



EJB

锦囊妙技

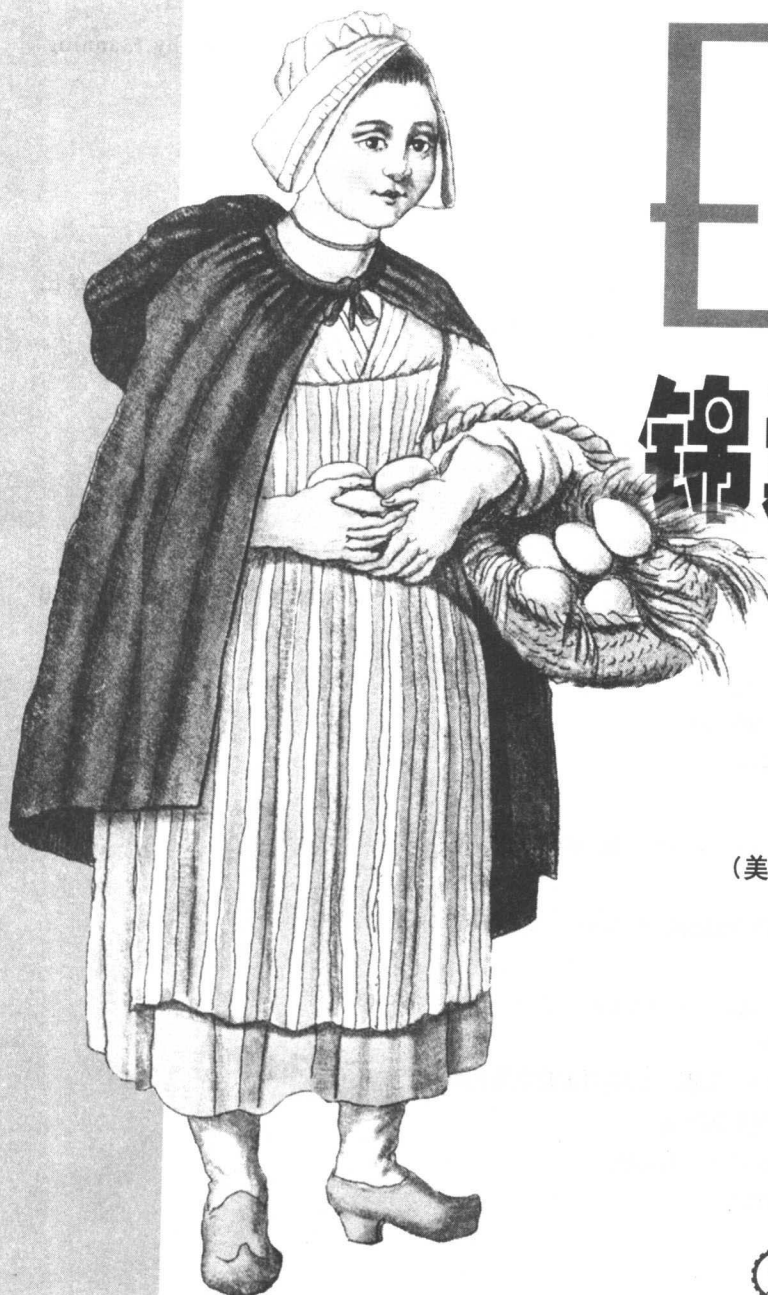
EJB COOKBOOK

(美) Benjamin G. Sullins 著
Mark B. Whipple
马朝晖 等译

 MANNING



机械工业出版社
China Machine Press



EJB

锦囊妙技

EJB COOKBOOK

(美) Benjamin G. Sullins 著
Mark B. Whipple

马朝晖 等译



机械工业出版社
China Machine Press

本书以问题/解决方案的格式进行组织,系统地收集了EJB开发过程中许多实际问题的解决方案。每个解决方案都有与之相关联的解决方案,使读者可以方便地进行查看和理解。

本书适合从事Java开发工作的技术人员阅读,对于那些对EJB感兴趣的读者同样适用。

Benjamin G. Sullins, Mark B. Whipple; EJB Cookbook(ISBN: 1-930110-94-4).

