

**Elder José Reoli Cirilo**

**GenArch: Uma Ferramenta Baseada em Modelos para  
Derivação de Produtos de Software**

**Dissertação de Mestrado**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio.

Orientadores: Carlos José Pereira de Lucena  
Uirá Kulesza

Rio de Janeiro  
Abril de 2008

**Elder José Reoli Cirilo**

## **GenArch: Uma Ferramenta Baseada em Modelos para Derivação de Produtos de Software**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

**Prof. Carlos José Pereira de Lucena**

Orientador

Departamento de Informática – PUC-Rio

**Prof. Uirá Kulesza**

Co-Orientador

Departamento de Informática e Matemática Aplicada - UFRN

**Prof. Arndt von Staa**

Departamento de Informática – PUC-Rio

**Prof. Ivan Mathias Filho**

Departamento de Informática – PUC-Rio

**Prof. José Eugenio Leal**

Coordenador Setorial do Centro  
Técnico Científico

Rio de Janeiro, 11 de abril de 2008

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

**Elder José Reoli Cirilo**

Graduou-se no Curso de Bacharelado em Ciência da Computação da Universidade Federal de Juiz de Fora (UFJF) em 2006.

Ficha Catalográfica

Cirilo, Elder José Reoli

GenArch: Uma ferramenta baseada em modelos para derivação de produtos de software / Elder José Reoli Cirilo ; orientador: Carlos José Pereira de Lucena ; co-orientador: Uirá Kulesza. – 2008.

100 f. : il. ; 30 cm

Dissertação (Mestrado em Informática)–Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2008.

Inclui bibliografia

1. Informática – Teses. 2. Linhas de produtos de software. 3. Desenvolvimento Dirigido por Modelos. 4. Engenharia de Domínio. 5. Ferramentas para Derivação de Linha de Produtos de Software. I. Lucena, Carlos José Pereira de. II. Kulesza, Uirá. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

A Deus e aos meus pais José Cyrilo e Maria Josefina

## Agradecimentos

Me sinto privilegiado por ter conhecido Uirá e ganhado a oportunidade de tê-lo como um dos meus orientadores. Esse trabalho não teria sido possível sem suas idéias, seus ensinamentos e principalmente, sua amizade. Serei sempre grato a ele por tudo isso.

Me sinto especialmente honrado por ter professor Lucena como um dos meus orientadores. Gostaria de agradecer-lo pelo acolhimento, liberdade, confiança e ensinamentos. Sei que tudo isso tornou a realização desse trabalho mais simples.

Agradeço a todos os professores do Departamento de Informática que contribuíram indiretamente para realização desse trabalho. Em especial gostaria de agradecer aos Professores Arndt von Staa e Renato Cerqueira que contribuíram para enriquecer meus conhecimentos. Gostaria de agradecer também a todos os colegas da PUC-Rio que tive o prazer de conquistar durante meu mestrado: Andrew, Carol, Camila, Cidiane, Cláudio (Baiano), Dani, Ingrid, Maíra, Márcio, Pádua e Rodrigo. Em especial, gostaria de agradecer a Paulo e Sergio que me acompanharam desde os meus primeiros dias de PUC-Rio e a Roberta minha primeira companheira de pesquisa que muito me ensinou e contribuiu para minha formação.

Um agradecimento especial a todos os amigos de Juiz de Fora, em especial, Carolina, Daniel Barral, Daniel Nicácio, Edmar, Fábio, Júlio, Poliana, Rafael, Vinícios, Tiago Melquiades, Thiago Senador.

Agradeço a PUC-Rio e a CAPES pelo apoio financeiro necessário a realização desse trabalho.

Gostaria de agradecer também as pessoas mais importantes da minha vida, minha família. Aos meus pais, José Cyrilo e Maria Josefina, pelo amor, carinho e confiança que sempre depositaram em mim. A minha tia Maria da Conceição por ser uma segunda mãe. As minha irmãs, Elaina e Elisana, e as minhas sobrinhas Maria Rita e Manuela pelos sorrisos e pensamento positivo. A Flávia, minha namorada, que sempre me manteve animado nos momentos difíceis da realização desse trabalho.

Finalmente agradeço a Deus, o realizador de todas essas maravilhas em minha vida.

