

# Contents

<b>1</b>	<b>Understanding Software Architecture</b>	1
1.1	What is Software Architecture?	1
1.2	Definitions of Software Architecture	2
1.2.1	Architecture Defines Structure	3
1.2.2	Architecture Specifies Component Communication	4
1.3	Architecture Addresses Nonfunctional Requirements	5
1.3.1	Architecture Is an Abstraction	6
1.3.2	Architecture Views	7
1.4	What Does a Software Architect Do?	8
1.5	Architectures and Technologies	9
1.6	Architect Title Soup	11
1.7	Summary	12
1.8	Further Reading	13
1.8.1	General Architecture	13
1.8.2	Architecture Requirements	13
1.8.3	Architecture Patterns	14
1.8.4	Technology Comparisons	14
1.8.5	Enterprise Architecture	15
<b>2</b>	<b>Introducing the Case Study</b>	17
2.1	Overview	17
2.2	The ICDE System	17
2.3	Project Context	19
2.4	Business Goals	21
2.5	Constraints	22
2.6	Summary	22
<b>3</b>	<b>Software Quality Attributes</b>	23
3.1	Quality Attributes	23
3.2	Performance	24
3.2.1	Throughput	24
3.2.2	Response Time	25

3.2.3	Deadlines .....	25
3.2.4	Performance for the ICDE System .....	26
3.3	Scalability .....	27
3.3.1	Request Load .....	27
3.3.2	Simultaneous Connections .....	29
3.3.3	Data Size .....	29
3.3.4	Deployment .....	30
3.3.5	Some Thoughts on Scalability .....	30
3.3.6	Scalability for the ICDE Application .....	30
3.4	Modifiability .....	30
3.4.1	Modifiability for the ICDE Application .....	33
3.5	Security .....	33
3.5.1	Security for the ICDE Application .....	34
3.6	Availability .....	34
3.6.1	Availability for the ICDE Application .....	35
3.7	Integration .....	35
3.7.1	Integration for the ICDE Application .....	36
3.8	Other Quality Attributes .....	36
3.9	Design Trade-Offs .....	37
3.10	Summary .....	37
3.11	Further Reading .....	38

#### **4 An Introduction to Middleware Architectures and Technologies .....**

		39
4.1	Introduction .....	39
4.2	Middleware Technology Classification .....	40
4.3	Distributed Objects .....	41
4.4	Message-Oriented Middleware .....	43
4.4.1	MOM Basics .....	44
4.4.2	Exploiting MOM Advanced Features .....	45
4.4.3	Publish-Subscribe .....	50
4.5	Application Servers .....	54
4.5.1	Enterprise JavaBeans .....	55
4.5.2	EJB Component Model .....	56
4.5.3	Stateless Session Bean Programming Example .....	57
4.5.4	Message-Driven Bean Programming Example .....	58
4.5.5	Responsibilities of the EJB Container .....	59
4.5.6	Some Thoughts .....	60
4.6	Summary .....	61
4.7	Further Reading .....	62
4.7.1	CORBA .....	62
4.7.2	Message-Oriented Middleware .....	62
4.7.3	Application Servers .....	63

<b>5</b>	<b>Service-Oriented Architectures and Technologies</b>	65
5.1	Background	65
5.2	Service-Oriented Systems	66
5.2.1	Boundaries Are Explicit	68
5.2.2	Services Are Autonomous	69
5.2.3	Share Schemas and Contracts, Not Implementations	69
5.2.4	Service Compatibility Is Based on Policy	70
5.3	Web Services	71
5.4	SOAP and Messaging	73
5.5	UDDI, WSDL, and Metadata	74
5.6	Security, Transactions, and Reliability	77
5.7	RESTful Web Services	78
5.8	Conclusion and Further Reading	79
<b>6</b>	<b>Advanced Middleware Technologies</b>	81
6.1	Introduction	81
6.2	Message Brokers	81
6.3	Business Process Orchestration	87
6.4	Integration Architecture Issues	91
6.5	What Is an Enterprise Service Bus	95
6.6	Further Reading	95
<b>7</b>	<b>A Software Architecture Process</b>	97
7.1	Process Outline	97
7.1.1	Determine Architectural Requirements	98
7.1.2	Identifying Architecture Requirements	98
7.1.3	Prioritizing Architecture Requirements	99
7.2	Architecture Design	101
7.2.1	Choosing the Architecture Framework	102
7.2.2	Allocate Components	108
7.3	Validation	110
7.3.1	Using Scenarios	111
7.3.2	Prototyping	113
7.4	Summary and Further Reading	114
<b>8</b>	<b>Documenting a Software Architecture</b>	117
8.1	Introduction	117
8.2	What to Document	118
8.3	UML 2.0	119
8.4	Architecture Views	120
8.5	More on Component Diagrams	123
8.6	Architecture Documentation Template	126
8.7	Summary and Further Reading	127

