



Bernardo Evangelho Miranda

**Conceitos Centrais e Componentização de
Diagramas de Classe UML representados
em Grafo**

Dissertação de Mestrado

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

Orientador: Prof. Arndt von Staa

Rio de Janeiro
Abril de 2013



Bernardo Evangelho Miranda

**Conceitos Centrais e Componentização de
Diagramas de Classe UML representados
em Grafo**

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

Prof. Arndt von Staa

Orientador

Departamento de Informática – PUC-Rio

Prof. Gustavo Robichez de Carvalho

Departamento de Informática – PUC-Rio

Profa. Simone Diniz Junqueira Barbosa

Departamento de Informática – PUC-Rio

Prof. José Eugenio Leal

Coordenador Setorial do Centro

Técnico Científico – PUC-Rio

Rio de Janeiro, 05 de abril de 2013

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

Bernardo Evangelho Miranda

Graduou-se em Engenharia de Computação na PUC-Rio em 2008. Desenvolvedor de sistema Web baseados em Java e C# e gerente de projetos.

Ficha Catalográfica

Miranda, Bernardo Evangelho

Conceitos centrais e componentização de diagramas de classe UML representados em grafo / Bernardo Evangelho Miranda ; orientador: Arndt Von Staa. – 2013.

101 f. : il. (color.) ; 30 cm

Dissertação (mestrado)—Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2013.

Inclui bibliografia

1. Informática – Teses. 2. Componentes. 3. Diagramas de classes. 4. Conceitos centrais. 5. Clusterização. 6. XML. 7. Caminhos mais curtos. 8. Métricas. I. Staa, Arndt von. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

À minha esposa,
Julia Ribeiro Ferreira

Agradecimentos

Ao meu orientador, Arndt von Staa, pela orientação, paciência e grande entendimento sobre os objetivos e execução do trabalho.

Ao CNPq e à PUC-Rio pelo financiamento da pesquisa e aos auxílios concedidos sem os quais essa pesquisa seria inviável.

À minha esposa, Julia Ribeiro Ferreira, pela compreensão perante as noites de pesquisa na frente do computador.

À minha família que me estimulou a realizar o programa de mestrado.

A todos os professores e funcionários da PUC-Rio que contribuíram de forma direta ou indireta para este trabalho.

Resumo

Miranda, Bernardo Evangelho; Staa, Arndt von (Orientador). **Conceitos Centrais e Componentização de Diagramas de Classe UML representados em Grafo**. Rio de Janeiro, 2013. 101p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

O objetivo do trabalho é o desenvolvimento de uma aplicação web capaz de realizar diferentes análises de diagramas UML. Uma das características da ferramenta é a compatibilidade com outras ferramentas de edição de diagramas UML. Para isto, foi implementado um parser genérico XMI que importa diagramas de classe e gera um grafo orientado equivalente. Com a posse do grafo, são realizados estudos capazes de adicionar informações extras aos diagramas. O primeiro estudo é a identificação de quais classes compõem o *core concept* (classes de alta importância para o sistema). Outro estudo é a clusterização deste grafo a fim de agrupar classes em propostas de componentes. Por fim, são discutidas estimativas de importância, do grau de coesão, além de métricas de acoplamento e tamanho do diagrama como um todo e dos componentes gerados.

Palavras-chave

Componentes; Diagramas de Classe; Conceitos Centrais; Clusterização; XMI; Caminhos Mais Curtos; Métricas.

Abstract

Miranda, Bernardo Evangelho; Staa, Arndt von (Advisor). **Core Concepts and Componentization of UML class diagrams represented in graph.** Rio de Janeiro, 2013. 101p. MSc. Dissertation – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

The goal of this dissertation is to develop a web application that explores different analysis of UML diagrams. One of the main features of this tool is the compatibility with other tools that create UML diagrams. In order to do that, we implemented a generic XMI parser that import class diagrams and provides as result a directed graph. With this graph in hand, we conducted studies that provide extra information regarding this diagram. The first study is the identification of the classes that can be considered “core concepts” (defining the important classes of the system). The other study performs graph clustering, aiming to create groups of classes, making it possible to generate components. Finally, we discuss estimates of importance and degree of cohesion, as well as size and coupling metrics of the diagram as a whole and the generated components.

Keywords

Components; Class Diagrams; Core Concepts; Clustering; XMI; Shortest Paths; Metrics.

Sumário

1	Introdução	14
1.1.	Motivação	15
1.2.	Especificação do Problema	16
1.2.1.	Enunciado	16
1.2.2.	Proposta	17
1.3.	Organização do Trabalho	19
2	Aplicabilidade	21
2.1.	Processo de Software	21
2.2.	Rational Unified Process (RUP)	22
2.3.	Scrum	23
3	Referências e Base Técnica	25
3.1.	Detecção de inconsistências extraídas de diagramas UML	25
3.2.	Transformação de modelos em grafo	26
3.3.	Algoritmos de agrupamento	27
3.4.	Obtenção dos core concepts	27
3.5.	Definições de métricas de diagramas UML	28
3.6.	Ferramentas de Análise UML	31
3.7.	Estado da arte	34
4	Conceitos Chaves	39
4.1.	UML	39
4.1.1.	Diagramas de classe	39
4.2.	XMI	41
4.2.1.	XML Schema e XML Namespace	42
4.2.2.	Estrutura do Documento	43
4.2.3.	Identificadores dos elementos	43

