

Defining Software Processes Through Process Workshops: A Multicase Study

Finn Olav Bjørnson¹, Tor Stålhane¹, Nils Brede Moe², and Torgeir Dingsøy²

¹ Department of Computer and Information Science,
Norwegian University of Science and Technology
NO-7491 Trondheim, Norway
{bjornson, stalhane}@idi.ntnu.no

² SINTEF Information and Communication Technology
NO-7465 Trondheim, Norway
{Nils.B.Moe, Torgeir.Dingsoyr}@sintef.no

Abstract. We present the application of the process workshop method to define revised work processes in software development companies. Through two empirical action research studies, we study the impact of company premises and goals on the execution and subsequently on the results of the method. We conclude that both premises and goals will influence the workshops, and suggest how the focus of the workshops should be altered to achieve better results depending on the context. We also strengthen previous claims that the process workshops are a good arena that fosters discussion and organizational learning, and that involvement in the workshops leads to higher acceptance and usage of the resulting process.

Keywords: Software Process Improvement, Project Workshop, Empirical Study, Action Research.

1 Introduction

The way we develop and maintain software, or the software process, has long been regarded as crucial for software quality and productivity [16]. In many companies, software development is performed in a rather informal fashion, and problems of late and unsatisfactory deliveries are not uncommon.

Problems related to the use of informal development include problems with transferring competence from one project to another, difficulties in establishing best practices, and the widely varying nature of problems to be solved. In order to address these challenges and to improve the quality of the software development process, a lot of companies develop process guides to structure their work.

The process workshop (PWS) method was designed as a lightweight method to help facilitate the development of such process guides. Apart from the original introduction of the process workshop [11] and a Finnish application of the same method [19], there is little empirical evidence on the practical application of this method. This paper aims to add to the body of knowledge on process workshops as a

tool for software process improvement, and describes how company context and goals affects the execution of the method and its results.

In the following we describe our work in two companies, hereafter referred to as Alpha and Beta Company. One is a small and one is a medium sized software company, and they both used process workshops to define their software process. Our focus is on the process workshop itself and how processes were constructed. The description of this process, i.e., how it will later appear in an electronic process guide, and the cost-benefit of the process workshop method is as such outside the scope of this paper. Our research goal which we want to answer in this paper is:

How do available information, company context and goals affect the execution and results of process workshops?

The paper is structured as follows: In chapter 2 we take a closer look at related work, and the method we adapted for our cases. Chapter 3 describes the research method employed in each case. Chapter 4 gives a deeper introduction to each case. Chapter 5 discusses the differences between the cases and our findings. Chapter 6 concludes our findings and describes possible routes for further research.

2 Related Work

When companies choose to design their own development processes, one option is to assign the task to a group of expert “process engineers” as described by Becker-Kornstaedt [7, 8]. One or more process engineers elicit process data from interviews, documents, surveys, e-mails and observation, and then interpret this data to produce a process model. This approach relies heavily on the experience and skill of the process engineer. Therefore, without any structured method, quality and repeatability cannot be ensured. It is, however, unlikely that the use of qualitative methods alone can compensate for experience in process modeling and software engineering [8]. When using a process engineer to formulate a process model, it is common to create a descriptive model. A descriptive model is a model, which expresses processes currently in use. Descriptive software process modeling is an important part of any software process improvement (SPI) program, because descriptive modeling allows process engineers to understand existing processes, communicate process and analyze existing practices for improvement [8]. For this reason, much work has been done on proposing languages, techniques and tools for descriptive process modeling.

An alternative to using process engineers is to involve the employees more in designing the process models, for example through workshops [1, 17]. This type of work takes up the heritage from employee participation in organizational development, a part of Scandinavian work tradition as well as in most work on improvement, from the Total Quality Management principles [10] to the knowledge management tradition in Communities of Practice [25]. Participation is also one of the most important foundations of organization development and change [17], and one of the critical factors for success in software process improvement [13].

Some studies have found that employee involvement lead to organizational effectiveness, measured through financial performance, turnover rate and workforce