

Kleinner Silva Farias de Oliveira

**Empirical Evaluation of Effort on
Composing Design Models**

TESE DE DOUTORADO

Thesis presented to the Programa de Pós-Graduação em
Informática of the Departamento de Informática, PUC-Rio
as partial fulfillment of the requirements for the degree of
Doutor em Informática

Advisor: Prof. Alessandro Garcia
Co-Advisor: Prof. Carlos José Pereira de Lucena

Rio de Janeiro
March 2012



Kleinner Silva Farias de Oliveira

Empirical Evaluation of Effort on Composing Design Models

**Thesis presented to the Programa de Pós-Graduação em
Informática, of the Departamento de Informática do Centro
Técnico Científico da PUC-Rio, as partial fulfillment of the
requirements for the degree of Doutor.**

Prof. Alessandro Garcia

Advisor

Departamento de Informática – PUC-Rio

Prof. Carlos José Pereira de Lucena

Co-Advisor

Departamento de Informática – PUC-Rio

Prof. Arndt von Staa

Departamento de Informática – PUC-Rio

Prof^a. Karin Koogan Breitman

Departamento de Informática – PUC-Rio

Prof. Toacy Cavalcante de Oliveira

Universidade Federal do Rio de Janeiro – UFRJ

Prof^a. Christina von Flach Garcia Chavez

Universidade Federal da Bahia – UFBA

Prof. José Eugenio Leal

Coordinator of the Centro Técnico Científico da PUC-Rio

Rio de Janeiro, 21 March 2012

Kleinner Silva Farias de Oliveira

He received his BSc in Computer Science from the Federal University of Alagoas in 2006 and in Information Technology from the Federal Institute of Alagoas in 2006. He received his MSc in Computer Science from the Pontifical Catholic University of Rio Grande do Sul in 2008.

Bibliographic data

Oliveira, Kleinner Silva Farias de

Empirical evaluation of effort on composing design models / Kleinner Silva Farias de Oliveira ; supervisor: Alessandro Garcia ; co-supervisor: Carlos José Pereira de Lucena. – 2012.

282f. : il. (color.) ; 30 cm

Tese (doutorado)–Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2012.

Inclui bibliografia

1. Informática – Teses. 2. Estudos empíricos. 3. Design de software. 4. Modelagem de software. 5. Métricos de software. I. Garcia, Alessandro. II. Lucena, Carlos José Pereira de. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

To my family

Acknowledgments

I am honored and grateful to have counted with excellent professionals along the way of this work. Especially, I would like to thank my supervisor, Prof. Dr. Alessandro Garcia, who provided constant encouragement, guidance, and freedom to develop this thesis. His competent and constructive criticism was essential to my success, my sincere admiration to him.

I am very grateful to have Prof. Dr. Carlos Lucena as one of my supervisors. I would like to thank him for all support on my research path by guiding me with his extensive knowledge, providing advice, and encouragement in the early stages.

I am indebted to Jon Whittle (Lancaster University, UK) for his in-depth reviews of many papers, and numerous interesting discussions and suggestions about the critical points throughout this work, lending a sense of reality to the research being conducted. I received so much from him to enlarge my vision on software engineering.

I could not forget to thank Dr. Toacy Oliveira, who trusted me fully from the beginning of my research career. Thanks to his mentoring, friendship, and all frequent discussions that help me to guide my professional life.

My gratefulness also goes to all my colleagues and professors from the Computer Science Department at PUC-Rio, especially to my friends from the Software Engineering Laboratory (LES) who gave me opportunities to grow as professional putting my ideas into action. It has been a privilege working on that stimulating environment.

During the development of this work, I was lucky to have collaborated with a number of research colleagues who contributed to this thesis in different ways and who allowed me to collaborate with their works as well. It was a pleasure to work with all of them on a number of papers.

I am also thankful to the members of my examination committee, who has generously contributed their time and expertise.

Thanks to my friends (whose names I did not quote to avoid being unfair to any of them right now) for giving invaluable tips, and giving lots of very useful advice. Especially, I am deeply grateful to my fiancée Carla Pedroso for putting up with me, and giving me so much that I could not even itemize them. Your love and understanding were instrumental in this journey. I love you so much.

