

Eberhard Wolff

Spring 2

Framework für die Java-Entwicklung

2., aktualisierte und erweiterte Auflage



dpunkt.verlag

Inhaltsverzeichnis

1	Einleitung	1
1.1	Warum Spring?	1
1.2	Was ist Spring?	1
1.3	Spring und Java EE	3
1.4	Woher kommt Spring?	5
1.5	Warum dieses Buch?	5
1.6	Patterns und dieses Buch	6
1.7	Wie man das Buch lesen sollte	7
1.8	Mit Spring entwickeln	8
1.9	Danksagung	9
2	Objekte finden zueinander: Dependency Injection	11
2.1	Übersicht	11
2.2	Die Beispielanwendung	11
2.2.1	Das fachliche Modell	12
2.2.2	Geschäftsprozesse in der Beispielanwendung	13
2.2.3	Benutzeroberfläche	14
2.3	Objektnetze in OO-Systemen	14
2.3.1	Netze weben	15
2.4	Der neue Ansatz: Dependency Injection	20
2.5	Dependency Injection mit Spring	22
2.5.1	Die Konfigurationsdatei	22
2.5.2	Die BeanFactory	26
2.5.3	Constructor Dependency Injection	27
2.5.4	Erzeugung mit Factories	28

2.6	Vorteile von DI	32
2.6.1	Wir rufen Sie an	32
2.6.2	Flexibilität	32
2.6.3	Eingebaute Konfigurierbarkeit	33
2.6.4	Das Singleton-Pattern	34
2.6.5	Das Factory-Pattern	36
2.7	Die Grenzen von Dependency Injection	37
2.8	Wir kennen uns: Autowiring	40
2.9	Die bessere BeanFactory: Der ApplicationContext	43
2.9.1	Zugriff auf Ressourcen	44
2.9.2	Events	46
2.9.3	Internationalisierung	48
2.9.4	Objekte nachbearbeiten	50
2.10	Fortgeschrittene Techniken	52
2.10.1	Spring-Konfiguration ohne XML	52
2.10.2	Komplexe Datentypen als Werte setzen	55
2.10.3	Wie Properties genau konfiguriert werden	59
2.10.4	Lebenszyklus	62
2.10.5	Fachliche Klassen mit @Configurable konfigurieren	65
2.10.6	Konfigurationen handhaben	67
2.10.7	Dynamische Sprachen mit Spring	73
2.11	Tests mit Spring	74
2.11.1	Spring und Unit Tests	75
2.11.2	Integrationstests	79
2.11.3	Funktionale Tests	83
2.11.4	Systemtests	85
2.12	Fazit	86
3	Aspektorientierte Programmierung mit Spring	87
3.1	Übersicht	87
3.2	Was sind Aspekte?	88
3.3	Aspekte mit Spring: Grundlagen	89
3.4	Spring-AOP-eigene Syntax	96
3.4.1	Pointcuts in Spring	96
3.4.2	Aspekte-Werkzeugkasten	100
3.4.3	Metadaten und Annotationen	101
3.4.4	Jakarta Commons Attributes	101
3.4.5	JDK-1.5-Annotationen	104
3.4.6	Introductions: Interface nachträglich implementieren	105

3.5	AspectJ-Syntax mit Spring AOP	106
3.5.1	Verwendung klassischer Spring-Aspekte mit AspectJ-Pointcuts	107
3.5.2	Aspekte mit Annotationen markieren	109
3.5.3	Mehr über AspectJ Pointcuts	115
3.5.4	Interfaces nachträglich implementieren mit AspectJ-Syntax	117
3.6	Fortgeschrittene Techniken mit Spring-Aspekten	119
3.6.1	Reihenfolge der Advices	119
3.6.2	Das Advised-Interface und Zugriff auf den Proxy	120
3.6.3	Zielobjekte austauschen	120
3.6.4	Methoden ersetzen	125
3.7	Aspekte in der Beispielanwendung	126
3.8	AspectJ	131
3.8.1	Migration von Spring AOP nach AspectJ	131
3.8.2	AspectJ-Aspekte mit Spring konfigurieren	132
3.9	Fazit	133
4	Transaktionen	135
4.1	Übersicht	135
4.2	Der Transaktionsmanager	136
4.2.1	Was ist eine Transaktion?	136
4.2.2	Die Pattern-Form	139
4.2.3	Das Exception-Übersetzer-Pattern	139
4.2.4	Die TransactionException-Hierarchie	141
4.3	Transaktionen selber managen	142
4.4	Transaktionen mit Templates verwalten	144
4.4.1	Das Template-Pattern	145
4.4.2	Das Transaction-Template	146
4.5	Deklaratives Transaktionsmanagement	148
4.6	Transaktionen mit Spring in der Praxis	156
4.7	Fazit	158
5	Persistenz	159
5.1	Übersicht	159
5.2	Das DAO-Pattern	160

5.3	JDBC-Helferklassen	163
5.3.1	Die JdbcTemplate-Klasse	163
5.3.2	NamedParameterJdbcTemplate	169
5.3.3	SimpleJdbcTemplate	170
5.3.4	Anfragen als Objekte	172
5.4	Datenbank-Exceptions	176
5.5	iBATIS-Unterstützung	177
5.6	Hibernate-Integration	180
5.7	JPA-Integration	186
5.8	Andere Technologien	192
5.8.1	JDO	193
5.8.2	Oracle TopLink	194
5.9	Fazit	194
6	Spring ins Netz	197
6.1	Übersicht	197
6.2	RMI	198
6.2.1	Das Exporter-Pattern	199
6.2.2	Der RmiServiceExporter	200
6.2.3	RMI-Objekte verwenden	201
6.2.4	Das Proxy-Pattern	201
6.2.5	Die RmiFactoryBean	202
6.3	HTTP-basierte Protokolle	204
6.4	SOAP	207
6.4.1	SOAP mit Axis	208
6.4.2	SOAP mit XFire	211
6.4.3	SOAP mit Spring Web Services	213
6.5	EJB	220
6.5.1	Was ist EJB?	220
6.5.2	EJB 2.1 Beans mit Spring	221
6.5.3	EJB 3	228
6.6	Infrastrukturen: EJB, Spring,	231
6.7	Fazit	234

7	Das Spring Web Framework	235
7.1	Übersicht	235
7.2	Was ist MVC 2?	236
7.3	Controller	237
7.3.1	Der AbstractController	237
7.3.2	MultiActionController: Mehrere Controller in einem	239
7.3.3	SimpleFormController für Formulare	242
7.3.4	AbstractWizardFormController	244
7.3.5	ExceptionHandler	246
7.4	Validierung	247
7.5	JSPs mit der Spring Tag Library	250
7.6	Alternative View-Technologien	256
7.6.1	XSLT-Views mit Spring	256
7.6.2	Excel-Views	259
7.6.3	PDF-Views	261
7.6.4	Weitere View-Technologien	263
7.7	Welche View wird genutzt?	264
7.7.1	InternalResourceViewResolver	264
7.7.2	ResourceBundleViewResolver	265
7.7.3	XmlViewResolver	266
7.7.4	Mehrere ViewResolver nutzen	267
7.8	Handler Mapping	268
7.9	Der Weg eines Requests durch Spring MVC	271
7.10	Spring und andere Webtechnologien	272
7.10.1	Spring und Struts	272
7.10.2	JavaServer Faces (JSF)	276
7.10.3	Andere Technologien	278
7.11	Scopes für Webanwendungen	279
7.12	Spring Web Flow	281
7.13	Fazit	289