

Java 开发专家

精通

SOA:

基于服务总线的 Struts+EJB+Web Service 整合应用开发

揭示SOA架构的核心思想与实现方法

展现当前最主流的Java技术与产品在SOA开发中的要领

基于Eclipse的Struts开发

基于Eclipse + Lombok的EJB 开发

基于Eclipse + Axis的Web Service开发

基于IBM Websphere RAD的EJB和Web Service开发

基于IBM Websphere 6.0 SIBus服务总线的SOA架构整合应用开发

基于BEA WebLogic 9.1 AquaLogic服务总线的SOA架构整合应用开发



梁爱虎

飞思科技产品研发中心 编著

监制



光盘内容包含书中实例源程序，以及将源程序、开发环境和运行环境集成在一起的Workspace。



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

Java 开发专家

精通 SOA:

基于服务总线的 Struts+EJB+Web Service 整合应用开发

梁爱虎

飞思科技产品研发中心

编著

监制



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

· 内 容 简 介 ·

Java 的出现,实现了跨操作系统平台的程序开发,以 Java 为基础的 J2EE 技术已经成为因特网服务技术的主流。然而,以 J2EE 为基础的 SOA 架构技术必将成为因特网服务技术的热点。笔者希望通过本书帮助广大 Java 开发人员快速从以技术为导向的编程转到以服务为导向的编程,以适应这个新到来的 SOA 时代。

本书通过实例,由浅入深、全方位地介绍了基于 Eclipse 和 Lomboz 插件实现 Struts 的 Web 应用开发,基于 Eclipse 和 Lomboz 插件实现 EJB 的 Session Bean 的开发及实现 Struts + Session Bean 的整合应用编程,基于 Eclipse 和 Lomboz 插件实现 EJB 的 CMP 的开发(包括创建复合主键和新的 finder 方法,以及实现 Struts + Session Bean + CMP 的整合应用编程),基于 Eclipse 和 Axis 实现 Web Service 的开发及实现 Struts + Session Bean + CMP + Web Service 的整合应用编程,基于 IBM Webshpere RAD 的 EJB Session Bean、CMP 和 Web Service 的应用开发,并且详细介绍了基于 BEA AquaLogic 服务总线实现 SOA 架构的整合应用开发。

最后通过一个网上书店的实例详细介绍了如何基于 IBM SIBus 服务总线实现 SOA 架构的 Struts、EJB 和 Web Service 的 J2EE 整合应用开发。

本书适合于广大的 Java 开发人员,特别是那些希望尽快掌握实际项目所需要的开发经验的人员,也适合于相关专业的广大学生和教师。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

精通 SOA: 基于服务总线的 Struts+EJB+Web Service 整合应用开发 / 梁爱虎编著. —北京:
电子工业出版社, 2007.1
(Java 开发专家)
ISBN 7-121-03448-4

I.精... II.梁... III.Java 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2006)第 135153 号

责任编辑:王树伟

印 刷:北京智力达印刷有限公司

装 订:北京中新伟业印刷有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:26.25 字数:672 千字

印 次:2007 年 1 月第 1 次印刷

印 数:5 000 册 定 价:46.00 元 (含光盘 1 张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,
联系电话:(010) 68279077; 邮购电话:(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

梁爱虎

IBM高级IT专家，博士。

2000年2月—2000年7月在美国硅谷Netpace公司任软件工程师，从事基于J2EE的网上知识竞赛的项目设计开发。

2000年7月—2000年12月在美国硅谷AngelEngineers公司任高级软件工程师，从事基于J2EE的数据挖掘Eyemine项目的设计开发。

2001年1月—2001年9月在硅谷Octasoft公司任高级软件工程师，从事基于J2EE的网上电子银行系统项目的设计开发。

2004年12月至今在中国IBM ISSC公司任高级IT专家(Advisory IT Specialist)。

笔者感言：

SOA架构实现了业务和技术的完全分离，达到了目前软件设计所能达到的最高境界，使现代企业能够迅捷应对瞬息万变的市场需求。我们可以体察到，SOA正在成为现代企业和广大IT人员的最佳选择，可以说，SOA预示着一个新的以服务为导向的IT时代的到来。



ISBN 7-121-03272-4

定价: 59.00 元



ISBN 7-121-02538-8

定价: 65.80 元

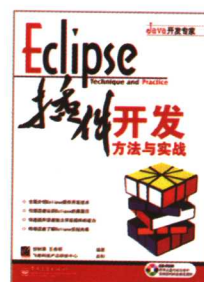


ISBN 7-121-03215-5

定价: 48.00 元



ISBN 7-121-03551-0



ISBN 7-121-03180-9

定价: 49.00 元



飞思科技

热烈祝贺“开发专家之Sun ONE”系列 被评为电子工业出版社“最佳品牌奖”

