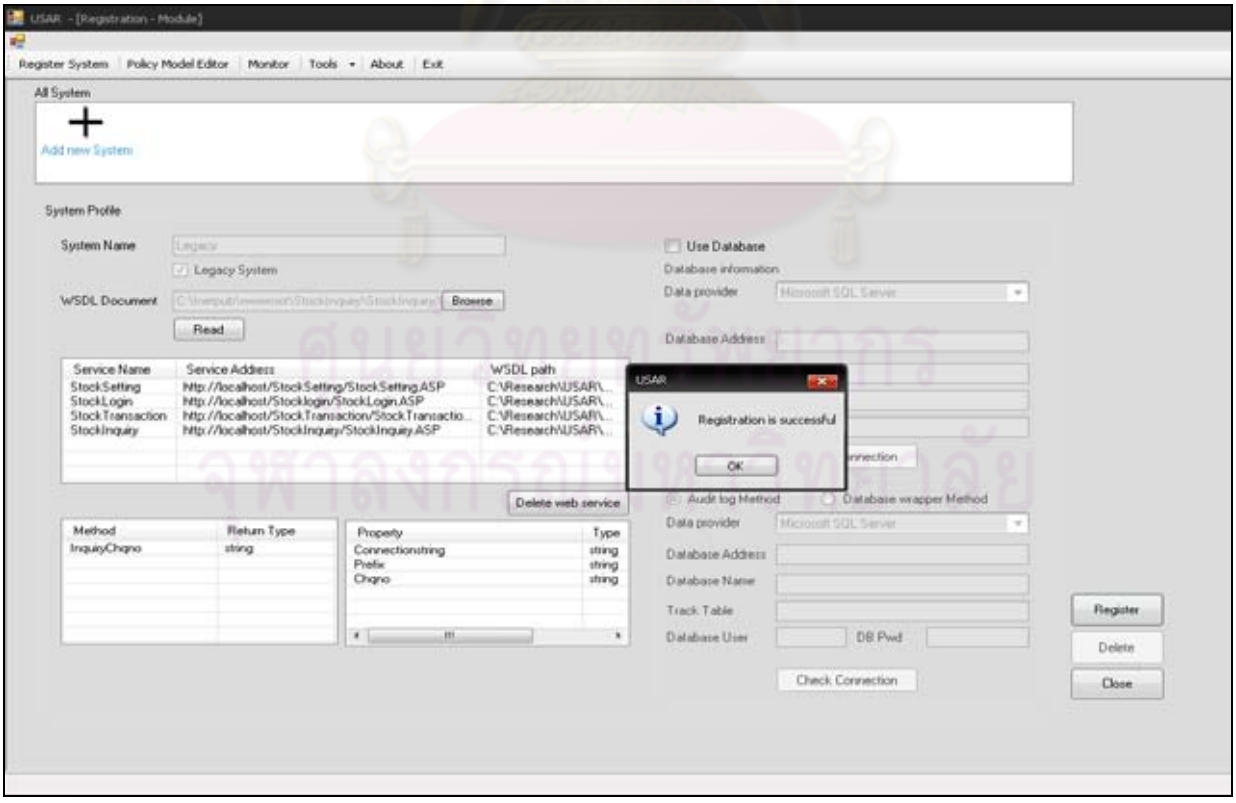


รูปที่ 4.3 แสดงส่วนต่อประสานผู้ใช้สำหรับลงทะเบียนระบบ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



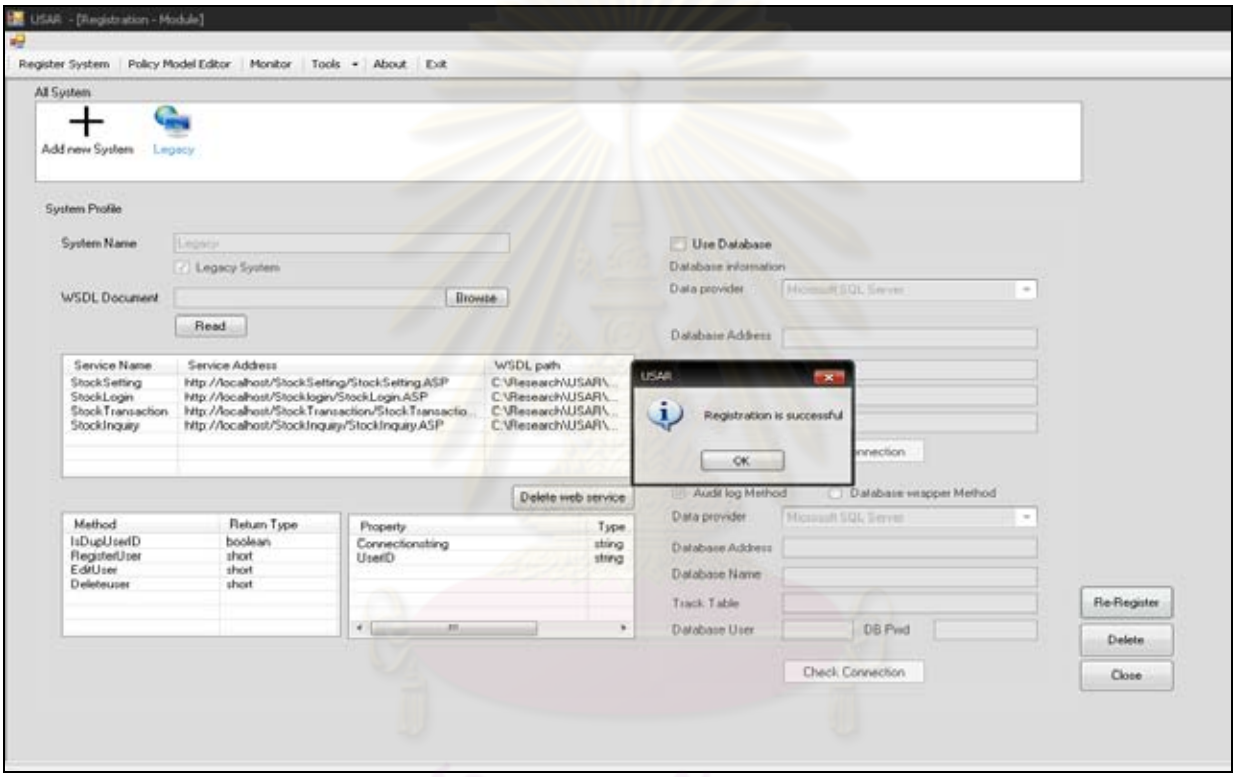
รูปที่ 4.4 แสดงการลงทะเบียนเว็บเซอร์วิซ



รูปที่ 4.5 แสดงการลงทะเบียนระบบเสร็จสมบูรณ์

4.3.2.1.2.<u>การแก้ไขข้อมูลระบบ</u> เป็นการแก้ไขข้อมูลของระบบที่ได้ลงทะเบียนไว้กับระบบสนับสนุน โดยมีขั้นตอนการทำงาน ดังนี้

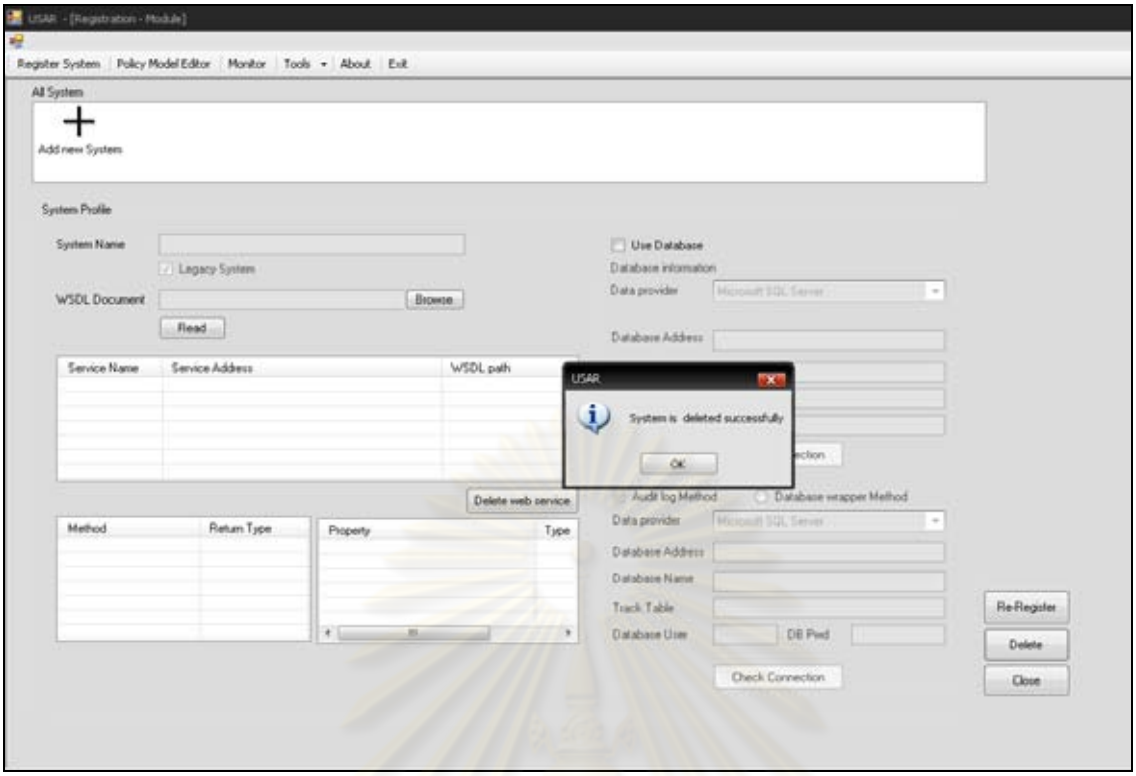
1. เลือกระบบที่ต้องการแก้ไขจากส่วน All System
2. ระบบสนับสนุนจะแสดงข้อมูลของระบบดังกล่าว
3. เมื่อแก้ไขข้อมูลเรียบร้อยแล้ว ให้กดปุ่ม"Re - Register" เมื่อระบบสนับสนุนตรวจสอบข้อมูลเสร็จสิ้น ระบบสนับสนุนจะนำข้อมูลไปปรับปรุงยังฐานข้อมูลของระบบสนับสนุน และจะแสดงกล่องข้อความ"Registration is successful" ดังรูปที่ 4.6



รูปที่ 4.6 แสดงการแก้ไขระบบเสร็จสมบูรณ์

4.3.1.3.<u>การลบข้อมูลระบบ</u> เป็นการลบข้อมูลระบบที่ได้ลงทะเบียนไว้กับระบบสนับสนุน โดยมีขั้นตอนการใช้งาน ดังนี้

1. เลือกระบบที่ต้องการลบจากส่วน All System
2. ระบบสนับสนุนจะแสดงข้อมูลของระบบดังกล่าวให้กดปุ่ม “Delete system”
3. ระบบสนับสนุนจะแสดงกล่องข้อความยืนยันการลบ ให้กดปุ่ม “OK”
4. ระบบสนับสนุนจะลบข้อมูลของระบบดังกล่าวจากฐานข้อมูลของระบบสนับสนุนและจะแสดงกล่องข้อความ"System is deleted successfully" ดังรูปที่ 4.7



รูปที่ 4.7 แสดงการลบระบบเสร็จสมบูรณ์

4.3.2.2. **ส่วนต่อประสานผู้ใช้สำหรับกำหนดนโยบาย** (user interface for policy editor) เป็นส่วนต่อประสานผู้ใช้สำหรับกำหนดนโยบายเพื่อใช้เป็นแนวทางในการคัดเลือกผลลัพธ์ให้กับระบบสนับสนุน โดยทำฟังก์ชันตรวจสอบไวยากรณ์ที่ผู้ติดตั้งระบบกำหนด หากนโยบายดังกล่าวไม่สามารถแปลความหมายได้หรือไม่สอดคล้องกับข้อมูลที่ได้ลงทะเบียนไว้ก็จะแจ้งให้กับผู้ติดตั้งแก้ไข โดยการกำหนดนโยบายนั้นมีขั้นตอนการใช้งาน ดังนี้

