

山东大学

硕士学位论文

基于J2EE构架技术的高校网络办公系统的设计与实现

姓名：董敏

申请学位级别：硕士

专业：软件工程

指导教师：刘宏

20080410

摘 要

随着 Internet/Intranet 的快速发展和应用,推进高校办公的自动化、网络化以及电子化已是大势所趋。目前,以传统客户/服务器模型设计的办公自动化系统已不能很好地满足网络办公的需求,也难于融合新的技术以充分发挥新管理模式的优势。基于此原因,本论文目的在于通过设计与实施一套高校网络办公自动化系统,总结探索出一种基于 J2EE 的高校办公自动化系统的设计方法与实现思路。

本文来源于基于 J2EE 应用技术的办公自动化系统的软件开发项目。它以“潍坊职业学院办公自动化系统”第一版开发项目为背景,通过对用户需求的具体分析,阐述了如何把基于 J2EE 的 MVC 模式的设计思想应用于开发平台的体系设计中来,在此基础上采用“控制、业务和显示”三种逻辑分开的设计模式,设计了一个适用于国内典型高校网络办公自动化业务的开发平台,并使用 Java 语言实现了服务器端程序。

本系统设计思想和方案突出实用性,实现了档案管理自动化和办公事务处理自动化,以提高办公效率和管理水平;实现了高校各部门日常业务工作的规范化、电子化、标准化,以增强各部门信息的可管理性;实现了信息的在线查询、借阅,以最终实现“无纸”办公。

本论文完成的主要工作如下:

1、对高校原有系统进行了分析总结,找出了需要改进和扩充的地方,在此基础上对新系统进行了总体规划与设计:包括功能改进需求、业务逻辑、系统功能模块设计以及数据库分析。

2、编程实现了系统八大模块的全部功能。在系统功能设计和用户接口设计上,使后台业务规则的修改与前台用户接口的修改实现了分离,保证了系统设计上的可扩展性。

3、综合运用各种软件开发技术,提高通用性和移植性,如压缩与解压缩、数据上传和下载等;应用模板思想建立一些业务功能的参照范例,如录入、查询、报名等;抽象和开发出一些具有通用性的功能模块,如用户授权模块、信息发布

模块等。

4、清晰地定义模块之间的接口关系和各模块的功能，并逐模块加以实现。

关键词：J2EE、高校、网络办公、自动化系统

ABSTRACT

With the rapid development and widespread application of internet/intranet, promoting the college office automation, networking and electronic has been a general trend. At present, the office automation system designed by conventional Clie n/Server mode could not well satisfied the need of network office and could not fully fuse the feature of new techniques, so it could not bring the advantage of new management mode into well play. According to these situations, with the design of a network office automation system for universities, this paper summarizes out an office automation system based on J2EE.

This article comes out of the development of a J2EE-based office automation system project. The OA (Office Automation) system studied in this paper was based on “The Network Office Automation System of Weifang Vocational College” of version 1st. Following the analysis of the special needs of OA users, how to use J2EE-based MVC model in the design of development platform is discussed. Then, the design mode which logically separates application control, business logic and display is used to design a typical domestic network of office automation business development platform. Finally, server-sided programs are developed using Java.

Applicability is implicit in the program design philosophy and scheme. The main purposes of this system is as follows: firstly, ealizing the automation of file management and office business to improve efficiency and optimize management, secondly, realizing the formalization, electornicness and standardization of daily operations in various departments to enhance the manageability of related information, and finally, realizing online query to achieve the goal of “paperless” officing.

The main work accomplished and the results obtained in this thesis are as follows:

1. The existing systems of colleges and universities were analyzed to identify the needs for the overall planning and design of new system: including demands for functional improvement, business logic, system design, function module and database analysis.

2. All eight function modules are realized in this system. With system function design and user interface design, the separation design mode is adopted to ensure the separation between the revision of the back-office rule and front-office user interface.

This also ensures the scalability of the system design.

3. Various software development technologies, such as compression and decompression, data upload and download, etc. are used integrately to guarantee the system generality and migration ability. Reference instances are established for some business functions, e.g., entry, query, application, etc.. Some general function modules are abstracted and developed, e.g., user authorization module、information dissemination module, etc..

4. The interfaces between modules and module functions are defined clearly and developed seperately.

Keywords: J2EE, Colleges and Universities, Network Office, Automation System

原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的科研成果。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律责任由本人承担。

论文作者签名：董敏 日期：2008年4月10日

关于学位论文使用授权的声明

本人完全了解山东大学有关保留、使用学位论文的规定，同意学校保留或向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅；本人授权山东大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或其他复制手段保存论文和汇编本学位论文。

(保密论文在解密后应遵守此规定)

论文作者签名：董敏 导师签名：刘宏 日期：2008年4月10日

第 1 章 绪论

1.1 课题研究背景、目的和意义

当前,许多单位的办公系统仍然采用传统的人工操作方式。随着数据量的日益增加,单位内部往来数据的处理常常需要耗费很大的人力及成本,效率低下,因此,如何应用现代化手段提高工作效率和管理水平成了一个亟待解决的问题。近年来人们普遍使用计算机来提高个人工作效率,但是在需要协同工作的现代工作环境中,我们更需要提高单位整体的工作效率。为此,我们需要利用网络通讯基础及先进的网络应用平台,建立一个安全、可靠、开放、高效的信息网络和办公自动化系统(OA, Office Automation),为管理部门提供现代化的日常办公条件及丰富的综合信息服务,实现办公管理自动化和办公事务处理自动化,以提高办公效率和管理水平;实现各部门日常业务工作的规范化、电子化、标准化,增强业务的可管理性;实现信息的在线查询、借阅,最终实现“无纸”办公。如何利用先进的计算机技术来编制网络办公系统,并在较短的时间内,方便、快捷地开发出适合于管理的网络办公系统就成了当前的一个重要任务。

办公自动化系统应是一个与本单位业务紧密结合,可以按单位的实际需要而定制的系统。换句话说,信息采集、查询、统计等功能应与单位的实际需要密切相关,操作人员只须点击一个按钮就可以得到想要的结果,从而极大地方便单位内部管理工作的开展和决策的制定。

办公自动化系统的应用环境应该比较宽松,既可以安装在局域网内,也可以在互联网上应用。比如,系统可以采用浏览器/服务器(Browser/Server, B/S)结构,只要在服务器端安装本软件,客户端不用安装其他软件,就可以通过浏览器来访问和使用。

办公自动化系统还是一个与世界联系的渠道,本单位的 Intranet 网络可以与 Internet 相联。一方面,老师和学生可以在 Internet 上查找有关的资料、科研论文、就业信息等,加强与外界的联系;另一方面,也可以通过网上办公系统让更多的考生、用人单位来了解我们,建立校友联络机制,从而起到宣传介绍的作用,这种网上交流的潜力将非常巨大。有效的利用网络办公系统,必将会提高工作效率、降低成本,从而提高高校的整体实力,为高校的发展做出巨大贡献。

1.2 国内外发展现状

美国曾制定了《政府纸张消除法案》，2003 年 10 月前已实现政府办公的无纸化作业，使美国公民与政府之间的互动关系电子化；英国已经有 40% 的政府服务可以通过互联网提供给公众，到 2004 年，已有 41% 的英国公众主要通过政府机构的网站享受政府服务；日本政府于 2000 年 3 月向国会提出了《电子签名与认证法案》，提出在 2003 年实现电子政务，完全采用电子化手段处理各种行政事务，日本将信息化的成败、IT 产业的兴衰与经济竞争力的强弱紧密联系，订下了信息化立国的目标，力争在五年内发展成为世界上最先进的 IT 国家；在 2004 年，加拿大 40% 公众通过政府机构的网站享受政府服务；同年，澳大利亚 34% 公众也通过政府机构的网站享受政府服务；新加坡电子政府可为公民提供 200 项以上的电子政务服务，公众可从“电子公民中心”站点轻松获取医疗保健、商务、法律法规、交通、家庭、住房、就业等各项网上信息和服务；韩国认为，21 世纪是知识和信息决定国际竞争力的时代。

在国外，欧美及日本等国在办公自动化领域的研究起步较早，通过政府部门长期努力，已基本实现了办公自动化，并正朝着无纸化办公的方向发展。以其发展过程来看，早期的系统主要也是定向开发的专用系统，并投入了巨大的人力和财力才逐渐完善和发展起来。

在国内，办公自动化经历了十多年的发展历程，部分单位的应用也取得了长足的进步，采用计算机辅助决策已经被人们普遍接受。随着高校信息化的发展和成熟，目前很多高校的 OA 系统已经整合了越来越多的应用，集成了人、财、物、信息、知识等诸多高校资源，逐步成为一个员工办公、中层管理、领导决策的知识协同平台，OA 这棵“老树”开始绽放“新花”，脱胎换骨，重新焕发出新的光彩。因此我们需要用发展的眼光来看待这种新的发展趋势。国内计算机产业界也一直致力于 OA 系统方面的研究和开发，同时也推出了一批 OA 系统投入应用。但是目前的各类软件由于缺乏统一的规划及标准，特别是受到应用平台的局限，难以普遍推广使用，因此开发适合我国办公自动化特点的跨平台办公自动化通用软件产品，并做好推广应用工作，已成为必然趋势，这也是办公自动化发展的根本方向。

中国要想与世界接轨，电子政务是一个必由之路，通过网络行政，不仅可以

实现与发达国家之间的远程办公、远程交流，还能通过电子政务带动信息化，进而带动工业化。对于一些发达城市，如北京、上海，在发展中对电子政务的运用非常好，借鉴了电子政务实行的较为完善的系统方案，将成本最小化、效益最大化。系统实现技术上，回顾我国政务信息化的进程，大致可分为三个阶段：

1、以桌面字处理工具为典型个人办公工具软件的阶段，计算机的应用提高了个人工作效率。

2、基于关系型数据库技术，以客户机/服务器（Client/Server，C/S）体系结构应用为特征的阶段，这一阶段基本实现了部门级的数据处理、公文处理等的自动化。

3、基于符合 Internet/Intranet 技术标准的平台应用阶段，这一阶段，不仅在技术上有了很大进步，而且应用范围已从部门内部、部门之间扩展到行业/系统内部，乃至跨部委跨系统。

在 Internet 应用方面，政府部门已陆续建立了自己的 Web 服务器和网站，有的还实现了一些简单的网上业务。但由于各相关单位在信息化建设中的技术差异和经验欠缺，电子政务并未在真正意义上得以实现，相反，不少政府站点在规划建设、管理维护以及安全性方面还停留在较低的水平上，各级政府部门还有待建立和完善从页面制作、审批、更新以及流程管理、应用开发、安全性保证等一整套行之有效的管理模式、运营方法和技术平台。办公自动化(OA)方面，自 90 年代中期以来，随着 Internet/Intranet 的发展，OA 技术的内涵及外延都在发生改变。尤其是群件产品的兴起，使实现办公自动化系统有了可靠的技术基础和产品基础，办公人员可以处理更为广泛的业务和办公信息。现代意义上的基于网络的 OA 已经不仅仅是提高像打字录入、编辑排版等个人事务处理效率，也提高了像公文收发、流转、签发、归档这样的群组作业效率。

事实上，网络时代的 OA 是一种跨部门运作的基础信息系统，不仅可以连结高校各个岗位上的各个工作人员，还连结着高校各类信息系统和信息资源的运行，不仅能够适应单位为提高运作效率而进行的机构调整，而且还可能促进资源合理配置，甚至可以改革机构运作的流程。也就是说，OA 在模拟手工作业的同时，还将反作用于传统的办公流程。

1.3 论文的主要内容

1、对高校原有系统进行了分析总结，找出了需要改进和扩充的地方，进而对新系统进行了总体规划与设计：包括功能改进需求、业务逻辑、系统功能模块设计以及数据库分析。

2、综合运用了计算机网络技术、网络办公系统设计思想和软件工程方法，确定了实现系统的技术方案和开发环境。

3、编程实现了系统八大模块的全部功能。在系统功能设计和用户接口设计上,使后台业务规则的修改与前台用户接口的修改实现了分离,保证了系统设计上的可扩展性。

第 2 章 高校网络办公系统的需求分析

2.1 系统调查

2.1.1 办公系统应用整体分析

全球性的网络化、信息化进程正改变着人们的生活方式，Internet 技术应用以及电子商务的飞速发展给人们的生活和工作的各个层面带来了深刻的影响。回顾我国办公信息化的进程，大致可分为三个阶段：（1）以桌面字处理工具为典型的个人办公工具软件阶段，计算机应用提高了个人工作效率。（2）基于关系型数据库技术，以 C/S 体系结构应用为特征阶段，这一阶段基本实现了部门级的数据处理、公文处理等的自动化。（3）基于符合 Internet/Intranet 技术标准的平台应用阶段，这一阶段，不仅在技术上有了很大进步，而且应用范围已从部门内部、部门之间扩展到行业/系统内部，乃至跨部门跨系统。

在 Internet 应用方面，单位部门已陆续建立了自己的 Web 服务器和网站，有的还实现了一些简单的网上业务。但由于各相关单位在信息化建设中的技术差异和经验欠缺，网络办公系统并未在真正意义上得以实现，相反，不少系统在规划建设、管理维护以及安全性方面还停留在较低的水平上，各级部门还有待建立和完善从页面制作、审批、更新到流程管理、应用开发、安全性保证等客户支持的一整套行之有效的管理模式、运营方法和技术平台。

2.1.2 主要办公的组织机构

高校一般下设人事处、教务处、资产处、科研处、团委、工会、财务处等部门。其中教务处是整个高校的核心部门，负责全院的课程设置、教学信息的发布、教学评估工作，人事处负责人才招聘、培训、考勤等管理。资产处负责学院设备的采购管理，科研处负责学院的科研，各系负责教学和学生管理工作。

2.2 问题定义

经过认真的调研，笔者认为网络时代用户关心的办公问题主要有以下几个方面：

- 1、机构规模扩大，分支机构增多使信息量迅速膨胀，但同时各分支机构却

成为信息孤岛，原有的办公平台已无法适应规模化的办公需求；

2、计算机网络健全，但离线用户的信息仍无法及时、快捷、方便的到达；

3、大中型机构规模的信息化建设需要继续扩大设备数量，所带来的维护、升级等的工作量太大；

4、同一体系的办公单位相对分布较远，相互之间的工作协同发生失误可能性大；

5、人员流动性大、信息的非结构化、信息难以安全和妥善管理、信息利用率低，制约着机构管理的发展；

6、公文编辑及流程太过繁琐，原有的办公平台无法实现公文、印章、手写批注一体化；

7、现有的办公系统在 IE 环境下文字处理能力差，无法满足办公需要；

8、金字塔型管理模式上下信息难以通畅，管理出现脱节。

面对这些问题，我们首先要将具体问题抽象成合理的数学模型，然后再把它转化到具体模块来解决，当然这个过程会遇到一些具体问题，比如怎么建立数据库，怎么连接数据库，以及所建立的数据库是怎么管理数据的等问题。系统各个模块都要尽量简化操作过程，降低操作难度，并尽可能多的给工作人员提供提示信息，以提供友好的操作界面。同时，系统还要对用户的误操作、错误的输入数据，给出适当的提示信息，使系统能继续稳定的运行。

2.3 需求分析

2.3.1 性能需求

高校网络办公系统是利用先进的 J2EE 技术，使办公室工作逐步物化于各种现代化设备当中，由办公室人员与设备共同构成服务于某种目标的人机信息处理系统。办公自动化系统两个最为重要的思想就是政务流程管理和信息集成。政务流程管理不但需要涵盖高校机构内部的政务工作流程，还需要包括本单位的上下级管理机构的相关流程，应具有以下几个性能：

从安全性而言，系统涉及到的数据大部分为我国政务、财政、金融等关键数据，因此要求更高的安全性。

从扩展性和标准化而言，系统对标准接口的要求非常严格，国家专门对电子

政务建设针对网络办公系统中的公文格式规范、设计与管理规范、网络基础设施部分、支撑技术、工程管理等多个方面做出规定，提出了明确的要求。

从易用性而言，目前，高校工作人员的计算机信息技术的培训相对较少，部分地方高校工作人员很少使用或者甚至没有使用过计算机，要想在高校内部实现办公自动化，就必须要求系统操作简洁、通俗易懂，便于多数人接受。因此这种情况对于系统提供商也提出了较高的要求。

