Cláudio Nogueira Sant'Anna

# On the Modularity of Aspect-Oriented Design: A Concern-Driven Measurement Approach

**DOCTORAL THESIS**

**COMPUTER SCIENCE DEPARTMENT**

Graduate Program in Computer Science

Rio de Janeiro
April 2008

PUC-Rio - Certificação Digital Nº 0410867/CA

Pontifícia Universidade Católica
do Rio de Janeiro

**Cláudio Nogueira Sant'Anna**

# On the Modularity of Aspect-Oriented Design: A Concern-Driven Measurement Approach

**Doctoral Thesis**

Thesis presented to the Graduate Program in Computer Science of the Pontifical Catholic University of Rio de Janeiro in partial fulfillment of the requirements for the degree of Doctor in Computer Science.

Supervisors: Carlos José Pereira de Lucena
Alessandro Fabricio Garcia

Rio de Janeiro, April 2008

PUC-Rio - Certificação Digital Nº 0410867/CA

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

**Cláudio Nogueira Sant'Anna**

# On the Modularity of Aspect-Oriented Design: A Concern-Driven Measurement Approach

Thesis presented to the Graduate Program in Computer Science of the Pontifical Catholic University of Rio de Janeiro in partial fulfillment of the requirements for the degree of Doctor in Computer Science. Approved by the following Examination Committee.

**Carlos José Pereira de Lucena**
Supervisor
PUC-Rio

**Alessandro Fabricio Garcia**
Co-supervisor
Lancaster University

**Arndt von Staa**
PUC-Rio

**Julio Cesar Sampaio do Prado Leite**
PUC-Rio

**Itana Maria de Souza Gimenes**
Universidade Estadual de Maringá

**Paulo César Masiero**
Universidade de São Paulo

**José Eugenio Leal**
Graduate Programs' Coordinator for the Center of Science and Technology - PUC-Rio

Rio de Janeiro, April 11th, 2008

PUC-Rio - Certificação Digital Nº 0410867/CA

All rights reserved. Copying portions or the entirety of the work is prohibited, except as otherwise permitted by the university, the author and the supervisors.

**Cláudio Nogueira Sant'Anna**

He received his BSc in Computer Science from the Universidade Federal da Bahia (UFBA) in 1997. He worked as a software developer from 1997 to 2001. He received his MSc degree in Computer Science from PUC-Rio in 2002.

PUC-Rio - Certificação Digital Nº 0410867/CA

Ficha Catalográfica

Sant'Anna, Cláudio Nogueira

On the modularity of aspect-oriented design: a concern-driven measurement approach / Cláudio Nogueira Sant'Anna ; orientadores: Carlos José Pereira de Lucena, Alessandro Fabricio Garcia. – 2008.

253 f. : il. ; 30 cm

Tese (Doutorado em Informática)–Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2008.

Inclui bibliografia

1. Informática – Teses. 2. Design de software. 3. Modularidade. 4. Arquitetura de software. 5. Métricas de software. 6. Desenvolvimento de software orientado a aspectos. I. Lucena, Carlos José Pereira de II. Garcia, Alessandro Fabrício. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

PUC-Rio - Certificação Digital Nº 0410867/CA

To my parents Moema and Roberto

## Acknowledgments

I consider myself fortunate and privileged to have Dr. Alessandro Garcia as one of my supervisors. I am deeply indebted for all the essential guidance and permanent encouragement that I received from him. I am very thankful for the enthusiasm with which he motivated and helped me to go on. This work would not have been possible without his support and friendship. Furthermore, Alessandro gave me the opportunity to work at the Computing Department at Lancaster University during one year of my PhD. To Alessandro, my deepest gratitude.

I am honored and especially grateful to have Prof. Carlos Lucena as one of my supervisors. I would like to thank him for giving me freedom and shaping my path to research by guiding me with his extensive knowledge. I owe him so much for every occasion in which he trusted me fully. At the same time, I want to thank him for the whole financial support that made possible my regular participation in a number of conferences.

I would like to thank all my colleagues and professors from the Computer Science Department at PUC-Rio for providing a stimulating work environment. My thankful admiration goes to Professor Arndt von Staa whose highly competent and constructive criticism on my work will always remain very precious to me. Besides, the seed of this work was planted during his course about software metrics. I also thank Professor Julio Leite who taught me two very interesting courses – on software evolution and software requirements – which enlarged my vision on software engineering and indirectly contributed to this thesis.

