

BEA 技术丛书

BEA WebLogic Workshop:

Building Next Generation Web services Visually

BEA WebLogic Workshop

—— 构建下一代 Web services

[美] Sean Christofferson Srinivas Jayanthi 著
Steven Traut Wira Pradjinata

唐明 宋杨 黄川徽 等译
BEA 系统 (中国) 有限公司 审校



电子工业出版社

Publishing House of Electronics Industry
<http://www.phei.com.cn>

BEA 技术丛书

BEA WebLogic Workshop

——构建下一代 Web services

BEA WebLogic Workshop: Building Next
Generation Web services Visually

Sean Christofferson

Srinivas Jayanthi

[美]

Steven Traut

著

Wira Pradjinata

唐 明 宋 杨 黄川徽 等译

BEA 系统 (中国) 有限公司 审校

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书介绍如何使用BEA WebLogic Workshop开发满足企业应用需求的Web services。本书通过四个主要部分,系统地讲述了BEA WebLogic Workshop 开发工具、Web services、JSP、XML、JavaScript 和Java 等技术和相关知识,结合一个贯穿全书的在线商店的例子,一步步地指导读者使用 WebLogic Workshop来开发Web services服务器端和客户端,帮助读者了解如何构建基于Web services的企业应用。

本书适用于Java和Web services的初学者使用,对于在这方面具有一定经验的资深开发人员,本书提供了更加简便快捷的开发工具和开发方法,因此,也具有相当的借鉴意义。

Sean Christofferson, Srinivas Jayanthi, Steven Traut, Wira Pradjinata: **BEA WebLogic Workshop: Building Next Generation Web services Visually**

ISBN 0-7645-1797-X

Copyright © 2002, John Wiley & Sons, Inc. All Rights Reserved. Authorized translation from the English language edition published by John Wiley & Sons, Inc.

No part of this book may be reproduced in any form without the written permission of John Wiley & Sons, Inc. Simplified Chinese translation edition Copyright © 2003 by John Wiley & Sons, Inc. and Publishing House of Electronics Industry.

本书中文简体字翻译版由 John Wiley & Sons 授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

版权贸易合同登记号: 图字: 01-2002-5602

图书在版编目(CIP)数据

BEA WebLogic Workshop: 构建下一代Web services / (美) 克里斯托弗森(Christofferson, S.)等著; 唐明等译. —北京: 电子工业出版社, 2003.8

(BEA技术丛书)

书名原文: BEA WebLogic Workshop: Building Next Generation Web services Visually

ISBN 7-5053-9088-0

I. B... II. ①克... ②唐... III. ①应用程序-服务器 ②JAVA语言-程序设计 IV. ①TP368.5 ②TP312
中国版本图书馆CIP数据核字(2003)第075083号

责任编辑: 窦 昊

印 刷 者: 北京兴华印刷厂

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

经 销: 各地新华书店

开 本: 787 × 980 1/16 印张: 20.5 字数: 425千字

版 次: 2003年8月第1版 2003年8月第1次印刷

定 价: 35.00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。质量投诉请发邮件至 zls@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

译者序

随着计算机技术的不断发展,今天,越来越多的企业摒弃了胖客户端的传统IT架构,更倾向于使用松耦合的架构构建企业级应用,尤其是在企业内部各个系统之间,或者在企业内部系统和外部其他系统之间的数据交互设计中,企业应用更要划分成为多个模块和组件,以面向服务的模式提供给内部其他系统或者业务伙伴的系统调用。

Web service 作为一种基于 Web 的分布式的应用,可以被多种客户端程序使用和共享。Web service 可以分布在不同操作系统甚至不同技术实现的服务器上,客户端可以通过多种不同的技术进行访问,因为 Web service 是使用标准的协议进行封装和传输的,例如通过标准的Internet协议HTTP来传输XML格式的消息,进行Web Service 调用,传递参数和取得结果。正是因为如此,Web service 逐渐被更多的企业所采用,构建他们的企业应用和企业集成。

BEA 公司作为 Web service 的倡导者和实践者之一,它的产品 WebLogic Platform将Web service作为“核心服务”,对Web service提供了丰富的支持。而WebLogic Workshop 更是为开发者提供了强大的 Web service 的开发和部署工具,通过WebLogic Workshop,可以快速、简单地开发出符合客户需要的 Web service,并且提供了对于 Web service 会话、异步调用和回调等先进技术的支持。

