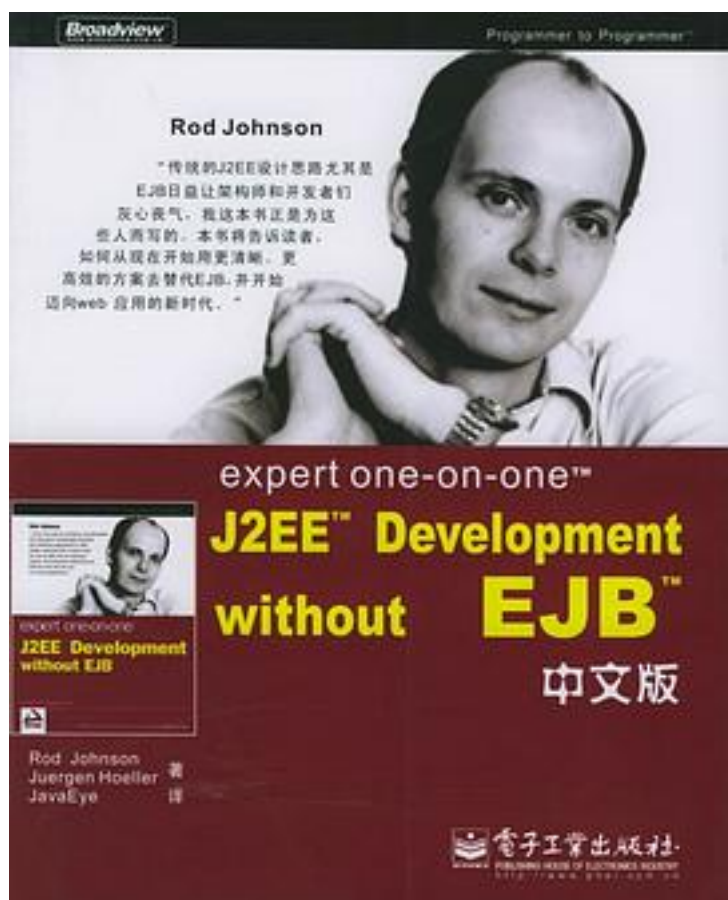


Expert One-on-One J2EE Development without EJB中文版



[Expert One-on-One J2EE Development without EJB中文版_下载链接1](#)

著者:[美] Rod Johnson

出版者:电子工业出版社

出版时间:2005-9

装帧:

isbn:9787121016844

乍一看这本书的名字，Expert one on one J2EE development without EJB并没有给人带来太冲击。毕竟关于J2EE的书太多了，而without EJB看上去有点象是故意挑衅EJB的感觉。一本J2EE的书怎么可能会给人带来信念或思

维的冲击呢？但是它做到了，它不仅使自己变成了不朽的经典，也使Rod Johnson成为了我最近一年的新偶像。

--xiecc

你的J2EE项目是否耗费了你太多的时间？它们是否难以调试？它们是否效率不彰？也许你还在使用传统的J2EE方案，然而这种主案太过复杂，而且并非真正面向对象。这里的很多问题都与EJB有关：EJB是一种复杂的技术，但它没有兑现自己曾经的承诺。

在这本实战手册中，你将看到另一种截然不同的方案：没有EJB，却可以创建质量更高的应用程序，所需的时间和成本则更低。你将学会如何充分利用各种实用的技巧和工具，包括时下流行的Spring框架和Hibernate两个开源工具。你将看到如何高效地解决企业级应用的核心问题，例如事务管理、持久化、远程调用和web设计。你将了解这种新的方案给可测试性、性能和可伸缩性带来怎样的影响，并亲身体验轻量级架构如何大幅降低项目开发所需的时间和工作量。

自从servlet、EJB、JSP等J2EE技术发布之初，本书作者Rod Johnson就一直在使用这些技术，他对于这些技术的优劣利弊了如指掌。现在，通过这本书，你将可以面对面地分享他的专家经验。

