



**Fabio Seixas Marques**

**Um Processo Baseado em MDA para a Especialização de  
Mecanismos de Persistência**

PUC-Rio - Certificação Digital Nº 0310827/CC

**Dissertação de Mestrado**

Dissertação apresentada como requisito parcial para  
obtenção do título de Mestre pelo Programa de Pós-  
Graduação em Informática da PUC-Rio.

Orientador: Carlos José Pereira de Lucena  
Co-Orientador: Ivan Mathias Filho

Rio de Janeiro, abril de 2006



**Fabio Seixas Marques**

## **Um Processo Baseado em MDA para a Especialização de Mecanismos de Persistência**

Dissertação apresentada como requisito parcial para obtenção do título de Mestre pelo Programa de Pós-Graduação em Informática da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

**Prof. Carlos José Pereira de Lucena**

Orientador  
Departamento de Informática - PUC-Rio

**Prof. Ivan Mathias Filho**

Co-Orientador  
Departamento de Informática - PUC-Rio

**Prof. Arndt von Staa**

Departamento de Informática - PUC-Rio

**Prof. Ricardo Choren Noya**

Seção de Engenharia de Computação - IME

**Prof. José Eugênio Leal**

Coordenador Setorial do Centro Técnico Científico - PUC-Rio

Rio de Janeiro, 10 de abril de 2006

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

### **Fabio Seixas Marques**

Graduou-se em Bacharelado em Informática pela Pontifícia Universidade Católica do Rio de Janeiro no ano de 2002. É funcionário da SAU (Sistemas de Administração Universitária) da PUC-Rio.

#### Ficha Catalográfica

Marques, Fabio Seixas

Um processo baseado em MDA para a especialização de mecanismos de persistência / Fabio Seixas Marques ; orientadores: Carlos José Pereira de Lucena, Ivan Mathias Filho. – Rio de Janeiro : PUC-Rio, Departamento de Informática, 2006.

88 f. : il. (col.) ; 29,7 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui referências bibliográficas.

1. Informática – Teses. 2. Engenharia de software. 3. MDA. 4. Transformações. 5. Persistência. 6. RDL. I. Lucena, Carlos José Pereira de. II. Mathias Filho, Ivan. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Para minha noiva, Ana Carina,  
meus pais, Antonio e Maria Lucia,  
e meu irmão, Junior,  
por todo apoio e amor nessa longa jornada.

## Agradecimentos

A minha noiva, Ana Carina Barros Queiroz, por todo apoio, amor e carinho incondicional durante todos esses anos de dedicação. Espero um dia conseguir retribuir.

Aos meus pais, Antonio e Maria Lucia, que sempre fizeram de tudo por mim e me criaram com tanto amor e carinho.

Ao meu irmão, Junior, por todo companheirismo e ajuda desde que éramos crianças.

Aos meus avós, por todos os lanchinhos feitos com tanto amor.

Ao meu orientador, Carlos José Pereira de Lucena, pela confiança e pelo apoio que me deu durante todo este longo caminho.

Ao meu co-orientador, Ivan Mathias Filho, por toda a paciência e apoio desde o meu projeto final de graduação. Guardarei para sempre tudo o que fez por mim.

Aos meus colegas de trabalho, por toda a ajuda e apoio nos momentos mais difíceis.

A Déborah e Emanuelle, por sempre conseguirem dar um jeito nos meus problemas.

## Resumo

Marques, Fabio Seixas; Lucena, Carlos José Pereira de; Mathias Filho, Ivan. **Um Processo Baseado em MDA para a Especialização de Mecanismos de Persistência**. Rio de Janeiro, 2006. 88p. Dissertação de Mestrado - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Com as constantes mudanças nos requisitos funcionais e não-funcionais do domínio das aplicações, há uma grande necessidade dos softwares se adaptarem a essas mudanças. Devido a isso, o processo proposto neste trabalho tem o intuito de automatizar algumas partes do desenvolvimento de aplicações, minimizando o trabalho manual necessário para adaptar os sistemas de software às constantes necessidades de mudanças determinadas pela realidade do mundo dos negócios. A abordagem Model Driven Architecture (MDA), definida pela OMG, foi utilizada como base para este processo por ter como objetivo principal a transformação de modelos; ou seja, gerar, a partir de um modelo independente de plataforma, um ou mais modelos específicos para uma plataforma. O processo proposto é dividido de acordo com as etapas da MDA, tendo como objetivo auxiliar nas etapas de geração de modelos e na adaptação de frameworks de persistência. Foi utilizado no processo em questão, o conceito de base de conhecimento, tendo como objetivo principal mapear as informações contidas no modelo com as regras de transformações a serem realizadas no mesmo. Essas regras são responsáveis pela definição das modificações a serem realizadas nos modelos independentes de plataforma com o objetivo de incluir um determinado framework de persistência em uma aplicação. A linguagem RDL foi utilizada para formalizar as regras de transformações utilizadas no processo, facilitando o desenvolvimento e a manutenção das mesmas. O processo tem como resultado um modelo específico para uma plataforma, contendo ainda um framework de persistência adaptado à arquitetura utilizada. Alguns arquivos de configuração também são gerados com o intuito de minimizar o trabalho manual.

