การปรับปรุงอัลกอริทึมการตรวจหาแบบรูปการออกแบบเชิงโครงสร้างด้วยกราฟเชิงความสัมพันธ์

นางสาวชลิดา เหลี่ยมวิเศษ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2556

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย
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 Graduate School.

IMPROVEMENT OF STRUCTURAL DESIGN PATTERNS DETECTION ALGORITHM USING RELATIONAL GRAPH

Miss Chalida Liamwiset

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Software Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2013
Copyright of Chulalongkorn University

ชลิดา เหลี่ยมวิเศษ : การปรับปรุงอัลกอริทึมการตรวจหาแบบรูปการออกแบบเชิงโครงสร้างด้วยกราฟเชิงความสัมพันธ์ (IMPROVEMENT OF STRUCTURAL DESIGN PATTERNS DETECTION ALGORITHM USING RELATIONAL GRAPH) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: รศ.ดร.วิวัฒน์ วัฒนาวุฒิ, 123หน้า

แผนภาพยูเอ็มแอลถูกนำมาใช้แสดงถึงการออกแบบระบบซอฟต์แวร์ โดยเฉพาะอย่างยิ่งแผนภาพคลาสที่ใช้สำหรับการแสดงถึงโครงสร้างของการออกแบบ แบบรูปการออกแบบในแผนภาพคลาสได้ถูกนำเสนอมาใช้ป้องกันปัญหาที่เกิดขึ้นซ้ำๆ และแนะนำแนวทางปฏิบัติที่ดีที่สุดในการออกแบบที่ยอมให้สามารถนำกลับมาใช้ใหม่และการบำรุงรักษา

วิทยานิพนธ์นี้นำเสนอการปรับปรุงขั้นตอนวิธีการตรวจจับแบบรูปการออกแบบเชิงโครงสร้างด้วยกราฟเชิงความสัมพันธ์ มุ่งเน้นแบบโครงสร้างของการออกแบบซอฟต์แวร์เพียงอย่างเดียว ผู้วิจัยได้แสดงการตรวจจับ 7 แบบรูปการออกแบบเชิงโครงสร้าง ประกอบไปด้วย แบบรูป Adaptor แบบรูป Bridge แบบรูป Composite แบบรูป Decorator แบบรูป Façade แบบรูป Flyweight และแบบรูป Proxy ข้อมูลนำเข้าเป็นแผนภาพคลาสที่ตรงตามมาตรฐานรูปแบบเอกซ์เอ็มไอและความสัมพันธ์ของกราฟที่ถูกสร้างขึ้น คุณลักษณะท้องถิ่นของแผนภาพคลาสที่เกี่ยวข้องจะถูกสกัดตามเงื่อนไข 12 ลำดับของเส้นความสัมพันธ์เข้า-ออก และด้วยความสัมพันธ์ที่ใกล้กันระหว่างโหนด ผลลัพธ์กระบวนการจับคู่และระบุตำแหน่งได้ผลทั้งการจับคู่แบบแม่นยำและการจับคู่ที่ไม่แม่นยำของแบบรูปการออกแบบ

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต.....

สาขาวิชา วิศวกรรมซอฟต์แวร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก.....

ปีการศึกษา 2556

547092112 : MAJOR SOFTWARE ENGINEERING
KEYWORDS : RELATIONAL GRAPH/DESIGN PATTERN/SOFTWARE DESIGN

CHALIDA LIAMWISET: IMPROVEMENT OF STRUCTURAL DESIGN PATTERNS DETECTION ALGORITHM USING RELATIONAL GRAPH. ADVISOR : ASSOC. PROF. WIWAT VATANAWOOD, 123pp.

The UML diagrams are typically used to represent the software model. Especially the class diagram is used for structural design model. Design patterns in class diagrams are introduced to prevent the recurring design problems and guide the best practices in design - yielding the reusability and maintainability.

This thesis proposes the improvement of the design patterns detecting algorithm using relational graph. The structural model of software design is solely focused. We demonstrate the detection of seven structural design patterns - Adaptor, Bridge, Composite, Decorator, Facade, Flyweight, and Proxy. The input class diagram is expected in the standard XMI format and the corresponding relational graph is generated. The local properties of the related class diagrams are then extracted in terms of their 12-tuples of fan-in and fan-out relations and also the adjacent relations among nodes. Our matching process results and locates both exact match and inexact match design patterns.

Department : Computer Engineering Student's Signature .....
Field of Study: Software Engineering Advisor's Signature .....
Academic Year : 2013 .....

หัวข้อวิทยานิพนธ์	การปรับปรุงอัลกอริทึมการตรวจหาแบบรูปการออกแบบเชิงโครงสร้างด้วยกราฟเชิงความสัมพันธ์
โดย	นางสาวชลิดา เหลี่ยมวิเศษ
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.บัณฑิต เอื้ออาภรณ์)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร. ธาราทิพย์ สุวรรณศาสตร์)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์)

.....กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.ทรงศักดิ์ รองวิริยะพานิช)

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยการได้รับคำแนะนำ คำปรึกษาจากอาจารย์ที่ปรึกษาหลัก รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ อีกทั้งยังแนะนำแนวทางการแก้ไขปัญหาต่างๆ ที่จำเป็นต่อการจัดทำวิทยานิพนธ์ นอกจากนี้ผู้วิจัยขอกราบขอบพระคุณเป็น อย่างสูงมา ณ ที่นี้

ขอขอบพระคุณรองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์ ประธานกรรมการสอบวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์ และ ผู้ช่วยศาสตราจารย์ ดร.ทรงศักดิ์ รองวิริยะพานิช กรรมการสอบวิทยานิพนธ์ ที่ได้ให้คำแนะนำและชี้แนะแนวทางที่เป็นประโยชน์ต่อการทำวิทยานิพนธ์ในครั้งนี้

ขอขอบคุณพี่ๆ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคน ที่ร่วมแลกเปลี่ยนความรู้ให้แง่คิดต่างๆ ตลอดระยะเวลาที่ดำเนินการวิจัย และเพื่อน ๆ ร่วมงานทุกท่านที่คอยให้คำแนะนำและช่วยเหลือด้วยดีอย่างเสมอมา

สุดท้ายขอขอบพระคุณ คุณพ่อ คุณแม่ ที่คอยสนับสนุนให้ผู้วิจัยได้มีโอกาสศึกษาเล่าเรียนตลอดจนถึงบัดนี้ และคอยเป็นกำลังใจให้โดยตลอดมา

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ง

บทคัดย่อภาษาอังกฤษ.....จ

กิตติกรรมประกาศ.....ฉ

สารบัญ.....ช

สารบัญตาราง.....ฎ

สารบัญรูป.....ฐ

บทที่ 1 บทนำ.....1

1.1. ความเป็นมาและความสำคัญของปัญหา.....1

1.2. วัตถุประสงค์การวิจัย.....2

1.3. ขอบเขตของงานวิจัย.....2

1.4. ขั้นตอนและวิธีดำเนินการวิจัย.....3

1.5. ผลงานที่ได้รับการตีพิมพ์.....4

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....5

2.1. ทฤษฎีที่เกี่ยวข้อง.....5

2.1.1. แบบรูปการออกแบบ (Design Pattern) 5

2.1.1.1. แบบรูปการออกแบบเพื่อการสร้าง (Creational design patterns).....6

2.1.1.2. แบบรูปการออกแบบเชิงโครงสร้าง (Structural design patterns).....7

2.1.1.3. แบบรูปการออกแบบพฤติกรรม (Behavioral design patterns).....14

2.1.2. เอกซ์เอ็มไอ (XMI).....16

2.1.3. การจับคู่ความสัมพันธ์ของกราฟ (Relational Graph Matching).....17

2.2. งานวิจัยที่เกี่ยวข้อง.....18

2.2.1. งานวิจัยเรื่อง “Design Pattern Detection using Similarity Scoring”.....18

2.2.2. งานวิจัยเรื่อง “Design Pattern Detection by Template Matching”.....19

2.2.3. งานวิจัยเรื่อง “A Review of Design Pattern Mining Techniques”.....19

2.2.4. งานวิจัยเรื่อง “Design Pattern Detection using Inexact Graph Matching”.....20

2.2.5. งานวิจัยเรื่อง “Mining Object-Oriented Design Models for Detecting Identical Design Structures”.....23

หน้า

บทที่ 3 การออกแบบและปรับปรุงอัลกอริทึมการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง.....24
3.1. กำหนดคำนิยามในการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง.....24
3.1.1. กราฟเชิงความสัมพันธ์.....24
3.1.2. ตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส(Local Properties Table: LPT)..27
3.1.3. ตารางความสัมพันธ์ของคลาส (Relational Table: RT).....28
3.1.4. โหนดอ้างอิง (Reference Node).....28
3.1.5. ความเหมือนแบบรูปการออกแบบและวิธีการคำนวณความเหมือน.....29
3.2. วิธีการแปลงแผนภาพคลาสแบบรูปการออกแบบเป็นกราฟเชิงความสัมพันธ์.....29
3.2.1. แบบรูปการออกแบบ Adapter.....29
3.2.2. แบบรูปการออกแบบ Bridge.....31
3.2.3. แบบรูปการออกแบบ Composite.....33
3.2.4. แบบรูปการออกแบบ Decorator.....35
3.2.5. แบบรูปการออกแบบ Façade.....37
3.2.6. แบบรูปการออกแบบ Flyweight.....39
3.2.7. แบบรูปการออกแบบ Proxy.....41
3.3. ขั้นตอนการสร้างตารางคุณลักษณะท้องถิ่น (LPT-S) และตารางความสัมพันธ์ (RT-S) ของแบบรูปการออกแบบ.....43
3.3.1. แบบรูปการออกแบบ Adapter.....43
3.3.2. แบบรูปการออกแบบ Bridge.....44
3.3.3. แบบรูปการออกแบบ Composite.....45
3.3.4. แบบรูปการออกแบบ Decorator.....46
3.3.5. แบบรูปการออกแบบ Façade.....47
3.3.6. แบบรูปการออกแบบ Flyweight.....48
3.3.7. แบบรูปการออกแบบ Proxy.....49
3.4. การเลือกโหนดอ้างอิงสำหรับแต่ละแบบรูปการออกแบบ.....49
3.4.1. โหนดอ้างอิงของแบบรูปการออกแบบ Adapter.....50
3.4.2. โหนดอ้างอิงของแบบรูปการออกแบบ Bridge.....51
3.4.3.โหนดอ้างอิงของแบบรูปการออกแบบ Composite.....51
3.4.4. โหนดอ้างอิงของแบบรูปการออกแบบ Decorator.....52

หน้า

3.4.5. โหนดอ้างอิงของแบบรูปการออกแบบ Façade.....52
3.4.6. โหนดอ้างอิงของแบบรูปการออกแบบ Flyweight.....53
3.4.7. โหนดอ้างอิงของแบบรูปการออกแบบ Proxy.....53
3.5 การกำหนดความเหมือนแบบรูปการออกแบบ.....54
3.5.1 ความเหมือนแบบรูปการออกแบบ Adapter.....54
3.5.2 ความเหมือนแบบรูปการออกแบบ Bridge.....54
3.5.3 ความเหมือนแบบรูปการออกแบบ Composite 55
3.5.4 ความเหมือนแบบรูปการออกแบบ Decorator.....56
3.5.5 ความเหมือนแบบรูปการออกแบบ Façade.....56
3.5.6 ความเหมือนแบบรูปการออกแบบ Flyweight.....57
3.5.7 ความเหมือนแบบรูปการออกแบบ Proxy.....57
3.6. ขั้นตอนวิธีนำแผนภาพคลาสของระบบมาตรวจจับแบบรูปการออกแบบ.....58
3.6.1. การแปลงแผนภาพคลาสระบบให้เป็นกราฟเชิงความสัมพันธ์.....58
3.6.2. ตารางคุณลักษณะท้องถิ่น (LPT-S) และตารางความสัมพันธ์ (RT-S) ของแผนภาพคลาสระบบตัวอย่าง.....59
3.7. การปรับปรุงอัลกอริทึมการตรวจจับแบบรูปการออกแบบ.....60
3.8. ตัวอย่างการตรวจจับแบบรูปการออกแบบ Adapter จากอัลกอริทึมที่ปรับปรุง....66
บทที่ 4 การวิเคราะห์ ออกแบบ และพัฒนาเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง....89
4.1. การวิเคราะห์การออกแบบการตรวจจับแบบรูปการออกแบบ.....89
4.1.1.แผนภาพยูสเคส.....89
4.1.2.แผนภาพคลาส.....95
4.2. แผนภาพองค์ประกอบ.....96
4.2.1. DSPT Controller (Detect Structure Pattern Tool).....96
4.2.2. XMI Word Cut.....97
4.2.3. Matching Node.....97
4.2.4. Report Generator.....97
4.3. หลักการทำงานของเครื่องมือตรวจจับแบบรูปการออกแบบ.....98
บทที่ 5 การทดสอบเครื่องมือการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง.....100
5.1. การทดสอบเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง.....100

5.1.1. การทดสอบกรณีทดสอบที่หนึ่ง.....101
5.1.2. การทดสอบกรณีทดสอบที่สอง.....104
5.1.3. การทดสอบกรณีทดสอบที่สาม.....108
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....111
6.1. สรุปผลการวิจัย.....111
6.2. ปัญหาและข้อจำกัดที่พบจากงานวิจัย.....112
6.3. ประโยชน์ที่ได้รับจากงานวิจัย.....113
6.4. แนวทางในการพัฒนางานวิจัยต่อ.....113
รายการอ้างอิง.....114
ภาคผนวก.....116
ภาคผนวก ก.....117
ภาคผนวก ข.....119
ภาคผนวก ค.....121
ประวัติผู้เขียนวิทยานิพนธ์.....123

สารบัญตาราง

หน้า

ตารางที่ 3.1 ลักษณะของเส้นความสัมพันธ์ 25
ตารางที่ 3.2 การนิยามค่าของแต่ละประเภทเส้นความสัมพันธ์.....26
ตารางที่ 3.3 ตัวอย่างการจัดเก็บข้อมูลในตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส.....27
ตารางที่ 3.4 ตัวอย่างการจัดเก็บข้อมูลในตารางความสัมพันธ์ของคลาส ตามรูปที่ 3.1.....28
ตารางที่ 3.5 ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter.....43
ตารางที่ 3.6 ตารางความสัมพันธ์ของแบบรูปการออกแบบ Adapter.....43
ตารางที่ 3.7 ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Bridge.....44
ตารางที่ 3.8 ความสัมพันธ์ของแบบรูปการออกแบบ Bridge.....44
ตารางที่ 3.9 ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Composite.....45
ตารางที่ 3.10 ความสัมพันธ์ของแบบรูปการออกแบบ Composite 45
ตารางที่ 3.11 ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Decorator.....46
ตารางที่ 3.12 ความสัมพันธ์ของแบบรูปการออกแบบ Decorator.....46
ตารางที่ 3.13 ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Façade.....47
ตารางที่ 3.14 ความสัมพันธ์ของแบบรูปการออกแบบ Façade.....47
ตารางที่ 3.15 ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Flyweight.....48
ตารางที่ 3.16 ความสัมพันธ์ของแบบรูปการออกแบบ Flyweight.....48
ตารางที่ 3.17 ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Proxy.....49
ตารางที่ 3.18 ความสัมพันธ์ของแบบรูปการออกแบบ Proxy.....49
ตารางที่ 3.19 ตารางคุณลักษณะท้องถิ่นของแผนภาพคลาสระบบตัวอย่าง.....59
ตารางที่ 3.20 ตารางความสัมพันธ์ของแผนภาพคลาสระบบตัวอย่าง.....60
ตารางที่ 3.21 คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ (LPT-S).....67
ตารางที่ 3.22 ความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S).....67
ตารางที่ 3.23 คุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P).....68
ตารางที่ 3.24 ตารางความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P).....68
ตารางที่ 4.1 คำอธิบายยูสเคสการสร้างแผนภาพคลาส.....90
ตารางที่ 4.2 คำอธิบายยูสเคสการส่งออกไฟล์เอกซ์เอ็มไอ.....90
ตารางที่ 4.3 คำอธิบายยูสเคสการสร้างตารางคุณลักษณะ.....91
ตารางที่ 4.4 คำอธิบายยูสเคสการตรวจคุณลักษณะคลาส.....91

หน้า

ตารางที่ 4.5 คำอธิบายยูสเคสการสร้างตารางความสัมพันธ์.....92
ตารางที่ 4.6 คำอธิบายยูสเคสการตรวจจับคลาสต้นทาง-ปลายทาง.....92
ตารางที่ 4.7 คำอธิบายยูสเคสการตรวจจับแบบรูปการออกแบบ.....93
ตารางที่ 4.8 คำอธิบายยูสเคสการตรวจจับโหนดอ้างอิง.....93
ตารางที่ 4.9 คำอธิบายยูสเคสการตรวจจับรายละเอียด.....94
ตารางที่ 4.10 คำอธิบายยูสเคสการออกรายงานสรุปผล.....94
ตารางที่ 5.1 ผลลัพธ์จากการทดสอบกรณีทดสอบที่หนึ่ง.....102
ตารางที่ 5.2 ผลลัพธ์จากการทดสอบกรณีทดสอบที่สอง.....105
ตารางที่ 5.3 ผลลัพธ์จากการทดสอบกรณีทดสอบที่สาม.....109

สารบัญรูป

หน้า

รูปที่ 2.1 ต้นแบบแผนภาพคลาสของแบบรูป Adapter....7
รูปที่ 2.2 ต้นแบบแผนภาพคลาสของแบบรูป Bridge....8
รูปที่ 2.3 ต้นแบบแผนภาพคลาสของแบบรูป Composite....9
รูปที่ 2.4 ต้นแบบแผนภาพคลาสของแบบรูป Decorator....10
รูปที่ 2.5 ต้นแบบแผนภาพคลาสของแบบรูป Façade....11
รูปที่ 2.6 ต้นแบบแผนภาพคลาสของแบบรูป Flyweight....12
รูปที่ 2.7 ต้นแบบแผนภาพคลาสของแบบรูป Proxy....13
รูปที่ 2.8 ตัวอย่างของรูปแบบมาตราฐานเอกซ์เอ็มไอ....16
รูปที่ 2.9 ตัวอย่างความคล้ายคลึงกันระหว่างกราฟ G1, G2 และ G3....17
รูปที่ 2.10 การแปลงแผนภาพคลาสเป็นโมเดลกราฟ....21
รูปที่ 3.1 ตัวอย่างแผนภาพคลาส....27
รูปที่ 3.2 โครงสร้างแผนภาพแบบรูปการออกแบบ Adapter....29
รูปที่ 3.3 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Adapter....30
รูปที่ 3.4 โครงสร้างแผนภาพแบบรูปการออกแบบBridge....31
รูปที่ 3.5 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Bridge....32
รูปที่ 3.6 โครงสร้างแผนภาพแบบรูปการออกแบบ Composite....33
รูปที่ 3.7 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Composite....34
รูปที่ 3.8 โครงสร้างแผนภาพแบบรูปการออกแบบ Decorator....35
รูปที่ 3.9 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Decorator....36
รูปที่ 3.10 โครงสร้างแผนภาพแบบรูปการออกแบบ Façade....37
รูปที่ 3.11 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Façade....38
รูปที่ 3.12 โครงสร้างแผนภาพแบบรูปการออกแบบ Flyweight....39
รูปที่ 3.13 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Flyweight....40
รูปที่ 3.14 โครงสร้างแผนภาพแบบรูปการออกแบบ Proxy....41
รูปที่ 3.15 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Proxy....42
รูปที่ 3.16 โหนดเริ่มต้นของแบบรูปการออกแบบ Adapter....50
รูปที่ 3.17 การตรวจสอบโหนดที่เกี่ยวของของแบบรูปการออกแบบ Adapter....50
รูปที่ 3.18 โหนดเริ่มต้นของแบบรูปการออกแบบ Bridge....51

หน้า

รูปที่ 3.19 โหนดเริ่มต้นของแบบรูปการออกแบบ Composite....51
รูปที่ 3.20 โหนดเริ่มต้นของแบบรูปการออกแบบ Decorator....52
รูปที่ 3.21 โหนดเริ่มต้นของแบบรูปการออกแบบ Façade....52
รูปที่ 3.22 โหนดเริ่มต้นของแบบรูปการออกแบบ Flyweight....53
รูปที่ 3.23 โหนดเริ่มต้นของแบบรูปการออกแบบ Proxy....53
รูปที่ 3.24 การกำหนดความเหมือนของแบบรูปการออกแบบ Adapter....54
รูปที่ 3.25 การกำหนดความเหมือนของแบบรูปการออกแบบ Bridge....55
รูปที่ 3.26 การกำหนดความเหมือนของแบบรูปการออกแบบ Composite....55
รูปที่ 3.27 การกำหนดความเหมือนของแบบรูปการออกแบบ Decorator....56
รูปที่ 3.28 การกำหนดความเหมือนของแบบรูปการออกแบบ Façade....56
รูปที่ 3.29 การกำหนดความเหมือนของแบบรูปการออกแบบ Flyweight....57
รูปที่ 3.30 การกำหนดความเหมือนของแบบรูปการออกแบบ Proxy....57
รูปที่ 3.31 ตัวอย่างแผนภาพคลาสที่ใช้ในการตรวจจับแบบรูปการออกแบบ....58
รูปที่ 3.32 แผนภาพกราฟเชิงความสัมพันธ์ของแผนภาพคลาสตัวอย่าง....59
รูปที่ 3.33 อัลกอริทึมจากงานวิจัยของ Manjari Gupta และคณะ....61
รูปที่ 3.34 อัลกอริทึมการตรวจจับโหนดอ้างอิงของแต่ละแบบรูปการออกแบบใดๆ....63
รูปที่ 3.35 (ก) อัลกอริทึมตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบใดๆ....64
รูปที่ 3.35 (ข) อัลกอริทึมตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบใดๆ....65
รูปที่ 3.36 ตัวอย่างแผนภาพคลาสระบบสำหรับการตรวจจับแบบรูปการออกแบบ....66
รูปที่ 3.37 โหนดอ้างอิงของแบบรูปการออกแบบ Adapter....68
รูปที่ 3.38 อัลกอริทึมที่ใช้ในการตรวจจับโหนดอ้างอิงแบบรูปการออกแบบ Adapter....68
รูปที่ 3.39 คุณลักษณะท้องถิ่นโหนดอ้างอิงของแบบรูปการออกแบบ Adapter....70
รูปที่ 3.40 คุณลักษณะท้องถิ่นแผนภาพระบบ (LPT-S)....70
รูปที่ 3.41 ความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S)....71
รูปที่ 3.42 คุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P)....71
รูปที่ 3.43 ความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P)....71
รูปที่ 3.44 การเปรียบเทียบ จำนวนเส้นความสัมพันธ์ระหว่างคลาส 12 ลำดับ....72
รูปที่ 3.45 การเปรียบเทียบค่าแต่ละลำดับองค์ประกอบของคุณลักษณะท้องถิ่น....72
รูปที่ 3.46 ผลจากการเปรียบเทียบกับโหนดอ้างอิงแบบรูป Adapter....73
รูปที่ 3.47 เปรียบเทียบคุณลักษณะของโหนดอ้างอิงกับโหนดในตารางคุณลักษณะท้องถิ่น....73

หน้า

รูปที่ 3.48 การใส่ชื่อโหนดอ้างอิง Client ในตาราง LPT-S....74
รูปที่ 3.49 Stack ของโหนดอ้างอิงแบบรูป Adapter....74
รูปที่ 3.50 การตรวจสอบโหนดที่เกี่ยวข้องของแบบรูปการออกแบบ Adapter....75
รูปที่ 3.51 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Target....75

รูปที่ 3.52 ผลลัพธ์การตรวจจับโหนดอ้างออิงแบบรูป Adapter....76
รูปที่ 3.53 (ก) อัลกอริทึมการตรวจจับรายละเอียดแบบรูปการออกแบบ Adapter....76
รูปที่ 3.53 (ข) อัลกอริทึมการตรวจจับรายละเอียดแบบรูปการออกแบบ Adapter....77
รูปที่ 3.54 ข้อมูลนำเข้า Stack S....78
รูปที่ 3.55 คุณลักษณะท้องถิ่นแผนภาพระบบ (LPT-S)....78
รูปที่ 3.56 ความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S)....79
รูปที่ 3.57 คุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P)....79
รูปที่ 3.58 ความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P)....79
รูปที่ 3.59 การค้นหาโหนดปลายทางจาก Stack *S* ซึ่งมีค่าเป็นโหนด Client....80
รูปที่ 3.60 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Currency กับ โหนด Target....80
รูปที่ 3.61 การบันทึกชื่อโหนด Target ลงในตาราง LPT-S....81
รูปที่ 3.62 การจัดเก็บโหนด Currency ลงใน Stack S....81
รูปที่ 3.63 การค้นหาโหนดต้นทางจาก Stack S ซึ่งมีค่าเป็นโหนด Curency....82
รูปที่ 3.64 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Product, Exchange กับ โหนด Adapter....82
รูปที่ 3.65 การบันทึกชื่อโหนด Adapter ลงในตาราง LPT-S....83
รูปที่ 3.66 การจัดเก็บโหนด Product และโหนด Exchange ลงใน Stack S....83
รูปที่ 3.67 การค้นหาโหนดปลายทางจาก Stack S ซึ่งมีค่าเป็นโหนด Product....84
รูปที่ 3.68 การบันทึกชื่อโหนด Adapter ลงในตาราง LPT-S....84
รูปที่ 3.69 ผลลัพธ์การตรวจจับที่มีความเหมือนกับแบบรูปการออกแบบ Adapter....85
รูปที่ 3.70 การค้นหาโหนดปลายทางจาก Stack S ซึ่งมีค่าเป็นโหนด Exchange....86
รูปที่ 3.71 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Current กับ โหนด Adaptee....86
รูปที่ 3.72 การบันทึกชื่อโหนด Adapter ลงในตาราง LPT-S....87
รูปที่ 3.73 Stack *S* มีค่าว่าง....87
รูปที่ 3.74 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ Adapter....88
รูปที่ 4.1 แผนภาพยูสเคสสำหรับการตรวจจับแบบรูปการออกแบบ....89
รูปที่ 4.2 แผนภาพคลาสสำหรับเครื่องมือตรวจจับแบบรูปการออกแบบ....95

หน้า

รูปที่ 4.3 แผนภาพองค์ประกอบของเครื่องมือ DSPT....96
รูปที่ 4.4 แผนภาพกิจกรรมการทำงานของเครื่องมือในการตรวจจับแบบรูปการออกแบบ....99
รูปที่ 5.1 กรณีทดสอบที่หนึ่งสำหรับแผนภาพคลาสของแบบรูปการออกแบบทั้งหมด....101
รูปที่ 5.2 กรณีทดสอบที่หนึ่ง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Adapter....103
รูปที่ 5.3 กรณีทดสอบที่หนึ่ง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Proxy....103
รูปที่ 5.4 กรณีทดสอบที่สองแผนภาพคลาส AtollGame....104

รูปที่ 5.5 กรณีทดสอบที่สอง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Façade.....106
รูปที่ 5.6 กรณีทดสอบที่สอง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Proxy.....107
รูปที่ 5.7 กรณีทดสอบที่สามแผนภาพคลาส Web Portal Caching.....108
รูปที่ 5.8 กรณีทดสอบที่สาม การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Bridge.....110
รูปที่ 5.9 กรณีทดสอบที่สาม การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Proxy.....110
รูปที่ ก.1 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ กรณีทดสอบที่หนึ่ง.....118
รูปที่ ข.1 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ กรณีทดสอบที่สองแผนภาพคลาส AtollGame.....120
รูปที่ ค.1 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ กรณีทดสอบที่สามแผนภาพคลาส Web Portal Caching.....122

บทที่ 1

บทนำ

1.1. ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันการออกแบบพัฒนาระบบซอฟต์แวร์นั้นได้นำแผนภาพยูเอ็มแอล (UML: Unified Modeling Language) มาใช้เป็นมาตรฐานหลักสำหรับการสร้างแบบจำลองเพื่ออธิบายถึงรายละเอียดของระบบงาน ซึ่งมาตรฐานของแผนภาพยูเอ็มแอลนั้นจะเป็นการนำเสนอแนวคิดการพัฒนาเชิงวัตถุ (OOP: Object-Oriented Development) ที่สนับสนุนแนวคิดเชิงวัตถุอย่างแผนภาพคลาส ซึ่งนักออกแบบระบบนิยมใช้ในการอธิบายการทำงานของภาพรวมของโครงสร้างระบบที่แสดงให้เห็นถึงการเรียกใช้งานวัตถุต่าง ๆ การออกแบบระบบควรคำนึงถึงการเปลี่ยนแปลงการของทำงานในอนาคต ที่อาจมีการปรับเพิ่มลดในส่วนของโครงสร้าง หรือการนำไปเชื่อมต่อกับระบบอื่นๆ เพื่อให้การดำเนินการปรับแก้ไข หรือการนำไปใช้งานต่อ การออกแบบระบบนั้นควรรองรับการทำงานดังกล่าวด้วย จากแนวคิดดังกล่าวในการออกแบบระบบจึงมีการนำแบบรูปการออกแบบ (Design Patterns) มาใช้สนับสนุนกระบวนการออกแบบระบบให้กับนักออกแบบระบบ

แบบรูปการออกแบบที่มักนำมาใช้ในการออกแบบระบบ คือ แบบรูปการออกแบบเชิงโครงสร้าง (Structural Design Patterns) ที่เน้นการสืบทอดคุณสมบัติของคลาสมาใช้งาน เพื่อลดความซ้ำซ้อนในการออกแบบระบบ จากการนำแบบรูปการออกแบบเชิงโครงสร้างมาใช้ในการออกแบบระบบ จึงมีหลายงานวิจัยที่คิดขั้นตอนวิธี (Algorithm) มาใช้ในการตรวจจับแบบรูปการออกแบบที่มีอยู่ในแผนภาพคลาส เพื่อให้ทราบถึงตำแหน่งของแบบรูปดังกล่าวที่มีอยู่ และทำให้นักออกแบบระบบเข้าใจถึงโครงสร้างระบบ ซึ่งส่งผลให้นักพัฒนาระบบสามารถปรับแก้ไข หรือเพิ่มเติมการทำงานของระบบได้ตรงจุด

ขั้นตอนวิธีการในการตรวจจับแบบรูปการออกแบบของงานวิจัยต่างๆ ตามที่ได้ศึกษาข้อมูลนั้น มีอยู่หลายวิธีการด้วยกัน ตัวอย่างเช่น วิธีการตรวจจับด้วยการคำนวณหาค่าคะแนนความคล้ายคลึง (Similarity Scoring Algorithm) [1] การตรวจจับแบบรูปการออกแบบด้วยการใช้แผ่นแบบการจับคู่ (Template Matching) [2] การตรวจจับแบบรูปการออกแบบจากแผนภาพกราฟ (Inexact Graph Matching) [3] วิธีการนำกราฟไมนิง (Mining Graph) ใช้ในการตรวจจับความเหมือนกันของการออกแบบเชิงโครงสร้างที่อยู่ในมุมมองระดับสูงของสถาปัตยกรรมการออกแบบซอฟต์แวร์เชิงวัตถุ (Object-oriented) [4] จากงานวิจัยที่ได้อ้างอิงวิธีการตรวจจับแบบรูปการ

ออกแบบมานั้น ทางผู้วิจัยได้เล็งเห็นถึงข้อบกพร่องของงานวิจัยการตรวจจับแบบรูปการออกแบบจากแผนภาพกราฟ [3] ที่ได้มีการนำเสนออัลกอริทึมที่ใช้ในการตรวจจับแบบรูปการออกแบบของเส้นความสัมพันธ์ระหว่างโหนด (Node) ที่อยู่ในรูปแบบของแผนภาพคลาส ซึ่งยังไม่ครอบคลุมลักษณะของเส้นความสัมพันธ์ที่มีอยู่ในมาตรฐานของยูเอ็มแอล (UML)

ดังนั้น งานวิจัยนี้จึงได้นำเสนอการปรับปรุงอัลกอริทึมที่มุ่งเน้นไปยังการตรวจจับการปรับปรุงอัลกอริทึมการตรวจจับแบบรูปการออกแบบเชิงโครงสร้างด้วยกราฟ โดยเพิ่มเติมการกำหนดจุดเชื่อมต่อจากเดิม 3 ลำดับ [3] เป็น 12 ลำดับ เพื่อให้ครอบคลุมลักษณะของเส้นความสัมพันธ์ตามมาตรฐานของยูเอ็มแอลและเพิ่มเติมการคำนวณค่าความเหมือนกันของแบบรูปการออกแบบเชิงโครงสร้างที่ปรากฏอยู่ในการออกแบบระบบซอฟต์แวร์พร้อมแสดงข้อแนะนำถึงการออกแบบที่ถูกต้องตรงตามแบบรูปการออกแบบเชิงโครงสร้างที่กำหนดทั้งหมด 7 รูปแบบ โดยการนำเข้าข้อมูลไฟล์เอกซ์เอ็มไอ (XMI) ที่อยู่ในรูปแบบของกราฟและนำมาเปรียบเทียบหาเส้นความสัมพันธ์ของคลาสที่สอดคล้องกันตรวจจับแบบรูปการออกแบบเชิงโครงสร้างที่ปรากฏอยู่ต่อไปซึ่งจะเป็นการปรับปรุงอัลกอริทึมในการตรวจจับที่มีอยู่แล้วให้มีประสิทธิภาพยิ่งขึ้น อีกทั้งยังทำการพัฒนาเครื่องมือตามการปรับปรุงอัลกอริทึมที่งานวิจัยนี้ได้เสนอขึ้น และรายงานสรุปผลการตรวจจับแบบรูปการออกแบบที่ตรวจพบให้นักออกแบบระบบได้ทราบเพื่อใช้พิจารณาในการปรับแก้ไขการออกแบบให้ถูกต้องยิ่งขึ้น

1.2. วัตถุประสงค์การวิจัย

เพื่อปรับปรุงขั้นตอนวิธีการและพัฒนาเครื่องมือการตรวจจับแบบรูปการออกแบบเชิงโครงสร้างโดยใช้กราฟเชิงความสัมพันธ์

1.3. ขอบเขตของงานวิจัย

1) ข้อมูลนำเข้าของงานวิจัยนี้ คือ ไฟล์เอกซ์เอ็มไอ รุ่น 2.1 ที่มีความถูกต้องในการออกแบบของแผนภาพคลาสที่จะประกอบด้วยชื่อคลาส ชนิด ชื่อเมธ็อด (Method) และระบุเส้นความสัมพันธ์ระหว่างคลาสเป็นอย่างน้อยที่มีความถูกต้องตรงตามหลักวากยสัมพันธ์ (Syntax) ของมาตรฐานยูเอ็มแอล
2) การปรับปรุงขั้นตอนวิธีการตรวจจับแบบรูปการออกแบบโดยใช้กราฟเชิงความสัมพันธ์
3) การตรวจจับแบบรูปการออกแบบเชิงโครงสร้างจะครอบคลุม ดังต่อไปนี้
    - แบบรูปการออกแบบ Adapter
    - แบบรูปการออกแบบ Bridge

- แบบรูปการออกแบบ Composite
- แบบรูปการออกแบบ Decorator
- แบบรูปการออกแบบ Facade
- แบบรูปการออกแบบ Flyweight
- แบบรูปการออกแบบ Proxy

4) พัฒนาเครื่องมือสำหรับการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง โดยสามารถรายงานผลการตรวจจับแบบรูปการออกแบบได้ ดังนี้
   - สามารถระบุชนิดของแบบรูปการออกแบบเชิงโครงสร้างในตรวจจับในกราฟของระบบได้
   - บอกจำนวนของแบบรูปการออกแบบเชิงโครงสร้างที่ตรวจพบ
   - แสดงผลค่าร้อยละความเหมือนของตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง
5) การตรวจสอบความแม่นยำของเครื่องมือ จากข้อมูล 2 ชุด คือ ข้อมูลที่สร้างขึ้นเองที่ออกแบบให้มีแบบรูปการออกแบบเชิงโครงสร้างปรากฏอยู่ทั้ง 7 รูปแบบ ตามข้อ 3) จำนวน 1 ระบบ และข้อมูลจากการออกแบบระบบจริง จำนวน 2 ระบบ

1.4. ขั้นตอนและวิธีดำเนินการวิจัย

1) ศึกษาข้อมูลวิธีการ/ขั้นตอน การตรวจจับแบบรูปการออกแบบและงานวิจัยที่เกี่ยวข้อง
2) ศึกษาความหมายของสัญลักษณ์ต่าง ๆ ของแผนภาพคลาสตามมาตรฐานของยูเอ็มแอล
3) ศึกษาแบบรูปการออกแบบเชิงโครงสร้างตามคุณลักษณะของแต่ละแบบรูป เพื่อมาใช้งานการสร้างข้อกำหนดในการตรวจจับแบบรูปการออกแบบในระบบซอฟต์แวร์
4) ออกแบบขั้นตอน/อัลกอริทึม สำหรับการตรวจจับแผนภาพการออกแบบเชิงโครงสร้างสำหรับแต่ละแบบรูปการออกแบบ
5) ออกแบบเครื่องมือการทดสอบการตรวจจับแผนภาพการออกแบบโครงสร้างและทดลองพัฒนาเครื่องมือตามขั้นตอนที่ออกแบบไว้
6) ทดสอบการใช้งานของเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้างตามขอบเขตการดำเนินงานวิจัย
7) สรุปผลงานวิจัยและข้อเสนอแนะ
8) จัดทำเอกสารวิทยานิพนธ์ฉบับสมบูรณ์

1.5. ผลงานที่ได้รับการตีพิมพ์

งานวิจัยในหัวข้อ “Detection of Design Patterns in Software Design Model using Graph” ผู้แต่งโดย คือ นางสาวชลิดา เหลี่ยมวิเศษ และรองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ ซึ่งได้รับการคัดเลือกให้นำเสนอในงานประชุมวิชาการระดับนานาชาติ “The 2013 2nd International Conference on Information Technology and Management Innovation (ITMI 2013)” ซึ่งจัดขึ้น ณ เมืองจูไฮ่ สาธารณรัฐประชาชนจีน ระหว่างวันที่ 23-24 กรกฎาคม พุทธศักราช 2556 ในเอกสารประกอบการประชุมวิชาการหน้าที่ 559-562

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1. ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องกับงานวิจัยนี้ประกอบไปด้วยเรื่อง แบบรูปการออกแบบเพื่อการสร้าง แบบรูปการออกแบบเชิงโครงสร้าง แบบรูปการออกแบบพฤติกรรม เอกซ์เอ็มไอ และการจับคู่ความสัมพันธ์ของกราฟ ซึ่งมีรายละเอียดดังนี้

2.1.1. แบบรูปการออกแบบ (Design Pattern) [2]

แบบรูปการออกแบบเป็นการนิยามความเชื่อมโยงของคลาสต่างๆ ที่นำมาใช้เป็นแนวทางการแก้ปัญหาที่เกิดขึ้นซ้ำ ๆ ของกระบวนการการออกแบบซอฟต์แวร์ ซึ่งแบบรูปการออกแบบนี้จะอธิบายแนวทางการนำไปประยุกต์ใช้ในสถานการณ์ต่าง ๆ ที่แตกต่างกันไปของการออกแบบซอฟต์แวร์ การนำไปใช้เป็นหลักในการพัฒนาโปรแกรมเชิงวัตถุ (Object-oriented programming: OOP) โดยทั่วไปแบบรูปการออกแบบจะประกอบไปด้วย 4 ส่วนสำคัญ ได้แก่

1) ชื่อแบบรูปการออกแบบ (Pattern name) ระบุถึงปัญหาของการออกแบบแนวทางแก้ไขและผลลัพธ์ เพื่อให้ผู้ใช้งานทราบถึงการนำไปประยุกต์ใช้งานกับปัญหาที่เกิดขึ้นในการออกแบบซอฟต์แวร์
2) ปัญหา (Problem) จะอธิบายรายละเอียดของปัญหาว่าเมื่อใด จะต้องมีการนำใช้แบบรูปการออกแบบมาใช้
3) แนวทางแก้ไขปัญหา (Solution) เป็นการอธิบายส่วนประกอบที่ใช้ในการออกแบบความสัมพันธ์ หน้าที่ความรับผิดชอบและการทำงานร่วมกันของส่วนประกอบต่าง
4) ผลที่ตามมา (Consequences) ผลที่เกิดขึ้นเมื่อมีการนำแบบรูปการออกแบบไปใช้แก้ปัญหา

การจัดหมวดหมู่ของแบบรูปการออกแบบ สามารถแบ่งได้เป็น 3 กลุ่ม ตามวัตถุประสงค์การใช้งานคือ แบบรูปการออกแบบเพื่อการสร้าง แบบรูปการออกแบบเชิงโครงสร้างและแบบรูปการออกแบบพฤติกรรม รายละเอียดของแต่ละกลุ่ม ดังนี้

2.1.1.1. แบบรูปการออกแบบเพื่อการสร้าง (Creational design patterns)

แบบรูปการออกแบบเพื่อการสร้างนั้นให้ความสำคัญกับการจัดการวัตถุ (Object) ที่มีความขึ้นต่อกันของการสร้างวัตถุ การกำหนดองค์ประกอบและการสร้างวัตถุที่ทำหน้าที่เป็นตัวแทน ซึ่งวัตถุเหล่านี้จะมีการสืบทอดคุณสมบัติต่าง ๆ จากหลากหลายคลาสไปให้กับวัตถุอื่น ๆ ที่สามารถใช้งานได้เช่นเดียวกัน แบบรูปการออกแบบเพื่อการสร้างนั้นจะประกอบไปด้วย

