

MVC 架构在 ASP 页面中的实现

沈才梁 (浙江工业职业技术学院 浙江 绍兴 312000)

摘要: MVC 技术是为合理解决代码和视图设计而产生的一种理论。目前,在 JAVA 开发环境中广泛采用 Struts 架构实现 MVC 技术。对于开发存在大量用户界面,并且业务逻辑复杂的大型应用程序,MVC 将会使软件在健壮性、代码重用和结构方面上一个新的台阶。尽管在最初构建 MVC 框架时会花费一定的工作量,但从长远角度看,它会大大提高后期软件开发的效率。本文介绍一种在 ASP 页面中实现 MVC 技术的途径。

关键词: ASP MVC 模型 视图 控制器

1 引言

MVC 是 Model View Controller 的缩写。MVC 通过合理的设计将软件的数据部分 (Model) 和界面部分 (View) 拆开,通过彼此间的定义接口相互通讯。这样一来,美工和代码设计人员就可以相对独立地完成各自的工作。许多软件中的“皮肤”可以随意变化而无需更改代码,使用的就是 MVC 技术。

2 MVC 架构

MVC 最初是在 Smalltalk-80 中被用来构建用户界面。M 代表模型 Model, V 代表视图 View, C 代表控制器 Controller。MVC 结构是为那些需要为同样的数据提供多个视图的应用程序而设计的,它很好的实现了数据层与表示层的分离。

单用户应用通常是以事件驱动的用户界面为组织结构的。开发人员用一个界面工具画了一个用户接口界面,然后编写代码根据用户输入去执行相应的动作,许多交互式的开发环境鼓励这么做,因为它强调先有界面然后再有功能。一些软件设计模式策略是这样的,然后经常将固定后的代码融入最后的系统当中。导致的结果就是,程序组织围绕用户界面元素和用户在那些界面元素上的动作,数据的存储,应用的功能以及用来显示的代码都杂乱无章的缠绕在一起。在单用户的系统里代码结构是可以这样的,因为系统需求不会频繁变化。但是对一个大的系统如大型 Web 系统,或电子商务系统来说就不太适用了。

通过把数据模式从各种可以被存取和控制的数据

中分离出来可以改善分布式系统的设计。MVC 设计模式由三部分组成,如图 1 所示。模型是应用对象,视图表示它在屏幕上的显示,控制器定义用户界面对用户输入的响应方式,负责把用户的动作转成针对 Model 的操作。首先控制器接收用户的请求,并决定应该调用哪个模型来进行处理。然后模型根据用户请求进行相应的业务逻辑处理,并返回数据。最后控制器调用相应的视图格式化模型返回的数据,并通过视图呈现给用户。

使用 MVC 架构最重要的一点是多个视图能共享一个模型,而同一个模型可以被不同的视图重用,大大提高了代码的可重用性。其次,由于 MVC 的三个模块相互独立,改变其中一个不会影响到其他两个,所以依据这种设计思想能构造良好的松耦合构件。最后,控制器提高了应用程序的灵活性和可配置性。控制器可以用来联接不同的模型和视图去完成用户的需求,这样控制器可以为构造应用程序提供强有力的手段。

3 在 ASP 页面中 MVC 技术的实现

MVC 的目的在于将用户视图和数据分离。在一个 ASP 页面中,视图可以由 HTML 实现。代码 1 中利用了“[#T1]”、“[#T2]”和“[#AC]”字符串分别代表页面的标题、内容和动作,将这些字符串嵌入在 HTML 页面中,在浏览器中显示其框架结构,如图 2 所示。

代码 1 利用 HTML 产生实现的视图

```
<html><head>
<title>[#T1]</title>
</head>
```

```
< form name = form1 method = " post" action = " [#  
AC]" >  
< table width = " 400" heigh = " 300" border = " 1" a-  
lian = " center" >
```

