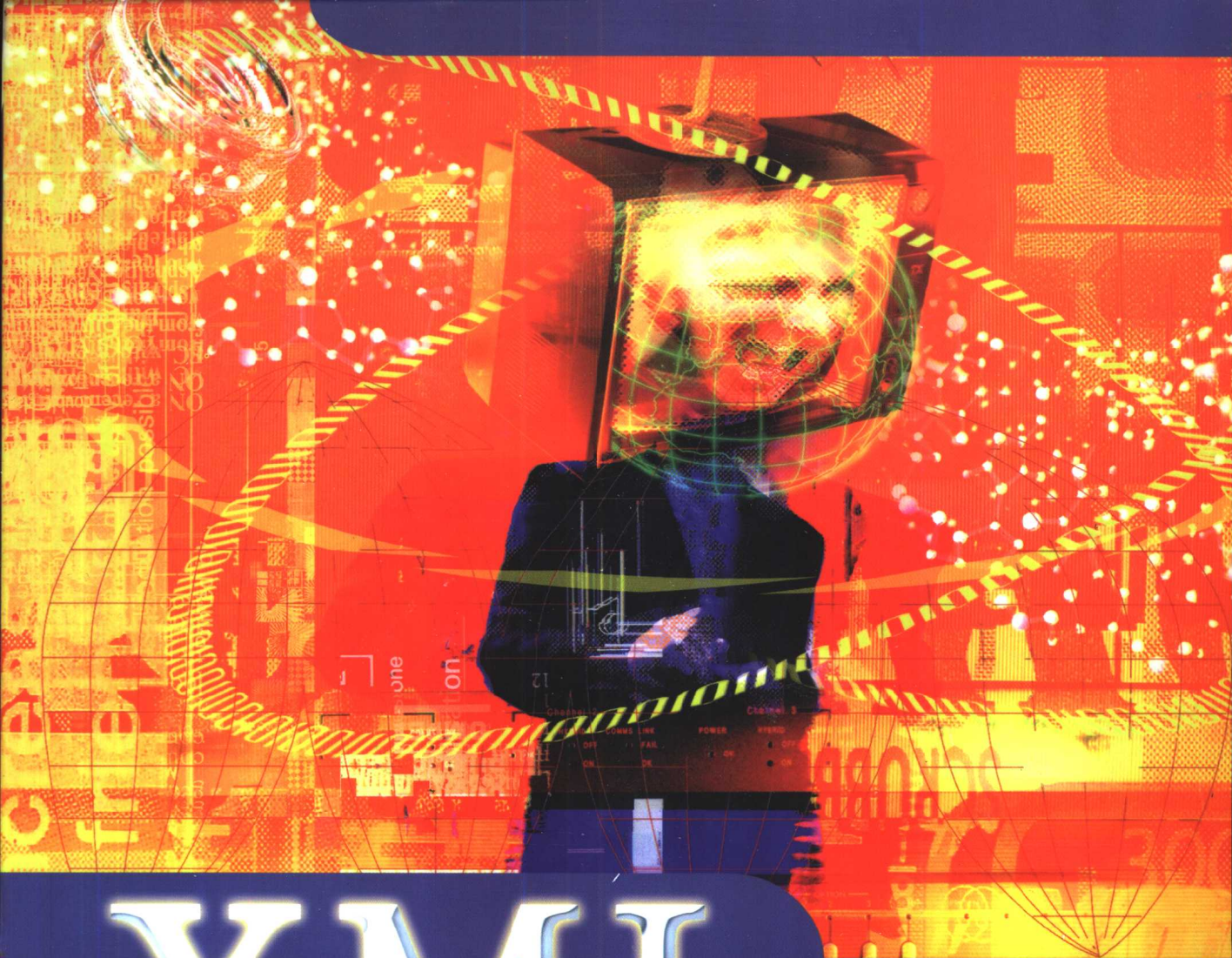




XML

网页开发实例教程



李永祥 张德海 唐 灿 编著



清华大学出版社

XML 网页开发实例教程

李永祥 张德海 唐 灿 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书结合大量的实例与详尽的理论分析,用通俗易懂的语言全面系统地介绍了“网络上的 ASCII 码”——XML。

