Undergraduate Topics in Computer Science
Series editor
Ian Mackie

Advisory board
Samson Abramsky, University of Oxford, Oxford, UK
Karin Breitman, Pontificial Catholic University of Rio de Janeiro, Rio de Janeiro, Brazil
Chris Hankin, Imperial College London, London, UK
Dexter Kozen, Cornell University, Ithaca, USA
Andrew Pitts, University of Cambridge, Cambridge, UK
Hanne Riis Nielson, Technical University of Denmark, Kongens Lyngby, Denmark
Steven Skiena, Stony Brook University, Stony Brook, USA
Iain Stewart, University of Durham, Durham, UK

Undergraduate Topics in Computer Science (UTiCS) delivers high-quality instructional content for undergraduates studying in all areas of computing and information science. From core foundational and theoretical material to final-year topics and applications, UTiCS books take a fresh, concise, and modern approach and are ideal for self-study or for a one- or two-semester course. The texts are all authored by established experts in their fields, reviewed by an international advisory board, and contain numerous examples and problems. Many include fully worked solutions.
Autonomic Computing

Principles, Design and Implementation
The autonomic computing initiative—the creation of self-managing systems—was a call to arms from IBM (Paul Horn, 2001) to the industry and academic computing communities and signified the mainstream adoption of such research fields as intelligent fault management.

Although IBM set it in the context of coping with the ever-increasing ‘systems of systems’ complexity and dealing with the total cost of ownership (TCO), researchers expanded this to next generation of computing and communications, for instance, pervasive computing, ubiquitous computing, swarm-based computing, agent-based computing, smart grid, on-demand computing, next-generation Internet, adaptive communications and the latest trends, green and cloud computing. These streams of research have found focus through such research conferences as ICAC, EASe, SEAMS, MUCS and SASO (International Conference on Autonomic Computing, Engineering of Autonomic and Autonomous Systems, Software Engineering for Adaptive and Self-Managing Systems, Managing Ubiquitous Communication and Services and Self-Adaptive and Self-Organizing Systems, respectively). The decade of research also saw the creation of related initiatives such as Autonomic Communications and Organic Computing, as well as communities, such as the IEEE Technical Committee on Autonomous and Autonomic Systems (TCAAS) and the Autonomic Communications Forum (ACF).

Due to the complexity of the challenges that this subject brings to the fore, the Autonomic Computing has to rely on expertise from many fields—software engineering, systems engineering, control theory and AI (artificial intelligence), among others. As such it is hard to ‘nail’ down the field which is possibly the reason there hasn’t emerged a good tutorial text—till now. As with many ‘hot’ research initiatives, the spotlight (and often the funding) moves on, yet the original long-term 2020–2030 needs have not. If anything, they have become more critical. The Software Crisis 2.0 (Fitzgerald 2012) highlighted more than ever the need to have our software self-managing, due to the demand for data from digital natives, coupled with the huge volume of data now generated through ubiquitous mobile devices, sensors and applications.
The success of the initiative has already been indicated by the notable move of ‘autonomicity’ from the previously mentioned conferences and communities to a standard topic in almost all computer- and communications-based conferences and communities. Ironically, the final success of the initiative may be marked as it no longer exists separately as a specialised field but as a standard, invisible and integrated part of our systems and software engineering.

For the autonomic systems research and development to make further leaps and bounds and move convincingly into the next decade to meet the Software Crisis 2.0 and its other longer term goals requires that it move beyond the research labs and PhD programmes to our graduate, undergraduate and CPPD (continuous professional and personal development) courses.

This book marks the enabler for that next stage.

University of Ulster, Northern Ireland
12 December 2012

Roy Sterritt

References

Preface

Autonomic computing seeks to render computing systems as self-managed. In other words, its objective is to enable computer systems to manage themselves so as to minimise the need for human input. Autonomic computing as an approach is guaranteed to change the way software systems are developed. Indeed, this new field is addressing some of the issues resulting from the ever-increasing complexity of software administration and the growing difficulty encountered by software administrators in performing their job effectively.