2.3.2 开发平台需求

基于 J2EE 的高校网络办公系统的开发平台应具有以下的特点：

1、步骤标准而简洁。通过编写少量的程序代码即可定制新业务和调整现有业务内容、流程等。

2、开放性。开发平台应提供基于可扩展标记语言标准的数据接口，与其它应用系统之间支持可定义、自解析的数据交换。

3、灵活性。开发过程具有跨平台化、参数化、构件化、标准化。定制各应用系统均只需定制模板、操作集、角色等，具有高度的灵活性，便于调整原业务和定制新业务，可以满足并带动业务不断发展，满足业务发展的需求。本系统能通过参数控制不同处理过程的组合，不同组合的过程实现不同的功能。

4、安全性。应具有完善的多级授权措施。能对使用权限和系统运行中各个环节进行监督确认，系统在处理过程中设有完善的交易日志及特殊处理日志以检验处理的可靠性，能够检测和发现系统的故障差错，提供系统错误和故障提示等容错机制，系统通过事务、校验等技术来保证交易、数据的一致性。

5、系统的保密性。开发平台对操作员实行分角色管理，不同角色，可访问不同的系统、功能和数据，并进行不同的操作。数据库中存放数据时，开发平台采用了数字签名校验、加密等技术防止数据库中的关键数据被非法修改。

6、快速性。开发平台采用共享内存、常用文件常开不关、循环优化、类人工神经元、多用户单连接、事务排他性等技术来提高系统处理效率。

7、易维护性。考虑了业务的多样性，以适应各种业务情况。

8、成熟性。应具有成功的应用实例。

2.3.3 系统业务要求

2.3.3.1 系统总体需求功能图：

高校（潍坊职业学院）下设人事处、教务处、资产处、科研处、团委、工会、财务处、后勤等部门。每个部门的工作范围与内容都不一样，总结他们各个部处的日常事务，公文流转程序等需求，可以得到以下高校（潍坊职业学院）办公自动化系统总体需求功能图 2.1。

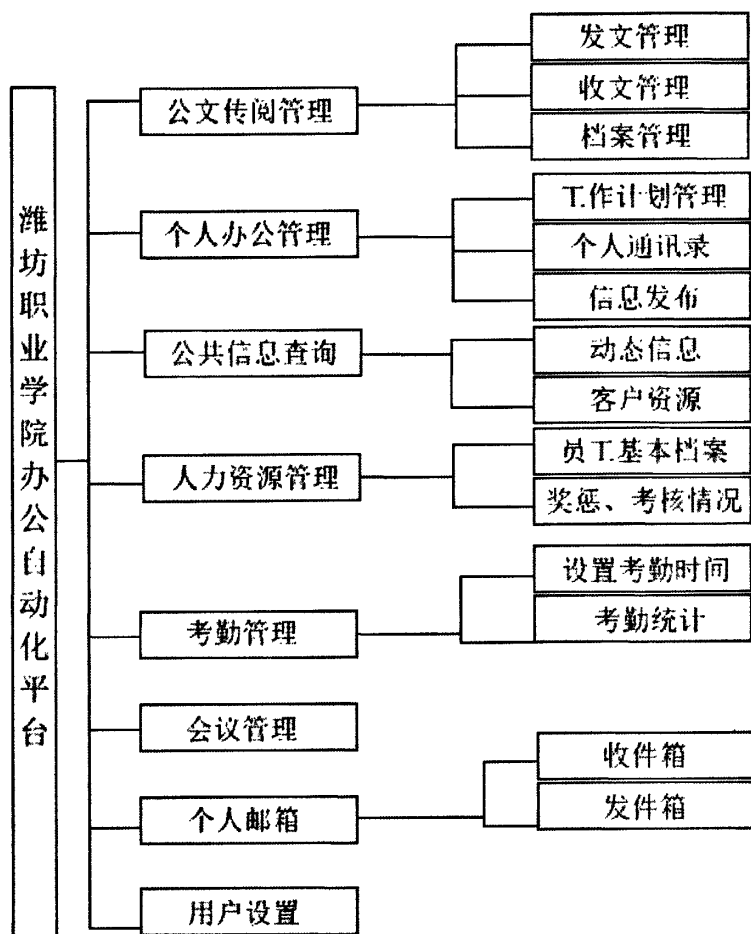


图 2.1 潍坊职业学院办公自动化系统总体需求功能图

Figure 2.1 Overall function requirements for the office automation system of Weifang Vocational College

2.3.3.2 系统功能模块详述：

1、公文传阅管理

公文管理不是传统意义上的静态文档管理，它不仅包括静态文档资料的管理

还包括流转审批，办理完毕后直接归档到静态文档管理系统中进行保存。它分为三个部分，一是发文管理、二是收文管理、三是档案管理。无论是收文管理还是发文管理，办结后都直接归档到档案管理中保存。而其中的收文管理及发文管理过程，就是一个公文流转的过程。

1) 发文管理

发文管理模块将实现拟稿、核稿、会签、批示、电子签名、办理、退稿、提交归档、发布、打印等业务操作。为保证发文处理的保密性，一方面不同业务人员应具备不同的权限，如阅读、修改、销毁、发布等权限需要合理地分配；另一方面，在发文处理的不同业务阶段，即使是同一人，其权限也在不断地发生变化，系统需要根据业务人员的职责和发文处理的业务阶段，恰当地处理这种情况。如发文经总裁审批后，拟稿人便失去对所发文稿的修改权限；又如未完成处理的发文，不允许销毁、公开，只有办理完毕后，才能执行上述操作。

一个大致的发文流程可以总结如图 2.2:

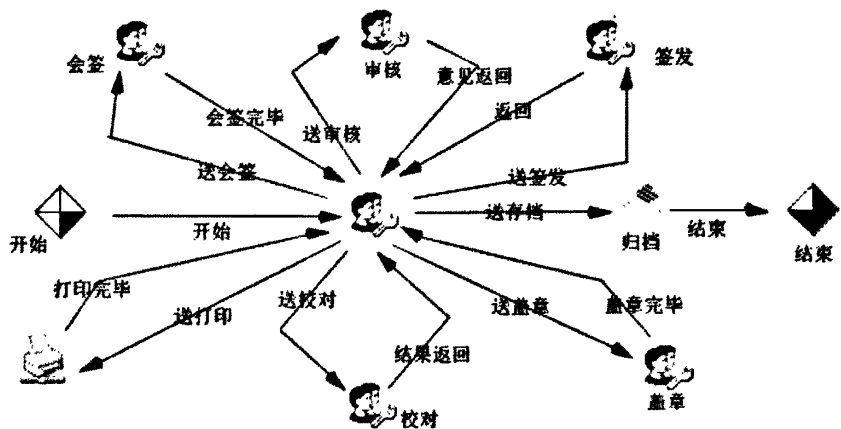


图 2.2 发文流程图

Figure 2.2 Flow chart of document dispatching

2) 收文管理

收文管理要完成来文登记、拟办、审批、承办、办结等全过程的管理。系统提供了收文的方式，即磁盘调入，标准格式的电子公文调入，扫描和直接输入后的公文可以送相关领导阅读或批示，批示流程可以根据需要自行定义，批示结束后，公文将被发送给最终承办人阅示，由承办人去办理相关事务，结束后公文将被送交文件管理部门进行归档。一个大致的收文流程如图 2.3 所示：

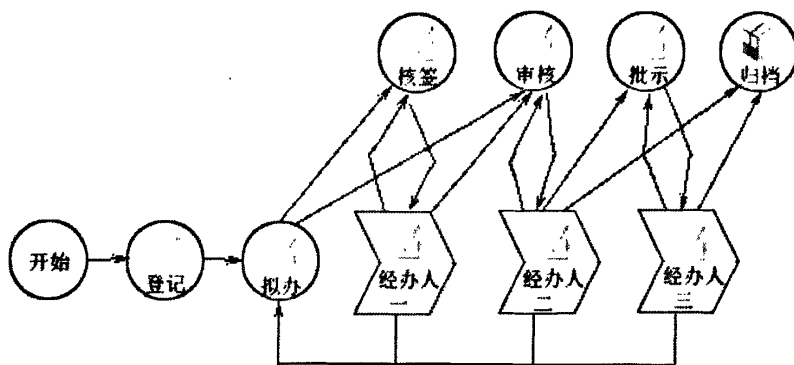


图 2.3 收文流程图

Figure 2.3 Flow chart of document receipt

3) 档案管理

针对历史数据，可以把条目信息直接导入到系统中进行保存；对于刚形成的数据，可以通过直接归档的方式归档到档案分类中进行保管，也可以直接录入到系统中进行保管。对于原件信息，电子件可以直接上载到系统中进行保存；系统应提供非常灵活的查询方式可以快捷的查询到所需文件；对于档案中的各种索引目标，系统应提供报表功能，方便的实现索引目录的制作。

2、 个人办公管理：

个人办公管理主要服务对象是个人，每个人可以利用系统提供的功能进行个人事务的管理工作。个人事务主要有以下内容：

1) 工作计划管理：

系统用户都可以把个人计划上载到 OA 系统中进行保管，并可以进行查询。通过权限分配，可以把工作计划开放给其它人查看。

2) 个人通讯录：

系统用户都可以把个人通讯录上载到 OA 系统中进行保管，便于查询。

3) 信息发布：

信息发布包括信息发布栏目的设置，信息发布过程的管理、信息发布方式的管理、信息发布范围的限定及发布信息的清理等内容。

3、 公共信息查询

公共信息查询模块主要为办公人员提供电子公告板、校办信息、规章制度、焦点服务，实现办公人员之间的信息交流与共享。

4、 人力资源管理

人力资源管理模块主要为了满足学校管理员工的基本档案,记载员工职位变动,记录员工奖惩、考核、工资档案等情况,方便教职工更好的了解自身工资的变动情况。系统中每一注册的人员为一个用户,建立他的相关信息时,将通过他的角色、职务和部门的设定,确定他在系统中的管理岗位与操作权限。同时通过其他信息的定义,方便他在系统中的工作。

5、 考勤管理

考勤管理模块功能主要为了满足学校日常管理的需要,对教职工的出勤情况进行汇总统计。包括:设置考勤时间、开始考勤、今日考勤统计、日考勤统计、月考勤统计。

6、 会议管理

会议管理模块提供会议计划、准备、记录、查询的功能。在会议召开前进行会议准备,内容包括合理地安排会议的参加人员、时间、场地、内容议题,以电子邮件或打印通知单的方式发放会议通知等。对已召开的会议可对出席情况、议题讨论结果、会议决议等内容作记录并整理会议纪要。同时还可以对所有会议室的规模、设备服务配置和使用时间安排进行管理。

7、 个人邮箱

个人邮箱模块提供了发邮件、收件箱、发件箱、废件箱等功能

8、 用户设置

功能包括:增加下属用户、管理下属用户、工作计划权限、通告公文权限、奖惩编辑权限、考核编辑权限、资源管理权限、考勤管理权限等等。

第 3 章 高校网络办公系统的设计

3.1 系统的设计思想和设计目标

3.1.1 设计思想

1、运用先进、主流的 Internet 网络和 B/S 结构，进行平台应用开发，实现突破时间和空间限制的应用，并以 Web 主页作为本地与远程信息交互的桥梁。

2、利用数据库技术来规范化数据文件和非规范文档文件，通过 J2EE 技术以数据库形式进行存储，更便于检索、分类和归档。

3、采用平台思想组织系统的设计开发，力求建立一种应用模式和平台。清晰处理模块之间的接口关系和模块功能，使之能够分块实现，具有可扩充和扩展性。

4、综合运用软件开发技术，提高通用性和移植性。如压缩与解压缩、数据上传和下载等，应用模板思想建立一些业务功能的参照范例，如录入、查询、报名等，抽象和开发出一些具有通用性的功能模块，如用户授权模块、信息发布模块等。

本着以人为本的开发思想，建立操作方便、友好的操作界面。通过对目标对象群体的需求分析，提出并建立网络办公自动化系统，建立和完善从页面制作、审批、更新、以及流程管理、应用开发、安全性保证等一整套行之有效的管理模式、运营方法和技术平台。集成 Internet 及局域网技术、数据库技术、安全保密技术、办公自动化技术等稳定成熟的技术，构建既可以面向现代化应用，又具有可扩展性和可扩充性的办公自动化系统应用平台。

3.1.2 设计目标

1、统一的界面和平台

本系统针对内部办公业务要求和与外部联系的需求，运用 Internet 技术，以 WEB 作为解决方案的核心，充分实现信息的共享和交流，为办公自动化和网络化提供技术支持。系统建立在 Internet 构架下，采用 B/S 模式，客户端通过浏览器完成系统内部的信息处理和信息查询。

2、实现移动和异地办公

系统基于 WEB 设计,可以随时随地的使用该信息系统处理各类事务,用户所需要的只是一台计算机和一个浏览器,即使出差在外,也可以通过 Internet 使用本系统,便捷地与同事交流信息,实现远程办公。

3、功能丰富

本系统可提供常用的内部办公功能。信息采集和处理、查询功能等等。

4、强大的安全机制

多层的安全控制机制确保只有经过授权的用户才能访问系统资源。安全日志和模块日志详细记录了用户的使用情况,并采用网络防火墙技术,使系统信息更为安全可靠。系统在操作权限许可内可细分为多个级别,由管理人员进行统一设置。用户只允许访问被授权的数据和功能菜单。

5、灵活完善的系统架构

系统以 web 服务器为核心,集成文件服务器和数据库,使系统结构灵活,易于扩展,能满足将来的需要。

3.2 系统架构

3.2.1 系统架构

基于J2EE的高校网络办公系统总体框架结构如图3.1所示,本系统由高校校园网和Internet网组成,办公系统外网的接入方式包括:Internet、手机、PDA等。校园内网提供给本地用户所使用。用户与内网系统的交互通过统一交互平台(高校资源门户网站和内网办公桌面)完成。统一身份认证体系处于统一交互平台和内网核心部分之间,用户通过用户统一交互平台,必须通过统一的身份认证系统确认该用户的合法身份才能与内网核心部分进行交互。内网核心部分由政务办公自动化应用系统(也就是本文所论述的开发平台所支撑开发的系统)中的决策支持、会议汇报、内容管理、检索、数据采集以及导入等部分组成,各系统之间均有互调接口和数据交换接口。

