

XML

高级编程 (第2版)

(英) Mark Birbeck 等著

裴剑锋 高伟 徐继伟 等译

p2p.wrox.com 提供在线支持



机械工业出版社
China Machine Press



乐思

Wrox 程序员参考系列

XML 高级编程

(第 2 版)

(英) Mark Birbeck 等著

裴剑锋 高 伟 徐继伟 等译



机械工业出版社
China Machine Press

本书是《XML 高级编程》的新版本，它更新了 XML 的最新技术发展，为核心 XML 技术提供了一个完整而实用的基础，并展示了在计算方面的一些重要应用。书中主要包括 XML 基础、编程、数据、表示、元数据及 B2B 应用。

本书适合有经验的网站开发人员、XML 应用开发人员阅读。

Mark Birbeck, et al: Professional XML, 2nd Edition.

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

Original copyright © 2001 by Wrox Press. All rights reserved.

Chinese simplified language edition published by China Machine Press.

本书中文简体字版由英国乐思出版公司授权机械工业出版社独家出版。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，侵权必究。

本书版权登记号：图字：01-2001-4404

图书在版编目（CIP）数据

XML 高级编程（第 2 版）/（英）布伯克（Birbeck, M.）等著；裴剑锋等译. - 2 版. - 北京：机械工业出版社，2002.5

（Wrox 程序员参考系列）

书名原文：Professional XML, 2nd Edition

ISBN 7-111-09855-2

I .X… II .①布…②裴… III .可扩充语言, XML-程序设计 IV .TP312

中国版本图书馆 CIP 数据核字（2002）第 011050 号

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：张金梅

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2002 年 5 月第 2 版第 1 次印刷

787mm×1092mm/16·67 印张

印数：0 001-4 000 册

定价：108.00 元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

译者序

XML 几乎在软件开发的每个方面都产生了巨大的影响。XML 具有一种开放的、可扩展的、可自描述的语言结构，它已经成为网上数据和文档传输的标准。为使 XML 能够进行有效性验证、定位、转换、链接、查询、描述和数据通信，与 XML 相关的技术正在以非常惊人的速度持续发展。

本书第 1 版讲述 XML 实现的基本技术，第 2 版则是第 1 版的扩展，主要讲述 XML 的相关技术，加入了 XML 技术最新发展的内容。因此本书不能取代第 1 版，本书适合于有一定基础的 XML 开发人员。全书共分六个部分。第一部分基础篇，包括 XML 介绍、XML 语法、Infoset、文档类型定义、XML 模式、XDR、XPath、XSLT、XLink、XPointer；第二部分编程篇，包括文档对象模型、SAX 2、基于模式的编程；第三部分数据篇，包括数据建模、XML 数据绑定、XML 查询；第四部分表示篇，包括 SVG、VoiceXML、XSL-FO；第五部分元数据篇，包括 RDF、RDDL；第六部分 B2B 应用，包括 SOAP、微软的 BizTalk B2B 解决方案、电子商务集成、WSDL 和 UDDI。

本书适用于那些想学习新的表示技术及使用 XML 改进内容管理的开发人员。对于企业解决方案的开发人员，特别是想知道如何用一种可扩展的方式通过网络标准来链接应用程序，或者是想了解新兴的网络服务功能如何使互操作性成为可能的开发人员，这是一本必读的书。通过阅读本书，还可以学习到如何将 XML 集成到现有的企业应用中，以及如何在数据库中使用 XML。总之，本书是 Wrox 公司继第 1 版后在 XML 方面又一富有创造性的巨著。对于任何想成为 XML 专家的人来说，本书都极具参考价值。

本书由欧阳宇组织翻译，主要的翻译人员有裴剑锋、高伟、徐继伟、黄毅、马震晗、李挺文、刘克科，参与翻译的人员还有牛韬、王安鹏、谢君英、姚继锋、张静、张小娣，杜启星、王小强、樊少菁、祁力、陆嫻、阳爱军、朱翠莲、武堂、胡建钧、唐子秀、余啸海、田青、杜芳、李新琳、邓浩、武泽民、裘强、文武、高健、李满秀、易重英、李平。由于翻译时间仓促，错误之处在所难免，敬请读者谅解并指正，我们接收广大读者意见的邮箱为：yingyu@263.net。

欧阳宇

2001 年 12 月于北京

作者简介

Mark Birbeck

Mark Birbeck 是国会通信有限公司的技术主管。他负责设计和建造他们的网站——ePolitix.com。他也是 x-port.net 有限公司 XML 顾问的常务理事，负责 spiked-online.com 的发布系统。尽管从事 XML 的研究已有多多年，但他对元数据却有着特别的兴趣，尤其是对 RDF 的应用。他对 Wrox 为宣传这些主题并使其成为主流所进行的积极活动表示非常欢迎。

Jon Duckett

Jon 自 1998 年与人合著了 Wrox 的第一个 XML 出版物以来，就开始使用 and 描述 XML。他过去三年一直在 Wrox 的英国伯明翰办事处工作，现在他在悉尼上班，所以能在上班时泡上一杯香浓的茶，从窗外感受一种不同的景观。

