

# Plano de Curso - Doutorado em Ciência da Computação

## Aspectos para Implementação de Requisitos Não-Funcionais

**Cristiano Amaral Maffort**

cristiano@decom.cefetmg.br

**Orientadora: Mariza Andrade da Silva Bigonha**

mariza@dcc.ufmg.br

**Co-orientador: Marco Túlio de Oliveira Valente**

mtov@pucminas.br

### 1 Introdução

Os modelos atuais para desenvolvimento de software são baseados principalmente em dois paradigmas: procedural e orientado por objetos. Tanto o paradigma procedural quanto o paradigma orientado por objetos não são completamente eficientes para modelar determinadas decisões de projeto que um programa deve implementar. No paradigma procedural, a modularização é restrita à implementação de abstrações dos procedimentos necessários à realização de uma tarefa. Já a programação orientada por objetos fornece um nível mais elevado de abstração, permitindo representar tipos abstratos de dados.

Em ambos os paradigmas, os requisitos que compõem um sistema podem ser classificados como funcionais e não-funcionais. Requisitos funcionais constituem as funcionalidades que se espera que o sistema disponibilize. Já os requisitos não-funcionais compreendem elementos específicos de projeto, tais como restrições nas quais o sistema deve operar ou propriedades do sistema (como distribuição, *logging*, segurança e persistência). Via de regra, orientação por objetos permite modularizar de forma eficiente requisitos funcionais. No entanto, não é capaz de modularizar adequadamente requisitos que não são mapeáveis diretamente em uma ou poucas classes de um sistema [5]. Esses requisitos que se espalham e se entrelaçam pelo código funcional da aplicação são denominados requisitos transversais (*crosscutting concerns*). Requisitos não-funcionais constituem o tipo mais comum de requisito transversal [13].

O problema da modularização de requisitos transversais pode ser classificado em duas categorias: espalhamento e entrelaçamento de código [6, 5]. O espalhamento (*scattering*) ocorre quando o código para implementação do requisito transversal fica disperso no programa, ou seja, fica espalhado em múltiplos módulos do sistema. Já o entrelaçamento (*tangling*) ocorre quando um módulo implementa simultaneamente múltiplos requisitos. Esses dois problemas dificultam a reusabilidade e a manutenibilidade do código de um programa, pois implicam em baixa coesão modular e alto grau de acoplamento entre módulos, comprometendo dessa forma a qualidade interna, a legibilidade e a evolução da aplicação [13].

Programação Orientada por Aspectos (AOP) [5, 4] propõe uma nova abstração, denominada aspecto, que tem por objetivo modularizar decisões de projeto que não podem ser adequadamente definidas por meio de programação orientada por objetos. Para tanto, AOP fornece mecanismos de composição que permitem descrever de forma mo-

dular propriedades que se encontram espalhadas e entrelaçadas em vários pontos de um sistema orientado por objetos. Além disso, AOP fornece mecanismos para instrumentação de código, permitindo inclusive alterar a estrutura interna das classes de um sistema.

Na presente proposta de trabalho de doutorado, pretende-se investigar o uso de aspectos na modularização de diversos requisitos não- funcionais (como distribuição, persistência, autenticação e autorização), os quais estão presentes na maioria dos projetos de desenvolvimento de sistemas. Além disso, pretende-se avaliar o uso de aspectos em novos domínios de aplicação, como computação móvel e sistemas de informação geográficos.

O restante deste plano de curso está organizado conforme descrito a seguir. A Seção 2 descreve algumas das abordagens já propostas para lidar com modularização de requisitos não-funcionais. A Seção 3 apresenta a experiência prévia do candidato na área de pesquisa proposta. A Seção 4 descreve a proposta de trabalho a ser realizado neste curso de doutorado. A Seção 5 apresenta um cronograma de atividades a serem realizadas durante o curso. E, por fim, a Seção 6 apresenta as principais motivações em relação ao curso de doutorado.

## 2 Trabalhos Relacionados

Diversos projetos de pesquisa têm investigado o uso de aspectos para modularizar serviços transversais. Por exemplo, já foram desenvolvidas ferramentas baseadas em aspectos para auxiliar na implementação de aplicações EJB, tais como Gotech [12] e AspectJ2EE [2]. Gotech é um *framework* que gera aspectos capazes de transformar classes Java em componentes EJB. AspectJ2EE é uma implementação orientada por aspectos de um *container* que disponibiliza serviços não-funcionais típicos de servidores EJB.

Soares, Borba e Laureano descrevem uma experiência bem sucedida na reestruturação de uma aplicação usando AspectJ [11]. Nesse trabalho, os autores mostram como os requisitos de distribuição e persistência foram modularizados usando aspectos implementados em AspectJ.

Aspectos já foram empregados também na implementação de aplicações baseadas em Serviços Web. Por exemplo, Courbis e Finkelstein propõem o uso de aspectos dinâmicos para customizar e adaptar aplicações usadas para coordenar a execução de chamadas a serviços Web [3]. Já WSML [14] é um *middleware* que utiliza aspectos para encapsular diversos requisitos transversais típicos de aplicações clientes de serviços Web, incluindo autenticação, *logging*, monitoramento, transações etc.

