Active Notebook: A Personal and Group Productivity Tool for Managing Information

Mark C. Torrance
MIT Artificial Intelligence Laboratory
545 Technology Square #705
Cambridge, MA 02171
and
Sun Microsystems Laboratories

Abstract

This paper describes Active Notebook, a collection of software tools which work together to allow individuals to label document-based information with conceptual classifications, and organize these documents into a semantic taxonomy for later browsing and retrieval. Tools are also provided to support sharing documents and references with peers. Novel techniques for taxonomy sharing avoid the need for a single, global taxonomy to which everyone adheres. Ongoing work on active reminding is described.

1 Introduction

More and more, we find ourselves in the information management business. This is especially true for people working on research, but it is apparent from reading popular press in other fields that the problem is pervasive. The number of commercial software tools for organizing and retrieving information is growing at a rapid pace.

We introduce Active Notebook, another tool in this category, which takes as its goal to make it easy for users to gain high level semantic control over their organization of information. By applying natural language technology, and introducing a simplified representation of taxonomically related concepts, we aim to provide the tools to put users in control of their documents.

2 Storing Documents

The Active Notebook depends on an easily accessible source of documents. When we began implementing the notebook tools in May, 1994, the World Wide Web (WWW) had just begun to experience the phenomenal growth which is now so often reported. We chose to implement our Active Notebook on the substrate of the World Wide Web, to take advantage of the growing volumes of information that are available in formats readable by any WWW browser. Any document that can be referenced with a URL can be included in an Active Notebook. Furthermore our tools work through a proxy server and a set of CGI scripts without modifying either the WWW browser or the server, so they can be used with the user's present Web environment and favorite tools.

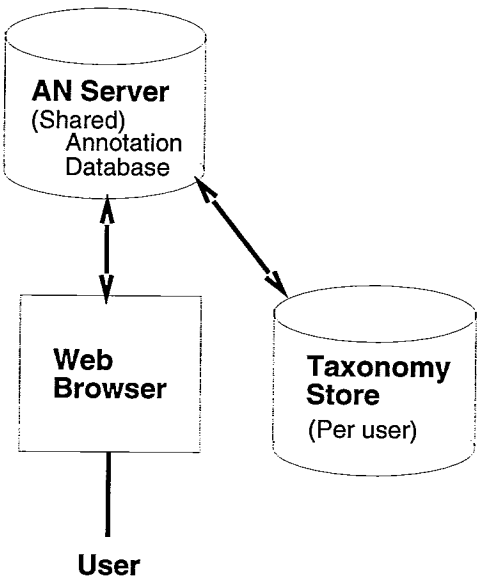

Figure 1: Programs comprising Active Notebook

The Active Notebook representation of a document includes a URL, a Title, and any number of concepts. The URL is a Uniform Resource Locator; a unique identifier describing how to retrieve the document from any computer on the Internet. The title is, as you would expect, a brief description of the content of the document. In the case of an HTML document, this is usually the document's declared title, though it can be overridden by the user. In the case of other documents, it is specified when the user creates the document's "paper homepage", described below. The concepts describe the categories the user has placed the document into; these are described in more detail in the next section.

Active Notebook is not only used for organizing collections of information written by others; its most useful feature is the ability to relate personal notes and comments about other documents to those documents. We support this by implementing an annotation server which keeps track of declared relationships among documents on the WWW. This annotation server is implemented as an HTTP proxy server which sits between the

user's WWW browser and the rest of the Web (labelled **AN Server** in Figure 1). Whenever a page is requested by the browser, the proxy returns the page with additional material appended at the bottom. This includes a number of hyperlinks to other web-accessible documents which annotate the current page. These annotations are first class HTML documents; they may themselves be annotated in order to structure a discussion.

The annotation server may be shared by several Active Notebook users; these users may choose to make their annotations visible to others who share their annotation server on a document-by-document basis. They may also define defaults which specify their usual permission settings.

