Prototype Preservation Environments

Reagan W. Moore and Richard Marciano

Abstract

The Persistent Archive Testbed and National Archives and Records Administration (NARA) research prototype persistent archive are examples of preservation environments. Both projects are using data grids to implement data management infrastructure that can manage technology evolution. Data grids are software systems that provide persistent names to digital entities, manage data that are distributed across multiple types of storage systems, and provide support for preservation metadata. A persistent archive federates multiple data grids to provide the fault tolerance and disaster recovery mechanisms essential for long-term preservation. The capabilities of the prototype persistent archives will be presented, along with examples of how the capabilities are used to support the preservation of email, Web crawls, office products, image collections, and electronic records.

Prototype Preservation Environments

The San Diego Supercomputer Center (SDSC) collaborates with the National Archives and Records Administration (NARA) on research on the development of a prototype persistent archive. The collaboration examines how advanced data management systems can be used to support the long-term preservation of data. The original goal included an assessment of mechanisms for management of technology obsolescence. The ability to migrate electronic records to new storage systems was called "infrastructure independence." The preservation system should be extensible and be able to use more cost-effective storage technologies as they become available. A second goal was the assessment of scalability mechanisms that would enable

LIBRARY TRENDS, Vol. 54, No. 1, Summer 2005 ("Digital Preservation: Finding Balance," edited by Deborah Woodyard-Robinson), pp. 144–162

support for archives holding hundreds of millions of files and hundreds of terabytes of data. The data management technology that meets these goals is called a "data grid." This article examines how data grids support preservation requirements.

Preservation is the process of migrating a digital entity forward in time while preserving its authenticity and integrity.[1] Authenticity is an assertion that a specific digital entity can be identified relative to the context in which it was created. The context includes provenance information such as the creator of the digital entity, procedural information such as the processes that were used to create the digital entity, and administrative information such as the institution that authorized the digital entity creation. The integrity of a digital entity is an assertion that the information content of it has not been modified, that the chain of custody can be verified, and that transformations on its encoding format were performed by identified archival procedures.

A digital entity can be an electronic record, a data file created by a scientific application, a text file created by a word processing system, an image taken by a remote sensor, or any string of bits that can be named. The preservation process requires the extraction of the digital entity from the environment in which it was created and the import of it into the preservation environment. Once the digital entity is under the control of the archivist, then the authenticity and integrity properties can be implemented with assurance that continued access is sustainable. This article looks at the challenges that must be overcome when extracting a digital entity from its creation environment, the technologies that can be used to manage authenticity and integrity, and some examples of preservation environments.

Preservation Challenges

The idea that a digital entity can be extracted from its creation environment is called *infrastructure independence* (Moore et al., 2000). A digital entity depends upon both software and hardware infrastructure to ensure its support and management. Thus, a file resides in a file system that provides a storage location, a name for the file, management of file properties, names for the persons who are allowed to manipulate the file, and controls on the type of permitted operations. The file properties typically include the size of the file, the owner of the file, the date the file was created, and the date the file was last modified. The extraction of the digital entity from this supporting environment requires the ability to impose

- storage of the digital entity at a location specified by the archivist
- a persistent naming convention for the digital entity that remains invariant as the digital entity is moved between storage systems

- management of file properties that are needed to assert authenticity and integrity
- persistent identifiers for the archivists who are managing the preservation environment
- persistent management of the access controls for allowed operations.

Infrastructure independence means that no matter where the digital entity is stored, the archivist retains the ability to control each of the support properties, independently of the mechanisms provided by a particular choice of storage system. Ideally, an archivist would be able to import a digital entity into a preservation environment that guarantees that the naming conventions will persist through all future choices of technology. One way to implement infrastructure independence is to insert a data management layer between a digital entity and the underlying storage environment. The archivist controls the persistent naming conventions through the data management layer. This approach is illustrated in figure 1.

In the original creation environment, the application that created the digital entity interacted directly with the storage system (shown by the dashed arrow). In the preservation environment, the applications that are used for display and manipulation now interact with a storage system through a data grid, in which the digital entities have been organized as a collection (Rajasekar, Marciano, & Moore, 1999). The data collection is used to assign metadata attributes to each digital entity to manage the authenticity and integrity properties.

The data grid provides its own naming conventions to describe the logical storage location, the logical file name, the metadata attributes, the distinguished names for the archivists, and the control and consistency mechanisms. Each logical name space that is managed by the data grid is essential for implementing infrastructure independence. The logical name spaces can be used to manage digital entities that are distributed across multiple storage systems and located at multiple sites around the country. The logical name spaces make it possible to use global identifiers that do not change when a digital entity is moved to another storage system. We can illustrate this by considering examples of how each logical name space would be used by a preservation environment.