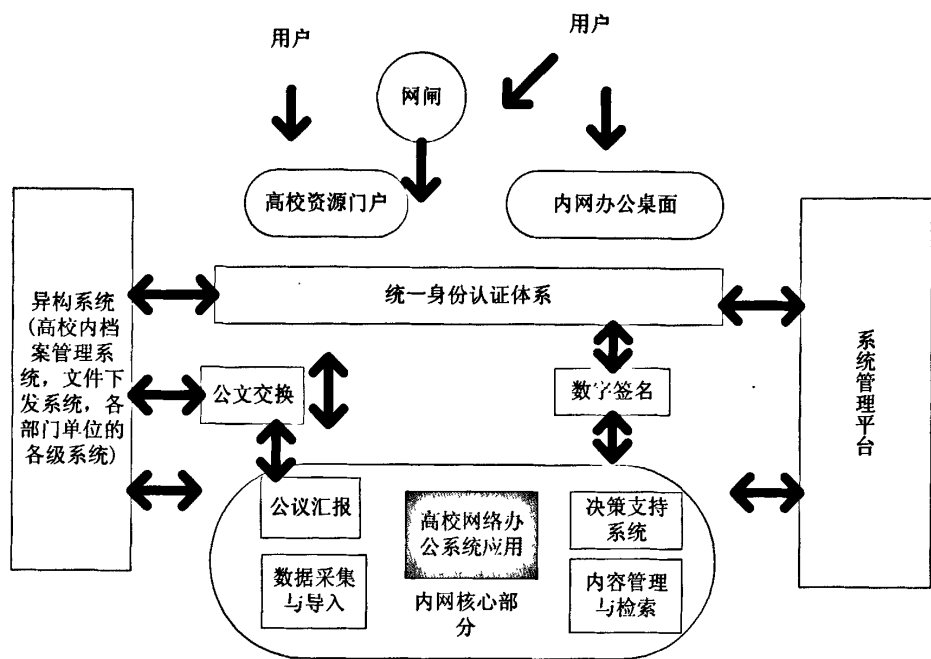


图3.1 基于J2EE的高校OA系统总体功能框架结构

Figure 3.1 Overall function framework of university OA system based on J2EE

3.2.2 开发平台构架

图3.2是一个高校OA系统开发平台整体结构图。笔者所设计实现的开发平台系统的定位是用于开发此应用体系中适用于政务办公自动化的WEB应用服务的支撑平台。

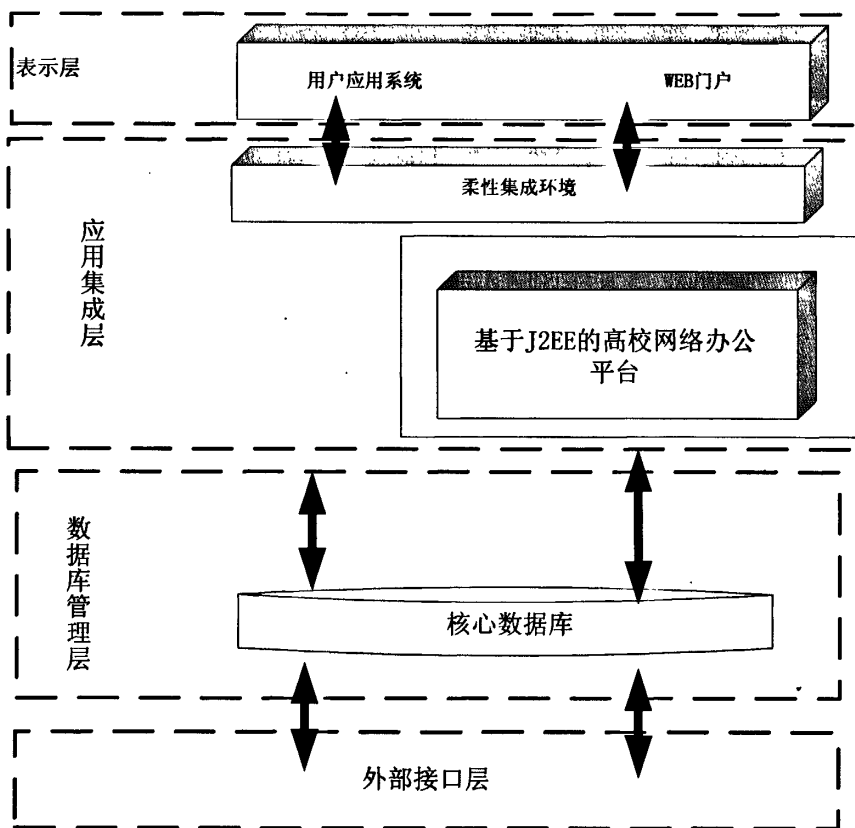


图3.2 高校OA系统开发平台整体结构图

Figure 3.2 Overall structure for the development platform of universities OA system

由图可知，系统由信息交换层、数据管理层、平台管理层、应用集成层和表现层五个层面。

1、 信息交换层

主要负责外部数据的注入与交换；为系统提供可靠的数据来源。以数据交换接口为表现形式，包括：外部格式数据交换接口和数据自动交换（信息邮差）接口，分别完成已知格式数据的自动加载和自定义格式数据的交换。

2、 数据管理层

该层是整个平台体系的数据资源管理中心与集散地，负责政务业务过程数据与结构数据的存取管理、系统使用的数据模型的管理、用户信息与权限的管理、元数据的存取优化管理、系统使用的各类应用工具方法管理以及系统运行的相关知识的管理。该层通过ODBC数据接口实现数据的访问，是支持多数据库平台的基础。平台采用自主知识产权的组件模块实现了对数据库的优化访问。元数据的管理为平台的灵活性提供了强有力的支持，它分为四个层面：用户应用集成环境元

数据、平台系统角色管理元数据、数据库操作元数据、界面集成元数据。

3、 平台管理层

该层将平台的各种组件、模块集中管理，协调工作，将模型、工具、元数据、模板、对象等组件有机的集成在一起，形成统一的应用开发平台，它是政务业务的集成与抽象。主要模型包括：数据库表注册模型、元数据模型、业务操作管理模型、用户角色模型、数据维护管理模型等。

该层实现的模板管理将输入输出界面图形化，使界面元素完全自定义，从而达到所见即所得的效果，满足了应用需求不断变化的实际情况。该层系统还集成了大量的工具，比如图形图像处理、邮件收发、Microsoft Office操作、公式计算、查询生成等。

4、 集成层

该层结合某具体政务业务的实际，将平台层管理的各类组件与模块同用户、角色、操作集管理结合起来，配以基于PKI的安全管理体系，从而形成面向实际应用的业务系统。

该层在将上述模块集成的过程中，使用了柔性集成的方式，即将各类模型组件、工具方法、业务过程统一定义，并通过UDDI等方式进行登记集成，最后根据应用需求进行科学的拼装。

用户角色管理实现了角色到办公实体的角色映射，以授权为核心，面向 workflow 和面向角色管理，实现了用户到角色，角色到数据库表，角色到数据库表字段，角色到 workflow 段的映射。角色网络管理的内涵如下：

管理一般是基于组织层次的，管理可能是多方面的，包括行业、市场等，政务信息系统也不例外。各级管理人员之间千丝万缕的关系构成了基本的角色网络。政务系统中的 workflow，也就是角色整体中的一些联系个体，由此将层次结构按照角色扁平化、网络化后分析可知，其实质上就是由角色作为顶点，各个 workflow 为边构成的网络，即角色网络。网络中每个角色都是节点，角色之间的关系构成网络的边。从数学意义上说，workflow 实质上是网络中边的组合数，它量上的变化是按照组合级数来增长的，不管 workflow 如何先进，角色也要处理太多 workflow，而且 workflow 本身也有许多不确定性，因此本平台不提供 workflow 的管理，这是本平台最大的特点。一个机构里的节点数是有限的，从节点出发，只管角色，每个角色管好了，无限的 workflow 就被组织起来了，这是本平台在理论上的突破。

信息流动是怎样实现的,某一个角色在业务系统中能进行什么样的工作是相对固定的,信息到达某角色的时候,此角色知道对此信息应进行什么样的处理,因此只需将此角色能处理的信息特点描述清楚即可,这是符合业务管理实际的。

5、表现层

为应用用户提供交互的环境,并可根据用户的个人喜好进行展现方式的个性化定制,真正实现应用需求主导。

3.3 核心层设计

控制、业务和显示三者分层实现,采用柔性集成技术实现对政务流程的控制是这个框架的核心技术。控制层接收用户请求后,首先将用户的请求数据封装成一个表单对象,检索“模型注册表”,来得到需要运行的业务逻辑层的组件信息,并把封装好的表单对象传递给被选中执行任务业务逻辑层的组件(模型)。业务逻辑层的组件对表单对象处理完毕后会将其交回给控制器,控制器继续将表单对象传递给视图显示控制器中相应的视图来完成显示任务。在上述的处理过程中,对象中的方法负责数据处理,而 HTML 页面由页面方法和模板文件共同负责生成。模板文件只负责数据的显示。模板文件中不但包含了一些页面的显示方式,而且包含一些动态代码,调用视图类中定义的属性和方法来获得动态数据。模型包括系统控制类和业务逻辑控制类,框架提供规范的接口来把这两类模型发布到“模型注册表”,屏蔽掉核心层的技术实现。

3.3.1 整体控制结构设计

本系统的整体控制结构设计遵循 MVC 模型的设计思想,其控制结构及原理可以用图 3.3 来描述。

其中,模型存储应用程序的数据,包含应用程序的状态及其它业务数据,视图负责显示模型中存储的数据,控制层接收用户请求后,检索“模型注册表”,调用业务逻辑模型来处理业务。业务逻辑层的组件做两件工作:一是处理请求并修改模型中的数据,二是把需要显示的处理结果数据写回到表单对象中,然后会将其交回给控制器。控制器得到处理后的表单数据对象,根据前面接收的客户端请求数据特征,检索“视图注册表”,调用一个视图,视图根据模型中的数据生

成 HTML，页面返回给浏览器。在上述的处理过程中，业务逻辑层模型中的方法负责数据处理，而 HTML 页面由页面方法和模板文件共同负责生成。模板处理方法只包含名称，可用于处理一个外部的模板文件。模板文件中包含了一些页面的显示方式和一些动态代码，这两者都以本系统自主设计的一套标签库为全集。控制器分析模板上标签后调用系统类库中定义的属性和方法来获得动态数据。

模型层由系统控制模型和业务逻辑模型组成。系统控制模型主要包括：人员组织管理，工作流程的支撑平台，权限控制，并发控制。其中系统控制模型与控制器相配合，对整个框架的控制机制起支撑作用，业务逻辑模型是封装好的对象，具有业务接口和方法，每一个新的应用都可以使用这些接口和方法生成业务逻辑模型的实例。业务逻辑模型代表了系统的功能组成。业务逻辑层是独立的功能模块，具有很好的可移植性和扩展性，主要包括收文管理、公文审批、痕迹保留、发文管理等。

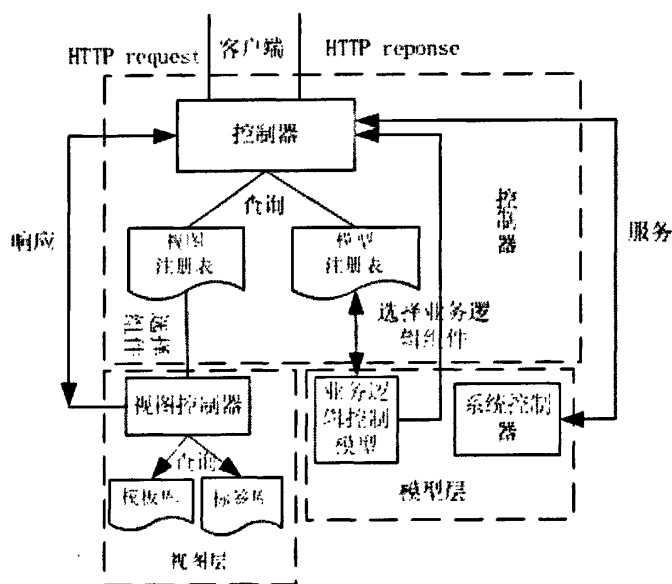


图 3.3 系统的 MVC 结构

Figure 3.3 MVC structure of the system

办公自动化模型由多个模型组成，如收文管理、发文管理、公文审批、痕迹保留等。办公自动化模型及其子模型均向“模型注册表”注册以备控制器查询。通过办公自动化模型的子模型管理配置接口可以为具体的应用配置使用哪些子模块来构成办公自动化的流程、子模块参数和子模块间的关联参数。收文管理子模型有设置收文单位接口，设置收件人接口，设置收文流程接口；公文审批有送

下一审批人接口，返回拟稿人接口，在审批流程中插入审批人接口等。用来实现这些接口的应用 J2EE 接口和方法生成办公自动化模块的实例，供控制器来调用。

3.3.2 功能结构

本文设计并实现的办公自动化开发平台分为七个功能部分。

- 1、系统控制模型：系统控制模型主要包括：人员组织管理，工作流程的支撑平台，权限控制，并发控制。其中系统控制模型与控制器相配合，对整个框架的控制机制起支撑作用；
- 2、业务逻辑模型（内部）：业务逻辑模型是封装好的对象，具有业务接口和方法，每一个新的应用都可以使用这些接口和方法生成业务逻辑模型的实例。本文实现了最典型的办公自动化中的公文管理系统所使用到的四个业务逻辑模型。
- 3、与内部业务逻辑模型的接口：是系统控制模型与业务逻辑模型（内部）之间的接口。
- 4、与外部业务逻辑模型的接口：是系统控制模型与业务逻辑模型（外部）之间的接口。
- 5、视图模型（内部）：四种最常用的视图显示方式的实现。
- 6、与内部视图模型的接口：是系统控制模型与内部视图模型之间的接口。
- 7、与外部视图模型的接口：是系统控制模型与外部视图模型之间的接口。

这七部分的关系可以用系统的功能结构框图（图3.4）

来表示。

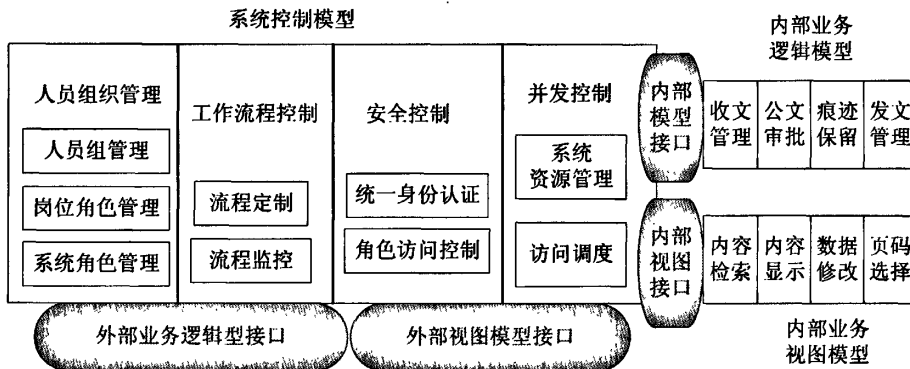


图3.4 系统功能结构框图

Figure 3.4 Function structure of the system

3.4 系统对象设计

3.4.1 模型

模型部分由系统控制模型和业务逻辑模型组成。控制模型主要包括：人员组织管理、工作流程的支撑平台、权限控制、并发控制。业务逻辑模型代表了系统的功能组成。业务逻辑层是独立的功能模块，具有很好的可移植性和扩展性，主要包括：收文管理、公文审批、痕迹保留、发文管理等。

3.4.1.1 系统控制模型

系统控制模型与controller配合，是控制整个框架的关键，下面简要介绍一下系统控制模型的设计思路。

1、 人员组织管理模块

人员组织管理模块的作用是管理高校机关的组织 and 人员信息。组织人事系统的管理功能包括：人员管理，组织管理，岗位角色管理和系统角色管理。人员管理将人员置于不同的组织下进行管理，组织可根据自己的实际情况来建立自己的人员组织结构。组织管理的功能包括管理系统里存在的各种组织以及管理组织之间的关系。组织之间的关系包括管理与被管理关系，协作关系等。岗位角色管理是管理岗位与人员、岗位与组织的关系，而系统角色管理管理的是岗位角色与系统角色以及系统角色与权限之间的关系。岗位角色的作用与高校实际管理中的岗位相类似，而系统角色在系统中表现为权限的集合，一个岗位角色可以被分配以多种系统角色，一个系统角色也可以被分配给多个岗位角色。岗位角色包括两类：一种是固定角色，和实际中存在的岗位可以相对应来建立，如部门主任，他被赋予相应的权限。另一种是临时角色，这种角色代表一定权限的集合，可以用在岗位权限转移时临时组合权限生成。用户登录时，在验证了用户的身份成功以后，系统将通过角色关联器来提取用户的岗位角色并将其转换成系统角色，系统将根据登录用户所拥有的不同的系统角色集合来生成不同的用户界面。

2、 公文流转控制模型设计

流程的创建过程

A. 通过对业务信息的分析，设计用户角色：从业务信息中提取出用户和角色的信息，包括用户和角色的全集及其映射关系。

B. 高校信息中提取出基本操作单元来设计显示模板。

C. 业务信息, 从中提取出用作定义和建立角色网络体系中的操作全集和相互的依赖关系, 并由这些操作组成政务处理过程中业务逻辑模型。

D. 最后组合用户角色系统、模板系统和操作集全集, 结合权限控制系统, 形成最后角色网络模型, 即包含了业务流程的全集。

3、 权限控制

A. 数据库读写控制

a、登录: 用户身份验证中心验证用户的身份有三种方式: 用户名 / 密码方式、UKey方式和CA验证中心。用户通过验证后, 控制器根据操作管理器、模板管理器以及角色关联器三者的关系提取用户的可操作全集, 存储在一个XML数据文件里, 再同视图控制器把文件中的数据“绘”在HTML模板上, 展示给用户。

b、模板的数据库访问: 在高校工作中, 对访问数据库资源的安全控制起着举足轻重的作用。在本系统中安全控制是由角色数据库管理表、角色数据库属性管理表、角色数据库字段管理表和角色数据库字段属性管理表四者相配合控制的。角色数据库管理表存储着角色对数据库中的某个表是否有访问权限; 角色数据库属性管理表用于存储角色在对数据库表的操作中所具有的属性, 如只读、修改、删除等; 角色数据库字段管理表用来存储角色在数据库中对在某个字段是否有访问权限; 角色数据库字段属性管理表用来存储角色在数据库中对字段操作权限的属性, 如可修改, 只读等。当用户要对数据库里的数据进行读写时, 系统的权限控制模块会根据以上四个表中相关的内容来判断用户要做的动作是否在系统对权限的安全控制允许范围之内。对内容信息的上传和访问权限均支持用户权限管理机制, 支持不同的用户和用户组, 也支持不同的用户权限和不同的用户级别。支持对内容按照密级进行分类。具有高密级的文件会被打上数字签名, 用数据加密方式进行传输, 同样, 具有被授权人的数字签名才能访问该文件。

