Distributed Digital Preservation: Lots of Copies Keep Stuff Safe

Victoria Reich
LOCKSS Program,
Stanford University Libraries
Stanford, CA 94301
1 650 725 1134
vreich@stanford.edu

David S.H. Rosenthal
LOCKSS Program
Stanford University Libraries
Stanford, CA 94301
1 650 725 1134
dshr@stanford.edu

ABSTRACT
The LOCKSS Program (www.lockss.org), based at Stanford University Libraries, provides libraries with tools that allow them, easily and inexpensively, to collect today's Web-published materials and preserve them for tomorrow's readers. In this paper, we review the threat model upon which the LOCKSS software was engineered, explain briefly how the software works, and describe a selection of ways that communities are currently using this resource.

Categories and Subject Descriptors
H [**Information Systems**]: H.3. Information Storage and Retrieval. H.3.7 Digital Libraries

General Terms
Design, Economics, Reliability, Security

Keywords
Digital Preservation, replicated storage

1.INTRODUCTION
Archiving systems are designed to keep content accessible for the very long term. In June 2005, the United States National Research Council recommended to the United States National Archives that the designers of a digital preservation system need a clear vision of the threats against which they are being asked to protect their system's contents [7]. To our knowledge, the LOCKSS (Lots Of Copies Keep Stuff Safe) Program, is the only preservation system designed around an explicit threat model [6]. It addresses an unusually broad set of threats that could interfere with keeping digital content safe, including those that are technical, economic, and social.

The LOCKSS Program, founded in 1998, is grounded in ACM award winning research [4]. The technology provides an OAIS-compliant, open source, peer-to-peer, decentralized, distributed, digital preservation infrastructure for preserving all formats and genres of static content published on the Web. As far as possible, it collects and preserves not merely the intellectual content of the content, but also its original look and feel. Libraries run a "LOCKSS Box" that collects the content they are interested in preserving. The Boxes form a peer-to-peer network that continually audits the content to detect any damage or loss, and to repair it from the copies at other Boxes. Content is preserved in its original format; if this format becomes obsolete the system can, on demand, create temporary access copies in a newer format to provide transparent, on-access format migration.

Libraries and publishers worldwide are using the LOCKSS distributed digital preservation technology. LOCKSS helps libraries stay relevant by building collections even as an increasing portion of today's content is "born digital" and published on the Web. LOCKSS replicates the traditional model of libraries keeping physical copies of books, journals, etc. in their collections, making it possible for libraries to house copies of digital materials long-term.

2.THREATS TO THE PERSISTANCE OF DIGITAL INFORMATION
Digital preservation systems have one fundamental goal, to protect material for a very long time. Threats to keeping content safe over protracted timescales include: technology failures, economic failures, and social failures.

The LOCKSS system designers judged these three key properties critical for any digital preservation system:

1. The system must have no single point of failure; it must not just tolerate the failure of any individual component, it must be designed to tolerate more than one simultaneous failure.
2. Media, software and hardware must flow through the system over time as they fail or become obsolete, and are replaced. The system must support diversity among its components to avoid monoculture vulnerabilities, to allow for incremental replacement, and to avoid vendor lock-in.
3. The system must audit the content at regular intervals frequent enough to keep the probability of failure at acceptable levels.

Specifically, the LOCKSS engineers designed the LOCKSS digital preservation system to guard against these thirteen threats.

1. Media Failure. All storage media must be expected to degrade with time, causing irrecoverable bit errors,

and to be subject to sudden catastrophic irrecoverable loss of bulk data such as disk crashes or loss of off-line media.

