

Estudo para Adaptação de um Processo Ágil de Desenvolvimento baseado em Framework para apoiar o Desenvolvimento de Software baseado em Modelos

Thais Cristina Costa^{1,2}, Franciene Duarte Gomes¹, Maria Istela Cagnin¹

¹UNIVEM - Centro Universitário Eurípides de Marília

Caixa Postal 2041, CEP 17525-901, Marília-SP

²FUNEC – Faculdades Integradas de Santa Fé do Sul, Santa Fé do Sul-SP

thaiscriscosta@yahoo.com.br, franciene@gmail.com, istela@univem.edu.br

Abstract. *There are many researches on model-driven development, especially those based on the MDA (Model-Driven Architecture) approach. One of the objectives of that development type is to elaborate models capable of capturing the semantics of the system and relevant information on it so that the supporting tools can create as many source code lines as possible for the software represented by these models, in order to increase the productivity of development teams. This paper shows a sketch of a model-driven process based on the MDA approach. This sketch was made through a detailed analysis of the PARFAIT/EA process documentation which uses frameworks as computational support.*

Resumo. *Diversas pesquisas sobre o desenvolvimento de software baseado em modelos são encontradas na literatura, especificamente utilizando a abordagem MDA (Model Driven Architecture). Uma das preocupações desse tipo de desenvolvimento é permitir que os modelos elaborados em um determinado projeto sejam capazes de capturar a semântica e as informações relevantes do sistema, para permitir que ferramentas de apoio consigam gerar a maior quantidade possível de linhas de código-fonte do software representado por tais modelos a fim de aumentar a produtividade das equipes de desenvolvimento. Este artigo apresenta um esboço de um processo baseado em modelos, que utiliza a abordagem MDA. Esse esboço foi definido por meio da análise detalhada da documentação do processo PARFAIT/EA que utiliza frameworks como apoio computacional.*

1. Introdução

Atualmente, devido às exigências do mercado e a grande competitividade, as empresas precisam desenvolver sistemas com prazos cada vez mais curtos. Para permitir isso, têm-se os processos ágeis, que englobam um conjunto de atividades baseadas nas práticas de métodos ágeis (Beck, 2000), produzindo uma primeira versão do software com qualidade o mais rápido possível. Além disso, tais métodos diminuem a burocracia

existente nos métodos tradicionais, facilitando a comunicação entre os participantes do projeto e diminuindo os riscos associados e prazos (Gomes, 2007).

PARFAIT/EA (Gomes e Cagnin, 2007) é um dos processos ágeis existentes na literatura e é de interesse deste trabalho. Esse processo é um dos recursos de um Arcabouço de Reengenharia Ágil¹ (ARA) (Cagnin, 2005) e tem como objetivo apoiar o desenvolvimento de sistemas orientados a objetos no domínio de Sistemas de Informação com o apoio de *frameworks* cuja construção tenha sido baseada em linguagens de padrões de análise. Segundo Fayad e Schmidt (1997), *frameworks* representam uma estrutura formada por blocos pré-fabricados de *software* que os programadores podem usar, estender ou adaptar para uma solução específica e linguagens de padrões (Appleton, 1997) são formadas por um conjunto de padrões de *software* para resolver um problema complexo em um determinado domínio.

Processos estão sendo definidos na literatura para apoiar o desenvolvimento baseado em modelos (*Model-Driven Development-MDD*) (Souza (2004), Silva (2004), Marques (2006) e Morgado (2007)). De acordo com Belix (2006), o desenvolvimento baseado em modelos é uma tentativa de elevar o nível de abstração no desenvolvimento de software para além das atuais linguagens de programação. Adicionalmente, MDD não exige nos modelos a definição de todos os detalhes de implementação necessários pelas linguagens de programação, ou seja, separa a especificação dos detalhes da implementação. Para apoiar o desenvolvimento baseado em modelos, várias abordagens estão sendo criadas, como é o caso da MDA - *Model Driven Architecture* (OMG, 2006), que é utilizada neste artigo por ser a mais popular e por ser um padrão da OMG² (*Object Management Group*).

