

## 摘 要

企业步入信息化时代,企业的管理行为和管理模式如果不实行现代化、信息化,则企业很难在信息时代保持灵敏的嗅觉和高度的灵活性和高效性。企业管理信息化的最低要求是企业可以通过互联网获取、发布和处理企业的信息。而企业管理的关键是实现对人的管理。为了实现对人科学、高效的管理,进而实现对其它资源的科学管理和高效支配,从而使企业创造最高的效益,我们设计并开发了这套基于.net 技术的电子化人力资源管理系统 eHR,它是以现代企业人力资源管理理念为理论基础而开发的基于互联网的现代化企业人力资源管理系统。eHR 的任务就是充分发挥人在企业发展中的作用,实现人力资源管理的系统性和整体性。论文紧紧围绕“以人为本”的宗旨,紧扣人力资源管理过程中和员工息息相关的部分,具体分析了实现人力资源管理时员工与企业决策者、员工与人力资源管理实施者以及员工与员工的关系,具体分析了管理者对人力资源实施管理的具体方法、理论依据和详细的实施过程。

本 eHR 系统根据处理信息主体的不同,将数据流按信息进行分类,分成系统信息、员工信息、专业信息、公共信息,在系统的实施过程中对应地将系统的功能模块分成四个子系统,它们分别是:系统设置与管理子系统、员工信息管理与服务子系统、专业 HR 管理工具与服务子系统、公共信息服务与管理子系统。用户通过 web 访问系统,系统内部各个子系统间可以很方便地交流信息,系统也提供了和外部交互的接口,增大了系统的可扩展性。

本 eHR 系统创造性地将办公自动化软件 OA 的部分功能和 HR 的功能相互结合起来,并集成为一体,使 eHR 系统功能更显得丰富、灵活和强大。其中 OA 的部分功能归类到公共信息服务与管理子系统中,包括文档的处理和邮件的接收和发送,企业员工可以不必脱离系统就可以方便地接收企业公共信息或个人信息并进行反馈,大大方便了员工与员工、员工与企业相互间的沟通。

在系统的安全方面除了有一整套安全方案以外,设计了基于 RBAC 的角色授权机制,将要访问的对象分解成对象元素,并对对象元素进行授权,拥有访问权限的角色才可以访问该对象元素。eHR 系统通过角色机制解决了管理系统中时常出现的权限的继承和权限互斥的问题;同时对关键密级数据设计了以 Rijndael 加密技术为基础的加密机制,企业用户可以根据需要订制需要保密的数据部分,实现系统中数据的安全有效的保证。

包括 asp.net WEB 页面设计技术、c#后台代码设计语言和 ado.net 数据库访问技术的 .net 技术软件平台为 eHR 系统的开发和实施提供了友好、高效、安全的保证。

总之，本论文以基于.net 技术的人力资源管理系统的设计、开发和实施为例，深入探讨和研究了大中型流程企业实现电子化人力资源管理系统解决方案,实践证明这一套方案为实现企业人力资源管理的信息化、决策的科学化提供了很好的保障。

**关键词：**EHR、、ERP、.NET、角色、加密技术、存储过程

## Abstract

With the development of information technology in enterprises, if the management behaves and models in enterprises can't be modernized and informationized, it is very difficult to make the enterprises keep smart nose and high-level flexibility and efficiency. The minimum requirement for infomationization in business management is that the enterprises can acquire 、 release and do with information on Internet. The key factor in business management is the management of people. For achieving scientific and effective management to people, and the next step, effective and scientific management and control to other resources, consequently creating the largest profits for the enterprises, we design and develop a networked human resource management system eHR based on .net technology. Being a modern system based on Internet, the system is implemented with many advanced enterprise human resource management ideas. The main task of eHR is to adequately develop the functions of humans in enterprise progress, to realize the systematization and integrality of human resource management. After analyzing the relations between employees and decision-maker, between employees and executer of human resource management, among the employees themselves, the paper discusses the theory foundation, method of system analysis and design and detailed implementation process.

By the difference of information objects, the data stream is divided four parts: system information, employee information, professional information, and public information. Accordingly, the system is divided four sub-system according to its function modules: system configure and management sub-system, employee information management and service sub-system, professional HR management tools, common information service and management sub-system. Users can access system by web and sub-systems can exchange information expediently with each other. The system also provide external interface and enlarge system extensibility.

Innovative integration with partial functions of OA software makes the functions of eHR system more plentiful, flexible and powerful. A part of OA functions, such as document processing and email sending and receiving, are classified into common information service and management sub-system. Users can receive public enterprise information or private information and reply conveniently without leaving system. It

greatly improve communication ability among employees, between employees and enterprise.

About the security, except a suit of integrated scheme, the system implements role based access control. RBAC is a kind of authorization model that get more and more concerns. RBAC model organizes authorization into roles and users execute their rights by acquiring role member. The RBAC reduces complexity of authorization management greatly. At the same time, by implementing the RBAC model, the system successfully solves the problem of authorization inheritance and authorization conflict that often happens in many information systems. To critical secret data, the system adopts encryption mechanism based on Rijndael encrypting technology. User can encrypt the secret data according to their requirements and realizes the data security and validity in the system.

Asp.net web page design technology, c# background code design language, ado.net database access technology, and other technologies provide a friendly, effective, security developing and design validation to the system.

In a word, by using the example of design, development and implementation of a human resource management system based on .net technology, the paper presents a in-depth discuss and research of the solving scheme about how to realize electronic human resource management system in middle and large enterprises. The practical application shows the scheme provides an excellent assurance for enterprises to realize human resource management infomationization and scientific decision.

**Keyword:** EHR、ERP、.NET、Role 、Encryption、Stored Procedures

# 华南理工大学

## 学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律后果由本人承担。

作者签名：

日期： 年 月 日

## 学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并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。本人授权华南理工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、缩印或扫描等复制手段保存和汇编本学位论文。

保密☐，在\_\_\_\_年解密后适用本授权书。

本学位论文属于

不保密☐.

(请在以上相应方框内打“√”)

作者签名：曾晓明

日期：2005年6月5日

导师签名：陈国治

日期：2005年6月2日

# 第一章：绪论

## 1.1 课题背景

信息化是我国产业优化升级和实现工业化、现代化的关键环节，在我国企业发展的过程中必须把推进国民经济和社会信息化放在优先位置。实现社会生产力的跨越式发展，也必须大力推进国民经济和社会信息化，以信息化带动工业化，发挥后发优势。

企业信息化是指企业利用现代信息技术，通过信息资源的深入开发和广泛利用，不断提高生产、经营、管理、决策、服务的效率和水平，进而提高企业经济效益和企业竞争力的过程。具体到一个企业，企业信息化就是要实现企业生产过程的自动化、管理方式的网络化、决策支持的智能化和商务运营的电子化。通过企业信息化，可以全面提高企业的管理水平和经济效益，增强企业的核心竞争力和参与国际竞争的能力。

以网络经济为主导的新经济形势下的现代企业之竞争愈来愈直接地反映为人才战略的竞争。在此背景下，企业管理的转型，尤其是人力资源管理的转型就日显重要。为适应快速变化的市场，企业需要更加灵活、快速反应的人力资源管理平台和解决方案。“人力资源管理”已真正成为企业管理炙手可热的话题。无论从形式还是从内容上，企业的确是越来越重视人力资源的管理与开发。

目前，相当多的企业或政府人力资源部门的信息处理手段和信息管理方式主要通过简单的人事管理系统，结合文档、表格处理软件来完成，已经在一定程度上影响了人力资源基础工作的效率和流程化进程，迫切需要借助信息技术和现代网络条件，通过建立先进、实用、基于网络的信息系统，提高工作效率和业务流程的规范性，并在此基础上实现人力资源管理工作自动化、系统化，为企业高层提供基于信息的决策支持。

基于以上背景，一套专业的、覆盖企业全面人力资源管理工作的人力资源管理系统（HRMS）参与企业科学管理的整体运作成为企业信息化不可缺少的一部分，e化的人力资源管理系统(e-HRMS，简称 eHR)的实施更有利于全面提高人力资源管理工作效率、有利于实现符合企业发展要求的人力资源管理目标。

## 1.2 课题意义

通过上一节对研究背景的简单分析，可以看到研究基于电子信息技术下的人力资源管理系统是一件非常有意义的事情。

首先，从社会发展的角度来说，软件系统的实施有助于加快企业信息化的步伐，同时为促进整个社会的信息化和社会进步作贡献。软件系统的普及率是一个社会是否达到信息化的重要参考，信息化对整个社会进步和生产力的提高的重要作用是有目共睹，没有良好的硬件环境作基础，就没有软件系统兴旺发达的基础，同样光有先进的硬件环境却没有优秀的软件作支撑，硬件就会变成好看不中用的摆设，于个人、企业都没有什么益处；正是大量优秀的软件被广泛地使用，推动了硬件的不断更新换代，为社会创造了巨大的价值。

其次，从管理的角度来看，eHR 系统在企业的全面实行可以改变企业的运行模式，提高企业的工作效率，增强企业竞争力。eHR 软件系统是人力资源管理理论和计算机技术相结合的产物，实践证明他是科学可行的，大量优秀的 eHR 软件系统的被使用，对于规范企业行为、合理利用资源、改善员工与企业的关系以及决定企业高层的科学决策等等方面取得了越来越重要的作用。

再次，从技术的层面来讲，本 eHR 系统采用 .net 技术进行开发，.net 技术是目前商业软件开发使用最广泛的技术之一，.net 技术开发应用软件系统不仅高效、功能强大、灵活，而且开发的系统界面友好、使用稳定、安全。另外，.net 技术带来了一种全新的商业模式。它将逐步转换今天依靠销售盒装软件的获利形式，而是把各种软件改造成为能够通过“订阅”方式使用的服务。本课题的研究是 .net 技术很好的一次应用实践。

### 1.3 主要研究的内容

本课题着眼于建立一个比较完善的基于网络的开放式 eHR 管理平台，目的在于实现企业人力资源的信息化，该平台拟实现以下功能要求：

- 1) 公共信息的实时发布和浏览
- 2) 邮件的发送和接收，办公文档的实时发布和传送；
- 3) 企业对员工基本信息的高效管理，员工根据自助系统及时准确地了解相关信息，包括考勤、薪酬、福利保险、休假等。
- 4) 人力资源管理人员能利用专业的人力资源管理工具如素质测评、职位评估、绩效评估、人力资源规划、专业测试等实现对企业人力资源高效、科学地管理，以提高工作效率。

### 1.4 论文结构

论文的结构有下面几部分构成：

第一章：绪论，包括课题背景、课题意义、主要研究的内容、论文结构。

第二章：eHR 综述介绍了 eHR 的概念、eHR 的发展历程以及 eHR 与 ERP 的

关系。

第三章：基于.NET 技术的 eHR 系统的总体分析与设计，主要介绍 eHR 系统开发的拓扑结构和系统的体系结构，介绍了 eHR 软件系统的设计特点、功能模块组成以及开发所要求的硬件环境和软件平台。

第四章：基于.NET 技术的 eHR 系统的详细设计与实现，主要介绍了 eHR 系统的四个子系统的详细设计过程和实现过程，详细介绍了基于 RBAC 的角色授权机制及在 eHR 系统中的应用和实现；详细阐述了职位评估功能模块在 eHR 子系统中的应用；详细介绍了实施 eHR 过程中的安全问题及安全策略；详细介绍了 Rijndael 加密技术在 eHR 系统中的应用和实现。

第五章：eHR 系统的实际应用，主要介绍了实践使用中的 eHR 系统的界面和功能。

第六章：总结与展望



## 第二章：eHR 综述

### 2.1 eHR 的概念

#### 一、eHR 的发展背景

传统的人力资源管理中，人力资源工作人员往往必须仍旧花费大量的时间、精力在日常的行政事务性操作上，被繁琐的日常工作所束缚，无暇顾及更为重要的策略性工作。技术的进步最终解放了人力资源的“双手和大脑”，使人力资源工作发生了较大的变革。在 eHR 中，行政事务上的工作可以由电子化系统完成，只需占用极少的精力和时间。HR 人员们真正的工作重心可以放在服务员工、支持公司管理层的战略决策，放在公司最重要的资产——员工和员工的集体智慧的管理上，把绝大部分精力放在为管理层提供咨询、建议上。eHR 的最终目的是达到革新企业的管理理念而不仅仅是改进管理方式，

#### 二、eHR 的定义

所谓 eHR(Electronic Human Resource Management System -EHRMS，简称 eHR)是指人力资源管理电子信息化，是指企业基于高速度、大容量的硬件和先进的 IT 软件的人力资源管理模式。通俗地说，就是人力资源管理信息化或自动化。

#### 三、eHR 的作用

现在不少企业已经认识到采用 eHR 的必要性，人力资源管理信息化给企业带来的好处也是不言而喻的：组织效率的提升、完善体系制度、加强流程控制、实现量化管理、开支的节省、员工满意度的提高、员工能力的改进和企业利润的增加、过渡到以人为本的管理。在人力资源管理中，eHR 到底可以为企业做那些事情呢？其主要作用通常可以概括为六个方面：

- (1)、全面整合与管理企业人力资源信息
- (2)、实现“公平、公正、合理”原则，吸引并留住人才
- (3)、健全制度体系，提高管理水平
- (4)、有效实现企业人力资源管理的流程控制
- (5)、eHR-MS 实现企业人力资源量化管理
- (6)、eHR-MS 实现企业以人为本的管理

### 2.2 eHR 的发展历程

#### 2.2.1 历史演变

HRMS 系统的五个阶段：

一、第一代 HRMS：20 世纪 60 年代末期，HRMS 的诞生。

第一代 HRMS 功能非常简单，只不过是一种自动计算薪资的工具，但是它的出现具有非常重要的意义，它为人力资源的管理展示了美好的前景。

二、HRMS：20 世纪 70 年代末，HRMS 功能逐步增强。

随着计算机技术的飞速发展，计算机的快速普及，计算机系统工具和数据库技术的发展，为 HRMS 的阶段性的发展提供了可能。第二代 HRMS 解决了第一代系统的主要缺陷，增加了较多的管理功能，诸如对非财务的人力资源信息和薪资的历史信息给予了考虑，然而却未能系统地考虑人力资源的需求和理念。

三、第三代 HRMS：20 世纪 80 年代末，先进的人力资源管理理念进入 HRMS。

引进先进的人力资源管理理念，从人力资源管理角度出发，成为企业加强人力管理的重要工具，将与人力资源相关的数据统一管理，形成了集成的信息源。随着全球经济化的浪潮，市场竞争加剧，人才成为企业最重要的资产之一，促使 HRMS 引进先进的人力资源管理理念。第三代 HRMS 从人力资源管理角度出发，成为企业加强人力管理的重要工具，将与人力资源相关的数据统一管理，形成了集成的信息源。HRMS 得到了飞速的发展，但是相比财务信息化的发展，HR 信息化程度已经明显落后。

四、第四代 HRMS：20 世纪 90 年代末，HRMS 发生革命性变革，运用网络技术，实现信息的实时共享。

90 年代末，随着企业管理理念和管理水平的大幅度提高，使社会对 HRMS 有了更高的需求；同时由于个人电脑的全面普及，数据库技术、客户/服务器技术，特别是 Internet/Intranet 技术的发展，促使 HRMS 发生革命性变革。第四代 HRMS 已经开始运用网络技术，实现信息的实时共享，第四代 HRMS 对网络的应用还有限，功能还不够强大。

五、第五代 HRMS：21 世纪初，智能化和电子化 HRMS(e 化 HRMS 即 eHR)。

进入 21 世纪，伴随信息化的普及及互联网的快速发展，企业需要思想、技术更为先进的 HRMS。第五代 HRMS，也即 IHRMS（智能化 HRMS）和 eHRMS（eHR），它们紧密联系企业人力资源管理实际，同时充分利用信息网络，从根本上改变了员工与企业的沟通方式。第五代 HRMS 能够为企业提供更人性化的管理模式，为企业提供更大量的决策信息。

## 2.2.2 技术演变

从 IT 技术的角度来讲，eHR 自身的发展基本经历了三个阶段：

第一阶段是主机/工作站技术阶段；

第二阶段是客户机/服务器技术阶段；

第三阶段是互联网技术阶段。

以网络为基础的 B/S 模式、面向对象的 eHR 系统从系统实施的角度看，比 C/S 模式易于安装和实施，同时系统易于使用，支持多媒体，而且费用较低。

## 2.3 eHR 与 ERP 的关系

### 2.3.1 ERP 定义

ERP—Enterprise Resource Planning 企业资源计划系统，是指建立在信息技术基础上，以系统化的管理思想，为企业决策层及员工提供决策运行手段的管理平台，包括财务会计、管理会计、生产计划及管理、物料管理、销售与分销、人力资源管理等主要功能模块。

### 2.3.2 ERP 的功能

其主要功能为：

- (1)收集、分析公司业务中产生的信息和资料；
- (2)决策者能萃取决策所需的数据；
- (3)能让决策者聚集数据并对数据进行处理；
- (4)在制定决策时能从不同角度研究数据或观察数据间的关系以便深入了解问题；
- (5)提供各种精确的数据，以满足不同决策者及决策作业的要求；
- (6)可满足不同层次用户的多方面、多阶段、多层次的智能过滤服务；
- (7)具有决策、控制和预测功能，能在复杂多变的外部环境中，辅助人作出科学的决策；
- (8)最后，ERP 将从面向事务处理的管理模式向面向智能、优化管理模式的方向发展，它可使管理人员按照设定的目标去寻找一种最佳的方案并迅速执行，这样就可紧紧跟踪、甚至可达到超前于市场的需求变化，快速作出正确的决策、并以最快的速度执行这些变化。

### 2.3.3 EHR 和 ERP 的发展趋势及相互关系

在构建企业信息化管理系统时，有的企业认为，既然企业已经部署了较为完善的 ERP 系统，为什么还要专门投资部署 EHR 系统呢？另外还有的企业认为，部署了 eHR 会成为投资的浪费，将来可以一步到位“直接上 ERP”。

其实，eHR 与 ERP 并不冲突，首先对于企业来说，企业的每一个流程环节，都会涉及到人力资源的利用与优化，eHR 实施的效果如何将直接关系到企业 ERP 建设的成败。

其次，在企业 ERP 的构建过程中，eHR 系统将会给 ERP 系统的实施提供相

应的信息资源, 如员工信息、机构信息、人工直接成本、职位鉴定、培训信息等。

再次, 人力资源的规划、行为规范、绩效考核以及成本分析本身就是企业 ERP 规划的内容。

因此, 企业可以首先把涉及人力资源管理的系统功能在 eHR 系统中建立和完善, 然后, 在现有成熟的 eHR 管理系统下, 使企业尽快纳入 ERP 流程化、标准化管理, 从而使企业 ERP 的实施难度大大降低。

考虑到 eHR 系统与 ERP 等系统的集成问题, 如果企业能够采用一套统一的系统集成问题会比较容易解决。至于先上 eHR (MS) 还是与 ERP 一起上, 客户可以根据自身的需求决定, 但关键是企业在实施信息化过程中应该有一个统一的规划, 然后再分布实施, 这样才能形成一个完整的体系。

## 2.4 eHR 系统体系结构

### 2.4.1 软件体系结构

无论是何种开发方式, 在技术层面上, 建立一个系统从宏观层面来看第一要著是使系统结构化, 结构化的途径有三个: 分层、管道和过虑、黑板<sup>[1]</sup>。

分层模式是一种解决大系统的良好方法。本质上讲分层是抽象的排列。因为大系统复杂性较高, 它的抽象层面也相应较多。

管道和过虑通过用来解决数据流的较好方式。

黑板则是用来解决不确定性问题的首要途径。当一个问题解决方法缺乏一定模式, 而现有的方法集中的每个方法都只能解决问题的一个部分, 并各自不相交。黑板可以成为较好的抽象机制。

### 2.4.2 三层体系结构

#### 一、传统两层结构

在过去应用系统开发过程中, CLIENT/SERVER 体系结构得到了广泛的应用。其特点是, 应用程序逻辑通常分布在客户和服务端两端, 客户端发出数据资源访问请求, 服务器端将结果返回客户端。但 CLIENT/SERVER 结构存在着很多体系结构上的问题, 比如: 当客户端数目激增时, 服务器端的性能会因为负载过重而大大衰减; 一旦应用的需求发生变化, 客户端和服务端的应用程序都需要进行修改, 给应用维护和升级带来了极大的不便; 大量的数据传输增加了网络的负载等等。

#### 二、三层结构介绍

所谓三层体系结构, 是在客户端与数据库之间加入了一个“中间层”, 也叫组件层。这里所说的三层体系, 不是指物理上的三层, 不是简单地放置三台机器就

是三层体系结构，也不仅仅有 B/S 应用才是三层体系结构，三层是指逻辑上的三层，即使这三个层放置到一台机器上。这三层可以抽象为：表示层、应用层、数据层。

三层体系的应用程序将业务规则、数据访问、合法性校验等工作放到了中间层进行处理。通常情况下，客户端不直接与数据库进行交互，而是通过 COM/DCOM 通讯与中间层建立连接，再经由中间层与数据库进行交互。

### 三、 eHR 系统中 Asp.Net(C#)+Sql Server 三层架构方案

ASP.NET 可以使用 .NET 平台快速方便的部署三层架构。ASP.NET 比较先前版本的变化是在网页中也使用基于事件的处理，可以指定处理的后台代码文件，可以使用 C#,VB,J#作为后台代码的语言。.NET 中可以方便的实现组件的装配，后台代码通过命名控件可以方便的使用自己定义的组件。显示层放在 ASP.NET 页面中，数据库操作和逻辑层用组件来实现，这样就方便的实现了三层架构。

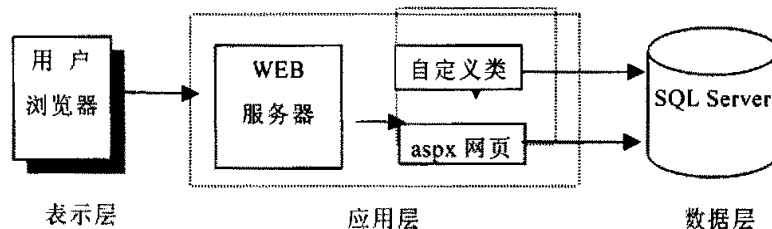


图 2-1 eHR 系统体系结构

## 2. 5eHR 系统 .NET 软件开发平台

### 2. 5. 1. NET 平台

Microsoft .NET 是 Microsoft XML Web services 平台<sup>[2]</sup>。XML Web services 允许应用程序通过 Internet 进行通讯和共享数据，而不管所采用的是哪种操作系统、设备或编程语言。Microsoft .NET 平台提供创建 XML Web services 并将这些服务集成在一起之所需。对个人用户的好处是无缝的、吸引人的体验。

Microsoft.NET 扩展了通过任何设备随时随地操作数据和进行通讯的能力，Microsoft.NET 使用分布式计算模型并基于开放标准（如 XML）将 PC 与其他智能设备连接在一起。Microsoft .NET 包括用于创建和操作新一代服务的 .NET 基础结构和工具、用于实施多信息客户端的 .NET 用户体验，以及用于启用新一代智能 Internet 设备的 .NET 构造块服务和 .NET 设备软件，其产品和服务，包括 Microsoft® Windows.NET（其核心集成了一组构造块服务）、MSN.NET、个

人订阅服务、Microsoft® Office.NET、Microsoft® Visual Studio.NET 和 Microsoft® bCentral™ for .NET。

## 2.5.2 .NET Framework 框架

.NET Framework 是支持生成和运行下一代应用程序和 XML Web services 的内部 Windows 组件。.NET Framework 旨在实现下列目标：

- (1) 提供一个一致的面向对象的编程环境，而无论对象代码是在本地存储和执行，还是在本地执行但在 Internet 上分布，或者是在远程执行的。
- (2) 提供一个将软件部署和版本控制冲突最小化的代码执行环境。
- (3) 提供一个可提高代码（包括由未知的或不完全受信任的第三方创建的代码）执行安全性的代码执行环境。
- (4) 提供一个可消除脚本环境或解释环境的性能问题的代码执行环境。
- (5) 使开发人员的经验在面对类型大不相同的应用程序（如基于 Windows 的应用程序和基于 Web 的应用程序）时保持一致。
- (6) 按照工业标准生成所有通信，以确保基于 .NET Framework 的代码可与任何其他代码集成。

.NET Framework 具有两个主要组件：公共语言运行库和 .NET Framework 类库。

其组成如图 1 所示：

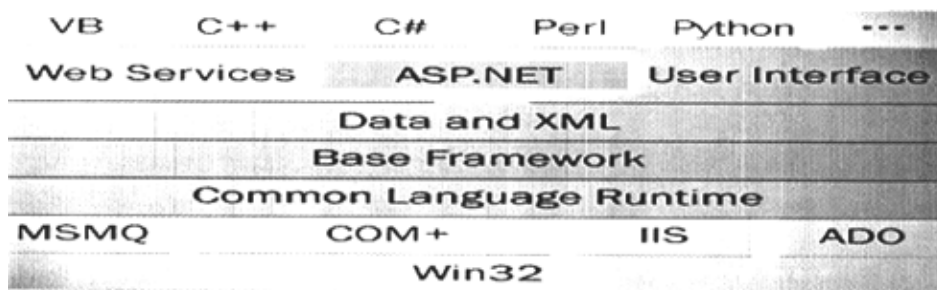


图 2-2 .NET Framework 由 .NET 运行时和类库组成

### 一、通用语言运行环境

公共语言运行库是 .NET Framework 的基础。您可以将运行库看作一个在执行时管理代码的代理，它提供内存管理、线程管理和远程处理等核心服务，并且还强制实施严格的类型安全以及可提高安全性和可靠性的其他形式的代码准确性。事实上，代码管理的概念是运行库的基本原则。以运行库为目标的代码称为托管代码，而不以运行库为目标的代码称为非托管代码。