Authorized translation from the English language edition published by Manning Publications Co.

Copyright © 2003 by Manning Publications Co.

All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2004 by China Machine Press.

本书中文简体字版由Manning Publications Co.授权机械工业出版社独家出版。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。

本书版权登记号: 图字: 01-2004-4838

图书在版编目(CIP)数据

EJB 锦囊妙技/ (美) 萨林斯(Sullins, B. G.) 等著, 马朝晖等译. —北京: 机械工业出版社, 2004.9

(Sun公司核心技术丛书)

书名原文: EJB Cookbook

ISBN 7-111-14984-X

I. E … II. ①萨… ②马… III. Java语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字 (2004) 第077163号

机械工业出版社 (北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 刘 渊

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2004年9月第1版第1次印刷

787mm×1092mm 1/16·17印张

印数: 0 001-4 000册

定价: 35.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换
本社购书热线: (010) 68326294

前言

随着日益壮大的开发队伍更多地接受Java 2平台企业版 (Java 2 Platform Enterprise Edition, J2EE), 与EJB (Enterprise JavaBeans) 相关的资源、指南、书籍和专家的数量也在不断增长。使用“Enterprise JavaBeans”这个词对一家流行的网上书店进行查询, 能找到20余本书, 而且还只包括书名中包含这个词的书籍。使用因特网搜索引擎Google (www.google.com) 能够获得大约744页[⊖]与EJB密切相关的网站链接。面对大量的知识、讲解和讨论资料, EJB开发人员经常需要花费很长时间去查找快捷的解决方案。有多少次你是翻遍了一本书或者几本书的很多页才找到所要的答案?

我们自己就有过这样解决问题的经历, 所以决定做一些事情去改善这一现状。编写本书的目的是想为许多重要问题提供解决方案资源。就像菜谱包括烹调各种美食的步进式指导一样, 本书提供了EJB问题的巧妙解决方案。在书中你会发现很多种问题的解决方案, 范围从简单的、日常的快速引用问题到使用EJB模式进行复杂设计的挑战。

本书对应的读者是至少具有一些EJB开发经验的开发人员。了解了这一点, 你就不会在本书中查找描述会话和实体bean基本构建原则的解决方案。我们假设你了解企业开发的概念和EJB编程的基础知识。本书使用的格式专门面向问题, 避免使用EJB关键词 (查看目录的内容就能证明这一点)。例如, 不要查找“设计模式”, 而要查找你要解决的实际问题 (如“改进网络性能”)。

我们编写的解决方案是相互关联的, 它们引用了其他与之相关的解决方案。我们希望读者可以拿起这本书并且能够从任何地方开始阅读。解决方案都是短小精练的, 可以在坐地铁、喝咖啡的时候或者任何有几分钟空闲的时候拿出来翻阅。

本书的一个有用的姐妹篇是《Bitter EJB》, 也是Manning出版社出版的[⊙]。它详细讲解了专门针对EJB开发的很多反模式 (antipattern), 并且提供了实用的设计途径: 如何通过研究与技术相关的最重要的问题和解决方案成为更好的程序员。

我们已经为你准备好了这本书, 现在邀请你共享我们的解决方案。你也许会从9章中进行挑选或者全部读完它们。尽情享受吧!