全书参照 W3C 的最新标准讲述了 XML 语言的主要基本技术,主要包括 XML 的起源、发展与应用,XML 语法,文档类型定义 DTD,XML Schema,XML 和名域,XSLT,XPath,XSL FO,CSS。

全书在内容上循序渐进,讲述上深入浅出,并配有大量的实例,是指引对 XML 技术感兴趣的读者朋友步入 XML 殿堂的入门书籍,同时对具有基础的读者朋友也有一定的参考价值。

版权所有,翻印必究。

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。

图书在版编目(CIP)数据

XML 网页开发实例教程/李永祥,张德海,唐灿编著.一北京:清华大学出版社,2002

ISBN 7-900643-45-1

I .X... II.①李... ②张... ③唐... III.可扩展语言,XML-程序设计-教材 IV.TP312

中国版本图书馆 CIP 数据核字(2002)第 041122 号

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:吴宏伟

印 刷 者:清华大学印刷厂

发 行 者:新华书店总店北京发行所

开 本:787×1092 1/16 **印张:**18.25 **字数:**404 千字

版 次:2002 年 9 月第 1 版 2002 年 9 月第 1 次印刷

书 号:ISBN 7-900643-45-1

印 数:0001~5000

定 价:30.00 元(附光盘)

前 言

XML 是 W3C（万维网联盟）在 1998 年 2 月正式推出的作为下一代网页发布的语言。它是一个精简的 SGML（标准通用标记语言），它保留了 SGML 的可扩展功能，并具有 HTML 的易用性。

作为一种新兴的计算机技术，XML 推动着 Web 飞速向前发展。它的出现给各行各业带来了无限的商机。尤其是 XML 技术同电子商务的结合，推动着传统产业高效的飞速发展。虽然 XML 自公布之日才短短的几年时间，但却得到了人们的极大关注，相关的研究及其应用不断涌现。

作为一本详尽讲述 XML 主要基础技术的书籍，本书最大的特点是既有理论分析，又有大量详尽的实例配合读者朋友加深对枯燥晦涩的概念的理解。全书讲述深入浅出，条理分明。通过对本书的学习，将为读者朋友深入学习 XML 技术的高级应用打下良好的基础。

全书共分 10 章，主要内容如下：

第 1 章：介绍了 XML 的起源、发展和应用，并对 XML 的各项基础技术作了“第一次的亲密接触”，为后面章节深入学习 XML 技术打下了良好的基础。

第 2 章：详细介绍了 XML 的基本语法，包括元素、属性、XML 文档的逻辑结构和物理结构等内容，并通过对实例的详尽分析，加深读者朋友对概念的理解。

第 3 章：介绍了 DTD 的基本要素、组成、实体、名称空间等内容。

第 4 章：介绍了 XML Schema 的基本知识和使用方法。

第 5 章：讲述了什么是名域，名域的作用及用法。

第 6~8 章：用丰富的实例详细讲述了 XSLT 以及 XPath 的用法。

第 9 章：介绍了 XSL 格式化对象 FO。

第 10 章：详细介绍了 CSS 及其使用方法。主要内容有 CSS 的基本概念、基本规则和基本属性。此外还介绍了 CSS2，包括 CSS2 的新增特性、有声样式单和可视格式化等。

本书是指引对 XML 技术感兴趣的读者朋友步入 XML 殿堂的入门书籍，同时对具有基础的读者朋友及从事 XML 技术应用的开发人员也有一定的参考价值。

本书第 1 章、第 2 章和第 7 章由李永祥编写，第 3~5 章和第 10 章由张德海编写，第 6 章、第 8 章和第 9 章由唐灿编写，喻智成、梁新元对本书的编写提出了众多宝贵意见并参与了其中的部分工作。本书在编写过程中有幸得到成都道向科技有限责任公司李香敏老师的指点，在此表示衷心的感谢！