1) แบบรูป Abstract Factory เป็นการจัดการอินเตอร์เฟสที่มีความสัมพันธ์หรือการขึ้นต่อกันของกลุ่มวัตถุ โดยไม่ต้องระบุให้อยู่ในคลาสรูปธรรม (Concrete class) สามารถนำไปใช้เมื่อระบบต้องการสร้างส่วนประกอบต่าง ๆ ที่ไม่มีความขึ้นต่อกัน

2) แบบรูป Builder เป็นการแยกโครงสร้างของวัตถุที่มีความซับซ้อนออกจากตัวแทนของวัตถุที่ยังคงคุณลักษณะเหมือนเดิมแต่มีตัวแทนที่ต่างออกไป การนำไปใช้เมื่อมีการขึ้นต่อกันของอัลกอริทึมสำหรับการสร้างวัตถุที่ซับซ้อน

3) แบบรูป Factory Method การกำหนดอินเตอร์เฟสเพื่อการสร้างวัตถุ แต่คลาสย่อยจะเป็นตัวกำหนดในการสร้างวัตถุให้กับคลาสใด ๆ จะนำไปใช้เมื่อคลาสไม่สามารถกำหนดได้ว่าจะมีการสร้างวัตถุ หรือจะต้องมีคลาสย่อยเพื่อทำหน้าที่ในการสร้างวัตถุ

4) แบบรูป Object Pool การรวมกันของวัตถุเพื่อเพิ่มประสิทธิภาพการทำงานของคลาสเพื่อให้สามารถเรียกการทำงานจำนวนมากไปยังคลาสสามารถทำงานได้ ซึ่งจะเน้นการนำวัตถุเดิมมาใช้งาน

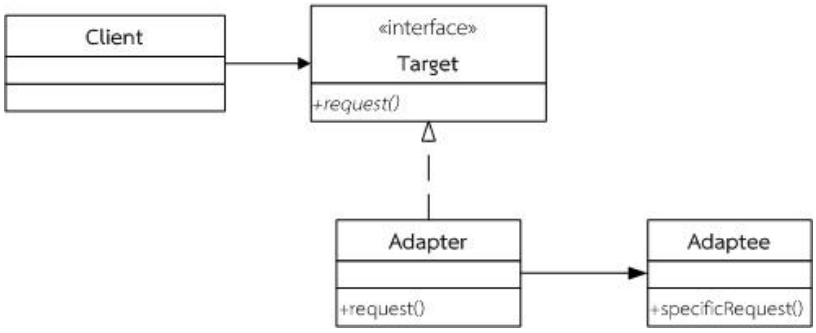
5) แบบรูป Prototype การสร้างวัตถุตามวัตถุต้นแบบที่มีการกำหนดคุณลักษณะและวิธีการทำงานของวัตถุนั้นไว้แล้ว

6) แบบรูป Singleton การสร้างคลาสตัวแทนเพียงคลาสเดียวเพื่อสำหรับการเรียกใช้งานจากคลาสอื่นๆ ผ่านช่องทางที่วัตถุทุกตัวสามารถเข้าใช้งานได้เสมอ ข้อดีคือการเรียกใช้งานผ่านคลาสตัวแทนไม่ต้องแก้ไขโค้ดบ่อย สามารถควบคุมการเข้าใช้งานจากคลาสอื่นๆ ได้

2.1.1.2. แบบรูปการออกแบบเชิงโครงสร้าง (Structural design patterns)

แบบรูปการออกแบบเชิงโครงสร้างนั้นเกี่ยวข้องกับคลาสและองค์ประกอบของวัตถุโดยการใช้การสืบทอดไปยังส่วนของคลาสอินเตอร์เฟสและการกำหนดวิธีการนำวัตถุมาประกอบกันเพื่อให้ได้การทำงานรูปแบบใหม่ ๆ แบบรูปการออกแบบถูกใช้ในการอธิบายการสื่อสารระหว่างคลาสในการแก้ปัญหาที่เกิดขึ้นของการออกแบบซึ่งจะช่วยให้การออกแบบมีโครงสร้างที่ดีและสามารถปรับเปลี่ยนได้รวมถึงการนำการออกแบบมาใช้ใหม่ในการพัฒนาซอฟต์แวร์แบบรูปการออกแบบเชิงโครงสร้างจะประกอบไปด้วย

1) แบบรูป Adapter เป็นการแปลงอินเตอร์เฟสของคลาสไปยังคลาสอื่น ๆ เพื่อให้ผู้ใช้งานสามารถเรียกใช้งานพร้อมกันได้จากคลาสที่มีความแตกต่างกัน ให้สามารถทำงานร่วมกันได้ การนำไปใช้เมื่ออินเตอร์เฟสนั้นไม่สัมพันธ์กับคลาสที่จะเรียกใช้งาน

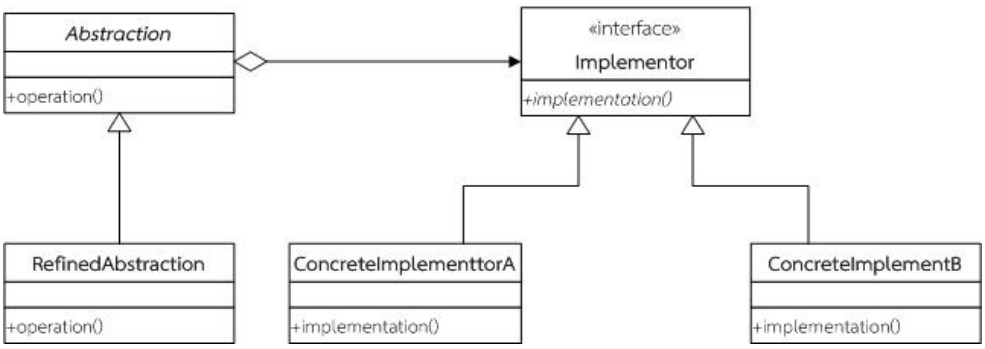


รูปที่ 2.1 ต้นแบบแผนภาพคลาสของแบบรูป Adapter [2]

คำอธิบายองค์ประกอบแบบรูป Adapter ที่แสดงในรูปที่ 2.1

- คลาส Target กำหนดคุณลักษณะเฉพาะของอินเตอร์เฟสให้กับผู้ใช้งาน
- คลาส Adapter การปรับเปลี่ยนอินเตอร์เฟสตัวแปลงไปยังอินเตอร์เฟสของคลาสTarget
- คลาส Adaptee กำหนดอินเตอร์เฟสที่ใช้งานตามความต้องการในการปรับเปลี่ยน
- คลาส Client การรวบรวมวัตถุที่เกี่ยวข้องในการใช้งานอินเตอร์เฟสของคลาส Target

2) แบบรูป Bridge แยกวัตถุของคลาสนามธรรมออกจากส่วนของการอิมพลีเมนท์เพื่อให้เกิดการทำงานอย่างอิสระยิ่งขึ้น

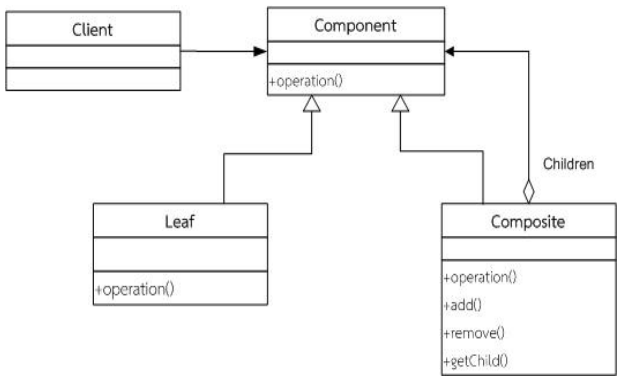


รูปที่ 2.2 ต้นแบบแผนภาพคลาสของแบบรูป Bridge [2]

คำอธิบายองค์ประกอบแบบรูป Bridge ที่แสดงในรูปที่ 2.2

- คลาส Abstraction เพื่อกำหนดนามธรรมของอินเตอร์เฟซด้วยการอ้างอิงไปยังวัตถุประเภทที่ใช้งาน
- คลาส RefinedAbstraction เพื่อสืบทอดการทำงานของอินเตอร์เฟสที่ถูกกำหนดโดยคลาสนามธรรม
- คลาส Implementor ใช้กำหนดอินเตอร์เฟสสำหรับการนำไปใช้งานของคลาส ซึ่งอินเตอร์เฟสจะไม่สอดคล้องโดยตรงกับของอินเตอร์เฟสนามธรรม โดยที่ทั้งสองอินเตอร์เฟสนี้มีความแตกต่างกัน ตามปกติแล้วการดำเนินการของอินเตอร์เฟสจะขึ้นกับโอเปอร์เรชันที่เริ่มต้นเท่านั้นและนามธรรมที่กำหนดอยู่ในการดำเนินการระดับสูงซึ่งจะเป็นจุดเริ่มต้นให้กับการดำเนินการต่อไป
- คลาส ConcreteImplementor เป็นคลาสที่การเรียกใช้งานคลาสอินเตอร์เฟสของคลาส Implementor และการกำหนดการใช้งานตามรูปธรรม

3) แบบรูป Composite เป็นวัตถุที่รวมกันกลายเป็นโครงสร้างต้นไม้ที่แสดงถึงส่วนของลำดับชั้นการทำงาน

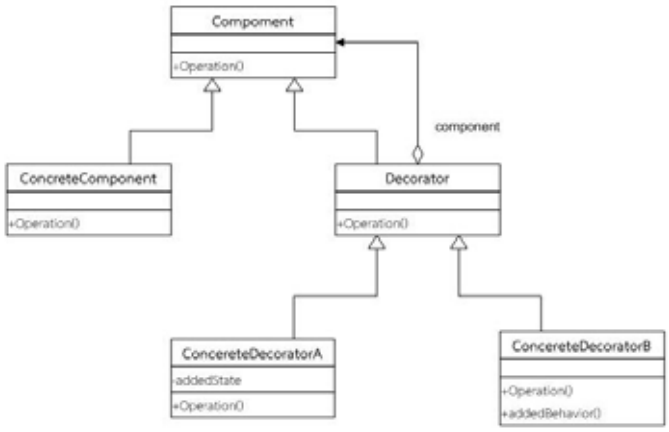


รูปที่ 2.3 ต้นแบบแผนภาพคลาสของแบบรูป Composite [2]

คำอธิบายองค์ประกอบแบบรูป Composite ที่แสดงในรูปที่ 2.3

- คลาส Component ประกาศอินเตอร์เฟสสำหรับวัตถุที่เป็นองค์ประกอบให้สามารถเรียกใช้งานส่วนของพฤติกรรมจากอินเตอร์เฟสของคลาสต่าง ๆ ที่มีเหมาะสมได้ เพื่อการเข้าถึงองค์ประกอบของคลาสสืบทอดในส่วนของโครงสร้างที่ถูกใช้งานซ้ำๆ ได้อย่างเหมาะสม
- คลาส Leaf นำเสนอการรวมกันของคลาสสืบทอดลำดับสุดท้ายของวัตถุซึ่งคลาสสืบทอดลำดับสุดท้ายจะไม่มีคลาสลูกในการสืบทอดอีก และยังเป็นคลาสที่กำหนดพฤติกรรมเริ่มต้นของวัตถุในส่วนประกอบ
- คลาส Composite การกำหนดพฤติกรรมสำหรับองค์ประกอบของคลาสลูก ซึ่งใช้ในการจัดเก็บองค์ประกอบของคลาสลูกและนำความสัมพันธ์ของคลาสลูกมาดำเนินการในองค์ประกอบของอินเตอร์เฟส
- คลาส Client ปรับแต่งวัตถุในส่วนประกอบผ่านทางคอมโพเนนท์อินเตอร์เฟสคลาส

4) แบบรูป Decorator การเพิ่มหน้าที่ความรับผิดชอบไปยังวัตถุให้มีทางเลือกในการทำงานที่ยืดหยุ่นยิ่งขึ้นของคลาสย่อยที่มีการสืบทอดหน้าที่การทำงานนั้น

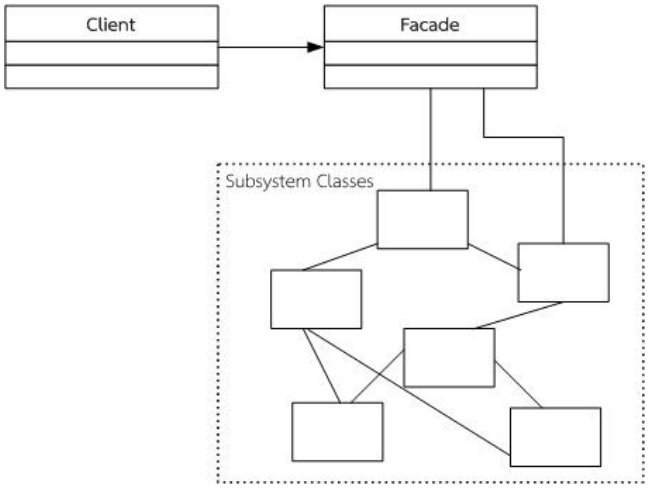


รูปที่ 2.4 ต้นแบบแผนภาพคลาสของแบบรูป Decorator [2]

คำอธิบายองค์ประกอบแบบรูป Decorator ที่แสดงในรูปที่ 2.4

- คลาส Component การกำหนดอินเตอร์เฟสของวัตถุที่สามารถเพิ่มหน้าที่ความรับผิดชอบได้อย่างยืดหยุ่น
- คลาส ConcreteComponent การกำหนดวัตถุด้วยการเพิ่มเติมหน้าที่ความรับผิดชอบที่สามารถถูกเพิ่มไปได้ขึ้นเรื่อย ๆ
- คลาส Decorator การปรับแต่งที่อ้างอิงวัตถุของคอมโพเนนท์และการกำหนดอินเตอร์เฟสที่สอดคล้องของอินเตอร์เฟสคอมโพเนนท์นั้นด้วย
- คลาส ConcreteDecorator การเพิ่มหน้าที่ความรับผิดชอบไปยังคอมโพเนนท์

5) แบบรูป Facade เป็นการทำงานของคลาสหลักที่มีหน้าที่รวบรวมการเข้าถึงการใช้งานจากคลาสลูกๆ

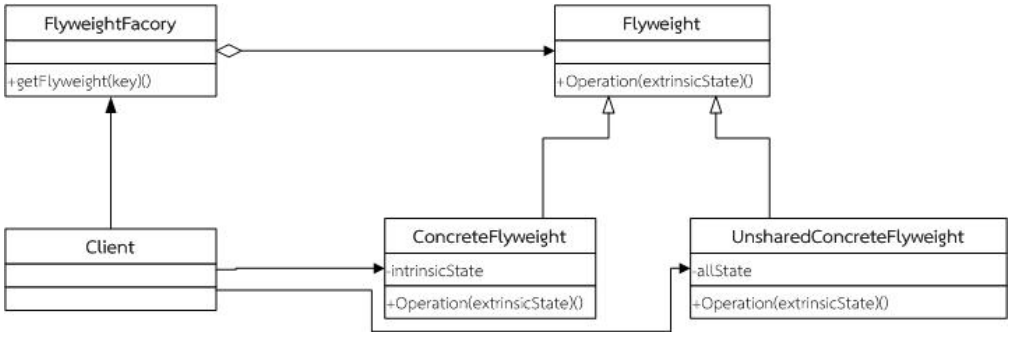


รูปที่ 2.5 ต้นแบบแผนภาพคลาสของแบบรูป Façade [2]

คำอธิบายองค์ประกอบแบบรูป Façade ที่แสดงในรูปที่ 2.5

- คลาส Façade มีหน้าที่ในการรับผิดชอบการเรียกใช้งานจากคลาสย่อยๆและตอบสนองการเรียกใช้งานตามความเหมาะสม
- คลาส Subsystem การจัดการงานที่ได้รับมอบหมายวัตถุของคลาส Façade

6) แบบรูป Flyweight ใช้ในการแบ่งปันวัตถุจำนวนมากให้ทำงานอย่างมีประสิทธิภาพและมีความมั่นคง

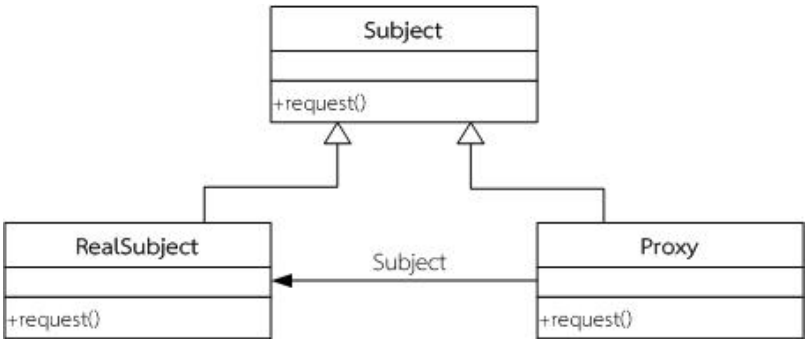


รูปที่ 2.6 ต้นแบบแผนภาพคลาสของแบบรูป Flyweight [2]

คำอธิบายองค์ประกอบของแบบรูป Flyweight ที่แสดงในรูปที่ 2.6

- คลาส Flyweight การประกาศอินเตอร์เฟสที่สามารถรับและดำเนินการกับสถานะภายนอกได้
- คลาส ConcreteFlyweight การใช้งานอินเตอร์เฟสจากคลาส Flyweight และการเพิ่มการจัดเก็บสถานะซึ่งจะต้องมีบริบทเป็นอิสระจากวัตถุของคลาสรูปธรรม Flyweight
- คลาส UnshareConcreteFlyweight กำหนดการทำงานของคลาสย่อยในคลาส Flyweight ที่ไม่จำเป็นจะต้องถูกแบ่งปันการใช้งาน
- คลาส FlyweightFactory สร้างและจัดการวัตถุของคลาส Flyweight
- คลาส Client การปรับแต่งการอ้างอิงของคลาส Flyweight คำนวณและจัดเก็บสถานะภายนอกของคลาส Flyweight

7) แบบรูป Proxy ตัวแทนการจัดการควบคุมการเข้าถึงจากวัตถุอื่น ๆ



รูปที่ 2.7 ต้นแบบแผนภาพคลาสของแบบรูป Proxy [2]

คำอธิบายองค์ประกอบของแบบรูป Proxy ที่แสดงในรูปที่ 2.7

- คลาส Proxy การอ้างอิงการเข้าถึงคลาส Proxy จริงซึ่งอาจมีการอ้างถึงคลาส RealSubject และ คลาส Subject อินเตอร์เฟสการจัดการอินเตอร์เฟสที่เหมือนกันกับคลาส Subject ดังนั้น คลาส Proxy สามารถแทนได้จาก RealSubject ควบคุมการเข้าถึงของคลาส RealSubject และสามารถตอบสนองการสร้างและการลบ realSubject ความสามารถอื่น ๆ ที่ขึ้นอยู่กับชนิดของคลาส Proxy ดังนี้
  - ชนิด remote proxies ที่สนับสนุนการเข้ารหัสของการร้องขอและการส่งไปยังวัตถุจริงที่แตกต่างของพื้นที่อยู่
  - ชนิด virtual proxies การแคชข้อมูลเพิ่มเติมเกี่ยวกับวัตถุจริงที่สามารถเข้าถึงได้
  - ชนิด protection proxies ตรวจสอบการเรียกสิทธิ์การเข้าถึงที่มีการร้องขอ
- คลาส Subject กำหนดอินเตอร์เฟสทั่วไปสำหรับ คลาส RealSubject และ คลาส Proxy ดังนั้น คลาส Proxy จึงสามารถใช้งานได้ทุกที่
- คลาส RealSubject กำหนดวัตถุจริงที่แทนคลาส Proxy

2.1.1.3.แบบรูปการออกแบบพฤติกรรม (Behavioral design patterns)

แบบรูปการออกแบบพฤติกรรมเป็นการกำหนดหน้าที่ความรับผิดชอบให้กับวัตถุที่มีการสื่อสารระหว่างกัน มีการสืบทอดพฤติกรรมระหว่างคลาสและการกระจายพฤติกรรมไปยังคลาสอื่น แบบรูปการออกแบบพฤติกรรมยังอธิบายลักษณะการควบคุมการไหลของการทำงานที่ซับซ้อนซึ่งยากในการติดตามขณะดำเนินการโปรแกรม (run-time) แต่มีความยืดหยุ่นในการทำงานร่วมกันระหว่างวัตถุ แบบรูปการออกแบบพฤติกรรมจะประกอบไปด้วย

1) แบบรูป Chain of responsibility เป็นแบบรูปที่แก้ไขปัญหาการเชื่อมต่อระหว่างผู้ร้องขอความต้องการ (Sender request) ไปยังผู้รับ (Receiver) โดยการกำหนดให้วัตถุหลาย ๆ ตัวจัดการสับเปลี่ยนการร้องที่มีการทำงานในลักษณะของห่วงโซ่ (Chain) สำหรับจัดการคำร้องขอระหว่างวัตถุต่าง ๆ

2) แบบรูป Command การใช้หลักการห่อหุ้ม (Encapsulate) คำสั่งเพื่อซ่อนรายละเอียดการทำงานของวัตถุนั้น ในการส่งการร้องขอของวัตถุจะมีค่าพารามิเตอร์ที่แตกต่างกันโดยจะมีวัตถุที่จัดการคำร้องขอตามลำดับ แบบรูปนี้สนับสนุนการยกเลิกคำสั่งและการย้อนกลับการทำงานในการดำเนินงาน

3) แบบรูป Interpreter เป็นการทำงานที่เกี่ยวข้องกับการแปลทางภาษาและการแทนค่าของประโยคด้วยภาษาสัญลักษณ์ (Symbol) หรือการพิจารณาจากบริบท (Context) อย่างใดอย่างหนึ่งให้เป็นให้สัญลักษณ์อีกแบบหนึ่ง

4) แบบรูป Iterator เป็นแบบแผนการจัดลำดับการเข้าถึงการรวมกลุ่มของวัตถุ โดยไม่ต้องเปิดเผยโครงสร้างภายในของวัตถุนั้น การใช้งานที่สนับสนุนการใช้งานอย่างทั่วถึงในกลุ่มของการรวมวัตถุที่มีโครงสร้างที่แตกต่างกันให้สามารถเรียกใช้งานได้

5) แบบรูป Mediator เป็นแบบรูปที่กำหนดการปฏิสัมพันธ์อย่างง่ายระหว่างคลาส โดยใช้การจัดการของตัวกลาง (Media) ที่มีการจัดการอย่างเป็นอิสระ การใช้งานแบบรูปนี้เหมาะสมกับกรณีที่การสื่อสารระหว่างวัตถุมีความซับซ้อนและมีจำนวนของวัตถุมากหรือการนำวัตถุนั้นกลับมาใช้งานใหม่สามารถทำได้ยาก

6) แบบรูป Memento เป็นแบบรูปที่มีการป้องกันการเข้าถึงการใช้งานของสถานะภายใน (Internal State) ของวัตถุในแต่ละช่วงเวลา ซึ่งการใช้งานวัตถุนั้นสามารถคืนสภาพการใช้งานได้ใหม่ภายหลังสถานะนั้น

7) แบบรูป Observer เป็นช่องทางการเปลี่ยนแปลงการแจ้งเตือนจากคลาสหนึ่งไปยังคลาสอื่น ๆ (one-to-many) ที่มีความขึ้นต่อกันของวัตถุ เมื่อมีการเปลี่ยนแปลงของ วัตถุหนึ่ง วัตถุอื่น ๆ ที่เกี่ยวข้องจะถูกแจ้งการเปลี่ยนแปลงและทำการปรับปรุงตามวัตถุนั้นไปโดยอัตโนมัติ (Auto Update)

8) แบบรูป State แบบรูปนี้เป็นการจัดการการเปลี่ยนแปลงพฤติกรรมของวัตถุ เมื่อเกิดการเปลี่ยนแปลงจากสถานะภายใน วัตถุนั้นจะแสดงการเปลี่ยนแปลงไปด้วย จะนำไปใช้เมื่อพฤติกรรมของวัตถุมีสถานะที่ขึ้นต่อกันหรือเมื่อมีคำสั่งการทำงานที่หลากหลายที่เกี่ยวข้องกับวัตถุนั้น

9) แบบรูป Strategy เป็นการห่อหุ้มขั้นตอนที่แตกต่างกันไว้ภายในคลาสเดียวกัน ซึ่งสามารถเปลี่ยนแปลงการทำงานกันได้ การนำไปใช้งานเมื่อความสัมพันธ์ของคลาสมีจำนวนมากมายแต่มีความต่างในส่วนของพฤติกรรม

10) แบบรูป Template method การกำหนดแม่แบบวิธีขั้นตอนของการดำเนินงานที่เป็นไปตามลำดับขั้นตอนของคลาสย่อย โดยไมมีการเปลี่ยนแปลงขั้นตอนการทำงานของคลาสนั้น จะถูกนำไปใช้เมื่อส่วนของขั้นตอนหนึ่งขั้นตอนใดไม่มีการเปลี่ยนแปลง หรือเมื่อพฤติกรรมระหว่างคลาสย่อยมีส่วนที่ซ้ำซ้อนกัน

11) แบบรูป Visitor เป็นการนำเสนอการดำเนินงานของส่วนประกอบโครงสร้างของวัตถุในการกำหนด การดำเนินการใหม่จะไม่มีการเปลี่ยนแปลงส่วนประกอบที่อยู่ในคลาสนั้น จะนำแบบแผนนี้มาใช้เมื่อคลาสประกอบด้วยหลาย ๆ อินเตอร์เฟสที่แตกต่างกันหรือไม่สัมพันธ์กันของการดำเนินการของวัตถุภายในโครงสร้างของวัตถุ

2.1.2. เอกซ์เอ็มไอ (XMI) [6]

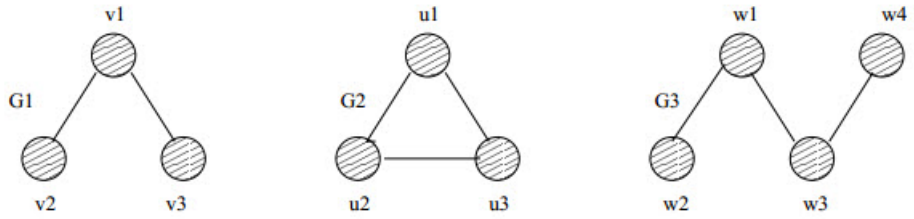
เอกซ์เอ็มไอ (XML Metadata Interchange: XMI) เป็นมาตรฐานขององค์กรโอเอ็มจี (OMG) สำหรับการแลกเปลี่ยนข้อมูลเมทาดาตา(metadata)และอธิบายโครงสร้างการจัดเก็บข้อมูลของยูเอ็มแอลโมเดล ผ่านทางภาษาเอกซ์เอ็มแอล (Extensible Markup Language: XML) ซึ่งเป็นภาษาเบื้องต้นที่ใช้นิยามมาตรฐานของเอกซ์เอ็มไอ ที่ประกอบไปด้วยมาตรฐานสำคัญ ได้แก่ ภาษาเอกซ์เอ็มแอล ยูเอ็มแอล และเอ็มโอเอฟ (MOF) เอกซ์เอ็มไอมีจุดมุ่งหมายที่ช่วยในการแลกเปลี่ยนรูปแบบระหว่างเครื่องมือการสร้างแบบจำลองด้วยภาษายูเอ็มแอลให้นักพัฒนาระบบสามารถใช้การออกแบบของภาษายูเอ็มแอลที่มีความแตกต่างกันของเครื่องมือที่ใช้ในการพัฒนาเพื่อให้สามารถแลกเปลี่ยนข้อมูลไปยังผู้พัฒนาระบบคนอื่น ๆ ได้ ตัวอย่างรูปแบบภาษาเอกซ์เอ็มแอลที่อยู่ในรูปแบบมาตราฐานเอกซ์เอ็มไอ ดังรูปที่ 2.8 [9]

```
<?xml version='1.0' encoding='UTF-8'?>
<xmi:XMI xmlns:xmi="http://schema.omg.org/spec/XMI/2.0"
         xmlns:xlink="http://www.w3.org/1999/Xlink"
         version='2.0' timestamp='Sun, 06 Mar 2005 18:21:02 +0200'>
    <documentation>. . .</documentation>
    <FSM:StateMachine xmlns:FSM="http://www.abo.fi/FSM/1.0"
                      xmi:id="e1" name="Example">
        <alphabet xmi:id="e2" name="A">
            <transition xmi:idref="e3" />
        </alphabet>
        <alphabet xmi:id="e4" name="B">
            <transition xmi:idref="e5" />
        </alphabet>
        <state xmi:id="e6" name="S1">
            <incoming xmi:idref="e5" />
            <outgoing xmi:idref="e3" />
        </state>
        <state xmi:type="AcceptingState" xmi:id="e7" name="S2">
            <incoming xmi:idref="e3" />
            <outgoing xmi:idref="e5" />
        </state>
        <transition xmi:id="e3" source="e6" target="e7" trigger="e2" />
        <transition xmi:id="e5" source="e7" target="e6" trigger="e4" />
    </FSM:StateMachine>
</xmi:XMI>
```

รูปที่ 2.8 แสดงตัวอย่างของรูปแบบมาตราฐานเอกซ์เอ็มไอ [9]

2.1.3. การจับคู่ความสัมพันธ์ของกราฟ (Relational Graph Matching) [3]

การจับคู่ความสัมพันธ์ของกราฟเพื่อแสดงว่ากราฟสองกราฟนั้นมีความเหมือนหรือว่ากราฟนั้นเป็นส่วนหนึ่งของกราฟอื่น ๆ ที่มีอยู่ โดยมีวัตถุประสงค์เพื่อค้นหาการจับคู่ระหว่างจุดยอด (Vertex) และเส้นเชื่อม (Edge) ที่แทนความสัมพันธ์โดย G = (V,E) ภายใต้ข้อกำหนดเมื่อ V เป็นเซตของจุดยอดและ E เป็นเซตของเส้นเชื่อม โดยมีวัตถุประสงค์เพื่อใช้ในการการจับคู่ความสัมพันธ์ของกราฟ ดังเช่น G_1= (V_1,E_1) เมื่อ G_1เป็นกราฟที่กำหนด ที่ปรากฏอยู่ใน G_2= (V_2,E_2) เมื่อ G_2 เป็นโมเดลกราฟที่จะนำมาหาความสัมพันธ์ของกราฟ ซึ่งสามารถแสดงโดยการใช้ฟังก์ชัน f : V_1→ V_2 จากจำนวนจุดเชื่อมต่อในกราฟ G_1 ไปยัง โมเดลกราฟของ G_2 ซึ่งฟังก์ชัน f จะประกอบไปด้วยเซตของการจับคู่ระยะห่างระหว่างจุดยอดที่มีความเป็นไปได้ระหว่าง 2 กราฟ มีการกำหนดดัชนีสำหรับจุดเชื่อมต่อของกราฟ G_1 ที่จับคู่กับโมเดลกราฟของ G_2 แสดงโดย (u,v) เป็นสมาชิกของ f โดยที่จุดเชื่อมต่อ u เป็นสมาชิกของ V_1และจุดยอด v เป็นสมาชิกของ V_2 [8] ดังรูปที่ 2.9 แสดงกราฟ G1, G2 และ G3 โดยที่ G1 ซึ่งเป็นกราฟที่มีความคล้ายคลึงไปยังกราฟ G2 และ กราฟ G3 ในการจับคู่ความสัมพันธ์ของจำนวนเส้นเชื่อมของกราฟ G1 สามารถเปรียบเทียบได้กับเส้นเชื่อมของกราฟ G2 และ กราฟ G3 ได้มากกว่า 1 จำนวนของเส้นเชื่อมที่จะมีความคล้ายคลึงกัน



รูปที่ 2.9 แสดงตัวอย่างความคล้ายคลึงกันระหว่างกราฟ G1, G2 และ G3 [11]

ทั้งนี้การจับคู่ความสัมพันธ์ของกราฟ โดยจะมีการกำหนดจุดยอดหลักของกราฟ G1 ที่ต้องการเปรียบเทียบแล้วนำไปเปรียบเทียบกับแต่ละจุดยอดของกราฟ G2 และกราฟ G3 ตามลำดับ ในแต่ละคู่ของการเปรียบเทียบจะพิจารณาทั้งส่วนของความสัมพันธ์ของจุดเชื่อมต่อและเส้นเชื่อมที่คล้ายคลึงกัน เพื่อที่ผลของการจับคู่ความสัมพันธ์จะแสดงความผิดพลาดที่เกิดขึ้นน้อยที่สุดระหว่างสองกราฟ ข้อผิดพลาดจะถูกกำหนดจากความแตกต่างระหว่างแต่ละคู่ของการจับคู่จุดยอดรวมกับการจับคู่ความสัมพันธ์ของเส้นเชื่อม ซึ่งสามารถดูได้จากระยะห่างระหว่างกราฟทั้งสอง โดยลำดับแรกของการจับคู่กราฟนี้จะพิจารณาจากข้อสรุปของข้อผิดพลาดในการจับคู่ของจุดยอดลำดับที่สองพิจารณาจากข้อสรุปของข้อผิดพลาดในการจับคู่ของเส้นเชื่อม โดยสามารถ แบ่งเป็น 2 หมวด ดังนี้ [1]

1) การจับคู่กราฟที่มีความแม่นยำ (Exact Graph Matching) เมื่อพบปัญหาในการจับคู่ระหว่างจุดยอดกับจุดยอด ที่มีความเหมือนกันของ 2 กราฟ โดยที่มีจำนวนของจุดยอดเท่ากัน ทำให้มีความเชื่อมโยงทั้งส่วนของจุดยอดและเส้นเชื่อม ในการตรวจหาแบบรูปการออกแบบการใช้อัลกอริทึมต้องตรวจสอบทุกความเป็นไปได้ของกราฟย่อยที่จะปรากฏในกราฟของระบบ

2) การจับคู่กราฟที่มีไม่มีความแม่นยำ (Inexact Graph Matching) เพื่อตรวจหาการจับคู่แบบหนึ่งต่อหนึ่งที่ดีที่สุดระหว่างจุดยอดของจุดยอดที่นำเข้ากับจุดยอด ที่ต้องการจับคู่ซึ่งอาจมีจำนวนมากที่มีความเป็นไปได้ของการจับคู่และยังมีวัตถุประสงค์เพื่อค้นหาการจับคู่ที่ดีที่สุด ตัวอย่าง เช่น อัลกอริทึมการคำนวณระยะห่างระหว่าง 2 กราฟเพื่อค้นหาจำนวนการแก้ไขที่ส่งผลไปยังกราฟอื่น ๆ

2.2. งานวิจัยที่เกี่ยวข้อง

ในหัวข้อนี้กล่าวถึงภาพรวมของงานวิจัยที่เกี่ยวข้องกับการตรวจหาแบบรูปการออกแบบถึงแนวคิด ทฤษฎี คำนิยาม รวมทั้งส่วนที่งานวิจัยนี้นำมาปรับปรุง ดังต่อไปนี้

2.2.1. งานวิจัยเรื่อง “Design Pattern Detection using Similarity Scoring” [1]

เสนอโดย Nikolaos และคณะ ในปี ค.ศ. 2006 เป็นการตรวจหาแบบรูปการออกแบบจากการคำนวณหาค่าคะแนนความคล้ายคลึง (Similarity Scoring Algorithm) ระหว่างจุดยอด (Vertices) ของกราฟแบบแผนการออกและกราฟของระบบ ในการตรวจหาความคล้ายคลึงจะแบ่งกราฟออกเป็นสองส่วน คือ ส่วนแรกแสดงกราฟที่ถูกปรับแต่งจากแบบรูปการออกแบบที่ยังคงเงื่อนไขของความสัมพันธ์ของจุดยอดนั้น ๆ อยู่ ส่วนที่สองแสดงกราฟย่อยที่เรียงลำดับอย่างง่ายของคลาสนั้นแล้วหาความสัมพันธ์ของเส้นเชื่อมระหว่างจุดยอด เช่น เส้นความสัมพันธ์การสืบทอดคุณสมบัติ (Generalization) เส้นความสัมพันธ์แบบทั่วไป (Association) เป็นต้น ด้วยการสร้างเมตริกซ์ที่แสดงเส้นความสัมพันธ์ของจุดยอดกับกราฟกลุ่มที่หนึ่งและระหว่างเส้นความสัมพันธ์ของจุดยอดกับกราฟกลุ่มที่สอง จากนั้นนำมาค่าคะแนนความคล้ายคลึงของกราฟในแต่ละกลุ่ม ผลรวมที่จากแต่ละกลุ่มมาทำการปรับค่าคะแนนให้เป็นปกติ (Normalization) เพื่อสรุปค่าคะแนนความคล้ายคลึงของกราฟจะอยู่ในช่วง [0,1] จากงานวิจัยนี้ได้นำกราฟมาคำนวณหาค่าความคล้ายคลึงในรูปของเมทริกซ์ที่สอดคล้องกับแผนภาพคลาส แต่งานวิจัยนี้ไม่ได้พิจารณาถึงค่าน้ำหนักของเส้นความสัมพันธ์ที่เชื่อมโยงจุดต่อในเมทริกซ์ที่คำนวณหาค่าคะแนนความคล้ายคลึง

2.2.2. งานวิจัยเรื่อง “Design Pattern Detection by Template Matching” [2]

โดย Jing Dong Yongtao Sun และ Yajing Zhao ในปี ค.ศ. 2008 เป็นการตรวจหาแบบรูปการออกแบบด้วยการใช้แผ่นแบบการจับคู่ (Template Matching) โดยการแปลงแผนภาพคลาสให้อยู่ในรูปแบบเมทริกซ์ที่แสดงความสัมพันธ์ของเส้นเชื่อมและจุดยอดของกราฟด้วยการแทนค่า 0 แสดงถึงความไม่สัมพันธ์กันของจุดยอดที่อยู่ในกราฟและแทนค่า 1 แสดงถึงความสัมพันธ์ของจุดยอดที่อยู่ในกราฟเมทริกซ์ของแต่ละกราฟจะนำมารวมกันโดยใช้ผลคูณสหสัมพันธ์ (Cross Correlation) จากนั้นผลรวมของเมทริกซ์ที่ได้จะถูกนำมาคำนวณหาค่าคะแนนความคล้ายคลึงเพื่อจะใช้ในการพิจารณาตรวจหาความสัมพันธ์ของจุดยอดที่ปรากฏอยู่ในเมทริกซ์แบบรูปการออกแบบที่อยู่ในรูปแบบของแผ่นแบบการจับคู่และเมทริกซ์ของระบบที่มีค่าคะแนนความคล้ายคลึงเหมือนกัน งานวิจัยนี้มีการสร้างเครื่องมือในการตรวจหาแบบรูปการออกแบบและรายงานผลการตรวจหาจากค่าคะแนนความคล้ายคลึง จากวิธีการดังกล่าวสามารถตรวจหาแบบรูปการออกแบบได้และยังสามารถระบุค่าความคลาดเคลื่อนของแต่ละจุดยอดในเมทริกซ์นั้นได้

2.2.3. งานวิจัยเรื่อง “A Review of Design Pattern Mining Techniques” [7]

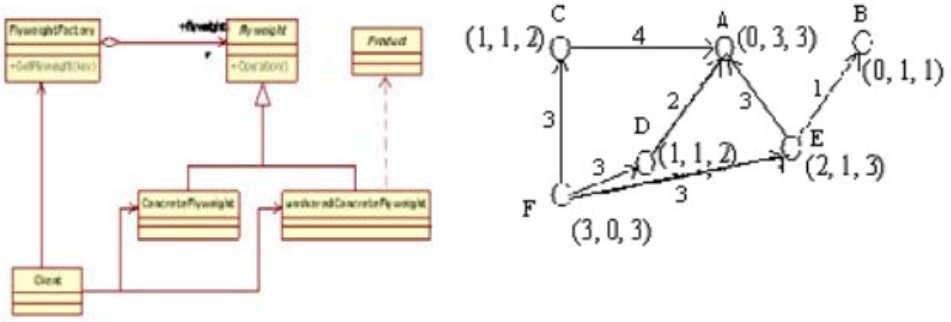
เสนอโดย Jing Dong Yajing Zhao และ Tu Peng ในปี ค.ศ. 2008 เป็นการรวบรวมวิธีการค้นหาแบบรูปการออกแบบจากงานวิจัยที่ค้นพบตั้งแต่ ปี 1996-2007 โดยแยกกลุ่มวิธีการเป็น 4 กลุ่ม คือ หนึ่งการใช้ตัวแทนที่ระหว่างการแปลงแบบรูป (Intermediate Representation) เป็นการแปลงซอร์สโค้ดไปเป็นการแทนค่าในระยะกลางที่ช่วยลดความซับซ้อนในการค้นหาแบบรูปการออกแบบ ซึ่งทางเลือกของการค่าในระยะกลางของระบบซอฟต์แวร์และแบบรูปการออกแบบตามวิธีการต่างๆ เช่น Abstract Syntax Tree (AST) Abstract Semantic Graph (ASG) บิต (bit) จุดต่อและเมทริกซ์ เป็นต้น สองการหาค่าความแน่นอนและค่าความใกล้เคียง (Exact and Approximate Matches) เป็นการค้นหาบางส่วนของการออกแบบสถาปัตยกรรมระบบซอฟต์แวร์เพื่อใช้ในการยืนยันองค์ประกอบเชิงโครงสร้างไปยังลักษณะของโครงสร้างของแบบรูปที่ถูกนิยาม ทั้งการออกแบบเชิงโครงสร้างและการออกแบบเชิงพฤติกรรมที่มีค่าตามความคาดหวังที่ถูกกำหนดหรือจนกว่าจะพบการกฎการจับคู่ที่มีความสอดคล้องกัน เพื่อที่จะสามารถยืนยันได้ว่าพบการจับคู่แล้ว แต่ถ้าบางส่วนยังไม่ถูกจับคู่จะถือว่าไม่มีการจับคู่กัน วิธีการดังกล่าวอาจจะเป็นการแก้ปัญหาของการตรวจหาแบบรูปการออกแบบ แต่บางกรณีการจับคู่ที่ถูกต้องอาจจะถูกคัดออกไปเนื่องมาจากกฎที่ใช้มีความเข้มงวด ดังนั้นวิธีการคำนวณหาลำดับความคล้ายคลึงซึ่งอัลกอริทึมในการจับคู่แบบรูปการออกแบบที่พบอย่างเช่น การใช้ค่าคะแนนความคล้ายคลึงและการใช้แผ่นแบบการจับคู่ เป็นต้น สาม

