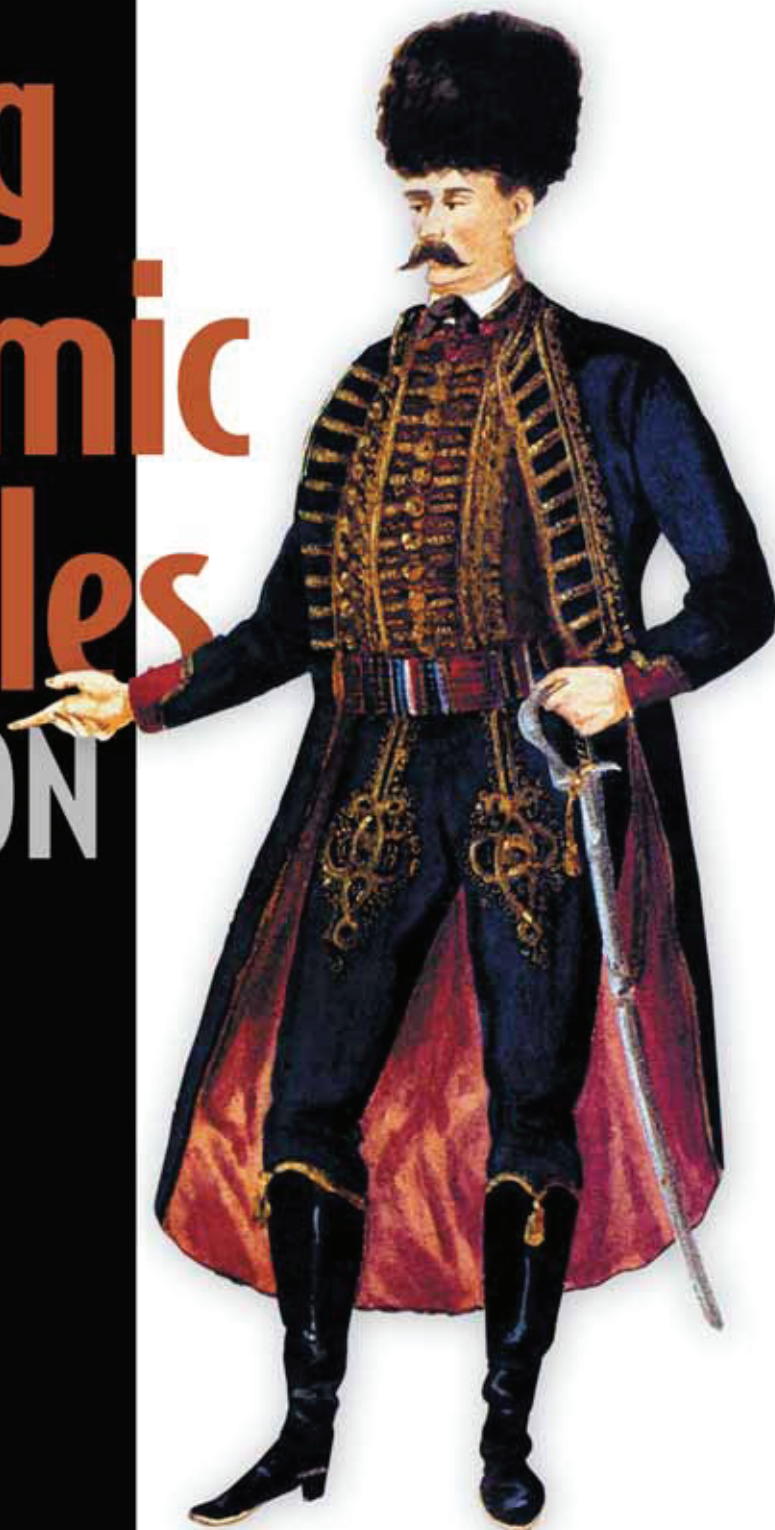


Spring Dynamic Modules IN ACTION

Arnaud Cogoluègnes
Thierry Templier
Andy Piper

FOREWORD BY PETER KRIENS

 MANNING



contents

<i>foreword</i>	<i>xv</i>
<i>preface</i>	<i>xvii</i>
<i>acknowledgments</i>	<i>xix</i>
<i>about this book</i>	<i>xxi</i>
<i>about the authors</i>	<i>xxv</i>
<i>about the title</i>	<i>xxvi</i>
<i>about the cover illustration</i>	<i>xxvii</i>