本书在编写过程中本着追求完美的理念，希望带领读者轻松步入 XML 的神秘殿堂。但由于时间仓促和编者水平所限，书中难免有不妥或不足之处，欢迎各位读者批评指正。

目 录

第 1 章 XML 简介.....	1
1.1 XML 的产生.....	2
1.1.1 XML 的发展历程.....	2
1.1.2 SGML、HTML 和 XML.....	3
1.2 XML 核心技术.....	6
1.2.1 XML1.0 规范.....	7
1.2.2 XML 文档定义和校验.....	7
1.2.3 XML 中的定位和链接.....	9
1.2.4 XML 的转换和处理.....	10
1.2.5 XML 的表现.....	11
1.3 XML 工具.....	13
1.3.1 编辑器.....	13
1.3.2 解析器.....	15
1.3.3 浏览器.....	16
1.4 XML 应用.....	16
1.4.1 XML 在行业中的应用.....	17
1.4.2 XML 在企业中的应用.....	18
1.5 总结.....	19
第 2 章 XML 语法细节	21
2.1 XML 文档.....	22
2.1.1 XML 文档结构.....	22
2.1.2 规范的 XML 文档.....	24
2.1.3 XML 的设计目标.....	25
2.1.4 EBNF 语法介绍.....	26
2.2 字符数据和标记.....	27
2.2.1 字符数据.....	27
2.2.2 XML 中的标记.....	28
2.3 XML 文档的物理结构.....	31
2.3.1 XML 中的元素.....	31
2.3.2 XML 中的属性.....	32

2.3.3 XML 中的实体	34
2.4 实例学习	35
2.4.1 确定 XML 文档结构	35
2.4.2 实例 XML 文档	37
2.5 总结	39
第 3 章 文档类型定义	41
3.1 DTD 语法	42
3.1.1 什么是 DTD	42
3.1.2 DTD 的结构	42
3.2 元素声明	48
3.2.1 根元素标志	48
3.2.2 序列	49
3.2.3 #PCDATA 字符串	50
3.2.4 ANY 自由元素	51
3.2.5 EMPTY 空元素	51
3.2.6 +、*、? 个数限定	53
3.2.7 嵌套结构、选择结构和混合结构	55
3.3 属性声明	58
3.3.1 数据类型	58
3.3.2 默认值	72
3.3.3 XML 的预定义属性	77
3.4 实体和 DTD	79
3.4.1 内部实体和外部实体	79
3.4.2 已析实体和未析实体	81
3.4.3 通用实体和参数实体	83
3.4.4 多个外部 DTD 合作方式	86
3.5 总结	89
第 4 章 XML Schema	91
4.1 XML Schema 简介	92
4.1.1 DTD 的特点	92
4.1.2 XML Schema 的特点	92
4.1.3 其他的文档定义方式	93
4.2 XDR 的指定	94
4.3 XDR 组件的细节	96
4.3.1 XDR 的框架结构	96

