Evaluating the Effort of Integrating Feature Models: A Controlled Experiment

Vinicius Bischoff, Kleinner Farias, Lucian José Gonçales
Post Graduate Program on Applied Computing (PPGCA)
University of Vale do Rio dos Sinos (Unisinos)
São Leopoldo, Brazil
viniciusbischof@edu.unisnos.br, kleinnerfarias@unisinos.br, lucianj@edu.unisinos.br

***Abstract*—The integration of feature models plays a key role in many tasks in software development, such as evolving Software Product Lines (SPL) to add new features. However, based on our experience in previous empirical studies, one of the main shortcomings to the widespread adoption of integration techniques is the lack of empirical knowledge about its effects on the effort of analysts and developers. This problem applies to integration techniques involving a set of operations (union and intersection) as well as the relationships between features and their elements. This article, therefore, reports on a controlled experiment that investigated the effort of (1) applying the integration techniques of feature models by professionals and students, and (2) detecting and resolving inconsistencies in the output-integrated models. The integration effort was evaluated through 10 evolution scenarios. The main results suggest that there is no significant difference regarding (1) the integration effort invested by professionals and students to produce a desired integrated model, and (2) the correctness rate of the integrations performed by professionals and students.**

Keywords - Feature Model; Integration; Experimental Study.

I. INTRODUCTION

Feature models (FM) can be seen as a "big-picture" of the functionalities of a software system. The integration of feature Models plays a pivotal role on software engineering tasks. For this, each developer performs tasks, such as changing or adding new features in a specific feature delta model, FM_B. These changes are often performed in parallel, and then each developer accommodates these changes into a base feature model, FM_A. Developers need to integrate these modifications to update the "big picture" of a software system. Specifically, integration of feature models might be briefly defined as a set of activities that should be performed over two input models, FM_A and FM_B, to produce a desired output-composed feature model, FM_{AB}.

However, developers may end up not producing the FM_{AB}. Instead, developers often produce an output-composed model, FM_{CM}, with problems (i.e., $FM_{CM} \neq FM_{AB}$) [1][11]. This happens because developers are usually unable to properly detect and resolving integration problems, such as conflicts and inconsistencies, given the problem at hand. Hence, to produce the FM_{AB} they must invest effort to resolve such conflicts and inconsistencies in the FM_{CM}. Conflicts are contradicting information found in FM_A and FM_B. In other words, conflicts are different values assigned to the properties of feature models (e.g., name). For example, the variability property of a given feature *Researcher* defined as *mandatory* in FM_A, while in FM_B its variability property is defined as *optional*. These contradicting values assigned to these specific features represent a conflict that developers must resolve. However, if this issue is not properly resolved, inconsistencies are inserted into the output-composed feature model, FM_{CM}. For example, *Researcher* variability property defined as *optional* denotes an inconsistency as the expected value should be *mandatory*. Model inconsistencies [10] can be briefly defined as a mismatch between FM_{CM} and FM_{AB}.

Previous works already investigated the effects of composition tasks on developers' effort, and their experiences [1][2]. In [1], the authors evaluated the effort invested to compose UML models using specification-based and heuristic-based techniques.; however, the integration of features models was not explored. In [2], the authors evaluated the impact of experience level on comprehension of C++ lambdas functions. Integration of feature models was not also the focus of the authors. To sum up, none of them investigated the effects of integration tasks of feature models on developer's effort. Also, there is a lack of empirical evidence regarding the effort of software developers on integrating feature models.

To account for this, this work conducts a controlled experiment to analyze the effort that developers invest on activities related to the integration of feature models. In particular, we seek to explore the effort invested by two categories of participants, including students and professionals. This experiment was executed based on well-defined guidelines [6]. In [1], the authors argue that this kind of study is important because it provides scientific evidence about the developer's performance on software engineering tasks. This prevents the development team's decisions limited to only on opinion of experts and evangelists, thus providing strong empirical evidence.