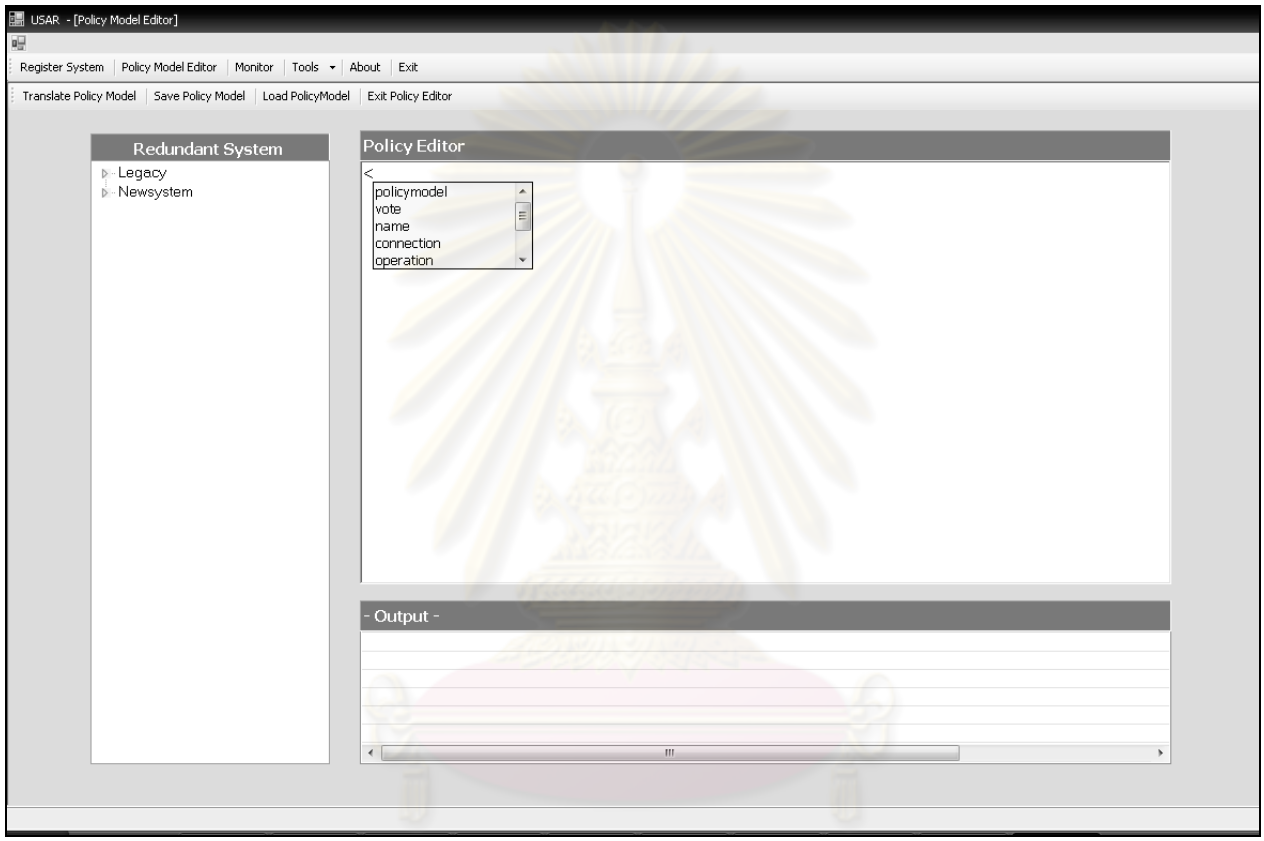
1. ผู้ติดตั้งระบบกำหนดนโยบายตามรูปแบบไวยากรณ์ที่กำหนดไว้ในบทที่ 3 หัวข้อ 3.2.2.3 ส่วนกำหนดนโยบาย โดยระบบสนับสนุนแนะนำคำสั่งที่เหมาะสมผ่านอินเทลลิเซนท์ (Intellisense) ดังรูปที่ 4.8

2. ระบบสนับสนุนตรวจสอบไวยากรณ์ หากไวยากรณ์ไม่ถูกต้องจะแจ้งให้ผู้ติดตั้งระบบทำการแก้ไขผ่านส่วนต่อประสานผู้ใช้ ดังรูปที่ 4.9

3. ระบบสนับสนุนตรวจสอบค่าที่ผู้ติดตั้งระบบกำหนดหากค่าที่ผู้ติดตั้งกำหนดขึ้นนั้นไม่เหมาะสมจะแจ้งให้ผู้ติดตั้งระบบทำการแก้ไข

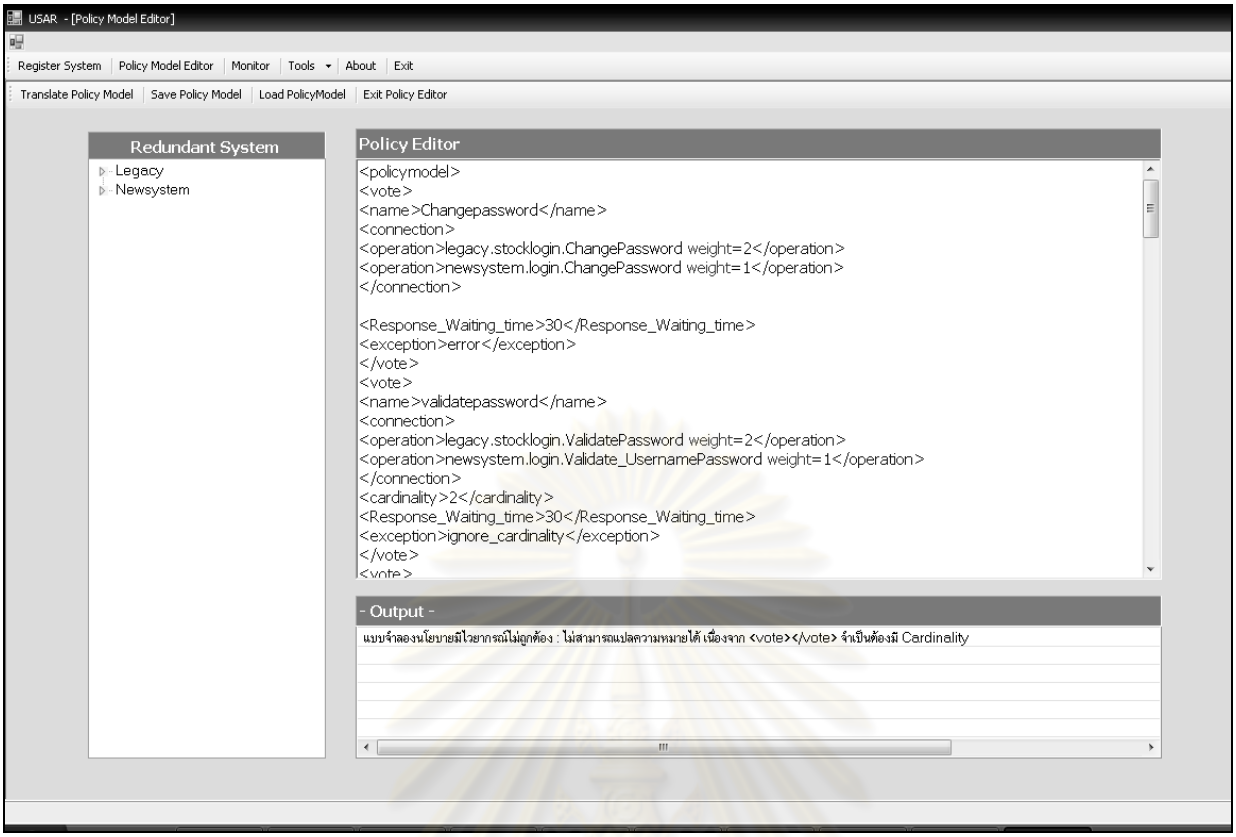
4. เมื่อผู้ติดตั้งระบบกำหนดนโยบายเสร็จสิ้น ให้กดปุ่ม “Translate Policy Model” ระบบสนับสนุนจะแปลนโยบายเพื่อสรุปสิ่งที่ผู้ติดตั้งระบบกำหนดในนโยบายและแสดงการแปลให้ผู้ติดตั้งได้ทราบ ดังรูปที่ 4.10

5. หากสิ่งที่ระบบสนับสนุนแปลตรงกับที่ผู้ติดตั้งต้องการ ให้ผู้ติดตั้งกดปุ่ม “Save Policy Model” พร้อมระบุตำแหน่งที่ต้องการจัดเก็บนโยบาย

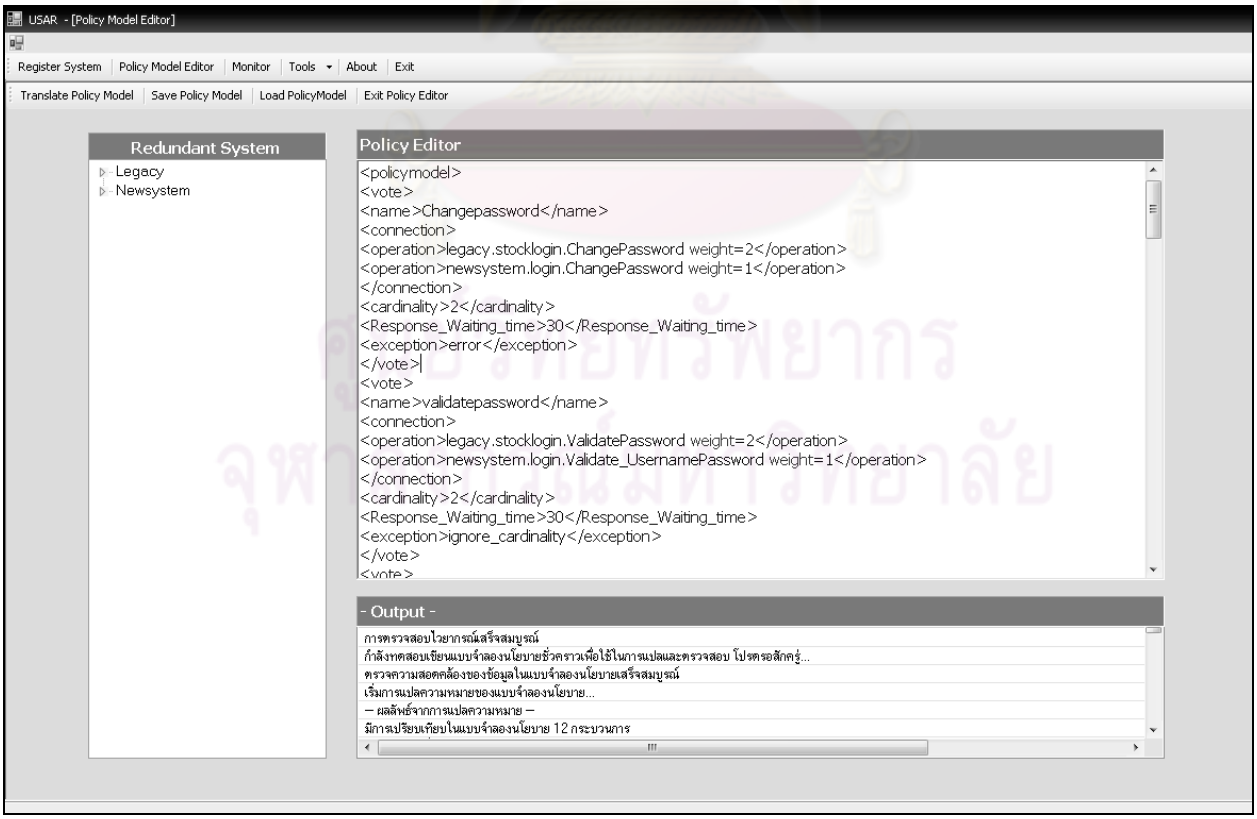


รูปที่ 4.8 การแนะนำคำสั่งด้วยอินเทลลิเซนท์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.9 แสดงการตรวจสอบไวยากรณ์ของนโยบาย



