



# Accessing LOCKSS Content Through SFX

Philip Gust

Stanford University LOCKSS Program

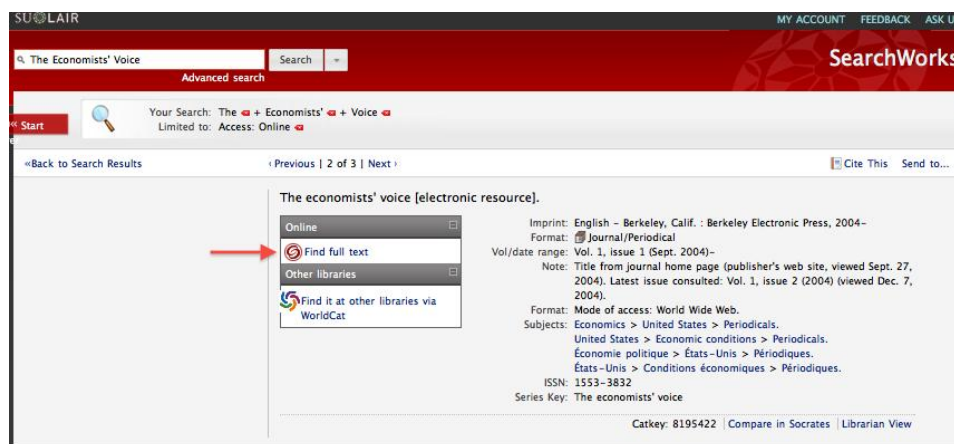
11 November 2010

## Introduction

The LOCKSS (*Lots of Copies Keep Stuff Safe*) digital preservation infrastructure gives libraries control over on-line content they have purchased, by ensuring that the content is preserved locally in case they lose access to it from the publisher. Local ownership is critical to long-term preservation. It separates payment from access, and insulates libraries from future price increases for preservation and access to content they already own. LOCKSS enables libraries to build and preserve real digital collections, with guaranteed perpetual availability, in the same way that they do with print media. This includes subscription and open access materials, e-journals, books, and a wide range of other web delivered content.

Beyond preservation, LOCKSS plays another role: making preserved content available to users when it is not available from the publisher. This is important when the publisher discontinues the publication, the library discontinues its subscription to the publication, or access to it though the publisher is temporarily interrupted because of a breakdown in IT infrastructure. In these situations, LOCKSS is like an “uninterruptable power supply” for preserved content.

To fulfill this role, a library's local LOCKSS box must be integrated into the library IT infrastructure, and make preserved content available through the same user interfaces that libraries normally provide. The most common user interface in a library is the Online Public Access Catalog (OPAC). When a user locates a title in the catalog that is available on-line, the displayed record includes a link to access the title:



The link provides a way to request content from one or more *target* content suppliers. The form that is displayed for a specific target enables the user to request an article using its bibliographic information:

SU LAIR Help

find it STANFORD

Source: The economists' voice [1553-3832]

---

**Full Text**

Full text available via **Berkeley Electronic Press**

Year:  Volume:  Issue:  Start Page:

Available from 2005 volume: 2 issue:5

In this example, the target is the publisher, who returns a landing page where a PDF rendition of the article can be downloaded.

THE BERKELEY ELECTRONIC PRESS™  
bepress™: the new standard in scholarly publishing since 1999

[My Account](#)

## The Economists' Voice

[Vol. 6 \(2009\) / Issue 2 / Columns](#)
[« Previous Column](#)
[Next Column »](#)

### Yes We Can, Secretary Geithner

**Luigi Zingales**, *University of Chicago-GSB*

**Summary**

Luigi Zingales makes suggestions to Secretary Geithner on how to resolve the financial crisis.

