Journal of Computer Sciences

Original Research Paper

Comparing SMarty and PLUS for Variability Identification and Representation at Product-Line UML Class Level: A Controlled Quasi-Experiment

Anderson S. Marcolino and Edson OliveiraJr

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

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

Abstract: Although variability management is one of the main activities of software product lines, current literature provides almost no empirical evaluations on variability management approaches based on UML. This paper aims at experimentally comparing two approaches and picks SMarty and PLUS as representative examples. Such comparison takes into account their effectiveness of expressing correctly and incorrectly variabilities in UML class diagrams. We used a 2×2 factorial design for this study. We calculated and analyzed data from participants using the T-Test. The Spearman technique supported correlation of the effectiveness of the approaches and the participants prior variability knowledge. In general, PLUS was more effective than SMarty. Generalization of results is not possible as this is an incipient evidence of PLUS and SMarty effectiveness based on graduate students and lecturers. However, counting on students and lecturers provides several contributions as we discuss in this paper.

Keywords: Controlled Experiments, Effectiveness, Experimental Evaluation, SMarty, Software Product Line, UML, Variability Management

Introduction

Software Product Line (SPL) has being consolidated in the last few years. Its core objective is the generation of specific products for a given domain. Such an approach comprises a set of essential activities, such as the management of variability, which is an essential issue for the success of SPLs (Chen *et al*., 2009). Variability Management (VM) is essential to derive consistent members of an SPL based on similarities and variabilities accurately identified and represented by a given approach (Capilla *et al*., 2013). According to (Bosch *et al*., 2015) "a significant amount of variability research and practice deals with the representational aspects of variability in space and time". VM basically encompasses identification of variability in SPL artifacts and product configuration analysis (Thurimella and Bruegge, 2012). Identification is responsible for mining similar and variable assets from a certain domain, usually based on specific domain needs and requirements specifications. Representation concerns on explicitly modeling where variability takes place in certain SPL assets. Product configuration allows one to analyze potential products of an SPL based on the modeled variabilities and their combinations of variation points and variants.

A considerable amount of variability management approaches are available in literature, according to (Capilla *et al*., 2013; Chen *et al*., 2009; Galster *et al*., 2014; Thurimella and Bruegge, 2012). Among them, several approaches are UML-based. Most of this kind of approach was not evaluated using classic and rigorous scientific methods (Chen and Babar, 2011). Ahnassay *et al*. (2014), in a systematic literature review, reveals that a large majority of empirical evaluations in SPL had not been sufficiently designed or reported. The need for improvement in quality research design and quality reporting is reinforced further with the findings of such a literature review. Bosch *et al*. (2015) claim that the diversity of notations and tools points to industry has not yet solved the variability management problem and continues to experiment with solutions and approaches. Thus, we cannot rely on the available body of knowledge. UML-based approaches, which use stereotypes and meta-attributes for describing variability, are especially considered in this paper as they represent an important portion of the existing literature on VM. Chen and Babar (2011) discusses in a systematic literature review on evaluation of VM approaches that from 97 reviewed primary studies, 25 (25.7%) are UML-based solution proposals.

Science Publications

In their survey on variability modeling in industrial practice, Berger *et al.* (2013) indicates 28.6% of surveyed companies adopt UML as a notation for variability representation, as the second most frequent notation. Feature modeling is the first one (77.1%). In addition, UML is a widely-adopted standard notation for specifying SPL analysis and design artifacts (Chen and Babar, 2011). Cruz-Lemus *et al.* (2011) performed a set of experiments providing evidence that stereotypes and meta-attributes improve UML diagrams comprehension. Therefore, this paper concentrates on UML-based VM approaches that use stereotypes to represent variability.

The PLUS method (Gomaa, 2004), Product Line UML-based Software Engineering, is widely referenced, and an important example of UML-based VM approach that use stereotypes (Thurimella and Bruegge, 2012; Capilla *et al.*, 2013; Chen *et al.*, 2009; Galster *et al.*, 2014; Chen and Babar, 2011). Another promising approach is Stereotype based Management of Variability (SMarty) (OliveiraJr *et al.*, 2010; 2013). SMarty manages variabilities in UML diagrams based on a profile and respective guide lines to apply the profile stereotypes to use case diagrams, class diagrams, component diagrams, activity diagrams and sequence diagrams.

UML-based approaches are promising considering the VM research field (Chen and Babar, 2011). However, they provide almost no empirical evidence on the identification and representation of variabilities in UML diagrams (Thurimella and Bruegge, 2012; Capilla *et al.*, 2013; Chen *et al.*, 2009; Galster *et al.*, 2014; Reinhartz-Berger and Sturm, 2014). Less than 10% of the papers studied by (Chen and Babar, 2011) are based on empirical evidence, but are rather based on a theoretical analysis. Such evidence is essential to promote variability management to specific stakeholders as for the increased industry adoption and is a way of acquiring knowledge by means of experiments. Therefore, this paper is based on an experimental study regarding to the effectiveness of the SMarty approach with relation to the PLUS method. SMarty is aimed at identifying and representing variability in UML use case diagrams, class diagrams and sequence diagrams. Note that in this paper we are focused on the SPL solution space, rather than problem space (Sanen *et al.*, 2009; Schaefer *et al.*, 2011). In addition, we took into consideration annotative approaches, rather than compositional ones.

Effectiveness is used in several works (Basili and Selby, 1987; Abdelnabi *et al.*, 2004; Coteli, 2013; Martinez-Ruiz *et al.*, 2011) as a measure of whether tasks carried out by one achieved specific goals with accuracy and completeness, taking into account: The number of tasks and the number of errors made by a user (ISO/IEC, 2004). For instance, (Basili and Selby, 1987) use effectiveness for fault detection at comparing testing strategies, in which participants are asked to detect the presence of faults by applying a given strategy. Reinhartz-Berger and Sturm (2014) identify the amount of correct and incorrect answers from activities of variability modeling for the ADO Mmethod for domain modeling using UML. Abdelnabi *et al.* (2014) adopt effectiveness for verifying the number of real defects with code reading techniques to object-oriented frameworks.

In this paper, we consider effectiveness as a measure of the number of correctly and incorrectly identified and represented (modeled) variabilities in class diagrams for a given variability management technique (SMarty and PLUS).We have already successfully applied the effectiveness measure in (Marcolino *et al.*, 2013; 2014).

Therefore, this paper is concerned with answering the following research question: “Is SMarty effective at identifying and representing variability in UML class diagrams?” In order to answer this question, an empirical studys was carried out as a controlled quasi-experiment and is presented in detail in this paper.

The results obtained with the empirical study provided an initial evidence that PLUS is more effective than SMarty at identifying and representing variabilities in class diagrams. Although SMarty did not provide evidence of its major effectiveness compared to PLUS for class diagrams, we performed important improvements.

The main contributions of this paper rely on: (i) an empirical comparative analysis of the effectiveness of UML stereotype-based VM approaches; (ii) the empirical evaluation of the SMarty approach; and (iii) improvement of the identification and representation of variability in UML-based VM modeling by enhancing SMarty.

The remainder of this paper is organized as follows: Section 2 discusses important concepts with regard to variability management, PLUS, and SMarty; Section 3 reports planning, execution and analysis, and interpretation of the empirical study taking into account the Jedlitschka *et al.* (2007) template; Section 4 discusses related work; and Section 5 presents conclusion and directions for future work.

Background

Software Product Line and Variability Management

The SPL approach aims at promoting the generation of specific products for a given application domain based on the reuse of well-defined artifacts and resources, the core assets (Capilla *et al.*, 2013).

Core assets are the main parts of an SPL; they contain the architecture of the SPL, and components represented in a way that makes the common and variable aspects of the potential products of the SPL clear. The ability and simplicity of producing products from an SPL depends on how well-designed is its core assets. The more generic the artifacts of the core assets are, the more specific products can be generated. This kind of design decision is treated as variabilities (Capilla *et al*., 2013; Chen *et al*., 2009; Galster *et al*., 2014; Thurimella and Bruegge, 2012).

Variability is the general term used to refer to the ability to derive different products (Capilla *et al*., 2013; Bosch *et al*., 2015). Due to its importance to SPL development and evolution, one of the essential SPL activities focuses on the variability management (Capilla *et al*., 2013; Bosch *et al*., 2015).