Oli Gauti Gudmundsson

Oli 为 SALT 工作。他是 SALT 系统的两名最重要的系统设计者之一，也是 SALT 在纽约的开发主管。他当前正在做的工作是将 XML 和 XSL 融入 SALT 的网页制作和内容管理系统中去。他还是冰岛大学计算机科学课中 Java 课程的讲师，Java 是他的专长，也是他的爱好。他在业余时间获取计算机工程学的学士学位。通过电子邮件地址 oli.gauti@salt.is 可以与 Oli 取得联系。

Pete Kobak

Pete Kobak 在 1978 年用一个工具包建造了他自己的第一台计算机并加上了软件，这台计算机有 256 个字节的 RAM 和一个单个的发光二极管输出。在 IBM 作为电子工程师进行了一段时间的拼搏以后，Pete 已经逐渐进入了大型计算机支持软件的开发行业。在 20 世纪 80 年代末期他为 Burroughs 公司的 Fortran 编译器进行改进时引入了 DO 循环的向量化，这为他赢得了“编程怪才”的称号。为了验证他的想法，他不断地进行冒险，在 1991 年 Pete 离开了 Burroughs 公司，投身于危难之中的 OS/2 环境下的医学试验室的软件开发。在 1997 年，Pete 被先锋创作组 (The Vanguard Group) 雇用进行 Solaris 的网络开发。自从那时起，他就开始帮助他们为网站增加一些新的特色，并专攻安全网络通信。

Pete 现在的兴趣是研究网络应用安全。他正在试着找出一种合适的技术来提高严格的金融制度所需要的强大的安全保障，一旦这种需求得到满足，商务关系将会得到迅速的扩展。Pete 非常感谢能引入有趣的网络技术，帮助上百万人实现其发财梦。可以通过电子邮件地址 kobak@ieee.org 与其联系。

Evan Lenz

Evan Lenz 是一个软件工程师，他目前在美国华盛顿州西雅图的 XYZFind 公司工作。他的主要专长是 XSLT，他喜欢探索新的方法来利用这种技术开发各种各样的项目。他在 XYZFind 公司的工作范围很广，从 XSLT 和 Java 开发，到撰写用户手册，到设计用于 XYZFind 的 XML 数据库软件的 XML 查询语言，几乎涉及每一件事情。他拥有一个专业音乐等级和一个哲学学士

学位，他希望有一天能把自己广泛的兴趣集中到一起，设计出一个精妙无比的方案。

Steve Livingstone

Steve Livingstone 是 IBM Global Services 的一个 IT 工程师，现在加拿大的温伯尼亚湖工作。他为众多的 Wrox 书籍和杂志撰写过文章，其文章的主题范围很广，从 XML 到电子商务都有他的文章。Steve 当前的兴趣包括电子商务、ebXML、.NET 和企业级应用体系结构。

Daniel Marcus

Daniel Marcus 博士在软件体系结构和设计上已有 20 年的经验。他是 Speechwise 公司的联合创始人之一，现任其总裁兼首席执行官。Speechwise 公司是一家应用软件公司，它专门从事综合语音、无线和网络技术的应用软件的开发。在来 Speechwise 公司之前，Daniel Marcus 博士是 Xpedior 的电子商务顾问的主管，负责 Global 2000 和 dot-com 客户端电子商务应用的策略、体系结构和调度。Marcus 博士曾经作为在普林斯顿工学院进行高级研究的一名访问学者，也是在 Lawrence Livermore 国家实验室的研究科学家，并撰写了 20 多篇有关计算科学的论文。他还是经过 SUN 认证的 Java 技术工程师，在伯克利的加利福尼亚大学，他获得了机械工程学的博士学位。

Stephen Mohr

Stephen Mohr 是美国费城 Omicron Consulting 公司的一个软件系统工程师。他在各种平台和组件技术上已经有十多年的工作经验。他的研究兴趣主要在于分布式计算和人工智能。他在美国 Rensselaer 理工研究所管理学院获得了学士和硕士学位。

Nikola Ozu

Nikola Ozu 是一个独立的系统工程师。他当前正在做的项目就是用 XML 将语义数据库、文本搜索和多媒体组件结合到一起——通向世外桃源之路。最近他的工作还包括一般的网页设计顾问、一些 XML 词汇表、还有一些基于 XML 的作品和医学参考书籍出版商及数据库全文检索系统。

在 20 世纪 90 年代初期，Nik 设计和开发了健康参考中心（Health Reference Center）——一个超文本数据库。接着又开发了高级版本的 InfoTrac。这两个项目都是书目和全文本数据库，都发行期刊的光盘，其内容每月进行更新。由于包含了大型文本数据库，所以不可避免的与 SGML 有一些联系。Nik 早先的工作是基于主机的图书馆系统，后来改为嵌入式微系统（电信设备、工业机器人、玩具、连拱廊游戏和视频游戏盒）。在 20 世纪 70 年代早期，他曾致力于接线板、穿孔卡、纸带、打印输出（和电传机）编程。

Jonathan Pinnock

早在摩尔定律提出以前，Jonathan Pinnock 就在自己学校只有 4KB 内存的 PDP 8/e 机上用 Pal III 汇编语言进行过编程。近来，他的大多数时间用在了开发和扩展 PlatformOne 系列产品上，PlatformOne 是其公司 JPA 面向金融服务团体的产品，该产品正在获得越来越大的成功。他似乎把所有的业余时间都用在为 Wrox 写作上了。JPA 的主页是 www.jpassoc.co.uk。

