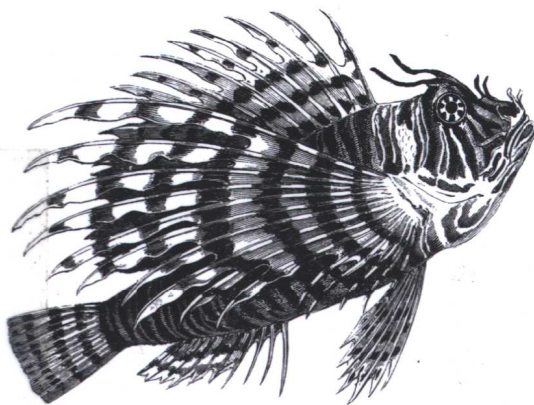




Java and SOAP



Java[™] 与 SOAP



3

O'REILLY[®]

中国电力出版社

Robert Englander 著

黄刚 译

Java™ 与 SOAP

Robert Englander 著

黄刚 译

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Paris • Sebastopol • Taipei • Tokyo

O'Reilly & Associates, Inc. 授权中国电力出版社出版

中国电力出版社

图书在版编目 (CIP) 数据

Java™ 与 SOAP/ (美) 英格兰德 (Englander, R.) 著; 黄刚译. - 北京: 中国电力出版社, 2002

书名原文: Java™ and SOAP

ISBN 7-5083-1312-7

I. J... II. ①英... ②黄... III. ① JAVA 语言—程序设计 ②通信协议, SOAP

IV. ① TP312 ② TN915.04

中国版本图书馆 CIP 数据核字 (2002) 第 089015 号

北京市版权局著作权合同登记

图字: 01-2002-3272 号

©2002 by O'Reilly & Associates, Inc.

Simplified Chinese Edition, jointly published by O'Reilly & Associates, Inc. and China Electric Power Press, 2002. Authorized translation of the English edition, 2002 O'Reilly & Associates, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly & Associates, Inc. 出版 2002。

简体中文版由中国电力出版社出版 2002。英文原版的翻译得到 O'Reilly & Associates, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly & Associates, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

书 名 / Java™ 与 SOAP

书 号 / ISBN 7-5083-1312-7

责任编辑 / 宋宏

封面设计 / Emma Colby、张健

出版发行 / 中国电力出版社 (www.infopower.com.cn)

地 址 / 北京三里河路 6 号 (邮政编码 100044)

经 销 / 全国新华书店

印 刷 / 北京市地矿印刷厂

开 本 / 787 毫米 × 1092 毫米 16 开本 19.5 印张 279 千字

版 次 / 2002 年 12 月第一版 2002 年 12 月第一次印刷

印 数 / 0001-5000 册

定 价 / 39.00 元 (册)

Java[™] 与 SOAP

O'Reilly & Associates 公司介绍

为了满足读者对网络和软件技术知识的迫切需求,世界著名计算机图书出版机构 O'Reilly & Associates 公司授权中国电力出版社,翻译出版一批该公司久负盛名的英文经典技术专著。

O'Reilly & Associates 公司是世界上在 UNIX、X、Internet 和其他开放系统图书领域具有领导地位的出版公司,同时是联机出版的先锋。

从最畅销的《The Whole Internet User's Guide & Catalog》(被纽约公共图书馆评为二十世纪最重要的 50 本书之一)到 GNN (最早的 Internet 门户和商业网站),再到 WebSite (第一个桌面 PC 的 Web 服务器软件), O'Reilly & Associates 一直处于 Internet 发展的最前沿。

许多书店的反馈表明, O'Reilly & Associates 是最稳定的计算机图书出版商——每一本书都一版再版。与大多数计算机图书出版商相比, O'Reilly & Associates 公司具有深厚的计算机专业背景,这使得 O'Reilly & Associates 形成了一个非常不同于其他出版商的出版方针。O'Reilly & Associates 所有的编辑人员以前都是程序员,或者是顶尖级的技术专家。O'Reilly & Associates 还有许多固定的作者群体——他们本身是相关领域的技术专家、咨询专家,而现在编写著作, O'Reilly & Associates 依靠他们及时地推出图书。因为 O'Reilly & Associates 紧密地与计算机业界联系着,所以 O'Reilly & Associates 知道市场上真正需要什么图书。

作者简介

Robert Englander 是 MindStream 软件公司 (www.mindstrm.com) 的总裁兼首席工程师。他的工作是为软件体系结构、设计和开发提供咨询,同时也为客户的项目开发基本框架。他致力于组件体系结构和分布式系统领域的工作。