## Resumo

Cirilo, Elder José Reioi; Lucena, Carlos José Pereira de; Kulesza, Uirá **GenArch: Uma Ferramenta Baseada em Modelos para Derivação de Produtos de Software**. Rio de Janeiro, 2008. 100p. Dissertação de Mestrado - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Este trabalho apresenta uma ferramenta baseada em modelos para derivação de produtos de LPSs, denominada GenArch. O objetivo principal da ferramenta é permitir que a comunidade de desenvolvimento de software tradicional, utilize conceitos e fundamentos de abordagens de LPSs na produção de seus sistemas ou partes de seus sistemas sem a necessidade do entendimento de modelos e conceitos complexos. A abordagem implementada pela ferramenta foi elaborada com base em fundamentos do desenvolvimento dirigido por modelos. Centrada na definição de três modelos (características, implementação e configuração), a ferramenta permite a derivação automática de produtos ou frameworks existentes. O trabalho também define um conjunto específico de anotações Java que possibilitam a geração automática dos modelos de derivação a partir dos elementos de implementação da arquitetura de uma LPS. A plataforma Eclipse e as tecnologia EMF e openArchitectureWare foram utilizadas como base para a implementação da ferramenta. Uma extensão da ferramenta que atende especificamente aos modelos de componente Spring e OSGi, também é proposta nessa dissertação. Tal extensão permite a instanciação automática da LPS e aplicações através de diferentes tipos de customizações, variando da configuração fina de propriedades de componentes até a seleção automática de quais componentes irão compor o produto final. Como parte de validação da abordagem, a ferramenta foi utilizada na derivação automática de três diferentes estudos de caso: (i) o framework JUnit; (ii) uma LPS de jogos J2ME; e (iii) uma aplicação web baseada em serviços. Diversas lições aprendidas e resultados do uso da ferramenta nestes três diferentes cenários são também apresentadas.

## Palavras-chave

Linhas de Produtos de Software; Desenvolvimento Generativo; Desenvolvimento Dirigido por Modelos, Engenharia de Domínio, Ferramentas para Derivação de Linha de Produtos de Software

## Abstract

Cirilo, Elder José Reoli; Lucena, Carlos José Pereira de; Kulesza, Uirá  
**GenArch: A Model-Based Product Derivation Tool.** Rio de Janeiro, 2008. 100p. Msc Dissertation – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

This work presents a model-based tool for product derivation, called GenArch, which aims to enable the mainstream software developer community to use the concepts and foundations of the SPL approach, without the need to understand complex concepts or models. The tool approach is build on top of model-driven development techniques. It is centered on the definition of three models (feature, implementation and configuration models), which enable the automatic instantiation of software product lines (SPLs) or frameworks. A set of specific Java annotations are also defined to allow generating automatically many of the models, based on existing implementations elements of SPL architectures. The Eclipse platform, and EMF and openArchitectureWare technologies are used as the base for the implementation of the tool. The dissertation also presents a GenArch extension that addresses the new abstractions provided by the Spring and OSGi component models. Different kinds of customizations are provided by this extension varying from fine-grained configuration of component properties to the automatic selection of components that will compose the final product. As part of the approach validation, the tool was used in the derivation of three case studies: (i) JUnit framework; (ii) a J2ME games SPL; (iii) a service oriented Web application. Several lessons learned and discussions resulting from the use of the tool also are described.

## Keywords

Software Product Lines; Generative Programming; Model-Driven Development, Domain Engineering, Software Product Lines Derivation Tool

# Sumário

1	Introdução	14
1.1.	Problemas	15
1.2.	Limitação das Abordagens Atuais	16
1.3.	Solução Proposta	16
1.4.	Objetivos	17
1.5.	Organização do Texto	17
2	Desenvolvimento de Linhas de Produtos de Softwares	19
2.1.	Linhas de Produtos de Software	19
2.2.	Desenvolvimento Generativo	23
2.3.	Implementando Linhas de Produtos de Software	25
2.3.1.	Frameworks Orientados a Objetos	25
2.3.2.	Orientação a Aspectos	26
2.3.3.	Tecnologia de Componentes	27
2.4.	Ferramentas para Derivação de Produtos de Software Baseadas em Modelo de Características	29
2.4.1.	Gears	29
2.4.2.	pure::variants	31
2.5.	Conclusão	33
3	GenArch: Uma Ferramenta Baseada em Modelos para Derivação de Produtos de Software	35
3.1.	Visão Geral da Abordagem	35
3.1.1.	Anotação do código-fonte com características e variabilidades	36
3.1.2.	Modelos de Derivação	37
3.1.3.	Criação Automática da Estrutura de Templates	38
3.1.4.	Refinamento e Sincronização dos Modelos de Derivação	39
3.1.5.	Documentação de Variabilidades	40
3.1.6.	Processo de Derivação de Produtos	41
3.2.	Arquitetura e Implementação da Ferramenta GenArch	41
3.2.1.	Visão Geral da Arquitetura	42

