

Web-based assignment submission and electronic marking

Peter Bancroft

Queensland University of Technology, Brisbane, Australia
p.bancroft@qut.edu.au

John Hynd

Queensland University of Technology, Brisbane, Australia
j.hynd@qut.edu.au

Fabian Dal Santo

Queensland University of Technology, Brisbane, Australia
f.dalsanto@qut.edu.au

Jim Reye

Queensland University of Technology, Brisbane, Australia
j.reye@qut.edu.au

Abstract: *The Online Assignment System (OAS) is a web-based assignment submission and electronic marking system for students at Queensland University of Technology. The OAS is used by students and teachers as a replacement for the old manual system which has been in place for 12 years. Both the old and new systems share many good points but the new system has several advantages for students and staff. Students find it convenient to use the OAS to submit assignments from home and later retrieve their marked assignments through the web rather than having to collect hard copies from the university. Online submission of assignments has not been well received by lecturers in the past because of the perceived difficulty of electronic marking and providing adequate feedback but this problem has been overcome by including a Java application (called F^2M^2), which is used to mark and insert comments through a graphical user interface. F^2M^2 is particularly useful for programming assignments, as the marker can compile and run students' programs at the push of a button without having to start a language compiler separately.*

Keywords: *web-based assignment submission; electronic marking; online examinations*

Introduction

In response to the growing number of students entering universities, many teachers have turned to the use of electronic methods for receiving and processing students' assignments and examinations. It is widely accepted that students acquire new knowledge by engaging in a learning activity, not by passively absorbing information from textbooks and lectures (Ben-Ari, 2001) – so it is essential that students write their own programs in order to learn how to program and that teachers provide appropriate and timely feedback. Software Development 1 is a first-year course in programming at QUT in which students are required to complete and submit four or five small programming assignments and at least one medium-size assignment

in their semester of study. Under the old manual system, this represents a large cost to the faculty and consumes a disproportionate amount of time for the teachers.

The way students study at university is also changing. Most Australian universities have more than one campus and many offer courses outside the country. This trend also applies to QUT's Faculty of Information Technology which has been prominent in attracting overseas students and in offering courses at the satellite campus at Carseldine. Students quite rightly expect to be able to submit and retrieve assignments wherever they have the web, not having to be physically present at QUT.

Under the old manual system of assignment handling in FIT, students submit a hard copy of their assignments (sometimes with a floppy disk) at the Technical Services Section. The assignments are manually batched to markers and the state of each assignment is tracked by entering various transactions into the FIT Student Information System (FITSIS) until the assignment is ready to be picked up by the student. Apart from the workload overheads, the manual assignment system delivers material to markers in a physical form that is not conducive to the use of automated processing, such as plagiarism scanners and automated-marking aids. These types of activities are favoured by information technology teachers (e.g., Joy & Luck, 1995; Thomas & Paine, ref?).

In this context, the OAS was initiated under a QUT large teaching and learning grant (QUT 2000) to overhaul assignment-handling in FIT. There are five stages in using the OAS:

- Stage 0: Setting up assignment requirements
- Stage 1: Student submission
- Stage 2: Batching (distribution to staff)
- Stage 3: Marking
- Stage 4: Student retrieval

In the following sections of this paper, the current state of the OAS will be described with respect to each of the stages listed above; more detail will be provided about the electronic marking application (F²M²) used by lecturers and tutors, obviating the need for hard copies of assignments; and other projects related to the OAS and electronic assessment in general will be discussed. The paper concludes by detailing some of the benefits of the OAS over manual assignment processing schemes and by outlining future developments.

The contribution of this paper is that it describes a support system for teaching and learning that has been widely accepted by students and teachers in FIT. The OAS may well surpass the old manual system and soon become the accepted method for dealing with student assessment.

OAS

The OAS is a web-based assignment handling system intended to replace an existing manual system that is resource-intensive and no longer meets the requirements of many students and lecturers. This section describes the various stages of the OAS from the specification of an assignment to the student retrieval when marking is complete. The FITSIS database is the repository of all student results for the faculty and is therefore an integral part of this discussion. Figure 1 illustrates how users interact with the OAS and how the OAS relates to FITSIS. As shown in the diagram, there are separate interfaces for students and staff.



