

国外信息科学经典教材

SAMS

用Java™构建Web服务

Building Web Services with Java™

— Making Sense of XML, SOAP, WSDL, and UDDI

(美)Steve Graham 等著

刘晓晖 麦中凡
吕庆中 薛学勤 译



机械工业出版社
CHINA MACHINE PRESS



国外信息科学经典教材

用 Java™ 构建 Web 服务

(美) Steve Graham 等著

刘晓晖 麦中凡 吕庆中 薛学勤 译

麦中凡 校

机械工业出版社

分布式计算的下一步发展是 Web 服务。按照开放式工业标准, Web 服务能使我们的软件以松耦合、简单和平台无关的方式与客户、业务伙伴的软件集成。本书介绍如何用 XML、SOAP、WSDL 和 UDDI 表达 Web 服务的各种概念,并详述了与 Web 服务有关的新标准,简单对象访问协议 (SOAP)、Web 服务描述语言 (WSDL)、通用描述、发现与集成 (UDDI),以及如何将 Web 服务运用于我们的业务中。本书适合有志参与 Web 服务制作的软件工作者,以及希望了解分布式计算下一步发展的软件专业的师生。

Simplified Chinese edition copyright © 2002 by Pearson Education North Asia Limited and China Machine Press.

Original English language title: Building Web Services with Java: Making Sense of XML, SOAP, WSDL and UDDI

ISBN 0-672-32181-5

Copyright © 1999 by Sams Publishing.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Sams Publishing.

本书中文简体字版由美国 Pearson Education (培生教育出版集团) 授权机械工业出版社在中国大陆境内独家出版发行, 未经出版者许可, 不得以任何方式抄袭、复制或节录本书中的任何部分。

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

北京市版权局著作权合同登记号: 图字: 01-2002-2697

图书在版编目 (CIP) 数据

用 Java™构建 Web 服务 / (美) 格雷厄姆 (Graham, S.) 等著; 刘晓晖等译. —北京: 机械工业出版社, 2003.1
(国外信息科学经典教材)

ISBN 7-111-11410-8

I. 用... II. ①格... ②刘... III. Java 语言—程序设计
IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 102779 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 车 忱 责任印制: 付方敏

三河市宏达印刷有限公司印刷·新华书店北京发行所发行

2003 年 1 月第 1 版·第 1 次印刷

787mm × 1092mm 1/16·27.25 印张·672 千字

0 001—5000 册

定价: 52.00 元

凡购本图书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话: (010) 68993821、68326677 - 2527

封面无防伪标均为盗版

出版说明

在人类迈入信息时代的今天,信息技术的应用无处不在,它也代表着一种潮流,我国对信息技术的重视和鼓励也达到了空前的程度。信息技术的发展速度很快,可谓日新月异,尤其在一些发达的国家更是如此。我国在信息技术领域起步较晚,但速度惊人,这正是改革开放的具体体现。在我国大力发展信息产业的今天,为了能融入国际的潮流和掌握最新的技术,从国外引进先进的知识和技术就显得格外重要。正是基于这样的出发点,我们决定引进一系列国外信息技术领域有代表性的优秀教材,将它们呈现给我们的学子、教师和 IT 业的有志之士,藉此为我国的信息产业贡献一份微薄之力。

随着我国加入 WTO,国际间的竞争将越来越激烈,而国际间的竞争实际上也就是人才的竞争、教育的竞争。为了加快培养具有国际竞争力的高水平技术人才,加快我国教育改革的步伐,使我国的高等教育尽快与国际接轨,这就需要引进先进的教学思想和教学方法,而引进国外的优秀教材无疑是一种很好的途径。同时,引进国外的优秀教材也有利于提高我国自编教材的水平,让我们的教育工作者从中得到启发。

我们在这套丛书的选题上,遵循“新、优、特”的原则,做到知识新、质量优和内容有特点。这套丛书涵盖了计算机、通信、电子技术等各个领域,每一本书都是精心挑选,在某个领域或学科内具有很强的代表性和很高的价值,很多在国外也被作为大学的教科书,由国际知名的出版公司出版。在引进过程中,我们邀请有关专家对书稿的整体水平进行了评定;在翻译过程中,我们聘请国内相关领域内的有很高学术水平的专家和学者担任译者,以保证书籍的水平和质量,做到对读者负责,为读者着想。