Data Grids

The software infrastructure that implements a collection-based data management infrastructure for distributed data is called a data grid (Foster & Kesselman, 1999). The software infrastructure runs as an application (or server) on each computer platform that manages a storage system. The data grid servers talk to each other in a federated environment. Messages can be sent between servers to move files, replicate files, and access files. The digital entity properties managed by the data grid are stored in a data-

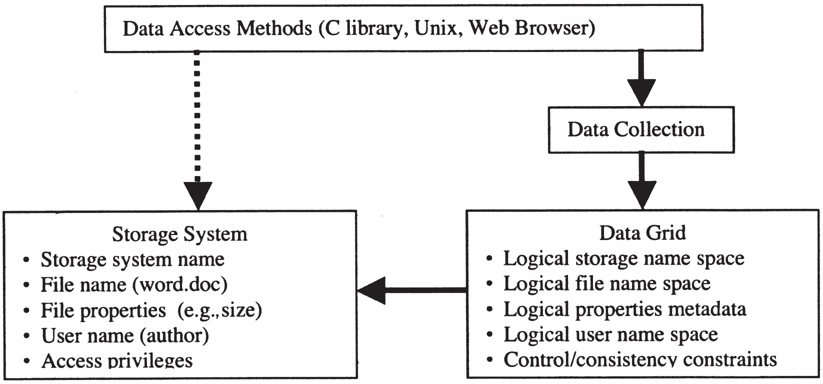
Data Access Methods (C library, Unix, Web Browser)
Data Collection
Storage System
• Storage system name
• File name (word.doc)
• File properties (e.g., size)
• User name (author)
• Access privileges
Data Grid
• Logical storage name space
• Logical file name space
• Logical properties metadata
• Logical user name space
• Control/consistency constraints

Figure 1. Implementing Infrastructure Independence

base as metadata attributes. The metadata attributes are updated after each data grid operation.

The logical storage system name is used to simplify the management of new versions of storage system technology. Assume that the archivist has successfully stored the digital entities in a cost effective storage system. At some point in the future, a new more cost-effective storage system becomes available. For an infrastructure independent system, the archivist would like to be able to swap out the old storage system and replace it with the new technology. From the point of view of the preservation environment, the storage system identity (logical storage system name) should not change, even though the physical address of the storage location (network Internet Protocol [IP] address) will change. Data grids accomplish this by maintaining a mapping from the logical storage system name to the actual physical location (network IP address) of the storage system. The logical storage name can represent multiple physical storage locations (that is, correspond to a list of network IP addresses). Writing to the logical resource name can force the creation of a replica at each physical storage location. Thus, to swap out an old storage system, the archivist adds the new storage system IP address to the list of storage addresses represented by the logical storage system name, replicates each digital entity onto the new storage system, and then removes the old storage system. From the point of view of the data grid, the storage system is still represented by the same logical resource name, even though the physical storage location has changed.

The digital entity replication process requires the use of the other logical name spaces listed in figure 1. The logical file name, similar to the logical storage system name, can represent a list of physical names, in this case multiple copies or replicas of a file. Operations on the logical file name, then, cause operations on each of the replicas, no matter where they are located. The data grid maintains a mapping from the logical file name to the location of each replica. When the digital entity is replicated, a new entry is added to the list of physical copies maintained for each logical file name.

This scenario is reasonable if the new storage system is managed by the same system administrator and the same user names (Unix Identifiers) are available on the new storage system. Now consider a case where the copy is made at a remote site that is not using the same Unix Identifiers. The name of the owner of the logical file should remain invariant in this process, even if the copy is made in a different administration domain at another site. Data grids provide a logical name space for users that is common across all of the federated storage systems. The logical name space for users is managed by the data grid independently of the storage systems. One can ask how the data grid is able to write files that are to be owned by an archivist who does not have a Unix account on the remote storage system. The answer is that the data grid stores all files under its own Unix Identifier, which is assigned to the data grid at each site by the system administrator. The data grid manages access controls for each file, independently of the storage systems. When a request is made to manipulate a file, the data grid authenticates the user, checks the data grid access controls to verify the user has permission to do the requested operation, authenticates the data grid to the remote storage system under the data grid Unix Identifier, and then performs the desired operation.

This sequence of operations is possible if the data grid manages the logical file name space, the logical user name space, and the access controls independently of the storage systems. The side benefits are many, including the following:

- The physical location of the digital entity can be automatically updated after grid operations since all accesses are through the data grid.
- Access controls are automatically preserved when the file is moved. The data grid manages access controls as constraints between the logical user name space and the logical file name space. Hence the access controls are independent of the actual storage location.
- The administrative burden on implementing the data grid is minimized because only a single Unix Identifier is required at each storage system.
- Authenticity and integrity properties remain associated with the digital entity across the data grid operations since they are managed as attributes of the logical name space.
- Integrity properties (such as audit trails) can be automatically updated as the file is moved because all accesses are done through the data grid software.

The ability to associate properties with the logical file names that are managed consistently by the data grid is essential for both authenticity and integrity preservation. The authenticity properties are treated as provenance metadata that is mapped onto each logical file name. They are not modified and are automatically associated with each replica of the digital entity. The integrity properties are automatically updated whenever the digital entity is moved and can be checked on demand. Examples include the following:

- Audit trails: the date and requesting person can be logged for all operations done on the digital entity; this makes it possible to track the chain of custody over time.
- Checksums and digital signatures: each digital entity can be analyzed for internal corruption by recreating the checksum and comparing it with the checksum metadata value.
- Annotations: archivist comments can be associated with each digital entity to track changes in policy.
- Access roles: the privileges that archivists may exercise are encapsulated as roles that allow addition of new records, update of metadata, creation of annotations, and use of audit trails.
- Versions: material that changes over time can be managed as versions of the original digital entity; this is useful for Web sites that hold material that has a limited lifetime.

The above examples all rely upon the ability of the data grid to manage properties that can be organized as metadata in a database. For infrastructure independence, the archivist also needs to be able to migrate from old database technology to new database technology. This can be accomplished if the digital entities that are being preserved are organized as a collection that is implemented as a catalog in a database. The data grid manipulates the collection and maps from operations on the collection to the operations that the database can perform. This is shown in figure 2.