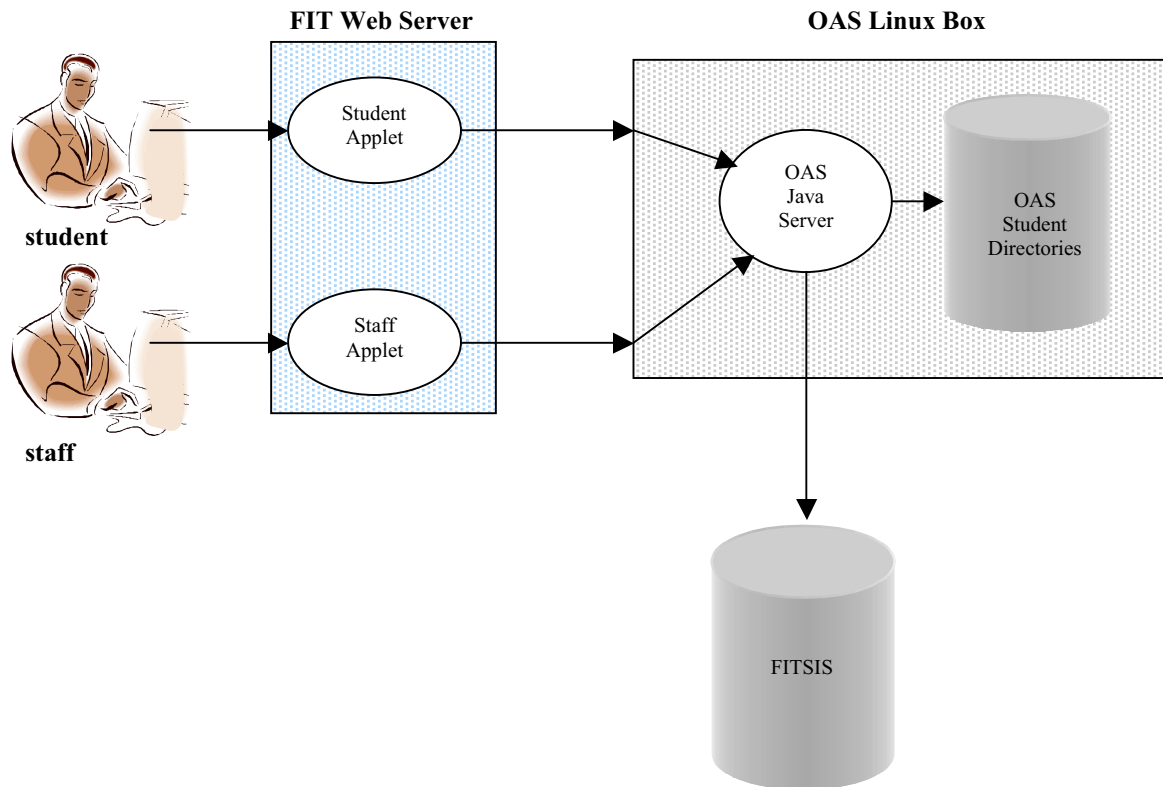


Figure 1: OAS and FITSIS

Stage 0: Setting up assignment requirements

The assessment requirements for all units are prescribed by the unit coordinator and entered into FITSIS at the start of each semester, noting which instruments are to use the OAS. For each of these, the unit coordinator specifies an assignment, establishes the marking criteria and decides what files are to be handed in by the students. The Technical Services Section receives this information and uses it to prepare the web pages for the subsequent stages (for students and teachers). This task will eventually be delegated to each individual lecturer when a web-based authoring tool has been developed to meet the needs of teachers.

