Infrastructure for SE Experiments Definition and Planning

Arilo Cláudio Dias Neto, Rafael Ferreira Barcelos, Wladmir Araújo Chapetta, Paulo Sérgio Medeiros dos Santos, Sômulo Nogueira Mafra, Guilherme H. Travassos
PESC – COPPE/UFRJ
{acdn, barcelos, wladmir, pasemes, somulo, ght}@cos.ufrj.br

Abstract

This position paper describes an infrastructure to support activities in Experimentation Process´s Definition and Planning phases. The purpose of this infrastructure is to provide a framework to allow the creation and filling of Experimental Plans during Experimentation Process enactment.

1. Introduction

No science can advance without experimentation and measurement [1]. Progress in any discipline involves building models that can be evaluated, through experimental study, to check whether the current understanding of the field is correct [2].

The Experimentation in Software Engineering has been the way that researchers have trying to make Software Engineering "more like science" and "less like art" through researches validation, knowledge accumulation and experience exchange. To allow this exchange, the information about experimental studies must be gathered and made available for consulting by the Software Engineering Community.

Nevertheless, the experiments execution, especially in Software Engineering, is a complex task, spending time and producing an amount of information and knowledge that are difficult and complex to manage [3].

Therefore, it is necessary the definition of an infrastructure to build experiments knowledge base, where experimental studies can be defined, planned, executed, analyzed and packaged, and the results stored and available. This infrastructure must control the experimental study information acquisition throughout the Experimentation Process execution.

Therefore, this work introduces a proposal for such infrastructure to support the execution of Software Engineering Experimentation Process activities.

To experiment our ideas, a limited scope of the whole SE experimentation process had been chosen. Just the Definition and Planning phases [4,5] are being considered. Despite their simplicity, these phases introduce more study risks and are time consuming, with a lot of clerical activities associated to them.

This position paper is organized in the following manner: in Section 2 we textually describe the infrastructure, including design decisions, restrictions and its components. Some final considerations and on-going works can be found in section 3.

2. Infrastructure for SE Experiment Definition and Planning

The main artifact produced by the end of Definition and Planning phases is called Experimental Plan [6]. It contains all information necessary to run and analyze the experimental study. Besides, some instructions about how the next experimental study should be replicated.

To justify our project decisions, and make possible the beginning of infrastructure development from its main components, we describe how we understand the accomplishment of the Experimental Plan creation and filling.

2.1 Experimental Plan Creation and Filling

During the Experimentation Process, some artifacts must be created, among them, the Experimental Plan. Usually, these documents can be built by using some sort of document model (templates), which can guide the experimenter in filling in throughout the experimentation process activities. Thus, the filling information order must follow that one defined by its corresponding template (what also defines a filling process). To exemplify, Travassos et al. [6] show an example of an Experimental Plan created by following a specific experimentation process.

Based on these issues, the infrastructure must focus on the Experimental Plan filling, providing initially the following functionalities: (1) make available knowledge to aid the correct filling of the Experimental Plan; (2) provide the filling mechanisms of the Experimental Plan trough well-defined process; (3) provide visualization mechanisms to show the information contained in the Experimental Plan.

2.2 Additional Requirements

Before modeling the idealized infrastructure's architecture, some additional requirements were established aiming a scope reduction. The requirements are the following ones:

(1) To adopt Web-based characteristics for the system's construction, due to the access and publication facilities; (2) To use e-services and

external components to support tasks in the Experimentation Process, like tools to fill in or visualize a respective piece of information from an Experimental Plan; (3) To allow the flexibility and independence between the conceptual representation of the expected artifacts, related to a filling process, and their computational representation, by abstracting Association Maps. According to our proposal, there are different specialized services for each different computational representation. For computational representation, we mean artifacts media and formats, like xml files, textual files, videos etc.
Based on these requirements and on the infrastructure's identified functionalities, the infrastructure architecture was defined and a prototype has been built to evaluate feasibility of such concepts.

2.3 Infrastructure: Main Components

The defined architecture is composed by three components.

The first component, called *Meta-Configurator*, has the functionalities to define and configure basics elements which will be used by the others components. These basics elements are: (1) *Document model*, the basis for artifact's creation produced during an Experimentation Process. In the scope of this position paper, a document model is a template for the experimental plan; (2) *Process model*, the basis for the definition of a process used for the Experimental Plan's filling; (3) *Configured service*, an information set related to e-services or external tools which can be used as services for supporting the filling in or visualization of the Experimental Plan´s information.

The second component, called *Instantiation Environment*, was defined with the purpose of providing functionalities to instantiate an environment which allows experimental studies planning, based on a *document model* and *process model*, defined by the first component functionalities. To reach this objective, some tasks have to be accomplished: (1) to define the relationship between a process model that will be followed during the planning realization, a document model that will be filled in by the researcher and the configured services set that will be used as tools to realize tasks trough an *Association Map;* (2) based on the selected map, this environment allows the instantiation of an *Execution Environment.*

The *Execution Environment* (third component) is responsible for providing facilities to accomplish the monitoring of the activities related to an experimental study planning. This monitoring is realized through the defined process, respecting the access control to the information and process activities. The access to the Experimental Plan's information is made by configured services for each process' artifact, where the Association Map is responsible to define this access. The basis for its working is the enactPro [7], a CASE tool that supports software engineering processes enactment and control, specifically amended to support this infrastructure.

3. Final Considerations and Future Works

This position paper described the current state of an infrastructure to support activities in Experimentation Process's Definition and Planning Phases. We have built a prototype (Zope platform, C++) to evaluate the infrastructure's development feasibility.

The idealized solution has been developed in a generic way, allowing that others specifics Software Engineering's processes could be supported by the infrastructure. In the ESE[1] team context, this solution makes part of an on-going research scenario concerned with Experimental Software Engineering.

Acknowledgements

We would like to thank FAPEAM and CAPES by the financial support. Dr Travassos is granted by CNPq.

References

[1] S. L. Pfleeger, "Albert Einstein and Empirical Software Engineering", IEEE Computer, 0018-9162/99, 1999.

[2] V.R. Basili, F. Shull, F. Lanubile, 1999, "Building Knowledge through Families of Experiments", IEEE Transactions of Software Engineering, vol. 25, N°. 4.

[3] F. Shull, J. Carver, G.H. Travassos, 2001, "An Empirical Methodology for Introducing Software Processes", 8th European Software Engineering Conference (ESEC) and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-9), Viena. Proceedings of the 8th ESEC - 9th ACM SIGSOFT FSE. ACM Press, 2001. p.288 – 296.

[4] C. Wohlin, P. Runeson, M. Host, B. Regnell, A. Wesslen, , 2000, "Experimentation in Software Engineering: An Introdution", Kluwer Academic Publishers, Boston, MA.

[5] E. G. Amaral., 2003, "Empacotamento de Experimento em Engenharia de Software", Dissertação (Engenharia de Sistemas e Computação) - Universidade Federal do Rio de Janeiro.

[6] G.H. Travassos; D. Gurov; E.A.G.G. Amaral, "Introdução à Engenharia de Software Experimental", Relatório Técnico ES-590/02-Abril, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ.

[7] S. N. Mafra, M. O. Barros, G. H. Travassos. enactPro: Automatizando Processos de Software. In: 18 SBES – Sessão de Ferramentas, 2004.

[1] http://www.cos.ufrj.br/~ese