Rob 已使用 Java 和 C++ 为从小商店到大集团公司之类的客户编写了很多软件。他还在业界会议上做过多次演讲,也为杂志和期刊撰写文章。他还是 O'Reilly 出版公司的《Developing Java Beans》一书的作者。

封面介绍

本书封面上的动物是一种叫做 *Pterois volitans* 的红色火鱼。这些鱼遍布于印度洋和太平洋中 100 英尺以下的温暖热带水域。它们喜欢一动不动地呆在海底,通常是在暗礁下,等待猎物(主要是一些小鱼和甲壳类动物)的到来。

红火鱼是鲷科鱼家族的一员——鲷是它们背鳍上的毒刺的名字。由于对保护自己的 ability 充满自信,火鱼即使在遇到人类时也常常不会回避;相反,它们会将毒刺指向入侵者。它们的自信是有道理的:被它们的毒刺扎一下,对人类来说尽管一般不会致命,但却极其疼痛。

除它们所习惯的海洋环境之外,也常常可以在家庭的热带鱼水族箱中看到红火鱼。它们可以长到 16 英寸之长,因此往往需要一个相当大的水族箱。不过尽管它们拥有可怕的刺和食肉的天性,它们通常还是非常友善的,而且与其他鱼类相处得很好——除非那些鱼小到可以被吃掉。

目录

前言	1
第一章 简介	7
RPC 和面向消息的分布式系统	8
自我描述数据	9
XML	10
API 规范与传输级规范	11
SOAP 概述	12
SOAP 实现	13
本书的编写方式	15
开始吧	15
第二章 SOAP 消息	17
HTTP 绑定	17
HTTP 请求	18
HTTP 响应	20
SOAP 封套	22
Envelope 元素	26

Header 元素	26
actor 属性	27
mustUnderstand 属性	28
encodingStyle 属性	30
封套版本	31
Body 元素	31
SOAP Fault	32
第三章 SOAP 数据的编码	36
模式与名称空间	36
串行化规则	39
指定数据类型	43
默认值	56
SOAP 的 root 属性	57
第四章 RPC 形式的服务	59
SOAP RPC 元素	59
一个简单的服务	66
部署服务	68
编写服务客户	79
使用请求级作用域进行部署	89
使用会话级作用域进行部署	91
传递参数	94
第五章 处理复杂数据类型	106
传递数组参数	106
返回数组	116
传递自定义类型参数	121
返回自定义类型	134

第六章 自定义串行化	142
自定义类型的编码	143
第七章 错误和异常	172
在 Apache SOAP 中抛出一个服务器端异常	172
在 Apache SOAP 中创建一个错误监听器	176
在 GLUE 中抛出和捕获异常	182
第八章 可供选择的技术	187
SOAP 消息传递	188
直接编码	200
第九章 SOAP 的互操作性与 WSDL	217
Web 服务描述语言	218
从 Apache SOAP 客户调用 GLUE 服务	229
一个使用 Apache SOAP 的代理服务	236
从 GLUE 客户调用 Apache SOAP 服务	241
访问 .NET 服务	249
编写 Apache Axis 客户	255
第十章 SOAP Header	259
Apache SOAP 的提供者和路由器	260
替换提供者和路由器类	261
一个能处理 SOAP 头的 Apache SOAP 服务	266
第十一章 JAX-RPC 与 JAXM	274
JAX-RPC	275
不使用 Ant 来完成工作	276

创建一个 JAX-RPC 服务	278
创建一个 JAX-RPC 客户应用程序	285
根据 WSDL 生成存根	287
动态调用接口	290
用一句话介绍 JAXM	290
接下来会是什么?	291
~	
词汇表	293

前言

SOAP (Simple Object Access Protocol, 简单对象访问协议), 是当前众多分布式计算技术中最新的一个。与其他分布式计算技术不同的是, 它是基于XML的, 这样就不需要花很多时间来重新定义计算世界。相反, SOAP 规范描述了数据的内容和结构等重要方面, 以及它们与RPC (Remote Procedure Call, 远程过程调用) 和消息传递系统 (message passing system) 这些众所周知的编程模型之间的关系。

这些规范存在于XML世界里。SOAP与特定的编程语言、计算平台或软件开发环境无关。有很多SOAP实现可以提供到各种编程语言 (如C#、Perl和Java) 的绑定。离开了这些实现, SOAP仍然是抽象的: 它是一个没有被具体化的重要概念。然而, 只有与特定的软件开发语言相结合, SOAP才能得以生存, 本书也体现了这一点。Java天生就适合处理XML, 并使它能够更好地建立SOAP服务和客户应用程序。如果你希望使用Java来编写SOAP软件, 那么本书正好可以作为你的入门书籍。

