Structure and navigation for electronic publishing

John Tillinghast, Giordano Beretta
Computer Peripherals Laboratory

HPL–97–162
December 1997

publishing, Internet, World Wide Web, documents, structure, knowledge management, storytelling, ontology

The sudden explosion of the World Wide Web as a new publication medium has given a dramatic boost to the electronic publishing industry, which previously was a limited market centered around CD-ROMs and on-line databases. While the phenomenon has parallels to the advent of the tabloid press in the middle of last century, the electronic nature of the medium brings with it the typical characteristic of 4th wave media, namely the acceleration in its propagation speed and the volume of information. Consequently, e-publications are even flatter than print media; Shakespeare's Romeo and Juliet share the same computer screen with Adams' Dilbert and Dogbert.

The most touted tool for locating useful information on the World Wide Web is the search engine. However, due to the medium's flatness, sought information is drowned in a sea of useless information. A better solution is to build tools that allow authors to structure information so that it can easily be navigated. We experimented with the use of ontologies as a tool to formulate structures for information about a specific topic, so that related concepts are placed in adjacent locations and can easily be navigated using simple and ergonomic user models. We describe our effort in building a World Wide Web based photo album that is shared among a small network of people.

To be published in and presented at *Electronic Imaging: Science and Technology—Color Imaging: Device Independent Color, Color Hardcopy, and Graphic Arts III*, San Jose, California, January 28–30, 1998. *SPIE Proceedings*, Volume 3300

© Copyright Hewlett-Packard Company 1997

1 Information is flat

The World Wide Web is difficult to navigate because it has become an amalgam of information blurred under epistemological pressures and the rush for a quick buck. To make an analogy with printed media, people usually do not subscribe to a series of news services. Instead, they subscribe to a newspaper, where a knowledgeable staff selects the news of interest to their particular audience. News do not happen, they are made by skilled editorial teams. Moderators in some Netnews and distribution lists are somewhat the Internet equivalent of the newspaper editor, but this rare transliteration is not sufficient for the acceleration brought forth by the e-media; more powerful tools are required to tame the information flood following the Internet tsunami.

The skill of grasping a large body of information has always been an important one in human societies, from storytelling in the pre-historic ages, to Gottfried Leibniz in the modern ages. Leibniz — the inventor of scientific journals — described the value of being able to find a linear order in a set of information and creating knowledge in the form of a linear thread, which he called filum Ariadnes. More recently, Vannevar Bush[3] has expanded on these ideas and described how a number of threads can be combined in a graph, building the foundation for hypertext systems. Finally, Ikujiro Nonaka[9] has brilliantly exposed the spiral from tacit knowledge to explicit knowledge and again to tacit knowledge. Nonaka has described how difficult but important the step from an intuitive form of justified true belief is towards knowledge that can be expressed and communicated. Our quest is for tools that can help users in performing this step.

On the other end of the intellectual quality spectrum, the tabloid press is not the only manifestation of a flat medium. On the Internet, the Netnews have cost the research community countless hours of time wasted skimming through endless requests for help and time spent compiling digests of frequently asked questions (FAQs). Closer to home, a host of picture taking households stores their precious pictorial memories in old shoe boxes.

1.1 Remembering a search result

The common fallacy is to think that the beast of information overflow can be tamed with the powerful computer by adding RAM and accelerating CPU clocks. It is true, that powerful search engines have been built. These engines can perform a morphological analysis of text to find which document has the closest proximity to a query phrase, or they can combine face recognition with histogram analysis to find all pictures of junior. However, to stay with the image example, a search engine will never be able to distill sufficient iconographic data* to find the one picture of junior among hundreds, where he was joyfully reacting to dad's feedback on his first spoken word. While humans can

* Iconography refers to the meaning of pictures.

quickly build semantic context for past experiences, for machines tasks like categorization and the interpretation of moods, emotions, and attitudes will always be very hard.

