

Why OpenURL?

Ann Apps and Ross MacIntyre
MIMAS, Manchester Computing, The University of Manchester, UK
{ann.apps, ross.macintyre}@manchester.ac.uk

Abstract

The improvement of access to scholarly literature caused by electronic journal publishing quickly led to the wish for seamless linking to referenced articles. This article looks at the evolution of linking technologies with a particular focus on OpenURL, now a NISO standard. The implications for stakeholders in the supply chain are explored, including publishers, intermediaries, libraries and readers. The benefits, expectations and business drivers are examined. The article also highlights some novel, existing and potential future, uses, including increased user-empowerment and possibilities beyond referencing traditional bibliographic material.

A Brief History of Linking

The advent of electronic publication of academic literature very rapidly produced an appreciation, and consequent expectation, among researchers of the availability of articles at their desktop, rather than the previous scenario of visiting the library to read a print journal issue. But this improvement of access to scholarly literature soon initiated further schemes and desires for more enhancement. Researchers would benefit even more if they could instantly move to articles referenced by the article they were reading, or if there were seamless access to the full text of an article from a discovery service.

This situation led to the first implementations of linking to articles developed by resource providers in a **static** way. Links from references to cited articles were 'hard coded', initially within a corpus of material such as an electronic publishing system, for example within a particular publisher's content, or within an aggregated electronic journals service [1]. Subsequently linking services were provided between content owned by different resource providers following particular agreements between them, in some cases needing an additional subscription agreement with a library. The implementation of these links generally used a **proprietary** syntax. Thus a discovery service had to implement a different set of source links for each publisher in order to link from a discovered article's citation details to the actual article.

Subsequently several linking syntaxes were devised that utilised standard methods of identification or attributes (metadata) of the discovered object, making the required URL **predictable**. Thus a link to a journal may be made using its ISSN, and to an article additionally using its volume, issue and page details, a solution that is used for example within a SIC1 code. Links into a content owner's electronic publications site would be via a web link that included this article metadata within the address, the URL being effectively **parameterised** or **predicated**. This is typically processed via a CGI program, which extracts the metadata in a URL querystring (i.e., following a question mark '?' within an HTTP address [2]) thus allowing access to multiple articles via the same gateway. However, these links still used proprietary formats that differed between resource providers.

The next development was of intermediary systems to determine where an article may be found. An example of this development is Digital Object Identifier (DOI) [3]. A publisher registers a unique and **persistent** identifier, a DOI, for a published article along with its location. A discovery service that knows the DOI of an article is then able to link an end user to the full text of the article. The DOI system has since been developed to enable registration and subsequent retrieval of metadata about an identified resource. CrossRef [4] is a cross-publisher linking service based on DOI. Thus DOI introduced both a syntax for a persistent identifier and a means of indirection to the text of an identified article.

But although these developments provide technical linking from a citation to an article, they do not consider the context of the user, in particular whether they have permission to read a linked article. Articles are often available from multiple services, each with different access policies and prices. An organisation such as a university library typically establishes agreements

by subscription payments with particular electronic journal or aggregation services. A user at that institution will be able to read an article from a subscribed service apparently freely. This user will not be impressed with an invalid link, and will not be happy with a request for payment if they in fact have a subscription to the article elsewhere. Providing a link to a copy of a work to which a user has a valid subscription, or to an open access version, is known as the 'appropriate copy' problem [5].

The original version of OpenURL, now designated OpenURL version 0.1, provided both a common linking syntax and a solution to the appropriate copy problem. The OpenURL concept was developed as part of a research project (called SFX, "special effects") by Herbert Van de Sompel and Patrick Hochstenbach at Ghent University in Belgium (1998-2000) [6]. It was then acquired by Ex Libris who currently sells the SFX OpenURL resolver [7], the initial product in the OpenURL resolver market, which now includes products from several vendors [8]. The OpenURL version 0.1 specification [9] has always been public and open, its rapid take-up within the scholarly information community making it effectively a 'de facto' standard.

The OpenURL Framework for Context-Sensitive Services was endorsed as an ANSI/NISO standard (Z39.88-2004) in April 2005 [10]. This standard has broadened the potential scope of OpenURL implementation beyond the scholarly information community [11] by allowing the registration of new formats and profiles for new domains, as well as providing an XML format with its additional potential. It has also included several entities that were previously being used within proprietary, private data sections in the original OpenURL specification. ANSI/NISO Z39.88-2004 enables details of a link's context, such as the user and the source of the link, to be described in a systematic way, opening up new possibilities for services appropriate to the user. The OpenURL Framework has separated the actionable link that was historically called the OpenURL into its payload and the means of transporting it across the network. The payload, which details the reference and its context, is known as the ContextObject. This separation of interests enables the use of the ContextObject within other applications beyond the immediate delivery of object information through a simple URL templating structure. The OpenURL Framework and the ContextObject are described in more detail elsewhere, including a NISO article [12] and the OpenURL Framework Implementation Guidelines [13].

How OpenURL Works

As well as providing a standard, and thus universally accessible, syntax, OpenURL linking solves the 'appropriate copy' problem because it is facilitated by OpenURL resolver software deployed by an organisation such as an academic institution's library. A source application, such as an electronic journals service that includes links from references in article bibliographies, or an abstracting and indexing service that provides links potentially to the full text of the article, no longer needs to send the request to a particular target journals site, but can instead redirect it to an OpenURL resolver at the user's institution.

