

# 深入理解OSGi



基于最新的OSGi R5.0规范编写，继《深入理解Java虚拟机》后的又一实力之作。  
全面解读OSGi规范，深刻揭示OSGi原理，并能掌握OSGi服务，本书介绍Equinox框架的实践，通过开源代码分析其工作机制，给出大量可操作性很强的  
解决方案和最佳实践。



周志明 谢小明 著



[深入理解OSGi 下载链接1](#)

著者:周志明

出版者:机械工业出版社

出版时间:2013-2-25

装帧:平装

isbn:9787111408871

本书是原创Java技术图书领域继《深入理解Java虚拟机》后的又一实力之作，也是全球首本基于最新OSGi R5.0规范的著作。理论方面，既全面解读了OSGi规范，深刻揭示了OSGi原理，详细讲解了OSGi服务，又系统地介绍了Equinox框架的使用方法，并通过源码分析了该框架的

工作机制；实践方面，不仅包含一些典型的案例，还总结了大量的最佳实践，极具实践指导意义。

全书共14章，分4个部分。第一部分（第1章）：走近OSGi，主要介绍了什么是OSGi以及为什么要使用OSGi。第二部分（第2~4章）：OSGi规范与原理，对最新的OSGi R5.0中的核心规范进行了全面的解读，首先讲解了OSGi模块的建立、描述、依赖关系的处理，然后讲解了Bundle的启动原理和调度管理，最后讲解了与本地及远程服务相关的内容。第三部分：OSGi服务与Equinox应用实践（第5~11章），不仅详细讲解了OSGi服务纲要规范和企业级规范中最常用的几个子规范和服务的技术细节，还通过一个基于Equinox的BBS案例演示了Equinox的使用方法，最重要的是还通过源码分析了Equinox关键功能的实现机制和原理。第四部分：最佳实践（第12~14章），总结了大量关于OSGi的最佳实践，包括从Bundle如何命名、模块划分、依赖关系处理到保持OSGi动态性、管理程序启动顺序、使用API基线管理模块版本等各方面的实践技巧，此外还介绍了Spring DM的原理以及如何在OSGi环节中进行程序测试。

## 作者介绍：

### 周志明

资深Java技术专家，对JavaEE企业级应用开发、OSGi、Java虚拟机和工作流等都有深入的研究，并在大量的实践中积累了丰富的经验。尤其精通Java虚拟机，撰写了大量与JVM相关的经典文章，被各大技术社区争相转载，是ITeye等技术社区公认的Java虚拟机方面的领袖人物之一，著有经典著作《深入理解Java虚拟机：JVM高级特性与最佳实践》（2年内重印十数次，获得业界多个图书奖项），广获读者好评。现任远光软件股份有限公司平台开发部经理兼平台架构师，先后参与过国家电网、南方电网等多个国家级大型ERP项目的平台架构工作，对软件系统架构也有深刻的认识和体会。

### 谢小明

资深Java技术专家，具有丰富的JavaEE企业级应用开发开发经验，对OSGi、Spring、Hibernate等技术和框架有比较深入的研究和认识。喜欢钻研运行与JVM之上的其他语言，如Scala和Clojure等。现任远光软件股份有限公司平台开发部平台架构师，先后参加与过国家电网、南方电网等多个国家级大型ERP项目的平台架构工作。

