



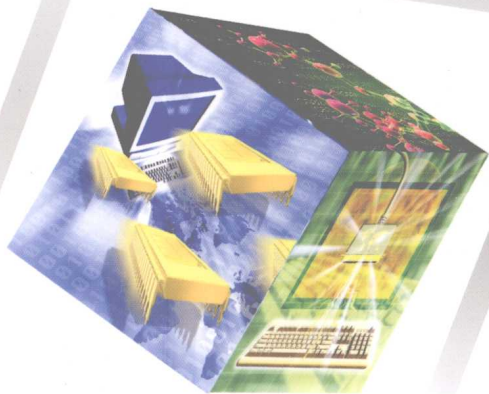
高等学校精品规划教材

计算机网络与通信系列

郝兴伟 编 著

Web程序设计

(第二版)



中国水利水电出版社

www.waterpub.com.cn

21 世纪高等学校精品规划教材

Web 程序设计

(第二版)

郝兴伟 编 著



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书以 Web 中的 B/S 三层结构为主线,以编者具体的研发项目为背景,按照“任务驱动”的编写方式,系统地讲解了 Web 编程问题,并提供了大量的案例和代码。全书共分 6 章,分别介绍了 Web 程序设计基础,计算模式与编程模式的演变,B/S 结构与 Web 应用,Java 技术、XML 技术、Web 服务等 Java 核心技术及 Web 的最新进展;Web 程序及运行环境,包括 Tomcat 服务器的配置,Web 应用的规划和部署;标记语言 HTML 规范;网页设计与制作,包括网页设计理念和 FrontPage 网页制作工具;Web 客户端编程,包括 JavaScript 和 AJAX 技术,数据获取及数据有效性验证;Web 服务器编程,包括 Java 程序设计基础、JavaBean、JSP 技术、文件及文件夹操作、数据库编程等内容。

本书配套有《Web 程序设计实训教程(第二版)》,特别适合作为培养应用型人才的高等学校计算机应用、信息管理及电子商务等专业学生的 Web 技术导论、Web 程序设计、互联网与 Web 编程等课程的教材,也可以作为非计算机专业高年级学生学习计算机网络及应用和 Web 编程的综合性书籍。

本书所配电子教案及所有案例和程序代码,都可以从中国水利水电出版社网站免费下载,网址为:<http://www.waterpub.com.cn/softdown/>。使用本书的读者也可与编者(hxw@sdu.edu.cn)联系,或登录该课程网站(<http://jcjy.sdu.edu.cn/>或<http://202.194.28.11/>),获取更多教学资源。

图书在版编目(CIP)数据

Web 程序设计 / 郝兴伟编著. —2 版. —北京: 中国水利水电出版社, 2008

21 世纪高等学校精品规划教材

ISBN 978-7-5084-6226-4

I. W… II. 郝… III. 主页制作—程序设计—高等学校—教材 IV. TP393.092

中国版本图书馆 CIP 数据核字(2008)第 214216 号

策划编辑: 雷顺加 责任编辑: 吴 萍 加工编辑: 许波林 封面设计: 李 佳

| | |
|-------|--|
| 书 名 | 21 世纪高等学校精品规划教材 Web 程序设计(第二版) |
| 作 者 | 郝兴伟 编 著 |
| 出版 发行 | 中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266(总机)、68367658(营销中心)、82562819(万水) |
| 经 售 | 全国各地新华书店和相关出版物销售网点 |
| 排 版 | 北京万水电子信息有限公司 |
| 印 刷 | 北京市天竺颖华印刷厂 |
| 规 格 | 184mm×260mm 16 开本 17.25 印张 427 千字 |
| 版 次 | 2008 年 12 月第 2 版 2008 年 12 月第 1 次印刷 |
| 印 数 | 0001—5000 册 |
| 定 价 | 28.00 元 |

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换
版权所有·侵权必究

前 言

进入 20 世纪 90 年代以后，人们在享受互联网给我们的工作和生活带来无限便利的同时，Web 技术也正在悄悄地影响和改变着我们的思维方式。在软件开发和程序设计领域，Web 程序设计正悄然兴起，C/S 结构的系统风光不再，B/S 结构在市场竞争中已经脱颖而出。现在，无论是企业的电子商务，还是政府办公系统，可以说，80%以上的计算机应用都是基于 Web 的，使互联网这个现代的通信平台找到了最广阔的应用市场。

身处教育行业和 IT 行业两重身份，一直对计算机软件技术的发展非常关注，希望所教授的知识和社会的需求紧密相关，培养的学生既有扎实、宽厚、系统的基础理论知识，又有优秀的开发能力。理论和实践的结合需要一个载体，对计算机软件来说，项目是再好不过的载体了。在软件项目的实施中，我们可以对相关的理论知识进行应用验证；反过来，在软件开发和应用中遇到的问题、难题，又推动了对理论问题的研究。

科研和教学就是这样地相辅相承，项目作为一种结合点，推动着科研和教学水平的不断提高。从 2001 年开始，我们开始基于 B/S 三层架构的软件开发工作，开发领域主要是 E-learning，虚拟实验室，还有一些通用的管理系统，例如会议管理、论文管理、作业管理等。在这些 Web 应用的开发中，遇到了许许多多的问题，也积累了很多的经验，有很多的体会。我们不断地将开发和研究中的体会，在通过大学讲台和学生们交流和分享的同时，先后于 2005 年、2007 年探索性地出版了《Web 技术导论》和《Web 开发技术》两个版本的教材，书中的例题和案例大都来源于这些研发项目。