为 .NET 框架上的应用程序提供通用的服务。可以用差不多任何一种语言编写程序, 包括 C、C++、C# 和 Microsoft Visual Basic, 还有较早的语言如 Fortran。这个环境协助完成许多最基本的编码任务。如内存管理、安全管理和出错处理等。

## 二、.NET 框架类库

它是一个综合性的面向对象的可重用类型集合, 您可以使用它开发多种应用程序, 这些应用程序包括传统的命令行或图形用户界面 (GUI) 应用程序, 也包括基于 ASP.NET 所提供的最新创新的应用程序 (如 Web 窗体和 XML Web services)

开发人员可能使用这个库中现成的功能组件, 快速扩充软件功能。这个库主要有三个关键组件:

ASP.NET ——用来开发网络应用程序和网络服务程序。

Windows 表单——可帮助开发“智能客户端”接口。

ADO.NET ——负责处理程序与数据库之间的连接

## 三、.NET 框架的优点

.NET 框架的主要优点有:

- (1) 、助软件使用者与网络服务商部署软件;
- (2) 、提供更稳定、更安全、更易扩容、更高性能的软件开发环境;
- (3) 、让开发人员更有生产力, 更易使用现有代码; 使他们更轻松地集成所支持的 20 多种编程语言编写出来的组件, 让他们用同样的工具和技能开发出更多样化的软件。

### 2.5.3 NET 互联网 WEB 页面设计技术

ASP. NET 是建立在通用语言运行环境 CLR(COMMON LANGUAGE RUNTIME)上的, ASP. NET 是 .NET 框架应用程序的执行引擎, 可以在服务器端构造一个强大的 Web 应用程序的编程框架。这是微软公司对 ASP. NET 的定义。它表明了两点意思: ASP. NET 的运行环境是通用语言运行环境; ASP. NET 是 .NET 框架的一个组成部分, ASP.net 提供许多比现在的 Web 开发模式强大的优势<sup>[1]</sup>。

(1) 执行效率的大幅提高 ASP.net 是把基于通用语言的程序在服务器上运行。不像以前的 ASP 即时解释程序, 而是将程序在服务器端首次运行时进行编译, 这样的执行效果, 当然比一条一条的解释强很多。

(2) 世界级的工具支持 ASP.net 构架是可以用 Microsoft (R) 公司最新的产品 Visual Studio.net 开发环境进行开发, WYSIWYG (What You See Is What You Get 所见即为所得) 的编辑。这些仅是 ASP.net 强大化软件支持的一小部分。

(3) 强大性和适应性 因为 ASP.net 是基于通用语言的编译运行的程序,所以它的强大性和适应性,可以使它运行在 Web 应用软件开发者的几乎全部的平台。通用语言的基本库,消息机制,数据接口的处理都能无缝的整合到 ASP.net 的 Web 应用中。ASP.net 同时也是 language-independent 语言独立化的,所以,你可以选择一种最适合你的语言来编写你的程序,或者把你的程序用很多种语言来写。

(4) 简单性和易学性 ASP.net 是运行一些很平常的任务如表单的提交客户端的身份验证、分布系统和网站配置变得非常简单。例如 ASP.net 页面构架允许你建立你自己的用户分界面,使其不同于常见的 VB-Like 界面。另外,通用语言简化开发使把代码结合成软件简单的就像装配电脑。

(5) 高效可管理性 ASP.net 使用一种字符基础的,分级的配置系统,使你服务器环境和应用程序的设置更加简单。因为配置信息都保存在简单文本中,新的设置有可能都不需要启动本地的管理员工具就可以实现。这种被称为"Zero Local Administration"的哲学观念使 Asp.net 的基于应用的开发更加具体,和快捷。一个 ASP.net 的应用程序在一台服务器系统的安装只需要简单的拷贝一些必须的文件,不需要系统的重新启动。

(6) 多处理器环境的可靠性 ASP.net 已经被刻意设计成为一种可以用于多处理器的开发工具,它在多处理器的环境下用特殊的无缝连接技术,将很大的提高运行速度。

(7) 自定义性和可扩展性 ASP.net 设计时考虑了让网站开发人员可以在自己的代码中自己定义"plug-in"的模块。这与原来的包含关系不同,ASP.net 可以加入自己定义的任何组件。网站程序的开发变得比以前简单。

(8) 安全性 基于 Windows 认证技术和应用程序配置,可以确保你的源程序的高度安全。

## 2.5.40# 后台代码技术

C#是一种先进,面向对象的语言,通过 C#可以让开发人员快速的建立大范围的基于 MS 网络平台的应用,并且提供大量的开发工具和服务帮助开发人员开发基于计算和通信的各种应用。由于 C#是一种面向对象的开发语言,所以 C#可以大范围的适用于高层商业应用和底层系统的开发。即使是通过简单的 C#构造也可以各种组件方便的转变为基于 WEB 的应用,并且能够通过 Internet 被各种系统或是其他开发语言所开发的应用调用。即使抛开上面所提到的优点,C#也可以为 C/C++开发人员提供快速的开发手段而不需要牺牲任何 C/C++语言的特点/优点。从继承角度来看,C#在更高层次上重新实现了 C/C++,熟悉 C/C++开发的人员可以很快的转变为 C#开发人员。



从开发语言的角度来讲 C#可以更好帮助开发人员避免错误，提高工作效率，而且同时具有 C/C++的强大功能。同时 C#它能完全地符合 .NET 框架。C#(C-Sharp)，它和 .NET 框架一起发布。C#将是很多 .NET 应用程序开发人员的首选语言<sup>[4]</sup>。

## 2.5.5 ADO .NET 数据库访问技术

ADO.NET 是对 Microsoft ActiveX Data Objects (ADO)一个跨时代的改进，它提供了平台互用性和可伸缩的数据访问<sup>[5]</sup>。由于传送的数据都是 XML 格式的，因此任何能够读取 XML 格式的应用程序都可以进行数据处理。事实上，接受数据的组件不一定要是 ADO .NET 组件，它可以是基于一个 Microsoft Visual Studio 的解决方案，也可以是任何运行在其它平台上的任何应用程序。在当今的所流行的各种管理系统中，我们常常使用数据库来存储数据。而且我们可以轻易的发现，使用 DBMS（数据库管理系统）是存储和处理应用程序数据的最简易方式。但对于不同的 DBMS，我们如何能够编写通用的计算机程序，使之互相兼容。微软给我们带来了最新的解决方案 ADO.NET 数据访问系统。

### 2.5.5.1 ADO.NET 的体系结构

ADO.NET 中最主要的两个组件是 .NET 数据提供者和数据集对象。它们之间的关系相辅相成，互相服务。数据集可以使用数据提供者的服务来检索信息。提供者提供足够长的连接时间让数据集检索数据。其关系如图

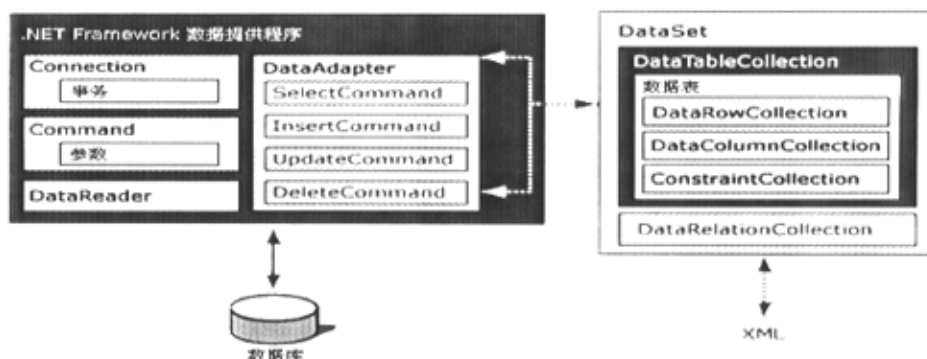


图 2.3.NET 数据提供者和数据集对象关系

### 2.5.5.2 ADO.NET 与 ADO 的比较

微软在提出 ADO.NET 前提出了 ADO(ActiveX Data Objects)。它是 DAO 和 RDO 的产物。它稳定、可靠且得到了广泛的支持，几乎所有的 Microsoft 产品都支持它。但是它也有自己的不足，主要分为以下几点：

- (1)、 一次只能管理一个表或一组记录
- (2)、 自身不支持文本传输 XML
- (3)、 不能简单的创建表间的关联
- (4)、 主要用于可与数据库直接连接的系统
- (5)、 数据通讯是基于 COM，平台无关性不好
- (6)、 消耗系统资源高
- (7)、 不利于穿越防火墙

ADO.NET 的到来解决了以上这些问题。它具有可断开连接的特性，在这里使用 DATASET 对象，它可以在程序与数据库进行连接的时候，将数据内容读入自己的体内（以下统称为数据集），在程序与数据库断开连接的时候，数据集依然可以很好的保存数据。

ADO.NET 的其它优点：

- (1)、ADO.NET 数据集包含表的集合以及表间的关系，因此可以处理更为复杂的数据结构
- (2)、ADO.NET 数据集提供了基于表的关系视图和基于 XML 的分层视图
- (3)、ADO.NET 可以穿越防火墙
- (4)、效率比 ADO 高
- (5)、版本兼容性比 ADO 好

## 2.6 小结

1.ADO.NET 是用于 .Net 框架的 ADO 下一代版本。

2.ADO.NET 是以 N 层、无状态性和 XML 为突出重点来创建的。这些方案中提供了两个新对象：DataSet 和 DataAdapter。

3.ADO.NET 可用于从流中获取数据，或将数据存储于缓存中以便更新。

4.可以直接对数据库执行命令，以进行插入、更新和删除操作。无需先将数据放入 DataSet 中即可对其进行插入、更新或删除操作。此外，可以使用 DataSet 来绑定到数据、在数据间移动和定位数据关系。

## 第三章：基于 .NET 技术的 eHR 系统的总体分析与设计

### 3.1 eHR 系统总体设计特点

#### (1) 完全基于 Web 的操作模式

基于 Web 的多层结构开发，数据集中管理，可实现信息的高度共享；无论组织机构如何复杂、办公地点如何分散，都能以浏览器方式提供准确、实时和有效的数据；系统的安装和维护仅限于服务器端，客户端无需安装；

#### (2) 方便易用

人性化的用户界面简单直观，您只需浏览器就可轻松完成一切工作；

#### (3) 数据处理和事务的流程化

将日常的人力资源管理工作以表格和流程体现，以帮助用户规范人力资源管理；

#### (4) 文档处理与 HR 管理工具的有机结合自成一体

本系统的特点之一是将文件处理和人力资源管理工具有效结合在一起，不仅可以实现对员工的基本管理，员工内部也可以在此系统上进行相应的信息交流。

#### (5) 高效的实用数据导入导出工具

为帮助客户快速、高效地搭建人力资源的信息平台，我们针对系统开发了辅助导入导出工具，能在单位已有的数据基础上快速地导入人员信息和其它数据和信息，另外为了安全和方便的需要，对拥有相应权限的用户还可以将系统内部的信息导出成其它格式的数据，以供备份或打印所需；

#### (6) 强大的报表功能

无需下载任何插件，即可支持浏览器打印报表，并可支持复杂格式的统计、分析报表的定制，实现互联网环境下的“所见即所得”打印；

#### (7) 严格的安全管理机制

提供了基于 RABC 的用户角色授权机制，有效地保证了系统的的天性，对重要的数据则进行加密处理和相应的备份策略，保证了数据的安全性。

### 3.2 eHR 系统各功能子系统功能介绍及其设计

#### 3.2.1 eHR 系统的网络拓扑结构

基于 .net 技术的 eHR 系统是构建于 Intranet/Internet 网络之上的，分布性是 eHR 系统最突出的拓扑结构的特点，主要表现为：

- 1) 企业内部的数据服务器、应用服务器是分布在 Intranet/Internet 网络的不同节点上的, 有利于实现负载平衡;
  - 2) 数据服务器可以有分布与子系统内的多台数据库服务器组成;
  - 3) 不同分支机构的系统一般来说都是在地理上分布的。
- 其网络拓扑结构如图 3-1 所示:

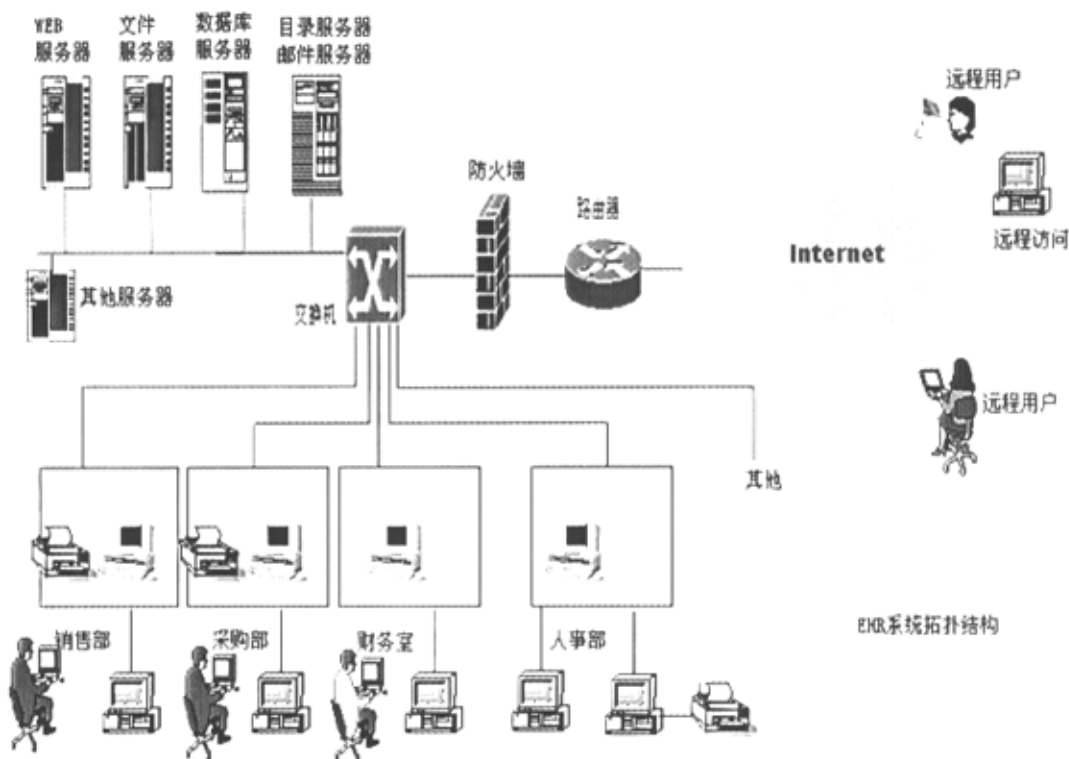


图 3-1 eHR 系统的网络拓扑结构

### 3.2.2 eHR 系统的功能结构及其组成

基于.NET 技术的人力资源管理系统将整个系统按照数据处理的主体的相似性的特点, 将系统集成四个既相互独立又相互关联的功能模块, 每一个模块又有对应的子功能模块构成。其结构图如图 3-2:

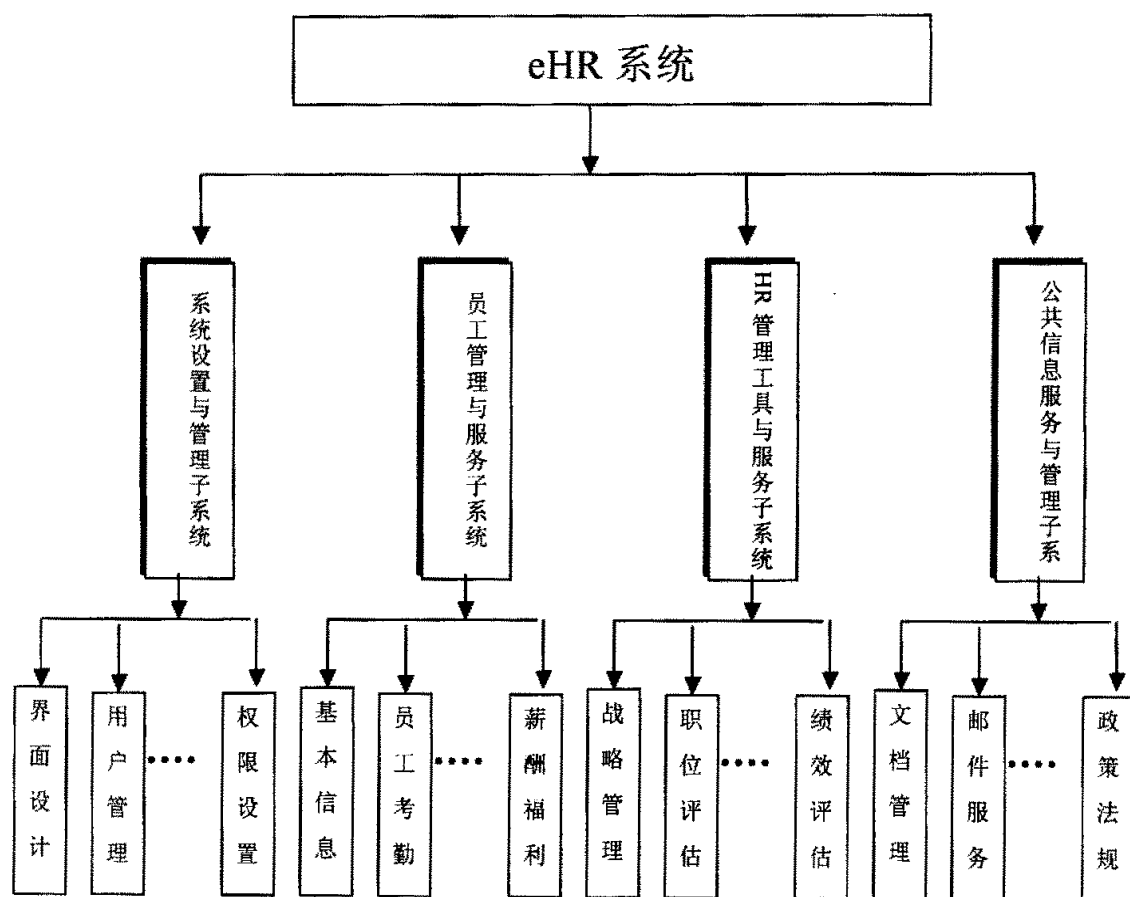


图 3-2 eHR 系统的功能结构及其组成图

### 3.2.2.1 eHR 之系统设置与管理子系统 (SSM)

对影响整个系统运行的相关部分进行统一的设置和管理，这些部分主要是指用户管理、权限设置、角色管理、部门设置和职务职能管理等。其功能模块如下图 3-3：

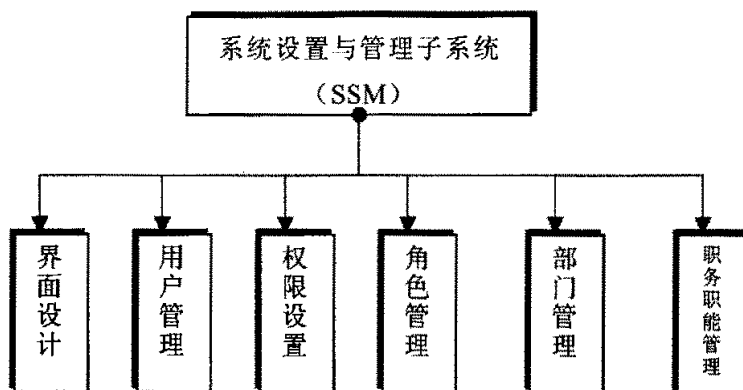


图 3-3 e-HR 之系统设置与管理子系统结构图

### 1. 界面设计

采用动态网页程序设计技术对系统的整体和部分对象及对象元素进行统一设计和规划，主要完成以下功能：

- (1) 设计人力资源系统使用的所有关键界面样式与字段；
- (2) 根据本公司或企业的具体业务与要求为用户定义数据结构与界面系统。

### 2. 用户管理

对所有的用户进行系统级统一的管理，包括：

- (1) 新用户的增加管理

新用户的注册、注册信息的审批确认（身份确认、部门确认、角色确认），注册审批成功后根据部门和角色的类型分配相应的权限；

- (2) 已注册用户的信息变更管理

已注册的销户管理

已注册关键注册信息变更审批管理

### 3. 权限设置

定义和设置系统权限的类别、每一组类别拥有所有每一项权限的名称以及每一项权限的具体内容说明；

### 4. 角色管理

定义用户角色的名称，拥有权限的类别，授予用户使用系统的权限组合，一个角色对权限可以自动获得、可以继承、也可以转移权限。

### 5. 部门管理

设置和规划企业的组织机构，包括顶级机构、一级机构、总部和分部机构等，组织机构的组成部分可以新建、可以添加、可以删除和修改。

### 6. 职务职能管理

- (1)按组织机构设定职务体系，包括职类、职种、职位、职层、职等、职级
- (2)按组织机构设定职位体系，并以职位等级划分各种方式的职位晋升道路，最终可为员工描绘一个具体、形象的职业发展远景，同时也协助用户建立了一套以组织架构为基础，以职位等级为基准的、规范的、标准的职位体系；
- (3)工作要求、目标、职责以及衡量标准等细致描述了一个职位的诸多方面，并成为招聘模块相应职位要求的可靠依据；
- (4)针对不同职位划分相应的职务级别，也便于管理梯队的定位、跟踪和管理；
- (5)监测公司各部门职位的动态配备状况，对缺编、超编情况随时分析、汇总，以便从各个层面了解公司的人力资源分布情况，为公司战略性目标的实施提供科学的数据支持和分析基础

### 3.2.2.2 e-HR 之员工信息管理与服务子系统（EMMS）

主要实现对与员工的利益息息相关的信息进行管理，如员工的基本信息的增加与删除、员工薪酬福利及保险信息、员工考勤情况、招聘培训信息等，另外还提供自主系统供员工实时地查看自己的相关信息。其功能模块如下图 3-4：

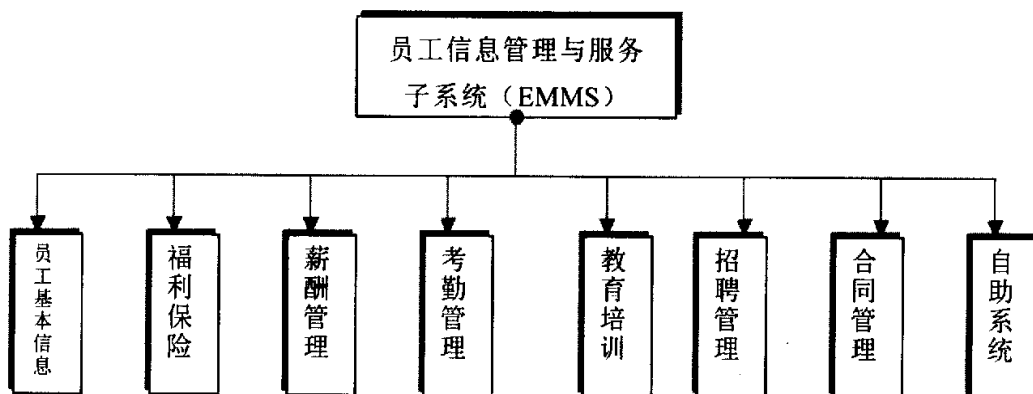


图 3-4 e-HR 之员工信息管理与服务子系统结构图

#### 1、员工信息管理与服务

实现对员工基本信息的管理，并对系统的其它部分提供相应接口，主要实现以下功能：

- (1) 员工档案基本信息
- (2) 员工（在职、兼职、离退）基本信息的管理，包括：增加、删除、修改；
- (3) 员工基本信息的快速浏览和打印（附近照）；
- (4) 近照、身份证和学历证书等图片资料的管理；

- (5) 提供员工履历表；
- (6) 在系统提供的标准人事信息基础上，用户可自行定义其他人事信息项目；
- (7) 便捷地浏览员工其他信息，包括：人事异动、薪金福利、培训与考核等；
- (8) 提供完整的信息记录，如：员工资源占用情况登记，以便及时了解公司资源的分布情况并在员工离职时给予相应的提示；
- (9) 根据用户定义可快速筛选，并定位所需人才，并提供包括基本、培训、考核等的详尽信息，便于人事部门向领导或直线经理推荐人才；
- (10) 员工花名册可获得各部门或整个公司人员的基本资料；
- (11) 员工生日提醒功能帮助人事部门或直线经理完成以前“无法完成的任务” -- 准确地记住每一个员工的生日，同时还可通过信息中心的人事信息发布系统告诉公司所有员工，实现规范、严格管理下的“亲情文化”

## 2. 福利保险

完成养老保险、医疗保险、失业保险、工伤保险、生育保险、住房公积金等保险福利的自动计算、缴纳（支持工资扣除）、提供个人帐户管理功能；并可自定义新的保险项目，对其进行自动计算、缴费、个人帐户管理。主要实现以下功能：

