

注： 这是样例，仅示意说明论文的章节安排及论文规范，不是标准、优秀的。

合理安排论文章节。一般而言，第一章为引言， 最后一章为测试及分析（或类似内容，章标题可以不同）。而正文中间可根据自己论文内容分为 3-4 章，各章内容及章标题自行拟定。

# 研究生学位论文进程管理系统设计与实现

## 摘 要

随着研究生招生规模的不断扩大,研究生学位论文管理过程中出现的工作效率低,交互性差等问题。为有效对研究生学位论文进程进行质量监控和管理,本文采用软件工程分析、设计的思想开发实现了一个基于 B/S 的研究生学位论文进程管理系统。

本系统采用三层架构,在 Microsoft Visual Studio 2008 开发平台下以 C# 语言进行开发,后台数据库采用 SQL Sever2005。在技术上整合和利用了 ASP.NET、AJAX、ADO.NET 等各项技术的优势。系统开发中使用 PowerDesigner 进行数据库设计,使用存储过程操作数据库中的表,实现了在数据处理层只需调用相关存储过程。使用 CSS 技术结合母版页达到控制系统界面风格统一,采用 Ajax 技术更好的实现页面刷新,用户密码存储采用 MD5 加密处理。本系统用户分为研究生,教师,教学秘书以及超级管理员等用户组。超级管理员有初始化系统,查看高校各学院的管理情况等功能;教学秘书有管理本学院研究生和教师信息,审核选题,分配答辩安排以及上传成绩表等功能;教师有申请公选选题,确定研究生选题,检查和审核研究生文档,查看研究生进程以及邮件联系研究生等功能;研究生有选题,上传文档,查看成绩,发布主题帖等功能。系统实现了对研究生论文全过程的管理和质量监控,有力地推进了研究生培养的规范化建设,有利于研究生论文质量的提高。

**关键词:** 学位论文进程管理; 三层架构; ASP.NET; AJAX

## **Design and Implementation of Dissertation Process Management System for Graduate Student**

With increase of graduation student number in university, efficiency and interaction are serious problem in the process of graduation dissertation. For efficiency quality control and management for graduate's dissertation, a dissertation process management system for graduate student based on B/S is developed, with the help of software engineering analysis and design method.

Three-layer structure is used as system design mode, Microsoft Visual Studio 2008 is chosen as the development platform, and C# is development language. Database environment of the system is Sql Server2005. ASP.NET, AJAX, and ADO.NET are integrated for making use of technical advantages. In system development, PowerDesigner is used for database design. Stored procedure is adopted for operation of table data in database. Thus, just call the name of stored procedure is enough in data processing layers. CSS technology combined with MasterPage is used to control system in the style of unified interface. Ajax technology is a better realization choice for refurbishing page. User password storage is encrypted by MD5. Users of this system are divided into four kinds of user groups, which are graduate students, teachers, teaching secretary and super administrator. Super administrator has the role of initialization system, checking the management situation of each college in university. Teaching secretary of each college has the roles of management graduate and teachers' information, audit dissertation title, arrangement graduation thesis defence, and uploading grade. The roles of teachers are applying the public project title, inspecting and auditing graduate's title, check graduate student dissertation process and E-mail contact. The roles of graduate student are selecting title, uploading documents, quering grades, and releasing the subject placard etc. The system has realized process management and quality control of graduate's dissertation, which effectively promote the standardization of graduate education, and contribute to improve the quality of graduate thesis.

**Key words:** Dissertation Progress Management; Three Layers Structure; B/S; ASP.NET; AJAX;

# 目 录

论文总页数：39 页

1	引言.....	1
1.1	课题背景.....	1
1.2	国内外研究现状.....	1
1.2.1	系统开发模式现状.....	1
1.2.2	系统实现功能发展状况.....	2
1.2.3	系统相关技术架构现状.....	2
1.2.4	系统安全性现状.....	3
1.3	本论文的意义.....	4
1.4	本论文的技术路线.....	4
1.5	全文体系.....	4
2	系统分析.....	5
2.1	系统三层架构设计.....	5
2.2	系统需求分析.....	5
2.2.1	系统功能模块描述.....	6
2.2.2	系统顶层数据流图.....	7
2.2.3	系统性能需求.....	8
2.3	系统架构设计.....	9
2.4	本章小结.....	9
3	系统详细设计.....	9
3.1	数据库设计.....	9
3.2	模型层类图设计.....	14
3.3	系统各模块结构设计.....	14
3.4	系统各层详细设计.....	16
3.4.1	模型层(Model)设计.....	16
3.4.2	数据处理层(DAL)设计.....	17
3.4.3	逻辑处理层(BLL)设计.....	19
3.4.4	显示层 (WEB)设计.....	23
3.5	系统部分功能时序图.....	25
3.6	本章小结.....	25
4	系统功能的实现.....	26
4.1	数据库连接.....	26
4.2	主要功能实现.....	28

4.2.1	教学秘书分配答辩安排.....	28
4.2.2	教师查看研究生进度.....	29
5	系统测试.....	30
5.1	测试概述.....	31
5.2	功能测试执行情况.....	31
5.3	性能测试执行情况.....	32
5.3.1	活动用户视图.....	33
5.3.2	每分钟点击数.....	33
5.3.3	吞吐率.....	34
5.3.4	事务响应时间.....	34
5.4	性能测试分析.....	34
5.5	本章小结.....	35
结 束 语	.....	35
参考文献	.....	36
致 谢	.....	38
声 明	.....	39

# 1 引言

## 1.1 课题背景

近年来,随着我国高等教育的迅速发展,特别是研究生的扩招形势下,研究生学位论文出现了新的问题:研究生数量迅速增加与学术水平下降的矛盾<sup>[1]</sup>。学位论文质量是研究生教育质量的一条重要衡量标准,新形势下研究生教育与管理的最主要的任务就是保证和提高学位论文的质量。学位论文质量管理,应将重点放在过程管理,如论文开题报告查新制度,论文中期考核制度,论文预答辩制度等<sup>[1]</sup>。此外,在学位论文的管理过程中,也存在诸多问题与不便。所以设计并实现合理的、科学的学位论文进程管理系统已是迫在眉睫,该系统也会极大的提高学位论文过程管理效率,从而提高学位论文质量。

## 1.2 国内外研究现状

将过程管理技术引入到硕士生学位论文研究中对提高论文质量具有重要的推动和借鉴作用<sup>[2]</sup>。从全周期的角度来分,研究生学位论文研究过程可以划分为六个活动,分别是论文选题、论文开题、技术攻关、中期检查、论文撰写、论文完成<sup>[5]</sup>。这种过程阶段的划分方式与目前已有的大多数毕业设计管理系统的过程划分方式大致相同,只是增加了技术攻关活动。尽管这个过程的划分在研究生与本科生之间没有太大的区别,但是在研究生的论文研究过程中,有更为严格的要求性,并要求论文的研究结果更有科学价值。所以在针对研究生学位论文进程管理系统的设计中会有一些自己的特点,比如增加预答辩阶段、课题组的资料共享等。当然,在很大程度上还是可以借鉴普通的本科毕业设计管理系统的。下面就目前的一般毕业设计管理系统的发展现状从多方面展开分析。

### 1.2.1 系统开发模式现状

目前信息管理系统(MIS)主要采用C/S和B/S两种开发模式<sup>[1]</sup>。浏览器/服务器模式(B/S),即瘦客户机模式,由于其具有实用环境广、可扩展性好、易于维护、客户端免安装、对客户端硬件配置要求低等特点,所以目前几乎所有的毕业设计管理系统都采用的这种开发模式。

基于WEB的B/S高校毕业设计管理系统,目前的系统设计大多数采用不完全的面向对象的设计,以.NET平台下的开发为例,大多数采用UML建模,但是在实现上却是采用MVC模式的两层结构。近年来三层架构(包括简单三层架构和工厂模式)的发展,使得系统设计趋于完全的面向对象。“三层结构”指:“表现层”、“中间业务层”、“数据访问层”<sup>[21]</sup>。表现层:位于最外层(最顶层),离用户最近。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。中间业务层:负责处理用户输入的信息,或者是将这些信息发送给数据访问层进行保存,

或者是调用数据访问层中的函数再次读出这些数据。中间业务层也可以包括一些对“商业逻辑”描述代码在里面。数据访问层:仅实现对数据的保存和读取操作。数据访问,可以访问数据库系统、二进制文件、文本文档或是XML 文档。“三层结构”具有可升级性、实用性、可维护性的优点。

### 1.2.2 系统实现功能发展状况

通过对搜索的相关文献分析,目前的毕业设计管理系统大多数都根据用户的不同需求来实现各自的功能,这就是多角色系统,同时这也符合毕业设计系统需要学生、导师、教务秘书多种角色的参与。系统的发展也是功能完善的过程,从最初的简单的实现选题、开题、中期检查、答辩等阶段的作业上传下载过程,到现在从质量保障、过程管理中的角色交互、论文的在线编辑、答辩的网络可视化、毕业设计效果统计分析等方面研究完善系统。

过程管理中的角色交互,即研究生、导师、专家及系统管理员相互之间从系统初始化投入使用后到答辩评审完结这段时间内能有效及时的联系沟通。目前解决这一问题的主要方式有:E-mail、系统公告、短信平台。其中普遍采用的是由SMTP实现的邮件方式,导师可以对需要下达的任务给学生群发邮件,除此之外,已经有系统通过创建讨论组的方式实现导师与学生之间的交互。系统管理员可以通过公告告之不同权限用户组相关事项,导师也可以通过公告给所带学生发出通知等。利用SMS技术在有服务提供商SP支持的情况下,实现了紧急事项的短信平台的交互<sup>[6]</sup>。

在中期检查阶段答辩前的论文初审中,导师可以通过浏览学生上传的论文在系统内在线编辑<sup>[6]</sup>,并可保存到服务器,从而更好的提高监督检查的效果。答辩的网络可视化,即能使答辩现场和远程答辩导师实现实时的图形、图像、视频、音频、文本及现场实况等多种媒体信息交流,这种实时交互可借助视频会议系统来进行<sup>[7,8]</sup>。管理员通过对各类数据的统计分析,得出毕业设计效果的分析<sup>[9]</sup>,不仅是对毕业设计管理系统的总结,也是系统和过程管理更好完善发展借鉴的依据。

### 1.2.3 系统相关技术架构现状

目前的毕业设计管理系统在技术上的实现主要采用两大类不同的架构方式。一类是以ASP、PHP、ASP.NET等结合sql sever系列、MySQL、Access、Oracle等数据库开发的,开发语言有VB、VC、C#等;另一类是以J2EE为架构结合sql sever系列、MySQL、Access、Oracle等数据库开发,编程语言采用Java。此外,不管是哪一类方式都有选择的结合了Ajax、JavaScript、CSS等相关技术。

在B/S模式下, ASP.NET+XML+Web服务三者结合使用体现了强大的功能。采用这种技术结合开发的毕业设计管理系统具有如下特点<sup>[10]</sup>:系统在数据传输上

不同于传统的B/S模式，更能充分利用InterNet的强大功能；系统升级方便，可移植性较强，数据传输性能高；利用Web服务，为应用程序提高了接口整合。而且ASP.NET相对于ASP很好的提供了与Web服务和XML的结合性。采用具有“黄金组合”之称的Linux+Apache+MySQL+PHP开发模式，系统具有更好的稳定性和安全性以及经济性<sup>[11]</sup>。

在J2EE平台下，目前开发的管理系统，主要是结合各类框架来实现的。基于Struts +Hibernate+Spring采用快速建立原型的软件工程方法实现了毕业设计流程从选题到质量控制到文件输入的自动化<sup>[12]</sup>，系统采用三层架构，包括表示层，业务逻辑层，数据层，表示层采用的是Struts框架的视图部分和控制部分，Struts中的模型部分主要分布在业务逻辑层，数据层指数据库服务器。Struts是目前Web开发中比较流行的一个开源框架，它主要是采用Servlet和JSP技术实现MVC的设计模式，很好地使显示、控制、模型相分离。开发人员利用它开发时，不需要再编码实现MVC模式各个部分，因此利用Struts节省了开发时间<sup>[15]</sup>。

C#作为.NET的固有语言，借鉴了许多Java的技术规则。C#中也有一个虚拟机，叫做公用语言运行环境(CLR)，它的对象也具有同样的层次；C#的设计具有丰富的系统类，可以减少大量的重复劳动；由于C#与Windows在体系结构上相似，因此C#很容易被开发人员所熟悉<sup>[11]</sup>。不仅如此，尽管C#是一种新型编程语言，但是因为其由C和C++派生而来，因此具有C++的功能，所以在做MIS型WEB系统开发时，C#语言就像Java一样。