To remain with the pictorial search engine, the question is what to do with the result of a search. This result can be sorted by some proximity metric, but that is only of ephemeral value, since it usually still requires a second manual search that takes into account semantic context. Because this second step is tedious when an image collection is large, the user will want to save this result, an operation that is done by assigning the search result to a category.

The new problem is now how to come up with appropriate categories for images. It is easy to quickly come up with a set like birth, christening, birthday, bar mitzvah, graduation, wedding, and funeral. However, if a picture taking household is to categorize all the 10,000 images they take during a lifetime, such a set of categories is clearly insufficient. Corinne Jörgensen[7] achieved a major breakthrough with her method based on Gestalt principles of organization* for constructing a set of categories for images. In essence, a number of subjects is given the task of categorizing a set of images so that they can easily retrieve them again. The results of all respondents are pooled so that a consensus categorization can be identified.

Although such a solution works for a constrained problem, like for instance the images archived at the Library of Congress, it will not work for an open problem, because it does not scale. For example, if we want to build an Internet based photo album for picture taking households to be deployed globally on the World Wide Web, we cannot prescient all possible categories for all cultures. If we encounter an image that does not fit well in any category, then we can either force it in the closest of the existing categories or we have to start from scratch.

For example, let us assume we have categorized all man-made objects. If we now come across a picture of a sewing machine on an operating table, we could categorize it under sewing tools and under surgical tools, but obviously both are incorrect (few people actually do know the package really contains a sewing machine). To classify the image we need a new category called surrealism. However, we cannot simply add this new category, because if we do add this category, we must, for example, review the classification of pictures with pipes (which pipes?) and urinals, which might have been misclassified now that we have gained this new insight.

* A conception of organization based on the idea that a perceived whole is not just the sum of its parts; rather, the parts are organized into structures in principled ways.[4]

1.2 Solutions for the professional cannot be re-purposed for the casual user

Let us assume that we can constrain or simplify the problem so much that we can agree on a categorization. This might be sufficient for professional applications, such as a stock photo agency, but it is not necessarily applicable to consumers, because it does not support storytelling; there is no support for easily weaving Ariadne's thread. For example, people do not store books the way they are stored in a library. As is clear from Jörgensen's methodology, people organize material so it is easy to find. For consumers the search criteria are different than for the professional. For example, a person might want to store a picture of the Golden Pavilion (Kinkakuji) under Kyoto instead of under religious buildings.

In summary, categorization alone does not solve the problem, the discovery process for categories is also important in itself because professionals and consumers use information differently. This difference is due to the different workflow in an industrial situation with high specialization and division of labor, versus the domestic situation where labor-saving devices permit everybody to do their own work. If the device is inadequate, the consumer will not buy it, and the company trading the device will not have a market. For example, Fig. 1 shows the difference between the workflow in a household and in a stock agency. The gap is so large that it is not possible to "down-strip" professional software for consumers; a new paradigm must be found, and if possible it should be so powerful that it can form the basis both for consumer use as well as foster a new generation of powerful tools for the professional user. The issue is not down-stripping but the scalability of a solution.