The Web incorporates many documents which are not in HTML, and there are still other documents, such as books, which are not available in any machine readable format. Our solution to this problem is to introduce the "paper homepage", an HTML page which contains relevant information about a document in a non-HTML format. In the case of a hardcopy paper, for example, the paper homepage you create might include the title, author, abstract, and any other relevant citation information. Ideally, these homepages would be maintained by the paper's author or publisher, but in lieu of that they may be created on an individual basis. These homepages provide a representation of the paper for purposes of the Active Notebook, offering an HTML point of reference which can be annotated, attached to other HTML documents as an annotation, and assigned concepts in the Active Notebook as described in the next section. Paper homepages for non HTML documents can be easily created through an HTML form; users can include minimal reference information, and can return to the homepage to add to the information later.

In order for the Active Notebook to be a useful tool, it needs to be extremely easy for users to add documents and notes to the notebook. We have implemented several simple techniques for adding documents, including an OpenLook based drag and drop interface that prompts the user for concepts. This interface includes a simple editor for online creation of new notes, and accepts text files dropped on it from any OpenLook compatible drag source, such as the File Manager, the Mail Tool, or the Textedit editing program. Defaults are selected for the title and URL of the new document based on information parsed out of the document. A second, command line interface to this program may be used by itself or as the backend for a scripting command. We are developing a set of Emacs keyboard macros which will facilitate adding documents to the Active Notebook.

These interfaces all share a common code base that provides preprocessing for the documents as they are added to the notebook. Mail messages, for example, are processed by Hypermail; software which creates an attractive threaded HTML archive of a number of standard mail messages. Calendar appointments included in mail messages are placed on the user's calendar; the Sun Calendar Manager is supported and work is underway to support Ical. These calendar appointments are extended with the URL of the mail message as it was placed in the user's Active Notebook, so the user can later get from the appointment to the extended description of its topic.

These Web based tools for storing documents also make it easy to share documents with others. Facilities in the Active Notebook interface make it easy to create new groups of users and to share documents you create with members of a selected group. The annotation proxy server makes sure to display annotations only to those users who are members of groups who can view the annotation. A second level of protection is provided by permission settings on the underlying document files and access control documents placed in their directories, used by their HTTP server to control access to the files themselves. Finally, a user can choose, while browsing, which authorized groups to "be a member of"; this can be used to focus the set of displayed annotations on documents of interest to a topic at hand. Future work will introduce standard interfaces for sharing references with others so that documents others think you should read will be introduced with a default priority and concept you choose into your todo list.

The document and note structuring tools by themselves provide a useful array of services, but the real benefit of Active Notebook comes from semantic organization of concepts assigned to documents by the user.

3 Assigning Concepts

Each document in a user's Active Notebook can be assigned one or more concepts. Concepts are simply phrases composed of one or more words which describe the document or its importance to the user. Concepts are presumed to be topic or category descriptions. One concept for this paper, for example, might be "Research Organization Tool". These concepts are arranged by the user into a taxonomy expressing the semantic subsumption relationships among the concepts, described in the next section.

The author has used Active Notebook for ten months during various stages of its development, mostly for organizing references to information available on the Web. My Active Notebook currently contains 150 documents, and includes 238 different concepts.

One entry in my active notebook, for example, contains this information:

```
URL       http://www.exploratorium.edu
TITLE     Exploratorium Home Page
CONCEPTS  Science Museum, Education
```

One problem with storing concepts in this free-text way is that concepts which are effectively the same may be stored in different ways. This could happen if they were capitalized differently when stored, such as "Graphics software and "graphics software", or if they are minor morphological variants of each other, such as "tool" and "tools". We have implemented a solution to

the first problem which applies a canonical title capitalization to concepts both as they are being stored and when they are used in a query. We are working with Bill Woods to incorporate genuine morphological analysis to handle the second problem as well.

4 User-Structured Taxonomy

The user's taxonomy expresses the semantic subsumption relationships among all concepts used to describe any document in the Active Notebook. So, for example, my taxonomy includes the facts that a *computer* is a kind of *information processing device*, and that *commerce* is a kind of *business activity*. These are stored as relationships in a semantic network, with *computer* a child of *information processing device*, for example. The parent-child relation here represents a deliberate conflation between "is an instance of" and "is a kind of"; we find that for the types of concepts used in practice, the system is more usable if this semantic distinction is not enforced. Some facts are assumed implicitly by the system; at present, multi-word concepts are assumed to be "a kind of" the class named by each of their component words. So *information processing device* would have (at least) three parents: *information*, *processing*, and *device*. While this algorithm introduces some strange axioms, the general effect is positive, and bad axioms can be manually deleted using the Active Notebook taxonomy editor.

