On the Road to the OpenURL

Basic Reference Linking Defined

Reference linking, also called "citation linking," simply describes the user's ability to move from an information object to another.[1] Reference linking can be a link from a bibliography in an online article to the corresponding full text, from a record in a library catalog to the corresponding full text, or from a record in an abstracting and indexing database (A&I) to the corresponding full text (see figure 1 below). In each of these three cases, a user is moving from traditional forms of bibliographic information to the corresponding full text, and initial linking efforts focused their energies on this sort of searching. In particular, initial efforts addressed the need for systematic schemas to enable linking to and from scholarly journals, specifically published journals.

To provide a bit of history, some of these initial efforts in the late 1990s included the Open Journal project, the NASA Astrophysics Data System, and PubMed, from the National Library of Medicine.[2] At the same time, the Library Automation team at the University of Ghent, which included Herbert Van de Sompel, was developing what would become the OpenURL, referred to at the time by the University of Ghent team as "Special Effects" (SFX).[3] In early 2000, Ex Libris acquired the "sole rights to the SFX Reference Linking Software Solution from the University of Gent (Belgium)"; Oren Beit-Arie of Ex Libris worked with Van de Sompel and Patrick Hochstenbach of the University of Ghent to co-author the OpenURL v. 0.1.[4]

Concurrently, a group of individuals and organizations met in 1999 to discuss the future of reference linking as well as the challenges associated with it. This group included the National Information Standards Organization (NISO), the Digital Library Federation (DLF), the National Federation of Abstracting and Information Services, and the Society for Scholarly Publishing. In short, the "participants identified three major components for constructing systems to support reference linking: identifiers for works; a mechanism for discovering the identifier from a citation; and a mechanism for taking the reader from an identifier to a particular item."[5] The OpenURL and other linking schemas discussed herein have attempted to address these components.

Others were working on the appropriate copy problem at the same time, attempting to find solutions to this issue. For example, Chemical Abstract Service (CAS) manager (ChemPort Marketing) Harry Boyle, noted in an interview with the author in 2003 that CAS collaborated with OhioLINK and the Ohio State University library on a linking service, which, in retrospect, was a precursor to the OpenURL. This collaboration centered on the recognition that the key to solving the appropriate copy problem was to get the user into his or her local library by virtue of linking into it. They hit on a solution that required the URL of the user's library system (known as the "base URL" in OpenURL terminology) and fielded metadata. Today, the CAS linking solution—called the "ChemPort Connection"—offers librarians local control of all linking options.

Again, most of the initial efforts focused on linking

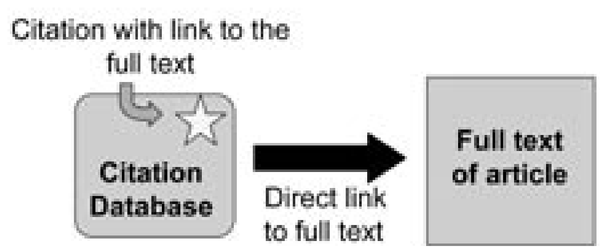

Figure 1
Basic Direct Linking

"Reference Linking for Journal Articles," by Priscilla Caplan and William Y. Arms
www.dlib.org/dlib/july99/caplan/07caplan.html

"Linking Electronic Journals: Lessons from the Open Journal Project," by Steve Hitchcock, et al.
www.dlib.org/dlib/december98/12hitchcock.html

"Reference Linking in a Hybrid Library Environment, Part 2: SFX, a Generic Linking Solution," by Herbert Van de Sompel and Patrick Hochstenbach
www.dlib.org/dlib/april99/van_de_sompel/04van_de_sompel-pt2.html

"In Brief" in D-Lib: *"Ex Libris Acquires Rights to SFX Reference Linking Software," February 2000*
www.dlib.org/dlib/february00/02inbrief.html#BEIT-ARIE

Ex Libris, "OpenURL Overview"
www.exlibrisgroup.com/sfx_openurl.htm

Herbert Van de Sompel
http://public.lanl.gov/herbertv

among traditional bibliographic information and scholarly electronic journals, but as defined previously, reference linking can be much more. Reference linking can also be linking to scholarly materials from informal e-mail messages or courseware systems such as WebCT or Blackboard. In essence, it can take the form of linking between and among any number of information objects, such as grey literature, preprints, postprints, or presentations. It is this sort of unpublished material that proves to be some of the most challenging for any linking schema, as explored in chapter VII, "Other Linking Issues."

