

国家骨干高职院校建设成果·计算机项目化系列教材

Web应用系统设计

亢华爱 主编 张海建 马东波 副主编

清华大学出版社

国家骨干高职院校建设成果·计算机项目化系列教材

Web应用系统设计

亢华爱 主编 张海建 马东波 副主编

清华大学出版社
北 京

内 容 简 介

在基于 Java 的 Web 应用系统开发中, J2EE 的 SSH(即 Struts、Spring 和 Hibernate)架构是当前的主流技术。本书以人事管理系统的开发为主线, 分别介绍 SSH 的理论基础并且使用 SSH 来构建简单的 Web 应用系统。本书的第 1、2 章介绍 Java Web 开发基础和开发环境的搭建; 第 3~5 章介绍 Struts 2 框架及其应用; 第 6~8 章介绍 Hibernate 框架及其应用; 第 9、10 章介绍 Spring 框架及其应用; 第 11 章是整合 Struts 2、Hibernate、Spring 框架的综合应用; 第 12 章给出人事管理系统中证件信息管理模块的需求、概要设计说明、详细设计说明等信息, 作为最后的实训环节的依据。

本书适合 Java Web 开发技术的初学者使用, 也可作为高等职业院校计算机专业的教材, 还可作为自学参考资料。

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 应用系统设计/亢华爱主编. —北京: 清华大学出版社, 2013. 2

(国家骨干高职院校建设成果. 计算机项目化系列教材)

ISBN 978-7-302-31109-6

I. ①W… II. ①亢… III. ①网页制作工具—高等职业教育—教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2012)第 308681 号

责任编辑: 刘翰鹏

封面设计: 何凤霞

责任校对: 李 梅

责任印制: 宋 林

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>, 010-62795764

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: 13.5 字 数: 308 千字

版 次: 2013 年 2 月第 1 版 印 次: 2013 年 2 月第 1 次印刷

印 数: 1~3000

定 价: 27.00 元



FOREWORD

J2EE 技术自产生以来得到了广泛的应用和认可。随着技术的演变, J2EE 技术平台已经日趋成熟, 成为当今电子商务平台开发的最佳解决方案。在 J2EE 技术中, 使用 Struts+Hibernate+Spring 进行整合开发被越来越多的开发者使用。

本书的主要内容是讲解 Struts 2、Hibernate、Spring 框架技术的理论基础与简单的 Web 应用。对于每个框架, 先介绍它的技术基本概念和基础知识, 然后再提供具体的示例项目来展示这个框架技术的使用方法, 最后将这三个框架进行整合实现一个基于 Spring、Hibernate 和 Struts 2 的 Web 应用系统。

本书共分 12 章, 采用理论和实践结合的方式来讲解 Struts 2、Hibernate、Spring 框架技术。

第 1 章对 Web 开发的基础理论、MVC 模式的设计思想和本书所采用的项目进行介绍。

第 2 章对系统开发前需要做的前期准备进行介绍, 包括运行环境和开发环境的搭建。

第 3 章通过简单登录实例介绍如何开发 Struts 2 的简单应用程序。

第 4 章介绍 Struts 2 的处理流程、Struts 2 的基本配置以及 Struts 2 的核心工作原理及配置文件的使用。

第 5 章以人事管理系统中的职称类别管理模块为例, 介绍如何进行 Struts 2 程序的开发。

第 6 章介绍 Hibernate 的相关概念、持久化技术以及实现方法, 对比不同持久化实现方法, 以及介绍 Hibernate 的结构和接口作用。

第 7 章通过实例的方式介绍如何应用 Hibernate 框架。

第 8 章通过完成人事管理系统中的用户管理模块的开发来介绍 Struts 与 Hibernate 的集成应用。

第 9 章先从 Spring 框架底层模型的角度描述该框架的功能, 然后介绍 Spring 面向切面编程和控制反转容器。

第 10 章以实例的方式演示控制反转、依赖注入和 Spring 持久化的应用。

第 11 章整合 Struts 2、Hibernate、Spring 框架, 开发人事管理系统中的部门管理



模块。

第12章给出人事管理系统中证件信息管理模块的需求说明、概要设计说明、详细设计说明、Web应用体系结构说明及证件信息管理模块的关键代码,以此作为最后实训环节的依据。

