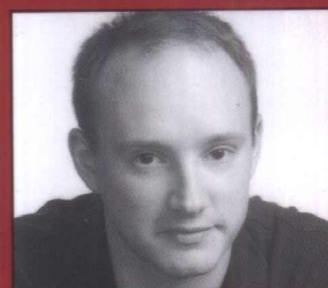
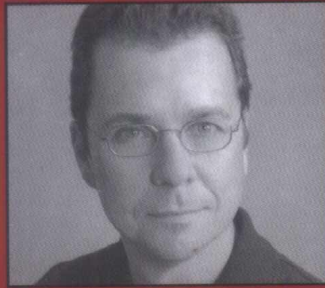
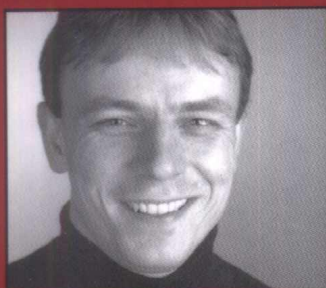


PROGRAMMER TO PROGRAMMER™



PROFESSIONAL ASP.NET Web Services

ASP.NET

Web 服务高级编程

Russ Basiura

Mike Batongbacal 等著 康博 译



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



ASP.NET Web 服务高级编程

Russ Basiura 等著

Mike Batongbacal

康 博 译

清 华 大 学 出 版 社

(京) 新登字 158 号

北京市版权局著作权合同登记号: 01-2001-4316

内 容 简 介

基于 XML 的 Web 服务被广泛认为是分布式计算技术的重要发展方向。在业界众多公司的推动下,人们对于逐渐形成的各种 Web 服务的相关技术与标准越来越关心。

本书主要介绍如何使用 ASP.NET 创建高水平的 Web 服务。书中介绍了 Web 服务架构的核心标准,并通过实例说明了这些标准如何集成到 ASP.NET 中。本书还通过深入分析代码实例介绍了创建和使用 Web 服务的技术。本书最后介绍了三个综合的案例研究,每一个案例研究都演示了一个完整的 Web 服务应用程序的解决方案。

本书面向具有一定经验的中高级 ASP.NET 程序员。

Russ Basiura, Mike Batongbacal: Professional ASP.NET Web Services

EISBN: 1-861005-45-8

Copyright© 2001 by Wrox Press Ltd.

Authorized translation from the English language edition published by Wrox Press Ltd.

All rights reserved. For sale in the People's Republic of China only.

Chinese simplified language edition published by Tsinghua University Press.

本书中文简体字版由清华大学出版社和英国乐思出版公司合作出版。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。

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。

书 名: ASP.NET Web 服务高级编程

作 者: Russ Basiura Mike Batongbacal 等著 康博 译

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.tsinghua.edu.cn>

责任编辑: 冯志强

印 刷 者: 北京牛山世兴印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 **印 张:** 41.25 **字 数:** 978 千字

版 次: 2002 年 6 月第 1 版 2002 年 6 月第 1 次印刷

书 号: ISBN 7-302-05464-9/TP·3221

印 数: 0001~5000

定 价: 69.00 元

前 言

Web 服务(Web Service)的出现被认为是分布式计算领域的一项重大飞跃。Web 服务预示着这一美好未来:当用户需要某种功能时,可以在 Internet 上以编程方式找到并访问。当然,有些批评家不以为然,认为 Web 服务仅仅是老的分布式计算系统的标准化翻版而已。

不管 Web 服务是一大步进展还是一小步革命,我们都必须承认它带来了许多有趣的技术。Web 服务蕴含的技术与标准被计算机业界的各类公司所采用,IBM、Sun Microsystems、Microsoft 等公司更是抢占先机。

IBM 和 Sun 公司选择 Java 语言作为 Web 服务的优选开发平台,对 Java Web 服务开发者提供各种工具与技术支持。Microsoft 公司最初提供工具支持 Visual Basic 和 Active Server Pages 开发者创建 Web 服务。在最新版本的 Active Server Pages,即 ASP.NET 中,Microsoft 公司已经完全包含了 Web 服务,将 Web 服务的功能嵌入到系统自身中。

