

# Developing an Integrated School Information System

Greg Baker

*Scotch College, Melbourne, Australia*

[Greg.Baker@scotch.vic.edu.au](mailto:Greg.Baker@scotch.vic.edu.au)

**Abstract:** This paper describes some of the issues in developing an integrated school information system that contributes to the management of a school. The system is based on an SQL relational database with a browser-based interface for all queries and transactions. The use of a standard interface makes it easy for end-users to navigate what is, in fact, a complex system. The user input and self-service provisions are important aspects of this system. This paper demonstrates that it is possible and feasible to develop an information system in a school that meets the needs of staff and is customized for the users' requirements.

**Key words:** Educational management, school information system, integrated school system, information infrastructure.

## 1 INTRODUCTION

Pauline Wilson walks into her first class for the day at Scotch College. She carries her lightweight notebook computer that is already started and connected securely to the school's wireless network. She calls her class list up on the notebook screen, quickly marks the absentees and then clicks the Update button. The results are immediately stored on the school's information system and are available to all those with authority to access it. Within minutes, accurate attendance records have been completed for all 1,850 students in the school. Students taking music instrumental lessons have been flagged, as have those going on an excursion to the Art Gallery.

Earlier in the day, the schools' rowing crews finished their early morning training on the river and the students have headed to the pavilion for breakfast. As they walk in the door, they pass their student cards under a

barcode reader to record their attendance at breakfast. At the end of breakfast, the teacher in charge authorizes a batch account to be processed and the charge for breakfast will be added to the student's next fee account.

## 2 INFORMATION NEEDS

Each school is different and has different information needs. Centrally designed systems can meet *most* of a school's needs *most* of the time, but not always. At Scotch College an information system that meets the needs of all the staff has been developed. As those needs change, so does the system. Functions are developed in response to requests from staff. In some cases, needs are anticipated and catered for.

### 2.1 Goals of the System

The goals of the system are to provide:

- Accurate, up-to-date data.
- An interface that staff find easy to use.
- Access from a variety of locations – at a desk, in a classroom, on the school oval, at home.
- Differing access rights to the data depending upon need.
- Designed so that it is integrated across the school
- Self-service access to reports.

## 3 INFORMATION SYSTEMS AT SCOTCH COLLEGE

A computer-based information system was first considered at Scotch College in 1978. An initial definition of the information system was prepared by the then Director of Computing and the Records Sub-committee of the school (Graham, 1978). The document was "... an attempt to define the records and information systems at Scotch College ... and to give an accurate and valid picture of our information system." (Graham, 1978).

At that time, the primary source of information about current and past students was known as the "Blue Card". The blue card was, in fact, a blue card that was regularly added to by members of staff. The blue card became the definitive document that recorded a student's progress through the school. In the early 1990's the blue cards were all transferred to microfiche and they remain a primary source of historical data for the school.

At that time also, "... a file of names and addresses of people associated with the school was kept on the Melbourne Stock Exchange System. These

records were the responsibility of the school's alumni group – the Old Scotch Collegians' Association (OSCA). This file was to form the basis of the school's information system.

The original coding specifications indicate that the school's initial decision was to develop the system using Cobol. However, when it came to time to carry out the development, the database chosen for use was DEC's Relation Database (DEC Rdb) and the programming language was VAX BASIC (Graham, 1985). A relational database model was chosen for the reasons that have subsequently become standard textbook criteria including:

- Minimization of duplication of data.
- Elimination of modification anomalies.
- Ease of data retrieval (Kroenke, 1983).
- Data consistency and accuracy.
- Multi level integrity (Hernandez, 2003).

With the increase in processing power available, the relational model has proven to be most useful in this context.