Stage 1: Student submission

When a student has completed an assignment, they are able to submit the required files from any computer with a web browser and the Java 1.4 Runtime Environment. This is achieved by logging in, as shown in Figure 2. A student is authenticated through QUT access and then validated as being currently enrolled in FIT. The student's unit enrolment details and results are then displayed as shown in Figure 3. There is a high level of security built into FITSIS and the OAS using signed applets (Daconta, 1998) and discrete queries (rather than a logged session) so that there is little chance of malicious activity. The student next selects the assignment they intend to submit and is presented with a screen showing details of the required files. The appropriate files are selected from the student's own system and are then uploaded with the issue of a receipt number.

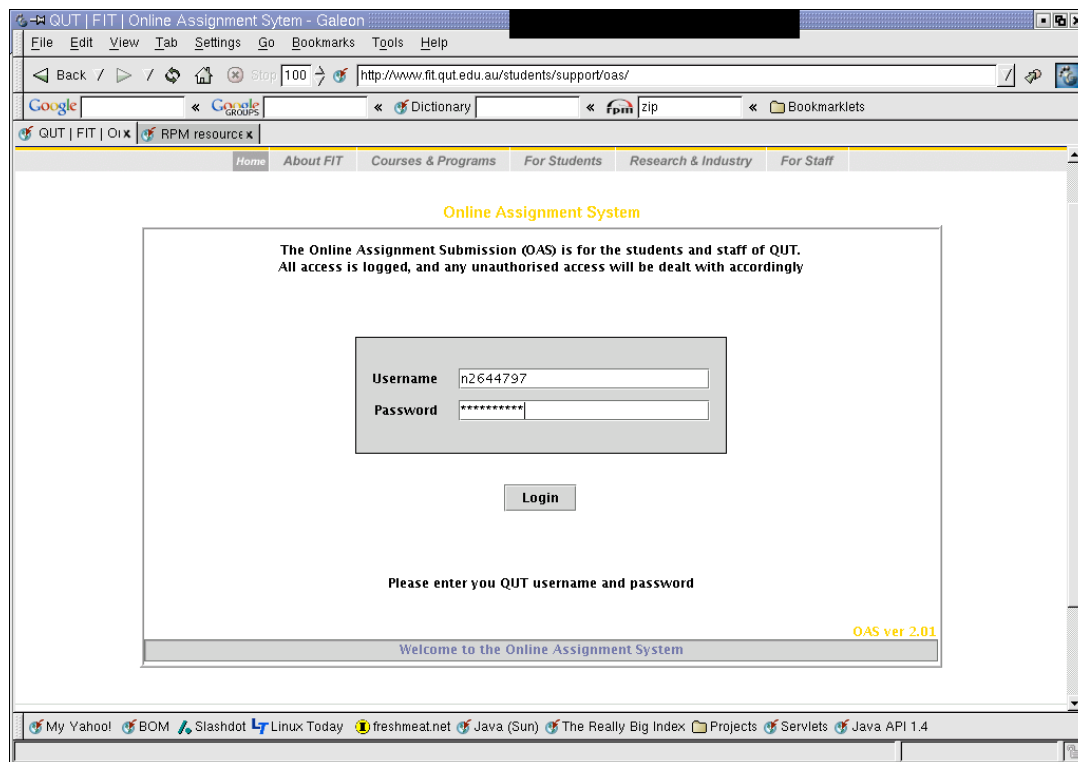


Figure 2: Login screen

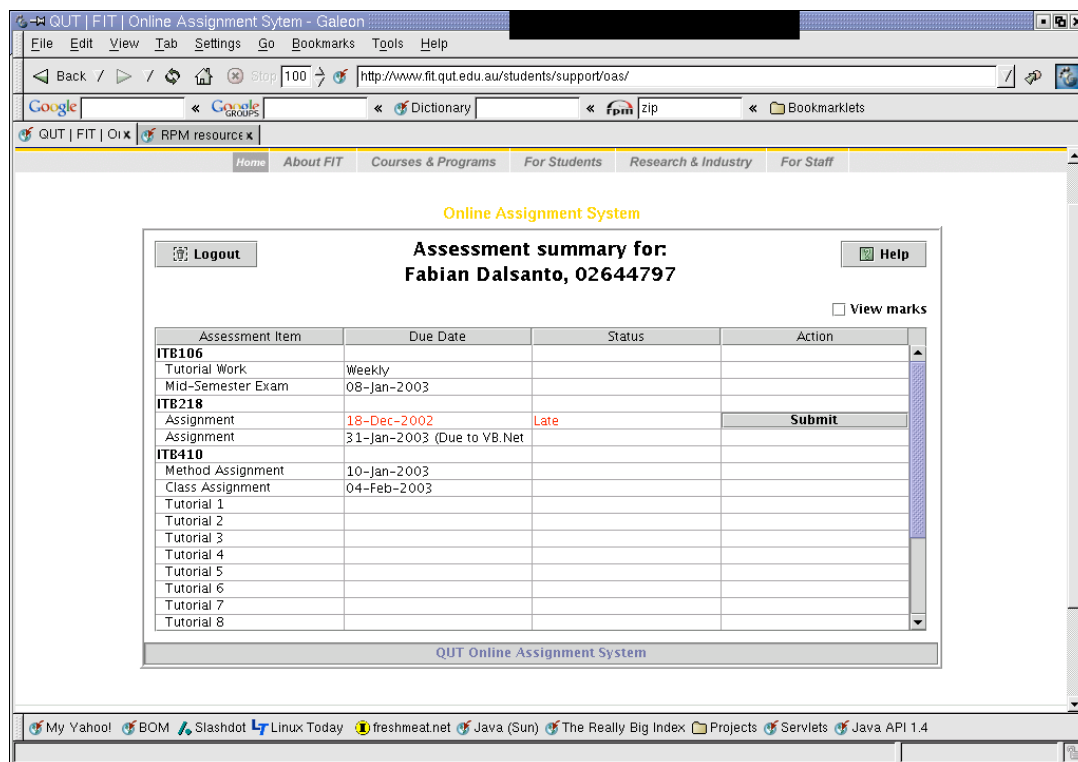


Figure 3: Submit screen

Stage 2: Batching

Under the old manual system, unit coordinators advise the FITSIS database administrator at the start of semester about the distribution of assignments to markers. The assignments are physically distributed and delivered, and the tracking details are entered into the database system. Under OAS, the unit coordinator logs on to FITSIS and enters batching details for their unit through a forms-based screen. Each time a marker loads the appropriate OAS web page to retrieve unmarked assignments, the OAS interrogates FITSIS for the batching details. In a sense, OAS is used to deliver the electronic files to markers in the same way a laboratory assistant delivers the manual assignments. A critical part of batching is the process whereby the OAS zips students' files, loads them on the OAS server and then preprocesses them into validated HTML version 4.01. In this format, the files are able to be loaded into the F²M² marking program while satisfying accessibility requirements.

Stage 3: Marking

There is some reluctance on the part of teachers to use the OAS because they have preconceived ideas about the difficulty of online marking. Many fear that they will not be able to access their batch of assignments easily and that they will have to print a hard copy of each student submission. These misconceptions may be based on early setbacks with the use of email, inadequate hardware and the lack of appropriate software support, in the form of GUI marking programs. The OAS offers a web interface for lecturers to download assignment files and includes the F²M² program for ease of marking. When teachers have finished marking an assignment, they upload the files back to the OAS server. FITSIS is notified by the OAS and students receive this information when they next log in. Figure 4 shows the screen presented to lecturers as they access their batch of assignments. F²M² will be described separately in Section 3. Because the OAS delivers files in electronic form, there is a range of options open to the marker apart from F²M². Teachers may choose to use passive viewers, to fill in electronic mark sheets, or even to print a hard copy. Several lecturers have successfully used automatic marking tools.