对于每一位任课教师，或者是学习 Web 开发的学生来说，往往会对自己上网过程中见到的良好的网页感兴趣，会想到能够借鉴其中的代码，来提高自己的程序质量，提高编程效率。这是每一个软件开发人员在长期的职业生涯中养成的习惯，即使自己不再编程，也愿意收集一些程序代码和编程宝典类的书籍，学习其中的编程技巧。这些也正是一些教科书中缺乏的东西，才有了许多软件公司人员的抱怨，说我们培养的学生不能编程。虽然市场上有许多专门的开发类书籍，但这些书籍的读者对象通常是专业开发人员，并不适合于教学。因此，结合市场需求和教学的需要，在前两本已出版的教材基础上，减少了一些过时的技术内容，增加了更多实用性技术、常用问题及代码，形成本书。

在本书中，我们把 Web 开发中最常用的一些实现代码以例题或案例的形式写出来，不但用以巩固所学到的知识点，更重要的是为我们未来的软件编程提供借鉴。本书共收集和讲解了 110 多段非常有用的程序代码，内容涉及 HTML 中的页面布局，CSS 的设计和应用，Web 中的弹出式菜单、树形菜单的设计和不同的实现方法，表单数据的获取，中文字符、email、用户账户、整数、浮点数数字输入的有效性验证，数据之间的类型转换，数组和集合类，页面之间的参数传递，页面安全，服务端的文件和文件夹操作，数据库的访问和操作，AJAX 异步通信等各种各样的实用代码。

本书分成 6 章，主要内容如下：

第 1 章 Web 程序设计基础。介绍了计算模式的演变，不同计算模式下的软件开发问题。

同时，对 Web 的工作机理、Java 技术、XML 技术、Web 服务、SOA 架构等相关技术，语义 Web、Web 2.0 等的新进展进行了概要性介绍。

第 2 章 Web 程序及运行环境。讲解了 Web 应用及其基本结构，对 Web 应用的运行环境进行了详细讲解。围绕 Java 技术，介绍了 Java、Tomcat 应用服务器的安装和配置。重点讲解了 Web 应用的规划和部署。

第 3 章 标记语言和 HTML 基础。标记语言是 Web 应用开发的基础，它是 Web 应用界面的实现手段。本章首先介绍了标记语言的概念，然后详细介绍了 HTML 标记语言的语法，对 css 技术进行了深入讲解，并安排了大量的例子解释每种元素的含义和使用。

第 4 章 网页设计与制作。网页作为 Web 应用的主要用户界面，在 HTML 基础上，加强了网页设计的讲解，包括：页面功能与内容设计、页面布局设计、页面视觉设计以及页面效果设计等。本章还介绍了可视化制作工具 FrontPage 的使用。

第 5 章 客户端编程。首先讲解了 Web 浏览器的基本工作原理，然后讲解了客户端脚本程序设计语言 JavaScript、浏览器对象模型 BOM、HTML 文档对象模型 DOM、Web 交互的内容，并讲解了 AJAX 技术。最后安排了四个综合性案例，详细讲解 JavaScript 中的菜单实现、表单数据的有效性验证、表单数据的处理、参数传递以及页面安全性等问题。这些综合案例中包含了大量的 Web 开发中所需要的代码，都来源于我们具体的研发项目，相信通过这些案例的学习，会大大提高编程人员的 Web 开发能力和代码质量。

第 6 章 服务端编程。首先介绍了 B/S 三层结构的概念，然后简单讲解了 Java 程序设计语言基础，介绍了 Java 技术的特点、类与对象、接口、包等基本概念，介绍了 Java Applet、JavaBean、Servlet 服务器程序的概念，这些概念是开展基于 Java 技术的服务端编程的基础。

本章的重点是 JSP 技术、数据库编程以及 MVC 设计模式的讲解。在 JSP 技术中，讲解了 JSP 的语法、JSP 中的数据类型及其转换、JSP 内置对象、JSP 中的参数传递方法等内容。然后讲解了服务端的数据库编程，并给出了一个完整的案例。最后，讲解了基于 Ajax 技术的在线聊天 Web 应用的整个开发过程，同时对常用的 Java 开发工具进行了介绍。

作为互联网的用户和 Web 技术的开发者和实践者，同时，作为一个公司派的高校教师，希望这本书的知识结构和内容对于您学习 Web 开发，进行 Web 编程能有所帮助。也希望其中的大量例子在您未来的 Web 开发中，为您的编程给以启发，节省宝贵的时间。软件开发是一个积累的过程，让我们一起在这种积累中进步，来享受成功的乐趣。

在本书写作的过程中，非常感谢我的同事巩裕伟老师、焦文江老师、杨兴强老师、吕刚老师、阚铮老师和李蕴老师的工作和他们提出的良好建议，还要感谢我的学生苏雪、常跃峰、崔旭、朱岩、郝凤杰、田容雨、尤凤英、董颖等，他们都参与了我们的许多项目的研发工作，编写了大量的程序代码，祝愿他们在以后的工作和生活中一切顺利，祝愿他们取得更大的成绩。此外，还要感谢中国水利水电出版社的雷顺加编审，在本书的策划和写作中，提出了很好的建议，特别是对编写方式及习题的策划，使得本书能够更好地用于教学。