Nesse contexto, este artigo apresenta o esboço de um processo ágil de desenvolvimento baseado em modelos. A definição do esboço do processo é abstraída a partir da documentação do processo ágil PARFAIT/EA (Gomes *et al.*, 2006). Com a definição do esboço do processo esperam-se fornecer indicações para facilitar a análise, a documentação e a implementação do *software* gerado com o uso do processo, bem como para apoiar a utilização de ferramentas baseadas em MDA. O artigo está organizado da seguinte maneira: na Seção 2 apresentam-se os trabalhos relacionados, na Seção 3 apresenta-se uma visão geral do PARFAIT/EA, na Seção 4 apresenta-se a definição do esboço do processo proposto e na Seção 5, apresentam-se as conclusões finais e perspectivas de trabalhos futuros.

2. Trabalhos Relacionados

Dentre os diversos trabalhos encontrados na literatura sobre processos e abordagens de desenvolvimento baseado em modelos utilizando a abordagem MDA, aqueles que estão mais próximos do enfoque deste trabalho são o processo RAPDIS (Morgado, 2007), a abordagem XIS (Silva, 2003) e o Processo de Desenvolvimento Baseado em Componentes Adaptado ao MDA (Sousa, 2004).

¹ Abordagem composta por processos, técnicas e ferramentas utilizada tanto para apoiar a reengenharia quanto o desenvolvimento ágil.

² <http://www.omg.org/>

O processo RAPDIS auxilia o desenvolvimento de Sistemas de Informação, oferecendo um suporte completo ao MDA, ou seja, na produção dos modelos específicos dessa abordagem, que são CIM (Modelo Independente de Computação), PIM (Modelo Independente de Plataforma) e o PSM (modelo Específico de Plataforma), até a geração do código fonte, que é realizada por meio de uma ferramenta específica.

O CIM representa os requisitos do sistema, descrevendo a situação na qual o sistema será utilizado. O PIM é o modelo que descreve todo o negócio do sistema, sem se preocupar em qual plataforma ou tecnologia será implementado. O PSM combina a especificação do modelo PIM com detalhes de uma determinada plataforma. Ele recebe detalhes da construção do sistema baseado na tecnologia escolhida. Esse modelo é o de mais baixo nível de abstração e seus elementos estão prontos para a geração de código.

O objetivo principal da abordagem XIS é o desenvolvimento de software baseado em modelos, com o apoio de MDA, centrado em arquiteturas de software e baseado em técnicas de geração automática de *templates* disponíveis. Essa abordagem preocupa-se apenas com a elaboração dos modelos PIM's, seguindo o padrão MVC (*Model View Controller*). As transformações desses modelos para uma determinada plataforma são definidas pelos arquitetos quando utilizam os *templates* correspondentes a uma determinada arquitetura de software e, finalmente, pelos programadores quando selecionam os modelos e os *templates* necessários para efetuar as transformações em componentes de software correspondentes.

O Processo de Desenvolvimento Baseado em Componentes Adaptado ao MDA inclui o tratamento explícito dos requisitos não-funcionais por meio do refinamento da arquitetura de software e do uso de um modelo de estruturação de componentes independentes de plataforma, mais especificamente o COSMOS (*Component Structuring Model for Object-Oriented Systems*) (Silva, 2003). Esse modelo permite um mapeamento entre as abstrações de uma descrição arquitetural para a construção em plataformas de componentes específicos.

Dentre os processos e abordagens citados, observou-se que nenhum trata explicitamente de algumas atividades específicas de desenvolvimento de software, como atividades de planejamento de projeto, de teste e de gerência de configuração. Salienta-se que apesar do processo RAPDIS não possuir uma atividade específica de teste, teste de aceitação, de unidade e de integração são considerados durante a execução de algumas atividades do processo. O processo proposto pretende tratar explicitamente dessas atividades.

3. Processo Ágil de Desenvolvimento PARFAIT/EA

O processo PARFAIT/EA (GOMES, 2007) tem como objetivo apoiar o desenvolvimento de *software* orientado a objetos com apoio computacional de *frameworks* de aplicação (Fayad e Schmidt, 1997), cuja construção do *framework* seja baseada em uma linguagem de padrões de análise (LPA). O *framework* é utilizado na construção do *software* e a LPA facilita tanto a análise do *software* desenvolvido, quanto na documentação, entendimento e uso do *framework*.