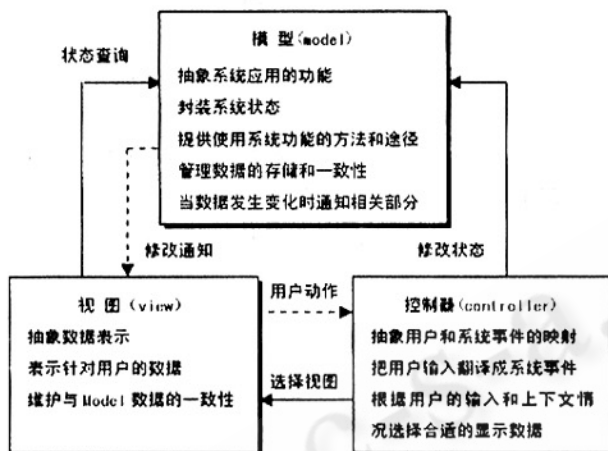


图 1 MVC 架构

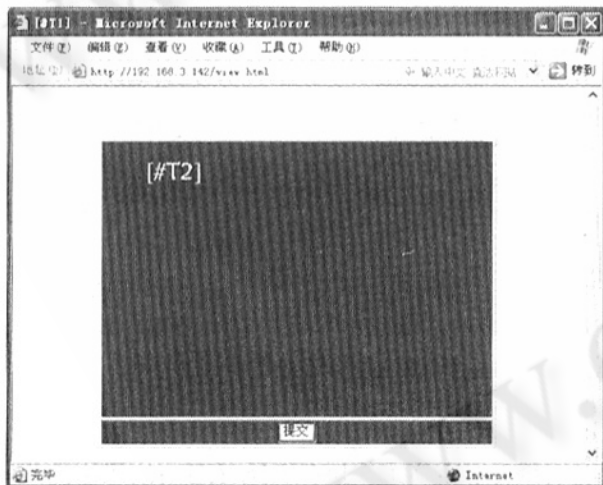


图 2 HTML 生成的视图样式

```
<tr>  
    <td width = " 341" height = " 40" bgcolor = " #  
666666" /td>  
    <div align = " center" style = " font - size : 12px ;  
color : #FFFFFF ; width : 387 ; height : 260" ;>  
        <p align = " left" >&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp;
```

```
&nbsp;   <font size = "5" face = "华文新魏">  
[ #T2] </font> </div> </tr>  
<br>  
<tr>  
<td bgcolor = "#666666">  
<div align = "center">  
<input type = "submit" value = "提交"> </div>  
> </td>  
</tr>  
</table>  
</form> </html>
```

利用 VBScript 脚本可以用于实现模型。代码 2 中定义了若干函数,这些函数实现的是数据的存储,如 Get_titlename 和 Get_text 函数;一些函数实现了执行的动作,如 Get_Action 函数;此外,OpenFile 函数实现了服务器文件的打开。

代码 2 利用 VBScript 脚本实现的模型

 $\leq \%$

Function Get_titlename()

Get titlename = " ASP 页面中 MVC 技术的实现"

End Function

Function Get_text()

Get_text = MVC 技术是为合理解决代码和试图设计而产生的一种理论。目前,在 JAVA 开发环境中广泛采用 Struts 架构实现 MVC 技术。本文介绍一种在 ASP 页面中实现 MVC 技术的途径。”

End Function

Function Get_Action ()

```
Get action = "http://www.zijpc.com/index.asp"
```

End Function

Function OpenFile(ByRef pr_Buffer, ByVal pv_FilePath)

```
Dim objFile, InStream, sFilename
```

```
Set objFile = Server.CreateObject( " Scripting. FileSystem-  
mObject" )
```

```
sFilename = pv_FilePath
```

```
If objFile.Exists( sFilename ) Then
```

```
Set InStream = objFile.OpenTextFile ( sFilename, 1,
```

False)

```
pr Buffer = InStream.ReadAll
```

Set InStream = Nothing

End If

```
Set objFile = Nothing
End Function
% >
```

模型和视图建立后,同样利用 VBScript 脚本实现控制器。首先调用 OpenFile 函数将视图框架文件打开,然后利用 replace 函数分别从模型中将相应的数据读到 bufhtml 变量中,最后利用 response 输出到浏览器,显示效果如图 3 所示。

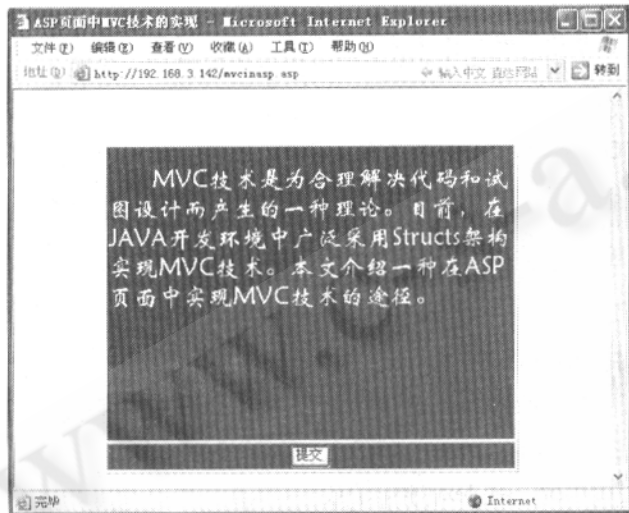


图 3 加载数据后的页面

代码 3 利用 VBScript 脚本实现控制器

```
<!--#include file = "model.asp" -->
<%
response.buffer = true
dim bufhtml Call OpenFile ( bufhtml, server.mappath
("view.html" ) )
bufhtml = replace ( bufhtml, "[ #T1]", get_titleName
())
bufhtml = replace ( bufhtml, "[ #T2]", get_text ( ) )
bufhtml = replace ( bufhtml, "[ #AC]", get_Action
())
response.write ( bufhtml)
% >
```

4 结语

利用 ASP 实现 MVC 的目的是将数据和视图分离。

只要能够做到这一点,应该说就是合理地实现了 MVC 架构。

参考文献

- 1 孙卫琴,基于 MVC 的 Java Web 设计与开发,电子工业出版社,2003。
- 2 李晓黎,ASP + SQL Server 网络应用系统开发与实例,2004。
- 3 Greg Buczek,即时应用 ASP 脚本,人民邮电出版社,2001。
- 4 卢云鹏,JavaScript 深度剖析,北京大学出版社,2004。