I've been particularly lucky to have collaborated with a number of research colleagues who contributed to this thesis in different ways. Individually, I want to thank: (i) Christina Chavez, for all the good pieces of advice and for introducing me to the software engineering group at PUC-Rio; (ii) Uirá Kulesza, for the insightful discussions and comments, and for being a great friend; (iii) Eduardo Figueiredo, for all the fruitful discussions and for the time we spent together measuring and analyzing measurement results; (iv) Thaís Batista, for all the

PUC-Rio - Certificação Digital Nº 0410867/CA

shared knowledge and experience on software architecture; (v) Miguel da Silva, for the time we spent together working on the implementation of COMET; and (vi) Cidiane and Nélio, for their contributions on the empirical studies used in this thesis. It was a pleasure to work with them on a number of papers.

This work would not have been possible without the stimulating environment in the Software Engineering Laboratory (LES) at PUC-Rio. I am very grateful for the privilege of having worked together with all my colleagues from LES. It is difficult to name them all but I will try. My thanks to Akeo, Anarosa, Andrew, Camila, Carol, Choren, Cidiane, Daflon, Dani Brauner, Elder, Espinha, Felipe, Firmo, Guga, Ingrid, Karla, Küsel, Léo Cunha, LF, Lyrene, Maíra, Miriam, Pádua, Rodnei, Roberta, Rodrigo Alagoano, Rodrigo Gaúcho and Viviane. I also want to express my sincere thanks to Vera for her friendship and for the kind way in which she always succeeded in solving my administrative issues.

This research work was partially conducted at the Computing Department at Lancaster University. I want to thank my colleagues from Lancaster for making my stay there an extraordinary experience. Thanks to Ambra, Américo, Awais, Chiara, Eduardo, Fabiano, Ivone, Jan Wloca, Luca, Neil, Nélio, Nelis, Nelly, Paula, Phil, Roberta, Safoora, Thiago and Vander. During my time in Lancaster, I also had the opportunity to collaborate with researchers from the University of Malaga in the context of the AOSD-Europe project. Thanks to Lidia Fuentes, Mónica Pinto and Nadia Gámez.

Very special thanks to all my friends from Salvador, in particular, Daniel, Diva, Eguinhas, Igor, Jupião, Karina, Kiko, Léo and Queiroz, that never forgot me despite being so far away for such a long time.

I am also thankful to the members of my examination committee, Arndt von Staa, Itana Gimenes, Julio Leite and Paulo Masiero, who have generously contributed their time and expertise.

My doctoral studies have been financially supported by CNPq and CAPES. The funders of this work have my gratitude.

PUC-Rio - Certificação Digital Nº 0410867/CA

Finally, I would like to warmly thank my wonderful parents, Moema and Roberto, for all their love and mental support, for believing in me since day one and for giving me all the opportunities in life.

PUC-Rio - Certificação Digital Nº 0410867/CA

## Abstract

Sant'Anna, Cláudio Nogueira; Lucena, Carlos José Pereira de; Garcia, Alessandro Fabricio. **On the Modularity of Aspect-Oriented Design: A Concern-Driven Measurement Approach.** Rio de Janeiro, 2008. 253p. Doctoral Thesis - Computer Science Department, Pontifical Catholic University of Rio de Janeiro.

Several modularity problems in software designs are related to the inadequate modularization of key broadly-scoped concerns, such as exception handling, distribution, and persistence. However, most of the current quantitative assessment approaches are not sensitive to concerns that drive the design, thereby leading to a number of shortcomings in the modularity evaluation process. Therefore, there is a need for measurement approaches that support a more effective identification of modularity anomalies related to crosscutting concerns. Also, this necessity becomes more apparent in an age that a number of different forms of design decompositions, such as aspect-oriented software development, are emerging. In this context, this thesis aims at investigating a novel approach for quantitative modularity assessment of software design by promoting the concept of concern as a measurement abstraction. Our concern-driven measurement approach encompasses a set of mechanisms for assessing software modularity from architectural to detailed design. The proposed concern-sensitive approach includes: (i) a suite of architectural metrics, (ii) a suite of detailed design metrics, (iii) a suite of design heuristic rules for supporting the interpretation of metrics in meaningful ways, and (iv) a tool, called COMET, that supports both concern-driven notation and measurement of architectural designs. We evaluated the usefulness of our concern-oriented measurement technique in a series of empirical studies, comparing the modularity of conventional and aspect-oriented software design.

