

ภาษาจำเพาะโดเมนสำหรับเขียนบทความส่งกระบวนการอีทีแอล



นางสาวสุนิสา จันทร์สว่าง

จุฬาลงกรณ์มหาวิทยาลัย

CHULALONGKORN UNIVERSITY

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย

The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the University Graduate School.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2559

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

A DOMAIN SPECIFIC LANGUAGE FOR SCRIPTING ETL PROCESS

Miss Sunisa Junsawang



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 2016

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอิทีแอล
โดย	นางสาวสุนิสา จันทร์สว่าง
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร. ญาใจ ลี้มปิยะกรณ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้รับวิทยานิพนธ์ฉบับนี้เป็นส่วน
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร. สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. สุกรี สินธุภิญโญ)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร. ญาใจ ลี้มปิยะกรณ์)

.....กรรมการภายนอกมหาวิทยาลัย
(อาจารย์ ดร. ภาสกร อภิรักษ์วรพิณิต)

5870974921 : MAJOR COMPUTER SCIENCE

KEYWORDS: DOMAIN SPECIFIC LANGUAGE, ETL, QUERY, DATA WAREHOUSE, SOFTWARE PROCESS IMPROVEMENT.

SUNISA JUNSAWANG: A DOMAIN SPECIFIC LANGUAGE FOR SCRIPTING ETL PROCESS. ADVISOR: ASSOC. PROF. YACHAI LIMPIYAKORN, Ph.D., 66 pp.

ETL (Extract-Transform-Load) is considered as general preprocessing for data preparation of heterogeneous sources such as data warehouse. Building the ETL process is complex and resource consuming. However, it is crucial as the integration layer in data warehouse environment and contributes to the accuracy and the correctness of data in a warehouse. ETL processes can be mainly categorized into 2 types: model-based and code-based. The model-based ETL is user-friendly but hardly scales up with large complex systems, whereas the code-based ETL requires programming skill. This research presents an approach of code-based ETL of which the commands are written in a domain specific language, rather than programming or query languages. The objective is to improve the productivity and usability of the ETL process. In addition to directly write the ETL script with the domain-specific language developed in this work, the input of ETL scripts written by non-technical users in Excel format is also allowed. The paradigm of code-to-code transformation is applied for transforming the textual, domain-specific language into Java code. The etlDSL code generator is implemented with MVC architecture design. A case study and the user evaluation of the proposed approach are also discussed. The overall rating is satisfaction.

Department: Computer Engineering Student's Signature

Field of Study: Computer Science Advisor's Signature

Academic Year: 2016

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความอนุเคราะห์อย่างยิ่งของรองศาสตราจารย์ ดร.ญาใจ ลิ้มปิยะกรณ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้สละเวลาให้ความรู้ ให้คำปรึกษา ตรวจสอบ ให้คำแนะนำแนวทางการวิจัย และสนับสนุนจนทำให้การวิจัยในครั้งนี้สำเร็จออกมา ด้วยดีข้าพเจ้าจึงขอกราบระลึกถึงพระคุณของรองศาสตราจารย์ ดร.ญาใจ ลิ้มปิยะกรณ ไว้ ณ ที่นี้

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ และ อาจารย์ ดร.ภาสกร อภิรักษ์วรพินิต กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลา ให้คำแนะนำ ตรวจสอบ และ แก้ไขวิทยานิพนธ์ฉบับนี้

ท้ายที่สุด ผู้เสนอวิทยานิพนธ์ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และครอบครัว สำหรับกำลังใจที่มีค่ายิ่ง รวมถึงขอขอบพระคุณผู้บังคับบัญชาในสายงาน เพื่อนร่วมงาน และมิตรสหาย ที่คอยติดตามให้กำลังใจ ให้การสนับสนุนและความช่วยเหลือในด้านต่างๆ และท่านอื่นๆ ที่มีได้กล่าวชื่อไว้ ณ ที่นี้ที่มีส่วนช่วยให้วิทยานิพนธ์ของข้าพเจ้าสำเร็จไปได้ด้วยดี

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ฅ
สารบัญภาพ.....	ญ
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย.....	2
1.3 ขอบเขตการวิจัย.....	2
1.4 ขั้นตอนการวิจัย.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์.....	3
1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์.....	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	4
2.1 ทฤษฎีที่เกี่ยวข้อง.....	4
2.2 งานวิจัยที่เกี่ยวข้อง.....	7
บทที่ 3 วิธีดำเนินการวิจัย.....	14
3.1 แนวคิดและวิธีวิจัย.....	14
บทที่ 4 การออกแบบและการพัฒนาระบบ.....	22
4.1 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา.....	22
4.2 การออกแบบสถาปัตยกรรม.....	22

4.3 ตัวอย่างส่วนแสดงผล และ จาวาที่ได้จากสคริปต์อีทีแอลดีเอสแอล.....	23
4.5 อีทีแอลหลังบ้าน (ETL Back-end).....	36
บทที่ 5 การประเมินและการวัดผล.....	45
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....	54
6.1 สรุปผลการวิจัย	54
6.2 ข้อจำกัด	54
6.3 แนวทางการวิจัยในอนาคต.....	54
รายการอ้างอิง.....	55
ภาคผนวก	56
ภาคผนวก ก.....	57
ภาคผนวก ข.....	59
ประวัติผู้เขียนวิทยานิพนธ์	66



สารบัญตาราง

ตารางที่ 1 การดำเนินการกับกลุ่มข้อมูลเดียวในรูปแบบพีชคณิตเชิงสัมพันธ์เทียบกับคำสั่งเอสคิวแอล.....	6
ตารางที่ 2 การดำเนินการกับ 2 กลุ่มข้อมูลในรูปแบบพีชคณิตเชิงสัมพันธ์เทียบกับคำสั่งเอสคิวแอล.....	6
ตารางที่ 3 คำอธิบายคำสำคัญวากยสัมพันธ์สำหรับสร้างไวยากรณ์ของดีเอสแอลในภาพที่ 19	18
ตารางที่ 4 ผลความพึงพอใจ.....	53
ตารางที่ 5 ชุดคำสั่งที่เขียนด้วยสคริปต์ดีเอสแอลอีทีแอล [9].....	64



สารบัญภาพ

ภาพที่ 1	สคริปต์ดีเอสแอลที่ทำหน้าที่เสมือนอินเตอร์เฟซสำหรับผู้ใช้งาน	4
ภาพที่ 2	ดีเอสแอลภายใน	4
ภาพที่ 3	ดีเอสแอลภายนอก	5
ภาพที่ 4	ตัวอย่างกระบวนการแปลงดีเอสแอลเป็นภาษาจาวา [4]	7
ภาพที่ 5	สถาปัตยกรรมระบบ [5]	8
ภาพที่ 6	Query Optimizer [6]	9
ภาพที่ 7	ตัวอย่างเอสคิวแอลคั่นถามหนังสือที่สามารถทำการจองได้ [6]	9
ภาพที่ 8	นิพจน์พีชคณิตเชิงสัมพันธ์ของเอสคิวแอลในภาพที่ 7 [6]	9
ภาพที่ 9	ต้นไม้คั่นถามที่สร้างจากนิพจน์พีชคณิตเชิงสัมพันธ์ในภาพที่ 8 [6]	9
ภาพที่ 10	(a) ข้อมูลที่รวมกัน; (b) ข้อมูลที่ไม่ตรงกัน	10
ภาพที่ 11	การส่งข้อมูลที่หมดอายุจากตารางมิติไปยังตารางที่เก็บข้อมูลในอดีต	11
ภาพที่ 12	ปรับปรุงมิติด้วยการแปลงข้อมูล	11
ภาพที่ 13	พีชคณิตเชิงสัมพันธ์ต้นไม้สำหรับลักษณะข้อมูลใหม่	12
ภาพที่ 14	โครงสร้างของการแปลงแบบจำลองเว็บดีเอสแอล	13
ภาพที่ 15	ดีเอสแอลเมทาโมเดล (DSL Meta-model)	14
ภาพที่ 16	กระบวนการอีทีแอลจากอีทีแอลดีเอสแอล	15
ภาพที่ 17	การแปลงดีเอสแอลให้กลายเป็นโค้ดอีทีแอล	15
ภาพที่ 18	ขั้นตอนการอ่านไวยากรณ์ดีเอสแอล	15
ภาพที่ 19	วากยสัมพันธ์สำหรับสร้างไวยากรณ์ของดีเอสแอล	17
ภาพที่ 20	ตัวอย่างกฎการแปลงเป็นเอสคิวแอล	19
ภาพที่ 21	การออกแบบสถาปัตยกรรมระบบด้วยเอ็มวีซี	23
ภาพที่ 22	ตัวอย่างการลบข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอล	24

ภาพที่ 23 ตัวอย่างจาวาที่ได้จากการลบข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอลในภาพที่ 22	25
ภาพที่ 24 ตัวอย่าง Ouput log ใน view component จากภาพที่ 23	25
ภาพที่ 25 ตัวอย่างการนำเข้าข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอล	26
ภาพที่ 26 ตัวอย่างจาวาที่ได้จากการนำเข้าข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอล	28
ภาพที่ 27 ตัวอย่าง Ouput log ใน view component จากภาพที่ 26	29
ภาพที่ 28 ตัวอย่างสคริปต์อีทีแอลดีเอสแอลเพื่อการแก้ไขข้อมูลในตารางโดยการปรับให้เป็นปัจจุบัน	30
ภาพที่ 29 ตัวอย่างจาวาที่ได้จากสคริปต์อีทีแอลดีเอสแอลเพื่อการแก้ไขข้อมูลในตารางโดยการปรับให้เป็นปัจจุบันในภาพที่ 28	32
ภาพที่ 30 ตัวอย่าง Ouput log ใน view component จากภาพที่ 29	32
ภาพที่ 31 ตัวอย่างสคริปต์อีทีแอลดีเอสแอลเพื่อแก้ไขข้อมูลในโดยการปรับให้อยู่ในรูปแบบปัจจุบันและนำเข้าข้อมูลในส่วนที่เดิมไม่เคยปรากฏในตาราง	33
ภาพที่ 32 ตัวอย่างจาวาที่ได้จากสคริปต์อีทีแอลดีเอสแอลเพื่อแก้ไขข้อมูลในโดยการปรับให้อยู่ในรูปแบบปัจจุบันและนำเข้าข้อมูลในส่วนที่เดิมไม่เคยปรากฏในตารางในภาพที่ 31	36
ภาพที่ 33 ตัวอย่าง Ouput log ใน view component จากภาพที่ 32	36
ภาพที่ 34 แผนภาพยูเอ็มแอลอีทีแอลหลังบ้าน	37
ภาพที่ 35 ตัวอย่าง Method ตัวควบคุมอีทีแอล	38
ภาพที่ 36 ตัวอย่าง Method ตัวควบคุมไฟล์ของคลาสไฟล์โอเปอเรชัน	39
ภาพที่ 37 ตัวอย่าง Method ตัวควบคุมไฟล์ในคลาสอ่านพารามิเตอร์	39
ภาพที่ 38 ตัวอย่าง Method ตัวควบคุมการแปลง	40
ภาพที่ 39 ตัวอย่าง Method ตัวควบคุมการดำเนินการฐานข้อมูล	43
ภาพที่ 40 ตัวอย่าง Method ของจาวาโมเดลสำหรับการเชื่อมต่อกับฐานข้อมูล	44
ภาพที่ 41 ตัวอย่าง Method ของจาวาโมเดลสำหรับไฟล์นำเข้าข้อมูล	44
ภาพที่ 42 ตัวอย่าง Method ของจาวาโมเดลสำหรับส่วนของการแสดงผล	44

ภาพที่ 43 ตัวอย่างรูปแบบของไฟล์เอกสาร Excel	46
ภาพที่ 44 สคริปต์ดีเอสแอลอีทีแอลจาก ภาพที่43	48
ภาพที่ 45 ตารางข้อมูลSTORE_DIM	48
ภาพที่ 46 ตารางข้อมูลSALES_HEADER_FCT	48
ภาพที่ 47 ตารางข้อมูลSALE_ATTRIBUTE	49
ภาพที่ 48 ตารางข้อมูลSTG_LOG.....	49
ภาพที่ 49 ตารางข้อมูลCAMPAIGN_FACT-1	49
ภาพที่ 50 ตารางข้อมูลCAMPAIGN_FACT-2.....	49
ภาพที่ 51 ผลลัพธ์ของตารางข้อมูลCAMPAIGN_FACT-1	49
ภาพที่ 52 ผลลัพธ์ของตารางข้อมูลCAMPAIGN_FACT-2	50
ภาพที่ 53 สคริปต์ดีเอสแอลอีทีแอลในเคสการเปลี่ยนข้อมูลในฐานข้อมูล	50
ภาพที่ 54 ตารางข้อมูลCAMPAIGN_FACT	51
ภาพที่ 55 ผลลัพธ์ของตารางข้อมูลCAMPAIGN_FACT จากสคริปต์ดีเอสแอลอีทีแอลในเคสการ เปลี่ยนข้อมูลในฐานข้อมูล	51
ภาพที่ 56 สคริปต์ดีเอสแอลอีทีแอลในเคสการนำข้อมูลเข้าฐานข้อมูลจากไฟล์	52
ภาพที่ 57 ไฟล์ testLoad.txt สำหรับการนำข้อมูลเข้าในตารางข้อมูล	52
ภาพที่ 58 ตารางข้อมูลTABLEA.....	52
ภาพที่ 59 การสร้างไฟล์นามสกุล .etl สำหรับการเขียนสคริปต์อีทีแอลดีเอสแอล	59
ภาพที่ 60 สร้างชื่อไฟล์ของสคริปต์ดีเอสแอลอีทีแอล.....	60
ภาพที่ 61 ตัวอย่างไฟล์สคริปต์ดีเอสแอลอีทีแอล	60
ภาพที่ 62 ตัวอย่างของโปรแกรมสำหรับเขียนสคริปต์ดีเอสแอลอีทีแอล.....	61
ภาพที่ 63 ตัวอย่างการเขียนสคริปต์ดีเอสแอลอีทีแอล.....	61
ภาพที่ 64 ตัวอย่างไฟล์จาวาที่ได้จากสคริปต์ดีเอสแอลอีทีแอล.....	62
ภาพที่ 65 ตัวอย่าง log การประมวลผลของสคริปต์ดีเอสแอลอีทีแอล	62

ภาพที่ 66 ตัวอย่าง excel สำหรับการเขียนสคริปต์ดีเอสแอลอีทีแอล.....63



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

อีทีแอล (ETL— Extract, Transform, Load) คือ กระบวนการดึงข้อมูลจากแหล่งข้อมูลหลายแหล่งส่งไปยังกระบวนการแปลงข้อมูล และโหลดข้อมูลลงในคลังข้อมูล (Data warehouse) เพื่อบูรณาการข้อมูลดิบจากหลายแหล่งกำเนิด ให้กลายเป็นสารสนเทศ (Information) หรือข้อมูลที่มีความหมาย สามารถนำไปใช้ประโยชน์ได้

อีทีแอล เป็นเครื่องมือที่มีลักษณะของการสั่งการ ซึ่งจำแนกได้เป็น 2 ประเภท คือ บนพื้นฐานของแบบจำลอง (model-based) และบนพื้นฐานของภาษา (code-based) โดยทั่วไป ประเภท model-based จะมีการสั่งการผ่านการลาก-วาง ส่วน code-based จะมีการสั่งการผ่านการเขียนคำสั่งด้วยภาษาคอมพิวเตอร์ ทั้งนี้ พบว่าการใช้เครื่องมืออีทีแอลแบบ model-based ในบางครั้ง อาจจะเกิดความผิดพลาดในการสร้างแบบจำลองที่ผิดเพี้ยนไปจากที่ออกแบบไว้ เช่น กรณีความสัมพันธ์ระหว่างตารางที่มีหลายตารางและเงื่อนไขการแปลงข้อมูลที่มีความซับซ้อนเป็นอย่างมาก ทำให้การลากวางส่วนประกอบในแผนภาพมีการทับซ้อน มองเห็นหรือเข้าใจได้ยาก นอกจากนี้ ยังพบว่ารูปแบบการเขียนเอสคิวแอล (SQL) มีความหลากหลายและขาดความเป็นมาตรฐาน เช่น ผู้พัฒนาบางรายจะเขียนด้วยอักขรตัวใหญ่ทั้งหมดหรือเขียนด้วยตัวเล็กทั้งหมด, ใช้คำสั่งระบุชื่อคอลัมน์หรือใช้เครื่องหมายดาว (*) แทน และบ่อยครั้งพบว่าเมื่อมีการเพิ่มคอลัมน์หรือแก้ไขคอลัมน์ในโครงสร้างตารางจะยากแก่การค้นหาถึงจุดที่ควรแก้ไข

งานวิจัยนี้ได้นำเสนอแนวทางการพัฒนาภาษาจำเพาะโดเมนหรือดีเอสแอล (Domain Specific Language: DSL) เพื่อใช้เป็นเครื่องมือ โค้ดเบส-อีทีแอล สำหรับการเตรียมข้อมูลนำเข้าคลังข้อมูล ข้อดีของการสั่งการด้วยดีเอสแอล คือ เป็นรูปแบบที่เขียน/อ่านเข้าใจง่าย ถึงแม้บุคคลนั้นไม่มีพื้นฐานการเขียนโปรแกรมมาก่อน ช่วยลดการเขียนข้อความอธิบายในส่วนของโปรแกรม ลดความซับซ้อนของการเขียนโปรแกรมในลักษณะอ็อบเจกต์ รวมทั้งสามารถลดความผิดพลาดที่อาจเกิดขึ้นจากการเขียนเอสคิวแอลได้หลายรูปแบบที่แตกต่างกัน โดยการแทนด้วยพีชคณิตเชิงสัมพันธ์ (Relational Algebra) ที่นำเสนอในงานวิจัยนี้ วิธีการที่นำเสนอจะช่วยลดปัญหาที่เกิดจากความผิดพลาดของมนุษย์ และลดปัญหาความเข้าใจผิดที่อาจเกิดจากการเปลี่ยนผู้รับผิดชอบโปรแกรมในส่วนนั้น ซึ่งท้ายสุดจะส่งผลให้ข้อมูลผลลัพธ์ของกระบวนการอีทีแอล มีความถูกต้องสมบูรณ์

1.2 วัตถุประสงค์ของงานวิจัย

พัฒนาไวยากรณ์ดีเอสแอลสำหรับการประยุกต์ใช้กับกระบวนการอีทีแอล เพื่อลดปัญหาความผิดพลาดจากความซับซ้อนของเงื่อนไขหรือกระบวนการ และลดความผิดพลาดที่อาจเกิดขึ้นจากความผิดพลาดของผู้พัฒนาหรือผู้รับช่วงต่อจากผู้พัฒนาเดิม

1.3 ขอบเขตการวิจัย

1. ไวยากรณ์ดีเอสแอลถูกสร้างภายใต้ภาษาจาวาและเครื่องมือที่ใช้ในการพัฒนาคือ Xtext
2. การแปลงดีเอสแอลเป็นเอสคิวแอลนั้นจะต้องสามารถอธิบายได้ด้วยพีชคณิตเชิงสัมพันธ์
3. ประเมินผลงานวิจัยโดยเปรียบเทียบผลลัพธ์ที่ได้จากเครื่องมืออีทีแอลที่พัฒนาขึ้นกับผลลัพธ์ที่คาดหวัง

1.4 ขั้นตอนการวิจัย

1. ศึกษาและทำความเข้าใจทฤษฎีและงานวิจัยที่เกี่ยวข้อง
2. ศึกษาและเลือกเครื่องมือสำหรับสร้างไวยากรณ์จากดีเอสแอล
3. ออกแบบข้อกำหนดลักษณะของดีเอสแอลเพื่อสร้างเครื่องมืออีทีแอลบนพื้นฐานภาษาจาวา
4. พัฒนาระบบ
5. ทดสอบความถูกต้องของดีเอสแอลในการสั่งการอีทีแอลเปรียบเทียบกับผลลัพธ์ข้อมูลที่คาดหวัง
6. วิเคราะห์และประเมินผลการวิจัย
7. สรุปผลการวิจัยและข้อเสนอแนะ
8. ตีพิมพ์ผลงานทางวิชาการ
9. จัดทำวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้แนวทางการสร้างไวยากรณ์ของภาษาดีเอสแอลเพื่อพัฒนาเครื่องมืออีทีแอล
2. ได้เครื่องมืออีทีแอลบนพื้นฐานภาษาดีเอสแอล
3. ลดปัญหาความคลาดเคลื่อนจากความซับซ้อนของเงื่อนไขและกระบวนการอีทีแอล
4. ลดปัญหาที่อาจเกิดขึ้นจากผู้พัฒนาหรือผู้รับช่วงต่อ

1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์

วิทยานิพนธ์นี้มีทั้งหมด 6 บท ดังต่อไปนี้ บทที่ 1 บทนำความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของการวิจัย ขอบเขตของการวิจัย ประโยชน์ที่คาดว่าจะได้รับและผลงานตีพิมพ์ บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง บทที่ 3 วิธีดำเนินการวิจัย บทที่ 4 การออกแบบและพัฒนาระบบตามแนวทางการวิจัยที่นำเสนอ บทที่ 5 วิธีประเมินและวัดผลการทดลองและบทที่ 6 สรุปผลการวิจัย ข้อเสนอแนะและแนวทางสำหรับอนาคต

1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์ได้รับการตีพิมพ์บทความวิชาการ 2 บทความ ประกอบด้วย

- 1) S. Junsawang and Y. Limpiyakorn, “A Domain Specific Language for Scripting ETL Process” ในรายงานการประชุมวิชาการนานาชาติสืบเนื่องจาก The 9th International Conference on Future Computer and Communication (ICFCC 2017), April 21-23, 2017, Nagoya, Japan.
- 2) S. Junsawang and Y. Limpiyakorn, “Case Study of ETL Java Code Generation from Domain Specific Language” ในรายงานการประชุมวิชาการนานาชาติสืบเนื่องจาก The 7th International Workshop on Computer Science and Engineering (WCSE 2017), June 25-27, 2017, Beijing, China.

