

HANSER

Inhaltsverzeichnis

Donald Brown, Chad Michael Davis, Scott Stanlick

Struts 2 im Einsatz

Übersetzt aus dem Englischen von Jürgen Dubau

ISBN: 978-3-446-41575-1

Weitere Informationen oder Bestellungen unter

<http://www.hanser.de/978-3-446-41575-1>

sowie im Buchhandel.



# Inhalt

<b>Vorwort</b> .....	<b>1</b>
<b>Danksagungen</b> .....	<b>1</b>
<b>Über dieses Buch</b> .....	<b>1</b>
<b>Teil 1: Struts 2 – Ein brandneues Framework</b> .....	<b>1</b>
<b>1 Struts 2 – Das moderne Framework für Webanwendungen</b> .....	<b>3</b>
1.1 Webanwendungen – Ein kurzer Überblick .....	4
1.1.1 Das Internet zum Erstellen von Anwendungen einsetzen .....	4
1.1.2 Die technologische Basis .....	4
1.1.3 Inspektion des Arbeitsbereichs .....	9
1.2 Frameworks für Webanwendungen .....	11
1.2.1 Was ist ein Framework? .....	11
1.2.2 Warum man mit einem Framework arbeiten sollte .....	12
1.3 Das Framework von Struts 2 .....	13
1.3.1 Ein kurzer historischer Abriss .....	13
1.3.2 Struts 2 aus 10.000 Metern Höhe: Das MVC-Muster .....	14
1.3.3 Wie Struts 2 arbeitet .....	17
1.4 Zusammenfassung .....	21
<b>2 Struts 2 – Der erste Kontakt</b> .....	<b>23</b>
2.1 Deklarative Architektur .....	24
2.1.1 Zwei Arten der Konfiguration .....	24
2.1.2 Zwei Mechanismen zur Deklaration der Architektur.....	25
2.1.3 Intelligente Defaults.....	28
2.2 Hello auf die Schnelle .....	29
2.2.1 Deployment der Beispielanwendung .....	29
2.2.2 Die HelloWorld-Anwendung .....	34
2.3 HelloWorld mit Annotationen .....	41
2.4 Zusammenfassung .....	44

<b>Teil 2: Zentrale Konzepte: Actions, Interceptors und Typkonvertierungen .....</b>	<b>45</b>
<b>3 Die Arbeit mit Actions .....</b>	<b>47</b>
3.1 Die Actions in Struts 2 .....	48
3.1.1 Was macht eine Action? .....	48
3.2 Verpacken der Actions .....	50
3.2.1 Das Struts 2 Portfolio .....	51
3.2.2 Die Organisation der Pakete .....	51
3.2.3 Die Arbeit mit den Komponenten des struts-default-Pakets .....	54
3.3 Implementierung von Actions .....	56
3.3.1 Das optionale Action-Interface .....	57
3.3.2 Die ActionSupport-Klasse .....	58
3.4 Daten auf Objekte transferieren .....	66
3.4.1 Objekt-gepufferte JavaBeans-Properties .....	67
3.4.2 ModelDriven-Actions .....	69
3.4.3 Domänenobjekte für Datentransfer – Eine Abschlussbemerkung .....	71
3.5 Dateiupload – Eine Fallstudie .....	72
3.5.1 Eingebauter Support über das Paket struts-default .....	73
3.5.2 Was macht der fileUpload-Interceptor? .....	73
3.5.3 Der Beispielcode von Struts 2 Portfolio .....	74
3.6 Zusammenfassung .....	77
<b>4 Den Workflow mit Interceptors ergänzen .....</b>	<b>79</b>
4.1 Abfangen von Anfragen .....	80
4.1.1 Das MVC aufräumen .....	80
4.1.2 Die Vorteile einfahren .....	82
4.1.3 Erstellen von Interceptors .....	83
4.2 Interceptors im Einsatz .....	84
4.2.1 Die Chefin: ActionInvocation .....	84
4.2.2 Wie die Interceptors feuern .....	84
4.3 Die in Struts 2 integrierten Interceptors .....	87
4.3.1 Utility-Interceptors .....	88
4.3.2 Interceptors für den Datentransfer .....	88
4.3.3 Workflow-Interceptor .....	90
4.3.4 Weitere Interceptors .....	95
4.3.5 Integrierte Stacks .....	97
4.4 Deklaration von Interceptors .....	97
4.4.1 Deklaration einzelner Interceptors und Interceptor-Stacks .....	98
4.4.2 Interceptors auf Actions mappen .....	100
4.4.3 Setzen und Überschreiben von Parametern .....	101
4.5 Erstellen eigener Interceptors .....	102
4.5.1 Implementierung des Interceptor-Interface .....	102
4.5.2 Der AuthenticationInterceptor .....	103
4.6 Zusammenfassung .....	107