4.3.2	属性声明和元素属性声明.....	97
4.3.3	元素声明和子元素声明.....	101
4.3.4	模型组和模型组定义.....	104
4.3.5	复杂类型定义.....	106
4.4	XDR 的数据类型.....	109
4.4.1	数据类型体系.....	109
4.4.2	数据类型声明.....	111
4.5	总结.....	115
第 5 章	XML 和名域.....	117
5.1	XML Schema 和名域.....	118
5.1.1	名域的概念及作用.....	118
5.1.2	名域的声明及使用.....	119
5.1.3	名域在 XML 中的使用——XML Schema.....	122
5.2	应用名域.....	124
5.2.1	使用限定的名称.....	124
5.2.2	名域的范围和默认.....	125
5.3	总结.....	127
第 6 章	XSL 初接触.....	129
6.1	XSL 简介.....	130
6.1.1	XSL 的历史.....	130
6.1.2	XSL 的应用范围.....	130
6.1.3	XSL、XSLT 和 XPath 的关系.....	131
6.1.4	XSL 在 XML 技术体系中的地位.....	132
6.2	XSL 的使用环境和使用方式.....	132
6.2.1	XSLT 处理器介绍.....	132
6.2.2	使用 MSXML 及 MSXSL.....	134
6.2.3	配置 XALAN.....	137
6.2.4	XSLT 的使用方式.....	138
6.3	XSLT 的处理模型.....	139
6.3.1	一个简单的例子.....	139
6.3.2	规则与节点转换.....	141
6.3.3	源树与结果树.....	141
6.3.4	<xsl:stylesheet>、<xsl:output>和<xsl:transform>.....	142
6.3.5	模板.....	144
6.3.6	模式上下文.....	146

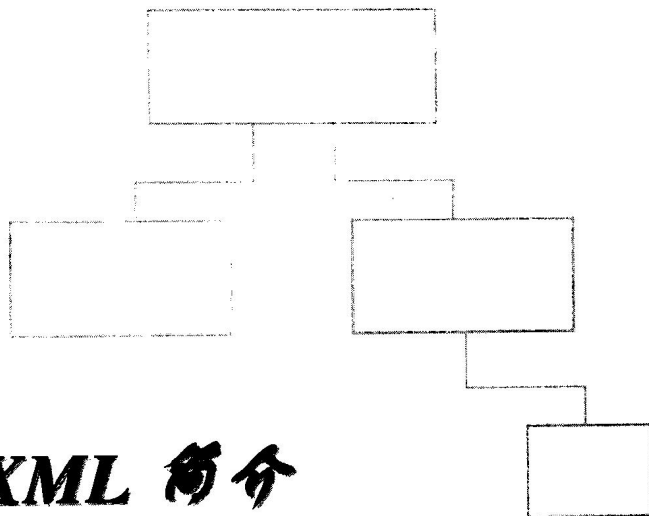
6.3.7 默认规则.....	146
6.3.8 文档处理顺序.....	147
6.4 总结.....	148
第 7 章 可扩展路径语言 XPath	149
7.1 XPath 的基本概念.....	150
7.1.1 什么是 XPath.....	150
7.1.2 表达式和定位路径.....	151
7.1.3 定位阶.....	152
7.1.4 定位路径举例.....	154
7.1.5 XPath 的简缩语法.....	156
7.2 Xpath 的函数.....	157
7.2.1 节点集及其函数.....	158
7.2.2 字符串及其函数.....	159
7.2.3 布尔值及其函数.....	161
7.2.4 数值及其函数.....	161
7.3 总结.....	162
第 8 章 XSL 再接触.....	163
8.1 XSLT 语法初步	164
8.1.1 取值元素.....	164
8.1.2 新元素和新属性的产生.....	176
8.1.3 常量和参数.....	180
8.1.4 分组和排序.....	184
8.1.5 XSLT 语法进阶	186
8.2 编写 XSLT 的原则和步骤	190
8.3 总结.....	191
第 9 章 XSL 格式化对象.....	193
9.1 格式化对象及属性.....	194
9.1.1 XSL FO 转换过程	194
9.1.2 转换为 PDF 的操作过程.....	195
9.1.3 FO 名域 (FO namespace)	196
9.1.4 格式化对象 (formatting object)	196
9.1.5 格式化属性 (formatting attribute)	197
9.2 页面布局和内容.....	198
9.2.1 初始化.....	198