由于本书涉及的内容非常广泛，在深度和广度上很难做到完美，同时，也由于作者本人的知识面和认识有限，书中错误和不足在所难免，敬请各位同行和广大读者批评指正。

作者 Email: hxw@sdu.edu.cn

作者

2008 年 10 月

目 录

前言

| | |
|--------------------------------|----|
| 第 1 章 Web 程序设计基础 | 1 |
| 1.1 计算模式与程序设计模式的演变 | 1 |
| 1.1.1 单机运行与集中式计算模式 | 1 |
| 1.1.2 客户/服务器 (C/S) 计算模式 | 2 |
| 1.1.3 浏览器/服务器 (B/S) 计算模式 | 2 |
| 1.1.4 软件设计与开发模式的演化 | 4 |
| 1.2 Web 及其工作原理 | 5 |
| 1.2.1 什么是 Web | 5 |
| 1.2.2 Web 的工作原理 | 6 |
| 1.2.3 Web 浏览器 | 6 |
| 1.2.4 Web 服务器 | 7 |
| 1.3 Web 相关技术 | 7 |
| 1.3.1 概念及术语 | 7 |
| 1.3.2 Java 技术 | 8 |
| 1.3.3 XML 技术 | 10 |
| 1.3.4 Web 服务 | 12 |
| 1.4 Web 发展趋势 | 13 |
| 1.4.1 Web 2.0 | 13 |
| 1.4.2 语义 Web | 14 |
| 本章小结 | 16 |
| 习题 1 | 16 |
| 第 2 章 Web 程序及运行环境 | 18 |
| 2.1 Web 应用及基本结构 | 18 |
| 2.1.1 传统程序与软件开发 | 18 |
| 2.1.2 Web 应用的基本结构 | 19 |
| 2.2 操作系统与 Web 服务器 | 21 |
| 2.2.1 Web 服务器 | 21 |
| 2.2.2 主流 Web 服务器简介 | 21 |
| 2.3 基于 Java 的 Web 应用运行环境 | 23 |
| 2.3.1 Apache 服务器的作用和功能 | 23 |
| 2.3.2 Tomcat 应用服务器的作用和功能 | 23 |
| 2.4 Tomcat 的安装和配置 | 24 |
| 2.4.1 Servlet/JSP 规范 | 24 |

| | | |
|--------------|---------------------------|-----------|
| 2.4.2 | 安装 Java 运行环境 | 24 |
| 2.4.3 | Tomcat 的安装和配置 | 29 |
| 2.4.4 | 建立并部署 Web 应用 | 35 |
| 2.4.5 | 使用虚拟目录 | 39 |
| 2.5 | Web 应用的规划与运行 | 40 |
| 2.5.1 | 规划 Web 应用 | 40 |
| 2.5.2 | Web 应用的运行 | 41 |
| | 本章小结 | 41 |
| | 习题 2 | 42 |
| 第 3 章 | 标记语言 HTML 基础 | 43 |
| 3.1 | 标记语言及其发展 | 43 |
| 3.1.1 | 标准通用标记语言 SGML | 44 |
| 3.1.2 | 超文本标记语言 HTML | 44 |
| 3.1.3 | 可扩展 HTML 规范 XHTML | 45 |
| 3.1.4 | 可扩展标记语言 XML | 45 |
| 3.2 | HTML 标记语言基础 | 45 |
| 3.2.1 | 标记类型与标记属性 | 46 |
| 3.2.2 | 文档结构 | 46 |
| 3.3 | 文件头标记及子标记 | 47 |
| 3.4 | 文件体标记及其属性 | 49 |
| 3.5 | 文档内容常用标记 | 51 |
| 3.5.1 | 标题、段落与文本标记 | 51 |
| 3.5.2 | 图像标记 | 52 |
| 3.5.3 | 超链接标记与书签 | 53 |
| 3.5.4 | 影像地图标记 | 54 |
| 3.5.5 | 表格 | 55 |
| 3.5.6 | 表单 | 57 |
| 3.5.7 | 其他标记 | 64 |
| 3.6 | div 标记与 span 标记 | 66 |
| 3.6.1 | 层次块 div 标记 | 66 |
| 3.6.2 | span 标记 | 67 |
| 3.7 | 对象和脚本程序标记 | 68 |
| 3.8 | 层叠样式表 CSS 技术 | 70 |
| 3.9 | 帧 | 76 |
| 3.10 | 其他相关技术 | 78 |
| 3.10.1 | DHTML 技术 | 78 |
| 3.10.2 | SHTML 技术 | 78 |
| | 本章小结 | 79 |
| | 习题 3 | 79 |

| | |
|--------------------------------|------------|
| 第 4 章 网页设计与制作 | 83 |
| 4.1 网页设计基础 | 83 |
| 4.1.1 页面功能与内容设计 | 83 |
| 4.1.2 页面布局设计 | 84 |
| 4.1.3 页面视觉设计 | 87 |
| 4.1.4 页面效果设计 | 88 |
| 4.2 使用 FrontPage | 89 |
| 4.2.1 FrontPage 主窗口 | 89 |
| 4.2.2 网站的新建与维护 | 90 |
| 4.2.3 新建网页 | 92 |
| 4.3 网页编辑 | 94 |
| 4.3.1 输入文本内容 | 94 |
| 4.3.2 插入图片 | 95 |
| 4.3.3 建立超链接或书签 | 96 |
| 4.3.4 图像地图 | 97 |
| 4.3.5 插入表格 | 98 |
| 4.3.6 插入表单 | 100 |
| 4.4 设置标记属性 | 101 |
| 4.4.1 使用 IntelliSense 技术 | 102 |
| 4.4.2 使用行为面板 | 102 |
| 4.5 定义和使用样式 | 103 |
| 4.5.1 定义样式 | 103 |
| 4.5.2 使用样式表文件 | 104 |
| 4.6 Frame 框架和 IFrame 框架 | 104 |
| 4.6.1 Frame 框架网页 | 105 |
| 4.6.2 使用浮动框架 IFrame | 108 |
| 本章小结 | 111 |
| 习题 4 | 111 |
| 第 5 章 客户端编程 | 114 |
| 5.1 浏览器与客户端脚本程序 | 114 |
| 5.1.1 浏览器与客户端脚本引擎 | 115 |
| 5.1.2 脚本语言规范与主要的客户端脚本语言 | 115 |
| 5.2 JavaScript 程序设计基础 | 117 |
| 5.2.1 JavaScript 基本符号 | 117 |
| 5.2.2 数据和数据类型 | 118 |
| 5.2.3 常量和变量 | 119 |
| 5.2.4 表达式和运算符 | 120 |
| 5.2.5 基本语句 | 120 |
| 5.2.6 函数 | 124 |

