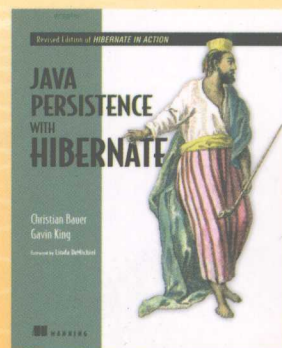


Java Persistence with Hibernate

Hibernate 实战 (第2版)

[德] Christian Bauer 著
[澳] Gavin King 著
杨春花 彭永康 俞黎敏 译

- Hibernate之父亲自执笔
- 无可替代的Hibernate和ORM技术圣经
- 职业Java程序员必备秘籍



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

Java 系列

Java Persistence with Hibernate **Hibernate**实战 (第2版)

[德] Christian Bauer 著
[澳] Gavin King
杨春花 彭永康 俞黎敏 译

人民邮电出版社
北京

图书在版编目 (CIP) 数据

Hibernate 实战: 第2版 / (德) 鲍尔 (Bauer, C.),
(澳) 金 (King, G.) 著; 杨春花, 彭永康, 俞黎敏译.
—北京: 人民邮电出版社, 2008.4

(图灵程序设计丛书)

ISBN 978-7-115-17448-2

I . H… II . ①鲍… ②金… ③杨… ④彭… ⑤俞…
III . JAVA 语言—程序设计 IV . TP312

中国版本图书馆 CIP 数据核字 (2008) 第 000463 号

内 容 提 要

本书全面讲述了 Hibernate 和 Java Persistence, 并对两者进行了详细的对比分析。书中通过一个应用将数百个例子融合起来, 不仅深入剖析了 Hibernate 3.2 和 Java Persistence 丰富的编程模型, 还深刻阐述了数据库设计、对象/关系映射 (ORM) 和优化等方面的基本原则、策略和最佳实践。书中处处闪烁着作者超凡的真知灼见, 将大大升华读者对 ORM 乃至企业级应用开发的理解。

本书适合广大的开发人员, 从 ORM 的初学者到经验丰富的开发人员。

图灵程序设计丛书

Hibernate 实战 (第2版)

- ◆ 著 [德] Christian Bauer [澳] Gavin King
译 杨春花 彭永康 俞黎敏
责任编辑 傅志红 张继发
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
新华书店总店北京发行所经销
- ◆ 开本: 800×1000 1/16
印张: 40
字数: 945 千字 2008 年 4 月第 1 版
印数: 1—5 000 册 2008 年 4 月河北第 1 次印刷

著作权合同登记号 图字: 01-2007-3048 号

ISBN 978-7-115-17448-2/TP

定价: 99.00 元

读者服务热线: (010)88593802 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Original English language edition, entitled *Java Persistence with Hibernate* by Christian Bauer and Gavin King, published by Manning Publications Co., 209 Bruce Park Avenue, Greenwich, CT 06830. Copyright © 2007 by Manning Publications Co.

Simplified Chinese-language edition copyright © 2008 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Manning Publications Co.授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

对本书的赞誉

“Hibernate对Java Persistence乃至EJB 3.0的技术方向产生了巨大影响……本书必将成为广大开发人员的知识宝库。”

——Linda Demichiel, Sun公司EJB 3.0和Java Persistence标准负责人

“本书从实践出发，将概念和应用结合。……针对你的项目，本书重在讲述方法……相信你一定可以从中看到无数美妙的风景。”

——杨戈 (Yanger), Spring中文论坛创始人

“本书是终极解决方案。如果你是Hibernate开发人员，不要犹豫了，到书店去，买上一本！”

——JavaLobby

“本书是结合了最新的JPA标准的Hibernate指南，既详细讲述JPA标准的方方面面，又讲述Hibernate框架的各个重要的细节用法，不但对于使用Hibernate的开发人员来是不可或缺的参考书。”

——范凯 (Robbin), JavaEye网站创始人

“本书是使用ORM实现Java持久化最全面和最权威的著作！”

——Will Wagers, C#online.net

“本书让人大呼过瘾。对于我来讲，可能第三章就已经值回书价了。”