Van de Sompel and Hochstenbach first referred to linking beyond classic bibliographic data to full text as "extended services linking" in 1999.[6] Van de Sompel and Beit-Arie further explain that extended services linking, as introduced by Van de Sompel and Hochstenbach, "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."[7] Linking to extended services can include links to interlibrary loan, to document-delivery services, to online bookstores, to library catalogs, and much, much more.

Although reference linking as a whole has come to encompass much more than scholarly journals, basic reference linking still remains the provision of a link from a bibliographic citation to the corresponding full text. In the basic reference-linking environment, two general questions need to be considered for every link. First, where is the link taking the user? And second, what type of link is being provided?

Where Is the Link Taking the User?

Regardless of the type of link, first and foremost, information professionals must understand the path a link is traveling. Before the OpenURL, most linking initiatives were either internal or external to a particular system. Internal linking is most often associated with the one-stop shop—the aggregator or the large publisher. Links can travel internally among reference lists and bibliographic citations in a given content provider. Internal linking, also called "intrasystem linking," is still in practice today, but very few content providers are exclusively utilizing internal linking because the comprehensive one-stop shop is a myth. No one content provider can possibly hope to offer all available scholarly literature, regardless of its size.

One main reason for this impossible dream is the increased demand for links from reference lists to corresponding full text as well as links to unpublished materials and other relevant information objects and extended services. To be blunt, no savvy publisher or content provider can afford to view itself as an island.

The one-stop-shop, internal-linking approach, however, remains alive and well due to large aggregators and publishers. Internal-linking systems are a necessity for these sorts of content providers, and as we shall see, the types of internal links that these content providers choose to use vary widely, depending on the amount and type of full-text content the providers offer to users. For example, with large full-text aggregators such as EBSCO, internal linking may mean "internal" to the provider—in other words, the user doesn't leave the EBSCO*host* environment—but the user may be moving among different databases within EBSCO*host*; therefore, EBSCO must have some sort of mechanism to determine whether or not a particular user has the rights to view material, for instance, in both

"Reference Linking in a Hybrid Library Environment, Part 1: Frameworks for Linking," by Herbert Van de Sompel and Patrick Hochstenbach
www.dlib.org/dlib/april99/van_de_sompel/04van_de_sompel-pt1.html

"Open Linking in the Scholarly Information Environment Using the OpenURL Framework," by Herbert Van de Sompel and Oren Beit-Arie
www.dlib.org/dlib/march01/vandesompel/03vandesompel.html

Academic Search Premier and Business Source Complete. The particulars of the mechanism that EBSCO uses to create these appropriate copy links, as well as approaches other vendors employ, is discussed in more detail in chapter V, "Linking without a Stand-Alone Link Resolver."

Internal linking may seem more attractive because, on the surface, there appears to be no need to address the appropriate copy. If a user has entrance to a content provider's front door, then the user will most likely have access to whatever lies within the provider's content database(s). That sort of reasoning is faulty, however, because in addition to the large aggregator issue previously described, no one provider can hope to include all literature cited in every reference list. Furthermore, many content providers contain a mixture of abstracting and indexing and full text, meaning the user will necessarily have to leave the confines of one content provider's walls in order to move from bibliographic content to the full text housed elsewhere.

For example, ScienceDirect from Elsevier now offers its own full text (both as current subscription and back-file purchase) as well as hosted full text from third parties such as the American Psychological Association (APA).[8] In addition, ScienceDirect offers abstracting and indexing databases such as EconLit, PsycINFO, Medline, and INSPEC.[9] Any given institution may subscribe to only part of what ScienceDirect offers and, therefore, may need to point users to external resources, such as the local library catalog for print materials.

Hence, the necessity of external linking has become commonplace. External linking is simply described as, "when the user leaves the confines of one content provider and enters the confines of another"; whereas internal linking may be conceptualized as closed, external, or intersystem, linking is necessarily open. When explaining linking to students, patrons, and others not well versed in linking technology, it is useful to employ the help of an analogy. Internal linking can be envisioned as the local mall. After entering the front door, one may or may not have access to all the stores within the mall, but the environment itself is closed. On the other hand, external linking can be envisioned as the shopping district downtown. There is no closed environment; shoppers leave one store and enter another in an open environment.

