



Java Web 服务 编程指南

〔美〕 Mack Hendricks 等著
Ben Galbraith

卢 刚 高春荣 朱 军 等译
李双庆 审校

Professional Java Web Services

Java Web服务编程指南

〔美〕 Mack Hendricks 等著
Ben Galbraith

卢 刚 高春荣 朱 军 等译
李双庆 审校

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 提 要

本书向读者介绍了当前Web服务的核心技术, 全书共14章, 主要内容包括SOAP、WSDL和UDDI等。本书主要讲述的Web服务是当前程序设计领域中的一项最新技术, 它用中性平台标准(例如HTTP和XML)在分布式应用程序之间实现跨平台的远程过程调用(RPC)。本书是为有经验的网络开发者们编写的, 适合以Java语言作为Web服务开发平台的专业Java程序开发人员阅读, 也适合学习计算机软件工程专业, 具有良好Java和XML知识基础的高年级大学生和研究生参考阅读。



Copyright©2002 Wrox Press. All rights reserved. No part of this book may be reproduced, stored in a retrieval system or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical articles or reviews.

本书英文版由Wrox公司出版, Wrox公司已将中文版独家版权授予电子工业出版社及北京美迪亚电子信息有限公司。未经许可, 不得以任何形式和手段复制或抄袭本书内容。

版权贸易合同登记号: 01-2002-1753

图书在版编目(CIP)数据

Java Web服务编程指南/(美)汉德里克(Hendricks, M.)等著; 卢刚等译. —北京: 电子工业出版社, 2002.10

书名原文: Professional Java Web Services

ISBN 7-5053-8016-8

I. J… II. ①汉… ②卢… III. JAVA语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2002)第071846号

责任编辑: 郝黎明

印 刷: 北京天竺颖华印刷厂

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编: 100036

北京市海淀区翠微东里甲2号 邮编: 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 32.125 字数: 820 千字

版 次: 2002年10月第1版 2002年10月第1次印刷

定 价: 52.00元

凡购买电子工业出版社的图书, 如有缺损问题, 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

关于作者

S. Jeelani Basha

Jeelani是美国加利福尼亚州弗里蒙特市Insync Information Systems公司的一位高级软件工程师。他是一位电气工程学士，是Sun认证的Java 2程序员。对于编程的兴趣使他从电气工程转移到软件编程技术方面。他已经有5年以上的编程经验，已经用J2EE技术完成了各种不同的项目。

他目前最感兴趣的课题是Web服务，现在正在制作一个Web服务，作为一种可行的企业应用解决方案。

感谢我的妻子，感谢她对我的爱。

Jeelani欢迎读者与他联系，邮件地址：s_jeelani@yahoo.com

Jeelani编写了本书的第14章。

Scott Cable

Scott 在5年前就获得了微软的认知科学 (cognitive Science) 认证，从那时起就一直从事Internet技术工作，例如，Java和XML。他目前担任一个新闻和金融信息供应商的顾问。他的邮件地址是：webservices@scottcable.com。

Scott编写了本书的第6章。

Ben Galbraith

Ben Galbraith开始接触编程时是用BASIC语言编写Atari 800程序，当时他只有16岁。后来，他不断地努力，使用的编程语言扩展到PASCAL、C、Perl和Delphi，当然还有Visual Basic。

他一直对Java Applet没有深刻的印象，因此，几乎总是避免使用这种语言。直到1999年末，他才突然完全改变了看法。他是一位业余历史学家，部分时间从事商业活动。目前，他住在美国西部，是一位自由作家。

“献给我的兄弟Reed，感谢他为我提供的范例。”

Ben编写了本书的第12章。

Mack Hendricks

Mach Hendricks目前在Sun Microsystems公司iPlanet分公司担任系统工程师。他获得了美国密歇根州罗切斯特市Oakland大学的计算机科学理科学士和理科硕士学位。他有8年以上在多种计算机领域的工作经验，包括系统管理、软件开发和企业技术支持等。他感兴趣的技术有：分布式计算系统、XML、Java、数据挖掘和面向对象的设计和操作系统等。在写作或

探索新技术之余，他总是喜欢与女儿Courtney和女友LaTorri McCray一起度过美好时光。

首先要感谢上帝，终于让我实现了出版一本技术书的梦想。其次要感谢我的父母，McLevyonne和Barbara Hendricks，当我还是个8岁男孩的时候就给我买了一台Commodore VIG 20计算机，看看我现在的情况！此外，还要感谢iPlanet的企业支持客户经理们在本书的写作过程中给我提供的支持和帮助。还要感谢我的好友Eric Hamilton在写书的整个过程中给我的鼓励。最后，更要感谢我的女友，在我写作时对我的耐心和支持，愿本书的出版赢得你的微笑。

