

Avaliação Experimental da Abordagem SMarty para Gerenciamento de Variabilidade em Linhas de Produto de Software Baseadas em UML

Anderson S. Marcolino, Edson Oliveira Jr

¹Programa de Pós-Graduação em Ciência da Computação (PCC)
Universidade Estadual de Maringá (UEM)
Departamento de Informática, Av. Colombo, 5790 - Zona 07
CEP 87020-900 - Maringá - PR - Brasil

andersonmarcolino@gmail.com, edson@din.uem.br

Abstract. *Software product line approach (SPL) allows reuse of many software artifacts by means of the variability management activity (VM), which manages variable artifacts in a SPL. To support the efficient management, approaches are adopted. Stereotype-based Management of Variability (SMarty) is an UML-based approach for VM and as other approaches, it needs experimental evidence of effectiveness to make it feasible to be applied to academic and industrial environments. This paper shows the extension of SMarty to identify variabilities in UML sequence models, as well as the realization of a set of experimental studies for evaluating the SMarty approach, in comparison with other significant approaches from the current literature. The results provided initial evidence that SMarty is effective to manage variabilities, increasing the quality and allowing the derivation of specific products using UML-based SPLs.*

Resumo. *A abordagem de Linha de Produto de Software (LPS) permite a reutilização de artefatos de software por meio da atividade de gerenciamento de variabilidades (GV), que gerencia os artefatos variantes de uma linha. Para apoiar o gerenciamento efetivo, abordagens são adotadas. Stereotype-based Management of Variability (SMarty) é uma abordagem de GV baseada em UML, e assim como as demais abordagens existentes, carecem de evidências de sua efetividade para que possam ser aplicadas em âmbito acadêmico e industrial. Este artigo apresenta a extensão de SMarty para a identificação de variabilidades em modelos de sequência da UML, bem como a realização de um conjunto de estudos experimentais que avaliam a efetividade de SMarty em comparação com outras abordagens significativas da literatura atual. Os resultados evidenciam que SMarty se mostra efetiva para gerir variabilidades, aumentando o nível de qualidade e garantia de derivação de produtos específicos em LPSs.*

1. Introdução

O panorama competitivo das companhias de desenvolvimento de software está mudando e se intensificando. O tamanho e a complexidade dos sistemas, juntamente com clientes mais exigentes e, assim, maior demanda por produtos altamente customizáveis, [Capilla et al. 2013] tem levado a adoção de paradigmas de reuso, com destaque para linhas de produto de software (LPS). Uma LPS, ou família de produtos, é um conjunto de

sistemas de software que compartilham características (*features*) comuns e gerenciáveis, que satisfazem as necessidades de um segmento particular ou de uma missão [SEI 2012].

Os membros da família são produtos específicos derivados sistematicamente a partir de uma infraestrutura comum de uma LPS, chamada núcleo de artefatos (*Core assets*) [SEI 2012]. Este núcleo é formado por um conjunto de características comuns (similaridades) e características variáveis (variabilidades) [Linden et al. 2007]. As variabilidades correspondem aos diversos artefatos que são reutilizados e estão associadas a diferentes níveis de abstração, dentre eles: a descrição da arquitetura, a documentação de projeto, o código fonte, o código compilado, o código ligado e o código executável [Capilla et al. 2013].

A obtenção dos benefícios propostos pela abordagem de LPS, tais como: redução de custo e tempo no processo de desenvolvimento, diminuição de riscos e de perdas, redução do *time to market* e melhora na qualidade, dependem do sucesso da atividade de gerenciamento das variabilidades (GV). Esta atividade é composta por quatro subatividades: a identificação de variabilidade, a delimitação de variabilidade, a implementação, e por fim, o gerenciamento de variantes [Chen et al. 2009, Capilla et al. 2013]. Devido a importância do GV, diversas propostas vêm sendo apresentadas pela literatura [Galster et al. 2013, Chen et al. 2009]. Entre elas, destacam-se as que utilizam a linguagem de modelagem unificada (UML): (i) a abordagem *Stereotype-based Management of Variability* (SMarty) [OliveiraJr et al. 2013, Fiori et al. 2012, OliveiraJr et al. 2010], (ii) o método *Product Line UML-based Software Engineering* (PLUS) [Gomaa 2004] e (iii) a abordagem de Ziadi et al. [Ziadi and Jezequel 2006].

Apesar da existência da abordagem SMarty e outras abordagens para o GV, a literatura é incipiente quanto a estudos experimentais sobre a efetividade destas abordagens em LPSs baseadas em UML, como apontou um mapeamento sistemático conduzido. Tais evidências são fundamentais para que a indústria possa adotar abordagens adequadas para GV em LPS, pois a indústria, segundo Mafra e Travassos [Mafra and Travassos 2005], ainda apresenta imaturidade quanto à escolha de processos e tecnologias, gerando incertezas e falta de credibilidade nos engenheiros de software, quando relacionados a novas propostas e à adoção de tecnologias emergentes, comprometendo a qualidade dos produtos derivados de uma LPS. E ainda, a lacuna entre pesquisa e prática, na utilização de tais abordagens, as colocam em questionamento quanto sua real efetividade [Galster et al. 2013, Chen et al. 2009].

Outra lacuna constatada é que SMarty, em sua versão 4.0, permite que variabilidades possam ser identificadas e representadas em modelos UML de casos de uso, classes, componentes e atividades. Porém, nota-se a necessidade de gerenciar variabilidades em modelos comportamentais da UML, com maior nível de detalhes, como o diagrama de sequência, já que tais modelos são de fundamental importância para LPSs nos processos de negócio e são pouco explorados em abordagens presentes na literatura, conforme identificado em mapeamento sistemático conduzido.

Destarte, com base nestas constatações, a Figura 1 apresenta os problemas e lacunas identificados, os objetivos almejados na condução do trabalho apresentado neste artigo, as metodologias empregadas para se atingir tais objetivos e a síntese dos principais resultados alcançados. Identificadores foram utilizados para facilitar o rastreamento dos

elementos e de suas relações no decorrer deste artigo (i.e Objetivo 3 identificador **O3**).