### **3.1 The First System**

The first system was developed internally and put into production in the early 1980s. The hardware used was a DEC VAX 11/750 using DEC Rdb, VAX BASIC and a fourth generation programming language called Datatrieve. The operating system was DEC's VMS. The system was a menu driven one with different access levels provided for different staff. Terminals were provided for all key staff and all teaching staff had access to the information system. Over a period of some 15 years, the system was enhanced, supported and maintained internally. During that period, the school had essentially *one* database that served the needs of the key constituents including:

- Staff.
- Admissions.
- Alumni (OSCA).
- Fund raising (development).

A key benefit of the system was that an address update in one area of the school was immediately reflected throughout the system and available to all authorized users.

### **3.2 A New System**

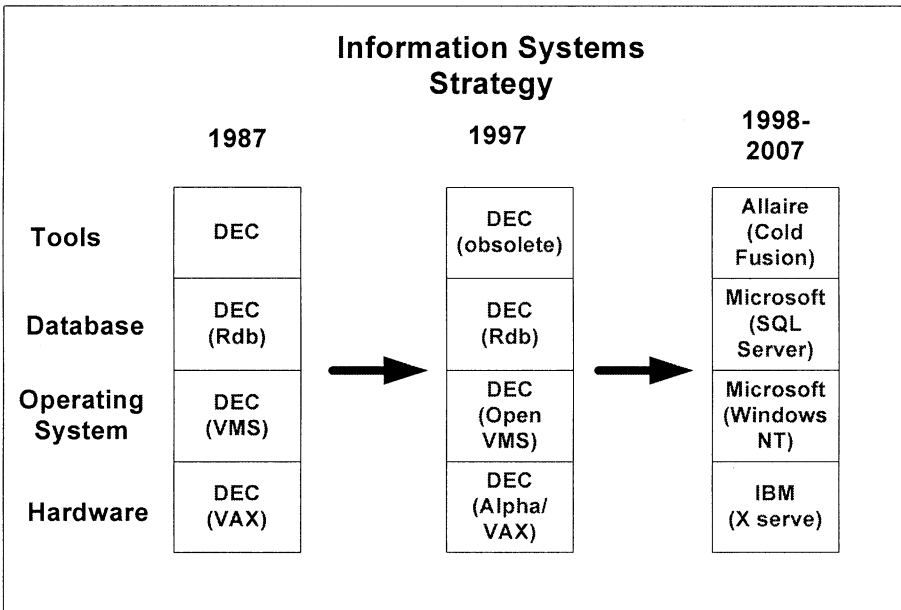
In 1996, Aspect Computing was engaged to advise on a replacement for the above system. The key reasons for considering a change were:

- The need to “... record and track relationships between students and school community members; and
- The need to furnish the Development Office with appropriate tools to assist in fund raising.” (Firkin, 1997).

Associated with changed needs were changes in the computer industry that affected the choice of tools and platform. In particular:

- DEC Computer Company divested many of its software tools to Oracle, including DEC Rdb.
- DEC itself was taken over by Compaq Computer Company, which was later absorbed into Hewlett Packard.

The first system was built using exclusively DEC tools – hardware, software including operating system and development tools, as well as support. The nature of the industry thus accelerated the need for change. At the same time, came an urgent need from within the school to develop a system that enabled the electronic recording of attendances. Figure 1 below shows the strategy used in terms of hardware, software and tools.



*Figure 1: Information System Strategy*

Other key reasons for developing the system internally included:

- No single externally produced software package met the school’s needs.
- The school’s needs continually change and there was a need to respond to those changes in a timely manner.

- The need for it to be supported on more than one platform (i.e. Microsoft Windows and Apple Macintosh).

At the time, most of the software packages that could be purchased ‘off the shelf’ were not Web-enabled.

### **3.3 Designing a New System**

A consultant from Aspect Computing assisted in the initial scope of the project (Aspect Computing, 1997). Key requirements of the project were to:

- Convert the existing school administration system to the new system with at least the same functionality.
- To have minimal impact on running of the school.
- To improve response times.
- To ensure that there was a standard user interface that was acceptable to users.
- To add new, required functionality to the system.

