Applying Formal Methods to Process Innovation

Antonella Santone and Maria Tortorella
Dipartimento di Ingegneria, University of Sannio, Benevento, Italy
{tortorella,santone}@unisannio.it

Abstract. Continuous improvement of the software processes, including both production and maintenance, is a necessary condition to ensure fundamental software quality. An improvement opportunity comes from the availability of innovative process components to be integrated in a working software process. Formal methods are more and more used for modeling and verifying software systems. In this paper formal methods are instead used for modeling an innovative process component and understanding its integrability degree in an operative software process.

Keywords: Formal methods, CCS, process innovation, process improvement, process modelling.

1 Introduction

Continuous improvement of the software processes, including both production and maintenance, is a necessary condition to ensure fundamental software quality [21]. Software quality problems are widely acknowledged to affect the development cost and time. To mitigate these problems, much attention has to be paid to developing approaches, models and standards for Software Process Improvement (SPI). However, the number of organisations that have systematically adopted strategies to process improvement is only a part of the entire population of the software organisations [13]. This is partly due to the complexity and length of the timeframe needed for reaching a certain recognised level with reference to the chosen models, for example CMMI or Spice. Therefore, an SPI approach is often considered an expensive undertaking. In addition, political pressures often focus more on obtaining a specific level than creating actual improvements. All this creates demotivations in facing software process improvement tasks [10]. Nevertheless, improving software processes is an important task to be performed within a software organisation for avoiding that innovative initiatives are individually undertaken from the software engineer without recognising and codifying them in the official software process.

The aim of this paper is to propose an approach to improvement through the integration of innovative process components within the software process to be improved. This integration has to guarantee the operativeness of the software process. In this paper, the term process component is used in a very generic manner and it can indicate: a guideline; a technique; a method; or a simpler process.
This paper faces the problem of exploiting process components already defined and proposed in literature. In particular, an approach is proposed for modelling and characterising a process component with the aim of using it in an operative context. The idea follows the work presented in [15] where a process characterisation framework, divided in specification and evaluation sections, was defined. The technique proposed in this paper is based on formal methods is proposed. Once the process component to be innovated is identified, the aim is to provide a support for: simulating the behaviour of the old process; checking the integrability of innovative process components; and verifying that properties and behaviour of the innovated process are held.

Using formal description techniques for process innovation permits having an unambiguous description of the analysed software processes. First, a methodology to specify software components is introduced based on the Calculus of Communicating Systems (CCS) [8]. Afterwards, equivalence notions are used for establishing whether a process component can be replaced by an innovative one and a temporal logic [14] is used for describing properties representing the innovation.

The paper is organised as follows: Section 2 discusses concepts regarding process improvement and formal methods; Section 3 shows the core of the approach; while Section 4 presents an application of the methodology through a simple example. Finally, considerations and conclusions are given in Section 5.

2 Preliminaries

The improvement of a software process may concerns: the extension of an existing process by introducing one or more innovative process components; the replacement of one or more components of the process by semantically equivalent innovative ones; or the modification of one or more software process components.

The above modalities also include building a new software process, where the constituting process components will be integrated by extending each other.

Four tasks can be considered for improving a software process:

1. identification of the improvement goals and innovative characteristics of the process component to be integrated in the process [26];
2. identification of the process components to be substituted or added for attaining the preset improvement goals;
3. definition and/or identification of the process components to be integrated;
4. quantitative assessment of the quality and integrability of the process components identified, and of the risks their adoption may entail.

This paper is mainly focussed on task 3 and faces the problem of exploiting process components already proposed in literature. In fact, the richest sources of innovative process components is the specialist literature in the Software Engineering field, reports of research projects, tools and documentation provided by software houses, information available on the Internet, etc. It is needed to