

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

ระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์มด้วยเอ็มคิวซีรีส์

Multi platform Data Interchange System using MQSeries



|                                     |                |
|-------------------------------------|----------------|
| วัน เดือน ปี.....                   | 15 ก.พ. 2550   |
| เลขทะเบียน.....                     | 02271          |
| เลขเรียกหนังสือ.....                | วท.ศ 4645 2547 |
| "ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล." |                |

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน  
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2547

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในห้องสมุดเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

\*H002271\*

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                  |   |
|------------------|---|
| ชื่อหัวข้อ       | ระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์มด้วยเอ็มคิวซีรีส์ |
| นักศึกษา         | นางสาว จิตลดา รอดชัย                                      |
| อาจารย์ที่ปรึกษา | ดร.ชนารัตน์ ชลิตาพงศ์                                     |
| ระดับการศึกษา    | วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ            |
| แขนงวิชา         | วิทยาการสารสนเทศ  |
| ปีการศึกษา       | 2547  |

### บทคัดย่อ

ระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์มด้วยเอ็มคิวซีรีส์ เป็นการพัฒนาระบบงาน เพื่อให้รองรับกับการรับและส่งข้อมูลข้ามแพลตฟอร์ม ซึ่งในปัจจุบันเทคโนโลยีต่างๆ ได้ก้าวหน้าและเปลี่ยนแปลงไปอย่างรวดเร็ว เกิดระบบเทคโนโลยีสารสนเทศใหม่ขึ้นมาเพื่อให้รองรับกับความต้องการที่เปลี่ยนแปลงไป และให้การปฏิบัติงานเกิดประสิทธิภาพมากที่สุด ทำให้เกิดปัญหาในเรื่องของการเชื่อมโยงข้อมูลระหว่างระบบเก่าและระบบใหม่ โครงการฉบับนี้จะพัฒนาระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์ม โดยใช้เอ็มคิวซีรีส์ ซึ่งจะสามารถรองรับการเชื่อมต่อข้อมูลข้ามแพลตฟอร์มของระบบปฏิบัติการ และแพลตฟอร์มของระบบการจัดการฐานข้อมูล โดยรูปแบบของข้อมูลที่ใช้ในการสื่อสารจะอยู่ในรูปของเอกสารอิเล็กทรอนิกส์ ทำให้เกิดความยืดหยุ่นในการทำงานมากขึ้น ระบบงานมีประสิทธิภาพ และทำให้ข้อมูลถูกต้อง เชื่อถือได้

**Title** Multi platform Data Interchange System using MQSeries  
**Student** Miss Chitlada Rodyoy  
**Advisor** Dr. Thanarat Chalidabhongse  
**Level of Study** Master of Science in Information Technology  
**Major** Information Science  
**Academic Year** 2004

## ABSTRACT

Nowaday, the technology is more advanced and evolving dramatically. The needs for new information systems or integrations of existing systems to support new business processes are also increasing. This report describes a development of a middleware for across-platform data exchanging using MQSeries. The system connects different computer systems, and provides communication mechanism between applications on different platforms. The messages passing between the system are in XML format.

## กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้สำเร็จได้ด้วยดี เพราะได้รับความกรุณาจากหลายฝ่าย ขอขอบพระคุณเป็นอย่างยิ่ง ในความกรุณาของ ดร.ธนารัตน์ ชลิตาพงศ์ อาจารย์ที่ปรึกษา ที่ให้การสนับสนุน ช่วยเหลือ ให้ความรู้ คำแนะนำที่ดี ในการปฏิบัติงาน ตลอดจนแนวความคิด ต่างๆ รวมทั้ง ช่วยเหลือในการแก้ไขปัญหาที่เกิดขึ้น และช่วยตรวจแก้ไขรายงานจนเสร็จสมบูรณ์

ขอขอบคุณ รองศาสตราจารย์ ดร.วิเชียร เปรมชัยสวัสดิ์ ที่ได้ให้คำแนะนำในการพัฒนาระบบงาน และให้ความรู้ในด้านต่างๆ

ขอขอบคุณ คณาจารย์ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกท่าน ที่ได้ประสิทธิ์ประสาท วิชาความรู้ให้แก่ข้าพเจ้าตลอดระยะเวลาที่ศึกษาเล่าเรียน

ขอขอบคุณ คุณมงคล รมเย็นที่ได้ให้ข้อมูล ตลอดจนคำแนะนำด้าน เอ็มคิวซีรีส์ และให้การช่วยเหลือ ในการปฏิบัติงาน

ขอขอบคุณ เพื่อนๆ พี่ๆ ทุกท่านที่ได้ให้กำลังใจที่ดี ในการเรียน การทำงาน และมิตรภาพที่ดีแก่กันและกันเสมอมาและตลอดไป

ท้ายที่สุด ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และครอบครัว ที่คอยเป็นกำลังใจในยามที่ข้าพเจ้าเหนื่อยล้า และให้การสนับสนุนส่งเสริมการศึกษา มาโดยตลอด

จิตลดา รอดชัย

23 มิถุนายน 2548

## สารบัญ

|   | หน้า |
|---|------|
| บทคัดย่อภาษาไทย.....  | I    |
| บทคัดย่อภาษาอังกฤษ.....                                     | II   |
| กิตติกรรมประกาศ.....  | III  |
| สารบัญ.....   | IV   |
| สารบัญตาราง.....  | VI   |
| สารบัญรูป.....  | VII  |
| บทที่   |      |
| 1. บทนำ.....  | 1    |
| 1.1 ความเป็นมา.....   | 1    |
| 1.2 วัตถุประสงค์ของการพัฒนาระบบงาน.....                     | 2    |
| 1.3 ขอบเขตของการพัฒนาระบบ.....                              | 2    |
| 1.4 ประโยชน์ที่จะได้รับจากโครงการ.....                      | 3    |
| 2. ทฤษฎีที่เกี่ยวข้อง.....                                  | 4    |
| 2.1 หลักการของระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์ม..... | 4    |
| 2.2 การพัฒนาระบบ.....                                       | 5    |
| 2.3 ทฤษฎีที่นำมาใช้ในการพัฒนาระบบ.....                      | 8    |
| 2.3.1 ภาษาจาวา.....   | 8    |
| 2.3.2 XML.....  | 10   |
| 2.3.3 Trigger และ Stored Procedure.....                     | 11   |
| 2.3.4 SQL/XML.....  | 13   |
| 2.3.5 Java Stored Procedure.....                            | 14   |
| 2.3.6 เอ็มคิวซีรีส์.....                                    | 16   |
| 3. การวิเคราะห์และออกแบบระบบงาน.....                        | 22   |
| 3.1 ความต้องการของระบบ.....                                 | 22   |
| 3.2 การวิเคราะห์และออกแบบระบบ.....                          | 23   |

## สารบัญ (ต่อ)

|  | หน้า |
|--|------|
| 4. การออกแบบฐานข้อมูล .....                              | 42   |
| 4.1 การออกแบบฐานข้อมูล .....                             | 42   |
| 5. การพัฒนาระบบงาน .....                                 | 48   |
| 5.1 การพัฒนาระบบงาน .....                                | 48   |
| 5.2 เครื่องมือและภาษาที่ใช้ในการพัฒนาระบบ .....          | 61   |
| 6. บทสรุป .....  | 62   |
| 6.1 สรุปผลการออกแบบและพัฒนาระบบงาน .....                 | 62   |
| 6.2 ประโยชน์ที่ได้รับจากการออกแบบและพัฒนาระบบ .....      | 62   |
| 6.3 ข้อจำกัดของระบบที่ออกแบบและพัฒนาขึ้น .....           | 63   |
| 6.4 ปัญหาและอุปสรรคระหว่างการออกแบบและพัฒนาระบบงาน ..... | 63   |
| 6.5 ข้อเสนอแนะและแนวทางในการพัฒนาระบบเพิ่มเติม .....     | 63   |
| บรรณานุกรม .....   | 64   |
| ประวัติผู้เขียน .....                                    | 65   |

## สารบัญตาราง

| ตารางที่   | หน้า |
|--|------|
| 4.1 รายละเอียดฐานข้อมูลของตาราง MSG_SCHM_MASTER .....  | 43   |
| 4.2 รายละเอียดฐานข้อมูลของตาราง MSG_SCHM_DETAIL .....  | 44   |
| 4.3 รายละเอียดฐานข้อมูลของตาราง RECONCILE .....        | 45   |
| 4.4 รายละเอียดฐานข้อมูลของตาราง LOOKUP_VIEW .....      | 45   |
| 4.5 รายละเอียดฐานข้อมูลของตาราง REPORT.....            | 45   |
| 4.6 รายละเอียดฐานข้อมูลของตาราง USR.....               | 46   |
| 4.7 รายละเอียดฐานข้อมูลของตาราง USERGROUP .....        | 46   |
| 4.8 รายละเอียดฐานข้อมูลของตาราง GEN_XML_CONDITION..... | 47   |

## สารบัญรูป

| รูปที่ |  | หน้า |
|--------|--|------|
| 2.1    | โครงสร้างของ 2 ระบบที่เชื่อมกันด้วย MQSeries .....               | 4    |
| 2.2    | การทำงานของระบบ .....  | 5    |
| 2.3    | แสดงการขั้นตอนการคอมไพล์โปรแกรม .....                            | 9    |
| 3.1    | Usecasé การ logon เข้าสู่ระบบ .....                              | 23   |
| 3.2    | แสดงUsecase การสื่อสารข้อมูลข้ามระบบ .....                       | 25   |
| 3.3    | แสดง Activity Diagram การบันทึกข้อมูลลง Log file .....           | 25   |
| 3.4    | แสดง Activity Diagram การ Create lookup view w .....             | 26   |
| 3.5    | แสดง Activity Diagram การ Create Report .....                    | 27   |
| 3.6    | แสดง Activity Diagram การ Create XML Schema.....                 | 28   |
| 3.7    | แสดง Activity Diagram การ Delete XML schema .....                | 29   |
| 3.8    | แสดง Activity Diagram ของ Editor .....                           | 30   |
| 3.9    | แสดง Activity Diagram การ Get Message from Queue.....            | 31   |
| 3.10   | แสดง Activity Diagram การ Put Message into Queue.....            | 32   |
| 3.11   | แสดง Activity Diagram การ Monitor Message.....                   | 33   |
| 3.12   | แสดง Activity Diagram การ Reconciliation.....                    | 34   |
| 3.13   | แสดง Activity Diagram การ Validate XML .....                     | 35   |
| 3.14   | แสดง Activity diagram ของ XML extractor .....                    | 36   |
| 3.15   | แสดง Activity diagram ของ XML generator .....                    | 37   |
| 3.16   | แสดง Activity diagram ของ Generate trigger script .....          | 38   |
| 3.17   | แสดง Activity diagram ของ write transaction log file.....        | 39   |
| 3.18   | แสดง Class diagram ของระบบ .....                                 | 41   |
| 4.1    | แสดง ER-Diagram ของระบบ .....                                    | 42   |
| 5.1    | แสดงหน้าจอการตรวจสอบสิทธิ์การเข้าใช้งานระบบ .....                | 48   |
| 5.2    | แสดงหน้าจอการเลือกฟิลด์เพื่อดึงข้อมูลสร้างไฟล์เอ็กซ์เอ็มแอล..... | 50   |

## สารบัญรูป (ต่อ)

| รูปที่ |   | หน้า |
|--------|---|------|
| 5.3    | แสดงหน้าจอ Configuration.....                                       | 51   |
| 5.4    | แสดงหน้าจอการ Monitor คู่มือการทำงานของระบบ.....                    | 52   |
| 5.5    | แสดงหน้าจอการ Reconciliation เพื่อตรวจสอบการทำงานของระบบ .....      | 53   |
| 5.6    | แสดงหน้าจอ Editor สำหรับแสดงและแก้ไขข้อมูลในไฟล์ .....              | 54   |
| 5.7    | แสดงหน้าจอ แจ้งให้ผู้ใช้ทราบหากมีความผิดพลาดเกิดขึ้นจากการรันคำสั่ง | 54   |
| 5.8    | แสดงหน้าจอแสดงผลการรันคำสั่งที่ถูกต้อง .....                        | 55   |
| 5.9    | แสดงหน้าจอ Lookup View เพื่อใช้กำหนด Lookup Table.....              | 56   |
| 5.10   | แสดงหน้าจอการ Validate XML .....                                    | 57   |
| 5.11   | แสดงหน้าจอการค้นหาไฟล์.....   | 57   |
| 5.12   | แสดงหน้าจอการค้นหาmessage schema.....                               | 58   |
| 5.13   | แสดงหน้าจอแสดงรายละเอียด XML Schema .....                           | 58   |
| 5.14   | แสดงหน้าจอแสดงผลข้อผิดพลาดจากการ validate XML .....                 | 59   |
| 5.15   | แสดงหน้าจอแสดงผลแจ้งผลที่ถูกต้องจากการ validate XML.....            | 59   |
| 5.16   | แสดงหน้าจอ Generate Script.....                                     | 60   |

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของโครงการพัฒนาระบบงาน

โครงการพัฒนาระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์มด้วยเอ็มคิวซีรีส์ เป็นการพัฒนาระบบงานเพื่อให้รองรับกับการรับและส่งข้อมูลข้ามแพลตฟอร์ม ซึ่งในปัจจุบันเทคโนโลยีต่างๆ ได้ก้าวหน้าและเปลี่ยนแปลงไปอย่างรวดเร็ว ทำให้เกิดระบบเทคโนโลยีสารสนเทศใหม่ขึ้นมาเพื่อให้รองรับกับความต้องการที่เปลี่ยนแปลงไป ซึ่งองค์กรต่างๆ ได้หันมาให้ความสนใจกับเทคโนโลยีสารสนเทศ และนำระบบการจัดการเทคโนโลยีสารสนเทศเข้ามาใช้ในบริหารงาน เพื่อให้เกิดความสะดวก รวดเร็ว เกิดความถูกต้องของข้อมูล ประกอบกับปัจจุบันจะเห็นได้ว่าระบบธุรกิจมีการแข่งขันกันสูงมาก และให้การปฏิบัติงานเกิดประสิทธิภาพมากที่สุด ซึ่งธนาคารก็เป็นองค์กรหนึ่งที่ได้นำระบบการจัดการเทคโนโลยีสารสนเทศเข้ามาใช้ในการบริหาร และ จัดการภายในองค์กร เนื่องจากเทคโนโลยีได้มีการพัฒนาอย่างรวดเร็ว และต่อเนื่อง จึงเกิดระบบสารสนเทศใหม่ๆ ขึ้นมากมาย และการเปลี่ยนแปลงระบบการจัดการเทคโนโลยีสารสนเทศจะต้องเปลี่ยนแปลงแบบค่อยเป็นค่อยไป ไม่สามารถทำการเปลี่ยนระบบใหม่พร้อมกันได้ทั้งองค์กร และซอฟต์แวร์ที่ใช้ไม่สามารถรองรับได้ทุกระบบงาน จึงทำให้องค์กรต้องใช้งานระบบเก่า(Legacy System) และมีบางระบบงานที่เป็นระบบใหม่ ซึ่งจะต้องมีการใช้ข้อมูลร่วมกันและต้องส่งข้อมูลเชื่อมต่อกันแบบออนไลน์ ระหว่างระบบการจัดการเทคโนโลยีสารสนเทศเดิมที่มีอยู่ และการจัดการเทคโนโลยีสารสนเทศระบบใหม่ ซึ่งความแตกต่างในด้านของเทคโนโลยีที่ใช้ เช่น ระบบงานที่ใช้ทำงานคนละแพลตฟอร์ม หรือมีการใช้ระบบการจัดการฐานข้อมูลที่แตกต่างกัน แต่ต้องการที่จะเชื่อมต่อข้อมูลเข้าด้วยกันแบบเวลาจริง จึงเกิดก่อให้เกิดปัญหาในเรื่องของการสื่อสารข้อมูลระหว่างระบบเก่าและระบบใหม่

ซึ่งในโครงการฉบับนี้จะพัฒนาระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์ม โดยใช้ เอ็มคิวซีรีส์ ซึ่งจะสามารถรองรับการเชื่อมต่อข้อมูลข้ามแพลตฟอร์มของระบบปฏิบัติการ และแพลตฟอร์มของระบบการจัดการฐานข้อมูล ซึ่งรูปแบบของข้อมูล (message) ที่ใช้ในการสื่อสารจะอยู่ในรูปของเอกสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.2 วัตถุประสงค์

1. เพื่อให้สามารถเชื่อมต่อข้อมูลข้ามแพลตฟอร์มได้
2. เพื่อลดการสูญหายของข้อมูล
3. เพื่อให้ตรวจสอบข้อผิดพลาดในการรับและส่งข้อมูลได้ง่าย
4. เพื่อให้เกิดความสะดวก และลดขั้นตอนการทำงาน หากมีการแก้ไขและเปลี่ยนแปลงรูปแบบข้อมูลที่ใช้ในการสื่อสารระหว่างระบบข้ามแพลตฟอร์ม
5. เพื่อเป็นการนำเทคโนโลยีสารสนเทศเข้ามาประยุกต์ใช้ กับระบบงานปัจจุบัน ให้มีประสิทธิภาพมากขึ้น

## 1.3 ขอบเขตของการพัฒนาระบบ

ทำการศึกษาลักษณะการสื่อสารข้อมูลข้ามแพลตฟอร์ม และรูปแบบการทำงานของ เอ็มคิวซีรีส์ เพื่อนำมาวิเคราะห์และออกแบบในการพัฒนาระบบ เพื่อให้ระบบสามารถทำงานได้อย่างถูกต้อง รวดเร็ว และมีประสิทธิภาพ ตลอดจนสามารถตรวจสอบข้อมูลที่ทำการสื่อสารข้ามแพลตฟอร์มย้อนหลังได้ รวมทั้งสามารถตรวจสอบข้อผิดพลาดได้

นอกจากนี้ยังสามารถออกรายงานต่าง ๆ เพื่อตรวจสอบ และนำไปวิเคราะห์ เพื่อประโยชน์ในการวิเคราะห์และวางแผน ซึ่งจะครอบคลุมการทำงาน ดังนี้

1. สามารถทำการกำหนดความต้องการจากระบบได้ว่าต้องการส่งชุดของข้อมูล ไปยังปลายทางที่ชุด ซึ่งแต่ละชุดประกอบด้วยข้อมูลที่สามารถดึงได้จากตารางที่มีอยู่ในระบบ หรือสามารถกำหนดตายตัวได้จากระบบ
2. สามารถทำการกำหนดจากระบบได้ว่าต้องการรับชุดของข้อมูลจากปลายทางที่ ชุด และเมื่อรับเข้ามาแล้วจะให้เก็บ ในตารางอะไรบ้าง และเก็บลงในคอลัมน์ใดของตาราง
3. สามารถตรวจสอบการรับ-ส่งชุดของข้อมูลได้ว่าขณะนี้ระบบทำการส่งข้อมูล และรับข้อมูลอะไรจากระบบปลายทางเข้ามาบ้าง และมีข้อมูลอะไรที่ค้างอยู่บ้าง
4. สามารถทำการเทียบ (reconcile) ชุดของข้อมูลที่ระบบทำการส่งออก และรับเข้าเทียบ กับปลายทางได้
5. สามารถตรวจสอบข้อมูลว่าระบบได้นำเข้าหรือส่งออกได้ถูกต้อง โดยจะมีหน้าจอที่เป็นเอดิเตอร์ ให้สามารถบันทึกออกเป็นรายงานแบบแอสซอร์ค ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. โปรแกรมสามารถทำการเจเนอเรตข้อมูลออกเป็นรูปแบบของเอกสาร เอ็กซ์เอ็มแอล ได้โดยที่สามารถเลือกได้ว่าจะให้ทำ และส่งไปยัง เอ็มคิวซีรีส์ แบบอัตโนมัติ หรือจะทำการเจเนอเรตตามต้องการด้วยผู้ใช้กำหนดเองได้

#### 1.4 ประโยชน์ที่จะได้รับจากโครงการ

ประโยชน์ที่คาดว่าจะได้รับจากการพัฒนาระบบนี้คือ

1. ทำให้ข้อมูลที่ได้มีความถูกต้องและเชื่อถือได้และทำให้ข้อมูลเชื่อมโยงกันข้ามแพลตฟอร์มของระบบปฏิบัติการและแพลตฟอร์มของระบบการจัดการฐานข้อมูล
2. สามารถเชื่อมโยงข้อมูลข้ามแพลตฟอร์มแบบ realtime
3. สามารถดึงข้อมูลที่มีอยู่ในฐานข้อมูลให้ออกมาในรูปแบบของเอกสาร เอ็กซ์เอ็มแอล
4. ลดความผิดพลาด และป้องกันการสูญหายของข้อมูล
5. ลดขั้นตอนในการปฏิบัติงาน และทำให้การทำงานมีความสะดวกมากขึ้น

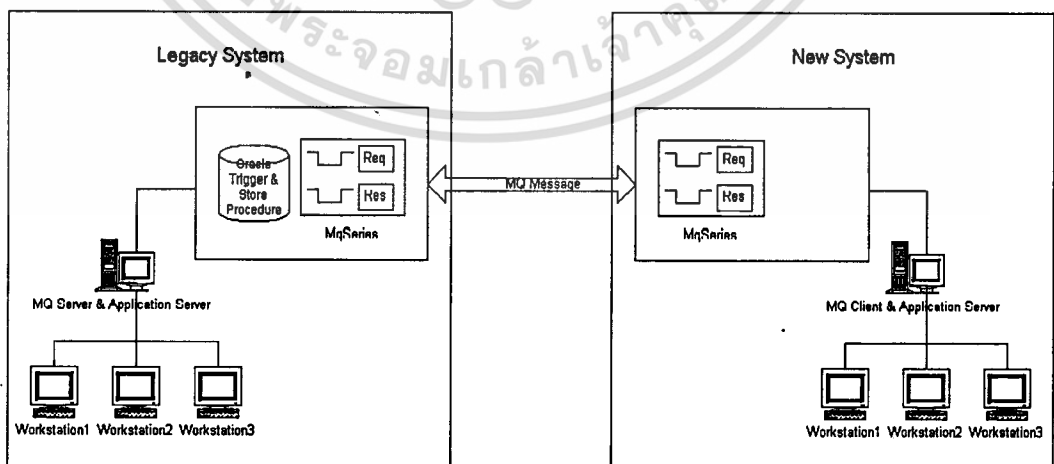
## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

#### 2.1 หลักการของระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์ม

ระบบการจัดการข้อมูลของธนาคารหลายๆระบบถูกพัฒนาขึ้นบน แพลตฟอร์มที่แตกต่างกัน ดังนั้นจึงใช้โปรแกรมมิดเดิลแวร์ที่ชื่อว่า ไอบีเอ็ม เอ็มคิวซีรีส์ เป็นตัวกลางในการจัดการ การสื่อสาร ข้อมูลระหว่าง แพลตฟอร์ม ซึ่งข้อมูลถูกจัดเก็บโดยระบบการจัดการฐานข้อมูลของออราเคิล โดย ที่ลักษณะการรับส่งข้อมูลจะเป็นแบบเวลาจริง ซึ่งจะใช้ความสามารถของออราเคิล ในการใช้ Trigger และ Store procedure รวมทั้ง java store procedure ที่พัฒนาด้วยภาษาจาวาแล้ว ทำการ import เข้า ออราเคิล ซึ่งหลักการทำงานแบ่งได้เป็น 3 ส่วน ดังนี้คือ (ibmMQSeries. n.p.d.)

1. ใช้เป็นตัวส่งข้อมูลที่ต้องการออกไปยัง ระบบอื่นๆ ด้วยการดึงข้อมูลออกจากตารางแล้ว ทำการ Generate ข้อมูลให้อยู่ในรูปของ เอ็กซ์เอ็มแอลไฟล์ จากนั้นจะทำการ put เอ็กซ์เอ็มแอล ไฟล์ ลงไปยัง MQ ต่อจากนั้นจะเป็นหน้าที่ ของ MQ ในการส่งข้อมูล ไปยังระบบปลายทาง และรับ ข้อมูลจาก MQ แล้ว Extract ลงข้อมูลที่เป็น เอ็กซ์เอ็มแอล ไฟล์ลงในตารางฐานข้อมูล
2. ใช้ในการรับข้อมูลออกมาจาก MQ แล้วทำการ จัดเก็บลงตาราง
3. ใช้ในการสร้าง Log file เพื่อเก็บรายการที่เข้าไปกระทำกับ Queue

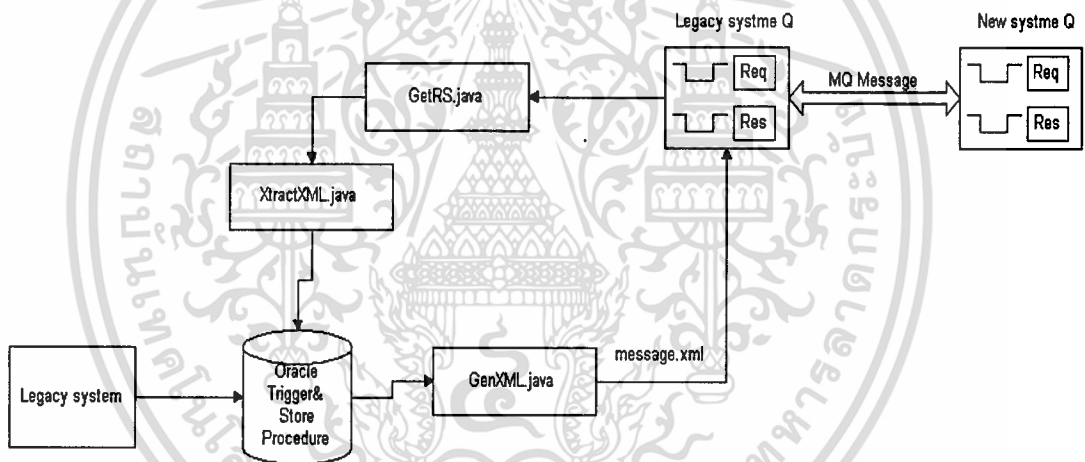


รูปที่ 2.1 แสดงโครงสร้างของ 2 ระบบที่เชื่อมกันด้วย MQ Series

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ลักษณะการทำงานของระบบ

เมื่อ Work station ของ Legacy System ทำการ update ข้อมูลลงบนฐานข้อมูล และถ้าข้อมูลเหล่านั้นต้องการ interface ไปยัง New System ด้วยแล้ว จะต้องทำการ update ข้อมูลเหล่านั้นลง Table กลาง ด้วย และเมื่อใดก็ตามที่ Table กลางถูก insert , update หรือ delete Trigger ซึ่งจะผูกอยู่กับ Table ที่มีการเปลี่ยนแปลงจะ ไปกระตุ้นให้ Java store procedure ซึ่งจะทำหน้าที่ Generate เอ็กซ์เอ็มแอลfile (MQ Series messages) เพื่อจะ put Message นั้น ลงไปยัง Queue ต่อจากนั้นจะเป็นหน้าที่ของ MQ Series ในการรับ-ส่งข้อมูลระหว่าง MQ ด้วยกันเอง เมื่อMQ Series อีกฝั่งได้รับข้อมูลจะทำการ call plug in เพื่อทำการ extract เอ็กซ์เอ็มแอลFile เข้าสู่ฐานข้อมูลต่อไป ดังรูปที่ 2.2 (orafaq. 2004.)