Instead of interacting directly with a database, the archivist issues requests to the data grid, which interacts with the database on the behalf of the archivist (Rajasekar & Moore, 2001). This makes it possible to implement operations on collections such as schema extension, automated SQL generation, bulk metadata import and export, and management of metadata in XML and HTML files.

Manipulation of a collection that is housed in a database requires the ability to move the collection onto new database technology. The combination of digital entity and catalog infrastructure independence can be thought of as the ability to encapsulate a preservation environment and migrate it onto new choices of storage and database technology. The collection is preserved, not just the digital entities that comprise the content of the collection.

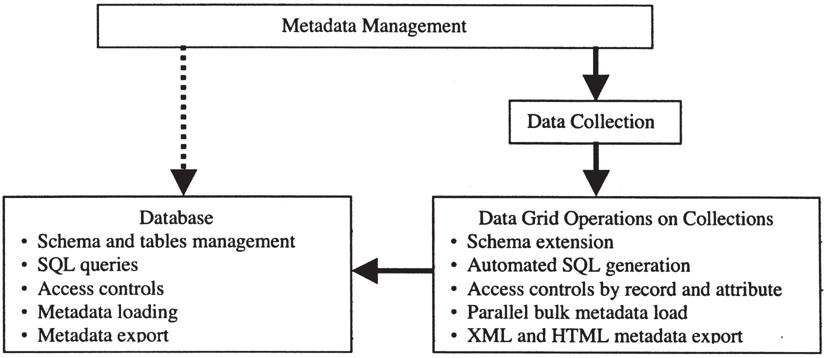
Metadata Management
Data Collection
Database
• Schema and tables management
• SQL queries
• Access controls
• Metadata loading
• Metadata export
Data Grid Operations on Collections
• Schema extension
• Automated SQL generation
• Access controls by record and attribute
• Parallel bulk metadata load
• XML and HTML metadata export

Figure 2. Collection Management in a Data Grid

Preservation Environment Infrastructure

The infrastructure that is used to implement a preservation environment manages the migration of digital entities and catalogs onto new technology. At the point in time when a migration is going to be performed, the preservation environment needs to be able to interact with both the old and new systems. This is precisely the set of interoperability mechanisms that are provided by data grids for dealing with heterogeneous storage systems. The sharing of data across spatially distributed heterogeneous storage systems requires the same type of interoperability mechanisms as needed to support migration of collections onto new systems over time.

The data grid infrastructure components are shown in figure 3. Five levels of software infrastructure are used to simplify the integration of new technology. The lowest level consists of the vendor supplied storage systems and databases. These systems are typically chosen as the most cost-effective storage that meets the preservation integrity requirements for reliability and robustness. Different types of storage systems are used to meet each of the preservation requirements:

- File systems: used to support interactive access to archived material. Commodity-based disk storage systems that are multiple terabytes in size currently cost (in 2004) about $650 per terabyte per year (Rajasekar et al., 2003). The cost includes capital equipment amortization, software and hardware maintenance, and operations labor support. The equipment is assumed to have a four-year lifetime, after which it will be replaced. Access time to data on file systems is measured in tenths of a second.
- Tape systems and archival storage systems: used to support minimal cost, long-term storage of the digital entities. Current cost estimates for tape-based systems that are petabytes in size (thousands of terabytes) are about $300 per terabyte per year. The cost includes the amortized

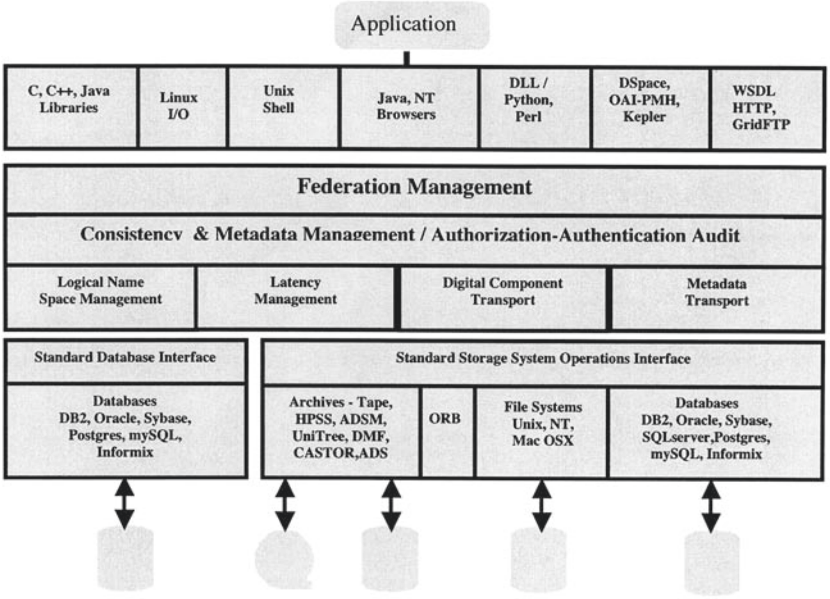
Application
C, C++, Java Libraries
Linux I/O
Unix Shell
Java, NT Browsers
DLL / Python, Perl
DSpace, OAI-PMH, Kepler
WSDL HTTP, GridFTP
Federation Management
Consistency & Metadata Management / Authorization-Authentication Audit
Logical Name Space Management
Latency Management
Digital Component Transport
Metadata Transport
Standard Database Interface
Databases DB2, Oracle, Sybase, Postgres, mySQL, Informix
Standard Storage System Operations Interface
Archives - Tape, HPSS, ADSM, UniTree, DMF, CASTOR,ADS
ORB
File Systems Unix, NT, Mac OSX
Databases DB2, Oracle, Sybase, SQLserver,Postgres, mySQL, Informix

Figure 3. Data Grid Infrastructure

media cost, the capital equipment amortization, software and hardware maintenance, and operations labor support. Both the media and equipment are assumed to have a four-year lifetime. Access time to files on tapes located in tape robots is measured in minutes.