| | | |
|--------------|-------------------------------|------------|
| 5.3 | 事件驱动及事件处理 | 124 |
| 5.3.1 | 事件驱动的程序执行过程..... | 124 |
| 5.3.2 | JavaScript 中的常用事件..... | 124 |
| 5.4 | 对象及其操作 | 126 |
| 5.4.1 | 类与对象的概念 | 126 |
| 5.4.2 | 对象的操作 | 127 |
| 5.5 | 常用内部对象及函数 | 128 |
| 5.5.1 | String 对象..... | 128 |
| 5.5.2 | Math 对象..... | 131 |
| 5.5.3 | Date 对象..... | 132 |
| 5.5.4 | Array 数组对象..... | 135 |
| 5.5.5 | 预定义函数 | 136 |
| 5.6 | JavaScript 浏览器对象模型 BOM..... | 138 |
| 5.6.1 | BOM 层次结构 | 138 |
| 5.6.2 | window 对象..... | 138 |
| 5.6.3 | frames 对象 | 143 |
| 5.6.4 | location 对象..... | 144 |
| 5.6.5 | history 对象 | 144 |
| 5.6.6 | screen 对象 | 145 |
| 5.6.7 | event 对象..... | 146 |
| 5.7 | HTML 文档对象模型 DOM..... | 147 |
| 5.7.1 | 文档对象模型 DOM | 147 |
| 5.7.2 | HTML DOM 对象..... | 148 |
| 5.8 | Web 交互 | 155 |
| 5.8.1 | 使用 form 实现 Web 页面的信息交互..... | 155 |
| 5.8.2 | 使用 frame 实现更复杂的交互 | 164 |
| 5.9 | 使用 AJAX 技术 | 170 |
| 5.9.1 | AJAX 基础..... | 170 |
| 5.9.2 | XMLHttpRequest 对象..... | 171 |
| 5.10 | 综合举例 | 173 |
| 5.10.1 | 创建折叠式菜单 | 174 |
| 5.10.2 | 创建树形菜单 | 177 |
| 5.10.3 | 表单数据综合处理 | 179 |
| 5.10.4 | 页面之间参数传递 | 188 |
| | 本章小结 | 195 |
| | 习题 5 | 196 |
| 第 6 章 | 服务端编程 | 199 |
| 6.1 | B/S 三层体系结构与 Web 服务器脚本程序 | 199 |
| 6.1.1 | B/S 三层体系结构 | 200 |

| | | |
|-------|--------------------------|-----|
| 6.1.2 | 脚本引擎与服务端脚本程序..... | 200 |
| 6.2 | Java 程序设计基础 | 201 |
| 6.2.1 | Java 程序设计语言 | 202 |
| 6.2.2 | 类与对象 | 204 |
| 6.2.3 | 接口 | 211 |
| 6.2.4 | 包 | 212 |
| 6.2.5 | Java 基础类库 | 213 |
| 6.2.6 | Java Applet..... | 214 |
| 6.2.7 | Servlet 服务器程序..... | 215 |
| 6.3 | JavaBean..... | 216 |
| 6.3.1 | 什么是 JavaBean | 216 |
| 6.3.2 | JavaBean 的属性、方法和事件 | 216 |
| 6.4 | JSP 技术 | 218 |
| 6.4.1 | JSP 的运行环境 | 219 |
| 6.4.2 | JSP 的语法结构 | 220 |
| 6.4.3 | JSP 中的数据类型及其转换..... | 222 |
| 6.4.4 | JSP 内置对象 | 224 |
| 6.4.5 | 在 JSP 中使用 JavaBean | 227 |
| 6.4.6 | JSP 中的文件与文件夹操作..... | 229 |
| 6.4.7 | JSP 中的参数传递方法 | 231 |
| 6.5 | JDBC 与数据库编程..... | 232 |
| 6.5.1 | JDBC 接口..... | 232 |
| 6.5.2 | 数据库操作 | 234 |
| 6.5.3 | 举例 | 235 |
| 6.6 | 综合举例——在线聊天程序..... | 244 |
| 6.6.1 | 系统分析 | 244 |
| 6.6.2 | 客户端设计 | 244 |
| 6.6.3 | 服务端设计 | 254 |
| 6.7 | Java 开发工具简介 | 260 |
| 6.7.1 | JDK..... | 260 |
| 6.7.2 | Sun NetBeans 集成开发环境..... | 261 |
| 6.7.3 | Eclipse 开发平台..... | 261 |
| 6.7.4 | JBuilder 开发环境..... | 262 |
| 6.7.5 | JDeveloper 开发框架 | 262 |
| | 本章小结 | 263 |
| | 习题 6 | 263 |
| | 参考资料 | 265 |

第 1 章 Web 程序设计基础



本章导读

随着互联网的普及和广泛应用，在软件开发中，基于 B/S 三层结构的 Web 应用框架已经成为市场的主流，传统 C/S 结构的程序风光不再。上述情形只是出现在软件公司中，在高等学校，专门讲授 Web 开发的课程还很少，Web 程序设计还处于起步阶段。

本章首先介绍计算模式与程序设计模式的演变，对不同模式下的计算机程序体系结构进行对比，然后给出 Web 应用（程序）的概念、体系结构，讲解 Web 的工作原理。最后给出 Web 中的主要概念，为后续章节的学习给出一个框架和知识铺垫。



本章要点

- 计算模式、软件设计与开发模式的演化
- Web 中的 B/S 三层体系结构及其工作原理
- Web 中的相关概念，包括 Java 技术、XML 技术、Web 服务等
- Web 新进展、Web2.0、语义 Web 等