## Keywords

Software design, modularity, software architecture, software metrics, aspect-oriented software development

PUC-Rio - Certificação Digital Nº 0410867/CA

## Resumo

Sant'Anna, Cláudio Nogueira; Lucena, Carlos José Pereira de; Garcia, Alessandro Fabricio. **Modularidade de Design Orientado a Aspectos: Uma Abordagem de Medição Dirigida por Interesses.** Rio de Janeiro, 2008. 253p. Tese de Doutorado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Muitos problemas de modularidade de design de software estão relacionados à modularização inadequada de interesses importantes e que têm impacto sistêmico no design, tais como tratamento de exceção, distribuição e persistência. No entanto, a maioria das abordagens atuais de avaliação quantitativas não leva em conta os interesses que guiam o design, o que acaba fazendo com que o processo de avaliação de modularidade se torne deficiente. Portanto, existe a necessidade de abordagens de medição que promovam uma identificação mais efetiva dos problemas de modularidade relacionados a interesses transversais. Além disso, essa necessidade se torna ainda mais evidente à medida que surgem novas formas de decomposição de design, tais como desenvolvimento de software orientado a aspectos. Nesse contexto, essa tese tem o objetivo de definir e investigar uma nova abordagem de avaliação quantitativa de modularidade de design de software que promove o conceito de interesse a uma abstração de medição. Esse trabalho define uma abordagem de medição dirigida por interesses que inclui um conjunto de mecanismos para a avaliação de modularidade de software desde o design arquitetural até o design detalhado. A abordagem sensível a interesses proposta é composta por: (i) um conjunto de métricas arquiteturais, (ii) um conjunto de métricas de design detalhado, (iii) um conjunto de regras heurísticas de design que dão apoio a interpretação das métricas, e (iv) uma ferramenta, chamada de COMET, que dá apoio tanto à notação quanto à medição dirigida por interesses de design arquitetural. A utilidade da técnica de medição dirigida por interesses proposta foi avaliada em uma série de estudos empíricos, onde a modularidade de designs convencionais e orientados a aspectos foram comparados.

## Palavras-chave

Design de software, modularidade, arquitetura de software, métricas de software, desenvolvimento de software orientado a aspectos

PUC-Rio - Certificação Digital Nº 0410867/CA

## Table of Contents


1 Introduction 24
1.1. Problem Statement 26
1.2. Limitation of Conventional Measurement Approaches 27
1.2.1. Inaccuracy on Identifying Non-localized Concerns 28
1.2.2. Inaccuracy on Identifying Dependence between Concerns 29
1.2.3. Inaccuracy on Identifying Instabilities 30
1.2.4. The Tyranny of Dominant Modularity Attributes 31
1.3. Proposed Solution 31
1.4. Empirical Evaluation 34
1.5. Thesis Outline 35

2 Design Modularity Measurement 36
2.1. Modularity Definition 36
2.2. Modularity Attributes 37
2.3. Conventional Architecture Metrics 38
2.3.1. Metrics by Martin 39
2.3.2. Metrics by Briand et al. 40
2.3.3. Metrics by Lung & Kalaichelvan 41
2.4. Conventional Detailed Design Metrics 42
2.5. Design Heuristics Rules 45

3 Aspect-Oriented Software Development 48
3.1. Aspect-Oriented Programming 48
3.2. Aspect-Oriented Architecture Design 52
3.2.1. AOGA 55
3.2.2. AO Visual Notation 57
3.3. Aspect-Oriented Metrics 59
3.3.1. Metrics by Ceccato & Tonella 59
3.3.2. Metrics by Sant'Anna et al. 60


PUC-Rio - Certificação Digital Nº 0410867/CA

3.3.3. Metrics by Zhao and Xu 61
3.3.4. Connection between Aspects and Classes 61

4 Concern-Driven Metrics 62
4.1. Classification of Software Concerns 62
4.2. Concern Representation 64
4.2.1. Architectural Design 66
4.2.2. Architectural Concern 69
4.3. Suite of Concern-Driven Architecture Metrics 73
4.3.1. Metrics for Concern Diffusion 76
4.3.2. Metrics for Interaction between Concerns 78
4.3.3. Concern-based Cohesion 80
4.3.4. Concern-Sensitive Coupling Metric 81
4.3.5. Number of Concern Interfaces Metric 82
4.3.6. Metrics for Coupling and Interface Complexity 83
4.4. Classification of the Metrics 86
4.4.1. Measurement Framework Criteria 87
4.5. Related Work 90
4.5.1. Metrics by Sant'Anna et al. 91
4.5.2. Metrics by Ducasse et al. 93
4.5.3. Metrics by Wong et al. 94
4.5.4. Metrics by Eaddy et al. 95