รูปที่ 2.2 แสดงการทำงานของระบบ

### 2.2 การพัฒนาระบบ

โครงการพัฒนาระบบงานได้ ใช้วิธีการพัฒนาระบบที่เรียกว่า SDLC (System Develop Life Cycle) มาใช้ในการพัฒนาระบบ โดยมีลักษณะการทำงานเป็นลำดับจากบนลงล่าง (Top-Down) ซึ่งสามารถแบ่งขั้นตอนการทำงานหลัก ๆ ได้ดังนี้ (melodySDLC. 2003.)

#### 1. Phase -1 Strategic Planning/Business System Planning

ต้องมีการพูดคุยกับผู้บริหารระดับสูงเพื่อทราบนโยบายหลักขององค์กรว่ามีวัตถุประสงค์อย่างไร มีเป้าหมายในการนำ Information System เข้ามาใช้ในงานเพื่ออะไร ต้องการให้องค์กรดีขึ้นอย่างไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. Phase -2 Preliminary Investigation /Feasibility Study

เป็นการสำรวจและศึกษาลักษณะและรูปแบบในการทำงานเบื้องต้น เพื่อดูความเป็นไปได้ในการพัฒนาระบบ และวิเคราะห์ถึงความต้องการที่ระบบควรมี พิจารณาขีดความสามารถของระบบ พยายามกำหนดขอบเขตของระบบงานและเสนอแนวทางในการแก้ปัญหา โดยอยู่ในรูปของทางเลือกที่มีความเป็นไปได้สูงในการแก้ปัญหา

## 3. Phase-3 Detailed Investigation/Fact Finding

เป็นการสำรวจและศึกษาระบบโดยละเอียด ลงไปดูว่าเขาทำงานกันอย่างไร พยายามมองภาพรวมของทั้งองค์กร สูดทำยต้องได้ User Requirement เพื่อนำมากำหนดเป็น System Need จะได้เห็นภาพของระบบที่จะสร้างว่ามีลักษณะอย่างไร

## 4. Phase-4 Design

นำเอา System Need นั้นมาใช้เป็นแนวทางในการออกแบบระบบ โดยตามหลักการออกแบบนั้นระบบที่ได้จะต้อง

Suitability คือ เหมาะสม ตรงกับความต้องการของ User และขีดความสามารถขององค์กร  
Reliability คือ ระบบต้องน่าเชื่อถือ มีความเป็นไปได้ในการทำงานจริง ๆ ซึ่งต้องพิจารณาถึงความเหมาะสมของเทคโนโลยีและงบประมาณ

Easy คือ ระบบต้องง่ายต่อการใช้งานของ User

Simplicity คือ ระบบต้องมีความเรียบง่าย มีขั้นตอนที่ชัดเจน (Step-By-Step)

Economics คือ ระบบต้องคุ้มค่าต่อการลงทุน

ขั้นตอนการออกแบบและวางระบบงาน จะประกอบไปด้วย

1. การออกแบบเอาท์พุท เอาท์พุทสำหรับงานคอมพิวเตอร์ในปัจจุบันมีหลายรูปแบบ ซึ่งอาจแบ่งได้เป็น 2 ลักษณะคือ แบบฮาร์ดคอปปี (Hard Copy) ซึ่งได้แก่รายงานต่างๆ ที่ออกมาทางเครื่องพิมพ์ และ แบบซอฟท์คอปปี (Soft Copy) ซึ่งจะหมายถึงข้อมูลที่แสดงผลออกทางจอภาพชนิดต่าง ๆ และ ไมโครฟอร์ม (Microform) เป็นต้น การออกแบบเอาท์พุทจึงหมายถึง การออกแบบเอาท์พุททางเครื่องพิมพ์ และการออกแบบเอาท์พุททางจอภาพชนิดต่าง ๆ การออกแบบเอาท์พุทถือว่ามีความสำคัญมาก เนื่องจากเอาท์พุทของระบบงานเป็นสิ่งสำคัญในการจะตอบสนองต่อความต้องการของผู้ใช้ระบบ

2. การออกแบบอินพุท เป็นการออกแบบฟอร์มหรือหน้าจอที่จะรับข้อมูลเข้าสู่ระบบ การออกแบบอินพุทจะมีผลกระทบต่อเอาท์พุทที่เราจะได้ออกมาจากระบบ การออกแบบอินพุทจะมีหลักการที่สำคัญแบ่งตามสภาพของอินพุตดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักสำคัญที่ใช้ในการออกแบบแบบฟอร์ม

- แบบฟอร์มควรมีลักษณะที่ง่ายต่อการกรอก
- แบบฟอร์มต้องตรงกับวัตถุประสงค์ที่ต้องการ
- แบบฟอร์มควรมีการออกแบบให้ตรวจสอบความถูกต้องได้
- แบบฟอร์มควรออกแบบให้มีลักษณะที่ดึงดูดต่อผู้ใช้

หลักสำคัญในการออกแบบจอภาพ

- พยายามให้การแสดงข้อมูลบนจอภาพดูเรียบง่ายไม่ซับซ้อน
- พยายามให้การแสดงผลบนจอภาพมีมาตรฐานแบบเดียวกัน
- การใช้สีสำหรับข้อมูลบางอย่างที่ต้องการจะเน้นให้เห็นถึงความแตกต่าง
- การตอบโต้ระหว่างผู้ใช้ระบบกับจอภาพเป็นไปโดยธรรมชาติมากที่สุด

### 3. การออกแบบเพิ่ม และ/หรือฐานข้อมูล

การออกแบบเพิ่มและ/หรือฐานข้อมูลเป็นจุดกึ่งกลางระหว่างอินพุตและเอาต์พุต เนื่องจากการออกแบบระบบที่จะใช้ในการเก็บข้อมูลจากอินพุตเอาไว้สำหรับระบบเพื่อที่ระบบงานจะสามารถนำเอาข้อมูลไปใช้ให้เกิดประโยชน์ได้เมื่อต้องการออกเอาต์พุต เพิ่มและ/หรือฐานข้อมูล มีคุณสมบัติที่จะเอื้อประโยชน์ให้ข้อมูลสามารถถูกเรียกใช้ร่วมกันได้จากระบบงานย่อยต่าง ๆ

ระบบงานคอมพิวเตอร์ทุกระบบในปัจจุบันต้องการกระบวนการที่จะเข้าถึงข้อมูลที่ต้องการได้อย่างสะดวกรวดเร็ว จึงต้องมีการออกแบบระบบฐานข้อมูลให้เกิดความสะดวกและลดความซ้ำซ้อนของข้อมูลเพื่อลดความยุ่งยากในการบำรุงรักษาฐานข้อมูล โครงสร้างระบบฐานข้อมูลในปัจจุบันมี 4 แบบ Relational Database, Hierarchical Database, Network Database และ Object Oriented Database การออกแบบฐานข้อมูลของระบบ เราจะพิจารณาถึงรูปแบบโครงสร้างของระบบ และจากนั้นจึงพิจารณาว่าระบบจะประกอบไปด้วยตารางฐานข้อมูลอะไรบ้าง แต่ละตารางประกอบด้วยฟิลด์ข้อมูลอะไรบ้าง และตารางแต่ละตารางมีความสัมพันธ์กันอย่างไร

### 5. Phase-5 Development

นำ Functional Specification ที่ได้ออกแบบไว้แล้วมาสร้างระบบ โดยออกแบบทั้งในส่วนที่เป็น Physical เช่น อุปกรณ์ เครื่องมือ สภาพแวดล้อม และในส่วนที่เป็น Program ซึ่งเมื่อเสร็จแล้วจะต้องทำการทดสอบระบบจนเป็นที่แน่ใจว่าต้องไม่มีข้อผิดพลาดใด ๆ

การพัฒนาระบบเพื่อนำไปใช้ปฏิบัติงานจริง จะประกอบไปด้วย

1. การพัฒนาโปรแกรมหรือการเขียนโปรแกรมในขั้นตอนนี้เราจะใช้ข้อมูล และ รายละเอียดต่าง ๆ ที่ได้มาจากขั้นตอนการวิเคราะห์ระบบและขั้นตอนการออกแบบระบบเพื่อนำมาใช้ในการเขียนโปรแกรมตามภาษาที่เลือก
2. การทดสอบโปรแกรม เป็นการนำโปรแกรมที่เขียนเสร็จแล้วมาทดสอบ เพื่อวัตถุประสงค์ 2 ประการ คือ เพื่อแก้ไขข้อผิดพลาดที่มีอยู่ในโปรแกรมตามภาษาที่เลือกใช้

#### 6. Phase-6 Implement

เมื่อระบบพัฒนาเสร็จสิ้นแล้ว ต้องมีการนำไปใช้งานจริง ซึ่งต้องพิจารณาทั้งเรื่องของขั้นตอนในการปฏิบัติงานที่เปลี่ยนไป การฝึกอบรม เอกสารคู่มือต่าง ๆ ที่ใช้ภายในระบบ

#### 7. Phase-7 Operating

มีการติดตามและประเมินผลว่า ระบบที่ได้ Implement ไปแล้วนั้นได้ตรงตามเป้าหมายหรือไม่ ตรวจสอบว่ามีปัญหาหรือข้อผิดพลาดใด ๆ หรือไม่

#### 8. Phase -8 Maintenance

หลังจากที่ได้ติดตั้งระบบเรียบร้อยแล้ว นักวิเคราะห์ระบบจะต้องมีการติดตามประเมินผลงาน โดยจะมีการติดตามทางด้านการตรวจสอบทางด้านเทคนิค เพื่อให้มั่นใจว่าระบบยังคงปฏิบัติงาน ได้อย่างถูกต้อง การประเมินความคิดเป็นของผู้ใช้ระบบ โดยให้ผู้ใช้ระบบได้วิจารณ์ระบบอย่างเต็มที่ เพื่อที่จะนำมาแก้ไขปรับปรุงต่อไป การประเมินความคิดเห็นอย่างเป็นทางการ โดยจะเป็นการรวมทีมระหว่างนักวิเคราะห์ระบบ ผู้ใช้ระบบ และตัวแทนจากฝ่ายบริหาร เพื่อที่จะมาร่วมพิจารณาประเมินผลอย่างเป็นทางการสำหรับระบบงาน

## 2.3 ทฤษฎีที่นำมาใช้ในการพัฒนาระบบ

### 2.3.1 ภาษาจาวา

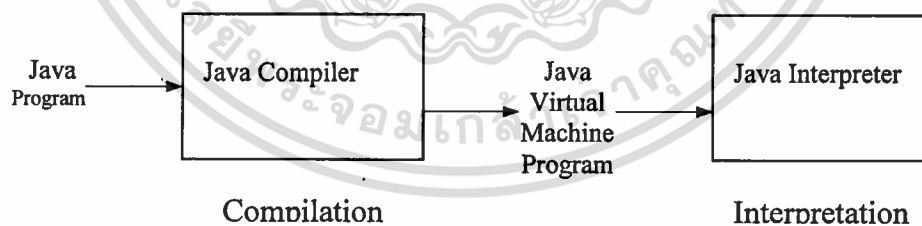
ภาษาจาวาเป็นภาษาเชิงวัตถุ (Object oriented programming language) ที่มีคุณลักษณะของการไม่ขึ้นอยู่กับระบบปฏิบัติการและฮาร์ดแวร์ (Independent platform) โดยโปรแกรมที่ถูกคอมไพล์แล้วสามารถนำไปใช้งานบนเครื่องคอมพิวเตอร์ใดๆ ก็ได้ และยังสามารถนำโปรแกรมนั้นมาใช้งานใหม่( reuse ) โดยอาจจะเปลี่ยนแปลงหรือเพิ่มเติมโปรแกรมบางส่วนได้ เพื่อให้เหมาะสมกับงานที่ต้องการคุณสมบัติที่สำคัญอีกประการหนึ่งของภาษาจาวาคือ โนครีนิที่โปรแกรมถูกส่งผ่านระบบเครือข่ายไปทำงานบนเครื่องคอมพิวเตอร์อื่นๆ จะมีข้อกำหนดดังกล่าว

เอกสารถูกส่งผ่านระบบเครือข่ายไปทำงานบนเครื่องคอมพิวเตอร์อื่นๆ จะมีข้อกำหนดดังกล่าว ยกตัวอย่างเช่น ภาษาจาวาไม่ยอมให้อ้างถึงค่าในหน่วยความจำผ่านทาง Point และจะตรวจสอบว่าค่าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอ้างถึงสมาชิกใน Array อยู่ในขอบเขตหรือไม่ การเปลี่ยนแปลงโปรแกรมหรือค่าในหน่วยความจำซึ่งอาจเป็นอันตรายต่อโปรแกรมอื่นด้วยวิธีนี้จึงไม่สามารถเกิดขึ้นได้ (JavaSun, 2003)

จากคุณสมบัติของการไม่ขึ้นกับแพลตฟอร์มใดๆ นั้น ภาษาจาวาต้องใช้วิธีการแปลภาษาทั้งแบบ Compilation และ Interpretation รวมทั้งการกำหนด Java Virtual Machine ดังรายละเอียดดังนี้

1. Compilation ทำหน้าที่วิเคราะห์ source code ซึ่งกระบวนการนี้จะได้ผลลัพธ์เป็น Executable code เรียกกระบวนการนี้ว่าเป็นการคอมไพล์โปรแกรม ข้อดีของวิธีนี้คือ Executable code ทำงานได้เร็วมากเพราะขั้นตอนในการแปลภาษาถูกแยกออกไปทำก่อนหน้าแล้ว และยังได้ปรับให้โปรแกรมมีขนาดเล็กมีประสิทธิภาพในการทำงาน
2. Interpretation ทำหน้าที่อ่านซอร์สโค้ดที่ละบรรทัดแล้วแปลโปรแกรมบรรทัดนั้นเป็น Executable code และทำงาน จากนั้นอ่านซอร์สโค้ดบรรทัดต่อไป ทำเช่นนี้เรื่อยไปจนจบโปรแกรม วิธีนี้มีทั้งการแปลภาษาและการโปรแกรมเกิดขึ้นสลับกันไป จึงทำให้โปรแกรมทำงานได้ช้ากว่าแบบแรก แต่มีข้อดีคือการสร้างทำได้ง่ายกว่าและโปรแกรมมีขนาดเล็กกว่าด้วย
3. Java Virtual Machine ภาษาจาวาได้นำวิธีการคอมไพล์โปรแกรมทั้ง 2 แบบ มารวมเข้าด้วยกัน เริ่มจากการคอมไพล์โปรแกรมด้วย Java Compiler ให้เป็นโปรแกรมของ Java Virtual Machine แล้วนำเอาโปรแกรมนี้ไปทำงานบนเครื่องคอมพิวเตอร์ใดๆ ก็ได้ที่มี Java Interpreter ดังรูป



รูปที่ 2.3 แสดงการขั้นตอนการคอมไพล์โปรแกรม



</character>

</book>

### ประโยชน์ของ เอ็กซ์เอ็มแอล

1. ใช้สำหรับสร้างข้อมูลที่สามารถอธิบายความหมายของตัวเองได้(Self-describe data)
2. ใช้สำหรับการแลกเปลี่ยนข้อมูล(data exchange) เอ็กซ์เอ็มแอล เป็นไฟล์ธรรมดาที่สามารถใช้ได้กับทุกแพลตฟอร์ม
3. เป็นรูปแบบข้อความในการสื่อสาร(messaging format) ระหว่างแอปพลิเคชันหรือโปรแกรม ซึ่ง เอ็กซ์เอ็มแอล เป็นรูปแบบการสื่อสารระหว่างองค์ประกอบต่างๆตามแนวความคิดของ Web services
4. ประโยชน์ในเชิงเทคโนโลยีอินเทอร์เน็ตและการพัฒนาเว็บ
5. เป็นรากฐานของภาษาใหม่ๆในการพัฒนาเว็บ
6. ใช้ในแวดวงธุรกิจแบบ B2B (Business to Business)
7. ถ้ามีการแก้ไขประเภทและขนาดของข้อมูล ก็ไม่ส่งผลกระทบต่อโปรแกรม

#### 2.3.3 Trigger และ Stored procedure

เนื่องจากระบบงานที่พัฒนาจะต้องสามารถรองรับการสื่อสารข้อมูลข้ามแพลตฟอร์มแบบ realtime จึงมีการใช้ความสามารถของระบบการจัดการฐานข้อมูลในการทำ Trigger และ Store procedure

Trigger เป็นการเขียนโปรแกรมแบบ pl/sql ซึ่งจะถูกระตุ้นให้ทำงานด้วยเหตุการณ์ที่เกิดขึ้นบน table และถูกจัดเป็นคำสั่งแบบ DML(Data Manipulation Language) ประเภทหนึ่ง ใช้เพื่อการค้นหาและปรับปรุงข้อมูล แต่การเขียน Trigger จะไม่สามารถใช้ COMMIT, ROLLBACK และ SAVEPOINT statement ได้ จัดเก็บอยู่ในรูปของ Text ไฟล์ และถูกคอมไพล์ ณ เวลาที่ทำการ execute โดยปกติแล้วจะไม่เขียน Script ไว้บน Trigger มาก แต่จะนิยมเขียนเป็น store procedure หรือ package แล้วเรียกใช้ โดย trigger ซึ่งจะช่วยเพิ่ม performance ให้กับระบบด้วย

การเรียกใช้ trigger สามารถเรียกใช้จาก 3 เหตุการณ์ดังนี้คือ

- INSERT
- UPDATE

เอกสารนี้ DELETED ที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และสามารถระบุให้เกิด Trigger กับก่อนหน้าเหตุการณ์หรือ หลังเหตุการณ์ 3 เหตุการณ์ข้างต้น และสามารถใช้ค่าจาก Table ได้ 2 ลักษณะคือ ค่าเก่า(Old Value) และ ค่าใหม่ (New Value) ดังตัวอย่าง

```
CREATE OR REPLACE TRIGGER  RMDBA.TG_UPD_FRGNCURRPOS
BEFORE INSERT ON RMDBA.FRGNCURRPOS FOR EACH ROW
DECLARE
BEGIN
IF INSERTING THEN
  RMDBA.SP_UPD_FRGNCURRPOS
  (:OLD.AMT1, :OLD.PAIR, :NEW.spot_rate,
  :NEW.fwd_rate, :NEW.split_spot,
  :NEW.split_rate, :NEW.split_pts,
  :NEW.curr_split, :NEW.pair_split);
END IF;
END;
```

Store Procedure เป็น PL/SQL ที่มีประโยชน์ดังนี้

1. เพิ่มความปลอดภัยของข้อมูลและสิทธิ์ในการเข้าถึงข้อมูล
2. เพิ่มประสิทธิภาพให้ระบบด้วยวิธีการดังนี้
  - หลีกเลี่ยงการ reparse
  - หลีกเลี่ยง PL/SQL parsing at run time ด้วยวิธี parsing at compile time
  - ลดการคับคั่งของ Network ด้วย bundling commands
3. ช่วยในการ maintenance ดังนี้
  - สามารถแก้ไขโปรแกรมได้โดยไม่มีผลกระทบในส่วนการทำงานของผู้ใช้
  - สามารถแก้ไขโปรแกรมได้โดยไม่มีผลกระทบต่อ Application อื่นที่มีการเรียกใช้งาน Store procedure
  - ลดปัญหาการทดสอบโปรแกรมที่ซ้ำซ้อนคือ เมื่อมีการแก้ไขการทำงานของ Store procedure แล้วไม่จำเป็นต้องทำการทดสอบ application ทุก application ที่มีส่วนเกี่ยวข้องกับ store procedure

แต่ในการศึกษานี้จะใช้ store procedure ที่เขียนด้วยภาษาจาวา ซึ่งจะใช้ความสามารถของ java และ

เขียนออกมาในรูปของ class และทำการ load เข้า ออราเคิล และสร้างให้ถูกเรียกทำงานด้วย Trigger

คำว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.4 SQL/XML

คือ ฟังก์ชันที่ ออราเคิล สร้างขึ้นเพื่อให้รองรับการทำงานเพื่อการประสานงานกันระหว่างฐานข้อมูลเชิงสัมพันธ์และข้อมูลในรูปแบบของ เอ็กซ์เอ็มแอล ซึ่งข้อมูลในรูปแบบของ เอ็กซ์เอ็มแอล เป็นที่นิยมในการใช้เพื่อการแลกเปลี่ยนข้อมูลระหว่างระบบธุรกิจ(OracleTechnology. n.p.d.)

Oracle9i ได้ถูกพัฒนาขึ้นเพื่อสนับสนุนการ Query ระบบฐานข้อมูลเชิงสัมพันธ์ให้ได้ผลลัพธ์ออกมาในรูปแบบของ เอ็กซ์เอ็มแอล ซึ่งเรียกกันว่า SQL/XML หรือ SQLX ซึ่งปัจจุบันนี้ถูกกำหนดให้อยู่ภายใต้มาตรฐาน ANSI/ISO SQL ซึ่งปัจจุบันระบบการจัดการฐานข้อมูลจากหลายๆ ค่าย เช่น ไอบีเอ็ม, ไมโครซอฟท์, ออราเคิลและซาเยเบส ได้เพิ่มความสามารถเหล่านี้ให้กับผลิตภัณฑ์ของตน

#### มาตรฐาน SQL/XML ประกอบด้วย

- XML: คือชนิดข้อมูลของเอกสาร XML
- XMLAgg: กลุ่มของฟังก์ชัน( aggregate function) เช่น group by
- XMLAttributes: ฟังก์ชันที่ใช้เพื่อระบุแอททริบิวต์ ของ เอ็กซ์เอ็มแอลอีลีเมนต์
- XMLConcat: ฟังก์ชันที่ใช้เพื่อรวมเอกสาร เอ็กซ์เอ็มแอล ตั้งแต่ 2 ชุดขึ้นไปเพื่อให้เป็นเอกสารเอ็กซ์เอ็มแอลชุดเดียว
- XMLElement: ฟังก์ชันที่ใช้เพื่อแปลงข้อมูลจากฐานข้อมูลเชิงสัมพันธ์ให้เป็น เอ็กซ์เอ็มแอล เช่น <elementName>value</elementName>
- XMLForest: ฟังก์ชันที่ใช้เพื่อการ เจเนอเรต ลิสต์ของ เอ็กซ์เอ็มแอล
- XMLNamespaces: ฟังก์ชันที่ใช้เพื่อประกาศ เนมสเปซ ภายใน เอ็กซ์เอ็มแอลอีลีเมนต์
- XMLSerialize: ฟังก์ชันที่ใช้เพื่อแปลง ข้อมูล เอ็กซ์เอ็มแอลให้อยู่ในรูปแบบของสายข้อมูลอนุกรม ซึ่งมีลักษณะคล้ายๆ ข้อมูลที่เป็นชนิดสตริง

การสร้างเอกสาร เอ็กซ์เอ็มแอล

สมมติว่าคุณทำงานเป็นมัคคุเทศน์ ซึ่งมีข้อมูลของนักท่องเที่ยวดังนี้

```
SQL> describe COUNTY
```

| Name        | Type         |
|-------------|--------------|
| COUNTY_NAME | VARCHAR2(10) |
| STATE       | VARCHAR2(2)  |

เอกสารนี้ SQL> describe ATTRACTION ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Name             | Type         |
|------------------|--------------|
| COUNTY_NAME      | VARCHAR2(10) |
| ATTRACTION_NAME  | VARCHAR2(30) |
| ATTRACTION_URL   | VARCHAR2(40) |
| GOVERNMENT_OWNED | CHAR(1)      |
| LOCATION         | VARCHAR2(20) |

ถ้าต้องการสร้างเอกสาร เอ็กซ์เอ็มแอลซึ่งประกอบด้วยรายชื่อผู้เข้ามาติดต่อทั้งหมด สามารถเขียน SQL ได้ดังนี้

```
SELECT XMLElement("Attraction",
    attraction_name)
FROM attraction;
```

จะได้ผลลัพธ์ดังนี้

```
XMLELEMENT("ATTRACTION",ATTRACTION_NAME)
```

```
<Attraction>Pictured Rocks</Attraction>
<Attraction>Da Yoopers ... </Attraction>
<Attraction>Valley Camp ...</Attraction>
```

### การจัดการข้อมูลที่ Element เป็น null

การทำงานบนฐานข้อมูลเชิงสัมพันธ์ นั้น แน่นอนว่าจะต้องมีข้อมูลที่เป็น null เกิดขึ้น ซึ่งเมื่อ generate XML จะทำให้ได้ Element ที่เป็น null ซึ่งมี 2 วิธีที่จะจัดการกับข้อมูลที่เป็น null คือ

1. ใช้ CASE Expression เพื่อดักจับค่า null
2. ใช้ฟังก์ชัน XMLForest เพราะ XMLForest จะ คืนผลลัพธ์ ค่า element ที่ไม่เป็น null

### 2.3.5 Java Stored Procedure

คือ การเขียน store procedure ด้วยภาษา java แล้วทำการโหลดโค้ดเข้าไปจัดเก็บในฐานข้อมูลของ ออราเคิล ซึ่ง java stored procedure จะถูกประมวลผลด้วย JVM (java virtual machine) ที่อยู่ในหน่วยความจำของฐานข้อมูล (memory space) ซึ่ง Java store procedure สามารถเขียนได้ทั้งแบบ JDBC และ SQLJ (Oracle Technology. n.p.d.)

ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### วิธีการเขียน java stored procedure

การเขียน Java Stored Procedures จะต้องปฏิบัติตามกฎเกณฑ์ดังต่อไปนี้

- ไม่จำเป็นต้องสร้าง คอนสตรัคเตอร์ เมธอด
- ตัวแปรและเมธอด จะต้องประกาศเป็นประเภท สเตติก เท่านั้น
- ต้องใช้ฐานข้อมูลเป็นตัวปัจจุบันที่เชื่อมต่ออยู่โดยไม่ต้องระบุ ชื่อผู้ใช้และรหัสผ่านอีก
- ตัวแปรที่เป็นค่าผลลัพธ์จะต้องประกาศเป็นชนิดอะเรย์ ดี ดังตัวอย่าง

```
public static void getEmpInfo(int empno, String[] empName, int[] salary) {
```

### วิธีการโหลด Java Stored Procedure เข้าสู่ฐานข้อมูล

prompt Loading java source into database...

create or replace java source named "Hello" as

```
public class Hello { /* Pure Java Code */
    static public String Msg(String tail) {
        return "Hello " + tail;
    }
}
```

### วิธีการดรอป Java Stored Procedure

ผู้ใช้สามารถดรอป Java Stored Procedure ได้โดยพิมพ์ผ่านบรรทัดคำสั่งเอ็ดิเตอร์ของออรากิเลดดังนี้

```
SQL*Plus: dropjava -u scott/tiger myPackage.myClass
```

### วิธีการเรียกให้ java store procedure ทำงาน

- เรียกใช้งานผ่าน SQL\*Plus: ด้วยคำสั่ง execute หรือ call
- เรียกใช้งานผ่าน JDBC: ด้วยคำสั่ง CallableStatement object

เรียกใช้งานผ่าน SQLJ: ด้วยคำสั่ง #sql { call ...} eg: #sql { call ACME.BEGINRENTAL(:in memberID, :in employeeID, :out, ...)};

### 2.3.6 เอ็มคิวซีรีส์

สำหรับแนวคิดที่จะใช้เอ็มคิวซีรีส์ เป็นตัวจัดการการรับและส่งข้อมูลข้ามแพลตฟอร์ม เพราะเอ็มคิวซีรีส์ เป็นโปรแกรมมิดเดิลแวร์ ที่ถูกพัฒนาขึ้นโดย บริษัทไอบีเอ็ม เพื่อให้รองรับการรับในการส่งข้อมูลข้าม แพลตฟอร์มของทั้งฮาร์ดแวร์ และ ซอฟต์แวร์ เพื่อรองรับทั้งภายในเน็ตเวิร์คเดียวกัน และต่างเน็ตเวิร์ค เพื่อลดปัญหาต่างๆ ดังนี้ (IBMMQSeries. n.p.d.)

1. ความต้องการสื่อสารข้อมูลระดับแอปพลิเคชัน
2. ความต้องการเชื่อมต่อขณะทำงาน (Active connections required )
3. ความแตกต่างกันในด้าน thought put rate ของระบบสองระบบที่ต้องการเชื่อมโยงข้อมูลระหว่างกัน (Different through put rates)
4. ความไม่สอดคล้องกันขอระบบระหว่าง protocol และ vendor (API's not consistent between protocols and vendors)
5. ความยุ่งยากในการคืนคืนข้อมูลหลังจากที่ระบบล้ม(Difficult to recover after failures )

#### 1. ประโยชน์ของ เอ็มคิวซีรีส์

1. ขจัดปัญหาและความยุ่งยากในการจัดการการสื่อสารข้อมูลข้าม เครื่องข่าย, platform, ต่าง protocol
2. มีการทำ loadbalance คืออนุญาตมีมากกว่า 1 ช่องทางในการติดต่อสื่อสาร
3. ไม่ขึ้นกับ protocol ใด ๆ
4. รองรับ แพลตฟอร์มได้มากกว่า middle ware ตัวอื่นๆ ซึ่ง แพลตฟอร์มที่ เอ็มคิวซีรีส์ รองรับได้มีดังนี้ MVS/ESA, AS/400,SINIX, DC/Osx, เอ็มคิวซีรีส์ SunOS:, HP-UX and เอ็มคิวซีรีส์ Sun Solaris, AIX, OS/2 Warp, WindowsNT, Tandem NonStop Kernel, Windows

#### 2. Perspective ของ เอ็มคิวซีรีส์

1. Time independent(asynchronous) processing
2. Connectionless communications
3. Assured message delivery รับประกันว่าชุดข้อมูลจะถูกส่งไปยังระบบปลายทางอย่างแน่นอน และถูกส่งแบบเวลาจริง (realtime)
4. Syncpoint control ซึ่งจะช่วยในการ สำรองข้อมูลและการกู้คืนข้อมูลเมื่อเกิดความผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 5. รองรับ แพลตฟอร์มได้ทั้ง Single และ มัลติ-แพลตฟอร์มAPI  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เขียนโปรแกรมได้อย่างรวดเร็วขึ้นเพราะ เอ็มคิวซีรีส์ มีเมธอดเตรียมพร้อมไว้ให้เรียกใช้ และมีการทำ Portable code เพื่อทำการย้ายโค้ดได้โดยไม่ต้อง แก้ไข เช่น จาวา

### 3. ระบบความปลอดภัยของ เอ็มคิวซีรีส์

1. มีการทำ Single Sign-on กรณีที่มีการ logon เข้าสู่ระบบมากกว่า 1 ครั้งในเวลาเดียว
2. มีการทำ Secure sign-on โดยการทำให้ password encryption
3. มีการทำ Authentication โดยการใช้ passwords , PIN number
4. มีการทำ Audit โดยเก็บ log ว่ามีใครเข้าใช้ระบบ เวลาเท่าไร
5. Data Integrity โดยการตรวจสอบสิทธิ์ เพื่อตรวจจับไม่ให้คนที่ไม่มีสิทธิ์เข้าใช้ระบบ มาทำการแก้ไขเปลี่ยนแปลงข้อมูล
6. Confidentiality ปกปิดข้อมูล ไม่ให้ระบบภายนอกเข้ามาเห็นหรือทำการแก้ไขเปลี่ยนแปลงข้อมูล
7. มีการทำ Authorization โดยกำหนดระดับของสิทธิ์ในการเข้าใช้ระบบ
8. การควบคุมการเข้าถึงข้อมูลภายในระบบ โดยใช้วิธีการ authorization (Access Control)

### 4. ระบบการจัดการ ชุดของข้อมูลของ เอ็มคิวซีรีส์

ในการรับส่งข้อมูลด้วยเอ็มคิวซีรีส์สามารถกำหนดชุดของข้อมูลที่ต้องการสื่อสารออกได้เป็น

1. one to one message flow คือกำหนดให้แอปพลิเคชันมีการรับและส่งชุดของข้อมูลกันแบบ 1:1 และสามารถกำหนดให้ชุดของข้อมูลสามารถค้างอยู่ที่คิวจนกว่าจะถูกดึงไปใช้งาน หรือจนกว่าจะหมดอายุ (expire)
2. one to many message flow คือกำหนดให้มีแอปพลิเคชันสำหรับส่ง ชุดของข้อมูล โดย 1 แอปพลิเคชันแต่มีหลายแอปพลิเคชัน ที่จะเข้ามาอ่านชุดของข้อมูลไปใช้งานได้ และกำหนดชุดของข้อมูลจะค้างอยู่ที่คิวจนกว่าจะมีตัวสุดท้ายเข้ามาอ่าน หรือจนกว่าจะหมดอายุ (expire) และเราสามารถกำหนดให้ระบบต้นทางส่ง acknowledge ออกไปบอก ระบบปลายทาง เพื่อให้ทราบ ว่า ชุดของข้อมูลได้ส่งถึงปลายทางแล้ว
3. many to one message flow คือกำหนดให้มีแอปพลิเคชัน ทำการส่งข้อมูลเข้าไปยังคิวได้มากกว่า 1 แอปพลิเคชัน แต่มีระบบต้นทาง ได้เพียงแค่ 1 ระบบ เท่านั้น
4. Publish/Subscribe การรับและส่งข้อมูลใน เอ็มคิวซีรีส์ ด้วยวิธีนี้ ตัว ระบบปลายทาง จะทำการ broadcast ชุดของข้อมูลออกไปโดยไม่รู้ว่า ระบบต้นทางที่จะรับที่จะรับเป็นใคร แต่ ผู้รับจะต้องรู้ว่าตัวเองจะรับชุดข้อมูลชุดใดเข้ามาดูหน้าไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Request/Reply การรับและส่งข้อมูลวิธีนี้ เกิดจากมี แอปพลิเคชัน ร้องขอ ชุดข้อมูล ไปยังตัวให้บริการ เมื่อตัวให้บริการได้รับ request message จะทำการเจเนอเรต reply message กลับไป ซึ่งกระบวนการนี้จะถือว่าสิ้นสุดเมื่อระบบผู้ร้องขอได้รับ reply message กลับมา
6. One-way datagram or send and forget pattern เป็นลักษณะของการส่ง ชุดข้อมูล ทางเดียวโดยไม่คาดหวังการ reply message กลับมา นั่นคือ เป็นการสื่อสารแบบ Asynchronous แต่เนื่องจาก เอ็มคิวซีรีส์ รับประกันการส่งว่าชุดข้อมูลจะถึงผู้รับปลายทาง ดังนั้นการส่งแบบ one-way datagram จึงใช้ได้ทั้งการสื่อสารแบบ one to one message flow และ publish/subscribe ซึ่งในการส่งชุดข้อมูลหากมีข้อมูลชุดใด ๆ ที่ละเมิดกฎของคิวจะทำให้ชุดข้อมูลชุดนั้นถูกปฏิเสธออกไปเก็บไว้ที่ Dead letter Queue

#### 5. การจัดการ ทรานแซคชัน ของ เอ็มคิวซีรีส์

ทรานแซคชัน คือหน่วยของการทำงาน ซึ่งจะต้องประกอบด้วยคุณสมบัติหลัก 4 ประการ ที่เรียกว่า ACID คือ

1. Atomicity เมื่อมีงานใดงานหนึ่งใน ทรานแซคชัน เกิดล้มเหลว งานอื่นๆ ใน ทรานแซคชัน เดียวกันจะต้องถูก rollback แต่ถ้าสำเร็จก็จะต้องสำเร็จทุกๆ งานที่อยู่ใน ทรานแซคชัน เดียวกัน เช่นการ โอนเงินเข้าบัญชี จะมีงาน 2 งานเกิดขึ้น คือถอนเงินจากบัญชีนายแดง และ โอนเงินเข้าบัญชีนายดำ ถ้าหากงาน โอนเงินเข้าบัญชีนายดำ ไม่สำเร็จ งานในส่วนของการถอนเงินจากบัญชีนายแดงจะต้องถูกยกเลิกด้วย
2. Consistency คือความคงสภาพของข้อมูล โดยที่ข้อมูลจะต้องมีความถูกต้องตรงกัน
3. Isolation คือ ทรานแซคชัน ต้องมีความเป็นอิสระของตัวเองไม่ขึ้นกับกระบวนการทำงานอื่น นั่นคือถ้ามีการทำงานพร้อมกันมากกว่า 1 ทรานแซคชัน ก็ให้มองเสมือนทำงานแบบอนุกรม
4. Durability ไม่ว่าจะเกิดการเปลี่ยนแปลงใดๆ เช่น ฮาร์ดแวร์หรือ ซอฟต์แวร์ ล้มเหลว ขณะที่ทรานแซคชันคอมมิต ข้อมูลจะต้องเก็บในพื้นที่ที่มีความปลอดภัย (persistence) โดยงานที่เกิดก่อนจุด sync point จะไม่สูญหายแต่งงานใดที่เกิดหลัง Sync point เมื่อเกิดความผิดพลาดจะทำให้ข้อมูลที่อยู่หลัง sync point อาจสูญหายได้

ในการสื่อสารข้อมูลด้วย เอ็มคิวซีรีส์ ระบบจะสามารถส่งถึงปลายทางได้ แต่ถ้ามีปัญหาในการส่ง ชุดข้อมูล ที่ถูกส่งออกมาจาก ทรานแซคชัน เดียวกันจะถูก rollback ทั้งหมด และถ้าหากมีการ rollback ทรานแซคชัน แล้ว ชุดข้อมูลใดๆ ที่ถูกส่งออกมาจาก ทรานแซคชัน เดียวกันจะถูกลบออกจากคิว โดย เอ็มคิวซีรีส์ เมื่อมี ทรานแซคชัน คอมมิต ชุดข้อมูลจะถูก ส่งและ ลงสู่

คิวและแอปพลิเคชัน อื่นๆจะมองเห็นได้ สามารถกำหนดให้ใช้ single phase commit หรือ two phase commit ในการทำ concurrency control

- Single-phase and two-phase commits เพื่อให้การทำทรานแซกชันบนระบบฐานข้อมูล

แบบกระจายมีคุณสมบัติ Atomicity คือทรานแซกชัน T ต้อง commit ในทุก ๆ ไซต์ หรือยกเลิกการทำทรานแซกชันในทุก ๆ ไซต์ ตัวประสานงานทรานแซกชันของทรานแซกชัน T ต้องมีการทำ commit protocol ซึ่งวิธีการของ commit protocol มีการใช้งานกันอย่างแพร่หลายคือ two-phase commit protocol (2PC) และ three-phase commit protocol (3PC) ซึ่งจะช่วยให้หลีกเลี่ยงข้อเสียของ 2PC แต่การดำเนินการจะมีความซับซ้อนมากกว่า ตัวอย่าง ให้ T เป็นทรานแซกชันเริ่มต้นที่ไซต์  $S_i$  และให้ ตัวประสานงานทรานแซกชันที่ไซต์  $S_i$  เป็น  $C_i$  เมื่อทรานแซกชัน T ได้ทำการประมวลผลเสร็จสิ้น นั่นคือ ทุก ๆ ไซต์ที่ทรานแซกชัน T ได้ประมวลผลได้แจ้งกลับไปที่  $C_i$  ว่า ได้ทำทรานแซกชัน T เสร็จเรียบร้อยแล้ว ตัวประสานงานทรานแซกชัน  $C_i$  จะเริ่มทำ โพรโตคอล 2PC

- Phase1  $C_i$  เพิ่มเรคอร์ด <prepare T> ลงไปใน log ไฟล์บน stable storage จากนั้นก็จะส่งสัญญาณ prepare T ไปให้กับทุก ๆ ไซต์ที่ประมวลผลทรานแซกชัน T เมื่อไซต์อื่น ๆ ได้รับสัญญาณ ตัวจัดการทรานแซกชันจะต้องตอบกลับไปหาตัวประสานงานทรานแซกชันว่าจะทำการ commit ทรานแซกชันได้หรือไม่ ถ้าไม่ได้จะเพิ่มเรคอร์ด <no T> ลงไปใน log ไฟล์และส่งสัญญาณ abort T กลับไปที่  $C_i$  ถ้าได้จะเพิ่มเรคอร์ด <ready T> ลงไปใน log ไฟล์ และตัวจัดการทรานแซกชันจะส่งสัญญาณ ready T กลับไปที่  $C_i$

- Phase2 ทรานแซกชัน T จะ commit ได้ก็ต่อเมื่อได้รับสัญญาณ ready T จากทุก ๆ ไซต์ที่มีส่วนร่วมในการทำทรานแซกชัน นอกเหนือจากนั้นทรานแซกชัน T จะต้องถูกยกเลิก ซึ่งก็จะต้องทำการเพิ่มเรคอร์ด <commit T> หรือ <abort T> ลงไปใน log ไฟล์ ถ้าทรานแซกชัน T สามารถ commit ได้ หรือ ยกเลิกการทำทรานแซกชัน จากนั้น ตัวประสานงานทรานแซกชันก็จะส่งสัญญาณ commit T หรือ abort T ไปให้กับทุก ๆ ไซต์ที่ร่วมกันทำทรานแซกชัน เมื่อแต่ละ ไซต์ได้รับสัญญาณก็จะทำการบันทึกเรคอร์ดนี้ลงไปใน log ไฟล์ จะเห็นว่าสัญญาณ ready T มีความสำคัญมาก เนื่องจากไซต์ต่าง ๆ ที่ทำทรานแซกชัน T จะต้องส่งสัญญาณ ready T กลับไปให้ตัวประสานงานทรานแซกชัน เพื่อบอกกับตัวประสานงานทรานแซกชันว่าพร้อมที่จะ commit แล้วซึ่งสัญญาณนี้จะมีผลต่อการทำ commit หรือ abort ของทรานแซกชัน จากลักษณะดังกล่าวอาจเป็นไปได้ว่าเมื่อไซต์หนึ่งได้ทำการส่งสัญญาณ ready T ไปให้ตัวประสานงานทรานแซกชันแล้ว อาจเกิดความล้มเหลวของไซต์ได้ ซึ่งจากโปรโตคอล 2PC เมื่อตัวประสานงานทรานแซกชันได้รับสัญญาณ ready T หรือ abort T ครบ ก็จะส่งสัญญาณ commit T หรือ abort T ไปให้กับไซต์ที่ร่วมทำทรานแซกชันเท่านั้น โดยไม่ได้สนใจว่าไซต์ต่าง ๆ

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหล่านั้นสามารถ commit หรือ abort ตามสัญญาที่ส่งไป ได้หรือไม่ ดังนั้นในการดำเนินการโปรโตคอล 2PC จะมีการส่งสัญญา acknowledge T กลับมาที่ตัวประสานงานทรานแซกชัน เพื่อเป็นการบอกว่าได้ดำเนินการ ในระยะที่สองเสร็จเรียบร้อยแล้ว เมื่อตัวประสานงานทรานแซกชันได้รับ acknowledge T ครบจากทุก ๆ ไซต์ ก็จะเพิ่มเรคอร์ด <complete T> เข้าไปใน log ไฟล์ ความสามารถในการรองรับภาษาในการเขียนโปรแกรมของ เอ็มคิวซีรีส์

เอ็มคิวซีรีส์ รองรับภาษาในการเขียน โปรแกรมได้มากมายดังนี้ MQI(Message Queue Interface) : เป็นภาษาที่ เอ็มคิวซีรีส์ มีให้, AMI(Application Message Interface), C++(รองรับการทำงานแบบ OO), ActiveX (รองรับการทำงานบน window platformมีการใช้ DCOM, COM+), จาวา (สามารถรองรับได้หลากหลาย platform), เจเอ็มเอส (java messaging service)

## 6. องค์ประกอบหลักในการสื่อสารข้อมูลแบบ เอ็มคิวซีรีส์

1. ชุดของข้อมูล คือ ข้อมูลที่อยู่ในรูปของสตริง ที่ต้องการสื่อสารซึ่งอาจจะเป็นการสื่อสารภายใน Application เดียวกัน ต่าง application หรือต่างแพลตฟอร์มก็ได้ ซึ่ง ชุดข้อมูล จะประกอบด้วย 2 ส่วนคือ

- message descriptor (Header).จะเป็นส่วนของ control information เช่น ประเภทของ ชุดข้อมูล หรือ priority ของ ชุดข้อมูล เป็นต้น
- application data (Body) คือส่วนของข้อมูลจริงๆที่ต้องการสื่อสารกัน ซึ่งโครงสร้างของข้อมูลจะถูกกำหนดโดย Application โดยที่ตัว MQSerie จะไม่เข้าไปมองหรือยุ่งกับข้อมูลส่วนนี้

2. คิว (Queues) คือที่สำหรับพักหรือเก็บชุดข้อมูลเพื่อรอให้คิวเมเนเจอร์จัดการส่งต่อไปยังปลายทาง ซึ่งอาจถูกเก็บในหน่วยความจำหลัก (เก็บชั่วคราวเพื่อรอส่งออก) หรือ ฮาร์ดดิสก์ (กรณีที่ต้องมีการคั่นคืนข้อมูลหลังจากที่ระบบล่ม)

คิว แบ่งได้เป็นสองประเภท คือ

- Local Queue จะถูกจัดการด้วยคิวเมเนเจอร์เพียงตัวเดียวและถูกเก็บใน Node เดียว (local system)
- Remote Queue จะถูกจัดการด้วยคิวเมเนเจอร์มากกว่า 1 ตัวและอยู่ต่างเครื่องกัน (remote system)

การเข้าถึงชุดข้อมูลในคิวสามารถเข้าถึงได้ 3 แบบ คือ FIFO (เข้าก่อนออกก่อน), กำหนด priority ของชุดของข้อมูล และกำหนดให้มีการเข้าถึงแบบโดยตรงได้

3. คิวเมเนเจอร์เป็นหัวใจหลักในการทำงานของ เอ็มคิวซีรี่ส์ เช่น สั่งจัดการ put ชุคข้อมูลลงคิวด้วยคำสั่ง MQPUT และรับชุดข้อมูลจากคิวด้วยคำสั่ง MQGET เป็นต้น และยังทำหน้าที่ให้บริการดูแลจัดการคิวซึ่งหน้าที่หลักๆ ของ คิวเมเนเจอร์ แบ่งออกได้ดังนี้
    1. ให้บริการการจัดการคิว ของชุดข้อมูล
    2. ติดต่อรับส่ง ชุดข้อมูล กับ แอปพลิเคชัน ด้วย MQI(Message Queue Interface)
    3. ติดต่อส่งผ่าน ชุดข้อมูล กับ คิวเมเนเจอร์ ตัวอื่นๆ โดยอาศัยเครือข่ายเป็นตัวกลาง
    4. มีฟังก์ชันเพิ่มเติมเพื่อให้ผู้ดูแลระบบทำการจัดการชุดข้อมูลในคิว ได้ เช่น putQueue() หรือ getQueue() โดยใช้เครื่องมือ RUNMQSC เพื่อรันคำสั่ง
- เอ็มคิวซีรี่ส์ เป็นมิดเคิลแวร์ที่น่าสนใจตัวหนึ่ง ที่มีลักษณะการทำงานที่ยืดหยุ่น สามารถรองรับการทำงานได้หลากหลายแพลตฟอร์มและหลากหลายภาษา อีกทั้งยังสามารถรองรับชุดข้อมูลขนาดใหญ่ได้ถึง 100 เมกะไบต์ สามารถกำหนดการเข้าถึงชุดข้อมูลได้ และ เอ็มคิวซีรี่ส์ยังมีระบบความปลอดภัย ทำให้มีระบบความปลอดภัย และเนื่องจาก MQI ได้มีออบเจ็ค เตรียมพร้อมให้เรียกใช้งานจึงทำให้การเขียน โปรแกรมสื่อสารข้อมูลด้วย เอ็มคิวซีรี่ส์ ง่ายและรวดเร็ว และจากการเปรียบเทียบ performance พบว่าถ้าส่งข้อมูลด้วยวิธี เอ็มคิวซีรี่ส์ ที่ระบบปฏิบัติการ วิน โดว์2000 จะทำให้ performance ต่ำกว่า MSMQ

## บทที่ 3

### การวิเคราะห์และออกแบบระบบงาน

#### 3.1 ความต้องการของระบบ

ความต้องการของระบบการสื่อสารข้อมูลด้วย MQ Series

1. สามารถกำหนดตาราง และฟิลด์ในการส่งข้อมูลได้
2. ข้อมูลที่ใช้ในการรับและส่งจะต้องอยู่ในรูปของเอ็กซ์เอ็มแอลไฟล์ ซึ่งสามารถกำหนดแอททริบิวต์, อีลีเมนต์, และ ระดับของแท็กได้
3. มีฟังก์ชันในการ validate xml หลังจากทีระบบ generate ข้อมูล และแจ้งผลเมื่อสิ้นสุดการตรวจสอบความถูกต้อง
4. สามารถกำหนดเหตุการณ์ในการส่งข้อมูล ในแต่ละ set ได้ เช่น ส่งข้อมูลเพื่อมีการเปลี่ยนแปลงค่าบางค่าของข้อมูลในตาราง หรือเมื่อมีการ เพิ่มข้อมูล หรือ ลบข้อมูลในตาราง หรือ สามารถตั้งเวลาในการส่งได้ เช่น ให้มีการส่งข้อมูลทุกๆ 2 นาที รวมทั้งส่งข้อมูลแบบเวลาจริง ได้ด้วยและสามารถเจเนอเรตข้อมูลไปเก็บยัง โพลเดอร์ที่ระบุโดยไม่ต้องมีการ อินเตอร์เฟซกับระบบใดๆ เพราะปัจจุบันมีข้อมูลที่ต้องรายงาน ธนาคารแห่งประเทศไทย ในรูปของ เอ็กซ์เอ็มแอล โดยวิธีการส่งต้อง เข้าไป แนบ ไฟล์ ที่ เว็บไซต์ ของธนาคารแห่งประเทศไทย

1. สามารถกำหนดได้ว่า ชุดข้อมูล จะถูกนำไปใช้งานที่ระบบ ซึ่งหลังจากทำการ get data ขึ้นมาจาก Queue แล้ว ชุดข้อมูล จะถูกลบออกจาก Queue ทันที หรือจะยังคง ชุดข้อมูล ไว้จนกว่าระบบสุดท้ายที่ต้องใช้ข้อมูลเข้ามา
2. สามารถแก้ไขและเปลี่ยนแปลงข้อมูลในการส่งได้ โดยไม่มีผลกระทบต่อโปรแกรม เช่น ส่งรายชื่อจากเดิม 50 อักขระ เปลี่ยนเป็น 45 โดยไม่มีผลกระทบต่อโปรแกรม หรือถ้าต้องการเพิ่มและลด element ในการส่งสามารถเข้าไปแก้ไข parameter ของระบบได้
3. ทุกครั้งที่มีการ interface data เกิดขึ้นจะต้องเก็บ log file เพื่อใช้ตรวจสอบ หากมีปัญหาใดๆเกิดขึ้น ซึ่ง log fileจะต้องบอกวันและเวลาที่ทำการ interface หน่วยงานที่ต้นทางที่ส่งข้อมูล และระบุหน่วยงานปลายทางที่รับข้อมูล เช่น

\*\*\*\*\* 18/09/03 17:19:50 \*\*\*\*\*

----> [DrsGetRs] Call Plugin : UpdateDealID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATE=20030804|TIME=17263900|REQ\_NAME=DRS|PLATFORM=UNX|DEST\_APP\_NAME=UAT|PR\_REP\_CODE=|SC\_REP\_CODE=|REC\_TYPE=|TRANS\_CODE=SU250000300001|PROD\_FLAG=TR|FILLER=|RqUID=|AgentInfo=|Status=0|ErrMsg=|KdealID1=39|KdealID2=39|ExtDealId=To-1|InstType=BUY-FCD|Event=NEW|Blk\_ID=|

\*\*\*\*\* 18/09/03 17:19:53 \*\*\*\*\*

----> [DrsGetRs] Get Request Msg. :

