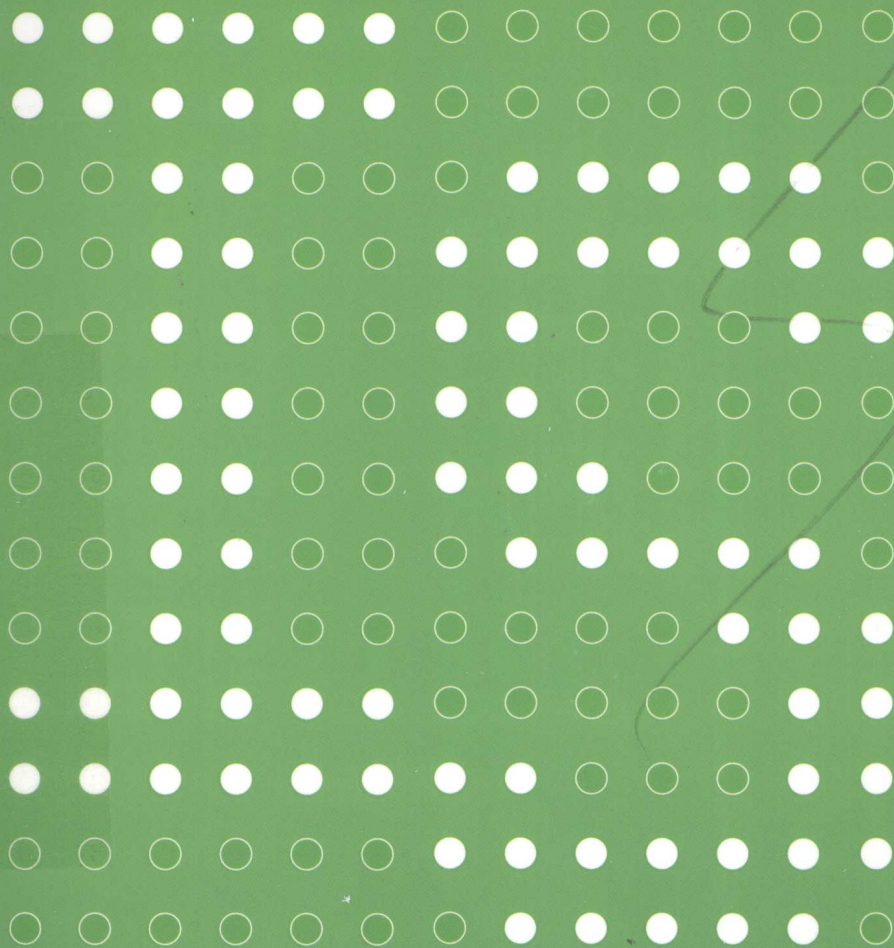


计算机科学与技术规划教材 信息系统方向

Web信息系统及其开发技术

石双元 主编

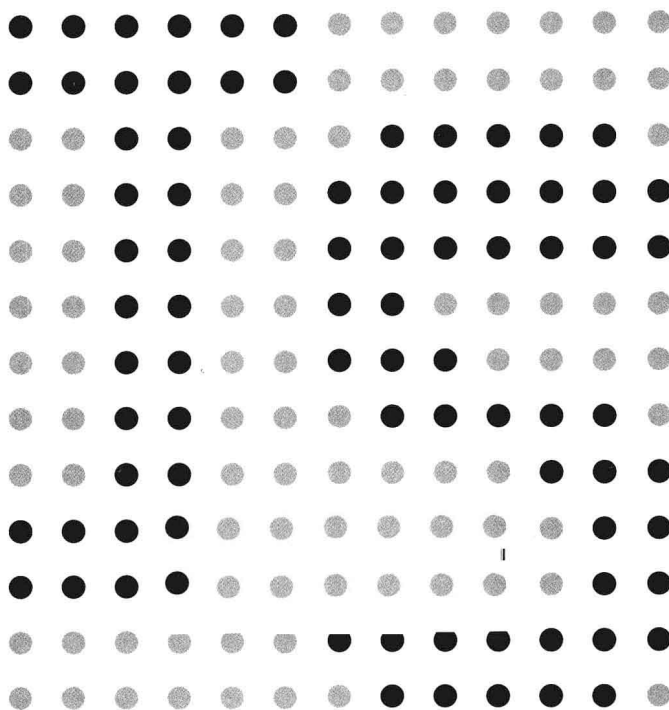


清华大学出版社
<http://www.tup.com.cn>

计算机科学与技术规划教材 信息系统方向

Web信息系统及其开发技术

石双元 主编



清华大学出版社

北京

内 容 简 介

本书以 Microsoft .NET 和 C# 语言为背景,系统、全面地介绍 Web 信息系统开发所涉及的内容和最新进展。全书共分 4 部分,即 Web 应用基础、服务器端开发、客户端开发和 Web 组件与系统开发。

