

基于 ASP.NET+SQL Server 的多层博客系统设计与实现*

Study and Implementation of Multi-layer BLOG System Based on Technology of ASP.NET and SQL Server

曹红苹¹, 刘建华²

CAO Hong-ping¹, LIU Jian-hua²

(1. 上海工程技术大学管理学院, 上海 201620; 2. 桂林航天工业高等专科学校计算机系, 广西桂林 541004)

(1. School of Management Studies, Shanghai University of Engineering Science, Shanghai 201620, China; 2. Department of Computer Technology, Guilin College of Aerospace Technology, Guilin, Guangxi, 541004, China)

摘要: 阐述一个基于 ASP.NET+SQL Server 的多层 Blog 系统的设计思想和整个系统关键部分的实现过程。多层 Blog 系统分为公共实体层, 业务逻辑层, 数据逻辑层, 数据操作层和用户界面 UI 层。多层 Blog 系统的结构清晰, 能有效降低耦合, 便于功能扩充, 有利于程序调试和维护。

关键词: 博客 多层架构 SQL Server ASP.NET

中图分类号: TP319 **文献标识码:** A **文章编号:** 1002-7378(2010)04-0490-03

Abstract: A multi-layer Blog System based on the technology of ASP.Net and SQL Server was proposed. The design idea and working theory were introduced and the realization of the system's key part is also expounded. The advantages of multi-layer architecture are clear and effective reduction of the coupling, helpful for program development, testing and maintenance.

Key words: blog, multi-layer architecture, SQL Server, ASP.NET

博客(Blog)是以网络作为载体, 简易迅速便捷地发布自己的心得, 及时有效轻松地与他人进行交流, 再集丰富多彩的个性化展示于一体的综合性平台。Blog 已成为家庭、公司、部门和团队之间越来越盛行的沟通工具, 因此它也逐渐被应用在企业内部网络(Intranet)。应用系统的开发, 经历了从最早的单机模式到多终端模式, 再到后来的客户服务模式, 发展到三层架构模式与多层模式^[1]。本文应用 ASP.NET, SQL Server 2005 设计一个多层 BLOG

系统。

1 基于 ASP.NET +SQL Server 的多层软件体系架构

基于 ASP.NET +SQL Server 的多层软件体系架构是针对 Web 应用的特点, 将 Web 应用系统分为: 用户界面层、业务逻辑层、数据逻辑层、数据操作层、公共组件实体层及公共操作层(图 1)^[2]。(1) 用户界面层为客户端提供对应用程序的访问。由 ASP.NET Web 窗体和代码隐藏文件组成, Web 窗体只是提供用户操作, 而代码隐藏文件实现各种控件的事件处理。用户界面层接受用户的输入, 然后提供给业务服务层, 在业务服务层中用户的输入得到验证、处理或操作之后, 用户界面层再把业务服务层所得到的结果给客户显示出来以响应用户。简而

收稿日期: 2010-08-09

修回日期: 2010-10-11

作者简介: 曹红苹(1968-), 女, 副教授, 主要从事软件工程和企信息化方面的研究。

* 上海市教委第四期教育高地信息管理与信息系统建设项目(编号: B-8200-09-06)资助。

言之,这一层是系统的控件库(自定义的控件逻辑都包含在内)。(2)业务逻辑层包括所有的业务规则、数据验证、数据操作、数据处理和应用程序的安全性,实现了企业业务处理的各种逻辑。业务逻辑层存在的意义是使软件能够获得不断增加的可维护性和可重用性。(3)数据逻辑层是通过与数据操作层进行交互来提取、插入、更新和删除数据信息,并不管理和存储数据,它只是在业务服务层和数据操作层之间的一个接口。另外,数据逻辑层的存在还有另一个意义。数据访问技术有很多种,例如:ODBC,DAO,OLEDB,ADO,ADO.NET等。而对于不同的技术,对数据的访问方式是不同的,如果把数据访问放到一个单独的、特定的层里,对于上述情况的适应性将会增大很多。(4)数据操作层在数据库中,主要采用存储过程来进行数据库的操作,这样使得数据逻辑层访问数据库时不是直接跟数据表交互,而是通过执行存储过程完成数据操作层,使数据操作速度加快,数据更加安全。(5)公共组件实体层是这六层架构设计的关键,也是将系统高度抽象的结果。系统中的全部实体均来自该层,其它各层都要依赖这一层。其优点是使系统各模块内容明确,其它层对各实体的操作,只要先继承这一层相应的实体就可以了。再有,在以后的系统维护中,如果删除某些实体,只要在这一层中删除,再在其它各层删去其调用就可以了,这在很大程度上增加了系统的可维护性^[3]。(6)公共操作层与业务无关,但是会被经常使用的功能抽取出来,作为独立的基础服务模块。这些模块具有较强的可重用性。

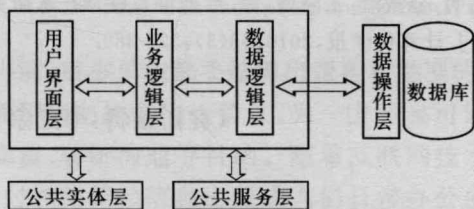


图1 Web应用系统的多层架构

采用基于 ASP.NET + SQL Server 的多层软件体系架构来设计和开发 Web 应用程序,主要有以下优点:(1)每个层次完成的功能比较单一,与实际问题有明显的对应关系。(2)各层的内聚性好,有利于采用面向对象的开发方法,提高代码的可重用性。(3)能够使系统的架构更加清晰,结构简单,有利于系统调试和开发实现。