<b>5</b>	<b>Datentransfer – OGNL und Typkonvertierung .....</b>	<b>109</b>
5.1	Datentransfer und Typkonvertierung: Übliche Aufgaben bei Webanwendungen .....	110
5.2	OGNL und Struts 2.....	111
5.2.1	Die Aufgabe von OGNL.....	111
5.2.2	Wie passt OGNL in das Framework? .....	113
5.3	Integrierte Typkonverter.....	116
5.3.1	Konvertierungen <i>out of the box</i> .....	117
5.3.2	Formularfeldnamen mit OGNL-Ausdrücken auf Properties mappen .....	118
5.4	Eigene Typkonvertierungen .....	131
5.4.1	Implementierung eines Typkonverters.....	132
5.4.2	Konvertierung zwischen Strings und Circles.....	133
5.4.3	Konfiguration des Frameworks zur Arbeit mit dem Konverter.....	134
5.5	Zusammenfassung .....	136
<b>Teil 3: Die View erstellen – Tags und Results .....</b>		<b>139</b>
<b>6</b>	<b>Die View erstellen – Tags.....</b>	<b>141</b>
6.1	Auf die Plätze .....	142
6.1.1	Der ActionContext und OGNL.....	142
6.1.2	Der ValueStack – Ein virtuelles Objekt.....	146
6.2	Die Tags von Struts im Überblick .....	148
6.2.1	Die Syntax der Tag-API von Struts 2 .....	148
6.2.2	Attribute für Tags mit OGNL setzen .....	150
6.3	Daten-Tags .....	153
6.3.1	Das property-Tag.....	154
6.3.2	Das set-Tag.....	154
6.3.3	Das push-Tag.....	156
6.3.4	Das bean-Tag.....	157
6.3.5	Das action-Tag.....	160
6.4	Control-Tags.....	162
6.4.1	Das iterator-Tag.....	162
6.4.2	Die Tags if und else .....	164
6.5	Verschiedene andere Tags .....	164
6.5.1	Das include-Tag.....	165
6.5.2	Das URL-Tag .....	166
6.5.3	Die Tags <code>is</code> und <code>text</code> .....	167
6.5.4	Das param-Tag.....	169
6.6	Die Verwendung von JSTL und anderen nativen Tags .....	170
6.7	Kurze Einführung in die Ausdruckssprache OGNL .....	170
6.7.1	Was ist OGNL?.....	170
6.7.2	Häufige Features einer Ausdruckssprache in Struts 2.....	171
6.7.3	Fortgeschrittene Features der Ausdruckssprache .....	177
6.8	Zusammenfassung .....	180
<b>7</b>	<b>UI-Komponenten-Tags .....</b>	<b>183</b>
7.1	Warum wir UI-Komponenten-Tags brauchen .....	184
7.1.1	Mehr als nur Formularelemente.....	184