你将从本书学到……

如何针对自己的应用程序找到最简单、最易维护的架构；在不使用EJB的情况下有效地管理事务；如何利用AOP和IoC解决企业级软件开发中的常见问题；web层设计，以web层在设计良好的J2EE应用中的地位；J2EE应用中最有效的数据访问技术，包括JDBC、Hibernate和JDO；如何利用开源产品提升生产率、减少编码量；如何从设计层面上改善性能和可伸缩性。

“传统的J2EE设计思路尤其是EJB日益让架构师和开发者们灰心丧气，我这本书正是为这些人而写的。本书将告诉读者，如何从现在开始用更清晰、更高效的方案去替代EJB，并开始迈向web应用的新时代。”

这本书拥有一大堆“看点”。譬如说，它的作者Rod Johnson拥有10年编写Java程序的经验，目前是Servlet和JDO 2.0两个JSR专家组的成员；再譬如说，书中着力介绍的Spring、Hibernate、WebWork等都是时下流行的开源框架，IoC、AOP之类都是时下流行的概念词汇。而最大的看点就赫然摆在这本书的封面上：“without EJB”。我们曾经在无数的书籍和文章中看到，EJB是J2EE的核心技术之一；而Rod Johnson的这本书竟然宣称，绝大多数的J2EE应用根本不需要EJB。这种近乎挑衅的姿态令任何一个负责的J2EE架构师很难不萌生一探究竟的念头——不论你是打算赞同他还是打算驳斥他。

但所有这些尽皆不是本书最大的价值所在。选择一种架构、一种技术的依据是什么？Rod Johnson认为，应该是基于实践的证据、来自历史项目或亲自试验的经验，而不是任何形式的偶像崇拜或者门户之见。书中谈到了企业应用方方面面的问题和解决办法，而这些方案无一不是这种“循证方法”的产物。除了把这些方案交给读者，Rod Johnson通过这本书希望传达的、更为重要的信息正是“循证”的工作方式——那原本就应该是程序员的工作方式。

作者介绍:

目录: 第1章 为什么要 “j2ee without ejb”

聚光灯下的ejb

j2ee还剩什么?

站在十字路口的j2ee

前行的路

主旋律

轻量级框架和容器

我们还应该使用ejb吗?

小结

第2章 目标

生产率

问题

传统j2ee方案解决生产率问题的办法

提升生产率更好的办法

OO

业务需求的重要性

经验过程的重要性

小结

第3章 各种架构

架构性构件

.业务服务层

向外部暴露业务对象

数据访问层, 或eis层

j2ee架构

两种ejb架构

两种非ejb架构

j2ee架构实例

“经典的” j2ee远程ejb架构

本地ejb架构

特制的非ejb架构

“轻量级容器架构”: 示例应用系统

确定是否采用应用服务器

小结

第4章 简单性的红利

复杂性的代价

在j2ee应用系统中, 导致复杂性产生的原因

导致复杂性的架构性原因

导致复杂性的文化性原因: 一个依靠复杂性为生的产业

复杂到什么地步就是过度了?

简单还是幼稚?

刚刚够好就行吗?

变化的趋势

总结

第5章 ejb, 五年间

炒作和经验

ejb和j2ee行业

实践中的ejb

一个过时的组件模型

java语言的进步

.net的挑战

web service

敏捷方法学的兴起

关于ejb目标的混淆

从未出现的组件市场

方兴未艾的新范式：aop
ejb, 我们真正需要什么？为什么无状态session bean如此流行？
声明性事务管理
远程调用
集群
线程管理
ejb实例池
资源池
安全
业务对象管理
总结：ejb的服务
ejb, 我们不想要什么？
容器的锁定
丑陋的结构，泛滥的类
部署描述文件的地狱
类加载器的地狱
测试
ejb的滥用
复杂的编程模型
简单的事情会变得困难
“让开发人员忽略企业应用的复杂性”，这个目标现实吗？
生产率的损失
可移植性的问题
ejb能浴火重生吗？
工具支持
ejb 3.0
神话与谬论
jee == ejb
使用ejb的可疑论据
继续前进
moving forward
选择是否使用ejb
传统的知识
今天的选择
后ejb时代的舆论
标准，创新，和开源
小结
第6章 轻量级容器与控制反转
轻量级容器
什么是轻量级容器？
我们到底为什么需要容器？
轻量级容器 vs. ejb容器
ejb的好处
管理业务对象
接口与实现的分离
ejb：不完善的解决方案
控制反转
ioc实现策略
ioc容器
ioc容器间的移植
对代码风格、测试以及开发过程的影响
代码风格
可测试性
开发过程

