

XML

XML

网页开发技术

XML

XML

网页开发技术

李 宋 编著

西安电子科技大学出版社

<http://www.xduph.com>



XML

XML 网页开发技术

李 宋 编著

导向科技资讯机构 策划

西安电子科技大学出版社

2001

内 容 提 要

本书以丰富的实例,详细介绍了可扩展语言XML的编程方法和应用技术。全书从一个简单的实例开始,详细阐述了XML的基本语法以及XSL、XLink和XPointer、DTD等的最新知识,还讲述了XML与JAVA的关系以及在JAVA和Visual C++中如何运用XML的有关知识。

本书既可用以辅导读者从入门开始学习XML,也可用作网络开发人员和系统维护人员的参考用书。

本书的配套光盘上附有书中的所有实例程序、JAVA和VC的接口API以及作者所用的XML工具,可与本书配合使用。

本书特点

W 简明易懂 W
W 经济适用 W
W 容易自学 W
W 提高迅速 W

图书在版编目(CIP)数据

XML网页开发技术/李宋编著,

—西安:西安电子科技大学出版社,2001.5

ISBN 7-5606-1015-3

I. X… II. 李… III. 主页制作—可扩充语言,XML—程序设计 IV. TP393.092

中国版本图书馆CIP数据核字(2001)第16519号

责任编辑:霍小齐 汪雨帆

出版发行:西安电子科技大学出版社

(西安市太白南路2号)

电 话:(029) 8227828 邮 编:710071

Http://www.xduph.com

E-mail:xdupfxb@pub.xaonline.com

经 销:新华书店

印 刷:西安兰翔印刷厂

版 本:2001年5月第1版 2001年5月第1次印刷

开 本:784×1092 1/16 印张 21.875

字 数:518千字

印 数:1~4000册

定 价:34.00元(含光盘)

ISBN 7-5606-1015-3/TP·0498

如有印装问题可调换

本书封面贴有西安电子科技大学出版社的激光防伪标志,无标志者不得销售。

前言

HTML 的出现在网络业产生了巨大的轰动：网络已经不再只是计算机业界的专门术语，而是逐渐成为整个人类的共同话题。现在，任何人都可以在互联网上发布文档、图像、声音甚至动画。人们可以在互联网上拥有令人眼花缭乱的 HTML 网页，访问热点的网站，还可以使用各种各样的多媒体效果——图像、声音甚至动画来美化自己的网页。但是，在网络飞速发展的今天，从企业的角度来看互联网上的网页就不难发现：虽然这些 HTML 的多媒体效果显示了网页的个人风格甚至个人的艺术爱好，但如果不从花哨的媒体修饰转变为企业的商务应用，一个企业或者说一个企业的网站就不可能在如火如荼的经济社会中立稳脚跟。在这样的一种情况下，XML 应运而生。

本书对各种 XML 规范和实例的论述都十分全面和详尽，其中的所有 XML、XSL 和 HTML 文档都经作者在 Microsoft 公司的 Internet Explorer 5.0 运行并通过，这些源程序代码可以在本书的配套光盘中找到。通过本书的学习，读者可以掌握目前最先进、最振奋人心的 Internet 技术。

本书的内容表述及体例约定如下：

正文：分四级标题排列；

主要内容：列出了该章的主要内容，以便于读者了解该章知识要点；

“XX”：菜单、文件、按钮。



提示

对未能在正文中表述的内容作简要提示，或是对更深内容作进一步的阐述。



注意

提醒读者注意可能出现的问题和容易犯的错误，对初学者易混淆的命令、选项、概念以及不能进行的操作和无法实现的功能或命令在此列出。



技巧

作者的经验介绍与总结，给读者指点的捷径、高招以及与其他软件配合使用的技巧。

本书由李香敏主编，李宋编著。全书由导向科技资讯机构策划、组织、校审。唐静、蒋蕾、张凯、李琦、冯明龙、宋玉霞、缪军、肖莉、杨文澜、曾雨苓、杨治国等人参与本书的排版校对工作，在此感谢参与本书工作的全体人员。由于作者水平有限，书中不当之处在所难免，敬请读者和同行批评指正。

读者在使用本书的过程中如有其他问题、意见、建议，可以访问[Http://www.dx-kj.com](http://www.dx-kj.com)或通过 dxkj@dx-kj.com、dxkj@21cn.com 与我们联系。