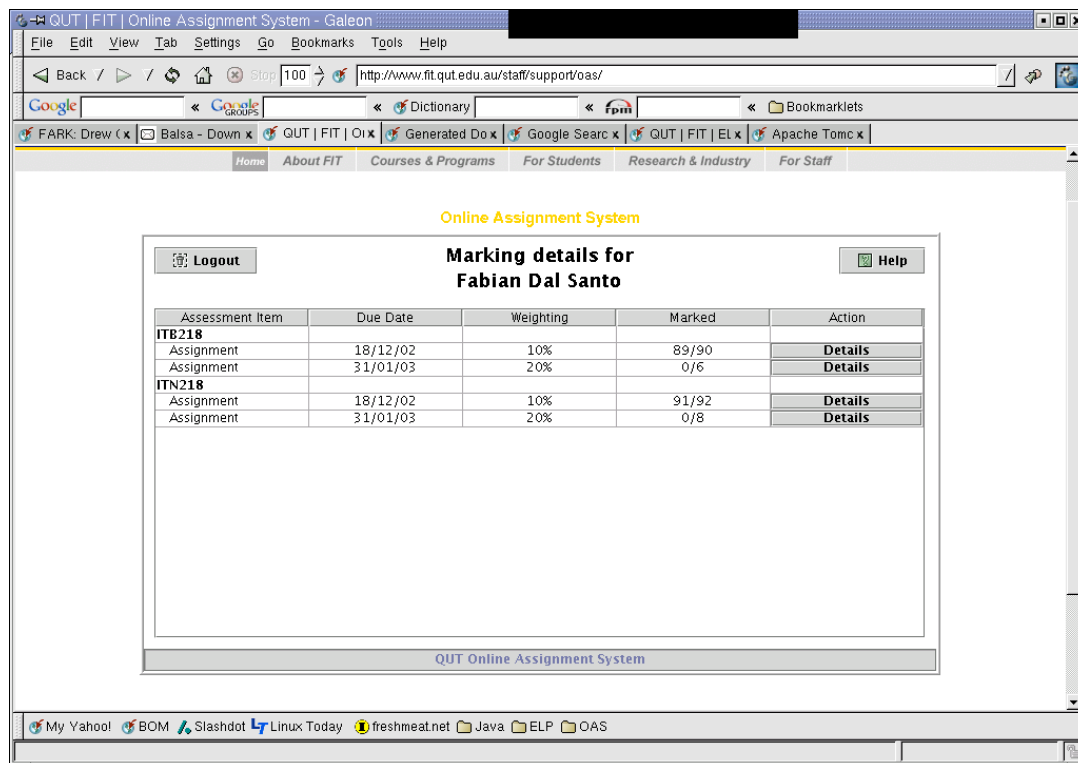


Figure 4: Lecturer batching screen

Stage 4: Student retrieval

When students access the OAS page shown in Figure 3, they can retrieve their marked assignments. Of course they can do this from home - they do not have to be on campus as long as they have web access. The files they download include their original submissions and HTML versions of the same, marked up with comments and hypertext links embedded in the documents by F²M². In most programming assignments, there is also a criteria sheet (in HTML format) which details the marks as prescribed in the assignment specification. This is a convenient process for students and they seem to like having the results delivered in this manner. The OAS is able to provide timely and useful feedback that cannot be matched by the manual system.

The OAS marking application - F²M²

It was realised early in the project that the most serious hurdle in convincing lecturers to use the OAS was in being able to provide a convenient facility to assist marking the electronic submissions. For example, teachers asked questions such as, "Will I have to print out each file?", "How do I put comments on the student's work?" or "How do I flick through the electronic pages?" F²M² is a Java application that teachers run to mark assignments. Students' submissions are preprocessed so that they are presented as HTML files inside the program. In Figure 5, it can be seen that there are 4 sections on the F²M² screen.

