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):

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.
Chapter 9
Conclusions

This chapter offers answers to the research questions in Chapter 1, discussion, contributions of this thesis, and future work.

9.1 Answers to Research Questions

As stated in Chapter 1, the overall problem addressed in this thesis is:

How can architectural knowledge vaporization be reduced?

This problem was refined into seven high-level research questions and 21 concrete research questions.

9.1.1 RQ1. How is architectural knowledge managed in practice?

Answering this high-level research question helped us understand the need for reducing architectural knowledge vaporization. Given the wide scope of RQ1, we focused our efforts in Chapter 2 on understanding challenges and potential solutions for managing architectural knowledge in the public sector. We had two reasons for this focus: research novelty and opportunity. Regarding novelty, no previous work existed on architectural knowledge management in the public sector, while much work existed in the private sector. Regarding opportunity, we took part in a research project, which offered us access to relevant practitioners who could offer insights on architectural knowledge management in practice.

For the study in Chapter 2, we interviewed eleven practitioners from four public and four private sector organizations, so that we could apply lessons from the private sector to the public sector. We identified challenges (e.g.
vaporization, sharing integration) for managing architectural knowledge and potential solutions (e.g. tool support, training, community building) to them. Furthermore, the study in Chapter 2 confirmed that architectural knowledge vaporization is a major challenge in organizations.

9.1.2 RQ2. How are architectural decisions made in practice?

Architectural decisions are a significant part of architectural knowledge. To propose approaches for reducing architectural knowledge vaporization, we had to understand real-world architectural decisions, so that the approaches would be applicable in practice.

We answered RQ2 in Chapter 3, in which we present a survey with 43 architects who described 86 decisions from their activity. We refined RQ2 in the following four research questions.

RQ2.1. What are the characteristics of architectural decisions?

To answer RQ2.1, we defined seven metrics to characterize architectural decisions (e.g. actual time spent making the decision, number of people directly involved). Answering RQ2.1 provided novel insights into real-world architectural decisions. For example, previously, little was known about how much time or how many people are involved in architectural decisions. In particular, in Chapter 3 we found out that most architectural decisions are made in groups. This finding encouraged us to propose the GADGET approach in Chapter 7.

RQ2.2. What factors make architectural decisions difficult?

From answering RQ2.2, we found out several factors that make decisions difficult, such as analysis effort and lack of similar previously made decisions. Identifying these factors encouraged us to propose approaches that help architects analyze their decisions and reduce architectural knowledge vaporization, so that previous decisions are available.

RQ2.3. What are the differences between junior and senior software architects?

Junior architects need help analyzing their decisions more than senior architects. In addition, senior architects consider more alternatives and quality
9.1. Answers to Research Questions

attributes at the start of the decision making. These insights encouraged us to propose the approaches in Chapters 6 and 7.

**RQ2.4. What are the differences between good and bad architectural decisions?**

We found statistically significant differences on the number of alternatives and quality attributes: good decisions have higher numbers than bad decisions.

9.1.3 **RQ3. What is the state of research on architectural decisions?**

To answer RQ3, we conducted a systematic mapping study, as reported in Chapter 4. We covered studies published between 2002 and 2012. We identified 144 relevant papers on architectural decisions. We refined RQ3 into the following six research questions, and we answered each of them using the set of relevant papers.

**RQ3.1. What are the papers on documenting architectural decisions?**

We identified much work on documenting architectural decisions, and we classified existing papers on their tool support and process for documenting. Overall, we found 120 papers that help document architectural decisions, 76 of the 120 papers present a process, and 52 of the 120 papers include tool support. Still, only five papers include open-source tool support.

**RQ3.2. Does current research on architectural decisions consider functional requirements and quality attributes?**

We found out that most (i.e. 114) of the relevant papers on architectural decisions address explicitly functional requirements and quality attributes.

**RQ3.3. What specific domains for architectural decisions are investigated?**

We found out that service-oriented and enterprise domains received much attention. However, other domains (e.g. mobile) have received little attention.

