

XML

完 全 探 索

I n s i d e X M L

(美) Steven Holzner 著
师夷工作室 译



中国青年出版社
CHINA YOUTH PRESS

**New
Riders**

(京)新登字 083 号

本书简体中文版由 New Riders 公司授权中国青年出版社独家出版。未经出版者书面许可, 任何单位和个人均不得以任何形式复制或传播本书的部分或全部。

Authorized translation from the English language edition, entitled Inside XML, 1st Edition, published by New Riders Publishing
Copyright © 2000

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from the Publisher.

Chinese Simplified language edition published by China Youth Press
Copyright © 2001

版权贸易合同登记号: 01-2001-0860

策 划: 胡守文
王修文
郭 光
责任编辑: 江 颖
赵 武
责任校对: 肖新民

书 名: 《XML 完全探索》
编 著: (美) Steven Holzner
出版发行: 中国青年出版社
地址: 北京市东四十二条 21 号 邮政编码: 100708
电话: (010) 84015588 传真: (010) 64053266

印 刷: 山东高唐印刷有限责任公司
开 本: 787mm × 1092mm 1/16
版 次: 2001 年 10 月北京第 1 版
印 次: 2001 年 10 月第 1 次印刷
印 数: 1-5000
书 号: ISBN 7-5006-4527-9/TP · 213
定 价: 89.00 元

译者序

在当今的 Internet 上，到处都有可扩展标记语言（Extensible Markup Language, XML）的身影。XML 的体系非常庞大，而且每时每刻都沿着不可预知的方向扩展，XML 是由万维网联盟（World Wide Web Consortium, W3C，网址 www.w3c.org）定义的一种语言，该联盟为万维网制定标准。

作者使用一本书的篇幅尽可能详细地涵盖了 XML 的所有方面，本书不仅囊括了从最基本的到最高级的语法，而且深入探讨了 XML 的各种具体应用。全书覆盖了数以百计的现实世界中的问题，其中包括：将 XML 与本地或 Web 服务器上的数据库之间的互联；在浏览器中阅读和使用 XML 文档；如何开发一个专用的基于图形的浏览器等等。

读者可以在本书中找到 XML 的正式标准，但对于该标准外的几个重要 XML 实现，本书也兼收并蓄。示例是学习 XML 最好的方法，基于这样的考虑，本书提供了丰富的已测试过的示例，读者可以直接使用这些示例。现今几种主要的浏览器对 XML 的支持越来越好，它们使用脚本语言为 XML 提供支持。在本书中将使用最流行和功能最强大的脚本语言 JavaScript，这样，在 IE 等浏览器中可以直接阅读 XML 文档。总之，对于学习和使用 XML 的读者来说，这是一本不可多得的好书。

本书由陶阳、蒿敬波、王锋、孟兆炜、徐进辉、李化、张锐、陈文彤、肖国尊、叶尚辉、潘晓辉、姜南、赵军锁、孙文明、邓波、王东霞、龚波、田丽韞、李志翻译。全书最后由陶阳、李化统稿。由于时间仓促，且译者的水平有限，在翻译过程中难免会出现一些错误，请读者批评指正。

译者

2001 年 7 月

关于作者

Steven Holzner 毕业于麻省理工学院，并在康奈尔大学获得博士学位，现同时任职于该两所大学。他曾担任《PC 杂志》的编辑。自 XML 标准发布以来，作者一直在从事 XML 书籍的撰写工作。现已发表了关于程序设计的著作达 63 部，销量达百万册。其著作被译成多达 16 国语言，其中很多成为最畅销书籍。

关于技术审校

他们在本书的写作过程中提供了宝贵的技术意见，在该书完稿后，又详细审阅了该书涉及的所有技术性内容、组织和流程。他们反馈回来的意见对本书是建设性的，保证该书能满足读者对最前沿的技术信息的需求。

Robert J. Brunner 在约翰·霍普金斯大学获得天体物理学博士学位，现为加州工学院天文学高级博士后学者。近几年一直致力于综合诸如 XML 和 Java 等各种新技术，设计一种大容量、高度分步协作的档案存储库。

Andrew J. Indovina 现于纽约州罗切斯特市，致力于电子商务领域的工作。曾与人合著《Visual Basic 6 Interactive course》、《Sams Teach Yourself Visual Basic Online in Webtime》和《Visual C++ 6.0 Unleashed》，且编著过大量书籍，范围涵盖 Java、Perl、Visual Basic、Visual C++、游戏开发以及项目管理。

