

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO



Luiz Antonio Ricci

**SHDM .Net - um Framework e Ambiente de
Desenvolvimento dirigido por Ontologias para Aplicações
Hipermedia**

PUC-Rio - Certificação Digital Nº 0410843/CA

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: Daniel Schwabe

Rio de Janeiro, setembro de 2006.



Luiz Antonio Ricci

**SHDM .Net - um Framework e Ambiente de
Desenvolvimento dirigido por Ontologias para Aplicações
Hipermedia**

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.

Daniel Schwabe
Orientador
PUC-Rio

Marco Antônio Casanova
PUC-Rio

Simone Diniz Junqueira Barbosa
PUC-Rio

Prof. José Eugenio Leal
Coordenador Setorial do Centro Técnico Científico - PUC-Rio

Rio de Janeiro, 06 de setembro de 2006.

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

Luiz Antonio Ricci

Graduou-se em Tecnólogo em Processamento de Dados na PUC-Rio em 2000, Pós-graduado em Análise, Projeto e Gerência de Sistemas pela Puc-Rio (CCE) em 2001. É Analista de Sistemas da Pontifícia Universidade Católica do Rio de Janeiro onde é responsável pelo desenvolvimento de aplicações web do setor acadêmico da universidade. Seus interesses incluem engenharia de software, desenvolvimento baseado em modelos e orientação a objetos.

Ficha Catalográfica

Ricci, Luiz Antonio

SHDM.Net : um framework e ambiente de desenvolvimento dirigido por ontologias para aplicações hipermídia / Luiz Antonio Ricci ; orientador: Daniel Schwabe. – 2006.

117 f. : il. ; 30 cm

Dissertação (Mestrado em Informática)– Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2007.

Inclui bibliografia

1. Informática – Teses. 2. Aplicações hipermídia. 3. Desenvolvimento dirigido por modelos. 4. Linguagem de domínio específico. 5. OOHDM. 6. SHDM. 7. DSL tools. 8. Framework. 9. Ontologias. I. Schwabe, Daniel. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD:004

Dedico este trabalho ao meu orientador Daniel Schwabe, por me prover ensinamentos e oportunidades importantes na minha vida acadêmica e profissional.

A minha namorada Juliana pelo amor e carinho, compreensão e contribuição nos momentos mais difíceis.

Aos meus pais, Antonio e Nilza, que me ensinaram muitos valores e me prepararam para a vida.

Agradecimentos

Gostaria de agradecer a Deus, à PUC-Rio e ao departamento de informática pela oportunidade que tive para a realização do mestrado.

Agradeço ao meu orientador, Prof. Daniel Schwabe, pela confiança, atenção e apoio na realização deste trabalho. Ao Professor Marco Antônio Casanova pelas orientações e ajuda nos procedimentos administrativos.

A Profa. Simone Diniz Junqueira Barbosa e ao Professor Marco Antônio Casanova, fico agradecido por tão prontamente atenderem ao meu pedido de participação da avaliação desta dissertação.

A Microsoft Research pelo suporte no desenvolvimento deste projeto.

Aos Desenvolvedores da Microsoft, Alan Cameron Wills, Gareth Jones, Grayson Myers, Stuart Kent pelo suporte na customização da ferramenta.

Aos meus amigos do laboratório TecWeb: Guilherme Szundy, Adriana Medeiros e Patrícia Seefelder pelo apoio e suporte na retirada de dúvidas.

A Deborah, Emanuele e Ruth pela ajuda nos procedimentos acadêmicos.

A minha gerente, Lea, por liberar horas tão preciosas para estudos. Aos meus amigos de trabalho da PUC-Rio, que me apoiaram nesta caminhada.

A minha namorada, Juliana, pelo incentivo, paciência, apoio e compreensão nos momentos de ausência.

Aos meus pais, Antonio e Nilza pela compreensão, educação, ensinamentos e por me prepararem para a vida.

Resumo

Ricci, Luiz Antonio; Schwabe, Daniel. **SHDM .Net - um Framework e Ambiente de Desenvolvimento dirigido por Ontologias para Aplicações Hiperímia**. Rio de Janeiro, 2006. 117p. Dissertação de Mestrado - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Esta dissertação apresenta a forma como foi desenvolvida uma arquitetura de desenvolvimento dirigida por modelos combinada com um ambiente de desenvolvimento para dar apoio ao desenvolvimento de aplicações Web, utilizando SHDM/OOHDM como métodos de desenvolvimento. Os modelos, descritos através de uma interface gráfica, são processados pela ferramenta. A interface da aplicação também é descrita utilizando modelos, e a interface produzida é definida por uma ontologia de interface concreta, especificada em RDFS, que faz o mapeamento entre os elementos de interface concretos e alguma tecnologia de destino como ASP.Net e XAML. O principal objetivo é tornar mais fácil o desenvolvimento de uma aplicação Web utilizando Visual Studio 2005 como IDE e SHDM como método de desenvolvimento, permitindo que o arquiteto de software concentre seus esforços na modelagem da aplicação. O SHDM .Net produz uma aplicação completa, criando linguagens específicas de domínio para cada modelo desenvolvido. A principal funcionalidade do SHDM .Net é a integração entre as fases de análise e de desenvolvimento de um projeto.

