

Web 开发

实用技术基础

刘於勋 主 编
韩 璐 胡江汇 苏小玲 副主编



清华大学出版社

Web 开发

实用技术基础

主 编 王 强 副 主 编 王 强 王 强
编 者 王 强 王 强 王 强 王 强 王 强



清华大学出版社

Web 开发实用技术基础

刘於勋 主编

韩璐 胡江汇 苏小玲 副主编

清华大学出版社

北 京

内 容 简 介

本书全面、详细地介绍了 Web 开发实用技术基础知识, 内容包括 Web 的相关概念 (Web 服务、Web 数据库、ASP.NET 技术、JSP 技术、Ajax 技术、语义 Web 服务技术、Web3D 技术)、HTML 及 XML 基础、网页与网站设计基础、Dreamweaver CS4 工具、Web 客户端脚本语言设计 (JavaScript 语言基础、浏览器对象模型 BOM、文档对象模型 DOM、Ajax 技术)、基于 ASP.NET 的服务器端程序设计以及基于 JSP 的服务器端程序设计等。

全书采用基本概念与实际应用相结合的方式组织内容, 使读者能够在深刻理解和全面领会 Web 开发程序设计特点和风格的基础上, 切实掌握面向对象的 Web 开发技术。

本书通俗易懂、结构合理、叙述简洁, 每章都有小结和思考题, 可帮助读者尽快掌握所学知识。本书可作为高等院校计算机或其他相关专业开设 Web 开发技术课程的教材, 也可作为自学 Web 开发技术的参考书。

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

Web 开发实用技术基础/刘於勋主编. —北京: 清华大学出版社, 2010.12

ISBN 978-7-302-23930-7

I. ①W… II. ①刘… III. ①主页制作-程序设计 IV. ①TP393.092

中国版本图书馆 CIP 数据核字 (2010) 第 195437 号

责任编辑: 郭新义 朱 俊

封面设计: 张 岩

版式设计: 牛瑞瑞

责任校对: 王 云

责任印制: 杨 艳

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

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

印 刷 者: 北京密云胶印厂

装 订 者: 三河市李旗庄少明装订厂

经 销: 全国新华书店

开 本: 185×260 印 张: 22.5 字 数: 516 千字

版 次: 2010 年 12 月第 1 版 印 次: 2010 年 12 月第 1 次印刷

印 数: 1~4000

定 价: 38.00 元

产品编号: 037333-01

前 言

由于 Internet 采用超文本和超媒体的组合方式将信息的链接扩展到整个 Internet 上,使得人们足不出户通过浏览器就能够访问到 Internet 上丰富多彩的网络资源。而 Web 就是一种超文本信息系统,它使文本不再固定在某一个位置,而是可以从一个位置跳到另外的位置。本书介绍了 Web 实用技术,其内容主要包括 Web 技术概述、Web 页面的基础知识(HTML 和 XML)、网页制作工具(Dreamweaver CS4)的操作方法、JavaScript 脚本在 Web 页面中的应用、运用 APS.NET 技术、JSP 技术在服务端开发应用程序等。

本书适合学过面向对象程序设计的读者使用。因为无论是客户端程序设计(JavaScript 脚本语言),还是服务器端程序设计(ASP、JSP),都是采用面向对象程序设计方法进行设计的,其目的是使读者学会使用类编程,领会面向对象程序设计的优势,进而牢固掌握面向对象程序设计的各种方法。

本教材的特点如下:

1. 组织结构新颖

本书首先介绍 Web 的相关概念、Web 服务、Web 数据库、ASP.NET 技术、JSP 技术、Ajax 技术、语义 Web 服务技术、Web3D 技术等,使读者首先对 Web 开发技术有比较系统、全面的了解和认识,并对 Web 技术产生兴趣。

2. 突出 Java 面向对象设计

全书紧紧围绕与 Java 面向对象相关的理论知识由浅入深地展开,由 JavaScript 脚本语言、Java 基础介绍,过渡到 JSP 编程,突显 JSP 服务器端程序的设计方法,重点介绍 JSP 的概况、JDK 的获取和安装、Tomcat 服务器的下载和配置、JSP 开发环境 Eclipse 的配置、JSP 的基础知识、JSP 的内置对象、4 种 JDBC 驱动程序(分别为 JDBC-ODBC Bridge、JDBC-NativeAPI Bridge、JDBC-middleware 和 PureJDBC-Driver)、DriverManager 类方法、使用 JDBC 连接 4 种数据库的方法、用 JavaBean 封装服务器数据库等内容,最后用一个会员管理系统作为范例,详细介绍如何在 Eclipse 环境下编写 JSP 动态网页。

