

XML



基础教程（第2版）



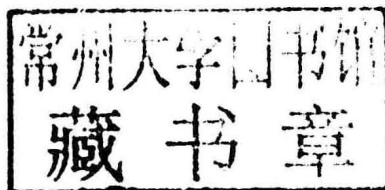
耿祥义 张跃平 编著

清华大学出版社

XML

基础教程（第2版）

耿祥义 张跃平 编著



清华大学出版社
北京

内 容 简 介

本书主要针对 XML 的初学者,重点放在 XML 中一些重要概念和技术上,所选例题具有较强的针对性,便于理解 XML 中的概念和技术,帮助读者在较短的时间内打下一个良好的基础。全书共分 9 章,内容包括 XML 简介、规范的 XML 文件、有效的 XML 文件、DOM 解析器、SAX 解析器、XPath 语言、XML 与数据库、XML 与 CSS 和 XML Schema 模式简介。

本书适合具有一定 Java 基础和初步 HTML 知识的读者阅读,也适合作为计算机、电子商务、信息类等专业的专业教材和社会培训机构相关专业的培训教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

XML 基础教程/耿祥义,张跃平编著. —2 版. —北京:清华大学出版社,2012.3

ISBN 978-7-302-27782-8

I. ①X… II. ①耿… ②张… III. ①可扩展语言,XML—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 281280 号

责任编辑:田在儒

封面设计:李 丹

责任校对:李 梅

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795764

印 刷 者:北京季蜂印刷有限公司

装 订 者:三河市李旗庄少明印装厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:12.5 字 数:286 千字

版 次:2006 年 4 月第 1 版 2012 年 3 月第 2 版 印 次:2012 年 4 月第 1 次印刷

印 数:1~3000

定 价:25.00 元

产品编号:043487-01

一、内容结构

本书是《XML 基础教程》的第 2 版,对第 1 版的内容进行了更新,对章节的先后顺序做了调整,使其更加适合教学。全书分为 9 章。第 1 章主要对 XML 做了简单的介绍,帮助读者对 XML 有一个大致的了解。第 2 章详细讲解规范的 XML 文件,使读者认识到规范性的重要作用。第 3 章主要讲解有效的 XML 文件,特别重点讲解 DTD 文件。第 4 章详细讲解 DOM 解析器,特别对如何使用 DOM 生成 XML 文件给予较详细的讲解。第 5 章详细讲解 SAX 解析器,重点体现 SAX 解析器的优势。第 6 章详细讲解 XPath 语言,重点体现 XPath 在检索 XML 文件中数据上的优势。第 7 章讲解 XML 与数据库,使读者领会二者互相转化的重要意义,并掌握如何实现二者转换的设计技术。第 8 章讲解 CSS 技术及如何使用 CSS 显示 XML 中数据的细节。第 9 章介绍 XML Schema 模式,讲解如何用模式约束 XML 标记的数据类型。

二、本书特色

- 强调 XML 基础知识;
- 突出 XML 解析器和 Xpath 查询;
- 结合实例、注重应用。

本书知识点明确,内容衔接流畅、通俗易懂,便于教学和自学。许多例题都是经过精心考虑,所有代码都测试通过。

三、资源下载

读者可登录清华大学出版社网站下载本书全部的例题代码。

四、作者简介

耿祥义,1995 年中国科学技术大学获理学博士学位。1997 年从中山大学博士后流动

站出站。现任大连交通大学教授。已编写出版《Java 设计模式》、《Java 2 实用教程》、《Java 课程设计》、《JSP 实用教程》、《JSP 程序设计》等 10 余部教材。

张跃平,现任大连交通大学讲师。已编写和参编出版《Visual FoxPro 课程设计》、《Java 2 实用教程》、《Java 设计模式》、《JSP 实用教程》和《Java 课程设计》5 部教材。