其中,第1、9~12章由亢华爱编写,第2~5章由张海建编写,第6~8章由马东波编写。全书由亢华爱统稿,王广峰审稿。由于时间仓促、水平有限,不当之处还望各位专家和读者批评指正。

编 者

2012年9月

目 录

CONTENTS

第 1 章 Web 应用程序开发基础	1
1.1 Web 开发概述	1
1.1.1 Web 技术的发展	1
1.1.2 静态 Web 和动态 Web 的区别与联系	2
1.1.3 Web 应用系统的开发模式	3
1.1.4 Model 1 和 Model 2	3
1.2 MVC 设计思想	4
1.3 项目概述	6
小结	8
习题	8
第 2 章 搭建运行和开发环境	9
2.1 搭建运行环境	9
2.1.1 下载并安装 JDK	9
2.1.2 Java 环境变量的设置	9
2.1.3 Tomcat 安装和配置	10
2.2 搭建开发环境	11
2.2.1 Eclipse 的安装	11
2.2.2 MyEclipse 插件的安装	12
2.3 使用 MyEclipse 创建 Web 项目	14
2.4 在开发环境中配置 Tomcat 服务器	18
小结	19

第 3 章 基于 Struts 2 的简单程序	20
3.1 Struts 概述	20
3.2 获取 Struts 2	21
3.3 基于 Struts 2 框架实现登录实例	21
3.3.1 创建一个新的 Web 项目	21
3.3.2 增加 Struts 2 支持	22
3.3.3 配置 web.xml 文件	23
3.3.4 从页面请求开始	24
3.3.5 部署 Struts 2 应用	26
3.3.6 实现控制器	28
3.3.7 改进控制器	31
小结	33
习题	33
第 4 章 Struts 2 体系	34
4.1 Struts 2 框架架构	34
4.2 Struts 2 的基本配置	35
4.2.1 配置 web.xml 文件	35
4.2.2 配置 Action 的 struts.xml 文件	36
4.2.3 配置 Struts 2 全局属性的 struts.properties 文件	38
4.3 Struts 2 的标签库	38
4.4 Struts 2 组件	39
4.4.1 Struts 2 的核心控制器: FilterDispatcher	39
4.4.2 业务控制器	40
4.4.3 Struts 2 的模型组件	41
4.4.4 Struts 2 的视图组件	41
4.5 Struts 2 的配置文件	41
4.5.1 常量配置	41
4.5.2 包配置	42
4.5.3 命名空间配置	43
4.5.4 包含配置	44
4.5.5 拦截器配置	44
小结	45
习题	45
第 5 章 使用 Struts 2 框架开发人事管理系统——职称类别管理	49
5.1 数据库设计	49

5.2	功能分析	50
5.2.1	模块功能	50
5.2.2	功能描述	50
5.2.3	操作序列	51
5.3	职称类别管理模块通用部分的实现	53
5.3.1	工程结构	53
5.3.2	功能实现	53
5.4	职称类别添加功能的实现	58
5.5	职称类别列表显示功能的实现	66
5.6	职称类别修改功能的实现	71
5.7	职称类别删除功能的实现	75
	小结	76
	习题	77
第 6 章	Hibernate 框架技术	78
6.1	持久化技术	78
6.2	持久层技术	79
6.2.1	持久层的概念	79
6.2.2	持久层技术的实现	79
6.3	ORM 概述	80
6.3.1	什么是 ORM	80
6.3.2	流行的 ORM 框架简介	80
6.4	Hibernate 体系结构	81
6.4.1	Hibernate 在应用程序中的位置	81
6.4.2	Hibernate 的体系结构	82
6.5	Hibernate 实体对象的生命周期	83
6.5.1	瞬态	83
6.5.2	持久态	84
6.5.3	游离态	84
6.5.4	实体对象的状态转换	85
6.6	Hibernate API 简介	85
6.6.1	Configuration 接口	85
6.6.2	SessionFactory 接口	86
6.6.3	Session 接口	86
6.6.4	Transaction 接口	87
6.6.5	Query 接口	87
6.6.6	Criteria 接口	87
	小结	87