“开发专家之Sun ONE”全新提升为“Java开发专家”系列
——源自精品 成就理想



知名IT女作家孙卫琴最新作品
网上提供增值服务——孙卫琴
语音版技术讲座。



“开发专家之 Sun ONE”全新提升为“Java 开发专家”系列
——源自精品 成就理想

出版说明

★ 从“开发专家之 Sun ONE”到“Java 开发专家”

“开发专家之 Sun ONE”系列丛书从诞生之日至今，已经四岁了。在这个系列里面，我们一直努力体现着这么一个理念：用一种较为敏锐的视角来跟踪信息技术的发展轨迹，并把可能为广大程序员所希望获得的知识，用图书出版的方式奉献给大家。

在这个系列中，我们陆续出版了约 30 种图书，有《Java 与模式》、《JSP 应用开发详解（第二版）》、《精通 EJB（第三版）》、《Tomcat 与 Java Web 应用开发详解》、《精通 Struts：基于 MVC 的 Java Web 设计与开发》、《JBoss 管理与开发核心技术（第三版）》、《精通 Spring》、《精通 Hibernate：Java 对象持久化技术详解》等一大批读者朋友耳熟能详的作品。很多作品都是在国内没有同类图书的情况下出版的。在这几年的出版工作中，我们时刻感受着市场的风险，也时刻收获着无数读者给我们的认可。

在这个系列中，凝聚了大量资深技术专家的心血。有大家熟知的阎宏、刘晓华、孙卫琴、罗时飞等，还有一些正在不断上升的开发高手。这些非常优秀的国内原创作者们一直都在支持着“开发专家之 Sun ONE”系列的出版工作，在这里，我们要向他们说声：谢谢。

桃李不言，下自成蹊。由于这些年“开发专家之 Sun ONE”在“两个效益”中的杰出表现，电子工业出版社授予这个系列“最佳品牌奖”。

时代不断前进，技术不断变革。为了顺应 Java 领域的技术发展态势，为了赋予这个经典的图书系列更强的生命力，我们将“开发专家之 Sun ONE”升级为“Java 开发专家”。我们将继承原有的出版理念，紧密跟踪技术热点和发展趋势，会聚更多优秀作者，全力奉献更经典的作品。

★ 规划你的 Java 开发之路

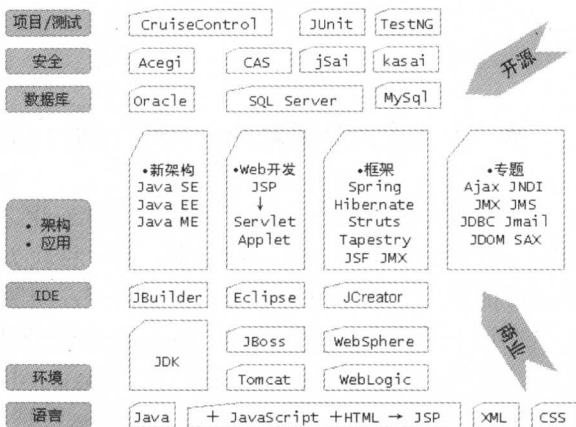
喜马拉雅山脉的最高峰在温室效应中不断地降低，而 Java 世界的颠峰永远都在技术人员的追求中不断升高。每个人都有不同的路，每个人都有不同的行路方式，不过，往往“到了山顶才发现，错误的路和正确的路就差那么几步”！

身处 Java 洪流中的程序员最累（不过大家都说 Java 程序员薪水最高，呵呵），我们简单整理了一下 Java 领域的相关技术、工具、架构，如下图所示。这个框图中的每一个英文单词（或缩写）都可以写成一本书。Java 领域还有一个特点，那就是商业产品和开源产品层出不穷，潮流不断。相比于其他领域，如 .NET，Java 开发更是体现了这句谚语：条条大路通罗马。

罗马只有一个，大路却有多条。看上去，似乎到罗马很容易，反正路多嘛。不过，路多却容易迷失方向。当你在 Java 领域中摸爬滚打几年后，发现自己在无数条道路上走了很久，却不知道罗马何日才能到达，甚至连罗马的方向都不知道，这时你肯定会很失落。