Other types of storage systems include Object Ring Buffers (ORBs), which are used to store real-time sensor data; Storage Resource Managers, which are used to manage the transaction load on archives and file systems; Storage Area Networks, which are used to support file systems for multiple computers; and object relational databases, which store digital entities as binary large objects (blobs).

The second level of the data grid infrastructure is the standard storage system operations interface and the standard database interface. Data grids provide a standard set of operations that can be performed upon digital entities in any of the storage systems. The standard operations include single file manipulation commands such as read and write, as well as bulk operations for manipulating a large number of digital entities simultaneously. The data grid maps from the standard operations to the operations that can be performed on a particular type of storage system. A separate storage system driver is written for each type of storage system. The result is the ability to apply the standard operations to files in archival storage systems, files in file systems (whether Unix or Windows), blobs in databases, objects in ORBs, etc. Preservation environments that are built on top of data grid technology have the ability to store the archived material on all types of storage systems.

The third level of the data grid architecture manages the consistency between the preservation metadata and the archived digital entities. This level tracks the location of the digital entities when they are moved, checks for completion of commands, manages the authentication and authorization of users, and supports the operations required to manage latency. Latency is the extra time that it takes to initiate interactions with a database or initiate data transport over a wide-area network. When a large number of digital entities are going to be manipulated, the total time that it will take can be greatly decreased if the latency overhead is incurred once for the entire set of digital entities, instead of being incurred for each digital entity. This can be accomplished through the use of bulk operations. Digital entities can be aggregated into a single physical container before transport. Metadata can be aggregated into an XML file before it is bulk loaded into a database. Multiple remote Input/Output (I/O) operations can be combined into a single call to a procedure that is executed at the remote storage system through a single command request. Data grids provide the latency management functions that enable data, metadata, and I/O command aggregation.

Another challenge that must be handled by a preservation environment is a "respect for storage." Storage systems are designed to handle a certain number of files, a certain transaction rate, and a certain data transport rate. If the preservation environment exceeds these capability limits, the performance degrades significantly and the storage system may even fail under the load and stop working. Data grids manage the storage system file name space limitation by aggregating digital entities into a physical container before storage. The data grid maintains the starting location of the digital entity as a file property in the data grid metadata catalog. The storage system only sees names for the containers, while the data grid maintains properties for each digital entity. An example is the preservation of material published on a Web site. The typical size of a digital entity retrieved from the Web is about 100 kBytes. A typical data grid container size is about 300 megabytes. Thus, 3,000 items retrieved from the Web are aggregated into a single container before storage. This means a Web crawl that retrieves 3 million digital entities is stored as only one thousand containers. Most storage systems are able to handle about 20 million files before their performance degrades.

The fourth level of the data grid architecture is a standard set of operations that can be invoked by an archivist or user of the preservation environment. The standard set of operations includes manipulation of digital entities, retrieval of digital entities, queries on descriptive metadata, manipulation of metadata, retrieval of metadata, etc. The operations are implemented in an Application Protocol Interface (API). The fundamental APIs are C library calls for access to the preservation environment from an application, shell commands for interactively invoking operations from a computer, and a Java class library for accessing the preservation environment from a Java applet. The function of the data grid can be viewed as the mapping of the standard access operations onto the operations supported by vendor-supplied storage systems and databases. Note that two levels of abstraction are used to accomplish this mapping: (1) access standard operation characterization, and (2) storage repository standard operation characterization. The data grid maps between two standard sets of operations. This makes it possible for the data grid consistency management (third level of the infrastructure) to be designed independently of the choice of access API and storage system.

The fifth level of the data grid architecture is the set of access mechanisms that are preferred by the archivist and user communities. The four sets of APIs listed on the right side in the top row (see figure 3) of the architecture are ported to the data grid through one of the three standard interfaces: C library, shell command, or Java. The APIs that can be used include the following:

- DSpace digital library and preservation system developed at the Massachusetts Institute of Technology.[2] DSpace provides standard services for importing a collection into a digital library. The processing steps include creation of preservation metadata, validation of the ingested material, and generation of an Archival Information Package (AIP)[3] using the Metadata Encoding and Transmission Standard (METS)[4] to encapsulate the authenticity metadata. The AIP integrates the digital entity with its authenticity metadata for storage as a single package. The DSpace integration with the data grid is done through a Java class library.
- Open Archives Initiative–Protocol for Metadata Harvesting (OAI-PMH).[5] This provides a standard transport mechanism for uploading metadata from the data grid or preservation environment into a central repository. The OAI-PMH interface was implemented using shell commands.
- Web browser interface for HTTP access. This provides the ability to interact with the preservation environment through a vendor supplied Web browser such as Microsoft Internet Explorer, Mac Safari, Netscape Navigator, etc. The Web browser interface was implemented using both shell commands and the C library interface.
- Kepler workflow processing interface.[6] Kepler is a workflow system that can be used to automate application of standard procedures on a collection. Kepler is based on the Ptolemy workflow environment developed at the University of California, Berkeley (Brooks et al., 2004). Kepler implements actors that can read and write files from the data grid. The actors are based on the Java class library.