การสร้างมโนภาพ (Visualization) มีความสำคัญในกระบวนการวิศวกรรมย้อนกลับ เพื่อช่วยให้ผู้ใช้งานระบบสามารถเข้าใจผลลัพธ์ของระบบได้ง่ายยิ่งขึ้น แบบรูปการออกแบบก็แสดงอยู่ในแผนภาพยูเอ็มแอล หรือแผนภาพต้นไม้การสืบทอดคลาส วิธีการสุดท้าย คือ การใช้เครื่องมืออัตโนมัติหรือการเครื่องมือสนับสนุนการโต้ตอบจากมนุษย์ (Automated or Human Interactive Tool Support) เป็นวิธีการที่ให้มนุษย์เป็นผู้ตัดสินความแม่นยำของกระบวนการในค้นหาแบบรูป ซึ่งการการให้มนุษย์มีส่วนร่วมในการวิเคราะห์ รวมถึงวิธีวิศวกรรมย้อนกลับกระบวนการกึ่งอัตโนมัติ ในการวิเคราะห์อาจจะมีการตรวจสอบผลที่ได้จากเครื่องมือ การตัดส่วนที่ไม่เกี่ยวข้องออกไปและการอธิบายผล แต่ข้อเสียที่นำมนุษย์เข้ามาตรวจหาแบบรูปจะทำให้กระบวนการต่างๆ ช้าลง การสรุปผลโดยตารางเปรียบเทียบวิธีการการค้นหาแบบรูปการออกแบบของแต่ละงานวิจัย รวมทั้งวิเคราะห์ผลการทดลองในแต่ละวิธีการซึ่งมีความแตกต่างกัน เพื่อเป็นการตรวจสอบปัญหาความผิดพลาดในการค้นหาแบบรูปการออกแบบจากงานวิจัยนี้ยังไม่ได้ครอบคลุมวิธีการค้นหาแบบรูปการออกแบบที่ค้นพบภายหลังแต่สามารถใช้ในการวิเคราะห์แนวโน้มของวิธีการค้นหาแบบรูปการออกแบบที่จะทำการพัฒนาต่อไปได้

2.2.4. งานวิจัยเรื่อง “Design Pattern Detection using Inexact Graph Matching” [3]

เสนอโดย Manjari Gupta และคณะ ในปี ค.ศ. 2010 ได้นำเสนอวิธีการตรวจหาแบบรูปการออกแบบ โดยการใช้อัลกอริทึมการจับคู่กราฟ ซึ่งกราฟนั้นจะแสดงถึงความตรงกันของความเส้นความสัมพันธ์ของกราฟที่มีอยู่ในแผนภาพ ยูเอ็มแอลของการออกแบบระบบตลอดจนแผนภาพยูเอ็มแอลของแบบรูปการออกแบบ ซึ่งแนวคิดของการจับคู่กราฟที่แม่นยำ (Exact graph matching) มีวัตถุประสงค์เพื่อกำหนดว่ากราฟ 2 กราฟเหมือนกันหรือว่าส่วนของกราฟย่อยปรากฏอยู่ในกราฟอื่นๆ ในการตรวจหากราฟที่จับคู่กันอย่างถูกต้องนั้นจะมีความล้มเหลวในการตรวจหากราฟที่มีความเหมาะสมกัน เนื่องจากข้อมูลอาจถูกบิดเบือนไป ทางเลือกหนึ่งในการจัดการปัญหาดังกล่าวคือการใช้การจับคู่ความสัมพันธ์กราฟที่ไม่แม่นยำ (Inexact graph matching) ทั้งนี้อัลกอริทึมที่ใช้จะถูกนำมาใช้ก็เพื่อตรวจหากราฟย่อยที่มีความตามวิธีการเอสตาร์ (A*) เพื่อหาการจับคู่กราฟที่ดีที่สุด โดยการตรวจหาและมีการปรับปรุงอัลกอริทึมให้มีประสิทธิภาพมากยิ่งขึ้นโดยวัดผลจากจำนวนรอบที่มีค่าน้อยลง งานวิจัยนี้ได้อ้างอิงทฤษฎีที่เกี่ยวข้อง คือ การแสดงความสัมพันธ์ของกราฟ (Relationship Graphs Representation) การกำหนดค่าให้กับจุดเชื่อมต่อ 3 ลำดับ (3-tuple) ได้แก่ t_1 แทน จำนวนคลาสหลัก t_2 แทน จำนวนคลาสย่อย และ t_3 แทน จำนวนของเส้นความสัมพันธ์ของจุดเชื่อมต่อ พร้อมกำหนดค่าของเส้นความสัมพันธ์ตามชนิดของเส้นความสัมพันธ์ โดยกำหนดค่าอย่างไม่มีความหมายหรือขึ้นกับคุณลักษณะของเส้นความสัมพันธ์ เช่น แทนค่า 1 สำหรับเส้นความสัมพันธ์

ที่ขึ้นต่อกันระหว่างคลาส (Dependency) แทนค่า2 สำหรับเส้นความสัมพันธ์การสืบทอดคุณสมบัติ (Generalization), แทนค่า 3 สำหรับเส้นความสัมพันธ์แบบทั่วไป (Association) และแทนค่า 4 สำหรับเส้นสัมพันธ์ที่เป็นส่วนหนึ่งของคลาส (Aggregation) ตัวอย่างการแปลงแผนภาพคลาสให้อยู่ในรูปแบบของโมเดลกราฟ ดังรูปที่ 2.10



ก) ข)

รูปที่ 2.10 การแปลงแผนภาพคลาสเป็นโมเดลกราฟ [3]
ก) แผนภาพคลาส ข) โมเดลกราฟที่แสดงลำดับจุดเชื่อมต่อและค่าของเส้นความสัมพันธ์

อีกทฤษฎีที่เกี่ยวข้องคือ อัลกอริทึมการจับคู่กราฟ (Graph Matching Algorithm) การกำหนดว่า 2 กราฟ มีความเหมือนกันหรือว่ากราฟนั้นเป็นกราฟย่อยที่อยู่ในกราฟอื่นๆ เพื่อให้ 2 กราฟนั้นมีการจับคู่กันระหว่างจุดเชื่อมต่อและเส้นเชื่อมเพื่อที่จะได้ค่าความคลาดเคลื่อนที่น้อยที่สุดระหว่าง 2 กราฟ ซึ่งความแตกต่างจะถูกกำหนดโดยค่าความคลาดเคลื่อนระหว่างแต่ละการจับคู่ของจุดเชื่อมต่อ รวมกับค่าความแตกต่างระหว่างความสัมพันธ์ของเส้นเชื่อมซึ่งเป็นการแสดงระยะห่างระหว่าง 2 กราฟ โดยขั้นแรกของอัลกอริทึมจะเป็นการจับคู่ค่าความคลาดเคลื่อนที่รวมมาจากการจับคู่ความคลาดเคลื่อนของจุดเชื่อมต่อ ขั้นตอนที่สองจะเป็นการจับคู่ค่าความคลาดเคลื่อนของเส้นเชื่อม โดยเป็นมีแนวคิดพื้นฐานของอัลกอริทึมใหม่ที่มีการตรวจหาซ้ำๆในจุดเชื่อมต่อที่เป็นไปได้ของการจับคู่และเลือกการจับคู่ที่ดีที่สุดของแต่ละรอบซึ่งมีสมมุติฐานที่สำคัญ คือ เบื้องหลังการลำดับอัลกอริทึมที่มีการจับคู่ที่ดีที่สุดระหว่าง 2 กราฟ หนึ่งในนั้นควรจะมีความคล้ายคลึงที่เหมาะสมของจุดเชื่อมต่อและเส้นเชื่อมที่มีความสัมพันธ์กันระหว่าง 2 กราฟ โดยขั้นตอนของอัลกอริทึมที่ใช้ในการตรวจหาแบบรูปการออกแบบ ดังนี้

ขั้นตอนการหาค่า (B) ในการจับคู่จุดเชื่อมต่อ

สำหรับการตรวจสอบในการจับคู่ใน B

1) คำนวณค่าความคลาดเคลื่อนของจุดเชื่อมต่อที่มีความคล้ายคลึงกันระหว่างการจับคู่จุดเชื่อมต่อ $|MG_{t1}\text{-}DPG_{t1}|+ |MG_{t2}\text{-}DPG_{t2}|+ |MG_{t3}\text{-}DPG_{t3}|$,เมื่อ MG_{ti} คือ i^{th} ที่เป็นส่วนประกอบของจุดเชื่อมต่อที่อยู่ในโมเดลกราฟและ DPG_{ti} ที่เป็นส่วนประกอบของจุดเชื่อมต่อที่จับคู่กับที่อยู่ในกราฟแบบรูปการออกแบบ
2) เพิ่มค่าความคลาดเคลื่อนที่ตรงกันของการจับคู่ความเส้นเชื่อม โดยใช้ความแตกต่างของการจับคู่เส้นเชื่อม ถ้ามีค่าเป็น 0 ทั้งเส้นเชื่อมที่มีความตรงกันก็จะแสดงว่ามีความสัมพันธ์เหมือนกัน นอกจากนี้ยังกำหนดค่าของการจับคู่ความคลาดเคลื่อนของเส้นเชื่อมให้กับค่า q เมื่อ q เป็นค่าที่มีจำนวนเป็นบวกที่มากซึ่งจะแสดงถึงความไม่ตรงกันของความสัมพันธ์ของเส้นเชื่อม
3) ทำการบันทึกค่าของการจับคู่ที่มีความคลาดเคลื่อนต่ำสุด

การตรวจหาแบบรูปการออกแบบโดยใช้อัลกอริทึมการจับคู่กราฟ มีการตรวจหาแบบรูปการออกแบบจากแผนภาพกราฟโดยแบ่งเป็น 3 กรณี ได้แก่ ความสัมพันธ์ของกราฟแบบรูปการออกแบบที่มีความเหมือนกันทุกประการ ความสัมพันธ์ของกราฟแบบแผนการออกแบบที่มีความเหมือนกันบางส่วน และกราฟแบบรูปการออกแบบที่ไม่มีความเหมือนกัน โดยมีขั้นตอนดังนี้

ขั้นตอนการตรวจหาแบบรูปการออกแบบ

1) การแปลงแผนภาพคลาสให้อยู่ในรูปแบบของกราฟ
2) การระบุค่าของลำดับจุดเชื่อมต่อและค่าระบุชนิดของเส้นความสัมพันธ์
3) คำนวณค่าความคลาดเคลื่อนของจุดเชื่อมต่อที่มีความคล้ายคลึงกันระหว่างการจับคู่จุดเชื่อมต่อของโมเดลกราฟกับกราฟแผนภาพแบบรูปการออกแบบ
4) หาค่าผลต่างความคลาดเคลื่อนของจุดเชื่อมต่อในรูปแบบของตารางเมทริกซ์
5) รวมค่าความคลาดเคลื่อนของแต่ละการจับคู่จุดเชื่อมต่อ
6) พิจารณาค่าที่น้อยที่สุดในแต่ละแถวของเมทริกซ์เทียบกับเมทริกซ์เปรียบเทียบแบบซ้ำไปเรื่อย ๆ จนพบจุดเชื่อมต่อที่มีความเหมือนกัน

จากงานวิจัยนี้ระบุลักษณะของเส้นความสัมพันธ์ยังไม่ครบทุกชนิดในการออกแบบระบบซอฟต์แวร์และยังอ้างอิงการเชื่อมโยงของคลาสโดยนับรวมเป็นการสืบทอดเท่านั้น ซึ่งไม่ได้มีแยกจำนวนของเส้นความสัมพันธ์ที่เข้า-ออกจากคลาสนั้น ๆ ด้วย

2.2.5. งานวิจัยเรื่อง “Mining Object-Oriented Design Models for Detecting Identical Design Structures” [7]

เสนอโดย Umut Tekin Ural Erdemir และ Feza Buzluca ในปี ค.ศ. 2012 ได้นำกราฟไมนิง (Mining Graph) ใช้ในการตรวจหาความเหมือนกันของการออกแบบเชิงโครงสร้างที่อยู่ในมุมมองระดับสูงของสถาปัตยกรรมการออกแบบซอฟต์แวร์เชิงวัตถุ (Object-oriented) โดยมีการกำหนดประเภทของโหนด (Nodes) และเส้นเชื่อมความสัมพันธ์ (Edges) ที่ซ่อนอยู่ในแต่ละคลาสที่สามารถบอกถึงข้อมูลของเงื่อนไขที่จะนำมาใช้ในการตรวจหาจำนวนความถี่ที่เหมือนกันที่ปรากฏอยู่ในกราฟหลัก (Frequent isomorphic sub-graph) และการจำกัดจำนวนความถี่ของกราฟหลัก (Closed Frequent sub-graph) ซึ่งวิธีการตรวจหาแบบรูปการออกแบบนั้นจะทำการแยกส่วนย่อยของโมเดลกราฟ แล้วนำมาตรวจหาถึงจำนวนจุดต่อทั้งหมด จำนวนเส้นเชื่อม จำนวนโมเดลกราฟย่อย และจำนวนของจุดต่อในแต่ละส่วนย่อย จากนั้นนำมาหาค่าความถี่ของกราฟย่อยในกราฟหลัก งานวิจัยนี้มีการสร้างเครื่องมือช่วยตรวจหาการออกแบบเชิงโครงสร้างที่สามารถรายงานผลการตรวจหาจำนวนแบบรูปการออกแบบในลักษณะต่าง ๆ ได้

บทที่ 3

การออกแบบและปรับปรุงอัลกอริทึมการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง

ในบทนี้ได้ทำอธิบายขั้นตอนวิธีที่ใช้ในการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง โดยสามารถแบ่งออกเป็น 9 ส่วน ดังนี้

1) การกำหนดคำนิยามในการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง
2) อัลกอริทึมในการตรวจจับแบบรูปการออกแบบ
3) วิธีการแปลงแผนภาพคลาสแบบรูปการออกแบบเป็นกราฟเชิงความสัมพันธ์
4) ขั้นตอนการสร้างตารางคุณลักษณะท้องถิ่น (LPT-P) และตารางความสัมพันธ์ (RT-P) ของแบบรูปการออกแบบ
5) การเลือกโหนดอ้างอิงสำหรับแต่ละแบบรูปการออกแบบ
6) การกำหนดความเหมือนแบบรูปการออกแบบ
7) ขั้นตอนวิธีนำแผนภาพคลาสของระบบมาตรวจจับแบบรูปการออกแบบ
8) การปรับปรุงอัลกอริทึมการตรวจจับแบบรูปการออกแบบ
9) ตัวอย่างการตรวจจับแบบรูปการออกแบบ Adapter จากอัลกอริทึมที่ปรับปรุง

3.1. กำหนดคำนิยามในการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง

ในส่วนนี้จะอธิบายถึงนิยามที่ใช้ในการตรวจจับแบบรูปการออกแบบ ซึ่งจะนิยามถึงกราฟเชิงความสัมพันธ์ ตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส ตารางความสัมพันธ์ของแผนภาพคลาส โหนดอ้างอิงและความเหมือนแบบรูปการออกแบบ

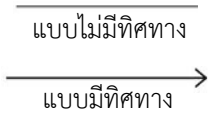
3.1.1. กราฟเชิงความสัมพันธ์

ในหัวข้อนี้ได้แสดงถึงการกำหนดนิยามของกราฟเชิงความสัมพันธ์ โดยแบ่งเป็น 5 นิยาม ซึ่งนิยามที่1 ได้กล่าวถึงการจับคู่แผนภาพคลาสให้อยู่ในรูปแบบของกราฟใดๆ นิยามที่2 กล่าวถึงลักษณะของเส้นความสัมพันธ์ตามมาตรฐานยูเอ็มแอล นิยามที่3 กล่าวถึงการกำหนดลำดับของประเภทเส้นความสัมพันธ์ที่เข้าออกจากคลาส นิยามที่4 กล่าวถึงการเทียบแผนภาพแบบรูปการออกให้อยู่ในรูปแบบของกราฟ และนิยามที่5 กล่าวถึงการเทียบแผนภาพของระบบให้อยู่ในรูปแบบของกราฟซึ่งอธิบายในรายละเอียดได้ดังนี้

นิยามที่ 1: กราฟ G = (V,E) ใดๆ เมื่อ V เป็นเซตของโหนดทั้งหมดและ E เป็นเซตของเส้นเชื่อมทั้งหมด การจับคู่ความสัมพันธ์ของกราฟ โดยกำหนดโหนด v_1,v_2 จับคู่กับส่วนของแผนภาพคลาสยูเอ็มแอล c_1,c_2 ในแผนภาพยูเอ็มแอล ในขณะที่เส้นเชื่อม e_1 จะแสดงความสัมพันธ์ระหว่าง c_1 และ c_2 ซึ่งเส้นความสัมพันธ์ที่เชื่อมระหว่างคลาสตามประเภทของเส้นความสัมพันธ์ในแผนภาพยูเอ็มแอล ซึ่งในแต่ละโหนดของ v_1 จะถูกระบุถึงประเภทของคลาส ซึ่งประกอบไปด้วยคลาสประเภทนามธรรม รูปธรรม และอินเตอร์เฟส

นิยามที่ 2: การกำหนดนิยามลักษณะของเส้นความสัมพันธ์ตามมาตรฐานยูเอ็มแอล ซึ่งในการออกแบบจะต้องมีการระบุทิศทางของเส้นความสัมพันธ์ที่มีระหว่างคลาส โดยสามารถแบ่งเป็น 6 ประเภท [10] สามารถอธิบายได้ ดังตารางที่ 3.1

ตารางที่ 3.1 ลักษณะของเส้นความสัมพันธ์

ประเภทของเส้นความสัมพันธ์	สัญลักษณ์	คำอธิบาย
แอสโซซิเอชัน		ความสัมพันธ์ที่เกิดขึ้นจากการทำงานของคลาสหนึ่งที่มีลักษณะขึ้นอยู่กับคลาสอื่น ๆและมีลักษณะเป็นความสัมพันธ์แบบโครงสร้าง
เจนเนอรัลไลเซชัน		ความสัมพันธ์ระหว่างคลาสในลักษณะของการสืบทอดคุณสมบัติ จากโครงสร้างคลาสหนึ่งไปยังโครงสร้างอีกคลาสหนึ่ง
อะกรีเกรชัน		ความสัมพันธ์แบบเป็นส่วนหนึ่งของ โดยจะมีคลาสที่ใหญ่ที่สุดเป็นวัตถุหลัก และมีคลาสอื่น ๆ เป็นส่วนหนึ่งของวัตถุ
คอมโพสิชัน		ความสัมพันธ์แบบเป็นองค์ประกอบของอีกคลาสที่เป็นคลาสหลักเมื่อคลาสหลักถูกทำลายคลาสที่เป็นองค์ประกอบจะใช้งานไม่ได้
ดีเพนเดนซี		ความสัมพันธ์ที่เกิดขึ้นจากการทำงานของคลาสหนึ่งที่มีลักษณะขึ้นอยู่กับคลาสอื่นๆ
อิมพลีเมนต์เตชันหรือรีไลเซชัน		ความสัมพันธ์ที่เกิดขึ้นจากการใช้งานระหว่างอินเตอร์เฟสกับคลาสอื่นๆ

นิยามที่ 3: การกำหนดนิยามลำดับเส้นความสัมพันธ์ของคลาสให้อยู่ในรูปแบบของ 12 ลำดับ (12-Tuple) โดยแต่ลำดับแทนด้วยเส้นความสัมพันธ์ตามมาตรฐานยูเอ็มแอล ดังตารางที่ 3.2 [10]

ตารางที่ 3.2 การนิยามค่าของแต่ละประเภทเส้นความสัมพันธ์

ลำดับ	ประเภทของเส้นความสัมพันธ์	คำอธิบาย
1,2	แอสโซซิเอชัน	A_1 แทน จำนวนเส้นความสัมพันธ์เข้า
		A_2 แทน จำนวนเส้นความสัมพันธ์ออก
3,4	เจนเนอรัลไลเซชัน	G_3 แทน จำนวนเส้นความสัมพันธ์เข้า
		G_4 แทน จำนวนเส้นความสัมพันธ์ออก
5,6	อะกรีเกรชัน	Ag_5 แทน จำนวนเส้นความสัมพันธ์เข้า
		Ag_6 แทน จำนวนเส้นความสัมพันธ์ออก
7,8	คอมโพสิชัน	C_7 แทน จำนวนเส้นความสัมพันธ์เข้า
		C_8 แทน จำนวนเส้นความสัมพันธ์ออก
9,10	ดีเพนเดนซี	D_9 แทน จำนวนเส้นความสัมพันธ์เข้า
		D_{10} แทน จำนวนเส้นความสัมพันธ์ออก
11,12	อิมพลีเมนต์เตชันหรือรีไลเซชัน	I_{11} แทน จำนวนเส้นความสัมพันธ์เข้า
		I_{12} แทน จำนวนเส้นความสัมพันธ์ออก

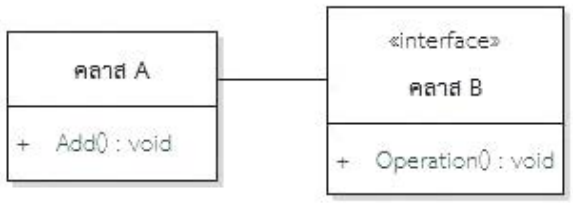
จากตารางที่ 3.1 สามารถแทนค่าลำดับของเส้นความสัมพันธ์ในแต่ละประเภทของเส้นความสัมพันธ์ ได้ดังนี้ (A_1,A_2,G_3,G_4,Ag_5,Ag_6,C_7,C_8,D_9,D_{10},I_{11},I_{12}) ซึ่งเป็นการจัดกลุ่มของเส้นความสัมพันธ์ตามจำนวนเส้นที่เข้าหาคลาส (Fan-in) และเส้นที่ออกจากคลาส (Fan-out) จากคลาสนั้น ๆ

นิยามที่ 4: การเทียบแผนภาพแบบรูปการออกแบบเชิงโครงสร้างให้อยู่ในรูปแบบของกราฟ กราฟแบบรูปการออกแบบ P = (V,E,L) เมื่อ V เป็นเซตของโหนดทั้งหมด E เป็นเซตของเส้นเชื่อม และ L เป็นเซตของเส้นความสัมพันธ์เข้าออก โดยกราฟแบบรูปการออกแบบ P จะถูกจับคู่ความสัมพันธ์เข้ากับแบบรูปการออกแบบเชิงโครงสร้าง

นิยามที่ 5: การเทียบแผนภาพของระบบให้อยู่ในรูปแบบของกราฟ กราฟระบบ S = (V,E,L) เมื่อ V เป็นเซตไม่จำกัดของโหนดทั้งหมด E เป็นเซตของเส้นเชื่อม และ L เป็นเซตของเส้นความสัมพันธ์เข้าออก โดยกราฟของระบบ S จะถูกจับคู่ความสัมพันธ์เข้ากับส่วนของแผนภาพคลาสยูเอ็มแอล

3.1.2. ตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส (Local Properties Table: LPT)

ในการออกแบบแผนภาพคลาสนั้น โครงสร้างของคลาสจะอธิบายถึงคุณลักษณะของวัตถุ โดยมีการระบุถึง ชื่อคลาส (Class Name) ชนิดของคลาส (Class type) ลักษณะประจำ (Attribute) เมท็อด (Method) และความสัมพันธ์ระหว่างคลาส (Class Relation) อธิบายรายละเอียดดังตารางที่ 3.1 ดังนั้นแล้ว จากคุณลักษณะที่มีของคลาสนั้น จะนำมาสร้างตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส เพื่อเก็บข้อมูลที่ใช้ในการวิเคราะห์ตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง ซึ่งตารางดังกล่าวจะทำการเลือกเก็บข้อมูลของคลาสทั้งหมดที่มีอยู่ในแผนภาพคลาส ลงในสี่คอลัมน์ซึ่งประกอบไปด้วย คอลัมน์ที่หนึ่งสำหรับจัดเก็บข้อมูลชื่อคลาส คอลัมน์ที่สองสำหรับจัดเก็บข้อมูลชนิดของคลาส คอลัมน์ที่สามสำหรับจัดเก็บข้อมูลชื่อเมธ็อดของคลาส และคอลัมน์ที่สี่สำหรับจัดเก็บข้อมูลเส้นความสัมพันธ์ระหว่างคลาสตามลำดับที่ได้กล่าวถึง ดังตารางที่ 3.1 จากการเลือกเก็บข้อมูลดังกล่าวสามารแสดงได้ตามตัวอย่างแผนภาพคลาสรูปที่ 3.1 และตัวอย่างตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส ตารางที่ 3.3



รูปที่ 3.1 ตัวอย่างแผนภาพคลาส

ตารางที่ 3.3 ตัวอย่างการจัดเก็บข้อมูลในตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส

คุณลักษณะท้องถิ่นของแผนภาพคลาสตัวอย่าง			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
คลาส A	Concrete	Add ()	(0,1,0,0,0,0,0,0,0,0,0,0)
คลาส B	Interface	Operation()	(1,0,0,0,0,0,0,0,0,0,0,0)

จากตารางที่ 3.3 สามารถอธิบายลำดับเส้นระหว่างคลาสโดยอ้างอิงจากกราฟเชิงความสัมพันธ์นิยามที่ 3 ซึ่งในแต่ละลำดับจะแสดงถึงจำนวนเส้นความสัมพันธ์ คือ คลาส A มีลำดับเส้นความสัมพันธ์ระหว่างคลาส (0,1,0,0,0,0,0,0,0,0,0,0) คลาส A มีเส้นความสัมพันธ์ออกจากคลาสเป็นแบบแอสโซซิเอชัน จำนวน 1 เส้น และคลาส B มีลำดับเส้นความสัมพันธ์ระหว่างคลาส (1,0,0,0,0,0,0,0,0,0,0,0) คลาส B มีเส้นความสัมพันธ์เข้าหาคลาสเป็นแบบแอสโซซิเอชัน จำนวน 1 เส้น

3.1.3. ตารางความสัมพันธ์ของคลาส (Relational Table: RT)

การออกแบบแผนภาพคลาสนั้น ส่วนของเส้นความสัมพันธ์ที่ใช้เชื่อมโยงระหว่างคลาส จะแสดงถึงการเรียกใช้งานหรือการสืบทอดคุณสมบัติ ซึ่งจะมีลักษณะที่แตกต่างกันออกไปตามความเหมาะของการออกแบบที่เหมาะสมกับการใช้งาน โดยทั่วไปแล้วประเภทของเส้นความสัมพันธ์จะมีดังนี้ แบบแอสโซซิเอชัน (Association) แบบเจนเนอรัลไลเซชัน (Generalization) แบบอะกรีเกรชัน (Aggregation) แบบคอมโพสิชัน (Composition) แบบดีเพนเดนซี (Dependency) และแบบรีไลเซชัน (Realization) หรืออิมพลีเมนต์เตชัน (Implementation) นอกจากนี้เส้นความสัมพันธ์ที่มีระหว่างคลาสยังแสดงถึงทิศทางของความสัมพันธ์ที่มีด้วย ดังนั้นแล้วในส่วนนี้จะแสดงให้เห็นถึงทิศทางของความสัมพันธ์ระหว่างคลาสที่ถูกจัดเก็บลงในตารางความสัมพันธ์ของคลาสได้ตาม ตารางที่ 3.4

ตารางที่ 3.4 ตัวอย่างการจัดเก็บข้อมูลในตารางความสัมพันธ์ของคลาส ตามรูปที่ 3.1

ความสัมพันธ์ของคลาสตัวอย่าง	
คลาสต้นทาง	คลาสปลายทาง
A	B

3.1.4. โหนดอ้างอิง (Reference Node)

ในงานวิจัยนี้ ผู้วิจัยได้กำหนดโหนดอ้างอิงขึ้นมาเพื่อใช้สำหรับการเริ่มต้นการตรวจจับแบบรูปการออกแบบก่อนที่ทำการค้นหาในรายละเอียดของแต่ละแบบรูปการออกแบบ ซึ่งโหนดอ้างอิงนั้นจะมีลักษณะเฉพาะที่เหมาะสมตามแต่ละแบบรูปการออกแบบ การเริ่มตรวจจับโดยใช้โหนดอ้างอิงจะทำให้การค้นหามีรวดเร็วขึ้น เนื่องจากไม่ต้องทำการค้นหาจากคลาสทั้งหมดแล้วจากนั้นค่อยเริ่มดำเนินการค้นหารายละเอียดของแต่ละแบบรูปต่อไป ซึ่งแนวคิดนี้ได้อ้างอิงจากงานวิทยานิพนธ์ของ นุชนาถ สัตย์วินิจ [14]

3.1.5. ความเหมือนแบบรูปการออกแบบและวิธีการคำนวณความเหมือน

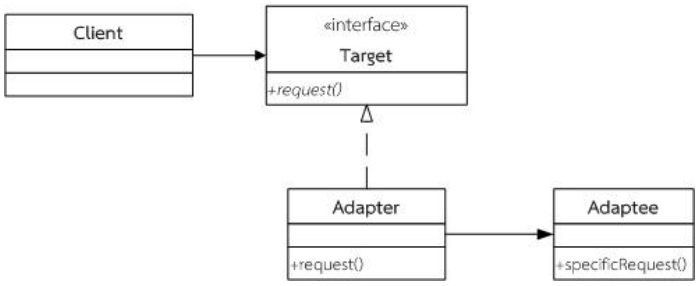
งานวิจัยนี้ ผู้วิจัยได้กำหนดความเหมือนแบบรูปการออกแบบขึ้นมาเพื่อใช้การพิจารณาการตรวจจับแบบรูปการออกแบบที่มีความเหมือนกับแบบรูปการออกแบบตามมาตรฐาน ซึ่งจะพิจารณาลำดับการตรวจจับคลาสของแบบรูปการออกแบบที่พบ โดยแต่ละแบบรูปการออกแบบจะทำการกำหนดลักษณะเฉพาะของแผนภาพคลาสที่มีความเหมือนกับแบบรูปการออกแบบ การวัดค่าความเหมือนแบบรูปการออกแบบจากสูตรร้อยละของจำนวนของคลาสที่ประกอบเป็นแบบรูปการออกแบบที่ตรวจพบส่วนจำนวนคลาสของแบบรูปการออกแบบตามมาตรฐาน

3.2. วิธีการแปลงแผนภาพคลาสแบบรูปการออกแบบเป็นกราฟเชิงความสัมพันธ์

ในส่วนนี้จะอธิบายถึงขั้นตอนวิธีการแปลงแผนภาพคลาสของแบบรูปการออกแบบเชิงโครงสร้างทั้ง 7 แบบรูป ให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ ซึ่งขั้นตอนจะแบ่งออกเป็นสองส่วน โดยส่วนแรกจะเป็นการอธิบายส่วนประกอบของแบบรูปการออกแบบ ส่วนที่สองแสดงถึงรายละเอียดของแบบรูปการออกแบบที่อยู่นี้รูปแบบของกราฟเชิงความสัมพันธ์โดย มีรายละเอียดดังนี้

3.2.1. แบบรูปการออกแบบ Adapter

ส่วนประกอบของแบบรูปการออกแบบ Adapter สามารถอธิบายรายละเอียดได้ในรูปที่ 3.2 ดังนี้



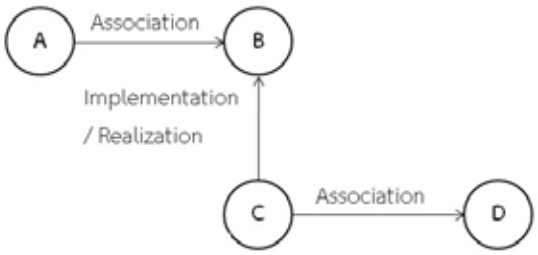
รูปที่ 3.2 โครงสร้างแผนภาพแบบรูปการออกแบบ Adapter [5]

1) ประกอบด้วยคลาสที่ทำหน้าที่ Target ซึ่งมีชนิดของคลาสเป็นคลาสอินเตอร์เฟส มีคลาสที่เกี่ยวข้องเป็นคลาสรูปธรรมซึ่งได้แก่ คลาสที่ทำหน้าที่เป็น Client มีชนิดของคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น Adapter มีชนิดของคลาสรูปธรรม และคลาสที่ทำหน้าที่เป็น Adaptee มีชนิดของคลาสรูปธรรม

2) คลาสที่ทำหน้าที่เป็น Target ต้องมีการอิมพลีเมนต์อย่างน้อย 1 คลาส คือ คลาส Adapter ซึ่งต้องมีความสัมพันธ์แบบอิมพลีเมนต์หรือรีไลเซชันเท่านั้น และยังมีความสัมพันธ์แบบแอสโซซิเอชันกับคลาสที่เป็น Client

3) คลาสที่ทำหน้าที่เป็น Adapter ต้องมีการอิมพลีเมนต์อย่างน้อย 1 คลาส คือ คลาส Adaptee ซึ่งต้องมีความสัมพันธ์แบบแอสโซซิเอชัน

จากแบบรูปการออกแบบ Adapter ที่อธิบายไว้ข้างต้นในรูปที่ 3.2 สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ที่ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้



รูปที่ 3.3 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Adapter

จากรูปที่ 3.3 แผนภาพกราฟแบบรูป Adapter สามารถอธิบายตามลำดับของเส้นความสัมพันธ์ได้ ดังนี้

1) กำหนดโหนด A แทน แผนภาพคลาส Client โดยที่ โหนด A มีจำนวนเส้นความสัมพันธ์เส้นออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,1,0,0,0,0,0,0,0,0,0,0)

2) กำหนดโหนด B แทน แผนภาพคลาส Target โดยที่ โหนด B มีจำนวนเส้นความสัมพันธ์เส้นเข้าหาโหนด จำนวน 2 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชันและแบบอิมพลีเมนต์เตชันหรือรีไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,0,0,0,0,0,0,0,0,1,0)

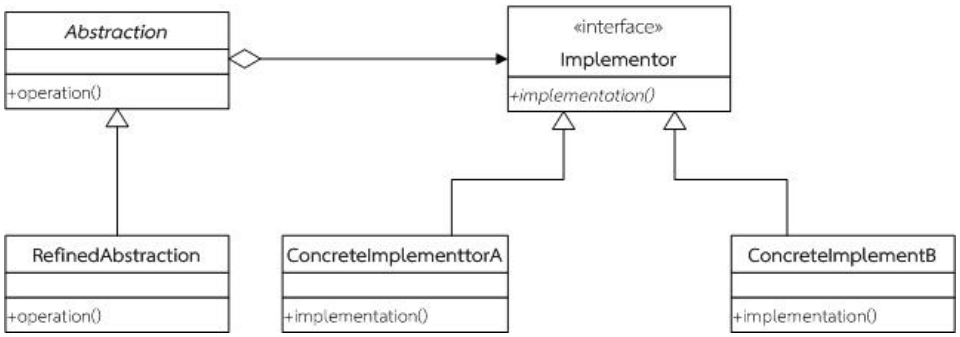
3) กำหนดโหนด C แทน แผนภาพคลาส Adapter โดยที่ โหนด C มีจำนวนเส้นความสัมพันธ์เส้นออกจากโหนด จำนวน 2 เส้น ซึ่งมีความสัมพันธ์แบบอิมพลีเมนต์เตชัน

หรือรีไลเซชันและแบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,1,0,0,0,0,0,0,0,0,0,1)

4) กำหนดโหนด D แทน แผนภาพคลาส Adaptee โดยที่ โหนด D มีจำนวนเส้นความสัมพันธ์เส้นเข้าหาโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,0,0,0,0,0,0,0,0,0,0)

3.2.2. แบบรูปการออกแบบ Bridge

ส่วนประกอบของแบบรูปการออกแบบ Bridge สามารถอธิบายรายละเอียดได้ในรูปที่ 3.4 ดังนี้



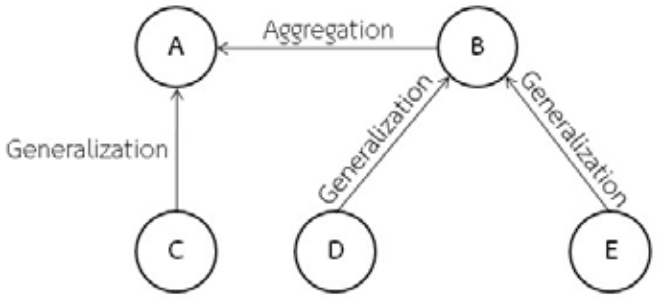
รูปที่ 3.4 โครงสร้างแผนภาพแบบรูปการออกแบบ Bridge [5]

1) ประกอบด้วยคลาสที่ทำหน้าที่ Implementor ซึ่งมีชนิดของคลาสเป็นคลาสอินเตอร์เฟส คลาสที่ทำหน้าที่เป็น Abstraction มีชนิดของคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น RefinedAbstraction มีชนิดของคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น ConcreteImplementtorA มีชนิดของคลาสรูปธรรม และคลาสที่ทำหน้าที่เป็น ConcreteImplementtorB มีชนิดของคลาสรูปธรรม

2) คลาสที่ทำหน้าที่ Implementor ต้องมีการอิมพลีเมนต์อย่างน้อย 1 คลาส คือ คลาส ConcreteImplementtorA และคลาส ConcreteImplementtorB ด้วยความความสัมพันธ์แบบเจนเนอรัลไลเซชันเท่านั้น

3) คลาสที่ทำหน้าที่เป็น Abstraction ต้องมีความสัมพันธ์แบบอะกรีเกชันกับคลาส Implementor และมีการอิมพลีเมนต์อย่างน้อย 1คลาสคือ คลาส RefinedAbstraction ที่มีความความสัมพันธ์แบบเจนเนอรัลไลเซชันเท่านั้น

จากแบบรูปมาตรฐาน Bridge สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ของจำนวนเส้นความสัมพันธ์ที่ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้



รูปที่ 3.5 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Bridge

จากรูปที่ 3.5 แผนภาพกราฟแบบรูป Bridge สามารถอธิบายตามลำดับของเส้นความสัมพันธ์ได้ ดังนี้

1) กำหนดโหนด A แทน แผนภาพคลาส Abstraction โดยที่ โหนด A มีเส้นความสัมพันธ์เส้นเข้าหาโหนด จำนวน 2 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชันและความสัมพันธ์แบบอะกรีเกชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,1,0,1,0,0,0,0,0,0,0)

2) กำหนดโหนด B แทน แผนภาพคลาส Implementor โดยที่ โหนด B มีเส้นความ สัมพันธ์เส้นเข้าหาโหนด จำนวน 2 เส้น ซึ่งทั้งสองเส้นมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,2,0,0,0,0,0,0,0,0,0)

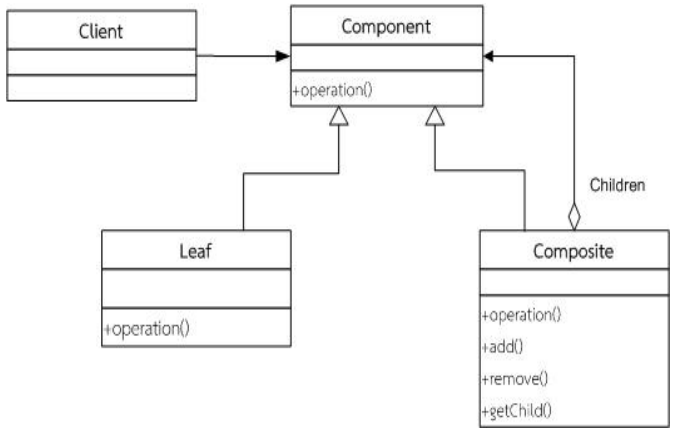
3) กำหนดโหนด C แทน แผนภาพคลาส RefinedAbstraction โดยที่ โหนด C มีเส้นความสัมพันธ์เส้นออกจากโหนด จำนวน 1 เส้นซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,0,0,0,0,0,0,0,0)

4) กำหนดโหนด D แทน แผนภาพคลาส ConcreteImplementtorA โดยที่ โหนด D มีเส้นความสัมพันธ์เส้นออกจากโหนด จำนวน 1 เส้นซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,0,0,0,0,0,0,0,0)

5) กำหนดโหนด E แทน แผนภาพคลาส ConcreteImplementtorB โดยที่ โหนด E มีเส้นความสัมพันธ์เส้นออกจากโหนด จำนวน 1 เส้นซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,0,0,0,0,0,0,0,0)

3.2.3. แบบรูปการออกแบบ Composite

ส่วนประกอบของแบบรูปการออกแบบ Composite สามารถอธิบายรายละเอียดได้ในรูปที่ 3.6 ดังนี้