引言

欢迎您阅读本书！本书希望以一册书的篇幅，尽可能以容易被接受和被理解的方式介绍 XML。XML 只是个标准而不是具体的实现，现在很多的研究项目都努力向其靠拢。在今日的 Internet 上，到处都有 XML 的身影，甚至在某些系统内部（例如在 Netscape Navigator 内部）。相对现有有关 XML 的书籍而言，本书更为全面地包含了最新 XML 的内容。

读者可以在本书中找到 XML 的正式标准，但本书对于该标准外的几个比较受欢迎的 XML 实现也兼收并蓄。

以上是本书的一个简单介绍，稍后我们将深入介绍 XML，尽可能地揭开 XML 的神秘面纱。例子是学习 XML 最好的方法，基于这样的考虑，本书提供了通过测试的丰富的实例，读者可以直接使用这些实例。

编写 XML 不是一件单调乏味的事情，相反，编写 XML 可以激发艺术性、奉献精神、激情、兴奋感、执着感，根本就不会有烦恼和挫折感。为了让读者体会到这些感觉，作者在本书中尽可能地捕捉 XML 中的这些精华。

本书的内容

本书目标是：用一本书的内容尽可能地涵盖 XML 所有方面，本书不仅囊括了从最基本的到最高级和最先进的语法，而且深入探讨了 XML 的各种具体应用。

本书覆盖了现实世界中数以百计的问题，其中包括：将 XML 与本地或 Web 服务器上数据库之间的互联；为了在现有浏览器中浏览 XML 页面所进行的格式化；在浏览器中阅读和使用 XML 文档；如何开发一个基于图形的浏览器等等。

以下列举说明一些本书中的主题（注意，每一个都包含有子主题，此处无法一一罗列）：

- 完整的 XML 语法
- 规范的 XML 文档
- 有效的 XML 文档
- 文档类型定义 (Document Type Definition, DTD)
- 名称空间
- XML 文档对象模型 (Document Object Model, DOM)
- 规范的 XML
- XML Schema 规范
- 使用 JavaScript 解析 XML
- XML 与数据绑定
- XML 与层叠样式表 (Cascading Style Sheet, CSS)
- XML 与 Java

- DOM 解析器
- SAX 解析器
- 可扩展的样式语言 (XSL) 变换
- XSL 格式化对象
- XLinks
- XPointers
- XPath
- XBase
- XHTML 1.0 和 1.1
- 资源描述框架 (Resource Description Framework, RDF)
- 频道定义格式 (Channel Definition Format, CDF)
- 矢量标记语言 (Vector Markup Language, VML)
- 无线标记语言 (Wireless Markup Language, WML)
- 用 JSP、ASP、JAVA servlet 或 Perl 编写服务器端 XML

本书从最基础的内容入手，并要求读者对 HTML 稍有了解。在本书开始部分将粗略地介绍如何创建一个 XML 文档，而后进一步说明如何检查 XML 文档的语法正确性。XML 最吸引人的特点就是它可以定义你自己的标记，例如下面这个例子中，`<DOCUMENT>`和`<GREETING>`就是标记，这个例子在第 1 章中还会进行更详细的说明。

```
<?xml version="1.0" encoding="UTF-8"?>
<DOCUMENT>
  <GREETING>
    Hello From XML
  </GREETING>
  <MESSAGE>
    Welcome to the wild and woolly world of XML.
  </MESSAGE>
</DOCUMENT>
```

由于 XML 允许定义个人的标记，因此说明这些个人标记应该遵循的语法也显得尤为重要（例如，上文给出的例子中的标记`<MESSAGE>`是否允许出现在标记`<GREETING>`中？）。XML 对此也进行了强调，有两种方法可用来定义这种语法：XML 文档类型说明 (Document Type Definition, DTD) 和 XML schema 规范。本书对这两种方法均有详细说明。

基于同样的原因，自己的标记应如何使用也取决于定义者，例如 Netscape Navigator 并不知道标记`<KILLER>`意指定义者收藏的一本书。一个标记的实际意义由定义者决定，因此如何在编程过程中处理 XML 文档是学习 XML 的一个重要部分。本书中使用了 JavaScript 和 JAVA 两种语言，并且本书在使用前会用充足的例子专门对其进行介绍，因此即便读者对这两种语言不熟悉，也不必查阅其他资料就能获得相关知识。

现今几种主要的浏览器对 XML 的支持越来越好，它们使用脚本语言为 XML 提供支持。

在本书中将使用最流行和功能最强大的脚本语言——JavaScript。通过使用 JavaScript 可以在 IE 等浏览器中直接阅读 XML 文档。

如何不依赖浏览器处理 XML 文档也是一个很重要的问题，因为有很多事情不是 JavaScript 可以解决的。现今 XML 的大部分发展都是基于 JAVA 的，并且在 Internet 上存在大量的免费资源。实际上，我们已经看到 JAVA 与 XML 的联系密切，可以使用 JAVA 读取解释 XML 文档，参见第 11 章“JAVA 和 XML DOM”。这并不一定意味着一定要成为 JAVA 专家，本书中已经介绍了需要掌握的 JAVA。同时，由于 JAVA 已经不能满足 XML 的迅速发展，我们也正在寻找使用新的更有力的工具。