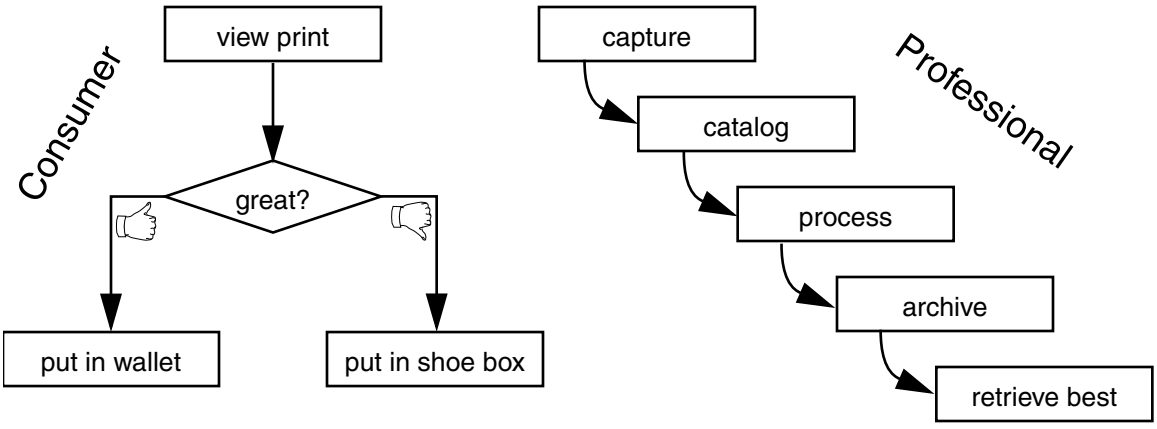

FIGURE 1. The workflow for archiving photographs by consumers and by stock agencies. Even if the consumer uses a digital camera, the shoe box method is more convenient than most gallery software. The stock agency employs a number of specialists, each highly skilled to perform one step in the workflow who then passes the picture to the next person in the flow.

When the solution has been found, a different application has to be built for the consumer and for the professional. The consumer needs tools that are simple and perform well one task, like an appliance. The professional needs a

powerful tool that offers the flexibility of extensive customization to a company's specific workflow.

In the remainder of this paper we will focus on the consumer application because it is a harder problem. We attempt to build a solution that scales so well that it is suitable both for consumer and professional applications.

2 Structure, navigation, and knowledge

For motivation we consider the contemporary family scattered around the world but connected with powerful electronic communication media. Our dream is that the extended family in the world can access a shared photo album on the World Wide Web and collaborate in the storytelling process. Different family members know different pieces of a story better and can add knowledge. For example, in wedding pictures not only a husband and a wife are united, but two families are united. Bonding the two families requires extensive storytelling, and at the beginning each family knows only their own stories.

A scenario for an application can look as follows. When a person logs into the system, the server uses one of the pictorial search engines mentioned above to find all pictures in which this person appears. For each picture that the person has not yet seen, the system asks that a comment be contributed if the person so desires. After this login step, the person is offered the choice between two tools, either tell-a-new-story or follow-the-thread-of-an-existing-story.

In summary, the idea is to use the World Wide Web to help users tell stories. The algorithms available today are either based on keywords or on pattern and histogram matching (image processing). The keyword methods are too difficult for consumers because we do not know how to come up with a good set of categories as well as aids that help the user find the correct categories. The image processing methods may be useful for the login example above, but in general when a set of images has been found it is not clear how to weave them into a story. What we need is a structuring and navigation tool.

2.1 Structure and knowledge

Fig. 2 illustrates Doug Engelbart's concept of categorization of evolving knowledge[5]. The set of all pictures in a shoe box is information, each picture stands by itself and we can make only few assertions about the connections among various pictures. Other examples of information are diaries, notes, address books, and the comments entered in the login procedure described above. On the right side we have structure, which in our case can be family relations, rank, and time. When the structure is applied to the information, knowledge is created.

