

## Loughborough University Institutional Repository

---

# *Building a co-operative, evolutionary intranet: experience in a University department*

This item was submitted to Loughborough University's Institutional Repository by the/an author.

**Citation:** STONE, R.G., 2001. Building a co-operative, evolutionary intranet: experience in a University department. IN: Dawson et al (eds.) Software Quality Management 9, British Computer Society, Loughborough University, April 2001.

**Additional Information:**

- This is a conference paper.

**Metadata Record:** <https://dspace.lboro.ac.uk/2134/2633>

**Publisher:** © British Computer Society

Please cite the published version.

This item was submitted to Loughborough's Institutional Repository by the author and is made available under the following Creative Commons Licence conditions.



For the full text of this licence, please go to:  
<http://creativecommons.org/licenses/by-nc-nd/2.5/>



# **Building a Co-operative, Evolutionary Intranet: Experience in a University Department**

Dr Roger G Stone  
Department of Computer Science  
Loughborough University, Loughborough, UK

## **Abstract**

Experience with building an intranet for staff & students of a Computer Science department is reported. The knowledge and experience of the people involved, and the requirements and budget available makes this task rather different from building a Corporate Intranet or Portal. The intranet has evolved to encompass three sites and is still growing in terms of overall size and use. The intranet is generally regarded as a success both by staff and students. This success is attributed more to policy and attitude than to technical excellence.

## **1 Background**

The intranet which is the subject of this paper is in its 3rd year. Prior to the arrival of the intranet the department administration was paper-oriented with just a few Microsoft Works files available from a file server. This service did not reach all staff because of the mix of desktop platforms (Macintosh, PC & Unix). As regards provision for the students one member of staff was making good progress with a system called TICSS (The Interactive Course Support System). Apart from electronic delivery of lecture material, TICSS had ambitions to carry a significant portion of the assessment load, at least for some subjects (similar to CEILIDH [1]). Unfortunately that member of staff who was the creator and main driving force left fairly abruptly. His departure aggravated an already severe situation with respect to staff-student ratio and student numbers were still rising. At this time a change of administrative duties produced a new Programme Director for the Computer Science Programme.

## **2 Initial Stages**

There was significant interest in the department regarding distance learning. One lecturer produced a non-web system called Courseware for delivering Powerpoint and Word documents to the students. A small group spent some time considering the merits of trying to offer an MSc by distance learning which caused us to look at systems such as WebCT [2]. However what we wanted was a straightforward delivery system which did not claim screen area for its own

navigation and functions. Some members of staff had produced material for the second, web-based version of TICSS and thus had some experience of producing web pages for teaching. This led to the idea of using a plain web-server to deliver the material. This provided the starting point for the student service. At this time the key concept was to create a "One-Stop Shop" for students of the department where they could find any available teaching or administrative information. In order to encourage other members of staff to come on board with minimal effort, no specific file format or presentation format was specified. Each member of staff was allocated a directory/folder and encouraged to build module folders within their personal folders. Once one of their modules was introduced onto the home-page index they were allowed to populate the module directory with word-processing files, an indexed list of powerpoint presentations, a tree of HTML documents, or whatever they felt was appropriate. By physically visiting members of staff in their offices, discussing their potential involvement, showing what was possible and setting up their machines the idea took hold. The concept was later endorsed by HoD, but that was in part because an opportunity was seen to reduce our extravagant photocopying bill.

So we had a simple web-system on a borrowed machine (An Apache server on a Sun workstation). The next step set the scene for the staff administrative side of the service. We were proud of our personal tutor system where staff recorded on paper the progress of their tutees during weekly visits. The difficulty was that if a question was raised in the department in relation to a student we knew that information was available but it was hard to access. The process of finding out who the personal tutor was, then discovering that they were unavailable because of teaching or away on a conference was very frustrating. Our technical support staff advised that it was possible to have two virtual servers on the one physical machine. So we embarked on the electronic version of the student progress record system, providing separate services for staff and students,

### **3 Progress Record System**

The idea was to record electronically the information that was being collected on the paper system. It was envisaged that the students would have one interface via their one-stop shop web-server and the staff would have another via their administration server. Because there was no Unix database readily available at the time, this system was coded in Perl using text-files for the data. The system was relatively successful. The students updated their progress records in advance of their personal tutorials and the staff were able to add comments. However the first version was too prescriptive. The students were forced to fill in every field of a form (5 days attendance + 3 entries for each of 6 modules ) and to fill in each week in order. The second version removed these irritations.