很遗憾，在这个简短的出版说明里面，我们无法告诉你每一条连贯的、不费周折的通往罗马的道路该如何走。或许，通过“Java 开发专家”系列中的某本书，你可以找到属于你的正确道路。在一般情况下，我们不会就某一项很窄的话题来单独写一本书，我们还是希望通过我们的专业知识和智慧，尽力把相关技术整合起来，用较为简明的方式表达出来，最后由你来选择。

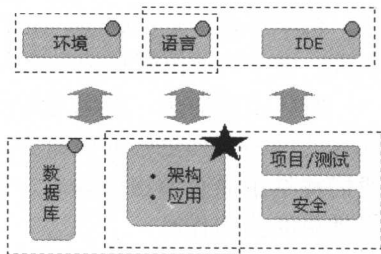
这里有句话与大家共勉：少走弯路，就是捷径！



★ “Java 开发专家”的奉献

犹如在上面那个框图中展现的那样，我们希望在各个层面、各个方向上都能给读者奉献出优秀的图书作品，全面体现技术与应用的结合。从宏观上看，我们会从语言、IDE、环境、数据库、架构与应用、安全、项目与测试等方面进行选择，选出一些读者迫切需要的技术来先行规划。

“Java 开发专家”虽然新禧初绽，但因其源自盛放的“开发专家之 Sun ONE”系列而根基稳健，两个系列会有一段很长的并行时间，我们会用一种优化的方式来保证读者的顺利选择。无论哪一个系列，必定都有大家喜欢的图书。



在技术上，有着持久化的方法，在学习上，也需要有持久化的精神。

从“开发专家之 Sun ONE”到“Java 开发专家”，希望可以带给你持久化的动力。

联系方式

咨询电话：(010) 68134545 88254160

电子邮件：support@fecit.com.cn

服务网址：http://www.fecit.com.cn http://www.fecit.net

通用网址：计算机图书、飞思、飞思教育、飞思科技、FECIT

本书的缘起

全球信息资源整合的时代已经来临, 电子商务 B2B 已经谈了很多年了, 什么是真正的电子商务 B2B, 难道仅仅是一些厂商将自己的产品发布到一个网站上让买家浏览就是电子商务 B2B 了?

真正的电子商务 B2B 至少应该包括物流、资金流和库存的信息流控制, 应该是各种信息服务资源的整合平台。比如说, 当一个买家在某一个电子商务平台决定购买一个产品时, 此电子商务平台将调用买家的资金账户管理服务系统, 冻结相应的资金; 此电子商务平台同时调用卖家的库存管理服务系统, 检查是否有足够的存货; 如果资金和存货都达到要求, 此电子商务平台将调用一家物流配送公司的物流管理服务系统把卖家的产品送到买家。

电子商务平台不需要所有者亲自开发买家的资金账户管理服务系统、卖家的库存管理服务系统, 以及物流配送公司的物流管理服务系统, 只需要将这些管理服务系统以插件的方式加入到这个电子商务整合平台即可。电子商务整合平台可以对相应的服务进行安全控制和服务状态监控等, 要实现这样的整合平台技术, 就需要 SOA 的架构技术。

SOA 是目前最流行的热门话题。SOA (Service Oriented Architecture) 的架构思想, 即以服务为导向的软件开发思想几年前就出现了。SOA 的本质需求是什么? 为什么需要 SOA? 作者去年参与了基于 J2EE 和 SOA 架构的美国 pepboys 汽车修理和加油连锁店 POS 系统项目的设计开发, 感觉大量的软件开发和设计人员所理解的 SOA 就是 Web Service, 也就是 SOAP。如果单纯从提供分布式服务的角度来看, CORBA 和 J2EE 的 EJB 就可以满足服务需求, 为什么一定需要 Web Service 呢?

决定一种新的技术是否有前途, 关键是看它是否能够满足市场的需求。这样就需要开发者首先知道目前最有潜力的市场需求是各种跨平台的软件服务的整合。

要实现跨平台的软件服务的整合, 就必须解放软件服务的传输协议的限制, Web Service 必然成为首选。然而 SOA 不仅仅要提供一种服务, 其本质是要提供一个整合和监控各种松散耦合 (或完全解耦) 服务的整合平台。

目前各个大公司如 IBM 等都在积极推出自己的 SOA 的服务整合平台技术, IBM 的 Websphere 6.0 所推出的 Websphere 6.0 SIBus 服务总线就是一个非常好的 SOA 技术平台。

