Gathering evidence: Model-driven software engineering in automated digital forensics
van den Bos, J.

Citation for published version (APA):

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 ix

I Overview and Analysis 1

1 Introduction 3

1.1 Automated Digital Forensics 5
1.2 Model-Driven Software Engineering 10
1.3 Towards Model-Driven Digital Forensics 12
1.4 Research Questions and Perspectives 15
1.5 Software and Technology 17
1.6 Origin of Chapters 18

2 Towards an Engineering Approach to File Carver Construction 19

2.1 Introduction 20
2.2 File Carving Techniques 20
2.3 File Carving Performance 24
2.4 Recoverability Example: GIF 25
2.5 Discussion 28
2.6 Conclusion 30

II Modularity and Efficiency 33

3 Bringing Domain-Specific Languages to Digital Forensics 35

3.1 Introduction 36
3.2 Digital Forensics Challenges 37
3.3 A DSL for Digital Forensics 41
Contents

3.4 Application: Carving .................................................. 44
3.5 Discussion ............................................................. 52
3.6 Related Work .......................................................... 54
3.7 Conclusion .............................................................. 55

4 Domain-Specific Optimization in Digital Forensics .......................... 57
  4.1 Introduction ............................................................ 58
  4.2 Background ........................................................... 59
  4.3 Transforming Derric Models ......................................... 65
  4.4 Evaluation ............................................................. 68
  4.5 Discussion ............................................................ 71
  4.6 Related Work .......................................................... 72
  4.7 Conclusion ............................................................ 73

III Maintainability .......................................................... 75

5 A Case Study in Evidence-Based DSL Evolution .............................. 77
  5.1 Introduction ............................................................ 78
  5.2 Background ........................................................... 79
  5.3 Observing Corrective Maintenance .................................. 80
  5.4 Experiment ............................................................. 81
  5.5 Results ................................................................. 84
  5.6 Analysis ............................................................... 85
  5.7 Discussion ............................................................ 88
  5.8 Conclusion ............................................................ 91

6 Trinity: An IDE for The Matrix ........................................... 93
  6.1 Background ........................................................... 94
  6.2 Trinity ................................................................. 96
  6.3 Implementation ....................................................... 99
  6.4 Related work .......................................................... 100
  6.5 Conclusion and Future Work ....................................... 101

IV Retrospective .................................................................. 103

7 Contributions .................................................................. 105
  7.1 Achieving Separation of Concerns ................................... 105
  7.2 Measuring Runtime Performance Costs ............................. 107
  7.3 Leveraging Model Transformation .................................... 108
  7.4 Evaluating Maintainability ............................................ 109
## Contents

8 **Conclusions**  
8.1 Model-Driven Software Engineering in Practice .......................... 113  
8.2 DERRIC: Applying MDSE in Automated Digital Forensics ............ 114  
8.3 RASCAL: DSL Engineering in Practice ....................................... 114  
8.4 Future Directions ................................................................. 115  

**Bibliography**  

**Summary**  

**Samenvatting**