本书介绍了如何使用 WebLogic Workshop 开发满足企业应用需求的 Web service,也是一本反映了当前Web service的最新技术水平的使用参考书,可以满足广大开发人员利用市场领先的BEA WebLogic Platform,借助 Web service,构建基于 Web 的企业应用和企业集成的需要。

本书涵盖了WebLogic Workshop的开发环境和 Web service 知识,并且通过一个在线商店的例子,循序渐进地解释了如何利用WebLogic Workshop进行Web service 的开发和开发 Web service 的客户端,同时,也介绍了一些和开发 Web service 有关的知识,例如JSP、JavaScript、XML、Java等。读者可以以在线商店的例子为基础,经过适当的剪裁和补充,开发出自己的应用。

本书由唐明、宋杨、黄川徽、刘小江、刘国华、陈建冰、张龙等翻译。限于译者的水平和能力,错误和不当之处在所难免,恳请广大读者给予批评和指正。

前 言

本书旨在介绍 BEA 公司的 Web service 开发工具——WebLogic Workshop。我有幸成为 WebLogic Workshop 最初版本的开发团队中的一员，投身于产品整个开发周期，并且有机会提出建议，使之成为最终产品。当我构思这本关于 WebLogic Workshop 的书时，我兴奋地向世人介绍我为之付出许多时间和精力产品。

我写本书的目的不是简单地罗列 WebLogic Workshop 的每一个功能，或者对开发 Web service 高级应用的程序员提出忠告。我希望那些 Web service 的初学者能够使用最短的时间快速掌握 WebLogic Workshop，使用它开发 Web service 应用。在本书的最后几章中，我着重介绍了 WebLogic Workshop 的一些复杂特性，但总体来说，本书依然遵循使初学者快速上手以开发 Web service 应用的宗旨。

虽然本书针对于那些没有太多 Java 和 Web service 开发经验的初学者，但如果读者是经验丰富的 Java 和 Web service 开发者，会发现 WebLogic Workshop 和这本书还有更多的用途。这本书是给程序员阅读的，但是我并没有对程序员的背景和经历做任何假设。对于 VB 脚本编写者和熟练的 Java 开发者，这本书有同样的益处。

而且，关于读者对 Web service 的了解，我没有做太多的假设。本书从实际应用的角度讲授 Web service。我没有花费大量的时间以学术方式来讨论什么是 Web service，而是使用大量的例子说明 WebLogic Workshop Web service 能够做什么，使读者凭借直觉理解 Web service，并且懂得如何使用 Web service 处理现实中的问题。

本书的组织方式

本书由四部分组成。第一部分简要介绍 Web service 的领域，讨论 WebLogic Workshop 的框架。接着，开始介绍创建一个基本的 Web service 所需的步骤。由于章节的需要，我们忽略了很多细节，但是在第 1 章最后，展示了一个功能完整的 Web service 应用。最后，更详细地介绍了 WebLogic Workshop IDE，详细介绍了它的不同部分。

第 2 章中介绍 WebLogic Workshop 的核心特性，首先介绍会话，并且基于会话展开详细的讨论，因为会话对于大多数人是陌生的。然后讨论 WebLogic Workshop 更强大的特性——异步调用。最后，介绍 WebLogic Workshop 的基础控件，使用基础控件能够访问其他 Web service 服务。

本书的第三部分介绍一些高级特性。本部分中介绍了更多的高级控件，通过这些高级控件，Web service可以访问一些大多数程序员暂时用不到的服务。然后讨论WebLogic Workshop所使用的协议以及如何使用这些协议调用控件。

最后一部分介绍如何处理JavaScript映射。JavaScript映射非常有用，最终每个人都想使用JavaScript映射。

Java入门基础作为附录附加在本书的最后。如果读者对Java知识了解不多，或者近来已很少使用Java语言，使用Workshop感到有些生疏，则Java附录是很好的手册，可帮助读者回顾一些知识。

