

Mitigating Risk of Data Loss in Preservation Environments

Reagan W. Moore
San Diego Supercomputer
Center
moore@sdsc.edu

Joseph F. JaJa
University of Maryland
joseph@umiacs.umd.edu

Robert Chadduck
National Archives and Records
Administration
robert.chadduck@nara.gov

Abstract¹

Preservation environments manage digital records for time periods that are much longer than that of a single vendor product. A primary requirement is the preservation of the authenticity and integrity of the digital records while simultaneously minimizing the cost of long-term storage, as the data is migrated onto successive generations of technology. The emergence of low-cost storage hardware has made it possible to implement innovative software systems that minimize risk of data loss and preserve authenticity and integrity. This paper describes software mechanisms in use in current persistent archives and presents an example based upon the NARA research prototype persistent archive.

1. Introduction.

Preservation environments, called persistent archives, support the long-term storage of digital records [1]. A persistent archive manages retention of both the digital record content and a preservation context that describes

the origin, relevance, and preservation properties associated with each digital record. Persistent archives assert that the digital record remains authentic, that the digital record is a true copy of the original digital record deposited into the archive, and that the person and judicial processes that created the digital record remain correctly identified. Authenticity requires multiple preservation properties: that the digital record remains unchanged, that the preservation context correctly tracks information about preservation processes performed upon the digital record, and that the chain of custody of the digital record remains unbroken. A persistent archive provides the mechanisms to validate assertions of authenticity, even while the technology used to implement the persistent archive evolves over time [2]. For the National Archives and Records Administration (NARA), a reasonable goal for the preservation period is the lifetime of the republic, nominally 400 years. During this time period, the persistent archive will need to migrate the preserved content across one hundred generations of storage systems. In effect, the preservation environment needs to manage the preserved material independently of the choice of current storage technology.

The traditional approach to data management is to assume that the selected storage repository is responsible for maintaining authenticity and integrity of the deposited records. Since current file systems do not manage the metadata attributes required for asserting authenticity, the preservation metadata can be packaged with each digital entity in an Archival Information

¹ The Storage Resource Broker was developed under the technical lead of Michael Wan and Arcot Rajasekar at the San Diego Supercomputer Center. Applications of the SRB technology were done by Wayne Schroeder, George Kremenek, Sheau-Yen Chen, Charles Cowart, Lucas Gilbert, Bing Zhu, and Marcio Faerman. This work was supported in part by the NSF NPACI ACI-9619020 (NARA supplement), the NSF Digital Library Initiative Phase II Interlib project, the NSF NSDL/UCAR Subaward S02-36645, the DOE SciDAC/SDM DE-FC02-01ER25486 and DOE Particle Physics Data Grid, the NSF National Virtual Observatory, the NSF Grid Physics Network, the NSF Southern California Earthquake Center, and the NASA Information Power Grid. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the National Science Foundation, the National Archives and Records Administration, or the U.S. government.

Package (AIP) [3]. The AIP is written to a storage repository, and a separate database is used to track the location of each electronic record. The requirements for preservation of authenticity can then be imposed as hardware capabilities that ensure against data corruption or data loss. The commercial storage system is tasked with replicating the data, mirroring the state information about the storage location of each AIP, maintaining a file name space for the AIP, maintaining information about archivists who are allowed to execute preservation processes, managing access controls, and managing audit trails to track what has happened. Such systems can implement the preservation requirements in hardware and incorporate software systems to manage the preservation context (typically a database). The archivist picks the solution that minimizes risk of loss of authenticity, while simultaneously minimizing total cost of preservation.

With the advent of low-cost commodity storage systems [4], it becomes possible to lower the cost of long-term preservation by moving risk mitigation technology into the preservation environment software. Instead of requiring that the storage system internally protect against data loss through hardware redundancy, the electronic records can be replicated onto multiple lower-cost storage systems. This paper looks at the types of software mechanisms that help mitigate the risk of data loss, examines how the software mechanisms are implemented in data grids, and provides an example of a preservation system based upon the NARA research prototype persistent archive. Special attention is paid to the scalability of the preservation software mechanisms, to ensure that preservation environments can sustain the petabytes of data and millions of records that are now being created.

2. Types of risk for loss of authenticity

One way to proceed with a risk analysis is to quantify the types of data and information that

must be preserved. We characterize the bits that comprise the digital records as the *content*, and the preservation metadata for asserting authenticity as the *context*. We can then consider systems that optimize the preservation of content (such as file systems and archival storage systems), systems that optimize the preservation of context (such as databases), and the systems that are needed to associate context with content (such as data grids). This approach makes it possible to implement management mechanisms that are scalable. Content is streamed through preservation procedures in bulk operations and stored on storage repositories after aggregation in containers. Context is managed in databases using bulk metadata manipulation procedures on metadata that has been aggregated into XML files.

The assertion of authenticity requires that all operations performed upon a digital entity can be tracked, and that the resulting state information can be maintained for examination at any future time. Data grids provide this capability. The preservation metadata is maintained in a database along with a handle that points to the storage location of the electronic record. The preservation context is archived as files through dumps of the database metadata. Thus both the preservation metadata and the electronic records can be encapsulated in files. If the integrity of the electronic records and the integrity of the database can be maintained, then the authenticity can also be maintained if the preservation context is updated after each operation on a preserved electronic record.