4.2.4. Representação do Modelo de Diagrama de Classes	43
4.3. Caminhos Mais Curtos	44
4.3.1. Guardar o Caminho	47
4.4. Clusterização	48
5 Abordagem	51
5.1. Grafo orientado	52
5.1.1. Parser Genérico do XMI	53
5.1.1.1. Parser XML	53
5.1.1.2. Arquivo XMI	54
5.1.1.3. Arquivo de Metamodelo	55
5.1.1.4. Arquivo de transformação	56
5.1.1.5. Leitura do Arquivo no <i>AnalizadorUML</i>	57
5.2. Core Concepts	60
5.3. Componentes	60
5.4. Estimativas de complexidade	62
5.5. Estimativas de coesão	62
5.6. Métricas de Tamanho	63
5.7. Métricas de Acoplamento	64
5.8. Atribuições de classes a desenvolvedores	65
5.9. Arquitetura e Uso	66
5.9.1. Configuração	66
5.9.2. Projeto	66
5.9.3. Banco de Dados	68
6 Aplicação	71
6.1. Base Amostral	71
6.1.1. Modelos de Design Patterns	72
6.1.1.1. Abstract Factory e Strategy	72
6.1.1.2. Bridge, Mediator e Chain of Responsibility	74
6.1.1.3. Command, Adapter e Proxy	76
6.1.1.4. Factory Method, Facade e Visitor	78
6.1.1.5. Observer, Flyweight e Adapter	80

6.1.1.6. Resultados Obtidos pela Ferramenta	82
6.1.2. Modelos Complexos	84
6.1.3. Modelos de Maior Nível de Abstração	91
6.1.3.1. Projeto Amplo	91
6.1.3.2. Talisman	92
6.1.3.3. Resultados Obtidos pela Ferramenta	93
6.2. Reflexão sobre os Pesos Atribuídos	94
7 Conclusão	96
7.1. Trabalhos Futuros	96
7.2. Resultados Parciais	97
7.3. Resultados Conquistados	98
8 Referências Bibliográficas	99

Lista de Figuras

Figura 1 - Estrutura do <i>AnalizadorUML</i> .	19
Figura 2 - Ferramentas de uma workbench.	35
Figura 3 - Metamodelo dos diagramas de classe.	40
Figura 4 - XML de declaração do namespace e da classe.	44
Figura 5 - Divisão do caminho p em dois caminhos distintos.	46
Figura 6 - Visão recursiva Floyd-Warshall.	46
Figura 7 - Pseudo-algoritmo Floyd Warshall.	46
Figura 8 - Visão recursiva Floyd Warshall guardando caminho.	47
Figura 9 - Pseudo-algoritmo k-means.	50
Figura 10 - Grafo orientado gerado na leitura do diagrama representado pelo XML.	52
Figura 11 - Trecho de código do arquivo XML.	54
Figura 12 - Trecho de código do arquivo de metamodelo.	56
Figura 13 - Trecho de código do arquivo XMI-Metamodelo.	57
Figura 14 - Modelo da composição <i>Abstract Factory</i> e <i>Strategy</i> .	73
Figura 15 - Modelo da composição <i>Bridge</i> , <i>Mediator</i> e <i>Chain of Responsibility</i> .	75
Figura 16 - Modelo da composição <i>Command</i> , <i>Adapter</i> e <i>Proxy</i> .	77
Figura 17 - Modelo da composição <i>Factory Method</i> , <i>Facade</i> e <i>Visitor</i> .	79
Figura 18 - Modelo da composição <i>Observer</i> , <i>Flyweight</i> e <i>Adapter</i> .	81
Figura 19 - Diagrama de classes do projeto amplo.	91
Figura 20 - Diagrama de classes de tipos de dados da ferramenta Talisman.	93

Lista de Tabelas

Tabela 1 - Métricas de tamanho do UML Class Diagram Metrics Tool.	32
Tabela 2 - Métricas de Acoplamento do UML Class Diagram Metrics Tool.	33
Tabela 3 - Métricas de Herança do UML Class Diagram Metrics Tool.	33
Tabela 4 - Ferramentas <i>Open Source</i> .	37
Tabela 5: Ferramentas Comerciais.	37
Tabela 6 - Tabela de Grau de Importância de modelos baseados em <i>design patterns</i> .	83
Tabela 7 - Tabela de componentes de modelos baseados em <i>design patterns</i> .	83
Tabela 8 - Tipos de relacionamento e pesos gerados pelos modelos de <i>design patterns</i> .	84
Tabela 9 - Métricas de tamanho e estimativas de complexidade dos projetos e a comparação com os <i>design patterns</i> .	86
Tabela 10 - Pesos atribuídos finais para cada tipo de relacionamento.	89
Tabela 11 - Graus de importância das classes de projetos abstratos.	94
Tabela 12 - Componentes gerados dos projetos abstratos.	94

*Qual seria a sua idade
se você não soubesse quantos anos você tem?*

Confúcio