- (1) 支持多种福利政策，如：员工贷款、车 / 房贴等，并体现在薪资中，参与计算；
- (2) 支持集团化的福利保险管理模式，各法人单位可根据实际情况和地区差异执行不同的福利保险政策，而且互不干扰；
- (3) 支持社保的新增 / 转入 / 转出 / 代缴，同时也可处理停保、续保人员；
- (4) 根据需要设置参加社保人员，并设定社保的“三险一金”提取规则，以便系统能自动计算出各保险<的缴费基数。对计算结果，用户还可进行调整；
- (5) 可分别计算出社保的个人缴纳金额和企业缴纳金额，并将个人缴纳金额转入工资帐套的相应工资项目中。
- (6) 提供标准的社保报表所需的统计和分析项目，可大大降低计算和分析工作量。

## 3. 薪酬管理

主要实现以下功能：

- (1) 对工资帐套进行跨区域、跨公司的管理，支持集团化薪酬管理；
- (2) 根据不同员工类型（如：合同工、临时工、兼职、计时工、计件工、外籍员工、外包工）、区域（如：地区、分部、部门、子公司等）定义不同的薪资帐套，每一个薪资帐套可以定义不同的薪资项目、计算公式以及考勤规则；
- (3) 集团版支持集团化的薪资管理方案，各公司或部门可形成不同的核算单元，并互不影响，同时集团总部可全局掌控，并可了解各公司薪资状况；



- (4) 每个薪资帐套的考勤系统支持多达五种的加班类型;
- (5) 结合考勤管理,完成加班、病假、事假等的工资计算,并根据需要打印出工资条、工资表以及部门汇总表;
- (6) 提供对基础工资和月工资数据的批量调整;
- (7) 灵活设定工资计算公式和数据来源,并支持福利保险、绩效考核等模块的数据导入;
- (8) 任意设定工资计算时的计算精度,如:四舍五入、见尾数进位、舍尾数、见分进元;
- (9) 根据不同薪资地区设定相关的计税规则、三险一金(社保基金及住房公积金)缴纳比例、基数和计算公式;
- (10) 提供多种薪资报表和分析图表,如:工资表、工资条、工资汇总表、部门汇总表等;

#### 4. 考勤管理

实现员工考勤的管理,对员工的考勤数据进行计算、核对、校正后,将每月的考勤计算结果数据提供给工资管理子系统,参加工资计算。主要实现以下功能:

- (1) 提供标准的考勤机接口或专用考勤机接口的定制;
- (2) 灵活设定员工班次、公休假期等信息;
- (3) 根据员工班次和分析考勤数据,自动核定迟到、早退或旷工等员工出勤状况;
- (4) 对于异常情况,如忘记打卡或外出等,可单独处理;
- (5) 统计员工的出勤数据,并可查询员工出勤明细;
- (6) 月考勤统计数据自动转入薪资模块,计算相应的薪资加 / 减项;

#### 5. 教育培训

主要实现以下功能:

- (1) 用户可以对不同的培训类型进行不同的管理,如:入职培训、内部培训、岗位培训等;
- (2) 用户可将常用培训形成培训模板,以便对日常培训快速实施。模板中可包含培训目的、培训课程、费用、时间安排等信息;
- (3) 用户可通过选择培训模板来快速编制培训计划,也可以新建培训计划;
- (4) 所有培训经历都将作为历史数据予以保留,便于查询检索;
- (5) 授权用户可审批员工的培训申请,并可直接批复或转审;
- (6) 对培训可进行整体和部分的反馈调查评估,并结合学员的培训结果,有助于优化培训计划的设计;

#### 6. 招聘管理

主要实现以下功能:

(1) 从招聘职位建立、候选人筛选及面试，直至最终的到职处理，在人事部门的招聘经理和直线经理之间形成一个快捷、方便的直线招聘流；

(2) 针对不同职位的特点，可建立不同的招聘流程，并可根据实际需要将空缺职位发布到员工自助中心，以便内部员工自荐或推荐合适人选；

(3) 可根据用户的需要和职位的特点，定义所需的简历夹，方便简历的管理；

(4) 独特的国内主流人才（如：招聘网、前程无忧、中华英才、中国人才热线、南方人才网）和各行业人才网站（如：建筑英才网、IT 人才网）数据接口，可极其方便的获得所需的各方面人才资料；渠道主要有三种：参加人才交流会、人才交流中心介绍、刊登报纸广告。

(5) 快捷的简历筛选，可缩短简历的筛选时间，也可极大减轻招聘经理的筛选工作量；

(6) 根据各招聘职位的需要设定并记录面试，便于面试结果的对比分析；

(7) 通过了筛选、面试合格的候选人，可进行“员工到职”处理，将其简历自动转入正式人员信息库，而其他人员则作为后备人才储备。

#### 7. 合同管理

主要实现以下功能：

(1)根据国家劳动法律法规和企业劳动人事制度、岗位设置情况，草拟合同文本，或在原有合同版本基础上进行修订，建立新版本合同。

(2)能够对不同时期的不同合同版本进行管理。

(3)能够进行新签合同、合同续签、合同终止、合同变更、补充合同等合同管理业务的处理。

(4)建立劳动争议台账，对劳动争议事由和处理结果、经济补偿和经济赔偿事由和处理结果进行记载。

(5)可以设置基于时间的合同管理事务预警处理。

(6)提供多种查询、统计和分析功能，可以对合同信息进行统计分析，产生相关的各类报表。

#### 8. 自助系统

主要实现以下功能：

(1) 员工可查看个人信息，并管理和维护部门信息。这样，既可减轻人事部门对个人信息维护量，又确保了员工信息的时效性和准确性，特别是经常变化的信息，如：联系方式、通讯地址等；

(2) 员工可获得个人的工资、保险、考勤、合同、休假、培训、绩效等详细信息；

(3) 员工可进行网上的年假申请，随时查看审批结果，并了解其他员工的休假情况，以便安排自己的工作进程；

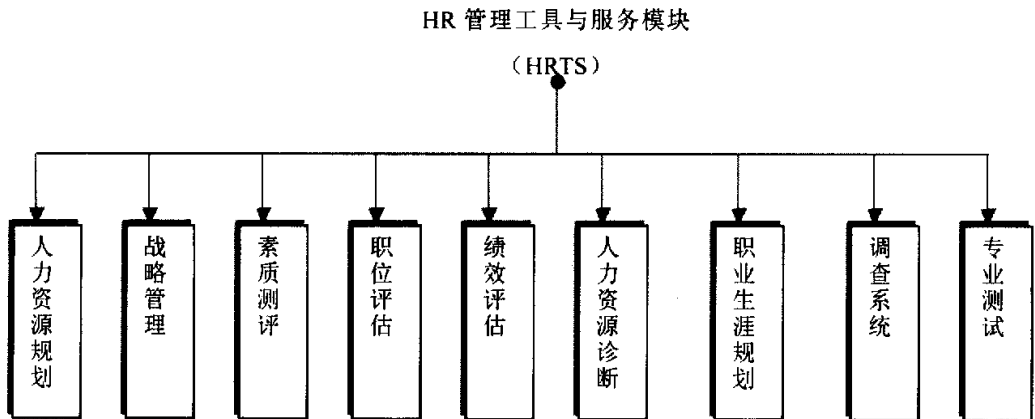
(4) 员工还可查看近期公司培训计划的内容，同时可直接进行网上申请，并随时查询培训审批的结果和已参加培训的考核结果。此外，参加培训的员工还可通过网上的培训调查系统，对培训计划、培训教员进行评估。这样，HR 的培训经理就能方便、准确地了解培训效果以及教员情况，以便进行培训计划和教员的调整；

(5) 员工还可在网上完成对自己、同事和上级主管的考核评估，并及时获得自己的考核结果；

(6) 员工可查询当前的空缺职位，并可直接进行网上申请或推荐；

### 3.2.2.3 e-HR 之人力资源专业管理工具与服务子系统 (HRTS)

这个模块主要由许多 HR 工具小程序模块构成，每一个小程序模块完成一个专业人力资源工作人员对企业或特定员工信息进行专业分析和管理的功能，此模块是人力资源管理理论在系统中的集中体现和系统集成，也是 eHR 系统区别与其他系统的灵魂所在。其功能模块如下图 3-5：



3-5 eHR 之人力资源专业管理工具与服务子系统结构

#### 1、人力资源规划

主要实现以下功能：

- (1) 实时查看本行业内主要竞争对手有关人力资源的基础情况；
- (2) 根据组织结构变化等因素预测人力资源年度需求状况，根据现有员工状况预测人力资源内部供给情况，为供需平衡提供依据；
- (3) 通过对企业现有人力资源状况的分析，根据人力资源需求状况，优化现有人力资源配置；
- (4) 根据企业历史数据及战略发展生成三年、五年人力资源中长期规划及人

员结构等素质规划；

(5) 根据人力资源供需情况，制定年度人力资源规划，并在此基础上形成组织整体的、各个部门的季度、月度实施计划；

(6) 结合人力资源战略要求和人力资源现有情况，生成更新率、增长率、离职率等规划指标，确定新进、淘汰、调动、培训开发的基本目标；

(7) 能对空缺职位进行管理，作为招聘计划的重要参考依据；

(8) 输出历史人力资源配置状况及人力资源成本的发展趋势报告；

(9) 根据组织规模及薪酬水平战略，进行人工成本预算的宏观管理。

## 2. 战略管理

主要实现以下功能：

(1) 持以全领域、地区、行业、竞争对手、本公司等多种角度对人力资源战略影响因素进行统计分析；

(2) 人力资源战略影响因素涵盖年龄分布及其均值、学历分布及其均值、薪酬分布及其均值、满意度均值、以及多种发展指标及其均值；

(3) 系统提供对各类资源进行柱状图、饼状图、曲线图等多种图表分析功能，对定制各种制度提出帮助；

(4) 实现对战略类别进行对照统计分析，形成图表，为人力资源规划构建参考指标。

## 3. 素质测评系统

主要实现以下功能：

(1) 系统包含人员数学能力测试、人员语言能力测试、类比推理能力测试以及人员综合能力测试等多种基本能力测试体系；

(2) 提供员工心理素质测试；

(3) 为员工录用、培训、职位变动等提供数据接口；

## 4. 职位评估

主要实现以下功能：

(1) 实现不同的 HR 咨询师对不同的职位进行评估；

(2) 实现对同一职位的不同评估结果的汇总；

(3) 为招聘、绩效考核、薪资福利提供接口。

## 5. 绩效考核

绩效评估主要将过程管理与 360 度考核相结合，针对企业不同人员的考核制订不同的考核标准，最终达到个人绩效、团队绩效以及公司发展相一致的需要。

主要实现以下功能：

(1) 定性和定量的 360 度评估方式；

(2) 支持年度、季度、月份和不定期等按时段方式的评测，并根据需要按期发布，以便员工通过自助中心进行评估表的填写；

(3) 用户可以根据不同员工类型以及部门设定评估的类型、项目；

(4) 用户可任意设计符合自己需求的评估表，包括：指标组成、指标权重以及评估方式；

(5) 评估人员可灵活、方便地设定；

(6) 系统对评估结果进行细致的跟踪，以便用户从不同角度进行分析和统计；

(7) 在记录员工绩效评估的历史情况的同时，可与薪资和培训计划相关联；

(8) 能够实现企业上级对下级的考核模式。

(9) 能够实现企业上级对下级考核+同级考核的考核模式。

(10) 能够实现企业上级对下级考核+员工自评的考核模式。

(11) 能够实现企业上级对下级考核+员工自评+同级互评的考核模式。

## 6、资源诊断

主要实现以下功能：

(1) 制人力资源诊断计划；

(2) 网上填写、回收、汇总人力资源诊断日志；

(3) 提供编制诊断报告功能，并通过网上进行申报；

(4) 提供统计分析，并生成多种统计图表；

(5) 支持制定人力资源改进计划。

## 7、职业生涯规划

职业生涯规划是指一个人在其工作中所经历的一些列职位、工作或职业，以及与之相关的价值观、工作态度、工作动机的变化过程的统称。

主要实现以下功能：

(1) 生成公司内部职业图表体系；

(2) 进行岗位基本能力、个人素质的分类统计；

(3) 形成员工职业发展曲线，设定职业发展计划；

(4) 支持对职业发展计划进行改进，并形成职业发展改进计划。

## 8、调查系统

主要实现以下功能：

(1) 制员工满意度调查计划；

(2) 网上填写、回收、汇总员工调查日志；

(3) 对调查数据进行分类并统计，支持生成分类调查报表；

(4) 提供编制调查报告功能，并通过网上进行申报；

(5) 提供统计分析，实现员工满意度的历史数据的对比分析；

(6) 支持编制员工满意度改进规划。

#### 9. 专业测试

主要实现以下功能：

(1) 供职业能力测试、职业适应性测试、职业发展倾向测试，为员工职业发展提供指导；

(2) 根据客户需要，可提供多种专业技能测试；

(3) 具有强大的统计分析功能，支持员工对比分析和历史分析，并生成多种形式统计图表。

#### 3.2.2.4 e-HR 之公共信息服务与管理子系统 (PMSM)

此模块主要完成对公共信息的管理，包括企业发布的公告政策法规、部门文件管理、邮件服务、报表管理等。其功能模块如下图 3-6：

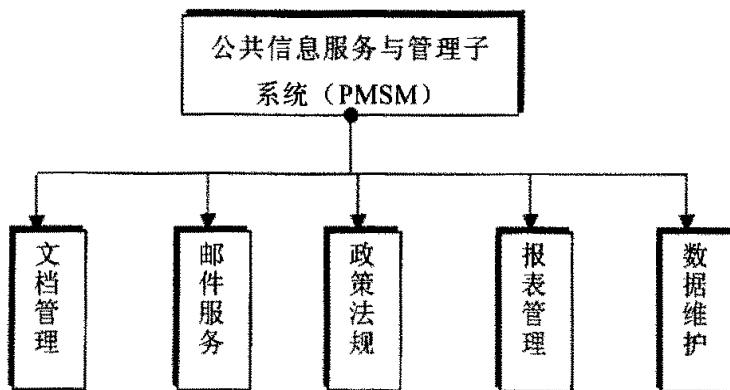


图 3-6 eHR 之公共信息服务与管理子系统结构图

##### 1. 档管理

主要处理部门与部门、部门与员工、员工与员工之间的书面文档，包括文件的发送，文件的接受，文件的删除，文件的编辑，发送方可以根据文件的查阅状况来跟踪文件的查阅情况，接收方则可以清楚实时地收文件。

##### 2. 邮件服务

实现内部与外部之间的电子邮件的发送与接收，此功能模块可以根据用户的需要设置一个或几个邮件服务器地址，接收来自互联网或内部网的邮件。

##### 3. 策法规

用户可根据公司管理的需要，将《公司章程》《员工守则》等规章制度以及相关的人事劳动法规，如：劳动法，发布到网上，使员工进一步了解公司制度及文化；人事部门也可将各种人事信息发布到网上，以便及时传达公司管理精

神和人事信息；完成对企业内容规章制度、新闻及通知的管理。

#### 4. 报表管理

报表分为设计报表和使用报表两部分。我们将各种报表进行归纳总结，将其制作方式大致分为：简易表、分组报表、交叉统计表和空白表四种。

(1) 灵活自定义各种查询和报表，所有报表的数据范围和查询条件可自由控制；

(2) 可灵活定义报表显示格式，生成多种分析图表；提供报表自动校验功能；

(3) 灵活输出报表并设置打印功能，包含树状查询、条件查询、统计报表等功能；

(4) 树状查询可满足所有员工在工作中对于组织结构及相关作人员信息的查询需求，条件查询可根据管理者的权限范围按其需要对员工所有信息进行查询；

(5) 统计报表对企业人事信息进行汇总；

(6) 本系统提供给用户自定义查询的功能可使用户根据自己的需要进行查询方案设定，并可保存查询方案。

#### 5. 数据维护

本系统设计了强大的数据转换功能，以方便对数据的更深一层次的处理，可以将 sql server 2000 的指定的表记录转换成（导出为）excel、access 数据或.txt 文本，也可以将 excel、dbf、access 数据或.txt 文本数据导入到指定的 sql server 2000 表中。

### 3. 2eHR 软件系统数据库的设计

在 eHR 系统中需要对庞大的数据进行相关的处理，如何高效有序地对系统中涉及到的数据库中的表进行操作是衡量系统好坏的标准之一。科学合理地设计数据库并规范设计方案是系统是否能正常运转的关键中的关键。

#### 3. 3. 1eHR 软件系统数据库结构和关键数据表

##### 1、数据库结构

在 eHR 系统中，所有的数据处理都是围绕者员工基本信息和企业相关信息以及公共信息和 HR 专业信息展开的，企业员工通过员工自助系统了解自己的利益密切相关的信息，也可以了解到企业的公共信息比如政策法规，最新通告文件等等；企业的管理者则不仅可以得到信息，也可以发布信息，并且可以以得到的员工和企业的基本信息和数据为基础，结合专业的 HR 工具和知识，对所有的数据进行分析、统计和汇总，进而形成企业高层决策的依据和调整企业管理方式和提高管理水平的参考。基于这一基本的认识，我们可以建立 eHR 数据库结构图，如下图所示：

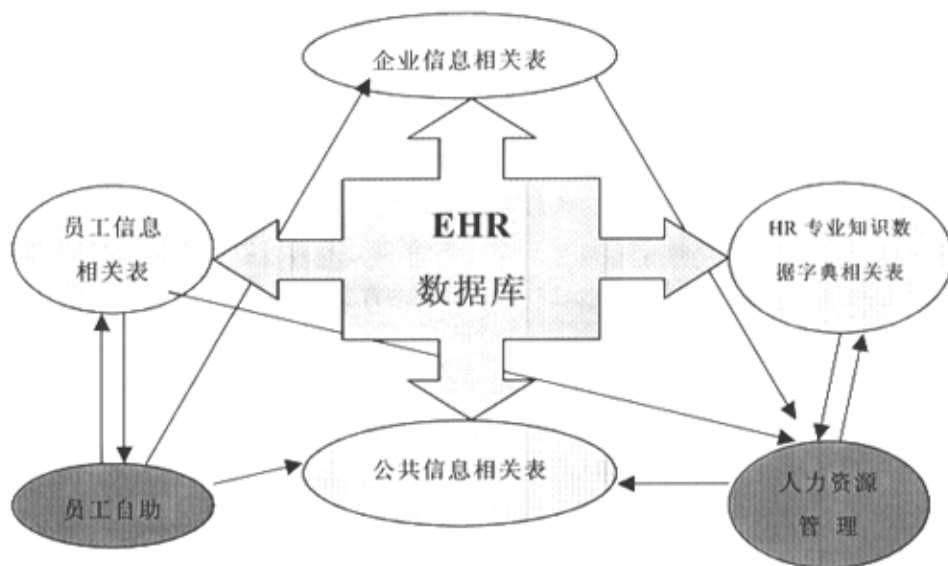


图 3-7 eHR 数据库结构

## 2、关键数据表：

(1) 员工信息相关表：员工信息相关表由记录员工的登录信息、员工详细资料、考勤记录、薪资信息、福利保险状况、教育培训记录等等涉及员工工作和生活方方面面的资料和信息的一系列不同的表构成。

(2) 企业信息相关表：企业信息相关表主要记录企业的基本信息、企业文档、职能职务体系、组织结构、权限设置、角色分配等。

(3) HR 专业知识相关表：HR 专业知识相关表是人力资源管理系统实施专业的评估、专业测试和考核的理论源泉，这些理论依据和参与计算时涉及的数据以数据表的字段内容的形式存在于数据库中，在系统运行时才被调出参与具体的操作，如职位评估、素质模型、绩效评估、企业诊断等。

(4) 公共信息相关表：此类表主要存放企业公共信息，如政策法规、企业公告、招聘信息等。

## 3.3. 2eHR 软件系统数据库主要存储过程

存储过程（Stored Procedure）是一组为了完成特定功能的 SQL 语句集，经编译后存储在数据库中。用户通过指定存储过程的名字并给出参数（如果该存储过程带有参数）来执行它<sup>[6]</sup>。存储过程是数据库中的一个重要对象，任何一个设计良好的数据库应用程序都应该用到存储过程。总的来说，存储过程具有以下一些优点<sup>[7]</sup>：



- 1、存储过程允许标准组件式编程
- 2、存储过程能够实现较快的执行速度
- 3、存储过程能够减少网络流量
- 4、存储过程可被作为一种安全机制来充分利用

在 eHR 系统使用的过程中，不断地要进行查询数据库，对数据库进行插入、修改、删除等操作。为了提高程序的执行效率，同时也为了尽量减少在程序中编写复杂的 SQL 语句，在数据库的设计中添加一些存储过程，可以大大提高代码的执行效率，也增加了代码的可读性。eHR 系统主要的存储过程如下：

#### 1)、添加用户 AddUser 存储过程

该存储过程在用户注册功能中使用，用于向数据库中添加一个新用户，用户必须提供用户名和密码等信息，如果添加成功则返回此用户的 ID。

#### 2)、用户合法性检查 CheckUserName 存储过程

此存储过程在注册时被调用，用来判断新注册的用户名是否已经被使用过，如果被使用则必须选择另外的用户名，因为在用户表（USER）中，用登录时的登录名(EngName)唯一标识每一个用户。

3)、用户信息显示 DisplayUserInfo 存储过程调用此过程可以查看到登陆用户的相关信息。

#### 4)、用户信息修改 UserInforModify 存储过程

用户通过调用此存储过程来修改数据库中的个人登陆信息，除用户登陆名以外，其它信息都可以修改。

#### 5)、用户登陆 UserLogin 存储过程

该存储过程用于用户登陆时进行身份人证，只有输入的用户名和密码同时和注册信息相匹配的用户才可以登陆成功，否则为登陆失败，不能进入 EHR 系统。

### 3.3.3 数据库公共访问类

为了提高代码的重用性，考虑实现一些公共类库，提供整个系统使用<sup>[8]</sup>。由于与数据库操作相关的代码很多，所以新建一个 PubSqlClass.cs 的类文件，把对数据库的操作封装在这个类里面，自定义一个公用的 PubSqlClass 类，可以实现非常方便地操作数据库，简化了其它页面对数据库操作的逻辑，增强了其他页面的代码的可读性。该类包含的常用函数如下：

#### 1)、open函数

OPEN方法用于打开到数据库的连接。

#### 2)、GetDataReader函数

GetDataReader 方法使用 DataReader 从数据库中读取数据，输入参数为连接数据库的字符串，返回一个 DataReader 数据集。

3)、ExcuteSQL函数

ExcuteSQL方法，执行SQL语句，并返回执行结果，返回影响数据库的行数。

4)、GetDataSet函数

GetDataSet 方法使用 DataSet 从数据库的指定表中读取数据，返回一个存放结果的数据集。

5)、Close函数

关闭到数据库的连接。

6)、AddUser函数

AddUser 方法调用 AddUser 存储过程，用于往数据库里面插入一条新用户的记录，并返回唯一的 Id 值。

7)、UserLogin函数

UserLogin 函数用于判断登陆用户是否合法，根据输入的登陆用户名和密码调用 UserLogin 存储过程在数据库中查找是否有匹配的记录，如果存在则为合法且返回所有信息，如果不合法则返回空。

8)、CheckUserName函数

CheckUserName 函数调用存储过程来判断新注册的用户所提交的登陆用户名是否已经被别人使用，如果没有则返回此用户名，否则返回空。

9)、DisplayUserInfo函数

DisplayUserInfo 函数调用存储过程根据登陆用户名取得用户的所有信息。

10)、UserInforModify函数

UserInforModify 函数调用存储过程对用户的信息进行修改。

### 3. 4eHR 软件系统的 web 页面设计

#### 3. 4. 1 设计原则

1)、Web 页面的设计必须遵循简洁、美观大方、直观明了。

Web 页面是直接面向用户的界面，所有的操作都在此页面下进行，所以界面必须友好、美观大方，不应太复杂。

2)、条理清晰，使用方便的原则。

3)、必须体现数据流和事务流在系统的使用中的合理性和人性化

#### 3. 4. 2 用户自定义控件

基于.NET 技术的 eHR 系统是面向互联网的应用系统，系统的界面由 ASP.NET 实现，网页上数据的输入和显示主要使用 ASP.NET 的标准 WEB 控件和用户自定义的控件。为了提高整个网站的可维护性和代码重用性，对多数页面都需要重复显示的。同一种元素，一般使用用户自定义控件来实现，然后在每个页面内引入

这些用户自定义控件。这些自定义控件有：标题头控件、显示版本信息控件、常用功能控件等。

### 3.5 eHR 系统环境

#### 3.5.1 eHR 系统硬件环境

系统运行需要的硬件环境为：

服务器端：PIV 2.1G CPU，40G 以上硬盘，512 兆内存，15 彩显，千兆网卡，智能交换机，防火墙；

客户端：PIII CPU，64 兆以上内存，2G 以上硬盘，15 彩显，10/100 兆网卡；

#### 3.5.2 eHR 系统软件环境

系统的运行需要的软件环境为：

服务器端：Windows 2000 Advance Server，Exchange Server 2000，.Net 框架，IIS 5.0 以上，SQL Server 2000

客户端：Windows 98 及以上版本操作系统，IE5.5 版本以上浏览器。

### 3.6 小结：

本章主要介绍了基于.NET 技术的 eHR 系统的总体分析与设计，介绍了 eHR 系统开发的拓扑结构和系统的三层体系结构，介绍了 eHR 软件系统的设计特点、功能模块组成及其详细功能，以及开发和实施 eHR 系统所要求的硬件环境和软件环境。

## 第四章：基于.NET 技术的 eHR 系统的详细设计与实现

### 4.1eHR 之系统设置与管理子系统的详细设计与实现

eHR 系统设置与管理子系统主要包括用户管理、权限管理、职能职务管理，在这个子系统里主要完成的功能为：设置企业的职务体系以及对应的每一个职务可以行使哪一些职能；回答和解决哪些用户属于哪些职位及对应这些职位扮演怎样的角色，这些角色拥有哪些权限的问题。其关系图如下：