3. 示例通俗易懂

书中所举示例实用、易理解、易掌握,均已通过上机调试、运行。

本书由河南工业大学信息科学与工程学院刘於勋主编,韩璐、胡江汇、苏小玲为副主编。参与编写的还有高山和孙宜贵等。其中第 1、5 章由刘於勋编写,第 2 章由苏小玲编写,第 3 章由韩璐编写,第 4 章由高山编写,第 6 章由孙宜贵编写,第 7 章由胡江汇编写。全



书由刘於勋统稿。

本书在编写的过程中得到了兄弟院校和同行们的支持和关心，他们对本书的编写提出了许多宝贵的建议和意见，在此对审定本教材编写大纲的专家们及在全书编写过程中给予帮助的同事们表示感谢。

由于编者水平有限，加之时间仓促，书中难免有不妥之处，敬请广大专家、读者批评指正。

编 者

目 录

第 1 章 Web 概述	1
1.1 Web 概念	1
1.1.1 Web 定义	1
1.1.2 Web 的五要素	1
1.1.3 Web 特性	6
1.2 Web 工作原理	6
1.3 Web 服务	7
1.3.1 什么是 Web 服务	7
1.3.2 Web 服务技术优势	12
1.3.3 Web 服务技术的研究方向及发展趋势	13
1.4 Web 技术	15
1.4.1 Web 数据库	15
1.4.2 ASP.NET 技术	18
1.4.3 JSP 技术	19
1.4.4 Ajax 技术	20
1.4.5 语义 Web 服务技术	22
1.4.6 Web3D 技术	25
1.5 小结	28
1.6 思考题	28
第 2 章 HTML 及 XML 基础	29
2.1 标记语言的发展历程	29
2.1.1 SGML	29
2.1.2 HTML	29
2.1.3 XHTML	30
2.1.4 XML	30
2.1.5 DHTML	30
2.1.6 SHTML	31
2.2 超文本标记语言 HTML	31
2.2.1 HTML 文件的页面结构	31
2.2.2 HTML 的基本标签	32
2.2.3 超链接	33



2.2.4	列表	35
2.2.5	表格	36
2.2.6	表单	46
2.2.7	框架	47
2.2.8	图像	49
2.2.9	文本格式及其他	51
2.2.10	修饰字体	52
2.2.11	网页设计	54
2.3	扩展标记语言 XML	56
2.3.1	XML 基础	56
2.3.2	XML 文档类型定义	57
2.3.3	XML 数据的底层结构	59
2.3.4	XML 文件的设计	64
2.3.5	XML 与 Java	65
2.3.6	XML 与 .NET	65
2.3.7	XML 应用实例	65
2.4	小结	70
2.5	思考题	70
第 3 章	网页与网站设计基础	71
3.1	网页的基础知识	71
3.1.1	什么是网页	71
3.1.2	网页的基本元素	72
3.1.3	网页设计原则	73
3.2	常用的网页制作工具	74
3.2.1	网页制作软件	74
3.2.2	图形图像处理软件	75
3.2.3	动画设计软件	76
3.2.4	动态网页制作技术	77
3.3	网站规划设计基础	78
3.3.1	网站设计流程	79
3.3.2	确定网站的类型	80
3.3.3	定位网站的主题	82
3.3.4	确定网站的栏目和板块	82
3.3.5	确定网站的整体风格	83
3.3.6	规划网站目录结构和链接结构	85
3.4	小结	87
3.5	思考题	87



第 4 章 Dreamweaver CS4 工具	88
4.1 Dreamweaver CS4 概述	88
4.1.1 Dreamweaver CS4 简介	88
4.1.2 Dreamweaver CS4 的系统要求和安装	88
4.1.3 启动 Dreamweaver CS4	90
4.1.4 Dreamweaver CS4 的操作界面	91
4.2 规划与创建 Dreamweaver 站点	95
4.2.1 规划站点结构	95
4.2.2 建立本地站点	97
4.3 创建站点文档	101
4.3.1 创建新文档	101
4.3.2 文档的设置	103
4.3.3 设置页面属性	103
4.4 文本的输入和编辑	105
4.4.1 输入文本	105
4.4.2 设置文本属性	106
4.4.3 输入特殊字符	108
4.5 图像处理	108
4.5.1 插入图像	109
4.5.2 设置图像属性	111
4.5.3 创建图像地图	112
4.6 建立网页链接	113
4.6.1 创建文字超链接	114
4.6.2 创建图像超链接	115
4.6.3 创建锚点超链接	116
4.6.4 创建电子邮件超链接	117
4.7 表格处理	117
4.7.1 插入表格	118
4.7.2 表格的基本操作	119
4.7.3 行/列/单元格的基本操作	121
4.7.4 表格的高级操作	122
4.8 CSS 样式表	123
4.8.1 创建 CSS 样式表	123
4.8.2 CSS 属性设置	125
4.8.3 应用 CSS 样式	125
4.9 嵌入表单元素	126
4.9.1 表单对象	126