Web 应用基础部分(第 1~4 章)主要介绍 Web 应用与开发的基本概念;服务器端开发部分(第 4~9 章)介绍 C# 语言基础知识、面向对象特性以及 Web 开发的背景知识,并全面介绍了 Microsoft .NET 的框架、控件和数据库开发技术;客户端开发部分(第 10~13 章)介绍客户端开发主流技术 JavaScript 及其对象,并详细介绍了 Ajax 技术及常用的框架与组件;Web 组件与系统开发(第 14 章)为高级应用部分,介绍了组件的设计与开发技术,并结合 EimsSite 组件介绍了企业 Web 应用开发实例。

本书可作为高等学校信息管理与信息系统、电子商务、工商管理及管理学以及计算机科学与技术等专业的教材,也可供相关专业技术人员、工程技术人员等参考。

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 信息系统及其开发技术 / 石双元主编. —北京: 清华大学出版社, 2012. 5

(计算机科学与技术规划教材 信息系统方向)

ISBN 978-7-302-28053-8

I. ①W… II. ①石… III. ①计算机网络—程序设计—高等学校—教材 IV. ①TP393.09

中国版本图书馆 CIP 数据核字(2012)第 025855 号

责任编辑: 张瑞庆 战晓雷

封面设计: 常雪影

责任校对: 胡伟民

责任印制: 何 芊

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>, 010-62795954

印 刷 者: 清华大学印刷厂

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 32.75 字 数: 820 千字

版 次: 2012 年 5 月第 1 版 印 次: 2012 年 5 月第 1 次印刷

印 数: 1~3000

定 价: 49.00 元

产品编号: 039178-01

前 言

Web 技术及其应用已发展成为新一代信息技术,并从传统 IT 和互联网脱颖而出,直接催生了云计算。大部分企业用户接触到的云计算是软件即服务,即通过网络向企业提供基础设施、计算和信息服务。可见,基于互联网的 Web 信息系统是云计算的核心。

Web 信息系统是随着 Web 应用的普及和深入而形成的研究企业应用领域 Web 应用系统开发的理论与方法的综合性技术。企业应用系统有两种主要模式,一种是基于 C/S 结构的桌面应用,另一种就是基于 B/S 结构的 Web 应用。随着互联网的普及和广泛应用,基于 B/S 三层结构的 Web 应用框架将成为市场的主流。近年来,Web 应用模式所占的比重正在逐年增加,尽管在近期内不可能完全取代桌面应用,但它必将是未来企业应用系统的主流和发展方向。

以 Ajax 和 JavaScript 为核心的 Web 开发框架与组件逐渐成为企业信息系统开发的新一代信息技术。一个新的信息系统时代即将来临,我们正处在从桌面应用到 Web 应用迁移的转折阶段。虽然这一转变仍处于早期,但它将最终改变企业应用领域中的信息获取与分享方式。不过,令人遗憾和尴尬的是,目前这种应用模式和技术方法的转变似乎只是出现在软件公司中,而在高等学校专门讲授 Web 开发的课程还很少,即使一些高校开设了 Web 课程,也只是从网站建设的角度来介绍,涉及的内容很浅,没有上升到应用系统开发的高度。

Web 应用开发技术是一门综合性技术课程,它涉及的内容非常多,如 HTML、CSS、C#(或 Java、PHP 等)、JavaScript、XML、JSON、DOM 及其最新的 Ajax 技术等,并且这些技术是相互关联和依存的,其中的每一项技术涉及的内容也非常丰富。尽管介绍 Web 开发技术方面的书籍很多,但其中绝大多数是突出单项技术,无法使学生系统、全面地了解 Web 应用开发技术的全貌。这也是编写本书的难点所在,Web 应用涉及的技术太多,要在有限的篇幅和课时的条件下全面安排这些内容是一件非常困难的事情,尽管我们力求使其合理,但不得不舍弃很多。