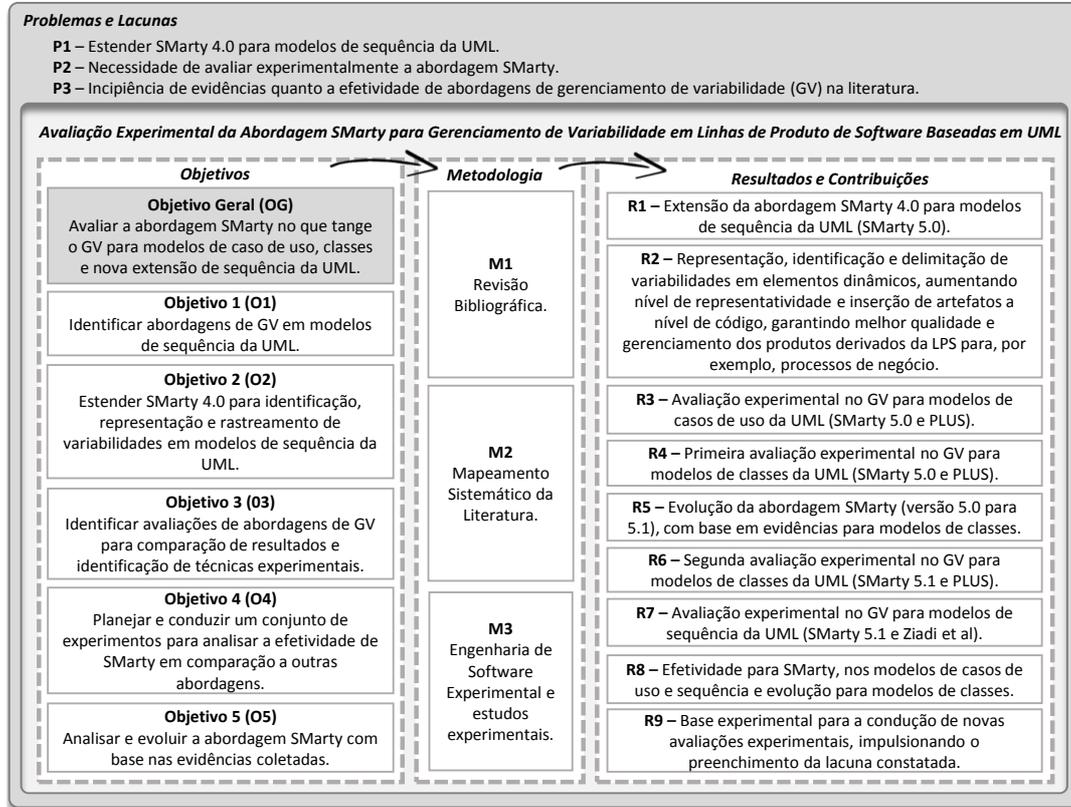


Figura 1. Problemas, Objetivos, Metodologia e Resultados da Dissertação.

A Figura 2 apresenta a dispersão temporal das atividades da Figura 1. A revisão bibliográfica (**M1**) da literatura forneceu embasamento teórico sobre LPS e conceitos relacionados. Já os dois mapeamentos sistemáticos (**M1** e **M2**) permitiram identificar: (i) as abordagens que apoiam modelos de sequência para propor a extensão de SMarty; e (ii) estudos experimentais para avaliação de abordagens de GV existentes, respectivamente.

A Figura 2 apresenta, ainda, as datas estimadas da realização dos quatro experimentos conduzidos para avaliar as abordagens de GV baseadas em UML para os modelos de casos de uso (**R3**), classes (**R4** e **R6**) e sequência (**R7**), assim como a extensão para modelos de sequência da UML da abordagem SMarty, versão 4.0 para 5.0 (**R1** e **R2**) e sua evolução na versão 5.1 (**R5**), após evidências coletadas no primeiro experimento de classes (**R4**) conduzido.

As demais seções deste artigo estão organizadas como segue. A Seção 2 apresenta a fundamentação teórica sobre gerenciamento de variabilidades com UML, a abordagem SMarty em sua versão 4.0 e as demais abordagens selecionadas na literatura para serem avaliadas com SMarty. A Seção 3 apresenta a extensão da abordagem SMarty para modelos de sequência. Na Seção 4 são apresentadas as avaliações conduzidas para obter evidências quanto a efetividade da abordagem SMarty. Finalmente, a Seção 5 apresenta as considerações finais e trabalhos futuros.

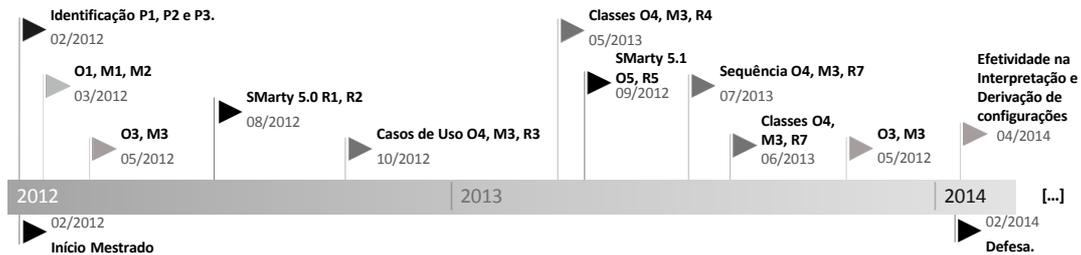


Figura 2. Linha do Tempo das Atividades Realizadas.

2. Gerenciamento de Variabilidades com UML

Segundo Weiss e Lai [Weiss and Lai 1999], variabilidade é a forma como os membros de uma família de produtos podem se diferenciar entre si. Variabilidade é o que permite distinguir os diversos produtos de uma LPS. Variabilidade é a habilidade de um software ou artefato ser modificado, personalizado ou configurado para uso em um contexto particular. As variabilidades surgem do adiamento de certas decisões do projeto de produtos de software. Se tomadas nas fases iniciais do projeto, se enquadram na abordagem tradicional de desenvolvimento. Ao postergar-se decisões de projeto, permite-se especificar as variabilidades, para uma posterior resolução destas. A resolução se dá quando ocorre a indicação da associação de uma ou mais variantes a determinado ponto de variação [Capilla et al. 2013].

Os pontos de variação são as resoluções de variabilidades em artefatos genéricos de uma LPS. De acordo com Jacobson et al. [Jacobson et al. 1997], “um ponto de variação identifica uma ou mais localidades nas quais a variação irá ocorrer” e podem ocorrer em artefatos genéricos e em diferentes níveis de abstração. Já as variantes, representam os possíveis elementos pelos quais um ponto de variação pode ser resolvido e também podem representar uma maneira de resolver uma variabilidade diretamente.

Para a resolução de um ponto de variação, mediante a seleção de uma ou mais variantes, existem as restrições entre variantes (i.e opcional, alternativa exclusiva, alternativa inclusiva, obrigatória, etc). Tais restrições definem os relacionamentos entre uma ou mais variantes, permitindo assim resolver um ponto de variação ou uma variabilidade [Linden et al. 2007] em seu devido tempo de resolução. O tempo de resolução pode ser de compilação, de ligação, de execução e de atualização [Linden et al. 2007].