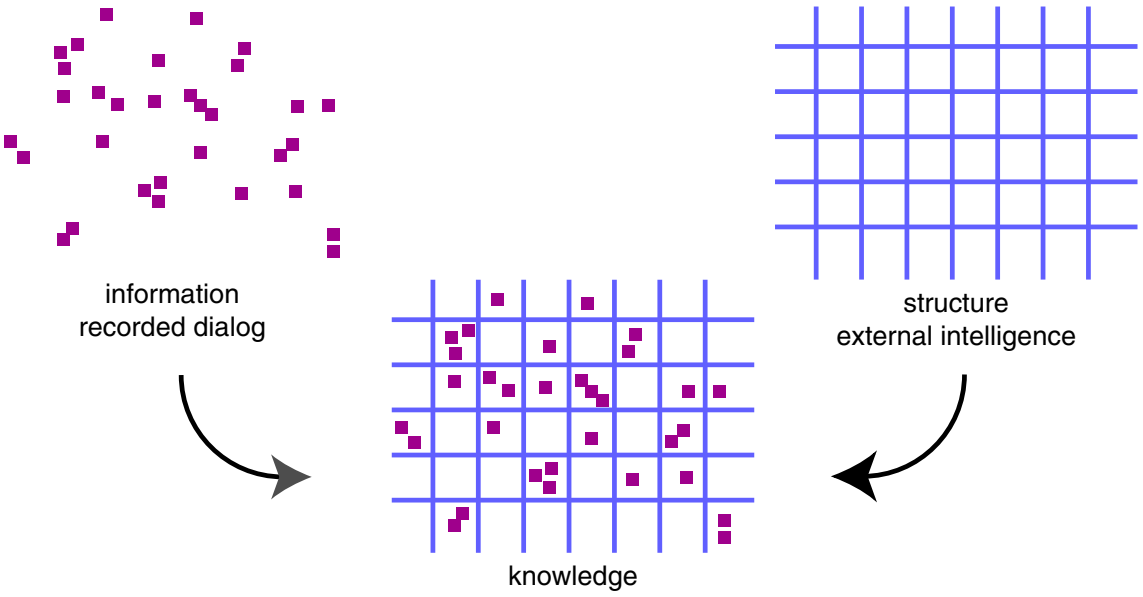

FIGURE 2. Categorization of evolving knowledge. We can easily accumulate information. We would like to create knowledge. If structure is provided as external intelligence, the creation of knowledge becomes an easier task.

For knowledge we are using a simple definition from the cognitive sciences[4], namely that entity *A* knows *p* if and only if

1. *A* believes that *p*,

2. *A* is justified in believing that *p*,

3. *p* is true.

Based on this view, knowledge is justified true belief. For this discussion belief is based on experience, skills, and mastery, i.e., we are interested in knowledge in its strict form of propositional knowledge—how to do things—rather than an accumulation of facts. Skills and mastery ensure that the belief is acquired by experience, ruling out that a belief meeting these conditions is possessed by sheer accident.

As Nonaka[9] notes, there are two manifestations of knowledge, tacit and explicit. Tacit knowledge consists of mental models, beliefs, and perspectives so ingrained that we take them for granted, and therefore cannot easily articulate them. Tacit knowledge is deeply rooted in action and in an individual's commitment to a specific context. Explicit knowledge is formal and systematic; it can easily be communicated and shared in product specifications or a computer program. The step of converting knowledge from tacit to explicit is very hard and is the task that we want to support with a computer based tool.

In the picture taking household, the shoe box is the recorded dialog. Frequently one family member is the predominant photographer, and this person over time builds the tacit knowledge of what is in the shoe box, how it might be organized, and how to quickly fetch a picture documenting a particular

event. For example, in a family reunion celebrating a silver wedding anniversary, the bride's father might be able to locate the picture of breaking the glass at the ceremony, but probably this task would be very hard and tedious for anybody else.

However, if somebody has collected the pictures in a photo album, this person has told a story, and any family member can easily retrieve the picture of breaking the glass. If we try to distill the knowledge directly from the information, then this task is very hard. However, if we use structures to organize the information as shown in Fig. 2, then this task can be performed systematically and with much less effort. Storytelling also becomes very easy because now all we have to do is traverse the structure to enumerate its elements and create a filum Ariadne as suggested by Leibniz.

2.2 Data structures and algorithms

If we look at a family, at first sight it looks like we can represent a family with a tree. There are algorithms to traverse a tree in level order, yielding a family structure by generation, depth first, yielding a family structure by lineage, etc. Some collections of images are not easily categorized with a tree structure. In such a case, a graph can be built to find the relationships among categories, to find cycles, or to introduce a partial order with such data structures as a minimum spanning tree. Other categories, such as time can be easily implemented with lists and other linear data structures.

