Journal of Computer Science

Original Research Paper

Comparing Approaches for Quality Evaluation of Software Engineering Experiments: An Empirical Study on Software Product Line Experiments

Viviane R. Furtado, Henrique Vignando, Victor França and Edson OliveiraJr

Informatics Department, State University of Maringá, Maringá-PR, Brazil

Article history
Received: 09-11-2018
Revised: 28-03-2019
Accepted: 23-04-2019

Corresponding Author:
Edson OliveiraJr
Informatics Department,
State University of Maringá,
Maringá-PR, Brazil
Email: edson@din.uem.br

Abstract: The Software Engineering (SE) research area must provide results of a certain quality for the sake of value. High quality research results may ensure experience and knowledge, which are essential for the technology to be transferred to the industry. One of the means to obtain such quality results is experimentation. Experimentation is a scientific method that aims to provide evidence of a theory over real-world observations establishing a cause-effect relation. Well conducted, auditable and repeatable experiments are vital for scientific evolution and novelty. Quality evaluation of controlled experiments and quasi-experiments in SE has been recently discussed in the literature as researchers desire to assess whether such experiments have improved by reporting information that enables the experiments to be replicated and the reader can understand the experiment and validate results. Thus, this work empirically compares four approaches for quality evaluation of SE experiments in the context of Software Product Lines (SPL). In addition, we are interested on verifying the quality of reporting experiments in a well-discussed reuse technique as SPL. The Pearson technique supported the correlation between pairs of evaluation approaches. In addition, the T-Test and Mann-Whitney-Wilcoxon U test were applied to the samples to verify whether there was a difference in the quality of experiments when using an experimental template. Preliminary results show a strong positive correlation between them, the hypothesis tests confirmed there is such a difference in quality when using experimental template and the SPL experiments report more the planning phase than the analysis and interpretation phase. Based on our results, we provide initial evidence two approaches are the best to reporting SPL experiments.

Keywords: Experiments, Quasi-Experiments, Quality Evaluation of Experiments, Software Product Line

Introduction

Experimentation in the Software Engineering (SE) area plays a central role at providing evidence of a certain theory in an objective, accurate and systematic way (Wohlin *et al*., 2012), as well as providing inductive support for hypotheses (Sjoberg *et al*., 2007) and decision making, aiding the comparison of different technologies, methods and tools (Kampenes, 2007).

Therefore, evaluating the quality of experiments and quasi-experiments (In this work we use the term "experiment" to denote both concepts of "experiment" and "quasi-experiment".) is essential for improving the means to carry out experiments (Kitchenham *et al*., 2012) towards development and sharing scientific knowledge and empirical decisions on software construction based on software engineering (Sjoberg *et al*., 2007).

The software engineering community has discussed how to evaluate the quality of experiments using approaches, such as: Simple quality criteria (Dieste *et al*., 2011), checklists (Kampenes, 2007; Kitchenham and Charters, 2007; Kitchenham *et al*., 2010), quality scales (Dieste *et al*., 2011) and inference validity and experiments reporting (Kampenes, 2007).

One of the points they discuss is the lack of consensus on the definition of quality in the research community. Thus, it has been suggested that quality is related to minimizing the bias and maximizing the internal and external validity of the experiments (Dieste *et al*., 2011). Such a suggestion was investigated by the Dieste *et al*. (2011) approach and the preliminary results affirmed the relationship between bias and internal validity. Some approaches have been applied in experiments in the SE domain, software inspection, pair programming and human-centric, selected throughout systematic literature reviews.

Existing literature provides four approaches to evaluate the quality of SE experiments:

- A1 - the Kitchenham and Charters (2007) checklist;
- A2 - the Kampenes (2007) checklist;
- A3 - the Kitchenham *et al*. (2010) checklist;
- A4 - the Dieste *et al*. (2011) quality scale.

Thus, it is necessary to know each one of them, as well as they are structured to evaluate the quality of the SE experiments. An emerging growth area is Software Product Line (SPL), which aims to generate specific products based on the reuse of a central infrastructure (Linden *et al*., 2007). A large number of experiments have been conducted on a wide range of subjects related to SPL. Such experiments are important in providing evidence to the industry and to the academia, seeking the transfer of technology through a reliable and auditable body of knowledge. In this way, being able to evaluate the quality of an experiment becomes essential to achieve such objectives with the researches.

Therefore, this paper describes an empirical study performed to comparing the mentioned approaches in terms of: (i) Quality of experiments, (ii) usage of a template for experimental reporting and (iii) granularity the approach questions.

Results obtained with this empirical study indicate a significant correlation between pairs of quality evaluation approaches of SE experiments applied in the SPL context. Pearson's correlation was applied and was strong positive for all pairs of evaluated approaches. The A2 (Kampenes, 2007) and A3 (Kitchenham *et al*., 2010) approaches pair presented the highest correlation with 0.883, which were the ones that achieved the best results in the variables "Usage of templates for experimental reporting" and "Granularity of questions from the approaches". Thus, A2 and A3 approaches are the best for reporting SPL experiments.

This paper is structured as follows: Section 2 presents the essential concepts with regard to the quality evaluation of experiments in SE, SPL, SPL experimentation and related work; section 3 reports the empirical study conducted; and section 4 presents conclusions and directions for future work.

Background and Related Work

This section presents initial concepts for this work.

Quality Evaluation of Experiments in Software Engineering

The quality of SE experiments can be observed in regarding to the amount of bias in the experimental results (Dieste and Juristo, 2013). As bias cannot be measured, there are approaches to evaluate it (Dieste *et al*., 2011; Dieste and Juristo, 2013), such as:

- **Simple approaches:** A set of validity criteria, usually answered in a qualitative way, applying a classification scale
- **Checklists:** Based on quality items, in which they are not punctuated numerically, such as a considerable number of quality related questions answered with "Yes/No"
- **Quality Scales:** Based on a series of quality items, numerically punctuated to provide a quantitative assessment of the overall quality of the study. Punctuation tends to be subjective, as it can be generated by weighing all items in the same way or assigning them different weights in relation to the importance of the evaluated criteria
- **Expert opinion:** One or several experts provide an evaluation of the quality of an experiment based on its nominal value, i.e., a subjective evaluation of the overall quality of the paper based on an ordinal scale of 5 points (excellent (5), very good (4), acceptable (3), poor (2) and unacceptable (1)), being able to distinguish experiments with high and low quality

The experimental quality in SE can also be evaluated considering the design and analysis of the experiments, in terms of statistical power, effect size analysis, quasi-experimentation and experiment report (Kampenes, 2007).

The following are approaches to quality evaluation of SE experiments found in the literature.

The Kitchenham and Charters Approach

The quality evaluation checklist for SE experiments proposed by Kitchenham and Charters (2007) contains 52 questions divided into design, conduct, analysis and conclusions, suggesting researchers to select only the most appropriate checklist questions to the context of their own research questions (see Table 6, Table 7 and Table 8 in Appendix A). In addition, such questions should be assigned to a measurement scale, using as a quality instrument a checklist, a quality scale, or both, when there is no simple answer (Yes/No). In this work, all 52 questions answered with "Yes/No" were used.

The Kampenes Approach

The quality evaluation checklist for SE experiments proposed by Kampenes (2007) (Appendix B) contains 26 questions of dichotomous responses, divided into subjects, experimental setting, design and analysis and validity/limitations, in which the objective is to help improving the integrity of the experimental reports in SE (see Table 9 in Appendix B).

The Kitchenham et al. Approach

The quality evaluation checklist for SE experiments proposed by Kitchenham *et al.* (2010) (Appendix C) contains 9 questions, where each question has subcategories: (i) "Questions on Aims", (ii) "Questions on Design, Data Collection and Data Analysis" and (iii) "Questions on Study Outcome" totaling 30 sub-questions of dichotomous answers (see Table 10 in Appendix C). The answer to each question is given by a four-point scale: 4 = all questions listed in the "consider" column can be answered with "yes"; 3 = the majority of all (but not all) questions listed in the "consider" column can be answered with "yes"; 2 = some (but the minority) of the questions listed in the "consider" column can be answered with "yes" and 1 = none of the questions listed in the "consider" column can be answered with "yes". At the end, a final score is added to each of the 9 questions.

The Dieste et al. Approach

The quality scale to evaluate for SE experiments proposed by Dieste *et al.* (2011) (Appendix D) contains 10 questions of dichotomous answers based on five dimensions: Experimental context, experimental design, analysis, presentation of results and interpretation of results (see Table 11 in Appendix D). At the end, a global quality score is the percentage of "yes" responses obtained by the experiment in relation to the total of evaluated questions, as presented in the Formula 1. Experiments with a global score close to 1.0 (100%) are high quality experiments. When the score is near 0.0 (0%), the experiments are considered low quality:

$$Quality\,score = \frac{Number\_of\_yeses}{Number\_of\_yeses + Number\_of\_nos} \quad (1)$$

Software Product Lines

An SPL is a set of products that address a particular market segment or a particular mission (Clements and Northrop, 2002). This set of products is also called the product family, in which the members of this family are specific products generated from the reuse of a common infrastructure, named core assets.

The core assets is composed of a set of common characteristics (similarities) and variable characteristics (variabilities) (Linden *et al*., 2007). This core forms the basis of an SPL and includes the Product-Line Architecture (PLA), reusable components, domain models, requirements assets, test plans and feature and variability models.

The PLA is one of the most important SPL artifacts (Linden *et al*., 2007), because it represents the abstraction of all possible product-specific architectures generated from such SPL. Some important PLA requirements are (Medvidovic and Taylor, 2010): to remain stable over the life of SPLs, suffering as few changes as possible; easy integration of new features during the architecture lifecycle; and explicitly represent the variations to provided for reuse.

The feature model contains all the features of an SPL and their interrelationships. According to Apel *et al.* (2013a), a feature is a characteristic or end-user-visible behavior of a software system. A feature may be mandatory, optional or alternative.

The feature model represents the SPL variabilities (Apel *et al*., 2013a). Variabilities are described by: Variation point that allows the resolution of variabilities in SPL generic artifacts; Variant represents the possible elements that can be chosen to solve a variation point; Constraints between variants establish the relationships between one or more variants in order to solve their respective variation points or variability in a given binding time (Halmans and Pohl, 2003; Linden *et al*., 2007; Pohl *et al*., 2005).

Variability is essential for an SPL, as it represents how members of a family of certain products can distinguish themselves from each other (Weiss and Lai, 1999). Thus, a variability can be modeled to allow the development of custom products by configuring and tuning reusable artifacts for a particular context (Pohl *et al*., 2005).

In this context, Pohl *et al*. (2005) developed the SPL engineering framework, which aims to incorporate the core concepts of traditional product-line engineering, providing artifact reuse and mass customization throughout variability. This framework is divided into two processes with their respective subprocesses and artifacts, as shown in Fig. 1 (Pohl *et al*., 2005).

In the Domain Engineering process, the similarities and variabilities of SPLs are identified and realized. This is composed of five main subprocesses: Product Management deals with the SPL scope and its market strategies; *Domain Requirements* Engineering addresses the elicitation and documentation of SPL requirements; Domain Design defines the SPL reference architecture used in the Application Design; *Domain Realization* deals with the design and implementation of common assets; and Domain Testing performs the validation and verification of reusable components.

DOI: 10.3844/jcssp.2019.1396.1429

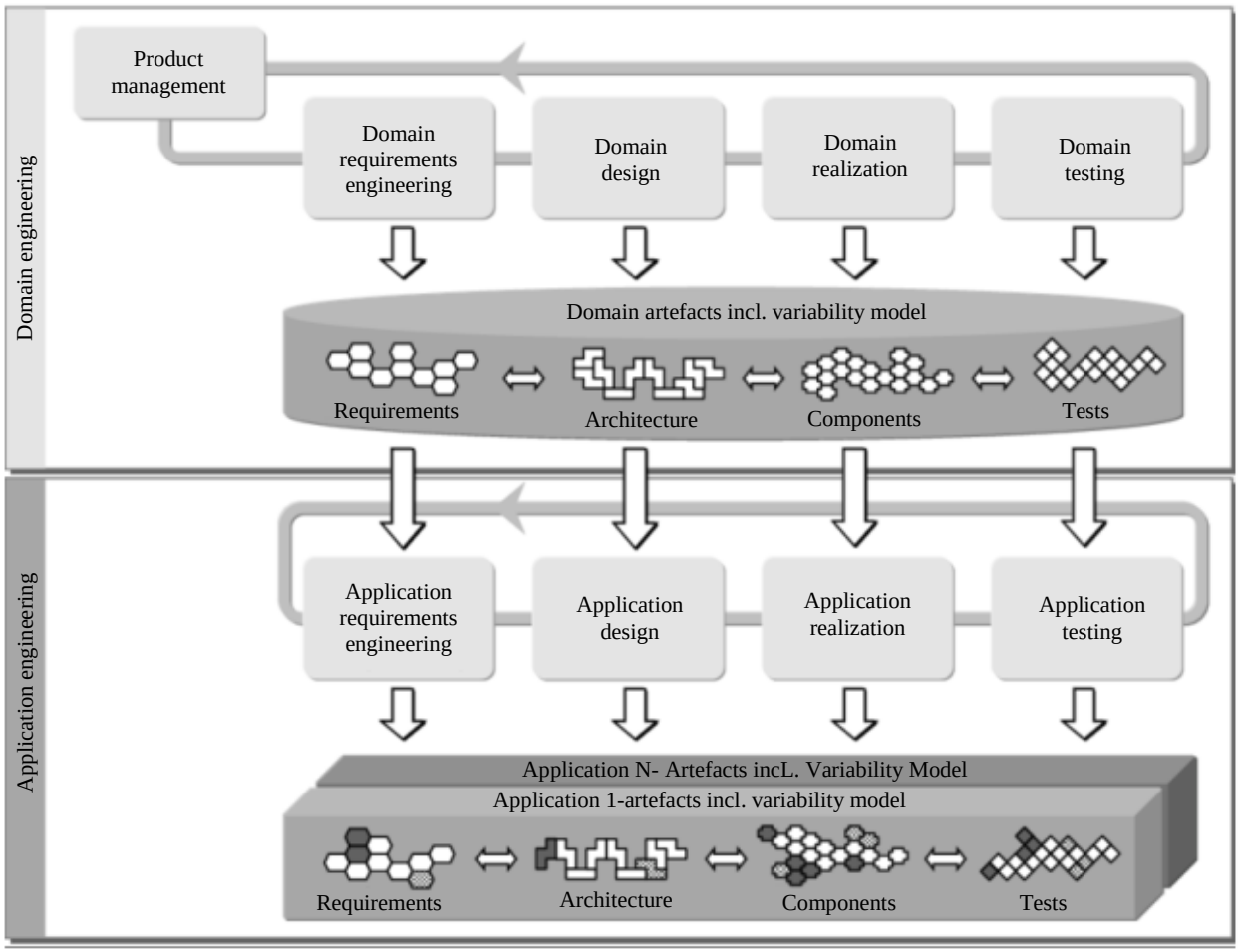

Fig. 1: The SPL engineering framework (Pohl *et al*., 2005)

As for the Domain Engineering artifacts that form the SPL platform and are stored in the same repository, we have: The variability model that presents the variation points and their variants, besides defining the dependencies and constraints; the requirements that are textual or model-based, which they are common to all applications and variables, allowing the derivation of customizable requirements for different applications; the PLA that defines the structure (static and dynamic decomposition that is valid for all SPL products) and the texture (set of common rules for designing and realizing the parts and their combinations to form the applications) of the SPL products; the components that are configurable and perform variability by means of appropriate parameters in its interface; and *tests* that contain the test plan, test cases and test case scenarios to assist in executing them.

In the Application Engineering process, products of an SPL are built by reusing domain artifacts and exploring variabilities. This process is composed of subprocesses: *Application Requirements Engineering* deals with the specification of the application requirements; Application Design derives the product architecture; *Application* Realisation makes the implementation of the products using assets of the subprocess of accomplishment of the domain to reduce the time and effort; and *Application Testing* to validate and verify the derived product.

With relation to the Application Engineering artifacts, they include: The *variability model* that documents the linkage of variability with its justification of selection for a specific application, in addition to being limited by the dependencies and constraints of variability determined in the domain variability model; the *requirements* that present the complete specification of the product, including reusable and application-specific requirements; the *PLA* of the product that is a specific instance of the domain architecture; the components of a specific product configured by parameters; and *tests* that document a given product in a traceable and repeatable way.

Experimentation in SPL

In this section we summarize a Systematic Mapping (SM) study conducted following the guidelines proposed in Kitchenham *et al*. (2015) and Petersen *et al*. (2015) to identify SPL experiments in the literature. A paper on this SM study is currently submitted to a journal.

For this, the search string presented in Table 1 was applied, in which it was adapted according to each data source.

The search for primary studies involved two stages: the first, an automatic search in 5 data sources (IEEE, ACM, ScienceDirect, Scopus and Springer) returned 909 documents; and second, a manual search in 15 Conferences and 11 journals of related areas, such as IET Software, Empirical Software Engineering, IEEE Software, IST, JSS, Empirical Software Engineering and Measurement (ESEM), Brazilian Symposium on Software Engineering (SBES), Evaluation and Assessment in Software Engineering (EASE) and Software Product Line Conference (SPLC) returned 130 documents. We obtained 1.039 primary studies. Filters and results of the process are shown in Fig. 2.

The primary studies selection was initially performed by reading the title, abstract and keywords, in which the inclusion and exclusion criteria were applied. After performing such reading, 219 duplicate studies and 553 studies that did not address SPL experiments were excluded, thus, 267 potential studies were selected.

Then, the selected primary studies were fully read and inclusion and exclusion criteria was applied again. Thus, 94 studies were excluded and a study that addressed a replicated experiment and quoted the original experiment, but the latter was not included in the initial set of studies, thus this study was included later. Therefore, 174 studies were selected for the application of quality assessment criteria (see Table 12 in Appendix E).

During quality assessment, we observed 89% (155) obtained high quality, 11% (19) had medium quality and none (0%) low quality, confirming the relevance and credibility of the selected studies. Thus, a final set of 174 primary studies were selected, presented in Appendix A.

As for the analysis carried out in these studies, we observed SPL experiments are on testing, architecture design optimization and feature model configuration domain, for example. With regard to the main artifacts we obtained feature model, SPL documentation, test cases, class diagrams and use cases, source code, among others. Choice and size of SPL, participants experience and number of SPL architecture were the most discussed threats to validity.

Related Work

To the best of our knowledge, no studies were found related to this empirical study, neither for comparing the SE experiment quality evaluation approaches nor for the SPL context.