A quantidade significativa de variabilidades e seus elementos para representar os diversos artefatos com níveis de abstração diferentes, refletem diretamente na precisão com que os produtos de uma família serão derivados, e também nas diversas abordagens que objetivam facilitar o GV. Deste modo, a literatura apresenta diferentes ferramentas e metodologias utilizadas para a representação de elementos que compõem a variabilidade. Existem abordagens baseadas em modelos de características [Kang et al. 1990], as que definem uma linguagem de domínio específica ¹ (DSL) [Linden et al. 2007] e as que utilizam notações conhecidas, como a UML [Gomaa 2004].

A linguagem de modelagem unificada (UML) é uma notação padronizada, amplamente aceita, que expressa em seus modelos representações que auxiliam no processo

¹do termo em inglês *domain specific language*

de desenvolvimento de software [OMG 2011]. A possibilidade de ampliar semanticamente seus elementos, visando atender novas especificações por meios de mecanismos de extensão, como os perfis UML e consequentemente, receber apoio de ferramentas automatizadas que manipulam arquivos XML *Metadata Interchange* (XMI), apresenta-se como modo atrativo e relevante, para representar as variabilidades nas abordagens existentes. Entre as abordagens que utilizam a UML, recebem destaque o método PLUS [Gomaa 2004], a abordagem de Ziadi et al. [Ziadi and Jezequel 2006] e SMarty.

O método *Product Line UML-based Software Engineering* (PLUS) [Gomaa 2004], permite o desenvolvimento de LPS por meio da sua integração a outros modelos de processo de software, como o processo de desenvolvimento unificado (*Rational Unified Process*). PLUS objetiva a modelagem explícita das características comuns e variáveis de uma LPS, por meio de atividades apoiadas por extensões UML, para modelos de casos de uso e classes. Inexiste um perfil UML, de modo que apresente a definição de meta-atributos e metaclasses presentes na UML, assim como não possui um processo que forneça diretrizes para a aplicação dos seus estereótipos na identificação de variabilidades em elementos UML. Deste modo, PLUS utiliza apenas estereótipos para a identificação das variabilidades e seus elementos no modelo da UML utilizado.

Ziadi et al. [Ziadi and Jezequel 2006] propõem uma das mais representativas abordagens de gerenciamento de variabilidades para modelos de sequência da UML, obtidos por meio da condução de mapeamento sistemático. A abordagem é composta de um perfil UML 2.0 que permite a integração com ferramentas UML, o que possibilita a representação de variabilidades para modelos de sequência. Há um conjunto de meta-atributos explícitos e metaclasses para permitir a anotação dos diversos elementos na representação de variabilidades. Contudo, não possui um conjunto de diretrizes para auxiliar sua aplicação e entendimento.

PLUS é considerado um dos métodos mais difundidos na literatura [Foschiani et al. 2013]. Assim, esse método foi selecionado na condução de avaliações experimentais (**R3**, **R4** e **R6**), pelo seu reconhecimento junto à comunidade acadêmica e por apoiar modelos UML (casos de uso e classes). Já a abordagem de Zaidi et al. foi considerada por ter um perfil especificado e estereótipos para modelos de sequência, o que outras abordagens não possuíam, e também, por ter sido retornada por meio de mapeamento sistemático conduzido. Assim, a abordagem de Ziadi et al. contribuiu para a extensão de SMarty (**R1** e **R2**) para modelos de sequência e foi utilizada na avaliação de SMarty (**R7**) no experimento executado para tal modelo.

2.1. A Abordagem SMarty 4.0

A abordagem *Stereotype-based Management of Variability* (SMarty), proposta com base nos conceitos vistos sobre UML, LPS e variabilidades, possui suporte, em sua versão 4.0, a modelos UML de casos de uso, classes, atividades e componentes. Composta de um perfil UML, o *SMartyProfile*, e do processo denominado *SMartyProcess*. SMarty tem como objetivo: permitir que as variabilidades de uma LPS possam ser gerenciadas de forma clara e explícita em modelos UML [OliveiraJr et al. 2013, Fiori et al. 2012, OliveiraJr et al. 2010]; e guia o usuário por meio do *SMartyProcess* na identificação e representação de variabilidades em modelos UML de uma LPS.

O perfil *SMartyProfile* é formado por um conjunto de estereótipos e meta-atributos

para representar variabilidades em modelos UML de LPS. Por meio da UML e seu mecanismo de perfil, *SMarty* permite a representação explícita de variabilidades. O *SMartyProfile* baseia-se no inter-relacionamento dos principais conceitos de LPS, apresentado previamente. Tais conceitos são aplicados aos elementos de interesse do metamodelo da UML, se beneficiando do conceito de pontos de extensão da UML em casos de uso, e de recursos próprios de cada modelo que suporta para representar relações entre pontos de variação e suas variantes. Ainda, a cardinalidade de variantes é representada na forma de um meta-atributo. Logo, a representação de variabilidades fica totalmente compatível com os metamodelos da UML e permite o apoio de ferramentas UML.

O *SMartyProcess* é um processo sistemático que guia o usuário na identificação, delimitação, representação, rastreamento de variabilidades e análise de configurações de produtos de uma LPS. Nele há um conjunto de diretrizes que permitem ao usuário a aplicação dos estereótipos do *SMartyProfile* de forma clara e objetiva. O *SMartyProcess* segue as atividades gerais relacionadas às especificadas no processo de desenvolvimento de LPS. [Pohl et al. 2005] [SEI 2012]. *SMarty* combina o *SMartyProfile* e o *SMartyProcess*, gerando uma abordagem guiada por diretrizes para gerenciar sistematicamente variabilidades de LPS. O *SMartyProcess* é realizado pelo engenheiro de LPS e é um processo iterativo e incremental. Iterativo, pois ocorre após a execução de cada atividade de desenvolvimento de LPS e incremental, pois o número de variabilidades tende a crescer, à medida que as atividades do *SMartyProcess* são executadas.

3. Extensão da Abordagem SMarty para Modelos de Sequência

SMarty 4.0 possui suporte aos modelos de UML de casos de uso, classes, atividades e componentes. Contudo, somente modelos de caso de uso e classes de domínio não são suficientes para apresentar uma visão completa do sistemas para que a fase de implementação ocorra. É na fase de projeto de interação que tais definições são feitas, mais especificamente no detalhamento de aspectos dinâmicos do sistema. Detalhamentos refletidos por meio de diagramas de interação, como o de sequência. A importância dos elementos e dos aspectos que os diagramas de sequência apresentam permitem a representação dos elementos que compõem as variabilidades, neste contexto é proposta a extensão da abordagem SMarty 4.0 para apoiar tais modelos.