相信这套丛书的出版对正在苦读和即将面临挑战的学子们会有很大的帮助和提高,也能让我们的教学工作者从中得到启发,同时对从事 IT 行业的工程技术和研究人员而言也是很难得的工具书。

机械工业出版社

译者序

当前分布式网络计算已逐渐成为网络应用的主流。Web 服务是集成网上资源实现业务应用的理想手段。Web 服务就是按标准的 XML 描述的应用服务,其中嵌入 Java 的 API 即为 Java 的 Web 服务。这对于谙熟网络技术和 Java 语言的高手并不是新概念,如同有了汇编语言即可写应用程序一样。然而,对于网络计算新手,Web 服务确是一项内容丰富的新技术。

近来业界发展了一系列 Web 服务的标准:简单对象访问协议(SOAP),Web 服务描述语言(WSDL),通用描述、发现与集成(UDDI)首先支持在面向服务的体系结构模型上构建 Web 服务。这无疑为有志进入 Web 服务的软件人员提供了福音。这些标准协议如同程序设计语言中提供了高层概念,如变量、数组、控制三结构、过程调用等。面向服务的体系结构模型如同编程模型。不仅使我们构建的 Web 服务更加规范,有章可循,更重要的是这些标准协议、模型反映了当前软件新技术:基于体系结构的构件软件开发。

《用 Java 构建 Web 服务》一书系统地介绍了这些标准协议。特别是任何通过 Web 服务将本组织地和业务伙伴的各种应用资源集成起来,以解决实际问题。本书概念清晰,面向实用,深入浅出。

本书的作者共有 7 位,都是 Web 服务方面的资深专家。有 W3C XML 协议组成员,有 UDDI 规范起草人,有大公司首席软件建筑师。为了快速即时传播这些技术,他们共同写了这本书。

译者于今年 4 月在机械工业出版社推荐下阅读了本书,获益匪浅。本书使我们在追踪 Web 技术中,思路清晰,可非常具体地用于项目开发。

我们邀请侨居新加坡的资深 IT 工程师刘晓晖先生主译。他也正想在转向新工作之前系统地补充一些新知识。吕庆中博士(第 3 章)薛学勤教授(第 1、2 章)参加了部分翻译工作。最后由麦中凡教授对全书作了审校,并补译了部分章节。由于争取即时性以及我们的水平所限,译文中不统一、不确切之处在所难免。欢迎读者批评指正。

在成稿过程中,胡斌、程勇、李烨都为打印、校对、插图做了大量工作,特别是对于部分重要程序进行了上机验证。

我们的通信地址:

mids@buaa.edu.cn

xiaohui888@sohu.com

译者

2002.10

引 言

欢迎来到 Web 服务世界！Web 服务是一组迅速发展的标准和实现技术，在应用集成和分布式计算领域有着广阔的应用前景。

在开始之前，我们需要先对本书的目的和结构作一些说明。

1. 本书的目的

本书的主要目的是要让读者熟悉 Web 服务的概念以及如何把 Web 服务作为商务处理的一部分。

我们将介绍 Web 服务的概念，并提出一个框架，告诉你应怎样看待各种与 Web 服务相关的新兴标准，如简单对象访问协议(SOAP)，Web 服务描述语言(WSDL)和通用描述发现与集成(UDDI)等。

我们将从商务和技术角度来确定 Web 服务的地位，解释并演示如何使用 Web 服务来解决各种商务问题，尤其是与应用集成相关的问题。

本书的另一个目的是要帮助开发者理解使用本书所介绍的技术构建 Web 服务时的有关问题和细节。例如，当你在规划 Web 服务战略时，需要些什么？在开发 Web 服务时，应注意哪些事项？我们提供了许多例子和可运行的代码来示范这些方案。我们也用可运行的例子详细地复审了 Apache Axis 这一 Web 服务基础结构(infrastructure)。我们也讨论了其他的工具和 Web 服务基础结构，但没有像 Apache Axis 那样详细。