Palavras-chave

Aplicações Hiperímia; Desenvolvimento Dirigido por Modelos; Linguagem de Domínio Específico; OOHDM; SHDM; Visual Studio .Net; C#; DSL Tools; Framework; Ontologias;

Abstract

Ricci, Luiz Antonio; Schwabe, Daniel. **SHDM .Net - a Framework and Development Environment driven by Ontologies for Hypermedia Applications.** Rio de Janeiro, 2006. 117p. MSc. Dissertation - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

This dissertation presents how a model driven development architecture and environment framework to support Web application development are being developed using SHDM/OOHDM as development methods. The models are described through graphical diagrams that are processed by the environment. The application interface is also produced using models, and the interface produced is defined by concrete interface ontologies, specified in RDFS, that maps the concrete *widgets* to some target technology such as ASP.Net and XAML. The main purpose is to ease the development of a web application using Visual Studio .Net 2005 as an IDE and SHDM as the development method, allowing the application architect to focus on the application modeling. SHDM .Net generates a complete runnable application; also producing domain-specific languages for each application modeled, resulting in a programming model much more concise and natural. The main feature of SHDM .Net is the integration between the analysis and development phases of a project.

Keywords

Hypermedia Applications; Model Driven Development; Domain Specific Language; OOHDM; SHDM; Visual Studio .Net; C#; DSL Tools; Framework; Ontologies;

Sumário

1	Introdução	15
1.1.	Motivação	15
1.2.	Objetivos	16
1.3.	Organização da Dissertação	17
2	Fundamentos	19
2.1.	OOHDM e SHDM	19
2.1.1.	Fases	19
3	Arquitetura de Implementação	28
3.1.	MDD	29
3.2.	RDF/RDFS	30
3.3.	Arquitetura de Desenvolvimento	30
3.3.1.	Ferramentas de apoio ao desenvolvimento	30
3.4.	Modelos SHDM .Net	31
4	Arquitetura do Framework	34
4.1.	Diagramas do SHDM .Net	34
4.2.	Diagrama de Classes Navegacionais	34
4.2.1.	Classe NavigationalMetaModel	37
4.2.2.	Classe NavigationalClass	37
4.2.3.	Classe Operation	46
4.3.	Diagrama de Contextos Navegacionais	47
4.3.1.	Classe NavigationalContextMetaModel	50
4.3.2.	Classe AccessStructure	51
4.3.3.	Classe Context	54
4.3.4.	Classe Scope	57
4.4.	Diagrama de Interface Abstrata	59

4.4.1. Classe <code>AbstractInterfaceModel</code>	61
4.4.2. Classe <code>AbstractInterfaceElement</code>	62
4.4.3. Classe <code>CompositeInterfaceElement</code>	63
4.4.4. Classe <code>ElementExhibitor</code>	63
4.4.5. Classe <code>SimpleActivator</code>	63
4.4.6. Classe <code>ValueCapturer</code>	64
4.4.7. Classe <code>ArbitraryValue</code>	64
4.4.8. Classe <code>PredefinedOptions</code>	64
4.4.9. Classe <code>ContinousGroup</code>	65
4.4.10. Classe <code>DiscreteGroup</code>	65
4.4.11. Classe <code>MultipleChoices</code>	65
4.4.12. Classe <code>SingleChoice</code>	65
4.5. Diagrama de Interface Concreta	66
4.5.1. <code>ConcreteInterfaceModel</code>	67
4.5.2. <code>AbstractInterfaceElement</code>	68
4.6. Ontologia de <i>widgets</i> concretos	69
4.6.1. <code>ConcreteInterface</code>	69
4.6.2. <code>ConcreteWidget</code>	69
4.7. Geração da DSL-SHDM	70
4.8. Arquitetura de Runtime	72
4.9. Restrições de Implementação	74
4.9.1. Restrições no design dos diagramas	74
4.9.2. Restrições tecnológicas	74
5 Exemplo de desenvolvimento de uma aplicação	75
5.1. Ambiente de Desenvolvimento	75
5.2. Criação do Diagrama de classes Navegacionais	80
5.3. Diagrama de Contextos Navegacionais	86
5.4. Diagrama de Interface Abstrata	92
5.5. Diagrama de Interface Concreta	93
5.6. Exemplo de uma Instância da Ontologia de Interface Concreta	96
5.7. Execução da Aplicação	101
6 Conclusões	103

6.1. Trabalhos Relacionados	103
6.2. Contribuições	105
6.3. Trabalhos Futuros	107
7 Bibliografia	109
Apêndice 1 – Exemplo da interface do site Css Zen Garden	112