我们也可以设计我们自己的文档并在现有的浏览器中直接显示，这可以使用两种方法——使用层叠样式表 (Cascading style sheet, CSS) 和可扩展样式语言 (Extensible Style Language, XSL) 来实现。使用 CSS 和 XSL，我们可以明确指定我们定义的标记——例如<PANIC>或者<BIG_AND_BOLD>或<AZURE_UNDERLINED_TEXT>——应该如何显示。在本书正文中，我们将深入讨论 XSL 的两个部分——XSL 变换和 XSL 格式化对象。

另外，本书中给出了所有 XML 规范，例如 XLinks、XBase 和 XPointers，通过使用 XPointers 我们可以以一种特殊的方式指向 XML 文档中某一特定的数据项。XML 规范是由万维网联盟 (World Wide Web Consortium, W3C) 制定的，通过对本书的学习，读者可以深入了解这些规范以获知在这些规范中规定了什么以及缺少什么。

本书结束部分将以几章的篇幅介绍一些现今在 Internet 上最流行的几种 XML 应用。XML 本质上是一种用来定义语言的语言，而现在已经有成百上千的基于 XML 的语言。其中部分已经得到了广泛的认可，本书将对其作一定程度上的讨论。

目前在 Internet 上有大量的 XML 素材，因此在本书中将会列出大量的 URI (在 XML 中，我们使用 URI 而非 URL，虽然目前实际上它们在用途上是一致的)。几乎每一章都列出了一些在线免费程序和资源。(然而，必须注意：在 Internet 上，URI 改变十分频繁，当我们寻找这些资源时，它们或许已经改变了。)

本书面向的读者

本书面向所有希望学习 XML 以及它如何使用的读者。在阅读本书以前，读者只需要对 HTML 有一定的认识即可，我们不必是 HTML 专家，但有一些关于 HTML 的知识对读者阅读本书将非常有帮助。

然而，现今的绝大多数 XML 软件都是基于 Windows 平台的，这意味着本书的大多数主题都必须依赖于 Windows 平台。第 7 章和第 8 章就 IE 对 XML 的支持作了详细说明。我们希望 XML 能够支持其他的操作系统，例如 UNIX，但迄今为止，XML 大多数都只支持 Windows。本书尽可能地为读者寻找更多的选择。不久的将来，越来越多的基于 Java 的 XML 工具会不断涌现，这些工具都是与平台无关的。

本书的层次结构

本书写作基于从基础到高级的多个层次，因为 XML 的领域很广。本书的基本原则是作

JS910/04

为 HTML 的进阶书籍。我们逐渐地从基础层次过渡到高级内容。

本书不要求读者在阅读本书前必须具有任何地编程知识（只在第 20 章“WML、ASP、JSP、Servlets 和 Perl”中涉及高级主题如 Java Server Pages 和在 XML 中使用 Perl 时才会用到编程知识）。本书中将会用到 Java 和 JavaScript，但阅读本书必须具备的关于这两门语言的知识都会有详细的介绍，并且这些知识不难掌握。

XML 的应用很广泛，本书就涉及到不少不同的软件包，文中用到的软件包都可从 Internet 上免费下载，并且书中给出了相应的网址。

本书使用的一些约定

请读者注意本书使用的几个约定。最重要的是当我们添加一些新的代码时，实际要讨论的几行都会突出显示（这个例子是用基于 XML 的 WML 语言编写的，WML 面向的是蜂窝电话和掌上电脑（PDA）的“微浏览器”（microbrowser）。

