

Java and XSLT



JavaTM 与 XSLT



O'REILLY®

中国电力出版社

Eric M. Burke 著

高伟 英宇 译

Java™ 与 XSLT

Eric M. Burke 著

高伟 英宇 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 XSLT/ (美) 伯克 (Burke, E. M.) 著; 高伟, 英宇译. - 北京: 中国电力出版社, 2002

书名原文: Java™ and XSLT

ISBN 7-5083-1311-9

I. J... II. ①伯... ②高... ③英... III. ① JAVA 语言 - 程序设计 ② 可扩展语言, XSLT - 程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2002) 第 099021 号

北京市版权局著作权合同登记

图字: 01-2002-3270 号

©2001 by O'Reilly & Associates, Inc.

Simplified Chinese Edition, jointly published by O'Reilly & Associates, Inc. and China Electric Power Press, 2003. Authorized translation of the English edition, 2001 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 2001。

简体中文版由中国电力出版社出版 2003。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 XSLT

书 号 / ISBN 7-5083-1311-9

责任编辑 / 陈维宁

封面设计 / Ellie Volckhausen, 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 32 印张 473 千字

版 次 / 2003 年 4 月第一版 2003 年 4 月第一次印刷

印 数 / 0001-5000 册

定 价 / 55.00 元 (册)

Java™ 与 XSLT

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

Eric M. Burke是一位具有广泛培训与指导经验的Java顾问。在致力于Java之前，他是一名C++程序员，从事各种Unix平台下的XWindow/Motif GUI开发。他现在主攻Java和XML技术，特别是XSLT、servlet、Swing和EJB。

Eric毕业于美国南伊利诺斯州大学卡本代尔分校，获计算机科学学士学位。目前他是美国密苏里州圣路易斯Object Computing公司的高级软件工程师。他每月都要将很多时间花在咨询与讲课上。他还撰写和发布很多课程，包括面向对象的原理、Java servlet、JSP和JavaBean等。Eric业余喜欢木器加工和家庭装修。他的联系方式为：*burke_e@yahoo.com*。

封面介绍

本书的封面动物是貂，也称为白鼬或者短尾鼬。貂生存于地球的北极圈与大约北纬40度之间。它们的身体，不包括尾部大概长7至12英寸，尾部一般为3至6英寸长，重量在1.5至11.5盎司之间，雄性比雌性的平均个头要大。

貂会游泳和爬树，虽然它们可以生活在各种环境中，比如草地、苔原和密林深处，但它们更喜欢生活在石林或者灌木丛地带。它们是食肉动物，主要吃一些啮齿类动物，但它们也喜欢吃小兔子和鸟、鱼、虫和蛋。（至于它们吃蛋时是煎一面还是两面至今还是一个争议。）

雌貂每年只产一窝仔。交配季节是在晚春，但授精后，貂胚停止长大，并且几个月内不会植入子宫。一旦植入子宫，怀孕只需大约一个月，并且在授精后的下一个春天生出幼貂。

貂的寿命约为8年，但如果它们生活在野外的话，则只有一年或者更少。威胁它们主要的杀手是饥饿，这取决于它们生活环境中啮齿动物的数量，但它们也经常被鹰、猫头鹰和人类所捕食。

貂皮，是我们从白鼬和短尾鼬身上得到的最大收获，尤其是貂在冬天时的皮毛。每年的冬天，貂皮通体洁白，只有尾部有一小点黑色。白色的貂皮袍子为全世界的君主所喜爱。

目录

前言	1
第一章 简介	9
Java、XSLT 和 Web	9
XML 回顾	12
超越动态网页	33
起 步	37
Web 浏览器对 XSLT 的支持	39
第二章 XSLT 第一部分 —— 基础	41
XSLT 简介	41
转换过程	50
另一个 XSLT 实例 —— 使用 XHTML	56
XPath 基础	62
循环和排序	69
输出动态属性	72