4.9.2 创建表单	127
4.10 添加多媒体元素	128
4.10.1 插入网页背景音乐	128
4.10.2 插入视频	130
4.10.3 插入 Flash 动画	131
4.11 框架的使用	133
4.11.1 创建框架集	133
4.11.2 框架和框架集的基本操作	134
4.11.3 设置框架属性	135
4.11.4 在框架中使用超链接	136
4.12 站点的整理维护与上传	136
4.12.1 本地测试站点	136
4.12.2 申请主页空间和域名	138
4.12.3 发布站点	139
4.12.4 网站的推广和宣传	141
4.13 小结	141
4.14 思考题	141
第 5 章 Web 客户端脚本语言设计	142
5.1 客户端脚本语言简介	142
5.1.1 常见的客户端脚本语言	142
5.1.2 JavaScript 脚本语言的特点	143
5.2 JavaScript 语言基础	144
5.2.1 标识符与关键字	144
5.2.2 数据类型	145
5.2.3 常量	145
5.2.4 变量	145
5.2.5 运算符/表达式	146
5.2.6 内置对象	146
5.2.7 内置函数	153
5.2.8 事件	153
5.3 JavaScript 自定义函数	154
5.3.1 函数的定义	154
5.3.2 函数的调用	155
5.4 JavaScript 控制语句	156
5.4.1 条件语句 if...else	157
5.4.2 选择语句 switch...case	158
5.4.3 循环语句	159





5.4.4 其他语句	162
5.5 浏览器对象模型 BOM	163
5.5.1 BOM 层次结构	164
5.5.2 浏览器对象的应用	167
5.6 文档对象模型 DOM	168
5.6.1 DOM 模型结构	168
5.6.2 文档对象的应用	169
5.7 Ajax 技术	171
5.7.1 Ajax 异步模型	171
5.7.2 Ajax 技术应用	175
5.8 小结	178
5.9 思考题	178
第 6 章 基于 ASP.NET 的服务器端程序设计	179
6.1 ASP.NET 简介	179
6.1.1 .NET 框架概述	179
6.1.2 开发环境简介	181
6.1.3 创建第一个 ASP.NET 网站	181
6.1.4 应用需求简介	184
6.2 在 VWD 2008 中进行 HTML 和 CSS 设计	185
6.2.1 使用 HTML 工具设计页面	186
6.2.2 使用 CSS 工具设计页面	189
6.3 使用 ASP.NET 服务器控件	195
6.3.1 ASP.NET 控件的类型	196
6.3.2 ASP.NET 服务器控件概述	197
6.3.3 使用 ASP.NET 服务器控件	199
6.4 数据库驱动的 ASP.NET 编程	203
6.4.1 利用 SQL 及存储过程处理数据	204
6.4.2 ADO.NET 技术概述	207
6.4.3 使用 ADO.NET 技术访问 SQL Server 数据库	208
6.4.4 显示和更新数据	217
6.5 创建外观一致的 Web 站点	221
6.5.1 主题与外观	221
6.5.2 用母版页统一页面布局	225
6.6 ASP.NET 内置对象及应用	229
6.6.1 常用内置对象简介	229
6.6.2 内置对象的综合应用举例	233