读者可以通过邮件地址mlhendri@oakland.edu与作者取得联系。

Mack写了本书的第2章、第3章和第11章。

Romin Irani

Romin Irani是美国加利福尼亚州弗里蒙特市Insync Information Systems有限公司的一位高级软件工程师。他毕业于印度的孟买大学，获得了计算机工程学学士学位。Romin所有的时间都用在研究Web服务技术和软件产品上，在网址www.webservicesarchitect.com能找到他编写的软件。当他没有阅读有关Web服务的资料和编写Web服务书籍的时候，仍然在考虑关于Web服务的问题。他坚信Web服务将获得诺贝尔和平奖（由于它使微软与Java世界结合在一起）和诺贝尔经济奖（由于它明显地降低了综合成本）。欢迎读者和他联系，邮件地址是romin@rocketmail.com。

首先要感谢我妻子Devyani对我的支持和理解。感谢Peter Fletcher和Mark Waterhouse在网址www.webservicesarchitect.com上给我提供了如此多的机会，使我有机会探讨和写作关于Web服务的书。感谢Wrox出色的人们给了我另一次写作的机会。感谢我父母Khushru和Gulrukh对我的教诲。最后，我想把我编写的这几章献给我可爱的妹妹Romana。Romana，这几章是为你而写的！

Romin编写了本书的第1章、第7章和第8章。

James Milbery

James Milbery是Kuromaku Partners LLC的负责人。他管理过不同的客户组，例如Oracle Corporation、William Blair Capital Partners和Allied Capital。他在Compuware、Ingres和Compaq (Digital Equipment Corporation) 公司当过多年的销售顾问和销售咨询经理。他还是《SYS-CON Media's Java Developer's Journal》、《XML Developer's Journal》和《Wireless Business and Technology》等期刊的产品评论编辑。他专为早期和中期的软件公司提供技术支持，他对培养预销售技术队伍非常精通，并善于提供智能竞争的策略。

James编写了本书的第6章和第9章。

Tarak Modi

Tarak Modi开发可升级的高级分布式应用程序已经6年多了。他的专业经验主要是在C++

和Java的核心编程技术方面；处理Microsoft的技术，例如COM、MTS、COM+和做.NET的试验；包括J2EE的Java平台和CORBA。他的专业经验是建立在坚固的教育基础之上的，他获得了电气工程学士学位、计算机工程学硕士学位和一个集中于信息系统的MBA。他在许多著名的软件杂志上发表过文章，包括《Dr. Dobbs Journal》、《Java Report》、《Java Developer's Journal》、《Java World》以及《Java Pro.》等期刊。当不在新建的家里从事新技术研究或工作的时候，Tarak喜欢做一些户外活动，打网球和高尔夫球。Tarak与他的妻子和一个4岁的儿子一起居住在美国佐治亚州的亚特兰大市。

首先要感谢我的母亲，感谢她在养育我的过程中付出的种种艰难和辛苦，她为儿子的前途安排了所有的事情。我也要感谢我的妻子，她对我是如此地理解，不仅允许而且还鼓励我实现自己的梦想。最后，我对儿子深表歉意，当他想跟我玩的时候和当他说：“爸爸在工作”这句话的时候，我就觉得对不起他。

Tarak编写了本书的第13章。

Andre Tost

Andre Tost是美国明尼苏达州罗切斯特IBM公司WebSphere软件组的方案设计师。他目前的工作是帮助IBM的战略软件伙伴与IBM的中间件产品成为一个整体。在此之前，他在IBM的San Francisco和WebSphere商业组件开发机构中担任过各种开发和设计工作。他是1996年初在San Francisco设计项目中开始用Java编程的，此后，就一直用这种语言开发软件。去年以来，他一直紧紧跟随着Web服务技术的进展，把大部分时间都用在该技术编程专题研讨会的咨询和教学工作上。

他出生和成长在德国的北方，1998年，与全家搬到了寒冷的明尼苏达。他是一个足球迷（即美国人叫做英式足球的那种真正的足球），喜欢参加和观看足球比赛。幸好，明尼苏达有个相当不错的英式足球场和有许多踢足球的机会。除了编程和踢足球之外，其他时间他喜欢与妻子和两个儿子一起度过（他们分别2岁和4岁，并且认为作一个IBM的程序员一定是最有趣的事情，可能会……）。

