PERIODICA POLYTECHNICA SER. EL. ENG. VOL. 45, NO. 3–4, PP. 253–264 (2001)

PROBLEM SOLVING SUPPORT BY THE MULTI-LAYER HYPERMEDIA MODEL

László FARKAS and István VAJDA

Budapest University of Technology and Economics
H–1111 Budapest, Egry J. u. 18, Hungary
e-mail: laci@supertech.vgt.bme.hu

Received: July 5, 2001

Abstract

Hypermedia is a suitable tool to describe knowledge and to support the problem solving process. Hypermedia nodes contain information elements and links represent relations between them. The users of present electronic materials, however, cannot modify the contents of the materials and include their individual knowledge. A new hypertext model, called Multi-Layer Hypermedia Model (MLHM) was developed to support the manipulation of the presented knowledge. With applying this model, electronic materials can be created that express writer's as well as user's knowledge and support the problem solving process. When using HTML as the basic layer of the model, the hypermedia system can expand the usability of HTML based materials.

Keywords: hypermedia, problem solving, hypermedia model.

1. Hypermedia Tools in the Problem Solving Process

Engineering problem solving (EPS) is the main activity of creative people and especially of engineers. Every invention and production requires the solution of various problems. The steps of the problem solving process are the following (JONASSEN, 1996):

1. Perception:
   which part of the invention has unsolved problems that are necessary for successful results
2. Exploration:
   what is the problem, what parts does it have and what is the scientific-technological environment of it
3. Formulation:
   making exact descriptions, formulas and equations that can be solved by EPS methods
4. Finding alternatives:
   evaluating the formulas under different circumstances by using EPS tools and methods
5. Choosing the solution:
   optimizing and verifying the solution by using the previous formulas.

Several software tools, such as TechOptimizer from Invention Machine, can be used in steps 3–5. Knowledge management (KM) tools, for example, database and hypermedia systems, can be used in steps 2 and 4 where the individual knowledge of the human engineer is required in the process. KM tools are suitable to generate and maintain *personal* knowledge.

Hypermedia and hypermedia systems can be used to build and modify knowledge, especially when the personal knowledge has to be created based on *existing* materials (books, textbooks, manuals etc.). A keypoint in EPS process is the availability of easy-to-use *materials that support the EPS process*.

2. Hypermedia as a Tool for Knowledge Representation

Problem solving can be described as a complex thinking process (JONASSEN, 1996) based on the Integrated Thinking Model (*Fig. 1*). Complex thinking (problem solving, designing and decision making) consists of three interacting thinking ways: content, critical and creative. Content thinking includes the processes of learning and retrieving what has been learned. It holds concepts, facts as well as their relations. Critical thinking involves the reorganization of knowledge. It includes a lot of activities (analysis, evaluation and modification) on structures, patterns and relationships. Creative thinking requires going beyond accepted knowledge to generate new knowledge. It uses synthesizing, elaborating and imagining. The three parts are closely connected together in order to originate the problem solving process.

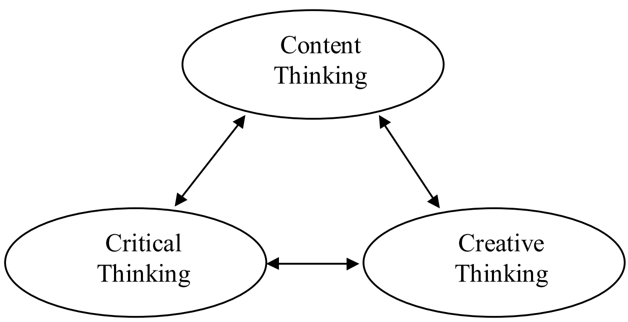

Fig. 1. The Integrated Thinking Model

By using the Integrated Thinking Model, we obtain the knowledge for problem solving consisting of three parts:

$$Knowledge = Information + Relations + Modifications, \quad (1)$$

where information involves objects, facts, concepts and data; relations mean connections and structures (networks, classes, sequences) between information pieces. Structures include different sets of relations.

The active user of the knowledge can expand the information and increase the relations by using different manipulations on them. He/she can also create new structures on the same information set. These modifications are necessary for increasing knowledge.

Hypermedia structures can represent information elements and relations between them. Nodes represent the elements while relations are represented by typed links.