## 目录: 前言

### 第一部分 走近OSGi

#### 第1章 Java模块化之路 / 2

##### 1.1 什么是OSGi / 2

###### 1.1.1 OSGi规范的演进 / 4

###### 1.1.2 Java模块化规范之争 / 7

##### 1.2 为什么使用OSGi / 11

###### 1.2.1 OSGi能让软件开发变得更容易吗 / 12

###### 1.2.2 OSGi能让系统变得更稳定吗 / 13

###### 1.2.3 OSGi能让系统运行得更快吗 / 14

###### 1.2.4 OSGi能支撑企业级开发吗 / 15

##### 1.3 本章小结 / 16

### 第二部分 OSGi规范与原理

#### 第2章 模块层规范与原理 / 18

##### 2.1 OSGi规范概要 / 18

##### 2.2 Bundle / 20

##### 2.3 描述元数据 / 21

###### 2.3.1 预定义标记 / 21

###### 2.3.2 使用可视化工具 / 27

##### 2.4 Bundle的组织与依赖 / 31

2.4.1 导出和导入Package / 31

2.4.2 约束规则与示例 / 38

2.4.3 校验Bundle有效性 / 44

2.5 OSGi的类加载架构 / 45

2.5.1 父类加载器 / 46

2.5.2 Bundle类加载器 / 47

2.5.3 其他类加载器 / 49

2.5.4 类加载顺序 / 50

2.6 定义执行环境 / 51

2.7 本地化 / 54

2.8 本章小结 / 55

第3章 生命周期层规范与原理 / 56

3.1 Bundle标识 / 56

3.2 Bundle状态及转换 / 57

3.2.1 安装过程 / 59

3.2.2 解析过程 / 61

3.2.3 启动过程 / 62

3.2.4 更新过程 / 63

3.2.5 停止过程 / 64

3.2.6 卸载过程 / 65

3.3 启动级别 / 65

3.3.1 设置启动级别 / 66

3.3.2 调整活动启动级别 / 67

3.4 事件监听 / 68

3.4.1 事件类型 / 69

3.4.2 事件分派 / 70

3.5 系统Bundle / 71

3.6 Bundle上下文 / 72

3.7 本章小结 / 73

第4章 服务层规范与原理 / 74

4.1 服务 / 74

4.2 OSGi服务示例 / 75

4.3 服务属性 / 80

4.3.1 属性过滤器 / 82

4.3.2 预定义属性 / 83

4.3.3 修改属性 / 84

4.4 服务工厂 / 85

4.5 服务跟踪器 / 86

4.6 引用服务 / 89

4.7 释放和注销服务 / 91

4.8 服务层事件 / 91

4.8.1 事件类型 / 92

4.8.2 事件分派 / 92

4.8.3 ServiceRegistration对象的提前请求 / 93

4.9 远程服务 / 94

4.9.1 准备远程服务环境 / 94

4.9.2 远程服务示例 / 96

4.9.3 远程服务属性 / 99

4.9.4 实现分析 / 100

4.10 服务钩子 / 101

4.10.1 EventListenerHook / 101

4.10.2 FindHook / 101

4.10.3 ListenerHook / 102

4.10.4 服务钩子示例 / 102

4.11 本章小结 / 105

第三部分 基于Equinox的OSGi应用与实践

第5章 Equinox启航 / 108

5.1 建立Equinox开发环境 / 109

5.1.1 建立运行环境 / 109

5.1.2 建立编译及调试环境 / 110

5.1.3 建立开发环境 / 112

5.2 Equinox常用组件简介 / 117

5.3 Equinox启动器 / 119

5.4 使用代码启动Equinox / 124

5.5 本章小结 / 125

第6章 Equinox基础应用与源码解析 / 126

6.1 实践项目—Neonat论坛 / 126

6.1.1 背景与需求 / 126

6.1.2 模块划分 / 127

6.1.3 基础资料模块 / 129

6.1.4 持久化模块 / 133

6.1.5 用户交互模块 / 135

6.1.6 运行效果 / 140

6.2 Equinox源码解析 / 142

6.2.1 OSGi容器启动 / 142

6.2.2 Bundle状态恢复 / 147

6.2.3 解析Bundle依赖关系 / 153

6.2.4 OSGi类加载器实现 / 157

6.3 本章小结 / 162

第7章 服务器端OSGi / 163

7.1 OSGi与Web服务器 / 163

7.2 HTTP Service规范简介 / 166

7.2.1 服务目标 / 166

7.2.2 服务接口 / 168

7.2.3 资源映射规则 / 169

7.2.4 请求处理过程 / 171

7.3 实践项目—Neonat论坛的Web模块 / 171

7.3.1 准备依赖项 / 172

7.3.2 使用HTTP Service / 174

7.3.3 实现Web交互功能 / 176

7.3.4 运行效果 / 178

7.4 HTTP Service源码解析 / 180

7.4.1 BridgeServlet与OSGi容器启动 / 180

7.4.2 BridgeServlet与HTTP请求委派 / 186

7.4.3 DelegateServlet实现原理 / 188

7.5 本章小结 / 192

第8章 用户管理服务 / 193

8.1 User Admin服务规范简介 / 193

8.1.1 服务目标与基础概念 / 193

8.1.2 验证用户身份 / 195

8.1.3 验证用户权限 / 196

8.1.4 User Admin事件 / 197

8.2 实践项目—Neonat论坛用户管理模块 / 198

8.2.1 需求与依赖项分析 / 198

8.2.2 用户与用户组的实现 / 200

8.2.3 页面权限 / 201

8.2.4 用户登录与身份验证 / 202

8.3 User Admin源码解析 / 206

8.3.1 用户管理实现 / 206
8.3.2 外部服务使用实践 / 208
8.4 本章小结 / 211
第9章 Preferences服务 / 212
9.1 Preferences服务规范简介 / 212
9.1.1 服务目标 / 212
9.1.2 数据结构 / 214
9.1.3 属性 / 215
9.1.4 并发处理 / 216
9.1.5 清理遗留数据 / 217
9.2 实践项目—Neonat论坛持久化模块 / 217
9.2.1 编码实现 / 217
9.2.2 模块热切换 / 220
9.3 Preferences源码解析 / 222
9.3.1 数据结构实现 / 224
9.3.2 属性存取 / 228
9.3.3 后端存储系统 / 229
9.4 本章小结 / 234
第10章 声明式服务 / 235
10.1 声明式服务规范简介 / 236
10.1.1 服务目标 / 236
10.1.2 定义Component / 236
10.1.3 Component类型 / 237
10.1.4 Component生命周期 / 240
10.1.5 Component属性 / 245
10.1.6 绑定与发布服务 / 245
10.1.7 激活与钝化方法 / 252
10.1.8 Component配置总结 / 254
10.2 实践项目—使用声明式服务改造Neonat论坛 / 259
10.2.1 可视化编辑工具 / 259
10.2.2 DS容器管理 / 263
10.3 DS容器源码解析 / 264
10.3.1 容器启动 / 264
10.3.2 加载Bundle中的Component / 267
10.3.3 动态依赖解析 / 272
10.4 本章小结 / 274
第11章 Subsystems服务 / 276
11.1 服务目标 / 276
11.2 Subsystem格式 / 277
11.3 Subsystem元数据 / 278
11.3.1 SUBSYSTEM.MF标识 / 278
11.3.2 DEPLOYMENT.MF标识 / 281
11.4 Subsystem类型与共享策略 / 283
11.5 组织管理Subsystem / 285
11.6 Subsystem部署与依赖策略 / 289
11.7 Subsystem生命周期 / 291
11.7.1 安装 / 292
11.7.2 解析 / 293
11.7.3 启动 / 294
11.7.4 停止 / 294
11.7.5 卸载 / 295
11.8 本章小结 / 295
第四部分 最佳实践
第12章 OSGi最佳实践 / 298