DATE=20030804|TIME=17263800|REQ\_NAME=DRS|PLATFORM=UNX|DEST\_APP\_NAME=UAT|PR\_REP\_CODE=|SC\_REP\_CODE=|REC\_TYPE=|TRANS\_CODE=SU250000300001|PROD\_FLAG=TR|FILLER=|RqUID=|AgentInfo=|Status=0|ErrMsg=|KdealID1=|KdealID2=|ExtDealId=SP-21|InstType=IMP-TT-OUT|Event=NEW|Blk\_ID=|

\*\*\*\*\* 18/09/03 17:19:55 \*\*\*\*\*

----> [DrsGetRs] Get Request Msg. :

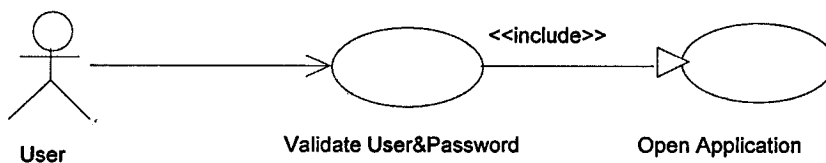
DATE=20030804|TIME=17263900|REQ\_NAME=DRS|PLATFORM=UNX|DEST\_APP\_NAME=UAT|PR\_REP\_CODE=|SC\_REP\_CODE=|REC\_TYPE=|TRANS\_CODE=SU250000300001|PROD\_FLAG=TR|FILLER=|RqUID=|AgentInfo=|Status=1|ErrMsg=error.CmMsgCallback.ServerDataMessageCallback : Import Server KIS Error message (level=1028 , Key=ForwardDeals , ErrorMessage\_get=The reference is not found. , DataMessage=ERROR Unknown reference Broker:

4. สามารถดูรายงานการ และ เทียบผลการทำงาน เพื่อตรวจสอบว่าข้อมูลทางฝั่งรับและฝั่งส่ง ถูกต้องตรงกัน

### 3.2 การวิเคราะห์และออกแบบระบบ

#### 3.2.1 Usecase Diagram

- Logon Application เป็น usecase เพื่อการตรวจสอบสิทธิ์ผู้เข้าใช้งานระบบ



รูปที่ 3.1 แสดงUsecase การ logon เข้าสู่ระบบ

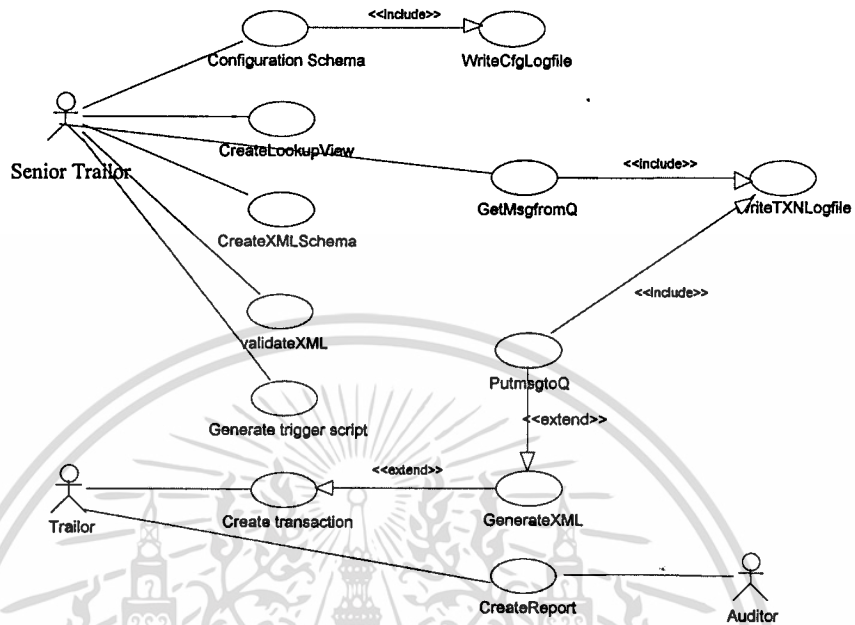
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญูญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Data Interface** เป็น Usecase เพื่อใช้ในการสื่อสารข้อมูล ซึ่งจะประกอบด้วย usecase หลักๆ ดังนี้

- Configuration Schema เป็น usecase สำหรับกำหนดการสร้าง และแก้ไข Message Schema
- Write Log file เป็น usecase สำหรับการเขียนบันทึกการเปลี่ยนแปลงจากการแก้ไข Message Schema เพื่อสามารถให้ดูการเปลี่ยนแปลงย้อนหลัง ได้
- Create Lookup View เป็น usecase สำหรับการสร้าง Lookup table เพื่อใช้ในกรณีที่ระบบมีตารางมาก แล้วไม่ได้มีการใช้งานทุกๆ ตาราง จึงไม่จำเป็นที่จะต้องแสดงตารางทั้งหมดที่มีอยู่ และเพื่อให้สะดวกในการใช้งานกับผู้ใช้ด้วย
- Create XML Schema เป็น usecase สำหรับการสร้าง schema เพื่อเป็นตัวกำหนดว่าระบบจะประกอบด้วยชุดข้อมูลอะไรบ้าง ประกอบด้วย Tag อะไรบ้าง และข้อมูลแต่ละ Tag นั้นถูกดึงมาจากตารางไหน และ ฟิลด์ใด ซึ่งผู้ใช้สามารถกำหนดระดับของ Tag ได้ด้วย
- Validate XML เป็น usecase สำหรับการตรวจสอบความถูกต้องของ เอกซ์เอ็มแอล ไฟล์ กับ สคิม่าที่มีอยู่
- Generate Trigger Script เป็น usecase สำหรับเจนเนอเรต ทริกเกอร์ เพื่อผูกไว้กับตารางที่ถูกกำหนดให้มีการดึงข้อมูลออกมาแปลงในรูปของ เอกซ์เอ็มแอล ไฟล์
- Create Transaction เป็น usecase ที่เป็นตัวเริ่มต้น ให้ ยูสเคส GenerateXML ทำงาน
- GenerateXML เป็นยูสเคสสำหรับให้ระบบ เจนเนอเรตเอกซ์เอ็มแอลออกมาตามสคิม่าที่กำหนด ซึ่งการเจเนอเรตเอกซ์เอ็มแอลไฟล์อาจไม่จำเป็นจะต้องทำทุกครั้งหลังจากที่มีการทำทรานแซกชันก็ได้ ถ้าหากว่าเงื่อนไขของการทำทรานแซกชัน ไม่ตรงกับเงื่อนไขที่ระบุไว้ในสคิม่า
- PutMsgtoQ เป็นยูสเคสสำหรับนำเอาเอกซ์เอ็มแอลที่ได้จากการเจเนอเรตวางลงบนคิวของเอ็มคิวซีรี่ส์ซึ่งการนำเอาเอกซ์เอ็มแอลมาวางลงบนคิวอาจจะไม่ต้องทำทุกครั้งหลังจากที่ทำยูสเคส GenerateXML เพราะบางครั้งถ้าหากผู้ใช้กดปุ่มเพื่อเจเนอเรตเอกซ์เอ็มแอลเองระบบจะนำไฟล์เอกซ์เอ็มแอลที่ได้เก็บไว้ใน โพลเดอร์ที่ทำการคอนฟิกไว้แต่แรก
- WriteTXNLogfile ทุกครั้งที่มีการติดต่อกับคิวของเอ็มคิวซีรี่ส์ระบบจะต้องเก็บล็อกไว้ทุกครั้งนั่นคือการทำงาน ของ WriteTXNLogfile
- GetMsgfromQ เป็นยูสเคสสำหรับมอนิเตอร์ดูข้อมูลบนคิวฝั่งขาเข้าเพื่อคอยตรวจสอบจับคู่ว่าถ้าเมื่อใดที่พบชุดข้อมูลบนคิวให้ทำการเอ็กซ์แทรคข้อมูลนั้นตามสคิม่าที่กำหนดไว้ตอนทำการคอนฟิกซ์เริ่มต้น แล้วทำการแยกข้อมูลแล้วเขียนลงบนฐานข้อมูล

ซึ่งการทำงานของระบบ โดยรวมสามารถเขียนเป็นยูสเคสไดอะแกรมได้ดังรูปที่ 3.2

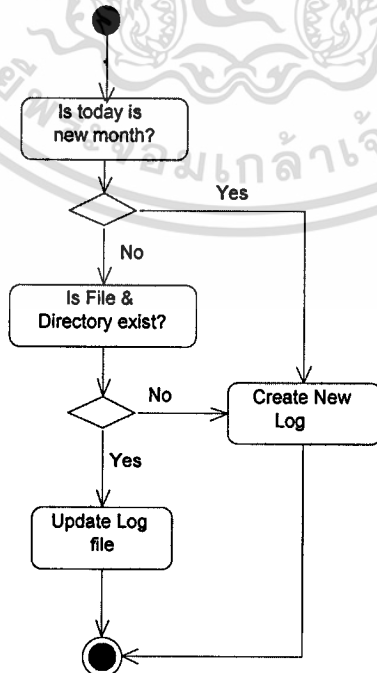
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงUsecase การสื่อสารข้อมูลข้ามระบบ

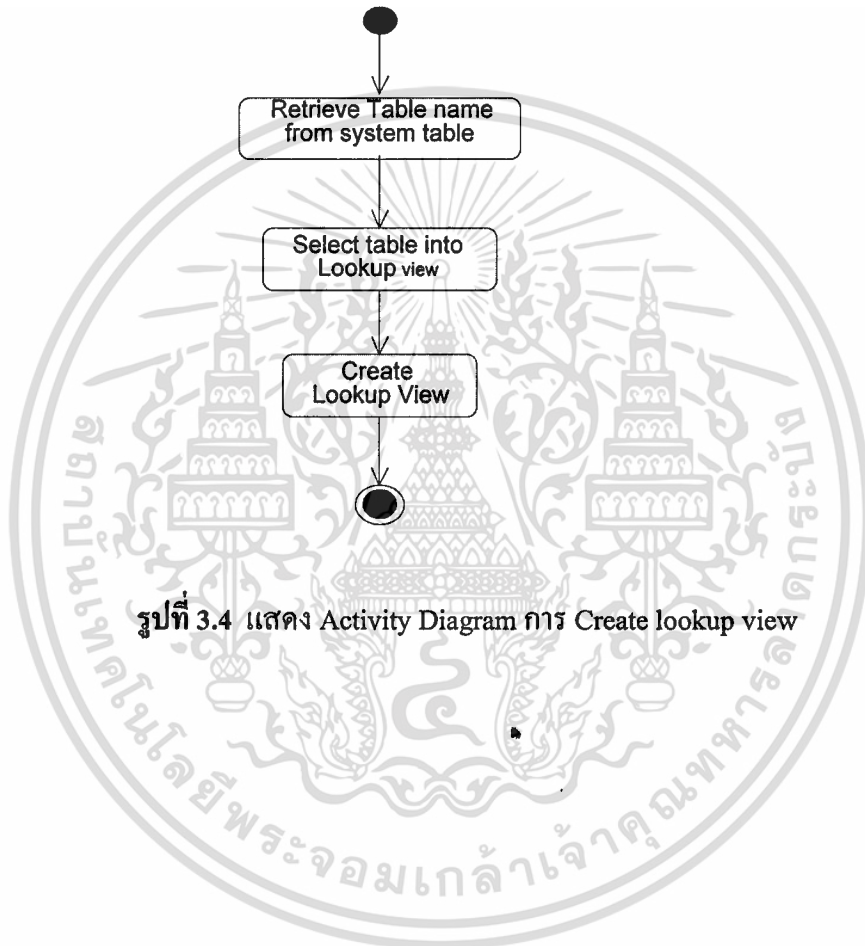
Activity Diagram

Configuration log file เป็นไดอะแกรมที่แสดงถึงการทำงานของระบบที่เก็บข้อมูลจากการทำคอนฟิกซ์ระบบลงล็อกไฟล์ โดยระบบจำทำการเจเนอเรตเป็นไฟล์ใหม่ทุกๆ เดือนโดยรันตามปีและเดือน ที่ทำการคอนฟิกซ์ ดังรูปที่ 3.3



รูปที่ 3.3 แสดง Activity Diagram การบันทึกข้อมูลลง Log file

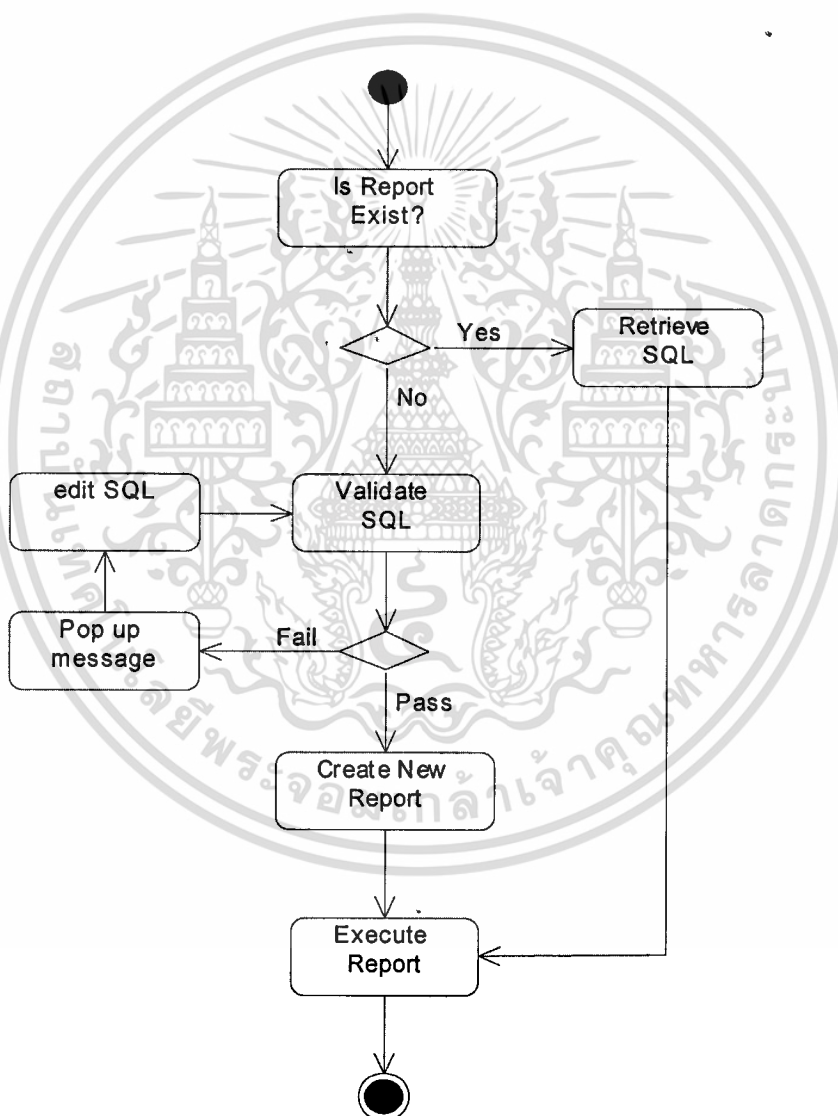
Create lookup view เป็นไดอะแกรมที่ใช้สำหรับสร้างตารางลูกคัพ ซึ่งจะมีความประโยชน์ในกรณีที่ระบบมีตารางอยู่มากมาย และตารางที่ใช้ในการทำอินเทอร์เน็ตเฟรมมีไม่กี่ตาราง โดยระบบจะทำการมองที่ซิสเต็มเทเบิล แล้วให้ผู้ใช้เลือกตารางที่ต้องการใช้เอง โดยสามารถเลือกได้ในระดับของฟิลด์ได้ด้วย ดังรูปที่ 3.4



รูปที่ 3.4 แสดง Activity Diagram การ Create lookup view

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

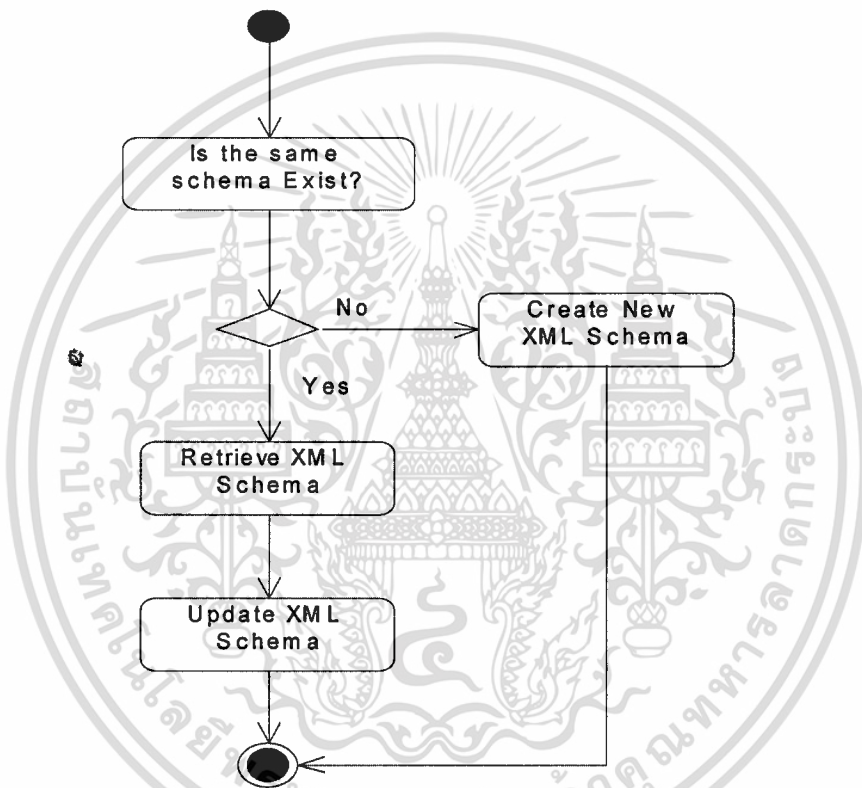
Create report เป็นโคแอมที่ใช้สำหรับให้ผู้ใช้สามารถสร้างรายงานดูข้อมูลที่ต้องการภายในระบบได้ เช่นต้องการดูว่าปัจจุบันระบบได้ส่งชุดข้อมูลออกไปยังระบบปลายทางเป็นจำนวนกี่ครั้ง หรือรับจากระบบปลายทางเข้าสู่ภายในระบบกี่ครั้งเป็นต้น ซึ่งผู้ใช้สามารถสร้างเป็น SQL ใหม่หรือดึงจากไฟล์ขึ้นมาทำการประมวลผลได้ ซึ่งระบบจะทำการตรวจสอบ SQL ถ้ามีข้อผิดพลาดจะแจ้งให้ผู้ใช้ทราบถึงความผิดพลาดที่เกิดขึ้น แต่ถ้าไม่พบความผิดพลาดใดๆ ระบบจะแสดงผลลัพธ์ออกมาทางหน้าจอ ดังรูปที่ 3.5



รูปที่ 3.5 แสดง Activity Diagram การ Create Report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

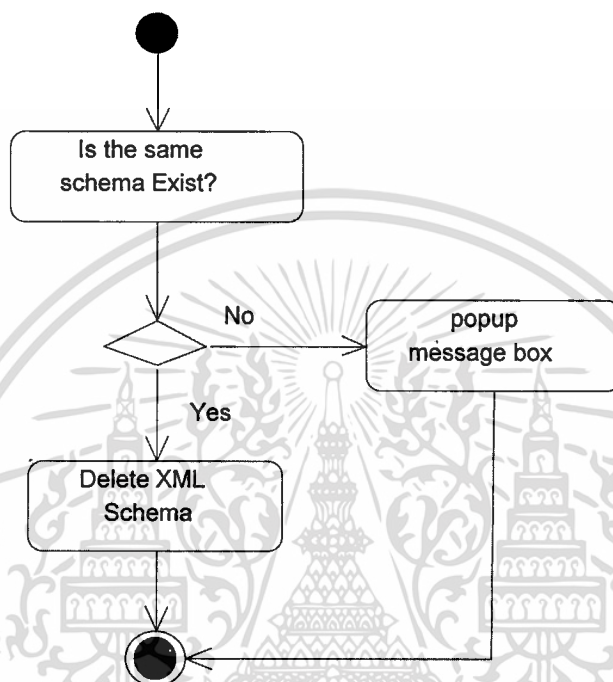
Create XML schema เป็น โค้ดโปรแกรมสำหรับสร้างสคีมาของชุดข้อมูลเอ็กซ์เอ็มแอล ซึ่งตัวสคีมาจะบอกถึงลักษณะข้อมูลของเอ็กซ์เอ็มแอลว่าประกอบด้วยแท็กอะไรบ้างและข้อมูลภายในแต่ละแท็กดึงมาจากตารางใด และคอลัมน์ใด แท็กแต่ละแท็กจะมีลักษณะเป็นแอททริบิวต์ ฟอเรสต์ หรือเป็นอีลีเมนต์ธรรมดาก็ได้ ซึ่งระบบสามารถสร้างสคีมาใหม่ หรือแก้ไขสคีมาที่มีอยู่ก็ได้ ดังรูปที่ 3.6



รูปที่ 3.6 แสดง Activity Diagram การ Create XML Schema

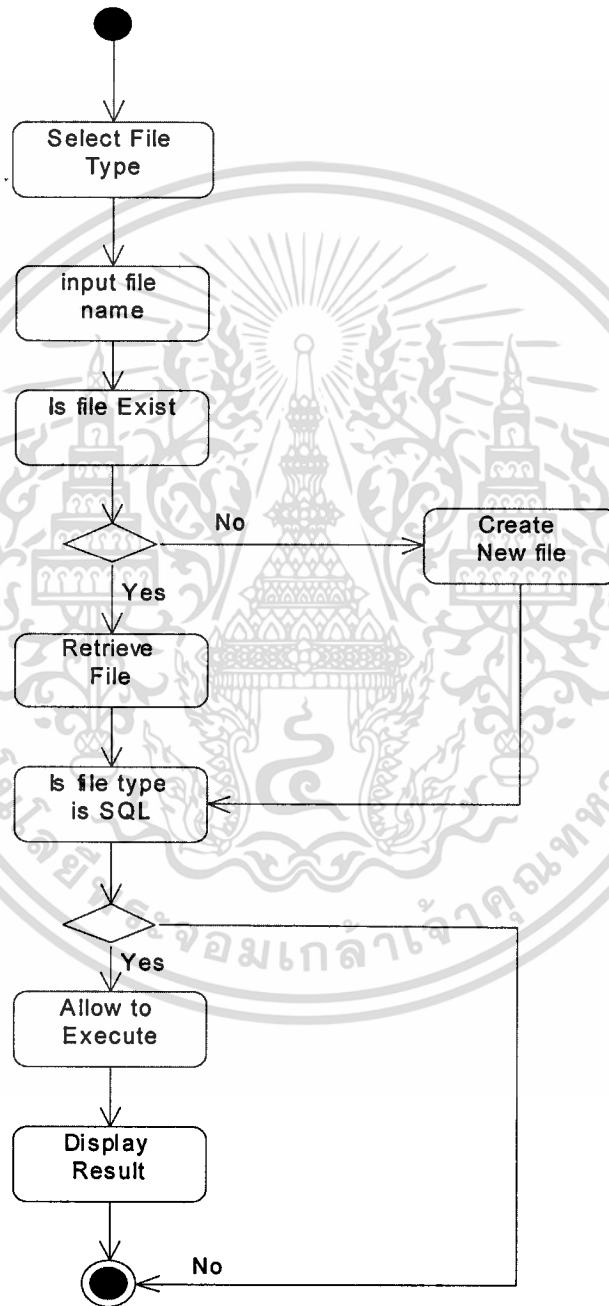
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Delete XML schema เป็นไคอะแกรมสำหรับลบสคีมามีอยู่ โดยระบบจะตรวจสอบว่า สคีมานั้นจะทำการลบมีอยู่ในระบบหรือไม่ ถ้าไม่มีระบบจะแจ้งให้ทราบว่ามีไม่พบสคีมานี้ในระบบ แต่ถ้ามีระบบจะอนุญาตให้ลบได้ ดังรูปที่ 3.7



รูปที่ 3.7 แสดง Activity Diagram ก15 Delete XML schema

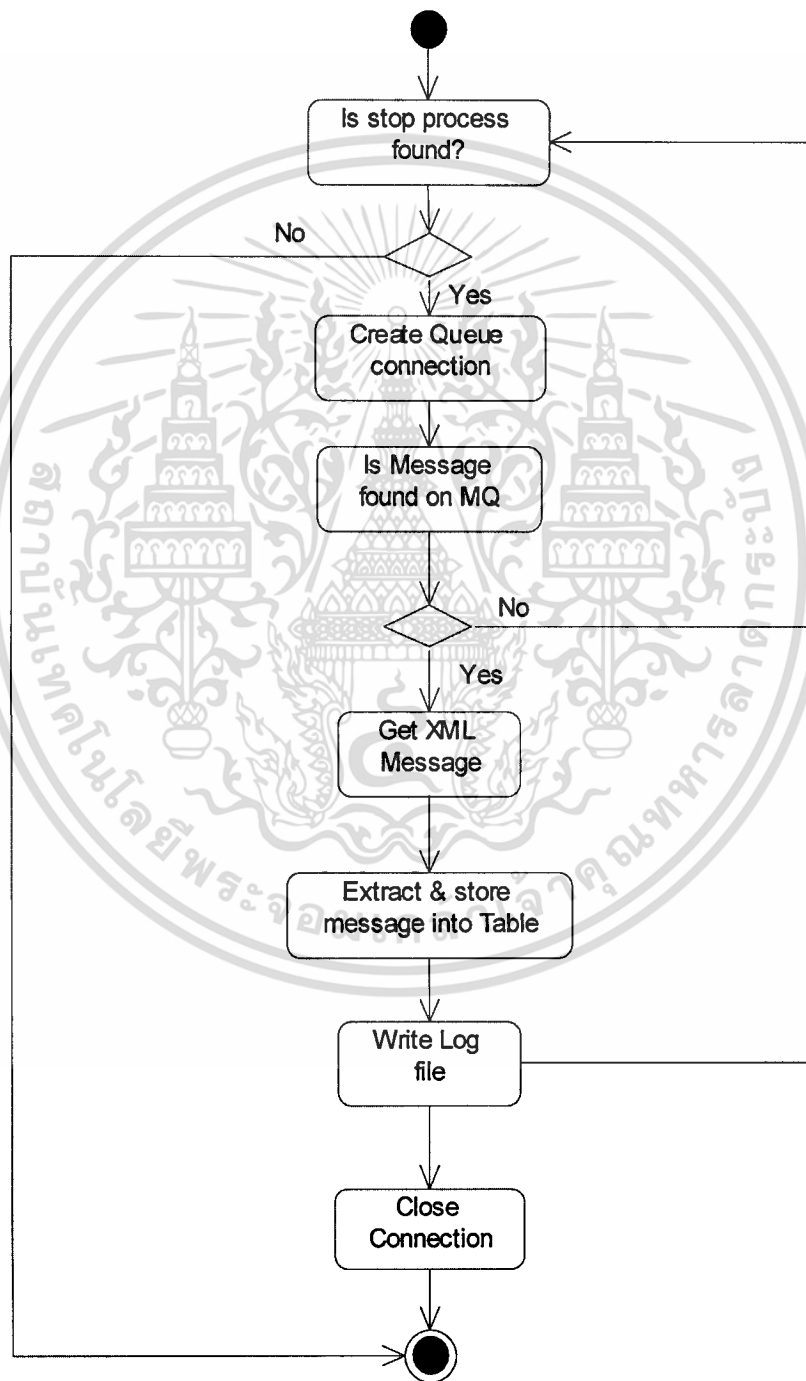
Editor เป็นโคอะแกรม สำหรับให้ผู้ใช้ทำการดึงข้อมูลจากตารางในระบบขึ้นมาดูแล้ว แสดงผลทางหน้า หรือทำการเปลี่ยนแปลงข้อมูลภายในตารางเช่น แทรกแถว, แก้ไขข้อมูลภายใน ตาราง, หรือลบข้อมูลภายในตาราง และอนุญาตให้ผู้ใช้สามารถสร้างทริกเกอร์และสโตร์โปรซีดเยอร์ได้ ดังรูปที่ 3.8