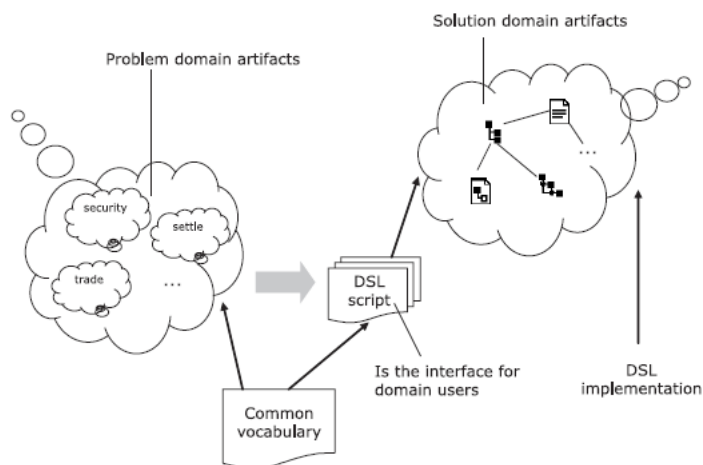
บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ภาษาจำเพาะโดเมนหรือดีเอสแอล

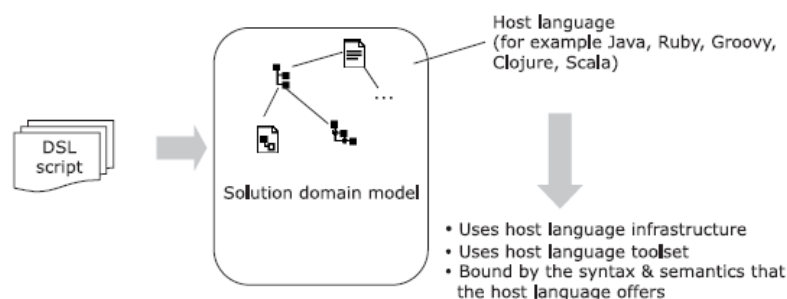
ดีเอสแอล [1] คือ ภาษาที่กำหนดให้อยู่ในรูปแบบที่ง่ายต่อการใช้งานและถูกออกแบบโดยเฉพาะสำหรับโดเมนปัญหาหนึ่งๆ โดยทั่วไป การสร้างคำสั่งในสคริปต์ดีเอสแอลจะประกอบด้วยคำศัพท์ที่ส่วนมากใช้กันในโดเมนปัญหานั้น ทำให้ภาษาดูเป็นธรรมชาติมากขึ้นง่ายต่อการใช้งาน ดังภาพที่ 1



ภาพที่ 1 สคริปต์ดีเอสแอลที่ทำหน้าที่เสมือนอินเตอร์เฟซสำหรับผู้ใช้งาน

ดีเอสแอล แบ่งเป็น 2 ประเภท ดังนี้

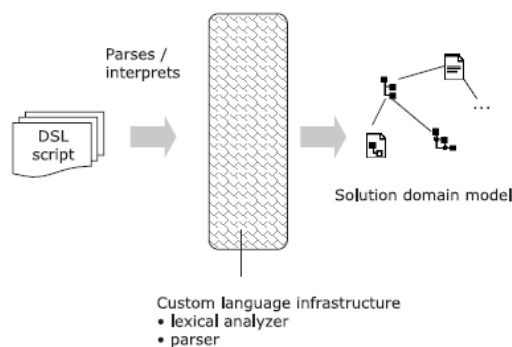
1. ดีเอสแอลภายใน (Internal/ Embedded DSL) เป็นการนำโครงสร้างพื้นฐานของภาษาที่มีอยู่เพื่อสร้างความหมายเฉพาะโดเมน ดีเอสแอลภายในจะถูกเขียนโดยใช้ภาษาโฮสต์และโครงสร้างพื้นฐานที่มีอยู่



ภาพที่ 2 ดีเอสแอลภายใน

2. ดีเอสแอลภายนอก (External DSL) เป็นการพัฒนาวากยสัมพันธ์ (syntax) และความหมาย (semantic) ของภาษาใหม่ โดยภาษาที่ถูกพัฒนาขึ้นนั้นจะไม่มีอ้างอิงถึงภาษาใดในปัจจุบันทั้ง

รูปแบบการพัฒนาและลักษณะการทำงาน ภาพที่ 3 แสดงให้เห็นถึงการพัฒนาภาษาดีเอสแอลภายนอกที่ต้องพัฒนากระบวนการแปลงภาษาขึ้นเพิ่มเติม สำหรับกระบวนการดังกล่าวจะประกอบด้วยการตัดคำ และการสร้างโค้ดที่มักพบเจอได้ในภาษาระดับสูง (high-level language)



ภาพที่ 3 ดีเอสแอลภายนอก

2.1.2 อีทีแอล

อีทีแอล [2] คือ กระบวนการดึงข้อมูลจากหลายแหล่งข้อมูลมารวบรวมลงในคลังข้อมูล ประกอบด้วยขั้นตอนต่อไปนี้

การดึง (Extraction): กระทำการระบุแหล่งข้อมูลต้นทางที่ตรงกับความต้องการปลายทาง และดึงมาประมวลผลในขั้นตอนต่อไป

การแปลง (Transformation): คือ กิจกรรมปรับเปลี่ยนแปลงข้อมูล ได้แก่

- การประยุกต์กฎใหม่ทางธุรกิจ เช่น การคำนวณ dimension และ measures
- การทำความสะอาดข้อมูล (Cleaning data) เช่น null เท่ากับ 0 หรือ Male เท่ากับ M และ Female เท่ากับ F
- การกรองข้อมูล (Filtering data) เช่น การเลือกเฉพาะข้อมูลที่ต้องการจัดเก็บ
- การแยกคอลัมน์ (Splitting a column)
- การผสานข้อมูล (Joining) เช่น lookup และ merge ข้อมูลที่มาจาก sources เดียวกัน
- แปลงข้อมูลแถว (Transposing rows) เปลี่ยนข้อมูลจากรูปแบบ row เป็น columns

การโหลด (Loading): การนำข้อมูลเข้าใน Data Mart หรือ Data Warehouse เป็นต้น

2.1.3 พีชคณิตเชิงสัมพันธ์ และ เอสคิวแอล

กระบวนการทำงานของเอสคิวแอลได้ถูกออกแบบภายใต้นิยามของพีชคณิตเชิงสัมพันธ์ [3] ซึ่งอธิบายด้วยทฤษฎีเรื่องของเซต (Set) และการดำเนินการที่เกี่ยวข้องกับเซตซึ่งจำแนกได้ 2 ประเภท คือ 1) การดำเนินการกับกลุ่มข้อมูลเดียว ได้แก่ การดึงข้อมูล, การเพิ่มข้อมูล, การลบข้อมูล, การ

แก้ไขข้อมูลเดิม และ 2) การดำเนินการกับ 2 กลุ่มข้อมูลขึ้นไป ได้แก่ การรวมกัน, การหาตัวแตกต่าง, การหาตัวซ้ำ เป็นต้น

การเทียบตัวดำเนินการในรูปแบบพีชคณิตเชิงสัมพันธ์กับคำสั่งเอสคิวแอลแสดงดังตาราง1และ2
ตารางที่ 1 การดำเนินการกับกลุ่มข้อมูลเดียวในรูปแบบพีชคณิตเชิงสัมพันธ์เทียบกับคำสั่งเอสคิวแอล

Relational Algebra	SQL
$\pi_{a, b}$	SELECT a, b
$\sigma_{(d > e) \wedge (f = g)}$	WHERE d > e AND f = g
$p \times q$	FROM p, q
$\pi_{a, b} \sigma_{(d > e) \wedge (f = g)} (p \times q)$	SELECT a, b FROM p, q WHERE d > e AND f = g; {must always have SELECT even if all attributes are kept, can be written as: SELECT *}
renaming	AS {or blank space}
$p := \text{result}$	INSERT INTO p result {assuming p was empty}
$\pi_{a, b} (p)$ (assume a, b are the only attributes)	SELECT * FROM p;

ตารางที่ 2 การดำเนินการกับ 2 กลุ่มข้อมูลในรูปแบบพีชคณิตเชิงสัมพันธ์เทียบกับคำสั่งเอสคิวแอล

Relational Algebra	SQL
$p \cup q$	SELECT * FROM p UNION SELECT * FROM q
$p - q$	SELECT * FROM p EXCEPT SELECT * FROM q Sometimes, instead, we have DELETE FROM
$p \cap q$	SELECT * FROM p INTERSECT SELECT * FROM q

สามารถอธิบายพีชคณิตเชิงสัมพันธ์ในรูปแบบของเซตและการดำเนินการเซตของกลุ่มข้อมูลได้ดังนี้

กำหนดให้ A, B และ C คือ เซตของกลุ่มข้อมูล

ยูเนียน (Union) : $A \cup B = \{x | x \in A \vee x \in B\}$

อินเตอร์เซกชัน (Intersection) $A \cap B = \{x | x \in A \wedge x \in B\}$

ความต่าง (Difference) $A - B = \{x | x \in A \wedge x \notin B\}$

ผลคูณคาร์ทีเซียน (Cartesian product) $A \times B = \{(x, y) | x \in A \wedge y \in B\}$

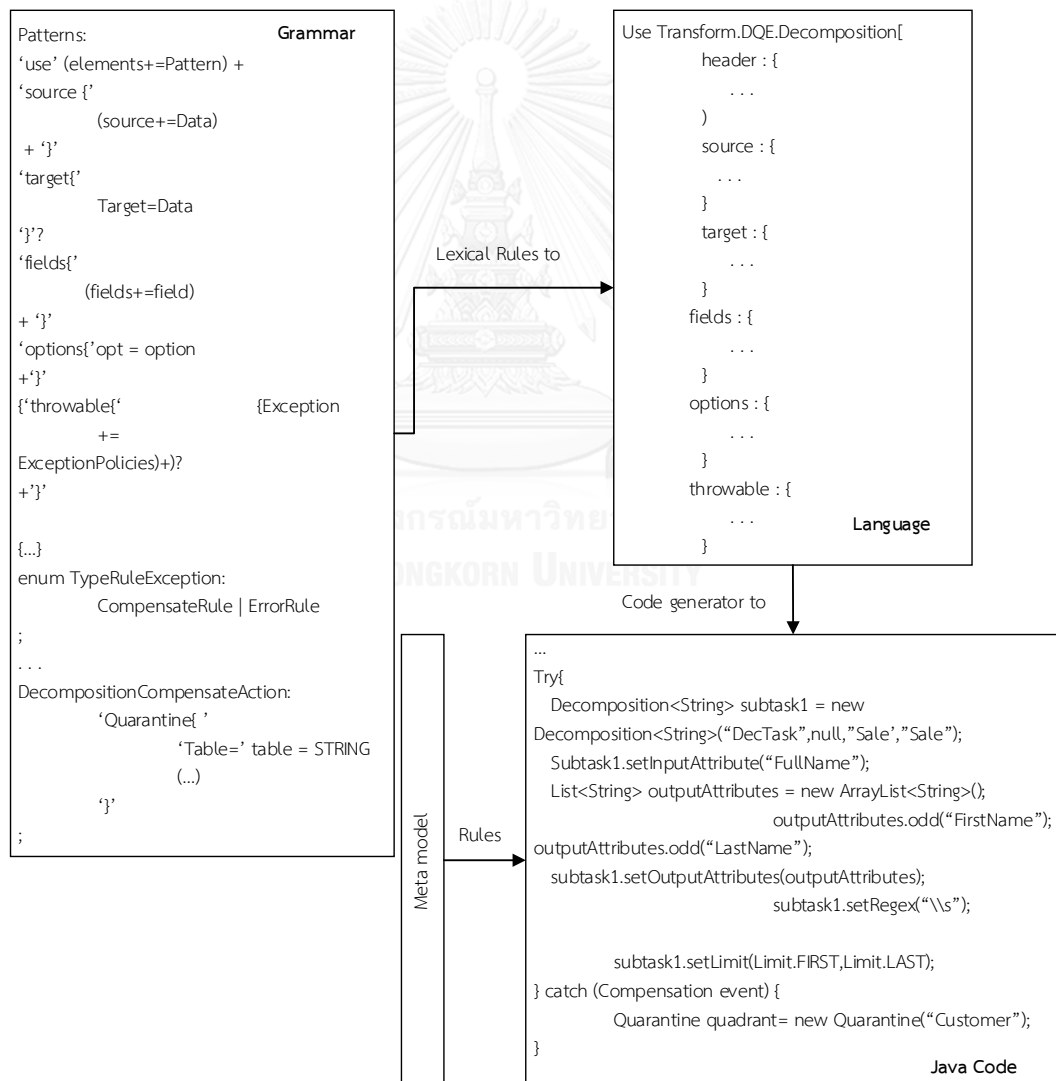
$A \times B \times C = \{(x, y, z) | x \in A \wedge y \in B \wedge z \in C\}$

จากข้างต้นสามารถแปลงเป็นรูปแบบของผลลัพธ์ได้เช่น $(A \cup B) \times (C \times A)$

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 On the specification of extract, transform, and load patterns behavior: A domain-specific language approach [4]

Oliveira และ Belo [4] นำเสนอแนวทางการใช้สัญกรณ์การจำลองแบบกระบวนการทางธุรกิจ (Business Process Modeling Notation: BPMN) เพื่อสร้างแบบจำลองขั้นตอนการทำงานของอีทีแอล โดยกำหนดรูปแบบไวยากรณ์ดีเอสแอลให้สามารถแปลงสร้างเป็นภาษาจาวา (ภาพที่ 4) ด้วยเครื่องมือแปลงภาษาชื่อว่า Xtext ซึ่งจะช่วยให้การแปลงสามารถเปลี่ยนแปลงได้อย่างอัตโนมัติทุกครั้งที่มีการแก้ไขภาษาดีเอสแอล อย่างไรก็ตาม ไวยากรณ์ดีเอสแอลที่นำเสนอในงานวิจัยนี้ ยังคงมีความยากในการทำความเข้าใจและเรียนรู้รูปแบบการใช้งาน

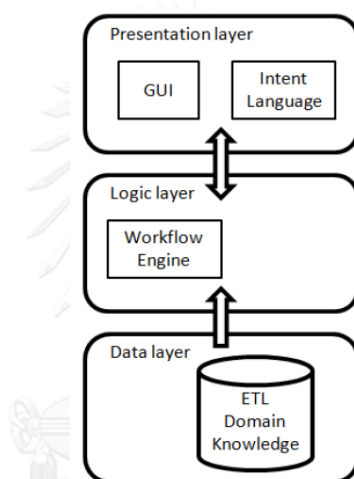


ภาพที่ 4 ตัวอย่างกระบวนการแปลงดีเอสแอลเป็นภาษาจาวา [4]

2.2.2 Domain Specific Model for Generating ETL Workflows from Business Intents [5]

Deneke [5] ได้พัฒนาดีเอสแอลด้วยภาษา C# สำหรับสร้างกระแสนงาน (workflow) อีทีแอล จากเจตนาทางธุรกิจ สถาปัตยกรรมระบบที่นำเสนอ (ภาพที่ 5) แยกเป็น 3 ชั้น ประกอบด้วย 1) ชั้นแสดงผล ส่วนนี้มีหน้าแสดงผลที่มี GUI และภาษาเจตนา (Intent Language) 2) ชั้นตรรกะเป็นส่วนที่ประมวลผลการทำงานของกระแสนงาน และ 3) ชั้นข้อมูล มีกระบวนการประมวลผลของอีทีแอล

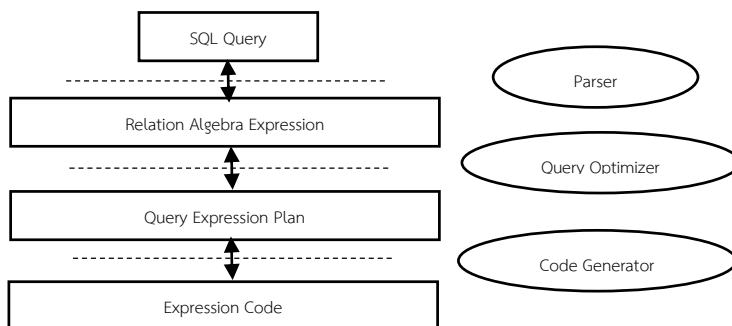
การเข้าถึงข้อมูลใช้โออาร์เอ็ม (ORM—object-relational mapper) ซึ่งเป็นกรอบงาน (framework) ที่ช่วยควบคุมการดึงข้อมูล แทนการใช้ข้อกำหนดการแปลงเอสคิวแอลเป็นอีทีแอล ส่วนหน้าแสดงผลจะแสดงผ่านดับเบิลยูพีเอฟ (WPF— Windows Presentation Foundation)



ภาพที่ 5 สถาปัตยกรรมระบบ [5]

2.2.3 A Study of Library Databases by Translating Those SQL Queries into Relational Algebra and Generating Query Trees [6]

Lasya และ Tanuku [6] ใช้กรณีศึกษาห้องสมุดเพื่อนำเสนอการแปลเอสคิวแอลเป็นนิพจน์พีชคณิตเชิงสัมพันธ์ (Relational Algebra Expression) และสร้างเป็นต้นไม้ค้นหา (Query tree) ของนิพจน์พีชคณิตเชิงสัมพันธ์ ภาพที่ 6 แสดงขั้นตอนวิธีการที่นำเสนอ เริ่มต้นจากการแจงส่วน (parse) เอสคิวแอลค้นหา ให้เป็นนิพจน์พีชคณิตเชิงสัมพันธ์ ซึ่ง query optimizer จะทำการแยกเป็นหน่วยย่อย เรียกว่า blocks สารสนเทศที่สำคัญจำเป็นในแต่ละ block จะถูกสกัดเพื่อสร้างเป็นแผนการประมวลผลการค้นหา (Query Execution Plan) แผนที่ให้ค่าประมาณต้นทุนต่ำสุดจะถูกนำมาสร้างเป็นต้นไม้ค้นหา เพื่อใช้ในการสร้าง Executable Code ต่อไป ภาพที่ 7 แสดงตัวอย่างเอสคิวแอลค้นหา ซึ่งถูกแปลเป็นนิพจน์พีชคณิตเชิงสัมพันธ์ ดังภาพที่ 8 และได้ผลลัพธ์ต้นไม้ค้นหาที่ให้ค่าประมาณต้นทุนต่ำสุด ดังภาพที่ 9



ภาพที่ 6 Query Optimizer [6]

```

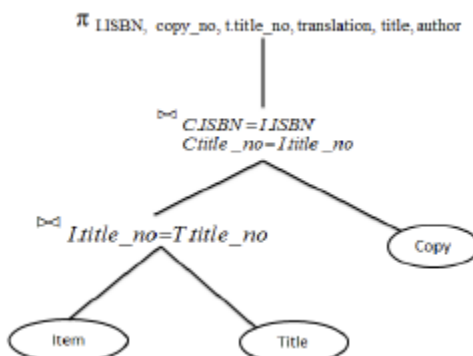
SELECT i.ISBN, copy_no, t.title_no,
translation, title, author
FROM copy As c
INNER JOIN item As i
ON i.isbn = c.isbn
AND i.title_no = c.title_no
INNER JOIN title As t
ON t.title_no = i.title_no;

```

ภาพที่ 7 ตัวอย่างเอสคิวแอลค้นถามหนังสือที่สามารถทำการจองได้ [6]

$$\begin{aligned}
 &\pi \text{ ISBN, copy_no, t.title_no, translation, title, author} \\
 &((\text{Copy} \bowtie_{\substack{C.ISBN=I.ISBN \\ C.title_no=I.title_no}} \text{Item}) \\
 &\quad \bowtie_{I.title_no=T.title_no} \text{Title})
 \end{aligned}$$

ภาพที่ 8 นิพจน์พีชคณิตเชิงสัมพันธ์ของเอสคิวแอลในภาพที่ 7 [6]



ภาพที่ 9 ต้นไม้ค้นถามที่สร้างจากนิพจน์พีชคณิตเชิงสัมพันธ์ในภาพที่ 8 [6]

2.2.4 Modelling ETL Conciliation Tasks Using Relational Algebra Operators [7]

Santos และ Belo [7] เสนอรูปแบบอ็อบเจกต์ที่แอตทริบิวต์ที่เกี่ยวข้องกับพีชคณิตเชิงสัมพันธ์ ดังนี้

$$\text{conc_dimtable} = \langle \text{SK}, \text{BKs}_1, \dots, \text{BKs}_n \rangle \quad (1)$$

$$\text{source_dimdatas}_x = \langle \text{BKs}_x, \text{Att}_1, \dots, \text{Att}_p \rangle \quad (2)$$

$$\text{conc_dimtables}_x = \langle \text{SKs}_x, \text{Att}_1, \dots, \text{Att}_p \rangle \quad (3)$$

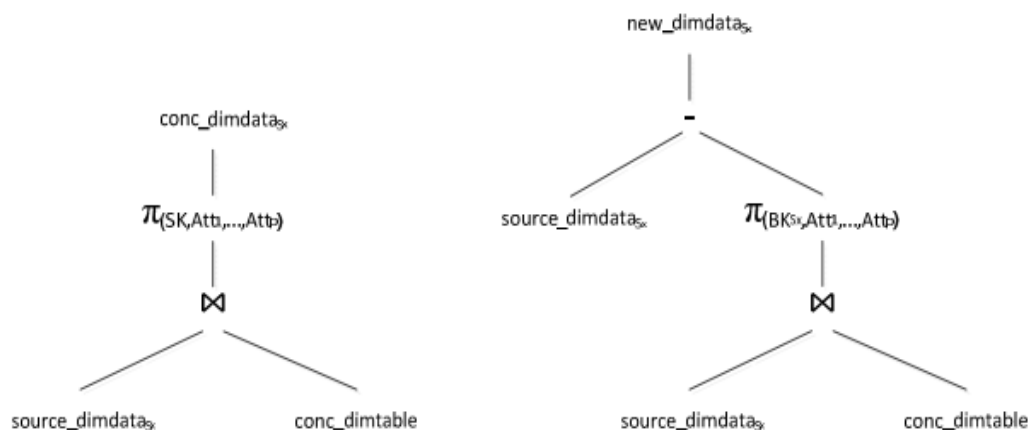
จากสมการที่ 1 ประกอบด้วยกุญแจตัวแทน (Surrogate Key : SK) ที่รับประกันข้อมูลจะสมบูรณ์ และกุญแจธุรกิจ (Business Keys : BK) จะรวบรวมข้อมูลจากต่างที่มา

ในสมการที่ 2 เมื่อกุญแจธุรกิจของแหล่งข้อมูล (Source) S_x โดยที่ค่าของ X อยู่ระหว่าง $1 \leq$ และ $\leq n$, และ $\text{Att}_1, \dots, \text{Att}_p$ เป็นคุณสมบัติเพิ่มเติมที่จำเป็นสำหรับมิติ (Dimension)