第三章 XSLT 第二部分 —— 进阶	76
条件处理	76
参数和变量	82
组合多个样式表	91
格式化文本和数值	94
模式转化	100
Ant 文档样式表	105
 第四章 基于 Java 的 Web 技术	119
传统方法	119
通用设计	130
XSLT 和 EJB	139
主要方法总结	141
 第五章 使用 Java 的 XSLT 处理	143
一个简单实例	143
JAXP 1.1 简介	150
输入与输出	162
样式表编译	186
 第六章 servlet 基础和 XSLT	193
servlet 语法	194
WAR 文件及部署	198
另一个 servlet 实例	203
再次考察样式表缓存	230
servlet 线程问题	233
 第七章 讨论论坛实例	239
全部过程	240
原型化 XML	242

动态化 XML	265
Servlet 实现	295
完成	317
第八章 其他相关技术	319
XSLT 页面布局模板	319
不使用 cookie 的会话跟踪	330
识别浏览器	335
servlet 过滤器	339
使用 XSLT 作为代码生成器	350
使用 XSLT 实现国际化支持	354
第九章 开发环境、测试及性能	372
开发环境	372
测试和调试	387
性能技术	412
第十章 无线应用程序	427
无线应用技术	427
无线应用的体系结构	429
Java、XSLT 和 WML	430
无线应用的前景	458
附录一 论坛实例代码	459
附录二 JAXP API 参考	474
附录三 XSLT 快速参考	488
词汇表	497

前言

Java 和 XSLT (Extensible Stylesheet Language Transformations, 可扩展样式表语言转换) 是两种完全不同的技术, 它们之间的关系是彼此补充而非相互竞争。Java 的优势在于可移植性、丰富的标准库集合和它已被大多数公司普遍接受这一事实。然而, Java 的弱点之一是其处理文本的能力。例如, 将 XML 文件转换为 XHTML 或 WML (Wireless Markup Language, 无线标记语言) 等格式, Java 并不是最好的技术。使用 Java 完成这项工作需要熟练掌握 DOM、SAX 或 JDOM 等 API。特别是对于网站来说, 页面的生成过程需要非常简化, 这样非程序员也能够参与网站的设计和维护。

显然, XSLT 是专为 XML 转换而设计的。利用 XSLT, XML 数据可以被转换为任意一种其他文本格式, 包括 HTML、XHTML、WML, 甚至某些意想不到的格式, 比如 Java 源代码。就复杂性和深奥程度而言, XSLT 要比 HTML 难, 但是比 Java 简单。也就是说, 网页制作人员或许能够顺利地掌握如何使用 XSLT, 但是在完成页面时, 仍将需要程序的协助。

XSLT 处理器负责解释并执行在 XSLT 中找到的指令。很多此类处理器是使用 Java 编写的。对于必须与 XML 和 XSLT 进行交互的应用程序来说, 使用 Java 是极好的选择。而应用了 XSLT 的网站, 仍然需要 Java servlet 和 EJB 来捕获客户端请求, 然后从数据库取得数据并实现业务逻辑。XSLT 可能会用于生成每一个 XHTML 网页, 但如果没有像 Java 这样的语言充当协调者, 这些是不可能做到的。

本书介绍了在 XSLT 标记语言背后最重要的概念，但并不是有关这个主题的一本全面参考。本书的重点在于它与 Java 的互操作性，特别是在 servlet 和 Web 应用上。每个概念都伴随一个应用实例，所有实例都可以在广泛可用的免费工具下正常工作。

读者群

本书的目标读者群是那些想要了解如何使用 XSLT 的 Java 程序员。读者最好具有 Java 编程经验，对 XML 的术语基本熟悉，但不是必需的。由于本书中的许多实例都围绕 Web 应用程序和 servlet，所以第四章和第六章专门介绍了这个主题，并提供一个 servlet 技术的快速教程。第二章和第三章包含一个详细的 XSLT 教程，因此不必事先具备 XSLT 的知识。

本书特别适合那些已经阅读了大量有关这些技术的书籍、但还没有在一个完整的应用程序中将它们结合使用的读者。例如，第七章提供了一个基于 Web 的论坛的从头到尾的实现过程。从第三章的一个 Ant 编译文件样式表的例子直到第八章的国际化技术，每一章都可以看到完整的应用实例。

