#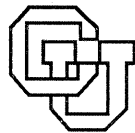 UNDERSTANDING THE ROLE OF STRUCTURE IN INFORMATION FILTERING IN THE CONTEXT OF GROUP MEMORIES: SOME APPLICATION AND USER REQUIRMENTS

Cathleen Wharton & Robin Jeffries

CU-CS-686-93   1993

University of Colorado at Boulder
DEPARTMENT OF COMPUTER SCIENCE

Understanding the Role of Structure in Information Filtering in the Context of Group Memories: Some Application and User Requirements

Cathleen Wharton

University of Colorado at Boulder
Department of Computer Science
Boulder, CO 80303-0430
(e-mail: cwharton@cs.colorado.edu)

Robin Jeffries

Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304
(current e-mail: jeffries@eng.sun.com)

CS Technical Report #CU-CS-686-93

Abstract

Starting in the fall of 1991 several of us at Hewlett-Packard Laboratories in Palo Alto, California examined issues concerning Group Memories and Information Filtering. As a result of this work two papers were produced. The first paper discussed the motivation and foundations for a prototype system named GMM (for Group Memory Manager) which was designed and created to handle an electronic group memory. The second paper discussed the everyday work and cognitive impact of TeamInfo (a second version of the GMM prototype). This report contains the first paper described above.

Understanding the Role of Structure in Information Filtering in the Context of Group Memories: Some Application and User Requirements

Cathleen Wharton

University of Colorado at Boulder
Department of Computer Science
Boulder, CO 80303-0430
(e-mail: cwharton@cs.colorado.edu)

Robin Jeffries *

Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304
(current e-mail: jeffries@eng.sun.com)

Overview

Starting in the fall of 1991 several of us at Hewlett-Packard Laboratories in Palo Alto, California examined issues concerning Group Memories and Information Filtering. Those involved with this project were: Lucy Berlin, Michael Creech, Dennis Freeze, Robin Jeffries, Vicki O'Day, Andreas Paepcke, and Cathleen Wharton.

As a result of this work two papers were produced. The first paper discussed the motivation and foundations for a prototype system named GMM (for Group Memory Manager) which was designed and created to handle an electronic group memory. The second paper discussed the everyday work and cognitive impact of TeamInfo (a second version of the GMM prototype).

This first paper was entitled "Understanding the Role of Structure in Information Filtering in the Context of Group Memories: Some Application and User Requirements." The paper was presented at the *Panel Discussion on Application and User Requirements for Bellcore Workshop on High-Performance Information Filtering Foundations, Architectures, and Applications.* This workshop was held in Morristown, New Jersey, from November 13-15, 1991.

The second core paper produced was "WHERE Did You Put It? Issues in The Design And Use Of A Group Memory." This paper appeared in the *Proceedings of INTERCHI'93* (Amsterdam, The Netherlands, April 24-29, 1993).

Of these two papers, the first has not been accessible to the general public due to the workshop proceedings being limited to only those at the Bellcore Workshop. To make this paper more accessible, this technical report was created. Additionally, material related to the paper at the Workshop is included. This report includes three items associated with the Bellcore Workshop: the initial workshop submission, the final workshop proposal, and the viewgraphs used during the Panel Discussion. Also included in this technical report is a listing of the official citations that should be used.

For more information about the two systems discussed in the above papers, please consult one of the authors in any of the above papers.

---

* Now at: SunPro, Mountain View, CA 94301 (jeffries@eng.sun.com).

Official Citations

For the Position Paper submitted for the Bellcore Workshop use:

- C. Wharton and R. Jeffries. "Understanding the Role of Structure in Information Filtering in the Context of Group Memories: Some Application and User Requirements." Position Paper for *Bellcore Workshop on High-Performance Information Filtering Foundations, Architectures, and Applications*, Morristown, New Jersey, November 13- 15, 1991. University of Colorado at Boulder, Computer Science, Technical Report #CU-CS-685-93, November 1993.

===

For the paper (Extended Abstract) that appeared in the Bellcore Workshop Proceedings use:

- C. Wharton and R. Jeffries. "Understanding the Role of Structure in Information Filtering in the Context of Group Memories: Some Application and User Requirements." Extended Abstract for *Panel Discussion on Application and User Requirements for Bellcore Workshop on High-Performance Information Filtering Foundations, Architectures, and Applications*, Morristown, New Jersey, November 13-15, 1991.

===

For the viewgraphs used during the Bellcore Workshop presentation use:

- C. Wharton and R. Jeffries. "Understanding the Role of Structure in Information Filtering in the Context of Group Memories: Some Application and User Requirements." Presentation for *Panel Discussion on Application and User Requirements for Bellcore Workshop on High- Performance Information Filtering Foundations, Architectures, and Applications*, Morristown, New Jersey, November 13-15, 1991. University of Colorado at Boulder, Computer Science, Technical Report #CU-CS-685- 93, November 1993.

===

Finally, for the paper that appeared in INTERCHI'93 use:

- L. M. Berlin, R. Jeffries, V. L. O'Day, A. Paepcke, and C. Wharton. "WHERE Did You Put It? Issues in The Design And Use Of A Group Memory." In *Proceedings of INTERCHI'93* (Amsterdam, The Netherlands, April 24-29, 1993), Association for Computing Machinery, New York, 1993, 23-30.

To obtain a copy of this paper, contact the Association for Computing Machinery, Inc.

Understanding the Role of Structure in Information Filtering in the Context of Group Memories

Cathleen Wharton
Hewlett-Packard Laboratories and
University of Colorado at Boulder

Robin Jeffries
Hewlett-Packard Laboratories

Keywords

CSCW (Computer-Supported Cooperative Work) • filtering • group (organizational) memory • groupware • information management • querying • structure

1 Introduction

Filtering and querying techniques impose structure upon both information and the way that users view that information. Whereas little structuring can cause too much information to be made available to the user, too much structuring may cause too little information to be returned. Depending upon the quantity and quality of information returned, the user may also need to restructure the information into a more personally usable form. This is because one of two situations can arise: one where the structure is sufficient for a user's goal and one where it is not. It is a case of there being a right or a wrong structure -- of there being good or poor matches of a filter (i.e. of structure) to a user's goals. Although a few tradeoffs are easily identified, there is really no "best" solution: we can move the boundaries, but not find an optimal placement. However, we may be able to find adequate solutions by understanding and classifying resulting tradeoffs. First, familiar and simple examples of structure are given and the role of structure in group memories is discussed. Then our current research into understanding such structuring issues, in the context of group memories, is presented.

