

Innovative Uses of the OpenURL

With the formal standardization of the OpenURL in v. 1.0, many developers and librarians are beginning to use the OpenURL in unique and creative ways. The menu of services offered by link resolvers can be configured to link to a variety of extended services, and depending upon the product, can be designed by librarians to give users a multitude of link options beyond traditional full text in e-journals or in aggregators' databases. Extended-services linking initially began as links to the local library catalog or to Interlibrary Loan (ILL) systems. Now, other extended services opportunities are available.

Linking to Extended Services

In its most narrow sense, extended-services linking usually requires that the library utilize the intermediary screen, or the menu of options presented to the user between the source and the target. It is possible, however, to bypass the intermediary screen, and this is sometimes called "invoking the full text." In a practical sense, invoking the full text means that when the user clicks on the link to the resolver in the source, he or she is taken directly to the full text housed at the target. Whether or not a link-resolver product offers this sort of one-click functionality is a factor libraries should consider when making purchasing decisions. Google Scholar has utilized this sort of invoking in its implementation of the OpenURL, which is explained in greater detail in chapter VII.

Because the library needs to have a high degree of confidence the links to the desired information object will function correctly, the choice to utilize invoking is a difficult one. If one-click access is activated, and the link fails, the user may not immediately have access to help screens or other assistance such as chat reference.

Links to assistance on the intermediary screen can take any number of forms, including virtual-reference services, direct e-mail messages, or online-help forms. Although these forms of assistance are librarian driven, the intermediary screen can also be configured so the savvy end user could possibly help himself or herself.

For example, most link-resolver vendors allow the metadata being passed from the source to the target to be altered by the user, but this takes place via a form on the intermediary screen. If the metadata sent from the source is somehow corrupted or inaccurate, the user can check the original citation in the source and correct the volume, date, page number, etc., on the intermediary screen, thereby changing what information is sent in the OpenURL to the target by the link resolver. Or, the source may not have sent enough data to resolve directly to the article, so the savvy end user can again check the source's citation information and add enough information to resolve to the article level. Of course, this requires the user to be fairly information literate, which is not always the case; therefore, the librarian-driven help features may be the user's first line of defense in the event of incomplete or inaccurate metadata in the OpenURL.

Another feature that link-resolver products may offer is something called a "citation finder" or a "citation linker." The citation finder is usually a separate Web form that allows an end user to populate such fields as journal title, year, start page, volume, author name, DOI, etc., and then create an OpenURL from scratch. In essence, the citation finder feature takes the A-to-Z list further; it allows the user to perform known-item searches for information objects. If the library has populated its instance of the knowledgebase with its print holdings (rather than simply linking to the local library catalog from the intermediary screen), it also allows the user to perform a simultaneous

search of the library's electronic and print holdings.

Invoking and the citation finder are only two innovative uses of the OpenURL. Again, many libraries are linking to any number of extended services from the intermediary screen itself. Libraries have chosen to include links to different citation styles, such as the American Psychological Association, Chicago, or Modern Language Association. Furthermore, depending on the ability of the link-resolver product, libraries may add links that download the citation into reference managers, such as EndNote, ProCite, or RefWorks.

Another example of creative extended-services linking is the ability to link from the intermediary menu and perform citation searching for author's name in citation databases, such as Web of Science. Depending on the link-resolver product, it may be possible to configure it to drop the author name into a search in Web of Science (or other citation searching software) so users can see what else the author has written on a topic.

Because OpenURLs take a user's affiliations into consideration, libraries may also choose to educate users—particularly faculty in the academic environment—about how to use OpenURLs as persistent links in course-management software (i.e., WebCT or Blackboard) or in other Web-based instructional materials for students. Link-resolver products can offer another link from the intermediary screen that allows users simply to cut and paste the OpenURL for a particular citation into instructional materials. Of course, the OpenURL would have to have the correct metadata to resolve directly to the article, so the citation finder may be a better choice because it allows an end user to build an OpenURL. OpenURLs may not be the best choice for persistent linking, however. For example, Scott A. Warren has explored the use of DOIs for deep-linked e-reserves, as DOIs have inherent stability. Warren further explores other uses for the DOI in the academic environment, such as in bibliographies or in institutional repositories.¹

Finally, extended-services linking also encompasses links to freely available search engines and other Web-based research tools. Because of the increased extensibility of OpenURL v. 1.0, to what extended services a library links may only be limited by the imagination of the librarian.

ContextObject in Span (COinS)

Heretofore, the discussion has focused on creative uses of the OpenURL via extended-services linking or other features, such as invoking or the citation finder. Innovations with OpenURL v. 1.0 widen beyond this sphere, and one example is ContextObject in Span or "COinS." COinS is the brainchild of several people. According to the COinS Web site, "OpenURL COinS: A Convention to Embed Bibliographic Metadata in HTML" (<http://ocoin.info>), the COinS specification (v. 1.0) itself was written by Eric

Hellman (from OCLC Openly Informatics), but the concept is based on "the ideas, suggestions, and comments of a number of people, including Peter Binkley, Dan Chudnov, Matthew Cockerill, Karen Coyle, Leigh Dodds, Thomas Dukleth, Alf Eaton, Jeremy Frumkin, Tony Hammond, Peter Murray, Ross Singer, Herbert van de Sompel, and Tom Ventimiglia."²

COinS grew out of a desire to create and embed latent OpenURLs into Web pages, thus allowing OpenURL linking on the free Web in sources such as blogs, wikis, open-access journals, search engines, and other resources that might not include subscriber information. In other words, COinS would allow Web developers to embed bibliographic information into their conventional Web pages. If the user had an appropriate plug-in or other activating-agent mechanism to read the COinS, then the user would be able to link from the free Web to his or her appropriate copy. Ross Singer, in a presentation at the NISO workshop OpenURL and Metasearch (September 2005), noted that COinS originally began as a concept called "OpenURL autodiscovery."³ Singer and his coauthors discussed this concept in the April 2005 *Ariadne* article, "Opening up OpenURLs with Autodiscovery."⁴

The COinS Generator (available at <http://generator.ocoin.info/>) is a tool that "will take bibliographic metadata for a citation and produce a 'COinS,' i.e., a snippet of HTML that can be placed on a Web page and processed by Web tools. The COinS generator also has a built-in link resolver, so you can generate a COinS by taking any OpenURL query string and appending it to the base URL, <http://generator.ocoin.info/>."⁵ A sample COinS would look like this:

```
<span class="Z3988"
title="ctx_ver=Z39.88-2004&rft_val
_fmt=info:ofi/fmt:kev:mtx:journal&rft
.atitle=Hegel%27s%20dialectic%20and%20the%2
0recognition%20of%20feminine%20difference&
mp;rft.jtitle=Philosophy%20today&rft
.isn=0031-8256&rft.aulast=Stone&rft
.date=2003&rft.spage=132&rft.epage
=139&rft.volume=47&rft.issue=5">
</span>6
```

The COinS concept allows Web developers to put OpenURL-like links in Web pages by default, so the links are there but are invisible to the ordinary user. To a user who has a link resolver and an activating agent (such as a browser plug-in or bookmarklet), the activating agent will notice these invisible OpenURLs and make use of them by providing context-sensitive links. Director of sales and marketing at Openly Informatics Tim McCormick noted that this is a big difference from the current model, in

which OpenURLs must be explicitly turned on, causing a lot of administration issues associated with the activation process. COinS, on the other hand, allows developers to place the OpenURL-like links in the pages a priori, in anticipation that a user will be able to make use of them by virtue of his or her institution's link resolver.

ContextObject in Span (COinS)

<http://ocoins.info>

"Shoehorning the Sacred into the Profane,"
by Ross Singer (presented at the NISO
September 2005 Workshop: *OpenURL*
and *Metasearch: New Standards, Current*
Innovations, and Future Directions)

www.niso.org/news/events_workshops/OpenURL-05-Agen-FINAL.html

"Opening up OpenURLs with Autodiscovery,"
by Daniel Chudnov, et al.

www.ariadne.ac.uk/issue43/chudnov

COinS Generator

<http://generator.ocoins.info>

COinS: Embedding Sites and List of Processors
(*Activating Agents*)

<http://ocoins.info/#id3205609424>

Hellman explained that once the metadata is in the Web page, there are many ways that a user might activate the OpenURL. Openly Informatics has plug-in software that works with Firefox to accomplish this activation and works with a variety of resolvers. For example, the University of Alabama (UA) could customize this plug-in for its users, so that when an UA-affiliated user downloaded this plug-in, it would recognize his or her base URL, and therefore, recognize the user's affiliation, institutional subscriptions, and profiles.

Bookmarklet software is also a possibility, as are possibilities that links could be activated in a proxy-server environment. Hellman emphasized that the key concept is to put the metadata in the Web page, and then any number of activating agents could make use of this metadata—perhaps even a mechanism that comes standard with Web browsers, much as browsers come standard with other sorts of plug-ins. Currently, the COinS Web site has a list of all the embedding sites as well as a list of COinS processors or activating agents (<http://ocoins.info/#id3205609424>).⁷

Other innovative uses of the OpenURL, such as the work being done with open-access materials and digital objects as well as Google Scholar and projects underway at OCLC, are explored in the next chapter, "Other Linking Issues." Suffice it to say that those working with the OpenURL and particularly the new extensibility of v. 1.0 are not satisfied with the status quo; they are continually pushing the boundaries and exploring new ways to link users to the content they need and desire.

Notes

1. Scott A. Warren, "DOIs and Deep-Linked E-Reserves: Innovative Links for the Future," *Technical Services Quarterly* 22, no. 4 (2005): 1-16.
2. Eric Hellman, "OpenURL COinS: A Convention to Embed Bibliographic Metadata in HTML" (2005), <http://ocoins.info/> (accessed December 8, 2005).
3. Ross Singer, "Shoehorning the Sacred into the Profane" (see chap. 3. n. 10).
4. Daniel Chudnov, et al., "Opening up OpenURLs with Autodiscovery," *Ariadne* 43 (April 2005), www.ariadne.ac.uk/issue43/chudnov (accessed December 8, 2005).
5. ContextObject in Span (COinS), "COinS Generator" (2005), <http://generator.ocoins.info> (accessed December 8, 2005).
6. Ross Singer, "Shoehorning the Sacred into the Profane."
7. Eric Hellman and Tim McCormick, telephone interview with the author, Oct. 5, 2005.