

Um Perfil UML para o Paradigma Orientado a Notificações (PON)

Luciana V. B. Wiecheteck¹, Paulo C. Stadzisz¹, Jean M. Simão¹

lucianavbw@gmail.com, stadzisz, jeansimao {@utfpr.edu.br}

Universidade Tecnológica Federal do Paraná - UTFPR

¹ Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial - CPGEI

Resumo

Este artigo apresenta uma proposta de perfil UML para a modelagem de software utilizando o Paradigma Orientado a Notificações (PON). A UML possui mecanismos de extensão que permitem a adição de novas sintaxe e semântica aos seus elementos visando a modelagem de domínios particulares de aplicação. Um conjunto desses mecanismos de extensão agrupados dentro de um pacote UML é denominado Perfil UML. Por sua vez, o PON se apresenta como uma alternativa aos Paradigmas de Programação Imperativa (PI), incluindo o Paradigma Orientado a Objetos (POO), e aos Paradigmas de Programação Declarativa (PD), propondo-se a eliminar deficiências destes nos aspectos de redundâncias e acoplamento de avaliações causais que impactam no desempenho e paralelismo/distribuição de aplicações. Este novo paradigma tem sido materializado em termos de programação, mas não possuía ainda um método formalizado para orientar os desenvolvedores na elaboração de projetos de software. Portanto, o perfil proposto define um primeiro passo na criação deste método: formalizar os principais conceitos do PON por meio da utilização de mecanismos de extensão - como estereótipos, valores etiquetados e restrições - que melhor adéquam a sua modelagem em UML. A aplicação do perfil UML é ilustrada com exemplos envolvendo a modelagem de um Simulador de Portão Eletrônico. Os resultados demonstram que o perfil criado possui grande valia na definição de programas em PON e visa sua aplicação em processos de projetos de software que utilizam este paradigma de programação.

Palavras chave:

Modelagem de Software, Perfil UML, Paradigma Orientado a Notificações (PON)

Abstract

This article presents a UML Profile for software modeling using the Notification Oriented Paradigm (NOP). UML has extension mechanisms that allow the addition of new syntax and semantic to its elements aiming at software modeling for specific domain applications. A set of these extension mechanisms grouped into a UML package is named UML Profile. NOP presents itself as an alternative to the Imperative Programming (IP) paradigms, such as Object Oriented Paradigm (OOP), as well as to the Declarative Programming (DP) paradigms, with the purpose of eliminating deficiencies of those paradigms concerning to redundancy issues and coupling of causal expressions, which affect the execution performance and parallelism/distribution of applications. This paradigm has already been consolidated in terms of programming but did not possess a formalized method for software design. Thus, the proposed profile defines a first step in the definition of this method: to formalize the main concepts of this paradigm by means of extension mechanisms usage - as stereotypes, tagged values and constraints - that better fit its modeling in the UML. The proposed UML Profile usage is illustrated by means of a modeling example of an electronic gate application. The results showed that the proposed UML profile has an important value in the modeling of NOP programs and can be used in software design processes that use this new programming paradigm.

Keywords:

Software Modeling, UML Profile, Notification Oriented Paradigm (NOP)



Introdução

A capacidade de processamento computacional tem crescido em função da evolução das tecnologias neste contexto [Tanenbaum e Van Steen, 2002]. Entretanto, recursos oferecidos por soluções computacionais modernas, tais como paralelismo e distribuição ou mesmo a utilização da capacidade plena de cada processador, nem sempre são devidamente aproveitados em função de limitações das técnicas de programação [Simão e Stadzisz, 2008, 2009].

Na verdade, técnicas de programação baseadas no estado da arte, como o chamado Paradigma de Programação Orientada a Objetos (POO) ou Sistema Baseados em Regras (SBR), sofrem de limitações intrínsecas de seus paradigmas. Estes paradigmas poderiam ser genericamente classificados como Paradigma Imperativo (PI) e Paradigma Declarativo (PD) que englobam respectivamente o POO e os SBR [Banaszewski, 2009].

Particularmente, estes paradigmas levam ao forte acoplamento de expressões causais e redundâncias decorrentes das suas avaliações. Estas limitações dificultam a execução paralela ou distribuída de programas e frequentemente comprometem o seu desempenho pleno mesmo em sistemas monoprocessados. Assim, existem motivações para buscas de alternativas aos PI e PD, com o objetivo de eliminar ou diminuir as desvantagens deles [Banaszewski et al., 2007][Banaszewski, 2009][Gabbrielli, Martini, 2010][Roy e Haridi, 2004][Simão e Stadzisz, 2008, 2009].