导向科技资讯机构

2001 年 1 月

目 录

第 1 章 XML 的过去、现在和未来 1	
1.1 XML 的过去、现在和未来 2	
1.1.1 XML 的过去..... 2	
1.1.2 XML 的现在..... 4	
1.1.3 XML 的未来..... 5	
1.2 XML 的特点 6	
1.3 XML 与其他语言的联系和区别 9	
1.4 XML 的应用程序 11	
1.4.1 数学标记语言 11	
1.4.2 化学标记语言 11	
1.4.3 通道定义格式 (CDF) 12	
1.4.4 同步化多媒体集成语言 12	
1.4.5 开放软件描述 12	
1.4.6 小结..... 13	
1.5 XML 的整体思想 14	
1.5.1 数据结构和名域 14	
1.5.2 数据发送处理 15	
1.5.3 解析 XML..... 15	
1.5.4 使用文档对象模式 (DOM) 处理和编辑数据 15	
1.5.5 用 HTML 显示 XML 数据 16	
1.5.6 增加 HTML 16	
1.5.7 转换和查询 XML..... 16	
1.5.8 设置字符和编码 17	
1.5.9 空格符..... 17	
1.6 XML 的前景 17	
1.6.1 XML/EDI 电子数据交换..... 18	
1.6.2 化学标记语言 CML 和 数学标记语言 Math ML..... 18	
1.6.3 开放式软件描述格式 OSD 18	
1.6.4 通道定义格式 CDF 18	
1.6.5 开放式财务交换 OFX..... 18	
1.6.6 发展现状 19	
第 2 章 XML 快速入门 21	
2.1 第一个 XML 文档..... 22	
2.1.1 创建一个简单的 XML 文档 22	
2.1.2 XML 程序的组成 23	
2.2 XML 的样式含义 25	
2.3 为 XML 文档准备样式表..... 26	
2.4 重新浏览 XML 文档..... 28	
2.4.1 将 XML 静态地转 变成 HTML..... 28	
2.4.2 将 XML 动态地转 变成 HTML..... 30	
2.5 编写一个大型的 XML 及 样式表..... 31	
2.5.1 编写 3.xml 的源程序 代码文档 31	
2.5.2 编写用于 3.xml 的样 式表 3.xsl 文档..... 33	
第 3 章 XML 语法 39	
3.1 XML 的简单定义 40	
3.1.1 XML 的设计和目標 40	
3.1.2 XML 术语 40	
3.2 XML 文档 41	
3.2.1 结构性 XML 文档 41	
3.2.2 字符 42	
3.2.3 通用语法结构 42	
3.2.4 字符数据和标记 43	

3.2.5 注释的用法.....	43	4.10 正规有效的 XML 文档.....	73
3.2.6 进程指令.....	43	4.10.1 格式良好的 XML 文档.....	73
3.2.7 CDATA 节说明.....	43	4.10.2 有效的 XML 文档.....	74
3.2.8 序言和文档类型声明.....	44	4.11 结构完整的 XML 文档.....	76
3.2.9 外围设备文档声明.....	45	4.11.1 起始标签和结束标签 必须匹配.....	76
3.2.10 空格处理.....	45	4.11.2 元素不能交迭.....	76
3.2.11 行尾处理.....	46	4.11.3 XML 标签对大小写 是敏感的.....	76
3.2.12 语言确认.....	46	4.11.4 表示空元素.....	77
3.3 XML 的逻辑结构.....	47	4.11.5 保留字符.....	77
3.3.1 开始标记、结束标记 和空标记.....	47	4.11.6 每个 XML 文档必须 有惟一的根元素.....	77
3.3.2 元素类型声明.....	47	4.11.7 一个完整的 XML 文档.....	77
3.3.3 属性列表声明.....	48	4.12 数据岛.....	81
3.3.4 条件节.....	49	4.13 一个实际的大型 XML 文档.....	83
3.4 XML 的物理结构.....	50	第 5 章 XSL 详解.....	97
3.4.1 字符实体参考.....	50	5.1 XSL 简介.....	98
3.4.2 实体声明.....	51	5.1.1 使用 XSL 处理器.....	98
3.4.3 语法分析的实体.....	52	5.1.2 XSL 的工作流程.....	99
3.4.4 XML 处理器处置实体 和参考.....	53	5.1.3 创建样式表 XSL.....	101
3.4.5 构造内部实体替代文本.....	55	5.2 XSL 中的 HTML.....	102
3.4.6 预定义实体.....	56	5.3 样式属性.....	103
3.4.7 标注声明.....	56	5.3.1 样式继承性.....	105
3.4.8 文档实体.....	56	5.3.2 样式选择.....	105
3.5 XML 处理器.....	57	5.4 选择目标.....	110
3.5.1 验证和非验证处理器.....	57	5.4.1 根规则.....	110
3.5.2 使用 XML 处理器.....	57	5.4.2 子元素和父元素.....	111
3.6 标注.....	58	5.4.3 通配符.....	112
第 4 章 XML 程序设计.....	59	5.4.4 属性.....	113
4.1 正规的 XML 文档.....	60	5.4.5 POSITION 位置.....	114
4.2 XML 的声明.....	61	5.4.6 冲突解决方案.....	116
4.3 XML 的注释.....	61	5.5 执行动作.....	117
4.4 XML 的实体参考.....	63	5.5.1 增加内容.....	117
4.5 XML 的 CDATA 节.....	65	5.5.2 选择.....	117
4.6 XML 的标记.....	67	5.6 MACRO 宏.....	118
4.7 属性.....	69	5.7 引入样式表.....	119
4.8 DTD 文档类型定义.....	70	5.8 样式规则.....	120
4.9 名域.....	71		