软件和版本

紧跟最新的技术始终是一种挑战，特别在编写 XML 相关的工具时。表 P-1 中列出的工具软件足以运行本书中的每个实例。

表 P-1：软件与版本

工具软件	URL	描述
Crimson	包括在 Apache 的 JAXP 1.1 中	XML 解析器
JAXP 1.1	http://java.sun.com/xml	用于 XML 处理的 Java API
JDK 1.2.x	http://java.sun.com	任何 Java 2 标准版 SDK 都可以
JDOM beta 6	http://www.jdom.org	开放源码，DOM 之外的选择
JUnit 3.7	http://www.junit.org	开放源码，单元测试框架
Tomcat 4.0	http://jakarta.apache.org	开放源码，servlet 容器
Xalan	包括在 JAXP 1.1 中	XSLT 处理器

当然还有其他工具，特别是可以从 <http://users.iclway.co.uk/mhkay/saxon> 得到的 SAXON XSLT 处理器。但它很容易被 Xalan 取代，因为 JAXP 具有独立于厂商的功能。

所有实例，以及在表 P-1 中列举的工具 JAR 文件，都可以从 <http://www.javaxslt.com> 和 O'Reilly 的网站 <http://www.oreilly.com/catalog/javaxslt> 上下载，其中的 README 文件包含了编译和运行这些实例的说明。

本书组织结构

本书包括 10 章和 3 个附录，如下所示：

第一章，简介

对本书涵盖的技术进行了全面的介绍，并讲解了 XML、XSLT、Java 及其他 API 是如何联系在一起的。对于那些熟悉 Java 但是没有多少 XML 经验的读者，本章也回顾了 XML 的基本概念。

第二章，XSLT 第一部分 —— 基础

通过一系列小例子和叙述介绍了 XSLT 的语法。描述了如何产生 HTML 和 XHTML 输出，并讲解了 XSLT 是如何作为一种语言工作的。本章也介绍了 XPath 的语法。

第三章，XSLT 第二部分 —— 进阶

继续通过前一章给出的实例，涵盖了更高级的 XSLT 功能，例如条件逻辑、参数和变量、文本和数字格式以及生成 XML 输出等。本章以一个更复杂的实例作为结束，它产生了 Ant 编译文件的报告摘要。

第四章，基于 Java 的 Web 技术

对流行的 Web 开发技术进行对比，将每一个 Java 和 XSLT 技术进行了比较。详细讨论了模型 - 视图 - 控制器体系结构，在此基础上讨论了 XSLT 的 Web 应用与 EJB 之间的关系。

第五章，使用 Java 的 XSLT 处理

展示了如何与 Java 应用及 servlet 一起使用 XSLT 处理器。其中提到了较老的 Xalan 和 SAXON 的 API，但是主要的重心是在 Sun 的 JAXP 上。主要的实例

示范了如何使用 XSLT 和 SAX 转换非 XML 文件和数据源、如何利用缓存技术改善性能以及如何与 DOM 和 JDOM 进行交互。

第六章，servlet 基础和 XSLT

详细回顾了 Java 的 servlet 编程技术。示范了如何创建 Web 应用程序和 WAR 文件、如何在这些 Web 应用程序内部部署 XML 和 XSLT 文件，以及如何从 servlet 运行 XSLT 转换。

第七章，讨论论坛实例

从头到尾实现了一个完整的 Web 应用程序。本章设计了一个基于 Web 的论坛并使用 Java、XML 和 XSLT 技术来实现。本章还介绍了 CSS 和 XSLT 之间的关系，并将 XHTML Strict 用于所有的 Web 页。

第八章，其他相关技术

涵盖了建立在前面几章介绍的概念之上的重要的 Java 和 XSLT 编程技术，结束部分详细地讨论了 XSLT 国际化。其他主题包括 XSLT 页面设计模板、无需 cookie 的 servlet 会话跟踪、浏览器识别以及 servlet 过滤器。

第九章，开发环境、测试及性能

提供了如何选择 XML 解析器、XSLT 处理器以及其他各种 Java 工具来结合使用的实用的参考建议，展示了如何解决与 XML JAR 文件不兼容而导致的冲突、如何使用 JUnit 编写简单的单元测试以及如何编写定制的 JAXP 错误处理。本章还讨论了性能技术与 XSLT、EJB 之间的关系。