五、交流沟通

希望本教材能对读者学习 XML 有所帮助,并请读者批评指正(xygeng0629@sina.com)。



目录

第 1 章 XML 简介	1
1.1 什么是 XML	1
1.2 XML 文件的编辑与保存	3
1.3 XML 和 HTML 有何不同	4
1.4 XML 解析器	7
1.5 XML 的优势	7
习题 1	8
第 2 章 规范的 XML 文件	10
2.1 规范性	10
2.2 XML 声明	11
2.2.1 version 属性	11
2.2.2 encoding 属性	11
2.2.3 standalone 属性	13
2.3 标记	13
2.3.1 空标记	14
2.3.2 非空标记	14
2.3.3 标记的名称	17
2.3.4 根标记	17
2.3.5 标记的子孙关系	17
2.4 特殊字符	18
2.5 CDATA 段	18
2.6 标记的文本数据	19

2.7 属性	19
2.7.1 属性的构成	19
2.7.2 使用属性的原则	20
2.8 注释	20
2.9 名称空间	21
2.9.1 有前缀和无前缀的名称空间	21
2.9.2 标记中声明名称空间	22
2.9.3 名称空间的作用域	22
2.9.4 名称空间的名字	23
习题 2	24
第 3 章 有效的 XML 文件	26
3.1 有效 XML 文件的定义	27
3.1.1 初识 DTD	28
3.1.2 DTD 文件的保存	29
3.1.3 XML 文件与 DTD 文件相关联	29
3.2 如何检查有效性	31
3.3 DTD 中的元素	33
3.3.1 约束标记只包含文本数据	33
3.3.2 约束标记的子标记	34
3.3.3 约束标记的混合内容	37
3.3.4 EMPTY 和 ANY	38
3.4 DTD 的完整性	39
3.5 DTD 中的属性约束列表	39
3.5.1 ATTLIST 属性约束列表	40
3.5.2 属性的默认值	42
3.5.3 属性类型	47
3.6 内部 DTD	53
习题 3	55
第 4 章 DOM 解析器	57
4.1 认识 DOM 解析器	57
4.1.1 DOM 标准	57
4.1.2 初识 JAXP	58
4.1.3 Document 节点	59
4.2 节点的类型	61
4.2.1 Node 接口	61
4.2.2 Node 接口的常用方法	62

4.2.3 节点的子孙关系	62
4.2.4 使用递归方法输出节点中的数据	62
4.3 Document 节点	64
4.4 Element 节点	64
4.5 Text 节点	67
4.6 Attr 节点	71
4.7 DocumentType 节点	72
4.8 处理空白	74
4.9 验证规范性和有效性	76
4.10 使用 DOM 生成 XML 文件	76
4.10.1 Transformer 对象	77
4.10.2 用于修改 Document 的常用方法	77
4.10.3 用 DOM 建立 XML 文件	78
习题 4	82
第 5 章 SAX 解析器	83
5.1 初识 SAX 解析器	83
5.1.1 SAX 解析器及其工作原理	83
5.1.2 创建 SAX 解析器的步骤与事件处理	84
5.2 文档开始与结束事件	87
5.3 标记开始与结束事件	89
5.4 文本事件	91
5.5 名称空间事件	94
5.6 错误事件	97
5.7 处理空白	101
习题 5	103
第 6 章 XPath 语言	104
6.1 XPath 简介	105
6.1.1 初识 XPath 路径表达式	105
6.1.2 使用 XPath API	106
6.2 Node 节点	107
6.2.1 节点之间的关系	107
6.2.2 节点的类型	108
6.2.3 节点的名字与值	110
6.3 XPath 路径表达式的结构	110
6.3.1 绝对路径与相对路径	110
6.3.2 定位步与节点集	111

6.3.3	轴及缩写	114
6.3.4	节点测试	116
6.4	谓词	119
6.4.1	谓词的格式与作用	119
6.4.2	寻找特殊位置的节点	121
6.4.3	寻找有特殊属性的节点	122
6.4.4	寻找有特殊关系节点的节点	124
6.4.5	使用谓词嵌套寻找节点	126
6.5	节点集上使用谓词	127
6.6	节点集的并运算	128
6.7	Java XPath API	129
6.7.1	按 NodeSet 计算	129
6.7.2	按 Node 计算	129
6.7.3	按字符串计算	130
6.7.4	按布尔值计算	131
6.7.5	按数值计算	131
6.8	节点集与函数	133
6.9	图书查询	134
	习题 6	137
第 7 章	XML 与数据库	141
7.1	JDBC	142
7.2	Microsoft Access 数据库	142
7.2.1	建立数据库	142
7.2.2	创建表	143
7.3	连接数据库	143
7.3.1	JDBC-ODBC 桥接器	143
7.3.2	ODBC 数据源	144
7.3.3	建立连接	145
7.4	XML 至数据库	146
7.5	数据库至 XML	148
	习题 7	151
第 8 章	XML 与 CSS	152
8.1	初识 CSS	152
8.2	XML 关联 CSS	153
8.3	标记与样式表	154
8.3.1	标记的名字与样式表的名称	154