2. 假设的背景

本书面向的是具有一定构建 Web 服务和分布式计算系统经验的计算技术专家。我们并不要求你精于分布式对象技术，但我们假定你对基于 Web 的体系结构和技术(如 HTTP、HTML)有所了解。如果你没有使用这些技术的经验，那么，对有些材料可能会有些发懵——尤其是某些代码例子——但是，你仍然能从本书中学到许多东西。

我们假定你熟悉 Java，尤其是 Java 服务器页面(JSP)和 Java servlet 技术。我们也将扼要讨论 EJB 和 Web 服务之间的关系，所以熟悉一些 EJB 也是有益的。若读者还想多了解一些这些技术，可参考市场上的许许多多关于 Java 编程，JSP，servlets 和 EJB 的优秀书籍。

你将会发现可扩展的置标语言(XML)是处在 Web 服务相关技术的核心位置。虽然我们只用了一章的篇幅来解说构建 Web 服务所需的 XML 核心技术，但是，你对 XML 了解得越多，在构建 Web 服务时将越成功。

3. 宗旨

组织一本关于 Web 服务的书是困难的。许多概念和标准相互依赖，很难孤立地阐明单个的论题，因为正是这些概念和标准的组合才使得 Web 服务对于分布式计算领域是重要的。本书的宗旨可以归结为 4 点：实用性、逐步揭示、演示实例和面向服务(指体系结构框架)。

(1) 实用性

本书力图让读者尽快接触编程示例和运行代码。尤其是,我们的重点是利用 Apache Axis Web 服务基础结构,构建和消费基于 SOAP 的 Web 服务。这是一个以 Java 为中心的构建 Web 服务的方法。尽管我们强调 Web 服务从根本上对编程语言是中性的,归根结底,任何一个给定的 Web 服务总是用某种编程语言技术来实现的,本书选择了 Java。在与其他编程语言编写的 Web 服务交互操作时,我们将指出可能出现的问题之处。虽然我们在第 8 章“可交互操作性、工具及中间件产品”给出了一些基本的 .NET 和其他环境的例子,但详细介绍其他的 Web 服务实现方法,如 Microsoft .NET,不属本书讨论范围。

(2) 逐步揭示

概述了 Web 服务之后,我们从 XML 基础开始,然后以商务计算问题为契机,逐层添加新概念。这些层次产生了一系列的 Web 服务技术“栈”。对于 Web 服务领域中的每一种技术和标准,我们的重点是帮助读者理解这些技术能够解决什么问题,而非详尽地描述技术本身。

(3) 演示实例

对于建立 Web 服务概念的每个技术和标准,我们都将结合可运行的例题(本引言的后文将介绍该例题)进行说明。利用可运行的例题可加深对书中概念解释的理解,并且,随着我们的举例,在解释中逐层增加 Web 服务技术,这也体现了本书逐步揭示的方法。这种方法有助于读者理解 Web 服务开发及部署的各种最佳实践方法。读者可从 www.sampublishing.com 中下载这些可运行例题的源代码。当你打开该网页后,在搜索框中键入本书的 ISBN 号(0672321815),即可获得有关本书和源代码的链接信息。

(4) 面向服务的体系结构

本书是在面向服务的体系结构(SOA,第 1 章“Web 服务概述”将作介绍)中讨论这些例子和 Web 服务概念。我们利用 SOA 框架有助于把 Web 服务的各种概念放置到更大的概念中。

4. 本书的组成概述

第 1 章以解释什么是 Web 服务技术开始本书。我们描述了什么是 Web 服务,有哪些标准和技术与 Web 服务相关联,利用 Web 服务能解决什么问题。本章也将介绍 SOA 概念框架,并解释各种不同的 Web 服务标准,如 SOAP、WSDL 和 UDDI 是如何组合在一起的。本章将为你阅读本书后文打下坚实的概念基础。

在我们进入 Web 服务的核心标准之前,第 2 章“XML 初阶”先对 XML 作了简短解释。因为 XML 处在所有 Web 服务标准和技术的中心位置,清晰地理解它是十分重要的。XML 是个大题目,我们只重点考察读者必须了解的部分,以便读者理解本书后文中的 Web 服务论题。

复习了 XML 后,第 3 章“简单对象访问协议(SOAP)”深入到“Web 服务调用”这一核心问题。我们将回顾在分布式计算环境中 XML 传递消息的论题,并将重点放在 SOAP 消息封装的标准上。SOAP 是 Web 服务环境中服务请求者和提供者之间通信的核心基础。

