

WebSphere software

IBM WebSphere
技术丛书

WebSphere 环境下的 Web服务开发技术

Ueli Wahli等著 瞿裕忠 廖力 彭晓晖等 编译

W 世界图书出版公司

IBM

IBM WebSphere 技术丛书

WebSphere 环境下的 Web 服务开发技术

Wahl 等著

瞿裕忠 廖力 彭晓晖 等编译

世界图书出版公司

北京·广州·上海·西安

本书由 IBM 公司授权世界图书出版公司北京公司独家出版,限于中国大陆地区发行。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。

图书在版编目(CIP)数据

WebSphere 环境下的 Web 服务开发技术/美国 IBM 公司著;瞿裕忠译.

—北京:世界图书出版公司北京公司,2002.9

ISBN 7-5062-4591-4

I.W... II.①美...②瞿... III.计算机网络—程序设计 IV.TP393

中国版本图书馆 CIP 数据核字(2002)第 068780 号

WebSphere 环境下的 Web 服务开发技术(IBM WebSphere 技术丛书)

编 译:瞿裕忠 廖力 彭晓晖 等

责任编辑:高蓉 彭强

装帧设计:张芸

出版:世界图书出版公司北京公司

发行:世界图书出版公司北京公司(北京朝内大街 137 号)

邮编:100010 电话:64077922

销售:各地新华书店和外文书店

排版:北京嘉芳园制作中心

印刷:北京宏大印刷有限公司印刷

开本:185×240 1/16 印张:27.8

字数:636 千

版次:2002 年 10 月第 1 版 2002 年 10 月第 1 次印刷

ISBN:7-5062-4591-4

定价:48.00 元

版权所有 翻版必究

WebSphere 引导电子商务发展

为了推动 WebSphere 技术在国内应用水平的提高, IBM 公司组织在国内长期从事电子商务和 WebSphere 技术研究的专家, 编译完成了这套中文版的 WebSphere 技术丛书。本丛书介绍了整个 WebSphere 产品的结构体系, 讲述了如何利用 WebSphere 技术建立电子商务平台, 开发企业应用, 整合企业资源, 快速完成企业的电子商务建设。

● WebSphere 是 IBM 在 Web Services 策略中的核心平台

WebSphere 是行业第一个产品化的完整的电子商务平台, 用于提供和部署 Web Services 的功能性基础架构, IBM 为 WebSphere 平台定义了一种崭新的金字塔结构模式, 如图:



图： WebSphere 金字塔结构模式

平台基础——为动态电子商务提供可靠的、具有良好可扩充性和高效性能的灵活的基础设施平台。

用户体验——帮助企业建立门户, 管理商业流程, 并且提供无线和语音服务。为用户提供更好、更丰富的用户体验与感觉, 从而增强用户对企业的忠诚度。

业务整合——缩短业务流程的处理周期和处理成本, 增强企业的投资回报。为业务与应用集成领域提供了最丰富、最完全的解决方案。

通过这种金字塔型的结构, WebSphere 能使软件开发人员快速创建、部署和维护各种 Web

Service 应用,并使之与当前的业务流程进行联接,同时整合企业中各种资源,提供在企业内部或不同企业之间动态管理的商务协作功能,而且通过 IBM 丰富的行业经验和管理理念使企业在电子商务投资中获得最大回报。

● WebSphere 在客户、合作伙伴和开发者中获得了强劲的市场动力

- 全球有超过 50,000 家客户使用 WebSphere 作为其 IT 基础架构平台的核心,包括大多数世界领先的商业银行、电信公司、医疗社会保健公司、证券公司和互联网公司。
- WebSphere 目前成为开发者开发电子商务平台的首选。
- 超过 9,000 家的独立软件开发商 (ISVs) 和 600,000 名程序员基于 WebSphere 开发电子商务应用程序,并且每年以 500% 的速度增长。
- 在 40 个国家有超过 130 个 WebSphere 开发者组织。与 WebSphere 策略联盟的商业伙伴每季度以超过 20% 的速度增长。

● WebSphere 的创新速度在业界比任何人都快

- 在 2001 年,IBM 软件部囊括了美国专利和注册商标局的 524 项软件专利,这个数量超过了主要竞争对手累加的总和。
- 在 2001 年,WebSphere 获得 224 项专利。
- 在 2000 年,WebSphere 赢得了超过 30 多个业界奖项。

