ARTICLES

D-Lib Magazine

March 2001

Volume 7 Number 3

ISSN 1082-9873

Open Linking in the Scholarly Information Environment Using the OpenURL Framework

Herbert Van de Sompel
Cornell University
herbertv@cs.cornell.edu

Oren Beit-Arie
Ex Libris (USA) Inc.
oren@exlibris-usa.com

Abstract

This paper provides insights into the concepts underlying the OpenURL framework for open reference linking in the web-based scholarly information environment. The discussion starts by briefly reiterating the problems with reference linking that were initially described and later addressed as part of the SFX research. Some notions crucial to the understanding of the issues at stake are briefly described: extended service-links, closed and non-context-sensitive linking, open and context-sensitive linking. Then, the paper details the OpenURL framework and reports on its current deployment in the scholarly information environment. As an illustration of the penetration of the OpenURL framework, special attention is accorded to a prototype in which the OpenURL framework is integrated with the DOI/CrossRef linking solution.

Introduction

This paper provides insights into the concepts underlying the OpenURL framework for open reference linking in the web-based scholarly information environment. The OpenURL framework has its roots in the SFX-research conducted by Herbert Van de Sompel and his colleagues at Ghent University, Belgium, from 1998 to 2000.

This discussion starts by briefly reiterating the problems with reference linking initially described and later addressed as part of the SFX research. Notions at the core of the SFX research are described: extended service-links, closed and non-context-sensitive linking, and open and context-sensitive linking.

Next, the paper details the OpenURL framework for open linking and reports on its current deployment in the scholarly information environment. A key component in the OpenURL framework is the OpenURL. The OpenURL provides a standardized format for transporting bibliographic metadata about objects between information services. As an illustration of the ongoing interest in the OpenURL framework, special attention is accorded to a prototype in which the OpenURL framework

is integrated with the DOI/CrossRef linking solution. This paper focuses on concrete solutions, leaving more abstract dimensions of the context-sensitive resolution mechanisms for a forthcoming paper.

Extended service-links

The term *extended service-link* was introduced (Van de Sompel and Hochstenbach 1999a) to refer to a link in scholarly information resources that goes beyond the classic notion of a reference link, which is typically understood to be a link from metadata to the full-content described by the metadata. The notion of *extended services* was introduced to refer to a bundle of such extended service-links, thereby stressing that:

- Many services that go well beyond the classical reference link can be delivered for a given metadata description;
- Delivery of such services becomes increasingly important to adequately navigate the web-based scholarly information environment.

Extended service links that are commonly encountered when exploring scholarly digital libraries lead:

- From a record in an abstracting and indexing database (A&I) to the full-text described by the record;
- From a record describing a book in a library catalogue to a description of the same book in an Internet book shop;
- From a reference in a journal article to a record matching that reference in an A&I database.

More creative extended services can be imagined, and have already been introduced in operational environments that use the SFX server software. Such service links lead:

- From a citation in a journal article to a record in a library catalogue that shows the library holdings of the cited journal;
- From a journal title to impact factor information in ISI's Journal Citation Reports;
- From a stock ticker symbol found in a record from the ABI/Inform database to on-line stock information;
- From title words or subject headings of a scholarly article to related information in Internet search engines.

Closed and non-context-sensitive linking frameworks

It has been argued (Van de Sompel and Hochstenbach 1999a) that the way in which the information industry provides such extended service-links is not satisfactory because the manner in which this is done is:

- Non-context-sensitive: the established linking frameworks provide service-links that fail to take into account the context of the user who follows a link;
- Limited in scope: the established linking frameworks are narrowly focused, both regarding the types of extended services that are being provided as well as regarding the action radius of those links, which is often limited by the scope of business agreements between information providers;
- Closed: the established linking frameworks do not allow third parties -- such as the user's library -- to compensate for these problems by enabling the delivery of self-defined context-sensitive extended service-links.

Therefore, such linking frameworks have been referred to as *closed* and *non-context-sensitive* linking frameworks. Although these terms have quite different meanings, they have frequently been used as synonyms to refer to the combination of the problems described above.

A specific instance of the problems introduced by non-context-sensitive linking frameworks is known as the *appropriate copy problem* for links to full-text (Caplan and Arms 1999). This problem refers to the fact that such linking frameworks fail to provide links that lead from a citation of a journal article to the appropriate full-text copy of that article. A full-text link typically leads to a publisher-defined default copy of the article, which usually resides in the publisher's repository. However, access to the copy of the article that is appropriate in the context of a certain user may very well require the provision of an alternative link:

- The user's library may hold a subscription to the electronic journal in which the article was published via an intermediary service. In this case, resolving the reference link to the publisher's default copy may result in a denial of access, because the publisher may not be directly aware of each individual entitled to have access to the desired article. Still, via the library's subscription, the user has legitimate access to the content of the article. In this example, the alternative resolution must take into account the fact that this particular user is allowed to access the article via an intermediary, not directly via the publisher. Hence, the resolution of the reference link must lead to the intermediary's services, where an alternative copy of the article may reside, or where access to the publisher's copy may be cleared.

- The user's library may store a copy of the article in a local repository. Again, resolving the reference link to the publisher's default copy may result in a denial of access. Even if access to the publisher's copy would be granted, such resolution is not the preferred one, because it bypasses the library's motivations to actually store content locally. Hence, the preferred resolution should lead to the appropriate copy of the article in the library's repository.

- The user's library may not have licensed access to an electronic copy of the article. However, the library may have other means to provide the required content to the user: it may hold a paper subscription to the journal; it may be able to handle a document delivery request; etc.

It has been shown that the appropriate copy problem for links to journal articles is in fact only an instance of a more general problem (Van de Sompel and Hochstenbach 1999a) that -- by analogy -- could be named the *appropriate extended-services* problem. This problem refers to the fact that established linking frameworks, in general, fail to take into account the context of the user in the provision of extended service-links. A simple illustration of this lack of context in the provision of extended services can be seen in the information industries' widespread use of the PubMed full-featured Entrez link-to mechanism. Publishers take advantage of the Entrez system to include links that lead from references in their articles into the PubMed database. These service-links -- leading into the default PubMed implementation of the Medline database -- are definitely attractive to the user. However, users may find it more attractive if these links would lead into the *appropriate* Medline implementation provided by their library to which the users are accustomed. And users may find it even more attractive if additional service links would allow them to look up the reference in their local OPAC system, etc. Many more examples have been given (Van de Sompel and Hochstenbach 1999a, Van de Sompel and Hochstenbach 1999c) that illustrate the common lack of context-sensitivity in operational linking solutions. These problems extend well beyond the appropriate copy problem.

The SFX research

In response to the dissatisfaction with closed and non-context-sensitive linking frameworks, research has been conducted to design a proper architecture that will address the problems described in the above. In this research, which went by the working title SFX (for Special Effects), the term *open and context-sensitive linking framework* has conveniently been used to refer to a proposed architecture that could make the difference. In the SFX research on context-sensitive linking, the notion of context has been related directly to a user's institutional affiliation. It encompasses contextual elements such as:

- The content that is accessible to the user via the institutional digital library:

  - The collection of A&I and citation databases to which the user has access;
  - The collection of electronic journals that are accessible to the user;
  - The OPAC system of the user's institution;
  - E-print systems that are accessible to the user;
- The specific implementation of access to the above content, e.g., which implementation of a specific A&I database is used;
- The user's preferences regarding the interaction with the digital library collection.

Over the course of three complex experiments, the SFX research demonstrated the feasibility of an open and context-sensitive linking framework for the scholarly information environment (Van de Sompel and Hochstenbach 1999b; Van de Sompel, Hochstenbach 1999c; Van de Sompel and et. al. 2000). As part of the research, a linking server-software by the name *SFX server* was implemented.

The OpenURL framework in the scholarly information environment

The fundamental notion introduced in the SFX research is that of *disconnecting* the provision of linking services for a work from the description of the work, as presented to users in electronic information resources (Figure 1 and Figure 2). A reference link associated with metadata describing a scholarly work should not be considered part of the metadata, but rather as a service that builds upon the metadata. Once this conceptual disconnection is made, the next logical step is to realize that such linking services for a given metadata description can be provided by many parties: it is no longer necessary that the information service that presents metadata to users be the sole provider of service links for that metadata. Rather, *overlay service components* can be introduced to provide extended linking services for scholarly works described in the various resources that are accessible to the user. If users -- irrespective of the resource they are navigating -- can call upon their chosen service component, it becomes possible to create a consistent linking environment that extends beyond the typically narrow focus of the linking solutions glued to a single resource. Moreover, if such a service component appropriately takes into account the user's library collection, the linking services will lead to *appropriate* services.

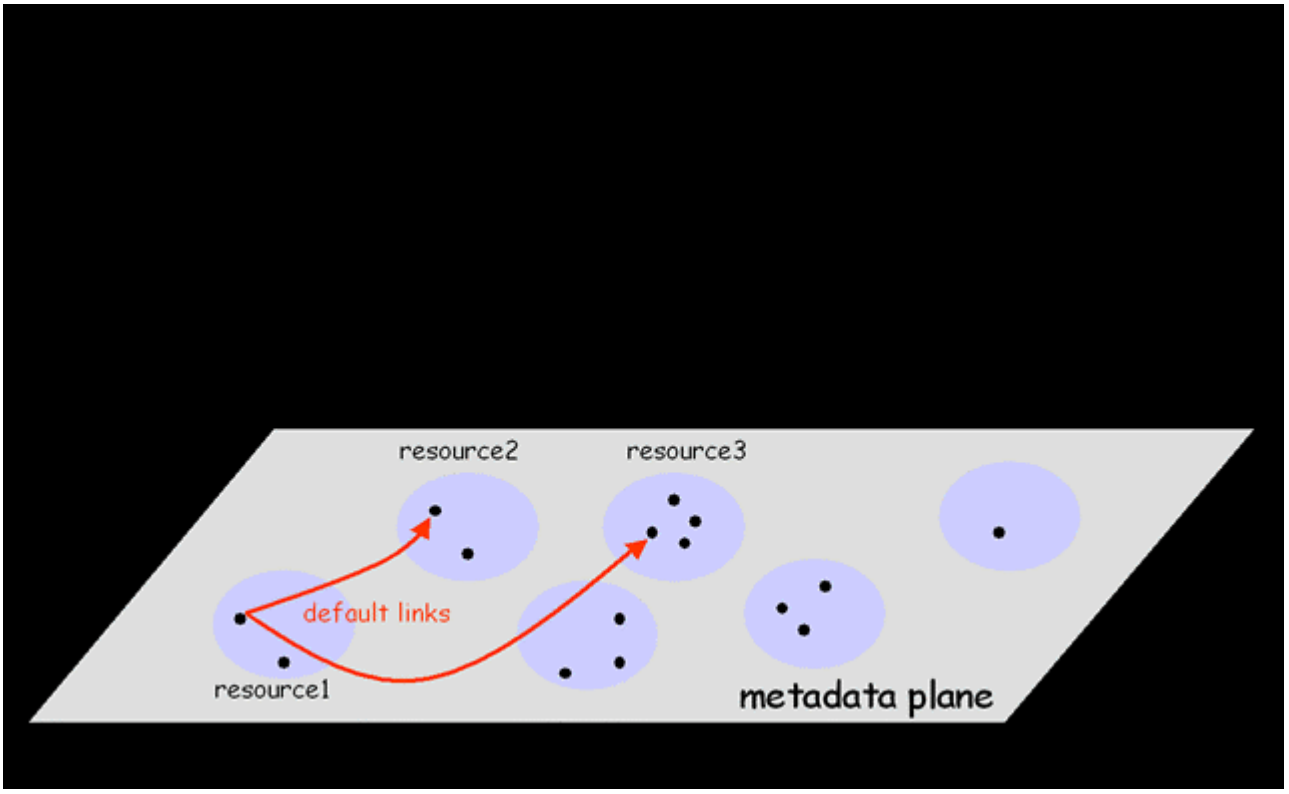

Figure 1: Default links provided by closed linking solutions tied-in with a specific information resource can be regarded as part of the metadata for which the links are provided. The links have a limited action radius, and typically are not sensitive to the context of the user. Moreover, since each resource has its own linking methodologies, the user's navigational experience is not consistent.

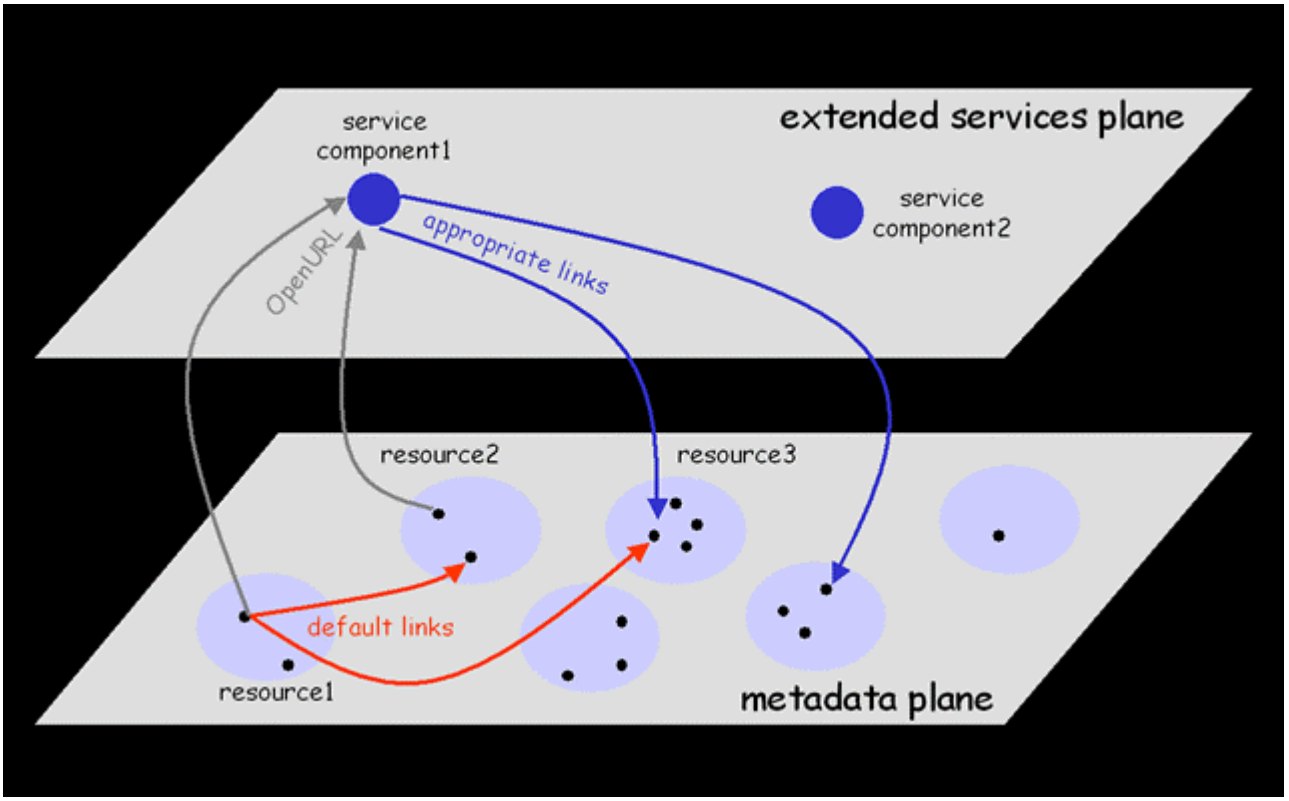

Figure 2: By disconnecting the provision of metadata from the provision of linking services, users can indicate a preferred service component that will provide links irrespective of the resource they are navigating. Typically, this service component is managed by the user's library and can provide the user with consistent links across its collection. The OpenURL is the hook connecting the resources holding the metadata with the service components that will provide the user's requested *appropriate* services. The OpenURL is the enabling mechanism for open linking from resources.

Implementation of an *open linking* framework that adheres to these concepts is fairly simple. The authors and an ever-increasing number of advocates evangelize an actual implementation in the scholarly information environment. This implementation is commonly referred to as the OpenURL framework. In the OpenURL framework, information resources allow for open linking by including a hook along with each metadata description that they present to users. This hook presents itself in the user's browser as an actionable URL called the OpenURL. It is a general rendering of the SFX-URL that was introduced in the SFX research (Van de Sompel and Hochstenbach 1999c). An OpenURL for a work contains a number of parameters that are relevant to the functioning of the framework, formatted in a standardized way (see Table 1). Most importantly, this OpenURL contains identifiers, metadata and/or a pointer to metadata of the work. The target of the OpenURL is the user's overlaying service component, typically operated by the user's library.

By clicking an OpenURL for a work, the user requests that the service component deliver extended services for that work. The service component takes the OpenURL as input and collects metadata and identifiers for the work. It can do this by directly parsing such information from the OpenURL and/or by fetching it using the metadata pointer that was provided in the OpenURL. This pointer can lead into the original resource or into another one. Once identifiers and metadata are collected, the service component will evaluate them and provide extended service links to the user. When the service component is appropriately tailored, these links will be sensitive to the context of the user. As has been

described at length in (Van de Sompel and Hochstenbach 1999c), the amount and quality of metadata that the service component is able to collect plays a crucial role in the nature and quality of the extended services that can be delivered for a given object. The same paper also gives insight into the ways in which an SFX server -- which was the first service component of this kind -- actually generates context-sensitive services.

It is important to emphasize that the OpenURL is the enabling mechanism for the *outbound* interface between resources and services components: the OpenURL is the interface leading from the resources to the service components. Inbound links, leading out of service components into information resources, are based on whichever link-to mechanisms are supported by the respective resources. It is beyond the scope of the OpenURL framework to specify that mechanism.