The relevance of the activity of variability management for SPLs deserves attention from many researches, as it can be observed in certain studies in the literature, such as, (Bosch *et al*., 2015; Capilla *et al*., 2013; Chen *et al*., 2009; Galster *et al*., 2014; Gomaa, 2004; Thurimella and Bruegge, 2012).

Although the number of variability management approaches is steadily growing, several existing approaches do not explicitly identify and represent variabilities in different kinds of artifacts; requirements specification and feature diagrams, for instance (Chen *et al*., 2009), especially UML-based ones. This kind of approach mostly takes stereotypes and properties representing SPL variabilities into account. Thus, the industry requires evidence on the effectiveness of these approaches to make them adoptable (Chen *et al*., 2009; Capilla *et al*., 2013; Galster *et al*., 2014; Chen and Babar, 2011).

In order to provide a more effective UML-based approach for variability management, the SMarty approach has been developed (OliveiraJr *et al*., 2010; 2013). It is supported by a profile and a set of guidelines to apply stereotypes and relationships. However, SMarty needs to gather evidence with regard to its effectiveness at identifying and representing variability by means of a set of empirical studies. Therefore, Gomaa's widely known PLUS method (Gomaa, 2004) was chosen, based on a secondary literature study to perform a set of empirical studies.

The selection of such approaches is supported by the following comparison criteria: definition of a UML 2.0 profile; guidelines for identifying and representing variability; use of UML stereotypes; explicit representation of: variation points, mandatory, optional, inclusive and exclusive variants, and constraints among variants; cardinality of: variabilities, variation points and variants; representation biding time, addition of new variants to variation points; and variability tracing. Table 1 shows such criteria for PLUS and SMarty.

The following sections present the essential concepts of the PLUS method and the SMarty approach.

The PLUS Method

The Product Line UML-based Software Engineering (PLUS) method, created by (Gomaa, 2004), proposes SPL activities for requirements engineering, analysis and design. The requirements activity provides SPL scope definition, use case modeling and feature modeling. The analysis activity is composed of: static modeling, object construction, dynamic modeling, finite state machine and feature/class dependency modeling.

The PLUS use case modeling and class modeling activities aim to explicitly model similarities and variabilities based on UML stereotypes. PLUS provides a set of concepts and techniques to extend UML-based design methods and processes for single systems to handle SPLs.

PLUS does not provide a definition of an UML profile, thus there are no explicit meta-attributes and meta-classes for the variability modeling activity. PLUS uses stereotypes to provide identification of variation points and variants, in which several of them are specific to certain UML diagrams.

Table 2 summarizes the stereotypes of PLUS to represent variabilities in use case diagrams and class diagrams.

Table 1: PLUS and SMarty summarized comparison

		Class Diagrams	
Criteria		PLUS	SMarty
Defines UML Profile 2.0			●
Provides Guidelines for identification an Representation of Variability			●
Uses UML Stereotypes		●	●
Variation Point			●
Variant Type			
	Mandatory	●	●
	Optional		
	Inclusive (OR)		●
	Exclusive (XOR)		●
Variability			●
	Complement		●
	Mutual Exclusion		●
Represents Variability/Variation Point or Variation Point/Variant Cardinality			●
Represents Binding Time			●
Allows Addition of New Variants to Variation Points			●
Allows Tracing Variabilities			●

Table 2: Overview of the PLUS Method Stereotypes (Gomaa, 2004)

		Is the stereotype applied to	
Stereotype	Description	Use Case?	Class?
<<kernel>>	Used to represent mandatory elements.	Yes	No
<<optional>>	Used to represent optional elements. It represents an element that can be selected or not in a specific product.	Yes	Yes
<<alternative>>	Used to represent alternative elements, mutually exclusive elements.	Yes	No

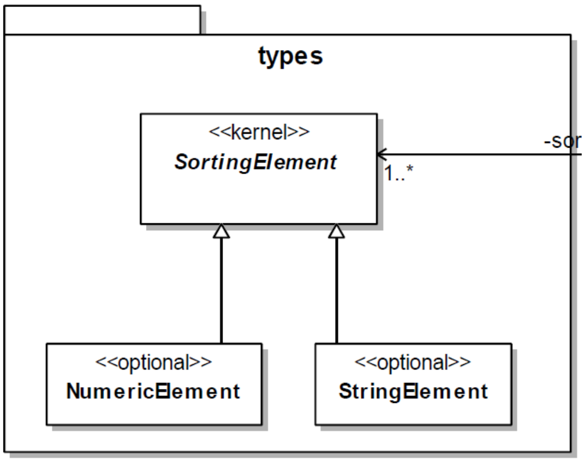


Fig. 1: Excerpt of an SPL class diagram representation of variability according to PLUS

The stereotypes presented in Table 2 allow the user to identify variabilities in UML use case diagrams and class diagrams. Figure 1 presents an example of the use of PLUS in use case diagrams. The use case Check Customer Account, for example, is tagged as << alternative>>. Such tagging does not make it explicit what kind of alternative constraint (inclusive or exclusive) represents. The e-Commerce SPL (Fig. 1) admits the generation of only two products. This use case is related to the e-Commerce product for B2C transactions; thus, it has constraints to be selected with other use cases, such as Confirm Delivery and Prepare Purchase Order.

In addition, in Fig. 1 two of the customer-initiated use cases (Browse Catalog and Make Purchase Request) are common to all electronic commerce systems and this becomes kernel use cases of the SPL. Similarly, two of the supplier-initiated use cases (Process Delivery Order and Confirm Shipment) are common to all electronic commerce systems. Thus, they become kernel use cases. On the other hand, two of the customer use cases (Create Requisition and Confirm Delivery) are initiated only in Business to Busines (B2B) systems, and a third use case (Check Customer Account) initiates only in business to customer B2C systems. On the supplier side, one use case (Send Invoice) is initiated only in B2B systems, and another use case (Bill Customer) is initiated only in B2C systems. Two purchase order use cases (Prepare Purchase Order and Deliver Purchase Order) are optional and could be initiated in either B2B or B2C systems.

Use cases initiated in either the B2B or the B2C system are alternative use cases. The purchase order use cases (Prepare Purchase Order and Deliver Purchase Order) are categorized as optional use cases.

The SMarty Approach

SMarty is an approach for Stereotype-based Management of Variability in SPL. It is composed of a UML 2 profile, the *SMartyProfile* and a process, the *SMartyProcess*. SMarty aims to allow variabilities of an SPL to be managed in a systematic way based on UML models (OliveiraJr *et al*., 2010; 2013). Currently, SMarty is in its 5.1 version, supporting use case, class, sequence, component and activity diagrams. This version has changed according to new extensions and improvements throughout empirical studies, as we can observe during the empirical study, presented in this paper.

SMarty fully complies with UML meta-models, avoiding ambiguity and is supported by a set of guidelines specific for each UML diagram for variability representation.

The SMartyProfile

The *SMartyProfile* contains a set of stereotypes and properties to represent variability in SPL models. Basically, *SMartyProfile* uses a standard object-oriented notation and its profiling mechanism (OMG, 2011) to provide an extension of UML as well as to allow graphical representation of variability concepts, as observed in Fig. 2. Thus, there is no need to change the system design structure to comply with the SPL approach.

The *SMartyProfile* comprises the stereotypes presented in Table 3. One of the major rationale for creating SMarty is to broadly represent the information needed for the user with no ambiguity, making such a representation process easier.

Stereotypes from Table 3 also have a set of properties to allow setting values for different variability abstraction levels. Two examples of such properties are: the identification of when a given variability must be resolved (the *bindingTime*), and the associated variants of a given variation point by means of the *variants* property.

