LOCKSS, A PERMANENT WEB PUBLISHING AND ACCESS SYSTEM: BRIEF INTRODUCTION AND STATUS REPORT

David S. H. Rosenthal and Vicky Reich

LOCKSS (http://lockss.stanford.edu) stands for Lots of Copies Keep Stuff Safe. It is an Internet "appliance", or "easy to use" software, designed to preserve access to authoritative versions of web-published materials. The current version of LOCKSS software is restricted to electronic journals.

Dr. David S. H. Rosenthal, Sun Microsystems Laboratories, 901 San Antonio Road, UMTV29-112, Palo Alto, CA 94303, USA
Tel: 1 650 336 1025
E-mail: dave.rosenthal@eng.sun.com:

Vicky Reich, Assistant Director and Digital Librarian, HighWire Press, Stanford University Libraries and Academic Resources, 1454 Page Mill Road, Stanford CA 94304
Tel: 1 650.725.1134
Fax: 1 650.725.6553
E-mail: vreich@stanford.edu

LOCKSS allows individual libraries to take custody of content in all formats delivered via HTTP, and safeguard their community's access to it. Using LOCKSS, a library can ensure that, for their readers, hyperlinks continue to resolve and content is delivered, even when in the Internet those links do not work and the content is no longer available. LOCKSS ensures that the locally held content maintains its integrity through a polling and reputation system; LOCKSS replicas co-operate to detect and repair preservation failures. LOCKSS is designed to run on very cheap hardware and to require almost no technical administration. The software will be distributed as open source.

Problem

Web published materials are increasingly the authoritative versions. There are no affordable, widely available techniques for preserving this "written record". The web is an effective publishing medium (data sets, dynamic lists of citing papers, e-mail notification of citing papers, hyperlinks, searching). As web editions increasingly become the 'version of record', paper versions of the same titles are merely a subset of peer-reviewed scholarly discourse. Librarians need an inexpensive, robust mechanism, that they control, to ensure that their communities maintain long-term access to this important literature.

Requirements

The solution to this problem is in three parts:

- the content must be preserved as bits;
- access to the bits must be preserved;
- the ability to parse and understand the bits must be preserved.

There is no single approach to solving this problem. Any single

solution would be perceived as vulnerable. By proposing LOCKSS, we are not discounting other digital preservation solutions. Other solutions must also be developed and deployed. Diversity is essential to successful preservation.

Technical details

At each library, LOCKSS uses off-the-shelf, open source software to manage a web cache for each journal that the library wishes to safeguard, and to pre-load the cache with the pages of the journal as they are published. Thus, pages will be preserved even if they are not read. Through these caches each library takes physical custody of selected web journals it purchases. Unlike normal caches, pages in these caches are never flushed; the caches grow indefinitely as the journals continue to publish. Over time, the disks holding an individual cache will fill up or fail. Librarians will be able to replace full or failed disks without interruption to the system. Nor will they loose access to any data previously cached.

A key innovation is the way LOCKSS caches detect and recover from failures: how the system ensures data preservation. They use a newly designed inter-cache protocol called LCAP [Library Cache Auditing Protocol]. LCAP allows caches to conduct "opinion polls", which provide assurance that the local copy of all or part of a journal matches the majority of other copies, and provide a lower bound on the number of other copies in existence. If a cache detects that its content is incomplete or otherwise corrupt, it asks the publisher or one of the other caches to provide a replacement copy. LOCKSS caches respect publisher's access control mechanisms. They will only provide content to caches that have proved in the past that they had a copy. The publisher web site's access to end user "click stream" data is not affected by a library's LOCKSS cache.

Depository models

There are two approaches to digital preservation and archiving: centralized and decentralized. Key questions are – what are the costs of preserving what kinds of materials and on whom do they fall?

A decentralized system has a large number of loosely controlled repositories. Each repository or node in the system: a) does some but not the whole job of preserving the content; b) uses relatively inexpensive hardware, and c) needs relatively little technical expertise to maintain the hardware and software.

The content at each repository is in constant use, under constant scrutiny, and undergoing continual repair. In a decentralized system, the publishers take little or no action to preserve the content they publish, whilst the librarians take action to preserve access for their local communities. Librarians bear the costs of digital preservation but the costs are spread across many participants and only the participants' gain value from the system.

A centralized system has a small number of tightly controlled repositories. Each repository:

a) does the entire job, and b) requires large expensive hardware, with sophisticated technical staff. Content is accessed after a "trigger" event (migration, publisher failure, etc.). To establish a centralized system, publishers and librarians must take co-operative legal and data management actions. The costs of preservation are borne by a few.

We predict that some combination of these approaches will ultimately be implemented.