使用企业级服务

小结

第7章 spring框架简介

来历与动机

一个分层的应用框架

基础构建模块

j2ee之上的spring

web应用中的spring

核心bean工厂

基础接口

通过xml组装bean

非xml格式的bean声明

组装应用对象

自动装配和依赖检查

构造子决议

生命周期回调

复杂的属性值

资源设置

典型的java/j2ee资源访问

bean容器中的资源声明

工厂bean

spring应用上下文

生命周期回调

信息源

文件资源

bean factory 后处理

小结

第8章 基于aop概念的声明性中间件

aop 101

动机

j2ee中的aop

定义

历史

作为aop子集的ejb

aop实现策略

动态代理

动态字节码生成

java代码生成

使用定制类加载器

语言扩展

aop实现

aspectj

aspectwerkz

jboss4

spring

nanning

aop联盟

aop设计问题

aop的危险性

aop设计的建议

随意点菜的j2ee

spring中的aop实践

使用proxyfactorybean

便利的factorybean

自动代理
编程用法
使用源码级元数据提供aop之上的抽象
.net范例
概念级元数据 vs. 实现级元数据
编程访问上下文信息
spring范例
ejb 3.0
编程风格的暗含意味
前后一致的命名规范
避免依赖aop基础设施
受控异常和增强
参考资料
书籍
论文
文章和在线资源
小结
第9章 事务管理
上层 (high-level) 事务管理
传统的j2ee事务管理
j2ee容器作为事务协调器
人见人爱的cmt
直接使用jta
插曲：远程事务传播
轻量级事务基础设施
spring framework的事务管理
事务声明
编程式事务处理
声明式事务管理
事务管理策略
选择j2ee服务器的提示
小结
第10章 持久化
常见持久化策略
持久化模式概览
流行的j2ee数据访问解决方案
选择一种持久化策略
透明持久化和领域对象的行为
java持久化技术简史
java o/r映射解决方案的缓慢成长
entity bean的败笔
实践中的数据访问技术
资源管理
jdbc
ibatis sql映射
jdo
hibernate
数据访问对象 (dao) 模式
业务对象与数据访问对象
dao和透明持久化
数据访问对象的种类
dao设计中的问题
dao基础设施的问题
使用spring框架进行数据访问

通用的数据访问异常
再论业务对象与数据访问对象的关系
jdbc
ibatis sql映射
jdo
hibernate
小结
第11章 远程调用
经典的j2se远程方案： rmi
访问和暴露rmi服务
用rmi调用器实现透明远程调用
经典的j2ee远程机制： ejb
通信协议
状态管理
访问远程ejb
部署远程ejb
基于wsdl的web services： jax-rpc
访问web services
servlet和ejb端点
轻量级远程方案： hessian和burlap
访问和暴露hessian和burlap服务
小结
第12章 替换其它的ejb服务
线程管理
线程神话
ejb线程模型
ejb实例池
何时需要实例池？
何时不需要实例池
ejb线程机制和缓冲池的替代方案
线程模型
实例池概述
声明性安全
ejb安全模型
ejb模型的缺陷
借助aop的声明式安全
jms和消息驱动bean
小结
第13章 web层设计
目标和体系结构的讨论
web层设计目标
用servlet和jsp定制的mvc
融入整体架构
请求驱动 web mvc框架
struts 1.1
webwork2
spring的web mvc框架
适宜的视图技术
web mvc的其它实现方式
portals和portlets
事件驱动的web mvc框架
小论asp.net
总结
第14章 单元测试与可测试性

为何测试很重要？

单元测试的目标

确保可测试性

编程风格

如何让你的代码难于测试

来自标准库的难题

提高可测试性的技巧

依赖倒置

aop

单元测试技巧

替换

模仿对象

编写有效测试

测试驱动开发 (tdd)