สมการที่ 3 คือการแทนกุญแจธุรกิจด้วยกุญแจตัวแทน

แนวคิดในการเทียบพีชคณิตเชิงสัมพันธ์กับเอสคิวแอลในงานวิจัย [7] ผู้วิจัยสามารถนำมาประยุกต์ใช้ในส่วนของการออกแบบเอสคิวแอลภายในเครื่องมืออ็อบเจกต์ที่แอตทริบิวต์ เพื่อดำเนินการในสถานการณ์ต่างๆ เช่น

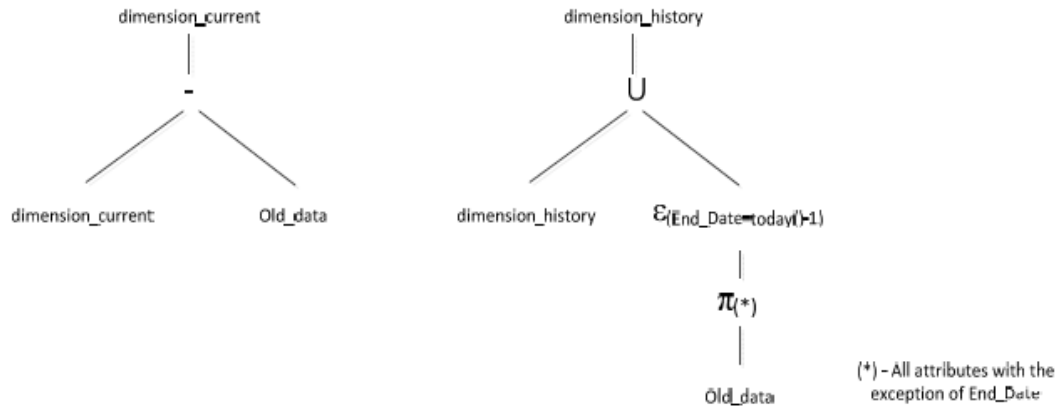
ภาพที่ 10 อธิบายพีชคณิตเชิงสัมพันธ์แบบต้นไม้ ภาพที่ 10 (a) แสดงกระบวนการรวมกันระหว่างแหล่งของข้อมูลและตารางจะให้ผลลัพธ์ในส่วนของการพิจารณาความเหมือนกับความต้องการของกุญแจตัวแทน ภาพที่ 10 (b) กระบวนการลบถูกใช้ในการลบจากข้อมูลต้นทางที่เหมือนกัน ตารางจึงเหลือส่วนที่ไม่เหมือนกันในตาราง โดยมีการจัดการทูเปิล (Tuple) ที่ถูกต้อง ผ่านการพิจารณาความเหมือนกับความต้องการของกุญแจตัวแทน



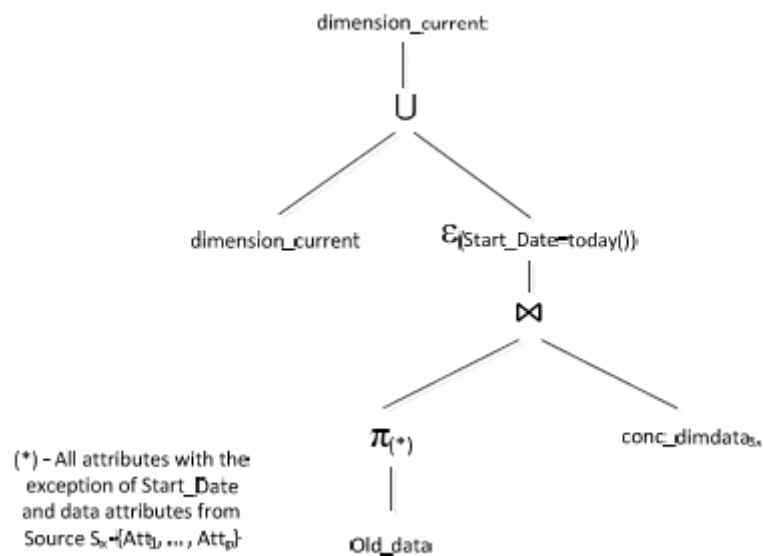
ภาพที่ 10 (a) ข้อมูลที่รวมกัน; (b) ข้อมูลที่ไม่ตรงกัน

ภาพที่ 11 นำเสนอการดำเนินการที่เหมาะสมในการถ่ายโอนข้อมูลที่หมดอายุจากมิติ โดยย้ายการจัดเก็บข้อมูลในปัจจุบันไปยังตารางที่เก็บข้อมูลในอดีตเพื่อรักษาข้อมูลในอดีต หลังจากลบ

ข้อมูลที่หมดอายุจากมิติ ขั้นตอนสุดท้ายที่จำเป็นคือ การเพิ่มค่าการปรับปรุงเพื่อให้มิติไม่สูญเสียการติดต่อกับข้อมูลที่ได้รับจากแหล่งอื่น ดังแสดงในภาพที่ 12

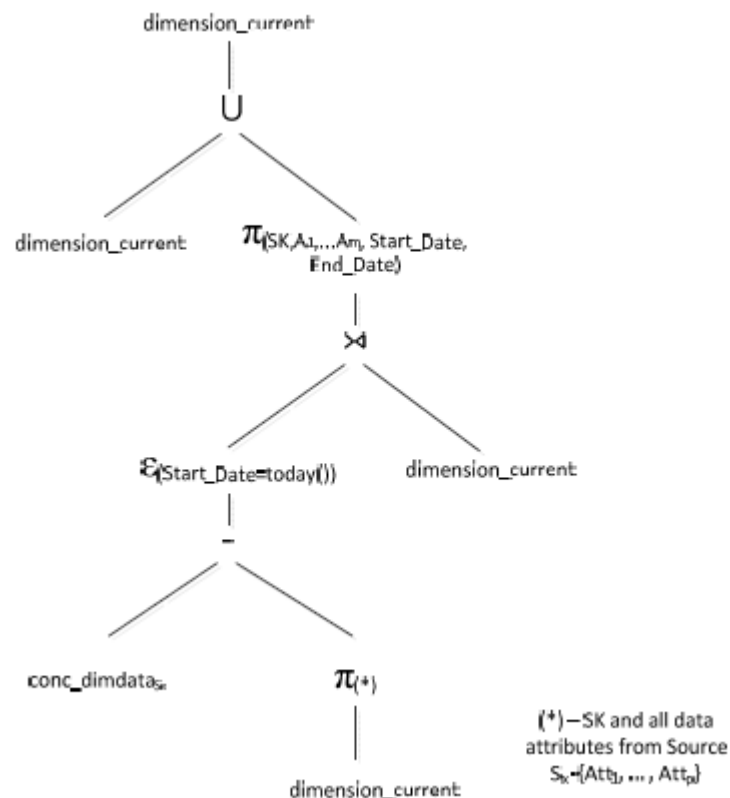


ภาพที่ 11 การส่งข้อมูลที่หมดอายุจากตารางมิติไปยังตารางที่เก็บข้อมูลในอดีต



ภาพที่ 12 ปรับปรุงมิติด้วยการแปลงข้อมูล

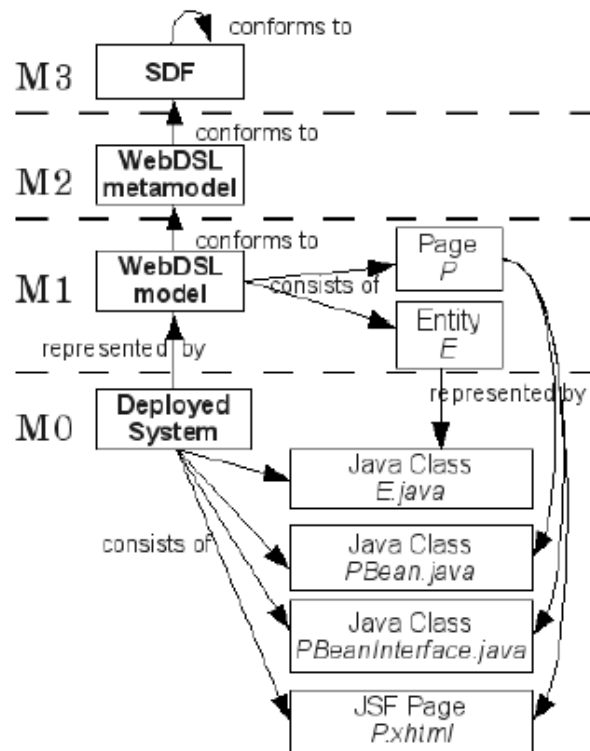
ภาพที่ 13 อธิบายความสัมพันธ์กับข้อมูลใหม่จาก conc_dimdata ซึ่งขึ้นอยู่กับ การตัดสินใจของทุเปิลที่กุญแจตัวแทนไม่มีอยู่ในมิติ หรือเรียกอีกอย่างว่ากระบวนการรวมกันแบบ Left Outer Join จะได้สมการดังนี้ Dimension_current = <SK,A1, ... , Am,DateFrom,DateTo>



ภาพที่ 13 พืชคณิตเชิงสัมพันธ์ต้นไม้อำหรับลักษณะข้อมูลใหม่

2.2.5 Code Generation by Model Transformation. A Case Study in Transformation Modularity [8]

Hemel และคณะ [8] นำเสนอกรณีศึกษาของการแปลงโค้ดโดยวิธีการแปลงแบบจำลอง (model transformation) ประยุกต์กับการพัฒนาเว็บดีเอสแอล (WebDSL) แบบไดนามิกเว็บแอปพลิเคชัน โดยโมเดลมี 4 ระดับของการแปลงจาวาเป็นเอกซ์เอ็มแอล ดังนั้น ระดับที่ M3 จะค้นหากรรมมาของวากยสัมพันธ์ที่กำหนดไว้ในระดับ M2 จะค้นหาเว็บดีเอสแอลเมทาโมเดล เช่น ไวยากรณ์ของเว็บดีเอสแอล ระดับ M1 จะค้นหาเว็บดีเอสแอลโมเดลของเว็บแอปพลิเคชันประกอบด้วยเอกลักษณ์ของหน้าเว็บ และระดับ M0 จะค้นหาเว็บที่ประกอบด้วยจาวาคลาสและหน้าXHTML ในภาพที่ 14



ภาพที่ 14 โครงสร้างของการแปลงแบบจำลองเว็บดีเอสแอล

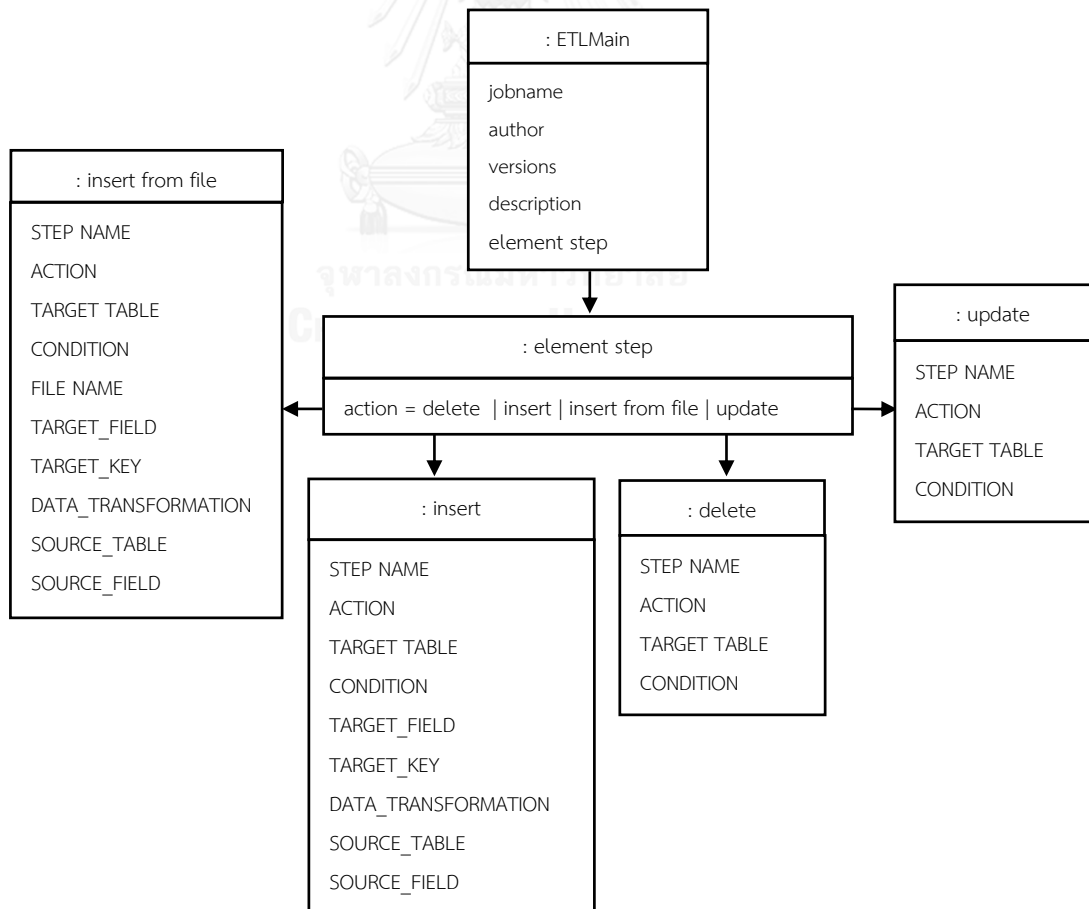
บทที่ 3

วิธีดำเนินการวิจัย

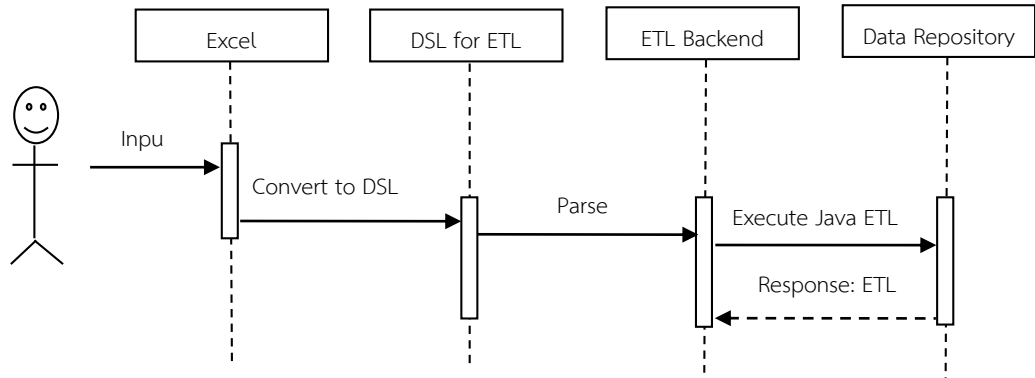
3.1 แนวคิดและวิธีวิจัย

3.1.1 การออกแบบอีทีแอลดีเอสแอล (etlDSL)

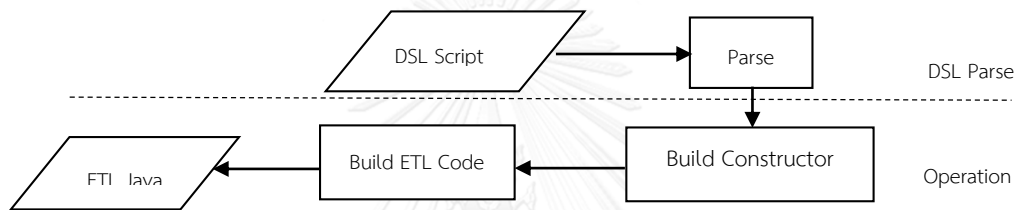
คำสั่งดีเอสแอลประกอบด้วยชุดคำที่คุ้นเคยและใช้งานเป็นประจำ ซึ่งชุดคำจะเกี่ยวข้องกับงานด้านอีทีแอลซึ่งออกแบบให้ง่ายต่อการใช้งาน งานวิจัยนี้พัฒนาดีเอสแอลเพื่อสร้างอีทีแอลสคริปต์ภาพที่15 ดีเอสแอลเมทาโมเดลที่อธิบายถึงแกรมมาของภาษา การออกแบบอีทีแอลดีเอสแอลพัฒนาเพื่อให้ใช้งานง่ายสำหรับผู้ที่ไม่มีความรู้พื้นฐานการเขียนโปรแกรม เช่น นักวิเคราะห์ธุรกิจส่วนของไอที (Business Analyst, BA) ซึ่งประกอบด้วยคำสำคัญหลักคือ คำอธิบายกระบวนการ, การลบข้อมูล, การแปลงข้อมูล, การนำเข้าข้อมูลจากข้อมูลที่มีอยู่ในฐานข้อมูลและจากไฟล์ข้อมูล ภาพที่16 คือภาพรวมของกระบวนการอีทีแอล โดยผู้ใช้งานจะทำการเขียนกระบวนการอีทีแอลในเอกสารรูปแบบ excel และเมื่อทำการบันทึกไฟล์จะถูกแปลงให้อยู่ในรูปแบบของสคริปต์อีทีแอลดีเอสแอลเพื่อใช้ในกระบวนการประมวลผลข้อมูลลงในตารางฐานข้อมูล



ภาพที่ 15 ดีเอสแอลเมทาโมเดล (DSL Meta-model)

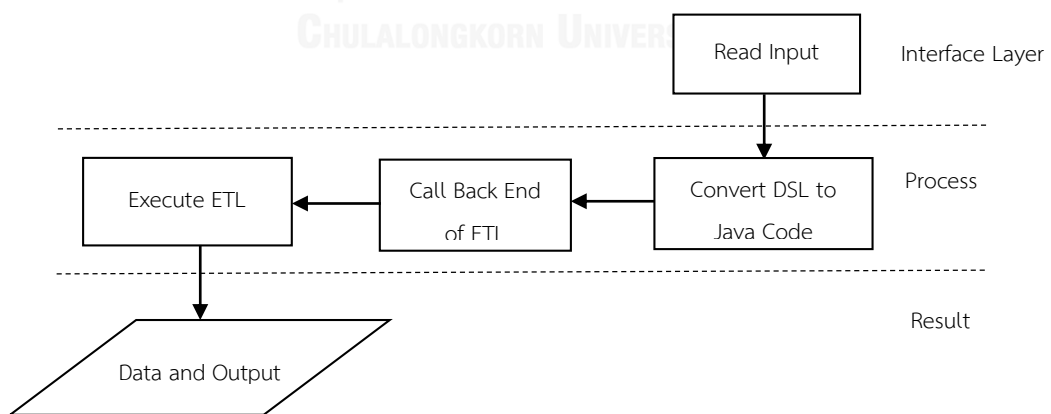


ภาพที่ 16 กระบวนการอีทีแอลจากอีทีแอลดีเอสแอล



ภาพที่ 17 การแปลงดีเอสแอลให้กลายเป็นโค้ดอีทีแอล

ภาพที่ 17 วิธีการสร้างไวยากรณ์ดีเอสแอลเพื่อสร้างโค้ดอีทีแอล โดยเริ่มจากการออกแบบไวยากรณ์ดีเอสแอลสคริปต์, กำหนดความหมายของไวยากรณ์ดีเอสแอล และสร้างวิธีการแยกส่วนไวยากรณ์ดีเอสแอลสคริปต์ ก่อนจะนำผลที่ได้มาจัดรูปแบบให้อยู่ในลักษณะของโค้ดอีทีแอลที่พร้อมใช้งาน โดยประยุกต์ใช้จากแนวทางที่นำเสนอในงานวิจัย [4]



ภาพที่ 18 ขั้นตอนการอ่านไวยากรณ์ดีเอสแอล

ภาพที่ 18 แสดงการประยุกต์ใช้แนวทางที่นำเสนอในงานวิจัย [5] สำหรับวิธีการใช้งานคำสั่งดีเอสแอลในงานวิจัยนี้ ซึ่งสามารถกำหนดข้อมูลเข้าได้ 2 แบบ คือ 1) สร้างสคริปต์อีทีแอลดีเอสแอล

ด้วย excel หรือ 2) การเขียนดีเอสแอลสคริปต์นำเข้าโดยตรงซึ่งเป็นการกำหนดลำดับคำสั่งการทำงานของอีทีแอล จากนั้นจึงทำการแปลงลำดับคำสั่งดังกล่าวให้อยู่ในลักษณะของคำสั่งโค้ดอีทีแอลเป็นภาษาจาวา เพื่อเรียกการทำงานจากโปรแกรมหลังบ้าน (Back End of ETL) ซึ่งโปรแกรมหลังบ้านจะประกอบด้วยชุดคำสั่งภาษาจาวาในการทำกระบวนการอย่างใดอย่างหนึ่ง หลังจากนั้นโปรแกรมจะทำการประมวลผลเพื่อให้บรรลุเป้าหมายที่ตั้งไว้

3.1.2 การพัฒนาตัวแจงส่วน (Parser)

การพัฒนาตัวแจงส่วนโดยใช้เครื่องมือXtext เพื่อสร้างไวยากรณ์สำหรับอีทีแอลดีเอสแอล โดยพัฒนาภาษาจากคำศัพท์ที่เป็นพื้นฐานของกระบวนการอีทีแอลและเอสคิวแอล ตามดีเอสแอลเมทาโมเดลภาพที่ 15 อธิบายถึงวากยสัมพันธ์ของภาษา เพื่อตอบสนองการใช้งานของผู้ใช้ให้ง่ายขึ้นซึ่งการออกแบบตามวากยสัมพันธ์ในภาพที่ 19 และ ตารางรายละเอียดดังตารางที่ 3

```

grammar th.ac.chula.eng.cp.dsl.Etl with org.eclipse.xtext.common.Terminals
generate etl "DSL_ETL"
ETLMain:
    "jobname:" jobname=ID
    "author:" author=TEXT
    "version:" version=VERSION
    "description:" descript=TEXT
    (element+=FullStepElement)* ;
FullStepElement: name=(DELETE_STEP|LOAD_STEP|FILELOAD_STEP) ;
DELETE_STEP:
"STEP" "NAME|" stepName=TEXT COLUMNDELIMITOR
"ACTION|" typeName=TYPENAME COLUMNDELIMITOR
"TARGET" "TABLE|" targetName=TEXT COLUMNDELIMITOR
"CONDITION|" condition=STRING COLUMNDELIMITOR ;
LOAD_STEP:
"STEP" "NAME|" stepName=TEXT COLUMNDELIMITOR
"ACTION|" typeName=TYPENAME COLUMNDELIMITOR
"TARGET" "TABLE|" targetName=TEXT ('(loadType=LOADTYPE)') COLUMNDELIMITOR
"CONDITION|" condition=STRING COLUMNDELIMITOR
"COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|S

