

Der Inhalt (im Überblick)

	Einführung	xxiii
1	Willkommen bei den Entwurfsmustern: <i>Einführung in Entwurfsmuster</i>	1
2	Ihre Objekte auf dem Laufenden halten: <i>Das Observer-Muster</i>	37
3	Objekte dekorieren: <i>Das Decorator-Muster</i>	79
4	In der OO-Bäckerei ...: <i>Das Factory-Muster</i>	109
5	Einmalige Objekte: <i>Das Singleton-Muster</i>	169
6	Aufrufe verkapseln: <i>Das Command-Muster</i>	191
7	Anpassungsfähigkeit beweisen: <i>Die Adapter- und Facade-Muster</i>	237
8	Algorithmen verkapseln: <i>Das Template Method-Muster</i>	277
9	Erfolgreiche Collections: <i>Die Iterator- und Composite-Muster</i>	317
10	Der (Zu-)Stand der Dinge: <i>Das State-Muster</i>	381
11	Objektzugriff kontrollieren: <i>Das Proxy-Muster</i>	425
12	Muster von Mustern: <i>Zusammengesetzte Muster</i>	493
13	Muster in der wahren Welt: <i>Schöner leben mit Mustern</i>	563
14	Anhang: <i>Übrig gebliebene Muster</i>	597

Inhalt (jetzt ausführlich)

Einführung

Für wen ist dieses Buch?	xxiv
Wir wissen, was Sie gerade denken	xxv
Und wir wissen, was Ihr Gehirn gerade denkt	xxv
Metakognition: Nachdenken übers Denken	xxvii
Machen Sie sich Ihr Hirn untertan	xxix
Fachgutachter	xxxii
Danksagungen	xxxiv

1

Willkommen bei den Entwurfsmustern

Es begann mit einer einfachen SimUDuck-App	2
Aber jetzt sollen die Enten FLIEGEN können	3
Aber irgendetwas ging furchtbar schief ...	4
Joe denkt über Vererbung nach ...	5
Wie wäre es mit einem Interface?	6
Was würden Sie an Joes Stelle tun?	7
Die einzige Konstante in der Softwareentwicklung	8
Das Problem eingrenzen	9
Veränderliches und Unveränderliches voneinander trennen	10
Das Entenverhalten entwerfen	11
Das Entenverhalten implementieren	13
Das Entenverhalten integrieren	15
Den Entencode testen	18
Verhalten dynamisch festlegen	20
Das große Ganze: Verkapseltes Verhalten	22
HAT-EIN ist besser als IST-EIN	23
Da wir gerade von Entwurfsmustern sprechen ...	24
Im Bistro um die Ecke aufgeschnappt ...	26
Im Büro nebenan aufgeschnappt ...	27
Die Stärke eines gemeinsamen Mustervokabulars	28
Wie setze ich Entwurfsmuster ein?	29
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	32

Das Observer-Muster

2

Ihre Objekte auf dem Laufenden halten

Die Wetterstation im Überblick	39
Willkommen zum Observer-Muster	44
Herausgeber + Abonnenten = Observer-Muster	45
Die Observer-Muster-Definition	51
Die Macht der losen Kopplung	54
Die Wetterstation entwerfen	57
Die Wetterstation implementieren	58
Die Wetterstation hochfahren	61
Das Observer-Muster in freier Wildbahn	65
Die lebensverändernde Applikation programmieren	66
Inzwischen bei Weather-O-Rama	69
Probefahrt für den neuen Code	71
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	72
Die Entwurfsprinzipien-Challenge	73

Das Decorator-Muster

3

Objekte dekorieren

Willkommen bei Starbuzz Coffee	80
Das Offen/Geschlossen-Prinzip	86
Wir stellen vor: das Decorator-Muster	88
Eine Getränkebestellung mit Dekoratoren aufbauen	89
Die Definition des Decorator-Musters	91
Unsere Getränke dekorieren	92
Den Starbuzz-Code schreiben	95
Getränke programmieren	96
Zutaten programmieren	97
Den Kaffee servieren	98
Dekoratoren in freier Wildbahn: Java I/O	100
Die java.io-Klassen dekorieren	101
Einen eigenen Java-I/O-Dekorator schreiben	102
Unseren neuen Java-I/O-Dekorator testen	103
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	105

Das Factory-Muster

4

In der OO-Bäckerei ...

