

微软.NET程序员系列

Microsoft Press

- ◆ 欧美读者评价 ★★★★★
- ◆ 微软权威专家精心编著
- ◆ 入围美国精品图书排行榜
- ◆ 诠释XML Web服务的原理、结构及实现技术
- ◆ 构建先进的Web分布式应用程序

构建XML Web服务

——基于MICROSOFT.NET平台

Building XML Web Services
FOR THE MICROSOFT .NET PLATFORM

[美] Scott Short 著

戴荣 马方平 吴健 等译
Visual Studio .NET 产品组 审校



Microsoft
.net

清华大学出版社

微软.NET 程序员系列

TP393.4
X47e

构建 XML Web 服务——基于 Microsoft .NET 平台

[美] Scott Short	著
戴 荣 马方平 吴 健	等译
Visual Studio .NET 产品组	审校

清 华 大 学 出 版 社

(京)新登字 158 号

内 容 简 介

本书是一本全面的XML Web服务开发指南,可帮助开发人员构建最高级的分布式Web应用程序。本书详细介绍了构建分布式对象Web应用程序需要了解的体系结构、协议和编程准则等,深入介绍了XML Web服务基本构建块:简单对象访问协议(SOAP)、Web服务描述语言(WSDL)、通用描述发现和集成(UDDI)以及Microsoft .NET框架技术,并结合示例和实用策略向读者演示如何构建自己的高端Web服务。此外,还介绍了Microsoft .NET My Services和其他基于XML的技术。

本书是 Web 服务开发人员的必备参考手册,也适合于相关程序设计人员阅读。

Building XML Web Services for the Microsoft .NET Platform

Microsoft Press

Copyright © 2002 by Microsoft Corporation

Original English language edition published by Microsoft Press, a Division of Microsoft Corporation.

All rights reserved.

No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher. For sale in the People's Republic of China only.

本书中文简体版由 Microsoft Press 授权清华大学出版社出版发行,未经出版者书面许可,不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号:图字 01-2002-1591 号

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

构建 XML Web 服务:基于 Microsoft .NET 平台/(美)斯考特著;戴荣等译.—北京:
清华大学出版社

(微软.NET 程序员系列)

书名原文: Building XML Web Serviers for the Microsoft .NET Platform

ISBN 7-302-05986-1

I. 构... II. ①斯... ②肖... ③戴... III. 互联网络—程序设计 IV. TP393.4

中国版本图书馆 CIP 数据核字(2002)第 079102 号

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)

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

<http://www.tup.tsinghua.edu.cn>

责任编辑:李祥青

印 刷 者:世界知识印刷厂

发 行 者:新华书店总店北京发行所

开 本: 787×960 1/16 印张: 23 彩插页: 4 字数: 482 千字

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

书 号: ISBN 7-302-05986-1/TP·3568

印 数: 0001~4000

定 价: 49.00 元

《微软.NET 程序员系列》序

自 2000 年 6 月微软宣布自己的 .NET 战略以来，在不到两年的时间里，.NET 已经从战略变成现实。.NET 带来了全新的、快速而敏捷的企业计算能力，也给软件开发商和软件开发人员提供了支持未来计算的高效 Web 服务开发工具。作为微软 .NET 战略的重要组成部分——Visual Studio .NET (中文版)已经于 2002 年 3 月 22 日正式在中国推出。

Visual Studio .NET 是一个功能强大、高效并且可扩展的编程环境。它充分展现了应用程序开发的潜能，并提供了生成应用程序所需的工具和技术。这些应用程序将给当今的企业、机构提供强大的支持，并推动下一代基于 XML Web 服务软件的发展。

有了 Visual Studio .NET，那些对全世界数百万的专业和业余程序员来说曾一度极端复杂、费时费力，甚至让人望而生畏的编程任务现在已不再神秘。更重要的是，Visual Studio .NET 使开发人员能运用既有的技能和知识来迎接新的编程挑战。

在 10 年前，Visual Basic 1.0 成为数以百万计的开发人员的革命性的应用程序开发语言。现在，Visual Studio .NET 为未来的 10 年做好了准备。