第十章，无线应用程序

揭开了无线技术世界的面纱，重点强调了无线标记语言 (WML)；展示了如何从 servlet 检测无线设备、如何编写针对这些设备的 XSLT 样式表，以及如何使用各种蜂窝电话模拟器进行测试。本章开发了一个在线电影院应用实例来强化这些概念。

附录一，论坛实例代码

包含第七章介绍的论坛实例的所有代码。

附录二，JAXP API 参考

列举并简要介绍了 1.1 版本的 JAXP API 的每个类。

附录三，XSLT 快速参考

包含了 XSLT 语言的快速参考，列举了所有 XSLT 元素及其必需的、可选的属

性以及每个元素中允许的内容。本附录还使用 W3C 的 XSLT 规范交叉引用了每个元素。

排版约定

本书使用下列排版约定。

斜体 (*italic*)

- 路径名称、文件名称和程序名称。
- 定义新术语的地方。
- Internet 网址，如域名和 URL。

等宽字体 (`constant width`)

- 出现在 Java 程序中的任何字面内容，包括关键字、数据类型、常量、方法名、变量、类名和接口名。
- 所有的 Java 代码列表。
- HTML、XML 和 XSLT 文档、标签和属性。

等宽斜体 (`constant width italic`)

- 通用占位符，指出该项将在具体的程序中被某实际值取代。

等宽黑体 (**`constant width bold`**)

- 命令行条目。
- Java 或 XML 源文件中的强调部分。

建议与评论

本书的内容都经过测试，尽管我们做了最大的努力，但错误和疏忽仍然是在所难免的。如果你发现有什么错误，或者是对将来的版本有什么建议，请通过下面的地址告诉我们：

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

本书的勘误表、样例和其他信息也可在网上找到。请访问以下网页:

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

询问技术问题或对本书的评论, 请发电子邮件到:

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

最后, 您可以在 WWW 上找到我们:

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

致谢

我要感谢我的妻子 Jennifer 在过去六个月中对于我不能陪伴她而表现出的宽容, 因为我把自己锁在地下室中研究、编写和思考。同样让我感到幸福的是两岁的小儿子 Aidan 早早地上床睡觉; 本书的很大部分都是在晚上 8:30 之后完成的!

拿出一份值得感谢的人士的名单是项艰巨的任务, 因为有如此众多的人士对本书的材料提供了意见, 只希望我不会遗漏任何人。

所有的技术审核人士都完成了大量的出色工作, 每个人都提出了独到的见解和有益的建议。正式的审核人士有 Dean Wette、Kevin Heifner、Paul Jensen、Shane Curcuru 和 Tim Brown。

我也要感谢 Weiqi Gao、Shu Zhu、Santosh Shanbhag 和 Suman Ganesh，他们对第八章有关国际化的实例提供了帮助。Dan Troesser 的一篇技术文章激发了我实现一个自己的 servlet 过滤器的灵感，Justin Michel 和 Brent Roberts 审核了我撰写的前面几章的部分内容。

我要衷心感谢两家公司。O'Reilly 在他们的主页上放置了“致信给我们”这个小链接。本书之所以问世，是因为某天我无意点击了这个链接并决定提交一个提议。虽然我的最初想法没有被接受，但是 Mike Loukides 与我在虚拟的自由讨论过程之后通过几次电子邮件交换了看法，并最终提出了编写本书的提议。我至今仍诧异于一个网站的不知名访问者会成为 O'Reilly 的作者。

我要感谢的另一家公司是 Object Computing, Inc. (OCI)，即我的雇主。他们拥有一个由非常聪明的软件工程师组成的非凡团队，他们总是热心地回答问题，提出建议并鼓舞我掌握更多内容。他们是我为 OCI 工作的原因，也是本书得以完成的原因。最后我要感谢 OCI 的 Mark Volkmann，他最先向我讲解有关 XML 的内容，并在过去的 5 年中回答了无数的问题。