

Software Process Improvement with Agile Practices in a Large Telecom Company

Jussi Auvinen¹, Rasmus Back¹, Jeanette Heidenberg¹,
Piia Hirkman^{2,3}, and Luka Milovanov^{2,4}

¹ Oy L M Ericsson Ab, Telecom R&D
Lemminkäisenkatu 14-18C, FIN-20520 Turku, Finland
Name.Surname@ericsson.com

² Turku Centre for Computer Science – TUCS, Åbo Akademi University
Lemminkäisenkatu 14 A, FIN-20520 Turku, Finland
Name.Surname@abo.fi

³ Institute for Advanced Management Systems Research

⁴ Department of Computer Science

Abstract. Besides the promise of rapid and efficient software development, agile methods are well-appreciated for boosting communication and motivation of development teams. However, they are not practical “as such” in large organizations, especially because of the well-established, rigid processes in the organizations. In this paper, we present a case study where a few agile practices were injected into the software process of a large organization in order to pilot pair programming and improve the motivation and competence build-up. The selected agile practices were pair programming, the planning game and collective code ownership. We show how we adjust these practices in order to integrate them into the existing software process of the company in the context of a real software project.

1 Introduction

Agile methods hold the promise of rapid and efficient software development. Reports from industry [1, 2, 3, 4], research [5, 6, 7] and educational [8, 9, 10] settings describe positive experiences of agile practices. While agile software development responds to the challenge of change, people is often stated to be one of its main focal points [11]. Also, issues related to individual agile practices, such as knowledge building [12], have been found alluring.

However, agile approaches also have their limitations and recommended application areas, as software development methods usually do. One of these issues is that many agile methods are best suited for small and medium projects [13]. For example, Extreme Programming does not easily scale for large projects [14]: all of the developers simply cannot work together in one big room.

Regardless of project size, the interest towards agile approaches rises to a great extent from the same needs, but the actual implementation is different. It requires much more tailoring in large companies than in smaller ones [15]. The challenge lies in fitting agile methods in existing processes where software development is only a small part

of the product development process. The question is whether corporations with well established and rigid processes can use just a few agile methods and still see significant benefits. Beck's discussion of the 80/20 rule would suggest not, as implementing all of the principles and practices creates synergy benefits [14], but experiences in practice have tried to prove otherwise [15, 16]. This paper presents an account of how agile methods were assessed in a case study for a large organization.

The pilot project was conducted at Ericsson, the largest supplier of mobile systems in the world. Ericsson's customers include the world's 10 largest mobile operators and some 40% of all mobile calls are made through its systems. This international telecommunications company has been active worldwide since 1876 and is currently present in more than 140 countries. The pilot project was conducted at a design department at Ericsson Finland.

This paper proceeds as follows: Section 2 presents the background and drivers for the pilot. Section 3 states the goals for the pilot and describes the means to achieve these goals. In Section 4 we describe the agile practices selected for and implemented during the pilot, while in Section 5 we present the results of the pilot in the context of the goals stated in Section 3. We discuss open issues in Section 6 and present our conclusions in Section 7.

2 Background

In early spring 2004, the design department in question arranged a workshop where different improvement areas for the software design process were identified. This workshop was a part of the continuous Software Process Improvement (SPI) [17] activity performed at the company. One of the identified areas was the motivation of the employees, and an SPI team was assigned to come up with innovative improvement proposals for this area. The results of a survey conducted within the company indicated that job satisfaction could be enhanced by changing the way the work was assigned, arranged and carried out. The organization would benefit from increasing the employees' motivation by promoting shared responsibilities among the designers, and increasing their competence in different areas of the large software systems they are working with.

An investigation of potentially suitable methods was conducted. The SPI team found that pair programming, an agile software development practice, could promote learning and shared responsibility, and hence increase motivation. The SPI team also noted that the chosen methods would need to be easy to implement within the existing process and easy to learn. Furthermore, such methods should be flexible enough in order to be changed and adapted to the standard process of the company to keep its integrity and strict deadlines. The SPI team considered an agile approach to be well-suited for the purpose.

The SPI team presented its ideas and results to the management in the summer of 2004. Based on this proposal, the management decided to pilot pair programming together with a number of other agile practices at the design department. The pilot was to be done in a real, live project so that the experiences from the pilot would be directly applicable in the organization. Since agile practices were new to the department