2002 年 6 月,根据 IDC 的统计,IBM 在 2001 年中国软件市场以 6.8% 的市场份额再度夺冠,WebSphere 更是连续 12 个季度保持了两位数的高速增长,成为业界成长最快的电子商务基础架构平台软件。学习和掌握 WebSphere 开发技术,可以帮助提升程序员的职业潜能,提高技术支持水平,并成为全球公认的高级技术人才。

在此,谨向为本套丛书从事编译工作的瞿裕忠、候迪老师,在出版过程中做出贡献的 IBM 软件部的各位同仁,组织出版工作的中国图书进出口总公司表示衷心的感谢。

期望这些图书的出版能对您的工作、学习、研究有所裨益。

IBM 公司大中华区软件部
AIM/WebSphere 销售总经理

林湧星

序 言

本书探究面向 J2EE 应用和 Web 服务的一种新的应用开发工具 WSAD(WebSphere Studio Application Developer)。本书介绍了 WSAD 的基本工具和小组开发环境,以及 Web 应用(JSP 和 servlet)、XML、数据、EJB 和 Web 服务的开发和部署。WSAD 是 IBM 的面向客户和服务应用的一种新的 Java 开发工具。它提供了一个 Java 集成开发环境,用来快速地开发基于 J2EE 的应用。它很好地集成了 WebSphere 应用服务器(版本 4),并提供一个内嵌的单机服务器用来测试 J2EE 应用。

Web 服务是一种新的 Web 应用。Web 服务是自包含的和自描述的模块化应用,它们可以在 Web 上被发布、定位和调用。Web 服务执行可调用的函数。这些函数可以是简单请求到复杂业务过程的任何功能。一旦 Web 服务被部署和注册,其它应用就可以发现和调用这个服务。Web 服务的基础是简单对象访问协议(SOAP)、Web 服务描述语言(WSDL)、通用描述/发现/集成(UDDI)框架。

本书包含三个部分。第一部分介绍一个在本书中使用的应用示例“汽车配件”;第二部分是关于 WSAD 的 J2EE 开发和部署;第三部分是关于 Web 服务技术以及 Web 服务在 WSAD 中的开发与部署。

本书的写作班子:

由来自世界各地的专家组创作,这些专家在国际技术支持组织(International Technical Support Organization, ITSO)的 San Jose 中心工作。



Ueli Wahli 是 ITSO 加州 San Jose 中心的一位信息技术顾问专家。在 18 年之前加入 ITSO 之前,Ueli 在 IBM 瑞士做技术支持。他从事 IBM 全球培训课程的写作与授课,涉及相关应用开发、对象技术、VisualAge for Java、WebSphere Studio 和 WebSphere Application Server 产品。Ueli 拥有一个来自瑞士联邦技术学院的数学学位。

Mark Tomlinson 是一个覆盖北欧的 IBM 应用与中间件集成技术销售团队的成员,该团队以英格兰的伦敦为基地。于 1995 加入 IBM 之后,他服务于 IBM 保险业务、IBM 全球服务、IBM 全

球网络和现在的 IBM 软件组织。最近 3 年,他的工作涉及为 IBM 的客户介绍并演示 VisualAge 和 WebSphere 等各种相关技术。他拥有英格兰 Warwick 大学的物理学学位。

Olaf Zimmermann 是一位信息技术顾问设计师,在德国海德堡的 IBM 全球服务和电子商务二次集成服务等部门工作。自 1994 年以来,他一直在为 IBM 工作,涉及产品开发、系统集成、技术销售咨询和授课。他的专长领域包括分布计算和中间件技术,还有企业应用集成、XML 和 J2EE。他拥有德国 Braunschweig 技术大学计算机科学学位。

Wouter Deruyck 是 EMEA AIM 合作伙伴技术支持团队的一位顾问,在比利时的 La Hulpe 工作。自 1999 年以来,他一直在为 IBM 工作。他作为 IBM 全球服务部门的一位信息技术专家起步,随后,他在 IBM 软件组织接受了挑战。他在对象技术方面有着 2 年的经验。他拥有 Leuven 大学的计算机科学学位。

Denise Hendriks 是 Perficient 公司的常务董事和 WebSphere 设计师。Denise 有一个物理学学位和计算机科学学位,并取得电子与计算机工程硕士学位。她加入 IBM 有十年之久,是 VisualAge 和 WebSphere 的资深开发者。她最近加入 Perficient 公司,她在那里作为 WebSphere 和 J2EE 的一位技术顾问充分施展着她的才能。

下列人员为本书做出了很大的贡献,特表谢意!

