

Configuração de Produtos em Linha de Produtos de Software

Juliana Alves Pereira, Eduardo Figueiredo

Programa de Pós-Graduação em Ciências da Computação (PPGCC)
Laboratório de Engenharia de Software (LabSoft), Departamento de Ciência da
Computação (DCC), Universidade Federal do Minas Gerais (UFMG)
Belo Horizonte - MG – Brasil

{juliana.pereira, figueiredo}@dcc.ufmg.br

Resumo. *Para melhorar a qualidade do software e reduzir os custos e prazos, muitas empresas estão adotando a abordagem de Linha de Produtos de Software (LPS). Um processo chave em LPS é a configuração de produtos, que é o processo de customização dos principais ativos de uma LPS. A fim de apoiar as empresas durante estas customizações, métodos e ferramentas de gerência de variabilidade e configuração de produtos têm sido desenvolvidos. Entretanto, através de um estudo exploratório foi identificado que os métodos e ferramentas existentes não são suficientes. Orientação e apoio são necessários para aumentar a eficiência das empresas ao lidar com milhares de combinações de ativos possíveis em uma LPS. Neste trabalho propomos um modelo, baseada em técnicas de busca e otimização, para apoiar as empresas na configuração semi-automática de produtos. O principal objetivo deste modelo é maximizar a satisfação de clientes e auxiliar empresas desenvolvedoras de LPS, em busca da configuração do produto que traga melhor custo-benefício para o cliente. O modelo proposto está sendo automatizado por uma ferramenta e avaliado em estudos experimentais.*

Palavras-Chave: *Linha de Produtos de Software, Gerência de Variabilidade, Configuração de Produtos.*

1. Introdução

A crescente necessidade de desenvolvimento de sistemas de software maiores e mais complexos exige um melhor suporte para artefatos reutilizáveis [7]. Muitas técnicas, como Linha de Produtos de Software (LPS), têm sido propostas para oferecer avançado suporte a reutilização de software [10]. LPS explora o fato de sistemas em um mesmo domínio serem semelhantes e terem potencial para reutilização [7]. Considerando que tem se tornando comum a comercialização de produtos que se diferenciam por variações em suas características, grandes empresas tais como: Boeing, Hewlett Packard, Nokia e Siemens; têm investido em LPS [8]. LPS permite às empresas rápida entrada no mercado por fornecer capacidade de reutilização em larga escala com customização em massa [6]. Com esse enfoque, promete-se obter ganhos na produtividade, produtos mais confiáveis e preço mais acessível [4]. Esses fatores contribuem para um aumento significativo na satisfação do cliente.

Um elemento de fundamental problema em LPS corresponde ao tamanho das LPS industriais. LPS industriais podem facilmente incorporar milhões de características variáveis. A complexidade causada por essa quantidade de variabilidade torna o

gerenciamento de variabilidade e as tarefas de derivação de produtos extremamente difícil [4]. A fim de apoiar as empresas na gerência de variabilidade e configuração de produtos, novos métodos e ferramentas têm sido desenvolvidos [10]. Entretanto, através de um estudo exploratório foi identificado que os métodos e ferramentas existentes não são suficientes para apoiar as empresas durante a configuração de produtos. Orientação e apoio são necessários para aumentar a eficiência das empresas ao lidar com milhares de combinações possíveis em uma LPS. A configuração de produtos em LPS é uma tarefa cuja complexidade aumenta exponencialmente à medida que aumenta o número de características envolvidas e restrições a serem consideradas [7]. Para abordar este problema, apresentamos um modelo para otimizar a variabilidade fornecida em LPS.

O restante do artigo está organizado da seguinte forma. A Seção 2 introduz os conceitos fundamentais de variabilidade em LPS. Na Seção 3 são detalhadas as contribuições deste trabalho. A Seção 4 apresenta métodos e ferramentas disponíveis para gerência de variabilidade e configuração de produtos em LPS. A Seção 5 descreve o problema e o modelo proposto. A Seção 6 descreve o estado atual do trabalho. Por fim, os trabalhos relacionados são discutidos na Seção 7.