3.2.2. Eclipse Modeling Framework (EMF)	44
3.2.3. openArchitectureWare (oAW)	47
3.2.3.1. Estendendo as funcionalidades da linguagem XPand	48
3.2.4. Feature Modeling Plug-in	49
3.3. Conclusão	50
4 Estudos de Caso	52
4.1. Framework JUnit	52
4.1.1. Anotando código do JUnit com Características e Variabilidades	54
4.1.2. Gerando versões iniciais dos modelos	55
4.1.3. Implementando Variabilidades do JUnit com Templates	58
4.1.4. Instanciando o JUnit	60
4.2. Linha de Produtos para Jogos J2ME	62
4.2.1. Descrição do Estudo de Caso	62
4.2.2. Anotando Características na LPS para Jogos J2ME	64
4.2.3. Gerando os Modelos do jogo <i>Rain of Fire</i>	64
Derivando Jogos para Diferentes Modelos de Celular	67
4.2.4.	67
4.3. Sumário	68
5 Derivando Aplicações Baseadas em Spring e OSGi	69
5.1. Spring e OSGi – Tecnologias de Componentes	69
5.2. Integração das Tecnologias Spring e OSGi	71
5.3. Arquitetura da Extensão	73
5.3.1. Estendendo os modelos GenArch para suportar Spring e OSGi	74
5.3.2. Construção automática dos modelos a partir de artefatos Spring e OSGi	75
5.3.3. Derivando produtos Spring/OSGi	80
5.4. Conclusões e Resultados	83
6 Trabalhos Relacionados	85
6.1. Abordagem para Derivação de Linhas de Produtos de Software	85
6.2. Abordagem para Instanciação de Frameworks OO	86
6.3. Abordagem para Configuração Automática de Aplicações Corporativas	88
7 Conclusão	90
7.1. Contribuições	91

7.2. Discussões e Lições Aprendidas	92
7.3. Trabalhos Futuros	94
7.4. Uso da Ferramenta GenArch na Indústria	95
Referências	97

## Lista de Figuras

Figura 1. Modelo de Domínio Generativo	23
Figura 2. Elementos da ferramenta Gears	30
Figura 3. Processo de derivação do pure::variants	33
Figura 4. Visão geral da abordagem	36
Figura 5. Exemplo de template criado automaticamente	39
Figura 6. Arquitetura da ferramenta GenArch	43
Figura 7. Meta-modelo do modelo de implementação	45
Figura 8. Meta-modelo do modelo de configuração	45
Figura 9. Meta-modelo do modelo de derivação	46
Figura 10. Fluxo de execução do plug-in oAW (Stahl et al. 2006)	48
Figura 11. Modelo de característica do Framework JUnit	50
Figura 12. Arquitetura Orientada a Aspectos do JUnit	53
Figura 13. Classe <code>TestCase</code> anotada	54
Figura 14. Versão Inicial dos Modelos GenArch do JUnit	55
Figura 15. Aspect <code>RepeatedTestGeneric</code> anotado	56
Figura 16. Modelos GenArch do JUnit – Versão Final	57
Figura 17. Mapeamento entre um elemento de implementação e uma característica	58
Figura 18. <code>TestCaseTemplate</code>	59
Figura 19. Configuração do modelo de característica	60
Figura 20. <code>RepeatedTestGenericTemplate</code>	61
Figura 21. Janela de derivação	62
Figura 22. Aspecto <code>LoadImgOnDemand</code> e <code>LoadImgOnInit</code> com anotações GenArch	64
Figura 23. Modelos GenArch do jogo <i>Rain of Fire</i>	66
Figura 24. Configurações do modelo de característica do jogo <i>Rain of Fire</i>	67
Figura 25. Jogo <i>Rain of Fire</i> executando em modelos diferentes de celulares	68
Figura 26. Arquitetura Spring/OSGi da aplicação JPetStore	72
Figura 27. Classe Java anotada	76
Figura 28. Versão inicial dos modelos de implementação e <i>deployment</i> da aplicação JPetStore	78
Figura 29. Versão final dos modelos de derivação da aplicação JPetStore	79

Figura 30. Versão inicial dos modelos de configuração e características	80
Figura 31. Versão final dos modelos de configuração e características	80
Figura 32. Template de um arquivo de configuração Spring	82

## Lista de Tabelas

Tabela 1. Anotações GenArch e seus atributos	37
Tabela 2. Pacotes gerados automaticamente pelo EMF	46
Tabela 3. Funções de extensão	48
Tabela 4. Anotações GenArch para Spring/OSGi e seus atributos	76