2 Information Filtering and The Role of Structure

Structure exists in everything we do. Without structure we would fail at or change the face of many tasks. With structure, tasks can be specialized and often accomplished more proficiently. For example, consider the everyday task of loosening soil in order to sow seeds. Depending upon how much soil you need to cultivate, from a plot to many acres,

the tool(s) you use in your work will make the job harder or easier. If you use a claw, spade, or trowel in an acreage situation, chances are you will give up before you are done. If you choose to use a plow, harrow, or tiller for a small plot, you may find it difficult to position the implement so that only the soil within the plot is turned over. It is a case of matching the specialized tool to the problem size; of understanding what structure is afforded both by land and tool. But you could also use a more generic tool such as a shovel or hoe to do the same task. In fact, if all you have is a shovel, chances are you will use the shovel to do the work. Certainly it is good at cutting or turning over soil. However, it may not provide you with enough control to actually smooth the soil. How well the tool matches your goal may thus be an issue. Use of specialized tools may obviate the need for additional work, but generic tools may yield an adequate solution.

Similarly, when a new tool is introduced into a collaborative working environment, its structure may either be ignored or cause problems. For instance, consider the study by Bullen and Bennett (1990) which examined various CSCW (Computer-Supported Cooperative Work) systems in an effort to understand how groupware functionality and design can influence, or in turn is influenced by, organizational issues. This study discovered a variety of situations where the introduced communication tools forced all users to deal with the same task structure. For instance, all users were required to classify e-mail messages according to type, but not all wanted to do so. Consequently, instead of performing this required step as was intended, many users would just select the first type on the menu without regard message intent. The task structure did not match their needs, intentions, or goals; thus, the users were not benefiting from or making use of the structure provided. They did not want additional commitments or overhead, be they mental or physical. To overcome such problems, a better approach might be the technique used by the Information Lens (Malone, Grant, and Turbak, 1986) or the Strudel toolkit (Shepherd, Mayer, Kuchinsky, 1990); where instead of requiring a certain message type to be specified, such a task step is made optional by way of semi-structured e-mail templates, or by customizable group conversation protocols. In sum, although it may be useful to bring structure to an unsystematic process or environment (Winograd and Flores, 1986), it can also be risky. Additionally, if the structure provided is counterproductive and hence the mental processes that the user must bring to bear are extensive, it can be quite annoying and make a tool less valued. (Other examples of mental processing issues that can arise in information management environments or tools are seen in the semantic processing distinctions of similarly structured nested menus within videotext systems (Young and Hull, 1982) and the food ordering keyboard which promotes various interaction techniques according to skill level or user interface interpretation (Lewis, 1990).)

2.1 Reading Newsgroups

Structure also exists in electronic based information filtering tasks at both individual and group levels. For individuals in a typical information broadcast situation, a filter can be used to organize or prune the information down so it is more manageable. In the case of newsgroups, by subscribing to a particular newsgroup, the main filter is the newsgroup and the information consists of the articles posted to it. Such retrieved material may also reflect a semblance of order due to the traditional method of sorting and presenting articles

according to posting date. Additional structure may also be imposed by the user through the powerful, automated search and filter facilities that can be used within and across newsgroups. For example, if a user subscribes to a newsgroup about databases, chances are the structure defined by the newsgroup name (e.g. relational databases vs. object-oriented databases) will not be enough. Instead, the user will probably need to use additional filters to follow threads of conversations (e.g. new product information). Thus, the newsgroup by itself does not provide enough structure, but the tool which allows you to access the newsgroup may provide the necessary functionality to make the retrieved material better match the user's goals and needs. Other techniques for imposing structure in the news article domain have also been used, e.g. virtual newsgroups for filtering cross newsgroup information (Fischer and Stevens, 1990) and user modelling techniques for predicting preferences (see Allen, 1990).

2.2 Structuring File Systems

Although users may not always be partial to the a priori structure of some tools, many users also do not appreciate being placed in the opposite situation of zero structure. Thus, it is a question of whether the user will actually do the work necessary to put structure in. This is a tricky question, as the various behaviors (below) show. For example, consider a traditional operating system with files and directories (or folders) that does not support a filtering technology whereby all files could be stored at the root level and dynamically grouped (i.e. structured) when needed. In such an operating system you would expect a variety of users to exist: those who have hundreds (perhaps even thousands) of files at the root level without any directories, to those users who have many (and often highly structured, deeply nested) hierarchical directories containing only a few files each. Those users with zero directories (except for root) may even be subdivided further into one of a several camps. There may be people who believe that their files do not have a hierarchical order by nature, some may be new users to the system, or it may be the case that many structures are possible, but instead of investing the time to select, create, use, or rebuild the structure as necessary, nothing happens. For instance, in this last case, suppose you have a collection of memos (stored in files): do you group them according to topic, who sent them, who they were sent to, or by date? Or, do you do nothing? Probably, the structure (or lack of one) that you select most likely will reflect your common tasks, even though that which is a "right" structure and common task now will probably become a "wrong" structure or obsolete task later).

So, how do we determine when a tool, process, filter, or information itself imposes the wrong structure? How do we determine if too little or too much structure is initially provided? This problem is not a curable one. We can adjust the boundaries and determine tradeoffs, but not find a definitive answer. For effective information management to occur, a tool must contain a reasonable amount of default structure, yet be malleable enough to work for both a variety of users and situations. It is difficult to do detailed user modelling analyses to observe and tease out mental processing phenomena. It is difficult to do verbal protocol analyses or to get qualitatively significant data by using the thinking aloud protocol. Thus, we need a way to capture what is going on behind the scenes by logging actual user behaviors in information management situations. Information environments that are highly active, providing a rich set of user situations

and reactions to observe, are ideal. Group work situations seem to be a fertile area for this. But since the user phenomena captured may be many, some control needs to be exerted over the type of information to be examined and managed. For this reason, we have limited our current project to information that would be stored in a group memory.

