The BoW Project

Dror G. Feitelson
Institute of Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, Israel

Abstract

The BoW project aims to create tools for the construction of an on-line bibliographic repository. The repository itself will be created and maintained by its users. Thus it will have a growth potential far beyond that possible with a repository maintained by an individual or small group. But with size come the problems of searching and data quality. The searching problem is solved by a hypertext concept index, to which all items may be linked dynamically by any user. The data quality problem is solved by a dual rating system. First, all search paths executed by users are recorded by the system, and rated according to the number of successes. In addition, users may add annotations to links and items. Thus each user can benefit from the experience of all others. These concepts can lead to a truly on-line publishing system.

1 Introduction

Many contemporary scientists (and indeed people in other fields) complain of information overload — they cannot keep up with the information that is generated worldwide and is pertinent to their work. One of the problems is actually finding and classifying information, in order to focus on that which is most important. The idea of the BoW project is to make this a communal effort, where each individual can benefit from classifications and evaluations done by others. It is a large scale extension of the concept of peer review, coupled with an indexing service. The important features are that the repository is not just an unordered collection of information: rather, it is an *indexed* collection, and the information is *rated* by its users.

The mechanisms and technology used are based on the Internet. The information repository is kept on the Web, and accessed using conventional web browsers. This enables wide accessibility from around the world. The indexing is based on using hypertext documents with links to related concepts. Again, this is traversed using web browsers. Dynamic changes and the recording of ratings are done using Perl-mode and JavaScript. Thus the project is based on technology that is now available and well-known, but was all but non-existent just a few years ago.

2 Background

2.1 The Info-Glut

In 1948, the University of Malaya, Singapore, acquired a complete set of the *philosophical Transactions of the Royal Society* covering the years 1665 through 1850. Derek de Solla Price, who was a postdoc there at the time, took custody of them until the university's library would be built. He arranged them in chronological piles against the bedroom wall in his home, and discovered the law of exponential growth of scientific literature [28].

In the 50 years that have passed since this discovery, it has been reaffirmed many times. The number of scientific papers that are published continues to rise, as well as the number of scientific journals that house them. The most recent contribution comes from the Internet, with its exponential rise in the number of sites [27, 29].

The problem we face today is therefore not the lack of information. There is more information available then ever before. The problem is how to find what we need in this huge sea of information. This actually has two facets. First, we need to distinguish the information pertaining to the topic in which we are interested, and reject information on other topics. However, even the amount of relevant information may be too big to assimilate. We therefore need to prioritize the information we have found according to its quality, so that we can focus on the most useful information.

2.2 Traditional Approaches

Traditional approaches to bibliographic search include the Science Citation Index, and indexing services such as INSPEC.

The Science Citation Index is based on the notion that related papers in the scientific literature are "linked" by their bibliographical citations: either they cite each other, or they share many citations. Thus if you have a starting point for your search, namely some scientific paper on this topic, you can use both *its* citations and citations *to it* to find additional related papers. Citations in a paper are easy to find: they appear in the paper itself. The Science Citation Index provides the other direction: for any given paper, it lists other papers that cite it.

Indexing services don't require you to have a starting point — only a good notion of what you are looking for. They survey the scientific literature as it is published, and collect papers by topic. Thus they provide a much needed mapping from topics to journal pages, much as an index at the end of a book provides a mapping from topics to the pages of the book.

The problems of these traditional approaches are threefold:

- They *lack interpretation.*

  This means that papers are associated according to superficial attributes (citations or keywords), not according to an understanding of what the papers are actually about and how they relate to each other. There is no real editorial work on classification and organization. Moreover, users cannot augment these mechanisms with private annotations that do contain interpretation.

- They are *source oriented.*

  This relates to the choice of papers that are covered: certain journals are selected, and all the papers that appear in them are included, starting from a certain year. Granted, an effort is made to select as many journals as possible, and to focus on the best journals. However, even good journals sometimes contain not-so-good papers, and some good papers are published in obscure journals. Results that are only published in conferences or technical reports are excluded outright. So are old papers that were published before the indexing commenced.

- They are *temporally fragmented.*

  This means that the service comes in monthly or yearly increments, and multiple volumes have to be searched to obtain the full information. This drawback is becoming less relevant with the advent of computerized versions of the services.

2.3 Searching on the Web

The World-Wide-Web has created new opportunities for the organization of and access to information. The most obvious innovation is the wide-spread use of hypertext documents that point to each other around the globe. However, many projects use such capabilities merely as technical improvements over past practices, rather than seeking truly new approaches.

2.3.1 Digital libraries and citations

One example from the field of bibliographic search is the NEC Research Index, which is based on an automatic tool that crawls the Web looking for scientific papers, analyzes them, and creates a citation index from them [25]. This results in a useful on-line citation index, including automatically finding related papers [10]. A related example is the hypertext bibliography project [21]. This project contains information about all papers published in a host of journals and conferences, mainly related to theoretical computer science, and maintains two-way links among these papers according to their citations. Thus it is essentially a version of the science citation index where you click on links instead of turning pages.

In addition to following citation-links, the hypertext bibliography project supports keyword searches of its database. Certain simple queries can be formulated, and matching entries from the database are displayed. This capability is available in practically all digital libraries, including those offered by the ACM [16] and IEEE Computer Society [20]. Somewhat more sophisticated search options are being investigated in the NSF digital libraries initiative [7, 12].

In fact, digital libraries may become just an indexing, search, and access service, rather than repositories of actual data. This, again, is a result of the nature of the Internet. Any individual with access to the Internet can generate and provide information. The role of the digital library is then to index all this information, support searches for useful data, and handle the commercial aspects of buying and selling data [12, 4, 3].

2.3.2 Internet search engines

While digital libraries may be viable for handling "official" publications in a specific domain, they cannot handle the whole internet. The Internet is huge and growing at an exponential rate. Tools for classifying information are not keeping up with it, and often fail to make intelligent classifications. Moreover, even if the classification is correct, there are no means to judge the quality of the data. Indeed, a recurring problem with web search engines is that the quality of their results is unacceptable [32]. And this problem of information quality is by no means unique to the World Wide Web [34].

As an example, here are some observations regarding the results produced by a query about "gang scheduling"[1].

Number of hits: Metacrawler [5] mercifully returned only 39 hits from all the different search engines it used: 9 from Lycos, 8 each from Infoseek, Excite, Thunderstone, and WebCrawler, and 1 from Yahoo (3 hits were found by two search engines). Obviously, this is obtained by interleaving the top few results of each engine, which also lead to relatively high relevance. However, it also missed a lot. A separate search using AltaVista [1] produced 862950 hits. Changing the query to "gang+scheduling" reduced this number to 491 hits, most of which were relevant.