Tom Bellwood: UDDI Development Lead, Emerging Internet Technologies, Austin, TX;

Peter Brittenham: Web 服务 Toolkit Architect, Emerging Technologies, Raleigh, NC;

Barbara McKee: AIM Architecture, Software Group, Austin, TX;

Frank Mueller: IBM Global Services, J2EE Architect, e-business Integration, Heidelberg, Germany;

Dan Gisolfi: Engagement Manager/Architect, jStart Emerging Technologies, Somers, NY;

Deborah Cottingham: International Technical Support Organization, San Jose Center.



译者序

面向服务的体系结构(Service Oriented Architecture,简称 SOA)是软件技术的一个发展方向。目前,SOA 的关键技术包括 Web 服务(Web Service)、Web 服务描述语言(WSDL)、通用描述/发现/集成(UDDI)框架和简单对象访问协议(SOAP)。Web 服务是一种新的 Web 应用,它是自包含的和自描述的模块化应用,可以在 Web 上被发布、定位和调用。一旦 Web 服务被部署和注册,其它应用就可以发现和调用这个服务。Web 服务及相关技术已经得到了工业界的普遍支持。

WSAD(WebSphere Studio Application Developer)是 IBM 的面向 J2EE 和 Web 服务的应用开发工具。本书(原书名:Web Services Wizardry with WebSphere Studio Application Developer)介绍 Web 服务及相关技术、面向服务的体系结构、WSAD 及其对开发 Web 服务的支持,并通过“汽车配件”应用示例展现如何使用 WSAD 开发和部署 J2EE 和 Web 服务。

本书内容实践性强,应用示例的开发循序渐进,分为四个阶段,贯穿全书。有关应用示例的源代码可从网上下载,相关资料的使用以及产品安装与配置可以在附录中找到。本书前 14 章均配有测验题,帮助读者更好地掌握书中的知识,测验题的答案可以在附录中找到。本书适合于作为计算机专业和软件学院相关课程的教材,也可作为软件设计与开发者的参考书。

本书由东南大学 XObjects Group(<http://xobjects.seu.edu.cn>)翻译,参加翻译工作的还有赵军、金志敏、陈壮生、徐涵和张祥。我们衷心期望这本书能够有益于我国在 Web 服务以及面向服务的软件技术方面的发展。在翻译过程中,译者参阅了大量的国内外资料。在此,谨向书中提到的和参考文献列出的作者表示感谢。本书的翻译工作得到了 IBM 中国有限公司的大力支持,在此,我们深表谢意!

限于译者的水平及时间,本书难免有些不足之处,恳请读者指正。

译者

2002 年 7 月于南京

目 录

第一部分 引言	1	2.4.1 视图(View)	13
第1章 应用示例—汽车配件	1	2.4.2 导航器视图(Navigator view)	13
1.1 应用的使用案例	1	2.4.3 编辑器(Editor)	13
1.1.1 本地经销商查询	2	2.4.4 轮廓视图(Outline view)	14
1.1.2 对汽车制造商的系统 进行查询	2	2.4.5 特性视图(Properties view)	14
1.1.3 对其它的配件制造商 进行动态的查询	2	2.4.6 任务视图(Tasks view)	14
1.1.4 经销商横向查询的实现	3	2.4.7 定制透视图	15
1.2 系统设计	3	2.4.8 Web 透视图	15
1.2.1 第一阶段:本地经销商查询	4	2.4.9 Java 透视图	16
1.2.2 第二阶段:对汽车制造商 的系统进行查询	4	2.4.10 J2EE 透视图	17
1.2.3 第三阶段:对其它的制造商 进行动态的查询	5	2.4.11 数据透视图	18
1.2.4 第四阶段:经销商横向查询 的实现	6	2.4.12 XML 透视图	18
1.2.5 系统框图	7	2.4.13 服务器透视图	19
1.2.6 数据模型	7	2.4.14 调试透视图	20
1.3 应用使用到的产品	8	2.4.15 帮助透视图	21
1.4 小结	8	2.5 项目	21
测验	8	2.5.1 Java 项目	22
第二部分 WSAD	10	2.5.2 EAR 项目	25
第2章 WSAD 概述	10	2.5.3 Web 项目	26
2.1 产品	10	2.5.4 EJB 项目	29
2.2 WebSphere Studio 工作台	11	2.5.5 服务器项目	31
2.2.1 Meta Object Framework:MOF	12	2.6 WSAD 内部系统文件	35
2.2.2 Concurrent Version System:CVS	12	2.7 总结	36
2.2.3 ClearCase LT(CCLT)	12	测验	36
2.3 工作台窗口	12	第3章 WSAD 的 Web 应用开发	37
2.4 透视图	12	3.1 “汽车配件”应用示例第一阶段 的方案纲要	37
		3.1.1 类图和顺序图	37
		3.2 开发前的准备工作	38
		3.2.1 创建一个新的工作空间	39
		3.2.2 项目配置	39
		3.2.3 考察自动生成的 Web	