PART 1 SPRING DM BASICS1

1 *Modular development with Spring and OSGi* 3

1.1 Java modularity 4

What is modularity and what is it good for? 4 ▪ *Java—the end and the beginning* 4 ▪ *Are your applications really modular?* 5

1.2 The Spring Framework 5

Loose coupling of classes 6 ▪ *Dependency injection* 7 ▪ *Aspect-oriented programming (AOP)* 8 ▪ *Enterprise support* 9

1.3 A new approach to modular development with OSGi 10

Aims of OSGi 10 ▪ *OSGi layers* 11 ▪ *What OSGi offers* 12

- 1.4 Using Spring in an OSGi environment with Spring DM 13
 - What is Spring DM?* 13
 - *Embedding Spring within an OSGi container* 15
 - *Benefits of Spring DM for real-life OSGi applications* 17
- 1.5 Spring DM Hello World 18
 - Provisioning the OSGi container* 18
 - *Writing the Spring DM-powered bundle* 21
 - *Deploying the bundle* 22
- 1.6 Summary 23

2 *Understanding OSGi technology* 24

- 2.1 OSGi components 25
 - Component structure* 25
 - *Component configuration* 26
 - OSGi containers* 28
 - *OSGi component lifecycle* 29
- 2.2 Component dependencies 30
 - Classloader isolation and chaining* 31
 - *Providing dependencies* 33
 - Consuming dependencies with the Import-Package header* 34
 - Consuming dependencies with the Require-Bundle header* 36
 - Matching and versioning* 38
- 2.3 Interacting with the OSGi container 42
 - Bundles* 43
 - *Lifecycle management* 44
 - *Properties* 46
 - *Event support* 46
 - *Persistent storage area* 48
 - *Bundle activator* 48
- 2.4 Service support in OSGi 50
 - Providing services* 51
 - *Using services* 54
 - *Service event support* 56
- 2.5 Handling native code 58
- 2.6 Diagnosing errors 59
 - Detecting components with problems* 60
 - *Detecting different kinds of problems* 61
- 2.7 Summary 62

3 *Getting started with Spring DM* 63

- 3.1 Using Spring in OSGi components 64
 - Embedding the Spring application context within components* 64
 - Spring DM's OSGi-aware application context* 66
 - *Spring DM's extender mechanisms* 67
 - *Kinds of supported bundles* 69
 - Spring DM's osgi namespace* 70
- 3.2 Installing Spring DM 72
 - Configuring a container* 72
 - *Provisioning a container for simple use* 73
 - *Provisioning a container for web use* 76

- 3.3 Using a fragment to configure the LOG4J bundle 80
 - Using the fragment configuration pattern* 80
 - *Implementing a fragment* 81
 - *Installing a fragment into the OSGi container* 82
- 3.4 Developing Spring DM bundles 82
 - Creating and configuring a bundle* 82
 - *Packaging a bundle* 85
 - Checking a bundle within a container* 86
 - *Developing an integration test* 87
- 3.5 Developing Spring DM web bundles 94
 - Creating and configuring a web bundle* 94
 - *Packaging a web bundle* 95
 - *Checking the operation of a web bundle in a container* 97
 - Developing an integration test for a web bundle* 97
- 3.6 Summary 100

