Embedded Interactive Concept Maps in Web Documents

Rob Kremer and Brian R. Gaines
Knowledge Science Institute
University of Calgary
Calgary, Alberta, Canada T2N-1N4

{kremer, gaines}@cpsc.ucalgary.ca

Abstract: Concept maps are an important means of knowledge representation. People find concept maps intuitive and easy to understand, and they are also amenable to formalization to provide computational services. Concept maps have been used in many fields including education, management, artificial intelligence, knowledge representation, knowledge acquisition, and linguistics. A concept map editor has been ported to the web as a Netscape plug-in where it provides embedded interactive maps within web documents. These maps may be used as an enhancement to web navigation, as a new live annotation form, and as a group interaction tool.

Introduction

Concept maps are an intuitive visual knowledge representation technique. Concept maps tend to be much easier for human users to understand than other knowledge representations such as pure text or predicate logic [Nosek 90]. Concept maps are graphs consisting of nodes with connecting arcs, representing relationships between nodes [Lamibotte 84]. The nodes are labeled with descriptive text, representing the "concept", and the arcs are labeled (sometimes only implicitly) with a relationship type.

Concept maps are used to structure argument forms and express relationships between ideas [Gaines 93]. In education, Novak and Gowin [Novak 84] have promoted the use of concept maps to investigate a student's understanding of a topic, and there are many different forms that have been applied in this field [Lambiotte 84]. In management, Axelrod [Axelrod 76] proposed *cognitive maps* as a means of representing the conceptual structures underlying decision making, and these have been used empirically to analyze organizational decision making [Eden 79] social systems [Banathy 91] and the policies of political leaders [Hart 77]. In artificial intelligence, Quillian [Quillian 68] developed a form of concept maps that came to be termed *semantic networks* and used extensively for formal knowledge representation.

In knowledge acquisition, concept maps are used to elicit knowledge from experts, for example the Wright-Patterson studies of the pilot's associate [McNeese 90]. In linguistics, Graesser and Clark [Graesser 85] have developed an analysis of argument forms in text in terms of structured concept maps with eight node types and four relation types. In the history of science, Thadgard [Thadgard 92] and Nersessian [Nersessian 89] have used the dynamics of concept maps to model processes of conceptual change in scientific revolutions. In the philosophy of science, Toulmin [Toulmin 58] developed a theory of scientific argument based on typed concept maps that provides a model of the rhetoric of Western thought [Golden 76].

Concept maps can represent knowledge on the very formal level, as in Gaines' KRS system, where concept map structures act as a complete interface to a knowledge representation language [Gaines 91] based on Classic

[Borgida 89]. Another formal concept mapping system is the *GrIT* [Eklund 93] system that edits Conceptual Graphs [Sowa 84] and processes them using PEIRCE [Ellis 92]. But concept maps can also represent knowledge on a much less formal level if the system puts fewer constraints on the graphical "syntax". These two factors, intuitive understanding and both formal and informal knowledge representation, are of great importance: Both novices and experts can potentially use the same media. Novices are free to express themselves informally and without constraint, while experts can express themselves under the constraints of very formal semantics which allows for computational support.

Figure 1 shows a typical concept map from an educational domain in which a student has been asked to show the way in which they think about water [Novak 84]. The map has two types of nodes, *concepts* shown by ovals, and *instances* shown by rectangles. These are linked by arrows labeled with relations such as *needed by*, *made of*, *changes*, and so on. The conceptual structure developed encompasses some of the physics and biological roles of water. The student has developed the map within broad guidelines as to what are concepts and instances, and that they are to be linked by labeled directed arrows denoting relations.