Keith Visco

Keith Visco 目前在 Intalio 有限公司工作，是商务过程管理的领导，也是基于 XML 技术的管理人员和项目负责人。Keith 还是 Castor 开放源代码数据绑定框架的项目负责人。自 1998 年起，

Keith 就一直积极地从事开发开放源代码项目的工作, 包括 Mozilla 项目。Keith 是 Mozilla 的 XSLT 处理器最初的作者, 这一处理器是由 Keith 的前任雇主——MITRE 公司捐赠的。Keith 还是当前 XSLT 模块的拥有者。

Andrew Watt

Andrew Watt 是一个独立顾问, 他喜欢做的事不多, 除了对技术进行研究外, 他很少去尝试别的事情。自从 20 世纪 80 年代中期用 6502 汇编语言和 BBC 基础编写了他的第一个程序之后, 他尝试使用了 Pascal、Prolog 和 C++ 等其他的编程语言。最近他集中精力研究包含 Lotus Domino、Java 和 HTML 在内的网络相关技术。他当前的兴趣在扩展标记元语言 (Extensible Markup Meta Language, XMML) 的各种应用上。XMML 有时被称为 XML, 这是不严密的, 并且很容易引起误解。目前他对 SVG、XSL-FO、XSLT、CSS、XLink、XPointer 等技术的看法是, 如果把它们融合在一起将会是一件令人激动不已的事情。尽管这种技术令人畏惧, 但其前途是光明的。他已经开始用 XQuery 研究这种技术。他不停地进行学习, 用他自己的话说, 这叫“生命不息, 学习不止”。

Kevin Williams

Kevin Williams 在计算机方面的第一次尝试是在 10 岁 (1980) 时, 他参加了当地社区大学举办的 PDP-9 上的 BASIC 培训班。12 岁时, 他连续 4 天在自己的 Atari 400 机器上人工汇编 6502 代码。他的职业生涯是专门从事 Windows 开发, 一开始是客户 - 服务器开发, 然后又转向了 Internet。他在许多方面都有所涉足, 从 VB 到 PowerBuilder、Delphi、C/C++、MASM、ISAPI、CGI、ASP、HTML、XML, 以及任何其他你可想到的术语缩写。不过最近这些日子, 他的确集中精力进行 XML 的研究了。Kevin 是 Equient 公司的高级系统工程师, Equient 公司是位于维吉尼亚北部的一个信息管理公司。Kevin 的电子邮箱是 kevin@realworldxml.com。

Zoran Zaev

Zoran Zaev 是一个高级网络工程师, 在位于华盛顿的 Hitachi Innovative Solutions 有限公司工作。在 20 世纪 80 年代, 他就已经从事技术工作了。在 20 世纪 90 年代中期, Zoran 开始从事于网络应用的开发。从那以后, 他通过网络应用帮助过许多大客户和小客户。他最近的重点是使用 XML、SOAP 和其他相关技术进行网络应用和网络服务的开发。通过电子信箱 zoran.zaev@hitachisolutions.net 可以与 Zoran 取得联系。

前 言

可扩展标记语言 (Extensible Markup Language, XML) 作为一种计算机技术出现不过是短短几年的事情。它是一种概念, 以其简便性颇得人心, 并为 Internet 应用程序开发方式带来了戏剧性的变化。本书是对第 1 版的修订, 以便跟踪快速变化的技术, 包括一些被更新的和新出现的技术。

本书的覆盖范围

本书解释和演示了设计和使用 XML 文档所必需的基本技术, 以及很多现在非常重要的相关技术。本书中几乎所有的内容都是依据万维网协会 (World Wide Web Consortium, W3C) 整理编撰的 XML 规范。这些规范完整程度都不一样, 而且其中的一些技术是刚出现的, 因为它们都非常重要和有用, 我们希望这些规范在确定之后能够得到广泛使用。虽然越来越多的 XML 社团在不断涌现, 并提出崭新的超出 W3C 控制之外的与 XML 相关的概念, 但对于 XML 的开发来说, W3C 仍然是非常重要并处于中心地位。

本书的焦点集中在如何使 XML 与现实生活中的应用结合在一起。它展示了很好的设计技术, 演示了如何在 XML 应用程序和 Web 应用程序之间提供接口。不管你的需要是面向数据交换还是数据表示, 本书将覆盖 XML 领域的所有相关的技术。

大多数章节都包含一个实践范例 (除非该技术刚出现, 在写作时还没有实际应用实现)。由于 XML 是一种独立于平台的技术, 所有这些例子覆盖了各种语言、解析器和服务器。由于不同平台下的技术都具有一些共性, 所以即便它们不是在你最习惯的平台上实现的, 你仍然可以从这些例子中得到许多有益的知识。

本书的读者

本书适用于有经验的开发人员, 他们已有一些 XML 方面的基础知识, 并且希望使用这项令人兴奋而又简单的技术构建高效应用。Web 站点的开发者能够学到使用 XSLT 样式表和其他一些技术, 从而把自己的站点提高一个层次。而编程者和软件系统程序员能够学会如何让 XML 和他们的系统相配合, 以及如何使用它来解决应用程序集成中的问题。