⊖ Google实际上可能列出123 000页搜索结果, 但是相关程度高的有744页。

⊙ 中文版即将由机械工业出版社出版。

致 谢

很多人为能在短时间内完成本书提供了帮助。我要感谢Manning出版社允许我们进行这项工作。感谢Marjan Bace和Alex Garrett引发了本书的构想。感谢Susan Capparelle和Mary Piergies帮助我们保持进度，还要感谢Helen Trimes完成了编辑工作。

再次感谢Manning出版社，我们从文字编辑Liz Welch那里得到了很大的帮助。她总是随时都能够投入紧张的工作。同样，我们还要感谢审阅者：Karl Moss、Barry Nowak、Lester Martin、Ravi Mathur、Naveen Gabrani、Jeff Sullins、Bruce Tate和Ganapathy Arunkumar。

最后，感谢Cyrus Dadgar提供了他的技术经验和洞察力。

BEN SULLINS——感谢我的妻子Jenny。我得到了聪明、风趣和性感的她永恒的支持，而且我将和她永远生活在一起。感谢上帝赐予我如此完美的家庭。最后，我要感谢我的合著者Mark，感谢他的努力工作并给予我完成本书的动力。

MARK WHIPPLE——感谢我的家庭忍受我夜间的工作和在度假期间仍呆在笔记本电脑前。没有他们的耐心和支持，完成本书是不可能的。感谢我的父母，在我的生命中他们的支持始终是我获得成功所必需的动力。我要特别感谢我的合著者Ben。他充沛的热情和对成功的信心是对我的激励，并且使我集中精力编写本书。

关于本书

除了两个解决方案以外，本书中的所有解决方案都是基于EJB 2.0规范的。创建EJB Web服务端点和EJB计时器服务的解决方案基于EJB 2.1规范。在几章中的适当地方讲解了其他技术或者框架。实际上，本书使用XDoclet、log4j、Ant和Cactus。例如，第8章讨论在EJB应用中使用log4j提供日志记录功能。在涉及外部框架或者工具的时候，只讨论与EJB有关的部分。因此尽管提供了介绍，你也需要参考其他文档来解决更进一步的问题。

本书是为有经验的EJB开发人员而编写的。也就是说，我们没有提供我们认为读者已经知道的解决方案。比如你不会在本书中发现描述EJB结构、构造或者部署的基础原则。很多解决方案给出了完整的范例，而其他可能只给出了伪代码或者部分范例。很多代码都可以从本书的站点(www.manning.com/sullins2)下载。

章目录

你可以选择从头到尾地阅读本书，也可以着重阅读所需要的特定主题。下面给出每章的简短介绍，以便读者可以按照自己的意愿进行选择。我们建议读者重点阅读在日常EJB开发工作中面临的问题的解决方案。各章都不是在前一章的基础之上建立的，所以无需按照顺序阅读。本书一定会是你书架上可以找到的先读最后一页而不会导致任何不便的书籍之一。

第1章 客户代码

第1章提供处理EJB客户遇到的问题的解决方案，其中的范例演示如何通过servlet、applet和Java服务器页面(JavaServer Pages)使用EJB。这一章还讨论在调用EJB的方法之前，如何使用EJB元数据获得关于EJB的信息。

第2章 使用XDoclet生成代码

第2章讲解通过引入和使用一种开放源代码的工具(称为XDoclet)生成代码。XDoclet正在迅速地成为很多EJB开发人员选择的代码生成工具。在这一章中，你将掌握如何使用XDoclet根据bean实现类生成文件，以便使所有EJB文件保持同步。XDoclet会生成部署描述文件、接口和其他类，比如值对象。

第3章 处理数据

第3章讨论当EJB处理数据时发生的最紧迫的问题，重点讲解对会话bean、bean管理的持久性实体bean以及容器管理的持久性实体bean使用Java数据库连接(Java Database Connectivity, JDBC)。此外还提供了使用实体bean的最佳方式，以及把大型结果集合返回给客户的解决方案。

第4章 EJB的活动

第4章提供一组处理EJB日常活动的解决方案。它们和其他章中的解决方案没有非常密切的联系，你会发现有一些很独特，比如用消息驱动bean创建异步过程以及把EJB与业务服务类实现隔离开。