7.2	Tags, Templates und Themes .....	190
7.2.1	Tags .....	192
7.2.2	Templates .....	192
7.2.3	Themes .....	193
7.3	Referenz der Tags der UI-Komponenten.....	195
7.3.1	Allgemeine Attribute .....	195
7.3.2	Einfache Komponenten .....	197
7.3.3	Von Collections gepufferte Komponenten .....	209
7.3.4	Bonus-Komponenten .....	218
7.4	Zusammenfassung .....	220
<b>8</b>	<b>Results unter der Lupe.....</b>	<b>223</b>
8.1	Leben nach der Action .....	224
8.1.1	Jenseits der Seite – Wie man mit eigenen Results Ajax-Anwendungen in Struts 2 erstellt.....	225
8.1.2	Implementierung eines JSON-Result-Typs .....	226
8.2	Häufig verwendete Result-Typen .....	235
8.2.1	Der RequestDispatcher alias Dispatcher.....	236
8.2.2	Das ServletRedirectResult alias redirect.....	242
8.2.3	Das ServletActionRedirectResult alias redirectAction .....	245
8.3	JSP-Alternativen .....	247
8.3.1	VelocityResult alias velocity .....	247
8.3.2	FreemarkerResult alias freemarker .....	249
8.4	Globale Results .....	251
8.5	Zusammenfassung .....	252
	<b>Teil 4: Optimieren der Anwendung.....</b>	<b>253</b>
<b>9</b>	<b>Integration mit Spring und Hibernate/JPA .....</b>	<b>255</b>
9.1	Warum bei Struts 2 mit Spring arbeiten? .....	256
9.1.1	Wofür ist Abhängigkeitsinjektion gut?.....	256
9.1.2	Wie Spring Objekte verwaltet und Abhängigkeiten injiziert .....	259
9.1.3	Mit Interfaces Implementierungen verstecken .....	260
9.2	Spring in Struts 2 einfügen.....	263
9.2.1	Spring verwaltet die Erstellung von Actions, Interceptors und Results .....	264
9.2.2	Autowiring zur Injektion von Abhängigkeiten in Actions, Interceptors und Results .....	266
9.3	Warum soll man bei Struts 2 die Java Persistence API nutzen?.....	270
9.3.1	Das Projekt für JPA mit Hibernate einrichten .....	270
9.3.2	Programmieren einer von Spring verwalteten JPA .....	275
9.4	Zusammenfassung .....	279
<b>10</b>	<b>Das Validierungsframework .....</b>	<b>281</b>
10.1	Das Validierungsframework.....	282
10.1.1	Die Architektur des Validierungsframeworks .....	282
10.1.2	Das Validierungsframework im Workflow von Struts 2 .....	284
10.2	Actions zur Validierung verschalten .....	287

10.2.1	Deklaration der Validierungsmetadaten mit ActionClass-validations.xml .....	288
10.2.2	Die integrierten Validatoren .....	292
10.3	Eigene Validatoren schreiben .....	294
10.3.1	Mit eigenem Validator die Passwortstärke prüfen .....	294
10.3.2	Die Arbeit mit dem eigenen Validator .....	297
10.4	Fortgeschrittene Themen des Validierungsframeworks .....	298
10.4.1	Validierung auf der Ebene des Domänenobjekts .....	298
10.4.2	Mit Validierungskontext die Validierungen verfeinern .....	302
10.4.3	Vererbung der Validierung .....	305
10.4.4	Vorzeitiges Beenden von Validierungen .....	305
10.4.5	Validierungen mit Annotationen deklarieren .....	306
10.5	Zusammenfassung .....	308
<b>11</b>	<b>Die Internationalisierung.....</b>	<b>311</b>
11.1	Java-i18n und das Framework von Struts 2 .....	312
11.1.1	Auslesen von lokalisiertem Text mit ResourceBundle und Locale .....	313
11.1.2	Wie Struts 2 die Arbeit mit i18n erleichtert .....	316
11.2	Eine i18n-Demo für Struts 2 .....	317
11.2.1	Eine kurze Demo für Struts 2 und i18n .....	317
11.2.2	Ein kurzer Blick hinter die Kulissen .....	319
11.3	i18n bei Struts 2 – Die Details .....	321
11.3.1	Der Default-Lokalisierungsalgorithmus von Struts 2 .....	321
11.3.2	Nachrichtentexte aus den Bundles auslesen .....	326
11.3.3	Ein Bundle über das i18n-Tag angeben .....	330
11.3.4	Parametrisierung der lokalisierten Texte .....	330
11.3.5	Formatierung von Datumsangaben und Zahlen .....	332
11.4	Überschreiben der Default-Erkennung des Gebietschemas .....	333
11.4.1	Interaktives Setzen von Sprach- und Ländereinstellungen .....	334
11.4.2	Sprach- und Ländereinstellungen programmatisch setzen .....	336
11.5	Zusammenfassung .....	337
<b>12</b>	<b>Struts 2 mit Plug-ins erweitern .....</b>	<b>341</b>
12.1	Plug-ins – Ein Überblick .....	342
12.1.1	Wie man Plug-ins findet .....	343
12.2	Bekannte Plug-ins .....	343
12.2.1	SiteMesh .....	344
12.2.2	Tiles .....	345
12.2.3	JFreeChart .....	347
12.3	Das interne Komponentensystem .....	349
12.3.1	Beans .....	349
12.3.2	Konstanten .....	350
12.3.3	Injektion .....	350
12.3.4	Interne Erweiterungspunkte von Struts .....	351
12.4	Ein Breadcrumb-Plug-in schreiben .....	353
12.5	Zusammenfassung .....	357