——豆瓣评论

“我衷心推荐本书！”

——Stuart Caborn, Thoughtworks公司资深顾问

“关于Hibernate的权威之作，适合任何开发人员。”

——Patrick Peak, BrowserMedia公司CTO, *Hibernate Quickly*一书作者

推 荐 序 一

Hibernate持久化框架已经诞生6年了，在这期间，Hibernate从当初不为人知的开源软件，到现在成为Java编程语言在操作关系数据库方面的持久化框架的统治者，不能不说是一个奇迹。

Java的持久化操作最早要追溯到1996年发布的JDBC接口。当时，JDBC是一个借鉴了ODBC标准，采用纯Java编写的数据库持久化接口。因为JDBC可以用同样的方式存取大多数主流关系数据库，可以做到操作系统无关、数据库无关、纯Java编写，且具备良好的可移植性，因此JDBC发布之后获得了极大的成功，时至今日，它还是Java程序员最常用的API之一。但是对程序员来说，JDBC API显得非常笨拙，如果数据库字段比较多，编写起来非常麻烦，代码量很大，而且不容易维护。因此，为了简化JDBC编程，随后又出现了很多基于JDBC的持久化框架，而Hibernate正是其中的佼佼者。

在Hibernate出现之前，最主流的持久化框架是1999年发布的EJB 1.1的实体bean，特别是在2001年EJB 2.0规范引入本地接口之后，一直到2003年，实体bean变得非常流行，成为应用服务器厂商力推的产品卖点之一。这段时间Java企业应用开发言必称EJB，好像不用EJB都显得不入流，你都不好意思说自己是Java高级程序员。但是，实体bean在持久化方面并没有起到积极的作用，反而极大增加了持久化代码的复杂度，因而逐渐被越来越多的开发人员反对和抛弃。

2001年，澳大利亚墨尔本一位名为Gavin King的27岁Java程序员，上街买了一本SQL编程的书，他厌倦了实体bean，认为自己可以开发出一个符合对象关系映射理论，并且真正好用的Java持久层框架，因此他需要先好好学习一下SQL。这一年的11月，Hibernate的第一个版本发布了。在整个2002年，实体bean还是万众瞩目的焦点，但是已经有很多人开始关注和使用Hibernate了。

2003年9月，Hibernate开发团队进入JBoss公司，开始全职开发Hibernate，从这个时候开始，Hibernate得到了突飞猛进的普及和发展。2004年，整个Java社区开始从实体bean向Hibernate转移，特别是在Rod Johnson的著作*Expert One-on-One J2EE Development without EJB*出版后，由于这本书以扎实的理论、充分的论据和翔实的论述否定了EJB，提出了轻量级敏捷开发理念之后，以Hibernate和Spring为代表的轻量级开源框架开始成为Java世界的主流和事实标准，Hibernate开始大红大紫。在2004年Sun领导的J2EE 5.0标准制定委员会当中，Gavin King已经是实体bean下一代标准的主要制定者，而J2EE 5.0标准当中的持久化框架标准正是以Hibernate为蓝本的。当2006年J2EE 5.0标准正式发布以后，持久化框架标准Java Persistent API(简称JPA)基本上是参考Hibernate实现的，而Hibernate从3.2版本开始，已经完全兼容JPA标准，成为最主流的JPA框架。

我是在2003年3月知道Hibernate框架的，当时我正在寻找一个可以替代实体bean的Java持久层框架，在考察了众多开源项目之后，发现Hibernate正是可以满足我需求的框架，因此在随后的几

个项目当中使用了Hibernate，并且获得了比较满意的效果。在2003年，虽然Hibernate在国外已经是非常有名气的Java开源框架，但是企业应用开发的主流仍然是EJB，因此国内的Java社区非常排斥这种既非J2EE官方标准，又没有大厂商支持的“草根”框架Hibernate。由于找不到一个可以交流Hibernate编程经验的地方，我萌生了自己创建一个Hibernate交流社区的想法，并在2003年9月11日凌晨正式开通了公共交流论坛，这就是JavaEye网站诞生的由来。JavaEye在2003年是国内唯一交流Hibernate的技术网站，不夸张地说，为国内推广和普及Hibernate立下了汗马功劳，影响了一大批Java开发人员。时至今日，JavaEye网站已经成为国内仅次于CSDN的第二大软件开发者交流社区。虽然JavaEye早已经不再是一个单纯的Hibernate技术讨论区，但是当初如果没有Hibernate的缘故，今天就不会有JavaEye网站了。