Table 1: General search string

"software"
AND
("product line" **OR** "product lines" **OR** "product-line" **OR** "productlines" **OR** "product line engineering" **OR** "product-family" **OR** "product-families" **OR** "product family" **OR** "product families" **OR** "family of products")
AND
("experiment" **OR** "experiments" **OR** "experimental" **OR** "experimentation" **OR** "controlled experiment" **OR** "controlled experiments" **OR** "quasi-experiment" **OR** "quasi-experiments" **OR** "quasi-experimental" **OR** "quasi-experimentation")

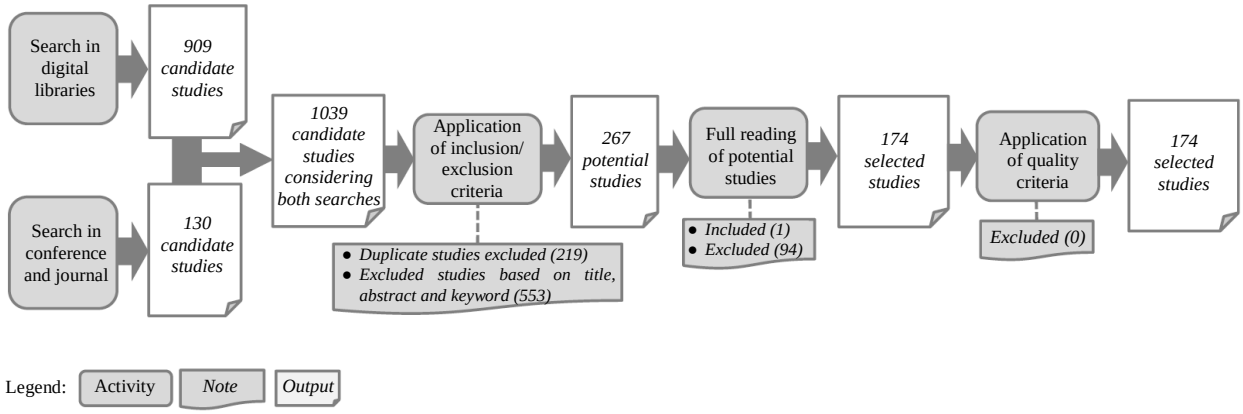

Fig. 2: Filters and results of the SM primary study selection process

Empirical Study

This section reports the empirical study carried out.

Study Planning

Goals

Based on the Goal-Question-Metric (GQM) model (Basili and Rombach, 1988), the empirical study objective was: **Compare** approaches of quality evaluation of experiments in SE, **with the purpose of** evidencing, **with respect to** the capability to report SPL experiments, **from the point of view of** SPL researchers, **in the context of** graduate students of the SE area from the State University of Maringá (UEM).

The main goal of our empirical study can be established in the following Research Question (**R.Q.**): "*What is the best approach for reporting SPL experiments based on approaches for quality evaluation of experiments in SE?*"

Participants

The participants in this study were the first three co-authors of this paper. Thus, a random selection was not performed. All the people involved in this project are master's students in the Graduate Program in Computer Science of the State University of Maringá (UEM). The first coauthor has experience with SE experiments for about two and a half years, while the second and third co-authors have been around for 1 year. Although the participants had low experience in SE experimentation, they were considered eligible to act as quality participants of the papers assessed in this study, due to the fact that they were under the supervision of Prof. Dr. Edson OliveiraJr and they attended a 1-semester Experimental SE graduate course.

Experimental Units

There is one main experimental unit involved in this empirical study: A set of 30 papers that discuss SPL experiments. This number of papers was chosen because it is considered the minimum adequate value for statistical analysis and because the evaluation approaches have so many questions. The selected papers are a subset of SM primary studies (Section 2.3). From the 174 SM papers, 30 papers were selected from a random function provided by Microsoft Excel®, which was sufficient to ensure that the selection of the papers was not intentional.

Material

The material used in this study were an Excel spreadsheet with the four approaches to quality evaluation of SE experiments.

Tasks

In this study each participant acted as a evaluator and evaluated each of the 30 papers using the following approaches to determine quality: A1 (Kitchenham and Charters, 2007) checklist, A2 (Kampenes, 2007) checklist, A3 (Kitchenham *et al*., 2010) checklist and A4 Dieste *et al*. (2011) quality scale.

Variables

Table 2 describes the dependent and independent variables of our study. Its has the abbreviation "N.A." (Not Available) used when there is no such information.

The independent variables were the approaches for quality evaluation of experiments in SE: A1, A2, A3 and A4.

The dependent variables were:

- **Quality of experiments:** Represents the quality of reporting of each SPL experiment for each approach
- **Usage of templates for experimental reporting:** Represents the quality of each SPL experiments for each approach when using or not using an experimental template for reporting
- **Granularity of questions from the approaches:** Represents the quality of each SPL experiment for each approach divided into experimental phases proposed by Wohlin *et al*. (2012)

This variables contribute to identify the best quality evaluation approach for reporting SPL experiments because three dependent variables deal with reporting quality aspects of the experiments, in which the first one is more generally, the second relates to the use of experimental template and the last one in a more specific way by experimental phase, since each approach categorized their questions with names similar to experimental phases.

Table 2: Dependent and independent variables description

Name of variable	Type of the variable (independent, dependent, moderating)	Abbreviation	Class (product, process, resource, method)	Entity (instance of the class)	Type of Attribute (internal, external,...)	Scale type (nominal, ordinal,...)	Unit	Range	Counting rule
Approaches for quality Evaluation of Experiments in SE	Independent	N.A.	Method	A1, A2, A3 and A4	N.A.	Nominal	N.A.	A1, A2, A3 and A4	N.A.
Quality of Experiments	Dependent	N.A.	Process	Weak, Moderate, Strong, Positive, Negative	External	Nominal, ordinal	Correlation scale	[-1.0,+1.0]	Pearson (p) formula
Usage of Templates for Experimental Reporting	Dependent	N.A.	Process		External				T-Test and Mann-Whitney-Wilcoxon U
Granularity of Questions from the Approaches	Dependent	N.A.	Process		External			Between 0 and 1	Dieste *et al.* formula

Design

We used one factor with more than two treatments design, in which the treatments are compared to each other (Wohlin *et al*., 2012). The factor in this study is the approaches for quality evaluation of experiments in SE and the treatments are: A1, A2, A3 and A4.

Procedure

For each participant the same 30 selected papers were assigned not randomly. For each paper, the participant answered questions in the A1, A2, A3 and A4 approaches. Each question or sub-question was answered with "yes" or "no". For each "yes" answer, the participants had the task of writing an observation that contained the page, section and/or paragraph in which the information was contained, in addition to noting the response time for each approach.

Analysis Procedure

After completing the answers in the spreadsheet of papers versus approaches, Kappa (Fleiss *et al*., 2003) agreement analysis was performed with the support of a tool (2Available at http://www.lee.dante.br/pesquisa/kappa/index.html. Accessed on 10/09/2018) developed by the Laboratory of Epidemiology and Statistics (Lee) of the Dante Pazzanese Institute of Cardiology to verify the agreement between the 3 participants.

Finally, a consensus was made between the participant's assessments of the responses of the approaches in each paper to subsequently obtain the quality of experiments, the usage of templates for experimental reporting and the granularity of questions from the approaches.

Execution

Preparation

Before starting the our empirical study, a pilot project was performed, in which the participants selected 2 random papers: Michalik *et al*. (2011a) and Murashkin *et al*. (2013a), which are not included in the 30 papers of the study, to calibrate and equalize the evaluation form of each participant.

For this preparation, the participants carried out the evaluation of each of the 2 papers together for the 4 proposed approaches in order to solve possible doubts with regard to the interpretation of questions.

Deviations

During the execution of our study, there were no deviations with regard to the planned instrumentation and the collection process. In addition, none of the participants dropped out of the study.

Analysis

We present the analysis of our empirical study in this section, initially with the Kappa agreement analysis, the quality of experiments, the usage of templates for experimental reporting and the granularity of questions from the approaches.

Kappa Agreement Analysis

To evaluate the level of agreement between the 3 participants, in relation to the answers obtained from the four approaches applied in the 30 papers, we performed the Kappa statistical test (Fleiss *et al*., 2003) and to get to know whether the results would be satisfactory or not we based on the interpretation proposed by Fleiss *et al*. (2003), as shown in the Table 3.

Table 4 presents the results obtained with the application of the coefficient Kappa for each approach in relation to the 30 papers.

Table 3: Interpretation of Kappa values proposed by (Fleiss *et al*., 2003)

Kappa values	Strength of Agreement
<0.40	Poor agreement
0.40-0.75	Fair to good agreement
>0.75	Excellent agreement

Table 4: Results of the Kappa Coefficient for each Paper (P) and Approach (A1 through A4)

Paper ID	A1	A2	A3	A4
P1	0.490	0.479	0.509	0.144
P2	0.583	0.422	0.422	0.607
P3	0.466	0.490	0.466	0.365
P4	0.433	0.328	0.186	0.345
P5	0.350	0.061	0.244	0.029
P6	0.308	0.458	0.507	0.492
P7	0.461	0.443	0.598	0.151
P8	0.436	0.597	0.432	0.542
P9	0.280	0.554	0.220	0.112
P10	0.477	0.498	0.293	-0.179
P11	0.438	0.527	0.630	0.139
P12	0.193	0.490	0.278	0.340
P13	0.435	0.810	0.626	0.340
P14	0.374	0.409	0.482	-0.138
P15	0.562	0.434	0.499	0.083
P16	0.562	0.434	0.499	0.083
P17	0.562	0.434	0.499	0.083
P18	0.661	0.683	0.460	1.000
P19	0.793	0.742	0.821	0.345
P20	0.654	0.519	0.656	0.505
P21	0.405	0.639	0.644	0.591
P22	0.422	0.494	0.814	0.340
P23	0.610	0.685	0.686	0.591
P24	0.462	0.647	0.386	0.365
P25	0.472	0.473	0.400	0.637
P26	0.584	0.507	0.545	0.628
P27	0.381	0.771	0.333	-0.031
P28	0.579	0.682	0.408	0.393
P29	0.508	0.647	0.666	0.293
P30	0.340	0.494	0.358	0.158

DOI: 10.3844/jcssp.2019.1396.1429

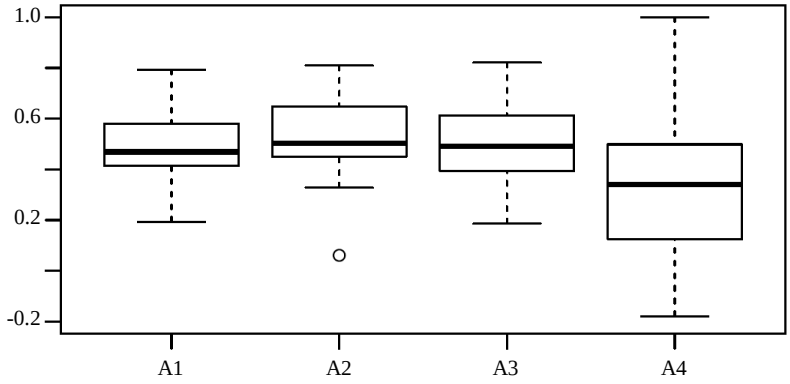

Fig. 3: Box plot for the Kappa values of each approach

As it can be seen in Fig. 3, the median shows a very close agreement between A1, A2 and A3 approaches, which are close to 0.4 and 0.5 indicating a fair to good agreement between the answers of the 3 participants for these approaches. Except for a disparity in the A2 approach (P5, 0.061) considered an outlier that was maintained in the analysis. For the A4 approach, the median was below 0.4, indicating a poor agreement among the participants.

Quality of Experiments

After analyzing the degree of agreement between the answers, the 3 participants reached a consensus for a single answer on each question of each of the four approaches in the 30 papers under study. To do so, when the response of two participants was: "yes", the final answer was "yes"; and "no", the final answer was "no". In the case where only one of the participants disagreed in the response, a discussion was held among the participants to reach a final answer.

Shortly after obtaining a single answer for each question in the 30 papers in each approach, it was possible to obtain the final quality score of the four approaches, as presented in Table 5. For this, in the approaches A1, A2 and A4 we used the Formula 1 (Section 2.1.4) proposed by Dieste *et al.* (2011), where it is composed of the number of "Yes" responses divided by the number of "Yes" responses added to the number of "No" responses. Whereas for the A3 approach, because the answers to the questions are given by a four-point scale, we applied the Formula 2, in which the sum of the 9 questions divided by 36 is made (which is the maximum value of the summation, that is, 9 (questions) * 4 (when all answers of the subquestions are "yes")), getting a value between 0 and 1, just as the result of the Formula 1:

$$Quality\ score = \frac{\Sigma Questions}{36} \quad (2)$$

Thus, the final quality score obtained in each approach (Table 5) was used to calculate the correlation between pairs of approaches. Based on this, the following hypotheses were stated:

- **Null Hypothesis (H_0):** There is no significant correlation between the X and Y approaches for evaluating the quality of SPL experiments
- **Alternative Hypothesis (H_1):** There is significant correlation between the X and Y approaches for evaluating the quality of SPL experiments

X and Y are replaced with each pair of approaches in this study.

As this variable is continuous, we applied the parametric correlation method of Pearson using the statistical tool R (R Core Team, 2014). This method allows to establish whether there is a correlation between two sets of data, represented by ρ, which assumes value between -1 and 1, as shown in Fig. 4.

Thus, the following values for ρ were obtained for the correlations:

- *Corr*(*A1 and A2*): ρ = **0.780** - Strong positive correlation
- *Corr*(*A1 and A3*): ρ = **0.830** - Strong positive correlation
- *Corr*(*A1 and A4*): ρ = **0.783** - Strong positive correlation
- *Corr*(*A2 and A3*): ρ = **0.883** - Strong positive correlation
- *Corr*(*A2 and A4*): ρ = **0.842** - Strong positive correlation
- *Corr*(*A3 and A4*): ρ = **0.800** - Strong positive correlation

Based on the correlations, there is initial evidence to reject the null hypothesis (H_0) of our study and accept the alternative hypothesis (H_1) (Section 3.3.2), which affirms there is a significant correlation between the approaches for quality evaluation of SPL experiments.

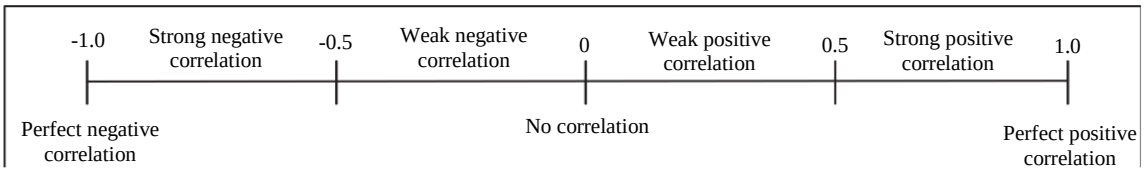

Fig. 4: Pearson correlation scale

Table 5: Final quality score of the papers (P) by approach (A1 through A4) after Kappa analysis

Paper ID	A1	A2	A3	A4
P1	0.385	0.692	0.778	0.455
P2	0.462	0.654	0.750	0.364
P3	0.519	0.731	0.722	0.636
P4	0.327	0.308	0.472	0.364
P5	0.558	0.731	0.778	0.364
P6	0.558	0.808	0.778	0.455
P7	0.596	0.731	0.778	0.545
P8	0.269	0.231	0.583	0.182
P9	0.635	0.731	0.750	0.636
P10	0.250	0.115	0.528	0.091
P11	0.500	0.577	0.667	0.364
P12	0.269	0.385	0.556	0.182
P13	0.500	0.769	0.778	0.545
P14	0.288	0.192	0.500	0.091
P15	0.500	0.615	0.778	0.545
P16	0.212	0.346	0.528	0.091
P17	0.442	0.500	0.694	0.455
P18	0.231	0.423	0.611	0.273
P19	0.365	0.577	0.667	0.364
P20	0.154	0.269	0.444	0.182
P21	0.231	0.615	0.639	0.364
P22	0.346	0.346	0.694	0.455
P23	0.154	0.308	0.500	0.273
P24	0.519	0.769	0.722	0.545
P25	0.250	0.154	0.528	0.000
P26	0.346	0.654	0.694	0.545
P27	0.154	0.385	0.500	0.182
P28	0.135	0.308	0.500	0.182
P29	0.231	0.462	0.611	0.182
P30	0.288	0.615	0.750	0.364

Figure 5 presents the scatter plots of the pairs of approaches, emphasizing that all have a positive trend line, confirming the rejection of the null hypothesis (H_0). Thus, the obtained values of the correlations show the four approaches produce similar results.

Figure 6 presents the quality of the SPL experiments based on the final score of the papers after evaluation in each of the selected approaches. It is possible to observe that the A3 approach obtained the quality of the experiments with a final score of close to 0.7. We understand A3 has more objective questions and the information regarding its questions is more present in the papers analyzed.

Usage of Templates for Experimental Reporting

In the 30 experiments selected, we checked whether they adopted any experimental template by referencing the paper that followed the guidelines or the structure of the experiment when they did not explicitly state. Of the 30 papers, only 13 papers used experimental template: 11 papers used by Wohlin *et al*. (2012), one paper by Kitchenham *et al*. (2002) and one paper by Jedlitschka *et al*. (2008). Based on this, the following hypotheses were stated:

- **Null Hypothesis (H_0):** There is no significant difference in the quality of SPL experiments when using an experimental template
- **Alternative Hypothesis (H_1):** There is significant difference in the quality of SPL experiments when using an experimental template

We applied the Kolmogorov-Smirnov normality test for both samples in each of the approaches: using and not using experimental template. We obtained the following results:

- For A1: (i) using template, sample size (**N**) 13, mean (μ) 0.4053, standard deviation (σ) 0.1264, we obtained p = 0.9027, i.e., with α = 0.05 (0.9027 > 0.361), the sample is normal; and (ii) not using template, sample size (**N**) 17, mean (μ) 0.3179, standard deviation (σ) 0.1563, we obtained p = 0.5479, i.e., with α = 0.05 (0.5479 > 0.318), the sample is normal
- For A2: (i) using template, sample size (**N**) 13, mean (μ) 0.6242, standard deviation (σ) 0.1144, we obtained p = 0.9767, i.e., with α = 0.05 (0.9767 > 0.361), the sample is normal; and (ii) not using template, sample size (**N**) 17, mean (μ) 0.4051, standard deviation (σ) 0.2177, we obtained p = 0.2698, i.e., with α = 0.05 (0.2698 < 0.318), the sample is not normal
- For A3: (i) using template, sample size (**N**) 13, mean (μ) 0.7052, standard deviation (σ) 0.0617, we obtained p = 0.9282, i.e., with α = 0.05 (0.9282 > 0.361), the sample is normal; and (ii) not using template, sample size (**N**) 17, mean (μ) 0.5948, standard deviation (σ) 0.1207, we obtained p = 0.2848, i.e., with α = 0.05 (0.2848 < 0.318), the sample is not normal
- For A4: (i) using template, sample size (**N**) 13, mean (μ) 0.4197, standard deviation (σ) 0.1258, we

obtained p = 0.6183, i.e., with α = 0.05 (0.6183 > 0.361), the sample is normal; and (ii) not using template, sample size (**N**) 17, mean (μ) 0.2835, standard deviation (σ) 0.1897, we obtained p = 0.3139, i.e., with α = 0.05 (0.3139 < 0.318), the sample is not normal

