The S-Link-S™ Framework for Reference Linking: Architecture and Implementation

Eric S. Hellman

Openly Informatics, Inc.

10 Columbus Avenue, Suite C
Montclair, New Jersey, 07042
USA
eric@openly.com

Abstract

The Scholarly Link Specification (S-Link-S™) Framework has been proposed as a possible approach towards a solution to the reference linking problem. S-Link-S provides a language for specifying URL generation formulae, and a metadata vocabulary for describing linkable scholarly internet resources. In this paper, I describe the origin of this approach, outline the architecture of the prospective system, and describe the implementation environment.

Keywords: linking, RDF, XML.

Introduction

The internet protocols commonly called World-Wide-Web were initially conceived at CERN as a medium for the communication of scientific research. It is therefore not so surprising that scholarly publishers have embraced the Web as the medium of the future for publication of scholarly journals, even in the face of potentially mortal dangers to their established business

models. Perhaps more surprising to non-publishers is the difficulty that publishers have encountered in taking advantage of capabilities that make the Web so well-suited to scientific communication. The global hyper-linking facility of the Web in particular has been exploited to only a small fraction of its potential.

Insight into the limited hyper-linking of references in internet presentations of scholarly journals can be drawn from my initial experiences with reference hyper-linking an internet journal. In 1995, I founded a free-to-read internet-only journal on behalf of the Materials Research Society, the MRS Internet Journal of Nitride Semiconductor Research (http://nsr.mij.mrs.org/). We began with a database of about 500 citations to the important literature in the field, and used reference lists of papers that we published to feed the database. The database was then used as a target and source for links to and from our published papers. We were very eager to add hyperlinks from the database to appropriate resources on the internet.

Our first reference hyperlinking effort was directed at the Astrophysics Data System (ADS) run with support from NASA by the Harvard-Smithsonian Center for Astrophysics[1]. ADS used database keys synthesized from bibliographic data to allow hyperlinks directly into their system. After implementing links to ADS in early 1996, I concluded that (1) the programming effort to implement each new class of hyperlinks was prohibitive and (2) the effort and cost of obtaining the needed information, and negotiating permissions, even with a willing partner, was also prohibitive.

The second reference hyper-linking project we undertook was equally enlightening. Amazon.com, the hugely successful seller of books on the internet had announced their Associates Program,[2] in which any web site could create links into their book catalog and would receive commissions on any book purchases which resulted. In contrast to many scholarly

---

[1] The NASA Astrophysics Data System (ADS), operated by the Harvard-Smithsonian Center for Astrophysics http://adswww.harvard.edu/

[2] Information about the Amazon.com Associates program is available at http://www.amazon.com/exec/obidos/subst/partners/associates/associates.html

publishers, who wanted us to pay dearly for the privilege of linking to them, Amazon,com was offering to *pay* us to link to them! Sooner than you can click, the theoretical impact of linking on new business models for scholarly journals was made manifest.

The seeds for S-Link-S were thus sown. The design goals were as follows:

1. A single software module must be able to implement a multitude of linking targets.

2. It should be easy to make internet resources linkable and to advertise their linkable status. Linking should not demand target publishers to be technically advanced.

3. Permissions to link should be based on establishment of conventions of linking etiquette rather than direct bilateral agreements.

4. The linking system should be economically self-sustaining and compatible with a variety of business models.

Before discussing the system being developed to meet these goals, it's important to note that many significant efforts to implement reference linking have been undertaken. An excellent discussion of these efforts has been published by Hitchcock et al. [3]. The result that is emerging is that many "puddles" of linking have been established. These puddles have nucleated around field specific databases, such as ADS and Medline/PubMed,[4] or within families of journals published by a single publisher or aggregator. While the ultimate solution may well be a super-puddle nucleated around a very large database, such as that envisioned by the International DOI Foundation,[5] a solution with the capacity to build connections between the puddles may be mare immediately realizable.

