Refereed article
The challenge of metasearching

Tamar Sadeh

The author

Tamar Sadeh is Marketing Manager, Information Services Division, Ex Libris, Jerusalem, Israel.

Keywords

Libraries, Library services, Information retrieval, Computer software, Portals

Abstract

The demand for metasearch capabilities – which enable users to simultaneously search heterogeneous information resources – is constantly increasing in the scholarly information environment as the number of available resources grows. To make efficient and accurate metasearching possible, library technology has begun to address several issues. First, information about resources must be accessible to metasearch systems. Such information, called resource metadata, can be made available to metasearch systems in various ways. Second, a metasearch system must be able to convert a unified query as necessary and adapt it to the requirements of each searched resource, retrieve the results, and display them to the end-user in a comprehensive and friendly manner. Finally, because some repositories are not available to metasearch systems, local indexes can be created to access them. The MetaLib library portal from Ex Libris is used to provide examples where relevant.

Electronic access

The Emerald Research Register for this journal is available at
www.emeraldinsight.com/researchregister

The current issue and full text archive of this journal is available at
www.emeraldinsight.com/0307-4803.htm

New Library World
Volume 105 · Number 1198/1199 · 2004 · pp. 104-112
 · ISSN 0307-4803
DOI 10.1108/03074800410526721

The number of information resources in the scholarly environment is growing, and with it the need for integrated and easy access. Users expect to enjoy a single point of access to all the information resources that constitute their search environment; furthermore, the typical library patron is not aware, and does not want to be aware, of differences between resources. Institutions strive to provide both novice and experienced users with an interface that will enable them to obtain high quality materials from appropriate resources with minimal effort. The challenge for all the stakeholders in the industry – libraries, software vendors, and information providers – is to provide a friendly, integrated environment in a heterogeneous world, an environment that is as easy to use as a Web search engine and also returns authoritative results from library-defined collections. This paper describes the basic concepts underlying integrated access as available today and puts forth some of the challenges. MetaLib, the library portal from Ex Libris, will be used to provide examples where relevant.

A metasearch example

Metasearching, also known as integrated searching, simultaneous searching, cross-database searching, parallel searching, broadcast searching, and federated searching[1], refers to a process in which a user submits a query to numerous information resources. The resources can be heterogeneous in many respects: their location, the format of the information that they offer, the technologies on which they draw, the types of materials that they contain, and more. The user's query is broadcast to each resource, and results are returned to the user.

The development of software products that offer such metasearching relies on the fact that each information resource has its own search engine. The metasearching software transmits a user's query to that search engine and directs it to perform the actual search. On receiving the results of the search, the metasearching software displays them to the user.

Let us examine an example of a metasearch process[2] that a user carries out via a product such as MetaLib™, the Ex Libris library portal. A student is interested in the works of Henrik Ibsen. Since the student knows that Ibsen is Norwegian, she submits a search query to several Norwegian resources with which she is familiar, such as the catalog of the National Library of Norway; the catalog of the University of Oslo; and the television, radio, and newspaper archives maintained by the National Library of Norway.

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

The student submits the query "author = Henrik Ibsen" to all these information resources. She then receives the results. The display of results by resource helps her identify those results that seem most relevant.

One result from the television archive is a program about the play *Peer Gynt*, written by Ibsen. Looking at this record, the student decides to focus solely on the work *Peer Gynt* rather than all of Ibsen's works. She then uses additional functions of the system to submit a second query, "title = *Peer Gynt*", to the same information resources. This time she receives different results, including the *Peer Gynt Suite*, composed by Edvard Grieg – a result from the radio archive that she did not obtain earlier. However, the Ibsen play *A Doll's House,* from the catalogue of the University of Oslo, did not appear this time, although it was on the previous result list.

One can take the process one step farther and ask another question: "How did the student know of the resources relevant to her research?" Perhaps she is knowledgeable in this field and thus aware of pertinent resources. If she is a novice, however, she may have relied on the default resources that her library offered her on the basis of her group affiliation. Alternatively, she might have requested that the system find resources relevant to her subject, perhaps from a specific geographic region or containing a certain type of material – thus creating a personal searching scope maintained for her by the system and available for reuse.

One-stop shopping

Most researchers today deal with content residing in a broad range of resources. The student in the previous example might want to access materials such as the script of the play in the form of a book or a PDF file; literary analyses of the play; various recordings of the suite; the score of the suite; or a video recording or poster of a specific performance. The immediate search result is typically a bibliographic record or other form of metadata describing the actual material. From the end-user's perspective, the bibliographic records serve only as a means of obtaining the material itself. Users do not want to be bothered with technical issues such as the format of the material they seek and the software that they need to access it – the library OPAC, Adobe® Acrobat® Reader®, Microsoft® Word or PowerPoint®, MP3, the MrSid viewer, or any other software that handles specific types of files.

To provide users with convenient access to materials contained in a range of resources, library portals need to integrate multiple software products under a seamless interface. The first type of information typically presented to users as a search result is a description of the material – the metadata – such as a bibliographic record representing a video recording. Ideally, the user should see the material on her screen – in this case, the recording – without having to concern herself about how to find the actual material and how to view it[3].

The link from the bibliographic record to the actual material can be direct, an explicit URL embedded in the metadata, as in the MARC 856 field of a bibliographic record in library catalogues. However, in many instances, the system must perform calculations to create the link – for example, when the bibliographic record resides in one information repository, such as an abstracting and indexing database, but the actual material resides elsewhere, such as in an e-journal repository or the library's printed collection. Nevertheless, the user expects to reach the actual material. A library can make this possible by configuring a context-sensitive linking server (Van de Sompel and Beit-Arie, 2001), which links the user to the actual material as part of a set of extended services and onward-navigation options. Such links might include the electronic full text of an article, the holdings in the user's library OPAC or any other relevant OPAC, the institution's document delivery service, citation information, a periodical directory, Internet searches, and information about the book in Internet bookstores or content-based services such as those offered by Syndetic Solutions. The software determines the list of links on the basis of the information in the specific bibliographic record and the institution's subscriptions and policies as predefined by the librarians.

Resource discovery and information discovery

The process of finding relevant materials for research consists, therefore, of two stages. During resource discovery, the first stage, the user locates and selects the resources most relevant to the specific search. Next is the information discovery stage, during which the search is executed in the selected information resources and the results are retrieved.

Institutions strive to provide their members – be they students, staff, or researchers – with high quality resources that offer information of real value. The librarians are responsible for determining the content of the institution's collections, both physical and virtual, and setting the collections' boundaries. Every member of the

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

institution should in turn be able to define a personal collection that derives from the institution's collection.

Once the user sets the scope of a search and submits a query, the information-discovery phase begins. The metasearch system delivers the query to the selected information resources, retrieves the results, and returns these to the user. The process requires that the system "understand" the expectations of the resources regarding the form of the query, on the one hand, and the nature of the results, on the other. It is up to the system to convert the unified query and adapt it to the requirements of each searched resource; to deliver the query in the form appropriate to each resource; to receive the results; and to manipulate them so that they comply with the system's unified format.

Resource metadata

Resource discovery requires that resources be described in such a way that users will be able to discover them. In the simplest discovery scenario, resources are presented in an alphabetical list for browsing; however, many users would not find this method useful because they are unfamiliar with resource names. More sophisticated methods are available for assisting in resource discovery, but these methods require resource descriptions that will enable a user to arrive at a knowledgeable decision about a resource's relevance.

Librarians can provide such collection-description metadata – or resource metadata; the creator of the resource or another party can also create the metadata and make it available to the library portal. This metadata serves the resource discovery process and, later on, enables the metasearch system to execute an effective search.

A number of initiatives, including the UKOLN Collection-Description Focus (UKOLN, n.d.a) and RSLP Collection Description (UKOLN, n.d.b) have attempted to define standards for describing resources, and some of these projects are still underway – for example, the Dublin Core Metasearch Initiative Collection Description Working Group (Dublin Core Metadata Initiative, n.d.). Resource metadata was also a subject of discussion at the NISO metasearch workshop that took place in May 2003 (Needleman, 2003; NISO, n.d.).

Resource metadata can be made available in several ways:

- Resource providers can offer metadata to any metasearch system that attempts to access them for the purpose of information retrieval.
- A central repository can offer resource metadata to any metasearch system.
- Metasearch systems can create and maintain their own repository of resource metadata.

The first method – in which a provider describes a resource when relevant – seems best, if one assumes that librarians are notified in some way about the existence of resources. Providers offer the most accurate information about their resources, information that other repositories need not replicate. The Z39.50 Explain function was based on this very premise; when external software needed to access an information resource, the software would extract the details of the impending interaction from the resource on the fly and use the information to formulate the exact steps of the interaction (Hammer and Fayaro, 1996). Apparently, few vendors implemented the Z39.50 Explain function; in June 2000, less than 1 per cent of Z39.50 servers had this capability (Bull, 2000), and those who did, implemented it in a variety of forms.

The Z39.50 International Next Generation (ZING) group has defined a search and retrieve Web service (SRW) that makes no distinction between a server and a database (ZING, n.d.a). The simplification that is expected from the elimination of the database concept may result in broader acceptance of the Explain function (ZING, n.d.b).

The development of the Semantic Web promises to take the idea one step farther. With this approach, a typical metasearch process involves an interaction between agents that exchange requests and information to construct the final product, which is the information requested by the end-user. This is the vision, but because today's Web does not allow for such interaction between agents, an automated interaction between the metasearch system and a resource's own search engine cannot be achieved at the present time (Sadeh and Walker, 2003).

The second method – building and maintaining a central repository – would assure the availability of resource metadata but would pose new challenges. First, a decision would need to be made about which kinds of resources such a repository would store. Then a format for the resource metadata would need to be specified, as well as protocols dictating the manner in which resource metadata finds its way to and from the repository. Finally, a decision would have to be made about who is responsible for storing information in the repository and keeping it updated – should the repository handle these tasks, through harvesting programs, or the resource provider?

The Information Environment (IE) Service Registry pilot project, driven by MIMAS in the UK, in collaboration with UKOLN and the

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

University of Liverpool (The Joint Information Systems Committee, n.d.) is currently exploring the creation of a central repository for the higher education community in the UK. The purpose of the project is to provide a registry of IE collections and services and examine its feasibility in terms of discovery, access, maintenance, sustainability, ownership and scalability.

The global information science community is watching this initiative with interest to determine whether such repositories become comprehensive and robust enough to provide services as necessary.

The third method is one that the various metasearch products have already implemented. Each such product holds the metadata, both descriptive and technical, of all the resources that it can access. Products differ in the amount of descriptive metadata that they release to the end-user, the way in which they display the metadata, and the functionality that they provide based on the metadata. They also differ in the degree to which they implement the search interaction and hence also in the amount of technical metadata that they store[4].

The method whereby each metasearch system maintains information about the resources has many drawbacks; the most obvious is that every vendor of a metasearch system has to configure and maintain the resource metadata.

MetaLib, like other products, provides a repository that includes the metadata of all the resources that it can access. However, the metadata are not maintained as part of the software; they are stored in the MetaLib Knowledge Base, a repository of resource data and rules. The software itself does not include any information that relies on specific resources; it extracts the information from the Knowledge Base. This information enables the user to select the resources, and MetaLib, to perform the actual search and retrieval (Sadeh, 2001). If, in the future, one of the first two options regarding the origin of the resource metadata materializes, MetaLib will only need to extract the required metadata from another repository.

The MetaLib Knowledge Base

The MetaLib Knowledge Base is a proprietary repository provided to institutions along with the MetaLib software. The Knowledge Base holds two types of metadata about resources:

(1) Descriptive metadata, such as the resource's name, coverage, language, data types, and publisher. The user sees this information and with it can make a reasoned selection of resources. It is the same information that enables the system to create resource lists based on the user's specifications and display the lists in a comprehensive way. In short, this information serves the resource discovery phase described earlier.

(2) Technical metadata, such as the type of protocol that the resource supports, the cataloguing format it uses, and the physical and logical structure of the records that it retrieves. We can describe this information as a set of rules that define the flow, interface, and manner of searching and that the software uses for searching, retrieving results, and manipulating them – that is, for the information discovery phase.

The resource metadata in the MetaLib Knowledge Base can also be divided into global metadata and local metadata:

- Global metadata is that part of the resource metadata that is universal and does not depend on the implementation of MetaLib at a specific institution. This metadata includes the name of the resource owner, the coverage, and the interfacing rules.
- Local metadata is institution-specific; it relates to the way in which the resource is used in the institution's environment and is presented to the institution's members. Such metadata includes elements of authentication *vis-à-vis* the provider of the resource, the authorization rules that apply to the use of the resource by institution members, and the categorization information that enables the software to offer the resource in specific contexts at the institution. For instance, one institution might categorize a certain resource as medicine, whereas an institution with a different orientation might categorize the same resource as social sciences.

Ex Libris maintains a master Knowledge Base, which, through automated routines, updates the Knowledge Base at each installation as necessary. Institutions localize the relevant metadata and add configurations to local resources.

Searching and retrieving

The process of searching and retrieving in a heterogeneous environment is far from trivial. Each resource has its own expectations regarding the form and manner in which it receives queries; even if the resource supports a standard interface, such as the Z39.50 protocol, the metasearch system needs to make further adjustments so that

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

the resource's engine will interpret the query correctly (Peterson, 2003).

To enable the system to search, the Ex Libris Knowledge Base maintains information in various areas, such as the following:

(1) *Access mode*. The Knowledge Base "knows" what kind of interfacing protocol the resource employs and whether the interface is structured and documented – such as Z39.50, the PubMed Entrez protocol, or a proprietary XML gateway – or an unstructured HTTP protocol that requires the use of HTML parsing techniques to access the resource.
(2) *Password control.* This information specifies how users access a particular licensed resource – for example, whether the information provider requires a user ID and password, which the metasearch system delivers when the connection is established; and whether the software should redirect the query via a proxy to grant the user access.
(3) *URL creation*. If a URL needs to be formulated to hold the specific query, the Knowledge Base must have information on the expected structure of that URL.
(4) *Character conversion.* The information specifies which character set the system uses at the resource end and whether it is the same as that of the end-user.
(5) *Query optimization.* Information in the Knowledge Base specifies how the query should be structured:
    - The exact syntax that the resource's system expects.
    - The mapping of the fields in the system to the fields of the resource. For example, the information should specify the field to which the system should map the "author" field selected by the user for a specific query.
    - The format in which the system expects to receive an author's name, such as <last name><, ><first name>; <last name>< ><first initial>; or some other format.
(6) *Normalization*. The actions that the system takes when the search engine at the resource end does not support a specific type of search. The system must know, for instance, what rules to apply if the user looks for a specific subject but the resource does not support a search by subject.

Once the information is available, the metasearch system can indeed adapt a single, unified query to the requirements of the specific resource, as in the following example.

A user submits a query for "title = dreams" and "author = Schredl, Michael" in these resources:

- Library of Congress (Z39.50 access to the Voyager library system from Endeavor).
- NLM PubMed (the Entrez HTTP protocol).
- HighWire Press® (HTML parsing).
- Ovid MEDLINE® (Z39.50 access via the SilverPlatter ERL platform).
- University of East Anglia (UEA) (XML access to the Ex Libris ALEPH ILS).

Even when looking at one brick of the process structure – the query syntax – one can clearly see the differences between the resources:

- The Library of Congress expects this query string: "1 = Schredl, Michael AND 4 = dreams".
- Ovid's MEDLINE via ERL, although accessed by the same protocol (Z39.50), expects this query string: "1003 = Schredl-M* AND 4 = dreams". (Note the phrasing of the author's name.)
- PubMed expects this query string: "term = dreams + AND + Schredl + M".
- HighWire expects the encoded form of the following URL: "author1 = Schredl, + Michael&author2 = &title = dreams".
- The ALEPH system at UEA expects the following encoded request: "wau = (Schredl, Michael) AND wti = (dreams)".

Because metasearch systems rely on the search engine at the resource end, the search that they offer must be compatible with the searches provided by these resources. Scholarly resources typically offer rigid searching options, such as searching by title, author, subject, and keywords; wildcards and truncation symbols are usually supported as well. A metasearch engine needs to identify and comply with the set of searching options that most resources support and yet also handle situations in which a resource does not support that set. However, metasearch systems cannot fully exploit sophisticated search options offered by small groups of resources or options that are tailored to specific types of data; therefore, such systems should always enable users to navigate to the search form of a resource when necessary.

Furthermore, because of the truly heterogeneous nature of the resources that are searched simultaneously, a metasearch system cannot employ search aids such as specific subject headings.

Web users may find the search options of metasearch systems somewhat conservative, particularly when compared to the fuzzy searching provided by Web search engines. Here, again, metasearch systems cannot offer search options that are not supported by the resource. However, metasearch systems can move forward to

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

approximate some of these options. For instance, a metasearch engine can apply algorithms to process a query before converting it to fulfill a specific resource requirement. Such processing might involve auxiliary resources for handling misspellings, or the soundex algorithm to identify a name that has been typed incorrectly or has various possible spellings (such as a Russian name that is transliterated into English). Every such processing event requires prior knowledge of the type of information that the metasearch system is accessing; the larger the variety of resources being accessed, the more complicated such implementation will be.

Presenting search results

Up to now we have discussed only the flow from the user to the resource. However, now that the query has been processed, the metasearch system needs to return the search results to the user.

Typically the interaction between the metasearch system and the resource consists of two phases. The first occurs after the search has been invoked: the resource returns the number of hits and some kind of reference to the result set. This phase is important because it gives the user some information about the search results and enables the user to refine the query before browsing through the results. For instance, if a user receives thousands of hits, she can modify the query to be more specific and thus reduce the number of results.

The second phase consists of retrieval: the metasearch system retrieves the first few records for each resource. This information is shown to the user instantly, even though the query might have resulted in hundreds or thousands of hits. Some systems, including MetaLib, allow for further retrieval upon request.

The reason for retrieving only the first few records is twofold. First, retrieval depends on the use of networks, which are still not as rapid as one would like. Retrieving a large number of records over a network is an extremely time-consuming process, and users are not likely to wait until it is completed. Second, people have difficulty handling immense result sets; hence, after seeing the number of hits for each resource, users will probably refine their query to obtain fewer hits.

Once retrieved from the resource, each result is converted to a unified format before the user sees it. The rules that define the manipulation of the retrieved data are part of the resource metadata and include information about the logical format, the cataloging format, the script, and the structure of certain fields, such as the citation field. For further processing to take place, the metasearch system must be able to apply these rules and convert all retrieved records, regardless of origin, to its internal format.

Such additional processing can include the: unified display of the records to end-users; merging of result lists from heterogeneous resources into one list; comparison of records to eliminate duplicates; creation of an OpenURL to allow context-sensitive reference linking; and saving of records in whatever format is required. Consequently, functionality that might have been lacking in the native interface of the resource, such as the provision of an OpenURL, is added to the same set of records by the metasearch system.

However, the display of result lists is not as straightforward as might be expected. Users are well acquainted with Web search engines and therefore have solid expectations regarding the display. They would like their results ranked, merged into one list, and filtered for a selected resource. Furthermore, they would like to be able to sort results by various attributes, such as title, author, or date.

Given that only the first few results are retrieved from the various resources, these expectations are not readily fulfilled. When the result sets are small, all records are in the system's cache memory and so the metasearch system can offer the expected functionality in a comprehensive manner. However, the larger the number of hits, the greater the value of merging, sorting, deduplication, and ranking – and the more difficult these features are to provide.

Consider, for instance, the merging of the lists. Results come in from some resources faster than from others, because of differences in network configurations, and are therefore available for display first. Furthermore, the number of hits can vary considerably from resource to resource. Would it be appropriate to merge the two hits received from one resource with the dozens or hundreds of hits received from another resource? And if so, in which order? Every resource returns results in a different sorting order – by date (ascending or descending), title, relevance, or some other attribute of which the users are not necessarily aware. Because only the first few records are retrieved, the issue of merging needs careful consideration.

Other concerns are the sorting capability and relevance ranking that users expect to find when looking at results, even if the resource itself does not support such functionality. One must ask whether it makes sense to rank and sort only those results that have been retrieved. For example, a metasearch system applies certain relevance-ranking algorithms to the records that have

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

actually been retrieved and sequences them accordingly in the display to the user. This display might turn out to be misleading, because the "best" hits are not necessarily those that were retrieved first. It could well be that if the user asks for more hits, better results will be retrieved. A similar problem applies to sorting; even though a system might enable the user to sort the records according to various parameters, this sorting would apply only to the set already retrieved.

Any relevance-ranking algorithm that a metasearch system employs needs to be "objective" – it must rely on the retrieved document alone, without any knowledge of other documents in the resource from which the document is coming. Standard relevance-ranking algorithms in the library environment take into consideration parameters that are not available to metasearch systems, such as the uniqueness of search terms in the resource data (the less common a search term is, the higher the rank of the document that includes it). In addition, library catalogues are capable of giving weight to circulation information, under the assumption that the most frequently borrowed books are those that users tend to look for. Another parameter that catalogues might use is the number of items that a library holds for a specific bibliographic record. The contribution that these latter two parameters would make is disputable, because they would "make the rich richer"; in other words, by causing the more popular items to appear first, these methods would ensure that such items are more likely to be borrowed.

In a similar manner, Web search engines base their relevance ranking on the nature of the Web – that is, on information that is not contained in the data (the Web pages) but rather is derived from the popularity of the page in the specific environment. The search engine Google™ is based on PageRank™, a system for ranking Web pages that Google creators Larry Page and Sergey Brin developed at Stanford University (Google, n.d.). PageRank is "an objective measure of . . . [a Web page's] citation importance that corresponds well with people's subjective idea of importance" (Brin and Page, 1998). According to Brin and Page (1998), "counting citations or backlinks to a page. . .gives some approximation of a page's importance or quality." Google also analyzes and ranks the pages from which links originate. The higher the rank, the more weight the page carries in the ranking of the linked-to page (Google, n.d.). This ranking system, along with Google's text-matching techniques, ensures that the best results are indeed displayed first. The popularity of Google contributes to the accuracy of the PageRank algorithm; Web sites that appear high up on a Google result list are more likely to be visited by Web users than sites that appear farther down (Brandt, 2002).

Relevance-ranking algorithms that are tailored to the needs of metasearch systems and also take into account the limited knowledge that such systems have of the repositories that they are searching are only beginning to be developed. A new ranking method might eventually evolve from the fact that metasearch systems "know" both the user and the resource. Among the factors that this method would employ to calculate a result's quality would be the user's affiliation and area of interest as well as the database from which the result was returned. The ABI/INFORM database, for example, would have a higher ranking than Google.

Local repositories and local indexes

End-users may wonder why other searching systems, primarily Web search engines, are able to provide them with large sets that are merged and ranked. The reason is that these systems use a different type of technology to provide the users with search results.

Metasearch systems are based on just-in-time processing. The system does not maintain any indexes of its information landscape locally; only when the information is required does the system access the various resources to obtain the results.

The approach of Web search engines is based on just-in-case technology. Immense effort is invested in preparing the information prior to users" requests so that when the information is needed, it is obtained immediately. Google, for example, holds indexes for the entire World Wide Web; these indexes include not only pointers to sites, but also other types of information, including PageRank data. When a user searches with Google, only the indexes are scanned (and not even all the indexes at first – only those with high PageRank scores); the information that Google displays on the screen is not from the sites themselves but from this vast repository of indexes. The search engine provides the actual access to a certain Web location only when the user selects it from the list. Needless to say, huge computing power and disk space along with sophisticated technologies for harvesting, evaluating, and maintaining the information are necessary for such powerful tools.

The use of local repositories of indexes in the library environment began some time ago. As opposed to union catalogues, which actually replicate the information that is located in local catalogues, repositories such as MetaIndex from Ex Libris hold only indexes to the bibliographic

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

materials that are kept in the resources. An example is the MetaIndex implementation at the Cooperative Library Network Berlin-Brandenburg (KOBV) (Lohrum *et al.*, 1999), which preceded the metasearch systems a few years ago. At KOBV, MetaIndex enables each consortium member to maintain its library system and cataloguing conventions while the consortium provides a single search interface for end-users. MetaIndex has now become a resource available to MetaLib at KOBV, along with other resources.

Undoubtedly, a local repository of indexes has many advantages. Information that is gathered and processed prior to queries can be organised, evaluated, and deduplicated and therefore can be accessed by end-users in a rapid, comprehensive manner. However, maintaining such a repository has a major drawback: the repository is another system, with hardware and software, and personnel must be available to look after it.

Considering libraries" budget constraints and limitations in the technical expertise available to them, a combination of just-in-case and just-in-time approaches would be optimal for metasearch systems. Local repositories would be useful in the following cases:

- *When no searching mechanism exists at the resource end.* This situation is typical of various types of local repositories, such as those holding research papers written by institution members or spreadsheets relevant to institutional activities. However, any other information that has not yet been made available to the public is also a good candidate for inclusion in a local index.
- *When information is scattered.* A local repository might be worthwhile if several resources that are mutually compliant form a single resource of value to the institution. An example is a world-wide organisation that has dozens of branches, each of which holds regionally relevant information, and wants to provide a simultaneous search capability that will cover all the local information. Creating an index such as MetaIndex would be preferable to requiring users to search all the repositories simultaneously.
- *When network access is not reliable.* Some institutions want to provide access to resources that are not always on-line or do not offer reliable networking for accessing them. In such cases, an institution might be better off harvesting the information and keeping it as a local repository.
- *When preprocessing is important.* Preprocessing tasks such as relevance ranking and the elimination of duplicate records can be of value for some institutions. However, a component like MetaIndex can provide a solution only if the search scope is defined and limited. For instance, at KOBV, MetaIndex applies only to the catalogues of the consortium members. To address the preprocessing needs, the mathematics department of the consortium was able to develop a sophisticated deduplication algorithm that permitted the construction of a comprehensive MetaIndex component (Lohrum *et al.*, 1999).

MetaIndex from Ex Libris is created through the harvesting of information from other repositories. One of the harvesting mechanisms is the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) (OAI, 2003). The use of such a protocol can facilitate the gathering of data and applies to a wide range of resources that are now becoming OAI compliant. Furthermore, MetaIndex itself is OAI compliant, thus serving as both a resource for MetaLib and a resource that enables other systems to harvest data from it.

Looking ahead

A truly integrated environment in a heterogeneous world may not yet be a reality, but with the active involvement of all the stakeholders, significant progress has been made. Just a few years ago, metasearch systems seemed like an impossible goal; today they are already a building block in the information resource environment serving the academic and research community.

Notes

1 The term "federated searching" is used by some to describe a process in which indexes are "pregenerated." We refer to this concept as "just-in-case" processing, as explained later in this paper.
2 We provide this example only to illustrate the process; references to specific resources are not necessarily accurate.
3 The issue of copyrights is not discussed in this paper. In this context, we assume that the system that offers the material handles copyright matters.
4 For instance, some products offer unified searching, but once the user requests the result record, the software links the user to the record in the resource's native interface. Such products do not need to maintain all the technical metadata required for manipulating the retrieved record and converting it to a unified format.

New Library World
Volume 105 · Number 1198/1199 · 2004 · 104-112

References

Brandt, D. (2002), "PageRank: Google's original sin", available at: www.google-watch.org/pagerank.html

Brin, S. and Page, L. (1998), "The anatomy of a large-scale hypertextual Web search engine", in *Proceedings of the 7th Intl. WWW Conf., Brisbane, Australia*, available at: www7.scu.edu.au/programme/fullpapers/1921/com1921.htm, pp 107-17.

Bull, R. (2000), "Z39.50 server list response", available at: http://lists.w3.org/Archives/Public/www-zig/2000Jun/0008.html

Dublin Core Metadata Initiative (n.d.), "DCMI Collection Description Working Group", available at: http://dublincore.org/groups/collections/

Google (n.d.), "Google searches more sites more quickly, delivering the most relevant results", available at: www.google.com/technology/

Hammer, S. and Fayaro, J. (1996), "Z39.50 and the World Wide Web", *D-Lib Magazine*, March, available at: www.dlib.org/dlib/march96/briefings/03indexdata.html

(The) Joint Information Systems Committee (n.d.), "Information Environment (IE) Service Registry", available at: www.jisc.ac.uk/index.cfm?name = project_iesr

Lohrum, S., Schneider, W. and Willenborg, J. (1999), "De-duplication in KOBV", available at: www.zib.de/PaperWeb/abstracts/SC-99-05/

Needleman, M. (2003), "The NISO metasearch workshop", *Serials Review*, Vol. 29 No. 3, pp. 256-7.

NISO (n.d.), "Metasearch initiative", available at: www.niso.org/committees/metasearch-info.html

OAI (2003), "The Open Archives Initiative Protocol for Metadata Harvesting", available at: www.openarchives.org/OAI/openarchivesprotocol.html.

Peterson, C. (2003), "Metasearching in the Lone Star State", *Library Journal*, Winter, available at: articleid = ca266434">http://libraryjournal.reviewsnews.com/index.asp?layout = article&articleid = CA266434

Sadeh, T. (2001), "MetaLib and SFX: managing heterogeneous resources in the scholarly environment", paper presented at CASLIN 2001: Library of Academy of Sciences of Czech Republic and National Library of the Czech Republic, Beroun, 27-31 May, available at: www.caslin.cz:7777/caslin01/sbornik/metalib.html

Sadeh, T. and Walker, J. (2003), "Library portals: toward the semantic Web", *New Library World*, Vol. 104 No. 1/2, pp. 11-19.

UKOLN (n.d.a), "Collection description focus", available at: www.ukoln.ac.uk/cd-focus/

UKOLN (n.d.b), "RSLP collection description", available at: www.ukoln.ac.uk/metadata/rslp/schema/.

Van de Sompel, H. and Beit-Arie, O. (2001), "Open linking in the scholarly information environment using the OpenURL framework", *D-Lib Magazine*, Vol. 7 No. 3, available at: www.dlib.org/dlib/march01/vandesompel/03vandesompel.html

ZING (n.d.a), "SRW – Search/Retrieve for the Web", available at: www.loc.gov/z3950/agency/zing/srw/background.html

ZING (n.d.b), "SRW – Search/Retrieve for the Web: SRW explain documentation", available at: www.loc.gov/z3950/agency/zing/srw/splain-doc.html