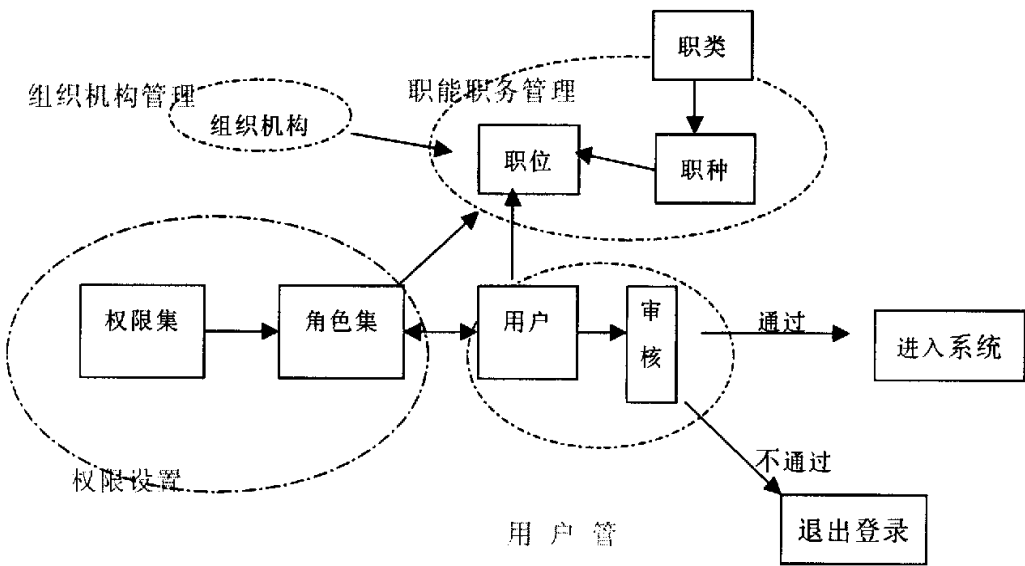


图 4-1 eHR 之系统设置与管理子系统功能关系图

基于以上的分析，在 eHR 人力资源管理系统中必须有一套成功的解决方案来解决企业现有职务体系和组织结构下的用户授权问题，即必须以用户拥有权限的多少来决定对资源的访问程度，为此我们设计了基于 RBAC 的角色授权机制作为解决方案，并成功地得到运行。

#### 4.1.1 基于 RBAC 的角色授权机制在 eHR 系统中的应用和实现

##### 4.1.1.1 RBAC 角色访问控制策略

对于在企业管理系统环境中的访问控制方法，一般有三种<sup>[9]</sup>：

1) 自主型访问控制方法。目前我国的大多数的信息系统中的访问控制模块中基本是借助于自主型访问控制方法中的访问控制列表(ACLs)。

- 2) 强制型访问控制方法。用于多层次安全级别的军事应用。
- 3) 基于角色的访问控制方法 (RBAC)。

RBAC (Role Based Access Control) 基于角色的系统访问控制模型是目前国际上流行的先进的安全管理控制方法，也是目前公认的解决大型企业的统一资源访问控制的有效方法。其显著的两大特征是：

- 1) 降低了授权管理的复杂度和企业管理成本；
- 2) 支持灵活的企业安全策略，企业管理的可扩展性得到了加强。

eHR 安全管理系统也根据自身的需要有选择性的吸收其部分思想。其特点是通过分配和取消角色来完成用户权限的授予和取消，并且提供了角色分配规则和操作检查规则。安全管理人员根据需要定义各种角色，并设置合适的访问权限，而用户根据其责任和资历再被指派为不同的角色。这样，整个访问控制过程就分成两个部分，即访问权限与角色相关联，角色再与用户关联，从而实现了用户与访问权限的逻辑分离，如下图所示，角色可以看成是一个表达访问控制策略的语义结构，它可以表示承担特定工作的资格。



图 4-2 基于角色控制的基本思想

RBAC 模型分为四个模型<sup>[10]</sup>：

#### 1、Core RBAC

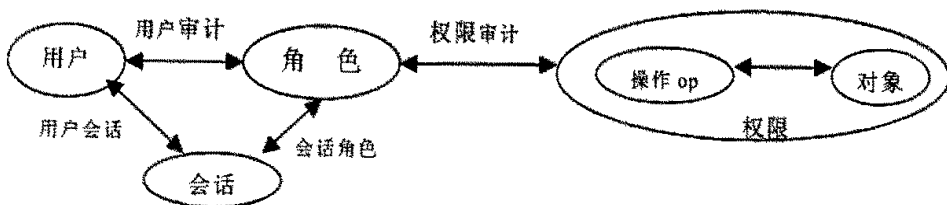


图 4-3 Core RBAC 模型

图 4-3 中的操作代表 operations, 对象代表 objects, 即规定的需要保护的资源, 又称目标。权限的定义包括了操作（类型）和对象（资源）的概念。之所以将操作和对象绑定在一起，这是因为很多的操作对于某对象才有意义。比如，发送操作对文件对象才有意义，对用户对象则没有意义。

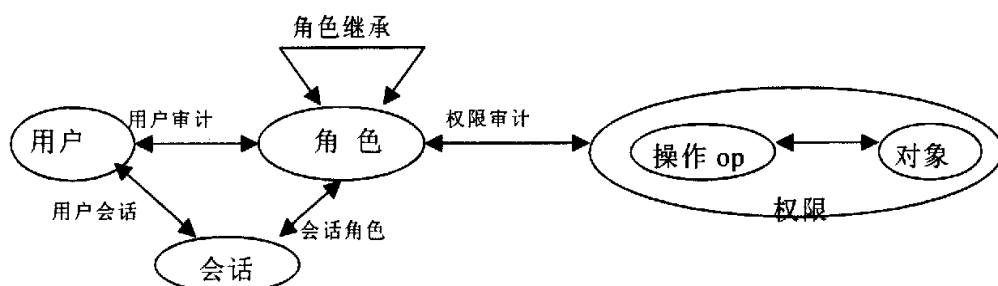


图 4-4 Hierarchal RBAC 模型

## 2、Hierarchal RBAC

这个模型是在角色间衍生出层次性的关系。如上图 4-4，这样就可以把一些较为基本的权限分配给层次较为低的角色，再利用继承的方式继承权限，这样就免去了每个角色都要重复一些基本的权限分配行为，从而减轻了维护上的负担。

同时，Hierarchical RBAC 又可以分为：

1)、General Hierarchical RBAC（一般性层次模型）：指上层的角色可以继承下层的所有权限，而且没有任何限制。

2)、Limited Hierarchical RBAC（限制性层次模型）：指上层的角色继承下层的权限是有限制范围的。

后面两种模型是权责分离（Separation of Duty）的。权责分离主要是防止使用者拥有的权限过多，而造成资源滥用或者利益冲突（conflict of interest）的情况发生。

### 3)、Static Separation of Duty Relations（静态权责分离）

又称强互斥。指具有冲突的角色不能同时分配给同一个使用者，以避免利益冲突（conflict of interest）

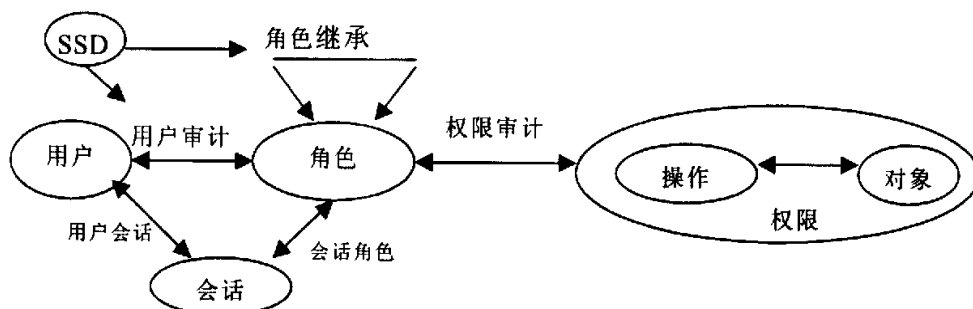


图 4-5 Static Separation of Duty Relations 模型

图 4-5 中的 SSD：Static Separation of Duty Relations 的缩写

#### 4)、Dynamic Separation of Duty Relations (动态权责分离)

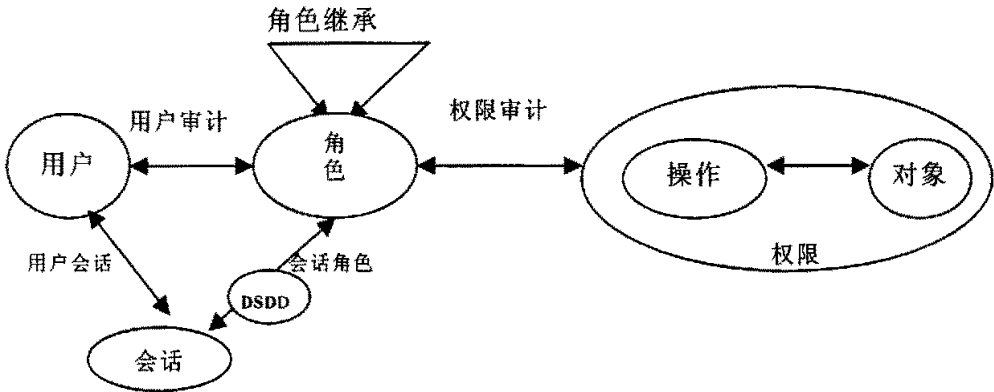


图 4-6 Dynamic Separation of Duty Relations 模型

图 4-6 中的 DSD: Dynamic Separation of Duty 的缩写。

动态权责分离又称为弱互斥。指使用者可以同时拥有相互冲突的角色，但是使用者不能在同一时间担任。

在 RBAC 中的各个模型中，都是由下面一些成分所组成：

- a)、基本的一些元素。
- b)、一序列的 RBAC 的关系，也包含了包含有效的笛卡儿积的子集。
- c)、一些映射功能。为一个给定的实例确定的一种元素中产生成员的实例。

##### 4.1.1.2 基于 RBAC 的角色授权机制在 eHR 系统中的实现

###### 1、相关的几个概念

**主体：**即可以象应用系统发出应用请求的任何实体，包括各种用户、其它与本系统有接口的应用程序、非法入侵者。系统必须具有识别主体的能力，接口实际上也是由用户登记的，故主要问题是校验用户身份的合法性，系统应建立用户鉴别机构以验证用户身份。

**用户：**用户就是一个可以独立访问计算机系统中的数据或者用数据表示的其它资源的主体，我们用 users 表示一个用户集合。用户在一般情况下是指人。

**权限：**权限是对计算机系统中的数据或者用数据表示的其它资源进行访问的许可。我们用 permission 表示一个权限集合。可分为对象访问控制和数据访问控制两种。

**权限对象的访问控制：**用一个二元组来表示：(访问类型,控制对象)。其中的访问类型是指对于相应的受控对象的访问控制，如：读取、修改、删除等等。控制对象表示系统中一切需要进行访问控制的资源。我们将引入一套完整的资源表示方法来对系统中出现的各类资源进行定义和引用。

权限的完整定义最终可以组合成如下形式：（访问类型，控制对象,逻辑值）。逻辑值用来来确定是否拥有该权限，当为真值时表示拥有，为假值时表示没有改权限。

角色：角色是指一个组织或任务中的工作或位置，它代表了一种资格、权利和责任。我们用 roles 表示一个角色集合。

用户委派：用户委派是 users 与 roles 之间的一个二元关系，我们用  $(u,r)$  来表示用户  $u$  被委派了一个角色  $r$ 。

权限配置：权限配置是 roles 与 permission 之间的一个二元关系，我们用  $(r,p)$  来表示角色  $r$  拥有一个权限  $p$ 。

粗粒度：表示类别级，即仅考虑对象的类别，不考虑对象的某个特定实例。比如，用户管理中，创建、删除，对所有的用户都一视同仁，并不区分操作的具体对象实例。

细粒度：表示实例级，即需要考虑具体对象的实例(the instance of object)，当然，细粒度是在考虑粗粒度的对象类别之后才再考虑特定实例。比如，合同管理中，列表、删除，需要区分该合同实例是否为当前用户所创建。

## 2、RBAC 基本思想的补充

在角色访问控制的基本原理上增加组（Group）的概念，其关系图 4-7：

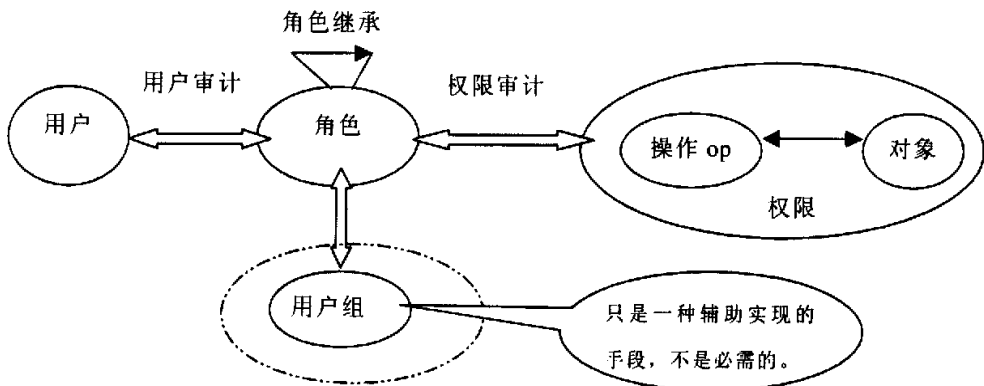


图 4-7 增加“组”的概念后的 RBAC 模型

用户(User)与用户组(Group)是多对多的关系。即一个用户可以属于多个组之中，一个组可以包括多个用户。子组与父组是多对一的关系。权限某种意义上类似于“操作 +对象”概念，但这里的操作仅包括操作类型不表示操作实体。组可以直接映射组织结构，角色(Role)可以直接映射组织结构中的业务角色，比较直观，而且也足够灵活。角色对系统的贡献实质上就是提供了一个比较粗粒度的分



配单位。组与权限是多对多的关系。组应和角色严格分离，组只用于对用户分类，不包含任何角色的意义；角色只授予用户，而不是组。如果用户需要还没有的多种操作的组合，必须新增角色。权限必须能够访问对象，同时带用户参数，这样权限控制就完备了。组和角色，只是一种辅助实现的手段，不是必需的。如果系统的角色很多，逐个授权违背了“简单，方便”的目的，那就引入组，将权限相同的角色组成一个组进行集中授权。角色也一样，是某一类权限的集合，是为了简化针对多个权限对象的操作。

### 3、RABC 在 eHR 系统中实现的思路：

1) 根据功能模块的不同整体规划供用户访问的访问对象，包括各种功能模块、数据、界面元素（包括菜单、按钮等各种界面上能控制的控件），按 ID 分类标识系统中可以使用的资源（页面、菜单），如 file\_类表示对文档的处理，email\_类表示对邮件的处理，hr\_类表示对人力资源专业分析工具模块的使用，system\_表示对系统设置的标识，pub\_类表示公共信息等等；如 file\_sendnew 表示发送新文件，file\_del 表示删除文件，hr\_jobevaluate 表示职位评估等等，每一个 ID 标识对应一个逻辑值(.True. or .False.)，当值为真(.True.)时表示可以访问，即拥有对此子项（功能模块）的访问的权限；当值为假(.False.)时表示不可以访问；即对此子项（功能模块）没有访问的权限。

2) 根据系统的规划建立角色数据表、用户数据表、权限数据表、职类表、职种表以下为规划中的相关表及说明：

表 4-1-1 职类表:JobType

字段名称	中文标题	类别	备注
JobType_Id	职类码	Char(20)	将职务分成：职类、职种、职位的层次，职类职务中的最高层。
JobType_Name	职类名称	Char(30)	

表 4-1-2 职种:JobDepart

字段名称	中文标题	类别	备注
JobDepart_Id	职种码	Char(20)	所属职类码可以表明该职种的职类。
JobDepart_Name	职种名称	Char(30)	
JobDepart_JTypeid	所属职类码	Char(30)	

表 4-1-3 登录用户表：User

字段名称	中文标题	类别	备注
User_Name	用户姓名	Char(20)	
User_Id	员工工号	Char(20)	一个用户只有一个员工号
User_LoginName	用户登录名	Char(20)	只能是英文字母
User_Password	用户密码	Char(20)	
User_EmailMax	邮箱容量	int(4)	
User_Depart	所在部门	Char(20)	
User_job	任职职位	Char(20)	
User_Roleset	角色码集	Char(200)	一个用户可能同时担任几个角色，各个角色用角色码表示，角色码间用“，”分开。如 100，200，。。。

表 4-1-4 角色表：Role

字段名称	中文标题	类别	备注
Role_Name	角色名称	Char(30)	
Role_Id	角色码	Char(100)	
Role_Mutex	互斥角色码集	Char(100)	存放与该角色互相排斥的角色的角色码
Role_UnMutex	继承角色码集	Char(100)	存放与该角色有继承关系的角色角色码
Role_Permission	权限码集	Char(200)	存放与该角色有关的权限代码的集合，一个角色可能有很多个权限。各个权限代码用“，”分开。如 100，200，。。。

表 4-1-5 权限表 Permission

字段名称	中文标题	类别	备注
Permission_Id	权限代码	Char(50)	权限包括操作和资源两部分
Permission_Name	权限名称	Char(100)	
Permission_About	操作描述	Char(100)	
Permission_Code	权限码	Char(1000)	此码由表示操作和资源的标识码组合而成

表 4-1-6 用户组表表 (Group)

字段名称	中文标题	类别	备注
Group_Id	用户组编码	Char(50)	
Group_Name	用户组名称	Char(20)	
Group_UserId	组用户编码集	Char(1000)	用来存放该组包含的用户 ID 集
Group_Permission	组权限码集	Char(1000)	用来存放该组包含的权限 ID 集
Group_About	用户组描述	Char(100)	

3) 在用户数据表中创建三类用户;

因为权限系统的核心由以下三部分构成:

a)创造权限,

b)分配权限,

c). 使用权限

所以, 在用户表中主要必须对应创建三种类型的用户, 对照如下:

a)Creator 类: 创造权限;

b). Administrator 类: 分配权限;

c). User 类: 使用权限。

Creator 类用户在设计 and 实现系统时会划分一个子系统或称为模块, 应该有哪些权限。这里完成的是操作类型与资源对象的声明, 并没有真正将操作类型与具体的资源对象联系在一起, 形成权限。

Administrator 类用户指定操作类型与资源对象的关联。在这一步, 操作类型真正与资源对象联系到了一起, 产生了权限实例。Administrator 利用权限这个基本元素, 来创造他理想中的权限模型。如, 创建角色, 创建用户组, 给用户组分配用户, 将用户组与角色关联等等...这些操作都是由 Administrator 来完成的。

User 类用户使用 Administrator 类用户分配给的权限去使用各个子系统。在 Administrator 类用户的心目中有一个比较适合他管理和维护的权限模型。Administrator 类用户可以按照他的意愿来建立他所希望的权限框架, 可以自行增加, 删除, 管理资源对象和权限之间的关系。可以自行设定用户和角色角色的对应关系。(如果将 Creator 类用户看作是发明者, Administrator 类用户就是使用者。)权限是这个系统中最关键的部分, 它是一个纽带, 一个系在程序设计员 Administrator 类用户, User 类用户之间的纽带。

4) 由 Creator 类用户创建权限;

5) 由 Administrator 类用户指定用户与权限 (操作类型和资源对象) 的关联, 创建角色, 创建用户组, 给用户组分配用户, 将用户组与角色关联;

6) User 类用户使用 Administrator 类用户分配给的权限去使用各个子系统。

4、RABC 在 eHR 中的技术实现

以对“我的文档”这个功能模块的实现来说明 RABC 在 eHR 系统中的实现过程，如图 4-8：

一、定义权限、设置角色和角色功能、建立用户并分配角色

1) 在“我的文档”功能模块中有四个功能模块(图 4-8)，分别是发送新文件、删除旧文件、编辑已收到的文件、查找文件，给这四个功能模块各自分配一个 ID，这个 ID 叫做功能代码：MyFile\_Send, MyFile\_Edit, MyFile\_\_Del, MyFile\_Find。

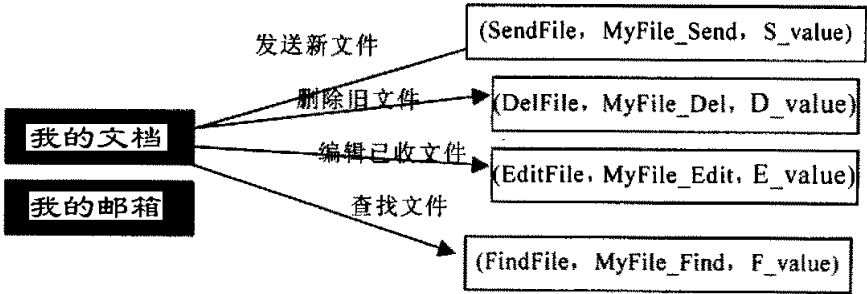


图 4-8 用三元组表示对象的访问权限

(SendFile,MyFile\_Send, S\_value) 三元组表示对我的文档访问权限表示成（访问类型，控制对象(ID)，逻辑值）的形式，当前访问类型为 SendFile(发送文件)，控制对象为 MyFile\_Send ,即对应“我的文档”功能模块的发送文件部分，当其 S\_value 的值为.True.时，表示拥有发送文件的功能，当其 S\_value 的值为.False.时，表示不能发送文件。

2) 建立一个角色(Role)，把上面的功能代码加到这个角色拥有的权限中，并保存到数据库中。角色包括系统管理员（如超级用户），测试人员等，如图 4-9。

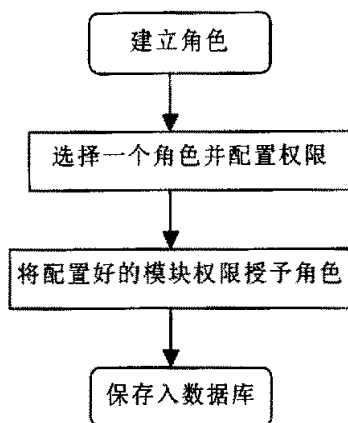


图 4-9 给角色分配权限

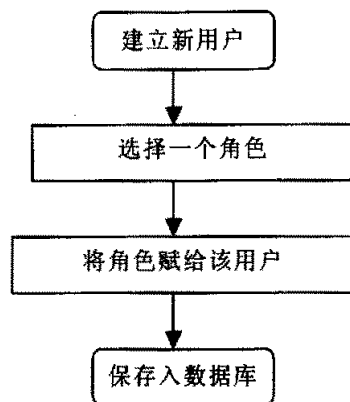


图 4-10 给用户分配角色

3) 建立一个员工的账号，并把一种或几种角色赋给这个员工。比如说这个员工既可以是系统管理人员，也可以是测试人员等。这样他登录到系统中将会只看到他拥有权限的那些模块，如图 4-10。

## 二、通过全局变量或 Session 来传递登录用户的身份信息

登录时，先到数据库中查找是否存在这个员工，如果存在，再根据唯一的员工工号查找员工的权限信息，把员工所有的权限信息（在这里表现为个控制对象的逻辑值的真假）都保存在一个全局的数组中，比如就把上面的 `S_value=true` 或 `S_value=false` 等放到这个全局的数组里。最后把这个数组里的值调出来放到 Session 中，这样在整个程序的运行过程中，系统随时都可以取得这个用户的身份信息。

## 三、根据用户的权限做出不同的显示。

可以对比当前员工的权限和给这个访问对象（各种功能模块、数据、界面元素（如菜单、按钮等各种界面上能控制的控件））分配的“功能 ID”判断当前用户是否有打开这个菜单的权限。例如：如果保存员工权限的数组中没有这三个 ID 的任何一个，那这个菜单就不会显示，如果员工的数组中有任何一个满足条件的 ID，那这个菜单都会显示。其访问控制如图 4-11：

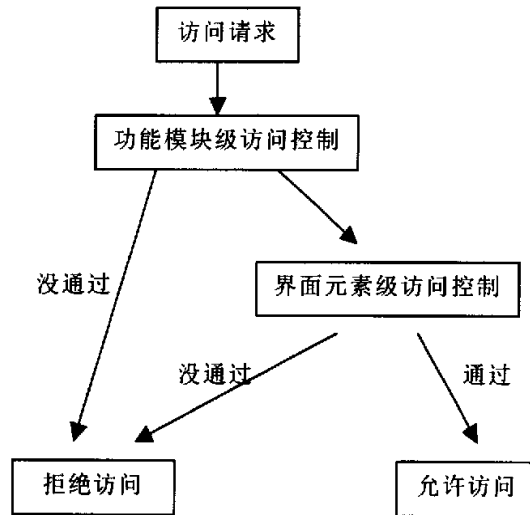


图 4-11 对界面元素的访问控制流程

一个用户可以拥有多种角色，但同一时刻用户只能用一种角色进入系统。角色的划分方法可以根据实际情况划分，按部门或机构进行划分的，至于角色拥有多少权限，这就看系统管理员赋给他多少的权限了。用户---角色---权限的关键是角色。用户登录时是以用户和角色两种属性进行登录的（因为一个用户可以拥有多种角色，但同一时刻只能扮演一种角色），根据角色得到用户的权限，登录后进行初始化。同一时刻某一用户只能用一种角色进行登录，避免了互斥的角色同时出现的混乱。而角色权限的继承主要通过设置权限的真值串和分解权限的真值串来体现。

