# Chronopolis: Preserving our Digital Heritage

**David Minor, Don Sutton**
UC San Diego, San Diego Supercomputer Center
minor, dsutton @sdsc.edu

**Ardys Kozbial**
UC San Diego Libraries
akozbial@ucsd.edu

**Michael Burek**
National Center for Atmospheric Research
mburek@ucar.edu

**Michael Smorul**
University of Maryland Institute for Advanced Computer Studies
toaster@umiacs.umd.edu

**Abstract**

The Chronopolis Digital Preservation Initiative, one of the Library of Congress' latest efforts to collect and preserve at-risk digital information, has completed its first year of service as a multi-member partnership to meet the archival needs of a wide range of cultural and social domains. In this paper we will explore the major themes within Chronopolis.

## Chronopolis

The Chronopolis digital preservation network, initially funded by the Library of Congress' National Digital Information Infrastructure and Preservation Program (NDIIPP), has completed its first year of service as a multi-member partnership to meet the archival needs of a wide range of domains.

Chronopolis is a digital preservation data grid framework developed by the San Diego Supercomputer Center (SDSC) at UC San Diego, the UC San Diego Libraries (UCSDL), and their partners at the National Center for Atmospheric Research (NCAR) in Boulder, Colorado and the University of Maryland's Institute for Advanced Computer Studies (UMIACS).

A key goal of the Chronopolis project is to provide cross-domain collection sharing for long-term preservation. Using existing high-speed educational and research networks and mass-scale storage infrastructure investments, the partnership is designed to leverage the data storage capabilities at SDSC, NCAR, and UMIACS to provide a preservation data grid that emphasizes heterogeneous and highly redundant data storage systems.

Specifically, each Chronopolis partner operates a grid node containing at least 50 TB of storage capacity for digital collections related to the Library of Congress' NDIIPP content. The Chronopolis methodology employs a minimum of three geographically distributed copies of the data collections, while enabling curatorial audit reporting and access for preservation clients. The partnership is also developing best practices for the NDIIPP community for data packaging and transmission among heterogeneous digital archive systems.

As of July, 2009, Chronopolis houses four diverse collections: a backup of the complete digital holdings of the Inter-university Consortium for Political and Social Research (ICPSR), based at the University of Michigan, "Web-at-Risk" collections from the California Digital Library (CDL), geospatial data resources from the North Carolina Geospatial Data Archiving Project, and several decades of data from research cruises from the Scripps Institution of Oceanography (SIO) at UC San Diego.

### The Chronopolis Model

The key concept underlying Chronopolis is a phased approach to the development of long-term preservation cyberinfrastructure that can be scaled and evolved over time. Such an approach must provide:

- A production system for collection management and preservation that is stable, can evolve with use and

technology, and scale with expansion of individual and aggregate collections.
• Smooth integration of new technologies as they are developed and tested, in order to increase capability and functionality without service disruption.
• Well-managed administration of the facility which includes the integration of policies and procedures governing the availability of data, data integrity, security, retention periods, collection selection, and metadata standards.
• The exploration of policies and cost models for long-term preservation that ensure the protection of critical data collections beyond the life-time of the projects and efforts which generated them, and provide a plan for future maintenance, curation and use.

The Chronopolis model seeks to integrate these elements to provide a model for the data management and preservation cyberinfrastructure that will be required to ensure availability, access, and usability of our most valued digital data holdings.

## Chronopolis Services

Chronopolis provides a suite of replication and preservation services. These services are the mechanics of the digital lifecycle for objects in Chronopolis, from ingest and replication to monitoring and managing.

### Data Ingest

The Chronopolis ingest process consists of several steps, including negotiation with data providers, data transfer, registration into Chronopolis, and quality assurance/quality control (QAQC) at various stages. The initial process starts with human negotiations between Chronopolis and the data provider personnel. During this discussion issues such as the number of collections, their sizes, naming of collections and transfer methods are discussed. The packaging and transfer process thus far has varied somewhat according to the data provider, however for the most part BagIt has been used during data collection transfer to SDSC storage devices. Starting with a BagIt filename that is accessible from the data provider, the collection must be retrieved, usually via *ssh* or the *wget* transfer protocol. Once onto an SDSC storage device a QAQC process is run against the authoritative collection manifest file originating from the data provider. This process includes checking inventory and individual data object checksums. The collection is transferred to a storage device that is a registered SRB resource so that the actual SRB ingest is merely a registration of the collection into the SRB MCAT. Once ingested, read permissions are granted to the NCAR and UMIACS SRB zones so that the replication process can proceed. The final step includes registration of the collection into the ACE monitoring system.

