

University of Groningen

Understanding and Supporting Software Architectural Decisions

Tofan, Dan

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version

Publisher's PDF, also known as Version of record

Publication date:

2015

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Tofan, D. (2015). Understanding and Supporting Software Architectural Decisions: for Reducing Architectural Knowledge Vaporization [Groningen]: University of Groningen

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.



university of
 groningen

Understanding and Supporting Software Architectural Decisions

For Reducing Architectural Knowledge Vaporization

PhD thesis

to obtain the degree of PhD at the
University of Groningen
on the authority of the
Rector Magnificus Prof. E. Sterken
and in accordance with
the decision by the College of Deans.

This thesis will be defended in public on

Friday 20 November 2015 at 12.45

by

Dan Constantin Tofan

born on 17 May 1981
in Iași, Romania

Supervisor:

Prof. P. Avgeriou

Co-supervisor:

Dr. M. Galster

Assessment committee:

Prof. P. Lago

Prof. I. Crnkovic

Prof. M. Aiello

Samenvatting

De architectuur van software systemen wordt bepaald door de architectuur beslissingen. Hierin worden onderwerpen als frameworks, patterns, programmeer talen behandeld, of manieren om het systeem op te delen. Deze beslissingen en hun rationale zijn een belangrijk onderdeel van de architectuur kennis van een software systeem. Architectuur kennis van software systemen kan verloren gaan. Een architect kan de redenen van een beslissing vergeten, een andere baan krijgen, of de documentatie van beslissingen uitstellen. Het verdwijnen van architectuur kennis heeft enorme consequenties. Het kan zijn dat de initieel beoogde architectuur ideeën niet meer kunnen worden nageleefd waardoor uitbreidingen duur worden en het moeilijk is om de consistentie tussen beslissingen te bewaren. Het hoofddoel van dit onderzoek is het verminderen van het verlies van deze architectuur kennis. Dit wordt gedaan door deze beslissingen en hun rationale beter te documenteren. De bijdrage van dit onderzoek bestaat uit drie fasen: het begrijpen van het huidige onderzoek en de praktijk, exploreren van nieuwe ideeën en het aandragen van een concrete aanpak om het verlies van architectuurkennis tegen te gaan. Als derde bijdrage hebben we een gebruiksvriendelijke open-source tool gemaakt voor het nemen en documenteren van beslissingen als groep of als individu. De bijdragen van dit proefschrift helpen mensen in de praktijk om minder architectuur kennis te verliezen. Daarnaast helpt het onderzoekers om de aard van architectuur kennis beter te begrijpen.

ISBN: 978-90-367-8270-8 (Printed version)

ISBN: 978-90-367-8269-2 (Electronic version)

Keywords: software architecture, architectural decisions, knowledge vaporization

The research presented in this thesis was performed in the Software Engineering and Architecture group, at the Johan Bernoulli Institute for Mathematics and Computing Science at the University of Groningen. The cover image is licensed from Getty Images.

Abstract

The architecture of a software system is the result of architectural decisions on various topics, such as frameworks, patterns, programming languages, or ways to decompose the software system. Such decisions and their rationales are a significant part of the architectural knowledge about a software system.

Architectural knowledge about a software system tends to vaporize. For example, architects might forget the rationales of decisions, change jobs, or postpone indefinitely documenting decisions to avoid disrupting their design flow. Architectural knowledge vaporization has major practical consequences, such as drifting away from the initially intended architecture, and expensive evolution, due to the substantial needed effort to understand previous decisions and to avoid conflicts with them.

The overall research problem addressed in this thesis is how to reduce architectural knowledge vaporization. The overall solution is to reduce architectural knowledge vaporization by documenting architectural decisions and their rationales.

The contributions of this thesis at solving this problem can be grouped in three phases: understanding the state of practice and research, exploring new ideas, and proposing concrete approaches to reduce architectural knowledge vaporization.