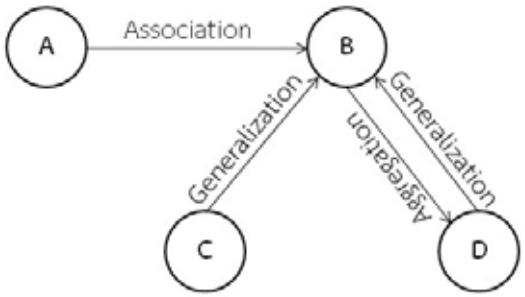
รูปที่ 3.6 โครงสร้างแผนภาพแบบรูปการออกแบบ Composite [5]

1) ประกอบด้วยคลาสที่ทำหน้าที่ Component ซึ่งมีชนิดของคลาสเป็นคลาสคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น Composite มีชนิดของคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น Leaf มีชนิดของคลาสรูปธรรม และคลาสที่ทำหน้าที่เป็น Client มีชนิดของคลาสรูปธรรม

2) คลาสที่ทำหน้าที่ Component ต้องมีการารอิมพลีเมนต์อย่างน้อย 2 คลาส คือ คลาส Composite และคลาส Leaf ด้วยความความสัมพันธ์แบบเจนเนอรัลไลเซชันเท่านั้น และและยังมีความสัมพันธ์แบบแอสโซซิเอชันกับคลาสที่เป็น Client

3) คลาสที่ทำหน้าที่ Composite ต้องมีความสัมพันธ์แบบอะกรีเกชันไปยังคลาส Component เท่านั้น

จากแบบรูปมาตรฐาน Composite สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ของจำนวนเส้นความสัมพันธ์ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้



รูปที่ 3.7 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Composite

จากแบบรูปการออกแบบ Composite ที่อธิบายไว้ข้างต้นในรูปที่ 3.7 สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ที่ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้

1) กำหนดโหนด A แทน แผนภาพคลาส Client โดยที่ โหนด A มีเส้นความสัมพันธ์เส้นออกจากโหนด จำนวน 1 เส้นซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,1,0,0,0,1,0,0,0,0,0,0)

2) กำหนดโหนด B แทน แผนภาพคลาส Component โดยที่ โหนด B มี เส้นความสัมพันธ์เส้นเข้าหาโหนด จำนวน 3 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน จำนวน 1 เส้น แบบเจนเนอรัลไลเซชันจำนวน 2 เส้น นอกจากนี้ยังมีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบอะกรีเกชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,2,0,0,1,0,0,0,0,0,0)

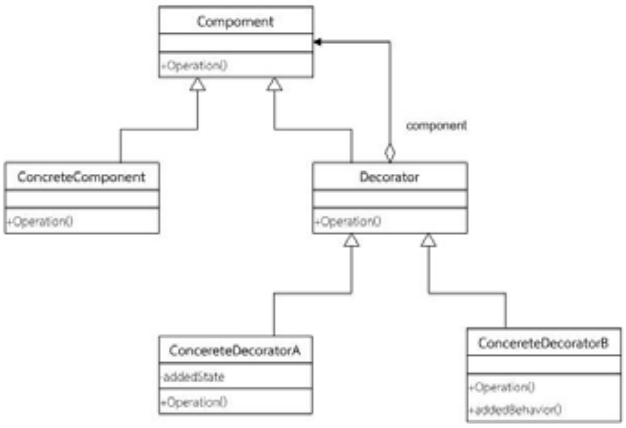
3) กำหนดโหนด C แทน แผนภาพคลาส Leaf โดยที่ โหนด C มีเส้นความสัมพันธ์เส้นออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,1,0,0,0,1,0,0,0,0,0,0)

4) กำหนดโหนด D แทน แผนภาพคลาส Composite โดยที่ โหนด D มี มีเส้นความสัมพันธ์เส้นเข้าหาโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบอะกรีเกชันและจำนวนเส้น

ความสัมพันธ์เส้นออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,1,0,0,0,0,0,0,0)

3.2.4. แบบรูปการออกแบบ Decorator

ส่วนประกอบของแบบรูปการออกแบบ Decorator สามารถอธิบายรายละเอียดได้รูปที่ 3.8 ดังนี้



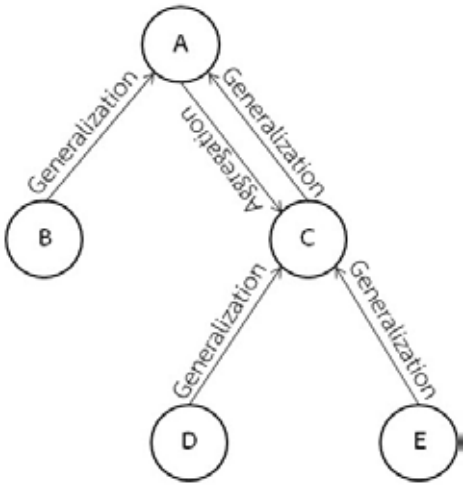
รูปที่ 3.8 โครงสร้างแผนภาพแบบรูปการออกแบบ Decorator [5]

1) ประกอบด้วยคลาสที่ทำหน้าที่ Component ซึ่งมีชนิดของคลาสเป็นคลาสคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น ConcreteComponent มีชนิดของคลาสรูปธรรมคลาสที่ทำหน้าที่เป็น Decorator มีชนิดของคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น ConcreteComponentA และคลาสที่ทำหน้าที่เป็น ConcreteComponentB มีชนิดของคลาสรูปธรรม

2) คลาสที่ทำหน้าที่ Component ต้องมีการารอิมพลีเมนต์อย่างน้อย 2 คลาส คือ คลาส ConcreteComponent และคลาส Decorator ด้วยความสัมพันธ์แบบเจนเนอรัลไลเซชันเท่านั้น

3) คลาสที่ทำหน้าที่เป็น Decorator ต้องมีการารอิมพลีเมนต์อย่างน้อย 1 คลาส คือ คลาส ConcreteComponentA และคลาส ConcreteComponentA ด้วยความความสัมพันธ์แบบเจนเนอรัลไลเซชันเท่านั้น และต้องมีความสัมพันธ์แบบอะกรีเกชันกับคลาส Component

จากแบบรูปมาตรฐาน Decorator สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ของจำนวนเส้นความสัมพันธ์ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้



รูปที่ 3.9 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Decorator

จากรูปที่ 3.9 แผนภาพกราฟแบบรูป Decorator สามารถอธิบายตามลำดับของเส้นความสัมพันธ์ได้ ดังนี้

1) กำหนดโหนด A แทน แผนภาพคลาส Component โดยที่ โหนด A มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 2 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน และเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบอะกรีเกชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,2,0,0,1,0,0,0,0,0,0)

2) กำหนดโหนด B แทน แผนภาพคลาส ConcreteComponent โดยที่โหนด B มีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,0,0,0,0,0,0,0,0)

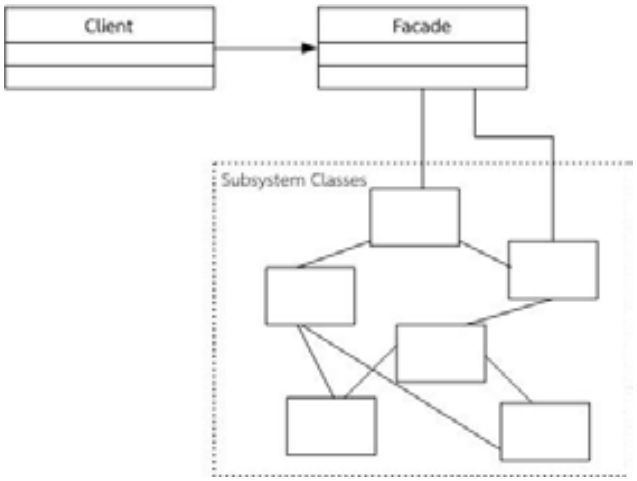
3) กำหนดโหนด C แทน แผนภาพคลาส Decorator โดยที่โหนด C มีเส้นความ สัมพันธ์เข้าหาโหนด ซึ่งมีความสัมพันธ์แบบอะกรีเกชัน และมีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,1,0,0,0,0,0,0,0)

4) กำหนดโหนด D แทน แผนภาพคลาส ConcreteComponentA โดยที่โหนด D เส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,0,0,0,0,0,0,0,0)

5) กำหนดโหนด E แทน แผนภาพคลาส ConcreteComponentB โดยที่โหนด E เส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,0,1,0,0,0,0,0,0,0,0)

3.2.5. แบบรูปการออกแบบ Façade

ส่วนประกอบของแบบรูปการออกแบบ Façade สามารถอธิบายรายละเอียดได้ในรูปที่ 3.10 ดังนี้

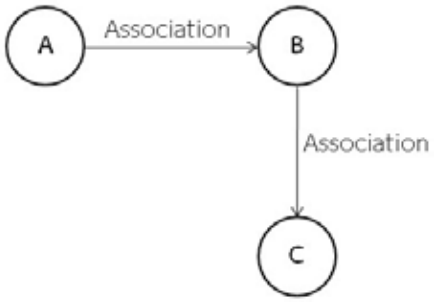


รูปที่ 3.10 โครงสร้างแผนภาพแบบรูปการออกแบบ Façade [5]

1) ประกอบด้วยคลาสที่ทำหน้าที่ Façade ซึ่งมีชนิดของคลาสเป็นคลาสคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น Client มีชนิดของคลาสรูปธรรม และคลาสย่อยๆ (Subsystem Classes) ที่ถูกรวบรวมอยู่ในลักษณะของแพ็กเก็จ (Package)

2) คลาสที่ทำหน้าที่ Façade มีความสัมพันธ์แบบแอสโซซิเอชันกับคลาสที่เป็น Client และมีความสัมพันธ์แบบแอสโซซิเอชันกับแพ็กเก็จที่รวมคลาสย่อยๆ อย่างน้อย 1 แพ็กเก็จ หรือคลาสย่อยจะไม่อยู่ในแพ็กเก็จก็ได้เช่นกัน

จากแบบรูปมาตรฐาน Façade สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ของจำนวนเส้นความสัมพันธ์ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้



รูปที่ 3.11 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Façade

จากรูปที่ 3.11 แผนภาพกราฟแบบรูป Façade สามารถอธิบายตามลำดับของเส้นความสัมพันธ์ได้ ดังนี้

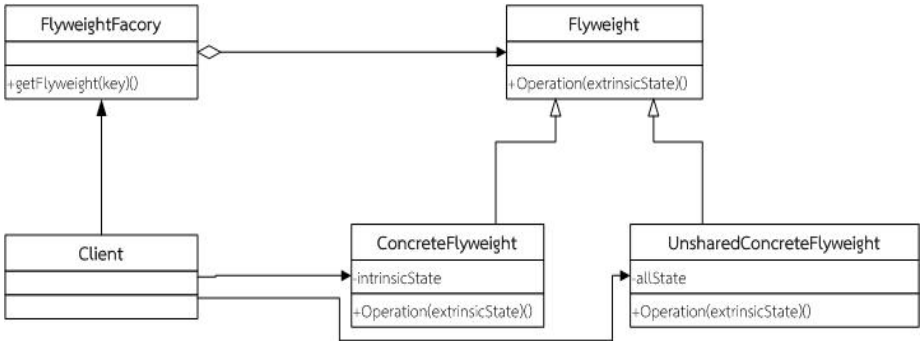
1) กำหนดโหนด A แทน แผนภาพคลาส Client โดยที่โหนด A มีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,1,0,0,0,0,0,0,0,0,0,0)

2) กำหนดโหนด B แทน แผนภาพคลาส Façade โดยที่ โหนด B มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน และมีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,1,0,0,0,0,0,0,0,0,0,0)

3) กำหนดโหนด C แทน แผนภาพคลาสที่เป็นแพ็กเก็จของคลาสย่อย โดยที่ โหนด C มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,0,0,0,0,0,0,0,0,0,0)

3.2.6. แบบรูปการออกแบบ Flyweight

ส่วนประกอบของแบบรูปการออกแบบ Flyweight สามารถอธิบายรายละเอียดได้ในรูปที่ 3.12ดังนี้



รูปที่ 3.12 โครงสร้างแผนภาพแบบรูปการออกแบบ Flyweight [5]

1) ประกอบด้วยคลาสที่ทำหน้าที่ Flyweight ซึ่งมีชนิดของคลาสเป็นคลาสคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น FlyweightFactory มีชนิดของคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น ConcreteFlyweight มีชนิดของคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น UnsharedConcreteFlyweight มีชนิดของคลาสรูปธรรม และคลาสที่ทำหน้าที่เป็น Client มีชนิดของคลาสรูปธรรม

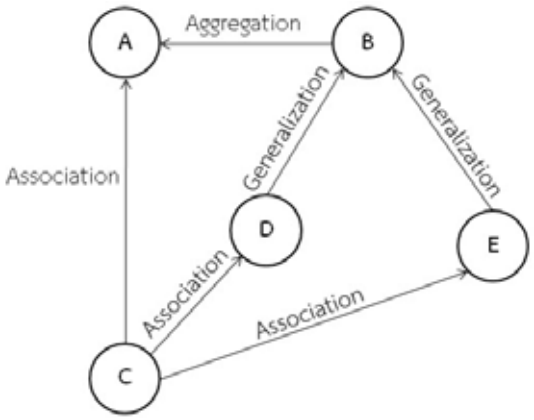
2) คลาสที่ทำหน้าที่ Flyweight ต้องมีการารอิมพลีเมนต์อย่างน้อย 2 คลาส คือ คลาส ConcreteFlyweight และ คลาส UnsharedConcreteFlyweight ด้วยความสัมพันธ์แบบเจนเนอรัลไลเซชันเท่านั้น

3) คลาสที่ทำหน้าที่เป็น FlyweightFactory ต้องมีความสัมพันธ์แบบอะกรีเกชันกับคลาส Flyweight และมีความสัมพันธ์แบบแอสโซซิเอชันกับคลาสที่เป็น Client

4) ที่ทำหน้าที่เป็น ConcreteFlyweight มีความสัมพันธ์แบบแอสโซซิเอชันกับคลาสที่เป็น Client

5) ที่ทำหน้าที่เป็น UnsharedConcreteFlyweight มีความสัมพันธ์แบบแอสโซซิเอชันกับคลาสที่เป็น Client

จากแบบรูปมาตรฐาน Flyweight สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ของจำนวนเส้นความสัมพันธ์ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้



รูปที่ 3.13 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Flyweight

จากรูปที่ 3.13 แผนภาพกราฟแบบรูป Flyweight สามารถอธิบายตามลำดับของเส้นความสัมพันธ์ได้ ดังนี้

1) กำหนดโหนด A แทน แผนภาพคลาส FlyweightFactory โดยที่โหนด A มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 2 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน และเส้นความสัมพันธ์แบบอะกรีเกชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,0,0,1,0,0,0,0,0,0,0)
2) กำหนดโหนด B แทน แผนภาพคลาส Flyweight โดยที่โหนด B มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 2 เส้น ซึ่งทั้งสองเส้นมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน และมีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีเส้นความสัมพันธ์แบบอะกรีเกชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,2,0,0,1,0,0,0,0,0,0)

3) กำหนดโหนด C แทน แผนภาพคลาส Client โดยที่โหนด C มีเส้นความสัมพันธ์ออกจากโหนด จำนวน 3 เส้น ซึ่งทั้งสามเส้นมีความสัมพันธ์แบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,3,0,0,0,0,0,0,0,0,0,0)

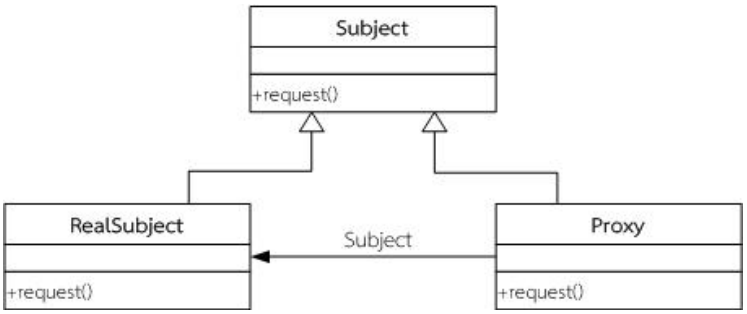
4) กำหนดโหนด D แทน แผนภาพคลาส ConcreteFlyweight โดยที่โหนด D มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน และมีเส้น

ความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,0,1,0,0,0,0,0,0,0,0)

5) กำหนดโหนด E แทน แผนภาพคลาส ConcreteFlyweight โดยที่โหนด E มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชัน และมีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,0,1,0,0,0,0,0,0,0,0)

3.2.7. แบบรูปการออกแบบ Proxy

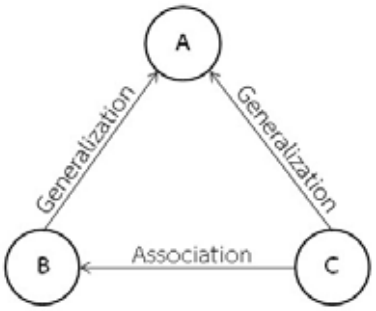
ส่วนประกอบของแบบรูปการออกแบบ Proxy สามารถอธิบายรายละเอียดได้ในรูปที่ 3.14 ดังนี้



รูปที่ 3.14 โครงสร้างแผนภาพแบบรูปการออกแบบ Proxy [5]

1) ประกอบด้วยคลาสที่ทำหน้าที่ Subject ซึ่งมีชนิดของคลาสเป็นคลาสคลาสรูปธรรม คลาสที่ทำหน้าที่เป็น RealSubject มีชนิดของคลาสรูปธรรม และคลาสที่ทำหน้าที่เป็น Proxy มีชนิดของคลาสรูปธรรม
2) คลาสที่ทำหน้าที่ Subject ต้องมีการารอิมพลีเมนต์อย่างน้อย 2 คลาส คือ คลาส RealSubject และ คลาส Proxy ด้วยความสัมพันธ์แบบเจนเนอรัลไลเซชันเท่านั้น
3) คลาสที่ทำหน้าที่เป็น Proxy มีความสัมพันธ์แบบแอสโซซิเอชันกับคลาสที่เป็น RealSubject

จากแบบรูปมาตรฐาน Proxy สามารถแปลงให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ของจำนวนเส้นความสัมพันธ์ได้ทำการเทียบลำดับของเส้นความสัมพันธ์กับตารางที่ 3.1 ดังนี้



รูปที่ 3.15 แผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Proxy

จากรูปที่ 3.15 แผนภาพกราฟแบบรูป Proxy สามารถอธิบายตามลำดับของเส้นความสัมพันธ์ได้ ดังนี้

1) กำหนดโหนด A แทน แผนภาพคลาส Subject โดยที่โหนด A มีเส้นความสัมพันธ์เข้าหาโหนด จำนวน 2 เส้น ซึ่งทั้งสองเส้นมีความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,0,2,0,0,0,0,0,0,0,0,0)

2) กำหนดโหนด B แทน แผนภาพคลาส RealSubject โดยที่โหนด B มีเส้นความ สัมพันธ์เข้าหาโหนด จำนวน 1 เส้น ซึ่งมีความสัมพันธ์แบบแอสโซซิเอชันและมีเส้นความสัมพันธ์ออกจากโหนด จำนวน 1 เส้น ซึ่งมีเส้นความสัมพันธ์แบบเจนเนอรัลไลเซชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (1,0,0,1,0,0,0,0,0,0,0,0)

3) กำหนดโหนด C แทน แผนภาพคลาส Proxy โดยที่โหนด C มีเส้นความสัมพันธ์ออกจากโหนด จำนวน 2 เส้น ซึ่งมีเส้นความสัมพันธ์แบบเจนเนอรัลไลเซชันและแบบแอสโซซิเอชัน สามารถสรุปเป็นลำดับเส้นความสัมพันธ์ได้ (0,1,0,1,0,0,0,0,0,0,0,0)

3.3. ขั้นตอนการสร้างตารางคุณลักษณะท้องถิ่น (LPT-P) และตารางความสัมพันธ์ (RT-P) ของแบบรูปการออกแบบ

จากขั้นตอนการแปลงแบบรูปการออกแบบให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ที่ได้อธิบายไว้ในข้อ 3.2 แล้วนั้นผลลัพธ์ที่ได้จะเป็นข้อมูลของแต่ละแบบรูปการออกแบบซึ่งจะถูกจัดเก็บลงในตารางคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ โดยข้อมูลส่วนนี้จะใช้เป็นส่วนที่ใช้ตรวจจับแบบรูปการออกแบบกับกราฟของระบบในลำดับต่อไป ซึ่งในแต่ละแบบรูปการออกแบบสามารถสรุปได้ ดังนี้

3.3.1. แบบรูปการออกแบบ Adapter

ข้อมูลจากแผนภาพคลาสและแผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Adapter รูปที่ 3.3 สามารถนำมาสร้างคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ได้ดังตารางที่ 3.5 และตารางที่ 3.6

ตารางที่ 3.5 คุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Adapter			
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,1,0,0,0,0,0,0,0,0,0,0)
Target	Interface	Request()	(1,0,0,0,0,0,0,0,0,0,1,0)
Adapter	Concrete	Request()	(0,1,0,0,0,0,0,0,0,0,0,1)
Adaptee	Concrete	SpecificRequest()	(1,0,0,0,0,0,0,0,0,0,0,0)

ตารางที่ 3.6 ความสัมพันธ์ของแบบรูปการออกแบบ Adapter

ความสัมพันธ์ของแบบรูปการออกแบบ Adapter	
คลาสต้นทาง	คลาสปลายทาง
Client	Target
Adapter	Target
Adapter	Adaptee

3.3.2. แบบรูปการออกแบบ Bridge

ข้อมูลจากแผนภาพคลาสและแผนภาพกราฟเชิงความสัมพันธ์แบบรูปการออกแบบ Bridge รูปที่ 3.5 สามารถนำมาสร้างคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ได้ดังตารางที่ 3.7 และตารางที่ 3.8

ตารางที่ 3.7 คุณลักษณะท้องถิ่นของรูปการออกแบบ Bridge

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Bridge			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Abstraction	Concrete	operation()	(0,0,1,0,1,0,0,0,0,0,0,0)
Implementor	Interface	implementation()	(0,0,2,0,1,0,0,0,0,0,0,0)
RefinedAbstraction	Concrete	operation()	(0,0,0,1,0,0,0,0,0,0,0,0)
ConcreteImplementtorA	Concrete	implementation()	(0,0,0,1,0,0,0,0,0,0,0,0)
ConcreteImplementtorB	Concrete	implementation()	(0,0,0,1,0,0,0,0,0,0,0,0)

ตารางที่ 3.8 ความสัมพันธ์ของแบบรูปการออกแบบ Bridge

ความสัมพันธ์ของแบบรูปการออกแบบ Bridge	
คลาสต้นทาง	คลาสปลายทาง
RefinedAbstraction	Abstraction
Implementor	Abstraction
ConcreteImplementtorA	Implementor
ConcreteImplementtorB	Implementor

3.3.3. แบบรูปการออกแบบ Composite

ข้อมูลจากแผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Composite รูปที่ 3.7 สามารถนำมาสร้างคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ได้ดังตารางที่ 3.9 และตารางที่ 3.10

ตารางที่ 3.9 คุณลักษณะท้องถิ่นของรูปการออกแบบ Composite

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Composite			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,1,0,0,0,1,0,0,0,0,0,0)
Component	Concrete	operation()	(1,0,2,0,0,1,0,0,0,0,0,0)
Leaf	Concrete	operation()	(0,1,0,0,0,1,0,0,0,0,0,0)
Composite	Concrete	operation() ,add(), remove(),getChild()	(0,0,0,1,1,0,0,0,0,0,0,0)

ตารางที่ 3.10 ความสัมพันธ์ของแบบรูปการออกแบบ Composite

ความสัมพันธ์ของแบบรูปการออกแบบ Composite	
คลาสต้นทาง	คลาสปลายทาง
Client	Component
Component	Composite
Leaf	Component
Composite	Component

3.3.4. แบบรูปการออกแบบ Decorator

ข้อมูลจากแผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Decorator รูปที่ 3.9 สามารถนำมาสร้างคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ได้ดังตารางที่ 3.11 และตารางที่ 3.12

ตารางที่ 3.11 คุณลักษณะท้องถิ่นของรูปการออกแบบ Decorator

คุณลักษณะท้องถิ่นของรูปการออกแบบ Decorator			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Component	Concrete	operation()	(0,0,2,0,0,1,0,0,0,0,0,0)
ConcreteComponent	Concrete	operation()	(0,0,0,1,0,0,0,0,0,0,0,0)
Decorator	Concrete	operation()	(0,0,0,1,1,0,0,0,0,0,0,0)
ConcreteComponentA	Concrete	operation()	(0,0,0,1,0,0,0,0,0,0,0,0)
ConcreteComponentB	Concrete	operation(), addBehavior()	(0,0,0,1,0,0,0,0,0,0,0,0)

ตารางที่ 3.12 ความสัมพันธ์ของแบบรูปการออกแบบ Decorator

ความสัมพันธ์ของแบบรูปการออกแบบ Decorator	
คลาสต้นทาง	คลาสปลายทาง
ConcreteComponent	Component
Component	Decorator
Decorator	Component
ConcreteComponentA	Decorator
ConcreteComponentB	Decorator

3.3.5. แบบรูปการออกแบบ Façade

ข้อมูลจากแผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Facade รูปที่ 3.11 สามารถนำมาสร้างคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ได้ดังตารางที่ 3.13 และตารางที่ 3.14

ตารางที่ 3.13 คุณลักษณะท้องถิ่นของรูปการออกแบบ Facade

คุณลักษณะท้องถิ่นของรูปการออกแบบ Facade			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,1,0,0,0,0,0,0,0,0,0,0)
Façade	Concrete	-	(1,1,0,0,0,0,0,0,0,0,0,0)
Package	Concrete	-	(1,0,0,0,0,0,0,0,0,0,0,0)

ตารางที่ 3.14 ความสัมพันธ์ของแบบรูปการออกแบบ Facade

ความสัมพันธ์ของแบบรูปการออกแบบ Facade	
คลาสต้นทาง	คลาสปลายทาง
Client	Façade
Façade	Package

3.3.6. แบบรูปการออกแบบ Flyweight

ข้อมูลจากแผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Flyweight รูปที่ 3.13 สามารถนำมาสร้างคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ได้ดังตารางที่ 3.15 และตารางที่ 3.16

ตารางที่ 3.15 คุณลักษณะท้องถิ่นของรูปการออกแบบ Flyweight

คุณลักษณะท้องถิ่นของรูปการออกแบบ Flyweight			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
FlyweightFactory	Concrete	getFlyweight()	(1,0,0,0,1,0,0,0,0,0,0,0)
Flyweight	Concrete	operation()	(0,0,2,0,0,1,0,0,0,0,0,0)
Client	Concrete	-	(0,3,0,0,0,0,0,0,0,0,0,0)
ConcreteFlyweight	Concrete	operation()	(1,0,0,1,0,0,0,0,0,0,0,0)
UnsharedConcreteFlyweight	Concrete	operation()	(1,0,0,1,0,0,0,0,0,0,0,0)

ตารางที่ 3.16 ความสัมพันธ์ของแบบรูปการออกแบบ Flyweight

ความสัมพันธ์ของแบบรูปการออกแบบ Flyweight	
คลาสต้นทาง	คลาสปลายทาง
Flyweight	FlyweightFactory
Client	FlyweightFactory
Client	ConcreteFlyweight
ConcreteFlyweight	Flyweight
Client	UnsharedConcreteFlyweight
UnsharedConcreteFlyweight	Flyweight

3.3.7. แบบรูปการออกแบบ Proxy

ข้อมูลจากแผนภาพกราฟเชิงความสัมพันธ์ของแบบรูปการออกแบบ Proxy รูปที่ 3.15 สามารถนำมาสร้างคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ได้ดังตารางที่ 3.17 และตารางที่ 3.18

ตารางที่ 3.17 คุณลักษณะท้องถิ่นของรูปการออกแบบ Proxy

ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Proxy			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Subject	Concrete	request()	(0,0,2,0,0,0,0,0,0,0,0,0)
RealSubject	Concrete	request()	(1,0,0,1,0,0,0,0,0,0,0,0)
Proxy	Concrete	request()	(0,1,0,1,0,0,0,0,0,0,0,0)

ตารางที่ 3.18 ความสัมพันธ์ของแบบรูปการออกแบบ Proxy

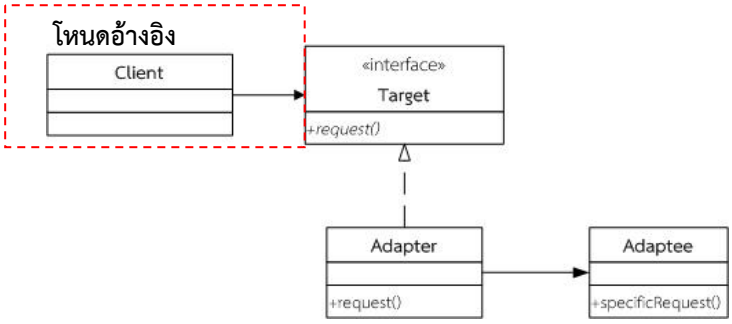
ความสัมพันธ์ของแบบรูปการออกแบบ Proxy	
คลาสต้นทาง	คลาสปลายทาง
RealSubject	Subject
Proxy	Subject
Proxy	RealSubject

3.4. การเลือกโหนดอ้างอิงสำหรับแต่ละแบบรูปการออกแบบ

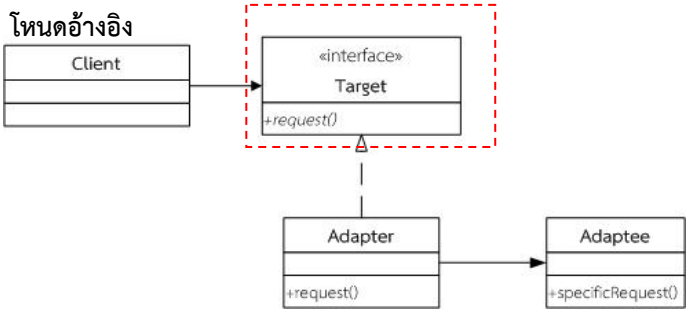
จากการสร้างตารางคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ให้กับแต่ละแบบรูปการออกแบบแล้วนั้นในส่วนต่อไปจะเป็นการกำหนดโหนดอ้างอิงของแต่แบบรูปการออกแบบ โดยในงานวิจัยนี้ได้พิจารณาถึงคลาสที่อยู่ในแบบรูปการออกแบบที่มีลักษณะเฉพาะที่แตกต่างกันไปในแต่ละแบบรูป ซึ่งมีการพิจารณาโหนดอ้างอิงของแต่ละแบบรูปจากคลาสที่มีเส้นความสัมพันธ์เข้า-ออก ที่มีลักษณะเฉพาะของแต่ละแบบรูปการออกแบบ เพื่อที่จะนำมาใช้ในการตรวจจับแบบรูปการออกแบบก่อนการตรวจจับรายละเอียด ซึ่งการสร้างโหนดอ้างอิงของแต่ละแบบรูปการออกแบบดังอธิบายต่อไปนี้

3.4.1. โหนดอ้างอิงของแบบรูปการออกแบบ Adapter

จากโครงสร้างของแบบรูปการออกแบบ Adapter กำหนดคุณสมบัติของโหนดอ้างอิงโดยพิจารณาจากคลาสรูปธรรมที่มีเส้นความสัมพันธ์ออกจากโหนดเป็นแบบแอสโซซิเอชัน มีจำนวนอย่างน้อย 1 เส้น ซึ่งจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (0,1,0,0,0,0,0,0,0,0,0) ตามรูปที่ 3.16 จากนั้นจะทำการตรวจสอบคลาสที่เกี่ยวข้องปลายทางที่เชื่อมโยงกับโหนดเริ่มต้นจะต้องมีประเภทของคลาสเป็นอินเตอร์เฟสเท่านั้นและมีเส้นความสัมพันธ์เข้าแบบอิมพลีเมนต์เตชันหรือรีไรเซชันมาจากโหนดอ้างอิงตามรูปที่ 3.17



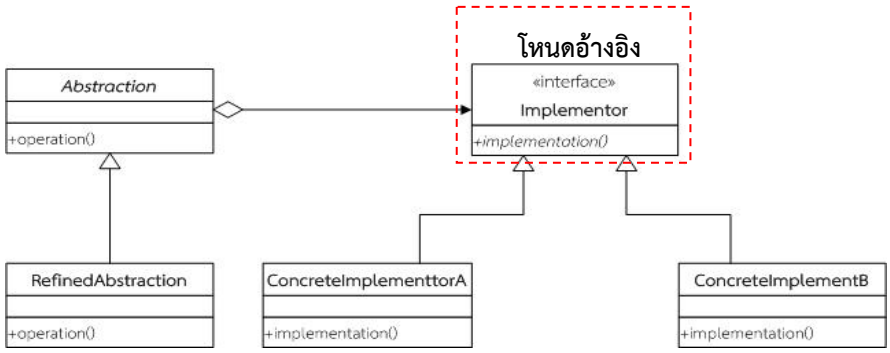
รูปที่ 3.16 โหนดอ้างอิงของแบบรูปการออกแบบ Adapter



รูปที่ 3.17 การตรวจสอบโหนดที่เกี่ยวของของแบบรูปการออกแบบ Adapter

3.4.2. โหนดอ้างอิงของแบบรูปการออกแบบ Bridge

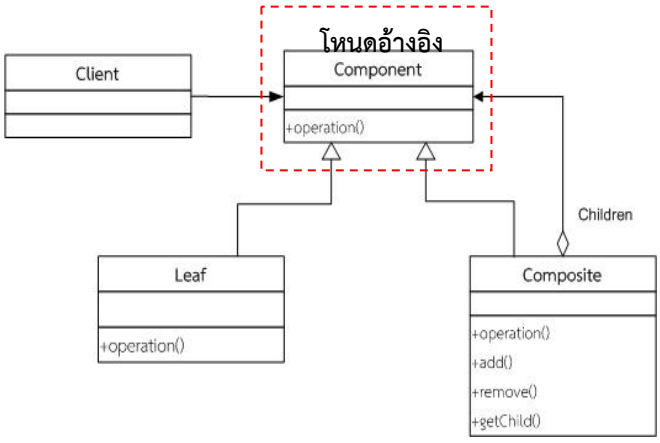
จากโครงสร้างของแบบรูปการออกแบบ Bridge กำหนดคุณสมบัติของโหนดอ้างอิงโดยพิจารณาจากคลาสที่มีชนิดเป็นอินเตอร์เฟสที่มีเส้นความสัมพันธ์เข้าหาโหนดเป็นแบบอะกรีเกชันและเส้นความสัมพันธ์เข้าหาโหนดแบบเจนเนอรัลไลเซชัน มีจำนวนอย่างน้อย 1 เส้น ซึ่งจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (0,0,2,0,1,0,0,0,0,0,0,0) ตามรูปที่ 3.18



รูปที่ 3.18 โหนดอ้างอิงของแบบรูปการออกแบบ Bridge

3.4.3. โหนดอ้างอิงของแบบรูปการออกแบบ Composite

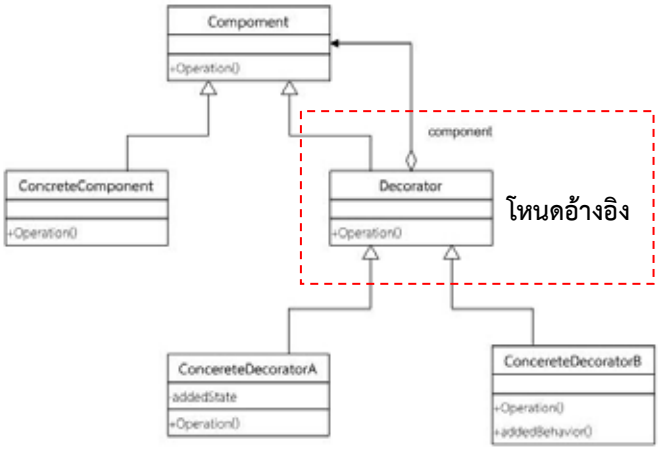
จากโครงสร้างของแบบรูปการออกแบบ Composite กำหนดคุณสมบัติของโหนดอ้างอิงโดยพิจารณาจากคลาสที่มีชนิดเป็นรูปธรรมที่มีเส้นความสัมพันธ์เข้าหาโหนดเป็นแบบแอสโซซิเอชัน มีจำนวนอย่างน้อย 1 เส้น มีเส้นความสัมพันธ์เข้าหาโหนดเป็นแบบเจนเนอรัลไลเซชัน มีจำนวนอย่างน้อย 1 เส้น และมีเส้นความสัมพันธ์ออกจากโหนดแบบอะกรีเกชัน มีจำนวนอย่างน้อย 1 เส้น ซึ่งจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (1,0,2,0,1,0,0,0,0,0,0,0) ตามรูปที่ 3.19



รูปที่ 3.19 โหนดอ้างอิงของแบบรูปการออกแบบ Composite

3.4.4. โหนดอ้างอิงของแบบรูปการออกแบบ Decorator

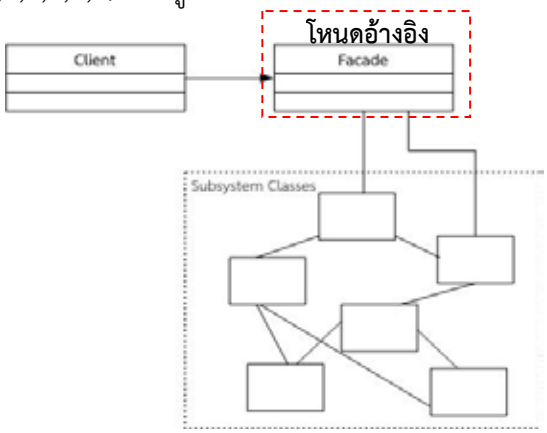
จากโครงสร้างของแบบรูปการออกแบบ Decorator กำหนดคุณสมบัติของโหนดอ้างอิงโดยพิจารณาจากคลาสที่มีชนิดเป็นรูปธรรมที่มีเส้นความสัมพันธ์เข้าหาโหนดเป็นแบบแบบอะกรีเกชันจำนวนอย่างน้อย 1 เส้น มีเส้นความสัมพันธ์เข้าหาโหนดแบบเจนเนอรัลไลเซชัน มีจำนวนอย่างน้อย 1 เส้น และมีเส้นความสัมพันธ์ออกจากโหนดเป็นแบบเจนเนอรัลไลเซชัน มีจำนวนอย่างน้อย 1 เส้น ซึ่งจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (0,0,2,1,1,0,0,0,0,0,0,0) ตามรูปที่ 3.20



รูปที่ 3.20 โหนดอ้างอิงของแบบรูปการออกแบบ Decorator

3.4.5. โหนดอ้างอิงของแบบรูปการออกแบบ Façade

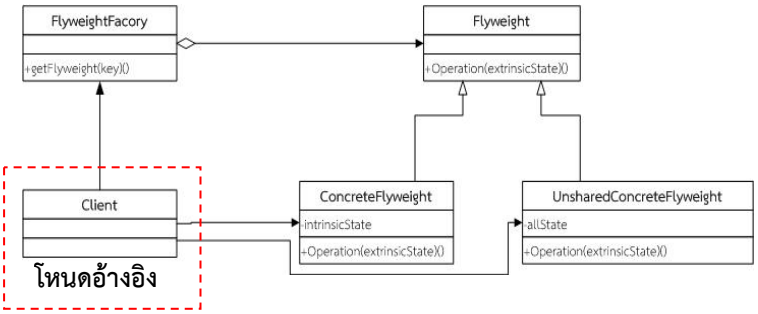
จากโครงสร้างของแบบรูปการออกแบบ Façade กำหนดคุณสมบัติของโหนดอ้างอิงโดยพิจารณาจากคลาสที่มีชนิดเป็นรูปธรรมที่มีเส้นความสัมพันธ์เข้าหาโหนดเป็นแบบแอสโซซิเอชันจำนวนอย่างน้อย 1 เส้น และมีเส้นความสัมพันธ์ออกจากโหนดแบบแอสโซซิเอชัน มีจำนวนอย่างน้อย 1 เส้น ที่ไปสัมพันธ์การคลาสย่อยหรือคลาสอื่นๆ โดยโหนดอ้างอิงจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (1,2,0,0,0,0,0,0,0,0,0,0) ตามรูปที่ 3.21



รูปที่ 3.21 โหนดอ้างอิงของแบบรูปการออกแบบ Façade

3.4.6. โหนดอ้างอิงของแบบรูปการออกแบบ Flyweight

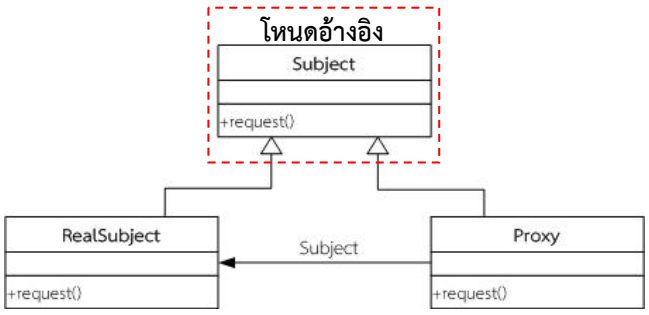
จากโครงสร้างของแบบรูปการออกแบบ Flyweight กำหนดคุณสมบัติของโหนดอ้างอิงโดยพิจารณาจากคลาสที่มีชนิดเป็นรูปธรรมที่มีเส้นความสัมพันธ์ออกจากโหนดเป็นแบบแอสโซซิเอชันจำนวนอย่างน้อย 3 เส้น ไปยังคลาสที่เกี่ยวข้องซึ่งคลาสที่เกี่ยวข้องมีชนิดเป็นรูปธรรม ซึ่งจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (0,3,0,0,0,0,0,0,0,0,0) ตามรูปที่ 3.22