本书适合的读者

本书是为那些想知道如何利用Java访问基于SOAP的Web服务以及如何使用Java

创建基于 SOAP 的 Web 服务的人编写的。本书的对象是已经熟悉 Java 的程序设计人员、学生和专业人士，因此本书不打算在 Java 语言的基本观念或语法上花费时间。如果你对 Java 还不是很熟悉，可以去买一本关于 Java 语言的书来看，如：O'Reilly 出版公司的《Learning Java》或《Java in a Nutshell》（译注 1）。

未来的演变

SOAP 规范目前仍在演变之中。这本书是依照 SOAP 规范的 1.1 版编写的。尽管其中一些现在提到的概念和技术在未来的 SOAP 版本中仍将沿用，但很明显，新版本发布时肯定会带来一些很重要的新东西。同时，我们使用的 Java 实现也在演变。显然，这本书中的描述和例子将会随时间的推移而变得过时甚至荒废（或迟或早），这取决于 Web 服务技术发展的速度。事实上，预兆已经出现了：本书中的许多例子都是基于 Apache SOAP 第 2 版编写的，但它注定要被 Apache SOAP 第 3 版（也称为 Axis）所取代。我们将在第九章中简要介绍现在可以得到的它的一个早期版本。尽管 JAX RPC 和 JAXM API 规范本身还在发展之中，但 Axis 还是被设计成支持这些规范。Sun 公司已针对这些规范提供了一个测试版本的参考实现（将在第十一章中讨论）；而这个版本的出现时间比 Axis 的最新版本更早。同样，如果认为 JAX Pack 规范的定稿将会终止这一系列演变过程，那也是很愚蠢的。不过，不管怎样，当这一切不可避免地到来的时候，但愿你的知识能够使你足以跟得上这种变化的步伐。

本书的组织

本书中各章的内容循序渐进，因此最好的方案是按章节顺序依次阅读。

第一章，简介

这一章对 SOAP 进行了简要介绍，包括相关的技术、存在的问题以及与其他解决方案的对比等。本章还介绍了 Apache SOAP 和 GLUE，本书会一直使用这两种 SOAP 实现。

译注 1：本书中文版《Java 技术手册》已由中国电力出版社引进出版。

第二章、SOAP 消息

这一章介绍了 SOAP 封套，它是一个携带了客户和服务端之间 SOAP 事务的有效负载的结构化 XML 文档。本章全面讲述了 SOAP 封套的所有方面，包括 Header、SOAP 的 Body 元素和 Fault。还介绍了 SOAP HTTP 绑定的一些细节。

第三章、SOAP 数据的编码

这一章全面介绍了 SOAP 事务的数据编码，包括对数据元素进行编码和串行化的规则。它首先介绍了名称空间，然后深入探讨了简单和复杂数据类型的串行化。

第四章、RPC 形式的服务

这一章深入讨论了基于 SOAP 的 RPC 形式的服务。给出了与服务方法和参数有关的广泛内容，包括服务的部署和活动机制的有关细节。

第五章、处理复杂数据类型

这一章介绍了如何建立包含使用复杂参数和返回值（比如数组和 Java bean）方法的服务。它全面讲述了在客户系统和服务器系统上将这些类型映射到 Java 类的机制。

第六章、自定义串行化

作为对第五章的补充，这一章全面介绍了非标准自定义数据类型的使用。讲述了一些用来传递自定义数据类型参数与返回值的工具和 API。本章还详细介绍了编写串行化和反串行化自定义类型 Java 类的方法。

第七章、错误和异常

这一章讲述了 SOAP Fault（错误），以及它与 Java 异常的关系。它介绍了默认的机制以及生成和扩展错误内容的技术。

第八章、可供选择的技术

这一章首先介绍了 SOAP 的消息形式的服务，这是一项 RPC 模型的替代技术。它还讲述了在 SOAP 封套中传递直接 XML 的技术，最后介绍了 SOAP 附件。

第九章，SOAP的互操作性与WSDL

这一章介绍了如何让使用不同技术开发的SOAP客户和服务端能够正确地共同工作。它对WSDL（Web Services Description Language，Web服务描述语言）做了简要介绍。本章还介绍了分别使用Apache SOAP和GLUE建立客户和服务器的例子，以及可以访问微软.NET服务的Java客户的例子，并对Apache Axis做了简要介绍。

第十章，SOAP Header

这一章讨论了SOAP Header的使用，它为在存在于SOAP Body之外的客户与服务之间传递数据提供了一种手段。本章讲述了一个用作其他服务的消息路由器的中间服务的开发过程。为了处理SOAP Header，我们还开发了一些用于扩展Apache SOAP框架的Java类。

