Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

The Journal of Systems and Software 79 (2006) 912–925

The Journal of Systems and Software

www.elsevier.com/locate/jss

An empirical study of groupware support for distributed software architecture evaluation process

Muhammad Ali Babar *, Barbara Kitchenham, Liming Zhu, Ian Gorton, Ross Jeffery

Empirical Software Engineering, National ICT Australia, University of New South Wales, Sydney, Australia

Received 6 December 2004; accepted 20 June 2005
Available online 2 February 2006

Abstract

Software architecture evaluation is an effective means of addressing quality related issues early in the software development lifecycle. Scenario-based approaches to evaluate architecture usually involve a large number of stakeholders, who need to be collocated for face-to-face evaluation meetings. Collocating a large number of stakeholders is an expensive and time-consuming exercise, which may prove to be a hurdle in the wide-spread adoption of disciplined architectural evaluation practices. Drawing upon the successful introduction of groupware applications to support geographically distributed teams in software inspection, and requirements engineering disciplines, we propose the concept of distributed architectural evaluation using Internet-based collaborative technologies. This paper presents a pilot study used to assess the viability of a larger experiment intended to investigate the feasibility of groupware support for distributed software architecture evaluation. In addition, the results of the pilot study provide some preliminary findings on the viability of groupware-supported software architectural evaluation process.

Keywords: Software architecture evaluation; Groupware systems; Empirical software engineering; Distributed software development; Software process improvement

1. Introduction

Software architecture (SA) evaluation is an effective mechanism for improving the quality of software intensive systems. The main objective of SA evaluation is to consider and address quality requirements at the SA level (Bass et al., 2003; Maranzano et al., 2005). There are various techniques and tools to assess the potential of the chosen architecture to deliver a system capable of satisfying desired quality requirements and identify potential risks. Most of the well-known SA assessment approaches are scenario-based methods (Ali-Babar et al., 2004) such as Architecture Tradeoff Analysis Method (ATAM) (Kazman et al., 1999), Software Architecture Analysis Method (SAAM) (Kazman et al., 1994) and Architecture-Level Maintainability Analysis (ALMA) (Lassing et al., 2002).

Scenario-based SA evaluation is a collaborative exercise that involves a number of stakeholders. Currently, it requires all the major stakeholders to be collocated for face-to-face (F2F) meeting to perform various activities, such as defining and refining business drivers, generating quality sensitive scenarios, and mapping the scenarios on to the proposed architecture. This is an expensive and time consuming process. Besides setting aside significant amount of time, stakeholders may have to travel if they are geographically distributed, which is highly likely as companies increasingly develop software using geographically distributed teams (Carmel and Agarwal, 2001; Herbsleb and Moitra, 2001; Mashayekhi et al., 1994; Perry et al., 2002). Organizational concerns about the cost and scheduling difficulties for collocating large number of stakeholders have been widely reported (Layzell et al., 2000; Perry et al.,

* Corresponding author. Tel.: +61 3 8374 5515; fax: +61 3 8374 5520.
E-mail addresses: malibaba@nicta.com.au (M.A. Babar), Barbara.kitchenham@nicta.com.au (B. Kitchenham), liming.zhu@nicta.com.au (L. Zhu), ian.gorton@nicta.com.au (I. Gorton), ross.jeffery@nicta.com.au (R. Jeffery).

0164-1212/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2005.06.043

2002). These difficulties may hinder the wide-spread adoption of SA evaluation practices.

In an attempt to find a cost effective and efficient alternative to F2F meeting-based SA evaluation, we suggest that Internet-based collaborative technologies may provide a mechanism of addressing some of above-mentioned issues (Collaborative technologies include web-based applications that support collaboration, e.g., groupware systems, collaborative and CSCW applications, etc.). Researchers and practitioners in various sub-disciplines of software engineering (such as requirements engineering, inspections and others) have successfully evaluated groupware supported processes as a promising way to introduce software shift-work, minimize meeting costs, maximize asynchronous work and conserve a number of precious organizational resources (Boeham et al., 2001; Gorton et al., 1996; Halling et al., 2001; Perry et al., 2002). Drawing on the positive results of using groupware systems in similar domains, we propose that the collaborative applications can be used to improve the SA evaluation process without compromising the quality of the artifacts and results.