รูปที่ 3.22 โหนดอ้างอิงของแบบรูปการออกแบบ Flyweight

3.4.7. โหนดอ้างอิงของแบบรูปการออกแบบ Proxy

จากโครงสร้างของแบบรูปการออกแบบ Proxy กำหนดคุณสมบัติของโหนดอ้างอิงโดยพิจารณาจากคลาสที่มีชนิดเป็นรูปธรรมที่มีเส้นความสัมพันธ์เข้าหาโหนดเป็นแบบเจนเนอรัลไลเซชันจำนวนอย่างน้อย 2 เส้น ซึ่งจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (0,0,2,0,0,0,0,0,0,0,0) ตามรูปที่ 3.23



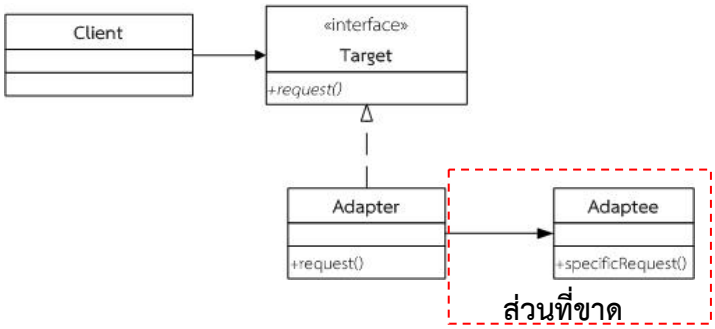
รูปที่ 3.23 โหนดอ้างอิงของแบบรูปการออกแบบ Proxy

3.5. การกำหนดความเหมือนแบบรูปการออกแบบ

การกำหนดความเหมือของแต่ละแบบรูปจะพิจารณาตามลำดับการตรวจจับที่เริ่มจากโหนดอ้างอิงของแต่ละแบบรูปและทำการตรวจจับในรายละเอียดของคลาสต่อไปที่มีความสัมพันธ์กันตามลำดับที่ระบุไว้ในตารางความสัมพันธ์ (RT-P) โดยจะกำหนดความเหมือนแบบรูปการออกแบบ แต่ละแบบรูป ดังต่อไปนี้

3.5.1. ความเหมือนแบบรูปการออกแบบ Adapter

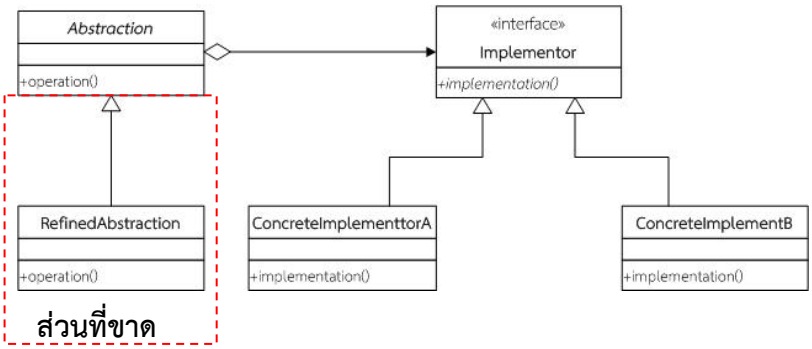
การกำหนดความเหมือนของแบบรูปการออกแบบ Adapter เพื่อใช้ในการระบุถึงการตรวจจับนั้นจะต้องตรวจพบคลาสที่ตรงตามคุณลักษณะของคลาส Client คลาส Target และคลาส Adapter ตามลำดับ ซึ่งจากคลาส Adapter มีเส้นความสัมพันธ์แบบแอสโซซิเอชันเชื่อมโยงไปยังคลาส Adaptee หากไม่สามารถตรวจจับตามคุณลักษณะของคลาส Adaptee ได้จะแสดงผลการตรวจจับว่าขาดเส้นความสัมพันธ์แบบแอสโซซิเอชัน ดังรูปที่ 3.24



รูปที่ 3.24 การกำหนดความเหมือนของแบบรูปการออกแบบ Adapter

3.5.2. ความเหมือนแบบรูปการออกแบบ Bridge

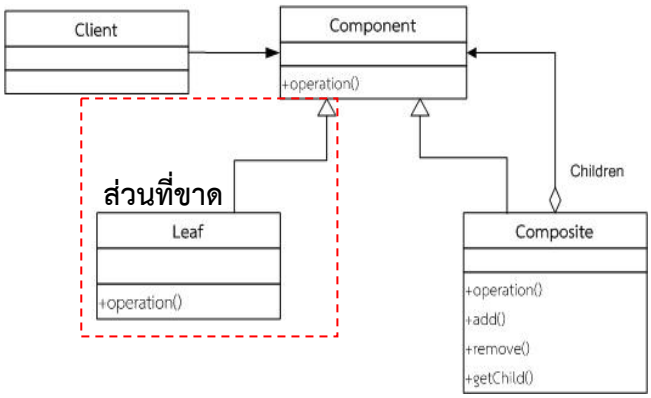
การกำหนดความเหมือนของแบบรูปการออกแบบ Bridge เพื่อใช้ในการระบุถึงการตรวจจับนั้นจะต้องตรวจพบคลาสที่ตรงตามคุณลักษณะของคลาส Implementor คลาส Abstraction คลาส ConcretImplementA และคลาส ConcretImplementB ตามลำดับ ซึ่งจากคลาส RefineAbstraction มีเส้นความสัมพันธ์เข้าหาคลาส Abstraction แบบเจนเนอรัลไลเซชัน หากไม่สามารถตรวจจับตามคุณลักษณะของคลาส RefineAbstraction ได้จะแสดงผลการตรวจจับว่าขาดเส้นความสัมพันธ์แบบเจนเนอรัลไลเซชัน ดังรูปที่ 3.25



รูปที่ 3.25 การกำหนดความเหมือนของแบบรูปการออกแบบ Bridge

3.5.3. ความเหมือนแบบรูปการออกแบบ Conposite

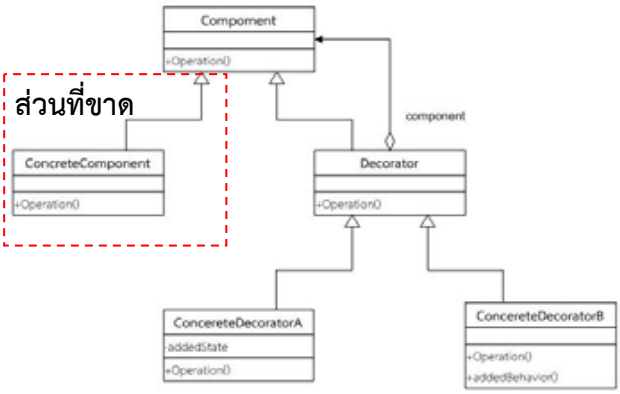
การกำหนดความเหมือนของแบบรูปการออกแบบ Composite เพื่อใช้ในการระบุถึงการตรวจจับนั้นจะต้องตรวจพบคลาสที่ตรงตามคุณลักษณะของคลาส Component คลาส Client และคลาส Composite ตามลำดับจากคลาส Leaf มีเส้นความสัมพันธ์เข้าหาคลาส Component แบบเจนเนอรัลไลเซชัน หากไม่สามารถตรวจจับตามคุณลักษณะของคลาส Leaf ได้จะแสดงผลการตรวจจับว่าขาดเส้นความสัมพันธ์แบบเจนเนอรัลไลเซชัน ดังรูปที่ 3.26



รูปที่ 3.26 การกำหนดความเหมือนของแบบรูปการออกแบบ Composite

3.5.4. ความเหมือนแบบรูปการออกแบบ Decorator

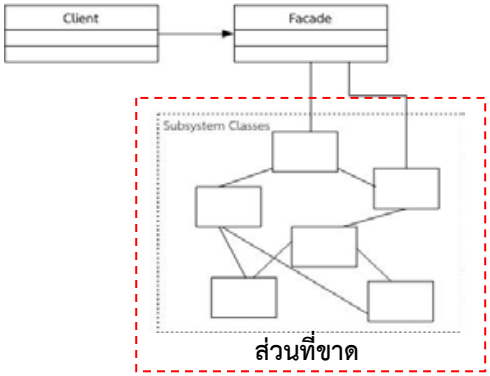
การกำหนดความเหมือนของแบบรูปการออกแบบ Decorator เพื่อใช้ในการระบุถึงการตรวจจับนั้นจะต้องตรวจพบคลาสที่ตรงตามคุณลักษณะของคลาส Decorator คลาส Component คลาส ConcreteDecoratorA และคลาส ConcreteDecoratorB ตามลำดับจากคลาส ConcreteComponent มีเส้นความสัมพันธ์เข้าหาคลาส Component แบบเจนเนอรัลไลเซชัน หากไม่สามารถตรวจจับตามคุณลักษณะของคลาส ConcreteComponent ได้จะแสดงผลการตรวจจับว่าขาดเส้นความสัมพันธ์แบบเจนเนอรัลไลเซชัน ดังรูปที่ 3.27



รูปที่ 3.27 การกำหนดความเหมือนของแบบรูปการออกแบบ Decorator

3.5.5. ความเหมือนแบบรูปการออกแบบ Façade

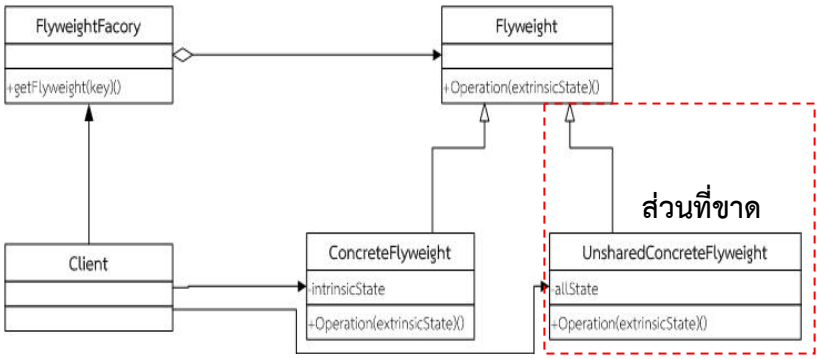
การกำหนดความเหมือนของแบบรูปการออกแบบ Façade เพื่อใช้ในการระบุถึงการตรวจจับนั้นจะต้องตรวจพบคลาสที่ตรงตามคุณลักษณะของคลาส Facade และคลาส Client ตามลำดับจากคลาส Façade มีเส้นความสัมพันธ์ออกจากคลาสแบบแอสโซซิเอชัน ไปยังคลาสย่อยหรือคลาสใดๆ หากไม่สามารถตรวจจับตามคุณลักษณะของคลาสย่อยหรือคลาสใดๆ ได้จะแสดงผลการตรวจจับว่าขาดเส้นความสัมพันธ์แบบแอสโซซิเอชัน ดังรูปที่ 3.28



รูปที่ 3.28 การกำหนดความเหมือนของแบบรูปการออกแบบ Façade

3.5.6. ความเหมือนแบบรูปการออกแบบ Flyweight

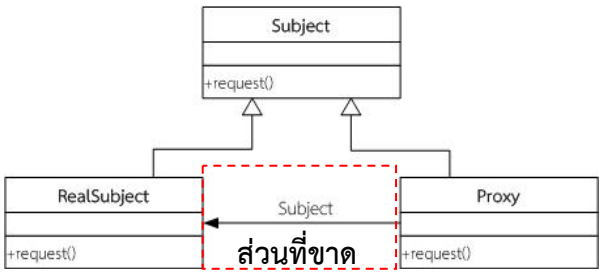
การกำหนดความเหมือนของแบบรูปการออกแบบ Flyweight เพื่อใช้ในการระบุถึงการตรวจจับนั้นจะต้องตรวจพบคลาสที่ตรงตามคุณลักษณะของคลาส Client คลาส FlyweightFactory คลาส Flyweight และคลาส ConcreteFlyweight ตามลำดับ จากคลาสUnsharedConcrete Flyweight มีเส้นความสัมพันธ์ออกจากคลาสแบบเจอเนอรัลไลเซชันไปยังคลาส Flyweight หากไม่สามารถตรวจจับตามคุณลักษณะของคลาส UnsharedConcreteFlyweight ได้จะแสดงผลการตรวจจับว่าขาดเส้นความสัมพันธ์แบบเจอเนอรัลไลเซชัน ดังรูปที่ 3.29



รูปที่ 3.29 การกำหนดความเหมือนของแบบรูปการออกแบบ Flyweight

3.5.7. ความเหมือนแบบรูปการออกแบบ Proxy

การกำหนดความเหมือนของแบบรูปการออกแบบ Proxy เพื่อใช้ในการระบุถึงการตรวจจับนั้นจะต้องตรวจพบคลาสที่ตรงตามคุณลักษณะของคลาส Subject คลาส Proxy และคลาส RealSubject ตามลำดับ ซึ่งจากคลาส Proxy ต้องมีเส้นความสัมพันธ์ออกไปยังคลาส RealSubject แบบแอสโซซิเอชันหากไม่สามารถตรวจจับตามคุณลักษณะของคลาส Proxy ได้จะแสดงผลการตรวจจับว่าขาดเส้นความสัมพันธ์แบบแอสโซซิเอชัน ดังรูปที่ 3.30



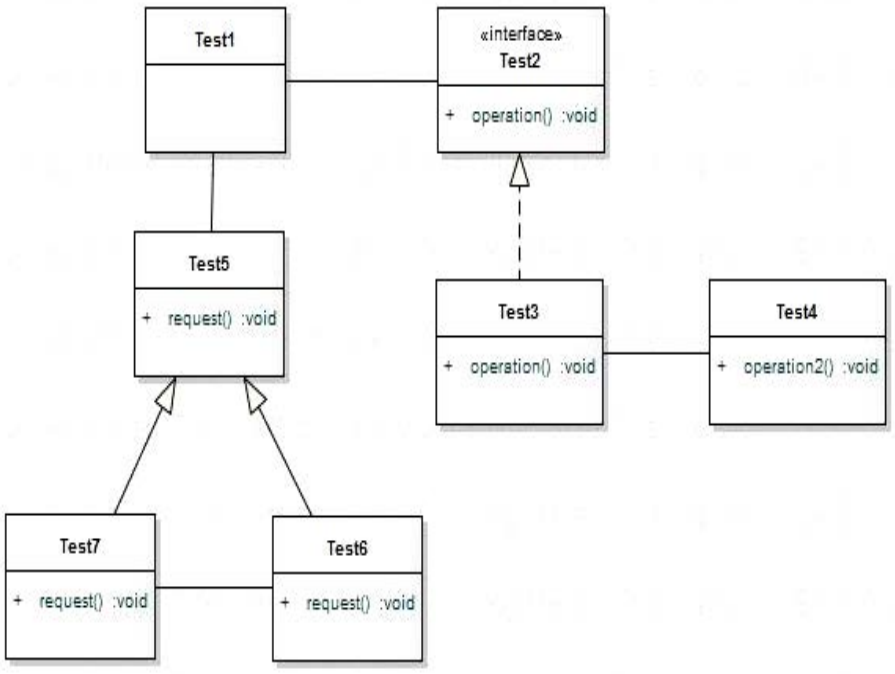
รูปที่ 3.30 การกำหนดความเหมือนของแบบรูปการออกแบบ Proxy

3.6. ขั้นตอนวิธีนำแผนภาพคลาสของระบบมาตรวจจับแบบรูปการออกแบบ

ในส่วนของแผนภาพคลาสของระบบที่จะทำมาตรวจจับแบบรูปการออกแบบนั้นจะต้องดำเนินการเช่นเดียวกันกับแบบรูปการออกแบบซึ่งขั้นตอนวิธีจะเริ่มตั้งแต่การแปลงแผนภาพคลาสของระบบเป็นกราฟ จากนั้นนำข้อมูลที่ได้จากแผนภาพคลาสจัดเก็บลงในตารางคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ เพื่อจะนำข้อมูลส่วนนี้มาใช้ในการเปรียบเทียบหาแบบรูปการออกแบบ ซึ่งสามารถอธิบายขั้นตอนจากตัวอย่างแผนภาพคลาสของระบบ ดังนี้

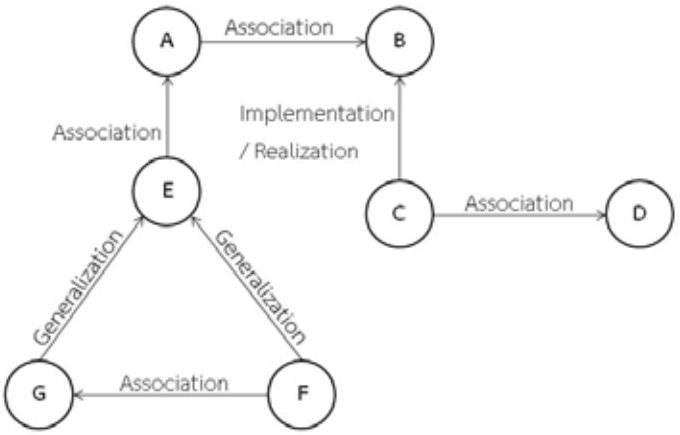
3.6.1. ตัวอย่างการแปลงแผนภาพคลาสระบบให้เป็นกราฟเชิงความสัมพันธ์

แผนภาพคลาสของระบบตัวอย่าง ที่มีการออกแบบตามมาตรฐานยูเอ็มแอลที่มีการระบุถึงชื่อคลาส ชนิดของคลาส เมธ็อดที่อยู่ในคลาสนั้น อีกทั้งยังมีเส้นความสัมพันธ์เชื่อมโยงกันระหว่างแต่ละคลาส สามารถแสดงได้ ดังรูปที่ 3.31



รูปที่ 3.31 ตัวอย่างแผนภาพคลาสที่ใช้ในการตรวจจับแบบรูปการออกแบบ

จากตัวอย่างแผนภาพคลาสตัวอย่างตามรูปที่ 3.31 ทำการแปลงแผนภาพคลาสให้อยู่ในรูปแบบของกราฟเชิงความสัมพันธ์ ดังแสดงในรูปที่ 3.32



รูปที่ 3.32 แผนภาพกราฟเชิงความสัมพันธ์ของแผนภาพคลาสตัวอย่าง

3.6.2. ตารางคุณลักษณะท้องถิ่น (LPT-S) และตารางความสัมพันธ์ (RT-S) ของแผนภาพคลาสระบบตัวอย่าง

ข้อมูลของแผนภาพคลาสตัวอย่างจะจัดเก็บลงในตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส ดังตารางที่ 3.19 ส่วนตารางความสัมพันธ์ของแผนภาพตัวอย่างแสดงได้ ดังตารางที่ 3.20 จากข้อมูลดังกล่าวจะถูกนำเข้าสู่กระบวนการตรวจจับแบบรูปการออกแบบต่อไป

ตารางที่ 3.19 คุณลักษณะท้องถิ่นของแผนภาพคลาสระบบตัวอย่าง

คุณลักษณะท้องถิ่นแผนภาพคลาสระบบตัวอย่าง			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Test1	Concrete	-	(1,1,0,0,0,0,0,0,0,0,0,0)
Test2	Interface	operation()	(1,0,0,0,0,0,0,0,0,0,1,0)
Test3	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,0,1)
Test4	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0,0)
Test5	Concrete	request()	(1,0,0,2,0,0,0,0,0,0,0,0)
Test6	Concrete	request()	(0,1,0,1,0,0,0,0,0,0,0,0)
Test7	Concrete	request()	(1,0,0,1,0,0,0,0,0,0,0,0)

ตารางที่ 3.20 ความสัมพันธ์ของแผนภาพคลาสระบบตัวอย่าง

ความสัมพันธ์ของแผนภาพคลาสระบบตัวอย่าง	
คลาสต้นทาง	คลาสปลายทาง
Test1	Test2
Test3	Test2
Test3	Test4
Test5	Test1
Test6	Test5
Test7	Test5
Test6	Test7

3.7. การปรับปรุงอัลกอริทึมการตรวจจับแบบรูปการออกแบบ

งานวิจัยนี้ ได้นำอัลกอริทึมจากงานวิจัยของ Manjari Gupta และคณะ [3] มาทำการปรับปรุงการตรวจจับแบรูปการออกแบบ ซึ่งจากงานวิจัย [3] ได้นำเสนออัลกอริทึมในการตรวจจับแบบรูปการออกแบบอธิบายได้ ดังนี้ ข้อมูลนำเข้าเป็นกราฟของแบบรูปการออกแบบ (DPG) และโมเดลกราฟ (MG) ทำการสร้างตารางเมทริกซ์จากจำนวนโหนดของ DPG และจำนวนโหนดของ MG โดยกำหนดให้เป็นตารางเมทริกซ์ P=(p_{ij}) ซึ่งใช้ในการจัดเจ็บค่าที่ไม่เหมือนกันของโหนด ตารางเมทริกซ์ที่สอง คือ B=(b_{ij}) ใช้ในการจับคู่โหนด ขั้นตอนแรกของอัลกอริทึมจะเริ่มต้นตารางเมทริกซ์ P กำหนดค่า p_{ij} = d (m1 (vi), m2 (vj)) เมื่อ p_{ij} คือ ระยะห่างระหว่าง v_i และ v_j ขั้นตอนแรกของการวนซ้ำเริ่มต้นที่เมทริกซ์ B ทำการกำหนดค่า(b_{ij}) ทั้งหมดเป็นศูนย์และกำหนดค่าเป็น 1 ขึ้นอยู่กับค่าที่ตรงกันกับเมทริกซ์ P ซึ่งในแต่ละแถวในเมทริกซ์ B ค่าน้อยที่สุดที่ตรงกับแถวเดียวกับเมทริกซ์ P จะให้ค่าเป็น 1 จากการจับคู่ที่เป็นไปได้จากเมทริกซ์ B ในส่วนของอัลกอริทึมการคำนวณค่าความคลาดเคลื่อนของโหนดและเส้นเชื่อม ทำการบันทึกค่าที่มีความคลาดเคลื่อนน้อยที่สุด ขั้นตอนที่สองทำการกำหนดค่าในแถวของเมทริกซ์ B เป็น 1 ของค่าน้อยสุดลำดับที่สองที่ตรงกันกับเมทริกซ์ P โดยอัลกอริทึมนี้จะทำการสกัดการจับคู่โหนดต่อโหนดจากเมทริกซ์ B และเพิ่มค่าลงเมทริกซ์ B ในขั้นตอนนี้ จะดำเนินการในขั้นตอนต่อไปส่วนเมทริกซ์ B’ ถูกนำมาเก็บค่าการจับคู่โหนดที่เป็นไปได้ที่ได้จากการพิจารณาจากอัลกอริทึม สำหรับเมทริกซ์ B จะใช้เป็นเมทริกซ์ชั่วคราว แต่ละขั้นตอนที่ดำเนินการในแถวของเมทริกซ์ B เสร็จสิ้น ในแต่ละแถวของเมทริกซ์ B จะมีลำดับแถวก่อนหน้าที่มีค่าของการ

จับคู่โหนดที่เป็นไปได้จากการตรวจสอบ สุดท้ายแล้วเมทริกซ์ B จะมีเฉพาะค่าที่จับคู่โหนดที่เป็นไปได้เท่านั้น จากขั้นตอนที่ได้อธิบายอยู่ในรูปของอัลกอริทึม ดังรูปที่ 3.33

```
Input: DPG and MG.
Output: matching between nodes in DPG and MG,
1. Initialize P as follows:
        For each pij, set pij= d (m1 (vi), m2 (vj)).
2. Initialize B as follows:
        For each bij , i =1,..., n and j = 1,...,m , set bij = 0.
3. While Current _ Phase < K
        If Current _ Phase = 1,
                Then for all i = 1,..., n
                        select the element with the smallest value in P that is not marked 1in B
                        and set it to 1 in B;
                        call Matching_Nodes(B).
        Else for all i = 1,..., n
        set B' = B
                for all j =1,...,m set bij = 0
                        select the element with the smallest
                        value in P that is not marked 1 in B'
                        and set it to 1 in B and B';
                        call Matching_Nodes(B);
                        set B = B'.
        If all elements in B are marked 1,
                Then set Current _ Phase = K
                Else add 1 to Current_Phase.
End
```

รูปที่ 3.33 อัลกอริทึมจากงานวิจัยของ Manjari Gupta และคณะ [3]

โดยอัลกอริทึมข้างต้นมีการเรียกใช้ฟังก์ชัน(B) เพื่อหาการจับคู่โหนด ซึ่งฟังก์ชัน (B) มีขั้นตอนการหาค่า (B) ในการจับคู่โหนด สำหรับการตรวจสอบในการจับคู่ใน B ดังนี้

1) คำนวณค่าความคลาดเคลื่อนของจุดเชื่อมต่อที่มีความคล้ายคลึงกันระหว่างการจับคู่โหนด $|MG_{t1}\text{-}DPG_{t1}|+ |MG_{t2}\text{-}DPG_{t2}|+ |MG_{t3}\text{-}DPG_{t3}|$,เมื่อ MG_{ti} คือ i^{th} ที่เป็นส่วนประกอบของโหนดที่อยู่ในโมเดลกราฟและ DPG_{ti} ที่เป็นส่วนประกอบของจุดเชื่อมต่อที่จับคู่กับที่อยู่ในกราฟแบบรูปการออกแบบ

2) เพิ่มค่าความคลาดเคลื่อนที่ตรงกันของการจับคู่ความเส้นเชื่อม โดยใช้ความแตกต่างของการจับคู่เส้นเชื่อม ถ้ามีค่าเป็น 0 ทั้งเส้นเชื่อมที่มีความตรงกันก็จะแสดงว่ามีความสัมพันธ์เหมือนกัน นอกจากนี้ยังกำหนดค่าของการจับคู่ค่าความคลาดเคลื่อนของ

เส้นเชื่อมให้กับค่า q เมื่อ q เป็นค่าที่มีจำนวนเป็นบวกที่มาก ซึ่งจะแสดงถึงความไม่ตรงกันของความสัมพันธ์ของเส้นเชื่อม

3) ทำการบันทึกค่าของการจับคู่ที่มีความคลาดเคลื่อนต่ำสุด

จากอัลกอริทึมข้างต้นมีการตรวจจับแบบรูปการออกแบบโดยการนำตารางเมทริกซ์ใช้ในการดำเนินการตรวจจับแบบรูปการออกแบบ ซึ่งอัลกอริทึมนี้มุ่งเน้นการจับคู่โหนดเป็นหลักซึ่งค่าที่ใส่ลงในตารางเมทริกซ์จะเป็นค่าที่มีความเป็นไปได้จากการจับคู่โหนด ทั้งนี้จากตารางเมทริกซ์ที่ใช้ดังกล่าวไม่ได้แสดงถึงการจับคู่เส้นเชื่อมหรือเส้นความสัมพันธ์ที่มีระหว่างโหนดและผลลัพธ์ของอัลกอริทึมนี้จะแสดงผลจากการตรวจจับว่ามีความเหมือนกับแบบรูปการออกแบบทุกประการ เหมือนกับแบบรูปการออกแบบบางส่วน หรือไม่มีเหมือนแบบรูปการออกแบบ จากผลของอัลกอริทึมนี้ทางผู้วิจัยจึงได้ทำการปรับปรุงแนวคิดจากอัลกอริทึมดังกล่าวให้สามารถพิจารณาเส้นความสัมพันธ์ระหว่างโหนดนอกจากนี้แล้วยังพิจารณาถึงคุณลักษณะของโหนดนั้นด้วย ซึ่งผลลัพธ์ของอัลกอริทึมที่ทำการปรับปรุงจะสามารถแสดงค่าความเหมือนแบบรูปการออกแบบและเส้นความสัมพันธ์ที่ขาดหายจากแบบรูปที่มีความเหมือน โดยอัลกอริทึมที่ได้ทำการปรับปรุงจะแบ่งออกเป็น 2 ขั้นตอน คือ การตรวจจับโหนดอ้างอิงของแบบรูปการออกแบบ และการตรวจจับรายละเอียดของแบบรูปการออกแบบ ซึ่งแสดงอัลกอริทึมทั้ง 2 ขั้นตอนได้ ดังรูปที่ 3.23 และรูปที่ 3.24

ขั้นตอนที่ 1: อัลกอริทึมการตรวจจับโหนดอ้างอิงของแต่ละแบบรูปการออกแบบใดๆ

ในขั้นตอนการตรวจจับโหนดอ้างอิงของแต่ละแบบรูปการออกแบบใดๆ จากตารางคุณลักษณะท้องถิ่นของแผนภาพคลาสระบบ มาทำการตรวจจับโหนดอ้างอิงของแต่ละแบบรูปการออกแบบ ซึ่งจะมีการตรวจหาคุณลักษณะท้องถิ่นของแต่ละโหนดที่ตรงกับโหนดอ้างอิง สำหรับโหนดที่ทำการตรวจจับแล้วมีคุณลักษณะตรงกับโหนดอ้างอิงจะทำการบันทึกเป็นโหนดที่ทำการตรวจจับแล้ว ผลลัพธ์ของขั้นตอนที่ 1 นั้นจะเป็นกองซ้อนของโหนดอ้างอิง ดังแสดงอัลกอริทึม ตามรูปที่ 3.34

Phase 1: Finder Reference Node for design pattern in System Local Properties Table
Input: 1) Reference Node *R*
2) System local properties table (LPT-S)
Output: 1) Stack *S*

//Compare properties value between LPT-S and Reference Node each of patterns
For each row_i of LPT-S
For (j=1 to 12)
if component(j, row_i) >= component(j, *R*)
then
flag = true
Else
flag = false
break forloop
End if
End for
//Adding Reference Node name into column 5 in LPT-S at the rows are same properties after that push the row of node in Stack

If (flag)
addColumn (LPT-S,$column_5$)
If propertyEquals(R, row_i)
Then insertField(5, row_i, getField(1, *R*))
push (row_i, Stack*S*)

End for row_i

รูปที่ 3.34 อัลกอริทึมการตรวจจับโหนดอ้างอิงของแต่ละแบบรูปการออกแบบใดๆ

คำอธิบายฟังก์ชันการตรวจจับโหนดอ้างอิง

1) addColumn (LPT-S, $Column_i$) คือ ฟังก์ชันที่เพิ่มคอลัมน์สำหรับใส่ชื่อโหนดอ้างอิงลงในตาราง LPT-S

2) propertyEquals (rowx,rowy) คือ ฟังก์ชันที่เปรียบเทียบค่า property ของ rowx และ rowy ว่าเท่ากันหรือไม่ โดย property ของ rowx ที่ประกอบไปด้วยคอลัมน์ที่แสดงชื่อคลาส ชนิดของคลาส ชื่อเมท็อด และเส้นความสัมพันธ์ระหว่างคลาส

3) insertField (column 5, rowx, getField (column 1, X)) คือ ฟังก์ชันที่เพิ่มชื่อของโหนดอ้างอิง X ที่อยู่ในคอลัมน์ที่ 1 จากตาราง LPT-P มาใส่ลงในคอลัมน์ที่ 5 ของ rowx

4) getField (column 1, X) คือ ฟังก์ชันที่ดึงค่าในคอลัมน์ที่ 1 ของโหนดอ้างอิง X จากตาราง LPT-P

5) push (rowx, Stack X) คือ ฟังก์ชันที่นำ rowx จัดเก็บลงใน Stack X

ขั้นตอนที่ 2: อัลกอริทึมการตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบใดๆ

ในขั้นตอนนี้เป็นการตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบใดๆ จากที่ได้ทำการตรวจจับโหนดอ้างอิงแล้ว ซึ่งขั้นตอนนี้จะนำเข้าข้อมูลจากขั้นตอนที่ 1 นำมาตรวจจับโหนดที่มีความสัมพันธ์อยู่ลำดับถัดไประหว่างตารางความสัมพันธ์ของระบบกับตารางความสัมพันธ์ของแบบรูปการออกแบบ จากนั้นนำมาตรวจสอบคุณลักษณะของโหนดต่อไป ผลลัพธ์จากขั้นตอนนี้จะได้เป็นรายชื่อของโหนดที่มีความแม่นยำและรายชื่อโหนดที่ไม่แม่นยำ ดังแสดงอัลกอริทึม ตามรูปที่ 3.35 (ก) และ รูปที่ 3.35 (ข)

```
Phase 2: Searching the Chaining of Expected Design Pattern
Input:  1) Stack S
        2) Pattern Relational Table (RT-P)
        3) System Relational Table (RT-S)
        4) System local properties table (LPT-S)
        5) Pattern local properties table (LPT-P)
Output : 1) List of nodes

// Pop peek node in Stack to check target node in RT-S with next node RT-P and compare
properties of node
addColumn (LPT-S,column6)

do while not empty (Stack S)
        Temp = Pop (Stack S)
        insertField(6, locate(Temp, LPT-S), '1')
        target1(target T1, Temp name )
        target2(target T2, Temp match)
        For each target node T1 of RT-S
                For each target node T2 of RT-P
                        For ( j=1 to 12 )
                                if component( j, row_i) >= component( j, row_j)
                                then
                                        flag = true
                                Else
                                        flag = false
                                        break forloop
                                End if
                        End for
                End for targetT2
```

รูปที่ 3.35 (ก) อัลกอริทึมตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบใดๆ

```
//Adding node name of patterns form LPT-S  into column 5 in LPT-S at the rows are same
properties after that push the row of node in Stack
                If (flag)
                        If propertyEquals(T2, T1)
                        Then insertField(5, row_i, getField(target, T2))
                                push(T1, StackS)
                        Else
                                flag F = true;
        End for target T1
// Check source node in RT-S with next node RT-P and compare properties of node
        source1(source S1, Temp name)
        source2(source S2, Temp match)
        For each source node S1 of RT-S
                For each source node S2 of RT-P
                        For ( j=1 to 12 )
                                if component( j, row_i) >= component( j, row_j)
                                then
                                        flag = true
                                Else
                                         flag = false
                                        break forloop
                                End if
                        End for
                End for source T2
//Adding node name of patterns form LPT-P into column 5 in LPT-S at the rows are same
properties after that push the row of node in Stack

                If (flag)
                        If propertyEquals(T2, S1)
                        Then insertField(5, row_i, getField(source, S2))
                                push(S1, Stack S)
                        Else
                                flagF = true;
        End for source node S1
//Result of inexact matching node
        If flagF
                Not match
        End If
End do
```

รูปที่ 3.35 (ข) อัลกอริทึมตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบใดๆ

คำอธิบายฟังก์ชันตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบใดๆ

1) insertField (column 6, locate(*Node*, LPT-S), value ‘1’) คือ ฟังก์ชันที่เพิ่มค่า 1 ลงในคอลัมน์ที่ 6 ของตาราง LPT-P เพื่อแสดงถึงการตรวจสอบโหนดนั้นแล้ว

2) target1(column 2 of RT-S, node name at row_i) คือ ฟังก์ชันที่ทำการค้นหาโหนดต่อไปในตาราง RT-S จาก row_i เพื่อให้ได้ชื่อโหนดปลายทาง

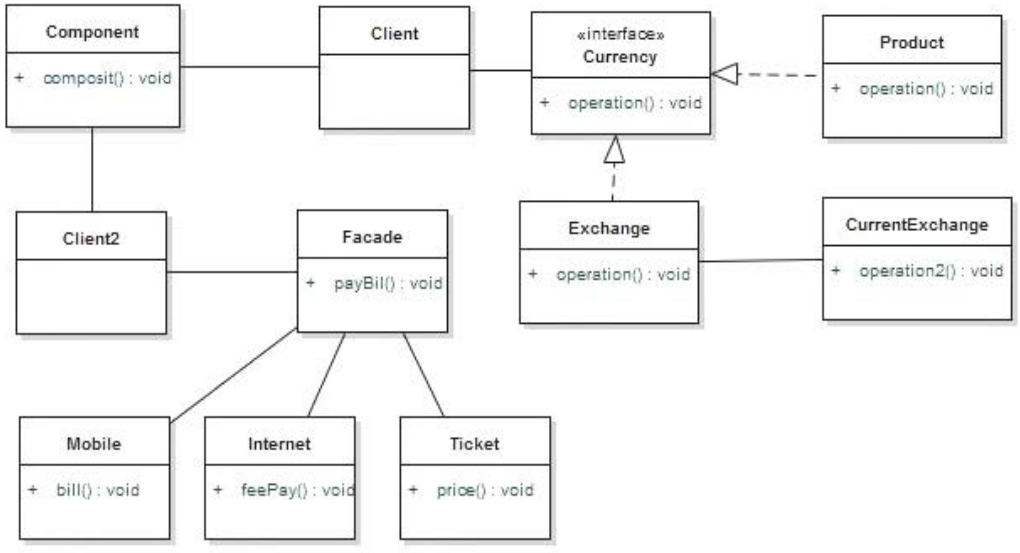
3) target2(column 2 of RT-P, node name at row_i) คือ ฟังก์ชันที่ทำการค้นหาโหนดต่อไปในตาราง RT-P จาก row_i เพื่อให้ได้ชื่อโหนดปลายทาง

4) source1(column 1 of RT-P, node name at row_i) คือ ฟังก์ชันที่ทำการค้นหาโหนดก่อนหน้าในตาราง RT-S จาก row_i เพื่อให้ได้ชื่อโหนดต้นทาง

5) source2(column 1of RT-P, node name at row_i) คือ ฟังก์ชันที่ทำการค้นหาโหนดก่อนหน้าในตาราง RT-P จาก row_i เพื่อให้ได้ชื่อโหนดต้นทาง

3.8. ตัวอย่างการตรวจจับแบบรูปการออกแบบ Adapter จากอัลกอริทึมที่ปรับปรุง

ตัวอย่างแผนภาพคลาสระบบ ดังรูปที่ 3.36 สำหรับการตรวจจับแบบรูปการออกแบบ Adapter โดยใช้อัลกอริทึมที่ได้ทำปรับปรุง



รูปที่ 3.36 ตัวอย่างแผนภาพคลาสระบบสำหรับการตรวจจับแบบรูปการออกแบบ

จากแผนภาพคลาสระบบตัวอย่าง ทำการจัดเก็บข้อมูลลงในตารางคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ ได้ดังตารางที่ 3.21 และ ตารางที่ 3.22

ตารางที่ 3.21 คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ (LPT-S)

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0,0)
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0,0)
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,0,2,0)
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,0,1)
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,0,1)
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0,0)
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0,0)
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0,0)
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0,0)
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0,0)
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0,0)

ตารางที่ 3.22 ความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S)

ความสัมพันธ์ของแผนภาพคลาสของระบบ	
คลาสต้นทาง	คลาสปลายทาง
Client	Component
Client	Currency
Product	Currency
Exchange	Currency
Exchange	CurentExchane
Client2	Component
Client2	Facade
Facade	Mobile
Facade	Internet
Facade	Ticket

สำหรับตารางคุณลักษณะท้องถิ่นและตารางความสัมพันธ์แบบรูปการออกแบบ Adapter ที่นำมาใช้ในการตรวจจับแสดงได้ดังตารางที่ 3.23 และตารางที่ 3.24

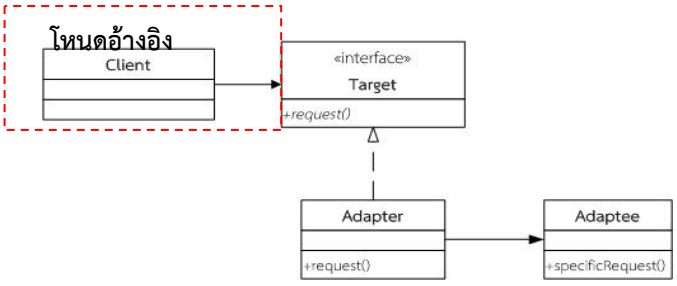
ตารางที่ 3.23 คุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P)

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Adapter			
ชื่อคลาส	ชนิดของคลาส	ชื่อเมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,1,0,0,0,0,0,0,0,0,0,0)
Target	Interface	Request()	(1,0,0,0,0,0,0,0,0,0,1,0)
Adapter	Concrete	Request()	(0,1,0,0,0,0,0,0,0,0,0,1)
Adaptee	Concrete	SpecificRequest()	(1,0,0,0,0,0,0,0,0,0,0,0)

ตารางที่ 3.24 ตารางความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P)

ความสัมพันธ์ของแบบรูปการออกแบบ Adapter	
คลาสต้นทาง	คลาสปลายทาง
Client	Target
Adapter	Target
Adapter	Adaptee

จากโครงสร้างของแบบรูปการออกแบบ Adapter กำหนดคุณสมบัติของโหนดอ้างอิงเป็นคลาส Client โดยพิจารณาจากคลาสรูปธรรมที่มีเส้นความสัมพันธ์ออกจากโหนดเป็นแบบแอสโซซิเอชัน มีจำนวนอย่างน้อย 1 เส้น ซึ่งจะมีจำนวนเส้นความสัมพันธ์ระหว่างคลาส คือ (0,1,0,0,0,0,0,0,0,0,0,0)
ดังรูปที่ 3.37