Coverage and quality: While most of the hits were relevant to gang scheduling, they had widely different levels of importance and usefulness. Some were generally useful links to papers or descriptions of projects dealing with gang scheduling, but others where notes about a bug in a specific system (e.g. a note that the documentation doesn't say that gang scheduling is the default). Naturally, related documents that used the term "coscheduling" rather than "gang scheduling" were not found (in a separate search, Metacrawler found 26 mostly related hits for coscheduling, and 34 mostly unrelated hits for co-scheduling).

Wrong hits: A large fraction of the Metacrawler hits, and obviously nearly all of the AltaVista hits in the first search, were not about gang scheduling. The were found simply because they included the word "gang", or "scheduling", or both. A good example was the totally irrelevant web page that included the sentence "The RV6 forum got off to a rocky start due to a scheduling misunderstanding with the Van's gang". There were also quite a lot of pages about the problems public schools have with street gangs.

[1] A scheduling technique used on parallel computers whereby a job's processes are scheduled simultaneously on distinct processors [19].

directory	*editors*	*categories*	*links*
Yahoo!	80+	–	1200000+
LookSmart	160	24000	800000
Open Directory	12000	100000	675000
Snap	60	50000	400000

Table 1: *The sizes of major internet directories, according to Search Engine Watch [36].*

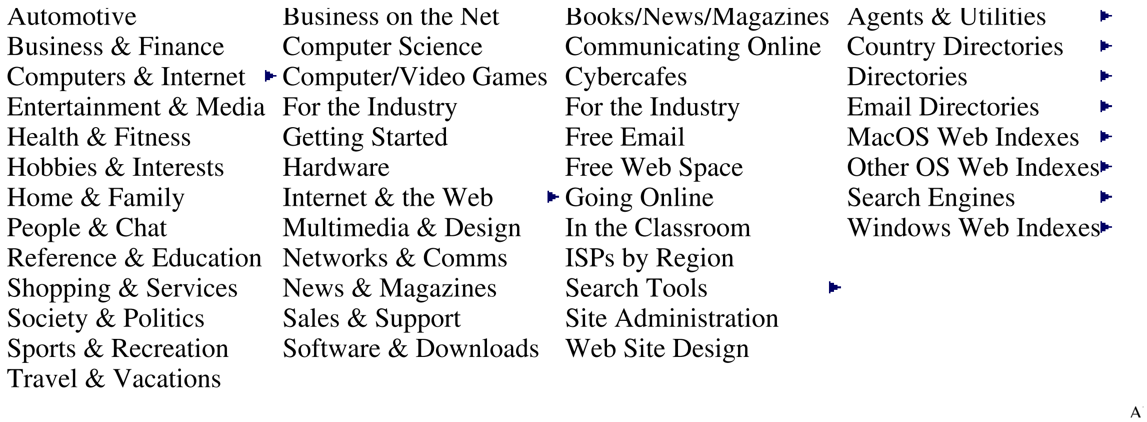

Figure 1: *Part of the AltaVista subject index. The little black arrow at the right of each column indicates which line has been expanded in the next column.*

Clutter and repetitions: The lists of hits often contain repeated entries of the same page or closely related pages from the same site, rather than a single top-level page for the concept being searched. This need not be so: for example, the Cha-cha system from Berkeley presents results in the context of their web environment [2].

Dangling links: Finally, there is always a problem with outdated links that don't point to anything, because the referenced page was removed or the server is down [23].

The above results are representative of keyword search in general, and stem from one basic problem: people using natural language tend to use terminology inconsistently. This has two facets. One is that people may use different terms to describe the same thing. The other is that people may the same term to describe different things. Automatic indexing according to keywords suffers from false negatives in the first case, and from false positives in the second case.

2.3.3 Human-generated directories

The only solution to this problem is to rely on human understanding. Indeed, most Internet indexing services now include large indexes (usually called Internet Directories) maintained by their staff in addition to the traditional keyword search facility (Table 1). These indexes

Arts Movies, Television, Music...	**Home** Kids, Houses, Consumers...	**Science** Biology, Psychology, Physics..
Business Jobs, Industries, Investing...	**News** Online, Media, Newspapers...	**Shopping** Autos, Clothing, Gifts...
Computers Internet, Software, Hardware...	**Recreation** Travel, Food, Outdoors, Humor...	**Society** People, Religion, Issues...
Games Video Games, RPGs, Gambling...	**Reference** Maps, Education, Libraries...	**Sports** Baseball, Soccer, Basketball...
Health Fitness, Medicine, Alternative...	**Regional** US, Canada, UK, Europe...	**World** Deutsch, Español, Indonesia...

Figure 2: *The top of the Open Directory [8].*

classify the whole Internet according to a hierarchical structure, and provide lists of generally useful web pages for each topic. An example of a path in the AltaVista index [1], looking for information about Internet indexing, is given in Fig. 1. The Yahoo! index is especially interesting, as it is a DAG rather than a tree, with explicit indication that some pages are shared by several branches [11]. The Open Directory also has a very useful structure (Fig. 2): The home page displays two levels of the directory, with links to pages on the main topics, and below each one, links to more focused subtopics.

However, it is obvious that it is impossible to actually index the whole Internet in this way [32], although some companies try and make a point of the fact that the classification is done by actual human beings [6]. What can indeed be done is to concentrate on a relatively narrow field, and try to index that. Indeed, such personal indices are an important resource for others to use [32]. However, maintaining them is a lot of work, and depends on the goodwill and the spare time of a single individual. Thus it is not practical to rely on such Herculean efforts as a general solution.

2.4 Volunteer Effort

A huge resource that has so far not been tapped in the quest for information classification and evaluation is the users of the data. It is the users who know what the data means, and what it is used for. Why not let them share this knowledge with each other?

Restricted forms of sharing knowledge exist in certain fields. For example, Biologists routinely submit data about DNA sequences and protein structures to centralized databases [30, 9]. However, this is limited to to very technical and raw data, and typically does not include much interpretation or more general derived information.

The most prominent examples of large numbers of people volunteering their time and intellectual output to contribute to a common cause come from software development. For example, the Linux operating system is based on contributions from thousands of hackers and software professionals around the world, whose only motivation seems to be the satisfaction of contributing and the kudos they receive [13]. Even more surprisingly, employers tolerate workers who do this on company time. The result is a system that is more advanced and more stable than any that can be produced by even the largest and most powerful software companies, if recent reports on Microsoft internal memos are to be believed.

