UNIVERSITI PUTRA MALAYSIA

REQUIREMENT-ORIENTED ASPECT REFACTORING FOR EARLY ASPECT FORMATION AND MAPPING IN SOFTWARE MAINTENANCE

HEMA A/P SUBRAMANIAM

FSKTM 2016 3

REQUIREMENT-ORIENTED ASPECT REFACTORING FOR EARLY ASPECT FORMATION AND MAPPING IN SOFTWARE MAINTENANCE

By

HEMA A/P SUBRAMANIAM

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfilment of the Requirement for the Degree of Doctor of Philosophy

January 2016

This thesis work is dedicated to my husband, Sivabalan, who has been a constant source of support and encouragement during the challenges of my academic career. I am truly thankful for having you in my life.

This work is also dedicated to my children, Kartti and Harinesvari, who have always loved me unconditionally.

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirement for the degree of Doctor of Philosophy

REQUIREMENT-ORIENTED ASPECT REFACTORING FOR EARLY ASPECT FORMATION AND MAPPING IN SOFTWARE MAINTENANCE

By

HEMA A/P SUBRAMANIAM

January 2016

Chair: **Hazura Binti Zulzalil, PhD**
Faculty: **Computer Science and Information Technology**

Modularity aims to increase the maintainability of a software program by fostering reusability, assist in reducing development cost and enhancing the quality of the software. Although modularity gaining popularity, yet it hard to be realized due to the existence of stakeholder interest in the primary programs. Thus, maintainability became an unresolved issue. This issue may be compromised by isolating stakeholder interests, yet the system still contains interests which hard to be located. Therefore, an essential approach for managing and controlling those crosscutting stakeholder interests became tremendously important. Accordingly, Aspect-Oriented Approach (AOA) is employed the concept of crosscutting concern and aspect as the representation of those stakeholder interests. At the present time, the software practitioners have preferred to conduct crosscutting concern isolation at coding level, which have been resulting in ambiguous situation. Therefore, the crosscutting concern isolation process was transformed into requirement specifications. Since Early Aspect concept stay in line with new software development, the attempt to make existing application's to be AOA compatible became the matter in question. For this reason, the refactoring effort which started at requirement specification has become absolutely necessary. In that case, this study seeks to address the issue by proposing a conceptual framework known as Requirement-Oriented Aspect Refactoring (ReqOAR). In order to view the issue comprehensively, the ReqOAR aims to handle concern at requirement level and to facilitate aspect candidate flow throughout existing software artefacts. To foster effective concern handler at requirement level, ReqOAR conceptual framework were included with the process components such as concern identification, concern isolation and aspect formation. Whereas, to ensure the flow of aspect candidate throughout the software artefacts, ReqOAR comprise of process called as concern mapping. Apart from process

components, ReqOAR also contain techniques components which associated with aforementioned process components. For instance, Crosscutting Concern Domain Library Listing (CCDLL) and Concern Associated Terms Glossary (CATG) are among the ReqOAR proposed techniques. Furthermore, ReqOAR was validated using empirical evaluation. According to the evaluation, an optimum accuracy value was recorded from the treatment technique compared to control technique. These current findings would definitely empower the growing body of AOA in the context of restructuring the requirement specification.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia
Sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

PENSTRUKTURAN SEMULA ASPEK BERORIENTASIKAN KEPERLUAN UNTUK PEMBENTUKAN AWAL ASPEK DAN PEMETAAN SEMASA PENYELENGGARAAN PERISIAN

Oleh

HEMA A/P SUBRAMANIAM

Januari 2016

Pengerusi: **Hazura Binti Zulzalil, PhD**
Fakulti: **Sains Komputer dan Teknologi Maklumat**

Kemodularan bertujuan untuk meningkatkan penyelenggaraan perisian dengan memupuk kebolehan untuk penggunaan semula, membantu dalam mengurangkan kos pembangunan dan meningkatkan kualiti perisian. Walaupun kemodularan semakin popular, tetapi ianya sukar untuk direalisasikan kerana adanya kepentingan pemegang taruh dalam aplikasi utama aplikasi. Oleh itu, penyelenggaraan menjadi isu yang tidak berkesudahan. Isu ini boleh dikompromi dengan mengasingkan kepentingan tersebut, tetapi kebanyakan sistem mengandungi kehendak yang sukar untuk dikesan. Jadi, satu pendekatan asas untuk mengurus dan mengawal kehendak tersebut amat diperlukan. Sehubungan itu, pendekatan berorientasikan-aspek (AOA) menekankan konsep saling pemangkasan kepentingan dan aspek mewakili kehendak pihak berkepentingan. Pada masa ini, pengamal perisian lebih suka melaksanakan aktiviti pemisahan saling pemangkasan kepentingan di peringkat pengekodan, yang mana ianya boleh menyebabkan keadaan ketaksaan (kekeliruan) berlaku. Oleh itu, pemisahan saling pemangkasan telah dialihkan ke spesifikasi keperluan. Oleh kerana, konsep Aspek Awal (EA) memberi perhatian kepada pembangunan perisian baharu, konsep EA menjadi persoalan untuk sistem yang sedia ada supaya serasi dengan pendekatan berorientasikan aspek. Atas alasan ini, usaha menstruktur semula aplikasi perisian pada spesifikasi keperluan amatlah diperlukan. Sehubungan itu, kajian ini bertujuan untuk membangunkan rangka kerja konseptual yang dikenali sebagai Penstrukturan Semula Aspek Berorientasikan Keperluan *(ReqOAR)*. Bagi melihat isu tersebut secara menyeluruh, ReqOAR berperanan mengendali saling pemangkasan kepentingan pada peringkat spesifikasi keperluan dan membantu aspek yang berpotensi bergerak melalui seluruh artifak perisian sedia ada. ReqOAR juga mengandungi komponen proses

seperti pengenalan kepentingan, pemisahan kepentingan dan pembentukan aspek bagi menggalakkan keberkesanan pengendali kepentingan pada peringkat keperluan. Manakala, untuk memastikan aspek yang berpotensi bergerak melalui seluruh artifak perisian, ReqOAR mengandungi proses yang digelar sebagai pemetaan kepentingan. Selain daripada komponen proses, ReqOAR juga mengandungi komponen teknik yang berkaitan dengan komponen prosesnya. Misalnya, *Crosscutting Concern Domain Library Listing* (CCDLL) dan *Concern Associated Terms Glossary (CATG)* adalah antara teknik yang dicadangkan berkaitan dengan ReqOAR. Selanjutnya, kaedah empirikal telah digunakan untuk menentusahkan keberkesanan rangka kerja ReqOAR yang dicadangkan. Berdasarkan kepada penilaian tersebut, ketepatan yang optimum telah direkodkan daripada teknik rawatan iaitu ReqOAR berbanding dengan teknik kawalan. Penemuan ini, semestinya akan memantapkan pengetahuan AOA dalam konteks penstrukturan semula spesifikasi keperluan.

ACKNOWLEDGEMENTS

First and above all, I praise to God, the almighty for providing me this opportunity and granting me the capability to proceed successfully. I would like to express the deepest appreciation to my supervisory committee chair, Associate Professor Dr Hazura Zulzalil, who has the attitude and the substance of a genius. She continually and convincingly conveyed a spirit of adventure in regard to research and an excitement in regard to teaching. I would like to thank my supervisory committee members, Associate Professor Dr Marzanah A.Jabar and Dr Saadah Hassan, whose work demonstrated to me that concern for global affair supported by an engagement in comparative literature and modern technology, should always transcend academia and provide a guest for our times. I must express my gratitude to Sivabalan, my husband for his continuous support and encouragement. I was continuously amazed by his willingness to proof read countless pages of this thesis and by the patience of my husband's family members who experience all of the ups and downs during my academic career. I would like to offer my sincere thanks to Faculty of Computer Science and Information Technology, Universiti Selangor, for giving me opportunity and space to conduct my research activities. Finally would like to thank the Faculty of Computer Science and Information Technology, Universiti Putra Malaysia and Higher Education Ministry of Malaysia for providing funding which allowed me to undertake this research.