2. Variabilidade em LPS

A variabilidade em uma LPS é dada pela seleção de componentes específicos que adicionam uma ou mais características aos produtos [6]. Características comuns a um domínio formam o núcleo e outras características definem pontos de variação [7]. Uma vez que o domínio do sistema é decomposto em características independentes e facilmente combináveis, diferentes versões de um sistema são possíveis mediante a inclusão ou exclusão por combinação de determinadas características [4]. Por exemplo, os automóveis de um mesmo modelo se diferenciam por itens como airbag e som. Existem características comuns a todos os produtos e outras características podem variar de um produto para outro.

Entretanto, um desafio predominante em LPS é gerenciar a variabilidade e prover ao cliente uma configuração de produto com melhor custo-benefício. LPS industriais podem facilmente incorporar milhares de produtos, devido à evolução contínua da LPS, resultando em uma explosão combinatória de variantes [4]. Por exemplo, a partir de uma LPS que oferece apenas 20 características variáveis, é teoricamente possível derivar 1.048.576 produtos. Este simples exemplo mostra que, mesmo um pequeno número de características variáveis resulta numa explosão combinatória de variantes.

3. Contribuições

Este trabalho tem como principais contribuições um estudo exploratório de métodos e ferramentas existentes que dão apoio ao gerenciamento de variabilidades em LPS. Este estudo permitiu: comparar funcionalidades oferecidas pelas ferramentas, identificar redundâncias e funcionalidades recorrentes em abordagens para LPS, e destacar melhorias que possam ser realizadas em ferramentas existentes. Baseando-se neste estudo, um modelo para apoiar as empresas na configuração semi-automática de produtos é proposto. Este modelo considera aspectos como (i) preço de desenvolvimento das características para a empresa desenvolvedora da LPS; (ii) orçamento que o cliente possui para aquisição do software; e (iii) grau de importância das características para o cliente. O principal objetivo é apoiar as empresas durante a

otimização da configuração das características, em busca da configuração de produto com melhor custo-benefício para o cliente.

4. Métodos e Ferramentas para Gerenciamento de LPS

Métodos e ferramentas, tais como, SPLOT¹, FeatureIDE², XFeature³, fmp⁴ e pure::variants⁵ são utilizados para apoiar a gerência de variabilidade e configuração de produtos em LPS. Grandes empresas de software estão adotando estas ferramentas para desenvolver seus produtos [8][10]. Devido à quantidade de métodos e ferramentas disponíveis, este trabalho realiza um estudo para identificar diferenças, redundâncias e limitações destas ferramentas. Inicialmente realizamos um estudo exploratório utilizando as ferramentas SPLOT [5] e FeatureIDE [11].

Este estudo inicial envolveu 56 estudantes que realizaram um curso avançado em Engenharia de Software no ano de 2012. Pedimos aos participantes para executar três tarefas usando SPLOT ou FeatureIDE. As tarefas incluem o uso de funcionalidades para (i) criar e editar um modelo de características; (ii) analisar automaticamente o modelo de características criado e observar as estatísticas provida pela ferramenta; e (iii) configurar um produto usando a funcionalidade de configuração de produtos. Finalmente, pedimos aos participantes para responderem duas perguntas simples sobre o que eles gostaram e o que não gostaram na ferramenta utilizada.

Em particular, o ponto que chamou a atenção foi à dificuldade e a falta de auxílio destas ferramentas durante a configuração de produtos. Aproximadamente 70% dos participantes realizaram algum comentário sobre isso. A partir desta observação, propomos um modelo que está sendo implementado como uma extensão da ferramenta FeatureIDE. Além de ser bastante utilizado, FeatureIDE foi escolhido por sua capacidade de suportar técnicas modernas de programação, tais como programação orientada a aspectos [3] e programação orientada a características [1]. Adicionalmente, a ferramenta está integrada a várias técnicas de composição, tais como AHEAD, FeatureHouse, Feature C++, AspectJ, Antenna e CIDE.