致谢

首先，感谢我的三位合作者：Srinivas Jayanthi、Steven Traut和Wira Pradjinata。没有他们的帮助，我不知道我能否完成这个浩大的工程。

我非常感激能有这个机会，参与像WebLogic Workshop这么有意义的项目。为此我要感谢Crossgain团队的每一个人；特别要感谢那些发现我并将我吸收到这个项目的同事，包括Adam Bosworth、Rod Chavez、Peter Horadan和Stephen McMenamin。同样，我不会忘记那些和我亲密合作的朋友：Kevin Zatloukal、Mark Rafn、Roger Weber和Terry Lucas。

另外，还要感谢Wiley和Hungry Minds出版公司的Jim Minatel、Chandani Thapa和Grace Buechlein，是他们最初与我联系并启动这个项目的。

Sean Christofferson

目 录

第一部分 开始

第 1 章 WebLogic Workshop Web service 简介	2
1.1 Web service 简介	2
1.2 WebLogic Workshop 简介	6
1.3 使用 WebLogic Workshop 不需要知道的事项	7
1.4 使用 WebLogic Workshop 需要知道的事项	7
1.5 小结	8
第 2 章 开始使用 WebLogic Workshop	10
2.1 WebLogic Workshop 简介	10
2.2 运行第一个 Web service	11
2.3 编写第一个 Web service	13
2.4 从头开始开发第一个 Web service	15
2.5 JWS 简介	26
2.6 小结	33
第 3 章 WebLogic Workshop IDE 入门	34
3.1 IDE 指南	34
3.2 工程树	39
3.3 源代码视图	45
3.4 结构面板	51
3.5 导航条	53
3.6 测试视图	54
3.7 调试器	61
3.8 小结	66

第二部分 基本概念

第 4 章 创建会话式 Web service	68
4.1 会话的三个阶段	69

4.2	在设计视图中操作会话	70
4.3	源代码中的属性	73
4.4	会话式 Web service 示例	75
4.5	测试会话式 Web service	81
4.6	控制更多会话的特性	83
4.7	小结	86
第 5 章	创建异步 Web service	87
5.1	为什么需要异步模式	87
5.2	WebLogic Workshop 异步机制	88
5.3	缓冲	88
5.4	回调	92
5.5	小结	101
第 6 章	XML 映射	102
6.1	XML 消息	102
6.2	基本的映射	103
6.3	映射的其他方面	112
6.4	小结	118

第三部分 高级概念

第 7 章	核心控件	120
7.1	WebLogic Workshop 控件简介	120
7.2	服务控件	121
7.3	时间控件	127
7.4	数据库控件	134
7.5	小结	144
第 8 章	高级控件	145
8.1	降低 J2EE 的开发难度	145
8.2	EJB 控件	146
8.3	JMS 控件	152
8.4	升级 Online Store 示例	170
8.5	使用控件工厂	182
8.6	小结	185

第 9 章 掌握 Web service 协议	186
9.1 消息传送	186
9.2 理解消息格式	198
9.3 理解 SOAP 报头	201
9.4 小结	207

第四部分 客户端组件

第 10 章 从 WebLogic Workshop 之外调用 Web service	210
10.1 为支持客户端而设计	210
10.2 产生代理	215
10.3 产生会话 ID	216
10.4 发送会话 ID	217
10.5 开发 JSP 客户端	218
10.6 开发 Swing 客户端	236
10.7 小结	242
第 11 章 在 XML 映射中使用 JavaScript	243
11.1 通过脚本来增强 XML 映射	243
11.2 JavaScript 基础	244
11.3 处理 XML 的 JavaScript 工具	250
11.4 编写映射脚本函数	263
11.5 XML 使用的扩展的小结	275
11.6 小结	277
附录 A Java 基础知识	278

第一部分 开始

- 第 1 章 WebLogic Workshop Web service 简介
- 第 2 章 开始使用 WebLogic Workshop
- 第 3 章 WebLogic Workshop IDE 入门

第 1 章 WebLogic Workshop Web service 简介

本章主要内容：

- Web service 简介
- J2EE 概要
- 说明 WebLogic Workshop 如何基于 J2EE 环境提供 Web service
- 说明使用 WebLogic Workshop 应该知道以及不需要知道的一些事项