Table 3: Overview of the *SMartyProfile* Stereotypes (OliveiraJr *et al*., 2010)

Stereotype	Description
<<variability>>	The concept of SPL variability and is an extension of the UML meta class Comment.
<<variant>>	The concept of SPL variant and is an abstract extension of the UML meta classes *Actor, Use Case, Interface,* and *Class*. This stereotype is specialized in four other non-abstract stereotypes which are: <mandatory>>, <<optional>>, <<alternative _OR>> and <<alternative_ XOR>>.
<<mandatory>>	A compulsory variant that is part of every SPL product.
<<optional>>	A variant that might be selected to resolve a variation point or a variability
<<alternative_OR>>	A variant that is part of a group of alternative inclusive variants. Different. combinations of this kind of variants may resolve variation points or variabilities in different ways.
<<alternative_XOR>>	A variant that is part of a group of alternative exclusive variants. This means that only one variant of the group can be selected to resolve a variation point or variability.
<<mutex>>	The concept of SPL variant constraint and is a mutually exclusive relationship between two variants. This means that when a variant is selected another variant must not be selected.
<<requires>>	The concept of SPL variant and is a relationship between two variants in which the selected variant requires the presence of another specific variant.
<<variable>>	An extension of the UML metaclass Component. It indicates that a component has a set of classes with explicit variabilities. This stereotype has the tagged value *classSet* which is the collection of classinstances that form a component.

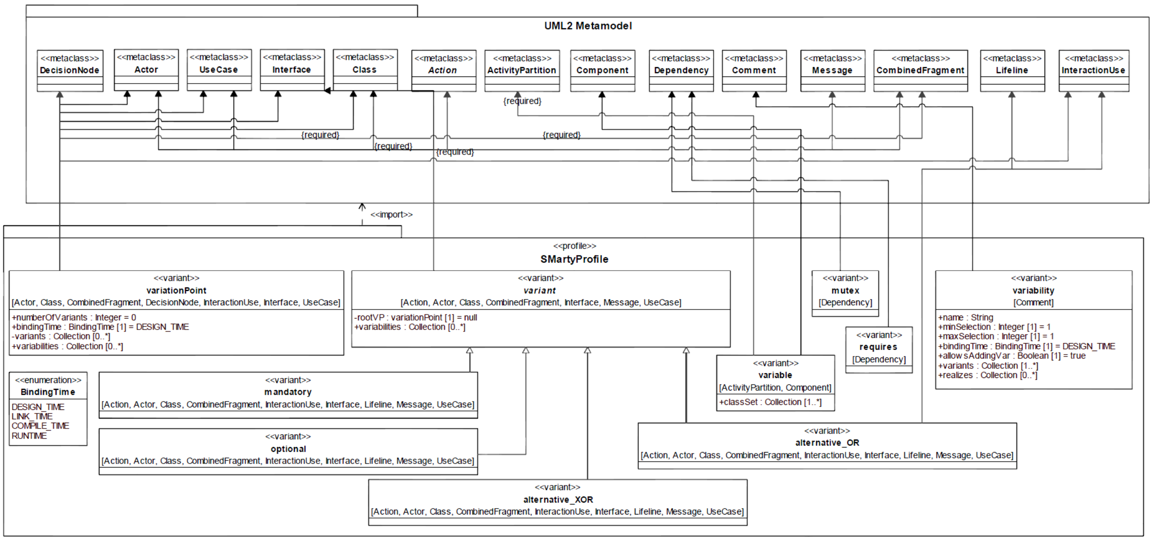

UML2 Metamodel
<<metaclass>> DecisionNode
<<metaclass>> Actor
<<metaclass>> UseCase
<<metaclass>> Interface
<<metaclass>> Class
<<metaclass>> Action
<<metaclass>> ActivityPartition
<<metaclass>> Component
<<metaclass>> Dependency
<<metaclass>> Comment
<<metaclass>> Message
<<metaclass>> CombinedFragment
<<metaclass>> Lifeline
<<metaclass>> InteractionUse
{required}
{required}
{required}
{required}
{required}
<<import>>
<<profile>>
SMartyProfile
<<variant>> variationPoint
[Actor, Class, CombinedFragment, DecisionNode, InteractionUse, Interface, UseCase]
+numberOfVariants : Integer = 0
+bindingTime : BindingTime [1] = DESIGN_TIME
-variants : Collection [0..*]
+variabilities : Collection [0..*]
<<variant>> variant
[Action, Actor, Class, CombinedFragment, Interface, Message, UseCase]
-rootVP : variationPoint [1] = null
+variabilities : Collection [0..*]
<<variant>> mutex
[Dependency]
<<variant>> requires
[Dependency]
<<variant>> variability
[Comment]
+name : String
+minSelection : Integer [1] = 1
+maxSelection : Integer [1] = 1
+bindingTime : BindingTime [1] = DESIGN_TIME
+allowsAddingVar : Boolean [1] = true
+variants : Collection [1..*]
+realizes : Collection [0..*]
<<enumeration>> BindingTime
DESIGN_TIME
LINK_TIME
COMPILE_TIME
RUNTIME
<<variant>> mandatory
[Action, Actor, Class, CombinedFragment, InteractionUse, Interface, Lifeline, Message, UseCase]
<<variant>> optional
[Action, Actor, Class, CombinedFragment, InteractionUse, Interface, Lifeline, Message, UseCase]
<<variant>> variable
[ActivityPartition, Component]
+classSet : Collection [1..*]
<<variant>> alternative_OR
[Action, Actor, Class, CombinedFragment, InteractionUse, Interface, Lifeline, Message, UseCase]
<<variant>> alternative_XOR
[Action, Actor, Class, CombinedFragment, InteractionUse, Interface, Lifeline, Message, UseCase]

Fig. 2: *SMartyProfile* 5.1

The SMartyProcess

The *SMartyProcess* is a systematic process that guides the user through the identification and representation of variabilities in SPL models. It is supported by a set of application guidelines as well as by the *SMartyProfile* and its set of stereotypes (Table 3) to represent variabilities.

The guidelines defined for the *SMartyProcess* are:

General Guidelines for Identification and Representation of Variability (RV):

- **RV.1** Variabilities with optional variants has multiplicity *minSelection* = 0 and *maxSelection* = 1
- **RV.2** Variabilities with exclusive variants (alternative XOR) has *multiplicity min-Selection* = *maxSelection* = 1
- **RV.3** Variabilities with inclusive variants (alternative OR) has multiplicity *minSelection* = 1 and *maxSelection* = size (variants) where size(x) is a function which brings the quantity of elements from the collection x
- **RV. 4** The value *bindingTime* must be defined by selecting one of the values from the *BindingTime* enumeration class, which are: DESIGN TIME, LINK TIME, COMPILE TIME, RUNTIME
- **RV. 5** The Boolean value from the property *allowsAddingVar* must be analyzed according to the possibility of maintaining the variation point opened (*true*, accepting new variants) or closed (*false*, accepting no new variants)
- **RV. 6** The values of the variants collection is formed by the instances of variants associated with the variation point or with the variability

Guidelines for Class Diagrams (CL)

- **CL.1** in class diagrams, variation points and their variants are identified in the following relationships: (a) generalization, the most general classes represent the variation points and the most specific ones are the variants; (b) interface realization, the suppliers (specifications) represent variation points and the implementations (clients) represent the variants; (c) aggregation association, the typed instances with hollow diamonds represent the variation points and the associated typed instances represent the variants; and (d) composite aggregation, the typed instances with filled diamonds represent the variation points and the associated typed instances represent the variants
- **CL.2** elements of class diagrams related to the association relationship in which the aggregation Kind attribute has no value, i.e., neither an aggregation nor a composition suggest either mandatory or optional variants
- **CL.3** variants in classes diagrams that require the existence of other variants must tag their relationships as <<requires>>
- **CL.4** mutually exclusive variants in classes diagrams for a certain product must tag their relationships as <<mutex>>

Figure 3 illustrates an example of the application of the *SMartyProfile* and the guidelines defined in the SMartyProcess for a use case diagram.

For the excerpt from the Arcade Game Maker (AGM), a pedagogical SPL created by SEI, presented in Fig. 3, the use case Check Previous Best Score is tagged as _optional_ and is related to a UML comment that defines the set of properties, to characterize the variability. The guideline RV.1, which indicates that variabilities with optional variants have multiplicity *minSelection* = 0 and *maxSelection* = 1, was used to complete the multiplicity information. Likewise, RV.4 was used to set the *bindingTime* as DESIGN TIME. As the property *allowsAddingVar* shows, it is possible to add new variants in a future extension of the SPL model, as suggested by RV.5. The property *variants* was specified according to RV.6, which indicated that the values of the variants collection is formed by the instances of variants associated to the variation point or to the variability, in this example, to the Check Previous Best Score use case.