---

[3] Steve Hitchcock,Les Carr, Wendy Hall, Steve Harris, Steve Probets, David Evans, David Brailsford, "Linking electronic journals", D-lib Magazine, December 1998, available at http://www.dlib.org/december98/12hitchcock.html

[4] Medline is a service of the National Library of Medicine, available at http://www.nlm.hih.gov/

[5] The International DOI Foundation, http://www.doi.org/

S-Link-S Architecture

The Scholarly Link Specification Framework (S-Link-S) has been developed to meet the design goals just discussed. In this section, I will outline the ways in which S-Link-S addresses each goal. I will not discuss the technical details or specifications, which are available (appropriately enough!) on the Internet.[6]

Software Implementing Multiple Linking Targets

The basic idea of S-Link-S is that URLs to journal related internet resource can often be computed from standard bibliographic data. In order to make a journal URL "calculator", there has to be a way to write URL formulae, which is what S-Link-S provides.

In consideration of the variety of ways that scholarly publishers have structured web-sites, S-Link-S incorporates a URL template language using XML syntax.[7] URL strings are then formed by "filling in the blanks". As an example of a S-Link-S template, consider a journal with a web-site at http://www.publisher.com/ in which articles have URL's based on the volume and page, as shown in table 1. A S-Link-S template for this site is shown in figure 1. In this example, "&startPage;" and "&volume;" are placeholders for the starting page number and the volume number in the URL. The "<pad>"markup indicates that padding 0's need to be added to make fixed-length strings. The template language allows almost any web-site structure to be accommodated. Even when the web-site structure cannot be directly expressed in this language, URL's using search queries can be constructed to the same effect.

Volume	Start Page	URL
3	25	http://www.publisher.com/003/25/
10	485	http://www.publisher.com/010/485/

Table 1. URL's for articles in a hypothetical journal.

---

[6] E.S. Hellman, "The S-Link-S Framework", available at http://www.openly.com/S-Link-S/ .

[7] Information about XML, the eXtensible Markup Language, is available at http://www.w3.org/XML/

```
<?xml version="1.0">
<!DOCTYPE S-Link-S SYSTEM "S-Link-S.dtd">
<S-Link-S ID="example">
<URL>http://www.publisher.com/<pad padChar="0" length="3">&volume;</pad>/&startpage;/</URL>
</S-Link-S>
```

Figure 1. Sample S-Link-S Template document, using XML syntax, for hypothetical journal of Table 1.

Using the Java programming language, Openly Informatics, Inc. has developed software that implements the S-Link-S template language. The use of Java enables one software module to run on operating systems ranging from Linux to Windows NT to MacOS. The software is in the alpha-test process at the time of this writing.

Forming the URL is only part of the technical challenge of linking. In order to cost-effectively build links to a reference, a publisher must (1) parse the reference into bibliographic data, (2) resolve the bibliographic data into a URL, (3) present links so that users will know what to do with them, and (4) maintain the integrity of the link. For each of these tasks, a publisher requires information about the provider of the linkable resource, such as (1) names, identifiers and name abbreviations for the cited articles, (2) how to use the URL strings, (3) the nature of the resource that the link leads to, and (4) the expected longevity and reliability of the link. To address these requirements, the S-Link-S framework defines a metadata vocabulary using the RDF (Resource Description Framework) Syntax.[8] Resource types such as "Books", "Serial", "LinkMethod" and "WebService" are defined along with properties that may be used to describe them. A typical WebService will have several LinkMethod's. each of which can be used to build hyperlinks into a Serial or Books. An "abbrevTitle" property is defined for resources of type "Serial" so that reference processing engines can discover the LinkMethod's applicable to a particular reference. LinkMethod's may have properties such as "linkedObjectType" which informs the linker of the sort of resource being offered, and properties such as "expires" which tell the linker how long the resource provider expects to service the URL's constructed according to the LinkMethod. These metadata can be used by the linking publisher to decide whether or not to offer the link, and if so, how to present it.

[8] Information about the Resource Description Framework is available at http://www.w3.org/RDF/

Authoring and Publishing Linking Metadata

Although RDF and XML are superb platforms for data and metadata exchange between computers, they are hardly friendly to humans. An easy-to-use linking infrastructure based on these platforms will require software that will let people author and publish linking information. Luckily, metadata authoring software enabled by the RDF standards is rapidly emerging.[9] The Internet has made the publishing of raw data extremely easy and cheap; Openly Informatics, Inc. intends to offer a S-Link-S metadata publishing service for free to the publishing community. This service and others like it will make it easy to be linkable.