**RQ3.4. What are the normative and descriptive papers?**

We identified 20 descriptive and 124 normative papers. Furthermore, for the descriptive papers, we analyzed the number of decisions, time spent for making decisions, number of participants, and classes of decisions.
RQ3.5. What are the papers on addressing uncertainty in architectural decisions?

We identified only nine papers on addressing uncertainty in architectural decisions. Given its importance, we consider that addressing uncertainty is an important future research topic for architectural decisions.

RQ3.6. What are the papers on group architectural decisions?

We identified 22 papers that refer to group architectural decisions. About half of these papers are descriptive work, offering evidence on group architectural decisions in the industry. The relatively low number of normative approaches for group architectural decisions encouraged us to propose the GADGET approach in Chapter 7.

9.1.4 RQ4. Can the Repertory Grid technique reduce architectural knowledge vaporization?

The answers to RQ1, RQ2, and RQ3 encouraged us to explore ideas for reducing architectural knowledge vaporization from other fields. The knowledge engineering field had the potential to offer useful ideas, since the field has much experience on capturing knowledge from experts. In the knowledge engineering field, the Repertory Grid technique is an established approach to capture knowledge. However, this technique has not been investigated for capturing architectural knowledge. We refined RQ4 into two research questions, which we investigated in Chapter 5 in two studies.

RQ4.1. What are the advantages and disadvantages of the Repertory Grid technique for capturing architectural knowledge?

Following a study with students, we identified the following advantages. The Repertory Grid is a systematic approach, which encourages reflection on the decisions, and helps architects with their decision making. Disadvantages include learning curve, tool support, and effort.

RQ4.2. Does the Repertory Grid technique reduce AK vaporization more than a template-based approach to document architectural decisions?

In a different study with students, we compared the documentation created with the Repertory Grid technique against the documentation created with a template for documenting decisions. We found out that the documentation
created with the Repertory Grid technique contains more alternatives, concerns and ratings, compared to the template-based documentation.

9.1.5 RQ5. How to support making and documenting individual architectural decisions?

Chapter 5 presents two studies with students from which we learnt that the Repertory Grid technique had much potential for reducing architectural knowledge vaporization. Based on this finding, in Chapter 6, we proposed REGAIN – an approach based on the Repertory Grid technique to make and document individual architectural decisions. To answer RQ5, we proposed the following three research questions on REGAIN.

RQ5.1. What are the advantages and disadvantages of REGAIN?

We asked practitioners to use REGAIN and then offer feedback on its advantages and disadvantages. We learnt that REGAIN is a systematic approach that offers decision-making support, documentation of decision rationale, and reasoning support to practitioners. Disadvantages include limited tool support, subjectivity, and effort.

RQ5.2. What are the improvement opportunities for REGAIN?

Practitioners indicated several improvement opportunities for REGAIN such as: concerns prioritization (addressed in RQ5.3), group decisions (addressed in RQ6), decision reuse, and sensitivity analysis.

RQ5.3. Which concerns prioritization approach to use for REGAIN?

We conducted an experiment with students to compare two prioritization approaches: the hundred-dollar and pairwise comparisons approaches. We investigated hypotheses on performance, users’ perceptions, and impact on REGAIN output for the two approaches. Following the experiment, we recommended using the hundred-dollar approach with REGAIN for most situations.

9.1.6 RQ6. How to support making and documenting group architectural decisions?

In Chapter 3 we found out that most architectural decisions are made in groups, rather than individually. In Chapter 4 we found little work on group
architectural decision making. Therefore, we were interested in supporting group architectural decisions. In Chapter 7, we propose and validate GADGET - an approach for increasing consensus in group architectural decision making. We refine RQ6 in the following five research questions. The first three research questions were answered by conducting a case study with practitioners and students, in which participants used GADGET. The last two research questions were answered by conducting an experiment with students, in which half of the participants used GADGET and the other half used ADHOC (i.e. group decision making without any prescribed approach).