However, there are a number of important issues that should be explored before making any conclusive claim about the effectiveness of the collaborative applications for distributed SA evaluation. For example, we need to understand the changes required in the existing SA evaluation approaches to allow for distributed environments. We also need to identify appropriate collaborative technologies to support distributed SA assessment and gain a better understanding of how they facilitate or hinder social processes. We intend to use experimentation to study these issues (Perry et al., 2002).

In order to evaluate the effectiveness of distributed SA evaluation, we have designed an empirical research program based on a framework of experimentation (Basili et al., 1986) and guidelines provided in Kitchenham et al. (2002). The experimental program consists of a pilot study followed by a large-scale experiment. This paper reports the results of the pilot study from two viewpoints. Firstly, the pilot study has provided some initial information about the use of groupware to support SA evaluation in distributed arrangement and secondly it has allowed us to refine our subsequent experimental program.

The salient features of this paper are:

- It briefly discusses the concept of distributed SA evaluation using collaborative technologies.
- The pilot study results provide an initial assessment of the effect of using distributed meeting for SA evaluation activities.
- We show how the results of the pilot study can be used to assess the number of experimental units needed in experiments.

The remainder of the paper is organized as follows. In the next section, we briefly review the work that has motivated our research program. We then present our idea of a distributed SA evaluation process. We describe experiment details in Section 4. Analysis and interpretation are presented in Section 5. We close the paper with the conclusions and plans for future research.

2. Background

2.1. Software architecture evaluation

Recently it has been widely recognized that quality attributes (such as maintainability, reliability, etc.) of complex software intensive systems largely depend on the overall SA of such systems (Bass et al., 2003). Since SA plays a vital role in achieving system wide quality attributes, it is important to evaluate a system's architecture with regard to desired quality requirements. SA community has developed several methods to support disciplined architecture evaluation practices. Most of the mature architectural evaluation methods are scenario-based such as Architecture Tradeoff Analysis Method (ATAM) (Kazman et al., 1999), Software Architecture Analysis Method (SAAM) (Kazman et al., 1994) and Architecture-Level Maintainability Analysis (ALMA) (Lassing et al., 2002).

Although there are differences among these methods (Ali-Babar et al., 2004), we have identified five common activities by comparing four main approaches to evaluate architecture (Ali-Babar and Gorton, 2004). Fig. 1 presents these five activities, which can make up a generic scenario-based SA evaluation process that can be supported by a groupware application. Following is a brief description of each activity in this generic SA evaluation process:

1. *Evaluation planning and preparation*—This is concerned with allocating organizational resources and setting goals for evaluation, selecting stakeholders, preparing

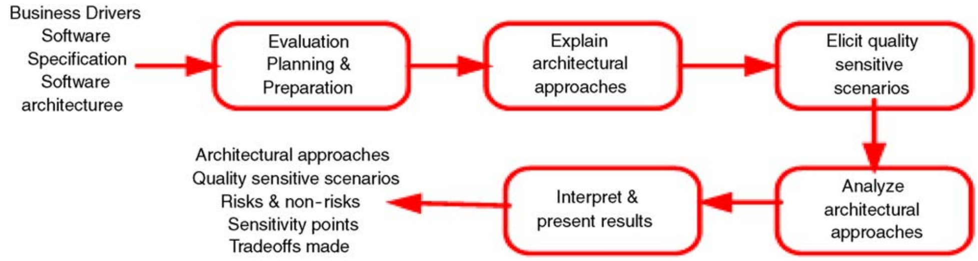

Fig. 1. A generic software architecture evaluation process.