Typically an OpenURL resolver includes a 'knowledge base' that captures an organisation's subscriptions to electronic journals and other resources, along with details of how to encode links to a large number of target electronic resources, both traditionally published and open access. Typically the library will maintain subscription details within the knowledge base and select appropriate target services to be shown to their users, whereas the vendor will maintain target-linking details.

Thus when a resolver receives an OpenURL from a reader who has clicked on a reference link, it can provide that reader with a link to a referenced work if the reader has a right of access. Generally, rather than providing a straight redirect through to the full text of an electronic article, a resolver will provide to a user a set of links to relevant services, including the full text of the article where appropriate. This menu will contain other links to related items of possible interest, such as an abstract of the article, a search for other works by the same author, or the library catalogue. The content and layout of this menu is decided by the library, an example being shown in Figure 1.

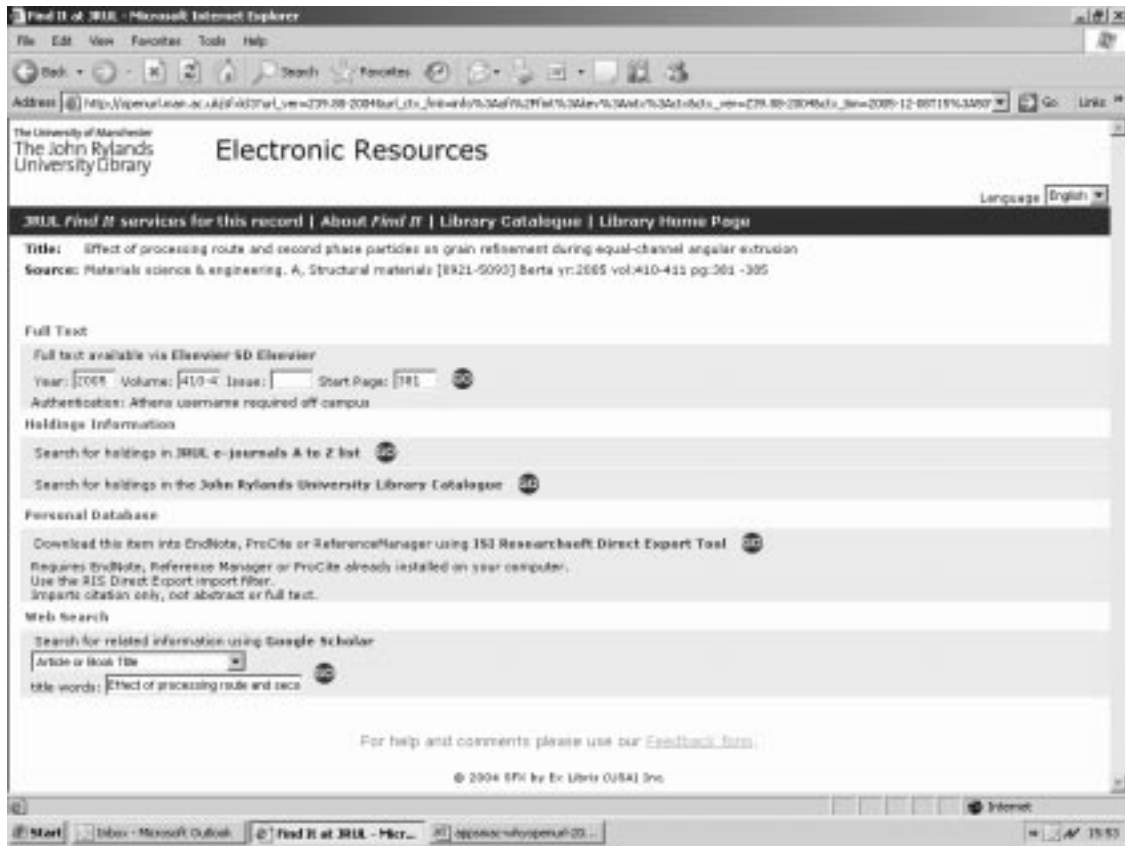


Figure 1. The SFX 'FindIt' menu at The University of Manchester

Resolver Location

Clearly there is an additional problem to solve, namely how a source service knows the address of a user's OpenURL resolver to which to send a request. Some services maintain private registries of OpenURL resolvers, expecting organisations to provide them with details. This may include an image and preferred text to allow customisation of the link shown to the end user, thus enabling a common appearance of OpenURL links across a library's resources. This strategy works well for services that authenticate access and thus know a user's location, and is used by services in the UK such as The British Library's Zetoc [14] and Thomson Scientific's Web of Knowledge [15].

For freely available services it would be inappropriate to ascertain a user's location and thus a different approach is needed. The most commonly used method is to record the resolver address in a cookie in a user's web browser and ask the user for their location details. Other methods for individual user personalisation based on extensions to the Firefox web browser [16] are also starting to emerge. An alternative method is for a source service to send all OpenURL requests to a public OpenURL resolver registry, which will then route requests to a user's resolver determined by their IP address. Within UK Higher and Further Education the OpenURL Router [17] provides this service for registered institutions. OCLC has released a prototype registry, currently publicly available, with mainly US coverage [18]. It is probable that services with private resolver registries will harvest resolver registrations from these public registries to augment their own records. Determining the appropriate resolver for a user is an area where there is still work to be done to encourage wider uptake of the OpenURL standard.