Java 的出现, 实现了跨操作系统平台的程序开发, 以 Java 为基础的 J2EE 技术已经成为因特网服务技术的主流。接下来, 以 J2EE 为基础的 SOA 架构技术必将很快成为下一个因特网服务技术的主流, 这就是本书“基于 SOA 架构的 Struts、EJB 和 Web Service 的 J2EE 整合应用开发”的缘起和由来。

本书的组织架构和主要内容

第 1 章 SOA 架构概述及准备工作

首先介绍了 SOA 的核心实质,以及基于 SOA 架构整合应用的技术组成。Eclipse 和 RAD 已经成为目前主流的 Java 集成开发工具(IDE)。事实上 RAD 就是 Eclipse 的升级版。Eclipse 的不需安装,直接复制到相应目录即可执行,以及以插件方式就可增加新功能的方式,越来越为广大 Java 开发人员所喜爱。Eclipse+Lomboz 插件一直是开发 J2EE EJB 等的利器。Eclipse 3.0 以上版本+Lomboz 插件来开发 J2EE 的 JSP 或 EJB 一直有各种各样的问题,严重影响了广大的 Java 开发人员的效率。作者解决了这些问题,并将 Eclipse 3.1.2+Lomboz 完全支持 JSP、Session Bean 和 Entity Bean 的可运行版本借本书公布出来,希望能对广大的 Java 开发人员有所帮助。

此外,本章一步一步详细介绍了安装 BEA WebLogic 9.1 和 Eclipse 3.1.2+Lomboz 插件的过程、如何创建 Web Module,以及如何检验安装是否正确。

第 2 章 基于 Eclipse 的 Struts 编程详解

以易懂的实例详细介绍了如何在 Eclipse 3.1.2 上创建基于 Struts 1.28 的 MVC Web 应用,包括创建和配置 Controller、Form Bean、Action Bean、Struts Validator,以及应用 Struts 的 html taglibs、bean taglibs、logic tag libs 创建 JSP。

第 3 章 基于 Eclipse 的 Struts + Session Bean 编程详解

以易懂的实例介绍了用 Eclipse 3.1.2+Lomboz 插件创建 EJB Session Bean,以及创建和实现 Struts 对 EJB Session Bean 的调用。

第 4 章 基于 Eclipse 的 Struts + Session Bean + Entity Bean 编程详解

以易懂的实例介绍了用 Eclipse 3.1.2+Lomboz 插件创建 EJB Entity Bean CMP,创建 CMP 的复合主键类,创建新的 finder 方法,在 WebLogics 9.1 上配置数据源,以及创建和实现 Session Bean 对 Entity Bean CMP 的调用。本章的目的在于介绍创建和实现基于 Eclipse 3.1.2 和 WebLogic 9.1 的从前台到后台的 Struts+Session Bean+CMP 的整合编程。

第 5 章 基于 Eclipse + Axis 的 Web Service、EJB、Struts 的整合编程详解

首先介绍了 Web Service 的出现原因,在于实现业务接口层和业务实现层的分离,以及服务接口和调用的标准化;然后详细介绍了 Web Service 实现分布式服务的基本原理。

接着以具体的实例介绍了基于 Eclipse+Lomboz 来创建 Java Bean 的 Web Service,进一步介绍了基于 Eclipse+Lomboz 来创建无状态会话 Bean 的 Web Service;又介绍了基于 Eclipse 和 Axis 创建 WSDL,以及基于 WSDL 创建客户端的 Java 代码,作者提供了相应的脚本,读者只要进行简单的修改就可直接应用到实际的项目之中。

最后介绍了实现客户端和 Web Service 的集成,以及创建 Axis 的 TCP Monitor 来监控 Web Service 的运行。

第 6 章 基于 BEA AquaLogic 的 SOA 架构的整合应用详解

介绍了基于 BEA AquaLogic Service Bus 构建 SOA 架构的基本原理,结合具体的实例,

详细介绍了基于服务总线创建构架 SOA 的服务集成的整个过程,最后介绍了基于 SOA 架构的整合应用的实现。

第 7 章 基于 Websphere RAD Session Bean + Entity Bean + Web Service 编程详解

目前 IBM 的 Websphere 服务器和 BEA 的 WebLogic 服务器是 J2EE 市场上两大主流企业级的应用服务器。上面章节的 Eclipse+Lomboz 可以完成除 Websphere 之外的所有应用服务器的 J2EE 开发支持。

RAD 是 WSAD 的升级版,也是基于 Eclipse 架构的专门支持 Websphere 服务器的 Java 集成开发环境 (IDE),本章以详细实例介绍了在 RAD 中创建 Session Bean 和 CMP,创建新的 finder 方法,在 RAD 上创建相应的数据源,在 RAD 上将 Session Bean 发布成 Web Service,以及实现整合应用编程。

