



## **The use of hypertext and the world-wide-web in teaching software engineering**

L. Spence, R.I. Ferguson, N. Parrington,  
J.B. Thompson

*University of Sunderland, School of Computing  
and Information Systems, Priestman Building,  
Green Terrace, Sunderland, Tyne/Wear,  
SR1 3SD, UK*

### **Abstract**

This paper describes the development of hypertext based open-learning and assessment materials for the teaching of software engineering. The system, created in the School of Computing and Information Systems at the University of Sunderland is available via the World-Wide-Web (the Web) and permits the delivery of the teaching material over the global academic computer network, the Internet (the Net). The system permits the incorporation of CASE tools into the hypertext environment to facilitate practical S.E. tutorial activity

Current practices in teaching Software Engineering at Sunderland are reviewed along with the problems they present such as high staff/student ratios and low contact time between staff and individual students. The teaching materials and modes of delivery are discussed and the rationale for the development of the new system presented

“State-of-the-art” networking technology has been used for the preparation and delivery of the new material. A review of the enabling technology and reasons for its adoption is given as well as a description of the finished delivery system and an account its development

Whilst the original teaching material was appropriate for traditional methods of delivery, it had to be radically altered for the new system. The structure of the new system, its hypertext nature built-in tutorial exercises and self assessment modules are presented.



## 94 Software Engineering in Higher Education

Several features of the Web make it an ideal medium for the support of “distance learning”. Features such as electronic submission of work and delivery of results can also be used to counteract the effects of reduced contact time. The system has been built to permit the addition of automatic marking for at least some of the exercises. Whilst the feature is not yet available, we give a proposal for its implementation.

One of the advanced features of the system is its ability to deliver CASE tool usage over the Web. The reasons for doing this and the methods underlying it are discussed.

The system “goes live” in the 94/95 academic year. The use of the open-learning paradigm will lead to greater quality in the delivering of material to students. The use of global hypertext allows the easy construction of material for student centred learning whilst on line CASE tools and tutorial exercises together with automatic marking be beneficial for courses run in distance learning modes

## 1 Current Practice

Within the School of Computing and Information Systems at the University of Sunderland, a module entitled Software Engineering is taught to classes of students from other numerate but non-computing disciplines. The content of the S.E. module aims to give students an understanding of the software engineering approach to software development, and an awareness of methods, techniques and tools available. On completing the course, students should possess an understanding of the processes involved in software system construction and testing as well as awareness and experience of automated tools which span the life-cycle.

In the past, the subject has been taught using the traditional approach of a series of lectures with group based tutorial sessions, involving case studies reinforced with practical tutorial exercises in an attempt to “tie together” the individual lecture topics where possible. Student numbers and time constraints have led to a situation where it is almost impossible to deliver the complete course in sufficient detail in the time prescribed. In such circumstances, descriptions become brief and possibly even inadequate.

The above is compounded by the fact that the prior knowledge of students varies greatly with, at the extreme, those who have no practical experience and those who are well versed in many of the techniques of S.E.

This has put increased pressure upon the lecturer to maintain a constant learning experience for all students.

One solution to this problem is the “open-learning” paradigm described by Birch<sup>1</sup>, Dorrell<sup>2</sup> and others, whereby lecture and tutorial material can be combined to form open-learning material which is made available in an interactive

format using the latest delivery technology. When combined in an environment with CASE tools to support practical activity a powerful and flexible learning medium can be created.

## 2 The Software Engineering hypertext support tool

To counter the above problems, a hypertext system has been developed which is based on existing teaching material but containing a great deal more information. By using the Software Engineering hypertext support tool, the student can look again at the material, but he/she will also have access to supplementary material which is available via hypertext links from the top-level material's sub-headings. These links lead to a more detailed description of the topic than could be presented with the original material which the students can read at their own pace. Pages of material (examples of which are in Figure 1) can be printed out for future reference. This activity can take place either in a tutorial session or at the student's convenience. Since the CASE tools required to undertake tutorials are incorporated within the tutorial material itself, the student can operate within a single unified environment.

The particular hypertext system chosen is the World Wide Web (WWW). This has the distinct advantage that material can easily be delivered over the academic computer network, the Internet.