在本章中，我们将通过说明在 Web service 这个新兴的舞台中 WebLogic Workshop 的通常用法来开始我们对于 WebLogic Workshop 的讨论。

首先，我们先简单讨论一下什么是 Web service，并回顾一些在 Web Service Community（也称为 Web service 社团，世界上有很多组织在讨论和制定各种各样适合于各种领域的 Web service 标准，这些组织通常称为 Web service Community，较著名的有 W3C、WS-I 等）中已被广泛接受的 Web Service 标准。然后，我们探讨 WebLogic Workshop 及其相关的两个领域：Web service 与称为 J2EE 的 Java 企业级应用标准。最后，我们讨论一下为了顺利地阅读本书以及使用 WebLogic Workshop，读者需要了解或者不需要了解的一些知识。

1.1 Web service 简介

Web service 的使用已越来越普遍。如果读者阅读本书，毫无疑问也是众多的对于使用 Web service 技术感兴趣的人们之一。当然，本书的读者可能有不同的知识背景，然而即使那些对于 Web service 技术有着很深理解的人们，相信也能够从本书中获得一些有用的知识和建议。如果读者对 Web service 技术了解不多，那么这个简介能够使你更好地理解 Web service。在这里，我们会接触很多 Web service 领域的概念和名词，但读者并不需要去完全领会它。毕竟，了解 Web service 的最好方法就是去亲手创建它，第 2 章将指导读者如何去这样做。

1.1.1 什么是 Web service

Web service 是通过网络对外提供服务的下一个方向。Web service 这个名字包括了过去几年来陆续提出的一系列标准。这些标准解决了通过网络对外提供服务时

所遇到的一些困难,例如不同操作系统上使用不同编程语言编写的服务之间的通信问题等等。关于这些问题,也有许多方案在争相使自己成为 Web service 的标准,但是随着 Web service 的逐渐成熟,有几个标准渐渐支配了整个领域,它们是 SOAP、WSDL 和 UDDI;我们将在以后的章节逐个讨论这些标准。然而,我们要谨记,在众多的标准之下,Web service 只提供一个简单的功能,即通过网络对外提供服务。

通过 Web service 技术对外提供的这些服务并不是什么新的东西,股票查询、提交订单、查阅产品名录等等今天已经存在的服务将重新包装,通过 Web service 技术提供给外界使用。Web service 的新颖之处并不在于如何将一个服务通过网络暴露在外,今天已经有很多技术可以做到这一点。Web service 的主要作用在于为这些技术提供一个可权衡的标准,而 WebLogic Workshop 将试图并希望能够成为使 Web service 开发尽可能简单容易的工具。

那些一直在苦苦探索计算机工业前进道路的人们,确实发现了应用 Web service 技术带来的几个新的发展方向。他们常常设想有一个巨大的服务目录,这些服务都用可被检索的标准格式来描述。一旦人们访问这样一个服务,这个服务的访问机制就会自动生成。不管这样的假设是否能够实现,现在的 Web service 技术处于为目前存在的各种技术提供标准的核心地位。我们的希望是,一旦提供了这样的标准,Web service 就成为了通过网络环境提供服务的媒介,并且统一目前所存在的各种技术,消除它们之间的藩篱。

让我们举个例子来说明一下。假设读者有一个网上书店,读者希望系统能够在图书缺货时自动地通知供应商。传统的做法是手工的过程,通常通过传真或者 E-mail 去完成。对端的供应商也会采用相同的手段接收读者的请求并给读者一个回应。但是,这些过程完全自动而不需要人工干预不是更好吗?读者的计算机系统和供应商的计算机系统能够互相对话、能够定时甚至是实时地对各种请求做出回应不是更好吗?这些问题的解决方案,在没有 Web service 之前,通常会为不同供应商因操作系统、编程语言的不同,为所写的程序都定义一套 API (Application Programming Interfaces, 应用程序开发接口),这样通常会比较麻烦。而 Web service 以及相关的标准能够使我们跳过并消除用常规做法所面临的障碍。

1.1.2 一些重要的标准

使用 WebLogic Workshop 经常接触到的两个标准规范是 WSDL (Web service Description Language, Web service 描述语言)和 SOAP (Simple Object Access Protocol, 简单对象访问协议)。WSDL 用于对一个 Web service 进行精确定义和描述,而 SOAP 则是 Web service 之间主要的通信协议。如果需要将 Web service 发布

