READ-ME

Monday, 3 March, 2025 7:51 Manhã

Table of Contents

Preface: Invalidating Axioms. xiii

1. Introduction. 1
   Defining Software Architecture 3
   Expectations of an Architect 8
   Make Architecture Decisions 9
   Continually Analyze the Architecture 9

Defining Software Architecture 3
Expectations of an Architect 8
Make Architecture Decisions 9
Continually Analyze the Architecture 9
Keep Current with Latest Trends 10
Ensure Compliance with Decisions 10
Diverse Exposure and Experience 11
Have Business Domain Knowledge 11
Possess Interpersonal Skills 12
Understand and Navigate Politics 12
Intersection of Architecture and… 13
Engineering Practices 14
Operations/DevOps 17
Process 18
Data 19
Laws of Software Architecture 19

Part I. Foundations

2. Architectural Thinking. . . . 23
Architecture Versus Design 23
Technical Breadth 25

Analyzing Trade-Offs 30
Understanding Business Drivers 34
Balancing Architecture and Hands-On Coding 34

3. Modularity. . . . 37
Definition 38
Measuring Modularity 40
Cohesion 40
Coupling 44
Abstractness, Instability, and Distance from the Main Sequence 44
Distance from the Main Sequence 46
Connascence 48
Unifying Coupling and Connascence Metrics 52
From Modules to Components 53

4. Architecture Characteristics Defined. . . . 55
Architectural Characteristics (Partially) Listed 58
Operational Architecture Characteristics 58
Structural Architecture Characteristics 59
Cross-Cutting Architecture Characteristics 59
Trade-Offs and Least Worst Architecture 63

5. Identifying Architectural Characteristics. . . . 65
Extracting Architecture Characteristics from Domain Concerns 65
Extracting Architecture Characteristics from Requirements 67
Case Study: Silicon Sandwiches 69
Explicit Characteristics 70
Implicit Characteristics 73

6. Measuring and Governing Architecture Characteristics. . . . 77
Measuring Architecture Characteristics 77
Operational Measures 78
Structural Measures 79
Process Measures 81
Governance and Fitness Functions 82
Governing Architecture Characteristics 82
Fitness Functions 83

7. Scope of Architecture Characteristics. . . . 91
Coupling and Connascence 92

Architectural Quanta and Granularity 92
Case Study: Going, Going, Gone 95

8. Component-Based Thinking. . . . 99

Architectural Quanta and Granularity 92
Case Study: Going, Going, Gone 95

8. Component-Based Thinking **99**
Component Scope 99
Architect Role 101
Architecture Partitioning 102
Case Study: Silicon Sandwiches: Partitioning 105
Developer Role 108
Component Identification Flow 108
Identifying Initial Components 108
Assign Requirements to Components 109
Analyze Roles and Responsibilities 109
Analyze Architecture Characteristics 109
Restructure Components 109
Component Granularity 110
Component Design 110
Discovering Components 110
Case Study: Going, Going, Gone: Discovering Components 112
Architecture Quantum Redux: Choosing Between Monolithic Versus Distributed Architectures 115

Part II. Architecture Styles

9. Foundations **119**
Fundamental Patterns 119
Big Ball of Mud 120
Unitary Architecture 121
Client/Server 121
Monolithic Versus Distributed Architectures 123
Fallacy #1: The Network Is Reliable 124
Fallacy #2: Latency Is Zero 125
Fallacy #3: Bandwidth Is Infinite 126
Fallacy #4: The Network Is Secure 127
Fallacy #5: The Topology Never Changes 128
Fallacy #6: There Is Only One Administrator 129
Fallacy #7: Transport Cost Is Zero 130
Fallacy #8: The Network Is Homogeneous 131
Other Distributed Considerations 131

10. Layered Architecture Style **133**
Topology 133
Layers of Isolation 135
Adding Layers 136
Other Considerations 138
Why Use This Architecture Style 139
Architecture Characteristics Ratings 139

11. Pipeline Architecture Style **143**
Topology 143
Pipes 144
Filters 144
Example 145
Architecture Characteristics Ratings 146

12. Microkernel Architecture Style **149**
Topology 149
Core System 150
Plug-In Components 153
Registry 157
Contracts 158
Examples and Use Cases 158
Architecture Characteristics Ratings 160