第 4 章“创建 Web 服务”在特定的 SOAP 基础结构:Apache Axis 项目的上下文中细化对 SOAP 的理解。第 4 章深入到 Axis 如何工作的细节,说明了如何使用 Axis 部署 Web 服务以及如何编写应用程序消费 Web 服务。

至此,你将对 SOAP 有了充分的理解,并至少掌握 SOAP 的一个实际使用方法:通过 Axis。但是,单有 SOAP 只能够实现非常简单的 Web 服务。第 5 章“SOAP 用于电子商务”通过解释

如何将完整的商务计算问题构建为 Web 服务,把第 3、4 章中所引入的概念作了补充细化。第 5 章讨论了 Web 服务如何处理有关分布式计算的各种问题,包括:安全、性能、服务质量、可靠性问题等。

第 6 章“描述 Web 服务”,介绍服务描述的重要概念。服务描述是 Web 服务作为应用集成技术的关键所在,也使得 Web 技术能够用于构建松散耦合系统。第 6 章讨论了 Web 服务如何利用服务描述来处理通信问题,以及服务请求者需要知道 Web 服务的哪些细节,才能正确地理解如何以及为什么要调用 Web 服务。

紧接着,你需要理解服务请求者如何获得服务描述。第 7 章“发现 Web 服务”,补充了第 6 章所遗留的问题,讨论了发现 Web 服务的各种技术。这一章讨论了有关查找的标准以及公司如何查找潜在商务伙伴提供了哪些 Web 服务。

第 8 章“可交互操作性,工具及中间件产品”通过考察各种其他的 Web 服务基础结构和工具环境,以补足你对 Web 服务领域中最佳实践的理解。

本书以展望将来的第 9 章“未来概念”作为结束,介绍了 Web 服务技术的一些未来应用,以解决分布式计算的其他问题。

注:

本书介绍了许多你可能不熟悉的术语。在本书末,我们作了一个术语汇编,作为本书所用术语的参考导引。我们用符号  来标记书末汇编的术语在书中的第一次出现。

在开始之前,我们引入一个虚构的公司:SkatesTown(滑板城)。SkatesTown 的例子将贯穿全书,我们将跟随 SkatesTown,学习如何开发 Web 服务以改善其商务。

5. SkatesTown 介绍

SkatesTown 是纽约的一家正在成长的小公司,它由三位爱好汽车和滑板,并具有机械专长的朋友们创立。他们开始是在 Dean Carroll 的汽车房设计并销售定制的预制滑板,不久他们的工作质量就传开了。他们革新了一些制造技术,几个月内,他们的定单就堆积了起来。现在,SkatesTown 在布鲁克林有了一间小的制作作坊,而公司已在全城的商店里销售滑板、服装和器械。Dean, Frank Stemkowski 和 Chad Washington 对他们业务的拓展感到无比欣慰。

三个人中,Chad 是一位实干的机械高手,他负责完成了许多大胆的结构和设计选型,使得 SkatesTown 达到今天这样的业绩。他是这个小组的领头人和公司的主席。Frank 从孩提时代就是一个合群的会说话的人,现在负责市场和营销。Dean 多年来都在密切关注计算机技术的革新,他是公司的技术主管。

几年前,Dean 就意识到网络技术正在发展壮大,他希望了解 SkatesTown 能否赶上这一技术浪潮,利用分布式计算来推动其商务。这的确是一个明智之举。

Dean 设立了一个网站,这样,SkatesTown 的客户就能及时获得产品信息,而 SkatesTown 也无须安排众多的职员来回答电话和询问。他也建立了在线订单处理系统,以加快与使用网络的客户之间的商务处理流程。近几个月来,越来越多营销 SkatesTown 产品的商店都在使用这一系统,并取得了显著效益。

6. 我们的故事由此开始

目前,Dean 对 SkatesTown 电子商务系统的运作相当满意。但是还有一些问题,Dean 相

信事情能做得更好。他意识到随着业务的扩大,与订单收集和库存补充相关的手工作业将会限制公司的成功。他始终是一个有前瞻性的人,Dean 听到了一些关于 Web 服务的传闻,于是他想知道得更多。在朋友的敦促下,他与银弹咨询(SBC)公司的承包人 Al Rosen 进行了接触。银弹公司专门从事 Web 服务的解决方案。经过与 Al 的几次会晤后,Dean 信服了,于是他雇佣 SBC 来评价 SkatesTown 的系统,并协助公司成长为以 Web 服务开拓业务的企业。

