The Journal of Systems and Software 83 (2010) 2441–2455

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

An exploratory study of architectural effects on requirements decisions☆

James A. Miller, Remo Ferrari*, Nazim H. Madhavji*

Department of Computer Science, University of Western Ontario, London, ON, Canada N6A 5B7

ARTICLE INFO

Article history:
Received 31 January 2009
Received in revised form 5 July 2010
Accepted 5 July 2010
Available online 15 July 2010

Keywords:
Software architecture
Requirements engineering
Empirical study
Software quality
Process improvement
Quantitative and qualitative research
Architecture and requirements technology

ABSTRACT

The question of the "manner in which an existing software architecture affects requirements decision-making" is considered important in the research community; however, to our knowledge, this issue has not been scientifically explored. We do not know, for example, the characteristics of such architectural effects. This paper describes an exploratory study on this question. Specific types of architectural effects on requirements decisions are identified, as are different aspects of the architecture together with the extent of their effects. This paper gives quantitative measures and qualitative interpretation of the findings. The understanding gained from this study has several implications in the areas of: project planning and risk management, requirements engineering (RE) and software architecture (SA) technology, architecture evolution, tighter integration of RE and SA processes, and middleware in architectures. Furthermore, we describe several new hypotheses that have emerged from this study, that provide grounds for future empirical work. This study involved six RE teams (of university students), whose task was to elicit new requirements for upgrading a pre-existing banking software infrastructure. The data collected was based on a new meta-model for requirements decisions, which is a bi-product of this study.

1. Introduction

No one would deny that if we were to extend an existing edifice, many of its functional and non-functional features would be of central importance in considering new requirements for the extension. Yet, in the software engineering (SE) literature, this is rather an understated issue—that is, consideration of existing system design is not a key factor in engineering new requirements. While in software practice many developers are indeed aware of the need to assess the fitness of new requirements with the existing system design, the approaches are rather subjective and experiential. SWEBOK (IEEE SWEBOK, 2004) – the SE body of knowledge – for example, does not describe any practices to deal with this issue. To explore this issue further, we conducted a preliminary survey of 17 professional requirements engineers and software architects. We found that the average rating of the importance of considering existing system architecture (SA[1]) when engineering new requirements was 4.5 (on a 1–5 Likert-scale)—implying that the respondents strongly agreed with this concept. Despite this, several respondents noted in the qualitative part of the survey that in actual practice, many organizations neglect this consideration, or only perform analysis on existing high-level feature descriptions of the current system, and not the system's architecture. In many situations, a lack of consideration for an existing system in the new requirements work can lead to rework of requirements and design, incurring extensive costs especially if further downstream in the development process (Boehm and Basili, 2001).

The uptake of this, architecture-requirements, issue in research is not impressive either. It was not until 1994 that the role of an existing SA in requirements engineering (RE) was recognised as important in a panel session. However, at that time, "we still [did] not have a clear understanding of [it]" (Shekaran, 1994a). Shortly thereafter, 5 of the 34 identified indicators of RE success were found to have links with SA (El Emam and Madhavji, 1995). A few years later, the question of an architecture's role in RE was raised again (Nuseibeh and Easterbrook, 2000). While the awareness of an architecture's role in the RE process has no doubt increased, to our knowledge, the effects of an existing SA on RE decisions have not been scientifically explored. It is not until such studies are conducted, and a dependable body of knowledge created, that practice can begin to use such knowledge in day-to-day projects. As a first step in this direction, this paper describes an exploratory case study on the effects of an existing SA on RE decisions. Specifically, we ask:

> "In which manner does an architecture affect requirements decision-making[2]?"

☆ A preliminary version of this paper was published in Miller et al. (2008).

* Corresponding authors.
E-mail addresses: rnferrar@csd.uwo.ca (R. Ferrari), madhavji@csd.uwo.ca (N.H. Madhavji).

[1] For the rest of the paper, the acronym SA refers to system (or software) architecture as a software artefact.

[2] Decision-making leads from recognition of a problem to be solved to a specification of that problem or a solution strategy.

0164-1212/$ – see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2010.07.006

We explore this question on two fronts: (1) the kind of role a SA plays in requirements decision-making and (2) the specific aspects of the architecture that affect RE decisions.

For point (1) above, it has already been suggested that a SA might *constrain* a RE process (Shekaran, 1994b). For example, while analysts could be eliciting requirements to employ a new technology that requires a specific communication protocol, the current legacy system has long implemented a conflicting communications protocol, thereby *constraining* the current RE strategy. For point (2) above, while SA aspects are likely largely unique to the domain of these cases, they would give us an indication of which parts of an existing software architecture can affect RE decision-making (e.g., non-functional SA areas outside the focus of an RE agent) and, consequently, which parts of the architecture are critical to document for use by requirements engineers.