Das Veränderliche finden	112
Die Objekterstellung verkapseln	114
Eine einfache Pizzafabrik erstellen	115
Die einfache Fabrik definieren	117
Ein Framework für die Pizzeria	120
Den Subklassen die Entscheidung überlassen	121
Eine Fabrikmethode deklarieren	125
Jetzt ist es endlich Zeit, das Factory Method-Muster kennenzulernen	131
Ein paralleler Blick auf Hersteller und Produkte	132
Die Definition des Factory Method-Musters	134
Ein Blick auf Objektabhängigkeiten	138
Das Prinzip der Umkehrung der Abhängigkeiten	139
Das Prinzip anwenden	140
Zutatenfamilien	145
Die Zutatenfabriken bauen	146
Die Pizzas überarbeiten ...	149
Unsere Pizzerien überarbeiten	152
Was haben wir getan?	153
Die Definition des Abstract Factory-Musters	156
Factory Method und Abstract Factory im Vergleich	160
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	162

Das Singleton-Muster

5

Einmalige Objekte

Die Implementierung des klassischen Singleton-Musters im Detail	173
Die Schokoladenfabrik	175
Definition des Singleton-Musters	177
^{Brüssel} Houston , wir haben ein Problem ...	178
Mit Multithreading umgehen	180
Können wir das Multithreading verbessern?	181
Inzwischen in der Schokoladenfabrik ...	183
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	186

Das Command-Muster

6

Aufrufe verkapseln

Ein Blick auf die Drittherstellerklassen	194
Inzwischen im Bistro ...	197
Vom Bistro zum Command-Muster	201
Unser erstes Command-Objekt	203
Das Command-Objekt verwenden	204
Befehle den Plätzen zuweisen	209
Die Fernsteuerung implementieren	210
Die Befehle implementieren	211
Die Fernsteuerung auf Herz und Nieren testen	212
Zeit für die Dokumentation ...	215
Was machen wir hier?	217
Zeit, den Rückgängig-Knopf auf seine Qualität zu testen!	220
Zustände für die Implementierung der »Rückgängig«-Funktion verwenden	221
Die Deckenventilator-Befehle mit einer »Rückgängig«-Funktion versehen	222
Jede Fernsteuerung braucht einen Partymodus!	225
Einen Makro-Befehl benutzen	226
Viele Verwendungsmöglichkeiten für das Command-Muster: Warteschlangen für Befehle	229
Weitere Anwendungen des Command-Musters: Aufträge protokollieren	230
Das Command-Muster in der wahren Welt	231
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	233

Die Adapter- und Facade-Muster

7 Anpassungsfähigkeit beweisen

Überall Adapter	238
Objektorientierte Adapter	239
Probelauf für den Adapter	242
Das Adapter-Muster erklärt	243
Die Definition des Adapter-Musters	245
Objekt- und Klassen-Adapter	246
Adapter im echten Leben	250
Einen Enumerator an einen Iterator anpassen	251
Unser eigenes Heimkino	257
Einen Film ansehen (auf die harte Tour)	258
Licht, Kamera, Facade!	260
Die Heimkino-Facade konstruieren	263
Die vereinfachte Schnittstelle implementieren	264
Einen Film anschauen (auf die sanfte Tour)	265
Die Definition des Facade-Musters	266
Das Prinzip der Verschwiegenheit	267
Wie man KEINE Freunde gewinnt und KEINE Objekte beeinflusst	268
Das Facade-Muster und das Prinzip der Verschwiegenheit	271
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	272

8

Algorithmen verkapseln

Zeit für mehr Koffein	278
Ein paar Kaffee- und Tee-Klassen zusammenrühren (in Java)	279
Kaffee und Tee abstrahieren	282
Das Design verbessern	283
prepareRecipe() abstrahieren	284
Was haben wir getan?	287
Willkommen beim Template Method-Muster	288
Was hat uns das Template Method-Muster gebracht?	290
Die Definition des Template Method-Musters	291
Eingehängt in die Template Method ...	294
Den Hook verwenden	295
Das Hollywood-Prinzip und das Template Method-Muster	299
Template-Methoden in freier Wildbahn	301
Mit dem Template Method-Muster sortieren	302
Was ist compareTo?	303
Enten mit Enten vergleichen	304
Ein paar Enten sortieren	305
Das Making-of der Entensortiermaschine	306
Swinging mit Frames	308
Eigene Listen mit AbstractList	309
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	313