Para a extensão do perfil de SMarty 4.0 foram identificados os elementos da UML que representam as restrições existentes entre as variabilidades (**M1 e M2**), bem como os elementos candidatos a pontos de variação e variantes. A Figura 3 apresenta um dos exemplos da extensão de SMarty para a identificação de um ponto de variação e de suas variantes mutuamente exclusivas, por meio do *CombinedFragment* do tipo “alt”. O fragmento de uma LPS é apresentado (Figura 3) e corresponde a uma câmera fotográfica, que possui uma interface com usuário (*Interface*), o sensor de captura (*Sensor*), o compressor das imagens capturadas (*Compressor*) e a memória (*Memory*) onde tais imagens são armazenadas.

A mensagem *StartCapture()* é obrigatória, e por convenção, elementos obrigatórios não deverão receber estereótipos. O *CombinedFragment* com o *interactionOperator* “alt”, indica que apenas um fluxo pode ser selecionado para a execução, logo, uma câmera que possuir ambas as variantes; representadas no conteúdo do *CombinedFragment* estereotipado por <<variationPoint>>, e também especifi-

cado como variabilidade, pelo comentário UML (`<<variability>>`), poderá ter ou não a opção de compressão (mensagem `Compress (Data)`) na solução do ponto de variação.

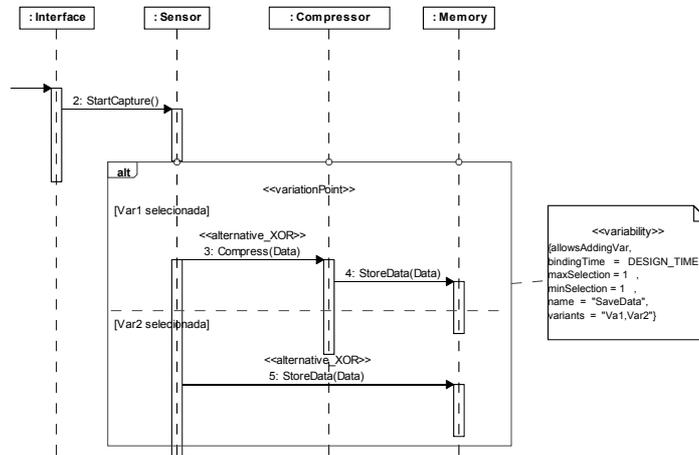


Figura 3. Excerto de Variabilidade em Diagrama de Sequência com SMarty.

Ainda no `CombinedFragment`, a primeira mensagem trocada internamente, em cada operando (`Var1` e `Var2`) são estereotipadas de acordo com o tipo de variação que sofrem. Neste exemplo de LPS, são estereotipadas como `<<alternativa_XOR>>`, assim, haverá somente uma variante selecionada para o ponto de variação. No exemplo, são apresentadas duas opções, mas poderiam ser inseridos um número arbitrário, maior que 1, de variantes. As variabilidades são identificadas na abordagem `SMarty` por meio do comentário UML, estereotipada com `<<variability>>`.

Com a extensão de novos elementos do metamodelo padrão da UML para sequência, foram criadas 6 novas diretrizes no `SMartyProcess` para apoiar a identificação e a representação de variabilidades por meio do `SMartyProfile`. Uma destas diretrizes é exemplificada a seguir:

SQ.1 Elementos de diagramas de sequência como `CombinedFragment` que possuem do `interactionOperator` do tipo “alt”(alternativa), indicam que apenas um fluxo do `CombinedFragment` será realizado, ou seja, sugerem variantes mutuamente exclusivas onde os pontos de variação serão anotados como `<<variationPoint>>` e serão relacionados a um comentário da UML especificando a variabilidade (`<<variability>>`). As variantes correspondentes às mensagens devem ser estereotipadas como `<<alternative_XOR>>`.

Com `SMarty` em sua versão 5.0, foi conduzida a avaliação experimental de `SMarty`, em relação ao método PLUS [Gomaa 2004], para casos de uso e classes, e abordagem de Ziadi et al. [Ziadi and Jezequel 2006] para modelos de sequência. Destaca-se que os procedimentos experimentais conduzidos foram especificados de tal modo que possam ser reutilizados na avaliação de outras propostas.

4. Avaliação da Efetividade da Abordagem SMarty

Avaliações experimentais são fundamentais no processo de evidenciar se novas ideias e teorias emergentes, seja de uma abordagem, de um novo paradigma de programação, ou

novas tecnologias, são realmente viáveis, e assim, possibilitam a sua adoção em projetos acadêmicos e pela indústria [Basili and Selby 1987]. Com base nas lacunas encontradas (**P1**, **P2** e **P3**), principalmente no que tange a incipiência de avaliações experimentais para o GV em LPSs baseadas em UML, um conjunto de estudos experimentais foram conduzidos para identificar a efetividade destas abordagens.

O indicador efetividade foi escolhido para avaliar um dos processos mais significativos e importantes para abordagens de gerenciamento de variabilidades baseadas em UML, que previamente mencionado, determina o sucesso na adoção de LPS. A efetividade é definida como o quão bem uma abordagem permite a identificação e a representação de variabilidades e seus elementos em modelos UML de LPSs, e foi proposta com base em estudos prévios [Basili and Selby 1987, Çöteli 2013]. Para o cálculo da efetividade, os participantes dos experimentos identificam os elementos variáveis em uma LPS, por meio da aplicação de uma abordagem de GV. Assim, são identificados e calculados o número de elementos variáveis corretamente e incorretamente modelados, por meio de oráculo para cada uma das LPSs utilizadas, bem como para as suas respectivas abordagens. Esses números são, então, aplicados na equação de efetividade:

$$\text{efetividade}(z) = \frac{nVarC - nVarI}{nVarI}$$

Na qual z é a abordagem de gerenciamento de variabilidade; $nVarC$ representa o número de variabilidades e seus elementos modelados corretamente, de acordo com a abordagem z ; e $nVarI$ é o número de variabilidade e seus elementos modelados incorretamente, de acordo com a abordagem z .

4.1. Definição dos Estudos Experimentais e Planejamento

O modelo GQM *Goal/Question/Metric* [Basili et al. 1986] foi adotado para a definição dos estudos experimentais. Em linhas gerais o objetivos de cada um dos quatro estudos convergem em **comparar** o método PLUS ou a abordagem de Ziadi et al. com a abordagem SMarty em suas diferentes versões, **com o propósito de** caracterizar o mais efetivo, **em relação à** capacidade de identificação e representação de variabilidades em modelos da UML em linhas de produto de software, **do ponto de vista de** arquitetos de linha de produto, **no contexto de** estudantes de graduação, pós-graduação e professores da área de engenharia de software de diferentes universidades como: Universidade Estadual de Maringá (UEM), da Universidade Tecnológica Federal do Paraná (UTFPR), Universidade Federal do Amazonas (UFAM), Instituto de Ciências e Matemáticas e de Computação da Universidade de São Paulo (ICMC-USP), Universidade Federal de São Carlos (UFS-Car), Universidade Federal do Paraná (UFPR) e Pontifícia Universidade Católica do Rio Grande do Sul (PUC-RS).