Finally, I would like to thank my family for the constant support and belief in me in every conceivable way. This thesis is dedicated to my grandmother, Valdenira (in Memoriam), and my parents, Sandra and Carlos, my sisters, Kelyne and Klyvia, and my brother, Kleberson, with all my love.

I would like to express here my gratitude to CAPES/CNPq for the financial support of this doctoral study.

Resumo

Oliveira, Kleinner Silva Farias; Garcia, Alessandro Fabricio, Advisor; Lucena, Carlos José Pereira de, Co-Advisor. **Avaliação Empírica de Esforço em Composição de Modelos de Projeto**. Rio de Janeiro, 2012. 282p. Tese de Doutorado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Composição de modelos desempenha um papel fundamental em muitas atividades de engenharia de software como, por exemplo, evolução e reconciliação de modelos conflitantes desenvolvido em paralelo por diferentes times de desenvolvimento. Porém, os desenvolvedores têm dificuldades de realizar análises de custos e benefícios, bem como entender o real esforço de composição. Sendo assim, eles são deixados sem qualquer conhecimento prático sobre quanto é investido; além das estimativas de evangelistas que frequentemente divergem. Se o esforço de composição é alto, então os potenciais benefícios tais como aumento de produtividade podem ser comprometidos. Esta incapacidade de avaliar esforço de composição é motivada por três problemas: (i) as abordagens de avaliação atuais são inadequadas para mensurar os conceitos encontrados em composição, por exemplo, esforço e conflito; (ii) pesquisadores não sabem quais fatores podem influenciar o esforço de composição na prática. Exemplos de tais fatores seriam linguagem de modelagem e técnicas de composição que são responsáveis para manipular os modelos; (iii) a falta de conhecimento sobre como tais fatores desconhecidos afetam o esforço de composição. Esta tese, portanto, apresenta uma abordagem de avaliação de esforço de composição de modelos derivada de um conjunto de estudos experimentais. As principais contribuições são: (i) um modelo de qualidade para auxiliar a avaliação de esforço em composição de modelos; (ii) conhecimento prático sobre o esforço de composição e o impacto de fatores que afetam tal esforço; e (iii) diretivas sobre como avaliar esforço de composição, minimizar a propensão a erros, e reduzir os efeitos negativos dos fatores na prática de composição de modelos.

Palavras-chave

Composição de modelos, esforço de desenvolvimento, estudos empíricos.

Abstract

Oliveira, Kleinner Silva Farias; Garcia, Alessandro Fabricio, Advisor; Lucena, Carlos José Pereira de, Co-Advisor. **Empirical Evaluation of Effort on Composing Design Models**. Rio de Janeiro, 2012. 282p. DSc Thesis – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Model composition plays a central role in many software engineering activities such as evolving models to add new features and reconciling conflicting design models developed in parallel by different development teams. As model composition is usually an error-prone and effort-consuming task, its potential benefits, such as gains in productivity can be compromised. However, there is no empirical knowledge nowadays about the effort required to compose design models. Only feedbacks of model composition evangelists are available, and they often diverge. Consequently, developers are unable to conduct any cost-effectiveness analysis as well as identify, predict, or reduce composition effort. The inability of evaluating composition effort is due to three key problems. First, the current evaluation frameworks do not consider fundamental concepts in model composition such as conflicts and inconsistencies. Second, researchers and developers do not know what factors can influence the composition effort in practice. Third, practical knowledge about how such influential factors may affect the developers' effort is severely lacking. In this context, the contributions of this thesis are threefold: (i) a quality model for supporting the evaluation of model composition effort, (ii) practical knowledge, derived from a family of quantitative and qualitative empirical studies, about model composition effort and its influential factors, and (iii) insight about how to evaluate model composition efforts and tame the side effects of such influential factors.

Keywords

Model composition, development effort, empirical studies.