根据不同的开发与应用平台,Web 应用与开发又有两大分支,一是基于 Java 平台,二是基于 Microsoft .NET 平台。不过,这两种平台也有很多的共性,比如,都涉及 HTML、XML 和数据库技术等,并且其客户端技术正在融合。本书以 Microsoft .NET 框架和 C# 语言为背景,围绕 Web 信息系统开发所涉及的技术由浅入深地展开,力图系统、全面地介绍 Web 信息系统所涉及的内容和最新进展。在内容和结构安排上力求做到系统性和连贯性。全书共分 4 大部分,即 Web 应用基础、服务器端开发、客户端开发和 Web 组件与系统开发。

Web 应用基础部分(第 1~4 章)主要介绍 Web 应用与开发的基本概念;服务器端开发部分(第 4~9 章)介绍 C# 语言基础知识与面向对象特性以及 Web 开发的背景知识,并全

面介绍了 Microsoft .NET 的框架、控件和数据库开发技术,按照由浅入深的顺序,分别阐述了 .NET 页面程序结构、Web 服务器控件、.NET 中的数据绑定技术以及控件与代码分离技术。

客户端开发部分(第 10~13 章)介绍了客户端开发主流技术 JavaScript 及其对象,并详细介绍了 Ajax 技术及常用的框架与组件,如 Microsoft .NET Ajax。

Web 组件与系统开发(第 14 章)作为高级应用部分,介绍了组件的设计与开发技术,并结合 EimsSite 组件介绍了企业 Web 应用开发实例。

本书可作为高等院校信息管理与信息系统、电子商务、工商管理及管理学以及计算机科学与技术等专业的教材和参考书,也可供相关专业人员、工程技术人员等参考。

本书由华中科技大学石双元编写。研究生张巨松、李仲辉、冯亚维、臧佳鹏、邱宗国、卢忠媛在实例调试和文字校对方面做出了大量的工作,在此表示感谢。本书的编写得到了华中科技大学管理学院蔡淑琴教授的大力帮助和指点,以及清华大学出版社张瑞庆编审的大力支持和帮助,在此对她们表示诚挚的谢意。在本书的编写过程中参考和引用了一些国内外文献资料,在此对相关作者表示深深的感谢。

由于本书的写作时间较短,以及编者水平有限,书中难免存在缺点和错误,敬请各位专家和读者批评指正。

石双元
2011. 12

目 录

第 1 章 Web 信息系统及其特征	1
1.1 信息系统的发展与演变	1
1.1.1 集中式计算模式	1
1.1.2 Client/Server 模式	1
1.1.3 Browser/Server 模式	2
1.2 Web 应用及其特征	3
1.2.1 什么是 Web 应用	4
1.2.2 Web 网站与 Web 应用	4
1.3 Web 应用的相关技术	5
1.4 Web 应用的分类及其差异	7
1.4.1 用户群体与使用行为的差异	8
1.4.2 用户使用模式的差异	8
1.4.3 业务处理模式的差异	9
1.4.4 Web 应用与桌面应用的差异	9
1.5 Web 应用与企业云计算	10
1.5.1 分散性经营模式与封闭式信息管理的矛盾	10
1.5.2 企业 Web 应用现状及问题	10
1.5.3 云计算及其在企业中的应用	11
第 2 章 Web 应用基础知识	15
2.1 HTML 概要	15
2.1.1 HTML 文档的基本结构	15
2.1.2 HTML 文档编辑与运行	17
2.1.3 HTML 文档结构标签	18
2.1.4 版面风格控制	21
2.1.5 HTML 中显示特殊字符	27
2.1.6 文件的链接	27
2.1.7 HTML 表单	30