รูปที่ 3.8 แสดง Activity Diagram ของ Editor

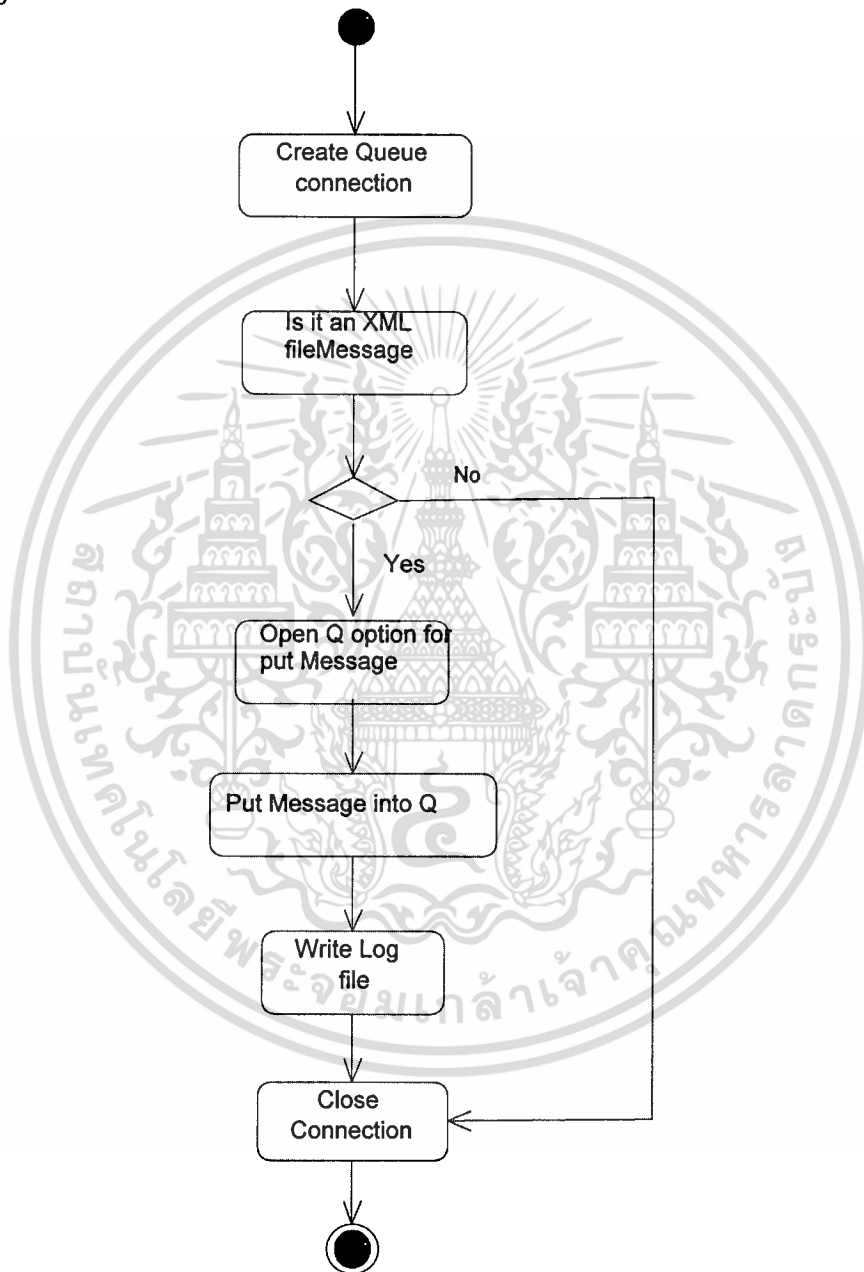
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Get message from MQ เป็นไคอะแกรม สำหรับการทำงานส่วนของแบคเอนด์ ซึ่งระบบจะทำการตรวจสอบว่ามีข้อมูลเข้ามาค้างที่คิวของฝั่งขาเข้าหรือไม่ ถ้ามีให้ดึงออกมาจากคิวแล้วทำการเอ็กซ์แทรคข้อมูลแล้วจัดเก็บลงตาราง โดยที่ระบบจะต้องทำการเก็บล็อกของการดึงข้อมูลจากคิวด้วย ดังรูปที่ 3.9



รูปที่ 3.9 แสดง Activity Diagram การ Get Message from Queue เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และใช้เพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

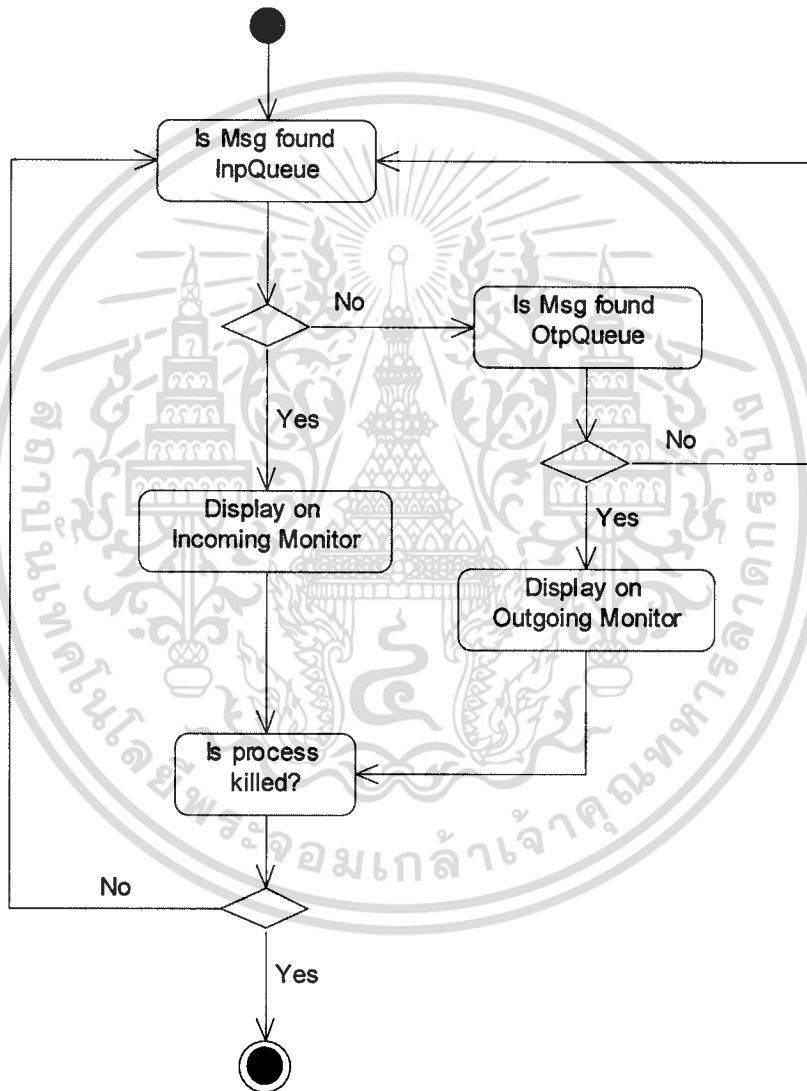
Put message into MQ เป็นไคอะแกรมสำหรับนำชุดข้อมูลที่ได้ในรูปของเอ็กซ์เอ็มแอลไฟล์ไปวางลงบนคิวฝั่งขาออก แล้วทำการเก็บล็อกทุกๆ ครั้งที่มีการนำข้อมูลไปวางบนคิวด้วย ดังรูปที่ 3.10



รูปที่ 3.10 แสดง Activity Diagram การ Put Message into Queue

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

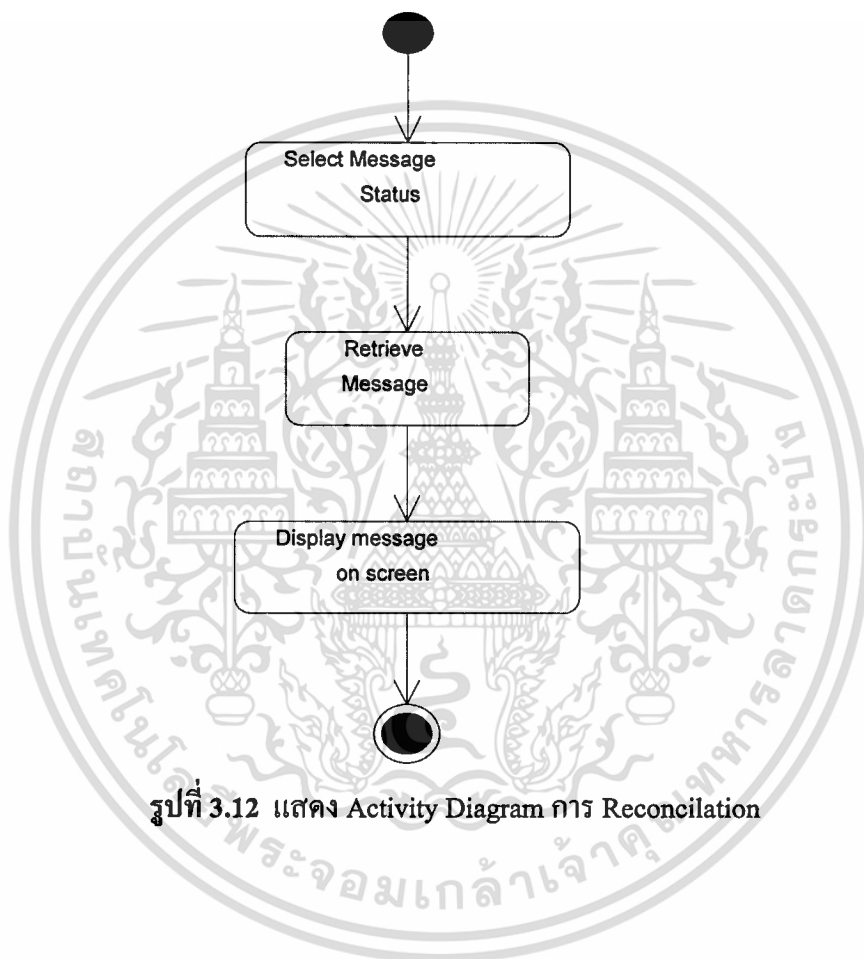
Message monitoring เป็นไคอะแกรมสำหรับมอนิเตอร์ความเคลื่อนไหวของชุดข้อมูลในคิวของเอ็มคิวซีรีส์ ซึ่งระบบจะคอยตรวจสอบอยู่ตลอดเวลาถ้าหากเจอข้อมูลในคิวจะแสดงผลออกทางหน้าจอ ดังรูปที่ 3.11



รูปที่ 3.11 แสดง Activity Diagram กับการ Monitor Message

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

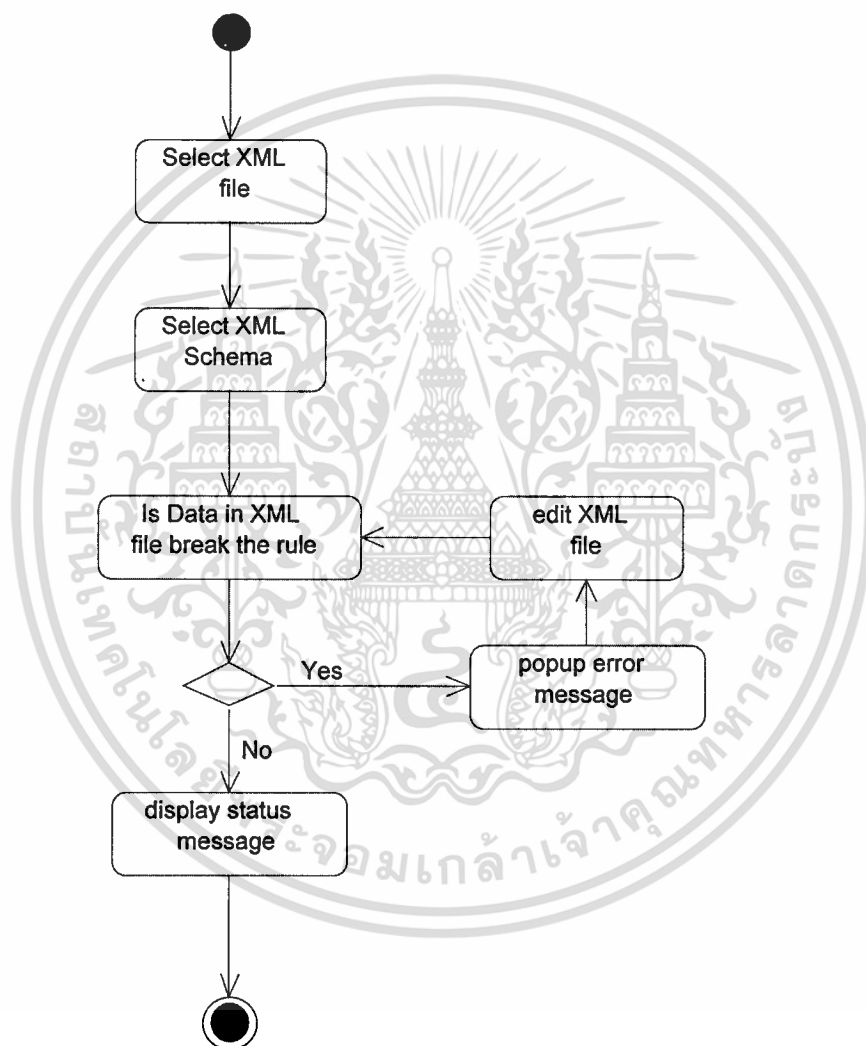
Reconciliation เป็นโคอะแกรมสำหรับตรวจสอบจำนวนของชุดข้อมูลที่ระบบทำการรับเข้ามาจากปลายทาง และส่งออกไปสู่ระบบปลายทาง โดยจะต้องเทียบระบบภายในกับระบบปลายทางแล้วแจ้งผลการตรวจสอบที่เกิดขึ้นทางหน้าจอ ดังรูปที่ 3.12



รูปที่ 3.12 แสดง Activity Diagram การ Reconciliation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XML validation เป็นไคอะแกรมสำหรับตรวจสอบความถูกต้องของ เอ็กซ์เอ็มแอลที่มีกับ สคีม่าที่ได้สร้างไว้ในระบบ ถ้ามีข้อผิดพลาดเกิดขึ้นระบบจะทำการแจ้งผลที่เกิดขึ้นให้ทราบ และ ผู้ใช้สามารถทำการแก้ไขข้อมูลในเอ็กซ์เอ็มแอลได้ แล้วสามารถทำการตรวจสอบต่อไปจนกว่าผลที่ได้ จะถูกต้อง ดังรูปที่ 3.13

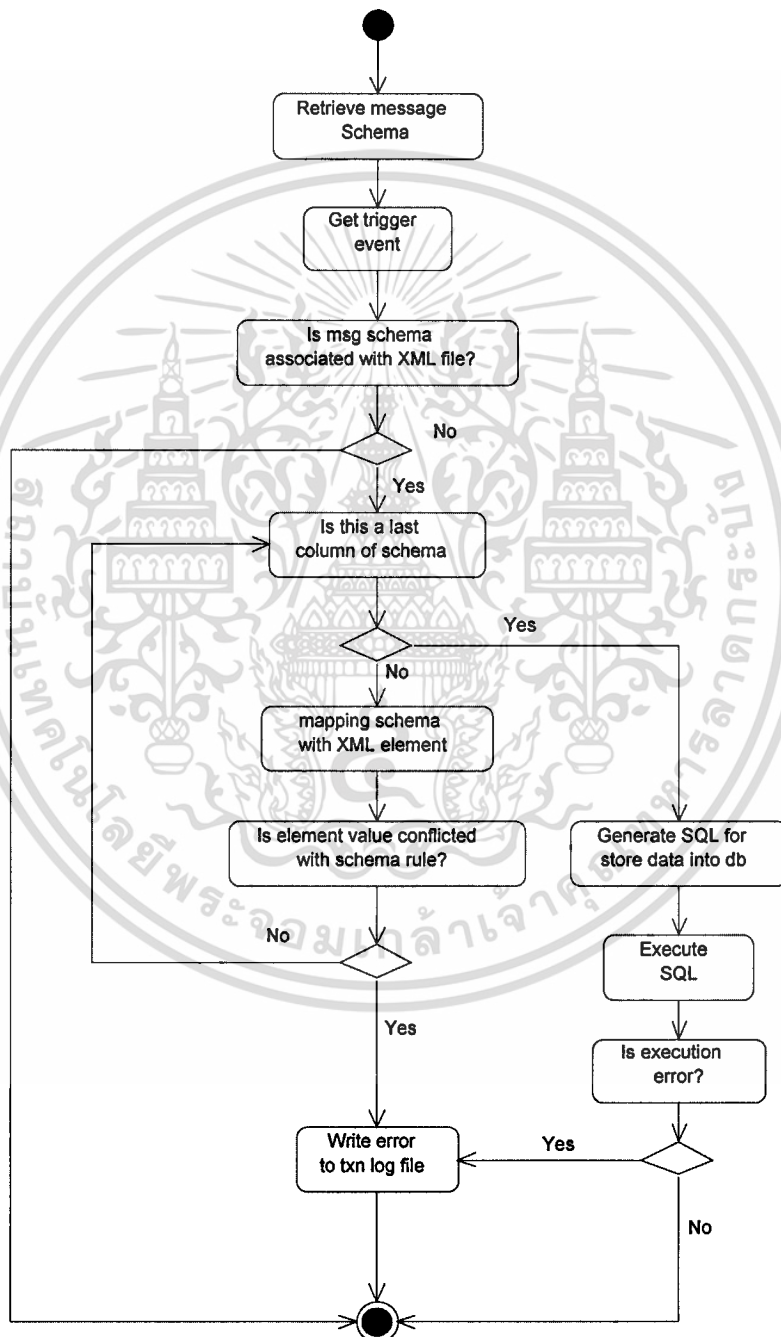


รูปที่ 3.13 แสดง Activity Diagram การ Validate XML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XML extractor เป็นไดอะแกรมสำหรับรับเอ็กซ์เอ็มแอลที่ได้จากระบบภายนอกเข้ามา เทียบกับสคีมามีอยู่ภายในระบบ แล้วทำการเอ็กซ์แทรคข้อมูลแล้วจัดเก็บลงในฐานข้อมูล ดังรูปที่

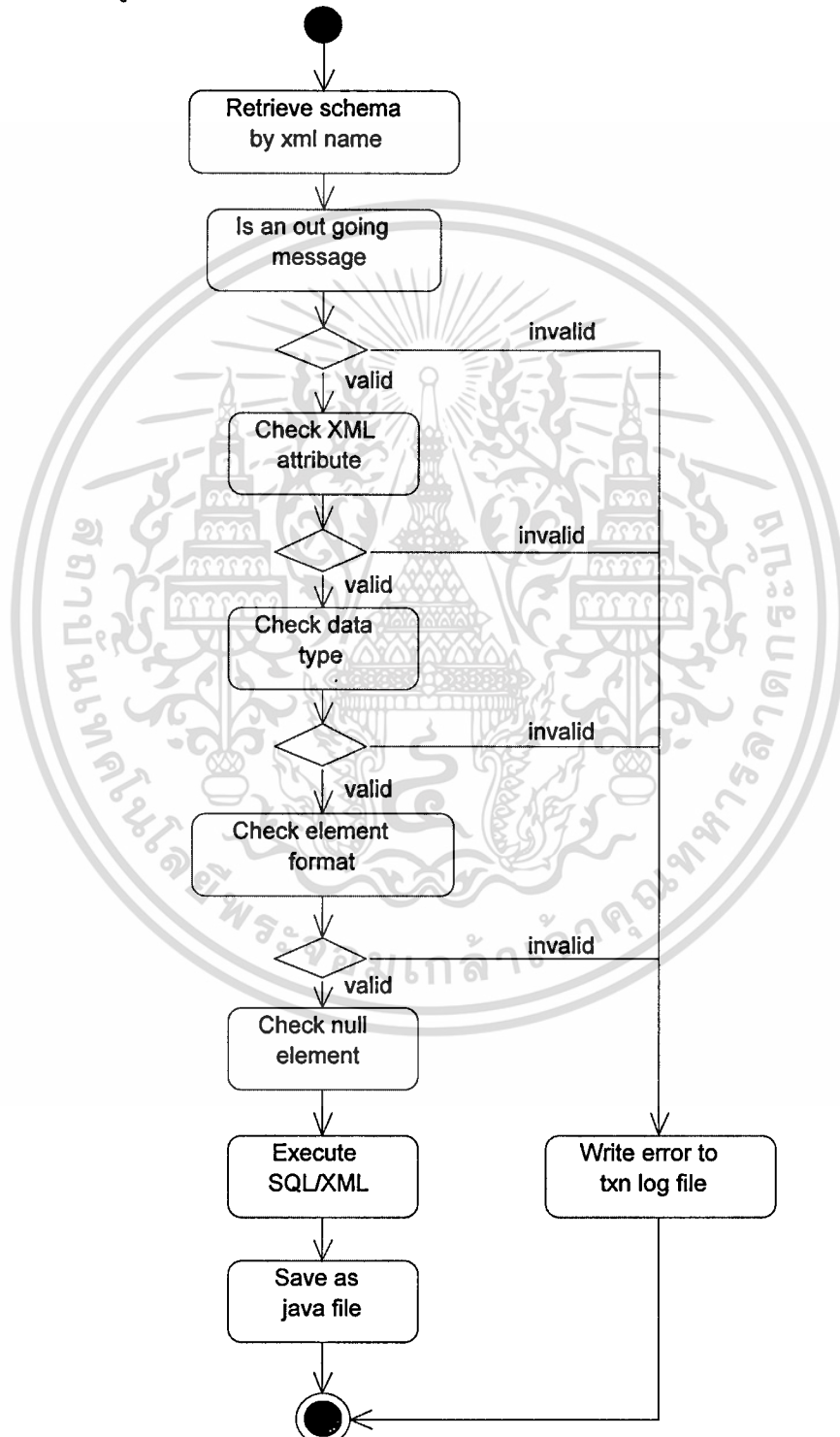
3.14



รูปที่ 3.14 แสดง Activity diagram ของ XML extractor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

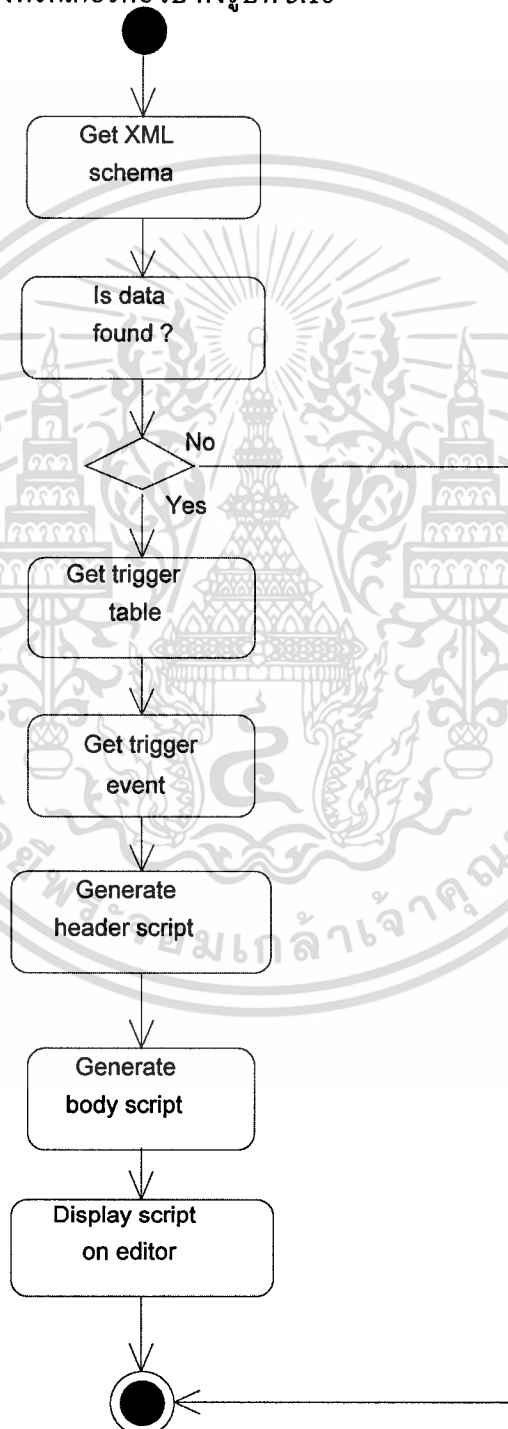
XML generator เป็นโคแอมแกรมสำหรับเจเนอเรตข้อมูลตามสคิม่าที่กำหนดให้อยู่ในรูปแบบของเอ็กซ์เอ็มแอลไฟล์ ถ้ามีข้อผิดพลาดใดๆ เกิดขึ้นขณะเจเนอเรตเอ็กซ์เอ็มแอลระบบจะทำการแจ้งออกทางหน้าจอ ดังรูปที่ 3.15