รูปที่ 4.10 แสดงการแปลความหมายจากนโยบาย

4.3.2.3 **ส่วนต่อประสานผู้ใช้สำหรับควบคุมระบบสนับสนุนและตรวจสอบ (user interface for system control and monitoring)** ดังรูปที่ 4.11 เป็นส่วนต่อประสานผู้ใช้ซึ่งภายในประกอบด้วยฟังก์ชันที่ควบคุมการทำงานต่างๆของระบบสนับสนุนให้ทำงานร่วมกัน เช่น การกระจายการร้องขอ การตรวจสอบสถานะฐานข้อมูล การเลือกผลลัพธ์ที่น่าเชื่อถือรวมทั้งการส่งผลลัพธ์ดังกล่าวกลับไปยังไคลเอนท์และรายงานให้กับผู้ดูแลระบบได้ทราบ ผ่านส่วนต่อประสานผู้ใช้ โดยส่วนต่อประสานผู้ใช้สำหรับควบคุมระบบสนับสนุนและตรวจสอบ มีขั้นตอนการทำงาน ดังนี้

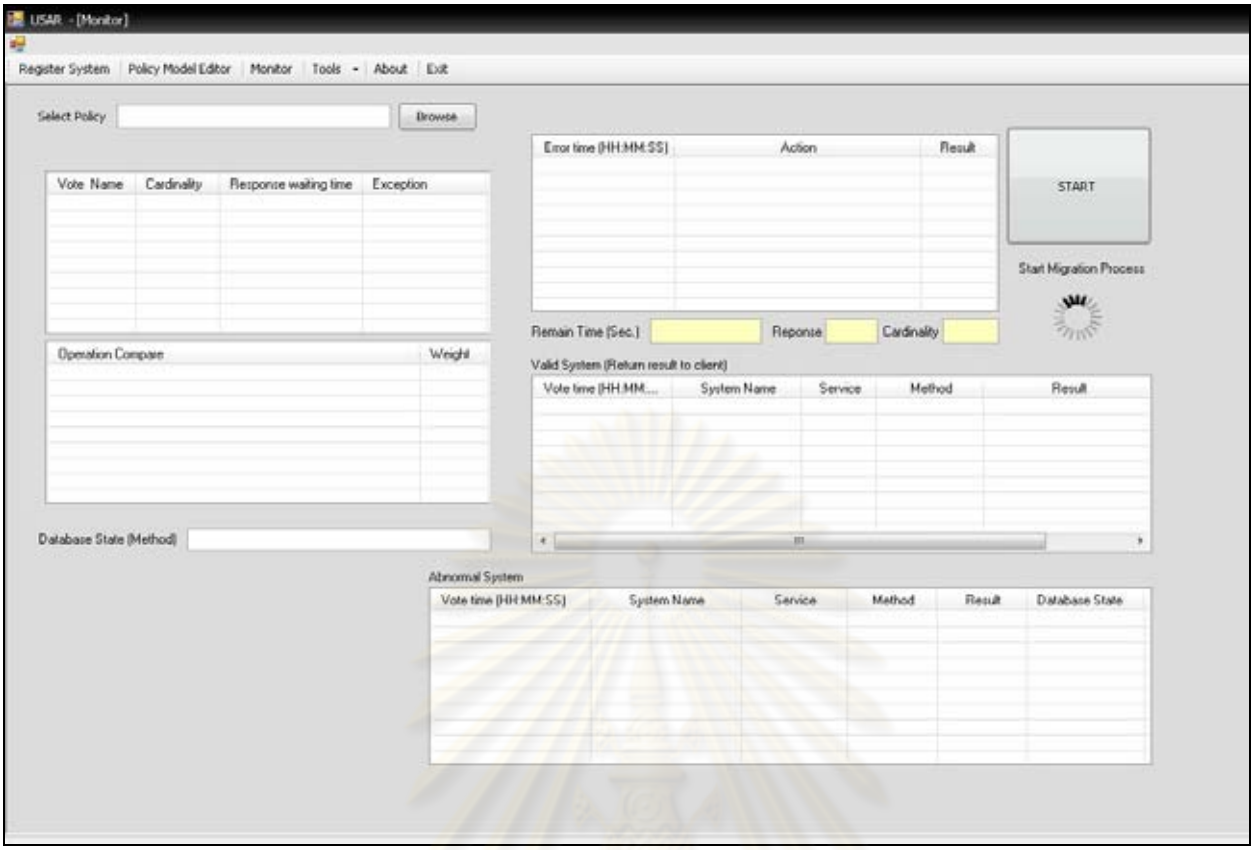
1. ผู้ติดตั้งระบบเลือกนโยบายที่ได้สร้างไว้ โดยกดปุ่ม “Browse”

2. ระบบสนับสนุนจะตรวจสอบความสอดคล้องของข้อมูลระบบภายในนโยบายกับข้อมูลที่ระบบสนับสนุนมี ในกรณีที่มีการเปลี่ยนแปลงข้อมูลของระบบ เช่น การแก้ไขระบบหรือการลบระบบซึ่งทำให้ข้อมูลที่ระบบสนับสนุนเก็บบันทึกนั้นไม่สอดคล้องกับข้อมูลที่กำหนดในนโยบาย ระบบสนับสนุนจะแจ้งเตือนให้กับผู้ดูแลระบบได้ทราบ

3. ระบบสนับสนุนจะแปลความหมายจากนโยบายและแสดงการแปลดังกล่าวให้ผู้ดูแลระบบได้ทราบ ให้ผู้แลระบบกดปุ่ม “Start” ระบบสนับสนุนจะเริ่มรอการร้องขอจากไคลเอนท์

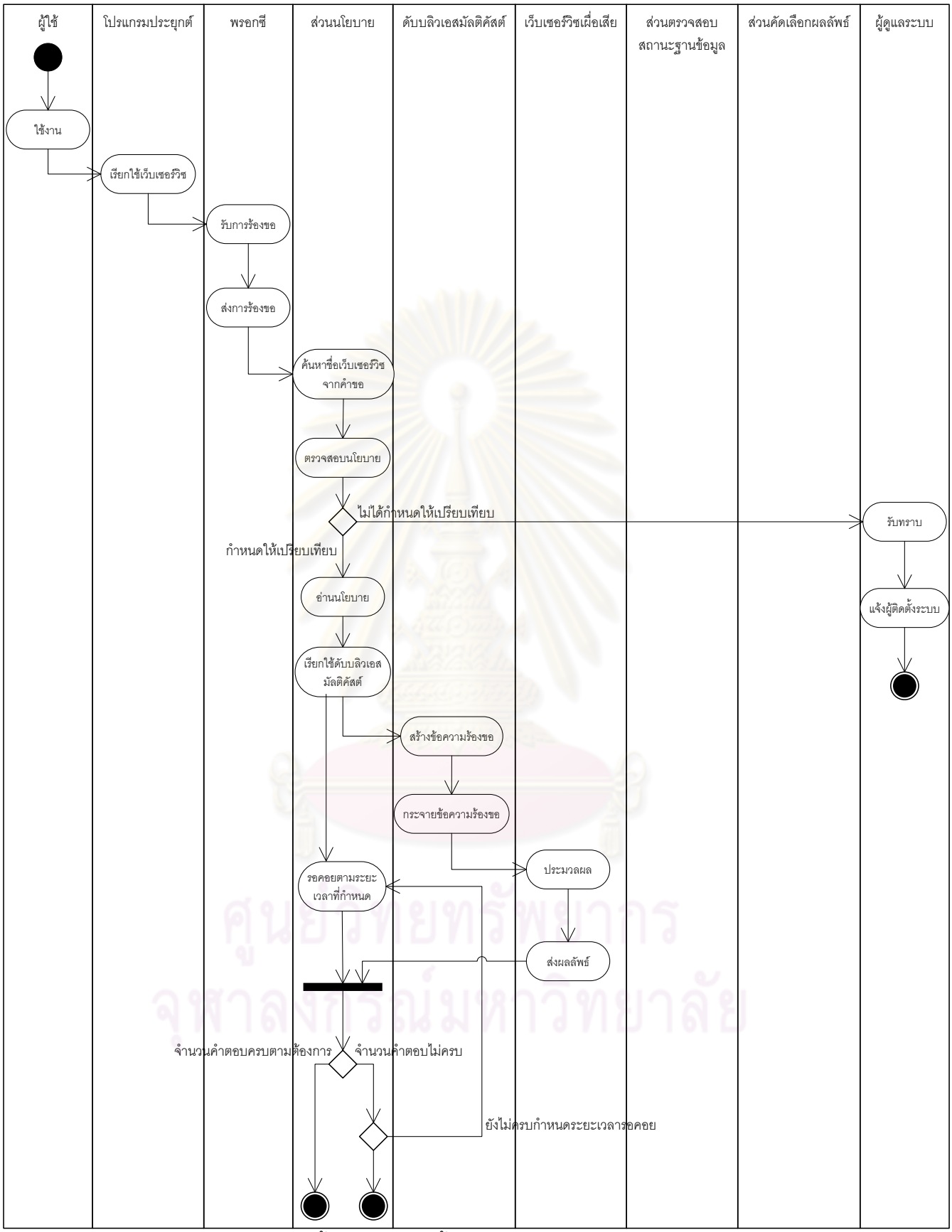
การทำงานทั้งหมดของระบบสนับสนุนเมื่อการร้องขอมาถึงได้แสดงเป็น Sequence diagram ไว้ดังรูปที่ 4.12