In the first phase (understanding), we investigated the state of practice in which architectural knowledge vaporization occurs, and the state of research that can help reduce architectural knowledge vaporization. To understand the state of practice, we conducted two surveys with practitioners. The first survey helps researchers understand the challenges for managing architectural knowledge in practice, and potential solutions to these challenges. The results of the first survey indicate that architectural knowledge vaporization is a major challenge in the industry, and that tool support is a potential solution. The second survey describes real-world architectural decisions, such as their characteristics, difficulties, and differences between good and bad architectural decisions. For example, we found out that most architectural decisions are group decisions.

To understand the state of research, we conducted a systematic mapping study on the last decade of research on architectural decisions. This study helped us understand existing work on reducing architectural knowledge vaporization and future promising research directions. For example, we identified a lack of research on group architectural decisions, despite the fact that most architectural decisions are group decisions. Furthermore, we identified very few open-source tools for architectural decisions.

In the second phase (exploring), we investigated using established approaches from the knowledge engineering field for reducing architectural knowledge vaporization. In particular, we conducted two surveys with students on using the Repertory Grid technique for documenting architectural decisions, to identify advantages and disadvantages of the technique. We found out that the main advantages are reducing architectural knowledge vaporization and reasoning support. The main disadvantages are the needed effort and lack of user-friendly tool support.

In the third phase (proposing), we made three contributions.

First, we contributed an approach based on the Repertory Grid technique for making and documenting individual architectural decisions. We did a survey with practitioners to identify advantages, disadvantages, and improvement opportunities of the approach. Advantages include reduction of architectural knowledge vaporization, and decision making support. Disadvantages include effort and insufficient tool support. Improvement opportunities include support for prioritizing concerns and for group decision making. To improve the approach, we did a controlled experiment with students to compare two concerns prioritization methods, and then we added the most suitable method to the approach.

Second, we contributed an extension of the approach for making and documenting group architectural decisions. We did a case study to identify benefits and potential improvements of the approach. Benefits include reduction of architectural knowledge vaporization, and increased consensus of the group. Furthermore, we did a controlled experiment with students to compare the approach against ad-hoc group decision making. Experiment results indicate that the proposed approach reduces architectural knowledge vaporization and increases consensus.

Third, we contributed with user friendly, open-source tool support for the two approaches for making and documenting individual and group architectural decisions.

Overall, the contributions of this thesis help practitioners reduce architectural knowledge vaporization. Furthermore, the contributions of this thesis help researchers understand various aspects of architectural decisions and architectural knowledge, so that researchers can propose approaches that satisfy the needs of practitioners.

Contents

Chapter 1

Introduction	1
1.1 Context	1
1.2 Problem statement	2
1.2.1 Research framework	4
1.3 Research questions	5
1.3.1 Iteration A - Understand	5
1.3.2 Iteration B - Explore	9
1.3.3 Iteration C - Propose	10
1.4 Research methods	11
1.5 Publications overview	13

Chapter 2

Architectural Knowledge Management in Practice	15
2.1 Introduction	16
2.2 Related Work	17
2.3 Research Method	18
2.3.1 Data Collection and Analysis Procedures	19
2.3.2 Organizations	20
2.3.3 Validity Threats	20
2.4 Challenges	21
2.4.1 Challenges in the public sector	22
2.4.2 Challenges in the private sector	24
2.5 Solutions	26
2.6 Discussion	29
2.7 Conclusions	31

Chapter 3

Architectural Decisions in Practice	33
3.1 Introduction	34
3.2 Related Work	35
3.3 Survey Design	37
3.3.1 Questionnaire Development and Evaluation	37

3.3.2	Data Collection	39
3.4	Results Analysis	40
3.4.1	Participants Background	40
3.4.2	RQ1 - Characteristics of Architectural Decisions	41
3.4.3	RQ2 - Difficulty of Decisions	43
3.4.4	RQ3 - Differences between Junior and Senior Architects	46
3.4.5	RQ4 - Differences between Good and Bad Decisions	49
3.5	Discussion	51
3.5.1	Validity Threats	53
3.6	Conclusions	53