From the above (Figure 2), it can be seen that, in essence, the OpenURL is an interoperability specification, which provides an interface between:

- Information resources that allow for open linking, by providing an actionable OpenURL as an additional hook associated with each metadata description that is presented to a user;
- Service components that provide extended services based on metadata and identifiers obtained through the OpenURL mechanism. The type and extent of the services offered depend on the metadata description for which the user clicked the OpenURL and the nature of the service component.

The OpenURL interface is designed so that a service component can passively receive as well as actively collect information to support the provision of extended services. While, so far, the usage of OpenURL has been restricted to scenarios in which the user interaction -- clicking the OpenURL -- has triggered the process of transferring such information to the service component, it can be seen that OpenURL could also be used in scenarios whereby a system takes on this role.

An appealing side-effect of the OpenURL framework derives from the fact that by clicking OpenURLs in distributed information resources, users are not only requesting extended services. They are also enabling their service component to record the request, and as such to accumulate a log of the users' actions across resources. Since it is fair to assume that users will mainly click OpenURLs for *preferred* works, the resulting log is a collection of user preferences that can become the knowledge base upon which to build recommendation services.

Citation (as found in an information resource):

Moll JR, Olive & M, Vinson C. Attractive interhelical electrostatic interactions in the proline- and acidic-rich region (PAR) leucine zipper subfamily preclude heterodimerization with other basic leucine zipper subfamilies. J Biol Chem. 2000 Nov 3 ; 275(44):34826-32. doi:10.1074/jbc.M004545200

Examples of possible OpenURL's that could be included by the information resource as a means to allow for open linking for the above citation. The OpenURL's that are shown comply with the current draft of the OpenURL specifications. They are encoded as HTTP GET requests:

http://sfxserv.rug.ac.be:8888/rug?id=doi:10.1074/jbc.M004545200

Legend:

red - BASE-URL of service component
blue - identifier of the resource where the user clicks the OpenURL
grey - metadata and identifiers

Generate more OpenURL's:

http://www.sfxit.com/OpenURL/generator.html

Table 1: Valid OpenURL's for a given citation, using the current OpenURL draft specifications

The deployment of the OpenURL framework

Starting in early 2000, a steady implementation of the OpenURL framework has taken place in the scholarly information environment. The following issues have played a fundamental role in the deployment of this open and context-sensitive linking framework:

- The provision of draft technical specifications -- known as the OpenURL draft -- aimed at enabling the implementation of an interoperable framework for context-sensitive provision of service links in the scholarly information environment (Van de Sompel, Hochstenbach, and Beit-Arie 2000);
- A persistent campaign -- by the Digital Library Federation, ICOLC, Ex Libris and the authors -- focused on raising awareness regarding the challenges that libraries face as a result of the lack of context-sensitivity in the established linking frameworks;
- A forum supported by NISO and the Digital Library Federation to discuss the issues involved in reference linking as well as their potential solutions;
- The beta-testing and marketing of the SFX server software, which was acquired by Ex Libris from Ghent University, in early 2000 (see http://www.sfxit.com/news/pr20000204.html). After an in-depth re-engineering of the software, the SFX server was beta-tested by several leading research libraries (Harvard University, Los Alamos National Laboratory, Caltech, Vanderbilt University, Indiana University - Purdue University, Ghent University) (see http://www.sfxit.com/news/pr20000612.html). At the time this article is being written, several beta-testers have moved SFX into production. Ex Libris is now actively marketing the SFX server software as an autonomous, reference linking service component that fits in the OpenURL framework. For libraries, the SFX-server provides a single point of administration of self-defined, context-sensitive, extended service links among the distributed information resources that form their digital collection. For users, the SFX-server is a hub for linking between those resources, in a manner defined by their library. It provides users with a consistent navigational system across their library's distributed collection.

As a result, at the time of writing, leading information providers such as ISI, Ebsco Publishing , Institute of Physics Publishing, SilverPlatter Information, Bell&Howell Information and Learning (ProQuest), ingenta, H W Wilson, Swets Blackwell, the Gale Group and Cambridge Scientific Abstracts have either made their resources compliant to the OpenURL draft or are committed to do so in the near future. Ex Libris' ALEPH500 library system is fully compliant, and other integrated library system (ILS) vendors have indicated their intention to become compliant in the near future. Also, more

subversive (Okerson and O'Donnell 1995) resources such as The Los Alamos preprint arXiv and the SLAC/SPIRES HEP database can present their metadata with OpenURL's. A list of compliant resources is currently maintained at http://www.sfxit.com/sources.html. The interest in the information industry is growing steadily, and many vendors are currently evaluating how they might best deploy the OpenURL in their services. Also, as will be shown below, a prototype currently investigates the integration of the OpenURL framework with the CrossRef DOI-based linking framework. The eventual outcome of this work may be the support of the OpenURL framework by the DOI-community. This would be another significant step forward in the deployment of the OpenURL framework.

Information providers involved in OpenURL-related activities unanimously agree that the implementation of the OpenURL capability in their resources is a straightforward matter. They consider it an attractive and cost-effective approach to serve their customers better by allowing them to integrate resources in self-defined manners. Typically, they offer the OpenURL open linking capability alongside their default, resource-specific links.

In December 2000, NISO approved a request for fast track standardization of the OpenURL submitted by the authors (see press release). It is expected that this standardization will lead to an even wider and faster acceptance of the OpenURL framework in the information industry. It is also expected that this will trigger offerings of other service components, besides the SFX server software. There are already several indications that libraries will consider support of OpenURL for open linking as an important criterion and as a requirement for the selection of electronic resources and library systems. Also, the NISO standardization process may look into extending the applicability of the OpenURL beyond the original scholarly application. In a forthcoming publication, the authors will present their perspective on the issues involved in such extension.

Integrating the OpenURL framework with the DOI/CrossRef linking system

Default resolution of DOIs

The CrossRef reference linking system builds on a collaboration between publishers. A collaborating publisher accords a digital object identifier -- DOI -- to a work in his collection. Next, the publisher deposits the DOI and metadata for the work in the CrossRef database. Other collaborating publishers can query this database that contains DOIs and corresponding metadata. Typically, those publishers will use metadata of a reference found in works they publish in order to obtain a matching DOI. For instance, if a publisher discovers the above citation in a publication, the CrossRef database can be queried using metadata elements of the citation to retrieve the DOI 10.1074/jbc.M004545200. Now, this publisher can include a DOI-based CrossRef link next to the citation in the publication. Such a link is an actionable URL of the form http://dx.doi.org/10.1074/jbc.M004545200.

When a user clicks this link [1 in Figure 3], the DOI proxy server that is the target of all links in the CrossRef/DOI namespace, will pass on the DOI -- 10.1074/jbc.M004545200 -- to the DOI handle [2 in Figure 3] server. The handle server resolves the DOI into a URL -- http://www.jbc.org/cgi/content/abstract/275/44/34826 -- to which the user is redirected [3 in Figure 3]. The association between a DOI and a URL is the result of a publisher registering a DOI for one of his works with the DOI handle system. Commonly, the URL registered by the publisher is that of a splash page that can lead to a copy of the full-content of the object represented by the DOI, which resides in the publisher's repository. The described resolution is closed and not-context-sensitive:

- If the user's library has an alternative copy of the paper, the link will not lead to it;

- If the user's library does not have access to a copy at all, the user will still be referred to the splash page;
- No other services are provided aside from the link to the splash page that can lead to the full-text.

These shortcomings in the current implementation have been described in (Van de Sompel and Hochstenbach 1999c) and are being addressed by the DOI-community (Paskin and Lannom 2000) as part of a larger effort to leverage the inherent design potential of the Handle System and the DOI, including multiple resolution and associated compulsory metadata. By integrating the OpenURL framework and the CrossRef/DOI linking system, it is possible to tackle these problems in a way that remains consistent across resources.

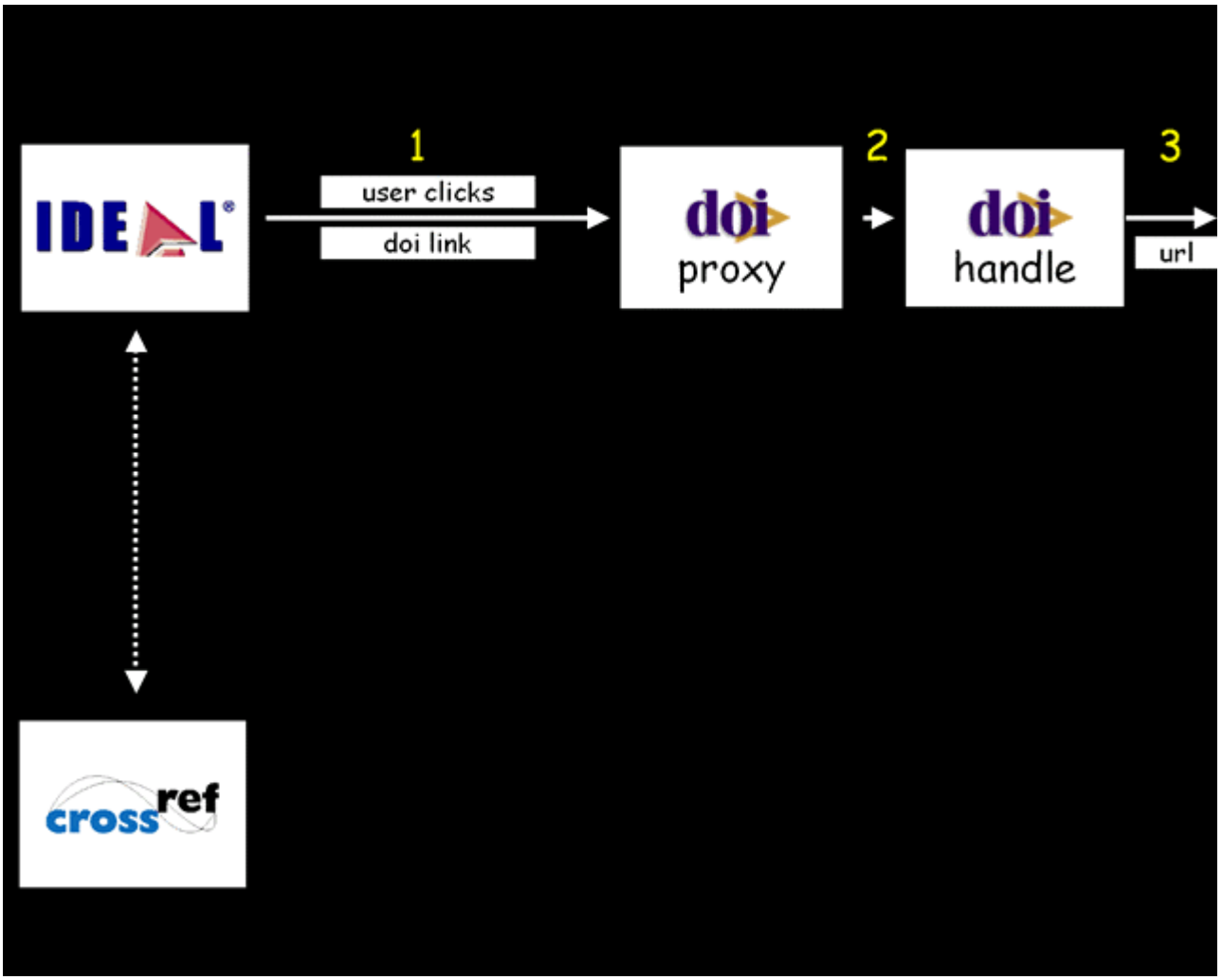

Figure 3: The default resolution of a DOI by the DOI handle system, through the DOI proxy.

Alternative resolution of DOIs

The obvious way to address the problem is to use the regular OpenURL approach: include the DOI -- 10.1074/jbc.M004545200 -- as one of the parameters on an OpenURL -- http://sfx1.exlibris-usa.com/demo?id=doi:10.1074/jbc.M004545200 -- that is shipped (along with a metadata record) to a user. As usual, the target of this OpenURL -- http://sfx1.exlibris-usa.com/demo -- would be the user's

chosen service component which could use the DOI contained in the OpenURL to query the CrossRef database for metadata, to fulfill its context-sensitive functions.

In an alternative approach (Figure 4), information resources do not necessarily need to include OpenURLs for objects that already come with DOI-based links. The approach uses the existing DOI-based links in those resources. For instance, the link found for DOI 10.1074/jbc.M004545200 in information resources remains http://dx.doi.org/10.1074/jbc.M004545200. As in the original DOI/CrossRef design, the DOI proxy remains the target of all DOI-based links. However, this proxy is now being enhanced to recognize whether an incoming request for DOI-resolution originates from users who have selected alternative resolution of DOIs via their OpenURL service component. The techniques that can be used to add this intelligence to the DOI proxy are basically the same as the ones described below, in the section "An open issue …". If the incoming request for resolution comes from a user without access to a service component, the DOI will go through the default resolution process as illustrated in Figure 3. If the request for resolution comes from a user with access to a service component [1 in Figure 4], the DOI proxy will hand the request over to the user's service component by transmitting an OpenURL [2 in Figure 4] that carries the DOI (e.g., http://sfx1.exlibris-usa.com/demo?id=doi:10.1074/jbc.M004545200). Next, the service component can use this DOI to query the CrossRef database for metadata [3&4 in Figure 4], if required. Next, it can provide the appropriate extended services to the user who clicked the DOI-based link [5 in Figure 4]. The reader can further explore alternative resolution of DOIs in Table 2.

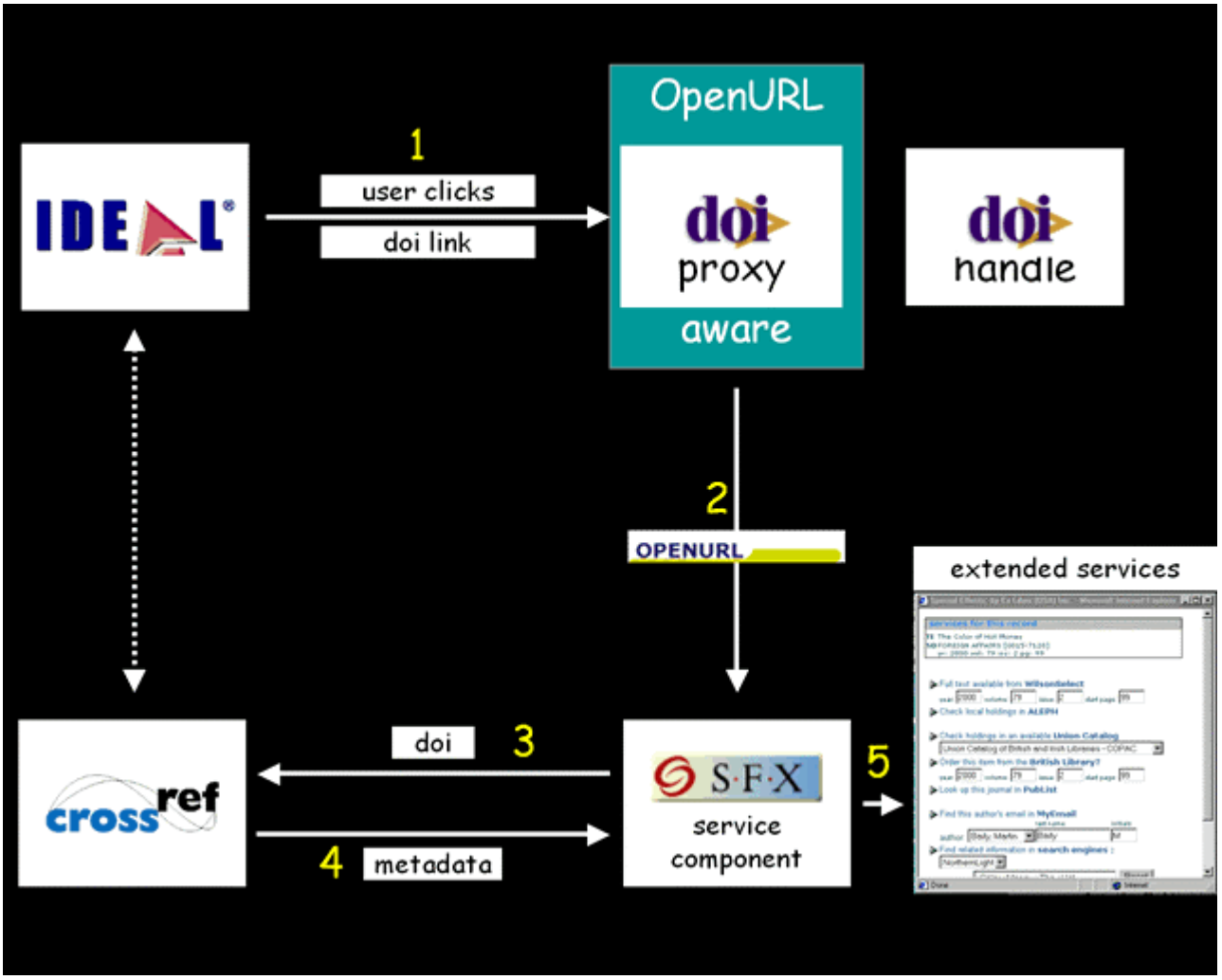

Figure 4: The alternative resolution of a DOI by a service component.

The latter approach is attractive in that it will lead to an appropriate resolution of DOI-based links irrespective of where those links are found, for instance, even if they appear in resources that do not explicitly support open linking or are in a PDF document stored on a user's PC. This property is achieved by taking advantage of the existence of a central spot in the DOI namespace -- the DOI Proxy -- where incoming requests for resolution can be redirected to a service component.