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

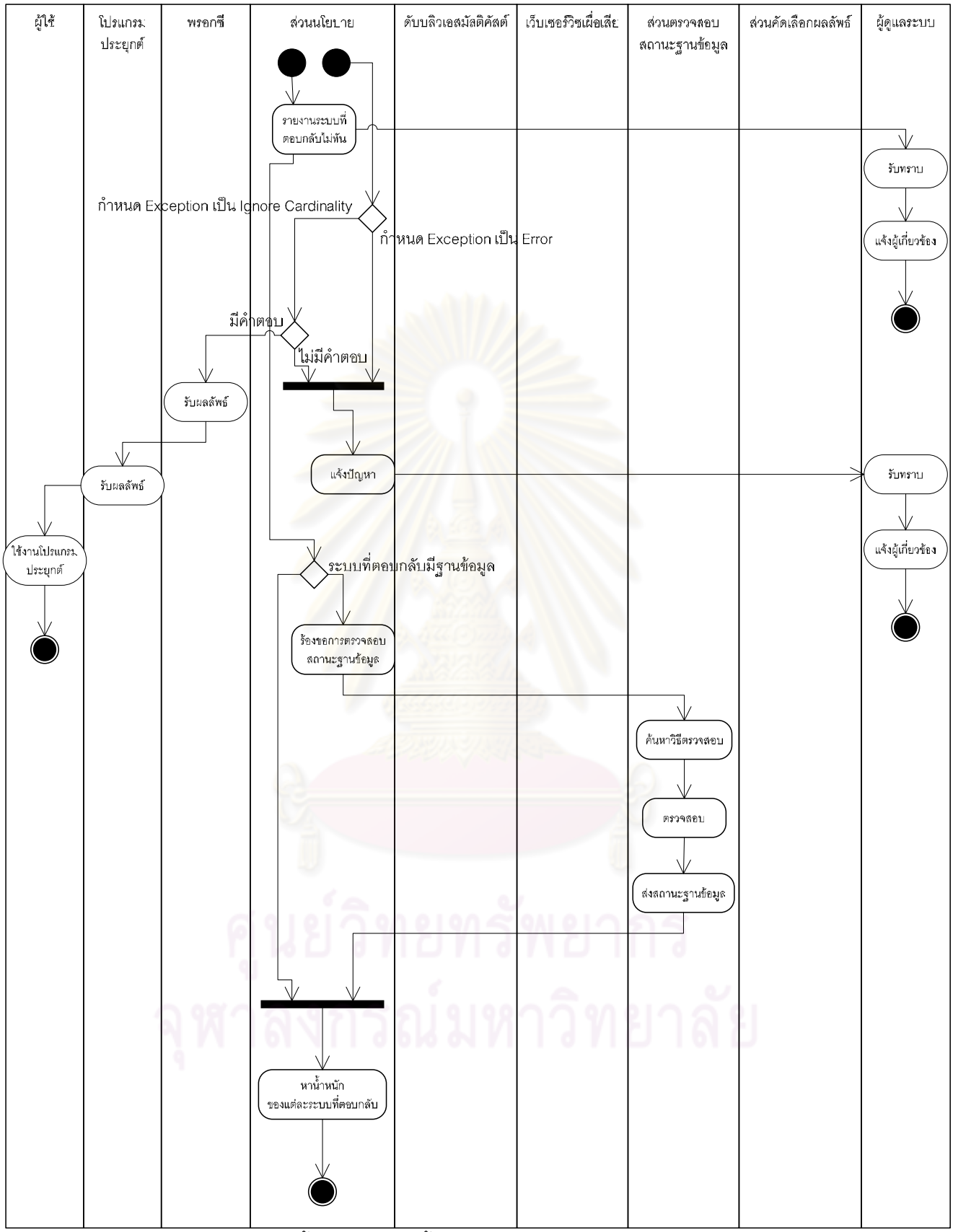


รูปที่ 4.11 ส่วนต่อประสานผู้ใช้สำหรับควบคุมระบบสนับสนุน

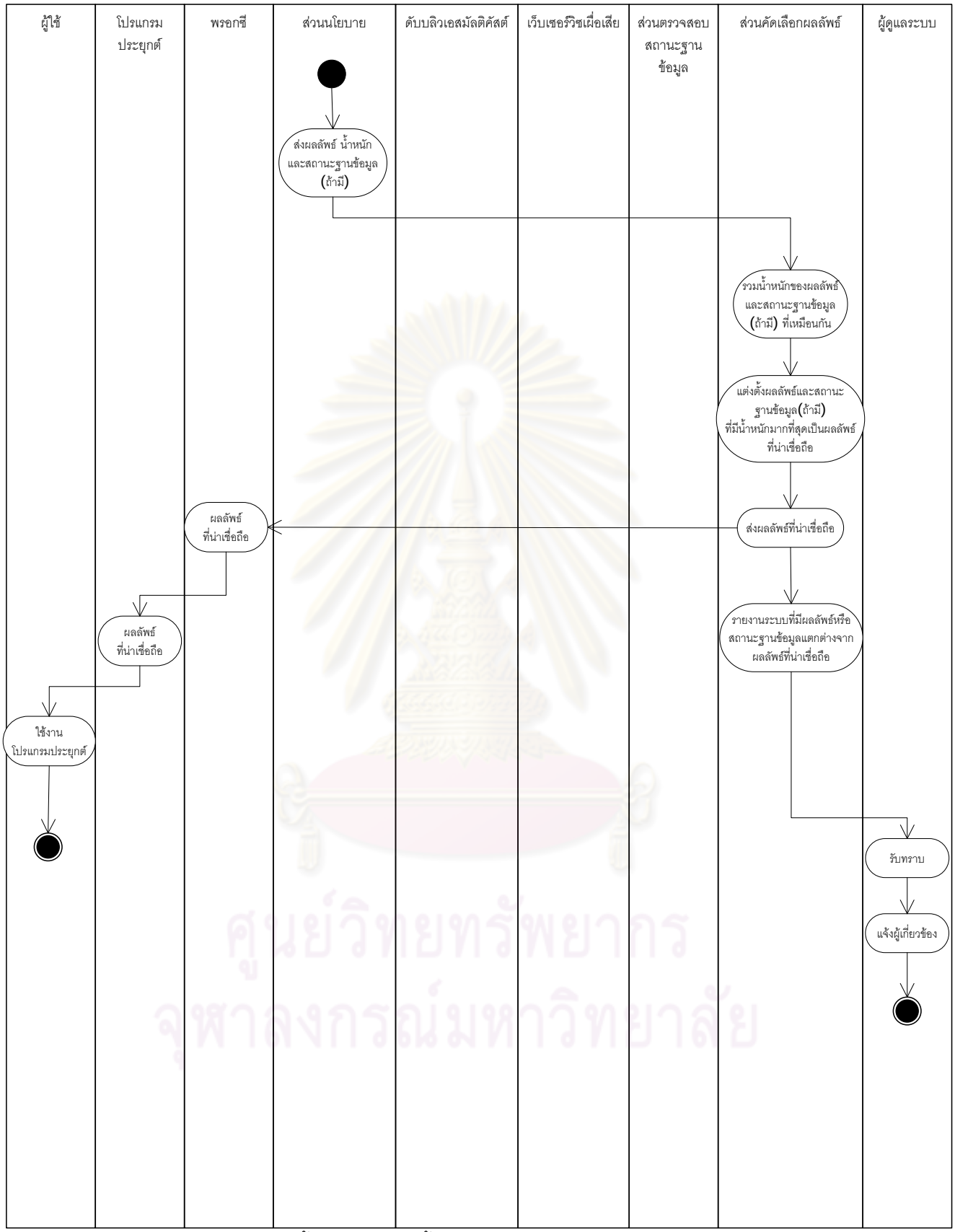
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.12 ขั้นตอนทำงานทั้งหมดของระบบสนับสนุน



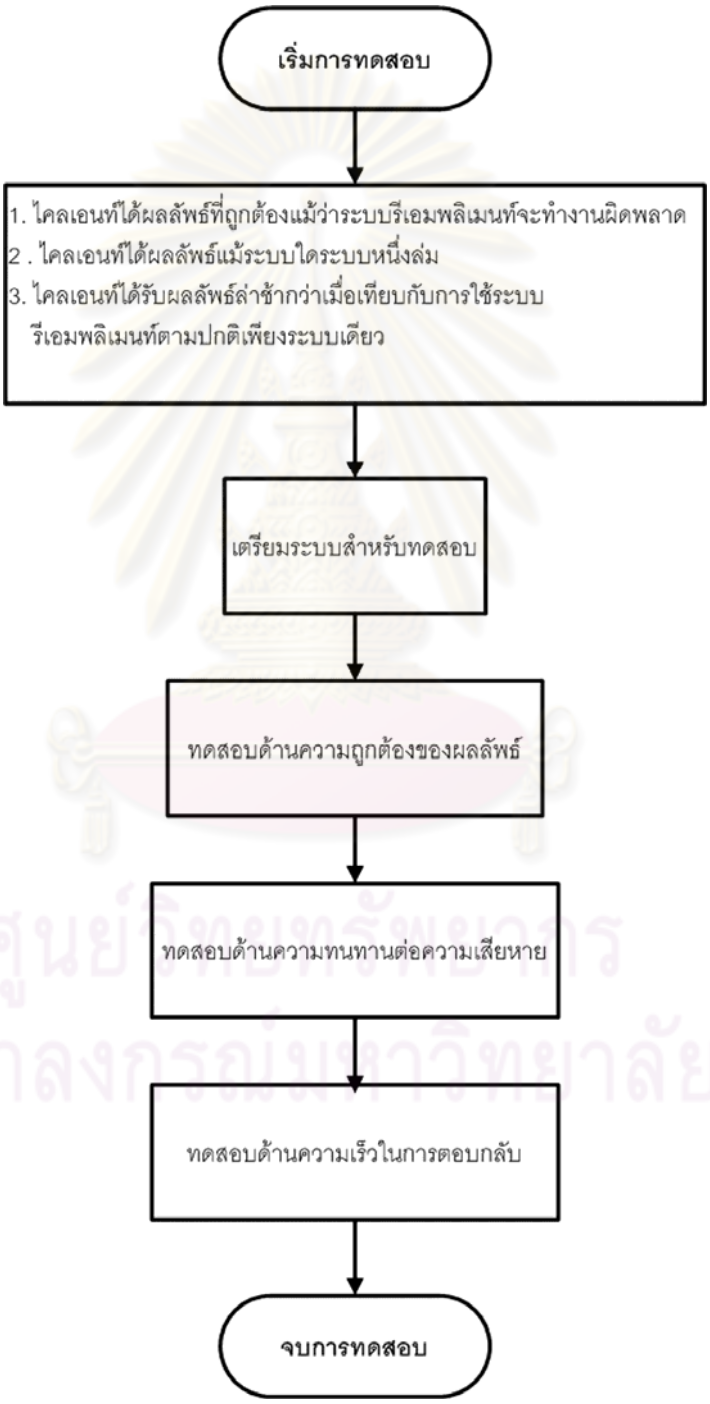
รูปที่ 4.12 ขั้นตอนทำงานทั้งหมดของระบบสนับสนุน (ต่อ)



รูปที่ 4.12 ขั้นตอนทำงานทั้งหมดของระบบสนับสนุน (ต่อ)

4.4 การทดสอบงานวิจัย

ในขั้นตอนการทดสอบนี้ ผู้วิจัยจะนำระบบสนับสนุนที่ได้ออกแบบไว้มาทดสอบร่วมกับระบบเลกาซีและระบบใหม่ที่ได้เตรียมไว้ โดยจะนำเสนอผลการทดสอบเป็นไปตามภาพรวมการทดสอบงานวิจัยตามรูปที่ 4.13 ดังที่จะกล่าวถึงต่อไป



รูปที่ 4.13 ภาพรวมการทดสอบงานวิจัย

4.4.1 การตั้งสมมติฐาน

สมมติฐานคือ ไคลเอนท์ได้ผลลัพธ์ที่ถูกต้องแม้ว่าระบบรีเอมพลิเมนท์จะทำงานผิดพลาด และไคลเอนท์สามารถยังคงได้รับผลลัพธ์แม้ว่าระบบใดระบบหนึ่งล่ม และไคลเอนท์ได้รับผลลัพธ์ล่าช้าเมื่อเทียบกับการใช้ระบบรีเอมพลิเมนท์ตามปกติเพียงระบบเดียว

4.4.2 การเตรียมระบบเพื่อทดสอบ

เนื่องจากงานวิจัยนี้มุ่งเน้นที่จะทำให้ระบบที่พัฒนาขึ้นใหม่นั้นสามารถทำงานทดแทนระบบเลกาซีเดิม ดังนั้นในการทดสอบจึงจำเป็นต้องมีการเชื่อมต่อกับระบบภายนอกอีกสองระบบนั่นคือ ระบบเลกาซีและระบบรีเอมพลิเมนท์เข้ากับระบบสนับสนุนก่อนจึงจะเริ่มกระบวนการทดสอบได้ โดยมีขั้นตอนการเตรียมระบบเพื่อทดสอบ ดังต่อไปนี้

1. เตรียมระบบเลกาซี ระบบเลกาซีที่ผู้วิจัยนำมาเป็นต้นแบบนั้นเป็นโปรแกรมประยุกต์ทางด้านการเงินที่ใช้งานจริงของธนาคารพานิชย์แห่งหนึ่งที่ทำหน้าที่ในการบันทึกการรับเช็คเดินทาง (travel cheque) มาจากตัวแทน และบันทึกการส่งเช็คเดินทางที่รับมาจากตัวแทนนั้นไปยังสาขาต่างๆ เนื่องจากฟังก์ชันที่ใช้งานจริงภายในโปรแกรมประยุกต์นี้มีมากมาย ดังนั้นในการทดสอบนี้ผู้วิจัยจึงได้ตัดฟังก์ชันบางส่วนออกไปเพื่อลดความซับซ้อนในการทดสอบลง

ระบบเลกาซีที่นำมาเป็นต้นแบบนี้ถูกพัฒนาด้วยชุดเครื่องมือไมโครซอฟต์วิชวลเบสิก รุ่น 6 (Microsoft Visual Basic version 6.0) โดยผู้วิจัยได้ย้ายฐานข้อมูลของระบบเลกาซีจากเดิมที่ใช้ไมโครซอฟต์เอสคิวแอล 2000 เซิฟเวอร์ เป็นไมโครซอฟต์เอสคิวแอล 2005 เซิฟเวอร์ โดยผู้วิจัยได้นำรหัสต้นฉบับบางส่วนในส่วนฟังก์ชันของระบบเลกาซีดังกล่าวมาแปลความหมายให้อยู่ในรูปของเอกสารคลังโปรแกรม (dll file) โดยกำหนดให้ระบบเลกาซีที่นำมาทดสอบนั้นประกอบด้วยฟังก์ชันการทำงาน ดังแสดงใน ตารางที่ 4.2 โดยได้กำหนดให้ฐานข้อมูลของระบบเลกาซีมีจำนวนระเบียนเริ่มต้นของแต่ละตาราง ดังแสดงในตารางที่ 4.3

ตารางที่ 4.2 ฟังก์ชันของระบบเลกาซี

เอกสารคลังโปรแกรม	ชื่อฟังก์ชัน	คำอธิบายฟังก์ชัน
Login.dll	Validatepassword	ตรวจสอบชื่อผู้ใช้และรหัสผ่าน
	CheckLevel	ตรวจสอบระดับของผู้ใช้
	ChangePassword	เปลี่ยนรหัสผ่านของผู้ใช้
Setting.dll	RegisterUser	เพิ่มผู้ใช้เข้าสู่ระบบ
	Edit_User	แก้ไขข้อมูลผู้ใช้