PART 2 CORE SPRING DM101

4 *Using Spring DM extenders* 103

- 4.1 Unleashing Spring DM's standard extender 104
 - A word about dependencies* 104
 - *Structure of standard Spring OSGi components* 106
 - *Initializing and destroying the Spring container* 110
 - *Customizing application context creation* 116
 - Listening to extender events with the whiteboard pattern* 118
 - Hard dependencies on the OSGi environment* 123
- 4.2 Unleashing Spring DM's web extender 126
 - Structure of Spring DM web OSGi components* 126
 - *Classloading in web bundles* 127
 - *OSGi-aware Spring web container* 129
 - Spring DM web deployer* 131
- 4.3 Summary 132

5 *Working with OSGi services* 133

- 5.1 Dependency injection and OSGi services 134
 - Combining OSGi services and dependency injection* 134
 - *XML-based registration and referencing* 136
 - *Annotation-based service referral* 139
- 5.2 The thread context classloader and its use in OSGi 141
 - Using the thread context classloader* 141
 - *Using the thread context classloader with OSGi* 142
- 5.3 Advanced OSGi service configuration 144
 - Configuration for registering services* 144
 - *Configuration for referencing services* 148

- 5.4 Handling OSGi service dynamics 151
 - Service registration and unregistration events* 152
 - Service bind and unbind events* 153
- 5.5 Handling collections of OSGi services 156
 - Configuring collections* 156
 - Sorting collections* 157
 - Greedy proxying* 159
 - Integrated support when collections are updated* 160
- 5.6 Programmatic service support 160
 - Registering a service programmatically* 161
 - Referencing services programmatically* 162
 - ServiceReference support* 162
- 5.7 Summary 163

6 *OSGi and Spring DM for enterprise applications* 164

- 6.1 Building an OSGi repository for enterprise applications 165
 - Using Java and Java EE frameworks in OSGi environments* 165
 - Choosing the right frameworks for OSGi* 166
 - Getting OSGi-ready artifacts* 168
- 6.2 OSGi-ifying libraries and frameworks 169
 - How to create OSGi-ified versions of libraries* 169
 - Converting by hand* 172
 - Converting using tools* 173
 - Packaging your own modules as OSGi bundles* 180
- 6.3 Designing OSGi enterprise applications 180
 - Organizing OSGi components* 181
 - Defining interactions between application bundles* 187
- 6.4 How Spring DM handles OSGi applications' dynamic behavior 189
 - Dealing with the appearance or disappearance of services* 189
 - Providing a new version of a component* 195
- 6.5 Summary 198

7 *Data access in OSGi with Spring DM* 199

- 7.1 Using JDBC within OSGi with Spring DM 200
 - JDBC concepts* 200
 - JDBC issues when used within OSGi* 202
 - Configuring JDBC data sources* 203
 - Provisioning the OSGi container for JDBC* 207
 - Using JDBC within OSGi with Spring DM* 207
- 7.2 Using ORM within OSGi with Spring DM 209
 - Object/relational mapping* 210
 - Load-time weaving* 215
 - Provisioning a container for JPA implementations* 218
 - Using JPA in OSGi with Spring DM* 220
 - JPA implementation specifics when used with Spring DM* 223
 - A JPA summary* 226

- 7.3 Transactions 227
 - Spring's transactional support* 227 ▪ *Using JPA transactions with Spring DM* 229
- 7.4 Using the open EntityManager in view pattern 232
 - The open EntityManager in view pattern* 232 ▪ *Using the open EntityManager in view pattern with Spring DM* 233
- 7.5 Summary 235

8 *Developing OSGi web components with Spring DM and web frameworks* 236

- 8.1 Using action-based web frameworks with Spring DM 237
 - Using Spring DM with action-based frameworks* 237 ▪ *Using Spring MVC with Spring DM* 238
- 8.2 Using component-based web frameworks with Spring DM 244
 - Using Spring DM with component-based frameworks* 244 ▪ *Using JSF with Spring DM* 245 ▪ *Using Wicket with Spring DM* 251
- 8.3 Using AJAX frameworks with Spring DM 256
 - Using Spring DM with AJAX frameworks* 257 ▪ *Using DWR with Spring DM* 257 ▪ *Using GWT with Spring DM* 262
- 8.4 Using web services with Spring DM 267
 - Using Spring DM with web service frameworks* 267 ▪ *Using Spring WS with Spring DM* 268 ▪ *Using Restlet with Spring DM* 273
- 8.5 Summary 279