Three perspectives: Readers, librarians, and publishers

The reader's perspective

A key goal of the LOCKSS system is to preserve a reader's access to content published on the web. Readers expect two kinds of access. They expect that:

- when they click on a link to it, or type in a URL, the relevant page will be delivered, with minimal delay and no further interaction;
- when they enter terms into a search engine that should match the relevant page, it will be among the returned matches.

Readers who use the web are learning that, if a link does not resolve to a page, or a search engine cannot find a page, further attempts to find the information that the page carries are unlikely to be worth the effort. This poses problems for those who use preservation techniques that concentrate on preserving bits. The bits may be preserved, yet the reader may

not know how to access them, or even that the preserved bits exist.

In contrast, the design of LOCKSS focuses on preserving the service of having links resolve to, or searches to find the relevant content. An institution using the LOCKSS system to preserve access to a journal, in effect, runs a web cache devoted to that journal. Readers use the cache as a proxy in the normal way. At intervals the LOCKSS cache crawls the journal publisher's web site and pre-loads itself with newly published (but not yet read) content. Just as other types of caches are invisible to their users, so are LOCKSS caches. They transparently supply pages and they preserve them, even if those pages are no longer available from the original publisher's web site.

An institution can include the contents of the cache among the pages indexed by its local search engine, and provide its readers with searching across all the journals to which it subscribes. At present, readers typically have to search individual collections of journals separately.

The librarian's perspective

Librarians subscribe to journals on behalf of their readers, in order to provide both immediate and long-term access. With the advent of the web, libraries are forced, for the most part, to lease rather than own the web-based content. Leasing provides immediate access but carries no guarantee of long-term access. Some journals provide their peer-reviewed content through off-line storage media (tape, CD-ROM, paper), but then links do not resolve and searching is harder to accomplish.

A major flaw with web publishing is that there has been no mechanism to implement the traditional purchase-and-own library model. The LOCKSS system is demonstrating that it is both easy and affordable to operate a purchase model for web journals. The subscribing library bears costs analogous to the costs of putting paper copies on shelves, keeping track of them and lending or copying them as needed. A library, using LOCKSS, caches to preserve access to a collection of journals and pays for the equipment and staff time to run and manage a cache containing the full content of the journals. Unlike normal caches, the LOCKSS cache is never flushed and, over the long term, the full content remains accessible.

Because individual libraries must pay for the preservation of the content to which they subscribe, it is essential that the price they pay be as low as possible. LOCKSS software is free and open-source. It is designed to run on inexpensive hardware. The machines that the LOCKSS team is using for the beta test cost less than $800 each and each machine is capable of storing the content of 5 years worth of a major journal's issues. Running a LOCKSS cache requires so little staff time that one alpha test site complained they learned nothing about the system over the course of 10 months while running it. The low cost and democratic structure of the LOCKSS system – each copy is as valuable as any other – empowers smaller institutions to take part in the process of digital preservation.

In normal operation, an ordinary cache will only act as a proxy for, and thus supply content to, the host institution's own readers but in a rough analog of inter-library loan, LOCKSS caches co-operate to detect and repair damage. If damage to a page is detected, the LOCKSS cache fetches a copy of the page from the publisher or from another cache. A LOCKSS cache will only supply a page to another LOCKSS cache, if the requesting cache at some time in the past proved that it had the requested page. In this way, LOCKSS prevents freeloading. Those who contribute to the preservation of the journal are rewarded with continued access. Those, who do not contribute to the journal's preservation, are not provided with replacement pages.

The publisher's perspective

Publishers want to maintain journal brand and image. They want material available for future society members and other subscribers. Most publishers will save money and serve their readers better, if the transition to electronic-only journals can be completed. They want to encourage libraries to purchase and/or activate online versions of journals. One major obstacle to libraries purchasing online journals is resistance to the rental model, with its lack of credible assurance of long-term access.

Many publishers are unhappy with a purchase model for electronic journals. They fear that the

journal content will be illegally replicated, or leaked, on a massive scale once copies are in the custody of others; they want their access control methods enforced. They want to retain access to reader usage data and have access to the record of the reader's interactions with their site.

The LOCKSS system solves the reader's and the librarian's problems. It enables librarians to collaborate to preserve readers' access to the content to which they subscribe, but it also addresses the publisher's concerns. Because content is provided to other caches only to repair damage to content they previously held, no new leakage paths are introduced. Because the reader is supplied preferentially from the publisher, with the cache only as a fallback, the publisher sees the same interactions they would have seen without LOCKSS caches.

The LOCKSS design has other advantages from the publisher's perspective:

It returns the responsibility for long-term preservation, and the corresponding costs, to the librarians. Although publishers have an interest in long-term preservation, they cannot do a credible job of it themselves. Failures or changes in policy by publishers are the event librarians are most interested in surviving.