รูปที่ 3.37 โหนดอ้างอิงของแบบรูปการออกแบบ Adapter

อัลกอริทึมที่ใช้ในการตรวจจับแบบรูปการออกแบบ Adapter

```
Phase 1: Finder Reference Node for Adapter pattern
Input:  1) Reference Node R
        2) System local properties table (LPT-S)
        3) Pattern local properties table (LPT-P)
        4) Pattern Relational Table (RT-P)
        5) System Relational Table (RT-S)
Output: 1) Stack S
//Compare properties value between LPT-S and Reference Node each of patterns
For each row_i of LPT-S
    For ( j=1 to 12 )
        if component( j, row_i) >= component( j, R)
        then
            flag = true
        Else
            flag = false
            break forloop
        End if
    End for
//Adding Reference Node name into column 5 in LPT-S at the rows are same properties after that
push the row of node in Stack
    If (flag)
        addColumn (LPT-S,column_5)
        If propertyEquals(R, row_i)
        Then insertField(5, row_i, getField(1, R))
        push (row_i, StackS)
End for row_i
//Check target node in RT-S with next node RT-P of reference adapter node and compare
properties of node
target1(target T1, Temp name )
target2(target T2, Temp match)
    For each target node T1 of RT-S
        For each target node T2 of RT-P
            For ( j=1 to 12 )
                if component( j, row_i) >= component( j, row_i)
                then
                    flag = true
                Else
                    flag = false
                    break forloop
                End if
            End for
        End for targetT2
        If (flag)
            If propertyEquals(T2, T1)
            push(Temp, StackS)
    End for target T1
```

รูปที่ 3.38 อัลกอริทึมที่ใช้ในการตรวจจับโหนดอ้างอิงแบบรูปการออกแบบ Adapter

ขั้นตอนที่ 1: ตรวจจับโหนดอ้างอิงของแบบรูปการออกแบบ Adapter

1) ข้อมูลนำเข้า 1. โหนดอ้างอิงของแบบรูปการออกแบบ Adapter จากตารางที่ 3.23 ดังรูปที่ 3.39

คุณลักษณะท้องถิ่นของโหนดอ้างอิง (R)

Client	Concrete	-	(0,1,0,0,0,0,0,0,0,0,0)

รูปที่ 3.39 คุณลักษณะท้องถิ่นโหนดอ้างอิงของแบบรูปการออกแบบ Adapter

2. ตารางคุณลักษณะท้องถิ่นแผนภาพระบบ (LPT-S) ดังรูปที่ 3.40

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ			
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0)
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,2,0)
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0)
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0)
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0)
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0)

รูปที่ 3.40 คุณลักษณะท้องถิ่นแผนภาพระบบ (LPT-S)

3. ตารางความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S) ดังรูปที่ 3.41

ความสัมพันธ์ของแผนภาพคลาสของระบบ	
คลาสต้นทาง	คลาสปลายทาง
Client	Component
Client	Currency
Product	Currency
Exchange	Currency
Exchange	CurentExchane
Client2	Component
Client2	Facade
Facade	Mobile
Facade	Internet
Facade	Ticket

รูปที่ 3.41 ความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S)

4. ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P) ดังรูปที่ 3.42

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Adapter			
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,1,0,0,0,0,0,0,0,0,0)
Target	Interface	Request()	(1,0,0,0,0,0,0,0,0,1,0)
Adapter	Concrete	Request()	(0,1,0,0,0,0,0,0,0,0,1)
Adaptee	Concrete	SpecificRequest()	(1,0,0,0,0,0,0,0,0,0,0)

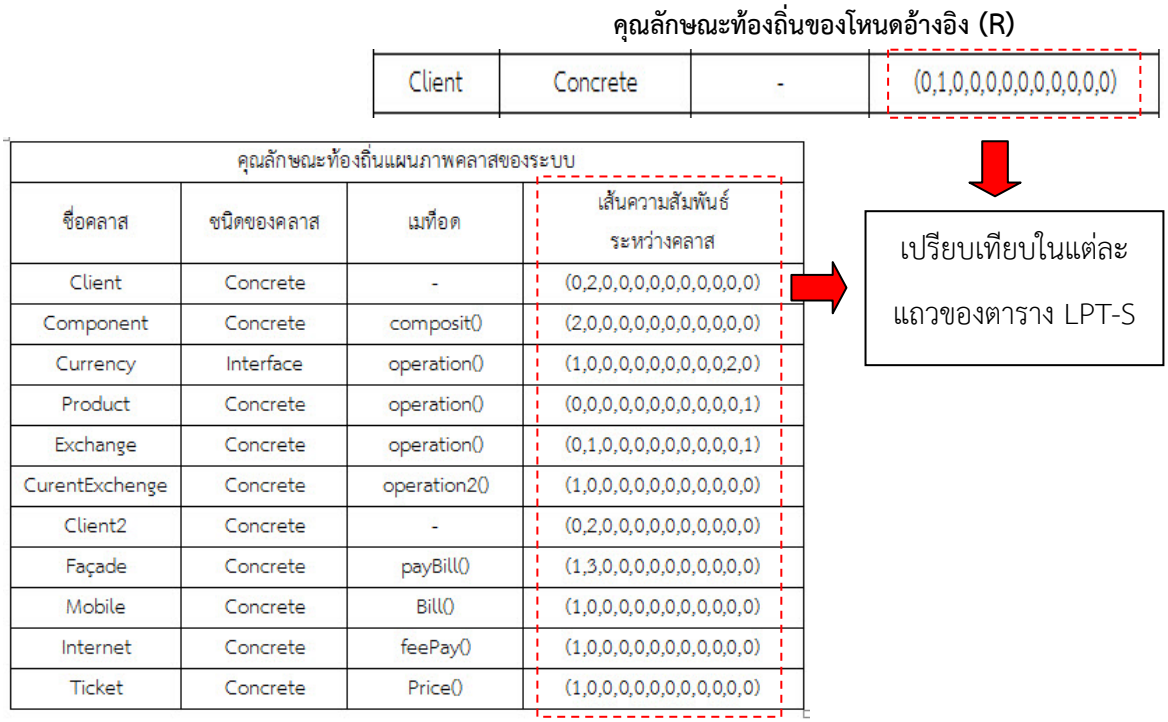
รูปที่ 3.42 คุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P)

5. ตารางความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P) ดังรูปที่ 3.43

ความสัมพันธ์ของแบบรูปการออกแบบ Adapter	
คลาสต้นทาง	คลาสปลายทาง
Client	Target
Adapter	Target
Adapter	Adaptee

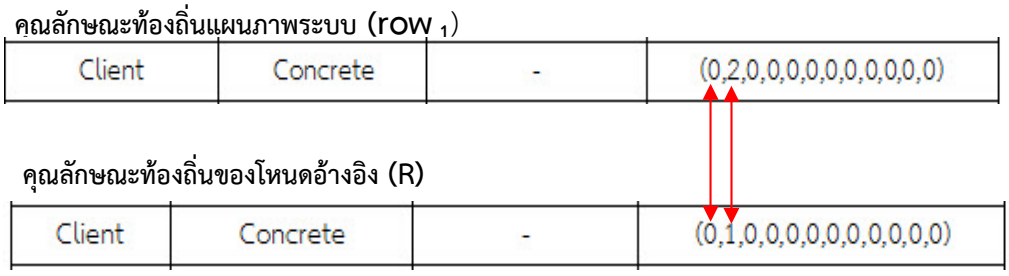
รูปที่ 3.43 ความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P)

2) ในแต่ละแถวของตารางคุณลักษณะท้องถิ่นแผนภาพระบบ จะทำการเปรียบเทียบในส่วนของคอลัมน์ที่ 4 คือ จำนวนเส้นความสัมพันธ์ระหว่างคลาสที่ระบุไว้เป็น 12 ลำดับกับโหนดอ้างอิง ดังรูปที่ 3.44



รูปที่ 3.44 การเปรียบเทียบ จำนวนเส้นความสัมพันธ์ระหว่างคลาส 12 ลำดับ

3) การเปรียบเทียบค่าแต่ละลำดับองค์ประกอบของคุณลักษณะท้องถิ่นแผนภาพระบบจะต้องมีค่ามากกว่าหรือเท่ากับค่าแต่ละลำดับองค์ประกอบของโหนดอ้างอิง รูปที่ 3.45



รูปที่ 3.45 การเปรียบเทียบค่าแต่ละลำดับองค์ประกอบของคุณลักษณะท้องถิ่น

ผลจากการเปรียบเทียบกับโหนดอ้างอิง Adapter ดังรูปที่ 3.46

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ			
ชื่อคลาส	ชนิดของคลาส	เมทีอด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)

รูปที่ 3.46 ผลจากการเปรียบเทียบกับโหนดอ้างอิงแบบรูป Adapter

4) เปรียบเทียบคุณลักษณะของโหนดอ้างอิงกับโหนดในตารางคุณลักษณะท้องถิ่นแผนภาพระบบ ดังรูปที่ 3.47

คุณลักษณะท้องถิ่นของโหนดอ้างอิง (R)

Client	Concrete	-	(0,1,0,0,0,0,0,0,0,0,0)

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ			
ชื่อคลาส	ชนิดของคลาส	เมทีอด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)

โหนดที่ตรงกับโหนออ้างอิง Adapter

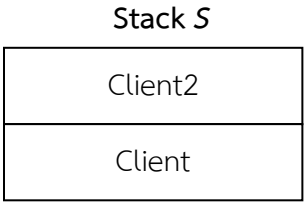
รูปที่ 3.47 เปรียบเทียบคุณลักษณะของโหนดอ้างอิงกับโหนดในตารางคุณลักษณะท้องถิ่น

5) ทำการใส่ชื่อโหนดอ้างอิงที่ตรงกับโหนดของตารางคุณลักษณะท้องถิ่นแผนภาพระบบลงในคอลัมน์ที่ 5 ดังรูป 3.48

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ				
ชื่อคลาส	ชนิดของคลาส	เมทือด	เส้นความสัมพันธ์ระหว่างคลาส	ชื่อคลาสของแบบรูป
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0)	-
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,2,0)	-
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)	-
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)	-
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0)	-
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)	-
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0)	-
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0)	-
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0)	-

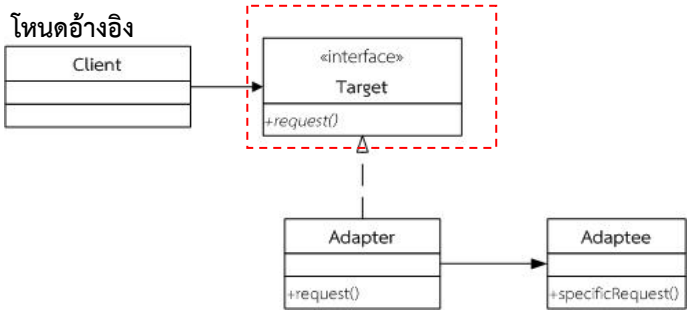
รูปที่ 3.48 การใส่ชื่อโหนดอ้างอิง Client ในตาราง LPT-S

6) ทำการบันทึกคุณลักษณะท้องถิ่นของโหนดที่ตรงกับโหนดอ้างอิงลงในกองซ้อน (Stack) ดังรูปที่ 3.49



รูปที่ 3.49 Stack ของโหนดอ้างอิงแบบรูป Adapter

7) ในกรณีที่หาโหนดอ้างอิงของแบบรูป Adapter จะมีการตรวจสอบโหนดที่เกี่ยวข้องด้วย ดังรูปที่ 3.50



รูปที่ 3.50 การตรวจสอบโหนดที่เกี่ยวข้องของแบบรูปการออกแบบ Adapter

8) เปรียบเทียบคุณลักษณะท้องถิ่นของคลาสปลายทางตาราง RT-P กับคุณลักษณะท้องถิ่นของคลาสปลายทางในตาราง RT-S ดังรูปที่ 3.51

ตาราง RT-S

ความสัมพันธ์ของแผนภาพคลาสของระบบ	
คลาสต้นทาง	คลาสปลายทาง
Client	Component
Client	Currency
Client2	Component
Client2	Facade

ความสัมพันธ์ของแบบรูปการออกแบบ Adapter	
คลาสต้นทาง	คลาสปลายทาง
Client	Target
Adapter	Target
Adapter	Adaptee

ตาราง LPT-P

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Adapter			
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Target	Interface	Request()	(1,0,0,0,0,0,0,0,0,0,1,0)

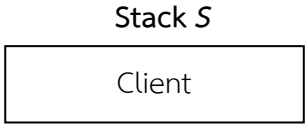
ตาราง LPT-S

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ			
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0,0)
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,0,2,0)
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0,0)

มีคุณลักษณะท้องถิ่นตรงกัน

รูปที่ 3.51 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Target

9) **ผลลัพธ์:** ดังนั้นแล้ว จากการหาโหนดอ้างอิงแบบรูปการออกแบบ Adapter จากตาราง LPT-S จะได้ คลาส Client เป็นโหนดอ้างอิงของแบบรูปการออกแบบ Adapter ดังรูปที่ 3.52 จากนั้นจะทำการตรวจจับในรายละเอียดของแบบรูปการออกแบบต่อไป



รูปที่ 3.52 ผลลัพธ์การตรวจจับโหนดอ้างออิงแบบรูป Adapter

ขั้นตอนที่ 2: อัลกอริทึมตรวจจับรายละเอียดแบบรูปการออกแบบ Adapter

```
Phase 2: Searching the Chaining of Expected Adapter Pattern
Input:  1) Stack S
        2) Pattern Relational Table (RT-P)
        3) System Relational Table (RT-S)
        4) System local properties table (LPT-S)
        5) Pattern local properties table (LPT-P)
Output : 1) List of nodes
// Pop peek node in Stack to check target node in RT-S with next node RT-P and compare
properties of node
addColumn (LPT-S,column6)
do while not empty (Stack S)
        Temp = Pop (Stack S)
        insertField(6, locate(Temp, LPT-S), '1')
        target1(target T1, Temp name )
        target2(target T2, Temp match)
        For each target node T1 of RT-S
                For each target node T2 of RT-P
                        For ( j=1 to 12 )
                                if component( j, rowi) >= component( j, rowi)
                                then
                                        flag = true
                                Else
                                        flag = false
                                        break forloop
                                End if
                        End for
                End for targetT2
```

รูปที่ 3.53 (ก) อัลกอริทึมการตรวจจับรายละเอียดแบบรูปการออกแบบ Adapter

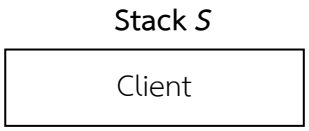
```
//Adding node name of patterns form LPT-P  into column 5 in LPT-S at the rows are same
properties after that push the row of node in Stack
                If (flag)
                        If propertyEquals(T2, T1)
                        Then insertField(5, row_i, getField(target, T2))
                                push(T1, StackS)
                        Else
                                flag F = true;
        End for target T1
// Check source node in RT-S with next node RT-P and compare properties of node

        source1(source S1, Temp name)
        source2(source S2, Temp match)
        For each source node S1 of RT-S
                For each source node S2 of RT-P
                        For ( j=1 to 12 )
                                if component( j, row_i) >= component( j, row_i)
                                then
                                        flag = true
                                Else
                                         flag = false
                                        break forloop
                                End if
                        End for
                End for source T2
//Adding node name of patterns form LPT-P into column 5 in LPT-S at the rows are same
properties after that push the row of node in Stack

                If (flag)
                        If propertyEquals(T2, S1)
                        Then insertField(5, row_i, getField(source, S2))
                                push(S1, Stack S)
                        Else
                                flagF = true;
        End for source node S1
//Result of inexact matching node
        If flagF
                Not match
        End If
End do
```

รูปที่ 3.53 (ข) อัลกอริทึมการตรวจจับรายละเอียดแบบรูปการออกแบบ Adapter

1) ข้อมูลนำเข้า 1. Stack *S* ดังรูปที่ 3.54



รูปที่ 3.54 ข้อมูลนำเข้า Stack *S*

2. ตารางคุณลักษณะท้องถิ่นแผนภาพระบบ (LPT-S) ดังรูปที่ 3.55

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ			
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0)
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,2,0)
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0)
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0)
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0)
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0)

รูปที่ 3.55 คุณลักษณะท้องถิ่นแผนภาพระบบ (LPT-S)

3. ตารางความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S) ดังรูปที่ 3.56

ความสัมพันธ์ของแผนภาพคลาสของระบบ	
คลาสต้นทาง	คลาสปลายทาง
Client	Component
Client	Currency
Product	Currency
Exchange	Currency
Exchange	CurentExchane
Client2	Component
Client2	Facade
Facade	Mobile
Facade	Internet
Facade	Ticket

รูปที่ 3.56 ความสัมพันธ์ของแผนภาพคลาสของระบบ (RT-S)

4. ตารางคุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P) ดังรูปที่ 3.57

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Adapter			
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส
Client	Concrete	-	(0,1,0,0,0,0,0,0,0,0,0)
Target	Interface	Request()	(1,0,0,0,0,0,0,0,0,1,0)
Adapter	Concrete	Request()	(0,1,0,0,0,0,0,0,0,0,1)
Adaptee	Concrete	SpecificRequest()	(1,0,0,0,0,0,0,0,0,0,0)

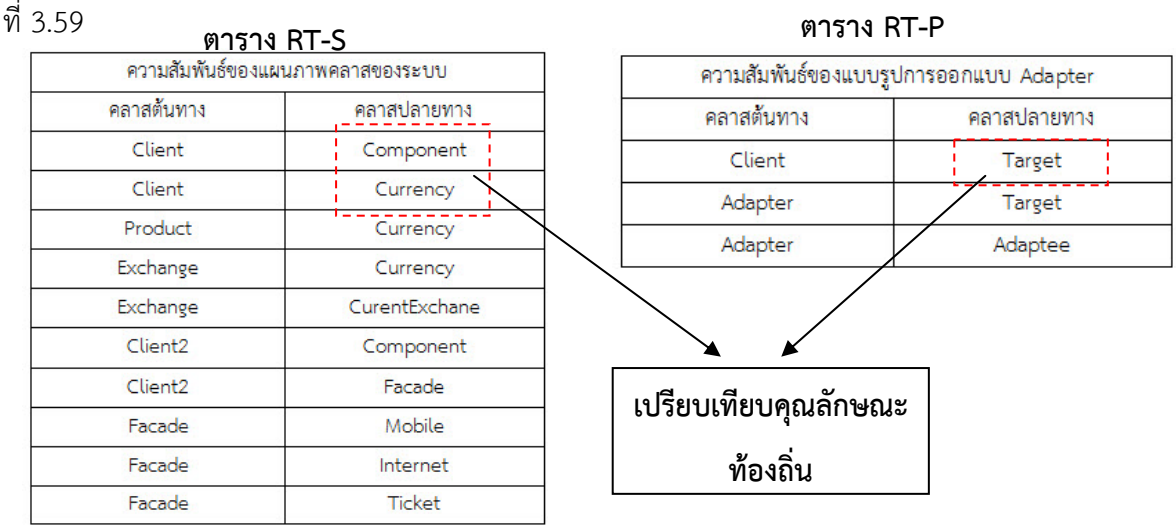
รูปที่ 3.57 คุณลักษณะท้องถิ่นของรูปการออกแบบ Adapter (LPT-P)

5. ตารางความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P) ดังรูปที่ 3.58

ความสัมพันธ์ของแบบรูปการออกแบบ Adapter	
คลาสต้นทาง	คลาสปลายทาง
Client	Target
Adapter	Target
Adapter	Adaptee

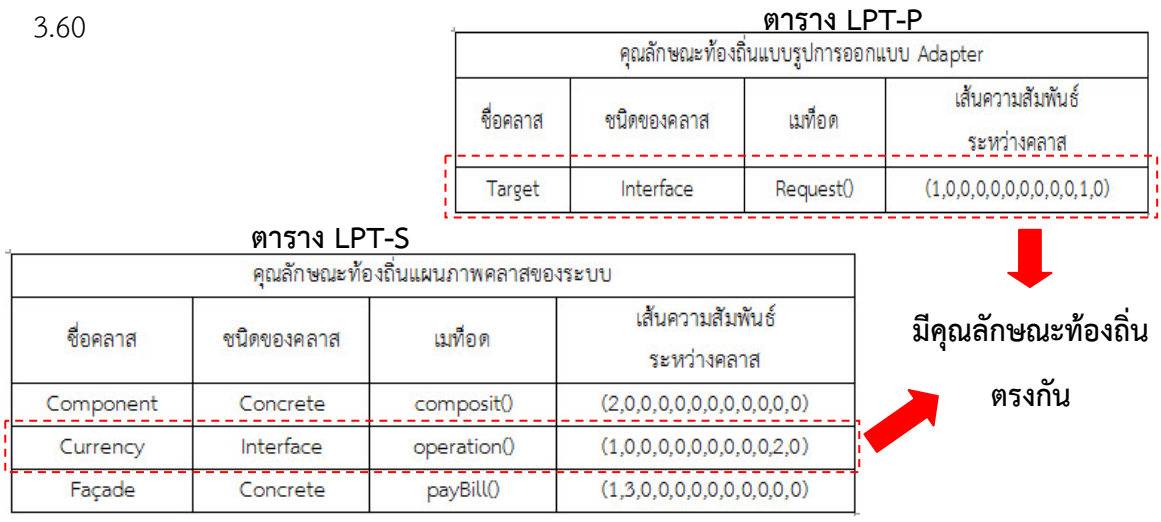
รูปที่ 3.58 ความสัมพันธ์ของแบบรูปการออกแบบ Adapter (RT-P)

2) ค้นหาโหนดปลายทางจาก Stack *S* ซึ่งมีค่า คือ โหนด Client โดยทำการหาตำแหน่งของโหนด Client ในตาราง RT-S และค้นหาโหนดที่สัมพันธ์กับโหนดอ้างอิงลำดับถัดไปในตาราง RT-P จากนั้นทำการเปรียบเทียบคุณลักษณะท้องถิ่นของแต่ละโหนดในตาราง RT-S กับโหนดในตาราง RT-P ดังรูปที่ 3.59



รูปที่ 3.59 การค้นหาโหนดปลายทางจาก Stack *S* ซึ่งมีค่าเป็นโหนด Client

3) เปรียบเทียบคุณลักษณะท้องถิ่นของแต่ละโหนดในตาราง RT-S กับโหนดในตาราง RT-P ดังรูปที่ 3.60



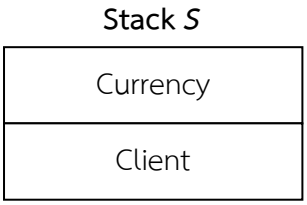
รูปที่ 3.60 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Currency กับ โหนด Target

4) เพิ่มค่า 1 ลงในคอลัมน์ตรวจสอบของโหนดที่ตรงกับชื่อคลาสของแบบรูป Client เพื่อแสดงว่าโหนดนี้ถูกตรวจสอบแล้ว และใส่ชื่อโหนดของแบบรูปการออกแบบ คือ Target ในตำแหน่งที่ทำการเปรียบเทียบลงในตารางคุณลักษณะท้องถิ่นแผนภาพระบบลงในคอลัมน์ที่ 5 ดังรูปที่ 3.61

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ					
ชื่อคลาส	ชนิดของคลาส	เมทือด	เส้นความสัมพันธ์ระหว่างคลาส	ชื่อคลาสของแบบรูป	ตรวจสอบ
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0)	-	0
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,2,0)	Target	0
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)	-	0
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)	-	0
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)	-	0
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0)	-	0

รูปที่ 3.61 การบันทึกชื่อโหนด Target ลงในตาราง LPT-S

5) ทำการบันทึกคุณลักษณะท้องถิ่นของโหนดที่ตรงกับโหนดของแบบรูปการออกแบบลงใน Stack *S* ดังรูปที่ 3.62



รูปที่ 3.62 การจัดเก็บโหนด Currency ลงใน Stack *S*

6) ค้นหาโหนดต้นทางจาก Stack *S* ซึ่งมีค่า คือ โหนด Curency โดยทำการหาตำแหน่งของโหนด Curency ในตาราง RT-S และค้นหาโหนดที่สัมพันธ์กับโหนดลำดับถัดไปในตาราง RT-P จากนั้นทำการเปรียบเทียบคุณลักษณะท้องถิ่นของแต่ละโหนดในตาราง RT-S กับโหนดในตาราง RT-P ดังรูปที่ 3.63

ตาราง RT-S

ความสัมพันธ์ของแผนภาพคลาสของระบบ	
คลาสต้นทาง	คลาสปลายทาง
Client	Component
Client	Currency
Product	Currency
Exchange	Currency
Exchange	CurentExchane
Client2	Component
Client2	Facade
Facade	Mobile
Facade	Internet
Facade	Ticket

ตาราง RT-P

ความสัมพันธ์ของแบบรูปการออกแบบ Adapter	
คลาสต้นทาง	คลาสปลายทาง
Client	Target
Adapter	Target
Adapter	Adaptee

เปรียบเทียบคุณลักษณะท้องถิ่น

รูปที่ 3.63 การค้นหาโหนดต้นทางจาก Stack *S* ซึ่งมีค่าเป็นโหนด Curency

7) เปรียบเทียบคุณลักษณะท้องถิ่นของแต่ละโหนดในตาราง RT-S กับโหนดในตาราง RT-P ดังรูปที่ 3.64

ตาราง LPT-P

คุณลักษณะท้องถิ่นแบบรูปการออกแบบ Adapter			
ชื่อคลาส	ชนิดของคลาส	เมทอด	เส้นความสัมพันธ์ระหว่างคลาส
Adapter	Concrete	Request()	(0,1,0,0,0,0,0,0,0,0,1)

ตาราง LPT-S

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ			
ชื่อคลาส	ชนิดของคลาส	เมทอด	เส้นความสัมพันธ์ระหว่างคลาส
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)

มีคุณลักษณะท้องถิ่นตรงกัน

รูปที่ 3.64 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Product, Exchange กับ โหนด Adapter

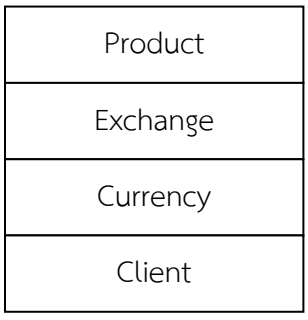
8) เพิ่มค่า 1 ลงในคอลัมน์ตรวจสอบของโหนดที่ตรงกับชื่อคลาสของแบบรูป Target เพื่อแสดงว่าโหนดนี้ถูกตรวจสอบแล้ว และใส่ชื่อโหนดของแบบรูปการออกแบบ คือ Adapter ในตำแหน่งที่ทำการเปรียบเทียบลงในตารางคุณลักษณะท้องถิ่นแผนภาพระบบลงในคอลัมน์ที่ 5 ดังรูปที่ 3.65

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ					
ชื่อคลาส	ชนิดของคลาส	เมทือด	เส้นความสัมพันธ์ระหว่างคลาส	ชื่อคลาสของแบบรูป	ตรวจสอบ
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0)	-	0
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,2,0)	Target	1
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)	Adapter	0
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)	Adapter	0
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)	-	0
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0)	-	0

รูปที่ 3.65 การบันทึกชื่อโหนด Adapter ลงในตาราง LPT-S

9) ทำการบันทึกคุณลักษณะท้องถิ่นของโหนดที่ตรงกับโหนดของแบบรูปการออกแบบลงใน Stack *S* ดังรูปที่ 3.66

Stack *S*



รูปที่ 3.66 การจัดเก็บโหนด Product และโหนด Exchange ลงใน Stack *S*

10) ค้นหาโหนดปลายทางจาก Stack *S* ซึ่งมีค่า คือ โหนด Product โดยทำการหาตำแหน่งของโหนด Product ในตาราง RT-S และค้นหาโหนดที่สัมพันธ์กับโหนดลำดับถัดไปในตาราง RT-P จากนั้นทำการเปรียบเทียบคุณลักษณะท้องถิ่นของแต่ละโหนดในตาราง RT-S กับโหนดในตาราง RT-P ดังรูปที่ 3.67

ตาราง RT-S

ความสัมพันธ์ของแผนภาพคลาสของระบบ	
คลาสต้นทาง	คลาสปลายทาง
Client	Component
Client	Currency
Product	Currency
Exchange	Currency
Exchange	CurentExchane
Client2	Component
Client2	Facade
Facade	Mobile
Facade	Internet
Facade	Ticket

ตาราง RT-P

ความสัมพันธ์ของแบบรูปการออกแบบ Adapter	
คลาสต้นทาง	คลาสปลายทาง
Client	Target
Adapter	Target
Adapter	Adaptee

โหนด Product ไม่มีโหนดปลายทางมาเชื่อมต่อจึงสิ้นสุดการตรวจจับ

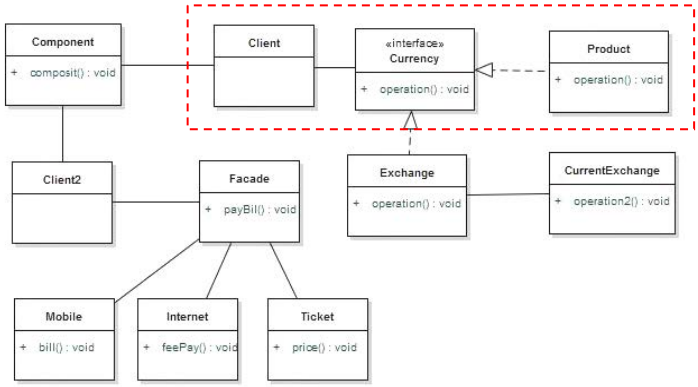
รูปที่ 3.67 การค้นหาโหนดปลายทางจาก Stack *S* ซึ่งมีค่าเป็นโหนด Product

11) เพิ่มค่า 1 ลงในคอลัมน์ตรวจสอบของโหนดที่ตรงกับชื่อคลาสของแบบรูป Adapter ของโหนด Product เพื่อแสดงว่าโหนดนี้ถูกตรวจสอบแล้ว ดังรูปที่ 3.68

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ					
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส	ชื่อคลาสของแบบรูป	ตรวจสอบ
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0)	-	0
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,2,0)	Target	1
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)	Adapter	1
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)	Adapter	0
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)	-	0
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0)	-	0

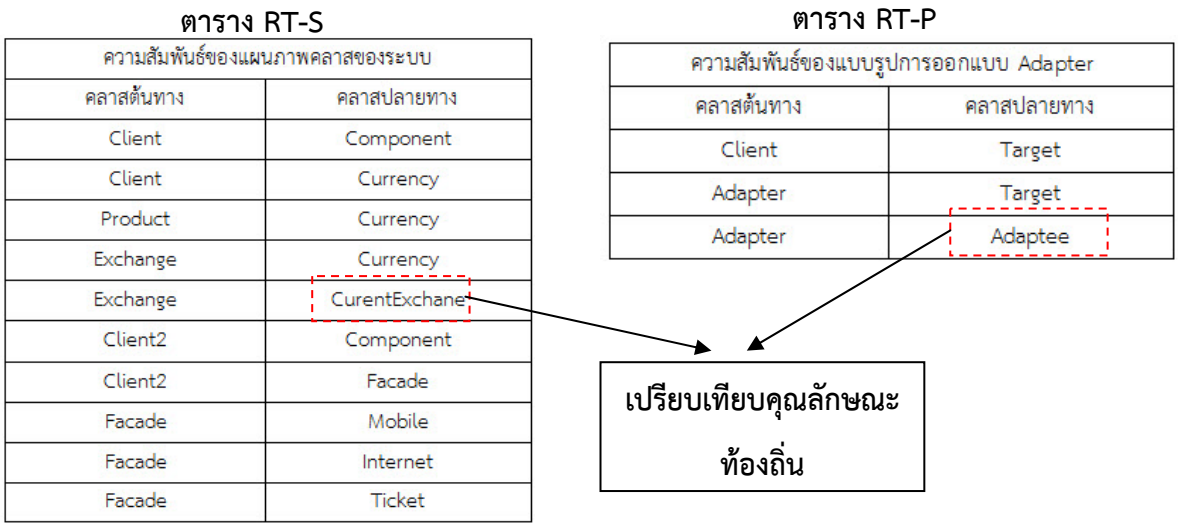
รูปที่ 3.68 การบันทึกชื่อโหนด Adapter ลงในตาราง LPT-S

12) เมื่อโหนด Product ไม่สามารถตรวจจับโหนดที่มีความสัมพันธ์ในลำดับถัดไปไปได้ จะทำการสรุปผลลัพธ์ที่ได้จะพิจารณาจากโหนดที่ถูกตรวจสอบแล้วมีค่าเป็น 1 ในคอลัมน์ตรวจสอบ จากแผนภาพคลาสระบบ จะพบแบบรูปที่มีความเหมือนกับแบบรูปการออกแบบ Adapter ได้ จำนวน 1 แบบรูป ซึ่งประกอบไปด้วยคลาส Client – Currency – Product ที่ 3.69



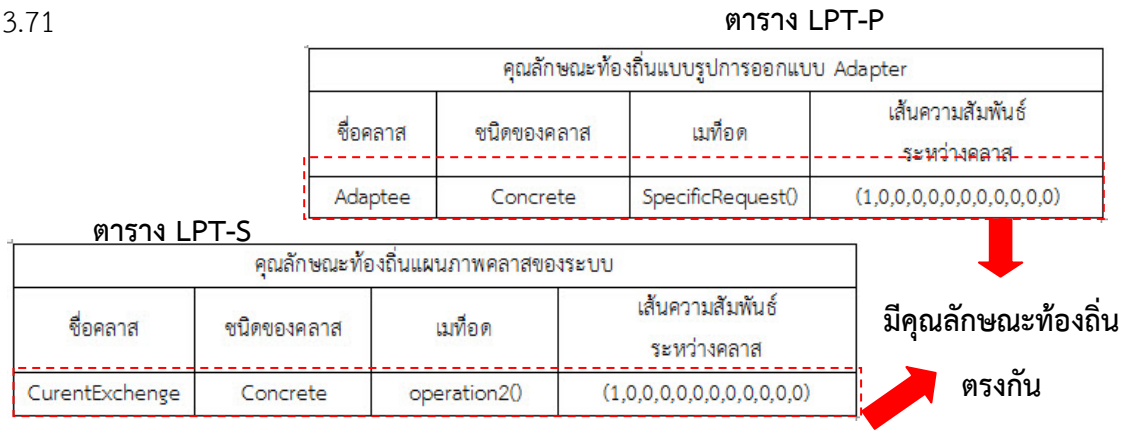
รูปที่ 3.69 ผลลัพธ์การตรวจจับที่มีความเหมือนกับแบบรูปการออกแบบ Adapter

13) ใน Stack *S* ยังไม่เป็นค่าว่าง จึงทำการค้นหาโหนดปลายทางจาก Stack *S* ซึ่งมีค่า คือ โหนด Exchange โดยทำการหาตำแหน่งของโหนด Exchange ในตาราง RT-S และค้นหาโหนดที่สัมพันธ์กับโหนดลำดับถัดไปในตาราง RT-P จากนั้นทำการเปรียบเทียบคุณลักษณะท้องถิ่นของแต่ละโหนดในตาราง RT-S กับโหนดในตาราง RT-P ดังรูปที่ 3.70



รูปที่ 3.70 การค้นหาโหนดปลายทางจาก Stack *S* ซึ่งมีค่าเป็นโหนด Exchange

14) เปรียบเทียบคุณลักษณะท้องถิ่นของแต่ละโหนดในตาราง RT-S กับโหนดในตาราง RT-P ดังรูปที่ 3.71



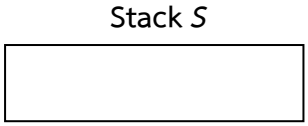
รูปที่ 3.71 เปรียบเทียบคุณลักษณะท้องถิ่นโหนด Current กับ โหนด Adaptee

15) เพิ่มค่า 1 ลงในคอลัมน์ตรวจสอบของโหนดที่ตรงกับชื่อคลาสของแบบรูป Adapter ของโหนด Exchange เพื่อแสดงว่าโหนดนี้ถูกตรวจสอบแล้ว และใส่ชื่อโหนดของแบบรูปการออกแบบ คือ Adaptee ในตำแหน่งที่ทำการเปรียบเทียบลงในตารางคุณลักษณะท้องถิ่นแผนภาพระบบลงในคอลัมน์ที่ 5 ดังรูปที่ 3.72

คุณลักษณะท้องถิ่นแผนภาพคลาสของระบบ					
ชื่อคลาส	ชนิดของคลาส	เมท็อด	เส้นความสัมพันธ์ระหว่างคลาส	ชื่อคลาสของแบบรูป	ตรวจสอบ
Client	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Component	Concrete	composit()	(2,0,0,0,0,0,0,0,0,0,0)	-	0
Currency	Interface	operation()	(1,0,0,0,0,0,0,0,0,2,0)	Target	1
Product	Concrete	operation()	(0,0,0,0,0,0,0,0,0,0,1)	Adapter	1
Exchange	Concrete	operation()	(0,1,0,0,0,0,0,0,0,0,1)	Adapter	1
CurentExchenge	Concrete	operation2()	(1,0,0,0,0,0,0,0,0,0,0)	Adaptee	0
Client2	Concrete	-	(0,2,0,0,0,0,0,0,0,0,0)	Client	1
Façade	Concrete	payBill()	(1,3,0,0,0,0,0,0,0,0,0)	-	0
Mobile	Concrete	Bill()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Internet	Concrete	feePay()	(1,0,0,0,0,0,0,0,0,0,0)	-	0
Ticket	Concrete	Price()	(1,0,0,0,0,0,0,0,0,0,0)	-	0

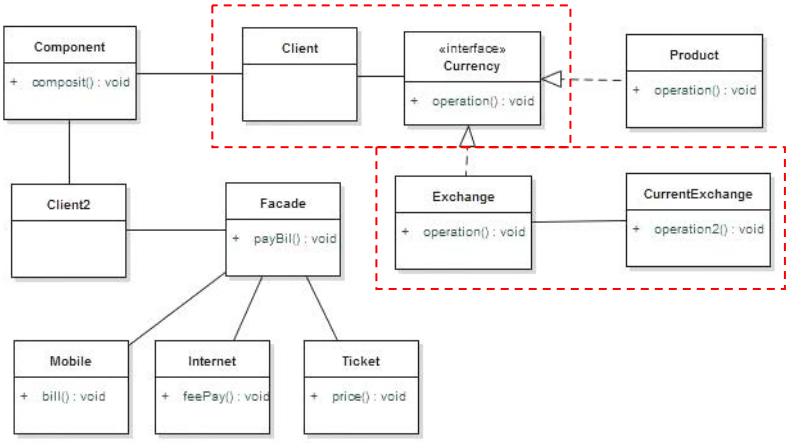
รูปที่ 3.72 การบันทึกชื่อโหนด Adapter ลงในตาราง LPT-S

16) จบการตรวจจับแบบรูปการออกแบบ Adapter เมื่อ Stack *S* มีค่าว่าง ดังรูปที่ 3.73



รูปที่ 3.73 Stack *S* มีค่าว่าง

17) **ผลลัพธ์:** การตรวจจับจากแผนภาพคลาสระบบ จะพบแบบรูปการออกแบบ Adapter ได้จำนวน 1 แบบรูป ผลลัพธ์ที่ได้จะพิจารณาจากโหนดที่ถูกตรวจสอบแล้วมีค่าเป็น 1 ในคอลัมน์ตรวจสอบ ซึ่งประกอบไปด้วยคลาส Client – Currency – Exchange – CurrentExchange ดังรูปที่ 3.74



รูปที่ 3.74 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ Adapter

บทที่ 4

การวิเคราะห์ ออกแบบ และพัฒนาเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง

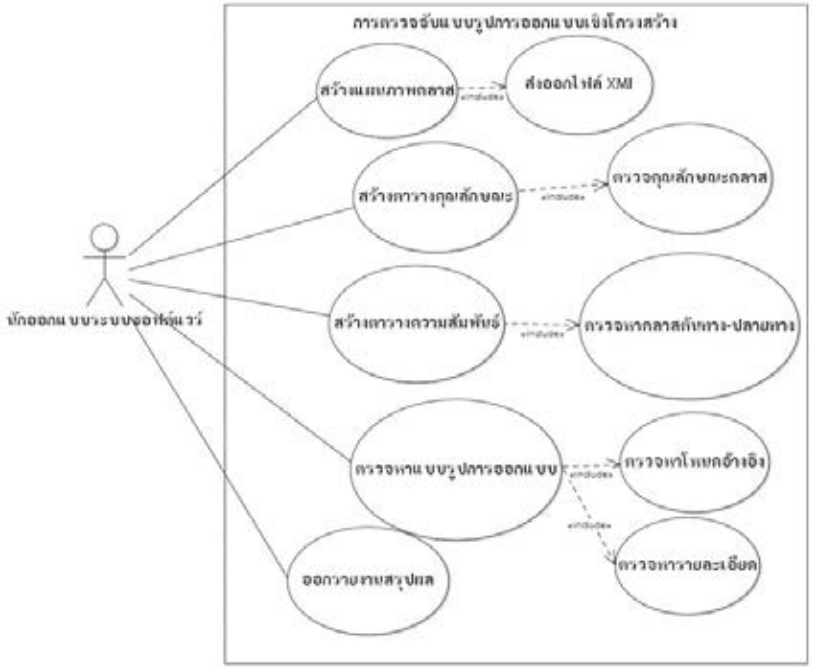
ในบทนี้จะอธิบายถึงการวิเคราะห์ ออกแบบ และพัฒนาเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้างซึ่งมีรายละเอียดแบ่งเป็น 3 ส่วน ดังนี้

4.1. การวิเคราะห์การออกแบบการตรวจจับแบบรูปการออกแบบ

ในการออกแบบเครื่องมือตรวจจับแบบรูปการออกแบบจะทำการวิเคราะห์การทำงานของเครื่องมือโดยใช้แผนภาพยูสเคส (Usecase Diagram) และแผนภาพคลาส (Class Diagram) อธิบายรายละเอียดได้ ดังนี้