Figure 3 depicts another variability in which an UML comment was provided, as with the previous example. The "play game" variability was filled according to its characteristics, but, unlike the "check score" variability, the variants were inclusive; RV.3 was used to fill the multiplicity. The *variants* related to the variability are composed of Play Brickels, Play Pong and Play Bowling, all of them related to the use case Play Selected Game, which indicates, by UC.1, that they were tagged as alternative inclusive.

Empirical Comparison Between PLUS and SMarty

This section reports the experiment carried out for class diagrams, following the guidelines proposed by Jedlitschka *et al*. (2007).

Research Objective

For our experiment carried out, the following research objective has been formulated: Compare the PLUS method and the *SMarty* approach, to identify the most effective one, with respect to the capability of identifying and representing variabilities in Software Product Line class diagrams, from the point of view of graduate students and lecturers playing the role of software product line architects, in the context of master and Ph.D. students of the Software Engineering area from the Pontifical Catholic University of Rio Grande do Sul (PUCRS).

Experimental Planning

This section describes the protocol used to perform our experiment and analyze the obtained results. We also present necessary information to allow replication of such experiment.

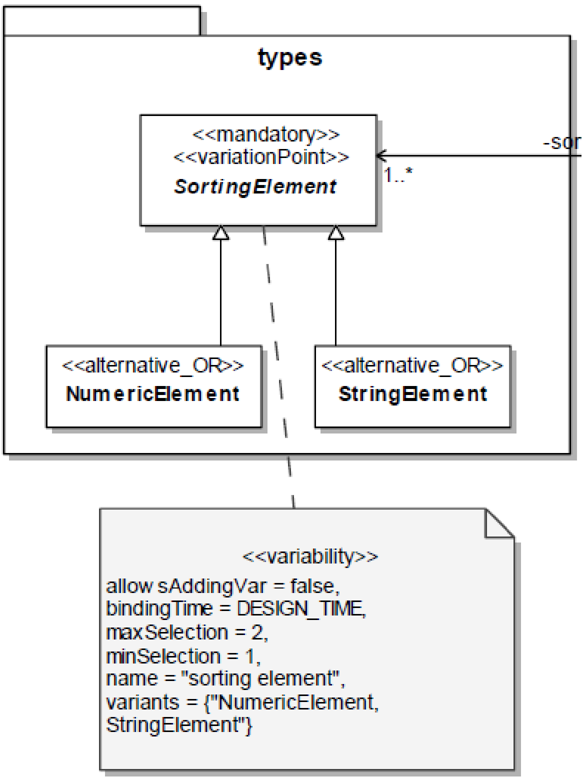


Fig. 3: Excerpt of an SPL Class Diagram Variability Representation According to SMarty

Goals

The main goals of our experiment can be stated as two research questions (R.Q.), as follows:

- **R.Q.1** What method/approach is more effective at identifying and representing variabilities in UML-based SPL class diagrams?
- **R.Q.2** Did the prior participant variability knowledge influence the obtained effectiveness of the method/approach?

Participants

The selection of participants was made based on convenience, due to the lack of participants with advanced knowledge on UML and moderate knowledge on variability. Thus, we selected graduate students and lecturers from the software engineering area of different universities, closely related to research projects using UML for single software development and SPL as a reusing technique, not essentially focused on variability based on UML. Although we set the study for the software product line architects' point of view, we recruited these graduate and lecturers due to the lack of practitioners attending our study. Therefore, this experiment is considered a quasi-experiment due to not perform randomization at particopants sampling.

Our experiment counted on 10 graduate students.

The sampling strategy adopted for the experiment was balancing the groups based on the similar characteristics and similar numbers, such as educational background and with regard to previous variability modeling knowledge. Then, once groups were balanced, we randomized the assignment of participants and objects to treatments as suggested by Wilkinson (1999) and discussed by Heinsman and Shadish (1996).

Participants neither were paid nor received educational credits for attending the experiment. The main motivation for participants taking part of the experiment was learning new concepts with regard to variability management approaches related to UML in order to allow practicing and evolving reuse concepts already acquired during undergraduate and graduate courses.

We obtained the participant consent by a consent form according to the general ethics terms in experiments involving human beings. Such a consent form also included several clauses with respect to the confidentiality of the study data.

Objects

The experiment was performed based on the following objects. Participants received two class diagrams. One class diagram is from the Arcade Game Maker (AGM) SPL with 13 classes, formed by 8 mandatory classes, 2 variation point classes and 3 inclusive variant classes. The other class diagram is from the E-Commerce SPL with 16 classes, in which 10 are optional and 6 are kernel.

Instrumentation

With relation to measurement instruments, we applied a characterization questionnaire with regard to the educational level, working environment (academic or industrial) and years of working experience. In addition, we enquired the prior knowledge on UML, SPL and variability, based on a Likert scale in which answers varied from "no experience", "read about", "basic use", "moderate use", and "advanced use". We used ordinary spreadsheets to collect and organize such data.

Participants were split into two groups. One group focused on the *X* approach (the PLUS method) and one group focused on the *Y* approach (the SMarty approach). We trained one group to identify and represent variabilities according to the *X* approach. We trained the other group to identify and represent variabilities according to the Y approach. Participants were trained taking sessions for SPL and variability concepts into consideration, for which we performed and graded essays and exercises *in loco*.

On the identification and representation of variability in class diagrams, the participants received guidelines, which provided directions on how to identify and represent variabilities according to the respective variability management approaches they were assigned to. Such guidelines were elaborated based on the available information from the main publications on variability management approaches.

The same instruments were available for each participant, in printed format, to be consulted during the study execution. Participants assigned to either *X* or *Y* approach received materials from specific approach.

Tasks

Participants were required to identify and represent variabilities in class diagrams of the E-Commerce (Gomaa, 2004) and AGMSPLs by means of the application of stereotypes and definition of properties. Therefore, we evaluated the variabilities whether they were correctly or incorrectly modeled in a given SPL. Note that for the study carried out in this paper, no UML modeling tools were used. See Section 3.2.8 for more details.

Correct and incorrect variability modeling was based on the following criteria: no points when there is no such a way for deriving a product according to the semantic rules for each approach; and one point when partial or full products can be derived according to the semantic rules for each approach.

Hypotheses, Parameters and Variables

The following hypotheses are stated for R.Q.1 and tested in the experiment (see Table 4 for abbreviations):

- **Null Hypothesis ($H_{0.RQ1}$)**: the approaches are equally effective at identifying and representing variabilities in UML-based SPL class diagrams:

  $H_{0.RQ1}$: μ(*Effectiv*(*X*)) = μ (*Effectiv*(*Y*))

- **Alternative Hypothesis ($H_{1.RQ1}$)**: *X* approach is less effective than *Y* approach:

  $H_{1.RQ1}$: μ(*Effectiv*(*X*)) < μ (*Effectiv*(*Y*))

- **Alternative Hypothesis ($H_{2.RQ1}$)**: *X* approach is more effective than *Y* approach:

  $H_{2.RQ1}$: μ(*Effectiv*(*X*)) > μ(*Effectiv*(*Y*))

Note that "*Y* approach" always refers to the treatment SMarty approach, whereas "*X* approach" refers to the control method/approach to which SMarty is compared to, in this case PLUS.

We stated the following hypotheses for R.Q.2 and tested them in the experiment (see Table 4 for abbreviations):

- **Null Hypothesis ($H_{0.RQ2}$):** the obtained effectiveness of the approaches are not affected by the participants' prior variability knowledge:
  $H_{0.RQ2}$: *Corr*(*VarKnowledge and VMApp X*) = *Corr*(*VarKnowledge and VMApp_Y*) = 0

- **Alternative Hypothesis ($H_{1.RQ2}$):** *X* approach effectiveness is less affected by the participants prior variability knowledge than the *Y* approach:

  $H_{1.RQ2}$: *Corr*(*VarKnowledge and VMApp X*) < *Corr*(*VarKnowledge and VMApp_Y*)

- **Alternative Hypothesis ($H_{2.RQ2}$)**: *Y* approach effectiveness is less affected by the participants prior variability knowledge than the *X* approach:

  $H_{2.RQ2}$: *Corr*(*VarKnowledge and VMApp_X*) > *Corr*(*VarKnowledge and VMApp_Y*)

Table 4 describes dependent and independent variables of the experiment.

In order to avoid any biases, we used diagrams from SPLs provided by the authors of PLUS, as well as the SEI AGM SPL.

Experimental Design

We used a 2×2 factorial design for our study.

