

## **APOIO COMPUTACIONAL AO GERENCIAMENTO DE VARIABILIDADES COM CVL EM LINHA DE PRODUTO DE SOFTWARE**

Lailson Miranda Tofanelli (PIBIC/CNPq-FA-UEM), Alisson Chiquitto (Co-autor), Edson Alves de Oliveira Junior (Orientador), Itana Gimenes (Co-orientador), e-mail: edson@din.uem.br, e-mail: itanagimenes@gmail.com.

Universidade Estadual de Maringá/Departamento de Informática/Maringá, PR.

**Área de Conhecimento: 1.00.00.00-3 Ciências Exatas e da Terra, 1.03.00.00-7 subárea: Ciência da Computação, 1.03.03.02-2 subárea: Engenharia de Software.**

**Palavras-chave:** Linha de Produto de Software, Gerenciamento de Variabilidade, CVL.

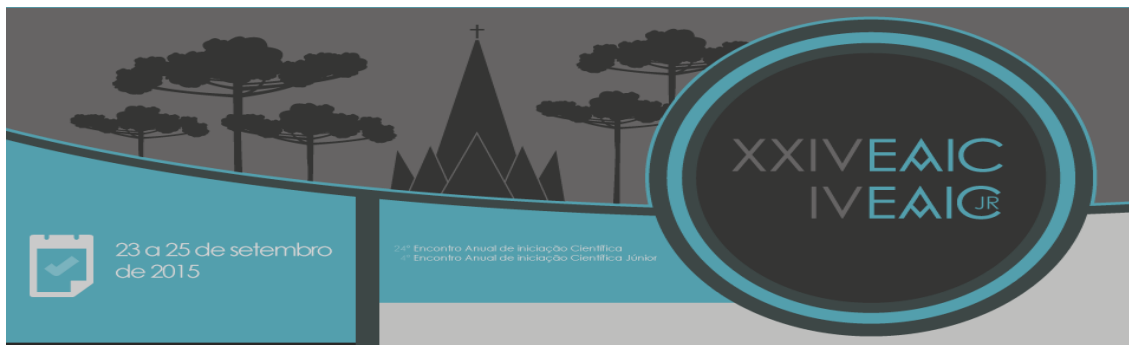
### **Resumo:**

Gerenciamento de variabilidade é uma das áreas-chave da abordagem de Linha de Produto de Software. Existem algumas linguagens para a representação e gerenciamento de variabilidades em LPS, entre elas a linguagem *Common Variability Language* (CVL), que é alvo deste projeto. Este projeto analisou ferramentas de apoio ao gerenciamento com CVL. Também analisou, ferramentas de apoio a especificação do Modelo Base. Com base nos estudos realizados, foi feita uma proposta de uma nova ferramenta para criar um mapeamento entre o Modelo de Variabilidade da CVL e o Modelo Base, realizando assim o processo de Materialização.

### **Introdução**

Gerenciamento de variabilidades é uma das principais etapas em um processo de desenvolvimento de uma LPS, nessa etapa ocorre a evolução da LPS. Uma aplicação pode ser superior a várias outras no mercado se a fase de gerenciamento for bem sucedida. Escolher onde uma LPS terá um ponto de variação não é complicado, mas definir quais serão as variabilidades não é simples, por isso um bom gerenciamento das variantes é tarefa obrigatória para obter uma LPS consistente.

O gerenciamento de variabilidades em uma LPS conta com diversas abordagens, todas elas tem seu diferencial, mas uma que tem chamado bastante atenção é a *Common Variability Language* (CVL), o gerenciamento de variabilidades com ela se mostra simples e fácil, pois o modelo de



variabilidades construído na linguagem se remete a uma árvore, objeto que é bastante estudado na academia.

Algumas metodologias definidas no desenvolvimento de uma LPS, utilizam geralmente uma abordagem para o gerenciamento de variabilidades, e uma abordagem para o Modelo Base. É o caso da metodologia *SyMPLES* (Silva, 2012), que utiliza a abordagem *SMarty* para o gerenciamento das variabilidades e *SysML* para o Modelo Base. Este projeto poderia vir a auxiliar outros projetos cuja proposta seria substituir o *SMarty* por CVL.