到一个公用的服务目录中，则读者还需要接触和熟悉 UDDI (Universal Description, Discovery, and Integration, 统一描述、发现和整合协议)。

SOAP

SOAP (Simple Object Access Protocol, 简单对象访问协议) 是一个允许计算机之间传递标准 XML 格式消息的网络协议。SOAP 是一个比较灵活的协议。为了能够适用于各种各样的应用程序，SOAP 协议提供了很大的灵活性。作为对 XML 的增强，SOAP 消息中还允许有任意的附件，以用于存放使用单纯的 XML 难于表述的二进制数据。

SOAP 协议被 Web services community 采用为远程过程调用 (remote procedure calls, RPC) 的标准消息协议。RPC 也是 SOAP 的设计者们的最主要的设计意图之一。因此，在 SOAP 协议标准规范的第七部分中，主要描述了如何使用 SOAP 去实现 RPC。

WebLogic Workshop 使用 SOAP 去传递或者请求 Web Service 中的方法。从“面向对象”的观点来看，一个 Web service 可以看成是一个“对象”，这个“对象”中存在着一系列的方法。一个 SOAP 请求通过特定的 URI 去定位一个特定的“对象”，并通过名字来引用该“对象”的一个特定方法。这个方法的参数通过 SOAP 请求体中的 XML 内容来传递。因为参数是 XML 格式的，所以可以很灵活地将各种各样的信息传递给该特定的方法。

WebLogic Workshop 也提供了灵活地处理 SOAP 消息中的 XML 内容的工具，而不需要使用一系列的 API 去处理一个 XML 文档。通过一种称为“XML 映射”(XML mapping) 的技术，程序员可以很方便地使用 WebLogic Workshop 去处理 XML 消息。“XML 映射”是 JavaScript 的一个扩展，稍后我们将讨论它。

SOAP 标准规范由 W3C 来维护，有兴趣的读者可以到 W3C 的网站上查阅相关的知识，网址是 www.w3.org/TR/SOAP。

WSDL

回顾我们以前提过的网上书店的例子。在读者的计算机系统可以和供应商的计算机进行会话之前，必须先确认会话的主题和通信格式。这时需要一个标准规范去定义哪些类型的消息可以通过 SOAP 协议去传递。对于网上书店而言，我们必须定义：我将发送给你一个缺货的消息，其中包括书的 ISBN 号、书名和数量；而你必须回应可以供应的数量以及价格。

被 Web services community 所采用、用于进行这种描述定义的标准规范是 WSDL，即 Web service 描述语言。WSDL 是又一个基于 XML 的、用于描述网络服

务的高灵活性的标准规范。对于任何服务而言，一个WSDL主要精确地描述了可以“进出”该服务的消息类型。如同Web service领域的其他标准规范一样，WSDL尽可能地覆盖大多数Web service的需求，因此一个WSDL文件是非常复杂的。幸运的是，这个文件通常由计算机自动生成，很少需要人工去检查这些文件。

WebLogic Workshop能够为它创建的任何Web service生成WSDL文件；同样地，它也可以处理其他Web service的WSDL文件。通过这个机制，WebLogic Workshop可以方便地与其他第三方的Web service产品进行交互。

与SOAP协议一样，描述WSDL的标准规范也是由W3C来维护的，读者可以通过www.w3.org/TR/wsdl去查看相关的知识。不过这并不必要，正如前面所述，WSDL通常是由计算机自动生成的，一般不需要人们去查看这些文件。

UDDI

UDDI可以认为是Web service的“黄页”。UDDI是一些软件行业中很重要的公司提出来的一个标准，目的在于为各种各样的服务提供一个可分类管理的通用目录格式。UDDI的范畴非常广泛，可以通过UDDI来查询任何内容，如从普通的电话号码到描述Web service的WSDL文件。UDDI并不是与Web service直接相关的一个标准，但是在将来，读者很可能通过查找一个UDDI目录来找到相应的Web service。