```

```

SOURCE_FIELD"
(value+=VALUE)* ;
FILELOAD_STEP:
"STEP" "NAME|" stepName=TEXT COLUMNDELIMITOR
"ACTION|" typeName=TYPENAME '|' delimiter=TEXT '|' header=TEXT '|'
footer=TEXT'|' quote=TEXT
"TARGET" "TABLE|" targetName=TEXT ('loadType=LOADTYPE') COLUMNDELIMITOR
"CONDITION|" condition=STRING COLUMNDELIMITOR
"FILENAME|" filename=TEXT COLUMNDELIMITOR
"COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|S
OURCE_FIELD"
(value+=VALUE)* ;
VALUE:
"VALUES|" targetCol=TEXT "|" colKey=(TEXT)? "|" transform=(TRANSFORM)? "|"
sourceTab=(TEXT)? "|" sourceCol=(TEXT)? ;
TRANSFORM:
("###"? (TEXT | INT | STRING | ((TEXT | INT | STRING) '(')* ((TEXT | INT | STRING) ','))*
(TEXT | INT | STRING) '(' (TEXT | INT | STRING) (',' (TEXT | INT | STRING))* ')' (',' (TEXT
| INT | STRING))* ('))* ) ("###"? | "###" "###" ;
TYPENAME: ("delete|"clear|"insert|"update") ;
VERSION:'v' INT (.'INT)* ;
COLUMNDELIMITOR: '|' (|)* ;
LOADTYPE:
("update""and""insert|"clear""and""insert|"insert|"insert""from""file|"update""from""v
alue|"update""from""table") ;
TEXT: ID (.' ) ? " ) ? ( ID ) * " ) ? ;

```

ภาพที่ 19 วากยสัมพันธ์สำหรับสร้างไวยากรณ์ของดีเอสแอล

ตารางที่ 3 คำอธิบายคำสำคัญวากยสัมพันธ์สำหรับสร้างไวยากรณ์ของดีเอสแอลในภาพที่ 19

คำสำคัญ	รายละเอียด	ตัวอย่างการใช้งาน
ETLMain	คำอธิบาย ของกระบวนการ อีทีแอล	jobname: InsetDataIntoCAMPAING_FACT author: Sunisa J version: v 1 description: This is the test script
DELETE_STEP	สำหรับ กระบวนการ ลบข้อมูล	STEP NAME Delete CAMPAING ACTION delete TARGET TABLE DW_CAMPAIGN_FACT CONDITION "(STORE_NM,DT_SR_KEY) in (select L.STORE_NM,L.DT_SR_KEY from STAGE_CAMPAIGN_FCT C, DW_CAMPAIGN_FACT L where c.STORE_NM = L.STORE_NM and c.DT_SR_KEY = L.DT_SR_KEY)"
LOAD_STEP	สำหรับ กระบวนการ นำเข้าข้อมูล	STEP NAME CAMPAING use_cards1 ACTION insert TARGET TABLE DW_CAMPAIGN_FACT (insert) CONDITION "DW_SALE_ATTRIBUTE.saleheaderid = DW_SALE_HEADER_FACT.sale_id COLUMNS TARGET_FIELD TARGET_KEY DATA_TR ANSFORMATION SOURCE_TABLE SOURCE_FIELD VALUES DT_SR_KEY key move direct DW_SALE_HEADER_FACT dt_sr_key

คำสำคัญ	รายละเอียด	ตัวอย่างการใช้งาน
FILELOAD_STEP	สำหรับนำ ข้อมูลเข้า จากไฟล์	STEP NAME Load From File ACTION insert comma yes no no TARGET TABLE TABLEA (clear and insert) CONDITION "" FILENAME testLoad.txt COLUMNS TARGET_FIELD TARGET_KEY DATA_TRANSFORMATION SOURCE_TABLE SOURCE_FIELD VALUES ATT1 move direct testLoad A

Rule1: Delete

1. Get parameters from DSL (type, table name, condition)
2. *If* type equals 'clear' *then* truncate table target *else* get target table name and condition to generate SQL *delete* statement
3. Execute statement of (2)

Rule3: Update from database/ input value(s)

1. Get parameters from DSL (type, table name, condition)
2. Load mapping values to mapping table
3. Create SQL *merge* statement from mapping table
If type equals 'update and insert' *then* generate 'when not match then insert' *else* does not generate 'when not match then insert'
4. Execute statement (3)
5. Clear data in mapping table

Rule2: Insert from file

1. Get parameters from DSL (type, table name, file name, mapping values)
2. Load mapping values to mapping table
3. Create temporary table based on column of file
4. Load file to temporary table
5. Generate SQL statement from data in mapping table
6. *If* type equals 'clear and insert' *then* delete data in target table and generate SQL *insert* statement on data in mapping table *else* generate SQL *insert* statement on data in mapping table
7. Execute statement of (6)
8. Drop temporary table and clear data in mapping table

ภาพที่ 20 ตัวอย่างกฎการแปลงเป็นเอสคิวแอล

ตัวอย่างกฎการแปลง (Transformation Rules) จากสคริปต์อีทีแอลดีเอสแอลเป็นเอสคิวแอล ประกอบด้วยคำสั่งการลบข้อมูล ซึ่งรับค่าพารามิเตอร์โดยเช็คเงื่อนไขว่าเข้ากรณีใด เช่น truncate คือ การลบข้อมูลทั้งหมดแบบไร้เงื่อนไข หรือ delete คือการลบข้อมูลแบบมีเงื่อนไข, การนำเข้าข้อมูลจากไฟล์โดยตรวจสอบเงื่อนไขที่กำหนดให้เป็นลักษณะใด จะเลือกทำการลบข้อมูลที่มีอยู่ก่อนหรือนำเข้าข้อมูลได้เลย หลังจากนั้นทำการแปลงข้อมูลหรือการปรับให้เป็นปัจจุบันตามที่กำหนดไว้ลงในฐานข้อมูล และนำเข้าข้อมูลจากฐานข้อมูลที่มีอยู่เดิมในฐานข้อมูลตามภาพที่ 20

เมื่อผู้ใช้เขียนสคริปต์อีทีแอลดีเอสแอลแล้วตัวแองจส่วนจะทำการตัดค่าตามความหมายเพื่อแปลงเป็นจาวาที่จะทำการเรียกใช้งานเครื่องมืออีทีแอลหลังบ้านดังนี้

1. การลบข้อมูล

เริ่มต้นจากการตรวจสอบว่าการลบข้อมูลดังกล่าวเป็นการลบข้อมูลแบบมีเงื่อนไขหรือไม่ จากนั้นทำการสั่งงาน อีทีแอลหลังบ้านเพื่อทำการลบข้อมูลในฐานข้อมูลตามเงื่อนไขที่กำหนด

```
«IF DELETE_STEP.condition.equals(null) || DELETE_STEP.typeName.equals("clear") »
result = etl.etlDelete("«DELETE_STEP.targetName»");
«ELSE»
result = etl.etlDelete("«DELETE_STEP.targetName»", "«DELETE_STEP.condition»");
«ENDIF»
```

2.การนำเข้าข้อมูล

เริ่มต้นจากการตรวจสอบว่าเป็นกระบวนการนำเข้าข้อมูลหรือไม่

```
«IF LOAD_STEP.typeName.equals("insert")»
ทำการลบข้อมูลเดิมที่มีอยู่ในตาราง Mapping Value
etl.etlDelete("MAPPING_VALUE_TABLE");
ทำการนำเข้าข้อมูลลงตาราง Mapping Value เพื่อเตรียมกระบวนการขั้นต่อไป
«FOR VALUE : element.eAllContents.tolterable.filter(VALUE)»
result =
etl.etlInsertMapTable("VALUES','«VALUE.targetCol»','«VALUE.colKey»','«VALUE.transform
»','«VALUE.sourceTab»','«VALUE.sourceCol»");
«ENDFOR»
```

จากนั้นทำการนำเข้าข้อมูลจากเงื่อนไขตามที่กำหนดไว้ในตาราง Mapping Value

```
result = etl.etlInsertFromMapping("«LOAD_STEP.targetName»",
"«LOAD_STEP.condition»");
```

3. การแปลงข้อมูล

เริ่มต้นจากการตรวจสอบเงื่อนไขว่าเป็นการแปลงข้อมูลหรือไม่

```
«IF LOAD_STEP.loadType.equals("update")»
```

ทำการลบข้อมูลเดิมที่มีอยู่ในตาราง Mapping Value และนำเข้าข้อมูลใหม่เพื่อเตรียมกระบวนการ
ขั้นต่อไป

```
etl.etlDelete("MAPPING_VALUE_TABLE");
```

```
«FOR VALUE : element.eAllContents.tolterable.filter(VALUE)»
```

```
result =
```

```
etl.etlInsertMapTable("VALUES','«VALUE.targetCol»','«VALUE.colKey»','«VALUE.transform  
»','«VALUE.sourceTab»','«VALUE.sourceCol»");
```

```
«ENDFOR»
```

จากนั้นทำการแปลงข้อมูลตามที่ระบุไว้ในตาราง Mapping

```
result = etl.etlUpdateData("«LOAD_STEP.targetName»", "«LOAD_STEP.condition»",
"«LOAD_STEP.loadType»");
```

4. การแปลงข้อมูลและนำเข้าข้อมูล

การแปลงข้อมูลและนำเข้าข้อมูลจะใช้กระบวนการที่คล้ายคลึงกับกระบวนการแปลงข้อมูล
เพียงแต่จะต่างตรงที่มีการส่งผ่านตัวแปรไปที่อีทีแอลหลังบ้านให้ทำการนำเข้าข้อมูลหลังจากมีการ
แปลงข้อมูลเสร็จเรียบร้อยแล้ว

บทที่ 4

การออกแบบและการพัฒนาระบบ

4.1 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา

4.1.1 สภาพแวดล้อม

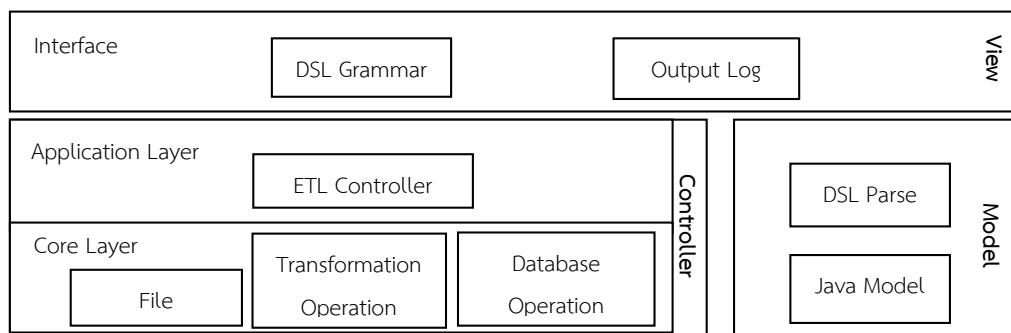
- 1.ระบบปฏิบัติการ Window 7 แบบ 64 bit
- 2.หน่วยประมวลผล Processor Core i3, CpU 2.10 GHz
- 3.หน่วยความจำ 4 กิกะไบต์ (RAM 4 GB)

4.1.2 เครื่องมือที่ใช้ในการพัฒนา

1. Eclipse IDE for Java Developers Version: Neon.1a Release (4.6.1)
2. Xtext

4.2 การออกแบบสถาปัตยกรรม

งานวิจัยนี้มุ่งเน้นที่จะนำเสนอวิธีการสร้างไวยากรณ์และพัฒนาดีเอสแอล เพื่อสนับสนุนการสั่งการอีทีแอลบนพื้นฐานของภาษา (code-based ETL) สำหรับเตรียมข้อมูลนำเข้าจากแหล่งข้อมูลวิวิธภักดิ์ (heterogeneous data sources) วิธีการที่นำเสนอขึ้นคือ การแปลงจากดีเอสแอลสคริปต์เป็นสคริปต์อีทีแอลจาวาโดยออกแบบสถาปัตยกรรมในรูปแบบของเอ็มวีซี Model-View-Controller (MVC) ในภาพที่ 21 ประกอบด้วย 1) ส่วนแสดงผล (View component หรือ interface layer) ประกอบด้วย ไวยากรณ์ดีเอสแอล (DSL grammar) และหน้าจอแสดงผล (output log) สำหรับรับคำสั่งและแสดงผล 2) ส่วนตัวควบคุม (Controller component) ประกอบด้วยตัวควบคุมอีทีแอล (ETL controller), ตัวควบคุมไฟล์ (File controller), ตัวควบคุมการแปลง (Transformation Operation controller), และ ตัวควบคุมการดำเนินการฐานข้อมูล (Database Operation controller) 3) ส่วนโมเดล (Model component) ประกอบด้วยตัวแจงส่วนดีเอสแอล (DSL parse) และจาวาโมเดล (Java model) คือการแปลงไวยากรณ์ดีเอสแอล และการรับส่งคำสั่งระหว่างตัวควบคุม และส่วนแสดงผล



ภาพที่ 21 การออกแบบสถาปัตยกรรมระบบด้วยเอ็มวีซี

4.3 ตัวอย่างส่วนแสดงผล และ จาวาที่ได้จากสคริปต์อีทีแอลดีเอสแอล

การออกแบบสถาปัตยกรรมระบบด้วยเอ็มวีซี หรือ Model-View-Controller ในภาพที่ 21 ประกอบด้วย 1) ส่วนแสดงผล (View component หรือ interface layer) ประกอบด้วยไวยากรณ์ดีเอสแอล (DSL grammar) และหน้าจอแสดงผล (output log) สำหรับรับคำสั่งและแสดงผล 2) ส่วนตัวควบคุม (Controller component) ประกอบด้วยตัวควบคุมอีทีแอล (ETL controller), ตัวควบคุมไฟล์ (File controller), ตัวควบคุมการแปลง (Transformation Operation controller), และ ตัวควบคุมการดำเนินการฐานข้อมูล (Database Operation controller) 3) ส่วนโมเดล (Model component) ประกอบด้วยตัวแจงส่วนดีเอสแอล (DSL parse) และจาวาโมเดล (Java model) คือ การแปลงไวยากรณ์ดีเอสแอล และการรับส่งคำสั่งระหว่างตัวควบคุม และส่วนแสดงผล

จุฬาลงกรณ์มหาวิทยาลัย

4.3.1 ตัวอย่างการเขียนสคริปต์อีทีแอลดีเอสแอลเพื่อลบข้อมูลจากราง DW_CAMPAIGN_FACT โดยเงื่อนไขการลบข้อมูลคือ (STORE_NM,DT_SR_KEY) in (select l.STORE_NM,l.DT_SR_KEY from STAGE_CAMPAIGN_FCT C, DW_CAMPAIGN_FACT l where c.STORE_NM = l.STORE_NM and c.DT_SR_KEY = l.DT_SR_KEY) และ สคริปต์อีทีแอลจาวาในภาพที่ 23 ที่ได้จาก สคริปต์อีทีแอลดีเอสแอลในภาพที่ 22

```

STEP NAME|Delete CAMPAING|
ACTION|delete|
TARGET TABLE|DW_CAMPAIGN_FACT|
CONDITION|"(STORE_NM,DT_SR_KEY) in (select l.STORE_NM,l.DT_SR_KEY from
STAGE_CAMPAIGN_FCT C, DW_CAMPAIGN_FACT l where c.STORE_NM =
l.STORE_NM and c.DT_SR_KEY = l.DT_SR_KEY)"|

```

ภาพที่ 22 ตัวอย่างการลบข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอล

```

        result=-99;
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("delete"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Starting Step Delete
CAMPAING");
        System.out.println(logDate+logTime+logType+"Deleting data on
DW_CAMPAIGN_FACT");
        result = etl.etlDelete("DW_CAMPAIGN_FACT",
"(STORE_NM,DT_SR_KEY) in (select l.STORE_NM,l.DT_SR_KEY from
STAGE_CAMPAIGN_FCT C, DW_CAMPAIGN_FACT l where c.STORE_NM =
l.STORE_NM and c.DT_SR_KEY = l.DT_SR_KEY)");
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        System.out.println(logDate+logTime+logType+"Deleted data done
"+result+" record(s)");
        if (result >= 0){
            logType = ("delete"+tab).substring(0,15);
            System.out.println(logDate+logTime+logType+"Finished
Step Delete CAMPAING");
        }else{
            logType = ("error"+tab).substring(0,15);
            System.err.println(logDate+logTime+logType+"Failed Step

```

```

Delete CAMPAING");

                System.exit(-1);

        }

```

ภาพที่ 23 ตัวอย่างจาวาที่ได้จากการลบข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอลในภาพที่ 22

2017-06-17	21:17:43	delete	Starting Step Delete CAMPAING
2017-06-17	21:17:43	delete	Deleting data on DW_CAMPAIGN_FACT
2017-06-17	21:17:43	delete	Deleted data done 0 record(s)
2017-06-17	21:17:43	delete	Finished Step Delete CAMPAING

ภาพที่ 24 ตัวอย่าง Ouput log ใน view component จากภาพที่ 23

4.3.2 ตัวอย่างการเขียนสคริปต์อีทีแอลดีเอสแอลเพื่อนำเข้าข้อมูลตาราง DW_CAMPAIGN_FACT และ สคริปต์อีทีแอลจาวาในภาพที่ 26 ที่ได้จาก สคริปต์อีทีแอลดีเอสแอลในภาพที่ 25

```

STEP NAME|CAMPAING use_cards1|
ACTION|insert|
TARGET TABLE|DW_CAMPAIGN_FACT (insert)|
CONDITION|"DW_SALE_ATTRIBUTE.saleheaderid = DW_SALE_HEADER_FACT.sale_id
and DW_SALE_HEADER_FACT.STORE_SR_KEY = DW_STORE_DIM.STORE_ID and
DW_SALE_ATTRIBUTE.ATTRIBUTE056 = '2' group by
dw_sale_header_fact.dt_sr_key, dw_sale_header_fact.channel_sr_key,
dw_sale_header_fact.store_sr_key, dw_store_dim.store_nm,
DW_SALE_ATTRIBUTE.attribute046, DW_SALE_ATTRIBUTE.saleheaderid,
DW_SALE_ATTRIBUTE.ATTRIBUTE054, DW_SALE_ATTRIBUTE.ATTRIBUTE055,
DW_SALE_ATTRIBUTE.ATTRIBUTE057"|
COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|SO
URCE_FIELD
VALUES|DT_SR_KEY|key|move direct|DW_SALE_HEADER_FACT|dt_sr_key
VALUES|CHANNEL_SR_KEY|key|move
direct|DW_SALE_HEADER_FACT|channel_sr_key
VALUES|STORE_SR_KEY|key|move direct|DW_SALE_HEADER_FACT|store_sr_key
VALUES|STORE_NM||move direct|DW_STORE_DIM|store_nm

```

```

VALUES|CARD_NUMBER||move direct|DW_SALE_ATTRIBUTE|attribute046
VALUES|SALE_CARD_SALEID||move direct|DW_SALE_ATTRIBUTE|saleheaderid
VALUES|USE_CARD_SALEID|||
VALUES|SALE_CARD_QTY||0|
VALUES|USE_CARD_QTY||nvl(sum(attribute043),0)|
VALUES|TRUN_TYPE||###USE###|
VALUES|DATA_LOAD_DT||sysdate|
VALUES|DATA_UPDT_DT||sysdate|
VALUES|ATTR||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE054
VALUES|ATTR2||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE055
VALUES|ATTR3||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE057
VALUES|ATTR4||### ###|
VALUES|ATTR5||### ###|
VALUES|NUM_SALE_CARD_QTY||0|
VALUES|NUM_USE_CARD_QTY||1|

```

ภาพที่ 25 ตัวอย่างการนำเข้าข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอล

```

result=-99;
logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("load"+tab).substring(0,15);
System.out.println(logDate+logTime+logType+"Starting Step CAMPAING
use_cards1");
System.out.println(logDate+logTime+logType+"Loading data on
DW_CAMPAIGN_FACT");
logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("load"+tab).substring(0,15);
System.out.println(logDate+logTime+logType+"Loading Mapping Table");
etl.etlDelete("MAPPING_VALUE_TABLE");
result = etl.etlInsertMapTable("VALUES','DT_SR_KEY','key','move

```

```

direct','DW_SALE_HEADER_FACT','dt_sr_key");
        if(result < 0){
logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("error"+tab).substring(0,15);
System.err.println(logDate+logTime+logType+"Error During Loading Mapping
Table");
System.err.println(logDate+logTime+logType+"VALUES','DT_SR_KEY','key','"+transfor
m+','DW_SALE_HEADER_FACT','dt_sr_key");
System.exit(-1);
}
result = etl.etlInsertMapTable("VALUES','CHANNEL_SR_KEY','key','move
direct','DW_SALE_HEADER_FACT','channel_sr_key");
if(result < 0){
logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("error"+tab).substring(0,15);
System.err.println(logDate+logTime+logType+"Error During Loading Mapping
Table");
System.err.println(logDate+logTime+logType+"VALUES','CHANNEL_SR_KEY','key','"+t
ransform+','DW_SALE_HEADER_FACT','channel_sr_key");
System.exit(-1);
}
...
        }
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("load"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Starting Insert data
to DW_CAMPAIGN_FACT");
        result = etl.etlInsertFromMapping("DW_CAMPAIGN_FACT",

```



```

"DW_SALE_ATTRIBUTE.saleheaderid = DW_SALE_HEADER_FACT.sale_id and
DW_SALE_HEADER_FACT.STORE_SR_KEY = DW_STORE_DIM.STORE_ID and
DW_SALE_ATTRIBUTE.ATTRIBUTE056 = '2' group by
dw_sale_header_fact.dt_sr_key, dw_sale_header_fact.channel_sr_key,
dw_sale_header_fact.store_sr_key, dw_store_dim.store_nm,
DW_SALE_ATTRIBUTE.attribute046, DW_SALE_ATTRIBUTE.saleheaderid,
DW_SALE_ATTRIBUTE.ATTRIBUTE054, DW_SALE_ATTRIBUTE.ATTRIBUTE055,
DW_SALE_ATTRIBUTE.ATTRIBUTE057");

        if(result < 0){
            logTime = (dtf.format(now).split(" ")[1])+datetab;
            logDate = (dtf.format(now).split(" ")[0])+datetab;
            logType = ("error"+tab).substring(0,15);
            System.err.println(logDate+logTime+logType+"Error During
Insert data to DW_CAMPAIGN_FACT");
            System.exit(-1);
        }
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("load"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Inserted data done
"+result+" record(s)");

        System.out.println(logDate+logTime+logType+"Finished Step
CAMPAING use_cards1");