B. 应用开发管理员模型

系统管理员权限的设计模式是系统非常重要的一部分。本框架设有一个总管理员和若干分组管理员, 各部分管理资源的操作原则是: 总管理员建立的资源低级管理员不能删除和修改, 只能对总管理员允许此人操作的部分资源进行分配、使用, 低级管理员只能对自己建立的管理资源删除、修改、分配、使用。目前只对低级管理员开放角色、岗位, 岗位和角色对应、为角色分配操作集、自建字典表, 总管理员可以建立分组分级管理员, 赋予他们不同权限。

a、分级管理员可分配权限资源描述

a) 角色资源

i. 能浏览总管理员允许其看到的角色

ii. 能建立本单位内使用的角色

iii. 为自建的岗位和从组织人事系统中的岗位分配角色, 包括a)和b)

中描述的角色

b) 岗位资源

i. 能浏览总管理员允许其看到的岗位

ii. 能建立本单位内使用的岗位

iii. 为自建的岗位和从组织人事系统中的岗位分配角色, 包括a)和b)

中描述的岗位

c) 岗位和角色对应关系资源

如 1.C和2.C所述

d) 为角色分配操作集的权限资源

可以为角色分配已有的操作集, 其中角色包括1.a和1.b中的角色, 操作集是总管理员允许此管理员操作的角色

e) 自建字典表资源

i. 能查看总管理员建立的字典表

ii. 能建立自定义的字典表

iii. 能为角色或岗位分配1.a和1.b中的字典表

b、分级管理员的数据库模型

对管理员权限的控制由八个表支撑完成: 分级管理员操作集总表、分级管理员操作管理主表、分级管理员角色操作管理表、分级管理员用户操作集管理表、分级管理员用户角色管理表、分级管理员用户单位内部岗位管理表、分级管理员用户单位内部岗位角色对应管理表、分级管理员用户单位内部字典表。表的作用简述如下:

a) 分级管理员操作集总表

i. 表中内容由总管理员分配给该管理员的操作集组成:

ii. 低级管理员不能修改此表的内容

b) 分级管理员操作管理主表

i. 表中内容由总管理员分配给该管理员可以操作的操作集全集的一部

分组成;

ii. 如果该管理员有自定义操作集的权限, 则可以在此表中添加记录, 否则不能。

iii. 如果该管理员有修改总管理员定义的操作集的权限, 则可以向服务器发请求修改操作管理主表中记录, 否则不能;

c) 分级管理员角色操作管理表

i. 表中内容由总管理员分配给该管理员可以操作的操作集全集的一部分组成。如果该管理员有自定义角色个性化操作集的权限, 则可以在此表中添加记录, 否则不能。

ii. 如果该管理员有修改总管理员定义的角色个性化操作集的权限, 则可以向服务器发请求修改主表中记录, 否则不能;

d) 分级管理员用户操作集管理表

i. 表中内容由总管理员分配给该管理员可以浏览的角色组成

ii. 如果该管理员有自定义角色的权限, 则可以在此表中添加记录, 否则不能。

iii. 如果该管理员有修改总管理员定义的角色权限, 则可以向服务器发请求修改主表中记录; 否则不能;

e) 分级管理员用户角色管理表

i. 表中内容由总管理员分配给该管理员可以浏览的角色组成

ii. 如果该管理员有自定义角色的权限, 则可以在此表中添加记录, 否则不能。

iii. 如果该管理员有修改总管理员定义的角色权限, 则可以向服务器发请求修改主表中记录; 否则不能;

f) 分级管理员用户单位内部岗位管理表

i. 表中内容由总管理员分配给该管理员可以浏览的岗位组成

ii. 如果该管理员有自定义岗位的权限, 则可以在此表中添加记录, 否则不能

iii. 如果该管理员有修改总管理员定义的岗位权限, 则可以向服务器发请求修改主表中记录, 否则不能;

g) 分级管理员用户单位内部岗位角色对应管理表

i. 表中内容由总管理员分配给该管理员可以浏览的岗位。角色对应关系组成

ii. 如果该管理员有自定义岗位角色对应关系的权限，则可以在此表中添加记录，否则不能；

iii. 如果该管理员有修改总管理员定义的岗位角色对应关系的权限，则可以向服务器发请求修改主表中记录，否则不能。

h) 分级管理员用户单位内部字典表

i. 表中内容由总管理员分配给该管理员可以浏览的字典记录关系组成

ii. 如果该管理员有自定义字典记录的权限，则可以在此表中添加记录，否则不能。

iii. 如果该管理员有修改总管理员定义字典记录的权限，则可以向服务器发请求修改主表中记录，否则不能；

i) 分组管理员对系统的操作权限的表示模型

分组管理员对系统的操作权限可以用一个操作类型符号来表示，本系统的操作类型符号中可以用表3.5来表达，表格中的每一个操作类型权限块可组合分配给新产生的分级管理员，从而满足权限分配的灵活性。

对主表操作权限 分级管理员操作资源	增	删	改	自定义
操作集总表	Operate1	Operate2	Operate3	Operate4
操作管理主表	Czglb1	Czglb2	Czglb3	Czglb4
操作管理表	Jsczgl1	Jsczgl2	Jsczgl3	Jsczgl4
用户操作集管理表	Yhoperaet1	Yhoperaet2	Yhoperaet3	Yhoperaet4
用户角色管理表	Ysgl1	Ysgl2	Ysgl3	Ysgl4
用户单位内部岗位管理表	Dwjgb1	Dwjgb2	Dwjgb3	Dwjgb4
用户单位内部岗位-角色对应管理表	PRMapping1	PRMapping2	PRMapping3	PRMapping4
用户单位内部字典表	Dictionary1	Dictionary2	Dictionary3	Dictionary4

表3.5 操作类型符号矩阵

Table 3.5 Symbol matrix of different operation types

4、 并发控制

同时访问电子政务软件的用户可能达到数百个，因此，服务器程序处理并发访问的能力非常重要。以一份公文的审批过程为例，同时可能送科长甲和科长乙审批，在审批顺序上没有并没有先后要求，假设两个科长的审批动作会修改数据库的同一个字段，于是存在一种可能，科长甲和科长乙在同一时段内打开了这份公文进行审查，而科长乙填写审批意见后先行科长甲进行了提交，此后科长甲的提交就可能覆盖了科长乙的审批意见数据，造成数据库的“脏”数据，影响高校正常的工作。在本系统中对于处理并发采用的是资源锁定机制：在应用的全局会话中设置一个UserLockedResources对象变量，当一个用户对一个修改类型的表单进行访问时，并发控制模块就会将用户已读取的字段登记在UserLockedResources对象中，这样当其它用户同样有访问修改类型的表单时，并发控制模块会到UserLockedResources对象中查询用户请求的表单上是否有字面正在被其它用户锁定，如果存在的话那么，此用户看到的HTML表单的相应字段上就会被打上“已经锁定”的提示标识。当此字段的修改权限被其它用户释放后，并发控制模块会发送一个“可以修改”的消息到正在等待的用户的客户端，提示用户并刷新用户界面。并发控制模型可以用图3.6来描述。

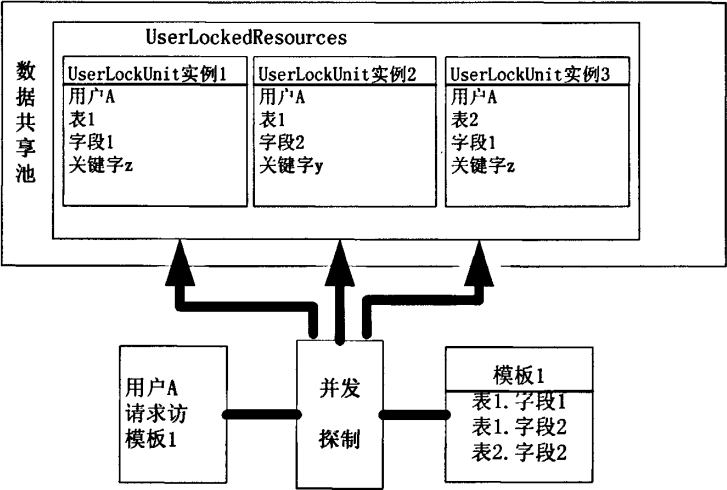


图3.6 并发控制模型

Figure 3.6 Concurrency control model

5、 数据库连接池

对于J2EE程序来讲，即使在某一较短的时间段内，其对数据库的操作请求数也远远不是一两次，而是数十、上百次，在这种情况下，系统开销是相当大的。

所以合理的安排用户对数据库的操作请求可以大大提高应用的运行效率。在本框架中我们采用自主开发的数据库连接池来解决数据库连接问题。当程序中需要建立数据库连接时，只须从数据库连接池中取一个来使用而不用新建。同样，使用完毕后，只需放回池中即可。而连接的建立、断开都有连接池自身来管理。同时，我们还可以通过设置连接池的参数来控制连接池中的连接数、每个连接的最大使用次数等等。通过使用连接池，将大大提高程序效率，同时，我们可以通过其自身的管理机制来监视数据库连接的数量、使用情况。数据库连接池工作过程可以用图3.7来描述。

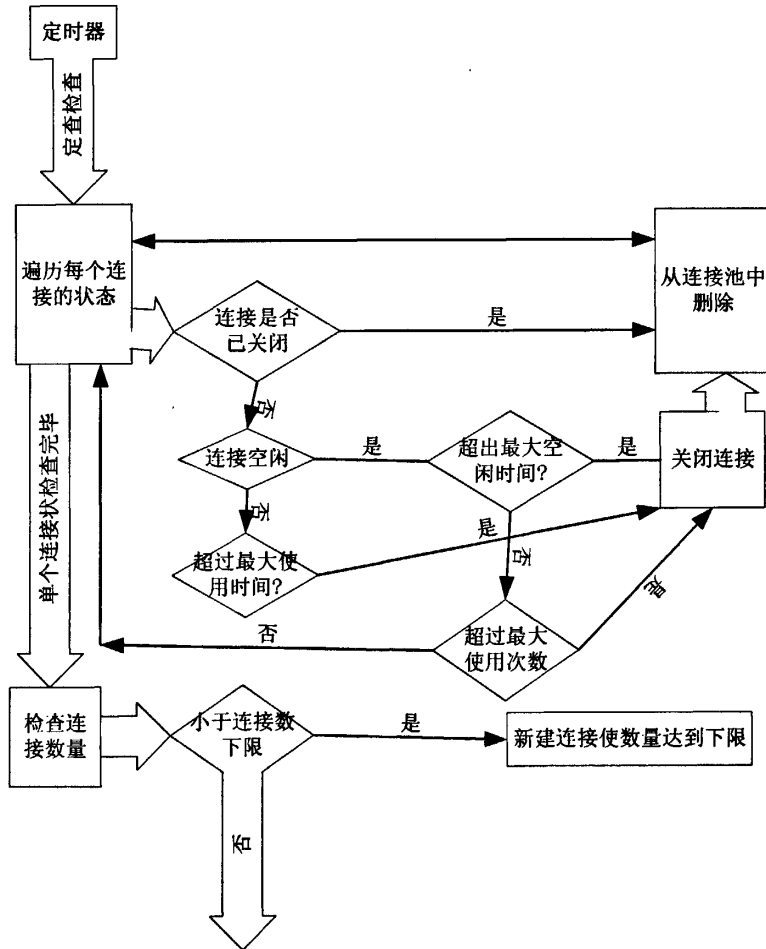


图 3.7 数据库连接池工作过程

Figure 3.7 Working process of the database connection pool

6、数据库适配器

高校网络办公自动化系统J2EE应用可能架构在各种厂家、各种类型的数据库

上,这就要求开发这样应用的框架也能支持多种厂家和多种类型的数据库。本系统开发了一个数据库适配器,安插在框架的控制逻辑与底层数据库操作之间,作为系统与数据库的“翻译”。这样做的好处有两点,一是框架逻辑不再有数据库相关性,可以提高系统的稳定性;二是基于本框架的二次开发的工作也不依赖固定的数据库类型,在业务体系相同的前提下可以全部平滑移植我们支持的其它类型的数据库之上,从而向二次开发人员屏蔽掉一定的数据库开发的异质性,提高了二次开发的效率和可重用性,目前支持的数据库有Microsoft SQL Seve2000、Oracle8i、Oracle9i。

3.4.1.2 业务逻辑模型

办公自动化模型由多个子模型组成,如收文管理、发文管理、公文审批、痕迹保留等。办公自动化模型及其子模型均向“模型注册表”注册以备控制器查询,见图3.8。通过办公自动化模型的子模型管理配置接口可以为具体的应用配置使用哪些子模块来构成办公自动化的流程、子模块参数和子模块间的关联参数。收文管理子模型包括设置收文单位接口、设置收件人接口、设置收文流程接口;公文审批包括送下一审批人接口、返回拟稿人接口及审批流程中插入审批人接口等。用来实现这些接口的是一些EJB类。新的应用可以使用这些接口和方法生成办公自动化模块的实例,供控制器来调用。

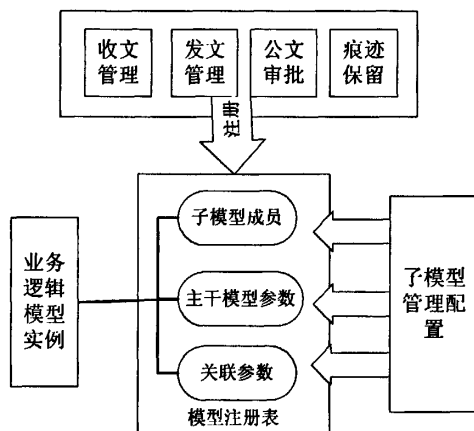


图 3.8 业务逻辑模型注册

Figure 3.8 Registration of business logic model

3.4.2 视图 (View)

利用本框架进行了二次开发的一个重要的开发工作就是制作模板。基本模板

是就一个HTML文件，通过载入标签库的标签来利用系统的增加记录、修改记录、删除记录以及组合查找等功能。与每一个功能相对应都有一个标签。本框架的服务器程序的视图控制部分负责解析模板上的标签，从控制器接收显示相关的数据，按照自定义标签库的规则来替换模板上的标签，生成HTML文件，返回客户端。

3.4.2.1 模板元素组织形式

用户界面的形式是一些带有本系统定义标签的HTML页面，我们称之为模板。本系统中使用组合模式的概念对模板页面进行分割。将页面分割若干个单元，多个单元中包含一些页面元素；同一个页面若有多个表单Form，每个表单应属于不同的单元。页面分割举例如图3.9。

