

Inhaltsverzeichnis

Vorwort **v**

I Einleitung **1**

1 Einführung in Aspektorientierung **3**

1.1 Von Assembler bis AspectJ 3

1.1.1 Maschinsprache und Assembler 4

1.1.2 Prozedurale Sprachen 4

1.1.3 Objektorientierte Programmierung (OOP) 6

1.1.4 Aspektorientierte Programmierung (AOP) 7

1.2 Klasse Konto – ein Fallbeispiel 7

1.2.1 Erweiterte Konto-Klasse 8

1.2.2 Refactoring 10

1.2.3 Erweiterung über Aspekte 11

1.3 Weaving 15

1.4 Was bedeutet AOP? 16

1.4.1 Crosscutting Concerns (Querschnittsbelange) 16

1.4.2 Probleme 19

1.4.3 Die Quadratur der Architektur 20

1.4.4 Die Antwort von AOP 21

1.4.5 Anatomie von AOP-Sprachen 22

1.4.6 Das Versprechen von AOP 23

1.5 Sprechen Sie Aspekt-Orientalisch?! 24

1.5.1 Concern und Crosscutting Concern 24

1.5.2 Aspekt 25

1.5.3 Joinpoint (Verbindungspunkt) 25

1.5.4 Pointcut (Schnittpunkt) 25

1.5.5 Advice (Empfehlung) 26

1.5.6 Introduction 27

1.6 Zusammenfassung 27

1.7 Übungen 28

2	Erste Schritte	29
2.1	Installation von AspectJ	29
2.1.1	Anforderungen	29
2.1.2	Installation des Compilers	30
2.2	Hello World	31
2.2.1	Einsatz als Java-Compiler	31
2.2.2	Hello-World-Aspekt	32
2.3	Ge»weav«te World	33
2.4	Welche IDE?	34
2.4.1	AspectJ-Plugin für Eclipse	35
2.4.2	Software-Update	35
2.4.3	Manuelle Installation	38
2.4.4	Installations-Check	39
2.5	Hello IDE	40
2.5.1	Projekt anlegen	40
2.5.2	World.java anlegen	41
2.5.3	WorldAspect.aj anlegen	41
2.5.4	AspectJ-Projekt kompilieren	44
2.6	Zusammenfassung	45
2.7	Übungen	45

II Einführung in AspectJ **49**

3	Joinpoints und Pointcuts	51
3.1	Joinpoints im Programm	51
3.1.1	Pointcuts im Programm	53
3.2	Joinpoints in AspectJ	53
3.2.1	call	53
3.2.2	execution	54
3.2.3	call-Konstruktor	54
3.2.4	execution-Konstruktor	54
3.2.5	get	54
3.2.6	set	55
3.2.7	handler	55
3.2.8	initialization	55
3.2.9	preinitialization	55
3.2.10	staticinitialization	55
3.3	Wildcards	56
3.3.1	Namensmuster	57
3.3.2	Unterklassen	58
3.3.3	Throws-Muster	58
3.4	Logische Operatoren	58

3.5	Signaturen	59
3.5.1	Typ-Signatur	59
3.5.2	Methoden-Signatur	60
3.5.3	Narrowing	61
3.5.4	Konstruktor-Signatur	63
3.5.5	Varargs...	63
3.5.6	Field-Signatur	64
3.5.7	Exception-Handler-Signatur	65
3.6	Zugriff auf den Kontext	65
3.6.1	thisJoinPoint-Methoden	65
3.6.2	thisJoinPointStaticPart-Methoden	69
3.6.3	Signatur ermitteln	69
3.6.4	SourceLocation getSourceLocation()	71
3.7	Zusammenfassung	71
3.8	Übungen	72
4	Pointcuts definieren	75
4.1	Anatomie eines Pointcuts	75
4.2	Beispiel: Verwaltung von Anforderungen	76
4.3	Primitive Pointcuts	78
4.3.1	Kombiniere...	79
4.3.2	Methoden-bezogene Pointcuts	79
4.3.3	Exception-Handling	86
4.3.4	Attribut-Zugriffe	88
4.4	Im Banne des Joinpoints	91
4.4.1	this	91
4.4.2	target	94
4.4.3	args	96
4.4.4	Args und Autoboxing in AspectJ 5	100
4.4.5	Args und Varargs	101
4.5	Kontrollfluss-basierte Bezeichner	101
4.5.1	cflow	101
4.5.2	cflowbelow	104
4.6	Programmtext-basierte Bezeichner	106
4.6.1	withincode	107
4.6.2	within	108
4.7	Dynamische Joinpoints	109
4.7.1	if	109
4.7.2	Eigene Muster mit if	111
4.8	adviceexecution	112
4.9	Anonyme Pointcuts	113
4.10	Zusammenfassung	114
4.11	Übungen	114