Lista de tabelas

Tabela 1 – Fases do desenvolvimento de uma aplicação e artefatos produzidos. 29

Lista de figuras

Figura 1 - Exemplo de UID representando uma aplicação de login	20
Figura 2 – Exemplo de modelo conceitual de uma Loja de CDs apresentado por [Jacyntho, 2002].	20
Figura 3 – Exemplo de Diagrama de Classes Navegacionais apresentado por [Jacyntho, 2002].	21
Figura 4 – Exemplo de um Diagrama de Contextos Navegacionais apresentado por [Jacyntho, 2002].	22
Figura 5 - Exemplo de Interface Abstrata apresentado por [Moura, 2004].	25
Figura 6 - Exemplo do Diagrama de Interface Concreta	26
Figura 7 - Processo de construção do editor de um diagrama utilizando Microsoft DSL-Tools.	32
Figura 8 - Processo de desenvolvimento de uma aplicação usando o <i>framework</i>	33
Figura 9 – Metamodelo do Diagrama de Classes Navegacionais do SHDM	35
Figura 10 – Modelo do diagrama de classes navegacionais	36
Figura 11 – Propriedades de um atributo ContextAnchor	42
Figura 12 – Parâmetros passados para o contexto StudentByResearchArea	43
Figura 13 – Propriedades de um atributo IndexAnchor	44
Figura 14 - Parâmetros passados para o índice StudentsByProfessor	44
Figura 15 - Propriedades de um atributo Index	45
Figura 16 - Parâmetros passados para o índice StudentsByProfessor	46
Figura 17 – Exemplo de diagrama de classes navegacionais	47
Figura 18 – Metamodelo de Contextos Navegacionais	48
Figura 19 – Modelo do diagrama de contextos navegacionais	49
Figura 20 – Exemplo de ContextElement baseado em todas as instâncias de uma classe	54
Figura 21 - Exemplo de ContextElement com filtro nas instâncias de uma classe	55
Figura 22 – Campo “Query” preenchido com uma expressão de filtro na DSL-SHDM	55

Figura 23 – Exemplo do diagrama de contextos navegacionais	58
Figura 24 – Metamodelo do Diagrama de Interface Abstrata	59
Figura 25 – Modelo do diagrama de interface abstrata no DSL-Tools	61
Figura 26 – Diagrama de Interface Abstrata	66
Figura 27 – Modelo do Diagrama de Interface Concreta.	68
Figura 28 – Ontologia de <i>widgets</i> concretos	70
Figura 29 – Fluxo de requisição de uma aplicação produzida pelo SHDM .Net	73
Figura 30 – Tela inicial Visual Studio	75
Figura 31 – Criando um novo projeto	76
Figura 32 – Adicionando um novo item no projeto	77
Figura 33 – Adicionando um diagrama de classes navegacionais	77
Figura 34 – Tela de Edição do diagrama de classes navegacionais	78
Figura 35 – Adicionando um diagrama de contextos navegacionais	79
Figura 36 – Vinculação do diagrama de contextos navegacionais no diagrama de classes navegacionais.	80
Figura 37 - Vinculação do diagrama de classes navegacionais no diagrama de contextos navegacionais.	80
Figura 38 – Adicionando um atributo a uma classe navegacional	81
Figura 39 – Alterando as propriedades do atributo	81
Figura 40 – Diagrama de classes navegacionais finalizado	84
Figura 41 – Diagrama de contextos navegacionais	87
Figura 42 – Diagrama de interface abstrata do contexto ProfessorAlpha	92
Figura 43 – Exemplo do mapeamento do CompositeInterfaceElement ProfessorsAlpha com a estrutura de acesso ProfessorsAlpha.	93
Figura 44 – Propriedades do diagrama de interface concreta.	94
Figura 45 - Propriedades do CompositeInterfaceElement ProfessorsAlpha	94
Figura 46 - Interface produzida pelo diagrama de interface concreta sem o CSS	102
Figura 48 – Interface padrão do site Css Zen Garden	113
Figura 51 – Exemplo do diagrama de interface concreta de parte do site Css Zen Garden	116

Lista de quadros

Quadro 1 – Exemplo de preenchimento do atributo code na classe Computed	46
Quadro 2 – Consulta Sparql que retorna todas as instâncias da classe Student	56
Quadro 3 – Exemplo do <i>template</i> de geração da DSL-SHDM de uma classe navegacional	72
Quadro 4 – Exemplo da DSL-SHDM produzida pelo diagrama da Figura 39.	84
Quadro 5 – Exemplo de parte da DSL-SHDM da classe Student	86
Quadro 6 – Exemplo do código da DSL-SHDM produzido da Estrutura de Acesso StudentsAlpha.	89
Quadro 7 – Exemplo do código da DSL-SHDM produzido do Contexto StudentAlpha.	91
Quadro 8 – Código da interface gerada pelo diagrama de interface abstrata ProfessorAlpha	95
Quadro 9 – Exemplo de uma instância de ontologia de interface concreta	101
Quadro 10 – Trecho do código da página do CSS Zen Garden	115
Quadro 11 – Trecho do código do site Css Zen Garden produzido pelo SHDM .Net	117