

<b>Grußwort</b> .....	<b>15</b>
<b>Vorwort</b> .....	<b>17</b>
<b>Einleitung</b> .....	<b>25</b>

---

## Teil I Strategisches Design

<b>1 Fachdomänen analysieren</b> .....	<b>29</b>
Was ist eine Fachdomäne? .....	29
Was ist eine Subdomain? .....	30
Arten von Subdomains .....	30
Subdomains vergleichen .....	33
Grenzen von Subdomains ermitteln .....	37
Beispiele für eine Analyse der Fachdomäne .....	40
Gigmaster .....	41
BusVNext .....	42
Wer sind die Domänenexperten? .....	44
Zusammenfassung .....	45
Übungen .....	45
<b>2 Domänenwissen ermitteln</b> .....	<b>47</b>
Fachliche Probleme .....	47
Erlernen von Fachwissen .....	48
Kommunikation .....	48
Was ist eine Ubiquitous Language? .....	50
Sprache der Domänenexperten .....	51
Szenarien .....	51
Konsistenz .....	52

Modell der Fachdomäne . . . . .	53
Was ist ein Modell? . . . . .	53
Effektives Modellieren . . . . .	54
Die Fachdomäne modellieren . . . . .	54
Kontinuierliche Arbeit . . . . .	55
Werkzeuge . . . . .	55
Herausforderungen . . . . .	56
Zusammenfassung . . . . .	57
Übungen . . . . .	58
<b>3 Die Komplexität einer Domain im Griff behalten . . . . .</b>	<b>61</b>
Inkonsistente Modelle. . . . .	61
Was ist ein Bounded Context? . . . . .	63
Modellgrenzen . . . . .	64
Verbesserte Ubiquitous Language . . . . .	65
Gültigkeitsbereich eines Bounded Context . . . . .	65
Bounded Contexts versus Subdomains. . . . .	66
Subdomains . . . . .	67
Bounded Contexts . . . . .	67
Die Verbindung zwischen Subdomains und Bounded Contexts . . . . .	67
Grenzen. . . . .	69
Physische Grenzen . . . . .	69
Zuständigkeitsgrenzen . . . . .	70
Bounded Contexts in der Realität. . . . .	70
Semantic Domains . . . . .	71
Wissenschaft . . . . .	72
Einen Kühlschrank kaufen. . . . .	72
Zusammenfassung . . . . .	74
Übungen . . . . .	75
<b>4 Bounded Contexts integrieren . . . . .</b>	<b>77</b>
Kooperation . . . . .	78
Partnership . . . . .	78
Shared Kernel. . . . .	79
Customer/Supplier . . . . .	81
Conformist . . . . .	81
Anticorruption Layer . . . . .	82
Open-Host Service . . . . .	83
Separate Ways. . . . .	84
Kommunikationsprobleme . . . . .	84
Generic Subdomains . . . . .	85
Modellunterschiede . . . . .	85

Context Map . . . . .	85
Wartung . . . . .	86
Einschränkungen . . . . .	86
Zusammenfassung . . . . .	87
Übungen . . . . .	88

---

## Teil II Taktisches Design

<b>5 Einfache Business-Logik implementieren . . . . .</b>	<b>91</b>
Transaction Script . . . . .	91
Implementierung . . . . .	92
So einfach ist es nicht!. . . . .	92
Wann man ein Transaction Script verwendet . . . . .	97
Active Record. . . . .	98
Implementierung . . . . .	98
Wann sollte Active Record genutzt werden? . . . . .	99
Seien Sie pragmatisch . . . . .	100
Zusammenfassung. . . . .	100
Übungen . . . . .	101
<b>6 Komplexe Business-Logik angehen . . . . .</b>	<b>103</b>
Geschichte . . . . .	103
Domain Model. . . . .	103
Implementierung . . . . .	104
Bausteine. . . . .	105
Komplexität managen. . . . .	122
Zusammenfassung. . . . .	123
Übungen . . . . .	124
<b>7 Die zeitliche Dimension modellieren . . . . .</b>	<b>127</b>
Event Sourcing. . . . .	127
Suche. . . . .	132
Analyse . . . . .	133
Source of Truth. . . . .	135
Event Store . . . . .	135
Event-Sourced Domain Model . . . . .	136
Vorteile . . . . .	138
Nachteile. . . . .	139
Häufige Fragen . . . . .	140

