Anais da 2ª Escola Regional de Engenharia de Software (ERES 2018)
em Dois Vizinhos, PR, Brasil, 22 a 24 de outubro de 2018

Reúso de Software: Do Oportunista ao Sistemático

Wesley K. G. Assunção[1], Willian D. F. Mendonça[1], Silvia R. Vergilio[2]

[1]COTSI – Universidade Tecnológica Federal do Paraná (UTFPR)
CEP 85902-490 – Toledo – PR – Brasil

[2]Dinf – Universidade Federal do Paraná (UFPR)
CP 19097 – Curitiba – PR – Brasil

wesleyk@utfpr.edu.br, williandouglasferrari@gmail.com, silvia@inf.ufpr.br

https://www.overleaf.com/6422976981wfqrbvtkvwzd

Abstract. *Opportunistic reuse, also known as copy-and-paste, is a common practice in industry. This strategy presents short term benefits, however, when the number of clone products grows, technical problems of management, maintenance, and evolution arise. Software Product Line (SPL) is a systematic reuse approach presented as an alternative to deal with such problems. In an SPL, a core of similar artifacts are shared among products, avoiding duplicity and independent development of common parts. In industry, we can see that products variants developed opportunistically are the base for the SPL development, by using a re-engineering process. This paper presents a generic re-engineering process and discusses benefits and difficulties to adopt systematic reuse. We aim at motivating industries to consider adoption of systematic reuse and develop products which are easily maintained, evolved and with better quality.*

Resumo. *O reúso oportunista, também chamado de copia-e-cola, é uma prática comum na indústria. Apesar de esta estratégia apresentar benefícios a curto prazo, quando a quantidade de produtos clonados aumenta, problemas técnicos de gerenciamento, manutenção e evolução surgem. Linha de Produtos de Software (LPS) é uma abordagem de reúso sistemático que se apresenta como alternativa promissora para estes problemas. Em uma LPS, um núcleo de artefatos similares é compartilhado entre os produtos, evitando duplicidades e desenvolvimento independente de partes comuns. Na indústria, observa-se a origem de sistemas desenvolvidos de forma oportunista, que são utilizados como base para o desenvolvimento da LPS, utilizando um processo de reengenharia. Este trabalho apresenta um processo genérico de reengenharia, além disso, discute benefícios e dificuldades para aplicar o reúso sistemático. Deseja-se motivar que indústrias considerem a adoção do reúso sistemático e desenvolvam produtos mais fáceis de manter, evoluir, e com melhor qualidade.*

1. Introdução

Reúso de software é o processo de construir novos produtos de software com base na reutilização de artefatos já existentes, ao invés de construí-los do princípio [Krueger 1992]. O reúso aumenta a qualidade e produtividade de novo software e reduz o custo de desenvolvimento.

Uma prática comum em empresas de desenvolvimento é o *reúso oportunista*, também conhecido como reúso *ad hoc* ou copia-e-cola [Holmes and Walker 2013]. Nesta prática, quando existe a demanda por um novo produto ou nova funcionalidade, artefatos existentes são copiados/clonados e modificados/adaptados para satisfazer os requisitos. O reúso oportunista é uma maneira simples e intuitiva de construir novos produtos, pois não exige muito investimento inicial, e obtém bons resultados em curto período de tempo.

Contudo, a adoção intensiva e não planejada de reúso oportunista pode gerar a necessidade de refatorações constantes e acarretar débitos técnicos, tais como: comportamentos inesperados, violação de restrições, não atendimento de requisitos não-funcionais, estrutura frágil, e software inchado [Kulkarni and Varma 2017]. Neste cenário, a manutenção e evolução de produtos desenvolvidos individualmente com reúso oportunista são tarefas complexas e suscetíveis a erros [Faust and Verhoef 2003].

Para contornar os débitos técnicos relacionados ao reúso oportunista, a abordagem de Linha de Produto de Software (LPS) é uma alternativa já estabelecida e consolidada [Pohl et al. 2005]. A LPS é uma abordagem de reúso sistemático em que uma família de produtos relacionados são desenvolvidos compartilhando partes comuns para atender um segmento de mercado ou domínio específico [Clements and Northrop 2001, Linden et al. 2007]. LPSs incorporam o reúso sistemático no processo de desenvolvimento de software [Galimberti and Wazlawick 2015].

A utilização de produtos existentes para o desenvolvimento de uma LPS é conhecida como uma abordagem extrativa [Krueger 1992]. Nesta abordagem os artefatos dos produtos desenvolvidos independentemente são analisados, organizados, e transformados, por meio de um processo de reengenharia, para alcançar o reúso sistemático. Na literatura é possível observar que a aplicação de uma abordagem extrativa é o caminho mais comum para a adoção de LPSs [Laguna and Crespo 2013, Assunção et al. 2017]. Uma vez que os produtos são migrados para uma LPS, eles não são mais mantidos e evoluídos individualmente, mas como um grupo que compartilha similaridades e implementa variabilidades.