## Palavras-chave

Engenharia de Software; MDA; Transformações; Persistência; RDL

## Abstract

Marques, Fabio Seixas; Lucena, Carlos José Pereira de; Mathias Filho, Ivan. **A Process Based on MDA for Specialization of Persistence Mechanisms**. Rio de Janeiro, 2006. 88p. Dissertation - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

According to the changes on functional and non-functional requirements on application field, there is a great need for softwares to adapt themselves to these changes. Due to it, the process involved on this work has the purpose of automate some parts of applications' developments, reducing the manual work needed to adapt the software systems to the need of constant changes determined by business world's reality. The approach Model Driven Architecture (MDA), defined by OMG, was used as base for this process, having as main goal the transformation of models; i mean, generate from an platform independent model, one or more platform specific models. The process suggested is divided according to the MDA steps, having as a goal the support of generate models steps and on the adaptation of Persistence Frameworks. It was used on the suggested process, the concept of base of knowledge, having as main goal map the informations held within the model with the transformation rules to be done on these models. These rules are responsible by the definition of changes to be done on platform-independent models aiming to include a specific Persistence Framework in an application. The language RDL was used to formalize the transformation rules used in the process, making the development and maintenance of themselves easy the process has as a result a platform-specific model, also a Persistence Framework adapted to the architecture used. Some configuration files also are generated, intending to reduce the manual work.

## Keywords

Software Engineering; MDA; Transformations; Persistence; RDL

# Sumário

1	Introdução	13
1.1.	Desenvolvimento de Software	13
1.2.	Generative Programming	13
1.3.	Desenvolvimento de Software Utilizando MDA (Model Driven Architecture)	14
1.4.	O Objetivo da Dissertação e o Resumo da Solução Proposta	15
1.5.	Lista das Contribuições da Dissertação	16
1.6.	Estrutura da Dissertação	17
2	Trabalhos Relacionados	18
2.1.	Odyssey-MDA	18
2.2.	UML Model Transformation Tool (UMT)	20
2.3.	AppFuse	23
2.4.	Model in Action (Mia)	24
2.5.	Metastorage	27
3	Tecnologias Relacionadas	31
3.1.	Model Driven Architecture (MDA)	31
3.1.1.	Introdução	31
3.1.2.	CIM (Computation Independent Model)	32
3.1.3.	PIM (Plataform Independent Model)	32
3.1.4.	PSM (Plataform Specific Model)	33
3.1.5.	Marcações	33
3.1.6.	Transformação do PIM em PSM	33
3.1.7.	Geração de Código	34
3.2.	XML Metadata Interchange (XMI)	34
3.3.	Reuse Description Language (RDL)	35
3.4.	Frameworks de Persistência	37
3.4.1.	Castor	38
3.4.2.	Hibernate	39
3.4.3.	iBatis	40



3.4.4. OJB	41
4 Processo de Transformação	43
4.1. Criação do Modelo PIM	46
4.2. Configuração do Modelo PIM (Marcações)	46
4.3. Configuração da Base de Conhecimento	47
4.4. Execução das Regras de Transformações	49
4.5. Criação do Modelo PSM	54
4.6. Considerações Finais	56
5 Estudos de Caso	58
5.1. Sistema de Sincronização dos Dados Acadêmicos do SAU e do AulaNet	58
5.2. Site de Leilão Virtual	64
6 Ferramenta para a Especialização de Mecanismos de Persistência	71
6.1. Introdução	71
6.2. Manutenção da Base de Conhecimento	72
6.3. Execução do Processo	76
7 Considerações Finais	82
7.1. Conclusão	82
7.2. Principais Contribuições	83
7.3. Trabalhos Futuros	84
8 Bibliografia	86