```

ภาพที่ 26 ตัวอย่างจาวาที่ได้จากการนำเข้าข้อมูลกระบวนการของสคริปต์อีทีแอลดีเอสแอล
ในภาพที่ 25

2017-06-17	21:17:43	load	Starting Step CAMPAING use_cards1
2017-06-17	21:17:43	load	Loading data on DW_CAMPAIGN_FACT
2017-06-17	21:17:43	load	Loading Mapping Table
2017-06-17	21:17:43	load	Starting Insert data to DW_CAMPAIGN_FACT
2017-06-17	21:17:43	load	Inserted data done 0 record(s)
2017-06-17	21:17:43	load	Finished Step CAMPAING use_cards1

ภาพที่ 27 ตัวอย่าง Output log ใน view component จากภาพที่ 26

4.3.3 ตัวอย่างการเขียนสคริปต์อีทีแอลดีเอสแอลเพื่อแก้ไขข้อมูลในตาราง DW_CAMPAIGN_FACT โดยการปรับให้เป็นปัจจุบัน และ สคริปต์อีทีแอลจาวาในภาพที่ 29 ที่ได้จาก สคริปต์อีทีแอลดีเอสแอล ในภาพที่ 28

```
STEP NAME|CAMPAING use_cards2|
ACTION|update|
TARGET TABLE|DW_CAMPAIGN_FACT (update from table)|
CONDITION|"DW_SALE_ATTRIBUTE.saleheaderid = DW_SALE_HEADER_FACT.sale_id
and DW_SALE_HEADER_FACT.STORE_SR_KEY = DW_STORE_DIM.STORE_ID and
DW_SALE_ATTRIBUTE.ATTRIBUTE056 = '2' group by
dw_sale_header_fact.dt_sr_key, dw_sale_header_fact.channel_sr_key,
dw_sale_header_fact.store_sr_key, dw_store_dim.store_nm,
DW_SALE_ATTRIBUTE.attribute046, DW_SALE_ATTRIBUTE.saleheaderid,
DW_SALE_ATTRIBUTE.ATTRIBUTE054, DW_SALE_ATTRIBUTE.ATTRIBUTE055,
DW_SALE_ATTRIBUTE.ATTRIBUTE057"|
COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|SO
URCE_FIELD
VALUES|DT_SR_KEY|key|move direct|DW_SALE_HEADER_FACT|dt_sr_key
VALUES|CHANNEL_SR_KEY|key|move
direct|DW_SALE_HEADER_FACT|channel_sr_key
VALUES|STORE_SR_KEY|key|move direct|DW_SALE_HEADER_FACT|store_sr_key
VALUES|STORE_NM||move direct|DW_STORE_DIM|store_nm
VALUES|CARD_NUMBER||move direct|DW_SALE_ATTRIBUTE|attribute046
```

```

VALUES|SALE_CARD_SALEID||move direct|DW_SALE_ATTRIBUTE|saleheaderid
VALUES|USE_CARD_SALEID|||
VALUES|SALE_CARD_QTY||0|
VALUES|USE_CARD_QTY||nvl(sum(attribute043),0)|
VALUES|TRUN_TYPE|###USE###|
VALUES|DATA_LOAD_DT||sysdate|
VALUES|DATA_UPDT_DT||sysdate|
VALUES|ATTR||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE054
VALUES|ATTR2||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE055
VALUES|ATTR3||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE057
VALUES|ATTR4|### ###|
VALUES|ATTR5|### ###|
VALUES|NUM_SALE_CARD_QTY||0|
VALUES|NUM_USE_CARD_QTY||1|

```

ภาพที่ 28 ตัวอย่างสคริปต์อีทีแอลดีเอสแอลเพื่อการแก้ไขข้อมูลในตารางโดยการปรับให้เป็นปัจจุบัน

```

result=-99;

    logTime = (dtf.format(now).split(" ")[1])+datetab;
    logDate = (dtf.format(now).split(" ")[0])+datetab;
    logType = ("update"+tab).substring(0,15);
    System.out.println(logDate+logTime+logType+"Starting Step
CAMPAING use_cards2");
    System.out.println(logDate+logTime+logType+"Update data on
DW_CAMPAIGN_FACT");
    logTime = (dtf.format(now).split(" ")[1])+datetab;
    logDate = (dtf.format(now).split(" ")[0])+datetab;
    logType = ("update"+tab).substring(0,15);
    System.out.println(logDate+logTime+logType+"Loading Mapping
Table");

    etl.etlDelete("MAPPING_VALUE_TABLE");
    result = etl.etlInsertMapTable("VALUES','DT_SR_KEY','key','move

```

```

direct', 'DW_SALE_HEADER_FACT', 'dt_sr_key');
        if(result < 0){
            logTime = (dtf.format(now).split(" ")[1])+datetab;
            logDate = (dtf.format(now).split(" ")[0])+datetab;
            logType = ("error"+tab).substring(0,15);
            System.err.println(logDate+logTime+logType+"Error During
Loading Mapping Table");

            System.err.println(logDate+logTime+logType+"VALUES', 'DT_SR_KEY', 'key', '"+
transform+", 'DW_SALE_HEADER_FACT', 'dt_sr_key");
            System.exit(-1);
        ...
        }
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("update"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Starting Update data
to DW_CAMPAIGN_FACT");
        result = etl.etlUpdateData("DW_CAMPAIGN_FACT",
"DW_SALE_ATTRIBUTE.saleheaderid = DW_SALE_HEADER_FACT.sale_id and
DW_SALE_HEADER_FACT.STORE_SR_KEY = DW_STORE_DIM.STORE_ID and
DW_SALE_ATTRIBUTE.ATTRIBUTE056 = '2' group by
dw_sale_header_fact.dt_sr_key, dw_sale_header_fact.channel_sr_key,
dw_sale_header_fact.store_sr_key, dw_store_dim.store_nm,
DW_SALE_ATTRIBUTE.attribute046, DW_SALE_ATTRIBUTE.saleheaderid,
DW_SALE_ATTRIBUTE.ATTRIBUTE054, DW_SALE_ATTRIBUTE.ATTRIBUTE055,
DW_SALE_ATTRIBUTE.ATTRIBUTE057", "update from table");
        if(result < 0){
            logTime = (dtf.format(now).split(" ")[1])+datetab;
            logDate = (dtf.format(now).split(" ")[0])+datetab;
            logType = ("error"+tab).substring(0,15);

```

```

        System.err.println(logDate+logTime+logType+"Error During
Update data to DW_CAMPAIGN_FACT");
        System.exit(-1);
    }
    logTime = (dtf.format(now).split(" ")[1])+datetab;
    logDate = (dtf.format(now).split(" ")[0])+datetab;
    logType = ("update"+tab).substring(0,15);
    System.out.println(logDate+logTime+logType+"Updated data done
"+result+" record(s)");
    System.out.println(logDate+logTime+logType+"Finished Step
CAMPAING use_cards2");

```

ภาพที่ 29 ตัวอย่างจาวาที่ได้จากสคริปต์อีทีแอลดีเอสแอลเพื่อการแก้ไขข้อมูลในตารางโดยการปรับให้เป็นปัจจุบันในภาพที่ 28

2017-06-17	21:17:43	update	Starting Step CAMPAING use_cards2
2017-06-17	21:17:43	update	Update data on DW_CAMPAIGN_FACT
2017-06-17	21:17:43	update	Loading Mapping Table
2017-06-17	21:17:43	update	Starting Update data to DW_CAMPAIGN_FACT
2017-06-17	21:17:43	update	Updated data done 0 record(s)
2017-06-17	21:17:43	update	Finished Step CAMPAING use_cards2

ภาพที่ 30 ตัวอย่าง Output log ใน view component จากภาพที่ 29

4.3.4 ตัวอย่างการเขียนสคริปต์อีทีแอลดีเอสแอลเพื่อแก้ไขข้อมูลในตาราง DW_CAMPAIGN_FACT โดยการปรับให้อยู่ในรูปแบบปัจจุบันและนำเข้าข้อมูลในส่วนที่เดิมไม่เคยปรากฏในตาราง และ สคริปต์อีทีแอลจาวาในภาพที่ 32 ที่ได้จาก สคริปต์อีทีแอลดีเอสแอลในภาพที่ 31

```

STEP NAME|CAMPAING use_cards3|
ACTION|update|
TARGET TABLE|DW_CAMPAIGN_FACT (update and insert)|
CONDITION|"DW_SALE_ATTRIBUTE.saleheaderid = DW_SALE_HEADER_FACT.sale_id
and DW_SALE_HEADER_FACT.STORE_SR_KEY = DW_STORE_DIM.STORE_ID and
DW_SALE_ATTRIBUTE.ATTRIBUTE056 = '2' group by

```

```

dw_sale_header_fact.dt_sr_key, dw_sale_header_fact.channel_sr_key,
dw_sale_header_fact.store_sr_key, dw_store_dim.store_nm,
DW_SALE_ATTRIBUTE.attribute046, DW_SALE_ATTRIBUTE.saleheaderid,
DW_SALE_ATTRIBUTE.ATTRIBUTE054, DW_SALE_ATTRIBUTE.ATTRIBUTE055,
DW_SALE_ATTRIBUTE.ATTRIBUTE057"
COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|SO
URCE_FIELD
VALUES|DT_SR_KEY|key|move direct|DW_SALE_HEADER_FACT|dt_sr_key
VALUES|CHANNEL_SR_KEY|key|move
direct|DW_SALE_HEADER_FACT|channel_sr_key
VALUES|STORE_SR_KEY|key|move direct|DW_SALE_HEADER_FACT|store_sr_key
VALUES|STORE_NM||move direct|DW_STORE_DIM|store_nm
VALUES|CARD_NUMBER||move direct|DW_SALE_ATTRIBUTE|attribute046
VALUES|SALE_CARD_SALEID||move direct|DW_SALE_ATTRIBUTE|saleheaderid
VALUES|USE_CARD_SALEID|||
VALUES|SALE_CARD_QTY||0|
VALUES|USE_CARD_QTY||nvl(sum(attribute043),0)|
VALUES|TRUN_TYPE||###USE###|
VALUES|DATA_LOAD_DT||sysdate|
VALUES|DATA_UPDT_DT||sysdate|
VALUES|ATTR||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE054
VALUES|ATTR2||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE055
VALUES|ATTR3||move direct|DW_SALE_ATTRIBUTE|ATTRIBUTE057
VALUES|ATTR4||### ###|
VALUES|ATTR5||### ###|
VALUES|NUM_SALE_CARD_QTY||0|
VALUES|NUM_USE_CARD_QTY||1|

```

ภาพที่ 31 ตัวอย่างสคริปต์อีทีแอลดีเอสแอลเพื่อแก้ไขข้อมูลในโดยการปรับให้อยู่ในรูปปัจจุบันและ
นำเข้าข้อมูลในส่วนที่เดิมไม่เคยปรากฏในตาราง

```

logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("upsert"+tab).substring(0,15);
System.out.println(logDate+logTime+logType+"Starting Step CAMPAING
use_cards3");
System.out.println(logDate+logTime+logType+"Update data on
DW_CAMPAIGN_FACT");
logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("upsert"+tab).substring(0,15);
System.out.println(logDate+logTime+logType+"Loading Mapping Table");
etl.etlDelete("MAPPING_VALUE_TABLE");
result = etl.etlInsertMapTable("VALUES','DT_SR_KEY','key','move
direct','DW_SALE_HEADER_FACT','dt_sr_key");
if(result < 0){
    logTime = (dtf.format(now).split(" ")[1])+datetab;
    logDate = (dtf.format(now).split(" ")[0])+datetab;
    logType = ("error"+tab).substring(0,15);
    System.err.println(logDate+logTime+logType+"Error During Loading
Mapping Table");
    System.err.println(logDate+logTime+logType+"VALUES','DT_SR_KEY','key','"+
transform+"','DW_SALE_HEADER_FACT','dt_sr_key");
    System.exit(-1);
}
...
result = etl.etlInsertMapTable("VALUES','NUM_USE_CARD_QTY','1','");
if(result < 0){
    logTime = (dtf.format(now).split(" ")[1])+datetab;
    logDate = (dtf.format(now).split(" ")[0])+datetab;
    logType = ("error"+tab).substring(0,15);
    System.err.println(logDate+logTime+logType+"Error During Loading

```

```

Mapping Table");
        System.err.println(logDate+logTime+logType+"VALUES','NUM_USE_CARD_Q
TY','"+transform+"','");
        System.exit(-1);
    }
logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("upsert"+tab).substring(0,15);
System.out.println(logDate+logTime+logType+"Starting Update data to
DW_CAMPAIGN_FACT");
result = etl.etlUpdateData("DW_CAMPAIGN_FACT",
"DW_SALE_ATTRIBUTE.saleheaderid = DW_SALE_HEADER_FACT.sale_id and
DW_SALE_HEADER_FACT.STORE_SR_KEY = DW_STORE_DIM.STORE_ID and
DW_SALE_ATTRIBUTE.ATTRIBUTE056 = '2' group by dw_sale_header_fact.dt_sr_key
        , dw_sale_header_fact.channel_sr_key
        ,
dw_sale_header_fact.store_sr_key, dw_store_dim.store_nm,
DW_SALE_ATTRIBUTE.attribute046, DW_SALE_ATTRIBUTE.saleheaderid
        ,
DW_SALE_ATTRIBUTE.ATTRIBUTE054, DW_SALE_ATTRIBUTE.ATTRIBUTE055 ,
DW_SALE_ATTRIBUTE.ATTRIBUTE057", "update and insert");
if(result < 0){
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("error"+tab).substring(0,15);
        System.err.println(logDate+logTime+logType+"Error During Update data to
DW_CAMPAIGN_FACT");
        System.exit(-1);
    }
logTime = (dtf.format(now).split(" ")[1])+datetab;
logDate = (dtf.format(now).split(" ")[0])+datetab;
logType = ("upsert"+tab).substring(0,15);
System.out.println(logDate+logTime+logType+"Updated and Inserted data done

```



```
" + result + " record(s)");
System.out.println(logDate + logTime + logType + "Finished Step CAMPAING
use_cards3");
```

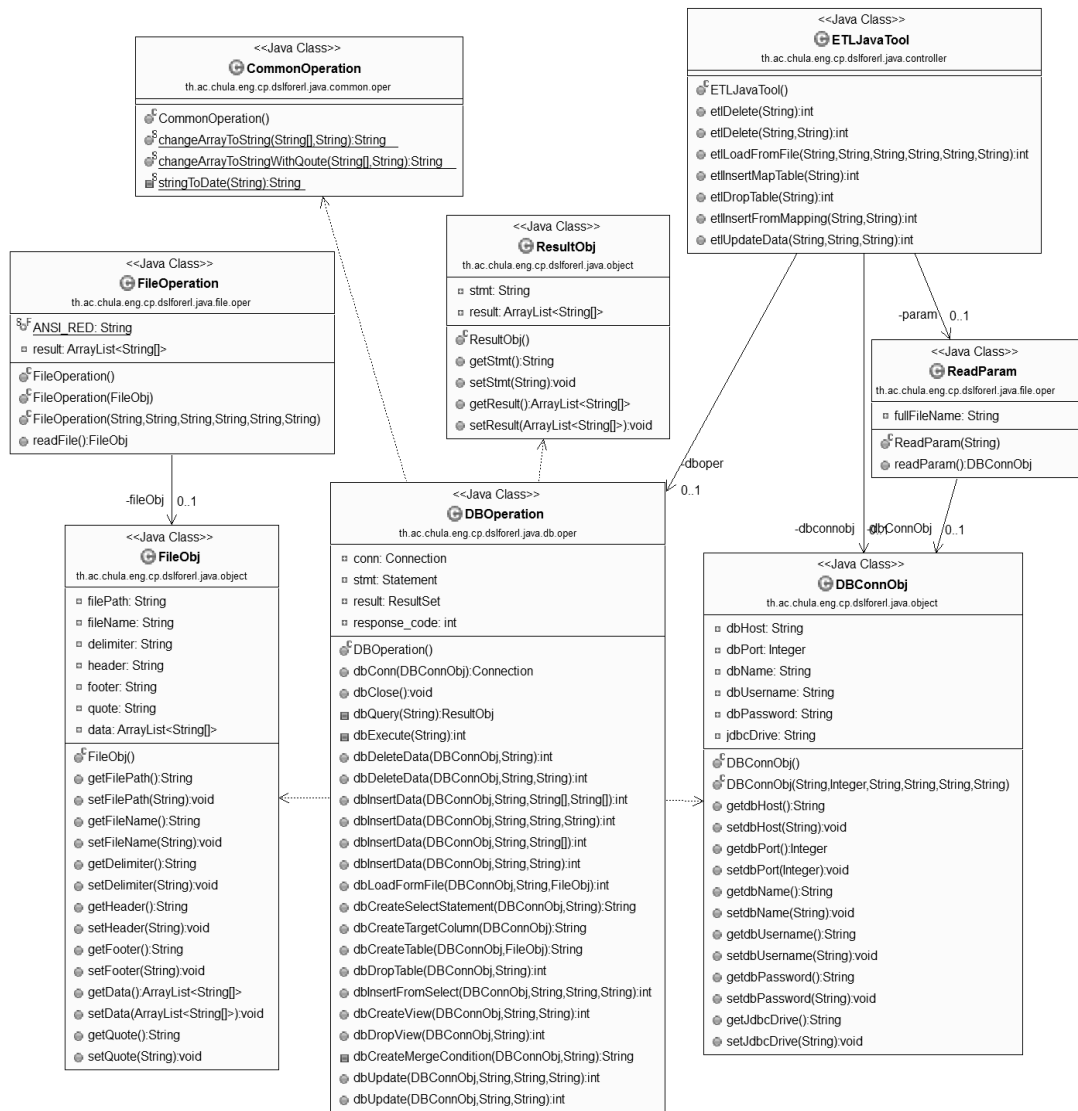
ภาพที่ 32 ตัวอย่างจาวาที่ได้จากสคริปต์อีทีแอลดีเอสแอลเพื่อแก้ไขข้อมูลในโดยการปรับให้อยู่ในรูปแบบปัจจุบันและนำเข้าข้อมูลในส่วนที่เดิมไม่เคยปรากฏในตารางในภาพที่ 31

2017-06-17	21:17:43	upsert	Starting Step CAMPAING use_cards3
2017-06-17	21:17:43	upsert	Update data on DW_CAMPAIGN_FACT
2017-06-17	21:17:43	upsert	Loading Mapping Table
2017-06-17	21:17:43	upsert	Starting Update data to DW_CAMPAIGN_FACT
2017-06-17	21:17:43	upsert	Updated and Inserted data done 0 record(s)
2017-06-17	21:17:43	upsert	Finished Step CAMPAING use_cards3

ภาพที่ 33 ตัวอย่าง Ouput log ใน view component จากภาพที่ 32

4.5 อีทีแอลหลังบ้าน (ETL Back-end)

หลังจากที่สคริปต์อีทีแอลดีเอสแอลได้แปลงเป็นสคริปต์จาวาแล้วนั้นสคริปต์จะไปเรียกเครื่องมืออีทีแอลหลังบ้านช่วยในการประมวลผลซึ่งประกอบด้วย ส่วนตัวควบคุม (Controller component) ประกอบด้วยตัวควบคุมอีทีแอล (ETL controller), ตัวควบคุมไฟล์ (File controller), ตัวควบคุมการแปลง (Transformation Operation controller), และ ตัวควบคุมการดำเนินการฐานข้อมูล (Database Operation controller) และ จาวาโมเดล (Java model) คือ การรับส่งคำสั่งระหว่างตัวควบคุม และส่วนแสดงผล ในภาพที่34



ภาพที่ 34 แผนภาพยูเอ็มแอลอีทีแอลหลังบ้าน

4.5.1 ตัวควบคุมอีทีแอล คือจาวาคลาสสำหรับกระบวนการทำงานของอีทีแอลโดยที่จาวาที่ได้จากสคริปต์อีทีแอลดีเอสแอลจะเรียกใช้จาวาคลาสในภาพที่35

```

public class ETLJavaTool {
    private DBConnObj dbconnobj;
    private ReadParam param;
    private DBOperation dboper;
    //ส่วนการสร้าง Object หลักของคลาส
    public ETLJavaTool(){
        ...
    }
}

```

```

//Method สำหรับการลบข้อมูลแบบไร้เงื่อนไข
public int etlDelete(String targetTable){
    ... }

//Method สำหรับการลบข้อมูลแบบมีเงื่อนไข
public int etlDelete(String targetTable, String condition){
    ... }

//Method สำหรับการนำเข้าข้อมูลจากไฟล์
public int etlLoadFromFile(String filePath, String fileName, String delimiter,
String header, String footer, String quote ){
    ... }

//Method สำหรับการนำเข้าข้อมูลลงตาราง Mapping Value
public int etlInsertMapTable(String columnName){
    ... }

//Method สำหรับการลบตารางออกจากฐานข้อมูล
public int etlDropTable(String tableName){
    ... }

//Method สำหรับการนำเข้าข้อมูลตามเงื่อนไขที่กำหนดในตาราง Mapping Value
public int etlInsertFromMapping(String targetTable, String condition){
    ... }

//Method สำหรับการแปลงข้อมูลในตารางตามเงื่อนไขที่ระบุ
public int etlUpdateData(String targetTable, String condition, String
updateType){
    ... }
}

```

ภาพที่ 35 ตัวอย่าง Method ตัวควบคุมอีทีแอล

4.5.2 ตัวควบคุมไฟล์ คือจาวาคลาสที่ใช้สำหรับการบริหารจัดการไฟล์เพื่อนำไปประมวลผลต่อในกระบวนการอีทีแอลดังภาพที่36และ ดังภาพที่37

```

public class FileOperation {
    public static final String ANSI_RED = "\u001B[m";
    private FileObj fileObj;
}

```

```

private ArrayList<String[]> result;

//ส่วนการสร้าง Object หลักของคลาสกรณีไม่มีตัวแปรนำเข้า
public FileOperation (){
    ... }

//ส่วนการสร้าง Object หลักของคลาสในกรณีมีตัวแปรนำเข้าเป็น File Object
public FileOperation (FileObj fileobj){
    ... }

//ส่วนการสร้าง Object หลักของคลาสในกรณีมีตัวแปรนำเข้าเป็นข้อมูลของไฟล์ข้อมูล
public FileOperation (String filepath, String filename, String delimiter, String
header, String footer, String quote){
    ... }

//Method สำหรับการอ่านไฟล์ข้อมูล
public FileObj readFile(){
    ... }
}