Chapter 4

State of Research on Architectural Decisions 55

4.1	Introduction	57
4.2	Research Methodology	58
4.2.1	Research Directives	61
4.2.1.1	Protocol Definition	61
4.2.1.2	Generic Decision Literature Survey	62
4.2.1.3	Research Questions Definition	64
	Research Questions Derived from Software Architecture Literature	64
	Research Questions Derived from Generic Decision Literature	65
4.3	Data Collection	67
4.3.1	Source Selection and Search String	67
4.3.2	Inclusion and Exclusion Criteria	69
4.3.3	Search Process	70
4.4	Results	73
4.4.1	Overview of Selected Papers	73
4.4.1.1	Empirical Evaluation Approaches	73
4.4.1.2	Publication Venues and Years	75
4.4.2	RQ1 – Documenting Architectural Decisions	77
4.4.3	RQ2 – Functional Requirements and Quality Attributes	79
4.4.4	RQ3 – Domain-specific Architectural Decisions	80
4.4.5	RQ4 – Descriptive and Normative Papers	82
4.4.6	RQ5 - Addressing Uncertainty in Architectural Decisions	84
4.4.7	RQ6 - Group Architectural Decisions	85
4.5	Discussion	86
4.5.1	Analysis and Synthesis of Results	86
4.5.1.1	Empirical Evaluation Approaches, Publication Venues and Years	86
4.5.1.2	Documenting Architectural Decisions	89

4.5.1.3.	Functional Requirements and Quality Attributes	90
4.5.1.4.	Domain-specific Architectural Decisions	91
4.5.1.5.	Descriptive and Normative Papers	91
4.5.1.6.	Uncertainty in Architectural Decisions	96
4.5.1.7.	Group Architectural Decisions	97
4.5.2	Implications for Researchers and Practitioners	100
4.6	Validity threats	102
4.6.1	Conclusion Validity	102
4.6.2	Construct Validity	103
4.6.3	Internal Validity	103
4.6.4	External Validity	104
4.7	Conclusions	104

Chapter 5

Reducing Vaporization with the Repertory Grid Technique 107

5.1	Introduction	109
5.2	The Repertory Grid Technique	110
5.3	Exploratory Study	111
5.3.1	Study Design	111
5.3.2	Study Results	113
5.3.3	Advantages	116
5.3.4	Disadvantages	117
5.3.5	Application Context	117
5.3.6	Validity Threats	117
5.4	Survey study	118
5.4.1	Conceptual Model to Capture Architectural Knowledge Using the Repertory Grid Technique	119
5.4.2	Repertory Grid Technique for Capturing Architectural Knowledge	121
5.4.3	Study Definition and Design	122
5.4.4	Survey Implementation	125
5.4.5	Survey Execution	127
5.4.6	Analysis of Survey Results	128
5.4.6.1.	Collecting Metrics for a Decision	128
5.4.6.2.	Analyzing Metrics for All Decisions	130
5.4.6.3.	Post Questionnaires	132
5.4.7	Discussion	134
5.4.8	Validity Threats	134
5.5	Conclusions	136

Chapter 6

Improve Individual Architectural Decisions **137**

6.1 Introduction	138
6.2 Phase 1 – Initial REGAIN Approach.....	139
6.2.1 Theoretical Foundations for REGAIN	139
6.2.2 The REGAIN Approach	141
6.2.3 REGAIN Output	143
6.2.4 Initial REGAIN Evaluations	144
6.3 Phase 2 – Investigate Industrial Applicability of REGAIN.....	145
6.3.1 Research Method, Data Collection and Analysis	145
6.3.2 RQ1 – REGAIN Advantages and Disadvantages.....	147
6.3.2.1. Post-questionnaire Analysis	147
6.3.2.2. Transcripts Content Analysis	149
6.3.3 RQ2 – REGAIN Improvement Opportunities.....	151
6.3.4 Discussion	153
6.4 Phase 3 - Investigate Prioritization Approaches	155
6.4.1 Participants.....	157
6.4.2 Experimental Materials and Tasks.....	158
6.4.3 Hypotheses and Variables	160
6.4.3.1. Performance	160
6.4.3.2. Users’ Perceptions	161
6.4.3.3. REGAIN Output	161
6.4.3.4. Summary	162
6.4.4 Experiment Design and Results	163
6.4.4.1. Results on Performance	164
6.4.4.2. Results on Users’ Perceptions	166
6.4.4.3. Results on REGAIN Output	167
6.4.5 Discussion	168
6.5 Validity Threats.....	171
6.5.1 Interview Study Validity Threats.....	171
6.5.2 Experiment Validity Threats	172
6.6 Related Work.....	174
6.7 Conclusions	177
6.8 Acknowledgments	177