## Lista de figuras

Figura 1 - Exemplos de Generative Programming [Iseger05]	14
Figura 2 - Visão geral do Model Driven Architecture [MDA]	15
Figura 3 - Cenário de utilização da Odyssey-MDA [Odyssey-MDA05]	18
Figura 4 - Exemplo de um arquivo XML contendo as regras de transformações [Odyssey-MDA05]	19
Figura 5 - Mecanismos genéricos de transformação (built-ins) [Odyssey-MDA05]	20
Figura 6 - Arquitetura da Ferramenta UMT (Independente de Plataforma) [UMT]	21
Figura 7 - Arquitetura da Ferramenta UMT (Específica para uma Plataforma) [UMT]	22
Figura 8 - Exemplo de um arquivo XSLT contendo as regras de transformação [UMT]	22
Figura 9 - Exemplo de configuração do Appfuse, para a geração de uma aplicação [AppFuse]	23
Figura 10 - Visão Geral da Ferramenta Model in Action [MIA]	25
Figura 11 - Exemplo de criação de um serviço, utilizando a parte gráfica da ferramenta Model in Action [MIA]	26
Figura 12 - Exemplo da criação de regras de transformação, sem utilizar a parte gráfica da ferramenta MIA [MIA]	26
Figura 13 - Arquitetura da Ferramenta Model in Action [MIA]	27
Figura 14 - Exemplo de criação de uma classe e seus atributos e métodos utilizando XML [Metastorage]	29
Figura 15 - Exemplo da criação de um report [Metastorage]	30
Figura 16 - Model Driven Architecture	32
Figura 17 - Transformação do Modelo PIM no Modelo PSM [MDA]	34
Figura 18 - Estrutura de um Script RDL [Oliveira05]	36
Figura 19 - Exemplo de um Modelo Simples	36
Figura 20 - Exemplo de um Script RDL	37
Figura 21 - Modelo Atualizado	37
Figura 22 - Arquitetura do Framework Castor JDO [Castor05]	38
Figura 23 - Exemplo do Funcionamento do Framework Hibernate	

[Hibernate]	40
Figura 24 - Visão Geral do Framework iBatis [iBATIS]	41
Figura 25 - Camadas do Framework OJB [OJB05]	42
Figura 26 - Visão Geral do Processo Proposto	44
Figura 27 - Modelo PIM	46
Figura 28 - Mapeamento Marcação (Modelo) x Camada (Base de Conhecimento)	47
Figura 29 - Base de Conhecimento	48
Figura 30 - Exemplificando a Base de Conhecimento	49
Figura 31 - Modelo PIM e seu XMI correspondente	50
Figura 32 - Tags da Camada de Persistência	51
Figura 33 - Arquivos RDL da Tag Hibernate	52
Figura 34 - Arquivo RDL utilizando o caracter “?”	53
Figura 35 - Exemplo de um Arquivo RDL	53
Figura 36 - Resultado da Transformação	54
Figura 37 - Modelo PSM gerado para o Framework de Persistência Castor	55
Figura 38 - Arquivo XML da configuração do banco para o Framework Hibernate	56
Figura 39 - Arquivo XML contendo o mapeamento tabela-classe do Framework Hibernate	56
Figura 40 - Modelo PIM do Sistema de Sincronização dos Dados Acadêmicos do SAU e do AulaNet	59
Figura 41 - Modelo PIM Configurado para a Camada de Persistência	60
Figura 42 - Arquivo RDL do Padrão DAO	61
Figura 43 - Arquivo RDL do Padrão Facade	62
Figura 44 - Modelo PSM Gerado	63
Figura 45 - Comparação do Modelo Original com o Modelo Gerado pelo Processo	64
Figura 46 - Modelo PIM do Site de Leilão Virtual	66
Figura 47 - Modelo PIM Configurado para a Camada de Persistência	67
Figura 48 - Arquivo RDL que contém as regras de transformações do framework de persistência Hibernate	68
Figura 49 - Modelo PSM Gerado através do Processo Proposto e Importado na Ferramenta CASE Poseidon	69
Figura 50 - Modelo da Camada de Persistência Finalizado pelo PSM Designer	70

Figura 51 - Visão Geral da Ferramenta	72
Figura 52 - Funcionamento do Módulo 1	73
Figura 53 - Modelo da Base de Conhecimento	73
Figura 54 - Janela de Manutenção de Camadas	74
Figura 55 - Janela de Manutenção de Tags	75
Figura 56 - Janela de Manutenção de Arquivos RDL	76
Figura 57 - Funcionamento do Módulo 2	77
Figura 58 - Janela de Importação do Arquivo XML Contendo o Modelo PIM	78
Figura 59 - Janela para Efetuar as Transformações (Escolha da Camada)	79
Figura 60 - Janela para Efetuar Transformações (Escolha da Tecnologia)	79
Figura 61 - Geração dos Arquivos XML de Configuração	80
Figura 62 - Janela de Exportação do Arquivo XMI Contendo o Modelo PSM Gerado	81