5.9 命名样式	121
5.10 在 XML 标记中包括样式	121
5.11 模式	122
5.12 JavaScript 语言脚本	123
5.12.1 作为属性值的脚本	124
5.12.2 EVAL	124
5.12.3 函数声明	124
5.12.4 XML 对象模型	125
5.12.5 内置函数	129
5.12.6 SCRIPT 脚本标记	131
5.13 连接到 XSL 样式表	132

第 6 章 XLink 和 XPointer

6.1 XLink 和 XPointer 简介	134
6.2 详解 XLink	135
6.2.1 一般的链接	135
6.2.2 扩展链接	137
6.3 详解 XPointer	139
6.3.1 绝对位置项	140
6.3.2 相对位置项	143
6.3.3 字符串匹配项	146
6.3.4 扩展链接组	146

第 7 章 DTD 详解

7.1 DTD 简介	148
7.2 DTD 与 Schema 的优势	149
7.2.1 DTD	149
7.2.2 Schema	149
7.3 用 DTD 验证文档	150
7.4 研究文档的结构	150
7.5 建立 XML 文档的 DTD	152
7.5.1 XML 文档的合法性	152
7.5.2 元素类型声明	153
7.5.3 规范子元素	160
7.5.4 使子元素可选	167
7.5.5 标记零个或者多个子元素	170
7.5.6 一个或者多个子元素	176
7.6 合并元素	171

7.6.1 允许作者选取的元素	181
7.6.2 嵌套括号	182
7.6.3 使用混合内容	184
7.7 空标记	194

第 8 章 XML 与 JAVA

8.1 XML 与 JAVA 的关系	200
8.2 JAVA 简介	201
8.2.1 类和对象	201
8.2.2 什么叫对象	201
8.2.3 什么是类	201
8.3 Java.ms.xml.om.Document 类	202
8.4 Java.ms.xml.om.Element 类	203
8.5 创建一个 XML 文档	203
8.6 创建访问 XML 文档的 控制台程序	206
8.6.1 启动 Visual J++	206
8.6.2 创建一个基于控制台的 应用程序	207
8.6.3 加入代码	207
8.6.4 编写完全的 JAVA 代码	210
8.6.5 运行 JAVA 程序	211
8.7 创建 JAVA 基于 Windows 的程序	213
8.7.1 启动 Visual J++	213
8.7.2 创建一个基于 Windows 的应用程序	213
8.7.3 加入代码	216
8.8 创建基于 Form 的 Windows 程序	225
8.8.1 创建一个基于 Form 的 应用程序	225
8.8.2 修改默认的界面风格	228
8.8.3 加入代码	231
8.8.4 全部代码	233
8.8.5 运行结果	237
8.9 在 XML 文档中查找元素	238
8.9.1 创建 XML 文档	238
8.9.2 生成程序框架	241