I certify that a Thesis Examination Committee has met on 21st January 2016 to conduct the final examination of Hema A/P Subramaniam on her thesis entitled "Requirement-Oriented Aspect Refactoring for Early Aspect Formation and Mapping in Software Maintenance" in accordance with the Universities and University Colleges Act 1971 and the constitution of the Universiti Putra Malaysia [P.U.(A) 106] 15 March 1998. The Committee recommends that the student be awarded the Doctor of Philosophy.

Members of the Thesis Examination Committee were as follows:

Masrah Azrifah Azmi Murad, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Rusli Abdullah, PhD
Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

Abu Bakar Md. Sultan, PhD
Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

Ana Moreira, PhD
Associate Professor
Faculty of Science and Technology
Universidade Nova De Lisboa
Portugal
(External Examiner)

ZULKARNAIN ZAINAL, PhD
Professor and Deputy Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 21st April 2016

This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee were as follows:

Hazura Binti Zulzalil, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Marzanah Binti A.Jabar, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

Saadah Binti Hassan, PhD
Senior Lecturer
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

BUJANG BIN KIM HUAT, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 21st April 2016

Declaration by Graduate Student

I hereby confirm that:

- this thesis is my original work;
- quotations, illustrations and citations have been duly referenced;
- this thesis has not been submitted previously or concurrently for any other degree at any other institutions;
- intellectual property from the thesis and copyright of thesis are fully-owned by Universiti Putra Malaysia, as according to the Universiti Putra Malaysia (Research) Rules 2012;
- written permission must be obtained from supervisor and the office of Deputy Vice-Chancellor (Research and Innovation) before thesis is published (in the form of written, printed or in electronic form) including books, journals, modules, proceedings, popular writings, seminar papers, manuscripts, posters, reports, lecture notes, learning modules or any other materials as stated in the Universiti Putra Malaysia (Research) Rules 2012;
- there is no plagiarism or data falsification/fabrication in the thesis, and scholarly integrity is upheld as according to the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) and the Universiti Putra Malaysia (Research) Rules 2012. The thesis has undergone plagiarism detection software.

Signature: _____ Date: _____

Name and Matric No.: Hema A/P Subramaniam (GS32465)

Declaration by Members of Supervisory Committee

This is to confirm that:

- the research conducted and the writing of this thesis was under our supervision;
- supervision responsibilities as stated in the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) are adhered to.

Signature: _____

Name of Chairman
of Supervisory Committee: Hazura Binti Zulzalil, PhD

Signature: _____

Name of Member
of Supervisory Committee: Marzanah Binti A.Jabar, PhD

Signature: _____

Name of Member
of Supervisory Committee: Saadah Binti Hassan, PhD

TABLE OF CONTENTS

IST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATION

SoC	Separation of Concern
AOA	Aspect Oriented Approach
AOSD	Aspect Oriented Software Development
IEEE	The Institutes of Electrical and Electronic Engineers
ReqOAR	Requirement Oriented Aspect Refactoring Framework
AOSD	Aspect Oriented Software Development
CCDLL	Crosscutting Concern Domain Library Listing
AORE	Aspect Oriented Requirement Engineering
AOM	Aspect Oriented Modelling
AOP	Aspect Oriented Programming
AOT	Aspect Oriented Testing
UML	Unified Modelling Language
CATG	Concern Associated Terms Glossary
ALG	Aspect Lexicon Guideline
HW	Health Watcher
NFR	Non-Functional Requirement

CHAPTER 1

INTRODUCTION

1.1 Overview

This chapter explores the requirement-oriented aspect refactoring as a research topic. Beginning with the overview of the Aspect - Oriented Approach (AOA), it covers a situation where aspect refactoring known to be a central challenge for requirement engineers. Apart from the limitation of existing aspect refactoring methods were explained, this chapter argues that the flow of aspect candidate throughout the software process might be a useful indicators for comprehensive aspect refactoring. Since the terminology used in AOA is similar to natural language, this chapter provides the research associated terminologies as well. The chapter also gives the list of contributions made by this thesis, followed by thesis scope. Finally, this chapter provides a brief explanation on the remaining chapters at the end of this chapter.

1.2 Research Background

Modularization is one of the sub-characteristics listed under the software maintenance quality factors. Modularity refers to an ability of a program to be breakdown into a smaller and more manageable units (Singh and Goel, 2007). Previously, ISO 9126 had excluded modularization from maintainability. However in recent years, modularization has gaining popularity among researchers and software practitioners. Indeed, ISO 25010 had defined maintainability by extending it with two more sub-characteristics: reusability and modularity (ISO Joint Technical Committee, 2011). Thus, it shows that there is an increasing interest on modularization among upcoming industries. As such, the software programs which equipped with a comprehensive modular decomposition techniques become the centre of focus.

Modularity refers to degree to which a program composed of a smaller components, such that changes in one part expected to have minimal impact on the another component (Yu, Ramaswamy and Vaidyanathan, 2012). This definition notably exhibits the low coupling among those components. So, the Object-Oriented Approach (OOA) has practices low coupling and high cohesion among their class instances. However, stakeholder interest such as response time, authorization, authentication, consistency and other non-functional requirements are still remain to be a part of class instances which can impact the modularity. Since it repeatedly used in many parts of an application, OOA seems not to be comprehensively adhere to modularization

requirements (Mcheick, Mili, Sadou, and El-kharraz, 2006**;** Madeyski and Szala, 2007).

Stakeholder interest explores stakeholder concern towards an application (Mussbacher, Amyot, and Edward, 2009). Thus, stakeholder interest also known as concern. Concern can be in a form of non-functional characteristics, functional characteristics or system constraints. Generally in object-oriented application, a concern is common to few numbers of classes and methods. In that case, a concern creates a tendency to crosscut among few numbers of classes and methods. Those crosscutting feature known as crosscutting concerns. The detachment of those crosscutting concerns and an explicit definition of it is essential in modularizing the software applications comprehensively.

For this reason, in the year of 1997 Aspect-Oriented Approach (AOA) was proposed to exhibit the concept of Separation of Concern (SoC) (Kiczales et al., 2007). In essence, the separated crosscutting concerns composed into aspect instances as illustrated in Figure 1.1. This indicate the embark AOA as one of the prominent modular decomposition approach (Yvonne, Hans-Arno, and Mario, 2006; Wehrmeister, Pereira, and Rammig, 2013).