习题	87
第 7 章 Hibernate 框架应用	88
7.1 安装 Hibernate	88
7.2 Hibernate 在 MyEclipse 中的应用	88
7.2.1 创建数据库	88
7.2.2 配置环境	90
7.2.3 配置数据库连接	94
7.2.4 开发持久化对象	95
7.2.5 编写映射文件	96
7.2.6 编写业务逻辑	97
小结	98
习题	98
第 8 章 使用 Struts+Hibernate 完成用户管理模块的开发	99
8.1 数据库设计	99
8.2 功能分析	100
8.3 配置环境	102
8.4 用户管理模块持久层设计	105
8.5 用户添加功能的实现	107
8.6 用户列表显示功能的实现	113
8.7 用户删除功能的实现	118
8.8 用户修改功能的实现	119
小结	124
习题	124
第 9 章 Spring 框架技术	125
9.1 Spring 框架简介	125
9.2 Spring 核心思想	127
9.2.1 控制反转	127
9.2.2 依赖注入	129
9.2.3 面向切面编程	131
9.3 装配 bean	133
9.3.1 bean 的基本装配	133
9.3.2 bean 的其他特性	134
小结	135
习题	135

第 10 章 Spring 框架的应用	136
10.1 Spring 的下载	136
10.2 Spring 开发环境的配置	136
10.3 Spring 控制反转应用	138
10.4 Spring 依赖注入应用	140
10.5 Spring 整合 Hibernate 的应用	143
小结	155
习题	155
第 11 章 使用 Struts 2+Hibernate+Spring 框架开发人事管理系统——	
部门管理模块	156
11.1 数据库设计	156
11.2 功能分析	157
11.2.1 模块功能	157
11.2.2 功能描述	157
11.2.3 操作序列	159
11.3 部门管理模块通用部分的实现	160
11.3.1 工程结构	160
11.3.2 在 MyEclipse 中新建 Web 工程	160
11.3.3 集成 Struts 2、Spring 和 Hibernate 框架	161
11.3.4 Hibernate 持久层设计	163
11.3.5 DAO 层设计	165
11.3.6 Service 层设计	168
11.4 查看所有部门信息模块的实现	170
11.4.1 创建查看所有部门信息的控制器	170
11.4.2 创建显示所有部门信息的页面	171
11.4.3 查看所有部门信息控制器的配置	172
11.4.4 显示所有部门信息运行结果	172
11.5 查看部门详细信息模块的实现	173
11.5.1 创建查看部门详细信息的控制器	173
11.5.2 创建显示部门详细信息的页面	174
11.5.3 显示部门详细信息控制器的配置	175
11.5.4 显示部门详细信息运行结果	175
11.6 添加部门信息模块的实现	175
11.6.1 创建添加部门信息的页面	176
11.6.2 创建添加部门信息的控制器	176
11.6.3 配置添加部门信息的控制器	178

11.6.4	添加部门信息运行结果	178
11.7	修改部门信息模块的实现	179
11.7.1	创建修改部门信息的页面	179
11.7.2	创建修改部门信息的控制器	179
11.7.3	修改部门信息控制器的配置	181
11.7.4	修改部门信息运行结果	181
11.8	部门信息删除模块的实现	181
11.8.1	创建删除部门信息的控制器	182
11.8.2	删除部门信息控制器的配置	182
11.8.3	删除部门信息的运行结果	183
小结		183
习题		183
第 12 章	人事管理系统中证件信息管理模块的开发	185
12.1	项目简介	185
12.2	证件信息管理模块分析和设计	185
12.2.1	证件信息管理模块的需求	185
12.2.2	证件信息管理模块的概要设计	186
12.2.3	证件信息管理模块的详细设计	187
12.3	Web 应用体系结构	194
12.3.1	表示层	194
12.3.2	持久层	194
12.3.3	业务层	194
12.3.4	域模型层	195
12.4	开发人事管理系统中的证件信息管理模块	195
12.4.1	域模型层的配置	195
12.4.2	持久层的配置	197
12.4.3	业务层的开发和配置	198
12.4.4	表示层的实现	200
小结		202
习题		202
参考文献		204

Web应用程序开发基础

本章将对 Web 开发的基础理论、MVC 模式的设计思想和本书所采用的项目进行介绍。通过本章的学习,可以达到以下目标:

- 了解 Web 的发展过程;
- 了解 Web 开发的主要技术;
- 理解 MVC 设计思想。

1.1 Web 开发概述

1.1.1 Web 技术的发展

随着 Internet 技术的发展,Web 技术已经被广泛地应用于 Internet 上,但早期的 Web 应用全部是静态的 HTML 页面,可以将一些文本信息呈现给浏览者,但 HTML 页面中的内容是固定不变的,因此不具备与用户交互的能力,没有动态显示功能。

很自然地,人们希望 Web 应用中应该包含一些能动态执行的页面,最早的 CGI (Common Gateway Interface,通用网关接口)技术满足了该要求,CGI 技术使得 Web 应用可以与客户端浏览器交互,不再需要使用静态的 HTML 页面。利用 CGI 技术可以从数据库中读取信息,将这些信息呈现给用户;还可以获取用户的请求参数,并将这些参数保存到数据库中。

CGI 技术开启了动态 Web 应用的时代,给了这种技术无限的可能性。但 CGI 技术存在很多缺点,其中最大的缺点就是开发动态 Web 应用难度非常大,而且在性能等各方面也存在限制。至 1997 年,随着 Java 语言被广泛使用,Servlet 技术迅速成为动态 Web 应用的主要开发技术。相比传统的 CGI 应用而言,Servlet 具有很大的优势。

Servlet 在 Web 应用中被映射成一个 URL (Uniform Resource Locator,统一资源定位符),该 URL 可以被客户端浏览器请求,当用户向指定 URL 对应的 Servlet 发送请求时,该请求将被 Web 服务器接收到,由 Web 服务器负责处理多线程、网络通信等工作,而 Servlet 的内容则决定了服务器对客户端的响应内容。

到了 1998 年,微软发布了 ASP 2.0。它是 Windows NT 4 Option Pack 的一部分,作

为 IIS 4.0 的外接式附件。它与 ASP 1.0 的主要区别在于它的外部组件是可以初始化的,这样,ASP 程序内部的所有组件都有了独立的内存空间,并且可以进行事务处理,这标志着 ASP 技术开始真正作为动态 Web 编程技术。

当 ASP 技术在世界上广泛流行时,人们很快感受到这种简单技术的魅力:ASP 使用 VBScript 作为脚本语言,它的语法简单、开发效率非常高。而且,世界上已经有了非常多的 Visual Basic 程序员,这些 Visual Basic 程序员可以很轻易地过渡成为 ASP 程序员。因此,ASP 技术迅速成为应用最广泛的动态 Web 开发技术。

随后,由 Sun 带领的 Java 阵营立即发布了 JSP 标准,从某种程度上来看,JSP 是 Java 阵营为了对抗 ASP 推出的一种动态 Web 编程技术。

ASP 和 JSP 的名称很相似,但它们的运行机制存在一些差别,这主要是因为 VBScript 是一种脚本语言,无须编译,而 JSP 使用 Java 作为脚本语句,但 Java 从来就不是解释型的脚本语言,因此 JSP 页面并不能立即执行。因此,JSP 必须编译成 Servlet,也就是说,JSP 的实质还是 Servlet。不过,编写 JSP 比编写 Servlet 要简单得多。

随着实际 Web 应用的使用越来越广泛,Web 应用的规模也越来越大,开发人员发现动态 Web 应用的维护成本也越来越大,即使只需修改该页面的一个简单按钮文本,或者一段静态的文本内容,也不得不打开混杂的动态脚本页面源文件进行修改,这是一种很大的风险,完全有可能引入新的错误。

这时候,人们意识到,使用单纯的 ASP 或者 JSP 页面充当过多角色是相当失败的选择,这对于后期的维护相当不利。慢慢地开发人员开始在 Web 开发中使用 MVC 模式。

随后 Java 阵营发布了一套完整的企业开发规范:J2EE(现在已经更名为 Java EE),紧跟着微软也发布了 ASP.NET 技术,它们都采用一种优秀的分层思想,力图解决 Web 应用维护困难的问题。

