

# Formalization of a Reverse Engineering Strategy

Gustavo Villavicencio

October 27, 2002

Submitted to the  
Department of Computer Sciences  
School of Physics, Mathematics and Natural Sciences  
of the National University of San Luis  
in fulfillment with the requirements  
for the degree of Master in Software Engineering

Copyright © 2002 Gustavo Villavicencio

Thanks, Mom!

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Structure of thesis - Thesis Outline . . . . .	3
<b>2</b>	<b>Background</b>	<b>5</b>
2.1	Reverse Engineering . . . . .	5
2.2	Program slicing . . . . .	6
2.3	Theoretic Frame . . . . .	8
2.3.1	Formal methods . . . . .	8
2.3.2	Functional programming . . . . .	10
2.3.3	Denotational / Operational Semantics . . . . .	13
2.4	Summary . . . . .	15
<b>3</b>	<b>Algebra of Programming</b>	<b>17</b>
3.1	Basic concepts . . . . .	17
3.2	Products toolbox . . . . .	19
3.3	Co-products toolbox . . . . .	21
3.4	Conditionals . . . . .	23
3.5	Recursion . . . . .	26
3.6	Anas, catas and hylomorphisms: An overview . . . . .	29
3.7	Monads . . . . .	33

<b>4 Formal Reverse Engineering</b>	<b>35</b>
4.1 Approaches to Formal Reverse Engineering . . . . .	35
4.1.1 Approach I . . . . .	35
4.1.2 Approach II . . . . .	37
4.2 Analysis . . . . .	44
4.3 Summary . . . . .	45
<b>5 Formalization of a Strategy for Reverse Engineering</b>	<b>47</b>
5.1 Introduction . . . . .	47
5.2 Scope . . . . .	47
5.3 Overview . . . . .	47
5.4 The <b>RPC</b> process . . . . .	48
5.4.1 Non terminate situations . . . . .	50
5.4.2 Accumulation parameter introduction . . . . .	52
5.5 Conditioned program slicing . . . . .	52
5.6 The role of algebra . . . . .	52
5.7 Case Studies . . . . .	54
5.8 Case study I . . . . .	54
5.9 Case Study II: Monads. . . . .	62
5.10 Case study III: Conditional slicing in the reverse calculation process . . . . .	67
5.11 Evaluation . . . . .	78
5.12 Summary . . . . .	78
<b>6 Conclusions and Future Work</b>	<b>79</b>
<b>A Appendix A: Source Code Examples</b>	<b>81</b>
A.1 Example I . . . . .	81
A.2 Example II: Monads . . . . .	82
A.3 Example III: Pointers . . . . .	85

<b>B Appendix B: Calculated Slices</b>	<b>89</b>
B.1 Calculated slices from Example I . . . . .	89
B.2 Dos to Unix: Monads . . . . .	91
B.3 Removing string (Pointers) . . . . .	102
B.4 Conditioned slicing . . . . .	107
<b>Bibliography</b>	<b>113</b>

*No hay orden establecido que sea  
duradero sino el que une el  
principio con el fin en un ciclo  
inmutable.*

La consolación de la filosofía

Boecio (480-524 D.C.)