### Data Replication

Replication from SDSC to UMIACS and NCAR is monitored using the SRB Replication Monitor installed at UMIACS. Prior to replication, each replica site designates an account local to its zone that will host the data. Using a local account ensures that accessing a partner's data is not dependent on any services running at a remote partner site. After this account has been established, the master site grants read-only access to this new account. This account will be used to pull data from the master site to remote peers.

The collection is then registered to the SRB Replication Monitor. After all data has been ingested and access permissions set on the master site, replica synchronization is started on the Replication Monitor. This synchronization will compare the files in the master collection with each registered partner. Any data that is different or non-existent on a partner site will be copied to the remote site. The resulting replica at the peer sites will be under the custody of the local SRB account on that partner.

As new data is added to collections on the master site, replication may be triggered multiple times to ensure that all partner sites have copies of the complete collection. The Replication Monitor only pulls data from the master site to partner sites. Any data that exists on partner sites that does not exist on the master site is not removed: a manual deletion is required. During the ingestion of collections into Chronopolis only one situation was encountered where the manual removal of files from a partner site was necessary.

### Data Auditing

An Audit Control Environment Audit Manager (AM)[i] has been installed at all three partner sites to monitor the integrity of replicated files. The three partner sites administer their ACE installations independent of other sites. After replication to a partner site finishes, that site registers the new collection into ACE for monitoring. During registration, collections are grouped by data provider, an audit policy is assigned to them, and connection information for the SRB is gathered. Each collection is assigned a unique audit policy determining when the AM will scan collections for changes. The current default policy in Chronopolis is to audit collections every 30 days.

During the initial audit, SHA-256 digests are registered for all files in the collection. These digests are secured as described previously and used to validate the contents of a collection during subsequent audits. After the collection has been registered, auditing will occur as dictated by the collection's policy or manually as triggered by an administrator. After each audit a report is generated summarizing what activity occurred during an audit. These may optionally be delivered via e-mail.

After a collection has been fully registered, an administrator may compare the collection to either a supplied manifest or to a peer site. After replication,

partner sites that recently received data should compare the new collections to the master collection. This will detect any files that may have not been properly replicated. In Chronopolis, both partners at UMIACS and NCAR will compare their collections against SDSC to ensure they have been replicated properly.

Performance testing of the AM installation at UMIACS has shown that the entire Chronopolis holdings can be audited in under one week.[ii] The table below shows the audit performance grouped by data provider.

| Installation | Files | Directories | Size | Time(h) |
|---|---|---|---|---|
| CDL | 46,762 | 28 | 4.291 TB | 20:32 |
| SIO-GDC | 197,718 | 5,230 | 815 GB | 6:49 |
| ICPSR | 4,830,625 | 95,580 | 6.957 TB | 122:48 |
| NC-State | 608,424 | 42,207 | 5.465 TB | 32:14 |

## Metadata Services

A Chronopolis Metadata Working Group developed a metadata model for Chronopolis' first phase services to meet the following requirements:

- replicate assets in multiple and geographically dispersed locations,
- monitor assets regularly to identify deterioration or corruption,
- develop mechanisms for replacing deteriorating or corrupt assets,
- deliver assets back to the Data Provider upon request.

This model must also:

- be conformant to community metadata standards,
- be extensible to support future development of Chronopolis services and community metadata standards,
- promote trust among data providers for Chronopolis.

In completing the Chronopolis metadata model, the Metadata Working Group made an analysis of the Chronopolis system, determining what metadata are created and used and how they are created at certain points in the Chronopolis life-cycle of a digital asset. Discussions were founded on two basic assumptions:

- Data providers need to be highly confident that the assets they submit to the system can be retrieved.
- Metadata is the foundation of that confidence and allows Chronopolis management to know that digital assets are the same as what was submitted or to identify those that are not and "cure" them via the replication technology utilized in the system.

The working group posited the life-cycle path of a digital asset in the Chronopolis system, noting eight types of events that the life-cycle triggered. These event types do not necessarily occur in a linear sequence.

- ET-1. Service Level Agreement
- ET-2. Acquisition Transfer
- ET-3. Acquisition Validation
- ET-4. Acquisition Registration into the SRB
- ET-5. Acquisition Registration into ACE
- ET-6. Inter-node Inventory Check
- ET-7. Acquisition Replication
- ET-8. File Integrity Check

Each event was then analyzed to determine how adequately it was represented in the system, what additional metadata might be needed to improve the representation, and whether the metadata were or could be automatically created by one of the Chronopolis sub-systems (MCAT, ACE, or Replication Monitor) or required human intervention (e.g., initial submission and integrity check).