ScienceDirect Info: Content—Journals—Participating Publishers
http://info.sciencedirect.com/content/journals/participating_publishers

ScienceDirect Info: Content—Abstract Databases
http://info.sciencedirect.com/content/databases

It is important to remember that it was not so long ago that content providers were wary of such open, external linking, fearing what this sort of external linking would mean for their products. In 1999, Herbert Van de Sompel and his colleagues at the University of Ghent called for the "cooperation of the information industry," but noted "many established players might be reluctant towards such an idea since it requires far-reaching openness of their services."[10]

Seven years later, it is difficult to imagine a time when content providers were reluctant to link to one another. Vestiges still remain, however, in the restrictions that some aggregators and publishers place on links appearing in their products, including which links can be locally configured, what locally defined text may appear, and more.

The politics of external linking are not the only roadblock. External linking is necessarily more complex because the environment is so very open. Although it can be relatively easy to see the path of internal linking, even within the large aggregators, the external linking path can be much more difficult to map and much more difficult to manage. A content provider has control of its own metadata and link-to syntaxes, therefore, the links traveling within one content provider can be more reliable. Once the environment is open, we see the entrance of variegated metadata schemes and link-to syntaxes. And without some sort of mechanism that controls for appropriate copy, we see users encountering dead ends because direct uni- or bidirectional linking may not take a user's institutional affiliations into context.

Initially, uni- or bidirectional linking was the linking of choice. In bidirectional linking, two content providers reach an agreement to link to one another, such as ProQuest linking to EBSCO and vice versa. Bidirectional linking is direct linking, meaning the link moves directly from ProQuest to EBSCO and/or vice versa. Other direct linking can include unidirectional linking from abstracting and indexing services to full text. There is no third-party intervention in the linking path, with the possible exception of the local information center.

Increasingly, content providers are allowing the local information center or library to configure what sorts of direct links appear within a given product via administrative modules. An example of this sort of local control of direct linking is SilverPlatter's SilverLinker service. SilverLinker is a linking architecture that allows SilverPlatter platform customers to link from SilverPlatter databases to other electronic content and is provided free of charge. Not all content providers offer administrative modules, but most offer local control of direct linking, perhaps through the provider's customer service system. Obviously, it is more convenient for the local library to be able to log into an administrative module to make changes to configure what direct links appear rather than send e-mail to customer-service representatives. Unfortunately, though, this sort of convenience is not uniformly offered. For example, the local library must still e-mail Thomson ISI for products (such as Web of Science) to ask that direct links to particular content providers either display or be suppressed. Subscribers must also work directly with Thomson ISI to set up OpenURL linking and linking to the local library catalog. Thomson ISI, however, currently has an administrative module in development, which should be released in early 2006.

Another example of direct, external linking is that offered by ABC-CLIO to a number of external sources. From an ABC-CLIO database, e.g., "America: History and Life" or "Historical Abstracts," libraries can set up direct links to sources, such as JSTOR and the History Cooperative. Libraries are given control over these links and can turn off links to those full-text resources to which they do not have access. The obvious disadvantage, though, is that if the database provider is not collaborating with the providers of the full-text resources to which a library subscribes, than users do not see links to the full text to which the library does have access, and users subsequently may assume there is no full text available to them.

Direct external linking can also include links from a content provider, particularly vendor A&I databases, to the local library catalog. This sort of "hooks to holdings" has been in development since the late 1980s, and many content providers now facilitate such linking.[11] Direct linking to the catalog can be particularly useful for directing users from citation databases to monograph holdings, because these are not usually available electronically and still require a physical trip to the local library to obtain them.

Any amount of direct, external linking, if configured and consistently maintained by the local information professional and adequately supported by the content provider, increases a user's chances of finding the desired information object. In some cases, direct, external linking may be the optimal choice for a number of reasons, which are explored in chapter V, "Linking without a Stand-Alone Link Resolver."

The discussion thus far has concentrated on linking without the OpenURL, and such linking is still widely used. It is critical that the information professional understand where the link is taking the user—internally or externally—because, inevitably, this is an imperfect world, and the system does not function perfectly every time. If the information professional generally understands where the system failed, then he or she can better help the user and feel less mystified by the dead link, whatever its form. Naturally—after having found information objects they desire and subsequently are denied access—users are quite frustrated by dead links. Although librarians may share this frustration, it is easier to deflate the situation if one feels adequately armed with the knowledge to determine the obstacle on the link path.

What Type of Link Is Being Provided?

Where the link is taking the user (to an internal or an external location) is only half of the issue. The second part of the equation the information professional must understand is what type of link is being provided. In short, there are two basic types of linking: static and dynamic.[12] True static linking is becoming a less common practice among most content providers. At its most elemental, static linking employs URLs, which are "notoriously unstable"; further, static linking "doesn't scale, as URLs generally have to be discovered and supplied manually."[13] By their very nature, URLs are locations, meaning they describe one instance of an information object at a particular location. Static linking usually refers to a one-to-one relationship between the link and one particular copy of an information object. Static linking often takes the form of a "link farm" or link database at a particular vendor, in which links must be periodically verified and updated. Van de Sompel and Hochstenbach explain static linking:

> Links between information entities are computed in advance using batch processes and are held in a linking database. . . . Static links are foolproof in the sense that following a pre-computed link will most certainly lead to the desired target. When considering solutions where bidirectional linking—from now on called interlinking—is the aim, building the linking solution required the availability of all data that needs to be interlinked under the control of the authority creating the environment.[14]

Because static linking is preprocessed, it is very labor intensive. And, as Van de Sompel and Hochstenbach explain, static linking is a centralized approach: all data must be in one entity's control to ensure consistent, working links.

Considering there can be multiple copies of any given information object as well as the practical infeasibility of data being under the control of the creating entity, static linking is necessarily problematic. For comprehensive static linking, a URL would need to be preprocessed for each instance of an information object! As Caplan notes, this simply does not scale.[15] Most linking in the current environment, therefore, is in some way dynamic.

Before explaining dynamic linking, however, it is relevant here to mention in a general way the use of unique identifiers for information objects. URLs are problematic because they describe locations or manifestations. Identifiers, such as the Digital Object Identifier (DOI) or the

PubMed Identifier (PMID), are more scalable in practice because they identify the object itself rather than a manifestation of the object.

Dynamic linking is linking created on the fly, or at the moment of clicking. Rather than relying on a database of preprocessed links, dynamic linking pulls metadata from whatever bibliographic information is provided and populates a URL query. As a simplified explanation, this query is then sent to some sort of resolution service that reads the information in the query and creates a link for the user.

The OpenURL uses dynamic linking technology, as do other linking solutions. Generally, dynamic linking can be less reliable than static linking, but it has several essential components that make it more attractive: it is scalable and as we shall see, it can address the appropriate copy issue. Whereas static linking is centralized, then dynamic linking is decentralized. With a reliable metadata and link-to syntax, dynamic linking does not require the entity creating the links to have in its control all the data needed to create the links. There is more room for error with dynamic linking—due the lack of standardization of metadata across content providers as well as other factors—but it is less labor intensive.

Both internal and external linking can employ either static or dynamic models, thus the need for information professionals to understand clearly where the link is taking the user and what type of link is being provided. Before moving into a more specific discussion of the development of one of the most popular and widespread uses of dynamic linking, the OpenURL, it is useful to take a look at a few concrete examples of basic reference linking in internal and external schemas.

Internal/Intrasystem Linking

When moving within the same system, users are traveling via internally constructed link paths. Prime examples of this include large aggregators and publishers. For instance, a user conducts a search in Expanded Academic ASAP, a Thomson Gale database. The user finds a relevant citation and may then be presented with a suite of possible links to full text, including text or text with graphics. This introduces another element of linking to full text that has not yet been discussed: the type of full text being provided. Based on his or her needs, the user might have the option to choose from the information formatted in HTML text, a mixture of HTML text and graphics, or a PDF.

Intrasystem linking can be either dynamic or static, depending on the content provider and how much internally hosted content a particular vendor may have. As they include more content and offer more extended services, many vendors are moving to dynamic schemas within their own environments. Gale uses a technology it calls "InfoMarks," which is "a stable URL," but it is really a dynamic technology that allows users to save and re-create searches based on the stable InfoMark identifier created for searches, articles, or lists of articles.[16]

Overview of Gale's InfoMark Technology: "What's an InfoMark?"
www.gale.com/infomarks/about.htm

External/Intersystem Linking

Examples of basic reference linking among systems abound and, again, can be either static or dynamic—although dynamic linking is becoming the norm. Most vendors these days offer some degree of external linking, either direct or via the OpenURL. For example, librarians can configure SilverPlatter databases to link to a variety of external services. A user simply conducts a search in a SilverPlatter database and might be presented link options to his or her local catalog, to Project MUSE, or to his or her library's link resolver, all based on the librarian's configuration. A direct link to a content provider, such as Project MUSE, uses the SilverLinker technology.

For external linking to work properly, the crucial element is local control, whether that control is exerted through OpenURL resolvers or through locally configured direct links via administrative modules or via customer-service representatives. The maintenance of internal linking is normally the responsibility of the content provider. In other words, EBSCO offers links to its own internal content and/or determines whether or not the user has rights to access other ESBCO databases once the user is in the initial EBSCO front door.

External linking, on the other hand, is increasingly becoming the responsibility of the local library, and this is how it should be. The local library knows best what content its users have the right to access and therefore is in the best position to determine and to configure those links.

The content provider's role is to facilitate this configuration, either via OpenURL compliance or via localized control of which direct links appear. The less control the local library has over which external links appear, the greater the possibility of a user encountering a dead end, which, of course, is the worse possible scenario. In librarianship, there should be no dead ends, because at the very least, interlibrary loan (ILL) exists to get users the content they need. The ultimate in external linking is the OpenURL because it allows the library to configure what external links appear in one place, including links to an array of extended services, such as ILL.

Notes

1. Priscilla Caplan and William Y. Arms, "Reference Linking for Journal Articles," *D-Lib Magazine* 5, no. 78 (July/August 1999), www.dlib.org/dlib/july99/caplan/07caplan.html (accessed November 28, 2005); Priscilla Caplan, "Reference Linking for Journal Articles: Promise, Progress, and Perils," *portal: Libraries and the Academy* 1, no. 3 (2001): 351.
2. Steve Hitchcock, et al., "Linking Electronic Journals: Lessons from the Open Journal Project," *D-Lib Magazine* (December 1998), www.dlib.org/dlib/december98/12hitchcock.html (accessed November 28, 2005); Caplan and Arms, "Reference Linking for Journal Articles."
3. Herbert Van de Sompel and Patrick Hochstenbach, "Reference Linking in a Hybrid Library Environment, Part 2: SFX, a Generic Linking Solution," *D-Lib Magazine* 5, no. 4 (April 1999), www.dlib.org/dlib/april99/van_de_sompel/04van_de_sompel-pt2.html (accessed November 28, 2005).
4. Oren Beit-Arie, contributor, "In Brief: Ex Libris Acquires Rights to SFX Reference Linking Software," *D-Lib Magazine* 6, no. 2 (February 2000), www.dlib.org/dlib/february00/02inbrief.html#BEIT-ARIE (accessed November 28, 2005); Ex Libris, "OpenURL Overview" (2005), www.exlibrisgroup.com/sfx_openurl.htm (accessed November 28, 2005); Herbert Van de Sompel, http://public.lanl.gov/herbertv (accessed November 28, 2005).
5. Caplan and Arms, "Reference Linking for Journal Articles."
6. Van de Sompel and Hochstenbach, "Part 1" (see chap. 1, n. 1).
7. Herbert Van de Sompel and Oren Beit-Arie, "Open Linking in the Scholarly Information Environment Using the OpenURL Framework," *D-Lib Magazine* 7, no 3 (March 2001), www.dlib.org/dlib/march01/vandesompel/03vandesompel.html (accessed November 28, 2005).
8. ScienceDirect Info: Content – Journals – Participating Publishers, http://info.sciencedirect.com/content/journals/participating_publishers (accessed November 28, 2005).
9. ScienceDirect Info: Content – Abstract Databases, http://info.sciencedirect.com/content/databases (accessed November 28, 2005).
10. Van de Sompel and Hochstenbach, "Part 1" (see chap. 1, n. 1).
11. William A. Britten et al., "Access to Periodicals Holdings Information: Creating Links between Databases and the Library Catalog," *Library Collections, Acquisitions, and Technical Services* 24 (2000): 8.
12. Van de Sompel and Hochstenbach, "Part 1."
13. Caplan and Arms, "Reference Linking for Journal Articles."
14. Van de Sompel and Hochstenbach, "Part 1."
15. Caplan and Arms, "Reference Linking for Journal Articles."
16. Gale's InfoMark Technology (Overview of), "What's an InfoMark?", www.gale.com/infomarks/about.htm (accessed November 28, 2005).