第5章 事务

第5章处理与事务相关的问题。你会找到处理容器管理和bean管理事务的解决方案。伴随着创建事务和传播事务，这一章还展示处理回滚和从错误中恢复的解决方案。

第6章 消息

第6章重点讲解EJB应用的消息方面的问题。这一章包括使用Java消息服务（Java Message Service, JMS）和消息驱动bean的解决方案。你会学到如何把不同类型的JMS消息发送到不同类型的目的地。这一章中还有把业务逻辑封装到消息驱动bean中的原则，以及处理错误和回滚的原则。

第7章 安全

第7章讨论通过使用EJB容器提供的安全机制解决安全问题。另外还包括保护实体bean和消息驱动bean的安全原则。这一章重点介绍EJB规范描述的EJB声明式安全模型，并且描述每个EJB可用的各种与安全 and 身份相关的方法。

第8章 日志记录

第8章讨论一种开放源代码的日志记录工具log4j。我们介绍了如何安装log4j以及把log4j合并到企业应用中。在导论部分之后，这一章提供对EJB应用进行日志记录的相关问题的解决方案，其中包括在集群中集中记录日志以及在多客户环境中将日志记录消息分类。

第9章 部署和单元测试

第9章的两个重点是Apache Ant和Apache Cactus。几个解决方案主要介绍使用Ant构造EJB类文件、生成存根以及把类文件打包在适当的JAR文件中。另一部分重点讲解使用Cactus对EJB提供的功能进行单元测试。这一章还提供对Cactus的介绍以及对实体和会话bean进行测试的解决方案。

附录A 相关解决方案

这个附录以交叉引用格式给出解决方案标题。这使你可以很容易地查找任何解决方案以及与其相关的解决方案。使用这种方式，你可以找到与本书中任何主题相关的所有适用信息。

附录B 附加资源

这个简短的附录列出了一些资源，帮助你获得书中提到的某些主题的更多信息。因为本书没有为提到的很多概念提供详细的指南，所以这个附录很重要。你可以通过这里列出的资源获得更加深入的讲解和研究资料。

代码

源代码可以从Manning的网站 (www.manning.com/sullins2) 免费获得。很多解决方案只包含演示一个要点的代码片断，并且这些片断不出现在可供下载的源代码中。能够下载的资源是那些包含完整代码的范例。在某些情况下，完整的源代码不出现在解决方案中。

在我们给出源代码时，有时候使用粗体来引起对特殊部分的注意。在本书中，**Courier** 字体用于表示代码 (XML、Java和HTML)，还表示Java方法和其他源代码标识符：

- 对方法的引用通常不包括签名，因为方法调用的形式可能不只一种。
- 对XML元素的引用包括括号，但是不包括属性或者封闭的标记 (`<action>`)。

如何使用本书

本书并不是一个故事——它没有开头或者结尾，不需要按照顺序去阅读。我们建议使用附录A或者目录查找你感兴趣的主题并且直接跳到那里。在目录和附录A中都能找到解决方案的标题。本书的出色之处在于能够在几分钟内讲述解决方案。我们希望它成为快捷的解决方案参考，而不是指导性的指南。

作者在线

读者在使用本书的同时，可以免费访问Manning出版社为本书专设的Web论坛，在论坛中可以评论书籍、提出技术问题以及从作者和其他用户那里获得帮助。具体网址是www.manning.com/sullins2。网页上的信息解释了如何在注册之后登录论坛、可以获得什么样的帮助以及论坛的管理规则。