```

ภาพที่ 36 ตัวอย่าง Method ตัวควบคุมไฟล์ของคลาสไฟล์โอเปอเรชัน

```

public class ReadParam {
    private String fullFileName;
    private DBConnObj dbConnObj;
    //ส่วนการสร้าง Object หลักของคลาส
    public ReadParam(String fileName){
        ... }

    //Method สำหรับอ่านตัวแปรฐานข้อมูลจากไฟล์
    public DBConnObj readParam(){
        ... }
}

```

ภาพที่ 37 ตัวอย่าง Method ตัวควบคุมไฟล์ในคลาสอ่านพารามิเตอร์

4.5.3 ตัวควบคุมการแปลง คือ จาวาคลาสที่ใช้สำหรับการแปลงข้อมูลให้อยู่ในรูปของอ็อบเจกต์เพื่อนำไปประมวลผลต่อในกระบวนการอีทีแอลดังภาพที่38

```
public class CommonOperation {
    //Method สำหรับการเปลี่ยนข้อมูลใน Array เป็นตัวอักษรโดยคั่นแต่ละข้อมูลใน Array
    ตามตัวคั่นที่กำหนดแบบมี Qoute
    public static String changeArrayToString(String[] srcArray, String delimiter){
        ... }
    //Method สำหรับการเปลี่ยนข้อมูลใน Array เป็นตัวอักษรโดยคั่นแต่ละข้อมูลใน Array
    ตามตัวคั่นที่กำหนดแบบไม่มี Qoute
    public static String changeArrayToStringWithQoute(String[] srcArray, String
    delimiter){
        ... }
    //Method สำหรับการเปลี่ยนตัวอักษรเป็นตัวอักษรในรูปแบบวันที่
    private static String stringToDate(String text){
        ... }
}
```

ภาพที่ 38 ตัวอย่าง Method ตัวควบคุมการแปลง

4.5.4 ตัวควบคุมการดำเนินการฐานข้อมูล คือ จาวาคลาสที่เป็นส่วนในการติดต่อกับฐานข้อมูลและประมวลผลดังภาพที่39

```
public class DBOperation {
    private Connection conn;
    private Statement stmt;
    private ResultSet result;
    private int response_code;
    //ส่วนการสร้าง Object หลักของคลาส
    public DBOperation(){
        ... }
    //Method สำหรับติดต่อกับฐานข้อมูล
    public Connection dbConn(DBConnObj obj){
```

```

        ... }

//Method สำหรับการติดต่อกับฐานข้อมูล
public void dbClose(){
    ... }

//Method สำหรับการดึงข้อมูลจากฐานข้อมูล
private ResultObj dbQuery(String query){
    ... }

//Method สำหรับการประมวลผลอื่นๆที่นอกเหนือจากการดึงข้อมูลจากฐานข้อมูล
private int dbExecute(String stmt){
    ... }

//Method สำหรับการลบข้อมูลจากฐานข้อมูลแบบไร้เงื่อนไข
public int dbDeleteData(DBConnObj dbconnobj, String tableName){
    ... }

//Method สำหรับการลบข้อมูลจากฐานข้อมูลแบบมีเงื่อนไข
public int dbDeleteData(DBConnObj dbconnobj, String tableName, String
whereStmt){
    ... }

//Method สำหรับการนำเข้าข้อมูล
public int dbInsertData(DBConnObj dbconnobj, String targetTable, String[]
columnName, String[] columnValue){
    ... }

//Method สำหรับการนำเข้าข้อมูล
public int dbInsertData(DBConnObj dbconnobj, String targetTable, String
columnName, String columnValue){
    ... }

//Method สำหรับการนำเข้าข้อมูล
public int dbInsertData(DBConnObj dbconnobj, String targetTable, String

```

```

columnValue){
    ... }

    //Method สำหรับการนำเข้าข้อมูลจากไฟล์
    public int dbLoadFormFile(DBConnObj dbconnobj, String targetTable,
FileObj fileObj){
    ... }

    //Method สำหรับการสร้าง Select Statement
    public String dbCreateSelectStatement(DBConnObj dbconnobj, String
condition){
    ... }

    //Method สำหรับการสร้างตารางใหม่
    public String dbCreateTargetColumn(DBConnObj dbconnobj){
    ... }

    //Method สำหรับการลบตารางในฐานข้อมูล
    public int dbDropTable(DBConnObj dbconnobj, String tableName){
    ... }

    //Method สำหรับการนำเข้าข้อมูลจาก Select Statement
    public int dbInsertFromSelect(DBConnObj dbconnobj, String targetTable,
String targetCol, String selStmt){
    ... }

    //Method สำหรับการสร้าง View ในฐานข้อมูล
    public int dbCreateView(DBConnObj dbconnobj, String targetView, String
selStmt){
    ... }

    //Method สำหรับการลบ View ในฐานข้อมูล
    public int dbDropView(DBConnObj dbconnobj, String viewName){
    ... }

    //Method สำหรับการสร้าง Merge Condition Statement
    private String dbCreateMergeCondition (DBConnObj dbconnobj, String
condition){
    ... }

```

```

//Method สำหรับการแปลงข้อมูลในตาราง
public int dbUpdate(DBConnObj dbconnobj, String targetTable, String
sourceView, String updateType){
    ... }
//Method สำหรับการแปลงข้อมูลในตาราง
public int dbUpdate(DBConnObj dbconnobj, String targetTable, String
condition){
    ... }
}

```

ภาพที่ 39 ตัวอย่าง Method ตัวควบคุมการดำเนินการฐานข้อมูล

4.5.5 จาวาโมเดล คือจาวาคลาสสำหรับการรับส่งคำสั่งระหว่างตัวควบคุม และส่วนแสดงผลดังภาพที่ 40-42

```

public class DBConnObj {
    private String dbHost;
    private Integer dbPort;
    private String dbName;
    private String dbUsername;
    private String dbPassword;
    private String jdbcDrive;
    public DBConnObj (){
        this.dbHost = null;
        this.dbPort = null;
        this.dbName = null;
        this.dbUsername = null;
        this.dbPassword = null;
        this.jdbcDrive = null;
    }
    public DBConnObj (String dbHost, Integer dbPort, String dbName, String
dbUsername, String dbPassword, String jdbcDrive){
        this.dbHost = dbHost;

```



```

        this.dbPort = dbPort;
        this.dbName = dbName;
        this.dbUsername = dbUsername;
        this.dbPassword = dbPassword;
        this.jdbcDrive = jdbcDrive;
    }
    //ประกอบด้วย Method Set และ Get
}

```

ภาพที่ 40 ตัวอย่าง Method ของจาวาโมเดลสำหรับการเชื่อมต่อกับฐานข้อมูล

```

public class FileObj {
    private String filePath;
    private String fileName;
    private String delimiter;
    private String header;
    private String footer;
    private String quote;
    private ArrayList<String[]> data;
    //ประกอบด้วย Method Set และ Get
}

```

ภาพที่ 41 ตัวอย่าง Method ของจาวาโมเดลสำหรับไฟล์นำเข้าข้อมูล

```

public class ResultObj {
    private String stmt;
    private ArrayList<String[]> result;
    //ประกอบด้วย Method Set และ Get
}

```

ภาพที่ 42 ตัวอย่าง Method ของจาวาโมเดลสำหรับส่วนของการแสดงผล

บทที่ 5

การประเมินและการวัดผล

การประเมินและวัดผลทำโดยการออกแบบสำรวจความพึงพอใจ ให้กับผู้ทำงานที่เกี่ยวข้องกับด้าน อีทีแอลวัดผลความพึงพอใจหลังจากที่ได้ทดลองใช้ ประกอบด้วยการประเมินใน 5 มิติ คุณภาพ (Quality Dimension)

1. ความสามารถในการใช้ปฏิบัติงานได้จริง (Operability) ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอลสะดวกแก่การนำมาใช้ในการปฏิบัติงานดีกว่าหรือเทียบเท่าเครื่องมืออีทีแอลในปัจจุบัน
2. ความง่ายในการเรียนรู้ (Learnability) ชุดคำสั่งภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอลสามารถจดจำเรียนรู้ได้ง่าย
3. ความถูกต้องของผลลัพธ์ (Correctness) ผลลัพธ์ที่ได้มีความถูกต้องเหมาะสมแก่การนำมาใช้งานในทางปฏิบัติ
4. ตอบสนองความต้องการใช้งาน(Fitness for Use) ภาษาจำเพาะโดเมนสามารถตอบสนองความต้องการใช้งานสำหรับกระบวนการอีทีแอล
5. ความพึงพอใจโดยภาพรวม (Overall Satisfaction) ความพึงพอใจโดยภาพรวมในการใช้ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอล

การทดสอบที่ 1 เขียนสคริปต์ดีเอสแอลอีทีแอลผ่านทางexcel

ผู้ที่ไม่มีความรู้พื้นฐานทางด้านการเขียนโปรแกรมสามารถเปลี่ยนเป็นสคริปต์ดีเอสแอลอีทีแอลได้โดยบันทึกไฟล์เอกสารต้นแบบ excel ในภาพที่ 43 และไฟล์เอกสารดังกล่าวจะแปลงเป็นสคริปต์ดีเอสแอลอีทีแอลอัตโนมัติตามภาพที่ 44 ตัวอย่างของกระบวนการลบข้อมูล (delete) และการแทรกเพิ่มข้อมูล (insert) เข้าสู่ตาราง CAMPAIGN_FACT และการปรับข้อมูลให้เป็นปัจจุบัน (update) ด้วย ซึ่งการแปลงข้อมูลแบ่งได้ 4 แบบ

1. ค่า null หรือการนำเข้าข้อมูลแบบมีเงื่อนไข (Constraint) เช่น move direct, null เป็นต้น
2. การเรียกใช้ฟังก์ชันการทำงานของฐานข้อมูล (Function) เช่น nvl(), sum() เป็นต้น
3. ค่าคงที่ตัวเลข (Number) เช่น 0, 1 เป็นต้น
4. ค่าคงที่ตัวอักษร (String) เช่น 'USE' เป็นต้น

STEP NAME	Delete CAMPAIGN				
ACTION	delete				
TARGET TABLE	CAMPAIGN_FACT				
CONDITION	"(STORE,DT_KEY) in (select l.STORE,l.DT_KEY from campaign_fact c, STG_LOG l where c.STORE = l.STORE and c.DT_KEY = l.DT_KEY) "				
STEP NAME	CAMPAIGN use_cards				
ACTION	insert				
TARGET TABLE	CAMPAIGN_FACT (update and insert)				
CONDITION	"sale_attribute.saleheaderid = sales_header_fct.sale_id and sales_header_fct.store = store_dim.store and sale_attribute.att056 = '2' group by sale_attribute.saleheaderid, to_char(to_date(substr(sale_attribute.att040, 1, 10),yyyy-mm-dd'),yyyymmdd'), sales_header_fct.dt_key, sales_header_fct.store, store_dim.store_nm, sales_header_fct.channel_sr_key, sale_attribute.att039,sale_attribute.ATT054,sale_attribute.ATT055 , sale_attribute.ATT057,sale_attribute.ATT046 "				
COLUMNS	TARGET_FIELD	TARGET_KEY	DATA_TRANSFORMATION	SOURCE_TABLE	SOURCE_FIELD
VALUES	DT_KEY		move direct	sales_header_fct	dt_key
VALUES	CHANNEL_SR_KEY		move direct	sales_header_fct	channel_sr_key
VALUES	STORE	key	move direct	sales_header_fct	store
VALUES	STORE_NM		move direct	store_dim	store_nm
VALUES	CARD_NUMBER	key	move direct	sale_attribute	att046
VALUES	SALE_CARD_SALEID		null		
VALUES	USE_CARD_SALEID	key	move direct	sale_attribute	saleheaderid
VALUES	SALE_CARD_QTY		0		
VALUES	USE_CARD_QTY		nvl(sum(att043),0)		
VALUES	TRUN_TYPE		###USE###		
VALUES	DATA_LOAD_DT		sysdate		
VALUES	DATA_UPDT_DT		sysdate		
VALUES	ATTR		move direct	sale_attribute	ATT054
VALUES	ATTR2		move direct	sale_attribute	ATT055
VALUES	ATTR3		move direct	sale_attribute	ATT057
VALUES	ATTR4		### ###		
VALUES	ATTR5		### ###		
VALUES	NUM_SALE_CARD_QTY		0		
VALUES	NUM_USE_CARD_QTY		1		

ภาพที่ 43 ตัวอย่างรูปแบบของไฟล์เอกสาร Excel

```

STEP NAME|Delete CAMPAIGN||||
ACTION|delete||||
TARGET TABLE|CAMPAIGN_FACT||||
CONDITION|"(STORE,DT_KEY) in (select l.STORE,l.DT_KEY from campaign_fact c,
STG_LOG l where c.STORE = l.STORE and c.DT_KEY = l.DT_KEY) "||||

STEP NAME|CAMPAIGN use_cards||||
ACTION|insert||||
TARGET TABLE|CAMPAIGN_FACT (update and insert)||||
CONDITION|"sale_attribute.saleheaderid = sales_header_fct.sale_id and
sales_header_fct.store = store_dim.store and sale_attribute.att056 = '2' group
by sale_attribute.saleheaderid, to_char(to_date(substr(sale_attribute.att040, 1,
10),'yyy-mm-dd'),'yyyymmdd'), sales_header_fct.dt_key,
sales_header_fct.store, store_dim.store_nm, sales_header_fct.channel_sr_key,
sale_attribute.att039,sale_attribute.ATT054,sale_attribute.ATT055 ,
sale_attribute.ATT057,sale_attribute.ATT046 "||||
COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|SO
URCE_FIELD
VALUES|DT_KEY||move direct|sales_header_fct|dt_key
VALUES|CHANNEL_SR_KEY||move direct|sales_header_fct|channel_sr_key
VALUES|STORE|key|move direct|sales_header_fct|store
VALUES|STORE_NM||move direct|store_dim|store_nm
VALUES|CARD_NUMBER|key|move direct|sale_attribute|att046
VALUES|SALE_CARD_SALEID||null||
VALUES|USE_CARD_SALEID|key|move direct|sale_attribute|saleheaderid
VALUES|SALE_CARD_QTY||0||
VALUES|USE_CARD_QTY||nvl(sum(att043),0)||
VALUES|TRUN_TYPE|###USE###|
VALUES|DATA_LOAD_DT||sysdate||
VALUES|DATA_UPDT_DT||sysdate||
VALUES|ATTR||move direct|sale_attribute|ATT054

```

```
VALUES|ATTR2||move direct|sale_attribute|ATT055
VALUES|ATTR3||move direct|sale_attribute|ATT057
VALUES|ATTR4||### ###|
VALUES|ATTR5||### ###|
VALUES|NUM_SALE_CARD_QTY||0|
VALUES|NUM_USE_CARD_QTY||1|
```

ภาพที่ 44 สคริปต์ดีเอสแอลอีทีแอลจาก ภาพที่43

เมื่อภาพที่ 44 ทำการประมวลจะนำแหล่งข้อมูลต้นทางในภาพที่ 45 – ภาพที่ 50 มาเชื่อมโยงกันตามสคริปต์ดีเอสแอลอีทีแอล จะประกอบด้วยสองขั้นตอนคือ Delete CAMPAIGN และ CAMPAIGN use_cards

- **Delete CAMPAIGN** จะทำการลบข้อมูลในตารางข้อมูล CAMPAIGN_FACT โดยจะลบข้อมูลที่ตรงกับเงื่อนไข inner join ระหว่างตารางข้อมูล CAMPAIGN_FACT และ STG_LOG เมื่อ CAMPAIGN_FACT.STORE = STG_LOG.STORE and CAMPAIGN_FACT.DT_KEY = STG_LOG.DT_KEY

- **CAMPAIGN use_cards** จะทำการนำเข้าข้อมูลและการปรับข้อมูลให้เป็นปัจจุบัน ที่ตารางข้อมูล CAMPAIGN_FACT โดยมีเงื่อนไขของตารางข้อมูลของ SALE_ATTRIBUTE, SALES_HEADER_FCT และ STORE_DIM โดยมีเงื่อนไข SALE_ATTRIBUTE.SALEHEADERID = SALES_HEADER_FCT.SALE_ID AND SALES_HEADER_FCT.STORE = STORE_DIM.STORE AND SALE_ATTRIBUTE.ATT056 = '2'

STORE	BRAND_SR_KEY	LOC_SR_KEY	COMPANY_SR_KEY	MONTH_ID	STORE_ID	STORE_NM	STORE_THAINM	STORE_ADD
1	1001	3	1001	101 Mar-2017	1001	Bang Bon	บางบอน	บางบอน
2	1002	3	1002	101 Mar-2017	1002	Bang Kapi	บางกะปิ	บางกะปิ
3	1003	3	1003	101 Mar-2017	1003	Bang Khae	บางเขน	บางเขน
4	1004	3	1004	101 Mar-2017	1004	Chatuchak	จตุจักร	จตุจักร
5	1005	3	1005	101 Mar-2017	1005	Din Daeng	ดินแดง	ดินแดง
6	1006	3	1006	101 Mar-2017	1006	Dusit	ดุสิต	ดุสิต
7	1007	3	1007	101 Mar-2017	1007	Pathum Wan	ปทุมวัน	ปทุมวัน
8	1008	3	1008	101 Mar-2017	1008	Phasi Charoen	ภาษีเจริญ	ภาษีเจริญ
9	1009	3	1009	101 Mar-2017	1009	Taling Chan	ตลิ่งชัน	ตลิ่งชัน
10	1010	3	1010	101 Mar-2017	1010	Thon Buri	ธนบุรี	ธนบุรี

ภาพที่ 45 ตารางข้อมูลSTORE_DIM

SALE_ID	DT_KEY	STORE	CHANNEL_SR_KEY	DAYPART_SR_KEY	CUST_SR_KEY	ORDERNUM	INV_NUM	TOTAL_SALE	TOTAL_NET_SALE	TOTAL_DEPOSIT	TOTAL_OTHER_SALE	TOTAL_DISC
1	7001 20170301	1006	1	2010301	4	1	1	520	520	0	0	250
2	7002 20170301	1006	2	2010301	4	1	1	645	645	0	0	250
3	7003 20170301	1007	1	2010301	4	1	1	840	840	0	0	250
4	7004 20170301	1004	1	2010301	4	1	1	640	640	0	0	250
5	7005 20170301	1008	2	2010301	4	1	1	635	635	0	0	250
6	7006 20170301	1002	1	2010301	4	1	1	865	865	0	0	250
7	7007 20170301	1004	1	2010301	4	1	1	520	520	0	0	250
8	7008 20170301	1009	2	2010301	4	1	1	635	635	0	0	250
9	7009 20170301	1010	1	2010301	4	1	1	856	856	0	0	250
10	7010 20170301	1001	1	2010301	4	1	1	635	635	0	0	250

ภาพที่ 46 ตารางข้อมูลSALES_HEADER_FCT

SALEHEADERID	ATT039	ATT040	ATT043	ATT054	ATT055	ATT056	DATASOURCETYPEID	DATA_LOAD_DT	DATA_UPDT_DT	
1	7001	2	2017-03-01	1	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
2	7002	1	2017-03-01	2	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
3	7003	2	2017-03-01	3	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
4	7004	1	2017-03-01	4	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
5	7005	2	2017-03-01	5	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
6	7006	1	2017-03-01	6	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
7	7007	2	2017-03-01	6	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
8	7008	2	2017-03-01	6	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
9	7009	2	2017-03-01	6	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017
10	7010	2	2017-03-01	6	THB	Bangkok	2	Thailand	01 มี.ค. 2017	01 มี.ค. 2017

ภาพที่ 47 ตารางข้อมูลSALE_ATTRIBUTE

STORE	DT_KEY	
1	1001	20170301
2	1002	20170301
3	1003	20170301
4	1004	20170301
5	1005	20170301

ภาพที่ 48 ตารางข้อมูลSTG_LOG

DT_KEY	CHANNEL_SR_KEY	STORE	STORE_NM	CARD_NUMBER	SALE_CARD_SALEID	USE_CARD_SALEID	SALE_CARD_QTY	USE_CARD_QTY	TRUN_TYPE	DATA_LOAD_DT
1	20170301	1	1001 Bang Bon	110011	(null)	110011	(null)	1	USE	01 มี.ค. 2017
2	20170301	2	1002 Bang Kapi	110241	(null)	110241	(null)	1	USE	01 มี.ค. 2017
3	20170301	1	1003 Bang Khae	110324	(null)	110324	(null)	1	USE	01 มี.ค. 2017
4	20170301	1	1004 Chatuchak	110745	(null)	110745	(null)	1	USE	01 มี.ค. 2017
5	20170301	2	1005 Din Daeng	110142	(null)	110142	(null)	1	USE	01 มี.ค. 2017
6	20170301	1	1006 Dusit	110435	(null)	110435	(null)	1	USE	01 มี.ค. 2017

ภาพที่ 49 ตารางข้อมูลCAMPAIGN_FACT-1

STORE	STORE_NM	CARD_NUMBER	DATA_UPDT_DT	ATTR	ATTR2	ATTR3	ATTR4	ATTR5	NUM_SALE_CARD_QTY	NUM_USE_CARD_QTY
1001	Bang Bon	110011	01 มี.ค. 2017	(null)	(null)	(null)	(null)	(null)	(null)	1
1002	Bang Kapi	110241	01 มี.ค. 2017	(null)	(null)	(null)	(null)	(null)	(null)	1
1003	Bang Khae	110324	01 มี.ค. 2017	(null)	(null)	(null)	(null)	(null)	(null)	1
1004	Chatuchak	110745	01 มี.ค. 2017	(null)	(null)	(null)	(null)	(null)	(null)	1
1005	Din Daeng	110142	01 มี.ค. 2017	(null)	(null)	(null)	(null)	(null)	(null)	1
1006	Dusit	110435	01 มี.ค. 2017	(null)	(null)	(null)	(null)	(null)	(null)	1

ภาพที่ 50 ตารางข้อมูลCAMPAIGN_FACT-2

CHULALONGKORN UNIVERSITY

ภาพที่ 51 และ ภาพที่ 52 คือผลลัพธ์จากสคริปต์ดีเอสแอลอีทีแอลภาพที่ 44 ซึ่งตารางข้อมูล CAMPAIGN_FACT จะถูกทำการลบข้อมูลก่อนในขั้นตอน Delete CAMPAIGN และ ทำการนำเข้าข้อมูลและปรับข้อมูลของขั้นตอน CAMPAIGN use_cards