WebLogic Workshop有一些简单的内建UDDI功能。UDDI是如此广泛，以至于不能在WebLogic Workshop中完全地整合进来。如果读者对UDDI感兴趣，可以查阅官方站点<http://uddi.org>，在该网站中，可以找到所有关于这个初生的协议的一些文档和信息。

1.1.3 为什么Web service 如此重要

最近，Web service得到了前所未有的宣传。读者可以看到，Web service可以解决（至少是声称可以解决）我们举的网上书店这个例子的所有问题；读者还可以看到无数“高手”向你举出无数的理由来说明Web service的普及，尽管各自强调的重点不同。

尽管这些理由甚至很有趣，但是这里面确实有些基本原因决定了Web service的普及。其中，最基本的原因在于Web service提供了一种低成本、高效地创建基于网络的服务的方法。这一点之所以重要，是因为有成千上万的地方希望他们的计算机系统能够方便地与其他计算机系统交互，并且方便地对外提供服务。当然，以HTML为中心的Web网站也为这个领域提供了一些可行的方法，使得为人们提供服务变得相对容易，但是，Web网站并不能解决这里提到的所有问题。因为它很少提

供有组织的目录服务(搜索引擎并不是一个很有组织的目录服务方式),而且服务本身是不标准的,通常还要依赖于人们的理解力才能够有效地使用。

基于 HTML 的服务最明显的弱点是,不适合于计算机与计算机之间的通信。HTML 可以有效地创建可以供人阅读的服务,但是不擅于创建计算机可以理解的服务。当今的大公司所面临的最大 IT 问题是如何整合他们的众多计算机系统,以便这些在不同环境下开发的独立系统能够互相通信,为现实的商务运作创造新的价值。而 Web service 正是为一个企业级环境中从 PC 机到大型机的各种独立系统的互联,提供了一个可重用的机制;为许多用不同编程语言开发、运行于不同操作系统上的系统,提供了相同的接口。

1.2 WebLogic Workshop 简介

WebLogic Workshop 是一个开发 Web service 的工具。它是一个基于 Java 的工具,意味着 Web service 的编程语言是 Java。并且, WebLogic Workshop 是运行在 WebLogic Server (BEA 公司的 J2EE 应用服务器)之上的。WebLogic Workshop 的独特之处在于,它提供基于 J2EE 的企业级 Web service 的开发。

1.2.1 J2EE 世界

读者也许熟悉也许不熟悉 J2EE。如果不熟悉,也不需要被这个复杂的企业级平台所吓倒。WebLogic Workshop 的好处在于,读者不必详细了解如何编写一个 J2EE 应用的所有细节就能够编写出一个良好的 J2EE 应用。也就是说,读者不需要详尽地了解 J2EE 的所有细节,而只需知道个大概即可。

J2EE 的核心概念是应用服务器 (Application Server)。应用服务器使得 J2EE 程序员可以创建访问众多企业级资源的 Java 程序:数据库、消息队列、安全与 DCOM 或者 CORBA 的通信等等。BEA 公司所提供的应用服务器是 WebLogic Server,这是一个在企业级环境中使用非常普遍的应用服务器。

提示:要想详细了解 WebLogic Server 的知识,可以参阅 Joe Zuffoletto 等人编写的 "WebLogic Server Bible" 一书。

在应用服务器中创建的程序通常可能是一些企业级的 Java Beans (EJB)。类似于面向对象编程中的“对象”,不过是增加了一些特殊行为的“对象”。它们被设计成为一种分布式组件,可以用来实现 workflow、异步调用、表现逻辑等等。

如果想要更加深入地了解 J2EE 或者 WebLogic Server 的知识,可以参阅我们提到的书籍。使用 WebLogic Workshop 并不需要知道那么多的 J2EE 知识,不过熟悉 J2EE 会更有助于对使用 WebLogic Workshop 开发复杂 Web service 的理解。

1.2.2 WebLogic Workshop 在 J2EE 中的位置

Web service 在 J2EE 世界并不是什么新东西。早就存在了一些基于 Java 的 Web service 解决方案,但是这些方案并不令人满意,它们既不容易使用又不能和 J2EE 的其他领域整合在一起。WebLogic Workshop 试图改变这个现状,它提供了一个基于 J2EE (因为它运行在 WebLogic Server 之上)且容易使用的 Web service 开发工具。