XML 应用是分布式的而且通常是面向 Web 的。本书专注于分布式和 Web 应用的开发, 所以你需要有一些多层体系结构方面的知识, 尤其是网络领域的。虽然重点在于介绍 XML 的新特性, 但由于读者可能缺乏一些 XML 方面的基础, 我们会先全面而简要地介绍所有的规范。

本书中的样例将用到各种编程语言, 但不要求读者对这些编程语言非常熟悉。本书介绍的技术都可以转换成其他的编程语言, 这是因为 XML 是一种平台语言无关的技术。本书主要使用 Java 语言, 这主要是因为它提供了很多管理 XML 的工具, 另外还会用到 JavaScript、VBScript、VB、C# 和 Perl 语言。我们希望读者熟悉其中一种编程语言即可。

本书的组织结构

虽然很多作者参与了本书的写作，但我们尽量把这些章节按照主题划分，本书主要分六个部分。

有一些章节依赖于前面的章节中介绍的技术，例如第 23 章，这样做的目的是为了清楚一些。大多数的章节都是很独立的。

XML 正在逐步进入标记技术相关的广泛领域，这种成长增强了 XML 应用。但是，随着这种成长也出现了分歧。不同的读者可能会抱着不同的期待来阅读本书。XML 对于不同的人来说是完全不同的事情。

1. 基础篇

第 1 章简单介绍了 XML 世界，讨论了目前以及将来要用到的一些相关技术。第 2 章（XML 基本语法）和第 3 章（XML 高级语法）覆盖了 XML 1.0 的基础内容。第 2 章介绍了 XML 文档的基本语法，第 3 章介绍了一些稍微高级的问题，例如命名空间。这些章节提供了理解和使用 XML 所必需的最基本内容，根据你的实际经验，可以跳过这些介绍性的章节。第 4 章介绍了 Infoset，它是一种描述 XML 的标准方式，该方式提供了 XML 数据的抽象描述。

第 5 章介绍了使用 DTD 进行文档验证方面的知识，虽然在后面的两章会介绍其他的替换 DTD 的基于模式的验证语言，但 DTD 仍然有它们的用武之地，因为很多 XML 解释器使用 DTD 而不是其他模式语言。第 6 章介绍了 XML Schema，并且介绍了如何通过 W3C 规定的新的基于 XML 的验证语言验证 XML 文档。第 7 章介绍了其他的基于模式的验证语言，包括 James Clark 的 TREX 建议和 Schematron。

第 8 章介绍了 XPath 规范，它是一种新的查找特定 XML 段的方法，通常和其他 XML 技术一起使用。主要包括在第 9 章中介绍的 XSLT。第 9 章会介绍如何根据特定样式表的声明，将 XML 文档转换为其他形式的方法。第 10 章介绍各种链接技术，例如 XLink 和 XPointer，同时还描述了 XML 文档片段交换规范。

这 10 章足够你了解 XML 技术中所有可以即时掌握的技术。你可能已经有很多使用 XML 的经验，但这些章节中介绍了一些更新的技术，每个人应该都能学会一些新的东西，尤其是因为 XML Schema 就在本书印刷之前两个月，已经处于建议推荐状态。虽然 XML 技术很多，但掌握这些技术会建立一个扎实的基础。

基础篇包括：

- 第 1 章 XML 简介
- 第 2 章 基本 XML 语法
- 第 3 章 深入的 XML 语法
- 第 4 章 XML 信息集合
- 第 5 章 验证 XML 有效性：文档类型定义
- 第 6 章 XML Schema 介绍
- 第 7 章 其他 Schema
- 第 8 章 XML 导航工具——XPath

- 第 9 章 转换 XML
- 第 10 章 文档片断、XLink 和 XPointer

2. 编程篇

由于 XML 可以被机器和人阅读，所以为了操作 XML 数据建立了一些标准 API。这些 API 通过 JavaScript、Java、Visual Basic、C++、Perl 以及其他很多语言实现。它们提供了操作和开发 XML 文档的标准方式。

第 11 章中，我们向大家讲解了第一个 API，来自 HTML 世界的文档对象模型（DOM），它是 W3C 发布的规范，并且最近发布了该规范的 Level 2。XML 数据可以被看做是层次的和面向对象的，DOM 提供了获取和操作 XML 节点的方法和属性。第 12 章讨论了用来替换 DOM 的 SAX，在操作 DOM 时，整个文档都被读入内存中，而使用 SAX 只需要获取和操作特定节点所需的数据即可。

第 13 章是该部分的最后一章，它介绍了 XML 的描述性编程，大多数程序员都使用过程化的语言，但 XML 和 XML 规范并不关心特定的语言和应用是如何完成任务的，只关心是否按照声明的去做。该章解释了如何使用模式设计应用。

编程篇包括：

- 第 11 章 文档对象模型
- 第 12 章 SAX 2
- 第 13 章 基于模式的程序设计

3. XML 数据篇

该部分总共有四章，主要讨论与 XML 相关的数据的存储、获取和操作。第 14 章介绍了如何正确设计你的项目，以及如何根据数据建立 XML 模型，并且根据模型建立更好的应用。第 15 章通过介绍 XML 的数据绑定进一步扩展了这种概念。查询 XML 一章中介绍了称为 XML 查询（XML Query）的新技术，它的目的是在 XML 格式中提供强大的 SQL 查询能力。这章很短，只是从编写的角度介绍了如何使用这种技术。