Publishers could run LOCKSS caches for their own journals and, by doing so, over time could audit the other caches of their journals. A non-subscriber cache would eventually reveal itself by taking part in the damage detection and repair protocol. The mere possibility of detection should deter non-subscribers from running LOCKSS caches. Just as publishers cannot be sure that they have found all the caches, the caches cannot be sure none of the other caches belongs to the publisher.

Project status:

The alpha test

The LOCKSS project started in 1999, funded by NSF, Sun Microsystems, and Stanford Libraries. The Alpha software was tested in year 2000 with ~15 caches of ~160MB from AAAS Science Online. Alpha sites were Stanford, U.C. Berkeley, LANL, Tennessee, Harvard and Columbia. The basic mechanisms of the software work. The system survived a fire at LANL, network problems at Stanford, relocation of the machine at Berkeley, and flaky hardware at Columbia.

The beta test

The worldwide beta test began 4/2001, funded by the Andrew W. Mellon Foundation, Sun Microsystems, and Stanford Libraries.

As of September 2001 45 participating libraries in five continents have signed onto the project (*see Appendix A*). 53 publishers are endorsing the LOCKSS beta test (*see Appendix B*).

The beta is testing LOCKSS security, usability, and software performance, including impact on network traffic. The publisher's web sites are simulated on shadow servers (~10-15 GB of PNAS, JBC, BMJ, Science Online) to isolate LOCKSS data streams and measure network traffic and test if the system works when the publisher "goes away". If resources allow, we plan to add to this test bed content from other publishers, particularly those who publish materials on publishing platforms not yet represented in the LOCKSS system.

Each beta site has a slightly different machine(s) and network configuration. Each one is currently running LOCKSS software version 06122001 and is participating in testing. For most libraries, the software was easy to install and is easy to maintain. This phase of testing, however, revealed the challenge of building a system to work easily with different international network configurations. These challenges have been met.

Formalized software testing has begun. The beta test, if funding allows, is scheduled to run to summer 2002.

APPENDIX A	**Libraries participating in the LOCKSS beta test**
Africa/Middle East	
Israel:	Hebrew University
South Africa:	University of Stellenbosch
Asia/Pacific	
Australia:	University of Melbourne
Hong Kong:	Hong Kong University of Science & Technology
New Zealand:	University of Auckland
	University of Otago
Singapore:	National University of Singapore
Europe	
Belgium:	University of Ghent
Finland:	Helsinki University of Technology
Germany:	University of Munich,
	University of Goettingen
Italy:	IEI-CNR, Italian National Council of Research.
Netherlands:	University of Amsterdam,
	University of Maastricht
Norway:	University of Bergen
Scotland:	Edinburgh University,
	University of Glasgow
Spain:	University of Alicante
Sweden:	Lund University
United Kingdom:	The British Library,
	Cambridge University,
	Imperial College,
	University of Leeds
North America	
Canada:	University of Toronto
United States:	Stanford University,
	University of California Berkeley,
	Columbia University,
	University of Tennessee,
	Los Alamos National Laboratory,
	Harvard University,
	Carnegie Mellon University,
	Cornell University,
	Emory University,
	Library of Congress,
	University of Chicago,
	University of Indiana, University of Minnesota,
	University of Texas Austin,
	Yale University,
	University of Oklahoma Health Science Center,
	University of Nevada Reno, New York Public Library,
	Case Western Reserve University,
	Iowa State University,
	National Agricultural Library,
	University of Virginia
South America	
Brazil:	BIREME

APPENDIX B Publishers endorsing the LOCKSS beta test

American Association for the Advancement of Science
American Physiological Society
Federation of American Societies for Experimental Biology
Biophysical Society
Annual Reviews
Rockefeller University Press
American Society for Biochemistry and Molecular Biology
American Association for Clinical Chemistry
National Academy of Sciences
British Medical Journal
American Psychiatric Publishing Inc.
Oxford University Press
Company of Biologists Ltd
New England Journal of Medicine
American Society for Clinical Investigation
Radiological Society of North America
Society for General Microbiology
The Endocrine Society
The Histochemical Society
American Thoracic Society
BMJ Publishing Group
American Society of Neuroradiology
Lipid Research Inc.
American Society for Investigative Pathology
American Society of Plant Biologists
The Royal College of Psychiatrists
Society for the Study of Reproduction
American Society for Microbiology
Cold Spring Harbor Laboratory Press
American Society for Pharmacology and Experimental Therapeutics
Society for Molecular Biology and Evolution
American Society for Nutritional Sciences
BioMed Central
Genetics Society of America
Investigative Ophthalmology and Visual Science
Botanical Society of America
American Heart Association
American Society of Hematology
The American Physical Society
American Academy of Pediatrics
American Society for Microbiology
Cold Spring Harbor Laboratory Press
American Society for Pharmacology and Experimental Therapeutics
Society for Molecular Biology and Evolution
American Society for Nutritional Sciences