随着我们往下阅读本书的其余部分,我们将关注 SkatesTown 如何采用一些技术,如 XML,和后来的 SOAP、WSDL 和 UDDI 技术以提高效率、生产率,并与它的客户和商务伙伴建立新的、有价值的联系。

目 录

出版说明

译者序

引言

第 1 章 Web 服务概述	1
1.1 什么是 Web 服务	1
1.1.1 业务视角	3
1.1.2 技术视角	3
1.2 Web 服务的机遇	3
1.2.1 企业应用集成	4
1.2.2 B2B	4
1.3 电子商务处理的趋势	5
1.4 为什么需要 Web 服务技术	7
1.4.1 问题的范围	8
1.4.2 核心技术	8
1.4.3 业界动态	9
1.5 面向服务的体系结构	11
1.5.1 Web 服务可交互操作栈	13
1.5.2 描述栈	14
1.5.3 发现栈	16
1.5.4 结合各交互操作栈	17
1.6 小结	17
第 2 章 XML 初阶	19
2.1 XML 的由来	20
2.2 以文档为中心与以数据为中心的 XML	21
2.2.1 以文档为中心的 XML	21
2.2.2 以数据为中心的 XML	22
2.2.3 文档的生存期	23
2.3 XML 实例	23
2.3.1 文档的序言	23
2.3.2 元素	25
2.3.3 属性	27
2.3.4 字符数据	29
2.3.5 一个更简单的购买订单	31
2.4 XML 名字空间	31

2.4.1 名字空间机制	32
2.4.2 名字空间语法	33
2.4.3 带名字空间前缀的属性	35
2.5 文档类型定义	36
2.5.1 良定义性和有效性	37
2.5.2 文档结构	37
2.5.3 DTD 足够了吗	38
2.6 XML 模式	39
2.6.1 XML 模式的基础	39
2.6.2 模式与文档结合	40
2.6.3 简单类型	41
2.6.4 复杂类型	44
2.6.5 购买订单模式	46
2.6.6 基本模式的可重用性	48
2.6.7 高级模式的可重用性	53
2.6.8 还有更多	60
2.7 处理 XML	60
2.7.1 基本操作	60
2.7.2 面向数据的 XML 处理	62
2.7.3 基于 SAX 的 checkInvoice(发票检查)	64
2.7.4 基于 DOM 的 checkInvoice(发票检查)	69
2.7.5 测试代码	74
2.8 小结	76
2.9 资源 (Resources)	77
第 3 章 简单对象访问协议 (SOAP)	79
3.1 XML 协议的演进	80
3.2 简单对象访问协议 (SOAP)	82
3.2.1 SOAP 的市场	82
3.2.2 SOAP 应做些什么	83
3.2.3 到底什么是 SOAP	83
3.3 SkatesTown 公司的电子商务系统	84
3.4 库存检查 Web 服务	87
3.4.1 选择 Web 服务引擎	87
3.4.2 服务提供者视图	88
3.4.3 服务请求视图	89
3.4.4 测试服务	90
3.4.5 传输线上的 SOAP	91
3.5 SOAP 封套框架	94
3.5.1 SOAP 封套	94