ASP.NET 中应用 Web 服务的功能变得非常简单,这并不是说在 ASP.NET 中开发 Web 服务不再需要任何学习了。尽管 Web 服务是从以前的分布式技术标准发展来的,但是它毕竟有许多新特征,要想充分利用它,对此必须加以认真研讨。

0.1 本书内容

本书力图对支撑 Web 服务的各项技术以及使用 ASP.NET 开发 Web 服务的方法做全面的介绍。

介绍 Web 服务

本书开始时首先介绍 Web 服务。第 1 章说明 Web 服务是什么。第 2 章和第 3 章说明如何用 ASP.NET 实现和应用基本的 Web 服务。

Web 服务标准

接下来将详细讨论 Web 服务赖以应用的标准。第 4 章介绍 WSDL(Web Service Description Language, Web 服务描述语言),它能够精确地描述 Web 服务。

第 5 章介绍 SOAP(Simple Object Access Protocol,简单对象访问协议),它允许客户与所应用的 Web 服务通信。第 6 章将讨论如何利用 SOAP 扩展来增强其功能。

第 7 章介绍一个重要的 Web 服务概念 discovery(发现),并说明如何利用 UDDI(Universal Description, Discovery and Integration,通用描述、发现与集成)的强大功能。

Web 服务技术

本书的大部分内容将详细介绍对于开发 Web 服务至关重要的各种技术和相关问题。第 8 章



先概括介绍 Web 服务带来的应用程序设计问题。

第 9 章将比较同步与异步 Web 服务,并说明创建异步 Web 服务的额外的编程要求。

第 10 章将介绍 Web 服务中的 stateful(状态化)概念,并介绍这一概念带来的结构设计问题。

第 11 章讨论事务处理,以及如何在 Web 服务中应用事务处理。

第 12 章说明高速缓冲存储器如何增强 Web 服务的性能和可伸缩性。

第 13 章和第 14 章讨论安全议题,演示如何创建能够鉴别用户的 Web 服务,并说明 SOAP 如何使得与 Web 服务的通信变得安全可靠。

第 15 章介绍 Web 服务的一些实际应用:从各种数据源中获取数据。

第 16 章的内容将围绕在分析用户的商务 Web 服务时遇到的技术问题与设计问题展开。

案例研究

第 17 章是案例研究,其中的 Web 服务使用 BizTalk 编写。

第 18 章是另一个案例研究,说明如何使用 Web 服务创建可重用的认证框架。

第 19 章是最后一个案例研究,介绍了一个处理繁重任务的系统,这些任务分布到了许多计算机中,由 Web 服务来协调。

附录

附录 A 简单介绍了当前最大的 Web 服务项目: .NET My Services (以前叫做 Hailstorm), Microsoft 计划在其中通过 Web 服务提供各种各样的功能。

附录 B 介绍了 Internet Explorer 5.5 + 中的 Web 服务行为,它们允许当前各种版本的 Internet Explorer 访问 Web 服务。

0.2 本书读者对象

本书的读者对象是希望掌握创建 Web 服务的详细信息的 ASP.NET 程序员。本书没有介绍 ASP.NET 的基础知识,我们假定读者已经阅读过《ASP.NET 高级编程》一书,或者在阅读本书之前已经掌握了相关的基础知识。这样做可以使本书集中精力介绍与创建 Web 服务相关的更有价值的信息。

为什么使用 C#

本书的代码示例全部使用 C# 编写。代码示例的完整的 VB.NET 版本和 C# 版本都可以从如下站点下载 <http://www.wrox.com>。

本书的前几章中使用到一些 VB.NET 代码来介绍定义 Web 服务的语法问题,以便充分展示 VB.NET 与 C# 间的区别。

我们认为使用单一的语言有利于使得本书的叙述更清晰,内容更充实。对于在这两种主要的 .NET 语言间如何选择,本书做了深思熟虑的考虑。最终选择 C# 是因为它是一种全新的语言,非常适合展示 Web 服务这样的专业应用项目。