5. Modelo Computacional Proposto

À medida que o tamanho e a complexidade dos sistemas aumentam, surge a necessidade da adoção de um paradigma de desenvolvimento baseado em LPS. Entretanto, conforme estudo apresentado anteriormente, apesar de diversas ferramentas apoiarem a configuração de produtos, nenhuma das ferramentas avaliadas oferece um método de customização de produtos para apoiar as empresas durante a configuração e recomendação de produtos ao cliente.

Como solução, este trabalho propõe um modelo de configuração de produtos em LPS que satisfaça restrições impostas por clientes que irão adquirir o produto. Como representado pela Figura 1, o método proposto prevê que o produto seja negociado em função das características da linha e do orçamento disponível pelo cliente. Assim, o

¹ <http://www.splot-research.org/>

² http://www.witi.cs.uni-magdeburg.de/iti_db/research/featureide/

³ <http://www.pnp-software.com/XFeature/>

⁴ <http://gsd.uwaterloo.ca/fmp/>

⁵ <http://www.pure-systems.com/>

objetivo é otimizar a configuração de características e apoiar as empresas na configuração de produtos que maximizem a satisfação dos clientes.

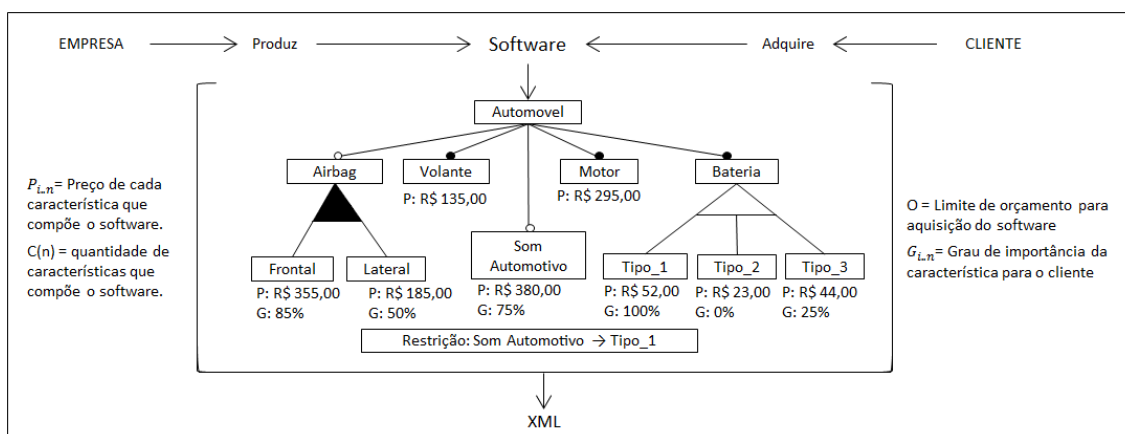


Figura 1 Modelo para apoiar a configuração de produtos em LPS

As principais constantes definidas pelo modelo do problema são: (i) o limite de orçamento que o cliente possui para aquisição do software ($O \in \text{Reais}$); (ii) o grau de importância da característica i para o cliente ($G_{1..n} \in [0, 1]$); (iii) o preço em adicionar a característica i ao produto a ser entregue ao cliente ($P_{1..n} \in \text{Reais}$); e (iv) a expressão proposicional na forma normal conjuntiva derivada a partir da árvore modelada ($E(C)$). Onde, $C(n) \in \text{Naturais}$ representa a quantidade de características presentes no modelo e $C_i \in \{0, 1\}$ representa a não inclusão ou inclusão da característica i no produto configurado. Adicionalmente, todas as restrições definidas pelo modelo devem ser satisfeitas.

$$\text{A função objetivo é maximizar } z(n) = \sum_{i=1}^n G_i C_i .$$

$$\text{Sujeito a: } \sum_{i=1}^n P_i C_i \leq O, \text{ satisfazendo } E(C).$$

Este modelo foi implementado utilizando técnicas de busca e otimização. Inicialmente utilizamos algoritmos de enumeração exaustiva com pré-processamento e backtracking [2]. Devido à natureza NP-Completo [2] deste problema, à medida que a instância do problema cresce, estes algoritmos passam a exigir muitos recursos computacionais (essencialmente, tempo de processamento) para encontrar a solução ótima. Portanto, é proposta a utilização de um algoritmo guloso de busca, baseado em heurísticas, para encontrar soluções suficientemente boas em um menor horizonte de tempo. Este modelo proporciona uma maior satisfação aos clientes que poderão adquirir produtos que atendam especificamente as suas necessidades.

