Browsing in a Hypermedia System with Nested Composite Nodes

Débora Christina Muchaluat
debora@inf.puc-rio.br

Luiz Fernando Gomes Soares
lfgs@inf.puc-rio.br

Marco Antonio Casanova
casanova@vnet.ibm.com

PUC-RioInf.MCC23/95 October, 1995

Abstract: Navigating through complex hypermedia documents is not an easy task for users as the number of nodes and the density of links become very large. In order to improve user orientation, it is necessary to provide information about the overall structure of the hyperdocument, showing connections among nodes and mainly details about the navigation possibilities. The implementation of these navigation tools is still a challenge to hypermedia system designers. The main goal of this report is to present some techniques to build graphical browsers for hypermedia systems that allow compositions of nodes. First, we present the basic concepts of hypermedia models with nested composite nodes, showing the importance of navigation tools in order to guide the user within the hyperspace. Then, we present our proposal of a navigation browser that extends the fisheye-view strategy to nested compositions. We also outline an implementation of the browser illustrating the ideas proposed. Lastly, we discuss the major characteristics of some existing hypermedia systems in terms of browsers providing spatial information to users, comparing them to our proposal.

Keywords: hypermedia, composite nodes, navigation, browser, fisheye-view

Resumo: A navegação em documentos hipermídia complexos não é uma tarefa fácil para os usuários principalmente à medida que o número de nós e a densidade de elos se torna muito grande. Com o propósito de melhorar a orientação do usuário, é necessário fornecer informações sobre a estrutura global do hiperdocumento, mostrando as ligações entre os nós e detalhes sobre as possibilidades de navegação. A implementação destas ferramentas de navegação ainda é um desafio para os projetistas de sistemas hipermídia. O objetivo principal deste relatório é apresentar algumas técnicas para a construção de browsers gráficos para sistemas hipermídia que permitem composições aninhadas de nós. Primeiramente, apresentamos nossa proposta de um browser que utiliza uma extensão da estratégia de olho-de-peixe para estruturas com composições aninhadas. Destacamos também uma implementação de um browser ilustrando as idéias apresentadas. Finalmente, discutimos as principais características de alguns sistemas hipermídia existentes em relação a browsers que fornecem informação espacial para os usuários, comparando-os com a nossa proposta.

Palavras-chave: hipermídia, nós de composição, navegação, browser, olho-de-peixe

1 - Introduction

Although hypermedia systems have been developed offering a number of features in terms of power and sophistication, there are still some issues that must be improved. The navigation techniques to explore large hyperdocuments still represent a challenge to designers of hypermedia systems. Parameters like a large number of nodes, a large number of links, many changes in the hyperdocument, bad response time to user actions, insufficient visual differences among nodes and links, and users that are visually disoriented are combined to make the existing navigation mechanisms difficult.

Disoriented users need some kind of information about context and scope to reestablish the sense of location. Basically, there are two types of information that are very important to orient users. One, which answers the question "where did I come from ?", gives the user a sense of temporal context in the hyperdocument. The other, which answers the question "where can I go now ?", gives the user a sense of spatial context in the hyperdocument.

An approach to give a user a sense of temporal location is to save his path when navigating through the nodes and links of the hyperdocument and then show him the path, permitting him to go back step by step or to access any of the nodes already visited in a direct way. This path is usually called a trail. Although trails are very useful when navigating in a hyperdocument, they are not the main goal of this report. Interesting discussions about trails can be found in [Zel89, UY89, MI89, Nie90, Muc95].

The most direct way to give a user a sense of spatial location is to build a map of the entire hyperspace and show the user his position, where he came from and where he can go. This map is called the hyperdocument browser.

The browser is a very important component of a hypermedia system in so far it reduces the so-called "lost in the hyperspace" problem [Hal88]. The role of the browser is to show the structure of the hyperdocument or part of it as a graph, giving an important measure of context and space. This helps the user to know which nodes he is accessing and how they are related to their neighbors in the graph.

When the interconnection network becomes more complex, it is essential to control the number of nodes and links shown in the browsers, which can be achieved by using filters to hide information. In this report, we present a way to build such browsers using the fisheye-view strategy first proposed by Furnas [Fur86], and discussed by many authors [TD92, Glo91, Noi93, SB92]. We propose an extension of this strategy to hypermedia data models that offer nested composite nodes.