<b>13</b>	<b>Best Practices.....</b>	<b>359</b>
13.1	Die Umgebung einrichten .....	360
13.1.1	Die IDE einrichten.....	360
13.1.2	Ressourcen erneut laden .....	361
13.2	Unit-Tests für Ihre Actions.....	361
13.2.1	Der Vorteil von IoC für Tests .....	362
13.2.2	JUnit und die Tests .....	363
13.2.3	Tests von validation.xml-Dateien .....	365
13.3	Die Wiederverwendbarkeit maximieren.....	366
13.3.1	Komponenten mit dem component-Tag erstellen .....	367
13.3.2	Arbeit mit den Template-Tags.....	368
13.3.3	Verbindung der Punkte von der UI zu den Objekten.....	369
13.4	Fortgeschrittene Verwendung der UI-Tags .....	370
13.4.1	Überschreiben vorhandener Templates.....	370
13.4.2	Eigene Templates schreiben .....	371
13.4.3	Eigene Themes schreiben .....	371
13.5	Zusammenfassung.....	372
<b>14</b>	<b>Migration von Struts Classic .....</b>	<b>373</b>
14.1	Der Wissenstransfer aus Struts Classic .....	373
14.1.1	Actions .....	374
14.1.2	Wo sind die ActionForms geblieben?.....	375
14.1.3	Austausch der Tag-Libraries.....	377
14.1.4	Verteilen der Nachrichtenressourcen.....	379
14.2	Schrittweise Konvertierung.....	381
14.2.1	Rom schrittweise erobern .....	382
14.2.2	Die Action-Mappings .....	383
14.2.3	Wo die Action das Formular trifft .....	385
14.2.4	Die Seite wandeln.....	386
14.2.5	No speak English.....	389
14.2.6	Die Datenpolizei.....	390
14.2.7	Könnten wir nicht einfach so klarkommen? .....	392
14.3	Zusammenfassung.....	394
<b>15</b>	<b>Themen für Fortgeschrittene.....</b>	<b>395</b>
15.1	Fortgeschrittene Verwendung von Actions .....	395
15.1.1	Alternative Methodenaufrufe.....	396
15.2	Dynamische Methodenaufrufe .....	397
15.2.1	Die Methodenauswahl per Wildcard .....	397
15.2.2	Dynamische Workflows .....	400
15.3	Mit Tokens doppelte Formularübermittlungen verhindern.....	401
15.3.1	Das <s:token/>-Formular-Tag .....	401
15.3.2	Ausnahmen zur token-Interceptor-Regel.....	403
15.4	Seiten mit Fortschrittsanzeige automatisch darstellen .....	405
15.4.1	User am Rande des Nervenzusammenbruchs .....	405
15.5	Eine einzige Action für CRUD-Operationen.....	406

15.5.1	Dieses CRUD .....	407
15.5.2	Interceptors und Interfaces.....	408
15.5.3	Die Teile verbinden .....	413
15.6	Tiles und Struts 2.....	415
15.6.1	Look & Feel der Website.....	416
15.6.2	Konfiguration des Zusammenspiels.....	417
15.6.3	Die Arbeit mit der deklarativen Architektur .....	419
15.6.4	Vorbereiten der Inhalte einer Webseite mit einem Tiles-Controller .....	421
15.7	Zusammenfassung .....	423
	<b>Register.....</b>	<b>425</b>