A new logical data model was created and the existing database was mapped into this new model. As noted by Tatnall and Davey (2001), the involvement of users is a key component of the success of a school system. Key users were consulted at each stage and helped specify the outcomes. Each of the various menu items was developed in response to a request from a member of staff – both teaching and non teaching. During 1998 the data structures were defined, created and brought across to the new system. The data transfer was successful.

The first new application developed was to track student attendances. This was done in about five working days and the first version was put into use at the start of 1999. The decision to use Allaire’s Cold Fusion as a rapid application development environment was taken after advice from a Gartner Group consultant at Gartner’s Annual IT Expo in 1998. That decision has proved successful but had some impact on staffing that is referred to later.

## **4 AN INTEGRATED SYSTEM**

The system was designed to ensure that it was integrated across a range of levels and functions within the organisation. The system can track a person from birth to death and beyond. The system is centred on a *person* as the key entity. A person has many different attributes as he or she interacts with different facets of the school. A person can be:

- A future student
- A current student
- A past student

- A future parent
- A current parent
- A past parent
- A future member of staff
- A current member of staff
- A past member of staff.

A person can have a number of those attributes at the one time (eg. a past student and a future parent). The previous system had used different entities for a person – eg. a student was one entity and his parents were different entities with different attributes.

Once the person is created on the system, he or she may never be deleted. (The only deletions from the system are those future enrollees to the school and their parents who never actually come to the school and have no other association with the school.) At the time of writing, there were more than 60,000 people on the system.

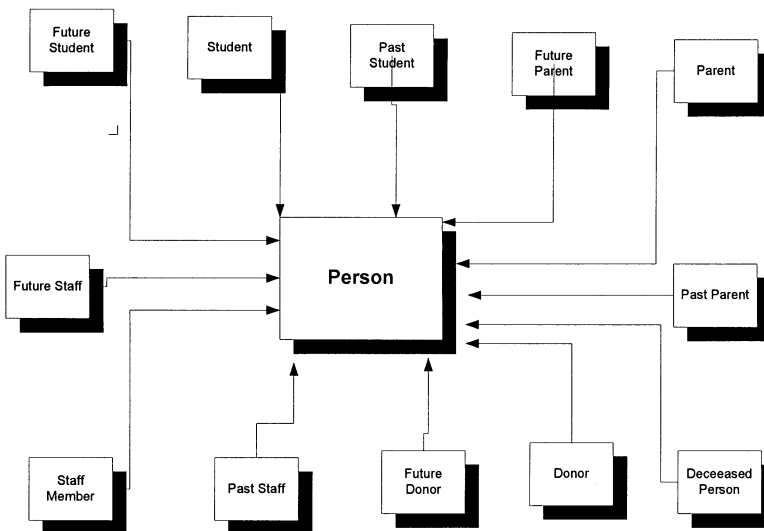


Figure 2: A person can have many different attributes.

## 4.1 Relational Database

The back-end system is a relational database made up of some 90 different tables. It would be nice to claim that it is fully normalized but, in the words of the database manager (Leon Seremelis), “It’s pretty good”. The relational database structure minimizes the duplication of data and makes it

relatively simple to update that data. The data storage, retrieval and querying is carried out using stored procedures in Microsoft SQL Server. The Cold Fusion pages generate queries that have been created by users. That is, the users can often decide on the type of data required.

## **4.2 Updating Addresses**

One of the key challenges is to keep address and contact details up-to-date and accurate. Address changes come from a variety of structures within the organization: They can come from:

- Teachers
- Reception
- Admissions
- Alumni
- Fund raising.

The following assumptions are made about addresses:

- Each person can have three addresses:
  - Home
  - Postal
  - Business
- A number of people can have the same address.

When an address is updated, the user is given the choice to make the change for:

- That person only, or
- The other persons at that address.

The system automatically shows all the other people who have the same address. Once the address is changed, it is immediately reflected throughout the system. For example, a parent's address change is immediately reflected in information looked at through the student view of the system.

