

<<精通EJB>>

图书基本信息

书名：<<精通EJB>>

13位ISBN编号：9787505370913

10位ISBN编号：750537091X

出版时间：2002-1-1

出版时间：电子工业出版社

作者：Ed Roman

页数：566

译者：王进亮

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<精通EJB>>

### 内容概要

本书是关于EJB开发的经典书籍，是EJB开发者的必备参考书。

全书共分4个部分15章，内容涵盖了EJB的概念、方法和部署，描述了构建企业级应用组件要解决的扩展性、分布式、事务处理、数据存储和安全性等问题，引导读者创建企业级可升级、安全可靠、可运行于多重平台且以商务为重点的应用程序，实现“服务器端仅写一次，即可随处运行”

（Server-Side Write Once, Run Anywhere）。

书中包含了大量的EJB实例，深入EJB内部进行阐释。

本书非常适合于Java程序员、网站开发人员和系统分析员作为EJB开发的核心参考书。

## <<精通EJB>>

### 作者简介

Ed Roman是世界上研究高端中间件技术的权威之一。

他从一开始就积极参与Sun Microsystems公司的Enterprise Bean解决方案，设计、建立、部署过很多不同的企业级应用，包括构建和开发完整的应用服务器产品。

平常他将大部分的时间用于更改和完善Sun公司的EJB规范，他是中间件邮件

## <<精通EJB>>

### 书籍目录

#### 目 录

##### 第一篇 概 述

##### 第1章 服务器端组件体系结构

###### 1.1 对服务器端组件体系结构的迫切需求

###### 1.1.1 软件组件

###### 1.1.2 组件体系结构

###### 1.1.3 对服务器端应用的需求

###### 1.1.4 多层体系结构

###### 1.2 服务器端组件体系结构解决方案

###### 1.2.1 Microsoft公司的DNA体系结构

###### 1.2.2 Sun公司的J2EE

###### 1.2.3 OMG组织的CORBA标准

###### 1.3 Java 2平台企业版

###### 1.3.1 为什么选择J2EE

###### 1.3.2 J2EE技术

###### 1.3.3 EJB

###### 1.3.4 RMI和RMI-IIOP

###### 1.3.5 JNDI

###### 1.3.6 JDBC

###### 1.3.7 JTA和JTS

###### 1.3.8 JMS

###### 1.3.9 Java Servlet和JSP

###### 1.3.10 Java IDL

###### 1.3.11 JavaMail

###### 1.3.12 连接器

###### 1.3.13 XML

###### 1.4 小结

##### 第2章 EJB概述

###### 2.1 EJB中到底有什么

###### 2.1.1 六个组成部分

###### 2.1.2 组件供应商

###### 2.1.3 EJB服务器和EJB容器供应商

###### 2.1.4 应用组装商

###### 2.1.5 EJB系统部署员

###### 2.1.6 系统管理员

###### 2.2 企业级Bean

###### 2.2.1 Bean的分类

###### 2.2.2 Bean组件多种分类的原因

###### 2.3 EJB容器和服务功能概述

###### 2.3.1 资源管理和Bean组件生存期管理

###### 2.3.2 状态管理

###### 2.3.3 事务处理

###### 2.3.4 安全性

###### 2.3.5 持久性

###### 2.3.6 远程访问和透明的资源定位

## <<精通EJB>>

2.3.7 连接代码和Bean组件安装工具

2.3.8 专用的容器特性

2.4 小结

第二篇 开发Bean

第3章 会话Bean介绍

3.1 Enterprise Bean的组成部分

3.1.1 Enterprise Bean类

3.1.2 EJB对象

3.1.3 Home对象

3.1.4 部署描述

3.1.5 特定的Bean属性

3.1.6 Ejb-jar文件

