

วิทยานิพนธ์นี้นำเสนอการเปรียบเทียบโครงสร้างคลาสดของซอฟต์แวร์ ระหว่างโครงสร้างคลาส ในความสัมพันธ์แบบแอสโซซิเอชันและเจเนอรัลไลเซชันใน 2 มุมมอง คือ (1) ประสิทธิภาพ (Performance) ของโครงสร้างคลาสด้วยซอฟต์แวร์ขณะประมวลผล โดยวัดจากจำนวนการรับส่งหรือเรียกใช้งานด้วยเมสเสจ (Message Calling) ระหว่างคลาสที่เป็นองค์ประกอบของซอฟต์แวร์และความเร็วในการประมวลผล (Response Time) และ (2) ความสามารถในการบำรุงรักษาซอฟต์แวร์ (Maintainability) เมื่อมีการเปลี่ยนแปลงซอฟต์แวร์ (Software Change) ตามความต้องการเชิงฟังก์ชัน (Functional Requirement) โดยวัดจากจำนวนคลาสและจำนวนเมธอดที่ได้รับผลกระทบจากการเปลี่ยนแปลง

งานวิจัยเป็นการวิจัยเชิงทดลอง โดยออกแบบและพัฒนาเครื่องมือ (ในรูปแบบซอฟต์แวร์) สำหรับทดลอง 2 ชุด คือ (1) เครื่องมือวัดประสิทธิภาพของโครงสร้างคลาสด้วยซอฟต์แวร์ขณะประมวลผล โดยหน่วยตัวอย่างที่ใช้ทำการทดลองเป็นซอฟต์แวร์ที่พัฒนาภายใต้โครงสร้างคลาสในความสัมพันธ์แบบเจเนอรัลไลเซชัน จำนวน 5 หน่วยตัวอย่าง แล้วนำมาเปลี่ยนเป็นโครงสร้างคลาสในความสัมพันธ์แบบแอสโซซิเอชันและพัฒนาเป็นซอฟต์แวร์ แล้วนำซอฟต์แวร์ทั้งสองโครงสร้างคลาสมาประมวลผลด้วยการทำทรานแซคชัน (Transaction) และ (2) เครื่องมือวัดผลกระทบในการเปลี่ยนแปลงความต้องการของซอฟต์แวร์ โดยหน่วยตัวอย่างเป็นความต้องการเชิงฟังก์ชันของซอฟต์แวร์จำนวน 30 หน่วยตัวอย่าง แล้วจัดเก็บจำนวนคลาสและจำนวนเมธอดที่เกี่ยวข้องกับการเปลี่ยนแปลงซอฟต์แวร์ที่พัฒนาภายใต้โครงสร้างคลาสในความสัมพันธ์แบบแอสโซซิเอชันและเจเนอรัลไลเซชันด้วยการเพิ่มความต้องการเชิงฟังก์ชัน

โดยผู้วิจัยกำหนดให้หน่วยตัวอย่างเป็นซอฟต์แวร์ที่พัฒนาด้วยภาษาจาวาและมีจำนวนคลาสการทำงานในเชิงธุรกิจอย่างน้อย 5 คลาส และวิเคราะห์ผลโดยวิธีการทางสถิติด้วยวิลคอกสันไซน์แรงค์เทสที่ระดับนัยสำคัญเท่ากับ 0.05

ผลการทดลองพบว่าจำนวนการรับส่งข้อความ ระยะเวลาในการประมวลผล รวมถึงจำนวนคลาสและจำนวนเมธอดที่ได้รับผลกระทบจากการเปลี่ยนแปลงของโครงสร้างคลาสแบบแอสโซซิเอชันน้อยกว่าแบบเจเนอรัลไลเซชัน จึงสรุปได้ว่า โครงสร้างคลาสแบบแอสโซซิเอชันมีประสิทธิภาพ และมีความสามารถในการปรับปรุง บำรุงรักษาซอฟต์แวร์ดีกว่าโครงสร้างคลาสแบบเจเนอรัลไลเซชัน

4782438526 : MAJOR BUSINESS SOFTWARE DEVELOPMENT

KEY WORD: CLASS MODEL / ASSOCIATION RELATIONSHIP / GENERALIZATION

RELATIONSHIP

SUPHANSA NEKAMANURAK : THE COMPARATIVE STUDY OF SOFTWARE'S PERFORMANCE AT RUN – TIME AND IMPACT FROM SOFTWARE CHANGE ON ASSOCIATION AND GENERALIZATION STRUCTURE.

THESIS ADVISOR : ASST. PROF. SOMJAREE PREEYANONT, Doc.Eng., 166 pp.

This thesis presents the model comparison between two software class models : Association and Generalization Relationships. There are two viewpoints to be considered. The first one is the performance of the software during the execution. It can be determined by number of message calling and response time. The second viewpoint is the maintainability for software change according to the functional requirements. It is obtained by the number of class and method of the change impact.

This research is an experimental research. The design and development of tool (software) for two works is carried out. The first work is tool for analyzing the performance of the class by the software during the execution. Five sample units of the experiment are the software developed under Generalization Relationship class model. Then they will be changed to the Association Relationship class model and developed to the software, respectively. These two class models of the software will be executed by the transactions. The second work is the tool for the analyzing the change impact of the software as number of class and number of method that are affected. There are 30 sample units of the functional requirements in order to modify the Association and Generalization Relationships class models.

This thesis's sample unit is software developed based on Java that consist of least five business classes. The results are analyzed by statistical method of Wilcoxon Signed Ranks at 0.05 significant level.

From the experiment, it is found that the number of message calling, response time, class and method that are affected from the software change with the Association Relationship class model less than the Generalization Relationship class model. It can be concluded that the performance and maintainability of software developed with the Association Relationship class model is better than software developed with the Generalization Relationship class model.