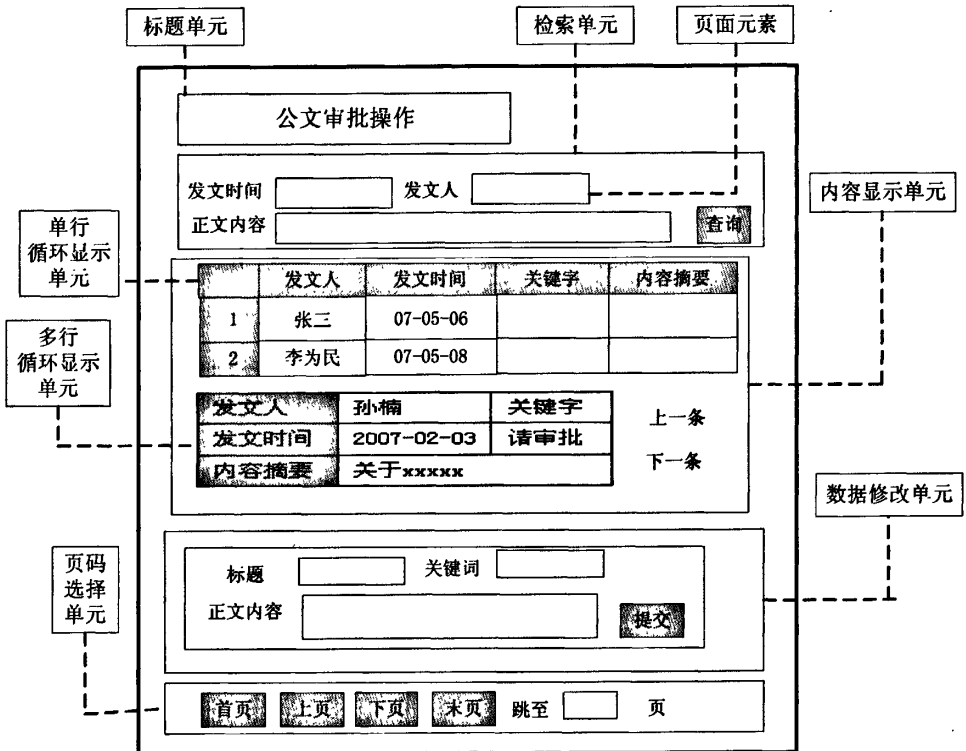


图3.9 模板页面上的分割单元示例图

Figure 3.9 Illustration of segmentation unit on the model page

对应于以上的页面的分割，控制器会将其处理成一组视图数据对象，即每个分割单元都会根据分类对象化，以树结构存储在系统中。根结点存储模板级信息；根结点下挂Form结点，对应模板上的表单；Form结点下挂页面元素，包括input框（text类型、hidden类型、checkbox类型和select类型）。目前此视图的结构只

支持三层，即分割单元之间只能进行三层以下的嵌套。

3.4.2.2 模板标签库

一般的J2EE系统开发平台都使用JSP来生成视图。例如Struts应用中的View部分是通过JSP技术实现的。Struts提供了自定义的标记库可以使用，通过这些自定义标记可以非常好的和系统的Model部分交互，通过使用这些自定义标记创建的JSP表单，可以实现和Model部分中的ActionForm的映射，完成对用户数据的封装，同时这些自定义标记还提供了像模板定制等多种显示功能。采用JSP可以嵌入逻辑，但是嵌入大量的java代码或是JSP Tag会使页面变得很复杂，使得可维护性和可读性变差。而且办公自动化系统应用的页面所涉及到的显示逻辑和业务逻辑可以抽象出来很多通用流程，这些流程比普通的WEB应用开发更抽象。一个完备的标签库可完全基于HTML使用，而实现页面定制“零代码”。因此为了尽量减少页面显示中的代码编写的任务量，提高页面编写的可重用度，本平台在扩展了HTML的基础上，设计了一套定制标记库。定制标记的语法类似于HTML标记，但其又能够像JSP页面一样访问运行环境和情境的所有信息，使用定制标志包装常用的功能以便在页面中复用。

本平台的标签库与Struts的taglib相比更适应高校OA系统应用的开发，更偏重于怎样与本系统的系统控制模型和业务控制模型相交互，以及快速高效的实现办公模板定制。本框架自定义标签的生效级别有两类：

- (1) 页面级，使用形式举例如下

```
<html>
...
<!GOVP定义->
operate_type:=modify;
form_type:=list;
loop_number:=3;
<-GOVP定义!>
...
</html>
```

其中<!GOVP 定义->与<-GOVP定义!>之间的代码表示本页面所具有的一些属性：operate_type:=modify; 表示本页面是一个提供给用户进行修改的页面；form_type:=list;表示本页面上的表单的显示形式是多行数据的循环显示；

loop_number:=3; 表示每页上的表单中显示的数据的行数是3行, 当数据量大于3行时要分页显示并生成上下翻页的按钮@GOVP_script1表示某一段系统定义的javascript, 用于一种处理办公自动化过程中的特殊用途, 比如痕迹保留, 此javascript完成任务的工作即调用已向系统注册的用于打开Word文档的VC控件为此word文档打开保留修改标记的功能的函数。

(2) 表单级, 使用形式举例如下

```
<form name="form1" GOVP_OperateType="modify"
GOVP_action="公文审批">
<input type="text" GOVP_DataSoure=[tableA.fieldB]>
...
</form>
```

其中GOVP_OperateType="modify"表示本表单是一个提供给用户进行修改的表单; GOVP_action="公文审批"表示本表单提交的数据是提供给“公文审批”的业务流程的; GOVP_DataSource=[tableA.fieldB]表示此处的数据是从数据库中的tableA表的fieldB字段提取出来的。字段提取出来的以上例子只是为办公自动化的视图控制机制的很小一部分, 我们还设计了更多的标签来适应高校网络办公系统的各种应用需求, 把这些控制标签插入HTML文件的过程很简单, 也可以做成可视化的操作, 在很大程度上提高了应用开发的效率和开发的手段与业务的契合度, 同时降低了应用开发的难度。

3.3.3 控制器 (Controller)

在MVC模式的一种基本应用形式中其控制器的运行过程(通过service 方法实现)比较简洁: 从操作库中获取操作, 然后调用它的perform方法, 操作的perform方法返回一个操作路径选择做为请求确定路由。另一种形式是在此基础上添加了事件处理机制, 即触发应用程序事件的MVC框架, 这种形式更好用, 因为应用程序通过对事件做出反应, 无需修改框架就可以扩展该框架的功能。这种扩展了事件处理机制的MVC模式其控制器服务方法的运行过程包括获取操作、产生事件、触发事件、执行操作、重新设置事件类型、再次触发事件、转向新的路由等多个步骤。本系统采用了第一种形式来设计控制器结构。

控制层由servlet组成。控制层servlet是用Java类和Java代码写成的, 因而适合

生成高度编程的内容。控制层servlet作为控制器，用于管理访问服务器端资源和客户机端的请求，并基于应用程序当前状态转发请求到其他Servlet或视图控制器。换句话说，控制层Servlet提供客户端导航和工作流的服务器端支持。Servlet接受用户请求后，首先将用户的请求数据封装成一个Form对象，结合“模型注册表”，来选择需要运行的业务逻辑层的组件，并把封装好的表单对象传递给被选中执行任务的模型。业务逻辑层的组件对表单对象处理完毕后会将其交回给控制器，控制器继续将表单对象传递给视图显示控制器中相应的视图来完成显示任务。Controller控制器工作的顺序如图3.10所示。

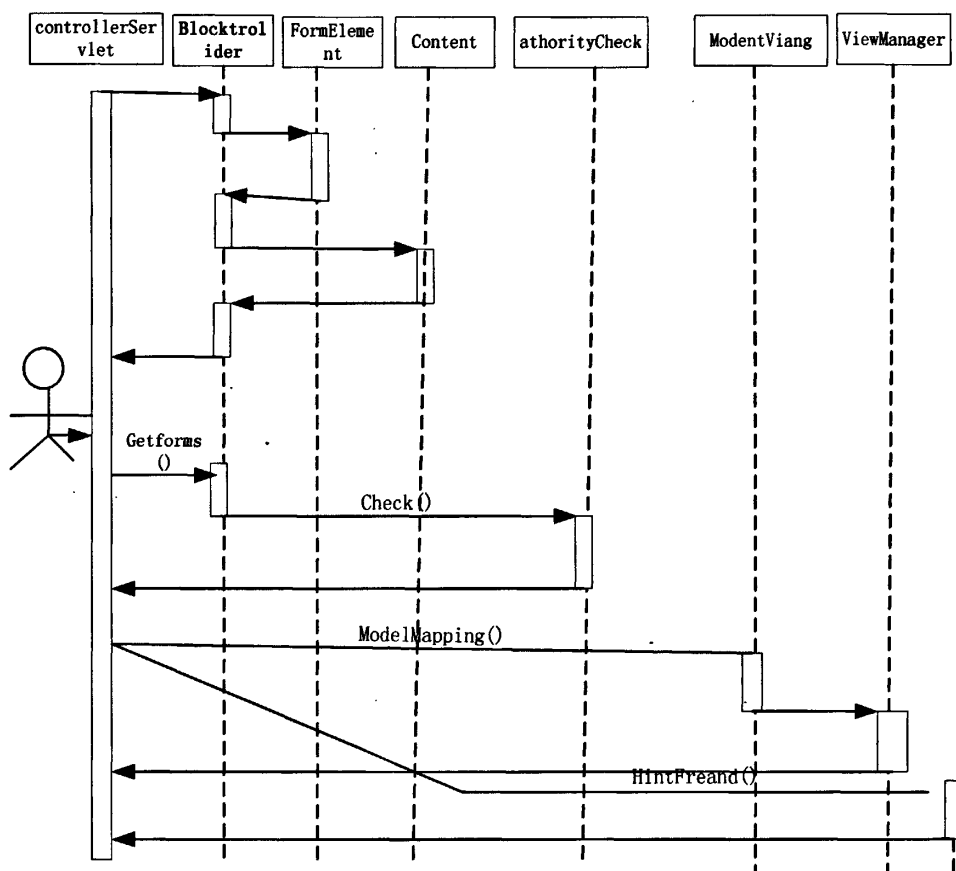


图3.10 Controller控制器工作的顺序图

Figure 3.10 Working process of the Controller

3.3.4 对象间交互设计

在系统内部，尽管对象间的交互是通过抽象难懂的消息传递机制来完成，但是对于用户服务来说只是一些非常方便的接口方法。该系统采用消息报颁发传递

的间接连接方法，而非简单的传统调用方法来实现对象间的交互，这样可以在外界环境和系统内部对象间完善的隔离，提高系统的安全性。由于消息分发机制是匿名的，所以就不会暴露任何对象的实现细节和其接口信息，甚至于一个对象存在与否对于其它对象来说都是未知的。消息分发机制共包括三个部分：消息发送对象、消息接收对象和消息分配器。消息发送对象为了发送信息必须知道消息接收对象的句柄，但是反过来除非明确的被告知消息报的来源否则消息报接收对象就不能够确切知道消息的来源。所有这二者间需要交换的数据都是在消息本身内存储的，并不需要对象管理这部分数据，可以保证消息的独立性，从而降低系统的耦合度。系统对象间的数据共享是用系统上下文来实现的。设计了一个 sysContext 类，以树状结构存储各个级别的共享变量。这些变量被实例化后，作为对象被以树状结构存储在 sysContext 类型的内存对象中。每个系统生存周期内的对象均可以访问这个共享变量池的句柄。以下是 sysContext 类对象的基本结构：

```
public class sysContext implements idtBag {
    Hashtable c;
    public sysContexto{
        c = new Hashtableo; )
    public void put(String itemName, Object item)
    public Object get(String itemName);
    public void remove(String itemName);}
```

3.5 系统数据库设计

3.5.1 网络办公系统基本表

基于 WEB 的在线网络办公自动化系统是一个大型的数据库应用程序，总体的设计思想是通过网络上的共享数据库来访问数据库进行读写等操作。针对高校办公管理的实际情况，本系统后台数据库开发环境采用 Microsoft SQL Server 2000。数据库共有 10 个数据表：用户信息表、联系人表、传阅的公文信息表、个人的日程信息（events）表、假期管理（Holiday）表、消息管理（message）表、公共信息管理（notice）表、部门管理（UserTeam）表、工作流管理（WorkFlow）表、工作审批流程（WorkFlowStep）。

数据表的结构详见如下表格。

(1) 用户信息表 (account)：用于保存系统中各个用户的相关信息。用户信息表的结构如表 3.11 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	ID	Int	4	自动编号	
2	TeamId	Int	4	所属部门	
3	username	nvarchar	50	登录用户名	
4	password	nvarchar	50	登录密码	
5	Privilege	Int	4	访问权限	
6	Oicq	nvarchar	100	Oicq	
7	Email	nvarchar	100	电子邮箱	
8	lastTime	smalldatetime	4	最后登录时间	
9	Friends	Ntext	16		
10	Telephone	nvarchar	255	固定电话	
11	mobilephone	nvarchar	255	移动电话	
12	Duty	nvarchar	50	职务	

表3.11 用户信息表

Table 3.11 User information

(2) 联系人表 (ContactBook)：用于保存用户个人的联系人的相关信息，表的结构如表 3.12 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	ID	Int	4	自动编号	设为主键
2	ContactName	nvarchar	50	联系人姓名	
3	Sex	nvarchar	50	性别	
4	Oicq	nvarchar	100	Oicq	
5	Email	nvarchar	100	电子邮箱	
6	MSN	nvarchar	50	MSN	
7	Telephone_H	nvarchar	255	固定电话(家)	
8	Telephone_0	nvarchar	255	固定电话(办公)	
9	mobilephone	nvarchar	255	移动电话	
10	Address	nvarchar	255	家庭住址	
11	postalcode	nvarchar	50	邮编	
12	Profile	Ntext	16	联系人备注信息	
13	Addtime	smalldatetime	4	添加时间	
14	Holder	nvarchar	50	拥有人	

表 3.12 联系人表

Table 3.12 ContactBook

(3) 传阅的公文信息表 (document)：用于保存需要传阅的公文的详细信息，信息表的结构如表 3.13 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	ID	Int	4	自动编号	
2	d_title	Nvarchar	255	公文标题	
3	d_content	Ntext	16	公文详细内容	

4	d_team	Nvarchar	255	接收部门	
5	d_user	Ntext	16	接收用户	
6	d_adder	Nvarchar	50	发布人	
7	d_hits	Int	4	点击次数	
8	d_level	Nvarchar	50	重要等级	
9	d_secret	Nvarchar	50	保密等级	
10	d_alluser	nvarchar	255	传阅用户列表	
11	d_del	nvarchar	255	删除标志	
12	d_delAdder	int	4	删除用户	
13	publishtime	smalldatetime	4	发布时间	
14	d_flag	int	4	状态标志	
15	Reader	ntext	16	已阅读用户列表	

表 3.13 传阅公文信息表
Table 3.13 Document information

(4) 个人的日程信息 (events) 表：用于保存单位内部的员工所个人日程相关信息。日程表的结构如表 3.14 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	e_id	int	4	自动编号	
2	e_content	ntext	16	详细内容	
3	e_date	datetime	4	日期	
4	e_hour	int	4	时	
5	e_minute	int	4	分	
6	e_addtime	datetime	4	添加时间	
7	e_username	nvarchar	50	所有者	
8	iseveryday	bit	1	是否每日必做	
9	Iscount	bit	1	是否倒计时	

表 3.14 个人日程信息表
Table 3.14 Personal scheduling

(5) 假期管理 (Holiday) 表：用于保存高校员工请假、审阅、批准，以及单位公假的信息。结构如表 3.15 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	ID	int	4	自动编号	设为主键
2	holidayName	nvarchar	255	假期说明	
3	StartDate	smalldatetime	4	起始日期	
4	EndDate	smalldatetime	4	结束日期	
5	Profile	ntext	16	假期备注	
6	Adder	nvarchar	100	发布人	
7	Assessor	nvarchar	100	审批人	
8	holidayType	nvarchar	50	假期类型	个人请假、公假
9	PublishTime	smalldatetime	4	发布时间	
10	AssessTime	smalldatetime	4	审批时间	
11	HolidayStatus	int	4	审核状态	等待审核，已审核，被返回

12	Postil	ntext	16	审批批注	
13	AdderDel	int	4	请假人删除标志	
14	AssessorDel	int	4	审批人删除标志	

表 3.15 假期管理表
Table 3.15 Holiday management

(6) 消息管理(message)表： 用于保存高校内的员工之间相互交流，发送消息的相关信息。结构如表3.16 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	m_ID	int	4	消息编号	设为主键
2	m_sender	nvarchar	50	发布人	
3	m_receiver	nvarchar	50	收件人	
4	m_title	nvarchar	255	标题	
5	m_content	ntext	16	消息内容	
6	m_flagSender	int	4	发送标志	是否已经发送
7	m_flagReceiver	int	4	接收标志	
8	m_isRead	int	4	消息状态	是否已经被阅读
9	m_sendTime	smalldatetime	4	发布时间	
10	m_file	nvarchar	250	消息附件	
11	m_SorR	int	4		
12	rDel	int	4		
13	sDel	int	4		
14	m_level	nvarchar	50	消息等级	普通、重要、紧急、特急

表 3.16 消息管理表
Table 3.16 Private message management

(7) 公共信息管理(notice)表： 用于保存高校发布的电子公告、政策文件、规章制度等公共信息。结构如表3.17 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	n_id	int	4	设为主键	
2	n_type	nvarchar	50	信息类型	公告、规章制度、政策文件等
3	n_title	nvarchar	50	标题	
4	n_content	ntext	16	详细内容	
5	n_time	datetime	4	发布时间	
6	n_hits	int	4	浏览次数	
7	n_url	nvarchar	255	相关链接	
8	n_adder	nvarchar	50	发布人	
9	n_level	nvarchar	50	信息等级	
10	Reader	ntext	16	已浏览人的列表	