8.3.2 CSS 的显示规则	155
8.4 数据结构与显示相分离	156
8.5 设置文本的显示方式	158
8.5.1 块方式	158
8.5.2 行方式	160
8.5.3 按列表方式	161
8.5.4 不显示	163
8.6 字体	163
8.7 文本样式	165
8.8 边框	167
8.9 边缘	169
8.10 颜色和背景	171
8.11 显示图像	171
8.12 设置鼠标的形状	173
8.13 处理层叠	174
习题 8	175
第 9 章 XML Schema 模式简介	177
9.1 什么是 XML Schema	177
9.2 XML Schema 中的标记	178
9.2.1 根标记	178
9.2.2 元素标记	179
9.2.3 属性标记	180
9.3 XML Schema 模式的验证	180
9.4 简单类型元素	184
9.5 复杂类型元素	186
9.6 属性	188
习题 9	190

第 1 章 XML 简介

主要内容

- 什么是 XML
- XML 文件的编辑与保存
- XML 和 HTML 有何不同
- XML 解析器
- XML 的优势

随着网络的迅速发展以及规模的扩大,对信息的规范性提出了更加严格的要求,XML 就是在这一背景下诞生的一种数据格式标准。

在学习这门课程之前读者应当初步了解 HTML,并有一定的 Java 语言基础。另外,需要强调的是,由于许多 Web 技术都和 XML 有关,因此本课程的学习对于进一步学习、理解 Web 技术是非常有帮助的。

1.1 什么是 XML

随着网络的迅速发展,万维网联盟(World Wide Web Consortium,W3C)认识到信息规范化的重要性,即用一种规范化的格式来处理信息,从而可以使得人们更加方便地交互信息。XML 是 eXtensible Markup Language 的缩写,是由万维网联盟定义的一种语言,称为可扩展标记语言。所谓可扩展性是指 XML 允许用户按照 XML 规则自定义标记。

XML 文件是由标记及其所标记的内容构成的文本文件,与 HTML 文件不同的是,这些标记可自由定义,其目的是使得 XML 文件能够很好地体现数据的结构和含义。W3C 推出 XML 的主要目的是使得 Internet 网络上的数据相互交流更方便,让文件的内容更加通俗易懂。

以下是一个简单的 XML 文件。

first.xml

```
<?xml version="1.0" ?>
<学生>
  <姓名> 张三
  <性别> 男 </性别>
```

```

    <出生日期> 1995/05/15 </出生日期>
</姓名>
<姓名> 翠花
    <性别> 女 </性别>
    <出生日期> 1998/08/08 </出生日期>
</姓名>
</学生>

```

尽管上述 XML 文件非常简单,但基本体现了 XML 文件的基本结构,总结如下(有关细节将在后续章节讲解)。

- XML 文件包含一个 XML 声明,其位置必须在 XML 文件的首行(有关 XML 声明将在 2.1 节详细介绍):

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

- XML 文件中包含有若干个标记,每个标记由开始标签和结束标签构成。
- XML 文件有且仅有一个根标记,其他标记都必须封装在根标记中,文件的标记必须形成树形结构。
- 标记的开始标签和结束标签之间的内容称为该标记所标记的内容,简称标记的内容。一个标记的内容中可以包含文本或其他标记,其中包含的标记称为该标记的子标记。

上面的 XML 文件的根标记的开始标签是“<学生>”,结束标签是“</学生>”,该根标记有 2 个子标记“<姓名>...</姓名>”。“<姓名>...</姓名>”的内容既有文本也有子标记,例如其中一个标记“<姓名>...</姓名>”中的文本是“张三”,子标记是“<性别>...</性别>”和“<出生日期>...</出生日期>”。