If one reflects on contemporary life, these classical data structures are too weak to express relations. Friends and friendships can change, marriages can end in divorces. In today's family the children are not necessarily siblings, because each parent can bring children from previous marriages and children can be adopted or born from surrogate parents.

This tells us that even for a simple application like the family shoe box, we need a number of relations to represent the ways in which people can be related. We also need relations to be able to tell different stories, for example based on time, rank, emotion, life event, location, etc. Simple graphs are too weak and we have to look for a more powerful framework.

2.3 General structure theory and ontologies

Fortunately the mathematicians have already solved this problem and developed a well understood theory, namely the general structure theory[2]. This theory teaches us how to deal with structure in an abstract way. It is based on three concepts: a set corresponding to our information, an n-tuple of relations on the set, and a system of axioms, which in our case constrains the problem to one specific application. A general structure takes the name from the set of axioms, so we talk of lattice structures or group structures.

The benefit of general structure theory is that it shows us how we can map one construct into another so that the relations are preserved. In mathematics this leads to the concept of the abstract structure, which is an equivalence class of isomorphic constructs. For the application at hand, it tells us that we can implement structures in an abstract way, independent from the underlying information. From a business point of view, we can build generic application frameworks or skeletons that can easily be adapted to a variety of applications.

Our next step is to look in the various areas of computer science for a framework that might be used as a starting point for our solution. Indeed, people working on knowledge systems have used for about a decade a concept known as ontology, and the main effort in our project has been to find out how suitable ontologies are for our problem. A good place to start for learning about this branch of computer science is http://mnemosyne.itc.it:1024/ontology.html

An ontology is a formal and declarative representation which includes the vocabulary (or names) for referring to the terms in that subject area and the logical statements that describe what the terms are, how they are related to each other, and how they can or cannot be related to each other. Note the correspondence to the to above concepts of set, axioms, and n-tuple of relations.

Ontologies have the following advantages for discovering structures for publishing on the World Wide Web:

1. Systematic creation — by their nature, ontology editors allow knowledge workers to specify axioms and relations very systematically, so that there is a high probability that the resulting structure is complete.
2. Consistency — an inference engine asserts each operation on the ontology. This ensures that the axioms are always satisfied and that the relations are valid.
3. Many relations — unlike graphs, ontologies allow for many relations, which allows the description of real world structures.
4. Reusability — the structure is decoupled from the information and can therefore be reused for many collections of information. In a World Wide Web photo album application, all users can share the same structure in an ontology.
5. Extensibility — it is easy to extend all elements in an ontology. Relations can be added or refined and axioms can be changed. The system's inference engine verifies the ontology's correctness at each editing step.
6. Decoupling — to minimize ontological commitment, an ontology can be assembled from orthogonal components. This allows a community of practice to share external knowledge and collectively build more complete and powerful ontologies.

A number of ontology tools are available and we have decided to use the Ontolingua Server from the Knowledge Systems Laboratory at Stanford University for our experiments. Ontolingua had three advantages: it is built so that one can easily perform small investigations, it can easily be accessed over the Internet by just using a commercial browser, and it is free.

3 The Ontolingua Server

The Ontolingua Server can be accessed directly through the following URL: http://www-ksl-svc.stanford.edu:5915/

3.1 Terminology

In the practical application, the author of an ontology deals mainly with three terms:

- Class — a representation for a conceptual grouping of similar terms. Example: person.
- Relation — used to describe a relationship among two or more terms. Example: mother–child.
- Instance — all of the terms in an ontology that have an associated definition are an instance of some class. Example: me, my mother.