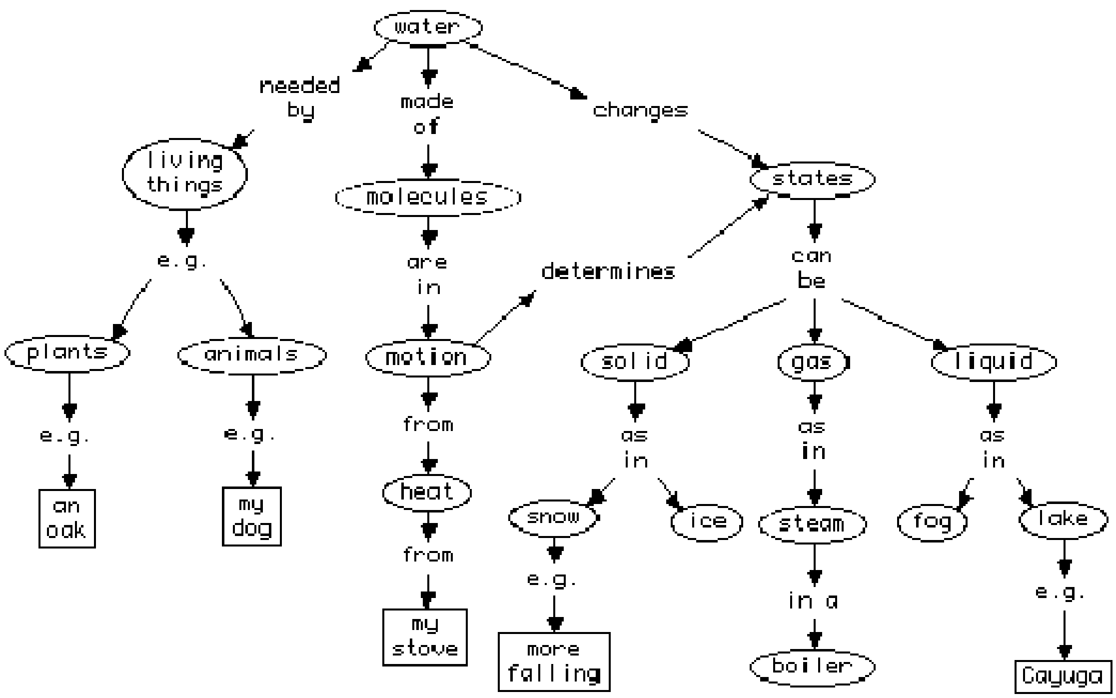

Figure 1: Concept map of student's knowledge [Novak 84]

Concept Map Editing

In spite of the many virtues of concept mapping, and its use in so many fields, it is not ubiquitous. Why should this be so?

The claim has already been made that concept maps are intuitively easy for people to understand and manipulate. One would therefore conclude that concept maps should be common representational media today. But this not the case (at least not outside of specialized circles). The reason for this apparent contradiction is that, before the recent age of powerful graphical workstations, concept maps were difficult and time consuming to produce. Concept maps had to be carefully pre-planned to avoid excessive erasures or recopying when large swaths of the map had to be moved to make room for an unanticipated insertion. Even worse, revisions or extensions to concept maps were usually just as difficult to produce as the original because it usually involved recopying the entire map. Compounding the problem was the fact that large concept maps need to be broken up into a web of composite concept maps, and navigation among these inherently non-linear references was difficult and awkward when imposed on linear media such as books. These problems kept concept mapping techniques confined to static media such as printed media and specialized purposes such as student modeling in education and language analysis. Even these applications had to confine themselves to relatively small conceptual structures.

That was before the introduction of cheap and easily accessible computers with full color graphical interfaces. Graphical workstations remove the labor-intensive task of concept map design. Graphical concept map editing programs such as KDraw [Gaines 91], Kmap, WebMap [Gaines 95a, 95b], and Smart Ideas allow users to quickly slap down ideas (nodes and arcs) onto the workspace without having to carefully pre-plan the layout. These editors allow authors to easily slide concept map nodes (or groups of nodes) around the workspace: the editor takes care of maintaining relationships (arcs) among the nodes. These and other features, allow for easy creation, modification, and copying of concept maps by non-experts. Computer workstations can also be used as a display media for concept maps. This affords easy navigation between related concept maps and other media; user annotation and extension; and user feedback to the author.