Default Resolvers

A further problem for a source service is what provision to make for users who do not have an OpenURL resolver available. Public resolver registries show a default web page to these users to provide some assistance in finding articles, as do services that maintain private registries. For

users in UK academia who are seeking journal articles and who do not have an OpenURL resolver, Zetoc provides a link to LinkFinderPlus [19] as a default resolver. LinkFinderPlus knows where many science-based articles are available but has no knowledge of a user's subscriptions so is unable to provide a link to an 'appropriate copy'. On the other hand, many institutions have access to major resources. Zetoc also provides links to the publicly available, science-based search engine Scirus [20], and to COPAC, the union catalogue of the UK national and major research libraries [21].

OpenURL is not a competitor with DOI-based linking initiatives such as CrossRef [22], but is indeed a complementary technology that enables localised linking to a resource so identified. If a resolver receives a DOI for an article, it can make use of it to retrieve bibliographic details about the reference before continuing its usual link provision. A CrossRef Z39.88-2004-compliant resolver [23], a gateway to some 19 million items, is now publicly available, though intended for personal rather than commercial or harvesting use. It has the potential to be used as a default resolver where request of an article from its publisher is desired, and it has an option to supply multiple hits. It functions as a reverse DOI lookup service enabling metadata retrieval, according to the CrossRef XML schema, about a DOI-identified article, possibly for subsequent inclusion in an OpenURL ContextObject.

Target Services

The primary, initial purpose of the OpenURL Framework is to support context-sensitive linking from source citations to relevant services including the full text. Details of how to link into target services are captured within a resolver's knowledge base. The only requirement for a service to be a target linking service is for it to publish its linking syntax and facilitate deep linking to the article level. However, interoperability will be promoted, and target details in the knowledge base simplified if linking into a target by OpenURL syntax is enabled. A service that supports OpenURL syntax for incoming target links is known as a 'Link-To' resolver.

OpenURL Stakeholders

OpenURL technology has very quickly become an established part of electronic publishing services provided by both libraries and publishers to academic readers, whether researchers, learners or teachers. Thus there are several stakeholders within the scholarly information community where OpenURL technology is in use. Publishers may potentially have a dual role, both as an OpenURL 'source', or starting point, *sending* links, and as an OpenURL 'target', or finishing point (otherwise known as a network endpoint), *receiving* links into their content.

Readers

There are clear benefits for end users from the provision of seamless delivery of the full text of a discovered article directly to them. Evaluation studies undertaken on the use of Zetoc [24] show that OpenURL linking is well used and appreciated, particularly by readers in institutions that have a resolver. Responses to questionnaires were very positive, even excited. Zetoc's logging of OpenURL linking shows considerable use.

The user experience is enhanced, particularly for novices, if an institution is able to provide a consistent interface into their OpenURL resolver's menu from all sources within the maze of available systems. Ideally users should be presented with appropriate, verified and unified links. Thus it is preferable if OpenURL source services allow customisation of the graphic and text of links by an institution.

The rapid uptake of OpenURL means that users now expect the provision of easy connection between resources. The multiple steps required to navigate to articles through the wide variety of electronic journal sites discovered via a catalogue could deter many potential readers, a hypothesis articulated as Moer's Law: "An information retrieval system will tend not to be used whenever it is more painful and troublesome for a customer to have information than for him to not have it" [25]. Eason concludes [26] in his evaluation of user behaviour, based on the use of OpenURL technology via Zetoc, within the array of services that are the users' 'ready to hand' working practices, that "most users will only try new services as minor variations on normal practice that are easy to explore". This is consistent with Heidegger's concept [27] that people undertake their everyday tasks by using tools and techniques so familiar to them that they do not have to think, and Zipf's law [28] regarding the 'principle of least effort'.

Libraries

Libraries are typically the organisations within academic institutions or companies that provide OpenURL resolution services to their members. They are responsible for purchasing a resolution system, incorporating into the resolver details of their subscriptions to electronic resources, maintaining the software and knowledge base with regular vendor updates, and designing the readers' view of the interface. This involves a considerable amount of work, as well as a significant cost investment. However, as indicated by another evaluation study [29], librarians are generally very enthusiastic about the technology because of the added value it provides to their readers.

Libraries that install an OpenURL resolver hope that the use of the resources to which they provide access will increase, thus justifying their investment in their electronic holdings. This was clear from background discussion as part of the aforementioned evaluation. Reducing user frustration by showing consistent links from all resources means that they will make the best use of library resources rather than turning to Google. Librarians appreciate the ability to customise their resolver's user interface. In particular they prefer to show the institution's branding, to reinforce to their readers that the seamless retrieval of full text articles is in fact available because the library has purchased these resources.

A secondary benefit to libraries of acquiring an OpenURL resolver is its availability for backroom processes. The resolver's knowledge base can be harnessed to assist in central resource management, dealing with recording holdings and acquisitions details, and collection development.

Unfortunately the cost of OpenURL resolvers is prohibitive to some, smaller institutions and those in developing countries, which may be creating some disenfranchisement of the 'have nots' within the research and learning arenas. An OpenURL resolver has become a 'must have' for major research universities that partake in the international research community. Cervone recently stated [30]: "OpenURL and federated searching are important new services in our field. They represent extensions of the library that further enable people to find their own information, which will be critical to the library of the future". As mentioned above there are a few free resolvers, although their use may be confusing to readers who do not understand why they cannot access resources that are apparently available. Such frustrated users will probably demand OpenURL resolver services from their libraries. One initiative that partly addresses this problem is ScholarSFX (from Google Scholar and SFX) [31]. This free service, to libraries that supply their holdings information to Google Scholar, enables OpenURL links to their readers within Google Scholar search results.

Funding Bodies

Another set of stakeholders in the resource supply chain includes the funding bodies that provide access to resource collections for their constituent communities, generally within their defined information environment architecture. The Joint Information Systems Committee (JISC) [32] negotiates access to resources for UK Higher and Further Education, and has a firm requirement that funded resources be consistent with the JISC Information Environment's technical architecture. Steps 3 and 4 of the '5 step guide to becoming a content provider in the JISC Information Environment' [33] require that resource platforms become an OpenURL source and an OpenURL target.

Publishers

Publishers are also stakeholders, being both potential sources and targets of OpenURLs. They may be suppliers of primary content, publishing journal articles or books, secondary publishers of resources such as abstract and indexing or citation based discovery services, or intermediaries such as content aggregators. Although it may seem that there is no advantage to a publisher to implement OpenURL links, because the links might take readers away from the publisher site, there are in fact benefits to publishers from implementing OpenURL both within the industry as a whole, and for themselves 'in house', as shown in the following paragraphs.

Publishers as OpenURL Sources:

Probably the main reason why a publisher should become an OpenURL source is a market requirement that they meet customer expectations, OpenURL technology now being considered

critical functionality. Increasingly libraries and funding bodies require OpenURL source links from resources to which they subscribe. Libraries worldwide regard OpenURL as a basic, standard service for accessing electronic journal content. As one librarian said: "in assessing new content for purchase, we always look at the linking possibilities". Users also expect OpenURL from their library services, although may not be aware of the technology, quoting from another librarian: "it's just too complicated without the OpenURL to navigate to articles". Thus publishers need to implement OpenURL, as a selling point, to maintain a competitive edge.

Possibly a cultural change is needed, in that OpenURL support has to be seen as a partnership, with a need to work with all stakeholders including one's competitors, a change that CrossRef participants have already acknowledged. Publishers will reap more benefit from OpenURL linking if they keep their lists of available content up to date with the major resolver vendors. This will benefit librarians, and ultimately readers, because they will be able to make full use of purchased content by reducing the maintenance headache of resource subscription date thresholds.

There are also benefits to publishers from providing OpenURL source links, taking away from them the responsibility to provide what users need to librarians and resolvers. User requirements were becoming increasingly onerous before the introduction of OpenURL technology. Standard format, interoperable links are easier to maintain than the earlier bilateral agreements between different organisations. Using standards for interoperability streamlines administration of the linking infrastructure for both publishers and librarians, thus decreasing support resource needs and hence lowering costs, increasing the quality of services, and supporting further innovation.

High on the wish list from librarians and readers is the OpenURL-enabling of references from journal articles and other bibliographic resources, a functionality that a few primary publishers are now beginning to provide. This brings additional value to the content that libraries purchase, with a consequent market advantage. Major publishers who are members of CrossRef probably already support DOI-based links that can be routed to an institutional link server if the institution has chosen to register that intent with CrossRef. But additionally offering OpenURL source links enables the provision of links from citations that do not have DOIs, and introduces future possibilities of adding value beyond that offered by the centralised CrossRef service.

The many secondary publishing resources, such as abstracting and indexing databases that are already enabled as OpenURL sources are popular. They have a clear selling point, because students and researchers increasingly are looking for journal articles in full text rather than just citations. OpenURL enabling a discovery service, such as Zetoc [34], adds significant value because it appears to become a source of full text articles to which the user's institution has a subscription, with no need for knowledge by the user of the relevant full text service.

Since the formal endorsement of the Z39.88-2004 standard, it is preferred if source links generated by publishers are hybrid, containing keys from both the new standard version 1.0 OpenURL, as well as the original OpenURL version 0.1, because it is not possible for a source service to know if a target resolver supports the standard version yet. The major resolver vendors have now upgraded their software to support Z39.88-2004, but each individual resolver has to be upgraded by its owning organisation. Although upgrading source links from the apparently simpler OpenURL version 0.1 is perceived as being a barrier, it is in fact quite straightforward as described in an appendix of the OpenURL Framework Implementation Guidelines [13].

Publishers as OpenURL Targets:

A publisher may become an OpenURL target simply by providing a transparent 'link to' syntax and advertising the details through a service such as Openly's JournalSeek [35]. However becoming an OpenURL 'Link-To' resolver increases interoperability generally, because it removes the need for different proprietary linking syntaxes for each publisher. The expected increased use of content, particularly for publishers of full text material, is clearly a marketing advantage. It also simplifies linking between publishers because it obviates custom linking methods. Readers who visit an electronic publishing site via an OpenURL link are taken into a branded environment with consequent advertising for the publisher of their e-journal services.

Some publishers already implement inbound linking to an article using a DOI. But a DOI may not be known, in particular when references have been discovered in printed documents. DOI lookup through the CrossRef system would involve this additional dynamic overhead to achieve

the desired link. Although using OpenURL syntax can result in much longer URLs than a simpler native syntax or DOI link, the benefits of providing the option of an interoperable syntax seem clear. Supporting both the version 1.0 syntax as well as the original version 0.1 syntax will promote interoperability of linking in the current period of transition to the standard format. Of course, if a DOI is known, it can be included in the OpenURL. The unique identifier provided by DOI would ensure accuracy of resolution, not being reliant on metadata quality, with the addition of context-sensitive information to provide an appropriate linking experience.

Proprietary (i.e., non-DOI) linking into a publisher's site has no guarantee of persistence, as any changes by the publisher can cause links to break until the target details in OpenURL resolver knowledge bases have been updated. User and librarian frustration with a publisher's broken links may cause a lack of trust in that publisher with a possible consequent loss of future business. Implementing OpenURL syntax for inbound links would allow a publisher to reorganise its electronic publishing site with no impact on access to its content.

Current Status: What is Missing?

Many major discovery resources now provide OpenURL source links, as do some journal publishers [36]. OpenURL resolvers are able to link to a large number of target resources [37]. However, in many cases linking to targets is via proprietary syntax captured in a resolver's knowledge base.

But there are still some major resources that do not provide OpenURL source links or enable target linking into their content. This situation causes concern to librarians who view such resources as significant gaps in the 'joined up' linking experience that they would like to provide to their readers using the OpenURL technology in which they have invested. In September 2005, a priority list was identified by European librarians who are members of SMUG (SFX/MetaLib Users Group) [38(p10)].

When wishing to read a journal article, obviously a user would prefer to link directly to that article. However, some electronic journal applications do not permit deep linking and are able only to link to issue or volume level, or in some cases to journal level [37]. This may be because of technical difficulties in supporting linking to article level, but in some cases it is a publisher's policy, fearing loss of branding by deep linking. But in an OpenURL-linked world, such policies may be counter-productive.

Novel Uses

Additional OpenURL Sources

Since the development of the OpenURL Framework standard, further uses are starting to appear [39], mostly as yet within the originating scholarly information environment. Google Scholar [40] now provides OpenURL source links for users from institutions that have registered their holdings and resolvers with Google. These links appear beside search results that are journal articles identified by a DOI or described with a PubMed record. The free service, social bookmarking tool, Connotea [41], includes hybrid, dual-version OpenURL source links [42].

Standard Bibliographic Citation Description

The standardisation of the OpenURL Framework means that journal and book metadata formats of the OpenURL ContextObject are now endorsed as a standard way to capture bibliographic citation details. The Dublin Core Metadata Initiative (DCMI) recommends using a ContextObject for capturing bibliographic citation information within a Dublin Core metadata description [43]. The NISO Metasearch Initiative's [44] Task Group 3 'Search and Retrieval Recommendations' bases its recommendations for Required Citation Level Data Elements on this DCMI recommendation and the Z39.88-2004 metadata formats.

Latent OpenURLs

A recently developed concept is that of 'embedded' or 'latent' OpenURLs. If a ContextObject were to be embedded in a web page, invisible to a human reader who might regard it as too cryptic, other applications such as web browser extensions could 'magically' provide extra functionality such as 'article autodiscovery' [45]. This has led to the development of the COinS specification [46], which embeds a ContextObject within an HTML 'span' element. COinS has a low implementation overhead, but provides powerful opportunities for additional services. It enables

an application to generate an active OpenURL simply by prefixing a resolver address. An extension to the Firefox web browser, such as Openly's OpenURL Referrer [47], allows a user to personalise their OpenURL linking via their chosen OpenURL resolver. Figure 2 shows the process of personalisation by selection of The University of Manchester's 'FindIt@JRUL' resolver. Further developments are appearing such as 'WAG the DOG' [48], which generates COinS using screen scraping, refined by holdings or ISBN lookups, in various applications, and provides an OpenURL-enabled 'more like this' service based on journal categories. Several services are COinS-enabled including: CiteULike [49], another free service that helps academics to share, store and organise the academic papers they are reading, and Citebase [50], a search and citation analysis tool for free, online research literature.