8.9.3 生成元素对象	246	第 10 章 FOML 应用实例	
8.10 数据库管理器	253	MathViewer	315
8.10.1 创建 XML 数据文档	253	10.1 FOML 和 MathViewer	316
8.10.2 创建 JAVA 程序框架	256	10.1.1 概述	316
8.10.3 实现管理功能	257	10.1.2 远程教育系统的开发	
8.10.4 实现添加记录	259	与 FOML 的研究	316
8.10.5 全部源程序代码	263	10.2 FOML 的语法	318
第 9 章 XML 与 Visual C++	275	10.2.1 FOML 语言语法概述	318
9.1 XML 与 Visual C++	276	10.2.2 FOML 中的数学符号	319
9.1.1 Visual C++ 与 XML		10.2.3 FOML 语言巴克斯范	
的关系	276	式定义	323
9.1.2 DOM、SAX 与 COM+		10.3 FOML 的使用	324
技术	277	10.3.1 能量公式的 FOML 表达	324
9.2 MSXML3	279	10.3.2 几个简单的实例	325
9.3 生成一个 XML 文档	280	10.3.3 综合实例	327
9.4 创建 XMLControler		10.4 MathViewer	329
二程	282	附录	331
9.4.1 启动 Visual C++	282	附录 A XML 专业术语	332
9.4.2 创建一个工程	282	附录 B ASCII 字符集	336
9.4.3 修改界面的风格	286	附录 C XML 网络资源	338
9.4.4 修改代码	287	附录 D 关于配套光盘	340
9.4.5 运行程序	312	附录 E 参考文献	341

第1章

XML 的过去、现在和未来

主要内容

- XML 简介
- XML 与其他语言之间的联系和区别
- XML 的特点
- XML 的应用及 XML 应用程序

本章主要介绍 XML 的发展过程。网络业的出现使网络已经不再只是计算机业界的专门术语，而是逐渐成了整个人类的共同话题。现在，任何人都可以在互联网上发布文档、图像、声音甚至动画。本章还介绍了 XML 与 SGML、HTML、JAVA 等其他语言之间的联系和区别，论述了 XML 的应用前景。XML 是一个新生事物，作为计算机的学习者，我们需要的是新的知识、新的想法、新的理论，我们不必恐惧新的事物，况且 XML 也并非洪水猛兽。

1.1 XML 的过去、现在和未来

在科学技术高速发展的今天,网络已经不再是计算机业界的专门术语,而是逐渐成为整个人类的共同话题,网络技术已渗透到了社会生活的每一个角落。要使网络平易近人并为大众所接受,单靠 TCP/IP 是远远不够的。由此,HTML——超文本标识语言便应运而生,简单精练的语法、极易掌握的通用性与易学性,使 Web 网页可以亲近于每一个人。

目前,几乎所有的 Web 页面都是用 HTML 编写的。尽管 HTML 简单易学,但是随着 Web 应用的日趋广泛,HTML 过于简单的弱点也更加明显了。而且,在许多复杂的 Web 应用中,HTML 遇到了强大的挑战,要解决这些问题,人们力不从心。因而,运用功能强大的 XML 来替代 HTML 作为 Web 页面的书写工具便成为一种不可阻挡的趋势。

电子商务、电子图书、远程教育等领域异军突起,迅猛发展并逐渐成为互联网世界必不可少且愈发重要的组成部分,随之而来的是 Web 文件的复杂化、多样化、智能化。于是,高容量、高信息量、高效率便成为网络信息传输技术发展的要求。与此同时,另一种需求也变得更加广泛而迫切,那便是:同样的数据能否根据不同用户的不同需求而以不同的效果和表达形式再现出来。举一个简单的例子:假设我们要在网上拍卖一幅油画,而在一个 Web 文件中包含了有关这幅油画的全部信息,包括油画名称、油画简介、作者名称、作者简介、拍卖行信息、拍卖底价、各竞价方名称、各竞价方出价、最高价等。在客户端有各种用户,如拍卖行的经纪人,他并不需要任何有关油画本身的信息,而只需看到一个包含所有竞价方及其报价的统计列表;而那些竞价方则希望看到有关油画的资料,他们有的喜欢作者,有的喜欢油画的内容,有的则只喜欢画的色彩,等等;另外还有一些“穷光蛋”,他们只想过过眼瘾,看看油画即可;也许还有一些“别有用心”的人,他们根本就不关心什么油画、拍卖,只想让计算机接收到这个文件后能够自动地将其分门别类,放到自己与之相对应的数据库中。这样一来,这个 Web 文件必须具有极高的复杂性与灵活性,以应付如此复杂多变的具体应用。HTML 所欠缺的恰恰就是这一点,它的简单与普遍性原则虽曾使 Web 的推广受益匪浅,但反过来也成为其自身发展的最大桎梏,尽管人们已付诸很多努力——包括各式各样的修改扩充,如增加表格、框架、脚本语言等,但均未能从根本上改变 HTML 的局限性。于是人们开始全力重新构造 Web,并相继出现了许多新的技术,但由于种种原因大都没能普及下来,惟有一项得以迅速发展,不仅在使用中得到各方肯定,而且在短短两年内迅速崛起,并得到 SUN、Microsoft、IBM 等各大公司的全力支持,竞相开发,并形成了一股不可遏制的席卷全球的浪潮,这便是 XML——可扩展标识语言。

