

HANSER



Inhaltsverzeichnis

Craig Walls

Spring im Einsatz

ISBN: 978-3-446-42388-6

Weitere Informationen oder Bestellungen unter

<http://www.hanser.de/978-3-446-42388-6>

sowie im Buchhandel.

Inhalt

Vorwort	XIII
Danksagungen	XV
Zu diesem Buch	XVII
Teil I: Der Kern von Spring	1
1 Spring ins kalte Wasser	3
1.1 Vereinfachte Java-Entwicklung	4
1.1.1 Die Kraft der POJOs entfesseln	5
1.1.2 Abhängigkeiten injizieren	6
1.1.3 Aspekte anwenden	10
1.1.4 Boilerplate-Code durch Vorlagen eliminieren	15
1.2 Beans im Container	17
1.2.1 Mit einem Anwendungskontext arbeiten	18
1.2.2 Ein Bohnenleben	19
1.3 Begutachtung der Spring-Landschaft	21
1.3.1 Die Spring-Module	21
1.3.2 Das Spring-Portfolio	24
1.4 Was ist neu?	28
1.4.1 Was ist neu bei Spring 2.5?	28
1.4.2 Was ist neu bei Spring 3.0?	29
1.4.3 Was ist neu im Spring-Portfolio?	29
1.5 Zusammenfassung	30
2 Verschalten von Beans	31
2.1 Deklarieren von Beans	32
2.1.1 Das Setup der Spring-Konfiguration	32
2.1.2 Eine einfache Bean deklarieren	34
2.1.3 Injektion über Konstruktoren	35
2.1.4 Geltungsbereiche für Beans	39
2.1.5 Beans initialisieren und zerstören	40
2.2 Injektion in Bean-Eigenschaften	42
2.2.1 Einfache Werte injizieren	43
2.2.2 Andere Beans referenzieren	44

2.2.3	Eigenschaften mit dem Spring-Namensraum p verschalten	48
2.2.4	Collections miteinander verschalten	48
2.2.5	Nichts (null) verschalten	53
2.3	Mit Ausdrücken verschalten	54
2.3.1	SpEL-Grundlagen	55
2.3.2	Operationen mit SpEL-Werten ausführen	58
2.3.3	Collections in SpEL	63
2.4	Zusammenfassung	67
3	Die XML-Konfiguration in Spring minimalisieren	69
3.1	Bean-Eigenschaften automatisch verschalten	70
3.1.1	Die vier Typen des Autowiring	70
3.1.2	Standard-Autowiring	74
3.1.3	Gemischter Einsatz von automatischer und expliziter Verschaltung	75
3.2	Verschalten mit Annotationen	76
3.2.1	Die Arbeit mit @Autowired	76
3.2.2	Auf Standards basierendes Autowiring mit @Inject	80
3.2.3	Annotationsinjektion und Expressions	83
3.3	Automatische Erkennung von Beans	84
3.3.1	Beans für Autodiscovery annotieren	84
3.3.2	Filtern mit component-scan	85
3.4	Die Java-basierte Konfiguration von Spring	87
3.4.1	Java-basierte Konfiguration einrichten	87
3.4.2	Eine Konfigurationsklasse definieren	88
3.4.3	Eine einfache Bean deklarieren	88
3.4.4	Injizieren mit der Java-basierten Konfiguration von Spring	89
3.5	Zusammenfassung	90
4	Aspektorientierung	91
4.1	Einführung in AOP	92
4.1.1	Die Terminologie zu AOP	93
4.1.2	Die AOP-Unterstützung in Spring	96
4.2	Joinpoints mit Pointcuts auswählen	98
4.2.1	Pointcuts schreiben	99
4.2.2	Der bean()-Bezeichner von Spring	100
4.3	Aspekte in XML deklarieren	101
4.3.1	Before- und After-Advice deklarieren	102
4.3.2	Around-Advice deklarieren	104
4.3.3	Parameter an Advices übergeben	106
4.3.4	Neue Funktionalitäten mit Aspekten	108
4.4	Aspekte annotieren	110
4.4.1	Around-Advices annotieren	112
4.4.2	Argumente an annotierte Advices übergeben	113
4.4.3	Introductions annotieren	114
4.5	AspectJ-Aspekte injizieren	115
4.6	Zusammenfassung	118