#### 1.2.4 系统安全性现状

在软件的设计阶段就要从各方面注意软件安全性，因为软件安全问题在进入开发之后才重视就会造成不可估计损失。所以通过对现有管理系统相关文献在安全性设计方面的总结，对于即将进入设计阶段的研究生学位论文进程管理系统有很大的借鉴意义。

通过角色权限的分配，可以保证系统的操作安全，防止操作越限<sup>[15]</sup>，同时给已经登录的用户设置操作超时限制，防止权限被利用。在不同角色拥有不同权限的基础上，管理员对数据库操作日志的分析可以更加安全的分析管理数据。在答辩部分的导师与学生的匹配过程中采取“双盲”机制<sup>[16,17]</sup>，更好的实现了公平性，同时在技术上设计随机抽取答辩专家组的算法来实现。

在数据库设计上主要利用了两类工具：Erwin4.1，PowerDesigner。尽管都是数据模型设计工具，但是后者涵盖了需求分析、设计、部署整个软件流程<sup>[18]</sup>。通过概念数据模型到物理数据模型的设计使得数据库的设计更加科学合理，同时在创建数据表的过程中不需要人工去Create Table，只需要工具生成，这样也给系统设计减少了出错的几率。



在数据库安全(权限、加密、安全恢复等)方面,目前主要是采用禁用Windows身份验证,防止利用系统漏洞攻击数据库,同时利用数据库本身的备份功能,预防不可恢复的数据破坏的发生。对于用户的密码安全管理,主要采用哈希散列类算法和MD5方法加密<sup>[19]</sup>。网络安全(防火墙、安全策略等)方面,采用双层防护技术:第一,采用专业防火墙,有效隔离外网恶意的入侵和攻击;第二,采用网络访问日志记录系统进行访问监测<sup>[10]</sup>。

为了避免数据处理层直接操作数据库中的表数据破坏数据的完整性,通过调用对应的存储过程来访问数据。存储过程可被作为一种安全机制来充分利用<sup>[20]</sup>。系统管理员通过对执行某一存储过程的权限进行限制,从而实现对相应的数据访问权限的限制,避免非授权用户对数据的访问,保证数据的安全。

### 1.3 本论文的意义

学位过程论文作为研究生学业完成前的一项比较重要的任务,研究生学位论文进程管理系统能更好的保证研究生的论文质量。本课题研究的意义重在如何使研究生学位论文质量得到保证的前提下,让参与其中的教师、研究生、教学秘书等能更加方便快捷的了解论文的相关信息和规范论文的写作、评审和管理工作的。

### 1.4 本论文的技术路线

通过分析目前网上毕业设计管理系统的功能和使用的技术情况,结合相关期刊文献的研究结论,从软件工程思想出发,在构建系统平台、数据库设计、系统安全性、系统功能划分等方面初步整理出思路。再与老师讨论该思路,结合老师的意见和建议重新整理出适合自己的研究方法。在系统进入开发阶段,除了学习相关需要的技术外,并做到及时与老师沟通。系统在Visual Studio 2008的平台上采用面向对象的三层架构设计,以C#语言进行开发,数据库采用SQL Sever2005,使用PowerDesigner辅助数据库设计。使用CSS技术结合母版页,控制系统界面风格统一;采用Ajax技术更好的实现页面刷新。用户密码存储采用简单的MD5加密处理,此外使用存储过程操作数据库中的表,在数据处理层只需调用相关存储过程。

### 1.5 全文体系

第一章,简述了系统的开发背景,详细分析了系统研究现状,介绍了课题研究的意义和开发本系统才去的研究方法。

第二章,从系统功能和性能等方面做了需求的阐述,此外在系统架构方面做了简单的介绍。

第三章,整理了系统开发整个过程中各方面的资料,包括系统数据库设计,编码阶段的三层架构每一层的数据处理条目等,即系统的详细设计。

第四章,主要介绍系统实现中部分功能源码和运行效果图。

第五章，是结束语。总结了全文的主要思想，分析了本次系统开发中还存在的不足和亟待改善的问题。

在网络高速发展的今天，我们更要借助网络来方便快捷的处理我们的工作，减少一些不必要的时间浪费。所以，本文通过分析设计“研究生学位论文进程管理系统”来帮助高校研究生论文整个过程的管理，当然这只是众多前人研究的基础上结合自己的分析理解重新整理一份报告，其中必然存在许多不完善的地方。相信在日后定会得到完善，并推而广之。

## 2 系统分析

### 2.1 系统三层架构设计

本系统采用三层架构设计实现，图2-1为系统三层架构设计思路图。

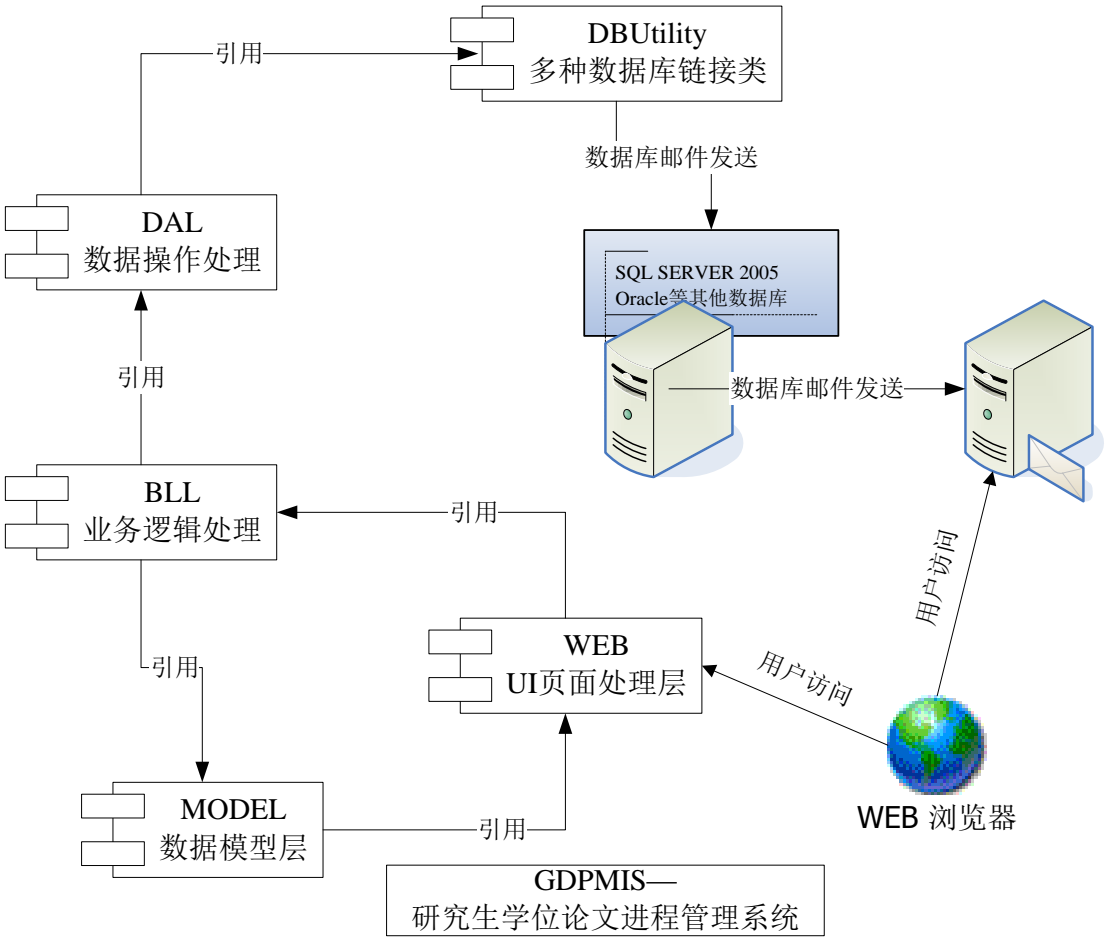


图 2-1 系统三层架构设计思路图

### 2.2 系统需求分析

本软件的最终用户有：研究生、教师、管理员。这些用户大多是高素质的用户，对于现在计算机技术都能熟练使用。本软件对于所有用户都要求能够熟练的使用Windows 操作系统，能使用IE 浏览器上网。对于研究生的要求是，能正确的按照一定规则上传下载论文；对于教师的要求是能熟练的使用word评审学生的

论文；对于管理员需要有数据库维护知识和熟练操作Excel。

同时本软件在系统开启后，分不同阶段不同用户使用频度会不同：在系统刚开启阶段，只有管理员能使用软件，初始化系统；在申请选题阶段主要是教师使用软件，管理员会参与审定选题；在选题阶段主要是研究生参与选题；在之后到答辩阶段主要是研究生与教师的交流，包括发帖、邮件以及上传下载；答辩阶段主要是教师负责打成绩；在系统开启整个阶段，管理员都有维护系统的责任。