The first informatics-related project based on volunteer effort is the Open Directory [8], with its 12000 editors. The current proposal deals with creating the technology for a much higher degree of collaboration, where all users (even sporadic visitors) contribute to the site's contents.

3 The BoW Approach

The previous sections demonstrate that it is probably unrealistic to hope that any one person or organization can create and maintain a comprehensive bibliographic database. Therefore the BoW project follows the "field of dreams" approach, which is the basis for the growth of the Internet itself [26]: we just *provide the technology* for creating the database, and *leave it to the users to supply the content*[2]. The resulting system is called a "**Dy**namic **R**anked **I**nformation-space", or DyRI for short.

The reason that a single individual or organization cannot maintain the desired service is the large volume involved in collecting the data and organizing it. These are therefore the tasks that are shifted to the community at large. Each user is allowed to (and indeed, invited to) add information to the database, be it a new entry, an evaluation of an existing one, or a new link between entries. This also ensures that the data is relevant for its users, timely, and at least has the chance of being consistent and complete — leading to high data quality [35]. The role of the service provider is then just to make verifications, judgements, and provide guidance.

3.1 User Types

Users of BoW are classified into three types: users, contributors, and editors.

Users are those who use BoW to search for bibliographic information. BoW provides a number of search options, as described below in Section 4. The main search method is by traversing a concept index that classifies the available information according to content. This allows for refining the search as one proceeds, rather than requiring one to have a clear notion of the required information at the outset. In addition, references can be found using keywords or based on their authors. This is described in detail in Section 4.

While general users do not add bibliographic data to the database, they do register feedback relating to existing data. One form of feedback is simply by traversing the concept index: the system keeps counts of visits to each page, and uses this information to identify the more popular ones to future users. In addition, users may register positive or negative

[2]However, in our case studies, we do prime the information spaces with enough content to make them sufficiently attractive so that potential users overcome the "new technology" barrier.

feedback to each page, to note their level of satisfaction. Again, the system displays this information as part of the indication of a page's popularity.

Contributors are users who not only search for data, but also contribute data. In principle any user may become a contributor; the only requirement is to identify oneself to the system. Such identification is required both in order to attribute contributions such as annotations to their authors, and in order to identify the arguing parties in case of disputes. In extreme cases of misuse, it may be necessary to limit certain users.

Contribution can take any of three forms:

- Adding a new bibliographic entry to the database.
- Adding an annotation to an existing entry, providing additional insight into its importance or content.
- Adding a link between related pages or entries.

Links create the fabric of the concept index, allowing it to be traversed incrementally, at the same time narrowing the scope of the search. Whenever a new entry is added to the database, it should be linked to appropriate pages in the concept index. Contributors who discover additional meaningful links later may also add them. All this is discussed in Section 5.

In addition, contributors can make minor modifications to existing data, e.g. in order to correct errors. Contributors can also *suggest* major modifications, such as deleting entries or links. However, acting upon such suggestions is left to *editors*, after proper solicitation of a rebuttal from the original contributor. Thus the editor's main task is to resolve conflicts and maintain the quality of the database, based on input from the contributors. Editing is discussed in Section 6.

Note that the construction of the concept index has not been mentioned so far. This is the responsibility of the *editor-in-chief*, and his alone. The reason for this restriction is the feeling that the conceptual integrity of the concept index is of utmost importance. Therefore its architecture and maintenance should be left to a single knowledgeable individual [18]. In very large databases, the domain can be divided among a small number of individuals, with each having jurisdiction over a well-defined sub-domain.

4 Searching

Searching the database for entries that are relevant to one's interests is obviously its main function. Therefore good support for this activity is of crucial importance. The problem is how to classify entries and index them so as to make them findable, given that different people may use different terminology to describe their work, and that each bibliographic entry pertains to several topics. As such, this is similar to the problem of indexing cases in case-based reasoning [24].

The BoW system supports three methods for searching the bibliographic database:

- Search by traversing a linked concept index.
- Conventional search by keywords.
- Search by authors.

As the concept index is the heart of the system, we start by describing it first.

4.1 The Concept Index

Most search engines are unsatisfactory because users are required to have a good notion of what they are looking for before they start. The most common approach is to describe the query using keywords and logical operations; for example, the query (scheduling & (parallel — distributed)) is read as "find documents including the word 'scheduling' and either of 'parallel' or 'distributed' ". The intent is probably to find references regarding scheduling in parallel or distributed systems. However, the issue of whether we mean job scheduling by the operating system, task scheduling by the runtime system, or task scheduling by the compiler is left open. A good search engine will find all three types (and maybe more) and leave it to the user to shift through them. Adding keywords can reduce this burden, but runs the risk of false negatives, where items of interest are rejected because they do not contain all the specified keywords (and without any typos 8-).

In contrast, the concept index allows users to formulate their search incrementally on-line, and does not depend on matching keywords. Essentially it can be viewed as a menu-driven search. The top level of the index (the root) contains links to several broad topics. Following such a link leads to a page representing the chosen topic (a *concept page*), and including a list of subtopics and/or bibliographic entries. The index is navigated using a hypertext interface such as web browser, by going from one concept page to another. The leaves contain only bibliographic entries that pertain to a narrow and focused topic.

An example of the top levels of a concept index for the domain of parallel systems is shown in Fig. 3. Using such a structure, a user looking for information on the scheduling of parallel jobs will follow the "operating systems and runtime support" link from the top level, and then the "Job scheduling & processor allocation" link. A user looking for information about on-line task scheduling would diverge at the second level, and choose the "controlling the parallelism" link. A user looking for information about task scheduling by the compiler would start with the "programming languages, program development, and compilation" link at the top level, and then choose "compilation".