13. Service-Based Architecture Style **163**
Topology 163
Topology Variants 165
Service Design and Granularity 167
Database Partitioning 169
Example Architecture 172

Topology Variants 165
Service Design and Granularity 167
Database Partitioning 169
Example Architecture 172
Architecture Characteristics Ratings 174
When to Use This Architecture Style 177

14. Event-Driven Architecture Style **179**
Topology 180
Broker Topology 180
Mediator Topology 185
Asynchronous Capabilities 196
Error Handling 197
Preventing Data Loss 201

Broadcast Capabilities 203
Request-Reply 204
Choosing Between Request-Based and Event-Based 206
Hybrid Event-Driven Architectures 207
Architecture Characteristics Ratings 207

15. Space-Based Architecture Style **211**
General Topology 212
Processing Unit 213
Virtualized Middleware 214
Data Pumps 219
Data Writers 221
Data Readers 222
Data Collisions 224
Cloud Versus On-Premises Implementations 226
Replicated Versus Distributed Caching 227
Near-Cache Considerations 230
Implementation Examples 231
Concert Ticketing System 231
Online Auction System 232
Architecture Characteristics Ratings 233

16. Orchestration-Driven Service-Oriented Architecture **235**
History and Philosophy 235
Topology 236
Taxonomy 236
Business Services 237
Enterprise Services 237
Application Services 237
Infrastructure Services 237
Orchestration Engine 238
Message Flow 238
Reuse…and Coupling 239
Architecture Characteristics Ratings 241

17. Microservices Architecture **245**
History 245
Topology 246
Distributed 247
Bounded Context 247

Granularity 248
Data Isolation 249

Granularity 248
Data Isolation 249
API Layer 249
Operational Reuse 250
Frontends 253
Communication 254
Choreography and Orchestration 256
Transactions and Sagas 260
Architecture Characteristics Ratings 263
Additional References 265

18. Choosing the Appropriate Architecture Style. 267
Shifting "Fashion" in Architecture 267
Decision Criteria 269
Monolith Case Study: Silicon Sandwiches 271
Modular Monolith 271
Microkernel 272
Distributed Case Study: Going, Going, Gone 274

Part III. Techniques and Soft Skills

19. Architecture Decisions. 281
Architecture Decision Anti-Patterns 281
Covering Your Assets Anti-Pattern 282
Groundhog Day Anti-Pattern 282
Email-Driven Architecture Anti-Pattern 283
Architecturally Significant 284
Architecture Decision Records 285
Basic Structure 285
Storing ADRs 291
ADRs as Documentation 293
Using ADRs for Standards 293
Example 294

20. Analyzing Architecture Risk. 297
Risk Matrix 297
Risk Assessments 298
Risk Storming 302
Identification 303

Consensus 304
Agile Story Risk Analysis 308
Risk Storming Examples 308
Availability 310
Elasticity 312
Security 313

21. Diagramming and Presenting Architecture. 315
Diagramming 316
Tools 316
Diagramming Standards: UML, C4, and ArchiMate 318
Diagram Guidelines 319
Presenting 321
Manipulating Time 321
Incremental Builds 322
Infodecks Versus Presentations 324
Slides Are Half of the Story 324
Invisibility 324

22. Making Teams Effective. 325
Team Boundaries 325
Architect Personalities 326
Control Freak 327
Armchair Architect 328
Effective Architect 330
How Much Control? 331
Team Warning Signs 335
Leveraging Checklists 338
Developer Code Completion Checklist 340

How Much Control? 331
Team Warning Signs 335
Leveraging Checklists 338
Developer Code Completion Checklist 340
Unit and Functional Testing Checklist 341
Software Release Checklist 342
Providing Guidance 343
Summary 346

23. Negotiation and Leadership Skills. 347
Negotiation and Facilitation 347
Negotiating with Business Stakeholders 348
Negotiating with Other Architects 350
Negotiating with Developers 351
The Software Architect as a Leader 353

The 4 C's of Architecture 353
Be Pragmatic, Yet Visionary 355
Leading Teams by Example 357
Integrating with the Development Team 360
Summary 363

24. Developing a Career Path. 365
The 20-Minute Rule 365
Developing a Personal Radar 367
The ThoughtWorks Technology Radar 367
Open Source Visualization Bits 371
Using Social Media 371
Parting Words of Advice 372

Appendix. Self-Assessment Questions. 373

Index. 383