3 Group Memories as Filtered Information

A group memory (akin to the term "organizational memory" (Walsh and Ungson, 1990)) is defined as a historical memory containing information about a particular project or various aspects of work that a group of people might share. It may be a configuration of both mental and documented kinds of decisions and discussions. For example, a group of people working on a project will often exchange e-mail or share work products (e.g. files). Collectively, for such a group, the e-mail and work products would form a group memory. A group memory is thus a large information store that can be searched, added to, or modified by all of its members. Each member, in turn, has his own view of the memory; a view that is tailored and hence categorized according to his own needs. Commonly, the group memory is instantiated as redundant paper or electronic copies of old e-mail messages and versions of working documents within each person's unique working environment.

Rarely does this kind of a memory exist as a single object (i.e. one global database) to which all users have access. When it does, the memory needs to be treated like a large information store that can be filtered through a variety of means, e.g. personal views of the global store and dynamical and categorical filters (akin to bulletin board -- newsgroup filters) that allow for information to be grouped according to some specification. Thus, the global memory would be something that affords structure. Other basic functionality needed for such a store to be used by its contributors includes a notification mechanism for indicating that new information has been put into the group repository and techniques for clustering or annotating existing information. The two scenarios below highlight some structuring issues that can arise when group memories are used. These scenarios reflect current usage patterns, not those that may necessarily result at a later date.

3.1 The Vacationing Worker

A group member goes away on vacation. While vacationing the other group members use their normal channel of communication (e-mail) to handle issues not addressed during face-to-face meetings, to record the minutes from those meetings, to record key decisions, and to plan additional meetings. The group memory for this group consists of individualized collections of e-mail messages and documents. During his absence the vacationing worker is only kept informed of events and decisions via e-mail. After his vacation, he rejoins the group but is now faced with trying to understand the work done in his absence by reading his e-mail. Consequently, he must establish both a retrospective and current view of the group in order to understand how current events and decisions relate to the work that was both going on before his vacation and done during his absence. Since he only has e-mail, which by default orders his messages by date, he needs

additional, self imposed structure to understand what has happened. For this returning group member, a common way to resolve this understanding problem is to print out every e-mail message on a separate piece of paper and to then filter the papers (and hence the e-mail) using the following process. First, he manually highlights the subject, date, and sender fields of each message. Then he physically sorts and regroups the paper copies by threads of conversations within topics. In effect, he has no electronic means to filter his e-mail accordingly, and the structure afforded by the e-mail tool he uses is not sufficient.

3.2 The Customer Support Engineer

At a customer support center there is a global information store that is used to record a customer's current call, current problem, and call/problem resolution history. The information store (i.e. group memory) contains both raw and distilled notes that may be in English or a foreign language. When a customer support engineer needs to solve a problem by examining similar problems and their resolutions, he looks for them in this group memory. Because it is a worldwide memory, the amount of potentially relevant information may be quite large. Therefore, information filtering is a crucial part to the support services task. Only filtering by keywords is allowed, thus there is not an automatic feature to allow for the filtering of these notes by either native language or note (quality) type. That is, the filter forces a different level of structure and larger granularity than that which the engineer wants or seeks. Consequently, the engineer must do additional, manual filtering.

At another level, the information store does not provide a convenient way to augment a user's view with personal information. The only kind of information that can be added to the store is that which fits into the supported "document" categories; that which can be made available worldwide. Consequently, most engineers often have personal notebooks to augment the information store and use these notebooks often during problem solving situations. This is an example where the online memory is too structured because the user cannot easily annotate the memory with personal notes. It is a case where the engineers need to supplement their memory (or revert to another medium) with a different type of structure.

4 Technical Issues Addressed and Approach Taken

4.1 Research Issues

As the scenarios above indicate, structure can appear in many guises and have profound effects on the way information is filtered. During the course of this project it is expected that a variety of issues will be both addressed and evaluated, but the issues appear to fall into two categories: qualitative and quantitative. Qualitative issues can best be evaluated or understood by interviewing users and having them describe their views and experiences. Quantitative issues can be evaluated by examining user data that is automatically gathered through a logging mechanism. Such a mechanism will capture keystroke level actions, but reconstructing group memory states and situations from

keystroke data, higher, task level actions can also be determined. However, the boundary between a qualitative issue and one that is quantitative is not as always clear cut. Thus, both interviews and logged user data will be used and needed in many evaluation cases. A sampling of key issues to be investigated is given below.

Qualitative Issues

- kinds of filters appropriate for a group memory
- ways that functionality or structure influence the way that a group memory is used or thought about
- amount of time spent maintaining structure versus doing actual work
- similarity of memory structure to user goals
- limitations in functionality

Quantitative Issues

- ways to cluster various subsets of a group memory
- changes in the structure of a group memory
- individualized views of the group memory
- growth or stability of structures
- adaptation to default structures

Finally, as an example of how one quantitative issue (that regarding the "adaptation to default structures") might be gleaned from the logging data, consider the following. By logging how many times particular nodes are linked and relinked as well as the other nodes involved, it should be possible to determine the kind of structures that are working or not for the user. As a simple case, to evaluate whether a user takes a default structure approach when viewing the group memory, the logging data can be examined to see if the individual ever adjusts the memory structure. As a more complicated case, to evaluate whether one type of structure is preferred over another, the logging data can be used to see if the individual either continually uses the same structure or if he does try new structures, whether he regularly reverts to the original one. Of course, understanding regularity and why some structures were good or bad in particular situations would have to come from interview data.

4.2 Research Approach

To understand these issues and in an attempt to investigate and classify the types of structure and information filtering behaviors they afford, this project will be relying upon a new tool, GMM, to manage global group memories. For each group there will only be one memory which all members within a project can access, and the memory will only contain non-mental (i.e. documented) types of information; in particular, information that is online and can be structured using a variety of automated techniques. The memories to be managed will be represented with an existing hypertext system.

In particular, the underlying foundation for the representation and GMM tool is Kiosk (Creech, Freeze, and Griss, 1991), a hypertext system to support software reuse built upon C++ and InterViews (Linton, Vlissides, and Calder, 1989). The Kiosk system is workstation based and supports such basic user functionality as node and link creation and component viewing according to object, functional, or inheritance views. The GMM tool will make all Kiosk functionality directly available to the user and provide additional functionality to support and study common group memory tasks like those listed below.