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
  <card id="Card1" title="First WML Example">
    <!--This is a comment -->
    <P>
      Greetings from WML.
    </P>
  </card>
</wml>
```

对于一些值得注意的东西或者针对讨论主题的附加信息，我们会使用特定的样式，如下所示：

将 ASCII 转换为 Unicode

如果想将 ASCII 文件转换为 Unicode 文件，可以使用 Sun Microsystem 提供的 JDK（Java Software Development Kit）工具箱中的应用程序 native2ascii。应用这个工具，使用如下命令 native2ascii file.txt file.uni 就可以完成转换过程。而且可以转换成多种其他格式，如压缩的 Unicode，UTF-8。

现在所有都准备就绪了。如果读者有什么建议，可以写信给我。本书是根据最新的 XML 标准撰写的，相信比此前的书籍会更完全和实用。读者如果有对本书改进和保持其前沿性的方法，也请与我联系。如果读者认为本书缺少某方面的内容，请告诉我，我会添加上相应的内容以保证本书处于领先地位。

目 录

第 1 章 XML 基础

1.1 标记语言	1
1.2 何为 XML?	3
1.3 XML 在浏览器中的显示	5
1.4 XML 的优点	7
1.4.1 简便的数据交换	7
1.4.2 定制标记语言	8
1.4.3 自描述数据	9
1.4.4 结构化和综合性的数据	9
1.5 规范的 XML 文档	10
1.6 有效的 XML 文档	10
1.7 解析 XML	11
1.8 XML 资源	14
1.9 XML 编辑器	15
1.10 XML 浏览器	17
1.10.1 Internet Explorer 5	17
1.10.2 Netscape Navigator 6	18
1.10.3 Jumbo	18
1.11 XML 解析器	19
1.12 XML 有效性分析程序	20
1.13 CSS 与 XSL	22
1.14 XLink 与 XPointer	23
1.15 URL 和 URI	24
1.16 ASCII、Unicode 和通用 字符系统	24
1.17 XML 应用	25
1.17.1 XML 应用：化学标记 语言	26
1.17.2 XML 应用：数学标记	

语言	27
1.17.3 XML 应用：频道定义 格式	28
1.17.4 XML 应用：同步多媒 体合成语言	29
1.17.5 XML 应用：HTML+ TIME	30
1.17.6 XML 应用：XHTML	31
1.17.7 XML 应用：开放软件 描述	33
1.17.8 XML 应用：可增量矢 量图形	34
1.17.9 XML 应用：矢量标记 语言	34
1.17.10 XML 应用：基于 XML 的用户界面语言	35
1.17.11 XML 应用：可扩展商 业报告语言	36
1.17.12 XML 应用：资源描述 框架	37

第 2 章 创建规范的 XML 文档

2.1 万维网联盟	39
2.2 规范的 XML 文档	40
2.3 标记和字符数据	45
空白	46
2.4 Prolog	47
2.5 XML 声明	48
2.6 注释	48
2.7 操作指令	50
2.8 标记和元素	50

2.8.1 标记名.....	51	3.2.2 ANY.....	83
2.8.2 空元素.....	52	3.2.3 子元素列表.....	83
2.9 根元素.....	52	3.2.4 #PCDATA.....	84
2.10 属性.....	53	3.2.5 处理多个子元素.....	86
2.10.1 属性名.....	56	3.2.6 DTD 序列.....	87
2.10.2 属性值.....	56	3.2.7 创建子序列.....	90
2.10.3 属性 xml:lang.....	56	3.2.8 选择项.....	92
2.11 构建规范的文档结构.....	57	3.2.9 DTD 注释.....	97
2.11.1 文档以 XML 声明开始.....	57	3.3 DTD 范例.....	99
2.11.2 包含至少一个元素.....	58	3.4 外部 DTD.....	103
2.11.3 在非空元素中包含开始 和结束标记.....	58	3.5 在 DTD 中使用 URL.....	104
2.11.4 空元素以 “/” 结束.....	59	3.6 公用 DTD.....	105
2.11.5 根元素必须包含其他所 有元素.....	59	3.7 同时使用外部和内部 DTD.....	106
2.11.6 正确嵌套元素.....	60	3.8 名称空间与 DTD.....	108
2.11.7 使用惟一的属性名.....	60	3.9 检验 DTD 的有效性.....	113
2.11.8 五个已定义的实体引用.....	61		
2.11.9 用引号包含属性值.....	61	第 4 章 DTD: 实体与属性	
2.11.10 用 “<” 和 “&” 分别 作为标记和实体的开 头.....	62	4.1 实体.....	117
2.12 CDATA 段.....	64	4.2 属性.....	120
2.13 XML 名称空间.....	66	4.3 创建内部普通实体.....	122
2.13.1 创建一个名称空间.....	66	4.4 创建外部普通实体.....	124
2.13.2 创建本地名称空间.....	68	4.5 用多个文档片断构建完整文 档.....	126
2.13.3 缺省名称空间.....	69	4.6 预定义通用实体引用.....	128
2.14 信息集 (Infosets).....	71	4.7 创建内部参数实体.....	129