1.1 计算模式与程序设计模式的演变

1946 年，第一台电子计算机“爱尼亚克”（ENIAC）在美国宾夕法尼亚大学莫尔电子工程学院诞生。这种计算技术的革命，透出了数字信息时代的第一缕曙光，开启了计算机程序设计的历史。进入 20 世纪 90 年代以后，随着 Internet 技术的不断发展，特别是 Web 的出现，对计算机的计算模式、软件开发模式、应用模式都产生了重要的影响。

在 Web 中，B/S 三层结构的提出，突破了传统的程序单机运行模式和基于局域网的 C/S 运行模式，突破了局域网的约束，将计算机应用分布到了整个互联网中，使计算机程序的设计和应用模式都发生了革命性的变化，进而推动了 Web 应用的快速发展。

1.1.1 单机运行与集中式计算模式

在计算机诞生和应用的初期，计算所需要的数据和程序都是集中在一台计算机上进行的，称为集中式计算。随着网络的发展，这种集中式计算往往形成一种由大型机和多个与之相连的终端组成的网络结构。当支持大量用户时，大型机自顶向下的维护和管理方式显示出集中式处理的优越性。它具有安全性好、可靠性高、计算能力和数据存储能力强以及系统维护和管理费用较低等优点。但是它也存在着一些明显的缺点，如：大型机的初始投资较大、可移植性差、资源利用率低以及网络负载大等。

此时的计算机程序采用了源程序、编译、连接到运行的这样一个简单的开发过程，最终

形成一个可在计算机操作系统上运行的可执行文件。这个可执行文件被安装在计算机上，它是早期程序主要的应用模式，同时也是 20 世纪 80 年代 C/S 计算模式下的主要应用形式。单机环境下的程序开发过程如图 1-1 所示。

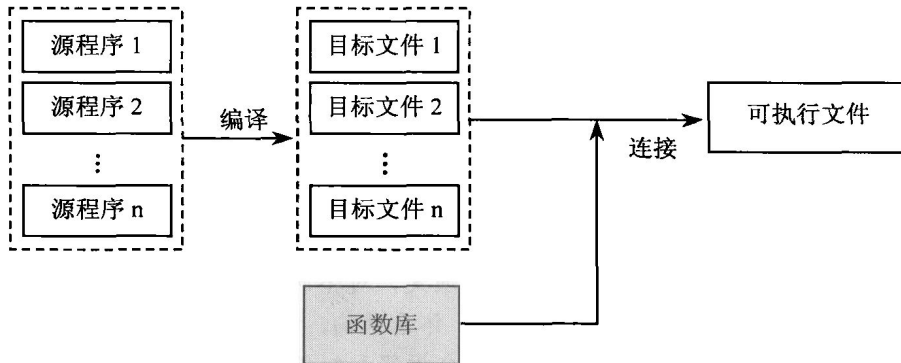


图 1-1 早期的编译型计算机程序开发过程

编程人员利用开发工具（如 C、Fortran）来编写程序，即源程序。然后对源程序进行编译、连接操作，最终形成一个可执行程序，即一个 exe 文件。这种开发模式一直影响到 C/S 计算模式，直到 Web 的出现，一种基于 Web 服务的程序开发和运行模式开始出现和被广泛地应用，基于 Web 的软件开发开始成为程序设计和应用的主流。

1.1.2 客户/服务器（C/S）计算模式

20 世纪 80 年代，随着微型计算机和网络的发展，数据和应用逐渐转向了分布式，即数据和应用程序跨越多个节点机，形成了新的计算模式，这就是客户/服务器（Client/Server, C/S）计算模式。C/S 模式是一种典型的两层计算模式，它将应用一分为二：前端是客户机，一般使用微型计算机，几乎所有的应用逻辑都在客户端进行和表达，客户机完成与用户的交互任务，具有强壮的数据操纵和事务处理能力。后端是服务器，可以使用各种类型的主机，服务器负责数据管理，提供数据库的查询和管理、大规模的计算等服务。

C/S 计算模式具有以下几个方面的优点：通过异种平台集成，能够协调现有的各种 IT 基础结构；分布式管理；能充分发挥客户端 PC 的处理能力，安全、稳定、速度快，且可脱机操作。但随着应用规模的日益扩大，应用程序的复杂程度不断提高，C/S 结构逐渐暴露出许多的缺点和不足，主要包括：它必须在客户端安装大量的应用程序（客户端软件），开发成本较高，移植困难，用户界面风格不统一，使用繁杂，不利于推广使用，维护复杂，升级麻烦，信息内容和形式单一，新技术不能轻易应用等。

1.1.3 浏览器/服务器（B/S）计算模式

在 Web 出现以前，计算机的应用模式经历了单机应用到网络应用两个阶段，这些不同的计算模式有各自的优点和不足。Web 的出现使得一种围绕 Web 服务的计算模式成为当前计算机应用的主流模式，并推动了软件开发、软件应用、应用集成方式上的重大改变。

1. 互联网及 Web 的发展

在互联网的发展历史上，有几个历史性的事件和时刻。首先是 1969 年美国国防部高级研

究计划署 ARPA 资助的一个有关广域网的项目，即阿帕网（ARPANet），它为后来 Internet 的出现奠定了基础。1969 年 11 月 21 日中午，6 名科学家聚会美国加利福尼亚大学洛杉矶分校的计算机实验室，观看这里的一台计算机与远在千里之外的斯坦福研究所的另一台计算机连通。这是一个历史性的时刻，正像 20 年后《时代》周刊的评论：这些研究者根本没有想到，他们不只是连接了两台计算机，而是宣告了网络世界的到来。