User Tasks Supported

- Grouping and regrouping of subsets of the memory
- Entering and expressing queries that GMM can use to retrieve particular pieces of information
- Automatic linking of incoming information into the memory according to a default, user, or group defined preference
- Notifying users about new information that has been added to the memory
- Annotation of existing information by a user
- Generation of new mail messages and the editing of existing documents
- Individualized views of the group memory

In an effort to avoid major setbacks that can occur when a new tool (with structure that does not meet a user's expectations) is introduced into a working environment, one philosophy of this project is to err on the conservative side by initially providing only a minimal amount of structure. That is, the default structure will be intuitive and highly similar to what groups are already used to, e.g. chronological orderings from e-mail, groupings according to sender, etc. This is an issue of forcing structure or not. Forcing someone to deal with new or additional structure without providing intuitive functionality or the support of old, existing, or commonly used functionality can be quite risky. New benefits will accrue as the users adopt the system. By mixing the old with the new, the transition should be smoother. But note, even if the structure is minimal and old functionality is available, new or other problems may occur.

As one last piece of the research plan, it is the goal to examine various groups of people who have different types of working relationships. In particular, the goal is to look at the following four types of groups: a paper co-authoring group, a research project group, and two software development groups. One software development group is concerned with creating application software, the other tools. By examining these various groups it is hoped that both a variety of structuring phenomena will be captured and that the GMM tool will become more general purpose.

5 Using the GMM Tool: Some Group Memory Examples

The last section of this paper serves to give examples of how the GMM tool might be used. In particular, the examples used address those concerns raised in the group memory scenarios about the Vacationing Worker and the Customer Support Engineer.

Suppose the vacationing worker has returned and is now trying to understand what has happened while he was gone. Initially, the group memory is displayed as a hypertext structure of nodes and links. The nodes represent new e-mail messages and suppose that the links are used to group messages according to sender. While linking by sender is sufficient on an everyday basis, now in order to put things in perspective, the user thinks it will be more convenient (perhaps after perusing the nodes) to regroup the nodes according to a cross reference field in the e-mail messages to get at the separate threads of conversation. His goal is to automate his manual paper filtering task from before. After doing this he has the option of keeping this new memory structure, reverting to the original one, or having both maintained. Thus, the GMM tool is capable of maintaining an initial memory structure, modifying one, or simply filtering structure when needed; all according to user specifications.

As for the Customer Support Engineer, the GMM tool provides him with the additional functionality which he can use to solve his aforementioned dilemmas. In the first situation, the engineer can now rely upon the GMM filtering mechanisms to additionally group those nodes that originate in the English speaking customer service centers or only those distilled notes that tend to be of more value in a problem solving situation. The engineer is thus able to improve his searches through more structured filters. In the second situation, the GMM tool allows the engineer to freely augment (online) the global store with personal annotations. The annotations can be filtered so that they are invisible to everyone else. Thus, for this case, the information store's structure can be relaxed and some engineer's tasks can be mainstreamed. For some engineers it may even be that the paper notebook medium will go away.

Acknowledgments

Many people have contributed to the organization and origination of this project. In particular, we would like to thank Annette Benjamin, Lucy Berlin, Deann Bucher, Mike Creech, Dennis Freeze, Nancy Kendzierski, Clayton Lewis, Vicki O'Day, and Andreas Paepcke for their time, many helpful ideas, examples, and discussions.

References

Allen, Robert B., (1990), "User Models: Theory, Method, and Practice," *International Journal of Man-Machine Studies*, May 1990, 32, 5, pp. 511-543, Academic Press Limited.

Bullen, Christine V. and Bennett, John, L., (1990), "Learning From User Experience With Groupware," *Proceedings of the Conference on Computer-Supported Cooperative Work*, pp. 291-302, October 7-10, 1990, Los Angeles, California, Association for Computing Machinery, New York.

Creech, Michael L., Freeze, Dennis F., and Griss, Martin L., (1991), "Using Hypertext in Selecting Reusable Software Components," To appear in *Proceedings of the December Hypertext 1991 Conference*. Also available as a Hewlett-Packard Laboratories Technical Report, #HPL-91-59, Palo Alto, CA.

Fischer, Gerhard and Stevens Curt, (1990), "Information Access in Complex, Poorly Structured Information Spaces," University of Colorado at Boulder, Department of Computer Science, Technical Report #CU-CS-461-90.

Lewis, Clayton, (1990), "A Research Agenda for the Nineties in Human-Computer Interaction," *Human-Computer Interaction*, 5, 2-3, pp. 125-143, Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey.

Linton, Mark A., Vlissides, John M., and Calder, Paul R., (1989), "Composing User Interfaces with InterViews," *IEEE Computer*, pp. 8-22, February.

Malone, Thomas W., Grant, Kenneth R., and Turbak, Franklyn A., (1986), "The Information Lens: An Intelligent System for Information Sharing in Organizations," pp. 1-8, *Proceedings of the CHI '86 Conference*, (April 13-17, 1986, Boston), Association for Computing Machinery, New York.

Shepherd, Allan, Mayer, Niels, and Kuchinsky, Allan, (1990), "Strudel -- An Extensible Electronic Conversation Toolkit," *Proceedings of the Conference on Computer-Supported Cooperative Work*, pp. 93-104, October 7-10, 1990, Los Angeles, California, Association for Computing Machinery, New York.

Walsh, James, P. and Ungson, Gerardo Rivera, (1990), "Organizational Memory," Draft of a paper written February, 1990.

Winograd, Terry and Flores, Fernando, (1986), Understanding Computers and Cognition: A New Foundation for Design, Ablex Publishing Corporation, Norwood, New Jersey.

Young, Richard M. and Hull, A., (1982), "Cognitive Aspects of the Selection of Viewdata Options by Casual Users," In M. B. Williams (Editor), *Pathways to the Information Society: Proceedings of the 6th International Conference on Computer Communication*, London.

Understanding the Role of Structure in Information Filtering in the Context of Group Memories: Some Application and User Requirements

Cathleen Wharton and Robin Jeffries
Hewlett-Packard Laboratories*
1501 Page Mill Road
Palo Alto, CA 94304-1126
(cwharton@cs.colorado.edu)
(jeffries@hplrmj.hpl.hp.com)

Keywords: Computer-Supported Cooperative Work • filtering • group (organizational) memory • information management • querying • structure

Introduction

Filtering and querying impose structure upon both information and the way that users view that information. Whereas little structuring can cause too much information to be made available to the user, too much structuring may cause too little information to be returned. Depending upon the quantity and quality of information returned, the user may also need to reorganize the information into a more personally usable form because of a poor mismatch to a user's goal. Although a few tradeoffs are easily identified, there is really no "best" solution: we can move the boundaries, but not find an optimal placement.