PART 3 ADVANCED TOPICS281

9 *Advanced concepts* 283

- 9.1 Configuring Spring DM core components 284
 - Fragment-based configuration* 285 ▪ *Features configurable through named beans* 285
- 9.2 Extending the standard extender 286
 - Beans usable for configuration* 287 ▪ *Task executor for creating application contexts* 288 ▪ *Task executor for destroying application contexts* 290 ▪ *Extender properties* 291 ▪ *Propagating application context events* 292 ▪ *Overriding application context creation* 293 ▪ *Adding postprocessing to application contexts* 297 ▪ *Overriding the default OSGi application context listener* 298 ▪ *How SpringSource dm Server customizes Spring DM's extender* 298

- 9.3 Extending the web extender and WAR deployer 300
 - Beans available for configuration* 300
 - *Overriding the WAR deployer* 301
 - *Overriding the context path strategy* 301
 - Overriding the WAR scanner* 302
- 9.4 Configuring embedded web containers 303
 - The basics of Spring DM's web support* 303
 - *Support for Tomcat* 305
 - *Support for Jetty* 308
- 9.5 Support for Java 2 security 311
 - The Java security model* 311
 - *The OSGi security model* 313
 - Integrating Spring DM into the OSGi security model* 315
- 9.6 Advanced patterns 316
 - Implementation provider pattern* 316
 - *Chained classloader pattern for proxy-based AOP* 319
- 9.7 Summary 321

10 *Testing with Spring DM* 323

- 10.1 Testing OSGi components with Spring DM 324
 - General concepts* 324
 - *Unit tests with Spring-based applications* 325
 - *Testing OSGi components* 327
- 10.2 Strict unit tests for OSGi components 330
 - Spring DM's OSGi mocks* 330
 - *Spring DM's OSGi mocks in action* 331
- 10.3 Integration tests for OSGi applications 335
 - Developing integration tests with Spring DM support* 335
 - Advanced features of Spring DM test support* 344
- 10.4 Summary 350

11 *Support for OSGi compendium services* 351

- 11.1 Overview of compendium services 352
 - What are compendium services?* 352
 - *Spring DM's support for compendium services* 353
- 11.2 Spring DM's Configuration Admin Service support 355
 - OSGi Configuration Admin Service* 356
 - *Using properties defined by the Configuration Admin Service* 358
 - *Support of managed entities* 360

- 11.3 Spring DM's Event Admin Service support 365
 - OSGi Event Admin Service* 365 ▪ *Linking Spring DM and the OSGi Event Admin Service* 367 ▪ *Implementing the bridge between Spring DM and Event Admin Service* 369 ▪ *Implementing OSGi event handlers* 371
- 11.4 Summary 372

12 *The Blueprint specification* 374

- 12.1 Standardization of Spring DM 375
 - The attraction of open standards* 375 ▪ *Standards development* 376 ▪ *Goals of the Blueprint specification* 376 ▪ *Scope of the specification* 377
 - 12.2 A taxonomy of Blueprint 377
 - A Blueprint example* 378 ▪ *Blueprint bundles* 379
 - 12.3 Blueprint manager syntax 379
 - Bean manager* 379 ▪ *Service manager* 381 ▪ *Reference manager* 382 ▪ *Reference-list manager* 383 ▪ *A Blueprint example* 384 ▪ *The Blueprint container and its metadata* 385 ▪ *Environment managers* 387
 - 12.4 Runtime support and lifecycle 388
 - Blueprint lifecycle* 388 ▪ *Type converters* 392
 - 12.5 Using Spring DM with Blueprint 394
 - 12.6 Summary 394
- appendix A Spring DM development with Eclipse* 397
- appendix B OSGi development with Maven 2* 419
- appendix C Spring DM development with Ant and Ivy* 458
- appendix D OSGi development with the Pax tools* 474
- index* 499