3.1.7 术语小结

3.2 什么是会话Bean

3.2.1 会话Bean的生命周期

3.2.2 对话型会话Bean与非对话型会话Bean

3.2.3 所有会话Bean的方法都是可序列化的

3.3 如何编写会话Bean

3.3.1 setSessionContext(SessionContext ctx)方法

3.3.2 ejbCreate(...)方法

3.3.3 ejbPassivate()方法

3.3.4 ejbActivate()方法

3.3.5 ejbRemove()方法

3.3.6 商务方法

3.4 如何调用会话Bean

3.4.1 查找Home对象

3.4.2 创建EJB对象

3.4.3 调用方法

3.4.4 清除EJB对象

3.5 小结

第4章 无状态会话Bean基础

4.1 无状态会话Bean特点

4.1.1 不存在对话状态

4.1.2 只有一种方法初始化无状态会话Bean

4.1.3 EJB容器收集并重用无状态会话Bean

4.1.4 EJB对象的decoupling

4.2 编写“Hello,World!”无状态会话Bean

4.2.1 创建“Hello,World!”Remote接口

4.2.2 实现“Hello,World!”Bean

4.2.3 创建“Hello,World!”Home接口

4.2.4 编写部署描述

4.2.5 环境属性

4.2.6 Ejb-jar文件

4.2.7 部署Bean

4.2.8 为无状态会话Bean编写客户端代码

4.2.9 运行系统

4.3 小结

## <<精通EJB>>

### 第5章 状态会话Bean基础

#### 5.1 状态会话Bean特点

##### 5.1.1 状态会话Bean具有收集功能

##### 5.1.2 对话状态遵从的规则

##### 5.1.3 激活/挂起回调方法

##### 5.1.4 一个简单的状态会话Bean

#### 5.2 状态还是无状态

#### 5.3 小结

### 第6章 给Bean增加功能

#### 6.1 EJB Context——通向EJB容器之门

#### 6.2 会话Bean Context

#### 6.3 理解EJB的安全性

##### 6.3.1 第1步：验证

##### 6.3.2 第2步：授权

##### 6.3.3 安全性的传播

#### 6.4 理解EJB对象句柄

#### 6.5 智力游戏“Fazuul”

##### 6.5.1 Fazuul是什么

##### 6.5.2 以EJB角度详细说明Fazuul

##### 6.5.3 通过环境属性使游戏成为数据驱动型游戏

##### 6.5.4 Fazuul游戏的实现

#### 6.6 运行客户端应用程序

##### 6.6.1 检验EJB对象句柄

##### 6.6.2 检验安全性

#### 6.7 小结

### 第7章 实体Bean介绍

#### 7.1 持久的概念

##### 7.1.1 Java对象序列化

##### 7.1.2 对象-关系型数据库映射

##### 7.1.3 对象数据库

#### 7.2 什么是实体Bean

#### 7.3 实体Bean的特征

##### 7.3.1 实体Bean是长久存在的

##### 7.3.2 实体Bean能够避免Bean失效

##### 7.3.3 实体Bean实例是一个对应到数据库中的视图

##### 7.3.4 几个实体Bean实例可能代表同一底层数据

##### 7.3.5 实体Bean实例可以被收集再利用

##### 7.3.6 保持实体Bean的两种方法

##### 7.3.7 实体Bean可以被创建、删除或查询

##### 7.3.8 实体Bean可以代表遗留下来的数据和系统

##### 7.3.9 实体Bean可以不通过EJB被修改

#### 7.4 开发和使用实体Bean

##### 7.4.1 实体Bean的创建：ejbCreate()

##### 7.4.2 查寻已存在的实体Bean：ejbFind()

##### 7.4.3 实体Bean的删除：ejbRemove()

#### 7.5 实体上下文

##### 7.5.1 getEJBObject()

## <<精通EJB>>

7.5.2 getPrimaryKey()