O processo PARFAIT/EA utiliza algumas práticas ágeis encontradas na literatura para apoiar a execução de cada atividade, como por exemplo: versões

pequenas, cliente presente, testes constantes, jogo do planejamento, propriedade coletiva do código, integração contínua, metáfora, entre outras.

Na Figura 1 apresenta-se uma visão simplificada do processo PARFAIT/EA, contendo as suas fases e atividades. Todas as atividades do processo são descritas por passos numerados para facilitar sua execução, mas não estão mostrados na figura por falta de espaço.

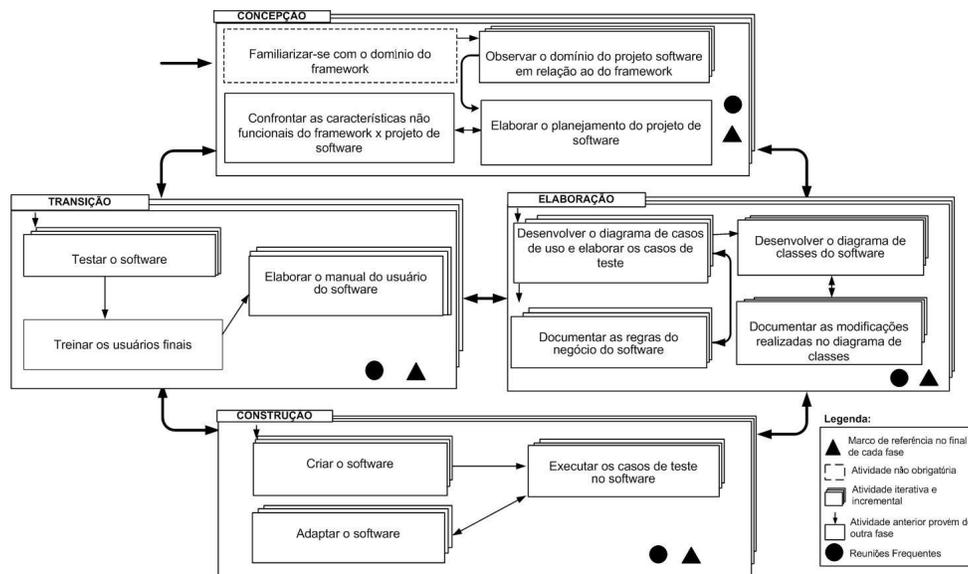


Figura 1 – Versão simplificada do processo PARFAIT/EA (adaptada de GOMES *et al*, (2006))

As atividades do processo PARFAIT/EA são classificadas como: obrigatória, não obrigatória e atividades iterativa e incremental. Nesse último tipo de atividade, o engenheiro de *software* pode reexecutá-la a fim de refinar os artefatos anteriormente elaborados.

Cada fase do processo possui marcos de referências, representados na Figura 1 por triângulos, e que tem como objetivo avaliar se a execução de cada atividade da fase foi aplicada corretamente e verificar como está o andamento do projeto de *software*. A atividade “Reuniões frequentes”, tem como objetivo a participação do cliente, podendo ser realizada no início ou no fim de cada fase ou na conclusão de cada atividade e está representada na Figura 1 por um círculo.

A primeira fase do processo é a **fase de Concepção** e tem como objetivo principal observar o projeto de *software* em relação ao *framework* disponível e os riscos em utilizar tal *framework*. A fase seguinte é a de **Elaboração** e tem como objetivo produzir documentação suficiente, com apoio da linguagem de padrões de análise, para apoiar a construção do *software*, as adaptações e futuras manutenções; bem como elaborar casos de teste funcional baseado em critérios de teste específicos para apoiar a atividade de teste nas fases seguintes. A próxima fase é a de **Construção** e tem como objetivo criar o *software*, com o apoio do *framework* disponível, de acordo com os requisitos identificados e priorizados pelo cliente para o ciclo corrente e permitir, quando necessário e identificados pelos casos de teste, que os requisitos sejam

adaptados para atender as exigências do projeto de *software*. A última fase é a de **Transição** e tem como objetivo garantir que a versão atual do *software* esteja de acordo com as exigências do ciclo para que possa ser disponibilizada para o cliente. Isso é feito com a aplicação dos casos de teste criados anteriormente.