微软出版社为了配合 Visual Studio .NET 的推广以及 .NET 技术的普及，邀请 Visual Studio .NET 项目开发组的核心开发人员和计算机图书专业作家精心编写了英文版《微软.NET 程序员系列》丛书；该丛书自面市以来，在美国图书销量排行榜上一直高居前列，颇受好评，成为程序开发人员和网络开发人员了解 .NET 技术的权威工具书。尤其是《Microsoft .NET Framework 程序设计》一书，长期占据美国及欧洲此类书籍的排行榜冠军位置，程序开发人员不可不读此书。

清华大学出版社为了满足中国广大程序开发人员、网络开发人员学习最新技术的渴望，在微软出版社的配合下，从《微软.NET 程序员系列》这套丛书中精选了 40 本翻译成中文，以满足国内广大读者的需要。这套丛书阵容庞大，几乎涵盖了 .NET 技术及其应用的各个方面；也正因为如此，翻译和编辑加工的工作量也大得惊人。但为了保持国外优秀技术图书的魅力，同时使读者领会新技术的真谛，本丛书的翻译和编辑都是经过严格筛选的、具有很高的翻译水平或丰富编辑经验的技术人员；另外，我们还聘请微软公司 Visual Studio .NET 产品组的技术专家审读每一本书，确保在技术上准确无误。

相信这套丛书定会帮助程序开发人员、网络开发人员以及那些具有一定编程基础的中、高级读者，快速、全面掌握 .NET 技术，协助他们为技术生涯的下一个 10 年做好准备，为培养新一代软件人才，并推动中国软件产业的快速发展起到积极的作用！

这套丛书中的《C#技术内幕》和《C#编程技术》已于去年与读者见面，并得到读者的广泛好评。

目前, 本从书中已出版和在编的共有 22 本, 已从 6 月份起陆续和读者见面。这些书目包括:

- **《Microsoft .NET Framework 程序设计》**
全面、详细地介绍了 Microsoft .NET Framework, 可以帮助开发人员和设计人员轻松、高效地创建高性能且安全可靠的 .NET 应用程序。
- **《应用程序升级——Visual Basic 6.0 到 Visual Basic .NET》**
升级 Visual Basic 6.0 程序代码的最佳指导。涵盖了 Visual Basic .NET 的全部新功能和各种方案, 这些方案能够以最小的中断进行代码的移植并维护混合环境。
- **《Visual C++ .NET 托管扩展程序设计》**
Visual C++ 权威专家撰写。全面阐述了 Visual C++ .NET 托管扩展, 讲解如何编写 .NET 库及应用程序。内容包括托管扩展的不同编程规则, 以及 Visual C++ 等兼容 .NET 的编程语言所具有的多种新特性等。
- **《Visual Basic .NET 实用标准》**
包含针对面向对象编程、文件操作、解决方案分发及更多最佳经验, 包括代码示例和将标准应用于 Visual Basic .NET 语法的建议。
- **《ASP .NET Web 应用程序开发新思维》**
介绍了最新 Web 应用程序构建技术, 带给您 Web 应用程序开发的新思想。帮助开发人员充分利用 Microsoft .NET, 开发高效、安全的 Web 应用程序。
- **《构建 Web 解决方案——应用 ASP.NET 和 ADO.NET》**
介绍 ADO.NET 与 ASP.NET 的综合应用, 使开发人员可以将 Web 开发技术与数据库开发技术完美地结合起来, 构建功能更加强大的 Web 应用程序和服务。
- **《Microsoft Windows 网络编程 (第 2 版)》**
主要讲述 Winsock 网络编程技术。详细介绍了如何编写高性能、可扩展的 Winsock 应用程序, 还讲述了如何使用 C# 开发 Winsock 程序。
- **《构建 XML Web 服务——基于 Microsoft .NET 平台》**
详细深入地讲解了用于创建 XML Web 服务的主要协议和工具, 并深入探讨了如何利用它们构建功能强大、高效的 Web 商业解决方案。还介绍了 Microsoft .NET My Services 以及其他基于 XML 的微软新技术。
- **《Microsoft .NET My Services 精解》**
该书由 My Services 开发组的专家共同编写。详细解释了各项 .NET 服务, 并介绍了 .NET My Services 消息界面模型、数据操纵语言、安全授权模型、系统文档结构以及管理模型, 助您创建出充分利用 .NET My Services 优点的 Web 服务。
- **《移动设备 .NET 应用程序设计》**
介绍了如何使用 ASP.NET、Visual Studio .NET 和移动通信 Internet 工具箱创建 Web 应用程