Para alcançar o objetivo definido (**OG**), duas questões de pesquisa foram definidas (Q.P): **Q.P.1** *Qual método/abordagem é mais efetiva na identificação e representação de variabilidades em diagramas de LPS baseadas em UML?* e **Q.P.2** *O conhecimento prévio dos participantes influenciaram no valor da efetividade obtidos para o método/abordagem?*

Os itens elencados e planejados para os estudos experimentais correspondem ao *contexto local*, no qual foram utilizados LPSs provenientes de estudos do Goma (e-commerce) e Ziadi et al. (*Banking*), bem como a LPS para fins pedagógicas *Arcade Game*

Maker (AGM) proposta pelo *Software Engineering Institute* (SEI) [SEI 2012]. Para a execução experimental, todos os estudos tiveram *treinamentos* aplicados. Estes treinamentos, com duração de até 60 minutos cada e realizados em ambiente acadêmico, além de explicarem conceitos básico de LPS, apresentaram cada uma das abordagens a serem utilizadas para modelar a LPS, entregues aleatoriamente a cada participante. Toda a instrumentação utilizada foi averiguada por meio de *projetos pilotos*, executados por especialistas na área de engenharia de software.

A *seleção dos participantes* ocorreu mediante grau de formação e com nível mínimo de conhecimento, não sendo selecionados aleatoriamente de uma população pelas dificuldades em obter-se voluntários. Contudo, a instrumentação utilizada pelos mesmos, assim como modelos das LPSs foram entregues de modo aleatório. Esta *instrumentação* é composta por um Termo de Adesão a Estudo Experimental, Questionário de Caracterização (aplicado também *online* e que permitiu quantificar em uma escala ordinal, o nível de conhecimento em LPS do participante), documento com Conceitos de LPS, da Abordagem X, da Abordagem Y, Elementos Gráficos da UML (explicando cada um dos elementos utilizados nos respectivos modelos da UML avaliados), Descrição Textual das LPSs e Formulários Experimentais (onde cada participante representava as variabilidades e similaridades, para posterior cálculo da efetividade).

A formulação das hipóteses para cada estudo experimental é expressa em dois conjuntos, um para a Q.P.1 e outro para a Q.P.2, em que X corresponde ao método (PLUS) ou abordagem (Ziadi et al.) comparada com Y que corresponde à abordagem SMarty.

Hipóteses para Q.P.1:

- **Hipótese Nula** ($H_{0,QP1}$): ambas as abordagens, X e Y, não existe diferença significativa de efetividade entre as abordagens para representar variabilidades em seus respectivos modelos UML ($H_{0,QP1} : \mu(\text{efetividade}(X)) = \mu(\text{efetividade}(Y))$);
- **Hipótese Alternativa** ($H_{1,QP1}$): A abordagem X é, em média, menos efetiva que a abordagem Y ($H_{1,QP1} : \mu(\text{efetividade}(X)) < \mu(\text{efetividade}(Y))$); e
- **Hipótese Alternativa** ($H_{2,QP1}$): A abordagem X é, em média, mais efetiva que a abordagem Y ($H_{2,QP1} : \mu(\text{efetividade}(X)) > \mu(\text{efetividade}(Y))$).

Hipóteses para Q.P.2:

- **Hipótese Nula** ($H_{0,QP2}$): a efetividade obtida pelas abordagens não sofreram influência do conhecimento prévio dos participantes ($H_{0,QP2} : \text{Corr}(\text{VarKnowledge} \text{ e } \text{VMApp}_X) = \text{Corr}(\text{VarKnowledge} \text{ e } \text{VMApp}_Y) = 0$);
- **Hipótese Alternativa** ($H_{1,QP2}$): A efetividade da abordagem Y é menos influenciada pelo conhecimento prévio dos participantes, que a abordagem X ($H_{1,QP2} : \text{Corr}(\text{VarKnowledge} \text{ e } \text{VMApp}_X) < \text{Corr}(\text{VarKnowledge} \text{ e } \text{VMApp}_Y)$); e
- **Hipótese Alternativa** ($H_{2,QP2}$): A efetividade da abordagem Y é mais influenciada pelo conhecimento prévio dos participantes, que a abordagem X ($H_{2,QP2} : \text{Corr}(\text{VarKnowledge} \text{ e } \text{VMApp}_X) > \text{Corr}(\text{VarKnowledge} \text{ e } \text{VMApp}_Y)$).

A Tabela 1 apresenta a descrição das variáveis independentes e dependentes, utilizadas para testar as hipóteses definidas.

De acordo com as variáveis (Tabela 1), para o experimento de caso de uso foi utilizado um fator aplicado a 24 participantes. Para os demais estudos foram aplicados

Tabela 1. Descrição das Variáveis Independentes e Dependentes

Nome da Variável	Tipo de Variável (Independente, dependente)	Abreviação	Classe (produto, processo, recurso ou método)	Entidade (instância de uma Classe)	Tipo de Atributo (interno, externo, ...)	Tipo de Escala (nominal, ordinal, ...)	Unidade	Escala	Regra de Mensuração
Abordagem ou Método de GV	Independente	VMAp	Método	GV baseado em UML	-	Nominal	-	SMarty 5.0, SMarty 5.1, PLUS e Ziadi et al.	-
Diagramas de LPSs		LPS	Produto	Diagramas da UML	-	Nominal	-	Casos de uso, classes e sequência.	-
Efetividade	Dependente	Effectiv	Processo	Variabilidades Corretas/Incorretas	Interna: efetividade, externa: qualidade	Ordinal	Número de variabilidades corretas menos as incorretas.	Qualquer inteiro.	Variabilidades corretas contam 1, incorretas -1.
Correlação entre conhecimento dos participantes e nível de conhecimento em variabilidade.		Corr (VarKnowledge e VMAp)	Processo	Conhecimento dos participantes em variabilidades e efetividade VMAp	Externa.	Ordinal	Escala de Spearman.	[-1.0, +1.0]	Fórmula de Spearman (p).

dois fatores para os 20 participaram no primeiro experimento de classes, 10 no segundo experimento e para os 20 participantes no experimento de sequência.