This report is organized as follows. Section 2 reviews the basic concepts of hypermedia models with nested composite nodes and the browsing requirements for these nodes. Section 3 gives a brief review of Furnas' well known fisheye-view concept and a detailed explanation of our extension to hypermedia models with nested composite nodes. Section 4 discusses the browsing implementation in HyperProp, the hypermedia system developed in our labs, and illustrates the concepts presented through an example. Section 5 summarizes some related work and compares them to our proposal. Finally, section 6 contains the conclusions.

2 - Composition Browsers

The definition of hypermedia documents is generally based on two familiar concepts, namely nodes and links. There are some conceptual models which permit grouping nodes into composite nodes and also allow nested compositions. Such hypermedia models, by permitting groups of nodes to be organized, hierarchically or not, offer, among other facilities, a better way to help reducing the disorientation problem.

A composite node groups together nodes and links, called components, including other composite nodes, in models that allow nesting. The components may be ordered, which could be very useful in some navigation mechanisms, but they do not necessarily form a set, because an entity may be included more than once in the composite node. Some examples of hypermedia systems with nested composite nodes are HyperPro [Ost92], CoVer [Haa92] and HyperProp [SCR95]. In HyperPro and HyperProp, these nodes are called context nodes and in CoVer, composite nodes.

Specializations of the class of composite nodes can be used in versioning mechanisms. Several systems, in order to address the problem of maintaining the history of a hyperdocument, introduce the concept of version groups. These structures are necessary to represent versions of the same object. Examples of these structures can be seen in the version contexts of HyperProp, the mobs in CoVer and the version groups of HyperPro.

The composite nodes are also useful to model user interactions with hyperdocuments, explained in conformance to the Dexter Model (DM) [HaS90] as follows. By DM, the storage layer contains the stable information, whose instantiations constitute the runtime layer. An instantiation, a presentation of a storage layer component to the user, is the fundamental concept of the runtime layer. Instantiations exist within DM work sessions, as shown in figure 1.

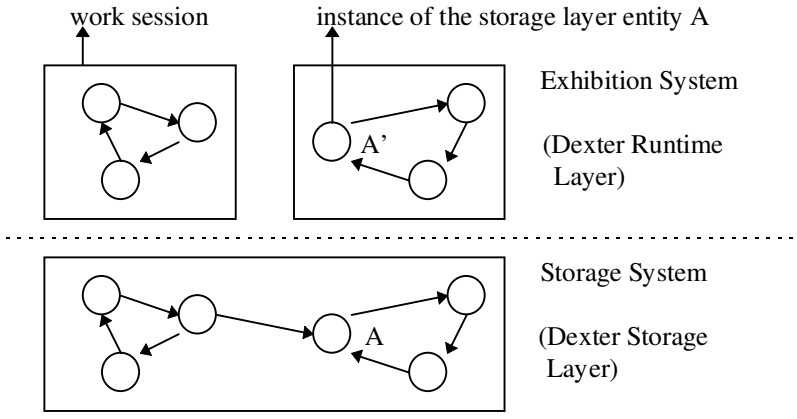


Figure 1 - The Dexter Model for storage and runtime layers

Work sessions and even the entire storage system can be modeled by composite nodes as in several hypermedia systems, including CoVer and HyperProp. Work sessions are modeled in HyperProp by private bases, and the storage system is modeled by the public hyperbase, both of them specializations of context nodes. An instantiation in HyperProp is just a version of a node of the public hyperbase created in a private base. Private bases,

as a specialization of context nodes, can contain other private bases, providing support for organizing a work session into several nested subsessions. The same idea can also be found in the tasks of CoVer.

Within the context of composite nodes, a browser is a map of their components. As a graphical tool for navigation and edition in a hyperdocument, and depending on the purpose of the browser, some peculiar characteristics arise.

If we regard composite nodes with the purpose of seeing just the first level of nesting, we should consider that the node structure should be simple enough to represent it as a pictorial view of all its components, permitting navigation and edition operations. This will be the same case of compositions used to group versions, when we also expect simple structures.