WebLogic Workshop 提供了一个简单易用的 IDE 界面,读者可以通过这个 IDE 界面开发运行在 WebLogic Server 之上的 Web service。也就是说,通过这个工具,可以快速地开发运行在一个企业级应用服务器之上的 Web service。

1.3 使用 WebLogic Workshop 不需要知道的事项

读者现在也许会急着使用 WebLogic Workshop,但在此之前,我们还需要知道一些事情。

WebLogic Workshop 的中心思想在于使得开发者可以使用 J2EE,而不需要深入了解纷繁复杂且庞大的 J2EE 规范。因此,即使我们对 J2EE 一无所知,也能使用 WebLogic Workshop。事实上,我们必须知道的是,WebLogic Workshop 是运行在 WebLogic Server 之上的。如果要运行一个基于 WebLogic Workshop 开发的 Web service,则首先要运行 WebLogic Server。

有一个东西我们不能使用 WebLogic Workshop 7.0 去创建,那就是 EJB。这超出了 WebLogic Workshop 的范围。而且如果我们开始创建 EJB,就必须去熟悉 J2EE 的知识。幸运的是,WebLogic Workshop 允许我们做许多事情,而不需要创建 EJB。

1.4 使用 WebLogic Workshop 需要知道的事项

WebLogic Workshop 使我们不需要了解很多 J2EE 的细节,但还是必须掌握有些知识。第一个要求是必须了解 Java 语言。本书假设读者熟悉 Java 语言,如果不熟悉,则必须先阅读一些有关 Java 的书籍。

1.4.1 JavaScript 编程

JavaScript 是一种类似于 Java 语言的脚本语言。作为一门简单的语言,它被广泛用于编写交互性网页。它是一种与 VBScript 相竞争的、用于编写动态网页的脚本语言。

WebLogic Workshop 中包含了一个可以快速简便地处理 XML 文档的扩展 JavaScript。因为 XML 在 Web service 中占有显著地位, 这种脚本语言就被用于处理能够在 Web service 和 Web service 客户端之间传递的 XML 文档。

尽管了解 JavaScript 并非必须, 但是在 WebLogic Workshop 中使用高级的 JavaScript 功能是非常有用的。当需要使用 WebLogic Workshop 开发复杂的 Web service 项目时, 就必须使用这些 JavaScript。我们将在本书的第 11 章讨论如何在 WebLogic Workshop 中使用 JavaScript, 并在那里提供一个 JavaScript 的快速入门。但是详细介绍 JavaScript 的知识超越了本书的范围, 读者还需要阅读一些关于 JavaScript 的书籍, 以了解这种脚本语言的每一个细节。

1.4.2 WebLogic Server 配置

读者不需要了解 WebLogic Server 的每一个细节, 但必须熟悉如何配置 WebLogic Server。WebLogic Server 的配置信息通常保存在一个称为 config.xml 的文件中。读者可以使用 WebLogic 控制台 (WebLogic console) 或者手工编辑这个 XML 文件。如果不熟悉应用服务器, 则 WebLogic 控制台是个简单易用的方式。在默认情况下, 可以通过输入 `http://localhost:7001/console` 进入 WebLogic 控制台。当读者需要自定义配置时, 对 WebLogic Server 进行配置就显得非常重要。比如, 读者不想使用 WebLogic Workshop 自带的数据库配置, 而想将它改为自己的数据库配置时, 就需要进行 WebLogic Server 的配置。

1.5 小结

此时, 读者也许还不能确定是否了解什么是 Web service 或者该如何使用它。如果觉得对 Web service 的概念和用法非常熟悉, 那自然是一件好事; 如果对它感到有些迷惑, 也不要沮丧。与用户界面或网页不同, Web service 并不是我们大多数人日常需要打交道的东西。因此, Web Service 对读者而言有些陌生也不足为怪。当我第一次听说 Web service 时, 也感到很陌生。最好的方法就是钻进去并且开始学习开发 Web service。这样, 就能够在短时间内对什么是 Web service 有一个很直观的理解, 并且很自然地去使用它们。

- Web service 是创建网络服务的标准方法。
- WebLogic Workshop Web service 基于 J2EE 之上。