第 8 章 基于 IBM SIBus 服务总线的 SOA 架构的网上书店系统实战详解

详细介绍了创建基于 Websphere 6.0 SIBus 服务总线的 SOA 架构的企业服务系统。以一个网上书店系统的实例,详细介绍了 SOA 的实施过程。完成了在 Websphere 6.0 上建立 SOA 的网上书店服务体系,然后在 WebLogic 9.1 中应用 Struts 调用 SOA Web Service 客户端,实现基于 SOA 架构的 Struts、EJB 和 Web Service 的 J2EE 跨服务器平台的整合应用开发。

本书的一个最大特点就是对所有的概念和原理的描述(如 Struts、Session Bean、CMP、Web Service、SOA 等)都是从市场需求的角度、开发人员的角度出发,以笔者实际工作的心得来讲述的,从而抓住了这些技术的根本。这些技术本来就是应市场的需求而生的,其目的在于使开发人员更加专注于服务和业务本身。如果看不到这两点,就不可能了解这些技术的本质,所学的东西在实际项目中无法发挥作用。

本书的另一个特点就是每一个章节都给出了完整的工作环境(不仅仅是单纯的源程序),读者只需要在打开 Eclipse 或者 RAD 时将路径指向相应的工作环境即可,不需要安装和编辑,即可运行和调试相应的例子,得到最佳的学习效果及实际项目真正需要掌握的技术经验。

本书的读者对象及如何学习本书

前一段时间作者所在部门要招收一些在校的实习生,作者面试了一些知名院校的名列前茅的计算机专业硕士研究生,深深地感觉到他们所掌握的计算机技术跟目前实际项目所需的技术要求相差太大,如果有一本很好的和实际项目相结合的技术书来指导他们,必然能够帮助他们尽快掌握实际项目中真正所需要的经验,这也是本书诞生的重要原因之一。

要学好一门技术,首先要知道为什么要学这门技术,这门技术为什么出现,这门技术所满足的市场需求是什么。J2EE 主要目的就是要将广大的软件设计和开发人员从软件技术本身解放出来,如从安全控制、程序流程监控等之中解放出来,让开发人员花更多的时间

去直接关注如何实现业务需求本身。

以 EJB 学习为例，作者认为学习人员不要花太多的时间去琢磨如何写 Remote Interface 接口类、Home Interface 接口类和 EJB Bean 的 ejbActivate() 方法等，因为这些都可以通过开发工具 IDE（如 Eclipse+Lomboz）直接生成，实际项目开发中也不可能让开发人员直接编写这些。因而开发人员应该多花时间去学习如何通过 IDE（如 Eclipse+Lomboz）来生成这些 EJB 接口，以及如何更好地调用这些接口。

作者认为所有的软件从业人员大致可以分为三类：软件基础理论和规范制订人员，如在 SUN 公司制订 J2EE 规范的人员；软件工具开发人员，如在 IBM 公司从事 Webshpere 服务器和 RAD 的开发人员；应用软件工具进行业务应用开发的人员。第三类人员占绝大多数，本书的主要读者也是第三类人员。

编 著 者

第 1 章 SOA 架构概述及准备工作	1
1.1 SOA 的核心实质	2
1.1.1 SOA 和 Web Service 的根本区别	2
1.1.2 SOA 架构的要点	3
1.2 基于 SOA 架构整合应用的技术组成	3
1.3 安装和配置 BEA	
WebLogic 9.1	4
1.3.1 安装 BEA WebLogic 9.1	4
1.3.2 配置 BEA WebLogic 9.1	5
1.4 安装 Eclipse 3.1.2 和 Lomboz 3.1.0 插件	9
1.4.1 下载和安装各种工具和插件	11
1.4.2 调出 Lomboz 菜单	12
1.4.3 在 Eclipse 3.1.2 的 Server 插件中加入 WebLogic 9.1	15
1.5 创建 Lomboz J2EE Project 和 Web Module	19
1.5.1 创建 Lomboz J2EE Project	19
1.5.2 创建 Web Module	21
1.5.3 将 Server 加入 Web Module	23
1.6 配置环境变量	23
1.7 改写 webModule 目录下面的 index.jsp	25
1.8 Lomboz J2EE View	26
1.9 运行本章的例子	27
1.9.1 打开本章的例子	27
1.9.2 发布 Web Module 到应用服务器	27
1.9.3 在 Eclipse 3.1.2 上运行 WebLogic 9.1	28
1.9.4 在 WebLogic 9.1 上安装和运行 Web Module	29
1.9.5 在 Eclipse 中运行浏览器 (Browser)	30
1.10 小结	31