As noted above, the structure of the concept index is of utmost importance. One approach is to use a general framework, such as the ACM computing classification system [14]. This is probably appropriate for a general purpose bibliography (and indeed it is used in ACM's newly announced Computing Research Repository [15]). However, for any more specialized domain, it seems advisable to create a special index based on a thorough understanding of the domain. This should be done with an eye for what users might look for. The important thing is to strive for a crisp subdivision into unambiguous topics. It is *not* important that these topics be of "equal size" — rather, if a certain topic turns out to be bigger than the

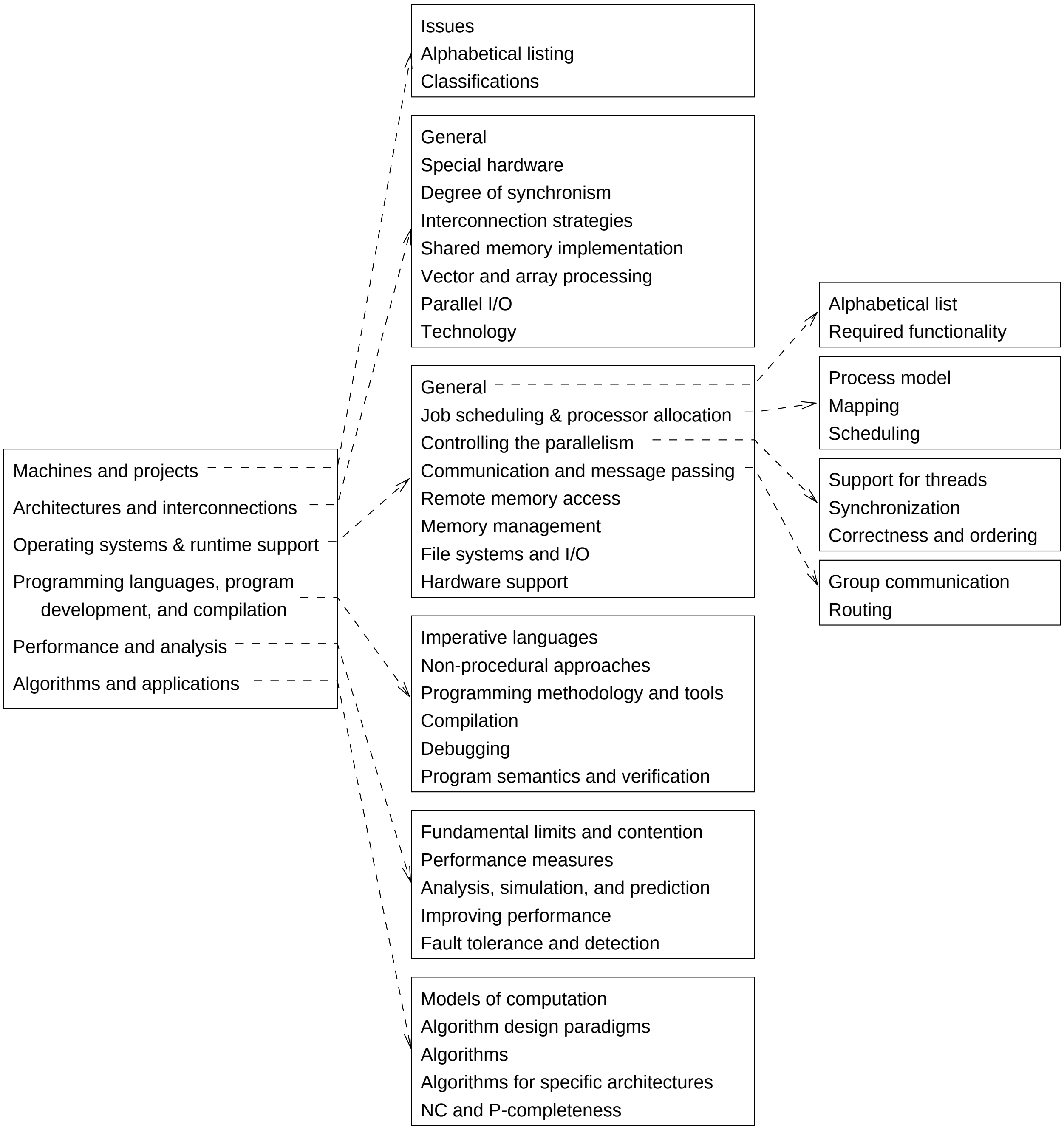

Figure 3: *Example of top levels of a concept index for a database on parallel systems.*

others, its index will be deeper. However, the topics need not be (and probably should not be) completely disjoint: the index structure can easily be a DAG rather than a tree. Thus any subtopic that is relevant to two or more larger topics (e.g. if it represents their intersection) is simply linked to all of them, and can be found by several distinct routes in the index. For example, it would be convenient if information on "virtual memory" was accessible both via "architectures" → "shared memory implementation" and "operating systems" → "memory management".

An important question is the "right" size for pages, and the resulting depth of the index [22]. The tradeoff is between scrolling and loading. Using small pages that do not require scrolling leads to a deeper index, and therefore requires more pages to be loaded from the server. If we want to reduce the average number of pages loaded in order to reduce the accumulated waiting time, we need to use larger pages. However, note that using relatively small figures such as a branching factor of 10 and 5 levels already leads to 1000 concept pages and 10000 leaves (bibliographic entries).

Making the pages too big also suffers from stressing the user, who must choose among the myriad options. A possible alternative is to use a relatively low branching factor, but show *two levels* of the index in each page (Fig. 4; this is also similar to the structure used in the Open Directory page shown in Fig. 2). This has two benefits. First, by showing the sub-sub-topics under each sub-topic in the page, we better define the scope of this sub-topic, making the search easier and more productive. Second, if one of the sub-sub-topics matches what we are looking for, we can go to it directly thus halving the number of pages that need to be traversed. True, the price is bigger pages, but their internal structure makes them easier to use.

4.2 User Feedback

An important part of the BoW interface is the support for registration and display of user feedback. This allows users to directly relate to the experience of previous users, as opposed to off-line methods such as referring to discussions in newsgroups [31].

The feedback feature is embedded naturally into the concept index, so as to be usable and useful without any training. Registering feedback about links in the concept index is done as a byproduct of traversing these links. One simple form of feedback is popularity: the system keeps count of the number of times that each link is traversed, and displays this *at the head of the link* (rather than displaying a count of visits in the page itself). Note that if a page has more than one link pointing at it, the counts for these links will be different, as they well should be, because they represent the perceived relevance of the page *in different contexts*.