We performed a pilot project to evaluate the instrumentation taking a small sample of graduate students and lecturers of software engineering into account. Thus, we made adjustments on such an instrumentation based on the pilot project results, which also evaluated the average required training and experimental session time. We discarded all the collected pilot data from the individual experimental study data analysis, as well as the participants, who did not attend the official experimental sessions.

The pilot project led us to use only one condition (VMApp, Table 4) for each participant in order to avoid the bias of participants learning from the use of a previous VMApp. Therefore, our experiment was established as between-participants design (Jedlitschka *et al.*, 2007).

Procedure

Participants received a 60-minute training session with regard to SPL and variability. In addition, a training session was given on identification and representation of variability inclass diagrams. We trained participants with either the PLUS method or the SMarty approach. Note that participants do not know the real names of the *X* and *Y* approaches to avoid any biases.

Table 4: Dependent and independent variables description

Name of the 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
Variability Management Approach/method	independent	VMApp	method	UML-based variability management SPL	N.A.	nominal	N.A.	SMarty PLUS	N.A
SPL class diagrams		SPL	product	UML diagrams	N.A.	nominal	N.A.	class	N.A
Effectiveness	dependent	Effectiv	process	correct/incorrect variabilities	internal: effectiveness; external: quality	ordinal	Number of correct minus number incorrect variabilities	any integer counts 1, incorrect counts 0	Correct variability
Correlation between Subjects Variability formula Knowledge and Effectivenes		Corr VarKnowledge and VMApp)	process	subjects variability knowledge and VMApp effectiveness	external	nominal	Spearman ranking correlation scale	[-1.0, +1.0]	Spearman (□)

The experiment took place in the academic environment of the Pontifical Catholic University of Rio Grande do Sul (PURS).

Standard procedures adopted for each participant were as follows:

1. the participant attends the place of the conducted study
2. the study coordinator gives the participant a set of documents
    - the empirical study consent form
    - the characterization questionnaire
    - essential concepts on variability management in SPL; and
    - the description of the E-Commerce SPL
3. the participant reads each given document
4. the study coordinator explains the given documents
5. the study coordinator randomly associates each participant to the X or Y approach
6. the study coordinator trains the participants on the respective approach
7. the participant reads and clarifies possible doubts about their assigned approach
8. the participant identifies and represents variabilities in the E-Commerce class diagram according to their given approach
9. the participant is dismissed

Analysis Procedure

Once the experimental sessions are finished, the experiment coordinator prepares the collected data to calculate the effectiveness of each variability management approach (VMApp Effectiv) sample and, then, correlates such an effectiveness to the variability knowledge (VarKnowledge) of each study. Data preparation involves the tabulation of data and the calculation of descriptive statistics.

We present the effectiveness formula as follows:

$$effectiveness(z) = nVarC - nVarI$$

Where:

z = A given variability management approach
$nVarC$ = The number of correctly identified and represented variabilities according to the z approach
$nVarI$ = The number of incorrectly identified and represented variabilities according to the z approach

We applied normality tests to the effectiveness samples in order to decide which hypothesis test is applied.

Then, the coordinator correlates the participants' prior knowledge on variability to the obtained effectiveness in each study.

Figure 4 summarizes the procedure for the experiments analysis.

We can observe in Fig. 4 that, for our experiment, Shapiro-Wilk normality tests were performed to identify whether the samples are normally distributed. In a positive case, we perform the parametric T-Test hypothesis test.

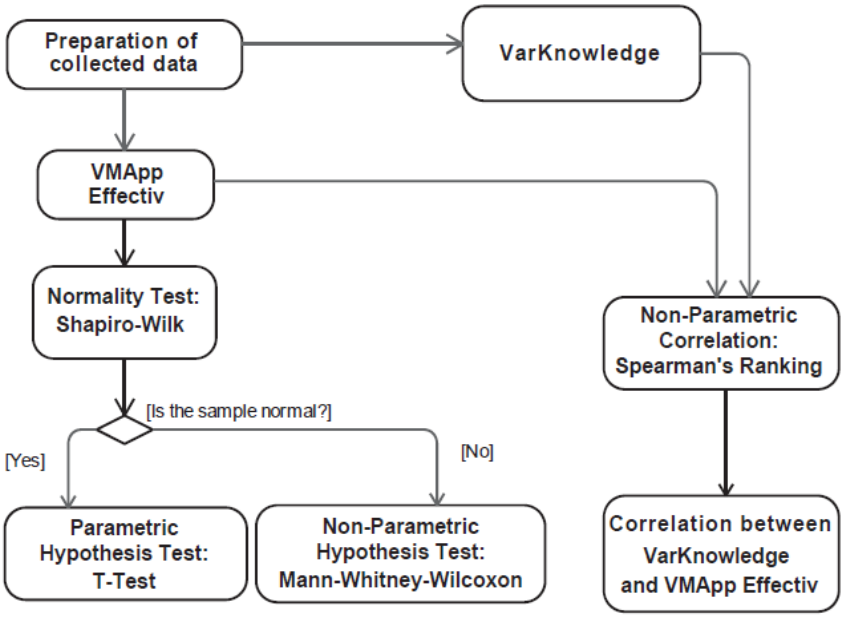
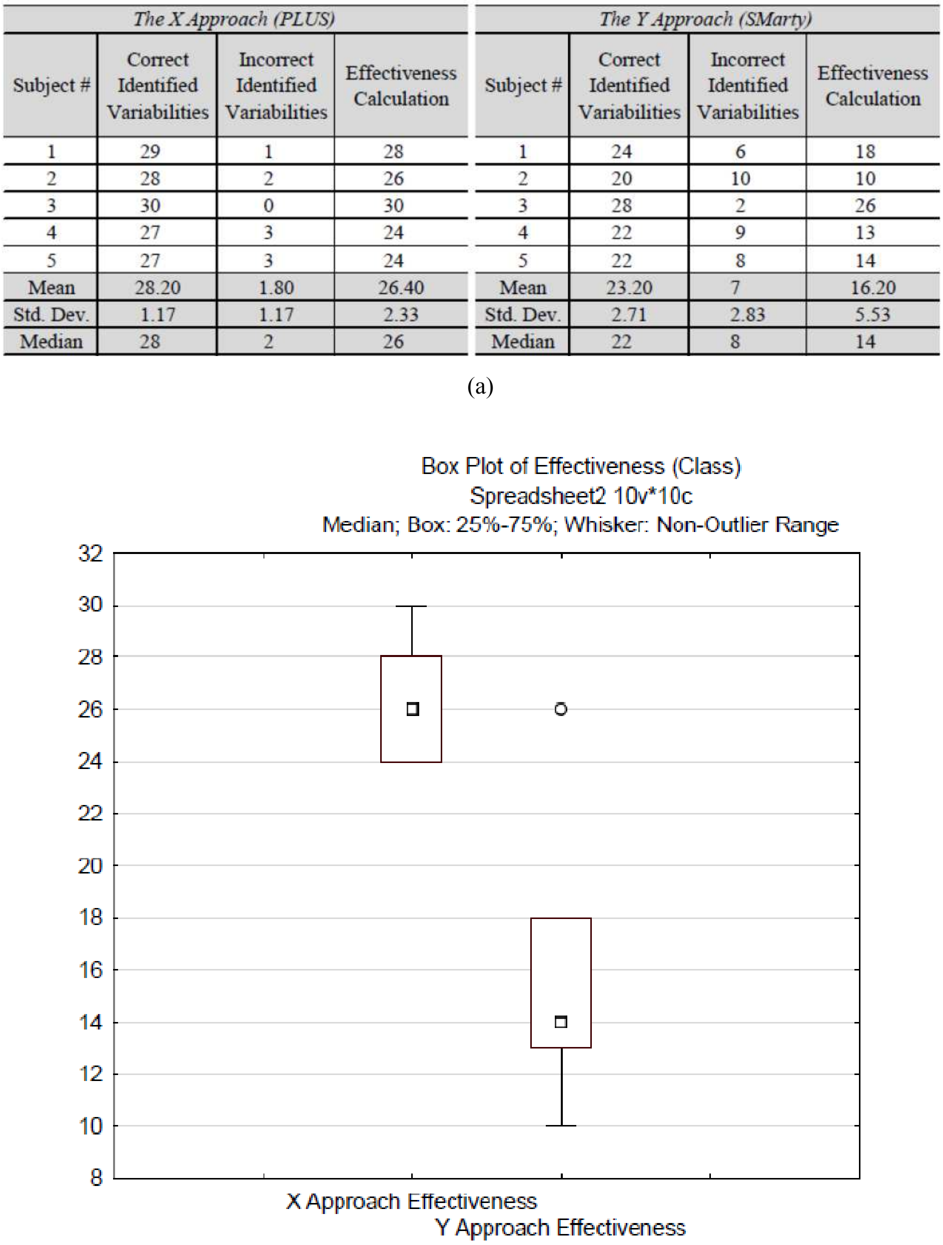