序，在各种移动设备上以适当格式动态显示相同页面。有助于学习如何使用微软的信息服务器，以及如何为移动设备提供 E-mail 访问。

- **《SQL Server 2000 与 Visual Basic .NET 编程》**

介绍如何使用 SQL Server 2000、Visual Basic .NET、ADO.NET、ASP.NET、XML Web 服务和其他数据工具，开发数据库访问应用程序。

- **《SQL Server 2000 与 XML 编程》**

本书内容是包含代码和指导的实用方法，介绍了如何为各行业、电子商务和 Web 创建强大的 XML 数据库应用程序。

- **《Microsoft .NET Server 企业解决方案》**

介绍如何计划、开发和部署企业电子商务解决方案。将理论与实际紧密结合，详细讲解了如何解决电子商务集成时遇到的问题，并进一步探讨了使用 Microsoft .NET Server 的各种工具和技术解决具体的问题。

为了使读者的学习目标更为明确，在本丛书中，又划分出了两个子系列，现有 9 本书：

- **Core Reference 系列**

目前共有 6 本：《Visual C++ .NET 技术内幕（第 6 版）》、《Visual Basic .NET 技术内幕》、《Visual C# .NET 技术内幕》、《ADO.NET 技术内幕》、《Microsoft .NET 程序设计技术内幕》和《Visual J# .NET 技术内幕》。已经列入我们的出版计划的还有《Microsoft .NET Framework 技术内幕》一书。

这些书分别介绍相应编程语言和 .NET 框架类库的指令、示例代码、最佳编程惯例和基于案例的解决方案，内容详尽，讲解深入，集指导性与实用性于一体，最适合相关的开发人员用作专业读本。

- **Language Reference 系列**

目前共有 3 本：《Visual Basic .NET 语言参考手册》、《Visual C# .NET 语言参考手册》和《Visual C++ .NET 语言参考手册》。以简洁、易于浏览和使用的形式，从 A 到 Z 列出了这 3 种编程语言的技术参考。

随着技术的发展，我们将根据读者的需要，不断增加新的书目。

丛书版式特色

本丛书在风格上力求文字精炼。并采用小 5 号字编排，内容紧凑，版面清晰美观，易于阅读。此外，书中还安排了一些特色段落，提供正文之外的一些细节知识：



注意：提醒阅读和操作过程中应注意的事项，避免出现错误或问题。



提示：指点一些操作捷径或实用技巧，使您少走弯路，阅读和操作更为高效。



要点：总结关键知识点或操作细节，助您适时掌握要领。



注：提示首次出现的编程元素，以及书中涉及到该元素的其他位置以供参考。

尽管我们倾心相注，精心而为，总有疏忽纰漏，恳请广大读者不吝赐教与指正，我们定会全力改进，以期在后续工作中得以完善。

本丛书在创作过程中得到了微软(中国)公司的大力支持。本丛书能够顺利出版，更是倾注了无数幕后人员的汗水和心力。在此，对他们的辛勤劳动一并表示衷心感谢！

编者

2002 年 6 月

前言

几乎在任何一本技术杂志、开发人员参考手册或者公司 IT 战略的文档中，您都可以看到有关 **Web** 服务的内容。那么所有这些广告宣传究竟是怎么回事？简单地说，就是允许开发人员通过 **Web** 服务创建不受限制的应用程序——这些应用程序可用于不同的操作系统、硬件平台，并且不受地域的限制。本书将解释什么是 **Web** 服务，以及如何应用 **Microsoft .NET** 平台来构建和使用它们。