- Windows browser. The iNQ windows browser provides a Windows file system style presentation to the digital entities within the data grid. The iNQ browser supports processing of query result sets, as well as reading and writing of files and drag and drop of files from a desktop environment. The iNQ browser only runs on Windows platforms and is based on a C++ object layer on the C library calls.
- Web Services Description Language (WSDL).[7] WSDL is used to implement Web services for interaction with a data grid. The Web services are designed to support file manipulation, metadata manipulation, file discovery, and file and metadata retrieval. The WSDL interface is based on the Java class library.
- Perl/Python/Windows load libraries. These interfaces are used with standard scripting languages to read and write files located in a data grid. The load libraries are based on the Unix shell command interface and the C library interface to the data grid.
- GridFTP.[8] A standard transport mechanism that is used in grids for moving files is based on the FTP protocol. The GridFTP interface augments FTP with support for Grid Security Infrastructure, partial file reads and writes, and parallel I/O. The interface is based on the C library call interface to the data grid.

Thus, the preservation environment has the ability to add new user access mechanisms as well as the ability to add new types of storage systems and database technology. Data grid infrastructure independence makes it possible for a new user access protocol to be used with a legacy storage system that was acquired before the new access protocol was available. In practice, this is one of the main uses of data grid technology—the application of new access methods to old storage technology. Data grids are also used to integrate "stove pipe" storage technologies that did not share a common access protocol. The data grid technology makes it possible to integrate digital entities residing within each "stove pipe" into a common data collection without having to modify either of the legacy systems.

Generic Preservation Environments

The technology that is used to provide infrastructure independence for preservation environments is equally applicable to other types of data management systems. A preservation environment that is based on data grid technology incorporates capabilities that are also useful for real-time sensor data archiving, collection building environments, data sharing environments, digital libraries, and data analysis environments (Moore & Baru, 2003). The common capabilities include the following:

- Management of distributed data; copies can be made and stored at multiple sites on multiple types of storage systems

- Association of metadata attributes or properties with each digital entity; the properties can include authenticity-, integrity-, descriptive-, and user-defined attributes
- Support for arbitrary types of storage systems
- Support for multiple vendor and nonproprietary databases
- Support for multiple types of access interfaces
- Management of the consistency between the digital entities and the digital entity properties
- Management of access latency for databases and wide area networks

The differences between a digital library, a data sharing environment, and a preservation environment can be supported through different choices for digital entity properties, for access mechanisms, and for consistency controls. An example of the ubiquity of application of data grid technology is the Storage Resource Broker (SRB) developed at the SDSC (Baru, Moore, Rajasekar, & Wan, 1998). The SRB provides all of the capabilities that have been discussed and is used in a production environment to support collections for federal agencies, including research projects for the National Science Foundation (NSF), data collections for the National Aeronautics and Space Administration (NASA), data grids for the Department of Energy, data grids for the National Institutes of Health (NIH), preservation environments for the University of California, and preservation environments for NARA and the National Historical Publications and Records Commission (NHPRC). The total amount of data stored at SDSC for these projects is over 330 terabytes and over 50 million files. Table 1 lists the amount of data and number of persons with access privileges for each collection.

The SRB data grid technology is scalable. The collection sizes range from a hundred gigabytes to a hundred terabytes in size. The number of files in a collection range from a few thousand to over 26 million. The number of users for which access privileges are kept range from a few tens to over 3,000. Each of the collections is distributed across multiple storage systems. The NSF National Partnership for Advanced Computational Infrastructure collection is housed on over 85 storage systems. The NIH Biomedical Informatics Research Network manages data that is distributed across 17 sites from the West Coast to the East Coast of the United States.

Preservation Environment Examples

The NARA research prototype permanent archive and the NHPRC Persistent Archive Testbed illustrate two different approaches to the creation of a preservation environment. Both projects use the SRB data grid technology but provide different management strategies for assuring integrity. Based on production storage experiences at the SDSC, all digital data is at risk of being lost through the factors listed in table 2.

Table 1. Data Collections Stored at SDSC Using the Storage Resource Broker

(SRB) Data Grids at SDSC (as of 9/27/2004)	GBs of Data Stored	Number of Files	Number of Users
Data Sharing Environments			
NSF/ITR—National Virtual Observatory[a]	53,778	9,507,399	80
NSF—National Partnership for Advanced Computational Infrastructure[b]	22,165	5,156,765	380
Hayden Planetarium—visualizations of the evolution of the solar system	7,201	113,600	178
NSF/NPACI—Joint Center for Structural Genomics[c]	5,228	652,031	50
NSF/NPACI—Biology and Environmental collections	8,704	21,881	67
NSF—TeraGrid, ENZO Cosmology simulations	104,370	908,600	3,247
NIH—Biomedical Informatics Research Network[d]	5,808	3,777886	172
Data Publication Environments			
NLM—Digital Embryo image collection[e]	720	45,365	23
NSF/NPACI—Long-Term Ecological Reserve	251	8,381	36
NSF/NPACI—Grid Portal	1,917	49,665	392
NIH—Alliance for Cell Signaling microarray data[f]	776	60,177	21
NSF—National Science Digital Library SIO Explorer collection[g]	2,122	758,233	27
NSF/NPACI—Transana education research video collection[h]	92	2,387	26
NSF/ITR—Southern California Earthquake Center[i]	88,199	1,790,319	59
Records Preservation Environments			
UCSD Libraries—image collection	128	203,930	29
NARA—Research Prototype Permanent Archives[j]	89	254,470	58
NSF—National Science Digital Library permanent archives[k]	3,571	26,908,350	122
Total	305 TB	50 million	4,967

[a] National Virtual Observatory (NVO), http://www.us-vo.org/.

