國立交通大學

資訊科學與工程研究所

碩 士 論 文

交 通 大 學 校 園 資 訊 電 子 化 及 其 整 合

NCTU Administrative e-Office System and Integratoin

研 究 生：游名楑

指導教授：謝筱齡/蔡文能　教授

中 華 民 國　九 十 五　年 六 月

交通大學校園資訊電子化及其整合
NCTU Administrative e-Office System and Integration

研 究 生：游名楑　　Student：Ming-Chieh Yu

指導教授：謝筱齡　　Advisor：Sheau-Ling Hsieh

蔡文能　　Wen-Nung Tsai

國 立 交 通 大 學

資 訊 科 學 與 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

June 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年六月

NCTU Administrative e-Office System and Integration

student：Ming-Chieh Yu

Advisors：Dr. Sheau-Ling Hsieh
Dr. Wen-Nung Tsai

Department（Institute）of Computer Science and Engineering
National Chiao Tung University

ABSTRACT

NCTU e-Office System has been under developing for years. Almost every division on campus, i.e., accounting, bursar, admission, registrar, personnel, library, etc, has established its own e-Document system. In other words, currently, each administrative division has its own electronic data management cluster (may not be implemented using database). Obviously, there is lacking of synchronization or unification among these systems. For administrative related issues, employees or students need to stop by individual division one-by-one in order to complete the process.

In this thesis, we introduce several divisions in NCTU campus. Then we discuss the problems discovered when we visit those divisions. Several integration methodologies will be proposed in this article, such as front-end integration, common database integration and interface integration. At least we design an system architecture to solve the mentioned problem.

Keywords: Campus information, Front-end Integration, Common database Integration, Interface Integration

交通大學校園資訊電子化及其整合

學生：游名楑　　指導教授：謝筱齡
蔡文能

國立交通大學資訊工程學系（研究所）碩士班

摘　　要

交通大學電子公務系統已經發展多年，幾乎在校園內每個部門，出納組、會計室、註冊組、人事室、圖書館等，都已經有他們自己的電子文書系統。換句話說，每個管理部門都有他們自己的電子資料管理群組(不一定是使用資料庫)。明顯地，在這些電子文書系統中缺乏同步和一致性。對於管理相關的議題，教職員和學生必須相當麻煩地依續到和程序相關的各個部門辦理手續。因此，必須要有一個行政電子公務系統來將這些電子文書系統整合在一起。

本篇論文一開始將對校園內數個部門做簡單的介紹，然後討論在我們尋訪這些部門所發現的問題。數個整合校園資訊的方法將被介紹，如共同資料庫整合、介面整合及前端整合。最後我們利用這些整合方法提出一個整合目前校園資訊的架構，以解決校園中所發現的問題。

關鍵字：校園資訊, 共同資料庫, 介面整合, 前端整合

誌　　謝

研究所二年，首先必需要感謝謝筱齡與蔡文能老師的指導，在研究所其間不僅傾囊教導課業上的知識，更導正我們正確的做事態度，對於學生平常生活也相當關心，二年的成長，有很多必須歸功於謝筱齡老師辛勤的付出與教導，亦師亦友對待實驗室中每個成員，以自己的言行態度薰陶每一位學生，認真努力的態度為我們樹立了學習的典範。

還有必需感謝實驗室的伙伴們宏文、東泰、嘉崑及薰哥，幾乎全年無休在實驗室裡打拼，修課時互相研究，彼此討論論文，也一起爆了不少的肝，懷念大家一起修課熬夜的日子，雖然辛苦，但也一起學習了不少東西。

最後也是最重要的要謝謝我的父母家人，辛苦撫養長大，對我平日的關心與精神上的支持，並給我相當大的自我發展空間，讓我無憂完成碩士學位。未來的路還很長，但我會帶著這二年所學與美好的回憶繼續努力下去的。

Table of Contents

List of Figures

CH 1 Introduction

1.1. Preface

NCTU e-Office System has been under developing for years. Almost every division on campus, i.e., accounting, bursar, admission, registrar, personnel, library, etc, has established its own e-Document system. In other words, currently, each administrative division has its own electronic data management cluster (may not be implemented using database). Obviously, there is lacking of synchronization or unification among these systems. For administrative related issues, employees or students need to stop by individual division one-by-one in order to complete the process.

Therefore, an Administrative e-Office System is desired to “glue” these clusters together. The ultimate goal: employees and students can retrieve all their relevant status from the System at one location. The System will be physically resided and maintained by Computer & Network Center.

The System will be designed based on “integrated” middleware technologies. The possible technologies involved can include: front-end integration, back-end integration, wrapping techniques plus integrated communication protocols. The key point is to remain and maintain every individual system intact.

Initially, a complete survey of every current system on campus is essential, i.e., acquiring, analyzing individual division’s features, requirements as well as functional dependencies possibly tangled. A clear interface between the System and the individual cluster will be defined. The interfaces among divisions may be different, depending upon the requirements. Finally, the System will be established, integrated covering heterogeneous division clusters. In addition, the System should address authentication, authorization security issues as well as transaction management, logs while handling paychecks or students’ grades.

1.2. Organization

In Chapter 2, we introduce several divisions and their features, also the e-office systems in the divisions are also introduced. In Chapter 3, we discuss some responses of members working in each division when visiting them. The following chapter consists of the integrated e-Office architecture and design methodologies. Chapter 5, we give the conclusion and future work.

CH 2 Background & Systems Analysis

2.1. Registrar Division

2.1.1. Background

The affairs of Registrar division are to maintain the student status, record the academic result, academic certificate application service and preserve the student status in campus. They provide student ID card application for students. Provide services of degree certificate and certificate printing for graduate students. Registrar division provides the students data for divisions in campus. If there is student status modification, a document will be delivered to relative divisions to inform the modification of student status.

2.1.2. Operations and Features

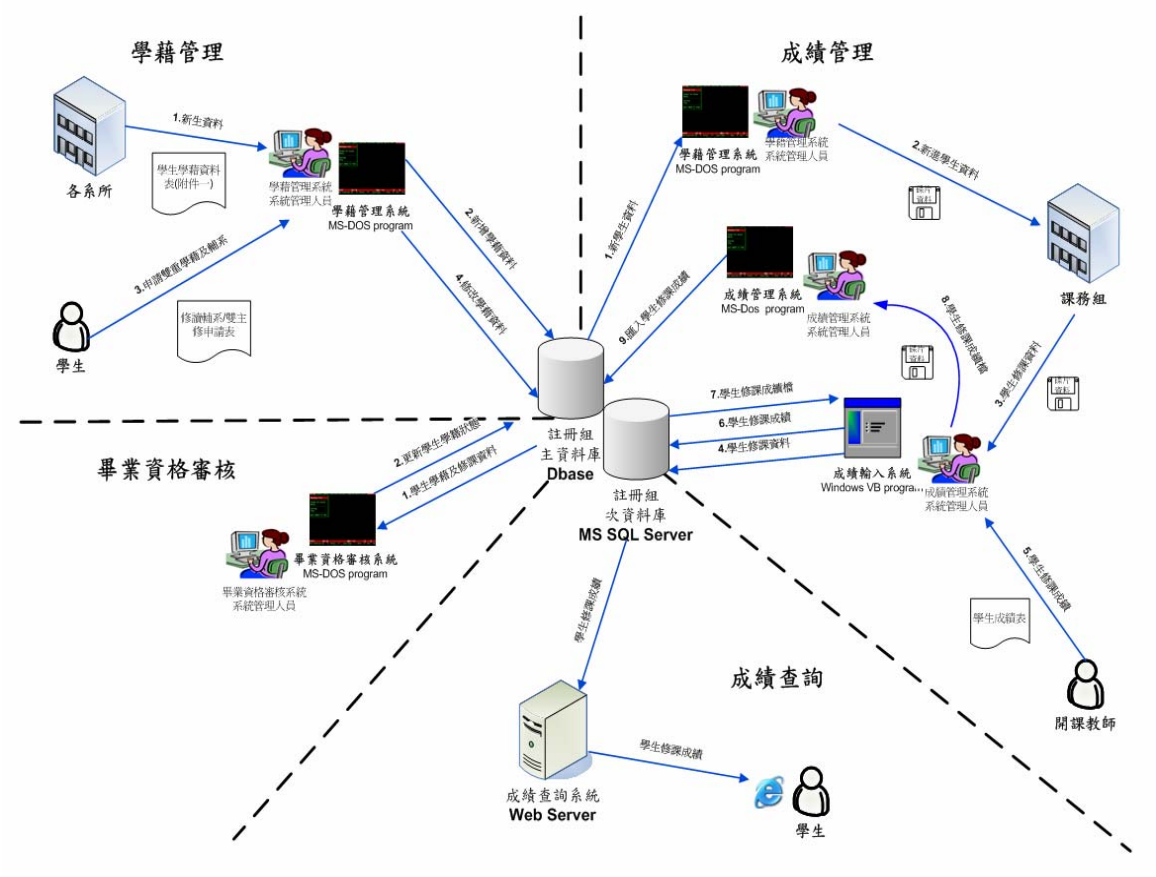


Figure 2-1 Procedures of the affairs in Registrar

Student Status Management Procedure

1. After finishing admission program，new students fill up their student status information forms. The student status information forms are delivered to the Registrar Division by the assistants of individual department.
2. The managers of student status management key in the information of the delivered student status information forms by operating the Student Status Management System.
3. Student applies for a second major or a minor study.
4. After the student's permission of applying for a second major or a minor study, the manager of student status management modifies student status of the student.

Academic Result Management Procedure

1. After all the student status information is built, the student status manager create new student data file by exporting the new students' data in the Student Status Management System.
2. The new student data file is delivered to the Curriculum division for course selection program.
3. After the course selection program, the course selection information of students is delivered to the academic result manager by the curriculum division.
4. The academic result manager import course selection data of students into the Registrar Minor Database by using the Academic Result Management System.
5. After termination of courses, teachers hand out academic result forms.
6. Academic result key-in employees key in the academic results by using Academic Result Key-in System.
7. After all the academic results typing work, academic result manager export the academic result of students by using Academic Result Key-in System.
8. The exported academic result file is delivered to the academic result manager.
9. The file is imported into the Registrar Major Database by using Academic Result Management System.

Degree Certificate Procedure

1. Degree Certificate System checks the earned credits of students to determine weather the students are qualified for graduation.
2. If students are qualified for graduation, the student status manager modified the student status of the students.

Academic Result Enquires

Students can login the Academic Result Enquiries System to enquire about their academic results.

2.1.3. e-Office Systems Environment

(1) e-Office Systems

The current e-Office systems and their environment is listed the following table.

System Name	Environment	System Functions	Developing Tool
Student Status Management System	OS :MS-DOS	Manage the student status in NCTU.	Dbase
Academic Result Management System	OS :MS-DOS	Manage the academic result of the students.	Dbase
Degree Certificate System	OS :MS-DOS	Degree Certificate for students who are going to graduate.	Dbase
Academic Result Enquire System	OS :Window 2000 Server Web Server: MS IIS 5.1	On-line academic result enquires for students.	Dbase
Academic Result Key-in System	OS :Windows XP	Record the academic result of students.	Visual Basic 6.0

(2) Database Environment

Because the windows edition e-Office systems of Registrar division are still under developing and only one windows environment e-Office systems, there are two databases in the division. One major DBMS which is Dbase, the other is minor DBMS which is MS SQL Server 2000. The databases in the division consist of student status information and the academic results. The student status data is composed of student id, Chinese name, English name, class, national ID, birthday, telephone number, student type and so on. The academic results include course id, course name, credit, grade, course type and academic year.

2.2. Curriculum Division

2.2.1. Background

Curriculum division arranges the course selection program and manages the classroom leasing application. Before the beginning of the semester, it is necessary to collect the course plan of each department. After the agreement of the course committee, Curriculum division charges the course registration of students. The credit fee of each student is accounted and sent to Bursar. The course registration data of students is delivered to Registrar division. The classroom leasing management begins after the course selection program and on-line classroom leasing application is provided for classroom leasing.

2.2.2. Operations and Features

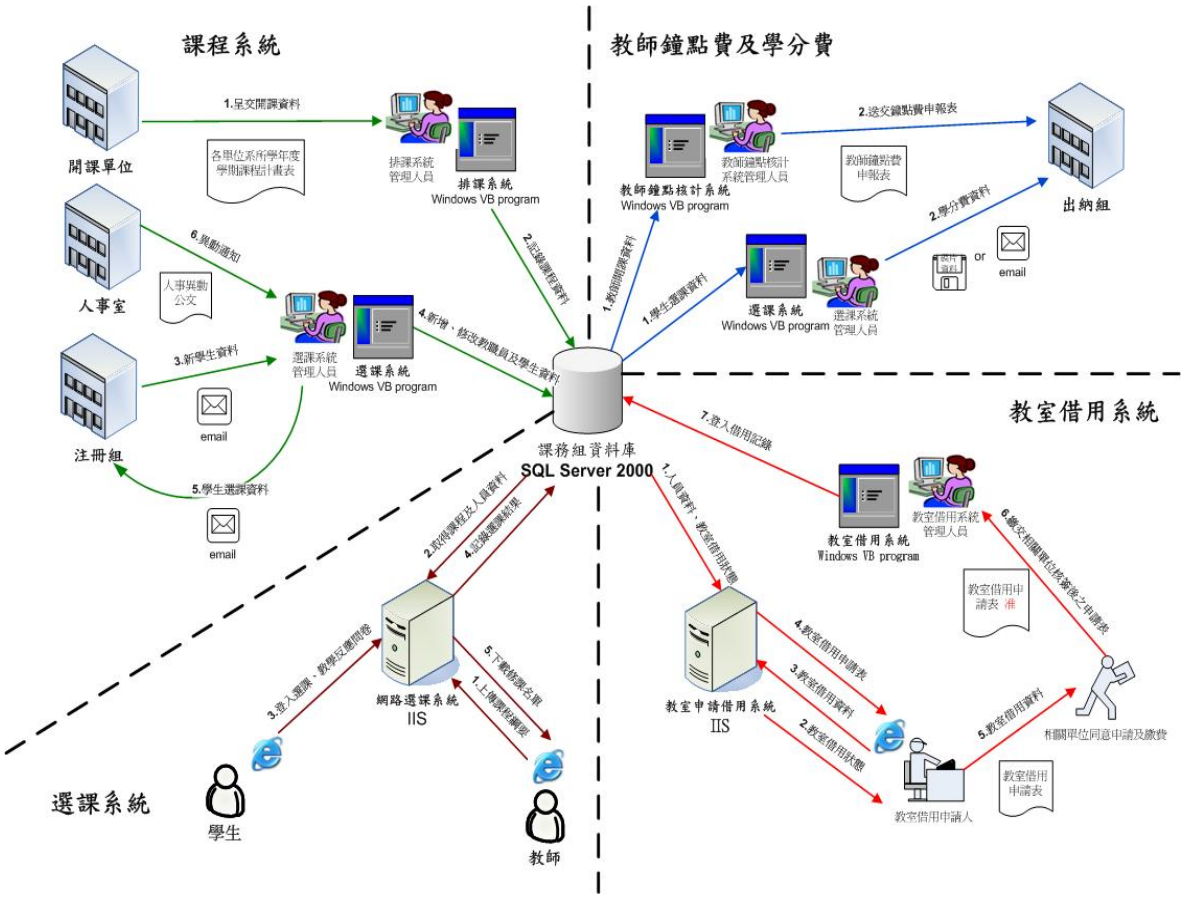


Figure 2-2 Procedures of the affairs in Curriculum Division

Course System Procedures

1. Each department proposes its course program in next semester. Courses information is input into Course scheduling System. Each course is assigned course id.
2. At the beginning of the semester, new student data is sent to Curriculum division and is imported into Course Selection System.
3. After course selection program, the course registration information is sent back to the Registrar division.
4. Curriculum division is notified of the promotion or resignation of someone then modified the personnel information in Course Selection System.