读者对象

本书面向具有一定编程基础的中、高级读者群。不管您熟悉哪种编程平台，都至少应该掌握面向对象编程(OOP)的概念和基本的编程构造。

此外，您还应该熟悉基本的 **C#** 语法。本书所有示例都是用 **C#** 语言编写的。这些示例非常简单，即使您原来的开发语言不是 **C#**，也可以很容易地将它们导入其他 **.NET** 语言，如 **Visual Basic**。

全书概览

本书采用由浅入深的方法来介绍 **Web** 服务以及 **.NET** 平台为创建和使用这些服务所提供的支持。如果您对 **Web** 服务不太了解，则很难深刻理解本书的内容。所以，本书前两章介绍了必要的背景知识。第 1 章解释了 **Web** 服务的基本原理，并对底层协议做了概述，最后介绍如何结合这两者设计出完整的解决方案。

第 2 章进一步讲述如何使用 **Visual Studio .NET** 来创建和使用 **ASP.NET** 平台上的 **Web** 服务。本章的主要目的是让您了解 **ASP.NET** 运行库如何为开发人员抽象底层协议，并说明在 **Web** 服务的环境中何时运用这些协议。

在第 3 章到第 5 章中，您将详细了解到核心的底层 **Web** 服务协议。如果您对 **Web** 服务还不甚了解，则可以跳过这些章节，以后再阅读。通过深入了解底层协议来理解 **Web** 服务是最好的方法，如果需要在各种平台上混合运用 **Web** 服务，则更是如此。

第 6 章到第 8 章是本书的核心部分，讲述了开发人员用来快速构建和使用 **Web** 服务的核心 **.NET** 技术：**ASP.NET** 和远程处理。这两种技术看似重复，其目标却截然不同：**ASP.NET** **Web** 服务的主要目的是维持在客户端和服务端之间传递的 **XML** 数据类型的实例的精确度；而远程处理则主要在于维持在客户端和服务端之间传递的 **.NET** 类型的实例的精确度。同时，这两个目标可以通过一组统一的技术来实现。

本书其余章节还涉及到了和大多数产品质量的 Web 服务相关的主题。第 9 章说明了如何利用 UDDI 和 DISCO 来为 Web 服务做广告宣传,以及如何发现其他的 Web 服务。第 10 章讲解了确保 Web 服务安全的策略。第 11 章介绍了如何对 Web 服务进行调试。第 12 章提供了确保 Web 服务满足伸缩性和可用性满足要求的策略。第 13 章分析了构建 Web 服务所涉及的一些问题,并介绍一些解决这些问题的新技术。

快速阅读

如果在参加一个项目时,需要使用我们知之甚少或者根本不了解的技术,我们就会设法去了解一些正好足以解决手中问题的知识。针对这种情况,本书内容安排的一个特点就是每章内容都可以独立阅读,并不需要以前面章节的内容作为背景。例如,您可以从讲述 ASP.NET 的第 6 章开始,而不需要首先阅读有关 SOAP、XML 架构或 WSDL 的章节。因此,如果您想尽可能快地开发和使用 Web 服务,建议您先阅读如下内容。

快速浏览第 1 章,了解组成 Web 服务的技术和协议是如何组织到一起的。

阅读第 2 章,并加载 Visual Studio .NET,按照书中给出的建立两个简单的应用程序的步骤进行操作。这将帮助您熟悉如何使用 Visual Studio .NET 工具集开发基本的 Web 服务。

选择学习第 6 章中适用于您的项目的重要内容。

阅读第 11 章中的“交互式调试”部分,这是关于使用 Visual Studio .NET 调试器来调试 Web 服务的内容。

根据需要阅读第 9 章和第 10 章(关于发现和安全)。

在项目的整个过程中,仔细阅读第 6 章和第 7 章,大致了解如何使用 ASP.NET 平台来建构和使用 Web 服务的内容。

最后根据项目需要阅读其他章节。比如,如果计划使用 UDDI,就阅读第 9 章中关于发布 Web 服务和发现其他 Web 服务的相关信息。

系统要求

要运行本书中的所有范例,您需要拥有表 I.1 所列出的硬件和软件。

表 I.1 硬件和软件要求

组 件	要 求
Visual Studio .NET	Enterprise Architect 版、Enterprise Developer 版、Professional 版或 Academic 版
处理器	Pentium III、600 MHz 或更高