第 2 章 基于 Eclipse 的 Struts 编程详解	33
2.1 Struts 的 MVC 架构的基本原理	33
2.2 在 Eclipse 3.1.2 上装载 Struts 库	34
2.2.1 装载运行时的 Struts 库	34
2.2.2 装载编辑时的 Struts 库	35
2.3 ActionServlet 类	37
2.3.1 ActionServlet 的工作原理	37
2.3.2 配置 ActionServlet	39
2.4 Form Bean 类	40
2.4.1 Form Bean 的结构和功能	40
2.4.2 创建静态 Form Bean	41
2.5 Action Bean 类	43
2.5.1 理解 Action Bean 类	43
2.5.2 创建 Action Bean	44
2.6 配置 Struts-config.xml	46
2.6.1 在 Struts-config.xml 定义转向 Dispatch 的路径	46
2.6.2 在 Struts-config.xml 中配置静态 Form Bean	47
2.6.3 在 struts-config.xml 中配置动态 Form Bean	47
2.6.4 在 Struts-config.xml 中配置 Action Bean	48
2.6.5 配置 controller	49
2.6.6 配置资源文件	49
2.7 创建基于页面输入校验的 Struts Validator	49
2.7.1 Struts Validator 的基本原理	50
2.7.2 配置 Struts 的页面校验文件	51
2.7.3 validator-rules.xml	52
2.7.4 Struts 所提供的校验	52
2.7.5 创建 validation.xml	53
2.7.6 在资源文件中配置错误信息	57
2.7.7 在 Struts-config.xml 中定义 validator-rule.xml 和 validation.xml	58

2.7.8	在 JSP 中输出错误信息	58
2.8	Struts Validator 的实例详解	58
2.8.1	required 实例	58
2.8.2	用 msg 自定义新的错误信息的实例	59
2.8.3	minlength 最小长度的校验实例	61
2.8.4	maxlength 最大长度的实例	62
2.8.5	正则表达式 mask 的 phone 实例	63
2.8.6	E-mail 的实例	64
2.8.7	float 的实例	65
2.8.8	floatRange 的实例	66
2.9	Struts 的标签库	67
2.9.1	Struts 的 HTML 标签库	67
2.9.2	Struts 的 Bean 标签库	75
2.9.3	Struts 的 logic 标签库	77
2.10	使用 JSP 开发实例	83
2.10.1	主页 Main.jsp 和链接页 Top.jsp	83
2.10.2	注册页面 Regis.jsp、成功页面 Success.jsp 和用户登录页面 Login.jsp	85
2.10.3	用户查询 UserList.jsp	90
2.11	创建 Struts 的 Model	91
2.11.1	创建 DTO (Data Transfer Object)	92
2.11.2	创建业务逻辑的应用接口	94
2.11.3	创建实现业务逻辑的类	94
2.12	运行本章的例子	96
2.13	小结	97

第 3 章 基于 Eclipse 的 Struts + Session Bean 编程详解

3.1	Stateless Session Bean 实现分布式应用的基本原理	99
3.1.1	用 RMI 协议实现分布式应用的基本原理	100
3.1.2	Stateless Session Bean 实现分布式应用的基本原理	100

3.2	用 Eclipse 和 Lomboz 实现 Session Bean 开发的基本原理和过程	102
3.3	通过 Lomboz 创建 EJB Module	104
3.3.1	创建 EJB Module	104
3.3.2	将 Server 加入 EJB Module ...	105
3.4	在 Eclipse 上创建 EJB Session Bean	106
3.4.1	通过 Lomboz 创建 EJB Session Bean	106
3.4.2	通过 Lomboz 创建 EJB Session Bean 的方法	109
3.4.3	实现 EJB Session Bean 的方法	112
3.5	通过 Lomboz 自动生成 Session Bean 的接口类	113
3.5.1	Lomboz 生成 Session Bean 接口类的基本原理	114
3.5.2	在 Eclipse 上将 EJB 加入到 Module	115
3.5.3	在 Eclipse 上创建 Session Bean 的接口类文件	117
3.5.4	远程接口 (Remote Interface)	118
3.5.5	远程 Home 接口 (Remote Home Interface)	119
3.5.6	本地接口 (Local Interface)	120
3.5.7	本地 Home 接口 (Local Home Interface)	120
3.5.8	EJB Session Bean	121
3.5.9	Session Bean Util 类	122
3.6	实现 Struts 和 EJB Session 的集成	123
3.7	运行本章的例子	126
3.8	小结	127