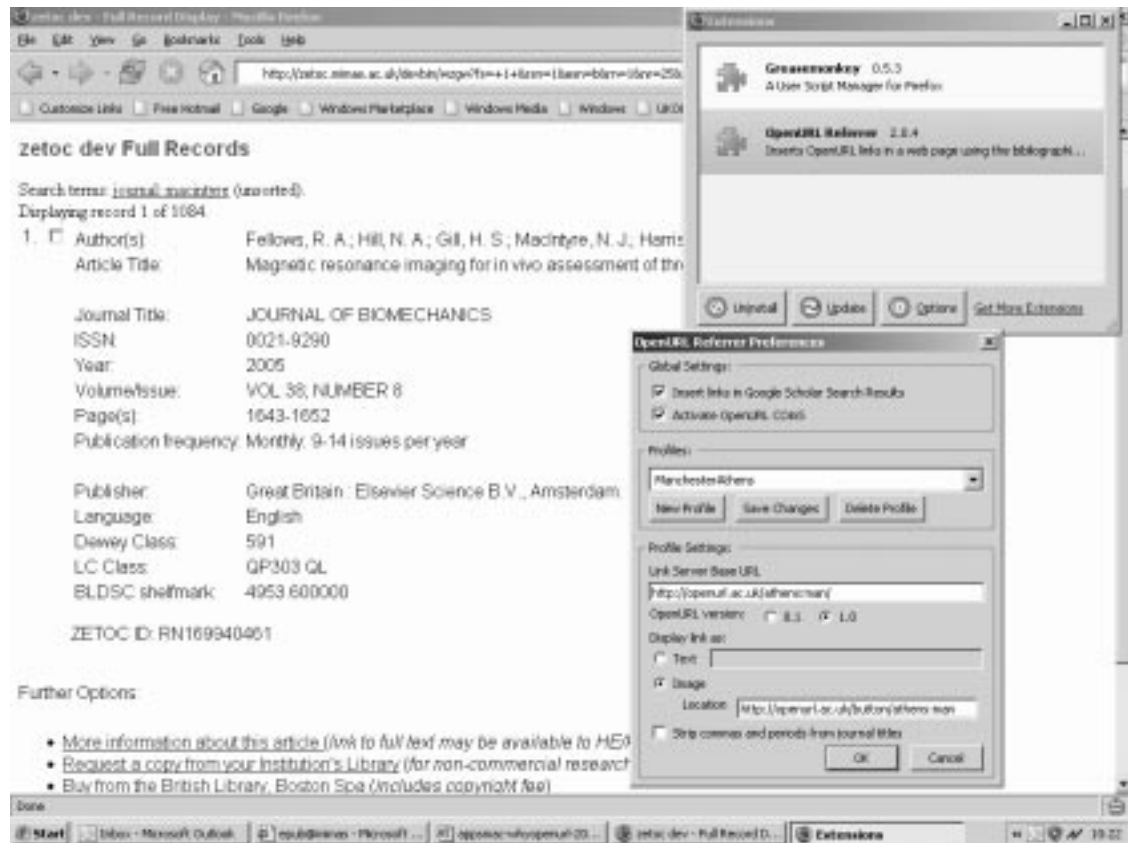


Figure 2. Selecting a Resolver with OpenURL Referrer

Zetoc has recently implemented COinS on the full citation record page, and also on the 'default' page that is seen by users whose institution does not have an OpenURL resolver. If a reader installs the OpenURL Referrer, profiled with their own choice of OpenURL resolver, as a Firefox extension, they will see the relevant button or text at the side of the OpenURL link in Zetoc. Clicking on this button, rather than their institution's (if one is shown) will send the OpenURL link to their chosen resolver. This would allow a user visiting another institution access to their 'home' resolver, or it would allow a user without an assigned resolver to activate OpenURL linking. Figure 3 shows a Zetoc page, as would be seen at an institution with no resolver, with a 'FindIt@JRUL' button, which will send the hidden ContextObject to The University of Manchester's resolver. Clearly, at present, this personalisation would be applicable to only a small number of users who understand OpenURL linking and to those who have a personal Firefox web browser, but one may anticipate future applications and installations that extend this functionality to naïve readers.



Figure 3. A Clickable COinS button and the Hidden ContextObject

XML Uses

The traditional OpenURL carried its metadata description of a referenced resource as a set of 'key and value' pairs on a URL querystring. While this format is still available to describe a ContextObject, the standard also includes an XML serialisation, opening up the possibility of using OpenURL ContextObjects for 'server to server' communication. A Web Services SOAP interface into Zetoc [51] describes the journal article bibliographic fields of the response record using elements from the XML ContextObject journal metadata format. An RSS 1.0 module has also been proposed and allows for the embedding of a ContextObject within an RSS feed, thus enabling the passing of contextual information to a downstream application that provides appropriate services to a consumer of the feed [52].

Describing Non-Bibliographic Resources

Version 0.1 of OpenURL defined a domain-restricted set of metadata covering only books, journals and conference proceedings. Whereas the OpenURL Framework defines metadata formats in an extensible registry [53], potentially extending use to a diverse range of domains. Los Alamos National Research Laboratory has built an OpenURL Resolver interface to a repository of complex digital objects to service dissemination requests [54]. The JISC Information Environment Service Registry (IESR) [55] OpenURL Link-To Resolver supplies a collection, service or agent XML description corresponding to a particular identifier, with the additional option to request the entity type using the OpenURL's Dublin Core metadata format. This resolver also implements resolution of IESR URIs (Uniform Resource Identifiers).

OpenURL: The Future

OpenURL developed from a wish by readers of scholarly literature for easier 'joined up' location and delivery of discovered resources of interest. Linking technology developed quite rapidly from the original *static* and *proprietary* solutions, through those which were *predictable*, to *parameterised*, to *persistent* but *publisher-controlled*, to the *prototype* OpenURL, to the eventual

standard, interoperable, flexible OpenURL Framework for Context-Sensitive Services. Of course, as with all developments, previous stages on the way have been subsumed into OpenURL linking. In particular, incorporating a DOI within an OpenURL, where available, will take advantage of the exact resolution capabilities of DOI. Developments still continue with 'latent' OpenURLs that enable capable users to choose their own resolver, and possibilities of adding ContextObjects to screens 'on the fly'. Chudnov recently described [56] this as a change from resolver rules and services that are pre-coordinated by libraries, to allowing the dynamic post-coordination of services by a user. Miller in his 'Principles of Web 2.0' [57] suggests that applications should work for the user "rather than forcing us to conform to the paths laid out for us", and that Web 2.0 "opens up the Long Tail" servicing the interests of individuals. His observations are based on O'Reilly's Web 2.0 Meme Map [58] that includes: "rich user experience"; "trust your users"; and "emergent: user behaviour not predetermined."

OpenURL has become a ready success because it satisfied several key factors: there was a clear need from end users for this kind of functionality; there are benefits to all the major stakeholders as this article describes; there is a low barrier to implementation, especially of the initial version; and, maybe most importantly, there has been a willingness to compromise by the stakeholders.

OpenURL is generally perceived as being a link to a full text article. But even the original OpenURL was intended to provide access to a range of services relevant to a referenced work, and it now has many more possibilities for use. Future opportunities based on the OpenURL Framework may include: applications outside the scholarly information domain; services appropriate to the context, such as those based on a requester's role, or RSS feeds for the referring journal; or collaborations and communications between resolvers sharing knowledge base content or checking use rights.

Future uses of OpenURL, whether prototype, suggested or not yet imagined, indicate that 'Why OpenURL?' does not have a final answer. Current developments are just steps on the way to achieving the vision of the OpenURL Framework, from its originator Herbert Van de Sompel, of expediting "a dynamic, personalised link structure on top of the existing static Web link structure" [59].

Acknowledgements

The authors wish to thank those who responded to the question 'Why use OpenURL?' whose comments have been incorporated into this article: Jenny Curtis, John Rylands Library, The University of Manchester; Chuck Hamaker, Atkins Library, University of North Carolina Charlotte, USA; Tony Hammond, Nature Publishing Group; Nick Lewis, University of East Anglia Library; and Mark Needleman, Sirsi Corporation. They also wish to acknowledge presenters at the NISO OpenURL Workshop held in September 2005 [60], some of whose ideas are incorporated in this article.

References

- [1] Apps, A. and MacIntyre, R. (1999). The SuperJournal Project: Data Handling Using SGML. In: *Proceedings of the Third ICCC/IFIP Conference on Electronic Publishing (Electronic Publishing '99)*, Ronneby, Sweden, 10-12 May 1999. (ICCC Press) pp 215-224, ISBN 1-891365-04-5.
<http://epub.mimas.ac.uk/papers/ep99apps.html>
- [2] Berners-Lee, T. (1994). Uniform Resource Locators (URL): A Syntax for the Expression of Access Information of Objects on the Web. <http://www.w3.org/Addressing/URL/url-spec.txt>
- [3] The Digital Object Identifier System. <http://www.doi.org/>
- [4] CrossRef. <http://www.crossref.org/>
- [5] Beit-Arie, O., Blake, M., Caplan, P., Flecker, D., Ingoldsby, T., Lannom, L.W., Mischo, W.H., Pentz, E., Rogers, S. and Van de Sompel, H. (2001). Linking to the Appropriate Copy. *D-Lib Magazine*, 7(9).
<http://www.dlib.org/dlib/september01/caplan/09caplan.html>
- [6] Van de Sompel, H. and Beit-Arie, O. (2001). Open Linking in the Scholarly Information Environment Using the OpenURL Framework. *D-Lib Magazine*, 7(3).
<http://www.dlib.org/dlib/march01/vandesompel/03vandesompel.html>
- [7] Ex Libris. SFX Context Sensitive Linking. <http://www.exlibrisgroup.com/sfx.htm>
- [8] OpenURL Products and Vendors. <http://www.loc.gov/catdir/lcpaig/openurl.html>
- [9] Van de Sompel, H., Hochstenbach, P. and Beit-Arie, O. (2000). OpenURL Syntax Description, Draft Version 0.1. <http://www.openurl.info/registry/docs/pdf/openurl-01.pdf>

- [10] NISO. (2005). Z39.88-2004, The OpenURL Framework for Context-Sensitive Services. http://www.niso.org/standards/standard_detail.cfm?std_id=783
- [11] Van de Sompel, H. and Beit-Arie, O. (2001). Generalizing the OpenURL Framework beyond References to Scholarly Works: the Bison-Futé model. *D-Lib Magazine*, 7(7/8). <http://www.dlib.org/dlib/july01/vandesompel/07vandesompel.html>
- [12] Hodgson, C. (2005). Understanding the OpenURL Framework. *NISO Information Standards Quarterly*, 17(3), 1-4.
- [13] NISO AX and Apps, A. (2004). Z39.88-2004: The Key/Encoded-Value (KEV) Format Implementation Guidelines. http://www.openurl.info/registry/docs/implementation_guidelines/
- [14] Zetoc, Electronic Table of Contents from the British Library. <http://zetoc.mimas.ac.uk/>
- [15] Web of Knowledge. <http://wok.mimas.ac.uk/>
- [16] Firefox. <http://www.mozilla.org/products/firefox/>
- [17] The OpenURL Router. <http://openurl.ac.uk/doc/>
- [18] OCLC OpenURL Resolver Registry. <http://www.oclc.org/productworks/urlresolver.htm>
- [19] Endeavor Information Systems, LinkFinderPlus. <http://www.endinfosys.com/prods/linkfinderplus.htm>
- [20] Scirus. <http://www.scirus.com/>
- [21] The COPAC research library online union catalogue service. <http://copac.ac.uk/>
- [22] OpenURL & CrossRef. <http://www.crossref.org/02publishers/16openurl.html>
- [23] CrossRef's OpenURL Resolver. http://www.crossref.org/02publishers/openurl_info.html
- [24] Eason, K., Harker, S., Apps, A. and MacIntyre, R. (2004). Towards an Integrated Digital Library: Exploration of User Responses to a 'Joined-Up' Service. *Lecture Notes in Computer Science*, 3232, 452-463. <http://epub.mimas.ac.uk/papers/eham-ecdl2004.html>
- [25] Mooer, Calvin. (1960). Mooer's Law. *American Documentation*, 11(3), p.ii
- [26] Eason, K., MacIntyre, R. and Apps, A. (2005). A 'Joined-Up' Electronic Journal Service: User Attitudes and Behaviour. In: *Libraries Without Walls 6: Evaluating the Distributed Delivery of Library Services*. Facet Publishing, London (accepted for publication). <http://epub.mimas.ac.uk/papers/lww6/easonetal-lww6.html>
- [27] Heidegger, M. (1977) The question concerning technology. Harper & Row, New York
- [28] Zipf, G.K. (1949). Human Behaviour and the Principle of Least-Effort. Addison-Wesley, Cambridge MA
- [29] Eason, K. and Harker, S. (2003). The Impact of OpenURLs on End Users. <http://metadata.mimas.ac.uk/ITAM/evaluation.html>
- [30] Cervone, F. (2005). What We've Learned from Doing Usability Testing on OpenURL Resolvers and Federated Search Engines. *Computers In Libraries*, 25(9), 10-14
- [31] ScholarSFX. http://www.exlibrisgroup.com/scholar_sfx.htm
- [32] JISC. <http://www.jisc.ac.uk>
- [33] Powell, A. (2002). 5 step guide to becoming a content provider in the JISC Information Environment. *Ariadne*, 33. <http://www.ariadne.ac.uk/issue33/info-environment/>
- [34] Apps, A. and MacIntyre, R. (2003). Using the OpenURL Framework to Locate Bibliographic Resources. In: *Proceedings of the 2003 Dublin Core Conference (DC2003 - Supporting Communities of Discourse and Practice - Metadata Research and Application)*, Seattle, Washington, USA, 28 September - 2 October 2003, pp 143-152, ISBN 0-9745303-0-1. <http://epub.mimas.ac.uk/papers/appsmacdc2003.html>
- [35] Openly JournalSeek Coverage for e-Journals. <http://www.openly.com/journalseek/publishers.html>
- [36] SFX Sources (OpenURL Enabled Resources). http://www.exlibrisgroup.com/sfx_sources.htm
- [37] SFX Targets. http://www.exlibrisgroup.com/sfx_targets.htm
- [38] SMUG (SFX / MetaLib Users Group) 4 EU. http://smug-4-eu.org/newsletter/smug4eu_issue2.pdf
- [39] Apps, A. and MacIntyre, R. (2005). Emerging Uses for the OpenURL Framework. In: *Proceedings of the Ninth ICC International Conference on Electronic Publishing (ELPUB2005 - From Author to Reader, Katholieke Universiteit Leuven, Belgium, 8-10 June 2005)* (Peeters Publishing Leuven), pp 283-289, ISBN 90-429-1645-1. <http://epub.mimas.ac.uk/papers/elpub2005/appsmac-elpub2005.html>
- [40] Google Scholar. <http://scholar.google.com/>
- [41] Connotea. <http://www.connotea.org/>
- [42] Lund, B., Hammond, T., Flack, M. and Hannay, T. (2005). Social Bookmarking Tools (II). *D-Lib Magazine*, 11(4). <http://www.dlib.org/dlib/april05/lund/04lund.html>
- [43] Apps, A. (2005). Guidelines for Encoding Bibliographic Citation Information in Dublin Core Metadata. <http://www.dublincore.org/documents/dc-citation-guidelines/>
- [44] NISO Metasearch Initiative. http://www.niso.org/committees/MS_initiative.html
- [45] Chudnov, D., Cameron, R., Frumkin, J., Singer, R. and Yee, R. (2005). Opening up OpenURLs with Autodiscovery. *Ariadne*, 43. <http://www.ariadne.ac.uk/issue43/chudnov/>
- [46] Hellman, E. (2005). OpenURL COinS: A Convention to Embed Bibliographic Metadata in HTML. <http://ocoins.info/>
- [47] OpenURL Referrer from Openly Informatics. <http://www.openly.com/openurlref/>
- [48] WAG the Dog. http://www.library.gatech.edu/research_help/WAG_faq.html
- [49] CiteULike. <http://www.citeulike.org/>

- [50] Citebase. <http://www.citebase.org/>
- [51] Apps, A. (2004). zetoc SOAP: a Web Services Interface for a Digital Library Resource. *Lecture Notes in Computer Science* **3232**, pp 198-208. <http://epub.mimas.ac.uk/papers/appsecdl2004.html>
- [52] Hammond, T., Hannay, T. and Lund, B. (2004). The Role of RSS in Science Publishing. *D-Lib Magazine*, **10**(12). <http://www.dlib.org/dlib/december04/hammond/12hammond.html>
- [53] The OpenURL Registry. <http://www.openurl.info/registry/>
- [54] Bekaert, J., Balakireva, L., Hochstenbach, P. and Van de Sompel, H. (2004). Using MPEG-21 DIP and NISO OpenURL for the Dynamic Dissemination of Complex Digital Objects in the Los Alamos National Laboratory Digital Library. *D-Lib Magazine*, **10**(2). <http://www.dlib.org/dlib/february04/bekaert/02bekaert.html>
- [55] Apps, A. (2005). A Middleware Registry for the Discovery of Collections and Services. In: *NCeSS2005: Proceedings of the First International Conference on e-Social Science, Manchester, UK, 22-24 June 2005*. <http://epub.mimas.ac.uk/papers/ncess2005/apps-ncess2005.html>
- [56] Chudnov, D. and Frumkin, J. (2005) Library Dialtone: Bootstrapping with Autodiscovery and Service Registries. In: *Access 2005, Edmonton, Canada, 17-19 October 2005*. <http://curtis.med.yale.edu/dchud/talks/20051019-access-dialtone/dialtone.pdf>
- [57] Miller, P. (2005). Web 2.0: Building the New Library. *Ariadne*, **45**. <http://www.ariadne.ac.uk/issue45/miller/>
- [58] O'Reilly, T. (2005). What is Web 2.0. <http://www.oreilly.com/go/web2>
- [59] Brunning, D. (2003). Interview with Herbert Van de Sompel, Creator of OpenURL/SFX. *The Charleston Advisor*, **4**(4). <http://charlestonco.com/features.cfm?id=124&type=np>
- [60] NISO OpenURL and Metasearch Workshop, September 19-21 2005, Washington DC, USA. http://www.niso.org/news/event_workshops/OpenURL-05-Agen-FINAL.html