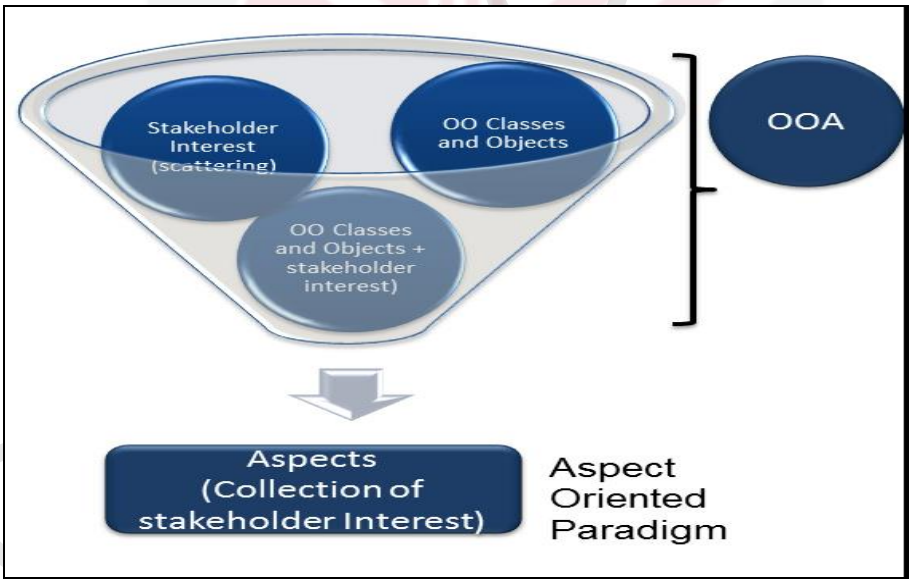

Figure 1.1. Aspect-Oriented Approach Overview

Knowingly the benefits of AOA, most of the software developers have started to manifest the aspect into their upcoming software projects. However, the focus on the existing application to be compatible with AOA was less explored. Therefore, the changes to the existing application were viewed as absolutely necessary (Miguel and Fernandes, 2006). For this reason, the refactoring efforts towards the existing applications are fundamentally important. Refactoring refers to the process of changing the internal structure of the program by remaining external behaviour of it**.** In this case, internal structure refers to the base program which already in object-oriented manner. Aspect refactoring aims to restructuring the existing applications to be compatible with AOA (Mortensen, Ghosh and Bieman, 2012).

Even though, the refactoring activities become a key enabler for an aspect composition, but the realization of it yet to be widely explore. The delay mainly caused by an abstract nature of crosscutting concerns. Moreover, the unclear representation of the crosscutting concerns in existing programs, had led to an ambiguous situation (Joncheere et al., 2014). There are considerable amount of researches were conducted in an effort to suggest the way to handle those ambiguous situation especially at the source code level (Kiczales et al., 1997; Marin, Deursen and Moonen, 2007; Marin, Deursen and Moonen, 2004).

Refactoring effort at source code level seems to root a number of challenges. For instance, it has creates an obstacle to locate and filter concerns at an extensive number of Line of Code (LOC). Hence, source code level refactoring had caused the precision and recall issues (Mens, Kellens, and Krinke, 2008). In fact, the previously proposed techniques (Marin et al., 2004), have highlighted on the issue of missing out crosscutting concerns while performing refactoring. Surely, the source code level refactoring viewed to be less efficient in improving the applications modularity. Indeed, legacy applications which contains huge number of LOC definitely create more problems while attempt to refactor.

As an alternative, researchers had explored on the other software artefacts for aspect identification and aspect isolation purpose. One of the most current discussions on aspect isolation was on conceptual models to discover crosscutting concerns as early as possible. Among those models, Early Aspect (EA) is one of the most widely used approaches. In fact, it had supported in the aspect candidate's discovery at the requirement specification via viewpoint strategy (Rashid, Sawyer, Moreira, and Araujo, 2002).

Recent development on EA, have heightened the needs for suitable techniques to identify and isolate crosscutting concerns comprehensively. In fact, there are number of studies were conducted on the concern identification and concern isolation at requirement specification (Abebe and Yoo, 2014; Antonelli, Rossi, and Sampaio, 2010; Joncheere et al., 2014). However, far too little attention has been paid to EA in the context of refactoring. Generally, most of the discussions were focused on the new software development instead of

restructuring back the existing applications (Syed Ali and Mohd Kasirun,2008; Antonelli, Rossi, and Sampaio, 2010; Rashid et al., 2002). After all, the focal point of attention was on crosscutting concern's identification and isolation process instead of aspect formation process. This creates a gap in the early aspect formation process while attempt to refactor. Additionally, number of findings from the current studies were indicated that the flow of aspect candidates from requirement specification was poorly mapped into the next level of software artefacts (Abebe and Yoo, 2014). Again, this definitely creates an ambiguous situation in handling crosscutting concerns at requirement level. Those findings had suggested that refactoring effort at requirement level should not be stopped at identification and isolation stage. Indeed, it should go further into aspect formation stage. Correspondingly, it ensures the comprehensive and cost effective flow of aspect candidates to the next software artefacts.

Thus, the existing requirement-oriented aspect refactoring process acquires few refinement in terms of crosscutting concern identification, isolation and aspect formation. Indeed, a clear flow on aspect candidates among the distinctive software artefacts would ensure smooth refactoring processes. So, this research investigates the requirement-oriented aspect refactoring's effort in the paradigm of concern handler at requirement level and crosscutting concern mapping.

The proposed Requirement-Oriented Aspect Refactoring (ReqOAR) conceptual framework strives to handle the crosscutting concerns through an enhancement on existing EA processes. For instance, concern identification process enhanced using the Crosscutting Concern Domain Library Listing (CCDLL). It was believed that crosscutting concerns can be handled effectively using domain analysis to facilitate identification of functional concern. Obviously, domain knowledge have ability to reveal more uniformly distributed concerns through the domain specific keywords (Li, 2009). Especially, at the requirement specifications, the existence of aspect candidate was likely to be uncertain due to abstract nature of it. In that case, domain knowledge work as early detector for crosscutting concerns (Baniassad,Clements,Araujo, Moreira,Rashid and Tekinerdogan, 2006).

As for the concern isolation process, the Concern Associated Terms Glossary (CATG) was proposed. CATG able to enhance the way to locate the crosscutting places. Meanwhile, Aspect Lexicon Guideline (ALG) proposed to emphasize on the aspect formation. ALG endeavour to give earlier suggestion on aspect candidate elements by looking at the verb form of the requirement statements. Apart from that, the Mapping Workflow proposed with the intention to facilitate the aspect candidate flow from one software artefacts to another.

This thesis targets to discuss on the related work on aspect refactoring at requirement specification as well as to propose a conceptual framework for that purpose. Identically, the proposed conceptual framework believed to serve the aspect refactoring as a whole. Moreover, the thesis also stresses out the empirical study which was conducted to know the strength of the proposed conceptual framework.

1.3 Problem Statement

Generally, requirement specification consists of stakeholder's needs. Use case scenario, viewpoints, main flow activities, alternative flow activities are among commonly used requirement specifications artefacts. So, requirement specification is known to be a textual based representation of user requests (Broden, 2011). EA model caters the aspect candidate's identification and isolation for new software development at those textual based representation. Indeed, it has suggests the way to capture stakeholder interest using viewpoint strategy. Viewpoint strategy is a kind of aspect-oriented requirement specification which based on the view of stakeholders. Further, it has striven the way to write requirement specification with focus on stakeholder interest for upcoming development. In contrast, requirement-oriented aspect refactoring activities does not involve any new user requirement capturing processes. So, the existing EA technique such as viewpoint strategy seems not appropriate to be adopted during refactoring activities (Rago, Marcos and Diaz-Pace, 2011; Yu et al., 2009). Therefore, it creates an unclear situation in handling crosscutting concerns while attempt to transform existing application to be AOA compatible. In particular, those unclear situations have fostered to the comprehensive EA formation and aspect candidate flow during the requirement-oriented aspect refactoring activities. But, this yet to be compromised although there are number of concern isolation techniques have been proposed at various software artefacts (Huang, Lu and Yang, 2010; Marin et al., 2007; McFadden and Mitropoulos, 2012). Most of the proposed solutions lead to the imprecise identification of functional concerns, incomplete crosscutting concern isolation and uncertain mapping of the concern flow from requirement to the next software specifications. These issues can be classified into two major categories; concern handlers at requirement level and aspect candidate flow. Chapter 4 explains further on those two major categories in details.