II. STUDY METHODOLOGY

A. Objective and research questions

The objective of this work is to *analyze* the integration techniques of feature models *for the purpose of* investigating *with regard to* effort and correctness *from the perspective of* students and professionals *in the context of* evolution of feature models. This objective is based on the GQM template [5].

DOI reference number: 10.18293/SEKE2018-202

Based on this objective, two Research Questions (RQ) are formulated:

- **RQ1:** What is the effort required to integrate FMs?
- **RQ2:** What is the rate of correctly integrated FMs?

B. Hypothesys formulation

This Section formulates the hypotheses that guide our experiment to answer the respective two formulated research questions. These hypotheses are described below:

H1. Null Hypothesis 1, ($H_{1\text{-}0}$): Professionals apply less or equal effort to integrate FMs (IE) manually than students.

$$H_{1\text{-}0}: IE_{professional}\ (FM_A, FM_B) \leqslant IE_{student}\ (FM_A, FM_B)$$

H2. Hypothesis Null 2, ($H_{2\text{-}0}$): The rate of correctly integrated elements (CIR) performed by professionals is equal or greater than one produced by students.

$$H_{2\text{-}0}: CIR_{professional}\ (FM_{CM}) \geqslant CIR_{student}\ (FM_{CM})$$

As in any experiment, the main objective is to reject these null hypotheses. In the context of this work, we conjecture that the professionals presented better results compared to the students.

C. Study variables

The dependent variable of the first hypothesis (H1) is Integration Effort (IE). The IE represents the time (in minutes) spent to integrate two input-feature models, FM_A and FM_B, to produce FM_{CM}. The dependent variable in the second hypothesis (H2) is the Rate of Correctly Integrated Features (CIR). CIR is a correctness rate. The CIR formula is the result of the number of participants who correctly answered the investigated question (NPAC), divided by the total number of participants (NPT), i.e., CIR = NPAC/TNP.

The independent variable of this experiment is the experience level of participants, which can assume two values: Students and Professionals. Professionals are active persons on software industry, while students are persons that studies in universities. Therefore, students are organized in tree groups, i.e., technical, graduate and postgraduate.

D. Context and partipants

The context of this study is related to the evolution of feature models. This means that, users must properly integrate the changes on a delta model FM_B into the base model FM_A. Therefore, the participants must choose the right answer among five options. 10 Evolution Scenarios (ES) were developed to evaluate the integrations.

A total of 25 participants attended this experiment. The professionals group contains 07 persons. The student group contains 05 undergraduate students; 03 graduate students, and 10 students from IT courses.

E. Experimental process

The experimental process consists of three steps: (1) Training; (2) Execution of feature integration activities; and (3) Participant Background and Data Collection. In the first step (1) all participants were trained to ensure that they acquired the necessary familiarity with model integration techniques. In the next step (2), developers concerned on feature integration tasks, i.e., participants analyze the input models (FM_A and FM_B) of each scenario based on descriptions of changes. In this step, they also resolved conflicts. Participants should resolve conflicts according to the change requests listed in each question to produce a composed model, FM_{CM}. Finally, they tried to produce the desired feature model. This activity consists of integrating the models, i.e., producing the FM_{AB}. In the last step (3), the participants provided background information such as their professional experience, graduate level, level of experience on software modeling and development. Finally, all produced data related to the experiment were collected.

F. Analysis procedures

Quantitative analysis. We performed descriptive statistics to analyze their normal distribution and statistical inference to test the hypotheses [6][7]. Our analysis was performed to test the hypotheses in both groups in all experimental tasks. We applied the *Student's t-test* to validate the hypotheses intrinsic to this research to check the normality of the variables, the *Kolmogorov-Smirnov - (Lilliefors)* test, which is a broad test of the distribution function of at the same time [6][7]. Although the data distribution is subdivided into treatment (groups), the validity hypothesis refers only to the group (professionals and students), however, an individual evaluation of the categories will be presented.