Hypermedia models can be classified into the following five groups (TURINE et al, 1997):

1. Application-oriented models
2. System-oriented models
3. Behavior methods
4. Topological structures
5. Design models

Several application-oriented models were developed to deliver electronic materials. These models, such as HyperText Markup Language (HTML) and Portable Document Format (PDF) (BIENZ and COHN, 1993), give little attention to permit the users to modify the contents of the materials. They cannot make annotations, new links and structures representing their individual knowledge.

Behavior methods and models, such as the Trellis model (SCOTTS and FURUTA, 1989) and Hypermedia Model Based on Statecharts (HBMS) (TURINE et al, 1997) are developed to model the dynamic behavior associated to hyper-document networks and the browsing semantics allowed for navigating through the net. These models, however, do not provide adequate tools for the user/reader of a hypermedia book to *modify and expand the presented knowledge*.

Hypermedia structure itself is not enough to meet the requirements of EPS supporting materials. In order to successfully using it, a hypertext system must have some more characteristics:

- It has to support the modification of the presented knowledge (nodes, links, structures) by the user.
- Special semantic extensions are required to make materials that improve semantic cognition:
  - Structures must be included in hypertext. The structures can be, for example, linear, hierarchical or concept-map like ones (NÜRNBERG et al, 1997).
  - Types of links (e.g. ‘important’, ‘notations’, ‘is-part-of’) must be presented to support semantic navigation.
- Integration of writer’s and user’s knowledge must be supported.

- The system must be divided into a handling part (browser) and the materials (books) (L. FARKAS and I. VAJDA, 1996).
- Online (network) and offline use must be supported.
- It has to be open and
- Has to use some (content or navigational) adaptivity for different users (EKLUND and BRUSILOVSKY, 1995).

Structures and typed links can greatly improve the reception of the knowledge (*Fig. 2*).

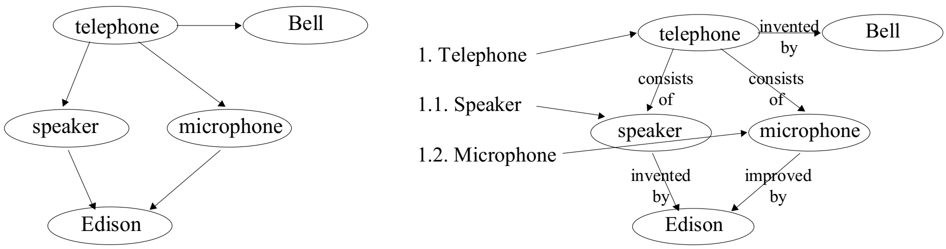

Fig. 2. Knowledge presented in unstructured (on the left) and in structured (on the right) hypermedia

3. The Multi-Layer Hypermedia Model (MLHM)

The basic concept of the model is that we can include additional tools and user activities in the hypermedia structure. Structures, table of contents, indices, guides, as well as user's annotations, structures and links can be handled as new node and link additions to the hypermedia structure. The results of any user's activity that improves his/her knowledge are information and relation sets, which can be presented as new links and nodes. These additions are organized into layers, which hold together the nodes and links of the same functions.

The layers are based on the hypermedia made by the author of the material or book (*Fig. 3*). This basic layer contains the nodes of all information provided by the author and some permanent links, e.g. explanation of words.

The next set of layers extends the nodes and links of one book. These extensions can be the table of contents and some sequential paths through the book as well as the semantic network of the whole material.

The third set of layers provides the connection between the books. These layers can be inter-book links created by the author or by the user.

The last set consists of system-wide layers, for example, bookmarks or automatically generated index containing the index of all books.

The function of the layers is not limited and the layers can be created by the author, the system or the user. The functions are handled by modules of the hypermedia system when the user activates them.

Layers can change during usage to adapt to the user's knowledge. This means that layers can be selected and included in the structure and the content of a layer (nodes and links) can vary during usage.

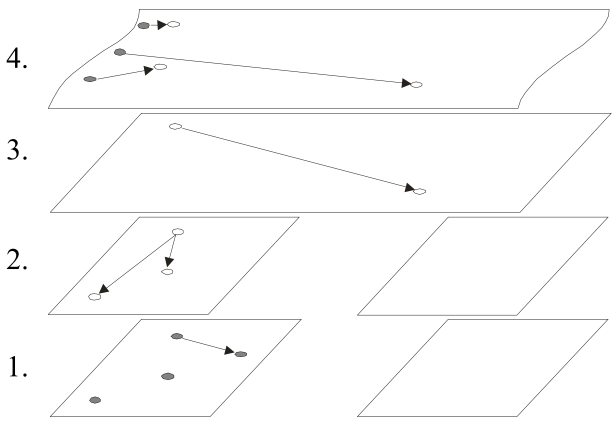

Fig. 3. Multi-layer structure of hypermedia. The layer groups are (from bottom to top): (1) basic nodes, (2) in-book, (3) between-books and (4) system layers