4.1.1. แผนภาพยูสเคส

แผนภาพยูสเคส จะอธิบายภาพรวมการใช้งานเครื่องมือการตรวจจับแบบรูปการออกแบบที่มีระหว่างนักออกแบบระบบซอฟต์แวร์กับเครื่องมือ ซึ่งจะมีการทำงานประกอบไปด้วย 10 ยูสเคส แสดงดังรูปที่ 4.1



รูปที่ 4.1 แผนภาพยูสเคสสำหรับการตรวจจับแบบรูปการออกแบบ

จากแผนภาพยูสเคส รูปที่ 4.3 สามารถอธิบายรายละเอียดของแต่ละยูสเคสได้ ดังตารางที่ 4.1-4.10

ตารางที่ 4.1 คำอธิบายยูสเคสการสร้างแผนภาพคลาส

ชื่อยูสเคส :	การสร้างแผนภาพคลาส
ผู้กระทำ :	นักออกแบบระบบซอฟต์แวร์
รายละเอียด :	สร้างแผนภาพคลาสด้วยเครื่องมือ Enterprise Architect รุ่นที่ 7.0 หรือใหม่กว่า
เงื่อนไขก่อนหน้า :	ไม่ระบุ
ขั้นตอน :	ออกแบบแผนภาพคลาสตามมาตรฐานยูเอ็มแอล โดยมีการระบุชื่อคลาส ชนิดของคลาส เมธ็อด และเส้นความสัมพันธ์ระหว่างคลาส เป็นอย่างน้อย
เงื่อนไขภายหลัง	ไม่ระบุ

ตารางที่ 4.2 คำอธิบายยูสเคสการส่งออกไฟล์เอกซ์เอ็มไอ

ชื่อยูสเคส :	ส่งออกไฟล์เอกซ์เอ็มไอ
ผู้กระทำ :	นักออกแบบระบบซอฟต์แวร์
รายละเอียด :	นำออกแผนภาพคลาสในรูปแบบของไฟล์เอกซ์เอ็มไอ
เงื่อนไขก่อนหน้า :	สร้างแผนภาพคลาส
ขั้นตอน :	1. สร้างแผนภาพคลาสตามมาตรฐานยูเอ็มแอล 2. ส่งออกไฟล์แผนภาพคลาสในรูปแบบของไฟล์เอกซ์เอ็มไอ รุ่นที่ 2.1
เงื่อนไขภายหลัง	ไม่ระบุ

ตารางที่ 4.3 คำอธิบายยูสเคสการสร้างตารางคุณลักษณะ

ชื่อยูสเคส :	สร้างตารางคุณลักษณะ
ผู้กระทำ :	นักออกแบบระบบซอฟต์แวร์
รายละเอียด :	การจัดเก็บข้อมูลชื่อคลาส ชนิดของคลาส ชื่อเมธ็อด และจำนวนเส้นความสัมพันธ์ระหว่างคลาส เพื่อใช้เป็นข้อมูลในการตรวจจับแบบรูปการออกแบบ
เงื่อนไขก่อนหน้า :	ไม่ระบุ
ขั้นตอน :	นำเข้าไฟล์เอกซ์เอ็มไอที่ได้จากการส่งออกไฟล์ในขั้นตอนการออกแบบแผนภาพคลาส
เงื่อนไขภายหลัง	ไม่ระบุ

ตารางที่ 4.4 คำอธิบายยูสเคสการตรวจคุณลักษณะคลาส

ชื่อยูสเคส :	ตรวจคุณลักษณะคลาส
ผู้กระทำ :	ไม่ระบุ
รายละเอียด :	อ่านข้อมูลแผนภาพคลาสที่อยู่ในรูปแบบของไฟล์เอกซ์เอ็มไอแล้วจัดเก็บข้อมูลชื่อคลาส ชนิดของคลาส ชื่อเมธ็อด และจำนวนเส้นความสัมพันธ์ระหว่างคลาส
เงื่อนไขก่อนหน้า :	การสร้างตารางคุณลักษณะ
ขั้นตอน :	กรองข้อมูลที่จำเป็นสำหรับจัดเก็บลงในตารางคุณลักษณะ
เงื่อนไขภายหลัง	ตารางคุณลักษณะของแผนภาพคลาส

ตารางที่ 4.5 คำอธิบายยูสเคสการสร้างตารางความสัมพันธ์

ชื่อยูสเคส :	สร้างตารางความสัมพันธ์
ผู้กระทำ :	นักออกแบบระบบซอฟต์แวร์
รายละเอียด :	การจัดเก็บข้อมูลคลาสต้นทางและคลาสปลายทาง ที่มีความสัมพันธ์กันในแผนภาพคลาส
เงื่อนไขก่อนหน้า :	ไม่ระบุ
ขั้นตอน :	นำเข้าไฟล์เอกซ์เอ็มไอที่ได้จากการส่งออกไฟล์ในขั้นตอนการออกแบบแผนภาพคลาส
เงื่อนไขภายหลัง	ไม่ระบุ

ตารางที่ 4.6 คำอธิบายยูสเคสการตรวจจับคลาสต้นทาง-ปลายทาง

ชื่อยูสเคส :	ตรวจจับคลาสต้นทาง-ปลายทาง
ผู้กระทำ :	ไม่ระบุ
รายละเอียด :	การจัดเก็บชื่อคลาสต้นทางและชื่อคลาสปลายทาง ที่มีความสัมพันธ์กันที่อยู่ในแผนภาพคลาส
เงื่อนไขก่อนหน้า :	การสร้างตารางความสัมพันธ์
ขั้นตอน :	กรองข้อมูลที่จำเป็นสำหรับจัดเก็บลงตารางความสัมพันธ์
เงื่อนไขภายหลัง	ตารางความสัมพันธ์

ตารางที่ 4.7 คำอธิบายยูสเคสการตรวจจับแบบรูปการออกแบบ

ชื่อยูสเคส :	ตรวจจับแบบรูปการออกแบบ
ผู้กระทำ :	นักออกแบบระบบซอฟต์แวร์
รายละเอียด :	ตรวจจับแบบรูปการออกแบบเชิงโครงสร้างที่ปรากฏอยู่ในแผนภาพคลาสของระบบ
เงื่อนไขก่อนหน้า :	ไม่ระบุ
ขั้นตอน :	1. ทำการตรวจจับโหนดอ้างอิงของแต่ละแบบรูปการออกแบบ 2. ตรวจจับในรายละเอียดของแบบรูปการออกแบบ
เงื่อนไขภายหลัง	ไม่ระบุ

ตารางที่ 4.8 คำอธิบายยูสเคสการตรวจจับโหนดอ้างอิง

ชื่อยูสเคส :	ตรวจจับโหนดอ้างอิง
ผู้กระทำ :	ไม่ระบุ
รายละเอียด :	ตรวจจับโหนดอ้างอิงตามที่ได้ทำการกำหนดไว้ตามแบบรูปการออกแบบเพื่อใช้เป็นจุดเริ่มต้นของการตรวจจับในรายละเอียดของแต่ละแบบรูป
เงื่อนไขก่อนหน้า :	การตรวจจับแบบรูปการออกแบบ
ขั้นตอน :	1. ทำการตรวจจับโหนดอ้างอิงทีละแบบรูปการออกแบบจนครบตามจำนวนของคลาสที่มี 2. บันทึกรายชื่อโหนดอ้างอิงที่ตรงตามแบบรูปการออกแบบ
เงื่อนไขภายหลัง	โหนดอ้างอิงของแต่ละแบบรูปการออกแบบ

ตารางที่ 4.9 คำอธิบายยูสเคสการตรวจจับรายละเอียด

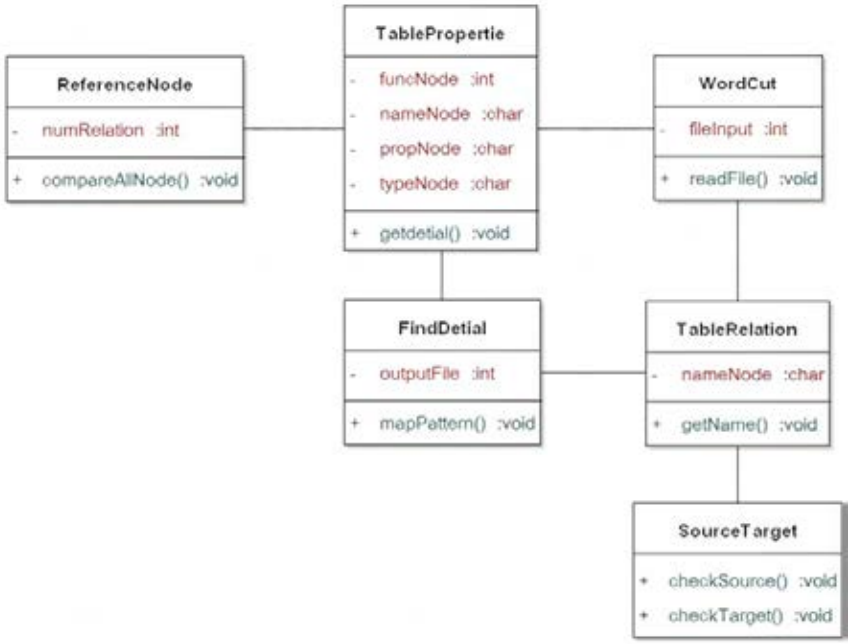
ชื่อยูสเคส :	ตรวจจับรายละเอียด
ผู้กระทำ :	ไม่ระบุ
รายละเอียด :	ตรวจจับรายละเอียดจากของแบบรูปการออกแบบจากโหนดอ้างอิงที่ตรวจพบ
เงื่อนไขก่อนหน้า :	การตรวจจับโหนดอ้างอิง
ขั้นตอน :	1. ตรวจจับรายละเอียดของแบบรูปจากโหนดอ้างอิง 2. ตรวจสอบความสัมพันธ์คลาสต้นทาง – คลาสปลายทาง 3. แสดงรายชื่อคลาสที่ตรงตามแบบรูปการออกแบบ
เงื่อนไขภายหลัง	รายชื่อคลาสที่ตรงตามแบบรูปการออกแบบ

ตารางที่ 4.10 คำอธิบายยูสเคสการออกรายงานสรุปผล

ชื่อยูสเคส :	ออกรายงานสรุปผล
ผู้กระทำ :	นักออกแบบระบบซอฟต์แวร์
รายละเอียด :	การสรุปผลจากการตรวจจับแบบรูปการออกแบบ โดยจะแสดงชื่อแบบรูปการออกแบบ ชนิดและจำนวนแบบรูปการออกแบบที่ตรวจพบ
เงื่อนไขก่อนหน้า :	ไม่ระบุ
ขั้นตอน :	แสดงข้อมูลการตรวจจับและส่งออกผลการตรวจจับแบบรูปการออกแบบ
เงื่อนไขภายหลัง	รายงานสรุปผลการตรวจจับแบบรูปการออกแบบ

4.1.2. แผนภาพคลาส

แผนภาพคลาสของเครื่องมือการตรวจจับแบบรูปการออกแบบ ซึ่งจะแสดงความสัมพันธ์ของคลาสและแสดงถึงโครงสร้างของข้อมูลที่ใช้ในเครื่องมือ ดังรูปที่ 4.2



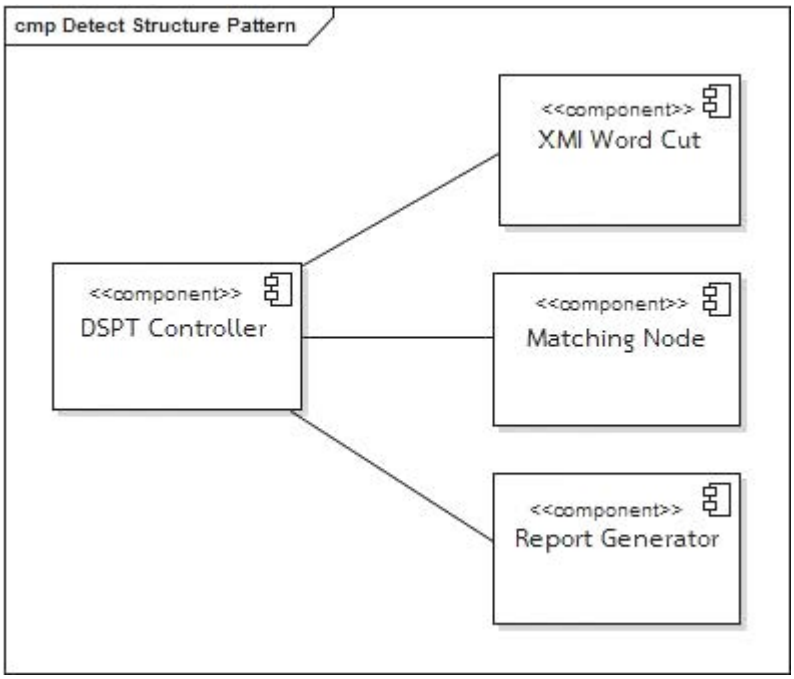
รูปที่ 4.2 แผนภาพคลาสสำหรับเครื่องมือตรวจจับแบบรูปการออกแบบ

จาก รูปที่ 4.2 สามารถอธิบายความสัมพันธ์ของแผนภาพคลาสและรายละเอียดของแต่ละคลาสได้ ดังนี้

1) คลาส TablePropertie สำหรับการบันทึกค่าคุณลักษณะของคลาส
2) คลาส TableRelation สำหรับบันทึกค่าความสัมพันธ์ของคลาสต้นทางและคลาสปลายทาง
3) คลาส WordCut สำหรับการกรองไฟล์เอกซ์เอ็มไอที่จำเป็นต่อการตรวจจับแบบรูปการออกแบบ เพื่อจัดเก็บข้อมูลลงในคลาส TablePropertie และคลาส TableRelation
4) คลาส Reference สำหรับการตรวจจับโหนดอ้างอิง
5) คลาส SourceTarget สำหรับการตรวจสอบคลาสต้นทางและคลาสปลายทางที่มีความสัมพันธ์กันกับแบบรูปการออกแบบ
6) คลาส FindDetail สำหรับตรวจจับรายละเอียดของแบบรูปการออกแบบพร้อมรายงานผล

4.2. แผนภาพองค์ประกอบ

แผนภาพองค์ประกอบของการออกแบบและพัฒนาเครื่องมือในการตรวจจับแบบรูปการออกแบบ ซึ่งมีรายละเอียดของการทำงานแบ่งเป็น 4 ส่วน ดังแสดงในรูปที่ 4.3



รูปที่ 4.3 แผนภาพองค์ประกอบของเครื่องมือ DSPT

จากแผนภาพองค์ประกอบของเครื่องมือสามารถอธิบายการทำงานได้ ดังนี้ หลังจากที่นักออกแบบระบบทำการสร้างแผนภาพคลาสด้วยเครื่องมือตามมาตรฐานของยูเอ็มแอล นักออกแบบจะต้องทำการส่งออกไฟล์ให้อยู่ในรูปแบบของไฟล์เอกซ์เอ็มไอและนำไฟล์ที่ส่งออกดังกล่าวเข้าสู่เครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง (Detect Structure Pattern Tool) ซึ่งเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้างจะแบ่งการทำงานออกเป็น 4 ส่วน ดังนี้

4.2.1. DSPT Controller (Detect Structure Pattern Tool)

เป็นส่วนที่ควบคุมการการทำงานหลักในการรับเข้าข้อมูล ส่งข้อมูลไปยังองค์ประกอบอื่นๆ ในการทำงานของเครื่องมือในส่วนอื่น

4.2.2. XMI Word Cut

เป็นส่วนที่ทำการอ่านไฟล์เอกซ์เอ็มไอซึ่งได้มาจากการออกแบบแผนภาพคลาสที่นำเข้า โดย XMI Word Cut จะทำการอ่านไฟล์เอกซ์เอ็มไอแล้วตัดส่วนที่ไม่จำเป็นต่อการทำงานของเครื่องมือออกไป ซึ่งข้อมูลที่จะจัดเก็บ ได้แก่ ชื่อคลาส ชนิดของคลาส ชื่อเมธ็อด เส้นความสัมพันธ์ระหว่างคลาส จำนวนเส้นความสัมพันธ์เข้าหาคลาส และจำนวนเส้นความสัมพันธ์ออกจากคลาส ของแต่ละคลาส จากนั้นจะทำการจัดเก็บค่าลงในตารางคุณลักษณะท้องถิ่นของแผนภาพคลาส และตารางความสัมพันธ์แผนภาพคลาสข้อมูลในตารางดังกล่าวจะถูกส่งต่อข้อมูลไปยังการจับคู่โหนด (Matching Node) เพื่อทำการตรวจจับแบบรูปการออกแบบเชิงโครงสร้างต่อไป

4.2.3. Matching Node

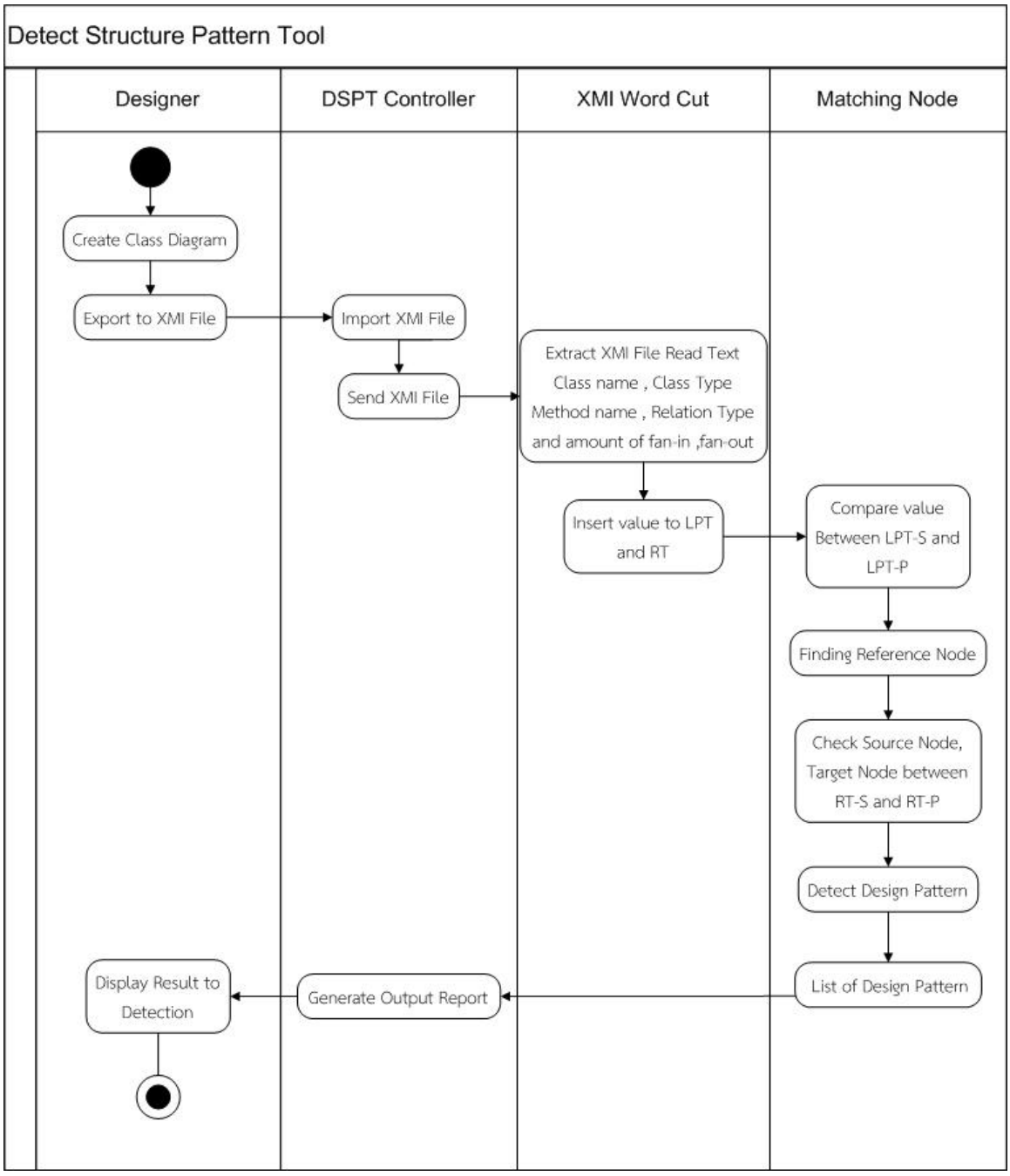
เป็นส่วนที่ใช้ค้นหาแบบรูปการออกแบบเชิงโครงสร้าง โดยจะทำการเปรียบเทียบค่าที่ได้จากตารางคุณลักษณะของแผนภาพคลาสและตารางความสัมพันธ์แผนภาพคลาสระหว่างแผนภาพคลาสของระบบกับแบบรูปการออกแบบเชิงโครงสร้างทั้ง 7 แบบรูป การในส่วนของการเปรียบเทียบนั้นจะทำการตรวจสอบชนิดของคลาส ชนิดของเส้นความสัมพันธ์ จากนั้นจะการตรวจสอบความสัมพันธ์ของคลาส ซึ่งในส่วนนี้จะมีการตรวจสอบคลาสต้นทางกับคลาสปลายทาง ที่ตรงกันของแผนภาพคลาสระบบและแผนภาพคลาสแบบรูปการออกแบบ จากนั้นจะทำการสรุปรายชื่อของคลาสทั้งหมดที่ตรงกับแบบรูปการออกแบบเชิงโครงสร้างที่ใช้ทำการตรวจจับ ในส่วนนี้จะส่งข้อมูลสู่การจัดทำรายงานต่อไป

4.2.4. Report Generator

เป็นส่วนที่ใช้ออกรายงานสรุปผลจากการตรวจจับแบบรูปการออกแบบ โดยรายงานสรุปผลจะแสดงถึงชนิดของแบบรูปการออกแบบเชิงโครงสร้างที่พบ และจำนวนของแบบรูปการออกแบบ พร้อมระบุชุดของคลาสที่ตรวจพบและแสดงผลค่าร้อยละความเหมือนของตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง

4.3. หลักการทำงานของเครื่องมือตรวจจับแบบรูปการออกแบบ

จากรายละเอียดการทำงานของเครื่องมือที่ได้อธิบายไว้ข้างต้น สามารถแสดงการทำงานของเครื่องมือในการตรวจจับแบบรูปการออกแบบ ได้ดังรูปที่ 4.4 โดยการทำงานของเครื่องมือเริ่มจากนักออกแบบทำการออกแบบแผนภาพคลาสด้วยเครื่องมือตามมาตรฐานยูเอ็มแอล จากนั้นทำการส่งออกแผนภาพคลาสให้อยู่ในรูปแบบของเอกซ์เอ็มไอไฟล์ ในส่วนของตัวควบคุมการทำงานจะทำการนำเข้าเอกซ์เอ็มไอไฟล์ เพื่อส่งต่อการทำงานให้กับส่วนกรองข้อมูลที่จะทำหน้าที่กรองส่วนสำคัญที่เกี่ยวข้องจากเอกซ์เอ็มไอไฟล์ทั้งหมดให้เหลือเพียงส่วนที่นำมาใช้งานต่อได้ ซึ่งได้แก่ ชื่อคลาส ชนิดของคลาส ชื่อเมธ็อด เส้นความสัมพันธ์ระหว่างคลาส จำนวนเส้นความสัมพันธ์เข้าหาคลาส และจำนวนเส้นความสัมพันธ์ออกจากคลาสของแต่ละคลาส โดยข้อมูลที่ได้จากการกรองแล้วนั้นจะจัดเก็บในส่วนของตารางคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ เพื่อใช้ในการตรวจจับแบบรูปการออกแบบซึ่งจะดำเนินการที่ละแบบรูปตามลำดับ ในการตรวจจับแบบรูปการออกแบบจะทำการพิจารณาจากโหนดอ้างอิงที่กำหนดขึ้นมาเพื่อใช้ตรวจจับลักษณะของของแบบรูปการออกแบบในเบื้องต้น โดยที่ไม่จำเป็นจะต้องพิจารณาจากคลาสทั้งหมด จากนั้นเมื่อได้โหนดอ้างอิงที่ตรงตามแบบรูปการออกแบบมาแล้วจะทำการตรวจจับในรายละเอียดของแบบรูปต่อไป ในการพิจารณารายละเอียดจะพิจารณาจากตารางความสัมพันธ์ที่แสดงถึงโหนดต้นทางและโหนดปลายทางที่มีความสัมพันธ์กันและต้องเป็นไปตามแบบรูปการออกแบบที่กำหนด ดำเนินการตรวจจับจนครบตามแบบรูปการออกแบบเชิงโครงสร้างทั้ง 7 แบบรูป และทำการออกรายงานสรุปผลการตรวจจับให้แก่นักออกแบบ ในส่วนของรายงานสรุปผลจะแสดงถึงรายละเอียดการตรวจจับถึงชนิดของแบบรูปการออกแบบเชิงโครงสร้างที่พบ และจำนวนของแบบรูปการออกแบบ พร้อมระบุชุดของคลาสที่ตรวจพบและแสดงผลค่าร้อยละความเหมือนของตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง ผลการตรวจจับที่แสดงความเหมือนแบบรูปการออกแบบเป็นค่าร้อยละนั้น นักออกแบบสามารถนำไปใช้วิเคราะห์การออกแบบในส่วนที่มีความเหมือนให้ตรงตามแบบรูปการออกแบบมาตรฐาน ซึ่งจากการสรุปผลดังกล่าวจะช่วยให้นักออกแบบระบบสามารถตรวจสอบความถูกต้องในการออกแบบได้ในเบื้องต้นก่อนการจะพัฒนาระบบจริง



รูปที่ 4.4 แผนภาพกิจกรรมการทำงานของเครื่องมือในการตรวจจับแบบรูปการออกแบบ

บทที่ 5

การทดสอบเครื่องมือการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง

สำหรับการทดสอบเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง เป็นการทดสอบการประมวลผลความถูกต้องของเครื่องมือที่ดำเนินการตามขั้นตอนวิธีที่ได้ทำการปรับปรุง โดยการทดสอบเครื่องมือจะประกอบไปด้วยกรณีทดสอบจำนวนสามกรณีซึ่งแบ่งเป็นกรณีทดสอบที่สร้างขึ้นเองและกรณีทดสอบจากการออกแบบระบบจริงซึ่งอธิบายรายละเอียดการทดสอบได้ ดังนี้

5.1. การทดสอบเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง

ในการทดสอบเครื่องมือดังกล่าวกำหนดกรณีทดสอบเป็นสามกรณี ซึ่งแยกตามลักษณะการทดสอบ ดังนี้

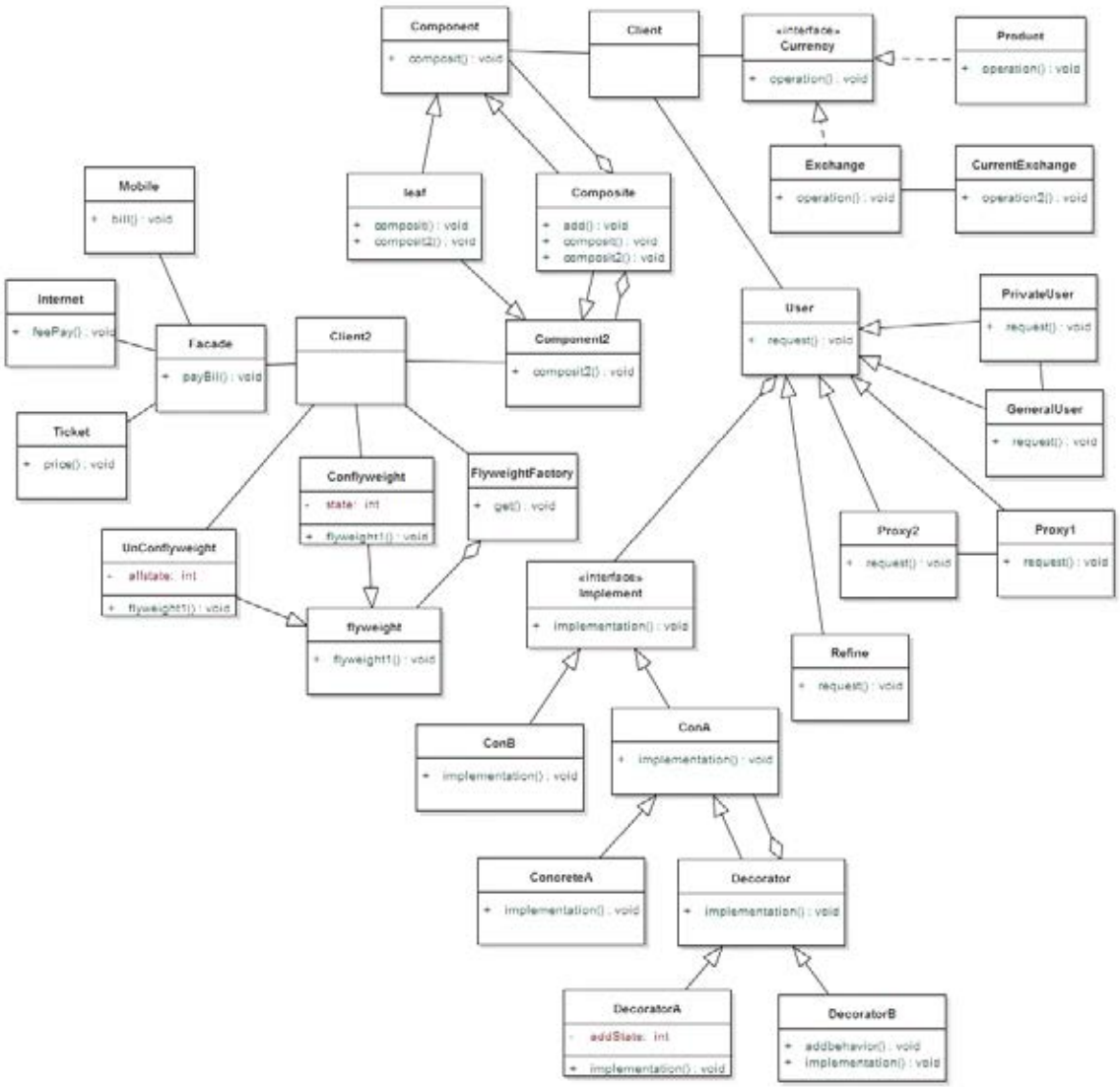
1) กรณีทดสอบที่หนึ่ง เป็นการทดสอบแผนภาพคลาสที่ประกอบไปด้วยแบบรูปการออกแบบเชิงโครงสร้างทั้งหมดเจ็ดรูปแบบ อันประกอบไปด้วย รูปการออกแบบ Adapter แบบรูปการออกแบบ Bridge แบบรูปการออกแบบ Composite แบบรูปการออกแบบ Decorator แบบรูปการออกแบบ Facadeแบบรูปการออกแบบ Flyweight และแบบรูปการออกแบบ Proxy ที่เชื่อมต่อกันทั้งแบบครบถ้วนและแบบที่ไม่ครบถ้วนตามแบบรูปการออกแบบ ซึ่งกรณีทดสอบนี้ทางผู้วิจัยได้สร้างขึ้นเอง

2) กรณีทดสอบที่สอง เป็นการทดสอบแผนภาพคลาสจากระบบจริง ซึ่งเป็นระบบการสร้างแบบจำลอง AtollGame เพื่อใช้เป็นแม่แบบในการกำหนดบทบาทให้กับผู้เล่นเกมตามสถานการณ์ต่างๆ

3) กรณีทดสอบที่สาม เป็นการทดสอบแผนภาพคลาสจากระบบจริงซึ่งเป็นระบบของ Web Portal Caching ที่ใช้แสดงถึงองค์ประกอบของกรอบงานของระบบ

5.1.1. การทดสอบกรณีทดสอบที่หนึ่ง

เป็นการทดสอบการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง จากแผนภาพคลาสที่ได้สร้างขึ้นเอง ประกอบไปด้วยแบบรูปการออกแบบเชิงโครงสร้างทั้งเจ็ดแบบรูป ซึ่งแผนภาพคลาสสำหรับการทดสอบแสดง ดังรูปที่ 5.1



รูปที่ 5.1 กรณีทดสอบที่หนึ่งสำหรับแผนภาพคลาสของแบบรูปการออกแบบทั้งหมด

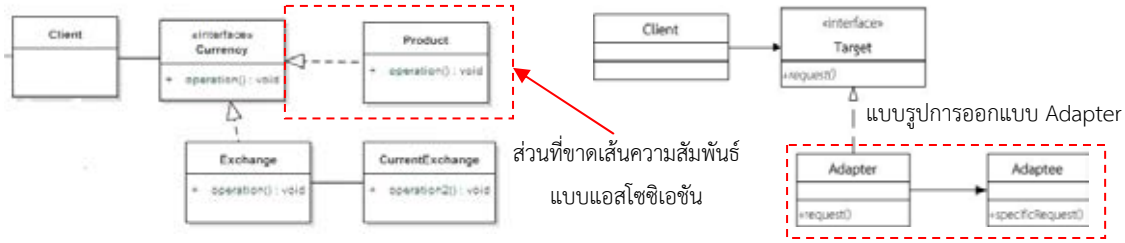
จากรูปที่ 5.1 สามารถตรวจจับแบบรูปการออกแบบด้วยเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้างได้ผลลัพธ์ ดังตารางที่ 5.1

ตารางที่ 5.1 ผลลัพธ์จากการทดสอบกรณีทดสอบที่หนึ่ง

แบบรูปการออกแบบ	จำนวนที่ตรวจพบ	ผลลัพธ์
Adapter Pattern	2	● CurrentExchange - Exchange - Currency - Client ● Product - Currency - Client - loss Association -75%.
Bridge Pattern	1	● Implement - User - GeneralUser - PrivateUser - Proxy1 - Proxy2 - Refine - ConA - ConB
Composite Pattern	2	● leaf - Composite - Client2 - Component2 ● Composite - leaf - Client - Component
Decorator Pattern	1	● Decorator - ConcreteA - DecoratorA - DecoratorB - ConA
Facade Pattern	1	● Facade - Internet - Mobile - Ticket - Client2
Flyweight Pattern	1	● flyweight - Conflyweight - FlyweightFactory - UnConflyweight - Client2
Proxy Pattern	3	● PrivateUser - GeneralUser - User ● Proxy2 - Proxy1 - User ● Refine - User - loss Association - 67%.

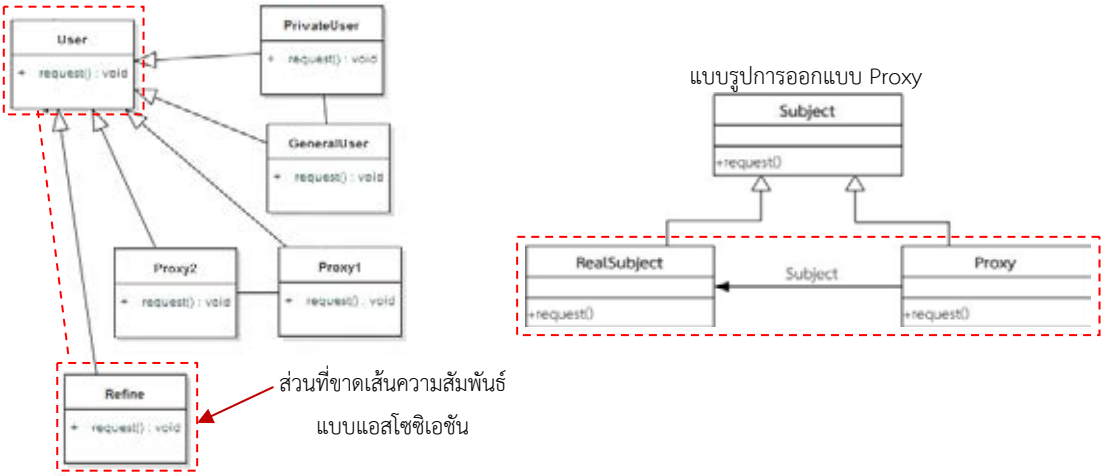
จากตารางที่ 5.1 ผลการตรวจจับแบบรูปการออกแบบจากแผนภาพคลาสที่สร้างขึ้นเองนั้น สรุปผล ดังนี้ สามารถตรวจจับแบบรูปการออกแบบเชิงโครงสร้างทั้งหมดที่มีอยู่ 7 แบบรูป นอกจากนั้นแล้วยังสามารถตรวจจับแบบรูปการออกแบบที่มีความเหมือนกับแบบรูปการออกแบบ คือ แบบรูปการออกแบบ Adapter ตรวจจับคลาสที่มีความเหมือน ได้แก่ Product - Currency - Client - loss Realization-75%. และแบบรูปการออกแบบ Proxy ตรวจจับคลาสที่มีความเหมือน ได้แก่ Refine - User - loss Association - 67%. ซึ่งค่าร้อยละความเหมือนกับแบบรูป จะพิจารณาจำนวนคลาสที่ตรวจพบกับจำนวนคลาสทั้งหมดของแบบรูปการออกแบบตามมาตรฐานและยังพิจารณาจากลำดับการตรวจจับคลาสที่มีความสัมพันธ์กัน ผลการตรวจจับแบบรูปการออกแบบ Adapte ของ

คลาสที่มีความเหมือน คือ Product - Currency - Client - loss Association-75%. นั้นขาดเส้นความสัมพันธ์แบบแอสโซซิเอชันที่เชื่อมไปยังคลาสที่เป็น Adaptee เมื่อนำไปเทียบกับแบบรูปการออกแบบ Adapter ตามมาตรฐานแล้วแสดงการแปรผลได้ ดังรูปที่ 5.2



รูปที่ 5.2 กรณีทดสอบที่หนึ่ง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Adapter

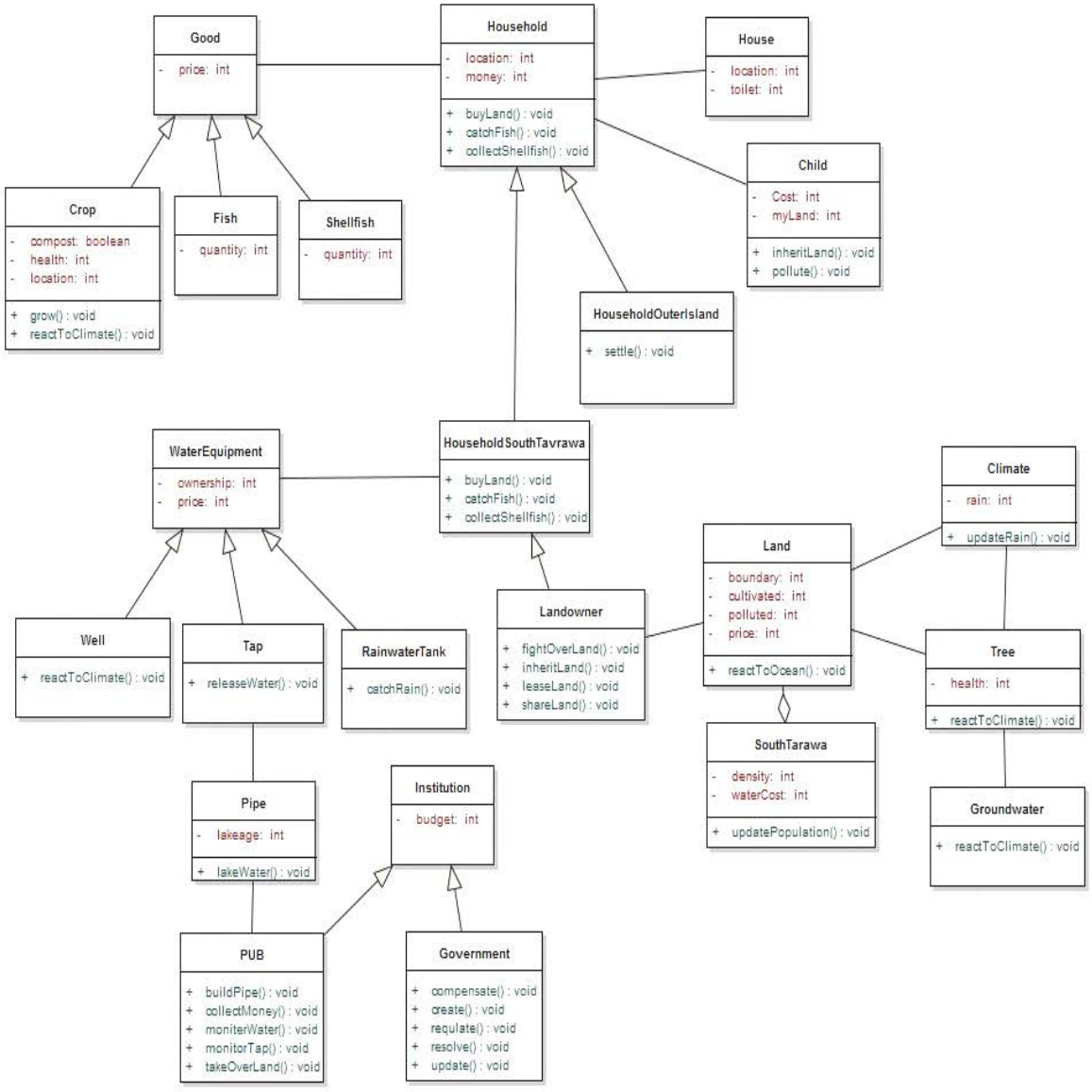
ผลการตรวจจับแบบรูปการออกแบบ Proxy ของคลาสที่มีความเหมือน คือ Refine - User - loss Association - 67%. นั้นขาดเส้นความสัมพันธ์แบบแอสโซซิเอชันที่เชื่อมไปยังคลาสที่เป็น RealSubject หรือ Proxy เมื่อนำไปเทียบกับแบบรูปการออกแบบ Proxy ตามมาตรฐานแล้วแสดงการแปรผลได้ ดังรูปที่ 5.3



รูปที่ 5.3 กรณีทดสอบที่หนึ่ง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Proxy

5.1.2. การทดสอบกรณีทดสอบที่สอง

เป็นการทดสอบการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง จากระบบจริงโดยใช้แผนภาพคลาสจากการสร้างแบบจำลอง AtollGame ซึ่งแผนภาพคลาสสำหรับการทดสอบแสดง ดังรูปที่ 5.4



รูปที่ 5.4 แสดงกรณีทดสอบที่สองแผนภาพคลาส AtollGame [12]

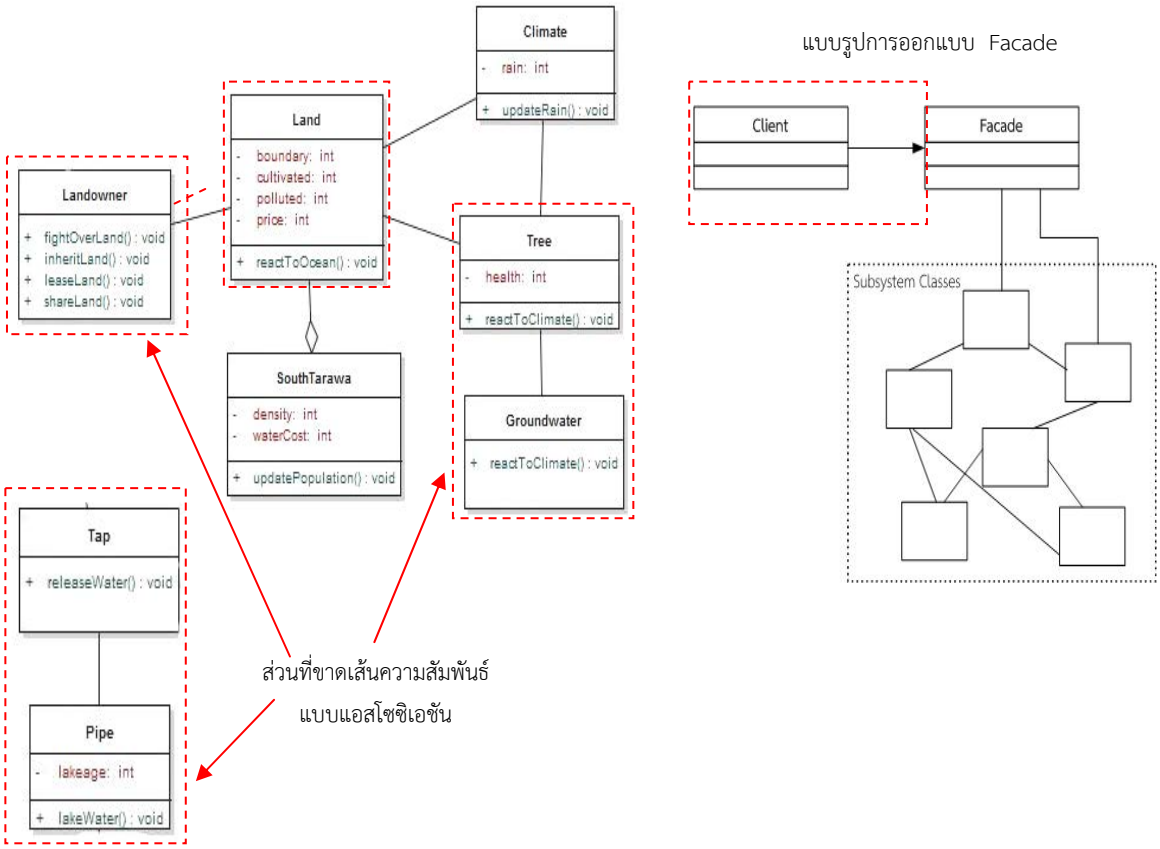
จากรูปที่ 5.4 สามารถตรวจจับแบบรูปการออกแบบด้วยเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้างได้ผลลัพธ์ ดังตารางที่ 5.2

ตารางที่ 5.2 ผลลัพธ์จากการทดสอบกรณีทดสอบที่สอง

แบบรูปการออกแบบ	จำนวนที่ตรวจพบ	ผลลัพธ์
Adapter Pattern	-	-
Bridge Pattern	-	-
Composite Pattern	-	-
Decorator Pattern	-	-
Facade Pattern	4	● Household - Child - Good ● Tree - Groundwater - loss Association from Client - 67%. ● Pipe - Tap - loss Association from Client - 67%. ● Land - Landowner – loss Association from Client - 67%.
Flyweight Pattern	-	-
Proxy Pattern	1	● HouseholdOuterIsland - HouseholdSouthTavrawa - Household - loss Association - 67%.