第十一章，JAX-RPC与JAXM

这一章讨论了一个正在形成的标准：基于XML的RPC的Java API（JAX-RPC）。它对Sun公司一个早期版本的参考实现做了简要介绍。本章包括了一个服务和一个客户的开发，还讲述了如何使用这些工具来开发代码，以访问由WSDL描述的服务。最后还介绍了JAXM。

排版约定

本书使用下列英文字体约定：

等宽字体（constant width）用于：

- Java程序中的所有代码，包括关键字、操作符、数据类型、常量、方法名称、变量名称、类名、接口名称及Java包名。
- 需要在屏幕上逐字输入的命令行和选项。
- 名称空间。

斜体（*italic*）用于：

- 路径名、文件名和Internet地址，例如域名和URL。斜体还用于表示可执行文件。

将一本书中所有的东西都精细地区分开来，这项工作就像一场痛苦的战争。但是我仍尝试着区别名称空间（使用等宽字体）和 URL（使用斜体），尽管两者看起来没什么不同。同样，我还将 Java 方法名（使用等宽字体，以一对圆括号作为结尾）和从 SOAP 服务中导出的方法（使用等宽字体，但没有圆括号）区别开来。

建议与评论

请向出版商提出有关本书的一些评论和问题：

美国：

O'Reilly & Associates, Inc.
101 Morris Street
Sebastopol, CA 95472

中国：

100080 北京市海淀区知春路 49 号希格玛公寓 B 座 809 室
奥莱理软件（北京）有限公司

本书拥有相关的网页，其中列出了勘误表、示例或者任何附加信息。你可以通过这个地址访问此页面：

<http://www.oreilly.com/catalog/javaxmlatabind>

要发表评论或者询问有关本书的技术问题，请向这个地址发送电子邮件：

bookquestions@oreilly.com
info@mail.oreilly.com.cn

有关书籍、会议、资源中心以及 O'Reilly Network 的更多信息，请参见 O'Reilly 公司的网站：

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

致谢

我的好朋友 Rinaldo DiGiorgio 直到今天还在鼓励我，让我对 Java 及其相关技术感兴趣。我认为再没有其他任何人能在 Java 工作上对我产生如此大的影响了。谢谢你，Rinaldo，谢谢你使我走上了一条正确的路。

非常感谢 David Askey 和 Anne Thomas Manes，他们审阅了此书并提供了许多有价值的意见。他们设法找出错误并且提供建议，没有他们的帮助本书不可能变得这么好。感谢 Lorraine Pecorelli 通读各章并使本书减少了词不达意的问题。还要对本书的编辑 Mike Loukides 表示最诚挚的谢意。在完成本书的过程中曾出现过很多困难，是 Mike 勇于承担义务和忠于事业的精神才使我们的努力变成这本书。同时也要将这份功劳归功于 O'Reilly 的设计和生产品人员。

最后，我要谢谢我的家人，Jessica 和 Carolyn，感谢他们的支持。这次我将不感谢我的朋友——他们全然没有帮忙！

第一章

简介

在软件发展的历史过程中，新方法的出现时常将已被丢弃的思想又重新使用到主流应用中。每当一个思想被重用时，它以往成功和失败的经验就成为无价之宝，能够帮助改进该思想本身并完善其实现，至少能使它更具实用性。我并不是说我们要穿新鞋走老路，相反，我们要做的是温故而知新。这样做往往会成为新思想和新技术的催化剂。

我们看到的主机和终端构成的集中计算模式，现在摇身一变，以应用服务器和瘦客户机的形式又回来了。我们曾经见过的 P-Code 概念现在又以 Java 和 Visual Basic 这样的解释语言的形式再次出现。软件开发领域的此消彼长就像现实宇宙的变化。经过足够久的时间之后，也许未来有一天你会发现又能用到你今天正在做的工作。

我要表达的观点是（事实上也确实如此）：好思想之所以是好思想，并不在于它是不是新出现的。重要的是要合乎时宜。正因为如此，分布式计算现在才大行其道。分布式计算的观念并不是才出现的，但是却常常被翻出来用。目前普遍使用的基础设施和技术（像 Internet、Web 浏览器和它们的相关协议）使我们又重新回到了分布式计算，并发展了它。这场变革的最新热点正是 Web 服务。

Web 服务（Web service）主要是一种服务器功能，它通过发布接口的机制来提供可访问的功能。有很多种技术能够实现这样的功能，它们有一个非常重要的共