Other terms that are important but occur less frequently in the actual creation of an ontology are:

- Assertion — any statement that is true in the ontology. Each time a term is created, it is asserted to define it in the ontology.
- Axiom — a sentence in first order logic that is assumed to be true without proof. In practice, axioms are used to refer to the sentences that cannot be represented using only binary relations and values on a data structure.
- Domain of a binary relation — class that restricts the terms on which the binary relation can be added.
- Range of a relation — a class that restricts the values which the relation can have.
- Frame — a named data structure (or object) which is used to represent some concept in a domain. A frame can be a class, instance, relation, axiom, etc. A frame allows one to group some related statements about that concept. Associated with each frame is a group of binary relations and values on them.
- Slot — used to describe a relationship between two terms. The first term must be an instance of the class that is the domain of the slot and the second must be an instance of the class that is the range of the slot.
- Facets — used to represent information about a slot on an object. Usually facets represent some constraint on an instance slot.

- Term — any object that has a definition. An axiom is not considered a term.
- Value — a term which is related to the current definition through a slot or facet.

Software tools like the Ontolingua Server and the Java Ontology Editor (JOE)[8] are editors. The final product is an ontology. An analogy is when we use a text editor to write a book. The final product is a printed book, and the text editor is no longer necessary.

3.2 Wedding album

To test the suitability of ontologies, we have implemented part of a wedding album and planned two tests. The first is to verify the extensibility. We felt this test to be important because Jörgensen's method is not easily extensible, as mentioned in Section 1.1, "Remembering a search result". The second test is universality, i.e., how easy it is to use the ontology for a different set of images. This is important for the commercial exploitation of the technology, because it allows the tool provider to sell classes, and the consumer then creates instances.

We have selected the wedding album because it is a good demonstration of how a photo album tells stories. In a wedding there are certain characters — namely the guests — who all have different histories and interactions. In addition there is a clear time-line that follows from the arrival of the guests, through small talk, meeting new people, the ceremony, and finally the reception. The participant's shared knowledge evolves as the families merge and bond.

The wedding album also shows that the generality of an ontology is necessary to represent the complexity of real life. Trees are too regimented, because they force each picture to be on just one branch. Graphs do not allow one to easily extend the number of relationships — human relationships are too complicated.

As mentioned earlier, the Ontolingua Server is freely available on the World Wide Web from the home page of the Knowledge Systems Laboratory at Stanford University and you are welcome to experiment with the wedding album. Once you have logged in, you will be in the page that lists the library of ontologies; scroll to the bottom of the page, where you find the list of unloaded ontologies. Click on the link labeled Wedding-Pics. The Ontolingua Server returns you a new page of the library of ontologies, and now it contains the ontology Wedding-Pics. Click on it to start experimenting.

For example, if you want to add a sister, follow the simple steps below:

1. Put the ontology editor in edit mode (the other mode is browse) by clicking on the icon at the top of the page.

2. Go to an instance of a person (e.g., brother) whose sister you want to add.
3. On the top frame click on “create new instance.”
4. Now you have a new unnamed instance. Type the sister’s name, make it an instance of Person, and then click on “assert new instance.”
5. In “has sibling” click on the value button. Type the name of the brother and click on “assert new value.”

When you are done with your experiments, you can click on the “ontology” icon on the top frame. You get a new top frame, in which you can click on the “file” button to save your changes and make them available to other experimenters.

If you want to add a picture to the album, that is also very simple, although it may be tedious to find a suitable picture in the shoe box. Our collection of shoe boxes is at http://www.hpl.hp.com/personal/Giordano_Beretta/Albums/rolls.html and the wedding pictures are in the shoe boxes number 1420, 4163, 4165, and 4313. Finding a picture is not easier than finding it in a physical shoe box; it is a tedious trial and error sequence. Once you have found a picture you like, you can add it to the ontology. Go again in the edit mode and in the top frame click on the “create” button to create a new instance (select “instance,” then click “create”). The instance name is the title of the picture. Look in one of the existing instances of the wedding album ontology to learn how we include a simple HTML tag to have Ontolingua display the figure.

4 Result

The experiments with the Ontolingua Server have shown that ontologies are an excellent tool to systematically specify a structure. We have confirmed that it scales well and is easy to extend. We ran out of time and where not able to verify the universality, namely that we can use the same classes and relations of an ontology and instantiate it with the images of a different wedding.