Concept Mapping and Hypermedia: Smart Ideas

Smart Ideas [Kremer 93], a program which allows users to draw concept maps with a graphical direct-manipulation interface, has been applied to brainstorming, planning, knowledge elicitation, documentation, presentation, decision support, and meeting facilitation in such diverse areas as research planning, computer networking, marketing, education, oil and gas strategy planning, computer program design, formal language design, academic paper planning, literature analysis, and personal planning. The authors' experience to date has been that most people enthusiastically adapt to reading and understanding informal concept maps (which concurs with [Lambiotte 84]) and readily learn to create and modify concept maps. Kremer has introduced computer-supported concept mapping to individuals who could then independently draw concept maps within five minutes, and to groups using separate workstations with a shared concept map workspace who where collaborating on their topic area within fifteen minutes.

Smart Ideas allows one to draw concept maps using an editor, but every node and link drawn in the map is also represented in an underlying hyperspace which is potentially shared with other users and other concept maps. Figure 2 illustrates the concept. The hyperspace contains "abstract" hypernodes that have hyperlinks between them. In this case, the nodes represent states in a simple communication protocol. The links between them all represent state transitions. Surrounding the hyperspace are several concept maps as they would appear to a user reading or editing them. Each concept map shows some subset of the nodes and links in the hyperspace. The concept map window labeled "The Complete Protocol" contains a view of the entire hyperspace.