Course Selection System Procedures

1. Teachers upload course syllabus to the On-line Course Selection System before the beginning of on-line course selection program.
2. Students login On-line Course Selection System and fill out the form of questionnaire on response to teaching, then they start their course registration.
3. On-line Course Selection System collects the course registration information of students.
4. The course registration data is written back to the database.

Teaching Hours Accounting System Procedures

1. Teaching Hours Accounting System retrieves teachers' personnel and course data from database.
2. After teaching hours accounting work, the teaching hours notification form is delivered to Bursar.

Course Credit Charge Procedures

1. Course Management System retrieves the course registration data of students from database and accounts credits fee of students..
2. The Course Selection System manager exports credits fee data file of students and deliver it to Bursar in form of an email or a disk file.

Classroom Leasing System

1. Classroom Leasing Application System retrieves the applier and classroom leasing status information form database.
2. The Applier browses classroom leasing status by web browser.
3. The Applier input the classroom leasing information and reserves the classroom.
4. The system sends the classroom leasing form then the applier prints out the application form.
5. The applier goes through the listed departments or divisions in the form to apply for the agreement of the classroom leasing. After that, the applier goes to Bursar to pay the payment of the classroom leasing.
6. After all the required steps are accomplished, the application form is sent back to Curriculum division.
7. The classroom leasing manager check the application form and records the classroom information by using Classroom Leasing Management System.

2.2.3. e-Office Environment

2.2.3.1 e-Office Systems

The current e-Office systems in Curriculum division are listed in the following table including the system name, software environment, system functions and the developing tools.

System Name	Environment	System Functions	Developing tools
Course Scheduling System (CSS)	Windows XP Operating System	Assign temporary course number and persistent course number to courses.	Visual Basic 6.0
Course Selection Management system(CSM)	Windows XP Operating System	Course selection data management and credit fee accounting.	Visual Basic 6.0
on-line Course Selection system(CS)	Windows 2000 Server and IIS 5.1 Web Server	Provide on-line course selection for students, course syllabus uploading and roster downloading for teachers. It also provides patch email function for teachers.	Microsoft Active Server Page
Teaching Hours Accounting System(THAS)	Windows XP Operating System	Account the teaching hours of teachers.	Visual Basic 6.0
Classroom Leasing Management system(CLM)	Windows XP Operating System	Classroom recording and Manage the usage of classroom and classroom leasing management	Visual Basic 6.0
Classroom Leasing Application(CLA)	Windows 2000 Server Operating System and IIS 5.1 Web Server	Classroom free sections browsing and application of Classroom leasing form filling up for appliers.	Microsoft Active Server Page

Architecture, Protocol Stacks & Components

The e-Office systems in Curriculum division are aimed for facilitating course selection and classroom leasing program. There are management program for inside work and the web applications for courses enquiries and personnel data management. Management programs, web server for teachers and students access and database are the main three components in Curriculum division. There are four management programs,(1) Course Scheduling System, (2) Course Selection System, (3) Classroom Leasing Management System and (4) Teaching Hours Accounting System. The two web servers are primary components. One is provided for course selection of students and teachers. The other is for the applier of the classroom leasing.

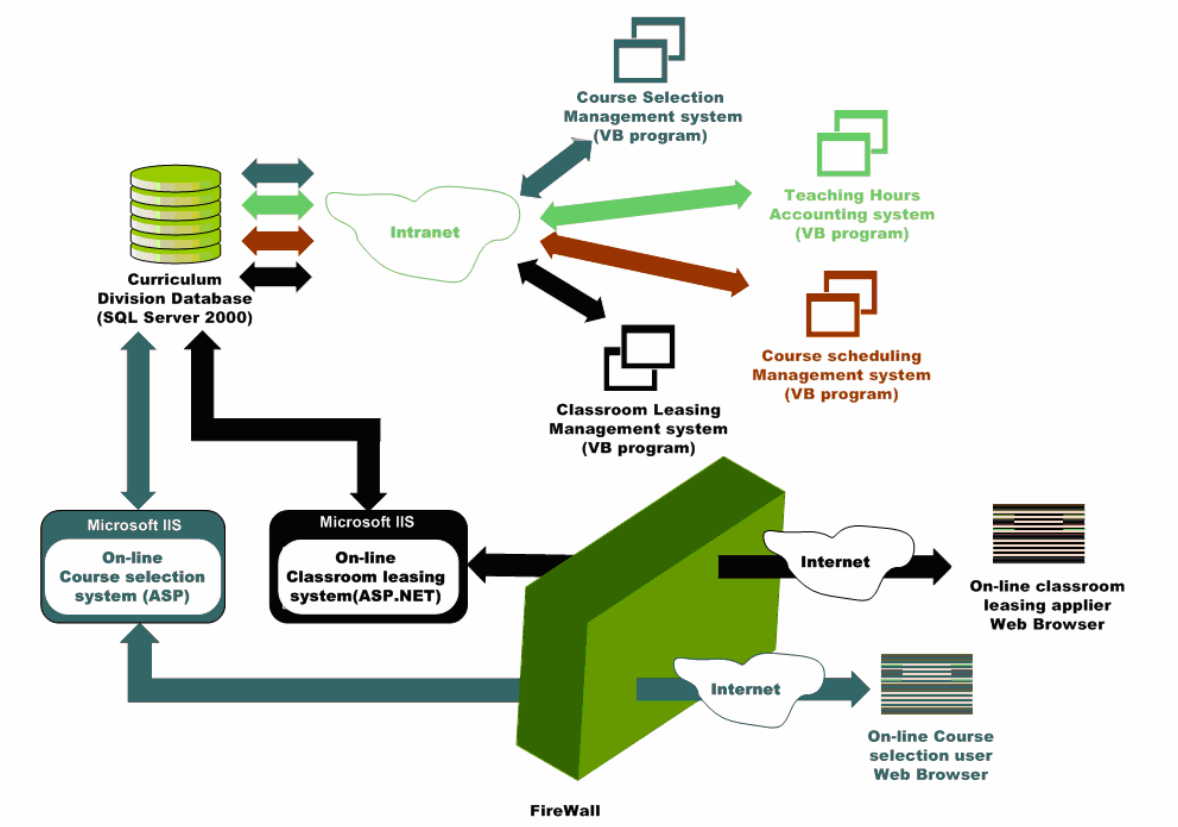


Figure 2-3 e-Office Systems Architecture and Components in Curriculum Division

The two web servers provide teachers and students for files uploading and downloading by using HTTP protocol. Course Selection Server and Course Leasing Application Server connect to the database based on ODBC database connectivity. The ODBC database connectivity is also used by Course Selection Management system (CSM), Classroom Leasing Management system (CLM), Course Scheduling System and Teaching Hours Accounting System. The communication protocols used by e-Office system in Curriculum division are (1) ODBC database connectivity and (2) HTTP protocol.

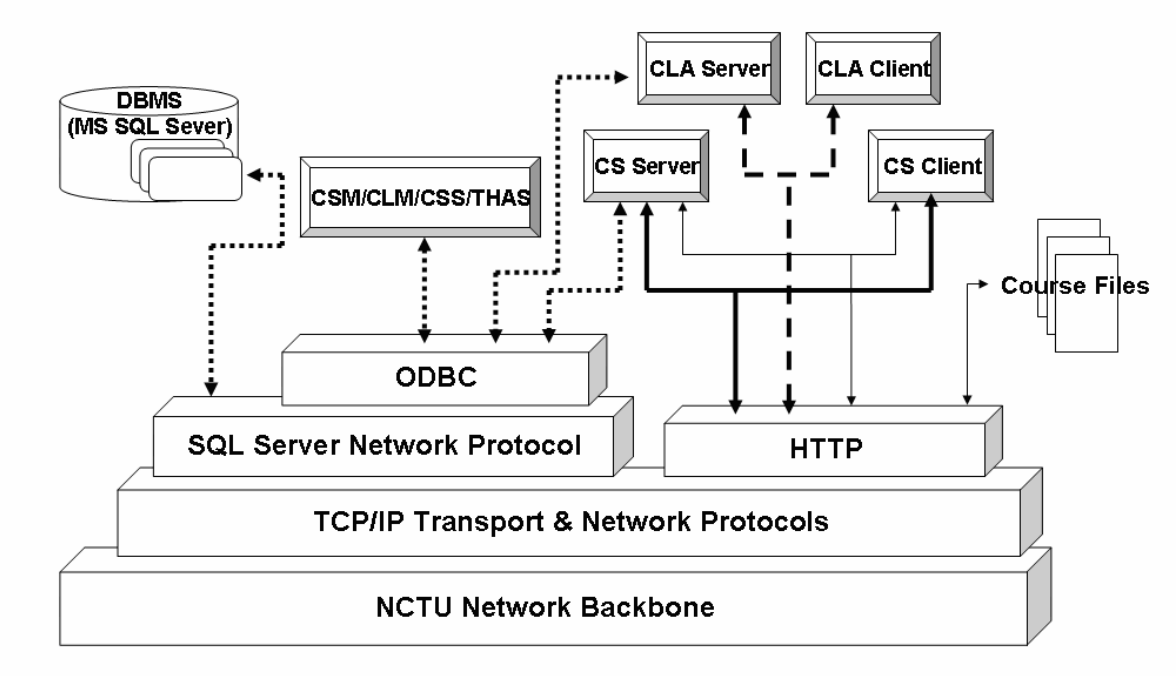


Figure 2-4 Curriculum Division e-Office System Protocol Stack Layers

3.2.3.2 e-Office Database

The DBMS in Curriculum Division is MS SQL Server 2000. The database in the curriculum division includes student data, course data and classroom data. After the course selection program, the course registration data of students is delivered to Registrar for academic result management. Student data consists of student id, student name, class and their password. Course Chinese name, persistent course number, course credit, course type, lecturer, department, classroom, class time, registered number, size limit and course hours. The classroom data includes classroom code, floor type, site number, privilege department, classroom level and so on.

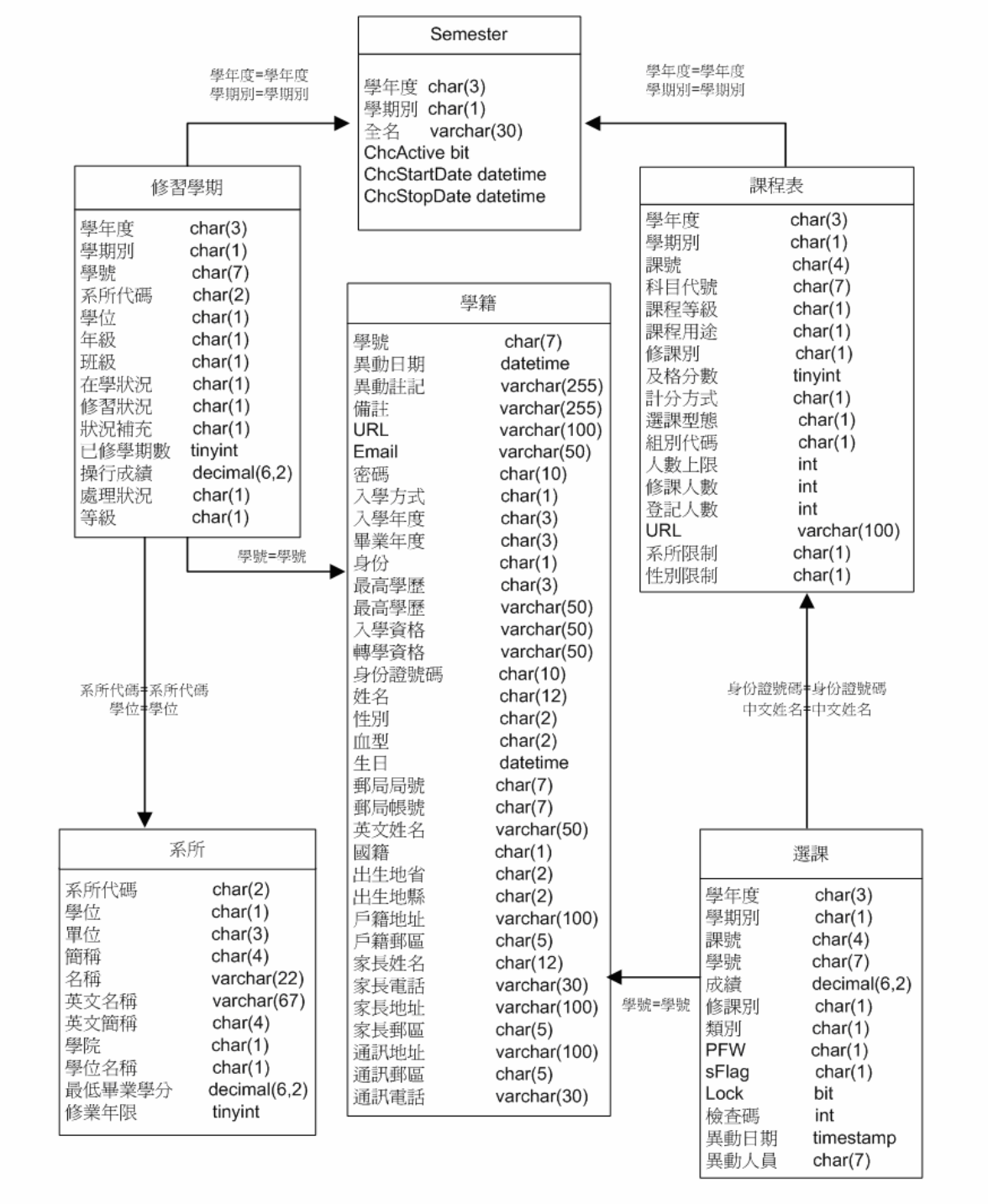


Figure 2-5 On-line Course Selection Database Schema

2.3. Bursar

2.3.1. Background

Bursar is responsible for Handling payroll and incidentals, receiving and paying checks, cash, remittance bills, certificate of deposits and secured bonds, initialing, registering and verifying checks according to the vouchers form accountant's office, collecting and handling tuitions and miscellaneous fees. It also offers salary, income tax fee and tuition fee enquires web pages for students and staff in campus. Fifteen every month is the payday, an email notification of paying salary service is provided.

2.3.3. E-Office Systems

The current e-Office systems in Bursar are as the following table which includes the software environment, system functions and their developing tools.

System names	Software environment	System functions	Developing tools
Revenues System	Operation System Windows XP	Record the receiving checks, cash and print the list of deposit of the day.	Visual Basic 6.0
Disbursement System	Operating System Windows XP	Record the numbers of vouchers and the corresponding check numbers. Print out the checks and list of telegraph from accountant's office.	Visual Basic 6.0
Tuition Fee accounting	Operating System Windows XP	Account and check tuition fee of students.	Visual Basic 6.0
On-line Income Tax Fee Enquiries	Operating System Windows 2000 Server Web Server Microsoft-IIS/5.0	Provide income fee enquires for staff and students in campus.	Microsoft Active Server Page
On-line Salary Enquires	Web Server Apache-Coyote/1.1	Provide salary enquires for staff and students in campus.	Java Sever Page

2.3.4. Database

The DBMS of Bursar is MS SQL Server 2000. The database consists of records of checks, receiving cash, vouchers with the corresponding checks' numbers, personnel information of people in the payment list. Currently, Bursar independently maintains personnel information in campus. The personnel data in Bursar is most comprehensive for the financial critical reason. Bursar also is notified of promotion or resignation of employees.

2.4. Campus Security Division

2.4.1. Background

The main duty of Campus Security division is to secure the entrance in campus, control the traffic and report the illegal parking vehicle. Vehicle Management System (VMS) is the e-Office system in the Campus Security Division. VMS is used to help control the legal vehicle in campus and provide students and staff to apply for a parking permit on-line. The e-Offices related procedure of affairs in the division is parking permit application.

2.4.2. Operations and Features

Procedure to Apply for a parking Permit

1.The applier uses web browser to connect to Parking Permit Application System and filled up required information in the web page, then print out the application form produced by the system.