Andre编写了本书的第4和第5章。

Alex Toussaint

Alex Toussaint是美国德克萨斯州奥斯丁Vignette公司的工程主任。他有10年以上的软件开发经验，具有自1996年起广博的Java开发经历和自1998年起使用J2EE技术的经验。Alex给在线杂志（例如Microsoft MSDN）撰写过许多Web应用和电子商务方面的文章。他是Wrox出版公司《Professional Site Server》一书的作者。Alex还被邀请到奥斯丁市德克萨斯大学的Red McCombs商业学校讲授电子商务和企业软件开发的有关课程。Alex现在与他的妻子Danielle和小狗Sasha居住在德克萨斯州的奥斯丁市。他非常欢迎读者给他发电子邮件，地址是：alex_toussaint@yahoo.com。

Alex编写了本书的第6章。

前 言

什么是Web服务

Web服务是一种最新的应用开发成果，已经引起了在所有平台上操作的开发者的兴趣。它基本的概念很简单——Web服务使我们能够对因特网或网络上的一个对象进行远程过程调用（Remote Procedure Calls, RPC）。Web服务并不是第一个允许我们实现RPC的技术，但是，与以前的技术不同的是，Web服务使用中性平台标准（例如HTTP和XML），这使我们可以对客户完全隐藏执行任务的细节，客户只需要知道这个服务的URL和方法调用使用的数据类型，而不需要知道该服务到底是一个在Linux上运行的还是一个在Windows上运行的用Java编程的ASP.NET服务。

本书将介绍多种技术：XML、XSL、XSLT、XML Schemas、SOAP、WSDL和UDDI等。本书将与读者共享我们的经验，介绍我们在开发基于SOAP的分布式应用程序过程中的经验。但是，由于Web服务还是个不够成熟和发展得很快领域，读者很可能感到本书讨论的工具和实用程序没有真正的生产价值。本书尽量不依靠这些工具，以使书中的应用程序范例有任务针对性。一个极好的经验工具的例子就是用IBM Web服务工具套件（IBM Web Service Toolkit, WSTK）帮助开发WSDL文档，采用手工编写WSDL文件的方法可以绕过使用这个工具。

本书着眼于利用Java平台采用不同的方法建立Web服务，并着重介绍开发工具、技术和协议等，这些我们都将会在开发过程中遇到。

本书内容的组织安排

本书分三部分：基础、实用Web服务的开发和Web服务的前景。

第1章介绍Web服务的结构，着重介绍Web服务的访问方法并概括性地描述Web服务系统的组成。

第2章详细介绍SOAP的技术规范，介绍它的来源、它的主要用途以及它现在的作用，接着在第3章将叙述这本书里要用的SOAP的实现（Apache SOAP）。

第3章（Apache SOAP v2.2）将详细地说明下载和配置必要软件的方法，以便启动和运行SOAP服务器。

从此时开始，本书将进入第二部分的内容。Web Services Description Language（Web服务描述语言）和它的工具，包括WSDL文档结构（服务、绑定、端口、操作和类型）将在第4章中介绍。UDDI（Universal Description、Discovery and Integration Registry，通用描述、显示和集成综合注册）以及它在商业应用中的搜索引擎将在第5章中介绍。

第6章将讨论SOAP、Web服务和XML签名的安全保障技术。没有安全性和个性化的保证，就谈不上完成了应用程序的开发。

在第7章，将对一些实例进行研究，这些实例把所有的Web服务技术（SOAP、WSDL和UDDI）都结合到一个集成的系统中。

第8章向读者展示一个现有的J2EE应用程序，怎样用它的功能性展示出一个真正的Web服务。

读者在第9章将看到用Web Service和SAP获得访问SAP功能性的方法。

最后，在本书的第10章（IBM和Web Services）中介绍可以利用的IBM工具，使用这些工具可以在现有的基础设施上实现Web服务。

本书的第三部分主要使读者了解市场上的新技术最近都有哪些动向。第11章（JAXM和JAX-RPC）向读者介绍用于XML消息和远程过程调用的Java API。用JAXM可以传输任何类型的XML文档，JAX-RPC则可用来定义进行RPC方法调用的机制和对使用的基于XML的协议（例如SOAP1.1）。

在第12章讨论JAXR，JAXR是为了通过某种形式的XML消息（比如SOAP）与XML注册通信的Sun的Java高级抽象层。本章将讨论JAXR的结构和公共API。