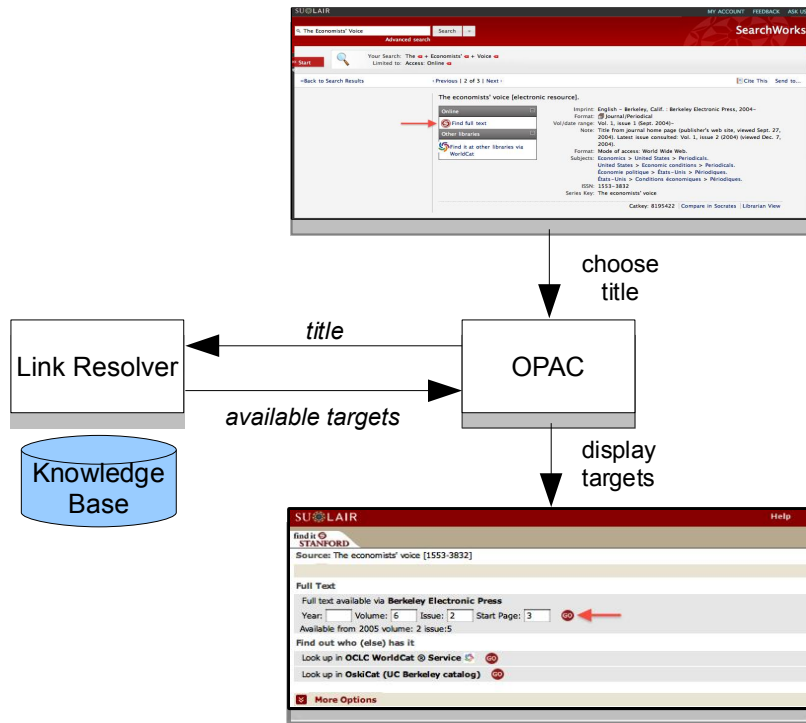
[Tell a Colleague](#)
  
[Print](#)

## Link Resolvers to the Rescue

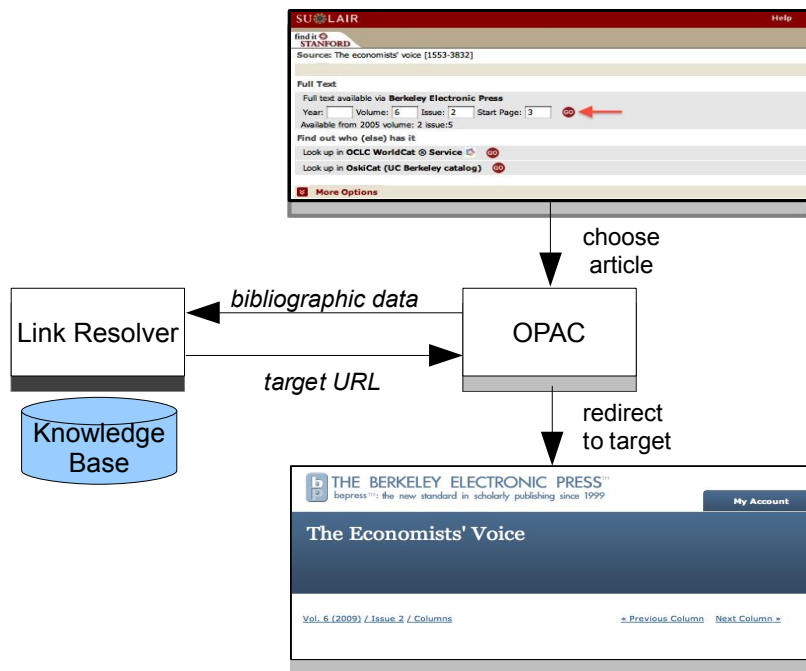
*Context-sensitive linking*, accessing on-line content based on bibliographic data, is a complex process. Rather than requiring each OPAC to implement this functionality, the library community developed a technology, known as a *link resolver* (or *link server*), that OPAC vendors can integrate with their systems. A link resolver transforms a request in terms of its bibliographic data into a Uniform Resource Locator (URL) “link” to the content at a target. The publisher is the only target in this example. If the title were also available from other targets, such as *Web of Science* or *Google Scholar*, the OPAC would display a line for each, and the link resolver could provide the URL to the content at any of them.

### **How they work**

To accomplish this, the link resolver maintains a *knowledge base* of titles that are available from each target. Entries in the knowledge base include title ISSN or ISBN, the range of volumes available from the target, and a target-specific method for transforming bibliographic information into a content URL. When an OPAC displays a title, it retrieves a list of available targets from the link resolver.



When the user selects a target and enters the bibliographic information for the content, the OPAC sends this information back to the link resolver. The link resolver generates and returns the URL for the requested content at the selected target. Finally, the OPAC *redirects* the browser to this URL.



## Benefits

The link resolver frees users from having to know which targets can supply specific content, and how to interact with a target to retrieve it. It also frees libraries from having to maintain their own knowledge base of titles and targets. Vendors who supply link resolver products revise their knowledge base from publisher data, and periodically push updates out to library customers under their product maintenance agreements. Most OPAC vendors support integrations with one or more link resolvers.

The first and one of the most widely used link resolvers is SFX (an abbreviation for “special effects”). It was developed by Herbert Van de Sompel at the University of Ghent in the late 1990s and subsequently purchased by the library automation company Ex Libris Ltd. Other companies who market link resolvers include OCLC (WorldCat Link Manager), Swets (SwetsWise Linker), Serials Solutions (360 Link), and many others. There are also several open-source link resolvers, including CUFTS and Umlaut.

## Integrating LOCKSS with the SFX Link Resolver

Integrating LOCKSS with link resolvers is an effective way to make LOCKSS content available to library users. OPACs already use link resolvers to provide access to a library's on-line content. The integration of a LOCKSS box as a link resolver target provides users with a reliable way to access content, regardless of whether it is still available from the publisher or another target. LOCKSS becomes the target of choice for accessing preserved content, because it automatically “fails over” to the content archived in the LOCKSS box if it is no longer available from the publisher.

The screenshot shows the SUOLAIR interface. At the top, there is a red header with 'SUOLAIR' and a 'Help' link. Below the header is a search bar with the text 'find it STANFORD'. The search results show 'Source: The economists' voice [1553-3832]'. Below this, there are two sections for 'Full Text'. The first section is for 'Berkeley Electronic Press' and has input fields for Year, Volume, Issue, and Start Page, with a 'GO' button. The second section is for 'LOCKSS' and has input fields for Year, Volume (6), Issue (2), and Start Page (3), with a 'GO' button and a red arrow pointing to it.

Since SFX is the most widely used link resolver, the initial integration focuses on SFX. Most of the work required to integrate with SFX can also be leveraged to integrations with other link resolvers.

## Enhancements to SFX

All the SFX-specific enhancements for integrating LOCKSS with SFX are to the SFX product, in cooperation with Ex Libris. The first is to add LOCKSS as a target in SFX. The LOCKSS target definition includes a parameter that allows a library to specify their local LOCKSS box as part of its local SFX configuration. Target URLs returned by SFX are to the LOCKSS box specified by this parameter. For example, the parameter for the LOCKSS box at XYZ University may be `http://lockss.xyz.edu:8082/`.

The second enhancement is to define a method for creating a LOCKSS target URL from bibliographic information. SFX refers to this as a *target parser*. It is a simple code module written in the Perl programming language that is associated with the target definition. LOCKSS has created a target parser that Ex Libris will distribute to its customers along with the LOCKSS target definition.

When the user selects the LOCKSS target for a title in an OPAC, SFX runs the LOCKSS target parser with the user's bibliographic data. The LOCKSS target parser formats the data into an OpenURL, an industry standard way to encode bibliographic information in a URL. SFX redirects the user's web browser to this location, and the LOCKSS box uses the information in the OpenURL to return the requested content. The OpenURL returned by the LOCKSS target parser for the earlier example is

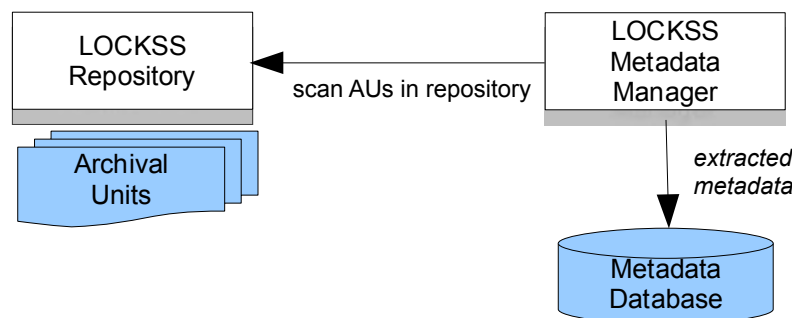
```
http://lockss.xyz.edu:8082/ServeContent?issn=1553-3832&volume=6&issue=2&spage=3.
```

The third enhancement is to add titles preserved in the LOCKSS box to the SFX knowledge base. LOCKSS will periodically supply Ex Libris with a current list of titles available for preservation. Ex Libris will add the list to the knowledge base it distributes to its customers. Ex Libris accepts title lists using an industry standard format specified by the Knowledge Bases and Relational Tools ([KBART](#)) working group. Since most link resolver vendors now accept this format, LOCKSS can send the same list to other vendors as we add to our list of supported link resolvers.