DT_KEY	CHANNEL_SR_KEY	STORE	STORE_NM	CARD_NUMBER	SALE_CARD_SALEID	USE_CARD_SALEID	SALE_CARD_QTY	USE_CARD_QTY	TRUN_TYPE	DATA_LOAD_DT
1	20170301	1	1006 Dusit	110435	(null)	110435	(null)	1	USE	01 มี.ค. 2017
2	20170301	1	1006 Dusit	635636	(null)	7001	0	1	USE	16 ก.ค. 2017
3	20170301	2	1006 Dusit	735636	(null)	7002	0	2	USE	16 ก.ค. 2017
4	20170301	1	1007 Pathum Wan	756356	(null)	7003	0	3	USE	16 ก.ค. 2017
5	20170301	1	1004 Chatuchak	636346	(null)	7004	0	4	USE	16 ก.ค. 2017
6	20170301	2	1008 Phasi Charoen	543545	(null)	7005	0	5	USE	16 ก.ค. 2017
7	20170301	1	1002 Bang Kapi	635246	(null)	7006	0	6	USE	16 ก.ค. 2017
8	20170301	1	1004 Chatuchak	117463	(null)	7007	0	6	USE	16 ก.ค. 2017
9	20170301	2	1009 Taling Chan	635346	(null)	7008	0	6	USE	16 ก.ค. 2017
10	20170301	1	1010 Thon Buri	462525	(null)	7009	0	6	USE	16 ก.ค. 2017
11	20170301	1	1001 Bang Bon	754543	(null)	7010	0	6	USE	16 ก.ค. 2017

ภาพที่ 51 ผลลัพธ์ของตารางข้อมูลCAMPAIGN_FACT-1

STORE	STORE_NM	CARD_NUMBER	DATA_UPDT_DT	ATTR	ATTR2	ATTR3	ATTR4	ATTR5	NUM_SALE_CARD_QTY	NUM_USE_CARD_QTY
1006	Dusit	110435	01 มี.ค. 2017	(null)	(null)	(null)	(null)	(null)	(null)	1
1006	Dusit	635636	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1006	Dusit	735636	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1007	Pathum Wan	756356	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1004	Chatuchak	636346	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1008	Phasi Charoen	543545	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1002	Bang Kapi	635246	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1004	Chatuchak	117463	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1009	Taling Chan	635346	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1010	Thon Buri	462525	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1
1001	Bang Bon	754543	16 ก.ค. 2017	THB	Bangkok	Thailand			0	1

ภาพที่ 52 ผลลัพธ์ของตารางข้อมูลCAMPAIGN_FACT-2

การทดสอบที่ 2 เขียนสคริปต์ดีเอสแอลอีทีแอลผ่านทางโปรแกรม Eclipse ในเคสการเปลี่ยนข้อมูล
 ในฐานข้อมูลโดยมีเงื่อนไขดังภาพที่ 53

```

STEP NAME|CAMPAING use_cards|
ACTION|update|
TARGET TABLE|DW_CAMPAIGN_FACT (update from value)|
CONDITION|"DT_SR_KEY is null"|
COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|SOURCE_FIELD
VALUES|USE_CARD_SALEID|||
VALUES|SALE_CARD_QTY||0|
VALUES|TRUN_TYPE||###USE###|
VALUES|DATA_LOAD_DT||sysdate|
VALUES|DATA_UPDT_DT||sysdate|
VALUES|ATTR4||### ###|
VALUES|ATTR5||### ###|
VALUES|NUM_SALE_CARD_QTY||0|
VALUES|NUM_USE_CARD_QTY||1|

```

ภาพที่ 53 สคริปต์ดีเอสแอลอีทีแอลในเคสการเปลี่ยนข้อมูลในฐานข้อมูล

สคริปต์ดีเอสแอลอีทีแอลในเคสการเปลี่ยนข้อมูลในฐานข้อมูล ภาพที่ 53 จะทำการประมวลเปลี่ยนข้อมูลในฐานข้อมูลที่แหล่งข้อมูลต้นทางในภาพที่ 54 ที่ตารางข้อมูลCAMPAIGN_FACT ที่เงื่อนไขDT_SR_KEY is null

DT_SR_KEY	CHANNEL...	STORE_S...	STORE_NM	CARD...	SALE_CA...	USE_CAR...	SALE_CA...	USE_CAR...	TRUN_TYPE	DATA_LO...	DATA_UP...
1	(null)	1	1006 Dusit	635636	(null)	7001	0	1	(null)	16 ก.ค. .	16 ก.ค. .
2	(null)	2	1006 Dusit	735636	(null)	7002	0	2	(null)	16 ก.ค. .	16 ก.ค. .
3	(null)	1	1007 Pathum Wan	756356	(null)	7003	0	3	(null)	16 ก.ค. .	16 ก.ค. .
4	(null)	1	1004 Chatuchak	636346	(null)	7004	0	4	(null)	16 ก.ค. .	16 ก.ค. .
5	20170301	2	1008 Phasi Charoen	543545	(null)	7005	0	5	USE	16 ก.ค. .	16 ก.ค. .
6	20170301	1	1002 Bang Kapi	635246	(null)	7006	0	6	USE	16 ก.ค. .	16 ก.ค. .
7	20170301	1	1004 Chatuchak	117463	(null)	7007	0	6	USE	16 ก.ค. .	16 ก.ค. .
8	20170301	2	1009 Taling Chan	635346	(null)	7008	0	6	USE	16 ก.ค. .	16 ก.ค. .
9	20170301	1	1010 Thon Buri	462525	(null)	7009	0	6	USE	16 ก.ค. .	16 ก.ค. .
10	20170301	1	1001 Bang Bon	754543	(null)	7010	0	6	USE	16 ก.ค. .	16 ก.ค. .

ภาพที่ 54 ตารางข้อมูลCAMPAIGN_FACT

หลังจากสคริปต์ดีเอสแอลอีทีแอลภาพที่ 53 ทำการประมวลผลจะได้ผลลัพธ์ดังตารางภาพที่ 55 ซึ่งข้อมูลจะมีการเปลี่ยนแปลงตามสคริปต์ดีเอสแอลอีทีที่กำหนดไว้

DT_SR_KEY	CHANNEL...	STORE_S...	STORE_NM	CARD_NU...	SALE_CA...	USE_CAR...	SALE_CA...	USE_CAR...	TRUN_TYPE	DATA_LO...	DATA_UP...
1	(null)	1	1006 Dusit	635636	(null)	(null)	0	1	USE	16 ก.ค. .	16 ก.ค. .
2	(null)	2	1006 Dusit	735636	(null)	(null)	0	2	USE	16 ก.ค. .	16 ก.ค. .
3	(null)	1	1007 Pathum Wan	756356	(null)	(null)	0	3	USE	16 ก.ค. .	16 ก.ค. .
4	(null)	1	1004 Chatuchak	636346	(null)	(null)	0	4	USE	16 ก.ค. .	16 ก.ค. .
5	20170301	2	1008 Phasi Charoen	543545	(null)	7005	0	5	USE	16 ก.ค. .	16 ก.ค. .
6	20170301	1	1002 Bang Kapi	635246	(null)	7006	0	6	USE	16 ก.ค. .	16 ก.ค. .
7	20170301	1	1004 Chatuchak	117463	(null)	7007	0	6	USE	16 ก.ค. .	16 ก.ค. .
8	20170301	2	1009 Taling Chan	635346	(null)	7008	0	6	USE	16 ก.ค. .	16 ก.ค. .
9	20170301	1	1010 Thon Buri	462525	(null)	7009	0	6	USE	16 ก.ค. .	16 ก.ค. .
10	20170301	1	1001 Bang Bon	754543	(null)	7010	0	6	USE	16 ก.ค. .	16 ก.ค. .

ภาพที่ 55 ผลลัพธ์ของตารางข้อมูลCAMPAIGN_FACT จากสคริปต์ดีเอสแอลอีทีแอลในเคสการเปลี่ยนข้อมูลในฐานข้อมูล

การทดสอบที่ 3 เขียนสคริปต์ดีเอสแอลอีทีแอลในเคสการนำข้อมูลเข้าฐานข้อมูลจากไฟล์ผ่านทางโปรแกรม Eclipse โดยมีเงื่อนไขดังภาพที่ 56

```
STEP NAME|Load From File|||
ACTION|insert|comma|yes|no|no
TARGET TABLE|TABLEA (clear and insert)|||
CONDITION|""|||
FILENAME|testLoad.txt|||
COLUMNS|TARGET_FIELD|TARGET_KEY|DATA_TRANSFORMATION|SOURCE_TABLE|SOURCE_FIELD
VALUES|ATT1||move direct|testLoad|ATTR1
VALUES|ATT2||move direct|testLoad|ATTR2
VALUES|ATT3||move direct|testLoad|ATTR3
```



```
VALUES|COLA|### ##|
VALUES|COLC|move direct|testLoad|ATTR5
```

ภาพที่ 56 สคริปต์ดีเอสแอลอีทีแอลในเคสการนำข้อมูลเข้าฐานข้อมูลจากไฟล์

สคริปต์ดีเอสแอลอีทีแอลในเคสการนำข้อมูลเข้าฐานข้อมูลจากไฟล์ข้อมูลดังภาพที่ 57 หลังจากการประมวลผลจะได้ผลลัพธ์ดังภาพที่ 58

```
*testLoad.txt
ATTR1,ATTR2,ATTR3,ATTR4,ATTR5
a,domain,specific,,language
ETL,extract,transform,,load
data,warehouse,DW,,
```

ภาพที่ 57 ไฟล์ testLoad.txt สำหรับการนำข้อมูลเข้าในตารางข้อมูล

	ATT1	ATT2	ATT3	COLA	COLC
1	a	domain	specific		language
2	data	warehouse	DW		
3	ETL	extract	transform		load

ภาพที่ 58 ตารางข้อมูลTABLEA

การเลือกกลุ่มตัวอย่างการประเมินจะเลือกเฉพาะบุคคลที่เกี่ยวข้องด้านไอทีที่ทำงานเกี่ยวข้องกับอีทีแอล ประกอบด้วยผู้ที่ชำนาญและทำงานเกี่ยวข้องกับงานด้านนี้โดยเฉพาะนักวิเคราะห์ธุรกิจไอที โดยการประเมินจะจัดในลักษณะของกลุ่มที่สนใจ (focus group) 6 คน ทั้งนี้แบบประเมินความพึงพอใจแสดงในภาคผนวก ก โดยระดับการประเมินความพึงพอใจแบ่งเป็น 5 ระดับ:

- 1 ไม่พึงพอใจอย่างมาก
- 2 ไม่พึงพอใจ
- 3 ปานกลาง
- 4 พึงพอใจ
- 5 พึงพอใจอย่างมาก

ผู้ทดลองใช้มีจำนวน 6 คน ประกอบด้วย ผู้จัดการโครงการไอที (project manager), นักวิเคราะห์ธุรกิจไอที (business analysts), นักทดสอบระบบไอที (tester), นักเขียนโปรแกรม (programmer) และ นักตรวจสอบคุณภาพ (QA) โดยเขียนสคริปต์ดีเอสแอลอีทีแอลภาพที่ 44, 53 และ 56 ผลการสำรวจความพึงพอใจของผู้ใช้ในมิติคุณภาพด้านต่างๆ ดังตารางที่ 4 สรุปค่าเฉลี่ยโดยภาพรวมในระดับที่ 4.07 จากคะแนนเต็ม 5

ตารางที่ 4 ผลความพึงพอใจ

	คะแนน					คะแนนโดยเฉลี่ย
	5	4	3	2	1	
ความสามารถในการใช้ปฏิบัติงานได้จริง (Operability)	4	1	1	0	0	4.5
ความง่ายในการเรียนรู้ (Learnability)	1	4	1	0	0	4
ความถูกต้องของผลลัพธ์ (Correctness)	6	0	0	0	0	5
ตอบสนองความต้องการใช้งาน(Fitness for Use)	0	1	5	0	0	3.17
ความพึงพอใจโดยภาพรวม(Overall Satisfaction)	0	4	2	0	0	3.67
ค่าเฉลี่ยโดยภาพรวม						4.07



บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลการวิจัย

งานวิจัยนี้พัฒนาดีเอสแอลเพื่อสร้างอีทีแอลสคริปต์ที่มีการออกแบบอีทีแอลดีเอสแอลพัฒนาเพื่อให้ใช้งานง่ายสำหรับผู้ที่ไม่มีความรู้พื้นฐานการเขียนโปรแกรม เช่น นักวิเคราะห์ธุรกิจส่วนของไอที จากการทดสอบผู้ทดลองใช้มีจำนวน 6 คน คือผู้จัดการโครงการไอที, นักวิเคราะห์ธุรกิจไอที, นักทดสอบระบบไอที, นักเขียนโปรแกรม และ นักตรวจสอบคุณภาพ ลงความเห็นว่าเป็นสามารถใช้งานง่ายโดยที่ไม่จำเป็นต้องเรียนรู้เอสคิวแอลและได้ผลลัพธ์ตามที่ต้องการเบื้องต้น และสามารถประยุกต์ใช้กับเอกสารที่ใช้ในการทำงานในปัจจุบันได้ ทำให้นักวิเคราะห์ธุรกิจไอทีสามารถดูตัวอย่างข้อมูลได้สะดวก และได้ผลลัพธ์ที่ถูกต้องในการทดลอง

6.2 ข้อจำกัด

- 1.งานวิจัยนี้ฟังก์ชันการใช้งานไม่หลากหลายเทียบเท่าเครื่องอีทีแอลในปัจจุบัน
- 2.ไม่สามารถตรวจสอบข้อผิดพลาดขณะที่เขียนดีเอสแอลแกรมมา
- 3.สามารถติดตั้งได้เฉพาะระบบปฏิบัติการวินโดวส์
- 4.ยังไม่รองรับดีเอสแอลแกรมมาในรูปแบบภาษาไทย
- 5.การทำงานเน้นคำสั่งในกลุ่ม Data Manipulation Language (DML) ยังขาดการทำงานของคำสั่งในกลุ่ม Data Definition Language (DDL) และชุดคำสั่งที่ซับซ้อน ได้แก่ nested SELECT ปัจจุบันระบบสนับสนุนการทำงานของคำสั่ง SQL ดังแสดงในภาคผนวก ข ตารางที่ 5

6.3 แนวทางการวิจัยในอนาคต

- 1.เพิ่มกลุ่มคำสั่งที่ซับซ้อนขึ้นเพื่อสามารถนำไปใช้ในชีวิตประจำวันได้ง่ายขึ้นและมีข้อจำกัดที่ลดลง
- 2.ปรับปรุงพัฒนาให้ใช้งานง่ายขึ้นในระดับของผู้ใช้โดยทั่วไปที่ไม่จำเป็นต้องรู้เรื่องด้านไอที

รายการอ้างอิง

- [1] D. Ghosh, *DSLs in Action*. Manning Publications, 2011.
- [2] N. Mali and S. Bojewar, "A Survey of ETL Tools," presented at the International Journal of Computer Techniques, 2015.
- [3] J. C. Franchitti, Relational Algebra, Relational Calculus, and SQL: New York University Computer Science Department Courant Institute of Mathematical Sciences. [Online]. Available.
- [4] B. Oliveira and O. Belo, "On the specification of extract, transform, and load patterns behavior: A domain-specific language approach," 2016.
- [5] W. Deneke, "A Domain Specific Model for Generating ETL Workflows from Business Intents," Doctor of Philosophy in Computer Science, University of Arkansas, Fayetteville, University of Arkansas, Fayetteville ScholarWorks@UARK, 2012.
- [6] S. Lasya and S. Tanuku, "A Study of Library Databases by Translating Those SQL Queries Into Relational Algebra and Generating Query Trees," 2011.
- [7] V. Santos and O. Belo, "Modelling ETL Conciliation Tasks Using Relational Algebra Operators," presented at the 2014 UKSim-AMSS 8th European Modelling Symposium, 2014.
- [8] Z. Hemel, L. C. L. Kats, and E. Visser, "Code Generation by Model Transformation. A Case Study in Transformation Modularity," presented at the International Conference on Model Transformation (ICMT'08), Lecture Notes in Computer Science, 2008.
- [9] *SQL Tutorial*. [online].Available: <http://www.w3resource.com/sql/tutorials.php>



ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ก

แบบสำรวจความพึงพอใจ

การใช้ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอล
(A Domain Specific Language for Scripting ETL Process)

วันที่.....เดือน.....พ.ศ.....

คำชี้แจง แบบสำรวจความพึงพอใจชุดนี้ จัดทำขึ้นเพื่อใช้ในการติดตามประเมินผลการทดลองใช้ และปรับปรุงคุณภาพภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอล จึงใคร่ขอความร่วมมือจากท่านโปรดกรอกข้อมูลตามความเป็นจริง โดยให้ท่านกรุณาทำเครื่องหมาย ✓ ลงในช่องระดับความพึงพอใจที่ตรงกับความรู้สึกที่แท้จริงของท่านมากที่สุด

ส่วนที่ 1 ข้อมูลเกี่ยวกับผู้ประเมิน	
ชื่อ	บริษัท
ส่วนงานที่เกี่ยวข้อง <input type="checkbox"/> Project Manager <input type="checkbox"/> Functional Specialist <input type="checkbox"/> Technical Specialist <input type="checkbox"/> อื่นๆ โปรดระบุ.....	
ระบุ.....	
ประสบการณ์การทำงาน <input type="checkbox"/> 1-5 ปี <input type="checkbox"/> 6-10 ปี <input type="checkbox"/> 11 ปีขึ้นไป	
เพศ <input type="checkbox"/> ชาย <input type="checkbox"/> หญิง <input type="checkbox"/> อื่นๆ โปรดระบุ.....	
ส่วนที่ 2 สสำรวจการใช้งานทั่วไป	
1. ท่านเคยใช้งานเครื่องมือที่เกี่ยวข้องกับอีทีแอลหรือไม่ <input type="checkbox"/> ใช่ <input type="checkbox"/> ไม่ใช่ เนื่องจาก.....	
2. ท่านเคยใช้เอสคิวแอลหรือไม่ <input type="checkbox"/> ใช่ <input type="checkbox"/> ไม่ใช่ เนื่องจาก.....	
3. จุดประสงค์ในการใช้งานเครื่องมืออีทีแอล <input type="checkbox"/> ใช้สำหรับการพัฒนากระบวนการอีทีแอล <input type="checkbox"/> ใช้สำหรับเรียกดูหรือนำออกข้อมูลจากฐานข้อมูล <input type="checkbox"/> อื่นๆ โปรดระบุ	
4. จุดประสงค์ในการใช้งานเอสคิวแอล <input type="checkbox"/> ใช้สำหรับการพัฒนากระบวนการอีทีแอล <input type="checkbox"/> ใช้สำหรับเรียกดูหรือนำออกข้อมูลจากฐานข้อมูล <input type="checkbox"/> อื่นๆ โปรดระบุ	

5. ระดับความซับซ้อนของการใช้เอสคิวแอล					
<input type="checkbox"/> ซับซ้อนมาก		<input type="checkbox"/> ค่อนข้างซับซ้อน			
<input type="checkbox"/> ปานกลาง		<input type="checkbox"/> ง่าย			
ส่วนที่ 3 การแสดงความคิดเห็นเกี่ยวกับการใช้ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอล					
หัวข้อการประเมินผล	ระดับความพึงพอใจ				
	พอใจ มาก ที่สุด (5)	พอใจ มาก (4)	ปาน กลาง (3)	พอใจ น้อย (2)	ไม่ พอใจ (1)
1. ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอลสะดวกแก่การนำมาใช้ในการปฏิบัติงานดีกว่าหรือเทียบเท่าเครื่องมืออีทีแอลในปัจจุบัน					
2. ชุดคำสั่งภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอลสามารถจดจำเรียนรู้ได้ง่าย					
3. ผลลัพธ์ที่ได้มีความถูกต้อง					
4. ภาษาจำเพาะโดเมนสามารถตอบสนองความต้องการใช้งานสำหรับกระบวนการอีทีแอล					
5. ความพึงพอใจโดยภาพรวมในการใช้ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอล					

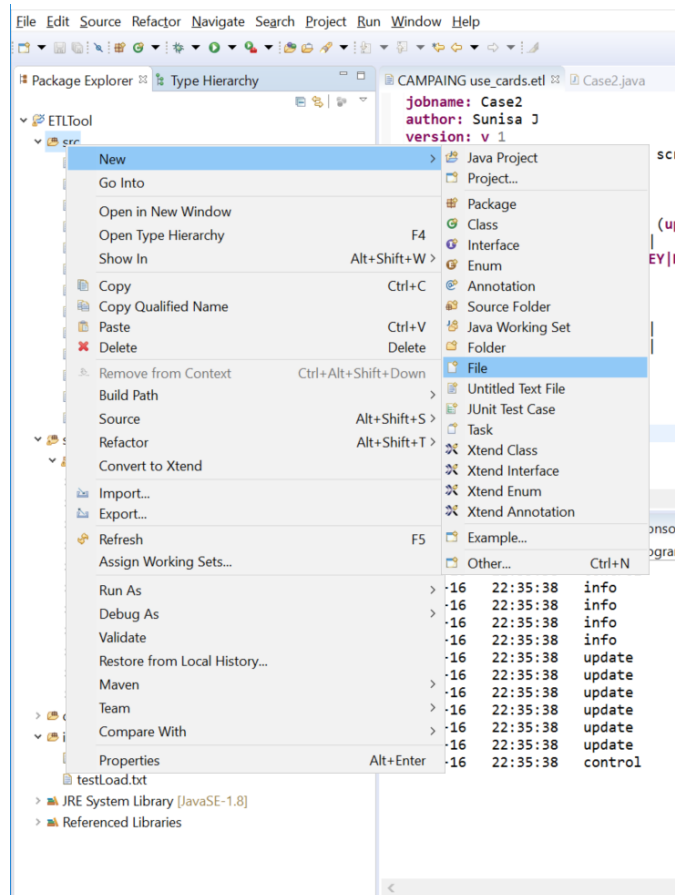
ส่วนที่ 4 ข้อเสนอแนะทั่วไป

1. สิ่งที่ท่านประทับใจในการใช้การใช้ภาษาจำเพาะโดเมนสำหรับเขียนบทคำสั่งกระบวนการอีทีแอล
2. ท่านคิดว่าควรปรับปรุงด้านใด
3. ข้อเสนอแนะ