我们也努力确保 VB.NET 程序员能够理解书中的示例,站点中的 VB.NET 代码可以为他们

使用。

0.3 本书软硬件要求

运行本书示例代码需要如下软硬件配置:

- Windows 2000 或 Windows XP。
- ASP.NET。可以是重新分布的(其中包括 .NET SDK)或者附赠版本。Beta 版的 ASP.NET 可能会有一些不兼容问题。
- 有些示例演示了 Visual Studio .NET 对某些任务的用法,书中提供的 ASP.NET 代码不需要 Visual Studio .NET。
- 有些示例代码使用到 SQL Server 中的数据仓库。

示例的完整源代码可以从我们的如下站点下载: http://www.wrox.com/Books/Book_Details.asp?isbn=1861005458。

0.4 客户支持

我们一向重视读者的反馈意见,希望知道读者对本书的看法:批评、赞扬和建议都欢迎。电子邮件地址是 feedback@wrox.com。

如何下载本书的示例代码

访问 Wrox 站点 <http://www.wrox.com/>, 只需从 Search 栏找到书名, 或者使用书名列表。单击 Code 栏的 Download, 或者在本书的详细页面中单击 Download Code。

站点中可供下载的文件使用 WinZip 格式压缩。读者在硬盘的文件夹中保存附件后, 需要使用 WinZip 或者 PKUnzip 解压缩。解压缩文件后, 代码通常解压缩到相应章节的文件夹中。开始解压缩进程时, 要确保 WinZip 或 PKUnzip 设置的文件夹名称正确。

勘误表

尽管我们努力确保本书的文本和代码准确无误, 但是仍旧会有挂一漏万之处。读者若发现书中的错误, 如拼写错误或者代码运行错误, 请把勘误表发往电子邮件地址 support@wrox.com。

在 <http://www.wrox.com/> 站点可以找到勘误表, 通过 Advanced Search 或者书名列表找到书名, 单击 Book Errata 链接, 它在本书详细页面的图像下面。

电子邮件支持

如果读者希望就书中的问题直接向掌握详情的专家询问, 可以发电子邮件到 support@wrox.com, 电子邮件中要带上书名和 ISBN 书号的后 4 位数字。通常电子邮件中要包括如下内容:

- 书名和 ISBN 书号的后 4 位数字, 主要问题所在的页码。
- 读者的姓名、联系方法, 以及主要问题。



我们不会向读者寄送邮件，这样可以节省双方的时间。对于读者的电子邮件一般会经过以下处理流程：

- 客户支持：读者的电子邮件首先寄送给我们的客户支持人员，他们能够回答询问频率最高的一些问题，并能就书中或者 Web 站点的通常问题立即解答。

- 编辑：较难的问题会送给负责本书的技术编辑，他们熟悉编程语言和软件产品，能够回答相关的细节性技术问题。

- 作者：最后，对于编辑无法解答的问题，将转发给作者。尽管我们不愿意干扰作者的写作工作，仍旧会向他们转发一些具体的问题。全体 Wrox 作者都会对他们的著作提供技术支持。他们会向客户支持人员或者编辑发送电子邮件，使得所有读者都能收益。

Wrox 的技术支持工作只对直接与书中内容相关的问题负责，对于超出正常支持范围的问题，可以通过 <http://p2p.wrox.com/forum> 站点上的讨论列表获得支持。

p2p.wrox.com

作者之间的讨论加入到 P2P 邮件列表中。我们的独一无二的系统通过邮件列表、论坛和新闻组提供了程序员到程序员(programmer to programmer)之间的联系，这是一种一对一的电子邮件支持系统。读者向 P2P 发送电子邮件查询时，会被许多 Wrox 作者和当时在邮件列表中的业界专家看到。读者可以从 p2p.wrox.com 站点找到许多不同的邮件列表，它们不仅对阅读本书有用，对读者开发实际的应用程序也大有帮助。对于本书比较适合的邮件列表是 `aspx` 和 `aspx_professional`。