12.1 Bundle相关名称命名 / 298
12.2 Bundle划分原则 / 300
12.2.1 恰如其分地分配Bundle粒度 / 300
12.2.2 分离OSGi代码 / 300
12.2.3 分离接口和实现 / 300
12.3 依赖关系实践 / 301
12.3.1 依赖分析工具 / 301
12.3.2 避免Require-Bundle / 303
12.3.3 最小化依赖 / 304
12.3.4 避免循环依赖 / 304
12.3.5 Equinox x-*依赖 / 305
12.4 Equinox专有类加载机制 / 306
12.4.1 Buddy Loading类加载机制 / 306
12.4.2 ClassLoaderDelegateHook类加载机制 / 307
12.5 Bundle生命周期实践 / 309
12.5.1 启动 / 309
12.5.2 停止 / 309
12.6 服务工厂的特殊性 / 309
12.7 处理非OSGi的JAR包 / 311
12.8 启动顺序实践 / 313
12.8.1 避免启动顺序依赖 / 313
12.8.2 Start Level的使用 / 313
12.9 Fragment Bundle实践 / 314
12.10 保持OSGi动态性 / 315
12.11 API Tools实践 / 317
12.11.1 API Baselines / 317
12.11.2 API Tools注解 / 319
12.11.3 API Version版本管理 / 322
12.11.4 二进制文件不兼容 / 322
12.12 本章小结 / 322
第13章 Spring Dynamic Modules实践 / 324
13.1 Spring DM入门 / 324
13.1.1 Spring DM项目简介 / 324
13.1.2 安装Spring DM / 325
13.1.3 简单的Spring DM示例 / 326
13.1.4 Bundle和Spring上下文 / 331
13.1.5 <osgi:*>命名空间 / 333
13.2 Spring DM进阶 / 337
13.2.1 Spring DM扩展配置 / 337
13.2.2 Web Extender / 344
13.2.3 Spring DM服务约束 / 345
13.2.4 在Spring上下文中使用BundleContext / 346
13.3 Spring DM企业应用 / 346
13.3.1 规划OSGi组件 / 347
13.3.2 在Spring DM中使用JPA / 348
13.3.3 事务管理 / 353
13.3.4 OSGi企业规范中的JPA / 358
13.4 Spring DM和Blueprint / 359
13.5 本章小结 / 360
第14章 构建可测试的OSGi系统 / 361
14.1 单元测试的必要性 / 362
14.2 单元测试的重要性 / 363
14.3 可测试代码的特征 / 364
14.4 OSGi单元测试 / 365

14.4.1 如何组织测试代码 / 366
14.4.2 如何进行OSGi单元测试 / 367
14.5 OSGi集成测试 / 373
14.5.1 Eclipse JUnit Plug-in Test / 374
14.5.2 Spring DM Test / 379
14.5.3 Pax Exam / 383
14.6 本章小结 / 384
附录A Java类加载器简介 / 385
附录B Equinox控制台命令 / 392
附录C OSGi子规范目录 / 397
附录D OSGi相关项目 / 399
附录E Equinox启动配置参数 / 401
• • • • • (收起)

[深入理解OSGi 下载链接1](#)

## 标签

OSGi

java

Java

编程

计算机

软件开发

J2EE

Web

## 评论

真以为可以深入，看来是我想多了

不错//因为看到Atlas用了这个，所以先了解。。周志明写的大概不会太差？开始了解这块

1. 和《深入理解Java虚拟机》同一作者，书的内容组织和文字很好，有借鉴意义。 2. 只看了第一和第二部分，其中背景和规范部分一看就懂，写得很好。 3. 可以在需要时重读。

值得一读，对于理解一些原理性的东西有帮助

感觉离使用还差着一些…

软件功能组件化，热插拔

一般

讲的深入浅出，最重要是融合了作者的实践经验，非常值得借鉴，不过建议有osgi使用经验的看看，完全没接触过osgi的就算了

[深入理解OSGi 下载链接1](#)

书评

[深入理解OSGi 下载链接1](#)