We are trying to understand the role of structure in situations where information filtering is used; in situations where the information must have some semblance of order if it is to be useful; in situations where the amount of information is large and continually growing. To do this means that we need to identify application requirements and meet user's needs. The particular application domain we have selected is that of group memories.

Group Memories as Information

A group memory (akin to the term "organizational memory" (Walsh and Ungson, 1990)) is defined as a historical memory containing information about a particular project or various aspects of work that a group of people might share. It may be a configuration of both mental and documented kinds of decisions and discussions; it may contain both electronic and paper media. For example, a group of people working on a project will often exchange e-mail or share work products (e.g. files). Collectively, for such a group, the e-mail and work products would form a group memory. A group memory is thus a large information store that can be searched, added to, or modified by all of its members. Each member, in turn, has his own view of the memory; a view that is tailored and hence categorized according to his own needs.

---

* Cathleen's current address: University of Colorado at Boulder, Department of Computer Science and Institute of Cognitive Science, Boulder, CO 80309-0430.

Commonly, the group memory is instantiated as redundant paper or electronic copies of old e-mail messages and versions of working documents within each person's unique working environment. Rarely does this kind of a memory exist as a single object (i.e. one global database) to which all users have access. When it does, the memory needs to be treated like a large information store that can be filtered through a variety of means, e.g. personal views of the global store and dynamic and categorical filters (akin to bulletin board -- newsgroup filters) that allow for information to be grouped according to some specification. It is this latter kind of memory that we are concerned with.

The GMM Tool

To better understand the issues associated with using and supporting such a memory and in an attempt to investigate and classify the types of structure and information filtering behaviors they afford, we have built a Group Memory Manager, GMM. For each group there is one memory which all members access, and the memory contains only non-mental (i.e. documented), online electronic information; that can be structured and filtered using a variety of automated techniques. The memories managed are represented with an existing hypertext system. In particular, the underlying foundation for this representation and the GMM tool is Kiosk (Creech, Freeze, and Griss, 1991).

A Group Memory Example

The largest memory currently being managed by GMM consists of more than 1000 e-mail messages, 150 nodes to which the e-mail messages are linked, and approximately 4000 hypertext links. The memory primarily contains material that is only a couple of years old, and each e-mail message is saved as a separate file. The group making use of the memory is comprised of four people, one manager and three MTSs, each of whom has made contributions to the memory and been a member of the group for at least one year.

Currently, when a member wants to add something to the memory he submits an e-mail message to the group and then GMM automatically links the e-mail message into the existing group memory structure. Once the e-mail message is automatically linked into the memory using keyword extraction, all group members are informed that new information has been added to the memory. At this point the group member can view or annotate the new information (i.e. e-mail message or corresponding links), and modify any of the existing links or add new ones.

A group member may also use automated techniques to both search through and filter the memory. For instance, a group member can search by keywords, filter information based on its node classification or the information owner, follow multiple conversation threads, or combine any of the above to create even more powerful techniques. Additionally, a group member can save the results of any queries using the above filtering techniques for later reuse and can customize personal views of the memory.

Application and User Requirements

Even though GMM supports a number of filtering techniques and users are capable of using them to structure and get at desired information, determining those aspects of a filtering task that should be handled by the system and those that should lie with the user

is difficult. We have tried to identify the necessary application and user requirements. Our initial findings follow.

User Requirements

For the user to effectively manage a group memory structure and its information (e.g. nodes, links, and e-mail messages), the user needs to be able to do the following.
- Group and regroup subsets of the memory
- Enter and express queries to filter and retrieve necessary information
- Save query results for later reuse
- Customize filters and views
- Access and annotate existing information
- Generate new information

Application Requirements

To meet the users needs above, the application used to manage the memory must minimally provide the following.
- Group memory filters
- Automatic linking of new information
- Customized views and filters
- Functionality to change the core structure and cluster the subsets of a group memory
- A notification mechanism

Early Results and Workshop Plans

We have recently put GMM into operation and are now having it manage the group memory described above. Consequently, we have yet to gather and harness the necessary usage data to do a thorough quantitative analysis of the system. For the workshop, however, we expect to have enough usage data and experience with the system to discuss preliminary findings.

References

Creech, Michael L., Freeze, Dennis F., and Griss, Martin L., (1991), "Using Hypertext in Selecting Reusable Software Components." To appear in *Proceedings of the Hypertext 1991 Conference*, December, 1991.

Walsh, James, P., and Ungson, Gerardo Rivera, (1990), "Organizational Memory." Draft of a paper written February, 1990.

Understanding the Role of Information Filtering in the Context of Group Memories: Some Issues in Design

Cathleen Wharton

Hewlett-Packard Laboratories and
University of Colorado at Boulder

and

Robin Jeffries

Hewlett-Packard Laboratories

Background Information

- Discussion based on
    - -- Work in progress
    - -- System design and implementation issues
    - -- First group of users selected and
      beginning to use it (exploratory phase)

- GMM (Group Memory Manager)

- Group memory: *a collection of artifacts containing information about a particular project or various aspects of work that a group of people might share*

Characteristics of a Group Memory

- Large repository of information

- Information is diverse (i.e., many topics)
  e.g.,   X Windows
          Portable computing technologies
          ABC design notes
          Restaurants
          Medical data and analyses
          Conference announcements

- Lacking a consistent structure

- Comprised of a variety of components
  e.g.,   paper documents
          e-mail messages
          minutes from a meeting
          workproducts
          assorted reports

- Something that can be searched, added to, or modified by all group members

The GMM Tool

- Automatically manage group memories

- Based on an existing hypertext based system, Kiosk, designed to support software reuse

- Manages electronic (i.e., online) information

- Hypertext links connect nodes of information

- Classification nodes used to combine various nodes into substructures

- Automatic linking based on keywords, conversation threads, etc.