Although an ontology editor is an excellent tool for building systematically a structure for a set of images, it is tedious to use in browse mode, i.e., for story-telling. The Ontolingua team has recently made substantial improvements to the implementation by developing the Open Knowledge Base Connectivity protocol (OKBC), which has a complete implementation in Java and can be used to talk to Ontolingua from small problem-specific programs like custom browsers implemented as Java applets. The Knowledge Systems Laboratory also has plans for a CORBA binding for OKBC.

Once the ontology editor has been used to define an ontology, the ontology can be hard-coded for a specific application or set of images in a popular language such as Java and with a limited user interface that only permits actions meaningful for the particular application. Frames in the ontology can be translated into data structures for any language. Classes in an ontology are

hierarchical, so an ontology is best implemented in an object oriented language, i.e., a language supporting inheritance.

Once the ontology has been "printed" the structure can no longer be changed, but information can still be added, because the relations are also encoded as objects, whose instance data is variable.

5 Conclusions

The World Wide Web started as a tool for building knowledge in a community of practice, namely the researchers working on high energy experiments at CERN. In the meantime, the World Wide Web has become a hot mass-medium, where flat information is accumulated and decisions are postponed indefinitely; nothing is too trivial or too important, and there is no mastery, just gossip. This situation was discussed in detail at last year's conference[1].

We have shown that the World Wide Web can be made available to a large variety of communities of practice if we provide tools that help them define structures, which will allow the communities to easily distill information into knowledge. We have experimented with an ontology editor and verified that it is effectively possible to systematically specify structures. For simplicity, we have operated on a specific example, namely a World Wide Web photo album, but the methodology is very general can be applied to a large number of interest areas.

In the case of a photo album, the resulting structure can be implemented as an application-specific browser that can traverse specific structures. In other types of information, other artifacts may be the best incarnation of a structure. For a document component database, for example, the best implementation is an Extensible Markup Language (XML) Document Type Definition (DTD). As radically different these two examples appear, they are both based on the concepts of general structure theory.

6 Acknowledgments

We thank Arindam Banerji, Michael Creech, Dennis Freeze, Allan Kuchinsky, Jim Rice, and Helen Werdegar for their assistance and encouragement. This research was funded by a grant from the HP-Labs Grassroots Basic Research Program.

7 References

1. Giordano Beretta, "Internet's impact on publishing," *IS&T/SPIE Symposium on Electronic Imaging Science & Technology*, Technical Conference 3019, pp. 200-211, San José (California), February 1997. Also available from http://www.hpl.hp.com/personal/Giordano_Beretta/Acrobat/Reports/hpl-97-33.pdf

2. Nicolas Bourbaki, *Elements of the History of Mathematics*, Springer-Verlag, Berlin, 1994.

3. Vannevar Bush, "As We May Think," *The Atlantic Monthly*, July 1945.

4. Charles E.M. Dunlop and James H. Fetzer, *Glossary of cognitive science*, Paragon House, New York, 1993.

5. Douglas C. Engelbart: refer to http://www2.bootstrap.org/biblio.htm for a complete bibliography.

6. Adam Farquhar, Richard Fikes, James Rice, "The Ontolingua Server: a Tool for Collaborative Ontology Construction," *KAW96*, November 1996. Also available from http://ksl-web.stanford.edu/people/axf/papers/KSL-TR-96-26.ps

7. Corinne Jörgensen, "Classifying Images: Criteria for Grouping as Revealed in a Sorting Task," *Proceedings of the 6th ASIS SIG/CR Classification Research Workshop*, pp. 65–78, Chicago, October 1995.

8. Kuhananda Mahalingam and Michael H. Huhns, "A Tool for Organizing Web Information," *IEEE Computer*, pp. 80–83, June 1997.

9. Ikujiro Nonaka, "The Knowledge-Creating Company," *Harvard Business Review*, pp. 2–9, November-December, 1991.