Para apoiar a documentação do processo, o planejamento e a execução de projetos com o apoio do PARFAIT/EA, a ferramenta PDocTool (Gomes, 2007) é utilizada.

4. Esboço do Processo Ágil de Desenvolvimento baseado em Modelos

Esta seção apresenta o esboço do processo ágil de desenvolvimento baseado em modelos, proposto neste artigo. Tal processo é abstraído a partir de uma análise detalhada da documentação do processo PARFAIT/EA. Para isso, cada atividade e cada passo desse processo existente serão analisados sob a perspectiva da alteração da forma de desenvolvimento, ou seja, de desenvolvimento baseado em *framework* para desenvolvimento orientado por modelos. Dessa forma, algumas atividades do processo PARFAIT/EA serão aproveitadas, outras serão adaptadas, retiradas ou adicionadas para compor o novo processo, conforme justificativas descritas a seguir.

Salienta-se que o processo PARFAIT/EA enfoca o reuso dos padrões da linguagem de padrões de análise e da hierarquia de classes do *framework*, já o esboço do processo proposto terá disponível para reuso os modelos (como é o caso do CIM - Modelo Independente da Computação, PIM - Modelo Independente de Plataforma e PSM - Modelo Específico de Plataforma) de diversos domínios de interesse, a medida que for sendo utilizado.

Para a definição das atividades da **fase de Concepção** do esboço do processo proposto, uma atividade dessa fase do processo PARFAIT/EA foi mantida, duas foram retiradas e uma foi adaptada. A primeira atividade “Familiarizar-se com o domínio do *framework*” foi retirada pois o esboço do processo proposto é fundamentado no desenvolvimento baseado em modelos e não baseado em *framework*, e tem como preocupação inicial a elaboração ou o reuso dos modelos no mesmo domínio do sistema que será desenvolvido.

A outra atividade da fase de Concepção “Observar o domínio do projeto de software em relação ao do *framework*” foi adaptada para permitir a identificação dos modelos disponíveis (CIM, PIM, PSM) para serem reutilizados no projeto de software e teve seu nome alterado para “Observar o domínio do projeto de software em relação aos modelos disponíveis” a fim de adequá-lo ao MDD. A atividade “Elaborar o planejamento do projeto de software” foi mantida na íntegra por ser uma atividade comum a qualquer processo ágil de desenvolvimento. Observou-se a necessidade de retirar a atividade “Confrontar as características não funcionais do *framework* x projeto de software” do PARFAIT/EA pois o esboço do processo proposto é fundamentado no desenvolvimento baseado em modelos e as características não funcionais também são especificadas nos modelos PIM e/ou PSM. Como o objetivo do processo proposto é fornecer reuso dos modelos disponíveis, a atividade “Elaborar os modelos do domínio de interesse” foi adicionada nesta fase pois caso não existam modelos disponíveis para apoiar o desenvolvimento do projeto, é necessário executar essa atividade. Nessa

atividade serão construídos os modelos CIM, PIM e PSM para um determinado domínio.

Para a definição das atividades da **fase de Elaboração** do esboço do processo proposto, duas atividades dessa fase do processo PARFAIT/EA foram mantidas, uma foi adaptada e uma foi retirada. A primeira atividade dessa fase “Desenvolver o diagrama de casos de uso e elaborar os casos de teste” foi mantida no esboço do processo proposto, pois também pertence ao contexto de desenvolvimento baseado em modelos. Como no processo PARFAIT/EA, os testes serão criados anteriormente à implementação, seguindo a prática ágil “testes constantes”, com o apoio dos critérios de teste funcional “particionamento de equivalência” e “análise do valor limite” (Myers, 2004). Todos os seus passos serão mantidos na íntegra. Outra atividade do processo PARFAIT/EA mantida na íntegra nessa fase é “Documentar as regras de negócio do software” pois também pertence ao contexto de desenvolvimento baseado em modelos.