1.1.1 XML 的过去

HTML 的出现,在网络业产生了巨大的轰动。从企业的角度来看,虽然这些 HTML 的多媒体效果显示了网页的个人风格甚至个人的艺术爱好,但是如果不从花哨的媒体修饰转

变为企业的商务应用，企业的网站不可能在如火如荼的经济社会中站稳脚跟。在这样的一种条件和时机之下，XML 便应运而生了。

也许您还在怀疑，也许您正在为顺应计算机业的发展潮流而在做必要的准备，也许您正在为如何快速地掌握网页编程语言而踌躇不前，那么，您应该知道：在如今，软件业中许多具有影响力的公司都在致力于使 XML 成为下一代互联网的发展方向，所以选择 XML 作为网页编程语言必是一个明智的选择。

XML 作为一种新的核心技术已经存在了数十年，并且一直在成千上万个工程应用中使用，它的正确性已毋庸置疑，在 XML 中，数据处理和文档处理基本上是同一类东西，只不过是 XML 中加入了编程语言的语法概念罢了。

下面，让我们回顾一下 XML 的前身——SGML 的发展史。

早在 20 世纪 60 年代后期，人们使用的系统是各自独立的，每一个系统使用不同的命令和语法，它们所生成的文档当然也是互不兼容的。SGML 的发明者——Charles F Goldfarb 和他的同事开始着手解决这个问题。他们经过研究后得出三条理论：各个程序都必须支持一种公共文档表示形式；公共格式必须专用于某一文档；要使计算机系统可靠地处理文档，就必须遵循某些共同的规则。早在 1969 年，Goldfarb 证明了“进行合法性检查的语法分析器”的可行性，也就是说 SGML 已经初具雏型了。它可以在不处理文档的前提下，用语法分析器检查标识的正确性，这与我们现在使用的 SGML 已经相差不远了。在 1978 年至 1986 年间，Goldfarb 领导一个技术开发小组，把他初期的成果进一步完善而为一个国际标准——ISO 8879，即“标准通用标识语言”（SGML）。

虽然 SGML 标准的制定花费了大量的时间，但是在它发布时还是超前于当时的市场。在 1986 年 SGML 被标准化的时候，它已经变得相当复杂了，功能也很强大。除了国际标准之外，SGML 还是一种大型复杂的文档交换标准，而且还被应用于各种各样的领域，如企业应用、商务应用和语言设计等。

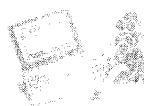
但是在后来的使用中，格式化标记对万维网的相互操作性和伸缩性产生了很大的束缚。为此，万维网协会（W3C）通过 3 种方式来解决这一问题：第一，使用样式表。他们发明了一种简单的专门用于 HTML 的样式表语言——“级联式样式表”（CSS）；第二，在 HTML 中加入抽象内容的简单机制。由于 HTML 的扩展性和 CSS 都不能彻底地改变现状，于是他们又使用了第三种方式，即开发一个 SGML 的子集，在保持 SGML 优点的同时，又符合万维网的规则——使用简单。这个子集就是 XML——Extensible Markup Language，即扩展标识语言。

以上就是 XML 的起源及其来历，与其他“新生代”的产品不同的是，XML 是基于 SGML 的一个特定的子集（如图 1-1 所示）。

SGML 全集

XML

图 1-1 XML 和 SGML 的关系图



注意

XML 是基于 SGML 的一个特定的子集。

1.1.2 XML 的现在

在互联网飞速发展的今天,XML 的出现给万维网添加了巨大的活力。在 XML 的网站内,人们可以更加自由、亲切地交流,万维网不但成为我们休闲的好去处,更重要的是其中蕴藏着无限的商机。XML 的出现,使计算机业的各大商业机构敏锐地感到它将改变人们对数据的看法,也将改变人们对万维网的看法。XML 已经影响了从文字处理、电子表格到数据库管理系统、电子邮件等所有类型的应用,因此万维网将出现巨大改变,数据的互相访问和共享将变得很轻松自如,这将成为新网络时代的标志。