表 3.17 公共信息管理表
Table 3.17 Public message management

(8) 部门管理 (UserTeam) 表：用于保存高校内部部门信息。结构如表3.18所示。

序号	字段名	数据类型	长度	字段描述	备注
1	Teamid	Int	4	部门编号	设为主键
2	teamname	nvarchar	50	部门名称	
3	teamProfile	nvarchar	255	部门描述	
4	createTime	datetime	4	创建时间	

表 3.18 部门管理表
Table 3.18 Department management

(9) workflow管理 (WorkFlow) 表： 用于保存发布、批阅工作内容的信息表。结构如表3.19 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	w_ID	int	4	自动编号	设为主键
2	w_title	nvarchar	255	标题	
3	w_content	ntext	16	详细内容	
4	w_adder	nvarchar	50	发布人	
5	w_level	nvarchar	50	重要等级	普通、重要、紧急、特急
6	w_alluser	nvarchar	255	审批人员列表	
7	w_del	nvarchar	255	删除标志	
8	w_delAdder	int	4	删除人	
9	w_pass	nvarchar	255	通过审批的人	
10	w_currentUser	nvarchar	50	当前审批人	
11	w_flag	int	4	状态标志	处理中，被返回，已完成等
12	publishtime	datetime	4	发布时间	

表 3.19 workflow管理表
Table 3.19 Workflow management

(10) 工作审批流程 (WorkFlowStep)：用于保存 workflow被批阅的相关信息。结构如表 3.20 所示。

序号	字段名	数据类型	长度	字段描述	备注
1	ID	int	4	自动编号	设为主键
2	w_ID	int	4	workflow编号	
3	StepName	nvarchar	255	流程步骤名	
4	username	nvarchar	50	审批人姓名	
5	w_pass	int	4	是否通过标志	1: 通过, 0: 未通过
6	Remark	ntext	16	审批意见	
7	w_passtime	smalldatetime	4	审批通过时间	

表 3.20 工作审批流程表
Table 3.20 Workflow authority information

第 4 章 高校网络办公系统的实现

4.1 系统开发平台

网络办公系统的实现,使得用户的本地机与 OA 服务器之间不再受到网段、地域的限制,极大地方便了办公用户。本软件所使用的开发工具主要为 JSP (Java Server Page) 技术, JSP 是 Sun 公司推出的新一代站点开发语言,可以在 Servlet 和 JavaBean 的支持下,完成站点程序的编写。其主要特性如下:

(1) 将内容的产生和显示进行分离

使用 JSP 技术, Web 页面开发人员可以使用 HTML 或者 XML 标识来设计和格式化最终页面。使用 JSP 标识或者小脚本来产生页面上的动态内容。产生内容的逻辑被封装在标识和 JavaBeans 群组件中,并且捆绑在小脚本中,所有的脚本在服务器端执行。如果核心逻辑被封装在标识和 Beans 中,那么其它人,如 Web 管理人员和页面设计者,在编辑和使用 JSP 页面的同时不影响内容的产生。在服务器端, JSP 引擎解释 JSP 标识,产生所请求的内容(例如,通过存取 JavaBeans 群组件,使用 JDBC 技术存取数据库),并且将结果以 HTML (或者 XML) 页面的形式发送回浏览器。这有助于作者保护自己的代码,而又保证任何基于 HTML 的 Web 浏览器的完全可用性。

(2) 强调可重用的群组件

绝大多数 JSP 页面依赖于可重用且跨平台的组件(如: JavaBeans 或者 EnterpriseJavaBeans)来执行应用程序所要求的更为复杂的处理。开发人员能够共享和交换执行普通操作的组件,或者使得这些组件为更多的使用者或者用户团体所使用。

(3) 采用标识简化页面开发

Web 页面开发人员不会都是熟悉脚本语言的程序设计人员。JavaServerPage 技术封装了许多功能,这些功能是在与 JSP 相关的 XML 标识中进行动态内容产生所需要的。标准的 JSP 标识能够存取和实例化 JavaBeans 组件,设定或者检索群组件属性,下载 Applet,以及执行用其它方法更难于编码和耗时的功能。

通过开发定制化标识库, JSP 技术是可以扩展的。今后,第三方开发人员和其它人员可以为常用功能建立自己的标识库。这使得 Web 页面开发人员能够使

用熟悉的工具和如同标识一样的执行特定功能的构件来工作。

4.2 系统主要对象的实现

4.2.1 模型(model)的实现

4.2.1.1 系统通用控制组件

(1) 柔性集成

A. 用户、角色管理器

用户角色管理实现了角色到办公实体的角色映射。以授权为核心,面向 workflow 和面向角色管理,实现用户到角色,角色到数据库表,角色到数据库表字段,角色到 workflow 段的映射。系统的超级管理员可以使用用户角色管理器建立用户,建立角色,为用户分配角色建立角色时可以对当前角色的工作权限,选择相应的表分配给当前的角色,并设定对这些表的操作类型。

B. 事务对象管理器

是描述事务临时状态集合的缓存,体现业务规则和多个角色的协作,并有自己的生存周期。一项事务从某个时间点诞生,在某些时间点通过操作算子处理,到某个时间点提交自杀请求(如要求和数据库同步,外设输出),从而完成一项任务。如图 4.1 为一个事务创建子对象。

文件登陆单

文件编号		收文时间			
来源单位					
文件字号					
文件类型		份数		经手人	
文件					
批示人				办给时间	

图 4.1 一个事务创建子对象

Figure 4.1 A transaction creation object

C. 操作集管理器

操作集就是以一般用户身份进入系统后,用来进行具体操作的命令集。可以为用户和角色分配具体的操作,设定操作类型等。

(2) 人员组织管理模块

人员组织管理模块的管理功能包括：人员管理，组织管理，岗位角色管理和系统角色管理。

A. 与各应用系统的交互方法：

该模块将人员管理数据以xml格式的形式提供给各个应用系统，各前端应用可以解析该文件生成中的数据，用javascript或applet生成可视化界面。xml的格式如下(其中idtdept表示单位，idtog表示部门，idtpos表示岗位，name为其名称，id为唯一标识。整个xml文件表示出了高校的机构组织之间的关系。)

提供给各应用的xml格式如下所示，各前端应用可以解析该文件生成中的数据，用javascript或applet生成可视化界面。

其中 idtdept 表示单位，idtog 表示部门，idtpos 表示岗位，name 为其名称，id 为唯一标识。整个 xml 文件表示出了市高校的机构组织之间的关系。

```
<?xml version="1. 0" encoding="GB2312"?>
<root>
<idtdept name= "XX 高校办系统" id="00 1" >
  <idtog name=" xx 办公室" id="002">
    <idtogl name= "内设机构" id="103" >
      <idtog namr=" 秘书一处" id=23" >
        <idtoos name=" 处长" id="25">
          <idtpos name=副处长(主管发文)" id=" 26" >
            <idtpos name=" 副处长(主管收文)" id="27">
              <idtpos name=" 科员(总收文)" id="30" / >
            <idtog>
          <idtog name= "秘书二处 " id= "40" />
            <idtog name= "教卫文体处" id="22"/>
          < / idtog>
        <idtog name>
          <idtog naille=" 工作部门" id=" 77" >
            <idtog name= "组成部门" id=" 80" >
              <idtog name= "XX 系" id="100"/>
```

```
</aidtog>
<idtdept>
< / root>
```

B. 可从应用系统的其它部分或其它应用系统提取用户可操作集

当各应用将本系统的角色与岗位进行对应之后，可根据传来的岗位编号返回相应的操作；各岗位编号之间以“；”相隔，便于平台的权限控制模型对权限数据的加载。

(3) 并发控制

被锁定资源信息是以UserLockUnit对象为一个存储单位的，其结构如下。被锁定资源信息是以UserLockUnit对象为一个存储单位的，其结构如下。

```
public class UserLockUnit{
    String lockedTable=null;//被锁定的字段所属的表名
    String lobkedField=null;//锁定的字段的名称
    String username=null;//定此资源的用户名称
    String keyword=null;//锁定的数据库纪录的关键词
    Public UserLockUnit (String lockedTable, String lockedField, String
keyword, String username)
    Throws Exception;//构造函数
}
```

UserLockedResources是UserLockUnit对象的集合类；其实例是一个全局变量，用来存储整个系统被锁定的资源信息，其结构如下：

```
Hashtable lockResoures;
Public UserLockedResoures(){
    LockResources=new Hashtable();
}
//锁定某资源
public void lock(UserLockUnit re) throws Exception;
//释放资源
public void unlock(UserLockUnit re) throws Exception;
//判断某资源是否已被锁定
public Boolean ifselfLocked(UserLockUnit re) throws Exception;
```

```
//判断某资源是否是已被锁定
public boolean isLocked(UserLockUnit re) throws Exception;
//打印出所在资源的状态
public void printResourceStatus() throws Exception;
//获得被锁定的资源列表
public Hashtable GetLockedSrc() throws Exception;
//复制被锁定的资源
public UserLockedResources Copy()throws Exception;
//设置被锁定的资源
void Set(Hashtable locksrc) throws Exception;
}
}
```

(4) 数据库连接池

首先构造内部类PooledConnection做为数据库连接单元，PooledConnection的变量有二：一个是connection类型的变量，用来存储与数据库的实际连接，一个是boolean类型的变量，标识是否有用户正在使用本连接构造ConnectionPool类就是承载数据库连接的“池子”了。ConnectionPool类最重要的数据成员就是一个由PooledConnection类型变量组成的Vector类型的变量和与一些连接池管理机制相关的函数。下面给出PooledConnection、ConnectionPool的主要结构

```
public class connectionPool{
    public static final String NAME="ConnectionPool";
    private String jdbcDriver="";
    private String dbUrl="";
    private String dbUsername="";
    private String dbPassword="";
    private String testable="";
    private int initialConnections=10;
    private int incrementalconnections=5;
    private int maxConnections=50;
    private Vector connections=null;
```

```

    public ConnectionPool(String jdbcDriver,String dbUrl,String dbUsername,String
dbPassword){
        this.jdbcDriver=jdbcDriver;
        this.dburl=dburl;
        this.dbUsername=dbUsername;
        this.dbPassword=dbpassword;
    }
    //返回连接池初始建立时的连接数目
    public int getInitialconnections();
    //设置连接池初始建立时的连接数目
    public void setInitialConnections(int initialConnections);
    //返回连接池数目不足建立新的连接时采用的增量
    public void getIncrementalConnections();
    //设置因连接池数目不足建立新的连接时采用的增量
    public void setIncrementalConnections;
    //返回连接数据库连接的最大数目
    public int getMaxConnections();
    public void setMaxconnections(int maxConnecti)
    //返回用测试的表名
    public string getTestTable();
    //设置用于测试的表名
    public void setTestTable(String testTable){
        this.testable=testable;
    }
    //创建一个连接池
    public synchronized void createPool() throws Exception;
    //为连接池创建一定数目的连接，并把这些连接放在池中
    private void createConnections(int numConnections) throws
        SQLException;
    //创建一个新的数据库连接
    private connection newconnection() throws SQLException;
    //调用getFreeConnection()获取一个可用的数据库连接，如果可用连接

```



```

//函数将以增量数目创建连接并等待直到有可用的连接
public synchronized connectiongetConnection() throws
    SQLException;
//获取一个可用的数据库连接
private connection getFreeConnection() throws SQLException;
//找到
private Connection getFreeConnection() throws SQLException;
private Boolean testConnection(Connection conn);
//释放一个使用完毕的数据库连接，将其BusyFlag置位
public void returnConnection(connection conn);
//刷新连接池中所有的数据库连接
public synchronized void rereshConnections() throws
    SQLException;
//关闭所有的连接，将数据库连接池清空
public synchronized void closeConnections() throws SQLException;
//等待
private void wait(int msecondes){
    try{
        Thread.sleep(mSeconds);
    }catch(InteruptedException e){
    }
}
//内部类，用来封装一个已被放连接池的数据库连接的一些属性
class PooledConnection{
    Connection connection=null;

    Boolean busy=false;//此连接是否正在被使用
    Public connection getConnection();
    Public void setConnection(Connection connection);
    Public Boolean isBusy();
    Public void setBusy(Boolean busy);
}

```

}

在系统控制模型中设有数据库适配的类：DBAdapter，将数据库操作的异构性屏蔽在系统控制的核心层中。而应用开发人员不必关心大部分的数据库异构性给系统带来的差异。下面是DBAdapter类的概述描述：

```
public class DBAdapter{
//数据库类型
static string sqlDB="MS SQL Server";
static string orcledb="Oracle";
static string db2db="IBMDB2";
//字段描述
static string ZF_fieldtype="chat";
static string sz_fieldtype="numeric";
static string rq_fieldtype="date";
static string wb_fieldtype="longfield";
static string lb_fieldtype="longbase64";
static string js_fieldtype="clobfield";
//字段类型描述标识符
static string filed_type_start="[~[";
}
```

4.2.1.2 业务逻辑组件

下面以痕迹保留模型为例介绍一下业务逻辑组件的实现。

4.2.1.2.1 痕迹保留模块的功能设计

痕迹保留的目的是方便对操作的查询，对文件的处理，数据的回滚，保证在业务运行中的数据的安全。痕迹保留的总体原则是：操作透明，有迹可查，权限回滚。

图4.2以发文流程为例说明了痕迹保留在公文流程中的作用位置。公文处于各个流转节点中，在每个节点的处理操作不同；但每个节点中对公文处理的信息都需要有自动归档保存的操作。

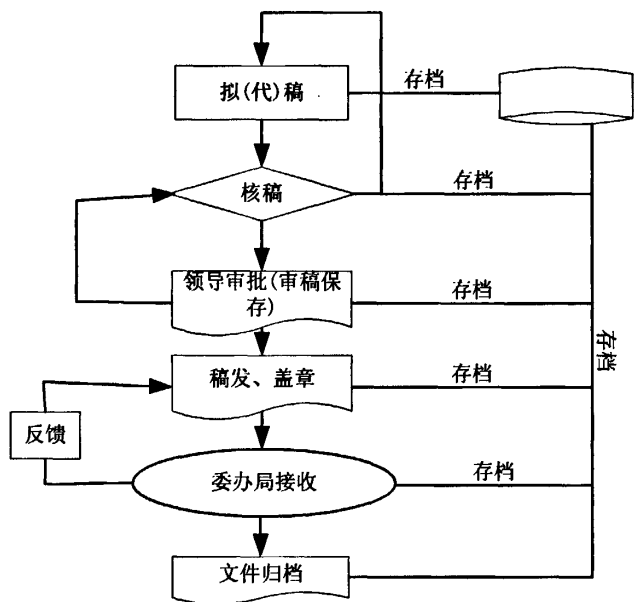


图4.2 发文流程中的痕迹保留图

Figure 4.2 Trace record of the process of document dispatching

公文管理系统中，公文信息中的数据元素分为两大类：一是流转过程中的属性信息，包括文件处理的时间、转发部门、承办单位、办理时限、部门批示等；二是电子公文文件，如以WORD形式存在的公文文档。针对数据元素类型不同，痕迹保留的实现过程不同。痕迹保留的数据库表结构如表4.3所示。其中字段1至字段8是流转过程中的属性信息，字段9是属于公文文档。

编号	字段名	描述
1	主键	唯一标识一条痕变保留纪录
2	文件编号	文件主表中的主键，关联用
3	修改人	当前修改的用户名称
4	修改人部门	当前修改的用户所在的部门
5	修改人级别	当前修改的用户的级别
6	修改人角色	当前修改的用户的角色
7	修改人时间	文档的修改时间
8	修改备注	提示用户当时的修改目的
9	内容	电子公文Word文档内容

表4.3 痕迹保留表结构

Table 4.3 Structure of trace record

4.2.1.2.2 保留对象的分类与实现

(1) 针对属性信息的痕迹保留

每次数据变更时都会对属性信息进行存档。属性信息的痕迹保留只可用作事后追查，不作为回滚的功用。因为属性信息数量繁多，而且不是所有的属性数据都要进行痕迹保留，因此对于是否进行保留存档，将留给用户一个接口，以使用户自行设定。例如，如果属性“转发部门”有必要进行存档，以便在事后进行查询，则可以通过接口进行设置，每次“转发部门”的修改的内容，时间，修改者等信息都可被存档。