2. Hardware Failure. All hardware components must be expected to suffer transient recoverable failures, such as power loss, and catastrophic irrecoverable failures, such as burnt-out power supplies.
3. Software Failure. All software components must be expected to suffer from bugs that pose a risk to the stored data.
4. Communication Errors. Systems cannot assume that the network transfers they use to ingest or disseminate content will either succeed or fail within a specified time period, or will actually deliver the content unaltered.
5. Network Services Failures. Systems must anticipate that the external network services they use, including resolvers such as those for domain names and persistent URLs, will suffer both transient and irrecoverable failures both of the network services and of individual entries in them. As examples, domain names will vanish or be reassigned if the registrant fails to pay the registrar, and a persistent URL will fail to resolve if the resolver service fails to preserve its data with as much care as the digital preservation service.
6. Media & Hardware Obsolescence. All media and hardware components will eventually fail. Before that, they may become obsolete in the sense of no longer being capable of communicating with other system components or being replaced when they do fail.
7. Software Obsolescence. Similarly, software components will become obsolete. This will often be manifested as format obsolescence when, although the bits in which some data was encoded remain accessible, the information can no longer be decoded from the storage format into a legible form.
8. Operator Error. Operator actions must be expected to include both recoverable and irrecoverable errors. This applies not merely to the digital preservation application itself, but also to the operating system on which it is running, the other applications sharing the same environment, the hardware underlying them, and the network through which they communicate.
9. Natural Disaster. Natural disasters, such as flood, fire and earthquake must be anticipated. Other types of threats, such as media, hardware and infrastructure failures, will typically manifest then.
10. External Attack. Paper libraries and archives are subject to malicious attack; there is no reason to expect their digital equivalents to be exempt. Worse, all systems connected to public networks are vulnerable to viruses and worms. Digital preservation systems must either defend against the inevitable attacks, or be completely isolated from external networks.
11. Internal Attack. Much abuse of computer systems involves insiders, those who have or used to have authorized access to the system. Even if a digital preservation system is completely isolated from external networks, it must anticipate insider abuse.
12. Economic Failure. Information in digital form is much more vulnerable to interruptions in the money supply than information on paper. There are ongoing costs for power, cooling, bandwidth, system administration, domain registration, and so on. Budgets for digital preservation must be expected to vary up and down, possibly even to zero, over time.
13. Organizational Failure. The system view of digital preservation must include not merely the technology but the organization in which it is embedded. These organizations may die, perhaps through bankruptcy, or their missions may change. This may deprive the digital preservation technology of the support it needs to survive. System planning must envisage the possibility the preserved content being transferred to a successor organization, or otherwise being disposed of properly.

For each of these types of failure, it is necessary to trade off the cost of defense against the level of system degradation under the threat that is regarded as acceptable for that cost.

2.1Economic Failure

Given the unprecedented world economic crisis; it is appropriate to focus for a few paragraphs on the threat, 'Economic Failure'. The biggest threat to digital preservation over long (or even short!) time frames is the uneven flow of money to support digital preservation. Library budgets are and have been for decades, under extreme pressure. Money spent on digital preservation means less money is available for other important tasks, such as acquiring new materials.

Unfortunately, digital preservation cannot be accomplished in fits and starts; it must be protected from uncertain funding cycles. Keeping digital content unchanged is an active process; the content must be continuously audited, repaired and preserved if it is to remain accessible. The LOCKSS system addresses this requirement in two ways, by minimizing the cost to each library, and by distributing copies of the content among a large number of independently funded libraries.

In the LOCKSS system, ingest, preservation and dissemination are all highly automated, minimizing the staff time required of participating libraries. The system does not require expensive, enterprise-scale technology; it works well with low-cost consumer technology using the replication and cooperation inherent in the preservation network to provide reliability.

Since each LOCKSS Box serves a limited community, there is no need to re-create expensive, high-volume publishing

platforms. Little of the technology is new; the system, for the most part, re-packages Web crawler, Web proxy, Web server and peer-to-peer technologies. The system is entirely open source, both in terms of the components it re-uses and the new technology the team developed to re-package them.

LOCKSS makes it economically and technically possible for even relatively small libraries to actively preserve their own content. The costs of development and support are spread widely, reducing the impact on individual sites, and improving sustainability by diffusing the impact of individual funding decisions.

3.HOW LOCKSS WORKS

3.1Making Content LOCKSS Compliant

The publisher (or whomever is hosting the content on the Web) gives explicit permission for the LOCKSS system to collect and preserve content via the Manifest Page.

The LOCKSS system requires the manifest page contain either a suitable Creative Commons license, or, one of the following texts:

- LOCKSS system has permission to collect, preserve, and serve this Archival Unit.
- LOCKSS system has permission to collect, preserve, and serve this open access Archival Unit.