2.The applier hand out the application form to the Campus Security Division. The administration assistant in the division verifies the ID card, driver license card and vehicle license of the applier and then inspect that whether the tickets of the applier are already paid by using Vehicle Management Manager (VMM).

2.5 If there are unpaid tickets, the applier must go to the bursar to pay for the tickets and get the receipt. Bring the receipt back to campus security division, the administration assistant checks the amount of money of the receipt and cancels the unpaid tickets' records by using VMM.

3.After the administration assistant verifies the identification of the applier, the applier goes to the bursar to pay the parking permit fee and get the receipt.

4.The applier takes the receipt back to campus security division. The administration assistant verifies the receipt and records the accepting information by using VM. At last, a parking permit is given to the applier.

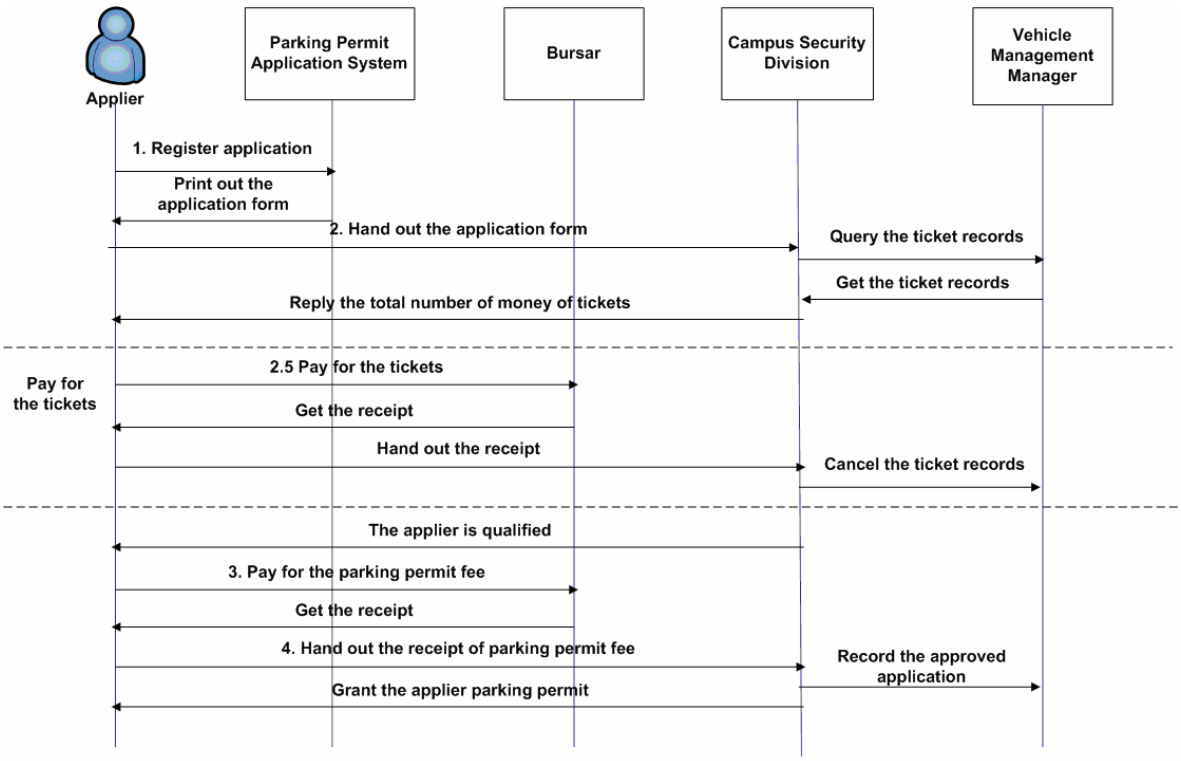


Figure 2-6 Procedure of Applying for a Parking Permit

2.4.3. e-Office Environment

(1) e-Office Systems

Vehicle Management System

Vehicle Management System (VMS) is developed by school administration information division of NCTU Computer Center. It aims on facilitating vehicles management in campus. It offers the parking permit applier to register his/her information and it also provides the manager to process the application of parking permit, tickets management, and information maintenance. VMS provides services for two kinds of users; one is parking permit applier, the other is vehicle manager. The parking permit applier uses web browser to login and registers information to the system. The manager process the parking card permit request, manages the tickets and maintains related information by using vehicle management VB program that connects to DBMS by ODBC. The parking permit applier should obey the announced procedure to apply her/his parking permit.

(2) Architecture, Protocol Stacks & Components

The VMS consists of applier part, manager part and database. The system has to provide various appliers coming from various platforms, the Web Client/Server architecture is used. For managers, a fast development and graphical VB program with ODBC database connection is used. VMS database offers data access for both appliers and managers. The framework or the protocol stack used in VMS is shown in the following figure. The framework consists of (1) ODBC database connectivity, (2) HTTP Server/Client Connection.

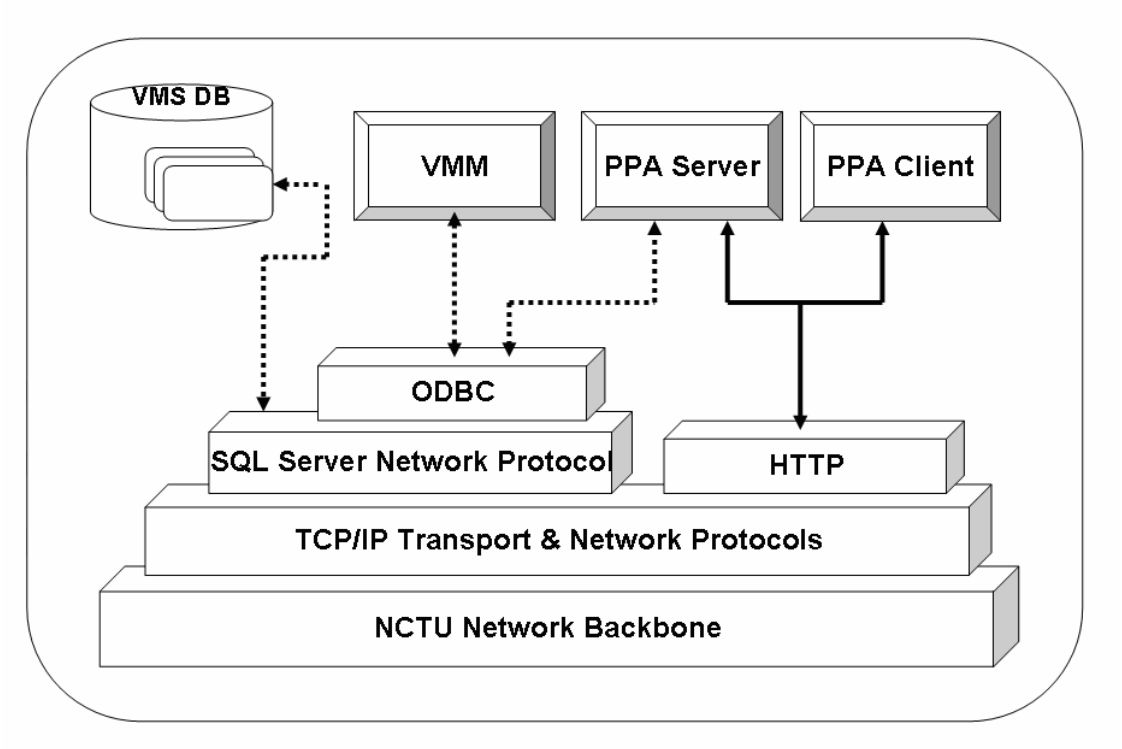


Figure 2-7 VMS Protocol Stack Layers

Figure 2-8 show as below illustrates Vehicle Management System architecture and components. The VMS can be divided into two parts, Parking Permit Application (PPA) which is an Web application written in ASP and Vehicle Management Manager (VMM) which is an management program written in VB. The DBMS is the MS SQL Server 2000 database.

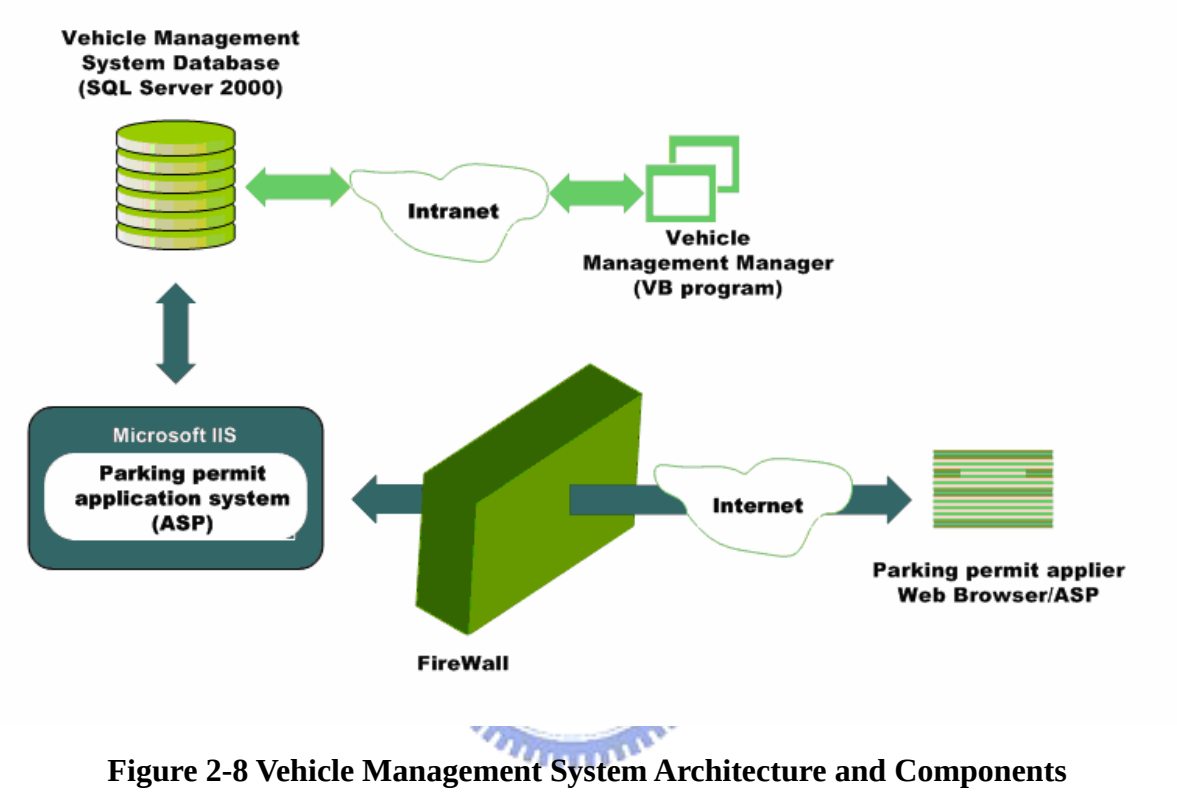


Figure 2-8 Vehicle Management System Architecture and Components

A. Parking Permit Application

Parking Permit Application is an ASP Web application that runs on Internet Information Server (IIS). The IIS is running on Windows 2000 Server operating system. It offer five kinds of people to apply the parking permit as the main menu presenting. The parking permit application page is show as figure 2-9 and its interactive diagram is shown in figure 2-10.