好处

对tdd的反对意见

tdd实践

学习tdd

案例研究：spring的经验

测试spring应用程序

对pojo进行测试

spring的抽象带来的好处

何时需要依赖spring api

使用替换配置进行测试

覆盖率分析和其他测试工具

测试生成器

覆盖分析工具

突变测试工具

资源

小结

第15章 性能与可伸缩性

定义

设置清晰的目标

体系结构的选择：影响性能和可伸缩性的关键因素

对象分布、集群和农场

数据访问

其他体系结构方面的问题

不同实现的选择

摆脱ejb服务设施对性能的影响

结果总结

摆脱ejb服务设施对性能的影响

缓存的代码优化

调优和部署

jvm

应用服务器

框架配置

数据库配置

一种循证的性能策略

基准测试

采样 (profiling)

诊断

资源

小结

第16章 示例应用系统

pet store (宠物店) 业务需求
ibatis jpetstore 3.1
中间层
远程调用机制
可改进的空间
spring jpetstore
中间层
数据访问层
web层
远程机制
编译和部署
war部署中的一些问题
部署spring jpetstore
小结
第17章 结语
回顾
前行
为你的应用选择最佳架构
轻量级容器架构
标准关键词
指导方针
编程风格
控制反转 (ioc) 和依赖注入
aop
测试
写在最后
索引
• • • • • ([收起](#))

[Expert One-on-One J2EE Development without EJB中文版_下载链接1](#)

标签

Java

j2ee

spring

软件开发

springframework

计算机

设计模式

Programming

评论

2006十大IT好书之一。

翻译的还好

过了那个 spring pk j2ee 的时间，读起来感觉不大。

rod在spring1.0成熟后推出的without EJB彻底将EJB2.X推向坟墓

现在读过来，AOP是一大亮点。很好

读过,但是感觉读的不够深入

几年前读的 是我架构思维的启蒙书

值的收藏的和反复读的书。

这本书的内容主要是作者对J2EE实践的经验和思辨，内容很赞。快速地浏览了一遍，只能加深了对一些核心概念的理解，而没法完全看得动这本书……再次体会，没有扎实的基础知识和足够的实践经验，有些好书真的看不动。。。

对J2EE社区而言，属于石破天惊的著作。

很经典，慢慢理解

本书难度较大，适合做过很多J2EE开发的人看，而不是只用SSH轻量级开发框架的人

写这书的目的完全是为了满足Spring之父对EJB的大力吐槽嘛

本来在5分档次的，可惜翻译得太烂。瑕不掩瑜。

这个经典了，做J2EE的几个没读过呢。

扫了一边。

经典的书，高屋建瓴。力荐！

一本j2ee界的奇书，必读，正在研磨中

我司也要抛弃ejb了

我只能说我并不是那么读得懂。

[Expert One-on-One J2EE Development without EJB中文版_下载链接1](#)

书评

记得数年前，还是EJB2.0刚开始流行的年代。第一次在tss上看到这本书的介绍，当时功力尚浅，是EJB的狂热追随者，对这本“非主流”的书当然是嗤之以鼻了。时至今日，做过大大小小的项目已经过亿元，终于知道了这本书的价值。而他也已经从“非主流java书”变成了公认的“J2EE经...

尽管没看过，但是很多朋友都说这书好，.net程序员也可以很好的借鉴其中的思想，不过e文实在太差，只好看中文的了。:-<

实际上我还没有读完这本书，而且我也没有看懂这本书，但是他确实给了我很多想法，或者叫灵感。这本书同样不推荐在不了解JAVA的情感下，去读这本书。可能又有人会觉得我故弄玄虚，其实只是想建议大家有写EJB程序的经验，至少写过几个例子，搞清楚了里面有几个XML文件，这些基...