Teil II: Wesentliches über Spring-Anwendungen	119
5 Zugriff auf die Datenbank	121
5.1 Die Philosophie des Datenzugriffs in Spring	122
5.1.1 Die Exception-Hierarchie beim Datenzugriff in Spring	123
5.1.2 Datenzugriffsschablonen	126
5.1.3 Klassen für den DAO-Support	128
5.2 Datenquelle konfigurieren	129
5.2.1 JNDI-Datenquellen verwenden	129
5.2.2 Pooldatenquelle verwenden	130
5.2.3 Auf JDBC-Treiber basierende Datenquelle	132
5.3 JDBC und Spring gemeinsam verwenden	133
5.3.1 Wie man JDBC-Code in den Griff bekommt	133
5.3.2 Mit JDBC-Vorlagen arbeiten	136
5.4 Hibernate und Spring integrieren	142
5.4.1 Hibernate – ein Überblick	143
5.4.2 Wie man eine Hibernate-Session-Factory deklariert	144
5.4.3 Spring-freies Hibernate bauen	146
5.5 Spring und die Java Persistence API	147
5.5.1 Entity-Manager-Factory konfigurieren	148
5.5.2 DAO auf JPA-Basis erstellen	152
5.6 Zusammenfassung	154
6 Transaktionen verwalten	155
6.1 Grundlagen zu Transaktionen	156
6.1.1 Transaktionen – mit vier Wörtern erklärt	157
6.1.2 Spring-Support für die Transaktionsverwaltung	158
6.2 Auswahl eines Transaktionsmanagers	159
6.2.1 JDBC-Transaktionen	160
6.2.2 Hibernate-Transaktionen	161
6.2.3 JPA-Transaktionen	161
6.2.4 JTA-Transaktionen	162
6.3 Transaktionen in Spring programmieren	163
6.4 Transaktionen deklarieren	165
6.4.1 Transaktionsattribute definieren	166
6.4.2 Transaktionen in XML deklarieren	170
6.4.3 Annotationsgetriebene Transaktionen definieren	172
6.5 Zusammenfassung	174
7 Webapplikationen mit Spring MVC erstellen	175
7.1 Der Start mit Spring MVC	176
7.1.1 Einem Request durch Spring MVC folgen	176
7.1.2 Das Setup von Spring	178
7.2 Einen Basis-Controller schreiben	180
7.2.1 Konfiguration eines annotationsgetriebenen Spring MVC	181
7.2.2 Den Homepage-Controller definieren	182

7.2.3	Auflösung von Views	184
7.2.4	Definition der Homepage-View	188
7.2.5	Den Spring-Anwendungskontext vervollständigen	191
7.3	Controller-Input bearbeiten	192
7.3.1	Controller schreiben, der Input verarbeitet	192
7.3.2	Den View darstellen	195
7.4	Formulare verarbeiten	196
7.4.1	Das Registrierungsformular darstellen	197
7.4.2	Formulareingaben verarbeiten	199
7.4.3	Eingaben validieren	201
7.5	Der Umgang mit Datei-Uploads	205
7.5.1	Datei-Upload-Feld im Registrierungsformular	205
7.5.2	Hochgeladene Dateien empfangen	206
7.5.3	Spring für Datei-Uploads konfigurieren	209
7.6	Zusammenfassung	210
8	Die Arbeit mit Spring Web Flow	211
8.1	Installieren von Spring Web Flow	212
8.1.1	Konfigurieren von Web Flow in Spring	212
8.2	Die Komponenten eines Flows	215
8.2.1	Zustände	215
8.2.2	Transitionen	219
8.2.3	Ablaufdaten	220
8.3	Zusammensetzen der Einzelteile: Der Pizza-Ablauf	222
8.3.1	Definieren des Basisablaufs	222
8.3.2	Kundeninformationen sammeln	226
8.3.3	Eine Bestellung bilden	231
8.3.4	Zahlungsannahme	234
8.4	Absichern von Web-Flows	236
8.5	Zusammenfassung	236
9	Spring absichern	239
9.1	Einführung in Spring Security	240
9.1.1	Der Start mit Spring Security	240
9.1.2	Der Konfigurationsnamensraum von Spring Security	241
9.2	Webanfragen absichern	242
9.2.1	Proxys für Servlet-Filter erstellen	243
9.2.2	Minimale Websicherheit konfigurieren	244
9.2.3	Requests abfangen	247
9.3	Die Absicherung von Elementen auf View-Ebene	250
9.3.1	Der Zugriff auf Authentifizierungsdetails	251
9.3.2	Darstellung nach jeweiliger Befugnis	252
9.4	Benutzer authentifizieren	254
9.4.1	Konfiguration eines speicherresidenten User-Repositorys	255
9.4.2	Authentifizierung gegenüber einer Datenbank	256
9.4.3	Authentifizierung gegenüber einem LDAP-Repository	258