9.2.2 页面布局.....	198
9.2.3 内容.....	201
9.3 部分对象详解.....	202
9.3.1 水平线.....	202
9.3.2 图形.....	204
9.3.3 链接.....	205
9.3.4 列表.....	207
9.3.5 表格.....	210
9.3.6 字符.....	216
9.3.7 序.....	217
9.3.8 脚注.....	218
9.3.9 浮动.....	221
9.4 XSL2.0.....	222
第 10 章 CSS—XML 文档表现的另一种选择	223
10.1 CSS 简介.....	224
10.1.1 什么是 CSS.....	224
10.1.2 HTML 中使用 CSS 和 XML 中使用 CSS 的对比.....	224
10.1.3 如何显示 XML 文档.....	224
10.2 CSS 基本规则.....	227
10.2.1 选择元素.....	227
10.2.2 继承.....	233
10.2.3 级联顺序.....	234
10.3 样式表和文档的联系.....	236
10.3.1 XML-STYLE SHEET 处理指令.....	236
10.3.2 @IMPORT 指令	236
10.3.3 其他方法.....	237
10.4 CSS 属性值的类型.....	237
10.4.1 长度值.....	237
10.4.2 URL 值.....	238
10.4.3 颜色值.....	238
10.4.4 关键字值.....	239
10.5 CSS 基本属性.....	239
10.5.1 display 属性	239
10.5.2 字体属性.....	241
10.5.3 color 颜色属性.....	246
10.5.4 background 背景属性.....	248

10.5.5	text 文本属性.....	252
10.5.6	框属性.....	259
10.6	CSS2 简介.....	265
10.6.1	CSS2 中的新特点.....	265
10.6.2	选择符的扩充.....	265
10.6.3	新增细节.....	271
10.7	总结.....	278

第1章

XML 简介



主要内容



XML 的起源和发展历程



XML 的主要技术



XML 的工具：编辑器、解析器、浏览器



XML 在企业中的应用



本章讲述 XML 的起源与发展，简要介绍 XML 的主要基础技术：用于 XML 文档定义和校验的 DTD 和 XML Schema，名域，用于 XML 定位和链接的 XPath、XPointer、XLink、XSLT、DOM、SAX 以及 CSS 和 XSL FO，最后描述 XML 技术在企业及电子商务中的应用。

1.1 XML 的产生

1.1.1 XML 的发展历程

1969 年,人类首次登上月球。与此同时,IBM 公司发明了第一种现代标记语言——通用标记语言(Generalized Markup Language, GML)。1978 年,美国国家标准化组织(American National Standards Institute, ANSI)在此基础上创建了标准通用标记语言(Standard Generalized Markup Language, SGML),并且于 1986 年正式公布了它的第一个标准。SGML 是一种元语言(meta-language)——能够描述其他语言及其语法和词汇表的语言。

SGML 的功能非常强大,它已经被一些政府机构及大型公司广泛采用。但它同时又是一种复杂的语言。由于其复杂性及其实现所需要的大量资金意味着大多数商业用户和个人用户无法享受此项技术所带来的益处。

于是,20 世纪 90 年代初期,欧洲粒子物理实验室(CERN)的科学家 Tim Berners Lee 使用 SGML 定义超文本标记语言(Hyper Text Markup Language, HTML),并于 1991 年夏天公开发布。HTML 是一种面向数据显示样式的标签语言,主要描述文档的标题、字体、颜色等外观方面的属性。由于 HTML 具有简单容易使用等优点,因此已在世界范围内得到了广泛应用。但是,Web 技术的飞速发展以及 HTML 不能描述数据内容等致命弱点使得人们不再满足于现有的 HTML 技术。人们开始寻求一种能够满足新情况的标记语言。这种语言应该具有 SGML 的强大功能和可扩展性的优点,同时又应该具有 HTML 的简单、容易使用等特点。

万维网联盟(World Wide Web Consortium, W3C)于 1996 年 7 月正式成立了专门的 SGML 编辑审议委员会(即现在的“XML 工作组”)来负责此项工作。Sun 公司的 Jon Bosak 担任该委员会的主席。在 1996 年的 11 月,他们完成了 SGML 简化版本的初始设计。该简化版本保留了 SGML 语言 80% 的功能,同时又大大缩减了 SGML 的复杂性,复杂程度仅相当于 SGML 的 20%。这一简化版本就是 Web 的基础——可扩展标记语言(eXtensible Markup Language, XML)。从此以后,XML 得到了不断的发展演化。1998 年 2 月 W3C 正式公布了 XML 的 1.0 版本。Web 上又一具有深远意义的技术诞生了。

