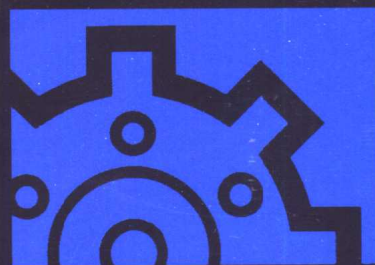


que

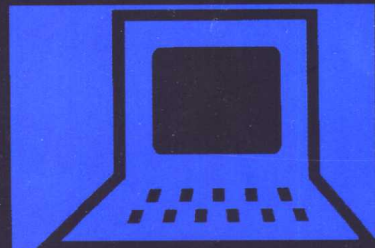
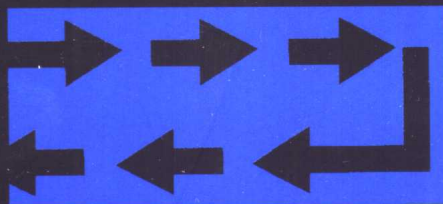
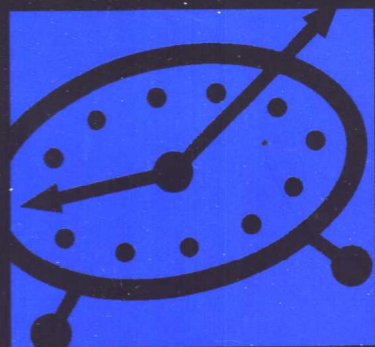


New Edition—Covers Final
XML Schemas
Recommendation

XML

示例程序导学

第二版



Benoit Marchal 著
杨洪涛 顾雷
郑震坤 译



清华大学出版社

XML 示例程序导学

(第二版)

Benoît Marchal 著

杨洪涛 顾雷 郑震坤 译

清华大学出版社

(京)新登字 158 号

著作权合同登记号: 01-2002-3072

内 容 简 介

本书采用概念与实例紧密结合的方法,通过对示例代码的解析,使读者在编程过程中体会 XML 强大的 Web 开发功能。

全书共分 12 章。前 11 章对 XML 与 HTML 的关系与差异、名称空间、DTD、模式、样式表、解析器、XSL、DOM、SAX、XPath、XPointer、SOAP 等做了全面详尽的阐述,最后一章综合讨论了一个完整的电子商务实例。附录 A “Java 速成”提供了理解本书示例所需的 Java 知识。跟随书中的精彩实例,您可以循序渐进、轻松掌握功能强大的 XML。

本书适用于广大 Web 开发人员,任何具有 HTML 编程基础的读者都完全可以理解本书的内容。本书也适用于从事电子商务的读者。

XML by Example, Second Edition

Copyright © 2002 by Que

All rights reserved. No part of this book shall 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. This edition is authorized for sale only in the People's Republic of China (excluding the special Administrative Region of Hong Kong and Macau).

本书中文简体字版由美国 Que 出版社授权清华大学出版社和北京科海培训中心独家出版。未经出版者书面允许不得以任何方式复制或抄袭本书内容。

版权所有, 盗版必究。

本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

书 名: XML 示例程序导学(第二版)

作 者: Benoît Marchal

译 者: 杨洪涛 顾 雷 郑震坤

出 版 者: 清华大学出版社(北京清华大学学研大厦, 邮编 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 郑雪梅

印 刷 者: 北京朝阳科普印刷厂

发 行 者: 新华书店总店北京科技发行所

开 本: 787×1092 1/16 印张: 26.875 字数: 654 千字

版 次: 2002 年 9 月第 1 版 2002 年 9 月第 1 次印刷

印 数: 0001~5000

书 号: ISBN 7-302-05812-1/TP·3436

定 价: 42.00 元

前 言

“示例程序导学”系列

“示例程序导学”系列如何使你成为一名优秀的程序员？“示例程序导学”系列使用最佳方法来讲解如何编写程序。在介绍每个概念之后，你都将见到一个或几个使用该概念的实例。本书就像你身边的一位老师，指导你如何使用刚刚学习的概念。本书使用了大量的实例，虽然你还不熟悉本书的知识，但你将看到一个又一个的实例演示如何使用刚刚学习的知识。