6. Estado Atual do Trabalho

Além do custo-benefício do produto para o cliente, o modelo está sendo alterado para considerar as dificuldades de desenvolvimento e manutenção do produto para a organização desenvolvedora da LPS. A organização desenvolvedora tem interesse em (i) minimizar o esforço de desenvolvimento e integração de características para compor um produto e (ii) gerenciar um conjunto reduzido de versões de produtos da LPS. Para ilustrar, considere que dentre os possíveis produtos que satisfaça o cliente, um deles já se encontra instanciado e configurado. Neste caso, o produto será selecionado para

satisfazer os interesses da organização desenvolvedora, por exemplo, os custos em testes não serão necessários para este novo cliente.

O modelo proposto foi inicialmente avaliado para pequenas LPS, está sendo automatizado por uma ferramenta e sua eficácia será demonstrada através de um estudo de caso para otimizar a variabilidade em grandes LPS industriais.

7. Trabalhos Relacionados

A abordagem proposta por Sewerjuk [9] examina possibilidades e consequências na introdução de variabilidade nos preços dos ativos em LPS. O objetivo é a derivação de um framework conceitual extensível para as empresas. A abordagem torna possível criar variantes do sistema de software e servir a segmentos de mercado específicos. Müller [6] propõe um modelo matemático que visa determinar portfólio de produtos, que maximizem o lucro da empresa desenvolvedora da linha. Este modelo identifica os ativos que têm o maior impacto positivo no lucro. Ambos os problemas se preocupam particularmente com os preços dos ativos que compõem a LPS. Entretanto, restrições impostas pelos clientes, como orçamento disponível para aquisição do produto e importância dos ativos, não são considerados nestes trabalhos.

Agradecimentos

Este trabalho recebeu apoio financeiro da FAPEMIG, processos APQ-02376-11 e APQ-02532-12, e do CNPq processo 485235/2011-0.

Referências

- [1] Batory, D.; Sarvela, J. N.; Rauschmayer, A. Scaling Step-Wise Refinement. IEEE Transactions on Software Engineering, v.30, n.6, 355-371, 2004.
- [2] Cormen, T. et al. Introduction to algorithms. Cambridge, EUA: Massachusetts Institute of Technology. 2009.
- [3] Kiczales, G. et al. Aspect-Oriented Programming. In: European Conference on Object-Oriented Programming, pp. 220-242, 1997.
- [4] Loesch F. and Ploedereder E. Optimization of Variability in Software Product Lines. 11th International Software Product Line Conference, pp. 151-162, 2007.
- [5] Mendonça, M. et al. S.P.L.O.T. – Software Product Lines Online Tools. Int’l Conf. on OO Programming, Systems, Languages, and Applications (OOPSLA). 2009.
- [6] Müller J. Value-Based Portfolio Optimization for Software Product Lines. 15th International Software Product Line Conference, pp. 15-24, 2011.
- [7] Pohl, K.; Bockle, G.; Linden, F. J. v. d. Software Product Line Engineering: Foundations, Principles and Techniques. Springer-Verlag New York. 2005.
- [8] Product Line Hall of Fame. Disponível em: <http://splc.net/fame.html>
- [9] Sewerjuk D. Pricing of software product lines. In Bichler M. et al. editors, Multikonferenz Wirtschaftsinformatik, MKWI 2008, München, Berlin, 2008.
- [10] Simmonds J. et al. Analyzing Methodologies and Tools for Specifying Variability in Software Processes. Computer Science Department, Universidad de Chile, 2011.
- [11] Thum T. et al. FeatureIDE: An Extensible Framework for Feature-Oriented Software Development. Science of Computer Programming. 2012.