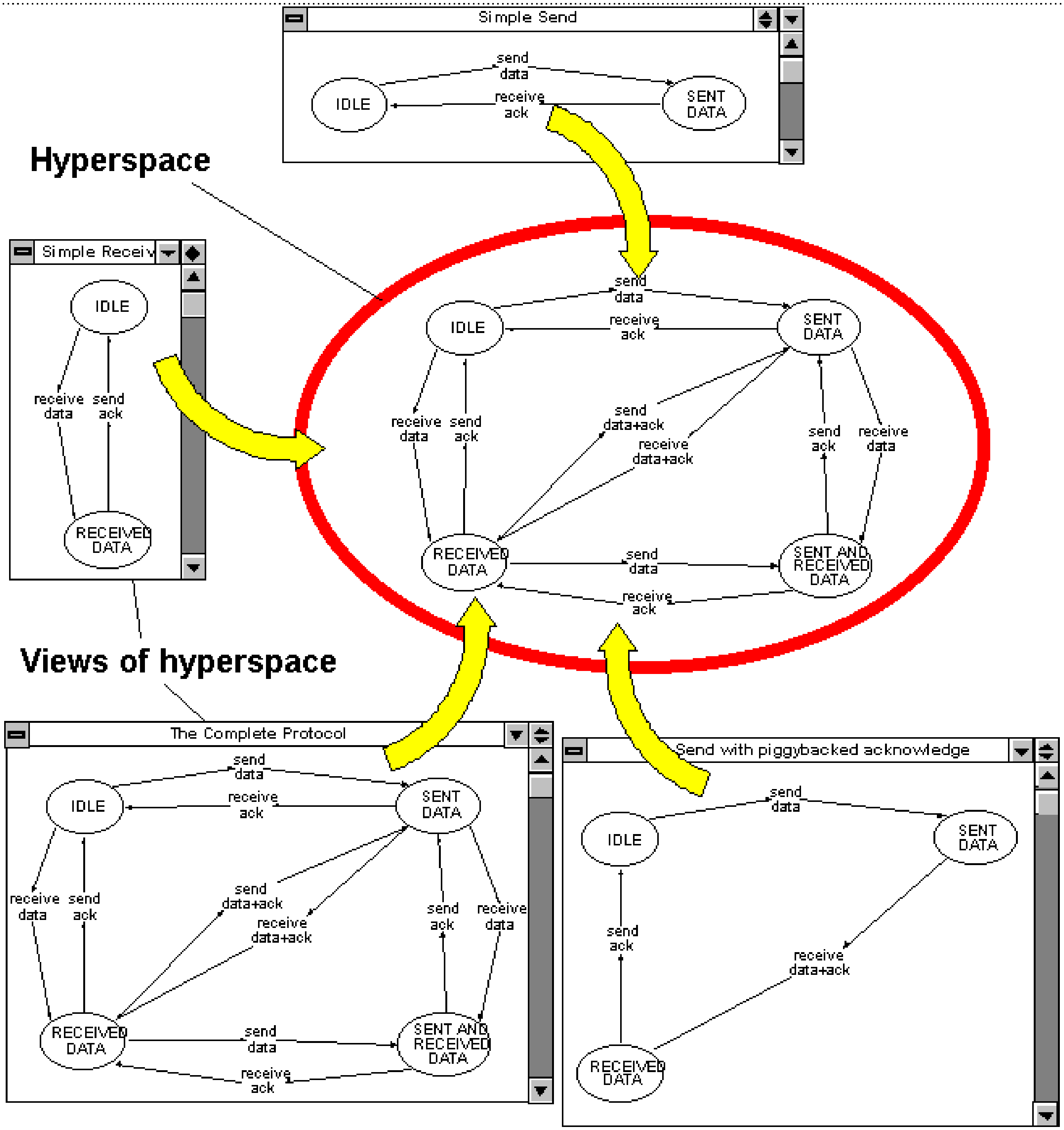

Figure 2: A hyperspace (large thick oval) containing several nodes and links represents a state transition diagram of a simple communication protocol. Four concepts are shown which contain subviews of the hyperspace.

Although Figure 2 shows similar graphical layouts between the concept maps, no such constraint is enforced by the system: users may drag nodes and link labels to achieve any layout they wish; the arcs always remain anchored to their terminating objects.

Whenever a user draws a new node in a concept map, a corresponding hypernode is always created in the hyperspace. Likewise, whenever a user draws a new link between two nodes in a concept map, a corresponding hyperlink is created between the corresponding hypernodes in the hyperspace. Nodes and links may be removed from the view with or without removing them from the hyperspace. Most operations on the concept maps, such as renaming nodes and links, have corresponding effects in the hyperspace.

One advantage of storing all the information in the hyperspace is that one can easily *expand* nodes in a concept map to reveal its neighbors in the hyperspace. An expansion operation on a concept map node looks up each of the hyperlinks that are anchored (either to or from) the corresponding hypernode. For each link, the other anchor node is either found in the current concept map or is drawn as a new concept map node, and the labeled link is drawn between the two anchor nodes. Users may request expansions to proceed recursively to any depth. For example, looking back to Figure 2, a user looking at the "Simple Send" view could perform a two-level expand of either node to have the system draw in all the information in the hyperspace (although the layout may not be optimal).

Smart Ideas uses the nodes in concept maps as a navigational tool. One can merely double-click on a node to hyperlink to an associated document. The document is typically the Smart Ideas hypernode which is represented by the concept map node. This may contain another concept map. Indeed, this answers the question of where Smart Ideas concept maps are stored-they are always stored as the contents of hypernodes in the hyperspace. A hyperspace contains nodes which contain views of the hyperspace itself. It might be pointed out at this time that the information in Figure 2 is a bit rarefied: assuming all of Figure 2's views reside in the same hyperspace, the hyperspace is a bit bigger than advertised, for it must also contain one node for each view:

In addition to creating drawing concepts, Smart Ideas allows users to convey information using text or arbitrary multimedia using the OLE protocol. These multimedia objects may be used either stand-alone or as annotations to concept maps.

Smart Ideas on the Web

Smart Ideas has recently been ported to the web as a Netscape plug-in application. Thus, the concept maps can not only be viewed, but also manipulated and used to navigate the Smart Ideas hypermedia database. Although, at the time of writing, the Smart Ideas plug-in is a read-only version of the Smart Ideas hypermedia, there are many operations available to the user. These include:

- rearranging the map by dragging nodes and link handles,
- double-clicking a node to navigate to deeper detail about a particular concept,
- querying a node about its relationships to other nodes to display all links anchored on that node as well as the related nodes.

Like all plug-in application, a Smart Ideas plug-in can be part of a HTML document (see figure 4) via the <embed> HTML directive, or it can be viewed as a document itself, filling the entire client area of the Netscape browser. To embed a Smart Ideas document in an HTML document one uses the following HTML code:

<embed src="test.sbk" width=500 height=150>

where "test.sbk" is the URL or a Smart Ideas "book" (file) and the width and height values can be any appropriate value in pixels. To view a Smart Ideas document directly one needs only navigate to the appropriate URL with the "application/smartIdeasBook" MIME type.