Our results indicate that the relationship between SA and RE is more complex than what is intuitively known in the literature. In particular, “SA as a constraint” is only one of the four types of effects observed in our study. The other three types of effects we found are: *enabled*, *influenced* and *neutral*. In short, an *enabled* effect is where the proposed solution (denoted by the new requirements) is made feasible because of the implemented decisions in the existing system; an *influenced* effect is where the architectural configuration has an effect on the requirements decision without affecting its feasibility; and a *neutral* effect is where there is no noticeable architectural effect on the decision. This paper gives quantitative measures on these effects from the study and qualitative interpretation of the findings. Also, in our study, nine architectural aspects were identified across 117-recorded decisions. Again, this paper gives quantitative measures and qualitative interpretations.

A deeper understanding of the role of SA in RE could open up new opportunities for RE and architecting methods, tools and processes. For instance, in the area of planning and risk assessment, the management could make more informed cost estimates of new requirements by considering how the SA has historically affected the various types of requirements. Likewise, in the area of technology improvement, RE and SA tools can be integrated so that analysts and architects can share, access and change requirements and architecture information more easily. We describe several other cases in the paper.

Our empirical study involved six RE teams that gathered new requirements for an existing system and were observed over the course of 2 months. The project was in the banking domain and required the RE teams to elicit and analyse new requirements based on a set of high-level features that needed to be integrated into the current architecture. A requirements decision meta-model was created as a basis for the development of a requirements-tool that served to gather data from the participants during the project on how requirements decisions they were making were affected by specific aspects of the existing architecture. This paper describes: the study context, participant details, project work involved, the underlying decision meta-model[3] for the data that is gathered, use of tools for gathering data, and the various threats to validity.

The key results are the quantitative characterization of the different interaction effects mentioned earlier. For example, for this particular system, nine SA aspects affected approximately 60% of the RE decisions. From the findings, we have derived four hypotheses that provide a basis for future studies. A general description of how each of these studies could be conducted is also described.

This paper is structured as follows: Section 2 discusses related work; Section 3 describes the exploratory study; Section 4 presents the results; Section 5 discusses various implications from the results; Section 6 discusses future empirical work and emergent hypotheses from this study, and Section 7 concludes the paper.

[3] The decision meta-model defines the type of data relevant to this study and is a basis for the tool developed for data gathering. The meta-model and tool are bi-products of this study.

2. Related work

This section describes the work that is related to our study. The section focuses on three key aspects: (i) observations, commentary and empirical work on the relationship between RE and SA, (ii) technological research spanning RE–SA, and (iii) recent technological-based research on architecture evolution. In Section 2.4, the section concludes with a reflection on the current state of research described in Sections 2.1–2.3.

2.1. RE and SA relationship

There is an increasing interest in exploring and refining the transitions between various activities in the software development process. In particular, the relationship between RE and SA, and their impact on each other was the focus of a couple of workshops 7–9 years ago (STRAW, 2001, 2003). In fact, even earlier, Jackson argued in a panel session (Jackson, 1994) for a tight coupling of the RE and SA processes, suggesting that the most successful developers are those who are able to move relatively more freely between stages within the development cycle. In Kozaczynski (2002), the author discusses that a level of foresight on the part of architects to focus on those requirements that are architecturally relevant can help to mitigate development risk in the software process, by being able to develop the architecture early without all requirements being elicited. This, early development, can then be fed back to the requirements process to further refine the requirements.

In our earlier work in El Emam and Madhavji (1995), they presented an instrument for measuring RE success. Through an industry field study to design this instrument, we found that in evolutionary work, the level of understanding of the existing software architecture can have an impact on the success of the RE process. In understanding the architecture, requirements engineers can provide requirements solutions that are consistent with the current technical and corporate orientation of its organization. In turn, this can lead to better cost/benefit analysis during RE. This early understanding, however, did not delve into the type of technical effects an existing architecture has on RE decision-making; in this paper, we investigate this issue further.

In Garlan (1994), he recognises that architectural families constrain system requirements. Further, he identifies that solutions can drive requirements. For example, the architecture of a family of systems determines the range of variability allowed in a product line. Though not explicitly stated, one can interpret this as not only architectures imposing “constrains” on requirements decision-making, but also as “enabling” and “influencing” such decision-making. This is a central aspect of the current paper.

In Bass et al. (2003), they discuss that different stakeholders of the architecture will have different needs for documentation, and the level of detail provided to them should reflect this. Depending on the stakeholders’ needs, they can be provided with *detailed information*, *some details* or *overview information* of the various architectural views available. The specific architectural aspects that could be important in RE, however, are not mentioned in Bass et al. (2003); our study uncovers these details.

Three previous studies of ours, described below, empirically examine RE–SA interaction issues from the viewpoints of: architecting-problems rooted in requirements, the effect of using different types of human agents when architecting, and the impact of an SA on requirements characteristics. In Ferrari and Madhavji (2008a), we report on a multiple-case study that investi-