III. STUDY RESULTS

This Section presents the results regarding the investigated research questions. Section III.A presents the results in relation to the RQ 1 that investigates the effort on integration techniques. Section III.B presents the results in relation to the RQ2 that investigates the influence of experience level on the rate of correctly integrated feature models. Finally, Section III.C presents some additional observations.

A. Effort and integration techniques

Descriptive statistics. This section discusses the descriptive statistics regarding the impact of experience level (professional and students) on the effort on integrating feature models (IE). Table 1 shows the descriptive statistics of the collected data. Group 1 shows that university students (GR and PG) apply less effort to integrate feature models, i.e., on average the effort is 1.88 minutes. Specifically, they applied 25.24% less effort to integrate the FMs in relation to technical students. In Group 2, University Students (Graduates and postgraduates) also spent less effort compared to professionals to integrate feature models. The effort is 1.88 min., which represents 2.09% less than professionals to integrate the FMs. In Group 3, professionals spent less effort to integrate features than technical students. Specifically, industry professionals applied average of 1.92 min. integrating features, i.e., 23.64%, less effort to integrate the FMs in relation to technical students. Finally, in Group 4, that is the general comparison between

Table 1. Descriptive statistics and hypothesis tests.

Group 1 Technical Students vs University Students	**Variables**	**Treatment 18 participants**	**SD**	**Min**	**25th**	**MD**	**75th**	**Max**	**Avg.**	**% Diff**	**t-test p-value**
	CIR	TECH	0.18	0.30	0.40	0.60	0.70	0.80	0.55	6.78	0.0716
		GR and PG	0.29	0.13	0.25	0.69	0.63	1	0.59		
	IE	TECH	0.58	1.61	2.33	2.65	2.67	3.57	2.51	25.24	0.015
		GR and PG	0.48	1	1.40	1.94	1.67	2.50	1.88		
Group 2 Professional vs University Students	**Variables**	**Treatment 15 participants**	**SD**	**Min**	**25th**	**MD**	**75th**	**Max**	**Avg.**	**% Diff**	**t-test p-value**
	CIR	PRO	0.23	0.14	0.20	0.43	0.43	0.86	0.44	26.17	0.197
		GR and PG	0.29	0.13	0.25	0.69	0.63	1	0.59		
	IE	PRO	0.71	1	1.33	1.71	1.67	3.40	1.92	2.09	0.884
		GR and PG	0.48	1	1,40	1.94	1.67	2.50	1.88		
Group 3 Professional vs Technical Students	**Variables**	**Treatment 17 participants**	**SD**	**Min**	**25th**	**MD**	**75th**	**Max**	**Avg.**	**% Diff**	**t-test p-value**
	CIR	PRO	0.23	0.14	0.20	0.43	0.43	0.86	0.44	20.91	0.276
		TECH	0.18	0.30	0.40	0.60	0.70	0.80	0.55		
	IE	PRO	0.71	1	1.33	1.71	1.67	3.40	1.92	23.64	0.054
		TECH	0.58	1.61	2.33	2.65	2.67	3.57	2.51		
Group 4 (General) Professional vs Students	**Variables**	**Treatment 25 participants**	**SD**	**Min**	**25th**	**MD**	**75th**	**Max**	**Avg.**	**% Diff**	**t-test p-value**
	CIR	PRO	0.23	0.14	0.20	0.43	0.43	0.86	0.44	22.81	0.146
		TECH, GR and PG	0.24	0.43	0.33	1.29	0.67	1.80	0.57		
	IE	PRO	0.71	1	1.33	1.71	1.67	3.40	1.92	12.39	0.274
		TECH, GR and PG	0.53	1.31	1.86	2.30	2.17	3.04	2.19		

Legend: Standard Deviation (SD), Minimum (Min), First Quartile (25^{th}), Median (MD), Third Quartile (75^{th}), Maximum (Max), Average (Avg.), Percentage Difference (% Diff), Correct integration rate (CIR), Integration effort (IE), Technician (TECH), Graduate (GR), Postgraduate (PG) and Professional (PRO).

professionals and students, shows that professionals spent on average 1.92 min. to integrate feature models, i.e., 12.39% less effort to integrate feature models in relation to students. Therefore, professionals tend to invest less effort to produce FM_{AB} using manual integration techniques.