7.6 实体Bean的生存周期

7.7 小结

第8章 编写Bean管理的持久实体Bean

8.1 实现Bean管理持久的指导方针

8.2 Bean管理持久的范例：一个银行账户

8.2.1 Account.java

8.2.2 AccountHome.java

8.2.3 AccountPK.java

8.2.4 AccountBean.java

8.2.5 AccountException.java

8.2.6 Client.java

8.2.7 部署描述

8.2.8 环境属性

8.2.9 建立数据库

8.3 运行客户端程序

8.3.1 服务器端的输出

8.3.2 客户端的输出

8.4 小结

第9章 编写容器管理的持久实体Bean

9.1 容器管理的域

9.2 主键类

9.3 实现容器管理持久的指导方针

9.4 容器管理持久的范例：产品系列

9.4.1 Product.java

9.4.2 ProductHome.java

9.4.3 ProductPK.java

9.4.4 ProductBean.java

9.4.5 Client.java

9.4.6 部署描述

9.5 运行客户端程序

9.5.1 服务器端的输出

9.5.2 客户端的输出

9.6 承诺和现实：Bean管理的持久与容器管理的持久

9.6.1 承诺：容器管理的持久会减少代码量

9.6.2 承诺：容器管理的持久会减少错误

9.6.3 承诺：容器管理的持久使连接到不同的数据库变得容易

9.7 解决EJB调试问题

9.8 小结

第三篇 EJB的高级概念

第10章 事务

10.1 事务产生的动机

10.1.1 原子操作

10.1.2 网络故障或机器故障

10.1.3 多用户共享数据

10.2 事务所带来的益处

10.3 事务的模型

## <<精通EJB>>

- 10.3.1 平面式事务模型
- 10.3.2 嵌套式事务模型
- 10.3.3 其他事务模型
- 10.4 事务在EJB组件中的应用
  - 10.4.1 事务模型的支持
  - 10.4.2 后台事务系统的提取
  - 10.4.3 声明型事务和编程型事务
  - 10.4.4 控制EJB组件加入事务的方式
  - 10.4.5 EJB事务的属性值
- 10.5 事务的隔离
  - 10.5.1 并发控制的需求
  - 10.5.2 隔离性与EJB
  - 10.5.3 不当读取问题
  - 10.5.4 不可重复读取问题
  - 10.5.5 幻影读取问题
  - 10.5.6 事务隔离性的总结
- 10.6 分布式事务
  - 10.6.1 稳定性和两阶段提交协议
  - 10.6.2 事务的通信协议和事务上下文
- 10.7 EJB中的编程型事务
  - 10.7.1 CORBA的对象事务处理服务
  - 10.7.2 Java事务处理服务
  - 10.7.3 Java事务处理API
  - 10.7.4 声明型事务和编程型事务的应用范例比较
  - 10.7.5 如何使用客户代码控制事务
- 10.8 在EJB中设计事务会话
- 10.9 小结
- 第11章 CORBA和RMI-IIOP
  - 11.1 何谓CORBA
  - 11.2 我们为何要关心CORBA
    - 11.2.1 CORBA的优点
    - 11.2.2 CORBA的缺点
  - 11.3 理解CORBA是如何工作的
    - 11.3.1 对象请求代理程序
    - 11.3.2 对象实现和对象引用
    - 11.3.3 对象适配器
    - 11.3.3 存储器
  - 11.4 OMG的接口语言
    - 11.4.1 将OMG IDL映射成具体的语言
    - 11.4.2 静态调用
    - 11.4.3 动态调用
  - 11.5 CORBA提供的众多服务
    - 11.5.1 命名服务
    - 11.5.2 事件服务
    - 11.5.3 对象事务服务
    - 11.5.4 并发控制服务
    - 11.5.5 安全服务



## <<精通EJB>>

- 11.5.6 CORBA组件
- 11.6 基于IIOP的RMI
  - 11.6.1 RMI-CORBA协同工作的需求
  - 11.6.2 RMI与CORBA的结合
  - 11.6.3 基于IIOP的RMI的优点
  - 11.6.4 RMI与IIOP结合所带来的问题
- 11.7 RMI与CORBA的协同工作综述
  - 11.7.1 带有CORBA对象实现的RMI-IIOP客户
  - 11.7.2 带有RMI-IIOP对象实现的CORBA客户
- 11.8 RMI-IIOP API
  - 11.8.1 javax.rmi
  - 11.8.2 基于IIOP的RMI的引导
- 11.9 理想的蓝图：CORBA与EJB的结合
  - 11.9.1 CORBA/EJB协同工作的优点
  - 11.9.2 CORBA/EJB协同工作的情景
  - 11.9.3 范例代码
  - 11.9.4 CORBA EJB客户范例
- 11.10 小结
- 第四篇 一个复杂的电子商务部署
- 第12章 设计一个电子商务对象模型
  - 12.1 最终产品的预览
  - 12.2 技术要求
    - 12.2.1 商务逻辑层的对象模型
    - 12.2.2 表达层的对象模型
  - 12.3 小结
- 第13章 实现电子商务实体Bean
  - 13.1 顾客实体Bean
    - 13.1.1 Customer.java
    - 13.1.2 CustomerBean.java
    - 13.1.3 CustomerHome.java
    - 13.1.4 CustomerPK.java
    - 13.1.5 部署描述
  - 13.2 订单条目实体Bean
    - 13.2.1 OrderLineItem.java
    - 13.2.2 OrderLineItemBean.java
    - 13.2.3 处理实体Bean引用
    - 13.2.4 OrderLineItemHome.java
    - 13.2.5 OrderLineItemPK.java
    - 13.2.6 部署描述
  - 13.3 订单实体Bean
    - 13.3.1 Order.java
    - 13.3.2 OrderBean.java
    - 13.3.3 OrderHome.java
    - 13.3.4 OrderPK.java
    - 13.3.5 OrderException.java
    - 13.3.6 部署描述
  - 13.4 小结