## Materiais e Métodos

O trabalho foi dividido em três etapas, a primeira consistia na fundamentação teórica dos conceitos relevantes para o projeto, sendo eles:

- **Linha de Produto de Software** - conjunto de sistemas de software com características (*features*) comuns que definem um produto, com o objetivo de atingir um mercado distinto baseado na similaridade entre os sistemas (Marcolino, 2014).
- **Variabilidade** - está relacionada com a atividade de personalizar um sistema de software, configurando as características variáveis dos produtos pertencentes a uma determinada família.
- **Gerenciamento de Variabilidade** - atividade que lida com a variação do sistema ao longo do tempo e espaço. Composto por quatro sub-atividades: Identificar, Delimitar, Implementar, e representar as variabilidades e variantes (Marcolino, 2014).
- **Linguagem CVL** - linguagem independente de domínio para especificar e resolver variabilidades (HAUGEN et al., 2008).
- **Abordagem SMarty** - abordagem que permite expressar os conceitos de variabilidade em *UML* (OLIVEIRA JUNIOR, 2010).
- **Linguagem SysML** - linguagem de modelagem gráfica de uso geral para especificar, analisar, projetar e verificar sistemas embarcados (Silva, 2012).

A segunda etapa do projeto foi a análise das ferramentas. Foi utilizado como modelo de avaliação os parâmetros definidos por Aguiar (2008). Os parâmetros consistiam em custo da ferramenta, como ela foi implementada, como os arquivos gerados eram, eficiência, se a ferramenta satisfaz as necessidades de usuário, entre outros parâmetros. Três ferramentas foram selecionadas para avaliação, sendo duas próprias para o gerenciamento das variabilidades por meio da linguagem CVL, que são CVL 2 Tool e CT-CVL. A última foi a ferramenta Papyrus que é voltada a especificação de modelos UML.

A terceira etapa do projeto consistiu da implementação de uma ferramenta, para realizar a Materialização, definida nos conceitos de CVL. A



Materialização é o processo de transformação de um modelo base em um modelo de produto por meio da resolução dos pontos de variação (HAUGEN et al., 2008).

## Resultados e Discussão

A ferramenta **CVL 2 Tool (SINTEF)** visa o gerenciamento de variabilidades em LPS. Ela é um protótipo para o padrão CVL 2. A ferramenta ainda está em fase de desenvolvimento, ela é liberada apenas como uma demo, mas todas suas funcionalidades já vem configuradas na aplicação e prontas para uso. A ferramenta é *open source*. Criada em linguagem de programação *Java*, ela é apenas um arquivo em *.jar*, assim necessita da máquina virtual *Java* (*JVM*) instalada no computador. Simples e fácil de usar, porém algumas atividades simples são lentas tornando assim um tempo de resposta ruim, algo que deverá ser melhorado quando a ferramenta estiver concluída.