In [3], the authors of AGENDA identify many benefits of allowing individual users to structure their own personal taxonomy or hierarchy of categories. Since that time, substantial effort has been spent on designing broadly useful taxonomies, such as CYC [4] and Wordnet [5], and on standards such as KIF[1] and Ontolingua[2] for sharing knowledge among different users and applications. We still find that for personal information organization needs, a personalized taxonomy provides distinct advantages, though it may be worth exploring the extent to which knowledge represented by a user in this relatively unstructured taxonomy may provide a basis for more formal knowledge representations.

A personal taxonomy can contain greater levels of detail in areas which represent parts of the world where the user makes finer distinctions. My taxonomy, for example, makes significant distinctions under the concept "software", where the taxonomy designed by a person who is not a professional in this field would be unlikely to make these distinctions. Conveniently, as a user's perspective changes over time, previously monolithic categories can be split into finer distinctions, and Active Notebook includes an interface for recategorizing collections of documents *en masse*.

Some problems which must be handled by a globally consistent taxonomy are minimized or avoided by a personal taxonomy. Individuals often use words or concepts in only one sense, even when there are, strictly speaking, several different senses for that word. This happens because people frequently work within a domain vocabulary that constrains the sense they mean. So where a taxonomy based on Wordnet might include five or six senses for the word "point", a user might only use it in one or two senses, and the taxonomy will appear simpler if unused senses are excluded.

Despite the simplification introduced by restricting the taxonomy to a single user, there will still be some occurrences of the multiple word senses problem. In my taxonomy, for example, I discovered I had used the concept *property* to refer both to properties of an object, and to real estate. When browsing the taxonomy, a user may notice a surprising conflation of documents reflecting such a multiple use of a word or concept. In these cases, a group reclassification interface we provide allows the user to split the word into two independent senses, relocate each of those senses under an appropriate set of parents, and select for each child whether it belongs under one sense, the other, or whether it too needs to be split along similar lines. This activity can be postponed until the presence of multiple senses in the taxonomy becomes a problem for the user. We are also working to develop techniques for automatically recognizing situations that are likely to reflect multiple word senses, either by comparing to word usage in a separate, reference taxonomy such as wordnet, or by direct analysis of relationships in the existing taxonomy.

Although these considerations greatly simplify the taxonomy maintenence function for a single user, it is still desirable to make use of relevant pieces of taxonomies constructed by other users or by the same user for other purposes. We address these considerations in the section below on Taxonomy Sharing.

5 Browsing and Retrieval

The Active Notebook taxonomy performs the important function of clustering documents into semantically related classes. From a document, the user can click on one of the concepts used to describe the document to browse a sorted list of related documents. These related documents are selected by having been assigned to a concept which is subsumed under the currently browsed concept in the taxonomy. They are sorted according to user specified criteria; by default, they are alphabetized by an algorithm that employs normal library alphabetization order, ignoring the articles *a* and *the*. This view of a document subcollection provides intuitive semantic clustering of related documents in the display, allowing the user to browse for other related documents. The display also shows, for each document, the list of concepts under which it has been explicitly classified; these provide another way to rapidly browse the taxonomy searching for other related-document subcollections.

A document can be instantly retrieved by clicking on its title in the Web browser. The document is retrieved using its original URL, so if the document has been updated at its network source location, the user will retrieve the latest version. The document footer interface, including user assigned concepts and personal and group

annotations, is automatically appended by the annotation proxy server.

6 Taxonomy Sharing

Despite our emphasis on personal taxonomy development and taxonomy maintenence, we recognize the benefit to being able to share taxonomic facts with others; primarily to allow a user to benefit from the classification work of others. While keeping the personal, user-centered taxonomy as the fundamental semantic structure in Active Notebook, we identify two ways taxonomic facts can be shared.