## <<精通EJB>>

### 第14章 实现电子商务会话Bean

#### 14.1 购物篮条目状态会话Bean

##### 14.1.1 QuoteLineItem.java

##### 14.1.2 QuoteLineItemBean.java

##### 14.1.3 QuoteLineItemHome.java

##### 14.1.4 部署描述

#### 14.2 购物篮状态会话Bean

##### 14.2.1 Quote.java

##### 14.2.2 QuoteBean.java

##### 14.2.3 QuoteHome.java

##### 14.2.4 QuoteException.java

##### 14.2.5 部署描述

#### 14.3 无状态计价会话Bean

##### 14.3.1 Pricer.java

##### 14.3.2 PricerBean.java

##### 14.3.3 PricerHome.java

##### 14.3.4 PricerException.java

##### 14.3.5 部署描述

#### 14.4 无状态银行出纳员会话Bean

##### 14.4.1 Teller.java

##### 14.4.2 TellerHome.java

##### 14.4.3 TellerBean.java

##### 14.4.4 TellerException.java

##### 14.4.5 部署描述

#### 14.5 小结

### 第15章 将Servlet连接到EJB

#### 15.1 Servlet在EJB部署中的角色

#### 15.2 实现Servlet

#### 15.3 Web店面Servlet

#### 15.4 在线目录Servlet

#### 15.5 商品概况Servlet

#### 15.6 商品详情Servlet

#### 15.7 购物篮查看Servlet

#### 15.8 购买Servlet

#### 15.8 Servlet属性

#### 15.9 运行完整的电子商务系统

##### 15.9.1 启动商务逻辑层

##### 15.9.2 启动表达层

##### 15.9.3 启动瘦客户

#### 15.10 优化及设计策略

##### 15.10.1 生成唯一的主键

##### 15.10.2 减少网络往返路程：消极调用Bean

##### 15.10.3 标识实体Bean的同步需求

##### 15.10.4 实体Bean与会话Bean的设计

##### 15.10.5 设计精巧的和设计粗糙的实体Bean

##### 15.10.6 查找大量的实体Bean

#### 15.11 小结

## <<精通EJB>>

### 附 录

#### 附录A 理解Java远程方法调用

##### A.1 远程方法调用

##### A.2 RMI体系结构

##### A.2.1 RMI和接口与实现

##### A.2.2 根程序和程序框架

##### A.2.3 根连接和RMI注册

##### A.2.4 RMI URL

##### A.2.5 查找远程对象

##### A.3 RMIC——RMI编译器

##### A.4 对象序列化和参数传递

##### A.4.1 值传递

##### A.4.2 对象序列化

##### A.4.3 对象是怎样序列化的

##### A.4.4 怎样标记暂态

##### A.4.5 使用对象序列化

##### A.4.6 RMI怎样模拟通过引用传递参数

##### A.4.7 输出远程对象

##### A.5 一个简单的例子

##### A.5.1 IFlip接口

##### A.5.2 Flip类

##### A.5.3 客户

##### A.5.4 编译这个程序

##### A.5.5 运行该程序

##### A.6 处理远程异常

##### A.6.1 RMI远程异常的限制

##### A.6.2 RMI技巧

##### A.7 高级RMI

##### A.7.2 RMI例子：用于分布式记录的消息队列

##### A.7.3 RMI例子：开发分布式垃圾收集器

##### A.8 RMI、CORBA和EJB

##### A.9 RMI和EJB

##### A.9.1 EJB对象

##### A.9.2 本地对象

##### A.10 小结

#### 附录B 了解Java命名和目录接口JNDI

##### B.1 命名和目录服务器

##### B.2 进入JNDI

##### B.3 JNDI概述

##### B.3.1 JNDI的结构体系

##### B.3.2 现有的服务技术类型

##### B.4 了解JNDI程序背后的概念

##### B.5 JNDI的程序设计

##### B.5.1 安装必需的软件

##### B.5.2 JNDI 包

##### B.5.3 探索Javax.naming包

##### B.5.4 实例：一个交互式浏览器

## <<精通EJB>>

B.6 高级JNDI：JNDI与JDBC的结合