6.7	小结	236
6.8	思考题	237
第 7 章	基于 JSP 的服务器端程序设计	239
7.1	JSP 简介	239
7.1.1	JSP 的特点	239
7.1.2	JSP 工作原理	240
7.1.3	JSP 的基本语法	241
7.1.4	JSP 和 Java Servlet 技术	242
7.2	JDK 的获取与安装	242
7.2.1	JDK 的下载	242
7.2.2	JDK 的安装	244
7.2.3	设置 JDK 环境变量	245
7.2.4	测试 JDK 环境变量	246
7.3	安装与配置 Tomcat	247
7.3.1	安装 Tomcat	247
7.3.2	测试 Tomcat	248
7.3.3	配置 Tomcat	249
7.4	Java 开发工具——Eclipse 简介	251
7.4.1	下载和安装 Eclipse	251
7.4.2	运行和配置 Eclipse	251
7.4.3	使用 Eclipse 开发一个简单的 Web 应用程序	253
7.5	JSP 基本知识	258
7.5.1	Java 语法基础	258
7.5.2	JSP 语法基础	278
7.6	JSP 常用的内置对象	296
7.6.1	内置对象的概述	296
7.6.2	处理客户请求信息对象 request	297
7.6.3	控制服务器相应信息对象 response	299
7.6.4	管理客户会话对象 session	304
7.6.5	Web 应用程序全局对象 application	305
7.6.6	向客户输出数据对象 out	307
7.6.7	读取 web.xml 配置信息对象 config	308
7.6.8	其他 JSP 内建对象	308
7.7	JDBC 技术	312
7.7.1	JDBC 技术简介	312
7.7.2	JDBC 驱动程序	312



7.7.3 JDBC 中的常用接口	313
7.7.4 JDBC 连接数据库的方法	316
7.8 JavaBean 封装数据库	318
7.8.1 JSP 与 JavaBean 的关系	319
7.8.2 JavaBean 组件	319
7.8.3 页面显示	322
7.9 开发范例	327
7.10 小结	343
7.11 思考题	343
参考文献	345

第 1 章 Web 概述

当前, Internet 已渗入到人们生活中的各个领域, 如网上学习、办公、购物、聊天、玩游戏、看电视、娱乐、发布信息、检索信息等, 它正改变着人们的工作、生活及学习方式。因此, 了解 Internet 所提供的 Web 服务和技术对于使用者来说是非常有必要的。本章将重点介绍 Web 概念、Web 工作原理、Web 服务、Web 技术及其发展趋势。

1.1 Web 概念

Web 是 World Wide Web (环球信息网) 的简称, 中文意思是“万维网”。它起源于 1989 年 3 月, 由欧洲粒子物理实验室 CERN (the European Laboratory for Particle Physics) 所发展出来的主从结构分布式超媒体系统。人们通过访问 Web 服务器, 就可以很迅速、方便地访问到所需要的信息资料。由于用户在使用 Web 浏览器访问信息资源的过程中, 无须关心技术性的细节, 就可以得到想要的信息, 所以, Web 一经推出, 就备受人们的关注, 并得到了迅速、蓬勃的发展。Web 是一种体系结构, 通过它可以访问分布于 Internet 主机上的链接文档, 它有多层含义: Web 是 Internet 提供的一种服务; 是存储在 Internet 网络中数量巨大的文档的集合; 是由彼此关联的文档组成的海量信息源, 这些文档称为主页或页面; Web 的内容保存在 Web 站点中, 用户可以通过浏览器访问 Web 站点。那么 Web 的定义是什么呢?

1.1.1 Web 定义

Web 是英国人 Tim Berners-Lee 于 1989 年在欧洲共同体的一个大型科研机构任职时发明的。互联网上的资源可以在一个网页中比较直观地表示出来, 网络资源之间在网页上可以相互链接。Web 就是一种超文本信息系统, 它使得文本可以从一个位置跳到另外的位置。如果想了解某一个主题的内容, 只要在这个主题上单击一下, 就可以跳转到包含这一主题的文档中。

1.1.2 Web 的五要素

构成 Web 的 5 个要素分别为 URL、HTTP、HTML、Web 浏览器和 Web 服务器。下面分别介绍这 5 个要素的具体含义。



1. URL

URL (Universal Resource Locator) 是统一资源定位器。它是方案集, 包含如何访问 Internet 上的资源的明确指令。另外, URL 是统一的, 无论寻找哪种特定类型的资源 (如音乐、邮件、文件、新闻等) 或描述通过哪种机制获取网络资源, 都采用相似的 URL 格式。

URL 格式为:

`protocol:// hostname[:port] / path / [;parameters][?query]`

其中, 方括号 ([]) 中的为可选项, 其中的参数解释如下。

- **protocol://**: 通信协议方案。常用协议有如下几种形式。
 - ☑ **file://**: 指本地计算机上的文件。
 - ☑ **ftp://**: 指通过 FTP 协议访问 FTP 服务器上的资源。
 - ☑ **http://**: 指通过 HTTP 协议访问网站资源。
 - ☑ **https://**: 指通过 HTTPS 安全协议访问网站资源。
 - ☑ **mms://**: 指通过 MMS 协议访问 Windows Media 发布点上的单播内容。
 - ☑ **ed2k://**: 指通过支持 ED2K 协议的 P2P 软件访问网络资源。
- **hostname**: DNS (域名系统) 中的主机名及其 IP 地址。
- **:port**: 端口号 (整数, 可选), 省略时使用协议的默认端口, 如 ftp 协议的默认端口为 21。
- **path**: 路径, 由零或多个 “/” 符号隔开的字符串, 一般用来表示主机上的一个目录或文件位置。
- **;parameters**: 用于指定特殊参数 (可选)。
- **?query**: 查询 (可选), 用于给动态网页 (如使用 CGI、ISAPI、PHP、JSP、ASP、ASP.NET 等技术制作的网页) 传递参数, 如果有多个参数, 用 “&” 符号隔开, 每个参数的名和值用 “=” 符号隔开。

2. HTTP

HTTP (Hypertext Transfer Protocol) 是超文本传输协议, 是用于从 Web 服务器传输超文本到本地浏览器的传送协议。它不仅保证计算机正确、快速地传输超文本文档, 还确定传输文档中的哪一部分, 以及哪一部分内容首先显示 (如文本先于图形) 等。

3. HTML

HTML (Hypertext Markup Language) 是超文本标记语言, 它之所以称为超文本标记语言, 是因为文本中包含了 “超链接” 点。所谓超链接, 就是一种 URL 指针, 通过单击它, 浏览器可以方便地获取新的网页。HTML 是一种标准, 它通过标记符号来标记要显示的网页中的各个部分。网页文件本身是一种文本文件, 通过在文本文件中添加标记符, 可以告诉浏览器如何显示其中的内容 (如文字如何处理、画面如何安排、图片如何显示等)。浏览器按顺序阅读网页文件, 然后根据标记符解释和显示其标记的内容, 对书写出错的标记不指出其错误, 且不停止其解释执行过程, 编制者只能通过显示效果来分析出错原因和出错部位。



但需要注意的是,对于不同的浏览器,对同一标记符可能会有不完全相同的解释,因而可能会有不同的显示效果,由此可见,网页的本质就是 HTML,通过结合使用其他的 Web 技术(如脚本语言、CGI、ASP、JSP 等),可以创造出功能强大的动态网页。因而,HTML 是 Web 编程的基础,也可以说万维网是建立在超文本基础之上的信息网络。

4. Web 浏览器

Web 浏览器(Web Browser)实际上是一个软件程序,用于与 Web 服务器建立连接,并与之进行通信。它可以通过访问 DNS 系统中的主机名确定信息资源的位置,并将用户感兴趣的信息资源返回到浏览器中的网页里,对 HTML 文件进行解释,然后将文字图像或者多媒体信息还原出来。目前,国内外常用的浏览器为 IE(Internet Explorer 微软)、NN(Netscape Navigator 网景)、Mosaic(美国伊利诺州的伊利诺大学的 NCSA 组织)、Opera(Opera Software ASA 公司)、Chrome(谷歌)、Safari(苹果)、Firefox(火狐浏览器)、TT(腾讯)、Maxthon(遨游)和 360 安全等。

5. Web 服务器

Web 服务器(Web Server)指驻留在 Internet 上的某种类型的计算机程序,它可以解析 HTTP 协议。当 Web 服务器接收到一个 HTTP 请求(Request)时,会返回一个 HTTP 响应(Response)。为了处理一个请求(Request),Web 服务器可以响应(Response)一个静态页面或图片,进行页面跳转(Redirect),或者把动态响应(Dynamic Response)的产生委托(Delegate)给一些其他的程序,如 CGI 脚本、JSP(Java Server Pages)脚本、Servlet、ASP(Active Server Pages)脚本,无论这些脚本的目的如何,服务器端的程序通常产生一个 HTML 的响应(Response)来让浏览器可以浏览。Tomcat、NCSAhttpd 和 Apache 服务器是基于 UNIX 和 Linux 平台的,而 IIS 是基于 Windows 平台的浏览器。下面简要介绍这几种 Web 服务器。

(1) Microsoft IIS

IIS(Internet Information Server)是 Microsoft 的 Web 应用程序服务器产品,它允许在公共 Intranet 或 Internet 上发布信息。IIS 是目前最流行的 Web 服务器产品之一,很多著名的网站都建立在 IIS 的平台上。IIS 提供了 Internet 服务管理器,它是基于图形界面的管理工具,主要可用于监视配置和控制 Internet 服务。