The technical problems that had to be sorted out to produce this service were primarily security issues. The student one-stop shop service was called co-teach and the staff system called co-admin. The co-teach web-service was set up to run without authentication in our labs (where the students had to login in order

to begin using a machine) but require a username/password combination from elsewhere on or off campus. This we felt kept our teaching material private to our own students. Because the web server was not privy to any original login, the students had to supply a username/password combination at the point of updating a progress record. The co-admin web-service for staff was restricted to the staff subnet and in addition required a username/password combination.

## **4 Four Web-Sites**

At this stage the department was maintaining 3 web-sites, co-teach, co-admin and the departments contribution to the University web-site ([www.lboro.ac.uk/deparments/co](http://www.lboro.ac.uk/deparments/co)) . It soon became clear that there was a category of information that could not conveniently be handled by any of these sites. The department wanted as much material as possible to be available to prospective students via a web service. Also the professional placement tutor wanted prospective employers to be able to view the existing information to tempt them to become involved. However the university directives on the content of the departmental web-site and the restrictions on the uploading of new material made it impractical to use that site. Finally a third virtual web-server was added (co-public) to carry unrestricted information.

## **5 The Second Year**

Rather than 'lose' all the data entered into the system in the first (academic) year it seemed best to reprogram many of the original scripts to offer *this* year's data as default but allow selection of the data for a different year via '*earlier*' and '*later*' buttons. Thus began the gradual appearance of historical data grouped into folders, one per session. In fact certain data concerning final year students, their supervisors and project titles that seemed to be regularly required for references etc was deliberately added historically for the three years before the system started. The historical aspect held out the promise of being very useful given the unreliability of the memory and office filing systems of our academics.

## **6 Intranet for Student Users – “co-teach”**

### **6.1 Facilities**

As far as the teaching material is concerned, certain members of staff have eagerly grasped the opportunity of using colour and animation in their teaching material. The department purchased a portable data projector and a pattern was established of presenting lectures using the data projector with the students knowing that they could not only re-run the presentations later from the web but also download copies onto their own machines. For this purpose some modules with large volumes of material offer in addition a zipped version of the module materials. Apart from Powerpoint presentations the opportunity was taken by some to produce animation by programming, either by Java applets or by server-side JavaScript. Some of the results have been quite elaborate. For example a module which was required to give an introduction to C programming used a web-page with a type-in area for a program and a Run button. A simple program

was read off the form, converted into JavaScript (by a script written JavaScript) and then executed by the browser. This meant that the students could start programming with a simple type-and-run system. This deferred the time when they had to deal with the complexities of a modern project oriented language system until they had understood the basics of programming. A second example showing how elaborate the animations became is provided by a module dealing with language systems & compilers. Various web-page toys were constructed to demonstrate top-down and bottom-up parsing with grammars supplied by the students. The toys showed the parsing steps and the growth of the parse-tree.

## **6.2 Evaluation**

The site has been a success with on-course students, prospective students and staff. The web provision for a module regularly appears as a positive comment on module feedback forms. The students complain if a module is not represented on the site. The usage statistics for the site suggest 40-80 pages requests per day per student. The site provides an attractive selling point for prospective students. The site has already proved helpful during TQA visits even though these are primarily looking at paper based evidence. The site has helped members of staff 'see' what others are teaching and provides a much better basis for the handover of a module from one member of staff to another.

When the students have been asked directly for comments on the site, they have been very complimentary with very few complaints being received. A few suggestions have been made along the lines of better graphics and more consistent provision from module to module. At the moment we are resisting any such action as it goes directly against the original principles upon which the site was based.

## **7 Intranet for Staff – “co-admin”**

### **7.1 Facilities**

Staff have privileged access to student lists, student photos, student progress records, coursework schedule, records of departmental meeting and various directories. There is a departmental diary which gives information about departmental events and the availability of individual members of staff. Much of the information on the co-admin site is directly editable using a web-based editor by any member of staff and the community ownership of the data is an important aspect of the appeal of the site.

### **7.2 Evaluation**

As far as administration is concerned the staff were very pleased to have electronic searchable lists of students, to be able to see colour pictures on screen of individual students (or groups). They find the depository of departmental records (e.g. minutes of meetings) relieves them from the vagaries of their own filing systems. It has been interesting to discover that even when the information is known to be inaccurate or not up to date (as is often the case with student information at the beginning of the academic year) the system is still used and is

not regarded as a failure. The usage statistics for the site suggest 10-20 pages requests per weekday per staff member.

## **8 Intranet for the General Public – “co-public”**

### **8.1 Facilities**

The site contains information that may be of use to students not yet on course who would not be able to access co-teach. It also contains information that the department wishes to make generally available which is ineligible or unsuitable for inclusion on the department's section of the university web-site. Information provided by the Professional Studies Tutor is maintained here for students and companies involved with student placement in their 'year out'.