1.1.2 静态 Web 和动态 Web 的区别与联系

静态网站是最初的建站方式,浏览者所看到的每个页面是建站者上传到服务器上的一个 .htm 或 .html 文件,对于这种网站每增加、删除、修改一个页面,都必须重新对服务器上的文件进行一次下载和上传操作。网页内容一经发布到网站服务器上,无论是否有用户访问,每个静态网页的内容都保存在网站服务器上,也就是说,静态网页是实实在在保存在服务器上的文件,每个网页都是一个独立的文件。静态网页的内容相对稳定,因此容易被搜索引擎检索到,但存在着以下两个缺点。

(1) 静态网页没有数据库的支持,在网站制作和维护方面工作量较大,因此当网站信息量很大时,完全依靠静态网页制作起来比较困难。

(2) 静态网页的交互性较差,在功能方面有较大的限制。

随着人们对动态效果的需求不断增加,出现了动态网站。这里所说的动态网页并不是指网页上简单的 GIF 动态图片或是 Flash 动画,与网页上的各种动画、滚动字幕等视觉上的“动态效果”没有直接关系。动态网页可以是纯文字内容的,也可以是包含各种动画的内容,这些只是网页具体内容的表现形式,无论网页是否具有动态效果,采用动态网站技术生成的网页都称为动态网页,都具备以下 3 个基本特征。

(1) 交互性：网页会根据用户的要求和选择而动态地改变和响应，浏览器作为客户端，成为一个动态交流的桥梁，动态网页的交互性也是 Web 今后发展的方向。

(2) 自动更新：即无须手动更新 HTML 文档，便会自动生成新页面，可以大大节省工作量。

(3) 因时因人而变：即当不同时间、不同用户访问同一网址时会出现不同页面。

动态网站在页面里嵌套了程序，将一些框架相同、更新较快的信息页面进行了内容与形式的分离，将信息内容以记录的形式存入网站的数据库中，以便网站各处的调用。这样用户看到的页面可能和服务器的 HTML 文件就不一一对应了，网页框架里调用了很多数据库记录中的内容。此外动态网页是与静态网页相对应的，也就是说，网页文件的扩展名不是 .htm、.html、.shtml、.xml 等静态网页的常见形式，而是以 .asp、.jsp、.php、.perl、.cgi 等为扩展名。

从网站浏览者的角度来看，无论是动态网页还是静态网页，都可以展示基本的文字和图片信息，但从网站开发、管理、维护的角度来看有很大的差别。静态 Web 无法进行数据库操作，而动态 Web 是可以进行数据库操作的。现在几乎所有的数据都是通过数据库来保存的，也正是由于这个原因，动态 Web 开发已经被广泛应用到各个行业之中。

1.1.3 Web 应用系统的开发模式

Web 应用系统有两种模式：C/S 模式和 B/S 模式。

(1) C/S(Client/Server, 客户端/服务器端)模式：这种模式功能强大，一般对环境要求比较高，实时交互性好，对于开发而言比较复杂，维护比较麻烦，需要分别安装客户端和服务端。例如，人们日常生活中使用的 QQ 或 MSN 等，都属于 C/S 模式。

(2) B/S(Browser/Server, 浏览器/服务器)模式：相当于在 C/S 模式中以浏览器作为客户端的情况。只是在一般情况下操作系统已经集成了这个客户端，因而不用再安装了，B/S 模式结构比较简单，维护比较方便，只需管理和维护服务器端即可。例如，购物网站或论坛都采用 B/S 模式。

1.1.4 Model 1 和 Model 2

对于动态 Web 编程技术而言，则经历了所谓的 Model 1 和 Model 2 时代。

所谓 Model 1 就是 JSP 大行其道的时代。在 Model 1 模式下，整个 Web 应用几乎全部由 JSP 页面组成，由 JSP 页面接收并处理客户端请求，之后直接做出响应。用少量的 JavaBean 来处理数据库连接、数据库访问等操作。图 1-1 显示了 Model 1 的程序流程。

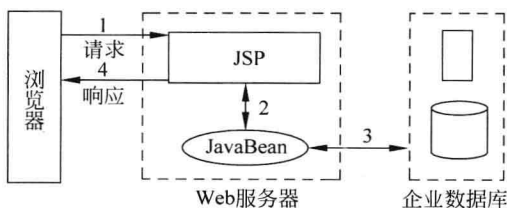


图 1-1 Model 1 的程序流程

Model 1 模式的实现比较简单,适用于快速开发小规模项目。但从工程化的角度来看,它的局限性非常明显:JSP 页面身兼 View 和 Controller 两种角色,将控制逻辑和表现逻辑混杂在一起,从而导致代码的重用性非常低,减小了应用的扩展性,增加了维护的难度。