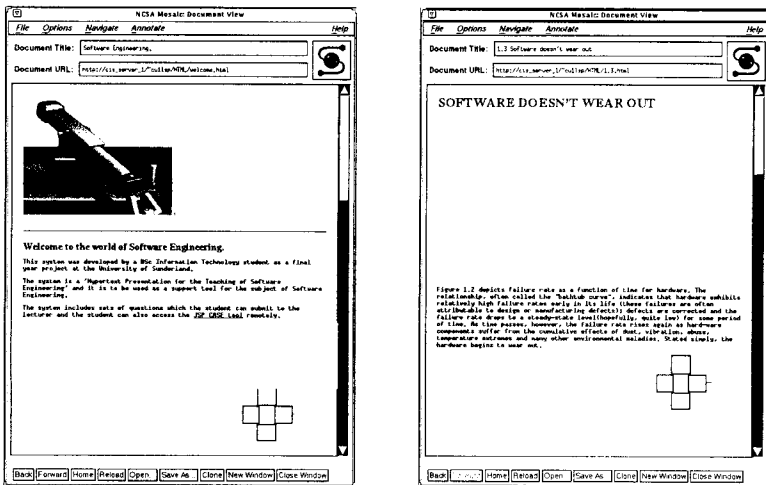


Figure. 1 : Example screens from the tool.



## 3 Background to the tools/techniques used

### 3.1 What is Hypertext?

The simplest way to describe hypertext is to contrast it with traditional text. All traditional text, whether in printed form or computer files, is sequential. That is, there is a single linear sequence defining the order in which the text is to be read. First, you read page one, then page two, page three and so on.

Hypertext is non-sequential, it is structured in “information-space” rather than by media constraints. This means that there is no single order which determines the sequence in which the text is to be read. Hypertext presents several different options, and the individual reader determines which of them to follow at the time of reading the text. As a consequence there are a number of alternatives for readers to explore rather than a single stream of information.

Hypertext documents have “active areas” in them. A student can click on an active area to go to another page, display a notice or close a window. An active area can be a range of text or an image. Active areas are used to navigate between items of related information. Links are set up between different parts of a document (or between parts of two different hypertext documents) so that the student can skip around from one place to another using the paths provided. This allows the students to follow their own interests and explore a subject non sequentially.

### 3.2 The Internet

The Internet (the net) is a global (mostly academic) computer network. It now has an estimated 20 million users, but until the recent advent of navigating tools such as the World Wide Web, presentation and retrieval of information over the net was a relatively complex task.

### 3.3 WWW

The World-Wide Web (WWW) (Whitcroft<sup>7</sup>) is an advanced tool for retrieving and displaying information from the Internet. It is based around the hypertext paradigm but enhancing that paradigm to the extent that the destination of the links may be a remote computer giving what has come to be called global hypertext.

The WWW world consists of documents, and links. Indexes are special documents which, rather than being read, may be searched. The result of such a search is another (“virtual”) document containing links to the documents found.

### 3.4 Mosaic

The Mosaic software is the client end of the Webs client/server architecture.



Mosaic retrieves and displays the documents associated with hypertext links. The hypertext documents viewed with Mosaic are written in hypertext mark-up language (HTML) and can be prepared in an ordinary word-processor

## 4 The Software Engineering Hypertext Tool

The system developed at Sunderland contains all of the lecture notes and tutorial exercises that were previously delivered in traditional manner as well as much supplementary material. Tutorial exercises have been provided on each topic that is covered in the course. Answer forms are provided on screen as an integral part of the material. Once a given form has been completed, it is delivered to the lecturer via email for marking, and ultimately returned to the student. Many of the exercises are based on a multiple choice answering scheme the automatic marking of these exercises is the subject of a current project.

The system will be going live during the current academic year. For the first year of operation at least, the system will be used as a backup to traditional modes of delivery. The system may also be utilised in a purely open-learning manner.

## 5 Technical Issues

### 5.1 Creation of the pages

Since the design of the material was to be based on existing material, this phase of development was of minimal difficulty. The actual construction of the Web pages was carried out "in the raw" with the HTML being entered directly in via a simple text editor. This approach has proved to be excessively time consuming. With the advent of tools, such as that described by Rousseau<sup>6</sup> to aid in the preparation of HTML documents, we envisage a considerable easing in the implementation phase with corresponding reduction in development time.

### 5.2 Integrated CASE tools.

Part of the specification for a hypermedia based tutoring system for this subject was a requirement for the provision of CASE tools as an integral part of the teaching environment. One of the aims of the course is to expose students to real CASE tools which can be used both in the completion of exercises and in

## 98 Software Engineering in Higher Education

the development of real software.

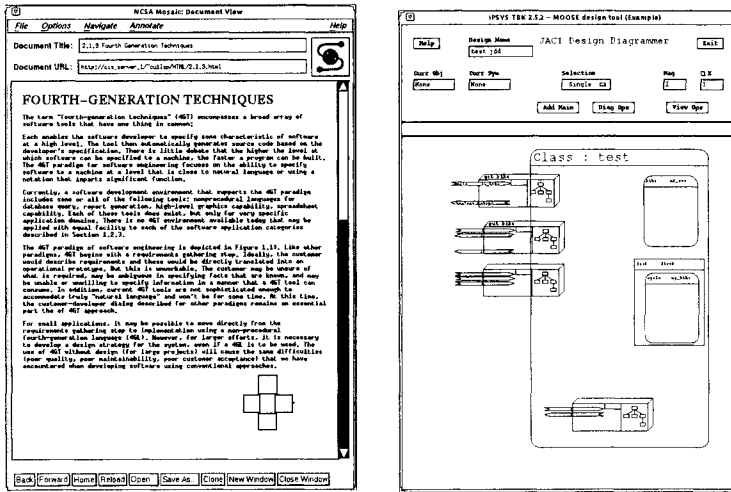


Figure. 2 : CASE tools integrated into the system

By making use of the Web's capabilities, we are able to allow students to remotely access CASE tools held centrally on our Web server. The tools have been developed by the CASE tool research group at the university. Off-site access has been tested via university researchers working at an international research laboratory in Geneva. The capacity of our current network links are severely stretched by this facility so whilst we consider the feasibility of serving CASE tools remotely to be proved, there are practical obstacles to implementing this in a full scale manner.

### 5.3 CASE tool research

Researchers from Sunderland have developed a set of case tools supporting various phases of the Software Development Lifecycle (e.g. Ferguson<sup>4</sup>) (including some supporting Object-Oriented Methods, described by Ferguson<sup>3</sup>). These tools are based on an Xwindows environment running on a UNIX based platform. They have been used to support S.E. teaching in previous years and thus their availability in the new system was desirable.

### 5.4 How CASE tools are served via the web

The Xwindows (or simply "X") environment (Mansfield<sup>5</sup>) is another client/server based system used to provide a windowing user interface. Programs that utilise "X" to generate their display (clients) send messages to the server program requesting a certain service such as the displaying of a piece of informa-

tion or the returning of a mouse position. The server then performs this action if possible.

When a Xclient and Xserver cooperate in this manner an Xsession is said to exist. The instigation of such a session involves the establishing of a link between client and server and some negotiation concerning computer addresses, terminal capabilities and security.

In our system, the web is used to negotiate between the machine running the web client ("the workstation") and the machine running the web server ("the host") to initiate an "X session" with the host running the CASE tool ("the Xclient") displaying its results on the workstation running an Xserver.

The CASE tools are thus run on the server, with only X events (mouse moves, key presses etc.) being transferred between the two. This keeps communication to a minimum, while allowing the (high performance) host to do the work and a minimal specification PC to act as the workstation.

## 6 Conclusion

Although the current academic year will see the first real use of the system, it has already fulfilled some of its aims. The feasibility of using the Web/Internet to remotely serve CASE tools has been established, as well as the technique of integrating tool access within other teaching material.

The delivery of teaching material using this method allows open learning to occur in a truly open manner. The benefits of having the material networked include the automatic delivery (via email) of completed exercise answer, model solutions, marks and even diagrams output from CASE tools.

Work still remains to be carried out. The development of full modules for a postgraduate course delivered in distance mode has commenced as has the extension of the system to facilitate automatic setting and marking of multiple choice tests.

## References

1. Birch, D., Latcham, J., *Managing Open Learning*, FESC, 1985
2. Dorrell, J. *Resource based learning: using open and flexible learning resources for continuous development*, McGraw-Hill, 1993
3. Ferguson R. I., Archibald J.M., Hedges S. & Parrington N.F., *Object Oriented Software Engineering using AI techniques*, Pro. ECOOP'94 workshop on Object Oriented Software Engineering and Artificial Intelligence
4. Ferguson R.I., Parrington N.F.& Dunne P., *MOOSE: A method designed for ease of maintenance*, Awaiting publication in Pro. IFIP ICQSP'94,



## 100 Software Engineering in Higher Education

Hong-Kong, 1994

5. Mansfield, N. & Davey, P., *The X window system*, Addison-Wesley, 1994
6. Rousseau, B. & Ruggier, M., *Writing Documents for paper and WWW*, (ed Ruggier M), Pro. First International Conference on the World Wide Web, Geneva, Switzerland, 1994, Elsevier, Amsterdam, forthcoming
7. Whitcroft, A. & Wilkinson, T. A., *A Tangled Web of Deceit*, (ed Ruggier M), Pro. First International Conference on the World Wide Web, Geneva, Switzerland, 1994, Elsevier, Amsterdam, forthcoming