在接下来的短短 30 多年的时间里，从 1969 年到 1984 年的这个时期，是 Internet 的提出、研究和试验阶段。1984 年到 1992 年期间，是 Internet 的实用发展阶段。特别是 1989 年，这一年出现的万维网技术给 Internet 赋予了强大的生命力，它是互联网历史上划时代的水分岭，万维网将 Internet 带入了一个崭新的时代。1992 年以后，Internet 开始进入它的商业化发展阶段，万维网的出现，使 Internet 的使用更简单、更方便，Internet 用户开始向全世界扩展，开创了 Internet 发展的新时期。

2. B/S 计算模式

客户/服务器（C/S）模式表现出了许多不足，特别是它的胖客户机和对局域网的依赖，已经不能适应 Web 的发展。人们需要利用互联网，将应用分布到整个 Web 中，而不是局限于企业局域网内部，这就导致了一种更加灵活的多级分布式计算模式，即浏览器/服务器模式（Browser/Server, B/S）的产生和发展。

浏览器/服务器（B/S）计算模式是一种基于 Web 的协同计算，是一种三层架构瘦客户机/服务器计算模式。第一层为客户端表示层，与 C/S 结构中的“胖”客户端不同，三层架构中的客户层只保留一个 Web 浏览器，不存放任何应用程序，其运行代码可以从位于第二层的 Web 服务器下载到本地的浏览器中执行，几乎不需要任何管理工作，是一种“瘦”客户机。第二层是应用服务器层，由一台或多台 Web 服务器组成，处理应用中的所有业务逻辑、对数据库的访问等工作。该层具有良好的可扩充性，程序的部署和管理主要在 Web 服务器上进行，相对于 C/S 而言无论是工作的复杂性还是工作量都大大减少。第三层是数据中心层，安装数据库服务器，负责整个应用中的数据管理。

B/S 计算模式与传统的 C/S 结构相比体现了集中式计算的优越性：具有良好的开放性，利用单一的访问点，用户可以在任何地点使用系统；用户可以跨平台以相同的浏览器界面访问系统；因为在客户端只需要安装浏览器，基本上取消了客户端的维护工作，有效地减少了整个系统的运行和维护成本。

3. Web 应用系统

和传统的单机运行模式不同，Web 下的程序和传统的可执行程序有着太多的区别。一般意义下，根据操作系统的不同，程序有 Windows 程序、UNIX 程序等，一个在 Windows 下运行的应用程序，在 UNIX 或 Linux 下是不能直接运行的，这就是程序的可移植性问题。

随着 Windows 操作系统的出现，应用程序在外观上和操作上有了一个基本的模式，那就是基于窗口的程序。每个程序都有一个主窗口，窗口上有标题栏、菜单栏、工具栏等，用户通过执行菜单命令或工具按钮来执行相应的操作。这就是传统的应用程序给我们的感受。

但是，在 Web 中，Web 程序或者说 Web 应用，和上述我们对程序的印象完全不同。一个 Web 应用是由组织在一个文件夹（Web 应用或站点主目录）下的大量的文件和子文件夹组成的，这些文件就是我们看到的一个个网页，它们之间通过超链接来表达业务逻辑，实现传统程序中的菜单功能。这些页面不再是传统意义上的一个个源程序经过编译、链接而成的，它们不再需要安装在用户的计算机上，而是被部署在一台称为 Web 服务器的计算机上，用户通过

Internet 运行 Web 服务器上的程序, 为互联网中的所有用户提供 Web 服务。

1.1.4 软件设计与开发模式的演化

从计算机诞生到现在, 计算机硬件技术在发展的同时, 计算机软件也在悄悄地发生变化。这种变化不仅表现在计算机软件的内涵、计算机软件的应用方式上, 同时也表现在软件的设计模式和开发方法上。

站在软件开发人员的角度, 往往希望软件开发能够满足对于开发效率、可靠性、易维护性、易管理等多方面的更高要求。无论在计算机发展的哪一个时期, 这都是软件开发人员永恒的目标。可以把这种目标方法的实现分成以下几个阶段:

(1) 面向机器语言的开发模式。在计算机诞生的初期, 需要根据不同平台的机器语言来开发代码。这一阶段, 程序规模很小, 谈不上系统的方法。编程人员使用机器语言和汇编语言编写代码, 效率很低, 代码几乎没有可移植性。

(2) 软件的生命周期开发模式。又称软件开发瀑布模型, 是面向功能或过程的软件开发方法。它将软件开发分成几个阶段: ①用户需求分析, 开发人员和业务人员交流, 制定用户需求说明书; ②系统设计, 开发人员根据需求说明书进行系统设计, 制定系统设计报告; ③系统开发, 根据系统设计报告, 进行系统编码; ④系统测试, 系统实现后双方组织人员进行测试; ⑤运行与维护, 系统测试结束后, 便进入系统的运行、维护期。

从理论上讲, 软件开发生命周期模式是非常科学的, 但是利用生命周期模式开发系统基于两个假设: ①用户能够清楚地、完整地提供系统要求; ②开发者能完整地、严格地理解和定义要求。但在实际开发中, 这两个假设是很难满足的。因为, 在开发初期, 用户很难清楚地描述系统需求, 或者系统需求将来可能发生较大变化。其次, 开发人员和业务人员在交流时可能存在理解上的不一致, 其结果是系统开发完毕后, 不能很好地满足用户需求。生命周期模式是封闭式的, 缺少灵活性, 特别是在用户需求定义方面。

面向过程的生命周期法主要流行于 20 世纪 80 年代, 当时的编程语言主要是 C、Fortran 等过程式程序设计语言。相对于早期的汇编程序, 高级语言的出现已经是一种很大的进步。它极大地提高了编程效率, 提高了程序的可移植性。直到现在, 这种思想一直还对软件设计和开发人员有着很深的影响。