The problem with mere counts is that they represent the *initial perception* of users, but not their *final satisfaction with their choice*. To capture user satisfaction, BoW pages have composite "back" buttons embedded in them (Fig. 5). By selecting the happy or sad face, the user can distinguish between a "happy back after finding what I wanted" and a "frustrated

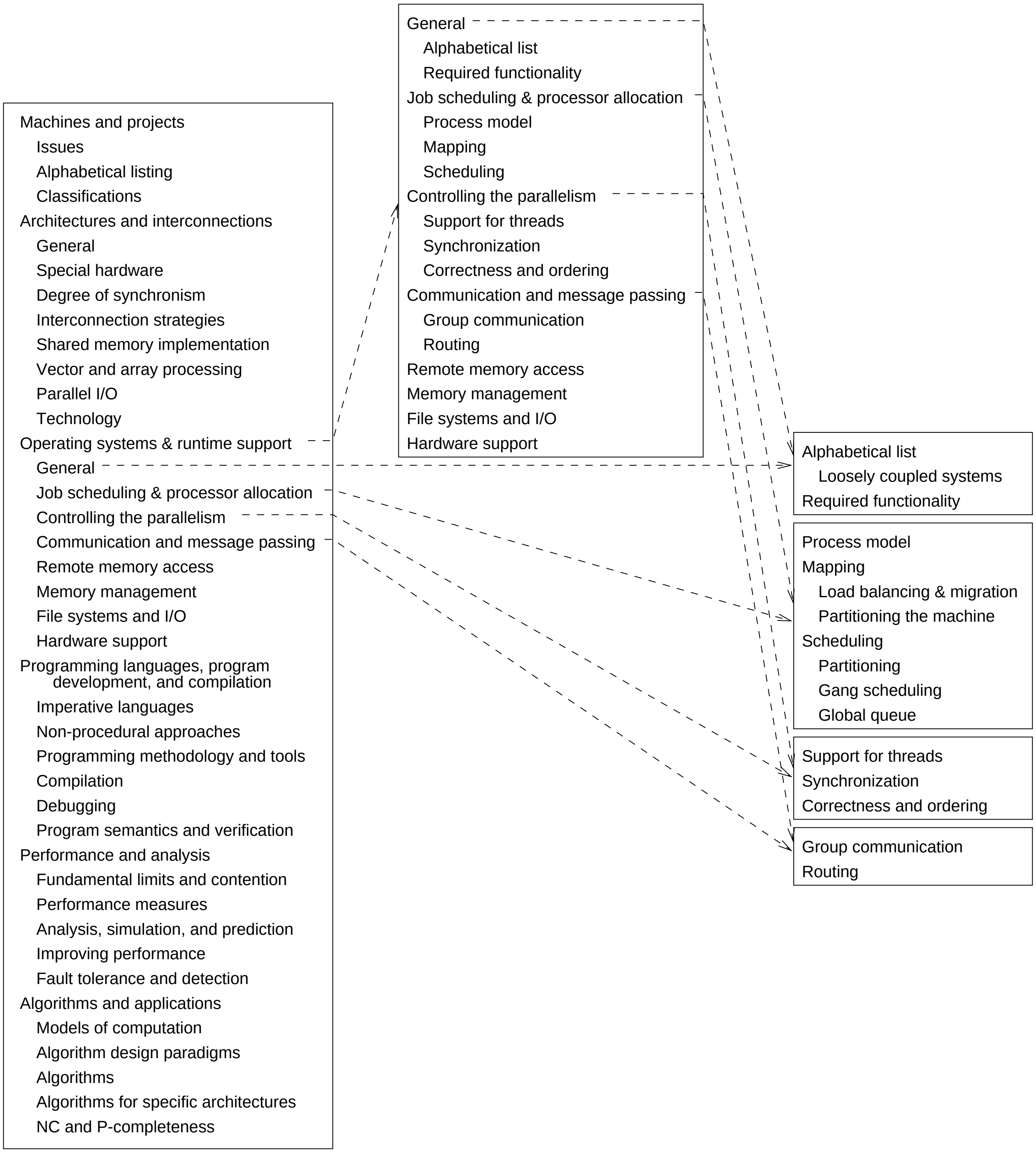

Figure 4: *Example of concept index showing two levels in each page. Compare with Fig. 3.*

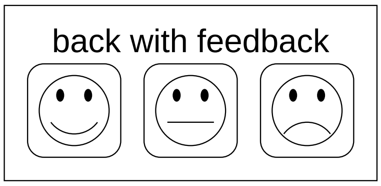

Figure 5: *Composite "back" button used to obtain feedback.*

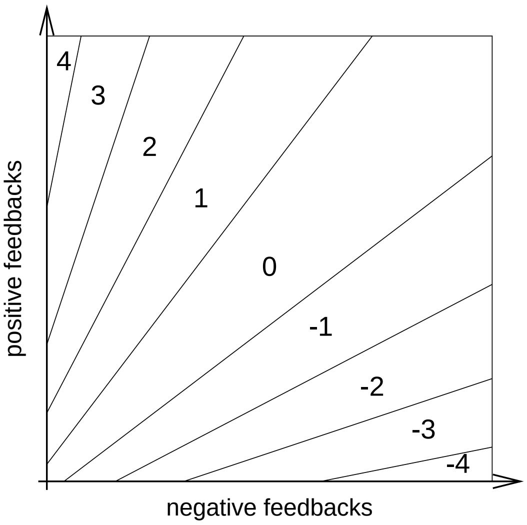

Figure 6: *dependence of feedback visualization on actual number of positive and negative feedbacks.*

back after failing". Clicking elsewhere just performs the back function, without registering any feedback.

The problem with the composite back buttons is that we do not want to burden the user with them. Therefore *inferential* feedback is used as well. One form of inferential feedback is that positive feedback is applied to the whole path from the root to the page on which the happy face is pressed, thus saving the need to go all the way back to register satisfaction. Negative feedback, on the other hand, is applied only to the last link, allowing for backtracking and trying of other links. Another form of inferential feedback is that using the "export" facility is deems to represent positive feedback, based on the assumption that the user is exporting data because he likes what he found. Export is described below.

Once the system has the feedback information, it should display it in a useful manner. The suggested approach is to decorate each link with a small icon that presents the information graphically. For example, a set of marks can be used, with the following properties:

- The type of mark signifies its meaning: green checkmarks denote positive experience, while red X's denote negative experience.
- The number of marks denotes the degree of confidence. More marks mean that there

were more feedbacks registered, and that the ratio of good to bad was more significant. An example is shown in Fig. 6, where the numbers indicate how many checkmarks (positive) or X's (negative) should be displayed[3]. Ties are displayed as a single circle.

- The size of the marks denotes the total number of feedbacks.

Such a display allows users to focus immediately on "many big green" links, which are those that many other users have found useful.

An additional use of feedback is the internal organization of the concept pages. It is envisioned that at lower levels of the index concept pages will be divided into topics, each listing a set of relevant bibliographic entries. The order of entries in such a set should be

1. New documents that were only recently added to the database. Such documents are kept on top for a few months or until they get some feedback from users.
2. Documents that have received positive feedback.
3. Documents about which users are ambivalent, or no feedback is available, possibly for lack of popularity.
4. Documents that have received negative feedback. These documents are candidates for removal, as discussed below.