项目文件	40	4.2 WSAD 中支持 XML 的工具	73
3.2.4 创建 ITSOWSAD 样本		4.3 示例的解决方案概要	74
数据库	41	4.3.1 类图	75
3.2.5 为已生成的 servlet 创建一个		4.3.2 顺序图	76
合适的 Java 包	42	4.4 开发的准备	77
3.3 构建应用	42	4.5 由 SQL 产生 XML 文件	
3.3.1 生成基本的 Web 应用程序	42	和 XML schema	79
3.3.2 设计表单并生成选项	47	4.6 浏览和编辑 XML 和 XSD 文件	80
3.3.3 查看已生成的各类文件	49	4.6.1 编辑 XML 文件	80
3.3.4 部署描述文件 web.xml	50	4.6.2 编辑 XSD 文件	81
3.3.5 改变背景	51	4.6.3 验证 XML 文件的有效性	82
3.3.6 增加图片显示的支持	52	4.6.4 编辑 XML schema	
3.3.7 如果使用 4.0.2 之前的版本		和数据文件	82
代码应作必要的修改	55	4.6.5 修改 XML 以匹配新	
3.4 部署和测试 Web 应用	57	的 schema	85
3.4.1 创建一个服务器项目	57	4.6.6 直接编辑 XML	85
3.4.2 创建一个服务器		4.6.7 创建 XML 到 XML 的映射	86
实例和配置	57	4.7 处理 XML	90
3.4.3 将项目与服务器配置关联	58	4.7.1 用样式表将 XML 转换成	
3.4.4 添加 JDBC 数据源	59	另一个 XM	90
3.4.5 执行 Web 应用	61	4.7.2 用样式表将 XML 转换成	
3.5 总结	63	HTML	92
测验	63	4.7.3 解析 XML 文件	93
第 4 章 WSAD 中的 XML 支持	64	4.7.4 由 Java 程序创建	
4.1 XML 基本知识	64	XML 文件	95
4.1.1 背景	64	4.8 展望	96
4.1.2 XML 基本概念	65	测验	96
4.1.3 XML namespace	68	第 5 章 WSAD 的 EJB 开发	97
4.1.4 XML schema	69	5.1 汽车配件应用示例第二阶段	
4.1.5 处理 XML	71	a 部分解决方案概述	97
4.1.6 XML 解析器	72	5.1.1 类图	98
4.1.7 XML 处理器	73	5.1.2 顺序图	100
4.1.8 小结	73	5.2 应用开发的准备工作	100