## **5 BROWSER-BASED**

An important aspect of the system is to make it easy for staff to use. A decision was made very early in the development process that user access would be via a browser-based system.

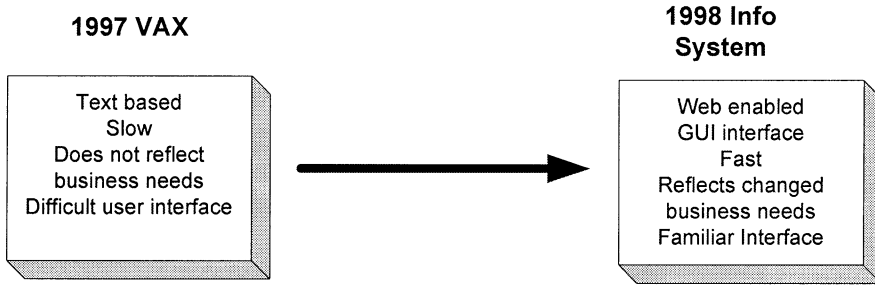


Figure 3 Moving from a text based system to web based system.

The standard browser used is Internet Explorer – an interface that all staff had become familiar with in using the Internet. One of the design goals was to make the system easy to use and easy to navigate. A browser-based system at the front-end poses some challenges in developing transaction-based databases in that interactions are stateless until a transaction is committed. The system that has been developed uses Microsoft SQL Server as the back-end database and Macromedia Cold Fusion as middle ware to provide the browser front-end.

The client or staff computer makes an HTML request that is sent to the Cold Fusion Server. This translates it into an SQL query that is forwarded to the SQL Server. The data is returned to the Cold Fusion Server and then presented in the browser to the user. Staff are able to access a wide range of data that is of use and to download appropriate data for mail merge or personal use.

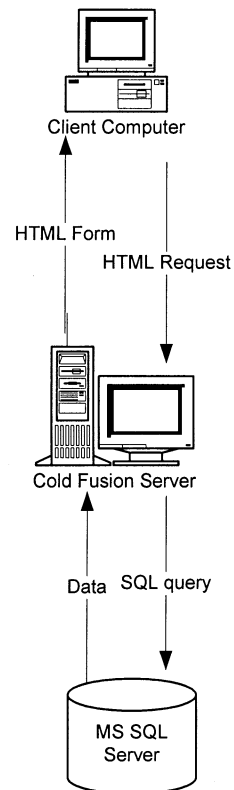


Figure 4 Microsoft SQL Server and Cold Fusion

## 6 MODULES AVAILABLE

The following are the key modules available:

- Student
  - Classes
  - Sports
  - Activities



- Learning difficulties
- Reporting
- Teachers
  - Classes
  - Excursions
  - Trips
- Attendance
- People
- Accounts
- Bookroom
- Alumni
- Fund Raising
- Enrolments
- Personnel

The following modules are planned for future development:

- Music School
- Swap shop
- Sporting Team Management

The modules and functionality are developed in response to requests from staff – both teachers and non teachers.

## **7 ACCESSIBLE ANYWHERE**

The data is accessible throughout the school campus for all staff. Each member of the teaching staff is provided with a notebook computer and an Ethernet connection at his or her desk. That connection provides access to the administrative network.

There are some 32 wireless access points throughout the campus providing access to the student network. Using a Cisco VPN Concentrator, staff can securely access the administration network and check class rolls and access all the information normally available at their desks. The Cisco VPN software includes a personal firewall on each staff member's notebook computer. At home, staff can access the information via the school's portal page using the Internet. Users are authenticated via the firewall and SSL encryption is used of queries and transactions.

The system is available to *all* staff, anywhere in the school. Access rights vary according to need. A limited number of staff are able to update information. Again, this is based on the member of staff's needs.