RQ6.1. **Is there a practical need for increasing consensus in group architectural decision making?**

By analyzing the feedback from case study participants, we learnt that there is a need to increase consensus in group architectural decision making for two reasons. First, conflicting perspectives occur in practice. Second, conflicting perspectives are time consuming.

RQ6.2. **What are the effort and benefits offered by GADGET?**

GADGET takes one to three hours to use for one decision. The main benefit is that participants developed an increased shared perspective of each other’s perspectives. Other benefits include capturing rationales of decisions, time savings by keeping the group discussions focused, equal engagement of participants, and traceability of the decision.

RQ6.3. **What are potential improvements to GADGET?**

Potential improvements include elimination of less promising alternatives, offering more concerns’ prioritization approaches, tool support, and confidence levels to ratings.

RQ6.4. **Compared to ADHOC, what is the impact of GADGET on increasing consensus among group architectural decision makers?**

We considered two components of consensus: general agreement and mutual understanding. From the experiments with students, we learnt that GADGET increases mutual understanding among participants better than ADHOC, but no difference between GADGET and ADHOC on increasing the general agreement among participants.
9.2 Discussion

RQ6.5. How do perceptions on GADGET vs. ADHOC differ among decision makers?

Participants perceived that, compared to ADHOC, GADGET captures more rationales for architectural decisions, thus reducing architectural knowledge vaporization than ADHOC.

9.1.7 RQ7. What tool can support REGAIN and GADGET?

Chapter 4 reports that there is a very low number of open source tool support for making and capturing architectural decisions. In Chapters 6 and 7, practitioners indicated the need for user-friendly, open source tool support for REGAIN and GADGET. Towards this, we developed open source tool support. Chapter 8 reports motivation, features, and development aspects of the tool.

9.2 Discussion

Chapter 1 presents five reasons that contribute to architectural knowledge vaporization. Based on the answers to research questions (Section 9.1), we discuss how the two proposed architectural decision making processes (i.e. REGAIN and GADGET) address the five reasons.

- **Unawareness** – improved processes should encourage architects to think more about their decisions. In Chapter 6, we found out that REGAIN offers systematic reasoning support for making architectural decisions, which encourages architects to think about their decisions. In Chapter 7, we found out that GADGET helps architects think about their decisions and clarify their points of view on the decisions.

- **Lack of training** – improved processes should have a low learning curve and provide sufficient advantages, so that architects are motivated to learn and use them. In Chapter 6, we found out that REGAIN has the disadvantage of a learning curve, but REGAIN has important advantages: systematic decision-making support, capturing decision rationale, and reasoning support. In Chapter 7, we found out the GADGET also has the disadvantage of a learning curve, but GADGET offers important advantages, such as increased focus of the group discussions, and capturing decision rationale. Both REGAIN
and GADGET have the advantage of friendly, open source tool support, which can motivate architects to learn and use them.

- **Difficulty** - improved processes should minimize documentation efforts and include steps to decompose the documentation task in small and easy to perform steps. In Chapters 6 and 7, we found out that REGAIN and GADGET have the disadvantage of effort. However, this disadvantage is specific to any systematic approach. The actual documentation effort is minimized, since REGAIN and GADGET are based on a minimalistic, core model of architectural knowledge, which encourages capturing the essential documentation of architectural decisions. Furthermore, both REGAIN and GADGET have small steps, which are easy to perform, after passing the learning curve.

- **Disruption** – improved processes should encourage architects to focus on their decision making. Both REGAIN and GADGET offer decision-making support for architects, and they also capture rationales of decisions, thus reducing architectural knowledge vaporization.

- **Natural causes** – improved processes should encourage immediate capturing of decisions rationales, to avoid the risk of architects forgetting them over time. Both REGAIN and GADGET help capture decisions rationales while decisions are made, thus reducing the risk of forgetting the rationales.

### 9.3 Contributions

This thesis brings the following contributions to the state of the art of the software architecture field.