Neste âmbito, uma alternativa é o Paradigma Orientado a Notificações (PON). O PON foi concebido a partir de uma teoria de Controle Discreto e Inferência [Simão, 2005][Simão e Stadzisz, 2008, 2009][Simão, Stadzisz e Tacla, 2009]. Ele se propõe a eliminar algumas das deficiências dos atuais paradigmas em relação a avaliações causais desnecessárias e acopladas, evitando o processo de inferência monolítico baseado em pesquisas por meio de um mecanismo baseado no relacionamento de entidades computacionais notificantes [Banaszewski et al., 2007][Banaszewski, 2009][Simão e Stadzisz, 2008, 2009].

Atualmente, o PON tende a se consolidar em termos de programação. Entretanto, não há um método formalizado para orientar os desenvolvedores na elaboração de projetos de software baseados em PON. Embora seja possível o emprego convencional da UML (Unified Modeling Language) e processos conhecidos no desenvolvimento de software com o PON, suas especialidades motivam o uso mais particularizado de modelagem e processos de software.

A UML fornece um conjunto rico de conceitos de modelagem e notações que foram desenvolvidos para satisfazer as necessidades de projetos típicos de software. No entanto, há casos em que são necessárias características adicionais de modelagem, além daquelas definidas pela UML padrão, como no caso de domínios particulares de aplicação (ex: tempo-real) ou implementação de tecnologias (ex: CORBA).

A UML permite a modelagem dessas necessidades específicas por meio dos seus mecanismos de extensão, que permitem a adição de novos elementos de modelagem ao metamodelo da UML. Oportunamente, um metamodelo define e descreve a estrutura e semântica de modelos, no caso, modelos UML. Um conjunto desses mecanismos de extensão, agrupados dentro de um pacote UML estereotipado <<profile>>, forma um Perfil UML [OMG, 2003] [Fuentes e Vallecillo, 2004]. Atualmente, existem vários perfis UML disponíveis para uso público, sendo que alguns têm sido adotados e padronizados pela OMG, como o Perfil UML CCM (CORBA Component Model) [OMG, 2005a] e o Perfil UML para Scheduling, Performance e Time [OMG, 2005b].

Neste contexto, este artigo apresenta um perfil UML por meio da especificação de mecanismos de extensão da UML - como estereótipos, valores etiquetados e restrições - que definem os principais conceitos do PON, melhor adequando sua modelagem em UML. O perfil UML criado para o PON é denominado Perfil PON (NOP Profile em inglês) e particulariza o metamodelo da UML a fim de suprir as necessidades específicas deste novo paradigma de programação.

O artigo está organizado da seguinte maneira: a seção 2 apresenta o Paradigma Orientado a Notificações (PON). A seção 3 descreve sucintamente a UML e seus mecanismos de extensão. A seção 4 apresenta o perfil UML criado para o PON. A seção 5 ilustra um exemplo de uso do perfil UML desenvolvido. E, por fim, na seção 6 são apresentadas as conclusões do trabalho.

Conclusões

O Paradigma Orientado a Notificações (PON) tem sido materializado em termos de programação, mas não possuía ainda um método formalizado para orientar os desenvolvedores na elaboração de projetos de software. Portanto, como primeiro passo na construção deste método, o presente artigo apresentou o NOP Profile, um perfil UML criado para o PON. O NOP Profile tem por objetivo propor convenções na utilização da UML para modelagem de software neste novo paradigma. A aplicação do perfil UML foi ilustrada com exemplos envolvendo a modelagem de um simulador de Portão Eletrônico.

A extensão proposta é suportada pela UML e, portanto, pode ser facilmente integrada em ferramentas CASE que suportam a UML. O NOP Profile está sendo avaliado por meio da sua aplicação em diferentes projetos de software. Adicionalmente, os modelos UML criados, que utilizam o NOP Profile, podem ser verificados quanto à sua sintaxe por meio de funcionalidades padrões disponíveis nas ferramentas CASE que validam as restrições OCL definidas no perfil. Assim, foi apresentada uma primeira versão do NOP Profile que pode vir a ter evoluções conforme ele seja aplicado nos diferentes projetos e sejam identificadas novas necessidades.