最后一章介绍了一个实例，它描述了如何让数据库同 XML 数据相关，如何以最佳的方式集成 XML 和 RDBMS。

XML 数据篇包括：

- 第 14 章 数据建模
- 第 15 章 XML 数据绑定
- 第 16 章 查询 XML
- 第 17 章 案例研究：XML 和数据库

4. 表示篇

第 18 章介绍了称为 SVG（Scalable Vector Graphics）的 XML 技术，这种技术同适当的阅读器（例如，Adobe SVG Viewer）配合，能显示和操作非常详细的图形文件。第 19 章介绍了 VoiceXML 技术，它能在 Web 上实现语音识别和处理。XML 数据能被转换为 VoiceXML，并且使用适当的技术，能通过电话发声和交互。

第 20 章介绍了最后一种技术 XSL-FO，它是一种新出现的技术，能精确调整网页的布局，

就像 PDF 处理的方式一样。它们的区别在于这里是 XML，并且可以通过你习惯的 XML 工具操作，如果用户没有 XSL-FO 阅读器，XSL-FO 格式也可以被转换为 PDF 格式。

XML 表示篇包括：

- 第 18 章 XML 的图形化表示
- 第 19 章 VoiceXML
- 第 20 章 XSL 格式化对象：XSL-FO

5. XML 元数据篇

在该部分中，主要讨论如何用 XML 来表示元数据，也就是数据的意义或语义，而不是数据本身。在第 21 章中介绍了设置 XML 数据的索引，这章使用了 Java 索引应用，实际上，这种技术可用于任何索引工具。第 22 章就是我们本部分的真正主题，这里介绍了 RDF，它是一种描述元数据的语言。在这一章中介绍了这种技术的基本元素和语法。第 23 章中，在描述 RDDL（一种在命名空间 URL 绑定资源的方法）前，我们介绍了一些 RDF 技术的应用实例，一个使用 RDDL 的应用能教你学会命名空间涉及的技术，这项技术是什么，以及它是如何访问 Schema 和标准转换的。

XML 元数据篇包括：

- 第 21 章 案例研究：生成站点索引
- 第 22 章 RDF
- 第 23 章 RDF 编码实例和 RDDL

6. XML 在 B2B 中的应用

本书的最后一部分描述可能是 XML 最重要的用途——B2B 和 Web 服务。在过去，B2B 的通信协议（例如 EDI）就开销和处理器负担而言都不是一般人能使用的，而且代价昂贵。使用 XML 词汇，它是一种能用在 B2B 商务中的开放的和可编程的模型。

在第 24 章中介绍了简单对象访问协议（Simple Object Access Protocol, SOAP），SOAP 是微软建立起来的（虽然 W3C 正在开发 XML 协议规范，它也应该同 SOAP 非常相似），它能允许两个应用程序使用 XML 指定服务。我们介绍这个协议的复杂部分，这样你可以用它网络化任何想要提及的服务。

第 25 章介绍了微软的 BizTalk 服务器，这种服务器可以使用开放 BizTalk 框架控制所有的 B2B 事务。BizTalk 只是使用 SOAP 的一种方法，来操纵业务事务，但它是微软提供的，并且使用非常广泛。第 26 章中讨论了一个使用 XML 集成的电子商务实例，该章中包括有相当多的金融业务标准。同时还解释了如何集成所有的标准，而不需要因为每个竞争的标准之间可能的 B2B 交易书写代码。

第 27 章是最后一章，在这一章讨论了 Web 服务描述语言，它允许我们通过定义 SOAP 或其他的客户能够连接的服务来规范 XML 词汇。WSDL 描述了每一种服务以及它所做的事情。另外，在该章还介绍了 UDDI（Universal Description, Discovery, and Integration，通用描述、发现与集成），它是提供了包含多种服务的自动发现和事务的方法。在很多情况下，它对于与人的查找服务交互来说不是必需的。并且 UDDI 使得使用公共注册服务成为可能。这两种技术都是新出现的，但随着在业务中的应用越来越广泛，它们会变得越来越重要。

B2B 篇包括:

- 第 24 章 SOAP
- 第 25 章 利用 Microsoft BizTalk Server 实现 B2B
- 第 26 章 电子商务集成
- 第 27 章 B2B 的未来: WSDL 和 UDDI

阅读本书前的准备

本书假设你已经掌握了一些关于 HTML 的知识。一些面向对象的编程语言 (例如, Java、VB 和 C++) 以及基本的 XML 知识。对于本书中的一些例子, 需要安装 Java 运行环境 (<http://java.sun.com/j2se/1.3/>), 某些章节还需要诸如 MS SQL 服务器、MS Index 服务器和 BizTalk 服务器等软件。

本书中的大部分代码都可以从 <http://www.wrox.com/> 下载, 详细情况在后面的“技术支持、勘误和 P2P”部分中有介绍。

本书约定

为了帮助读者从本书中获得更多的知识, 同时跟踪正在发生的事情, 在本书中使用了一些约定。

例如:

框中的文字表示与当前主题相关的重要信息。

这样的文字表示一些题外的信息。

技术支持、勘误和 P2P

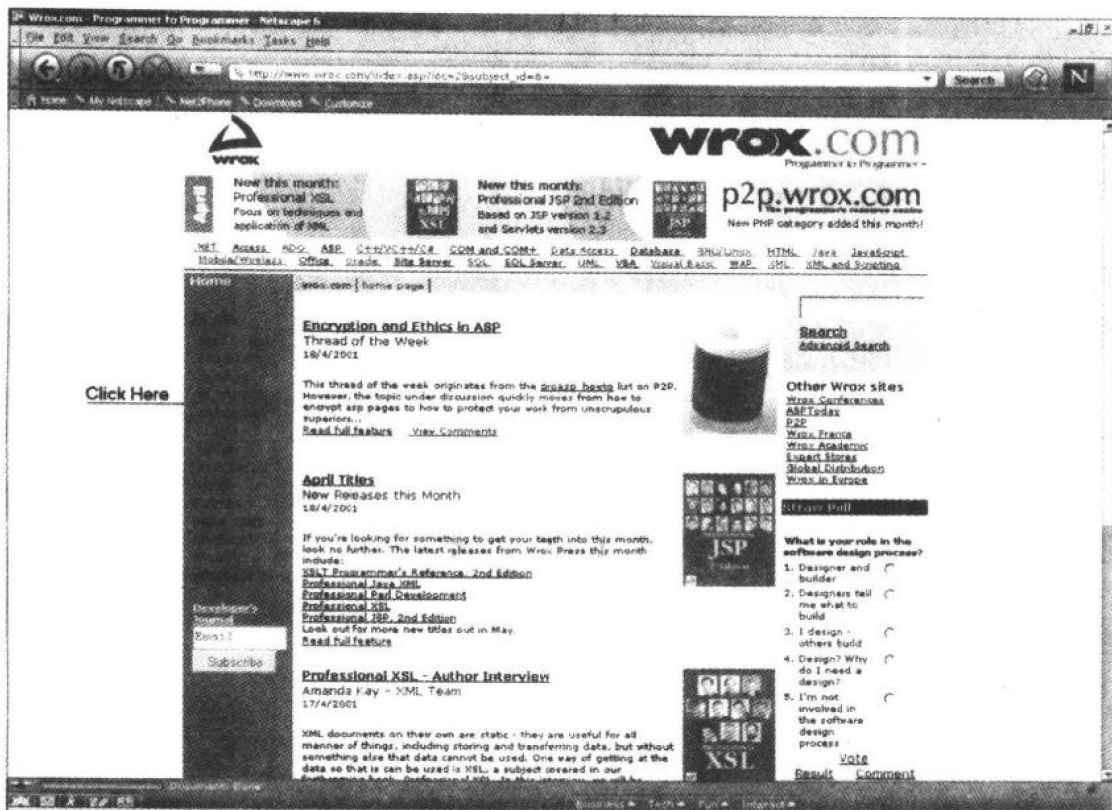
本书的印刷和销售只是我们与你联系的开始。如果有什么问题, 或者是代码和本书中的解释不正确, 我们欢迎你给我们发 E-mail, 我们的邮件地址是 support@wrox.com, 我们会在两至三天内回复 (根据支持小组的繁忙程度而定)。

另外, 我们也在线发布勘误表, 这样如果你有什么问题, 可以访问 Wrox 的 Web 站点, 看是否已经更新了信息。首先, 访问 www.wrox.com, 然后单击网页左边的 Books | By Title (Z-A) 或者 Books | By ISBN 链接, 如下图所示。

找到该书 (ISBN 为 1861005059), 然后单击。除了提供本书的相关信息外, 还提供了下载代码、查看勘误表和寻求支持的选项。只需单击相关链接即可。所有我们发现的勘误都会加到网站上, 另外针对更新的软件对代码所做修改的信息也会在这里看到, 同时还包括印刷或代码错误。

本书的全部代码都可以从 <http://www.wrox.com> 下载, 这些代码包括在压缩文件里, 以章节名命名。

另外, 我们在 p2p.wrox.com 建立了程序员与程序员讨论组, 这里有一些书的相关信息, 并且你可以在这里贴出你的问题, 并且将得到 Wrox 的其他人或者其他开发社团成员的解答。访问 <http://p2p.wrox.com>, 并且订阅讨论列表, 所有的列表都将自动发送到你的信箱里。



保持联系

我们会努力工作力图让这本书对大家尽可能有用。因此我们需要知道大家的想法，我们希望知道大家想要什么和需要知道什么。

如果大家能够对本书提出宝贵的意见和建议，我们将不胜感激。我们的电子邮件地址是：

feedback@wrox.com

或者可以通过反馈链接与我们联系：

<http://www.wrox.com>

目 录

译者序
作者简介
前言

第一部分 基础篇

第 1 章 XML 简介	1
1.1 XML 的起源和目的	1
1.1.1 标记语言	1
1.1.2 SGML 和文档标记语言	4
1.1.3 XML 的目标	5
1.2 W3C 规范各个阶段	10
1.3 XML 核心	11
1.3.1 XML 1.0 语法	11
1.3.2 描述和验证	14
1.3.3 XML 命名空间	17
1.3.4 XML 信息集合	18
1.3.5 定位和链接 XML	18
1.3.6 转换 XML	20
1.4 XML 实际应用	21
1.4.1 XML 词汇表	21
1.4.2 XML 编程	25
1.5 XML 表示	26
1.6 用于服务器和数据库的 XML	28
1.7 可扩展通信协议	29
1.8 电子商务的基础	30
1.9 语义和元数据	32
1.10 小结	33
1.11 参考资料	34
第 2 章 基本 XML 语法	36
2.1 标记语法	36
2.1.1 字符	37
2.1.2 简单命名	39
2.2 元素	40
2.2.1 标记	41