“示例程序导学”系列的理念十分简单：讲解计算机编程的最佳方式是使用大量的实例。命令描述、格式化语法以及语言参考都不足以使一位新手掌握一门编程语言。只有通过大量实例，立即使用新学到的命令，并运行示例程序，学习者才能够真正掌握它们。

本书的读者对象

《XML示例程序导学》要求读者具有基本的HTML编程经验，但不要求你是这方面的专家。如果你可以编写一个简单的HTML页面，了解主要的标记（例如<p>，<title>，<h1>），那么你就完全可以理解本书的内容。

然而，只有与脚本或编程语言结合起来，XML才能够真正发挥它的价值。本书从第5章“XSL转换”开始，到第7章“解析器和DOM”及后续章节中介绍的一些高级技术都需要读者熟悉脚本编程和JavaScript。为了理解这些章节，你需要了解循环、变量、函数和对象的概念。记住，这些都是高级技术，因此即使你不是十分精通JavaScript，也仍然能够领会本书中讲解的很多有用的技术。

如果你具备下面的条件之一，那么本书正是为你准备的：

- 你精通HTML，有一定的编程经验，并希望在Internet发布方面更上一层楼。
- 你需要通过Web、CD-ROM、纸张或这些媒介的组合来发布大型的、动态的文档，而且还听说过使用XML可以大大简化你的发布工作。
- 你是一位Web开发人员，精通Java、JavaScript或其他的程序语言，你曾听别人说过XML十分简单但却具有十分强大的功能。
- 你从事电子商务或EDI方面的工作，希望了解XML在你的专业领域提供了哪些功能。
- 你使用Microsoft、IBM、Oracle、Corel、Sun或其他很多公司的带有XML功能的软件产品，需要了解如何更好地使用它们。

MJS 8/1/98

本书的组织方式

本书将教会你使用XML——可扩展标记语言。XML是一种新的标记语言，开发它是为了克服HTML的局限性。

XML之所以会出现，要归功于HTML的成功。因此，XML具有HTML的很多优秀特性。HTML越来越难以满足新的需求也是导致XML出现的原因之一。因此，XML在很多方面打破了HTML的常规。

本书提供了一个简明实用的学习XML的方法。通过真正的实例来介绍XML的思想和概念，以便你在学习这些概念的同时理解它们的应用。通过这些实例，你马上就可以看到XML所带来的好处以及所付出的代价。

你将会看到，XML有两种类型的应用：发布和数据交换(也称为应用程序集成)。数据交换应用包括大多数的电子商务应用。本书中的大部分例子都来源于数据交换应用，因为它们是XML最流行的应用。本书也给出了一个完整的Web站点发布的实例。

本书假设你熟悉Web，至少可以阅读、理解和编写基本的HTML页面，以及能够阅读和理解简单的JavaScript应用程序。学习XML并不需要你是HTML的专家，也不需要你精通JavaScript，但是需要你对它们有基本的了解。

本书中的大部分代码都是基于XML和XML样式表的。在需要编写程序时，尽可能使用JavaScript。然而，JavaScript不适用于本书最后的一个例子，因此在那里改用Java。

本书不需要你熟悉Java。然而，因为我们还是涉及到了很小一部分Java(再次指出，最后的实例中的大部分代码是XML)，附录A“Java速成”将教会你理解本书中的例子所需的Java知识。

本书的约定



实例使用图标标出：**EXAMPLE**

程序清单和代码使用等宽字体，例如：

```
<?xml version="1.0"?>
```

注意：

每章都有“注意”部分作为补充的阅读材料。这些内容用于阐明一些概念和过程。

提示：

本书中大量的“提示”提供了常见问题的解决方案和操作捷径。

警告:

警告你在编写XML时可能出现的错误。仔细阅读“警告”部分有助于节省你的时间，减少你的麻烦。

下一步

开发XML的目的是为了克服HTML的局限性。尽管在一定时期内这两种技术可能仍然会共存，但XML的重要性将会逐渐增加。重要的是，你需要了解XML的长处和局限性，以便你可以准备好应对技术的演变。

访问下面的Web站点，下载示例代码或与本书有关的附加材料：

http://www.quepublishing.com/series/by_example/

<http://www.marchal.com>

推荐你订阅以上Web站点上的免费新闻月刊，它是广泛了解XML最新动态和Internet开发的一个最佳专栏。

现在，翻到下一页，从今天开始，通过实例学习XML！

目 录

第1章 XML概述	1
1.1 简介	1
1.2 本书的适用范围	3
1.3 XML初步	3
1.3.1 没有预定义的标记	4
1.3.2 更严格的语法	4
1.4 文档结构初步	5
1.5 标记语言的历史	8
1.5.1 标记	9
1.5.2 过程标记	9
1.5.3 通用编码	10
1.5.4 标准通用标记语言	12
1.5.5 超文本标记语言	13
1.5.6 可扩展标记语言	18
1.6 XML的应用	20
1.6.1 文档应用	20
1.6.2 数据应用	21
1.7 伙伴标准	24
1.7.1 XML名称空间	24
1.7.2 样式表	25
1.7.3 DOM和SAX	26
1.7.4 XLink和XPointer	27
1.8 XML软件	27
1.8.1 XML浏览器	27
1.8.2 XML编辑器	28
1.8.3 XML解析器	28
1.8.4 XSL处理器	29
1.9 下一步	29
第2章 XML语法	30
2.1 XML基础语法	30
2.1.1 XML标记初步	30
2.1.2 元素的起始和结束标记	32

2.1.3 XML中的名称.....	33
2.1.4 属性	34
2.1.5 空元素	35
2.1.6 元素嵌套	35
2.1.7 根	37
2.1.8 XML声明	37
2.2 高级主题	38
2.2.1 注释	39
2.2.2 Unicode.....	39
2.2.3 实体	40
2.2.4 特殊属性	42
2.2.5 处理指令	42
2.2.6 CDATA节.....	43
2.3 XML的常见问题	43
2.3.1 代码缩进	44
2.3.2 为什么要使用结束标记.....	44
2.3.3 XML和语义.....	46
2.4 4 种常见错误	47
2.4.1 忘记结束标记	47
2.4.2 忘记XML是区分大小写的.....	48
2.4.3 在元素名称中引入空格.....	48
2.4.4 忘记属性值使用引号.....	48
2.5 XML的两种应用	48
2.5.1 发布	49
2.5.2 电子商务	50
2.6 XML编辑器	52
2.7 下一步	54
第3章 XML名称空间.....	55
3.1 名称空间所解决的问题	55
3.2 名称空间	59
3.2.1 名称空间的名称	60
3.3 URI	61
3.3.1 URI的需求	61
3.3.2 URL和URN	62
3.3.3 名称是什么	63
3.3.4 注册域名	64
3.4 作用域	65
3.5 数字签名：一个名称空间实例	66

3.6 下一步	69
第4章 XML模型	70
4.1 DTD和XML模式	70
4.2 DTD语法	71
4.2.1 元素声明	73
4.2.2 加号、星号和问号的秘密	73
4.2.3 逗号和竖线的秘密	73
4.2.4 关键字	74
4.2.5 混合内容	74
4.2.6 无二义模型	74
4.2.7 属性	75
4.3 DTD和文档的关系	76
4.3.1 验证文档	77
4.3.2 为什么使用DTD	78
4.4 DTD的高级概念	78
4.4.1 内部子集和外部子集	78
4.4.2 公共标识符	79
4.4.3 独立文档	80
4.4.4 实体	80
4.4.5 常规实体和参数实体	80
4.4.6 内部实体和外部实体	81
4.4.7 已解析实体、未解析实体和符号	82
4.4.8 条件节	83
4.5 模式语法	83
4.5.1 简单类型定义	86
4.5.2 复杂类型定义	87
4.5.3 简单内容和复杂内容	88
4.5.4 混合内容	89
4.5.5 全局元素声明	89
4.6 名称空间和模式的其他高级概念	90
4.6.1 对全局元素的引用	93
4.6.2 名称空间和文档	94
4.6.3 批注	95
4.6.4 深入全局和本地元素	96
4.6.5 名称组	97
4.7 XML文档建模	98
4.7.1 尽量避免建模	98
4.8 使用对象模型对文档建模	99