5.2.1 项目配置	100	6.2 部署到 WebSphere AEs	128
5.2.2 EJB 映射方法回顾	101	6.2.1 导出 EAR 文件	129
5.3 构建实体 EJBs	101	6.2.2 启动 AEs 服务器	129
5.3.1 创建实体 EJBs	102	6.2.3 启动管理控制台	131
5.3.2 查看生成的文件	104	6.2.4 创建 JDBC 驱动器和	
5.3.3 EJB 类的回顾	107	数据源	131
5.3.4 产生的 EJB 元数据	107	6.2.5 安装 Almaden Autos	
5.3.5 生成映射	107	企业应用	133
5.3.6 添加对主键域的		6.2.6 安装 Mighty Motors	
getter 方法	109	企业应用	134
5.3.7 定义 getter 方法为		6.2.7 保存配置	136
只读属性	110	6.2.8 停止 AEs 服务器	137
5.3.8 将 EJBs 绑定到 JDBC		6.2.9 使用 SEAppInstall 工具	
数据源	111	部署企业应用	137
5.3.9 验证项目的有效性	112	6.2.10 部署 EJB 测试客户机	137
5.3.10 生成部署代码	112	6.2.11 验证企业应用	139
5.4 测试实体 EJBs	113	6.2.12 重新生成 plug-in	140
5.4.1 将 EAR 项目与服务器		6.3 在 AEs 中测试部署的应用	141
配置相关联	113	6.3.1 使用内嵌 HTTP	
5.4.2 执行 EJB 客户端测试		服务器测试	142
参考程序	114	6.3.2 使用 IBM HTTP 服务器	
5.5 开发前端会话 EJB	118	(AEs)测试	142
5.5.1 创建会话 EJB	118	6.4 部署到 WebSphere AE	142
5.5.2 完成前端会话 EJB	119	6.4.1 启动管理服务器	142
5.5.3 提升新的方法至		6.4.2 启动管理控制台	142
EJB 远程接口	123	6.4.3 创建 JDBC 驱动器	
5.5.4 定义 EJB 引用	123	和数据源	143
5.5.5 验证项目有效性	125	6.4.4 创建应用服务器	144
5.5.6 生成部署代码	125	6.4.5 安装 Almaden Autos	
5.6 测试会话 EJB	125	企业应用	145
5.7 总结	126	6.4.6 安装 Mighty Motors	
测验	126	企业应用	147
第 6 章 WebSphere 环境下 Web 和 EJB		6.4.7 启动 WebSphere AE	
应用的部署	128	应用服务器	149
6.1 WebSphere AEs 和 AE 的区别	128	6.5 在 WebSphere AE 上测试	
		部署的应用	149

6.5.1 使用内嵌 HTTP 服务器 测试	150	7.5.1 下载并安装 CVS	167
6.5.2 使用 IBM HTTP 服务器 (AE)测试	151	7.5.2 创建新的资源库	167
6.6 使用 WSAD 进行远程单元测试	151	7.5.3 创建 Windows 服务	168
6.7 总结	155	7.5.4 创建新用户	168
测验	155	7.6 模拟小组开发	168
第 7 章 小组开发	156	7.6.1 配置	168
7.1 小组开发概述	156	7.6.2 顺序开发情景	169
7.1.1 可扩展的体系结构	156	7.6.3 在单个流中并行 开发的情景	171
7.1.2 CVS 与 ClearCase 之 间的区别	157	7.6.4 使用多个流分别开发的情景	174
7.2 工作空间	158	7.7 小组开发的其它话题	176
7.2.1 本地历史信息	158	7.7.1 确定需要管理的文件	177
7.2.2 设定多工作空间	158	7.7.2 备份 CVS 资源库	177
7.2.3 在服务器上执行 WSAD 的可执行代码	159	7.7.3 资源库管理	177
7.3 小组开发术语介绍	159	7.7.4 实现安全	177
7.3.1 流	159	7.7.5 构建脚本	178
7.3.2 最有利的小组模型	160	7.7.6 管理类路径	178
7.3.3 分支	160	7.7.7 使用 CVS 宏	179
7.3.4 版本	161	7.7.8 监视文件的调整	179
7.3.5 术语矩阵	161	7.7.9 其它 CVS 命令	179
7.4 小组开发透视图(Team)	161	测验	179
7.4.1 透视图概述	161	第三部分 Web 服务	180
7.4.2 连接 CVS 资源库	162	第 8 章 Web 服务体系结构概述	180
7.4.3 查看 HEAD 流中的项目	163	8.1 技术动机	180
7.4.4 浏览资源的历史信息	163	8.1.1 技术基础	181
7.4.5 比较资源库中一个文件 的两个版本	164	8.2 面向服务的体系结构介绍	182
7.4.6 添加项目至工作空间	165	8.2.1 服务的角色	183
7.4.7 改变项目类型	166	8.2.2 SOA 栈	183
7.4.8 配置项目	167	8.2.3 先看几个示例	185
7.5 安装本地 CVS 资源库	167	8.2.4 SOA 的实现	187
		8.2.5 接下来是什么	188
		8.3 开发 Web 服务	189
		8.3.1 开发步骤	189
		8.3.2 提供者和请求者	190