Testing hypotheses. We also performed statistical tests to evaluate whether the measures of *Effort*(FM_A, FM_B), *eff* (FM_A, FM_B), *diff*(FM_{CM}, FM_{AB}), and *iff*(FM_{CM}) are statistically significant. We hypothesize that professionals in relation to students tend to require less effort than their counterparts. According to the hypothesis test previously described $H_{1\text{-}0}$, the t-test failed to reject the null hypothesis, with the p-value is 0.95. Therefore, there is no significant difference between the applied effort to integrate features between professionals and students.

B. Correctness and integration techniques

Descriptive statistics. This section analyzes the collected data regarding the impact of integration techniques on the correctness rate (CIR). For this, we also calculated descriptive statistics to understand the distribution of the data, see Table 2. It was possible to verify in Group 1 that the CIR (rate of correctness) is higher for GR and PR (undergraduate and graduate students), which reached an average of 0.59 of correct answers, i.e., 6.78% more successful than technical students when integrating the FMs. In Group 2, professionals obtained a higher rate of correct answers in relation to the graduated and postgraduates students, i.e., an average of 0.59 correct answers, representing that university students were 26.17% less precise to integrate FMs. In Group 3, the correctness rate (CIR) is superior for the technical students, who reached an average of 0.55 correct answers, i.e., professionals were 20.91% less precise when integrating the FMs. Finally, in Group 4 we can verify that the rate of correct answers is superior for the students, who reached an average of 0.57 correct answers, i.e., professionals were 22.81% less precise to integrating feature models.

Testing hypotheses. It evaluates the experience level in relation to the CIR (rate of correctly integrated models). The rows identified with CIR shows the statistic p-values for comparisons between groups. The results show that t-test (T) rejects the null hypothesis $H_{2\text{-}0}$, with the p-value equal to 0.146. This means the level of experience does not have correlation with the rate of correctly integrated features. Specifically, the hypothesis test failed to reject the null hypothesis.

C. Additional observations

All participants in this research have previously submitted to a small training with 15 minutes to explain what a feature is, how it behaves, what existing relationships, and ultimately examples of integration.

Academics tend to be more prepared than professionals with respect to the application of modeling techniques. As understood in the company surveyed professionals perform short meetings with the tasks to be developed. They do not follow models, only requirements described in their documentation. This way, we can extend the interpretations if they are going to undergo changes, since the documentation is not usually updated.

Considering our results, the average effort applied is two minutes per question, which implies in 20 minutes running the 10 questions. However, the degree of difficulty proposed for this research for integration between the FMs (syntactic and

semantic) is small, considering what is applied in the industry. This demonstrates the need for automation of integration techniques, as well as the possibility of working collaboratively between analysts and developers. To facilitate its visualization and the possible set of updates that is necessary. Another revealed fact refers to the corrected rate, which we believe can be improved with the application of a semiautomatic technique. In this way, indicating when any inconsistency occurs, so that the developer can make the decision that suits him best applied the Wilcoxon test and the t-test to check the H2-0.

IV. RELATED WORKS

The integration of feature models is a research field of interest in academia. The integration of features is important for composing software product lines [3]. Recently, the research initiatives focused on proposing techniques for features integration. However, there is a lack of experimental studies. In [3], several composition operators can be defined, depending on the combination strategies and semantic framework for developers and researchers to plan and carry out qualitative and quantitative research, as well as to reproduce and reproduce empirical studies. In [9], authors demonstrate how FMs can be reduced to propositional formulas or constraint satisfaction problems. Benefits are tools that propagate constraints (so that incorrect specifications can be detected automatically). Finally, this expected in Feature Models.