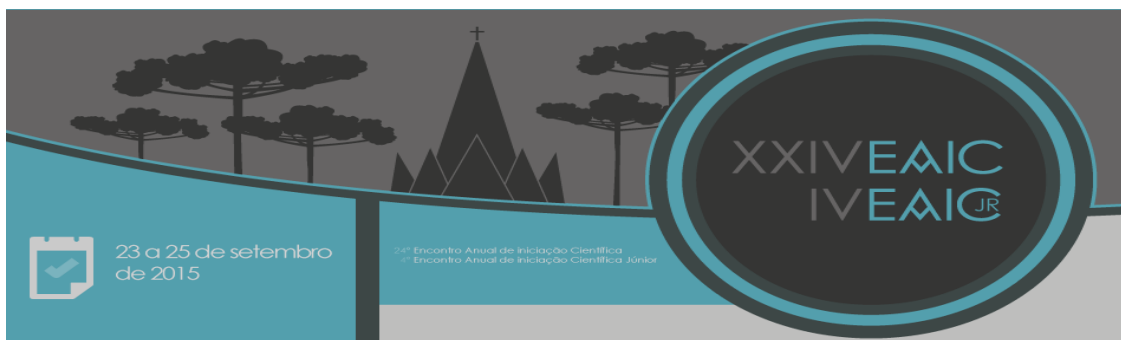
O **CT-CVL (SINTEF)**, ferramenta de configuração com CVL, é um *plugin* do Eclipse para a criação de modelos de variabilidades usando CVL. A aplicação é *open source*. A documentação é pequena e simples, assim é fácil entender seu funcionamento. A documentação inclui instruções de instalação e gerenciamento de variabilidades de LPS. Construída também na linguagem de programação *Java*, porém requer o ambiente *Eclipse*. A ferramenta se mostra rápida, embora alguns problemas quanto a resolução das variabilidades foram encontrados, em alguns momentos não há possibilidade de obter uma resolução.

**Papyrus (All4Tec)** é uma ferramenta de modelagem, apoia *UML* e linguagens relacionadas, como por exemplo *SysML*, que oferece suporte completo, na qual inclui todos os editores gráficos específicos necessários. A ferramenta é *open source*, tem documentação ampla, contando com tutorias, vídeos, exemplos, e tudo isso pode ser obtido diretamente da internet, o que torna a ferramenta fácil de aprender e utilizar. É um *plugin* do *Eclipse*, basta adicionar por meio do próprio *Eclipse*. A ferramenta se mostra muito útil ao desenvolver algum modelo.

A ferramenta construída é capaz de gerar um modelo de produto, baseado na resolução do modelo de variabilidades. É necessário os arquivos gerados nas ferramentas analisadas, o modelo de variabilidades e modelo de resolução são definidos nas aplicações com suporte ao CVL, o modelo base é definido na ferramenta Papyrus.

## Conclusões

A fase de representação de variabilidades deve ser feito com todo o cuidado, pois é nessa etapa que tem a garantia do processo e a evolução da LPS.



Contudo, escolher ferramentas para apoiar o gerenciamento é uma tarefa complicada, existe um grande estoque de ferramentas disponíveis para ajudar o processo, algumas delas foram apresentadas nesse projeto, que se mostraram úteis para a atividade que são destinadas, o uso é simples em algumas funcionalidades, e principalmente são *open source*, o que permite qualquer indivíduo ou indústria obter as ferramentas sem o gasto de valores.

## Agradecimentos

Agradeço ao Programa de Iniciação Científica CNPq-FA-UEM por incentivar e financiar a realização deste trabalho.

## Referências

HAUGEN, Øystein. **Common Variability Language (CVL)**. Disponível em: <<http://www.omgwiki.org/variability/doku.php>>. Acesso em: 15 jun. 2015.

LIMA JÚNIOR, R. Aguiar. **COMPARAÇÃO ENTRE FERRAMENTAS PARA LINHA DE PRODUTOS DE SOFTWARE**. 2008. 43 f. TCC (Graduação) - Curso de Engenharia da Computação, Escola Politécnica de Pernambuco, Recife, 2008.

MARCOLINO, A. M. **Avaliação experimental da abordagem SMarty para gerenciamento de variabilidade em linhas de produto de software baseadas em UML**. 2014. 165 f. Dissertação (Mestrado) – Programa de Pós-graduação em Ciência da Computação, Universidade Estadual de Maringá, Maringá, 2014.

OLIVEIRA JUNIOR, E. A. **System-PLA: um método sistemático para avaliação de arquitetura de linha de produto de software baseada em UML**. 2010. 305 f. Tese (Doutorado) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2010.

SILVA, R. F. **SyMPLES: Uma Abordagem de Desenvolvimento de Linha de Produto para Sistemas Embarcados baseada em SysML**. 2012. 110 f. Dissertação (Mestrado) – Programa de Pós-graduação em Ciência da Computação, Universidade Estadual de Maringá, Maringá, 2012.