## Chronopolis Advanced Access Portal

The current implementation of the Chronopolis system contains a collection of software systems that are loosely coupled in their management interfaces. Expert level knowledge of the Chronopolis system components is required to perform the various functions within the system. The project is currently working to make this operational functionality available to a broader group of Chronopolis users, with focus on the current expert users, data providers, and project stakeholders. To this end, software interfaces to the current components focused on users' needs are currently under development or are contemplated for future development. These tools integrate the information from the existing components into a single, easy to understand portal.

For example, the information that is required for monitoring the status and error conditions can currently be found in Chronopolis components if one knows where to look. Since various software components are installed at each of the Chronopolis archiving institutions, the user must possess a mental map of the installations. This is knowledge that the designers have, but a typical user is far less likely to acquire it. To facilitate a less complex interaction for users, the Chronopolis project has designed a web-based status display that integrates information from all sites into an integrated page. The aim of this status display is to pull status information from all Chronopolis components and integrate it, so that users can quickly ascertain the state of collections of interest, find any replication or verification errors, then drill into the information to discover the cause. This interface will also provide access to collections' metrics and reports.

The ACE Audit Manager provides Javascipt Object Notation (JSON)[iii] access to most functions. Among these are collection status, state of an individual collection, event log browsing, and item level browsing. Access permission to the JSON services requires an ACE account. Within Chronopolis a common read-only account has been created at all partner sites so that various harvesting software may automatically retrieve data from audit managers for display

in a portal. Specifically, only the following access is required at all three sites: overall collection status, item level browsing, log retrieval, error report retrieval, activity report viewing, download collection digests, duplicate detection, and token downloading. This access allows remote sites to pull enough information to determine what differences may exist among their collections and their peers' collections, and to show overall collection health.

## Chronopolis Tools

Chronopolis is comprised of several technologies. These have been designed to work together to provide a seamless preservation environment of geographically replicated content. One of the explicit goals in Chronopolis is to investigate emerging tools which are particularly appropriate for the digital preservation community.

### BagIt transfer format

BagIt[iv] was chosen as the principal format for collection packaging during the transfer process from data provider to the Chronopolis system. BagIt was originally developed by the California Digital Library (CDL) and the Library of Congress. The Library of Congress has used it to transfer over 80TB of highly heterogeneous materials between differing storage systems. The BagIt specification was written to provide a simple, generic, easy to use method to accomplish data transfers. The key features of BagIt are its inclusion of a clear inventory (including a collection directory structure, object names and checksums), and its inherent ability to parallelize the transfer process for high-speed exchange.

Much like a .tar or .zip file, the BagIt format is simply a specification for aggregating a collection of files into a single package file. A BagIt file has a minimum of two housekeeping or extra files beyond those of the collection. These include a "manifest-algorithm.txt" file, and a "bagit.txt" file. Along with these two files at the root level the collection objects are placed in a /data directory. The "bagit.txt" file is a two-line file specifying BagIt version and character set used. The "manifest-algorithm.txt" is a key file which includes a complete inventory of the collection giving pathname and checksum for each data object. In practice the "algorithm" in the "manifest-algorithm.txt" is replaced with the checksum method used (MD5, SHA-1, etc.).

To enable fast parallelizable network transfers, a large bag can be transferred with "holes" in it, that is, with files that are missing but that can be retrieved by URL. The transfer of a large "holey" bag can be greatly sped up by fetching the missing files with multiple parallel retrievals using ordinary HTTP-aware tools. The holey bag option requires an additional "fetch.txt" file at the root level of the bag. A URL pointer or identifier must be listed for each object in the collection. This requires the data provider to assure the data objects are accessible via an http server.

Perhaps the foremost benefit of the BagIt format is that it contains an authoritative inventory and file checksum that can be used at various processing steps to assure the completeness of the collection. This turns out to be quite useful for automation of integrity checking of large collections as they are processed through the Chronopolis system. Additionally a significant BagIt feature is that entire collections can be moved around referencing a single file. When using a holey bag this file can be quite small, amounting to a fraction of the overall collection size. Holey bags allow the transfer process to be sped up by parallelizing the exchange process. Open source code exists which instantiates up to 16 parallel processes to tackle the URL transfers.[v]

Division of collections into bags is not always a straightforward decision. In the Chronopolis project the data providers have been encouraged to put whole collections up to 5 TB into single BagIt files. Experience shows that beyond this size it makes more sense to divide the collection into smaller parts.