Chapter 7

Improve Group Architectural Decisions **179**

7.1 Introduction	181
7.2 The GADGET Process.....	182

7.3 GADGET Case Study	185
7.3.1 Case Study Design	185
7.3.2 Results	188
7.3.2.1 Case Study Participants and Execution	188
7.3.2.2 Analysis Results	189
RQ1 - Need for consensus in group architectural decision making	189
RQ2 - Effort and benefits	190
RQ3 – Improvements	191
7.3.3 Discussion	192
7.3.3.1 Recommendations for Practitioners	192
7.3.4 Implications for Research	193
7.4 GADGET Experiment.....	193
7.4.1 Research Goal and Questions	194
7.4.2 Participants	195
7.4.3 Experimental Materials and Tasks	197
7.4.4 Hypotheses for RQ1 – Consensus.....	200
7.4.4.1 Hypothesis on General Agreement	200
7.4.4.2 Hypothesis on Mutual Understanding on the Priorities of Concerns	200
7.4.4.3 Hypothesis on Mutual Understanding on Ratings	201
7.4.5 Hypotheses for RQ2 - Perceptions.....	202
7.4.6 Results	204
7.4.6.1 Analysis Procedure	204
7.4.6.2 Participants’ Background	205
7.4.6.3 Answer to RQ1 - Consensus	206
7.4.6.4 Answer to RQ2 - Perceptions	207
7.4.7 Discussion	208
7.4.7.1 Interpretation of Results	209
7.4.7.2 Cross-study Discussion	210
7.4.7.3 Limitations of GADGET	211
7.5 Validity Threats.....	211
7.5.1 Case Study Validity Threats.....	211
7.5.2 Experiment Validity Threats	212
7.6 Related Work	213
7.7 Conclusions	214
7.8 Acknowledgments	215

Chapter 8

Tool Support for REGAIN and GADGET	217
8.1 Introduction	218
8.2 Motivation for a new tool.....	218
8.3 Features	219

8.3.1	REGAIN Support.....	220
8.3.1.1.	Concerns Prioritization	223
8.3.2	GADGET Support.....	224
8.4	Tool Development and Deployment.....	226
8.5	Conclusions	227
8.6	Acknowledgments	227
Chapter 9		
Conclusions		229
9.1	Answers to Research Questions	229
9.1.1	RQ1. How is architectural knowledge managed in practice? 229	
9.1.2	RQ2. How are architectural decisions made in practice?	230
9.1.3	RQ3. What is the state of research on architectural decisions? 231	
9.1.4	RQ4. Can the Repertory Grid technique reduce architectural knowledge vaporization?	232
9.1.5	RQ5. How to support making and documenting individual architectural decisions?.....	233
9.1.6	RQ6. How to support making and documenting group architectural decisions?.....	233
9.1.7	RQ7. What tool can support REGAIN and GADGET?	235
9.2	Discussion.....	235
9.3	Contributions	236
9.4	Future Work.....	237
Appendices		241
10.1	Appendix for Chapter 3.....	241
10.1.1	Questionnaire for Survey	241
10.2	Appendix for Chapter 4.....	251
10.2.1	Selected Papers	251
10.2.2	Publication Venues	259
10.3	Appendix for Chapter 6.....	261
10.3.1	Phase 2 – Additional Details	261
10.3.2	Phase 3 – Additional Details	262
References		263
Acknowledgments		277