Chapter 2 contributes with insights from real-world implementations of architectural knowledge management in the private sector, and insights into architectural knowledge management challenges and solutions for the public and private sector.

Chapter 3 contributes with insights on characteristics of real-world architectural decisions, on factors that make decisions difficult, on differences between junior and senior architects, and on differences between good and bad decisions.
9.4. Future Work

Chapter 4 summarizes a decade of research on architectural decisions and proposes promising future research directions, such as addressing uncertainty in decision making and group architectural decision making.

Chapter 5 contributes with initial evidence on the potential of the Repertory Grid technique to reduce architectural knowledge vaporization.

Chapter 6 proposes and validates REGAIN – an approach based on the Repertory Grid technique for making and documenting individual architectural decisions.

Chapter 7 proposes and validates GADGET – an approach that extends REGAIN for making and documenting group architectural decisions.

Chapter 8 contributes with open-source tool support for REGAIN and GADGET.

9.4 Future Work

This thesis answers several research questions and makes certain contributions to the software architecture field. In an ideal world, such answers would be definitive and would necessitate no further inquiry, but this is not the case. Therefore, we indicate below future work items for the academic and industrial communities.

In Chapter 2, we reported challenges and solutions for managing architectural knowledge. Particularly, we identified two challenges: the role of organizational culture, and the integration of architectural knowledge management with organizational goals. These challenges have received little attention so far in the literature, and they require further research to understand their role in architectural knowledge management efforts. In addition, researchers can use the results of the study in Chapter 2 to propose taxonomies of challenges and solutions for architectural knowledge management.

Chapter 3 provides insights into real-world architectural decisions. Still, more work is needed to further understand real-world architectural decisions, in particular to understand characteristics of decisions and difficulty factors across domains. This would help researchers to offer practitioners domain-specific decision support, to address the specific difficulty factors from each domain.
Chapter 4 shows that uncertainty in architectural decisions has received little attention so far. In addition, since there are few studies on real-world group architectural decision making (as identified in Chapter 4, and investigated already in Chapter 7), there is a need to further study group architectural decision making in practice. Furthermore, as discussed in Chapter 4, there is a need for stating explicitly the number and classes of decisions in descriptive papers, so that findings can be compared across studies.

In Chapter 5, we explored a technique from the knowledge engineering field (i.e. the Repertory Grid technique) to capture architectural knowledge. However, the knowledge engineering field may offer additional techniques with high applicability for capturing architectural knowledge. A systematic mapping study of knowledge engineering literature would be the first step to identify additional techniques and their potential applicability to the software architecture field.

As follow-up work to the REGAIN and GADGET approaches proposed in Chapters 6 and 7, we plan to propose an ISO42010-compliant documentation framework with viewpoints that help reuse architectural knowledge and capture dependencies among architectural decisions. Furthermore, we only used ratings from one to five in REGAIN and GADGET, but in future work we consider adding other types of ratings, such as specific categories. Additionally, there is a need to investigate how to increase the practitioners’ acceptance of approaches on architectural decisions, so that practitioners can benefit from approaches (such as REGAIN and GADGET) proposed by researchers. For example, the Technology Acceptance Model (Venkatesh and Davis, 2000) offers a powerful model on how users accept and use a new technology.

As future work, there is a need to support practitioners to tackle other challenges of group architectural decision making, in addition to consensus, which we studied in Chapter 7. In addition, since there is a need for treating uncertainty in architectural decision making (identified in Chapter 4), we will update REGAIN and GADGET to include support for uncertainty in individual and group architectural decision making. Finally, there is a need to define criteria for evaluating various group architectural decision making processes across different studies.
In Chapter 8, we presented open source tool support for REGAIN and GADGET. In future work, we plan to add additional features to the tool, such as facilitating reuse of concerns based on concern-specific keywords. Also, we plan to evaluate the tool to understand how much time practitioners save by using the tool, and further improve the tool. Finally, given the importance of handling dependencies among decisions, we will add new features to help architects analyze dependencies among decisions.