2.1.8 HTML 样式表	34
2.2 XML 结构及语法	38
2.2.1 XML 及其特征	38
2.2.2 XML 文档结构	41
2.2.3 XML 文档格式定义	44
2.2.4 XML 文件的显示与转换	47
2.3 ASP 动态页面技术	52
2.3.1 静态网页与动态网页	52
2.3.2 ASP 及其特征	54
2.3.3 ASP 文件结构	54
2.3.4 ASP 页面的运行环境	55
2.3.5 VBScript 语法简介	56
2.3.6 ASP 与 ADO 的连接	61
2.3.7 ASP 内置对象	67
2.3.8 JavaScript、JScript 和 VBScript 的简要说明和比较	71
第 3 章 页面控制与 UI 设计	73
3.1 什么是 CSS	73
3.2 CSS 的基本结构	73
3.2.1 CSS 样式规则	73
3.2.2 CSS 的基本模型	74
3.2.3 CSS 的引用方式	76
3.2.4 CSS 使用例子	77
3.2.5 CSS 的工作原理	78
3.3 CSS 常用属性	78
3.3.1 背景属性(background)	78
3.3.2 文本属性	80
3.3.3 字体属性(Font)	81
3.3.4 列表(List-style)	83
3.3.5 表格属性	84
3.3.6 CSS 链接属性	85
3.4 CSS 选择器	86
3.4.1 CSS 元素选择器	86
3.4.2 CSS 类选择器	86
3.4.3 CSS ID 选择器	88
3.5 伪类和伪元素	88
3.6 组织元素(span 和 div)	91
3.6.1 用 span 组织元素	91
3.6.2 用 div 组织元素	92

3.7 CSS 盒状模型(Box Model)	93
3.7.1 什么是 CSS 盒状模型	93
3.7.2 margin 属性	95
3.7.3 padding 属性	96
3.7.4 margin 与 padding 的区别	97
3.7.5 边框(border)属性	97
3.8 CSS 定位(Positioning)	100
3.8.1 相对定位	100
3.8.2 绝对定位	100
第 4 章 C# 语言基础	102
4.1 C# 语言及其特点	102
4.2 中间语言代码与 .NET 的跨平台	103
4.3 一个简单的 C# 程序	104
4.4 命名空间	108
4.4.1 什么是命名空间	108
4.4.2 命名空间的引用	108
4.5 C# 程序的构成	109
4.6 C# 的类和对象定义	111
4.7 C# 的格式与数据类型	113
4.7.1 C# 的标识符	113
4.7.2 书写规则	114
4.7.3 C# 的数据类型	115
4.7.4 数据类型转换	117
4.7.5 常量	118
4.7.6 变量	119
4.7.7 枚举与数组	121
4.8 运算符与表达式	124
4.8.1 运算符	124
4.8.2 表达式	128
4.9 方法定义与使用	128
4.10 方法的定义	130
4.11 流程控制语句	131
4.11.1 结构化程序设计的 3 种基本流程	131
4.11.2 分支语句	131
4.11.3 循环语句	134
4.11.4 跳转语句	136
4.11.5 递归调用	138

第 5 章 C# 的面向对象特性	140
5.1 类的定义与 C# 的修饰符	140
5.1.1 类的定义	140
5.1.2 C# 修饰符	142
5.2 创建对象与构造方法	143
5.2.1 创建对象	143
5.2.2 构造方法	144
5.2.3 析构方法	147
5.2.4 类的修饰符	148
5.2.5 类的嵌套	150
5.3 类的成员及其修饰符	151
5.3.1 变量及其修饰符	151
5.3.2 方法及其修饰符	157
5.4 访问控制符	162
5.4.1 公共访问控制符 public	162
5.4.2 默认访问控制符	163
5.4.3 保护访问控制符 protected	163
5.4.4 私有访问控制符 private	163
5.5 C# 的继承及其实现	165
5.5.1 继承的定义	165
5.5.2 继承的实现	166
5.6 C# 语言中的多态与实现	168
5.6.1 什么是多态	168
5.6.2 构造方法的继承与重载	170
5.6.3 C# 多态的实现	171
5.7 接口及其实现	173
5.7.1 单重继承与多重继承	173
5.7.2 C# 接口概述	173
5.7.3 接口的定义	174
5.7.4 实现接口	175
5.8 异常处理	176
5.8.1 异常产生的原因	176
5.8.2 捕获异常与抛弃异常	177
第 6 章 .NET 开发平台及服务器控件	179
6.1 .NET 及其特征	179
6.1.1 从 ASP 到 .NET	179
6.1.2 ASP.NET 页面及其主要特性	180