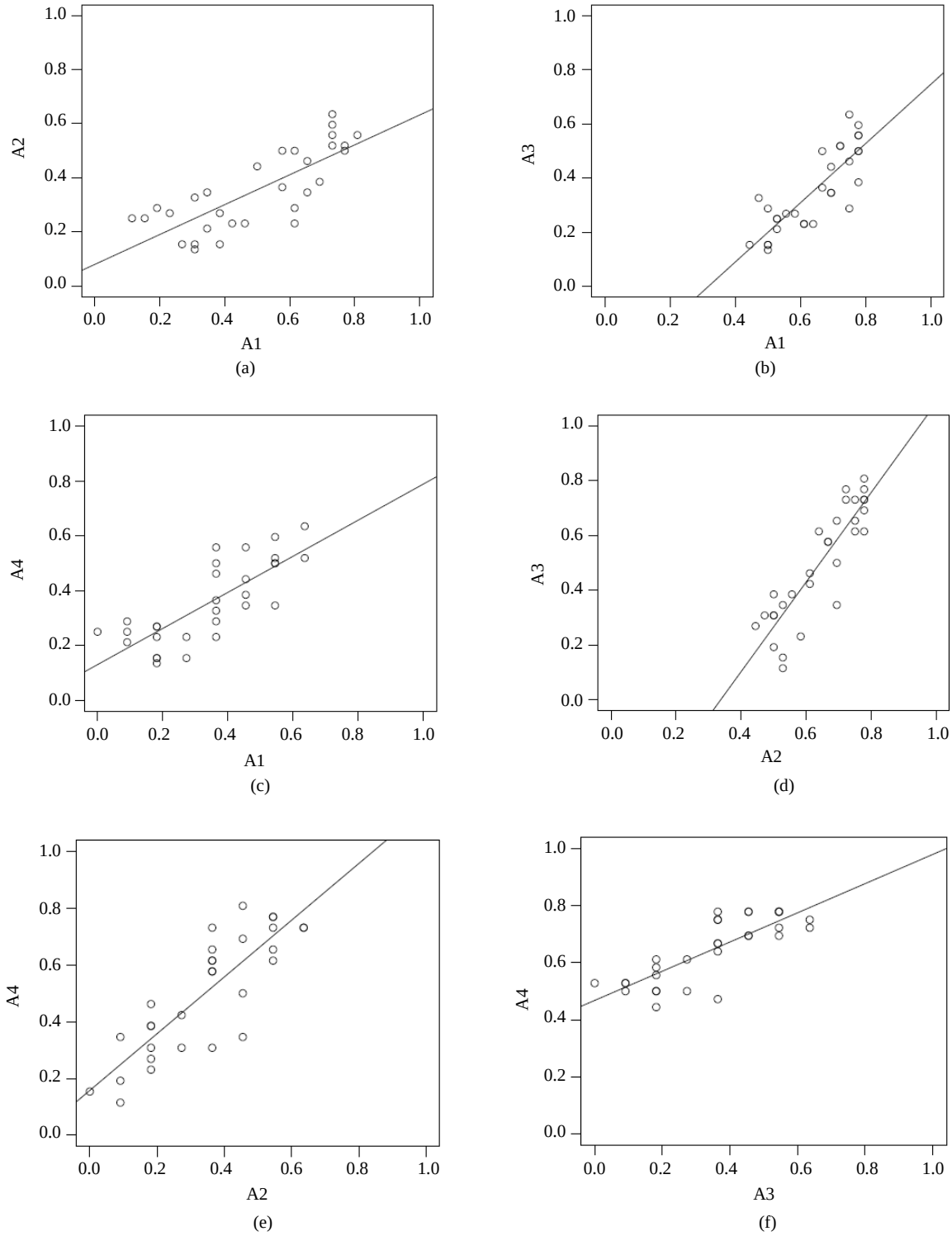

Fig. 5: Scatter plots of the pair of approaches to the final quality scores; (a) A1 X A2; (b) A1 X A3; (c) A1 X A4; (d) A2 X A3; (e) A2 X A4; (f) A3 X A4

DOI: 10.3844/jcssp.2019.1396.1429

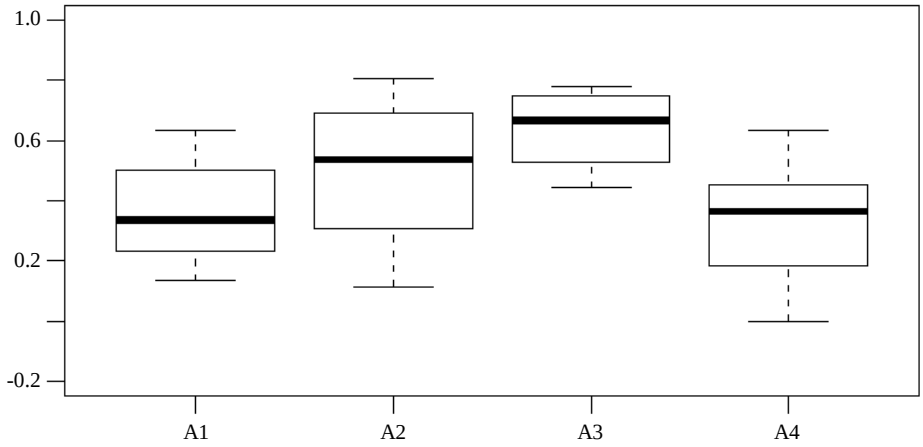

Fig. 6: Box plot for the quality of SPL experiments

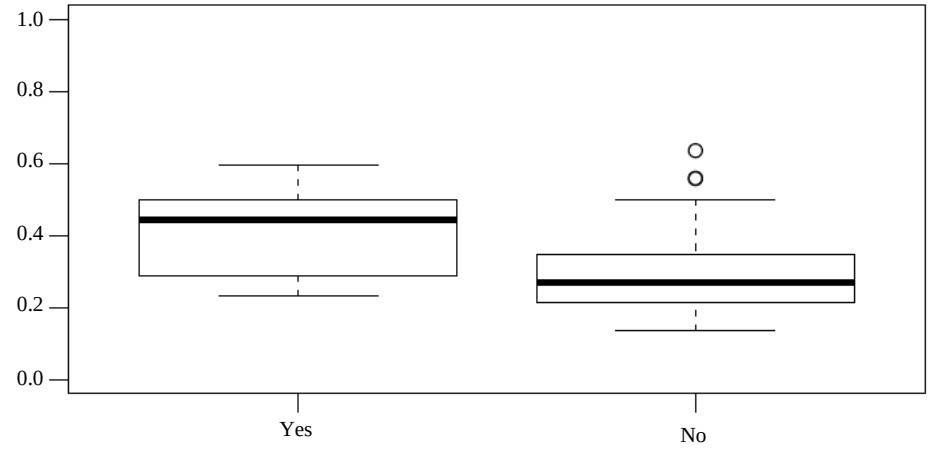

Fig. 7: Box plot for the usage template in A1 approach

In the A1 approach, both samples (using and not using experimental template) are normal, thus we applied the T-Test, obtaining a value for $t_{calculated}$ = 1.6922 and degree of freedom (gl) = 28. When searching the index (gl) in the table of critical values of the T-Test, a value for $t_{critical}$ = 2.05 was found, with a (α) significance level of 0.05. Thus, comparing the $t_{critical}$ with the $t_{calculated}$, the null hypothesis could not be rejected ($t_{calculated}$(1.6922) < $t_{critical}$(2.05)).

Figure 7 presents the "Yes" and "No" box plots for the usage of experimental template for A1. In the "No" box plot, we see two outliers that were above the median of experimental template, which is a possible evidence of not having rejected the null hypothesis (H_0). Thus, we remove these two outliers that correspond to three data point from this sample, obtaining a size sample 14. Then we re-applied the T-Test presenting the result for $t_{calculated}$ = 3.2756 and degree of freedom (gl) = 23 ($t_{critical}$ = 2.07), thus the null hypothesis (H_0) could be rejected ($t_{calculated}$(3.2756) > $t_{critical}$(2.07)). Thus, the result of T-Test was considered, after removal of the outliers in the sample.

A2, A3 and A4 approaches, samples using experimental template ($\boldsymbol{N_A}$) are normal, but those that do not use experimental template ($\boldsymbol{N_B}$) are non-normal, thus we applied the Mann-Whitney-Wilcoxon U. In all approaches, the ($\boldsymbol{N_A}$) sample has 13 data and the ($\boldsymbol{N_B}$) 17 data, so in the table of critical values of that test the value for $U_{critical}$ = 63 was found, with a (α) significance level of 0.05. For the $U_{calculated}$, the values obtained for the A2, A3 and A4 approaches are shown below:

- For the A2 approach, $U_{calculated}$ = 46.5, comparing with the $U_{critical}$ the null hypothesis (H_0) could be rejected ($U_{calculated}$(46.5) <= $U_{critical}$(63))
- For the A3 approach, $U_{calculated}$ = 54.5, comparing with the $U_{critical}$ the null hypothesis (H_0) could be rejected ($U_{calculated}$ (54.5) <= $U_{critical}$(63))

- For the A4 approach, $U_{calculated}$ = 62.5, comparing with the $U_{critical}$ the null hypothesis (H_0) could be rejected ($U_{calculated}$ (62.5) <= $U_{critical}$ (63))

Figure 8 presents the box plots for the papers quality regarding the usage of experimental template in the reporting of the SPL experiments for the four approaches. We can note is the median quality for experiments using experimental template is higher than those that did not use it.

Among the approaches, A2 and A3 were the ones obtained the highest quality using experimental template, in which the final quality score is between 0.6 and 0.7. Thus, we understand the experimental template influences the quality of the experiments.

Therefore, the T-Test and the Mann-Whitney-Wilcoxon U test confirmed that there is a significant difference in

the quality of SPL experiments when using experimental template in the A1, A2, A3 and A4 approaches, which was already expected.

Granularity of Questions from the Approaches

We divided the questions from the four approaches for quality evaluation of experiments into four phases, according to Wohlin *et al*. (2012): Scope, Planning, Execution and Analysis and Interpretation, in order to evaluate their granularity. For each phase, we calculated the quality score in each of the approaches using Formula 1 (Section 2.1.4). For the A3 approach, the questions from 1 to 9 were disregarded because they used a four-point scale and because they did not have the same weights, thus with their subquestions it was possible to apply Formula 1.

Figure 9 presents the granularity of the questions in the planning phase. It is observed that the A2 approach was the one that obtained a score of higher quality, close to 0.6, whereas the others were close to 0.5.

Figure 10 presents the granularity of the questions in the analysis and interpretation phase. It is observed that the A3 approach was the one that obtained a quality score of approximately 0.6. Approaches A1 and A4 had a score of quality close to 0.3 and A2 close to 0.2.

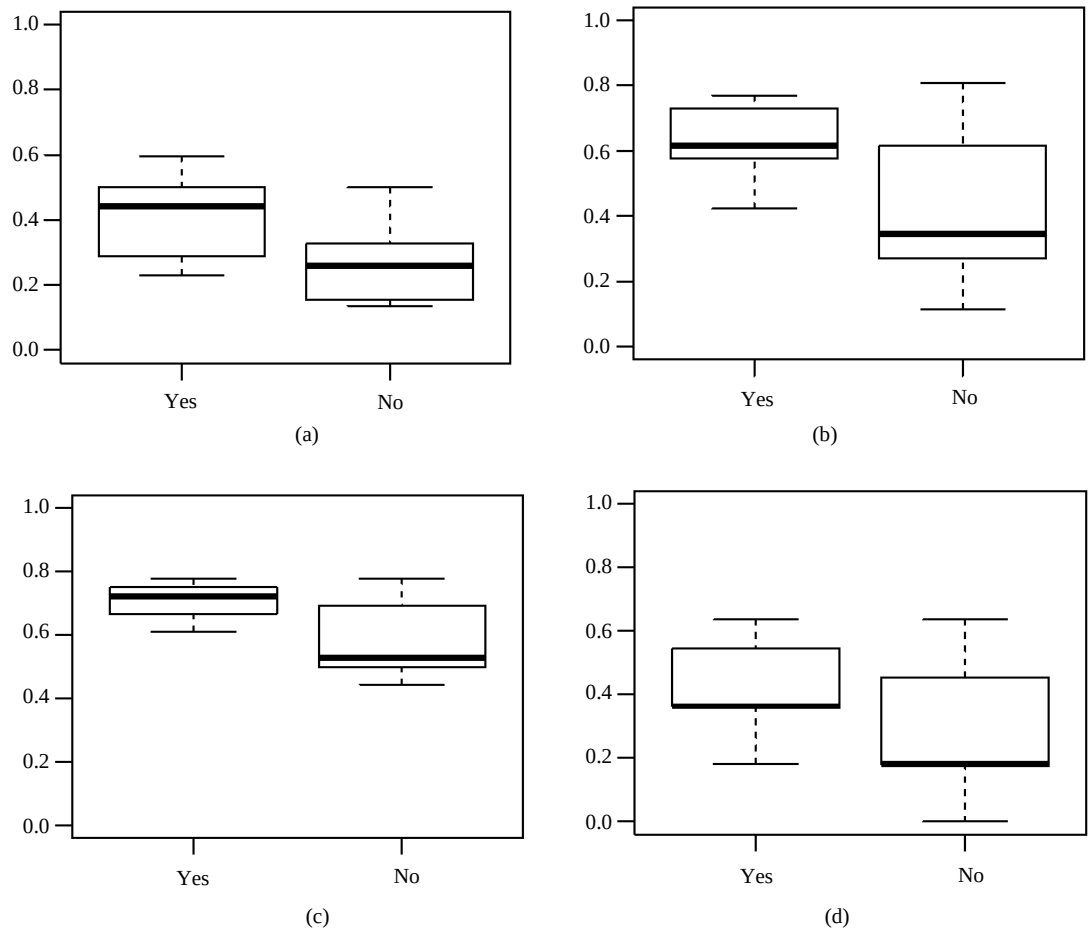

Fig. 8: Box plot for the usage and non-usage of template in relation to the four approaches; (a) A1 (b) A2 (c) A3 (d) A4

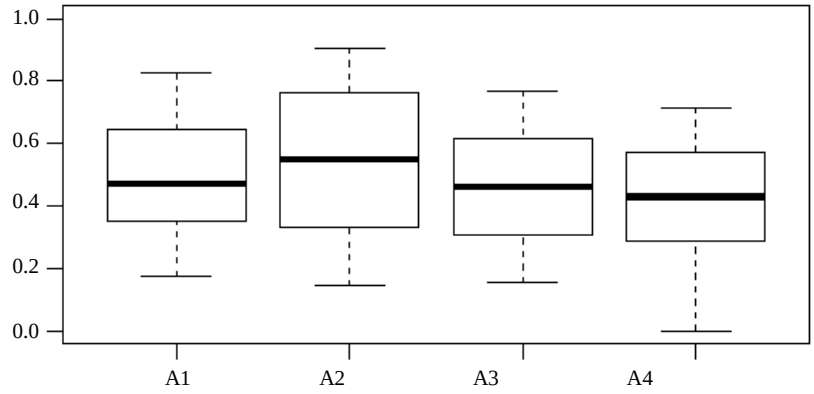

Fig. 9: Box plot for the granularity of questions in the planning phase

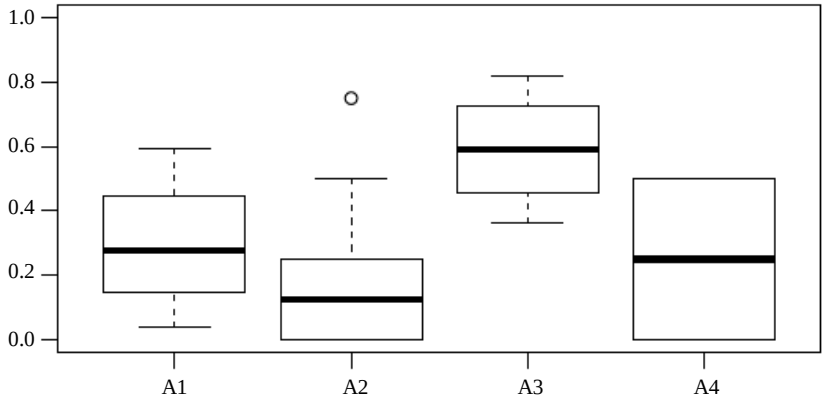

Fig. 10: Box plot for the granularity of questions in the analysis and interpretation phase

The scope and execution phases were those that presented few questions, obtaining a very small sample, therefore, we did not analyze such phases.

When comparing Fig. 9 and 10 with Fig. 6 showing the quality of the SPL experiments in each approach, it is observed that the A1, A2 and A4 approaches obtained a score of higher quality in the planning phase (Fig. 9) and only the A2 approach obtained a score of lower quality in the analysis and interpretation phase (Fig. 10) as presented in Fig. 6. Thus, we understand the SPL experiments report more the planning phase than the analysis and interpretation phase, which also is a phase of paramount importance.

Discussion

This section provides an overall discussion of the obtained results and implications, as well as threats to validity of the empirical study carried out.

Evaluation of Results and Implications

With the results obtained in our empirical study and the performed analysis, there is a significant correlation between pairs of approaches for quality evaluation of SPL experiments. This correlation was strongly positive for all pairs of approaches evaluated and was based on three criteria: Quality of experiments, usage of templates for experimental reporting and granularity of questions from the approaches.

The pair of approaches A2 and A3, which obtained the highest correlation with 0.883 compared to the other pairs, was the one that presented the best result in the evaluated criteria. It is possible to observe that the A3 approach was the one that presented the highest score in the “Quality of the experiments”, “Granularity of the questions” and the “Usage of templates for experimental reporting”.

With regard to the pair of approaches A2 and A4, with correlation 0.842, such approaches are those that contain fewer questions compared to others, an indication that the results were found in the criterion “Granularity of the questions”. In the other two criteria, the A4 approach presented more difference because of the complexity of its questions, with few papers documenting the information needed.

For the pair of approaches A1 and A3, with a correlation of 0.830. both approaches were developed by Kitchenham with the support of other authors, which

makes it possible to infer that the results are close to the criteria of "Granularity of the questions" and "Usage of templates for experimental reporting". They differ in the criteria of "Quality of the experiments", because A1 contains 52 questions, which has more questions of all approaches and many of them were not present in the papers being studied, for example, "*Did untoward events occur during the study?*" and "*Is the sample representative of the population to which the results will generalise?*".

Considering the pair of approaches A3 and A4, with a correlation of 0.800, the approaches approximate the criteria of "Granularity of the questions", by having a division of the questions into very similar dimensions. However, they disagree on the other two criteria, because A4 has questions that analyze information not very reported in the papers, for example, "*Was randomization used for selecting the population and applying the treatment?*" and "*Are the statistical significances mentioned with the results?*".

With relation to the pair of approaches A1 and A4, with a correlation of 0.783, these approaches resemble the dimensions that have been divided into their questions, providing evidence for being in the criterion of "Granularity of the questions". Furthermore, they obtained a near-quality score as presented in the "Quality of the experiments" criterion. As for the criterion of "Usage of templates for experimental reporting", A1 presented a score of better quality compared to A4 because of the level of complexity of the A4 approach questions, for example, "*Is an appropriate blinding procedure used (e.g., blind allocation of materials, blind marking)?*".

In the pair of approaches A1 and A2, which obtained the lowest correlation (0.780), it is observed that in the criterion "Granularity of the questions", both approaches had very close results. With relation to the other criteria, which are based on the quality score obtained from the papers, A2 stands out because it contains more objective questions, for example, "*Sample size*", "*The use of tools*" and "*Assignment procedure (randomized or quasi)*".