根据我四年多的Hibernate编程经验，Hibernate并不是一个非常易学易用的框架，它的掌握难度要远远高于同样十分流行的Spring框架。由于对象的持久化操作本身就包含非常多的概念和难点，运用不善还很容易导致性能低下，因此尽管现在Hibernate在国内已经非常普及和流行了，几乎每个软件公司在招聘Java程序员的时候都要求掌握Hibernate，但实事求是地讲，能把Hibernate用好的Java程序员凤毛麟角，在JavaEye网站上对Hibernate的置疑声音也从来就没有停歇过。其实善于使用Hibernate，能把Hibernate用好的案例也有不少，比如Google公司就在一些非常大规模的项目当中使用了Hibernate和WebWork框架（有传言说是Adwords项目，但我没有考证过），并且贡献了基于Hibernate的Hibernate Shards开源项目和从WebWork项目当中抽取出来的Google Guice开源项目。

因此要想把Hibernate用好，对于Hibernate框架的很多重要的运用细节需要非常的了解和熟悉，除了在项目当中摸索和总结Hibernate使用经验之外，由Gavin King共同执笔编写的这本书就是最好和最权威的Hibernate指南。这本书的第1版叫做*Hibernate in Action*，在Amazon上销量很好，但是由于版权问题，很遗憾未能引入国内。而本书的第2版，即现在这版已经是结合了最新的JPA标准的Hibernate指南，既详细讲述JPA标准的方方面面，又讲述Hibernate框架的各个重要的细节用法，不但对于使用Hibernate的开发人员来说是不可或缺的参考书，同样对于使用其他JPA实现的开发人员来说，也是重要的参考。

范凯（Robbin），JavaEye网站创始人
<http://www.JavaEye.com>

推 荐 序 二

Hibernate自2001年诞生以来，在Java的ORM领域几乎已经成为事实的标准。“掌握Hibernate”已经近乎成为使用Java的软件公司招聘程序员的一个基本要求。之所以形成这一流行做法，有Hibernate本身优秀的理念、成熟的设计与广泛的正面使用体验的因素，也有Hibernate本身的开源、免费的因素，尤其重要的是它包含内容丰富、更新及时的文档。

正因为Hibernate本身基于LGPL开源，我翻译了其2.0版本的文档。该文档在网上公开后获得了始料未及的反应，并且由于Hibernate社区的开放性，它被作为Hibernate文档的正式中文翻译版本纳入了其源代码管理，这是对我翻译工作的最好肯定。正因为其广泛的流传，在Hibernate 3.0版本发布的时候，得以组织起20余人的团队，在短短的三周时间内就推出了经过审校的质量相当高的文档翻译版本，并且不断根据新版本的发布进行更新。仅在“满江红”主站，这一版本就有超过15万次的PDF下载量，HTML版本访问量更是数百万次计。这一中文版本的文档形成了学习Hibernate的良好氛围，对其普及做出了极为重要的贡献，也直接促成了“满江红开放技术研究组织”的成立，同时对后续的很多文档的发布产生了非常积极的示范效应。

但是，官方文档仅有200余页，着重描述的是对其使用参数的设置说明，以及对HQL、关联等概念的使用说明，解释“如何做”而非“为何要这样做”，这对广大学习者仍然有较大的学习障碍。Hibernate的原作者Gavin King和Christian Bauer编写了*Hibernate in Action*（即《Hibernate实战》第1版）一书对其进行了详细的解释。该书英文版一推出，我们就开始了翻译工作，但是后来因版权问题，没能第一时间引进，这实在是一大憾事。