2 多层 Blog 系统设计与实现

2.1 系统的架构与功能设计

多层 Blog 系统包括用户实体、文章实体、评论实体、文章分类实体、激活代码实体。采用基于.NET + SQL Server 的多层软件体系架构将多层 Blog 系统的架构设计为用户界面层、公共实体层、业务逻辑层、数据逻辑层、数据操作层,具有用户管理、日志分类、日志操作、个人博客管理功能。

用户管理功能完成用户注册、登录、密码找回、资料设定和查看。日志分类功能完成添加、删除和管理博客日志分类操作。日志操作功能完成日志发表、修改、浏览和评论。个人博客管理功能完成个人博客首页日志列表、个人博客分类列表、首页最新日志列表、首页最新加入成员列表,和首页热点博客列表。

2.2 系统的关键部分实现

在多层 Blog 系统的项目结构(图2)中,WebUI 是用户界面层、Entity 是业务实体层、Business 是业务逻辑层、DataAccess 是数据逻辑层(用于进行与数据操作层的交互)、Common 是公共操作层。用户界面 WebUI 层提供与用户交互的界面,对用户的指令进行翻译,然后根据用户指令调用业务逻辑层相应接口,并把数据传递给业务逻辑层,如果用户修改完文章,点击提交按钮,则调用业务层的 ModifyArticle()。业务实体层用于定义系统中所用到的实体,包括用户实体,文章实体,评论实体,文章分类实体,激活代码实体。业务逻辑层将 WebUI 层传递来的数据进行一定的处理,然后调用数据访问层的函数来获取结果。比如注册一个用户业务层逻辑的实现为:

```
public bool Register(ref UserInfoEntity ue)
{
    ue.Password = Common.md5(ue.Password);
    return new DataAccess.UserInfoAccess().
    Register(ref ue);
}
```

数据操作层的主要功能是利用实体变量将数据传给存储过程,并执行相应的存储过程^[4]。本系统的数据库中包含多个数据表以及多个存储过程。数据表包括用户表、文章表、评论表、文章分类表、激活代码表等,存储过程用来访问数据库,对其中的数据进行相应的操作,这样使得应用程序访问数据库时不是直接跟数据表交互,而是通过执行存储过程完

成,极大地提高了各种操作的运行效率和速度,同时也很好地屏蔽了数据表的逻辑结构和数据表之间的关系。公共操作层与业务无关,但是被经常使用的功能抽取出来,作为独立的基础服务模块。这些模块具有较强的可重用性。包括一些文件操作、邮件发送等相关的函数。



图2 多层 Blog 系统的项目结构

3 结束语

基于 ASP.NET + SQL Server 的多层 Blog 系统采用多层体系架构,将系统分为公共实体层,业务逻辑层,数据逻辑层,数据操作层和用户界面 UI

层。多层 Blog 系统结构清晰、各层功能明确、易于扩展,如果增加业务功能,只需在业务服务层增加功能模块即可,能有效降低耦合,有利于程序调试和开发。多层 Blog 系统的应用服务器层根据功能分成若干个层,并利用单独的项目进行管理,极大地提高了系统的稳定性、可扩展性和移植性。

参考文献:

- [1] Microsoft Corporation. Designing distributed applications[DB]. MSDN Library For Visual Studio, NET, 2005: 212-214.
- [2] 梅宏, 申峻嵘. 软件体系结构研究进展[J]. 软件学报, 2006, 17(6): 1257-1275.
- [3] 刘军, 阳小华, 杨星. 教学信息发布与管理系统的设计与实现: 基于 .NET 组件技术[J]. 计算机工程与应用, 2006(2): 99-02.
- [4] 郭磐君, 张艳丽, 王芳芳. ASP. NET AJAX 入门与案例精解[M]. 北京: 机械工业出版社, 2007.

(责任编辑: 邓大玉)

(上接第 485 页)

参考文献:

- [1] Wang R Y, Trong D M S. Beyond accuracy: what data quality means to data consumers[J]. Journal of Management Information Systems, 1996, 12(4): 5-33.
- [2] Pipino L, Lee Y, Wang R Y. Data quality assessment [J]. Communications of the ACM, 2002, 45(5): 211-218.
- [3] 韩京宇, 徐立臻, 董逸生. 数据质量研究综述[J]. 计算机科学, 2008, 135 (12): 1-5, 12.
- [4] Batini C, Cappiello C, Francalanci C, et al. Methodologies for data quality assessment and improvement[J]. ACM Computing Surveys, 2009, 41(3): 1-52.
- [5] 杨青云, 赵培英, 杨冬青, 等. 数据质量评估方法研究 [J]. 计算机工程与应用, 2004, 9: 3-4, 15.
- [6] 高明, 金澈清, 王晓玲, 等. 数据世系管理技术研究综述 [J]. 计算机学报, 2010, 33(3): 373-389.

(责任编辑: 韦廷宗)