In the case of the storage system seen in all nesting-level details, the corresponding composition will certainly have a complex structure of nodes and links, requiring sophisticated methods to build the browsing tool. Filters to hide information, and possibly land markers (special nodes that always appear) will generally be needed. Fortunately, the complexity of the tool may be reduced, because usually no edition operation is permitted over the hyperbase, just navigation.

In the case of a browser for a user work session to be seen in all nesting-level details, we could have a very complex structure, in which edition and navigation operations are permitted. Thus, there will be a need for a tool with all the characteristics of both previous browsers mentioned.

3 - Fisheye-View for Models with Nested Composite Nodes

The fundamental motivation of the fisheye-view strategy for hyperdocuments [Fur86] is to balance local detail and global context with respect to the node being accessed by the user. Local detail is needed for the local interaction within a hyperdocument. Global context informs the existence of other nodes in the hyperdocument and where they are located, giving a sense of the user's position within the overall structure of nodes and links.

The basic fisheye-view strategy proposes a "Degree of Interest" (DOI) function which assigns to each node a value representing its interest to the user. This function is decomposed in two components. The first is called a priori importance and describes the intrinsic importance of a node regarding the structure of the hyperdocument. The second component determines the distance between a node and the node which is the focus for the user. The essence of the fisheye-view is that the DOI function increases with the a priori importance (API) and decreases with the distance (D).

Since, in hypermedia models with nested composite nodes, we allow different composite nodes to contain the same node and composite nodes to be nested to any depth, we need a way of identifying through which sequence of composite nodes a given node is being observed. This is captured by the notion of perspective of a node [SCR95].

A perspective for a node N is a sequence P = (N_1,...,N_m), with m ≥ 1, such that N_1=N, N_{i+1} is a composite node and N_i is contained in N_{i+1}, for i ∈ [1,m), with m>0. Since N is implicitly given by P, we will refer to P simply as a perspective. Note that there can be several different perspectives for the same node N, if this node is contained in more than one composite node. The current perspective of a node is that traversed by the last navigation to that node.

Hence the degree of interest of the user regarding each node depends on the perspective through which it is being analyzed. The same node will have a DOI value to each of its perspectives. The degree of interest function will indeed be calculated for every perspective of the hyperdocument structure.

To measure the distance with respect to links, we must now observe that a link is also a relation among perspectives, defined as follows.

A link contains a set of source points and a set of destination (or target) end points. Multiple source and destination end points allow the definition of many-to-many connections (many hypermedia models, however, only allow one-to-one connections).

Each of the end points of a link is defined by <(N_k,...,N_2,N_1),A> where N_k,...,N_2,N_1 is a list of nodes and A is an anchor. N_{i+1} must be a composite node and N_i must be contained in N_{i+1}, for all i ∈ [1,k) with k>0.

To describe our browser algorithm, it is first important to define some requirements that must always be satisfied by browsers for models with nested composite nodes to be seen on all levels of nesting detail. A better discussion of these requirements can be found in [Muc95]. The requirements are:

- if a node is shown in the map, its perspective is also shown, so that the sense of composition will not be lost;
- all nodes x directly linked to the focus y are always shown in the map;
- given a link of the form (<(N_k,N_{k-1},...,N_1,x),A>,<(K_n,K_{n-1},...,K_1,y),B>), where y is the focus, the node N_k is always shown and nodes N_k,N_{k-1},...,N_1 have descending priority of exhibition;
- if a composite node is the focus, its components are always shown in the map.

In order to satisfy all the above requirements, we have to extend the fisheye-view strategy, defining the components of the DOI function as follows:

- the a priori importance of a perspective x, denoted by API(x), is given by the negative value of its nesting-level, regarding the most external composite node. The higher the nesting-level of a perspective, the lower its a priori importance;

- the distance between the perspective x to the focused perspective y, denoted by D(x,y), is calculated by the following function:

$$D(x,y) = min(D_c(x,y),D_l(x,y))$$

where $D_c(x,y)$ is the distance between x and y considering only navigation through composite nodes and $D_l(x,y)$ is the minimal distance between x and y considering only navigation through links. If there are no links connecting x and y, the distance D(x,y) is equal to the distance considering only context nodes $D_c(x,y)$.