ตารางที่ 4.2 ฟังก์ชันของระบบเลกาซี (ต่อ)

เอกสารคลังโปรแกรม	ชื่อฟังก์ชัน	คำอธิบายฟังก์ชัน
	Delete_User	ลบข้อมูลผู้ใช้
	IsDupUserID	ตรวจสอบการซ้ำของชื่อผู้ใช้
Transaction.dll	SendChequetoBranch	ส่งเช็คเดินทางไปยังสาขา
	ReceivefromAgency	รับเช็คเดินทางมาจากตัวแทน
	IsValidCheque	ตรวจสอบเช็คเดินทางภายในคลังกลาง
	Isdupcheque	ตรวจสอบการซ้ำของเช็คเดินทางเมื่อรับมาจากตัวแทน
Inquiry.dll	InquiryCheque	ค้นหาเช็คเดินทาง

ตารางที่ 4.3 จำนวนระเบียนแต่ละตารางภายในฐานข้อมูลของระบบเลกาซี

ชื่อตาราง	จำนวนระเบียน
Branch	4
Currency	3
Deno	9
Login	5
Agency	2
Transaction	10,000

2.สร้างระบบรีเอมพลิเมนท์ ระบบรีเอมพลิเมนท์นั้นผู้วิจัยได้มุ่งหวังที่จะให้ระบบดังกล่าวทำงานได้เหมือนกับระบบเลกาซีเดิม ดังนั้นผู้วิจัยจึงพัฒนาระบบรีเอมพลิเมนท์ให้มีฟังก์ชันเหมือนกับระบบเลกาซีเดิมแต่ให้ทำงานด้วยเทคโนโลยีเว็บเซอร์วิซ

โดยระบบรีเอมพลิเมนท์ดังกล่าวถูกพัฒนาด้วยชุดเครื่องมือไมโครซอฟต์วิชวลสตูดิโอ รุ่น 2005 และใช้ไมโครซอฟต์เอสคิวแอล 2005 เซิฟเวอร์ เป็นระบบจัดการฐานข้อมูล โดยผู้วิจัยได้นำเอารหัสต้นฉบับส่วนของฟังก์ชันมารีเอมพลิเมนท์ให้อยู่ในรูปของเว็บเซอร์วิซ ส่วนรหัสต้นฉบับส่วนของการแสดงผลนั้นผู้วิจัยได้รีเอมพลิเมนท์ให้อยู่ในรูปของเว็บแอปพลิเคชัน โดยระบบรีเอมพลิเมนท์ประกอบด้วยฟังก์ชันการทำงานในรูปของเว็บเซอร์วิซทั้งหมด ดังแสดงใน ตารางที่ 4.4 และกำหนดให้ฐานข้อมูลของระบบรีเอมพลิเมนท์มีจำนวนระเบียนของแต่ละตารางเท่ากันกับฐานข้อมูลของระบบเลกาซี ดังแสดงไว้ในตารางที่ 4.3

ตารางที่ 4.4 ฟังก์ชันของระบบรีเอมพลิเมนท์

ชื่อเว็บเซอร์วิซ	ชื่อเมท็อด	คำอธิบายเมท็อด
Login.asmx	Validatepassword	ตรวจสอบชื่อผู้ใช้และรหัสผ่าน
	CheckLevel	ตรวจสอบระดับของผู้ใช้
	ChangePassword	เปลี่ยนรหัสผ่านของผู้ใช้
Setting.asmx	RegisterUser	เพิ่มผู้ใช้เข้าสู่ระบบ
	Edit_User	แก้ไขข้อมูลผู้ใช้
	Delete_User	ลบข้อมูลผู้ใช้
	IsDupUserID	ตรวจสอบการซ้ำของชื่อผู้ใช้
Transaction.asmx	SendChequetoBranch	ส่งเช็คเดินทางไปยังสาขา
	ReceivefromAgency	รับเช็คเดินทางมาจากตัวแทน
	IsValidCheque	ตรวจสอบเช็คเดินทางภายในคลังกลาง
	Isdupcheque	ตรวจสอบการซ้ำของเช็คเดินทางเมื่อรับมาจากตัวแทน
Inquiry.asmx	InquiryCheque	ค้นหาเช็คเดินทาง

3.ห่อระบบเลกาซี การทำให้ระบบเลกาซีสามารถเปรียบเทียบผลลัพธ์กับระบบรีเอมพลิเมนท์ได้นั้นต้องทำให้ระบบเลกาซีอยู่บนมาตรฐานเดียวกับระบบรีเอมพลิเมนท์ก่อนด้วยตัวห่อ โดยในการเลือกใช้ตัวห่อนั้นจะขึ้นอยู่กับลักษณะระบบเลกาซีที่นำมาห่อ เนื่องจากตัวห่อแต่ละชนิดนั้นมีข้อจำกัดในการห่อระบบเลกาซีที่แตกต่างกัน ซึ่งในการทดสอบนี้ ผู้วิจัยได้เลือกใช้ชุดเครื่องมือโซป รุ่น 2 ซึ่งเป็นตัวห่อที่เหมาะสมกับระบบเลกาซีที่นำมาทดสอบเนื่องจากฟังก์ชันของระบบเลกาซีดังกล่าวอยู่ในรูปเอกสารคลังโปรแกรม

4. ลงทะเบียนเว็บเซอร์วิซของระบบเลกาซีกับระบบสนับสนุน ผู้วิจัยได้ห่อฟังก์ชันของระบบเลกาซีให้กลายเป็นเว็บเซอร์วิซแล้ว อย่างไรก็ตามการทำให้ระบบสนับสนุนสามารถเปรียบเทียบเว็บเซอร์วิซของระบบรีเอมพลิเมนท์ใหม่ได้นั้น จำเป็นต้องทำให้ระบบสนับสนุนรู้วิธีการเรียกใช้งานเว็บเซอร์วิซของทั้งสองระบบก่อน โดยการนำเอกสารดับบลิวเอสดีแอลของระบบเลกาซีที่ได้จากข้อ 3 มาลงทะเบียนกับระบบสนับสนุน โดยในการทดสอบนี้ผู้วิจัยได้ตั้งชื่อให้กับระบบเลกาซีที่มาลงทะเบียนว่า Legacy เพื่อใช้อ้างอิงในขั้นตอนการกำหนดนโยบาย เนื่องจากระบบเลกาซีที่นำมาทดสอบนั้นมีการใช้งานฐานข้อมูล ซึ่งผู้วิจัยต้องการทดสอบสถานะฐานข้อมูลด้วย จึงระบุข้อมูลเกี่ยวกับฐานข้อมูลของระบบเลกาซีให้กับระบบสนับสนุน และระบุวิธีการตรวจสอบสถานะฐานข้อมูล ในกรณีที่ระบบจัดการฐานข้อมูลของทั้งสองระบบ มีฟังก์ชัน

การสร้างตารางติดตาม สามารถเลือกตรวจสอบสถานะฐานข้อมูลด้วยตารางติดตามหรือด้วยตัวห่อฐานข้อมูล ในกรณีที่ระบบจัดการฐานข้อมูลของระบบใดระบบหนึ่งไม่มีฟังก์ชันการสร้างตารางติดตาม จะสามารถเลือกตรวจสอบสถานะฐานข้อมูลด้วยตัวห่อฐานข้อมูล ได้วิธีเดียวเท่านั้น โดยการเลือกวิธีการตรวจสอบสถานะฐานข้อมูลของทั้งสองระบบนั้นจำเป็นต้องเลือกให้เหมือนกัน เพื่อให้สามารถเปรียบเทียบสถานะฐานข้อมูลระหว่างกันได้ โดยในการทดสอบนี้ทั้งระบบเลกาซีและระบบรีเอมพลิเมนท์ใช้ระบบจัดการฐานข้อมูลเป็นไมโครซอฟต์เอสคิวแอลเซิฟเวอร์ ซึ่งมีฟังก์ชันการสร้างตารางติดตามชื่อไมโครซอฟต์เอสคิวแอลโปรไฟล์เออร์ (microsft sql profiler) ดังนั้นผู้วิจัยจึงสามารถทดสอบวิธีในการหาสถานะฐานข้อมูลได้ทั้งสองวิธี โดยได้แสดงผลการทดสอบเมื่อใช้วิธีการตรวจสอบสถานะฐานข้อมูลที่แตกต่างกัน ไว้ดังตารางที่ 4.7

5. <u>ลงทะเบียนเว็บเซอร์วิซของระบบรีเอมพลิเมนท์กับระบบสนับสนุน</u> เมื่อลงทะเบียนเว็บเซอร์วิซของระบบเลกาซีเข้ากับระบบสนับสนุนแล้ว ขั้นตอนต่อมาคือการนำเว็บเซอร์วิซระบบที่รีเอมพลิเมนท์ขึ้นมาใหม่มาลงทะเบียนเข้าสู่ระบบสนับสนุนเพื่อใช้เปรียบเทียบกับระบบเลกาซี โดยการนำเอกสารดับบลิวเอสดีแอลมาลงทะเบียนกับระบบสนับสนุน โดยในการทดสอบนี้ผู้วิจัยได้ตั้งชื่อให้กับระบบรีเอมพลิเมนท์ว่า *Newsystem* เพื่อใช้อ้างอิงในขั้นตอนการกำหนดนโยบาย และระบุข้อมูลฐานข้อมูลของระบบรีเอมพลิเมนท์พร้อมกับเลือกวิธีการหาสถานะฐานข้อมูลซึ่งเหมือนกับระบบเลกาซี

6. <u>กำหนดนโยบาย</u> เมื่อลงทะเบียนเว็บเซอร์วิซของระบบเลกาซีและรีเอมพลิเมนท์เข้าสู่ระบบสนับสนุนแล้ว ขั้นตอนต่อมาคือการทำให้ระบบสนับสนุนสามารถทราบถึงแนวทางการคัดเลือกผลลัพธ์ส่งให้กับไคลเอนท์ โดยใช้การกำหนดผ่านนโยบาย ซึ่งในการทดสอบนี้ผู้วิจัยต้องการเปรียบเทียบฟังก์ชันของระบบเลกาซีและระบบรีเอมพลิเมนท์รวมทั้งสิ้น 12 ฟังก์ชัน เนื่องจากผู้วิจัยได้ใช้ระบบเลกาซีเป็นต้นแบบ จึงได้กำหนดให้ทุกฟังก์ชันของระบบเลกาซีมีน้ำหนักมากกว่าระบบรีเอมพลิเมนท์ และกำหนดเวลาในการรอคอยที่ส่วน Response_waiting_time เท่ากับ 90 วินาทีและกำหนดจำนวนคำตอบที่ต้องการในการคัดเลือกผลลัพธ์เท่ากับ 2 ในกรณีที่ระบบใดระบบหนึ่งเสียหายทำให้จำนวนคำตอบไม่ครบทำให้ไม่สามารถคัดเลือกผลลัพธ์ได้ ผู้วิจัยกำหนดให้ระบบส่งคำตอบที่ระบบสนับสนุนมีส่งให้กับไคลเอนท์ก่อนเพื่อไม่ให้กระทบกับธุรกรรมขององค์กรจึงกำหนดในส่วน exception เป็น Ignore_Cardinality โดยผู้วิจัยได้กำหนดนโยบายทั้งหมดสำหรับทดสอบงานวิจัยนี้ไว้ ดังนี้

```
<PolicyModel >
 <Vote>
    <Name>Validate_password</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function.Validatepassword weight=2</operation>
        <operation>New_system.WSStockTC.Validatepassword  weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
 </Vote>
```

รูปที่ 4.14 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบชื่อผู้ใช้และรหัสผ่าน

```
<Vote>
   <Name>Check_Level</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function.CheckLevel weight=2</operation>
        <operation>New_system.WSStockTC.CheckLevel weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.15 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบระดับของผู้ใช้