自 XML 诞生以来,其相关的各项技术和应用不断涌现,其标准和规范也在不断完善规范之中。

图 1-1 反映了 XML 标记语言的发展历程。



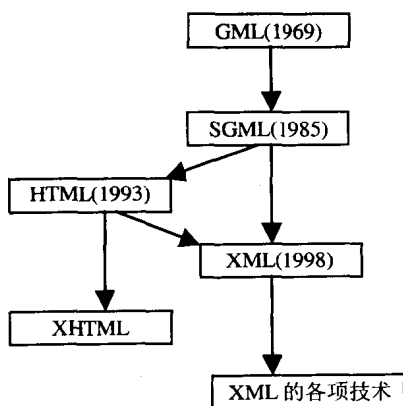


图 1-1 XML 的发展历程

1.1.2 SGML、HTML 和 XML

1. SGML、HTML 和 XML 之间的联系

既然 XML 源于 SGML，与 HTML 是一对孪生兄弟，那么它们之间有什么联系呢？

SGML 是描述电子文档的国际化标准，是用于书写其他语言的元语言。SGML 文档已经在美国军方及美国航空业使用多年。XML 和 HTML 都是从 SGML 发展而来。因此，它们都有一些共同点，例如相似的语法和均使用带括弧的标记符。

但是，HTML 是用 SGML 所定义的，而 SGML 是元语言，任何一种用元语言定义的语言称为该元语言的一种应用，因此 HTML 是 SGML 的一个应用。而 XML 是保留了 SGML 的主要实用功能的一个简化版本，或者称 XML 为 SGML 的一个子集。所以 HTML 基本上不能用来定义新的应用，而 XML 可以定义新的应用。例如，资源描述格式（Resource Description Format, RDF）和频道定义格式（Channel Definition Format, CDF）都是使用 XML 定义的应用。

同时，在设计 XML 之初，考虑到 SGML 在一些政府机构和大型公司仍在使用，因此 XML 的设计目标之一是要与 SGML 兼容。XML 文档可以通过任何 SGML 制作或通过浏览工具阅读，但是 XML 比 SGML 要简单得多。XML 摒弃了 SGML 中许多很少使用的功能。SGML 常用的部分只占 20%，XML 抛弃了 SGML 中不常用的部分，使它一下就精简了 80%。这样一来，XML 的语法说明书只有 30 页，而 SGML 的却有 500 多页。

2. XML 的特征及优点

当前在 Web 上广泛应用的 HTML 只使用 SGML 中很小一部分标记，例如 HTML 4.0 仅定义了 90 多种标记。为了便于在计算机上实现，HTML 中规定标记是固定的，HTML 这种固定的语法使它易学易用，在计算机上开发 HTML 的浏览器也十分容易。正是由于

HTML 的简单性，使 Web 技术得到飞速的发展。

近年来，随着 Web 的应用越来越广泛和深入，HTML 过于简单的语法反过来成为阻碍 Web 发展的最大障碍。主要表现在以下方面：

- HTML 缺乏可扩展能力

随着 Web 应用的广泛和深入，对数据的要求也越来越高。HTML 静态的、有限的标记集无法满足日益增长的数据描述要求，应用开发者不得不局限于浏览器厂商所提供的现有标记集。

- HTML 只能描述数据的显示样式

HTML 是一种面向描述的语言，而非面向对象的语言，因此，HTML 标记不会给出内容的含义。

- HTML 数据的可重用性差

用 HTML 描述数据时，数据与其显示样式是完全混合在一起的，这就导致了数据的可重用性不高。

- HTML 缺乏表达数据语义的能力

HTML 只是一种面向描述的语言，它不能揭示 HTML 中标记所隐含的含义。由于 HTML 缺乏表达数据语义的能力，因此它在电子数据交换（EDI）、查询数据库中的数据等诸多方面存在极大的不足。