[b] National Partnership for Advanced Computational Infrastructure Data Intensive Computing Environment (NPACI) thrust area, http://www.npaci.edu/DICE/.

[c] Joint Center for Structural Genomics (JCSG), http://www.jcsg.org/.

[d] Biomedical Informatics Research Network, http://nbirn.net/.

[e] Visible Embryo Project, "Human Embryology Digital Library and Collaboratory Support Tools," part of the Next Generation Internet Initiative and funded by the National Library of Medicine, http://netlab.gmu.edu/visembryo.htm.

[f] Alliance for Cell Signaling (AfCS), http://www.signaling-gateway.org/.

[g] SIO Explorer Digital Library Project to provide educational material from oceanographic voyages in collaboration with the National Science Digital Library (NSDL), http://ndsl.sdsc.edu/.

[h] Transana—education research tool for the transcription and qualitative analysis of audio and video data, http://www.transana.org/.

[i] Southern California Earthquake Center, http://www.scec.org/.

[j] NARA Persistent Archives project, http://www.sdsc.edu/NARA/.

[k] National Science Digital Library, http://www/nsdl.org/.

Table 2. Types of Risk and Risk Mitigation Mechanisms at the SDSC

Entity at Risk	Size	Problem	Frequency	Minimum Number of Replicas Needed to Mitigate Risk
File	~2 MB	Corrupted media, disk failure	1 year	2 copies in single system
Tape	~200 GB	The above plus simultaneous failure of 2 copies	5 years	3 copies in homogeneous systems
System	~10 TB	The above plus systemic errors in vendor software, or malicious user, or operator error that deletes multiple copies	1–15 years	3 independent, heterogeneous systems
Archive	~1 PB	The above plus natural disaster, obsolescence of standards	10–100 years	3 distributed, heterogeneous systems

Every digital entity relies upon a software and hardware infrastructure for its long-term preservation. The infrastructure can be compromised by hardware failures such as media corruption for tapes and disk failure for file systems. The simple way to protect against this is to replicate the digital entity onto a physically separate set of media. For commodity-based disk systems, SDSC sees a disk failure about every 80 disk-months of use. For tape media, SDSC finds that the media lifetime exceeds five years but that individual files on tape may be lost due to tape robot drive malfunction. SDSC migrates to new tape media about every three years to minimize the total cost of storage, decrease the number of tape cartridges that must be managed, and recover floor space.

At longer time intervals, on the order of five years, having multiple copies is insufficient. Infrequent simultaneous failure of the original copy and the backup copy can occur. This can be a combination of unexpected failures, such as the replication procedure failing because of an operational error and the original copy being lost because of media failure. A third copy is required on an independent storage system.

The source of integrity risk also depends upon the technology provided by the software and hardware vendor. An example is the release of the Pentium processor for commodity use and the discovery after use for a year that the processor occasionally generated bad results. To protect against vendor manufacturing problems, storage systems from multiple vendors should be used. A similar risk arises from operator error, in which operational proce-

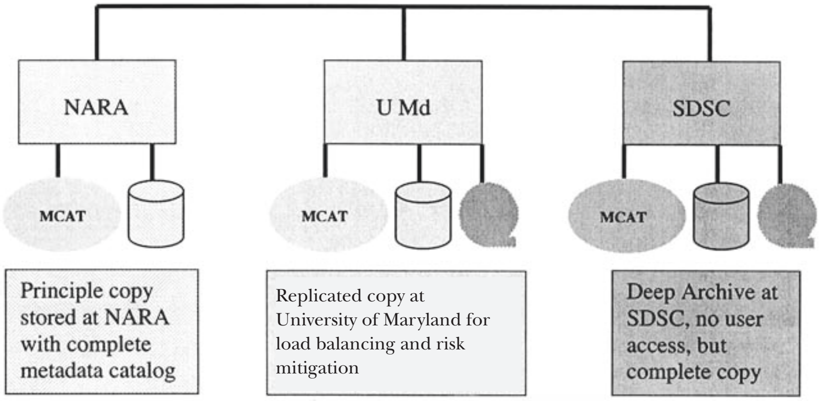
NARA
U Md
SDSC
MCAT
MCAT
MCAT
Principle copy stored at NARA with complete metadata catalog
Replicated copy at University of Maryland for load balancing and risk mitigation
Deep Archive at SDSC, no user access, but complete copy

Figure 4. NARA Prototype Persistent Archive Federation

dures are applied with unexpected consequences. Unintended overlap of operational procedures can lead to interference between storage systems that causes data loss. The solution is to have one of the copies under the operational management of an independent site.