To calculate the distance considering context nodes $D_c(x,y)$, we use depth navigation from x to y, counting the number of composite nodes accessed, up and down.

To calculate the distance considering links $D_l(x,y)$, we navigate through links from x to y. If there is no direct link from x to y, we should compute the minimal distance considering all the possible paths between x and y. Note that the links are considered to be bi-directional.

For a link with the general format ($<(N_k,...,N_1),A>,<(M_n,...,M_1),B>$), the distance between its end points N_1 and M_1 is given by the expression:

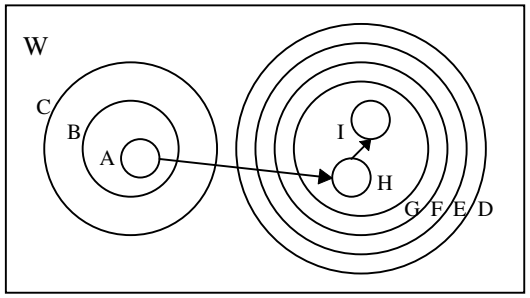
$$D_l(N_1,M_1) = (k - 1) + n$$

The distance between N_1 and M_n is given by k and so on.

Finally, to compute the DOI function for each node x in the hyperdocument, having y as focus, we use the following expression:

$$DOI(x,y) = API(x) - D(x,y)$$

For example, consider a hypothetical composite node W with the following odd structure shown in figure 2. To calculate the DOI function for the node A, having the node I as focus, we have:



API(A) = -3
D(A,I) = min (D_c(A,I),D_l(A,I))
D_c(A,I) = 8 (path: B-C-W-D-E-F-G-I)
D_l(A,I) = 8 (path: B-C-D-E-F-G-H-I)
then, D(A,I) = min (8,8) = 8

as DOI(A,I) = API(A) - D(A,I)
DOI(A,I) = -3 -8 = -11.

Figure 2 - Example of a DOI function

After calculating the components of the DOI function, we can calculate the DOI value for each perspective x depending on the focus y. The information filtering is done by defining a value K which will be a threshold for the DOI function. Only the perspectives which have a DOI value greater than or equal to K will be exhibited in the browser.

If we calculate the DOI function all compositions, we will note that highest value of the function is assigned to the nodes which pertain to the current perspective in focus. This fact already satisfies our first requirement for browsers. In order to satisfy the other requirements, we should limit the maximum value of K, which defines the visible

perspectives in the map. If we consider a generic example of composition, it is easy to prove that the value of K should not exceed the highest value of the DOI function minus 2 units, to satisfy all the requirements stated previously. For a detailed explanation of this condition, see [Muc95].

We should note that it is not convenient to compute the DOI function for all nodes on all nesting levels of compositions all the time. Only the nodes already visited by the user and the ones on the first level of nesting will be considered for the calculation. This limits the scope of nodes and enhances the response time of the algorithm. In addition, the legibility of the map is improved, as the user will only see more details about the nodes that really interest him. The case in which all the nodes are considered when calculating the DOI function is specific. This happens when the user visits all the composite nodes in the browser.

4 - Browsing in HyperProp

This section presents an implementation of the proposed browsing techniques for the HyperProp[1] hypermedia system [SCR95]. As we have seen in section 2, depending on the structure we would like to present to users, we can have different types of browsers. In the case of HyperProp, we have three different types.

The first one is called the context browser and is used to exhibit the first nesting content of a user context node or a version context node. This browser does not necessarily need a filter to hide information because the number of components of a context node is typically small, and we do not want to show its internal details. This tool permits navigation and edition operations over the context node.

The second type of browser is the hyperbase browser and covers the public hyperbase. It incorporates a mechanism for filtering information, based on the techniques described in section 3. This tool permits only navigation operations through the nodes and links.

The third type of browser is the base browser and applies to private bases. It also incorporates a mechanism for filtering information, also based on the extended fisheye techniques. This tool combines the context browser and hyperbase browser characteristics, permitting edition and navigation through the nodes and links of a work session.

In what follows we will present an example of the base browser, which is the most interesting one to illustrate our proposal.