```
<Vote>
   <Name>Change_Password</Name >
   <Connection>
        <operation>Legacy_system.StockTC_function.ChangePassword weight=2</operation>
        <operation>New_system.WSStockTC.ChangePassword weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.16 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันเปลี่ยนรหัสผ่านของผู้ใช้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
<Vote>
    <Name>RegisterUser</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function.RegisterUser weight=2</operation>
        <operation>New_system.WSStockTC.RegisterUser weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.17 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันเพิ่มผู้ใช้เข้าสู่ระบบ

```
<Vote>
   <Name>Edit_User</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function. Edit_User weight=2</operation>
        <operation>New_system.WSStockTC. Edit_User weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.18 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันแก้ไขข้อมูลผู้ใช้

```
<Vote>
   <Name>Delete_User</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function.Delete_User weight=2</operation>
        <operation>New_system.WSStockTC.Delete_User weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.19 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันลบข้อมูลผู้ใช้

```
<Vote>
    <Name>IsDupUserID</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function.IsDupUserID weight=2</operation>
        <operation>New_system.WSStockTC.IsDupUserID weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.20 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบการซ้ำของชื่อผู้ใช้

```
<Vote>
   <Name>SendChequetoBranch</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function.SendChequetoBranch weight=2</operation>
        <operation>New_system.WSStockTC.SendChequetoBranch weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.21 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันส่งเช็คเดินทางไปยังสาขา

```
<Vote>
    <Name>ReceivefromAgency</Name>
   <Connection>
        <operation>Legacy_system.StockTC_function.ReceivefromAgency  weight=2</operation>
        <operation>New_system.WSStockTC.ReceivefromAgency  weight=1</operation>
   </Connection>
   <Cardinality>2</Cardinality>
   <Response_waiting_time>90</Response_waiting_time>
   <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.22 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันรับเช็คเดินทางมาจากตัวแทน

```
<Vote>
  <Name>IsValidCheque</Name>
  <Connection>
      <operation>Legacy_system.StockTC_function.IsValidCheque weight=2</operation>
      <operation>New_system.WSStockTC.IsValidCheque weight=1</operation>
  </Connection>
  <Cardinality>2</Cardinality>
  <Response_waiting_time>90</Response_waiting_time>
  <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.23 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบเช็คเดินทางภายในคลังกลาง

```
<Vote>
   <Name>IsdupCheque</Name>
  <Connection>
      <operation>Legacy_system.StockTC_function.IsdupCheque weight=2</operation>
      <operation>New_system.WSStockTC.IsdupCheque weight=1</operation>
  </Connection>
  <Cardinality>2</Cardinality>
  <Response_waiting_time>90</Response_waiting_time>
  <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.24 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันตรวจสอบการซ้ำของเช็คเดินทาง