Threats to Validity

The threats to validity identified during this empirical study were categorized into internal, external, construct and conclusion, according to Wohlin *et al*. (2012).

Internal Validity

One of the threats was the difference in the knowledge level of the participants, especially in relation to the selected approaches. To minimize this threat, we conducted a bibliographic study of each approach, followed by a pilot project to unify the understanding of each question in each evaluation approach, providing the balance of knowledge necessary to carry out our empirical study.

Another threat was related to the influence of the participants during the study. To minimize such a threat, the spreadsheets that the participants completed were stored locally by each one and were only shared between them after all had been filled out. In addition, each participant worked in their own environment at a time most suitable for him/her.

External Validity

One of the threats was in relation to the selected participants, since they are master students in the SE area. However, more studies that include researchers with experience in SE Experimental and SPL should be conducted, thus the results can be generalized.

Another threat was the instrumentation used in relation to the heterogeneity of the sample, in which the selected papers obtained a moderate quality, as presented in Fig. 6. Thus, new studies must be conducted, stratifying the experiments with high, moderate and low quality.

Construct Validity

The dependent variable *correlation between the pairs of approaches* was calculated according to Pearson. The independent variable *approaches for quality evaluation of experiments in SE* was guaranteed by the pilot project, in which the participants became familiar with such approaches applied in two papers that were not part of the execution of the study, evaluating the feasibility of the applied approach.

Conclusion Validity

A threat considered as a risk to affect statistical validity was the sample size (N = 30), mainly in the analysis of the criterion "Usage of templates for experimental reporting", in which such sample was divided into two samples, one with size 13 and the other with 17. Thus, the size of such sample should be increased during prospective replications of this study.

Study Packing and Sharing

All documents related to the study are available online via the web (https://doi.org/10.5281/zenodo.2575487) in order to promote possible future replications.

Conclusion and Future Work

The quality evaluation of experiments is fundamental to build a body of reference and reliable knowledge. In this paper, we carried out an empirical study, aiming to compare the existing approaches for quality evaluation of SE experiments, in order to indicate which approach was the best to reporting experiments in the SPL domain, based on three criteria presented in Section 3.1.6.

The Pearson's correlation was used between pairs of approaches, which showed strong positive correlations to analyze the "Quality of experiment". The T-Test and

Mann-Whitney-Wilcoxon U test were applied to the samples to analyze the "Usage of templates for experimental reporting" to verify whether there was a difference in the quality of the experiments when using an experimental template, in which in which the result was positive. In "Granularity of questions from the approaches" the SPL experiments report more the planning phase.

Therefore, the results obtained provided initial evidence A2 and A3 approaches are the best for reporting SPL experiments.

Directions for future work include planning the internal and external replication of this study to corroborate with the results obtained. Taking into account the actions discussed in Section 3.4.2 to mitigate threats to validity.

Funding Information

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) - Finance Code 001.

Author's Contributions

Viviane F. Furtado, Henrique Vignando and Victor França: Coordinated the experiment performed at the Informatics Department of the State University of Maringá by planning, executing, analyzing and interpreting it. They also contributed in writing this paper.

Edson OliveiraJr: Supervised Viviane, Henrique and Victor during the experiment, contributed to planning, executing, analyzing and interpreting the experiment results. He also contributed in proofreading and writing this paper.

Ethics

Experiment participants were previously invited, thus they attended such experiments as volunteers. Participants were informed the experiment will contribute to a nonprofit research project coordinated by Dr. Edson OliveiraJr.

References

Accioly, P., P. Borba and R. Bonifacio, 2012. Comparing two black-box testing strategies for software product lines. Proceedings of the 6th Brazilian Symposium on Software Components Architectures and Reuse, Sept. 23-28, IEEE Xplore Press, Natal, Brazil, pp: 1-10. DOI: 10.1109/SBCARS.2012.17

Accioly, P.R.G., P. Borba and R. Bonifacio, 2014. Controlled experiments comparing black-box testing strategies for software product lines. JUCS, 20: 615-639.

Acher, M., A. Cleve, G. Perrouin, P. Heymans and C. Vanbeneden *et al*., 2012. On extracting feature models from product descriptions. Proceedings of the International Workshop on Variability Modeling of Software-Intensive Systems, Jan. 25-27, ACM, Leipzig, Germany, pp: 45-54. DOI: 10.1145/2110147.2110153

Adam, S. and K. Schmid, 2013. Effective requirements elicitation in product line application engineering: An experiment. Proceedings of the International Working Conference on Requirements Engineering: Foundation for Software Quality, Apr. 08-11, Springer, Essen, Germany, pp: 362-378. DOI: 10.1007/978-3-642-37422-7_26

Ahmed, Z., 2007. Measurement analysis and fault proneness indication in Product Line Applications (PLA). Proceedings of the International Conference on New Software Methodologies, Tools and Techniques, (MTT' 07).

Akim, C.H.P., S. Khurshid and D. Batory, 2012. Shared execution for efficiently testing product lines. Proceedings of the IEEE 23rd International Symposium on Software Reliability Engineering, Nov. 27-30, IEEE Xplore Press, Dallas, TX, USA, pp: 221-230. DOI: 10.1109/ISSRE.2012.23

Al-Hajjaji, M., S. Krieter, T. Thum, M. Lochau and G. Saake, 2016. IncLing: Efficient product-line testing using incremental pairwise sampling. Proceedings of the International Conference on Generative Programming: Concepts and Experiences, Oct. 31-Nov. 01, ACM, Amsterdam, Netherlands, pp: 144-155. DOI: 10.1145/2993236.2993253

Al-Hajjaji, M., T. Thum, J. Meinicke, M. Lochau and G. Saake, 2014. Similarity-based prioritization in software product-line testing. Proceedings of the 18th International Software Product Line Conference Sept. 15-19, ACM, Florence, Italy, pp: 197-206. DOI: 10.1145/2648511.2648532

Al-Msie'deen, R., M. Huchard, A.D. Seriai, C. Urtado and S. Vauttier, 2014. Automatic documentation of [mined] feature implementations from source code elements and use-case diagrams with the REVPLINE approach. Int. J. Software Eng. Knowl. Eng., 24: 1413-1438. DOI: 10.1142/S0218194014400142

Almeida, E.S., E.C.R. Santos, A. Alvaro, V.C. Garcia and S.L. Meira *et al*., 2008. Domain implementation in software product lines using OSGi. Proceedings of the 7th International Conference on Composition-Based Software Systems, Feb. 25-29, IEEE Xplore Press, Madrid, Spain, pp: 72-81. DOI: 10.1109/ICCBSS.2008.19

Andersen, N., K. Czarnecki, S. She and A. Wasowski, 2012. Efficient synthesis of feature models. Proceedings of the 16th International Software Product Line Conference, Sept. 02-07, ACM, Salvador, Brazil, pp: 106-115. DOI: 10.1145/2362536.2362553

Apel, S., D. Batory, C. Kästner and G. Saake, 2013a. Feature-Oriented Software Product Lines: Concepts and Implementation. 1st Edn., Springer, ISBN-10: 3662513005, pp: 332.

Apel, S., A.V. Rhein, P. Wendler, A. Groblinger and D. Beyer, 2013b. Strategies for product-line verification: Case studies and experiments. Proceedings of the 35th International Conference on Software Engineering, May 18-26, IEEE Xplore Press, San Francisco, CA, USA, pp: 482-491. DOI: 10.1109/ICSE.2013.6606594

Arrieta, A., G. Sagardui and L. Etxeberria, 2015. Test control algorithms for the validation of cyber-physical systems product lines. Proceedings of the 19th International Conference on Software Product Line, Jul. 20-24, ACM, Nashville, Tennessee, pp: 273-282. DOI: 10.1145/2791060.2791095

Arrieta, A., S. Wang, G. Sagardui and L. Etxeberria, 2016. Search-based test case selection of cyber-physical system product lines for simulation-based validation. Proceedings of the 20th International Systems and Software Product Line Conference, Sept. 16-23, ACM, Beijing, China, pp: 297-306. DOI: 10.1145/2934466.2946046

Asadi, M., G. Groner, B. Mohabbati and D. Gasevic, 2016a. Goal-oriented modeling and verification of feature-oriented product lines. Software Syst. Model., 15: 257-279. DOI: 10.1007/s10270-014-0402-8

Asadi, M., S. Soltani, Gasevic, D. and M. Hatala, 2016b. The effects of visualization and interaction techniques on feature model configuration. Empirical Software Eng., 21: 1706-1743. DOI: 10.1007/s10664-014-9353-5

Asadi, M., S. Soltani, D. Gasevic, M. Hatala and E. Bagheri, 2014. Toward automated feature model configuration with optimizing non-functional requirements. Inform. Software Technol., 56: 1144-1165. DOI: 10.1016/j.infsof.2014.03.005

Bagheri, E., F. Ensan and D. Gasevic, 2012a. Decision support for the software product line domain engineering lifecycle. Automated Software Eng., 19: 335-377. DOI: 10.1007/s10515-011-0099-7

Bagheri, E., F. Ensan and D. Gasevic, 2012b. Grammar-based test generation for software product line feature models. Proceedings of the Conference of the Center for Advanced Studies on Collaborative Research, Nov. 05-07, IBM Corp., Toronto, Ontario, Canada, pp: 87-101.

Bagheri, E. and D. Gasevic, 2011. Assessing the maintainability of software product line feature models using structural metrics. Software Quality J., 19: 579-612. DOI: 10.1007/s11219-010-9127-2

Barreiros, J. and A. Moreira, 2014. A cover-based approach for configuration repair. Proceedings of the 18th International Software Product Line Conference, Sept. 15-19, ACM, Florence, Italy, pp: 157-166. DOI: 10.1145/2648511.2648528

Basili, V.R. and H.D. Rombach, 1988. The TAME project: Towards improvement-oriented software environments. IEEE Trans. Software Eng., 14: 758-773. DOI: 10.1109/32.6156

Becan, G., R. Behjati, A. Gotlieb and M. Acher, 2015. Synthesis of attributed feature models from product descriptions. Proceedings of the 19th International Conference on Software Product Line, Jul. 20-24, ACM, Nashville, Tennessee, pp: 1-10. DOI: 10.1145/2791060.2791068

Ben-David, S., B. Sterin, J.M. Atlee and S. Beidu, 2015. Symbolic model checking of product-line requirements using sat-based methods. Proceedings of the IEEE/ACM 37th IEEE International Conference on Software Engineering, May 16-24, IEEE Xplore Press, Florence, Italy, pp: 189-199. DOI: 10.1109/ICSE.2015.40

Bera, M.H.G., E. OliveiraJr and T.E. Colanzi, 2015. Evidence-based smarty support for variability identification and representation in component models. Proceedings of the 17th International Conference on Enterprise Information Systems, Apr. 27-30, Science and Technology Publications, Barcelona, Spain, pp: 295-302. DOI: 10.5220/0005366402950302

Bodden, E., T. Toledo, M. Ribeiro, C. Brabrand and P. Borba *et al*., 2014. SPL^{LIFT}: Statically analyzing software product lines in minutes instead of years. Proceedings of the 34th ACM SIGPLAN Conference on Programming Language Design and Implementation, Jun. 16-19, ACM, Seattle, Washington, USA, pp: 355-364. DOI: 10.1145/2491956.2491976

Bonifacio, R., P. Borba, C. Ferraz and P. Accioly, 2017. Empirical assessment of two approaches for specifying software product line use case scenarios. Software Syst. Model., 16: 97-123. DOI: 10.1007/s10270-015-0471-3

Burdek, J., T. Kehrer, M. Lochau, D. Reuling and U. Kelter *et al*., 2016. Reasoning about productline evolution using complex feature model differences. Automated Software Eng., 23: 687-733. DOI: 10.1007/s10515-015-0185-3

Burdek, J., M. Lochau, S. Bauregger, A. Holzer and A. Von Rhein *et al*., 2015. Facilitating reuse in multi-goal test-suite generation for software product lines. Proceedings of the International Conference on Fundamental Approaches to Software Engineering, Apr. 11-18, Springer, London, UK, pp: 84-99. DOI: 10.1007/978-3-662-46675-9_6

Calvagna, A., A. Gargantini and P. Vavassori, 2013. Combinatorial testing for feature models using CitLab. Proceedings of the IEEE Sixth International Conference on Software Testing, Verification and Validation Workshops, IEEE Xplore Press, Luxembourg, pp: 338-347. DOI: 10.1109/ICSTW.2013.45

Cavalcanti, R.D.O., E.S.D. Almeida and S.R.L. Meira, 2011. Extending the RiPLE-DE process with quality attribute variability realization. Proceedings of the Federated Events on Component-Based Software Engineering and Software Architecture, Jun. 20-24, ACM, Boulder, Colorado, USA., pp: 159-164. DOI: 10.1145/2000259.2000286

Chiquitto, A.G., I.M.S. Gimenes and E. OliveiraJr, 2015. SyMPLES-CVL: A SysML and CVL based approach for product-line development of embedded systems. Proceedings of the 9th Brazilian Symposium on Components, Architectures and Reuse Software, Sept. 21-22, IEEE Xplore Press, Belo Horizonte, Brazil, pp: 21-30. DOI: 10.1109/SBCARS.2015.13

Cirilo, E., I. Nunes, A. Garcia and C. Lucena, 2011. Configuration knowledge of software product lines: A comprehensibility study. Proceedings of the 2nd International Workshop on Variability and Composition, Mar. 21-21, ACM, Porto de Galinhas, Brazil, pp: 1-5. DOI: 10.1145/1961359.1961361

Clements, P. and L. Northrop, 2002. Software Product Lines: Practices and Patterns. 3rd Edn., Addison-Wesley, Boston, ISBN-10: 0201703327, pp: 563.

Colanzi, T.E. and S.R. Vergilio, 2014a. A comparative analysis of two multi-objective evolutionary algorithms in product line architecture design optimization. Proceedings of the IEEE 26th International Conference on Tools with Artificial Intelligence, Nov. 10-12, IEEE Xplore Press, Limassol, Cyprus, pp: 681-688. DOI: 10.1109/ICTAI.2014.107

Colanzi, T.E. and S.R. Vergilio, 2014b. A feature-driven crossover operator for product line architecture design optimization. Proceedings of the IEEE 38th Annual Computer Software and Applications Conference, Jul. 21-25, IEEE Xplore Press, Vasteras, Sweden, pp: 43-52. DOI: 10.1109/COMPSAC.2014.11

Colanzi, T.E. and S.R. Vergilio, 2016. A feature-driven crossover operator for multi-objective and evolutionary optimization of product line architectures. J. Syst. Software, 121: 126-143. DOI: 10.1016/j.jss.2016.02.026

Conejero, J.M., E. Figueiredo, A. Garcia, J. Hernandez and E. Jurado, 2012. On the relationship of concern metrics and requirements maintainability. Inform. Software Technol., 54: 212-238. DOI: 10.1016/j.infsof.2011.09.003

Cordy, M., P. Heymans, A. Legay, P.Y. Schobbens and B. Dawagne *et al*., 2014. Counterexample guided abstraction refinement of product-line behavioural models. Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering, Nov. 16-21, ACM, Hong Kong, China, pp: 190-201. DOI: 10.1145/2635868.2635919

Cordy, M., P.Y. Schobbens, P. Heymans and A. Legay, 2012. Behavioural modelling and verification of real-time software product lines. Proceedings of the 16th International Software Product Line Conference, Sept. 02-07, ACM, Salvador, Brazil, pp: 66-75. DOI: 10.1145/2362536.2362549

Cordy, M., P.Y. Schobbens, P. Heymans and A. Legay, 2013. Beyond boolean product-line model checking: Dealing with feature attributes and multi-features. Proceedings of the 35th International Conference on Software Engineering, May 18-26, IEEE Xplore Press, San Francisco, CA, USA, pp: 472-481. DOI: 10.1109/ICSE.2013.6606593

Cunha, R., T. Conte, E.S.D. Almeida and J.C. Maldonado, 2012. A set of inspection techniques on software product line models. SEKE.

Denger, C. and R. Kolb, 2006. Testing and inspecting reusable product line components: First empirical results. Proceedings of the International Symposium on Empirical Software Engineering, Sept. 21-22, ACM, Rio de Janeiro, Brazil, pp: 184-193. DOI: 10.1145/1159733.1159762

Dermeval, D., T. Tenorio, I.I. Bittencourt, A. Silva and S. Isotani *et al*., 2015. Ontologybased feature modeling: An empirical study in changing scenarios. Expert Syst. Applic., 42: 4950-4964. DOI: 10.1016/j.eswa.2015.02.020

Dieste, O., A. Grim, N. Juristo and H. Saxena, 2011. Quantitative determination of the relationship between internal validity and bias in software engineering experiments: Consequences for systematic literature reviews. Proceedings of the 5th International Symposium on Empirical Software Engineering and Measurement, Sept. 22-23, IEEE Xplore Press, Banff, AB, Canada, pp: 285-294. DOI: 10.1109/ESEM.2011.37

Dieste, O. and N. Juristo, 2013. Challenges of Evaluating the Quality of Software Engineering Experiments. In: Perspectives on the Future of Software Engineering, Münch, J. and K. Schmid (Eds.), Springer Berlin Heidelberg, pp: 159-177.

Duran-Limon, H.A., C.A. Garcia-Rios, F.E. Castillo-Barrera and R. Capilla, 2015. An ontology-based product architecture derivation approach. IEEE Trans. Software Eng., 41: 1153-1168. DOI: 10.1109/TSE.2015.2449854

Elfaki, A.O., S.L. Fong, P. Vijayaprasad, M.G.M. Johar and M.S. Fadhil, 2014. Using a rule based method for detecting anomalies in software product line. Res. J. Applied Sci. Eng. Technol., 7: 275-81. DOI: 10.19026/rjaset.7.251

Eyal-Salman, H. and A. Seriai, 2016. Toward recovering component-based software product line architecture from object-oriented product variants. Proceedings of the International Conference on Software Engineering and Knowledge Engineering, (EKE' 16), San Francisco, USA, pp: 1-7.

Eyal-Salman, H., A.D. Seriai and C. Dony, 2013. Feature-to-code traceability in legacy software variants. Proceedings of the 39th Euromicro Conference on Software Engineering and Advanced Applications, Sept. 4-6, IEEE Xplore Press, Santander, Spain, pp: 57-61. DOI: 10.1109/SEAA.2013.65

Eyal-Salman, H., A.D. Seriai and C. Dony, 2014. Feature location in a collection of product variants: Combining information retrieval and hierarchical clustering. Proceedings of the 26th International Conference on Software Engineering and Knowledge Engineering, (EKE' 14), Vancouver, Canada, pp: 426-430.