### **8.2 Evaluation**

This site plays a small but crucial part in the overall scheme. It is the smallest and least active. The majority of users may not be aware that they are using the site as most accesses are made using links from the other two sites within the intranet.

## **9 The Intranet as a Whole**

### **9.1 Facilities**

Some of the features of the intranet spread over two or more of the servers. The progress record has already been mentioned. Another example of the way related information is distributed over the sites is provided by Programme Regulations. Regulations are placed on co-public to allow access by prospective students but with links from co-teach and co-admin. However co-teach offers in addition a script which draws programme structure diagrams thus giving extra information to on-course students. In contrast co-admin hosts information for staff on regulation updates and changes from year to year. The home pages for the sites do not attempt to establish an imposing graphics presence. Rather they are minimal text-based categorised lists of links. For example the bulk of the co-teach home page is made up of links to the teaching materials for 60 different modules.

### **9.2 Evaluation**

The combination of sites which we refer to as our intranet is well regarded. An indication of this is the continued growth in size and increasing use as evidenced by the logs.

## **10 Innovative Aspects & Reasons for Success**

The system was not designed to be delivered in a hierarchical fashion by a web-master or team. Rather a co-operative view of joint ownership has been fostered. So taking as an example the departmental telephone list, the bottom of the web-page provides a link to an editor where any member of staff can update any part of the information at any time. This requires a level of trust and responsibility but sets a tone of community ownership of the data. It is prudent to keep

multiple backup copies of data that is subject to community update. The data is at higher risk from many novice updaters than it would be from a single expert. In fact the current editor script makes and preserves a copy of the initial data prior to every update.

Starting in the second year of our system the university has gradually brought online its own campus-wide initiatives for student learning materials (“learn.lboro.ac.uk”) and staff intranet (“internal.lboro.ac.uk”). We have not felt able to embrace either of these initiatives because they are seen to be inferior to our own provision. Members of staff felt that the learn server’s approach of making the material for all modules available to all students was unsatisfactory. The internal server is seen as potentially valuable to research groups which span departments. However there is no facility for server-side scripting so it could not replace our co-admin service.

## **11 The Future**

The server has recently been upgraded and now offers PHP and SQL services and it is the intention that the system will be gradually reworked to use PHP scripting to replace the Perl and an SQL database to replace the text files. Some testing has been done on a different server that suggests that the conversion may be a little tedious but there are no real problems.

In the future it may be possible to change the behaviour of university central administration and be able to share data more successfully. One frustration felt by the department is in supplying the university central administration with data that belongs to us and then not being able to access it freely ourselves. For example we provide module descriptions, including booklists, module registrations per student and assessment marks. This data, once sent, is either not available to us electronically or, if it is, then only via web-pages. This has led to some rather inelegant scripting. In order to create a “Reading List” for a particular year group of students on a particular programme, a script accesses all the web pages containing relevant module descriptions, scans them for keywords and the reading lists extracted from amongst the HTML.

Various student projects are underway to extend or improve various aspects. For example there is one to create a portal concept with the intention that staff & possibly students can personalise their interfaces. Other examples are an electronic logging system for managing professional training placements and for managing student final year projects.

## **12 Conclusions**

The intranet was launched using a borrowed computer with minimal facilities (e.g. no database). It has evolved and grown and been re-hosted, but is still



essentially the same. It is a well-liked, well-used system offering far more than the more recent university-wide provision. The key to its success is thought to be the attitude to ownership and format of the information. Staff are free to put whatever teaching material they wish onto a module site in whatever format they feel is appropriate. The administration site maintains a sense of community ownership of the data.

## Appendix – Size and Usage Statistics

### Size of the sites

co-teach	2.6 GigaBytes
co-admin	19 MegaBytes
co-public	140 MegaBytes

### Scripted Facilities on co-admin

Facility	no of scripts	total lines
Booklist	3	151
Committee (includes SSC)	3	263
Coursework schedule	3	502
Diary	7	1073
Directories	4	495
Edit File	1	105
External Appointments	3	200
Forum	5	759
Keypad	1	59
Module Database	3	149
Progress record	16	2313
Special Needs	1	76
Teaching Allocation	1	88
<b>Totals</b>	<b>51</b>	<b>6233</b>

### Usage Statistics

	Session:	1999-2000	1999-2000	2000-2001
	Semester:	1	2	1
co-teach	average # pages/day	26466	28547	47533
	# distinct files requested	18623	29881	20830
co-admin	average # pages/weekday	590	442	853
co-public	average # pages/day	752	712	755

### Web References

1. CEILIDH (Computer Environment for Interactive Learning in Diverse Habitats)  
<http://www.cs.nott.ac.uk/~ceilidh/>
2. WebCT <http://www.webct.com>