第13章和第14章介绍Sun ONE和AXIS。Sun ONE是Sun公司的一种技术规范，该规范规定了描述道路图的方法，以便根据开放式标准组建一个服务驱动型网络。AXIS是一个表示第3版Apache SOAP的项目名，AXIS的目的是扩展当前Apache SOAP 2.0版技术规范的功能。

本书适合哪些读者

本书是为有经验的网络开发者而编写的，供那些在J2EE平台上已经有一些开发经验或实践的人阅读。我们还假设本书的读者是比较熟悉Web编程的。

运行本书的范例需要哪些软件支持

为了运行书中的范例，需要有：

- Java 2企业版（J2EE）
- Java 2标准版（J2SE）
- WebLogic Server 6.1
- Tomcat 3.2+
- IBM的Web Services ToolKit 2.4（WSTK）
- CapeClear公司的Cape Connect 3.0和Cape Studio 1.1

书中使用的惯例

为了帮助读者在阅读中获取最多知识，并紧跟作者的思路，全书使用了一些惯例。

例如，文本中字体风格的含义：

- 文中重点的词用粗体表示
- 键盘键入的内容表示如下：Ctrl-A
- 文本中的文件名和代码的表示是：doGet()
- 用户接口和URL的文本用黑体表示：Menu

书中用不同的方式表示代码。方法和属性的定义用黑体代码表示：

```
protected void doGet(HttpServletRequest req, HttpServletResponse resp)
    throws ServletException, IOException
```

范例代码表示如下：

```
In our code examples, the code foreground style shows new, important,
pertinent code
while code background shows code that's less important in the present context,
or has been seen before.
```

在代码范例中，用明显的方式衬托出新的、重要的部分代码。而用普通方式显示对当前上下文相对不重要的或以前出现过的代码。

用户支持

我们一贯重视来自读者的信息，希望了解读者对本书的看法：你喜欢什么和不喜欢什么，下次哪些地方需要我们改进。你可以给我们提出建议，把你的意见写在本书随附的反馈卡寄给我们或通过电子邮件发到feedback@wrox.com均可。请勿忘记在你的信息上注明书名。

怎样下载本书的范例代码

当访问Wrox站点（<http://www.wrox.com/>）的时候，可以简单地通过我们的Search工具或用一个书名列表找出想下载的书名。然后，单击Code栏中的Download命令项或者在本书的详细页面上在Download Code命令上单击。

站点上供下载的文件是用WinZip压缩存档的，因此，当把它们下载到硬盘上的文件夹中之后，还需要用解压缩程序（例如WinZip或PKUnzip）把这些文件提取出来。当提取文件的时候，代码通常放在以章命名的文件夹中。当你启动解压缩程序的时候，确保软件正确地设置了文件夹名。

勘误表

我们用了各种方法来保证书中的文字和代码正确无误。但是，任何事都不可能尽善尽美，错误仍然会存在。如果您在我们的书中发现了错误（比如拼写或代码错误），如果您能把勘误表反馈回来，我们将十分高兴。把勘误表寄给我们不仅使其他读者因避免了运行失败而节约了时间，而且也帮助我们把更高质量的信息提供给广大读者。您可以简单地通过电子邮件把信息发送到support@wrox.com，我们会检查您提供的信息是否正确，如果正确就把它

列在书的勘误表中，并在该书的后续版本中改正。

想在Web站点上查找勘误表的读者，可以访问<http://www.wrox.com>，通过Advanced Search或书名列表进行查找，在Book Errata链接上单击，该链接位于该书细节页面上封面图的下面。

E-mail支持

如果读者想直接询问一个与本书有关的问题，请一位详细了解本书内容的专家回答，可以发电子邮件，地址是support@wrox.com。需要在电子邮件的主题字段写上书名和ISBN（国际标准图书编号）的最后4位数字（本书最后4位是3757）。典型的电子邮件应当包括以下内容：

- 在主题字段键入书名、ISBN的最后4位数字和问题的页码。
- 您的姓名、联系方式和在消息的主要部位写上您的问题。

我们不会给您发送垃圾邮件。我们需要详细资料以节省您的和我们的时间。当您发给我们一份电子邮件的时候，邮件会通过以下支持环节：

- 用户支持——读者的消息首先递交给我们的用户支持工作人员阅读。他们有一些常见问题的文件，任何关于本书或Web站点的常见问题都能立即作出回答。
- 编辑——比较深入的问题交给本书的技术编辑去处理。他们有使用程序语言或特定软件的经验，能够更详细地回答有关的技术问题。
- 作者——最后，在极少数的情况下，如果编辑都回答不了读者的问题，他就会把问题送给作者。虽然我们总是力图保护作者，尽量不打扰他们的写作，但我们也十分高兴地把一些特殊的问题提供给他们。所有Wrox的作者都会提供有关他们书的帮助。他们会发送电子邮件，回答读者或编辑的问题，而所有的读者也将从中受益。

