Empirical Studies on Framework-Based (Re)-Engineering

Maria Istela Cagnin\(^1\*\), Rosana T.V. Braga\(^1\), Rosângela Penteado\(^2\), José Carlos Maldonado\(^1\), Paulo Cesar Masiero\(^1\), Fernão S.R. Germano\(^1\)

\(^1\)Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo
\(^2\)Departamento de Computação – Universidade Federal de São Carlos

\{istela, rtvb\}@icmc.usp.br; rosangel@dc.ufscar.br; \{jcmaldon, masiero, fernao\}@icmc.usp.br

Abstract

This paper presents our experience with experimentation using frameworks, as well as the perspectives we have to apply it in the near future. We also discuss insights and lessons learned from our experience in this field.

1. Introduction

Few works can be found in the literature, concerning experimentation in reverse engineering and reengineering. However, several works have been presented in the areas of software engineering [3, 5, 7], which show the importance of having controlled experiments to statistically validate proposed techniques and also their applicability in different contexts and environments. Braga et al. [3] and Shull et al. [7] present well succeeded results of experiments using frameworks to support software engineering. In Juristo and Moreno [5] there are reports about experiences obtained through experiments conducted recently in the software engineering area, both in the academic and in the industrial environment.

This paper has the goal of presenting our experience with experimentation using frameworks, as well as to discuss some new perspectives regarding experimentation, mainly in the context of framework-based reengineering.

2. Our Experience

Our group at ICMC, University of São Paulo, has developed the GRN pattern language [1], composed of fifteen analysis patterns for the business resource management domain. This pattern language guides the analysis of systems pertaining to this domain, so the result of its application is an analysis model of the target system. A framework, named GREN, was built to support the development of systems based on GRN, together with an instantiation tool (GREN-Wizard) [2] to automate the instantiation process. A series of experiments were conducted to evaluate GRN, GREN and GREN-Wizard. We have used the structure proposed by Wholin et al. [8] to plan and document our experiments, and it has shown to be satisfactory for our purposes.

The first three experiments aimed at comparing the quality of the analysis models produced using the GRN pattern language compared to the analysis models produced by using an “ad hoc approach” [3]. They were performed with different kinds of subjects: undergraduate students, graduate students, and information technology professionals. The task assigned to the subjects was to model two different systems (but with similar complexity levels) in two phases, each of which using a different approach. The first approach, named “ad hoc approach”, consisted of modeling a system using only their own knowledge about object orientation, supported by the UML notation and by an “ad hoc process” for modeling. The second approach, named “GRN approach”, consisted of modeling a system using the GRN pattern language and its corresponding process for system analysis. The results have shown that the analysis models produced using the GRN pattern language have less errors than those produced using an ad hoc system analysis approach. Another point observed during the experiment was that the usefulness of using GRN is influenced by the experience of its users, i.e., there seems to be a trend that GRN is more useful to inexperienced developers than to those that already have some practice in the domain. The experiments have also helped us to identify several problems in the GRN pattern language and its documentation. This allowed the modification of GRN patterns to improve them, easing their use in subsequent experiments.

Our second series of experiments involved comparing the effort to instantiate the GREN framework using an “ad hoc approach” versus using a “GRN approach”. In these experiments, GREN was manually instantiated, i.e., the GREN-Wizard tool was not used. The “ad hoc approach” consisted of using examples as a basis to the instantiation process, which is the approach...
that has shown the best results in an experiment conducted by Shull et al. [7]. In that experiment, Shull had compared the instantiation based on examples with the instantiation based on the study of the framework class hierarchy and documentation. The “GRN approach” consisted of basing all the instantiation process in the GRN pattern language, i.e., the developer needs only to know the patterns applied and the roles played in each pattern. The results were not statistically conclusive, but we could learn several lessons and gain insights from it. We could be sure that the GRN approach was feasible, as the students could follow it without problem and spending about the same time as with the “ad hoc approach”. Some students gave a testimony that the GRN approach was more systematic, even though it led to less knowledge about the GREN framework structure. One of the results obtained by Shull et al [7] was confirmed by our experiments: participants that use the “ad hoc approach” have a strong trend to include in their systems functionality that was not mentioned in the requirements, just because the functionality is available in the example and is easy to include.

Finally, a third series of experiments was done to assess the usability of GREN-Wizard. They were not comparative, but only observational experiments, and could help us to identify problems with the GUI and the sequence of patterns applied, among others.

3. Perspectives

Our experience with empirical studies is certainly positive, as they allowed us to learn more about our products in order to be able to improve them. Some problems we have faced concern the choice of participants, as it has been difficulty for us to apply the experiments out of the academic environment. Students are subjects that participate because they need to get their grades, so we are not sure if the results would be the same in an industrial environment, in which other factors are present (e.g., schedules and costs).

We are planning a laboratory package aiming at applying experiments based on the use of frameworks during reengineering, in an industrial environment, to observe the problems that have occurred in the academic environment. The experiment has the goal of evaluating the agile process of framework-based reengineering, called PARFAIT [4]. The qualitative focus is the effectiveness of the process under the perspective of software engineers interested in software reengineering. The PARFAIT effectiveness will be observed in relation to the “ad hoc” reengineering process. The hypothesis formulation will be based on the reuse level of analysis, design and implementation during the reengineering, on the quality of the resulting product, and on the efforts and time spent in both processes. The experiment planning will be done in blocks, using two techniques in two reengineering projects. Techniques and projects will be exchanged between groups. Groups will be chosen so that they have the same number of persons and the same level of knowledge and experience.

References