3.1. Categories of Layers

We can classify the layers in the model by different criteria:

1. Placement:
   - Inside a book – e.g. table of contents, paths, guides corresponding to one book
   - Between the books – e.g. inter-book links
   - General – e.g. bookmarks, general index handled by the hypermedia system
2. Variability:
   - Static – created by the author (basic layer)
   - Preferred – selected once by the user, e.g. default path through a book based on the experience of the reader

- Dynamic – created by the user, e.g. bookmarks, which can be added and deleted freely
- Adaptive – changing during usage, e.g. layer implementing adaptive navigation

3. Content (*Fig. 4*):
   - List – ordered list of one node and a corresponding link, e.g. table of contents
   - Set – new nodes added to the hypertext structure, e.g. user annotations
   - Links – only new links, e.g. user-defined links between different books
4. Placement of anchors:
   - Exact – jump-points and end-points appear inside nodes, for example general links
   - Referred – links making connections between whole nodes, e.g. table of contents, local maps

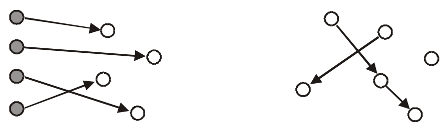

Fig. 4. The layer on the left contains a list and on the right only links (filled circles stand for new nodes)

4. Semantic Tools in the Model

The multi-layer hypermedia model can be used in a wide range of applications. In order to reach our goals – to create EPS supporting hypermedia materials – we have to include other semantic elements into the model.

4.1. Semantic Layers

1. Table of Contents. This layer is an internal list because it connects one Contents element (section) to one hypermedia node. We can separate the main and the additional parts of the textbook: the main part consists of the 'Table of Contents' nodes.
2. 'See also' layer. It connects one node to several others indicating the type of the associations.
3. User links made by the user of the system can represent his extensions, associations and structures.

4.2. Knowledge Types and Levels

A *knowledge type* is a ‘keyword’ summarizing the content of a node. We assign one element of the ‘knowledge type’ set to each node. Types of the links can also be one of the knowledge types or the types of their target nodes. ‘*See also*’ lists can also be sorted by these types. Using knowledge types, we can indicate the semantic network (or concept map) of the material.

*Knowledge level*s indicate which information is suitable for a certain aim or purpose. In practice, levels equal the set of nodes having the same type. Using knowledge levels, we can make levels in the hypermedia structure itself (*Fig.* 5). The first level can be a basic one, while the others can present more and more specialized knowledge. The reader can distinguish the levels based on the types of the links that lead to the nodes.

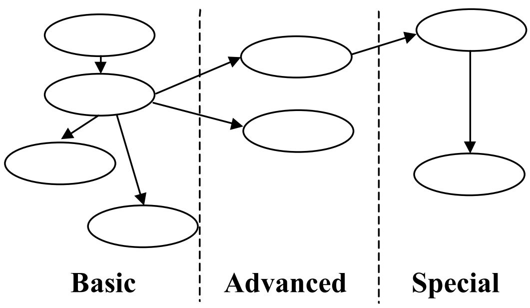

Fig. 5. Knowledge levels in the hypermedia

4.3. Views

A *view* is a selection of hypermedia nodes based on a set of knowledge levels. We can define several views on the same hypermedia. For example, we can define a teacher’s and a student’s view. Teacher’s view includes notices, advises and other educational extensions that do not appear in the student’s view. The nodes of the two views are almost the same but there are nodes that belong only to the teacher view. The actual view can influence the contents of some layers, for example the ‘table of contents’ layer.

Permanent links, that are included in the basic layer, are hard to handle when using views. In some cases these links may change their destination or can be erased in different views. If we use a layer to store the links, then we can change the links more easily.

5. The Minerva II system

5.1. Using HTML as the Basic Layer of the MLHM

The most important hypermedia system is the World Wide Web. WWW uses HTML standard to describe nodes and links. Unfortunately links are stored inside the nodes which make the use of views more complicated. (In the future WebDAV (FIELDING et al, 1998) standard may use separated links.)

If we want to change (or erase) a stored link, we have to change the node that contains it. This means that we have to create a new node with the changed content but in this case the links that point to this node are also changed (*Fig.6*).