As the goal of the HyperProp project is to build flexible tools, the browsers will allow the user to choose the level of detail that will be presented in the map, defining the number of nodes and links exhibited. Hence either the entire network of interconnections or just what is necessary to give users a sense of location can be visualized.

[1] The work reported in this report was partially financed by a grant from the HyperProp-PROTEM project of CNPq-Brazil.

To build the map of nodes and links, the nested-set notation [Fei88] is used, separating the sense of composition among nodes from the sense of connection between nodes using links. This gives an adequate view of the hyperdocument structure.

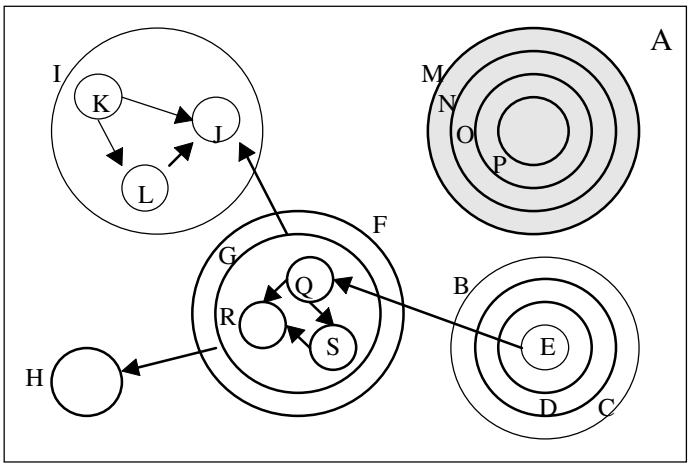
The navigation in the browser can be done by nodes or links, which gives considerable flexibility to users. With the purpose of orienting users about what nodes have already been accessed, all the visited nodes are drawn differently in the map. If the map has a great number of links which could threaten its legibility, the user can choose to hide all the links and visualize just the composition hierarchy.

Since a node may be found in different contexts, it may also be drawn in different positions in the map. On the other hand, if it is shown by one of its perspectives, this does not mean that it will be shown in all the other perspectives.

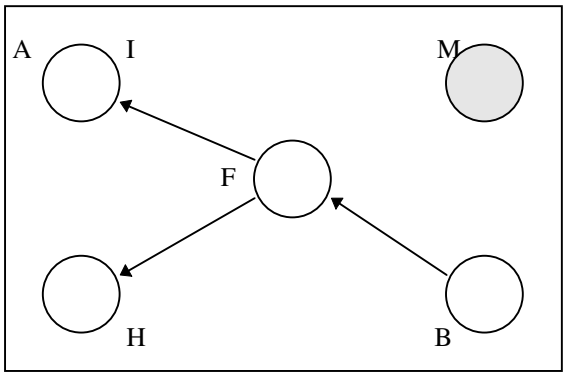
As the position of a node depends on what nodes are shown in the map, the nodes of the hyperdocument do not have a fixed position on the screen. To reduce this problem, the user can define certain nodes as land markers, i.e., nodes that will always be shown in the map, helping the user to improve his sense of location. The set of land markers is specific to each user, and one does not have to agree with the set of land markers of any other user.

As the node structure of the HyperProp Model is defined by composite nodes, land markers will be defined depending on the nesting-level being presented. For example, a node N that is a land marker will be shown if any other node that has the same a priori importance as N is shown in the map. In fact, when a node is defined as a land marker, its perspective is so defined. So, even if the land marker node is not shown, one of the nodes in its perspective will always be shown in the map.

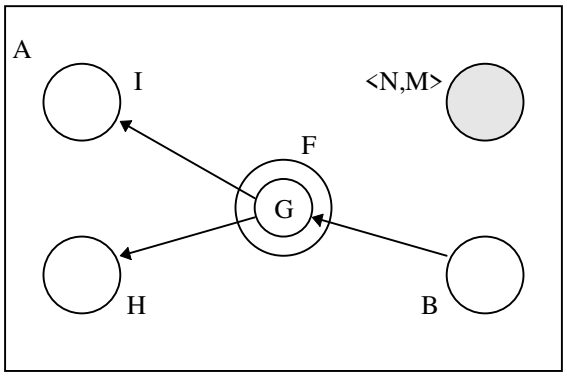
To better illustrate the algorithm implemented by the base browser, figure 3 presents an example of a private base A, showing how the map is modified depending on the focus and on the level of details chosen by the user.