6.1.3 .NET 框架概述	181
6.2 .NET 环境配置	182
6.2.1 Web 服务器安装与配置	182
6.2.2 ASP.NET 开发环境的配置	187
6.2.3 .NET 常用文件类型	187
6.3 .NET 中的类与命名空间	188
6.3.1 .NET 中的类	188
6.3.2 .NET 的命名空间	188
6.4 .NET 的页面表单	189
6.4.1 .NET 的页面表单	189
6.4.2 .NET 页面的程序结构	192
6.4.3 控制页面导航	197
6.5 Web 服务器基础控件	200
6.5.1 Web 服务器控件	200
6.5.2 Web 控件的基本属性	203
6.5.3 文本类控件及其使用	204
6.5.4 按钮类控件的属性和方法	207
6.5.5 列表类控件的属性与方法	209
6.5.6 服务器基础控件综合应用	223
6.6 .NET 验证控件	225
6.6.1 表单信息的验证	225
6.6.2 常用验证控件的属性与方法	226
6.7 NET 高级控件	230
6.7.1 .NET 容器控件	230
6.7.2 树形视图 TreeView 控件	237
6.7.3 工具条 ToolBar 控件	243
6.7.4 选项卡 TabStrip 控件	244
第 7 章 ADO.NET 与数据绑定	248
7.1 ADO.NET 的特征及主要对象	248
7.1.1 ADO.NET 及其特征	248
7.1.2 ADO.NET 的主要对象	249
7.2 ADO.NET 的访问方法	250
7.3 .NET 数据库高级访问技术	256
7.3.1 在 ASP.NET 中实现动态查询	256
7.3.2 在查询中使用参数	258
7.3.3 使用 SQL 存储过程	259
7.4 DataSet 对象的使用	262
7.4.1 DataReader 与 DataSet	262

7.4.2 使用 DataAdapter 对象填充数据集	263
7.5 ASP.NET 中的数据绑定	265
7.5.1 什么是数据绑定	265
7.5.2 绑定数据源到列表控件	266
7.5.3 利用数据源控件绑定数据	271
7.6 .NET 数据源控件与数据显示控件	272
7.7 用 Repeater 控件显示数据	277
7.7.1 绑定 Repeater 控件到数据源	277
7.7.2 在 Repeater 控件中使用模板	279
7.7.3 在 Repeater 控件内绑定其他控件	280
7.7.4 用 Repeater 控件实现主从表	282
7.8 用 DataList 控件显示数据	284
7.9 用 GridView 显示数据	286
7.9.1 绑定 GridView 控件到数据源	287
7.9.2 在 GridView 控件中创建列	289
7.9.3 在 GridView 控件中实现排序	295
7.9.4 在 GridView 中实现分页	297
7.10 Repeater、DataList 与 GridView 的比较	300
第 8 章 程序代码与页面分离技术	302
8.1 创建和使用 .NET 组件	302
8.1.1 业务组件及其优势	302
8.1.2 .NET 组件的建立	303
8.1.3 ASP.NET 组件的使用	305
8.1.4 使用和访问组件中类的属性	306
8.2 代码隐藏技术	310
8.3 用户控件	313
8.3.1 用户控件的定义与使用	314
8.3.2 使用用户控件的属性与方法	316
8.4 自定义控件	318
8.4.1 自定义控件及其特性	318
8.4.2 为控件添加属性和方法	321
第 9 章 Web Service 与业务逻辑计算	325
9.1 Web Service 及其特征	325
9.1.1 Web Service 产生的背景	325
9.1.2 Web Service 及其作用	326
9.1.3 Web Service 的优势与特点	328