的开发策略	190	9.1.4 Web 服务 DADX 组 配置向导	228
8.3.3 服务生命周期	191	9.1.5 UDDI 浏览器(入口,出口)	229
8.3.4 接下来是什么	193	9.2 WSTK	229
8.4 SOAP 介绍	193	9.2.1 WSTK, WSAD 和其他 IBM 工具 之间的关系	229
8.4.1 概述	193	9.2.2 WSTK 体系结构	229
8.4.2 SOAP 消息的结构	194	9.2.3 设计时组件	230
8.4.3 URN	195	9.2.4 运行时组件	231
8.4.4 封装	195	9.2.5 其他组件和功能	231
8.4.5 高级概念	197	9.3 WSAD 4.0 版	231
8.4.6 Apache SOAP 实现	199	9.4 DB2	232
8.4.7 SOAP 小结	202	9.5 Versata	232
8.5 WSDL 入门	204	9.5.1 Versata XML 和 Web 服务 工具包	232
8.5.1 概述	204	9.5.2 VLS 中作为 Web 服务 的导出规则	233
8.5.2 WSDL 文档的结构	205	9.5.3 由业务规则调用 Web 服务	233
8.5.3 服务实现文档	208	9.6 总结	234
8.5.4 服务接口文档	209	测验	234
8.5.5 绑定	210		
8.5.6 WSDL 的 API	213	第 10 章 静态 Web 服务	235
8.5.7 WSDL 小结	214	10.1 “汽车配件”应用示例第二阶段 b 部分的解决方案概要	235
8.6 UDDI 概述	215	10.1.1 类和顺序图	236
8.6.1 UDDI 注册系统结构	215	10.2 准备开发	237
8.6.2 标识(identification)和分类 (categorization)	218	10.2.1 创建新项目 ITSOMightyWeb	237
8.6.3 UDDI API 概述	219	10.3 Web 服务类型	237
8.6.4 UDDI 注册系统的结构	220	10.4 创建 Mighty Motors 的 Web 服务 InquireParts	238
8.6.5 已有的注册系统	221	10.5 创建 JavaBean wrapper	238
8.6.6 UDDI 小结	221	10.5.1 为 JavaBean 创建一个包	238
8.7 总结	222		
测验	224		
第 9 章 对 Web 服务的产 品支持	225		
9.1 WSAD	225		
9.1.1 Web 服务向导	226		
9.1.2 Web 服务客户端向导	227		
9.1.3 Web 服务骨架 JavaBean 向导	228		

10.5.2 创建 JavaBean	239	观察消息内容	262
10.5.3 实现 JavaBean: InquireParts	239	10.8 创建 Almaden Autos 的 Web 服务客户端	266
10.5.4 测试 JavaBean	243	10.8.1 拷贝 WSDL 的服务实现	267
10.6 创建 Web 服务	244	10.8.2 创建 Web 服务客户端	267
10.6.1 使用 Web 服务向导	246	10.8.3 检查生成的客户端文件	268
10.6.2 配置 Web 服务特性	246	10.8.4 测试 Web 服务请求者	268
10.6.3 Web 服务的作用域	247	10.8.5 创建客户端应用程序	269
10.6.4 使用安全 SOAP	248	10.8.6 创建 XSL 样式表	269
10.6.5 选择 JavaBean 方法 和编码类型	248	10.8.7 创建新的 servlet	271
10.6.6 XML/Java 之间的映射 ..	249	10.8.8 完成 servlet	271
10.6.7 定义 Java 至 XML 的映射	249	10.8.9 在 Web 应用程序中添加 新的 servlet 的链接	272
10.6.8 代理生成	250	10.8.10 测试 Web 服务客户端 ..	275
10.6.9 定义 XML 至 Java 的映射	251	10.8.11 关于加强样本程序功能 的考虑	275
10.6.10 校验 SOAP 绑定	252	10.9 利用会话 EJB 创建 Web 服务	275
10.6.11 Web 服务测试客户端 ..	252	10.9.1 利用向导创建会话 EJB 的 Web 服务	275
10.6.12 生成样本客户端	252	10.9.2 测试会话 EJB 的 Web 服务	276
10.6.13 发布 Web 服务	253	10.10 总结	277
10.7 检查生成的文件	253	测验	278
10.7.1 使用先前创建的 XML Schema	254	第 11 章 动态 Web 服务	279
10.7.2 Web 服务的 WSDL 文件	254	11.1 “汽车配件”应用示例第三阶段 的解决方案概要	279
10.7.3 修改服务实现	254	11.1.1 类图	281
10.7.4 修改服务接口	255	11.1.2 顺序图	281
10.7.5 生成 SOAP 部署描述文件	256	11.2 准备开发	282
10.7.6 SOAP 路由 servlet	257	11.2.1 安装 Plenty Parts 的 Web 服务	283
10.7.7 检查部署好的 Web 服务	258	11.2.2 静态 Web 服务(重温)	283
10.7.8 Web 服务客户端代理	259		
10.7.9 Web 服务样本客户端	261		
10.7.10 使用 TCP/IP 监视服务器 ..	262		