รูปที่ 3.15 แสดง Activity diagram ของ XML generator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญูญาติให้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

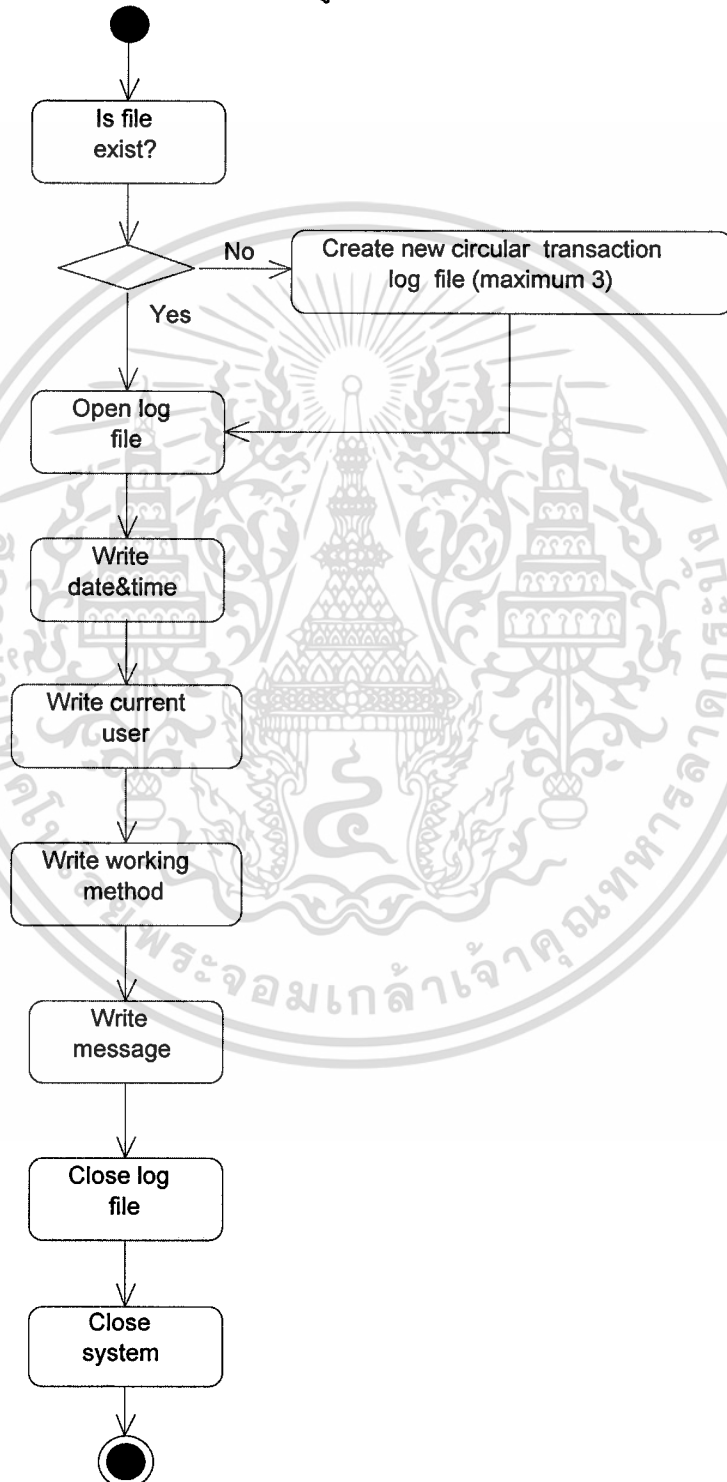
Generate trigger script เป็นไดอะแกรมสำหรับเจเนอเรตทริกเกอร์สคริปต์ตามสคิม่าที่เลือก ซึ่งระบบจะตรวจสอบว่าสคิม่าที่เลือกผูกอยู่กับตารางใด และมีเหตุการณ์ใดที่ทำให้ระบบต้องทำการเจเนอเรตเอ็กซ์เอ็มแอลแล้วทำการเจเนอเรตเป็นทริกเกอร์สคริปต์แล้วแสดงสคริปต์ที่ได้ออกทางหน้าจอ ให้ผู้ใช้งานไปสร้างทริกเกอร์ต่อไป ดังรูปที่ 3.16



รูปที่ 3.16 แสดง Activity diagram ของ Generate trigger script

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Write transaction log file เป็นไคอะแกรมสำหรับเก็บล็อกของการติดต่อกับคิวซึ่งระบบจะทำการวนการสร้างล็อกทั้งหมด 3 ไฟล์ด้วยกันถ้าไฟล์แรกเต็มระบบจะสร้างไฟล์ที่ 2 และ 3 ต่อไปเมื่อไฟล์ที่ 3 เต็มระบบจึงจะวนทับไฟล์ที่ 1 ดังรูปที่ 3.17

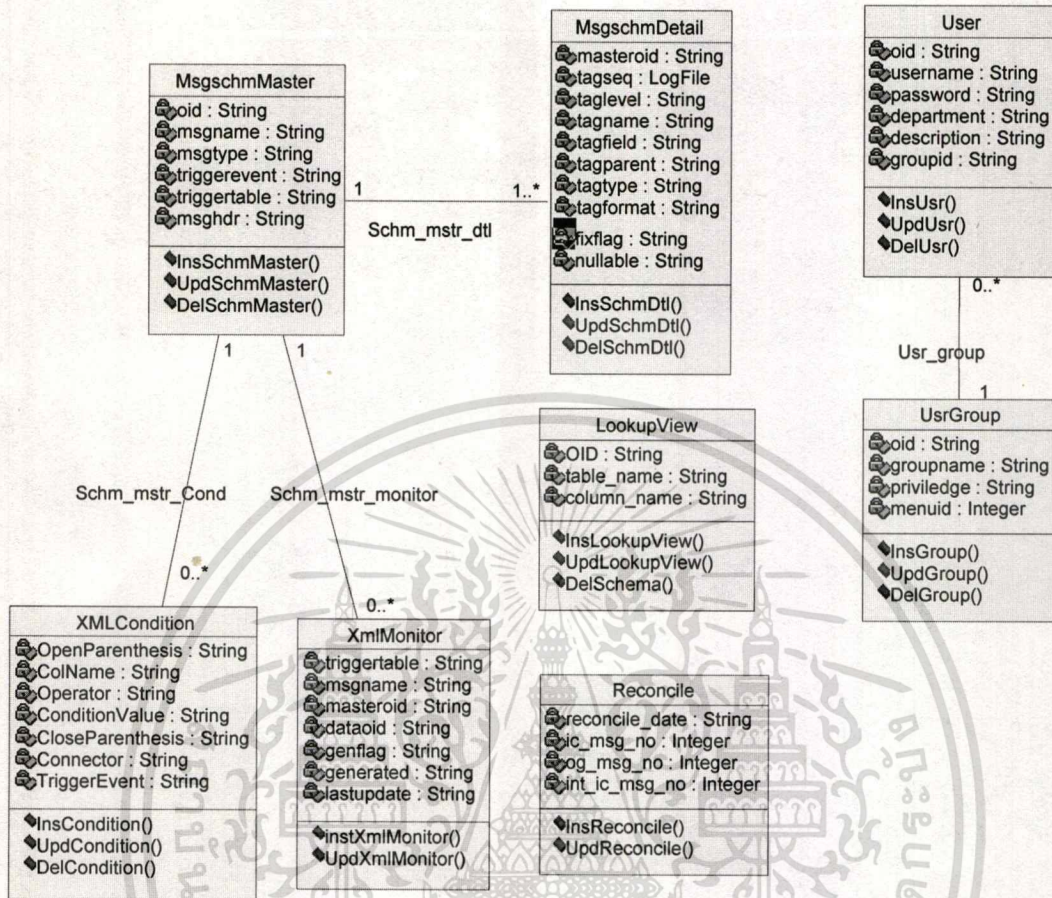


รูปที่ 3.17 แสดง Activity diagram ของ write transaction log file

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Class diagram แสดงคลาสและความสัมพันธ์ระหว่างคลาสของระบบซึ่งประกอบด้วยคลาสทั้งหมด 9 คลาสดังนี้

- MsgschmMaster เป็นคลาสที่แสดงถึงข้อมูลหลักของสคีมา เช่น ชื่อของชุดข้อมูล ประเภทของชุดข้อมูล เหตุการณ์ที่เป็นตัวขับเคลื่อนให้ระบบสร้างไฟล์เอ็กซ์เอ็มแอล ตารางที่เป็นตัวขับเคลื่อนให้เกิดการสร้างไฟล์เอ็กซ์เอ็มแอล
  - MsgschmDetail เป็นคลาสที่แสดงถึงรายละเอียดของสคีมา เช่น ประกอบด้วย แท็กอะไรบ้าง และข้อมูลภายในแท็กดึงมาจากตารางใด และคอลัมน์ใดลักษณะของแท็กเป็นแบบแอททริบิวต์, ฟอเรสต์ หรือเป็นอีลีเมนต์ ข้อมูลภายในแท็กเป็นชนิดอะไร สามารถเป็น null ได้หรือไม่ เป็นต้น
  - XMLCondition เป็นคลาสที่แสดงถึงเงื่อนไขในการสร้างไฟล์เอ็กซ์เอ็มแอล
  - XMLMonitor เป็นคลาสที่แสดงรายการข้อมูลที่จะทำการสร้างไฟล์เอ็กซ์เอ็มแอล ซึ่งจะมีแฟล็กบอกรายการข้อมูลใดบ้างที่รอการเจเนอเรต และมีรายการข้อมูลใดที่ได้เจเนอเรตออกไปแล้ว
  - LookupView เป็นคลาสที่แสดงชื่อตารางและคอลัมน์ในตารางที่ระบบสามารถมองเห็นและนำไปใช้งานได้
  - Reconcile เป็นคลาสที่แสดงจำนวนชุดของข้อมูลที่ระบบได้ส่งออกไปยังระบบปลายทางและรับจากระบบปลายทางเข้ามา เพื่อเทียบรายการรับและส่งข้อมูลกับระบบปลายทางและเป็นการตรวจสอบความถูกต้องของการรับและส่งข้อมูลในแต่ละวัน
  - User เป็นคลาสที่แสดงรายชื่อผู้ใช้ และรหัสผ่าน ที่สามารถเข้าใช้ระบบได้ โดยจะผูกความสัมพันธ์กับสิทธิ์ในการเข้าใช้งานระบบ
  - UserGroup เป็นคลาสที่แสดงสิทธิ์ในการเข้าใช้งานระบบ
- ซึ่งการทำงานของระบบโดยรวมสามารถเขียนเป็นคลาสไดอะแกรมได้ดังรูปที่ 3.18



รูปที่ 3.18 แสดง Class diagram ของระบบ

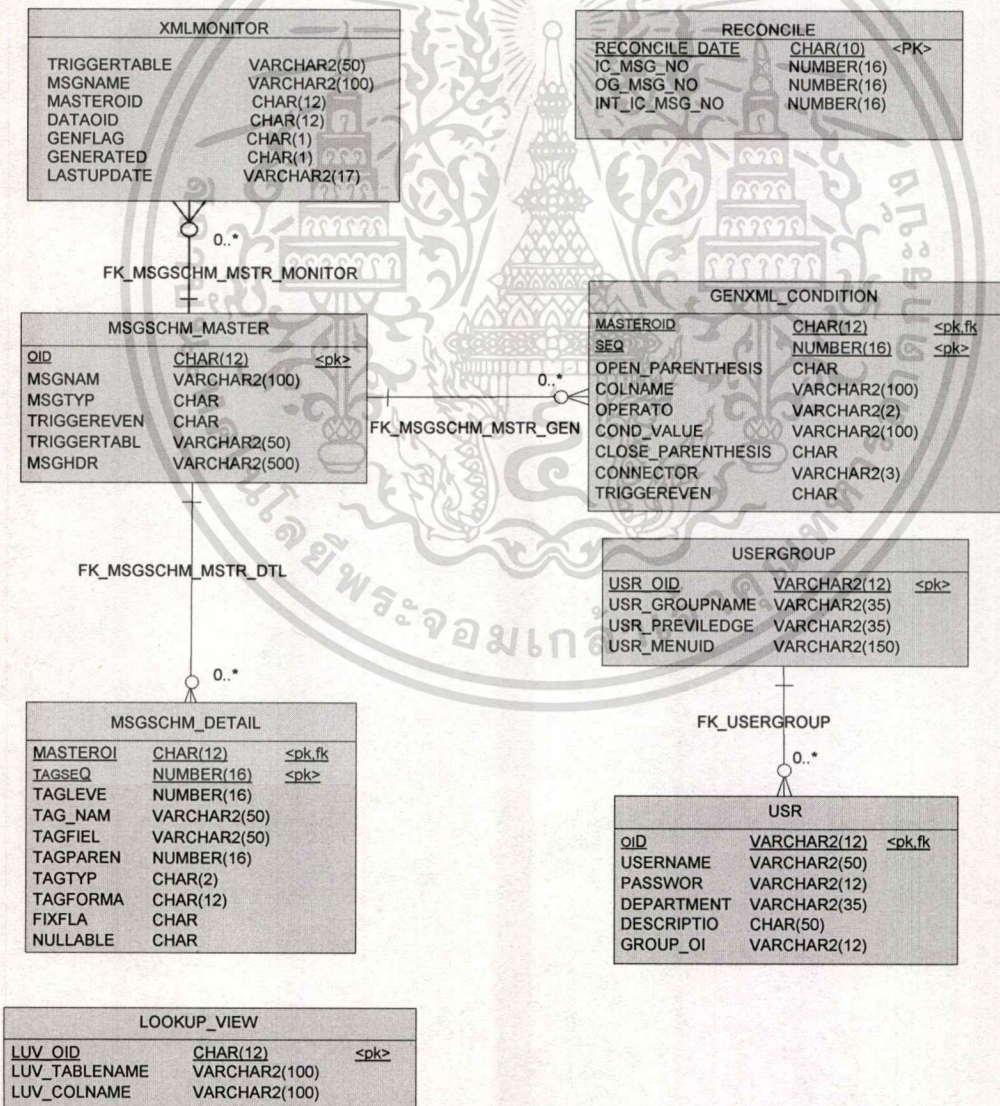
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การออกแบบฐานข้อมูล

#### 4.1 การออกแบบฐานข้อมูล

ระบบตัวกลางการสื่อสารข้อมูลข้ามแพลตฟอร์มด้วย เอ็มคิวซีรีส์ ใช้ Oracle เป็นฐานข้อมูล ซึ่งประกอบด้วยตารางที่เกี่ยวข้องทั้งหมด 7 ตาราง ซึ่งสามารถแปลงจาก class diagram ให้อยู่ในรูปของฐานข้อมูลเชิงสัมพันธ์ดังแสดงในรูปที่ 4.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.1 แสดง ER-Diagram ของระบบ  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- MSG\_SCHM\_MASTER เป็นตารางที่ใช้เก็บข้อมูลหลักของเอ็กซ์เอ็มแอลสคีมาดังแสดงตารางที่ 4.1
- MSG\_SCHM\_DETAIL เป็นตารางที่เก็บรายละเอียดของเอ็กซ์เอ็มแอลสคีมาดังตารางที่ 4.2
- RECONCILE เป็นตารางที่ใช้เก็บข้อมูลของจำนวนชุดข้อมูลที่มีการเข้าสู่ระบบและออกสู่นอกระบบ ดังแสดงตารางที่ 4.3
- LOOKUP\_VIEW เป็นตารางที่ใช้เก็บข้อมูลตารางและคอลัมน์ เพื่อเจาะจงให้ระบบมองเห็นได้เฉพาะที่กำหนดไว้ในตารางนี้เท่านั้น ดังแสดงตารางที่ 4.4
- REPORT เป็นตารางที่ใช้เก็บข้อมูลรายละเอียดเอสคิวแอลสคริปต์ที่ใช้ในการออกรายงาน ซึ่งแต่ละรายงานจะมีการระบุได้เป็นเอสคิวแอลหรือจะทำเป็นสโตร์โพรซีเจอร์แล้ววัน PL/SQL ก็ได้ ดังแสดงตารางที่ 4.5
- USER เป็นตารางที่ใช้เก็บรายละเอียดของผู้ใช้ที่สามารถเข้าใช้ระบบ ดังแสดงตารางที่ 4.6
- USERGROUP เป็นตารางที่ใช้เก็บรายละเอียดของกลุ่มของผู้ใช้งานระบบซึ่งระบบสามารถกำหนดสิทธิ์การเข้าใช้งานระบบเป็นกลุ่ม ดังแสดงตารางที่ 4.7
- GENXML\_CONDITION เป็นตารางที่เก็บเงื่อนไขการสร้างเอ็กซ์เอ็มแอลดังตารางที่ 4.8

ตารางที่ 4.1 รายละเอียดฐานข้อมูลของตาราง MSGSCHM\_MASTER

| Fields Name  | Description  | Data Type    | Key | Null | Ref. table |
|--------------|--|--------------|-----|------|------------|
| Oid          | Object ID  | Varchar(12)  | PK  | N    |            |
| MsgName      | ชื่อของ XML  | Varchar(100) |     | N    |            |
| MsgType      | ประเภทของไฟล์ XML<br>-(I) Incoming Message<br>-(O)Outgoing Message | Char(1)      |     | N    |            |
| TriggerEvent | เหตุการณ์ที่ทำให้เกิดการสร้างไฟล์                                  | Char(1)      |     | N    |            |
| TriggerTable | ชื่อตารางที่เก็บข้อมูลเพื่อใช้สร้างไฟล์                            | varchar(50)  |     | N    |            |
| MsgHdr       | Message Header   | varchar(500) |     | Y    |            |
| MsgTailor    | Message Tailor   | varchar(500) |     | Y    |            |
| GenFlag      | แฟล็กฟิลด์ใช้เพื่อระบุการสร้างไฟล์                                 | char(1)      |     | Y    |            |
| Generated    | แฟล็กฟิลด์ที่ใช้เพื่อระบุว่าได้มีการสร้างไฟล์ไปแล้วหรือยัง         | Char(1)      |     | Y    |            |

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 รายละเอียดฐานข้อมูลของตาราง MSG\_SCHM\_DETAIL

| Fields Name | Description  | Data Type   | Key       | Null | Ref. table             |
|-------------|--|-------------|-----------|------|------------------------|
| MasterOid   | Object ID ของตารางแม่ เพื่อเป็นคีย์ต่าง                          | Varchar(12) | PK<br>,FK | N    | MSGSCH<br>M_MAS<br>TER |
| TagSeq      | ลำดับของ Tag(Auto Generate)                                      | Number      | PK        | N    |                        |
| TagLevel    | ลำดับชั้นของ Tag   | Number      | N         | Y    |                        |
| Tag_name    | ชื่อ Tag   | Varchar(50) | N         | Y    |                        |
| TagField    | ชื่อของฟิลด์ที่ใช้เก็บข้อมูลของ Tag นั้นๆ                        | Varchar(50) | N         | Y    |                        |
| TagParent   | Tag แม่ของ Tag ปัจจุบัน<br>ถ้าเป็น Null แสดงว่าตัวมันเป็น Parent | Number      | N         | Y    |                        |
| TagType     | ประเภทของ Tag  | Char(2)     | N         | Y    |                        |
| TagFormat   | รูปแบบของข้อมูลที่แสดงบน ไฟล์ XML                                | Char(1)     | N         | Y    |                        |
| FixFlag     | ค่าของ tag มีการ fix ได้หรือไม่                                  | Char(1)     | N         | Y    |                        |
| NullAble    | ค่าของ tag เป็น null ได้หรือไม่                                  | Char(1)     | N         | Y    |                        |
| ParentType  | ประเภทของความสัมพันธ์ tag  | Char(1)     | N         | Y    |                        |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 รายละเอียดฐานข้อมูลของตาราง RECONCILE

| Fields Name    | Description                                     | Data Type | Key | Null | Ref. table |
|----------------|---|-----------|-----|------|------------|
| Reconcile_date | วันที่ทำการเทียบ                                | Char(10)  | PK  | N    |            |
| ic_msg_no      | จำนวน message ที่เข้าสู่ระบบของระบบปลายทาง      | Number    | N   | Y    |            |
| og_msg_no      | จำนวน message ที่ออกสู่ระบบภายนอกของระบบปลายทาง | Number    | N   | Y    |            |
| int_ic_msg_no  | จำนวน message ที่เข้าสู่ระบบของระบบภายใน        | Number    | N   | Y    |            |

ตารางที่ 4.4 รายละเอียดฐานข้อมูลของตาราง LOOKUP\_VIEW

| Fields Name   | Description                | Data Type    | Key | Null | Ref. table |
|---------------|----------------------------|--------------|-----|------|------------|
| Luv_OID       | Object ID ของ Lookup table | varchar(12)  | PK  | N    |            |
| Luv_TableName | ชื่อ Table                 | varchar(100) | N   | N    |            |
| Luv_ColName   | ชื่อ Column                | varchar(100) | N   | N    |            |

ตารางที่ 4.5 รายละเอียดฐานข้อมูลของตาราง REPORT

| Fields Name    | Description  | Data Type   | Key | Null | Ref. table |
|----------------|--|-------------|-----|------|------------|
| Rpt_OID        | Object ID ของ Report   | varchar(12) | PK  | N    |            |
| Rpt_ReportName | ชื่อรายงาน   | varchar(50) | N   | N    |            |
| Rpt_SPName     | ชื่อ Store procedure ที่เก็บ SQL ที่ใช้ในการ Retrieve ข้อมูลเข้าสู่ Report | varchar(50) | N   | N    |            |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 รายละเอียดฐานข้อมูลของตาราง USR

| Fields Name     | Description                                       | Data Type   | Key | Null | Ref. table |
|-----------------|---|-------------|-----|------|------------|
| Usr_OID         | Object ID ของผู้ใช้งานระบบ                        | varchar(12) | PK  | N    |            |
| Usr_UserName    | ชื่อผู้มีสิทธิ์เข้าใช้งานระบบ                     | varchar(50) | N   | N    |            |
| Usr_Password    | รหัสผ่านผู้มีสิทธิ์เข้าใช้งานระบบ                 | varchar(12) | N   | N    |            |
| Usr_department  | ฝ่ายงานที่ผู้ใช้งานระบบสังกัดอยู่                 | varchar(35) | N   | N    |            |
| Usr_description | รายละเอียดอื่นๆ เพิ่มเติม                         | char(50)    | N   | N    |            |
| Usr_groupOID    | กลุ่มของสิทธิ์ของผู้ใช้ที่สามารถเข้าใช้งานระบบได้ | char(50)    | FK  | N    | USER GROUP |

ตารางที่ 4.7 รายละเอียดฐานข้อมูลของตาราง USERGROUP

| Fields Name   | Description   | Data Type    | Key | Null | Ref. table |
|---------------|---|--------------|-----|------|------------|
| Usr_OID       | Object ID ของตาราง USERGROUP  | varchar(12)  | PK  | N    |            |
| Usr_groupname | กลุ่มของสิทธิ์ในการเข้าใช้งาน   | varchar(35)  | N   | N    |            |
| Usr_Privilege | สิทธิ์ในการเข้าใช้งานระบบ เช่น สิทธิ์ในการอ่านข้อมูล, สิทธิ์ในการเพิ่มข้อมูล, สิทธิ์ในการแก้ไขข้อมูล, สิทธิ์ในการลบข้อมูล | varchar(35)  | N   | N    |            |
| Usr_menuid    | รายละเอียดเมนูที่ผู้ใช้สามารถเข้าใช้งาน หรือมองเห็นได้  | varchar(150) | N   | N    |            |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 รายละเอียดฐานข้อมูลของตาราง GENXML\_CONDITION

| Fields Name       | Description   | Data Type    | Key       | Null | Ref. table             |
|-------------------|---|--------------|-----------|------|------------------------|
| masteroid         | Object ID ของตาราง<br>XMLSCHM_MASTER  | varchar(12)  | PK,<br>FK | N    | MSGSCH<br>M_MAS<br>TER |
| seq               | ลำดับของ เงื่อนไขในการ generate<br>XML(Auto Generate)   | Number       | PK        | N    |                        |
| open_parenthesis  | วงเล็บเปิด จะมีหรือไม่ก็ได้ แต่<br>บางครั้งการกำหนดเงื่อนไขจำเป็น<br>จะต้องมีวงเล็บเพื่อให้ได้เงื่อนไข<br>ตรงกับความต้องการ | Char(1)      | N         | Y    |                        |
| colname           | ชื่อคอลัมน์ใน table   | varchar(100) | N         | N    |                        |
| operator          | โอเปอเรเตอร์ เช่น =, <, <=, ><br>, >=   | varchar(2)   | N         | N    |                        |
| cond_value        | ค่าที่ต้องการกำหนดให้เงื่อนไข   | varchar(100) | N         | N    |                        |
| close_parenthesis | วงเล็บปิดจะมีหรือไม่ก็ได้ แต่ถ้ามี<br>วงเล็บเปิดจะต้องมีวงเล็บปิดเสมอ   | Char(1)      | N         | Y    |                        |
| connector         | ตัวเชื่อมเงื่อนไข เช่น And, Or, Not<br>ไม่จำเป็นต้องใส่ถ้า expression นั้น<br>เป็นตัวสุดท้ายของประโยค                       | varchar(3)   | N         | Y    |                        |
| triggerEvent      | เหตุการณ์ที่เป็นตัวกระตุ้นให้ ต้องมี<br>การเรียก script   | Char(1)      | N         | N    |                        |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### การพัฒนาระบบ

จากการศึกษา วิเคราะห์ และออกแบบระบบงานที่ได้กล่าวมาแล้วนั้น ทำให้สามารถทำการพัฒนาระบบงาน โดยแยกตามลักษณะของการทำงานได้เป็น 6 ส่วน คือ การกำหนดค่าเริ่มต้นของระบบ, การ Monitor ความเคลื่อนไหวของการสื่อสารข้อมูล, การตรวจสอบความถูกต้องของการสื่อสารข้อมูล, การออกรายงานโดยจากหน้าจอเอดิเตอร์ (Editor), การสร้างตารางดูคัพ (Lookup view) และการ ตรวจสอบความถูกต้องของไฟล์เอ็กซ์เอ็มแอล (XML Validation)