续表

组 件	要 求
内存	对 Windows 2000 或者 Windows XP Professional 要求 128 MB 以上；对 Windows .NET Server 要求 256 MB 以上
硬盘	在系统盘上有 500 MB，在安装的驱动器上有 3 GB 空间
操作系统	Windows .NET Server、Windows XP、Windows 2000 或者 Windows NT 4.0
CD-ROM 或者 DVD-ROM 驱动器	需要
显示器	800×600 真彩(16 位或者更高)
鼠标	Microsoft 鼠标或者兼容的定位设备

示例代码

访问本书封底所示的示例代码下载地址，可下载书中采用的大部分示例的源代码。将示例代码复制到本地机器上就可以使用了。

支持

作者和编辑都尽了最大的努力来保证本书中的内容和示例代码的准确无误。尽管如此，在文中还是不可避免地会出现一些错误和疏漏。您可以查看公布在 <http://www.microsoft.com/mspress/support/> 上的更新和勘误表。

如果您发现了错误，或者对本书有什么建议，请发信至下列地址之一：

邮政地址

Microsoft Press
Attn: Building XML Web Services for the Microsoft .NET Platform Editor
One Microsoft Way
Redmond, WA 98052-6399

电子邮件

mspinput@microsoft.com

请在电子邮件的标题行或正文中提及本书的书名。

请注意，上述地址并不提供产品支持。要了解有关产品支持的更多信息，请访问 <http://www.microsoft.com/support>。

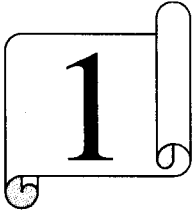
目 录

前言	IX	3.3.2 复合类型	33
第 1 章 Web 服务概述	1	3.3.3 通过引用传递参数	40
1.1 Web 服务构建块	3	3.4 协议绑定	45
1.2 Web 服务设计决策	4	3.5 本章小结	47
1.2.1 选择传输协议	4	第 4 章 XML Schema	49
1.2.2 选择编码模式	5	4.1 描述 XML 文档	50
1.2.3 选择格式设置约定	5	4.2 内置数据类型	51
1.2.4 选择描述机制	6	4.2.1 整数	52
1.2.5 选择发现机制	7	4.2.2 字符串	52
1.3 Web 服务中的欠缺	7	4.2.3 二进制数据	54
1.4 本章小结	8	4.3 命名空间	55
第 2 章 创建基本的 Web 服务	9	4.3.1 targetNamespace 属性	56
2.1 简单的商务应用程序	9	4.3.2 xmlns 属性	56
2.1.1 创建一个 Web 窗体	9	4.3.3 schemaLocation 属性	58
2.1.2 创建 Payment Web 服务	12	4.3.4 noNamespaceSchemaLocation	
2.1.3 更新 Order Web 窗体	14	属性	59
2.2 Web 文件共享应用程序	16	4.3.5 XML Schema 和 XML Schema	
2.2.1 创建 WebFileShare Web		Instance 命名空间	60
服务	16	4.4 元素定义	60
2.2.2 创建 WebFileUtil 程序	18	4.5 自定义数据类型	61
2.3 本章小结	22	4.5.1 简单类型	61
第 3 章 SOAP	23	4.5.2 复杂类型	66
3.1 SOAP 消息剖析	24	4.5.3 元素和属性组	73
3.1.1 SOAP 角色	25	4.5.4 命名空间的范围	73
3.1.2 Header 元素	25	4.5.5 多态	75
3.1.3 Body 元素	28	4.5.6 约束继承	79
3.2 使用 SOAP RPC 消息	30	4.6 本章小结	81
3.3 SOAP Encoding	32	第 5 章 使用 WSDL 记录 Web 服务	83
3.3.1 简单类型	33	5.1 WSDL 文档语法	84