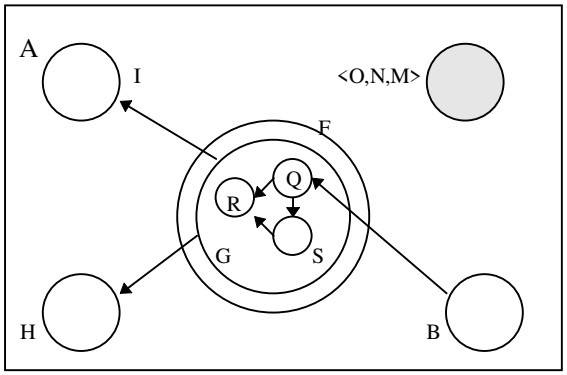
Private Base A. The perspective <P,O,N,M> is a land marker.



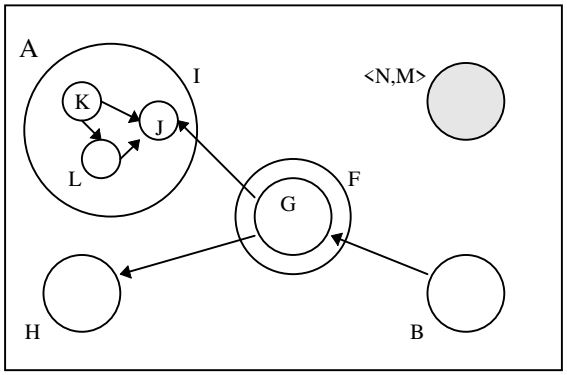
First vision of node A



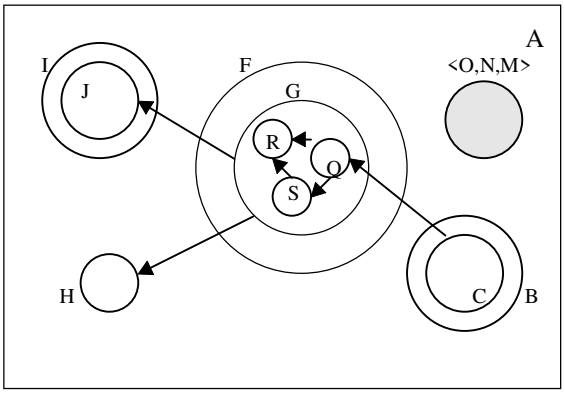
Focus on node F, DOI ≥ -4



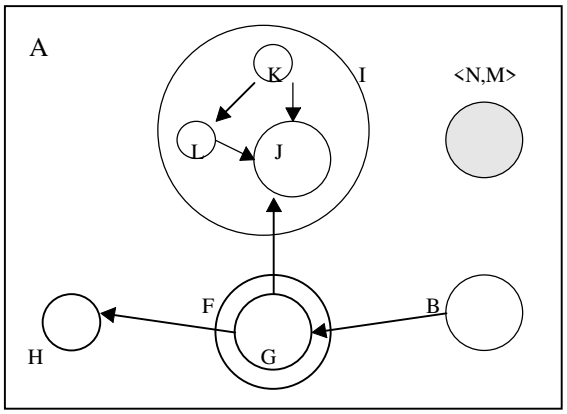
Focus on node G, DOI ≥ -4



Focus on node I, DOI ≥ -4



Focus on node G with more detail, DOI ≥ -5



Focus on node J, DOI ≥ -5

Figure 3 - Example of a hypothetical composite node A.

In the HyperProp Hypermedia System, there are basically two types of links: hyperlinks, which describes structural relationships among nodes and synclinks, which are responsible for the definition of synchronism in the presentation of nodes. The browser features described so far do not consider temporal aspects of a hyperdocument's

presentation, thus, although synclinks are drawn in the diagram, there is no distinction between structural and synchronization connections among nodes. In order to present this temporal relationships, there is another feature of the browser that exhibits the time line of the presentation of the focus node and its related nodes defined by synclinks, allowing users to edit and visualize the synchronization relationships in a hypermedia document. This feature is being developed in another work and is not described in this report.