Table of Contents

1 Introduction	18
1.1. Problem Statement	20
1.2. Limitations of Related Work	22
1.3. Study Methodology	26
1.4. Thesis Contributions	28
1.5. Thesis Outline	30
2 Background and Related Work	35
2.1. Purpose of Using Model Composition	35
2.2. Properties of the Design Modeling Languages	37
2.3. Purpose of Using Design Models	39
2.4. Model Composition Techniques	40
2.4.1. Traditional Composition Algorithms	41
2.4.2. IBM Rational Software Architect	43
2.4.3. Epsilon	44
2.4.4. Limitations of Related Work on Model Composition Techniques	46
2.5. Design Modeling Languages	48
2.5.1. Unified Modeling Language	48
2.5.2. Aspect-Oriented Modeling	49
2.5.3. Limitations of Related Work on Design Modeling Languages	53
2.6. Design Characteristics	55
2.6.1. Model Stability	55
2.6.2. Limitations of Related Work on Design Characteristics	57
2.7. Concluding Remarks	58
3 A Quality Model for Model Composition Effort	60
3.1. Motivation	61
3.2. Limitations of Related Work	63
3.3. A Quality Model for Model Composition Effort	66
3.3.1. Model Composition Effort and Change Categories	66

3.3.2. Composition Conflicts and Inconsistencies	68
3.3.3. Abstract Syntax of the Quality Model	70
3.3.4. Quality Notions	75
3.3.5. Levels of the Quality Model	80
3.4. Concluding Remarks	85
4 Effort on the Application of Composition Techniques	87
4.1. Effects of Composition Techniques on the Composition Effort	87
4.1.1. Experiment Planning	89
4.1.2. Experimental Results	99
4.1.3. Limitations of Related Work	107
4.1.4. Threats to Validity	108
4.1.5. Concluding Remarks of the First Study	110
4.2. Analyzing the Effort of Composing Design Models of Large-Scale Software	111
4.2.1. Background	113
4.2.2. Composition Scenario	114
4.2.3. Study Methodology	122
4.2.4. Study Results	133
4.2.5. Limitations of Related Work	149
4.2.6. Concluding Remarks of the Second Study	153
5 Effort on the Detection of Inconsistency	155
5.1. Background	157
5.1.1. Aspect-Oriented Modeling	157
5.1.2. Model Inconsistency	159
5.1.3. Inconsistency Detection Effort	162
5.2. Study Methodology	163
5.2.1. Experiment Definition	163
5.2.2. Hypothesis Formulation	164
5.2.3. Selection of Subjects	167
5.2.4. Experiment Design	167
5.2.5. Operation and Material	168
5.2.6. Variables and Quantification Method	170

5.2.7. Operation	171
5.2.8. Analysis Procedures	171
5.2.9. Qualitative Data	172
5.3. Experimental Results	172
5.3.1. RQ4.1: Detection Rate in AO and OO models	173
5.3.2. RQ4.2: Detection Effort in AO and OO models	174
5.3.3. RQ4.3: Misinterpretation Rate in AO and OO models	176
5.4. Discussion	177
5.5. Limitations of Related Work	180
5.6. Threats to Validity	181
5.7. Concluding Remarks	182
 6 Effort on the Resolution of Inconsistency	 184
6.1. Effect of Model Stability on Inconsistency Resolution	184
6.1.1. Background	188
6.1.2. Study Methodology	193
6.1.3. Results	205
6.1.4. Limitations of Related Work	220
6.1.5. Threats to Validity	223
6.1.6. Concluding Remarks	229
6.2. Impact of Design Language on Inconsistency Resolution Effort	230
6.2.1. Aspect-Oriented Modeling for Architectural Models	231
6.2.2. Study Methodology	233
6.2.3. Composition Effort Analysis	243
6.2.4. Limitations of Related Work	253
6.2.5. Threats to Validity	255
6.2.6. Conclusions and Future Work	255
 7 Conclusions	 257
7.1. Summary	257
7.2. Contributions	259
7.3. Future Works	265
 8 References	 268