- Assorted filtering and querying techniques

- Supports individual and group views

Example

- Memory size and structure for first group

    -- 1100 e-mail messages

    -- 150 classification type nodes

    -- 4000 hypertext links

Presentation

Understanding the Role of Structure in Information

Page 6 of 21
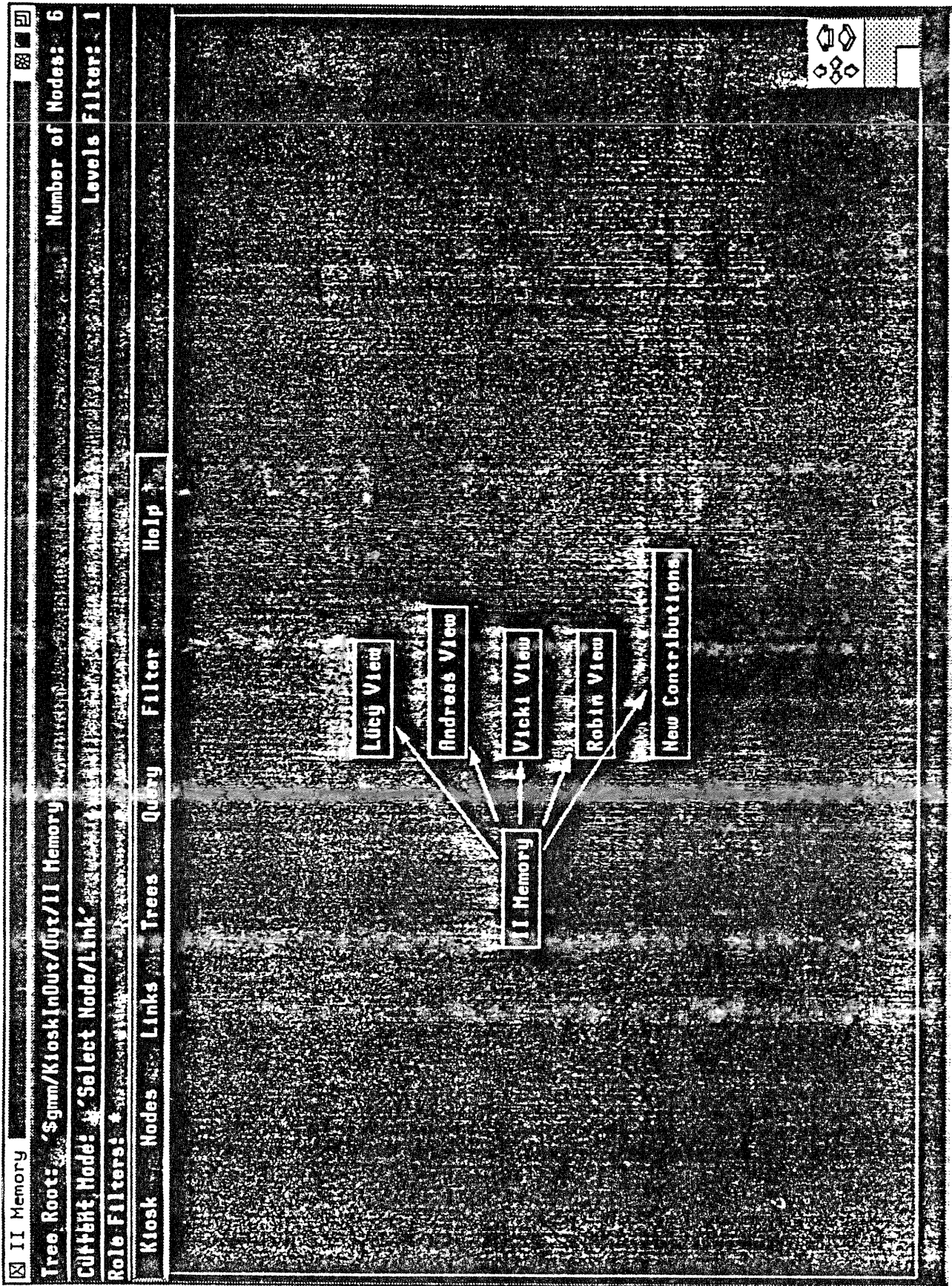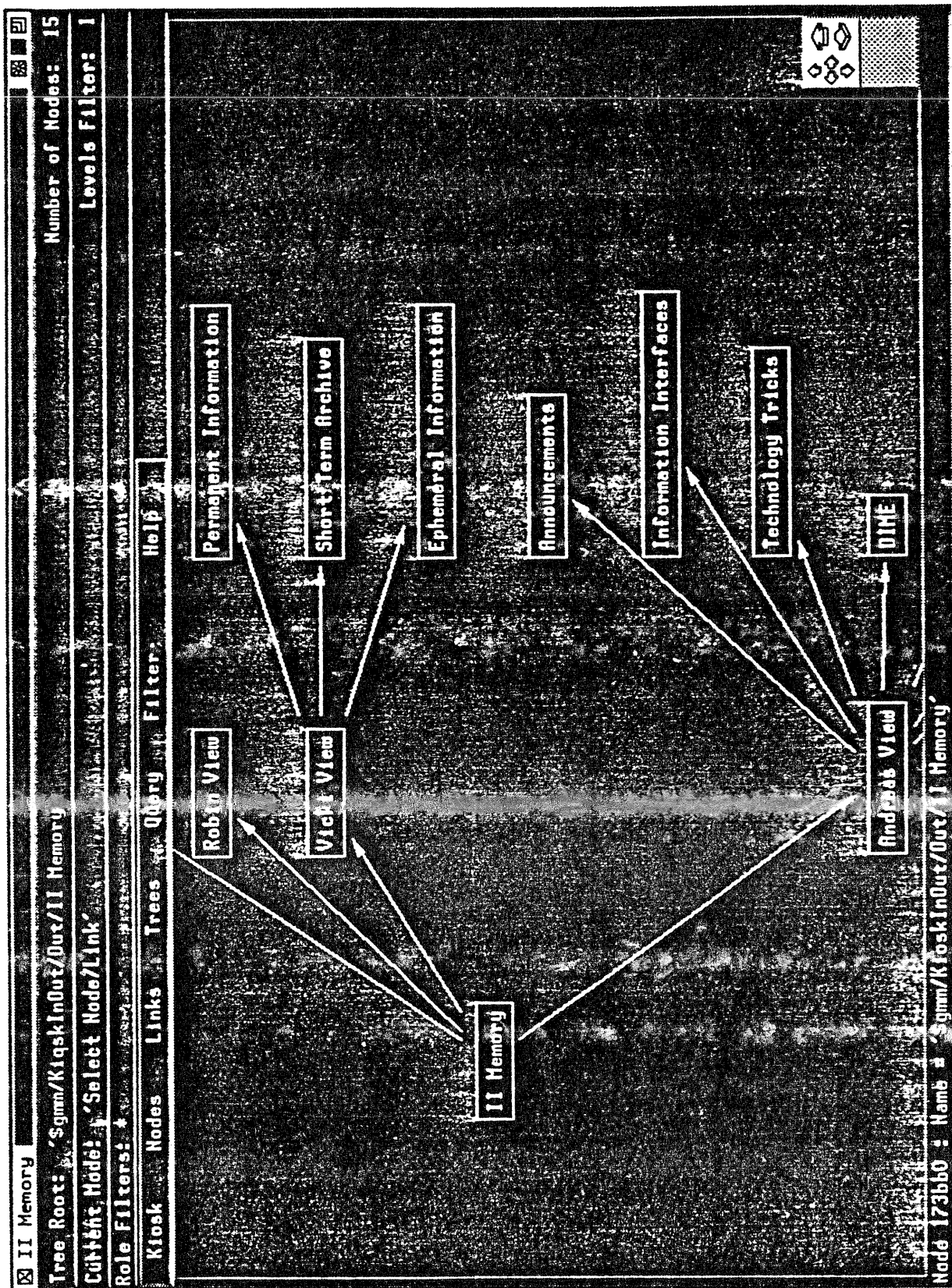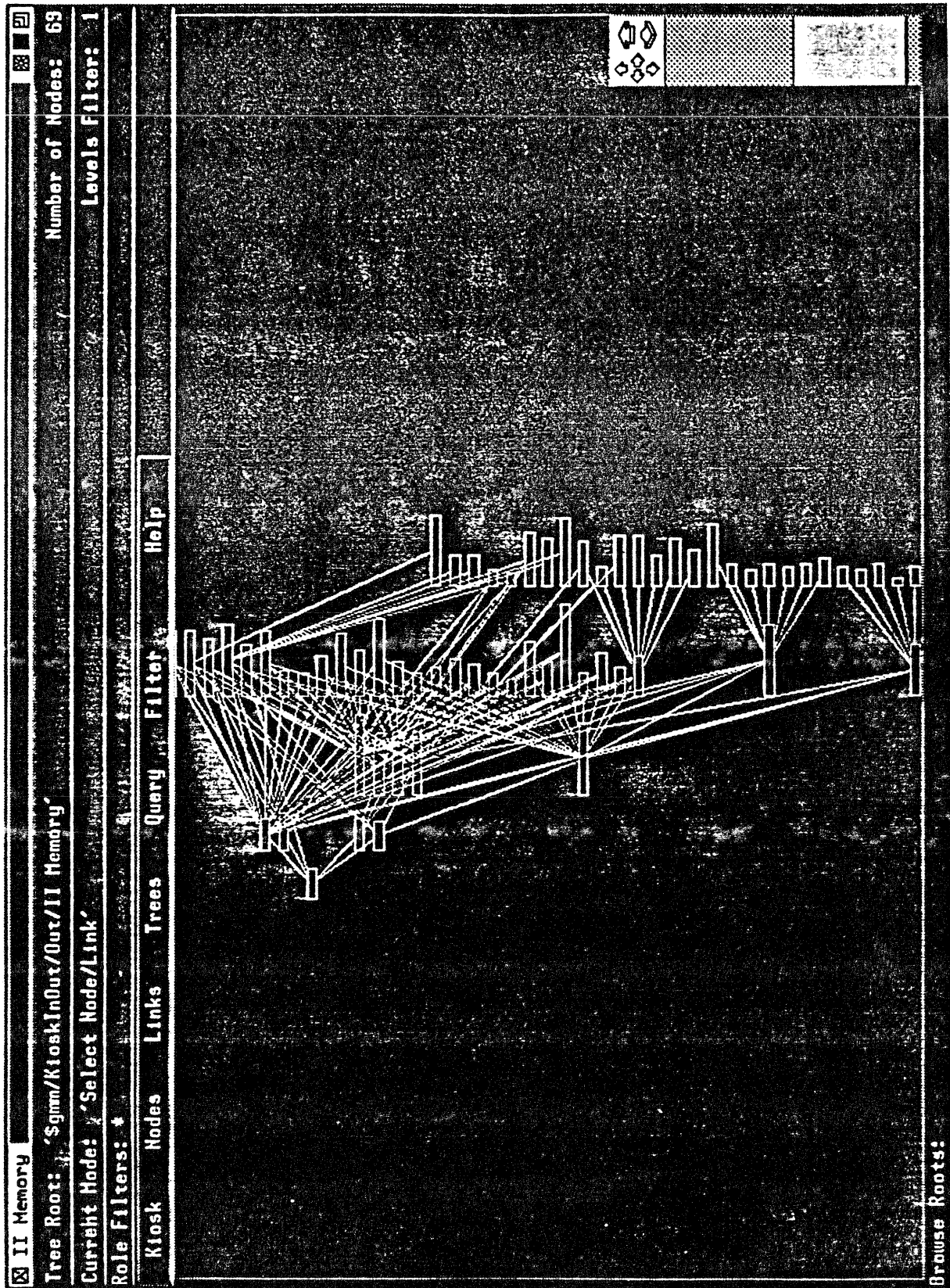
Automatic Linking