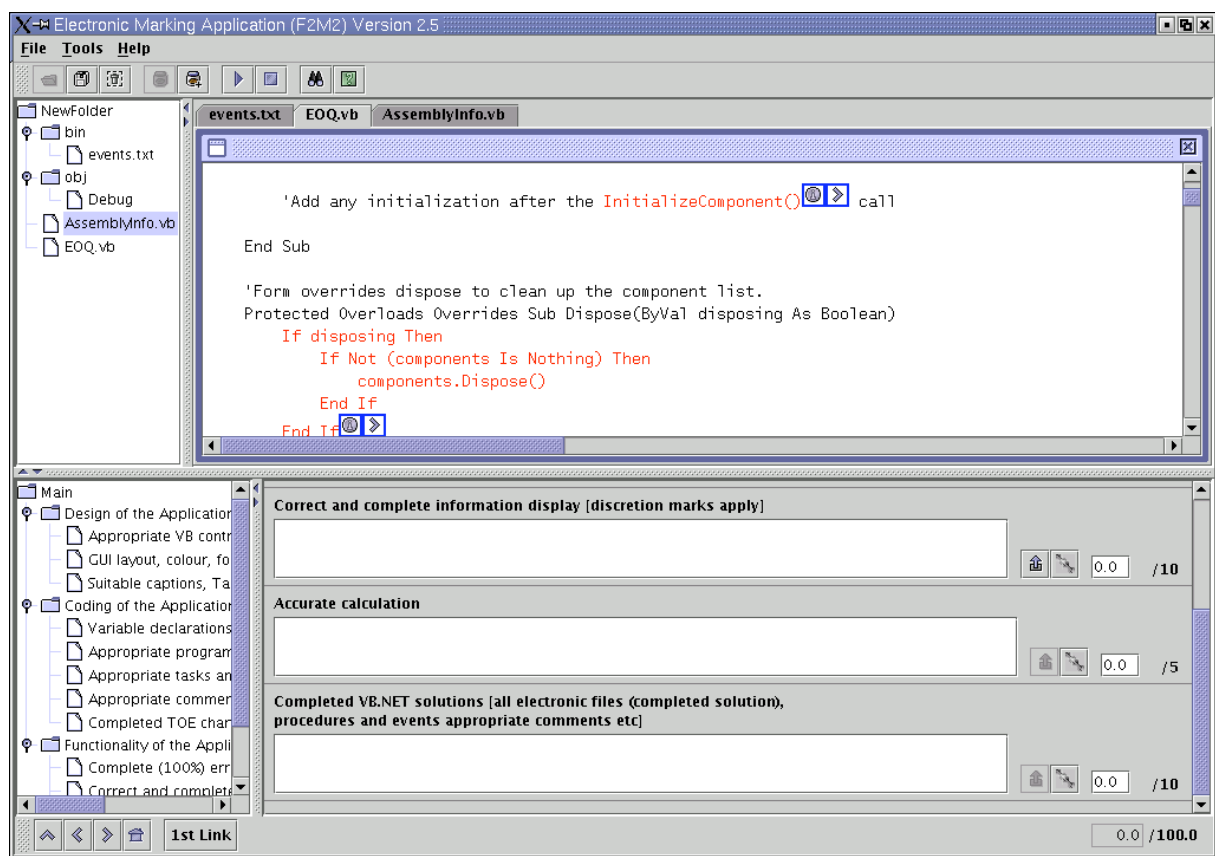


Figure 5: Marking program - F²M²

The top left pane is a directory tree used to navigate student files which may be loaded into the top right tab pane for viewing and mark-up. The bottom half of the F²M² screen is an HTML representation of the assignment criteria sheet, as specified by the lecturer prior to submission. The criteria sheet may be in the form of a check list of features required in the

assignment. For example, in a programming assignment, the teacher may be looking for certain syntactic structures in the solution or for testing and documentation standards. The bottom left pane is a tree allowing the marker to navigate the criteria sheet and the bottom right pane is where marks and comments are entered.

The marker cannot edit the student's work directly in the top right pane but they can compose comments in the bottom right pane's text areas and insert hypertext links into the student's work, as shown in Figure 5. This is a convenient way for students to examine their mistakes especially if the same error has recurred. F²M² allows comments to be linked together so that the student can search backwards and forwards throughout the document. On the top button bar, the arrow (fourth from the right) is very important for marking programming assignments. This button is linked to a command which allows markers to run the student's program. For example, Figure 5 shows a student's Visual Basic program which can be run from inside F²M² rather than having to exit and start the runtime system separately.