## 4.2 小结：

eHR 管理系统根据自身的需要有选择性地吸收 RBAC 和扩展了访问控制的思想，通过分配和取消角色来完成用户权限的授予和取消，并且提供了角色分配规则和操作检查规则。本节通过对 RBAC 模式的分析，介绍了 RBAC 的原理，再以具体的实例说明了在 eHR 系统中该机制的实现思路 and 过程。

## 4.3 eHR 之员工信息管理与服务子系统的详细设计与实现

### 1、 基本信息

主要完成基本信息的修改、查找等操作。

表 4-2-1 Employee (职员详细信息表)

字段名称	中文标题	类别	备注
ID	工号	Char(12)	
ChsName	中文名字	Char(14)	
EngName	英文名字	Char(20)	
Sex	性别	Char(1)	
SocialId	身份证号码	Char(20)	
Dateofbirth	出生年月日	Smalldatetime	
nationality	国籍	Char(10)	
Place of birth	出生地	Char(30)	
Co address	通信地址	Char(40)	
hometel	住所电话	Char(14)	
Mobile	移动电话	Char(14)	
emergencycall	紧急联络电话	Char(14)	
Email	E-MAIL	Char(20)	
Zip	邮编	Char(8)	
RprLoc	户口所在地	Char(30)	
RprType	户口性质	Char(8)	
FWorkId	外来人员就业证号	Char(20)	
TlocId	暂住证号	Char(20)	
Marriage	婚姻情况	Char(4)	
Nation	民族	Char(8)	
Policety	政治面貌	Char(10)	
OtherName	曾用名	Char(12)	
Photo	相片	image	
SpName	配偶姓名	Char(12)	
Degree	学位	Char(12)	
Major	所学专业	Char(20)	
School	毕业学校	VChar(50)	
Memo	备注	Vchar(5000)	

## 2、 福利保险

表 4-2-2:EmpBoon (享受福利)

字段名称	中文标题	类别	备注
EmployeeId	工号	Char(12)	
BoonType	福利类别	Char(8)	
BoonId	帐号	Char(20)	
PayLoc	交款地点	Char(30)	
PayMode	交款方式	Char(12)	

## 3、 薪酬管理

表 4-2-3:EmpBenefits(薪资信息)

字段名称	中文标题	类别	备注
EmployeeId	工号	Char(12)	
Housefund	住房基金	Float	
Houseaccfund	住房公基金	Float	
Paygrade	工资等级	Char(2)	
Pensionfund	养老金	Float	
Medicalfund	医疗基金	Float	
Unemployfund	失业基金	Float	
Medicalaccount	医疗帐号	Char(20)	
Trainingaccount	训练帐号	Char(20)	
Pensionaccount	养老金帐号	Char(20)	
Housefundaccount	住房基金帐号	Char(20)	
Houseaccount	住房补贴帐号	Char(20)	
Overseapay	海外津贴	Float	
Paytype	薪资性质	Char(8)	
MedicalSchemePolicy	医疗福利政策	Char(30)	
HolidayType	休假性质	Char(8)	
BenefitPolicy	休假政策	Char(30)	



表 4-2-4: Altersalary (薪资异动记录表)

字段名称	中文标题	类别	备注
RecNum	记录号	Int	
Employeeid	员工代码	Char(12)	
Fromvalue	薪资变动从	Float	
Tovalue	薪资变动到	Float	
Alterdate	薪资变动日期	SamIldatetime	
Type	薪资项目代码	Char(4)	

4、 考勤管理

表 4-2-5: AbsentCard (缺勤记录表)

字段名称	中文标题	类别	备注
RecNum	记录号	Int	
Absentdate	缺勤日期	SmallDate	
Absenttime	缺勤时间	Float	
employeeid	员工代码	Char(12)	
Endtime	结束时间	SmallTime	
Starttime	开始时间	SmallTime	
Vacationid	休假代码	Char(4)	

5、 教育培训

(1) 教育培训流程图如图 4-12

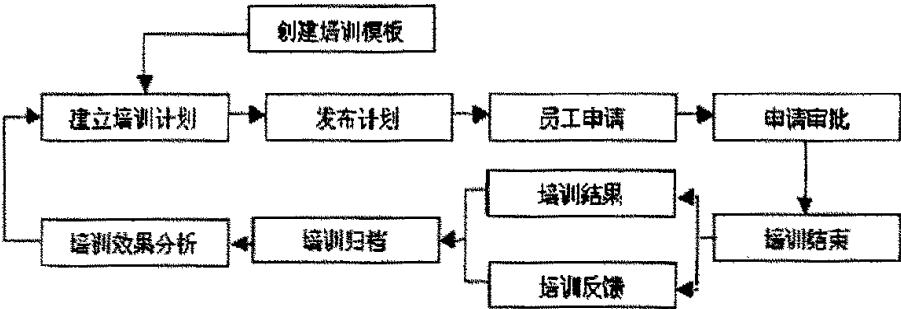


图 4-12 教育培训流程图

(2) 关键表表设计如下：

表 4-2-6 EmpEducation (接受教育情况表)

字段名称	中文标题	类别	备注
EmployeeId	工号	Char(12)	
RecNum	记录号	Int	
learnfrom	起始日期	Smalldatetime	
learnto	终止日期	Smalldatetime	
School	就读学校	Char(30)	
Major	专业(主修)	Char(20)	
Qualification	资格	Char(20)	
degree	学历	Char(12)	
Certificate	证书	Char(50)	

6、 招聘管理

1) 招聘管理流程如图 4-13

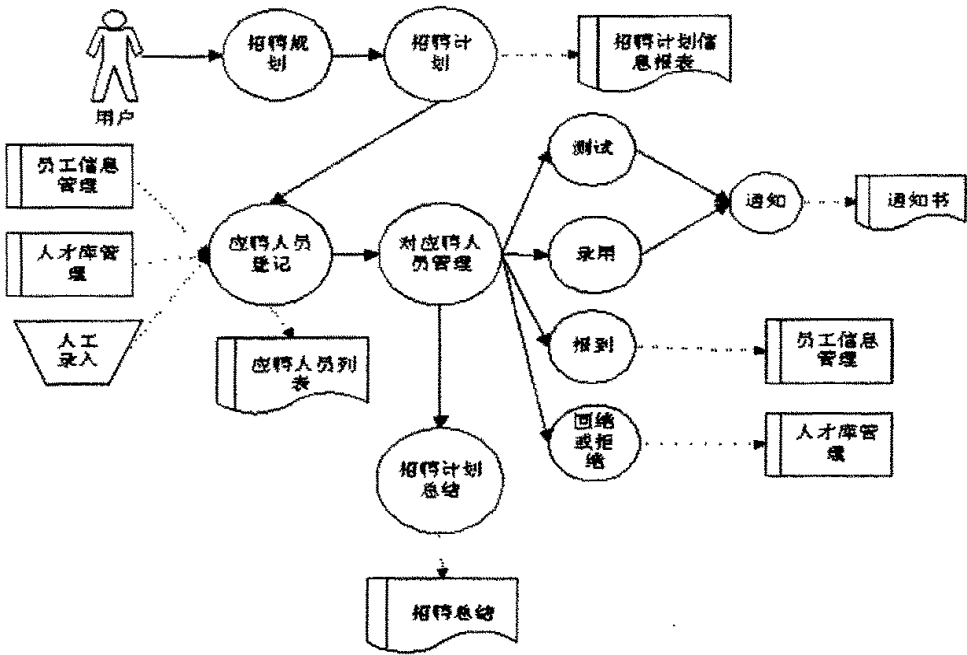


图 4-13 招聘管理流程

(1) 关键表设计如下：

表 4-2-7 Recruit(招聘表)

字段名称	中文标题	类别	备注
RecNum	记录号	Int	
Recruitid	招聘号	Char(30)	
Serialno	序列号	Char(30)	
Times	招聘日期	smallint	
Requisitiondate	申请日期	Smalldatetime	
Department	部门	Char(12)	
Position	职位名称	Char(12)	
Reason	申请原因	Char(60)	
Budgetheadcount	预算人力	Float	
Curheadcount	现有人力	Float	
Newheadcount	新募人力	Float	
Positionquota	本职位名额	Float	
Quota	招聘名额	Float	
Media	招募媒体	Char(20)	
Startsalary	起薪	Float	
requirement	招聘要求	Char(40)	
NumberActual	实际人数	Smallint	
NumberBudget	计划人数	Smallint	
NumberBalance	差额	Smallint	
Numberapplied	应聘人数	Smallint	
Publish	招聘方式	Char(20)	
Fromdate	招聘开始日期	Smalldatetime	
Todate	招聘结束日期	Smalldatetime	
FinBudget	招聘计划费用	Float	
FinActual	招聘实际费用	Float	
FinBalance	招聘费用差额	Float	
Entrydate	输入日期		
Testdepartment	面试部门代码	Char(12)	
Unitcost	单位成本	Float	
Recruirate	招聘比率	Float	
Hirate	录用比率	Float	
Agefrom	年龄 自	Smallint	
Ageto	年龄 至	Smallint	
Experience	工作经历及其他要求	Text	
Sex	性别	Char(1)	
Degree	文化程度		
Language	交谈语言	Char(30)	
Remark	备注	Char(255)	

表 4-2-8 EmployeeHire (录用情况)

字段名称	中文标题	类别	备注
EmployeeId	工号	Char(12)	
StartWorkDate	进入本企业日期	Smalldatetime	
RecMode	招募方式	Char(12)	
TestResult	考试成绩	Char(6)	
Commendor	推荐人	Char(12)	
ComComanpy	推荐人所在单位	Char(30)	
ChangeReson	变更原因	Char(20)	
Department	部门代码	Char(10)	
NormalPay	正式薪金	Float	
TryPay	试用期薪金	Float	
UseMode	用工形式	Char(8)	
HireRequire	公司录用要求	Char(30)	
HireDate	录用日期	Smalldatetime	
OldRelation	与原单位关系	Char(20)	
Position	职位代码	Char(4)	
Rank	职等代码(岗位)	Char(4)	
Shift	班别代码	Char(4)	
ChangeDate	转正日期	Smalldatetime	
WorkId	工作证号	Char(16)	
Bank	开户银行	Char(20)	
Account	个人银行帐号	Char(16)	
WorkBook	劳动手册	Char(20)	
SocialSecId	社会保障号	Char(16)	

## 7、 合同管理

表 4-2-9:EmpContract (合同情况)

字段名称	中文标题	类别	备注
EmployeeId	工号	Char(12)	
Recnum	记录号	Int	
Type	合同类型	Char(8)	
BeginDate	合同期（起始）	Smalldatetime	
TerminateDate	合同期（终止）	Smalldatetime	
MakeDate	合同签订日期	Smalldatetime	
TerminateType	合同终止性质	Char(12)	

表 4-2-9:Contract (合约属性记录表)

字段名称	中文标题	类别	备注
Id	合约属性代码	Char(10)	
Description	合约属性名称	Char(30)	

## 8、 自助系统

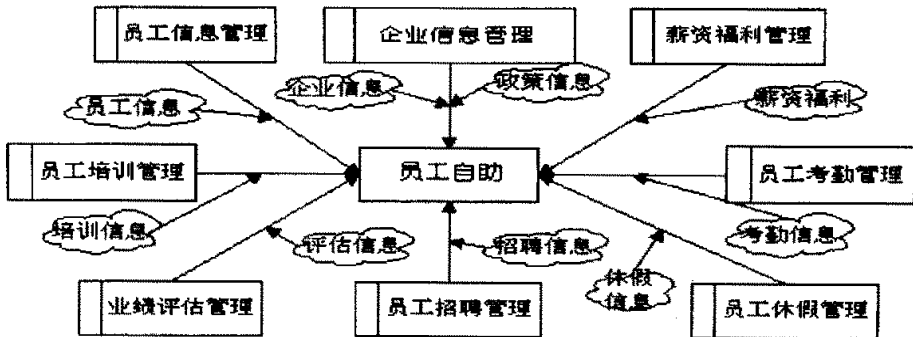


图 4-14 自助系统流程

## 4. 4eHR 之人力资源专业管理工具与服务子系统的详细设计与实现

### 4. 4. 1 eHR 软件系统中职位评估的实现

“职位”是人力资源管理中的一个关键性概念。对职位的定义包括三个方面的内容<sup>[11]</sup>:

职位的内容：责任、任务、行为、功能和义务；

职位的资格：技能、能力、经历和教育；

职位的报酬和奖励：工资、晋升和内在满足。

职位评估是企业人力资源管理里面非常重要的一个环节，是一个系统客观地度量职位相对价值的级别评估方法。它是以职位为中心，依据一定的标准来判断不同职位对组织的价值大小，并根据此结果建立职位序列。企业通过这个职位序列明确各类职位之间的横向关系，设计企业的薪资框架，建立员工的职业发展通道。职位评估的作用包括：

1) 职位评估为企业员工提供了一个工作行为的“标杆”；

2) 职位评估不仅可以使企业清楚招聘的雇员具体需要完成什么任务，对雇员而言可以用白纸黑字明确职责。

目前比较常见的职位评估方法为要素评估法。不同的咨询公司会采用自己的

职位评估系统，但所选择的要素基本上大同小异。一般包括职位职责大小、职位范围、任职资格、影响范围等。本系统采用北京华夏基石人力资源顾问公司的一套职位评估方法，这套方法经实践证明是一套行之有效的职位评估方法，被华为技术、白沙集团、南方航空公司、中国技术研究院等多家大型企业成功应用。

职位评估的结果是企业招聘、薪资福利以及员工职业生涯规划、企业决策的实施的一个有效的参考和基础。企业的每个职位都由三个维度以及构成维度的要素组成，对职位的评估就是对职位维度、维度要素的分析和评价，最后以总分、差分、乘积、外型、职位等级等的实际评分体现出来，并作为一个非常有力的科学的依据，供专业的咨询师或企业人力资源管理人员在实际的企业运作中加于使用。

一、职位评估模块要完成的功能：

- 1) 通过评估得出每一个职位的评估结果；
- 2) 实现对同一职位多个评估人的评估结果的汇总；
- 3) 打印评估结果；
- 4) 查询评估结果（支持简单查询和复合查询）；

二、职位评估中的几个专业术语<sup>[12]</sup>：

评估指引表：是评估职位的过程中要用到的以数学为基础针对人力资源生成的职位评估参照表，职位评估要涉及的表有知能指引表、解决问题（思考力）指引表、应负职责指引表、级数对照表；

维 度： 构成职位评估指引表的因素，在职位评估中指的是知能、解决问题（思考力）和应付职责三个部分，每一个维度在职位评估完成后会产生一个值；

根据公式可以算出每个职位对应的总分和差分。

要 素： 是指构成维度的更细的组成元素。

三、职位评估模块数据表的设计及其关系图：

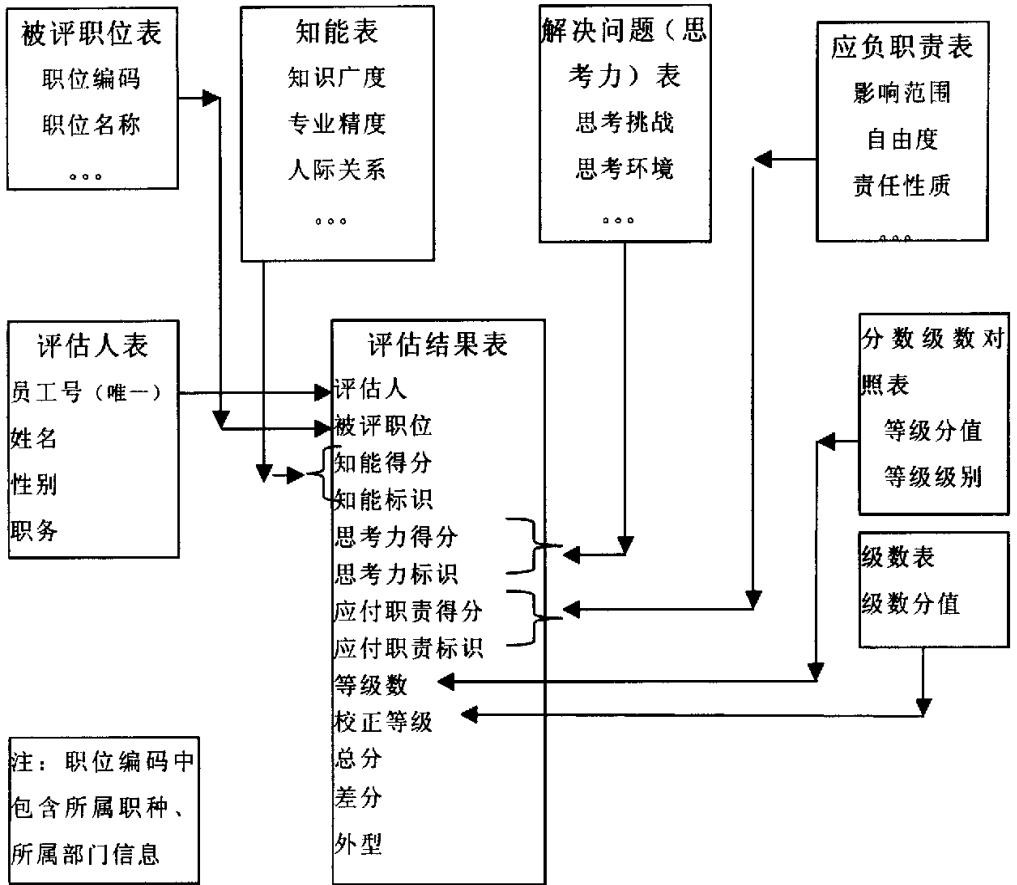


图 4-15 职位评估模块数据表的关系图：

#### 四、评估模式

##### 1) 职位评估模式有两种模式：按职种评、按部门评。

###### (1) 按职种进行评估：系统默认的评估方式。

系统一进入系统自动切换到按职种进行评估状态，参评职种将列出企业内部所有的职种供用户选择，每一个职种对应属于本职位的所有职位，用户一旦选择了职种后，参评职位将显示所有属于此职种的职位供用户选择以进行评估，同时，系统自动显示被选中的职位所在的部门，一个职位对应一个部门，一个部门可以对应很多个职位。

###### (2) 按部门进行评估：

用户也可以选择按部门进行职位评估，系统将所有的职位归属为不同的部门，部门名和职位名可以唯一确定职位的全部信息，用户在部门下拉组合框选取指定

部门后，职位下拉组合框会显示出所有属于该部门的全部职位，当用户选择要评估的职位后，系统自动显示出该部门的该职位所属职种。

2) 职位评估模式分类依据：

职位评估模式分类依据是以职务体系和企业组织结构为基础的，同一员工在组织结构中属于某个部门，某个部门包含职务体系的一个或多个职位，每个职位都属于组织结构里的某个部门，所以职位是职务体系和企业组织结构的连接点。如图 4-16。

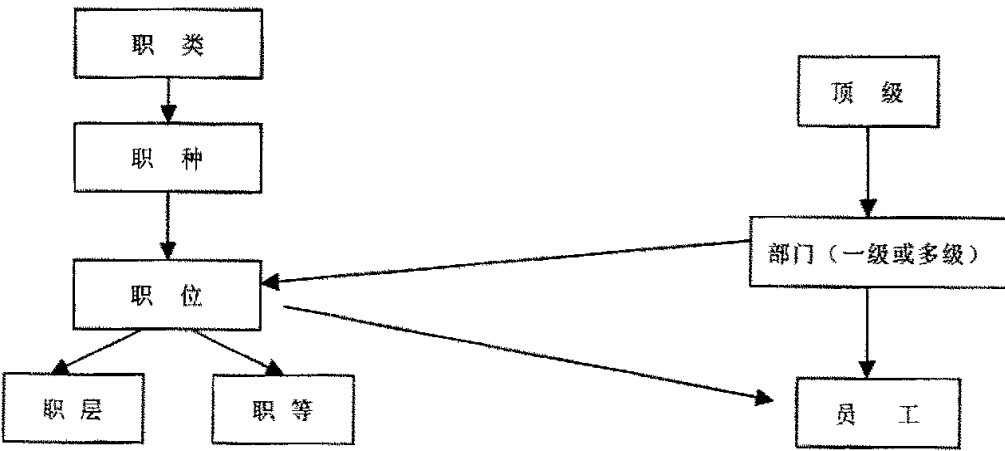


图 4-16 职务体系和企业组织结构关系图

五、评估过程：

(1)职位评估

职位评估是对每一个职位的维度及其各个要素进行的评估，每一个职位都包含三个维度：知能维度、解决问题维度、应负职责维度。

每个维度又包含若干个要素项，用户通过对每个维度的各要素的评估，形成维度标识值，根据维度标识值再对照各维度的各要素评估表，计算出维度的分值，然后计算出总分、差分、乘积、再查表外形表和职位等级表计算出该职位的外形值以及职位等级值等，最后的结果就形成整个职位的评估结果。其中总分、差分、乘积的计算公式为：

总分=知能分值+知能分值\*解决问题分值+应负职责分值

乘积=知能分值\*解决问题分值

差分=知能分值\*解决问题分值-应负职责分值

(2)保存评估结果



当三个维度的各个要素评估完成后，必须把评估结果保存起来，在保存的过程中系统会进行相关的计算，评估结果和计算的结果将写入数据库中，评估结果可以在系统中进行查询和汇总。

### (3) 查询评估结果（分两种）

可以指定评估人查询所有属于此人评估过的职位，指定评估人查询的结果不可进行汇总，因为同一评估人对同一职位只有一个评估结果。

也可以根据职位名称、所属部门、职种类别进行组合查询，查询的结果可以进行汇总，因为对同一职位可检索出多个评估结果。

### (4) 汇总

在评估的过程中，当同一个职位有多个结果时，就可以进行综合汇总，以便找到最适合的结果。汇总时必须先用复合查询找到参加汇总的记录，然后再执行，汇总的记录必须两条或两条以上，否则会有出错信息。汇总后的结果，评估人这一项统统用“汇总结果”加以标识以示区别。

评估流程图如图 4-17 所示：

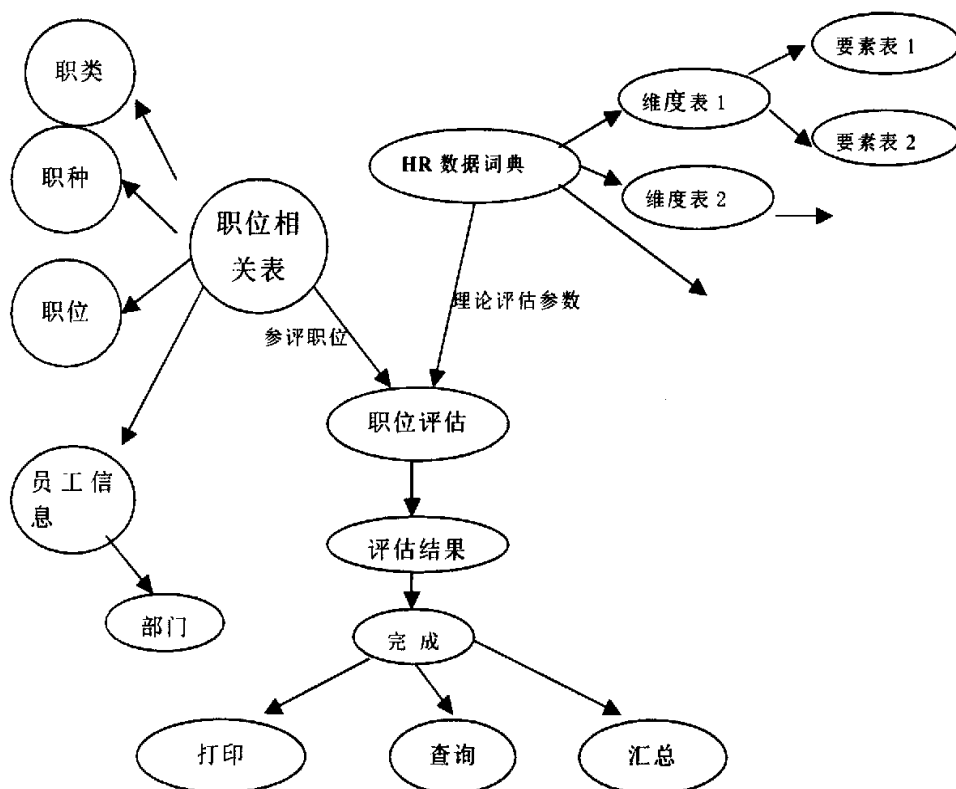


图 4-17 职位评估流程图

## 4. 5eHR 之公共信息服务与管理子系统的详细设计与实现

### 4. 5. 1 文档管理子系统

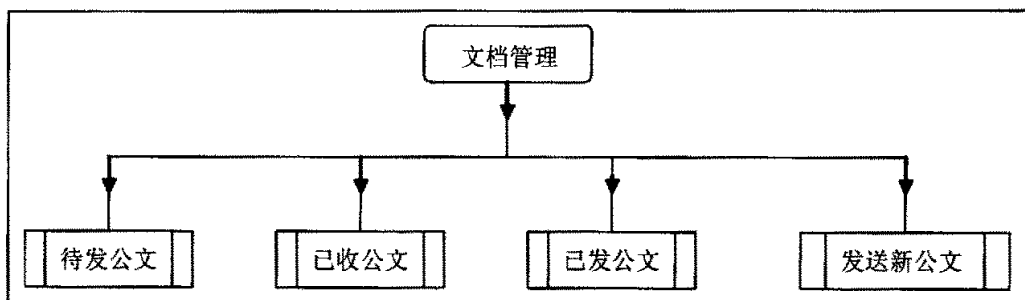


图 4-18 文档管理功能结构图

本部分主要是采用动态网页技术来实现基本的公文文书处理，后台使用 SQL Server 2000 的支持。

(1) 待发公文：准备发给同事或上下级的公文存放处；在“发送新公文”处编辑好的公文，可直接按“保存”系统将自动保存在待发公文处。

(2) 已收公文：可接收单位同事或上下级发来的公文，来文可按保密等级、发文单位、主题词、文件标题、文件内容、日期进行检索。

(3) 已发公文：对已发送的公文作记录，或对已发的公文稍加编辑，再转发给别人。

(4) 发送新公文：包括公文的起草及办理（发送公文、发文拟稿、内部审核、文件类型），文件类型分为：特急、急办、一般三个等级。

#### 1、公文系统的技术指标分类

技术指标包括：公文是否已阅、保密等级、发文单位、主题词、文件标题、文件内容、发文日期。

#### 3、Sql 数据表如下：

表 4-4-1: file\_district（发文单位表）

序号	中文字段	英文字段	类型	备注
1	Id 号	fid	Int(4)	
2	发文单位	File_district	Varchar(50)	