The mechanisms that ensure scalability must interoperate with the mechanisms that protect against loss of authenticity. We consider software systems that support bulk operations on replicated content and federated context. Replication of content corresponds to the creation of multiple copies and the tracking of each copy. Federation of context corresponds to synchronization of two independent databases (preferably from different vendors) that each

hold the preservation metadata under constraints imposed for both access controls and update consistency [5].

Typical risks and the associated software risk mitigation mechanisms are:

- Media failure – handled by replication on multiple media. Examples of risks include disk crashes and tape corruption. At the San Diego Supercomputer Center, current commodity disks have a mean lifetime of 6 years. A 15-Terabyte disk farm made of 200-GB disks has a disk failure on average once a month. Tape lifetimes are similar (on the order of five to ten years). Both disks and tapes are replaced predominantly to recover floor space by using higher capacity media. Disk and tape errors typically compromise files in a storage volume.
- Vendor-hardware/software systemic error – handled by replication onto another vendor product. Examples of risks include corruption caused by RAID controllers when writing data and loss of location information through database corruption. Such problems can compromise all electronic records on the corrupted system.
- Operational error – handled by federation with an independent preservation environment. Examples of risks include procedures that are compromised during upgrades to new storage systems. The assumptions under which a procedure is executed may no longer be valid for the new storage environment. Such problems can compromise all electronic records written after the system upgrade.
- Natural disaster – handled by federation with a preservation environment at a geographically remote site. Examples of risks include fire, flood, and earthquake. While quite infrequent, the entire archive may be compromised.
- Malicious user – handled by federation with a deep archive. Examples of risks include compromise of the Unix operating system environment through security holes that are

not related to the storage system. The authenticity of electronic records cannot be assured when either the preservation context or the content may be changed surreptitiously. A deep archive is a preservation environment that forces all accesses to be local, that handles all updates through creation of versions, and that prohibits external user access. Content and context are sent to a staging area for ingestion into the deep archive. Data and metadata are moved from the staging area under the control of archivist procedures.

Protecting against all types of risk requires the distribution of data across multiple sites using multiple vendor products. A minimum of two sites is required with the second site acting as a slave to the first site. Data and metadata that are registered into the first site are asynchronously sent to the second site. The second site can be implemented at a geographically remote location using storage systems and databases provided by different commercial vendors than used for the first site. The second site storage systems can be operated independently of the first site, with upgrades to new equipment and procedures done at different times. Each site supports a separate preservation metadata database.

The deep archive can be implemented as a separate data management system at one of the sites using an independent database and storage repository. Since both the preservation metadata and electronic records must be staged into the deep archive, the deep archive can also be implemented at a third geographic location. The goal is to minimize the opportunity for a single person to compromise all copies of a record and the associated preservation metadata.

The distribution of replicas and metadata across multiple sites using heterogeneous storage systems managed under separate administrative domains requires the use of data grid technology. Data grids provide the interoperability mechanisms needed to manage

data distributed across multiple types of storage systems. Federations of data grids provide the authenticity coordination needed to replicate preservation metadata. By replicating both preservation context and the preservation content onto multiple systems, it is possible to decrease the reliability requirements for each individual system, while providing an opportunity to address risks that are inherent in relying upon technology from a single vendor.

3. Minimizing cost of preservation

Minimization of preservation cost is achieved by relying upon commodity storage platforms. The San Diego Supercomputer Center uses Grid Bricks [4] to provide low-cost commodity-based disk storage for multi-terabyte collections. The Grid Bricks are modular storage systems that integrate a 1.7 Ghz CPU, a gigabyte of memory, a Gigabit Ethernet network connection, and 5 Terabytes of disk. At the end of calendar year 2004, such systems could be implemented at a cost of \$2000 per terabyte.

Each Grid Brick provides a minimal set of capabilities:

- naming convention for files (physical file name)
- naming convention for users (file owner)
- storage location (network address)
- association of a context with each digital entity (typically creation time, file size, ownership, update time)
- consistency controls for the update of the context and access controls (implemented through a Linux file system)

A preservation environment must extend these capabilities across multiple Grid Bricks, across distributed storage sites and between federated data grids. This is equivalent to implementing a name space for each capability that is controlled and managed by the preservation environment, independently of the commodity storage system. The preservation environment needs to own and manage five name spaces [6]:

- naming convention for the digital entities (logical file name)
- naming convention for the users (distinguished user names)
- naming convention for the storage resources (logical resource name)
- naming convention for the context attributes (metadata name space)
- naming convention for the consistency constraints (knowledge concept space)

Each naming convention implements a logical name space for the associated identifiers. The logical name space for digital entities is taken as the principal name space onto which the preservation context, access control and consistency constraints are mapped [7]. The persistent archive maps from the logical file name space to the physical file name space as provided by a particular vendor product (storage repository file name, database binary large object, object in a ring buffer). The distinguished name space for users provides a single sign-on authentication environment, using systems such as Grid Security Infrastructure certificates or a challenge response authentication mechanism. The user names are managed independently of the storage systems by having each electronic record written under a Unix ID associated with the data grid. Access controls are maintained by the data grid on the logical file name space. A user authenticates himself/herself to the data grid, the data grid checks the access controls for permission to manipulate the electronic record, the data grid then authenticates itself to the storage repository, and then the file is retrieved and transmitted to the user.

The logical resource name space makes it possible to associate operations with sets of physical resources. Replication can be implemented as a write operation on a logical resource name with the constraint that the write completes when copies exist on each of the storage systems in the list. Load leveling can be implemented as a write operation on a logical