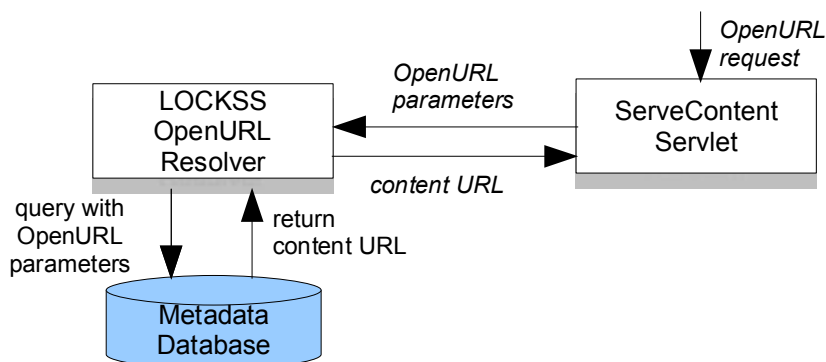
## Enhancements to LOCKSS

Several enhancements are required for the LOCKSS to process OpenURL requests from a link resolver. LOCKSS manages *Archival Units* (AUs) for titles. AUs are collections of files that are convenient as units of preservation. Accessing preserved content using bibliographic information requires an additional database of metadata about individual articles, issues, and volumes contained in AUs, and their locations. This database is maintained by a new LOCKSS subsystem called the *Metadata Manager*.

Extracting metadata is a time-consuming process, so the Metadata Manager works in the background, scanning recently preserved AUs, and extracting metadata such as title ISSNs, volume and issue numbers; and article DOIs, titles, authors, and page numbers. The Metadata Manager adds this information to the metadata database, along with the content locations.



Finally, the LOCKSS box's *ServeContent* has been extended to recognize OpenURLs, and pass them to a new subsystem, the *OpenURL Resolver*. The OpenURL Resolver accepts bibliographic information from OpenURL parameters, and uses the metadata database to return the URL of the requested content. *ServeContent* and the OpenURL Resolver accept a full range of OpenURL queries, in both the simpler OpenURL 0.1 and the richer OpenURL 1.0 formats.



If the content specified by the returned URL is preserved by the local LOCKSS box and is the most recent version from the publisher, ServeContent returns the content in the local LOCKSS repository. If the content is not preserved by the local LOCKSS box or the publisher's version is more recent, ServeContent returns the content from the publisher. This is similar to a web browser returning locally cached content.

## Summary

The integration of LOCKSS and link resolvers like SFX is an effective way for libraries to provide reliable access to the digital content that they own, even if they lose access to it through the publisher. Most OPACs allow users to access on-line content from title records through close integrations with commercial and open-source link resolvers. Together, LOCKSS, link resolvers, and OPACs simplify access by using existing library IT infrastructure and user interfaces that are already familiar to library users.

Although the details of integrating LOCKSS with a link resolver are specific to the vendor, many of the vendors have adopted common practices, enabling the work done for one integration to be leveraged to integrations with other vendors' systems. LOCKSS has been enhanced to retrieve archived content in terms of its bibliographic data, encoded in the industry-stand OpenURLs format. As a result, link resolvers can simply encode a user request as an OpenURL, and redirect the request to the local LOCKSS box.

For content that is archived in a local LOCKSS box, the LOCKSS target for a title displayed by a library's OPAC is the most reliable source, because it automatically “fails over” to the archived version if access through the publisher becomes unavailable, like an “uninterruptable power supply” for content.

## Further Reading

“OpenURL”. [<http://en.wikipedia.org/wiki/OpenURL>].

“The Lure of Linking”. Library Journal.com. 4/1/2004. [<http://www.libraryjournal.com/article/CA405398.html>].

“LOCKSS”. [<http://www.lockss.org/>].

Reich, Victoria A. “Lots of Copies Keep Stuff Safe as a Cooperative Archiving Solution for E-Journals”, Issue in Science and Technology Librarianship. Fall 2002. [<http://www.istl.org/02-fall/article1.html>].

“SFX”. [[http://whatis.techtarget.com/definition/0,,sid9\\_gci927719\\_00.html](http://whatis.techtarget.com/definition/0,,sid9_gci927719_00.html)].

“SFX: the OpenURL Link Resolver and Much More”. [<http://www.exlibrisgroup.com/category/SFXOverview>].