订阅邮件列表遵循以下步骤：

- 进入站点 <http://p2p.wrox.com/>。
- 从左侧的菜单栏中选择合适的类别。
- 单击希望加入的邮件列表。
- 遵循相关指令进行订阅，并输入您的电子邮件地址和密码。
- 确认希望接受电子邮件。
- 使用订阅管理器加入更多的邮件列表，并设置电子邮件优选项。

本系统为什么能提供最佳支持

读者可以加入邮件列表，并每周接收一次。如果读者没有时间，或者不方便接收邮件列表，可以搜索我们的在线档案室。垃圾邮件已经被删除，读者的电子邮件地址由独一无二的 Lyris 系统保护。任何加入或推出邮件列表订阅的请求，或者其他有关邮件列表的请求，都会发送到 listsupport@p2p.wrox.com。

目 录

第 1 章 Web 服务	1
1.1 基于组件的分布式计算与 Web 服务	2
1.2 为什么使用 Web 服务	4
1.2.1 Web 服务是一种优秀的分布式计算技术	4
1.2.2 防火墙能挫败系统管理员吗	6
1.2.3 SOAP 比较简单	6
1.2.4 混乱的现实	6
1.2.5 集中信息	7
1.2.6 Web 服务带来的生财之道	8
1.3 什么是 Web 服务	8
1.3.1 利用已有的技术	9
1.3.2 Web 服务堆栈	10
1.4 如何利用 ASP.NET 实现 Web 服务	15
1.5 本章小结	17
第 2 章 创建 ASP.NET Web 服务	18
2.1 使用 Notepad 创建 Web 服务	18
2.2 应用程序设置示例	18
2.2.1 把应用程序设置外置为 Web 服务	21
2.2.2 部署 Web 服务	24
2.2.3 测试 Web 服务	24
2.2.4 从传统 ASP 使用 Web 服务	28
2.2.5 使用独立的装配件	29
2.3 使用 Visual Studio .NET 创建 Web 服务	30
2.3.1 创建和运行 HelloWorld 示例	30
2.3.2 VS.NET 剖析: 生成的 Web 服务	33
2.3.3 为什么使用 Visual Studio .NET	34
2.4 进一步创建 Web 服务	35
2.4.1 WebService 指示符	35
2.4.2 WebService 属性	36
2.4.3 WebMethod 属性	39
2.5 从 WebService 类派生	47
2.6 本章小结	49



第 3 章 使用代理消费 Web 服务	50
3.1 代理的作用	50
3.1.1 电话类比	50
3.1.2 代理无处不在	51
3.2 创建代理类	52
3.2.1 使用 WSDL.EXE 创建代理类	53
3.2.2 引用 DLL	58
3.3 使用 Visual Studio .NET 创建代理	59
3.4 代理的分类	63
3.4.1 透明代理	63
3.4.2 实际代理	63
3.5 同步化和异步化	63
3.6 修改默认的代理设置	67
3.6.1 修改 URL	68
3.6.2 超时	68
3.6.3 代理服务器	71
3.6.4 编码	71
3.6.5 用户代理	71
3.6.6 重定向	72
3.7 本章小结	72
第 4 章 Web 服务描述语言	74
4.1 什么是 WSDL	74
4.2 如何在 .NET 和 COM 中创建 WSDL 文件	76
4.2.1 WSDL 文档和 COM 对象	76
4.2.2 WSDL 文档和 .NET	85
4.3 深入解析 WSDL 文档	88
4.3.1 样本 WSDL 文档	89
4.3.2 逐节浏览 WSDL 文档	92
4.4 WSDL 屏幕剪贴示例	105
4.4.1 HelloWorld 应用程序的 WSDL 文档	106
4.4.2 创建 HelloWorld 应用程序	107
4.5 本章小结	111
第 5 章 标准线上格式与数据类型	112
5.1 什么是线上格式	112
5.2 常用线上格式	113
5.2.1 HTTP GET	113
5.2.2 HTTP POST	114