早期有采用大量 JSP 技术开发的 Web 应用,这些 Web 应用都采用了 Model 1 架构。

Model 2 已经是基于 MVC 架构的设计模式。首先,在 Model 2 架构中,Servlet 作为前端控制器,负责接收客户端发送的请求,在 Servlet 中只包含控制逻辑和简单的前端处理程序;其次,调用后端 JavaBean 来完成实际的逻辑处理;最后,转到相应的 JSP 页面处理显示逻辑。其具体实现方式如图 1-2 所示。

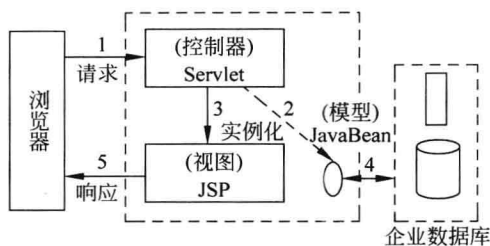


图 1-2 Model 2 的程序流程

从图 1-2 中可以看到,在 Model 2 下 JSP 不再承担控制器的责任,它只在表现层发挥作用,仅用于将结果呈现给用户,JSP 页面的请求将与 Servlet(控制器)交互,而 Servlet 负责与后台的 JavaBean 通信。在 Model 2 模式下,模型(Model)由 JavaBean 充当,视图(View)由 JSP 页面充当,而控制器(Controller)则由 Servlet 充当。

由于引入了 MVC 模式,使 Model 2 具有组件化的特点,更适用于大规模应用的开发,但也增加了应用开发的复杂程度。原本需要一个简单的 JSP 页面就能实现的应用,在 Model 2 中被分解成多个协同工作的部分,需要花更多时间才能真正掌握其设计和实现过程。

Model 2 已经是 MVC 设计思想下的架构,下面简要介绍 MVC 设计思想。

1.2 MVC 设计思想

Java EE (Java Enterprise Edition)是在 Java SE 基础上建立起来的一种标准开发架构,主要应用于企业级应用程序的开发。在 Java EE 的开发中以 B/S 模式作为主要的开发模式。在整个 Java EE 中最核心的设计模式就是 MVC(Model-View-Controller)设计模式,并且被广泛应用。M 指模型,V 指视图,C 指控制器。引入 MVC 模式的目的是实现 Web 系统的职能分工。模型用于实现系统中的业务逻辑,通常可以用 JavaBean 或 EJB 来实现。视图用于实现与用户的交互,通常用 JSP 来实现。控制器层是模型与视图之间沟通的桥梁,它可以分派用户的请求并选择恰当的视图以用于显示,同时它也可以解释用户的输入并将它们映射为模型可执行的操作。

MVC 并不是 Java 语言所特有的设计思想,也并不是 Web 应用所特有的思想,它是

所有面向对象程序设计语言都应该遵守的规范。

MVC 思想将一个应用分成 3 个基本部分：模型、视图和控制器，这 3 个部分以最小的耦合协同工作，从而能够提高应用的可扩展性及可维护性。

起初，MVC 模式是针对相同的数据需要不同显示的应用而设计的，其整体结构如图 1-3 所示。

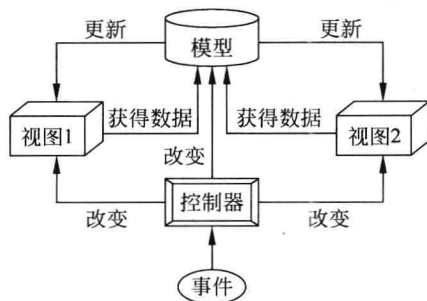


图 1-3 MVC 结构

在经典的 MVC 模式中，事件由控制器处理，控制器根据事件的类型改变模型或视图；反之亦然。具体地说，每个模型对应一系列的视图列表，这种对应关系通常采用注册的方法来完成，即把多个视图注册到同一个模型，当模型发生改变时，模型向所有注册过的视图发送通知，接下来，视图从对应的模型中获得信息，然后完成视图显示的更新。

