

# Comparações quantitativas e qualitativas entre o Paradigma Orientado a Objetos e o Paradigma Orientado a Notificações sobre um simulador de jogo

Adriano F. Ronszcka<sup>1</sup>, Danillo L. Belmonte<sup>1</sup>, Glauber Z. Valença<sup>2</sup>, Márcio V. Batista<sup>1</sup>, Robson R. Linhares<sup>1,2</sup>, Cesar A. Tacla<sup>1,2</sup>, Paulo C. Stadzisz<sup>1,2</sup>, Jean M. Simão<sup>1,2</sup>

Universidade Tecnológica Federal do Paraná - UTFPR

<sup>1</sup> Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial - CPGEI

<sup>2</sup> Programa de Pós-Graduação em Computação Aplicada - PPGCA

ronszcka, belmontepg, gvalencio, marcio.venancio {@gmail.com}, robson@dainf.ct.utfpr.edu.br, tacla, stadzisz, jeansimao {@utfpr.edu.br}

## Resumo

Este artigo apresenta uma revisão dos conceitos relacionados ao Paradigma Orientado a Notificações (PON) e uma comparação, qualitativa e quantitativa, de duas versões de uma mesma aplicação. A primeira desenvolvida de acordo com os princípios do Paradigma Orientado a Objetos (POO) e a segunda inclusive de acordo com os princípios do PON. O PON se apresenta como uma alternativa aos Paradigmas de Programação Imperativa (PI), incluindo o POO, e aos Paradigmas de Programação Declarativa (PD), propondo-se a eliminar deficiências destes no que tange a aspectos de redundâncias e acoplamento de avaliações causais que impactam no desempenho e paralelismo/distribuição de aplicações. A comparação apresentada neste artigo abrange particularmente questões relacionadas à implementação e ao desempenho. O experimento demonstra que, embora o desempenho do PON tenha sido um tanto inferior ao do POO para a aplicação desenvolvida, em função de características da aplicação e de um ambiente de execução ainda não totalmente adaptado ao paradigma, existem aspectos relativos à maneira como se concebe os programas em PON que podem ser levados em consideração e incentivar a utilização deste em aplicações com requisitos de paralelismo ou distribuição.

## Palavras chave:

Paradigma Orientado a Notificações (PON), Simulador de jogo em POO e PON, Comparações entre POO e PON.

## Abstract

This paper presents a review of the concepts related to the Notification-Oriented Paradigm (NOP) and a qualitative and quantitative comparison of two versions of a certain application. The first developed according to the Object-Oriented Paradigm (OOP) principles and the second developed according to the NOP principles. The NOP presents itself as an alternative to the Imperative Programming (IP) paradigms, such as 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 evaluations, which affect the execution performance and parallelism/distribution of applications. The presented comparison includes issues related to implementation and relative performance of the applications. The experiment demonstrates that, even though the NOP performance is somewhat worse than OOP performance for the developed application, due to application characteristics and a runtime environment not completely adapted to NOP, there are some aspects related to the manner with the programs are designed in NOP which can be taken into consideration and encourage the use of NOP on applications requiring parallelism and distribution.

## Keywords:

Notification-Oriented Paradigm (NOP), Game simulator in OOP and NOP, Comparisons between OOP and 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 os Sistemas 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, 2001, 2005; Simão e Stadzisz, 2002, 2008, 2009; Simão, Stadzisz e Tacla, 2009; Simão, Stadzisz e Künzle, 2003]. 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].

Neste artigo são apresentados resultados de comparações efetuadas entre uma implementação segundo o POO de um simulador de um jogo similar ao conhecido Pacman<sup>1</sup> e outra implementação deste mesmo simulador segundo o PON. A análise dos resultados tem como foco primário os aspectos de desempenho. Não obstante, também é objetivo do artigo refletir sobre a aplicabilidade do PON e as potencialidades em relação ao desenvolvimento de técnicas e ferramentas voltadas para este paradigma.

Este artigo está organizado como segue: a Seção 2 sucintamente reflete o estado da arte em paradigmas de programação. A Seção 3 apresenta o PON. A Seção 4 apresenta o caso de estudo, com os detalhes de implementação do simulador do jogo. A Seção 5 apresenta a discussão sobre os experimentos. A Seção 6, por fim, apresenta as conclusões e perspectivas de trabalhos.

## Conclusões

Apesar dos resultados com a versão do simulador concebido em POO apresentar melhor desempenho do que as versões do simulador concebidas em PON, tais resultados não são suficientes para julgar o POO superior em relação ao PON. Além disso, alguns fatores contribuíram para essa discrepância entre os resultados, tais como a falta de um compilador próprio para o PON, a falta de técnicas concretas de modelagem visando à facilidade de composição de programas neste paradigma e a falta de uma versão do simulador desenvolvida completamente sobre os conceitos do PON.