4.8.1 元素和属性	105
4.8.2 DTD和模式	106
4.9 建模：从零开始	106
4.9.1 灵活性	107
4.9.2 对XML文档建模	108
4.9.3 元素命名	110
4.10 辅助工具	111
4.11 下一步	112
第5章 XSL转换	113
5.1 为什么需要样式	113
5.1.1 CSS	113
5.1.2 XSL	114
5.2 XSL	114
5.2.1 Xalan	114
5.2.2 XSLT的概念	114
5.3 XSLT的基础知识	115
5.3.1 在浏览器中查看XML	116
5.3.2 一个简单的样式表	118
5.3.3 stylesheet元素	122
5.3.4 output元素	123
5.3.5 模板元素	123
5.3.6 XPath	123
5.3.7 属性匹配	124
5.3.8 条件匹配	125
5.3.9 匹配文本和函数	125
5.3.10 处理器操作	125
5.4 支持不同的标记语言	128
5.4.1 WML转换	130
5.5 何时何处使用样式表	132
5.5.1 特殊提示——Internet Explorer 5.0	133
5.5.2 修改样式表	137
5.6 高级XSLT	137
5.6.1 indent参数	140
5.6.2 使用参数	140
5.6.3 在输出文档中创建特殊节点	141
5.6.4 创建目录	143
5.6.5 优先级	145
5.7 下一步	145

第6章 XSL格式化对象和层叠样式表	146
6.1 不使用HTML显示XML	146
6.2 CSS和FO基础	148
6.3 简单的CSS	149
6.3.1 注释	152
6.3.2 选择符	152
6.3.3 CSS属性	153
6.4 简单的FO	153
6.4.1 FO元素	154
6.4.2 应用样式表	156
6.4.3 FO样式表概述	159
6.5 流对象和区域	161
6.5.1 流对象	161
6.5.2 属性继承	161
6.5.3 矩形区域或方框	162
6.6 属性取值	164
6.6.1 长度	164
6.6.2 百分比	165
6.6.3 颜色	165
6.7 方框属性	166
6.7.1 边距属性	166
6.7.2 补白属性	166
6.7.3 空白属性	167
6.7.4 边框样式属性	167
6.7.5 边框宽度属性	167
6.8 文本和字体属性	167
6.8.1 字体名称	168
6.8.2 字体大小	168
6.8.3 字体样式和粗细	168
6.8.4 文本修饰	169
6.8.5 文本对齐	169
6.8.6 文本缩进和行高度	169
6.9 一些高级功能	170
6.10 何时选择何种方案	176
6.11 下一步	177
第7章 解析器和DOM	178
7.1 什么是解析器	178
7.1.1 解析器	178