2.15 规范的 XML.....	73	4.8 外部参数实体.....	131
第 3 章 有效的 XML 文档: 创建 文档类型定义		4.9 使用 INCLUDE 和 IGNORE.....	134
3.1 创建文档类型声明.....	80	4.10 关于属性.....	136
3.2 创建文档类型定义.....	81	4.10.1 在 DTD 中声明属性.....	137
3.2.1 元素声明.....	82	4.10.2 为属性设定缺省值.....	141
		4.10.3 属性类型.....	146
		4.11 在文档中嵌入非 XML 数 据.....	160
		4.12 在文档中嵌入多个未析实 体.....	162

第5章 创建 XML 模式

- 5.1 Internet Explorer 中的 XML 模式 165
- 5.2 W3C 的 XML 模式 169
- 5.3 声明类型和元素 172
 - 5.3.1 指定元素出现的次数 176
 - 5.3.2 指定元素的默认值 176
- 5.4 指定属性约束和缺省值 177
- 5.5 创建简单类型 178
 - 5.5.1 使用平面创建简单类型 179
 - 5.5.2 使用匿名类型定义 184
- 5.6 创建空元素 185
- 5.7 创建混合内容元素 186
- 5.8 模式注释 188
- 5.9 创建选择 190
- 5.10 创建序列 191
- 5.11 创建属性组 192
- 5.12 创建全组 194
- 5.13 模式和名称空间 194
 - 5.13.1 使用无验证的局部 195
 - 5.13.2 使用验证的局部 197

第6章 熟悉 JavaScript

- 6.1 什么是 JavaScript 201
 - 6.1.1 Netscape 的 JavaScript 205
 - 6.1.2 Microsoft 的 JScript 205
 - 6.1.3 ECMAScript 205
- 6.2 面向对象的 JavaScript 206
 - 6.2.1 使用 JavaScript 对象的属性和方法 207
 - 6.2.2 JavaScript 事件 208
- 6.3 JavaScript 编程 210
 - 6.3.1 使用 JavaScript 处理数据 211
 - 6.3.2 JavaScript 注释 213
 - 6.3.3 使用 JavaScript 操作符 213

- 6.3.4 创建 JavaScript if 语句 216
- 6.3.5 创建 JavaScript if...else 语句 218
- 6.3.6 创建 switch 语句 219
- 6.3.7 创建 JavaScript for 循环语句 220
- 6.3.8 创建 while 循环语句 222
- 6.3.9 创建 do...while 循环 223
- 6.3.10 创建 JavaScript 函数 223
- 6.3.11 将数值传递给函数 225
- 6.3.12 创建 JavaScript 对象 227
- 6.3.13 使用 JavaScript String 对象 228
- 6.3.14 使用 Array 类创建数组 231
- 6.3.15 使用事件 234
- 6.3.16 创建事件信息 243
- 6.3.17 处理鼠标事件 245

第7章 使用 JavaScript 处理 XML 文档

- 7.1 W3C DOM 249
 - 7.1.1 XML DOM 对象 251
 - 7.1.2 DOMDocument 对象 252
 - 7.1.3 XMLDOMNode 对象 255
 - 7.1.4 XMLDOMNodeList 对象 257
 - 7.1.5 XMLDOMNamedNodeMap 对象 257
 - 7.1.6 XMLDOMParseError 对象 258
 - 7.1.7 XMLDOMAttribute 对象 258
 - 7.1.8 XMLDOMElement 对象 260
 - 7.1.9 XMLDOMText 对象 261
- 7.2 载入 XML 文档 263
- 7.3 按名字获取元素 271
- 7.4 从 XML 元素中获取属性值 273
- 7.5 在程序中解析 XML 文档 276

7.5.1 解析 XML 文档并显示结 点类型和内容.....	280	9.3.6 创建边框.....	354
7.5.2 解析 XML 文档并显示属 性值.....	283	9.3.7 显示图像.....	356
7.6 载入文档时处理事件.....	286	9.3.8 绝对定位.....	358
7.7 在 Internet Explorer 中验证 XML 文档.....	288	9.3.9 相对定位.....	359
7.8 为 XML 元素编写脚本.....	291	9.4 正式的样式属性规范.....	361
7.9 在 Internet Explorer 中编辑 XML 文档.....	292	9.4.1 文本属性.....	363
第 8 章 XML 和数据绑定		9.4.2 字体属性.....	365
8.1 Internet Explorer 中的数据 绑定.....	297	9.4.3 背景和颜色属性.....	367
8.2 使用数据源对象.....	297	9.4.4 表格属性.....	369
8.2.1 数据和 HTML 元素绑定.....	299	9.4.5 定位和块属性.....	370
8.2.2 XML 数据绑定.....	305	9.4.6 方框属性.....	372
8.2.3 表格数据绑定和 XML.....	312	9.4.7 视觉效果属性.....	376
8.3 XML 和分层数据.....	319	9.4.8 列表属性.....	377
