

## Objektorientierte Softwareentwicklung mit UML

# DAS INHALTS- VERZEICHNIS

» Hier geht's  
direkt  
zum Buch

# Inhalt

<b>Vorwort</b> .....	<b>IX</b>
<b>Zusatzmaterial zum Buch</b> .....	<b>XIII</b>
<b>1 Grundbegriffe der objektorientierten Softwareentwicklung</b> .....	<b>1</b>
1.1 Einführung .....	1
1.2 Konzepte und Notationen .....	7
1.2.1 Basismodell .....	7
1.2.2 Statisches Modell .....	11
1.2.3 Dynamisches Modell .....	23
1.2.4 Modell der Systemnutzung .....	28
<b>2 UML – Unified Modeling Language</b> .....	<b>31</b>
2.1 Entwicklung der Sprache .....	31
2.2 Anwendungsfallmodelle .....	34
2.2.1 Beschreibung von Anwendungsfällen .....	37
2.2.2 Beschreibung von Szenarien und Anwendungsfällen .....	42
2.3 Klassenmodelle .....	66
2.3.1 Klassen und Objekte .....	66
2.3.2 Metaklassen .....	88
2.3.3 Schnittstellen .....	92
2.3.4 Generische Klassen .....	96
2.3.5 Pakete .....	99
2.3.6 Objekte .....	102
2.3.7 Komponenten .....	103
2.3.8 Abhängigkeiten .....	106
2.3.9 Entwurfsmuster .....	111
2.4 Verhaltensmodelle .....	118
2.4.1 Zustandsdiagramm .....	118
2.4.2 Aktivitätsdiagramm .....	139
2.5 Object Constraint Language (OCL) .....	163
2.5.1 Einführung .....	163
2.5.2 Sprachkonstrukte .....	164

2.5.3	Operationen und Iteratoren .....	168
2.5.3.1	Vordefinierte Operationen auf allen Objekten .....	170
2.5.3.2	Operationen auf den Basistypen Set, Bag und Sequence ...	171
2.5.4	Schlussbemerkungen .....	175
<b>3</b>	<b>Von der Analyse zur Implementierung .....</b>	<b>177</b>
3.1	Überblick .....	177
3.2	Analyse .....	186
3.2.1	CRC-Karten .....	186
3.2.2	Anwendungsfallanalyse .....	190
3.2.3	Modellbasierte Analyse .....	190
3.2.4	Geschäftsprozessanalyse .....	196
3.3	Entwurf .....	198
3.3.1	Anwendungsfallorientierter Entwurf .....	198
3.3.2	Von der Analyse zum Entwurf .....	199
3.3.3	Entwurfsmuster .....	201
3.3.4	Unterstützung der Modelltransformationen .....	218
3.4	Implementierung .....	221
3.4.1	Anwendungsfallorientierte Vorgehensweise .....	221
3.4.2	Generalisation versus Aggregation .....	221
3.4.3	Interface versus abstrakte Klasse .....	223
3.4.4	Herausforderungen bei objektorientierten Programmen .....	224
3.4.4.1	Konsistenz beim Verhalten .....	224
3.4.4.2	Invarianz, Kovarianz und Kontravarianz .....	226
3.5	Werkzeugunterstützung .....	239
<b>4</b>	<b>Kollaborative Analyse und Design .....</b>	<b>241</b>
4.1	Einführung .....	241
4.1.1	Besonderheiten und Qualität von Software .....	241
4.1.2	Softwareentwicklung als interdisziplinärer Modellbildungsprozess ...	245
4.2	Benutzerorientierte Entwicklungsansätze .....	246
4.2.1	User-Centered Design (UCD) .....	246
4.2.2	Partizipative Softwareentwicklung .....	249
4.3	Software- und Kontextmodelle .....	250
4.3.1	Softwaremodelle .....	251
4.3.1.1	User Stories .....	251
4.3.1.2	Skizzen und Prototypen für Benutzungsoberflächen .....	253
4.3.2	Kontextmodelle .....	255
4.3.2.1	Kognitive Aufgabenmodelle .....	256
4.3.2.2	Personas .....	264
4.3.2.3	Szenarien .....	266
4.4	Systematischer und kreativer Umgang mit Modellen .....	269
4.4.1	Kollaboratives Erstellen von Anwendungsfällen .....	270
4.4.2	Szenarienbasierte Systemgestaltung .....	271
4.4.3	Integration von Use Cases und Personas .....	275

4.4.4	Integrierte Nutzung verschiedener Softwaremodelle . . . . .	281
4.4.4.1	Verbindung von Use Cases, UI-Prototypen und Klassen . . . . .	281
4.4.4.2	Verbindung von Use Cases, UI-Prototypen und Zustands- diagrammen . . . . .	283
4.5	Zusammenfassung von Kapitel 4 . . . . .	287
	<b>Literatur . . . . .</b>	<b>289</b>
	<b>Index . . . . .</b>	<b>295</b>