5	Advice	117
5.1	Ziehung der Lottozahlen	117
5.2	Definition	119
5.3	Kontext	121
5.3.1	Formale Parameter	121
5.3.2	Bleibende Wirkung	124
5.3.3	Boxing	125
5.3.4	Zugriff über Reflexion	127
5.3.5	Advices und Exceptions	129
5.4	Die verschiedenen Advice-Typen	131
5.4.1	Before-Advice	132
5.4.2	After-Advice	134
5.4.3	Around-Advice	140
5.5	Rangleien	150
5.5.1	Bestimmung der Vorrangregeln	151
5.5.2	Zirkuläre Abhängigkeiten	155
5.6	Zusammenfassung	155
5.7	Übungen	157
6	Intertype-Deklaration	159
6.1	Erweiterung bestehender Klassen	159
6.1.1	Methoden hinzufügen	160
6.1.2	Attribut hinzufügen	160
6.1.3	Logger-Objekte einstreuen	161
6.1.4	Für und Wider	163
6.2	Noch ein Beispiel: Persistierung	165
6.3	Übersicht über Intertype-Deklarationen	167
6.3.1	Konkrete Methoden	167
6.3.2	Abstrakte Methoden	168
6.3.3	Konstruktoren	168
6.3.4	Felder	169
6.4	Zugriffskontrolle	169
6.4.1	private	170
6.4.2	protected	170
6.5	Konfliktsituationen	171
6.5.1	Konfliktfreie Namensgleichheit	171
6.5.2	Konfliktbehebung über Vorrangregeln	172
6.5.3	Konflikte mit Klassen-Elementen	173
6.5.4	Vererbungsregeln	173
6.6	Eltern adoptieren	173
6.6.1	Interface hinzufügen	173
6.6.2	Mehr zu »declare parents«	174
6.6.3	Zwischenebene einziehen	175

6.7	Interfaces implementieren	176
6.7.1	Persistenzlösung über Interfaces	176
6.7.2	Alternative: abstrakte Oberklasse	179
6.7.3	Konfliktpotenzial	179
6.7.4	Dreiecksverhältnisse	181
6.7.5	Konstruktoren	182
6.7.6	Weitere Möglichkeiten	183
6.8	Deklaration der Rangliste	184
6.8.1	Der Tante-Emma-Laden	185
6.8.2	Aspekte mehrfach deklariert	187
6.8.3	Abstrakte Aspekte	187
6.8.4	Zirkuläre Abhängigkeiten	190
6.9	Aufzählungstypen (Enums).....	191
6.10	Statische Querschnittsbelange	192
6.10.1	Statisch oder nicht statisch?	193
6.10.2	Eigene Compiler-Meldungen	193
6.10.3	Softening von Exceptions	195
6.11	Zusammenfassung	199
6.12	Übungen.....	200
7	Aspekte	203
7.1	Aspekt-Struktur.....	203
7.2	Wohin mit den Aspekten?.....	204
7.2.1	Separate Aspekt-Datei	205
7.2.2	Kombiniere... (Aspekte mit Klassen)	206
7.3	Vererbungslehre (Extensions)	207
7.3.1	Abstrakte Aspekte	207
7.3.2	Klassen und Interfaces beerben	210
7.4	Instanziierung und Assoziationen	212
7.4.1	Singleton	212
7.4.2	Instanziierung pro Objekt	212
7.4.3	Instanziierung pro Kontrollfluss.....	215
7.4.4	Instanziierung pro Typ	217
7.5	Zugriff auf fremde Instanzen	218
7.5.1	Instanz pro Objekt	219
7.5.2	Instanz pro Typ	220
7.6	Privilegien	220
7.7	Zusammenfassung	222
7.8	Übungen.....	222