The browser will offer a complete set of edition and navigation operations for users, allowing easy manipulation of structural and synchronization links. When a new focus node is defined, the diagram of nodes and links will be redrawn and a new time line related to that node will be shown, giving information about the structure of nodes and about temporal relationships among them.

5 - Related Work

Browsing in hyperdocuments has been investigated by various hypertext and hypermedia designers. In general, they do not offer a generic tool that could be modeled to satisfy the user's interest when navigating through the structure of nodes and links. Usually, either the system offers a global map showing a pictorial view of all nodes and links of a hyperdocument, or it shows a local map indicating the user's position, where he came from and where he could go, just presenting nodes that are adjacent to the current node.

Global maps of the hyperdocument structure were proposed in Neptune [CG88], Notecards [Hal88, HMT87] and CYBERMAP [Glo91]. Neptune's graph browser shows a pictorial view of all the structure of nodes and links. Notecards uses very simple filters based on the types of nodes and links. This filter has little effect, since it does not consider the instantaneous context where the user is situated (the present interest of the user). In CYBERMAP, the idea is quite different. The system groups together nodes into another structure called HyperDrawers, by analyzing their content and characteristics and disregarding the links of the hyperdocument. After defining HyperDrawers, the system presents a pictorial view of these dynamic structures. This browser, however, ignores the semantics present in the relations modeled by the links, and the creation of HyperDrawers is usually complex.

Local maps were proposed in the Electronic Document System [YM85] and in the Intermedia system [UY89]. In the former, the local map shows in the center of the screen the current page, on the left side all the pages that the user could have come from, and on the right side all the pages that he could go to. This map shows only nodes adjacent to the current node, limiting the user's view to the local context. In Intermedia, we have the Web View, which combines a local map and a user's trail. It shows the user's path on the top of the screen and, at the bottom, the links and nodes that the user can access having the current node as the source point.

Unlike the previous maps, instead of showing a global map, Thoth-II [UY89] proposes offering one that is updated (expanded) dynamically as a user navigates through the nodes. When a user selects a node, the map is expanded to show all its connections to the other nodes. The problem here is that the browser creates new instantiations of each

adjacent node when a node is selected. When the structure is large, the map becomes illegible.

Building enhanced browsers for complex hyperdocuments is not an easy task as we cannot just present a pictorial view of the hyperdocument structure showing all the nodes and links. This was experienced by other hypermedia systems, such as Intermedia [UY89], when it was concluded that it is impossible to help the user's sense of location with such global maps. A way to solve this problem was proposed in CYBERMAP [Glo91], as previously presented, reducing the number of nodes that would be exhibited to the user. In hypermedia models with nested composite nodes, those groups of nodes are already given by the composition hierarchy and by the existing links. This fact allows the construction of global maps that do not necessarily show all the existing nodes of the hyperdocument, but only those related to a specific point of view.

Thus, in our proposal, we suggest a map which balances local and global information, permitting a user to choose the level of detail being presented. The browsing tool explores the characteristics of nested composition models to guarantee the legibility of maps, hiding or showing components of composite nodes.

The basic fisheye-view strategy proposed by Furnas [Fur86] applies to structures where the components of the DOI function can be easily defined, for example, lists and trees. Other authors proposed some extensions to that strategy trying to solve the disorientation problem of unstructured hyperdocuments [TD92,Glo91]. Our work extends the fisheye model for structured hyperdocuments that allow two kinds of relation among nodes. The first one is given by the nesting compositions and the second uses link connections to relate nodes.

6 - Conclusions

This report discussed browsing navigation in hypermedia systems with nested composite nodes with the purpose of giving the user a better sense of location in the interconnection network. Browsing navigation tools should always guarantee the clarity of maps and maintain legibility, allowing users to choose the level of detail of the information being presented.

We tried to combine the best characteristics of existing browsers, providing a good navigation tool, offering global and local information, and improving the user's sense of location. The final tool set exhibits features which are more general than those offered by the other systems discussed in the report.