Every Smart Ideas document has a designated "home page", and this page is opened by default whenever the document is first displayed. From there, one can navigate through the rest of the Smart Ideas hypertext. Navigation is accomplished by simply double-clicking on a node, which causes the Smart Ideas display (whether embedded in an HTML document or as a full size document) to be replaced with the new page referenced by the concept map node. Note that, in the usual case of the node referencing a page within the same Smart Ideas document, this page change does not force a URL access, but is strictly local within the Smart Ideas sub-window: Netscape's back button will not cause the Smart Ideas page to roll back, and enclosing document context is strictly maintained. Since a user may none-the-less wish to roll back Smart Ideas pages, a back button is proved at the bottom of every Smart Ideas window (as part of a caption bar that always displays the name of the current page).

Concept map node references may reference a page within its document, a page in another Smart Ideas document, or any external URL. Unlike the first case, the later two cases will cause a URL fetch to be generated and Netscape will handle the paging as it does with any normal hyperlink event.

This navigation design was chosen to allow Smart Ideas concept maps to be used flexibly as either a web navigational aid (using external node references) or as a "live picture" concept map within a larger context (using internal node references). As an example of concept maps as web navigational aids, Figure 3 is a two-level breakdown of the documents navigable from the "Plug-in SDK" page on Netscape's web site. This concept map gives a quick and clear picture of the structure of the document set and may allow a user to directly navigate to the page of interest without having to traverse, scan, and scroll through all the intervening documents. The node types (as reflected by color and shape - HTML documents are yellow ellipses, GIF files are white rectangles, etc.) also assists the user to zero in on the appropriate document.