Die Iterator- und Composite-Muster

9 Erfolgreiche Collections

Große Neuigkeiten: Das Restaurant und das Pfannkuchenhaus von Objectville fusionieren	318
Sehen wir uns die Gerichte an	319
Die Spezifikation implementieren: unser erster Versuch	323
Können wir die Iteration verkapseln?	325
Willkommen zum Iterator-Muster	327
DinerMenu mit einem Iterator versehen	328
Die Restaurant-Speisekarte mit einem Iterator überarbeiten	329
Den Kellnerin-Code aufmöbeln	330
Den Code testen	331
Unser aktueller Entwurf auf dem Prüfstand ...	333
Aufräumen mit java.util.Iterator	335
Die Definition des Iterator-Musters	338
Die Struktur des Iterator-Musters	339
Das Prinzip der einzelnen Verantwortlichkeit	340
Willkommen zu Javas Iterable-Interface	343
Javas erweiterte for-Schleife	344
Ein Blick auf die Speisekarte des Cafés	347
Iteratoren und Collections	353
Die Definition des Composite-Musters	360
Speisekarten mit dem Composite-Muster entwerfen	363
MenuComponent implementieren	364
Das Gericht (MenuItem) implementieren	365
Die Komposita-Speisekarte implementieren	366
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	376

Ein echter Java-Plombenzieher	382
Kurze Einführung in Zustandsautomaten	384
Den Code schreiben	386
Interner Testlauf	388
Sie haben es geahnt – ein Änderungswunsch!	390
ZUSTÄNDE wie bei Hempels unterm Sofa ...	392
Der neue Entwurf	394
Das Interface State und die Klassen definieren	395
Umbau des Kaugummiautomaten	398
Ein Blick auf die komplette Klasse GumballMachine ...	399
Weitere Zustände implementieren	400
Die Definition des State-Musters	406
Wir müssen uns wieder dem 1-von-10-Kaugummispiel widmen	409
Das Spiel fertigstellen	410
Demo für den CEO von Mighty Gumball, Inc.	411
Stimmt alles?	413
Das haben wir fast vergessen!	416
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	419

11 Objektzugriff kontrollieren

Den Überwachungscode schreiben	427
Den Überwachungscode testen	428
Einführung in entfernte Methodenaufrufe	433
Den Kaugummiautomaten (GumballMachine) als entfernten Dienst einrichten	446
Bei der RMI-Registry anmelden ...	448
Die Definition des Proxy-Musters	455
Bereitmachen für den virtuellen Proxy	457
Den virtuellen Proxy für die Albencover entwerfen	459
Den Bild-Proxy schreiben	460
Partnervermittlung für Geeks in Objectville	470
Die Person implementieren	471
Fünf-Minuten-Drama: Subjekte schützen	473
Das große Ganze: Einen dynamischen Proxy für Person erstellen	474
Der Proxy-Zoo	482
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	485
Der Code für den Albumcover-Viewer	489

Mustergültige Zusammenarbeit	494
Ein Wiedersehen mit den Enten	495
Was haben wir getan?	517
Ein Blick aus der Vogel Entenperspektive: das Klassendiagramm	518
Der König der zusammengesetzten Muster	520
Willkommen zu Model-View-Controller	523
Genauer hingesehen ...	524
MVC als eine Reihe von Mustern verstehen	526
MVC, um den Beat zu steuern	528
Erstellung der Einzelteile	531
Ein Blick auf die konkrete Klasse BeatModel	532
Der View	533
Den View implementieren	534
Und damit zum Controller	536
Die Einzelteile zusammensetzen ...	538
Strategy erforschen	539
Das Model adaptieren	540
Noch ein Probelauf ...	542
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	545

Schöner leben mit Mustern

13

Muster in der wahren Welt

Definition von Entwurfsmustern	565
Ein genauerer Blick auf die Entwurfsmusterdefinition	567
Möge die Macht mit Ihnen sein	568
Sie wollen also eigene Entwurfsmuster schreiben	573
Entwurfsmuster ordnen	575
In Mustern denken	580
Denk-Muster	583
Vergessen Sie nicht die Macht des gemeinsamen Vokabulars	585
Spritztour durch Objectville mit der Gang of Four	587
Ihre Reise hat gerade erst begonnen!	588
Der Muster-Zoo	590
Mit Anti-Mustern das Böse auslöschen	592
Werkzeuge für Ihren Entwurfs-Werkzeugkasten	594
Abschied von Objectville ...	595

14 Anhang: Übrig gebliebene Muster

Bridge	598
Builder	600
Chain of Responsibility	602
Flyweight	604
Interpreter	606
Mediator	608
Memento	610
Prototype	612
Visitor	614



Index

617