

Em Busca de uma Arquitetura de Referência para Frameworks de Aplicação Dirigidos por Modelos para Sistemas de Informação

Valdemar Vicente GRACIANO NETO¹; Juliano Lopes DE OLIVEIRA¹

¹Instituto de Informática - Universidade Federal de Goiás (UFG)
Caixa Postal 131 – Campus II – CEP 74001-970 – Goiânia – GO – Brasil
Universidade Federal de Goiás - UFG
{valdemarneto, juliano}@inf.ufg.br

Resumo. *Frameworks de Aplicação (FA) promovem a produtividade e o reúso em projetos de desenvolvimento de software. Conceitos de Desenvolvimento de Software Dirigido por Modelos (DSDM) têm sido empregados em FA para apoiar a construção de Sistemas de Informação (SI). No entanto, não há na literatura uma proposta de arquitetura de referência consistente que oriente a construção deste tipo específico de software. Este artigo apresenta resultados de uma dissertação de mestrado que visa investigar e criar uma arquitetura de referência para FA de SI baseados em DSDM.*

Palavras-chave: Framework de Aplicação, Sistemas de Informação, Arquitetura de Referência, Desenvolvimento de Software Dirigido por Modelos.

1. Introdução

Desenvolvimento de Software Dirigido por Modelos (DSDM) é um novo paradigma de desenvolvimento de software que consiste em transformações sucessivas de modelos em alto nível de abstração para produzir um software funcional. DSDM aumenta o nível de abstração e automação na construção de software [Izquierdo and Molina 2010] pois o foco são os *modelos*.

Frameworks de Aplicação (FA) são ferramentas que empregam técnicas de reúso de código e de projeto para o desenvolvimento de aplicações específicas de domínio. Eles constituem aplicações semi-completas, estruturadas como uma coleção organizada de componentes de software reutilizáveis para facilitar a implementação de aplicações de software customizadas [Fayad and Schmidt 1997]. Suas principais características são a modularidade, reúso, extensibilidade e Inversão de Controle (IC) [Fayad and Schmidt 1997, Fowler 2005]. Eles têm se tornado o padrão *de-facto* para implementação de software [Mailloux 2010].

Os *hot spots* definem onde o framework pode ser customizado, gerando novos serviços e características [Fayad and Schmidt 1997]. Pontos de imutabilidade constituem o núcleo do framework, e são chamados de *frozen spots* [Markiewicz and de Lucena 2001].

Com a popularização de DSDM começaram a surgir FA para DSDM. Tais frameworks encapsulam detalhes de projeto e código relativos à utilização de DSDM para fornecer elementos como transformadores de modelos, validadores de modelos, meta-modelos, etc. Entretanto, conceitos de DSDM tem sido empregados também em FA não específicos para DSDM, como Frameworks de Aplicação para construção de Sistemas de Informação (SI) dirigidos por Modelos (FASIM). A aplicação de princípios de DSDM em FA para outros fins como SI não está bem compreendida.

FASIM devem encapsular similaridades e variabilidades referentes especificamente à produção de software de SI, tais como funcionalidades CRUD (*Create, Read, Update, Delete*), regras de negócio, processos de negócio, aspectos de interação, mapeamento objeto-relacional e representação de dados do negócio. Logo, os frameworks específicos para DSDM cujas funcionalidades mais comuns são regras de transformação, metamodelos, verificadores de modelos, dentre outras, não satisfazem a todos os aspectos relevantes de software de SI.

AR descrevem conceitos em alto-nível e terminologia que permitem discutir aspectos comuns de implementação de um domínio em particular [Heitmann et al. 2009]. Cada AR deve ser capaz de representar os subsistemas fundamentais comuns aos sistemas de um domínio, bem como as relações entre estes subsistemas. Logo, uma AR pode ajudar na análise e escolha entre diferentes opções de projeto, e servir como modelo para concepção de novos sistemas e reengenharia dos existentes [Grosskurth and Godfrey 2006]. A falta de AR para FASIM é o que motiva esta pesquisa.