1.3.1 Ambiguity in Concern Handler

A fundamental theory of EA model was introduced by Awais Rashid with the focus on new software development (Rashid et al., 2002). However, several practical questions raised when endeavour to adopt the EA model for refactoring purpose. For instance, there are increasing beliefs (Glinz, 2007; Rashid, Moreira and Araujo, 2003; Yu et al., 2009) that crosscutting concerns and aspects are mainly from non-functional requirements. Meanwhile, on the refactoring part there is a contradictory finding about this. Whereby, crosscutting concern fulfils the both functional and non-functional stakeholder interests (Fox, 2007). Apart from that, the available evidence (Zhang and Su, 2010) had suggested that crosscutting concern exists in various places at requirement specification such as at system constraints levels, at high level requirements, at business rules and etc. Thus, it shows a distinctive view between EA and its adoptability towards refactoring effort. Particularly, those consensus views have affected the way of handling crosscutting concerns at requirement specification which lead to an ambiguous situation. Moreover it provides a serious impact on aspect candidate's accuracy which isolated at the requirement specifications. Thus, this ambivalent environment has acquires the need to restructure the current requirement-oriented aspect refactoring processes.

1.3.2 Uncertain Aspect Candidate Flow

It is become increasingly difficult to ignore that an aspect formation only can be completed when there is a clear flow of aspect candidates during refactoring process (Fox, 2011). In contrast, most of the current refactoring processes had paid wide attention at one of the software artefacts only. Therefore, the aspect candidates which formed at the earlier stage become unusable since it was suspended at that phase itself without move further into next software artefacts (Conejero, Hernandez, Jurado and Berg, 2010). This suspension viewed as a serious problem because the actual purpose of refactoring activity was to refactor aspect completely at the source code level. Obviously, this uncertainty has fostered the further revision on requirement-oriented aspect refactoring.

1.4 Research Objective

The main objective of this research is to develop a conceptual framework for requirement-oriented aspect refactoring. From the listed problem statement, it is anticipated that the proposed conceptual framework contains two important components. Such as unambiguous concern handler at requirement level and a complete aspect candidate's mapping workflow. In order to achieve those stated research purposes, two main objectives together with their respective sub-objectives stated as follows:

1. To propose concern handling processes and its associated techniques in eliminating the ambiguity situation. This objective involves the following sub-objectives:
   a) To propose the crosscutting concern listing which would enhance the ability of concern identification based on domain its belong to. The collection of crosscutting concern generated through systematic review of literature.
   b) To propose a concern associated terms glossary for the purpose of concern isolation at the requirement level. This would eliminate the ambivalent situation in identifying terminology which contain same meaning.
   c) To propose lexicon-oriented aspect formation technique which expected to suggest possible aspect candidate elements such as joinpoint, pointcut and advice at requirement specification itslef using postagger.

2. To propose the mapping workflow by specifying the flow of crosscutting concern from requirement specification to the next subsequent software artefacts.

1.5 Scope of Study

Scope statement means to limit the research elements into more manageable and appropriate context. Indeed, scope statement for this research is even needed more since requirement is an abstract thing and may vary from one requirement engineer to another requirement engineer (Aboutaleb and Monsuez, 2015). Thus, the scope of this research discussed in terms of refactoring effort, Aspect-Oriented Requirement Engineering (AORE) concept, aspect related definition and quality characteristics.

Scope 1: Refactoring Concept

Refactoring refers to the restructuring activity towards the existing system. This restructuring activity normally executed without making any changes to the external behaviour of an application. There are many ways refactoring can be applied. Such as creating abstract component, breaking the program into more manageable component and rearranging the code according to the standard programming language syntax (Thompson and Li, 2013). In accordance with this general definition of refactoring, this research highlights the aspect refactoring in the following facets:

i) Facet 1: create abstract component by identifying crosscutting components
ii) Facet 2: breaking the program by isolating those identified components
iii) Facet 3: rearranging by forming the requirement- oriented aspect candidates.

Scope 2: Aspect Oriented Requirement Engineering (AORE) Concept

AOA is a new software development paradigm which emphasizes on SoC concepts. Aspect-Oriented Requirement Engineering (AORE) is the representation of requirement engineering processes of those AOA. Since, this research focuses on AOA refactoring; the concepts of AOA and AORE in accordance to the existing system used in determining its generic processes. Notably, the focus into new software development would not be a part of this research.

Scope 3: Aspect related Definition

The various definition of EA and crosscutting concerns at requirement specifications were defined previously (Fox, 2007; Amirat, 2007). This research was limited to the following definition in the context of crosscutting concern and aspect:

i) Crosscutting concern is a requirement that under every possible translation from the problem space to the solution space is expressed in more than one modularization unit in a lower level of abstraction. This means the crosscutting concerns will have an equal impact on the functional and non-functional requirements.
ii) Aspect is the representation in the solution space of a requirement (problem space) that under every possible translation to the solution space is expressed in more than one modularization unit e. g. class, component, and function; depending on the underlying architectural framework of the solution space.

Scope 4: Quality characteristic that has been studied

Quality characteristic which is studied during this research is limited to aspect candidate's accuracy. Preciseness, recalling ability and F-measures are among those focused quality characteristics.

1.6 Research Associated Terms

There are few terminologies which commonly used in this research. Among those terminologies, almost all are similar to natural language idioms. Hence, it might confuse readers on the terminology usage. So, all the terminologies which are regularly practiced in this research listed and explained as follows:

a) Concern – referring to any interest, constraints, requirements that exist in an application
b) Crosscutting concern- referring to concern which cut across the core concern of an application
c) Aspect candidates- referring to the isolated crosscutting concern with specified advice, joinpoint and pointcut.
d) Advice, Joinpoint and Pointcut – referring to the components of an aspect
e) Stakeholder interest- referring to the concern which relate to the interest express by stakeholders. Generally stakeholder interest will differ from main functionality of an application.

1.7 Contribution of the Thesis

This research has been set out to determine and propose a conceptual framework for requirement-oriented aspect refactoring. As such, there are list of contribution expected to be produced at the end of this research. Figure 1.2 lists the research contributions in the context of four main research activities:

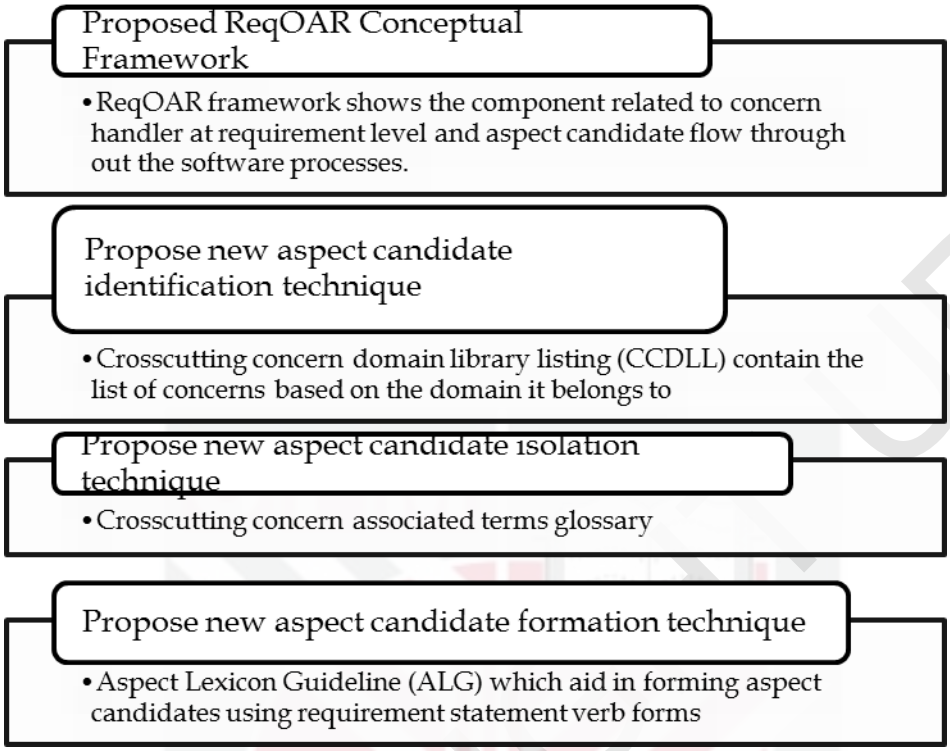

Figure 1.2. Conceptual View of Thesis Contribution

The Requirement-Oriented Aspect Refactoring (ReqOAR) conceptual framework is the main contribution of this research. It is expected to serve as reference for those who wish to refactor the existing programs to be compatible with the AOA especially on the text based artefacts. ReqOAR which consist of CCDLL, would instrument the crosscutting concern identification at the requirement specifications. It mainly focuses on the domain oriented crosscutting concern identification.

Besides that, ReqOAR able to serve in when isolating those crosscutting concerns from requirement specifications using Crosscutting Concern Associated Terms Glossary (CATG). Again this would be another contributing component of this research. Whereas, the Aspect Lexicon Guideline (ALG) expected to serve as guidance for requirement engineers in forming an aspect. In particular, ALG indicates the aspect candidate's elements by referring to the requirement statements. Moreover, the mapping between aspect idioms and language lexicon inside ALG would be the major contributor in determining aspect candidate's elements such as joinpoint, pointcut and advice.

1.8 Thesis Organization

This thesis is divided into six chapters. The first chapter is the introduction of the thesis. A brief background of the research is included in this chapter. Additionally, introduction chapter has merely covered the problem statement, scope of the thesis, objective that achieved by the thesis and etc.

The second chapter is literature review. It presents the Aspect-oriented Software Development (AOSD) backgrounds. In particular, the discussion focused on the crosscutting concern and the concept of SoC. Apart from that, the related works on aspect refactoring also covered in this chapter.

The third chapter is the research methodology. The chapter outlines the methods that been used in the process of creating ReqOAR conceptual framework. The explanation is further expanded into empirical validation strategy.

The fourth chapter provides explanation on the proposed ReqOAR conceptual framework. The chapter covers the flow of extracted crosscutting concern from requirement to design and further into source code.

The fifth chapter comprises about the empirical validation result and analysis. The experiment was explained in details and the results were discussed. The sixth chapter includes the conclusion and future work. It gives a general conclusion and suggests the research path that can be further explored in future.

REFERENCES

Amirat, A. (2007). Towards a requirements model for crosscutting concerns. *Information Technology Journal*, 6(3): 332–337.

Abebe, M. and Yoo, C. (2014). Trends , Opportunities and Challenges of Software Refactoring: A Systematic Literature Review, *International Journal of Software Engineering and Its Applications*, 8(6): 299–318.

Aboutaleb, H., and Monsuez, B. In *Towards a Holistic Definition of System Engineering Paradigm and Modeling Requirements*. Proceeding of the Twenty-third International Conference on Systems Engineering, Las Vegas, August, 2015. Springer International Publishing, 2015.

Amin, F. E., Mahmood, A. K., and Oxley. A. (2010). A Review on Aspect Oriented Implementation in Software Product Line, *Information Technology Journal*. 9(6):1262-1269.

Amirat, A., Meslati, D. and Laskri, M. T. In *An Aspect-Oriented Approach in Early Requirements Engineering,* Proceeding of the IEEE International Conference on.Computer Systems and Applications, Dubai, UAE, March. 8-8, 2006, IEEE Computer Society: 2006

Amalio, N., Kelsen, P., Ma, Q. and Glodt, C.(2010). Using VCL as an Aspect-Oriented Approach to Requirement Modelling. *Transaction on AOSD VII*: 151–199.

Anbalagan, P and Xie, T. In *Automated Inference of Pointcuts in Aspect-Oriented Refactoring*. Proceeding of the 29th International Conference on Software Engineering (ICSE'07), Minneapolis, Minnesota, May. 20-26, 2007. IEEE Computer Society: 2007

Antonelli, L., Rossi, G., and Sampaio, J.C. In *Early identification of crosscutting concerns in the domain model guided by states*, Proceeding of the Symposium on Applied Computing, Sierre, Switzerland, March. 22-26, 2010. ACM Press, 2010.

Antonelli, L. and Rossi, G. In *Deriving requirements specifications from the application domain language captured by Language Extended Lexico*, Proceeding of the Proceeding of the Workshop on Engenharia de Requisitos, Buenos Aires, Argentina, April. 24-27, 2012.

Araújo, J and Moreira, A. In *An Aspectual Use Case Driven Approach.* Proceeding of the Conference on Software Engineering and Databases, Alicante, Spain, Nov. 12-14, 2003, Thompson: 2003

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*, New York: ACM Press.

Baniassad, E., Clements, P.C., Araujo, J., Moreira, A., Rashid, A., Tekinerdogan, B. (2006). Discovering Early Aspects, *IEEE Software.* 23(1): 61–70.

Becker, M. and Diaz-herrerat, J. L. *Creating Domain Specific Libraries : a methodology and design guidelines*, Proceedings of the Third International Conference onSoftware Reuse: Advances in Software Reusability, Rio de Janeiro, Nov. 1-4, 1994.

Berg, K. V. D., Conejero, J. M. and Hernández, J. (2007). Analysis of Crosscutting in Early Software Development Phases Based on Traceability. Transaction on AOSD III : 73–104.

Bois, B. D., Gorp, P. V., Amsel, A., Eetvelde, N.V., Stenten, H., Demeyer, S. and Mens, T. *A Discussion of Refactoring in Research and Practice*, 2004.

Broden, L. (2011). *Requirements Traceability Recovery- A Study of Available Tools*. Unpublished master dissertation, Lund University, Sweden.

Breu, S. In *Extending Dynamic Aspect Mining with Static Information*. Proceeding of the Fifth IEEE International Workshop on Source Code Analysis and Manipulation (SCAM'05), Budapest, Hungary, Sept 30-Oct 1, 2005, IEEE Computer Society: 2005

Breu, S. *Aspect Mining Using Event Traces*, Proceeding of the Workshop Software-Reengineering, Bad Honnef, May 2004.

Budwell, C. C. and Mitropoulos, F. J. In *The SLAI Methodology: An Aspect-Oriented Requirement Identification Process*. Proceeding of the 2008 International Conference on Computer Science and Software Engineering, Hubei, China, Dec. 12-14, 2008, IEEE Computer Society: 2008.