The discussion thus far has emphasized the use of feedback as a side effect of normal work, with the goal of reducing the burden on users. In addition, we plan to provide a mechanism for more formal review, in which users can provide detailed comments that are sent to the site editors and can be forwarded (anonymously) to authors. Such a mechanism will enable the information space to become a respected venue of publication, ultimately replacing conventional mechanisms such as professional journals, as described in Section 7.2.

4.3 Keyword Search

Apart from searching via the concept index, BoW also supports conventional searching based on keywords. This is done both as a service in itself, and as an aid for constructing the concept index, as described below.

In many cases, a keyword pertains to many different elements in the database, leading to a long list of "hits". The main challenge in providing a service for searching via keywords is therefore prioritizing the results of the search. In BoW this is done by associating keywords with concept pages, rather than only with bottom-level bibliographic entries. Given such an association, two policies are employed:

[3]The prototype uses the following equation for checkmarks:

$$n = \lg \left(\frac{g-b}{b+1} + 0.7 \right) + 1$$

where g and b are the numbers of good and bad feedbacks, respectively. For X's, exchange g and b.

- If a keyword matches a concept page, then that concept page is entered into the results list. All lower-level pages and entries (accessible from this concept page) that ware found solely based on the same keyword are removed from the list.
- The resulting list is ordered from the more specific to the more general. Thus pages or entries that match more keywords appear first, and only then those that represent a partial match. In terms of the concept index, this means that low-level specific entries appear first, and then higher-level more general pages.

The combination of these two rules differentiates between the BoW keyword search and other search engines: in conventional search engines, you typically get a long list of matching entries, sorted from those that are perceived to be good matches to those that are inferior. In BoW you get a short list, starting with specific entries that represent a good match, and followed by more general concept pages that each represent a whole group of entries that only provide a partial match.

For example, consider a search for the following keywords: "operating system", "shared memory", and "cedar". Such a query should return the following, in this order:

1. Entries about Xylem, the Cedar operating system (matches all three).
2. Entries about the Cedar system (a shared memory multiprocessor), or a concept page for the cedar system if one exists.
3. Concept pages about shared memory and operating systems.

Note that it will *not* return hundreds of entries relating to all aspects of shared memory and operating systems, including all the systems that employ shared memory or have operating systems. If relevant, such entries can be found using the returned concept pages.

If only one relevant concept page or entry is found, it is returned directly, rather than returning a singleton list.

An important question is how to generate the keywords associated with the concept pages. In principle, the keywords of a concept page should represent the intersection of the keywords of all the entries accessible from that concept page. Rather than computing this intersection, we propose a reverse approach. When a concept page is created, the editor is responsible for assigning one or more keywords to this page. Then, when other concept pages or entries are linked to this concept page, the keywords are propagated to them. This ensures that high-level concept pages only have few general keywords (such as "operating system"), while low-level concept pages and entries have many keywords that represent the links to them from the concept index. In addition, the use of a domain-specific concept thesaurus is desirable, in order to overcome the possibly inconsistent use of terminology by humans [38].

A special case of keyword search is searching by an author's name. BoW maintains a page for each author, and automatically lists all bibliographic entries co-authored by this individual in this page. Searching for a name as a keyword returns this page.

4.4 Data Export

The outcome of a bibliographical search is a list of references. This can be used for either of two purposes: either going to the library to read the referenced material, or referencing them in a new work. In either case, it is desirable to export the data from the database rather than copying it by hand.

It should be acknowledged, however, that when the database is browsed interactively, one may find interesting references at any time. It is therefore desirable to have a mechanism by which such references are marked and accumulated as one searches for more. At the end, it is then possible to export the whole list of selections. In the context of BoW marking documents for export has a special side effect: it indicates user satisfaction, and is tabulated for display and suggestion to future users.

A simple solution for the first case, producing a human-readable list, is to press the browser's "print" button. Another option is to create a file in a predefined format suitable for computerized processing. For example, it should be possible to create a `.bib` file for subsequent processing by BibTeX; an output format suitable for MS Word is also desirable.

5 Updating

In order to keep the bibliographic repository useful, it must be continuously updated and improved. This can be done by any user who chooses to contribute. The only restriction is that the user has to identify himself to the system, so that it can keep track of activity patterns and the sources of information. With current technology, this will require initial registration for new users, and login for registered users (as automatic authentication is problematic and not standard in the Web environment [37]).

Letting anyone and everyone contribute has the drawback of lack of control, and runs the risk of having low-quality or self-interest data entered into the system. The BoW project is based on the assumption that this will not be a cardinal problem, because such data will be identified and purged by other users. Thus it will only have a transient effect, and not degrade the quality of the system on the long run.

5.1 Data Entry

The most important form of contribution is the entry of new data into the repository, and its indexing.

The basic entries in our repository contain bibliographic data (journal and conference papers, books, etc.). As this sort of data is highly structured, it is possible to design forms with predefined fields that have to be filled out. Thus the contributing user must first select the type of entry (article, conference paper, book, thesis), and then gets the appropriate form to fill. The fields in each form depend on the type, as shown in Table 2.

Once the form is filled and submitted, the system must first ascertain that the entry is indeed new, that is, that it does not appear already in the repository. This is complicated by

field	article	conf. paper	book	thesis	tech. rep.
author	+	+	+	+	+
title	+	+	+[a]	+	+
journal	+				
volume	+	+	+		
issue	+				
book title		+	+		
series			+		
editor		+	+		
number		+			+
pages	+	+	+		
month	+	+		+	+
year	+	+	+	+	+

[a] of part (e.g. chapter)

Table 2: *Fields in forms for different bibliographic entry types.*

the fact that different formats are possible, especially in author names, but also in the titles of works and publications; moreover, typos can add to the confusion. The solution is to use the regular structure of the domain to our advantage: instead of verifying uniqueness based on the author names and title, the verification will be based on the publishing medium. Thus we only need to check that we don't have another entry referring to, for example, the same pages in the same issue of the same volume of the same journal. Numerical fields (such as volume, issue, and pages) are easy to check with practically no ambiguity. Fields like journal name will be given in the forms as a menu, so as to ascertain a consistent format.

After verifying that the entry is new, the system will prompt the user to add links to the new entry from the concept index. To ease this task, the system will first scan the entry's title and annotation for keywords. Using these keywords, it will automatically conduct a search of the concept index, and return a list of matching pages as described in Section 4.3. These pages constitute a list of suggested entry points into the concept index.

5.2 Quality Enhancement

In addition to adding actual bibliographical entries, users can contribute to the quality of the repository by making small additions to augment existing data. This can come in several forms:

Additional indexing. Any user can add links from the concept index to existing entries, not only to new entries that he has just introduced. This enables the concept index to reflect changes in the perception of entries, and allows for the indexing according to concepts that were not originally perceived as relevant.