V. THREATS TO VALIDITY

Statistical validity. The independent and dependent variables were submitted to suitable statistical methods We minimized this threat by checking whether. We test all hypotheses considering the significance level at 0.05 level ($p \leqslant 0.05$). **Construct validity.** The measures applied in this study, i.e., the effort and the correctness are widely applied on controlled experiments on software engineering [1][8]. **Internal validity.** The dependent variables variated appropriately according to corresponding independent variables. **External validity.** Some aspects must be followed to reproduce the results of this study such as: participants must have the familiarity with feature integration models must have similar sizes, and the same variables must be collected.

VI. CONCLUSIONS AND FUTURE WORKS

This article reported on a controlled experiment that explored three points: application effort of integration techniques of feature models by professionals and students, and detection and resolution effort of inconsistencies found in output-integrated models.

Both hypothesis tests failed rejecting the null hypothesis. This means that has no significant difference on performance on both groups on integration of Features Models. However, overall results on descriptive statistics show that professionals tend to invest less effort to integrate feature models, but they produce integration with more errors than students. There are few studies that evaluate the effort required to use model integration techniques. Further empirical studies are still required to better understand whether these findings are confirmed or not in other contexts, considering other FMs, encompassing different evolution scenarios, and evaluating other integration techniques. Finally, we hope that the issues outlined throughout the paper encourage other researchers to replicate our study in the future under different circumstances and that this work represents a first step in a more ambitious agenda on better supporting feature model integration tasks.

ACKNOWLEDGMENT

Thank you to UNISINOS for the teaching and research environment in which they provided to support this research.

REFERENCES

[1] K. Farias, A. Garcia, J. Whittle, C. v. F. G. Chavez and C. Lucena, "Evaluating the effort of composing design models: a controlled experiment," *Software & Systems Modeling,* vol. 14, pp. 1349-1365, 2015.

[2] P. M. Uesbeck, A. Stefik, S. Hanenberg, J. Pedersen, and P. Daleiden. "An empirical study on the impact of C++ lambdas and programmer experience". *In Proceedings of the 38th International Conference on Software Engineering* (ICSE '16). ACM, New York, NY, USA, pp. 760-771, 2016.

[3] M. Acher, P. Collet, P. Lahire and R. France, "Composing feature models," em *International Conference on Software Language Engineering*, 2009.

[4] M. M. Alam, A. I. Khan and A. Zafar, "A Comprehensive Study of Software Product Line Frameworks," *International Journal of Computer Applications,* vol. 151, 2016.

[5] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical software engineering,* vol. 14, p. 131, 2009.

[6] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén. "*Experimentation in software engineering".* Springer Science & Business Media. 2012.

[7] B. W. Yap and C. H. Sim, "Comparisons of various types of normality tests," *Journal of Statistical Computation and Simulation,* vol. 81, pp. 2141-2155, 2011.

[8] A. Oliveira, V. Bischoff, L. J. Gonçales, K. Farias and M. Segalotto, "BRCode: An interpretive model-driven engineering approach for enterprise applications," *Computers in Industry,* vol. 96, pp. 86-97, 2018

[9] D. Batory, D. Benavides e A. Ruiz-Cortes, "Automated analysis of feature models: challenges ahead," *Communications of the ACM,* vol. 49, pp. 45-47, 2006.

[10] K. Farias, A. Garcia, C. Lucena, "Evaluating the Impact of Aspects on Inconsistency Detection Effort: A Controlled Experiment," In: 5th Int. Conf. on Model-Driven Eng. Languages and Systems, Vol. 7590, pages 219-234, 2012.

[11] K. Farias, L. Gonçales, M. Scholl, T.C. Oliveira, M. Veronez.. "Toward an Architecture for Model Composition Techniques," In *27th Int. Conf. on Software Engineering and Knowledge Engineering (SEKE'15),* pp. 656-659, 2015.