This book provides a practical perspective on autonomic computing. Implementing self-managed systems remains a true challenge today. Thus, beyond giving necessary explanations about the objectives and interests of autonomic computing, this book goes through the different software engineering techniques that are currently available for organising and developing self-managed software systems. In summary, this book uniquely:

• Provides a structured and comprehensive introduction to autonomic computing with a software engineering perspective, as far as we are aware this is the first book to do so
• Presents highly up-to-date information on techniques implementing self-monitoring, self-knowledge, decision-making and self-adaptation
• Provides a downloadable learning environment and source code that allows students to develop, execute and test autonomic applications at an associated website

Authors have created the aforementioned learning environment and placed it on a web page that will be regularly updated. The environment represents an autonomic pervasive computing application that simulates a digital home. A dedicated development environment has been designed around this; it allows the student to execute autonomic code in a runtime simulation that provides concrete, visual feedback of the behaviours illustrating what the student has programmed.

This book is aimed at students and practitioners working on software projects where system self-management would redress maintenance complexity and cost issues. Several aspects of this book have been tested in a classroom, which makes this book ideal for a 10-week lecture programme.
Content Level: master student/professional

Keywords: Autonomic computing, Software engineering, Software architectures, Software monitoring, Software adaptation, Knowledge and reasoning

Related Subjects: Software engineering

Authors: Authors are practitioners and recognised researchers in the field. They have published more than 200 publications in international conferences and journals.

Grenoble, France   Philippe Lalanda
London, UK   Julie A. McCann
Paris, France   Ada Diaconescu
Acknowledgments

Many useful discussions with colleagues and students helped in the preparation of this book. The authors would like to thank the following people who, without reward, reviewed and critiqued the text; their comments and suggestions have been invaluable in ensuring the quality of this book:


We are also grateful to Simon Rees of Springer who encouraged us to write the book and provided invaluable assistance in the production of the final copy.

We would also like to thank our colleagues, friends and family for their constant support, encouragement and patience. Ada thanks Mr. Smith for regularly changing the subject. Julie thanks husband Grant and son Carter—you can now use my laptop to watch 1950s cartoons. Philippe thanks his wife, now an expert in autonomic computing, and his two sons, Grégoire and Arthur—experts to come!
# Contents

1 Software Engineering to Autonomic Computing ........................................ 1  
  1.1 Software Complexity ........................................................................ 1  
  1.2 The Software Life Cycle ................................................................. 5  
    1.2.1 Software Development ......................................................... 5  
    1.2.2 Software Deployment .......................................................... 7  
    1.2.3 Software Maintenance ......................................................... 9  
  1.3 Maintenance Challenges ................................................................ 11  
  1.4 Autonomic Computing .................................................................. 14  
  1.5 Book Structure ............................................................................. 17  
  1.6 Key Points .................................................................................... 19  
References ................................................................................................. 20  

2 Autonomic Systems ............................................................................. 23  
  2.1 Autonomic Computing .................................................................... 24  
    2.1.1 Definitions .......................................................................... 24  
    2.1.2 Goals ................................................................................. 26  
    2.1.3 Context .............................................................................. 28  
  2.2 The Origins and Motivations Behind Autonomic Computing ............. 30  
  2.3 Self-* Properties and Expected Qualities ........................................ 34  
    2.3.1 Autonomic Key Features ..................................................... 34  
    2.3.2 Fundamental Self-* Features ............................................. 35  
    2.3.3 Extended Self-* Capabilities .............................................. 37  
  2.4 Benefits, Challenges and Degrees of Autonomy ................................ 39  
    2.4.1 Benefits of Autonomic Computing ...................................... 39  
    2.4.2 Challenges of Autonomic Computing .................................. 40  
    2.4.3 An Incremental Approach to Autonomic Computing .......... 41  
  2.5 Similar Initiatives, Current Status and Relation to Software Engineering .................................................. 43  
    2.5.1 Autonomicity in Existing Systems ...................................... 43  
    2.5.2 Top-Down Versus Bottom-Up Approaches ......................... 43  
    2.5.3 Similar Initiatives ............................................................... 44
3 Sources of Inspiration for Autonomic Computing