5.1.1 definitions 元素	85	第 7 章 XML 序列化	180
5.1.2 types 元素	85	7.1 控制 XML 序列化	181
5.1.3 message 元素	88	7.2 定义 PurchaseOrder 根数据类型	182
5.1.4 portType 元素	89	7.3 定义 Items 数组	188
5.1.5 binding 元素	91	7.4 创建派生数据类型	192
5.1.6 service 元素	93	7.5 创建开放式 PurchaseOrder 架构	194
5.1.7 扩展性元素	94	7.6 定义 AcceptPO Web 方法	196
5.1.8 SOAP 扩展	94	7.7 服务器端验证	196
5.1.9 HTTP GET/POST 扩展	98	7.8 实现自定义序列化	197
5.1.10 import 元素	104	7.9 本章小结	198
5.1.11 记录文档	104	第 8 章 使用远程处理建立和使用 Web 服务	199
5.2 Calculator Web 服务的 WSDL 文档	105	8.1 远程处理与 ASP.NET	199
5.3 本章小结	110	8.2 Grabber.NET 应用程序	200
第 6 章 ASP.NET	111	8.2.1 远程处理的结构	201
6.1 创建 ASP.NET Web 服务	111	8.2.2 创建 IIS 承载的 Web 服务 ..	203
6.2 传输协议和绑定	117	8.2.3 创建 WinForm 承载的 Web 服务	207
6.3 Web 服务文档	119	8.2.4 访问 Web 服务	210
6.4 引发错误	122	8.2.5 添加 SOAP 头	216
6.5 SOAP 编码样式	125	8.2.6 生成 WSDL	219
6.6 接口继承	131	8.2.7 WSDL 扩展元素 Suds	220
6.7 管理状态	135	8.3 本章小结	220
6.7.1 会话状态	136	第 9 章 Web 服务的发现机制	222
6.7.2 应用程序状态	140	9.1 UDDI	222
6.8 定义和处理 SOAP 头	144	9.1.1 UDDI 体系结构	223
6.9 使用 SOAP 扩展	153	9.1.2 UDDI API	224
6.9.1 SOAP 扩展属性	153	9.1.3 UDDI SDK	226
6.9.2 SOAP 扩展类	155	9.1.4 UDDI 企业服务器	227
6.10 使用 WSDL 实用程序生成 代理代码	167	9.1.5 注册买方	228
6.10.1 代理类	169	9.1.6 注册供方	242
6.10.2 cookie 程序	177	9.1.7 Visual Studio .NET 集成	246
6.11 本章小结	178	9.2 DISCO	248

9.3 本章小结	251	11.3 规范化 Web 服务	283
第 10 章 建立安全的 Web 服务	253	11.3.1 跟踪	283
10.1 威胁模型简介	253	11.3.2 事件日志	292
10.1.1 分析威胁	254	11.3.3 性能计数器	294
10.1.2 选择可减轻威胁的技术	255	11.4 调试提示与技巧	298
10.2 Web 服务安全技术	257	11.5 本章小结	299
10.2.1 Web 服务身份验证	258	第 12 章 伸缩性和可用性	300
10.2.2 Web 服务授权	262	12.1 扩展 Web 服务的规模	300
10.2.3 Web 服务的保密性和 完整性	263	12.1.1 升级	301
10.3 .NET 框架中的安全技术	264	12.1.2 扩展	301
10.4 未来的 Web 服务安全技术	266	12.1.3 克服伸缩性瓶颈	305
10.5 常见的安全错误	266	12.2 维持高可用性	306
10.5.1 错误#1: 以不安全的方式 存储机密数据	267	12.2.1 高度可用的升级资源	307
10.5.2 错误#2: 不正确地连接到 SQL Server	268	12.2.2 高度可用的扩展资源	308
10.5.3 错误#3: 建立不安全的 SQL 字符串	268	12.2.3 根据高度可用的 资源编程	309
10.6 一个完整的示例	269	12.3 第三方 Web 服务和可用性	309
10.6.1 不安全的版本	269	12.3.1 把故障转移到备用 Web 服务	310
10.6.2 安全的解决方案	270	12.3.2 创建脱机操作模式	314
10.7 本章小结	273	12.4 性能优化	315
第 11 章 调试 Web 服务	274	12.5 本章小结	322
11.1 交互式调试	274	第 13 章 Web 服务的前景	323
11.1.1 调试基础	274	13.1 .NET My Services 简介	323
11.1.2 远程调试	275	13.1.1 .NET My Services 安全	326
11.1.3 Web 服务友好的调用 堆栈	276	13.1.2 使用 .NET My Services	327
11.2 调试器所需的信息	278	13.1.3 .NET My Services 请求示例	329
11.2.1 程序集元数据	279	13.2 通用型 XML Web 服务体系 结构(GXA)	331
11.2.2 程序数据库	279	13.2.1 WS-Inspection	332
11.2.3 跟踪信息	280	13.2.2 WS-Security 和 WS-License	334
11.2.4 调试动态编译的源代码	282	13.2.3 WS-Routing	340