而 XML 相对 HTML 而言，有如下特征和优点：

- XML 是可扩展的

XML 是 SGML 的一个子集，是一种元语言。因此，它可以创建自定义元素以满足实际应用的需要。允许不同的行业开发与自己行业相关的标记语言。这方面的应用如数学标记语言 MathML。

- XML 是结构化的

使用文档类型定义（Document Type Definition, DTD），XML 能够被结构化，这样内容和语法都能被验证。从而可以创建标准化而且有效的文件。

- XML 比 SGML 更容易存取数据元素

因为 XML 具有良好的结构，XML 比 SGML 更容易存取数据元素。因此程序员可以容易地编写软件来描述 XML 文档。XML 使用简单的原则来区分文档内容和 XML 标记元素。

- XML 是国际化的

由于 XML 要在全球范围内使用，所以不能局限于 7 位的 ASCII 字符集。XML 采用了 Unicode 字符集。Unicode 能够统一表示世界各国的文字。

！ 访问 <http://www.cs.tut.fi/%7Ejorpela/chars/index.html>，可以看到关于字符编码问题的一个很好的介绍。

- XML 具有强大的链接查询功能

XML 的链接查询功能同 HTML 类似，但是 XML 在功能上更强大、更具灵活性，例如

XML 可以用链接去定义相似文档的关系, 确定浏览文档的顺序, 允许链接多于两个以上的资源等一系列优点。

- XML 利于数据交换

XML 由于其能很好地表达标记的含义及其采用 Unicode 字符集等特性。因此, 它有利于在不同的系统上进行信息交流。

- XML 容易使用。

XML 比 HTML 稍微复杂一些。随着越来越多的浏览器支持 XML 以及越来越多的 XML 支持工具的大量涌现, 越来越多的开发者将从 XML 中受益。

由于 XML 表现出的上述种种优势, 面对复杂的企业需求及飞速发展的 Web 技术, XML 得到了广泛的应用。

3. XML 是否将取代 HTML

既然 XML 是一种像 HTML 一样容易学习和使用, 但同时又具有 SGML 主要功能的语言。那么, XML 是否将会取代 HTML 呢?

回答这个问题之前, 首先看看下面的 Ex1-1.html 和 Ex1-2.xml。



Ex1-1.html

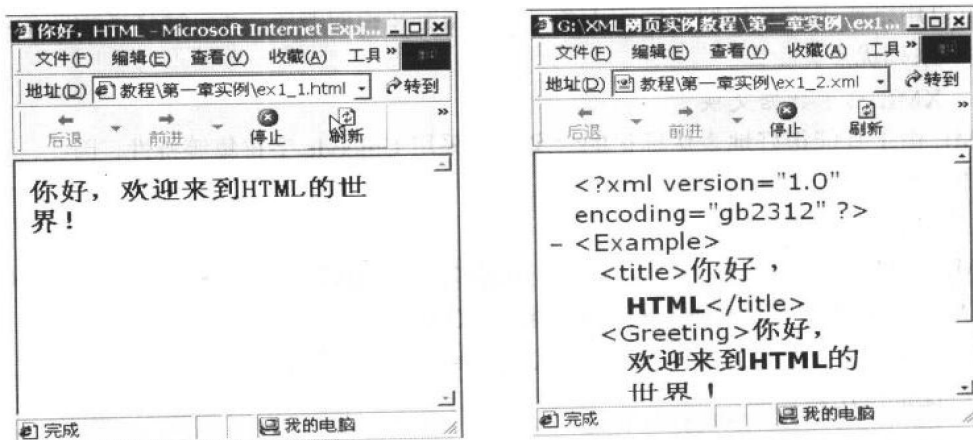
```
<html>
  <head>
    <title>你好, HTML</title>
  </head>
  <body>
    <h3>你好, 欢迎来到 HTML 的世界! </h3>
  </body>
</html>
```