8.4 搜索 XML 数据.....	328	第 10 章 了解 Java	
第 9 章 层叠样式表		10.1 Java 资源.....	379
9.1 将样式表连接到 XML 文档.....	337	10.2 编写 Java 程序.....	380
9.2 在样式表规则中选取元素.....	338	10.2.1 Java 是完全面向对象 的.....	382
9.2.1 选择符中的分组元素.....	339	10.2.2 取得 Java SDK.....	383
9.2.2 通过 ID 选取.....	343	10.3 创建 Java 文件.....	384
9.2.3 使用上下文相关选择符.....	344	10.3.1 编写代码: 创建一个 应用程序.....	384
9.2.4 使用内联样式.....	345	10.3.2 编译代码.....	387
9.2.5 使用继承性.....	346	10.3.3 运行 Java 应用程序.....	388
9.2.6 理解层叠的含义.....	347	10.3.4 注释代码.....	389
9.3 创建样式规则.....	348	10.3.5 引入 Java 包和类.....	389
9.3.1 创建块元素.....	348	10.4 在 Java 中创建变量.....	391
9.3.2 设置文本样式.....	349	10.5 在 Java 中创建数组.....	394
9.3.3 设置颜色和背景.....	350	10.6 在 Java 中创建字符串.....	397
9.3.4 边缘、缩进和对齐方式.....	353	10.7 Java 操作符.....	398
9.3.5 将样式应用于列表.....	354	10.8 Java 条件语句: if、if... else、switch.....	399
		10.9 Java 循环: for、while、 do...while.....	402

10.10 声明和创建对象.....	405	12.2.6 处理可忽略空白.....	493
10.10.1 重载构造器.....	405	12.2.7 处理元素结束.....	495
10.10.2 给对象赋值.....	406	12.2.8 处理错误和警告.....	495
10.11 在 Java 中创建方法.....	407	12.3 过滤 XML 文档.....	499
10.12 创建 Java 类.....	409	12.4 创建窗口化的浏览器.....	504
10.12.1 创建新类.....	410	12.5 创建图形浏览器.....	508
10.12.2 创建一个构造器.....	411	12.6 在 XML 文档浏览.....	512
10.12.3 使用图形对象.....	412	12.7 修改 XML 文档.....	516
10.12.4 关闭应用程序窗口.....	416		
第 11 章 Java 和 XML DOM		第 13 章 XSL 变换	
11.1 获得 XML Java 包.....	420	13.1 在 XML 文档中应用 XSLT	
11.2 设置 CLASSPATH.....	421	样式表.....	521
11.3 创建解析器.....	421	13.1.1 XSL 样式表.....	522
11.4 显示一个完整的文档.....	436	13.1.2 进行 XML 变换.....	522
11.4.1 处理文档结点.....	438	13.1.3 James Clark 的 XT.....	523
11.4.2 处理元素结点.....	438	13.1.4 XML for Java.....	523
11.4.3 处理属性.....	439	13.1.5 用浏览器变换 XML 文	
11.4.4 处理 CDATA 段结点.....	441	档.....	525
11.4.5 处理文本结点.....	441	13.2 创建 XSLT 样式表.....	526
11.4.6 处理操作指令结点.....	442	13.2.1 xsl:apply-templates 元素.....	528
11.4.7 结束元素标记.....	443	13.2.2 使用 xsl:value-of 得到结	
11.5 过滤 XML 文档.....	446	点值.....	530
11.6 创建窗口化的浏览器.....	450	13.2.3 使用 xsl:for-each 处理多	
11.7 创建图形浏览器.....	455	个选择.....	531
11.8 在 XML 文档中导航.....	460	13.2.4 为 match 属性指定匹配	
11.9 修改 XML 文档.....	463	模式.....	532
第 12 章 Java 和 SAX		13.2.5 为 select 属性指定模式.....	540
12.1 应用 SAX.....	472	13.2.6 理解 XPath.....	540
12.2 显示一个完整的文档.....	486	13.2.7 缺省 XSLT 规则.....	550
12.2.1 处理文档的开始.....	487	13.3 基于输入改变文档结构.....	551
12.2.2 处理操作指令.....	488	13.3.1 创建属性模板.....	551
12.2.3 处理元素的开始.....	489	13.3.2 创建新元素.....	553
12.2.4 处理属性.....	490	13.3.3 创建新属性.....	554
12.2.5 处理文本.....	492	13.4 用 xsl:comment 生成注释.....	555
		13.5 用 xsl:text 生成文本.....	556

13.6	拷贝结点	557
13.7	排列元素	558
13.8	使用 xsl:if	560
13.9	使用 xsl:choose	561
13.10	控制输出类型	562

第 14 章 XSL 格式化对象