9.2 Web Service 平台的相关技术	330
9.2.1 Web Service 数据描述格式(XML)	331
9.2.2 Web Service 的信息交换的协议(SOAP)	331
9.2.3 Web Service 描述语言(WSDL)	332
9.2.4 Web Service 注册与查找机制(UDDI)	335
9.3 创建与使用 Web Service	336
9.3.1 创建 Web Service 的方法	336
9.3.2 创建一个简单的 Web 服务	338
9.3.3 创建一个简单功能的 Web Service	340
9.3.4 在 Web 应用程序中调用 Web Service	342
9.3.5 调用其他 Web 服务器中的 Web Service	344
9.3.6 动态调用 Web 服务	345
9.4 创建数据库 Web Service	347
9.5 在客户端进行 Web Service 调用	351
9.5.1 使用 Ajax 调用 Web Service	351
9.5.2 客户端调用页面的 Web 方法	353
第 10 章 JavaScript 语言基础	356
10.1 JavaScript 语言概述	356
10.1.1 什么是 JavaScript	356
10.1.2 JavaScript 的发展历程与特征	357
10.1.3 JavaScript 和 Java 的区别	358
10.1.4 编写第一个 JavaScript 程序	359
10.2 JavaScript 语法基础	362
10.2.1 数据类型与变量	362
10.2.2 表达式和运算符	367
10.3 JavaScript 程序结构	368
10.3.1 程序控制流程	368
10.3.2 JavaScript 函数与事件	372
10.4 JavaScript 中的 JSON	377
10.4.1 JSON 及其描述方法	377
10.4.2 JSON 与 XML 的比较	381
10.4.3 JSON 对象与字符方式转换	383
第 11 章 JavaScript 对象简介	386
11.1 JavaScript 对象的特征	386
11.2 JavaScript 对象的操作	387
11.2.1 JavaScript 对象的引用	387
11.2.2 JavaScript 对象的操作语句	388

11.3	JavaScript 对象与函数	390
11.3.1	JavaScript 对象的创建	390
11.3.2	JavaScript 的函数对象	393
11.4	JavaScript 内部对象	396
11.4.1	JavaScript 的 String(字符串)对象	396
11.4.2	JavaScript 的 Date(日期)对象	399
11.4.3	JavaScript 的 Boolean(逻辑)对象	401
11.4.4	JavaScript 的 Number(数值)对象	402
11.4.5	JavaScript 的 Array(数组)对象	403
11.4.6	JavaScript 的 Math 对象	406
11.4.7	JavaScript 中的系统函数	408
11.5	JavaScript HTML DOM 对象	409
11.5.1	什么是 HTML DOM 对象	409
11.5.2	Document 对象的属性与方法	411
11.5.3	Document 节点对象的访问	413
第 12 章	基于 Ajax 的 Web 应用	418
12.1	传统 Web 的弊病	418
12.2	Ajax 带来的变化及优势	419
12.2.1	Ajax 带来的变化	419
12.2.2	Ajax 技术的优势	420
12.3	Ajax 及技术实现	421
12.3.1	Ajax 简介	421
12.3.2	Ajax 工作原理	422
12.3.3	第一个 Ajax 程序	423
12.3.4	Ajax 核心对象 XmlHttpRequest	425
12.3.5	一个典型的 Ajax 调用过程	429
12.4	Ajax 请求与响应实例	430
12.5	基于 Ajax 的 Web 应用实例	433
12.5.1	Ajax 访问数据库实例	434
12.5.2	Ajax 访问 XML 数据文件实例	436
12.5.3	Ajax ResponseXML 应用实例	439
12.6	Ajax 与传统的 Web 应用与开发方式的差异	441
12.6.1	数据加载模式的比较	441
12.6.2	Ajax 编程的技术难点	442
第 13 章	Ajax 框架与组件	444
13.1	Ajax 框架与组件简介	444
13.2	ASP.NET Ajax 的组成	446

13.3	核心组件 ASP.NET Ajax Extensions	451
13.3.1	ScriptManager 控件	451
13.3.2	UpdatePanel 控件	453
13.3.3	Ajax 与 Web Service 引用	459
13.4	Ajax Library 类库	464
13.5	应用工具包 AjaxControlToolkit	468
13.6	Ajax 编程的本质及开发途径	470
第 14 章	企业级开发组件及应用	472
14.1	企业级 Web 应用及关键问题	472
14.1.1	企业应用从桌面向 Web 迁移的必然性	472
14.1.2	企业级 Web 应用开发所面临的挑战	473
14.2	基于组件的 Web 应用开发	474
14.3	企业级组件框架及其开发技术	476
14.3.1	企业 Web 应用开发套件 EimsSuite	476
14.3.2	.NET 组件设计基本方法	476
14.3.3	EimsSuite 设计与开发	478
14.3.4	功能模块详解	484
14.3.5	EimsGrid 典型功能的详细设计与技术实现	488
14.3.6	控件的封装	494
14.4	EimsSuite 套件的功能及主要特征	498
14.5	EimsGrid 的使用方法	499
14.6	EimsGrid 控件应用实例	500
参考文献		508

