

# 三层架构在学位论文管理系统中的应用研究\*

岳付强

(西昌学院,四川 西昌 615013)

**【摘要】**在现代应用系统开发中,越来越多地使用多层架构。通过分层设计可以分散关注、松散耦合、逻辑复用以及标准定义等。本文在介绍ASP.NET中的三层架构技术的基础上,详细阐述了三层架构在学位论文管理系统实现中的应用。

**【关键词】**三层架构;学位论文;应用研究;工厂模式

**【中图分类号】**TP315 **【文献标识码】**A **【文章编号】**1673-1891(2010)04-0072-03

## 1 引言

西昌学院每年都要管理众多毕业生学位论文相关数据。目前仍然采用手工方式对这些数据进行管理,效率极低且极大地影响了学校学位科、二级学院、教师和学生之间的信息传递、交流与沟通。针对西昌学院学位论文管理的特点和需求,运用ASP.NET 4.0网络编程技术和SQL Server 2008 R2数据库管理技术,开发了一套基于B/S(Browser/Server)模式的学位论文管理系统<sup>[1]</sup>。本文将在介绍ASP.NET中的三层架构技术的基础上,详细阐述三层架构在学位论文管理系统实现中的应用。

## 2 三层架构技术

分层架构将整个系统分为不同的逻辑块,大大降低了系统的开发和维护成本;将数据访问和逻辑操作都集中到组件中,增强了系统的复用性;同时还大大增强了系统的扩展性。故本系统设计采用了ASP.NET中的三层架构技术,包括Web表示层、业务逻辑层以及数据访问层,其体系结构如图1所示<sup>[2]</sup>。



图1 ASP.NET中的三层架构

### (1) Web表示层

Web表示层用于显示数据和接收用户输入的数据,为用户提供一种交互式的操作界面,主要由ASP.NET Web窗体文件和隐藏代码文件组成。其中Web窗体文件由HTML构成,主要提供给用户操作界面;而隐藏代码文件则实现各控件的事件处理。

### (2) 业务逻辑层

业务逻辑层(BLL)是Web表示层和数据访问层之间通信的桥梁,主要负责数据的传递和处理,例如数据有效性的检验、业务逻辑描述等,主要表现为众多的类文件,为Web表示层提供调用接口,同时又调用数据访问层。

### (3) 数据访问层

数据访问层(DAL)用于实现与数据库的交互和访问,从数据库获取数据或者保存数据到数据库。其中的数据库访问类(DBUtility)是对ADO.NET的封装,封装了一些常用的重复的数据库操作<sup>[3]</sup>。

## 3 简单三层架构的实现

为了实现系统三层架构,将每层作为一个独立的项目进行设计,这样可以更好地提高内聚性,降低模块之间的耦合性,也便于以后系统的维护和扩展。在Visual Studio中,新建“ArticleManageSystem”解决方案,然后在其中添加5个相关项目,如表1所示<sup>[4]</sup>。

表1 解决方案需要添加的项目

序号	项目名称	项目类型	描述
1	Model	类库项目	数据实体
2	DBUtility	类库项目	数据库操作类库
3	DAL	类库项目	数据访问层
4	BLL	类库项目	业务逻辑层
5	Web	Web程序	表示层

添加完成后的简单三层架构实际解决方案项目,如图2所示。



图2 简单三层架构解决方案

收稿日期:2010-10-11

\*基金项目:四川省教育厅项目(NO.P09374);西昌学院科研项目(NO.ZZSSA0718)。

作者简介:岳付强(1979- ),男,重庆忠县人,讲师,硕士,研究方向:数据库与数据挖掘、软件过程技术与方法以及信息系统工程与技术。

### 3.1 业务实体 Model 的实现

Model 封装的实体类,一般用于映射数据库的数据表或视图,用来描述业务中客观存在的对象。Model 实体类将贯穿 Web 层、BLL 层和 DAL 层,并在三层之间进行调用和传递。在 Model 项目中将添加以下业务对象的实体类:

(1) 学生信息类 StudentInfo。该类是对学生实体的相关数据字段的描述,实例化此类将得到与之对应的学生的具体信息,如学号、姓名、性别、密码、院系、专业、邮箱、电话等。

(2) 教师信息类 TeacherInfo。该类是对教师实体的相关数据字段的描述,实例化此类将得到与之对应的某个教师的具体信息,为实现教师的相关功能服务。该类包含的相关字段主要有教师登录名、姓名、性别、密码、院系、职称、学历、联系电话、邮箱、个人简历等。

(3) 题目信息类 SubjectInfo。该类是对论文题目实体的相关数据字段的描述,实例化此类将得到与之对应的某个题目的具体信息,为实现教师出题和学生选题的相关功能服务。该类作为系统的核心实体类之一,主要包括编号、标题、简介、教师、院系、路径、状态、限选人数、当前选择人数等。

(4) 论文信息类 ArticleInfo。该类是对论文实体的相关数据字段的描述,实例化此类将得到与之对应的某篇论文的具体信息,为实现论文的上传、审核及查询相关功能服务。该类作为系统的又一个核心实体类,主要包含论文编号、学生编号、题目编号、中文摘要、英文摘要、中文关键字、英文关键字、指导教师、系别、专业、路径、状态、上传日期等。

此外,还有院系信息类 DepartmentInfo 和专业信息类 MajorInfo,它们是为系统的其它实体类提供基础数据;留言信息类 MessageInfo,该类是对留言实体的相关数据字段的描述,为系统在线留言功能模块提供服务,主要包含编号、标题、内容、作者、日期、回复等。

### 3.2 数据库操作类库的实现

DBUtility 类库是 DAL 层的一个辅助类库,其直接执行 SQL 语句与底层数据库交换数据,同时为上层的 DAL 层提供相关的服务。在 DBUtility 项目中主要有一个 SQLDBHelper 类,在这个类中实现了以下一些方法:

(1) 执行简单 SQL 语句,返回受影响的行数: `int ExecuteSql(string sql)`。该方法提供了执行简单 SQL 语句并返回结果行数的功能。例如在业务逻辑层需要统计某个论文题目的选择人数,就可以通过

在数据访问层调用此方法来实现。

(2) 执行简单 SQL 语句,返回 SqlDataReader: `SqlDataReader GetDataReader(string sql)`。该方法提供了执行简单 SQL 语句并返回一个 SqlDataReader 的功能。例如在业务逻辑层中验证用户登录密码是否正确这样轻量级的数据库访问就可以通过在数据访问层调用此方法来实现。

(3) 执行简单 SQL 语句,返回 DataSet: `DataSet GetDataSet(string sql)`。该方法提供了执行简单 SQL 语句并返回 DataSet 的功能。例如系统后台管理员需要查看当前有哪些同学上传了论文,即可调用此方法连接 Article 和 Student 表来返回数据集。

(4) 执行简单 SQL 语句,返回 DataTable: `DataTable GetDataTable(string sql)`。此方法与 GetDataSet() 类似,但是只能从一个数据表中获得数据。

此外, `int ExecuteSql ( string sql , params SqlParameter [ ] cmdParms )`、`SqlDataReader GetDataReader ( string sql , params SqlParameter [ ] cmdParms )`、`DataSet GetDataSet ( string sql , params SqlParameter [ ] cmdParms )`、`DataTable GetDataTable ( string sql , params SqlParameter [ ] cmdParms )` 和 `void PrepareCommand ( SqlCommand cmd , SqlConnection conn , SqlTransaction trans , string cmdText , SqlParameter [ ] cmdParms )` 等方法分别对应前面执行简单 SQL 语句的方法,不同的是这些方法可以执行带参数的 SQL 语句。

### 3.3 数据访问层的实现

DAL 层通过 DBUtility 数据操作类库来直接执行 SQL 命令,需要添加对 DBUtility 的项目引用。同时在 DAL 层中也会用到 Model 层的实体类来传递数据,故也需要添加对 Model 项目的引用。在 DAL 项目中添加一个名为 Article 的类,主要包含以下方法:

(1) 是否存在某条记录 `Exists(int Id)`: 该方法用于判断某条论文信息是否存在,业务逻辑层传递一个论文的 Id,数据访问层通过调用此方法来实现。

(2) 增加一条数据 `Add(Model.Article model)`: 该方法通过接收业务逻辑层传来的 Article model 实体,获得一条论文信息的所有数据,从而为业务逻辑层提供添加一条论文信息的功能。

(3) 更新一条数据 `Update(Model.Article model)`: 该方法通过接收业务逻辑层传来的 Article model 实体,获得一条论文信息的所有数据,从而为业务逻辑层提供更新一条论文信息的功能。

(4)删除一条数据 Delete(int Id):该方法通过接收业务逻辑层传来某条论文的Id编号,为业务逻辑层提供删除一条论文信息的功能。

(5)获得数据列表 GetList(string strWhere):业务逻辑层通过调用此方法来获得满足一定条件的所有论文信息列表,其中strWhere为设定的查询条件。

(6)分页获取数据列表 GetList(int PageSize, string strWhere):业务逻辑层调用此方法可以分页获得满足一定条件的所有论文信息列表,除了strWhere为设定的查询条件外,还必须提供PageSize即分页大小参数值。

### 3.4 业务逻辑层的实现

业务逻辑层主要负责系统业务的处理,逻辑数据的生成、处理及转换,判断输入的逻辑性数据的正确性及有效性。在BLL层中会用到Model项目的实体类以及调用DAL层来访问数据库,所以需要添加对这两个项目的引用。例如在该层中添加一个Article类,其中就包含某条论文是否存在 ArticleExists(int Id)、增加一条论文数据信息 ArticleAdd(Model.Article model)、更新一条论文数据信息 ArticleUpdate(Model.Article model)、删除一条论文数据信息 ArticleDelete(int Id)以及获得论文数据列表 ArticleGetList(string strWhere)等方法。

从上面这个类在业务逻辑层的实现来看,好像业务逻辑层的作用并不大。但是从实际大型工程项目的实际需求来看,业务逻辑层的作用不可小视。另外,系统学生选题功能模块在业务逻辑层的实现就需要将判断是否已选题、所选题目的选题人数是否已达上限、对所选题目的选题人数加以这些操作放在一个名为SubChoose的方法中,与它在数据访问层的实现是有很大区别的。

### 3.5 表现层的实现

表现层主要用于实现与用户的交互、接收用户请求或返回用户请求的数据结果的展现,主要由ASP.NET Web窗体文件和隐藏代码文件组成。通常为了统一用户界面的样式和风格,还会用到CSS(Cascading Style Sheet)技术<sup>[9]</sup>,这样便于系统维护与升级。

在上传论文的页面中,只需要在页面收集用户输入的数据并将其赋值给Model层的ArticleInfo实体类,然后再调用业务逻辑层的Add(Model.Article model)方法即可完成论文及相关信息的提交。整个过程表现层并不需要知道信息是如何提交到系统

数据库的,它只需将收集到的数据通过ArticleInfo实体交给业务逻辑层完成即可,具体的提交过程由业务逻辑层调用数据访问层,再由数据访问层调用数据库操作类的相关方法将数据保存到数据库中,数据在各层间的传递由ArticleInfo实体完成。又如在系统管理后台需要显示所有已上传的论文信息,以供管理人员查看,只需要在论文的显示页面调用业务逻辑层的GetList(string strWhere)方法即可实现。

## 4 实现基于工厂模式的三层架构

前面实现的系统只能支持SQL Server数据库,随着客户需求和系统复杂度的增加以及应用规模的扩大等,有的客户希望系统能支持Oracle数据库,甚至可以同时支持两种及以上的数据库。为了满足这样的需求,可以在原来简单三层架构的基础上运用工厂模式来使系统满足其要求。

在“ArticleManageSystem”解决方案中,首先将原来的DAL项目命名为SQLServerDAL,同时增加一个OracleDAL项目来实现对Oracle数据库的访问;然后增加一个IDAL抽象接口类库项目用以实现数据访问层的抽象接口以及一个DALFactory类库项目来按照IDAL接口中的定义分别创建SQLDBDAL层和OracleDBDAL层。这样,不需要修改其它层次的代码即可实现系统对SQL Server和Oracle两种数据库的访问。实现后的基于工厂模式的三层架构实际解决方案项目如图3所示。



图3 基于工厂模式的三层架构

## 5 结束语

软件的分层架构体现了“分而治之”的思想,具有分散关注、松散耦合、逻辑复用以及标准定义等优点。当然,分层架构也不可避免地具有某些缺陷。例如,如果分层太多,则开发的工作量加大,管理起来也会比较麻烦;某种程度上降低了系统的性能甚至有时会导致级联的修改等。只有在了解三层架构的真正思想和意图后,才能很好地设计出符合实际业务需要的系统结构。本文所介绍的ASP.NET三层架构技术的设计思想及其实现方法,特别是基于工厂模式的三层架构技术,对众多典型的信息系统开发具有一定的参考价值 and 借鉴作用。

(下转78页)

#### 4 结束语

通过对控制系统硬件的可靠性设计以及对控

制系统的输入、输出单元的可靠性研究,能够使伺服驱动柜更加完善、更加可靠。

#### 注释及参考文献:

- [1]SIEMENS公司.SIMATIC S7-200 系统手册[S].2002.3.
- [2]常恒毅.可编程控制器[M].北京:人民邮电出版社,1991.
- [3]陈宇,段鑫.可编程控制器基础及编程技巧[M].广州:华南理工大学出版社,2002.3.
- [4]袁任光.可编程控制器应用技术及实例[M].广州:华南理工大学出版社,2003.10.

### Reliability Research of The PLC Control System and its Application in the Server Driver

JIANG Chun-lei<sup>1</sup>, HUANG Peng<sup>1</sup>, SUN Tao<sup>2</sup>, YANG Dao-yong<sup>2</sup>

(1.School of Automotive and Electronic Engineering, Xichang College, Xichang, Sichuan 615013; 2.Xichang Satellite Launch Center, Xichang, Sichuan 615000)

**Abstract:** With the more and more wide use of PLC control system, it's necessary to research its reliability. This paper gives the discussion and study of how to improve the reliability of the PLC control system in the server driver, and provides reference for how to design and study other PLC control systems.

**Key words:** Reliability; PLC(Programmable Logic Controller); Automatic control system; Driver cabinet

(上接 74 页)

#### 注释及参考文献:

- [1]秦光源,肖斌,汪敏.基于B/S架构的毕业设计过程管理系统的分析与设计[J].计算机与信息技术,2009(9):12-13.
- [2]李天平..NET深入体验与实战精要[M].北京:电子工业出版社,2009:494-500.
- [3]胡程,李文杰.ADO.NET脱机数据的开放式并发控制研究[J].计算机与现代化,2010(3):262-265.
- [4]北大青鸟.在.NET框架下开发三层结构数据库应用系统[M].北京:科学技术文献出版社,2008:14-18.
- [5]温谦,主编程.别具光芒—CSS网页布局案例剖析[M].北京:人民邮电出版社,2010.

### Research on the Applications of Three-tier Architecture in the Thesis Management System

YUE Fu-qiang

(Xichang College, Xichang, Sichuan 615013)

**Abstract:** In the development of modern application system, multi-layer structure is used more and more. Attention can be distributed, coupling can be loosen, logic can be reused and standard can be defined through the hierarchical designation. This paper expatiated the implementation of three-tier architecture in the thesis management system in detail, based on the introduction of three-tier architecture technology in ASP.NET.

**Key words:** Three-tier architecture; Thesis; Application research; Factory pattern