Preparation of collected data
VarKnowledge
VMApp Effectiv
Normality Test: Shapiro-Wilk
[Is the sample normal?]
[Yes]
[No]
Parametric Hypothesis Test: T-Test
Non-Parametric Hypothesis Test: Mann-Whitney-Wilcoxon
Non-Parametric Correlation: Spearman's Ranking
Correlation between VarKnowledge and VMApp Effectiv

Fig. 4: Analysis procedure of the experiments



The X Approach (PLUS)				*The Y Approach (SMarty)*			
Subject #	Correct Identified Variabilities	Incorrect Identified Variabilities	Effectiveness Calculation	Subject #	Correct Identified Variabilities	Incorrect Identified Variabilities	Effectiveness Calculation
1	29	1	28	1	24	6	18
2	28	2	26	2	20	10	10
3	30	0	30	3	28	2	26
4	27	3	24	4	22	9	13
5	27	3	24	5	22	8	14
Mean	28.20	1.80	26.40	Mean	23.20	7	16.20
Std. Dev.	1.17	1.17	2.33	Std. Dev.	2.71	2.83	5.53
Median	28	2	26	Median	22	8	14

(a)

(b)

Fig. 5: Experimental Study Data: (a) Collected Data, Descriptive Statistics and Effectiveness Calculation (b) Effectiveness Box plot

Otherwise, we perform the nonparametric Mann-Whitney-Wilcoxon U (MWW U) hypothesis test. In parallel, we realized the non-parametric Spearman's Ranking correlation with regard to the participants' variability knowledge (VarKnowledge) and the calculated effectiveness (VMApp).

Experimental Analysis

We present the analysis of our experiment in this section in terms of collected data, descriptive statistics and normality and hypothesis tests.

Collected Data and Descriptive Statistics

This section presents the collected data, descriptive statistics and the calculation of the effectiveness during the execution of our experiment.

Figure 5 presents: (a) collected data, descriptive statistics, effectiveness calculation; and (b) box plot for the effectiveness of the observed values. For each participant ("Subject #" column), the following data were collected for the approach they were assigned to: the number of correctly and incorrectly identified and represented elements of variabilities; and the effectiveness calculation.

Normality and Hypothesis Testing for R.Q.1

This section presents the normality and hypothesis tests performed for our experiment in order to answer R.Q.1 (Section 3.2.1).

We performed Shapiro-Wilk normality test for the E-Commerce and AGM samples providing the following results:

- for the *X* approach with sample size (*N*) 5, mean value (μ) 26.4, standard deviation value (σ) 2.33, *p* = 0.42, which means that, with α= 0.05 (0.42>0.05), the sampleis normal
- for the *Y* approach with sample size (*N*) 5, mean value (μ) 16.2, standard deviation value (σ) 5.53, *p* = 0.51, which means that, with α = 0.05 (0.51>0.05), the sample is normal

The T-Test was applied for *X* and *Y* samples. Firstly, the value of *T* was obtained, which allows the identification of the range in the statistical table *T* (*student*). This value is calculated using the average of the *Y* sample (μ1 = 16.2) and the *X* sample (μ2 = 26.4), standard deviation value of both (σ1 = 5.53 and σ2 = 2.33), and the sample sizes (N = 5). Thus, value $t_{calculated}$ = –4.07 was obtained.

By taking the sample size (N = 5), we obtained the degree of freedom (*df*), which, combined with the *t* value, indicates which value of *p* in the *T* table must be selected.

By searching the index *df* = 8 and defining the value *t* at the *T* table (*student*), a value for critical t of 2.3 ($t_{critial}$ = 2.3), with a significance level (α) of 0.05. Thus, comparing the $t_{critial}$ with the $t_{calculated}$ the null hypothesis $H_{0.RQ1}$ is rejected and ($H_{1.RQ1}$) is accepted ($t_{calculated}$(–4.3078) > = $t_{critial}$(–2.3)). It means that the X approach (PLUS method) is more effective than the Y approach (SMarty approach) when representing variability in class diagrams.

Correlation Analysis for R.Q.2

This section presents the correlation analysis of the participants, variability knowledge and the effectiveness for each performed study in order to answer R.Q.2 (Section 3.2.1).

We obtained the variability knowledge from the characterization questionnaire. It is collected by means of a Likert scale with five labes: "no previous experience", "heard something about", "basic experience", "moderate experience", and "advanced experience". Therefore, the observed values for variability knowledge are considered nonnormal.
Hence, the non-parametric correlation technique used was the Spearman's Ranking Correlation.

Equation 1 shows the formula to calculate the Spearman ρ correlation, where *n* is the sample size:

$$\rho = \left\{ 1 - \frac{6}{n(n^2-1)} \sum_{n(n^2-1)}^{n} d_i^2 \right. \qquad (1)$$

Table 5 presents the data needed to calculate the Spearman correlation for *X* and *Y* effectiveness and the participants variability knowledge.

Table 5: Spearman's Correlation Data between Effective and VarKnowledge of the Experimental Study

Subject #	Effective	r_{a1}	VarKnowle dge	r_{b1}	r_{a1}-r_{b1}	$d1^2$
The X Approach (PLUS - E-Commerce						
1	16	1.5	3	1	1	0
2	16	1.5	2	2.5	-1	1
3	14	3.0	1	4.5	-2	2
4	12	4.0	1	4.5	-1	0
5	10	5.0	2	2.5	3	6
The *X* Approach (*PLUS* - *AGM*)						
1	14	2.0	1	4.5	-3	6
2	14	2.0	3	1	1	1
3	14	2.0	1	4.5	-3	6
4	12	4.0	2	2.5	2	2
5	10	5.0	2	2.5	3	6
The *Y* Approach (*SMarty* - *E-Commerce*)						
1	26	1.0	2	4.5	-4	12
2	18	2.0	3	2.5	-1	0
3	14	3.0	2	4.5	-2	2
4	13	4.0	3	2.5	2	2
5	10	5.0	5	1	4	16
The *Y* Approach (*SMarty* - *AGM*)						
1	14	2.5	5	1	2	2
2	14	2.5	3	2.5	0	0
3	12	5.0	3	2.5	3	6
4	14	2.5	2	4.5	-2	4
5	14	2.5	2	4.5	-2	4

Thus, the following values for ρ were obtained, as well as the classification scale of Spearman:

- *CorrClass* ($X_{E\text{-}Commerce}$): ρ = 0.50 - Weak positive correlation
  CorrClass(X_{AGM}): ρ = –0.10 - Weak negative correlation
- *CorrClass*($Y_{E-Commerce}$): ρ = –0.65 - Strong negative correlation
- *CorrClass*(Y_{AGM}): ρ = 0.17 - Weak positive correlation

Discussion

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

Evaluation of the Results and Implications

The results obtained from the experiment carried out are explained in this section based on statistical significance and practical importance as suggested by (Kitchenham *et al*., 2002). Table 6 summarizes all the empirical study results, although they were analyzed and calculated individually.

As we can observe in our performed study, even with no statistically significant sample size, the study rejected the null hypothesis for both R.Q.1 and R.Q.2 with alpha 0.05, which is an acceptable significance level for hypothesis tests and correlation analysis. However, such results could not be generalized. On the other hand, these results provided practical-importance for the main research objectives of this paper.

In general, the evidence collected throughout statistical tests allowed to provide an indicator that PLUS had a superior effectiveness in variability management for class diagrams, as it can be observed in Table 6 and Section 3.3.

