

KMi Technical Report

KMi–TR–85



Contextually-enriched Documents: Publishing for Organizational Learning

Marek Hatala

Knowledge Media Institute
The Open University

January 2000

Contextually-enriched Documents: Publishing for Organizational Learning

Marek Hatala

Knowledge Media Institute
The Open University
Milton Keynes, MK7 6AA, UK
<http://kmi.open.ac.uk/people/marek/>
M.Hatala@open.ac.uk

ABSTRACT

We are looking at how new tools and forms of document interface can be used to support document centered discourse, to enrich documents with contextual information, and ultimately, to support models of organizational learning. Building on the Digital Document Discourse Environment (D3E) we have developed tools to support the publishing of digital documents tightly coupled with discussion spaces supporting the sharing of knowledge both throughout organizations and geographically dispersed communities. Based on the requirements arising from the three industrial trials we have constructed tools for publishing enriched documents within an organization to the support the capturing of organizational knowledge and practices.

Keywords

Knowledge, Ontologies, Organizational Learning, Digital Documents, Electronic Publishing, Hypertext, World Wide Web

INTRODUCTION

Organizational learning is today a hot research topic supported by governmental and corporate funding agencies [5]. Several companies [12] are building systems to capture organizational knowledge and integrate working and learning.

The goal of our research¹ [15] is to develop tools and methodologies for integrating working and learning within knowledge-intensive organizations. The core of our approach centers on enabling practitioners to progressively enrich their representation of work with important contextual cues and information arising from social discourse processes. In our view, contextually-enriched documents are the main information media that serve the practitioners in the new knowledge economies since documents form the core of many business practices and are the objects that many practitioners work with daily. The Contextually-Enriched Document Archive (CEDAR) will provide practitioners with a document-centered discourse space for structuring discussions around

representations of work; client tools enabling users to articulate and refine domain concepts by incrementally enriching their representations of work with related discussions and underlying knowledge models; knowledge delivery and access mechanisms supporting individual and group learning; and an intranet-based organizational memory server for knowledge capture, structuring and maintenance.

The CEDAR toolkit builds on several existing technologies: Digital Document Discourse Environment (D3E) [14] supporting document-centered discourse, WebOnto [4] a tool for building ontologies and knowledge modeling in OCML [11], CBR-Answers for the case base retrieval of web documents.

In this paper we describe the design of the CEDAR publishing toolkit. The CEDAR publishing toolkit builds on D3E. D3E comes with a generic publishing toolkit which can be used to generate a variety of highly tailorable sites. Despite the availability of the publishing toolkit the publication process is quite demanding – publishing a document takes reasonable amount of time and requires a high degree of process knowledge. The original toolkit has been used in three trial industrial applications to provide feedback for new tool development. Guided by the requirements from the trials we are following our objective to provide practitioners with a system that enables publication while preserving the high tailorability of the resulting document.

The remainder of this paper begins with brief presentation of D3E and D3E toolkit. The next section describes the industrial trials using the original D3E toolkit and their findings. We transform findings into the design principles for the CEDAR publishing toolkit. Finally, we show how these principles are realized in the toolkit and we conclude with future work and plans for further research.

¹ EU Esprit 4 Project – ENRICH: Enriching Representations of Work to Support Organisational Learning (<http://kmi.open.ac.uk/projects/enrich/>).

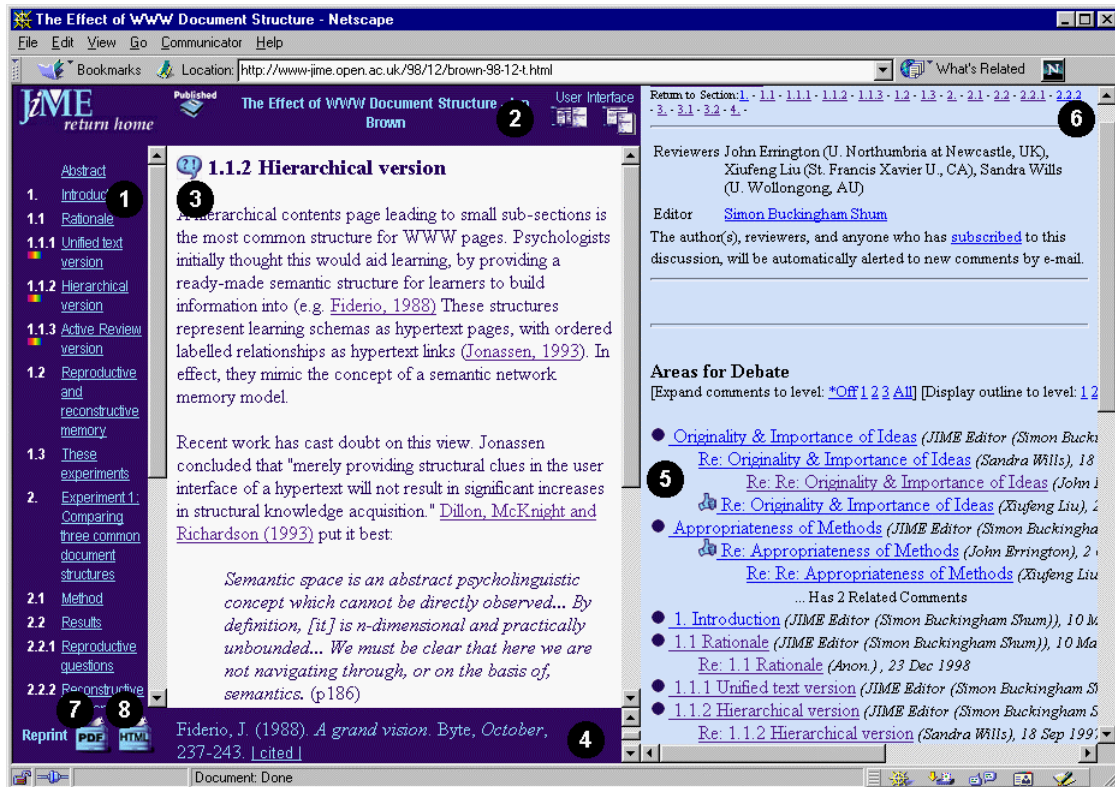


Figure 1 Output of the D3E publishing toolkit: On the left is an Article Window, on the right the Discussion Window shows a structured discussion about the document. Key: (1) active contents list; (2) banner; (3) Comment icon embedded in each section: displays section specific comments; (4) Citation linked from and reverse-linked to the article window; (5) general discussion topics defined in the toolkit and topics for section-specific comments, (6) links back into article; (7) and (8) download icons for full article.

D3E: DIGITAL DOCUMENT DISCOURSE ENVIRONMENT

Digital Document Discourse Environment (D3E) [14] is a

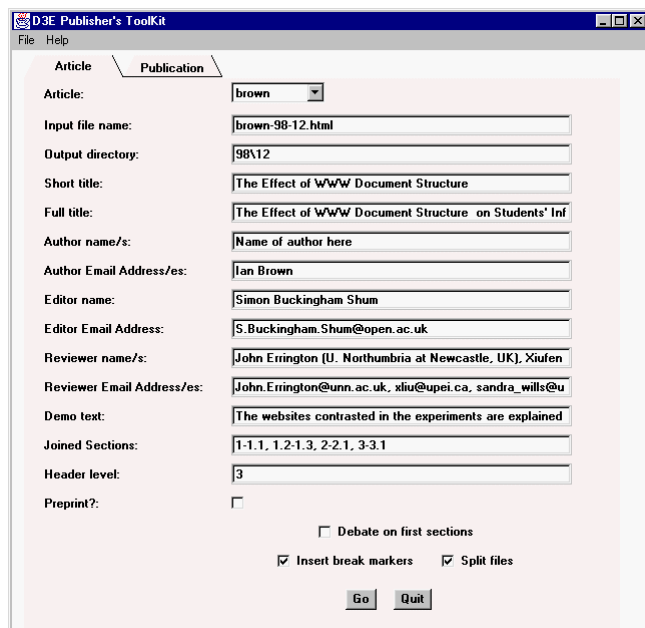


Figure 2 D3E publishing toolkit

suite of technologies supporting the publication of web-based documents integrated into a virtual discourse space supporting structured discussions between distributed communities. To date, D3E has been used in numerous contexts [2, 6, 7, 8]. D3E consists of tools for generating and managing a site and tools supporting the document interface. The Publisher toolkit is a Java™ application that takes HTML file as input and parses this file to create a new HTML files with special forms of navigation and integrated discourse facilities. The look and feel of a site is defined by creating a set of HTML template files containing D3E-specific variables that indicate where different kinds of document content should go. In the document interface the discourse facilities use a tailored version of HyperNews [9].

The D3E publishing toolkit (Figure 2) is used by an article publisher. First, the input HTML is imported into the toolkit on the editor's desktop computer by selecting the input file. Next, the article-specific information is filled into the form. On clicking the 'Go' button the toolkit takes the input HTML file and generates a new set of HTML files, creating the document interface shown in Figure 1.

The example in Figure 1 is taken from the Journal of Interactive Media in Education [6], an electronic journal published within Knowledge Media Institute. Comment icons (3 in Figure 1) which take the reader directly into the relevant area of the discussion space are embedded into each article section. An active table of contents (1) is constructed from the HTML header tags up to the level specified in the publishing toolkit. The document is split into the smaller sections and navigation buttons to the previous, next, and top sections are inserted at the foot of each file. The toolkit also generates seed files for the discussion space shown in the right frame (5) in Figure 1. The discussion topics match the paper's section headings, plus sections for general discussions added by the publisher (e.g. Originality). Links (6) back into the different sections of the article facilitate smooth navigation. The files generated by the toolkit are typically fine-tuned by the publisher. When the publisher is satisfied with the result, he or she moves the set of output files onto the web-server and announces the new URL.

TRIALS AND SCENARIA

Our project research strategy does not separate the phases of development and the deployment. Instead, the process of methodology and tool development is intertwined with deployment. In the early stages of deployment we have used existing tools that provide important feedback and are driving further tools development. Specifically, we use a case approach: three pilot projects provide the context for all tools development and evaluation activities. Each case involves both types of CEDAR users - developers and end-users - but in different domains and business contexts.

In the following description of the trials we concentrate on the publishing process and findings related to the publishing process. The first trials have used the existing D3E publishing toolkit for publishing documents in CEDAR. The trial findings set up the requirements driving the CEDAR publishing toolkit development

DFKI/Saarbergwerke Case: The ProGroup Electronic Manual

Saarbergwerke AG is an international energy and technology company with over 18,000 employees. In previous projects with DFKI GmbH, Saarbergwerke AG developed and deployed a system supporting the systematic recording, structuring, and dissemination of knowledge and experiences relevant to the maintenance of complex mining machines [1]. During machine operation, events, observations, and actions are documented, classified and stored in an organizational memory.

The systematic revision of entries supports the continuous evolution of the accumulated knowledge. Model-oriented, heuristic retrieval mechanisms realize a goal-oriented knowledge dissemination as an effective communication support. The results clearly demonstrated the high possible benefit which can be gained from IT support for

organizational learning. In the ProGroup Electronic manual case the goal is to integrate multiple group memories at various locations into a common discourse environment, using web-based techniques provided by the CEDAR toolkit providing a media for organizational learning.

In the first trial existing fault records were exported from the fault recording system (ESB) [1] and manually transformed into the D3E form using the D3E publishing toolkit. As a result, the fault records were enriched with the discussion spaces and they became accessible to the trial group formed at Saarberg. Now, when a new fault record is entered into the ESB system, it is exported in the HTML form and e-mailed to the system administrator/publisher at DFKI who converts the file into the D3E form.

The trial produced following findings/requirements:

Req-1: A big time lag in introducing new records into the D3E form: the need for 'real time' document conversion.

Req-2: Inability to easily create a table of contents.

Req-3: An extra step of moving generated files to the server prevents the end user from publishing documents: the automatic placement of documents on the server required.

Req-4: D3E templates difficult to manage: a template editor needed.

Req-5: Support for the modification/extension of an existing D3E document.

Req-6: A batch processing mode.

TecInno/Siemens Case: The SIMATIC Help Desk

Siemens Automation & Drives (A&D) is an international division of the Siemens AG. The product line of Siemens A&D covers the product and systems business in manufacturing and process industry. FAQs and all other information concerning SIMATIC products are collected by Siemens A&D on the Intranet. Using a TecInno product CBR-Answers [10], the SIMATIC knowledge manager tool enables an intelligent access to FAQs and other documents and their information. The knowledge manager incorporates knowledge about the SIMATIC product line. This enables him/her to recognize the relevant documents for a product.

In the TecInno/Siemens case the enrichment of technical documents will enable Siemens engineers and technical support people to share their expertise. In this case scenario individuals will tackle a new situation based on the reflection on known problem solutions supported by case-based retrieval. The documented new product application will be shared among the technical support and sales people as well as with engineers. Finally, the

CEDAR will allow help desk personnel and engineers to discuss their respective viewpoints.

In the first trial the TecInno converted a subset of an existing archive of technical documents into the D3E format and provided the technical support staff with the enriched environment. Two more requirements were produced:

Req-7: The need for support for working on a set of documents.

Req-8: Generation of table of contents for the set of documents.

BAe Case: The Team Workbook

British Aerospace has in place a change program called 'Benchmark BAe' designed to help the Company achieve its goal to become a benchmark company through the pursuit of excellence. Five values have been identified – Customers, People, Performance, Innovation & Technology, and Partners – as keys to achieving this goal. As part of this larger programme, a Team Workbook has been developed to help promote the understanding and application of these values in the operational and business environment. The Team Workbook is a planning tool designed to stimulate more effective working of both team leaders and their teams. The original Workbook is paper based and is in the process of being deployed across the entire company (44,000 employees). During the project a digital Value-Enhanced Team Workbook underpinned by a corporate knowledge base is being constructed.

During the first trial the original D3E Publishing Toolkit has been used to produce a first digital version of the workbook. The workbook itself contains more than 140 questions guiding a team leader through a three step process of preparing, declaring and delivering, and reviewing and improving values-based plan and work practices. The aim was to produce a workbook looking graphically similar to the original version and enabling a team to answer/discuss questions in the D3E discussion space.

Because of the length and structure of the input document² the conversion process using the original publishing tool was lengthy and required a lot of manual pre- and post-editing actions at the level of HTML files. The findings of the trial produced were similar to the DFKI case and one more requirement has been produced:

Req-9: Need for support for allowing flexible conversion of a document.

PUBLISHING FOR ORGANIZATIONAL MEMORY

From an analysis of the trial results it became clear that the requirements set by trials could not be achieved by the

² The original Workbook has more than 60 pages and main hooks for the discussion are list items instead of the headings.

modification of the existing publishing toolkit. We have therefore proposed a new publishing toolkit based on a client-server architecture and supporting most of the requirements. In the following section we summarize the trial requirements and present a design rationale for the new publishing toolkit. In subsequent sections we present a prototype of the toolkit.

Requirements and design rationale

Table 1 summarizes the requirements found by the trial applications. Based on these requirements and an analysis of the trials scenarios we have identified the following design requirements for the new CEDAR publishing toolkit:

Client-server architecture

This satisfies the main objective of CEDAR to be an organization-wide system supporting organizational learning. Using client-server architecture the requirements Req-1, Req-3, Req-6, Req-7 and Req-8 could be supported.

Document conversion functions located at the server

To support the automatic placement of generated files (Req-3) in the CEDAR the conversion algorithms must reside on the server to avoid the transmission of files. This also supports requirements Req-1 and Req-6.

Organization-wide and personal templates for automatic publishing

Templates are required if we want to support a batch

Table 1 CEDAR publishing toolkit requirements

DFKI	TecInno	BAe	Requirements	Supported by
•		•	'Real-time' conversion into D3E format	Client-server architecture, templates, fine-tuning
•	•	•	Flexible table of content	Content template, manual fine-tuning
•		•	Automatic placement of documents on the server	Document conversion functions located on server
•	•	•	Flexible D3E layout/templates editor	Template editor, manual fine-tuning
•	•	•	Extension/modification of existing D3E document	Not supported
•	•		Batch mode of operation	Server support, templates
	•		Processing of sets of documents	Server support, templates
	•		Content table(s) supporting sets of documents.	Server support, templates
		•	Flexible changing of conversion parameters	Manual fine tuning

mode of operation (Req-6) and the processing of document sets (Req-7). The requirements Req-1 and Req-8 are supported by templates as well.

Easy to use template editor

This design requirement directly reflects the Req-4 and supports the previous design requirement.

Tool for the fine-tuning of documents

The tool will support the requirement Req-9 and will provide additional support for requirements Req-1, Req-2, and Req-4.

A prototype of the CEDAR publishing toolkit has been implemented as a tool which can run within any web-browser. The prototype consists of a Java applet and server support. The division of functionality between the applet and server has been guided by the requirement for a batch mode of operation. The main document publishing functions are located on the server together with the templates database. The applet provides a template editor, manual publishing and manual fine-tuning support. Table 1 also shows how the requirements are satisfied by the CEDAR publishing toolkit.

Manual publishing steps

The CEDAR publishing toolkit applet is a wizard helping the user to publish his/her HTML document on the server. The publishing process enables the user to immediately see the result – a published D3E document.

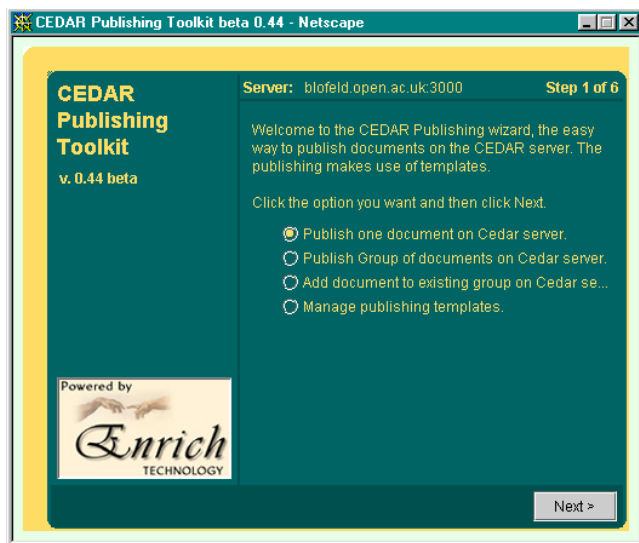


Figure 3 Step 1 of publishing process

Step 1

In the first step (Figure 3) the user selects one of the following: manually publishing one document, publishing a set of documents, adding a document to an existing set, or template management. According to the selected action the wizard follows an appropriate sequence of publishing steps.

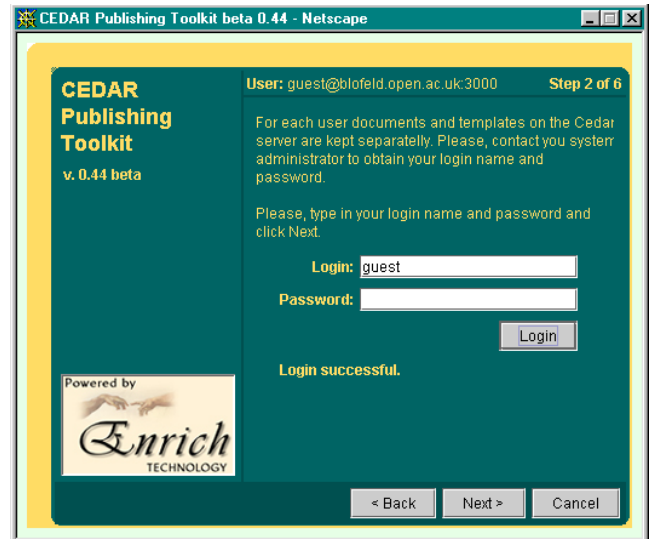


Figure 4 Step 2 of publishing process

Step 2

In step 2 (Figure 4) the user logs into the system. By logging the user can access his/her templates. The system informs the user whether the login was successful or either a login name or password were incorrect. The top banner of the applet shows the current user and the server name the user is logged in.

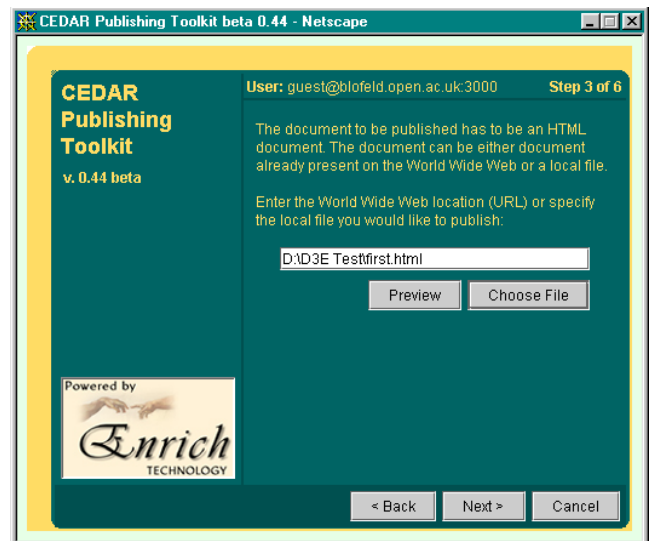


Figure 5 Step 3 of publishing process

Step 3

In step 3 (Figure 5) the user selects an input file or a set of files either on the Web or by choosing files on his/her desktop system. If the action is to publish a single document, the user can preview the document. If the action is to publish a set of documents, the user is allowed to choose multiple files within one directory or a whole directory. In this case the systems asks for the confirmation whether the user wants to publish the whole directory.

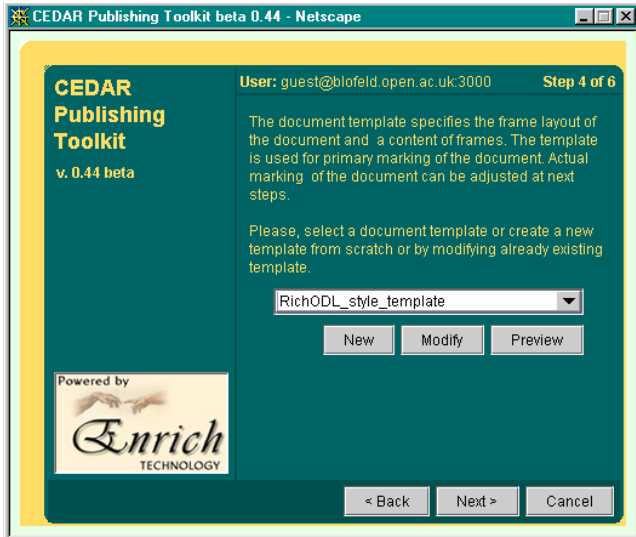


Figure 6 Step 4 of publishing process

Step 4

Step 4 (Figure 6) enables the user to select and/or manage his or her templates. A combo box displays a list of all templates available to the user. The list is a union of the organization-wide templates and user's own templates. The user can Preview, Modify or create a New template (see the section below).

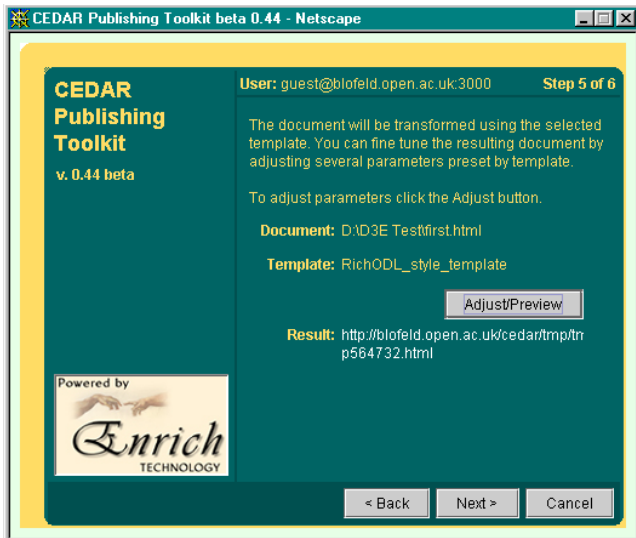


Figure 7 Step 5 of publishing process

Step 5

In step 5 (Figure 7), if the user publishes a single document he/she can preview how the document will be processed using the selected template and manually adjust the converting parameters (see the section on Fine Tuning below). When multiple documents are published the first document can be previewed but adjustment is not allowed.

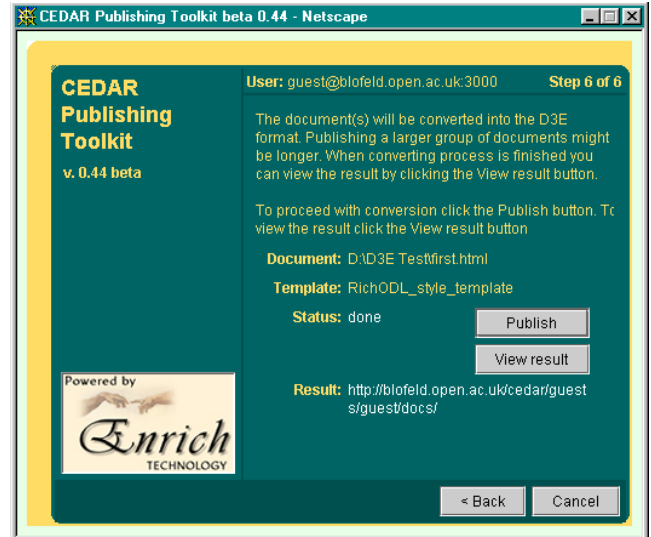


Figure 8 Step 6 of publishing process

Step 6

Finally, in step 6 (Figure 8) the document is converted and the user can browse the result. If the user is not satisfied with the result he/she can dismiss it and backtrack to some previous step.

Templates

The requirement of real time publishing can be satisfied only when the tool already knows how to convert the document. This raises the necessity to have a set of publishing templates specifying how a document is broken down into smaller pieces, what the resulting layout will be, how a table of contents is formed and where the discussions will be hooked to the document. There is twofold benefit that the templates provide: they allow a batch mode and multiple documents publishing, and they provide an initial format before the fine-tuning for manual publishing mode.

In the prototype we support two levels of templates: organization-wide templates and personal templates. The organization-wide templates developed by the CEDAR administrator cover the most frequent types of documents that are being published within the organization (e.g. quarterly business plan). The personal templates are those developed by CEDAR users for their group-specific documents that often undergo the discussion process within the group.

The Document template is a collection of templates specifying different aspects of a resulting document. The Frame layout template specifies the frame layout of the resulting D3E document. Frames in the Frame layout template can have assigned either a source URL or one of the following templates: Body template, Content template, Discussion template, and Banner template. These templates represent the most frequent types of documents included into the frames. If the user wants to place

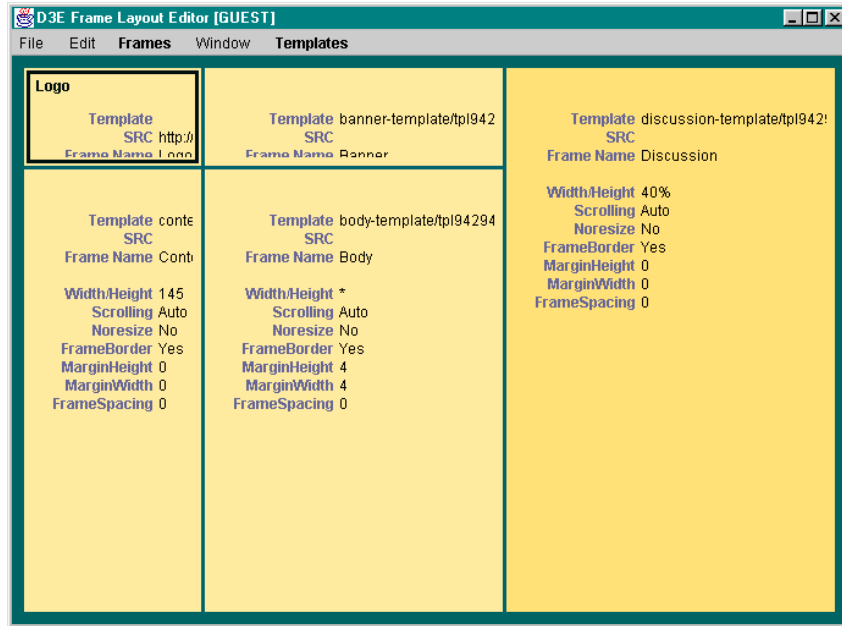


Figure 9 Frame layout editor

another document into the frame he/she can create an HTML document and refer to it using a standard frame parameter (SRC). The CEDAR publishing toolkit provides dialogs and graphics for editing templates with the possibility to preview results. In the following subsections we characterize each of the templates.

Frame layout template

The Frame layout template specifies the resulting layout of the D3E document. The graphical frame layout editor shown in the Figure 9 enables the user to split/join frames and drag their boundaries.

providing the URL of the document that will fill the frame. Another option is to assign a template to the frame. In this case the content of the frame is generated from the input document when this is converted to the D3E format.

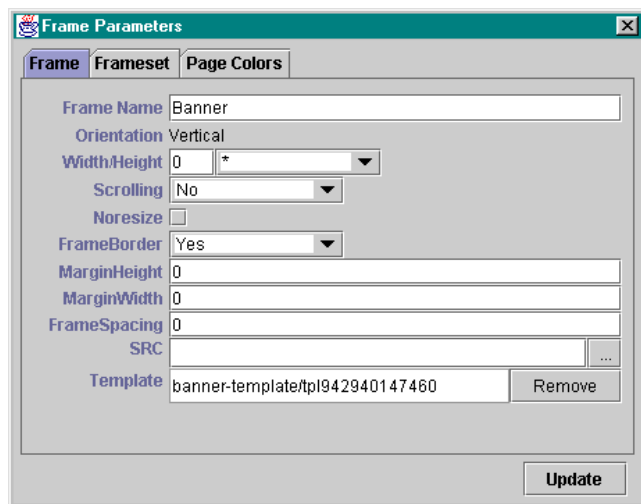


Figure 10 Frame Properties Dialog

Properties Dialog

Using the Properties dialog shown in Figure 10 the user can set the standard frame parameters. A frame content can be specified using two last fields in the Properties dialog. A standard way is to specify the SRC parameter by

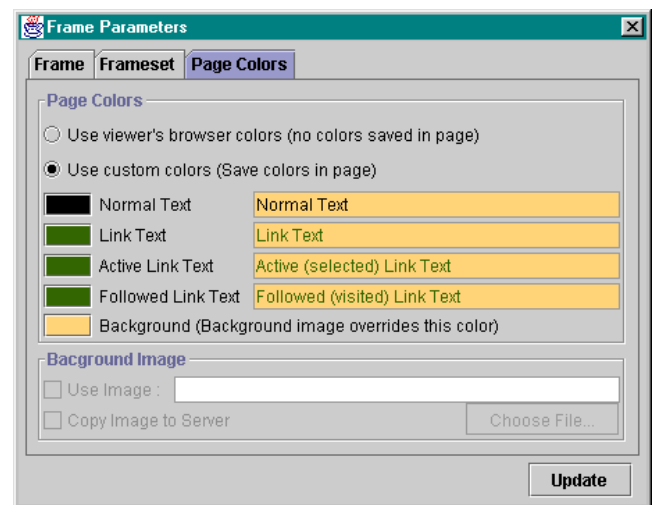


Figure 11 Page Colors tab.

The Page Colors tab (Figure 11) in the Properties dialog is active only when the frame has assigned a template. The page colors and background image can be specified in the standard way as for the HTML documents.

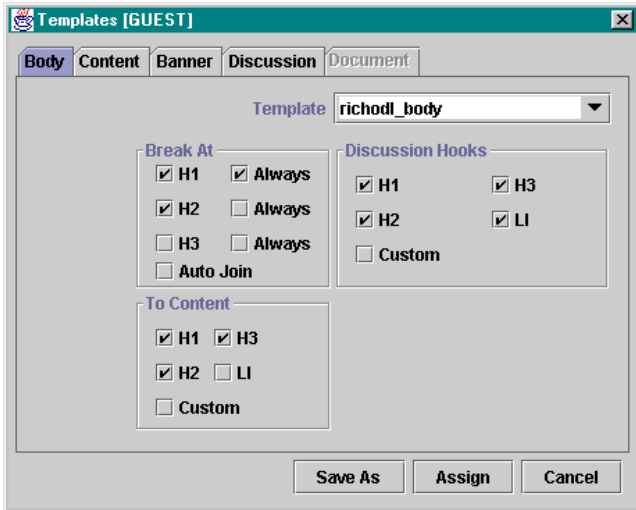


Figure 12 Body template

Body template

The Template dialog consists of several tabs. The Body tab shown in Figure 12 specifies parameters for the frame where document text will be displayed. If the text of the original document is too long it can be split into smaller sections linked with the next/previous/top buttons. How the document will be split can be specified in the Break At group. The To Content group specifies which places in the document will be referenced from the generated content. The Discussion Hooks group defines where in the document body bubbles providing links to an associated discussion space will be placed.

The other way how to specify above parameters is to use toolkit specific tags. The <DISC-HOOK> tag is recognized by the toolkit and places the discussion hook in any place in the document. The topic can be specified either by parameter TOPIC or by the text between the <DISC-HOOK> and </DISC-HOOK> tags.

The <BREAK-AT> tag specifies the exact place where the document will be split to sections. It can have up to two parameters: TOC-ITEM specifies the text which will form a table of contents item with link to the created section. If the TOPIC parameter is present it will place a discussion hook into the beginning of the created section with the specified topic.

We remind the reader that the parameters preset in the Body template can be used in two ways. When document is published automatically it is converted according to these parameters. However, when the publishing is done manually by the user, the parameters are used for pre-setting places in the document and the user can manually adjust them and modify a final look of the converted document.

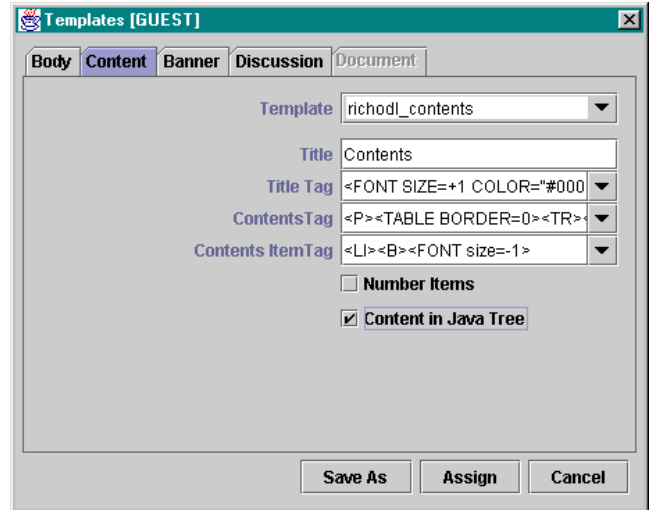


Figure 13 Content template

Content template

A Content tab shown in Figure 13 specifies how the content will be formatted. The content items are those specified by the Body template in the To Content section. If the Content in Java Tree is checked a page containing the Java applet with the content outline is generated and user can swap between the HTML and Java versions of table of contents. The Java Tree content is generated as a default when converting a group of documents.

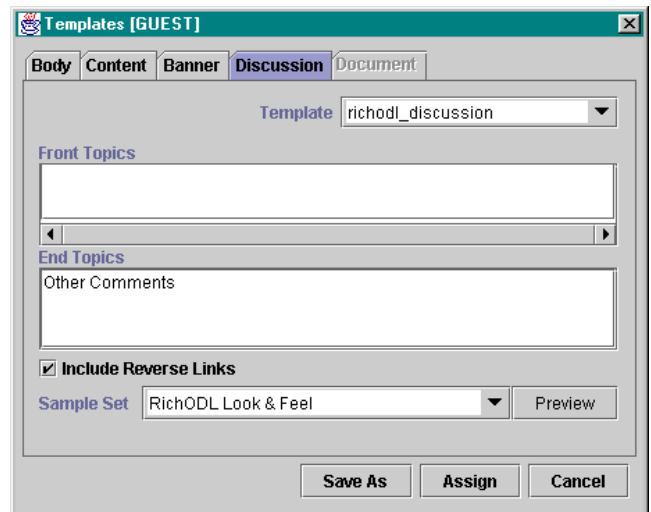


Figure 14 Discussion template

Discussion template

The discussion space associated with the document is formed by discussion topics hooked to those points in the document body that were specified in the Discussion Hooks group in the Body template. In the Discussion template the user can specify additional discussion topics that will be added either in front of or after discussion topics generated from the document body. This is useful when generating a repository of documents with a unified discussion structure or when a non document specific topic

is to be generated (e.g. the topic Originality when publishing a journal article).

The Sample Set specifies a set of graphic buttons and color scheme used for discussion. However, it also provides the navigation buttons for the body sections and discussion hook icon.

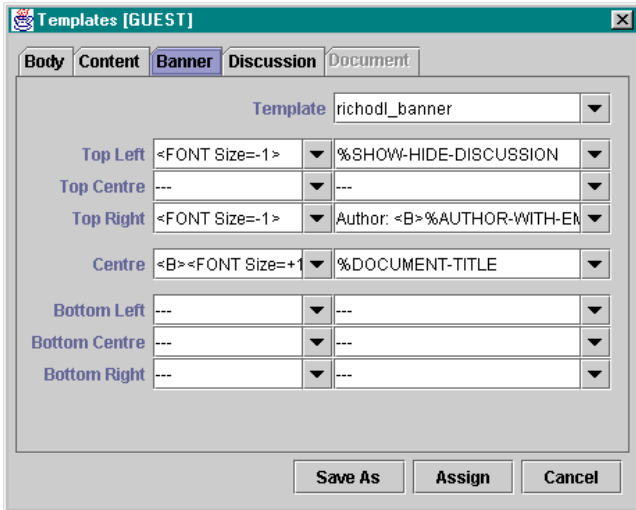


Figure 15 Banner template

Banner template

This is a rather general template enabling the user to specify what goes to which part of the frame. Figure 15 shows the Banner tab in the Template dialog. The banner is formatted as a table. The user can specify what goes to which part of the table and how it is formatted. The

combo boxes contain a predefined set of variables that are user and document dependent. They are replaced during the publishing process with the values provided by the user or extracted from the document (e.g. from the <TITLE> tag).

The Banner template differs from previous templates as more than one banner template can be assigned to frames in the layout.

Fine tuning

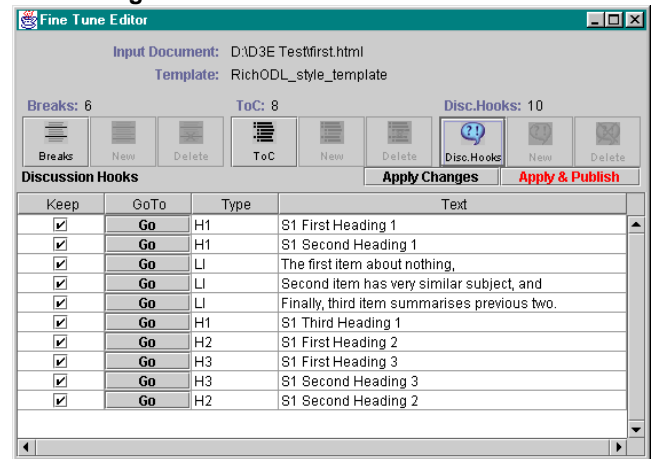


Figure 17 Fine Tune Editor

The use of templates for converting a document is the only way to automatically convert documents. However, when documents are converted manually the templates as any other pre-set mechanism might be to restrictive. In the

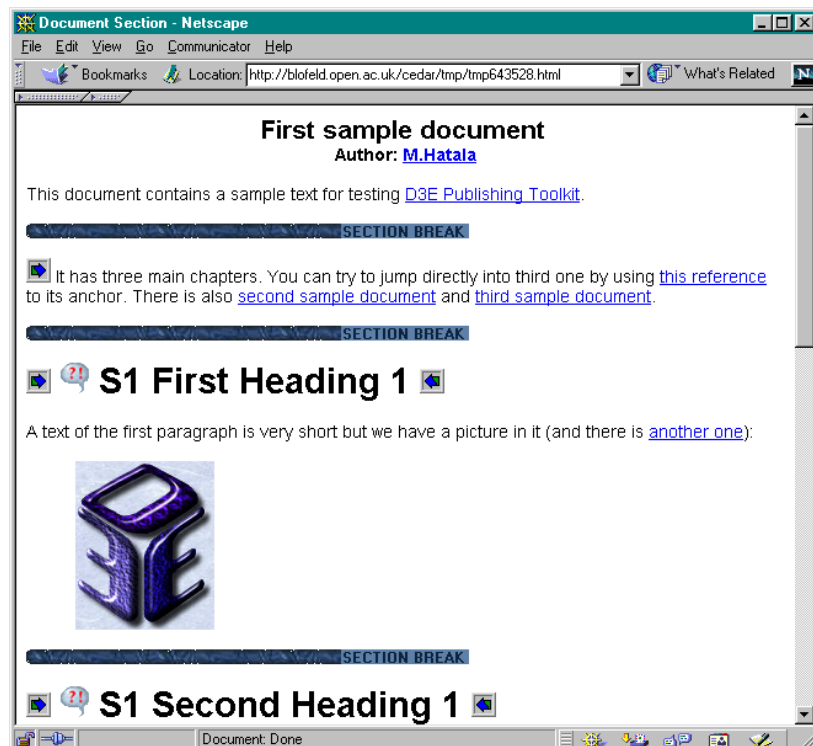


Figure 16 Preview of marked document

publishing process we have introduced one more step – fine tuning. When manually publishing a single document the templates are used to determine how the document will be broken into smaller pieces, where the discussion topics will be hooked and from which items a table of contents will be formed. The user is presented with marked documents for fine tuning (Figure 16). By using a special tool shown in Figure 17 the user can remove/modify conversion parameters:

- Remove breaking points
- Remove or modify content items including the wording of the content item
- Remove or modify discussion hooks including discussion topic titles.

The adjustments made by the user are recorded on the server and is valid only for the current document.

Figure 18 Document Data form

Publish it! (server support)

The final step is to actually convert the document into the D3E format. If the document has been already published on the CEDAR server the systems asks the user whether he/she wants to republish the document (Figure 19). Otherwise the user is directly presented with the document data form (Figure 18). The user fills in the document data into the form. The form is pre-filled with the user specific information. The fields that will appear in the resulting document are marked with an asterisks.

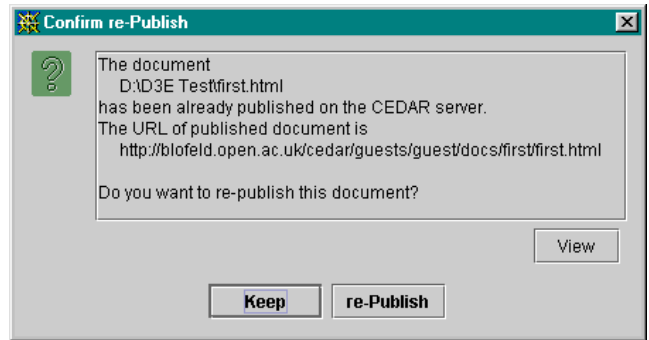


Figure 19 Re-publish dialog

A request is sent to the server that generates all the required files according to the templates or adjustments made by the user. The librarian module at the server takes care of the location of the files and of the setting of access rights for the generated document³. Generated files are placed directly onto the server and the result is immediately viewable by the user. A link to the resulting D3E document is added to the New Additions folder in the user’s directory hierarchy. An example of published document is shown in Figure 20.

The situation is slightly different when converting a group of documents. The documents are converted as in the previous case. Moreover they are interconnected together by next/previous links at the document level. A new directory containing the structure of converted files is added to the user’s directory structure.

The two publishing methods described above are supported by the server service that is accessible through a socket connection. This enables a batch mode operation when input files are generated by some other software and need to be converted without user intervention (e.g. in the DFKI case). However, the current version of batch mode works on a single file basis. Simple Perl script performs the role of the client: open a connection to the CEDAR server, send a conversion request and transmit the input file. The file is converted in usual way and is added into the collection it belongs to.

Implementation notes

The CEDAR publishing toolkit is a client-server application. The server side running either under Windows 95/98/NT, Solaris OS, or Linux consists of a set of collaborating parts. Publishing algorithms and templates management are implemented in LispWeb [13] – a Common Lisp based web-server. The generated D3E files are located in the web space of an Apache web server. The discussion support is provided by the D3E

³ A standard web-server authentication mechanism is currently being use to control access to the files. Three categories of users are currently supported – the owner, the owner’s group, and others.

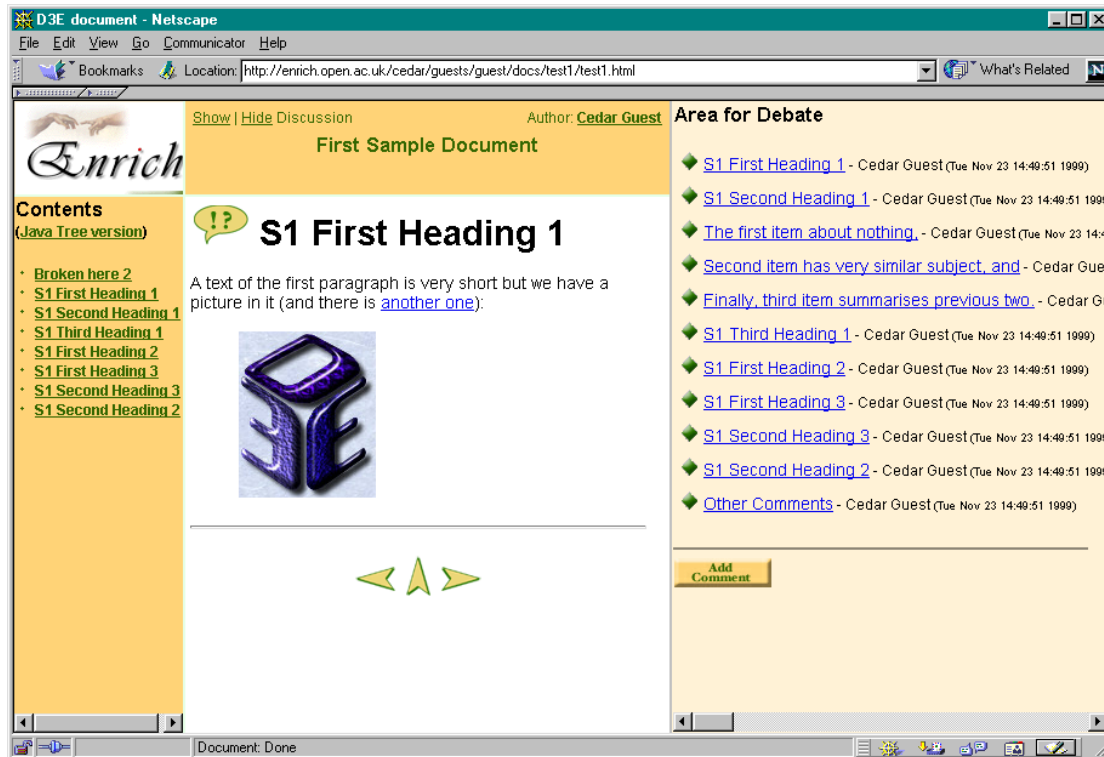


Figure 20 Published document

environment that at the server sides consists of a aGNeS News Forum system.

CONCLUSION AND FUTURE WORK

In summary, we have described the three settings where contextually enriched documents are used to provide media for organizational learning. The CEDAR publishing toolkit makes the publishing of documents straightforward for end users and preserves high tailorability as requested by the systems developers/publishers. Using the CEDAR publishing toolkit the HTML documents are converted into the documents surrounded by the discourse space. The original documents are enriched as surrounding discourse is reified and embedded back into the documents themselves. The toolkit enables the 'real time' publishing of emerging documents into a form supporting the exchange of views and preserving the processes of solution finding in organizational memories. Other tools that are targeted by our research will support the enriching of documents with knowledge concepts and the effective retrieval of documents from the organizational memory.

The prototype of CEDAR publishing toolkit is now being evaluated in a second set of trials both by developers/publishers and end users. While awaiting the trial results we are working on toolkit enhancements. We are integrating the annotator enabling the user to annotate generated document with ontology concepts. This will make possible to search for documents using knowledge model. A work continues on the enhancement of the template editor and template hand-edit support for

advanced users enabling them to create more distinguished documents. The server side is being enhanced in several directions, including: rebuilding the librarian and user management, revising algorithms for the conversion of document sets, and document sets content generation.

ACKNOWLEDGEMENTS

This work is fully funded by the Enrich project (EU project 29015 in the IT for Learning and Training programme). Special thanks to John Domingue for his help with this paper. Java is a registered trademark of Sun Corp., LispWorks is a registered trademark of Harlequin Corp.

REFERENCES

1. Bernardi, A. Electronic Fault Recording: A Corporate Memory for maintenance support of complex machines, *Symposium on the Management of Industrial and Corporate Knowledge*, Compiegne, France, (October 1997), 20-21.
2. Buckingham Shum, S., and Hammond, N. (eds.). World Wide Web Usability: Special Issue, *Int. J. Human Computer Studies*.47(1), 1997, (entire issue).
3. Buckingham Shum, S., and Sumner, T. New Scenarios in Scholarly Publishing and Debate, in Eisenstadt, E., and Vincent, T. (eds.) *The Knowledge Web: learning and Collaborating on the Net*, Kogan PageLtd, UK, 1998, 135-151
4. Domingue, J. Tadzebao and WebOnto: Discussing, Browsing and Editing Ontologies on the Web,

- Proceedings of Knowledge Acquisition Workshop KAW'98, (Banff, Canada, April 18-23, 1998)
5. IT for Learning and Training in Industry, Esprit Program, <http://www.cordis.lu/esprit/src/ltihome.htm>
 6. JIME, An Interactive Journal for Interactive Media, *Journal of Interactive Media in Education*, 1996, <http://www-jime.open.ac.uk/>
 7. KMi, Dearing Report Discussion Site, 1997, <http://kmi.open.ac.uk/Dearing/>
 8. Kmi, Learning.org, 1997, <http://kmi.open.ac.uk/learning.org>
 9. LaLiberte, D. HyperNews, 1997, <http://www.hypernews.org/Hypernews/get/hypernews.html>
 10. Lenz, M., Hübner, A., Kunze, M. Question Answering with Textual CBR. in: T. Andreasen, H. Christiansen, H.L. Larsen (Eds.): *Flexible Query Answering Systems*, Lecture Notes in Artificial Intelligence 1495, Springer Verlag, 1998
 11. Motta, E. An Overview of the OCML Modeling Language. In Proceedings of the 8th Workshop on Knowledge Engineering: Methods & Languages (KEML'98), Karlsruhe, Germany, January 21-22, 1998.
 12. O'Leary, D.E. Using AI in Knowledge Management: Knowledge Bases and Ontologies, *IEEE Intelligent Systems* 13, 3 (May/June 1998), 34-39.
 13. Riva, A. and Ramoni, M. LispWeb: a Specialized HTTP Server for Distributed AI Applications, *Computer Networks and ISDN Systems*, 28(7-11), a996, 953-961
 14. Sumner, T., and Buckingham Shum, S. From Documents to Discourse: Shifting Conceptions of Scholarly Publishing, in *Proceedings of CHI '98* (Los Angeles, April 1998), ACM Press, 95-102.
 15. Sumner, T., Domingue, J., Zdrahal, Z., Hatala, M., Millican, A., Murray, J., Hinkelmann, K., Bernardi, A., Wess, S., and Traphöner, R. Enriching Representations of Work to Support Organisational Learning, Proceedings of the ECAI'98 Workshop: Building, Maintaining and Using organizational Memories, (Brighton, UK, August 1998), 109-127.