7.1.2 验证解析器和非验证解析器.....	179
7.2 解析器和应用程序.....	180
7.2.1 XML程序的结构.....	180
7.2.2 基于对象的接口.....	181
7.2.3 基于事件的接口.....	182
7.2.4 对标准的需求.....	183
7.3 文档对象模型.....	184
7.4 DOM初步.....	184
7.4.1 一个DOM应用程序.....	185
7.4.2 DOM节点.....	188
7.4.3 NodeList.....	190
7.4.4 NamedNodeMap.....	190
7.4.5 Document对象.....	191
7.4.6 Element对象.....	191
7.4.7 Attr对象.....	192
7.4.8 Text对象.....	192
7.4.9 遍历元素树.....	193
7.4.10 一个更标准的版本.....	195
7.5 状态管理.....	198
7.5.1 维护状态的DOM应用程序.....	199
7.5.2 关于结构的说明.....	203
7.6 常见错误及解决办法.....	203
7.6.1 XML解析器是严格的.....	204
7.6.2 错误消息.....	204
7.6.3 XSLT的常见错误.....	205
7.7 DOM和Java.....	205
7.7.1 DOM和IDL.....	206
7.7.2 一个Java版本的DOM应用程序.....	206
7.7.3 三个主要的差别.....	208
7.7.4 解析器.....	209
7.8 应用程序中的DOM.....	209
7.8.1 浏览器.....	209
7.8.2 编辑器.....	214
7.8.3 数据库.....	214
7.9 下一步.....	214
第8章 另一种API: SAX.....	215
8.1 为什么要使用另一种API.....	215
8.1.1 基于对象的接口和基于事件的接口.....	216

8.1.2 基于事件的接口	216
8.1.3 为什么要使用基于事件的接口	219
8.2 SAX: 功能强大的API	220
8.2.1 SAX初步	220
8.2.2 编译示例程序	223
8.2.3 对事件处理器的逐步分析	224
8.3 常用的SAX接口和类	226
8.3.1 主要的SAX事件	226
8.3.2 XMLReader	227
8.3.3 XMLReaderFactory	227
8.3.4 InputSource	228
8.3.5 ContentHandler	228
8.3.6 Attributes	229
8.3.7 Locator	229
8.3.8 DTDHandler	229
8.3.9 EntityResolver	230
8.3.10 ErrorHandler	230
8.3.11 SAXException	230
8.4 状态维护	231
8.4.1 分层结构	239
8.4.2 状态	239
8.4.3 转换	241
8.4.4 经验教训	242
8.5 灵活性	243
8.5.1 建立灵活性	243
8.5.2 实施对结构的要求	243
8.6 下一步	244
第9章 编写XML	245
9.1 解析器的反向组件	245
9.2 使用DOM修改文档	246
9.2.1 插入节点	249
9.2.2 保存为XML	251
9.3 Netscape对DOM的支持	251
9.4 用于创建和修改文档的DOM方法	255
9.4.1 Document	256
9.4.2 Node	256
9.4.3 CharacterData	257
9.4.4 Element	257

9.4.5 Text.....	257
9.5 使用DOM创建新文档.....	258
9.5.1 创建节点.....	261
9.5.2 创建根元素和Document对象.....	262
9.5.3 显示结果.....	263
9.6 使用DOM创建文档.....	263
9.7 不使用DOM创建文档.....	264
9.7.1 非DOM数据结构.....	268
9.7.2 编写XML.....	268
9.7.3 隐藏语法.....	269
9.7.4 从非XML数据结构创建文档.....	270
9.8 XML文档的其他操作.....	271
9.8.1 将文档发送到服务器.....	271
9.8.2 保存文档.....	272
9.9 编写文档时考虑灵活性.....	273
9.9.1 使用XSLT支持多种XML模型.....	274
9.9.2 调用XSLT.....	277
9.9.3 为文档选择哪种结构.....	278
9.9.4 XSLT和自定义函数的比较.....	278
9.10 下一步.....	279
第10章 重要的XML模型.....	280
10.1 结构化和可扩展性.....	280
10.1.1 超级模型方法.....	281
10.1.2 模块化方法.....	283
10.1.3 经验教训.....	289
10.2 Xlink.....	290
10.2.1 简单链接.....	291
10.2.2 扩展链接.....	292
10.2.3 XLink和浏览器.....	295
10.3 XHTML.....	295
10.3.1 什么是XHTML.....	295
10.3.2 XHTML的优势.....	296
10.4 电子商务、XML/EDI和ebXML.....	297
10.4.1 何种电子商务.....	297
10.4.2 XML和电子商务.....	298
10.4.3 电子商务的先驱.....	299
10.4.4 经验教训.....	299
10.5 恰当的抽象层次.....	299