5 Concern-Driven Design Heuristic Rules 97
5.1. Limitation of Conventional Heuristic Rules 98
5.2. Concern Representation at Detailed Design 100
5.2.1. Detailed Design 101
5.3. Concern-Driven Metrics for Detailed Design 103
5.3.1. Concern Diffusion 105
5.3.2. Interaction between Concerns 106
5.3.3. Concern-based Cohesion 107
5.3.4. Concern-Sensitive Coupling 108
5.3.5. Concern-Sensitive Size 110
5.4. Concern-Driven Design Heuristic Rules 111

PUC-Rio - Certificação Digital Nº 0410867/CA

5.4.1. Crosscutting Concern Analysis 112
5.4.2. Octopus and Black Sheep 116
5.4.3. Concern-Aware Bad Smells 117
5.4.4. The Issue of Threshold Values 119

6 Tool Support for Concern-Driven Measurement 121
6.1. Limitation of Related Work 122
6.2. The Concern-Oriented Measurement Tool 123
6.2.1. User Interface 126
6.2.2. Extracting an Architecture Specification 131
6.2.3. Managing the Architecture Model and Assigning Concerns 132
6.2.4. Applying metrics and heuristic rules 135
6.3. Concern Templates 136
6.3.1. Composition Rules 139
6.3.2. Using Concern Templates 142
6.3.3. Related Work 150

7 Evaluation of the Architectural Metrics 152
7.1. Study Procedures 153
7.2. AspectT and MobiGrid study 155
7.2.1. The AspectT Architectures 157
7.2.2. The MobiGrid Architectures 161
7.2.3. Results 164
7.2.4. Discussion 171
7.3. Health Watcher study 173
7.3.1. The Health Watcher Architectures 175
7.3.2. Results and Discussion 178
7.4. Mobile Media Study 186
7.4.1. Mobile Media Architectures 189
7.4.2. Results 193
7.4.3. Discussion 198
7.5. Study Constraints 200

8 Evaluation of the Detailed Design Metrics and Heuristics 202

PUC-Rio - Certificação Digital Nº 0410867/CA

8.1. Design Heuristic Rules Study 202
8.1.1. Target Systems 203
8.1.2. Accuracy of the Heuristic Rules 205
8.1.3. Detection of Specific Design Flaws 210
8.1.4. Solving Measurement Shortcomings 212
8.2. Detailed Design Metrics Study 214
8.2.1. Study Format and Procedures 215
8.2.2. Hypothesis and Results 217
8.3. Study Constraints 219

9 Conclusions 221
9.1. Contributions 222
9.2. Future Work 224

10 References 227

Appendix A – Mobile Media Architecture Description 237

Appendix B – Detailed Design Metrics Study: Form and Metrics Values 249

PUC-Rio - Certificação Digital Nº 0410867/CA

## List of Figures


Figure 1: Architecture of the Health Watcher system 28
Figure 2: Design to illustrate Martin's coupling metrics 40
Figure 3: Example of an aspect in AspectJ 51
Figure 4: Java (left side) and AspectJ (right side) version of the same program. 52
Figure 5: AOGA architecture elements 56
Figure 6: AO Visual Notation: Aspectual Connectors 57
Figure 7: Simpler notation for aspectual connectors 58
Figure 8: Simplified representation of the Health Watcher software architecture 64
Figure 9: Mapping between concerns and design elements 65
Figure 10: Aspect-oriented design alternative of Health Watcher architecture 68
Figure 11: Simplified representation of the Health Watcher system architecture 76
Figure 12: Transition points 92
Figure 13: Distributed Map 93
Figure 14: Design slice of an OpenOrb-compliant middleware system 99
Figure 15: Observer and Factory Method patterns used in the design of an OpenOrb-compliant middleware system 104
Figure 16: Concern classification 113
Figure 17: Design heuristic rules for crosscutting concern analysis 114
Figure 18: Design heuristics rules for Black Sheep and Octopus 117
Figure 19: Heuristic rules for detecting bad smells 118
Figure 20: Overview of COMET's modules 124
Figure 21: Architecture measurement meta-model 124
Figure 22: Heuristic rules supported by COMET 127
Figure 23: COMET Views 128
Figure 24: The Projects View 129