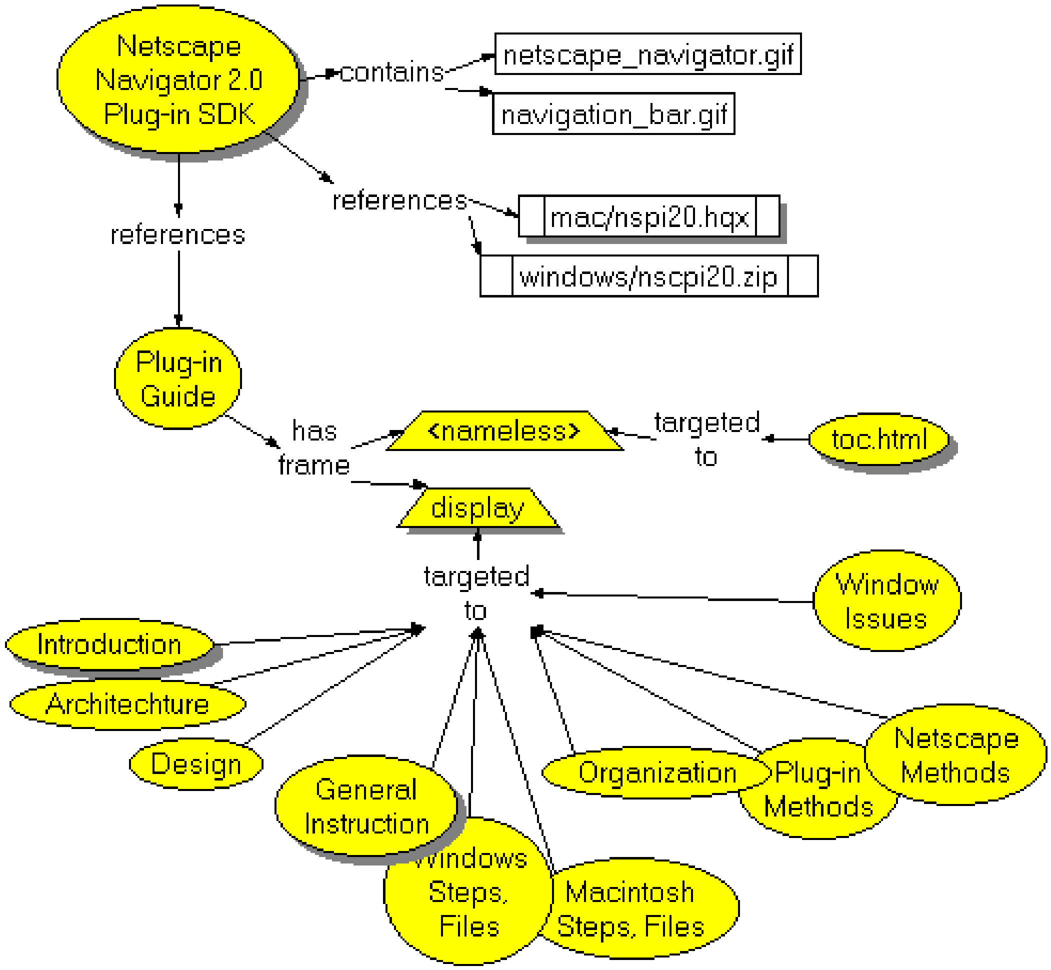

Figure 3: A two-level breakdown of the documents navigable from the Plug-in SDK page on Netscape's web site.

Figure 4 is an example of a live concept map embedded in an HTML document. The HTML document is a description of set of C++ class libraries. The concept map is embedded as a graphic picture of one of the libraries, but users can obtain more detail about any of the concepts in the map by double-clicking on them to replace the overview picture without loosing their context in the enclosing HTML document. The sub-window back button can be used to navigate back to the original view.