大致翻了一下样章第一章，看着挺累，或许和我之前从未接触J2EE有关，不过至少许多文法语法方面读起来有些稀奇古怪，影响阅读速度。怀疑直接看英文版也能有这样的速度。关于技术书籍，我同意侯捷先生的主张：能保留英文原文的尽量原样保留。

我原来读的是电子版的，就是有点累，比较多，但是内容很好，尤其对那些正在抉择是否使用ejb的朋友，会给你很多不使用的理由

读这本书首先需要有相当的技术知识和实战经验。

至于翻译问题，可能有，但是应该不会太离谱。
最重要的是你要有实力明白书中语句的意义。
有些词的翻译，本来就没有定论的，或者约定的。
其实技术书籍的中文翻译，如果把一些关键词的中文后面加个括号，把原来的英文放进去，读...

当然最好能现看一下Rod Johnson的成名作：《Expert One-on-One J2EE Design and Development》——这本书是spring的前身。反正2本书都应该好好看看

当年看一份spring的文档，作者讲述了一个印度同事对此书的赞许，因此知道了这本书。前几天同事（资深j2ee工程师）
问我有啥j2ee的书推荐的，我第一反应就是这本书。虽然自己还没看过，但是知道它的主旨，无论如何，都是好书，因为有了这本书才有了现在的很多概念

有些术语翻译的还需要拿捏斟酌一下，比如“受管对象”，如果翻译成国内惯用的“托管对象”，会不会好一些？听说rod Johnson的语言生硬，可能有些地方没有意译好。没用过ejb，作者大量使用ejb进行对比，可能需要大概了解一些ejb的概念，更方便对本书所述概念的理解；目前章节已...

这不是一本开发细节的书，而是由替换EJB(2.0)所引发的对J2EE整体架构的一个解构，涵盖架构，开发，测试，无一不入木三分。虽是“without EJB”，但是EJB的“阴魂”却贯穿全书的始终，EJB的不足，EJB在特定领域的优势，EJB的替代……可以说，EJB构成了本书的基石和参照。另一方面...

需要说明软件开发的技术与思想是在不断迭代发展的工程中，新的技术相对来说是会比就技术要清晰简洁，比较Spring与EJB，90%的应用场景下的确Spring是个较优的选择，经过不同思想、技术的演化，Spring比较不同技术的优缺点，在多种约束下进行了权衡，也做了很多改进，但是我们不...

俺就看的中文版，感觉浪费了很多时间，我知道 Rod Johnson 是大师级的人物，但感觉不太好。我还是想把有限的时间用到看如何应用框架的技术上，在此基础上讲解含义，我比较务实，推荐《精通Spring2.x--企业应用开发详解》感觉不错，大家别骂我呀。没入行的不要评论我说的话，...

刊这本书是2年以前的事情了，本书是我阅读的所有技术书籍当中看的最仔细最认真的一本，也是我接触并应用spring+hibernate框架开发较长时间后读到的醍醐灌顶的感受效果。本书改变了我对软件架构的固有看法，使得像我这样的技术人员从更加高的角度去看待系统设计的问题，选择什么...

名气大的很，让你想不看都觉得落伍的一本书。在没有用spring之前看，对作者的思想敬佩不已；在用过spring做一个项目后再看，仍获益良多。虽然它不像spring in action那样实用，然而知其然而知其所以然，此书又不可多得。

书是好书，只是我太迟看了，书中很多思想已经在其他书或文章中接触过。所以我读这本书时并没有像很多人一样有很大的思想冲击。

尽管gigix在序言里说“还负责全书的文字修润”，但是如果首次的翻译已经不堪入目，那么再加修润无疑是愈行愈远，毫无帮助；何况从某些章节的翻译来看，根本就是脱离了英文原文在那儿做盲目的所谓修润（如果有的话）。而技术书籍的根本我个人以为“忠实原文，行文流畅”是至少应...

对于这本书最早的认识,是在一期程序员上透明推荐的"J2EE开发四书五经",在推荐这本书之前,是Sun的那本"Core J2EE Patterns",推荐语说到如果你看了without EJB,对于core patterns也就没有兴趣了.
我并没有开发大型J2EE应用的经验,就目前来说我所写过的程序还都是一些Toy,我也没有...

不管翻译的好坏，首先得尊重人家的劳动成果。
我目前正在看这本书，个人感觉看看还是挺好的，可能是因为我技术实力太差的缘故。

[Expert One-on-One J2EE Development without EJB中文版_下载链接1](#)