说明：发文单位可通过系统动态添加

表 4-4-2: file\_read (阅读表)

序号	中文字段	英文字段	类型	备注
1	Id 号	id	Int(4)	不为空值
2	文件是否已阅	Plan_read	Char(10)	已阅 1,未阅 0

表 4-4-3: file\_seclevel (保密等级)

序号	中文字段	英文字段	类型	备注
1	Id 号	id	Int(4)	不为空值
2	保密等级	Sec_name	Varchar(10)	机密, 一般,绝密
3	等级英文标识	Sec_title	Varchar(50)	

4.5.2 邮件发送接收子系统

提供用户以电子邮件方式进行联系。邮件系统可直接支持内、外部邮件的收发,并配合以相应的信息提示<sup>[13]</sup>。邮箱系统的操作界面采用的是类 Outlook 窗口。具体分布如下子功能:

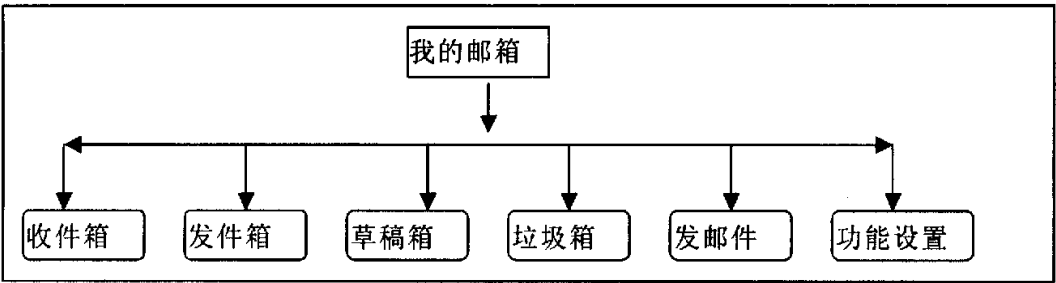


图 4-19 我的邮箱结构图

1、 Sql 数据表如下:

表 4-4-4: mail\_box (邮箱表)

序号	中文字段	英文字段	类型	备注
1	Id 号	Id	Int(4)	
2	信件来自	Mail_from	Varchar(100)	
3	发信人	Mail_formname	Varchar(100)	
4	收信人	Mail_to	Varchar(500)	
5	群发	Mail_toall	Varchar(5000)	
6	抄送	Mail_cc	Varchar(500)	
7	暗送	Mail_bcc	Varchar(500)	
8	标题	Mail_subject	Varchar(100)	
9	内容	Mail_body	Text(16)	
10	附件	Mail_attach	Varchar(5000)	
11	信件大小	Mail_size	Int(4)	
12	发送等级	Mail_level	Char(1)	
13	发信日期	Mail_date	Datetime(8)	
14	信件阅读情况	Mail_visited	Bit(1)	0 表示未阅邮件, 1 表已阅邮件
15	信件类型	Mail_type	Char(1)	
16	收件箱	Mail_inbox	Bit(1)	默认值(1)
17	草稿箱	Mail_draft	Bit(1)	
18	垃圾箱	Mail_rubbish	Bit(1)	
19	已发邮件所属用户	Mail_inoutbox	Varchar(50)	
20	草稿所属用户	Mail_indraft	Varchar(50)	
21	发件箱	Mail_outbox	Bit(1)	

表 4-4-5: mail\_innerset (内部邮箱设置表)

序号	中文字段	英文字段	类型	备注
1	Id 号	id	Int(4)	
2	用户 id	User_id	Int(4)	
3	用户名	User_name	Varchar(50)	
4	用户电子邮箱	User_email	Varchar(50)	
5	SmtP 服务器	User_smtpserver	Varchar(50)	
6	SmtP 端口	User_smtpport	Smailint(2)	
7	Stmp 认证	User_issmtP	Char(1)	
9	SmtP 用户名	User_smtpuser	Varchar(20)	
10	SmtP 密码	User_smtpPass	Varchar(50)	
11	用户邮件账号	User_accout	Varchar(50)	
12	用户邮箱密码	User_password	Varchar(50)	
13	邮件服务器	User_mailserver	Varchar(50)	
14	Pop3 端口	User_pop3port	Smailint(2)	

## 2、活动目录 AD 的原理、结构及其组成

活动目录 AD(Active Directory<sup>TM</sup>)是 Windows 2000 完全实现的目录服务,也是 Windows 2000 网络体系的基本结构模型<sup>[14]</sup>。目录服务并不是一个新概念,随着这几年网络规模的迅速增长,迫切需要一种能够有效地管理网络系统资源的手段,一方面可以简化网络管理,同时也加强管理工作的控制能力。目录服务就是在这样的应用背景下发展起来的,目录就是存放各项网络资源的所在地址,目录服务则是提供访问这些资源的服务,它管理网络中所有的实体信息,比如计算机、用户、打印机、文件、应用等,它也充分考虑了现代应用的业务需求,为这些应用提供了基本的管理对象模型,比如用户账户对象具有办公电话、手机、呼机、住址、上司、下属、电子邮件等属性。几乎所有的应用可以直接利用系统提供的目录服务结构,而且活动目录具有很好的扩充能力,允许应用程序定制目录中对象的属性或者添加新的对象类型。同时目录服务也提供了命名、描述、查找、访问以及保护这些实体信息的一致的方法。简单来讲,目录服务提供了以下一些好处:

1.简化管理。目录服务采用集中管理的手段,便于管理员对网络资源进行一致的管理。

2.加强了安全性。目录服务增加了网络管理员的控制能力,目录服务与操作系统提供的安全机制结合起来能够有效地保护网络资源。

3. 标准化特性。目录服务提供了一个基础平台，它不仅要考虑与其他系统模块之间的标准接口能力，还必须考虑与其他网络系统和网络应用的互操作能力，这就使得目录服务必须走向标准化。

活动目录的逻辑结构非常灵活，它为活动目录提供了完全的树状层次结构视图，为用户和管理员查找、定位对象提供了极大的方便。

活动目录中的逻辑单元包括：域、组织单元(Organizational Unit，简称 OU)、域树、域林。

活动目录的物理结构则侧重于网络的配置和优化、着眼于活动目录信息的复制和用户登录网络时的性能优化。

### 3、ADSI 接口和 LDAP 协议

Windows 2000 为活动目录提供了标准的面向对象的接口 ADSI(Active Directory Service Interface)<sup>[15]</sup>，管理员可以用一些简单的语句把管理工作描述出来，然后由操作系统提供的脚本运行进程负责执行管理任务。

同时，活动目录使用 LDAP(Light-weight Directory Access Protocol)目录访问协议作为它与其他应用或者目录服务交换信息的手段。LDAP 已经成为目录服务的标准，在 Windows 2000 的活动目录服务中将提供更为全面的支持，LDAP 协议采用命名的格式，通过名字信息访问目录对象。如果用户或者应用要使用某个目录对象，那么他必须要知道该对象的名字或者对象的属性。目前在目录服务中可以使用多种名字格式，有的是标准定义的格式，而有的则是事实上的标准，活动目录支持大多数的格式类型。

### 4、创建 AD 用户的代码：<sup>[16]</sup>

—打开对象

```
Set objContainer = GetObject("LDAP://" & ServerName & "/CN=" & site & ", " & DomainName)
```

—创建帐号

```
Set objUser = objContainer.Create("User", recip)
ObjUser.Put "samAccountName", emailname '帐号
objUser.Put "sn", LastName '用户姓
objUser.Put "givenName", FirstName '用户名
objUser.Put "DisplayName", emailname & "@hina.org" '显示的名字
objUser.Put "mail", emailname & "@&edm&" '邮箱名
objuser.put "Description", Description '信息描述
objUser.Put "userPrincipalName", emailname & "@&edm&" '用户登录帐号
objUser.SetPassword password '用户密码
objUser.AccountDisabled = False
```

objUser.SetInfo '保存信息到 AD

5、 创建邮箱的代码

```
Set objContainer = GetObject("LDAP://" & ServerName & "/CN="&site&","
& DomainName)
```

//获取操作系统容器对象信息

```
Set objMailbox = objUser // objUser 为 AD 已经创建了的的用户对象
```

```
objMailbox.CreateMailbox "LDAP://"&servername&"/" & _
```

```
"CN=Mailbox Store ("&org&"),CN=First Storage Group," & _
```

```
"CN=InformationStore,CN="&org&","CN=Servers," & _
```

```
" CN=First Administrative Group,CN=Administrative Groups," & _
```

```
"CN="&exchange&","CN=Microsoft Exchange,CN=Services," & _
```

```
"CN=Configuration,"&domainname&"" //利用 LDAP 协议创建邮箱
```

```
objuser.SetInfo //更新用户对象，在 AD 中写入新建邮箱信息
```

## 4. 6eHR 系统数据库主要存储过程和数据库公共访问类的实现

### 4. 6. 1 数据库主要存储过程的实现

#### 1) 添加用户 AddUser 存储过程

该存储过程在用户注册功能中使用，用于向数据库中添加一个新用户，用户必须提供用户名和密码等信息，如果添加成功则返回此用户的 ID。

```
CREATE Procedure AddUser
```

```
(
```

```
@UserName nvarchar(14),
```

```
@EngName nvarchar(20),
```

```
@Password nvarchar(12),
```

```
@Sex char(2),
```

```
@EmailStore int(3),
```

```
@Depart nvarchar(50),
```

```
@Roles nvarchar(200),
```

```
@ID int OUTPUT
```

```
)
```

```
AS
```

```
INSERT INTO Users
```

```
( UserName,
```

```
EngName,
```

```

        Password,
        Sex,
        EmailStore,
        Depart,
        Roles
    )
VALUES
(
    @UserName,
    @EngName,
    @Password ,
    @Sex,
    @EmailStore ,
    @Depart,
    @Roles,
)
SELECT
    @ID = @@Identity
GO

```

## 2) 用户合法性检查 CheckUserName 存储过程

此存储过程在注册时被调用，用来判断新注册的用户名是否已经被使用过，如果被使用则必须选择另外的用户名，因为在 USER 表中，用 EngName 唯一标识每一个用户。

```

CREATE Procedure CheckUserName
(
    @EngName nvarchar(20)
)
AS
SELECT
    *
FROM
    users
WHERE
    @EngName=EngName
GO

```



### 3) 用户信息显示 DisplayUserInfo 存储过程

调用此过程可以查看到登陆用户的相关信息。

```
CREATE Procedure DisplayUserInfo
```

```
(
```

```
    @EngName nvarchar(20)
```

```
)
```

```
AS
```

```
SELECT
```

```
    UserName,
```

```
    EngName,
```

```
    Password,
```

```
    Sex,
```

```
    EmailStore,
```

```
    Depart,
```

```
Roles
```

```
FROM
```

```
    users
```

```
WHERE
```

```
    @EngName=UserName
```

```
GO
```

### 4) 用户信息修改 UserInforModify 存储过程

用户通过调用此存储过程来修改数据库中的个人登录信息，除用户登录名以外，其它信息都可以修改，包括自己修改或授权修改两部分。

```
CREATE Procedure UserInforModify
```

```
(
```

```
    @UserName nvarchar(14),
```

```
    @EngName nvarchar(20),
```

```
    @Password nvarchar(12),
```

```
    @Sex char(2),
```

```
    @EmailStore int(3),
```

```
    @Depart nvarchar(50),
```

```
    @Roles nvarchar(200),
```

```
)
```

As

UPDATE users

set

@Password=Password,

@Sex=Sex,

@EmailStore=EmailStore,

@Depart=Depart,

@Roles=Roles,

WHERE

EngName = @EngName

GO

#### 5) 用户登陆 UserLogin 存储过程

该存储过程用于用户登陆时进行身份人证，只有输入的用户名和密码同时和注册信息相匹配的用户才可以登陆成功，否则为登陆失败，不能进入 EHR 系统。

CREATE Procedure CheckUser

(

@EngName nvarchar(20),

@Password nvarchar(12),

)

As

Select

EngUsers,

Password,

From

user

WHERE

@EngName =EngName

And

@Password=Password

GO

## 4.6.2 数据库公共访问类的实现

//定义数据库操作过程中用到的全局和私有变量

```
private SqlConnection MyConn;
```

```
public string MyConnString = "data source=localhost;Initial Catalog=ehr;User ID=sa;Password=";
```

// data source 为数据库所在服务器的名字, Initial Catalog为数据库的名字, 为数据库名称, User Id为登录数据库的用户名, Password为登录数据库的密码

### 1、open函数

OPEN方法用于打开到数据库的连接

```
public SqlConnection Open()
{
    MyConn = new SqlConnection(MyConnString);
    MyConn.Open();
    return( MyConn );
}
```

### 2、GetDataReader函数

GetDataReader方法使用DataReader从数据库中读取数据, 输入参数为连接数据库的字符串, 返回一个DataReader数据集

```
public SqlDataReader GetDataReader(string Commandtring)
{
    SqlCommand CmdCommand=new
SqlCommand(Commandtring,MyConn);
    SqlDataReader MyDataReader=CmdCommand.ExecuteReader();
    return MyDataReader;
}
```

### 3、ExcuteSQL函数

ExcuteSQL方法, 执行SQL语句, 并返回执行结果, 返回影响数据库的行数。

```
public int ExcuteSQL(string Commandtring)
{
    SqlCommand CmdCommand=new
SqlCommand(Commandtring,MyConn);
    int nAffected=CmdCommand.ExecuteNonQuery();
    return nAffected;
```

```
}
```

#### 4、GetDataSet函数

GetDataSet方法使用DataSet从数据库的指定表中读取数据，返回一个存放结果的数据集

```
public DataSet GetDataSet(string Commandrtring, string TableName)
{
    SqlDataAdapter adAdapter = new SqlDataAdapter();
    adAdapter.SelectCommand=new
SqlCommand(Commandrtring,MyConn);
    DataSet drDataSet=new DataSet();
    adAdapter.Fill(drDataSet,TableName);
    return drDataSet; //结果返回到数据集
}
```

#### 5、Close函数

关闭到数据库的连接

```
public void Close()
{
    this.MyConn.Close();
}
```

#### 6、AddUser函数

AddUser方法调用AddUser存储过程，用于往数据库里面插入一条新用户的记录，并返回唯一的Id值。

```
public String AddUser(string UserName, string EngName,string Password, string
Sex, int EmailStore, string Depart, string Roles)
```

```
{
    //创建数据库连接和命令的对象
    SqlConnection myConnection = new
SqlConnection(ConfigurationSettings.AppSettings["ConnectionString"]);
    SqlCommand myCommand = new SqlCommand("AddUser",
myConnection);
    //指明Sql命令的操作类型是使用存储过程
    myCommand.CommandType = CommandType.StoredProcedure;
    //给存储过程添加参数
    SqlParameter parameterUserName = new SqlParameter("@UserName",
SqlDbType.NVarChar, 14);
```

```

parameterUserName.Value = UserName;
myCommand.Parameters.Add(parameterUserName);
SqlParameter parameterPassword = new SqlParameter("@Password",
SqlDbType.NVarChar, 12);
parameterPassword.Value = Password;
myCommand.Parameters.Add(parameterPassword);
SqlParameter parameter EngName = new SqlParameter("@EngName ",
SqlDbType.NVarChar, 20);
parameterEngName.Value = EngName;
myCommand.Parameters.Add(parameterEngName);
SqlParameter parameterSex = new SqlParameter("@Sex", SqlDbType.
NVarChar, 2);
parameterSex.Value = Sex;
myCommand.Parameters.Add(parameterSex);
SqlParameter parameterEmailStore = new SqlParameter("@EmailStore",
SqlDbType.Int, 3);
parameterEmailStore.Value = EmailStore;
myCommand.Parameters.Add(parameterEmailStore);
SqlParameter parameterDepart = new SqlParameter("@Depart",
SqlDbType.NVarChar, 50);
parameterDepart.Value = Depart;
myCommand.Parameters.Add(parameterDepart);
SqlParameter parameterRoles = new SqlParameter("@Roles",
SqlDbType.NVarChar, 200);
parameterRoles.Value = Roles;
myCommand.Parameters.Add(parameterRoles);
SqlParameter parameterID = new SqlParameter("@ID", SqlDbType.Int, 4);
//指出该参数是存储过程的OUTPUT参数
parameterID.Direction = ParameterDirection.Output;
myCommand.Parameters.Add(parameterID);
try
{
    //打开数据库连接
    myConnection.Open();
    //进行数据库操作

```

```

        myCommand.ExecuteNonQuery();
        //关闭数据库连接
        myConnection.Close();
        //使用存储过程的输出参数返回UserID
        int UserId = (int)parameterID.Value;
        return UserId.ToString();
    }
    catch
    {
        return String.Empty;
    }
}

```

### 7、UserLogin函数

UserLogin函数用于判断登陆用户是否合法，根据输入的登陆用户名和密码调用UserLogin存储过程在数据库中查找是否有匹配的记录，如果存在则为合法且返回所有信息，如果不合法则返回空。

```

public SqlDataReader UserLogin(string EngName, string password)
{
    //创建数据库连接和命令的对象
    SqlConnection myConnection = new
    SqlConnection(ConfigurationSettings.AppSettings["ConnectionString"]);
    SqlCommand myCommand = new SqlCommand("UserLogin",
    myConnection);
    //指明Sql命令的操作类型是使用存储过程
    myCommand.CommandType = CommandType.StoredProcedure;
    //给存储过程添加参数
    SqlParameter parameterEngName = new SqlParameter("@EngName",
    SqlDbType.NVarChar, 20);
    parameterEngName.Value = EngName;
    myCommand.Parameters.Add(parameterEngName);
    SqlParameter parameterPassword = new SqlParameter("@Pass",
    SqlDbType.NVarChar, 20);
    parameterPassword.Value = password;
    myCommand.Parameters.Add(parameterPassword);
    //打开数据库连接
    myConnection.Open();
}

```

```

        //进行数据库操作
        SqlDataReader dr =
myCommand.ExecuteReader(CommandBehavior.CloseConnection);
        return dr;
    }

```

#### 8、CheckUserName函数

CheckUserName函数调用存储过程来判断新注册的用户所提交的登陆用户名是否已经被别人使用，如果没有则返回此用户名，否则返回空。

```

public SqlDataReader CheckUserName(string EngName)
{
    //创建数据库连接和命令的对象
    SqlConnection myConnection = new
SqlConnection(ConfigurationSettings.AppSettings["ConnectionString"]);
    SqlCommand myCommand = new SqlCommand("CheckUserName",
myConnection);
    //指明Sql命令的操作类型是使用存储过程
    myCommand.CommandType = CommandType.StoredProcedure;
    //给存储过程添加参数
    SqlParameter parameterEngName = new SqlParameter("@EngName",
SqlDbType.NVarChar, 20);
    parameterEngName.Value = EngName;
    myCommand.Parameters.Add(parameterEngName);
    //打开数据库连接
    myConnection.Open();
    //进行数据库操作
    SqlDataReader dr =
myCommand.ExecuteReader(CommandBehavior.CloseConnection);
    return dr;
}

```

#### 9、DisplayUserInfo函数

DisplayUserInfo函数调用存储过程根据登陆用户名取得用户的所有信息

```

public SqlDataReader DisplayUserInfo(string EngName)
{
    //创建数据库连接和命令的对象
    SqlConnection myConnection = new
SqlConnection(ConfigurationSettings.AppSettings["ConnectionString"]);

```

```

        SqlCommand myCommand = new SqlCommand("DisplayUserInfo",
myConnection);
        //指明Sql命令的操作类型是使用存储过程
        myCommand.CommandType = CommandType.StoredProcedure;
        //给存储过程添加参数
        SqlParameter parameterEngName= new SqlParameter("@EngName",
SqlDbType.NVarChar, 20);
        parameterEngName.Value = EngName;
        myCommand.Parameters.Add(parameterEngName);
        //打开数据库连接
        myConnection.Open();
        //进行数据库操作
        SqlDataReader dr
myCommand.ExecuteReader(CommandBehavior.CloseConnection);
        return dr;
    }

```

#### 10、UserInforModify函数

UserInforModify函数调用存储过程对用户的信息进行修改

```

public void UserInforModify(string UserName, string EngName,string Password,
string Sex, int EmailStore, string Depart, string Roles)
{
    //创建数据库连接和命令的对象
    SqlConnection myConnection = new
    SqlConnection(ConfigurationSettings.AppSettings["ConnectionString"]);
    SqlCommand myCommand = new SqlCommand("UserInforModify",
myConnection);
    //指明Sql命令的操作类型是使用存储过程
    myCommand.CommandType = CommandType.StoredProcedure;
    //给存储过程添加参数
    SqlParameter parameterUserName = new SqlParameter("@UserName",
SqlDbType.NVarChar, 14);
    parameterUserName.Value = UserName;
    myCommand.Parameters.Add(parameterUserName);
    SqlParameter parameterPassword = new SqlParameter("@Pass",
SqlDbType.NVarChar, 12);

```



```

parameterPassword.Value = Password;
myCommand.Parameters.Add(parameterPassword);
SqlParameter parameter EngName = new SqlParameter("@EngName ",
SqlDbType.NVarChar, 20);
parameterEngName.Value = EngName;
myCommand.Parameters.Add(parameterEngName);
SqlParameter parameterSex = new SqlParameter("@Sex", SqlDbType.
NVarChar, 1);
parameterSex.Value = Sex;
myCommand.Parameters.Add(parameterSex);
SqlParameter parameterEmailStore = new SqlParameter("@EmailStore",
SqlDbType.Int, 3);
parameterEmailStore.Value = EmailStore;
myCommand.Parameters.Add(parameterEmailStore);

SqlParameter parameterDepart = new SqlParameter("@Depart",
SqlDbType.NVarChar, 50);
parameterDepart.Value = Depart;
myCommand.Parameters.Add(parameterDepart);
SqlParameter parameterRoles = new SqlParameter("@Roles",
SqlDbType.NVarChar, 200);
parameterRoles.Value = Roles;
myCommand.Parameters.Add(parameterRoles);
try
{
    //打开数据库连接
    myConnection.Open();
    //进行数据库操作
    myCommand.ExecuteNonQuery();
    //关闭数据库连接
    myConnection.Close();
    return;
}
catch
{

```

```
        return;  
    }  
}
```

## 4.7 实施 eHR 过程中的安全问题及安全策略

### 4.7.1 可能的安全问题

#### 1、外部 Internet 的安全

由于广域网采用公网来进行数据传输，信息在广域网上传输时被截取和利用就比局域网要大得多。因此，必须采取必要的手段，使得在广域网上的发送和接收信息时能够保证：

- 1) 除了发送方和接收方外，其他人是无法知悉的（隐私性）
- 2) 传输过程中不被篡改（真实性）
- 3) 发送方能确知接收方不是假冒的（非伪装性）
- 4) 发送方不能否认自己的发送行为（不可抵赖性）

#### 2、内部 Intranet 的安全

由于局域网采用的是以广播为技术基础的以太网，任何两个节点之间的通信数据包，不仅为这两个节点的网卡所接收，也同时为处在同一以太网上的任何一个节点的网卡所截取。因此，只要接入以太网上的任一节点进行侦听，就可以捕获发生在这个以太网上的所有数据包，对其进行解包分析。从而窃取关键信息。这就是局域网固有的安全隐患。

### 4.7.2 采取的对策

#### 一、一般措施

##### 1、实体安全措施

就是要采取一些保护计算机设备、设施（含网络、通信设备）以及其他媒体免地震、水灾、火灾、有害气体和其他环境事故（如电磁污染）破坏的措施、过程。这是整个管理信息系统安全运行的基本要求。满足防火、防磁、防水、防盗、防电击、防虫害等要求，配备相应的设备。

##### 2、运行安全措施

为保障整个系统功能的安全实现，提供一套安全措施，来保护信息处理过程的安全，其中包括：风险分析、审计跟踪、备份恢复、应急等。

制定必要的、具有良好可操作性的规章制度，去进行制约，是非常必要和重要的，而且是非常紧迫的。

形成一支高长自觉、遵纪守法的技术人员队伍，是计算机网络安全工作的又一重要环节。要在思想品质、职业道德、经营、管理、规章制度、教育培训等方

面，做大量艰苦细致的工作，强化计算机系统的安全管理，加强人员教育，来严格有效地制约用户对计算机的非法访问，防范法用户的侵入。只有严格的管理，才能把各种危害遏止最低限度。