Ex1-2.xml

```
<?xml version="1.0" encoding="gb2312"?>
<Example>
  <title>你好, HTML</title>
  <Greeting>你好, 欢迎来到 HTML 的世界! </Greeting>
</Example>
```

编写好上述两个实例后, 在 IE5.0 中的浏览结果分别如图 1-2 (a)、1-2 (b) 所示。



(a) 实例 Ex1-1.html 的浏览效

(b) 实例 Ex1-2.xml 的浏览效

图 1-2 XML 文档和 HTML 文档的比较

从上述两个实例可看出, XML 文档和 HTML 文档非常相似。但它们在浏览器中显示的结果却差别较大, 主要原因是 HTML 侧重于文档的显示, 而 XML 文档侧重于文档语法的表达。因此, 就目前状况而言, XML 不可能在短时间内取代 HTML。事实上, XML 和 HTML 在许多网络应用开发中, 总是协同使用。同时 XML 和 HTML 之间的联系越来越密切, 例如 W3C 于 2000 年 1 月 26 日正式公布的 HTML 的下一代——可扩展超文本标识语言 (eXtensible HyperText Markup Language, XHTML) 1.0 版本——就是一个用 XML 定义的 HTML。

表 1-1 列举了 HTML 和 XML 的差别。

表 1-1 XML 和 HTML 的比较

比较内容	HTML	XML
可扩展性	不允许用户定义自己的扩展标签	允许用户定义自己的扩展标签
结构描述	不支持深层次的嵌套表达	支持深层次的嵌套表达, 适合表示结构复杂的文档数据
可读性及可维护性	难以阅读、维护	结构清晰, 便于阅读、维护
数据和显示的关系	内容描述和表现形式合为一体	内容描述和表现形式相分离

1.2 XML 核心技术

在对 XML 的起源、发展与应用有了一个初步的认识后, 下面简要介绍 XML 的各项主要技术。

1.2.1 XML1.0 规范

当前 W3C 发布的正式的 XML 标准的版本是 1.0 第二版,它规定了如何定义 XML 文档,以及 XML 的语法规则。同时提出了两个扩展的建议:XML Linking 和 XSL。XML Linking 定义了如何用超链接来联系 XML 文档,XSL 描述了如何表示 XML 文档。

1.2.2 XML 文档定义和校验

1. 传统的 DTD

在 XML 还没有问世之前,在 SGML 中就采用文档类型定义(Document Type Definition, DTD)对文档结构进行定义。SGML DTD 提供了 SGML 的 Schema(称之为大纲或模式)来确定文档的基本框架。XML 源于 SGML,因此自然而然地借用 SGML 中的 Schema 机制——DTD 用来规定 XML 文档的结构。

但为什么在 XML 中需要 DTD 呢?一个规范的 XML 文档必须满足两方面的条件,一是必须满足格式良好的要求;二必须是一个有效的 XML 文档。格式良好的 XML 文档指的是文档必须遵守 XML 文档语法,而有效的 XML 文档指文档必须遵守 DTD。有效的 XML 文档一定是格式良好的。打个比方说,一篇文章如果没有错别字、没有病句,可称其为符合语法要求,称其为“格式良好”;但它不一定是一篇符合逻辑或者称之为“有效”的文章。例如下面的实例 Ex1-3.xml 是一个格式良好的 XML 文档,但不是一个有效的 XML 文档。



Ex1-3.xml

```
<?xml version="1.0" encoding="gb2312"?>
<Class>
  <Student>
    <Name>张三</Name>
    <!--Number 是学生的学号-->
    <Number>20</Number>
    <Number>25</Number>
    <Sex>男</Sex>
  </Student>
</Class>
```

该文档能在 IE5.0 中正常显示,但该文档只是一个格式良好的 XML 文档。原因是每个