Wrox Support只提供与我们出版的图书直接有关问题的支持。对普通的图书支持以外的问题，我们通过<http://p2p.wrox.com>论坛的团体列表来提供支持。

p2p.wrox.com

为了使作者和同等水平人的讨论加入到P2P邮件列表中来，除了我们独特的一对一的邮件系统之外，还在邮件列表、论坛和新闻组上提供了程序员对程序员（programmer to programmer）的接触。如果读者向P2P提出一个问题，可以确信这个问题会被我们邮件列表上的许多Wrox的作者和其他业界专家们看到。在p2p.wrox.com网址上，许多不同的列表都可以给读者提供帮助，不仅在读者阅读本书的时候，而且在读者开发自己应用程序的时候。对本书有用的列表是xml_web_services、java_webservices和java_xml。

如果要订阅邮件列表，可按下列步骤进行：

1. 进入<http://p2p.wrox.com/>。
2. 从左边菜单栏中选择相应的类别。
3. 单击您想加入的邮件列表。
4. 按照指令订阅并填上您的E-mail地址和密码。
5. 答复您收到的确认电子邮件。

6. 利用订阅管理程序加入更多的列表，并设置你的邮件优先权。

为何该系统提供了最好的支持

读者可以选择参加邮件列表，或者接收邮件列表每周一次的分类摘要。如果读者没有时间（或者工具）接收邮件列表，也可以搜索我们的在线档案。垃圾或广告邮件会被删除，而您自己的电子邮件地址会被独特的Lyris系统保护起来。若要查询关于加入或退出列表的问题，以及其他有关列表的普通问题，可把电子邮件发到listsupport@p2p.wrox.com。

译 者 序

Web服务是程序设计领域中的一项新技术。它的优点是，在完全不同的平台之间具有互操作性，通过遍布全世界的因特网，实现不同应用程序之间的远程过程调用。它是一种不管应用程序使用哪种操作系统，不管用什么程序语言或对象模型，都能在分布式程序之间交换信息的技术。**Web服务**通过发布、发现和绑定的标准模式，使企业应用程序集成（EAI）变得更加容易实现。本书向读者介绍了当前**Web服务**的核心技术，包括**SOAP**、**WSDL**和**UDDI**。

本书由多位作者合作编写，是他们集体智慧的结晶。书中介绍了**Web服务**背后的重要技术和有关技术规范，使读者对**Web服务**的结构、访问方法和系统组成形成清晰的概念。通过对几个实例的研究，把多种**Web服务**技术都应用在一个集成的系统中，不仅进行代码分析，而且还介绍不同技术之间的差异和应用场合。书中提供了作者们在开发分布式应用程序过程中的实际经验，还包括许多实现**Web服务**的最新信息。

本书的前言和第1章~第4章由李双庆翻译；第5章、第6章由朱军翻译；第7章~第9章由高春蓉翻译；第10章~第14章和附录部分由卢刚翻译。全书由李双庆审校。参加部分翻译和校对工作的还有彭苏鲁、金克亮、李苏苏、刘求真、赖庆来、关达、申晓陵、李春子、姚远、甘壁文等。在此，对他们的努力工作和热情帮助表示诚挚的感谢。由于技术新，若本书在翻译上有错误和不当之处，敬请读者指正。

目 录

第1章 Web服务的结构	1
Web服务的模式	1
基本Web服务栈	2
J2EE中的基本Web服务结构	9
详细结构	11
小结	23
第2章 SOAP	24
SOAP简介	24
SOAP什么样	29
安全性问题	31
Java中的SOAP结构	32
协议	34
HTTP中的SOAP	47
小结	50
第3章 Apache SOAP 2.2	51
Apache SOAP的历史	51
Apache SOAP的前景	51
安装Apache SOAP	53
展开和运行“Hello World”服务	55
工作履历档案库服务	64
运行服务	69
开发SOAP服务	70
开发SOAP客户机	79
调试SOAP客户机和服务	92
小结	93
第4章 WSDL	95
WSDL文档结构	95
WSDL4J API	110
从WSDL调用动态服务	123
小结	131