(3) 原型法开发模式。和生命周期模式不同, 原型法要求经过对用户需求的简单快速分析, 利用高级开发工具及环境, 快速完成原型系统的设计和实现, 提供给用户评价。一个原型系统就是系统的一个可运行的早期版本, 它反映了最终系统的部分重要特征, 在评价过程中, 开发人员不断从用户那里得到反馈信息, 修正原型的用户需求定义, 进而对原型系统作相应改进, 逐步减少分析与交互过程中的误解, 弥补遗漏, 从而提高最终系统的质量。

原型法的核心是用交互的、快速建立起来的原型取代形式的、不易修改的大块的规格说明, 用户通过在计算机上实际运行和试用原型而向开发者提供真实的反馈意见。原型法开发模式的实现得益于面向对象的语言 (Smalltalk、C++、Java 等) 和可视化的第四代开发工具的出现。从宏观上讲, 原型法比生命周期更实用, 但是在每一个原型的设计和开发过程中, 都离不开生命周期的科学思想。

在软件工程的实践中, 生命周期法和原型法的有效结合表现出了强大的生命力和可操作性。这种结合就是, 整个软件的开发表现为一个个原型的向前推进, 在每一个原型的内部, 又

是按照生命周期的思想来设计。

(4) 面向组件 (Component) 的模式。随着软件开发规模的扩大, 在涉及分布式、异构等复杂特征的环境中, 代码级别的重用性差、可维护性差、效率低的弱点是不可逾越的, 因此人们以架构运行环境 (如 .Net、J2EE 等) 来提供完善的支撑平台, 从而把开发者解放出来, 更专注于业务核心的开发。而这些业务功能 (Business Function) 以组件的形式 (DCOM、EJB 等) 发布运行在架构运行环境中, 软件开发的复用模式也上升到业务组件的级别。

(5) 面向方面编程 (Aspect Oriented Programming, AOP) 模式。AOP 是 OOP 的延续, 设计模式孜孜不倦追求的是调用者和被调用者之间的解耦, AOP 可以说也是这种目标的一种实现。AOP 使原本复杂的调用与被调用和错综复杂的耦合关系变得清晰, 使程序的整体架构保持高内聚, 低耦合, 这对于一个大型复杂系统来说是非常重要的。

(6) 面向服务 (SOA) 的模式。当软件的使用范围扩展到更广阔的范围, 往往会面对更加复杂的 IT 环境和更加灵活多变的需求。服务 (Service) 的概念出现了, 人们将应用 (Application) 以业务服务 (Business Service) 的形式公布出来供别人使用, 而完全不需要考虑这些业务服务运行在哪一个架构体系上, 因为所有的服务都讲着同样的语言。

SOA 同样也强调重用 (Reuse), 但是相对于传统的代码重用、对象重用和部件重用, SOA 的重用粒度更粗。SOA 的核心体现在企业应用或者业务功能上的“重用”和“互操作”, SOA 的重用在于业务级的应用, 即服务的重用, 而不再把 IT 与业务对立起来, 这可以被视为在 IT 驱动业务的方向上迈出的重要一步。

1.2 Web 及其工作原理

1990 年, 瑞士日内瓦世界上最大的粒子物理研究实验室欧洲核子研究中心 CERN (the European Organization for Nuclear Research) 提出了 World Wide Web (WWW) 的概念, 它是 Internet 技术、超文本技术和多媒体技术相结合的产物。当时, 核物理的研究是分散在不同国家进行的, 各地的研究人员通过计算机网络和 Internet 进行学术交流。在 Internet 中信息交流还没有一种统一的手段, 因此根据交流的信息不同 (如图片、文字等) 需要调用不同的 Internet 服务, 很不方便。1989 年 3 月, CERN 的 Tim Berners-Lee 开发了一个超级文本系统, 1990 年年底, 第一个基于字符界面的 Web 客户浏览程序开发成功, 1991 年 3 月, 客户浏览程序开始在 Internet 上运行, 1991 年年底 CERN 向高能物理学界宣布了 Web 服务。

1.2.1 什么是 Web

什么是 Web 呢? 从万维网诞生起, 人们并没有给它一个确切的定义。我们可以从 Internet 的构成和服务来理解 Web。

Internet 是一个网络上的网络, 或者说是一个全球范围的网间网。在 Internet 中, 分布了成千上万的无以计数的计算机, 这些计算机扮演的角色和所起的作用不同。有的计算机可以收发用户的电子邮件, 有的可以为用户传输文件, 有的负责对域名进行解析, 更多的机器则用于组织并展示本网络的信息资源, 方便用户的获取。所有这些承担服务任务的计算机统称为服务器。根据服务的特点来区分, 又分成邮件服务器、文件传输服务器、DNS 服务器、Web 服务器等。

所谓 Web 服务器, 就是将本地的信息用超级文本组织, 向用户提供在 Internet 上进行信息

浏览服务的计算机。因此，Web 或者说 World Wide Web，是由 Internet 中称为 Web 信息服务器的计算机组成的，它们由那些希望通过 Internet 发布信息的机构提供并管理。在 Web 世界里，每一个 Web 服务器除了提供自己独特的信息服务外，还可以用超链接指向其他的 Web 服务器。那些 Web 服务器又可以指向更多的 Web 服务器，这样一个全球范围的由 Web 服务器组成的 World Wide Web（万维网）就形成了。

1.2.2 Web 的工作原理

万维网的运行是一种典型的浏览器/服务器（Browser/Server, B/S）模式。典型的 B/S 体系结构将计算机应用分成三个层次，即：客户端浏览器层、Web 服务器层和数据库服务器层。B/S 体系结构有许多优点，它简化了客户端的维护，所有的应用逻辑都是在 Web 服务器上配置的。更主要的是，B/S 应用模式突破了传统的 C/S 模式中计算机应用对局域网的局限，从而将计算机应用分布到整个互联网中，用户可以在任何地方登录 Web 服务器，按照用户角色，执行自己的业务程序。