### 3、信息安全措施

数据是信息的基础，是企业的宝贵财富。信息管理的任务和目的是通过对数据采集、录入、存储、加工，传递等数据流动的各个环节进行精心组织和严格控制，确保数据的准确性、完整性、及时性、安全性、适用性和共享性。

制定良好的信息安全规章制度，是最有效的技术手段。而且不仅仅是数据，还应把技术资料、业务应用数据和应用软件包括进去。

## 二、防病毒措施

计算机病毒泛滥，速度之快，蔓延之广，贻害社会之大，为有史以来任何一种公害所无可比拟。从 CIH 到 红色代码和尼姆达，已充分说明了病毒的难以预知性、潜藏性和破坏性，另一方面也说明了防毒的重要性。

可以采用了瑞星网络看防病毒软件，运行在 Win2000 服务器上。

该软件是瑞星软件公司推出的新一代反病毒产品，瑞星杀毒运用主动内核技术，其基础是无缝连接技术，这种技术可以保证反病毒模块从底层内核，在发生病毒入侵反应时，反病毒操作不会伤害及到操作系统内核，同时确保杀灭来犯病毒。

此外，瑞星杀毒还有很强的网管能力，其可利用 Vxd 技术，进行实时反病毒。软件可实现自动安装，只要连接互联网，通过域管理方式可实现自动升级。

## 三、内部网络安全

### 1、针对局域网采取安全措施

为了解决这局域网固有的安全隐患，采取了以下措施：

1) 网络分段：由于局域网采用以交换机为中心、路由器为边界的网络格局，又基于中心交换机的访问控制功能和三层交换功能，综合应用物理分段与逻辑分段两种方法，来实现对局域网的安全控制，其目的就是将非法用户与敏感的网络资源相互隔离，从而防止可能的非法侦听，这是一重要的措施。

2) 以交换式集线器代替共享式集线器：由于部分网络最终用户的接入是通过分支集线器而不是交换机，而使用最广泛的分支集线器通常是共享式集线器。这样，当用户与主机进行数据通信时，两台机器之间的数据包还是会被同一台集线器上的其他用户所侦听。如一种危险的情况是：用户 TELNET 到一台主机上，由于 TELNET 程序本身缺加密功能，用户所键入的每一个字符（包括用户名、密码、关键配置等重要信息），都将被明文发送，这就是一个很大的安全隐患。

因此，应该以交换式集线器代替共享式集线器，使单播包仅在两个节点之间传送，从而防止非法侦听。

3) 划分 VLAN (虚拟专用网): 本系统中采取划分 VLAN 的方法, 进一步克服了以太网的广播问题。

目前的 VLAN 技术主要有三种: 基于交换机端口的 VLAN、基于节点 MAC 地址的 VLAN 和基于应用协议的 VLAN。基于交换机端口的 VLAN 虽然稍欠灵活, 但却比较成熟, 在实际使用中较多, 且效果显著。所以 eHR 系统采取了这种方式。

在集中式网络环境下, 我们是将中心的所有主机系统集中到一个 VLAN 里, 在这个 VLAN 里不允许任何用户节点, 从而较好地保护敏感的主机资源。在分布式网络环境下, 我们按机构或部门的设置来划分 VLAN。各部门内部的所有服务器和用户节点都在各自的 VLAN 内, 互不侵扰。VLAN 内部的连接采用交换实现, 而 VLAN 与 VLAN 之间的连接则采用路由实现。

## 2、强化服务器端的管理和安全措施

### 1) 加强对用户授权机制的管理

由于在这种安全体系中, 应用系统成为隔离用户和数据库的防火墙, 其本身就必须具备相当的安全特性。尤其是用户授权管理机制, 其严密将直接影响整个系统的安全, 所以必须加强用户授权机制的管理。

### 2) 采用智能型日志

日志系统具有综合性数据记录功能和自动分类检索能力。在该系统中, 日志将记录自某用户登录时起, 到其退出系统时止, 这所执行的所有操作, 包括登录失败操作, 对数据库的操作及系统功能的使用。日志所记录的内容有执行某操作的用户标识、执行操作的机器 IP 地址、操作类型、操作对象及操作执行时间等, 以备日后审计核查之用。

智能型日志不仅可以分类检索日志内容, 系统还能根据已记录的日志内容, 通过智能型扫描, 自动找出可能存在的不安全因素, 并实时触发相应的警告, 信息以及及时通知系统管理员及用户。

### 3) 完善备份及恢复机制

诚然, 日志能记录任何非法操作, 然而要真正使系统从灾难中恢复出来, 还需要一套完善的备份方案及恢复机制。

为了防止存储设备的异常损坏, 本系统中采用了可热插拔的 SCSI 硬盘所组成的磁盘容错阵列, 以 RAID5 的方式进行系统的实时热备份。

为了防止人为的失误或破坏, 本系统中建立了强大的数据库触发器以备份重要数据的删除操作, 甚至更新任务。保证在任何情况上, 重要数据均能最大程度地有效恢复。具体而言, 对于删除操作, 将被操作的记录全部存贮在备份库中。而对于更新操作, 考虑到信息量过于庞大, 仅仅备份了所执行的 SQL 语句。这样, 既能查看到被删的内容, 又能相当程度地减小备份库存贮容量。

而在需要跟踪追溯数据丢失或破坏事件的全部信息时, 则将系统日志与备份

数据有机地结合在一起真正实现系统安全性。

#### 四、外部 Internet 的安全

因为采用的是基于 TCP/IP 协议族, Internet 协议族自身的开放性极大地方便了各种计算机的组网和互联, 并直接推动了网络技术的迅猛发展。但是由于在早期网络协议设计上对安全性的忽视, 至使 Internet 在使用和管理上的无政府状态, 逐渐使 Internet 自身的安全受到威胁。

外网安全的威胁主要表现在: 非授权访问、冒充合法用户、破坏数据完整性、干扰系统正常运行、利用网络传播病毒、线路窃听等。

针对上述情况, 本系统采取了防火墙技术、入侵检测技术和网络防病毒技术相结合的方法。

#### 五、eHR 系统安全

- 1) 对数据库进行加密
- 2) 严格的权限管理, 设定用户对系统不同模块、子模块乃至数据项的不同级别操作权限
- 3) 建立日志文件, 跟踪记录用户对系统每一次操作的详细情况
- 4) 建立数据定期备份机制并提供数据灾难恢复功能

### 4.7. 3eHR 系统的数据加密技术及其实现

#### 4.7.3. 1eHR 软件系统的数据加密技术<sup>[17]</sup>

在 eHR 系统开发的过程中, 处于安全和保护个人隐私的需要, 对用户的登录信息和对某些特定的数据信息比如企业机密级文件和密级商业信息, 甚至是对开发程序的源程序, 都有必要经过一定的加密处理, 以确保信息不会被人为地恶意破坏, 从而造成不可挽回的损失。总结起来, 加密用于达到以下目的:

- 1) 保密性: 防止用户的标识或数据被读取。
- 2) 数据完整性: 防止数据被更改。
- 3) 身份验证: 确保数据发自特定的一方。

为此我们必须对密码知识有足够的了解并在系统的设计过程中运用到密码技术。

##### 1、密码技术的产生和发展

密码学是一门古老而深奥的学科, 密码技术包括加密技术和解密技术两个方面。它对一般人来说是陌生的, 因为长期以来, 它只在很少的范围内, 如军事、外交、情报等部门使用。

在古代, 目前记录的比较早的是一个在公元前 2 世纪, 由一个希腊人提出来的。

在近代, 在第二次世界大战里, 密码机得到了比较广泛应用的技术, 同时破

译密码的技术也得到了发展，出现了一次性密码技术。同时密码技术也促进了计算机的发展。

在现代，由于计算机和计算机网络的出现，特别是 internet 网络的兴起和电子商务的广泛应用，对密码技术提出了更高的要求，密码算法层出不穷，并开始走向国际化的道路，出现了 DES, AES 等国家(美国的)标准。同时各个国家和政府对密码技术也越来越重视，密码技术也得到不断的发展和应用。

## 2、对称和非对称加密技术

### 一、对称加密技术 (symmetric encryption)

对称加密技术的对称算法 (symmetric algorithm)，有时又叫传统密码算法，就是加密密钥能够从解密密钥中推算出来，同时解密密钥也可以从加密密钥中推算出来。而在大多数的对称算法中，加密密钥和解密密钥是相同的。所以也称这种加密算法为秘密密钥算法或单密钥算法。它要求发送方和接收方在安全通信之前，商定一个密钥。对称算法的安全性依赖于密钥，泄漏密钥就意味着任何人都可以对他们发送或接收的消息解密，所以密钥的保密性对通信性至关重要。

常用的采用对称密码术的加密方案有 5 个组成部分 (如图 4-20 所示)：

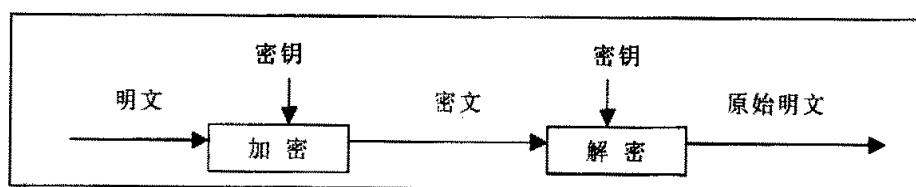


图 4-20 对称密码术的加密方案组成图

- 1) 明文：原始信息。
- 2) 加密算法：以密钥为参数，对明文进行多种置换和转换的规则和步骤，变换结果为密文。
- 3) 密钥：加密与解密算法的参数，直接影响对明文进行变换的结果。
- 4) 密文：对明文进行变换的结果。
- 5) 解密算法：加密算法的逆变换，以密文为输入、密钥为参数，变换结果为明文。

特点分析：

对称加密的优点在于算法实现后的效率高、速度快。

对称加密的缺点在于密钥的管理过于复杂。如果任何一对发送方和接收方都有他们各自商议的密钥的话，那么很明显，假设有  $N$  个用户进行对称加密通信，如果按照上述方法，则他们要产生  $N(N-1)$  把密钥，每一个用户要记住或保留  $N-1$  把密钥，当  $N$  很大时，记住是不可能的，而保留起来又会引起密钥泄漏可能性的

增加。

对称加密系统最著名的是美国数据加密标准 DES、AES(高级加密标准)和欧洲数据加密标准 IDEA。

1977 年美国国家标准局正式公布实施了美国的数据加密标准 DES, 公开它的加密算法, 并批准用于非机密单位和商业上的保密通信。随后 DES 成为全世界使用最广泛的加密标准。加密与解密的密钥和流程是完全相同的, 区别仅仅是加密与解密使用的子密钥序列的施加顺序刚好相反。

AES(高级加密标准)内部有更简洁精确的数学算法, 而加密数据只需一次通过。AES 被设计成高速, 坚固的安全性能, 而且能够支持各种小型设备。

AES 和 DES 的性能比较:

- (1) DES 算法的 56 位密钥长度太短;
- (2) DES 算法有不安全的因素, 有些算法已经被破解;
- (3) AES 算法设计简单, 密钥安装快、需要的内存空间少, 在所有平台上运行良好, 支持并行处理, 还可抵抗所有已知攻击;
- (4) AES 很可能取代 DES 成为新的国际加密标准。

总之, AES 比 DES 支持更长的密钥, 比 DES 具有更强的安全性和更高的效率, 比较一下, AES 的 128bit 密钥比 DES 的 56bit 密钥强 1021 倍。随着信息安全技术的发展, 已经发现 DES 很多不足之处, 对 DES 的破解方法也日趋有效。AES 会代替 DES 成为 21 世纪流行的对称加密算法。

## 二、非对称加密技术

非对称加密 (dissymmetrical encryption), 有时又叫公开密钥算法 (public key algorithm)。这种加密算法是这样设计的: 用作加密的密钥不同于用作解密的密钥, 而且解密密钥不能根据加密密钥计算出来 (至少在合理假定的长时间内)。之所以又叫做公开密钥算法是由于加密密钥可以公开, 即陌生人可以得到它并用来加密信息, 但只有用相应的解密密钥才能解密信息。在这种加密算法中, 加密密钥被叫做公开密钥 (public key), 而解密密钥被叫做私有密钥 (private key)。

常用的采用非对称密码术的加密方案有 5 个组成部分 (如图 4-21 所示):

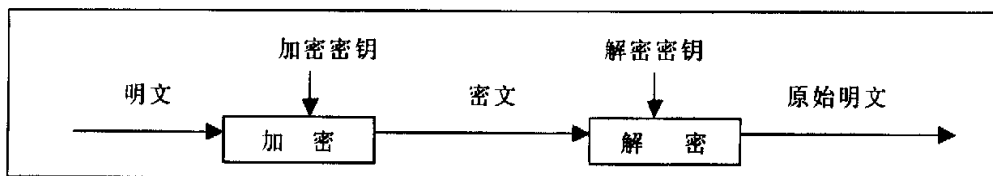


图 4-21 非对称密码术的加密方案组成图

特点分析：

非对称加密的缺点在于算法实现后的效率低、速度慢。

非对称加密的优点在于用户不必记忆大量的提前商定好的密钥，因为发送方和接收方事先根本不必商定密钥，发放方只要可以得到可靠的接收方的公开密钥就可以给他发送信息了，而且即使双方根本互不相识。但为了保证可靠性，非对称加密算法需要一种与之相配合使用的公开密钥管理机制，这种公开密钥管理机制还要解决其他一些公开密钥所带来的问题。

自非对称加密问世以来，学者们提出了许多种非对称加密方法，它们的安全性都是基于复杂的数学难题。根据所基于的数学难题来分类，有以下三类系统目前被认为是安全和有效的：

大整数因子分解系统(代表性的有 RSA)、椭圆曲线离散对数系统(ECC)和离散对数系统 (代表性的有 DSA)。

当前最著名、应用最广泛的非对称加密系统 RSA 是由 Rivet、Shamir、Adelman 提出的(简称为 RSA 系统)，它的安全性是基于大整数素因子分解的困难性，而大整数因子分解问题是数学上的著名难题，至今没有有效的方法予以解决，因此可以确保 RSA 算法的安全性。RSA 系统是公钥系统的最具有典型意义的方法，大多数使用非对称加密密码进行加密和数字签名的产品和标准使用的都是 RSA 算法。

### 三、Rijndael 加密技术

2004 年 8 月 17 日的美国加州圣巴巴拉，正在召开的国际密码学会议(Crypto' 2004)来自山东大学的王小云教授做了破译 MD5、HAVAL-128、MD4 和 RIPEMD 四个著名 HASH 算法的破译结果的报告，引起密码界的巨大轰动，可见加密技术的选择在系统开发中有着至关重要的地位。

鉴于对称加密技术的高效率、快速度、高安全性以及适合应用于大型数据系统的特点，在 eHR 系统开发中我们选择 AES 的 Rijndael 加密技术作为对密码和数据信息的加密保护技术方案。

Rijndael 加密技术是美国国家技术标准委员会(NIST) 经过三年漫长的过程，最终从进入候选的五种方案中挑选出来的比利时的研究成果"Rijndael" (发音为 "Rhine dale")，是高级加密标准 (Advanced Encryption Standard) 的基础。

#### Rijndael 算法<sup>[18]</sup>

Rijndael 是带有可变块长和可变密钥长度的迭代块密码。块长和密钥长度可以分别指定成 128、192 或 256 位。

Rijndael 中的某些操作是在字节级上定义的，字节表示有限字段 GF(28) 中的元素，一个字节中有 8 位。其它操作都根据 4 字节字定义。

Rijndael 迭代块密码数据块要经过多次数据变换操作，每一次变换操作产生



一个中间结果，这个中间结果叫做状态。状态可表示为二维字节数组，它有 4 行，Nb 列，且 Nb 等于数据块长除 32。如图 4-22 所示。

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$	$a_{0,4}$	$a_{0,5}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$

图 4-22 变换状态表

数据块按  $a_{0,0}, a_{1,0}, a_{2,0}, a_{3,0}, a_{0,1}, a_{1,1}, a_{2,1}, a_{3,1}, a_{0,2} \dots$  的顺序映射为状态中的字节。在加密操作结束时，密文按同样的顺序从状态中抽取。

密钥也可类似地表示为二维字节数组，它有 4 行，Nk 列，且 Nk 等于密钥块长除 32。算法变换的圈数 Nr 由 Nb 和 Nk 共同决定，如图 4-23 所示。

Nr	Nb = 4	Nb = 6	Nb = 8
Nk = 4	10	12	14
Nk = 6	12	12	14
Nk = 8	14	14	14

图 4-23 Nb 和 Nk 决定的 Nr 的值

#### 4.7.3.2 Rijndael 加密技术在 eHR 系统中的应用和实现

在 eHR 实际开发和使用过程中，加密技术主要在用户信息、文档数据中起加密作用，有效地保证了信息的安全性。其中用户信息在创建用户成功的同时就对其登录密码进行加密，所有用户初始化的密码都是预先有具有管理权限的管理人员分配的，允许用户登录后根据自己的需要进行重新设置。其加密流程如下图 4-24 和图 4-25：

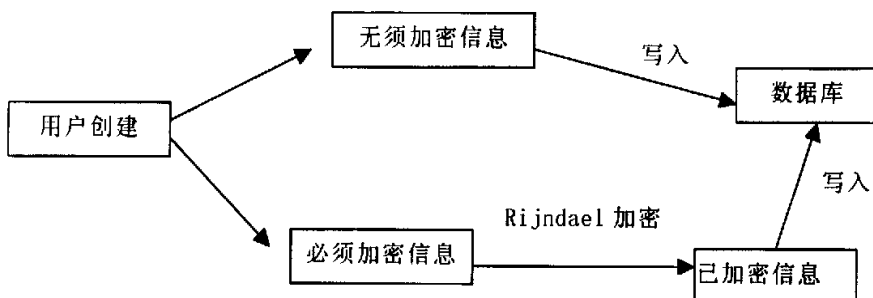


图 4-24 用户信息加密流程

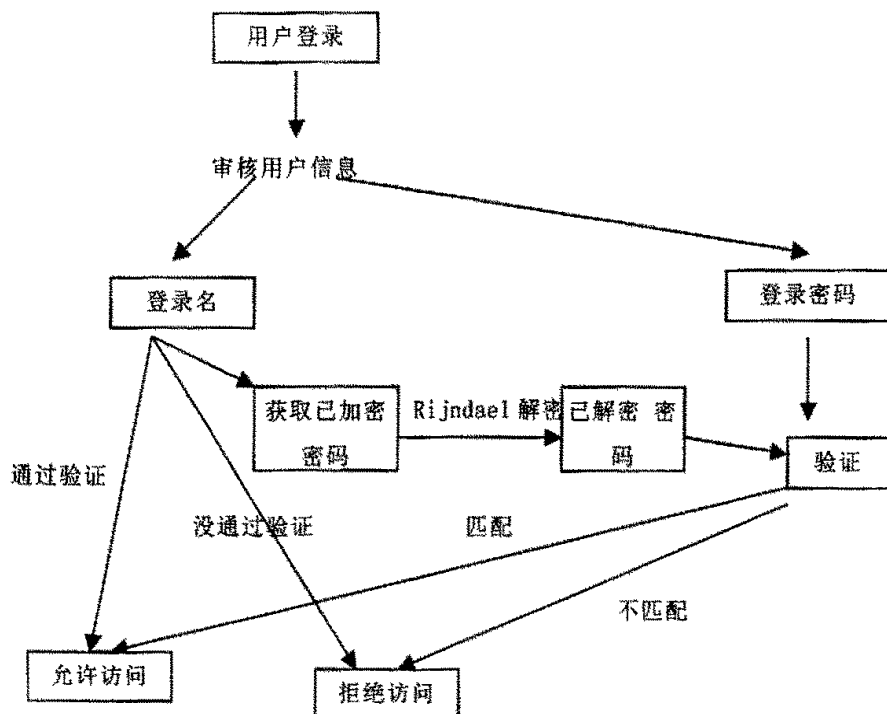


图 4-25 用户登录信息解密流程

.net 将所有常用的加密算法集成在类 System.Security.Cryptography 中，编程人员只要调用 Cryptography 中的相关的加密算法类就可以对所要求的数据进行加密<sup>[19]</sup>，以下是基于.net 技术加密解密实现的 C#程序：

#### 1、 字符加密程序

```

public string Encrypto(string Source)
{
    byte[] bytIn = UTF8Encoding.UTF8.GetBytes(Source);
    MemoryStream ms = new MemoryStream();
    mobjCryptoService.Key = GetLegalKey();
    mobjCryptoService.IV = GetLegalIV();
    ICryptoTransform encrypto = mobjCryptoService.CreateEncryptor();
    CryptoStream cs = new CryptoStream(ms, encrypto,
    CryptoStreamMode.Write);
    cs.Write(bytIn, 0, bytIn.Length);
    cs.FlushFinalBlock();
    ms.Close();
}
  
```

```
byte[] bytOut = ms.ToArray();
return Convert.ToBase64String(bytOut);
}
```

“Source” 是待加密的字符串

返回 Convert.ToBase64String(bytOut)是经过加密的字符串

## 2、 字符解密程序

```
public string Decrypto(string Source)
{
    byte[] bytIn = Convert.FromBase64String(Source);
    MemoryStream ms = new MemoryStream(bytIn, 0, bytIn.Length);
    mobjCryptoService.Key = GetLegalKey();
    mobjCryptoService.IV = GetLegalIV();
    ICryptoTransform encrypto = mobjCryptoService.CreateDecryptor();
    CryptoStream cs = new CryptoStream(ms, encrypto,
CryptoStreamMode.Read);
    StreamReader sr = new StreamReader(cs);
    return sr.ReadToEnd();
}
```

“Source” 是待解密的串

返回 sr.ReadToEnd()是经过解密的串

## 3、 加密文件的函数程序

```
public void encryption(string textBox,string readfile,string writefile)
{ byte [] key = System.Text.Encoding.Default.GetBytes( textBox );
byte [] iv = rij.IV;
Rijndael crypt = Rijndael.Create();
ICryptoTransform transform = crypt.CreateEncryptor(key ,iv);
    //写进文件
    FileStream fswrite = new FileStream(writefile,FileMode.Create);
    CryptoStream cs = new
CryptoStream( fswrite ,transform ,CryptoStreamMode.Write );
    //打开文件
    FileStream fsread = new FileStream(readfile,FileMode.Open);
    int length;
    while((length = fsread.ReadByte() )!= -1)
    cs.WriteByte((byte)length);
```

```

fsread.Close();
cs.Close();
fswrite.Close();
enresult = true ;//成功加密
MessageBox.Show("已经成功完成加密任务！");
}

```

textBox 是加密密码，readfile,是要加密的文件的文件名（包括路径），writefile 是加密后的文件的文件名。

#### 4、用于文件解密的函数程序

```

public void decryption(string textBox,string readfile,string writefile)
{
    byte [] key = System.Text.Encoding.Default.GetBytes( textBox );
    byte [] iv  = rij.IV;
    Rijndael crypt = Rijndael.Create();
    ICryptoTransform transform = crypt.CreateDecryptor(key,iv);
        //读取加密后的文件
    FileStream fsopen = new FileStream(readfile,FileMode.Open);
    CryptoStream          cs          =          new
CryptoStream( fsopen ,transform ,CryptoStreamMode.Read );
        //把解密后的结果写进文件
    FileStream fswrite = new FileStream(writefile,FileMode.OpenOrCreate);
    int length;
    while ((length = cs.ReadByte()) != -1)
    fswrite.WriteByte((byte)length);
    fswrite.Close();
    cs.Close();
    fsopen.Close();
    deresult=true; //成功解密
    MessageBox.Show("已经成功完成解密任务！");
}

```

textBox 是解密密码，因为使用的是对称加密技术，所以，解密密码和加密密码是一样的，readfile 是要解密的文件的文件名（包括路径），writefile 是解密后的文件的文件名。

## 第五章：eHR 系统的实际应用

### 5.1 系统登录

进入 eHR 系统，首先出现登录界面：

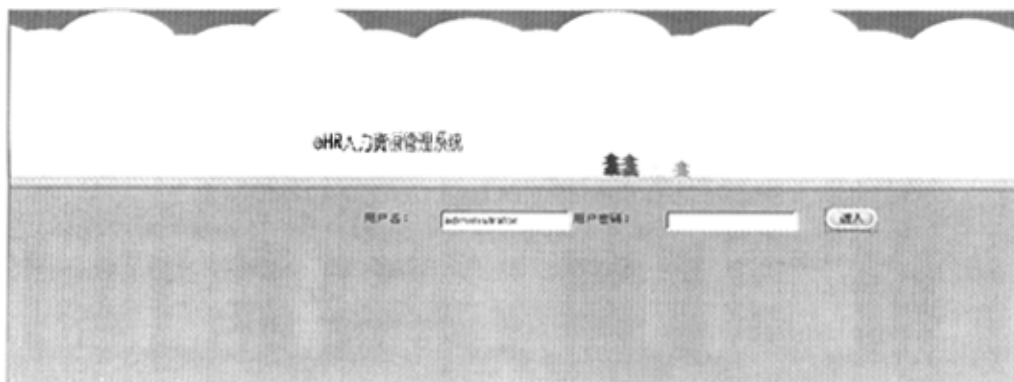


图 5-1 eHR 系统登录界面

输入用户名和密码，系统将验证用户的用户名和密码，通过验证后系统调用用户角色表提取权限，并根据权限显示系统的界面和相关菜单以及权限有关的功能键；如果没有通过验证，系统会有相关报错信息提示，直到改正错误为止。