5.2.3	BXXP/BEEP	114
5.2.4	ebXML	114
5.2.5	GIOP/IOP	114
5.2.6	XML-RPC	115
5.2.7	XMLP	116
5.2.8	SOAP	117
5.3	SOAP 简介	117
5.3.1	为什么使用 SOAP	117
5.3.2	SOAP 格式	120
5.3.3	SOAP 封套	120
5.3.4	SOAP 报头	121
5.3.5	SOAP 主体	124
5.3.6	编码数据类型	128
5.4	本章小结	140
第 6 章	自定义 SOAP 技术	141
6.1	SOAP 扩展	141
6.1.1	SOAP 扩展的工作原理	143
6.1.2	使用 SOAP 扩展	144
6.1.3	创建 SoapLog 扩展	145
6.2	定形 SOAP 消息	157
6.2.1	消息的完整编码	159
6.2.2	使用 XmlSerializer	162
6.2.3	使用属性定形 XML	166
6.3	传递复杂的数据类型	177
6.3.1	传递数据集	177
6.3.2	XML 字符串	185
6.3.3	二进制数据	190
6.4	本章小结	192
第 7 章	Web 服务发现机制	194
7.1	为什么需要发现机制	195
7.2	3 种主要的发现机制 XML 文件	196
7.3	.disco 文件	196
7.3.1	.disco 文件的结构	198
7.3.2	发现元素	199
7.3.3	contractRef 元素	200
7.3.4	discoveryRef 元素	201
7.3.5	schemaRef 元素	201



7.3.6 soapRef 绑定元素	202
7.4 .vsdisco 文件	202
7.5 添加 Web 引用	206
7.5.1 向工程中添加单个 Web 服务	206
7.5.2 向工程中添加多个 Web 服务	207
7.6 .map 文件	209
7.7 创建用户自己的 .disco 文件	210
7.8 创建用户自己的本地 Web 服务库	213
7.8.1 本地 Web 服务器上的 Web 引用	215
7.8.2 创建一个 Web 服务的 HTML 库	216
7.8.3 专家搜索引擎	217
7.8.4 推广 Web 服务	219
7.9 UDDI	219
7.9.1 UDDI Web 服务	219
7.9.2 谁来控制 UDDI	220
7.9.3 UDDI Web 服务定义	220
7.9.4 理解 UDDI WSDL 计划的结构	221
7.9.5 使用 ASP.NET SDK 搜索 UDDI 注册库	222
7.10 Disco.exe	223
7.11 发现机制的未来发展	225
7.12 本章小结	226
第 8 章 设计标准	227
8.1 Web 服务何时符合设计要求	227
8.1.1 消费 Internet 上的资源	228
8.1.2 销售商业服务	228
8.1.3 向合作伙伴外置商务逻辑	229
8.1.4 系统间内部互联	229
8.2 何时不使用 Web 服务	230
8.2.1 利用 DLL 内联到应用程序	231
8.2.2 .NET 间的通信使用 Remoting	231
8.2.3 有时最好不要使用分布式应用程序	232
8.3 协调现有的设计方法	232
8.3.1 架构问题	232
8.3.2 设计问题	234
8.3.3 性能问题	239
8.4 本章小结	240

第 9 章 异步化编程	241
9.1 同步化调用与异步化调用的比较	241
9.1.1 同步化处理实例	242
9.1.2 什么时候采用异步化处理比较好	243
9.2 Web 服务的例子	244
9.3 .NET 中的异步化编程	251
9.3.1 .NET 运行时环境提供的机制	252
9.3.2 委托的概念	252
9.4 异步化调用 Web 服务	254
9.4.1 深入了解 Web 服务的代理类	254
9.4.2 使用回撤的异步化方法调用	256
9.5 设计的考虑事项	259
9.5.1 超时处理	259
9.5.2 提供取消方法	260
9.5.3 异步化服务器处理	260
9.6 本章小结	261
第 10 章 管理状态	262
10.1 概述	262
10.2 ASP.NET 中的状态管理	263
10.2.1 Application 对象	264
10.2.2 Session 对象	266
10.2.3 状态管理需要考虑的事项	270
10.3 在 Web 服务里管理状态	270
10.3.1 继承 WebService 类	272
10.3.2 EnableSession 属性	273
10.3.3 创建一个支持状态的 Web 服务	273
10.3.4 为支持状态的服务开发客户程序	276
10.4 设计问题	278
10.4.1 应用程序状态	279
10.4.2 会话状态	281
10.5 本章小结	282
第 11 章 使用 Web 服务进行事务处理	283
11.1 提供事务性 Web 服务	283
11.1.1 继承来自 WebMethodAttribute 的支持	285
11.1.2 StatusUpdate WebMethod	286
11.1.3 我们的 Web 服务的输出	288
11.1.4 使用 Enterprise Services	291