3.5.2	SOAP 版本	94
3.5.3	SOAP 消息头	95
3.5.4	SOAP 消息体	96
3.6	利用 SOAP 的可扩展性	96
3.6.1	服务请求者视图	97
3.6.2	服务提供者视图	99
3.6.3	测试 Web 服务	101
3.6.4	传输线上的 SOAP 消息	102
3.7	SOAP 中间节点	102
3.7.1	需要中间节点的理由	103
3.7.2	SOAP 的中间节点	104
3.7.3	综合示例	104
3.8	SOAP 中的错误处理	107
3.9	SOAP 的数据编码	108
3.9.1	指定不同的编码算法	108
3.9.2	SOAP 数据编码规则	109
3.9.3	选择数据编码方式	114
3.10	用 Web 服务构建分布式系统	118
3.10.1	消息传递	119
3.10.2	消息传递与 RPC	121
3.10.3	基于 SOAP 的 RPC	123
3.11	处理订单提交的 WEB 服务	125
3.11.1	购货订单和发票模式	125
3.11.2	服务请求者视图	129
3.11.3	服务提供者视图	131
3.11.4	测试 Web 服务	132
3.11.5	传输线上的 SOAP 消息	132
3.12	SOAP 协议束定	133
3.12.1	总体考虑	134
3.12.2	HTTP/S	135
3.12.3	带附件的 SOAP 消息	136
3.12.4	基于 SMTP 的 SOAP	137
3.12.5	其他协议	137
3.13	总结	137
3.14	前面的路	138
3.15	资源	139
第 4 章	创建 Web 服务	140
4.1	为什么要用 Axis 及什么是 Axis	141
4.2	Axis 体系结构	141

4.2.1 Axis 部件	142
4.2.2 定位服务链	148
4.2.3 XML 语法分析	149
4.3 安装 Axis	150
4.4 设置 Axis	150
4.5 安全性	155
4.6 简单 Web 服务	155
4.7 客户端编程	156
4.8 高级 Web 服务部署	158
4.9 以文档为中心的服务	159
4.10 数据编码/解码	162
4.11 创建处理器	163
4.12 专用支点处理器(供应者)	164
4.13 错误	165
4.14 消息模式	166
4.15 创建和部署中间节点	166
4.16 SOAP V1.2	167
4.17 监控	167
4.18 小结	168
第 5 章 SOAP 用于电子商务	169
5.1 Web 服务的安全性	169
5.1.1 示例场景	171
5.1.2 SSL 及 HTTP 基本身份验证	171
5.1.3 数字签名	181
5.1.4 XML 加密	185
5.1.5 公证服务	189
5.1.6 授权	190
5.1.7 安全声明	193
5.1.8 公开密钥基础结构与密钥管理	194
5.1.9 如何实现安全性	198
5.2 企业应用集成	199
5.2.1 基于 J2EE 的 SOAP 服务器	199
5.2.2 事务处理	201
5.2.3 ACID 与两阶段执行	206
5.2.4 可靠消息传递	209
5.2.5 J2EE 的安全模型	216
5.3 服务质量	219
5.3.1 企业 SOAP 服务器	219
5.3.2 高可用性	219

5.3.3 系统管理	221
5.3.4 企业安全性	222
5.4 小结	222
5.5 资源	223
第6章 描述 Web 服务	226
6.1 为什么需要服务描述	226
6.2 服务描述的任务	227
6.3 良定义的服务	227
6.3.1 功能描述	228
6.3.2 非功能描述	228
6.3.3 聚合/协调描述	229
6.3.4 服务描述栈小结	229
6.4 IDL 的历史	230
6.5 Web 服务定义语言(WSDL)	233
6.5.1 WSDL 信息模型	233
6.5.2 WSDL 语言的元素	235
6.5.3 PortType	242
6.5.4 Operation	242
6.5.5 Message	245
6.5.6 Binding	248
6.5.7 Port	255
6.5.8 Service	255
6.5.9 Definitions	256
6.5.10 Documentation	256
6.5.11 import 元素的常规使用	256
6.5.12 WSDL 扩展机制	259
6.6 WSDL 与 Java	262
6.6.1 由 WSDL 导出代码	262
6.6.2 由代码导出 WSDL	280
6.7 未来的服务描述	281
6.7.1 Web 服务端点语言(WSEL)	281
6.7.2 Web 服务流语言(WSFL)	282
6.8 小结	283
第7章 发现 Web 服务	284
7.1 服务发现的任务	284
7.2 注册处的任务	284
7.2.1 设计阶段与运行时的服务发现	285
7.2.2 服务发现的多种机制	285
7.2.3 示例场景修改	287