The Manifest Page also links to new content as it is published.

3.2Bringing a LOCKSS Box Online

A library uses LOCKSS software to turn a low-cost PC into a digital preservation appliance called a LOCKSS Box.

The administrator downloads a CD image, burns it to a CD, and uses the CD to boot a PC, turning it into a LOCKSS Box that automatically joins the international distributed preservation network [2]. The LOCKSS Box is a persistent Web cache that is never flushed. Library staff administers their LOCKSS Box via a Web user interface. The interface enables new content preservation, monitors the preservation of existing content, controls access to the appliance, and a wide variety of other functions [Figure 1].

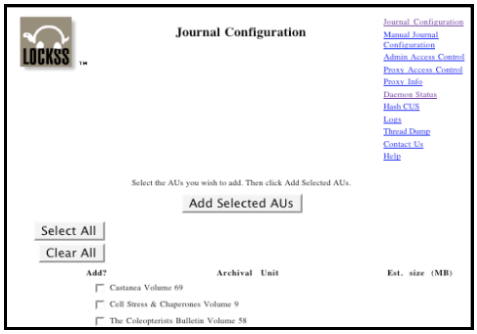

Figure 1: LOCKSS Box Journal Configuration user interface.

The LOCKSS Box then performs four functions.

3.3Ingest

LOCKSS Boxes ingest content directly from target Web sites using a Web crawler similar to, but not the same as, those used by search engines. The LOCKSS crawler has specific information about each publisher's Web site through a small XML file called a plugin. Every publishing site requires a plugin. The plugin knows where to find the publisher's LOCKSS permission statement, how far to follow the chains of Web links, what times of the day are permissible for content collection, how fast content can be collected, etc.

Once the Plugin is written and tested, it is distributed to authorized LOCKSS Boxes. The Stanford University Libraries LOCKSS team uses a set of infrastructure tools to manage the content ingest process, including a Plugin repository, which is a Web server with signed JARs of plugins; and a Title database, which is a list of content available for preservation.

Each LOCKSS Box independently collects its content from the publisher's Web site. The Boxes then compare their collected content to each other's using the polling mechanism described in the next section. The content's authoritative version is established by agreement among the Boxes.

3.4Preservation

The LOCKSS Boxes continually audit the content they are preserving via the LOCKSS peer-to-peer polling protocol. If the content in one LOCKSS Box is damaged or incomplete, that LOCKSS Box receives repairs from the other LOCKSS Box peers. The LOCKSS audit mechanism proves at regular intervals that a copy agrees with the consensus of the copies at peer libraries. The more LOCKSS Boxes hold a copy of given content, the safer it is. The more organizations preserve given content, the stronger the guarantee that they will all have continued access to it. Seven copies is the recommended minimum for normal conditions.

The cooperative polling mechanism among the LOCKSS Boxes performs a number of functions. The initial polls confirm that what is being preserved is exactly what the publisher published, The polls detect and repair damage, avoiding the need to back up a individual LOCKSS Box. It also provides unambiguous reassurance that the system is performing its function and that the correct content will be available to readers when accessed. The LOCKSS Box interface displays how the content is being preserved [Figure 2].

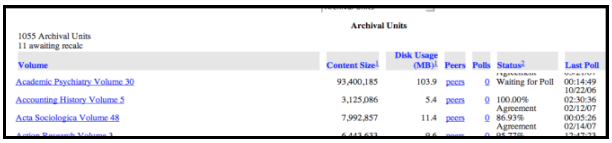

Figure 2. LOCKSS Box preservation status user interface

The open source software is updated on a six-week cycle. Updates are automatically pushed to LOCKSS Boxes via a property server that controls the system parameters.

3.5Dissemination

LOCKSS Boxes provide an institution's readers with continuous access to content preserved in that institution's LOCKSS Box. When the LOCKSS Box is properly configured within the institutional network, it will transparently deliver content to readers when the original Web site is unavailable. LOCKSS Boxes provide access to content via one of two methods: transparent proxy and serving content.