XML 文件必须符合一定的语法规则(后续的章节会详细地讲解 XML 的语法规则),只有符合这些规则,XML 文件才可以被 XML 解析器解析,以便利用其中的数据,如图 1.1 所示。



图 1.1 XML 解析器

下面的 XML 文件都是错误的,其中的“errorOne.xml”没有根标记,“errorTwo.xml”虽然有根标记,但根标记的两个“节目”子标记有交叉,导致标记没有形成树形结构。

errorOne.xml

```

<节目> 乡村爱情
    <播出时间> 20 时 22 分</播出时间>
</节目>
<节目> 借枪
    <播出时间> 22 时 38 分</播出时间>

```

</节目>

errorTwo.xml

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

```
<国贸大厦>
```

```
  <商品> 电视机
```

```
    <价格> 6368 元
```

```
  </商品>
```

```
    </价格>
```

```
  <商品> 手机
```

```
    <价格> 2678 元</价格>
```

```
  </商品>
```

```
</国贸大厦>
```

“商品”与“价格”标签出现交叉

1.2 XML 文件的编辑与保存

1. 编辑与保存

XML 文件是具有特殊扩展名(.xml)的文本文件,因此需使用纯文本编辑器来编辑 XML 文件。本书以 Windows 系统自带的“记事本”做编辑器。XML 文件保存的扩展名必须为“.xml”,例如“Example.xml”、“hello.xml”等。需要特别注意是,名字区分大小写,hello.xml 和 Hello.xml 是完全不同的 XML 文件的名字。

对于初学者不建议使用集成开发工具来编写 XML 文件,尽管各种集成开发工具也提供了用于编辑 XML 文件的编辑器,但同时也屏蔽了 XML 的许多知识点,这非常不利于学习 XML。在掌握了 XML 知识后,具体开发项目时,应当选择一个流行的、成熟的集成开发工具来编写 XML 文件,这有利于更好、更快地编写 XML 文件,例如 XMLSpy 就是非常优秀的 XML 集成开发工具。对于已掌握 XML 知识的人,学习使用各种集成开发工具是没有任何困难的。

一个 XML 文件应当以 XML 声明作为文件内容的第一行,在其前面不能有空白或其他的处理指令或注释。XML 声明以“<? xml”标识开始,以“? >”标识结束。注意“<?”和“xml”之间,以及“?”和“>”之间不能有空格。

以下是一个最基本的 XML 声明:

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

一个简单的 XML 声明中可以只包含属性 version,目前该属性的值取 1.0(1.1 还没有正式公布,1.1 增加了一些极少被使用的功能),指出该 XML 文件使用的 XML 版本。XML 声明中还可以指定 encoding 属性的值,该属性规定 XML 文件采用哪种字符集进行编码。例如:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

如果在 XML 声明中没有显示地指定 encoding 属性的值,那么该属性的默认值为 UTF-8 编码。如果 encoding 属性的值为 UTF-8,XML 文件必须按照 UTF-8 编码来保存,这样

XML 解析器就会识别 XML 中的标记并正确解析标记中的内容。

使用文本编辑器“记事本”编辑 XML 文件,在保存文件时,必须将“保存类型”选择为“所有文件”,将“编码”选择为“UTF-8”。如果在保存文件时,系统总是给文件名尾加上“.txt”,那么在保存文件时可以将文件名用双引号括起,如图 1.2 所示。



图 1.2 保存 XML 文件

2. 如何检查 XML

XML 的语法规则非常严格,这一点和 HTML 有很大的不同,HTML 本身语法十分不严格,严重影响网络信息传送和共享。W3C 吸取了 HTML 发展的教训,对 XML 指定了严格的语法标准,例如,标记要有一个开始标记和结束标记、所有的标记都必须合理嵌套,即形成树形结构。也就是说 XML 文件必须符合一定的语法规则,只有符合这些规则,XML 文件才可以被 XML 解析器解析,以便利用其中的数据。