第5章 UDDI	132
引言	132
UDDI注册表	134
UDDI数据结构	138
UDDI API	150
UDDI for Java (UDDI4J)	152
例子: 创建<tModel>和<businessService>元素	165
小结	171
 第6章 Java Web服务的安全性	172
安全服务	172
安全技术和方法	175
Web应用安全	180
Java Security API、工具套件和SDK	196
小结	198
资源	199
 第7章 健康管理系统案例分析	201
Wrox Insurance	201
软件需求和设置	205
Wrox Insurance Web Services Approach	209
提供者Web服务	213
Wrox Insurance - ProviderProxy	221
Wrox Insurance Web站点	231
小结	247
 第8章 使用Web服务的J2EE应用	248
J2EE应用——WroxCuisine	248
启动Web服务	281
CapeConnect Web服务平台	282
Web服务客户机	288
小结	309
 第9章 SAP和Web服务	310
mySAP和R/3	310
SAP的可编程界面	311
SAP因特网商业框架	314
SAP的中间件结构	318

通过bTalk使用Java语言与R/3连接	324
SAP未来技术展望	333
小结	335
第10章 IBM和Web服务	337
IBM服务器技术平台	337
使用IBM的UDDI注册表	342
实现一个简单的Web服务	345
使用IBM的Web Services ToolKit	352
用IBM UDDI4J进行展开和发布	364
小结	367
第11章 JAXM和JAX-RPC	368
JAXM与JAX-RPC对比	368
JAXM	369
JAX-RPC	385
小结	395
第12章 JAXR	397
XML注册表	397
JAXR体系结构	398
JAXR API: 怎样使用	399
小结	413
第13章 Sun ONE介绍	414
ONE体系结构	415
示例	425
Microsoft.NET	441
小结	445
第14章 AXIS介绍	446
AXIS的关键功能	446
体系结构	447
Web服务展开描述符 (WSDD)	450
组件的展开	452
Alpha 3中的特性	454
开发一个AXIS服务示例	458
下一版中预计的功能	479

小结	480
附录A 安装CapeConnect Three	482
附录B 安装CapeStudio 1.1	494

第1章 Web服务的结构

本章将介绍Web服务的基本概念，它们有什么好处和怎样构成一个Web服务系统。在这里，结构指的就是可以帮助我们组成一个Web服务系统的不同组件或构建模块。更具体地说，本章将：

- 使读者了解Web服务的模式，并能在一个Web服务环境中识别出Web服务不同角色和操作。
- 介绍基本的Web服务栈和SOAP、WSDL和UDDI。
- 学习在J2EE中基本的Web服务结构。
- 详细探讨Web服务的结构。
- 针对现实中的电子商务问题（例如交易、服务的质量、Web服务的管理和安全性等），讨论对一个Web服务系统还有哪些其他要求。
- 讨论Web服务的性能问题。

Web服务的模式

Web服务给人们带来许多令人吃惊的好处。已经听到的有：它具有在完全不同平台之间的互操作性、在无所不在的网络技术上调用一种Web服务的能力等。但是，它到底是怎样工作的呢？只要观看一个概念上的Web服务模式就能找到最好的答案。这个模式突出强调了任何系统都有的两个重要的方面，即角色和工作方式。在这里，角色指的是不同类型的实体，而操作指的是为了使Web服务工作，这些实体所完成的功能。图1.1是一个Web服务模式的示意图，这个图可以帮助读者理解Web服务的模式。

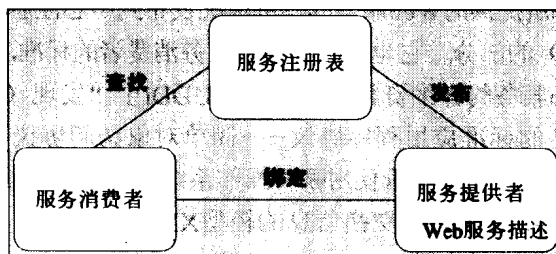


图1.1 Web服务的模式

在这个典型的Web服务模式图上可以看出，在典型的Web服务环境中有三种角色，并可以看到为了使Web服务起作用，这些角色所执行的操作。图中的角色如下：

- **服务提供者** 服务提供者是创建该Web服务的实体。典型情况下，服务提供者把他们机构的某种业务功能展示成一种Web服务，供任何其他机构调用。以一个想把在线订购图书的服务作为Web服务的网上图书销售商为例。该服务提供者需要做两件事来完