3.1 Overview of Influences

3.1.1 Introduction

3.1.2 Natural Systems

3.1.3 Adaptive Computing Systems

3.2 Biology

3.2.1 Overview

3.2.2 Introduction to Biological Nervous Systems

3.2.3 Structure of the Human Nervous System

3.2.4 Function of the Human Nervous System

3.2.5 Reflexes and Autonomic Control Loops

3.2.6 Different Nervous System Architectures and Features

3.2.7 Summary of Inspiration from Nervous Systems

3.2.8 Bio-inspiration Beyond Nervous Systems

3.3 Control Systems

3.3.1 Introduction

3.3.2 Feedback Control

3.3.3 The PID Controller

3.3.4 Oscillations, Overshooting, Damping and Stability

3.3.5 Control and Autonomic Computing

3.4 Artificial Intelligence

3.4.1 Introduction to Intelligence

3.4.2 Introduction to Software Agents

3.4.3 Building Artificial Intelligence

3.4.4 Summary of AI Relevance for AC

3.5 Complex Systems

3.6 Key Points

4 Autonomic Computing Architectures

4.1 Autonomic Elements

4.2 Architecture of Autonomic Elements

4.2.1 IBM Reference Architecture

4.2.2 Sensors

4.2.3 Effectors

4.2.4 Autonomic Manager

4.2.5 Architectural Properties of Autonomic Managers

4.3 Autonomic Manager Reference Architecture

4.3.1 The MAPE-K Model

4.3.2 Monitoring

4.3.3 Analysis

4.3.4 Planning
### 3 Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.3.5 Execution</td>
<td>117</td>
</tr>
<tr>
<td>4.3.6 Summary</td>
<td>120</td>
</tr>
<tr>
<td>4.4 Architecture with Multiple Autonomic Elements</td>
<td>121</td>
</tr>
<tr>
<td>4.4.1 Introduction</td>
<td>121</td>
</tr>
<tr>
<td>4.4.2 Hierarchical Versus Decentralised Organisation</td>
<td>122</td>
</tr>
<tr>
<td>4.4.3 The ANS Example</td>
<td>124</td>
</tr>
<tr>
<td>4.5 Key Points</td>
<td>126</td>
</tr>
<tr>
<td>References</td>
<td>127</td>
</tr>
<tr>
<td>5 The Monitoring Function</td>
<td>129</td>
</tr>
<tr>
<td>5.1 Introduction to Monitoring</td>
<td>130</td>
</tr>
<tr>
<td>5.2 Performance Monitoring</td>
<td>132</td>
</tr>
<tr>
<td>5.3 Knowing What to Monitor and Monitoring Overheads</td>
<td>134</td>
</tr>
<tr>
<td>5.4 Profiling</td>
<td>137</td>
</tr>
<tr>
<td>5.5 Monitoring Overheads</td>
<td>138</td>
</tr>
<tr>
<td>5.6 Monitoring for Free</td>
<td>139</td>
</tr>
<tr>
<td>5.7 Building Probes</td>
<td>141</td>
</tr>
<tr>
<td>5.8 Examples of Monitoring Tools, Frameworks and Platforms</td>
<td>145</td>
</tr>
<tr>
<td>5.9 Monitoring the Monitors: Adaptive Monitoring</td>
<td>148</td>
</tr>
<tr>
<td>5.10 Key Points</td>
<td>149</td>
</tr>
<tr>
<td>References</td>
<td>150</td>
</tr>
<tr>
<td>6 The Adaptation Function</td>
<td>153</td>
</tr>
<tr>
<td>6.1 Software Adaptation</td>
<td>154</td>
</tr>
<tr>
<td>6.2 Code Adaptation</td>
<td>156</td>
</tr>
<tr>
<td>6.2.1 Upgrading Code</td>
<td>156</td>
</tr>
<tr>
<td>6.2.2 Integrating Code</td>
<td>159</td>
</tr>
<tr>
<td>6.3 Code Adaptation Techniques</td>
<td>161</td>
</tr>
<tr>
<td>6.3.1 OS-Level Adaptation</td>
<td>161</td>
</tr>
<tr>
<td>6.3.2 Program-Level Adaptation</td>
<td>163</td>
</tr>
<tr>
<td>6.3.3 Component-Level Adaptation</td>
<td>168</td>
</tr>
<tr>
<td>6.3.4 Software Services</td>
<td>170</td>
</tr>
<tr>
<td>6.4 OSGi</td>
<td>173</td>
</tr>
<tr>
<td>6.4.1 Modularity</td>
<td>173</td>
</tr>
<tr>
<td>6.4.2 Service</td>
<td>175</td>
</tr>
<tr>
<td>6.4.3 Conclusion</td>
<td>177</td>
</tr>
<tr>
<td>6.5 iPOJO</td>
<td>177</td>
</tr>
<tr>
<td>6.6 Conclusion</td>
<td>181</td>
</tr>
<tr>
<td>6.7 Key Points</td>
<td>182</td>
</tr>
<tr>
<td>References</td>
<td>182</td>
</tr>
<tr>
<td>7 The Decision Function</td>
<td>185</td>
</tr>
<tr>
<td>7.1 Introduction to Knowledge</td>
<td>186</td>
</tr>
<tr>
<td>7.1.1 Definition</td>
<td>186</td>
</tr>
<tr>
<td>7.1.2 Forms of Knowledge</td>
<td>187</td>
</tr>
<tr>
<td>7.1.3 Knowledge Representation</td>
<td>189</td>
</tr>
</tbody>
</table>
7.2 Knowledge in Autonomic Managers ............................................... 190
  7.2.1 Introduction ........................................................................ 190
  7.2.2 Rule-Based Autonomic Systems ....................................... 191
  7.2.3 Model-Based Autonomic Systems .................................... 193
  7.2.4 Goal-Based Autonomic Systems ....................................... 194
  7.2.5 Utility-Based Autonomic Systems .................................... 196
  7.2.6 Autonomic Systems That Learn ........................................ 197