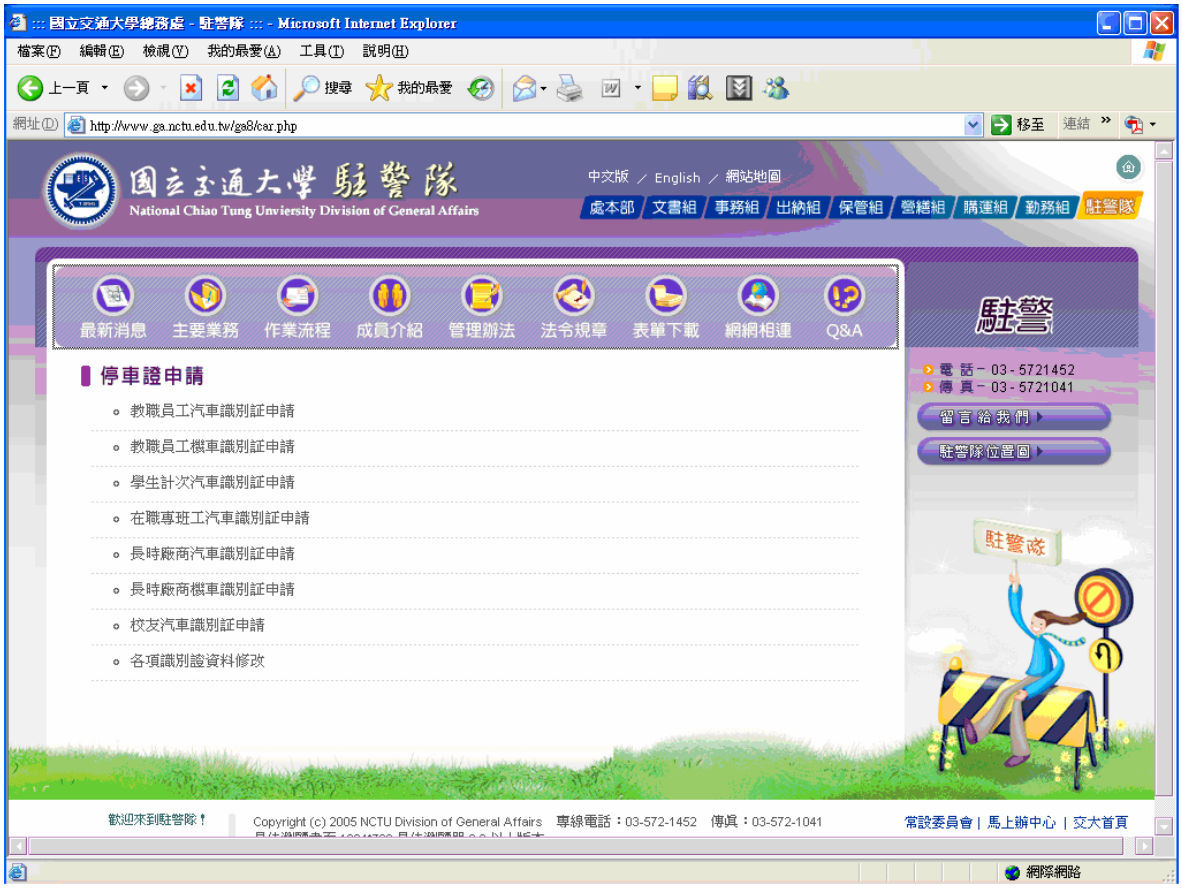


Figure 2-9 Application of Parking Permit Main Manu

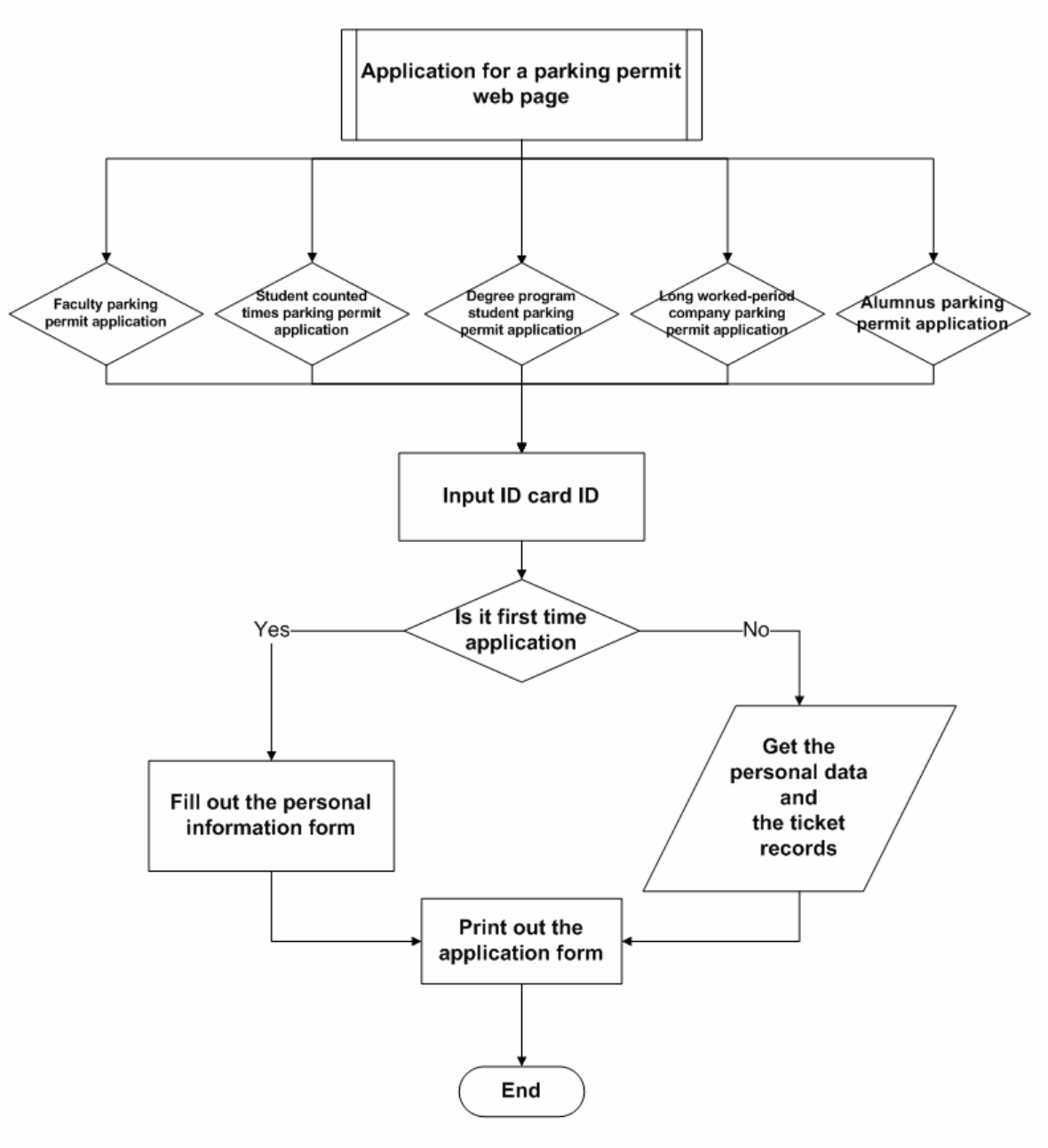


Figure 2-10 Application of Parking Permit Interactive Diagram

B. Vehicle Management Manager

VMM is a VB program running on the manager's personal computer. Figure 2-11 shows VM main menu. Before running VMM program, we have to set up the ODBC connections to VMS database. The Parking Card applier uses web browser to connect Parking Card Application Server and registers the required information.

Figure 2-11 Vehicle Management Main Menu

2.4.4. Database

The VMS database manages appliers' information records, parking permit record, parking tickets records and the records of campus police who gives the parking tickets and so on. The information of the staff and students in the database is managed by the campus security division themselves. The appliers' information is identified by the administration assistant. The administration assistant inspect the appliers' ID card or students' ID card and driver license card to see weather the appliers are qualified when the applier applies the parking permit at the first time. There would be a data inconsistent problem when a staff member leaves his/her job. The application of parking permit would be approved. This may cause a security hole in campus. Figure 2-12 shows the VMS database ER diagram.

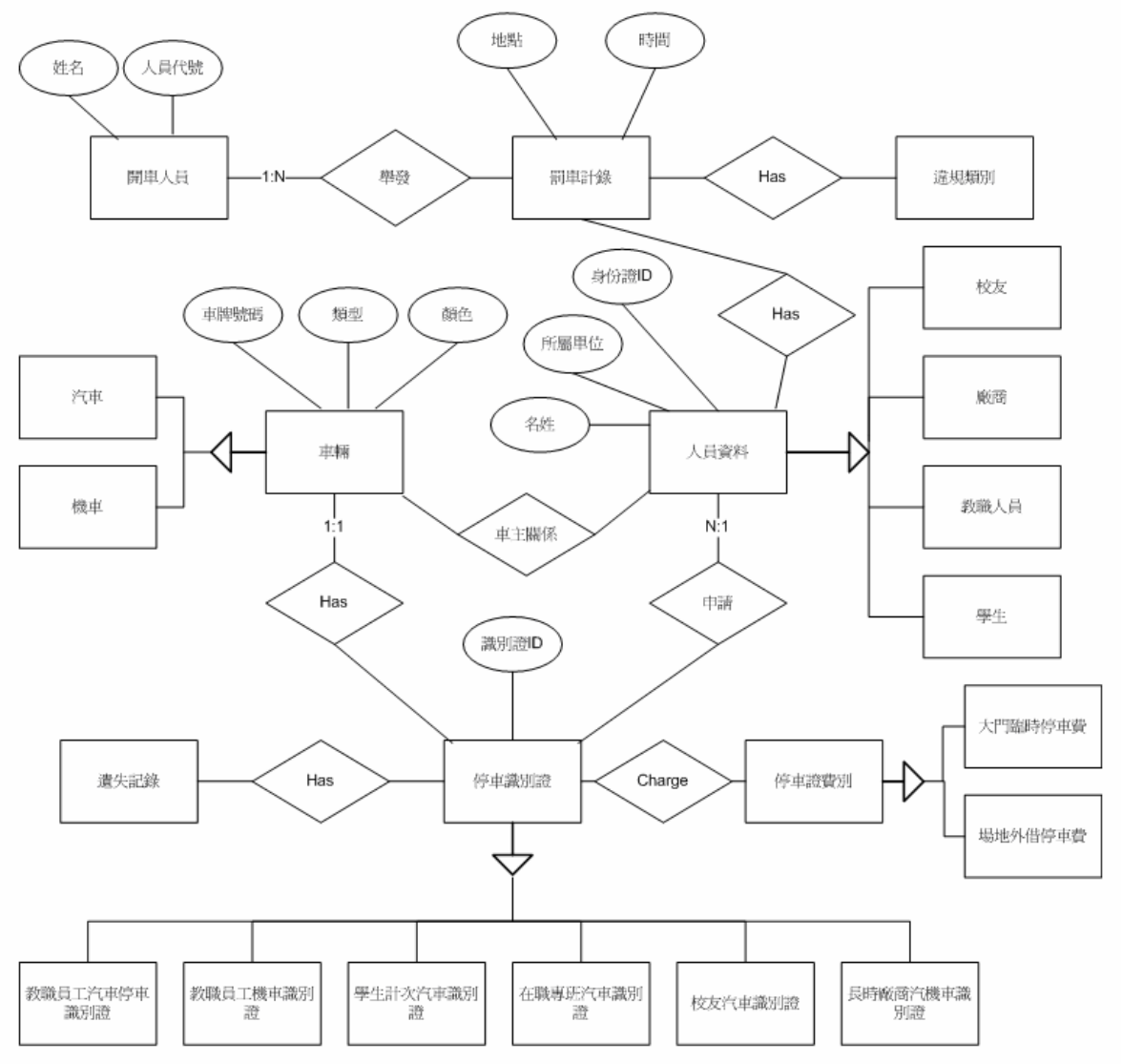


Figure 2-12 VMS Database Entity-Relationship Diagram

2.5. Document and File Division

2.5.1. Background

File and Document Division is responsible for receiving, managing and copying the official documents. They also deal with letters mailed to NCTU and official documents of departments in campus. All the mails send to NCTU are collected and distributed to other departments and dormitories by Document and File Division.

2.5.2. Operations and Features

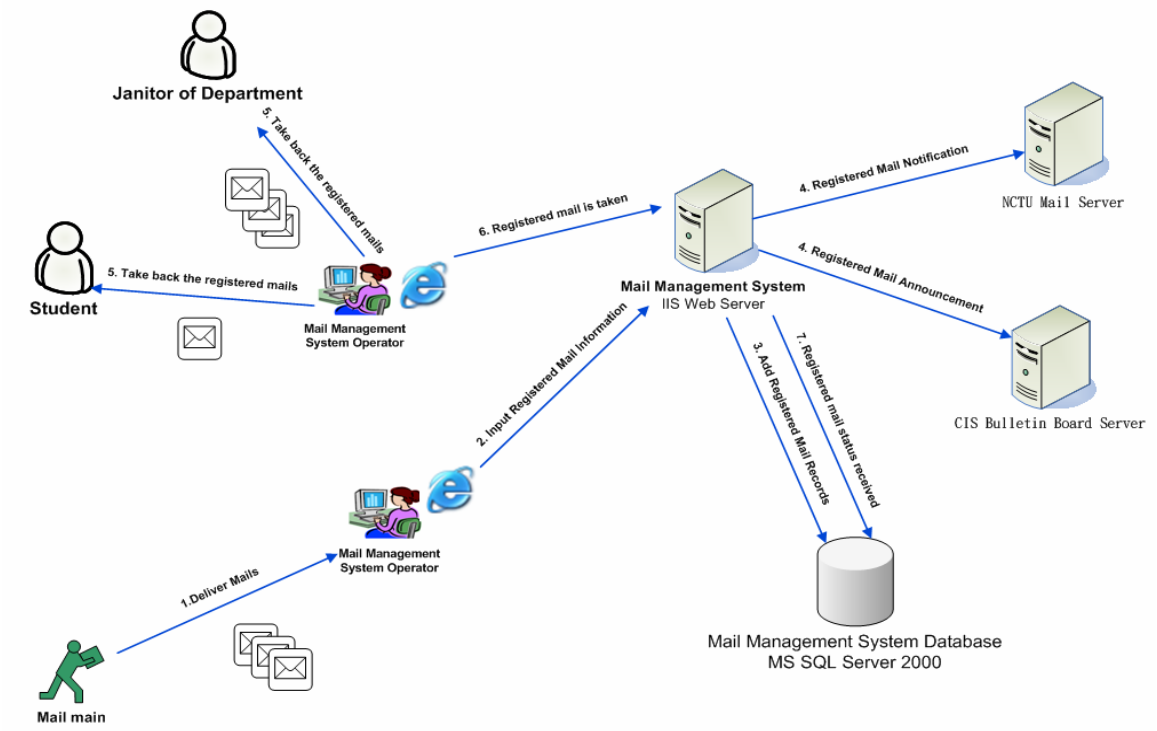


Figure 2-13 Procedure of Mail Management

Mails management

1. Mail man brings mails sent to NCTU to Document and File Division.
2. Mail Management System (MMS) operator input information of registered mails into MMS.
3. MMS add the registered mails records into the database.
4. MMS posts registered mails announcement up CIS BBS and emails notification mail to the registered mails receivers.
5. Students and janitors of each department take back their registered mails.
6. MMS operator inputs the taker information of registered mails into MMS.
7. MMS modifies the registered mails taken.

2.5.3. E-Office systems

Mail Management System

Mail Management System is used for facilitating registered mails processing. The employees in Document and File division input registered mails information into MMS everyday. MMS operator uses the email function to notify the mail receivers of their mails are delivered to the division. MMS is an ASP web application that uses MS SQL Server as their database management system. Windows 2000 Server is the using operating system. IIS 5.1 is the running web server. The functions of MMS are as followings:

1. Add and modify registered mails records.
2. Email notification of registered mails.
3. Post registered mails announcement up BBS.
4. Staff and students data management.
5. Registered mails reporting and printing.

Personal English Name Booking

There are many English mails in NCTU every day. The lack of student and staff's English names leads it difficult to process English mails in the division. For this reason, Document and File division develop a personal English name booking web site to provide students and staff in campus to enter their English name. English mails will be delivered more smoothly by referencing the English name students or staff entered. Also, Personal English Name Booking function is built on the same environment with MMS.

2.5.4. Database

The data used in MMS database comes from many divisions in campus. There are two data sources of students' data. One is register division, the other is dormitories. All the students' data comes from register division. We also need the name lists of students in dormitories to decide the destination to deliver the students' mails. The staff data is from common database in campus. The common database administrator opens a view table for staff data accessing for MMS. Also, the dormitories open a view table containing the paired student id and dormitory list of students live in dormitories. It is required to update database data every short period of time. The students' data includes student id, name, English name, email, dormitory, national ID number, department. The staff data contains staff id, name, email, national ID and department. The mail record includes the receiver name, department, receiving date, taken date, taker name, mail number, mail type, mailed location.

2.6. Personnel Common Database

2.6.1. Background

Each division in the campus has already developed their e-Office systems that require personnel data. The objective of common database is to provide the personnel data for each division as a personnel data sources. The procedure for applying personnel data with management policy have already constituted. Each division applies for the personnel data according to the procedure then the common database manager give the user/password to access the view table which contain the applied personnel data in the common database.

2.6.2. Software Environment

Operating Systems: Windows 2000
DBMS: MS SQL Server 2000

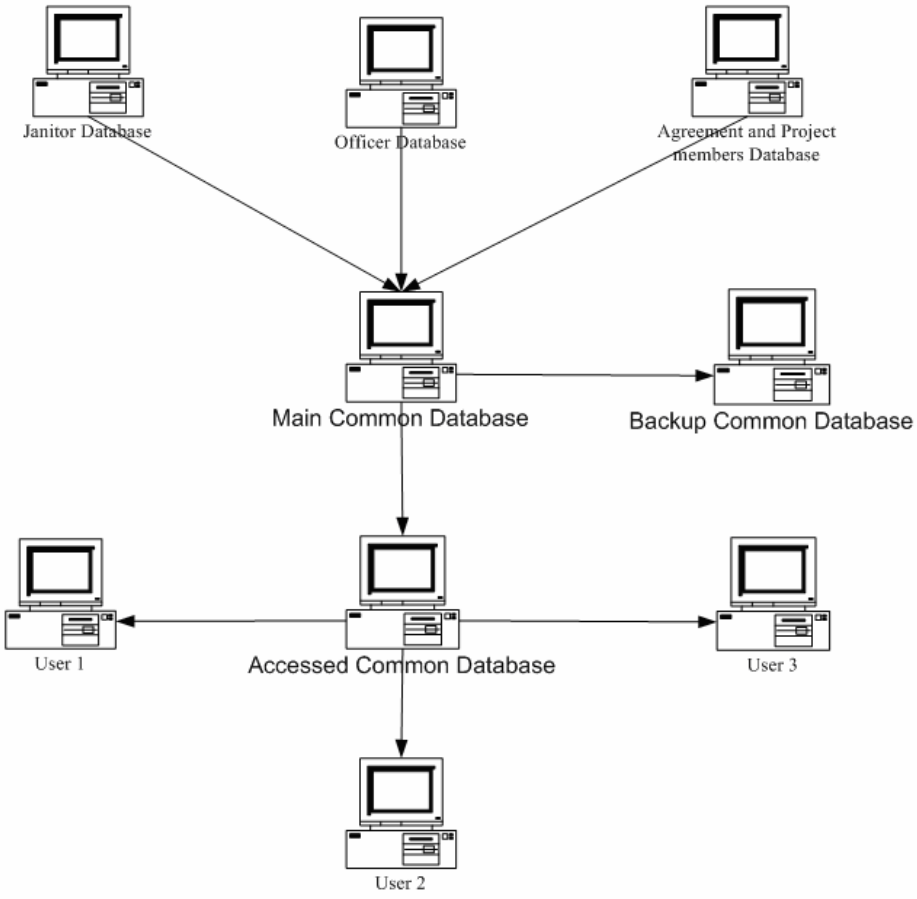


Figure 2-14 Common Database Data Source

2.6.3. Data Source

There are three kinds of employees in the campus: (1) Officer (2) Agreement and project member (3) Janitor. The officer data is gathered for the Officer Personnel Information Management System in the Personnel division. Officer Personnel Information Management System is developed by Central Personnel Administration. Agreement and project members' data is gathered from Agreement and Project Members Personnel Information Management System in the Personnel division. The janitor personnel information is collected from the Officer Personnel Information Management System in the affair division. Common database is developed and maintained by Computer and Network Center in NCTU. It is constructed according to the requirement result from discussing, meeting with each division and questionnaire work. The final database structure made is as the Figure 2-15 shown as following.