第 4 章 基于 Eclipse 的 Struts + Session Bean + Entity Bean 编程详解

4.1	CMP 实现数据层操作.....	130	4.7.1	在 Eclipse 上自动生成 EJB CMP 接口文件	152
4.1.1	CMP 实现数据库表的 insert (增加)	130	4.7.2	远程接口 (Remote Interface)	154
4.1.2	CMP 实现数据库表的 select (查询)	131	4.7.3	远程 Home 接口 (Remote Home Interface)	155
4.1.3	CMP 取得数据库表记录中 的字段.....	132	4.7.4	本地接口 (Local Interface)	156
4.1.4	CMP 修改数据库表记录中 的字段.....	132	4.7.5	本地 Home 接口 (Local Home Interface)	157
4.1.5	CMP 实现数据库表的 delete (删除)	133	4.7.6	CMP 实体 Bean.....	158
4.2	用 Eclipse 和 Lomboz 实现 CMP 开发的基本原理和 过程.....	133	4.7.7	Data Access Bean	160
4.3	数据库设计	134	4.7.8	Bean Util (Bean 辅助类)	161
4.3.1	创建 MySQL 数据库表格	135	4.8	创建 Entity Bean CMP 的 复合主键类	163
4.3.2	创建初始数据	136	4.8.1	创建 Entity Bean CMP	163
4.4	在 WebLogic 上创建 Data Source 数据源	136	4.8.2	用 Eclipse Lomboz 加入 一个主键	164
4.4.1	将 JDBC 的 Driver (类) 放入 WebLogic Server 的 Server class path.....	137	4.8.3	定义复合主键类名	165
4.4.2	在 WebLogic 9.1 中创建 数据源.....	137	4.8.4	修改 ejbCreate()方法返回 复合主键	166
4.5	在 Eclipse Lomboz 上创建 EJB CMP	140	4.8.5	Lomboz 自动生成的复合 主键类 Primary Key Class	167
4.5.1	在 Eclipse Lomboz 上创建 EJB CMP	141	4.8.6	与复合主键有关的远程 Home 接口	170
4.5.2	Lomboz 所创建的 Entity Bean CMP.....	144	4.8.7	与复合主键有关的本地 Home 接口	171
4.5.3	实现 ejbCreate()方法	148	4.8.8	在 CMP 实体 Bean 中 加入 finder 的注释行.....	172
4.6	在 Eclipse 上将 EJB Entity Bean 加入 EJB Module.....	149	4.8.9	在远程 Home 接口中生成 相应的 finder 方法	172
4.6.1	用 bean.xml 查看所加入的 EJB Entity Bean	150	4.8.10	在本地 Home 中形成 finder 方法	173
4.6.2	用 xdoclet.xml 查看所加入 的 EJB Entity Bean	151	4.9	配置 CMP 数据源	174
4.7	在 Eclipse 上自动生成 EJB CMP 接口类	152	4.10	实现 Session Bean 和 CMP 的集成.....	176
			4.11	Struts1.2.8 + Session Bean + Entity Bean 的实例.....	181
			4.12	小结	182

第 5 章 基于 Eclipse + Axis 的 Web Service、EJB、Struts 的整合编程详解	183
5.1 Web Service 的基本概念	184
5.1.1 快速理解 WSDL	185
5.1.2 快速理解 SOAP	189
5.1.3 快速理解 UDDI	190
5.2 Web Service 实现分布式服务的基本原理	190
5.2.1 基于 JAX-RPC 实现 Web Service 的过程	191
5.2.2 实例讲解 JAX-RPC 实现远程分布式透明调用的基本原理和机制	191
5.3 基于 Eclipse + Axis 开发 Web Service 的环境配置	195
5.4 基于 Eclipse 和 Axis 创建 Java Bean 的 Web Service	198
5.4.1 创建服务器端的 Java Bean	198
5.4.2 发布 Web Service 到 Web 服务器	198
5.4.3 测试基于 Java Bean 的 Web Service	200
5.5 基于 Eclipse 和 Axis 创建无状态会话 Bean 的 Web Service	201
5.5.1 为 Web Service 修改传输数据类的数据类型	202
5.5.2 创建无状态会话 Bean	204
5.5.3 发布无状态会话 Bean 的 Web Service	205
5.6 基于 Eclipse 和 Axis 创建 WSDL	207
5.6.1 创建 WSDL 的 ANT 脚本	208
5.6.2 简介自动生成的 WSDL 文件	210
5.7 基于 WSDL 创建客户端的 Java 代码	212