7.3	UDDI	288
7.3.1	UDDI 使用模型	289
7.3.2	UDDI 的 tModel 概念	295
7.3.3	在 UDDI 注册处发布商务信息	307
7.3.4	在 UDDI 注册处发布服务信息	313
7.3.5	在 UDDI 注册处查找信息	323
7.3.6	在 UDDI 注册处查找商务及服务细节	330
7.3.7	UDDI 1.0 版小结	331
7.4	私营 UDDI 注册处	331
7.4.1	为什么公司要设立私营的 UDDI 注册处?	331
7.4.2	私营 UDDI 的 5 种类型	333
7.5	UDDI 规范 2.0 版介绍	337
7.5.1	UDDI2.0 版的修改概述	337
7.5.2	第三方分类法	337
7.5.3	建立 businessEntity 项之间的关系模型	339
7.5.4	查询 API 的修改	341
7.5.5	发布 API 的修改	348
7.5.6	杂项修改	349
7.6	结合 WSDL 与 UDDI	351
7.6.1	储存基于 WSDL 的 businessService	352
7.6.2	复杂的 WSDL 及相应的 UDDI 注册项	354
7.6.3	完整示例:基于 WSDL 的 UDDI 与动态查找	359
7.7	小结	368
第 8 章	可交互操作性、工具及中间件产品	369
8.1	可交互操作性	369
8.1.1	SOAP 构建者(Soapbuilders)社团	370
8.1.2	可交互操作性实验室(The Interoperability Lab)	370
8.1.3	W3C:标准化的 SOAP	371
8.2	Web 服务的广阔前景	372
8.2.1	SOAP 系统的构建者	372
8.2.2	其他语言及环境	373
8.2.3	SOAP::Lite-使用 Perl 的 Web 服务	374
8.2.4	.NET Web 服务入门	375
8.2.5	GLUE:构建 Java Web 服务的另一工具	381
8.3	小结	383
8.4	资源	384
第 9 章	未来概念	385
9.1	计算技术作为工具	385
9.2	本体及语义 Web	388

9.2.1 资源描述框架	389
9.2.2 本体	390
9.2.3 RDF 与 Web 服务的关系	390
9.3 软件代理	391
9.4 对等网络计算技术	392
9.5 网格计算技术	393
9.6 嵌入式 Web 服务	394
9.7 小结	395
9.8 资源	395
术语表	397

第 1 章 Web 服务概述

本章提要

- 什么是 Web 服务
- Web 服务的机遇
- 电子商务处理的趋势
- 为什么需要 Web 服务技术
- 面向服务的体系结构
- Web 服务可交互操作栈

本章将介绍基本术语和一组概念,这些概念构成本书其余部分的论域环境。我们将定义所谓的 Web 服务是什么意思,并且描述 Web 服务将起重要作用的各种场合。我们将描绘一个简单的框架,它称为:面向服务的体系结构(service-oriented architecture),它有助于构造 Web 服务技术的应用。我们还将以三种“可交互操作”栈的形式,提供一个框架,以明确各种 Web 服务技术,如:简单对象访问协议(SOAP), Web 服务描述语言(WSDL)和通用描述发现和集成(UDDI)等的地位和相互关系。

本书的其余部分是对这里所提及的基本概念的深化。

1.1 什么是 Web 服务

这是一本关于构建 Web 服务的书。如果不首先明确 Web 服务是什么,就无法描述如何构建 Web 服务。

当前,术语“Web 服务”得到了大量的关注和契机。许多软件商(大的或小的)声称已启动和采用(参见“Web 服务市场动态”)了 Web 服务。许多机构都参与 Web 服务标准的细化工作。不过,要达成对这一术语的共同理解似乎是一个缓慢的过程,还不存在单一的、广泛采用的关于“Web 服务”的定义。这种情况使人回想起面向对象程序设计的早期情况:直到继承、封装和多态性被正式定义后,面向对象程序设计才被接受为开发方法学中的主流。

几个主要 Web 服务基础结构的提供者发布了他们的 Web 服务定义:

IBM 在 <http://www4.ibm.com/software/solutions/webservices/pdf/WSCA.pdf> 中提出了如下定义:

“Web 服务是一种接口,它描述了一组操作,可以使用标准的 XML 消息传递技术,经由网络访问这些操作。Web 服务可完成指定的任务或任务组。Web 服务采用标准的、形式化的 XML 概念进行描述,称之为服务描述。服务描述提供了与该服务进行交互所必须的所有细节,包括消息格式(即操作的细节)、传输协议和位置。