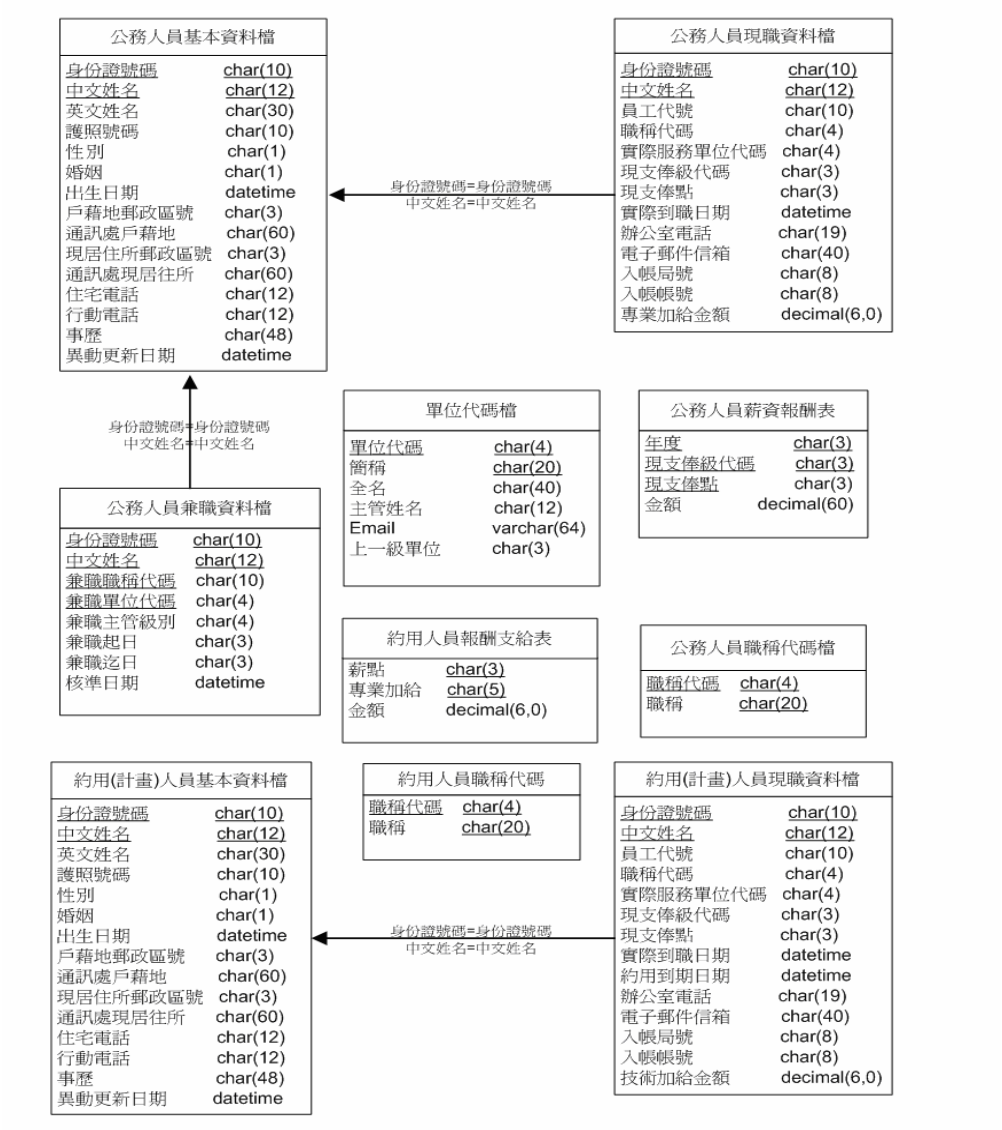


Figure 2-15 Database Schema of Common Database

2.7. Library

2.7.1. Background

NCTU library offers reading service, book borrow and return service, referencing service for students and staff in NCTU. The E-Office systems in the Library are developed by company named TRANSTECH. The system is an automated library system which includes System Management Module, Making up Booking List Module, Circulation Module, Purchasing module, WebPAC module and Journal module. System management module manages the system user accounts and access permission of users. Making up book list module offer the librarian to make up and modify the book list. Circulation module consists of circulation management subsystem and Counter subsystem. Counter subsystem offers the librarian reader management, borrow and return book management, book renewal and reservation and switching suspension of right to fines. Circulation management module provides to setup the right of readers and informing message for the due date to return borrowed book. Purchasing module processes the purchasing task, importing bibliopole data, duplicate copy checking and so on. WebPAC module provides the services of viewing records, book renewal, book reservation and book enquiries. The information that Library need from the administrative division in campus is the data of student and staff.

2.7.2. Software Environment

The e-Office system of the Library is developed by using Object-Oriented developing tool Delphi. The system is based on Client/Server architecture. The system user should set up the client application in Windows Operating System, login into the system server, then start to use the system. The Environment of the Database Server and the host that provides enquiries functions for readers in the Library is as following table.

Host	IP Address	System Edition	Database Edition
Database Server	140.113.39.66	SunOS 5.8 Generic_108528-27 sun4u sparc SUNW ,Sun-Fire-880	Sybase Adaptive Server Enterprise/12.5.0.2/EBF 10746 ESD#1
Web Server	140.113.39.50	Windows 2003 Server (service pack1)	

2.7.3. Database

The DBMS in Library is Sybase database. Because different characters have their related borrowing rights and limitation, the librarian has to determine which character the applier is before adding a new user. There are two integration issues in library:

1. The characters of numbers are significant to the number of book to borrow from, the limitation and so on. There is difficulty to update the data of staff by using the data of common database. Because the character field of staff data is not mach well with the character field in the library database. The characters and their corresponding right is as the following diagram.
2. The account id is the student id/staff id plus number of lost times of ID card. This is another attention should be paid when integrating the library database.

身份類別	類別名稱
A	交大-大學生
B	交大-研究生
C	教師研究員副研究員
D	職/工營
J	選舉生
K	實習輔導員
N	校友
P	退休人員
R	借調(交大職員)
S	留職停薪(交大職員)
U	交大-助理研究員

Reader types in the library

代碼類別	內容	維護單位
A	博士後研究員或客座米助理	人事室
B	全職工讀生	出納組
D	諮商師	人事室
E	編制內約聘人員(人事費)	人事室
F	使用學校經費約雇人員	人事室
G	使用各單位經費約雇人員	人事室
H、Z	計畫助理	人事室
I	國防役	人事室
J	工友、技工、管工	事務組
K	臨時工	事務組
L	稀少性科技人員	人事室
M	教官	人事室
N	國科會支薪人員	人事室
P	警衛	人事室
PH、PQ	校外人士	出納組
PR	兼職人員	出納組
PT、PTA	兼任教授	人事室
PX	計畫研究教師	人事室
R	退休人員	人事室
S	職員	人事室
T	教師	人事室
W	退休工友	事務組
X	約聘教師->系所造冊	人事室
XB	約聘教師	人事室

Personnel types in campus

Figure 2-16 Tables of reader types in library and personnel types in campus

CH 3 Problems

We introduce discovered problems after visiting and consulting several divisions. We collect these problems that should be solved when integrating e-Office systems in campus. We describe the problems for each division and then summarize these problems in the integration points of view.

3.1 Discovered Problems in Divisions

The contents of the problems in divisions are as the followings. We individually describe the problems of each division.

Registrar Division

1) e-Office systems in Registrar division are still running on MS-DOS environment. Only small part of Windows e-Office system is developed. They need to transfer data between new and old e-Office systems to complete their business.
2) Many divisions in campus require student data to process their businesses. The student data file is delivered to the divisions by email or disk.

Curriculum Division

1) Sometimes the teaching hours miss accounting is caused by late notification of personnel status data modification. Up to date teacher's personnel data is required by Curriculum division.
2) The personnel data is also maintained by Curriculum division. There is no electronic data transferred between Personnel Division and Curriculum division. All data modification notification is paper work. Although there is common database currently, they still don't find the way to make use of it.

File and Document Division

1) Currently, there is no complete data source of personnel English names. It makes English mails process difficult. An English name booking system is developed. Students and faculty access the system to book their English names.

Campus Security Division

1) The personnel data and student data are independently maintained in Campus Security division. Campus Security division is not notified of personnel status modification.

Personnel Common Database

1) Only several divisions currently uses Common database. The comprehensiveness and correctness of personnel data is still challenged.
2) There is only personnel data in common database. Student data is not integrated into it yet.

Library

1) The reader id of Library is inconsistent with employee id and student id. The reader id is one digit more then student id or employee id.
2) The types of readers can't match the personnel status or types of students. The access right and borrow limitation are related to the reader types. This makes difficult to transfer data and add readers account.

Bursar

1) Bursar maintains their selves own personnel data independently. For financial critical responsibility, the personnel data in Bursar is the most comprehensive in campus. This adds another burden to Bursar.

3.2 Problems summery

We summarize the problems mentioned above in this section. The summery is as the followings.

1) The same data is maintained by several data. For example, Personnel division, Curriculum division, Bursar and Campus Security division maintain personnel data individually. This may produce data inconsistency problem.
2) It wastes efforts and resources to maintain public sharing data in several divisions, such as personnel data.
3) Lack of comprehensive and correct personnel and student data source.
4) There is no effective mechanism of resources sharing and data exchange channel and forming many isolated information islands.
5) Organization-based structure of web pages is not convenient for users. We need user-based web pages.
6) Lacking of standard of data make it difficult to transfer data.

CH 4 Integrated e-Office Architecture and Design Methodology

4.1. Integration Categories

There are many kinds of integration methodologies which have their advantages and disadvantages. Interface integration, common database integration and frond-end integration are involved in our designed e-Office system architecture. We introduce these three integration categories shown as the followings.

4.2. Interface Integration

Interface integration means that the integration is accomplished by the interfaces defined between the entities. Interface integration, in administrative e-Office system here, shows that we define the interfaces between individual cluster of each division and the system to glue all clusters together. In other word, the administrative e-Office system will communicate with each individual cluster through the contracted interfaces.

There are steps about interface integration. There is a lot of data each division, the first step is to find out what data is required and what is not for integrated e-Office systems. The data we need here is the data that is related to people in the campus. The second step is to define the interfaces to retrieve the data. Of course there should be some implemented objects behind the defined interface, we only focus on the interfaces and the object can be implemented in several ways. The last step is to define the mediator. The mediator here contains two roles in the integration. One is the broker that connects to each divisions, the other is to offer more convenient access interface to access data of divisions. Hiding tiresome individual location of interfaces each division is another function of the mediator. The relationship of mediator, interfaces, individual databases of the divisions and the integrated e-Office system is showed as the following diagram. The following options describe the steps about interface integration in the integrated e-Office system.

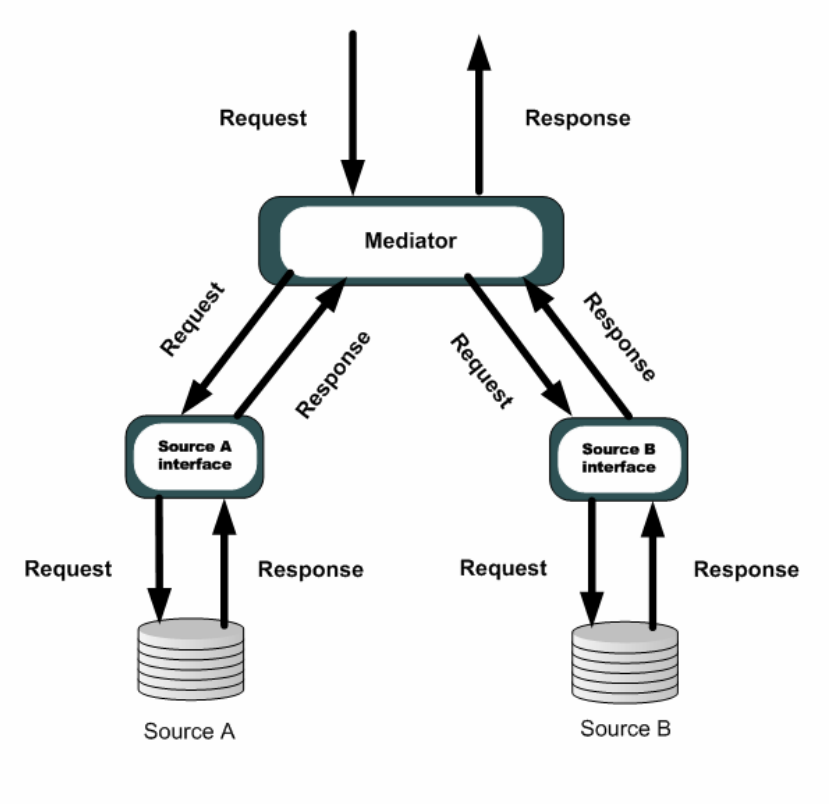


Figure 4-1 Interface Integration Diagram

Find out data related to people in each division

The first stage of interface integration is to find out what data we need in each division. Each division consists of their self won features and data related to people in campus. We have to find out those data that will be accessible for the system. According to the background description, we can find out the required data as the following tables.

Division	Data
Registrar	Student Status Academic Result
Curriculum	Teaching Hours Accounting Fee
Campus Security	Parking ticket
Document & File	Mail Records
Library	Borrow Book records
Bursar	Salary Tuition Fee Income Tax Fee
Personnel Common Database	Personnel Data

Define interfaces for each division

After the required data is found, we are going to define the interfaces for each division for retrieve their data. In the following sections, we design interfaces of each division based on their features and operations. The features and operations of individual division are described at the beginning. Then the table lists of operations and their message types are followed by their descriptions.

4.2.1. Registrar Division Interface

Register division management student status, student data and academic result of students. Therefore, the services register division can provide are students' data enquires and academic result enquires. Students may need to modify their contact information, an operation for modifying the contact information is provided. Also, there are employees in divisions whose affairs are responsible for some classes in campus. An operation for access students' data by class is provided.

Operations/Message Types

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getStudenData	Input	*getStudentDataRequest*	*StudnetName*	*String*
			StudentID	*StudentIDtype*
	Output	*getStudentDataResponse*	*Response*	*StudentData*
	Fault	*InvalidName*		*Client*
	Fault	*InvalidID*		*Client*

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getStudentDataDetails	Input	*getStudentDataDetailsRequest*	*StudnetName*	*String*
			StudnetID	*StudentIDType*
	Output	*getStudentDataDetailsResponse*	*Response*	*StudentDataDetail*
	Fault	*InvalidName*		*Client*
	Fault	*InvalidStudentID*		*Client*

Operation	Msg. Type	Message	Parameters	Type
getAcademicResult	Input	*getAcademicRequest*	*StudnetID*	*StudentIDType*
			Semester	*SemesterType*
	Output	*getAcademicResponse*	*Response*	*AcademicResult*
	Fault	*InvalidStudnetID*		*Client*
	Fault	*InvalidSemester*		*Client*

Operation	Msg. Type	Message	Parameters	Type
getClassList	Input	*getClassListRequest*		
	Output	*geClassListResponse*	*Return*	*ClassList*

Operation	Msg. Type	Message	Parameters	Type
getClassMemberList	Input	*getClassMenberListRequest*	*Class*	*ClassType*
	Output	*geClassMemberListResponse*	*Return*	*StudnetList*
	Fault	*InvalidClass*		*Client*

Operation	Msg. Type	Message	Parameters	Type
modifyContactInfor	Input	*modifyContactInforRequest*	*StudentID*	*StudentIDtype*
			Address	*Address*
			PhoneNumber	*PhoneNumber*
			Email	*Email*
	Output	*ackModify*	*Response*	*boolean*
	Fault	*InvalidStudnetID*		*Client*

Operation	Msg. Type	Message	Parameters	Type
getNewStudnet	Input	*getNewStudentRequest*		
	Output	*getNewStudnetResponse*	*Response*	*StudentDataList*

- *getStudentData*

The purpose of this operation is to retrieve the student data by give the student id or student's name. We only need to give either student id or student's name to the operation parameters. If both of two parameters are input, the response will base on the student id. *InvalidName* is returned when there is no student whose name is the same with the input name. The *InvalidStudnetID* indicates the input student id does not exist.