Linking Etiquette

There are many legitimate reasons for publishers to be hesitant of making linking information completely public. Misuse of linking and appropriation of intellectual property are common concerns. For example, the indiscriminate use of linking formulae to index a site can tax system resources and result in denial of service to legitimate users of a journal. Another abuse of linking is the practice of "framing" a web site so that the content appears to be part of the framer's web offering. Protection from these abuses is one advantage of having legal agreements to govern linking arrangements. On the other hand, putting a legal agreement in place for every possible bilateral linking arrangement is prohibitively expensive. A better arrangement would be for each publisher to enter into a usage agreement with an independent third party, such as the provider of a metadata publishing service such as that described in the previous section.

There are numerous examples of situations were the Internet has evolved conventions such as "Netiquette"[10] and the "Robot Exclusion Protocol"[11] to promote appropriate usage; in

---

[9] See for example, "Reggie, the Metadata Editor" , at http://metadata.net/dstc/

[10] Virginia Shea, "The Core Rules for Netiquette", available at http://www.albion.com/netiquette/corerules.html

[11] Martijn Koster, "A Standard For Robot Exclusion", available at

general, these conventions have worked best when backed up with technical and legal enforcement mechanisms. For example, framing has not been a serious problem on the internet because (1) a technical countermeasure is available, and (2) framers have been served with lawsuits. Establishment of linking etiquette conventions which can be enforced by usage agreements should allay most of publishers' concerns about providing linking metadata.

Sustainability

Economic sustainability is one of the most important aspects of a linking infrastructure, but almost nothing about it has been written. Here's a list of some example business models:

1. Government funding. Both ADS and PubMed, most prominently, have successfully implemented article-database-centered linking in cooperation with scholarly publishers. Both rely on funding from government agencies. The continuance of this infrastructure is thus dependent on governmental stability and the political clout of the user base.

2. Consortium funding. The International DOI Foundation is expected to offer article-database-driven linking funded, at least for now, by membership dues.

3. Subscription sales. Publishers of databases such as ISI's Web of Science are using sales of database licenses to support their linking offerings.

4. Aggregation. Large publishers such as Springer and Elsevier, for example, are offering link services as part of content aggregation services.

A journal-level database such as S-Link-S is potentially very cheap to implement. The size of the database is small enough to put on a low end personal computer of 1999, and data collection costs are minimized by putting metadata authoring tools in the hands of participating publishers. The remaining costs are low enough that they could be offset by software sales and sales of derivative products and services. As an extreme case of how the internet can enable new

---

http://info.webcrawler.com/mak/projects/robots/norobots.html

.

linking business models, consider CiteSeer,[12] which is being implemented as a community effort without any prospective revenue.

Implementation

The biggest challenge to implementation of a system like S-Link-S is the classic chicken-or-egg problem. How do you get a large number of publishers to participate in a linking system that is not useful until a large number of publishers participate? Our strategy is to begin including publishers in a S-Link-S database, without their cooperation, based strictly on the material publicly available on their web sites. This will have two consequences:

1. The database will be immediately useful.

2. Publishers wishing to assert control over the linking into their web-sites will be motivated to participate.

Our efforts in this direction have been rather successful. So far, the staff of Openly Informatics, Inc.has surveyed 76 web-sites which present journal-related resources. These web-sites contain resources for a total of 4,053 journals. Figure 2 shows the level of access to these resources attainable through S-Link-S templates. In this sample, S-Link-S provides public, full-text access to 427 journals, abstract access to 545, and table-of contents access to 2,375 journals. Thus, the S-Link-S linking system should be useful at launch, even before it gains widespread publisher participation.

---

[12] Steve Lawrence, C. Lee Giles, Kurt Bollacker, "CiteSeer Autonomous Citation Indexing", available at http://www.neci.nj.nec.com/homepages/lawrence/citeseer.html

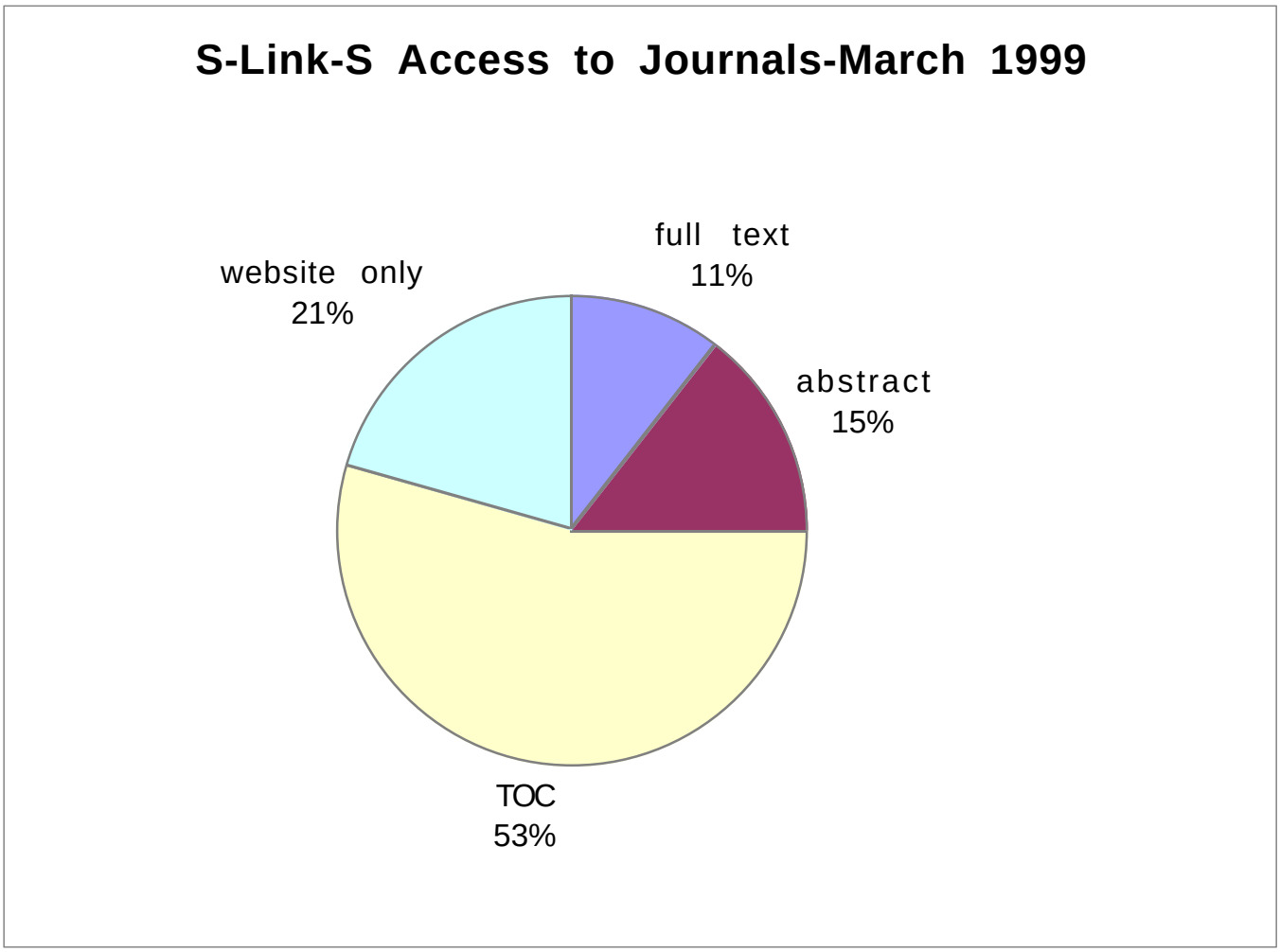

Figure 2. Access level into internet journal resources using S-Link-S, during March 1999. The survey size is 4,053 Journals. Note that many of the sites offering TOC (Table Of Contents) access through S-Link-S allow access to abstracts or full-text of articles, but not using S-Link-S compatible URL's.

Conclusion

S-Link-S offers a rapidly realizable means to deliver reference linking into a broad spectrum of resources published on the internet. It offers a means to interconnect the "pools of linking" that are now nucleating, and potentially enables new business models for scholarly publishing, thus making the scholarly communication infrastructure more useful and robust as it moves into the 21st century.

Acknowledgments

The author would like to credit Miles Poindexter for his work on the S-Link-S survey of internet journals.