

The Engineering of Emergence in Complex Adaptive Systems

by

Anna Elizabeth Gezina Potgieter

submitted in partial fulfilment of the requirements for the degree of

Philosophiae Doctor
(Computer Science)

in the

Faculty of Engineering, Built Environment and
Information Technology

University of Pretoria

Pretoria

2004

Acknowledgements

This thesis is dedicated to my family. To my husband Herman for his help during this quest and for his daily joy in experiencing the emergence of life, and to my dear children, Thys, Jessie and Annie, who emerged from our love.

Thanks to my supervisor, Professor Judith Bishop, for her assistance and guidance.

Thanks to my parents, for starting me off on this journey.

Thanks to the National Research Foundation for their financial assistance.

The Engineering of Emergence in a Complex Adaptive System

Student: Anna Elizabeth Gezina Potgieter
Study Leader: Professor Judith Bishop
Department: Computer Science
Degree: PhD (Computer Science)

SYNOPSIS

Agent-oriented software engineering is a new software engineering paradigm that is ideally suited to the analysis and design of complex systems. Open distributed environments place a growing demand on complex systems to be adaptive as well. Complex systems that can learn from and adapt to dynamically changing environments are called complex adaptive systems. These systems are characterized by emergent behaviour caused by interactions between system components and the environment. Agent-oriented software engineering methodologies attempt to control emergence during analysis and design by engineering the complex system in such a way that the correct emergent behaviour results during run-time. In a complex adaptive system however, emergent behaviour cannot be predicted during analysis and design, as it evolves only after implementation. By restricting emergent behaviour, as is done in most agent-oriented software engineering approaches, a complex system cannot be fully adaptive as well.

We propose the BaBe methodology that will enable a complex system to be adaptive by learning from its environment and modifying its behaviour during run-time. This methodology adds a run-time emergence model consisting of distributed Bayesian behaviour networks to the agent-oriented software engineering lifecycle. These networks are initialised by the human software engineer during analysis and design and deployed by Bayesian agencies (also complex adaptive systems). The Bayesian agents are simple, and collectively they implement distributed Bayesian behaviour networks. These networks, being specialized Bayesian networks, enable the Bayesian agents to collectively mine relationships between emergent behaviours and the interactions that caused them to emerge, in order to adapt the behaviour of the system. The agents are organized into hierarchies of agencies, where each agency activates one or more component behaviour depending on the inference in the underlying Bayesian behaviour network. These agencies assist the human software engineer to bridge the gap between the implementation and the understanding of emergent behaviour

in complex adaptive systems. Due to the simplicity of the agents and the minimal communication amongst them, they can be implemented using a commercially available component architecture. We describe a prototype implementation of the Bayesian agencies using Sun's Enterprise JavaBeans™ component architecture.

Keywords: complex adaptive systems, emergence, hyperstructures, Bayesian networks, agents, multi-agent systems, agencies, heterarchies, agent-oriented software engineering, component-based systems.

Table of Contents

Acknowledgements	ii
The Engineering of Emergence in a Complex Adaptive System	iii
Table of Contents	v
List of Figures	ix
List of Tables	xi
Chapter 1 Introduction.....	1
1.1 Complex Adaptive Systems	1
1.2 Adaptive Agent Architectures	1
1.3 Agent-oriented software engineering	2
1.4 Goal of this Thesis.....	3
1.5 Research Contribution	4
1.6 Thesis Organization.....	5
Chapter 2 Complex Adaptive Systems, Emergence and Engineering: The Basics	6
2.1 Introduction	6
2.2 Properties and Mechanisms of Complex Adaptive Systems	7
2.3 The Engineering of Emergence	8
2.3.1 What is Emergence?	8
2.3.2 Hyperstructures.....	10
2.3.3 External Observation Mechanisms	12
2.3.4 Internal Observation Mechanisms	13
2.4 Conclusion.....	15
Chapter 3 Bayesian Networks as Hyperstructures	17
3.1 Introduction	17
3.2 Basic Concepts	18
3.2.1 Propositions	18
3.2.2 Variables	18
3.2.3 Probabilities.....	19
3.2.4 Conditional Independence	20
3.3 What is a Bayesian Network?.....	21
3.4 Dynamic Bayesian Networks	24
3.5 Conditional Independence in Bayesian Networks.....	25
3.5.1 Conditional Independence involving the Parents of a Node	25