1. 不完美的 HTML

比尔·盖茨在 1996 年 2 月说:“HTML(超文本链接标识语言)是我们的数据类型。”

这么多年的发展让我们更加清楚地看到,HTML 只是一种表达的技术。它并不一定能揭示 HTML tag 的含义。举一个最简单的例子,<h2>Apple</h2>这句话在网络浏览器中有特定的表现。但是 HTML 却并没有告诉我们它到底是什么。Apple 只是一个英文单词罢了,它在不同的环境之下可能会有不同的意义,是一个计算机公司、一个水果,还是一个姓氏?HTML 并没有告诉我们 Apple 的具体含义。

在 HTML 中有一个大问题就是它的 tag 集合是固定的。用户不能新增供他人使用的 tag。网络浏览器是一个应用平台,以 HTML 作为数据标准,网络上的应用程序依赖于服务器上的 CGI(公共网关接口)脚本来处理网页上的数据。这样的结构虽然非常清晰,但 HTML 却使服务器一端要处理的数据量过大,以致降低网络速度与效率。

2. 完善的 XML

1996 年秋天,Microsoft 的众多开发小组都在寻求一种可以在网上进行交互操作的开放文档格式。这种格式就是 XML 中的一个应用“频道定义格式”(Channel Definition Format,简称 CDF)。此后的数月里,几乎所有的 Microsoft 小组都使用 XML 编写应用程序,他们所需要的正是 XML 那种简单的数据表示语法。1997 年 10 月,软件业的天才——Bill Gates 宣布:XML 是“一个突破性的技术”,相信有了 Microsoft 的支持,XML 的发展将没有后顾之忧。

现在乃至将来,XML 都是一块令人向往的沃土。实际上,除了 Microsoft 外,各大软件商都纷纷表示了对 XML 的支持。其中,Borland 公司宣布其 Inprise Delphi 5 对 XML 的支持将在其后以三个版本出现。数据库的专家——Oracle 公司则免费推出常用编程语言 Java、C、C++、PL/SQL 的 XML 接口,Oracle XML Utilities 和 XSQL Servlet 产品可用来



对 Oracle 数据库读出或写入 XML 数据;此外, Oracle 还推出了 Oracle XML Praser for Java 的 2.0 版本, 包括 Extensible Stylesheet Language Transformation 处理器。另外, IBM 公司的 AlphaWorks 部门发布了 EasyXML Bean Suite, 其中包括 XMLHolder、XMLElement、XMLAttribute 和 XMLFileGenerator Java beans, easyXML beans 允许 Java 的应用程序处理 XML。Vertex Industries 公司宣布了支持 IBM MQ Series 的接口, 而且该公司已经计划推出下一代中间软件工具的新部件, 这个名为 Vixen 的部件为信息在 XML 文件和有效数据结构之间的传输和翻译提供了有效的手段。经过演变的 XML 支撑技术和它的跨平台功能使得在不同环境中运作的应用程序可以互相交换数据。

另外, IBM、SUN、LOTUS、JAVA、Commerce One 等各大公司都纷纷推出了支持 XML 的部件。XML 之所以能旋风式地崛起, 与各种商业机构的支持是密不可分的。

1.1.3 XML 的未来

众所周知, XML 的发展是属于未来的。尽管 XML 是一个非常新的网络语言, 但是作为一个计算机的学习者来说, 我们需要的是新的知识、新的想法、新的理论, 而不必恐惧新的事物, 况且 XML 也并非洪水猛兽。

网上有一句名言: 网络的未来是属于商业的。未来的商业操作已经不是简单的网页广告、有限的信息反馈、狭窄的交易空间, 取而代之的是多元化的服务、友好的界面、丰富的数据共享等。XML 需要强大的新工具以在文档中显示丰富且复杂的 XML 数据, 今后我们可以在动态变化的分层数据上映射用户友好显示层来实现这一目的。

到那时, 顾客服务会从电话购物或亲自到商场去选购自己的商品而转移到 Web 站点上来。我们相信: 在不久的将来, 我们完全可以在家里通过计算机网络进行购物, XML 的强大功能将使顾客受益匪浅。由于大多数商业应用软件包括数据的处理和转移, 比如订货单、发货单、顾客信息、合同资料和图纸等, XML 将会改变终端用户在 Internet 上的行为而使许多商业应用能够实现。另外, 通过使用基于 XML 的面向企业内部互联网的词汇库, Web 上的信息, 无论是储存在文档中的还是储存在数据库中的, 都可以被标识。这些词汇也能够对那些需要在顾客和供应商之间交换信息的中小型企业提供帮助。