11.2	使用事务性 Web 服务	293
11.2.1	消费我们的 StatusUpdate WebMethod	293
11.2.2	在事务处理中调用 Web 服务	296
11.3	本章小结	300
第 12 章	数据高速缓存	301
12.1	高速缓存概述	301
12.1.1	什么是高速缓存	301
12.1.2	高速缓存的优点和缺点	302
12.2	输出高速缓存	302
12.2.1	测试 Web 服务	303
12.2.2	输出高速缓存潜在的缺陷	307
12.3	数据高速缓存	307
12.3.1	使用 Application 对象高速缓存	307
12.3.2	使用 Cache 对象高速缓存	314
12.3.3	基于时间的高速缓存终止	319
12.3.4	基于从属高速缓存终止	321
12.3.5	目录从属高速缓存	325
12.3.6	多重文件从属高速缓存	325
12.3.7	基于时间和基于文件从属的高速缓存	331
12.3.8	基于键标的从属高速缓存	331
12.3.9	从 Cache 对象中手工移出高速缓存数据项	335
12.3.10	自动的高速缓存终止	336
12.3.11	高速缓存数据项优先级	336
12.3.12	使用回撤功能	338
12.4	高速缓存性能计数器	341
12.5	本章小结	346
第 13 章	验证	348
13.1	什么是安全机制	348
13.2	Web 服务的安全体系	349
13.3	Web 服务可用的安全选项	350
13.4	ASP.NET 验证体系	350
13.5	IP 和 DNS 层安全机制	351
13.5.1	IP 和 DNS 层安全机制的优点	351
13.5.2	IP 和 DNS 层安全机制的缺点	352
13.5.3	实现 IP 和 DNS 安全机制	352
13.6	Windows 安全机制	352
13.6.1	访问控制列表	353

13.6.2 接下来是什么	354
13.7 角色	355
13.8 基本验证	357
13.8.1 基本验证的优点	357
13.8.2 基本验证的缺点	358
13.8.3 使用的原因和场合	358
13.8.4 实现基本验证	358
13.9 简要验证	361
13.9.1 简要验证的优点	361
13.9.2 简要验证的缺点	361
13.9.3 使用的原因和场合	362
13.9.4 实现简要验证	362
13.10 集成 Windows 验证	362
13.10.1 集成 Windows 验证的优点	362
13.10.2 集成 Windows 验证的缺点	363
13.10.3 使用的原因和场合	363
13.10.4 实现集成 Windows 验证	363
13.11 表单验证	364
13.11.1 表单验证的优点	364
13.11.2 表单验证的缺点	364
13.11.3 使用的原因和场合	364
13.11.4 实现表单验证	364
13.12 SSL	370
13.12.1 SSL 的优点	371
13.12.2 SSL 的缺点	372
13.12.3 使用的原因和场合	372
13.13 结合 SSL 的基本验证	372
13.14 客户授权证书验证	375
13.14.1 使用的原因和场合	376
13.14.2 配置客户授权证书	377
13.14.3 怎样读取客户授权证书的内容	378
13.14.4 怎样验证用户	379
13.14.5 给代理外置客户授权证书	381
13.15 利用 Windows 账号映射验证客户授权证书	383
13.16 扮演	387
13.16.1 WindowsPrincipal 对象	387
13.16.2 实现	388
13.17 URL 验证	389