从设计模式的角度来看，MVC 思想非常类似于一个观察者模式，但与观察者模式存在少许差别：在观察者模式下观察者和被观察者可以是两个互相对等的对象，但对于 MVC 思想而言，被观察者往往只是单纯的数据体，而观察者则是单纯的视图页面。

概括起来，MVC 有如下特点。

(1) 多个视图可以对应一个模型。按 MVC 设计模式，一个模型对应多个视图，可以减少复制及维护的代码量，一旦模型发生改变，也易于维护。

(2) 模型返回的数据与显示逻辑分离。模型数据可以应用任何的显示技术，例如，使用 JSP 页面、Velocity 模板或者直接产生 Excel 文档等。

(3) 应用被分隔为 3 层，减小了各层之间的耦合，提高了应用的可扩展性。

(4) 控制层的概念也很有效，由于它把不同的模型和不同的视图组合在一起来完成不同的请求，因此，控制层可以说是包含了用户请求权限的概念。

(5) MVC 更符合软件工程化管理的精神。不同的层各司其职，每一层的组件具有相同的特征，有利于通过工程化和工具化产生管理程序代码。

相对于早期的 MVC 思想，Web 模式下的 MVC 思想则又存在一些变化，因为对于一个应用程序而言，可以将视图注册给模型，当模型数据发生改变时，及时通知视图页面发生改变；而对于 Web 应用而言，即使将多个 JSP 页面注册给一个模型，当模型发生变化时，模型无法主动发送消息给 JSP 页面（因为 Web 应用都是基于请求/响应模式的），只有当用户请求浏览该页面时，控制器才负责调用模型数据来更新 JSP 页面。

1.3 项目概述

本书所采用的项目是某环境规划院的人事管理系统。人事管理是企业生存的主要因素,人员的增减、变动将直接影响到企业的整体运作,企业每天都要涉及人员管理问题。企业员工越多、分工越细、联系越紧密,所要做的统计工作就越多,人事管理的难度就越大。本书完成的系统为某环境规划院人事管理系统,该系统基本上能满足现代企业人事管理的需求,人事管理系统中保存了相关的人员信息,方便查询、浏览、修改等操作。

本系统的使用者为某环境规划院的管理人员。通过该系统用户能够对系统进行系统设置、员工资料、人事资料的管理,而且可以对各种资料进行统计查询。

本系统具有的功能包括系统设置、员工管理、人事管理、统计管理、工资管理、休假管理。每个功能的具体描述如下。

1. 系统设置

(1) 部门管理:对部门信息进行设置。

(2) 职位管理:对职位信息进行设置,包括的职位有办公室主任、院长、总工、副院长等。

(3) 职称管理:对职称信息进行设置,包括的职称有工程师、教授等。

(4) 奖惩信息设置:对奖惩信息进行设置,如迟到、早退、旷工等。

(5) 用户管理:对使用该系统的人员信息进行设置。

2. 员工管理

员工管理主要是对院内员工的基本信息进行管理,主要包括以下几部分的功能。

(1) 员工基本信息录入:录入员工的基本信息。

(2) 员工信息编辑:编辑修改员工信息。

(3) 员工信息删除:删除不在本单位工作的员工信息(为了保证信息的完整性,此处的删除功能为逻辑删除)。

(4) 模糊查询员工信息:根据输入的模糊查询条件,查询出所需要的员工信息。

(5) 员工信息打印:打印员工的基本信息。

(6) 员工相关资料管理:对员工的业绩考核资料及其他资料进行管理及维护操作。

(7) 员工基本信息的导入与导出:用于导入和导出员工信息,可以直接进行打印或保存为 Excel 文件。

(8) 员工状态操作:如实习、转正、退休等。

(9) 员工附加信息、简历等管理。

(10) 组织撤销:相应组织被撤销后仍保留历史薪资统计数据。

(11) 岗位调动时自动更新上下级关系(可使用拖拉操作)。

(12) 可快速查询各组织内职位和人员配备情况。

(13) 员工证件管理。

(14) 合约到期和试用期到期提醒功能。

(15) 可预先设定员工离职日期,系统自动处理离职信息。