โดยในบทนี้ จะกล่าวถึงการพัฒนาโปรแกรมและผลที่ได้จากการพัฒนาโปรแกรมโดยแสดงออกมาเป็นลักษณะจอภาพส่วนต่างๆ ของโปรแกรมได้ ดังนี้

เริ่มการใช้งานระบบโดยจะมีการตรวจสอบผู้ใช้ และสิทธิการเข้าใช้งานดังรูปที่ 5.1



รูปที่ 5.1 แสดงหน้าจอการตรวจสอบสิทธิการเข้าใช้งานระบบ

ที่ แท็บเพจ Configuration จะเป็นหน้าจอสำหรับตั้งค่าเริ่มต้นก่อนจะใช้งานระบบ ซึ่งหน้าจอจะประกอบด้วย 3 ส่วนหลักๆ ดังนี้

1. Message Configuration จะประกอบไปด้วย 3 ส่วนดังนี้

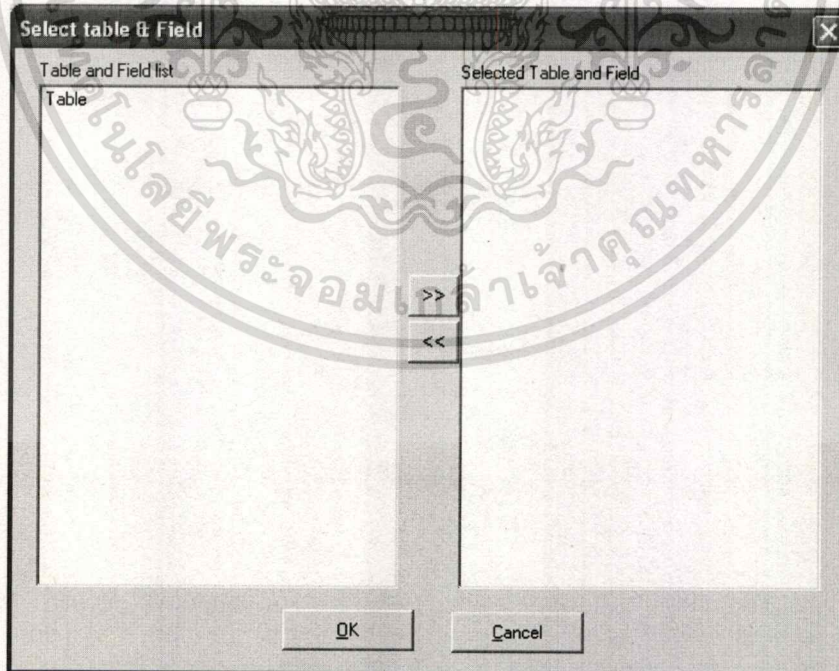
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ส่วนของการกำหนด สัญลักษณ์เพื่อระบุแท็กซึ่งจะเป็นการกำหนดสัญลักษณ์เพื่อใช้เป็นตัวกำหนดแท็กเปิด(Begin Tag) และ แท็กปิด (End Tag) เช่น `<Name> Chitlada Rodyoy </Name>` จะต้องกำหนดแท็กเปิดเป็น `<>` และแท็กปิด เป็น `</>`
  2. ส่วนของการกำหนด Configuration Log Path ซึ่งจะเป็นการกำหนด path ที่ใช้เก็บ Configuration Log คือทุกครั้งที่มีการเปลี่ยนค่า Configuration file จะมีการเก็บ Log ไว้ด้วยทุกครั้ง
  3. ส่วนของการกำหนด ทรานแซกชัน Log Path ซึ่งจะเป็นการกำหนด path ที่ใช้เก็บ ทรานแซกชัน Log นั่นคือจะเก็บความเคลื่อนไหวของทุกๆ ทรานแซกชันว่าในแต่ละวันมีการส่ง file หรือ รับ file อะไรบ้าง และจะเป็นประโยชน์ในการทำ Reconciliation ด้วย
2. Message Schema Master เป็นส่วนที่ใช้เพื่อกำหนดคุณสมบัติต่างๆ ของ message จะประกอบไปด้วย 4 ส่วนดังนี้
1. Msg File name คือ ชื่อของชุดข้อมูลที่ใช้ในการเชื่อมต่อระหว่างระบบ
  2. Msg Type คือ ประเภทของชุดข้อมูลซึ่งจะมีทั้งหมด 2 ประเภท คือ
    1. Incoming message คือชุดข้อมูลที่จะรับเข้ามาจากระบบภายนอก
    2. Outgoing message คือชุดข้อมูลที่จะส่งออกไปยังระบบภายนอก
  3. Trigger Event คือ เหตุการณ์ที่จะกระตุ้นให้ระบบทำการสร้างชุดข้อมูลในรูปแบบของไฟล์เอ็กซ์เอ็มแอลซึ่งจะมีทั้งหมด 4 เหตุการณ์คือ
    1. Insert เมื่อมีการแทรกข้อมูลเข้ามายังตารางที่ระบุไว้ในสคิม่า
    2. Update เมื่อมีการแก้ไขหรือเปลี่ยนแปลงข้อมูลมายังตารางและคอลัมน์ที่ระบุ ซึ่งสามารถระบุได้ว่าถ้ามีการเปลี่ยนแปลงจะต้องเป็นค่าที่ระบุเท่านั้น ระบบถึงจะทำการเจเนอเรตไฟล์
    3. Delete เมื่อมีการลบข้อมูลของตารางที่ระบุไว้ในสคิม่า
    4. On Request เมื่อระบบได้รับคำสั่งจากผู้ใช้
  4. Trigger Table คือ ลิสต์ของ table ทั้งหมดที่อยู่ใน Lookup View
    - Trigger Column ในกรณีที่เลือกเหตุการณ์เป็น update ระบบจะเปิดช่อง Trigger Column ให้สามารถใช้งานได้
    - Trigger Value ในกรณีที่เลือกเหตุการณ์เป็น update ระบบจะเปิดช่อง Trigger Column ให้สามารถใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ซึ่งงานเพื่อเป็นประโยชน์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Message Schema Detail เป็นส่วนที่ใช้เพื่อกำหนดรายละเอียด ของชุดข้อมูล จะประกอบไปด้วย 9 ส่วนดังนี้

1. MasterOID คือ Identification Number ของ message schema master table ซึ่งเป็น คีย์ต่าง (foreign key) ของตาราง message schema detail
2. TagSeq คือ ลำดับที่ของแท็กในไฟล์เอ็กซ์เอ็มแอล
3. TagLevel คือ ระดับของแท็กในไฟล์เอ็กซ์เอ็มแอล
4. TagName คือ ชื่อของ
5. TagField คือ คอลัมน์ของตาราง ซึ่งจะเป็นตัวที่ระบุว่าข้อมูลในแท็กนี้ถูกดึงมาจากคอลัมน์ใดในตาราง ซึ่งการใส่ข้อมูลใน TagField ผู้ใช้สามารถใส่ได้ 2 วิธีคือ
  - คีย์เข้า
  - ดับเบิ้ลคลิกภายในฟิลด์เพื่อเปิด หน้าจอ ลิสต์รายการของตารางที่มี โดยผู้ใช้สามารถคลิก drag and drop เพื่อเลือก หลังจากนั้น กด OK เพื่อกลับไปสู่หน้าจอหลัก หากผู้ใช้ต้องการยกเลิกการเลือกสามารถยกเลิกได้โดยการปิดหน้าจอจาก ปุ่ม Cancel หรือ กดกากบาท ดังรูปที่ 5.2



รูปที่ 5.2 แสดงหน้าจอการเลือกฟิลด์เพื่อใช้ในการดึงข้อมูลสำหรับเจเนอเรตไฟล์เอ็กซ์เอ็มแอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. TagParent ในกรณีนี้ที่แท็กมีลักษณะเป็นแม่-ลูก แล้ว สามารถระบุแท็กแม่ได้

เช่น <Name>

<First Name> Chitlada</FirstName>

<MidName>Wilai</MidName>

<LastName>Rodyoy</LastName>

</Name>

7. TagType คือชนิดข้อมูลของแท็กซึ่งแบ่งได้เป็น 4 ประเภท คือ

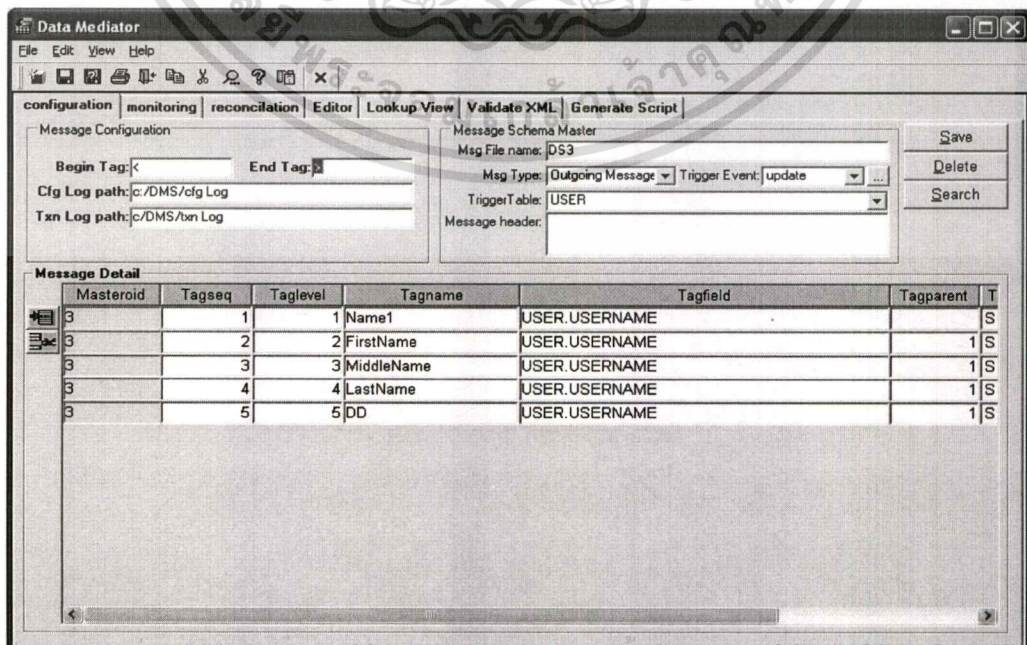
1. Date
2. DateTime
3. String
4. Numeric

8. TagFormat คือ รูปแบบของข้อมูล เช่น

@@/@@/@@@@

@-@@@@-@@@@

9. FixFlag คือ แฟล็กที่ใช้ระบุว่าข้อมูลในแฟล็กนี้เป็นข้อมูลคงที่ ซึ่ง ถ้าแฟล็กเป็น Yes ระบบจะนำข้อมูลที่ระบุไว้ที่ TagField เป็น ข้อมูลจริงในลงในไฟล์เอ็กซ์เอ็มแอล แทนที่การดึงมาจากฟิลด์ แต่ถ้าแฟล็กเป็น No หมายถึงระบบจะทำการดึงข้อมูลจากตารางและคอลัมน์ที่กำหนดมาทำการเจเนอเรตลงในไฟล์เอ็กซ์เอ็มแอลดังรูปที่ 5.3



เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่ออนุญาตให้เผยแพร่ไปยังระบบอื่นใดเป็นการค้า  
รูปที่ 5.3 แสดงหน้าจอ Configuration  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ แท็บเพจ Monitoring จะเป็นหน้าจอเพื่อดูการรับและส่งข้อมูลของระบบ ซึ่งจะแบ่งเป็น 2 ส่วน คือ

1. Incoming Message คือ ส่วนที่ดูความเคลื่อนไหวของชุดข้อมูลที่จะวิ่งเข้าสู่ระบบ เมื่อ message เข้าสู่ระบบเรียบร้อยแล้วข้อมูลของชุดข้อมูลตัวนั้นก็จะหายไปจากหน้าจอ
2. Outgoing Message ส่วนที่ดูความเคลื่อนไหวของ ชุดข้อมูล ที่จะวิ่งออกจากระบบ เมื่อชุดข้อมูลเข้าสู่ระบบปลายทางเรียบร้อยแล้วข้อมูลของชุดข้อมูลตัวนั้นก็จะหายไปจากหน้าจอ ดังรูปที่ 5.4

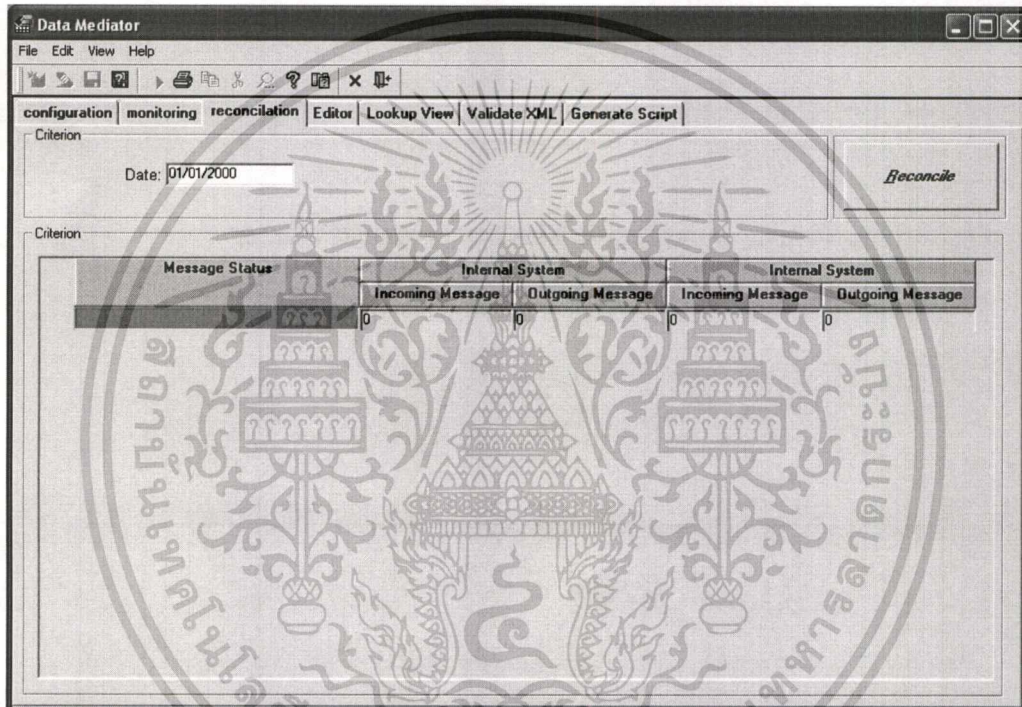


รูปที่ 5.4 แสดงหน้าจอการ Monitor การทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ Tabpage Reconciliation เราสามารถตรวจสอบการทำงานโดยเลือกดูจากสถานะของข้อมูลได้ ซึ่งมีสถานะของmessage เป็น 2 แบบ คือ

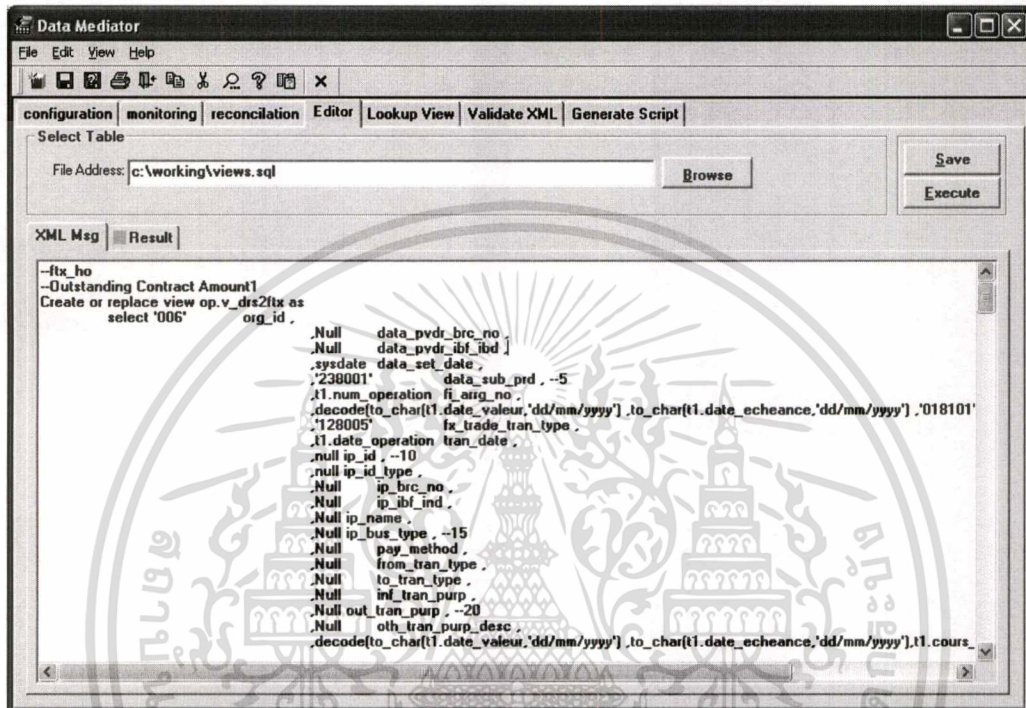
1. Complete ตรวจสอบว่ามีชุดข้อมูลใดบ้างที่มีการทำ interface ได้สำเร็จ คือ สามารถส่งออก และรับเข้าได้ตรงกัน
2. Error ตรวจสอบว่ามีชุดข้อมูลใดบ้างที่มีการทำ interface ได้ไม่สำเร็จ คือ ระบบต้นทางและปลายทางไม่สามารถสื่อสารได้ถูกต้องตรงกันดังรูปที่ 5.5



รูปที่ 5.5 แสดงหน้าจอการ Reconciliation เพื่อตรวจสอบการทำงานของระบบ

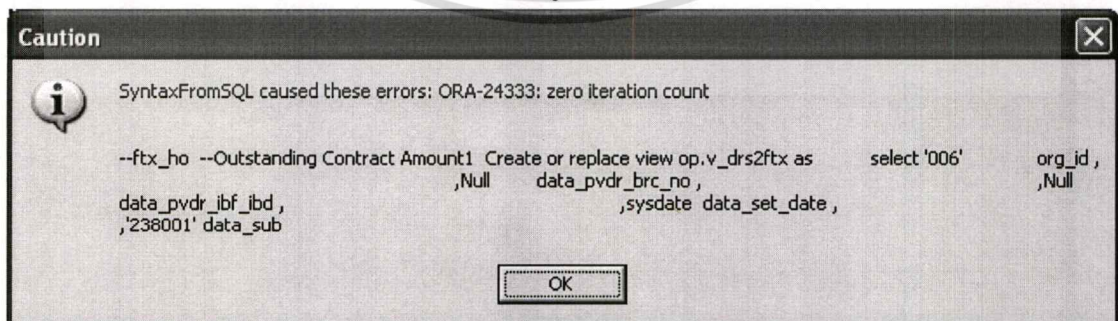
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ Tabpage Editor เป็นหน้าจอสำหรับแสดงผลไฟล์ใดๆ และนอกจากนี้ยังสามารถใช้เป็น Editor เพื่อใช้ retrieve ข้อมูลจากฐานข้อมูลได้ด้วยดังรูปที่ 5.6



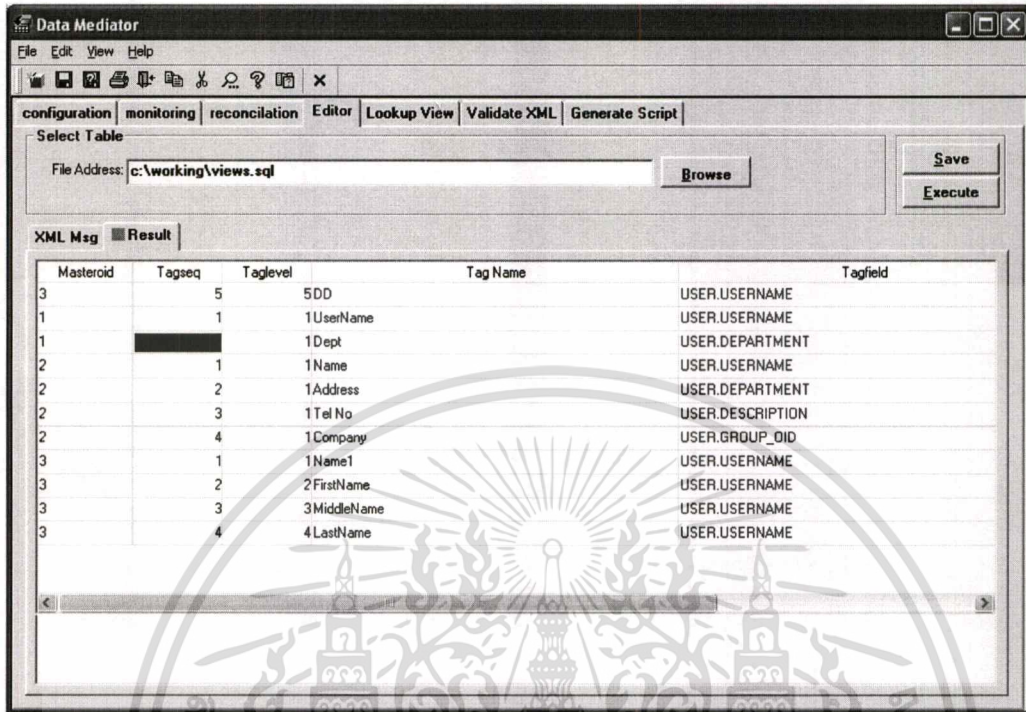
รูปที่ 5.6 แสดงหน้าจอ Editor สำหรับแสดงและแก้ไขข้อมูลในไฟล์

เมื่อผู้ใช้คลิกปุ่ม Execute ที่รันคำสั่งจะปรากฏผลการรันที่ tabpage Result ซึ่งหากผลของการรันคำสั่งถูกต้อง ไฟที่อยู่หน้า tabpage Result จะเปลี่ยนจากสีเทา เป็นสีเขียว แต่ถ้ามีข้อผิดพลาดเกิดขึ้น ไฟจะเปลี่ยนจากสีเทาเป็นสีแดง และมีข้อความแจ้งเตือนดังรูปที่ 5.7



รูปที่ 5.7 แสดงหน้าจอ แจ้งให้ผู้ใช้ทราบหากมีความผิดพลาดเกิดขึ้นจากการรันคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



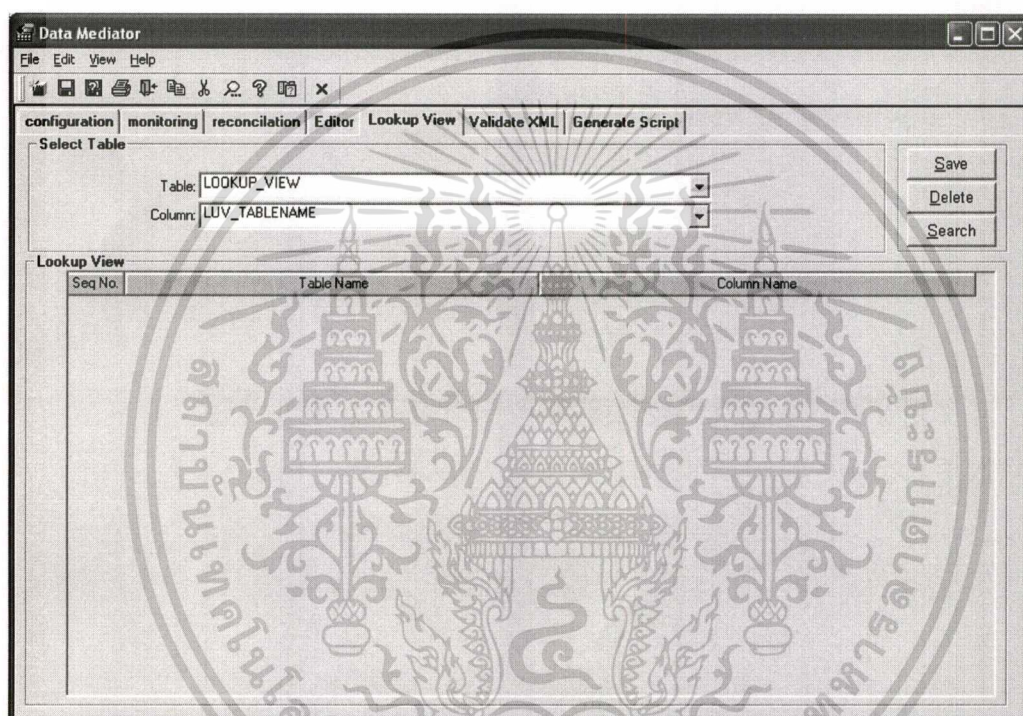
The screenshot shows the Data Mediator interface with a table of XML tags. The table has columns for Masterid, Tagseq, Taglevel, Tag Name, and Tagfield. The data is as follows:

| Masterid | Tagseq | Taglevel    | Tag Name | Tagfield        |
|----------|--------|-------------|----------|-----------------|
| 3        | 5      | 5DD         |          | USER.USERNAME   |
| 1        | 1      | 1UserName   |          | USER.USERNAME   |
| 1        |        | 1Dept       |          | USER.DEPARTMENT |
| 2        | 1      | 1Name       |          | USER.USERNAME   |
| 2        | 2      | 1Address    |          | USER.DEPARTMENT |
| 2        | 3      | 1Tel No     |          | USER.DESCRPTION |
| 2        | 4      | 1Company    |          | USER.GROUP_OID  |
| 3        | 1      | 1Name1      |          | USER.USERNAME   |
| 3        | 2      | 2FirstName  |          | USER.USERNAME   |
| 3        | 3      | 3MiddleName |          | USER.USERNAME   |
| 3        | 4      | 4LastName   |          | USER.USERNAME   |

รูปที่ 5.8 แสดงหน้าจอ แสดงผลการรันคำสั่งที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