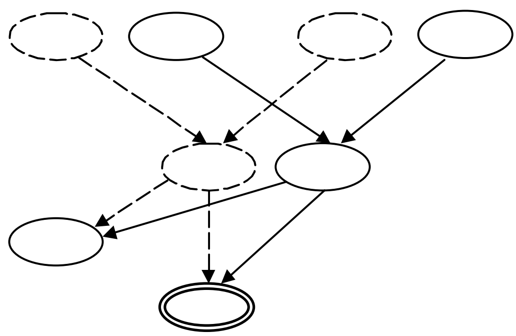

Fig. 6. Permanent links have to be changed when modifying a node or link (double-lined node is changed)

The way of applying a new view has the following steps:

1. The content of a node is changed
2. For every link that points to the changed node: their container nodes have to change: recourse to step 1.

It can be seen that if there are a lot of permanent links, then a lot of nodes have to be changed.

A more convenient way to apply variable links and content is if we use see-also lists. A see-also list is a layer itself, and the content of the layer can be easily changed. One of the main advantages of using layers is the adaptivity of their content.

Another advantage is that they can use typed links, which cannot be implemented in HTML. The third advantage is that layers do not change the content of the basic layer, in this case the HTML pages, only extend its functions.

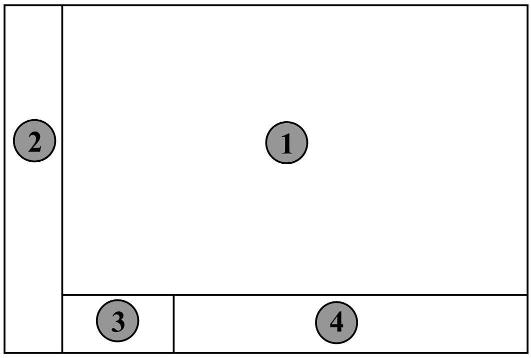

Fig. 7. The architecture of the appearance of the system

5.2. The Application of MLHM

The Minerva II system consists of two main parts (*Fig. 7*): the handler of the hypermedia node (1) and the handlers of the layers (2–4) called *modules*. Modules can be system-wide (2), layers provided by the author for the actual node (3) and layers created by the user (4) for the actual node.

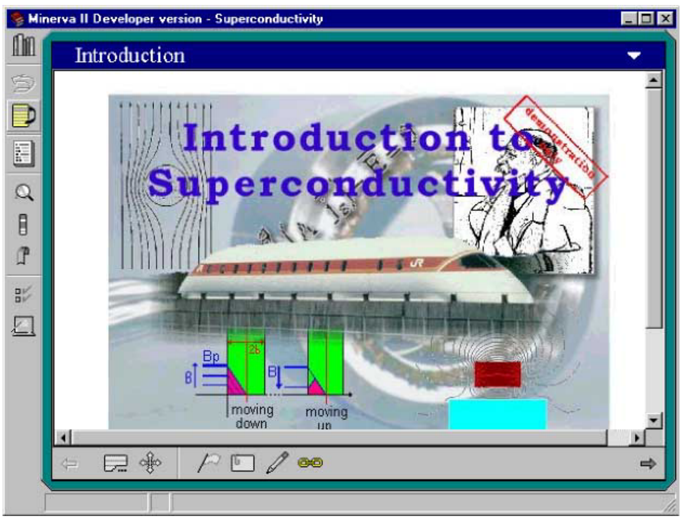

Fig. 8. The Minerva II system

The system itself can be seen in *Fig. 8*. The implemented system-wide modules are the following (from top to bottom): library (available books), back, last

