

Software process improvement as emergent change: A structural analysis

I. Allison ^{a,*}, Y. Merali ^b

^a *School of Computing and Informatics, Nottingham Trent University, NG11 8NS, UK*

^b *Warwick Business School, University of Warwick, CV4 7AL, UK*

Available online 11 February 2007

Abstract

This paper presents a framework that draws on Structuration theory and dialectical hermeneutics to explicate the dynamics of software process improvement (SPI) in a packaged software organisation. Adding to the growing body of qualitative research, this approach overcomes some of the criticisms of interpretive studies, especially the need for the research to be reflexive in nature.

Our longitudinal analysis of the case study shows SPI to be an emergent rather than a deterministic activity: the design and action of the change process are shown to be intertwined and shaped by their context. This understanding is based upon a structural perspective that highlights how the unfolding/realisation of the process improvement (intent) are enabled and constrained by their context. The work builds on the recognition that the improvements can be understood from an organisational learning perspective. Fresh insights to the improvement process are developed by recognising the role of the individual to influence the improvement through facilitating or resisting the changes. The understanding gained here can be applied by organisations to enable them to improve the effectiveness of their SPI programmes, and so improve the quality of their software.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Software process improvement; Software quality; Software package development; Structuration theory

1. Introduction

Software process improvement (SPI) facilitates the identification and application of changes to the development and management activities in order to improve the product. Perry et al. [26] show that without understanding the technological, social and organisational aspects of software development we cannot hope to significantly improve processes. This work therefore extends the existing literature by investigating the effect of contextual and social factors on the changes in software processes as they are enacted. An explanatory theory of the SPI change process is developed from the experiences of a specific software package organisation over a 10-year period providing an understanding of how and why software process improvements occur and what the consequences of the change process

are within this specific case. This understanding is based upon a structural perspective that highlights how the process improvements are enabled and constrained by their context. An emergent view of software process change helps to understand the way the actions intertwine to inform each other, and shape and are shaped by the context they are in. The outcomes of software products and processes emerge from this intertwining.

The paper begins by showing that in contrast to the software engineering literature that tends to be restricted to a rational, deterministic view of change, the on-going nature of the software processes needs to be placed at the heart of the analysis: opening up the facets of the change not taking it as a given. To support this analysis a theoretical framework is outlined; the framework is drawn from a combination of the case study data and elements of the existing literature. The qualitative methods adopted in this study and the case study methodology are explained to enable the reader to appreciate the basis of the findings.

* Corresponding author. Tel.: +44 115 8488357.

E-mail address: ian.allison@ntu.ac.uk (I. Allison).

The core of the paper is a chronological analysis of the case. The case analysis accentuates the way in which the changes emerge and develop through time, often differing to intended actions or from the way in which the literature suggests the process improvement initiative or software engineering techniques should be instigated. This narrative adopts the structure of the theoretical framework for the case narrative, and evaluates the outcomes of the process changes.

Finally, lessons are drawn from the study in three areas. To inform SPI theory we highlight the importance of incorporating an understanding of the complexities of the dynamics that occur as software processes emerge over time. For software engineering practice, lessons are learnt by considering how SPI programmes would benefit from recognising the emergent nature of the improvement, and that more fully acknowledging how the processes can support the business objectives would help to focus the improvements. And then, for other qualitative researchers, we highlight lessons learnt from our approach to this study with respect to improving the relevance of such studies for software practice.

2. Software process improvement research

Much of the current understanding of software process improvement has been derived from the work of the Software Engineering Institute. To support improvement programmes, the software engineering community has developed a set of normative maturity models for organisations to follow and enable the assessment of current capability. Within such norm-based models, improvement in the software process is considered to result in the maturing of the activities undertaken by a software development group [17]. Evidence shows that benefits can be achieved as a result of this adoption [14]. Consequently, the majority of the work to date has concentrated on developing such models [13].

The normative models, though, are criticised for the rigidity of the pre-defined actions and their underlying deterministic assumptions about implementation [6]; and for their inflexibility and the emphasis of technology rather than people [21]. Subsequently, not all companies have found software process improvement to be beneficial with many abandoning SPI programmes [14]. Some leading commercial software producers do not, therefore, follow software process improvement according to the maturity models or quality standards [9]. However, the software engineering literature has tended to assume that an ability to instigate, plan and direct all forms of change can be taken for granted, creating ‘the illusion of manageability’ [16, p. 6]. Indeed, change is far from controllable and can only be influenced to a limited extent; the intended purpose of the intervention is often overcome by unexpected or unintended outcomes. So the challenge is to understand change not as a predictable or designed causal outcome,

but as an emergent process developed from the relationship between people and their context.

A holistic approach is necessary to study all the relevant factors in a given software context [26]. What is required therefore is an integrative theory, building on research that examines the enabling and constraining factors on quality management practice [29]: an understanding of change that reflects a more complex, dynamic and unpredictable world. Yet, SPI research does not pay enough attention to the organisational factors that enable or constrain the process improvements. So, it is appropriate to examine whether organisational issues arise as software development groups move towards a more structured, process-oriented environment. In summary the main questions of this study are:

- How does the software process improvement initiative unfold within the context of a packaged software organisation, and how does this compare to stated intent?
- What are the critical influences on the software process improvement activity as it is enacted? How and why do these influences enable/constrain changes to the processes?
- In relation to adopting process innovations, how and why do the behaviours of individuals affect the dynamics of the software practice?

3. SPI as situated change: a structural perspective

In order to highlight the emergence, interplay and outcomes occurring from the software practice a contextualist and processual perspective is adopted. Taking a processual view enables us to attend to the unfolding interplay between the espoused process improvement logic and the perceived outcomes of the realised process.

A framework was devised to enable case analysis to be more attentive to the emergent nature of the SPI activity. The framework both provides a conceptual understanding of the process of change drawn from the case study and acts as a lens through which the case can be discussed. It rests on a combination of the case study data and elements of the existing literature. Like all models this framework is a simplification of a complex reality intended to attune the researcher to key concepts within this form of change activity. It is intended to be used to inform the research and is not seen as a rigid model to be adhered to, thereby inhibiting the interpretation of events. The aspects highlighted are not intended to be a definitive or exhaustive, but these concepts can sensitise the analysis of a software process improvement initiative.

A central theme of the framework is the way that process improvement interweaves with product development, within their specific historical context (Fig. 1). The data analysis therefore includes this view of practice, with explanations of the changes that weave together the actions and decisions of practitioners in the use and adoption of software processes, and the dynamics of situated practice

Download English Version:

<https://daneshyari.com/en/article/550644>

Download Persian Version:

<https://daneshyari.com/article/550644>

[Daneshyari.com](https://daneshyari.com)