Neste trabalho o objetivo é apresentar um processo genérico de reengenharia e discutir suas características, bem como benefícios e dificuldades que são encontrados durante a condução deste processo para a construção de LPSs.

As contribuições do trabalho são: (i) fornecer para empresas de software uma visão geral sobre uma abordagem extrativa para construção de LPSs; (ii) provocar a discussão e troca de experiências entre academia e empresas em relação à utilização de LPSs; e (iii) motivar e popularizar a adoção na prática de reúso sistemático.

2. Linhas de Produtos de Software

Uma *Linha de Produtos de Software* é um conjunto de sistemas que compartilham um conjunto comum e gerenciável de características (do inglês, *features*) para satisfazer as necessidades de um segmento de mercado ou domínio específicas [Clements and Northrop 2001]. Uma *característica* é um aspecto, qualidade, ou funcionalidade de um software visível ou percebido pelo usuário do sistema [Kang et al. 1990]. Características são os blocos de construção de uma LPS.

Uma LPS é composta por duas partes principais: (i) as *partes comuns* (também conhecidas como similaridades) que são reusadas em todos os produtos, e (ii) as *partes*

variantes (também conhecidas como variabilidades) relacionadas com as características que estão presentes em apenas alguns produtos.

Engenharia de LPS (ELPS) é a atividade responsável pelo desenvolvimento de LPSs. A ELPS explora o desenvolvimento e gerenciamento das similaridades e variabilidades para a instanciação dos produtos [Clements and Northrop 2001]. A ELPS baseada em características é a abordagem de desenvolvimento mais utilizada atualmente [Apel et al. 2013]. A Figura 1 apresenta a estrutura desta abordagem. Existem basicamente duas atividades principais:

- A *engenharia de domínio* (no topo da figura), que tem como objetivo principal a organização e representação dos produtos que compõem o segmento ou domínio em que a LPS está inserida;
- A *engenharia de aplicação* (na base da figura), responsável pelos aspectos de implementação das similaridades e variabilidades, e pela derivação dos produtos reutilizando-se os artefatos desenvolvidos e as características selecionadas.

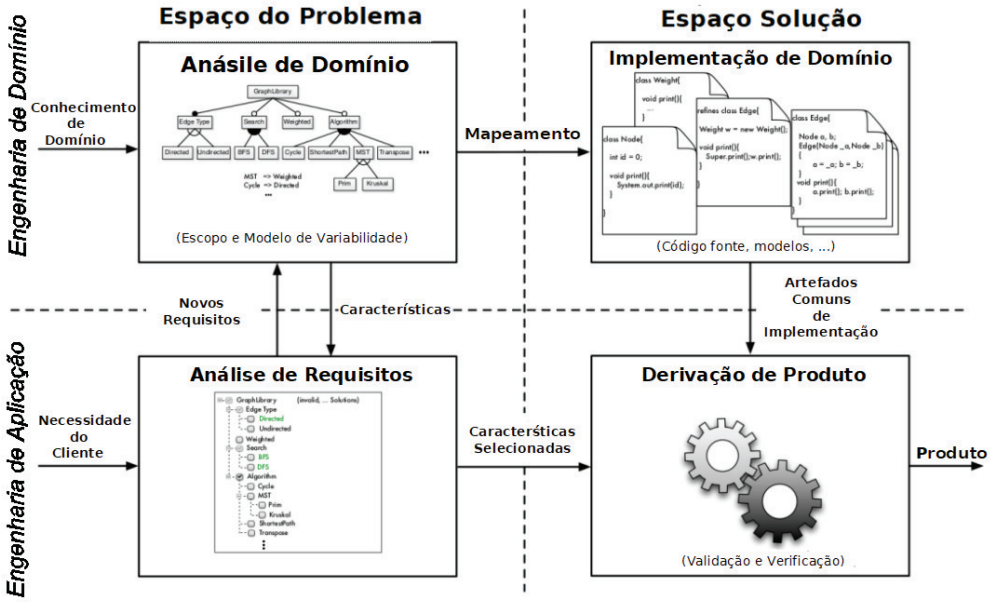

Figura 1. Framework de ELPS, adaptado de [Apel et al. 2013]

Durante as atividades de ELPS, dois artefatos principais são construídos: (i) o *modelo de características*, que representa a organização das similaridades/variabilidades e dá suporte à comunicação dos envolvidos com a LPS; e (ii) a *arquitetura da linha de produtos*, responsável por descrever como as características/funcionalidades são implementadas e quais artefatos são utilizados na instanciação de cada produto.