Eyal-Salman, H., A.D. Seriai and C. Dony, 2015. Feature-level change impact analysis using formal concept analysis. Int. J. Software Eng. Knowl. Eng., 25: 69-92. DOI: 10.1142/S0218194015400045

Farias, K., A. Garcia and C. Lucena, 2014. Effects of stability on model composition effort: An exploratory study. Software Syst. Model., 13: 1473-1494. DOI: 10.1007/s10270-012-0308-2

Federle, E.L., T.D.N. Ferreira, T.E. Colanzi and S.R. Vergilio, 2015. Optimizing software product line architectures with OPLA-tool. Proceedings of the 7th International Symposium on Search Based Software Engineering, Sept. 5-7, Bergamo, Italy, pp: 325-331. DOI: 10.1007/978-3-319-22183-0_30

Feigenspan, J., C. Kastner, S. Apel, J. Liebig and M. Schulze *et al*., 2013. Do background colors improve program comprehension in the# ifdef hell? Empirical Software Eng., 18: 699-745. DOI: 10.1007/s10664-012-9208-x

Feigenspan, J., M. Schulze, M. Papendieck, C. Kastner and R. Dachselt *et al*., 2011. Using background colors to support program comprehension in software product lines. Proceedings of the 15th Annual Conference on Evaluation and Assessment in Software Engineering, Apr. 11-12, IEEE Xplore Press, Durham, UK, pp: 66-75. DOI: 10.1049/ic.2011.0008

Feigenspan, J., M. Schulze, M. Papendieck, C. Kastner and R. Dachselt *et al*., 2012. Supporting program comprehension in large preprocessor-based software product lines. IET Software, 6: 488-501. DOI: 10.1049/iet-sen.2011.0172

Fernandes, P., C. Werner and E. Teixeira, 2011. An approach for feature modeling of context-aware software product line. JUCS, 17: 807-829.

Fernandez-Amoros, D., R. Heradio, J.A. Cerrada and C. Cerrada, 2014. A scalable approach to exact model and commonality counting for extended feature models. IEEE Trans. Software Eng., 40: 895-910. DOI: 10.1109/TSE.2014.2331073

Ferreira, F., P. Borba, G. Soares and R. Gheyi, 2012. Making software product line evolution safer. Proceedings of the 6th Brazilian Symposium on Software Components, Architectures and Reuse, Sept. 23-28, IEEE Xplore Press, Natal, Brazil, pp: 21-30. DOI: 10.1109/SBCARS.2012.18

Ferreira, F., R. Gheyi, P. Borba and G. Soares, 2014. A toolset for checking SPL refinements. JUCS, 20: 587-614.

Fleiss, J.L., B. Levin and M.C. Paik, 2003. Statistical Methods for Rates and Proportions. 1st Edn., Wiley Series in Probability and Statistics, ISBN-13: 978-0-471-52629-2, pp: 800.

Galindo, J.A., D. Dhungana, R. Rabiser, D. Benavides, G. Botterweck and P. Grunbacher, 2015. Supporting distributed product configuration by integrating heterogeneous variability modeling approaches. Inform. Software Technol., 62: 78-100. DOI: 10.1016/j.infsof.2015.02.002

Gamez, N. and L. Fuentes, 2013. Architectural evolution of famiware using cardinality-based feature models. Inform. Software Technol., 55: 563-580. DOI: 10.1016/j.infsof.2012.06.012

Gheyi, R., T. Massoni and P. Borba, 2011. Automatically checking feature model refactorings. JUCS, 17: 684-711.

Ghezzi, C. and A.M. Sharifloo, 2013. Model-based verification of quantitative non-functional properties for software product lines. Inform. Software Technol., 55: 508-524. DOI: 10.1016/j.infsof.2012.07.017

Goncalves, T.L., I.M.D.S. Gimenes, M. Fantinato, G.H. Travassos and M.B.F.D. Toledo, 2011. Experimental studies of e-contract establishment in the pl4bpm context. Int. J. Web Eng. Technol., 6: 243-265. DOI: 10.1504/IJWET.2011.040724

Gonzalez-Huerta, J., E. Insfran and S. Abrahao, 2013. Defining and validating a multimodel approach for product architecture derivation and improvement. Proceedings of the 16th International Conference on Model-Driven Engineering Languages and Systems, Sept. 29-Oct. 04, Springer, USA, pp: 388-404. DOI: 10.1007/978-3-642-41533-3_24

Guana, V. and D. Correal, 2013. Improving software product line configuration: A quality attribute-driven approach. Inform. Software Technol., 55: 541-562. DOI: 10.1016/j.infsof.2012.09.007

Guo, J., Y. Wang, P. Trinidad and D. Benavides, 2012. Consistency maintenance for evolving feature models. Expert Syst. Applic., 39: 4987-4998. DOI: 10.1016/j.eswa.2011.10.014

Guo, J., J. White, G. Wang, J. Li and Y. Wang, 2011. A genetic algorithm for optimized feature selection with resource constraints in software product lines. J. Syst. Software, 84: 2208-2221. DOI: 10.1016/j.jss.2011.06.026

Halmans, G. and K. Pohl, 2003. Communicating the variability of a software-product family to customers. Software Syst. Model., 2: 15-36. DOI: 10.1007/s10270-003-0019-9

Hartmann, H. and T. Trew, 2008. Using feature diagrams with context variability to model multiple product lines for software supply chains. Proceedings of the 12th International Software Product Line Conference, Sept. 8-12, IEEE Xplore Press, Limerick, Ireland, pp: 12-21. DOI: 10.1109/SPLC.2008.15

Hashim, N.L., H.W. Schmidt and S. Ramakrishnan, 2005. Test order for class-based integration testing of java applications. Proceedings of the 5th International Conference on Quality Software, Sept. 19-20, IEEE Xplore Press, Melbourne, Victoria, Australia, pp: 11-18. DOI: 10.1109/QSIC.2005.64

Henard, C., M. Papadakis, M. Harman and Y. Le Traon, 2015. Combining multi-objective search and constraint solving for configuring large software product lines. Proceedings of the IEEE/ACM 37th IEEE International Conference on Software Engineering, May 16-24, IEEE Xplore Press, Florence, Italy, pp: 517-528. DOI: 10.1109/ICSE.2015.69

Henard, C., M. Papadakis, G. Perrouin, J. Klein and P. Heymans *et al*., 2014. Bypassing the combinatorial explosion: Using similarity to generate and prioritize t-wise test configurations for software product lines. IEEE Trans. Software Eng., 40: 650-670. DOI: 10.1109/TSE.2014.2327020

Henard, C., M. Papadakis, G. Perrouin, J. Klein and Y. Le Traon, 2013a. Assessing software product line testing via model-based mutation: An application to similarity testing. Proceedings of the IEEE Sixth International Conference on Software Testing, Verification and Validation Workshops, Mar. 18-22, IEEE Xplore Press, Luxembourg, pp: 188-197. DOI: 10.1109/ICSTW.2013.30

Henard, C., M. Papadakis, G. Perrouin, J. Klein and Y.L. Traon, 2013b. Multi-objective test generation for software product lines. Proceedings of the 17th International Software Product Line Conference, Aug. 26-30, ACM, Tokyo, Japan, pp: 62-71. DOI: 10.1145/2491627.2491635

Heradio-Gil, R., D. Fernandez-Amoros, J.A. Cerrada and C. Cerrada, 2011. Supporting commonality-based analysis of software product lines. IET Software, 5: 496-509. DOI: 10.1049/iet-sen.2010.0022

Hervieu, A., B. Baudry and A. Gotlieb, 2011. PACOGEN: Automatic generation of pairwise test configurations from feature models. Proceedings of the IEEE 22nd International Symposium on Software Reliability Engineering, Nov. 29-Dec. 2, IEEE Xplore Press, Hiroshima, Japan, pp: 120-129. DOI: 10.1109/ISSRE.2011.31

Hervieu, A., D. Marijan, A. Gotlieb and B. Baudry, 2016. Practical minimization of pairwise-covering test configurations using constraint programming. Inform. Software Technol., 71: 129-146. DOI: 10.1016/j.infsof.2015.11.007

Hierons, R.M., M. Li, X. Liu, S. Segura and W. Zheng, 2016. Sip: Optimal product selection from feature models using many-objective evolutionary optimization. Trans. Software Eng. Methodol., 25: 1-17. DOI: 10.1145/2897760

Ianzen, A., R.M. Fontana, M.A. Paludo, A. Malucelli and S. Reinehr, 2015. Scoping automation in software product lines. Proceedings of the International Conference on Enterprise Information Systems, (EIS' 15), Barcelona, Spain, pp: 82-91.

Jaksic, A., R.B. France, P. Collet and S. Ghosh, 2014. Evaluating the usability of a visual feature modeling notation. Proceedings of the International Conference on Software Language Engineering, (SLE' 14), Springer, Cham, pp: 122-140.

Jedlitschka, A., M. Ciolkowski and D. Pfahl, 2008. Reporting Experiments in Software Engineering. In: Guide to Advanced Empirical Software Engineering, Shull, F., J. Singer and D.I.K. Sjøberg (Eds.), Springer, London, pp: 201-228.

Jirapanthong, W., 2008. An approach to software artefact specification for supporting product line systems. Software Engineering Research and Practice, pp: 548-554.

Jirapanthong, W. and A. Zisman, 2009. XTraQue: Traceability for product line systems. Software Syst. Model., 8: 117-144. DOI: 10.1007/s10270-007-0066-8

Johansen, M.F., Ø. Haugen and F. Fleurey, 2012. An algorithm for generating t-wise covering arrays from large feature models. Proceedings of the 16th International Software Product Line Conference, Sept. 02-07, ACM, Salvador, Brazil, pp: 46-55. DOI: 10.1145/2362536.2362547

John, I., 2006. Capturing Product Line Information from Legacy User Documentation. In: Software Product Lines: Research Issues in Engineering and Management, Käköla, T. and J.C. Duenas (Eds.), Springer, pp: 127-159.

John, I. and A. Silva, 2011. Evaluating variability instantiation strategies for product lines. Proceedings of the 5th Workshop on Variability Modeling of Software-Intensive Systems, Jan. 27-29, ACM, Namur, Belgium, pp: 105-113. DOI: 10.1145/1944892.1944905

Kamischke, J., M. Lochau and H. Baller, 2012. Conditioned model slicing of feature-annotated state machines. Proceedings of the 4th International Workshop on Feature-Oriented Software Development, Sept. 24-25, ACM, Dresden, Germany, pp: 9-16. DOI: 10.1145/2377816.2377818

Kampenes, V., 2007. Quality of design, analysis and reporting of software engineering experiments: A systematic review. PhD Thesis, University of Oslo.

Kim, C.H.P., D. Marinov, S. Khurshid, D. Batory and S. Souto *et al*., 2013. SPLat: Lightweight dynamic analysis for reducing combinatorics in testing configurable systems. Proceedings of the 9th Joint Meeting on Foundations of Software Engineering, Aug. 18-26, ACM, Saint Petersburg, Russia, pp: 257-267. DOI: 10.1145/2491411.2491459

Kitchenham, B. and S. Charters, 2007. Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report.

Kitchenham, B., D.I.K. Sjøberg, O.P. Brereton, D. Budgen and T. Dyba *et al*., 2010. Can we evaluate the quality of software engineering experiments? Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, Sept. 16-17, ACM, Bolzano-Bozen, Italy, pp: 1-8. DOI: 10.1145/1852786.1852789

Kitchenham, B.A., D. Budgen and P. Brereton, 2015. Evidence-Based Software Engineering and Systematic Reviews. 1st Edn., CRC Press, ISBN-10: 1482228661, pp: 299.

Kitchenham, B.A., S.L. Pfleeger, L.M. Pickard, P.W. Jones and D.C. Hoaglin *et al*., 2002. Preliminary guidelines for empirical research in software engineering. IEEE Trans. Software Eng., 28: 721-734. DOI: 10.1109/TSE.2002.1027796

Kitchenham, B.A., D.I.K. Sjøberg, T. Dyba, D. Pfahl and P. Brereton *et al*., 2012. Three empirical studies on the agreement of reviewers about the quality of software engineering experiments. Inform. Software Technol., 54: 804-819. DOI: 10.1016/j.infsof.2011.11.008

Kolesnikov, S., A. von Rhein, C. Hunsen and S. Apel, 2014. A comparison of product-based, feature based and family-based type checking. Proceedings of the International Conference on Generative Programming: Concepts and Experiences, 49: 115-124. DOI: 10.1145/2637365.2517213

Lamine, S.B.A.B., L.L. Jilani and H.H.B. Ghezala, 2005. A software cost estimation model for a product line engineering approach: Supporting tool and UML modeling. Proceedings of the 3rd ACIS International Conference on Software Engineering Research, Management and Applications, Aug. 11-13, IEEE Xplore Press, Mount Pleasant, MI, USA, pp: 383-390. DOI: 10.1109/SERA.2005.16

Lee, S.B., J.W. Kim, C.Y. Song and D.K. Baik, 2007. An approach to analyzing commonality and variability of features using ontology in a software product line engineering. Proceedings of the 5th ACIS International Conference on Software Engineering Research, Management and Applications, Aug. 20-22, IEEE Xplore Press, Busan, South Korea, pp: 727-734. DOI: 10.1109/SERA.2007.41

Lengauer, P., V. Bitto, F. Angerer, P. Grunbacher and H. Mossenbock, 2014. Where has all my memory gone?: Determining memory characteristics of product variants using virtual-machine-level monitoring. Proceedings of the 8th International Workshop on Variability Modelling of Software-Intensive Systems, Jan. 22-24, ACM, Sophia Antipolis, France, pp: 13-13. DOI: 10.1145/2556624.2556628

Lian, X. and L. Zhang, 2015. Optimized feature selection towards functional and non-functional requirements in software product lines. Proceedings of the IEEE 22nd International Conference on Software Analysis, Evolution and Reengineering, Mar. 2-6, IEEE Xplore Press, Montreal, QC, Canada, pp: 191-200. DOI: 10.1109/SANER.2015.7081829

Neto, L., C. Rodrigues, E.S.D. Almeida and S.R.D.L. Meira, 2012. A software product lines system test case tool and its initial evaluation. Proceedings of the IEEE 13th International Conference on Information Reuse and Integration, Aug. 8-10, IEEE Xplore Press, Las Vegas, NV, USA, pp: 25-32. DOI: 10.1109/IRI.2012.6302986

Neto, L., C. Rodrigues, I.D.C. Machado, V.C. Garcia and E.S.D. Almeida, 2013. Analyzing the effectiveness of a system testing tool for software product line engineering. Proceedings of the 25th International Conference on Software Engineering and Knowledge Engineering, (EKE' 03), At Boston, pp: 584-588.

Linden, F.J.V.D., K. Schmid and E. Rommes, 2007. Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering. 1st Edn., Springer, New York, ISBN-10: 3540714375, pp: 333.

Lity, S., M. Lochau, I. Schaefer and U. Goltz, 2012. Delta-oriented model-based SPL regression testing. Proceedings of the 3rd International Workshop on Product LinE Approaches in Software Engineering, Jun. 4-4, IEEE Xplore Press, Zurich, Switzerland, pp: 53-56. DOI: 10.1109/PLEASE.2012.6229772

LiZhang, X.L., 2014. An evolutionary methodology for optimized feature selection in software product lines. Proceedings of the International Conference on Software Engineering and Knowledge Engineering, (EKE' 14).

Lochau, M., S. Mennicke, H. Baller and L. Ribbeck, 2016. Incremental model checking of delta-oriented software product lines. J. Logical Algebraic Meth. Programm., 85: 245-267. DOI: 10.1016/j.jlamp.2015.09.004

Lopez-Herrejon, R.E., L. Linsbauer, J.A. Galindo, J.A. Parejo and D. Benavides *et al*., 2015. An assessment of search-based techniques for reverse engineering feature models. J. Syst. Software, 103: 353-369. DOI: 10.1016/j.jss.2014.10.037

Machado, I.C., E.S. Almeida, G.S.S. Gomes, P.A.M. Silveira Neto and R.L. Novais *et al*., 2012. A preliminary study on the effects of working with a testing process in software product line projects. Proceedings of the Workshop Latinoamericano de Ingenierıa de Software Experimental, (ISE' 12).

Machado, I.D.C., P.A.D.M. Silveira Neto, E.S.D. Almeida and S.R.D.L. Meira, 2011. RIPLE-TE: A process for testing software product lines. SEKE.

Marcolino, A., E. OliveiraJr and I.M.D.S. Gimenes, 2014a. Variability identification and representation in software product line UML sequence diagrams: Proposal and empirical study. Proceedings of the Brazilian Symposium on Software Engineering, Sept. 28-Oct. 3, IEEE Xplore Press, Maceio, Brazil, pp: 141-150. DOI: 10.1109/SBES.2014.11

Marcolino, A., E. OliveiraJr, I.M.D.S. Gimenes and E.F. Barbosa, 2014b. Empirically based evolution of a variability management approach at UML class level. Proceedings of the IEEE 38th Annual Computer Software and Applications Conference, Jul. 21-25, IEEE Xplore Press, Vasteras, Sweden, pp: 354-363. DOI: 10.1109/COMPSAC.2014.58

Marcolino, A., E. OliveiraJr, I.M.D.S. Gimenes and T.U. Conte, 2013a. Towards validating complexity-based metrics for software product line architectures. Proceedings of the 7th Brazilian Symposium on Software Components, Architectures and Reuse, Sept. 29-Oct. 4, IEEE Xplore Press, Brasilia, Brazil, pp: 69-79. DOI: 10.1109/SBCARS.2013.18

Marcolino, A., E. OliveiraJr, I.M.D.S. Gimenes and J.C. Maldonado, 2013b. Towards the effectiveness of a variability management approach at use case level. Proceedings of the International Conference on Software Engineering and Knowledge Engineering, (EKE' 13), At Boston, pp: 214-219.

Mariani, T., T.E. Colanzi and S.R. Vergilio, 2016. Preserving architectural styles in the search based design of software product line architectures. J. Syst. Software, 115: 157-173. DOI: 10.1016/j.jss.2016.01.039

Mariani, T., S.R. Vergilio and T.E. Colanzi, 2015. Search based design of layered product line architectures. Proceedings of the IEEE 39th Annual Computer Software and Applications Conference, Jul. 1-5, IEEE Xplore Press, Taichung, Taiwan, pp: 270-275. DOI: 10.1109/COMPSAC.2015.30

Marijan, D., A. Gotlieb, S. Sen and A. Hervieu, 2013. Practical pairwise testing for software product lines. Proceedings of the 17th International Software Product Line Conference, Aug. 26-30, ACM, Tokyo, Japan, pp: 227-235. DOI: 10.1145/2491627.2491646

Medeiros, A.L., E. Cavalcante, T. Batista and E. Silva, 2015. ArchSPL-MDD: An ADL-based model-driven strategy for automatic variability management. Proceedings of the 9th Brazilian Symposium on Components, Architectures and Reuse Software, Sept. 21-22, IEEE Xplore Press, Belo Horizonte, Brazil, pp: 120-129. DOI: 10.1109/SBCARS.2015.23

Medeiros, F.M., E.S.D. Almeida and S.R.D.L. Meira, 2010a. Designing a set of service-oriented systems as a software product line. Proceedings of the 4th Brazilian Symposium on Software Components, Architectures and Reuse, Sept. 27-29, IEEE Xplore Press, Bahia, Brazil, pp: 70-79. DOI: 10.1109/SBCARS.2010.17

Medeiros, F.M., E.S.D. Almeida and S.R.L. Meira, 2010b. SOPLE-DE: An approach to design service-oriented product line architectures. Proceedings of the 14th International Conference on Software Product Lines: Going Beyond, Sept. 13-17, Springer, Jeju Island, South Korea, pp: 456-460. DOI: 10.1007/978-3-642-15579-6_36

Medvidovic, N. and R.N. Taylor, 2010. Software architecture: foundations, theory and practice. Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering, May 01-08, ACM, Cape Town, South Africa, pp: 471-472. DOI: 10.1145/1810295.1810435

Mefteh, M., N. Bouassida and H. Ben-Abdallah, 2015. Implementation and evaluation of an approach for extracting feature models from documented UML use case diagrams. Proceedings of the Symposium on Applied Computing, Apr.13-17, ACM, Salamanca, Spain, pp: 1602-1609. DOI: 10.1145/2695664.2695907

Mello, R.M.D., E. Nogueira, M. Schots, C.M.L. Werner and G.H. Travassos, 2014. Verification of software product line artefacts: A checklist to support feature model inspections. JUCS, 20: 720-745.