13.2.4	WS-Referral.....	342
13.2.5	动态应用程序拓扑	345
13.3	Web 服务编排	346
13.4	本章小结	350
附录	XML Schema 内置类型.....	352



Web 服务概述

目前，基于组件的编程越来越流行。现在创建的任何一个应用程序几乎都会以某种形式利用通常来自于不同供应商的组件。随着应用程序越来越复杂，人们对分布在远程机器上的组件进行利用的需求也在不断增长。

基于组件的应用的一个例子就是端对端的电子商务解决方案。驻留在 Web 农场中的电子商务应用程序需要向后端的 ERP(企业资源规划，Enterprise Resource Planning)应用程序提交订单。在很多情况下，ERP 应用程序驻留在不同的硬件上，而且可能运行在不同的操作系统上。

Microsoft 的分布式组件对象模型(Distributed Component Object Model, DCOM)是一个分布式对象基础结构，它允许应用程序调用安装在另一台服务器上的组件对象模型(Component Object Model, COM)组件。DCOM 已可以与许多非 Windows 平台对接，但是它在这些平台上还未被广泛接受，所以很少利用它来在 Windows 和非 Windows 计算机之间进行通信。ERP 软件供应商常为通过专用协议与后端系统通信的 Windows 平台创建组件。

电子商务应用程序所利用的某些服务可能根本就没有驻留在数据中心内。例如，如果电子商务应用程序接受顾客用信用卡付账，它就必须调用商业银行的服务来处理顾客的信用卡信息。但是出于实用的缘故，DCOM 和相关的技术，比如 CORBA 和 Java RMI，都仅限于企业数据中心所安装的应用程序和组件。这个问题的两个主要原因是这些技术都默认利用专用协议，而且这些协议本质上都是面向连接的。

通过 Internet 与服务器进行通信的客户端面临大量与服务器进行通信的潜在障碍。实际上，世界各地具有安全意识的网络管理员都已采用企业路由器和防火墙来保护网络，禁止通过 Internet 来进行任何类型的通信。通常情况下，要让网络管理员打开多于最低限度数量的端口是非常之难的。

或许您比较幸运，能够让网络管理员打开相应的端口来支持您的服务，但您的客户端就不一定会如此幸运。因此，像 DCOM、CORBA 和 Java RMI 所使用的专用协议对于 Internet 的情况并不实用。

与这些技术有关的另一个问题是，由于它们本质上是面向连接的，因此不能很好地处理网络中断。因为 Internet 并非在我们的直接控制之下，所以不能对连接的质量和可靠性做任何假设。如果发生网

络中断，客户端对服务器的下一次调用就可能失败。

这些技术面向连接的特性也使得建立一个对于获得高伸缩性所必需的负载平衡的基础结构更有挑战性。一旦客户端和服务端之间的连接被切断，问题就不是只将下一个请求路由到另一个服务器上那么简单了。

开发人员已经试图通过利用称为无状态编程(stateless programming)的模型来克服这些局限，但只取得了有限的成果，因为这些技术相当复杂，所以与远程对象重建连接的成本很高。

因为顾客信用卡的处理是由 Internet 上的远程服务器完成的，所以 DCOM 并不是增进电子商务客户端和信用卡处理服务器之间通信的理想选择。比如在一个 ERP 解决方案中，客户端的数据中心常安装有第三方组件(本例中是由信用卡处理方案的供应商提供)。该组件的作用不过是充当代理，使电子商务软件和商业银行能够通过专用协议进行通信。