10.5.1 破坏性和非破坏性的转换.....	300
10.5.2 标记	303
10.5.3 避免太多的选项	306
10.6 属性和元素的比较	310
10.6.1 使用属性	310
10.6.2 使用元素	311
10.6.3 经验教训	312
10.7 下一步	312
第11章 n层结构和XML.....	313
11.1 什么是n层应用程序	313
11.1.1 客户/服务器应用程序	314
11.1.2 3层应用程序	315
11.1.3 n层	316
11.2 XCommerce应用程序	316
11.2.1 简化	317
11.2.2 为何如此麻烦	317
11.3 XML如何提供帮助	318
11.3.1 中间件	318
11.3.2 交换文件	320
11.3.3 简单对象访问协议(SOAP).....	321
11.4 SOAP编程	323
11.4.1 编写一个SOAP端点	323
11.4.2 编写一个SOAP客户机	327
11.4.3 关于灵活性的阐述	329
11.5 XCommerce体系结构	330
11.5.1 Web浏览器	330
11.5.2 表现层服务器	331
11.5.3 中间层	331
11.5.4 数据库	331
11.6 服务器端编程语言	331
11.6.1 Perl.....	332
11.6.2 JavaScript.....	332
11.6.3 Python.....	333
11.6.4 Omnimark	333
11.6.5 Java	333
11.7 下一步	333

第12章 整合：一个电子商务实例	334
12.1 创建Xcommerce.....	334
12.1.1 JDK.....	334
12.1.2 Classpath和属性.....	335
12.1.3 配置文件.....	336
12.1.4 编译和运行应用程序.....	340
12.1.5 一个更简单的方案.....	343
12.2 第1层：数据库.....	344
12.3 第2层：SOAP服务.....	355
12.3.1 Product类.....	361
12.3.2 ProductDetails类.....	362
12.3.3 OrderLine类.....	364
12.4 第3层：表现层Servlet.....	365
12.4.1 初始化.....	374
12.4.2 请求处理.....	375
12.4.3 doMall()和doProduct().....	376
12.4.4 样式表.....	377
12.4.5 doCheckout().....	379
12.4.6 doTakeOrder().....	380
12.5 工具类：字符串比较.....	381
12.6 下一步.....	381
附录A Java速成	383
A.1 正确看待Java.....	383
A.1.1 服务器端应用程序.....	384
A.1.2 JavaBean.....	384
A.2 下载Java工具.....	384
A.2.1 Java环境.....	385
A.2.2 XML组件.....	385
A.2.3 servlet引擎.....	386
A.2.4 数据库和JDBC.....	387
A.3 你的第一个Java应用程序.....	387
A.3.1 理解classpath.....	390
A.3.2 流程控制.....	390
A.3.3 变量.....	391
A.3.4 类.....	392
A.3.5 创建对象.....	392
A.3.6 访问域和方法.....	392
A.3.7 static.....	392

A.3.8 方法和参数.....	393
A.3.9 构造函数	393
A.3.10 包	394
A.3.11 JAR文件	394
A.3.12 导入	395
A.3.13 访问控制	395
A.3.14 注释和Javadoc.....	396
A.3.15 例外	397
A.4 servlet.....	398
A.5 你的第一个servlet.....	399
A.5.1 servlet程序清单.....	399
A.5.2 servlet描述符.....	401
A.5.3 WAR文件	402
A.5.4 运行示例	402
A.5.5 继承	404
A.5.6 doGet().....	404
A.5.7 地区和多语言应用程序.....	405
A.6 Java语言的更多概念	406
A.6.1 this和super	406
A.6.2 接口和多继承.....	406
A.7 下一步	407
附录B DTD和XML模式简单类型	408
B.1 DTD支持的简单类型	408
B.1.1 元素简单类型.....	408
B.1.2 属性简单类型.....	408
B.2 XML模式支持的简单类型.....	409
B.2.1 简单类型.....	409
B.2.2 Facet.....	410