9.4.4	Die Funktionalität „Auf diesem Computer merken“	261
9.5	Methoden sichern	262
9.5.1	Mit @Secured annotierte Methoden sichern	263
9.5.2	@RolesAllowed von JSR-250 verwenden	263
9.5.3	SpEL-Sicherheit mit Pre- und Post-Invocation	264
9.5.4	Sicherheits-Pointcuts auf Methodenebene deklarieren	269
9.6	Zusammenfassung	269
Teil III: Spring integrieren		271
10	Die Arbeit mit Remote-Diensten	273
10.1	Das Spring-Remoting im Überblick	274
10.2	Die Arbeit mit RMI	276
10.2.1	Einen RMI-Service exportieren	277
10.2.2	Einen RMI-Service verschalten	279
10.3	Remote-Services mit Hessian und Burlap veröffentlichen	282
10.3.1	Bean-Funktionalitäten mit Hessian und Burlap verfügbar machen	282
10.3.2	Zugriff auf Hessian-/Burlap-Services	285
10.4	Springs HTTP-Invoker verwenden	286
10.4.1	Beans als HTTP-Services verfügbar machen	287
10.4.2	Zugriff auf Services via HTTP	288
10.5	Webservices veröffentlichen und weiterverarbeiten	289
10.5.1	Spring-fähige JAX-WS-Endpunkte erstellen	291
10.5.2	JAX-WS-Services auf Client-Seite mit Proxy versehen	294
10.6	Zusammenfassung	296
11	Spring und REST	297
11.1	Representational State Transfer (REST)	298
11.1.1	Die Grundlagen von REST	298
11.1.2	Wie Spring REST unterstützt	299
11.2	Ressourcenorientierte Controller schreiben	299
11.2.1	Analyse eines nicht-REST-konformen Controllers	300
11.2.2	REST-konforme URLs bearbeiten	301
11.2.3	Die REST-Verben ausführen	304
11.3	Ressourcen repräsentieren	308
11.3.1	Die Repräsentation der Ressource verhandeln	308
11.3.2	Die Arbeit mit HTTP-Nachrichtenkonvertierern	311
11.4	REST-Clients schreiben	315
11.4.1	Die Operationen von RestTemplate	316
11.4.2	Ressourcen mit GET bekommen	317
11.4.3	Ressourcen mit PUT beziehen	320
11.4.4	Ressourcen mit DELETE löschen	322
11.4.5	Ressourcendaten mit POST zum Server bringen	323
11.4.6	Ressourcen austauschen	325
11.5	REST-konforme Formulare übermitteln	327

11.5.1	Versteckte Methodenfelder in JSP darstellen	328
11.5.2	Den echten Request demaskieren	329
11.6	Zusammenfassung	330
12	Messaging in Spring	333
12.1	Kurze Einführung in JMS	334
12.1.1	Die Architektur von JMS	335
12.1.2	Die Vorteile von JMS	337
12.2	Einen Message-Broker in Spring einrichten	339
12.2.1	Erstellen einer <code>ConnectionFactory</code>	339
12.2.2	Deklaration eines Nachrichtenendpunkts für ActiveMQ	340
12.3	Das JMS-Template von Spring	341
12.3.1	Umgang mit unübersichtlichem JMS-Code	342
12.3.2	Die Arbeit mit JMS-Templates	343
12.4	Erstellung nachrichtengetriebener POJOs	349
12.4.1	Erstellen eines Message-Listeners	350
12.4.2	Message-Listener konfigurieren	351
12.5	Die Arbeit mit nachrichtenbasierten RPCs	352
12.5.1	Die Arbeit mit nachrichtengetriebenen RPC in Spring	352
12.5.2	Asynchrones RPC mit Lingo	355
12.6	Zusammenfassung	357
13	Verwalten von Spring-Beans mit JMX	359
13.1	Spring-Beans als MBeans exportieren	360
13.1.1	Methoden über den Namen bereitstellen	363
13.1.2	Über Interfaces, die Operationen und Attribute von MBeans definieren	365
13.1.3	Die Arbeit mit von Annotierungen gesteuerten MBeans	366
13.1.4	Der Umgang mit MBean-Kollisionen	367
13.2	Remoting von MBeans	369
13.2.1	Exportieren von Remote-MBeans	369
13.2.2	Zugriff auf Remote-MBeans	370
13.2.3	Proxy für MBeans	371
13.3	Der Umgang mit Benachrichtigungen	373
13.3.1	Auf Benachrichtigungen lauschen	374
13.4	Zusammenfassung	375
14	Krimskrams	377
14.1	Die Konfiguration auslagern	377
14.1.1	Property-Placeholder ersetzen	378
14.1.2	Eigenschaften überschreiben	381
14.1.3	Externe Eigenschaften verschlüsseln	382
14.2	JNDI-Objekte verschalten	384
14.2.1	Die Arbeit mit konventionellem JNDI	385
14.2.2	Injektion von JNDI-Objekten	386
14.2.3	Verschaltung von EJBs in Spring	390
14.3	Der E-Mail-Versand	391

14.3.1	Konfiguration eines Mail-Senders	391
14.3.2	Konstruktion der E-Mail	393
14.4	Scheduling und Hintergrundaufgaben	399
14.4.1	Geplante Methoden deklarieren	399
14.4.2	Asynchrone Methoden deklarieren	401
14.5	Zusammenfassung	402
14.6	Das Ende ...?	403
	Register	405