ภาคผนวก ข

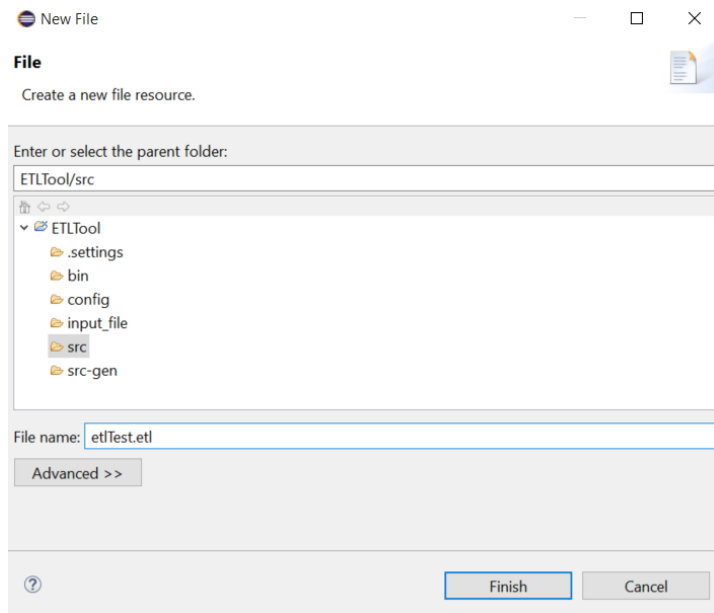
คู่มือการใช้งาน

สำหรับการเขียนสคริปต์ดีเอสแอลอีทีแอลโดยใช้ Eclipse IDE for Java Developers Version: Neon.1a Release (4.6.1) ผู้ใช้จะเข้าในหน้าจำลองโปรแกรม และสามารถสร้างไฟล์นามสกุล .etl เพื่อการเขียนสคริปต์ดีเอสแอลอีทีแอลด้วยการคลิกขวาที่แฟ้ม src เลือก New และ File ตามภาพที่ 59

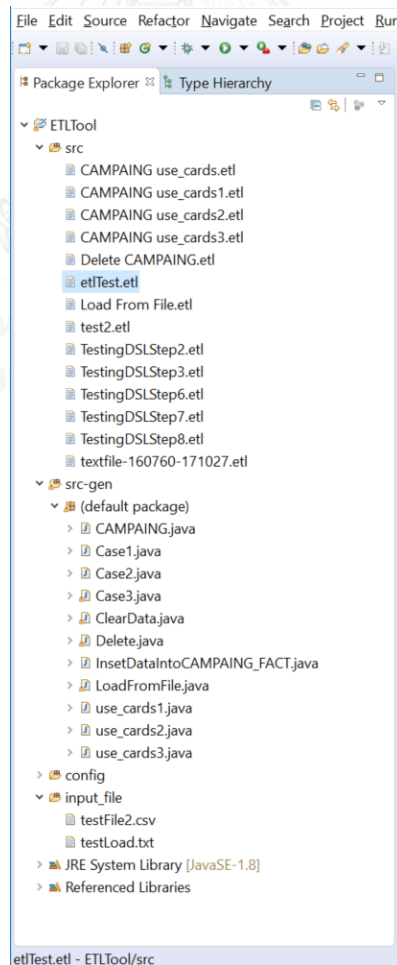


ภาพที่ 59 การสร้างไฟล์นามสกุล .etl สำหรับการเขียนสคริปต์อีทีแอลดีเอสแอล

เมื่อทำตามภาพที่ 59 แล้วจะแสดงหน้าต่างโปรแกรมในภาพที่ 60 สามารถสร้างไฟล์นามสกุล .etl ได้ตามที่ต้องการ เมื่อคลิกปุ่ม Finish ไฟล์นามสกุล .etl ที่สร้างขึ้นจะแสดงในแฟ้ม src ในภาพที่ 61

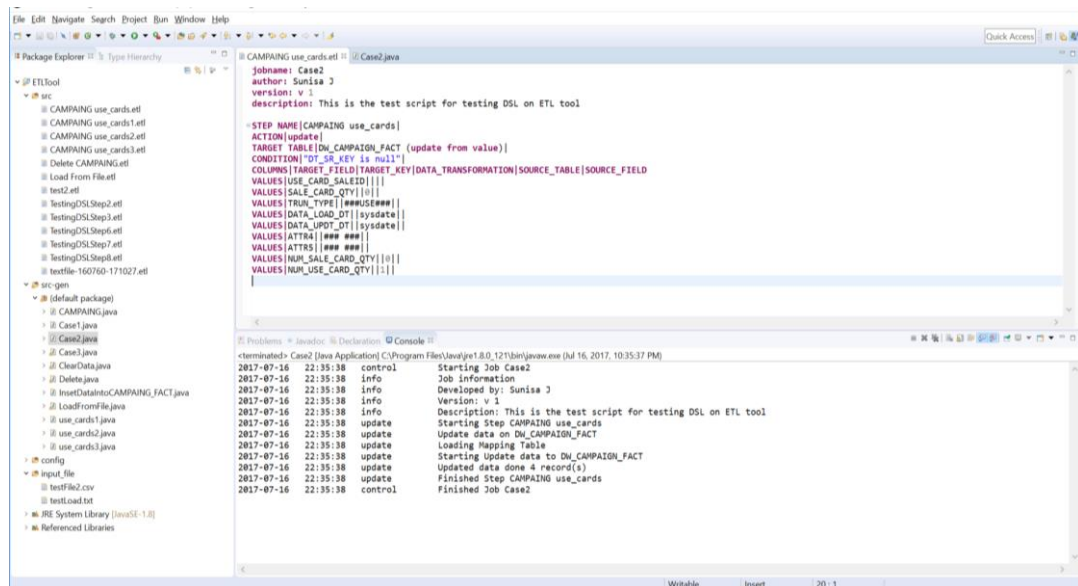


ภาพที่ 60 สร้างชื่อไฟล์ของสคริปต์ดีเอสแอลอีทีแอล



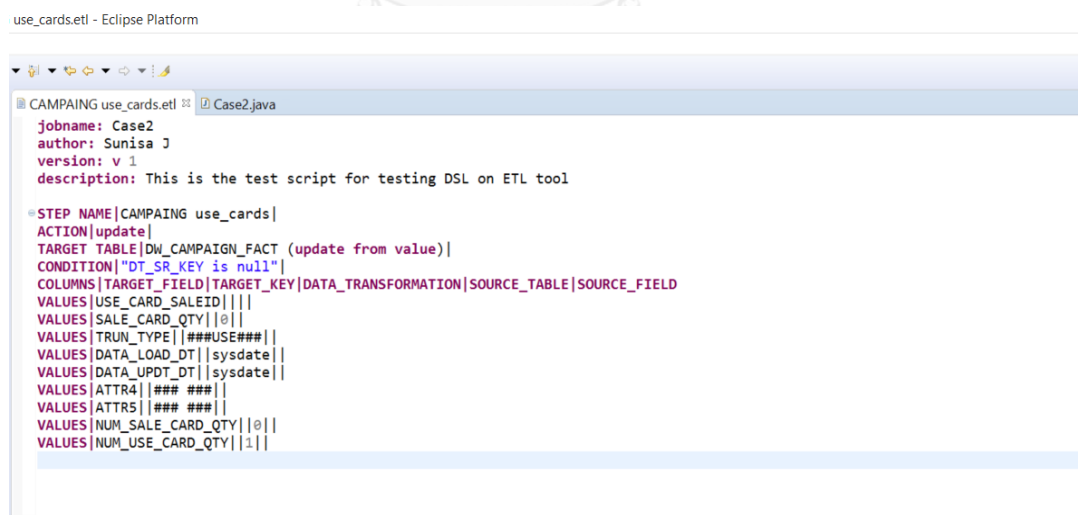
ภาพที่ 61 ตัวอย่างไฟล์สคริปต์ดีเอสแอลอีทีแอล

โปรแกรมสำหรับเขียนสคริปต์ดีเอสแอลอีทีแอลดั่งภาพที่ 62 ประกอบด้วยหน้าต่างสำหรับเขียนสคริปต์ดีเอสแอลอีทีแอล และ log ที่ได้จากการประมวลผล



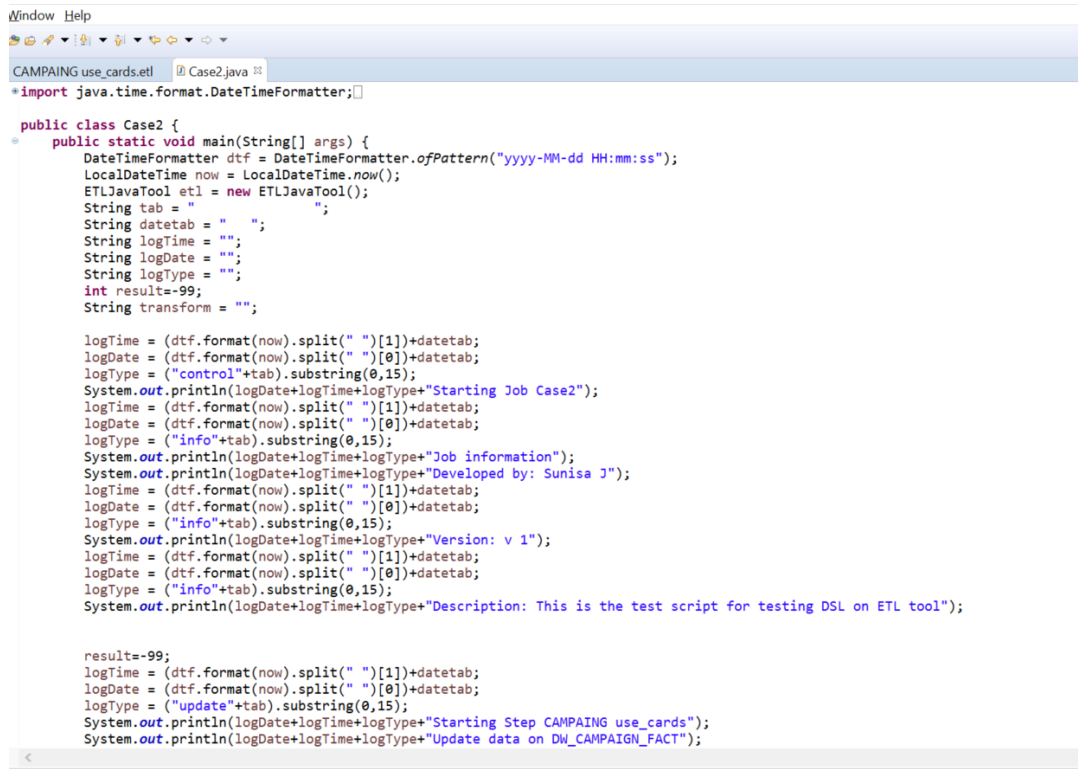
ภาพที่ 62 ตัวอย่างของโปรแกรมสำหรับเขียนสคริปต์ดีเอสแอลอีทีแอล

การเขียนสคริปต์ดีเอสแอลอีทีแอลสามารถใช้เครื่องหมาย | สำหรับการค้นค่าสั่งและค่าสำคัญของไวยากรณ์ดีเอสแอล และเครื่องหมาย ### ### ใช้แทน ‘ ’ หรือ “ ” ตามภาพที่ 63



ภาพที่ 63 ตัวอย่างการเขียนสคริปต์ดีเอสแอลอีทีแอล

หลังจากทำการบันทึกไฟล์สคริปต์ดีเอสแอลอีทีแอลนั้นจะได้ไฟล์จาวาอัตโนมัติสำหรับ
กระบวนการประมวลผลในลำดับต่อไปตัวอย่างในภาพที่ 64



```

Window Help
CAMPAING use_cards.etl Case2.java
*import java.time.format.DateTimeFormatter;

public class Case2 {
    public static void main(String[] args) {
        DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
        LocalDateTime now = LocalDateTime.now();
        ETLJavaTool etl = new ETLJavaTool();
        String tab = " ";
        String datetab = " ";
        String logTime = " ";
        String logDate = " ";
        String logType = " ";
        int result=-99;
        String transform = " ";

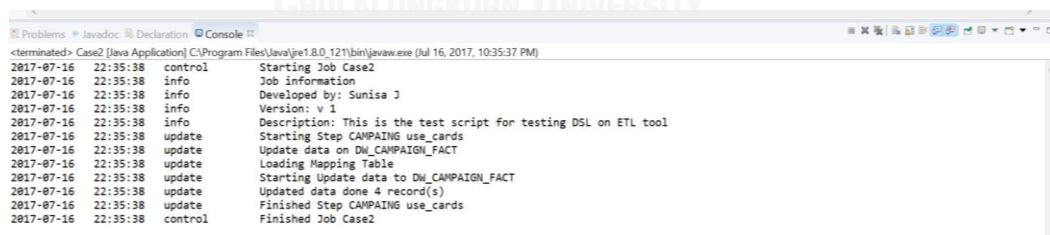
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("control"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Starting Job Case2");
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("info"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Job information");
        System.out.println(logDate+logTime+logType+"Developed by: Sunisa J");
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("info"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Version: v 1");
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("info"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Description: This is the test script for testing DSL on ETL tool");

        result=-99;
        logTime = (dtf.format(now).split(" ")[1])+datetab;
        logDate = (dtf.format(now).split(" ")[0])+datetab;
        logType = ("update"+tab).substring(0,15);
        System.out.println(logDate+logTime+logType+"Starting Step CAMPAING use_cards");
        System.out.println(logDate+logTime+logType+"Update data on DW_CAMPAIGN_FACT");
    }
}

```

ภาพที่ 64 ตัวอย่างไฟล์จาวาที่ได้จากสคริปต์ดีเอสแอลอีทีแอล

หลังจากทำการประมวลผลภาพที่ 64 โปรแกรมจะแสดงขั้นตอนการประมวลผลที่ละขั้นตอน
ของกระบวนการอีทีแอลทีละขั้นตอนตัวอย่างในภาพที่ 65



```

Problems Javadoc Declaration Console
<terminated> Case2 [Java Application] C:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (Jul 16, 2017, 10:35:37 PM)
2017-07-16 22:35:38 control Starting Job Case2
2017-07-16 22:35:38 info Job information
2017-07-16 22:35:38 info Developed by: Sunisa J
2017-07-16 22:35:38 info Version: v 1
2017-07-16 22:35:38 info Description: This is the test script for testing DSL on ETL tool
2017-07-16 22:35:38 update Starting Step CAMPAING use_cards
2017-07-16 22:35:38 update Update data on DW_CAMPAIGN_FACT
2017-07-16 22:35:38 update Loading Mapping Table
2017-07-16 22:35:38 update Starting Update data to DW_CAMPAIGN_FACT
2017-07-16 22:35:38 update Updated data done 4 record(s)
2017-07-16 22:35:38 update Finished Step CAMPAING use_cards
2017-07-16 22:35:38 control Finished Job Case2

```

ภาพที่ 65 ตัวอย่าง log การประมวลผลของสคริปต์ดีเอสแอลอีทีแอล

การเขียนสคริปต์ดีเอสแอลอีทีแอลนั้นสามารถใช้ ETL-Template ในภาพที่ 66 เป็นเครื่องมือสำหรับการสร้างไฟล์นามสกุล .etl ได้อัตโนมัติ ซึ่งวิธีการนี้จะช่วยให้การสร้างสคริปต์ดีเอสแอลอีทีแอลทำได้ง่ายกว่าการเขียนสคริปต์ดีเอสแอลอีทีแอลผ่านโปรแกรม Eclipse และยังเป็นการประยุกต์จากวิธีการเดิมที่ใช้งานในปัจจุบัน เงื่อนไขการแปลงข้อมูลแบ่งได้ 4 แบบ

1. ค่า null หรือการนำเข้าข้อมูลแบบมีเงื่อนไข (Constraint) เช่น move direct, null เป็นต้น
2. การเรียกใช้ฟังก์ชันการทำงานของฐานข้อมูล (Function) เช่น nvl(), sum() เป็นต้น
3. ค่าคงที่ตัวเลข (Number) เช่น 0, 1 เป็นต้น
4. ค่าคงที่ตัวอักษร (String) เช่น 'USE' เป็นต้น

The screenshot shows an Excel spreadsheet titled 'etlDSL-Template'. It contains the following content:

ETL Spec
 jobname: Case1
 author: Sunisa J
 version: v 1
 description: This is the test script for testing DSL on ETL tool

STEP NAME	Delete CAMPAIGN				
ACTION	delete				
TARGET TABLE	CAMPAIGN_FACT				
CONDITION	'(STORE_DT_KEY) in (select STORE_IDT_KEY from campaign_fact c, STG_LOG1 where c.STORE = 1STORE and c.DT_KEY = 1DT_KEY) '				
STEP NAME	CAMPAIGN use_cards				
ACTION	insert				
TARGET TABLE	CAMPAIGN_FACT (update and insert)				
CONDITION	"sale_attribute.saleheaderid = sales_header_fct.sale_id and sales_header_fct.store = store_dim.store and sale_attribute.att056 = '2' group by sale_attribute.saleheaderid, to_char(to_date(substr(sale_attribute.att040, 1, 10), 'yyyy-mm-dd'), 'yyyymmdd'), sales_header_fct.dt_key, sales_header_fct.store, store_dim.store_nm, sales_header_fct.channel_sr_key, sale_attribute.att039, sale_attribute.ATT054, sale_attribute.ATT055, sale_attribute.ATT057, sale_attribute.ATT046 "				
COLUMNS	TARGET_FIELD	TARGET_KEY	DATA_TRANSFORMATION	SOURCE_TABLE	SOURCE_FIELD
VALUES	DT_KEY		move direct	sales_header_fct	dt_key
VALUES	CHANNEL_SR_KEY		move direct	sales_header_fct	channel_sr_key
VALUES	STORE	key	move direct	sales_header_fct	store
VALUES	STORE_NM		move direct	store_dim	store_nm
VALUES	CARD_NUMBER	key	move direct	sale_attribute	att046
VALUES	SALE_CARD_SALEID		null		
VALUES	USE_CARD_SALEID	key	move direct	sale_attribute	saleheaderid
VALUES	SALE_CARD_QTY		0		
VALUES	USE_CARD_QTY		nvl(sum(att043),0)		
VALUES	TRUN_TYPE		###USE###		
VALUES	DATA_LOAD_DT		sysdate		
VALUES	DATA_UPDT_DT		sysdate		
VALUES	ATTR		move direct	sale_attribute	ATT054

ภาพที่ 66 ตัวอย่าง excel สำหรับการเขียนสคริปต์ดีเอสแอลอีทีแอล

ตารางที่ 5 ชุดคำสั่งที่เขียนด้วยสคริปต์ดีเอสแอลทีแอล [9]

ประเภท	ชุดคำสั่ง
SQL Insert Statement	<ul style="list-style-type: none"> - insert (การนำข้อมูลเข้าสู่ตาราง) - clear and insert (ลบข้อมูลเดิมในตารางก่อนทำการนำเข้าสู่ข้อมูลสู่ตาราง) - Insert from file (การนำเข้าข้อมูลจากไฟล์สู่ตาราง)
SQL Update Statement	<ul style="list-style-type: none"> - update (การปรับข้อมูลให้เป็นปัจจุบัน) - update and insert (การปรับข้อมูลให้เป็นปัจจุบันก่อนการนำเข้าข้อมูลสู่ตาราง) - update from value (การปรับข้อมูลให้เป็นปัจจุบันจากข้อมูลเฉพาะเจาะจง) - update from table (การปรับข้อมูลให้เป็นปัจจุบันจากข้อมูลที่มีอยู่ตารางอื่นหรือตารางเดิม)
SQL Delete	<ul style="list-style-type: none"> - delete (การลบข้อมูลในตาราง) - clear (การลบข้อมูลทั้งตารางแบบไม่มีเงื่อนไข)
SQL Select Statement	<p>SELECT <column_list> FROM <table_name> (การดึงข้อมูลจากราย)</p>
SQL Joins	<p>SELECT <column_list> FROM TableA, TableB WHERE <TableA.column1 = TableB.column1 > (การดึงข้อมูลจากราย 2 ตารางขึ้นไปแบบมีเงื่อนไขความเชื่อมโยงกันของแต่ละตาราง)</p>
SQL Where Clause	<p>WHERE <condition> ได้แก่</p> <ul style="list-style-type: none"> - Comparison : =, >, >=, <, <=, <> - Range filtering : BETWEEN - Match a character pattern : LIKE - List filtering [Match any of a list of values] : IN - Null testing : IS NULL <p>(การกำหนดเงื่อนไขสำหรับกรณีที่เกี่ยวข้อง)</p>

ประเภท	ชุดคำสั่ง
SQL Aggregate Function	<ul style="list-style-type: none"> - SQL Count function (การนับจำนวนข้อมูลในตารางแบบมีเงื่อนไขและแบบไม่มีเงื่อนไข) - SQL Sum function (การบวกรวมข้อมูลในตารางแบบมีเงื่อนไขและแบบไม่มีเงื่อนไข) - SQL Average function (การหาค่าเฉลี่ยของข้อมูลในตารางแบบมีเงื่อนไขและแบบไม่มีเงื่อนไข) - SQL Max function (การหาค่าสูงสุดข้อมูลในตารางแบบมีเงื่อนไขและแบบไม่มีเงื่อนไข) - SQL Min function (การหาค่าต่ำสุดข้อมูลในตารางแบบมีเงื่อนไขและแบบไม่มีเงื่อนไข)
SQL Character Function	<ul style="list-style-type: none"> - lower() (การแปลงข้อมูลตัวอักษรในภาษาอังกฤษให้เป็นตัวอักษรตัวเล็กในภาษาอังกฤษ) - upper() (การแปลงข้อมูลตัวอักษรในภาษาอังกฤษให้เป็นตัวอักษรตัวใหญ่ในภาษาอังกฤษ) - trim() (การลบช่องว่างที่เกิดขึ้นข้างหน้าและหลังของชุดตัวอักษร)

ประวัติผู้เขียนวิทยานิพนธ์

นางสาว สุนิสา จันทร์สว่าง เกิดเมื่อวันที่ 19 พฤษภาคม พ.ศ. 2534 ที่จังหวัดกาญจนบุรี สำเร็จการศึกษาหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วท.บ.) สาขาวิทยาการคอมพิวเตอร์ สำนักวิชาเทคโนโลยีสารสนเทศ มหาวิทยาลัยแม่ฟ้าหลวง ในปีการศึกษา 2556 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาศาสตรคอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2558