Michalik, B., D. Weyns, N. Boucke and A. Helleboogh, 2011a. Supporting online updates of software product lines: A controlled experiment. Empirical Software Engineering and Measurement, pp: 187-196.

Michalik, B., D. Weyns, N. Boucke and A. Helleboogh, 2011b. Supporting online updates of software product lines: A controlled experiment. Proceedings of the International Symposium on Empirical Software Engineering and Measurement, Sept. 22-23, IEEE Xplore Press, Banff, AB, Canada, pp: 187-196. DOI: 10.1109/ESEM.2011.27

Millo, J.V. and S. Ramesh, 2012. Relating requirement and design variabilities. Proceedings of the 19th Asia-Pacific Software Engineering Conference, Dec. 4-7, IEEE Xplore Press, Hong Kong, China, pp: 35-42. DOI: 10.1109/APSEC.2012.67

Murashkin, A., M. Antkiewicz, D. Rayside and K. Czarnecki, 2013a. Visualization and exploration of optimal variants in product line engineering. Proceedings of the 17th International Software Product Line Conference, Aug. 26-30, ACM, Tokyo, Japan, pp: 111-115. DOI: 10.1145/2491627.2491647

Murashkin, A., M. Antkiewicz, D. Rayside and K. Czarnecki, 2013b. Visualization and exploration of optimal variants in product line engineering. Proceedings of the 17th International Software Product Line Conference, Aug. 26-30, ACM, Tokyo, Japan, pp: 111-115. DOI: 10.1145/2491627.2491647

Neiva, D.F.S., E.S.D. Almeida and S.R.D.L. Meira, 2009. An experimental study on requirements engineering for software product lines. Proceedings of the 35th Euromicro Conference on Software Engineering and Advanced Applications, Aug. 27-29, IEEE Xplore Press, Patras, Greece, pp: 251-254. DOI: 10.1109/SEAA.2009.32

Nie, K., L. Zhang and Z. Geng, 2012. Product line variability modeling based on model difference and merge. Proceedings of the IEEE 36th Annual Computer Software and Applications Conference Workshops, Jul. 16-20, IEEE Xplore Press, Izmir, Turkey, pp: 509-513. DOI: 10.1109/COMPSACW.2012.95

Niu, N., J. Savolainen, Z. Niu, M. Jin and J.R.C. Cheng, 2014. A systems approach to product line requirements reuse. IEEE Syst. J., 8: 827-836. DOI: 10.1109/JSYST.2013.2260092

Nunes, C., U. Kulesza, C. Sant'Anna, I. Nunes and A.F. Garcia *et al*., 2009. Assessment of the design modularity and stability of multi-agent system product lines. JUCS, 15: 2254-2283.

Olaechea, R., D. Rayside, J. Guo and K. Czarnecki, 2014. Comparison of exact and approximate multi-objective optimization for software product lines. Proceedings of the 18th International Software Product Line Conference, Sept. 15-19, ACM, Florence, Italy, pp: 92-101. DOI: 10.1145/2648511.2648521

Oliveira, D.R.F., B.L.D. Bezerra, E.L.S.X. Freitas and A.M.A. Maciel, 2015. Adoption of software product line to a voice user interface environment. Proceedings of the International Conference on Software Engineering and Knowledge Engineering, Proceedings of the International Conference on Software Engineering and Knowledge Engineering, (EKE' 15).

Oliveira, R.P.D. and E.S.D. Almeida, 2015. Requirements evolution in software product lines: An empirical study. Proceedings of the 9th Brazilian Symposium on Components, Architectures and Reuse Software, Sept. 21-22, IEEE Xplore Press, Belo Horizonte, Brazil, pp: 1-10. DOI: 10.1109/SBCARS.2015.11

OliveiraJr, E., I.M.D.S. Gimenes and J.C. Maldonado, 2012. Empirical validation of variability-based complexity metrics for software product line architecture. SEKE.

OliveiraJr, E., J.C. Maldonado and I.M.D.S. Gimenes, 2010. Empirical validation of complexity and extensibility metrics for software product line architectures. Proceedings of the 4th Brazilian Symposium on Software Components, Architectures and Reuse, Sept. 27-29, IEEE Xplore Press, Bahia, Brazil, pp: 31-40. DOI: 10.1109/SBCARS.2010.13

Pascual, G.G., R.E. Lopez-Herrejon, M. Pinto, L. Fuentes and A. Egyed, 2015. Applying multiobjective evolutionary algorithms to dynamic software product lines for reconfiguring mobile applications. J. Syst. Software, 103: 392-411. DOI: 10.1016/j.jss.2014.12.041

Patel, S., P. Gupta and V. Shah, 2013a. Combinatorial interaction testing with multi-perspective feature models. Proceedings of the IEEE 6th International Conference on Software Testing, Verification and Validation Workshops, Mar. 18-22, IEEE Xplore Press, Luxembourg, pp: 321-330. DOI: 10.1109/ICSTW.2013.43

Patel, S., P. Gupta and V. Shah, 2013b. Feature interaction testing of variability intensive systems. Proceedings of the 4th International Workshop on Product LinE Approaches in Software Engineering, May 20-20, IEEE Xplore Press, San Francisco, CA, USA, pp: 53-56. DOI: 10.1109/PLEASE.2013.6608666

Petersen, K., S. Vakkalanka and L. Kuzniarz, 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. Inform. Software Technol., 64: 1-18. DOI: 10.1016/j.infsof.2015.03.007

Pohl, K., G. Bockle and F.J.V.D. Linden, 2005. Software Product Line Engineering: Foundations, Principles and Techniques. 1st Edn., Springer, New York, ISBN-10: 3540243720, pp: 467.

Pohl, R., K. Lauenroth and K. Pohl, 2011. A performance comparison of contemporary algorithmic approaches for automated analysis operations on feature models. Proceedings of the 26th IEEE/ACM International Conference on Automated Software Engineering, Nov. 6-10, IEEE Xplore Press, Lawrence, KS, USA, pp: 313-322. DOI: 10.1109/ASE.2011.6100068

Pohl, R., V. Stricker and K. Pohl, 2013. Measuring the structural complexity of feature models. Proceedings of the 28th IEEE/ACM International Conference on Automated Software Engineering, Nov. 11-15, IEEE Xplore Press, Silicon Valley, CA, USA, pp: 454-464. DOI: 10.1109/ASE.2013.6693103

R Core Team, 2014. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Rahmat, A., S. Kassim, M.H. Selamat and S. Hassan, 2016. Actor in multi product line. Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication, Jan. 04-06, ACM, Danang, Viet Nam, pp: 61-61. DOI: 10.1145/2857546.2857608

Reinhartz-Berger, I., K. Figl and Ø. Haugen, 2014a. Comprehending feature models expressed in CVL. Proceedings of the 17th International Conference on Model Driven Engineering Languages and Systems, Sept. 28-Oct. 3, Valencia, Spain, pp: 501-517. DOI: 10.1007/978-3-319-11653-2_31

Reinhartz-Berger, I. and A. Sturm, 2014b. Comprehensibility of UML-based software product line specifications. Empirical Software Eng., 19: 678-713. DOI: 10.1007/s10664-012-9234-8

Reinhartz-Berger, I. and A. Tsoury, 2011. Experimenting with the comprehension of feature-oriented and UML-based core assets. Proceedings of the International Workshop on Business Process Modeling, Development and Support, (MDS' 11), Springer, pp: 468-482. DOI: 10.1007/978-3-642-21759-3_34

Reuling, D., J. Burdek, S. Rotarmel, M. Lochau and U. Kelter, 2015. Fault-based product-line testing: Effective sample generation based on feature-diagram mutation. Proceedings of the 19th International Conference on Software Product Line, Jul. 20-24, ACM, Nashville, Tennessee, pp: 131-140. DOI: 10.1145/2791060.2791074

Ribeiro, M., P. Borba and C. Kastner, 2014. Feature maintenance with emergent interfaces. Proceedings of the 36th International Conference on Software Engineering, May 31-Jun. 07, ACM, Hyderabad, India, pp: 989-1000. DOI: 10.1145/2568225.2568289

Rodrigues, G.N., V. Alves, V. Nunes, A. Lanna and M. Cordy *et al*., 2015. Modeling and verification for probabilistic properties in software product lines. Proceedings of the IEEE 16th International Symposium on High Assurance Systems Engineering, Jan. 8-10, IEEE Xplore Press, Daytona Beach Shores, FL, USA, pp: 173-180. DOI: 10.1109/HASE.2015.34

Rodrigues, I.P., A.P.T. Bacelo, M.S. Silveira, M.D.B. Campos and E.M. Rodrigues, 2016. Evaluating the representation of user interface elements in feature models: An empirical study. SEKE.

Romanovsky, K., D. Koznov and L. Minchin, 2011. Refactoring the documentation of software product lines. Proceedings of the 3rd IFIP TC 2 Central and East European Conference on Software Engineering Techniques, Oct. 13-15, Springer, Brno, Czech Republic, pp: 158-170. DOI: 10.1007/978-3-642-22386-0_12

Saeed, M., F. Saleh, S. Al-Insaif and M. El-Attar, 2016. Empirical validating the cognitive effectiveness of a new feature diagrams visual syntax. Inform. Software Technol., 71: 1-26. DOI: 10.1016/j.infsof.2015.10.012

Santos, A.R., I.D.C. Machado and E.S.D. Almeida, 2016. RiPLE-Hc: Javascript systems meets SPL composition. Proceedings of the 20th International Systems and Software Product Line Conference, Sept. 16-23, ACM, Beijing, China, pp: 154-163. DOI: 10.1145/2934466.2934486

Santos, I.D.S., R.M.D.C. Andrade and P.D.A.D. Santos Neto, 2013. A use case textual description for context aware SPL based on a controlled experiment. CaiSE Forum.

Santos, W.B., E.S.D. Almeida and S.R.D.L. Meira, 2012. TIRT: A traceability information retrieval tool for software product lines projects. Proceedings of the 38th Euromicro Conference on Software Engineering and Advanced Applications, Sept. 5-8, IEEE Xplore Press, Cesme, Izmir, Turkey, pp: 93-100. DOI: 10.1109/SEAA.2012.40

Santos Neto, P.D.A., R. Britto, R.D.A.L. Rabelo, J.J.D.A. Cruz and W.A.L. Lira, 2016. A hybrid approach to suggest software product line portfolios. Applied Soft Comput., 49: 1243-1255. DOI: 10.1016/j.asoc.2016.08.024

Sayyad, A.S., T. Menzies and H. Ammar, 2013. On the value of user preferences in search-based software engineering: A case study in software product lines. Proceedings of the 35th International Conference on Software Engineering, May 18-26, IEEE Xplore Press, San Francisco, CA, USA, pp: 492-501. DOI: 10.1109/ICSE.2013.6606595

Schroter, R., N. Siegmund, T. Thum and G. Saake, 2014. Feature-context interfaces: Tailored programming interfaces for software product lines. Proceedings of the 18th International Software Product Line Conference, Sept. 15-19, ACM, Florence, Italy, pp: 102-111. DOI: 10.1145/2648511.2648522

Segura, S., D. Benavides and A. Ruiz-Cortes, 2011. Functional testing of feature model analysis tools: A test suite. IET Software, 5: 70-82. DOI: 10.1049/iet-sen.2009.0096

Segura, S., J.A. Parejo, R.M. Hierons, D. Benavides and A. Ruiz-Cortes, 2014. Automated generation of computationally hard feature models using evolutionary algorithms. Expert Syst. Applic., 41: 3975-3992. DOI: 10.1016/j.eswa.2013.12.028

Shatnawi, A., A. Seriai and H.A. Sahraoui, 2015. Recovering architectural variability of a family of product variants. Proceedings of the 14th International Conference on Software Reuse, Jan. 4-6, Springer, Miami, FL, USA, pp: 17-33. DOI: 10.1007/978-3-319-14130-5_2

Shi, R., J. Guo and Y. Wang, 2010. A preliminary experimental study on optimal feature selection for product derivation using knapsack approximation. Progress Inform. Comput., 1: 665-669.

Siegmund, N., M. RosenmuLler, C. KaStner, P.G. Giarrusso and S. Apel *et al*., 2013. Scalable prediction of non-functional properties in software product lines: Footprint and memory consumption. Inform. Software Technol., 55: 491-507. DOI: 10.1016/j.infsof.2012.07.020

Silveira Neto, P.A.D.M., I.D.C. Machado, Y.C. Cavalcanti, E.S.D. Almeida and V.C. Garcia *et al*., 2010. A regression testing approach for software product lines architectures. Proceedings of the 4th Brazilian Symposium on Software Components, Architectures and Reuse, Software Components, Architectures and Reuse, Sept. 27-29, IEEE Xplore Press, Bahia, Brazil, pp: 41-50. DOI: 10.1109/SBCARS.2010.14

Silveira Neto, P.A.D.M., I.D.C. Machado, Y.C. Cavalcanti, E.S.D. Almeida and V.C. Garcia *et al*., 2012. An experimental study to evaluate a SPL architecture regression testing approach. Proceedings of the IEEE 13th International Conference on Information Reuse and Integration, Aug. 8-10, IEEE Xplore Press, Las Vegas, NV, USA, pp: 608-615. DOI: 10.1109/IRI.2012.6303065

Sinnema, M. and S. Deelstra, 2008. Industrial validation of covamof. J. Syst. Software, 81: 584-600. DOI: 10.1016/j.jss.2007.06.002

Sjoberg, D.I.K., T. Dyba and M. Jorgensen, 2007. The future of empirical methods in software engineering research. Proceedings of the Future of Software Engineering, May 23-25, IEEE Xplore Press, Minneapolis, MN, USA, pp: 358-378. DOI: 10.1109/FOSE.2007.30

Soltani, S., M. Asadi, D. Gasevic, M. Hatala and E. Bagheri, 2012. Automated planning for feature model configuration based on functional and non-functional requirements. Proceedings of the 16th International Software Product Line Conference, Sept. 02-07, ACM, Salvador, Brazil, pp: 56-65. DOI: 10.1145/2362536.2362548

Sousa, G., W. Rudametkin and L. Duchien, 2016. Extending feature models with relative cardinalities. Proceedings of the 20th International Systems and Software Product Line Conference, Sept. 16-23, ACM, Beijing, China, pp: 79-88. DOI: 10.1145/2934466.2934475

Souto, S., D. Gopinath, M. d'Amorim, D. Marinov and S. Khurshid *et al*., 2015. Faster bug detection for software product lines with incomplete feature models. Proceedings of the 19th International Conference on Software Product Line, Jul. 20-24, ACM, Nashville, Tennessee, pp: 151-160. DOI: 10.1145/2791060.2791093

Souza, I.S., R.M.D. Mello, E.S.D. Almeida, C.M.L. Werner and G.H. Travassos, 2016a. Experimental evaluation of FMCheck: A replication study. Proceedings of the 15th Brazilian Symposium on Software Quality, (SSQ' 16), At Maceió, pp: 121-135.

Souza, M.L.D.J., A.R. Santos, I.D.C. Machado, E.S.D. Almeida and G.S.D.S. Gomes, 2016b. Evaluating variability modeling techniques for dynamic software product lines: A controlled experiment. Proceedings of the 10th Brazilian Symposium on Software Components, Architectures and Reuse, Sept. 19-20, IEEE Xplore Press, Maringa, Brazil, pp: 1-10. DOI: 10.1109/SBCARS.2016.15

Stein, J., I. Nunes and E. Cirilo, 2014. Preference-based feature model configuration with multiple stakeholders. Proceedings of the 18th International Software Product Line Conference, Sept. 15-19, ACM, Florence, Italy, pp: 132-141. DOI: 10.1145/2648511.2648525

Stricker, V., A. Metzger and K. Pohl, 2010. Avoiding redundant testing in application engineering. Proceedings of the International Conference on Software Product Lines, (SPL' 10), Springer, Berlin, pp: 226-240. DOI: 10.1007/978-3-642-15579-6_16

Tan, L., Y. Lin, H. Ye and G. Zhang, 2013. Improving product configuration in software product line engineering. Proceedings of the 36th Australasian Computer Science Conference, Jan. 29-Feb. 01, Australian Computer Society, Inc., pp: 125-133. https://dl.acm.org/citation.cfm?id=2525415

Thum, T., D. Batory and C. Kastner, 2009. Reasoning about edits to feature models. Proceedings of the IEEE 31st International Conference on Software Engineering, May 16-24, IEEE Xplore Press, Vancouver, BC, Canada, pp: 254-264. DOI: 10.1109/ICSE.2009.5070526

Thum, T., J. Meinicke, F. Benduhn, M. Hentschel and A. von Rhein *et al*., 2014. Potential synergies of theorem proving and model checking for software product lines. Proceedings of the 18th International Software Product Line Conference, Sept. 15-19, ACM, Florence, Italy, pp: 177-186. DOI: 10.1145/2648511.2648530

Thum, T., M. Ribeiro, R. Schroter, J. Siegmund and F. Dalton, 2016. Product-line maintenance with emergent contract interfaces. Proceedings of the 20th International Systems and Software Product Line Conference, Sept. 16-23, ACM, Beijing, China, pp: 134-143. DOI: 10.1145/2934466.2934471

Thurimella, A.K. and B. BruGge, 2013. A mixed-method approach for the empirical evaluation of the issue-based variability modeling. J. Syst. Software, 86: 1831-1849. DOI: 10.1016/j.jss.2013.01.038

Uzuncaova, E., D. Garcia, S. Khurshid and D. Batory, 2008. Testing software product lines using incremental test generation. Proceedings of the 19th International Symposium on Software Reliability Engineering, Nov. 10-14, IEEE Xplore Press, Seattle, WA, USA, pp: 249-258. DOI: 10.1109/ISSRE.2008.56

Uzuncaova, E., S. Khurshid and D. Batory, 2010. Incremental test generation for software product lines. IEEE Trans. Software Eng., 36: 309-322. DOI: 10.1109/TSE.2010.30

Villela, K., J. Dorr and I. John, 2010. Evaluation of a method for proactively managing the evolving scope of a software product line. Proceedings of the International Working Conference on Requirements Engineering: Foundation for Software Quality, (FSQ' 10), Springer, Berlin, pp: 113-127. DOI: 10.1007/978-3-642-14192-8_13

Wang, S., S. Ali and A. Gotlieb, 2015. Cost-effective test suite minimization in product lines using search techniques. J. Syst. Software, 103: 370-391. DOI: 10.1016/j.jss.2014.08.024

Wang, S., D. Buchmann, S. Ali, A. Gotlieb and D. Pradhan *et al*., 2014. Multi-objective test prioritization in software product line testing: An industrial case study. Proceedings of the 18th International Software Product Line Conference, Sept. 15-19, ACM, Florence, Italy, pp: 32-41. DOI: 10.1145/2648511.2648515

Weiss, D.M. and C.T.R. Lai, 1999. Software Product-line Engineering: A Family-based Software Development Process. 1st Edn., Addison-Wesley Reading, ISBN-10: 0201694387, pp: 426.