- ## Mail Messages

------- MESSAGE CLASSIFICATION INFORMATION -------

REPLY TO MESSAGE NUMBER:

FROM:

KEYWORDS:

MESSAGE TYPE:

NODES TO LINK TO:

- ## Classification and Component Nodes

ALIASES: group memory, GMM, organizational memory

SEARCH FOR: group memory, GMM, organizational memory

☒ LinkPresenter

LINK PRESENTER for 'hplhoop.14235.688586484.264945'

Owner : Cost++

Role   : consists of

[THESE VALUES CANNOT BE EDITED!]

Source : $gmm/KioskInOut/Out/Announcements
Dest   : $gmm/KioskInOut/Out/Talks

[COMMIT accepts new values.]
[SOURCE/DEST implicitly commits and traverses.]
[UNDO restores old values. CANCEL exits.]

Commit   Source   Dest   Undo   Cancel   Help

Node Presenter: $gmm/Contributions/636   (unmodified)

LinkFilter: ALL

Kiosk   Nodes   Links   Collections   Filter   Help

3. What is the lowest cost way to get a picture of information flow and
work organization over the product cycle (I'm sure Janice needs this
too)?

4. Can we come up with questionnaires or other low cost ways of
collecting info (e.g., about informal communications)?

------ MESSAGE CLASSIFICATION INFORMATION ------

MESSAGE NUMBER: 636

REPLY TO MESSAGE NUMBER:

FROM: II Outsider

KEYWORDS: info needs, CPCD

MESSAGE TYPE:

NODES TO LINK TO: Product Development

Role: "info needs"   Local @ 2775
Role: "ii outsider"   Local @ 3332
Role: "info needs"   Local @ 3355
Role: "cpcd"   Local @ 3367
Role: "cpcd"   Local @ 3367
Role: "product development"   Local @ 3407
Role: "product development"   Local @ 3407

<< "Information Needs" @ 112
<< "II Outsider" @ 0
<< "Information Needs" @ 70
<< "Collaborations" @ 78
<< "CPCD" @ 57
<< "Collaborations" @ 62
<< "Product Development" @ 0

$gmm/KioskInOut/Out/Information Needs

Node Presenter: $gmm/KioskInOut/Out/Information Needs   (unmodified)

LinkFilter: ALL

Kiosk   Nodes   Links   Collections   Filter   Help

Name: Information Needs

ALIASES: info needs, information needs

SEARCH FOR: info needs, information needs

```
Role: "consists_of"        Local @ 0   << "Andreas View" @ 0
Role: "consists_of"        Local @ 0   << "II Projects and Concerns"
Role: "information needs"  Local @ 0   >> "753" @ 2350
Role: "information needs"  Local @ 0   >> "752" @ 956
Role: "information needs"  Local @ 0   >> "751" @ 4279
Role: "information needs"  Local @ 0   >> "750" @ 1700
Role: "information needs"  Local @ 0   >> "749" @ 2974
Role: "information needs"  Local @ 0   >> "748" @ 1108
Role: "information needs"  Local @ 0   >> "747" @ 5716
Role: "information needs"  Local @ 0   >> "746" @ 1419
```

GMM Basics

- Add to the memory by cc:ing an e-mail alias

- Incoming e-mail automatically linked into memory

- Group members perform various filtering and retrieval tasks using memory

Common User Tasks

- Filter memory (i.e. group/regroup subsets)
    e.g., Follow multiple conversation threads
         Find all information submitted by Joe
         about ABC design issues

- Enter and express queries to search for and
  retrieve necessary information

- Save query results for later reuse

- Customize filters and views

- Directly access and annotate existing information

- Generate new information

Core GMM Functionality

- Automatic linking of new information
  - e.g., -- Contributor
    - -- Keyword
    - -- Conversation thread

- Mechanisms for filtering and querying
  - e.g., -- Regular expression based queries
    - -- Equivalence database based queries
    - -- Nodes to categorize information
    - -- Personalize style parameters
    - -- Filter by link type or owner
    - -- Filter nodes according to content

- User creation and deletion of links and nodes

- Annotation and direct editing of information

- Multiple users

- Notification (?)

QueryBox

* QUERY EDITOR *

○Exprs Enter regular expr: installation|
○Paths Enter search path: $IIContributions]

[Checkboxes toggle global settings!]

☐ Recursive Descent?
☐ Search Link Files?
☐ Case Sensitive?
☒ Result?[Find all matches]
● perl ○mgrep [PERL based search]
○File ◆Sentence ○Line [Granularity]
☐Pass [Intersection of all sets]

[Invoke by choosing presenter type (Node/Tree).]
[UNDO restores old values. CANCEL exits.]

Node   Tree   Undo   Cancel

⊠ Filter for 'II Memory' ▦ ▦ ▣

Filter for 'II Memory'

LINK ROLES:

Enter Role: ◁Empty▷
○ Select from Preferred Menu
Select from Browser (double-click):

```
┌──────────────────────┐ ⬆   Select Mode:
│                      │     ☐ Set Role
│                      │     ⊠ Add Role
│                      │ ⬇   ☐ Remove Role
└──────────────────────┘
```

LINK OWNERS:

Enter Owner: ◁Empty▷
○ Select from Preferred Menu
Select from Browser (double-click):

```
┌──────────────────────┐ ⬆   Select Mode:
│                      │     ☐ Set Owner
│                      │     ⊠ Add Owner
│                      │ ⬇   ☐ Remove Owner
└──────────────────────┘
```

NODE TYPES:

Enter Type: ◁Empty▷
○ Select from Preferred Menu
Select from Browser (double-click):

```
┌──────────────────────┐ ⬆   Select Mode:
│                      │     ☐ Set Type
│                      │     ⊠ Add Type
│                      │ ⬇   ☐ Remove Type
└──────────────────────┘
```

NODE SEARCH TEXT:

Enter Text: ◁Empty▷

LINK/NODE DIRECTIONS:

☐ Any	⊠ Circular (intra-node)
☐ Inbound	☐ Internal (ValueLink)
⊠ Outbound	☐ No Relation

LATTICE LEVELS:

Enter Number of Levels: ▮

[COMMIT = save   UNDO = cancel changes]

(Commit)   (Undo)   (Close)

The Role of Information Filtering --
Some Design Issues Influencing GMM

- What does it mean if we have an evolving group memory structure?

- Should a group memory have only one view (the group view) or multiple, personal views?

Structure of the Group Memory

- Lacks a consistent organization because information
  - -- is in various formats
  - -- is diverse
  - -- is large
  - -- is always growing
  - -- tends to come in spurts
  - -- can become "dated"

- Memory structure is constantly changing
  - -- No "best" or "single" organization
  - -- Tasks and needs change with time
  - -- Multiple contributors
  - -- Ways of thinking changes with time

Personal vs. Group Views

- Only one view -- the group view
    - -- Everyone has same view -- "group solidarity"
    - -- Everyone agrees to place the information
      into some acceptable group organization

- Multiple views -- one per group member
    - -- Incompatible views
        - e.g., Conferences vs. Call for Papers
    - -- Tailorability
        - e.g., Filtering preferences
          Interface presentation style
    - -- How make information accessible/known?
        - -- WHAT was put in the memory?
        - -- WHERE it is?
        - -- WHO put it in?
        - -- WHEN it was put in?

Research Issues and Plans

- Issues

  - The two issues previously discussed
    -- Memory structures
    -- Personal vs. group views

  - Other factors
    -- Usage of GMM
    -- Ways a group memory is used,
       viewed, and evolves
    -- Ways group interactions and work
       patterns, are modified or enhanced
    -- Archival memories or not and notification

- Plans

  - Different working groups
    -- research vs. development vs. co-authoring