B.7 高级JNDI：JNDI和EJB相结合

B.7.1 JNDI和EJB主对象

B.7.2 JNDI的其他用途

B.8 高级JNDI：连接JNDI和Java RMI

B.8.1 IFlip.java

B.8.2 Flip.java

B.8.3 FlipClient.java

B.8.4 运行这个例子

B.9 小结

附录C 了解XML

C.1 XML的商业需要

C.1.1 电子商务

C.1.2 现有技术的不足

C.2 XML

C.2.1 XML的优点

C.2.2 XML和EDI的比较

C.2.3 XML和SGML的比较

C.2.4 XML和HTML 的比较

C.3 XML的概念

C.3.1 Prolog

C.3.2 XML元素

C.3.3 XML实体

C.3.4 CDATA片段

C.3.5 格式正确的文档

C.3.6 XML DTD

C.3.7 XML小结

C.4 XML和EJB

C.4.1 EJB部署描述

C.4.2 作为在线数据格式的XML

C.5 小结

附录D 了解EJB 1.1

D.1 可移植的部署描述

D.1.1 Ejb-Jar文件的部署描述

D.1.2 EJB 1.1 XML部署描述的例子

D.2 实体Bean的授权支持

D.3 RMI-IIOP API标准

D.4 JNDI的各个方面

D.4.1 如何从客户代码中查找主对象

D.4.2 如何从Bean中查找主对象

D.4.3 如何查找资源工厂

D.4.4 如何查询环境属性

D.4.5 如何查找JTA UserTransaction接口

D.5 正确的Bean引用

D.5.1 钝化和活化的提高

D.5.2 持久性的改进

D.6 事务处理程序的阐明和提高

## <<精通EJB>>

- D.6.1 实体Bean必须使用容器管理的事务处理
- D.6.2 陈述性事务处理程序的变化
- D.7 安全更新
  - D.7.1 安全上下文传递的变化
  - D.7.2 Java 2的安全模式
  - D.7.3 在EJB 1.1系统中加入程序性的安全措施步骤
  - D.7.4 在EJB 1.1系统中加入说明性的安全措施步骤
- D.8 新的主句柄
- D.9 EJB 1.1里的其他重要变化
- D.10 更多的信息
- D.11 EJB 1.1以后的发展
- D.12 小结
- 附录E 购买EJB产品
  - E.1 符合EJB规范
  - E.2 支持实体Bean
  - E.3 持久性
    - E.3.1 Bean自己管理的持久性
    - E.3.2 容器管理持久性
  - E.4 选择基于CORBA与RMI的EJB产品
  - E.5 RMI-IIOP API的使用
  - E.6 经JNDI的JTA可存取性
  - E.7 协议问题
  - E.8 集成层次的支持
  - E.9 内存数据的高速缓存
  - E.10 可测量性
  - E.11 高度的实用性
  - E.12 安全性
  - E.13 IDE集成
  - E.14 智能负荷平衡
  - E.15 无状态的透明的结束出错
  - E.16 分组
  - E.17 合理地关机
  - E.18 实时部署
  - E.19 分布式事务处理
- E. 现有企业系统的集成
  - E.21 异步通信的支持
  - E.22 具有Bean厂家的集成
  - E.23 专业化的服务
  - E.24 非技术标准
  - E.25 小结
- 附录F EJB快速索引向导
  - F.1 会话Bean的图表
  - F.2 状态会话Bean的图表
  - F.3 实体Bean的图表
  - F.4 EJB API索引
    - F.4.1 CreateException
    - F.4.2 DuplicateKeyException

## <<精通EJB>>

- F.4.3 EJBContext
- F.4.4 EJBException
- F.4.5 EJBHome
- F.4.6 EJBMetaData
- F.4.7 EJBObject
- F.4.8 EnterpriseBean
- F.4.9 EntityBean
- F.4.10 EntityContext
- F.4.11 FinderException
- F.4.12 Handle
- F.4.13 HomeHandle
- F.4.14 NoSuchEntityException
- F.4.15 ObjectNotFoundException
- F.4.16 RemoveException
- F.4.17 SessionBean
- F.4.18 SessionContext
- F.4.19 SessionSynchronization
- F.5 事务索引

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>