3.5.2 D-Separation	25
3.6 Bayesian Learning	27
3.6.1 Known Structure	28
3.6.2 Unknown Structure	28
3.7 Bayesian Inference	30
3.7.1 Belief Propagation	34
3.8 Bayesian Networks as Hyperstructures	37
3.9 Conclusions	38
Chapter 4 Agent Architectures	39
4.1 Overview	39
4.2 Agents: what Confusion!	40
4.3 Agencies – Order out of Chaos	41
4.4 Hierarchies and Heterarchies	43
4.5 Multi-agent Systems	44
4.6 Agent Architectures	44
4.6.1 What is an Agent Architecture?	44
4.6.2 Deliberative Agent Architectures	46
4.6.3 Reactive Agent Architectures	52
4.6.4 Adaptive Agent Architectures	62
4.6.5 A Comparison between the Different Agent Architectures	64
4.7 Conclusion	65
Chapter 5 Agent-Oriented Software Engineering	66
5.1 Introduction	66
5.2 Managing Complexity	66
5.3 Component-based Software Engineering – A Brief Overview	67
5.4 A Comparison between Objects, Components and Complex Agents	69
5.4.1 Overview	69
5.4.2 Similarity between Objects, Components and Complex Agents	69
5.4.3 Difference between Objects, Components and Complex Agents	69
5.5 Component-Based Agent Architectures	70
5.6 Agent-Oriented Software Engineering Methodologies	73
5.6.1 Overview	73
5.6.2 The Gaia Methodology	73

5.6.3 Coordination-Oriented Methodology	74
5.7 Conclusion	76
Chapter 6 BaBe: An Adaptive Agent Architecture	77
6.1 Overview	77
6.2 Bayesian Behaviour Networks	78
6.3 Competence Sets	83
6.4 Pearl's Belief Propagation Algorithm	84
6.5 The BaBe Components	85
6.5.1 The Node Component	87
6.5.2 The Link Component	89
6.5.3 The Belief Propagation Agents	91
6.6 Bayesian Agencies	96
6.7 Bayesian Agencies in Web Personalization	101
6.8 Applicability, Use and Value of the BaBe Agent Architecture	103
6.8.1 A General Application of the BaBe Agent Architecture	103
6.8.2 The Limitations of the BaBe Architecture	105
6.8.3 The Benefits of the BaBe System	106
6.8.4 How and where can the general BaBe application be used?	107
6.9 The BaBe Agent Architecture: A Complex Adaptive System	108
6.10 Conclusion	110
Chapter 7 The BaBe Methodology	112
7.1 Overview	112
7.2 The Gaia Models	113
7.3 Coordination Models	115
7.4 The BaBe Models	116
7.4.1 Overview	116
7.4.2 The Environment Model	120
7.4.3 The Roles Model	121
7.4.4 Interactions Model	124
7.4.5 Agency Model	124
7.4.6 Internal Behaviour Model	125
7.4.7 External Behaviour Model	125
7.4.8 The Emergence Model	125

7.5 The BaBe Methodology	126
7.5.1 Graphical Diagrams	126
7.5.2 The Analysis Process	126
7.5.3 The Design Process	127
7.5.4 The Execution Phase	127
7.5.5 The Limitations of the BaBe Execution Phase Application.....	128
7.5.6 The Benefits of the BaBe Execution Phase Application.....	128
7.5.7 How and where can the BaBe Execution Phase Application be used?	129
7.6 Conclusion	129
Chapter 8 BaBe: The Prototype	131
8.1 Overview	131
8.2 Configuring the Bayesian Behaviour Network	132
8.3 Querying the Bayesian Behaviour Network	136
8.4 Learning the Bayesian Behaviour Network	141
8.5 The Limitations of the BaBe Prototype	144
8.6 Conclusion	144
Chapter 9 Conclusions and Future Research	145
9.1 Future Research: The BaBe Architecture	145
9.2 The Complex Adaptive Enterprise: Sustaining the Competitive Advantage using Bayesian Agencies.....	147
9.2.1 Introduction.....	147
9.2.2 The Chain of Sustainability.....	148
9.2.3 The Relationship between Emergence and Knowledge in the Complex Adaptive Enterprise	151
9.2.4 Future Research: BaBe and the Engineering of Emergence in the Complex Adaptive Enterprise	151
9.3 Conclusion	152
Bibliography	154
APPENDIX A SOFTWARE LISTINGS.....	160