PUC-Rio - Certificação Digital Nº 0410867/CA

Figure 25: The Architecture View for the Health Watcher system 130
Figure 26: The Properties View for the transactionExceptionalEvent operation 130
Figure 27: The Properties View for the saveEntity interface 130
Figure 28: The Concern Model View for the Health Watcher system. 131
Figure 29: The Properties View for the Distribution concern 131
Figure 30: Wizard for creating a new project 132
Figure 31: Adding new architecture element 133
Figure 32: Adding new concerns 133
Figure 33: Selecting concerns related to an architecture element 134
Figure 34: Selecting architecture elements related to a concern 135
Figure 35: Metrics View 136
Figure 36: Heuristic Rules View 136
Figure 37: Concern Template: Distribution 138
Figure 38: BNF description of language for low-level composition rules 140
Figure 39: Add primitive 141
Figure 40: Connect primitive 141
Figure 41: Health Watcher Architecture 143
Figure 42: Concern template for the persistence concern 145
Figure 43: Low-level composition rules (continuation of the persistence concern template shown in Figure 42) 146
Figure 44: Concern template for the exception handling concern 148
Figure 45: Low-level composition rules (continuation of the exception handling concern template shown in Figure 44) 149
Figure 46: The AspectT architecture 159
Figure 47: Details of AspectT interfaces 160
Figure 48: Non-AO mediator-based architecture equivalent to AspectT architecture 162
Figure 49: The non-AO MobiGrid architecture 163
Figure 50: The AO MobiGrid architecture 163
Figure 51: Non-AO architecture of the Health Watcher system 176
Figure 52: Aspect-oriented architecture of the Health Watcher system 177

PUC-Rio - Certificação Digital Nº 0410867/CA

Figure 53: Business_Rules component: Number of Concern Interfaces and Concern-Sensitive Coupling measures 184
Figure 54: Simplified Mobile Media feature model 187
Figure 55: Non-AO architecture of the Mobile Media product line 191
Figure 56: AO architecture of the Mobile Media product line 192
Figure 57: Concern Diffusion and Interaction between Concerns metrics for favorite feature 194
Figure 58: Concern Diffusion and Component-level Interlacing between Concerns metrics for the exception handling concern 196
Figure 59: Concern Diffusion and Interface-level Interlacing between Concerns metrics for the label media mandatory feature 197
Figure 60: Concern Diffusion over Architectural Operations metric for the favorite feature 200
Figure 61: Observer and Factory Method patterns used in the design of an OpenOrb-compliant middleware system 213
Figure 62: Concern-driven metrics for Façade and Singleton 214
Figure 63: Non-AO architecture of the Mobile Media product line – Releases 1 and 2 237
Figure 64: Non-AO architecture of the Mobile Media product line – Releases 3 and 4 238
Figure 65: Non-AO architecture of the Mobile Media product line – Release 5 239
Figure 66: Non-AO architecture of the Mobile Media product line – Release 6 240
Figure 67: Non-AO architecture of the Mobile Media product line – Release 7 241
Figure 68: Non-AO architecture of the Mobile Media product line – Release 8 242
Figure 69: AO architecture of the Mobile Media product line – Release 2 243
Figure 70: AO architecture of the Mobile Media product line – Release 3 244

PUC-Rio - Certificação Digital Nº 0410867/CA

Figure 71: AO architecture of the Mobile Media product line – Release 4 244
Figure 72: AO architecture of the Mobile Media product line – Release 5 245
Figure 73: AO architecture of the Mobile Media product line – Release 6 246
Figure 74: AO architecture of the Mobile Media product line – Release 7 247
Figure 75: AO architecture of the Mobile Media product line – Release 8 248