```
<Vote>
   <Name>InquiryCheque</Name>
  <Connection>
      <operation>Legacy_system.StockTC_function.InquiryCheque  weight=2</operation>
      <operation>New_system.WSStockTC.InquiryCheque  weight=1</operation>
  </Connection>
  <Cardinality>2</Cardinality>
  <Response_waiting_time>90</Response_waiting_time>
  <Exception>Ignore_Cardinality</Exception>
</Vote>
```

รูปที่ 4.25 นโยบายการคัดเลือกผลลัพธ์ของฟังก์ชันค้นหาเช็คเดินทาง

4.4.3. ผลการทดสอบ

เมื่อเตรียมระบบทดสอบเสร็จสิ้นแล้ว ผู้วิจัยจึงเริ่มทดสอบโดยได้แบ่งการทดสอบออกเป็นด้านต่างๆ ดังนี้

4.4.3.1 **การทดสอบด้านความถูกต้องของผลลัพธ์** ในส่วนของการทดสอบด้านความถูกต้องของผลลัพธ์นั้น ผู้วิจัยต้องการทดสอบว่า หากระบบใหม่ทำงานผิดพลาดจะสามารถใช้ผลลัพธ์ที่ถูกต้องของระบบเลกาซีมาช่วยสนับสนุนได้หรือไม่ ผู้วิจัยจึงกำหนดให้ระบบใหม่ทำงานผิดพลาดในบางฟังก์ชัน พบว่าเมื่อส่งข้อมูลทดสอบเข้าไปในแต่ละฟังก์ชันของระบบเลกาซีจะได้ผลลัพธ์ตรงตามที่คาดหวังไว้ แต่เมื่อส่งข้อมูลทดสอบเข้าไปในแต่ละฟังก์ชันของระบบใหม่ จะได้ผลลัพธ์ไม่ตรงตามที่คาดหวังไว้บางฟังก์ชัน แต่เมื่อนำมาใช้งานร่วมกับระบบสนับสนุนได้ผลลัพธ์เหมือนกับระบบเลกาซีซึ่งเป็นผลลัพธ์ที่ถูกต้อง ดังที่ได้สรุปผลไว้ในตารางที่ 4.5

ตารางที่ 4.5 ผลการทดสอบด้านความถูกต้องของผลลัพธ์การทำงาน

เมท็อด	ข้อมูลทดสอบ	ผลลัพธ์		
		ระบบเลกาซี	ระบบรีเอมพลิเมนท์	เมื่อใช้งานร่วมกับระบบสนับสนุน
Validatepassword	Bbluser,bblpwd	True	False	True
CheckLevel	Bbluser	01	02	01
ChangePassword	Bbluser,bblpwd2	1	0	1
RegisterUser	Bbluser_n,01	1	1	1
EditUser	Bbluser_n,Administrator	1	1	1
DeleteUser	Bbluser	1	1	1
IsDupUserID	BblUser_n	False	True	False
SendChequetoBranch	Dc,1,100,0053	1	1	1
ReceivefromAgency	Interpayment,dc,1,100,Usd,50	1	1	1
IsDupCheque	Dc,31,1	False	True	False
IsValidCheque	Dc,1,100	True	False	True
InquiryCheque	Dc80	ท่าอากาศยาน	คลังกลาง	ท่าอากาศยาน

4.4.3.2 **การทดสอบด้านความคงทนต่อความเสียหาย** การทดสอบด้านความคงทนต่อความเสียหายนั้น พบว่าเมื่อเว็บเซอร์วิซของระบบเลกาซีไม่สามารถทำงานได้ แต่เว็บเซอร์วิซของระบบรีเอมพลิเมนท์สามารถใช้งานได้ปกติ ไคลเอนท์ก็ยังสามารถทำงานได้ตามปกติ หรือ

ถ้าในขณะที่เว็บเซอร์วิซของระบบเลกาซีสามารถทำงานได้ตามปกติ แต่เว็บเซอร์วิซของระบบบรีเอมพลิเมนท์ไม่สามารถทำงานได้ ไคลเอนท์ก็ยังสามารถทำงานได้ตามปกติ ดังที่ได้สรุปผลไว้ในตารางที่ 4.6

ตารางที่ 4.6 ผลการทดสอบด้านความคงทนต่อความเสียหาย

เมท็อด	ข้อมูลทดสอบ	ผลลัพธ์		
		ระบบเลกาซี	ระบบบรีเอมพลิเมนท์	เมื่อใช้งานร่วมกับระบบสนับสนุน
Validatepassword	Bbluser,bblpwd	True	-	True
CheckLevel	Bbluser	01	-	01
ChangePassword	Bbluser,bblpwd2	1	-	1
RegisterUser	Bbluser_n,01	1	-	1
Edit_User	Bbluser_n,Admini	1	-	1
Delete_User	Bbluser	1	-	1
IsDupUserID	BblUser_n	True	-	True
SendChequetoBranch	Dc,1,100,0053	1	-	1
ReceivefromAgency	Interpayment,dc,1	1	-	1
IsValidCheque	Dc,1,100	True	-	True
IsDupCheque	Dc,31,1	False	-	False
InquiryCheque	Dc80	-	คลังกลาง	คลังกลาง

4.4.3.3 การทดสอบด้านความเร็วในการตอบผลลัพธ์ การทดสอบด้านความเร็วในการตอบผลลัพธ์นั้น พบว่าเมื่อนำระบบบรีเอมพลิเมนท์มาใช้งานร่วมกับระบบสนับสนุนจะทำให้ใช้เวลาในการตอบกลับมากขึ้นเมื่อเทียบกับการใช้งานระบบบรีอิมพลิเมนท์ตามปกติเพียงระบบเดียว เนื่องจากในการใช้งานร่วมกับระบบสนับสนุนนั้นจะต้องรอทั้งระบบเลกาซีที่เป็นต้นแบบและระบบบรีอิมพลเมนท์ตอบกลับมาก่อนจึงเริ่มกระบวนการสรุปผลลัพธ์ส่งให้กับไคลเอนท์

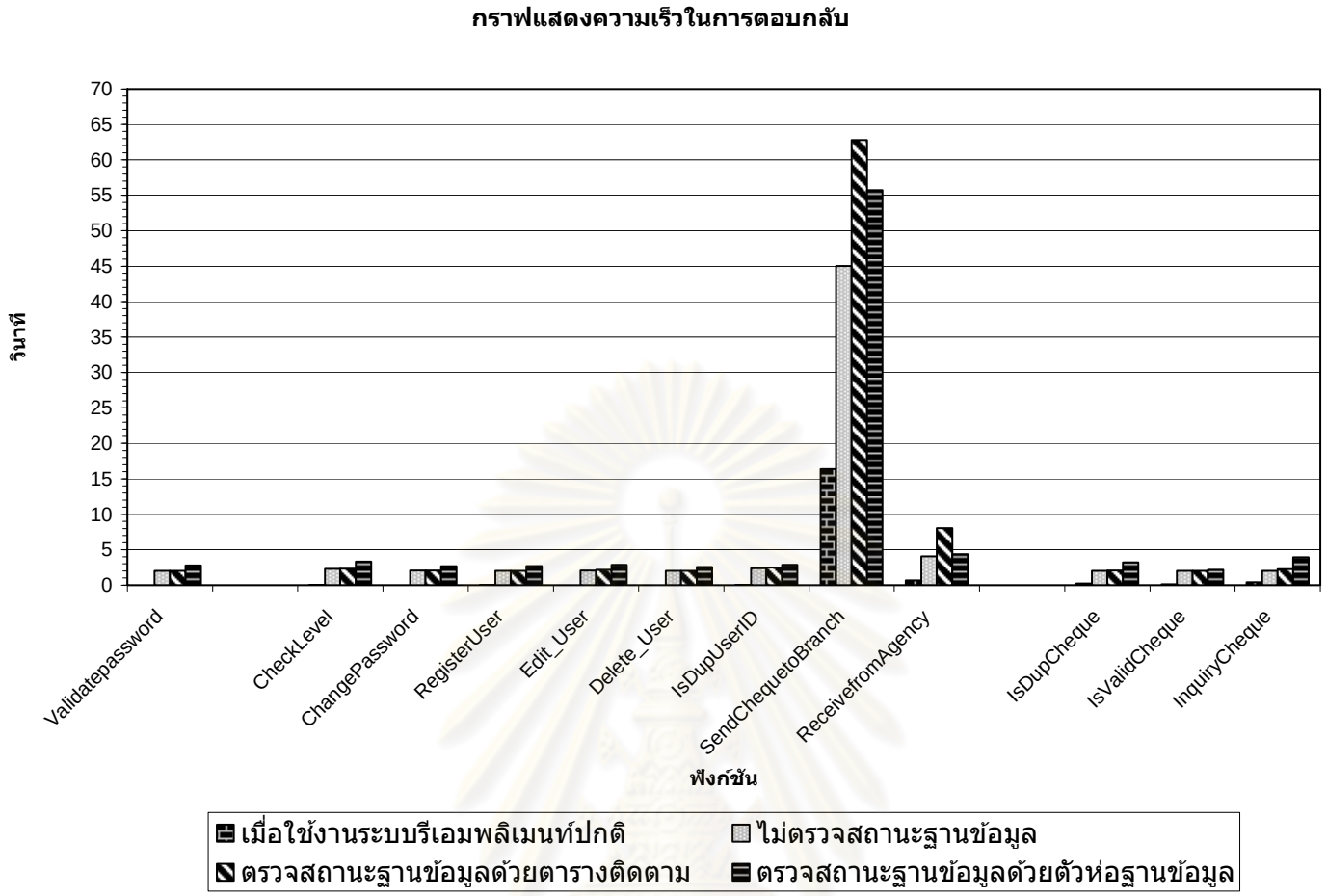
ในกรณีที่มีการตรวจสอบสถานะฐานข้อมูลของระบบบรีเอมพลิเมนท์ร่วมกับระบบเลกาซีนั้น ผู้วิจัยพบว่าหากตรวจสอบสถานะฐานข้อมูลด้วยตัวห่อฐานข้อมูลจะใช้เวลาในการตอบกลับมากขึ้นในทุกฟังก์ชันเมื่อเทียบกับการไม่ตรวจสอบสถานะฐานข้อมูล ส่วนการตรวจสอบด้วยตารางติดตามนั้นแม้ว่าเมื่อทดสอบกับฟังก์ชันส่วนใหญ่ จะใช้เวลาในการตอบกลับน้อยกว่าการตรวจสอบด้วยตัวห่อฐานข้อมูล แต่ผู้วิจัยพบว่าหากฟังก์ชันใดมีการใช้งานคำสั่งจัดการฐานข้อมูลจำนวนมากก็จะทำให้การตรวจด้วยตารางติดตามใช้เวลาในการตอบกลับมากกว่าการตรวจสอบด้วยตัวห่อฐานข้อมูล ยกตัวอย่างเช่น ฟังก์ชัน ReceivefromAgency ซึ่งเป็นฟังก์ชันสำหรับบันทึกการ

รับตั๋วเดินทางมาจากตัวแทน ซึ่งในกรณีที่รับตั๋วเดินทางมาทั้งหมด n ใบ เมื่อบันทึกข้อมูลตั๋วเดินทางลงในฐานข้อมูลของแต่ละระบบแล้ว ระบบจัดการฐานข้อมูลยังต้องบันทึกคำสั่งจัดการฐานข้อมูลดังกล่าวทั้งหมด n คำสั่งลงในตารางติดตามด้วย โดยผู้วิจัยได้สรุปผลการทดสอบด้านความเร็วในการตอบกลับแยกตามกรณี ไว้ในตารางที่ 4.7 และนำมาวาดกราฟดังรูปที่ 4.26

ตารางที่ 4.7 ผลการทดสอบด้านความเร็วในการตอบกลับ

เมท็อด	ข้อมูลทดสอบ	ความเร็วในการตอบผลลัพธ์ (วินาที)			
		เมื่อใช้งานระบบรีเอมพลิเมนท์ปกติ	เมื่อใช้งานร่วมกับระบบสนับสนุน		
			ไม่ตรวจสถานะฐานข้อมูล	ตรวจสถานะฐานข้อมูลด้วยตารางติดตาม	ตรวจสถานะฐานข้อมูลด้วยตัวห่อฐานข้อมูล
Validatepassword	Bbluser,bblpwd	0.01	2.033	2.033	2.774
CheckLevel	Bbluser	0.04	2.314	2.330	3.315
ChangePassword	Bbluser,bblpwd	0.01	2.053	2.053	2.684
RegisterUser	Bbluser_n,01	0.02	2.043	2.043	2.704
Edit_User	Bbluser_n,Administrator	0.01	2.053	2.169	2.874
Delete_User	Bbluser	0.01	2.033	2.033	2.574
IsDupUserID	BblUser_n	0.05	2.394	2.487	2.874
SendChequetoBranch	Dc,1,1000,0225	16.393	45.015	62.776	55.703
ReceivefromAgency	Interpayment,dc,1,1000	0.671	4.046	8.051	4.356
IsDupCheque	Dc,31,1,100	0.25	2.033	2.063	3.204
IsValidCheque	Dc,1,100	0.121	2.033	2.043	2.161
InquiryCheque	Dc80	0.41	2.023	2.274	3.915

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.26 กราฟแสดงความเร็วในการตอบกลับ

ในบทนี้แสดงให้เห็นถึงการออกแบบระบบสนับสนุนโดยมีฟังก์ชันการทำงานอิงตามโมเดลที่ได้ออกแบบไว้ในบทที่ 3 นอกจากนี้ผู้วิจัยยังได้นำระบบเลกาซีและระบบรีเอมพลิเมนท์มาทดสอบร่วมกับระบบสนับสนุนที่ได้ออกแบบไว้ ซึ่งในบทต่อไปจะเป็นการสรุปผลการทดลองรวมทั้งข้อเสนอแนะในการพัฒนาโมเดลนี้ให้มีประสิทธิภาพมากขึ้นต่อไป

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้มุ่งเน้นการเพิ่มความเชื่อมั่นให้กับการใช้งานระบบบรีเอมพลิเมนท์ที่นำมาทดแทนระบบเลกาซีเดิม ดังนั้นผลลัพธ์ที่ระบบบรีเอมพลิเมนท์ตอบกลับไปยังผู้ใช้จึงเป็นสิ่งสำคัญ ผู้วิจัยจึงได้ทดสอบด้านความถูกต้องของผลลัพธ์ ไว้ดังตารางที่ 4.5 โดยได้ทดสอบไว้ 2 กรณี คือ เมื่อใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียวตามปกติ และนำมาทำงานร่วมกับระบบสนับสนุน โดยเมื่อใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียวนั้นไคลเอนท์จะได้ผลลัพธ์ที่ผิดพลาด โดยที่ไคลเอนท์ไม่สามารถทราบได้ว่าคำตอบที่ได้มานั้นถูกต้องหรือไม่ ซึ่งก่อให้เกิดความเสียหายให้กับธุรกรรมขององค์กร แต่เมื่อนำระบบบรีเอมพลิเมนท์ใช้งานร่วมกับระบบสนับสนุน พบว่าไคลเอนท์ได้รับคำตอบที่ถูกต้องแม้ว่าระบบบรีเอมพลิเมนท์จะทำงานผิดพลาด นอกจากนี้ระบบสนับสนุนยังแสดงชื่อฟังก์ชันของระบบบรีเอมพลิเมนท์ที่ให้ผลลัพธ์แตกต่างจากฟังก์ชันของระบบเลกาซีผ่านทางส่วนต่อประสานผู้ใช้ทำให้ผู้บำรุงรักษาระบบสามารถแก้ไขได้อย่างแม่นยำมาก ดังนั้นในด้านความถูกต้องของผลลัพธ์นั้นผู้วิจัยสามารถสรุปได้ว่า งานวิจัยนี้สามารถช่วยให้การทำงานของระบบบรีเอมพลิเมนท์มีความน่าเชื่อถือมากขึ้นเมื่อเทียบกับการใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียว เนื่องจากมีผลลัพธ์ของระบบเลกาซีมาช่วยสนับสนุน

สิ่งที่ได้จากงานวิจัยนี้นอกเหนือจากความน่าเชื่อถือของผลลัพธ์ คือ ทั้งสองระบบนั้นสามารถทำงานทดแทนกันได้หากระบบใดระบบหนึ่งได้รับความเสียหาย ดังนั้นผู้วิจัยจึงได้ทดสอบด้านความทนทานต่อความเสียหายไว้ดังตารางที่ 4.6 โดยการจำลองการเสียหายด้วยการปิดบริการเซอร์วิซของระบบบรีเอมพลิเมนท์ พบว่าไคลเอนท์ยังคงได้รับคำตอบจากระบบเลกาซีอยู่ และเมื่อผู้วิจัยทดลองปิดบริการเซอร์วิซของระบบเลกาซี พบว่าไคลเอนท์ยังคงได้รับคำตอบจากระบบบรีเอมพลิเมนท์อยู่ ดังนั้นในด้านความถูกต้องของผลลัพธ์นั้นผู้วิจัยสามารถสรุปได้ว่า งานวิจัยนี้สามารถช่วยเพิ่มความทนทานให้กับธุรกรรมของไคลเอนท์แม้ระบบใดระบบหนึ่งจะเกิดความเสียหาย ธุรกรรมของไคลเอนท์ก็ยังคงดำเนินต่อไปได้เนื่องจากมีฟังก์ชันของอีกระบบสำรองอยู่

อย่างไรก็ตามเมื่อนำระบบบรีเอมพลิเมนท์มาใช้งานร่วมกับระบบสนับสนุนแล้วนั้นไคลเอนท์ต้องใช้เวลาในการทำงานมากขึ้นเมื่อเทียบกับการใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียว เนื่องจากต้องรอระบบสนับสนุนสรุปผลลัพธ์ของทั้งสองระบบก่อนจึงจะสามารถส่งคำตอบให้กับไคลเอนท์ได้ โดยผู้วิจัยได้แสดงผลการทดสอบไว้ดังตารางที่ 4.7 โดยผู้วิจัยได้ทดสอบไว้ 4 กรณีทดสอบ ดังนี้

1.<u>การใช้ระบบบรีเอมพลิเมนท์เพียงระบบเดียวตามปกติ</u> พบว่า เมื่อใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียว ไคลเอนท์สามารถรับผลลัพธ์ได้รวดเร็วที่สุด เนื่องจากไม่ต้องรอผลลัพธ์ของระบบเลกาซี

2.<u>การใช้งานร่วมกับระบบสนับสนุนโดยไม่ตรวจสถานะฐานข้อมูล</u> พบว่า ไคลเอนท์ต้องใช้เวลาในการทำงานมากขึ้นเมื่อเทียบกับการใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียว เนื่องจากต้องรอระบบสนับสนุนสรุปผลลัพธ์ของทั้งสองระบบก่อนจึงจะสามารถส่งคำตอบให้กับไคลเอนท์ได้

3.<u>การใช้งานร่วมกับระบบสนับสนุนโดยตรวจสถานะฐานข้อมูลจากตารางติดตาม</u> พบว่าไคลเอนท์ใช้เวลาในการทำงานมากขึ้นเมื่อเทียบกับการใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียว แต่สามารถตอบผลลัพธ์กลับได้รวดเร็วใกล้เคียงกับการไม่ตรวจสถานะฐานข้อมูล ในกรณีที่ฟังก์ชันทดสอบดังกล่าวมีการใช้งานคำสั่งจัดการฐานข้อมูลจำนวนน้อย แต่เมื่อฟังก์ชันทดสอบมีการใช้งานคำสั่งจัดการฐานข้อมูลจำนวนมาก เช่น ฟังก์ชัน *receivefromagency* และ *sendchequetobranch* พบว่าการใช้งานร่วมกับระบบสนับสนุนโดยตรวจสถานะฐานข้อมูลจากตารางติดตามนั้น จะใช้เวลาในการตอบกลับมากที่สุด เนื่องจากฟังก์ชันดังกล่าวมีการใช้งานคำสั่งจัดการฐานข้อมูล *insert* และ *update* เป็นจำนวนมาก ดังนั้นเมื่อระบบจัดการฐานข้อมูลของแต่ละระบบใช้คำสั่งจัดการฐานข้อมูล n คำสั่งกับฐานข้อมูลแล้ว ระบบจัดการฐานข้อมูลของแต่ละระบบยังต้องบันทึกคำสั่งจัดการฐานข้อมูล n คำสั่งลงในตารางติดตามด้วย