IIS 是一种 Web 服务组件,提供 Web 服务器、FTP 服务器、NNTP 服务器和 SMTP 服务器等多种功能,主要用于浏览网页、传输文件、新闻服务和发送邮件。通过 IIS 服务器可以很容易地在 Internet 或 Intranet 上发布信息。

(2) IBM WebSphere

IBM WebSphere 是一种功能完善、开放的 Web 应用程序服务器,它基于 Java 的应用环境,用于建立、部署和管理 Internet 和 Intranet Web 应用程序。WebSphere 服务器针对以 Web 为中心的开发人员,支持 HTTP 协议和动态网页编程技术。IBM 为用户从简单的 Web 应用程序转移到电子商务世界提供 WebSphere 产品系列支持。



(3) BEA WebLogic

BEA WebLogic 是一种多功能、基于 Java 应用环境的 Web 应用服务器，为企业构建自己的应用提供了坚实的基础。各种应用开发、部署所有关键性的任务、集成各种系统和数据库、提交服务、跨 Internet 协作，起始点都是 BEA WebLogic 服务器。BEA WebLogic 服务器在使用应用服务器成为企业应用架构的基础方面继续处于领先地位，为构建集成化的企业级应用提供了稳固的基础。由于它具有全面的功能，所以，基于 Internet 的企业都选择它来开发、部署最佳的应用。

(4) Apache

Apache 是一款使用最广的、基于 UNIX、Windows、Linux 系统平台的 Web 应用服务器。它源于 NCSAhttpd 服务器，当 NCSA Web 服务器项目停止后，那些使用 NCSA Web 服务器的人们开始交换用此服务器的补丁，这也是 Apache 名称的由来（pache，补丁）。世界上很多著名的网站都是 Apache 的产物，它的成功主要在于它的源代码开放、有一支开放的开发队伍、支持跨平台的应用及可移植性等。

(5) Tomcat

Tomcat 是一个开放源代码、运行 Servlet 和 JSP Web 应用软件的基于 Java 的 Web 应用软件容器。它是基于 Apache 许可证下开发的自由软件，是 Java Servlet 2.2 和 JavaServer Pages 1.1 技术的标准实现，是完全重写的 Servlet API 2.2 和 JSP 1.1 兼容的 Servlet/JSP 容器。Tomcat 使用了 JServlet 的一些代码，特别是 Apache 服务适配器。随着 Catalina Servlet 引擎的出现和 Tomcat 6.0 性能的提升，使它成为一个值得考虑的 Servlet/JSP 容器，因此，目前许多 Web 服务器都采用 Tomcat。

Web 服务器有 3 个主要发展趋势：

(1) 从 HTML 到 XML (Extensible Markup Language, 可扩展标记语言)

HTML 作为 Web 的开发语言，被称为“第一代 Web 语言”，它对 Web 应用的发展起到了关键性的作用。但它的致命缺点是只适合于人与计算机的交流，而不适合计算机与计算机的交流。HTML 通过大量的标记来定义文档内容的表现方式，它仅描述了应如何在 Web 浏览器页面上布置文字、图形，而没有对 Internet 中最重要的信息含义本身进行描述。Web 服务器向 Web 浏览器提供的信息都来自有一定结构的数据库，在数据库中，为了检索和管理的方便，信息按照它本身的意义（如学号、姓名、班级、成绩等）被存放在相应的字段里，一旦这些数据被调出来，经过 CGI、ASP.NET、JSP、PHP 等转换成 HTML 后，其原来的意义无法转移到 HTML 标记中来，用户也就无法按照信息本来的意义去阅读。此外，HTML 还有一个问题，就是它的标记集合是固定的，所以用户不能根据自己的需要增加标记。

由于操作系统以及数据库的不同，不同的系统及应用层面之间要想互相理解对方的数据格式是相当困难的，这就需要一种新技术或标准，能够将最初保存在数据库服务器中的原始数据结构在不同的系统层面共享，这种新技术就是 XML。

W3C 对 XML 作了这样的描述——XML 描述了一类被称为 XML 文档的数据对象，并部分描述了处理它们的计算机程序的行为。XML 是 SGML 的一个应用实例。从结构上说，XML 文档遵从 SGML 文档标准。同 HTML 一样，XML 也是一种基于文本的标记语言，都是从 SGML (Standard Generalize Markup Language, 标准通用标记语言) 发展而来的，二者