11.3 使用 WSAD 中的 UDDI 浏览器	284	12.3.1 创建 JavaBean 框架	309
11.3.1 发布 UDDI 业务实体	285	12.3.2 检查生成的文件	311
11.3.2 把服务的 WSDL 文件导出到 UDDI 注册系统	288	12.3.3 实现 JavaBean 框架	312
11.3.3 使用 Web 浏览器访问测试注册系统	290	12.3.4 测试 Web 服务	314
11.3.4 从 UDDI 注册系统中导入服务的 WSDL 文件	291	12.4 创建 Santa Cruz Sports Cars 的 Web 应用	314
11.3.5 撤消和更新 UDDI 注册系统中的条目	293	12.4.1 拷贝 WSDL 文件到 ITSOSantaWeb 项目中	314
11.4 利用 WSDL 文件生成 SOAP 客户端代理	293	12.4.2 启动服务器	315
11.5 使用 UDDI API	294	12.4.3 创建 Web 服务的客户端	315
11.5.1 在 UDDI 注册系统中查找服务提供者	295	12.4.4 测试代理	315
11.6 更新 Almaden Autos 的 Web 应用程序	300	12.4.5 添加 XSL 样式表	315
11.6.1 UDDI 查询 servlet	300	12.4.6 创建调用 Web 服务的 servlet	316
11.6.2 创建 XSL 样式表	301	12.4.7 在 Web 应用程序中添加 servlet 的链接	317
11.6.3 更新原有的 Web 应用程序	302	12.5 测试 Santa Cruz Sports Cars 的 Web 应用	317
11.7 单元测试	303	12.6 总结	318
11.8 在 WebSphere4.0 中部署增强的应用程序	304	12.7 测验	318
11.9 总结	304	第 13 章 WebSphere 环境下 Web 服务的部署	320
11.10 测验	304	13.1 部署前的准备	320
第 12 章 复合 Web 服务	305	13.2 有关 URL 的注意事项	320
12.1 “汽车配件”应用示例第四阶段的解决方案概要	305	13.3 导出企业应用	321
12.1.1 类和交互图	306	13.3.1 导出 EAR 文件	321
12.2 准备开发	307	13.3.2 添加 SOAP 管理客户端	321
12.2.1 创建新的经销商	308	13.4 部署到 WebSphere AEs	323
12.3 创建 Almaden Autos 的 Web 服务	309	13.4.1 卸载以前的企业应用通过命令除去	323
		13.4.2 安装企业应用	324
		13.4.3 为 AEs 定义 8080 端口	324
		13.4.4 重新生成 plug-in	324