White, J., D. Benavides, D.C. Schmidt, P. Trinidad and B. Dougherty *et al*., 2010. Automated diagnosis of feature model configurations. J. Syst. Software, 83: 1094-1107. DOI: 10.1016/j.jss.2010.02.017

White, J., J.A. Galindo, T. Saxena, B. Dougherty and D. Benavides *et al*., 2014. Evolving feature model configurations in software product lines. J. Syst. Software, 87: 119-136. DOI: 10.1016/j.jss.2013.10.010

White, J., D.C. Schmidt, E. Wuchner and A. Nechypurenko, 2007. Automating product-line variant selection for mobile devices. Proceedings of the 11th International Software Product Line Conference, Sept. 10-14, IEEE Xplore Press, Kyoto, Japan, pp: 129-140. DOI: 10.1109/SPLINE.2007.19

White, J., D.C. Schmidt, E. Wuchner and A. Nechypurenko, 2008. Automatically composing reusable software components for mobile devices. J. Brazilian Comput. Society, 14: 25-44. DOI: 10.1007/BF03192550

Wohlin, C., P. Runeson, M. Host, M.C. Ohlsson and B. Regnell *et al*., 2012. Experimentation in Software Engineering. 1st Edn., Springer Science and Business Media, New York, ISBN-10: 3642290442, pp: 236.

Yoshimura, K., T. Forster, D. Muthig and D. Pech, 2008. Model-based design of product line components in the automotive domain. Proceedings of the 12th International Software Product Line Conference, Sept. 8-12, IEEE Xplore Press, Limerick, Ireland, pp: 170-179. DOI: 10.1109/SPLC.2008.20

Yu, L., F. Duan, Y. Lei, R.N. Kacker and D.R. Kuhn, 2014. Combinatorial test generation for software product lines using minimum invalid tuples. Proceedings of the IEEE 15th International Symposium on High-Assurance Systems Engineering, Jan. 9-11, IEEE Xplore Press, Miami Beach, FL, USA, pp: 65-72. DOI: 10.1109/HASE.2014.18

Appendix A: Quality evaluation checklist for SE experiments proposed by Kitchenham and Charters (2007)

Table 6: Part 1 of quality evaluation checklist for SE experiments proposed by Kitchenham and Charters (2007)

Question	Quantitative Empirical Studies (no specific type)	Correlation (observational studies)	Surveys	Experiments	Source
Design					
Are the aims clearly stated?	X	X	X	X	[11], [10]
Was the study designed with these questions in mind?			X		[25]
Do the study measures allow the questions to be answered?			X	X	[10], [25]
What population was being studied?			X		[25]
Who was included?			X		[12]
Who was excluded?			X		[12]
How was the sample obtained (e.g. postal, interview, web-based)?			X		[10], [12], [25]
Is the survey method likely to have introduced significant bias?			X		[25]
Is the sample representative of the population to which the results will generalise?			X	X	[10], [25]
Were treatments randomly allocated?			X		[10]
Is there a comparison or control group?	X		X	X	[12]
If there is a control group, are participants similar to the treatment group participants in terms of variables that may affect study outcomes?	X		X	X	[10], [12]

Was the sample size justified	X		X	X	[10], [12]
If the study involves assessment of a technology, is the technology clearly defined?	X	X	X	X	[11]
Could the choice of subjects influence the size of the treatment effect?				X	[10], [11], [19],[25]
Could lack of blinding introduce bias?				X	[10]
Are the variables used in the study adequately measured (i.e. are the variables likely to be valid and reliable)?	X	X	X	X	[10], [11], [19],[25]
Are the measures used in the study fully defined?	X	X	X	X	[11]

Table 7: Part 2 of quality evaluation checklist for SE experimen2t5s proposed by Kitchenham and Charters (2007)

Are the measures used in the study the most relevant ones for answering the research questions?	X	X	X	X	[11], [19], [25]
Is the scope (size and length) of the study sufficient to allow for changes in the outcomes of interest to be identified?	X		X	X	[19], [12], [25]
Conduct					
Did untoward events occur during the study?	X	X	X	X	[10]
Was outcome assessment blind to treatment group?	X			X	[19], [12], [25]
Are the data collection methods adequately described?	X	X	X	X	[11]
If two groups are being compared, were they treated similarly within the study?				X	[12], [25]
If the study involves participants over time, what proportion of people who enrolled at the beginning dropped out?	X		X	X	[10], [11]
How was the randomisation carried out?				X	[10]
Analysis					
What was the response rate?			X		[10], [25]
Was the denominator (i.e. the population size) reported?			X		[25]
Do the researchers explain the data types (continuous, ordinal, categorical)?	X	X	X	X	[11]
Are the study participants or observational units adequately described? For example, SE experience, type (student, practitioner, consultant), nationality, task experience and other relevant variables.	X	X	X	X	[12], [25]
Were the basic data adequately described?	X	X	X	X	[10]
Have “drop outs” introduced bias?	X		X	X	[11], [12], [25]
Are reasons given for refusal to participate?	X		X	X	[11]
Are the statistical methods described?	X	X	X	X	[10], [11], [19]
Is the statistical program used to analyse the data referenced?	X	X	X	X	[11]
Are the statistical methods justified?	X	X	X	X	[11]
Is the purpose of the analysis clear?	X	X	X	X	[11]
Are scoring systems described?	X			X	[11]
Are potential confounders adequately controlled for in the analysis?	X	X	X	X	[11]
Do the numbers add up across different tables and	X	X	X	X	[10], [11]

Table 8: Part 3 of quality evaluation checklist for SE experiments proposed by Kitchenham and Charters (2007)

subgroups?					
If different groups were different at the start of the study or treated differently during the study, was any attempt made to control for these differences, either statistically or by matching?	X		X	X	[12], [25]
If yes, was it successful?	X		X	X	[25]
Was statistical significance assessed?	X	X	X	X	[10]
If statistical tests are used to determine differences, is the actual p value given?	X	X	X	X	[11]
If the study is concerned with differences among groups, are confidence limits given describing the magnitude of any observed differences?	X		X	X	[11]
Is there evidence of multiple statistical testing or large numbers of post hoc analysis?	X	X	X	X	[10], [25]
How could selection bias arise?	X		X	X	[10], [25]
Were side-effects reported?					[10]
Conclusions					
Are all study questions answered?	X	X	X	X	[11]
What do the main findings mean?	X	X	X	X	[10]
Are negative findings presented?	X	X	X	X	[11]
If statistical tests are used to determine differences,	X	X	X	X	[11]

is practical significance discussed?					
If drop outs differ from participants, are limitations to the results discussed?	X		X	X	[11]
How are null findings interpreted? (I.e. has the possibility that the sample size is too small been considered?)	X	X	X	X	[10], [12]
Are important effects overlooked?	X	X	X	X	[10]
How do results compare with previous reports?	X	X	X	X	[10]
How do the results add to the literature?	X	X	X	X	[12]
What implications does the report have for practice?	X	X	X	X	[10]
Do the researchers explain the consequences of any problems with the validity/reliability of their measures?	X	X	X	X	[11]

Appendix B: Quality evaluation checklist for SE experiments proposed by Kampenes (2007)

Table 9: Quality evaluation checklist for SE experiments proposed by Kampenes (2007)

		Extent of reporting. Number of experiments		
Information attributes	Variables	N	Total	%
Subjects	Sample size	113	113	100
	Mortality rate	24	113	21.2
	Type (student/professionals)	112	113	99.1
	Recruitment (Voluntarily/mandatory)	41	113	36.3
	Some kind of background information	99	113	87.6
	- Programming experience	37	113	32.7
	- Work experience	24	113	21.2
	- Task related experience	80	113	70.8
	- Grades	6	113	5.3
Experimental setting	Task	113	113	100.0
	Duration	69	113	61.1
	Application system	101	113	89.4
	Size of materials	67	113	59.3
	Location	40	113	35.4
	The use of tools	62	113	54.9
Design and analysis	Well-defined population	1	113	0.9
	Statistical power	1	92	1.1
	Effect size *	27	92	29.3
	Information available for estimation of at least one effect size	64	92	69.6
	Assignment procedure (randomized or quasi)	86	113	76.1
	Randomization method	3	66	4.5
Validity/limitations	Discussion of internal validity	71	113	62.8
	Threats to internal validity	26	113	23.0
	Discussion of external validity	78	113	69.0
	Discussing of statistical conclusion validity†	5	99	5.1
	Discussion of construct validity†	12	113	10.6

Note: Which experiments and articles that are included in these assessments is described in Appendix A; * Extent of reporting refers to the number of experiments with at least one effect size reported; † The number of experiments that discuss statistical conclusion validity and/or construct validity is based on the explicit use of these terms. The reporting of these types of validity needs to be investigated more thoroughly in future work.

Appendix C: Quality evaluation checklist for SE experiments proposed by Kitchenham et al. (2010)

Table 10: Quality evaluation checklist for SE experiments proposed by Kitchenham *et al*. (2010)

# Question	Things to consider
Category: Questions on Aims	
1. Do the authors clearly state the aims of the research?	*Do the authors state research questions, e.g., related to time-to-market, cost, product quality, process quality, developer productivity and developer skills?* *Do the authors state hypotheses and their underlying theories?*
Category: Questions on Design, Data Collection and Data Analysis	
2. Do the authors describe the sample and experimental units (=experimental	*Do the authors explain how experimental units were defined and selected?* *Do the authors state to what degree the experimental units are representative?* *Do the authors explain why the experimental units they selected were the most appropriate for*

materials and participants as individuals or teams)?	*providing insight into the type of knowledge sought by the experiment?* *Do the authors report the sample size?*
3. Do the authors describe the design of the experiment?	*Do the authors clearly describe the chosen design (blocking, within or between subject design, do treatments have levels)?* *Do the authors define/describe all treatments and all controls?*
4. Do the authors describe the data collection procedures and define the measures?	*Are all measures clearly defined (e.g., scale, unit, counting rules)?* *Is the form of the data clear (e.g., tape recording, video material, notes, etc.)?* *Are quality control methods used to ensure consistency, completeness and accuracy of collected data?* *Do the authors report drop-outs?*
5. Do the authors define the data analysis procedures?	*Do authors justify their choice/describe the procedures/provide references to descriptions of the procedures?* *Do the authors report significance levels and effect sizes?* *If outliers are mentioned and excluded from the analysis, is this justified?* *Do the authors report or give references to raw data and/or descriptive statistics?*
6. Do the authors discuss potential experimenter bias?	*Were the authors the developers of some or all of the treatments? If yes, do the authors discuss the implications anywhere in the paper? (If the authors developed the treatments (or parts of them) without discussing the implications, the answer to question 6 is "not at all".)* *Was there random allocation to treatments?* *Was training and conduct equivalent for all treatment groups?* *Was there allocation concealment, i.e., did the researchers know to what treatment each subject was assigned?*
7. Do the authors discuss the limitations of their study?	*Do the authors discuss external validity with respect to subjects, materials and tasks?* *If the study was a quasi-experiment, do the authors discuss the design components that were used to address any study weaknesses?* *If the study used novel measures, is the construct validity of the measures discussed?*
Category: Questions on Study Outcome	
8. Do the authors state the findings clearly?	*Do the authors present results clearly?* *Do the authors present conclusions clearly?* *Are the conclusions warranted by the results and are the connections between the results and conclusions presented clearly?* *Do the authors discuss their conclusions in relation to the original research questions?* *Are limitations of the study discussed explicitly?*
9. Is there evidence that the E/QE can be used by other researchers / practitioners?	*Do the authors discuss whether or how the findings can be transferred to other populations, or consider other ways in which the research can be used?* *To what extent do authors interpret results in the context of other studies / the existing body of knowledge?*

Appendix D: Quality scale to evaluate for SE experiments proposed by Dieste et al. (2011)

Table 11: Quality scale to evaluate for SE experiments proposed by Dieste *et al*. (2011)

Dimension	Question	Recommendation
Experimental Context	Does the introduction contain the industrial context (entities, attributes and measures) and description of the techniques to be reviewed? For experiments that evaluate techniques developed in industry. (**Q1**)	In experiments evaluating techniques developed in industry, experimenters should understand how the technique works in the industrial setting before developing a version of the technique for experimental purposes. This is due to the fact that techniques developed in industrial settings are highly complex, and such complexity is difficult to reproduce in academia. The treatments that are tested in an experiment must be well defined in the report for the experiment to be able to be replicated or simply for the results to be able to be transferred to industry.
	Does the report summarize and discuss earlier similar experiments that have been conducted? (**Q2**)	Describing earlier research that is similar to this study and how they are related can help to build an integrated body of knowledge about a phenomenon in SE.
	Are the hypotheses being laid and are they synonymous with the goal discussed before in introduction? (**Q3**)	Specific hypotheses that are being tested in the study should be clearly established beforehand based on a theory.
Experimental Design	Does the researcher define the population from which objects and subjects are drawn? (**Q4**)	It is necessary to define the population from which the subjects and objects have been extracted to be able to extract inferences from the experimental results.
	Does the researcher define the process by which he applies the treatment to objects and subjects (e.g. randomization)? (**Q6**)	The subjects and objects should be allocated to the treatments in an unbiased manner so as not to compromise the experiment.
	Was randomization used for selecting the population and applying the treatment? (**Q7**)	The subjects and objects should be representative of the population to be able to extract conclusions from the experimental results.
	Does the researcher define the process from which the objects and subjects are selected (e.g. random sampling)? (**Q5**)	
	Is an appropriate blinding procedure used (e.g. blind allocation of materials, blind marking)? (**Q10**)	A double-blinding procedure, as run in medicine, is not possible in SE experiments, but other types of blinding are; these types of blinding can be applied to the allocation of materials, marking and analysis.
Analysis	Is an appropriate blinding procedure used	The information on treatments should be somehow encoded to prevent

	(e.g. blind allocation of materials, blind marking)? (**Q10**)	analysts from knowing the treatment to which it corresponds and being able to introduce bias into the results of the analysis.
Presentation of results	Are the statistical significances mentioned with the results. (**Q9**)	The experiment should report the quantitative data including the effect size and the confidence limits.
Interpretation of results	Is mention made of the threats to validity and also how these threats affect the results and findings? (**Q8**)	Experimenters should discuss the limits of the study, at least threats related to internal and external validity.

Appendix E: Selected primary studies

Table 12: Selected Primary Studies

ID	Title	Author(s)	Year	Publication Type	Publication Venue
S1	A Comparative Analysis of Two Multi-objective Evolutionary Algorithms in Product Line Architecture Design Optimization	Colanzi and Vergilio (2014a)	2014	Conference	ICTAI
S2	A Comparison of Product-based, Featurebased, and Family-based Type Checking	Kolesnikov *et al*. (2014)	2014	Conference	GPCE
S3	A Cover-based Approach for Configuration Repair	Barreiros and Moreira (2014)	2014	Conference	SPLC
S4	A feature-driven crossover operator for multi-objective and evolutionary optimization of product line architectures	Colanzi and Vergilio (2016)	2016	Journal	JSS
S5	A Feature-Driven Crossover Operator for Product Line Architecture Design Optimization	Colanzi and Vergilio (2014b)	2014	Conference	COMPSAC
S6	A genetic algorithm for optimized feature selection with resource constraints in software product lines	Guo *et al*. (2011)	2011	Journal	JSS
S7	A hybrid approach to suggest software product line portfolios	Santos Neto *et al*. (2016)	2016	Journal	Applied Soft Computing
S8	A mixed-method approach for the empirical evaluation of the issuebased variability modeling	Thurimella and BruGge (2013)	2013	Journal	JSS
S9	A performance comparison of contemporary algorithmic approaches for automated analysis operations on feature models	Pohl *et al.* (2011)	2011	Conference	ASE
S10	A preliminary experimental study on optimal feature selection for product derivation using knapsack approximation	Shi *et al*. (2010)	2010	Conference	PIC
S11	A Preliminary Study on the Effects of Working with a Testing Process in Software Product Line Projects	Machado *et al*. (2012)	2012	Conference	ESELAW
S12	A Regression Testing Approach for Software Product Lines Architectures	Silveira Neto *et al*. (2010)	2010	Conference	SBCARS
S13	A Scalable Approach to Exact Model and Commonality Counting for Extended Feature Models	Fernandez-Amoros *et al*. (2014)	2014	Journal	IEEE Transactions on Software Engineering
S14	A Set of Inspection Techniques on Software Product Line Models	Cunha *et al*. (2012)	2012	Conference	SEKE
S15	A software cost estimation model for a product line engineering approach: supporting tool and UML modeling	Lamine *et al*. (2005)	2005	Conference	SERA
S16	A software product lines system test case tool and its initial evaluation	Neto *et al*. (2012)	2012	Conference	IRI
S17	A Systems Approach to Product Line Requirements Reuse	Niu *et al*. (2014)	2014	Journal	IEEE Systems Journal
S18	A Toolset for Checking SPL Refinements	Ferreira *et al*. (2014)	2014	Journal	JUCS
S19	A use case textual description for context aware SPL based on a controlled experiment	Santos *et al*. (2013)	2013	Journal	CAiSE
S20	Actor in multi product line	Rahmat *et al*. (2016)	2016	Conference	IMCOM
S21	Adoption of software product line to a voice user interface environment	Oliveira *et al.* (2015)	2015	Conference	SEKE
S22	An Algorithm for Generating T-wise Covering Arrays from Large Feature Models	Johansen *et al.* (2012)	2012	Conference	SPLC