Chavez, C Von Flach G., Lucena, C.J.P., In *A Theory of Aspects for Aspect-Oriented Software Development* , Proceeding of the Simposia Brazilero De Engenheria De Software, IOS Press: 2003

Chitchyan, R. (2013). Semantics-Based Composition for Textual Requirements. In A. Moreira, R. Chitchyan, J. Araújo, & A. Rashid, *Aspect-Oriented Requirements Engineering* (pp. 61–75). Springer Berlin Heidelberg.

Chitchyan, R., Pinto, M., Rashid, A., and Fuentes, L. (2007). COMPASS : Composition-Centric Mapping of Aspectual Requirements to Architecture. Transaction on AOSD IV: 3–53.

Clemente, P. J., Hernández, J., Conejero, J. M., & Ortiz, G. (2011). Managing crosscutting concerns in component based systems using a model driven development approach. *Journal of Systems and Software*, 84(6): 1032–1053.

Conejero, J.M., Hernandez, J., Moreira, A., and Araujo, J. In *Discovering Volatile and Aspectual Requirements Using a Crosscutting Pattern*. Proceeding of the 15th IEEE International Requirements Engineering Conference (RE 2007), New Delhi, India, Oct. 15-19, 2007, IEEE Computer Society: 2007

Conejero, J.M., Hernandez, J., Jurado, E. and Berg, K. (2010). Mining early aspects based on syntactical and dependency analyses. *Science of Computer Programming*, 75(11): 1113–1141.

Committee, J.T, *ISO/IEC 25010:2011.Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models.* https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en. (2011).

Eaddy, M., Aho, A., and Murphy, G. C. In *Identifying, Assigning, and Quantifying Crosscutting Concerns*. Proceedings of the First International Workshop on Assessment of Contemporary Modularization Techniques (ACoM '07), Minneapolis, MN, May. 20-26, 2007. IEEE Computer Society, 2007.

Elrad, T., Aksits, M., Kiczales, G., Lieberherr, K., and Ossher, H. (2001). Discussing aspects of AOP. *Communications of the ACM*, 44(10): 33–38.

Fox, J. (2007). A Taxonomy of Aspects in Terms of Crosscutting Concerns. In Brinksma, E.,Harel, D., Mader., A., Stevens, P and Wieringa, R. *Methods for Modelling Software Systems (MMOSS)*(pp. 1-27). Internationales Begegnungs.

Fox, J. In *An Exploration of Crosscutting Concerns in Software Requirements and Design*. Proceedings of The 3rd International Multi-Conference on Engineering and Technological Innovation (IMETI 2011), Orlando, Florida, July. 19-22, 2011, International Institute of Informatics and Systemics, 2011.

Glinz, M. In *On Non-Functional Requirements*. Proceeding of the 15th IEEE International Requirements Engineering Conference (RE 2007), New Delhi, India, Oct. 15-19, 2007, IEEE Computer Society, 2007.

Golbeck, R. M., Selby, P and Kiczales, G. (2010). In Late Binding of AspectJ Advice. *Objects, Models, Components, Patterns* (pp.173–191). Berlin: Springer Berlin Heidelberg.

Grundy, J. In *Aspect-oriented Requirements Engineering for Component-based Software Systems,* Proceeding of the IEEE International Symposium on Requirement Engineering, June. 11-11, 1999, IEEE Computer Society: 1999.

Gupta, N., Rawal, A., Narasimhan, V. L. and Shiwani, S. (2013). Accuracy , Sensitivity and Specificity Measurement of Various Classification Techniques on Healthcare Data, *IOSR Journal of Computer Engineering,* 11(5): 70–73.

Hamza, H.S. and Darwish, D. In *On the Discovery of Candidate Aspects in Software Requirements.* Proceeding of the Sixth International Conference on Information Technology: New Generations, Nevada, USA, April. 27-29, 2009, IEEE Computer Society: 2009

Huang, J., Lu, Y. and Yang, J. In *Aspect Mining Using Link Analysis*. Proceeding of the Fifth International Conference on Frontier of Computer Science and Technology, Changchun, Jilin Province, China, August. 18-22, 2010, IEEE Computer Society, 2010

Jacobson, I., and Ng, P.W. (2005). *Aspect-Oriented Software Development with Use Cases.* Addison-Wesley Professional.

Jedlitschka, A. and Pfahl, D, In *Reporting Guidelines for Controlled Experiments in Software Engineering*, Proceeding of the International Symposium on Empirical Software Engineering, Queensland, Australia, Nov. 17-18, 2005.IEEE Computer Society, 2005

Joncheere, N., Gunther, S., Straeten, V.D.S., Jonckers, V., (2014). Improving workflow modularity using a concern-specific layer on top of Unify. *Science of Computer Programming*, 87: 62–94.

Kassab, M., Ormandjieva, O. and Constantinides, C. In *Providing quality measurement for aspect-oriented software development*. Proceeding of the 12th Asia-Pacific Software Engineering Conference (APSEC'05),Taipei, Taiwan, Dec. 12-15, 2005, IEEE Computer Society: 2005

Kiczales, G. (2003). AspectJ(tm): Aspect-Oriented Programming in Java. *Objects, Components, Architectures, Services, and Applications for a Networked World* (pp. 1–1). Springer Berlin Heidelberg.

Kiczales, G. In *Context, Perspective, and Programs.* Proceeding of the 22nd ACM SIGPLAN Conference on Object-oriented Programming Systems and Applications Companion, Montreal, Canada, Oct. 21-25, 2007. ACM Press, 2007.

Kiczales, G., Lamping, J., Lopes, C. V., Maeda, C., Mendhekar, A., and Murphy, G. In *Open implementation design guidelines*. Proceedings of the 19th international conference on Software engineering (ICSE '97), Boston, MA, USA, May. 17-23, 1997. ACM: New York, 1997.

Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C.V., Loingtier, J.M., Irwin, J. In *Aspect-Oriented Programming,* Proceeding of the European Conference on Object-oriented Programming, Finland, June 9-13, 1997. Springer-Verlag: 1997.

Kiczales, G., Hugunin, J., Hilsdale, E., Kersten, M., Palm, J, Lopes, C.,Griswold, B., Isberg, W. (2003). *Aspect oriented programming*; Air Force Research Laboratory: Rome, New York, 2003.

Kitchenham, B., Pfleeger, S. L., Pickard, L. M., Jones, P. W., Hoaglin, D. C., El Emam, K. and Rosenberg, J. (2002). Preliminary guidelines for empirical research in software engineering. *IEEE Transactions on Software Engineering*, 28(8): 721–734.

Kitchenham, B. *Guidelines for performing Systematic Literature Reviews in Software Engineering;* Software Engineering Group, Keele University, Durham, UK, 2007.

Klein, J., Fleurey, F. and Jezequel, J.M. (2008). Weaving Multiple Aspects in Sequence Diagrams. *Transaction on AOSD III:* 167–199.

Lee, S.H., Cho, B.H. and Song, Y.J. In *A Study on Crosscutting Refactoring Using Progam Dependency Relation*. Proceeding of the 9th International Conference on Computer and Information Science, Yamagata, Japan , Aug. 18-20, 2010, IEEE Computer Society: 2010

Lencastre, M., Moreira, A., & Castro, J. (2013). Aspect Composition in Problem Frames. In Moreira, Chitchyan, Araujo and Rashid, *Aspect-Oriented Requirements Engineering* (pp. 121–141). Springer Berlin Heidelberg.

Li, G. (2009). *Identifying Crosscutting Concerns on Requirement Specification-A Case Study*, Unpublished masters dissertation, Queen's University, Canada.

Ligang, T. and Shenbo, Y. In *The Development and Application of User Domain Parts Library on Knowledge Based Engineering*. Proceeding of the Third International Conference on Measuring Technology and Mechatronics Automation, Shangshai, China, Jan. 6-11, 2011.