- *getStudentDataDetails*

This operation mostly is the same with *getStudnetData* operation besides the response will give more detailed data of the student.

- *getAcademicResult*

The academic result of the student in the designated semester will be give by using this operation. The academic of the student whose student id is *StudentID* in the Semester result will be returned if the parameters are valid. If the input parameters are not valid, *InvalidStudnetID* or *InvalidSemester* will be return. *InvalidStudentID* means the input student id is not valid. *InvalidSemester* means the input semester is not correct.

- *getClassList*

This operation will return the list of classes in campus.

- *getClassMemberList*

The member list of the designated *Class* will be return by using this operation. If the input class does not exist, *InvalidClass* message will be returned.

- *modifyContactInfor*

This operation provides a way to modify the contact information of student. The contact information will be modified when the input parameters are not empty. For example, if Address parameter is empty, then the address information of the student will remain the same. The *InvalidStudentID* fault massage will be returned, if the input student id is not valid. If every thing goes well, *ackModify* message is returned.

- *getNewStudnet*

It is needed to get the new students in the semester for some employees in divisions. This operation provides to get the list of new students data.

4.2.2. Personnel Common Database Interface

Personnel common database contains the personnel information of staff in campus. The only two operations are needed. One is to get the data of the staff and the other is the one to allow staff to modify the contact information themselves.

Operations/Message Types

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getStaffData	Input	*getStaffDataRequest*	*StaffName*	*String*
			StaffID	*StaffIDType*
	Output	*getStaffDataResponse*	*Response*	*StaffData*
	Fault	*InvalidName*		*Client*
	Fault	*InvalidID*		*Client*

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
modifyContactInfor	Input	*modifyContactInforRequest*	*StaffID*	*StaffIDType*
			Address	*Address*
			PhoneNumber	*PhoneNumber*
			Email	*EmailAddress*
	Output	*ackModify*	*Response*	*boolean*
	Fault	*InvalidStaffID*		*Client*

- *getStaffData*

The purpose of this operation is to retrieve the student data by give the student id or student's name. We only need to give either student id or student's name to the operation parameters. If both of two parameters are input, the response will base on the student id. *InvalidName* is returned when there is no student whose name is the same with the input name. The *InvalidStudnetID* indicates the input student id does not exist.

- *modifyContactInfor*

This operation provides a way to modify the contact information of the staff in campus. The contact information will be modified when the input parameters are not empty. For example, if Address parameter is empty, then the address information of the staff will remain the same. The *InvalidStafftID* fault massage will be returned, if the input staff id is not valid. If every thing goes well, *ackModify* message is returned.

4.2.3. Campus Security Interface

The main e-office system in campus security division is Vehicle Management System. Beside the application for a Parking permit, the parking ticket is also relative to students and staff in campus. Therefore, we define parking tickets enquires in campus security division.

Operations/Message Types

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getParkingTicketRecord	Input	*getParkingTicketRecordRequest*	*CampusID*	*CampusIDType*
	Output	*getParkingTicketRecordResponse*	*Return*	*ParkingTicketRcord*
	Fault	*InvalidCampusID*		*Client*
	Fault	*NoRecord*		*Client*

- *getParkingTicketRecord*

We can get the parking ticket records of some people in campus by using this operation. When there are parking ticket records, the parking ticket records are returned. If there is no parking ticket that belongs to the people, *NoRecord* message is returned. If the *CampusID* does not exist, *InvalidCampusID* message is returned.

4.2.4. Document and File Division Interface

All the mails sent to NCTU are collected in Document and File division. Mail Management System records the registered mails information. Therefore, it is required to get the records of someone's registered mail.

Operations/Message Types

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getMailRecord	Input	*getMailRecordRequest*	*CampusID*	*CampusIDType*
	Output	*getMailRecordResponse*	*Return*	*MailList*
	Fault	*InvalidCampusID*		*Client*
	Fault	*NoRecord*		*Client*

- *getMailRecord*

This operation is used for getting the personal new mails that are not took back yet. *CampsID* is either student id or staff id in campus. If the input *CampusID* is valid, the list of new mails is returned. The fault massage *InvalidCampusID* indicates the input *CampusID* is not valid. *NoRecord* means there are no new mail of the people**.**

4.2.5. Library Interface

The administrative issue related to the library is the borrow book record of people in campus. A typical example is checking the borrow book records of students in the leave-school procedure. Therefore, an operation to get the borrow book records is needed.

Operations/Message Types

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getBorrowBookRecord	Input	*getBorrowBookRecordRequest*	*CampusID*	*CampusIDType*
	Output	*geBorrowBookRecordResponse*	*Return*	*BorrowBookRecord*
	Fault	*InvalidCampusID*		*Client*
	Fault	*NoRecord*		*Client*

- *getBorrowBookRecord*

This operation is used to query the personal borrow book records in the library. The *CampusIDType* is either staff id type or student id type. The response are the borrow book records of the person whose student id or staff id is *CampusID*.

4.2.6. Bursar Interface

Salary records, income tax fee of employees and tuition fees of students are the information related to the people in campus. Therefore, we need the operations to get the salary, tuition fee, income tax fees of someone. The operations are as followings.

Operations/Message Types

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getPaymentRecord	Input	*getPaymentRecordRequest*	*PaymentID*	*PaymentIDType*
	Output	*getPaymentRecordResponse*	*Return*	*PaymentRecord*
	Fault	*InvalidPaymentID*		*Client*

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getSalaryRecord	Input	*getSalaryRecordRequest*	*CampusID*	*CampusIDType*
			Month	*Date*
	Output	*getSalaryRecordResponse*	*Return*	*PaymentRecord*
	Fault	*InvalidNationalID*		*Client*
	Fault	*NoRecord*		*Client*

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getTuitionFee	Input	*getTuitionFeeRecordRequest*	*StudentID*	*StudentIDType*
	Output	*getPaymentRecordResponse*	*Return*	*PaymentRecord*
	Fault	*InvalidStudentID*		*Client*
	Fault	*NoRecord*		*Client*

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getIncomeTaxRecord	Input	*getIncomeTaxRecordRequest*	*CampusID*	*CampusIDType*
	Output	*getIncomeTaxRecordResponse*	*Return*	*InComeTaxRecord*
	Fault	*InvalidCampusID*		*Client*
	Fault	*NoRecord*		*Client*

- *getPaymentRecord*

The input parameter *PaymentID* is the payment id of the required payment. The payment record contains payer's name, the amount of money, date and check number is returned if there is payment record corresponding to the payment id.

- *getSalaryRecord*

This operation takes student id or staff id as the input parameter and returns the salary records. The fault messages consist of *NoRecord* which means there is no salary record for the input parameter and *InvalidCampusID* which is not a valid student id or staff id.

- *getTuitionFee*

The tuition fee record of the input parameter is returned. StudnetID parameter is student id. Two fault types mean the input student id is not a valid student id and there is no record for this student id.

- *getIncomeTaxRecord*

Income tax fee record of someone is the return data of this operation. Input parameter is student id or staff id of the people.

4.2.7. Curriculum Interface

Course selection is the primary features of Curriculum division. therefore, the operations to get credit fees and courses registration information are needed. Teaching hours accounting is also another duty work in the division. An operation of getting teaching hours fees would be included.

Operations/Message Types

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getCreditFee	Input	*getCreditFeeRequest*	*StudentID*	*StudentIDType*
	Output	*getCreditFeeResponse*	*Return*	*CreditFeeRecord*
	Fault	*InvalidStudentID*		*Client*
	Fault	*NoRecord*		*Client*

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getTeachingHoursFee	Input	*getTeachingHoursFeeRequest*	*SfaffID*	*StaffIDType*
	Output	*geTeachingHoursFeeResponse*	*Return*	*CreditFeeRecord*
	Fault	*InvalidStaffID*		*Client*
	Fault	*NoRecord*		*Client*

Operation	**Msg. Type**	**Message**	**Parameters**	**Type**
getCourseRegistration	Input	*getCourseRegistrationRequest*	*StudentID*	*StudentIDType*
	Output	*getCourseRegistrationeResponse*	*Return*	*CourseRegistrationType*
	Fault	*InvalidStudentID*		*Client*
	Fault	*NoRecord*		*Client*

- *getCreditFee*

This operation takes student id as its parameter and returns the credit fee record of the student. One of fault messages is *InvalidStudentID* which means the input student id doesn't exist in the curriculum division. The other means the student doesn't have credit fee record. He/she might not take any course in the semester.

- *getTeachingHoursFee*

The teaching Hours Fee record of the teacher whose staff id is *StaffID* is returned by this operation. The fault messages mean the value of the input parameter *StaffID* is invalid or there is no records of the teacher.

- *getCourseRegistration*

This operation might be used by the register division when the course selection is finished. The course registration record of the student whose student id is *StudnetID* is the returned data. The fault messages are the same with getCreditFee operation.

4.2.8. Mediator Interfaces

Because the data of division locates in different place, it is required to remember a location for each division. If we are going to access data from various divisions, we have to remember many access locations. Moreover, the campus information system keeps involving. If there is an interface to be modified, it may also be required to modify many other places of the system. The design of a mediator will reduce the range of the modification and we only have to access the mediator rather than accessing many other places. The mediator delivers the requests of users to the proper locations, receives the responses and sends back the responses to the users.

The interface definition of the mediator bases on the requirement of the application. We may only simply define it as the collection of interfaces of divisions, process the conflict names then redirect the corresponding interface functions to the proper locations. The user should understand the interfaces before using them. Another approach is to define the global data model of the system and provide SQL query for users. The interface of the mediator is used to receive SQL query. SQL query is processed through query reformulation, query optimization and is executed by the execution engine. This is the research of data integration, we don’t discuss it here.

4.3. Front-End Integration

4.3.1. Current Front-End in NCTU

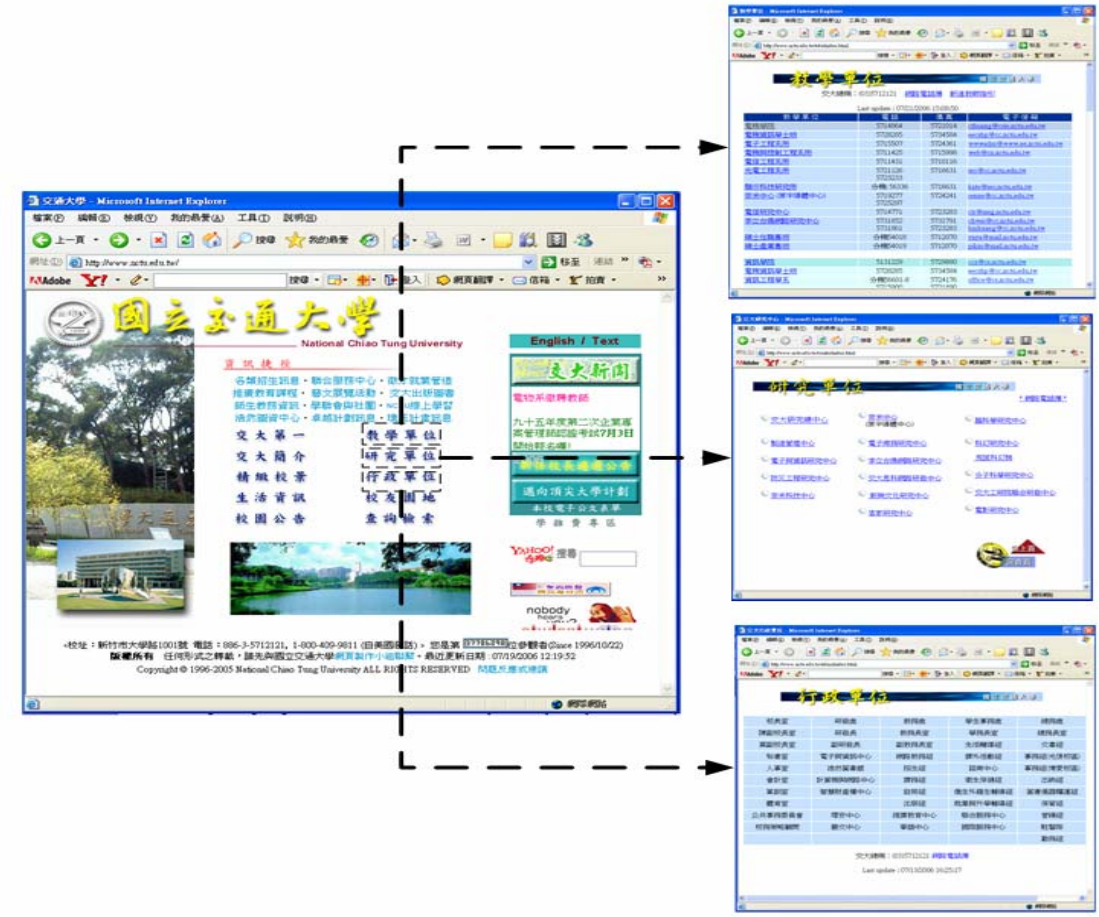


Figure 4-2 Current Front-End Organization in NCTU

Front-end is the interactive portion between users and the system. Front-end design of a system is also one of the keys to differentiate good systems from bad ones. The campus information system keeps involving. At the beginning, the static HTML document appears. It only can provide the static material browsing or downloading. Latter on, the dynamic web pages show up and provide custom dynamic web pages. Currently, each division has its own web site that is used to publish the announcement, apply for an application, enquiries and register the chosen states. The connection of web pages is based on the organization hierarchical structure. The root page is NCTU main page (www.nctu.edu.tw). We can link to the web pages of the division based on the hierarchical structure. Figure 4-2 shows the web pages organization in NCTU main page. In other word, NCTU main page

integrates web sites of administration divisions, research center and the academic departments in NCTU. This kind of web pages distribution is convenient for the divisions. Each division only needs to put their material on their web pages. In other word, the pages only contain information related to the individual division. There are so many divisions in campus. It is not possible for users to visit the web pages of individual divisions one by one. Therefore, the announcement of divisions may be not effective. Furthermore, the web application pages that are provided for people to access in campus locate in many different URLs. A web application assigns a user/password for each user. It is not convenient for users to access the web pages. How to provide users useful information and eliminate tiresome user/password input are the key points of front-end integration.

Apparently, the NCTU web pages are currently organization-based. We need a front-end integration for users. In other word, we need a user-based front-end integration. There are different issues that depend on the integration level of front-end integration.

4.3.2 Hyperlinks Collection

Hyperlink collection means to gather the related hyperlinks for users to access by creating a new web page. A complete survey of web sites is necessary at the beginning. The disadvantages of hyperlink collection are that users are still required to input the passwords one by one to access the different information. This is the fastest and easiest way in front-end integration. The following diagram shows that Hyperlink collection page integrate tuition enquiries, course selection, WebPAC, salary enquires and in come tax fee enquiries for users.