在万维网中，通过 HTTP 协议实现客户端（浏览器）和 Web 服务器的信息交换，Web 的基本工作原理如图 1-2 所示。

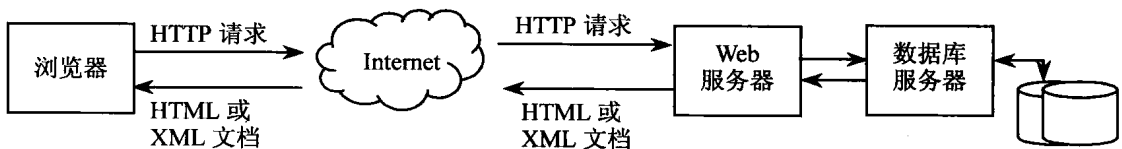


图 1-2 Web 的工作原理

在浏览器地址栏中，用户输入要访问的网页网址 URL（http://网址/路径/文件名.扩展名），向 Web 服务器提出 HTTP 请求。Web 服务器根据 URL 中指定的网址、路径和网页文件，调出相应的 HTML、XML 文档或 JSP、ASP 文件，根据文档的类型，Web 服务器决定是否执行文档中的脚本程序，还是直接将网页文件传送到客户端。

一般情况下，所有的 Web 应用几乎都要用到数据库管理数据。对数据库的管理和操作都是通过数据库服务器完成的，这些程序代码被以服务器端脚本的方式编写在 ASP、JSP 等页面中，负责和数据库服务器建立连接并完成必要的数据库查询、插入、删除、更新等数据库操作，然后利用获得的数据产生一个新的包含动态数据的 HTML 或 XML 文档，并将其发送给客户端 Web 浏览器。最后由 Web 浏览器解释该文档，在浏览器窗口中显示给用户。

1.2.3 Web 浏览器

Web 浏览器（browser）就是前面经常提到的 Web 客户端程序，用户要浏览 Web 页面必须在本地计算机上安装浏览器软件。通过在浏览器地址栏中输入 URL 资源地址，将 Web 服务器中特定的网页文件下载到客户端计算机中，并在浏览器打开。因此，从本质上讲，浏览器是一种特定格式的文档阅读器，它能够根据网页内容，对网页中的各种标记进行解释显示。此外，浏览器又是一种程序解释机，如果网页中包含客户端脚本程序，浏览器将执行这些客户端脚本代码，从而增强网页的交互性和动态效果。不同版本的浏览器都需要遵循 HTML 规范中定义

的标记集，同时为了便于脚本编程，每个浏览器程序本身也提供了相应的浏览器内置对象，类似于传统软件开发中的函数库及其标准库函数。

在 Web 发展初期，浏览器程序主要分成两类。一类为以 Lynx 为代表的基于字符的 Web 客户机程序，主要在不具备图形图像功能的计算机上使用。Lynx 是由美国堪萨斯大学的 Lou Montulli 研制的，同类的还包括 CERN 的 LineMode Browser。另一类是以 NCSA(National Center of Supercomputing Application) Mosaic 为代表的面向多媒体计算机的 Web 客户机程序，它可以在各种类型的小型机上运行，也可以在 IBM PC 机、Macintosh 机以及 UNIX 操作系统软件平台上运行。目前，使用较多的浏览器是微软的 IE (Internet Explorer) 浏览器。除此之外，一些新的浏览器产品也不断推向市场，如 Maxthon (傲游)、Firefox (火狐狸)、Opera 等。

1.2.4 Web 服务器

上面已经多次提到 Web 服务器，那么什么是 Web 服务器呢？我们知道，万维网是由分布在 Internet 中的 Web 服务器组成的。在 Internet 中，Web 服务器有两个层面的含义，一是指安装了 Web 服务器的计算机，二是指 Web 服务器程序。所谓 Web 服务器程序，简单地讲就是一个服务程序，它仅仅需要监听合适的端口，建立连接，然后发送数据给客户端。服务器程序的开发总是和客户端软件的开发相辅相成的。

要使一台计算机成为一台 Web 服务器，一般需要服务器操作系统，如 UNIX、Windows Server 2003、Linux 等网络操作系统，并且还要安装专门的信息服务器程序，如 Windows 中的 Internet 信息服务器 IIS (Internet Information Server)、Apache/Tomcat 等。Web 服务器的种类在一定程度上决定了 Web 应用的开发环境。

1.3 Web 相关技术

在互联网中，Web 几乎成为 Internet 的代名词，它给我们的工作和生活带来了无限的便利。在 Web 领域，新的理念不断产生，相关技术层出不穷，这在推动 Web 技术发展的同时，也无形中提高了 Web 应用开发的复杂性和入门的难度。从宏观上，建立一个准确的 Web 相关技术整体框架对于 Web 技术的理解，以及未来的 Web 应用开发具有极其重要的意义。

1.3.1 概念及术语

在 Web 中，新的概念、术语很多，随着 Web 应用的普及，这些本来是专业的概念和术语已经大众化了。下面从计算机专业的角度对 Web 中的一些常用概念进行简要介绍。

(1) 网站 (Web Site)。网站又称 Web 站点，是 Internet 中提供信息服务的机构，这些机构的计算机连接到 Internet 中，向用户提供 Web 服务。

从技术上讲，一个 Web 站点是由一个主目录和主目录下的文件夹和大量的网页文件构成的，这些网页文件通过超链接连接在一起，形成特定的应用逻辑，构成一个特定的 Web 应用。

(2) 超文本 (Hypertext)。超文本是一种文本显示与连接技术，可以对文本中的有关词汇或句子建立链接 (即超链接)，使其指向其他段落、文本或链接到其他文档。通过超链接，可以在文档之间、文档内部之间跳转，这种文本的组织方式与人们的思维方式和工作方式比较接近。

当超文本显示时，建立了链接的文本、图片通常以下划线、高亮等不同的方式显示，来