List of Figures

Figure 1: Minsky's A- and B-Brain (Minsky, 1988).....	14
Figure 2: A Bayesian Network.....	23
Figure 3: Naïve Bayes Model.....	24
Figure 4: A Dynamic Bayesian Network	25
Figure 5: Conditional Independence via Blocking Nodes (Nilsson, 1998).....	26
Figure 6: Bayesian Inference Example.....	31
Figure 7: Separation of Evidence by a Link	35
Figure 8: Separation of Evidence by a Node.....	36
Figure 9: The Belief-Desire-Joint-Intention Agent Architecture	51
Figure 10: The Subsumption Architecture	53
Figure 11: Communication between Competence Levels 0 and 1	55
Figure 12: Object Process Diagram for a Simple Toy Robot.....	57
Figure 13: A Behaviour Network	59
Figure 14: The Component-Based Design Pattern (Bachman et al., 2000).....	68
Figure 15: Component-Based Agent Architecture (Griss & Pour, 2001)	70
Figure 16: Internal Agent Components (Skarmas & Clark, 1999)	72
Figure 17: A Bayesian Behaviour Network	79
Figure 18: Belief Propagation in a Bayesian Behaviour Network	82
Figure 19: Belief Propagation	85
Figure 20: BaBe Agent Architecture Components.....	86
Figure 21: Node Component Diagram for Node X	88
Figure 22: Link Component Diagram for Link XY_j	90
Figure 23: The Belief Propagation Agent	92
Figure 24: Bayesian Agencies State Diagram	96
Figure 25: Heterarchy of Bayesian Agencies	98
Figure 26: Example BaBe Components	99
Figure 27: PersonaliseAgency Output.....	100
Figure 28: Bayesian Agencies in Web Personalization.....	102
Figure 29: BaBe – A General Application	104
Figure 30: Relationships between the Gaia Models (Wooldridge et al., 2000).....	114
Figure 31: Coordination Models (Zambonelli et al., 2000).....	116

Figure 32: The BaBe Methodology	119
Figure 33: Example Environment Model.....	120
Figure 34: The BaBe Roles Model	121
Figure 35: BaBe Role Schema	122
Figure 36: Liveness Properties OPD.....	123
Figure 37: Safety Properties OPD.....	123
Figure 38: Example Role	124
Figure 39: General Application for the BaBe Methodology Execution Phase	128
Figure 40: JMS Link Definitions	132
Figure 41: Bayes Components	133
Figure 42: J2EE Server Start-up Trace	134
Figure 43: Original Beliefs – No Evidence.....	135
Figure 44: Setting Evidence for a Query	136
Figure 45: Belief Propagation Agency Output	139
Figure 46: Beliefs of Nodes after Belief Propagation (Evidence and No Learning)	140
Figure 47: Setting Evidence that must be learnt	141
Figure 48: Beliefs of Nodes after Belief Propagation (Evidence and Learning)	142
Figure 49: New Beliefs after Learning	143
Figure 50: Reflective BaBe Architecture.....	146
Figure 51: The Chain of Sustainability (April, 2002).....	150

List of Tables

Table 1: A Training Set.....	27
Table 2: Comparison between Agent Architectures.....	64
Table 3: Conditional Probability Matrix for Node B	80

*It is heart-of-watermelon red
Mellow like an amber slice of the moon
As it emerges from high rock and low cloud
Suspended near the blueless sky
A spectre between nothing and nothing
Without a single ray of light
As if simple to say
Don't you know the world is remarkable*

- Keorapetse Kgositsile