S23	An approach for feature modeling of context-aware software product line	Fernandes *et al*. (2011)	2011	Journal	JUCS
S24	An Approach to Analyzing Commonality and Variability of Features using Ontology in a Software Product Line Engineering	Lee *et al*. (2007)	2007	Conference	SERA
S25	An approach to software artefact specification for supporting product line systems	Jirapanthong (2008)	2008	Conference	SERP
S26	An assessment of search-based techniques for reverse engineering feature models	Lopez-Herrejon *et al*. (2015)	2015	Journal	JSS
S27	An evolutionary methodology for optimized feature selection in software product lines	LiZhang (2014)	2014	Conference	SEKE
S28	An experimental study on requirements engineering for software product lines	Neiva *et al*. (2009)	2009	Conference	SEAA
S29	An experimental study to evaluate a SPL architecture regression testing approach	Silveira Neto *et al*. (2012)	2012	Conference	IRI
S30	An Ontology-Based Product Architecture Derivation Approach	Duran-Limon *et al*. (2015)	2015	Journal	IEEE Transactions on Software Engineering
S31	Analyzing the effectiveness of a system testing tool for software product line engineering	Neto *et al*. (2013)	2013	Conference	SEKE
S32	Applying multiobjective evolutionary algorithms to dynamic software product lines for reconfiguring mobile applications	Pascual *et al*. (2015)	2015	Journal	JSS
S33	Architectural evolution of FamiWare using cardinality-based feature models	Gamez and Fuentes (2013)	2013	Journal	IST
S34	ArchSPL-MDD: An ADL-Based Model-Driven Strategy for Automatic Variability Management	Medeiros *et al*. (2015)	2015	Conference	SBCARS
S35	Assessing Software Product Line Testing Via Model-Based Mutation: An Application to Similarity Testing	Henard *et al*. (2013a)	2013	Conference	ICSTW
S36	Assessing the maintainability of software product line feature models using structural metrics	Bagheri and Gasevic (2011)	2011	Journal	Software Quality Journal
S37	Assessment of the Design Modularity and Stability of Multi-Agent System Product Lines	Nunes *et al*. (2009)	2009	Journal	JUCS
S38	Automated diagnosis of feature model configurations	White *et al*. (2010)	2010	Journal	JSS
S39	Automated generation of computationally hard feature models using evolutionary algorithms	Segura *et al*. (2014)	2014	Journal	Expert Systems with Applications
S40	Automated planning for feature model configuration based on functional and non-functional requirements	Soltani *et al*. (2012)	2012	Conference	SPLC
S41	Automatic documentation of [Mined] feature implementations from source code elements and use-case diagrams with the REVPLINE approach	Al-Msie'deen *et al*. (2014)	2014	Journal	International Journal of Software Engineering and Knowledge Engineering
S42	Automatically Checking Feature Model Refactorings	Gheyi *et al*. (2011)	2011	Journal	JUCS
S43	Automatically composing reusable software components for mobile devices	White *et al*. (2008)	2008	Journal	JBCS
S44	Automating Product-Line Variant Selection for Mobile Devices	White *et al*. (2007)	2007	Conference	SPLC
S45	Avoiding redundant testing in application engineering	Stricker *et al*. (2010)	2010	Conference	SPLC
S46	Behavioural Modelling and Verification of Real-time Software Product Lines	Cordy *et al*. (2012)	2012	Conference	SPLC
S47	Beyond Boolean product-line model checking: Dealing with feature attributes and multi-features	Cordy *et al*. (2013)	2013	Conference	ICSE
S48	Bypassing the combinatorial explosion: Using similarity to generate and prioritize t-wise test configurations for software product lines	Henard *et al*. (2014)	2014	Journal	IEEE Transactions on Software Engineering
S49	Capturing product line information from legacy user documentation	John (2006)	2006	Book Chapter	Software Product Lines: Research Issues in Engineering and Management
S50	Combinatorial Interaction Testing with Multi-perspective Feature Models	Patel *et al*. (2013a)	2013	Conference	ICSTW
S51	Combinatorial Test Generation for	Yu *et al*. (2014)	2014	Conference	HASE

	Software Product Lines Using Minimum Invalid Tuples				
S52	Combinatorial Testing for Feature Models Using CitLab	Calvagna *et al*. (2013)	2013	Conference	ICSTW
S53	Combining Multi-Objective Search and Constraint Solving for Configuring Large Software Product Lines	Henard *et al*. (2015)	2015	Conference	ICSE
S54	Comparing Two Black-Box Testing Strategies for Software Product Lines	Accioly *et al*. (2012)	2012	Conference	SBCARS
S55	Comparison of Exact and Approximate Multi-objective Optimization for Software Product Lines	Olaechea *et al*. (2014)	2014	Conference	SPLC
S56	Comprehending feature models expressed in CVL	Reinhartz-Berger *et al*. (2014a)	2014	Conference	MODELS
S57	Comprehensibility of UML-based software product line specifications A controlled experiment	Reinhartz-Berger and Sturm (2014b)	2014	Journal	Empirical Software Engineering
S58	Conditioned model slicing of feature-annotated state machines	Kamischke *et al*. (2012)	2012	Conference	FOSD
S59	Configuration knowledge of software product lines: A comprehensibility study	Cirilo *et al*. (2011)	2011	Conference	VariComp
S60	Consistency maintenance for evolving feature models	Guo *et al*. (2012)	2012	Journal	Expert Systems with Applications
S61	Controlled experiments comparing black-box testing strategies for software product lines	Accioly *et al*. (2014)	2014	Journal	JUCS
S62	Cost-effective test suite minimization in product lines using search techniques	Wang *et al*. (2015)	2015	Journal	JSS
S63	Counterexample Guided Abstraction Refinement of product-line behavioural models	Cordy *et al*. (2014)	2014	Conference	FSE
S64	Decision support for the software product line domain engineering lifecycle	Bagheri *et al*. (2012a)	2012	Journal	Automated Software Engineering
S65	Defining and validating a multimodel approach for product architecture derivation and improvement	Gonzalez-Huerta *et al*. (2013)	2013	Conference	MODELS
S66	Delta-oriented model-based SPL regression testing	Lity *et al*. (2012)	2012	Conference	PLEASE
S67	Designing a Set of Service-Oriented Systems as a Software Product Line	Medeiros *et al*. (2010a)	2010	Conference	SBCARS
S68	Do background colors improve program comprehension in the #ifdef hell?	Feigenspan *et al*. (2013)	2013	Journal	Empirical Software Engineering
S69	Domain implementation in software product lines using OSGi	Almeida *et al*. (2008)	2008	Conference	ICCBSS
S70	Effective requirements elicitation in product line application engineering- An experiment	Adam and Schmid (2013)	2013	Conference	REFSQ
S71	Effects of stability on model composition effort: an exploratory study	Farias *et al*. (2014)	2014	Journal	Software & Systems Modeling
S72	Efficient synthesis of feature models	Andersen *et al*. (2012)	2014	Journal	IST
S73	Empirical assessment of two approaches for specifying software product line use case scenarios	Bonifacio *et al*. (2017)	2015	Journal	Software & Systems Modeling
S74	Empirical validating the cognitive effectiveness of a new feature diagrams visual syntax	Saeed *et al*. (2016)	2016	Journal	IST
S75	Empirical Validation of Complexity and Extensibility Metrics for Software Product Line Architetures	OliveiraJr *et al*. (2010)	2010	Conference	SBCARS
S76	Empirical Validation of Variabilitybased Complexity Metrics for Software Product Line Architecture	OliveiraJr *et al*. (2012)	2012	Conference	SEKE
S77	Empirically Based Evolution of a Variability Management Approach at UML Class Level	Marcolino *et al*. (2014b)	2014	Conference	COMPSAC
S78	Evaluating the representation of user interface elements in feature models: An empirical study	Rodrigues *et al*. (2016)	2016	Conference	SEKE
S79	Evaluating the usability of a visual feature modeling notation	Jaksic *et al*. (2014)	2014	Conference	SLE
S80	Evaluating Variability Instantiation Strategies for Product Lines	John and Silva (2011)	2011	Conference	VaMoS

S81	Evaluating Variability Modeling Techniques for Dynamic Software Product Lines: A Controlled Experiment	Souza *et al*. (2016b)	2016	Conference	SBCARS
S82	Evaluation of a method for proactively managing the evolving scope of a software product line	Villela *et al.* (2010)	2010	Conference	REFSQ
S83	Evidence-based SMarty support for variability identification and representation in component models	Bera *et al.* (2015)	2015	Conference	ICEIS
S84	Evolving feature model configurations in software product lines	White *et al*. (2014)	2014	Journal	JSS
S85	Experimental Evaluation of FMCheck: A Replication Study	Souza *et al*. (2016a)	2016	Conference	SBQS
S86	Experimental studies of e-contract establishment in the PL4BPM context	Goncalves *et al.* (2011)	2011	Journal	IJWET
S87	Experimenting with the comprehension of feature-oriented and UML-based core assets	Reinhartz-Berger and Tsoury (2011)	2011	Journal	Enterprise, Business-Process and Information Systems Modeling
S88	Extending feature models with relative cardinalities	Sousa *et al.* (2016)	2016	Conference	SPLC
S89	Extending the RiPLE-DE process with quality attribute variability realization	Cavalcanti *et al*. (2011)	2011	Conference	QoSA + ISARCS
S90	Facilitating reuse in multi-goal test-suite generation for software product lines	Burdek *et al*. (2015)	2015	Conference	FASE
S91	Faster bug detection for software product lines with incomplete feature models	Souto *et al*. (2015)	2015	Conference	SPLC
S92	Fault-based Product-line Testing: Effective Sample Generation Based on Feature-diagram Mutation	Reuling *et al*. (2015)	2015	Conference	SPLC
S93	Feature interaction testing of variability intensive systems	Patel *et al.* (2013b)	2013	Conference	PLEASE
S94	Feature location in a collection of product variants: Combining information retrieval and hierarchical clustering	Eyal-Salman *et al*. (2014)	2014	Conference	SEKE
S95	Feature maintenance with emergent interfaces	Ribeiro *et al*. (2014)	2014	Conference	ICSE
S96	Feature-context Interfaces: Tailored Programming Interfaces for Software Product Lines	Schroter *et al.* (2014)	2014	Conference	SPLC
S97	Feature-level change impact analysis using formal concept analysis	Eyal-Salman *et al*. (2015)	2015	Journal	International Journal of Software Engineering and Knowledge Engineering
S98	Feature-to-Code Traceability in Legacy Software Variants	Eyal-Salman *et al*. (2013)	2013	Conference	SEAA
S99	Functional testing of feature model analysis tools: a test suite	Segura *et al*. (2011)	2011	Journal	IET Software
S100	Goal-oriented modeling and verification of feature-oriented product lines	Asadi *et al*. (2016a)	2016	Journal	Software & Systems Modeling
S101	Grammar-based Test Generation for Software Product Line Feature Models	Bagheri *et al*. (2012b)	2012	Conference	CASCON
S102	Implementation and Evaluation of an Approach for Extracting Feature Models from Documented UML Use Case Diagrams	Mefteh *et al*. (2015)	2015	Conference	SAC
S103	Improving Product Configuration in Software Product Line Engineering	Tan *et al*. (2013)	2013	Conference	ACSC
S104	Improving software product line configuration: A quality attribute-driven approach	Guana and Correal (2013)	2013	Journal	IST
S105	IncLing: Efficient Product-line Testing Using Incremental Pairwise Sampling	Al-Hajjaji *et al*. (2016)	2016	Conference	GPCE
S106	Incremental model checking of delta-oriented software product lines	Lochau *et al*. (2016)	2016	Journal	Journal of Logical and Algebraic Methods in Programming
S107	Incremental Test Generation for Software Product Lines	Uzuncaova *et al*. (2010)	2010	Journal	IEEE Transactions on Software Engineering
S108	Industrial validation of COVAMOF	Sinnema and Deelstra (2008)	2008	Journal	JSS
S109	Making Software Product Line Evolution Safer	Ferreira *et al.* (2012)	2012	Conference	SBCARS
S110	Measurement analysis and fault proneness indication in product line applications (PLA)	Ahmed (2007)	2007	Conference	SOMET
S111	Measuring the structural complexity of feature models	Pohl *et al*. (2013)	2013	Conference	ASE

S112	Model-Based Design of Product Line Components in the Automotive Domain	Yoshimura *et al*. (2008)	2008	Conference	SPLC
S113	Model-based verification of quantitative non-functional properties for software product lines	Ghezzi and Sharifloo (2013)	2013	Journal	IST
S114	Modeling and Verification for Probabilistic Properties in Software Product Lines	Rodrigues *et al*. (2015)	2015	Conference	HASE
S115	Multi-objective test generation for software product lines	Henard *et al*. (2013b)	2013	Conference	SPLC
S116	Multi-objective Test Prioritization in Software Product Line Testing: An Industrial Case Study	Wang *et al*. (2014)	2014	Conference	SPLC
S117	On Extracting Feature Models from Product Descriptions	Acher *et al*. (2012)	2012	Conference	VaMoS
S118	On the relationship of concern metrics and requirements maintainability	Conejero *et al*. (2012)	2012	Journal	IST
S119	On the value of user preferences in search-based software engineering: A case study in software product lines	Sayyad *et al*. (2013)	2013	Conference	ICSE
S120	Ontology-based feature modeling: An empirical study in changing scenarios	Dermeval *et al*. (2015)	2015	Journal	Expert Systems with Applications
S121	Optimized feature selection towards functional and non-functional requirements in Software Product Lines	Lian and Zhang (2015)	2015	Conference	SANER
S122	Optimizing software product line architectures with OPLA-tool	Federle *et al*. (2015)	2015	Conference	SSBSE
S123	PACOGEN: Automatic Generation of Pairwise Test Configurations from Feature Models	Hervieu *et al*. (2011)	2011	Conference	ISSRE
S124	Potential synergies of theorem proving and model checking for software product lines	Thum *et al*. (2014)	2014	Conference	SPLC
S125	Practical minimization of pairwisecovering test configurations using constraint programming	Hervieu *et al*. (2016)	2017	Journal	IST
S126	Practical pairwise testing for software product lines	Marijan *et al*. (2013)	2013	Conference	SPLC
S127	Preference-based Feature Model Configuration with Multiple Stakeholders	Stein *et al*. (2014)	2014	Conference	SPLC
S128	Preserving architectural styles in the search based design of software product line architectures	Mariani *et al*. (2016)	2016	Journal	JSS
S129	Product Line Variability Modeling Based on Model Difference and Merge	Nie *et al*. (2012)	2012	Conference	COMPSAC
S130	Product-line maintenance with emergent contract interfaces	Thum *et al*. (2016)	2016	Conference	SPLC
S131	Reasoning about edits to feature models	Thum *et al*. (2009)	2009	Conference	ICSE
S132	Reasoning about product-line evolution using complex feature model differences	Burdek *et al*. (2016)	2016	Journal	Automated Software Engineering
S133	Recovering Architectural Variability of a Family of Product Variants	Shatnawi *et al*. (2015)	2015	Conference	ICSR
S134	Refactoring the documentation of software product lines	Romanovsky *et al*. (2011)	2011	Conference	CEE-SET
S135	Relating Requirement and Design Variabilities	Millo and Ramesh (2012)	2012	Conference	APSEC
S136	Requirements Evolution in Software Product Lines: An Empirical Study	Oliveira and Almeida (2015)	2015	Conference	SBCARS
S137	RiPLE-HC: JavaScript systems meets SPL composition	Santos *et al*. (2016)	2016	Conference	SPLC
S138	RiPLE-TE: A process for testing Software Product Lines	Machado *et al*. (2011)	2011	Conference	SEKE
S139	Scalable prediction of non-functional properties in software product lines: Footprint and memory consumption	Siegmund *et al*. (2013)	2013	Journal	IST
S140	Scoping automation in software product lines	Ianzen *et al*. (2015)	2015	Conference	ICEIS
S141	Search Based Design of Layered Product Line Architectures	Mariani *et al*. (2015)	2015	Conference	COMPSAC
S142	Search-based Test Case Selection of Cyber-physical System Product Lines for Simulation-based Validation	Arrieta *et al*. (2016)	2016	Conference	SPLC
S143	Shared Execution for Efficiently Testing Product Lines	Akim *et al*. (2012)	2012	Conference	ISSRE
S144	Similarity-based prioritization in software product-line testing	Al-Hajjaji *et al*. (2014)	2014	Conference	SPLC
S145	SIP: Optimal product selection from feature	Hierons *et al*. (2016)	2016	Journal	TOSEM

	models using manyobjective evolutionary optimization				
S146	SOPLE-DE: An approach to design service-oriented product line architectures	Medeiros *et al*. (2010b)	2010	Conference	SPLC
S147	SPLat: Lightweight dynamic analysis for reducing combinatorics in testing configurable systems	Kim *et al*. (2013)	2013	Conference	ESEC/FSE
S148	SPLLIFT- Statically analyzing software product lines in minutes instead of years	Bodden *et al*. (2014)	2014	Conference	PLDI
S149	Strategies for product-line verification: Case studies and experiments	Apel *et al*. (2013b)	2013	Conference	ICSE
S150	Supporting commonality-based analysis of software product lines	Heradio-Gil *et al*. (2011)	2011	Journal	IET Software
S151	Supporting distributed product configuration by integrating heterogeneous variability modeling approaches	Galindo *et al*. (2015)	2015	Journal	IST
S152	Supporting Online Updates of Software Product Lines: A Controlled Experiment	Michalik *et al*. (2011b)	2011	Conference	ESEM
S153	Supporting program comprehension in large preprocessor-based software product lines	Feigenspan *et al*. (2012)	2012	Journal	IET Software
S154	Symbolic Model Checking of Product-Line Requirements Using SAT-Based Methods	Ben-David *et al*. (2015)	2015	Conference	ICSE
S155	SyMPLES-CVL: A SysML and CVL Based Approach for Product-Line Development of Embedded Systems	Chiquitto *et al*. (2015)	2015	Conference	SBCARS
S156	Synthesis of Attributed Feature Models from Product Descriptions	Becan *et al*. (2015)	2015	Conference	SPLC
S157	Test control algorithms for the validation of cyber-physical systems product lines	Arrieta *et al*. (2015)	2015	Conference	SPLC
S158	Test order for class-based integration testing of Java applications	Hashim *et al*. (2005)	2005	Conference	QSIC
S159	Testing and inspecting reusable product line components: First empirical results	Denger and Kolb (2006)	2006	Conference	ISESE
S160	Testing Software Product Lines Using Incremental Test Generation	Uzuncaova *et al*. (2008)	2008	Conference	ISSRE
S161	The effects of visualization and interaction techniques on feature model configuration	Asadi *et al*. (2016b)	2016	Journal	Empirical Software Engineering
S162	TIRT: A Traceability Information Retrieval Tool for Software Product Lines Projects	Santos *et al*. (2012)	2012	Conference	EUROMICRO SEAA
S163	Toward automated feature model configuration with optimizing nonfunctional requirements	Asadi *et al*. (2014)	2014	Journal	IST
S164	Toward recovering component-based software product line architecture from object-oriented product variants	Eyal-Salman and Seriai (2016)	2016	Conference	SEKE
S165	Towards the effectiveness of a variability management approach at use case level	Marcolino *et al*. (2013b)	2013	Conference	SEKE
S166	Towards Validating Complexity-based Metrics for Software Product Line Architectures	Marcolino *et al*. (2013a)	2013	Conference	SBCARS
S167	Using a rule-based method for detecting anomalies in software product line	Elfaki *et al*. (2014)	2014	Journal	Research Journal of Applied Sciences, Engineering and Technology
S168	Using background colors to support program comprehension in software product lines	Feigenspan *et al*. (2011)	2011	Conference	EASE
S169	Using Feature Diagrams with Context Variability to Model Multiple Product Lines for Software Supply Chains	Hartmann and Trew (2008)	2008	Conference	SPLC
S170	Variability Identification and Representation in Software Product Line UML Sequence Diagrams: Proposal and Empirical Study	Marcolino *et al*. (2014a)	2014	Conference	SBES
S171	Verification of Software Product Line artefacts: A checklist to support feature model inspections	Mello *et al*. (2014)	2014	Journal	JUCS
S172	Visualization and exploration of optimal variants in product line engineering	Murashkin *et al*. (2013b)	2013	Conference	SPLC
S173	Where has all my memory gone? Determining memory characteristics of product variants using virtualmachine-level monitoring	Lengauer *et al*. (2014)	2014	Conference	VaMoS
S174	XTraQue: Traceability for product line systems	Jirapanthong and Zisman (2009)	2009	Conference Modeling	Software and Systems