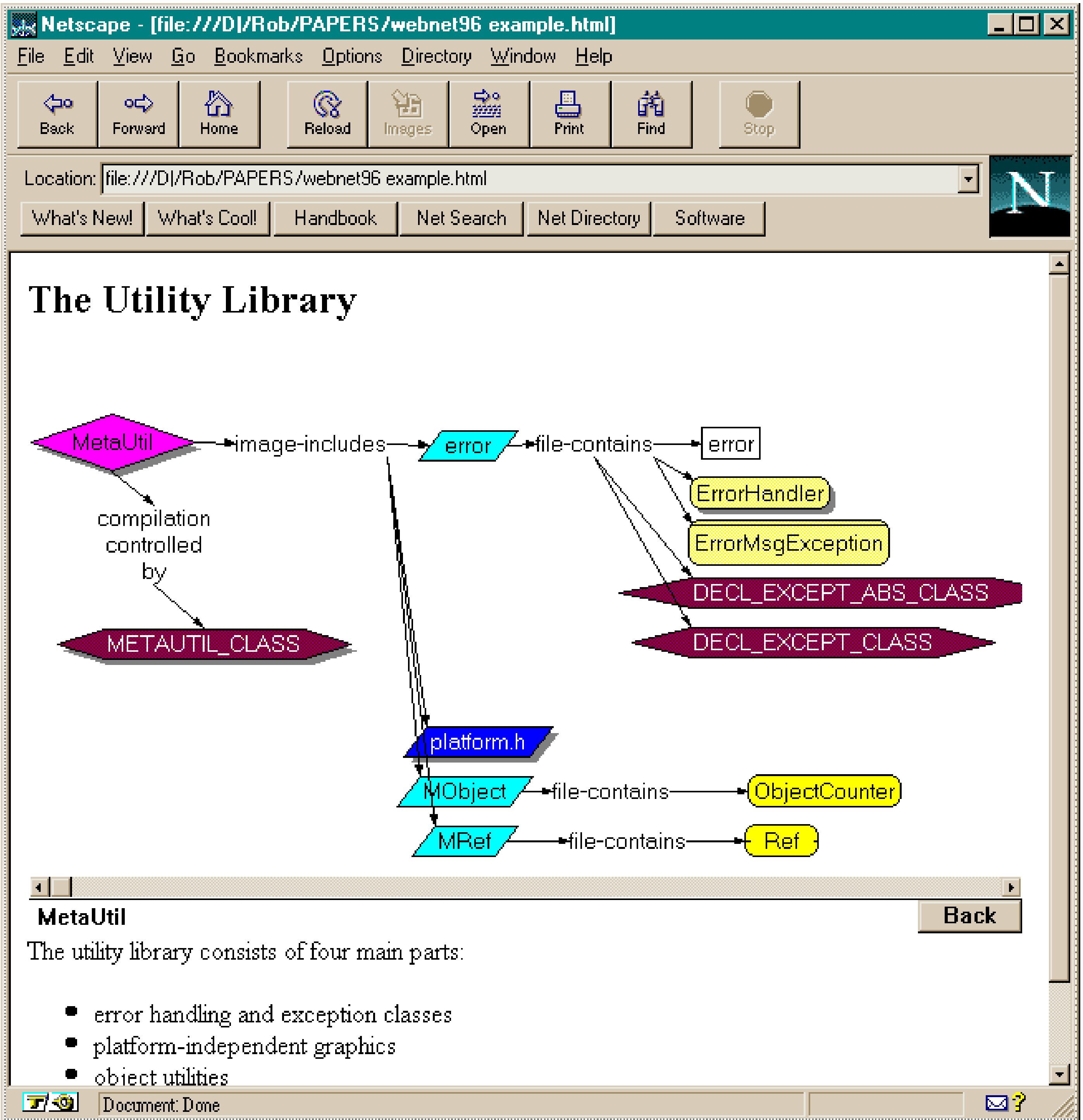

Figure 4: An example of a live concept map embedded in an HTML document

Future Work

Live concept maps on the web suggest several areas of future work.

The web map shown in Figure 3 suggests a tool could be build to automatically generate a concept map given an HTML document as a starting point. The tool would examine the HTML document for contained objects and references to other pages, entering these into the Smart Ideas hypertext and views. Either these maps could be generated and used on-the-fly, or the generated Smart Ideas document could then be edited (using the stand-alone version of Smart Ideas) and used an index into the HTML document set.

The stand-alone version of Smart Ideas encourages users to not only read and navigate concept maps, but also to extend and annotate them. There is no reason why sufficiently privileged users could not do the same using the web version of the program. This would be an excellent tool for distance education and other forms of group interaction.

Once one starts to think of web concept maps as editable, one cannot help but jump to the vision of groupware concept mapping tool on the web. Such a tool might allow more than one user full interactive access to a map with WYSIWIS (what you see is what I see) interaction: Every time one user moved or otherwise edited a part of a concept map, every other user who happened to be viewing the same map would have the update automatically animated on their view of the map. This could be accomplished through the use of special registration servers that the plug-in would automatically query for a list of other current users of the map. Socket connection would then be established between the plug-ins.

Another area of research currently under way is the construction of graph theory which would allow users to constrain the extension of a graph such that a single tool could emulate many of the formal graph theories such as KRS [Gaines 91] and Conceptual Graphs [Sowa 84]. Blending this work with concept maps on the web would be powerful tool which would allow users to develop ideas informally, and then cast these ideas in an appropriate formalism for automated support and development [Kremer 94, 95].

One drawback of the current web version of Smart Ideas it has been ported only to Microsoft Windows environment. Ports to other platforms would be desirable, and one very good way to accomplish this may be to port the code from C++ to Java [Sun 95]. While Java has a few drawbacks (such as slower speed) its portability and automatic loading over the web offers significant advantages over custom C++ code. Porting to Java will be investigated in the near future.

Conclusion

Concept maps are a valuable and important means of knowledge representation. They have significant potential as an enhancement to web navigation, as a new live annotation form, and as a group interaction tool.

References

[Axelrod 76] R. Axelrod, *Structure of Decision*, Princeton, New Jersey: Princeton University Press. 1976.

[Banathy 91] B.H. Banathy, "Cognitive mapping of educational systems for future generations," *World Futures*, vol. 30, no. 1 pp. 5-17, 1991.

[Borgida 89] A. Borgida, R. J. Brachman, D. L. McGuiness, L. A. Resnick. "CLASSIC: A Structural Data Model for Objects." *Proceeding of 1989 SIGMOD Conference on the Management of Data,* New York, ACM Press, pp. 58-67.

[Eden 79] C. Eden, S. Jones, and D. Sims, *Thinking in Organizations*, London: Macmillan. 1979.