Every storage center is at risk of data loss from a natural disaster such as an earthquake, flood, or hurricane. One of the copies needs to be located at a geographically remote site. Every storage center is also at risk due to acts by malicious users. In this case, a copy needs to be made to a site that restricts user access and that requires independent archivist actions to manipulate the preservation environment. Such a site is characterized as a "deep archive," in which material is written once, no overwrites are possible, user access is limited to archivists, and material is staged through a preservation workbench. A deep archive is differentiated from a "dark archive" through the ability to federate the deep archive with active archives. A deep archive is a component of a larger preservation environment that supports active access to electronic records.

The NARA research prototype persistent archive implements a preservation environment that is designed to mitigate against all of these types of risk. The system is described in figure 4.

The preservation environment is implemented as the federation of three independent data grids. Each data grid manages its own preservation metadata in a separate metadata catalog. Each data grid manages its own set of storage systems. Consistency constraints are implemented between the data grids to control which digital entities may be replicated between the data grids, how the preservation metadata is synchronized between the data grids, how user identifiers are replicated between the data grids, and how resources are shared between the data grids. The consistency controls are specified on each of the five name spaces identified in figure 1.

The types of integrity risk are now managed by a combination of the following replicas:

- Multiple copies are kept at University of Maryland (U Md). U Md uses a High Performance Storage System (HPSS) to replicate files that are provided for public access on a commodity-based disk storage system. This mitigates against media loss.
- U Md replicates data onto a commodity disk system at NARA. This protects against operational error at U Md and protects against simultaneous loss of the two copies at U Md. The U Md and NARA metadata catalogs are implemented in different database technologies (Informix and Oracle) to protect against systemic vendor product failure.
- A copy is replicated into a deep archive at SDSC. This protects against natural disaster (tornados), and also provides a copy that has restricted access to protect against malicious users.

The combination of the three sites makes it possible to mitigate against the multiple sources of risk. The types of collections that are housed in the preservation environment include email collections, Web crawls, image collections, office product collections, Graphical Information System (GIS) systems, state department communiqués, binary data, etc. The types of preservation metadata that are maintained about each digital entity are governed by the NARA Life Cycle Data Requirements Guide. The electronic records are organized by collection, record group, record series, file unit, and file entity. The preservation metadata is organized in a preservation description catalog and mapped onto the global name space that is provided for each digital entity by the SRB data grid technology. The single largest collection is the Electronic Access Project (EAP) image collection, with over 350,000 files and over 1 terabyte of data.

The NHPRC Persistent Archive Testbed (PAT)[9] links archives across multiple state institutions. A single data grid is used to link the sites with the metadata catalog maintained at the SDSC. Each site uses a 1–2 terabyte commodity disk system to house a local copy of its preservation collection. A replica is also kept on tape at SDSC. Each site preserves a different type of material and investigates how to automate the archival procedures that are used for its collection. The sites and types of collections are

- Kentucky Department for Libraries and Archives (Web collection)
- Minnesota Historical Society (spatial data records on land use)
- Ohio Historical Society (email collection)
- Stanford Linear Accelerator Center Archives and History Office (high-energy physics project)

- Michigan Department of History, Arts, and Libraries (records stored in a records management application).

The focus of the PAT project is on automation of archival processes of appraisal, accession, arrangement, description, preservation, and access. While all of the collaborating sites are examining how description can be automated, each partner has selected a different set of preservation processes to automate. The approach followed in the PAT collaboration is to first put the material to be archived under archivist control by loading it onto the data grid, then developing processing scripts that allow the organization and description to be characterized. Scripts are developed to extract the preservation metadata and to organize the digital entities. The scripts are then applied to create a new collection within the data grid that provides the appropriate structure and metadata. The digital entities are then replicated onto a tape archive at SDSC using container technology.

Summary

The development of a preservation environment is strongly driven by the desire to support infrastructure independence, the ability to preserve digital entities as a collection, and the ability to migrate the collection to new choices for storage and database technology. Data grid technology provides this capability and has been shown to scale to the size of digital holdings that are now being considered for preservation. The NARA research prototype persistent archive and NHPRC Persistent Archive Testbed illustrate two different approaches to the implementation of a preservation environment. Both models are useful and provide different ways to federate independent collections into a sustainable preservation system.

Acknowledgements

The results presented here were supported by the InterPARES project, NSF NPACI ACI-9619020 (NARA supplement), 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, and the NASA Information Power Grid. The collaborators on the NARA prototype persistent archive are Dr. Joseph JaJa (University of Maryland) and William Underwood (Georgia Tech). The collaborators on the NHPRC PAT project are Charles E. Arp (Ohio Historical Society), Jean Deken (Stanford Linear Accelerator), Robert Horton (Minnesota Historical Society), Glen McAninch (Kentucky Department for Libraries and Archives), and Caryn Wojcik (State Archives of Michigan). 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, or

the final findings of InterPARES, which is still in the testing phase. More information about the example permanent archive based on the SDSC Storage Resource Broker can be found at http://www.npaci.edu/DICE/SRB/index.html and http://www.sdsc.edu/NARA/.

Notes

[1] Definitions are given for *authenticity* and *integrity* in the final report from InterPARES 1 (2002).

[2] For information on DSpace, see http://www.dspace.org/, and the article by MacKenzie Smith in this issue of *Library Trends.*

[3] See Reference Model for an OAIS (2002) for information on AIP.

[4] For information on METS, see http://www.loc.gov/standards/mets/.