(2) 针对电子公文文件的痕迹保留具有以下功能：

① 提供对电子公文文件修改过程的详细记录：详细记录包括操作时间及操作人；操作人对此文稿所做的操作类型（浏览，修改，删除等），这个信息与属性数据的痕迹保留相同；

② 文稿自动存档保存：应用系统对文稿的自动备份。有两种触发方式，一是系统自动存档，即当用户每次提交时由系统进行，二是手工存档，即当用户觉得有特殊必要时，可以额外进行手工存档；

本系统中设计了两个微操作：“痕迹保留微操作”和“痕迹保留查看微操作”来实现公文流转中的痕迹保留功能。一个具有激活状态且能够具有回滚功能的“痕迹保留微操作”在系统中的xml描述如下：

```
<markkept switch="ON" rollback="ON" tablename="tableA">
<mapping>
<mappintunit src="tableA.fieldA" des="MARK_KEPT_TABLE.fieldA"/>
<mappingunit src="tableA.fieldB"
des="MARK_KEPT_KEPT_TABLE.fieldB"/>
</mapping>
</markkept>
```

switch描述了是否打开此微操作；rollback标签描述了是否可以进行回滚操作，二标签描述了保留修改痕迹时的数据映射关系。

当应用开发人员把上面的微操作分配给某角色时，具有该角色的用户在系统中操作表tableA时，就具有对tableA中的公文信息修改记录的权限。这个动作是由控制器在解析用户操作的模板时发现表示“打开痕迹可留开关”的标签触发的。

4.2.1.2.3 对痕迹保留数据进行处理权限控制

由于事后追查和文稿恢复都是针对公文或相关流转属性进行操作,涉及到公文走向和阅览权限,因此必须对这两个操作的权限进行严格控制。本系统会根据各个部门,各个使用者的实际情况,进行权限的合理分配。需要分配的权限有对某种公文是否可查看历史存档、是否可进行版本回滚、是否可进行事后追查等三种。在用户角色表中增加用户级别。级别高的用户可以回滚级别低的用户对公文电子文档所做的操作。级别的组合设计为组合算符,在角色管理中维护和分配。

4.2.2 视图(View)的实现

(1) 检索单元实现

办公自动化系统开发中对信息检索单元的重用度是最高的。下面以一个信息检索单元在页面上的使用为例阐述系统中对应的实现。下面是对应此信息检索单元的代码。

```
<form name="form1" method="post" os_param="Query" action="公文审批">
  <table>事实编号<input type="text" name="t_fact_id" size=11>
    <option>AND</option>
    <option>OR</option>
  </select> 事实名称<input type="text" name="t_fact" size=26 class=1>
  <select size=1 name=d2 [AND/OR] class=1>
    <option>AND</option>
  </select>事实备注<input type="text" name="t_remark" size=18 class=1>
</font><select size=1 name=d3 class=1>
  <option>模糊</option>
  <option>精确</option>
</select>
</table>
</form>
```

检索单元相关的标签有: {AND/OR} 和 {FUZ/ACC}, 它们是内嵌在一个<select>标记中使用的, 配合<select>标记中的可选分别表达条件组合的方式(“并且”或“或者”)、查询的方式(“精确”或“模糊”); os_param=“Query”, 指明标form是一个检索单元。

(2) 内容显示单元实现

A. 标签与用户界面

以单行循环显示的开发代码为例：

```
<form name="add" method="post" os_param="browse">
<table>
  <b>事实列表</b>
  <b>总记录数: [count]</b>
  <input type=button value=添加选中事实到前件表>
<tr>
<td>编号</td>
<td>事实编号</td>
<td>事实名实</td>
</tr>
<tr singleLoop>
<td .....>[NO]</font></td>
<td.....><input type=checkbox name=c1 value=ON></td>
<td.....><fact.fact_id.numeric.order_asc></td>
<td.....>hotlink</td>
</table>
</form>
```

[Count]标记可以使用在用来显示单行循环数据的form中的任何一部分，当业务模型数据提取完毕后，会被视图转换成本页上需要单行循环的记录数目。而 SingleLoop只能内嵌在一对<tr>标记中使用，有 SingleLoop出现的<tr>标记和与之对应的</tr>标记之间的所有内容会被重复显示。Order_asc标记指出页面数据排序依据的数据列.hotlink指出页面上生成的需要生成热点链接的数据列。

B. 程序设计

单行循环显示和多行循环显示是内容显示单元的两个基本组成部分。此处用到了Factory的设计模式。工厂模式的定义是提供创建对象的接口，工厂模式就相当于创建实例对象的new。根据实际需要：内容显示单元可能会有很多种表现形式，如果不采用工厂模式，我们以后每增加一种形式的内容显示单元，或者修改以前使用过的内容显示单元的名称，都要改动与其初始化相关的代码，设计一个“内容显示单元”的工厂类，可以随意增加新的形式的内容显示单元类，而只需要修改工厂类的代码即可。采用工厂模式的设计带来更大的可扩展性和尽量

少的修改量。内容显示单元中的工厂模式如图4.4。

我们设计的工厂类代码如下。

```
public class displayFactory{
private string type="";

static public idtDisplay gendislay(form form_one) throws badviewException{
if (form_one instanceof idtlist){
    return new listview(form_one);
}
else if (form_one instanceof idtNotlist){
    return new notListView(form_one);
}
else
{ throw new badviewException("Bad view request");
}
}
}
```

内容单元接口：

```
public interface idtdisplay{
void show();
void writeshow (idtcontext cout,helper helper) throws exception;
```

实现单行循环显示单元和多行循环显示单元的都实现了idtDisplay接口。

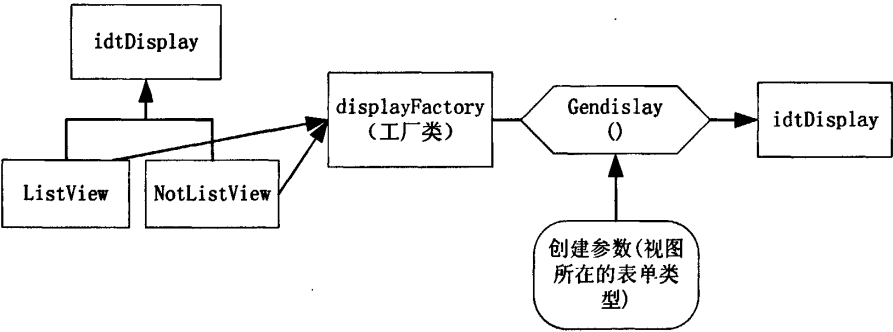


图4.4内容显示单元中的工厂模式

Figure 4.4 Factory model of the contents representation unit

(3) 页码选择单元

每个数据显示页面上的页码和页码选择按钮是自动生成的。系统按照当前页面上的数据提取需求向数据库请求数据，查询完毕后统计数据纪录的条数时读取本页的最大显示纪录条数的配置，如果总纪录条数小于等于显示纪录条数，就将纪录全部显示在页面中，反之，则显示最大条数的纪录，并将当前已经显示的纪录条数附加在翻页请求的链接中，然后把此链接作为翻页请求按钮的写在HTML页面上。

4.2.3 控制器（Controller）的实现

控制器部分的主要形式是一个Servlet。其中Coontext类封装了整个系统的上下文环境，FormElement封装了表单的数据对象。BlockBuilder类封装了被提交的页面的数据模型。ModelManager封装了模型的管理机制，而ViewManager封装了视图的管理机制。athorityCheck类封装了权限检查机制。控制器有序地与这些类的交互。在上下文环境、表单的数据对象、页面的数据模型、模型和视图之间传递控制权和数据，完成客户端数据被业务逻辑模型处理最后被视图显示的过程。

4.3 模型接口的实现

4.3.1 内部业务逻辑模型接口与内部视图模型接口

内部业务逻辑模型接口在“模型注册表”中的实现形式和内部视图模型接口在“视图注册表”中的实现形式是两张数据库表，分别存储了内部业务逻辑模型和内部视图模型的一些属性，接口是由平台的核心控制部分来检索数据并生成模型和视图的“路由”信息的。

4.3.2 外部业务逻辑模型接口

内部模型接口是系统控制模型与业务逻辑模型(内部)之间的接口。是系统控制模型与业务逻辑模型(外部)之间的接口。这部分是“模型注册表”的组成部分，其形式是一个符合XML1.0标准的文件。

```
<modelNavigate>
<business-name>公文审批(校长)</business -name>
<controller-name>controller]<controller-name>
<model-class>com.idt.model.DocumentProcess.OadocumentType</model-class>
```



```
>
    <init-param >
        <param-name>Name</param-name>
        <param-name>Time</param-name>
        ....
    </modelNavigate>
```

4.3.3 外部视图模型接口

内部视图模型接口是系统控制模型与内部视图模型之间的接口。外部视图模型接口是系统控制模型与外部视图模型之间的接口。这部分是“视图注册表”的组成部分。其形式是一个符合XvILL0标准的文件。举例如下：

```
<viewNavigate>
<business-name>公文审批<business -name>
<controller-name>controller</controller-name>
<view-class>com.idt.view.NotListView.OadocumentType</view-class>
<init-param >
    <param-name>Name</param-name>
    <param-name>Time</param-name>
    ....
</modelNavigate>
```

4.3.4 接口解析器

系统控制模型采用一个接口解析器来解析“模型注册表”和“视图注册表”。这个解析器是基于JDOM1.0。编写的，由一个工具类包，两个解析类和一个接口组成。其中XMLTools类包扩展开发了通用解析工具，XMLModelPaser类和XMLViewPaser分别实现了XMLPaser接口，封装了模型解析器和视图解析器。

4.4 系统实现界面浏览

应用程序反馈：运行程序时系统随时将系统正在做什么的信息告诉用户，否则，用户就不知道是计算机出了问题还是应用系统有问题。

默认值：是指只要能预知答案，就可以设置默认值，这样节省了用户的工作。

复原：用户可能会出错，想退回去并且重新开始。界面应该提供返回恢复先前状态的能力。另外，系统应该具有较高的容错能力。即使用户某一操作错误，也不应该对系统产生致命的、不可恢复的影响。所以系统应该具有允许用户出错，并且能够恢复出错前的能力。对于一些具有破坏性的操作，如删除等，系统应该给用一次确认的机会，防止由于误操作对系统造成的破坏。

一致性：人机界面的一致性主要体现在输入输出方面的一致性，具体是指在应用程序的不同部分，甚至不同应用程序之间，具有相似的界面外观，布局，相似的人机交互方式以及相似的信息显示格式。一致性有助于用户学习，减少用户的学习量和记忆量。

4.4.1 系统登录及密码修改模块

实现目标：程序启动后，首先进入“系统登录”界面验证操作员密码。系统设置口令，以增加保密性，同时可以通过登录用户的身份不同区分主程序的应用权限。如，当用户以系统管理员身份登录时可以获得主程序的全部权限；当用户以普通用户的身份登录时，用户只能获得与普通用户职能相符的权限。这样能更加保证系统的安全性，更符合现代高校内部的管理习惯。

系统登录窗口主要实现如下功能：

- 可选择操作员，并凭口令进入系统，支持键盘操作。
- 操作员错误或密码错误提醒并返回错误点。
- 实现登录《高校网络办公自动化管理系统》主界面。
- 输入三次错误密码自动退出系统。

