Contents

Foreword V

List of Figures XIV

List of Process Models XV

1. Introduction 1

2. Agents, Multiagent Systems and Software Engineering 9
2.1 Intelligent Agents 9
2.1.1 What's an Agent, anyway? 9
2.1.2 Roles 12
2.1.3 Architectures 13
2.1.4 Agents, Roles and Architectures 14
2.2 Systems of Agents 15
2.2.1 Interaction 16
2.2.2 The Social Dimension 17
2.3 Related Fields in Computer Science 19
2.4 Agent-Oriented Software Engineering 21
2.4.1 Aspects of Programming Paradigms 21
2.4.2 A Historic Perspective 28
2.4.3 The Bottom Line 30
2.4.4 Where Next? 32
2.5 Summary 33

3. Basic Concepts in Software Engineering 35
3.1 Cognitive Aspects of Software Engineering 35
3.1.1 Basic Human Information Processing 36
3.1.2 Software Engineering as a General Design Task 38
3.1.3 Designs and Models 40
3.1.4 A General Model of Engineering 41
3.1.5 The Basic Engineering Cycle 43
3.1.6 Basic Skills in Software Engineering 46
3.2 Requirements for Software Engineering Support 50

3.3 A General Model of Software Engineering 51
3.4 Software Engineering Product Models 53
3.4.1 A Generic Product Model 54
3.4.2 Software Blueprints: The Unified Modeling Language . 55
3.5 Software Engineering Process Models 57
3.5.1 Classical Process Models 58
3.5.2 Novel Trends in Software Engineering 67
3.5.3 Development Methods for Multiagent Systems 78
3.5.4 Discussion 91
3.6 Quality Management and Systematic Learning 91
3.6.1 The Quality Improvement Paradigm 92
3.6.2 Experience Factory 92
3.7 Summary 95

4. The Conceptual Framework of MASSIVE 97
4.1 The Foundations of MASSIVE 97
4.2 Knowbbles 99
4.3 Views 101
4.3.1 What and Why? 102
4.3.2 View-Oriented Analysis 106
4.3.3 A View System for Multiagent Systems 108
4.4 Iterative View Engineering 114
4.5 Putting It All Together 117
4.6 Summary 120

5. MASSIVE **Views** 121
5.1 A Brief Introduction to Train Coupling- and Sharing (TCS) . 122
5.2 Environment View 125
5.2.1 Developers Perspective 125
5.2.2 Systems Perspective 129
5.3 Task View 130
5.3.1 Use Case Analysis 130
5.3.2 Functional Requirements 131
5.3.3 Nonfunctional Requirements 138
5.4 Role View 138
5.4.1 Role Definition 139
5.4.2 Role Assignment 144
5.5 Interaction View 144
5.5.1 Intent Layer 145
5.5.2 Protocol Layer 148
5.5.3 Transport Layer 165
5.6 Society View 167
5.6.1 Characterization of Social Systems 167
5.6.2 Designing Social Systems 169
5.7 Architecture View 174

5.7.1 System Architecture . . . 175
5.7.2 The Architectural Feature Space . . . 177
5.7.3 Agent Architecture . . . 184
5.8 System View . . . 190
5.8.1 User Interface Design . . . 190
5.8.2 Exception Handling . . . 194
5.8.3 Performance Engineering . . . 196
5.8.4 Deployment . . . 201
5.9 Summary . . . 203

6. Further Case Studies . . . 205
6.1 The TEAMWORK LIBRARY . . . 205
6.1.1 Environment View . . . 205
6.1.2 Task View . . . 206
6.1.3 Role View . . . 208
6.1.4 Interaction View . . . 210
6.1.5 Society View . . . 214
6.1.6 Architecture View . . . 214
6.1.7 System View . . . 217
6.2 Personal Travel Assistant: Intermodal Route Planning . . . 219
6.2.1 Environment View . . . 220
6.2.2 Task View . . . 222
6.2.3 Role View . . . 226
6.2.4 Interaction View . . . 228
6.2.5 Society View . . . 231
6.2.6 Architecture View . . . 231
6.2.7 System View . . . 234
6.3 Summary . . . 241

7. Conclusion . . . 243

A. Toolkits for Agent-Based Applications . . . 247
A.1 SIF . . . 247
A.2 ZEUS . . . 251
A.3 Swarm . . . 253
A.4 Summary . . . 254

B. Basic Problem Solving Capabilities of TCS Agents . . . 255
B.1 Planing Algorithm for a Single Task . . . 255
B.2 Plan Integration Operator . . . 256
B.3 Decision Functions . . . 259
B.4 Plan Execution Simulation . . . 259

C. Protoz Specification of the Contract-Net Protocol . . . 261

Bibliography 265

Glossary 281

Index 283