The BagIt format is growing in popularity, particularly among the NDIIPP partners, many of whom have adopted this format in their projects. At the June, 2009 NDIIPP Partners Meeting, several presentations included discussion of the BagIt format.[vi]

### Storage Resource Broker (SRB)

The Storage Resource Broker (SRB) is a data handling middleware package that provides uniform access to data collections stored within a data grid.[vii] The data grid may consist of heterogeneous storage devices distributed across multiple organizations. The primary benefit of the SRB is that it allows for a single uniform access to manage collections regardless of whether they are on a tape drive across the world or sitting on your desktop. Chronopolis uses SRB to manage all the data collections housed across its currently configured three zones located at SDSC, NCAR, and UMIACS.

As part of its management scheme the SRB uses a metadata catalog (MCAT) which sits on top of a database. The MCAT's main purpose is to manage access level metadata for individual objects, recording attributes such as storage location addresses, pathnames, filenames, ownership, security information, and user permissions. All attributes are needed to store and retrieve objects across distributed systems.

The SRB is, in essence, the glue that holds the data grid together. As part of the configuration process SRB systems are set up at each participating organization. The SRB systems are configured as unique zones and federated into a data grid. Individual data storage devices are registered into the MCATs at each zone. Users are set up and read/write permissions are configured at each site, as well as across zones. The end result is a robust data grid.

The SRB has several human and machine interface methods.[viii] Since the SRB is truly used as a middleware tool in Chronopolis, the interface used is machine level and at the command line using Scommands. Machine level

APIs are used to integrate communication to other developed tools such as the Replication Monitor and ACE. The human level Scommands are used primarily during the initial ingest process.

The SRB, which has been in existence for almost a decade, is being replaced by a more advanced tool, iRODS.[ix] iRODS has a framework quite similar to the existing SRB but includes a rule-based level allowing customizations of many aspects of the data grid. Future work includes transitioning Chronopolis to the iRODS middleware package.

## Replication Monitor

The SRB Replication Monitor[x] is an automated web-based application developed to monitor the copying of collections within the Storage Resource Broker. The monitor was designed to provide an easy-to-use, hands-off mechanism to reliably transfer data between zones in the SRB. Copying small collections will usually occur within a few hours in a relatively stable environment. However, copying the millions of files and terabytes of data required by Chronopolis requires days, if not weeks, during which time any number of transfer errors may occur. The Replication Monitor attempts overcome these errors by retrying operations several times during different time windows in an effort to complete a file copy. The Replication Monitor is able to detect unusually high failure rates and pause itself while it waits for the network or software to stabilize.

The Replication Monitor replicates data on a per-collection basis. Each collection has one or more replica sites registered. Each site has its own independent replication policy. This policy determines how many simultaneous copies may occur between a master and replica site. Determining simultaneous copies is a function of network latency, average file size in a collection, and the capability of the MCAT database at each site. For example, on a collection with large files, NCAR will have a policy that attempts more simultaneous copies than UMIACS. This is due to NCAR having a much faster (10Gb/s) connection than UMIACS (1Gb/s). Conversely, when managing collections with many small files, NCAR will generally configure fewer simultaneous copies than UMIACS as the bottleneck will be the MCAT. UMIACS is able to benefit from more connections due to the higher latency between request and response.

After collections have been registered and policy set, a replication process is started. While each replica site may appear to be linked, they in fact execute tasks independently from each other. This prevents an outage at one replica site from affecting the performance of another. Replication is a two-part process, both parts operating in parallel. In the first part, a list of files is gathered from the master site, which is compared to the replica site. Any files that do not exist on the replica site, or have a different checksum than the master, are added to a queue to be replicated. The second part consists of a pool of threads monitoring the work queue for files that need to be replicated. When a thread retrieves a file from the queue, it will attempt to copy the file to the replica site. After replication is finished, any files seen on the master site, which were not able to be replicated, are flagged with errors. In the event of a network outage, all parts of a replication will pause until the network or software is back online. As an example, during replication to NCAR, there were several maintenance downtimes on their SRB services. The replication process was able to continually pause and resume as services went offline and became available again.

## Auditing Control Environment (ACE)

ACE[xi] is an integrity-monitoring platform based on creating a small-size integrity token for each digital object upon its deposit into the archive (or upon registration of the object in an existing archive). This token is stored either with the object itself or in a registry at the archive as authenticity metadata.

These tokens are linked together through time spans by an auditable third party. For each time interval, cryptographic summary information (CSI) that depends on all the objects registered during that time interval is generated. The summary information is very compact and is size-independent of the number or sizes of the objects ingested. The period of each round is currently defined in seconds but can adapted as needed by the archive.