XML 文件分为规范的 XML 文件和有效的 XML 文件,符合 W3C 制定的基本规则的 XML 文件称为规范的 XML 文件,规范的 XML 文件如果再符合额外的一些约束就称为有效的 XML 文件。有关 XML 的详细语法将从第 2 章开始讲述。为了检查 XML 文件是否规范,一个简单的办法就是用浏览器,例如 IE6.0。打开 XML 文件,如果 XML 是规范的,浏览器将显示 XML 源文件,否则将显示出错信息。图 1.3 显示了前面第 1.1 节中规范的 XML 文件 first.xml;图 1.4 显示了前面第 1.1 节中不规范的 XML 文件 errorOne.xml。

```
<?xml version="1.0" ?>
- <学生>
  - <姓名>
    张三
    <性别>男</性别>
    <出生日期>1995/05/15</出生日期>
  </姓名>
  - <姓名>
    翠花
    <性别>女</性别>
    <出生日期>1998/08/08</出生日期>
  </姓名>
</学生>
```

图 1.3 打开规范的 XML 文件

XML 文档只能有一个顶层元素。处理资源
'file:///C:/00xml/first.xml' 时出错。

```
<节目> 借枪
-^
```

图 1.4 打开不规范的 XML 文件

注意: 编写保存一个 XML 文件后,养成用浏览器检查 XML 文件是否有语法错误的习惯。

1.3 XML 和 HTML 有何不同

尽管 XML 和 HTML 都是 W3C 定义的语言,但二者有很大的不同,阐述如下。

1. HTML 的核心

HTML 是用来编写 Web 页的语言(超文本标记语言),HTML 的核心是把数据和数

据的显示外观捆绑在一起,即 HTML 中的标记的出发点不是为了体现数据的含义,而是体现数据的显示格式,整个 HTML 文件的目的是为了组织数据,而是为了组织某些数据的显示外观。如果某个应用系统只想使用 HTML 中所包含的数据而不需要它的显示外观,可以想象,该应用系统将 HTML 中的数据和外观分离是多么的困难。

另外,HTML 不允许用户自定义标记,目前的 HTML 有一百多个标记。HTML 不能体现数据的组织结构,只能够描述数据的显示格式。下面的 HTML 将数据分别用黑体 1、黑体 2 和黑体 3 来显示。

show.html

```
<html>
  <H1> 张小山
    <H2> 男 </H2>
  </H1>
  <H2> 李翠花
    <H3> 女 </H3>
  </H2>
</html>
```

对于上述 HTML 文件,一个不认识汉字的外国人,并不知道“张小山”和“李翠花”是人的名字。因为 HTML 中的标记的出发点不是为了体现数据的含义,而是为了体现数据的显示格式,并不体现数据的组织结构。浏览器能将 HTML 中标记“<H1>...</H1>”,“<H2>...</H2>”和“<H3>...</H3>”所标记的文本内容分别用黑体 1、黑体 2 和黑体 3 显示在浏览器中,如图 1.5 所示。

张小山

男

李翠花

女

图 1.5 用浏览器打开 HTML 文件

2. XML 的核心

和 HTML 不同的是,XML 的核心是描述数据的组织结构,使 XML 可以作为数据交换的标准格式。XML 可自定义标记,其标记名称是对所标记的数据内容含义的抽象,而不是数据的显示格式,而且 XML 文件通过其中的标记来表示数据的组织结构。例如,下述 XML 文件 second.xml。

second.xml

```
<student>
  <name> 张小山
    <sex> 男 </sex>
  </name>
  <name> 李翠花
    <sex> 女 </sex>
  </name>
</student>
```

对于上述 XML 文件,即使是一个不认识汉字的外国人,也很清楚地知道“张小山”和“李翠花”是学生的名字,并清楚地知道 second.xml 中数据的组织结构。

XML 非常关心数据的组织结构,以便 XML 解析器按照其组织结构分解出数据(有关解析器的详细内容将在 4、5 章讲述),XML 本身不提供数据的显示格式。

XML 有效地分离数据的组织结构和显示外观,即不将显示外观和其中的标记直接进行关联,因此浏览器不能直接显示 XML 文件中的标记的内容。如果需要浏览器显示 XML 文件中标记的内容,就必须以某种方式告诉浏览器如何显示,例如使用层叠样式表(CCS),如图 1.6 所示(显示 XML 文件中标记的内容的有关知识将在第 8 章讲述)。