你现在手中的这本《Hibernate实战（第2版）》就是对*Hibernate in Action*的深化升级。这本书的优点在于其实用性，加入了大量的图片和示例辅助说明，比Hibernate随附的文档要清晰明了得多，其英文版我也一直放在电脑包中随身携带，作为速查手册使用。这是一本“高级版文档”和权威的学习指南。

本书是由俞黎敏组织并翻译、审校的。俞黎敏长期工作在开发第一线，对Hibernate有深刻的理解，是“满江红”的管理员之一，也是Gavin King新项目Seam的中文文档翻译负责人。他对本书的理解相当到位，翻译准确度很高。本书正弥补了当初没有能成功引进*Hibernate in Action*的遗憾。在此对他表示由衷的祝贺。

Java作为目前的主流开发语言，Hibernate和JPA仍然具有长期的生命力。希望本书能够进一步推进Hibernate和JPA在国内的应用。

曹晓钢

“满江红开放技术研发组织”共同创始人

<http://www.redsaga.com>

译者序

现在回想起来，自己最早接触数据库相关技术应当可以追溯到12年前，也就是我在大学里开始自学dBASE编程的时候。当时，我一次次地溜进企管系正在进行会计电算化试验的学校机房，翻开自己的笔记本，一行行地敲入程序，然后运行，输入生成表单所要求的数据后，再显示。看到屏幕上显示着一行行不停地滚动的数据，我心中一阵狂喜，感觉这一切是那么地神奇、有趣！

时至今日，无论是因特网应用程序还是企业应用系统，大多数都少不了数据库系统。关系数据库是现代企业的核心，真可谓“无数据库不成应用”。在Java的阵营里，JDBC是数据库操作最基本的API，但JDBC有它的不足之处，如工作量大，不易于重构，等等。Java开发人员之所以觉得难以使用JDBC API，是因为它只是数据库系统之上的一个薄层，只实现通信协议，所有与数据库交互的其他方面都必须由开发人员来承担。而且关系型系统中数据的表格表示法与用于Java应用程序的面向对象方式之间也存在本质上的不匹配，大大影响了Java应用程序的开发效率。于是，Hibernate应运而生。它通过ORM方法，在关系型数据和Java对象之间架起了桥梁，以一种非常实际、直接并且现实的方式很好地解决了对象/关系不匹配的问题。

虽然ORM工具为开发人员承担了95%的对象持久化工作，但是它并不是所有持久化任务的一颗银弹。你应当学会如何为充分利用ORM工具、为自己提高效率而进行相应的抉择，也应当能够充分地认识到它所带来的影响，以及何时需要和不需要它。

本书的作者是开发Hibernate项目的核心人员，他们所提供的见解是最前沿、最直接的。毋庸置疑，本书是Hibernate方面的权威之作，它展示了用Java的面向对象世界映射现实世界的范式，提供了对ORM深层问题和Hibernate背后设计选择的深刻见解，可以让读者对ORM作为一种企业技术的有效应用有深刻的理解。它比较适合中高级水平的读者，当然也适合刚准备进入ORM领域的读者，但是他们至少需要具备面向对象软件开发的基础知识和实践经验，并且熟悉Java编程语言和UML以及数据库的基本知识。

本书的前三章对想要利用Hibernate迅速创建应用程序的开发人员来说是不可或缺的，同时它们对于真正想要理解Hibernate概念、框架、方法以及形成框架设计的原因的人来说也显得尤为重要。

如果你知道Session.get()与Session.load()的区别，了解Hibernate的save()方法与JPA的persist()的区别，并且知道JPA没有Hibernate Session API所提供的evict(object)方法来剔除单个持久化实例的话，那就可以从任一部分开始阅读。

对于老用户来说，如果你了解Hibernate 3.x与Hibernate 2.x有何不同，可以认真看看Hibernate 3到底在内核持久化引擎上面做了哪些改动。比如，新内核引擎基于一个事件/监听器的