Outros trabalhos têm como objetivo incorporar abstrações para programação distribuída em linguagens orientadas por aspectos. DJcutter [10] é uma linguagem que acrescenta em AspectJ o conceito de *pointcuts* remotos, os quais viabilizam a captura de pontos de junção pertencentes à execução de programas em máquinas remotas. GluonJ [1] é uma extensão de AspectJ com suporte a injeção de dependência.

## 3 Experiência na Área de Pesquisa Proposta

No curso de mestrado, com dissertação intitulada “Aspectos para Construção de Aplicações Distribuídas”, investiguei o uso de aspectos para modularização do requisito de distribuição provido por duas plataformas de *middleware* nativas de ambientes de desenvolvimento Java: Java RMI e Java IDL. Como resultado, foi proposta uma solução para isolar a funcionalidade de distribuição da lógica de negócio de aplicações implementadas em Java. Além disso, foi elaborada uma ferramenta que produz todos

os artefatos necessários para introdução do requisito de distribuição em uma aplicação, tornando automática a adaptação e instanciação da solução proposta no trabalho de mestrado.

**Publicações e Premiações:** Os primeiros resultados do trabalho desenvolvido no curso de mestrado foram publicados em um artigo aceito no XX Simpósio Brasileiro de Engenharia de Software (SBES 2006), o qual foi classificado como quinto melhor artigo do evento [7]. Além disso, estudos preliminares para adição de suporte a Serviços Web foram apresentados no III Workshop Brasileiro de Desenvolvimento de Software Orientado a Aspectos [8]. Resultados consolidados do trabalho, incluindo três estudos de caso, foram publicados em um artigo no *Journal of the Brazilian Computer Society* [9].

Por fim, a dissertação de mestrado foi classificada em terceiro lugar no Concurso de Teses e Dissertações (CTD), realizado em julho de 2008, durante o Congresso da SBC.

#### 4 Plano de Trabalho

No trabalho de doutorado, tendo em vista os resultados alcançados no meu trabalho de mestrado, pretendo continuar a investigação sobre ferramentas, linguagens e métodos para implementação de sistemas orientados por aspectos. Particularmente, pretendo ampliar os interesses e aplicações investigadas no meu trabalho de mestrado. A seguir são descritas possíveis linhas para a pesquisa de doutorado proposta:

- Investigar a viabilidade de introduzir, na solução proposta, suporte a outras plataformas de *middleware* de comunicação. Inicialmente, pretendo introduzir suporte a Serviços Web, tornando a solução compatível com as três principais plataformas de *middleware* para sistemas desenvolvidos em ambientes Java.
- Investigar o uso de aspectos para modularizar outros requisitos não-funcionais (além de distribuição). Como exemplo desses requisitos, pode-se citar: persistência, autenticação, autorização, *logging* e controle de transações. Um item interessante a ser considerado nessa linha de pesquisa, é investigar a inter-relação entre esses requisitos, de forma que eles sejam plenamente compatíveis entre si, possibilitando criar módulos de requisitos que possam ser dinamicamente acoplados em uma aplicação alvo.
- Investigar o uso de aspectos em novos domínios de aplicações. Inicialmente, pretende-se investigar o uso de aspectos em requisitos que são peculiares a ambientes de computação móvel, como adaptação dinâmica, gerenciamento de memória, controle de desempenho, usabilidade e gerenciamento de energia.

#### 5 Cronograma

As atividades previstas para o doutorado foram divididas anualmente conforme a seguir:

**1º Ano:** Cumprimentos dos créditos, sendo que pretendo solicitar o reaproveitamento de 16 créditos referentes às seguintes disciplinas cursadas no mestrado do Programa de Pós-graduação em Informática da PUC Minas: (i) Tópicos em Computação Distribuída - Projeto e Implementação de Sistemas Distribuídos (4 créditos); (ii) Projeto e Análise de Algoritmos (4 créditos); (iii) Redes de Computadores (4 créditos); (iv) Fundamentos Teóricos da Computação (4 créditos).

Pretendo também solicitar reaproveitamento de 4 créditos referentes à disciplina Tópicos em Compiladores - Otimização de Código, cursada como disciplina isolada no Departamento de Ciência da Computação da UFMG no primeiro semestre de 2008.

Assim, as disciplinas que pretendo cursar no curso de doutorado são as seguintes:

- 2009/2: (i) Engenharia de Software (4 créditos); (ii) Sistemas Operacionais (4 créditos); e, (iii) Estágio em Docência I (2 créditos).
- 2010/1: (i) Arquitetura de Computadores (4 créditos); (ii) Banco de Dados (4 créditos); (iii) Semântica Formal (4 créditos); e, (iv) Estágio em Docência II (2 créditos).

Desta forma, ao final do primeiro ano de curso, pretendo cumprir integralmente a carga de disciplinas requerida.

**2º Ano:** Realização do exame de qualificação durante o 2º semestre de 2010. Elaboração e defesa da proposta de doutorado.