3.<u>การใช้งานร่วมกับระบบสนับสนุนโดยตรวจสถานะฐานข้อมูลจากตัวห่อฐานข้อมูล</u> พบว่าวิธีดังกล่าวใช้เวลาตอบผลลัพธ์ได้ล่าช้ามากที่สุดเมื่อเทียบกับอีก 3 กรณีทดสอบข้างต้น เนื่องจากต้องตรวจสอบโดยการเปรียบเทียบตารางจากฐานข้อมูลของแต่ละระบบที่ละตาราง อย่างไรก็ตามวิธีดังกล่าวสามารถประยุกต์ใช้ได้กับทุกระบบจัดการฐานข้อมูลที่โอเลดีบีสามารถเชื่อมต่อได้ โดยสามารถลดความยุ่งยากในการติดตั้งและบำรุงรักษาตารางติดตามที่มีความแตกต่างกันตามแต่ละระบบจัดการฐานข้อมูล

ในการทดสอบด้านความเร็วในการตอบกลับนั้น ผู้วิจัยสามารถสรุปได้ว่า หากใช้งานร่วมกับระบบสนับสนุนแล้ว ไคลเอนท์จะใช้เวลาในการทำงานมากขึ้นเมื่อเทียบกับการใช้งานระบบบรีเอมพลิเมนท์เพียงระบบเดียว

5.2 ปัญหาที่พบจากงานวิจัย

งานวิจัยนี้มีผลการวิจัยดังที่กล่าวมาแล้ว อย่างไรก็ตามงานวิจัยนี้ยังมีข้อด้อยหรือข้อจำกัดของงานวิจัยด้วยดังต่อไปนี้

1. การตรวจสอบสถานะฐานข้อมูลนั้น แต่เดิมผู้วิจัยได้ใช้วิธีการตรวจสอบคำสั่งจัดการฐานข้อมูลจากตารางติดตามเพื่อนำไปค้นหาสถานะฐานข้อมูล อย่างไรก็ตามระบบจัดการฐานข้อมูลต่างๆนั้นถูกสร้างจากหลายผู้ผลิตทำให้มีรูปแบบของตารางติดตามที่แตกต่างกัน เช่น เป็นเอกสาร (file) หรือตาราง ทำให้ต้องออกแบบระบบสนับสนุนอิงไปตามแต่ละระบบจัดการฐานข้อมูล ไม่สามารถออกแบบให้สามารถใช้ได้ทุกระบบจัดการฐานข้อมูลได้ นอกจากนี้บางระบบจัดการฐานข้อมูลยังไม่มีการสร้างตารางติดตาม ผู้วิจัยจึงใช้ตรวจสอบสถานะฐานข้อมูลด้วยวิธีห่อฐานข้อมูล (database wrapper) ซึ่งวิธีดังกล่าวสามารถตรวจสอบสถานะฐานข้อมูลได้ทุกระบบจัดการฐานข้อมูลผ่านโอเลดีบี [12] แต่วิธีดังกล่าวก็ยังมีข้อด้อยอยู่ เนื่องจากวิธีตรวจสอบด้วยการห่อฐานข้อมูลนั้นต้องคัดลอกข้อมูลจากฐานข้อมูลของแต่ละระบบมาเก็บไว้ที่ตัวห่อบนระบบสนับสนุนก่อนจึงจะสามารถตรวจสอบความเปลี่ยนแปลงของสองข้อมูลทั้งสองชุดด้วยเมท็อดของตัวห่อฐานข้อมูลได้ ซึ่งวิธีดังกล่าวจำเป็นต้องใช้ทรัพยากรเครื่องค่อนข้างสูงและทำงานได้ล่าช้า

2. ฐานข้อมูลของระบบที่นำมาเปรียบเทียบจำเป็นต้องเป็นฐานข้อมูลแบบสัมพันธ์และมีโครงสร้างฐานข้อมูลที่เหมือนกันจึงสามารถนำมาเปรียบเทียบกันได้

3. ฟังก์ชันของระบบเลกาซีนั้น เมื่อถูกเปลี่ยนให้อยู่ในสภาพแวดล้อมของเว็บเซอร์วิซแล้ว ความเสถียรในการใช้งานฟังก์ชันของระบบเลกาซีจะลดลงเนื่องจากธรรมชาติของเว็บที่มีความเสถียรต่ำ ถึงแม้ว่าจะมีเว็บเซอร์วิซของระบบรีเอมพลิเมนท์ช่วยทำงานแต่คำตอบที่ได้นั้นอาจผิดพลาดเนื่องจากไม่มีคำตอบของระบบเลกาซีมาสนับสนุน ดังนั้นช่วงเวลาที่ระบบเลกาซีล่มจึงมีความเสี่ยงหากไคลเอนท์มีการทำธุรกรรมที่สำคัญ

4. ระบบเลกาซีที่นำมาทดสอบนั้นประกอบด้วยฟังก์ชันซึ่งมีการส่งค่ากลับมาเป็นชนิดข้อมูลแบบพื้นฐาน (standard type) แต่พบว่าบางระบบเลกาซีมีการส่งชนิดข้อมูลแบบซับซ้อน (complex type) ร่วมด้วย ทำให้ชุดเครื่องมือโซปไม่สามารถห่อได้

5.3 แนวทางการวิจัยต่อ

จากปัญหาที่พบและข้อจำกัดบางประการนั้น ทำให้งานวิจัยชิ้นนี้ยังมีข้อด้อยอยู่บ้าง ดังนั้นควรจะมีการวิจัยเพิ่มเติม ดังต่อไปนี้

1. การตรวจสอบสถานะฐานข้อมูลด้วยวิธีที่สามารถตรวจสอบสถานะฐานข้อมูลได้ทุกระบบจัดการฐานข้อมูลโดยใช้ทรัพยากรระบบที่ต่ำ

2. การเปรียบเทียบสถานะฐานข้อมูลโดยไม่ขึ้นกับโครงสร้างฐานข้อมูล โดยสามารถศึกษาได้จากงานวิจัยของ ทิแรน [20]

3. นิยามไวยากรณ์ของนโยบายให้ซับซ้อนมากขึ้น โดยอาจใช้ไวยากรณ์เชิงควบคุมหรือเชิงการตัดสินใจเพื่อเพิ่มประสิทธิภาพให้กับโมเดลนี้สามารถเผชิญสถานการณ์อื่นๆ ได้

4. นิยามนโยบายให้สามารถรองรับผลลัพธ์ที่มีความแม่นยำสูงเช่น จุดทศนิยม โดยอาจนำวิธีการทำ error threshold มาประยุกต์เพื่อให้การเปรียบเทียบผลลัพธ์นั้นสามารถยืดหยุ่นได้ (inexact matching)

5. ควรมีการบันทึกความผิดปกติที่เกิดขึ้นกับระบบที่นำมาทำงานคู่ขนานกันในรูปของเอกสารหรือรายงานเพื่อให้สามารถตรวจสอบแก้ไขข้อบกพร่องได้ในภายหลัง

จากปัญหา 5 ข้อ ข้างต้น สมควรขยายระบบให้สามารถให้สามารถรับผลลัพธ์ได้โดยตรงจากระบบเลกาซี โดยไม่ต้องผ่านการห่อให้เป็นเว็บเซอร์วิซ ก็จะสามารถลดปัญหาความไม่เสถียรลงได้ โดยการดัดแปลงเช่นนี้จะมีค่าใช้จ่ายสูงแต่ถ้าเป็นระบบวิกฤต (critical systems) ขององค์กรก็น่าจะคุ้มค่าต่อการลงทุน

งานวิจัยนี้ได้ช่วยให้การทำงานของระบบใหม่มีความถูกต้องขึ้น หลังจากที่ได้นำเอาระบบสนับสนุนไปใช้ร่วมกันกับระบบใหม่ รวมทั้งยังมีความทนทานต่อความเสียหายของระบบทั้งระบบเก่า หรือ ระบบใหม่ ระบบสนับสนุนจะยังคงใช้งานได้ตามปกติ ซึ่งช่วยสร้างความเชื่อมั่นในการเปลี่ยนจากระบบเก่าไปเป็นระบบใหม่ให้กับองค์กร แม้ระบบใหม่ทำงานผิดพลาด จะไม่ส่งผลกระทบต่อการทำธุรกรรมขององค์กร รวมทั้งใช้เป็นเครื่องมือในการทดสอบให้กับระบบใหม่ภายใต้สภาพแวดล้อมจริง ทั้งนี้ทำให้ธุรกรรมขององค์กรดำเนินต่อไปได้ แม้ระบบใดระบบหนึ่งจะล่มก็ตาม

รายการอ้างอิง

[1] Bisbal, J., Lawless, D., Wu, B., Grimson, J., Wade, V., Richardson, R., and O'Sullivan, D. An Overview of Legacy System Migration. Proceedings of the Fourth Asia-Pacific Software Engineering and international Computer Science Conference, pp. 529-530. Washington DC, USA: IEEE Computer Society, 1997.

[2] Schneider, F. B. Implementing fault-tolerant services using the state machine approach: a tutorial. ACM Computer Surveys 22 (December 1990): 299-319.

[3] Hall, S., Dobson, G. and Sommerville, I. A Container-Based Approach to Fault Tolerance in Service-Oriented Architectures [Online]. 2005. Available from: http://digs.sourceforge.net/papers/2005 icse-paper [2008, OCT 18]

[4] Sneed, H. M. Wrapping Legacy COBOL Programs behind an XML-Interface. Proceedings of the Eighth Working Conference on Reverse Engineering, pp. 189-197. Washington DC, USA: IEEE Computer Society, 2001.

[5] ASNA. Screen Scraping: A Wolf in Sheep's Clothing [Online]. 2004. Available from: www.asna.com [2007, OCT 29]

[6] Salas, J., Pérez-Sorrosal, F., Patiño-Martínez, M. and Jiménez-Peris, R. WS-replication: a framework for highly available web services. Proceedings of the 15th international conference on World Wide Web, pp. 357 – 366. Edinburgh, Scotland: ACM, 2006.

[7] Bass, J. M., Latif-Shabgahi G. R. and Bennett, S. Experimental Comparison of Voting Algorithms in Cases of Disagreement. Proceedings of 23rd Euromicro Conference: New Frontiers of Information Technology, pp. 516-523. Budapest, Hungary, 1997.

[8] Standeven, J., Colley, M.J. and Lyons, D.M. A Hardware Voter for Fault-tolerant Transputer Systems. Microprocessors and Microsystems 13 (November 1989): 588-596.

[9] Parhami, B. Voting networks. Reliability, IEEE Transactions on 40 (August 1991): 380-394.

[10] Lorczak, P. R., Caglayan, A. K. and Eckhardt, D. E. A Theoretical Investigation of Generalised Voters. Digest of papers FTCS'19 : IEEE 19th Ann. Int. Symp. On Fault-Tolerant Computing Systems, pp. 444-451. Chicago, 1989.

[11] Blakeley, J. Universal Data Access with OLE DB. Proceedings of the 42nd IEEE International Computer Conference, pp. 2-7. New York, USA, 1998.

[12] Microsoft. OLE DB for OLAP, Version 1.0 specification [Online]. 1998. Available from: http://www.microsoft.com/data/oledb/01.asp [2008, Feb 6]

[13] Rahgozar, M. and Oroumchian, F. An effective strategy for legacy systems evolution. Journal of Software Maintenance and Evolution: Research and Practice 15(October 2003): 325 – 344.

[14] Moore, M. M. and Moshkina, L. Migrating Legacy User Interfaces to the Internet: Shifting Dialogue Initiative. Proceedings of the Seventh Working Conference on Reverse Engineering (Wcre'00) WCRE, pp. 52-58. Brisbane, Australia: IEEE Computer Society, 2000.

[15] Sneed, H. M. Wrapping Legacy COBOL Programs behind an XML-Interface. Proceedings of the Eighth Working Conference on Reverse Engineering (Wcre'01) WCRE, pp. 189-197. Stuttgart, Germany: IEEE Computer Society, 2000.

[16] Aversano, L., Canfora, G., Cimitile, A. and De Lucia, A. Migrating Legacy Systems to the Web: an Experience Report. Proceedings of the Fifth European Conference on Software Maintenance and Reengineering (CSMR), pp. 148-157. Lisbon, Portugal: IEEE Computer Society, 2001.

[17] Aversano L., Canfora G., and De Lucia A. Migrating Legacy Systems to the Web: a Business Process Reengineering Oriented Approach. in Polo M. (ed.) Advances in Software Maintenance Management: Technologies and Solutions, pp. 151-181. USA: Idea Group Publishing, 2003.

[18] Zou, Y. and Kontogiannis, K. Web-based specification and integration of legacy services. Proceedings of the 2000 Conference of the Centre For Advanced Studies on Collaborative Research. pp. 17. Mississauga, Ontario, Canada: IBM Press, 2000.

[19] Rauchwerger, L. Run-time parallelization: a framework for parallel computation. Ph. D. Thesis. University of Illinois. 1995.

[20] Thiran, P., Hainaut, J., and Houben, G. Database Wrappers Development: Towards Automatic Generation. Proceedings of the Ninth European Conference on Software Maintenance and Reengineering. pp. 207-216. Washington DC, USA: IEEE Computer Society, 2005.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียนวิทยานิพนธ์

นายมนัสวิน เอมะศิริ เกิดเมื่อวันที่ 10 กุมภาพันธ์ พ.ศ. 2525 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาหลักสูตรปริญญาบริหารธุรกิจบัณฑิต (บธ.บ.) สาขาคอมพิวเตอร์ธุรกิจ คณะบริหารธุรกิจ มหาวิทยาลัยกรุงเทพ ในปีการศึกษา 2547 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาศาสตร์คอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2549 และจบการศึกษาในปีการศึกษาในปี 2551

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย