The Future of Digital Libraries

Harald Krottmaier
Institute for Information Systems and Computer Media
Graz, University of Technology
8010 Graz, Austria

ABSTRACT

Digital Libraries are omnipresent nowadays. Almost every single institution is hosting its own. Sometimes they even use their own servers and customized software to provide information to users. However, a single isolated Digital Library is valueless. It must be connected "to the world": to users as well as to content and services provided by other Digital Library systems. Since it is not possible for an information provider to host every single information of a specific topic locally at its server, a portal might be the right solution. In this article we will give an overview of some necessary features of Digital Libraries. Linking to electronic and traditional resources are essential and must be available to users of the systems. Interactive features such as adding annotations to existing material must also be available. A sketch of aspects about personalization will close the discussion.

Keywords: integration of services, problems of reusing content and services, personalization

1 INTRODUCTION

The technique of using portals to integrate content and services is a common approach for information providers. Single sources of information are adopted and rendered using different styles. Users usually do not notice that this adopted information is not stored physically on the portal-system because the look-and-feel of the content does not differ to other content provided by the system.

Since the portal might be seen as "single entry to the web" it is obvious, that a highly scalable and powerful system must be available to provide users with a single sign-on system. This system should be able to integrate heterogeneous types of content and metadata. This might be technically done using highly sophisticated XML and XSLT processors. Many out-of-the-box systems (e.g. Cocoon, an open-source framework, see e.g. [1]) are already available to information providers.

Different types of services must be available to users of a portal. Services such as hyperlinking of content, without rights to change the content which will be linked, must be available. This technology is well known since Xanadu but rarely implemented.

One of the main advantages of using the portal approach in a Digital Library is the single point of personalization. Nowadays it is necessary to personalize every accessed Digital Library itself. This is time consuming and fault-prone. If interests change — and this happens regularly — this information must be updated in every single Digital Library.

It is not possible for a single library to store every single document about a specific area. Thus resources available in external systems — these systems are not limited to Digital Libraries — must be integrated and reused. A system should provide users with access to content regardless of storage-location. Document content as well as services provided by the remote Digital Library should be available to users. Unfortunately this portal-approach is not commonly used in Digital Libraries. Users are required to explicitly open another web-browser, connect to the selected Digital Library and repeat the search-query on the new system. This is ridiculous nowadays. The Daffodil-system (see e.g. [2]) supports the user in this aspect: search-queries are distributed to selected systems and the results are displayed in a single listing.

The portal-software should be responsible for distributing search-queries to all selected Digital Libraries and provide search-results to users. This list of results must be presented to the user in a usable and re-usable fashion, permitting sorting, email, saving and the like. Users are (usually) unconcerned with the particular details of the Digital Library hosting an article, the article itself being the primary object of interest to the user!

Using a portal-approach for a Digital Library implicitly leads to many problems, including access rights to con-

tent, access to services provided by the remote system, selection of remote systems, handing of the results of functions called at the remote system to mention just the most obvious ones.

In the next section we will take a look at a collection of features which should be available to users of a Digital Library. Thereafter we discuss aspects of personalization.

2 FOUR NECESSARY FEATURES OF DIGITAL LIBRARIES

In this section we take a look at some of the necessary features we expect in the field of Digital Libraries. Some of them are currently under development, others are in the design-phase. Very obvious features — such as providing content in different electronic document formats to serve different types of client — are not mentioned in the following discussion[1].

Links from and to Electronic Resources

When we take a closer look to articles available in electronic format we notice in the reference section of the article links to electronically available resources. This is a nice and easy to use feature. It helps users to explore the references easily.

In the area of scientific publishing a well known structure (i.e. abstract, content, special reference section) of the document is available. Related material and resources are listed in the reference section. According to the given style-guide of the documents the reference section is usually located at the end of each article. Each entry starts at a new line and begins with a pattern (often something like `[Name(s), Year]`).

However, to get more information out of a simple unidirectional link, the paradigm of links must be extended. Currently most links used in hypertext are unidirectional. The inherent information attached to such a link is: “We are using this as a reference”. This information is “nice-to-have”, but it does not substantial differ to information available in traditional printed versions of a document.

In some available (closed) systems links are created from the source *to* the destination and vice-versa. Internally two unidirectional links are created. Using this approach adds solid information to the destination of the link, i.e. the destination is referenced by some other work.

In our prototype we use a Hyperwave Information Server ([3]) where links are stored separately from the content in a link-database. Links are bidirectional, therefore no additional implementation effort is necessary to make “destinations” of links visible to the user. One may imagine that often used references are more valuable to uses. These references should probably been read.

At the moment bidirectional links are limited to locally published articles. Since links must point to objects stored in the database we have restricted targets of links to intra-server objects. However, surrogate-objects may be created in the database representing any type of reference. In the used database system it is possible to create so called “remote-objects” with special attributes. These surrogate-objects will be used as link-targets, therefore bidirectional links may be created to a target of any type. When extending the current prototype attention must be payed to duplicated objects representing the same remote reference. This might be tricky because content is duplicated and accessible via different URLs. In this case similarity algorithm will help in finding the duplicated contents.

As already mentioned links are stored in a link-database. This implies, that it is possible to add links to documents which do not support links per se. To give an example: PostScript was designed to be printed on high quality printers, not to be viewed online. Therefore Adobe did not take hyperlinks into consideration when specifying the document format. However, in Hyperwave it is possible to add links even to PostScript documents, because links are stored in the link-database rather than the document itself. Obviously, it is necessary to use a specially designed viewer to present the links to the user.

PostScript is now superceded by Adobe’s Portable Document Format (PDF) where links are specified and viewers are available for many operating systems. Features of PDF include annotations, minor editing, highlighting etc. To create links in PDF it is necessary to use specially designed programs such as Adobe’s Acrobat. Unfortunately this program is not available for every client platform, as an example, clients on Unix-platforms are precluded from the editing process.

Links to Traditional Resources

In digital journals and publishing systems articles are already parsed and links from the reference in the

[1] Extended versions of the discussions are published in [10], [7], and [11]

text to the reference-section are automatically generated. These links are called ”intra-document links”. As already argued in [10] intra-document links are not enough. It is often desirable to explore the reference itself. Electronically available references may be linked as described in the previous section. This feature is already implemented in some Digital Library systems (e.g. in ACM-Digital Library).

Traditional material must be handled in a different way. Depending on personal preferences these links should be either directed to book-resellers — such as Amazon — or directed to the catalog of local libraries. Currently we are implementing such a linking feature to an electronic catalog available at Graz, University of Technology.

The local catalog in Graz is stored in the ALEPH-catalog system, an implementation of OPAC. ALEPH is widely used in Europe, therefore it is very likely that the upcoming implementation will be reusable for universities. The key issue in linking to traditional material is the unique identification of the resource. The following identification systems are currently available for content and should therefore be used to identify traditional resources:

International Standard Serial Number (ISSN):
[5]: ”The ISSN is an eight-digit number which identifies periodical publications as such, including electronic serials. More than one million ISSN numbers have so far been assigned. It is managed by a world wide network of 75 National Centers coordinated by an International Center based in Paris, backed by UNESCO and the French Government. The ISSN is used by various partners throughout the information chain: libraries, subscription agents, researchers, information scientists, newsagents (through its barcode version).”

International Standard Book Number (ISBN):
ISBN is a unique identifier for books used in many countries. As mentioned in [4]:

”The ISBN is a unique machine-readable identification number, which marks any book unmistakably. This number is defined in ISO Standard 2108. The number has been in use now for 30 years and has revolutionized the international book-trade. 165 countries and territories are officially ISBN members. The ISBN accompanies a publication from its production onwards.”

With the appropriate ISBN for books or ISSN for periodicals it is possible to identify the respective entry in an online-catalog of books. Unfortunately these two tags are not available in every reference entry. In reference management systems (such as BibTeX) this attribute is not mandatory.

Services like this — i.e. creating links to material available in traditional libraries — might be extended to all features available to users of the catalog-system. Typical features such as management of users and books, reservation of books, notifications of new arrivals etc. should also be available in the reference list of an electronic document.

Active Annotations

Annotations are basically notes of different type attached to content. First implementations supported just text-based notes. It is possible to attach arbitrary multimedia documents as annotations when using a Hyperwave Information Server as database.

Objects are linked via typed-links (link-type is “annotation”) to the source object. To add even more information to links (such as: this annotation is a question, or answer, note etc.) we added additional attributes to the link object. With these additional attributes it is possible to create *Active Annotations* and thus facilitate discussion about articles. The corresponding author of an article is alerted by the system via an email about the newly created annotation, making it possible to answer questions or explain some thoughts in more detail. The publishing process therefore does not end after a successful submission of an article. It continues as long as questions are asked or comments are made.

Additional attributes of annotations are related to access-control (read, write, modify, ...) of those annotations. Three types of annotations should be possible: private-, group- and public- annotations. Generally every user must be able to add at least private annotations to an article. These annotations are only visible to the user who created them. It must be possible for this user to change or delete the annotation.

Some users (e.g. users from a research-group) must be allowed to read and write group-annotations. This is necessary for group members who collaborate on a specific topic or task. A closed mini-discussion about an article with a defined group of users is therefore possible. As implemented in many discussion forums, it must not be possible to edit annotations after they have been annotated by another user for obvious reasons (stable annotations).

Public annotations are visible to every user, therefore

the system and the administrator should take care of this feature. In some circumstances (e.g. in a learning environment) public annotations must be restricted. Readers should, for example, be forced to read certain parts of a document before being allowed to annotate an article or ask questions about an article. Online user-tracking and user activity-logging are evident pre-requests to implement such features.

To get rid of so called *spam-annotations*, i.e. annotations, which contain material unrelated to an article, annotations should also be integrated in a quality or relevance rating system. Users should be able to judge annotations as "great" or "poor", or according to some suitable set of criteria. Thus many "poor" annotations will disappear from the listing of annotations and users can therefore be spared the time reading them.

Some users are skilled in reading a hyper-linked version of an article on screen, others want to print out the PostScript or PDF-formatted version of the article. Different document formats are therefore offered to the user, and in J.UCS for example, HTML, PostScript and PDF are the supported document-formats. The contribution of an author is coded in these different document-formats and is then stored in a so called article-collection. Abstracts are converted to HTML and are available free to users. Annotations to an article are attached to the article-collection and are therefore available for viewing in all the different document-formats. It is possible (and already implemented in a prototype) to simply add a page of annotations to the PDF-formatted paper.

Reuse of Available Material

The idea of transclusions was born in 1960, when Ted Nelson invented Xanadu, a revolutionary information- and document management system. "Reuse without duplicating document fragments with the original context available" ([14]) is the key issue of the transclusion concept. Newly assembled documents are also called *compound documents*, on the other hand, documents reused in one or more compound documents are called *transcluded documents*.

As there is no duplication of document fragments when reusing content many advantages over ordinary cut-and-pasting of data arise (see e.g. [7]). To mention just some of them:

Intellectual Property: Parts of a document are not "stolen" but simply "reused in another context". It must be visible to the reader of a document that parts of the document are no original contributions, but references or inclusions of some other work. Dependent on the electronic format of the compound document this is easily implemented by adding hyperlinks to the original work before and after the inclusion of the reused material. Please note that in HTML documents it is possible to reuse whole images in any context stored on any server system. The reader of the document will not notice this fact without reading the source-code of the HTML document.

Disk Space: Since parts of a document are not copied into a new document there is no need to save the reused part more than once. Surrogates of the reused parts must be stored in a predefined format. There will be no waste of disk space, if the surrogate definition requires less disk space than the referenced part. It is obvious that transcluded documents must be available to the system when assembling the document. Without coping the reused part and without sophisticated caching mechanisms this fact may limit the applications of transclusions to intranet or even intra-server applications.

Update: There is simply no need of manually updating referenced parts because the content is always requested from the *original source*. Since this is an advantage in many applications (e.g. a collection of different course material which is automatically up-to-date) there are drawbacks in some other applications where many people are responsible for the content and access control is not an issue. Some action must be taken if reused parts are changed. At least the author of the compound document must be informed (e.g. via email) by the system if there are any changes in the reused part. While there is no need to update a document it might be a time-intensive task to assemble a compound document. Therefore reusing a document which reuses parts of another document (which itself reuses parts of other documents etc.) must be limited to a certain level. Nevertheless, improvements in network infrastructure and processing speed will reduce this problem.

Two Way Reading: When displaying the transcluded document the system must add additional navigation facilities around the quotes in the compound document. This action is necessary to indicate the reuse of content (see "Intellectual Property" above). Links to the original source of the transclusion are provided. This makes it possible for the reader of a document to take a look at the original context of a quote. The reader of the transcluded document on the other hand has the ability to explore which parts of a document are no original contributions. If a fragment of a document is reused it is

very likely that this fragment is important!

These are some of the most important advantages and drawbacks when using transclusions. Authors of compound documents must be aware of the risks when using this technology. The information system where these documents are stored must support authors in many ways (i.e. creation of transclusions, notification of changes, types of changes, etc.).

There are many applications of transclusions (see e.g. [8] for detailed explanations), e.g. in the field of electronic publishing, and in the creation of course material.

3 ASPECTS OF PERSONALIZATION

Personalization is not just about content and interface adaptation, but also about links. Adding a link to some content is like adding an annotation to the content. When user-access-rights are attached to link-objects, it is possible to simply deactivate unrelated links. To give an example of a Digital Library used as background library in an e-Learning course: students new to a course may need more links from the content to a dictionary or thesaurus explaining a word as compared to more experienced students from later course stages.

To enable personalization the user must have the possibility to express attributes of the profile. User-models are abstract representations of users. One can see implementations of user-models as "metadata about users". User-models must be effective and efficient, but most important: the model must be accepted by the user. Acceptance is achieved by supporting users in many respects.

In [13] the aims of user-models are explored in detail. Let us here summarize and comment the results.

active information providing: the user-model should describe topics of interests of users. Therefore the system may actively inform the user via some notification service about new and related information. This technology is known as *push-technology*.

reduce information overload: a study ([12]) produced by faculty and students at the School of Information Management and Systems at the University of California at Berkeley shows the current information overload. The team attempted to measure how much information is produced in the world each year. While in the year 2000 the amount of unique information was between 1 and 2 exabytes, the amount was about 5 exabytes in 2003. 92% of the new information was stored on magnetic media, mostly in hard disks. Film represents 7% of the total, paper 0.01%, and optical media 0.002%. This is about 800 MB per person. It is not possible to implement a personalization concept (e.g. described in [9]) without the use of user-models.

support of handling: depending on experience and/or education of the user, the system may help with appropriate hints.

improvement of queries using users-knowledge: if the number of result-objects is too high (or low), the system may change the query-expression to create an optimal set of results. To give an example: when too may objects are found by the search-engine, they may be filtered by e.g. preferred authors of the user etc. to reduce the number of objects. If the query is to narrow it should be widen by the system automatically.

improvement of queries using expert-knowledge: the system may use profiles of experts in a specific topic to improve search-queries, e.g. by adding related terms — which are unknown to the user — to the query.

ranking of results: search-results are usually ranked because of relevance to the query-string. Results may be clustered related to interests of the user. If user A is interested in topic X, Y and Z then the results should be grouped in these 3 topics.

team-support: to extend the knowledge of a singe-team member it may be of benefit to other members to know about the "knowledge-space" (i.e. user-profile) of other members in the same team.

In the next paragraphs we summarize properties to be stored in a user-profile (and therefore must be modeled in a user-model). It is clear, that many properties must be available in the context of groups. Reflecting on the categories of data, it becomes clear that users are not interested in sharing provided information with any other party.

personal data: nickname, firstname, lastname, date of birth, gender, education, and other related information. Similar data for groups.

interests: obviously users (or group) interests are of interest to the system. This includes keywords of topics, clusters of related classification-systems (e.g. ACM-classification system) etc.

personal preferences: this includes e.g. color-schema to use and customization of different services provided by the system.

personal experience: if the user is an expert or novice with the system itself. How much help and which functionality should be provided by the system.

The category "personal data" transparently shows problems about access for other service providers to attributes described there. Attribute "nickname" or "gender" may be accessed, but users may not allow certain information- or service providers access attributes such as "date of birth" or "education". Therefore a very flexible architecture must be available for users to express their needs.

4 CONCLUSIONS AND FUTURE WORK

This article showed some necessary features and aspects of personalization in the field of Digital Libraries. Many features are already described in detail in the references. Bidirectional-linking, i.e. connecting different parts of content, is essential to support users in exploring information. It must be visible to the user, which documents are used in which context. Transclusions are an extension to references. With this technology it will be possible to include parts of content in new documents.

We are currently in the design phase of a large Digital Library project and are evaluating different systems to implement a portal at Graz, University of Technology.

ACKNOWLEDGMENT

This work is partly supported by DELOS, a Network of Excellence on Digital Libraries (EU FP6, G038-507618).

References

[1] Apache Cocoon Project `http://cocoon.apache.org` (2004/05/15) .

[2] Fuhr, N., Klas, C-P., Schaefer, A and Mutschke, P. (2002) Daffodil: An Integrated Desktop for Supporting High-Level Search Activities in Federated Digital Libraries Proceedings of 6th European Conference on Digital Libraries, ECDL 2002

[3] Hyperwave `http://www.hyperwave.com` (2004/04/19) .

[4] ISBN (2003). International standard book number. `http://www.isbn-international.org` (2004/03/23)

[5] ISSN (2003). International standard serial number. `http://www.issn.org` (2004/03/23) .

[6] J.UCS (2004). Journal of Universal Computer Science. `http://www.jucs.org` (2004/04/05)

[7] Krottmaier, H. and Maurer, H. (2001). Transclusions in the 21st Century. Journal of Universal Computer Science, 7(12):1125–1136. `http://www.jucs.org/jucs_7_12/transclusions_in_the_21st` (2004/02/12)

[8] Krottmaier, H. and Helic, D. (2002). Issues of Transclusions. Proceedings of E-Learn (E-Learn 2002), page 1730-1733

[9] Krottmaier, H. (2003). Stop Reading (Useless Parts)! Proceedings of the 7th ICCC/IFIP International Conference on Electronic Publishing (ELPUB 2003)

[10] Krottmaier, H. (2003). Links to the Future Journal of Digital Information Management

[11] Krottmaier, H. (2004). The Need for Sharing User-Profiles in Digital Libraries Proceedings of the 8th ICCC/IFIP International Conference on Electronic Publishing (ELPUB 2004)

[12] Lyman and Varian (2003). How Much Information `http://www.sims.berkeley.edu/how--much-info-2003` (2004/05/07)

[13] Möller, G. (1999). Brevis: Benutzermodelle in IR: interne Repräsentation, Erstellung und Visualisierung Master Thesis, Oldenburg

[14] Nelson, T. (1995). The Heart of Connection: Hypermedia Unified Transclusion. *Communications of the ACM*, 38:31–33.