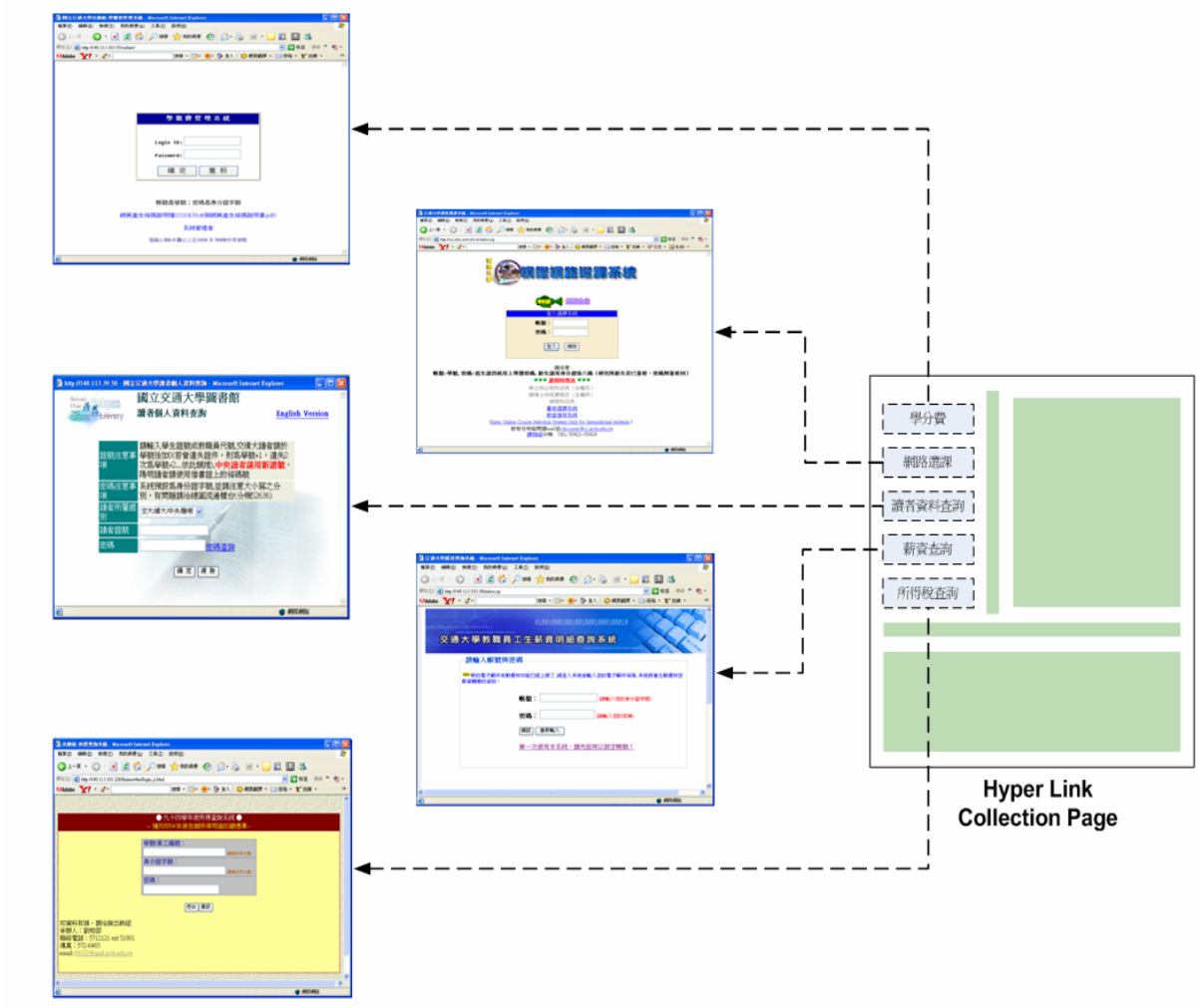


Figure 4-3 Hyperlinks Collection Diagram

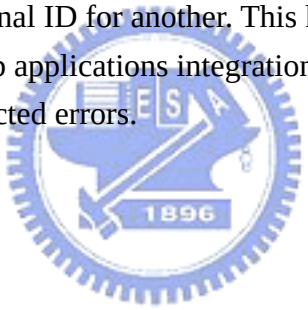
4.3.3 Integrated Authentication Pages

This kind of front-end integration involves the authentication of web application. We introduce two types of integrated authentication as the followings.

We need to modify the authentication portion of the original web applications to achieve the integrated authentication. A new authentication page will be written, too. All the web applications are integrated into one big web application. There are restrictions of this integration method unfortunately. The descriptions of the restrictions are as followings.

Restrictions:

1) All web applications are written in the same dynamic web page language.
2) If the user ids are different among the web applications, it is required to solve it first. The account matching is another difficulty here. The account id is student id in one web application and it might be national ID for another. This kind of integration would be suitable for a few number of web applications integration. Integrated too many web applications may lead to unexpected errors.



4.4. Common Database Integration

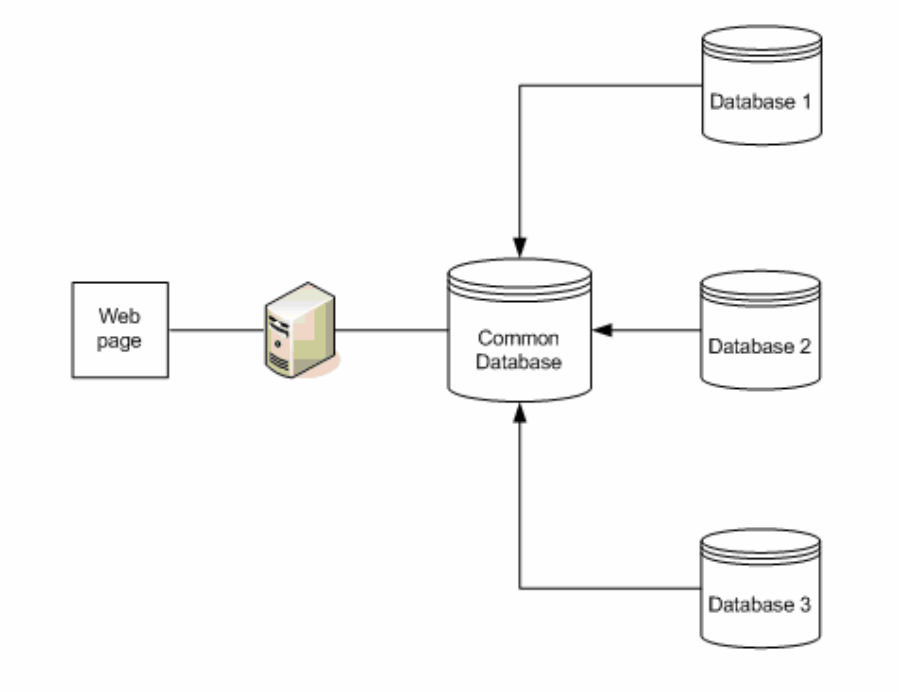


Figure 4-4 Common Database Integration diagram

There are many divisions in campus and each of them has its own data for operations. We can observe that although the operations in each division works well, the e-Offices systems are developed by individual divisions and maintain data individually. There is no sharing data between divisions and it causes the so called data inlands. It may lead to data inconsistency among divisions and the mistakes of operations. Many public sharing data are maintained by several divisions. It results in resources wasted and data source disorder. For administrative related issues, employees or students need to stop by more individual divisions one-by-one inn order to complete the process. Common database is aimed on solving the problem mentioned above. It integrates the data of individual divisions to form the data center in campus and provides data services for the divisions. Common database is a data union of data in all divisions. Furthermore, we can develop more advanced applications that base on common database. The common database will be the base of Campus Informationization.

The construction of common database is based on the data of the divisions. Process and collect the data of divisions. The integrated data is loaded into common database. How can collect the correct and overall data in campus and integrate them is the key of common database construction. The following options are the steps about common database construction:

4.4.1. Data Gathering

Gathering overall data in campus is the first and necessary stage of common database construction. Each division contains identical features and operations. The difficulties to gather the data is also different. We have to find out the data features of individual division through visiting all divisions. Gathering overall data in campus is the mission of this stage. The fully supports of school president and cooperation of the divisions is also necessary.

4.4.2. Data Analysis and Process

Each individual division is independent. Sharing data is not convenient. There are many public sharing data that are maintained by several divisions. This may cause data disordering and data inconsistency. For example, there are many divisions that need to use the department names and ids in NCTU. They maintain their selves own department names and ids. Each division defines its own department data individually. This makes it difficult to transform and share data. In this stage, we need to identify those public sharing data and data that is dedicate to the division. Figure 4-5 shows the public sharing data and data dedicated to the division. Each circle represents data in the division. All intersection portions belong to public sharing data. The other data in the division is data dedicated to the division. After classifying data, we determine what data to put in common database. We have to understand the operations and features of divisions in order to find out the exact data.

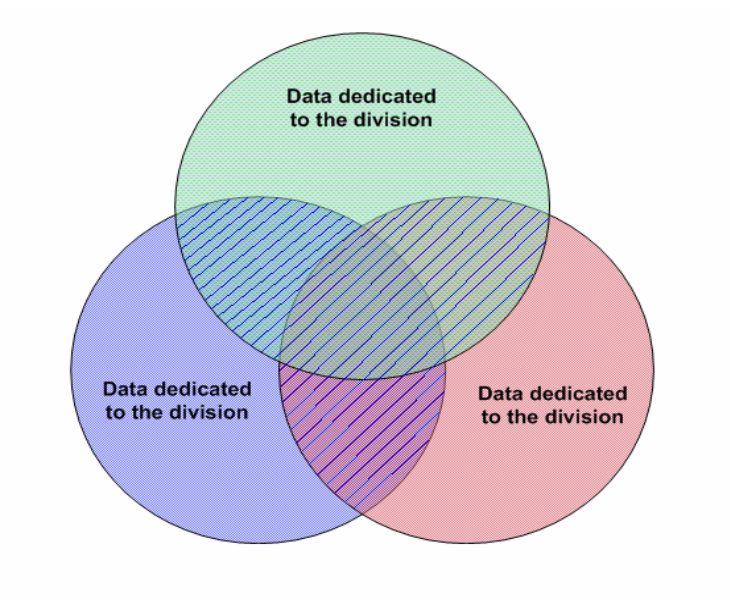


Figure 4-5 Public Sharing Data and Dedicated Data

4.4.3. Data Standardization

After analyzing what data should be put in common database, standardize the data is the next stage. The primary mission of this stage is to standardize the data of each division and determine the standard of the public sharing data and data related to people in campus. Integrate those standardized data into common database to achieve data unification. The definition of data standardization is related to the division that is responsible for the data. We need to consult those divisions and get the data requirements before we define the data standard.

4.4.4. Database synchronization

The last stage is to solve the data synchronization between common database and individual division. The features of the data in different divisions are not the same. The update frequency is also different. There are many database products have already provide good solution to data synchronization and real time renewal. There are some things we have to consider. The too high update frequency will reduce the e-Office systems performance in divisions. Therefore, it is required to determine the proper update frequency for each division. The update frequency is determined by the requirement and the urgency of the data. The critical data is real time updated. We can apply batch update to the other data.

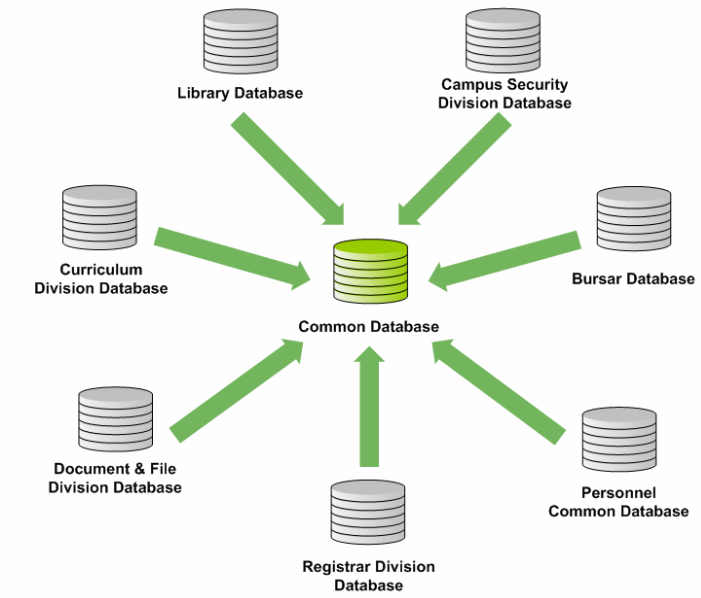


Figure 4-6 Integrate Data of divisions into Common Database

The above figure shows the common database based on the data of the divisions we visited. The followings are the data that we determine to be put into common database by analyzing the collected data and the background description above.

Currently we determine that the common database contains the following data.

Personnel Common Database

For the reason that the data of personnel common database is the designed data after visiting and analyzing the requirement of the divisions, we inherent the data of personnel common database. It contains the department names and codes established by the Secretariat. Personnel data of officers, agreement employees and project members are also included.

Bursar Database

The tuition fee is related to students. Salary records and income tax fee records have relationship with all people in the campus.

Registrar Database

Student status and academic results

Document and File Division Database

Registered mail records

Library Database

Borrow book records

Campus Security Division Database

Parking Tickets Records

Curriculum Database

Teaching hour fee and credit charged fee

4.5. Integrated e-Office System Architecture

The integrated e-Office system architecture is composite of the previous integration categories. Each kind of integration has its own benefits and integration scopes. According to the mentioned problems in campus, we design the architecture for satisfying those requirements.

Figure 4-7 shown below illustrates our integrated e-Office System Architecture. The gray components in the diagram currently exist in campus, for example, Curriculum Division Database, Library Database, Mail System Database, On-line Course Selection Server and so on. The other components belong to our designed portion.

The interface integration portion integrates the data dedicated to each division. Such as library enquiries interface, mail enquiries interfaces, bursar enquiries interface and parking ticket enquires interfaces are interfaces to access the data dedicated to the divisions. The interfaces' definition of the divisions is mentioned in Chapter 4.2.

The Data Access Server is actually the mediator introduced in the interface integration chapter. It provides overall campus data access services for Personnel Information Server and Data Management Server.

The Common Database contains the public sharing data such as personnel data, student data, departments data and so on. The Common Database acts as the centralized data source and solves the data inconsistency among divisions in campus.

Personnel information server provides the personnel web page for users. Manager server offer the manger more privileged data access. Overall, the designed architecture provides integration of data in campus, more user-friendly web application for users and the privileged data access web pages for employees in the divisions to collect required data. We describe those components introduced above as the following sections.

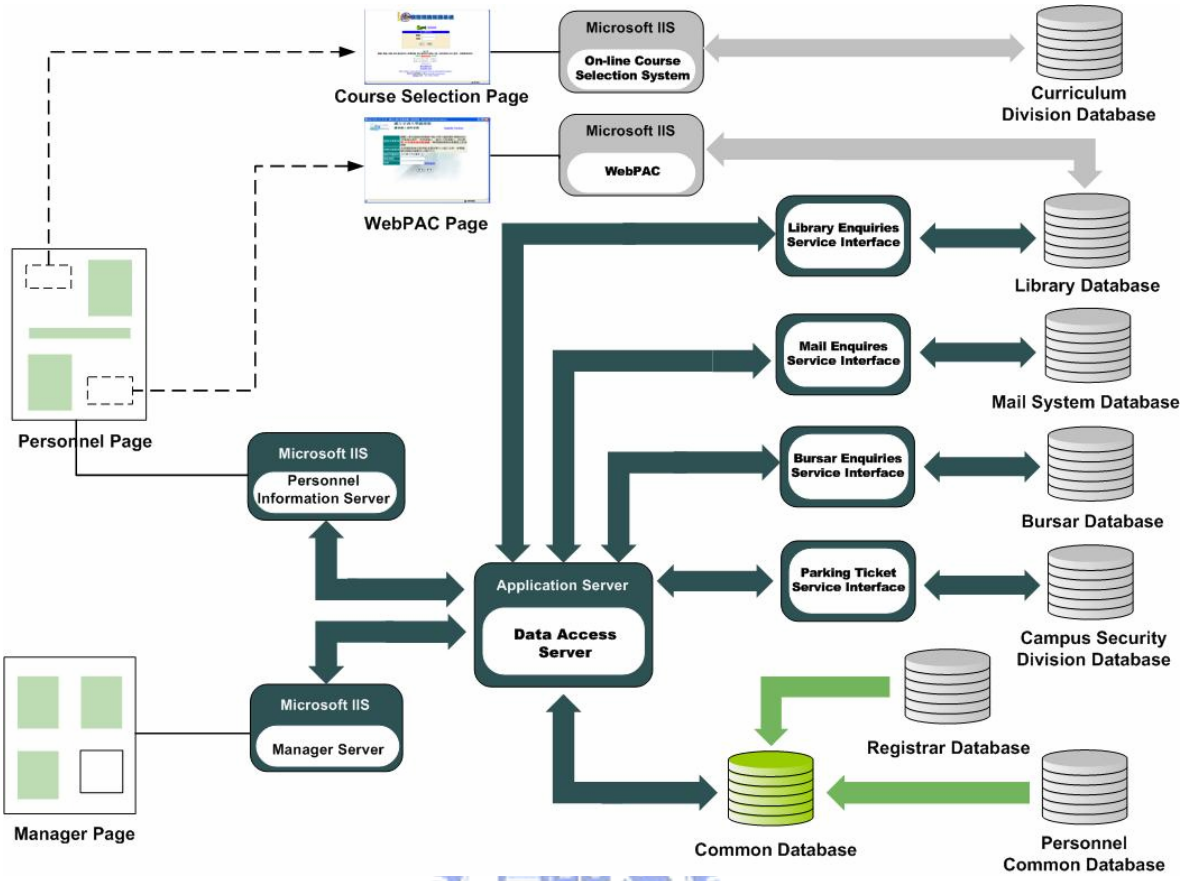


Figure 4-7 Integrated e-Office System Architecture

4.5.1. Front-end

The objective of front-end integration is to provide more user-friendly interfaces to users. The problems we illustrate previously, the NCTU web pages are built based on organization-based structure. It is not convenient for users to access. Currently, many divisions provide dynamic web pages for users to access their information. Most of them are only simple query functions. Furthermore, the scope of privileged web pages is not big enough for single sign on solutions. Those dynamic web pages which provide simple query functions can be replaced with new personnel web pages after integrating all divisions' data. The front-end portion that we design contains two web portions. Each one serves different kind of requirements. There are two roles which we are going to integrate the front-end for. Their descriptions are as followings:

Personnel web page and Personnel Information Server

Personnel web page integrates those dynamic web pages of applications or registration by hyperlinks collection. Those dynamic web pages written in the same dynamic web page language are also integrated by an integrated authentication page. For example, on-line course selection web page. After the user login the personnel web page, he/she can get all his/her status in campus.