4.2. Procedimento de Execução e Análise

O procedimento padrão adotado para a execução experimental resume-se: (a) o participante é recebido no ambiente onde o estudo será conduzido; (b) o coordenador do estudo entrega ao participante o termo de adesão, os documentos sobre conceitos de GV em LPSs e a descrição das LPSs a serem modeladas em ordem aleatória (*e-commerce* para modelos de casos de uso, *AGM* e *e-commerce* para classes (primeiro e segundo estudos) e *AGM* e *Banking* para sequência); (c) os participantes leem os documentos recebidos; (d) o coordenador explica os documentos; (e) o coordenador, aleatoriamente, determina uma abordagem para cada participante; (f) o coordenador treina os participantes nas suas respectivas abordagens; (g) os participantes sanam possíveis dúvidas sobre a abordagem; (h) os participantes identificam e representam as variabilidades nos respectivos diagramas da UML, de acordo com sua abordagem; e como último procedimento (i) os participantes entregam ao coordenador a instrumentação e formulários e são liberados do ambiente de execução (sem nenhum bônus pela participação, além do conhecimento adquirido).

A Figura 4 resume as atividades que foram conduzidas no procedimento de análise dos resultados. Uma vez finalizada as sessões experimentais, o coordenador prepara os dados coletados para o cálculo da efetividade para cada uma das amostras das abordagens aplicadas (*VMAp Effectiv*). Na próxima etapa, a efetividade calculada é correlacionada com o nível de conhecimento de cada participante (*VarKnowledge*) por meio da correlação de *Spearman* para cada estudo, respondendo a **Q.P.2**. Os mesmos valores de efetividade são aplicados ao teste de normalidade Shapiro-Wilk [Royston 1982], para definir qual teste de hipótese (se paramétrico ou não-paramétrico) deverá ser submetido, permitindo responder a **Q.P.1**. O teste paramétrico corresponde ao teste T e o teste não-paramétrico ao teste Mann-Whitney-Wilcoxon.

4.3. Análise Experimental

A Tabela 2 apresenta o resumo dos resultados para cada um dos estudos executados. No experimento de casos de uso (**R3**) [Marcolino et al. 2013], são obtidas evidências

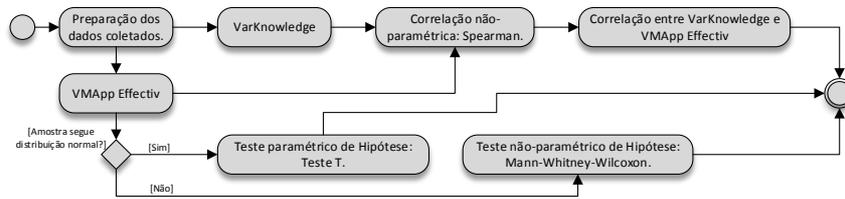


Figura 4. Diagrama de Atividades para a Fase de Análise.

da maior efetividade para a abordagem SMarty 5.0 nos modelos de caso de uso da UML (Q.P.1) e é identificada uma menor correlação para SMarty, entre o nível de conhecimento prévio dos participantes e a efetividade coletada (Q.P.2). O experimento de sequência (R7) [Marcolino et al. 2014a] também apresentou evidências da efetividade para a abordagem SMarty 5.0, porém como uma correlação maior para a abordagem SMarty 5.0 e menor para a abordagem de Ziadi et al.

Tabela 2. Resumo dos Resultados dos Estudos Experimentais.

Diagrama da UML		Caso de Uso		Sequência		Classes (1º Estudo)		Classes (2º Estudo)	
VMAApp		PLUS	SMarty 5.0	Ziadi et al.	SMarty 5.0	PLUS	SMarty 5.0	PLUS	SMarty 5.1
Q.P. 1	Effectiv (VMAApp)	1,75	4,33	8,57	24,86	23	13,6	26,4	16,2
		SMarty é mais efetiva.		SMarty é mais efetiva.		PLUS é mais efetiva.		PLUS é mais efetiva.	
	Rejeitar $H_{0,OP1}$? / Hipótese Aceita?	Sim / $H_{1,OP1}$		Sim / $H_{1,OP1}$		Sim / $H_{2,OP1}$		Sim / $H_{2,OP1}$	
Q.P. 2	Corr(VarKnowledge e, VMAApp)	0,61	-0,25	-0,07	0,61	0,53	0,18	0,22	0,29
		Positiva forte	Negativa Fraca	Negativa Fraca	Positiva Forte	Positiva Forte	Positiva Fraca	Positiva Fraca	Positiva Fraca
	Rejeitar $H_{0,OP2}$? / Hipótese Aceita?	Sim / $H_{2,OP2}$		Sim / $H_{1,OP2}$		Sim / $H_{2,OP2}$		Sim / $H_{1,OP2}$	

Dois experimentos de classe foram conduzidos (R4 e R6) [Marcolino et al. 2014b]. O primeiro apresentou maior efetividade para o método PLUS e uma maior correlação da efetividade com o nível de conhecimento prévio, também para PLUS. Com base nas evidências identificadas, melhorias na abordagem SMarty 5.0 foram identificadas e aplicadas, evoluindo a abordagem para a sua versão 5.1. A evolução ocorreu em duas diretrizes de classes, melhorando o entendimento das mesmas e auxiliando na aplicação e interpretação da abordagem SMarty para tais modelos. Após a evolução SMarty 5.1 um novo experimento de classes foi conduzido. Neste segundo experimento, PLUS novamente foi evidenciado como mais efetivo, apresentando uma efetividade maior que no primeiro experimento, contudo SMarty 5.1 também apresentou melhorias na efetividade. Nova análise foi conduzida para averiguar o motivo de tais evidências e notou-se uma situação peculiar no método PLUS.

O método apresenta apenas dois estereótipos em modelos de classes, um para os elementos comuns (<<kernel>>) e outro para elementos variantes (<<optional>>), o que torna a aplicação e identificação das variabilidades trivial, elevando a sua efetividade. Contudo, o processo de classificação e identificação das classes facilitado não significa, porém a construção de produtos consistentes. Arquitetos de LPS podem, ao se deparar com o estereótipo <<optional>> resolver um ponto de variação de forma diferente, ou mesmo, confundir um ponto de variação com uma variante. Como tal estereótipo representa restrições entre variantes do tipo mutuamente exclusiva (XOR), inclusiva (OR) e

também opcional, a construção de um produto de forma consistente, torna-se mais difícil de ser executada. Esse exemplo é apresentado na Figura 5.