7.3 Model-Driven Autonomicity ........................................................... 199
  7.3.1 Introduction ........................................................................ 199
  7.3.2 Model Representation ........................................................ 200
  7.3.3 Architectural Models ......................................................... 201

7.4 Reasoning Techniques ..................................................................... 204
  7.4.1 Programming Languages ................................................... 204
  7.4.2 Search-Based Reasoning ................................................... 205
  7.4.3 Logic-Based Reasoning ..................................................... 206
  7.4.4 Classifiers and Statistical Learning Methods ..................... 208

7.5 Bayesian Networks Example ........................................................... 208

7.6 Key Points ........................................................................................ 213

References ................................................................................................. 214

8 Evaluation Issues ..................................................................................... 217
  8.1 Evaluating Autonomic Systems ....................................................... 218
  8.2 Evaluation Elements ........................................................................ 219
  8.2.1 Quality of Service .............................................................. 219
  8.2.2 Cost .................................................................................... 219
  8.2.3 Adaptivity .......................................................................... 221
  8.2.4 Time to Adapt and Reaction Time ..................................... 221
  8.2.5 Sensitivity .......................................................................... 222
  8.2.6 Stabilisation ....................................................................... 223
  8.2.7 Failure Avoidance (Robustness) and Autonomy ................... 224
  8.2.8 Interfacing to the Outside World ....................................... 225
  8.2.9 Centralisation Versus Decentralisation ................................ 226
  8.2.10 Granularity/Flexibility ....................................................... 227
  8.3 Some Evaluation Metrics for Emergent Systems ............................ 228
  8.3.1 Price of Anarchy (PoA) ..................................................... 229
  8.3.2 Equilibrium ........................................................................ 229
  8.4 Benchmarking .................................................................................. 230
  8.5 The Autonomic Computing Benchmark: A Summary .................... 232
  8.6 Key Points ........................................................................................ 233

References ................................................................................................. 234

9 Autonomic Mediation in Cilia ................................................................. 235
  9.1 Software Integration ....................................................................... 236
  9.2 Cilia .......................................................................................... 239
  9.3 Autonomic Cilia ............................................................................... 242
  9.3.1 Overview ............................................................................ 242
  9.3.2 Cilia Touchpoints ............................................................... 243