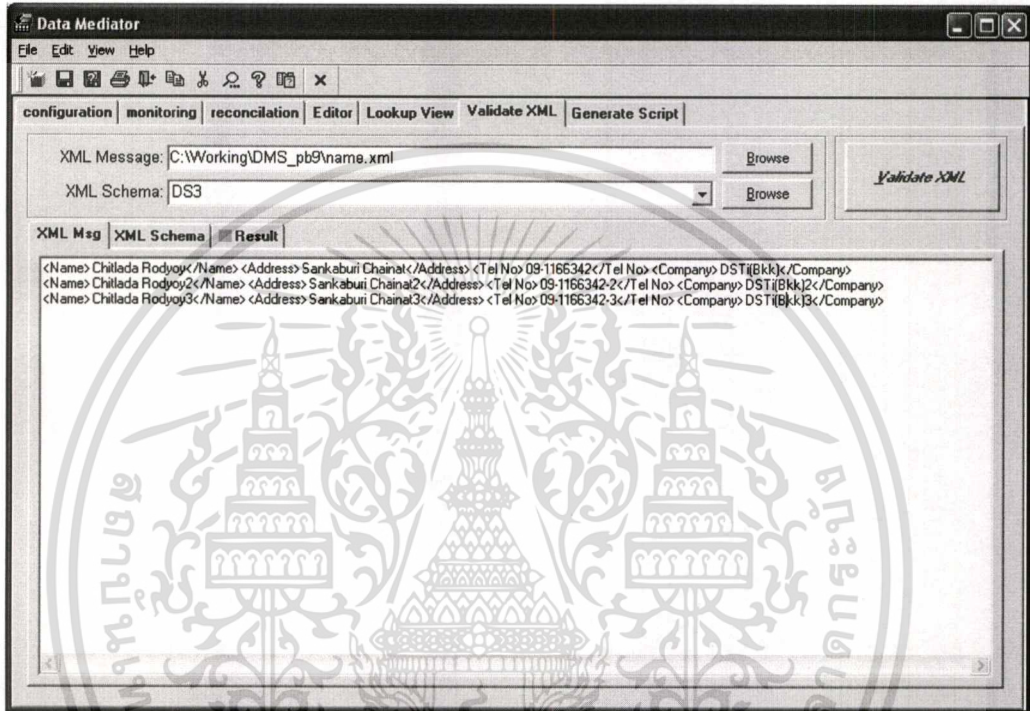
ที่ Tabpage Lookup View เป็นส่วนของการกำหนด Lookup Table ที่จะใช้ในระบบซึ่งในกรณีที่เป็นระบบงานขนาดใหญ่ จะมี Table มากมายในฐานข้อมูล แต่มี Table เพียงไม่กี่ Table ที่จะใช้เพื่อการทำ interface ดังนั้นจะไม่สะดวกหากจะแสดง Table ทั้งหมด ดังนั้นหน้าจอนี้จึงเป็นการกำหนด table เพื่อจะใช้ในการทำ interface และนอกจากนี้แล้วระบบยังลงรายละเอียดไปจนถึงคอลัมน์อีกด้วย



รูปที่ 5.9 แสดงหน้าจอ Lookup View เพื่อใช้กำหนด Lookup Table

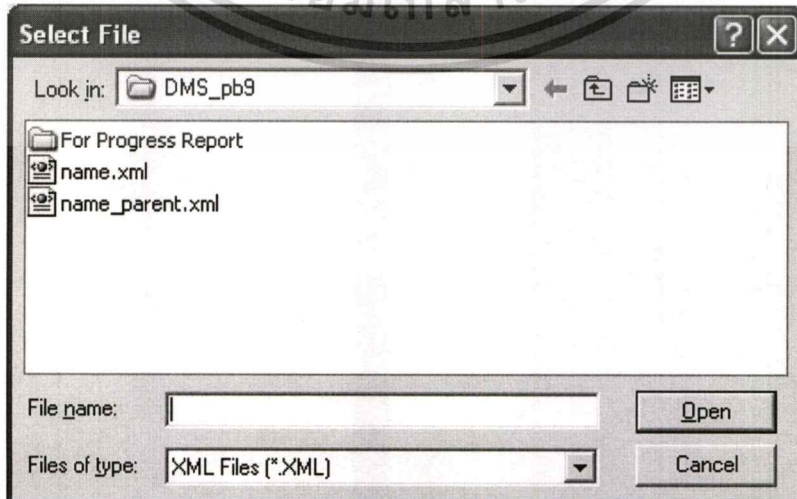
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ Tabpage Validate XML เป็นส่วนของการตรวจสอบไฟล์เอ็กซ์เอ็มแอลกับสคิม่าว่าเอ็กซ์เอ็มแอลที่ได้ตรงกับถูกต้องตรงกับสคิม่าที่ระบุหรือไม่ซึ่งถ้าไม่ถูกต้องระบบจะทำการแจ้งผลของการตรวจสอบความถูกต้อง แต่ถ้าถูกต้องระบบจะแจ้งว่าชุดข้อมูลที่ได้ถูกต้องตรงกับสคิมาดังรูปที่ 5.10



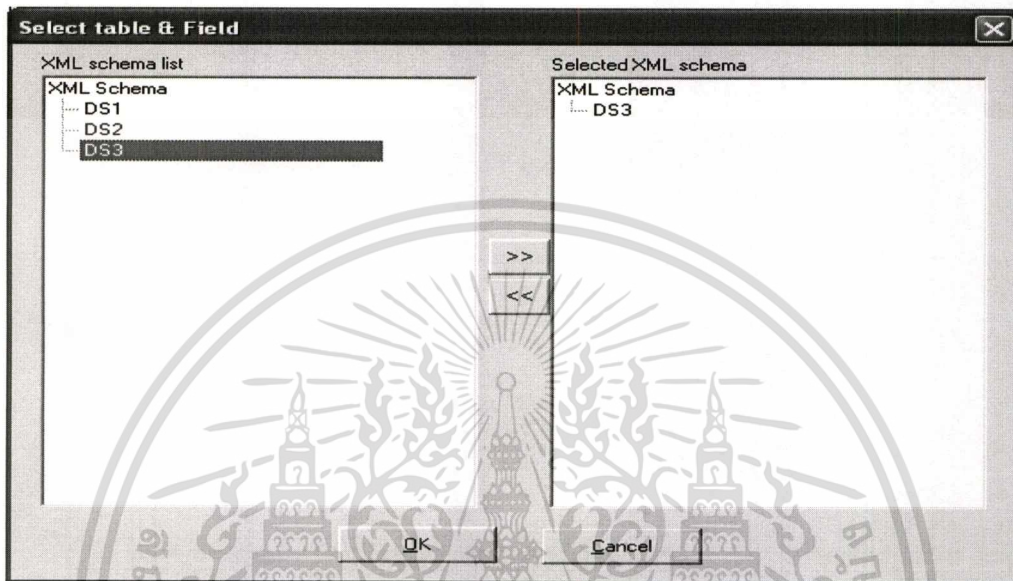
รูปที่ 5.10 แสดงหน้าจอการ Validate XML

ผู้ใช้สามารถคลิกปุ่ม browse เพื่อเลือกไฟล์เอ็กซ์เอ็มแอลที่ได้บันทึกไว้ในดิสค์ ซึ่งจะปรากฏหน้าจอค้นหาไฟล์ดังรูปที่ 5.11



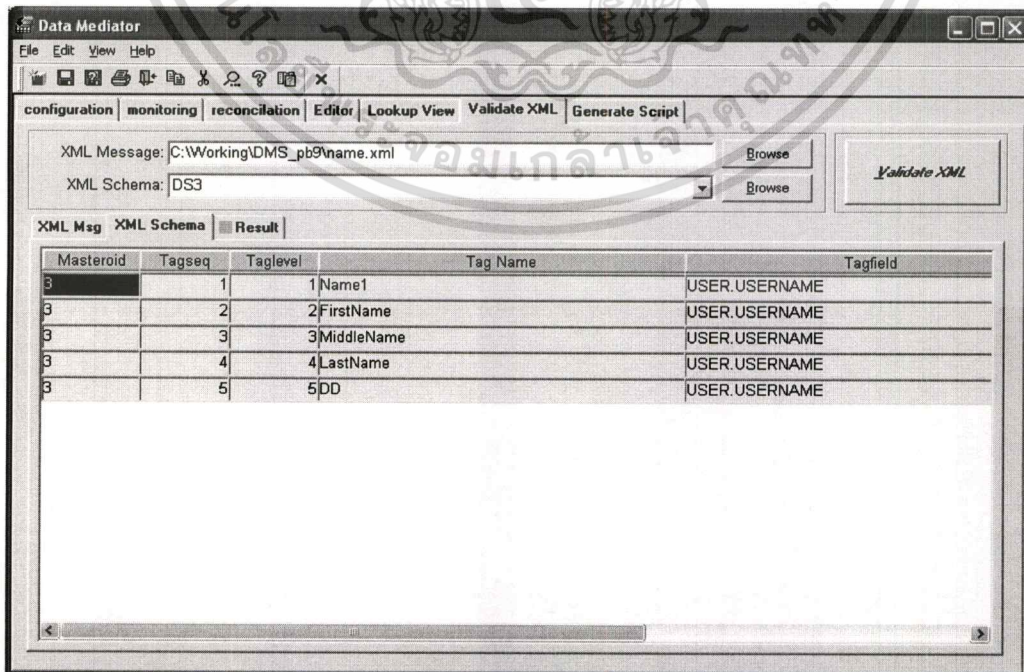
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 5.11 แสดงหน้าจอการค้นหาไฟล์ ภายใต้งานไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้เลือก ไฟล์แล้วจะปรากฏข้อมูลในไฟล์ลงใน tabpage XML Msg ดังรูปที่ 5.10 แต่ถ้าผู้ใช้ กดปุ่ม browse เพื่อค้นหา XML Schema จะปรากฏหน้าจอค้นหา Message schema ดังรูปที่ 5.12



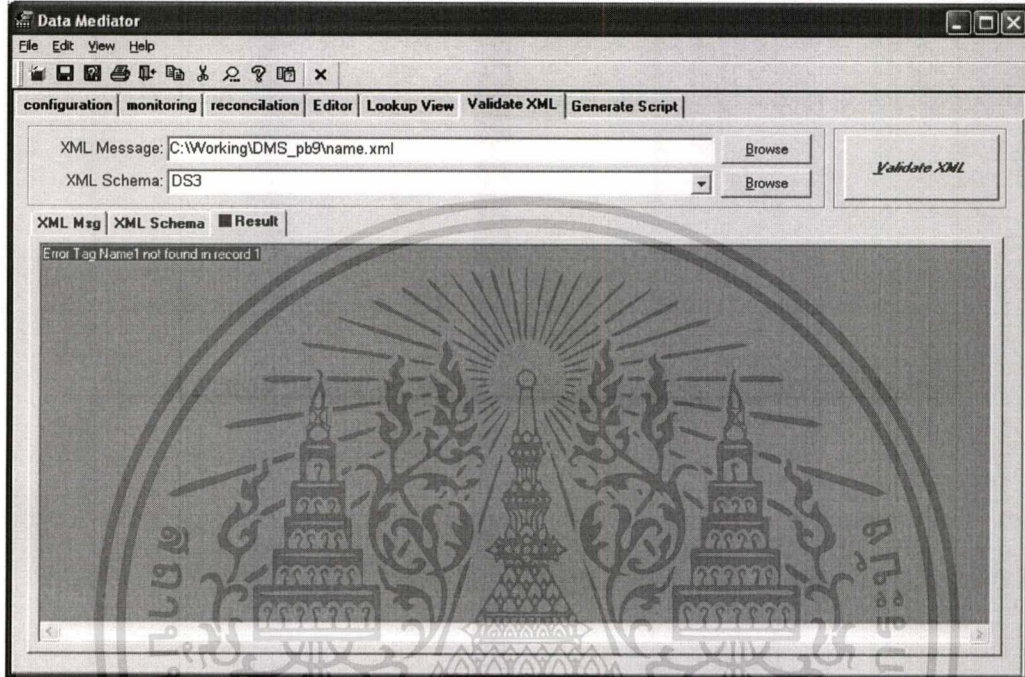
รูปที่ 5.12 แสดงหน้าจอการค้นหาmessage schema

เมื่อผู้ใช้ทำการเลือก message schema แล้วจะปรากฏรายละเอียดของ schema บน tabpage XML Schema ดังรูปที่ 5.13

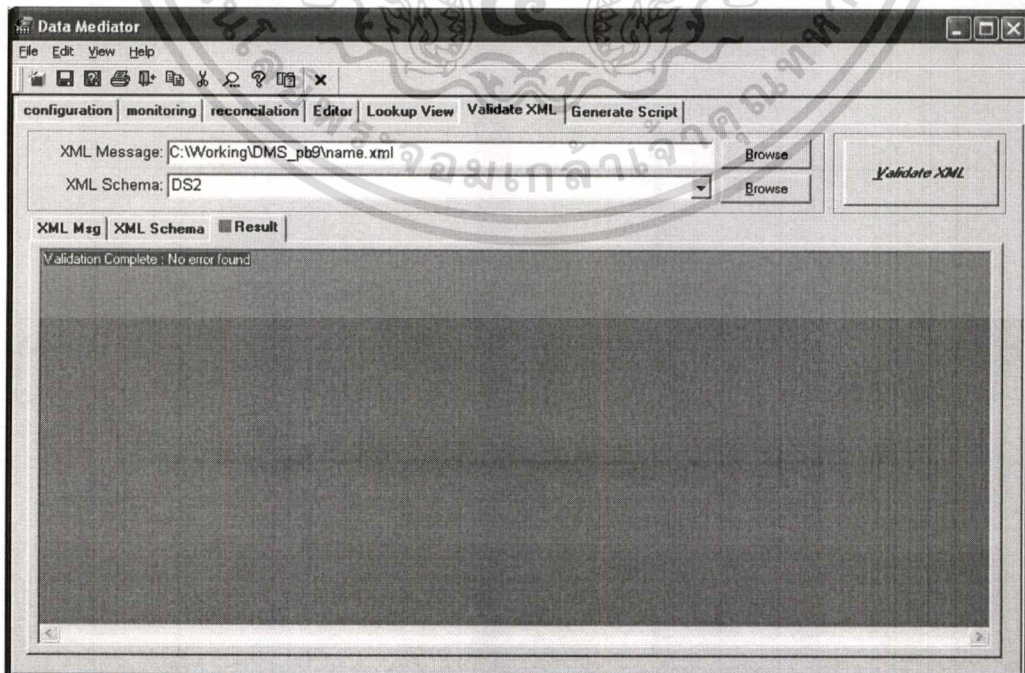


เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ 5.13 แสดงหน้าจอแสดงรายละเอียด XML Schema ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้กดปุ่ม Validate XML เพื่อตรวจสอบข้อมูลเอ็กซ์เอ็มแอลกับสคีม่าจะปรากฏไฟลีย์เขียวที่ tabpage result และไฟลีย์แดงที่ tabpage result ถ้าผลการตรวจสอบถูกต้องและผิดตามลำดับ ดังรูปที่ 5.14



รูปที่ 5.14 แสดงหน้าจอแสดงผลข้อผิดพลาดจากการ validate XML

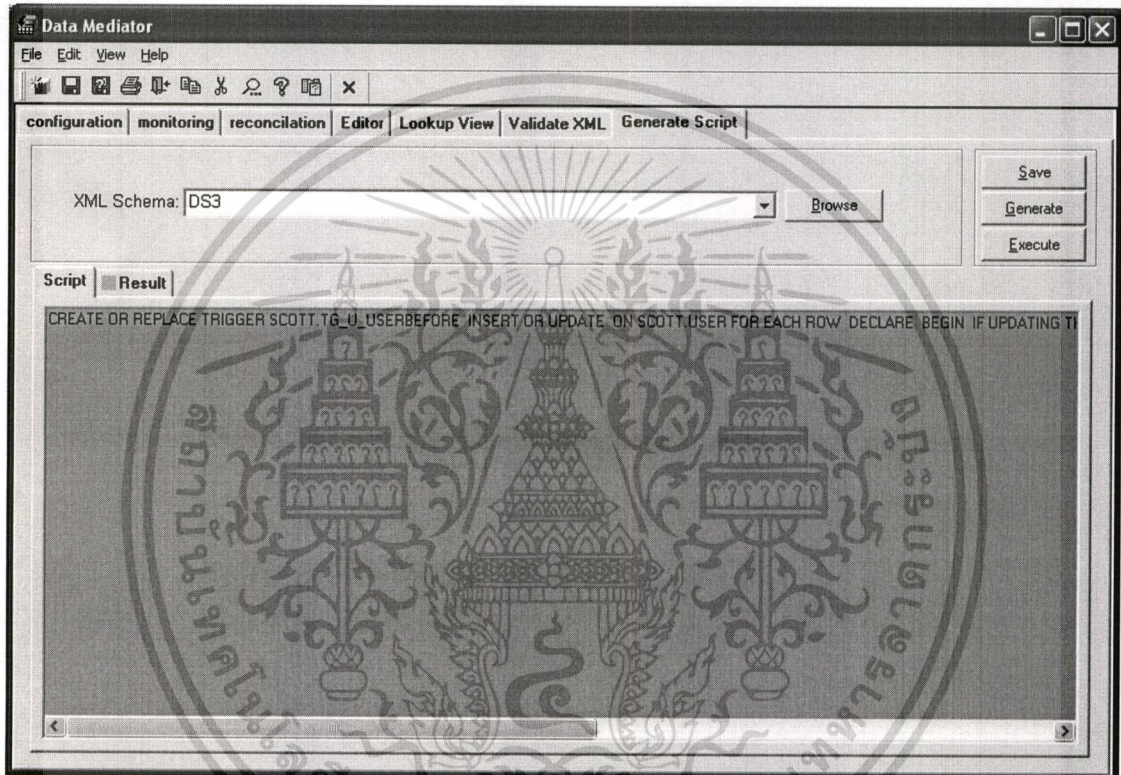


รูปที่ 5.15 แสดงหน้าจอแสดงผลแจ้งผลที่ถูกต้องจากการ validate XML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ tabpage Generate Script เป็นหน้าจอสำหรับเจนเนอเรตทริกเกอร์สคริปต์ เพื่อใช้เรียกเอ็กซ์เอ็มแอล Generator.class ให้ทำการสร้างไฟล์เอ็กซ์เอ็มแอลไปวางในคิวของ เอ็มคิวซีรีส์ ซึ่งเมื่อผู้ใช้เจนเนอเรตแล้วจะได้โค้ด และสามารถ รันโค้ดโดยการกดปุ่ม Execute และผลจากการรันจะปรากฏที่หน้า tabpage result ดังรูปที่ 5.15



รูปที่ 5.16 แสดงหน้าจอ Generate Script

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

### บทสรุป

#### 6.1 สรุปผลการออกแบบและพัฒนาระบบงาน

ในโครงการนี้ได้ดำเนินการศึกษาข้อมูล ทฤษฎี มาตรฐาน และเทคโนโลยีที่เกี่ยวข้องกับ IBM เอ็มคิวซีรีส์ และการพัฒนาโปรแกรมด้วยภาษา Power Builder9.0 , จาวา, เอ็กซ์เอ็มแอล , พร้อมกันนี้ได้ศึกษาเทคโนโลยี SQL/XML ของ Oracle9.0i มาใช้ในการเจเนอเรตข้อมูลให้อยู่ในรูปของเอกสารเอ็กซ์เอ็มแอล และศึกษาวิธีการใช้ Trigger ไปเรียกใช้ java store procedure โดยนำความรู้ที่ได้มาทำการออกแบบและพัฒนาระบบการสื่อสารข้อมูลข้ามแพลตฟอร์ม เพื่อให้ได้ระบบการสื่อสารข้อมูลข้ามแพลตฟอร์มที่ใช้งานง่าย สะดวก และมีความปลอดภัยของข้อมูล ซึ่งจะทำให้การเชื่อมต่อข้อมูลระหว่างระบบที่อยู่ต่างแพลตฟอร์มกันไม่ใช่เรื่องยุ่งยากอีกต่อไป

พร้อมกันนี้ ในช่วงการวิเคราะห์จนถึงการออกแบบระบบงาน ได้ทำการศึกษาทฤษฎีสำหรับการออกแบบและพัฒนาระบบเพิ่มเติม คือ Unified modeling Language (UML) จากนั้นจึงทำการออกแบบและพัฒนาระบบ ซึ่งในระหว่างการพัฒนาได้มีการทดสอบ ปรับปรุงโครงสร้างและฟังก์ชันการทำงานของระบบ จนสามารถใช้งานได้ตรงความต้องการ

#### 6.2 ประโยชน์ที่ได้รับจากการออกแบบและพัฒนาระบบ

ประโยชน์ที่ได้รับจากการพัฒนาระบบในโครงการนี้สามารถสรุปได้ดังนี้

1. ได้เรียนรู้การพัฒนาโปรแกรมด้วยภาษา Powerbuilder 9.0 , จาวา, และสามารถนำมาประยุกต์ในการเขียนโปรแกรมให้สำเร็จลุล่วงได้เป็นอย่างดี
2. ได้ศึกษา มาตรฐาน และ เทคโนโลยี ของ Middleware ของค่ายไอบีเอ็มเพื่อใช้ในการสื่อสารข้อมูลข้ามแพลตฟอร์ม
3. ได้ศึกษา เทคโนโลยี เอ็กซ์เอ็มแอล เพื่อใช้ในรูปแบบของเอกสารที่ใช้ในการเชื่อมโยงข้อมูลข้ามแพลตฟอร์ม
4. ได้ศึกษาเทคโนโลยี SQL/XML และ Java Store Procedure ซึ่งเป็นความสามารถของ Oracle เพื่อช่วยให้การเขียนโปรแกรมง่ายขึ้นและมีประสิทธิภาพมากขึ้นได้รับความรู้ และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ทักษะการออกแบบ และพัฒนาระบบงานด้วย UML โดยที่สามารถนำไปประยุกต์ใช้ในการออกแบบระบบงานอื่นๆ ได้
6. ได้ระบบการสื่อสารข้อมูลข้ามแพลตฟอร์มด้วย เอ็มคิวซีรีส์ ที่ใช้งานง่าย และ มีความยืดหยุ่น และไม่ต้องมีการแก้ไขโปรแกรมทุกครั้งที่เกิดการเปลี่ยนแปลงรูปแบบของ ชุดข้อมูล

### 6.3 ข้อจำกัดของระบบที่ออกแบบและพัฒนาขึ้น

1. ระบบที่ได้ยังไม่รองรับกับระบบการจัดการฐานข้อมูลค่ายอื่นๆ ยกเว้น Oracle
2. ระบบที่ได้ยังไม่สามารถตรวจจับดูความเคลื่อนไหวของชุดข้อมูลได้
3. ระบบที่ยังไม่สามารถตรวจสอบความถูกต้องไฟล์ เอ็กซ์เอ็มแอล กับ ไฟล์ เอ็กซ์เอสดี ได้ การตรวจสอบเอ็กซ์เอ็มแอลจะต้องตรวจสอบกับ ตารางสคิม่า ในระบบเท่านั้น

### 6.4 ปัญหาและอุปสรรคระหว่างการออกแบบและพัฒนาระบบงาน

เนื่องจากการพัฒนาระบบจะต้องใช้เทคโนโลยีที่หลากหลาย ทำให้ต้องใช้เวลาในการศึกษา เพื่อทำความเข้าใจและทดลองใช้ ประกอบกับฮาร์ดแวร์และระบบปฏิบัติการที่มีไม่อยู่ไม่สนับสนุน กับ ไอบีเอ็ม เอ็มคิวซีรีส์ ที่มีอยู่ จึงต้องเสียเวลาในการเปลี่ยนระบบปฏิบัติการและเตรียม สิ่งแวดล้อมใหม่

### 6.5 ข้อเสนอแนะและแนวทางในการพัฒนาระบบเพิ่มเติม

เพื่อให้ระบบมีประสิทธิภาพมากขึ้นควรจะมีการพัฒนาเพิ่มดังนี้

1. ให้โปรแกรมสามารถรองรับกับระบบการจัดการฐานข้อมูลค่ายอื่นๆ ได้ด้วย
2. ให้โปรแกรมสามารถตรวจจับดูความเคลื่อนไหวของชุดข้อมูลได้แบบเวลาจริง
3. ให้โปรแกรมสามารถตรวจสอบความถูกต้องของไฟล์เอ็กซ์เอ็มแอล (XML) กับไฟล์เอ็กซ์เอสดี (XSD) ได้ด้วย
4. ให้ระบบสามารถรับและส่งข้อมูลแบบเวลาจริงได้

## บรรณานุกรม

- กิตติ ภัคดีวัฒนกุล และ กิติพงษ์ กลมกล่อม. 2544. **UML วิเคราะห์และออกแบบระบบเชิงวัตถุ**  
กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.
- Craig Larman. 1998. **Applying UML And Patterns** , Prentics Hall : PTR Prntic-Hall,Inc.,
- Somnuk Keretho. **Object-Oriented Software Development with Unified Modeling Language**. Bangkok : Kasetsart University.
- Dennis, A. et al. 2002. **System Analysis & Design An Object-Oriented Approach with UML** :  
Wiley
- IBM Learning Services. 1999. **MQSeries Application Programming**. Bangkok : np
- IBM. n.p.d. **IBM MQSeries** [Online] Available: <http://www.ibm.com/MQSeries>
- Javasun. n.p.d. **Java forum** [Online] Available:<http://forum.java.sun.com/thread.jspa?threadID=568599&messageID=2809916>
- Pfeiffer , Ralph L. n.p.d. **XML Tutorial for Programmers** [online] Available:<http://review.software.ibm.com/>
- Oracle. n.p.d. **Oracle FAQ** [Online] Available: <http://www.orafaq.com>
- Oracle. n.p.d. **Oracle technology** [Online] Available:<http://www.oracle.com/technology/oramag/oracle/03-may/o33xml.html>
- Melody. 2003. **SDLC** [Online] Available:<http://www.melody.com>

## ประวัติผู้เขียน

ชื่อ- นามสกุล นางสาวจิตลดา รอดชัย  
วัน/เดือน/ปี เกิด 13 มิถุนายน 2521  
สถานที่เกิด จังหวัดชัยนาท

### ประวัติการศึกษา

มัธยมศึกษา  
ปริญญาตรี

โรงเรียนนครสวรรค์ จังหวัดนครสวรรค์  
มหาวิทยาลัยนเรศวร สาขาวิทยาการคอมพิวเตอร์

### ประวัติการทำงาน

พ.ศ.2543 - พ.ศ.2545

โปรแกรมเมอร์  
บริษัทไทยเทเลซอฟท์ จำกัด

พ.ศ.2545 - พ.ศ.2546

แอนาליสต์โปรแกรมเมอร์  
บริษัท เอเชิร์ท จำกัด

พ.ศ.2546-พ.ศ.2547

โปรแกรมเมอร์  
บริษัทกรู๊ปไทยคอมพิวเตอร์ เซอร์วิสเชส จำกัด

พ.ศ.2547 – ปัจจุบัน

แอนาליสต์ โปรแกรมเมอร์  
บริษัทดีเอสที อินเทอร์เน็ตเนชั่นแนล จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้