13.18	本章小结	391
第 14 章	SOAP 安全体系	392
14.1	SOAP 封套	392
14.2	自定义 SOAP 报头	393
14.3	自定义验证和授权	398
14.4	SOAP 扩展	405
14.4.1	用 SOAP 扩展可以做什么	405
14.4.2	跟踪 SOAP 消息	405
14.5	加密和解密 SOAP 消息	413
14.6	本章小结	421
第 15 章	通过 Web 服务外置数据	422
15.1	外置 ADO.NET	422
15.1.1	创建 ProductService Web 服务	423
15.1.2	创建 ProductSupply 客户机	428
15.2	创建强类型化的 Web 服务	431
15.2.1	创建类型化的数据集	433
15.2.2	创建 GetSupplierList()Web 方法	434
15.2.3	为 CompanyService 创建 Web 服务客户机	436
15.3	创建嵌套的 Web 服务	440
15.3.1	Building the CountrySupplier Web Service	440
15.3.2	快速浏览 CompanyByCountry 服务的 Web 服务客户机	443
15.4	外置 XML	444
15.4.1	创建 CreateXmlFile 控制台应用程序	444
15.4.2	创建 GetProductList()Web 方法	446
15.5	外置活动目录	447
15.5.1	MusicListeners Web 服务	447
15.5.2	使用 XML Designer 创建类型化的数据集	449
15.6	遗留系统和应用程序	449
15.7	本章小结	450
第 16 章	商务中的 Web 服务	451
16.1	需要解决的商务问题	452
16.1.1	内部的商务	452
16.1.2	外部的合作伙伴	453
16.1.3	推销给消费者	456
16.2	法律问题	457
16.2.1	屏幕剪贴和知识产权	457

16.2.2	安全性	458
16.2.3	QOS	459
16.2.4	许可	460
16.3	支付手段	461
16.3.1	接受验证	461
16.3.2	彻底购买	462
16.3.4	微量支付	462
16.3.4	订阅	463
16.3.5	Favorites 服务模式	464
16.3.6	盘点支付模式	465
16.4	商务案例研究	465
16.4.1	SalCentral	466
16.4.2	家中的 SETI	467
16.4.3	Microsoft 公司的 Passport	467
16.4.4	使用这一框架	468
16.5	向领导汇报 Web 服务	468
16.5.1	特点和优点: 无需协议和代码	468
16.5.2	放下自私和成见	469
16.5.3	制定计划	469
16.5.4	从客户的角度表达问题	470
16.5.5	关注需求和解决方案	470
16.6	本章小结	470
第 17 章	案例研究 —— BizTalk	472
17.1	BizTalk 概述	472
17.1.1	BizTalk Framework	472
17.1.2	BizTalk.org	473
17.1.3	BizTalk Server 2000	473
17.2	应用程序概述	474
17.3	定义消息	475
17.4	映像消息	478
17.5	消息处理端口	479
17.6	定义频道	480
17.7	开始工作	480
17.8	制定 Orchestration 进度表	482
17.8.1	“从队列获取消息”任务	484
17.8.2	COM 交互操作性	486
17.8.3	“信用卡预授权”任务	487



17.8.4	PreauthOK 任务	491
17.8.5	结束	493
17.9	本章小结	494
第 18 章	案例研究 —— 护照样式的验证服务	495
18.1	VPASS Web 服务概述	495
18.2	建立 Web 服务	496
18.2.1	软件需求	496
18.2.2	软件的安装与配置	497
18.3	VPASS 结构	502
18.3.1	应用程序概述	503
18.3.2	可伸缩性	504
18.3.3	VPASS 安全性	505
18.3.4	目录服务和 LDAP	505
18.3.5	对象和装配件	507
18.4	使用 Web 服务	516
18.4.1	测试服务	516
18.4.2	创建代理	517
18.4.3	应用程序接口	517
18.5	本章小结	527
第 19 章	案例研究——分布式处理	528
19.1	工作集和工作单元	530
19.1.1	应用程序开发逻辑	531
19.1.2	建立数据库	532
19.2	创建样本数据	532
19.3	创建服务	535
19.4	建立代理	538
19.4.1	创建工程	538
19.4.2	注册代理	549
19.4.3	创建 Distributed.LogAnalyzer	555
19.4.4	请求工作单元	565
19.4.5	返回结果单元	580
19.4.6	重复工作	591
19.5	负载测试	594
19.6	分析结果	598
19.7	本章小结	598