并保存配置	325	14.5.3 小结	361
13.4.5 把必需的 JAR 文件添加 到 AEs 类路径	325	14.6 高级 UDDI 话题	361
13.4.6 使用 AEs 测试应用	325	14.6.1 UDDI operator cloud	361
13.5 部署到 WebSphere AE	326	14.6.2 电子市场 UDDI	362
13.5.1 卸载以前的企业应用	326	14.6.3 门户 UDDI	362
13.5.2 安装企业应用	327	14.6.4 合作目录 UDDI	363
13.5.3 定义 AE 的 8080 端口	327	14.6.5 内部企业应用集成 UDDI	364
13.5.4 把必需的 JAR 文件添加到 AE 类路径	328	14.6.6 试验床 UDDI	364
13.5.5 使用 AE 测试应用	328	14.6.7 管理 UDDI 节点间的联系	365
13.6 SOAPEAREnabler 工具	328	14.6.8 IBM WebSphere UDDI Registry	366
13.7 总结	329	14.7 用 Web 服务进行业务 过程管理	366
测验	329	14.7.1 IBM WSFL 提议	366
第 14 章 Web 服务高级话题	330	14.7.2 WSFL 流程模型	367
14.1 高级 SOAP 编程	330	14.7.3 Web 服务和 MQSeries Workflow	370
14.2 编程方式部署	330	14.7.4 Web 服务和 MQSeries Integrator	373
14.3 编码和类型映射选择	331	14.8 总结	375
14.3.1 编码和映射概述	331	测验	375
14.3.2 用 JavaBean 作为参数 实现 Web 服务	333	第 15 章 体系结构和设计事项	376
14.3.3 创建定制映射	337	15.1 体系结构	376
14.3.4 带 XML 元素和其它 参数的消息	342	15.1.1 Web 服务公共的使用 场景是什么?	376
14.3.5 小结	343	15.1.2 Web 服务在体系结构中 的合理定位是什么?	377
14.4 面向消息的通信	344	15.1.3 对操作模型有何影响?	379
14.4.1 准备开发	345	15.1.4 Web 服务如何与相似 技术关联?	379
14.4.2 从 Web 服务客户端 访问消息服务	348	15.1.5 安全	380
14.4.3 测试消息服务	351	15.1.6 管理和服务质量(QoS)	381
14.4.4 小结	352		
14.5 其它 Web 服务支持	353		
14.5.1 从 DADX 文件 创建 Web 服务	353		
14.5.2 从 URL 生成 Web 服务	360		

15.1.7 性能	382	B.1.1 状态转换图	396
15.1.8 互操作性	382	B.1.2 Not existing →	
15.1.9 SOA 当前版本中有 无不足之处?	383	HEAD stream(1)	396
15.1.10 如何开始我的项目? ...	383	B.1.3 Not existing →	
15.2 设计	383	Branch stream(2)	397
15.3 早期的最佳实践	385	B.1.4 Branch stream →	
15.4 故障分析	386	Not existing	397
第 16 章 IBM 的 jStart 计划	388	B.2 项目生命周期	397
16.1 jStart 提供优势	388	B.2.1 状态转换图	397
16.2 关于此项计划	388	B.2.2 Not existing → Unassigned in workspace(1)	398
16.2.1 我们是谁	388	B.2.3 Unassigned in workspace → Assigned in workspace (2)	398
16.2.2 我们如何工作	389	B.2.4 Assigned in workspace → In repository stream(3) ...	398
16.2.3 我们做些什么	389	B.2.5 In repository stream → Assigned in workspace(4)	399
16.3 竞争优势从何而来	389	B.2.6 In repository stream → Repository version(5)	399
16.4 Jstart 的 engagement 模型	389	B.2.7 Repository version → In repository stream(6) ...	399
16.5 联系我们	390	B.2.8 Repository version → Assigned in workspace(7)	399
附录 A 产品安装与配置	391	B.2.9 Unassigned in workspace → In repository stream(8) ...	400
A.1 DB2 v7.2	391	B.2.10 Unassigned in workspace → Not existing(9)	400
A.1.1 创建数据库示例	391	B.2.11 Assigned in workspace → Not existing(10)	400
A.1.2 JDBC 版本 2	391	B.2.12 Repository version → Not existing(11)	400
A.2 WAS 高级版单机 服务器版(AEs)	392	B.3 文件生命周期	400
A.3 WAS 高级版(AE)	392	B.3.1 状态转换图	400
A.4 WSAD	392		
A.4.1 IBM 代理控制器	393		
A.5 IBM WebSphere UDDI 注册系统	393		
A.6 定义并装载 ITSOWSAD 数据库	393		
A.6.1 示例数据	393		
附录 B WSAD 中资源的状态转换	396		
B.1 流生命周期	396		