**3º Ano:** Neste terceiro ano do curso, pretendo realizar um doutorado-sanduíche no exterior. Os possíveis pesquisadores com os quais pretendo trabalhar são os seguintes: Prof. Geoff Coulson (Lancaster University), Prof. Eric Wohlstadter (University of British Columbia) e Prof. Eric Eide (University of Utah).

**4º Ano:** Finalização da pesquisa, elaboração do texto e defesa da tese. Pretende-se publicar os resultados da pesquisa em pelo menos uma conferência e em um periódico de grande prestígio internacional.

## 6 Motivação

Atualmente, sou professor efetivo do Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG). Nesta Instituição há um Plano de Desenvolvimento Institucional (PDI) que incentiva fortemente a capacitação acadêmica, técnica e profissional de seu corpo docente. Dentre esses incentivos pode-se citar a redução parcial ou total dos encargos didáticos para professores regularmente matriculados em programas de doutorado. Além disso, o CEFET-MG possui alguns programas próprios de pós-graduação, nos quais o docente com título de doutor tem maior probabilidade de aprovação das suas propostas de pesquisa. Assim, o curso de doutorado representa, pra mim, uma possibilidade de ascensão acadêmica e profissional, através do desenvolvimento de conhecimentos e habilidades intrinsecamente relacionadas à carreira docente e à pesquisa científica. Conforme afirmado, esse objetivo é plenamente apoiado pelas políticas institucionais do CEFET-MG, que busca garantir educação de qualidade a seus alunos, assim como produzir conhecimento e inovação tecnológica.

## Referências

- [1] Shigeru Chiba and Rei Ishikawa. Aspect-oriented programming beyond dependency injection. In *19th European Conference on Object-Oriented Programming*, volume 3586 of *Lecture Notes in Computer Science*, pages 121–143. Springer, 2005.

- [2] Tal Cohen and Joseph Gil. AspectJ2EE = AOP + J2EE. In *18th European Conference on Object-Oriented Programming*, volume 3086 of *LNCS*, pages 219–243. Springer-Verlag, 2004.
- [3] Carine Courbis and Anthony Finkelstein. Towards aspect weaving applications. In *27th International Conference on Software Engineering*, pages 69–77, October 2005.
- [4] Gregor Kiczales, Erik Hilsdale, Jim Hugunin, Mik Kersten, Jeffrey Palm, and William G. Griswold. An overview of AspectJ. In *15th European Conference on Object-Oriented Programming*, volume 2072 of *LNCS*, pages 327–355. Springer Verlag, 2001.
- [5] Gregor Kiczales, John Lamping, Anurag Mendhekar, Chris Maeda, Cristina Lopes, Jean-Marc Loingtier, and John Irwin. Aspect-Oriented Programming. In *11th European Conference Object-Oriented Programming*, volume 1241 of *LNCS*, pages 220–242. Springer Verlag, 1997.
- [6] Ramnivas Laddad. *AspectJ in Action: Practical Aspect-Oriented Programming*. Manning, Greenwich, USA, 2003.
- [7] Cristiano Amaral Maffort and Marco Túlio de Oliveira Valente. Aspectos para construção de aplicações distribuídas. In *XX Simpósio Brasileiro de Engenharia de Software*, October 2006.
- [8] Cristiano Amaral Maffort and Marco Túlio de Oliveira Valente. Aspectos para construção de serviços web. In *III Workshop Brasileiro de Desenvolvimento de Software Orientado a Aspectos*, October 2006.
- [9] Cristiano Amaral Maffort and Marco Túlio Oliveira Valente. Modularizing communication middleware concerns using aspects. *Journal of the Brazilian Computer Society*, 13(4):81–95, 2007.
- [10] Muga Nishizawa, Shigeru Chiba, and Michiaki Tatsubori. Remote Pointcut: A Language Construct for Distributed AOP. In *3rd International Conference on Aspect-Oriented Software Development*, pages 7–15. ACM Press, 2004.
- [11] Sergio Soares, Paulo Borba, and Eduardo Laureano. Distribution and persistence as aspects. *Software Practice and Experience*, 36(7):711–759, 2006.
- [12] Eli Tilevich, Stephan Urbanski, Yannis Smaragdakis, and Marc Fleury. Aspectizing server-side distribution. In *Automated Software Engineering Conference*, pages 130–141. IEEE Press, October 2003.
- [13] Fabio Tirelo, Roberto da Silva Bigonha, Mariza da Silva Bigonha, and Marco Túlio de Oliveira Valente. Desenvolvimento de Software Orientado por Aspectos. XXIII Jornada de Atualização em Informática, XXIV Congresso da Sociedade Brasileira de Computação, August 2004.
- [14] B. Verheecke, W. Vanderperren, and V. Jonckers. Unraveling crosscutting concerns in web services middleware. *IEEE Software*, 23(1):42–50, January 2006.

## **Assinaturas**

Aluno: \_\_\_\_\_  
Cristiano Amaral Maffort

Orientadora: \_\_\_\_\_  
Profa. Dra. Mariza Andrade da Silva Bigonha