So far, according to the data we collect now, personnel web page should integrate registered mail records, tuition fee, credit charged fee, salary record, income tax fee, parking tickets, academic results and book borrow records for users. Therefore, we can replace many current existing web applications including tuition fee enquiries, salary enquires and income tax fee enquires web application in Bursar. Also the academic results enquires web application in Registrar division can be replaced.

The on-line course selection system and classroom leasing system are ASP web applications in the Curriculum division, so we can integrate them with the personnel web page by integrated authentication page. Because the reader enquiries web page of the library is developed by the company outside the university, it is different system. We only collect the hyperlink of reader enquiries web page into the personnel web page.

Another function of personnel web page is the announcement of divisions. There will be an interface for divisions to announce information in the personnel information server. Users can also modify their contact information through personnel web page. This can ease the work required by the modification of users' contact information.

Manager web page and Manager Server

A new designed web page for the employees in divisions to access information related to their business. This is privilege access of data in campus. The purpose is to provide more convenient data access and save the back and forth data collection efforts. For example, the employee in Registrar division can access the status of the students within all divisions when he/she is responsible for the leaving school procedure of graduated students. The functions of manager page depend on the requirement of employees in divisions. More concrete functions will be discovered by consulting the employees in the divisions.

4.5.2. Common Database

Common database is aimed on becoming the data union of data in all divisions. Once there is data requirement, we can access those data in common database. Public sharing data mentioned in Common database integration is data that is needed by several divisions, but should not be maintained individually in each division. Data dedicated to each division has its own property. Some data changed every day and other almost remains the same. Putting those data that changed every day into common database is required to consider database synchronization issues. The more data in the common database, the more complexity the integration is. Furthermore, the traffic between common database and database in the division also increases the system load and reduces the performance. The followings are the disadvantages of putting data dedicated to each division into common database:

1) The more data in common database, the more traffic is required to maintain database synchronization.
2) The complexity of common database increases. It becomes harder to maintain common database.
3) Each division develops its own e-Office systems. The quality of e-Offices systems is diverse and maintained by different people. Integrating those data into common database may influence the service quality of common database.

Depending on the collected information currently, we can identify the public sharing data as the followings: Personnel Data, Student Data and Department data. The standard of the data should consult more divisions in campus before defining it.

Therefore, our consideration is that the common database contains only pubic sharing data in campus and it only focus on saving data inconsistency. Data dedicated to divisions can be maintained by each division. We can integrate those data by interface integration. It can not only reduce the back and forth traffic between the common database and databases in divisions, but also simplified the common database complexity.

4.5.3. Interfaces Integration

The interfaces integration primarily deals with the access of data dedicated to each division and provides more convenient access of overall data. After comprehensive data collection, data analysis and process, we can identify the dedicated data of divisions. According to those identified data, we can design the interface of each division. For example, the parking ticket record is the dedicated data of Campus Security division and a parking ticket records access interface will be designed. Another important part of interface integration is the mediator. The following diagram shows the situation without the mediator. Each new web application connects many interfaces of divisions to access the required data. One interface modification of the division influences many other web applications.

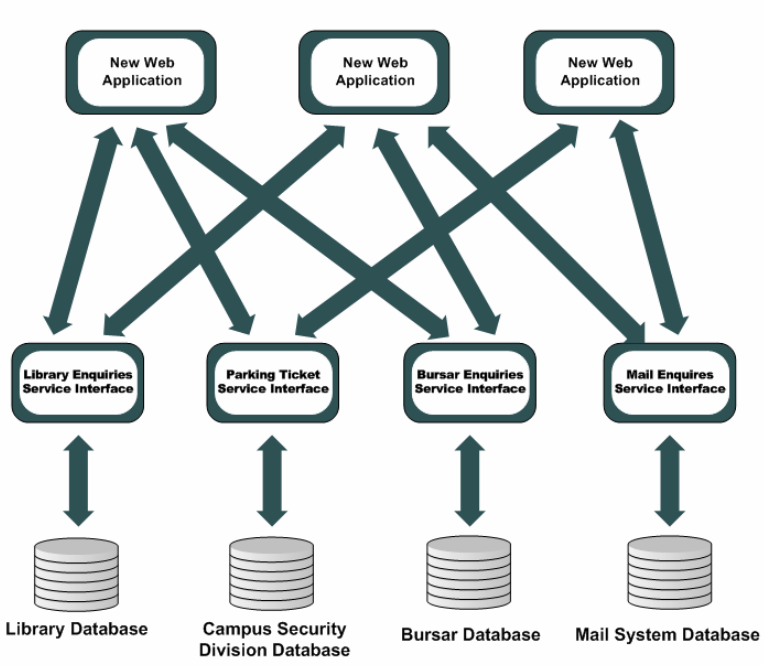


Figure 4-8 Interface Integration without the Mediator

The situation with the mediator is shown as following diagram. The mediator simplified the connections among new web applications and divisions' interfaces. New web application only needs to maintain the connection with the mediator. New web application will be more easily to be added into the system.

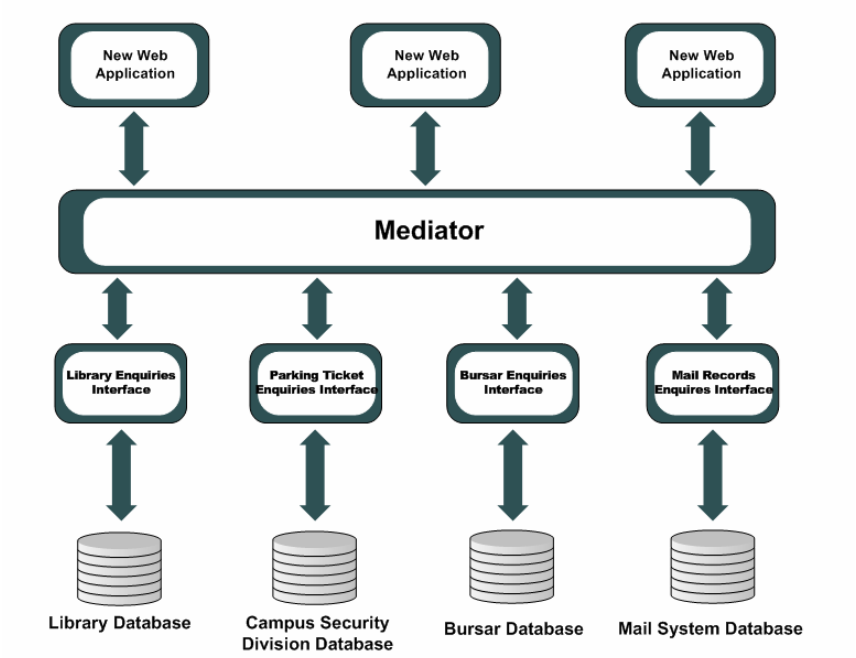


Figure 4-9 Interface integration with the Mediator

Figure 4-9 shows the interactions among Data Access Server, Mail Enquiries Interface and Mail System Database. We describe it as the followings.

1) A mail record request for the employee with *Employee ID* as his employee id is sent to Data Access Server from the application.
2) Data Access Server receives the request and find out the corresponding location, then send out GetMailRequest message defined in the contracted interface and *Employee ID* as its parameter.
3) The GetMailRequest message invokes GetMailRecord operation defined in the Mail Enquires Interface. GetMailRecord operation does necessary steps to connect to Mail System Database and produces SQL query, then transfer it to Mail System Database.
4) After the execution of the SQL query for the mail records of the employee, mail records are sent back to GetMailRecord operation.
5) GetMailRecord operation produces the contracted GetMailRecordResponse and pack the mail records into it, then send the response message back to Data Access Server.
6) Data Access Server receives the response messages and responses the mail records to the application.

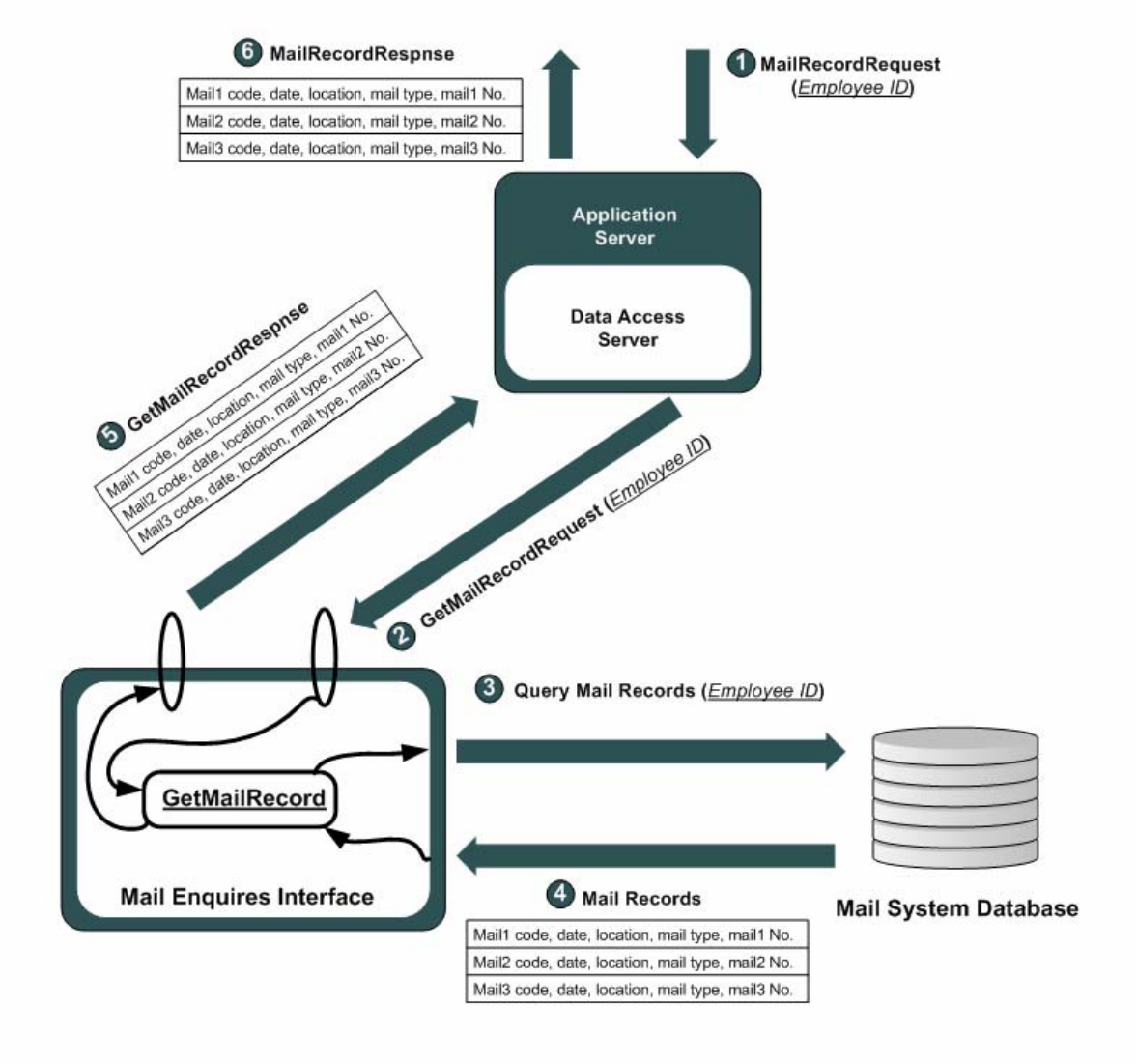


Figure 4-10 Data Access Server Mail Records Acquire Scenario

CH 5 Conclusion and Future Work

In this section, we make a conclusion and mention our future work.

5.1 Conclusion

Campus information integration is a huge work. A complete survey of every current system on campus is essential. Collecting comprehensive data and analyzing features and problems found to acquire the requirement of campus information integration, then design the system integration according to the requirements. In chapter 2, we introduce several visited divisions, their operations, features and current e-Office systems. We also describe the problems we discovered and responses of employees in each division. After so many years of campus information development, NCTU web pages have become big and mass. A front-end integrated page for users is necessary. Authentication integration is also one of important integration jobs. We mention several kinds of front-end integrations. Interface integration is to define the contract interface between the division and system. The mediator is designed for single location access and more convenient usages to division interfaces. Common database integration provides an integrated data sources by collecting, analyzing and processing data, standardization and database synchronization. An integrated e-Office system architecture is introduced. Isolating the public sharing data and data dedicated to each division is a more appropriate way. We consider the features of each division and data synchronization issues to design the integrated architecture.

5.2 Future Work

There are still a lot of integration works in campus. Our work is only the beginning of the integration. There are still much data that should be collected, because a comprehensive data collection is very important for the success of integration. We can build up a campus data center by improving the common database. After that Common database will be the base of campus information. Whole new campus applications can be built base on it. A mechanism that delivers the published information to proper users for each division will be developed. Implementation of the administrative e-Office system is also included in future work.

References

[1] NCTU Web Page http://www.nctu.edu.tw

[2] NCTU Computer and Network Center Web Page http://www.cc.nctu.edu.tw

[3] Christine Parent. Issues and Approaches of Database Integration. *Communications of the ACM*, Volume 41, May 1998.

[4] Yuyou Shi and Lifang Peng. Data Integration of campus informationization. *ACM International Conference Proceeding Series*; Vol. 113:837-829, 2005.

[5] Web Services Architect: EAI and Web Services
http://www.webservicesarchitect.com/content/articles/samtani01.asp

[6] Enterprise Application Integration Techniques
http://www.cs.ucl.ac.uk/staff/ucacwxe/lectures/3C05-02-03/aswe21-essay.pdf

[7] Web Data Integration
http://www.ccf-dbs.org.cn/idke/xfmeng/publication/talk/WSID%202003-10-11%20Yocsef%20short.pdf

[8] WS-I. Supply Chain Management Sample Application Architecture
http://www.ws-i.org/SampleApplications/SupplyChainManagement/2003-12/SCMArchitecture1.01.pdf