3.5.1Transparent Proxy

The first access method is via Web proxy. Institutions often run Web proxies, to allow off-campus users to access their journal subscriptions, and Web caches, to reduce the bandwidth cost of providing Web access to their community.

An institution can integrate their LOCKSS Box into these systems, which intercept requests from the community's browsers to the journals being preserved [1]. Then, whenever a reader requests a preserved page, whether by clicking on a link to it, selecting a bookmark for it, or typing the URL, behind the scenes the proxy infrastructure intercepts the request and forwards it to the LOCKSS box, which forwards it to the original publisher. If the publisher delivers the content, then that's what the reader gets. If the publisher doesn't deliver content to the reader, for example if a journal subscription was canceled, then the LOCKSS Box does. In this way, when the proxy access method is used, the page remains accessible at its original URL so that links, bookmarks, etc. still work. The reader need not know the LOCKSS Box exists.

3.5.2Serving Content

The second access method is by serving the content. In this method, the LOCKSS Box acts as another Web server from which a reader can obtain the content, thus the LOCKSS Box is visible to readers. Readers are guided to the content they need by their library's text search and URL resolution systems such as SFX, so these systems need to know about the LOCKSS Box and its contents

If the content has internal links, for example the links from Page 1 to Page 2 of an article and vice versa, the LOCKSS Box needs to rewrite these in the version it sends to the reader so that they point to Page 2 and Page 1 in the LOCKSS Box, not to Page 2 and Page 1 at the original publisher, as they do in the preserved version that the LOCKSS Box is storing. Unlike the proxy method, a few links in content delivered in this way may not work because the URL rewriting cannot be guaranteed to be perfect.

3.6Format Migration

The LOCKSS system stores only the original content. It postpones migration until it is needed, when a reader is using a browser that cannot render the original bits. When this format is determined in this way to be obsolete a LOCKSS Box can, on demand, create temporary access copies in a format that the browser does understand to provide transparent, on-access format migration. Web formats become obsolete only when the majority of browsers no longer render that format. This is unlikely to happen quickly, if at all.

The LOCKSS approach to format obsolescence also minimizes costs by re-using existing technologies [5]. This approach exploits the time value of money and allows each reader to see the result of the state-of-the-art in migration at the time of his or her access. It leverages the pervasive adoption of open source browser technology for format migration and object rendering by providing a framework in which both open and closed source format converters can be deployed as they become available.

From the Library of Congress, "If a format is widely adopted, it is less likely to become obsolete rapidly, and tools for migration and emulation are more likely to emerge from industry without specific investment by archival institutions…. Evidence of wide adoption of a digital format includes bundling of tools with personal computers, native support in Web browsers. [3].

4.Distributed Preservation Networks

LOCKSS is currently being used to preserve content in two distinct types of environments: a public LOCKSS Network and a variety of Private LOCKSS Networks (PLNs). The publics LOCKSS network are preserving materials of general interest to a wide community. The Private LOCKSS networks are preserving more specialized materials for smaller communities.

4.1Public Networks

The public network preserves materials that are generally available on the Web, including subscription-only material. Sufficient replication is ensured because the materials preserved in the public network are those that the wider community has agreed they wish to preserve. The public network is maintained by the Stanford University-based LOCKSS staff with funding provided by the LOCKSS Alliance.

The public LOCKSS network comprises nearly 200 research institutions worldwide. Libraries can join LOCKSS at no charge; however, libraries serious about building local collections join the LOCKSS Alliance for a nominal annual fee. Alliance member's LOCKSS Boxes can collect and preserve materials to which they subscribe, along with other materials such as open access titles. The preservation requirement for more than seven copies is achieved because the content is of general interest. Most of the institutions participating in the public LOCKSS network preserve most of the content available to them, especially the content to which they subscribe. The average replication factor in the public LOCKSS network is approximately 40.

Access to content held in an institution's public LOCKSS Box is "light" to that institution's authorized users. LOCKSS members are permitted to access the content to which they subscribe whenever that content is unavailable from the publisher.

4.2Private Networks

The LOCKSS technology also enables librarians and archivists to create and manage their own preservation network via a "Private LOCKSS Network" (PLN). A PLN is a scaled down version of the public LOCKSS network.

