The Usefulness of CSCW Systems in Process-Sensitive Software Engineering Environments

Rafael Duque and Crescencio Bravo

Escuela Superior de Informática
University of Castilla – La Mancha
Paseo de la Universidad 4, 13071 Ciudad Real Spain
{Rafael.Duque, Crescencio.Bravo}@uclm.es

Abstract. Software creation, development and maintenance are activities that require a great quantity of cooperative work. In the last years, several languages to model the diverse software processes have been proposed. However, these software Process Modelling Languages (PML) have not ended up being standardized nor being implanted by a majority in industry. The so-called Process-sensitive Software Engineering Environments (PSEE) are software engineering environments with support to manage PML. Each PSEE proposed has a PML associated, so there is a clear dependence between the software engineering environment and its modelling language. In this article, the cooperative support that a PSEE should offer is analyzed, and an environment with support for software PMLs is presented. This environment provides workspaces for distributed synchronous collaboration where work teams can develop a software project.

Keywords: CSCW, PSEE, PML, Software Engineering, Cooperative Engineering.

1 Introduction

The statement "Software Processes are software too" [17] made by Osterweil in 1987 has raised an important number of investigations and initiatives around the concept of Software Process. Among these, we can mention the so-called software Process Modelling Languages (PML) as an attempt to represent software processes in all their extension (activities, roles, products, tools, people) [9]. In the last years, some proposals have been made in order to model software processes. However, these proposals have not been adopted as an industry standard and they have not reached a predominant position in the market as a consequence of their use. The software engineering environments with support for the creation and manipulation of software processes models are called Process-sensitive Software Engineering Environment (PSEE).

A PSEE should offer cooperation support to the users who participate in a same software development project. This is an essential point to overcome the difficulties that usually arise due to the complexity inherent to software processes, since software applications are difficult to develop and test. Software applications frequently exhibit unexpected behaviours and undesired results that can originate important problems. This is, above all, due to the fact that software development is usually directed by exceptions and is subject to frequent changes of requirements. Consequently, software

Y. Luo (Ed.): CDVE 2006, LNCS 4101, pp. 156 – 163, 2006.

processes are unpredictable and difficult to model because of the many peculiarities of each process. A high level of cooperation among all the participants of the project is required in order to reduce these problems. Therefore, a basic requirement of PSEEs is the support for cooperative work. Moreover, the increase in the use of the Internet and the proliferation of wireless technologies make groupware [13] an interesting application field in the software development processes.

It is also necessary to take into account the high level of cooperation that is required among the people that participate in specific activities of the software development life cycle because this has an influence in terms of costs. Concretely, it is considered that 70% of the time and effort of developers is used in cooperating with other developers, which, translated into economic terms, results in 85% of the total costs [19].

Some examples of groupware tools that can be useful in different phases of the software development are shown in Table 1. For example, shared editors can be helpful in the creation of source code by several programmers, and Group Decision Support Systems (GDSS) can be used by clients (users) and developers to identify and elicit requirements.

Table 1. Groupware for different software development phases

Software development phase	Groupware
Requirements collection	Group Support Systems (GSS) Group Decision Support Systems (GDSS)
Design specification in a high abstraction level	Synchronous systems Distributed electronic meeting systems
Design specification in a low abstraction level	Face-to-face meeting systems
Programming	Shared editors
Installation	Synchronous electronic meeting systems

This article shows how SPACE-DOMAIN [5] can offer a collaborative support in software processes. This system supports software process modelling as well as the realization of specific tasks within a software production life cycle. Taking into account the importance of cooperative work in both functionalities, SPACE-DOMAIN provides shared workspaces for distributed synchronous collaboration. The software processes and the modelling tools (for specific tasks) are defined using system-independent languages based on XML to achieve interoperability.

In the next section we analyze the most significant PSEEs that approach the integration of tools with support for cooperative work. In Section 3, the SPACE-DOMAIN environment and the tools it integrates are described. Finally, we draw some conclusions and show the research work lines that we aim at approaching in the future.

2 Related Work

Traditionally, software engineering environments have lacked support for people communication and for process modelling. The main systems that have tried to bridge this gap have been Spade and Serendipity.