1.1 信息系统的发展与演变

作为信息技术在管理领域应用的一个重要分支,管理信息系统(或称为信息系统)已在管理中起到了举足轻重的作用。随着网络与信息技术的广泛使用,信息系统不再是锦上添花或可有可无的摆设,而是支持企业运作的基础条件,很多先进的管理理念与方法也必须借助于信息系统才能实现。在企业信息化建设实践中,各种先进的信息技术得到了广泛应用,尤其是从互联网中派生出的 Web 技术,为新技术背景下的企业信息化的理论和实践带来了突破,互联网技术使企业信息系统突破时间和空间的限制,能在更广泛的范围实现信息的实时分享,并使得处于不同地域的员工能在统一的平台上协调工作。

根据不同的计算模式,企业信息系统主要有两种主流结构模式,即客户端/服务器(Client/Server,C/S)模式和浏览器/服务器(Browser/Server,B/S)模式。信息系统的计算模式与计算机技术密切相关,随着计算机技术的发展,计算模式经历了从集中式计算、C/S 计算发展到目前最流行的 B/S 计算模式的发展。下面简要回顾计算模式的发展历程,并分析比较 C/S 模式与 B/S 模式。

1.1.1 集中式计算模式

第一代计算模式是集中式计算(即主机模式),其数据和应用逻辑在一台主机上,由大型机和多个与之相连的哑终端组成。哑终端(Dumb Terminal)即功能有限的计算机终端,本身几乎没有处理能力,只被用作文字编辑或向中央计算机提出需求。这种计算模式的主要优点是:安全性好,可靠性高,计算能力和数据存储能力强,系统维护和管理费用较低。但是它也存在着一些明显的缺点,例如,硬件的初始投资高,可移植性差,资源利用率低,网络负载大。集中式计算模式仍是一些特定应用场合下重要的计算处理方式,如金融领域等。尽管其他的计算模式在计算能力和可伸缩性方面已接近大型机,但当支持大量用户和数据时,大型机自顶向下的维护和管理方式仍显示着集中式处理的优越性。

1.1.2 Client/Server 模式

Client/Server 模式是 20 世纪 80 年代末逐步成长起来的一种计算模式。随着 PC 的普及,计算模式从集中式转向了分布式(数据和应用逻辑跨越多个节点),尤为典型的是 Client/Server 结构(简称 C/S 结构)。与集中式模式相比,Client/Server 的特点在于功能的分布,Client 与 Server 各自具有不同的功能角色,是分布式的应用环境。C/S 结构的最大优

点是能充分发挥客户端计算机的处理能力,并且其客户端响应速度快。

Client/Server 的发展经历了两个阶段:从两层结构到三层结构。两层 C/S 结构由两部分构成:前端是客户机,通常是 PC,主要完成用户界面显示,接受数据输入,校验数据有效性,向后台数据库发出请求,接受返回的结果,并处理应用逻辑;后端是服务器,运行 DBMS,提供数据库的查询和管理。其应用逻辑主要在客户端,如在服务器端则是以存储过程的形式存在于 DBMS 中。两层 C/S 结构在上世纪 80 年代及 90 年代初得到了广泛应用。

但两层 C/S 结构存在以下几个局限。

- (1) 它是单一服务器且以局域网为中心的,难以扩展至广域网;
- (2) 软、硬件的组合及集成能力有限;
- (3) 难以管理大量的客户机。

因此,三层 C/S 结构应运而生。

三层 C/S 结构是将应用分为表示层、功能层和数据层 3 部分。其解决方案是对这三层进行明确分割,并在逻辑上使其相对独立。原来的数据层作为 DBMS 已经独立出来,其关键是将表示层和功能层分离成各自独立的程序,并且这两层间的接口简洁明了。

表示层是应用的用户接口部分,它负责用户与应用间的对话功能,用于接受和检查用户输入的数据,并显示应用输出的数据,检查的内容也只限于数据的形式和值的范围,不包括业务本身的处理逻辑。在变更用户接口时,只需改写显示控制和数据检查程序,而不影响其他两层。

功能层是应用的主体,实现具体的业务处理逻辑,而处理所需的数据则从表示层或数据层取得。通常,在功能层中还包含有确认用户对应用和数据库的存取权限,以及记录系统处理日志的功能。表示层和功能层之间的数据交换应尽可能简洁。