Conforme citado na Seção 3, o PON atualmente está materializado na forma de um framework/wizard desenvolvido sobre a linguagem C++. Apesar de não ser o cenário ideal, o PON tem sido frequentemente otimizado e com isso, ganhos de desempenho consideráveis vem sendo alcançados. Definitivamente, uma linguagem de programação e um compilador devem ser desenvolvidos para que o PON alcance todo o seu potencial.

<sup>1</sup> Os créditos do jogo e seus direitos autorais pertencem ao indivíduo que o produziu ou a empresa que o publicou. A utilização neste trabalho visa apenas o estudo acadêmico, sem fins lucrativos. Portanto, qualquer outro uso para este conteúdo, poderá estar violando os direitos do autor.

Observou-se que a hibridização de paradigmas e, principalmente, a conversão de estruturas de decisão (i.e. ifs) do POO para Rules do PON, não foram suficientes para manter o fluxo de execução dos programas operando de maneira semelhante. Tal fato implicou na criação de Rules adicionais para a coordenação do fluxo de notificações do PON, de maneira a operar de maneira sequencial como no POO.

De fato, esta alternativa se mostrou inviável e indesejável, uma vez que impactou diretamente no desempenho das aplicações. Certamente, o simulador deveria ser totalmente repensado e refeito em PON, partindo da remodelagem totalmente baseada em notificações, uma vez que a mudança de paradigma implica na mudança no estilo de conceber programas.

Em trabalhos futuros, uma versão completa do simulador, com todas as classes desenvolvidas sobre os princípios do PON, deve ser desenvolvida, possivelmente aplicando técnicas de Inteligência Artificial para a criação automática de regras. Ainda, o simulador apresenta características interessantes que permitem explorar a distribuição ou paralelização dos elementos PON, visando um melhor aproveitamento do processamento dos computadores disponíveis atualmente.

## 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](http://arquivos.cpgei.ct.utfpr.edu.br/Ano_2009/dissertacoes/Dissertacao_500_2009.pdf).
- [3] [Brookshear, 2006] Brookshear, J. G. "Computer Science: An Overview". Addison Wesley, 2006.
- [4] [Forgy, 1982] Forgy, C. L. "RETE: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem", Artificial Intelligence N. 19, pg 17-37 - North- Holland, 1982.
- [5] [Friedman-Hill, 2003] Friedman-Hill, E. Jess in Action: Rule Based System in Java. Greenwich, CT, USA: Manning Publications Co, 2003.
- [6] [Gabbrielli e Martini, 2010] Gabbrielli, M., Martini, S. "Programming Languages: Principles and Paradigms. Series: Undergraduate Topics in Computer Science". 1st Edition, 2010, XIX, 440 p., Softcover. ISBN: 978-1-84882-913-8.
- [7] [Pittman, 2011] Pittman, J. "The Pac-Man Dossier". Disponível em: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>. Acessado em: 16 de Junho de 2011.
- [8] [Roy e Haridi, 2004] Roy, P. V., Haridi, S. "Concepts, Techniques, and Models of Computer Programming". MIT Press, 2004.
- [9] [Scott, 2000] Scott, M. L. "Programming Language Pragmatics", 2º Edition, p. 8, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc, 2000.
- [10] [Simão, 2001] Simão, J. M. "Proposta de uma Arquitetura de Controle para Sistemas Flexíveis de Manufatura Baseada em Regras e Agentes". Dissertação de Mestrado, CPGEI/UTFPR, Curitiba, 2001.
- [11] [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](http://arquivos.cpgei.ct.utfpr.edu.br/Ano_2005/teses/Tese_012_2005.pdf).

- 
- [12] [Simão e Stadzisz, 2002] Simão, J. M.; Stadzisz, P. C. "An Agent-Oriented Inference Engine applied for Supervisory Control of Automated Manufacturing Systems". In: J. Abe & J. Silva Filho, *Advances in Logic, Artificial Intelligence and Robotics* (Vol. 85, pp. 234-241). Amsterdam, The Netherlands: IOS Press Books, 2002.
- [13] [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. N° INPI Provisório 015080004262. N° INPI Efetivo PI0805518-1. Patente submetida ao INPI. Brasil, 2008.
- [14] [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.
- [15] [Simão, Stadzisz e Künzle, 2003] Simão, J. M.; Stadzisz, P. C.; Künzle, L. "Rule and Agent-oriented Architecture to Discrete Control Applied as Petri Net Player". 4th Congress of Logic Applied to Technology, LAPTEC, 101, p. 217, 2003.
- [16] [Simão, Stadzisz e Tacla, 2009] Simão, J. M., Stadzisz, P. C.; Tacla, C. A. "Holonic Control Meta-Model". *IEEE Transactions on Systems, Man and Cybernetics. Part A, Systems and Humans*, 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.