List of Figures

Figure 1: Illustrative example	42
Figure 2: An illustrative example of AO models used in our study	55
Figure 3: Overview of model composition effort: an equation	67
Figure 4: Illustrative example	68
Figure 5: Abstract syntax of the quality model for model composition (based on (Lange, 2007))	70
Figure 6: The purposed quality model (based on (Lange, 2007a))	85
Figure 7: The Experimental process	97
Figure 8: The correctness of the output composed model	104
Figure 9: A real-world collaborative model composition leading to two critical overlapping points	116
Figure 10: A real-world use scenario of model composition (A). The change descriptions performed by the developers (B).	118
Figure 11: The Base Model (A) and the Intended model (B)	120
Figure 12: The model versions created by Peter (P2) (above) and Steve (S3) (below).	121
Figure 13: The model versions created by Bill (B4).	122
Figure 14: The flow of activities during the studies	129
Figure 15: Histogram of the application effort measures	136
Figure 16: An illustrative example of aspect-oriented models used	159
Figure 17: Example of composition of the Mobile Media product line	191
Figure 18: The intended and composed model produced following the union heuristic	191
Figure 19: Box-plot of inconsistencies	207
Figure 20: Box-plot of resolution effort in relation to the intended model	214 232
Figure 21: AOM language for architectural models	238
Figure 22: The input models: the AO base and AO delta model	240
Figure 23: Output AO models produced by override and merge	

algorithms	241
Figure 24: AO intended model (from Figure 22) and AO output model produced following the union heuristic	242
Figure 25: Inconsistency rate produced by the override algorithm	244
Figure 26: Inconsistency rate produced by the merge algorithm	245
Figure 27: Inconsistency resolution effort to recover the output model produced by override algorithm	250
Figure 28: Effort to recover the output model produced by merge algorithm	251

List of Tables

Table 1: List of direct and indirect publications	34
Table 2: Metrics of semantic inconsistencies (Farias et al., 2008)	69
Table 3: Definition of chapters where quality notions are investigated	79
Table 4: Characteristics of design models	81
Table 5: Metrics for class	83
Table 6: Metrics for interface	84
Table 7: Metrics for components	84
Table 8: Tested hypotheses	93
Table 9: The tasks of the evolution scenarios	94
Table 10: Descriptive statistic for the composition effort	100
Table 11: Wilcoxon test results for application and detection effort	101
Table 12: Wilcoxon test results for the resolution and general effort	102
Table 13: Statistical test for the Friedman Test	102
Table 14: Descriptive statistic for the inconsistency rate	105
Table 15: McNemar test results for correctness	105
Table 16: Friedman test result for inconsistency rate	106
Table 17: Wilcoxon test results for the correctness	106
Table 18: Metrics used	124
Table 19: The collected measures of the case studies	124
Table 20: Descriptive Statistics: Subjects' Background	126
Table 21: Descriptive statistics for application effort	134
Table 22: Descriptive statistics for detection effort	138
Table 23: Descriptive statistics for resolution effort	141
Table 24: Measures of the diagram used in the study	169
Table 25: Descriptive statistics	173
Table 26: Hypotheses testing	174
Table 27: Metrics used	189
Table 28: The inconsistencies used in our case study	193

Table 29: Descriptions of the evolution scenarios	198
Table 30: Descriptive statistics of the inconsistency rate	206
Table 31: Mann-whitney test and Spearman's correlation analysis	208
Table 32: Descriptive statistics of the resolution effort	213
Table 33: Mann-whitney test and Spearman's correlation analysis	215

List of Acronyms and Abbreviations

IBM – International Business Machine
UML – Unified Modeling Language
AO – Aspect-Oriented
AOM – Aspect-Oriented Modeling
CBO – Coupling Between Object Classes
EMF – Eclipse Modeling Framework
GQM – Goal Question Metric
IBM – International Business Machine
IDE – Integrated Development Environment
LCOM – Lack of Cohesion in Methods
MDD – Model Driven Development
MVC – Model View Controller
OCL – Object Constraint Language
OMG – Object Management Group
OO – Object-Oriented
RQ – Research Question
SPL – Software Product Line
UML – Unified Modeling Language
ECL – Epsilon Comparison Language
EML – Epsilon Merge Language

Believe in your dreams

Kleinner Farias