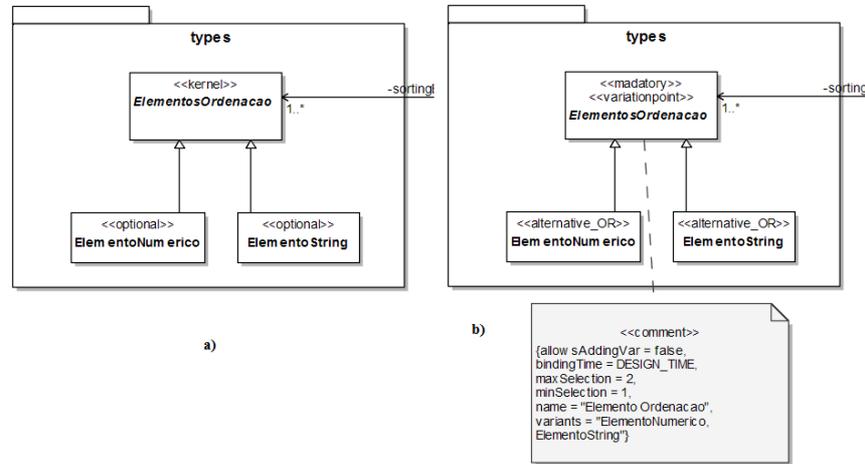


Figura 5. Excerto de Modelo de Classes de uma LPS representada com Método PLUS (a) e SMarty (b)

Na Figura 5 b) é apresentado um exemplo de aplicação dos estereótipos da abordagem SMarty 5.1. Nela é identificado o ponto de variação e sua obrigatoriedade, bem como duas variantes com restrição inclusiva (OR). Já a Figura 5 a) apresenta a mesma variabilidade, com os estereótipos do método PLUS. A restrição entre variantes opcionais indica que a variante pode ser incluída ou não no produto derivado, assim, o uso de tal estereótipo pode levar a um entendimento incorreto, pois no PLUS, tal elemento indica, não apenas opcionalidade, mas restrições inclusivas ou mutuamente exclusivas.

4.4. Avaliação de Validade do Conjunto de Estudos Experimentais

A Tabela 3 apresenta as principais ameaças identificadas no decorrer dos estudos experimentais. Os pacotes experimentais estão disponibilizados² permitindo a realização de replicações, bem como para a execução de novas avaliações.

Tabela 3. Principais Ameaças a Validade e Soluções.

Tipo de Ameaça	Descrição	Medida de Mitigação/Solução
Validade de Conclusão	Tamanho das amostras.	Replicações futuras.
Validade de Construção	Escala aplicada ao nível de conhecimento dos participantes, para condução da correlação.	Identificação, para trabalhos futuros, de outras escalas, ou métodos de mensuração de nível de conhecimento, para a correlação.
Validade Interna	Diferença de conhecimento entre participantes.	Sessões de treinamento.
	Efeitos de Fadiga.	Os experimentos tiveram duração média de 100 minutos, para casos de uso; 80 minutos, para classes (primeira e segunda execução); e 20 minutos, para o estudo experimental de sequência. Assim, a fadiga não foi considerada relevante.
	Influência entre participantes.	Supervisão de um observador humano (coordenador).
Validade Externa	Não utilização de LPSs da Indústria.	Replicações futuras com LPSs reais.
	Não participação de profissionais da Indústria.	Replicações futuras com profissionais.

5. Consideração Finais e Trabalhos Futuros

A abordagem de LPS tem destaque, entre as diversas abordagens de reuso de software, pois mostra-se efetiva no que tange reutilização de artefatos de modo não oportunista.

²<http://www.din.uem.br/~smarty/trab1.html>

LPS permite englobar outras contribuições no processo de desenvolvimento, como melhora na qualidade dos artefatos e produtos finais. Contudo, seus benefícios são obtidos quando ocorre o gerenciamento de variabilidades de modo conciso e sistematizado. A importância das variabilidades torna a atividade de gerenciamento uma das mais importantes e, assim, GV é foco de diversas propostas, com destaque para as baseadas em UML.

A abordagem baseada em UML, SMarty, mostra-se ampla, tanto por adotar à UML 2.0 na definição de um perfil, o *SMartyProfile*, quanto por seu processo sistematizado, o *SMartyProcess* que, por meio das diretrizes definidas, auxiliam a aplicação do seu perfil, tanto para reconhecer variabilidades em modelos UML ainda não avaliados, quanto para representar as já existentes.

A necessidade de realizar a avaliação experimental de SMarty, se deve: (i) pela lacuna existente no que diz respeito a avaliação experimental de abordagens propostas na literatura, identificada por meio de levantamento bibliográfico e de mapeamento sistemático; (ii) a necessidade de avaliação dos modelos apoiados por SMarty e seus recursos (perfil e processo), que por meio dos experimentos puderam ser analisados e evoluídos; e (iii) a necessidade de consolidá-la para possibilitar sua adoção na indústria e em maior grau na academia.

Após a extensão de SMarty 4.0 para SMarty 5.0, possibilitando a identificação e representação de variabilidades de comportamento dinâmico, por meio de modelos de sequência da UML, um conjunto de estudos experimentais para os três dos modelos mais representativos de SMarty: casos de uso, classes e sequência, foram conduzidos. Os resultados apresentaram evidências da aplicação de SMarty para identificar e representar variabilidades em casos de uso e sequência, e a efetividade de PLUS para modelos de classes, que levou à melhorias para SMarty e uma nova versão, a 5.1. Assim, SMarty 5.1 se mostra efetiva para gerir variabilidades, aumentando o nível de qualidade e garantia de geração de produtos específicos em LPSs.

As evidências iniciais quanto a efetividade de SMarty permitem que novas pesquisas possam ser realizadas, tais como a avaliação do impacto do uso da abordagem pela comunidade, bem como promovendo sua melhoria sob diferentes aspectos [Geraldi et al. 2015, Bera et al. 2015, Falvo et al. 2014, Falvo Júnior et al. 2014]. Ainda que a realização dos experimentos tenha adotado modelos de LPSs com nível mediano de complexidade e participantes de nível acadêmico, devido à restrições de tempo e disponibilidade de profissionais da indústria, os resultados permitiram uma análise mais aprofundada e formal dos diferentes mecanismos que as abordagens possuem. Adicionalmente, por meio de tais análises, pode-se realizar uma adoção de abordagens baseado em evidências, preferível a uma seleção *ad hoc*.

Destarte, somas-se aos resultados e contribuições apresentados (Seção 3, Figura 1) as publicações dos processos experimentais conduzidos, que tornam possíveis a realização de avaliações de outras abordagens de gerenciamento de variabilidade presentes na literatura, preenchendo a lacuna de pesquisa existente (**P3**) [Marcolino et al. 2014a, Marcolino et al. 2014b, Marcolino et al. 2013].

Finalmente, vantagens trazidas pelas abordagens baseadas em UML, em especial a SMarty, como a possibilidade de identificar novas variantes em modelos UML, vai de encontro com a busca por novas variações de produtos nas indústrias e a ampliação de

seus portfólios. Deste modo, a adoção de SMarty torna possível a evolução das LPSs, ampliando a quantidade de configurações de produtos instanciáveis e comercializáveis. Além desta vantagem, a adoção de uma abordagem com um perfil UML propicia o apoio de ferramentas UML, que auxiliam na verificação das configurações da LPS com o uso de regras formais, como a *Object Constraint Language* (OCL) e, assim, o que antes era realizado manualmente passa a ser automatizado.