In the process of marking, sections of the student submission are annotated and linked to the relevant criteria and marks are allocated in the criteria sheet, automatically totalled and displayed at bottom right. The marking options are flexible; the unit coordinator can provide checkboxes to give or not give fixed part marks, or 'out of' fields to allow any mark up to some maximum; they can initially set marks to zero so that marks are given as the criteria are checked, or to initially give full marks so that marks need only be taken off where defects are found. While text fields can be provided in the lower right pane for individual comments on each criterion, often the criterion name, plus the bi-directional linkage to the occurrences in their code, makes it clear to the student without needing further annotation. Finally overall comments and discretionary marks may be provided, if the unit coordinator chooses to allow this.

All of this rich detail of marking scheme and feedback is available to the student. They receive what is essentially a read-only version of the F²M² environment in pure HTML so that they need no special viewer.

F²M² is one of the outstanding features of the OAS and most markers have commented on how much time they save (some suggest up to 30%) by not having to deal with bundles of paper assignments, not to mention the convenience of being able to access files and mark from home.

Related work

Electronic submission of assignments has been widely used by information technology teachers for many years, taking the form of email submission, transferring files to a specified directory or providing marker access to directories and files on a unix system. A brief search of the web still reveals many examples where these practices are continued. They have had varying degrees of success but the following comment taken from an email list is quite revealing. "I was hoping to use an email-based submission system, but there are still a few bugs. For now, you will need to put your file in a directory on RCS (also accessible from CS workstations)."

A simple, modern form of electronic assignment marking is to use MS Word which is popular and affords some advantages over manual methods. Feedback comments can be provided through hidden text, the *Comment* facility or by using different fonts and colours. Changes can be examined through the *Track Changes* facility. This may be satisfactory for written-

word assignments but is totally inappropriate for computer programs, which are presented in pure text form so that they can be compiled and run.

There are many examples of more sophisticated uses of electronic-based assessment reported in the literature, ranging from online questionnaire generators to comprehensive systems which may include submission of assignments and examinations, plagiarism checking and marking. The Hot Potatoes suite (Hot Potatoes, 2002) includes six applications, enabling the user to create interactive multiple-choice, short-answer, jumbled-sentence, crossword, matching/ordering and gap-fill exercises for the web. Again this does not provide the facilities needed for programming assignments.

One of the most successful examples of electronic courseware in Information Technology is Course Master (Course Master, 2000), a client-server system for delivering course-based programming. It provides functions for automatic assessment of students' work, administration of the resulting marks, solutions and course materials. It is also able to detect plagiarism in students' programs. A student is able to develop a program, submit it to the server for marking or evaluation and get instant feedback. CourseMaster is a complete re-implementation after 10 years of experience using the Ceilidh system (Foxley, 1999) in many institutions.

The BOSS system (Joy & Luck, 1998) is closer to OAS in terms of its intended use. This system allows programming assignments to be submitted online, tested against prescribed criteria and manually marked in a secure environment. Recently facilities have been added for electronic marking using 'electronic marksheets'. The project is driven by demands of increasing student numbers and is restricted to particular operating systems.

The OAS provides all of the functionality of BOSS and improves on this and other existing systems in a number of ways. Firstly, it was motivated to some extent by large class sizes but more so by the desire to provide more useful feedback to students, which has proved to be popular. Secondly, the OAS may be used across all units in a course of study and is not restricted to particular course materials like Course Master. Thirdly, OAS is web-based which represents a familiar and easy to use environment for students and is not constrained by certain operating systems. Finally, lecturers find that the F²M² marking program is easy to use and many have stated that it is time-saving compared to manual systems. F²M² is a Java application, which means that it can run on any system with a Java Virtual Machine installed. The electronic delivery of programming assignments also means that, before the files are submitted to F²M², they have already been virus checked, are already compiled and may easily have been run against a test suite. A further advantage of the OAS is that it is a single integrated system unlike BOSS which is a suite of related programs.