2.2.2 空元素标记	41
2.2.3 标记：一个简单的例子	42
2.3 XML 数据结构	43
2.3.1 层次树结构	44
2.3.2 字符串	47
2.4 属性	47
2.5 字符数据	49
2.5.1 空白	49
2.5.2 行尾的处理	50
2.6 元素和属性的语法摘要	51
2.7 字符和实体引用	51
2.7.1 字符引用	51
2.7.2 实体引用	52
2.8 专用标记	54
2.8.1 注释	54
2.8.2 处理指令	55
2.8.3 CDATA 节	56
2.9 文档结构	57
2.9.1 序言	57
2.9.2 主体	58
2.9.3 尾声	58
2.10 格式正规的 XML 文档	59
2.11 有效的 XML 文档	60
2.12 XML 解析器	60
2.12.1 解析器级别	60
2.12.2 解析器实现	61
2.12.3 一些特殊解析器	62
2.12.4 解析器链接网站	63
2.12.5 XML 语法小结	63
2.13 实例：简单的发货单	64
2.14 小结	66
第 3 章 深入的 XML 语法	67
3.1 进一步要考虑的问题	67
3.1.1 特殊属性	67
3.1.2 属性值规范化	70

3.2 现有标准的使用	71	5.1.1 数据需要的描述和验证	106
3.2.1 XML 1.0 参考	71	5.1.2 有效的 XML	106
3.2.2 ISO	73	5.1.3 文档类型定义	107
3.2.3 Unicode	75	5.2 DTD 结构	109
3.2.4 IETF	77	5.3 将 DTD 和 XML 数据关联	110
3.2.5 IANA	83	5.3.1 文档类型声明	110
3.3 XML 的命名空间	84	5.3.2 内部和外部 DTD 子集	113
3.3.1 XML 命名空间的概念	84	5.4 基本 DTD 声明	113
3.3.2 如何保持 XML 命名的惟一性	85	5.4.1 元素类型声明	114
3.3.3 声明命名空间	87	5.4.2 属性声明	123
3.3.4 默认命名空间	88	5.4.3 表示法: 非 XML 数据	132
3.3.5 作用域	89	5.5 实体	133
3.3.6 命名空间和属性	90	5.5.1 已解析实体	134
3.3.7 小结: 命名空间	91	5.5.2 未解析实体	134
3.4 XML Base	91	5.5.3 实体引用	134
3.5 实例: 使用命名空间的发货单	93	5.5.4 一般实体	136
3.6 小结	95	5.5.5 参数实体	140
3.7 参考资料	95	5.5.6 ISO 和其他标准化实体集	141
第 4 章 XML 信息集合	97	5.5.7 标准字符实体引用集	141
4.1 XML 信息集合的概念	97	5.6 条件部分	142
4.1.1 实体引用	98	5.7 独立文档声明	145
4.1.2 基 URI	98	5.8 有效性验证 XML 解析器	146
4.2 信息项	98	5.9 DTD 的局限性	148
4.2.1 文档信息项	98	5.10 <Toysco> 例子	150
4.2.2 命名空间信息项	99	5.11 小结	153
4.2.3 元素信息项	99	第 6 章 XML Schema 介绍	155
4.2.4 属性信息项	100	6.1 为什么模式很重要	155
4.2.5 字符信息项	101	6.1.1 W3C XML Schema 的背景	156
4.2.6 注释信息项	101	6.1.2 W3C XML Schema 的优点	157
4.2.7 处理指令信息项	101	6.1.3 选择一个解析器	159
4.2.8 CDATA 开始和结束标记信息项	101	6.2 使用 XML Schema	160
4.2.9 内部实体信息项	102	6.2.1 主要组件	160
4.2.10 外部实体信息项	102	6.2.2 复杂类型	167
4.2.11 未解析实体信息项	102	6.2.3 声明属性	171
4.2.12 未展开的实体引用信息项	103	6.2.4 内容模型	172
4.2.13 实体开始和结束标记信息项	103	6.2.5 二级组件	174
4.2.14 文档类型声明信息项	103	6.2.6 雇员模式	181
4.2.15 表示法信息项	104	6.3 创建自己的数据类型	182
4.3 小结	104	6.3.1 数据类型的特征	183
第 5 章 文档类型定义	105	6.3.2 派生和刻画	185
5.1 为什么要验证 XML 的有效性	105		