## 8 SELF-SERVICE ACCESS TO REPORTS

The provision of self-service access is of key importance in giving control to the users. Typically, the user provides the criteria for selection and the query is processed by the system. A range of files can be downloaded for use on the staff members' computer. In particular, class lists, sporting lists etc can be downloaded.

Communication with students and/or parents can be done by email – classes or other groups can be emailed from within the system. Immediately a class is changed for any reason, the email list dynamically changes. The need to alter groups in a Microsoft Exchange server is not required. Staff can download files based on their selections to be used in mail merge documents. A large range of standard merge fields is provided and these can be used in documents and mailing labels. Templates are provided for the mailing labels.

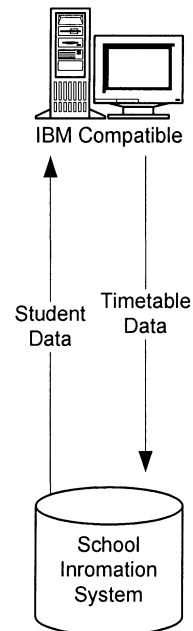
One of the outcomes of this process is that staff are not reliant on centrally produced reports or documents but are able to generate these themselves *when* they need them.

## 9 SYSTEMS INTEGRATION

There are, of course, other database systems operating within the school. These are being integrated into the architecture of the system so that updates between systems are made when required. One such example is the development of the school's timetable. Timetabling is a complex and specialized function. Rather than re-inventing the wheel, a timetabling package is used. In this case, student data is sent to the timetable package, student choices are entered into that package and a timetable is created. The timetable data is then sent back to the main information system. The timing of this is at the discretion of the person in charge of the timetable (i.e. in the hands of the key user).

Figure 5: School Information and Timetable System.

A second example of system integration is help desk software called Heat from Front Range. This is also SQL based and a help desk call looks up the data about users on the Scotch Information System and pulls it into Heat. Thus, user data is maintained in one source, the SIS, but help desk functionality



remains in Heat. Systems integration is a key issue for the future as we investigate further on line services and individual portal interfaces for staff, student, parents and alumni.

## 10 USER INTERFACE

The user interface is designed to:

- Make it intuitive for users to find information.
- Provide links to relevant information.
- Make it easy to carry out every day tasks.
- Be consistent.

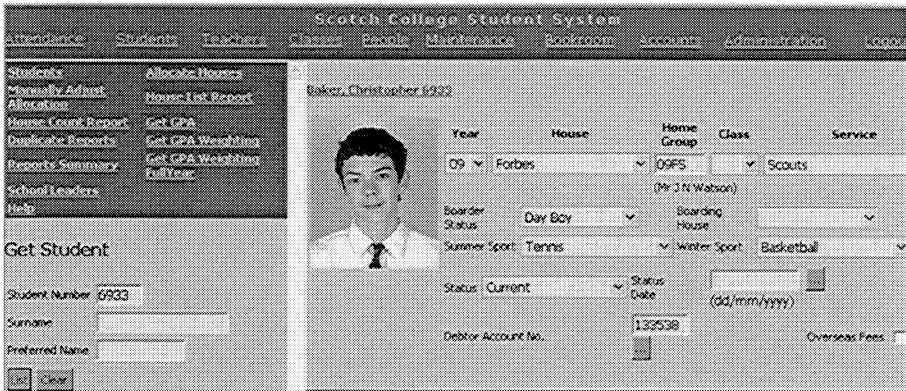


Figure 6: Information about a student

Figure 6 shows part of the user interface. The following are important parts of the design:

- Menus across the top as hyperlinks.
- Menus at the left, also as hyperlinks.
- Query window – student number etc.

The result of the query is in the right side of the window (partially shown in figure 6).

In Figure 7 it can be seen that a range of information is available by clicking on the hyperlinks. This is typical of the system – a vast amount of information available for all those who need to access it.

However, as the system has grown the menus have become somewhat clumsy in places and a new design is being implemented. Figure 8 shows a first version of that design that will be tested with users before implementation.