5.7.1 WSDL 创建客户端 Java 代码的脚本	212
5.7.2 WSDL 所创建的 Java 客户端代码	214
5.8 Struts 和 Web Service 的集成	218
5.9 创建 Axis 的 TCP Monitor 监控 Web Service 的运行	219
5.9.1 创建 TCP Monitor 的目录	220
5.9.2 运行 Axis TCP Monitor	220
5.9.3 修改 ServiceLocator	221
5.9.4 运行 Web Service 并实现 TCP Monitor 的监控	222
5.9.5 监控 Web Service 的 SOAP 消息	223
5.10 运行本章的例子	225
5.11 小结	225

第 6 章 基于 BEA AquaLogic Service Bus 的 SOA 架构的整合应用详解	227
6.1 基于 BEA AquaLogic Service Bus 构建 SOA 架构的基本原理	227
6.1.1 企业服务总线 (ESB)	227
6.1.2 ESB 的标准	228
6.1.3 基于 BEA AquaLogic Service Bus 构建 SOA 架构的整合应用的基本原理	228
6.2 基于 AquaLogic Service Bus 构建 SOA 架构的服务集成	229
6.2.1 安装 AquaLogic Service Bus	229
6.2.2 实现基于 Service Bus 的 SOA 架构	231
6.2.3 创建 WebLogic Service Bus 项目 Project	232
6.2.4 导入 WSDL 资源文件	234
6.2.5 创建代理服务 (Proxy Service)	235

6.2.6 创建业务服务 (Business Service)	239	7.8.1 创建新的 CMP 的 ejbCreate 方法	269
6.2.7 将代理服务指向业务服务	242	7.8.2 创建 ejbCreate 方法输入 参数的 DTO 类	270
6.3 基于 SOA 架构的整合应用	245	7.8.3 将 ejbCreate 方法提升到 本地 Home 接口	271
6.4 运行本章的例子	246	7.9 创建新的 CMP 的 finder 方法	272
6.5 小结	247	7.10 在 RAD 中创建 Session Bean	276
第 7 章 基于 Websphere RAD 的 Session Bean + Entity Bean + Web Service 编程详解	249	7.11 在 RAD 中实现 Session Bean 和 CMP 的集成	278
7.1 安装 RAD 和 DB2	249	7.11.1 创建 Session Bean 调用 CMP 的引用 (Reference)	278
7.2 数据库设计	250	7.11.2 创建 Session Bean 对 CMP 调用的 lookup 方法	280
7.3 用 RAD 实现 Session Bean、 Entity Bean 和 Web Service 开发的基本原理和过程	252	7.11.3 实现 Session Bean 和 CMP 的集成	281
7.3.1 RAD 的工作环境	252	7.11.4 将 Session Bean 方法提升 到 Remote Interface	286
7.3.2 RAD 开发 EJB 和 Web Service 的基本原理和过程	253	7.12 在 RAD 中加入应用 服务器 Server	287
7.4 在 RAD 上创建 EJB Project	255	7.12.1 将 Enterprise Application Project 加入 Server	289
7.5 在 RAD 中创建 Enterprise Application Project	256	7.12.2 运行服务器 Server 和 打开 Admin Console	290
7.6 在 RAD 中创建 CMP Entity Bean	257	7.13 创建数据源和在 CMP 中 配置数据源	291
7.6.1 在 RAD 中创建 CMP Entity Bean	257	7.13.1 创建安全机制	291
7.6.2 创建 CMP Entity Bean 的 字段	259	7.13.2 创建 JDBC	293
7.6.3 RAD 自动生成的 CMP Entity Bean	260	7.13.3 创建 JDBC 驱动程序 Driver 的路径变量	295
7.6.4 RAD 自动生成的 CMP 主键类	263	7.13.4 创建数据源	296
7.7 在 RAD 中建立 CMP 与数据 库表之间的映射关系	265	7.13.5 测试数据源连接	298
7.7.1 在 RAD 中创建 meet-In-the- Middle 的 Mapping 关系	265	7.13.6 配置 CMP 指向数据源的 JNDI 名	298
7.7.2 在 CMP 和数据库表间建立 映射关系	268	7.14 在 RAD 上创建基于 Session Bean 的 Web Service	299
7.8 创建新的 CMP 的 ejbCreate 方法	269	7.15 RAD 生成的 WSDL	305
		7.16 Struts 和 Web Service 集成 ...	310