Como trabalhos em andamento, novas avaliações experimentais estão sendo realizadas, como a aplicação de abordagens de GV baseadas em UML para avaliar a efetividade na interpretação de modelos de classes de duas LPSs, bem como a geração de configurações e produtos por meio de modelos desenvolvidos com tais abordagens. Além disso, SMarty será adotada para a concepção de uma LPS para o desenvolvimento de aplicações móveis educacionais.

Referências

- Basili, V. and Selby, R. (1987). Comparing the Effectiveness of Software Testing Strategies. *IEEE Trans. Software Eng.*, SE-13(12):1278–1296.
- Basili, V. R., Selby, R. W., and Hutchens, D. H. (1986). Experimentation in Software Engineering. *IEEE Trans. Software Eng.*, 12(7):733–743.
- Bera, M. H. G., Oliveira Jr, E., and Colanzi, T. E. (2015). Evidence-based smarty support for variability identification and representation in component models. *Proceedings of the Internat. Conf. on Enterprise Information Systems*, v. 2.:p. 295–302.
- Capilla, R., Bosch, J., and Kang, K. (2013). *Systems and Software Variability Management: Concepts, Tools and Experiences*. SpringerLink : Bücher. Springer.
- Çöteli, M. B. (2013). Testing Effectiveness and Effort in Software Product Lines a Thesis. Master's thesis, Middle East Technical University.
- Chen, L., Ali Babar, M., and Ali, N. (2009). Variability Management in Software Product Lines: a Systematic Review. In *Proc. Int. Software Product Line Conference*, pages 81–90, Pittsburgh, PA, USA. Carnegie Mellon University.
- Falvo, V., Duarte Filho, N., Oliveira Jr, E., E., and Barbosa, E. (2014). A Contribution to the Adoption of Software Product Lines in the Development of Mobile Learning Applications. In *Frontiers in Education Conference (FIE), 2014 IEEE*, pages 1–8.
- Falvo Júnior, V., Duarte Filho, N. F., Oliveira Jr, E., and Barbosa, E. F. (2014). Towards the establishment of a software product line for mobile learning applications. *International Conference on Software Engineering and Knowledge Engineering (SEKE)*, 1:678–683.
- Fiori, D. R., Gimenes, I. M., Maldonado, J. C., and Oliveira Jr, E. (2012). Variability Management in Software Product Line Activity Diagrams. In *Proc. Int. Conf. on Distributed Multimedia Systems*, pages 89–94.
- Foschiani, F. Y. S., Tizzei, L. P., and Rubira, C. M. F. (2013). A SPL Infrastructure for Supporting Scientific Experiments in Petroleum Reservoir Research Field. *Simp. Brasileiro de Comp., Arquitetura e Reutilização de Software, Brasília.*, 1:85–94.
- Galster, M., Weyns, D., Tofan, D., Michalik, B., and Avgeriou, P. (2013). Variability in software systems - a systematic literature review. *IEEE Transactions on Software Engineering*, 99(Preliminary):1.

- Geraldi, R. T., Oliveira Jr, E., Conte, T. U., and Steinmacher, I. F. a. . (2015). Checklist-based inspection of smarty variability models: Proposal and empirical feasibility study. *Proceedings of the Int. Conf. on Enterprise Information Systems*, v. 2.:p. 268–275.
- Gomaa, H. (2004). *Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures*. Addison Wesley, Redwood City, CA, USA.
- Jacobson, I., Griss, M., and Jonsson, P. (1997). *Software Reuse: Architecture, Process and Organization for Business Success*.
- Kang, K. C., Cohen, S. G., Hess, J. A., Novak, W. E., and Peterson, A. S. (1990). Feature-Oriented Domain Analysis (FODA) Feasibility Study. Technical report, Carnegie-Mellon University Software Engineering Institute.
- Linden, F. J. v. d., Schmid, K., and Rommes, E. (2007). *Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering*. Springer-Verlag New York.
- Mafra, S. N. and Travassos, G. H. (2005). Estudos Primários e Secundários Apoiando a Busca por Evidência em Engenharia de Software. Technical report, RT-ES 687/06.
- Marcolino, A., Oliveira Jr, E., and Gimenes, I. M. (2014a). Variability identification and representation in software product line UML sequence diagrams: Proposal and empirical study. In *Brazilian Symp. on Soft.Eng.*, pages 141–150.
- Marcolino, A., Oliveira Jr, E., Gimenes, I. M., and Barbosa, E. F. (2014b). Empirically based evolution of a variability management approach at UML class level. In *Annual Comp. Soft. and Applications Conf. COMPSAC*, pages 354–363.
- Marcolino, A., Oliveira Jr, E., Gimenes, I. M., and Maldonado, J. C. (2013). Towards the Effectiveness of a Variability Management Approach at Use Case Level. In *Proc. of the Int. Conf. on Soft. Eng. & Knowledge Eng.*, pages 214–219.
- Oliveira Jr, E., Gimenes, I. M., and Maldonado, J. C. (2010). Systematic Management of Variability in UML-based Software Product Lines. *J. Universal Computer Science*, 16(17):2374–2393.
- Oliveira Jr, E., Gimenes, I. M., and Maldonado, J. C. (2013). Systematic Evaluation of Software Product Line Architectures. *J. of Universal Comp. Science*, pages 25–52.
- OMG (2011). UML, Superstructure Version 2.4.1. Disponível em <http://www.omg.org/spec/UML/2.4.1/Superstructure> Acessado em 12 de Jun. 2012.
- Pohl, K., Böckle, G., and Linden, F. J. v. d. (2005). *Software Product Line Engineering: Foundations, Principles, and Techniques*. Springer-Verlag, Secaucus, NJ, USA.
- Royston, J. P. (1982). An Extension of Shapiro and Wilk's W Test for Normality to Large Samples. *Journal of the Royal Statistical Society. Series C*, 31(2):115–124.
- SEI (2012). Software Engineering Institute A Framework for Software Product Line Practice - Version 5.0. Disponível em http://www.sei.cmu.edu/productlines/frame_report/index.html Acessado em 20 de Jul. 2012.
- Weiss, D. M. and Lai, C. T. R. (1999). *Software Product-line Engineering: a Family-based Software Development Process*. Addison-Wesley Longman Publishing Co.
- Ziadi, T. and Jezequel, J.-M. (2006). Software Product Line Engineering with the UML: Deriving Products. In *Software Product Lines*, pages 557–588. Springer Berlin.