6.4 ToysCo 实例	190	8.4 XPath 函数	291
6.5 小结	198	8.4.1 节点集合函数	291
第 7 章 其他 Schema	199	8.4.2 布尔函数	292
7.1 早期的 Schema 建议	199	8.4.3 数值函数	292
7.1.1 DDML	200	8.4.4 字符串函数	292
7.1.2 DT4DTD	200	8.4.5 XPath 语法的变化形式	293
7.1.3 SOX	201	8.4.6 非缩写的相对定位路径	294
7.1.4 XML-Data	202	8.4.7 非缩写的绝对定位路径	300
7.2 XML-DATA 简化	202	8.4.8 缩写的相对定位路径	301
7.2.1 XDR Schema 结构	203	8.4.9 缩写的绝对定位路径	303
7.2.2 XDR 和 XML Schema	214	8.5 XPath 是 XSLT 的基础	304
7.2.3 实例——发货单的 XDR Schema	217	8.6 XPath 是 XPointer 的基础	305
7.2.4 XDR 小结	221	8.7 展望 XPath	306
7.3 可供选择的验证工具	221	8.7.1 XPath 2.0	306
7.4 RELAX	222	8.7.2 XPath 和 XML Schema	307
7.4.1 RELAX 的结构	224	8.7.3 XPath 和正规表达式	307
7.4.2 RELAX 的元素	225	8.7.4 XPath 和 XQuery	307
7.4.3 数据类型	234	8.8 小结	308
7.4.4 实现	235	第 9 章 转换 XML	309
7.4.5 RELAX 小结	235	9.1 XSLT 的起源和目的	310
7.5 TREX	236	9.2 什么是转换	310
7.5.1 TREX 的元素	237	9.3 定位数据: XPath	312
7.5.2 数据类型	249	9.3.1 定位路径	312
7.5.3 TREX 处理程序	250	9.3.2 函数	315
7.5.4 TREX Schema	251	9.3.3 在转换中使用 XPath	316
7.5.5 实例	252	9.4 转换数据: XSLT	316
7.5.6 TREX 小结	262	9.4.1 XSLT 的元素	316
7.6 Schematron	262	9.4.2 完整列表	317
7.6.1 Schematron 的结构	263	9.4.3 常用元素	317
7.6.2 Schematron 的元素	266	9.4.4 XSLT 编程风格	329
7.6.3 Schematron 的实现	274	9.5 执行转换	332
7.6.4 Schematron 小结	276	9.5.1 在代码中转换	332
7.7 小结	276	9.5.2 Saxon	334
第 8 章 XML 导航工具——XPath	277	9.5.3 VBXML 的 XSL 测试器	334
8.1 XPath 简介	277	9.6 XSLT 用于表示: XML 转换成 HTML 的实例	334
8.1.1 XPath 及其作用	278	9.6.1 书籍目录源文档	335
8.1.2 XPath 的工作原理	281	9.6.2 输出结果	342
8.1.3 XPath 数据模型	283	9.7 XSLT 用于数据转换: XML 之间转换 的实例	342
8.2 XPath 表达式及定位路径	288		
8.3 XPath 语法	289		

9.7.1 转换	342	11.2.3 DOM 在出版过程中的使用方案	409
9.7.2 进一步处理	344	11.3 使用 DOM 和 XML 的实例应用	412
9.8 小结	344	11.3.1 一个简单的客户端实例	412
第 10 章 文档片断、XLink 和 XPointer	346	11.3.2 另一个实例(使用 Java)	417
10.1 XML 文档片断交换	346	11.3.3 DOM Level 2	422
10.1.1 文档片断的使用	350	11.3.4 对核心规范的修改	422
10.1.2 问题: Bare 文档片断并不总是充分的	352	11.3.5 视图规范	426
10.1.3 解决办法: 上下文信息	353	11.3.6 事件规范	426
10.1.4 实例	355	11.3.7 样式规范	432
10.1.5 如何传输文档片断	357	11.3.8 遍历和范围规范	432
10.1.6 小结	358	11.3.9 一个快速实例	437
10.2 链接	359	11.3.10 DOM Level 3	439
10.2.1 链接的概念	359	11.4 DOM 和 XML 的未来	440
10.2.2 W3C 的规范: XLink	361	11.5 小结	440
10.2.3 简单链接	362	第 12 章 SAX 2	442
10.2.4 扩展链接	364	12.1 概述	442
10.2.5 XLink 小结	379	12.2 对 SAX 的需求	442
10.3 XPointer	380	12.2.1 SAX 和 DOM 的对比	433
10.3.1 HTML 指针	380	12.2.2 选择哪一个	433
10.3.2 XPointer 规范	381	12.3 理解 SAX	444
10.3.3 如何指定文档片断标识符	382	12.4 设置环境	444
10.3.4 XPointer 中的错误	385	12.5 装载阅读器	446
10.3.5 小结	385	12.6 内容处理程序	448
10.4 结论	385	12.7 错误处理程序	455
第二部分 编 程 篇		12.8 DTD 处理程序	458
第 11 章 文档对象模型	387	12.9 有效性验证	458
11.1 文档对象模型的概念	387	12.10 默认处理器	459
11.1.1 XML 文档结构	388	12.11 常见缺陷	459
11.1.2 为什么使用 DOM	389	12.11.1 属性的顺序	460
11.1.3 实践中的考虑	391	12.11.2 文本数据	460
11.1.4 DOM 规范	392	12.11.3 没有预读	460
11.1.5 现实世界中的 DOM	404	12.12 实例	461
11.1.6 特殊 XML DOM 实例: HTML DOM	404	12.12.1 获得数据	461
11.2 使用 DOM	407	12.12.2 字数计数器	466
11.2.1 DOM API	407	12.12.3 使用过滤器	469
11.2.2 客户端和服务端	408	12.13 小结	477
		第 13 章 基于模式的程序设计	478
		13.1 简短的历史	479
		13.2 基于模式程序设计的目标	481
		13.2.1 代码明晰	482