The PLN model offers institutions with synergistic collections a means to ensure the survival of content that is outside the

collection development scope of most institutions. In other words, content preserved in a PLN is often more akin to a library's special collections, digitized images, local Web sites, etc.

Private LOCKSS Network partners explicitly agree to work together, to share in the preservation of each other's specialized content. Network members confirm that a potential member institution will bring value to the network, for example, by contributing appropriate content and having a working LOCKSS Box online. Typically a PLN has seven to 15 institutional participants.

Each PLN requires some technical administration. The network needs to be monitored and the two databases associated with the ingest process need to be maintained. A community can manage its own technical infrastructure or the Stanford University-based LOCKSS staff can manage the infrastructure. Most often the LOCKSS staff helps with technical installation, and then a local community accepts as much responsibility as is appropriate and comfortable for them. Local, relatively non-technical staffs are using the software to implement, manage, and govern their own distributed digital preservation networks.

Each PLN establishes its own policies and practices:

- Governance. Governance and administrative structures vary from formal contractual agreements to collegial understanding among colleagues.
- Funding. PLNs are funded by self-administered fees, via grants, or contributed effort.
- Collection development. PLN participants share an interest in preserving a particular type of publication (for example government documents or ETDs) or a specific subject area.
- Access. Each PLN community sets its own access policies based on local needs, resources, and the intellectual property rights associated with the content. Most PLNs are dark, however, with content access via a hosting platform such as ContentDM or DSpace.

While there is variance among PLNs, all institutions participating in a PLN are LOCKSS Alliance members. The continued development of the open source LOCKSS software is essential for their PLN to be successful.

4.2.1Private LOCKSS Network, Examples

More details, and links to each of these initiatives is available, http://www.lockss.org/lockss/Private_LOCKSS_Networks

- PeDALS . Preserving State public records.
- DataPass. Preserving Social Science data.
- MetaArchive. Cooperative. Preserving Southern Culture.
- CLOCKSS. Preserving published scholarly ejournals and ebooks.
- LOCKSS Docs. Preserving United States Federal Documents.
- Council of Prairie and Pacific University (COPPUL) Consortium. Preserving content of interest to the Western Canadian Region.
- Alabama Digital Preservation Network (ADPN). Preserving content of interest to the state of Alabama.

5.CONCLUSION

Most libraries, even those with small staffs, can easily and affordably take possession of and preserve Web published content, including purchased books and journals.

6.REFERENCES

[1] Integrating a LOCKSS Box into network infrastructure. http://www.lockss.org/lockss/Proxy_Integration

[2] Installing a LOCKSS Box. http://www.lockss.org/lockss/Installing_LOCKSS

[3] Library of Congress. National Digital Information Infrastructure and Information Preservation Program (Jul. 2007). Sustainability for Digital Formats: Planning for Library of Congress Collections. http://www.digitalpreservation.gov/formats/sustain/sustain.shtml

[4] Maniatis, Petros, et. al, (Oct. 2003). Preserving Peer Replicas By Rate-Limited Sampled Voting. 19th ACM Symposium on Operating Systems Principles (SOSP), Bolton Landing, N.Y. http://www.eecs.harvard.edu/~mema/publications/SOSP2003.pdf

[5] Rosenthal, David S. H., et. al. (Jan. 2005). Transparent Format Migration of Preserved Web Content. D-Lib Magazine, Volume 11, Number 1. http://www.dlib.org/dlib/january05/rosenthal/01rosenthal.html

[6] Rosenthal, David S. H. et al. (Nov. 2005). Requirements for Digital Preservation Systems, A Bottom-Up Approach. D-Lib Magazine, Volume 11 Number 11 ISSN 1082-9873, http://www.dlib.org/dlib/november05/rosenthal/11rosenthal.html

[7] Sproull, Robert F. and Eisenberg, Jon, Editors, Committee on Digital Archiving and the National Archives and Records Administration, National Research Council (2005). Building an Electronic Records Archive at the National Archives and Records Administration: Recommendations for a Long-Term Strategy. ISBN-10: 0-309-09696-0 http://www.nap.edu/catalog/11332.html