Contents

1 Introduction to Software Architecture 1
1.1 A Brief History of Software Development 2
1.1.1 The Evolution of Programming Language—Abstract Level 2
1.1.2 The Evolution of Software Development—Concerns 4
1.1.3 The Origin and Growth of Software Architecture 6
1.2 Introduction to Software Architecture 9
1.2.1 Basic Terminologies 9
1.2.2 Understanding IEEE 1471—2000 12
1.2.3 Views Used in Software Architecture 15
1.2.4 Why We Need Software Architecture 25
1.2.5 Where Is Software Architecture in Software Life Cycle 29
1.3 Summary 31
References 32

2 Architectural Styles and Patterns 34
2.1 Fundamentals of Architectural Styles and Patterns 34
2.2 Pipes Filters 38
2.2.1 Style Description 38
2.2.2 Study Case 39
2.3 Object-oriented 42
2.3.1 Style Description 42
2.3.2 Study Case 43
2.4 Event-driven 51
2.4.1 Style Description 51
2.4.2 Study Case 55
2.5 Hierarchical Layer 62
2.5.1 Style Description 62
2.5.2 Study Case 64
2.6 Data Sharing 70
2.6.1 Style Description 70
2.6.2 Study Case 72

Bibliografische Informationen
http://d-nb.info/988160307

digitalisiert durch DEUTSCHE NATIONAL BIBLIOTHEK

2.7 Virtual Machine ····· 76
2.7.1 Style Description ····· 76
2.7.2 Study Case ····· 77
2.8 Feedback Loop ····· 81
2.8.1 Style Description ····· 81
2.8.2 Study Case ····· 82
2.9 Comparison among Styles ····· 83
2.10 Integration of Heterogeneous Styles ····· 85
2.11 Summary ····· 86
References ····· 87

3 Application and Analysis of Architectural Styles ····· 89
3.1 Introduction to SMCSP ····· 89
3.1.1 Program Background ····· 89
3.1.2 Technical Routes ····· 91
3.1.3 Function Design ····· 93
3.2 System Realization ····· 97
3.2.1 The Pattern Choice ····· 97
3.2.2 Interaction Mechanism ····· 101
3.2.3 Realization of Mobile Collaboration ····· 104
3.2.4 Knowledge-based Design ····· 111
3.3 Summary ····· 115
References ····· 116

4 Software Architecture Description ····· 117
4.1 Formal Description of Software Architecture ····· 117
4.1.1 Problems in Informal Description ····· 117
4.1.2 Why Are Formal Methods Necessary ····· 120
4.2 Architectural Description Language ····· 123
4.2.1 Introduction to ADL ····· 123
4.2.2 Comparing among Typical ADLs ····· 127
4.2.3 Describing Architectural Behaviors ····· 133
4.3 Study Case: WRIGHT System ····· 135
4.3.1 Description of Component and Connector ····· 136
4.3.2 Description of Configuration ····· 141
4.3.3 Description of Style ····· 143
4.3.4 CSP—Semantic Basis of Formal Behavior Description ····· 146
4.4 FEAL: An Infrastructure to Construct ADLs ····· 160
4.4.1 Design Purpose ····· 160
4.4.2 FEC ····· 161
4.4.3 FEAL Structure ····· 163
4.4.4 FEAL Mapper ····· 164
4.4.5 Examples of FEAL Application ····· 164
4.5 Summary ····· 166

References 167

5 Design Strategies in Architecture Level 169
5.1 From Reuse to Architecture Design 170
5.2 Architectural Design Space and Rules 171
5.3 SADPBA 172
5.3.1 Overview 173
5.3.2 Split Design Process with Design Space 173
5.3.3 Trace Mechanism in SADPBA 176
5.3.4 Life Cycle Model of Software Architecture 177
5.3.5 SADPBA in Practice 178
5.4 Study Case: MEECS 180
5.4.1 Introduction to MEECS 180
5.4.2 Applying SADPBA in MEECS 182
5.5 Summary 189
References 190

6 Software Architecture IDE 191
6.1 What Can Software Architecture IDE Do 191
6.1.1 A Comparison with Formalized Description Approach 191
6.1.2 Important Roles of Architecture IDE 192
6.2 Prototype 195
6.2.1 User Interface Layer 196
6.2.2 Model Layer 197
6.2.3 Foundational Layer 199
6.2.4 IDE Design Tactics 200
6.3 ArchStudio 4 System 201
6.3.1 Introduction 201
6.3.2 Installing ArchStudio 4 204
6.3.3 ArchStudio 4 Overview 206
6.3.4 Using ArchStudio 4 214
6.4 Summary 218
References 220

7 Evaluating Software Architecture 221
7.1 What Is Software Architecture Evaluation 222
7.1.1 Quality Attribute 222
7.1.2 Why Is Evaluation Necessary 224
7.1.3 Scenario-based Evaluation Methods 225
7.2 SAAM 228
7.2.1 General Steps of SAAM 228
7.2.2 Scenario Development 230
7.2.3 Architecture Description 230
7.2.4 Scenario Classification and Prioritization 231

7.2.5 Individual Evaluation of Indirect Scenarios ····· 232
7.2.6 Assessment of Scenario Interaction ····· 233
7.2.7 Creation of Overall Evaluation ····· 233
7.3 ATAM ····· 234
7.3.1 Initial ATAM ····· 235
7.3.2 ATAM Improvement ····· 237
7.3.3 General Process of ATAM ····· 238
7.3.4 Presentation ····· 241
7.3.5 Investigation and Analysis ····· 242
7.3.6 Testing ····· 244
7.3.7 Present the Results ····· 245
7.4 Comparison among Evaluation Methods ····· 246
7.4.1 Comparison Framework ····· 246
7.4.2 Overview and Comparison of Evaluation Methods ····· 250
7.5 Summary ····· 269
References ····· 270

8 Flexible Software Architecture ····· 274
8.1 What Is Flexibility for ····· 274
8.2 Dynamic Software Architecture ····· 276
8.2.1 π-ADL: A Behavior Perspective ····· 278
8.2.2 MARMOL: A Reflection Perspective ····· 284
8.2.3 LIME: A Coordination Perspective ····· 291
8.3 Flexibility: Beyond the Dynamism ····· 299
8.3.1 Concept of Flexible Software Architecture ····· 299
8.3.2 Trade-off of Flexibility ····· 300
8.4 Study Cases ····· 303
8.4.1 Rainbow ····· 303
8.4.2 MADAM ····· 305
8.5 Summary ····· 307
References ····· 308

9 A Vision on Software Architecture ····· 313
9.1 Software Architecture in Modern Software Industry ····· 313
9.1.1 Categorizing Software ····· 313
9.1.2 Software Product Line ····· 318
9.2 Software Architecture Used in Other Fields ····· 325
9.2.1 The Outline of Software Architecture Application Practice ····· 325
9.2.2 The Development Trends of Domain-Specific Software ····· 325
9.3 Software Architecture's Future Research ····· 330
9.4 Summary ····· 331
References ····· 332

Index ····· 333