Madeyski, L., and Szala, L. (2007). Impact of aspect-oriented programming on software development efficiency and design quality: an empirical study. *IET Software Journal*. 1(5):180–187.

Marin, M., Deursen, A.V and Moonen, L. (2007). Identifying Crosscutting Concerns Using Fan-In Analysis. *ACM Transactions on Software Engineering and Methodology*, 17(1): 1–37.

Marin, M., Deursen, A.V and Moonen, L. In *Identifying Aspects using Fan-In Analysis,* Proceedings of the 11th Working Conference on Reverse Engineering, Delft, The Netherlands, Nov. 8-12, 2004. IEEE Computer Society,2004.

Marin, M., Moonen, L. and Deursen, A.V. In *An Approach to Aspect Refactoring Based on Crosscutting Concern Types*. Proceeding of the First International Workshop on the Modeling and Analysis of Concerns in Software, Missouri, USA, May. 16, 2005. ACM Press: 2005

Marin, M., Moonen, L. and Deursen, A.V. In *An Integrated Crosscutting Concern Migration Strategy and its Application to JHOTDRAW.* Proceeding of the Seventh IEEE International Working Conference on Source Code Analysis and Manipulation, Paris, France, Sept 30-Oct 1, 2007, IEEE Computer Society: 2007

Matthews, A. *Producing High-Quality Software with Aspect-Oriented Programming*; White Paper, Post Sharp, 2011.

Maxwell, J. (2013). *Qualitative Research Design An Interactive Approach.* SAGE Publications, Inc.

McFadden, R. R. and Mitropoulos, F. J. In *Aspect mining using model-based clustering*. Proceeding of the IEEE Southeastcon, March. 15-18, 2012, IEEE Computer Society: 2012

Mcheick, H., Mili, H., Sadou, S., and El-kharraz, A. In *A comparison of aspect oriented software development techniques for distributed applications,* Proceeding of the IADIS International Conference on Applied Computing, San Sebastian, Spain, Feb 25-28, 2006. IADIS Press: 2006

Mens, K., Kellens, A., and Krinke, J. In *Pitfalls in Aspect Mining*. Proceedings of the 15th Working Conference on Reverse Engineering, Antwerp, Belgium, Oct. 15-18, 2008, IEEE Computer Society, 2008.

Monga, M. *On Aspect-Oriented Approaches*. Paper presented at 1st European Interactive Workshop on Aspects in Software, Berlin, Germany. Sept 2004.

Monteiro, M.P. and Fernandes, J. M. In *Refactoring a Java code base to AspectJ: an illustrative example.* Proceeding of the 21st IEEE International Conference on Software Maintenance (ICSM'05), Los Alamitos, California, Sept. 26-29, 2005, IEEE Computer Society: 2005

Monteiro M.P and Fernandes, J.M. (2006). Towards a Catalogue of Refactorings and Code Smells for AspectJ. In **Rashid,** A, **Aksit**, M. *Transaction on Aspect Oriented Software Development I* (pp.214–258). Springer.

Moreira, A and Araújo, J. In *Handling unanticipated requirements change with aspects*, Proceeding of the Software Engineering and Knowledge Engineering Conference, Banff, Canada, June. 20-24, 2004.

Moreira, A., Araújo, J and Brito, I. In *Crosscutting Quality Attributes for Requirements Engineering*, Proceeding of the Software Engineering and Knowledge Engineering Conference, Ischia, Italy, July. 15–19, 2002, ACM Press: 2002.

Moreira, A., Rashid, A and Araújo, J. In *Multi-Dimensional Separation of Concerns in Requirements Engineering*. Proceeding of the 13th IEEE International Conference of

Requirement Engineering, Paris, France, August 29-September 2, 2005. IEEE Computer Society: 2005

Mortensen, M., Ghosh, S. and Bieman, J. M., In *Testing During Refactoring : Adding Aspects to Legacy Systems*, Proceeding of the 17th International Symposium on Software Reliability Engineering, Raleigh, North Carolina, Nov 7-10, 2006, IEEE Computer Society: 2006

Mortensen, M., Ghosh, S., and Bieman, J. M. (2012). Aspect-Oriented Refactoring of Legacy Applications: An Evaluation, *IEEE Transaction on Software Engineering.* 38(1): 118–140.

Mussbacher, G., Amyot, D., Araújo, J. and Moreira, A. (2010). Requirements Modeling with the Aspect-oriented User Requirements Notation (AoURN): A Case Study. *Transaction on AOSD VII* : 23–68.

Mussbacher, G., Amyot, D., and Edward, K. In *On Modeling Interactions of Early Aspects with Goals*, Proceedings of the ICSE Workshop on Aspect-Oriented Requirements Engineering and Architecture Design (EA '09), Washington, DC, USA, IEEE Computer Society, 2009

Mussbacher, G., Amyot, D. and Weiss, M. (2007). Visualizing Early Aspects with Use Case Maps. *Transaction on AOSD* III:105–143.

Mussbacher, G., Amyot, D. and Whittle, J. (2013). Composing Goal and Scenario Models with the Aspect-Oriented User Requirements Notation Based on Syntax and Semantics. In Moreira, Araujo, Rashid and Chitchyan, *Aspect-Oriented Requirements Engineering* (pp. 77–99).

Netinant, P. In *Separation of concerns for Multithreads Object-Oriented Programming*, Proceeding of the International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Pathumtani, Thailand, May. 6-9, 2009. IEEE Computer Society, 2009

Niu, N. and Easterbrook, S. (2007). Analysis of Early Aspects in Requirements Goal Models : A Concept-Driven Approach. *Transaction on AOSD III*: 40–72.

Petticrew, M. and Roberts, H. (2006). *Systematic Reviews in the Social Sciences: A Practical Guide*. USA:Blackwell Publishing Ltd.

Ping, L. In *A Quantitative Approach to Software Maintainability Prediction*. Proceeding of the International Forum on Information Technology and Applications, Kunming, China, July, 16-18, 2010. IEEE Computer Society, 2010.

Przyblek, A. In *Separation of Crosscutting Concern at Design Level: An Extension of UML MetaModel.* Proceeding of the International Multiconference on Computer Science and Information Technology, Wisła, Poland,Oct. 20-22, 2008, IEEE Computer Society: 2008

Qin, J., Zhou, Y. and Chau, M., *Building Domain-Specific Web Collections for Scientific Digital Libraries : A Meta-Search Enhanced Focused Crawling Method,* Proceeding of the 2004 Joint ACM/IEEE Conference on Digital Libraries, Arizona, USA, June. 7-11, 2004

Qu, C., Zhang, L., Yu, Y., and Wang, S. (2008). In *Modeling of Equipment Maintenance Support System Domain Library and its Application in HLA Simulation System*. Proceeding of the International Conference on Electrical, Electronic and Computer Engineering, ICEEC '04. Sept. 5-7, 2004.

Rago, A., Marcos, C. and Diaz-Pace, J. A. (2011). Uncovering quality-attribute concerns in use case specifications via early aspect mining. *Requirements Engineering*, 18(1):67–84.

Ramos, R. A., Castro, J., Araújo, J., Moreira, A., Alencar, F., & Penteado, R. In *Early Aspects Refactoring*. Proceeding of the Conferencia Iberoamericana de Software Engineering - CIbSE, February 11-15, 2008

Rashid, A., Moreira, A. and Araujo, J. In *Modularisation and Composition of Aspectual Requirements*, Proceedings of the 2nd international conference on Aspect-oriented software development, Boston, USA, ACM Press, 2003.