数据层就是 DBMS,负责管理对数据库数据的读写操作,一般从功能层传送到数据层的请求大都使用 SQL 语言。

与两层 C/S 结构相比,三层 C/S 结构具有如下优势。

(1) 具有灵活的硬件系统构成。对于各个层可以选择与其处理负荷和处理特性相适应的硬件。例如,最初用一台 UNIX 工作站作为服务器,将数据层和功能层都配置在这台服务器上。随着业务的发展,用户数和数据量逐渐增加,这时就可以将 UNIX 工作站作为功能层的专用服务器,另外追加一台专用于数据层的服务器。若业务进一步扩大,用户数进一步增加,则可以继续增加功能层的服务器数目。清晰、合理地分割三层结构并使其独立,可以使系统构成的变更非常简单。

(2) 提高程序的可维护性。在三层 C/S 结构中,应用的各层可以并行开发,各层也可以选择各自最适合的开发语言。

(3) 利于变更和维护应用技术规范。因为是按层分割功能,所以各个程序的处理逻辑变得十分简单。

(4) 进行严密的安全管理。在三层 C/S 结构中,识别用户的机构是按层来构筑的,对应用和数据的存取权限也可以按层进行设定。此外,系统管理简单,可支持异种数据库。

1.1.3 Browser/Server 模式

随着互联网的飞速发展,移动办公越来越普及,这就要求支持企业运作的信息系统具有

扩展性和远程数据处理能力。而 C/S 结构的远程访问需要专门的技术,不仅成本高,而且要对系统进行专门的设计来处理分布式的数据。其原因在于,C/S 体系结构虽然采用的是开放模式,但仍是在相对封闭的环境下运行,无论是 Client 端还是 Server 端都还需要特定的软件支持,没能提供用户真正期望的开放环境,已经很难适应互联网环境下企业对分散信息处理的需求和不同区域员工协同工作的需要,而且代价高、效率低。

浏览器/服务器模式(Browser/Server, B/S)是把传统的 C/S 模式中的 Client 作为 WWW 浏览器,把 Server 分解成一个 Web 服务器和一个或多个数据库服务器,实质上是由各种服务器群和 WWW 浏览器组成。B/S 结构是三层 C/S 结构向互联网的延伸,第一层是表示层,第二层是功能层,第三层是数据层。

表示层(WWW 浏览器):主要完成用户接口的功能,即在客户端向由 URL(Uniform Resource Locator,统一资源定位器)所指定的 Web 服务器提出服务申请,Web 服务器对用户进行身份验证后,用 HTTP 协议把所需的文件资料传送给用户,客户端只是接收文件资料,并显示在 WWW 浏览器上。

功能层(即 Web 服务器):主要利用服务器完成客户的应用功能,即 Web 服务器接受客户请求,并与数据库连接进行请求处理,而后将处理结果返回 Web 服务器,再由 Web 服务器以页面的形式传至客户端。

数据层(数据库服务器):数据库服务器应客户请求独立地进行处理。

C/S 结构是以服务器为中心进行服务的,B/S 结构把 C/S 结构中的服务器从 LAN 中脱离出来,移到更大的网络上,使服务器独立于物理位置,在这种模式中,提供服务的是分布在不同地点的网络上的多台服务器,这样就把 C/S 服务模式变成了客户/网络(Client/Network)服务模式。因此 B/S 结构是以网络服务的形式实现企业信息的收集与发布服务。

该结构的特点是:

- (1) 把应用从客户机中分离出来,使之不再支持应用,变成一个简单的客户机或浏览器;
- (2) 摆脱了由于客户有多个应用而造成的复杂运行环境的维护;
- (3) 应用的增加、删减、更新不影响用户个数和执行环境;
- (4) 当来自客户端的访问频繁,造成第三层的服务器负荷过重时,可分散、均匀负荷而不影响客户环境。

企业的信息系统可以采用统一的模式,也可包含不同的模式,但应确定相应的互联规范,互联规范的主要内容包括:采用何种通信协议、接口方式、传输能力及相应的联网设备。

1.2 Web 应用及其特征

Web 应用是从 Web 页面演变而来的。最初的 Web 页面只是信息显示终端。随着网络技术的发展,页面上具有了计算功能,网页的表现更像应用程序,从而带来更快速的响应时间以及更好的呈现形式。技术和设计的改变,模糊了应用程序和网站的界限,使得建立基于互联网的 Web 应用系统成为可能。