With regard to R.Q.1, issues related to class diagrams, such as the number of stereotypes used to represent variabilities by the PLUS method, seem to impact SMarty better results of effectiveness. As initial evidence, SMarty provides more essential information for consistent products derivation than PLUS, respecting models variability resolution. Thus, new studies must be conducted to evaluate whether this additional information, if compared to PLUS, allows the evaluation and which approach is better in deriving SPL products. The interpretation and derivation of products based on the UML models are one of the key aspects for the success of an SPL approach adoption.

It is clear that the PLUS method provides explanations about the constraints in a class diagram, but it needs more documents to help the user. On the other hand, SMarty provides full use of its profile applied to class diagrams. An important piece of evidence is related to the *SMartyProcess*. SMarty participants have less variability knowledge then PLUS and Ziadi *et al.* participants'. Thus, this is an indicator that the use of guidelines might have facilitated the identification of variabilities using SMarty.

Based on the effectiveness analysis, we can interpret that the guidelines may have facilitated the application of the SMarty stereotypes. This is, therefore, essential to obtain the benefits of SPL. This difference with relation to the other approaches may allow practitioners to adopt SMarty, assuring more quality to their models, besides the possibility of evolution of the SPL in general, as SMarty guidelines support discovering new variabilities.

Next paragraphs discuss the results of our study carried out with relation to R.Q.2.

The SMarty approach showed a strong negative correlation for the E-Commerce SPL and a weak positive correlation for AGM SPL. SMarty, by providing guidelines that assist in the process of identifying variabilities in different UML diagrams, obtained results which do not seem to be due, at first, to the prior variability participants knowledge, unlike PLUS.

PLUS presented weak positive and weak negative correlations, which might lead to interpret that the variability previous participants knowledge had, possibly, a major influence on the effectiveness of PLUS. Another predominant fact to the PLUS greater effectiveness is that it has only two stereotypes, one for the commonalities (<<kernel>>) and one for variabilities (<<optional>>), then the AGM and the E-Commerce elements modeling become trivial.

Table 6: Summary of study results

	UML Diagram	Class Diagrams	
	VMApp	PLUS	SMarty
R.Q.1	Effectiv (VMApp)	26.4	16.2
	Reject $H_{0,RQ1}$?/Accepted Hypothesis?	PLUS is more effective Yes / $H_{2,RQ1}$	
R.Q.2	Corr (VarKnowledge, VMApp)	0.22 Weak Positive	0.29 Weak Positive
	Reject $H_{0,RQ2}$? / Accepted Hypothesis?	Yes/$H_{1,RQ2}$	

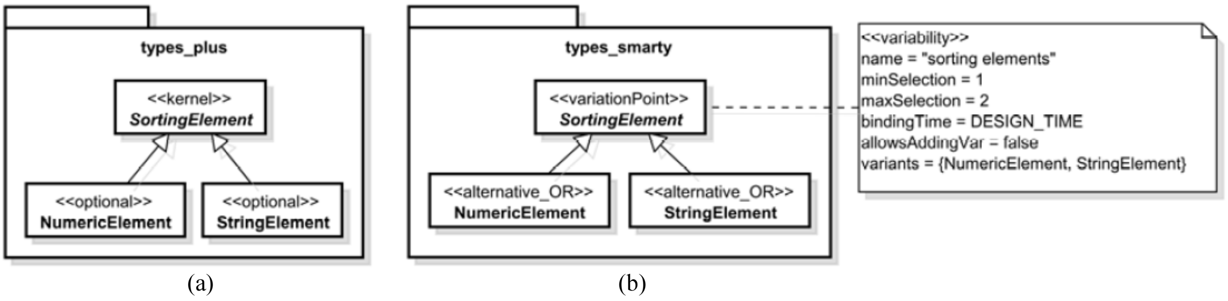


Fig. 6: Variability excerpt of an SPL modeled with (a) PLUS and (b) SMarty

Apparently, PLUS makes the process of modeling classes easier, but it might complicate the process of deriving specific products. As an example, when an SPL engineer faces the stereotype <<optional>>, he/she may understand that the stereotype represents a variant or variability of mutually exclusion. If the choice was wrong, the generated product will be incorrect. This example is shown in Fig. 6. Figure 6b presents the application of inclusive stereotypes, which is easy to identify (SMarty approach). Figure 6a shows the same variability, but using PLUS.

The optional constraint variant indicates that the variant can be included or not in a derived product; thus, using a stereotype to represent alternative variants with PLUS may lead the SPL architect to generate inconsistent products. Hence, to solve this issue, separate variability descriptions are necessary to make the derivation of correct products legal.

Threats to Validity

One of the key issues in an empirical study is evaluating the validity of results (Wohlin *et al*., 2000). In this section, we discuss the potential threats that are relevant for our study. For each threat, we describe the actions we took to address them according to the Conceptual Model of Neto and Conte (2013).

Internal Validity

One of the main internal threats was the experiment duration: 80 minutes. Participants from long-term experiments may get bored during the experiment, thus we allowed them to “take a break”, go to the toilet or have something to eat. In order to avoid participant communication, we introduced human observers in the experimental environment, even in the “break area”.

Another threat was the training effect. We consider that the quality of the trainings on each approach could have affected the overall results if the training on the *X* approach was inferior to the one on the *Y* approach. Thus, we provided variability management training sessions in a similar way avoiding biases.

External Validity

An important threat identified in our study was the participants’ representativeness.

As our sample was small, a heterogeneous sample was not possible, thus reducing the participant representativeness. However, we provided more reliability to the conclusion, selecting homogeneous samples.

Construct Validity

The main threat observed in our study was the different behavior of the participants when they were observed. As we introduced human observers to the experiment environment to reduce potential internal threats, we could have performed preliminary tests in order to familiarize the participants with such an environment. However, as it could have affected their performance, we have chosen not to perform preliminary tests.

Conclusion Validity

We acknowledge that the small number of data points is not ideal from the statistical point of view. Since the number of participants is reduced, the data extracted from these study can only be considered indicators and not conclusive. Nonetheless, according to (Wohlin *et al*., 2000), it might not be possible to get homogeneous samples; hence the statistical conclusions may be drawn with less significance. In this sense, even with small samples, the results from this study were important for the evaluation and potential evolution of SMarty and PLUS.

Related Work

The majority of VM approaches has not yet been evaluated using rigorous scientific methods (Chen and Babar, 2011). In addition, a large majority of empirical evaluations in SPL had not been sufficiently designed or reported (Ahnassay *et al*., 2014). Thus, a few works focused on empirically evaluating UML-based VM approaches in terms of the effectiveness of expressing variabilities can be found in the literature.

Reinhartz-Berger and Sturm (2014) examined the comprehensibility of domain models specified in

ADOM, an SPL method. They conducted a controlled experiment in which 116 undergraduate students answered comprehension questions with regard to a domain model with explicit reuse guidance and/or variability specification. Although the experiment showed that explicit specification of variability increased comprehensibility only to a limited extent, specification of reuse guidance without variability was better than variability without specification of reuse guidance. Reinhartz-Berger and Sturm adopted the definition of comprehensibility related to requiring participants to perform tasks that assess the participants ability to use the knowledge represented in a schema, then determine whether and how certain information is available and correct from the schema. In the same way, in our study, the participants were trained with specific variability management approach/method and were required to perform variability modeling in SPL UML diagrams using specific acquired knowledge. Then, we analyzed whether the modeling is correct. Although both Reinhartz-Berger and Sturm study and ours analyzed the same UML class diagrams type, the main difference is that our study compared the effectiveness of SMarty with relation to PLUS, whereas Reinhartz-Berger and Sturm analyzed reuse guidance versus variability specification, but they did not compare the approaches/methods.

The study presented by Genero *et al*. (2008) is not directly related to empirical evaluation of variability management based on UML, but we judge it importantas it aims at identifying the effectiveness of the use of stereotypes in UML diagrams. An experiment was planned and conducted with two replications, in which students identified their level of understanding in UML diagrams with and without stereotypes. Evidence showed that the stereotypes in such diagrams could help to improve the understanding of the modeling. Such a work is interesting for providing background on the effectiveness analysis, as we proposed in this paper, as well as demonstrating the importance of providing guidelines/instructions on how to get correct stereotype-based modeling diagrams for prospective product derivations.

Conclusion

New theories and technologies must be empirically evidenced before they can be accepted in industry and effectively be adopted by software engineering practitioners. In this paper, it is shown how the effectiveness of variability management approaches (SMarty and PLUS) was analyzed to facilitate and improve VM activities in an SPL perspective. Effectiveness was analyzed by modeling variability in SMarty class diagrams.