Este trabalho apresenta resultados parciais da investigação sobre critérios e características arquiteturais de FASIM, uma categoria de frameworks emergente para a qual não há muitos relatos na literatura. O restante do texto estrutura-se como segue: Seção 2 descreve a metodologia de pesquisa bem como os passos já executados do plano elaborado; a Seção 3 apresenta os resultados e a discussão; a Seção 4 conclui o artigo com uma avaliação sobre os resultados obtidos até o momento; a Seção 5 apresenta os agradecimentos.

2. Metodologia de Trabalho

O trabalho sendo desenvolvido prevê a execução das atividades de pesquisa citadas a seguir:

1. **Fundamentação Teórica:** levantamento bibliográfico para investigação e análise de modelos, métodos, definições e arquiteturas de FASIM;
2. **Engenharia de Requisitos:** identificação de características necessárias para uma AR para FASIM. Esta atividade engloba a Engenharia de Domínio (comum no desenvolvimento de frameworks) e levantamento de variabilidades e similaridades.
3. **Design:** projeto arquitetural e detalhado de uma AR que contemple as características identificadas na fase anterior;
4. **Validação:** confirmação da adequação da AR para FASIM identificados na literatura e implementação de protótipos que validem a AR;
5. **Redação:** elaboração do texto da dissertação de mestrado.

O levantamento bibliográfico e a Engenharia de Requisitos de FASIM já foram realizadas, permitindo a identificação dos requisitos para o desenvolvimento das demais atividades. A atividade de Design está sendo realizada e os esforços concentram-se na conclusão do projeto detalhado de componentes do framework e na identificação dos *hot spots* e *frozen spots* do framework. A redação da dissertação acontece de forma concorrente com as demais fases, e vem sendo realizada de forma incremental desde a concepção do projeto.

3. Resultados e Discussão

Foram encontrados três trabalhos correlatos: [Okanović et al. 2010], [Almeida et al. 2009] e [Langegger et al. 2006]. Estes trabalhos descrevem FA que modelam e geram SI web de modo semi-automático e baseado em modelos. [Okanović et al. 2010] e [Almeida et al. 2009] não apresentam a arquitetura dos frameworks tampouco as características de Inversão de Controle (IC) ou *hot/frozen spots*. [Langegger et al. 2006] menciona o conceito de *interaction spots* (pontos de extensão referentes à interação com o usuário), ilustra uma proposta arquitetural em alto nível e menciona indiretamente a existência de IC. Não há detalhes a respeito de outros *hot spots*, nem de *frozen spots*.

Não foi encontrado trabalho diretamente correlato para a pesquisa proposta até onde a investigação foi conduzida. As descrições de AR mais próximas presentes na literatura como [Heitmann et al. 2009, Grosskurth and Godfrey 2006] apresentam AR para aplicações de web semântica e navegador web, respectivamente.

Conhecimentos sólidos obtidos de resultados de pesquisa anteriores estão sendo aplicados para viabilizar a construção da AR [Almeida et al. 2009, da Costa and de Oliveira 2010].

Uma proposta de AR para FASIM está sendo estruturada utilizando camadas. Na camada de apresentação/interação com o usuário, vislumbrou-se o conceito de *Es-*

teriótipo de Interface (EI) como um ponto de extensão (*hot spot*). EI consiste em uma abstração da intenção da interface, independente da aplicação ou do SI subjacente [da Costa and de Oliveira 2010]. Sendo assim, o EI constitui uma representação genérica, extensível, abstrata e completável de interfaces, tais como CRUD, Portal, etc. Logo, pode ser caracterizado como um *hot spot*.

Na camada de aplicação, as funcionalidades CRUD foram identificadas como *frozen spots*. Tais funcionalidades são o âmago de software de SI e caracterizam-se como uma similaridade notável deste tipo de software, devendo ser modelada como *frozen spot* em um FASIM.