PUC-Rio - Certificação Digital Nº 0410867/CA

## List of Tables

Table 1: Summary of the suite of concern-driven architectural metrics 75
Table 2: Classification of our architecture metrics according to Figueiredo and colleagues' measurement framework (Figueiredo et al., 2008a) 90
Table 3: Classification of related metrics according to Figueiredo and colleagues' measurement framework (Figueiredo et al., 2008a) 96
Table 4: Summary of the suite of concern-driven detailed design metrics 105
Table 5: Conventional metrics used in the definition of the heuristic rules 111
Table 6: Primitives for defining low-level composition rules 142
Table 7: AspectT and MobiGrid study configuration 158
Table 8: AspectT: concern diffusion measures 165
Table 9: MobiGrid architectures: concern diffusion measures 167
Table 10: AspectT architectures: coupling and cohesion measures 168
Table 11: MobiGrid architectures: coupling and cohesion measures 168
Table 12: AspectT architectures: interface complexity measures 170
Table 13: MobiGrid architectures: interface complexity measures 170
Table 14: Health Watcher study configuration 175
Table 15: Health Watcher architectural concerns 177
Table 16: Health Watcher: concern diffusion measures 179
Table 17: Health Watcher: interaction between concerns measures 180
Table 18: Health Watcher: concern-based cohesion measures 182
Table 19: Health Watcher: interface complexity measures 183
Table 20: Summary of the change scenarios 186
Table 21: Summary of the evolution scenarios implemented in Mobile Media 188
Table 22: Mobile Media study configuration 189
Table 23: Systems and concerns used in the evaluation study 204

PUC-Rio - Certificação Digital Nº 0410867/CA

Table 24: Comparing the specialists opinion with the rules outcomes 206
Table 25: Results of the heuristics application in the object-oriented version of the systems 207
Table 26: Results of the heuristics application in the aspect-oriented version of the systems 209
Table 27: Statistics about the application of the heuristic rules 209
Table 28: Concern-driven vs. Conventional heuristic rules: statistics about the application of the heuristic rules for detecting bed smells 211
Table 29: Results - identification of Divergent Change and Shotgun Surgery 217
Table 30: Results of the metrics for the Health Watcher design 249
Table 31: Results of the metrics for the Health Watcher design 250

PUC-Rio - Certificação Digital Nº 0410867/CA

## List of Acronyms and Abbreviations

ADL - Architecture Description Language

AFI - Architectural Fan-in

AFO - Architectural Fan-out

AMT - Aspect Mining Tool

AO - Aspect-oriented

AOGA - Aspect-Oriented Generative Approach

AOP - Aspect-Oriented Programming

AOSD - Aspect-oriented software development

ATAM - Architecture Tradeoff Analysis Method

CAE - Coupling on Advice Execution

CBC - Coupling between Components

CBO - Coupling between Object Classes

CC - Changing Classes

C&C - Component-and-connector

CDA - Crosscutting Degree of an Aspect

CDAC - Concern Diffusion over Architectural Components

CDAI - Concern Diffusion over Architectural Interfaces

CDAO - Concern Diffusion over Architectural Operations

CDC - Concern Diffusion over Components

CDLOC - Concern Diffusion over Lines of Code

CDO - Concern Diffusion over Operations

CFA - Coupling on Field Access

CIBC - Component-level Interlacing Between Concerns

CIM - Coupling on Intercepted Modules

CM - Changing Method

CMC - Coupling on Method Call

CME - Concern Manipulation Environment

COF - Coupling Factor

COMET - Concern-Oriented Measurement Tool

PUC-Rio - Certificação Digital Nº 0410867/CA

CONC - Concentration

CSC - Concern-Sensitive Coupling

DAC - Data Abstraction Coupling

DAOP - Dynamic Aspect-Oriented Platform

DEDI - Dedication

DOF - Degree of Focus

DOS - Degree of Scattering

FEAT - Feature Exploration and Analysis Tool

GUI - Graphical User Interface

ICP - Information-flow-based Coupling

ICSC - Intra-component Concern-Sensitive Coupling

IIBC - Interface-level Interlacing Between Concerns

LCC - Lack of Concern-based Cohesion

LCO - Lack of Cohesion in Operations

LCOM - Lack of Cohesion in Methods

LCOO - Lack of Cohesion in Operations

MAS - Multi-agent System

MPC - Message Passing Coupling

MVC - Model-View-Controller

NC - Number of Components

NCA - Number of Concern Operations

NCI - Number of Concern Interfaces

NCO - Number of Concern Operations

NI - Number of Interfaces

NO - Number of Operations

NOA - Number of Attributes

NOO - Number of Operations

OMG - Object Management Group

OO - Object-oriented

OOBC - Operation-level Overlapping Between Concerns

RFC - Response for a Class

RFM - Response for a Module

SAAM - Software Architecture Analysis Method

PUC-Rio - Certificação Digital Nº 0410867/CA

UML - Unified Modeling Language

WMC - Weighted Methods per Class

PUC-Rio - Certificação Digital Nº 0410867/CA

PUC-Rio - Certificação Digital Nº 0410867/CA

*To measure is to know*
Lord Kelvin, *n.d*