

# A scalable teaching and learning environment built on Apache

Lennie Au, David Morton, Sandra Silcot

Information Systems Development

Information Technology Services, The University of Melbourne, Australia.

Resources Automated For Teaching (WebRAFT) is a secure and reliable Web-based delivery system built on Apache to support online teaching and learning at The University of Melbourne. Driven by data from various administrative systems, Webraft automates the configuration of Apache to provide a staff-administered website for a University subject. Core features include a public and secure website for staff to publish course materials, free-text searching, usage statistics, and automatic updating of enrolled students. Other collaborative learning tools such as online conferencing and syndicate group publishing are optional. WebRaft provides an easy-to-use 'zero-administration' website where academic users can focus on content issues. In eighteen months, academic staff teaching in over 950 University subjects have chosen to use WebRaft. This session will discuss the design and rationale of the system.

Acknowledgements	2
1.0 Introduction	3
1.1 History of Webraft development.....	3
1.2 Design goals.....	3
1.3 Initial prototype development.....	4
1.4 University-wide rollout.....	4
2.0 Technical architecture	5
2.1 Overview.....	5
2.2 Authentication Process.....	7
2.2.1 Subject coordinators (staff/postgraduate).....	8
2.2.2 Enrolled students.....	8
2.3 Filesystem layout.....	8
2.4 Configuration of publishing authorisations (FTP).....	9
2.5 Apache Configuration: web access authorisations.....	10
2.5.1 Student and group level "prot" directories: .htaccess.....	10
2.6 Web Crossing forum authorisations.....	11
2.7 Web Crossing coordinator administration functions.....	11
2.8 User access "portal".....	12
3.0 Other popular features	13
3.1 Infoseek.....	13
3.2 CDROM snapshots.....	13
3.3 WebRaft redirection services.....	13
3.4 Web usage statistics.....	13
3.5 FTP access statistics.....	14
4.0 Academic usage in practice	14
5.0 System performance and reliability	14
7.0 Current developments	15
8.0 Conclusion	15
9.0 References	16

© 2000 Lennie Au, Sandra Silcot, David Morton.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the authors.

## **Acknowledgements**

### **Technical**

WebRaft was conceived and prototyped in 1998 by David Morton, Miri Goldenfarb, Jon Pearce and Michael Nott. The major contributors to its technical development and their roles are listed below. The developers are members of the University of Melbourne's Information Technology Services department. Work on WebRaft since 1998 has been effectively part-time, as the developers are all responsible for a variety of non-WebRaft related software and systems projects.

#### **David Morton (Systems Architect, Information Systems Development)**

Conception; system architecture; project leadership.

#### **Miri Goldenfarb (Director, Information Systems Development)**

Project leadership; user liaison; funding.

#### **Sandra Silcot (Web Systems Developer, Information Systems Development)**

Web Crossing integration; Perl scripts/cgi's.

#### **Lennie Au (Web Systems Developer, Information Systems Development)**

Perl programming, Apache configuration.

#### **Terrence Miao (Web Systems Developer, Information Systems Development)**

Perl and PHP programming.

#### **Rob Sturrock (Systems Programmer, Systems and Networks)**

System architecture (Solaris); Account management (ARS); systems programming.

#### **Chris Stroud (Systems Programmer, Systems and Networks)**

WU-FTPD implementation; Radius/PAM; systems consulting.

#### **Bob Ankerlin (Systems Support, Systems and Networks)**

CAP modifications for PAM.

#### **Andrew Humphrey (Web Systems Developer, Information Systems Development)**

Perl scripts; prototype development. (1998)

### **Academic**

Most importantly, many academic staff at The University of Melbourne have generously contributed their ideas and feedback about the system. This has enabled the further development of Webraft. These people include: Jon Pearce, Michael Nott, Glyn Rimmington, Graham Moore, Paul Gruba and Liz Sonenberg.

### **Support**

Thanks to Rae Langfield and Joy Kastannias for providing on-going support to users required throughout the project.

### **Editorial**

Thank you to Pam Andrew for her valuable suggestions and proof-reading of this paper.

## 1.0 Introduction

The University of Melbourne was the first University in Victoria, Australia, being established in 1854. It currently has over 35,000 undergraduate and postgraduate students, offering over 3,000 undergraduate level subjects across eleven faculties. The faculties are: Architecture, Building and Planning; Arts; Economics and Commerce; Education; Engineering; Land and Food Resources; Law; Medicine, Dentistry and Health Science; Music; Science and Veterinary Science. In addition to its large scale undergraduate teaching activities, the University is renowned for its research and higher degree programs, particularly in Medical disciplines where it attracts large amounts of funding from external sources. It is a founding member of Universitas 21, an international group of about 18 universities, currently engaged in collaborative efforts to explore the potentials of online delivery in the global education market.

**Web Resources Automated for Teaching (WebRAFT)** is a secure and reliable web-based teaching and learning environment for on-line teaching. It is managed and located on a central computer at Information Technology Services (ITS), where the Information Technology (IT) infrastructure is maintained by the ITS staff. The system is integrated with the University Student System (Merlin) and central staff/student email accounts, the Account Registration System (ARS).

The system is built with an open architecture in mind. It is expected that the system is capable of integrating other open standard software and "best of breed" specialist software (e.g. conferencing). The published materials served on the system are simple http, and not bound to any proprietary products or backend databases.

### 1.1 History of Webraft development

The University of Melbourne has shown interest in embracing flexible learning and online course delivery for a considerable period of time. Some departments have conducted distance education courses for well over 10 years. While faculties and departments funded the local development of online (usually web-based) teaching resources, the administration of many distributed back-end server systems and tasks such as the manual update of student enrolments proved tedious and expensive. This was especially the case for those academics without access to local Information Technology (IT) support staff.

This set of environmental problems was the genesis of WebRAFT.

The central Information Systems Development group (ISD) in the University of Melbourne's Information Technology Services department (ITS) is responsible for corporate administrative software and web-based systems. The central administrative systems include student administration (MERLIN) from which subject enrolments are obtained, and the personnel administration system (GENESYS) from which staff data can be obtained. The Systems and Networks Group (SNG) provide free student and staff email POP accounts to all members of the university community on a central ITS machine. All central machine accounts are administered through a complex set of shell scripts developed by SNG - the Account Registration System (ARS).

This suite of existing IT infrastructure formed the core elements of a solution. Senior developers in ISD realised that the available data and systems could be integrated and software could be developed to automatically configure websites on a central machine, using existing email accounts for authentication via RADIUS. In essence, WebRAFT is the "glue" which bolts these components together, resulting in the automated maintenance of required user accounts, creation of website directory structures, central authentications via existing email accounts, and Apache/FTP level authorisations for web access and publishing.

### 1.2 Design goals

The specifications for WebRAFT were drafted with a small group of academics who had already been involved in a subject which employed web-based teaching methods in a sophisticated manner: <http://webraft.its.unimelb.edu.au/615280/pub/>

The initial objective was to automatically manage a central, rather than a locally run web-server and to automate the administration tasks of student enrolments in a single prototype subject - 615280.

Functionally, the prototype included a number of features specified by the academics - public and private (restricted) websites for staff, students, and "syndicate" groups of students; integrated

email, online conferencing using the Web Crossing package; free-text searching and automatically generated student lists published online. Staff and students needed to be able to publish content from their primarily Macintosh desktops.

One final design goal was to build an architecture that can support and integrate Open Standard software. Webraft was to set up an environment that allows the academic community to evaluate specialised products with the possibility of integrating them with the system.

### 1.3 Initial prototype development

A prototype system was developed in semester two, 1998 using subject, 615280. This was quite quickly achieved by developing the prototype on the same machine as housed the student email, a DEC Alpha OSF 4 machine. That meant several authorisation and authentication problems were already solved.

This prototype involved manually setting up the directory structure for a single subject; creating the required staff user accounts for the subject coordinators; modifying ARS to create website directories with appropriate permissions for enrolled students; generating the Apache configuration files by hand and creating the apache staff/student group entries by simply "grepping" through /etc/group. CAP - Columbia Access Protocol - was installed to facilitate desktop "drag-and-drop" uploading of users' files, in addition to FTP which was already in place on the student email machine. The final requirement for a working prototype was maintaining student membership in syndicate groups. This was initially managed by using a spare field in the student administration system to assign a student to a syndicate. ARS was modified to access this data in addition to subject enrolments.

Within four to six weeks, a rough prototype system was in place and functioning.

It was clear early in the prototype's operations that the academic staff in 615280 were very enthusiastic about the benefits WebRaft had brought to their teaching. It had completely relieved the academic staff from time consuming web and system administrative tasks particularly in the area of user management and had provided a very reliable central web environment for staff and student web publishing. They, their students and syndicates, could publish their requirements in either public (world accessible) or private (restricted to coordinators and enrolled students) without having to lift a finger to administer a machine, accounts, take backups and the myriad of other technical support issues involved in running a web server.

The success of this "proof of concept" prototype enabled the Manager of ISD to obtain funding for a multi-subject prototype in Semester 1 1999, to be followed by a University wide rollout for delivery by Semester 2 1999.

The requirements for usage of the system by any University subject would be:

- the subject-code and student enrolment data must exist in Merlin;
- subject coordinators require an active University (central) staff email account;
- students must activate their University ITS email account.

The same features which had been built for 615280 would then be freely available to all university subjects, if the academics wished to use them.

### 1.4 University-wide rollout

A Solaris machine<sup>1</sup> for WebRaft was purchased in 1999 (Enterprise 450, 1 GB ram, 50GB raided HD). It was planned to deliver 6-7 subjects during first semester 1999 during a "bedding down" period.

Webraft now evolved from a single subject (hack) prototype, to an engineered generic system for large numbers of academic staff, students and subjects with these objectives :

- to satisfy a wide-scale common need for websites to deliver teaching/course materials to students;

---

<sup>1</sup> Solaris was chosen, rather than OSF (the prototype system's OS) because it supported the Unix Plug-in Authentication Module (PAM) which was not available under OSF.

- to protect intellectual property;
- for content-providers to be free to choose and use what tools they like to make content - content neutrality;
- to provide a framework or toolkit for online "collaborative" teaching and learning: websites;
- a search engine and online conferencing for information exchange;
- to be simple to use for staff and students;
- to be fast, reliable and have a scalable infrastructure;
- to efficiently leverage existing ITS infrastructure and systems (Email, ARS, Merlin).

The rest of 1998 and early 1999 was spent building the generalised solution. That is, scripts were written to automate what had been manually done as a "one-off" for 615280. Modifications were made to the conferencing system to:

- handle more than one online forum (one per subject - optional) with appropriate permissions for staff to moderate and students to participate within each forum;
- provide an administrative interface for coordinators to manage their own webrast feature set independently of ITS e.g. add/remove coordinators; assign students to groups.<sup>2</sup>

A key development required for the mass rollout was the development of PAM-RADIUS modules. The Apache RADIUS module was used to provide Apache authentication to the student and staff email machines. PAM was used to enable the FTP server (WU-FTPD) to also authenticate staff and student uploads without requiring passwords to be maintained on the WebRaft machine itself. Accounts are still required on the WebRaft machine to allow file system access to maintain their web sites. Note that the need for students to publish, and the need for syndicate group publishing, became optional features.

One unanticipated difficulty was the different Access Control List (ACL) semantics between the OSF and Solaris operating systems, and in particular the way these interacted with CAP. ACL's provided a means to overcome the limitations of the Unix permissions system. The latter was unable to deal with complex access requirements for staff to have full write access over student/group sites, and for syndicate group members (students) to have write access over the syndicate website. ACL's were put in place and automatically applied by ARS to get the desired results. In brief, this worked much better under OSF than under Solaris because OSF handled ACL inheritance more effectively than Solaris. It was not until well in 1999 that a fully stable solution was finalised under Solaris.

After being overwhelmed by the demand from other academic teaching staff during the first few weeks of semester one 1999, the system was released for general usage much earlier than had been originally planned. No announcements or marketing had taken place yet by the end of July 1999, there were 305 subjects enabled on the system. In early September 2000, over 950 subjects were using WebRAFT as a web delivery system. It appeared that WebRAFT's essentially simple feature set had met a genuine and pressing need. In addition, the rapid takeup rate would appear to be at least in part related to an internal organisational target set by the Vice Chancellor that 20%, and later 50%, of all academics were to become "multimedia enabled". Having a website was sufficient criteria to deem this target to be met.

## 2.0 Technical architecture

### 2.1 Overview

Webrast is tightly integrated with the University Administrative systems and relies on the account information provided from ARS. This data is extracted nightly. The high level overview of how the

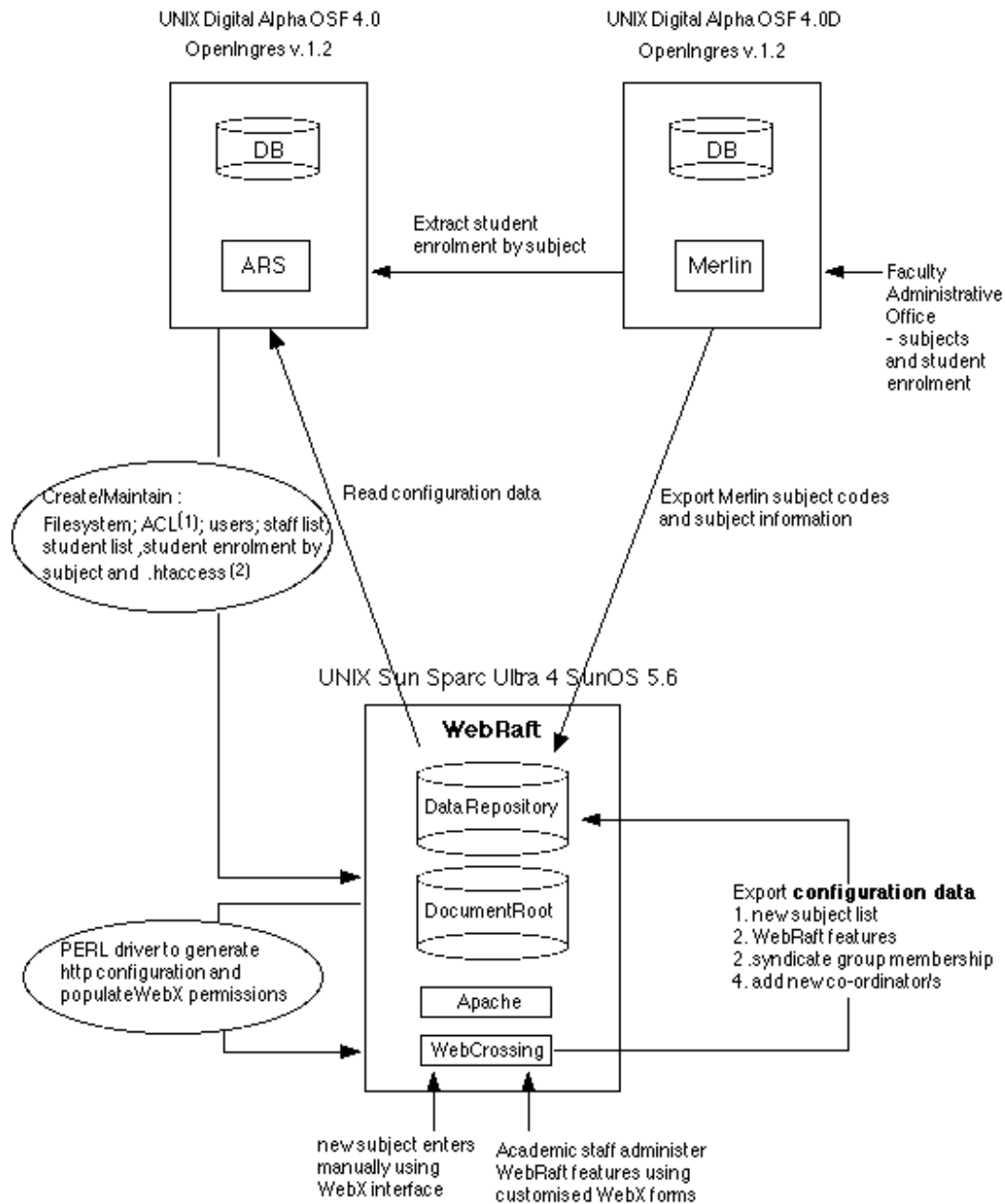
---

<sup>2</sup> The initial solution of using a field in Merlin for student syndicate group membership had not worked well. Academic staff did not have access to this system, and found it inconvenient to be constantly asking administrative staff to update this data. In response, it was decided to use the customisation features of Web Crossing to provide a simple web-based interface so that coordinators could maintain this data themselves.

whole system works is shown in Figure. 1.

Most of Webraft is implemented in Perl and UNIX shell scripts for text-based manipulation and system operations respectively.

**Figure 1: High level system architecture**



Notes to chart:

- (1) See sample ACL listing in Section 2.4.
- (2) See sample .htaccess file in Section 2.5.1.

The system process flow is completely automated, with the exception of new subjects.

Academics submit a request for a new subject via a web form. This is received and manually entered into WebRAFT via WebCrossing, an Object Oriented database PHP-like language. Once a subject has been created, subject coordinators can maintain their own feature sets, add/remove other coordinators, assign students to groups, manage the scheduling of CD-ROM image creation jobs, and manage (host) their own online forums. This information is exported from WebCrossing as the configuration data for ARS to process.

Using the configuration data, ARS extracts the relevant enrolment information from Merlin using SQL queries and performs user setup, unix filesystem and access control list (ACL) setup, generate enrolment

details, staff userlist and student userlist. These are text-based delimiter files.

WebRAFT also uses these new feeds to populate the WebCrossing permissions in the discussion forum.

The subject list and the new student list for each subject are generated overnight. The httpd.conf file and the http group file is populated from the data feed from the configuration data. The http group file is required for web authentication to the protected subject and group areas. This is discussed in Section 3.5.

A typical format in the http group file is shown below :

```
s615280-admin:jonmp
s615280:user1 user2 ... userNN
s615280-0001: user1 user2 user8
```

The first field is the groupname, followed by a colon, followed by the list of usernames belonged to the group.

## 2.2 Authentication Process

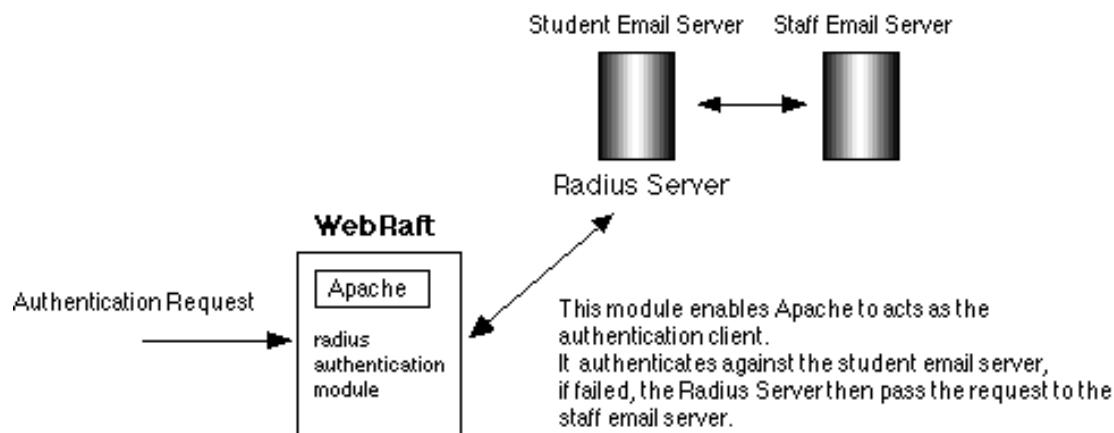
All protection in Webrft is simple http basic authentication. This is considered good enough security for the nature of the system.

Users authenticate themselves with their central email username and password.

The server-side software (wu-ftpd and CAP) have been modified to use a Unix Plug-in Authentication Module (PAM) module which queries Radius authentication servers on the Student Email System and the Staff Email System.

Any web requests to browse protected area will invoke users to authenticate via the "Basic http authentication", mod\_auth in Apache which has been customised to use a customised version of mod\_auth\_radius. Figure 2 shows the Radius authentication process against the student and staff email server.

Figure 2: Radius Authentication Process



The Apache RADIUS authentication module allows the Apache web-server to become a RADIUS client for authentication. The module is configured at compilation of Apache.

A feature of the authentication module is that it hands back a time-stamped MD5 encrypted cookie to the client on a successful authentication. This cookie is then accepted a subsequent access as sufficient validation for access. This avoids the need to a new radius query for each subsequent HTTP request. The cookie has a configurable expiry time, after it can no longer be used to authenticate. Upon expiry of the cookie, the user will be required to re-authenticate.

These Radius authentication configuration are added to the httpd.conf file for the WebRAFT site.

```
# # Radius authentication configuration
#
AddRadiusAuth localhostname:port <shared-secret>
AddRadiusCookieValid 60
```

This specified the Radius server, the shared secret key and the Cookie expires in 60 minutes.

### 2.2.1 Subject coordinators (staff/postgraduate)

Staff users **must authenticate using an ITS staff email account**, otherwise staff will not have access to the protected areas on the system.

Each subject must have at least one coordinator.

Subject coordinators have:

- read/write access over the entire subject website, this includes the student and group areas;
- "Host" privileges over online forums;
- access to Webraft administrative functions for defining the WebRaft feature set. Figure 4 shows the subject administration page available to co-ordinator/s of the subject.

### 2.2.2 Enrolled students

Students must have activated their ITS student email accounts and be enrolled in the WebRaft subject for the current semester.

Student functions depend on the WebRAFT functionalities defined by coordinators:

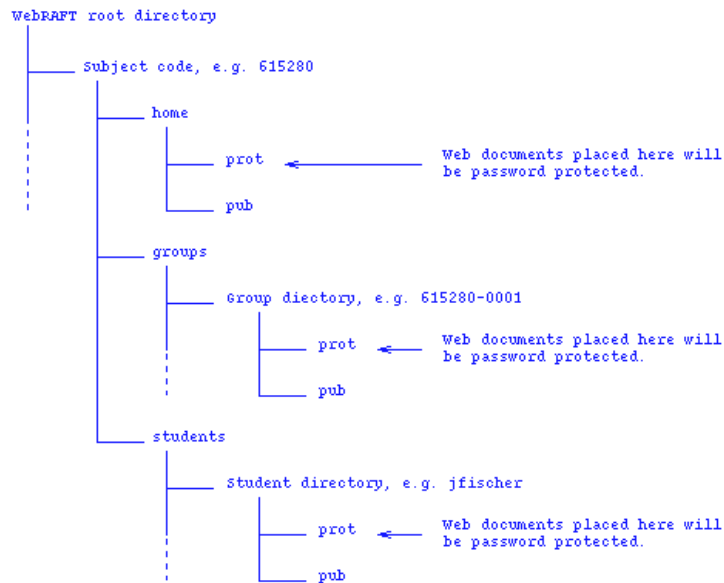
- at a minimum, they can access protected pages in the subject's home area;
- if an online forum is enabled, they can have participant access to the subject's Web Crossing online forum;
- if student publishing is enabled, they get their own subject-specific website for publishing public or protected pages.
- Student "protected" pages can be accessed by the individual student *and* coordinators.
- if group publishing is enabled and a student has been assigned to be a member of that group, they have read/write access to the Group's website.

## 2.3 Filesystem layout

At its heart, WebRaft simply provides websites for users with an FTP system overlaid on top to enable users to publish. Directory structures and file permissions therefore are at the core of the system.

All the files for a WebRAFT subject are stored under a single directory. The top-level directory is named using the university subject code as it exists in Merlin. The **home** directory is used to store course materials and information related to the subject. The **group** directory is used for student group projects. The **student** directory is used to store student individual assignment or project for the subject.



**Figure 3: WebRAFT directory layout**

## 2.4 Configuration of publishing authorisations (FTP)

The typical ACL/permissions setup of the home directory and its subdirectory "latchkey" directories above is shown below:

```

# file: home/
# owner: root
# group: s615280-adm user::rwx user:cwis:r-x
#effective:r-x user:http:r-x
#effective:r-x group::r-x
#effective:r-x group:s615280-adm:r-x
#effective:r-x mask:rwx other:---

```

```

# file: home/pub
# owner: root
# group: s615280-adm user::rwx user:cwis:rwx
#effective:rwx user:http:r-x
#effective:r-x group::rwx
#effective:rwx group:s615280-adm:rwx
#effective:rwx mask:rwx other:---
default:user::rwx
default:user:cwis:rwx
default:user:http:r-x
default:group::rwx
default:group:s615280-adm:rwx
default:mask:rwx
default:other:---

```

```
# file: home/prot/
# owner: root
# group: s615280-adm user::rwx user:cwis:rwx
#effective:rwx user:http:r-x
#effective:r-x group::rwx
#effective:rwx group:s615280-adm:rwx
#effective:rwx mask:rwx other:---
default:user::rwx
default:user:cwis:rwx
default:user:http:r-x
default:group::rwx
default:group:s615280-adm:rwx
default:mask:rwx
default:other:---
```

This setup is purely for FTP access to the filesystem when users require to upload files into their home area. The owner of the subject can **execute** into the pub and prot area to perform file upload. It is not allowed to write immediately under the home directory.

Similarly ACLs are applied to student and group directories so that members of the "-adm" (coordinators) group have full read/write access over these areas.

The Apache level configuration of web access to this structure is explained below in Section 3.5.

## 2.5 Apache Configuration: web access authorisations

A PERL script runs each morning which checks that ARS has completed successfully and has provided the necessary student enrolment data for WebRaft subjects. If so this script executes a series of other scripts which handle the Apache configuration work. When these jobs have finished successfully, the httpd.conf file is updated with template based entries for every subject in the configuration data exported from Webraft. Each subject's entry in Apache's httpd.conf file looks like this:

```
# Directory setup for subject s615280
Alias /615280/students/ /localdir/webraft/docs/615280/ students/
Alias /615280/groups /localdir/webraft/docs/615280/groups/
Alias /615280/ /localdir/webraft/docs/615280/home/

<Directory /localdir/webraft/docs/615280/home/prot>
AuthRadiusAuthoritative on
AuthRadiusCookieValid 60
AuthGroupFile /localdir/conf/group
AuthType Basic AuthName "WebRaft Protected Area"
<Limit PUT GET POST>
allow from all
require group s615280 s615280-adm
require user t615280
</Limit>
</Directory>
```

The <Directory> directive specifies the document root of the subject. It tells the Apache web server to use Radius authentication and requires valid user in group s615280 or s615280-adm or user t615280, if failed, access is denied.

These scripts also generate the Apache "groups" file which lists the user names which are in the enrolled student group (e.g. s615280) and in the subject coordinators group (e.g. s615280-adm). These files are generated independently of the contents of the system level /etc/group file. This is because only the need for web publishing requires the need for a system level account (for FTP/CAP access) and web publishing by students and syndicates is an optional system feature.

### 2.5.1 Student and group level "prot" directories: .htaccess

For web access to protected syndicate group files, and for student access to their individual protected files, authorisations are managed by using .htaccess files. The above method was considered too unwieldy to apply to individual students and syndicate groups. The .htaccess files are generated by the ARS system using template files and are written into the "prot" directory. An example of a syndicate group .htaccess is as follows:

```
AuthRadiusAuthoritative on
AuthRadiusCookieValid 60
AuthGroupFile /serversdir/conf/group
```

```

AuthType Basic
AuthName "WebRaft Protected Area"
<Limit PUT GET POST>
allow from all
require group s615280-0001 s615280-adm
</Limit>

```

This tells Apache to use Radius authentication and to set the cookie to expire in 60 minutes. The user realm to which they are authenticating is "WebRaft Protected Area". The above example code requires that `mod_auth_radius` return a valid user name who is a member of group "s615280-0001" or "s615280-adm", otherwise access is denied.

## 2.6 Web Crossing forum authorisations

WebCrossing is a specialised commercial online web-based conferencing system. The academics in 615280 insisted that it be used. They had been using and evaluating forum software for several years and had decided upon using Web Crossing (WebX).

Part of the process flow mentioned above is to internally populate WebX's forum access control list structures. These allow individuals or groups to have various levels of access to "locations" in its Object Oriented database. Top level "folders" with WebX ACLs applied were used to implement individual subject forums.

WebRAFT uses the student enrolment feeds to populate WebCrossing student groups named s615280-students. These groups perform the same function for WebX that the web access Apache groups s615280 perform for Apache.

For example, subject 615280 will have the following ACL for its discussion forum which allows students to participate in discussions, staff (listed as individuals and the technical support group) to moderate/edit discussions, with no access to others or guests:

```

.ee6c53c
<access guestPermit=00000000 otherPermit=00000000>
<host>
  jonmp
  cwis-users
<participant>
  s615280-students
</access>

```

These ACL configuration files are imported into WebCrossing. The ".ee6c53c" is the internal reference id WebCrossing uses for this location in the OODatabase (which is a WebRaft subject's optional online forum). Users "jonmp" and group "cwis-users" are the "hosts" and student group "s615280-students" are "participants" in this forum.

The group names in WebCrossing is populated in the same manner. The import file has the similar format shown below.

```

<group name="cwis-users">
  <user name="user1">
  <user name="user2">
</group>

```

## 2.7 Web Crossing coordinator administration functions

WebX provides a customisable Object Oriented database and a proprietary PHP-like "template" language<sup>3</sup>. It is simple to build web-based forms which update the database. It's database can be exported in an sgml-like format, and all of its configuration data can be imported as simple text files such as those shown above.

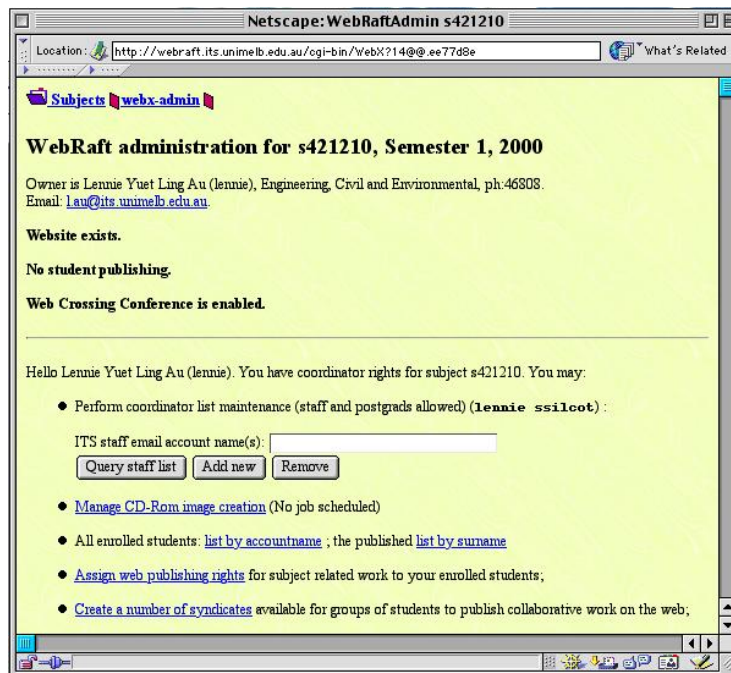
Because of these features, and because forums meant that WebX had to have all user authorisation data in any case, it was decided to write the web-based functions for coordinators to administer their WebRaft subjects using Web Crossing. This was done by providing a special top level location, named "webx-admin", which only coordinators can access. Within that branch of the WebX database, the top level structure is replicated i.e. each subject has a "forum" which has been customised to provide access to administrative functions such as:

<sup>3</sup> This is being slowly replaced by Mozilla Server-side java script. The latter enables a developer to build custom objects and have them stored in the WebX database.

- add/remove coordinators
- activate an online forum
- activate student publishing
- activate syndicate publishing
- change the number of syndicate groups
- assign students to syndicate groups

Therefore staff are able to self-administer the WebRaft features in use for their subject.

**Figure 4: Subject Administration Page**



## 2.8 User access "portal"

The **WebRAFT portal** is the starting point for authorised users to access the system. When a user has successfully authenticated into the portal, the system gets the access privilege about the user based on the user account name and it displays hypertext links to all functions and the features which the user is authorised to access.

**Figure 5: WebRAFT Portal**

Search ?

**Welcome to the WebRaft Portal**

You are logged in as **lennie**. These resources are available:

Real user: **lennie is superuser.**

User:

Action:

---

**General resources**

- Apply for a [test account](#) (lets you test the students' view of your subject) **New!**
- [Webraft news](#) for staff (last modified Jul 21 15:48)
- [WebRaft home page](#) (includes the [Users Manual](#) and [help with access problems](#).)
- As a member of staff, you may complete this form to [create a new webraft subject](#)
- [The Coordinator's forum](#) (a Web Crossing conference for Webraft coordinators)
- [A Guided Tour of Web Crossing](#) (live tutorial on using the online forums)
- [Web Crossing](#) (the top level of the online forums)
- Public areas of [all WebRaft subjects](#)

**421-210 Environmental Engineering (sem 1)**

- Subject home pages: [public](#) and [protected](#). The [list of students](#).
- Edit and participate in the [421-210 online discussion forum](#).
- Staff coordinator access: [s421210-administration](#) and FTP logs: [home http-access-log](#)
- [Web access statistics](#)

**421-376 Environmental Engineering Design 1 (sem 2)**

- Subject home pages: [public](#) and [protected](#). The [list of students](#).
- [Student websites](#).
- [Syndicate websites \[10\]](#).

### 3.0 Other popular features

#### 3.1 Infoseek

Each subject can point to a page which invokes the University's Infoseek server. This provides a subject-specific website free-text search facility.

#### 3.2 CDRom snapshots

One of the most popular features of the system is the ability for staff to schedule offline jobs which create a CDRom ISO9660 image file of their websites, and optionally of student/group publishing areas. These images can then downloaded to a local machine with a CD burner. So long as users have used relative links, these "snapshots" provide a fully functional offline browsing capacity. This has proven to be useful for backup and demonstration usages and in minimising student internet access charges.

Many staff who use student/group publishing in assessment use CDRom images as their assessment source. In many ways this is more effective than online submission tools - once the CDimage has been created at the scheduled time, there is no way students/groups can modify their work.

#### 3.3 WebRaft redirection services

A simple CGI has been developed which enables other web-services to "piggy-back" off WebRaft. By pointing at this CGI from a protected page and passing the target url, the latter is the target of a redirect and username and subjectcode data is passed to it. This feature is actively used by the University's MultiMedia Unit to provide an authorised entry point to specialist interactive media on their own servers.

This is a good example of how "openness" is built into WebRaft's design and features.

#### 3.4 Web usage statistics

A further example of WebRaft's open design philosophy is to allow the plug-in of other tools. One case is the use of pwebstats to generate subject web access statistics.

### 3.5 FTP access statistics

The WU-FTPD logs are used to provide coordinators access (browse and download) to the FTP logs. This feature has proven very useful in helping academics assess the veracity of students claims of "Oh but I uploaded it on such and date and time".

### 4.0 Academic usage in practice

One of the most important issues of a web delivery system is the flexibility of its functions which enable academics to design and to develop resource-based learning materials for their course according to their pedagogical approach. While the user evaluation is beyond the scope of this paper, the positive feedback on the use of WebRaft as a tool for teaching and learning can be provided as evidence. Some of the benefits reported by academics and students who have used the system are:

- Reading materials maybe posted on WebRaft which provides convenient access for students, and reduces or avoids the cost of hardcopy distribution.
- The use of a monthly electronic diary published by students in their protected area. This information is accessible only by staff and the individual student. This aspect encourages reflective learning - students report on their own progress in learning. Academic teaching staff are able to evaluate his/her teaching methods through qualitative analysis of these reports.
- Collaborate learning occurs in online discussion, and group syndicate publishing projects. Also the University has many overseas students who can keep in touch with campus-based students.
- Easy integration of a wide range of web-based functions through hyperlinks. This allows the creation of a useful repertoire of teaching and learning environments.
- Less reliance is needed on traditional "time and place"-based lecture/tutorials for large groups of students.
- More postgraduate students have been able to participate in learning activities irrespective of their physical distance from campus.
- Students acquired generically computing skills as a by-product e.g. FTP, html page construction.

### 5.0 System performance and reliability

During the period of 1 January 2000 to 11 September 2000. The uptime of Webraft in the Service Window from 8am-6pm Monday to Friday is 99.98%. The overall uptime is 99.26%. WebCrossing has an uptime of 98.83% in the Service Window and an overall of 98.84%. These figures have not excluded scheduled outages for system maintenance.

## 6.0 Problems and Limitations

Webraft does not in any way provide a "complete" solution for online teaching and learning. It does a limited job extremely well: it provides a cheap, efficient, secure, reliable website environment for subject course materials, student publishing and online discussions. It provides the technical infrastructure which is only a small part of the story for online teaching and learning system.

It is by no means, an authoring environment. It is content neutral; it simply acts as an environment that can deliver any web contents.

Security issues and operating system quirks are the source of some limitations in WebRAFT:

1. No Common Gateway Interface (CGI) scripts for any subject;
2. No "Guest" access as the system uses Radius authentication against the university ITS's staff and student email machines. This means only university staff or student who have activated their email account can access to the protected area.
3. CD images job for the subject website can not be greater than 650 MB, otherwise, manual intervention is required.
4. Under the Solaris Operating System, users can not be in more than 16 groups including their primary group. This means that a member of staff cannot act as a coordinator in more than 15 subjects. The

work around has been to force a manual ACL to be applied to the relevant subject websites at the system level. A group is created manually which contains a list of co-ordinators' username. ARS then applies this ACL to the required websites.

5. This aspect of the system is most unsatisfactory. We are exploring options such as upping the Solaris kernel limit to 32 or applying individual, rather than group based, ACLs..

## 7.0 Current developments

Webraft development plans are currently targeted at some functional gaps, such as:

1. Allow guest access

This will allow visitors or academics from other universities to participate in the subject forum and browse the protected area of the subject/s they have been authorised to access. This involves the development of a PHP subsystem using a LDAP server to record and authenticate visitors. Perldap, Radiusperl and mod\_perl will be used for the new Apache authentication module.

2. Convert WebCrossing discussion forums from SGML-like data to HTML for archival purposes. Currently, discussion forums are rendered live by the WebCrossing cgi. The internal data is impossible to access off-line. Hence the need for an export/conversion tool. This will be optionally integrated into the CDRom job creation script. The forum Document Type Definition (DTD) has been defined and an omnimark program has been developed at ISD to process a subject in April 2000, it is scheduled for further development in year 2001.
3. Integrating a "best of breed" online test/quiz system.

## 8.0 Conclusion

WebRaft is designed to be a scalable web delivery system. Its primary objective is to provide basic WWW services - a "tabula rasa" - to staff and students of official University subjects. WebRaft manages the administrative tasks very well and satisfies many academic requirements. These include protecting intellectual property, content-neutrality, and providing online collaborative tools such as forums and group work. It is simple to use for staff and students.

In no way does it pretend to be a complete solution for online teaching. Staff and students, who provide the content which is published on WebRaft, do the really hard part. Its success does however demonstrate the continued relevance of large-scale centrally provided IT infrastructure in an era of distributed personal computing.

## 9.0 References

Morton. D, Silcot. S, *Using WebRAFT*, Information Technology Services, The University of Melbourne, Australia, May 1999.

Morton. D, Silcot. S, *WebRAFT : Information for Academic*, Information Technology Services, The University of Melbourne, Australia, September 1999.

Apache Software Foundation, *Server Documentation - Apache 1.3 User's Guide*, Apache Software Foundation, 2000. <<http://www.apache.org/doc>>

Spencer. H, Apache HTTP Server Project, *Apache\_1.3.4-mod\_auth\_radius source*, Apache Software Foundation, 1994.

Pearce. J, Goldenfarb. M, *Merlin Meets WebRAFT: Online Subject Management, Delivery and Assessment*, Presentation at the Multimedia Seminar, The University of Melbourne, June 1999.

Moore. G, *Problem Based Learning with WebRAFT*, Presentation at the Multimedia Seminar, The University of Melbourne, Australia, April 2000.

Gruba. P, *Integrating communication skills into project-based teaching and learning*, Presentation at the Multimedia Seminar, The University of Melbourne, Australia, April 2000.

Web Crossing Inc., *Online Technical Documentation*, Web Crossing Inc. USA.1999.