The authors first demonstrated the feasibility of the above scenario at the Reference Linking Workshop held at CNRI, in July 2000 (NISO 2000; Van de Sompel 2000). That demonstration led to broad acceptance by the meeting participants of the proposed approach. The approach is currently being explored in depth in a prototype implementation that involves CrossRef, the International DOI Foundation, CNRI, the Digital Library Federation, Ex Libris, the Los Alamos National Laboratory, OhioLink and the University of Illinois at Urbana Champaign. In the prototype, Los Alamos and OhioLink use the Ex Libris SFX server as alternative resolvers for DOIs, while Illinois is building its own OpenURL-compliant service component.

Step-by-step demonstration of the alternative resolution of a DOI (there are no guarantees regarding the longevity of this demonstration that builds

upon the prototype described in the above)

- Click http://dx.doi.org/10.1074/jbc.M004545200 to ask for the default resolution of DOI 10.1074/jbc.M004545200;
- Click here, to force the DOI proxy to set a cookie that allows it to understand that your service component is the demo SFX server from Ex Libris, with BASE-URL http://sfx1.exlibris-usa.com/demo;
- Now, click http://dx.doi.org/10.1074/jbc.M004545200 again, and receive services from your chosen service component, which does alternative resolution of the DOI 10.1074/jbc.M004545200;
- Make sure to click here to turn the cookie off again. If you don't, all future request for DOI resolution will lead into the Ex Libris demo SFX server.

Screencam (Windows avi format ; no audio; size = 3.8 Mb ; duration = 2 minutes 15 seconds)

This Screencam shows the local resolution of a DOI via the SFX service component of the Los Alamos National Laboratory:

- The screencam starts by showing the default resolution of a DOI: A user goes to an article published by the Institute of Physics Publishing (IOP) and scrolls down to the reference section. One of the references has a DOI-based CrossRef link which the user clicks. Via the default resolution of this link through the DOI handle system, the user is led into the full-text collection of Academic Press; actually to an abstract page of the Annals of Physics article that was referenced in the IOP article. However, for a researcher at the Los Alamos National Laboratory, this resolution is not appropriate because the Library at Los Alamos stores all Academic Press content locally.
- Next, the user returns to the opening page, where a facility is available that allows informing the DOI system about the fact that the user has access to a service component located at Los Alamos.
- Next, the screencam shows the alternative resolution of the DOI: The user goes back to the IOP article. Again, the DOI-based CrossRef-link is clicked and the resolution of the DOI is now handed over by the DOI-proxy to the Los Alamos SFX service component. Behind the scenes, that service component fetches the metadata -- that corresponds with the given DOI -- from the CrossRef database, and then presents a list of extended services to the user. The first one is a link to the appropriate copy of the Academic Press article stored in the Los Alamos repository. But other, appropriate extended services are presented: an option to do a federated search across all A&I databases stored at Los Alamos using author names of the paper cited in the IOP article; an option to look up citations of that paper or of its authors in the Los Alamos implementation of the ISI Citation Databases; a facility to search for the authors' e-mail address, etc.

Table 2: Exploring alternative resolution of DOIs

An open issue in the deployment of the OpenURL framework

An important element in the deployment of the OpenURL framework is the creation of a means to match users with their chosen overlaying service component. This is necessary in order to assure that the user's click on an OpenURL ends up at the chosen service component. Unfortunately, in the current

Web-infrastructure there are no solid mechanisms to achieve this. The problem is related to recognizing user credentials across organizations, which itself is still a largely unconcurred domain. Several scenarios can be imagined to address this issue. They differ in the division of the responsibility between web-client and server systems, for the determination of the location of the user's service component. Addressing such scenarios is beyond the scope of this paper; they will be discussed in a forthcoming publication.

In the current deployment of the OpenURL framework, the issue is addressed on a resource-by-resource basis and the problem is transformed into: how does a user notify a specific information resource about the location of his preferred service component, or -- looking at the same issue from another perspective -- how can an information resource introduce an OpenURL targeted at the *appropriate* service component? Actually, this problem is twofold:

- The information resource needs to know whether a user actually has access to a service component. If not, the provision of an actionable OpenURL seems senseless;
- If the user has access to a service component, the information resource needs to know the server location. The service component's location is used as the BASE-URL -- that is the http://<host>:<port>/<path> in (Berners-Lee and others 1994) -- of the OpenURL to be inserted for each object.

In the current rendering of the OpenURL framework, these issues are addressed in pragmatic ways. The most popular solutions so far are:

- Including the BASE-URL in the user's profile if one is maintained by the information resource;
- Forcing the information resource to write the BASE-URL as a cookie in the user's browser by means of the CookiePusher mechanism, which has been described at length in (Van de Sompel and Hochstenbach 1999c) and (Van de Sompel and Hochstenbach 2000) ;
- Deriving the BASE-URL from a table that links IP addresses with BASE-URLs.

Conclusion

It seems reasonable to conclude that the future of open linking in the web-based scholarly information environment looks bright.

Libraries are looking for appropriate ways to link across information resources, irrespective of their providers, and have been pressuring information providers to deliver solutions. The OpenURL framework provides a cost-effective manner for information providers to respond to this demand. It is easy for them to implement, and it serves their customers' needs by finally allowing them to play an active role in linking, and by offering them the ability to use extended services as a run up to the creation of an automated reference desk (Arms 2000).

The standardization of OpenURL specifications by NISO will eventually broaden the number of service components on the market. Ex Libris currently owns the only truly operational service component for open reference linking, so it is not surprising that they have actively promoted the adoption and standardization of the OpenURL. However, the OpenURL draft specification was made public at the outset of the deployment of the OpenURL framework, thereby allowing interested parties to create competing service components. Also, from the outset, interested parties have actively been encouraged to provide input into the draft specification. The upcoming standardization of OpenURL by NISO will no doubt stimulate potential providers of service components to move into action.

The open issue of matching a user with a service component requires attention, but should not prevent

the further deployment of the OpenURL framework. It is unfortunate that the Web does not provide an adequate infrastructure to tackle the issue. On the other hand, there are a number of pragmatic solutions to the problem. The nightmare of IP-based authorization has not kept libraries from delivering information services to their users. Therefore, one should feel confident that also this issue will be surmountable.

Another issue may be related to the innovative nature of the OpenURL framework and the need for information providers and libraries alike to comprehend the benefits offered. While some information providers are still evaluating their position regarding open linking, a growing group has fully embraced the opportunities that OpenURL presents. The OpenURL framework frees information providers from dealing with localization issues and allows them to provide better services to their users. Also, the framework increases the visibility and accessibility of their resources by providing the facility to link across resources. While -- at first glance -- this means a user can easily leave an information provider's resource (a concept that some information providers find threatening), it also means many users can easily enter that resource. Librarians who have adopted linking solutions within their environments are the first to appreciate the value now offered through the OpenURL linking framework, especially the value of optimizing their link management. The experience of the last half-year of the deployment of the SFX server and the OpenURL framework clearly shows that if any initial hesitation still exists, more often than not it is being replaced with active support and even enthusiasm.

References

Arms, William Y. 2000. Automated Digital Libraries: How Effectively Can Computers Be Used for the Skilled Tasks of Professional Librarianship? *D-Lib Magazine*. 6(7/8). <http://www.dlib.org/dlib/july00/arms/07arms.html>

Berners-Lee, T.; Masinter, L., and McCahill, M. 1994. RFC1738: Uniform Resource Locators (URL). <http://search.ietf.org/rfc/rfc1738.txt?number=1738>

Caplan, Priscilla and Arms, William Y. 1999. Reference Linking for Journal Articles. *D-Lib Magazine*. 5(7/8). <http://www.dlib.org/dlib/july99/caplan/07caplan.html>

NISO. 2000. Report on the Meeting held July 24, 2000 at CNRI, Reston, Va. NISO/DLF/CrossRef Workshop on Localization in Reference Linking. <http://www.niso.org/CNRI-mtg.html

Okerson, Ann and O'Donnell, James. 1995. Scholarly Journals at the Crossroads; A Subversive Proposal for Electronic Publishing. Association of Research Libraries; Washinton DC. <http://www.arl.org/scomm/subversive/toc.html>.

Paskin, Norman and Lannom, Larry. 2000. From One to Many. <http://dx.doi.org/10.1000/190>

Van de Sompel, Herbert and Hochstenbach, Patrick. 1999a . Reference Linking in a Hybrid Library Environment. Part 1: Frameworks for Linking. *D-Lib Magazine*. 5(4). <http://www.dlib.org/dlib/april99/van_de_sompel/04van_de_sompel-pt1.html>

Van de Sompel, Herbert and Hochstenbach, Patrick. 1999b. Reference Linking in a Hybrid Library Environment. Part 2:SFX, a Generic Linking Solution. *D-Lib Magazine*. 5(4). <http://www.dlib.org/dlib/april99/van_de_sompel/04van_de_sompel-pt2.html>

Van de Sompel, Herbert and Hochstenbach, Patrick. 1999c. Reference Linking in a Hybrid Library Environment. Part 3: Generalizing the SFX solution in the "SFX@Ghent & SFX@LANL" experiment.

D-Lib Magazine. 5(10). <http://www.dlib.org/dlib/october99/van_de_sompel/10van_de_sompel.html>

Van de Sompel, Herbert. July 2000. CrossRef -- DOI -- OpenURL -- SFX demo using IDEAL metadatabase & enhanced DOI proxy. < http://www.sfxit.com/crossref/prototype1.html>

Van de Sompel, Herbert; Hochstenbach, Patrick and Beit-Arie, Oren. 2000. OpenURL syntax description. <http://www.sfxit.com/openurl/openurl.html>

Van de Sompel, Herbert and Hochstenbach, Patrick. 2000. Cookiepusher document. <http://www.sfxit.com/openurl/cookiepusher.html>

Van de Sompel, Herbert; Krichel, Thomas, Nelson, Michael L. and others. 2000. The UPS Prototype: An Experimental End-User Service across E-Print Archives. *D-Lib Magazine*. 6(2). <http://www.dlib.org/dlib/february00/vandesompel-ups/02vandesompel-ups.html>

Acknowledgements

The authors wish to thank the following parties and individuals for their active involvement in the initial DOI/CrossRef/OpenURL demonstration; the ongoing prototype implementation; and/or their contribution to this paper: Patrick Hochstenbach (Ghent University), Tony Hammond (Academic Press), Larry Lannom (CNRI), Sean Reilly (CNRI), Ed Pentz (CrossRef), Mark Kosinski (CrossRef), Rick Luce (Los Alamos National Laboratory), Miriam Blake (Los Alamos National Laboratory), Nettie Lagace (Ex Libris), Jenny Walker (Ex Libris), Dana Sharvit (Ex Libris), Michael L. Nelson (University of North Carolina), Sandra Payette (Cornell University), and Naomi Dushay (Cornell University).

The current prototype involves the following parties: CrossRef, the International DOI Foundation, CNRI, the Digital Library Federation, Ex Libris, the Los Alamos National Laboratory, OhioLink and the University of Illinois at Urbana Champaign.

(At the request of Oren Beit-Arie, on 16 May 2001 a correction was made to the link in the fourth list item in Table 2. The line reads, 'Make sure to click here to turn the cookie off again. If you don't, all future request for DOI resolution will lead into the Ex Libris demo SFX server.' and the correct URL is http://www.doi.org/cgi-bin/pullcookie.cgi?Redirect=http://www.dlib.org/dlib/march01/vandesompel/03vandesompel.html#tab2.)

Top | Contents
Search | Author Index | Title Index | Back Issues
Previous Article | Next Article
Home | E-mail the Editor

DOI: 10.1045/march2001-vandesompel

Originnally published in D-Lib Magazine, Volume 7, Number 3, March 2001 (http://www.dlib.org/dlib/march01/vandesompel/03vandesompel.html).