A atividade do PARFAIT/EA “Desenvolver o diagrama de classes do software” teve que ser adaptada uma vez que PARFAIT/EA tem como apoio a linguagem de padrões de análise utilizada na construção do *framework* e o esboço do processo tem como apoio os modelos disponíveis no mesmo domínio do projeto sendo desenvolvido. Observou-se que a atividade “Documentar as modificações realizadas no diagrama de classes” do PARFAIT/EA teve que ser desconsiderada do esboço pois é necessária para apoiar a implementação manual (adaptação) dos requisitos e regras de negócios do sistema não cobertos pela linguagem de padrões e, conseqüentemente, não fornecidas pelo *framework*. No caso do esboço do processo, não há como identificar as partes do modelo que não terão código fonte correspondente de maneira adequada. As discrepâncias do comportamento desejado pelo usuário em relação ao comportamento fornecido pelo código fonte gerado pela ferramenta de apoio serão observadas a partir da atividade de teste realizada por meio da execução da atividade “Executar os casos de teste no software” (Fase de Construção), discutida posteriormente.

Para a definição das atividades da **fase de Construção** do esboço do processo proposto, duas atividades dessa fase do processo PARFAIT/EA foram adaptadas e uma foi mantida. A atividade “Criar o software” foi adaptada uma vez que o PARFAIT/EA utiliza como apoio computacional nesta atividade *frameworks* e algumas atividades do esboço do processo proposto utilizarão ferramentas de apoio a MDA, como por exemplo, AndromDA³, RAPDIS (Morgado, 2007) e Odyssey-MDA (Werner, 2005). Já, a atividade “Executar os casos de teste no software” foi mantida na íntegra no esboço do processo pois também pertence ao contexto de desenvolvimento baseado em modelos. Observou-se a necessidade de modificar a atividade “Adaptar o software” pois essa atividade no PARFAIT/EA é necessária para implementar manualmente os requisitos e regras de negócios do sistema não cobertos pela linguagem de padrões e, conseqüentemente, não fornecidas pelo *framework*. No caso do esboço do processo, as partes dos modelos que não são transformadas corretamente em código fonte devido a deficiência da ferramenta de apoio, deverão ser implementadas manualmente.

Finalmente, para a definição das atividades da **fase de Transição** do esboço do processo proposto, todas as atividades dessa fase do processo PARFAIT/EA foram

³ <http://www.andromda.org/>

mantidas na íntegra pois também pertencem ao contexto de desenvolvimento baseado em modelos.

5. Conclusões e Trabalhos Futuros

Este trabalho apresentou a análise realizada para mudança de tecnologia de desenvolvimento baseado em *framework* para desenvolvimento orientado por modelos, resultando na proposição do esboço de um processo ágil de desenvolvimento orientado por modelos. A análise foi feita tomando como base a documentação detalhada das atividades e passos de um processo existente de desenvolvimento baseado em *framework*, denominado PARFAIT/EA. Cada atividade e passo foram analisados sob a perspectiva de reutilização, quando possível.

Observou-se que as atividades do PARFAIT/EA independentes da tecnologia foram totalmente reutilizadas. Aquelas parcialmente dependentes de tecnologia (no caso, *frameworks*) foram modificadas e as totalmente dependentes foram removidas. Em contrapartida, atividades específicas para a criação de modelos do domínio de interesse e para o reuso de tais modelos foram adicionadas.

Além das atividades, a documentação do esboço do processo proposto será composta de passos, reuniões e marcos de referência, como no processo PARFAIT/EA. Para permitir isso, esses elementos estabelecidos pelo processo PARFAIT/EA serão adaptados para o contexto de desenvolvimento orientado por modelos. Para definir completamente o processo ágil de desenvolvimento orientado por modelos, estudos de caso planejados, de acordo com Wholin *et al.* (2000), serão conduzidos para refinar o esboço apresentado neste trabalho, bem como observar os pontos positivos e/ou negativos obtidos. Pretende-se utilizar a ferramenta PDocTool como apoio a documentação do processo e ao planejamento e execução dos projetos.