14.1	格式化 XML 文档	565
14.2	创建 XSLT 样式表	566
14.3	将文档转换成格式化对象形式	568
14.4	创建格式化的文档	570
14.5	XSL 格式化对象	572
14.5.1	XSL 格式化属性	574
14.5.2	处理格式化对象	577

第 15 章 XLink 和 XPointer

15.1	概述：使用 XLink 和 XPointer 链接	605
15.2	关于 XLink	607
15.2.1	XML Base (XBase) 规范	610
15.2.2	声明 XLink 属性	611
15.2.3	xlink:type 属性	612
15.2.4	使用 xlink:href 定位资源	612
15.2.5	描述资源：xlink:role 和 xlink:title	613
15.2.6	xlink:show 属性	613
15.2.7	xlink:actuate 属性	616
15.2.8	扩展链接	617
15.2.9	使用 xlink:from 和 xlink:to 属性创建弧	620
15.2.10	内联和外联链接	624
15.3	关于 XPointer	629
15.3.1	XPointer 轴	630

15.3.2	XPointer 结点测试	630
15.3.3	XPointer 谓词断言	631
15.3.4	XPointer 位置单元集函数	631
15.3.5	使用 XPointer 点	634
15.3.6	使用 XPointer 范围	635
15.3.7	使用字符串范围	636
15.3.8	XPointer 缩写	637

第 16 章 XHTML 基础

16.1	XHTML 版本	644
16.1.1	XHTML 版本 1.0	644
16.1.2	XHTML 版本 1.1	645
16.1.3	XHTML 基础版本	646
16.2	XHTML 核对表	646
16.2.1	XHTML 与 HTML	646
16.2.2	从 HTML 到 XHTML 的自动转换	647
16.2.3	XHTML 文档的合法化检查	649
16.3	XHTML 编程	650
16.3.1	文档元素 (<html>)	650
16.3.2	创建 Web 页题头 (<head>)	651
16.3.3	文档标题 (<title>)	652
16.3.4	文档主体 (<body>)	653
16.3.5	注释 (<!-->)	656
16.3.6	标题 (<h1>到<h6>)	656
16.3.7	处理文本	658

第 17 章 XHTML 的应用

17.1	显示图像 ()	683
17.2	创建超链或锚 (<a>)	685
17.3	设置链接信息 (<link>)	687
17.4	创建表格 (<table>)	689

17.4.1 创建表格行 (<tr>)	690	18.3.4 通过参照关系引用资源 ...	729
17.4.2 创建表头 (<th>)	692	18.4 在属性元素中使用 XML	729
17.4.3 创建表格数据 (<td>)	693	18.5 使用简短的 RDF 语法	730
17.5 创建文档框架 (<frameset>)...	695	18.6 RDF 容器	731
17.6 在 XHTML 中使用样式表	699	18.6.1 使用 Bag 容器	732
17.6.1 用 XHTML 创建嵌入式 样式表 (<style>)	700	18.6.2 使用 Seq 容器	733
17.6.2 在 XHTML 中使用内联 样式	701	18.6.3 使用 Alt 容器	733
17.7 使用脚本编程 (<script>)	702	18.6.4 生成关于容器的语句	734
17.8 创建 XHTML 表单 (<form>)	705	18.6.5 生成关于容器中的项目 的语句	735
17.8.1 创建控件 (<input type ="button">)	707	18.6.6 通过前缀选择容器项目 ...	735
17.8.2 创建文本域 (<input type="text">)	708	18.7 创建 RDF 模式	736
17.9 扩展的 XHTML 1.0	710	18.8 CDF 概览	736
17.10 关于 XHTML 1.1 模块	712	18.9 CDF 语法	739
17.10.1 XHTML 1.1 摘要模块 ...	713	18.10 创建 CDF 文件	740
17.10.2 XHTML 1.1 模块工具 集	715	18.10.1 添加<ITEM>子元素	741
17.10.3 扩展 XHTML 1.1	717	18.10.2 连接一个 CDF 文件与 一个 Web 页	741
第 18 章 资源描述框架 (RDF) 和频 道定义格式 (CDF)		18.10.3 描述一个频道	742
18.1 RDF 概览	719	18.10.4 安排动态更新	745
18.2 RDF 语法	723	18.10.5 关掉预缓冲	747
18.2.1 RDF 根元素	723	18.10.6 在分级目录中下载 网页	747
18.2.2 RDF 名称空间	723	18.10.7 记载用户访问	748
18.2.3 RDF 描述元素	724	18.11 设置频道的基本 URI	749
18.2.4 RDF 属性元素	725	18.12 设置最近修改日期	750
18.3 Dublin Core	725	18.13 设置频道用法	751
18.3.1 描述多属性	727	18.13.1 更新桌面成分	751
18.3.2 描述多资源	728	18.13.2 通过电子邮件更新	752
18.3.3 嵌套资源	728	18.13.3 更新屏保	752
		18.13.4 更新软件	753
		第 19 章 矢量标记语言 VML	
		19.1 创建 VML 文档	756
		19.2 VML 元素	758

19.2.1	共有属性	759