模型，如果Hibernate需要保存一个对象，就会触发一个事件。这些细节的底层实现与思想聆听作者为你娓娓道来，你可以有更加深刻的体会。

通过本书，你可以知道如何高效地使用HQL的最佳实践，也可以从中知道HQL与Criteria的优缺点，以及Criteria API在什么情况下会是更好的选择；你可以清楚地获知JBoss Cache支持什么样的事务，也可以由详细地对比分析JPA与Hiberante之后得到，譬如，如果你必须使用定位参数，要记住Hibernate是从0开始计数，而Java Persistence则从1开始计数，你必须给JPA QL查询字符串中的每一个问号都添加一个数字。而且可以知道如何减少内存消耗并防止很长的脏查询周期，但是JPA却不支持，以及更多其他信息，如distinct在SQL结果中重复吗？它是在SQL级别上操作还是内存中进行二次过滤呢？等等。

本书旨在使你更快速、更专业地进入ORM——这个企业级应用程序开发领域最核心的技术之一，并从事数据库相关的设计工作。当阅读本书时，你会遇到许多需要动手进行验证的实例，可以利用本书附带的示例程序进行练习与实践。下载地址为：<http://caveatemptor.hibernate.org/>。CaveatEmptor示例应用程序代码以原生的Hibernate、Java Persistence和JBoss Seam三种实现进行提供，方便读者学习与使用。

正如在翻译过程中发现原著的错误一样，虽然我们在翻译过程中竭力以求信、达、雅，但限于自身水平，必定仍会有诸多不足，还望各位读者不吝指正。大家可以通过访问我的博客<http://YuLimin.JavaEye.com/>或者发送电子邮件到YuLimin@163.com进行互动。

翻译时所发现的原书错误通过与作者的交流得到了确认，详细情况可以参见我的博客或Hibernate官方论坛 http://forum.hibernate.org/search.php?search_author=YuLimin。

关于术语的翻译，可以到<http://yulimin.javaeye.com/blog/70124>讨论。

感谢ajax.cn的李锟（dlee）对我在翻译中碰到的问题进行的深入讨论，并对本书翻译时所采用的术语进行了认真的磋商；感谢JavaEye的网友们术语表讨论贴中提出许多中肯的建议；感谢满江红开源组织的曹晓钢提供的一些翻译注意事项和热情的帮助；感谢图灵公司的编辑。

本书由我组织翻译，杨春花负责翻译第1章到第8章，彭永康负责翻译第9章到第16章，我负责翻译第17章并对全书所有章节进行全面审校。参与翻译与审校的还有：蒋凌锋、魏伟、叶荣生、凌家亮、黄健、万国辉、宋可信、李宗波等，在此再次深表感谢。

本书章节安排合理，内容承上启下，阅读起来有一气呵成之感，不会让你感觉到是在啃“砖块”，而是在享受。快乐分享，实践出真知，最后，祝大家能够像我一样在阅读中享受本书带来的乐趣！

Read a bit and take it out, then come back read some more.

俞黎敏

2007年年底于广州

译者简介

杨春花

满江红开放技术研究组“Seam 2.0 Reference”中文翻译工作之核心翻译与审校人员。

彭永康

现任景德镇陶瓷学院信息工程学院信息与计算科学教研室主任，副教授。曾参与国家863CIMS单元目标产品项目“陶瓷产品CAD集成系统”，中小企业创新基金项目“陶瓷产品CAD集成软件”等。主要研究方向：计算机图形学、多媒体技术。

俞黎敏（ID：YuLimin，网名：阿敏总司令）

技术顾问，深入了解电力、电信行业的系统，并负责核心系统研发与管理工作。开源爱好者，曾经参与Spring中文论坛组织“Spring 2.0 Reference”中文翻译的一审与二审工作，满江红开放技术研究组织的“Seam 1.2.1 Reference”中文翻译工作，并组织和完成“Seam 2.0 Reference”中文翻译工作。利用业余时间担任CSDN、CJSDN、Dev2Dev、Matrix、JavaWorldTW、Spring中文等Java论坛版主，在各大技术社区推动开源和敏捷开发做出了积极的贡献。博客地址为：<http://YuLimin.JavaEye.com>

第 2 版序

两年前, *Hibernate in Action* 一出版就被公认为是关于Hibernate的权威性图书, 同时它也成为了ORM (对象/关系映射) 方面的权威之作。