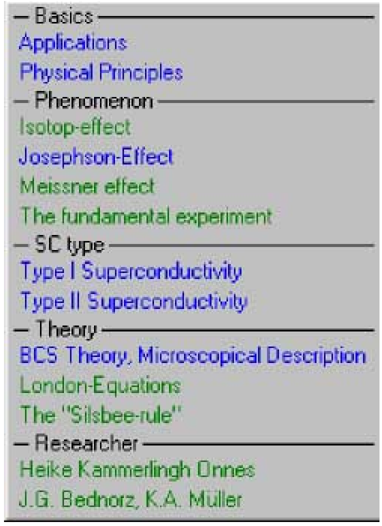

Fig. 9. The list of 'see also' links. The links are arranged by knowledge types

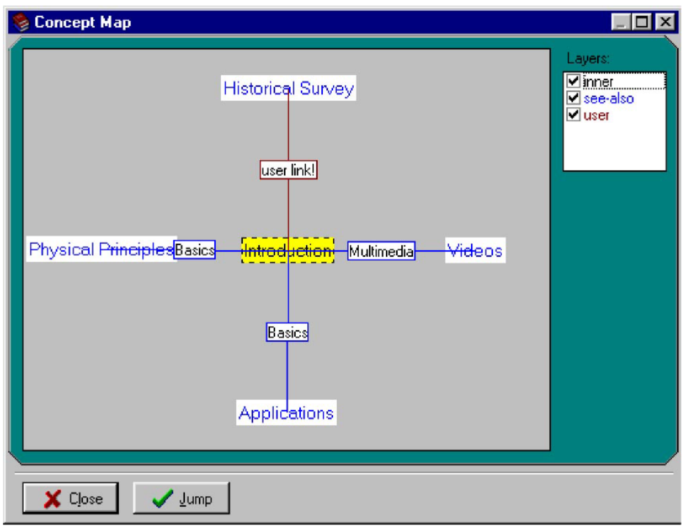

Fig. 10. The concepts map. The map shows relations inside three layers with different colors

visited nodes, table of contents (of the actual view), searching, redline, bookmarks, test and guides. The modules made by the author of the book (from left to right):

previous-next section, see also list and concepts map. The first supports navigation in the hypermedia and the second lists the connected nodes of the current node sorted by knowledge types (*Fig. 9*). The third module shows a map of the relating nodes of different layers (*Fig. 10*). User modules are the following (from left to right in *Fig. 8*): selection of interested information, annotation, highlighting and making of new links.

The hypermedia handler uses Internet standards (HTML, Java etc.) so the creating of hypermedia nodes is the same as creating Internet pages. The separation of the main and additional information is indicated by using different colors: the link to a main topic is blue while the link to an additional topic is green. The knowledge type of the selected link is indicated at the bottom of the system window.

6. Conclusions

Knowledge for problem solving includes concepts and facts, relations and structures and modifications on them. The active user of the knowledge can expand the information and increase the relations by using different modifications on them. Modifications are necessary for increasing his/her knowledge.

The Multi-Layer Hypertext Model helps the manipulation of the presented knowledge. The model has three main properties:

1. It uses layers as holders for nodes and links of different functions.
2. Nodes and links are typed to support cognitive navigation and handling.
3. The model enables different views on the same information set.

With applying this model, problem solving supporting electronic materials can be created that express writer's as well as user's knowledge. These materials can be integrated into a super-hypermedia system that represents the knowledge of the user.

When using HTML as the basic layer of the model, the hypermedia system can extend the usability of HTML based materials without changing the HTML content.

References

[1] BIENZ, T. – COHN, R., *Portable Document Format Reference Manual*, Adobe Systems Incorporated, 1993.

[2] EKLUND, J. – BRUSILOVSKY, P., The Value of Adaptivity in Hypermedia Learning Environments: A Short Review of Empirical Evidence, *Proceedings of the 2nd Workshop on Adaptive Hypermedia and Hypermedia*, Pittsburg, 1998, pp. 13–20.

[3] FARKAS, L. – VAJDA, I., Introducing Superconductivity with Multimedia, *Proceedings of 2nd Japan-Central Europe Joint Workshop on Modelling of Materials and Combustion*, Technical University of Budapest, 1996, pp. 163–166.

[4] FIELDING, R. et al., Web Based Development of Complex Information Products, *Communications of the ACM*, **41** No. 8 (1998), pp. 84–92.

[5] JONASSEN, D., *Computers in the Classroom*, Prentice Hall, 1996, pp. 23–40.
[6] NÜRNBERG, P. – LEGGETT, J. – SCHNEIDER, E., As We Should Have Thought, *Proceedings of the Eighth ACM Conference on Hypermedia: Hypermedia 97*, ACM Press, 1997, pp. 96–101.
[7] STOTTS, P. D. – FURUTA, R., Petri-Net-Based Hypermedia: Document Structure with Browsing Semantics, *ACM Transactions on Information Systems*, **7** No. 1 (1989), pp. 3–29.
[8] TURINE, A. – OLIVEIRA, M. – MASIERO, P., A Navigation-Oriented Hypertext Model Based on Statecharts, *Proceedings of the Eighth ACM Conference on Hypermedia: Hypermedia 97*, ACM Press, 1997, pp. 102–111.