8	Annotations	225
8.1	Annotations in Java 5	225
8.1.1	Retention-Policy	227
8.1.2	Annotations zur Laufzeit	227
8.1.3	Vererbte Annotations	228
8.2	AspectJ-Annotations	228
8.2.1	Joinpoints und Annotations	229
8.2.2	Beispiel-Pointcuts	234
8.3	Annotations zur Laufzeit	237
8.3.1	@this, @target und @args	237
8.3.2	@within, @withincode und @annotation	239
8.4	Auf Annotations zugreifen	240
8.5	Annotation-Vererbung	241
8.6	Declare-Anweisungen mit Annotations	244
8.6.1	Errors und Warnings	244
8.6.2	declare parents	244
8.6.3	declare precedence	245
8.7	Declare Annotation	245
8.7.1	@type	245
8.7.2	@method und @constructor	246
8.7.3	@field	247
8.8	@AspectJ-Annotations	247
8.8.1	@Aspect-Deklaration	248
8.8.2	@Pointcuts	249
8.8.3	Advices	251
8.8.4	Intertype-Deklaration	254
8.8.5	@Declare-Anweisungen	255
8.9	Zusammenfassung	257
8.10	Übungen	258
9	Der AspectJ-Werkzeugkasten	261
9.1	Ausgelieferte Dokumentation	262
9.1.1	AspectJ-Guide	262
9.1.2	Eclipse-Online-Hilfen	263
9.1.3	AspectJ-API	263
9.2	Der Compiler (ajc) und seine Optionen	264
9.2.1	Quelldateien	264
9.2.2	Compiler-Optionen	265
9.2.3	Fauler thisJoinPoint	269
9.2.4	Xlint-Option	270
9.2.5	Xlint-Beispiele	272
9.2.6	Erziehungsmaßnahmen	274
9.2.7	Eclipse-Einstellungen	275

9.2.8	Warnungen unterdrücken	275
9.2.9	OutOfMemoryError	277
9.3	Der AspectJ-Doc-Generator(ajdoc)	277
9.3.1	ajdoc-Optionen	278
9.3.2	Input	279
9.3.3	Ein kleines Beispiel	280
9.3.4	Eine kleine Nachbetrachtung	283
9.4	Einbindung in Ant	284
9.4.1	Vorbereitung	284
9.4.2	Aufruf des Compilers	286
9.4.3	Erzeugung verschiedener Programm-Varianten	287
9.4.4	Nachträgliche Instrumentierung	287
9.4.5	Generierung der API-Dokumentation	289
9.5	Zusammenfassung	292
9.6	Übungen	292
10	Die AspectJ-IDE in Eclipse	295
10.1	Eine Übersicht	295
10.2	Warum Eclipse?	296
10.2.1	Eine Bestandsaufnahme	297
10.2.2	AspectJ-Projekte	298
10.2.3	Die Übersetzung	299
10.3	Der AspectJ-Visualizer	300
10.4	Crosscutting Comparison View	301
10.5	Debugging	302
10.5.1	toString() überschreiben	302
10.5.2	Keine Reflexion	303
10.6	Eclipse 4	303
10.7	Zusammenfassung	304
10.8	Übungen	304