### 5.2 主界面

当用户完成登录界面进入到系统主程序后，出现 eHR 系统的主界面。



图 5-2 eHR 系统主界面

主界面分为几个部分：导航区、下拉框功能菜单区、主显示区。

导航区包括：桌面、我的文档、政策法规、员工管理、自助系统、帮助、退出等，是进入系统功能模块的快捷方式；

下拉框功能菜单区包括：我的文档、我的邮箱、HR 工具、员工管理和系统设置以及系统设置和外部连接，展开菜单项有具体的系统子功能项，用户双击就可以执行相应的功能。

主显示区：这个区域用来显示执行子功能时的界面或显示执行结果，刚开始登录时显示最新的文件、企业新的公告（政策法规）和邮箱信息。

### 5.3 我的文档

包括五个子功能：待发文档、已收文档、已发文档、发送新文档。管理员可以根据权限来控制用户是否可以使用某个子功能项，如高级用户既可以接收也可以发送公文，如图 5-3，而一般员工则只可以接受公文和删除自己收到的公文，如图 5-4。

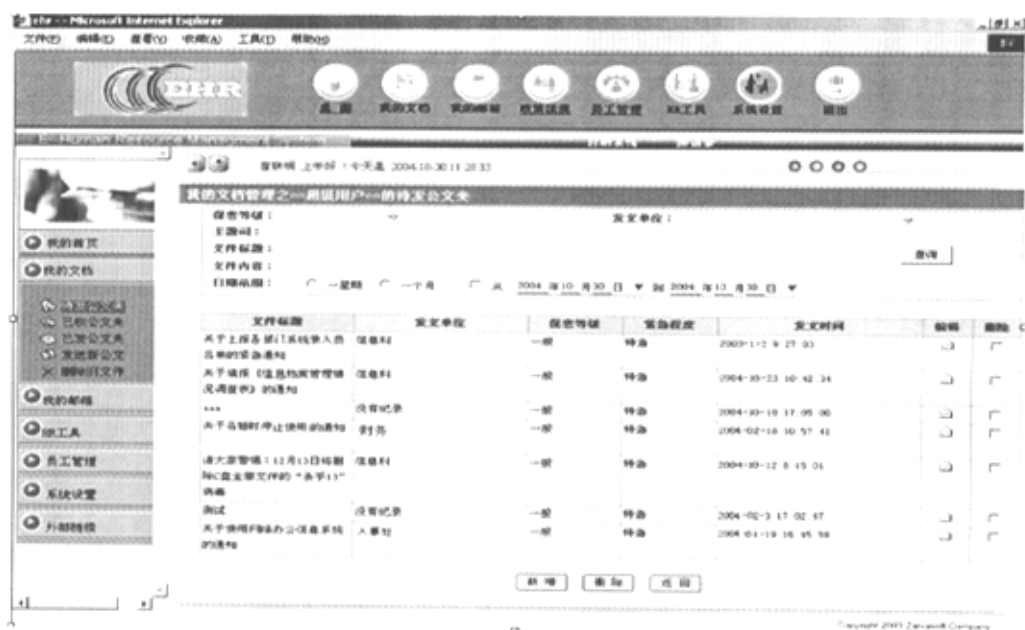


图 5-3 高级用户登录后“我的文档”界面



图 5-4 普通用户登录后“我的文档”界面

5.4 我的邮箱

包括常用的六个子功能：收件箱、发件箱、草稿箱、垃圾箱、发邮件和功能设置，如图 5-4。



图 5-5 “发送新邮件”的界面

“收件人”旁边有两个通讯录，分别是“部门通讯录”和“个人通讯录”，里面有整个企业内部个人和部门的 e\_mail 地址,发邮件时非常方便。被显示的地址自动加入到收件人地址栏里。支持多个地址群发，有多个地址时地址与地址之间有逗号区别开，如图 5-6。



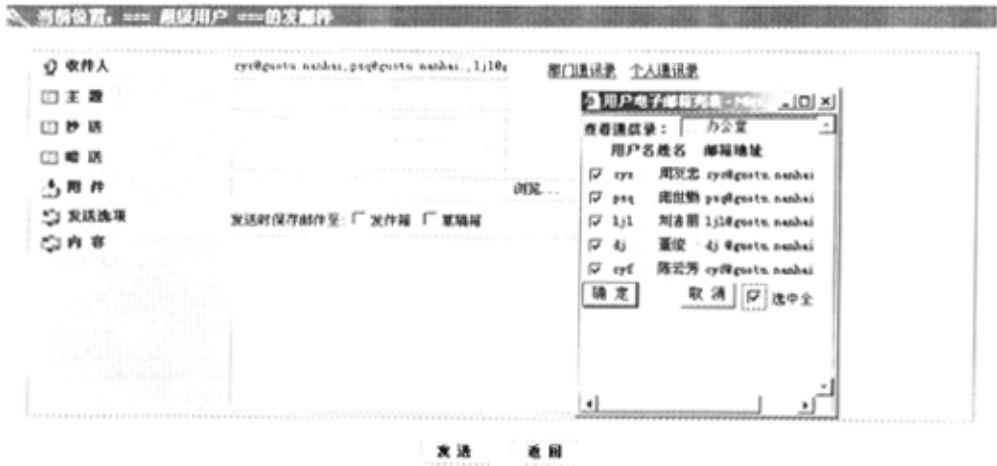


图 5-6 打开“部门通讯录”后的发送邮件界

用户还可以根据需要灵活设置邮箱的配置，如 5-7 图：

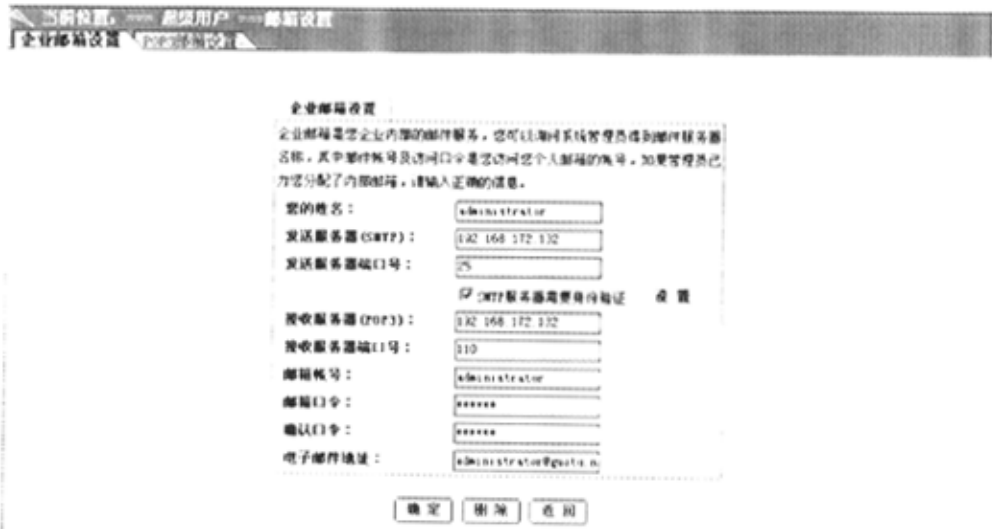


图 5-7 邮箱功能设置界面

## 5.5HR 专业工具

选择“职位评估”，可以实现“知能”维度(如图 5-8)和“解决问题”维度以及“应负职责”维度（如图 5-9）的评估

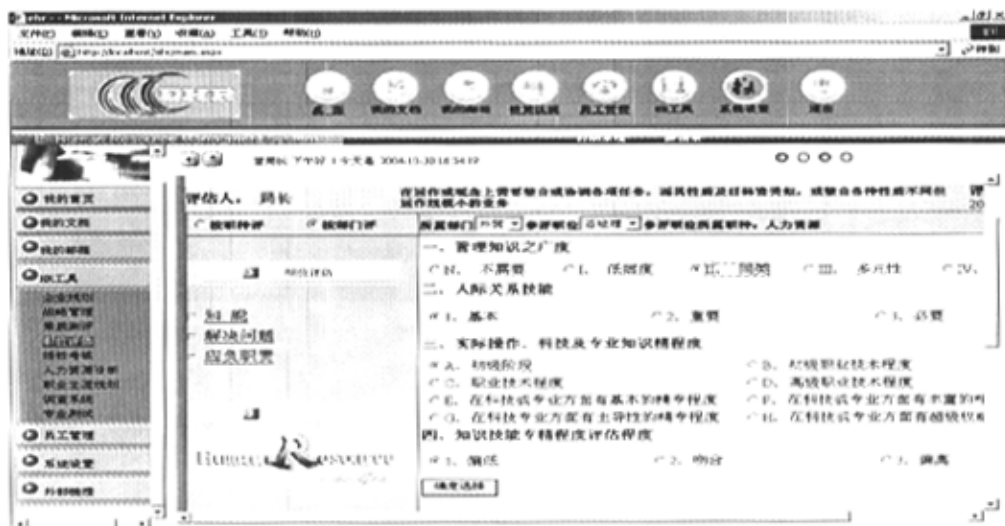


图 5-8 职位评估之“知能”维度的评估界面



图 5-9 职位评估之“应负职责”维度的评估界面

评估完成后, 评估结果马上可以显示出来, 评估人可以对评估结果进行组合查找和删除操作, 同一职位有多个结果时可以汇总成一个综合结果, 汇总后的综合结果直接写入数据库中, 评估人则显示为“汇总结果”, 如图 5-10。

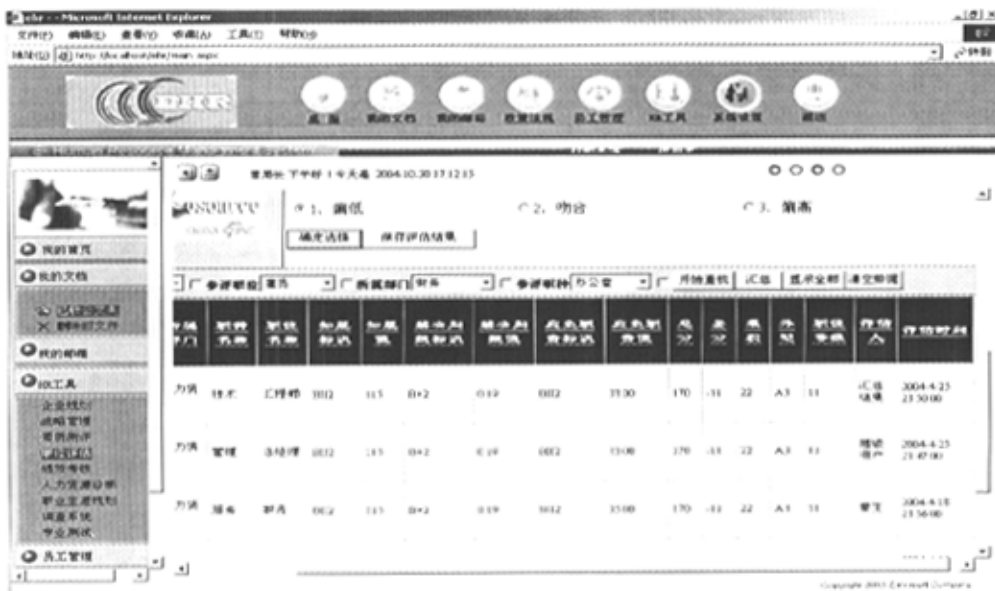


图 5-10 职位评估完成后的评估结果界面

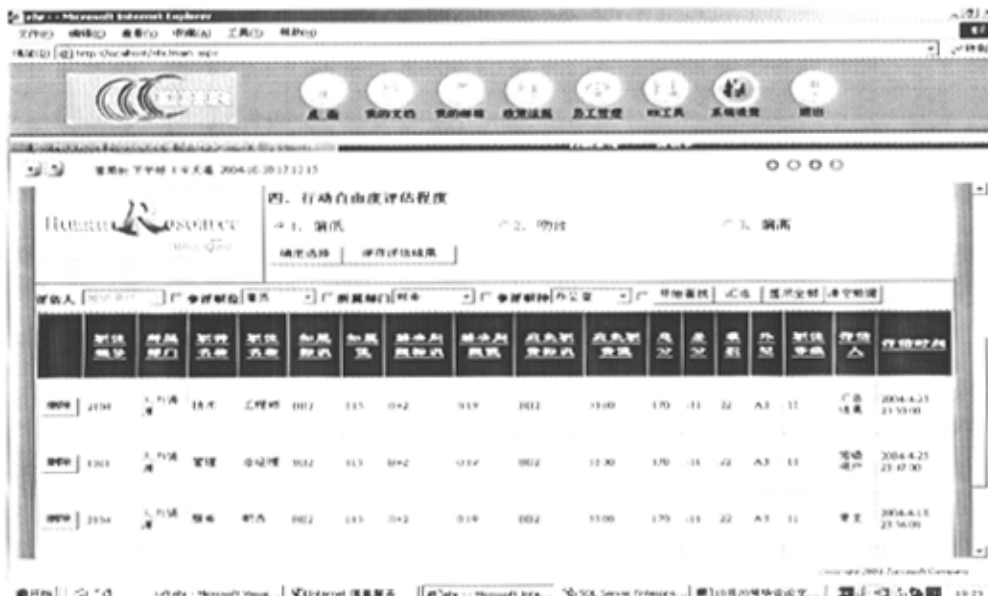


图 5-11 职位评估完成后隐藏了导航区和菜单区的评估结果界面

选择“素质测评”，进入第一步“确定行为模式及要素要项”，如图 5-12



图 5-12 素质评估第一步“确定行为模式及要素要项”界面

素质评估第一步“确定行为模式及要素要项”完成后显示结果，如图 5-13



图 5-13 素质评估第一步“确定行为模式及要素要项”完成

## 5.6 员工管理

图 5-14 为员工情况详细表，员工可以对自己的资料进行查看。

图 5-14 展示了员工情况详细表的界面。该界面包含以下信息：

工号	英文名字	性别
50000000000000000000	JACK	男
中文姓名	出生日期	国籍
江某	1975-05-20	中国
国籍	通信地址	身份证号码
中国	佛山市南海区南海大道	50000000000000000000
移动电话	住宅电话	E-MAIL
13922758857	0757-89536247	jack@public.nanhai.com
户口所在地	邮编	户口性质
广东广州	510332	非农
外来人员就业证号	居住证号	婚姻状况
00041234	1234567	已婚
曾用名	学历	专业
无	大学本科	计算机应用
毕业学校	华南理工大学	

图 5-14 员工情况详细表

员工同时还可以对自己的资料进行修改，但工号和中文姓名在用户创建时设置，普通用户没有权限进行修改，必须申请由系统管理员或相当于同等权利的用户才有权修改此两项，如图 5-15 所示。

图 5-15 展示了员工详细情况编辑界面。该界面与图 5-14 类似，但增加了以下功能：

- 在“工号”和“中文姓名”字段右侧，增加了“不可修改”的提示。
- 在“备注”字段下方，增加了“保存”、“确定”和“取消”按钮。

图 5-15 员工详细情况编辑界面

## 5.6 系统设置

包括用户管理、部门管理、职务管理、角色设置、权限设置；这些管理和设置有加、查询、修改和删除四个部分的操作，数据维护则有数据导入导出两部分。



图 5-16 用户管理之“添加用户”界面



图 5-17 组织机构管理之“添加机构”界

## 5.7 外部连接

主要将经常访问的网站的网址列出供选择。

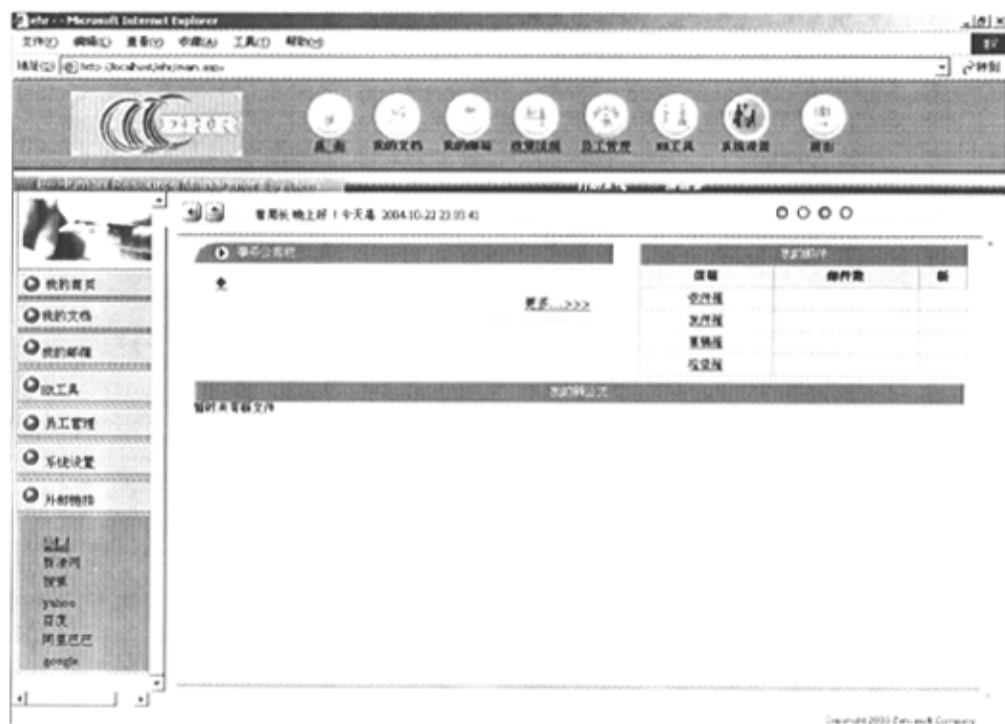


图 5-18 外部链接

## 第六章：总结与展望

### 6.1 总结

本论文分六个章节的内容介绍了基于.net 的人力资源管理系统（eHR）的设计和实现过程，介绍了 eHR 系统的产生、发展和现状；介绍了.net 技术背景、系统的体系结构、功能模块组成和部分功能模块的实现；介绍了 eHR 系统数据库设计、页面设计以及自定义控件的设计、着重介绍了基于 RBAC 角色访问控制的授权机制在系统中的应用及实现以及 Rijndael 加密技术在 eHR 系统中的应用和实现，同时论述了 eHR 系统在实际实施时需要部署的安全解决方案。

本设计的最大特点是采用微软的.net 技术，将 OA 的部分功能创造性地和由先进的 HR 管理理念构成的软件工具相互结合，自成一体，真正意义上实现了企业信息化的目标和人力资源的电子化。特别是 RBAC 用户授权机制的进一步扩展和完善，使整个系统的运作更显灵活和安全，作为一种授权机制，它也可以广泛地应用到其他信息管理系统中。

本文所研究的 eHR 系统中的职位评估子系统已经在南方航空股份有限公司得到了有效的应用，整个的评估工作取得了很大的成功。以强大的 HR 管理工具为依托的 eHR 系统的卓越功能和给企业带来的价值，将在 eHR 实际的使用中得到更充分的体现。

### 6.2 展望

eHR 发展趋势：HR+IT+咨询。

最近，据 IDC 统计（IDC 是全球著名的市场咨询和顾问机构，在全球 43 个国家设有分支机构，其中亚太区有 17 个国家），在全球的企业管理信息系统中，人力资源管理系统投资占 10%~15%，是其中最重要的部分之一。随着 Internet 技术的广泛应用，企业将在“电子商务”时代背景下竞争与发展。未来，稳定的、机械性的、重复性的工作，基本上被机器所取代，或者将被“外包”，最终只剩下脑力的、创造性的工作。信息技术与人力资源管理的结合将是人力资源管理 e 化的必然趋势，企业 eHR 的实施不是该不该上而是什么时候上的问题。

同时，无论是“本土”产品还是“洋”产品，软件产品的功能是决定其能否生存和发展的首要条件。信息技术和人力资源管理毕竟是两个不同领域，软件开发者既要理解人力资源管理，又要理解哪些层面的东西是可以信息化的，并找到两者和最佳契合处，这样的产品才具备不衰的生命力，因为没有一个是软件系统可以帮



助客户解决他所有的问题。一个优秀的 eHR 产品最关键的就在于如何找到 HR 管理的理念、运作体系与 IT 的结合点。

在技术上，软件开发方需要寻求合作，当它深入到企业内部管理时，亦可以与专业咨询公司合作，以寻求最佳的解决方案。因为，全面实施 eHR 系统要分几个阶段。第一个阶段是先解决非管理层面技术性的问题，如提高工作效率、改善流程。第二个层面上升到管理技术层面，如果企业自己做不了，最好由咨询公司做，如绩效管理中绩效指标的设立、对关键员工胜任素质模型的建立等等。

综上所述，企业 e 化的人力资源管理是一个综合问题，作为 HR 软件开发方需要集合各方智慧和擅长才能真正给企业带来适用的、卓有成效的解决方案。而 HR 管理理念和 IT 技术和咨询实践的结合，即 HR+IT+咨询模式，笔者以为是目前最科学的、最卓有成效、最专业的 eHR 解决方案，也是 eHR 软件系统未来的发展趋势。

## 参考文献

- [1] 陆惠恩、陆陪恩著. 软件工程. 电子工业出版社, 1997 年
- [2] 黄志超、李婷婷著. Visual Basic.Net 项目开发实践. 中国铁道出版社,
- [3] 陈峰棋著. 深入浅出 ASP.NET 程序设计. 中国水利水电出版社, 2004 年
- [4] H.N.DEITEL 著. 葛昊晗、汤涌淘、李强译. C#大学教程. 清华大学出版社, 2003 年
- [5] 王华杰、李律松、孙一波等著. 精通 C#数据库开发. 清华大学出版社, 2004 年
- [6] 方盈编著. Sql Server 7.0 从入门到精通. 中国铁道出版社, 1999 年
- [7] Kalen Delaney、Ron Soukup 著. SQL Server 7.0 技术内幕. 北京大学出版社, 1999 年
- [8] Geetanjali Arora、Balasubramaniam Aiaswamy、Nitin Pandey 著. 徐成敖、王雷等译. C#专业项目实例开发. 中国水利水电出版社, 2003 年
- [9] 陈南平、方亮、张立峰等著. RBAC 机制实现 WWW 环境中的安全访问控制. 华中科技大学学报自然版, 2002 年: 53-55
- [10] David Ferraiolo、Rick Kuhn National Institute of Standards and Technology  
Gathersburg、Maryland Ravi Sandhu George Mason University Fairfax、  
Virginia. Proposal for Fast-Tracking NIST Role-Based Access Control  
Standard, 1998 年
- [11] 彭剑峰、饶征著. 基于能力的人力资源管理. 中国人民大学出版社, 2003 年
- [12] 饶征、孙波著. 以 KPI 为核心的绩效管理. 中国人民大学出版社, 2003 年
- [13] 东名、Riki Lin 编著. Exchange 2000 Server 系统务实经典. 清华大学出版社, 2001 年
- [14] Alistair G.Lowe-Norris 著. 肖雪莲、邓晓蓓译. Windows 2000 活动目录. 中国电力出版社, 2001 年
- [15] 王晓亮、陶红著. 使用 ADSI 和 Exchange CDO 管理活动目录. 计算机应用, 2002 年: 91-93
- [16] David Iseminger 主编. Windows 2000 活动目录开发人员参考库 ADSI 程序员指南. 机械工业出版社, 2001 年
- [17] 祁明主编. 电子商务安全保密. 高等教育出版社, 2002 年
- [18] 张焕国、刘玉珍著. 高级加密标准 AES 算法——Rijndael 的设计. 武汉大

学出版社，2000 年

- [19] Richard Anderson、Brian Francis 等著. ASP.NET 1.0 高级编程. 清华大学出版社，2002 年

## 在学期间发表与学位论文内容相关的学术论文

序号	作者(全体作者,按顺序排列)	题 目	发表或投稿刊物名称、级别	发表的卷期、年月、页码	相当于学位论文的哪一部分(章、节)	被索引收录情况
1	曾联明	.基于人力资源管理系统的用户信息管理策略研究	.佛山科学技术学院学报	2005.6	第四章第五节	录用
2	曾联明	基于能力的人力资源管理信息系统及其授权体系的设计	.计算机工程与应用	2005.8	第四章第五节	录用
3	曾联明	用 C 语言链表解决大整数运算的精度问题	电脑学习	2002,3:31		发表
4	朱扬清、曾联明	.新一代网络考试系统计算平台的设计	微型机与应用	2003,9:41-43		发表
5						

## 致 谢

目前，eHR 的产品可谓名目繁多，但是否真正给企业带来革命性的变化，还需要实践来验证，不同的企业可以根据自己的需要选择适合自己的产品。鉴于笔者能力和学识的不足，在这方面的研究和实践还非常的浅薄，希望能得到各位专家、大师的批评指正。

在研究生就读期间和毕业设计研究中，首先我要特别感谢我的指导老师韩国强教授，他敏锐的思维、广博的学识和严谨的治学精神令我获益匪浅；他为人和蔼亲切、宽容、胸襟坦荡，是我处世的楷模，正是导师的指点和悉心教诲，使我不断克服困难，勇往直前。

我在攻读同等学历硕士期间，有幸得到华南理工大学计算机学院的各位老师无私的传授和教导，使我的见识和能力都得到了很大的提高，他们无私的给予成就了我克服困难、完成课题的研究和论文写作的知识基础，在此，对他们表示衷心的感谢。

谢谢在论文写作过程中给予我大力支持的计算机系叶小静老师、也同时谢谢我的同事钟勇博士的支持和帮助。

最后，感谢我的家人给予我的支持和鼓励。