Residential Address: 37 Grove Rd HAWTHORN VIC 3122  
Postal Address: 37 Grove Rd HAWTHORN VIC 3122  
Medical Form Learning Profile

---

Email Teachers

Get Time Table for this Student

Get Demerit and Detention Report for this Student

Music Timetable

Year

Year

Figure 7: Further information about a student

Attendance	Student	Teachers	Classes	People	Self	Bookroom	Maintenance	Accounts	Administration
Mr Savage Main Office Dinner Groceries Membership Functions Donations Personnel									
Find Person									
Surname <input type="text" value="Savage"/>									
First Name <input type="text"/>									
Class Year <input type="text" value="9"/>									
Post Code (comma separated int) <input type="text"/>									
None <input type="checkbox"/>									
Business <input type="checkbox"/>									
Postal <input type="checkbox"/>									
None <input type="checkbox"/>									
Used Recently <input type="checkbox"/>									
<input type="button" value="Refresh"/>									
Person ID <input type="text"/>									
Current Role in School is Vice Principal									
Employment Status is Current									
									
Summer Sport:									
Winter Sport:									
Other Sport:									
Service:									
Person reports for: Dr Donaldson									
Qualifications: BSc Economics POCE VWAes MACE									
Department(s) Position(s) Held									
History Vice Principal									
<a href="#">Click here to view extended information or edit this information.</a>									
Today is 05 Apr 2004 this is NOT a School Day									
Day Period 1 Period 2 Period 3 Period 4 Period 5 Period 6									
1 History 095901 40									

Figure:8 A new interface

## **11 BUILDING LINKS WITHIN THE COMMUNITY**

The next key steps are to build links with the school community. The system currently provides personalized access for staff members, based on their login details and their position within the school.

The future involves development of the web interface and providing access to both students and parents when and where they need it. These groups can expect to have access to secure, personalised access:

- Student reports.
- Forthcoming events.
- Sporting activities and teams.
- Sports results.
- Activity sign-on.
- Fee account status, approval and payment.

The challenge is to ensure that these functions are:

- Developed using agreed standards.
- Communication is secure and encrypted.
- Integrated with specialized systems provided by third parties.

## **12 CONCLUSION**

This system was one of the first, if not the first, school administration packages developed in Australia using only a web front end for all users. This provides users with a familiar, intuitive interface that is easy to navigate. The user self-service provisions give control to users over *what* they need and *when* they want it. The system meets the needs of an individual school but could be developed to meet the needs of a wider range of schools.

The future integration into homes provides an exciting school/home interface that will improve communication and help the school to better meet the needs of its students and parents.

## **13 REFERENCES**

- Aspect Computing, (1997). "Scotch College Administration System Scope Issue 1.0", 10 December 1997, internal publication.
- Baker, G. M. (1997). "Information System Strategy", November 1997, internal document.
- Firkin, H. (1997). "Scotch College School Administration System Proposal Issue 1.0", Aspect Computing Pty Ltd, 31 October 1997.
- Graham, J. D. and the Records Sub-committee of Scotch College (1998), "Definition of the Information System of Scotch College, Melbourne", 1978, internal publication.

- Graham, J.D. (1985) – in conversations from the mid 1980s.
- Hernandez, M. J. (2003) “Database Design for Mere Mortals”, 2<sup>nd</sup> ed, Addison-Wesley, USA
- Kroenke, D. (1983). “Database Processing 2<sup>nd</sup> edition”, Science Research Associates, USA.
- Rees, M, White, A, White, B, (2001). “Designing Web Interfaces”, Prentice Hall, USA.
- Tatnall, A. and Davey, B. (2001). Open ITEM Systems are Good ITEM Systems. *Institutional Improvement through Information Technology in Educational Management*. Nolan, P. Dordrecht, The Netherlands, Kluwer Academic Publishers: 59-69.