Re-animation of computer programs
Meijer, F.J.

Citation for published version (APA):
Meijer, F. J. (2001). Re-animation of computer programs

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
CONTENTS

Preface

1 THE APPROACH TO THE CODE TRANSFORMATION PROBLEM
1.1 Introduction 11
1.2 A particular code transformation problem 12
1.3 Levels of difficulty 14
1.4 Value of a solution to the problem 16
1.5 Overview 17

2 THE CODE RECONSTRUCTION PROBLEM
2.1 Introduction 19
2.2 The definition of computer and information 19
2.2.1 Main computer characteristic 20
2.2.2 The time aspect 21
2.2.3 The language aspect 21
2.2.3.1 Different language levels 22
2.3 Hardware and software aspects 23
2.3.1 Differences in hardware 24
2.3.2 Differences in the language of the software 25
2.4 Different routes to solve the particular code reconstruction problem 26
2.5 Recapitulation 28

3 HISTORY OF THE PROBLEM
3.1 Introduction 31
3.2 Historical development 31
3.2.1 The problem, relatively old 32
3.3 Hardware 33
3.4 Software 34
3.4.1 A Universal computer language 34
3.4.2 The Program Transferability Study Group 36
3.4.3 Rules to avoid future problems 37
3.4.4 Views held by others 38
3.5 Re-engineering 39
3.6 Attempts to reconstruct machine code programs 40
3.6.1 Code disassembly 40
3.6.2 Dynamic analysis 41
3.7 Summary 41

4 TRANSFORMATION PROBLEMS
4.1 Introduction 43
4.2 The origin of the ambiguity problem 43
4.2.1 Earlier efforts 44
4.2.2 Problems to be solved 44
4.2.3 Problem and control operands 46
4.3 Solutions to the ambiguity problem:
   Static and dynamic analysis 47
4.3.1 Static and dynamic analysis, advantages and disadvantages 47
4.3.2 Using dynamic analysis in MALIN 48
4.3.2.1 Inspecting the generated information 49
4.3.2.2 Determining the type of branch instructions 51
4.3.2.3 Rearranging the data in separate files 52
4.4 Using static analysis in MALIN 52
4.5 The problem of determining event sequences 53
4.6 The code transformation problem 53
4.6.1 Different routes to solve the transformation problem 53
4.6.2 Choosing a route to solve the transformation problem 55
4.7 Summary 56

5 PROBLEMS AND THEIR SOLUTION
5.1 Introduction 57
5.2 Demands to programs to be reconstructed 57
5.3 The distinction between architecturally correct
   and error free 58
5.3.1 Order of errors 61
5.3.2 Implementation in MALIN 62
5.3.3 First order errors and their solution 62
5.3.3.1 The jump instruction defies correct program architecture 65
5.4 Working method 65
5.5 Requirements 66
5.6 Summary 67
6 CODE RECONSTRUCTION AND GENERATION
6.1 Introduction 69
6.2 The design of MALIN 69
6.2.1 The development of MALIN 71
6.2.2 Other utilities of MALIN 73
6.3 Solving branch problems 74
6.4 Testing transformed code 75
6.5 Solving ad hoc problems 76
6.6 Comparison of original and transformed code 78
6.7 Differences in instruction sets 79
6.8 Examples of code reconstruction and generation 79
6.8.1 The disassembler 79
6.8.2 Word Star 80
6.9 Recapitulation 82

7 DISCUSSION
7.1 Introduction 83
7.2 Looking for a justification 84
7.3 The role of the user 85
7.3.1 Addressing experience 86
7.4 The kind of knowledge involved 86
7.5 Recapitulation 88

Appendix
A.1 Computer development 91
A.2 The influence of expanding computer technology 92
A.3 The operating system, a support system 92
A.3.1 The need for an operating system 93
A.3.2 Some personal computers and their operating system 93
A.3.2.1 Differences in operating systems for personal computers 95
A.3.3 Working method of the operating system 95
A.3.3.1 Different levels of operating systems software 96
A.3.3.2 The Graphics Environment Manager 96
A.3.3.3 The Applications Environment Manager 97
A.3.3.4 Direct and indirect function calls 97
A.3.4 The working environment 97
A.4 The computer, a special technical support system 97
A.5 The interface 99
A.5.1 The importance of interfaces 100
A.6 The technical aspects of the process of transformation 100
A.6.1 Modules 101
A.6.2 Computers involved in this study 102
A.7 The process of transformation 102
A.7.1 Emulation of branches 103
A.7.2 A software monitor 103
A.7.3 Possible adaptations of the transformed program 107
A.8 Requirements to be posed to users of MALIN 107
A.9 Technical requirements 108
A.10 Flowcharts 109
A.11 Number of executed 68000 instructions 111
A.12 Number of runtime instructions for transformed code 117
A.13 Z-80 instruction set and transformation for the 68000 processor 119

Dutch summary 127

Bibliography 139

Index 159

MALIN Manual

The listings