之后, 持久化 (persistence) 的前景也随着JPA (Java Persistence API) 的发布而改变, 这个新标准用以解决Java EE和Java SE的ORM, 并在Java Community Process的主导下开发, 成为EJB 3.0规范的一部分。

在JPA的开发过程中, EJB 3.0专家组从已经在Java社区应用的ORM框架的经验中受益颇丰。Hibernate作为先驱者之一, 已经对Java Persistence技术的导向产生了非常显著的影响力。这不仅是由于有Gavin King和EJB 3.0标准化成果的其他Hibernate团队成员的参与, 很大程度上还归功于Hibernate带给ORM的直接且实用的方法, 以及其API的简单、清楚和强大——因此它们的结果激励了Java社区。

除了对Java Persistence的贡献之外, Hibernate的开发人员还使得Hibernate有了很大进展, 这体现在本书所描述的Hibernate 3中。这些进展包括支持对大型数据集的操作、特别适用处理遗留数据库的额外和更复杂的映射选项、数据过滤器、管理对话的策略, 以及与利用JSF和EJB 3.0进行Web应用程序开发的新框架Seam的集成。

因此, 本书远远不只是*Hibernate in Action*的第2版。它全面概括了所有JPA的功能和Hibernate 3的功能, 并对这两者进行了详细的对比分析。它描述了如何使用Hibernate实现Java Persistence标准, 以及如何利用Hibernate扩展来实现Java Persistence。

更重要的是, 在介绍Hibernate和Java Persistence的过程中, Christian Bauer和Gavin King举例并解释了在设计和使用ORM框架时需要考虑到的一些基本原则和决策。他们对ORM深层问题的深刻见解, 让读者对于ORM作为一种企业技术的有效应用有了深刻的理解。

因此, 本书适合广大的开发人员, 从ORM的初学者到经验丰富的开发人员, 帮助他们学习更多在Java社区中已经出现和正在继续出现的尖端技术。

Linda Demichiel

(Sun公司规范主管, EJB 3.0和Java Persistence首席架构师)

第 1 版序

毋庸置疑，关系数据库是现代企业的核心。

包括Java在内的现代程序设计语言，提供了应用程序级业务实体的直观、面向对象的视图，但这些业务实体底层的企业数据本质上是密切相关的。此外，相对于早期的导航模型以及后来的OODB模型，关系模型的主要优势在于，它被设计为与其提供数据的编程处理和应用程序级视图在本质上是无关的。

许多人已经做了很多尝试来搭建关系和面向对象技术之间的桥梁，或者用一个取代另一个，但是这两者之间的差异是当今企业计算面临的难点之一。通过其ORM方法，Hibernate架起了关系型数据和Java对象之间的桥梁。Hibernate是以一种非常实际、直接并且现实的方式来迎合这一挑战的。

正如Christian Bauer和Gavin King在书中所述，除了最简单的企业环境之外，最有效地应用ORM技术要求理解关系型数据和对象以及它们之间的中介的工作原理，并对中介进行配置。这要求开发人员通晓应用程序及其数据需求，以及SQL查询语言、关系存储结构和关系技术提供的最优化潜能。

Hibernate不仅提供了一个应对这些需求的功能齐全的解决方案，而且还提供了一个灵活的可配置的架构。Hibernate开发人员专注于设计，并使它具有模块化、可热插拔、可扩展以及可用户定制等特点。结果，在它最初发布的几年里，Hibernate迅速变成了企业开发人员所适用的一种前沿ORM技术——这也是理所当然的。

本书全面概括了Hibernate，涵盖了如何利用它的类型映射能力和工具来为关联和继承建模；如何利用Hibernate查询语言有效地获取对象；如何在托管和非托管环境下配置Hibernate；以及如何使用它的工具。此外，本书还提供了对ORM深层问题和Hibernate背后设计选择的深刻见解。这些见解可以让读者对ORM作为一种企业技术的有效应用有深刻的理解。

*Hibernate in Action*是当今企业计算中利用Hibernate和ORM的权威指南。

Linda Demichiel
(Sun公司，EJB首席架构师)