系统登录界面如图 4.5所示：



图 4.5 系统登录界面

Figure 4.5 System login interface

为了用户使用方便，增强数据的安全性，在用户正常登录后，主程序系统功能菜单中，有修改用户个人信息（包括密码）功能，每位用户只能修改自己的个人信息。

4.4.2 主程序窗体模块

实现目标：程序启动后，在“系统登录”界面验证操作员密码后，将进入主程序界面。主程序界面主要实现如下功能：

- 默认显示我的办公桌面。
- 可通过菜单和工具栏快速选取项目，调用相应子程序。
- 操作具有权限的项目。
- 显示部门名称、系统日期和时间。

主程序界面如（图 4.6）所示：

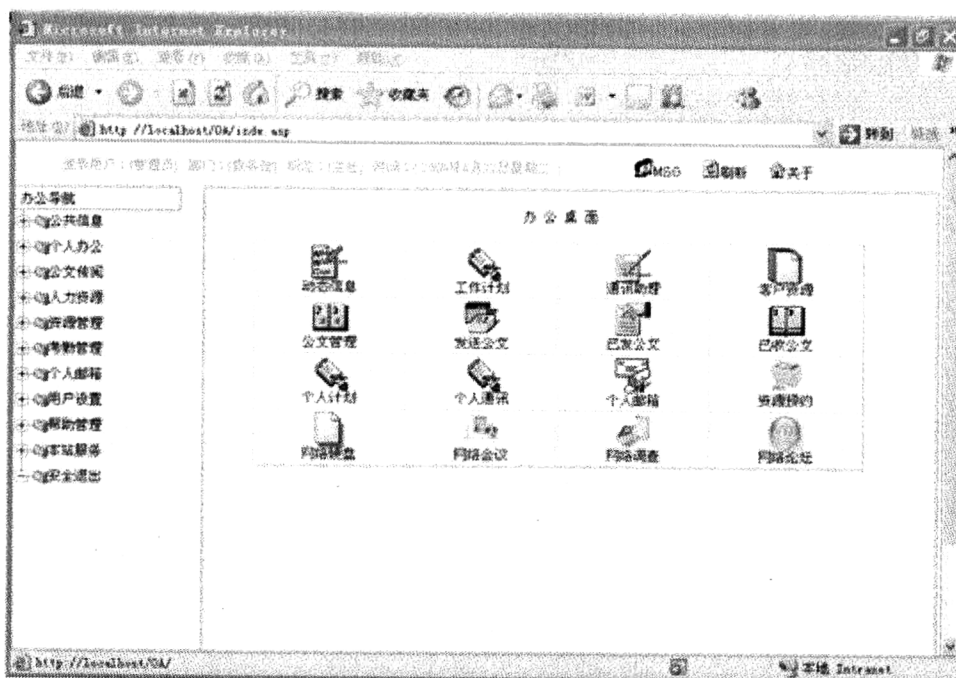


图4.6 主程序界面

Figure 4.6 Interface of main program

4.4.3 公共信息查询

公共信息查询模块主要是能够查询单位发布的各种办公信息，规章制度信息查询如图4.7 所示：

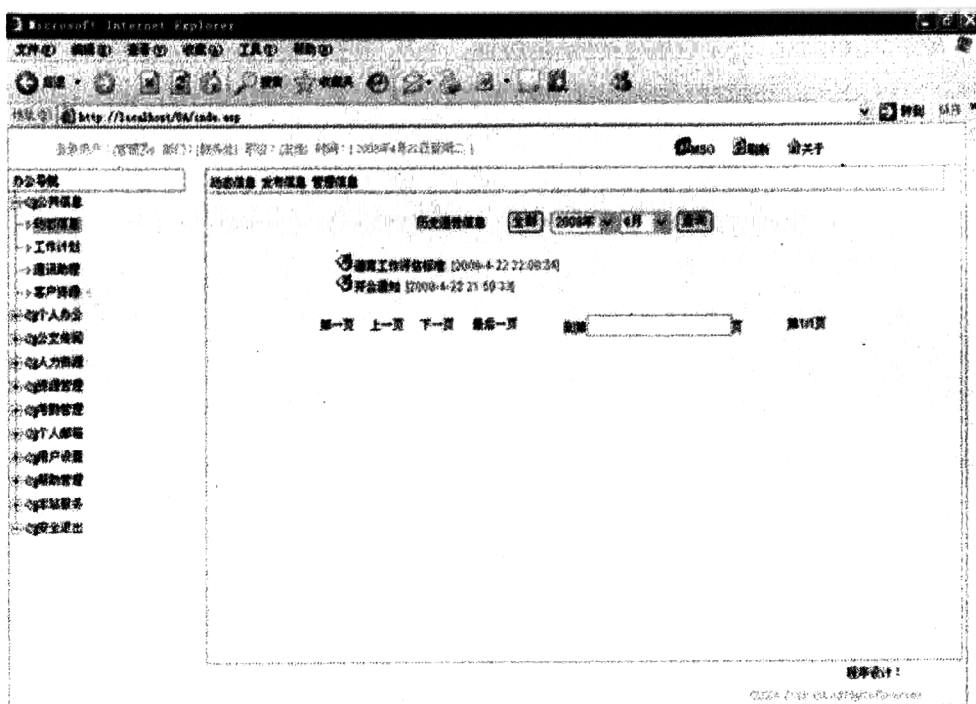


图4.7 规章制度公共信息查询界面

Figure 4.7 Query interface of rules and regulations of public information

4.4.4 个人办公管理模块

个人办公管理模块的功能主要是方便用户制定个人工作计划、修改个人档案、增加个人通讯录以及个人资料维护等。如图4.8 所示：

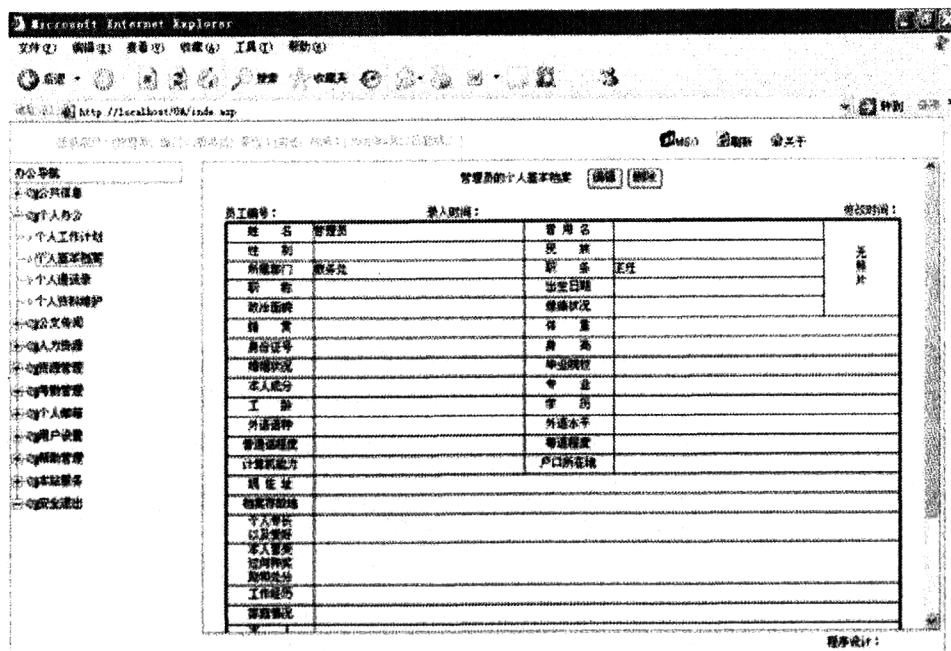


图 4.8 个人办公管理界面

Figure 4.8 Interface of personal office management

[illegible]

Figure 4.9 Interface of newly created document forwarding

4.4.6 人力资源管理模块

55

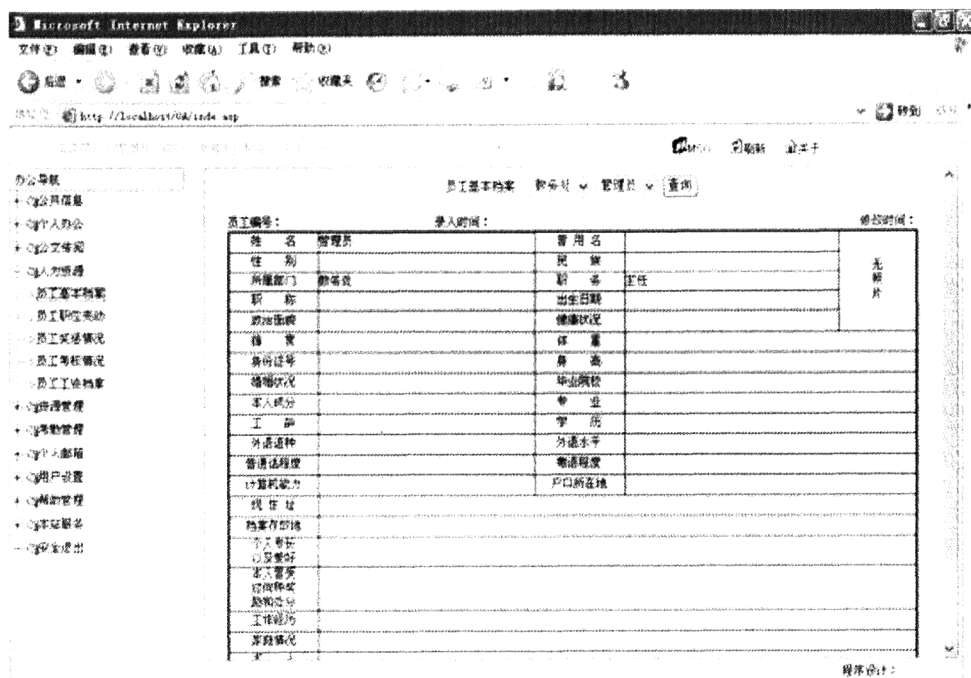


图4.10 人力资源管理界面

Figure 4.10 Interface of Human Resource Management

4.4.7 考勤管理模块

考勤管理模块的主要功能是根据工作需要制定考勤时间，分别记录日、月考勤统计的状态信息。

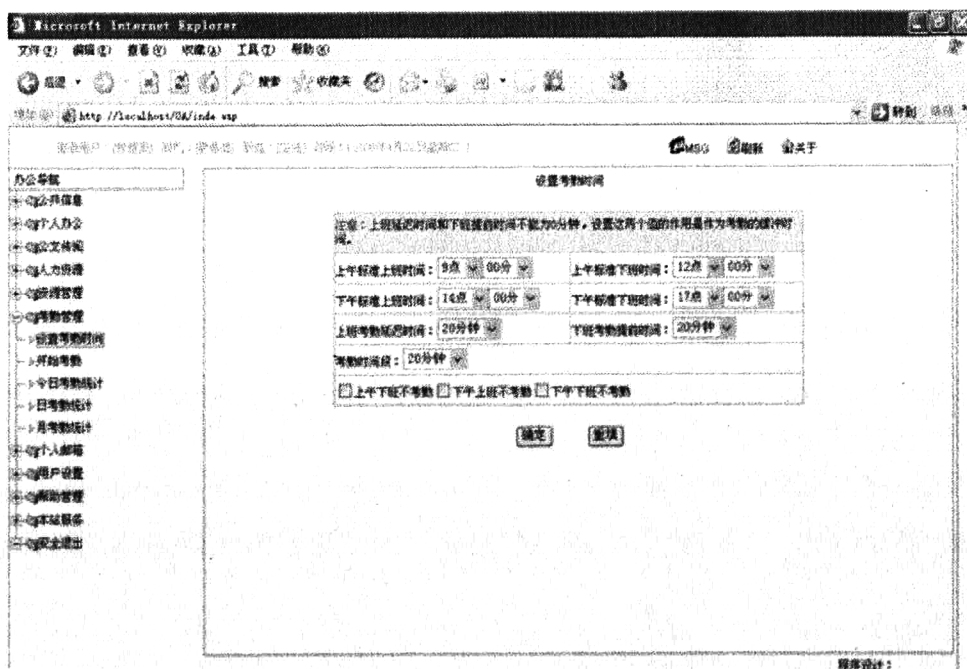


图4.11 考勤管理模块界面

Figure 4.11 Interface of attendance management module

4.4.8 其他管理模块

本系统另外还包括了个人邮箱、用户设置、帮助管理等模块。个人邮箱实现了个人与外界信息的接收与发送。在用户设置中根据用户的级别设置了不同的管理权限。另外，在帮助管理模块中提供了“使用系统说明”等项目。

第 5 章 结论与展望

5.1 总结

本文介绍了高校网络办公自动化系统的内涵、发展以及基于 JZEE 构架技术的 OA 系统设计思想。在此基础上, 设计和实现了一个基于 Web 的高校网络自动化办公系统, 介绍了它的组成及各个组成部分的功能。整个系统采用当今流行的 JZEE 架构, 通过跨平台的 Java 技术以及制作动态网页的 JSP 技术开发 WEB 应用程序来完成。整个系统采用模块化设计, 实现了工作流程自定义、部门功能自定义, 提供电子邮件功能, 并支持各种大型数据库。

在使用上, 用户只需要通过浏览器就可访问此系统, 且只需简单的点击及文本的输入就可轻松完成工作, 不用安装客户端软件, 而且因为使用简单, 培训费用相对低廉, 此外对于开发人员和维护人员来说, 程序的维护和管理只需要在服务器端进行, 集中、方便, 易于升级和修改。

5.2 展望

由于笔者的水平和时间有限, 所以还有很多问题在本文中讨论不深或是没有讨论, 有待进一步的研究, 这些问题主要包括:

- 1、基于柔性集成的工作流程控制技术有待于进一步完善。

尽管本文所设计的基于柔性集成的流程控制体系能够满足一般的政务办公自动化系统对公文流转等办公流程的实现、组合和定制的需求, 但距离真正灵活的流程定制还有一定距离。

- 2、控制器的职责及其承担的任务有待于进一步明晰。

随着整个框架体系的完善, 需要控制器解决的问题会越来越多, 清晰地划分出控制器的职责是非常重要的, 如果需要的话, 可以将控制器承担的任务分类, 设计出双控制器或者是多控制器, 可能会一定程度上提高控制器工作的效率。

- 3、文件格式的相互转换及自动信息采集有待于进一步研究。

本文对以下的问题并未详细探究其实现方式: 提供友好的信息录入界面, 支持各种文件如 TXT、WPS、FZ 及高校部门专用格式文件等信息的成批注册; 提供如 DBF、ACCESS、ORACLE、SQL SERVER 等外部数据库格式的数据成批导入的功能等。

参 考 文 献

- [1] F. Arcieri, E. Cappadozzi, E. Nardeli et al. "SIM: A working example of an e-government service infrastructure for mountain communities." Workshop on Electronic Government (DEXA-eGoVOI), Conf. on Databases and Expert System Applications (DEXA'01). Munich, Germany, IEEE Computer Society Press, Sept 2001.
- [2] 李峰, 刘新华, 快速构建基于.Net和Domino的简单电子政务系统, 太原理工大学学报, 2004(4): 427-430。
- [3] 刘艳霞, 张磊, 焦作市电子政务信息系统的实现, 福建电脑2004(7): 55-56。
- [4] 古锋, 应用J2EE的一站式办公机制的研究和实现, 北京机械工业学院学报。
- [5] 房质文, 高校机关办公自动化当议, 吉林大学社会科学学报, 2005.3: 92-94。
- [6] 崔俊杰, 丁文浩, 我国电子政务发展探讨, 湖北社会科学, 2004.5: 99-100。
- [7] Paul Jackson, Noah Curthoys. E-government: Developments in the US and UK-12th International Workshop on Database and Expert Systems pplications Munich, Germany, September 03-07, 2001.
- [8] Douglas C. Schmidt, Frank Buschmann. Paterns, frameworks, and middleware: their synergistic relationships. Proceedings of the 25th international conference on Software engineering, Portland, Oregon ,May 2003.
- [9] 吴明晖, 应晶, 何志均, 基于构件的框架式开发方法及其重用库系统, 计算机工程与应用2000.9。
- [10] 任中方, 张华, 闰明松等, MVC模式研究的综述计算机应用研究2004.10: 1-8。
- [11] Michael J. Mahem of Lrraine J. Johnston. Handling Multiple Domain Objects with Model-View-Controler. 32nd International Conference on Technology of Object-Oriented Languages.Melbourne, Australia,November 22-25, 1999.
- [12] Michael Matsson, Jan Bosch, Mohamed E. Fayad, Framework Integration, problems,causes, solutionsvraham Lef James T. Rayfield,Web-Application Development Using the Model/View/Controler Design Patern, Fifth IEEE International Enterprise Distributed Object Computing Conference.September 04-07, 2001.
- [13] W.F. Tichy,A Catalogue of General-Purpose Software Design

- Patems, Technology of Object-Oriented Languages and Systems .July 28-August 01, 1997 .Santa Barbara, California.
- [14] E.E. Jacobsen, B.B. Kristensen, P. Nowack. Patems in the analysis, design and implementation of frameworks, COMPSAC'97-21st International Computer Software and Applications Conference. August, 1997. Washington, DC.
- [15] 于森, 王延章一种基于角色网络模型的电子政务系统框架的研究与实现, 计算机工程与应用2003, 39(2): 31-35.
- [16] 于森, 面向电子政务办公系统的角色网络理论、模型及其应用研究, 大连: 大连理工大学, 2004.
- [17] 李强, 王延章, 元数据在政务办公系统业务中的应用, 计算机应用, 2004年.
- [18] 于森, 王延章陈雪龙, 元数据在电子政务办公系统实现中的应用研究计算机工程与应用, 2003年16期: 28-31.
- [20] 赵宏, 顾明, 应用J2EE Service的“一站式”服用框架模型, 计算机工程, 2004年8月, 第30卷, 第16期.
- [21] 岳坐刚, 张凯兵, 应用J2EE服务的框架结构分析, 四川工业学院学报第23卷增刊.
- [22] 余浩, 窦延平, 应用J2EE services的高校应用的新型动态集成框架, 计算机仿真, 2004年3月, 第21卷第3期.
- [23] Lupu, E. C., and M. S. Sloman (1997a). Reconciling Role Based Management and Role Based Access Control. In (RBAC' 1997)[R], pp. 135-142 Wm Treese, 3CvL, web services, and XML. Volume 6, Issue 3 ,September 2002.
- [24] Joon S. Park, Keith P. Costello, Teresa et al. Neven A composite rbac approach for large, complex organizations .Proceedings of the ninth ACM symposium on Access control models and technologies. Yorktown Heights, New York, USA, June 2004.
- [25] Ninghui Li, Mahesh V. Tripunitara, Security analysis in role-based access control Proceedings of the ninth ACM symposium on Access control models and technologies .Proceedings of the ninth ACM symposium on Access control models and technologies. Yorktown Heights, New York, USA, June 2004.
- [26] Indrakshi Ray, Na Li, Robert France, et al. Using UML to visualize role-based

- access control constraints .Proceedings of the ninth ACM symposium on Access control models and technologies.Yorktown Heights, New York, USA, June 2004.
- [27] Dirk Riehle.Composite design patterns. Proceedings of the 12th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications, Atlanta,Georgia, United States ,October 1997-October 1997.
- [28] Ant6nio Rito Silva, Francisco Assis Rosa, Teresa. Framework description using concern-specific design patterns composition. ACM Computing Surveys (CSUR), 32(1):16-65.
- [29] Sihem Amer-Yahia, Divesh Srivastava A mapschema and interface for XML stores. Proceedings of the 4th international workshop on Web information and data management. McLean, Virginia, USA, November 2002.
- [30] Hasan M. Jami, Giovanni A. Modica An object-oriented extension of XML for autonomous web applications. Proceedings of the eleventh international conference.Information and knowledge management. McLean, Virginia, USA, November 2002.
- [31] C. F. Codella, D. N. Dilenberger, D. F. Ferguson, et al. Support for enterprise JavaBeans in component broke. IBM Systems Journal,37(4).
- [32] Eric Jung, Dave Writz, Steven Brodhead. Java)GAL Programmer's Reference. UK:Wrox Press Ltd.,2001.
- [33] 刘运龙, 黄烟波一种基于双Servlet控制器的MVC模式的设计与研究, 计算机应用。2005, 25(1): 248-250。

致 谢

本论文的工作是在我的导师刘宏副教授的悉心指导下完成的,刘老师严谨的治学态度和科学的工作方法给了我极大的帮助和影响。我衷心感谢刘老师一直对我的谆谆教诲和无微不至的关怀。我也从刘老师身上学到不少做人和做学问的道理。尽管刘老师工作非常繁忙,但他总是抽出时间为我们指导、从无怨言。刘老师平易近人的为师之道、一丝不苟的治学态度,渊博的专业知识和高尚的人格魅力都使我终生受益。在此谨致以诚挚的敬意和衷心的感谢!

感谢给予我帮助的所有老师、朋友和同学们。

感谢我的家人,是他们给予我无限的支持与关心,希望他们能看到我的进步与成绩。

基于J2EE构架技术的高校网络办公系统的设计与实现

作者：[董敏](#)

学位授予单位：[山东大学](#)

本文链接：http://d.g.wanfangdata.com.cn/Thesis_Y1349136.aspx