[Ellis 92] G. Ellis, R. Levinson, eds. *Proceedings of the First International Workshop on PEIRCE: A Conceptual Graphs Workbench,* Las Cruccs, New Mexico, 1992.

[Eklund 93] P. W. Eklund, J. Leane, C. Nowak. "GRIT: Toward a Standard GUI for Conceptual Structures." *Proceedings of the Second International Workshop on PEIRCE: A Conceptual Graphs Workbench,* Laval University, Quebec, Canada, August 7, 1993.

[Gaines 91] B. R. Gaines. "An Interactive Visual Language for Term Subsumption Languages." *Proceedings of IJCAI-91*, Sydney, Australia, August 24-30, 1991.

[Gaines 93] B. R. Gaines and M. L. G. Shaw, "Supporting the creativity cycle through visual languages," in *AAAI Spring Symposium: AI and Creativity*. AAAI: Menlo Park, California. p. 155-162, 1993.

[Gaines 95a] B. R. Gaines and M. L. G. Shaw. Concept maps as hypermedia components. International Journal Human-Computer Studies, 43(3), 323-361. 1995.

[Gaines 95b] B. R. Gaines and M. L. G. Shaw. WebMap: concept mapping on the web. World Wide Web Journal 1(1) 171-183. 1995.

[Golden 76] J.L. Golden, G.F. Berquist, and W.E. Coleman, *The Rhethoric of Western Thought*, Dubuque, Idaho: Kendall/Hunt. 1976.

[Graesser 85] A.C. Graesser and L.F. Clark, *Structures and Procedures of Implicit Knowledge*, New Jersey: Ablex. 1985.

[Kremer 93] R. Kremer. "A Concept Map Based Approach to the Shared Workspace." MSc. Thesis, University of Calgary, Canada, June, 1993.

[Kremer 94] R. Kremer. "Concept Mapping: Informal to Formal." *Proceedings of the Third International Conference on Conceptual Structures, Knowledge Represetnation Workshop,* University of Maryland, 1994.

[Kremer 95] R. Kremer. "The Design of a Concept Mapping Environment for Knowledge Acquisition and Knowledge Representation." *Proceedings of the Banff Knowledge Acquisition Workshop,* Banff, Alberta, Canada. 1995.

[Lambiotte 84] J. G. Lambiotte, D. F. Dansereau, D. R. Cross, S. B. Reynolds. "Multirelational Semantic Maps." *Educational Psychology Review,* **1**(4), 1984, pp. 331-367.

[Hart 77] J.A. Hart, "Cognitive maps of three latin american policy makers," *World Politics*, vol. 30, no. 1 pp. 115-140, 1977.

[McNeese 90] M.D. McNeese, B.S. Zaff, K.J. Peio, D.E. Snyder, J.C. Duncan, and M.R. McFarren, An Advanced Knowledge and Design Acquisition Methodology for the Pilot's Associate. Harry G Armstrong Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio. 1990.

[Nersessian 89] N.J. Nersessian, "Conceptual change in science and in science education," *Synthese*, vol. 80, no. 1 pp. 163-184, 1989.

[Nosek 90] J.T. Nosek, I. Roth. "A Comparison of Formal Knowledge Representation Schemes as Communication Tools: Predicate Logic vs. Semantic Network." *International Journal of Man-Machine Studies*, **33,** 1990. pp. 227-239.

[Novak 84] J.D. Novak and D.B. Gowin, *Learning How To Learn*, New York: Cambridge University Press. 1984.

[Quillian 68] M.R. Quillian, "Semantic memory," in *Semantic Information Processing,* M. Minsky, Editor. MIT Press: Cambridge, Massachusetts. p. 216-270, 1968.

[Sowa 84] J. F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, Reading, Massachusetts, 1984.

[Sun 95] Sun Microsystems Inc. http://java.sun.com/1.0alpha3/doc/overview/java/index.html. 1995.

[Thadgard 92] P. Thadgard, *Conceptual Revolutions*, Princeton, New Jersey: Princeton University Press. 1992.

[Toulmin 58] S. Toulmin, *The Uses of Argument*, Cambridge, UK: Cambridge University Press. 1958.