XML 作为表示结构化数据的一个工业标准, 为组织、软件开发者、Web 站点和终端使用者提供了许许多多的有利条件。这些有利条件使更多的纵向市场数据格式得以建立, 并被应用于关键市场, 诸如高级的数据库搜索、网上银行、医疗、法律事务、电子商务和其他领域。当站点不仅仅是提供数据浏览而是更多地进行数据分发时, XML 语言就大显身手了。

另外, 一个亟待开发的市场是开发建立 Web 站点的工具, 包括用来从数据库信息和使用者界面中产生 XML 数据的工具。开发用来描述从数据库中产生 XML 的可视化工具具有很好的市场前景。

当然, 为了了解 XML 的过去、现在和将来, 您可能还在想象 XML 对网络的深远意义。XML 为网络提供了全新的构思, 注入了新的活力。而网络无疑是一个充满挑战、机会、机遇和传奇的虚拟的世界, 在这里, 没有永远的赢家, 也没有永远的失败者, 只要您能把握机会, 奇迹就会在您身上出现。



1.2 XML 的特点

让我们首先来看一段 XML 的简写代码：

```
<person id="let us do from here" sex="M">
  <name>
    <surname>Song</surname>
    <lastname>Li</lastname>
  </name>
  <birth>
    <data>Oct 7th, 1976</data>
  </birth>
  <someotheritem>
    <someotheritem_sub>Something is here now!</someotheritem_sub>
  </someotheritem>
  <someotheritem_2>
    <someotheritem_sub_2>Something else is here now!</someotheritem_sub_2>
  </someotheritem_2>
</person>
```

即使一个不熟悉 XML 的人，也可以从以上的 XML 代码中明白该段程序的意义。

可以说，XML 是万维网新的通用语言，XML 的出现必然有其优点和发展前途。总的说来，XML 具有以下几个明显的特点：

1. 更有意义的搜索

数据可以被 XML 惟一地标识。如果没有 XML，搜索软件必须了解每个数据库是如何构建的，这实际上是不可能的，因为每个数据库的描述数据都是不同的。而有了 XML，书籍就可以很容易地以标准的方式按照作者、标题、ISBN 序号或其他的标准分类，对书的搜索也就变得十分方便了。

2. 开发灵活的 Web 应用软件

XML 数据一旦建立，它就能被送到其他应用软件、对象或者中间层服务器中作进一步的处理，也可以被发送到桌面用浏览器浏览。XML 和 HTML、脚本、公共对象模式一起为灵活的三层 Web 应用软件的开发提供了所需的技术。

3. 不同来源的数据集成

现在要搜索多种的不兼容数据实际上是不可能的。XML 能够使不同来源的结构化数据容易地结合在一起。软件代理商可以在中间层的服务器上对从后台数据库和其他应用软件发送来的数据进行集成。然后，数据就能够被发送到客户或其他服务器，以进一步的集成、处理和分发。

4. 数据交互性好

XML 的扩展性和灵活性允许它描述不同应用软件中的数据，这些数据包括从描述搜集 Web 页到数据记录。同时，由于基于 XML 的数据是自我描述的，因此数据不需要有内部描述就能被交换和处理。

5. 本地计算和处理

XML 格式的数据被发送给客户后，客户可以用应用软件解释数据，并对数据进行编辑和处理。使用者可以用不同的方法处理数据，而不仅仅是显示它们。XML 文档对象模式 (DOM) 允许用脚本或其他编程语言处理数据，数据计算不需要回到服务器就能进行。使用简单、灵活、开放的格式，分离使用者观看数据的界面，可以给 Web 创建功能强大的应用软件，这些软件原来只能建立在高端数据库上。

6. 数据的多层显示

数据发到桌面后，能够用多种方式进行显示。通过以简单开放扩展的方式描述结果化的数据，XML 补充了 HTML 用以广泛地描述使用者界面。HTML 描述数据的外观，XML 则描述数据本身。由于数据显示内容分开，XML 定义的数据允许指定不同的显示方式，使数据更加合理地表现出来。CSS 和 XSL 为数据的多层显示提供灵活的机制。

7. 粒状地更新

通过 XML，数据可以粒状地更新。每当一部分数据变化后，不需要重发整个结构化的数据。变化的元素必须从服务器发送给客户，变化的数据不需要刷新整个使用者的界面就能够显示出来。而现在的情况是：只要一条数据变化了，整一页都必须重建，这就严重限制了服务器的升级性能。