The solution proposed takes into account the fact that, in models with nested composite nodes, the compositions usually structure the hyperdocument. Thus, it is possible to build browsers that explore the feature of "exploding" or "imploding" composite nodes, showing or hiding their components, depending on the user's current interest. This feature limits the amount of detailed information presented in the map, improving its legibility.

References

[CG88] B. Campbell and J. M. Goodman, HAM: A General Purpose Hypertext Abstract Machine, *Communications of the ACM*, Vol. 31, n. 7, July, 1988

[Fei88] S. Feiner, Seeing the Forest for the Trees: Hierarchical Display of Hypertext Structure, *ACM Transactions on Office Information Systems*, pp. 205-212, March, Palo Alto, CA, 1988

[Fur86] G. Furnas, Generalized Fisheye Views, *Proceedings of CHI'86 Human Factors in Computing Systems*, Boston, April, pp. 16-23, 1986

[Glo91] P. Gloor, CYBERMAP Yet Another Way of Navigating in Hyperspace, *Proceedings of the Third ACM Conference on Hypertext*, pp. 107-121, San Antonio, Texas, December, 1991

[Haa92] A. Haak, Cover: A Contextual Version Server for Hypertext Applications, *Proceedings of European Conference on Hypertext, ECHT'92*, Milano, December, 1992

[Hal88] F. G. Halasz, Reflections on Notecards: Seven Issues for the Next Generation of Hypermedia Systems, *Communications of the ACM*, Vol. 31, n. 7, pp. 836-852, April, 1988

[HaS90] F. G. Halasz and M. Schwartz, The Dexter Hypertext Refernce Model, *NIST Hypertext Standardization Workshop*, Gaithersburg, January, 1990

[HMT87] F. G. Halasz and T. P. Moran and R. H. Trigg, Notecards in a Nutshell*, Proceedings of the 1987 ACM Conf. on Human factors in Computing*, pp. 45-52, Toronto, Ontario, April, 1987

[MI89] C. Marshall and P. Irish, Guided Tours and On-Line Presentations: How Authors Make Existing Hypertext Intelligible for Readers, *Proceedings of HYPERTEXT'89,* Pittsburg, Pennsylvania, pp.15-26, November, 1989

[Muc95] D. C. Muchaluat, Browsers e Trilhas para Sistemas Hipermídia Atuais, Technical Report, Departamento de Informática, PUC - Rio, Brasil, August, 1995

[Nie90] J. Nielsen, The Art of Navigation through Hypertext*, Communications of the ACM*, pp. 296-310, Vol. 33, n. 3, March, 1990

[Noi93] E. G. Noik, Exploring Large HyperDocuments: Fisheye Views of Nested Networks, *Proceedings of the 5th ACM Conference on Hypertext*, Seatle, Washington, USA, pp.192-205, November, 1993

[Ost92] K. Osterbye, Structural and Cognitive Problems in Providing Version Control for Hypertext, *Proceedings of European Conference on Hypertext, ECHT'92*, Milano, December, 1992

[SB92] M. Sarkar and M. Brown, Graphical Fisheye View of Graphs, *Proceedings of CHI'92 Human Factors in Computing Systems*, pp. 83-91, May, 1992

[SCR95] L. F. G. Soares and M. A. Casanova and N. L. R. Rodriguez, "Nested Composite Nodes and Version Control in an Open Hypermedia System". *IEEE Transactions on Information Systems, Special issue on Multimedia Information Systems (accepted for publication)*

[TD92] K. Tochtermann and G. Dittrich, Fishing for Clarity in Hyperdocuments with Enhanced Fisheye- Views, *ACM ECHT Conference*, pp. 213-221, Milano, Italy, November, 1992

[UY89] K. Utting and N. Yankelovich, Context and Orientation in Hypermedia Networks, *ACM Transactions on Office Information Systems*, Vol. 7, n. 1, pp. 58-84, January, 1989

[YM85] N. Yankelovich and N. Meyrowitz, Reading and Writing the Electronic Book, *IEEE Computer*, Vol. 18, n. 10, pp. 15-30, October, 1985

[Zel89] P. Zellweger, Scripted Documents: A Hypermedia Path Mechanism, *Proceedings of HYPERTEXT'89,* Pittsburg, Pennsylvania, pp.1-14, November, 1989