III Fortgeschrittenes AspectJ 307

11	Instrumentierung von Bibliotheken	309
11.1	Aspekte implantieren... ..	309
11.2	Anpassung fertiger Bibliotheken	311
11.3	Erweiterung der JDK-Bibliothek	313
11.3.1	Compiler-Aufruf	313
11.3.2	Generierung mit Ant	313
11.3.3	Start der VM	314
11.3.4	Entwicklung unter Eclipse	315
11.3.5	Warnung zum Schluss	316

11.4	Zusammenfassung	317
11.5	Übungen	317
12	Pattern	319
12.1	Das Observer-Pattern	319
12.1.1	Die Java-Lösung	320
12.1.2	AspectJ-Lösung	321
12.2	Fertige Pattern	324
12.3	Pattern-Testing	325
12.3.1	NullPointerException	325
12.3.2	Stacktrace ausgeben	326
12.4	AspectJ-Pattern	327
12.4.1	Optimierungen	327
12.4.2	Robot1	327
12.4.3	Robot2	328
12.4.4	Robot3	329
12.4.5	Aussicht	330
12.5	Zusammenfassung	331
12.6	Übungen	331
13	Tipps	333
13.1	Troubleshooting	333
13.1.1	Classpath-Probleme	334
13.1.2	Zu viele Joinpoints erwischt!	335
13.1.3	System.out.println	338
13.1.4	Zombie-Pointcuts	338
13.1.5	Pointcuts in abgeleiteten Klassen	341
13.1.6	Mein Pointcut »matcht« nicht	342
13.2	Missverständnisse	344
13.2.1	Interfaces	344
13.2.2	Unterschiede zwischen »Call« und »Execution«	344
13.2.3	call() && execution()	345
13.2.4	call() oder execution()	346
13.2.5	this() und target()	346
13.3	Kurze Methoden	347
13.3.1	Refactoring	348
13.3.2	Sprechende Namen	349
13.3.3	Auswirkung auf AOP	349
13.4	Setter- und Getter-Methoden	350
13.5	Testen	352
13.5.1	Test-Aspekte	352
13.5.2	Kontext aufsetzen	353
13.5.3	Abgestuftes Testen	356

13.5.4	Design by Contract	360
13.6	Vom Umgang mit Aspekten	361
13.6.1	Die dunkle Seite der Aspekte	361
13.6.2	Knappe Pointcuts	362
13.7	Aspekte in der Praxis	363
13.7.1	Performance-Analyse	363
13.7.2	Profiler-Aspekt	363
13.7.3	Performance-Einbußen	364
13.8	Sonstige Tipps	365
13.8.1	Endung »aj« verwenden	365
13.8.2	Zusammenlassen, was zusammengehört	365
13.8.3	Einheitliches Namensschema	366
13.8.4	Getrennter Source-Pfad	366
13.8.5	Nachträgliche Instrumentierung im Build-Prozess	367
13.9	Zusammenfassung	368
13.10	Übungen	368
14	Goto AspectJ	371
14.1	Gründe für AspectJ	371
14.1.1	Erwarteter Nutzen	372
14.1.2	Performance	372
14.1.3	Gefahren	373
14.1.4	Erfahrungen	373
14.2	Einführungsstrategien	374
14.2.1	Top-down (Unterstützung des Managements).....	374
14.2.2	Bottom-up (Don't tell)	375
14.2.3	Über die QM-Schiene	376
14.3	Zukunftsaspekte	376
14.3.1	Hype or Hope?	377
14.3.2	Der Blick nach vorn	377
14.4	Zusammenfassung	379
IV	Anhang	381
A	Abkürzungen	383
B	Glossar	385
C	Einstellungssache	389
C.1	Asserts aktivieren	389
C.2	Inkrementelle Kompilierung	389
C.3	Java 5	389

C.4	»Cross References«-View	389
D	Übungsbeispiele und Lösungen	393
D.1	Die Webseite zum Buch.....	393
D.2	Die Fehler zum Buch.....	394
D.3	Die Lösungen zu den Übungen	394
	Abbildungsverzeichnis	405
	Literaturverzeichnis	409
	Index	411