3. O Processo de Reengenharia

Muitos trabalhos são encontrados sobre a extração de LPSs a partir de produtos variantes. Em um estudo recente foi definido um processo de reengenharia com base nas atividades comuns na extração de LPSs [Assunção et al. 2017]. A Figura 2 apresenta uma visão geral das fases do processo de reengenharia. Do lado esquerdo da figura apresentam-se

as variantes desenvolvidas com reúso oportunista. A linha sólida apresenta o processo completo de reengenharia, produzindo uma LPS, apresentada do lado direito da figura. As fases do processo, apresentadas por linhas tracejadas, são descritas a seguir.

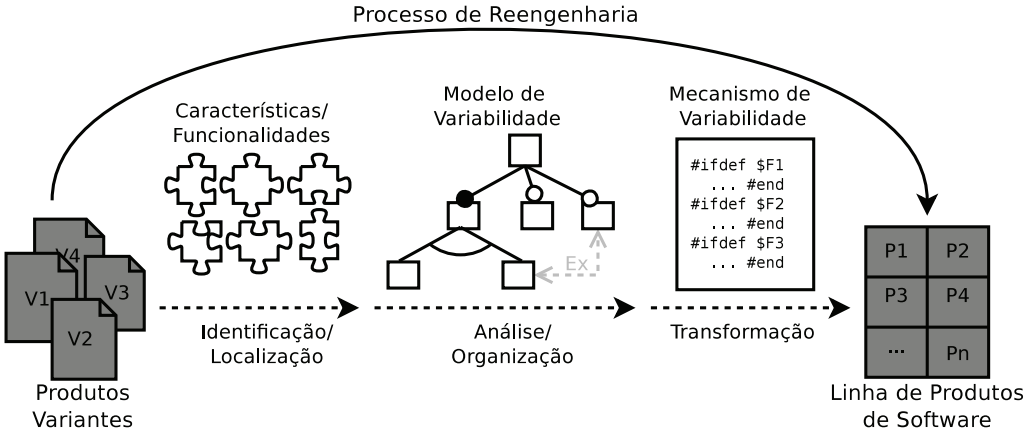

Figura 2. Processo de Reengenharia, adaptado de [Assunção et al. 2017]

O processo de reengenharia é formado por três fases genéricas: (i) *Identificação/Localização* é responsável por identificar as características/funcionalidades similares e variantes espalhadas pelos produtos, e ainda, fazer a rastreabilidade entre características e os artefatos de implementação. (2) *Análise/Organização* é responsável por definir como as características são organizadas, para a criação de um modelo de características. (3) *Transformação* tem como foco a construção de artefatos de LPS, como por exemplo a arquitetura da linha de produtos, e a alteração dos artefatos de implementação para incluir mecanismo de variabilidades, tal como diretivas `#ifdef`, possibilitando a posterior instanciação de produtos.

4. Discussões

Exitem relatos da adoção de LPS por um processo de reengenharia na indústria. Ommering apresentou a aplicação na Philips [van Ommering 2005], já Jansen et al. demonstram a aplicação da reengenharia em organizações de pequeno e médio porte [Jansen et al. 2008]. Galimberti e Wazlawick reportam a utilização de LPS com foco em internacionalização de produtos [Galimberti and Wazlawick 2015]. Contudo, estes trabalhos não apresentam explicitamente vantagens e desvantagens observadas durante a condução dos estudos. A partir de um estudo prévio [Assunção et al. 2017], e com base na literatura recente, foram identificados benefícios da adoção de LPS e dificuldades na condução do processo de reengenharia.

Em geral, os benefícios da adoção de reúso sistemático são:

- *Facilidade para desenvolver novos produtos*: uma vez que as características dos sistemas estão organizadas em artefatos reutilizáveis, a criação de novos produtos, aumentando o portfólio das empresas, é uma atividade simples, pois compreende somente a seleção de uma nova configuração ou o desenvolvimento de apenas uma nova funcionalidade, não requerendo adaptações *ad hoc*;
- *Facilidade para manter os produtos*: para os produtos que reutilizam uma base comum de artefatos, a manutenção das características similares são executadas

somente uma vez, e automaticamente serão propagadas para todos os produtos variantes. Já no reúso oportunista, todos os produtos deveriam ser mantidos individualmente;
- *Facilidade para evoluir o sistema*: atividades de evolução, como por exemplo a troca de plataforma *desktop* para *web*, pode ser conduzida mais facilmente, já que uma vez migradas as similaridades, todos os produtos já terão o novo comportamento, restando apenas a evolução das variabilidades.
- *Melhor comunicação entre os envolvidos*: artefatos gerados durante o processo de reengenharia, tal como o diagrama de características e a arquitetura da linha de produtos, oferecem aos envolvidos com o sistema uma visão ampla e geral de como os produtos são organizados, o que implementam, quais as funcionalidades mais reutilizadas, etc. Isso auxilia a comunicação e tomada de decisões tanto técnicas quanto gerenciais;
- *Geração de casos de teste melhores*: através da visão geral do sistema, casos de teste que exercitam interações específicas entre características, ou que preveem possíveis "gargalos" de funcionamento, podem ser definidos, aumentando a qualidade do produto.