目 录

前言
致谢
关于本书
作者在线

第一部分 导 论

第1章 客户代码	3
1.1 从另一个EJB调用本地EJB	3
1.2 从另一个EJB调用远程EJB	5
1.3 从servlet访问EJB	7
1.4 从Java服务器页面调用EJB	10
1.5 从JMS系统调用EJB业务逻辑	13
1.6 将EJB实例的引用持久化	15
1.7 获得并且使用持久化的EJB引用	16
1.8 将主对象引用持久化	18
1.9 比较两个EJB引用是否相同	20
1.10 对EJB使用反射	21
1.11 从applet调用EJB	23
1.12 改善客户端的EJB查找代码	26
第2章 使用XDoclet生成代码	29
2.1 生成主、远程、本地和本地主接口	32
2.2 为主接口添加和定制JNDI名称	37
2.3 使EJB部署描述文件保持最新	38
2.4 为实体bean创建值对象	40
2.5 生成主键类	45
2.6 避免硬编码的XDoclet标记值	48
2.7 使用工具对象帮助进行bean查找	49
2.8 生成供应商特有的部署描述文件	53
2.9 在bean源代码中指定安全角色	54
2.10 生成和维护方法权限	55

2.11 为实体主接口生成查找器方法	56
2.12 生成ejbSelect方法XML	57
2.13 把主方法添加到生成的主接口	58
2.14 把实体关系XML添加到部署描述文件	59
2.15 把目的地类型添加到消息驱动bean的部署描述文件	60
2.16 把消息选择器添加到消息驱动bean的部署描述文件	61

第二部分 主体部分

第3章 处理数据	65
3.1 使用数据源	65
3.2 创建EJB 2.0容器管理的持久性	68
3.3 对不同的用户使用不同的数据源	71
3.4 使用数据库序列为实体bean生成主键值	73
3.5 为实体bean使用组合主键	76
3.6 在单一步骤中获得多个实体bean	79
3.7 建立一对一实体数据关系的模型	81
3.8 为实体bean创建一对多的关系	84
3.9 使用实体关系创建层叠删除	87
3.10 开发不能创建的、只读的实体bean	89
3.11 从EJB调用存储过程	91
3.12 使用EJB-QL创建定制的查找器方法	93
3.13 把实体数据持久化到数据库视图中	96
3.14 基于实体数据的变化发送通知	97
3.15 为实体数据创建接口	100
3.16 获得实体数据集合的信息	102
3.17 减少调用实体bean的次数	104

3.18 在大型结果集中分页	105
第4章 EJB的活动	111
4.1 获得环境变量	111
4.2 实现EJB的toString() 功能	113
4.3 为所有EJB提供通用方法	114
4.4 减少未实现的bean 方法造成的混乱	115
4.5 从EJB发送电子邮件	119
4.6 使用EJB 2.1的计时器服务	121
4.7 从EJB发送JMS消息	122
4.8 把EJB当作Web服务使用	124
4.9 创建EJB客户的异步行为	125
4.10 不使用消息驱动bean创建异步行为	129
4.11 隔离EJB和服务类实现	131
4.12 创建批处理机制	132
第5章 事务	135
5.1 调整EJB的容器事务控制	137
5.2 不使用容器处理事务的管理	139
5.3 回滚当前的事务	140
5.4 试图使用错误恢复避免回滚	142
5.5 在方法完成之前强制回滚	144
5.6 对事务加以时间限制	145
5.7 把实体更新组合到单一事务中	146
5.8 在事务边界管理EJB状态	147
5.9 在一个方法中使用多个事务	148
5.10 在回滚之后管理EJB状态	150
5.11 抛出异常且不导致回滚	151
5.12 把事务传播到另一个EJB业务方法	152
5.13 把事务传播到非EJB类	154
5.14 在客户层启动事务	155
5.15 在多个Java服务器页面之间维持事务	157
5.16 在一个事务中更新多个数据库	158
第6章 消息	161
6.1 发送公布/预订JMS消息	161
6.2 发送点对点JMS消息	163
6.3 创建消息驱动的EJB	164