您是否已看到了一种模式？由于现有技术在促进计算机系统之间的通信方面有一定局限性，所以软件供应商通常采取构建自己的基础结构的方式。这就意味着本应用于改进 ERP 系统或者信用卡处理系统功能的资源被用来编写专用的网络协议了。

为了更好地支持通过 Internet 进行通信，Microsoft 最初采用的策略是提升其现有技术，包括 COM Internet Services (CIS)。CIS 可让您通过端口 80 在客户端和远程组件之间建立 DCOM 连接。但由于各种原因，CIS 并没有被广泛接受。

显然我们需要一个新的方案。因此 Microsoft 决定完全解决这个问题。让我们来看看一个解决方案要获得成功所必须要满足的一些要求：

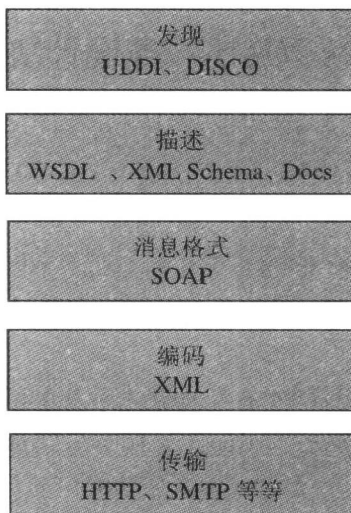
- **互操作性** 远程服务必须能为其他平台上的客户端所使用。
- **Internet 友好性** 该解决方案必须能很好地支持客户端通过网络访问远程服务器。
- **强类型接口** 远程服务所收发的数据类型必须是确定的。而且，远程服务所定义的数据类型应该能很好地映射为由大多数编程语言所定义的数据类型。
- **利用现有 Internet 标准的能力** 远程服务的实现应该尽可能地利用现有的 Internet 标准，避免重复开发解决问题的方案。基于已被广泛采纳的 Internet 标准构建的解决方案可利用现有的工具包和为该技术创建的产品。
- **支持任何语言** 该解决方案不应和某种编程语言紧密联系在一起。比如，Java RMI 和 Java 语言紧密联系在一起，要想从 Visual Basic 或 Perl 调用远程 Java 对象的功能就会很困难。无论使用哪一种编程语言编写客户端，客户端都应该能够实现新的 Web 服务或使用现有的 Web 服务。
- **支持任何分布式组件基础结构** 该解决方案不应和特定的组件基础结构紧密结合。事实上，不应只为了构建一种新的远程服务或使用现有服务而需要购买、安装或者维护一个分布式对

象基础结构。应该能够利用底层协议在现有的分布式组件对象基础结构(如 DCOM 和 CORBA) 之间进行基本的通信。

正如本书的名称所示, Microsoft 所创建的解决方案被称为 Web 服务并不令人觉得奇怪。Web 服务公开了一个接口, 以便代表客户端来调用特定的活动。客户端可通过使用 Internet 标准来访问 Web 服务。

1.1 Web 服务构建块

在两个应用程序之间进行远程通信所需的核心构建块如下图所示:



下面来讨论每个构建块的用途。因为许多读者都熟悉 DCOM, 所以这里还将介绍每个构建块的 DCOM 等效项。

- **发现** 要访问 Web 服务所公开功能的客户端应用程序需要一种方式来解析远程服务的位置。这通常是通过一个称为发现(discovery)的过程完成的。通过集中化的目录和其他特殊方法都可以实现发现。在 DCOM 中, 发现服务由 Service Control Manager (SCM)提供。
- **描述** 一旦解析出了特定 Web 服务的端点, 客户端就需要足够的信息来与它正确交互。Web 服务的描述既包括有关客户端应用程序将要使用的接口的结构化元数据, 也包括有关 Web 服务文档(包括用法示例)。DCOM 组件通过类型库来公开关于接口的结构化元数据。组件的类型库中的元数据是以专用的二进制格式存储的, 可以通过专用应用程序编程接口(API)进行访问。
- **消息格式** 为了交换数据, 客户端和服务端必须在消息编码和格式设置的方式上取得一致。标准的数据编码方式可保证由客户端进行编码的数据能被服务器正确转换。在 DCOM 中,