第 2 版前言

本书第1版*Hibernate in Action*的前言开始引用了Anthony Berglas的话：“不能只因为某人能用鼻子把细枝沿地面向前推，就认为那就是收集柴火的最好办法。”自第1版出版后，软件开发人员赖以管理信息的Hibernate项目和策略以及理念都得到了进一步发展。然而，基本的问题仍然一样——我们每天打交道的每家公司仍然在使用SQL数据库，并且Java在行业中成了企业应用程序开发的首选。

在关系型系统中数据的表格表示法与用于面向对象的Java应用程序的对象网络仍有本质上的区别。我们仍然看到对象/关系不匹配，而且经常看到这种不匹配的重要性和代价被低估了。

另一方面，我们现在有各种各样的工具和解决方案可用于处理这个问题。我们已经捡好了柴火，并且袖珍打火机也换成了火焰投射器。

现在Hibernate已经准备好了第3次重要的发布，Hibernate 3.2就是本书所讲述的版本。相比Hibernate的老版本，这个全新的重要发布具有两倍的特性——因此本书的厚度也几乎是*Hibernate in Action*的两倍。其中大部分特性都是包括你在内的每天使用Hibernate的开发人员所需要的。有时我们说，Hibernate能解决Java应用程序开发人员在创建数据库应用程序时必须处理的所有问题的90%。有了最新的Hibernate版本，这个数字就接近99%了。

随着Hibernate的成熟以及其用户基数和社区的不断成长，许多开发人员发现缺少数据管理和数据库应用程序开发的Java标准。我们甚至在*Hibernate in Action*中告诉过你不要使用的EJB 2.x的实体bean。

下面谈谈EJB 3.0和新的Java Persistence标准。这个新的行业标准对Java开发者社区而言是向前迈出了重要的一步。它定义了一个轻量级、简化的编程模型和强大的对象/关系持久化。这个新标准的许多关键理念都模仿Hibernate和其他成功的对象/关系持久化方案。最新的Hibernate版本实现了Java Persistence标准。

因此，除了这个新的集各种功能于一身的Hibernate之外，现在你可以像使用其他Java Persistence提供程序（provider）一样使用Hibernate，不管是否使用其他EJB 3.0组件和Java EE 5.0服务。Hibernate与这么一个丰富的编程模型的深入整合，使你能够设计和实现以前用手工难以创建的应用程序功能。

本书旨在提供一个对Hibernate和Java Persistence（以及所有相关的EJB 3.0理念）完整而准确的指南。希望你也会喜欢学习Hibernate，并且把本书放在案头作为每日工作的参考。

第 1 版前言

不能只因为某人能用鼻子把细枝沿地面向前推，就认为那就是收集柴火的最好办法。

——Anthony Berglas

今天，许多软件开发人员都用企业信息系统（EIS）。这种应用程序创建、管理和存储结构化信息，并在多个物理位置的许多用户之间共享信息。

EIS数据存储涉及大量使用基于SQL的数据库管理系统。我们曾经工作的每家公司至少都用一个SQL数据库；大部分公司的业务核心完全依赖关系数据库技术。

在过去的5年里，Java程序语言的广泛应用已经使软件开发中面向对象的范式占据了统治地位。现在，开发人员享受着面向对象的好处。然而，大量的业务还是依赖于对昂贵的关系数据库系统的长期投资。不仅仅是特殊的供应商产品不容易更改，而且现有遗留数据也必须可用于（且经由）这个全新的面向对象的Web应用程序。

然而，在关系型系统中数据的表格表示法与应用在面向对象的Java应用程序中的对象网络仍有本质上的不同。这种区别导致了所谓的对象/关系范式（object/relational paradigm）的不匹配。

传统上，这种不匹配的重要性和代价被低估了，并且解决这种不匹配的工具也不充足。同时，Java开发人员抱怨针对不匹配的关系技术；数据专家则抱怨对象技术。

ORM是对这种不匹配问题的自动化解决方案的命名。对于厌烦了冗长乏味的数据访问代码的开发人员来说，好消息是ORM已经成熟了。用ORM中间件创建的应用程序会更加便宜、更高性能、更少的供应商依赖性，并且更能够应付内部对象或者底层SQL模式的改变。令人惊异的是，这些好处现在都可以供Java开发人员免费享用。