### 2.2.1 系统功能模块描述

#### ➤ 系统管理

用户管理：分为权限设置，资料修改，用户创建

用户创建：创建用户，如学生，教师，评审员（直接从excel表导入学生和教师信息）

权限设置：用于修改用户权限，主要修改为：提升部分教师为管理员

学生权限：{用户登录，资料修改，公告浏览，选题管理，开题报告管理，中期报告管理，论文管理，帖子管理}

教师权限：{用户登录，资料修改，公告浏览，发布公告，选题审核，选题管理，开题审核，中期报告审核，论文审核，评价表管理}

管理员权限：{用户登录，公告浏览，答辩安排发布，答辩评审表管理，成绩汇总表。其中超级管理员才具有权限管理和系统设置的权限}

资料修改：用于修改密码，以及修改电话，邮箱，住址等资料

公告管理：分为公告浏览，公告发布

公告浏览：浏览公告信息，如答辩安排，选题通知等

公告发布：用于发布公告

主题管理：用于主题帖的发布和浏览以及跟帖等

用户登录：系统用户登陆，并且进行权限判断

#### ➤ 选题管理

选题管理：用于教师提出的公选课题的管理，可以发布/删除/修改课题

自选选题管理：学生发布/删除/修改自拟的选题，并等待教师审批

选题审核：教师浏览学生自拟选题，并进行审批

#### ➤ 开题管理

选题选择：管理学生的论文，为每篇论文分配评审员并且能够管理评审员的评价意见，功能包括论文上传，论文审核，分配评审员以及评价表的管理。

开题报告管理：用于管理答辩阶段，功能有答辩的安排以及答辩评审表的管理。

开题审批：用于将教师评审，评审员评审以及最终答辩的成绩综合为最终成

绩，并纳入汇总表中，最后打印提交教务处。

➤ 中期管理

中期报告管理：用于学生提交/修改/删除中期报告。

中期报告审核：教师浏览学生中期报告，并进行评审。

➤ 论文管理

论文管理：学生提交/修改/删除论文。

论文审核：教师和评审员浏览并评审论文

评价表管理：管理论文评价表，分为教师论文评价表和评审员论文评价表。

分配评审小组：教师为评审小组成员分配权限，并为评审小组分配论文。

答辩管理：分为答辩安排发布，答辩评审表管理

答辩安排发布：以公告形式发布答辩时间安排。

答辩评审表管理：提交/修改/删除 答辩评审表。

➤ 表格管理

成绩汇总表：将教师成绩，评审成绩和答辩成绩进行汇总，形成成绩汇总表，生成文件并打印。

系统功能模块划分详见图 2-2 所示。

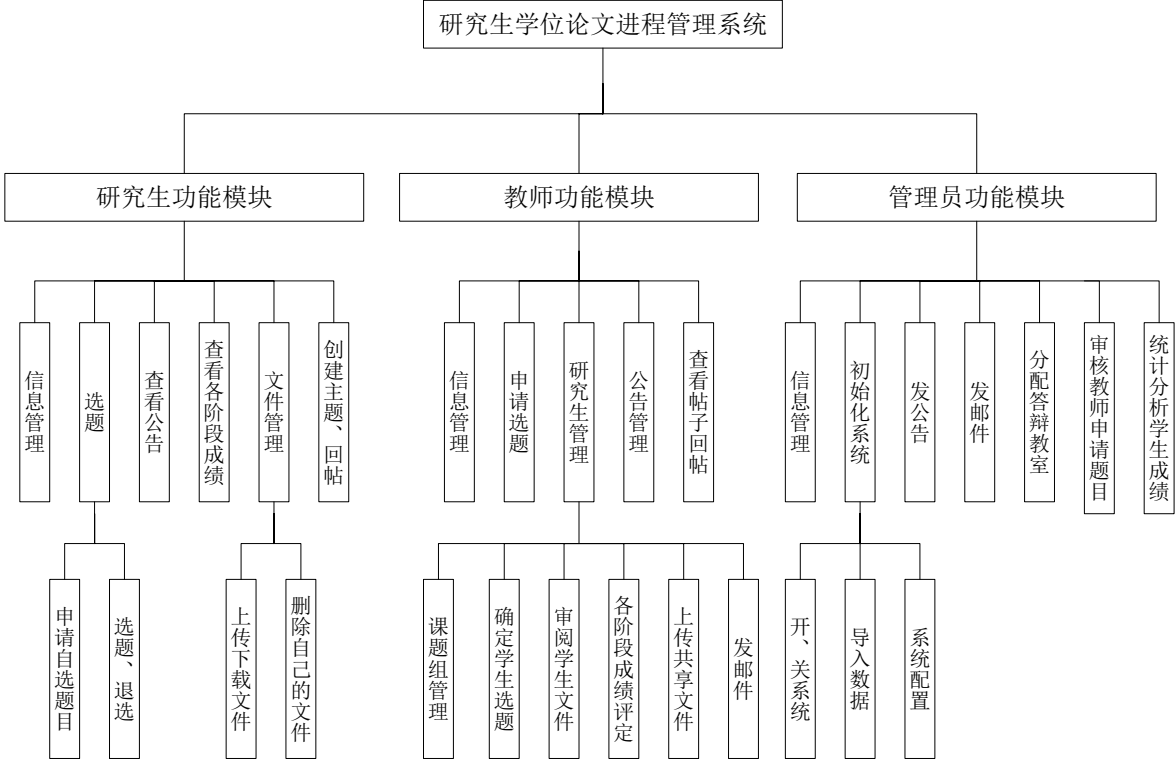


图 2-2 系统模块功能模块图

2.2.2 系统顶层数据流图

本系统的外部项/外部实体为教师、研究生和管理员。系统顶层数据流设计详见图 2-3 所示。

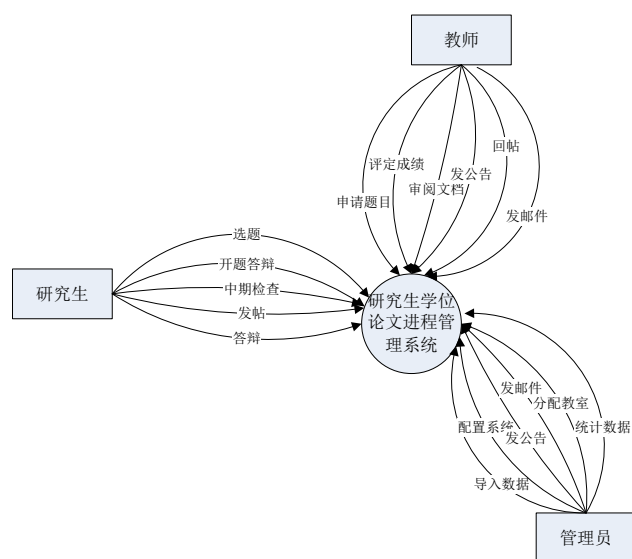


图 2-3 系统顶层数据流图

### 2.2.3 系统性能需求

系统在性能方面需满足的需求，详见如下几方面：

#### ➤ 系统精度需求

**输入数据精度：**该系统输入基本为客户端所发出的数据请求，以及更新请求，基本为文本形式，所以输入精度不需要很高。此外，系统管理员导入Excel信息表，需按照软件说明正确的设置数据列才能导入。（后面按这样排）

**输出数据精度：**输出的数据基本为数据库中的数据，和输入精度相当。

**传输精度：**采用post传输方式，对传输采取保密性。

#### ➤ 安全性

本系统由各个相互独立的模块组成，当用户要进行某种操作，必须进行登陆，系统设置用户身份验证，根据用户身份分配相应的系统操作权限，限定用户在合法的范围内进行数据操作，对于不合法用户的操作请求予以回绝严格，保证系统的数据只在规定的范围内进行正常的处理，只有在数据保密期结束后根据用户的规定将数据信息进行存档，在履行规定的程序后确定数据的访问方式和途径。本系统的密码管理采用集成数据库管理，只有使用者知道密码。

#### ➤ 易用性

用户管理是通过浏览器以B/S方式完成的，通过数据的加密传输及验证，保证安全的基础上，大大提高了系统易用性，系统管理员或业务操作员可以在任意允许的客户机上完成相应的管理工作。

#### ➤ 扩展性

模块化的开发方式，对于以后每项新增的业务都可便利的增加到系统中，业务作到了与用户管理无关性。

## 2.3 系统架构设计

系统架构设计详见图 2-4 所示。

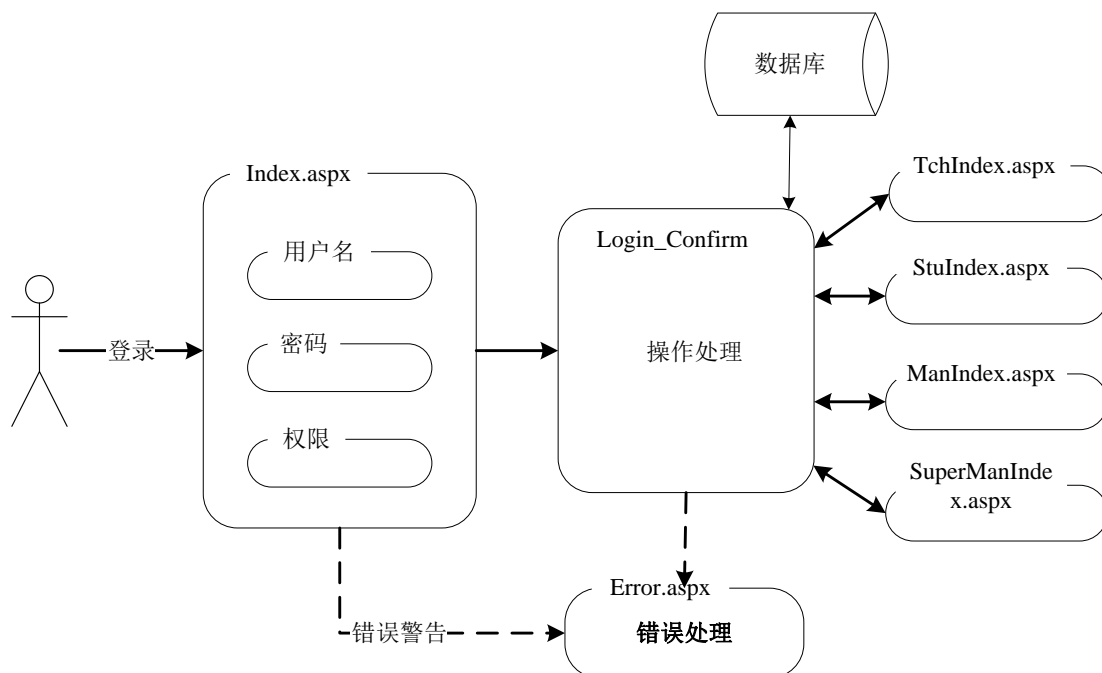


图 2-4 系统总体架构设计

## 2.4 本章小结

本章首先在系统设计概念(简单三层架构模式)上做了简单的探讨，接着，分析了“研究生学位论文进程管理系统”的功能需求，在数据处理上做了简单的分析，并在系统性能需求方面从多角度做了阐释。最后本章介绍了系统采取的架构设计。

# 3 系统详细设计

## 3.1 数据库设计

系统数据库表设计详见表 3-1 所示。

表 3-1 数据库表名

序号	表名	表信息说明
1	tb_answerNote	回帖信息表，记录的是帖子的所有回帖信息
2	tb_BATask	答辩教室分配表，记录的是不同阶段教室的分配情况
3	tb_bulletin	公告记录表，记录的是各学院的公告信息
4	tb_chooseTask	选题信息表，记录的是所有选题的信息
5	tb_config	系统配置表，记录的是各阶段时间安排的配置信息
6	tb_fileManage	文件信息表，记录的是各种文件的相关信息
7	tb_manager	管理员信息表，记录的是教学秘书和超级管理员的相关基本信息

8	tb_note	发贴信息表，记录的是主题帖的相关信息
9	tb_paperScore	成绩表，记录的是研究生不同阶段的成绩汇总表
10	tb_student	研究生信息表，记录的是研究生的基本信息表
11	tb_teacher	教师表，记录的是教师的基本信息
12	tb_theme	课题表，记录的是选题所属课题的基本信息

部分主要表设计文档：表 3-2 为答辩教室分配表，表 3-3 为选题信息表，表 3-4 为系统各阶段时间配置表，表 3-5 为研究生信息表，表 3-6 为教师信息表，表 3-7 为论文成绩表。

表 3-2 tb\_BATask

序号	列名	数据类型	长度	说明
1	bat_id	int	4	开题 答辩编号
2	bat_classRoom	varchar	8	教室
3	bat_beginTime	datetime	8	开始时间
4	bat_state	int	4	开题 答辩状态

表 3-3 tb\_chooseTask

序号	列名	数据类型	长度	说明
1	ct_id	int	4	选题编号
2	s_num	varchar	16	学号
3	t_num	varchar	16	教师工号
4	th_id	int	4	课题编号
5	ct_theme	varchar	64	选题题目
6	ct_summary	varchar	512	选题简介
7	ct_auditState	int	4	审核状态
8	ct_state	int	4	是否被选
9	ct_type	int	4	选题类型

表 3-4 tb\_config

序号	列名	数据类型	长度	说明
1	c_id	int	4	配置编号
2	c_sysState	int	4	系统状态
3	c_stuBeginTime	datetime	8	学生开始时间
4	c_stuEndTime	datetime	8	学生结束时间
5	c_teaBeginTime	datetime	8	教师开始时间
6	c_teaEndTime	datetime	8	教师结束时间
7	c_CTBeginTime	datetime	8	选题开始时间
8	c_CTEndTime	datetime	8	选题结束时间
9	c_ATBeginTime	datetime	8	答辩开始时间
10	c_ATEndTime	datetime	8	答辩结束时间

11	c_INBeginTime	datetime	8	中期检查开始时间
12	c_INEndTime	datetime	8	中期检查结束时间
13	c_BeginTime	datetime	8	开题开始时间
14	c_EndTime	datetime	8	开题结束时间

表 3-5 tb\_student

序号	列名	数据类型	长度	说明
1	t_num	varchar	16	教师工号
2	bat_id	int	4	开题 答辩编号
3	t_psw	varchar	32	教师密码
4	t_realName	varchar	32	教师姓名
5	t_sex	int	4	教师性别
6	t_level	varchar	32	教师职称
7	t_department	varchar	32	系别
8	t_mobile	varchar	16	教师手机号
9	t_email	varchar	32	教师邮件
10	t_emailPSW	varchar	32	教师邮件密码
11	t_state	int	4	开题 答辩（中期）状态
12	t_order	int	4	开题 答辩（中期）顺序

表 3-6 tb\_teacher

序号	列名	数据类型	长度	说明
1	t_num	varchar	16	教师工号
2	bat_id	int	4	开题 答辩编号
3	t_psw	varchar	32	教师密码
4	t_realName	varchar	32	教师姓名
5	t_sex	int	4	教师性别
6	t_level	varchar	32	教师职称
7	t_department	varchar	32	系别
8	t_mobile	varchar	16	教师手机号
9	t_email	varchar	32	教师邮件
10	t_emailPSW	varchar	32	教师邮件密码
11	t_state	int	4	开题 答辩（中期）状态
12	t_order	int	4	开题 答辩（中期）顺序

表 3-7 tb\_paperScore

序号	列名	数据类型	长度	说明
1	ps_id	int	4	成绩编号
2	s_num	varchar	16	学号
3	ps_score1	numeric	5	成绩一



4	ps_score2	numeric	5	成绩二
5	ps_score3	numeric	5	成绩三
6	ps_score4	numeric	5	成绩四
7	ps_score5	numeric	5	成绩五
8	ps_score6	numeric	5	成绩 6
9	ps_score7	numeric	5	成绩 7
10	ps_score	numeric	5	成绩小计（教学秘书计算过程）
11	ps_type	int	4	成绩类型：1-文献综述；2-文献翻译；3-开题报告；4-中期检查；5-论文答辩。

系统的数据库使用工具PowerDesigner软件辅助设计，系统数据库概念模型设计详见图 3-1、3-2 所示。

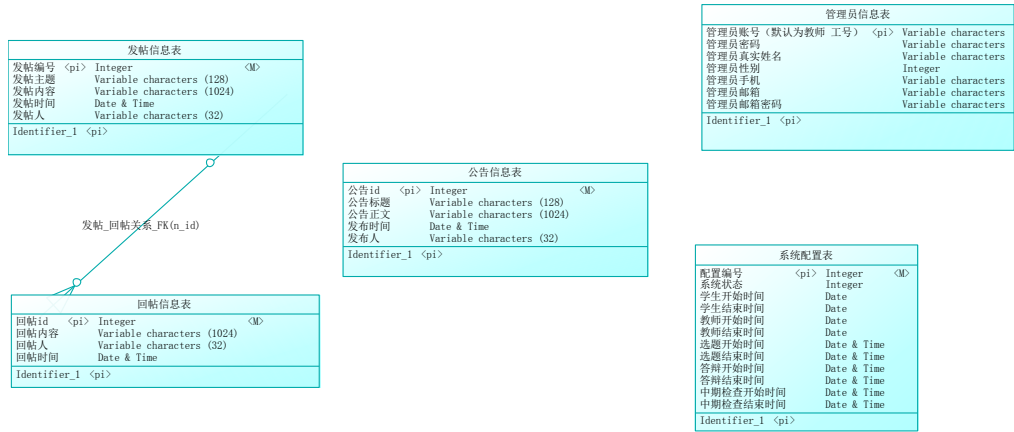


图 3-1 概念模型 1

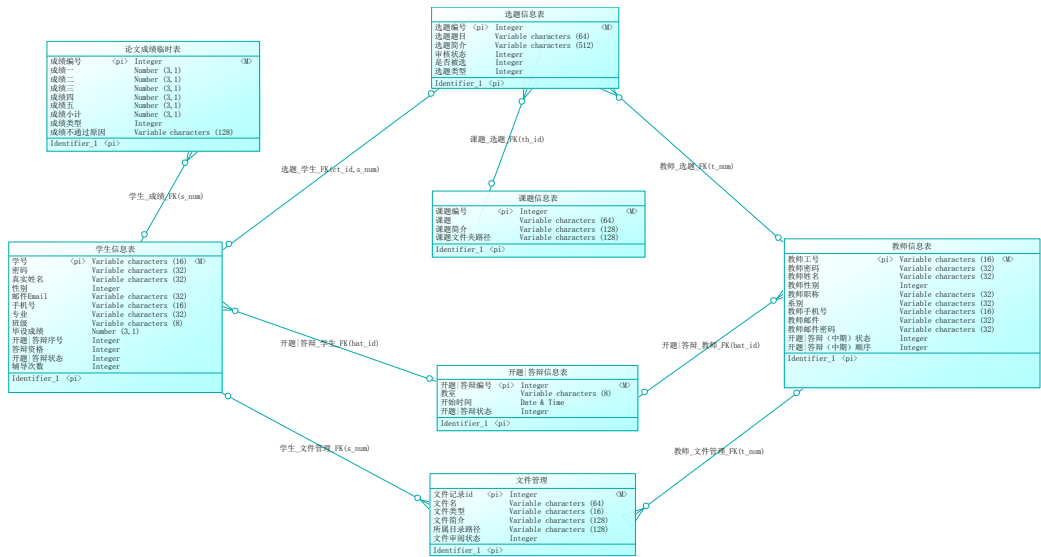


图 3-2 概念模型 2

在确定了各表主键字段的基础上,依据表与表之间相关字段的联系建立了各表之间的关系，系统数据表之间关联关系详见图 3-3 所示。

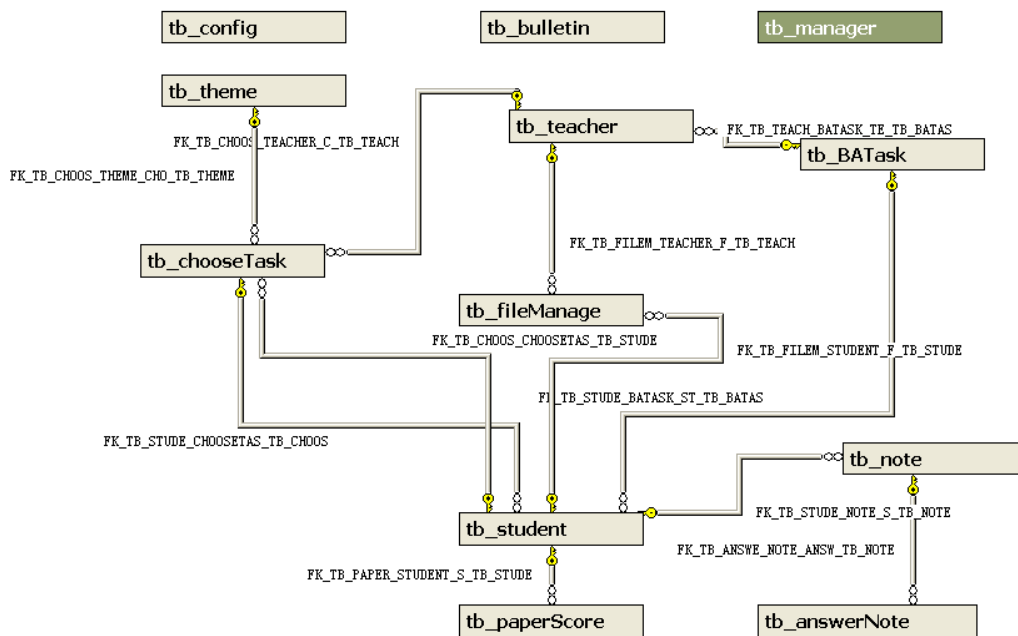


图 3-3 表关联图

存储过程设计详见表 3-8 所示。

表 3-8 存储过程简表

过程名	过程说明	过程名	过程说明
PR_InitDataBase	初始化系统	Pr_fileManage_GetModel	文件详细
PR_DataBackUp	备份数据库	Pr_fileManage_GetList	文件列表
PR_DataRestore	恢复数据库	Pr_manager_Exists	管理员是否存在
PR_SysAutoEmail	系统自动邮件	Pr_manager_ADD	添加管理员
Pr_answerNote_ADD	添加回帖	Pr_manager_Update	更新管理员信息
Pr_answerNote_GetModel	获取回帖	Pr_manager_Delete	删除管理员
Pr_answerNote_GetList	获取回帖列表	Pr_manager_GetModel	获取管理员信息
Pr_BATask_ADD	增加答辩安排	Pr_manager_GetList	获取管理员列表
Pr_BATask_Update	更新答辩安排	Pr_note_ADD	发布新帖
Pr_BATask_Delete	删除答辩安排	Pr_note_GetModel	获取帖子信息
Pr_BATask_GetModel	获取答辩安排	Pr_note_GetList	获取帖子列表
Pr_BATask_GetList	获取答辩安排列表	Pr_paperScore_ADD	增加成绩
Pr_bulletin_ADD	添加公告	Pr_paperScore_Update	更新成绩
Pr_bulletin_Delete	删除公告	Pr_paperScore_GetModel	获取成绩详细
Pr_bulletin_GetModel	获取公告详细	Pr_paperScore_GetList	获取成绩列表
Pr_bulletin_GetList	获取公告列表	Pr_student_Exists	研究生是否存在
Pr_chooseTask_ADD	增加选题	Pr_student_ADD	添加研究生
Pr_chooseTask_Update	修改选题	Pr_student_Update	研究生更新信息
Pr_chooseTask_Delete	删除选题	Pr_student_GetModel	获取研究生信息
Pr_chooseTask_GetModel	获取选题详细	Pr_student_GetList	获取研究生列表
Pr_chooseTask_GetList	获取选题列表	Pr_teacher_Exists	教师是否存在



表 3-10 系统模块概要设计

编号	功能名称	子模块	子程序	功能描述
1	研究生功能模块	登陆	StuLogin Config.GetModel	研究生登录
		信息查看、修改	Student.GetModel Student.Update	研究生基本信息管理
		选题	ChooseTask.GetList ChooseTask.Update	选择公选选题
		自定义选题	ChooseTask.Add	研究生申请自定义选题
		退选选题	ChooseTask.Update	教师审核还未确定，即可退选
		查看公告	Bulletin.GetModel Bulletin.GetList	查看各种公告
		创建主题帖	Note.Add	发布新的帖子
		回复主题贴	AnswerNote.Update Student.Update	回复自己的帖子或则别人的帖子
		上传下载文档	FileManage.Add FileManage.Update FileManage.GetModel FileManage.GetList	文档程序等的上传下载
		查看答辩安排	BATask.GetModel	各阶段的答辩安排情况查看
		查看各阶段成绩	Paper.GetModel	各阶段的成绩查看
2	教师功能模块	登陆	Teacher.Login Config.GetModel	教师登陆
		信息查看、修改	Teacher.GetModel Teacher Update	教师基本信息管理
		公选选题管理	ChooseTask.Add ChooseTask.GetModel ChooseTask.GetList ChooseTask.Delete Theme.GetList	申请公选选题，查看选题审核状态，修改选题，删除未通过的选题
		审核研究生选题	ChooseTask.Update	审核研究生选择的选题
		发布公告	Bulletin.Add	发布公告
		发邮件	Smtp 服务	给研究生发邮件
		回复研究生主题贴	AnswerNote.Add Student.Update	回复研究生主题贴，回复一次增加研究生辅导次数
		下载研究生文档查看	FileManage.GetModel FileManage.GetList Student.Update	下载研究生文档查看，重新上传，则增加研究生辅导次数一次

		查看答辩安排	BATask.GetModel	各阶段的答辩安排情况查看
		教师评审成绩	Paper.Add Paper.Update	各阶段的成绩评审（评审委员会）
		上传下载共享文件	FileManage.Add FileManage.GetList FileManage.GetModel	上传下载共享文件
3	教学秘书 功能模块	登陆	Manager.Login	教学秘书登陆
		信息查看、修改	Manager.Update Manager.GetModel	教学秘书基本信息管理
		注册教学秘书	Manager.Add	新增本学院新的教学秘书
		审核教师申请的选题	ChooseTask.Update ChooseTask.GetModel	审核教师申请的选题，不通过的增加理由
		公告管理	Bulletin.Add Bulletin.GetModel Bulletin.GetList Bulletin.Delete	新增公告、删除公告
		发邮件	Smtp 服务	群发邮件通知
		分配答辩教室	Teacher.GetList Student.GetList BATask.Add BATask.Update	分配各阶段的答辩教室，包括评审委员会的分配
		成绩统计	Paper.GetList	成绩的统计分析
		研究生管理	Student.GetList	管理研究生信息，设置研究生状态
4	超级管理员 功能模块	登陆	Manager.Login	超级管理员登陆
		基本信息管理	Manager.Update Manager.GetModel	超级管理员基本信息管理
		系统初始化	Sys.ImportExcel Sys.Init	初始化系统，导入研究生表、教师表
		备份系统数据	Sys.BakUp	备份数据库
		配置系统	Config.Add Config.Update Config.GetModel	设置系统各阶段的时间
		公告管理	Bulletin.Add Bulletin.GetModel Bulletin.GetList Bulletin.Delete	新增公告、删除公告
		发邮件	Smtp 服务	群发邮件通知
		研究生管理	Student.GetList	查看各学院研究生信息
		教师管理	Teacher.GetList	查看各学院教师信息

### 3.4 系统各层详细设计

#### 3.4.1 模型层(Model)设计

模型层模型设计详细信息见表 3-11。

表 3-11 Model 层设计

编号	模型类名称	属性、方法	功能描述
1	Model.AnswerNote	tb_answerNote 表各字段的 set{},get{}	回帖实体类构造
2	Model.BATask	tb_BATask 表各字段的 set{},get{}	教室分配实体类构造
3	Model.Bulletin	tb_bulletin 表各字段的 set{},get{}	公告实体类构造
4	Model.ChooseTask	tb_chooseTask 表各字段的 set{},get{}	选题实体类构造
5	Model.Config	tb_config 表各字段的 set{},get{}	配置表实体类构造
6	Model.FileManage	tb_fileManage 表各字段的 set{},get{}	文件管理实体类构造
7	Model.Manager	tb_manager 表各字段的 set{},get{}	教学秘书实体类构造
8	Model.Note	tb_note 表各字段的 set{},get{}	主题帖实体类构造
9	Model.PaperScore	tb_paperScore 表各字段的 set{},get{}	论文成绩实体类构造
10	Model.Student	tb_student 表各字段的 set{},get{}	研究生实体类构造
11	Model.Teacher	tb_teacher 表各字段的 set{},get{}	教师实体类构造
12	Model.Theme	tb_theme 表各字段的 set{},get{}	课题实体类构造

### 3.4.2 数据处理层(DAL)设计

数据处理层设计详细信息见表 3-12。

表 3-12 DAL 层设计

编号	处理类名称	属性、方法	功能描述
1	DAL.AnswerNote 数据访问类 AnswerNote	public void Add(Model.AnswerNote model)	增加一条数据
		public Model.AnswerNote GetModel(int an_id)	得到一个对象实体
		public DataSet GetList(int n_id)	获取列表 DataSet
2	DAL.BATask 数据访问类 BATask	public void Add(Model.BATask model)	增加一条数据
		public void Update(Model.BATask model)	更新一条数据
		public void Delete(int bat_id)	删除一条数据
		public Model.BATask GetModel(int bat_id)	得到一个对象实体
		public DataSet GetListDS(string strWhere)	获取 DataSet 列表
		SqlDataReader GetListDR(string strWhere)	获取 datareader 列表
3	DAL.Bulletin 数据访问类 Bulletin	public void Add(Model.Bulletin model)	增加一条数据
		public void Update(Model.Bulletin model)	更新一条数据
		public void Delete(int be_id)	删除一条数据
		public Model.Bulletin GetModel(int be_id)	得到一个对象实体
		public DataSet GetList(string strWhere, string strOrderBy)	获取列表
4	DAL.ChooseTask 数据访问类 ChooseTask	public int Add(Model.ChooseTask model)	增加一条数据
		public void Update(Model.ChooseTask model)	更新一条数据
		public void Delete(int ct_id)	删除一条数据

		public Model.ChooseTask GetModel(int ct_id)	得到一个对象实体
		Model.ChooseTask GetModel(string s_num)	得到一个对象实体
		public DataSet GetListDS(string strWhere)	获取列表
5	DAL.Config 数据访问类 Config	public void Add(Model.Config model)	增加一条数据
		public void Update(Model.Config model)	更新一条数据
		public Model.Config GetModel(int c_id)	得到一个对象实体
		public DataSet GetList(string strWhere, string strOrderBy)	获取列表
6	DAL.FileManage 数据访问类 FileManage	public bool Exists(string fm_userNum, string fm_fileName)	是否存在
		public void Add(Model.FileManage model)	增加一条数据
		public void Update(Model.FileManage model)	更新一条数据
		public void Delete(int fm_id)	删除一条数据
		public Model.FileManage GetModel(int fm_id)	得到一个对象实体
		public DataSet GetList(string strWhere)	获取列表
7	DAL.Manager 数据访问类 Manager	public bool Exists(string m_num)	是否存在
		public void Add(Model.Manager model)	增加一条数据
		public void Update(Model.Manager model)	更新一条数据
		public void Delete(string m_userName)	删除一条数据
		Model.Manager GetModel(string m_userName)	得到一个对象实体
		public DataSet GetListDS(string strWhere)	获取列表
		SqlDataReader GetListDR(string strWhere)	获取列表
8	DAL.Note 数据访问类 Note	public void Add(Model.Note model)	增加一条数据
		public void Update(Model.Note model)	更新一条数据
		public void Delete(int n_id)	删除一条数据
		public Model.Note GetModel(int n_id)	得到一个对象实体
		public DataSet GetList(string s_num, string s_departMent)	获取列表
9	DAL.PaperScore 数据访问类 PaperScore	public int Exists(string s_num, int ps_type)	是否存在
		public void Add(Model.PaperScore model)	增加一条数据
		public void Update(Model.PaperScore model)	更新一条数据
		public void Delete(int ps_id)	删除一条数据
		public Model.PaperScore GetModel(int ps_id)	得到一个对象实体
		public DataSet GetListDS(string strWhere)	获取列表
		SqlDataReader GetListDR(string strWhere)	获取列表
10	DAL.Student 数据访问类 Student	public bool Exists(string s_num)	是否存在
		public void Add(Model.Student model)	增加一条数据
		public void Update(Model.Student model)	更新一条数据

		public void Delete(string s_num)	删除一条数据
		public Model.Student GetModel(string s_num)	得到一个对象实体
		public DataSet GetListDS(string strWhere)	获取列表
		SqlDataReader GetListDR(string strWhere)	获取列表
11	DAL.Teacher 数据访问类 Teacher	public bool Exists(string t_num)	是否存在
		public void Add(Model.Teacher model)	增加一条数据
		public void Update(Model.Teacher model)	更新一条数据
		public void Delete(string t_num)	删除一条数据
		public Model.Teacher GetModel(string t_num)	得到一个对象实体
		public DataSet GetListDS(string strWhere)	获取列表
		SqlDataReader GetListDR(string strWhere)	获取列表
12	DAL.Theme 数据访问类 Theme	public void Add(Model.Theme model)	增加一条数据
		public void Update(Model.Theme model)	更新一条数据
		public void Delete(int th_id)	删除一条数据
		public Model.Theme GetModel(int th_id)	得到一个对象实体
		SqlDataReader GetListDR(string strWhere)	获取列表
		public DataSet GetListDS(string strWhere)	获取列表

### 3.4.3 逻辑处理层(BLL)设计

逻辑处理层设计详细信息见表 3-13。

表 3-13 BLL 层设计

编号	处理类名称	属性、方法	功能描述
1	BLL.AnswerNote 逻辑处理类 AnswerNote	public void Add(Model.AnswerNote model)	增加一条数据
		public Model.AnswerNote GetModel(int an_id)	得到一个对象实体
		public DataSet GetListByN_ID(int n_id)	获取主题的所有回帖
2	BLL.BATask 逻辑处理类 BATask	public void Add(Model.BATask model)	增加一条数据
		public void Update(Model.BATask model)	更新一条数据
		public void Delete(int bat_id)	删除一条数据
		public Model.BATask GetModel(int bat_id)	得到一个对象实体
		public DataSet GetAllListByState(int bat_state,string bat_departMent)	获取某状态的所有教室或分组
		public SqlDataReader GetNotCheckByState(int bat_state, string bat_departMent)	获取某状态未被分配的教室或分组
		public SqlDataReader GetCheckedByState(int bat_state, string bat_departMent)	获取某状态已被分配的教室或分组
3	BLL.Bulletin 逻辑处理类 Bulletin	public void Add(Model.Bulletin model)	增加一条数据
		public void Update(Model.Bulletin model)	更新一条数据
		public void Delete(int be_id)	删除一条数据



		public Model.Bulletin GetModel(int be_id)	得到一个对象实体
		public DataSet GetListByDepart(string departMent)	根据系部公告列表
		public DataSet GetAllList()	获得所有公告列表
4	BLL.ChooseTask 逻辑处理类 ChooseTask	public int Add(Model.ChooseTask model)	增加一条数据
		public void Update(Model.ChooseTask model)	更新一条数据
		public void Delete(int ct_id)	删除一条数据
		public Model.ChooseTask GetModel(int ct_id)	得到一个对象实体
		Model.ChooseTask GetModel(string s_num)	得到一个对象实体
		public DataSet GetListAllAudit(string t_departMent)	获取所有待审核选题
		public DataSet GetListByThemeAudit(int th_id)	获取课题所有待审核选题
		public DataSet GetListByTchAudit(string t_num)	获取老师所有待审核选题
		public DataSet GetCTList(int th_id)	根据课题查找可选的选题
		public DataSet GetCTList(string t_num)	获取教师所有可选的选题
		public DataSet GetCTAllList(string t_departMent)	获取某系所有可选择的选题
		public DataSet TchGetMyList(string t_num)	获取教师所有选题-教师管理自己选题
		public int StuChooseCT(string s_num, int ct_id)	研究生选择选题
		public void DelSNum(int ct_id)	研究生退选选题
		public Model.ChooseTask StuHasChoosed(string s_num)	判断研究生是否已经选题
5	BLL.Config 逻辑处理类 Config	public void Add(Model.Config model)	增加一条数据
		public void Update(Model.Config model)	更新一条数据
		public Model.Config GetModel(int c_id)	得到一个对象实体
		public Model.Config GetConfig()	获取配置表
6	BLL.FileManage 逻辑处理类 FileManage	public bool Exists(string fm_userNum, string fm_fileName)	是否存在
		public void Add(Model.FileManage model)	增加一条数据
		public void Update(Model.FileManage model)	更新一条数据
		public void Delete(int fm_id)	删除一条数据
		public Model.FileManage GetModel(int fm_id)	得到一个对象实体
		public DataSet GetStuAllFile(string s_num)	获取学生所有文件列表
		public DataSet GetTchShareFile(string t_num)	获取教师共享文件

			信息
		public string GetStuFileFolder(string s_num)	获取学生文件夹路径
		public DataSet GetStuCheckFile(string s_num)	获取学生需要审核的所有文件
		public string GetStuShareFolder(string s_num)	获取学生共享文件夹路径
		public string GetShareFolder(int th_id)	获取共享文件夹路径
		public int UpLoadFile(Model.FileManage m)	更新文件信息: 审查文件才需更新数据库
		public DataSet GetListExcel(string departMent)	获取某系所有答辩安排 Excel 文件
		public DataSet GetListExcelPaper(string departMent)	获取某系所有成绩 Excel 文件
7	BLL.Manager 逻辑处理类 Manager	public bool Exists(string m_num)	是否存在
		public void Add(Model.Manager model)	增加一条数据
		public void Update(Model.Manager model)	更新一条数据
		public void Delete(string m_userName)	删除一条数据
		Model.Manager GetModel(string m_userName)	得到一个对象实体
		public int Login(string userName, string userPsw)	管理员登陆
		public DataSet ExecleDs(string filePath, string fileName)	读取 Excel 数据表
		public void BackUp(string backFileName, string dataBaseName)	备份数据库
		public void InitDataBase()	初始化系统, 删除所有表数据
		public int HasDepartMentMan()	判断是否生成了教学秘书
		public int AddDepartMentMan()	一键生成各学院教学秘书
		public SqlDataReader GetListByDepartMent(string m_departMent)	获取某系的所有教学秘书列表
8	BLL.Note 逻辑处理类 Note	public void Add(Model.Note model)	增加一条数据
		public void Update(Model.Note model)	更新一条数据
		public void Delete(int n_id)	删除一条数据
		public Model.Note GetModel(int n_id)	得到一个对象实体
		public DataSet GetNoteDepart(string s_departMent)	获取某系的所有帖子

		public DataSet GetNoteSNum(string s_num)	获取某学生的所有帖子
9	BLL.PaperScore 逻辑处理类 PaperScore	public int Exists(string s_num, int ps_type)	是否存在
		public void Add(Model.PaperScore model)	增加一条数据
		public void Update(Model.PaperScore model)	更新一条数据
		public void Delete(int ps_id)	删除一条数据
		public Model.PaperScore GetModel(string s_num, int ps_type)	得到一条成绩记录
		public double GetTotalScore(string s_num)	获取研究生的加权成绩
10	BLL.Student 逻辑处理类 Student	public bool Exists(string s_num)	是否存在
		public void Add(Model.Student model)	增加一条数据
		public void Update(Model.Student model)	更新一条数据
		public void Delete(string s_num)	删除一条数据
		public Model.Student GetModel(string s_num)	得到一个对象实体
		public int Login(string userName, string userPsw)	研究生登陆
		public void UpdateHelpCount(string stuNum)	更新教师辅导次数
		public SqlDataReader GetStuListByTch(string t_num)	获取教师所带的所有学生
		public DataSet GetStuListByTchDS(string t_num)	获取教师所带的所有学生
		public SqlDataReader GetStuListByTch(string t_num, int s_answerquality)	获取教师某阶段所带的所有学生
		public SqlDataReader GetListByBAT(int bat_id)	获取某教室的所有学生
		public Model.ChooseTask GetCTByStu(string s_num)	获取研究生的选题信息
		public SqlDataReader GetStuListByBat(string t_num)	获取教师评审的学生列表
11	BLL.Teacher 逻辑处理类 Teacher	public bool Exists(string t_num)	是否存在
		public void Add(Model.Teacher model)	增加一条数据
		public void Update(Model.Teacher model)	更新一条数据
		public void Delete(string t_num)	删除一条数据
		public Model.Teacher GetModel(string t_num)	得到一个对象实体
		public int Login(string userName, string userPsw)	教师登陆
		public SqlDataReader GetDepartMent()	获取系部列表
		public SqlDataReader GetListDR(string t_departMent, int state)	获取教师姓名列表
		public DataSet GetListNotChecked(string t_departMent,int _state)	获取某系还没有分配教室的教师列表

		public SqlDataReader GetListByBATDR(int bat_id)	获取某教室的所有老师
		public DataSet GetAllTch(string t_departMent)	获取某系所有教师
12	BLL.Theme 逻辑处理类 Theme	public void Add(Model.Theme model)	增加一条数据
		public void Update(Model.Theme model)	更新一条数据
		public void Delete(int th_id)	删除一条数据
		public Model.Theme GetModel(int th_id)	获取某系所有课题
		public SqlDataReader GetListByDepartMent(string th_departMent)	获取列表
		public SqlDataReader GetListByTch(string t_num)	获取教师的所有被选的选题的所属课题列表
13	BLL.ChineseSpell 获取拼音简码	public string GetChineseSpell(string strText)	获取拼音简码
		private string getSpell(string cnChar)	获取字符的简码
14	BLL.DESEncrypt 加密解密	public string GetEncrypt(string psw)	加密字符串
		public string GetDecrypt(string psw)	解密字符串
15	BLL.MyWord Word 处理	public MyWord(string newWordName)	实例化 Word 对象
		public MyWord(string dotFileName, string newWordName)	实例化 Word 对象, 通过模板
		public void UpdateBookMart(string bookMart, string bookMartText)	更新模板中的书签
		public void SaveAsWord()	存储 word
16	BLL.Email 邮件发送	public bool SendEmail(string fromEmail, string[] toEmailArr, string emailPsw, string subject, string contend)	发送邮件

### 3.4.4 显示层 (WEB)设计

研究生模块详细设计见表 3-14。

表 3-14 研究生 WEB 设计

编号	页面相对路径	功能描述
1	~/Student/StuInfo.aspx	研究生个人信息查看、修改
2	~/FileManage/FileUpLoad.aspx	研究生文件管理, 上传、下载、删除
3	~/FileManage/FileShare.aspx	研究生查看自己选题所属课题组的共享资料
4	~/Bulletin/BulletinList.aspx	研究生查看本学院的公告
5	~/Note/AddNote.aspx	研究生发布主题帖
6	~/Note/NoteList.aspx	研究生查看所有主题帖
7	~/Note/MyNote.aspx	研究生管理自己发的所有主题帖, 查看、删除
8	~/ChooseTask/CTList.aspx	研究生查看公选选题列表、在规定时间内选题
9	~/ChooseTask/AddCT.aspx	研究生自拟题目
10	/ChooseTask/StuMyCT.aspx	研究生查看自己的选题状态、条件允许可以退选

11	~/BATask/ExportExcel.aspx	研究生查看答辩安排、各阶段的答辩安排表下载
----	---------------------------	-----------------------

教师模块详细设计见表 3-15。

表 3-15 教师 WEB 设计

编号	页面相对路径	功能描述
1	~/Teacher/TeacherIndex.aspx	教师登陆首页
2	~/Teacher/TeacherInfo.aspx	教师个人信息查看、修改
3	~/Student/Tch_StuManList.aspx	教师管理自己带的研究生、查看信息、发邮件
4	~/FileManage/TchFileUpLoad.aspx	教师文件管理，上传下载学生文件，上传共享文件、删除共享文件
5	~/FileManage/FileShare.aspx	教师查看按照不同的课题查看共享文件
6	~/Bulletin/AddBulletin.aspx	教师发布公告
7	~/Bulletin/BulletinList.aspx	教师查看自己学院或超级管理员的公告
8	~/Note/NoteList.aspx	教师查看学生的主题帖，回帖
9	~/ChooseTask/AddCT.aspx	教师增加公选选题
10	~/ChooseTask/TchMyCTList.aspx	教师查看自己申请的公选选题的状态、修改、删除、确定研究生选择自己的选题
11	~/BATask/ExportExcel.aspx	教师查看答辩安排，下载各阶段的答辩分配表
12	~/FileManage/TchCheckFile.aspx	教师检查自己所带研究生的相关文件
13	~/FileManage/AuditFile.aspx	教师作为评审委员时，评审该小组的学生文件

教学秘书模块详细设计见表 3-16。

表 3-16 教学秘书 WEB 设计

编号	页面相对路径	功能描述
1	~/Manager/ManInfo.aspx	教学秘书个人信息查看、修改
2	~/Teacher/Man_TchList.aspx	教学秘书管理教师、查看教师所带研究生等
3	~/Manager/ManIndex.aspx	教学秘书登陆首页
4	~/Manager/ImportExcel.aspx	教学秘书导入研究生信息表、教师信息表
5	~/Manager/AddManager.aspx	教学秘书注册新的教学秘书
6	~/Bulletin/AddBulletin.aspx	教学秘书发布公告
7	~/Bulletin/DelBulletin.aspx	教学秘书查看公告、删除公告
8	~/Theme/AddTheme.aspx	教学秘书增加课题
9	~/Theme/ThemeList.aspx	教学秘书管理课题，修改
10	~/ChooseTask/AuditCTList.aspx	教学秘书审核公选选题、自拟选题
11	~/BATask/AddBATask.aspx	教学秘书根据不同阶段增加答辩教室
12	~/BATask/BATaskList.aspx	教学秘书查看教室分配情况
13	~/BATask/AssignBATask.aspx	教学秘书分配各阶段的答辩安排
14	~/BATask/ExportExcel.aspx	教学秘书更新答辩分配的 Excel 表，下载
15	~/Paper/TchAddPaper.aspx	教学秘书上传研究生成绩单，更新研究生成绩表

超级管理员模块详细设计见表 3-17。

表 3-17 超级管理员 WEB 设计

编号	页面相对路径	功能描述
1	~/Manager/ManInfo.aspx	超级管理员个人信息查看、修改
2	~/Manager/ImportExcel.aspx	超级管理员导入研究生信息表和教师信息表
3	~/Manager/SuperManIndex.aspx	超级管理员登陆首页
4	~/Manager/DepartmentManager.aspx	超级管理员为各学院自动生成一名教学秘书
5	~/Manager/SysInit.aspx	超级管理员初始化系统：清空系统数据，备份系统数据
6	~/Manager/SysConfig.aspx	超级管理员配置系统各阶段时间安排
7	~/Bulletin/AddBulletin.aspx	超级管理员发布公告
8	~/Bulletin/DelBulletin.aspx	超级管理员查看、删除各学院公告

3.5 系统部分功能时序图

本节主要介绍该系统主要部分的功能时序图。

文件上传时序状态详见图 3-5 所示。



图 3-5 文件上传时序图

邮件发送时序状态详见图 3-6 所示。

教师确定研究生选题时序状态详见图 3-7 所示。

3.6 本章小结

本章首先介绍了系统的数据库方面的设计，包括数据库的表设计、数据库的

概念模型设计以及数据处理(存储过程)设计。接着介绍了系统各模块的结构设计，分别从模型层、数据处理层、逻辑处理层和 WEB 显示层几个方面介绍了系统的详细设计。最后介绍了系统部分重要功能的时序图设计。



图 3-6 发送邮件时序图

4 系统功能的实现

4.1 数据库连接

数据访问涉及四个主要的组件：Web 应用程序 (ASP.NET)、数据层 (ADO.NET)、数据提供程序，以及真正的数据源。这些组件之间的关系构成了所有数据识别 Web 应用程序的基础结构。ADO.NET 是对传统 ADO 的改进，可用于

表 4-1 ADO.NET 数据提供程序的核心对象

对象	描述
Connection	建立到指定资源的连接
Command	对一个数据源执行命令。公开 Parameters,在 Connection 的 Transaction 范围内执行。
DataReader	从一个数据源读取只进的只读数据流。

DataAdapter	填充一个 DataSet，解析数据源的更新。
-------------	------------------------

创建分布式的数据共享应用程序。它是一种高级的应用程序编程接口，面向支持对数据进行断开连接访问的松耦合的、n 层的、基于 Internet 的应用程序。它是 Microsoft .NET Framework 的核心组件。

表 4-1 显示组成 ADO.NET 数据提供程序的四个核心对象。



(你还没有审核！)

(你还没有配置邮箱密码，所以不能给学生自动发邮件。)

Msg.ShowMsg("成功发送邮件!")

Msg.ShowMsg("邮件发送失败，请确定你的邮箱配置！")

Msg.ShowMsg("审核数据已经更新!")

图 3-7 教师确定研究生选题时序图

本系统为方便数据库的更改，主要通过使用 SqlConnection 对象连接一个数据库（建立一个类 DB），代码如下：

```
private static readonly string connectionString = ConfigurationManager.ConnectionStrings["GDPMISConn"].ToString();
private SqlConnection con = null;
private void Open()
{
    // 打开数据库连接
    if (con == null)
    {
        con = new SqlConnection(ConfigurationManager.ConnectionStrings["GDPMISConn"].ConnectionString);
        if (con.State == ConnectionState.Closed)
```



```

    {
        try{//打开数据库连接
            con.Open();}
        catch (Exception ex)
        { DbHelperSQL.CreateErrorMsg(ex.Message);}
        finally{//关闭已经打开的数据库连接
            con.Close();}
    }
}

```

在 Web.Config 文件中添加如下代码，以方便用户在不同计算机上运行本软件时，对数据库进行必要的修改。

```

<connectionStrings>
<add name="GDPMISConn" connectionString="data Source=.;Initial Catalog
=GDPMIS;Persist Security Info=True;user id=sa;pwd=null" providerName="System.Data.SqlClient"/>
</connectionStrings>

```

## 4.2 主要功能实现

本节就系统中部分功能的实现做简单的介绍。

### 4.2.1 教学秘书分配答辩安排

该功能实现关键源码：(源码实现了一个教室最多只能分配六个教师)

```

protected void cbl_TeacherList_SelectedIndexChanged(object sender,EventArgs e)
{
    DataTable dt = new DataTable();
    dt.Columns.Add("t_num", typeof(string));
    dt.Columns.Add("t_nameLevel", typeof(string));
    int count = 0;
    for (int i = 0; i < cbl_TeacherList.Items.Count; i++)
    {
        if (cbl_TeacherList.Items[i].Selected)
        {
            count++;
            if (count == 6)
            {
                for (int j = 0; j < cbl_TeacherList.Items.Count; j++)
                {
                    if (!cbl_TeacherList.Items[j].Selected)
                    { //选择六个教师后，其余的不可选状态
                        cbl_TeacherList.Items[j].Enabled = false;
                    }
                }
            }
        }
    }
}

```

```

    }
    if (count == 5)
    {
        for (int j = 0; j < cbl_TeacherList.Items.Count; j++)
        { //选择五个教师后，全部呈可选状态
            cbl_TeacherList.Items[j].Enabled = true;
        }
    }
    DataRow dr = dt.NewRow();
    dr["t_num"] = cbl_TeacherList.Items[i].Value;
    dr["t_nameLevel"] = cbl_TeacherList.Items[i].Text;
    dt.Rows.Add(dr);
}
}
if (dt.Rows.Count < 1)
{ lbl_Msg2.Visible = true; } else
{ lbl_Msg2.Visible = false; }
rbl_TchChecked.DataSource = dt.DefaultView;
rbl_TchChecked.DataBind();
}

```

该功能效果，详见图 4-1 所示。

【李若娜--计算机学院-教学秘书】 Logout

### 答辩教室分配

答辩状态:

分配教室:

教学秘书:

可选择教师列表:

<input type="checkbox"/> 陈 运--教授	<input type="checkbox"/> 张洪伟--教授	<input type="checkbox"/> 王 臻--副教授
<input type="checkbox"/> 何 嘉--教授	<input type="checkbox"/> 陈 麟--副教授	<input checked="" type="checkbox"/> 魏 维--副教授
<input type="checkbox"/> 李 超--教授	<input checked="" type="checkbox"/> 方 睿--副教授	<input type="checkbox"/> 徐 虹--副教授
<input checked="" type="checkbox"/> 李 飞--教授	<input type="checkbox"/> 高献伟--副教授	<input type="checkbox"/> 徐振明--副教授
<input type="checkbox"/> 路而红--教授	<input checked="" type="checkbox"/> 金 虎--副教授	<input type="checkbox"/> 余贞侠--副教授
<input checked="" type="checkbox"/> 王海春--教授	<input type="checkbox"/> 王 鹏--副教授	<input checked="" type="checkbox"/> 常 征--副研究员
<input type="checkbox"/> 杨义先--教授		

友情提示

分配答辩安排成功，请及时更新分配的Excel表！

确定

已选择教师列表：必须从以下教师中选择一个作为主席

<input checked="" type="radio"/> 李 飞--教授	<input type="radio"/> 方 睿--副教授	<input checked="" type="radio"/> 魏 维--副教授
<input type="radio"/> 王海春--教授	<input type="radio"/> 金 虎--副教授	<input type="radio"/> 常 征--副研究员

评审主席为：魏 维--副教授

确定

图 4-1 答辩分配效果图

#### 4.2.2 教师查看研究生进度

该功能实现关键源码：(源码实现了研究生进度信息绑定)

```

private void gvw_StuListBind()
{

```

```

BLL.Student Stu = new BLL.Student();
DataSet ds = Stu.GetStuListByTchDS(Session["UserNum"].ToString());
int count = ds.Tables[0].Rows.Count;
if (count < 1)
{ lbl_Msg.Visible = true; }
else{
    lbl_Msg.Visible = false;
    ds.Tables[0].Columns.Add("s_url", typeof(string));
    ds.Tables[0].Columns.Add("s_helpCountCH", typeof(string));
    ds.Tables[0].Columns.Add("s_scoreCH", typeof(string));
    BLL.PaperScore Paper = new BLL.PaperScore();
    Model.PaperScore mp = new Model.PaperScore();
    for (int i = 0; i < count; i++){
        mp = Paper.GetModel(ds.Tables[0].Rows[i]["s_num"].ToString(),
Convert.ToInt32(ds.Tables[0].Rows[i]["s_answerquality"].ToString()));
        if (mp == null)
        { ds.Tables[0].Rows[i]["s_scoreCH"] = "无成绩"; }
        else{
            ds.Tables[0].Rows[i]["s_scoreCH"] =
mp.ps_score.ToString(); }
            ds.Tables[0].Rows[i]["s_helpCountCH"] =
ds.Tables[0].Rows[i]["s_helpCount"].ToString() + "次";
            ds.Tables[0].Rows[i]["s_url"] = "~/Image/Progress" +
ds.Tables[0].Rows[i]["s_answerquality"].ToString() + ".gif";
        }
    }
    gvw_StuList.DataSource = ds.Tables[0];
    gvw_StuList.DataBind();
}

```

该功能效果，详。见图 4-2 所示

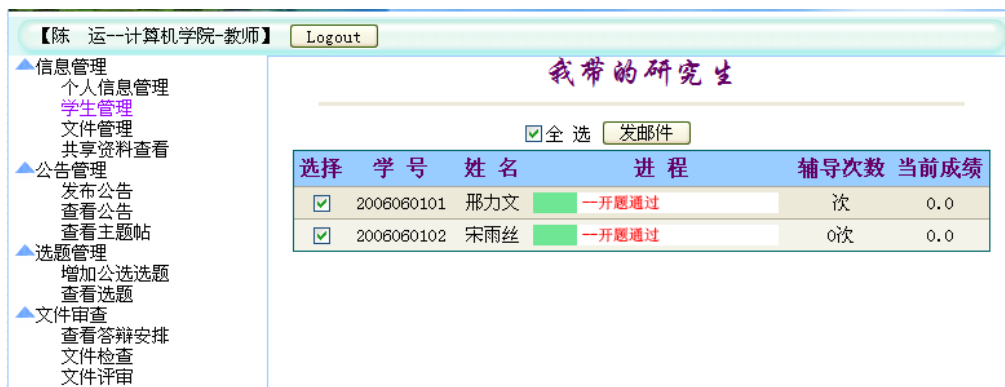


图 4-2 研究生进程效果图

## 5 系统测试

5.1 测试概述

软件测试的目标是以最少的时间和人力，系统地找出软件中潜在的各种错误和缺陷。自动化测试工具：其中有 Web 的，应用软件的，还有单元测试的。目前具有代表性常用测试软件有：Selenium, Watir, Winrunner, Loadrunner, Junit 等。

本文的系统主要运用 LoadRunner 测试工具。LoadRunner 是一种预测系统行为和性能的负载测试工具。通过以模拟上千万用户实施并发负载及实时性能监测的方式来确认和查找问题，LoadRunner 能够对整个企业架构进行测试。通过使用 LoadRunner，企业能最大限度地缩短测试时间，优化性能和加速应用系统的发布周期。

LoadRunner 工具的体系结构详见图 5-1 所示。

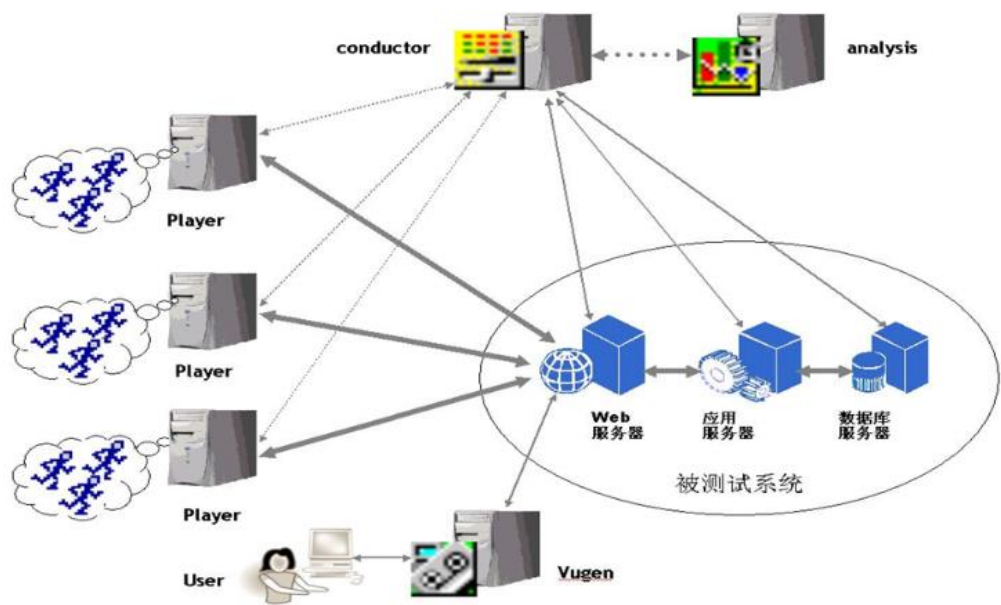


图 5-1 LoadRunner 体系结构

系统测试机器的配置环境如下：

操作系统：Microsoft windowXP Professional SP2

浏览器：Microsoft IE 8.0

CPU：AMD Athlon(tm) 64 X2 Dual Core Processor 4000+ 2.11 GHz

内存：1024M

硬盘：160G

5.2 功能测试执行情况

功能测试用力分析详见表 5-1 所示。

表 5-1 功能测试分析

序号	测试模块	测试数据			测试结果
		A	B	C	

1	登陆	用户名和密码正确	用户名或密码错误	不做任何输入	登陆成功
2	个人信息修改	基本信息格式正确，密码正确	基本信息格式错误，密码正确	基本信息正确，密码错误	更新信息成功
3	新增公告	公告标题和内容都正确	只有标题或内容中一项	全不输入	新增公告成功
4	发帖	帖子标题和内容都正确	帖子只有标题或内容中一项	全不输入	新增帖子成功
5	研究生选题	未选择题目或自拟	已经选择选题	选题并发	选题成功
6	研究生上传文档	文档格式正确	文档格式错误		上传成功
7	管理员配置系统	时间格式正确	时间格式错误		配置成功
8	管理员导入研究生、教师信息 Excel 表	表格式正确	表格式错误（文件格式错误，表中列对应错误）	没有选择任何文件	更新信息成功
9	秘书生成答辩安排表	选择了答辩方式	没有选择答辩方式	该答辩方式没有分配表	生成成功
10	教师查看学生进程	选择学生	没有选择学生	学生没有留邮箱	邮件发送成功

### 5.3 性能测试执行情况

本文的系统主要运用 LoadRunner 测试工具。LoadRunner 是一种预测系统行为和性能的负载测试工具。通过以模拟上千万用户实施并发负载及实时性能监测的方式来确认和查找问题，LoadRunner 能够对整个企业架构进行测试。通过使用 LoadRunner，企业能最大限度地缩短测试时间，优化性能和加速应用系统的发布周期。

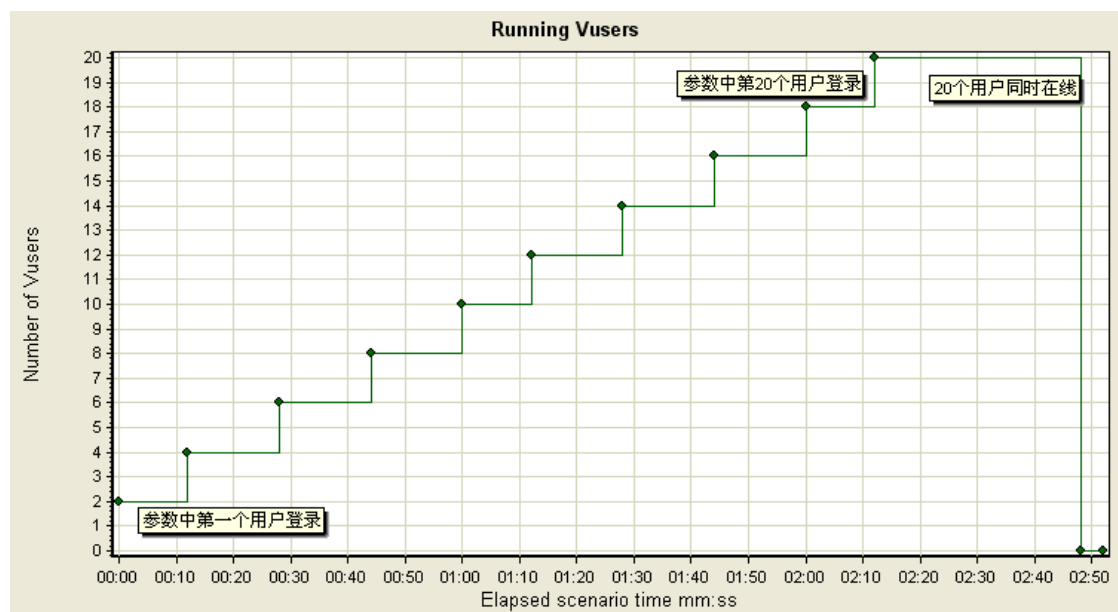


图 5-2 活动用户视图

本次性能测试，模拟了 20 个研究生虚拟用户访问系统客户端。首先录制了客户端的基本操作，包括登陆,修改信息,查看公告，发帖，回帖，查看选题，查看答辩安排，查看成绩，退出登陆等；然后设置了 100 个研究生的登陆信息参数，通过设置执行情景来实施性能测试，打开监控窗口监控系统运行状况；最后通过 LR 的分析工具分析得到结果报告。

5.3.1 活动用户视图

图 5-2 为 20 个虚拟用户访问系统客户端的运行结果，从图中可以看出有 20 个虚拟用户在同时访问系统客户端。

5.3.2 每分钟点击数

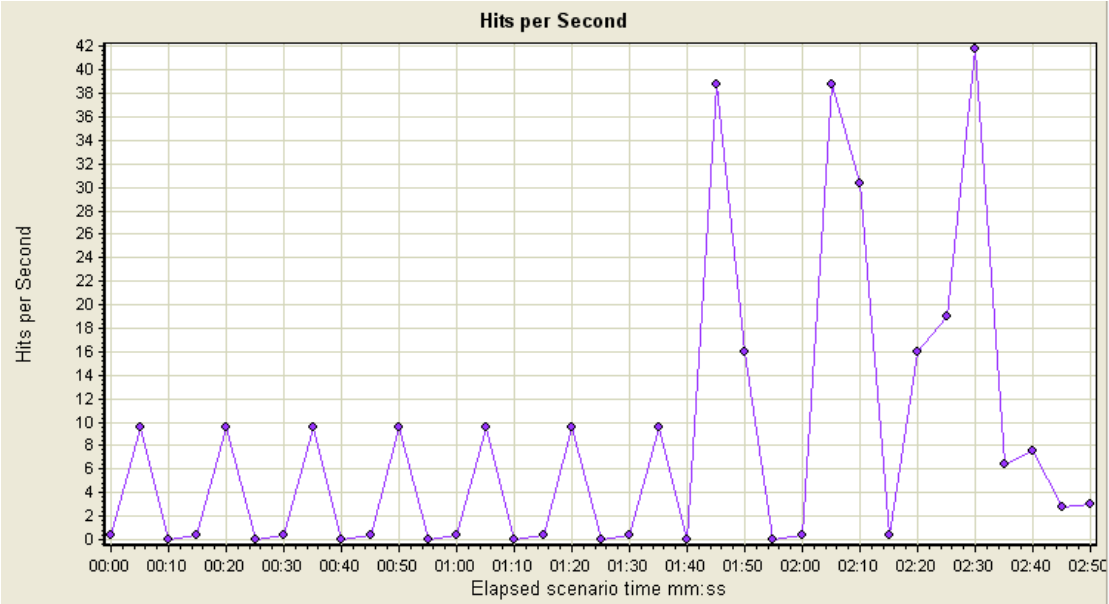


图 5-3 每分钟点击数

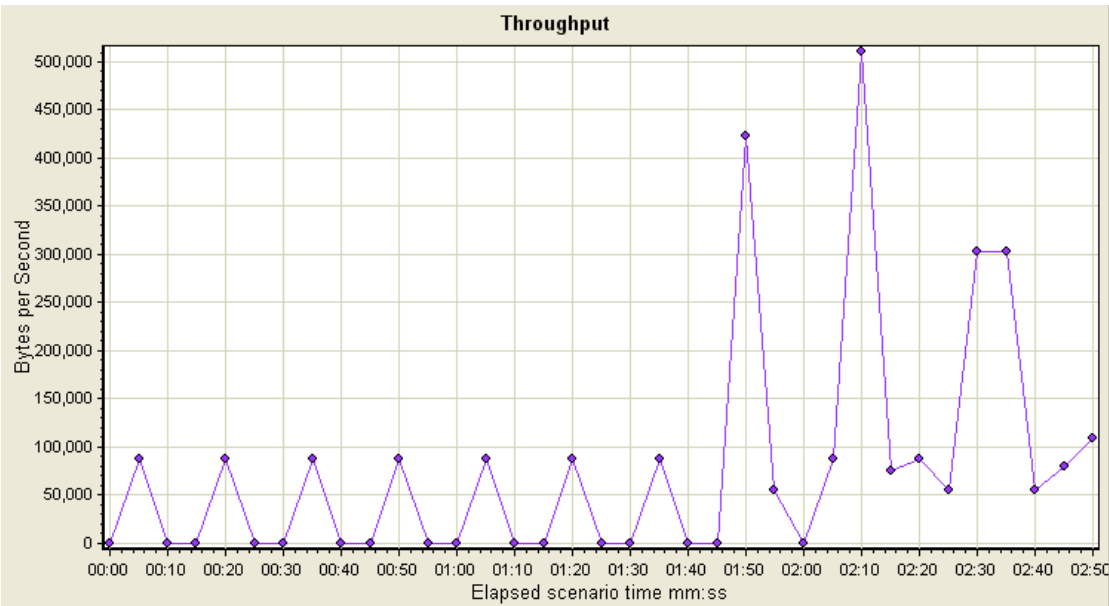


图 5-4 吞吐率

图 5-3 为 20 个用户并发访问客户端时，点击运行结果图。

### 5.3.3 吞吐量

图 5-4 为 20 个用户访问时吞吐量运行结果图，从图中可以看出，由于在提交数据时设定了集合点，吞吐量大都集中在填写完表单、提交数据的时候。

### 5.3.4 事务响应时间

图 5-5 为 20 个并发用户的设置的事务的响应时间图。其中红色为虚拟用户登录系统的响应时间，深绿色为退出系统的响应时间，紫色为虚拟用户在系统中操作时的响应时间。从图中可以看出，在修改信息提交和发帖提交页面中，反应时间最长。

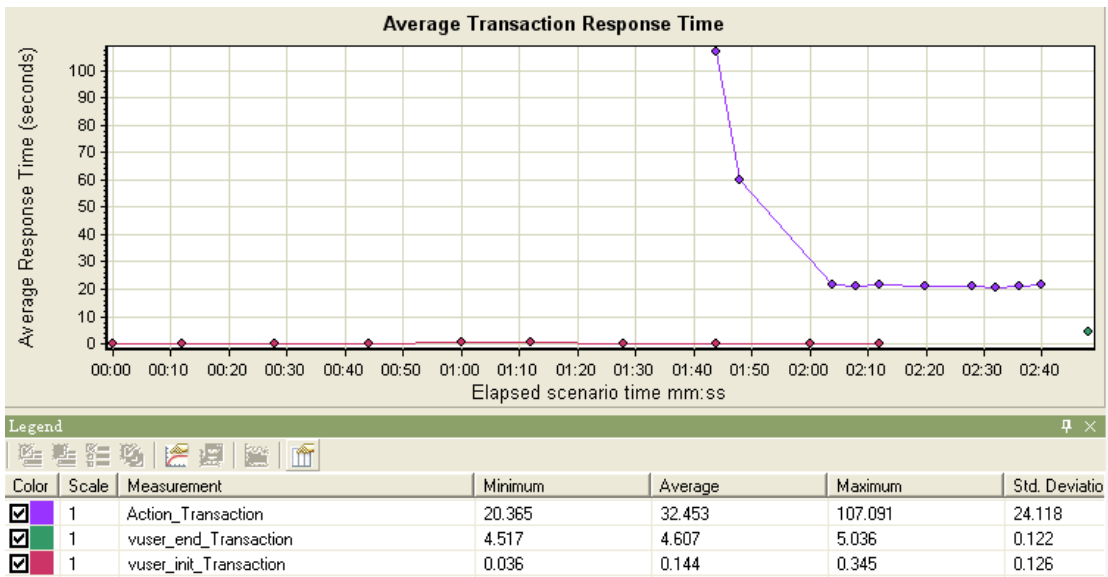


图 5-5 事务响应时间

## 5.4 性能测试分析

事务统计详见图 5-6 所示。

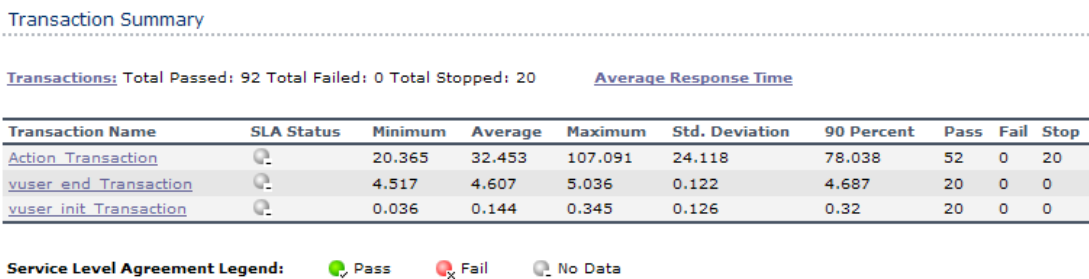


图 5-6 事务统计

综述上图分析得到结论详见表 5-2。

表 5-2 性能分析结果

并发用户数	通过交易总数	总吞吐量	总点击数	平均点击数	平均吞吐量
20	92	13481060(b)	1448(b/s)	8.419(次/s)	78378(b)

各操作下载流量详细情况见图 5-7 所示。





## 参考文献

- [1] 封旭红.研究生学位论文保障体系探究[J].天津市教科院学报,2008,(4):71-72.
- [2] 廖薇,梁晓珍,陈媛.加强过程管理,提高硕士学位论文质量[J].河北建筑科技学院学报(社科版),2005,(4):86-87.
- [3] 陈仲生,杨拥民,葛哲学,陈敏,王新峰.强化科学过程管理,提高硕士研究生学位论文质量[J].高等教育研究学报.2008,(3):38-39.
- [4] 鲁晓莹,傅德彬.MIS 解决方案[M].北京:国防工业出版社,2005.
- [5] 史景钊,王万章,陈新昌,李祥付,何玉静.基于 B / s 结构和短信平台的毕业设计管理系统[J].中国教育信息化(软件开发),2008,(12):43-45.
- [6] 刘智勇,李俊龙,梁敬东.基于.NET 的毕业设计(论文)管理系统的设计与实现[J].现代计算机(开发案例),2007,(12):121-127.
- [7] 张德勤,龚道华,杨勇.对研究生学位论文实行网上评阅的探析[J].学位与研究生教育(管理),2008,(3):34-37.
- [8] 鲍佳.基于网络的毕业设计(论文)支持服务系统的设计与实现[D].上海:华东师范大学(硕士学位论文),2005.4:29-30.
- [9] 张强,陈灿,祝湘凌.基于 B/S 模式的研究生学位管理信息系统的开发[J].科技信息(博士·专家论坛),2007:347-348.
- [10] 段文敏,向长喜,孙洪淋.基于 WebService 的高校毕业设计管理信息系统的设计[J].企业技术开发,2006,(9):19-21.
- [11] 张建臣.基于 PHP 技术的毕业设计管理系统的设计与实现[J].数据库及信息管理,2008:1210-1212.
- [12] 陈尚松,杜旭英,俞欢军.基于 Struts+Hibernate+Spring 框架的毕业设计管理系统[J].计算机工程与设计,2008,(15):4054-4057.
- [13] 崔晓红,郭景辉,壬久策.基于 Struts 的毕业设计管理系统[J].高校教育研究,2008, (8): 93-94.
- [14] 杨海.高职毕业设计质量控制系统的构建与实现[D].大连:大连理工大学(硕士学位论文),2007.5:42-45.
- [15] 丁光惠,杨伟,李晓霞.毕业设计成绩管理系统的设计与实现[J].计算机教育,2007:1750-1751.
- [16] 马革兰,涟漪.严格学位论文各环节管理·提高研究生学位论文质量[J].教书育人(管理),2007,(8):27-29.
- [17] 程萍,陈静.研究生学位论文质量过程监控系统设计与实现[J].武汉理工大学学报(信息与管理工程),2007,(3):57-60.

- [18] PowerDesigner 建模工具简介[OL].百度百科. <http://baike.baidu.com/view/1021650.htm>.
- [19] 赵生翼.基于.NET 平台的毕业设计管理系统中若干关键技术的研究与实现[J].运营探讨,2008,(3):64-68.
- [20] 廖成香.基于 WEB 的毕业设计信息管理系统设计开发[D].广东:广东工业大学(硕士学位论文),2005.12:30-32.
- [21] 侯群.基于 ASP.NET 的毕业设计管理系统开发[J].信息技术,2009,(11):104-109.

## 致 谢

本课题在选题及研究过程中得到了魏维老师的悉心指导，他为人随和热情，治学严谨细心。在闲聊中他总是能像知心朋友一样鼓励我，在论文的写作和措辞等方面他也总以“专业标准”严格要求我，从选题、定题开始，一直到最后论文的反复修改、润色，魏老师始终认真负责地给予我深刻而细致地指导，帮助我开拓研究思路，精心点拨、热忱鼓励。正是魏老师的无私帮助与热忱鼓励，我的毕业论文才能够得以顺利完成，谢谢魏老师。

在此，我还要感谢在一起愉快的度过大学生活的 415 舍友们，没有你们的帮助和支持，我很难克服一个一个的困难和疑惑，直至本文的顺利完成。在这里请接受我诚挚的谢意！

我还要感谢培养我长大含辛茹苦的父母，谢谢你们！

最后向在百忙之中评审本文的各位专家、老师表示衷心的感谢！

作者简介：

姓 名：姜永强

性别：男

出生年月：1986 年 10 月 16 日

民族：汉

E-mail: null79@qq.com

## 声 明

本论文的工作是 2009 年 11 月至 2010 年 6 月在成都信息工程学院计算机学院完成的。文中除了特别加以标注地方外，不包含他人已经发表或撰写过的研究成果，也不包含为获得成都信息工程学院或其他教学机构的学位或证书而使用过的材料。除非另有说明，本文的工作是原始性工作。

关于学位论文使用权和研究成果知识产权的说明：

本人完全了解成都信息工程学院有关保管使用学位论文的规定，其中包括：

- (1) 学校有权保管并向有关部门递交学位论文的原件与复印件。
- (2) 学校可以采用影印、缩印或其他复制方式保存学位论文。
- (3) 学校可以学术交流为目的复制、赠送和交换学位论文。
- (4) 学校可允许学位论文被查阅或借阅。
- (5) 学校可以公布学位论文的全部或部分内容(保密学位论文在解密后遵守此规定)。

除非另有科研合同和其他法律文书的制约，本论文的科研成果属于成都信息工程学院。特此声明！

作者签名：

2009 年 06 月     日