Por outro lado, através da observação da literatura e relatos de experiências, pode-se verificar que existem dificuldades em conduzir o processo de reengenharia:

- *Falta de ferramentas*: muitos trabalhos apresentam ferramentas com propósito muito específico para a reengenharia, o que impossibilita a sua aplicação em outros cenários. Sem uma ferramenta para automatizar e dar suporte ao processo, empresas necessitam muito esforço humano, algumas vezes inviabilizando a adoção de LPS;
- *Mudança de paradigma nas empresas*: empresas que construíram produtos sem considerar reúso sistemático durante o ciclo de desenvolvimento podem necessitar de uma maior conscientização para que essa nova perspectiva seja bem-sucedida. Em situações em que a reengenharia é um processo complexo devido ao domínio do sistema, a equipe deve estar coesa para alcançar o objetivo;
- *Poucos casos de sucesso*: apesar da existência de relatos na literatura, empresas ainda necessitam de exemplos mais concretos para decidir sobre os benefícios da adoção de LPS. A maior parte da literatura refere-se à empresas de grande porte, mas em geral empresas menores, que são mais numerosas, não são consideradas.

5. Considerações Finais

Este trabalho apresenta um processo genérico de reengenharia de produtos existentes para uma LPS, permitindo que empresas resolvam problemas técnicos associados ao reúso oportunista. A abordagem extrativa é uma alternativa promissora para sistematizar o reúso.

Os benefícios da adoção do reúso sistemático são a facilidade para desenvolver novos produtos por meio do reúso, executar atividades de manutenção e evolução, possibilitar melhor comunicação e tomada de decisões, e apoiar a derivação de casos de teste melhores. Entretanto, existe a carência de ferramentas mais genéricas para apoiar o processo de reengenharia e, além disso, é necessário considerar o comprometimento dos envolvidos no processo de adoção do reúso sistemático. Outro fator é a existência

de poucos casos de sucesso apresentados com dados concretos, para motivar e servir de exemplos para empresas que visam à adoção de LPSs.

Como trabalho futuro deseja-se conduzir o processo de reengenharia em um estudo de caso. Todas as etapas serão documentadas para criar um conjunto de informações, facilitando-se o entendimento por parte dos interessados em adotar o reúso sistemático.

Referências

Apel, S., Batory, D., Kästner, C., and Saake, G. (2013). *Feature-Oriented Software Product Lines*. Springer.

Assunção, W. K. G., Lopez-Herrejon, R. E., Linsbauer, L., Vergilio, S. R., and Egyed, A. (2017). Reengineering legacy applications into software product lines: A systematic mapping. *Empirical Software Engineering*, pages 1–45.

Clements, P. and Northrop, L. (2001). *Software Product Lines: Practices and Patterns*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Faust, D. and Verhoef, C. (2003). Software product line migration and deployment. *Software: Practice and Experience*, 33(10):933–955.

Galimberti, M. and Wazlawick, R. (2015). Active internationalization of small and medium-sized software enterprises - cases of french software companies. *Journal of Technology Management & Innovation*, 10(4):99–108.

Holmes, R. and Walker, R. J. (2013). Systematizing pragmatic software reuse. *ACM Transactions on Software Engineering and Methodology*, 21(4):20:1–20:44.

Jansen, S., Brinkkemper, S., Hunink, I., and Demir, C. (2008). Pragmatic and opportunistic reuse in innovative start-up companies. *IEEE Software*, 25(6):42–49.

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.

Krueger, C. W. (1992). Software reuse. *ACM Computing Surveys*, 24(2):131–183.

Kulkarni, N. and Varma, V. (2017). Perils of opportunistically reusing software module. *Software: Practice and Experience*, 47(7):971–984.

Laguna, M. A. and Crespo, Y. (2013). A systematic mapping study on software product line evolution: From legacy system reengineering to product line refactoring. *Science of Computer Programming*, 78(8):1010–1034. Special section on software evolution, adaptability, and maintenance & Special section on the Brazilian Symposium on Programming Languages.

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, Inc., Secaucus, NJ, USA.

Pohl, K., Böckle, G., and Linden, F. J. v. d. (2005). *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

van Ommering, R. (2005). Software reuse in product populations. *IEEE Transactions on Software Engineering*, 31(7):537–550.