19.2.2	VML 使用 CSS	759
19.3	<shape>元素	760
19.4	使用预定义图形	764
19.4.1	<rect>元素	764
19.4.2	<roundrect>元素	766
19.4.3	<line>元素	767
19.4.4	<oval>元素	767
19.4.5	<polyline>元素	770
19.4.6	<curve>元素	771
19.4.7	<arc>元素	773
19.5	染色	774
19.6	缩放	775
19.7	定位	775
19.7.1	静态位置样式	775
19.7.2	相对位置样式	776
19.7.3	绝对位置样式	777
19.7.4	z-index 位置样式	778
19.7.5	旋转位置样式	780
19.7.6	翻转位置样式	781
19.7.7	<group>元素	782
19.8	<shadow>元素	784
19.9	<fill>元素	786
19.9.1	创建梯度填充	788
19.9.2	创建图案填充	791
19.9.3	创建图片填充	792

19.10	使用<shapetype>元素	793
-------	-----------------------	-----

19.11	更高级的 VML	797
-------	----------------	-----

第 20 章 WML、ASP、JSP、Servlets 和 Perl

20.1	XML 和 ASP	800
------	-----------------	-----

20.2	XML 和 Java Servlet	802
------	--------------------------	-----

20.3	Java Server Pages	806
------	-------------------------	-----

20.4	XML 和 Perl	808
------	------------------	-----

20.5	无线标记语言	819
------	--------------	-----

20.5.1	WML 的开端	821
--------	---------------	-----

20.5.2	设置文本对齐方式	823
--------	----------------	-----

20.5.3	基本文本样式	824
--------	--------------	-----

20.5.4	按钮	824
--------	----------	-----

20.5.5	后退按钮	826
--------	------------	-----

20.5.6	超链	827
--------	----------	-----

20.5.7	表格	828
--------	----------	-----

20.5.8	文本输入	829
--------	------------	-----

20.5.9	选择元素	830
--------	------------	-----

20.5.10	定时器	833
---------	-----------	-----

20.5.11	连接服务器	834
---------	-------------	-----

20.5.12	图像	835
---------	----------	-----

附录 XML 1.0 说明书

摘要	838
----------	-----

本文档的状态	838
--------------	-----

可扩展标记语言 (XML) 1.0	839
-------------------------	-----

第 1 章 XML 基础

欢迎进入可扩展标记语言（Extensible Markup Language, XML）的世界。本书是您正确的选择，它将引导您畅游这个世界。XML 的体系非常庞大，而且每时每刻都沿着不可预知的方向扩展，但通过本书，读者可以详细地了解它。本书的覆盖面很广，因为 XML 正沿着一些最令人振奋的方向发展。

XML 是由万维网联盟（World Wide Web Consortium, W3C，网址 www.w3c.org）定义的一种语言，该联盟为互联网制定标准。第 1 章概述 XML 语言，同时介绍如何使用它。例如，如前文所述，XML 可以定制你自己的标记语言，这样 XML 就超越了诸如超文本标记语言（Hypertext Markup Language, HTML）等标记语言。在 HTML 中，所有的元素都是预先定义好的，这些元素远远不敷使用。本质上来说，XML 是一种元标记语言，因为通过它可以创建自己的标记语言。

1.1 标记语言

标记语言（Markup Language, ML）用来描述文档的格式，也就是描述文档内容如何显示。最为人们所熟知的标记语言当然是 HTML，它用来写标准的 web 页面。下面是一个 HTML 页面的示例：

```
<HTML>
  <HEAD>
    <TITLE>Hello From HTML</TITLE>
  </HEAD>
  <BODY>
    <CENTER>
      <H1>
        Hello From HTML
      </H1>
    </CENTER>
    Welcome to the wild and woolly world of HTML.
  </BODY>
</HTML>
```

在 Netscape Navigator 中浏览该例如图 1.1 所示。注意，该页面中的 HTML 标记如 <CENTER>、<H1>等是用来指导浏览器的。这正是标记的作用，指导浏览器如何显示文档内容。

HTML 与 XML 有一定的亲缘关系，两者都是脱胎于标准通用标记语言（Standard Generalized Markup Language, SGML）。从字面上就可以看出 SGML 是一种非常通用的标