Additional annotations. Any user can also add annotations to existing entries. All the annotations for an entry will be displayed one after the other, in the order that they were entered (changing this requires editorial intervention). Users can choose to display their annotation either with a signature or anonymously. However, the system maintains the annotation's author information in either case, for future reference, and possible solicitation of rebuttals.

Adding links. This includes links to external sites that are relevant to the reference, links to full text copies of the reference that are stored externally, and internal links among documents (e.g. from a preliminary version to the final version).

5.3 Corrections

One of the basic concepts of the BoW project is the mobilization of all its users to participate in the task of keeping the data in the repository at a high quality. Thus, in additions to the mechanisms for adding data and feedback as described above, we need a mechanism to allow any user to correct existing data.

The problem with such corrections is that it is not necessarily obvious that the correction is better than the original — in fact, it may happen that the original data is fine and the correction is erroneous. Therefore corrections will be mediated by the editor. Users will only be able to *suggest* corrections, and then the old and new versions will be forwarded to an editor for verification. In cases of factual dispute, the editor will solicit a rebuttal from the author of the original data.

6 Editing

Unlike conventional web sites, the editors of BoW are not directly responsible for the site's contents: this is left to the users. Editors are responsible for the *structure* of the site, and for handling special cases and difficulties as needed.

The most important editorial task is creating and maintaining the structure of the concept index. This is the responsibility of the editor-in-chief, possibly based on suggestions of contributors and other editors. To ease this task, tools geared towards hypertext management will have to be created [17].

The problem is that the division of an information space into discrete pages and the interrelations between them depend on the perception of the concepts embodied in the information space, and may differ among individuals. In addition, even handling the complex structures you created yourself may cause a large cognitive load. Tools that display the structure and allow it to be modified consistently are therefore required. Naturally, it is not possible to display the full complexity of a cross-linked information space on a limited 2D computer display. Therefore the tools need to be able to focus on certain areas and cross sections, as required by the current editing task. Expected features include

- A condensed graphical view of the whole information space.
- A detailed view of a selected page or set of pages, that can be used in editing.
- The automatic uploading of pages that are linked to the page being edited.
- Automatic consistency checks to detect dangling links and orphan pages.
- The ability to define "macros" whereby sequences of related operations can be parameterized and repeated. Some such macros will be predefined based on experience from maintaining early versions of the example information spaces.

An intriguing option that we plan to investigate is the use of the server-side-include feature of HTML. This allows pages that are presented to users as a unit to be composed of parts that are actually stored individually. In particular, such parts can be incorporated in multiple distinct pages. Thus we can share fractions of pages as needed, rather than just linking pages to more than one branch in the concept index.

More mundane tasks for editors are handling disputes among users regarding certain entries or links. A special but crucially important case is the handling of deletion of items from the repository: entries should not be kept indefinitely! Instead, only those that have withstood the test of time should be maintained in the repository. Others should be purged to reduce clutter. Alternatively, old and unused entries can be demoted to "hidden" status, where they exist and can be found using keywords, but are not linked to the concept index.

Any user will be able to suggest that an entry or link be deleted, and in addition, the system will automatically suggest the deletion of links that receive strong negative feedback. Such suggestions will be forwarded (anonymously) to whoever inserted them for comment, and then passed to the editor for judgement.

Finally, editors will have the power to possibly to black-list bogus and irresponsible contributors, and prevent them from causing additional damage.

7 Extensions

The BoW project has been ongoing since 1998, and a prototype implementation with the basic features exists, as well as a reasonably-sized information space that can serve as a case study (with over 3000 entries on the topic of parallel systems, organized using a concept index of about 150 pages). There is much to do in straightforward additions to the prototype, first to implement all the ideas described above, and then to improve upon them. In this section we look further ahead at more distant extensions.

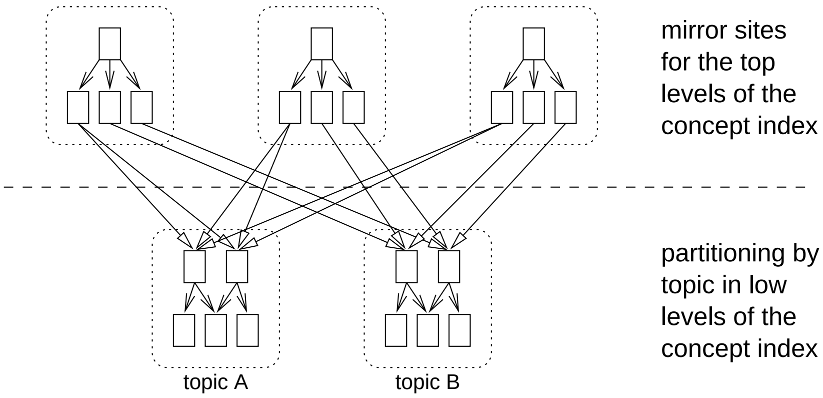

Figure 7: *Mirror sites and partitioning to support large-scale heavily used concept indexes.*

7.1 World-Wide Coverage

A single web server cannot effectively support a world-wide community of users. Therefore creating and using large scale information spaces such as BoW requires the cooperation of many servers.

The conventional approach to improving world coverage is to use mirror sites (e.g. [37]). But in a user-maintained repository such as BoW user interaction with mirror sites would lead to multiple versions that need to be updated and integrated. This can be done nightly by forwarding all the updates to the next site in a chain around the globe.

An alternative is to *partition* the concept index. Only the top few levels, which do not tend to change much, will be replicated in mirror sites (Fig. 7). The lower levels of the concept index can then be partitioned into disjoint sets of pages according to topic, with each set stored on a different server. While this complicates the task of global editing of the information space, it allows sub-spaces to be assigned to geographically dispersed subject editors.

7.2 From Bibliography to Publishing System

While the BoW project was conceived as a bibliographical repository, with bibliographical entries at the leaves of the concept index, it is not limited to such usage. Ultimately, the concept index can be augmented with actual papers at the leaves. This opens the way to a truly electronic publishing system, which is a whole new concept in publishing: the whole review is electronic, and it is updated continuously and dynamically. It breaks the conventional distinction between producers and consumers of publications, and offers much more than just electronic dissemination of conventional publications.

This is expected to work as follows. Authors of technical works will publish them by entering them into the information space, and linking to the appropriate concept pages. This will make the work immediately available to the whole community. Interested readers will be able to review the work and send their evaluation to the editors. The editors can then decide whether and when to ratify the work as “accepted”, based on the reviews and the standing of the reviewers. Works thus ratified will be marked by a special icon or presented in a larger font. The main differences from the conventional procedure are

- The publication delay is shortened to zero.
- The editors do not have to hunt for referees — rather, they simply wait for referees to volunteer. Works who do not attract any volunteers are probably not worth the effort.
- The review process can go on indefinitely.