Rashid, A., Thomas, C., Greenwood, P., Chitchyan, R., Meunier, R., Coelho,R., Sudholt, M. and Joosen, W. (2010). Aspect-Oriented Software Development in Practice: Tales from AOSD-Europe, *Computer*, 43(2): 19–26.

Rashid, A., Sawyer, P., Moreira, A., and Araujo, J. In *Early Aspects: a Model for Aspect-Oriented Requirements Engineering*, Proceeding of the IEEE Join International Conference on Requirement Engineering, Essen, Germany, Sept. 9-13, 2002, IEEE Computer Society, 2002.

Razzaq, A. and Abbasi, R. In *Automated Separation of Crosscutting Concerns : Earlier Automated Identification and Modularization of Cross-Cutting Features at Analysis Phase*. Proceeding of the 15th International Multitopic Conference, Islamabad, Pakistan, Dec. 13-15, 2012, IEEE Computer Society: 2012

Rebernak, D., Mernik, M., Henriques, P.R., Cruz, D. and Varanda Pereira, M.J., In Specifying Languages Using Aspect-Oriented Approach: AspectLISA, Cavtat, Croatia, June. 19-22, 2006, IEEE Computer Society: 2006.

Salleh, N. *A Systematic Review of Pair Programming Research,* Proceeding og the New Zealand Computer Science Research Student Conference, Christchurch, New Zealand, April 2008.

Salleh, N., Mendes, E. and Grundy, J. (2010). Empirical Studies of Pair Programming for CS / SE Teaching in Higher Education : A Systematic Literature Review. *IEEE Transaction on Software Engineering.* 37(4): 509-525.

Sampaio, A., Rashid, A., Chitchyan, R. and Rayson, P. (2007). EA-Miner : Towards Automation in Aspect-Oriented Requirements Engineering. *Transaction on AOSD III:* 4–39.

Sampaio, A., Chitchyan, R., Rashid, A. and Rayson, P. In *EA-Miner : a Tool for Automating Aspect-Oriented Requirements Identification*, Proceeding of the 20th IEEE/ACM International Conference on Automated Software Engineering, California, USA, November. 7-11, 2005, IEEE Computer Society: 2005

Singh, Y. and Goel, B. (2007). A Step Towards Software Preventive Maintenance ACM SIGSOFT Software Engineering Notes, *32*(4): 1–5.

Smucker, M. D., Allan, J. and Carterette, B. In *A Comparison of Statistical Significance Tests for Information Retrieval Evaluation.* Proceeding of the ACM Sixteenth Conference on Information and Knowledge Management, Lisboa, Portugal, Nov. 6-8, 2007, ACM Press: 2007

Solberg, A., Simmonds, D., Reddy, R., Ghosh, S and France, R. In *Using Aspect Oriented Techniques to Support Separation of Concerns in Model Driven Development*. Proceeding of the 29th Annual International Computer Software and Applications Conference, Edinburgh, Scotland, July. 26-28, 2005, IEEE Computer Society: 2005.

Sotos, A.E.C., Vanhoof, S., Noortgate, W.V.d. and Onghena, P. (2007). Students' misconceptions of statistical inference: A review of the empirical evidence from research on statistics education. *Educational Research Review*, 2(2): 98–113.

Silva, L. F. and Prado Leite, J. C. S. (2013). Aspect-Oriented Goal Modeling and Composition with AOV-Graph. In Moreira, Chitchyan, Araujo and Rashid, *Aspect-Oriented Requirements Engineering*. Springer Berlin Heidelberg.

Syed Ali, B. and Mohd Kasirun, Z. (2008). Crosscutting concern identification at requirements level, *Malaysian Journal of Computer Science*. 21(2): 78–87.

Sousa, G., Soares, S., Borba, P. and Castro, J. In *Separation of Crosscutting Concerns from Requirements to Design : Adapting an Use Case Driven Approach*. Proceeding of the Early Aspect Workshop at AOSD: 2004

Steimann, F. In *The Paradoxical Success of Aspect-Oriented Programming*, Proceeding of the 21st ACM SIGPLAN Symposium on Object-oriented Programming Systems, Languages, and Applications, Oregan, USA, Oct. 22-26, 2006. ACM Press, 2006.

Sánchez, P., Fuentes, L., Jackson, A., and Clarke, S. (2008). Aspects at the Right Time. Transaction on AOSD IV: 54–113.

Thompson, S. and Li, H. (2013). Refactoring tools for functional languages. *Journal of Functional Programming*, *23*(03): 293–350.

Tonella, P. and Ceccato, M. In *Aspect Mining through the Formal Concept Analysis of Execution Traces*. Proceedings of the 11th Working Conference on Reverse Engineering (WCRE'04), Delft, Netherlands, Nov. 8-12, 2004, IEEE Computer Society: 2004

Vidal, S. A. (2013). A Catalog of Aspect Refactorings for Spring / AOP. *Journal of Uviversal Computer Science*. 19(1): 157–182.

Wehrmeister, M. A., Pereira, C. E., and Rammig, F. J. (2013). Aspect-Oriented Model-Driven Engineering for Embedded Systems Applied to Automation Systems, *IEEE Transaction on Industrial Informatics. 9*(4): 2373–2386.

Wohlin, C., Runeson, P., Host, M., Ohlsson, M.C., Regnell, B. and Wesslen, A. (2012). *Experimentation in Software Engineering: An Introduction*. London: Kluwer Academic Publishers.

Yu, L., Ramaswamy,S., and Vaidyanathan,A. In *Understanding the Effects of Code Clones on Modularity in Software Systems,* Proceeding of the 19th Asia-Pacific Software Engineering Conference, August 2012. IEEE: 105–111. 2012

Yang, S, Zhong, P.X. In *Recovering Crosscutting Concern from Legacy Software Based on Use-Cases Driven Formal Concept Analysis.* Proceeding of the International Conference on

Computational Intelligence and Software Engineering, Wuhan, China, Dec. 10-12, 2010, IEEE Computer Society: 2010

Yu, Y., Niu, N., Baixauli, G., Mylopoulos, J., Easterbrook, S. and Sampaio, J.C. (2009). Requirements Engineering and Aspects, In Lyytinen, K.,Loucopoulos, Pericles, Mylopoulos, J., Robinson, B. *Design Requirements Engineering: A Ten-Year Perspective* (pp.432–452). Springer Berlin Heidelberg.

Yvonne, C., Hans-Arno, J., & Mario, S. (2006). Editorial: AOP Systems , Software and Middleware. In **Rashid,** A, **Aksit**, M. *Transaction on Aspect Oriented Software Development II*, (pp.66–68).Springer.

Zhang, C. and Jacobsen, H.A. In *Efficiently Mining Crosscutting Concerns*, Proceeding of the 6th International Conference on Aspect-Oriented Software Development, Vancouver, British Columbia, March. 12-16, 2007, ACM Press: 2007

Zhang, P., & Su, Y. In *Understanding the aspects from various perspectives in aspects-oriented software reverse engineering*. Proceeding of the International Conference on Computer Application and System Modeling (ICCASM 2010), North University of China, Taiyuan, China, Oct. 22-24, 2010. IEEE Computer Society, 2010.

Zhu, W., Zeng, N. and Wang, N., In *Sensitivity , Specificity , Accuracy , Associated Confidence Interval and ROC Analysis with Practical SAS ® Implementations*, Proceeding of the NESUG 2010, Baltimore, Maryland, Nov. 14-17, 2010.