Gavin King于2001年后期开始开发Hibernate，这时他发现流行的持久化解决方案——CMP Entity Beans——还没有发展成带有复杂数据模型的重要应用程序。Hibernate开始了作为一个独立的、非商业的、开源项目的生涯。

Hibernate团队（包括作者）通过勤奋的实践已经掌握了ORM——也就是说，通过聆听用户需求并实现满足这些需求所需要的东西。结果，Hibernate成了一个实用的解决方案，增进了开发人员的生产力和技术领导能力。Hibernate已经被数以万计的用户使用，并且用到了数以千计的生产应用中。

当那个时代的需求变得势不可挡时，Hibernate团队断定这个项目未来的成功（和Gavin持续的明智）要求专业开发人员将全部精力投入到Hibernate。Hibernate在2003年后期加入了jboss.org，现在具有了商业特征。可以从JBoss公司购买商业支持和培训，但是商业培训不应该是学习

Hibernate的唯一途径。

很明显，许多（可能甚至大多数）Java项目都受益于ORM解决方案（如Hibernate）的使用，虽然这在几年前还不明显。随着ORM技术日益成为主流，产品文档（如Hibernate的免费用户手册）已不再充足。我们认识到Hibernate社区和新的Hibernate用户需要一本全面的图书，不仅仅学习利用Hibernate来开发软件，还要理解和领悟对象/关系的不匹配，以及Hibernate设计背后的动机。

本书是我们利用一年多所有空闲时间努力创作的结晶，也是许多激烈辩论和学习经验的来源。我们希望本书成为关于Hibernate的优秀指南（即“Hibernate圣经”，如一位评论家所言），并且大体上也是对象/关系不匹配和ORM的第一个全面的文档。

希望你发现它对你有所帮助，并且喜欢用Hibernate工作。

关于本书

我们编写本书时有3个目标，因此你可以把它当作以下3种书来读：

- 一本关于Hibernate、Java Persistence和EJB 3.0的教程，指导你完成利用这些解决方案的第一步。
- 一本关于ORM、对象处理、查询、性能优化和应用程序设计的所有基础和高级Hibernate特性的学习指南。
- 一本在你需要一个关于Hibernate和Java Persistence功能的完整而技术准确的定义时的参考书。

图书通常要么是教程要么是参考指南，这种伸缩性是要付出很多努力的。如果你刚接触Hibernate，建议你从头开始读本书，从第1章和第2章的教程开始。如果你已经用过第1版的Hibernate，那么应该快速阅读前两章以获得整体概念，然后阅读第3章。

适当的时候，我们会告诉你某个特定的节或者主题是否为可选的，或者是否为第一次阅读时可以忽略的参考资料。

导读

本书分成三大部分。

第一部分介绍对象/关系范式的 mismatch，并阐述ORM（Object/Relational Mapping，对象/关系映射）背后的基本原理。通过一个实际操作的教程，让你开始第一个Hibernate、Java Persistence或者EJB 3.0项目。我们探讨领域模型的Java应用程序设计和创建ORM元数据的方法。

第二部分重点介绍如何把Java类和属性映射到SQL表和列。我们利用XML映射文件和Java注解，探讨Hibernate和Java Persistence中所有基础和高级的映射方法；介绍如何处理继承、集合和复杂的类关联。最后讨论与遗留数据库模式的整合，以及一些特别灵活的映射策略。

第三部分介绍关于对象的处理，以及如何使用Hibernate和Java Persistence加载和存储数据。我们介绍编程接口、如何编写事务性和支持对话的应用程序，以及如何编写查询。随后，把关注点放在分层Java应用程序的正确设计和实现上。我们讨论最常用于Hibernate的设计模式，例如DAO（Data Access Object，数据访问对象）模式和EJB Command模式。你将学会如何轻松地测试Hibernate应用程序，以及使用ORM软件时有哪些相关的其他最佳实践。

最后介绍JBoss Seam框架，它进一步深化了许多Hibernate概念，使你能够轻松地创建会话的Web应用程序。我们确信你会对这一章感兴趣，即使你不打算使用Seam。