<b>9 Case Study Design</b>	129
9.1 Overview	129
9.2 ICDE Technical Issues	129
9.2.1 Large Data	129
9.2.2 Notification	131
9.2.3 Data Abstraction	131
9.2.4 Platform and Distribution Issues	131
9.2.5 API Issues	132
9.2.6 Discussion	133
9.3 ICDE Architecture Requirements	133
9.3.1 Overview of Key Objectives	133
9.3.2 Architecture Use Cases	134
9.3.3 Stakeholder Architecture Requirements	134
9.3.4 Constraints	136
9.3.5 Nonfunctional Requirements	136
9.3.6 Risks	137
9.4 ICDE Solution	137
9.4.1 Architecture Patterns	137
9.4.2 Architecture Overview	138
9.4.3 Structural Views	139
9.4.4 Behavioral Views	142
9.4.5 Implementation Issues	145
9.5 Architecture Analysis	145
9.5.1 Scenario Analysis	145
9.5.2 Risks	146
9.6 Summary	146
 <b>10 Middleware Case Study: MeDICI</b>	 147
10.1 MeDICI Background	147
10.2 MeDICI Hello World	148
10.3 Implementing Modules	151
10.3.1 MifProcessor	151
10.3.2 MifObjectProcessor	151
10.3.3 MifMessageProcessor	152
10.3.4 Module Properties	152
10.4 Endpoints and Transports	153
10.4.1 Connectors	153
10.4.2 Supported Transports	154
10.5 MeDICI Example	157
10.5.1 Initialize Pipeline	158
10.5.2 Chat Component	159
10.5.3 Implementation code	161
10.6 Component Builder	161
10.7 Summary	163
10.8 Further Reading	163

<b>11</b>	<b>Looking Forward</b>	165
11.1	Introduction	165
11.2	The Challenges of Complexity	165
11.2.1	Business Process Complexity	166
11.3	Agility	167
11.4	Reduced Costs	168
11.5	What Next	169
<b>12</b>	<b>The Semantic Web</b>	171
12.1	ICDE and the Semantic Web	171
12.2	Automated, Distributed Integration and Collaboration	172
12.3	The Semantic Web	173
12.4	Creating and Using Metadata for the Semantic Web	174
12.5	Putting Semantics in the Web	176
12.6	Semantics for ICDE	178
12.7	Semantic Web Services	180
12.8	Continued Optimism	181
12.9	Further Reading	182
<b>13</b>	<b>Aspect Oriented Architectures</b>	185
13.1	Aspects for ICDE Development	185
13.2	Introduction to Aspect-Oriented Programming	186
13.2.1	Crosscutting Concerns	186
13.2.2	Managing Concerns with Aspects	187
13.2.3	AOP Syntax and Programming Model	188
13.2.4	Weaving	189
13.3	Example of a Cache Aspect	190
13.4	Aspect-Oriented Architectures	191
13.5	Architectural Aspects and Middleware	192
13.6	State-of-the-Art	193
13.6.1	Aspect Oriented Modeling in UML	193
13.6.2	AOP Tools	193
13.6.3	Annotations and AOP	194
13.7	Performance Monitoring of ICDE with AspectWerkz	195
13.8	Conclusions	197
13.9	Further Reading	198
<b>14</b>	<b>Model-Driven Architecture</b>	201
14.1	Model-Driven Development for ICDE	201
14.2	What is MDA?	203
14.3	Why MDA?	205
14.3.1	Portability	205
14.3.2	Interoperability	206
14.3.3	Reusability	207

14.4	State-of-Art Practices and Tools .....	208
14.4.1	AndroMDA .....	208
14.4.2	ArcStyler .....	209
14.4.3	Eclipse Modeling Framework .....	209
14.5	MDA and Software Architecture .....	210
14.5.1	MDA and Nonfunctional Requirements .....	211
14.5.2	Model Transformation and Software Architecture .....	211
14.5.3	SOA and MDA .....	212
14.5.4	Analytical Models are Models Too .....	212
14.6	MDA for ICDE Capacity Planning .....	214
14.7	Summary and Further Reading .....	216
<b>15</b>	<b>Software Product Lines .....</b>	<b>219</b>
15.1	Product Lines for ICDE .....	219
15.2	Software Product Lines .....	220
15.2.1	Benefiting from SPL Development .....	222
15.2.2	Product Lines for ICDE .....	223
15.3	Product Line Architecture .....	223
15.3.1	Find and Understand Software .....	224
15.3.2	Bring Software into the Development Context .....	225
15.3.3	Invoke Software .....	225
15.3.4	Software Configuration Management for Reuse .....	225
15.4	Variation Mechanisms .....	227
15.4.1	Architecture-Level Variation Points .....	227
15.4.2	Design-Level Variation .....	227
15.4.3	File-Level Variation .....	228
15.4.4	Variation by Software Configuration Management .....	228
15.4.5	Product Line Architecture for ICDE .....	228
15.5	Adopting Software Product Line Development .....	229
15.5.1	Product Line Adoption Practice Areas .....	231
15.5.2	Product Line Adoption for ICDE .....	231
15.6	Ongoing Software Product Line Development .....	232
15.6.1	Change Control .....	232
15.6.2	Architectural Evolution for SPL Development .....	233
15.6.3	Product Line Development Practice Areas .....	234
15.6.4	Product Lines with ICDE .....	234
15.7	Conclusions .....	235
15.8	Further Reading .....	236
<b>Index</b>	<b>.....</b>	<b>239</b>



<http://www.springer.com/978-3-642-19175-6>

Essential Software Architecture

Gorton, I.

2011, XVI, 242 p., Hardcover

ISBN: 978-3-642-19175-6