8. 在 Web 上发布数据

由于 XML 是一个开放的基于文本的格式,它可以和 HTML 一样使用 HTTP 进行传送,不需要对现存的网络进行变化。

9. 升级性

由于 XML 彻底把标识的概念同显示分开,处理者能够在结构化的数据中嵌套程序化的描述以表明如何显示数据。这令人难以置信的强大机制使得客户计算机同使用者间的交互作用尽可能地减少,同时也减少了服务器的数据交换量和浏览器的响应时间。另外,XML 使个人的数据只能通过更新的布告发生变化,减少了服务器的工作量,大大增强了服务器的升级性能。

10. 压缩性

XML 的压缩性能很好,因为用于描述数据结构的标签可以重复使用。XML 数据是否要压缩不仅要根据应用来定,而且还取决于服务器与客户之间的数据传递量;另外,XML 能够使用 HTTP 1.1 中的压缩标准。

11. 开放的标准

XML 使用的标准是 Web 优化过的。微软和其他一些公司以及 W3C 中的工作组正致力于确保 XML 的互用性,以为开发人员、处理人员和不同系统和浏览器的使用者提供支持,并进一步发展 XML 的标准。

XML 所包含的一套相关的标准是:

- 可扩展标识语言 (XML) 标准;
- XML 名域标准;
- 文档对象模式 (DOM) 标准;
- 可扩展类型语言 (XSL) 标准;
- 可扩展连接语言 (XLink) 标准和 XML 指针语言 (XPointer) 标准是当前的工作草案。

12. 微软产品的支持

微软 IE 4.0 以上产品支持通用的解析器、XML 对象模式和 XML 数据源对象。IE 5.0 和 Office 2000 支持许多添加的 XML 特性和技术。虽然微软支持的产品不一定能够取得成功,但是没有微软支持的产品却是很难取得成功的。



13. 新的机会

作为表示结构化数据的工业标准, XML 为组织者、软件开发者、Web 站点和终端使用者提供了许多有利条件。更多的纵向市场数据格式建立了起来, 并应用于市场, 诸如高级的数据库搜索、网上银行、医疗、法律事务、电子商务和其他领域, 这都使其机会进一步地扩大了。



提示

从根本上讲, XML 是一种文档格式。它是由一系列有关结构和文本的规则组成, 具有结构性和有效性。

目前, 顾客服务正从电话和地理位置转移到 Web 站点上来, 而且将来会由于 XML 的强大功能而更加方便。由于大多数商业应用软件包含数据的处理和转移, XML 将会改变终端用户在 Internet 上的行为, 另外, 通过使用基于 XML 的面向企业内部互联网的词汇库, Web 站点上的信息, 无论是储存在文档中还是在数据库中, 都可以被标识。这些词汇也能够对那些需要在顾客和供应商之间交换信息的中小型企业提供帮助。

1.3 XML 与其他语言的联系和区别

通过前面的描述可以知道, XML 是基于 SGML 的一个特定子集。SGML 的全称是 Standard Generalized Markup Language, 即通用标识语言标准。它是国际上定义电子文件结构和内容描述的标准。HTML 的全称是 HyperText Markup Language, 即超文本标识语言, 它是 SGML 在网络上的一个特殊应用。XML 是 SGML 的一个子集, 它和 HTML 都是从 SGML 发展而来的文档形式, 其间有着一些共同之处, 如相似的语法和带括弧标记符号的使用。有一点必须清楚: HTML 只是 SGML 的一个应用, 而 XML 是 SGML 的一个子集, 它只是省略了 SGML 中一些复杂和不常用的部分。HTML 不能用来定义新的应用, 而 XML 则可以, 这就是 XML 与 HTML 之间最大的区别, 也是本质区别。例如 Resource Description Format (DRF) 和 Channel Description Format (CDF) 就是使用 XML 定义的具体应用。与 HTML 相比较, XML 具有以下优点:

- 可以用 XML 设计自己的文件类型;
- 信息的内容更加丰富, 使用更加方便。在 XML 中超文本链接的能力比 HTML 更强;
- XML 使得浏览器的表现形式更加丰富;
- 处理 XML 文件比处理 SGML 文件更简单, 信息的开放性和重复使用能力也有很大提高;
- 合法的 XML 文件在 SGML 中也是合法的。