The first is through inheritance of facts from a shared taxonomy. Taxonomies of concepts representing categories relevant to a group, a laboratory, or a field can be maintained on separate servers. We are developing protocols by which a user's taxonomy can "subscribe to" or "inherit from" a number of other taxonomies, which can bear this multiple inheritance relationship to still other more general taxonomies in turn. Modelled on the notion of multiple inheritance present in modern object oriented languages, our distributed suite of taxonomy servers will provide facts about the relationship of new concepts to previously categorized ones. An implemented taxonomy server responds to socket-based connection requests to return lists of documents subsumed under a given concept, to compute and return the semantic relationship of a pair of concepts, and to modify the taxonomy by adding concepts or documents. This server can connect to other copies of the same program implementing higher level taxonomies, using the same interface to cascade the request to a more general taxonomy that may be able to handle it if the current one cannot. We are continuing to work on how to resolve conflicts and how to represent the dependencies of inherited taxonomic facts so they can be tracked in case the higher level taxonomy is changed. We are also working to define methods by which a user can export part of a personal taxonomy into a higher level shared taxonomy.

A second way for sharing taxonomic facts is by direct import from another user's taxonomy. When another user shares a document with me, for example, I might try to classify it in my taxonomy using the concepts she assigned to it. If these concepts are not present in my taxonomy, my taxonomy server may first ask her taxonomy server to explain the relationships it has represented among these concepts, until it reaches concepts my taxonomy server knows about. This provides a candidate default addition to my taxonomy, though many issues remain to be resolved in this proposed implementation.

7 Active Reminding

In the longer term, our work will address Active Reminding. Active reminding is a technique whereby a computer system with some understanding of the current topic of conversation or focus of attention can actively suggest relevant pieces of information from a user's Active Notebook.

While composing a new document, for example, natural language techniques may be applied to the partial document, trying to identify the important concepts from syntactic and semantic constraints. These concepts can then be classified in the user's personal Active Notebook, and the user may be reminded of highly relevant documents through any number of passive and active channels. The most likely channel will be a separate window on the user's screen, configured to change fairly infrequently so as not to disturb the user's composition process too much.

During a meeting or presentation, additional tools we are developing will help computers in a specially equipped conference room to have a continual text-based representation of the topic of conversation. To the extent that this can be accurately extracted from speech, written and typed discourse, participants can be actively reminded of relevant documents from their personal Active Notebooks online, during the meeting. This may prompt them to share interesting references with each other, for example.

8 Conclusion

We are at the dawning of a new age in information organization. As the balance of information becomes available in machine readable formats, the switch from paper based information management to more powerful tools is afoot.

Applying semantic processing to unrestricted text-based domains has always been problematic. By providing a convenient user interface and a simple and robust underlying representation of concepts, we hope to make maintaining a personal taxonomy into a useful tool for information organization.

9 Acknowledgements

The development of Active Notebook was supported in part through the author's participation in a summer internship program at Sun Microsystems Laboratories under the supervision of Dr. William Woods, and through his subsequent part-time employment at that laboratory. Additional development is continuing at the MIT Artificial Intelligence Laboratory in the context of an ongoing Ph.D. dissertation, under the supervision of Professor Lynn Andrea Stein. The author is a member of the AP group at the MIT AI Lab, and is presently employed as a research scientist coordinating the activities of the Human-Computer Interaction project at the MIT AI Lab, supported under contract number F30602-94-C-0204 from the Advanced Research Projects Agency of the Department of Defense as part of the Human-Computer Interaction Initiative. Great insight has been gained from discussions with Larry Bookman, Rod Brooks, Michael Frank, Bob Kuhns, and John Mallery.

References

[1] Richard Fikes, Mark Cutkosky, Tom Gruber, and Jeffrey Van Baalen. Knowledge sharing technology: Project overview. Technical Report KSL 91–71, Knowledge Systems Laboratory, Stanford University, November 1991.

[2] Tom R. Gruber. Ontolingua: A mechanism to support portable ontologies. Technical Report KSL 91–66, Knowledge Systems Laboratory, Stanford University, November 1991.

[3] S. Jerrold Kaplan, Mitchell D. Kapor, Edward J. Belove, Richard A. Landsman, and Todd R. Drake. AGENDA: a personal information manager. *Communications of the ACM*, 33(7):105–116, July 1990.

[4] D. B. Lenat and R. V. Guha. *Builiding Large Knowledge Based Systems: Representation and Inference in the Cyc Project.* Addison-Wesley, Reading, 1990.

[5] George Miller. Wordnet: An on-line lexical database. *Internatoinal Journal of Lexicography*, 3(4), 1990. (Special Issue).