Com base nos resultados produzidos, os autores julgam que o emprego de um Perfil UML para fins de meta-modelagem não só é viável quanto representa uma abordagem muito prática e alinhada com a linguagem UML. O perfil criado possui grande valia na definição de programas em PON e visa sua aplicação em processos de projetos com esse paradigma de programação. Portanto, o próximo passo é a aplicação do NOP Profile em um processo de software. O objetivo é criar um método que será empregado em projetos de software baseados em PON. Para tanto, o NOP Profile deve ser aplicado nas fases de modelagem, viabilizando uma modelagem consistente com os conceitos do framework do PON.

Referências

- [1] [Banaszewski et al., 2007] Banaszewski, R. F.; Stadzisz, P. C.; Tacla, C. A.; Simão, J. M. "Notification Oriented Paradigm (NOP): A software development approach based on artificial intelligence concepts," in Proceedings of the VI Congress of LAPTEC, Santos, Brazil, 2007.
- [2] [Banaszewski, 2009] Banaszewski, R. F. "Paradigma Orientado a Notificações: Avanços e Comparações". Dissertação de Mestrado, CPGEI/UTFPR, Curitiba, 2009. Disponível em: http://arquivos.cpgei.ct.utfpr.edu.br/Ano_2009/dissertacoes/Dissertacao_500_2009.pdf.
- [3] [Döll, 2002] Döll, L. M. (2002). Proposta de uma metodologia para a modelagem da dinâmica de sistemas orientados a objetos usando redes de petri predicado/transição. Dissertação de mestrado, CEFET-PR.
- [4] [Fuentes e Vallecillo, 2004] Fuentes, L., Vallecillo, A.(2004). An Introduction to UML Profiles. The European Journal for the Informatics Professional, vol. 5, nº2.
- [5] [Gabbrielli e Martini, 2010] Gabbrielli, M., Martini, S. "Programming Languages: Principles and
- [6] Paradigms. Series: Undergraduate Topics in Computer Science". 1st Edition, 2010, XIX, 440 p., Softcover. ISBN: 978-1-84882-913-8.
- [7] [Guedes, 2004] Guedes, G. T. (2004). UML Uma abordagem Prática. Novatec.
- [8] [Lima, 2008] Lima, A. S. (2008). UML 2.0: do requisito à solução. São Paulo, Editora Érica, 3ªed.
- [9] [OMG, 2003] OMG Unified Modeling Language Specification (2003). Version 1.5, OMG document formal/03-03-01.

- 
- [10] [OMG, 2005a] UML Profile for CORBA Components (2005). Version 1.0, OMG document formal/05-07-06.
- [11] [OMG, 2005b] UML Profile for Schedulability, Performance, and Time Specification (2005). Version 1.1, OMG document formal/05-01-02.
- [12] [Roy e Haridi, 2004] Roy, P. V., Haridi, S. "Concepts, Techniques, and Models of Computer Programming". MIT Press, 2004.
- [13] [Simão, 2005] Simão, J. M. "A Contribution to the Development of a HMS Simulation Tool and Proposition of a Meta-Model for Holonic Control". Tese de Doutorado, CPGEI/UTFPR, Curitiba, 2005. Disponível em: http://arquivos.cpgei.ct.utfpr.edu.br/Ano_2005/teses/Tese_012_2005.pdf.
- [14] [Simão e Stadzisz, 2008] Simão, J. M.; Stadzisz, P. C. "Paradigma Orientado a Notificações (PON) - Uma Técnica de Composição e Execução de Software Orientado a Notificações". Pedido de Patente submetida ao INPI/Brazil (Instituto Nacional de Propriedade Industrial) em 2008 e a Agência de Inovação/UTFPR em 2007. Nº INPI Provisório 015080004262. Nº INPI Efetivo PI0805518-1. Patente submetida ao INPI. Brasil, 2008.
- [15] [Simão e Stadzisz, 2009] Simão, J. M.; Stadzisz, P. C. "Inference Process Based on Notifications: The Kernel of a Holonic Inference Meta-Model Applied to Control Issues". IEEE Transactions on Systems, Man and Cybernetics. Part A, Systems and Humans, Vol. 39, Issue 1, 238-250, Digital Object Identifier 10.1109/TSMCA.2008.20066371, 2009.
- [16] [Simão, Stadzisz e Tacla, 2009] J. M. Simão, C. A. Tacla, and P. C. Stadzisz, "Holonic Control Meta-Model". IEEE Transactions on Systems Man & Cybernetics, Part A. Vol. 39, NO. 5, September 2009 Pg. 1126-1139.
- [17] [Tanenbaum e Van Steen, 2002] Tanenbaum, A.S.; Van Steen, M. "Distributed Systems: Principles and Paradigms", Prentice Hall, 2002.