Na camada de negócio, as classes constituintes do metamodelo do FA foram identificados como um ponto de extensão. Apesar de não ser convencional utilizar a relação “é uma instância de” para relacionar entidades do negócio com suas respectivas meta-entidades, para FASIM a relação de instância é perfeitamente compreensível visto que é o preenchimento dos dados em nível conceitual que possibilita a geração de representações para a entidade do negócio em nível concreto.

Na camada de persistência os mapeamentos objeto-relacionais são efetuados como transformações entre modelos. Para evitar dependências de tecnologias, as regras de transformação foram vislumbradas como um ponto de extensão. Elas podem ser especificadas de modo abstrato e genérico. Isso faz com que mudanças na plataforma de persistência de destino possam ser realizadas sem grandes alterações no funcionamento total do framework.

4. Conclusões

A proposta de uma Arquitetura de Referência (AR) para Frameworks de Aplicação (FA) para Sistemas de Informações dirigidos por Modelos (FASIM) é de interesse para a comunidade de Engenharia de Software, visto que não foram encontradas propostas correlatas para este tipo específico de FA. A concepção de AR evidencia similaridades entre softwares de um mesmo domínio, valorizando o conhecimento construído pela prática e pela comunidade científica, protegendo o capital intelectual das equipes de desenvolvimento e favorecendo a construção de software de qualidade.

Elementos essenciais da AR já foram identificados. O trabalho encontra-se em fase de implementação. *Hot spots* e *frozen spots* foram identificados, possibilitando o início da definição da AR. Protótipos tem sido desenvolvidos para o transformador de modelos utilizado no framework, componente de persistência, componente de interação com o usuário e componente de gerência de processos de negócio.

Protótipos para componentes de Persistência, Interação com Usuário, Gerência de

Processo de Negócio e Transformação de Modelos já estão sendo desenvolvidos como parte da eliciação e da validação dos requisitos da AR que está sendo proposta.

5. Agradecimentos

À CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo apoio financeiro concedido durante todo o decorrer deste projeto.

References

- Almeida, A. C., Boff, G., and Oliveira, J. L. (2009). A framework for modeling, building and maintaining enterprise information systems software. In *Anais do XXIII Simpósio Brasileiro de Engenharia de Software*, pages 115–125. Fortaleza, Brasil.
- da Costa, S. L. and de Oliveira, J. L. (2010). Construção e Manutenção Baseadas em Modelos de Interfaces para Usuário em Sistemas de Informação. In *Anais do III Workshop de Teses e Dissertações de Sistemas de Informação do VI Simpósio Brasileiro de Sistemas de Informação*, Maraba, PA, Brasil.
- Fayad, M. and Schmidt, D. C. (1997). Object-oriented application frameworks. *Communications of ACM*, 40(10):32–38.
- Fowler, M. (2005). Inversion of control. <http://martinfowler.com/bliki/InversionOfControl.html>.
- Grosskurth, A. and Godfrey, M. W. (2006). A reference architecture for web browsers. *Journal of Software Maintenance and evolution: research and practice*, 18(2):1–7.
- Heitmann, B., Hayes, C., and Oren, E. (2009). Towards a reference architecture for semantic web applications. In *Proc. of the 1st International Web Science Conference*.
- Izquierdo, J. L. C. and Molina, J. G. (2010). An architecture-driven modernization tool for calculating metrics. *IEEE Software*, 27(4):37 – 43.
- Langegger, A., Palkoska, J., and Wagner, R. (2006). Davinci - a model-driven web engineering framework. *International Journal of Web Information Systems*, 2:119 – 134.
- Mailloux, M. (2010). Application frameworks: how they become your enemy. In *Proceedings of the ACM International Conf. on OOP Systems Languages and Applications*, SPLASH '10, pages 115–122, New York, NY, USA. ACM.
- Markiewicz, M. E. and de Lucena, C. J. P. (2001). Object oriented framework development. *Crossroads*, 7:3–9.
- Okanović, V., Donko, D., and Mateljan, T. (2010). Frameworks for model-driven development of web applications. In *Proc. of 9th WSEAS International Conference on Data Networks, Communications, Computers*, DNCOCO'10, pages 67–72, Stevens Point, Wisconsin, USA. WSEAS.