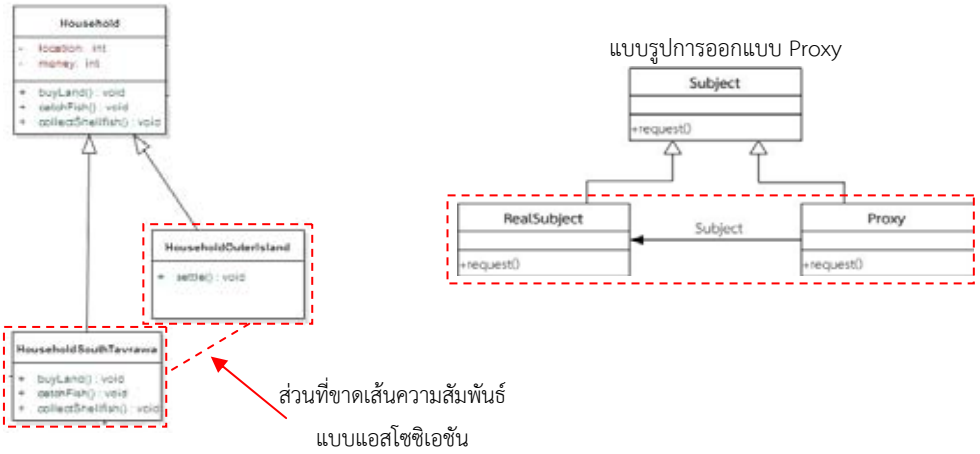
จากตารางที่ 5.2 ผลการตรวจจับแบบรูปการออกแบบจากแผนภาพคลาสกรณีทดสอบที่สองนั้นสรุปผล ดังนี้ สามารถตรวจจับแบบรูปการออกแบบเชิงโครงสร้างทั้งหมดที่มีอยู่สองแบบรูป คือ แบบรูปการออกแบบ Façade ตรวจจับคลาสที่ตรงตามแบบรูปการออกแบบทุกประการได้ จำนวน 1 แบบรูปและตรวจจับคลาสที่มีความเหมือนได้จำนวน 3 แบบรูป ได้แก่ Tree - Groundwater - loss Association from Client -67%., Pipe - Tap - loss Association from Client - 67%., Land - Landowner – loss Association from Client -67%. และตรวจพบแบบรูปการออกแบบ Proxy ซึ่งมีคลาสที่มีความเหมือนกับแบบรูปการออกแบบได้ จำนวน1 แบบรูป คือ HouseholdOuterIsland - HouseholdSouthTavrawa - Household - loss Association - 67%. โดยค่าร้อยละความเหมือนกับแบบรูป จะพิจารณาจำนวนคลาสที่ตรวจพบกับจำนวนคลาสของ

แบบรูปการออกแบบตามมาตรฐานและยังพิจารณาจากลำดับการตรวจจับคลาสที่มีความสัมพันธ์กัน โดยแบบรูปการออกแบบ Façade นั้น คลาสที่มีความเหมือนจะขาดเส้นความสัมพันธ์แบบแอสโซซิเอชันที่เชื่อมไปยังคลาสที่เป็น Client เมื่อนำไปเทียบกับแบบรูปการออกแบบ Facade ตามมาตรฐานแล้วแสดงการแปรผลได้ ดังรูปที่ 5.5



รูปที่ 5.5 กรณีทดสอบที่สอง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Façade

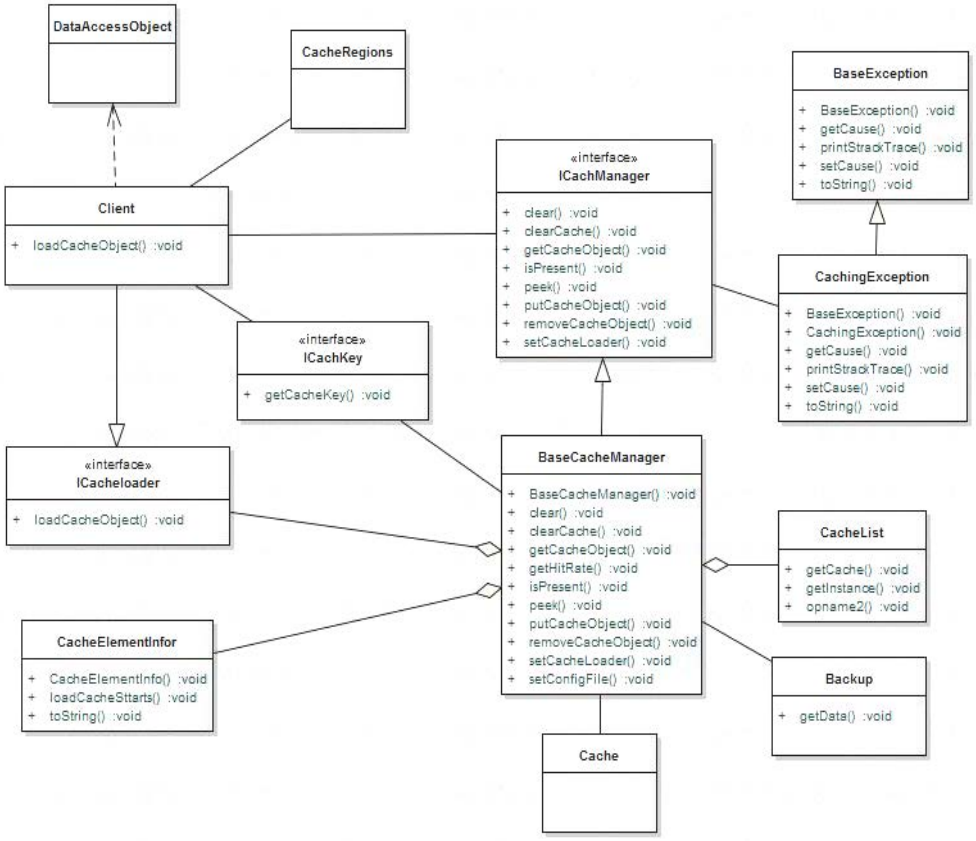
ส่วนแบบรูปการออกแบบ Proxy นั้น คลาสที่มีความเหมือนจะขาดเส้นความสัมพันธ์แบบแอสโซซิเอชันที่เชื่อมระหว่าง Proxy ไปยังคลาส RealSubject เมื่อนำไปเทียบกับแบบรูปการออกแบบ Proxy ตามมาตรฐานแล้วแสดงการแปรผลได้ ดังรูปที่ 5.6



รูปที่ 5.6 กรณีทดสอบที่สอง การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Proxy

5.1.3. การทดสอบกรณีทดสอบที่สาม

เป็นการทดสอบการตรวจหาแบบรูปการออกแบบเชิงโครงสร้าง จากระบบจริงโดยใช้แผนภาพคลาสจากการทำงานของระบบ Web Portal Caching ซึ่งแผนภาพคลาสสำหรับการทดสอบแสดง ดังรูปที่ 5.7



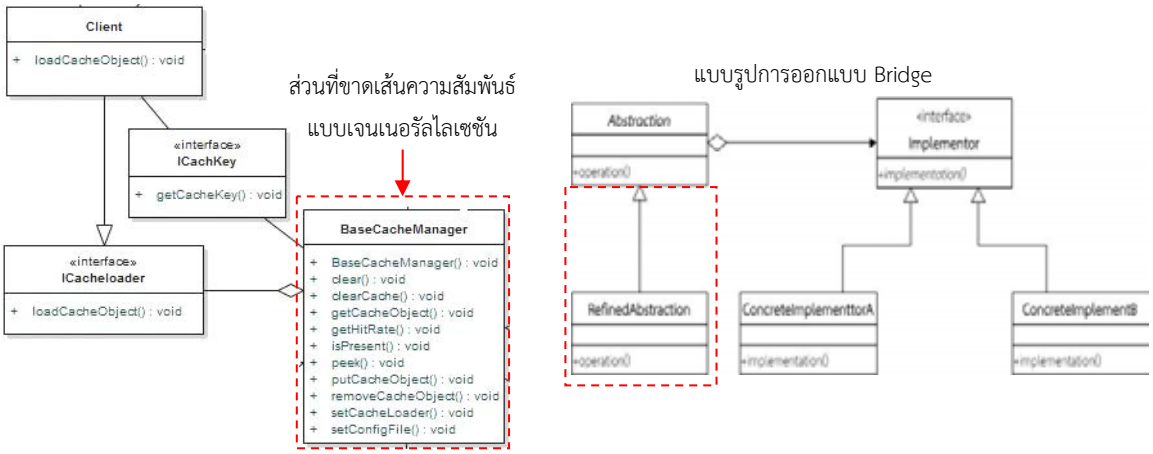
รูปที่ 5.7 กรณีทดสอบที่สามแผนภาพคลาส Web Portal Caching [13]

จากรูปที่ 5.7 สามารถตรวจจับแบบรูปการออกแบบด้วยเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้างได้ผลลัพธ์ ดังตารางที่ 5.3

ตารางที่ 5.3 ผลลัพธ์จากการทดสอบกรณีทดสอบที่สาม

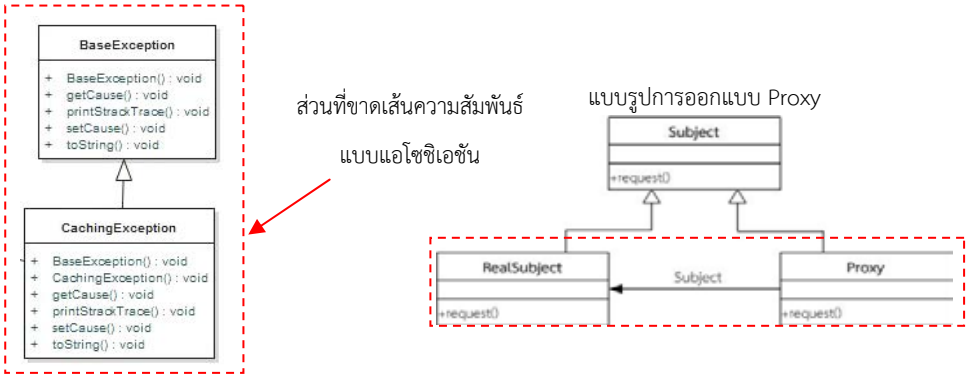
แบบรูปการออกแบบ	จำนวนที่ตรวจพบ	ผลลัพธ์
Adapter Pattern	-	-
Bridge Pattern	1	● ICacheloader - Client - BaseCacheManager - loss Generalization - 60%.
Composite Pattern	-	-
Decorator Pattern	-	-
Facade Pattern	1	● BaseCacheManager - Backup - Cache
Flyweight Pattern	-	-
Proxy Pattern	1	● CachingException - BaseException - loss Association - 67%.

จากตารางที่ 5.3 ผลการตรวจจับแบบรูปการออกแบบจากแผนภาพคลาสกรณีทดสอบที่สามนั้นสรุปผล ดังนี้ สามารถตรวจจับแบบรูปการออกแบบเชิงโครงสร้างทั้งหมดที่มีอยู่สามแบบรูป คือ แบบรูปการออกแบบ Bridge ตรวจจับคลาสที่มีความเหมือนได้ จำนวน 1 แบบรูป ได้แก่ ICacheloader - Client - BaseCacheManager - loss Generalization - 60%. ตรวจพบแบบรูปการออกแบบ Facade ตรวจจับคลาสที่ตรงตามแบบรูปการออกแบบทุกประการได้ จำนวน 1 แบบรูป ได้แก่ BaseCacheManager - Backup – Cache และตรวจพบแบบรูปการออกแบบ Proxy ตรวจจับคลาสที่มีความเหมือนได้ จำนวน 1 แบบรูป ได้แก่ CachingException - BaseException - loss Association 67%. โดยค่าร้อยละความเหมือนกับแบบรูป จะพิจารณาจำนวนคลาสที่ตรวจพบกับจำนวนคลาสของแบบรูปการออกแบบตามมาตรฐานและยังพิจารณาจากลำดับการตรวจจับคลาสที่มีความสัมพันธ์กันโดยแบบรูปการออกแบบ Bridge นั้น คลาสที่มีความเหมือนจะขาดเส้นความสัมพันธ์แบบเจนเนอรัลไลเซชันที่เชื่อมไปยังคลาสที่เป็น RefineAbstraction เมื่อนำไปเทียบกับแบบรูปการออกแบบ Bridge ตามมาตรฐานแล้วแสดงการแปรผลได้ดังรูปที่ 5.8



รูปที่ 5.8 กรณีทดสอบที่สาม การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Bridge

ส่วนแบบรูปการออกแบบ Proxy นั้น คลาสที่มีความเหมือนจะขาดเส้นความสัมพันธ์แบบแอสโซซิเอชัน ที่เชื่อมระหว่าง Proxy ไปยังคลาส RealSubject เมื่อนำไปเทียบกับแบบรูปการออกแบบ Proxy ตามมาตรฐานแล้วแสดงการแปรผลได้ดังรูปที่ 5.9



รูปที่ 5.9 กรณีทดสอบที่สาม การแปรผลการตรวจจับความเหมือนแบบรูปการออกแบบ Proxy

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงสรุปผลการวิจัย ปัญหาและข้อจำกัดที่พบจากงานวิจัย ประโยชน์ที่ได้รับจากงานวิจัย และแนวทางการพัฒนางานวิจัยต่อ จากการตรวจจับแบบรูปการออกแบบเชิงโครงสร้างทั้ง 7 แบบรูป ซึ่งอธิบายรายละเอียด ดังนี้

6.1. สรุปผลการวิจัย

งานวิจัยนี้ได้นำเสนอการปรับปรุงขั้นตอนวิธีการตรวจจับแบบรูปการออกแบบและเครื่องมือที่ใช้ในการตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง โดยสามารถตรวจจับแบบรูปการออกแบบเชิงโครงสร้างได้ 7 แบบรูปการออกแบบ ประกอบไปด้วย

1) แบบรูปการออกแบบ Adapter
2) แบบรูปการออกแบบ Bridge
3) แบบรูปการออกแบบ Composite
4) แบบรูปการออกแบบ Decorator
5) แบบรูปการออกแบบ Facade
6) แบบรูปการออกแบบ Flyweight
7) แบบรูปการออกแบบ Proxy

โดยการขั้นตอนวิธีการตรวจจับแบบรูปการออกแบบนั้นได้นำกราฟเชิงความสัมพันธ์มาใช้ในการพิจารณาการตรวจจับและมีการกำหนดลำดับของเส้นความสัมพันธ์ระหว่างคลาสตามประเภทของเส้นความสัมพันธ์เข้า–ออกที่แบ่งเป็น 12 ลำดับ เพื่อทำให้การตรวจจับมีความแม่นยำยิ่งขึ้น อีกทั้งยังทำการพัฒนาเครื่องมือมาใช้ในการตรวจจับแบบรูปดังกล่าวด้วย ซึ่งการทำงานของเครื่องมือจะทำการนำเข้าข้อมูลในลักษณะของไฟล์เอกซ์เอ็มไอที่มาจากการออกแบบแผนภาพคลาส เครื่องมือจะทำการกรองข้อมูลที่จำเป็นแล้วเลือกจัดเก็บลงในตารางคุณลักษณะท้องถิ่นและตารางความสัมพันธ์ เพื่อใช้เป็นข้อมูลในการเปรียบเทียบกับแบบรูปการออกแบบต่อไปขั้นตอนการตรวจจับแบบรูปการออกแบบจะแบ่งออกเป็นสอง

ส่วนด้วยกัน ส่วนแรกจะทำการตรวจจับโหนดอ้างอิงของแต่ละแบบรูปการออกแบบจากแผนภาพคลาสทั้งหมด เพื่อที่จะทำให้การตรวจจับทำได้รวดเร็วขึ้น จากนั้นจะดำเนินการในส่วนที่สองคือการตรวจจับรายละเอียดของแต่ละแบบรูปการออกแบบที่ได้ตรวจพบจากโหนดอ้างอิง ซึ่งโหนดอ้างอิงที่ได้มาจะนำมาหาโหนดที่เกี่ยวข้องสัมพันธ์กันกับแบบรูปการออกแบบ เครื่องมือนี้สามารถตรวจจับแบบรูปการออกแบบแล้วแสดงผลชื่อคลาส จำนวน ชนิดของแบบรูป อีกทั้งยังสามารถบอกถึงความเหมือนของแผนภาพคลาสที่คาดว่าจะเป็นแบบรูปการออกแบบเชิงโครงสร้างได้

จากการทดสอบขั้นตอนวิธีและเครื่องมือสามารถใช้ในการตรวจจับแบบรูปการออกแบบเชิงโครงสร้างทั้งเจ็ดแบบรูป ซึ่งสามารถตรวจจับได้ในลักษณะที่มีความเหมือนกับแบบรูปการออกแบบทุกประการและความเหมือนบางส่วนที่พิจารณาจากค่าร้อยละของจำนวนคลาสที่ตรวจพบต่อจำนวนคลาสตามมาตรฐานของแต่ละแบบรูปการออกแบบ ในทดสอบเครื่องมือจะใช้ข้อมูลสองรูปแบบ คือ กรณีศึกษาที่สร้างขึ้นเองซึ่งจะมีจำนวนแบบรูปการออกแบบเชิงโครงสร้างครบทั้ง 7 แบบรูปและข้อมูลจากระบบจริงที่มีแบบรูปการออกแบบอยู่ทั้งแบบเหมือนทุกประการและมีความเหมือนบางส่วน จากการทดสอบจะพบได้ว่าสามารถตรวจจับแบบรูปการออกแบบเชิงโครงสร้างที่มีความเหมือนทุกประการและมีความเหมือนบางส่วนได้

6.2. ปัญหาและข้อจำกัดที่พบจากงานวิจัย

1) การออกแบบแผนภาพคลาสต้องเป็นไปตามมาตรฐานยูเอ็มแอล โดยมีการระบุถึงชื่อคลาส ชนิดของคลาส ชื่อเมธ็อดและเส้นความสัมพันธ์ที่มีการระบุทิศทางระหว่างคลาส เป็นอย่างน้อย

2) เครื่องมือที่นำออกแผนภาพคลาสเป็นเอกซ์เอ็มไอ นั้นจะต้องนำออกด้วยเครื่องมือ Enterprise Architect รุ่นที่ 7.0 เป็นอย่างน้อย

3) ข้อมูลนำออกในรูปแบบของไฟล์เอกซ์เอ็มไอต้องเป็นเอกสารที่มีความถูกต้องและสมบูรณ์ (Well-formed) เท่านั้น

4) แบบรูปการออกแบบในหมวดอื่นๆ ที่มีลักษณะคล้ายกับแบบรูปการออกแบบเชิงโครงสร้าง เครื่องมือตรวจจับแบบรูปการออกแบบจะแสดงว่าเป็นแบบรูปการออกแบบเชิงโครงสร้าง
5) การตรวจจับแผนภาพคลาสที่มีลักษณะของการสืบทอดคุณสมบัติ จะต้องมีการระบุชื่อเมธ็อดที่สืบทอดมาด้วย เพื่อใช้เป็นข้อมูลในการพิจารณาในการตรวจจับที่มีความแม่นยำยิ่งขึ้น

6.3. ประโยชน์ที่ได้รับจากงานวิจัย

1) นำเสนอทางเลือกของขั้นตอนวิธีสำหรับการตรวจจับแบบรูปการออกแบบเชิงโครงสร้างโดยใช้กราฟเชิงความสัมพันธ์

2) ได้เครื่องมือการตรวจจับแบบรูปการออกแบบที่แบบรูปการออกแบบเชิงโครงสร้าง

3) เครื่องมือการตรวจจับแบบรูปการออกแบบที่นำเสนอ สามารถตรวจจับแบบรูปการออกแบบเชิงโครงสร้างที่มีความเหมือนบางส่วนกับแบบรูปการออกแบบ ซึ่งสามารถนำข้อมูลที่ได้มาใช้วิเคราะห์ปรับแก้ไขการออกแบบแผนภาพคลาสให้ตรงตามแบบรูปการออกแบบเบื้องต้นได้

6.4. แนวทางในการพัฒนางานวิจัยต่อ

งานวิจัยนี้มีแนวทางในการพัฒนาเพิ่มเติมและข้อเสนอแนะ เพื่อเป็นแนวทางในการพัฒนาในหลายด้าน ดังนี้

1) พัฒนาขั้นตอนวิธีการตรวจจับแบบรูปการออกแบบอีก 2 ประเภท คือ แบบรูปการออกแบบการสร้างวัตถุและแบบรูปการออกแบบเชิงพฤติกรรม

2) พัฒนาขั้นตอนวิธีการตรวจจับแบบรูปการออกแบบที่เป็นมาตรฐานกลางที่สามารถใช้ในการตรวจจับได้ทุกแบบรูปการออกแบบ โดยไม่ต้องพิจารณาจากโหนดอ้างอิงในการเริ่มต้นการตรวจจับ

3) พัฒนาขั้นตอนวิธีให้สามารถรายงานผลการตรวจจับแบบรูปการออกแบบที่มีความเหมือนบางส่วนให้แปรผลการตรวจจับเทียบกับแต่ละแบบรูปการออกแบบโดยละเอียดและแม่นยำยิ่งขึ้น

รายการอ้างอิง

[1] Nikolaos, T., Alexander, C., George S., and Spyros, T.H. Design Pattern Detection using Similarity Scoring. IEEE Transactions on Software Engineering (TSE), 2006

[2] Jing, D., Yongtao, S., and Yajing, Z. Design Pattern Detection by Template Matching. Symposium on Applied Computing (SAC), 2008

[3] Manjari, G., Rajwant, S.R., and Anil, K.T. Design Pattern Detection using Inexact Graph Matching. presented at the Communication and Computational Intelligence (CCI), India December 27-29, 2010.

[4] Umut, T., Ural, E., and Feza, B. Mining Object-Oriented Design Models for Detecting Identical Design Structures, International Workshop on Software Clones (IWSC), 2012

[5] Gamma, E. Design Patterns Elements of Reusable Object-Oriented Software, 1995

[6] Marcus, A., and Ivan, P. Model Interchange Using OMG Standards. Software Engineering and Advanced Applications (SEAA), 2005

[7] Jing D., Yajing, Z., and Tu, P. A Review of Design Pattern Mining Techniques. International Journal of Software Engineering and Knowledge Engineering (IJSEKE), 2008

[8] Linda, G.S., and Robert, M.H. A metric for comparing relational descriptions. IEEE PAMI 7 pp. 90-94. 1985.

[9] Marcus, A., and Ivan, P. Model Interchange Using OMG Standards. Conference on Software Engineering and Advanced Applications (SEAA), 2005.

[10] IBM Corporation. Relationship types of UML [Online]. 2005. Available from: http://publib.boulder.ibm.com/infocenter/rsdvhelp/ [2012,11November].

[11] Apostolos, N.P., and Yannis, M. Structure-Based Similarity Search with Graph Histograms. Database and Expert Systems Applications (DEXA), 1999.

12] Pascal, P., Christophe, L.P., Patrick, D.A., and Ian, W. AtollGame: A Companion Modeling Experience in The Pacific. Complex Science for a Complex World: Exploring Human Ecosystems with Agents (pp. 255-280). 2006

[13] Web portal caching class diagram. Object Caching in a Web Portal Application Using JCS. [Online]. 2003 Available from: http://www.onjava.com/pub/a/onjava/2003/12/23/caching.html [2013,16Septermber].

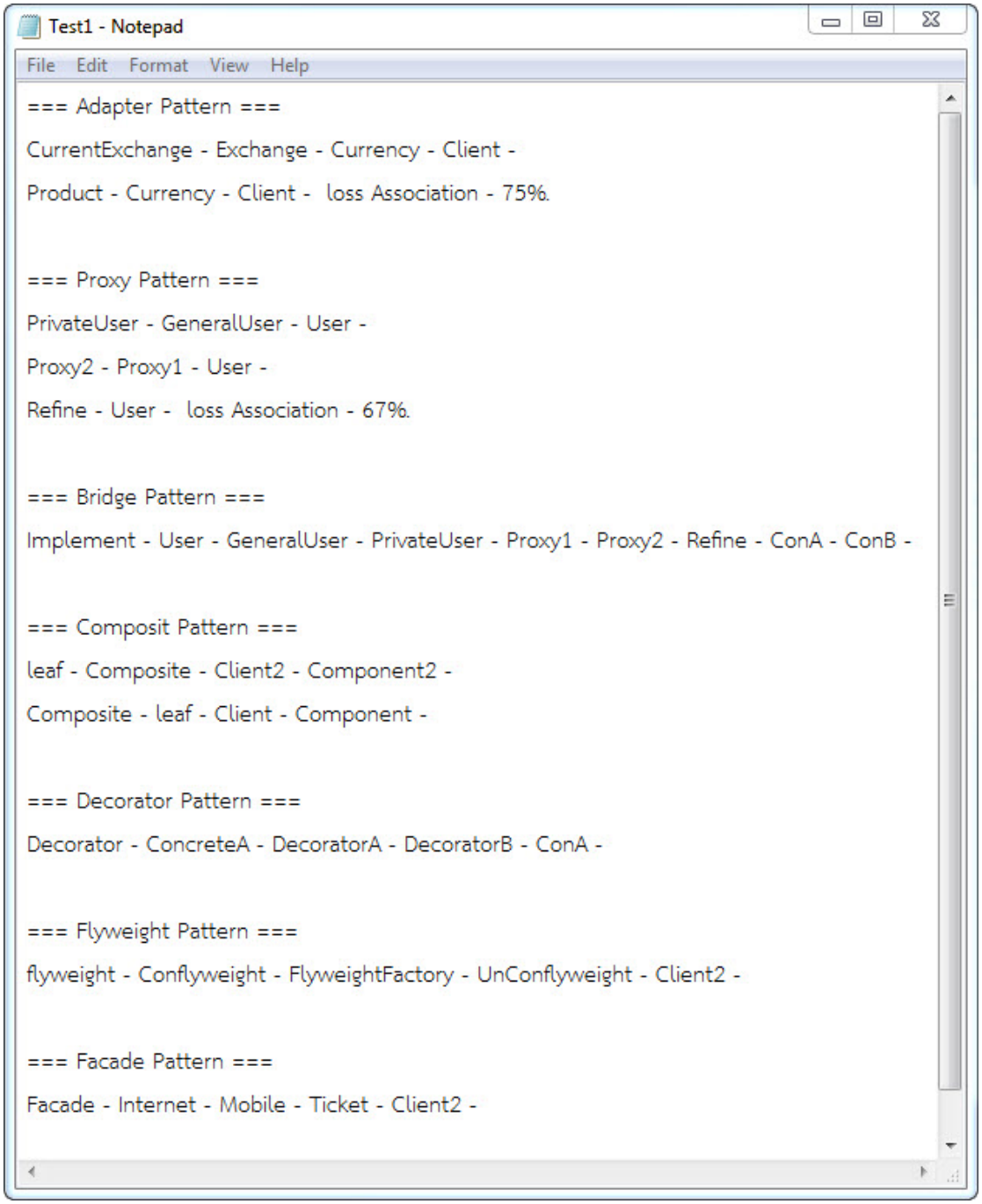
[14] นุชนาถ สัตย์วินิจ,การตรวจจับแบบรูปการออกแบบเชิงพฤติกรรมด้วยแผนภาพคลาส, วิทยานิพนธ์ปริญญามหาบัณฑิต,ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2554.

ภาคผนวก

ภาคผนวก ก

ผลลัพธ์ที่ได้จากการทดสอบขั้นตอนวิธีและเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง กรณีทดสอบที่หนึ่ง

ผลลัพธ์ที่ได้จากการทดสอบขั้นตอนวิธีและเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง กรณีทดสอบที่หนึ่ง

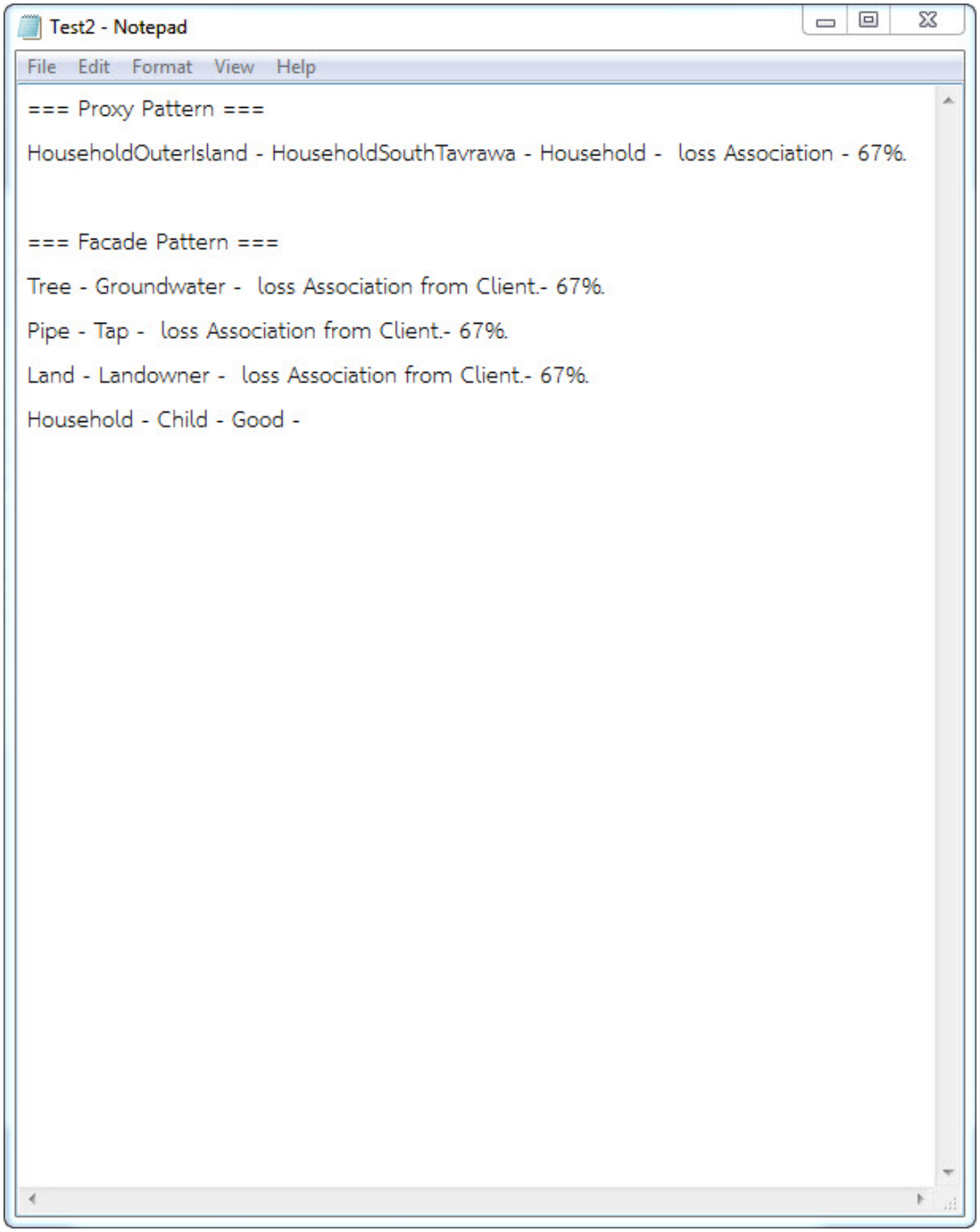


ภาพที่ ก.1 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ กรณีทดสอบที่หนึ่ง

ภาคผนวก ข

ผลลัพธ์ที่ได้จากการทดสอบขั้นตอนวิธีและเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง กรณีทดสอบที่สองแผนภาพคลาส AtollGame

ผลลัพธ์ที่ได้จากการทดสอบขั้นตอนวิธีและเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง กรณีทดสอบที่สองแผนภาพคลาส AtollGame

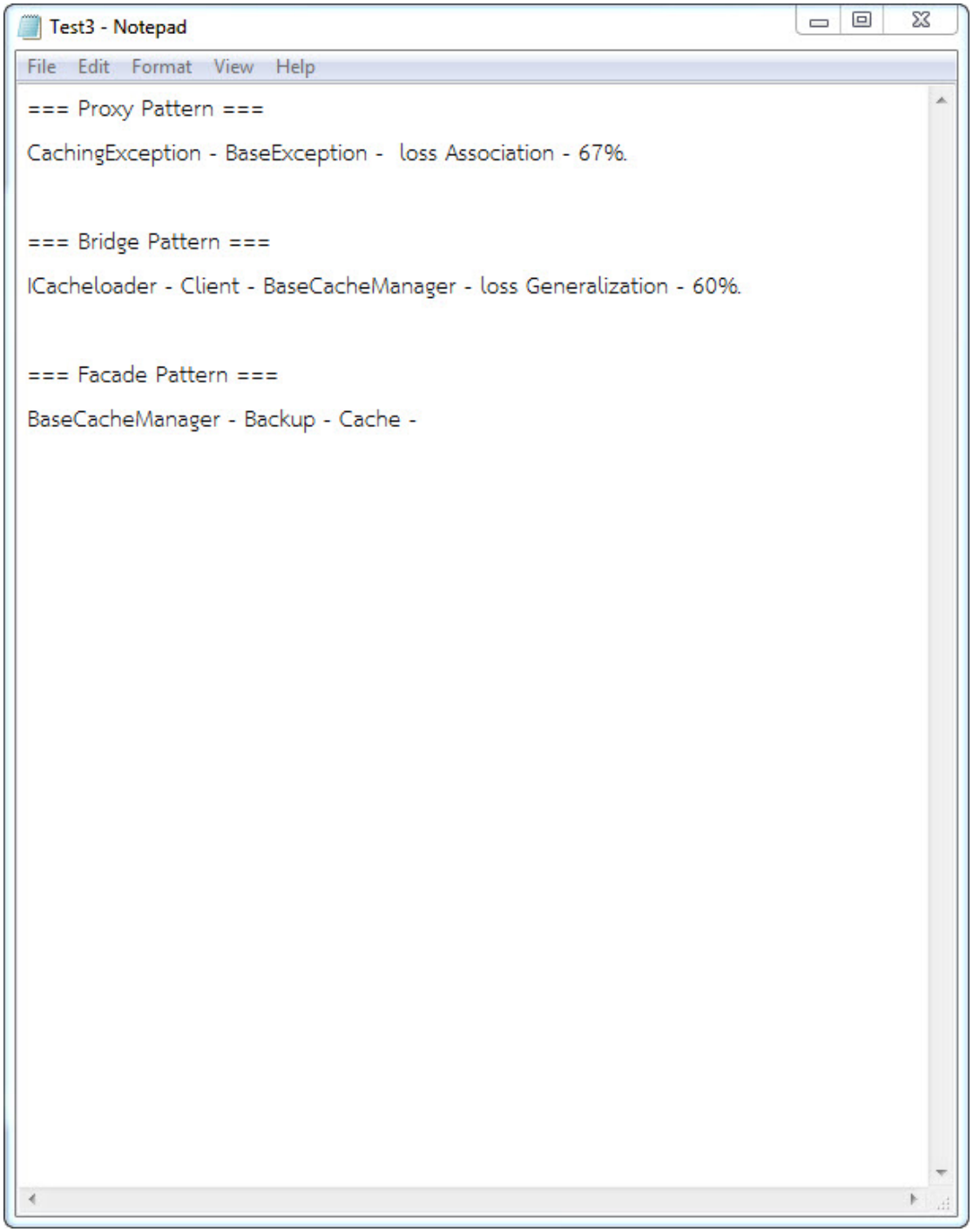


ภาพที่ ข.1 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ
กรณีทดสอบที่สองแผนภาพคลาส AtollGame

ภาคผนวก ค

ผลลัพธ์ที่ได้จากการทดสอบขั้นตอนวิธีและเครื่องมือตรวจจับแบบรูปการ
ออกแบบเชิงโครงสร้าง กรณีทดสอบที่สามแผนภาพคลาส Web Portal Caching

ผลลัพธ์ที่ได้จากการทดสอบขั้นตอนวิธีและเครื่องมือตรวจจับแบบรูปการออกแบบเชิงโครงสร้าง กรณีทดสอบที่สามแผนภาพคลาส Web Portal Caching



ภาพที่ ค.1 ผลลัพธ์การตรวจจับแบบรูปการออกแบบ
กรณีทดสอบที่สามแผนภาพคลาส Web Portal Caching

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวชลิดา เหลี่ยมวิเศษ เกิดเมื่อวันที่ 24 สิงหาคม พุทธศักราช 2531 ที่จังหวัดสระบุรี สำเร็จการศึกษาระดับปริญญาวิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศธุรกิจ เกียรตินิยมอันดับสอง จากคณะเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยศิลปากร ในปีการศึกษา 2554 และในปีการศึกษา 2554 เข้าศึกษาต่อในระดับปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย