

Linking Experiences: Issues Raised Developing Linkservices for Resource Based Learning and Teaching

Hugh C. Davis and Su White

IAM: Learning Technologies, The University of Southampton, UK.

hcd@ecs.soton.ac.uk, saw@ecs.soton.ac.uk

Abstract

Microcosm is an open hypertext system designed to facilitate a resource-based approach to learning and teaching. In spite of winning accolades the system never established itself as a mainstream educational product. This paper describes the technical and pedagogical features of Microcosm, and attempts to explain the reason for its limited take up. The paper is in the form of a case-study of both good and questionable practice.

1. Introduction

1988 was a good year for educational technology. The Apple Macintosh was well established, and came bundled with Hypercard; OWL had produced a hypertext system called Guide, which was available, on both the Macintosh and the Windows 2 platform, which had recently established itself as the GUI front-end of choice for DOS based PCs. Videodisc technology was advancing rapidly and genlock cards were enabling video output to appear within Windows on a computer screen. PCs were becoming affordable, massive storage was possible on CD-ROMs and local area networking was a reality.

At this point in time a number of researchers at Southampton sat down to design a new hypertext system. The Microcosm system was first described in 1990 [1]. In 1994 the system became the platform for a number of projects within the Teaching and Learning Technology Programme (TLTP) in the UK. At about the same time a spin-off company, Multicosm Ltd, was established to exploit the intellectual property. In 1995 Microcosm won a European Software (ITEA) Award. In 1996 it won the British Computer Society's Software Innovation Prize and StoMP, an application of Microcosm for teaching of Physics, won a European Academic Software Award.

In spite of all its successes, Microcosm never became a big player in the educational technology market, and the company that currently licenses the product is no longer actively marketing it. In this paper we examine the good points that Microcosm embodied, both from a technological and pedagogical point of view, and then we

analyze the reasons why such apparently notable features have failed to impress the purchasing public

2. The Pedagogical Perspective

The fundamental premise of the Microcosm design team was that learning is enhanced by access to an information system that allows a user to explore a set of resources related to a specific domain. Students who actively query a resource-base will build a better model of the topic than those who passively receive information [2]. An advanced hypertext presentation was seen as being a good way to provide such a learning environment; links may be used to associate a phrase representing a concept to related information. However, simple browsing and clicking on blue links does not necessarily encourage deep learning [3], so the team designed an environment which encourages the user to actively engage with the materials. This encouragement is provided in a number of ways:

1. The user has access to a document explorer window which maintains a hierarchy of the corpus of available documents. This is not rocket science, but many hypertext systems fail to provide this obvious parallel to the table of contents in a book.
2. The author may give the documents attributes (e.g. keywords) for the user to find appropriate documents.
3. Links may be hidden - rather than standing out in blue underlined text. To activate such links, the user must select an enclosing chunk of text and choose "Show Links" from a menu, as shown in figure 1. The theory is that this makes users ask the question "what would I like to be linked?" rather than waiting to find links in the text [4]. Of course it is important that there are lots of links, so that they are generally successful in finding links. If not they will quickly give up.
4. Users may make queries, using much the same approach as a modern web based search-engine, by typing in words they might expect to find in the document they are looking for. Similarly a user can select a chunk of text and choose "Compute Links" from a menu. In both cases the system will produce a ranked list of other documents (from within the corpus) that contain the similar vocabulary.

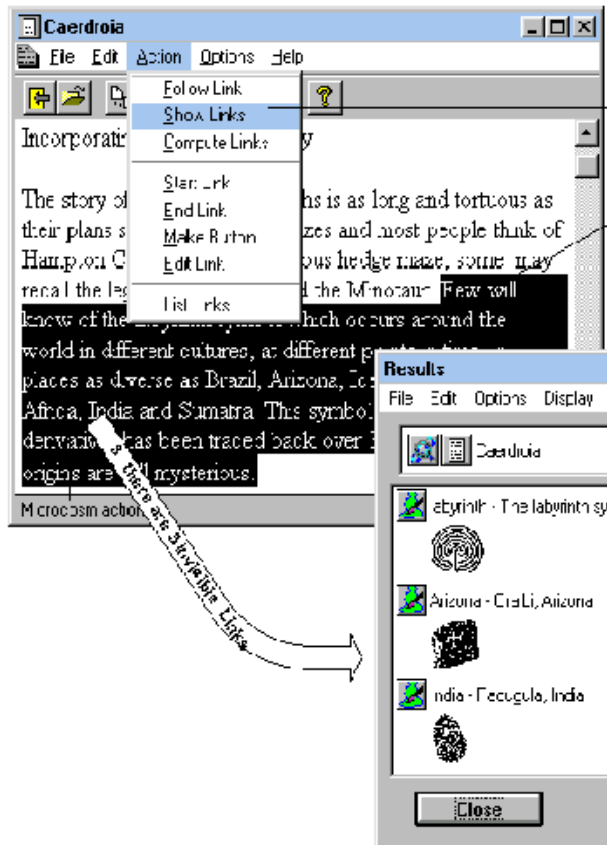


Figure 1: Finding invisible links. The user selected text and asked to show links. Three were found.

5. The user may create their own links and trails, and even add their own documents to the corpus.
6. The user may add annotations to existing documents.

Another important concern of the designers was to provide a system in which it would be possible to provide different links depending on *context*. So it is possible, for a given corpus of documents, to have a different set of links depending, for example, on whether the user is a beginner to the subject or an expert, or perhaps reflecting the user's intended purpose for the document. This is now referred to as "link level adapted hypertext" [5].

The designers wanted it to be possible for authors to include the full range of media types within their corpus of documents. This meant that it must be possible to create links into and out of any type of file, including sound (e.g. on a CD), video (e.g. on a videodisc), word processed documents, spreadsheets, bitmapped pictures, vector drawings etc.

Building a resource-base is a time consuming task. If you add the requirement that the resource base is hypertext linked then the task grows exponentially with the number of documents. Early experiments building small hypertexts using packages such as Hypercard had encouraging results with students, but were prohibitively

expensive to build [6] and that effort did not scale well to producing large corpuses with thousands of documents and tens of thousands of links. Consequently the Microcosm designers considered it very important to produce a system which enabled rapid development of learning materials; the goal was to produce a system which would allow the author to take existing documents (of any media type) and to import them into the document hierarchy. The index for querying would be built automatically, and existing links would be automatically re-purposed for use within the new document, thus making the task of linking much faster.

3. The Technological Perspective

The most significant feature of Microcosm is that the links and anchors [7], which are the fundamental components of any hypertext system, are not held in the document (as they are, for example, as HREF's in html), but in separate databases, called *linkbases*. When a user loads a document, the viewer queries all currently selected databases for any links that should be shown in this document, and displays them appropriately. The links have enough information stored with them so that the viewer can tell where in the document the link belongs, how to display the link, and the anchor at the other end of the link. This separated link approach has the immediate benefits that:

1. Authors, when making links can choose in which linkbase to store their links and users (or some agent acting on the user's behalf) may select which linkbases to activate. It is thus possible, for example, to keep links suitable for a beginner in a linkbase separate from that holding links suitable for an advanced student, and links suitable for an engineering student in a linkbase separate from that holding links intended for a business studies student.
2. Users may keep their own linkbases, in which they keep their private links and trails, without altering the set of links delivered by the author.
3. Documents do not need to be altered in order to add a link to the document. This is important if users are to be able to make their own links (it would not be acceptable to allow users to alter the author's documents). It also means that links can be made in read-only documents such as those on CD-ROMs, videodiscs and those to which we do not have write access.
4. Data formats do not need to be adapted in order to allow linking. Microsoft have recently extended the data formats of their office products to allow embedded HREFs but, for example, if you wish to put a link in a bitmap the only way this can be done is to hold the link externally.

5. The final advantage is the facility to make *generic links*. These are links which are associated with some word or phrase, rather than the position that they occur within a document. So for example an author might create a link on the phrase "insertion sort" which might link into a document about sorting methods, to a section about insertion sort. Now, wherever *any* document includes the phrase "insertion sort", this phrase will automatically be made into a link at runtime. The power of generic links is significant for two reasons. First, it is possible to automatically generate links to documents using their keywords (e.g. glossaries). Secondly, a generic link, once made, applies to all documents added to the corpus, so link authoring grows linearly with the number of documents, rather than growing exponentially.

A second important feature of the Microcosm architecture was the ability to display links in a very large number of data formats. In earlier versions of Microcosm this was achieved in two ways. a) by writing specialized viewers (e.g. for bitmaps, GIFs, MPEGs etc.) that understood the protocol to talk to the linkservice and to save and retrieve links. b) by adapting existing packages (such as MS-Word and emacs) using their built-in macro programming facility. Later versions of Microcosm used a third party viewer that had been adapted at source code level to talk to the linkservice; this viewer could present the vast majority of known data formats.

4. Analysis

From the point of view of the research group at Southampton, Microcosm was an enormous success [8]. The architecture allowed plug-in modules to be inserted into the system which in turn enabled us to experiment with new ideas in hypertext. The linkservice approach turned out to be very timely, and became one of the reference systems for the work of the Open Hypertext Systems Working Group [9]. The approach to resource-based learning was seen as important [10], [11] and led to funding from the UK University funding councils including the SCOLAR project [12] based in Southampton. Microcosm and the ideas it embodied formed the basis of the Multimedia Research Group which consisted of 6 people in 1990 and grew to a team of about 60 by 2000. Microcosm has been available as a commercially supported and maintained product since 1996, and is sold into the training, educational and technical documentation markets.

So the question which must be asked is "Why has the take up of this product been limited, and what lessons can be learned?"

4.1. The World Wide Web was mind narrowing

At about the same time that Microcosm was being introduced, the World Wide Web was also becoming available. Microcosm was implemented initially on Windows, and conversion to work across the Internet happened too late. The World Wide Web is without doubt one of the killer applications of the age; it certainly killed hypertext research and development for a few years. The problem was that it introduced a whole generation of users to the idea of hypertext as embedded unidirectional binary links that were used almost exclusively in a structural manner (contents to sections, this page to next page etc.). It is rare to see "associative hypertext" which links a concept to related information, and web browsers are only now becoming capable of hypertext features that were commonplace in research lab systems 10 years ago.

However, an important corollary to this lesson is that global access to shared information from heterogeneous platforms is a more fundamental requirement than the clever navigation and seamless information retrieval offered by Microcosm.

4.2 Systems must pass the elevator test

The elevator test [13] requires that you must be able to explain what your system does in the time that it takes to complete a ride in an elevator. You might for example describe hypertext as the ability to put buttons as signposts in your documents so that users could elect to follow links to further information. Much more than that and you lose the attention of the average decision maker.

The problem for systems like Microcosm is that simple descriptions as above do not distinguish them from everyday web browsers. So why would anyone want the product? The answer to that question requires you to read sections two and three of this paper, or at least whatever parts might be relevant to your needs.

The fact is that no-one wants to buy a general purpose hypertext widget; all major sales of Microcosm have been in response to user-defined needs and specifications.

4.3 Users give conflicting feedback

The people who first authored in Microcosm were enthusiastic "early adopters" [13]; they had used other systems such as CBT or the web and had found limitations. We received a great deal of advice and requests for enhancements. Being academics the design team always took the wide view; any individual enhancement is an instance of a class of enhancement, and the whole class must be solved.

The resultant system was enormously feature rich, and when users with less background knowledge tried to use it

they were baffled by the array of features and didn't know where to start. So the designers tried to hide the less important features and then the experienced users (and the pundits who have a view on everything but knowledge of nothing) complained again. This leads us to the final lesson.

4.4 A user-friendly system is a simple system

Some of the important design features of Microcosm were hard to implement in a simple and user-friendly way.

For example, integration with third party applications (using DDE/OLE) was fraught with problems. What worked on one machine did not work on another, and versions of Microsoft software became impossible to track.

Microcosm requires users to let the system know something about context (e.g. ability and topic of interest) in order to decide which linkbases to use to retrieve and store links. But users find this too great a cognitive overhead; they want the system to "just know" what linkbases to turn on and where to store personalizations.

When designing Microcosm we set out to produce a system that was simple enough for a non-technical computer user to easily put together a corpus. We overestimated the level of computer literacy of the average academic. We had assumed familiarity with the file system.

The bottom line is that research teams are not necessarily the best people to specify systems as they are more interested in the solution to tricky problems than the interface which allows an ordinary user to understand what they are doing.

5. Conclusion

Many of the ideas in Microcosm were visionary. Separating the links from content and associating rich resource bases are vital prerequisite to individual learning environments. Systems that provide such functionality are now emerging in the Web and internet environment. These include GENTLE [14] (based on Hyper-G), the DLS [15] which is the Southampton research group's offering and Portal Maximizer [16] which is produced by the company that originally commercialized Microcosm. The question remains whether these successor systems will be sufficiently straightforward for academic authors to achieve widespread use.

6. References

[1] Fountain, A., Hall, W., Heath, I. and Davis, H.C.. MICROCOSM: An Open Model for Hypermedia with Dynamic

Linking. In: A. Rizk, N. Streitz and J. Andre. Eds. *Hypertext: Concepts, Systems and Applications. Proc European Conference on Hypertext*, INRIA, France, November 1990, Cambridge University Press, 1990

[2] Jonasson, D., Mayes, J.T. and McAleese, R. A manifesto for a constructivist approach to uses of technology in higher education. In: Duffy, T.M., Lowyck, J. and Jonassen, D.H. Eds. *Designing Environments for Constructivist Learning*, p 231-247, Springer Verlag, 1993

[3] Hammond N. Tailoring Hypertext for the Learner. In: Kommers, P.A.M., Jonassen, D.H. and Mayes, J.T. Eds. *Cognitive Tools for Learning*. Springer Verlag, 1991

[4] Hall, W. Ending the Tyranny of the Button. *IEEE Multimedia* 1(1). 1994

[5] Brusilovsky, P., Kobsa, A. and Vassileva, J. *Adaptive Hypertext and Hypermedia*. Kluwer. 1998

[6] Lechtenberg, S and Joubert, G.R. Effort Estimation for Multimedia Information Systems Development. In Roger, J-Y., Stanford-Smith, B and Kidd, P.T. *Technologies for the Information Society: Developments and Opportunities*. pp 615-622. IOS Press. 1996

[7] Halasz, F. & Schwartz, M. The Dexter Hypertext Reference Model. In: *Proc Hypertext Standardization Workshop*. pp 95-133, Gaithersburg. US Government Printing Office. Jan. 1990

[8] Hall, W, Davis, H.C., & Hutchings, G.A. *Rethinking Hypermedia: The Microcosm Approach*. Kluwer. 1996

[9] Reich, S, Wiil, U.K., Nürnberg, P.J., Davis, H.C., Grønbaek, K., Anderson, K.M., Millard, D.E. & Haake, J.M., Addressing interoperability in open hypermedia: The design of the open hypermedia protocol. *New Review of Hypermedia and Multimedia*, Vol 5. pp 207-248. 1999

[10] Davis, H.C., Hutchings G.A. and Hall, W. A Framework for Delivering large-scale Hypermedia Learning Material. In: Hermann Maurer. ed. *Educational Multimedia and Hypermedia Annual. Proceedings of ED-MEDIA '93, Orlando, Florida, USA*, pp 115-122. AACE. 1993

[11] Hall, W., Hutchings, G.A. and White, S.A. Breaking Down the Barriers: An Architecture for Developing and Delivering Resource Based Learning Materials. In: *The Proceedings of The World Conference on Computers in Education*, Birmingham, England. July 1995

[12] White S.A. SCOLAR - A campus wide structure for Multimedia Learning. In: *The Proceedings of the AETT Annual Conference: Designing for Learning*, Glasgow, Kogan Page. 1995

[13] Moore, G. *Crossing the Chasm*. Harper Collins, 1995

[14] Dietinger T and Maurer H.: GENTLE – General Network Training and Learning Environment, Proc. of ED-MEDIA98/ED-TELECOM98, Freiburg, Germany, AACE, June 1998

[15] Carr, L.A., DeRoure, D., Hall, W and Hill. G.J. The Distributed Link Service: A Tool for Publishers, Authors and Readers. *Proc Fourth International World Wide Web Conference: The Web Revolution*, Boston, USA. 1995

[16] Portal Maximizer. <http://www.activenav.com/>. 2001