6.4 按照先进先出(FIFO)方式从消息 队列中处理消息	167
6.5 把消息驱动bean和业务逻辑改动隔 离开	170
6.6 把数据流发送给消息驱动EJB	171
6.7 使用单一JMS消息触发两个或者更 多的消息驱动bean	174
6.8 加速消息驱动bean的消息传送	176
6.9 为消息驱动EJB过滤消息	178
6.10 在消息驱动EJB中封装错误处理代码	180
6.11 异步发送电子邮件消息	182
6.12 在消息驱动bean中处理回滚	183
第7章 安全	187
7.1 在EJB方法内找到调用者的身份和 角色	188
7.2 分配和确定EJB客户安全角色	189
7.3 把客户凭证传递给EJB容器	190
7.4 对特定用户禁用方法	191
7.5 为EJB分配角色	194
7.6 防止对实体数据的访问	195
7.7 使用EJB处理对LDAP源的简单验证	196
7.8 保证消息驱动bean的安全	197

第三部分 补 充

第8章 日志记录	203
8.1 对日志消息进行格式化	206
8.2 改进日志性能	208
8.3 使用日志生成报告	210
8.4 把日志消息发送给JMS主题	211
8.5 将日志记录到XML文件	212
8.6 为Web浏览器创建日志文件视图	214
8.7 在集群环境中创建集中化的日志 文件	215
8.8 跟踪EJB的生命周期	217
8.9 在运行时使用不同的配置	219

8.10 按照客户将日志消息分类	221	9.5 创建有状态会话bean的单元测试	236
第9章 部署和单元测试	223	9.6 创建实体bean的单元测试	238
9.1 编译EJB	226	9.7 测试案例执行的自动化	240
9.2 构造ejb.jar文件	229	9.8 使用UI执行测试案例	244
9.3 构造EJB的存根类	231	附录A 相关解决方案	247
9.4 创建无状态会话bean的单元测试	234	附录B 附加资源	259

第一部分

导 论

本书的第一部分讨论EJB (Enterprise JavaBeans) 开发过程中的两个“前端”问题。第1章讨论EJB客户，这是很多EJB应用的前端。第2章介绍本书中使用的第一个开放源代码的工具——XDoclet，用它来生成与EJB相关的文件和源代码。

第1章讲解EJB客户很可能会遇到的各种问题。在这一章中，你会看到处理applet、servlet和作为客户的其他EJB的解决方案，还包括几个超越客户类型并且可以应用到很多客户情况中的问题的解决方案。第2章详细介绍XDoclet。XDoclet是一种开放源代码的工具，它提供生成EJB源代码和维护XML部署描述文件的功能。这一章考察了很多与EJB开发工作相关的XDoclet功能。

“嘿，朋友！我喝酒，吃干酪，还消磨时光。”

——唐纳德·萨瑟兰饰演的Oddball，出自电影《战略大作战》

EJB客户是与EJB有联系的任何对象。为了让这种联系起作用，EJB需要了解如何彼此找到对方。本章中的解决方案提供了能够让EJB相互定位的方法。

EJB客户可以分成两大类：远程客户和本地客户。这是个非常重要的划分，因为它影响客户查找需要使用的EJB的方式。例如，创建InitialContext实例来查找同一虚拟机中的EJB和为查找远程EJB而创建实例是完全不同的。本章中讲解的客户包括Java服务器页面（JavaServer Pages，JSP）、servlet、applet和其他EJB。每种客户都有其自己的问题集合。

在本章中，你会看到下面这些任务的解决方案：

- 从另一个EJB调用本地EJB的方法
- 从另一个EJB调用远程EJB的方法
- 从servlet与EJB取得联系
- 从JSP与EJB取得联系
- 使用Java消息服务（Java Message Service，JMS）调用EJB业务逻辑
- 以一种持久性格式保存EJB引用
- 使用持久的EJB引用
- 存储EJB主对象（home object）引用
- 判断两个EJB引用是否引用相同的bean
- 使用反射获得EJB方法的信息
- 从applet使用EJB
- 改进EJB查找