[5] For information on OAI-PMH, see http://www.openarchives.org/OAI/openarchivesprotocol.html.

[6] For information on the Kepler Project, a System for Scientific Workflows, see http://kepler.ecoinformatics.org/.

[7] For more information on WSDL, see http://www.w3.org/TR/wsdl.

[8] For more information on GridFTP, see http://www.globus.org/datagrid/gridftp.html.

[9] For more information on the NHPRC Persistent Archive Testbed, see http://www.sdsc.edu/PAT/.

References

Baru, C., Moore, R., Rajasekar, A., & Wan, M. (1998). The SDSC storage resource broker. In S. A. MacKay (Ed)., *Proceedings of CASCON '98 Conference, Nov. 30–Dec. 3, 1998, Toronto, Canada* (p. 5). Toronto, Ont.: IMB Canada.

Brooks, C., Lee, E. A., Liu, X., Neuendorffer, S., Zhao, Y., & Zheng, H. (Eds.). (2004). *Heterogeneous concurrent modeling and design in Java (Volume 1: Introduction to Ptolemy II). Technical Memorandum UCB/ERL M04/27, University of California, Berkeley.*

Foster, I., & Kesselman, C. (1999). Data intensive computing. In I. Foster & C. Kesselman, *The grid: Blueprint for a new computing infrastructure* (pp. 105–31). San Francisco, CA: Morgan Kaufmann.

InterPARES 1. (2002). *The Long-term Preservation of Authentic Electronic Records: Findings of the InterPARES Project.* Retrieved January 19, 2005, from http://www.interpares.org/ip1/ip1_index.cfm.

Moore, R., & Baru, C., (2003). Virtualization services for data grids. In F. Berman, G. Fox, & T. Hey (Eds.), *Grid computing: Making the global infrastructure a reality* (pp. 409–36). New York: Wiley.

Moore, R., Baru, C., Rajasekar, A., Ludascher, B., Marciano, R., & Wan, M., et al. (2000). Collection-based persistent digital archives—Parts 1& 2. *D-Lib Magazine, 6*(3). Retrieved January 19, 2005, from http://www.dlib.org/dlib/march00/moore/03moore-pt1.html and http://www.dlib.org/dlib/april00/moore/04moore-pt2.html.

Rajasekar, A., Marciano, R., & Moore, R. (1999). Collection based persistent archives. In *Proceedings of the 16th Annual IEEE Symposium on Mass Storage Systems, March 15–18, 1999, San Diego, CA* (pp. 176–84). Los Alamitos, CA: IEEE Computer Society Press.

Rajasekar, A., & Moore, R. (2001). Data and metadata collections for scientific applications. In *High Performance Computing and Networking: 9th International Conference, HPCN Europe, Amsterdam, the Netherlands, June 25–27, 2001* (pp. 72–80). New York: Springer.

Rajasekar, A., Wan, M., Moore, R., Kremenek, G., & Guptil, T. (2003). Data grids, collections, and grid bricks. In *Proceedings of the 20th IEEE Symposium/11th NASA Goddard Conference on Mass Storage Systems and Technologies, April 7–10, 2003, San Diego, CA* (pp. 2–9). Los Alamitos, CA: IEEE Computer Society Press.

Reference Model for an Open Archival Information System (OAIS). 2002. *Blue Book, Issue 1* [Adopted as ISO 14721:2003]. Retrieved January 19, 2005, from http://ssdoo.gsfc.nasa.gov/nost/isoas/ref_model.html.

Reagan W. Moore, Director, Data Intensive Computing Environments, San Diego Supercomputer Center, 9500 Gilman Drive, MC-0505, La Jolla, CA 92093-0505, moore@sdsc.edu, and Richard Marciano, Director, Sustainable Archives and Library Technology Laboratory, San Diego Supercomputer Center, 9500 Gilman Drive, MC-0505, La Jolla, CA 92093-0505, marciano@sdsc.edu. Dr. Reagan W. Moore is Director for Data and Knowledge Systems at the San Diego Supercomputer Center. He coordinates research efforts on digital libraries, data grids, and persistent archives for thirteen research grants ranging from the NSF National Virtual Observatory, to the NSF National Science Digital Library persistent archive, to the DOE Particle Physics Data Grid and the NARA Research Prototype Persistent Archive. Moore has a Ph.D. in Plasma Physics from the University of California, San Diego (1978) and a B.S. in Physics from the California Institute of Technology (1967).

Richard Marciano is Director of the Sustainable Archives and Library Technologies (SALT) Laboratory and Lead Scientist in the Data and Knowledge Systems (DAKS) Group at the San Diego Supercomputer Center (SDSC), at the University of California San Diego (UCSD). He is also an affiliate professor in the Urban Studies and Planning Program in the Division of Social Sciences and founding member of the Regional Workbench Consortium (RWBC) at UCSD. The SALT Lab is an interdisciplinary unit focused on developing information technology strategies and conducting research in the area of digital materials and records collection and preservation. Dr. Marciano's interests are in data management, digital archiving, and long-term preservation. Current research projects include InterPARES 2, Persistent Archives Testbed (PAT), Preservation of Electronic Records in an RMA (PERM), and Incorporating Change Management in Archival Processes (ICAP). He holds degrees in Avionics and Electrical Engineering (National School of Civil Aviation, Toulouse, France) and an M.S. and Ph.D. in Computer Science from the University of Iowa; he also worked as a postdoc in computational geography.