The parametric T-Test was applied to the sample. These test analyzed the effectiveness of the PLUS method and the SMarty approach. Then, the correlation of the level of knowledge in SPL and variability was performed based on the Spearman correlation test.

The results obtained provided initial evidence that PLUS is more effective than SMarty for VM in UML class diagrams.

Since variability management is important to keep any SPL and their products consistent, PLUS and SMarty can be initially seen as accurate approaches to support effective identification and representation of variabilities. Therefore, the main contributions of this paper rely on: (i) an empirical comparative analysis of the effectiveness of UML stereotype-based VM approaches; (ii) the empirical evaluation of the SMarty and PLUS; and (iii) the improvement of the identification and representation of variability in UML-based VM modeling by potentialy enhancing SMarty and PLUS.

New empirical studies and replications will be planned and conducted to make it possible to reducing the threats to validity, increasing the effectiveness of SMarty/PLUS and generalizing the results. As new studies: (i) we plan to replicate internally and externally this study to corroborate the obtained results; (ii) we will verify the results to identify possible VM approaches improvements; (ii) we are planning an empirical study to analyze the effectiveness of SMarty for use case and sequence diagrams using SPLs more representative (real) practitioners from industry; and (iv) we are planning to conduct an empirical study to analyze the accuracy of deriving products from SPL models with SMarty and PLUS approaches taking use case and sequence diagrams into consideration.

Acknowledgments

The authors would like to thank CAPES, a Brazilian Funding Agency and the Pontifical Catholic University of Rio Grande do Sul (PUCRS) students and lecturers for participating in the experimental study.

Author’s Contributions

Anderson S. Marcolino: Contributed in the design, execution and analysis and interpretation, and sharing of this controlled experiment.

Edson OliveiraJr: Designed the experiment, supervised the experiment running and analyzed collected data. Both authors equally contributed at writing and reviewing this paper.

Ethics

Participation during pilot and experiment trials are voluntary and participants are made known that their feedbacks will be contributing to a non-profit research project.

References

Abdelnabi, Z., G. Cantone, M. Ciolkowski and D. Rombach, 2004. Comparing code reading techniques applied to Object-oriented software frameworks with regard to effectiveness and defect detection rate. Proceedings of the International Symposium on Empirical Software Engineering, Aug. 20-20, IEEE Xplore Press, Redondo Beach, pp: 239-248. DOI: 10.1109/ISESE.2004.1334911

Ahnassay, A., E. Bagheri and D. Gasevic, 2014. Empirical evaluation in software product line engineering. Technical Report TR-LS3-130084R4T, Ryerson University, Laboratory for Systems, Software and Semantics.

Basili, V.R. and R.W. Selby, 1987. Comparing the effectiveness of software testing strategies. IEEE Trans. Software Eng., 13: 1278-1296. DOI: 10.1109/TSE.1987.232881

Berger, T., R. Rublack, D. Nair, J. Atlee and M. Becker *et al*., 2013. A Survey of variability modeling in industrial Practice. Proceedings of the 7th International Workshop on Variability Modelling of Software-intensive Systems, Jan. 23-25, ACM. New York, USA, pp: 1-8. DOI: 10.1145/2430502.2430513

Bosch, J., R. Capilla and R. Hilliard, 2015. Trends in systems and software variability [Guest editors' introduction]. IEEE Software, 32: 44-51. DOI: 10.1109/MS.2015.74

Coteli, M.B., 2013. Testing effectiveness and effort in software product lines. Master's Thesis, Middle East Technical University.

Capilla, R., J. Bosch and K.C. Kang, 2013. Systems and Software Variability Management: Concepts, Tools and Experiences, 1st Edn., Springer Science and Business Media, ISBN-10: 3642365833, pp: 317.

Chen, L. and M.A. Babar, 2011. A systematic review of evaluation of variability management approaches in software product lines. Inform. Software Technol., 53: 344-362. DOI: 10.1016/j.infsof.2010.12.006

Chen, L., M. Ali Babar and N. Ali, 2009. Variability management in software product lines: A systematic review. Proceedings of the International Software Product Line Conference, Aug. 24-28, Carnegie Mellon University, pp: 81-90.

Cruz-Lemus, J.A., M. Genero, D. Caivano, S. Abrahão and E. Insfrán *et al*., 2011. Assessing the influence of stereotypes on the comprehension of UML sequence diagrams: A family of experiments. Inform. Software Technol., 53:1391-1403. DOI: 10.1016/j.infsof.2011.07.002

Galster, M., D. Weyns, D. Tofan, B. Michalik and P. Avgeriou, 2014. Variability in software systems: A systematic literature review. IEEE Trans. Software Eng., 40: 282-306. DOI: 10.1109/TSE.2013.56

Genero, M., J. Cruz-Lemus, D. Caivano, S. Abrah˜ao and E. Insfrán *et al*., 2008. Assessing the influence of stereotypes on the comprehension of UML sequence Diagrams: A controlled experiment. Proceedings of the International Conference on Model Driven Engineering Languages and Systems, (MODELS' 08), Springer International Publishing AG, pp: 280-294. DOI: 10.1007/978-3-540-87875-9_20

Gomaa, H., 2004. Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures. Addison Wesley, Redwood City, CA, USA.

Heinsman, D. and W. Shadish, 1996. Assignment methods in experimentation: When do nonrandomized experiments approximate answers from randomized experiments? Psychol. Meth., 1: 154-169. DOI: 10.1037/1082-989X.1.2.154

ISO/IEC, 2004. Software Engineering-Product Quality - Part 4: Quality in Use Metrics.

Jedlitschka, A., M. Ciolkowski and D. Pfahl, 2007. Reporting Experiments in Software Engineering. In: Guide to Advanced Empirical Software Engineering, Shull, F., J. Singer and D.I. Sjøberg (Eds.)., Springer-Verlag New York, Inc., Secaucus, NJ, USA.

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

Marcolino, A., E. OliveiraJr and I. Gimenes, 2014. Towards the effectiveness of the SMarty approach for variability management at sequence diagram level. Proceedings of the International Conference on Enterprise Information Systems, April 27-30, Lisboa, Portugal, pp: 249-256. DOI: 10.5220/0004889302490256

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

Martinez-Ruiz, T., F. Garcia, M. Piattini and J. Munch, 2011. Modelling software process variability: An empirical study. IET Software, 5: 172-187. DOI: 10.1049/iet-sen.2010.0020

Neto, A.A. and T. Conte, 2013. A conceptual model to address threats to validity in controlled experiments. Proceedings of the International Conference on Evaluation and Assessment in Software Engineering, April 14-16, New York, USA, pp: 82-85. DOI: 10.1145/2460999.2461011

OliveiraJr, E., I. Gimenes, J. Maldonado and P. Masiero, 2013. Systematic evaluation of software product line architectures. J. Universal Comput. Sci., 19: 25-52.

OliveiraJr, E., I.M.S. Gimenes and J.C. Maldonado, 2010. Systematic management of variability in UML-based software product lines. J. Universal Comput. Sci., 16: 2374-2393.

OMG, 2011. Unified Modeling Language, Superstructure Version 2.4.1.

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

Sanen, F., E. Truyen and W. Joosen, 2009. Mapping problem-space to solution-space features: A feature interaction approach. SIGPLAN Notes, 45: 167-176. DOI: 10.1145/1837852.1621633

Schaefer, I., A. Lamprecht and T. Margaria, 2011. Constraint-oriented variability modeling. Proceedings of the IEEE Software Engineering Workshop, Jun. 20-21, IEEE Xplore Press, Limerick, Ireland, pp: 77-83. DOI: 10.1109/SEW.2011.17

Thurimella, A.K. and B. Bruegge, 2012. Issue-based variability management. Inform. Software Technol., 54: 933-950. DOI: 10.1016/j.infsof.2012.02.005

Wilkinson, L., 1999. Statistical methods in psychology journals: Guidelines and explanations. Am. J. Psychol., 54: 594-604. DOI: 10.1037/0003-066X.54.8.594

Wohlin, C., P. Runeson, M. Host, M.C. Ohlsson and B. Regnell *et al*., 2000. Experimentation in software engineering: an introduction. Kluwer Academic Publishers, Norwell, MA, USA.