Performance . . . . .	140
Daten löschen. . . . .	142
Warum kann ich nicht einfach ...? . . . . .	142
Zusammenfassung . . . . .	143
Übungen . . . . .	144
<b>8 Architektur-Patterns . . . . .</b>	<b>145</b>
Business-Logik versus Architektur-Patterns . . . . .	145
Schichtenarchitektur. . . . .	146
Presentation Layer . . . . .	146
Business Logic Layer . . . . .	147
Data Access Layer . . . . .	148
Kommunikation zwischen den Schichten . . . . .	148
Variation . . . . .	149
Wann Schichtenarchitektur genutzt werden sollte . . . . .	153
Ports & Adapters . . . . .	154
Terminologie . . . . .	154
Dependency Inversion Principle . . . . .	154
Integration von Infrastrukturkomponenten. . . . .	155
Varianten . . . . .	156
Wann man Ports & Adapters verwendet . . . . .	157
Command-Query Responsibility Segregation. . . . .	157
Polyglot Modeling . . . . .	157
Implementierung . . . . .	158
Read Models projizieren . . . . .	158
Herausforderungen . . . . .	161
Model Segregation . . . . .	161
Wann CQRS verwendet wird . . . . .	162
Scope. . . . .	162
Zusammenfassung . . . . .	164
Übungen . . . . .	164
<b>9 Kommunikations-Patterns. . . . .</b>	<b>167</b>
Model Translation . . . . .	167
Stateless Model Translation. . . . .	168
Stateful Model Translation . . . . .	171
Aggregates integrieren. . . . .	173
Outbox. . . . .	175
Saga . . . . .	177
Process Manager . . . . .	180
Zusammenfassung . . . . .	184
Übungen . . . . .	184

---

## Teil III Domain-Driven Design in der Praxis umsetzen

<b>10 Designheuristiken</b> .....	<b>189</b>
Heuristik .....	189
Bounded Contexts .....	190
Implementierungs-Patterns für Business-Logik .....	191
Architektur-Patterns .....	193
Teststrategie .....	194
Testpyramide .....	195
Testdiamant .....	195
Umgekehrte Testpyramide .....	195
Entscheidungsbaum für taktisches Design .....	196
Zusammenfassung .....	197
Übungen .....	197
<b>11 Evolution von Designentscheidungen</b> .....	<b>199</b>
Änderungen an Domains .....	199
Core nach Generic .....	200
Generic nach Core .....	200
Supporting nach Generic .....	201
Supporting nach Core .....	201
Core nach Supporting .....	202
Generic nach Supporting .....	202
Strategische Designaspekte .....	203
Taktische Designaspekte .....	203
Transaction Script nach Active Record .....	204
Active Record nach Domain Model .....	204
Domain Model nach Event-Sourced Domain Model .....	206
Alte Übergänge generieren .....	206
Migrationsevents modellieren .....	207
Organisationsänderungen .....	208
Partnership nach Customer/Supplier .....	209
Customer/Supplier nach Separate Ways .....	209
Domänenwissen .....	209
Wachstum .....	210
Subdomains .....	211
Bounded Contexts .....	212
Aggregates .....	212
Zusammenfassung .....	213
Übungen .....	214