1.1 从另一个EJB调用本地EJB

◆问题

希望从一个EJB调用同一个EJB容器内的另一个EJB的方法。

◆背景

在大多数EJB应用中，在一个EJB内完成业务逻辑要涉及到调用另一个EJB。另外，实体bean的访问通常通过会话bean外观。当创建EJB应用时，了解如何与同一个EJB容器内的其他EJB进行联系是很重要的。解决这个问题需要了解如何找到同一个容器内的另一个bean。

◆解决

为了调用同一个容器内的另一个EJB，在客户EJB中需要添加一个方法来查找所需的EJB的主对象。例如，代码1-1中的方法查找EJB PasswordBean的主对象PasswordHome。

代码1-1 简单的查找方法

```
private PasswordHome getPasswordHome() {
    try {
        Context ic = new InitialContext();
        PasswordHome passwordHome = ( PasswordHome )
            ic.lookup("ejbAccess.passwordHome");
        return passwordHome;
    }
    catch (NamingException ne) {
        return null;
    }
}
```

创建默认的InitialContext实例

执行EJB主对象的JNDI查找

◆讨论

使用主对象的引用可以调用接口声明的任何方法。通常情况下，可以调用create()方法获得EJB（远程的或者本地的）业务接口的引用，以便执行业务逻辑或者数据访问。

当然，查找主对象只是使用EJB的一部分。获得主对象之后，就可以创建或者定位EJB实例以便使用（这取决于bean的类型）。EJB 2.0规范的完成引入了本地接口（local interface）的概念。通常情况下，如果知道EJB将只用在（或者应该只用在）本地EJB容器中（就是说没有远程客户），那么EJB就应该实现本地主接口和本地业务接口。这与主接口和远程接口是一样的，只是本地接口没有通过Java远程方法调用（Java Remote Method Invocation, RMI）暴露，并且只能用于同一个虚拟机中。使用本地接口改善了本地客户使用的EJB的性能，并且加强了安全性。

◆参见

1.2——从另一个EJB调用远程EJB

- 1.12——改善客户端的EJB查找代码
- 2.1——生成主、远程、本地和本地主接口
- 2.7——使用工具对象帮助进行bean查找
- 7.6——防止对实体数据的访问

1.2 从另一个EJB调用远程EJB

◆问题

希望从一个EJB调用位于远程EJB容器中的另一个EJB的方法。

◆背景

在大多数EJB应用中，完成EJB中的业务逻辑要涉及到调用另一个EJB。另外，实体bean的访问通常是通过会话bean外观来完成的。在大多数情况下，从另一个EJB发出的EJB访问都发生在同一个EJB容器中。但是，大型企业应用可能会跨主机分割业务逻辑，或者调用另一个EJB应用来完成工作流程。在这种情况下，可能需要从其他EJB访问远程EJB。为了达到这个目的，需要了解如何创建初始的上下文来找到远程EJB容器，以便查找或者创建远程EJB。

◆解决

和1.1节中的解决方案不同，并不是只创建默认的Initialcontext实例就能调用远程EJB。必须传递一些属性，这些属性使上下文能够找到远程EJB使用的Java命名和目录接口（Java Naming and Directory Interface, JNDI）系统。例如，代码1-2中的私有方法在Weblogic EJB容器中查找远程EJB的主对象。

代码1-2 使用初始化的InitialContext的查找方法

```
private EquityHome getEquityHome() {
    try {
        // Get an InitialContext
        Properties props = new Properties();
        props.put( Context.INITIAL_CONTEXT_FACTORY, "weblogic.jndi.WLInitialContextFactory" );
        props.put( Context.PROVIDER_URL, "http://MyRemoteHost:7001" );
        Context ic = new InitialContext( props );
        EquityHome equityHome = ( EquityHome )
            ic.lookup( "ejbAccess.equityHome" );
        equityHome = ( EquityHome )
            PortableRemoteObject.narrow( equityHome,
            EquityHome.class );
    }
}
```

创建带有环境属性的InitialContext实例

缩减返回值