Conclusions and further work

The OAS has not replaced the faculty's manual system entirely. However it has proved very successful since being trialed in semester 2 of 2001. FIT's Technical Services Section took over the OAS at the completion of the project and entirely revised the code to produce a stable production environment from the prototype. Gradually more lecturers adopted the OAS in their units last year. Some of this is a result of student demand, while in other cases teachers have the convenience of introducing fully automatic marking schemes through the OAS. The OAS is amenable to all nature of assignments and is used in subjects such as C programming, database, Visual Basic, Java programming and written documents. Some of these use automatic marking and others use F²M².

The functionality of the OAS is being expanded further. In the second half of 2002, a subsystem was developed and successfully trialed to handle online, real-time examinations. This is known as the Online Examination System (OES) and is useful for facilitating laboratory practical exams. One shortcoming of the OAS is the lack of an authoring tool for lecturers to create their assignment specifications and marking criteria online in a forms-based web page. Such a facility would reduce the workload on the OAS application programmers who currently have to enter into the OAS the assignment details in the required format for each assignment in each subject.

Should the OES prove successful, final examinations could be handled in the same way as assignments. The marked scripts could be returned to the students for feedback, which would subsume the faculty's current exam review process. This is a tedious mechanism whereby students make appointments to see their scripts, office staff extract the scripts from the many unit bundles, supervise the student viewing, and collect the scripts afterward. Online exams could subsequently take on a more formative role by providing the students better feedback rather than being perceived as a dead-end with a purely summative result.

At various stages throughout the OAS project, feedback was sought about the system, in particular by applying web-based surveys using WOLF (Nulty, Bancroft, Brewster & Smith, 1998). The response from students has been consistently positive in recent times after initial technical difficulties. Students have on occasion expressed their dismay when, having been exposed to the OAS, they find it is not being used in a subsequent unit. The lecturers who are using the OAS are unreservedly enthusiastic and this suggests that the uptake will continue to increase. .

References

- Ben-Ari, M. (2001). Constructivism in computer science education. *Journal of Computers in Mathematics & Science Teaching*, 20(1), 24-73.
- Commonwealth of Australia. (2000, Thursday, 29th November 2001). Overseas student statistics. Retrieved from <http://www.dest.gov.au/highered/statistics/tables/students2000.htm>
- CourseMaster. (2000). School of Computer Science & IT, The University of Nottingham, UK. Retrieved from http://www.cs.nott.ac.uk/CourseMaster/cm_com/index.html
- Daconta, M. (Ed.), Monk, E. & Saganich, A. (1998). Java 1.2 and Javascript for C and C++ Programmers. John Wiley & Sons.
- Foxley, E. (1999). Ceilidh on the World Wide Web. Retrieved from <http://www.cs.nott.ac.uk/~ceilidh/>
- Hot Potatoes (2003). Retrieved from <http://web.uvic.ca/hrd/halfbaked/>
- Joy, M. & Luck, M. (1995). On-line submission and testing of programming assignments. In J. Hart (Ed.) *Innovations in computing teaching*. London: SEDA.
- Joy, M. & Luck, M. (1998). Effective electronic marking for on-line assessment. In *Proceedings of the 6th Annual Conference on the Teaching of Computing/3rd Annual Conference on Integrating Technology into Computer Science Education*, 134-138.
- Nulty, D., Bancroft, P., Brewster, S., & Smith, D. (1998). Web online feedback system. Retrieved from <http://www.fit.qut.edu.au/wolf/>
- QUT (2000) <http://www.talss.qut.edu.au/talss/stafftlgrants/gen/>
- QUT. (2003-2007). *Teaching and learning plan*. QUT. Retrieved from <http://www.qut.edu.au/chan/pr/planning/StratPlan/stratplan.htm>.
- Thomas, P. G., & Paine, C. B. (2000). *Tools for observing study behaviour*. Paper presented at the Psychology of Programming Interest Group, April, Cosenza, Italy.

Copyright 2003 © Peter Bancroft, John Hynd, Jim Reye & Fabian Dal Santo: The authors assign to HERDSA and educational non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to HERDSA to publish this document in full on the World Wide Web (prime sites and mirrors) on CD-ROM and in printed form within the HERDSA 2003 conference proceedings. Any other usage is prohibited without the express permission of the authors.