<b>12 EventStorming</b> .....	<b>215</b>
Was ist EventStorming? .....	215
Wer sollte am EventStorming teilnehmen? .....	215
Was brauchen Sie zum EventStorming? .....	216
Der EventStorming-Prozess .....	217
Schritt 1: Unstrukturiertes Erforschen .....	217
Schritt 2: Zeitachsen .....	218
Schritt 3: Pain Points .....	219
Schritt 4: Pivotal Events .....	219
Schritt 5: Commands .....	220
Schritt 6: Policies .....	221
Schritt 7: Read Models .....	222
Schritt 8: Externe Systeme .....	223
Schritt 9: Aggregates .....	223
Schritt 10: Bounded Contexts .....	224
Varianten .....	224
Wann Sie EventStorming einsetzen .....	225
Tipps für das EventStorming .....	226
Achten Sie auf die Dynamik .....	227
Remote EventStorming .....	227
Zusammenfassung .....	227
Übungen .....	228
<b>13 Domain-Driven Design in der Praxis</b> .....	<b>229</b>
Strategische Analyse .....	230
Die Fachdomäne verstehen .....	230
Das aktuelle Design untersuchen .....	231
Modernisierungsstrategie .....	232
Strategisches Modernisieren .....	233
Taktisches Modernisieren .....	235
Eine Ubiquitous Language kultivieren .....	235
Pragmatisches Domain-Driven Design .....	239
Domain-Driven Design schmackhaft machen .....	239
Undercover Domain-Driven Design .....	240
Zusammenfassung .....	242
Übungen .....	243

---

## Teil IV DDD und andere Methodiken und Patterns

<b>14</b>	<b>Microservices</b> . . . . .	<b>247</b>
	Was ist ein Service? . . . . .	247
	Was ist ein Microservice? . . . . .	248
	Method as a Service: Perfekte Microservices? . . . . .	249
	Designziel . . . . .	250
	Systemkomplexität . . . . .	251
	Microservices als tiefe Services . . . . .	252
	Microservices als tiefe Module . . . . .	253
	Domain-Driven Design und die Grenzen von Microservices . . . . .	255
	Bounded Contexts . . . . .	255
	Aggregates . . . . .	257
	Subdomains . . . . .	258
	Die öffentlichen Schnittstellen von Microservices komprimieren . . . . .	259
	Open-Host Service . . . . .	259
	Anticorruption Layer . . . . .	260
	Zusammenfassung . . . . .	261
	Übungen . . . . .	261
<b>15</b>	<b>Event-Driven Architecture</b> . . . . .	<b>263</b>
	Event-Driven Architecture . . . . .	263
	Events . . . . .	264
	Events, Commands und Messages . . . . .	264
	Struktur . . . . .	265
	Event-Typen . . . . .	265
	Design einer Event-gesteuerte Integration . . . . .	270
	Verteilter Big Ball of Mud . . . . .	271
	Zeitliche Kopplung . . . . .	272
	Funktionelle Kopplung . . . . .	272
	Implementierungskopplung . . . . .	273
	Die Event-gesteuerte Integration refaktorisieren . . . . .	273
	Event-gesteuerte Designheuristiken . . . . .	274
	Zusammenfassung . . . . .	276
	Übungen . . . . .	276
<b>16</b>	<b>Data Mesh</b> . . . . .	<b>279</b>
	Analytisches Datenmodell versus transaktionales Datenmodell . . . . .	279
	Fact-Tabelle . . . . .	280
	Dimension-Tabelle . . . . .	282
	Analytische Modelle . . . . .	283

Plattformen zum Managen analytischer Daten . . . . .	284
Data Warehouse . . . . .	285
Data Lake . . . . .	287
Herausforderungen von Data Warehouse und Data Lake Architectures . . . . .	289
Data Mesh . . . . .	289
Daten anhand von Domains unterteilen . . . . .	290
Daten als Produkt . . . . .	291
Autonomie ermöglichen . . . . .	292
Ein Ökosystem schaffen . . . . .	292
Data Mesh und Domain-Driven Design kombinieren . . . . .	293
Zusammenfassung . . . . .	295
Übungen . . . . .	295
<b>Abschließende Worte . . . . .</b>	<b>297</b>
<b>Anhang A DDD anwenden: eine Fallstudie . . . . .</b>	<b>303</b>
<b>Anhang B Antworten auf die Übungsfragen . . . . .</b>	<b>319</b>
<b>Literatur . . . . .</b>	<b>327</b>
<b>Index . . . . .</b>	<b>329</b>