6. Referências

- ABRAHAMSSON, P.; SALO, O.; RONKAINEN, J.; WARSTA, J. Agile Software Development Methods: Review and Analysis. ESPOO (Technical Research Centre of Finland), 2002.
- APPLETON, B. Patterns and software: Essential concepts and terminology. URL: <http://www.cmcrossroads.com/bradapp/docs/patterns-intro.html>, 1997.
- BASS, L.; CLEMENTS, P.; KAZMAN, R. Software Architecture in Practice. Addison-Wesley, 2003.
- BECK, K. Extreme programming explained: Embrace change. Second ed. Addison-Wesley, 2000.
- BELIX, J. E. Um Estudo Sobre MDA: Suporte Fornecido pela UML e Reuso de Soluções Pré-Definidas. Tese de Mestrado, Escola Politécnica da Universidade de São Paulo, São Paulo – SP, 2006.
- BETTIN, J. Model Driven Software Development. MDA Journal. Abril, 2004. Disponível em: <http://www.bptrends.com/publicationfiles/04-04%20COL%20MDSD%20Frankel%20-%20Bettin%20-%20Cook.pdf>. Acesso em Março/2007.
- CAGNIN, M. I. PARFAIT: Uma contribuição para Reengenharia de Software baseada em Linguagens de Padrões e Frameworks. Tese de Doutorado, ICMC-USP, São Carlos-SP, 2005.

- FAYAD, M. E., SCHMIDT, D. C. Object-oriented Application frameworks. Communications of the ACM, Vol. 40, 10 p., 1997.
- GOMES, F. D.; CAGNIN, M. I.; MALDONADO, J. C. Esboço de um Processo Ágil de Desenvolvimento baseado em Framework. In: XXXII Conferência Latino-Americana de Informática, Santiago-Chile, 2006.
- GOMES, F. D.; CAGNIN, M. I. Evolução de um Processo Ágil de Desenvolvimento baseado em Framework. In: Jornada Ibero-Americana de Engenharia de Software e Engenharia do Conhecimento, 2007, Lima-Peru. III Jornada Ibero-Americana de Engenharia de Software e Engenharia do Conhecimento. p. 1-10.
- GOMES, F. D. PARFAIT/EA – Processo Ágil de Desenvolvimento baseado em framework. Dissertação apresentada ao Programa de Mestrado, UNIVEM, Marília-SP, 2007.
- JOHNSON, R. E.; FOOTE, B. Designing Reusable Classes. Journal of Object Oriented Programming – JOOP, v. 1, n. 2, p. 22-35, 1998.
- MENDES, A. Arquitetura de Software: desenvolvimento orientado para arquitetura. Rio de Janeiro: Campus, 2002.
- MAYERS, G. J. The art of software testing. 2 ed. Wiley, 2004.
- MORGADO, G. P. RAPDIS: Um Processo e um Ambiente MDA para o Desenvolvimento de Sistema de Informação. Tese de Mestrado, UFRJ, Rio de Janeiro - RJ, 2007.
- MARQUES, F. S. Um Processo Baseado em MDA para a Especialização de Mecanismos de Persistência. Tese de Mestrado, PUC-RJ, Rio de Janeiro - RJ, 2006
- OMG. MDA Guide Version 1.0.1 Disponível em: <http://www.omg.org/docs/omg/03-06-01.pdf>. Acesso em Agosto/2006.
- SILVA, A. R. Abordagem XIS ao Desenvolvimento de Sistema de Informação. IV Conferência da Associação Portuguesa de Sistema de Informação (CAPSI'2003), Porto, 2003.
- SILVA, M.C. COSMOS – Um Modelo de Estruturação de Componentes para Sistemas Orientados a Objetos. Dissertação de Mestrado, IC-Unicamp, Campinas-SP, 2003.
- SOMMERVILLE, I. Engenharia de Software. 6^a ed. São Paulo: Pearson Addison Wesley, 2003.
- SOUSA. M. C. F. Um Processo de Desenvolvimento Baseado em Componentes Adaptado ao Model Driven Architecture. Tese de Mestrado Profissional, UNICAMP, Campinas – SP, 2004.
- WERNER, C. M.; BLOIS, A. P. B.; MAIA, N. E. N. Odyssey-MDA: Uma ferramenta para transformações de modelos UML. In: Sessão de Ferramentas do XIX Simpósio Brasileiro de Engenharia de Software, Uberlândia-MG 2005.
- WHOLIN, C.; RUNESON, P.; HÖST, M.; OHLSSON, M.; REGNELL, B.; WESSLÉN, A. Experimentation in software engineering: An introduction. Kluwer, 2000.