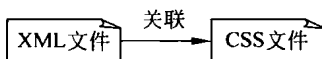


图 1.6 XML 文件关联负责显示数据的 CSS

下面的 XML 文件将自己关联到一个层叠样式表,以便浏览器能显示 XML 文件中标记的文本内容。层叠样式表中最重要的一部分就是样式表,其作用是说明 XML 文件中的标记内容用何种方式来显示。例如,如果想用黑体来显示 XML 文件中标记“<name>...</name>”所标记的文本内容,就可以在层叠样式表文件中包含如下的样式表:

```
name
{ display:block;font-size:36pt;font-style:normal;font-weight:bold
}
```

该样式表的作用是告知浏览器将 XML 文件中标记“<name>...</name>”所标记的文本内容用黑体显示在一个“块区域”中;而

```
sex
{ display:line;font-size:12pt;font-style:italic;
}
```

告知浏览器将 XML 文件中标记“<sex>...</sex>”所标记的文本内容用斜体显示在一行中。

注意:当使用层叠样式表和 XML 关联时,XML 文件中的标记的名字不要含有非 ASCII 码字符(例如,名字中不能有汉字)。

以下的“showXML.css”是一个简单的层叠样式表,该文件可以保存为编码为“ANSI”的文件或编码为“UTF-8”的文件,并和其关联的 XML 文件存放在同一目录中。

showXML.css

```
name
{ display:block;font-size:18pt;font-weight:bold
}
sex
{ display:line;font-size:16pt;font-style:italic
}
birthday
{ display:line;font-size:9pt;font-weight:bold
}
```

XML 文件使用操作指令:

```
<?xml-stylesheet ?>
```

关联到某个层叠样式表,例如:

```
<?xml-stylesheet href="showXML.css" type="text/css" ?>
```

以下是一个与 showXML.css 层叠样式表相关联的 XML 文件 three.xml,用浏览器打开 three.xml 文件,浏览器就会按照 showXML.css 中的样式表来显示 XML 文件中标记的文本内容,效果如图 1.7 所示。

three.xml

```
<?xml version="1.0" ?>
<?xml-stylesheet href="showXML.css" type="text/css" ?>
<student>
  <name> 张小山
    <sex> 男 </sex>
    <birthday> 1995 年 05 月 15 日 </birthday>
  </name>
  <name> 李翠花
    <sex> 女 </sex>
    <birthday> 1997 年 07 月 27 日 </birthday>
  </name>
</student>
```

张小山	男	1995年05月15日
李翠花	女	1997年07月27日

图 1.7 使用 CSS 显示 XML 中的数据

注意: CSS 是用于布局网页外观的技术,在 XML 中经常会用到 CSS。

1.4 XML 解析器

XML 解析器是 XML 和应用程序之间的一个软件组织,其目的是为应用程序从 XML 文件中解析出所需要的数据。例如,应用程序可能需要 XML 文件所标记的商品价格,并对价格做数据分析处理。现在普遍使用的 XML 解析器都是 Java 语言编写的,本书将采用这样的解析器,有关解析器的详细内容将在 4、5 章讲述。

1.5 XML 的优势

XML 作为表示结构化数据的行业标准,已获得广泛的行业支持。XML 在采用简单、柔性的标准化格式表达以及在应用间交换数据方面是一个革命性的进步。XML 的威力在于不仅提供了直接在数据上工作的通用方法,而且将数据的结构和显示相分离,允许不同来源数据的无缝集成和对同一数据的多种处理。从数据描述语言的角度看,XML 是灵活的、可扩展的,有良好的结构和约束;从数据处理的角度看,它足够简单且易于阅读,几乎和 HTML 一样易于学习,同时又易于被应用程序处理。许多网络应用都在大量使用 XML,XML 已经成为网络应用技术的基础。

通过本书的进一步学习,读者会逐步体会到 XML 的优点。例如,通过 1.3 节的学习已经初步体会到了 XML 是如何将数据的组织结构和显示相分离的,如果想修改 three.xml 的显示外观,只需修改它所关联的 CSS 即可。