Naturally, the acceptance of such a procedure is not guaranteed, due to its marked divergence from conventional practices [33].

8 Conclusions

The idea of a user-based bibliographic repository, with its embedded refereeing mechanism, seems very promising. However, its acceptance depends on getting the user interfaces right, and matching various unknown sociological and psychological factors. For example, success also depends on *lack of competition*. The whole community's effort should be concentrated, rather than fragmented among competing projects. In this respect, computer technology in general and the Internet in particular are a drawback, because it is just too easy to create yet another information service. Will BoW succeed in the real world? We have to wait and see. The proof of the pudding is in the eating.

Acknowledgements

This paper was completed as the first prototype was being implemented by David Er-El and Roy Peleg. As such it was influenced by their design decisions and contains some of their ideas.

References

[1] “*AltaVista: the most powerful and useful guide to the net*”. URL http://www.altavista.com/.

[2] “*Chacha search*”. URL http://cha-cha.berkeley.edu.

[3] “*Digital libraries*”. *Comm. ACM* **38(4)**, Apr 1995. (Special issue).

[4] “*Digital library initiative*”. *Computer* **29(5)**, May 1996. (Special issue).

[5] “*Metacrawler*”. URL http://www.metacrawler.com/.

[6] “*The mining co.: professional expert guidance from hundreds of guides for thousands of topics*”. URL http://home.miningco.com/.

[7] "*NSF/DARPA/NASA digital libraries initiative projects*". URL http://www.cise.nsf.gov/iis/dli_home.html.

[8] "*Open directory project*". URL http://dmoz.org/.

[9] "*Protein data bank*". URL http://www.pdb.bnl.gov/.

[10] "*ResearchIndex: the NECI scientific literature digital library*". URL http://csindex.com/.

[11] "*Yahoo!*". URL http://www.yahoo.com/.

[12] N. Adam, Y. Yesha, et al., "*Strategic directions in electronic commerce and digital libraries: towards a digital agora*". *ACM Comput. Surv.* **28(4)**, pp. 818–835, Dec 1996.

[13] J. Alper, "*From army of hackers, an upstart operating system*". *Science* **282(5396)**, pp. 1976–1979, 11 Dec 1998.

[14] Association for Computing Machinery, "*ACM computing classification systems*". URL http://www.acm.org/class/.

[15] Association for Computing Machinery, "*ACM computing research repository*". URL http://www.acm.org/repository/.

[16] Association for Computing Machinery, "*ACM digital library*". URL http://www.acm.org/dl/.

[17] P. Brereton, D. Budgen, and G. Hamilton, "*Hypertext: the next maintenance mountain*". *Computer* **31(12)**, pp. 49–55, Dec 1998.

[18] F. P. Brooks, Jr., *The Mythical Man-Month: Essays on Software Engineering.* Addison-Wesley, 1975.

[19] D. G. Feitelson and L. Rudolph, "*Parallel job scheduling: issues and approaches*". In *Job Scheduling Strategies for Parallel Processing*, D. G. Feitelson and L. Rudolph (eds.), pp. 1–18, Springer-Verlag, 1995. Lect. Notes Comput. Sci. vol. 949.

[20] IEEE Computer Society, "*Computer society digital library*". URL http://www.computer.org/epub/.

[21] D. M. Jones, "*The hypertext bibliography project*". URL http://theory.lcs.mit.edu/~dmjones/hbp/.

[22] S. H. Kim and C. M. Eastman, "*An experiment on node size in a hypermedia system*". *J. Am. Soc. Inf. Sci.* **50(6)**, pp. 530–536, 1999.

[23] W. Koehler, "*An analysis of web page and web site constancy and permanence*". *J. Am. Soc. Inf. Sci.* **50(2)**, pp. 162–180, 1999.

[24] J. Kolodner, *Case-Based Reasoning.* Morgan Kaufmann Publishers Inc., 1993.

[25] S. Lawrence, C. L. Giles, and K. Bollacker, "*Digital libraries and autonomous citation indexing*". *Computer* **32(6)**, pp. 67–71, Jun 1999.

[26] R. W. Lucky, "*New communications services — what does society want?*". *Proc. IEEE* **85(10)**, pp. 1536–1543, Oct 1997.

[27] Matrix Information and Directory Services, Inc., "*MIDS internet growth graphs*". URL http://www.mids.org/growth/internet/.

[28] L. R. Morris, "*Derek de Solla Price and the Antikythera mechanism: an appreciation*". *IEEE Micro* **4(1)**, pp. 15–21, Feb 1984.

[29] Network Wizards, "*Internet domain survey*". URL http://www.nw.com/zone/WWW/top.html.

[30] NIH National Library of Medicine, "*National center for biotechnology information*". URL http://www.ncbi.nlm.nih.gov/.

[31] R. M. O'Keefe and T. McEachern, "*Web-based customer decision support systems*". *Comm. ACM* **41(3)**, pp. 71–78, Mar 1998.

[32] E. Soloway and R. Wallace, "*Does the Internet support student inquiry? don't ask*". *Comm. ACM* **40(5)**, pp. 11–16, May 1997.

[33] C. Speier, J. Palmer, D. Wren, and S. Hahn, "*Faculty perception of electronic journals as scholarly communication: a question of prestige and legitimacy*". *J. Am. Soc. Inf. Sci.* **50(6)**, pp. 537–543, 1999.

[34] D. M. Strong, Y. W. Lee, and R. Y. Wang, "*10 potholes in the road to information quality*". *Computer* **30(8)**, pp. 38–46, Aug 1997.

[35] D. M. Strong, Y. W. Lee, and R. Y. Wang, "*Data quality in context*". *Comm. ACM* **40(5)**, pp. 103–110, May 1997.

[36] D. Sullivan, "*Search engine watch*". URL http://www.searchenginewatch.com/.

[37] S. W. Thomas, K. Alexander, and K. Guthrie, "*Technology choices for the JSTOR online archive*". *Computer* **32(2)**, pp. 60–65, Feb 1999.

[38] L. W. Wright, H. K. Grosetta Nardini, A. R. Aronson, and T. C. Rindflesch, "*Hierarchical concept indexing of full-text documents in the unified medical language system information sources map*". *J. Am. Soc. Inf. Sci.* **50(6)**, pp. 514–523, 1999.