At the end of each day, all CSIs generated are aggregated into a final witness value. This witness value is a single number that is used to verify all CSIs issued during the previous day. The value is expected to be stored on reliable, read-only media, and published over the internet. An independent auditor, given a trusted witness, may assert the integrity of all CSIs for a given time period. Once CSIs are certified, they may be used to validate all tokens covered by the summaries. Once tokens are validated, an auditor may assert that any file whose cryptographic digest matches its token has not been tampered with to a high probability.

Regular audits will be continuously conducted, which will make use of the integrity tokens and the summary integrity information to ensure the integrity of both the objects and the integrity information. In the Chronopolis implementation, audits can also be triggered by an archive manager or by a user upon data access. However, it is assumed that the auditing services are not allowed to change the content of the archive even if errors are detected. The responsibility for correcting errors is left to the archive administrator after being alerted by the auditing service.

The ACE system consists of two components, the first of which is an Integrity Management Service (IMS) which gathers token requests into rounds and generates Integrity Tokens (IT) at the end of each round. The IMS is also responsible for publishing nightly witness values. UMIACS currently hosts a publically available IMS for any party to use. The second component of ACE is a suite of multiple, independent Audit Managers (AM) that are

installed locally at archives and that periodically check the integrity of monitored objects according to a locally defined policy.

The ACE Audit Manager is a web-based interface that is able to monitor multiple collections across different types of storage. The AM periodically scans different collections according to a customizable policy. Each scan checks the integrity of files in a collection, and can be configured to check the integrity of the digests securing those files. The AM keeps a detailed audit trail describing all changes that have been observed. At any point, reports showing the current state of a collection may be generated, as well as historical reports showing collection changes over time.

## Future Directions for Chronopolis

Chronopolis is conceived as an ever-evolving enterprise. To this end there will always be additions, corrections and improvements to be done. There are several which are already planned for the near future.

**Updated auditing procedures** The process of auditing collections will be expanded in several areas. Currently auditing is contained within a single partner site and comparison between collections is a manually triggered process. In the future, this comparison will be automated and included as part of a collection's audit policy. This will allow partner sites not only to assert that their holdings have not been modified, but also will allow them to assert their holdings are identical to partner holdings.

**Updated portal** In the future additional pages will be added to support the integrated Chronopolis view, extending the functionality to control functions of the Chronopolis system. Functions here could include including starting and stopping replications using a drag and drop interface, restoring data zone-to-zone, and restoration of collections to the data provider. Data ingest could also be automated, using graphical controls to control tools like Bagit, which are doing the actual transfers. For validation, starting inner or intra zone validation can be done with a drag and drop interface.

In addition, for engineering system monitoring, interfaces will be added to monitor parameters like network performance, disk usage, and system loading that would help assess the overall health of the Chronopolis system over time.

**Automation of collection ingest** There are several areas of the ingest process that can be automated. Initial collection level metadata can be obtained and ingested by having the data provider fill out some sort of electronic form. With the basic collection level attributes this form could also specify retrieval information, including transfer format, collection package naming and location details which all could be managed via a metadata schema. There are a couple of methods we wish to explore for automation of the Chronopolis SRB ingest. One is to use the BagIt technology and explore developments to automate its ingest directly into the SRB/iRODS storage devices and population of the necessary MCAT attributes. The other is to utilize existing SRB/iRODS lightweight clients. In this scenario the clients would be developed specifically for the Chronopolis system and thus easily dropped at the data provider's organization and federated into the data grid. Existing collections at the data providers' organizations can be registered into their SRB/IRODS and automatically replicated to the Chronopolis archival zones. With some research, this may work as an interoperability method for data exchange with the existing MetaArchive LOCKSS system.

**New collections and storage nodes** The Chronopolis project is investigating the addition of new collections and new storage nodes. The addition of new collections is a given, and negotiations have begun with potential data providers. It is likely that most new data coming into Chronopolis will not be from current NDIIPP collections. This gives the project a chance to work with an even greater diversity of content and sources. New storage nodes are also a possibility. This will enhance the preservation abilities of the network allowing the creation of new storage locations, which are also likely to be in geographical locations different from the current nodes.

**Fully-fledged business model** Chronopolis is currently evolving from a project, fully-funded by a single source (NDIIPP), into a broader-reaching, fee-for-service model. This requires a larger stable of Service Level Agreements (SLAs) with data providers and stricter contracts among the storage nodes in the network. These documents are anticipated to be written and in place by the fall of 2009.

**TRAC certification** One of the significant tasks planned for the next year is a full TRAC certification[xii] for the Chronopolis network. This is viewed as an important step in verifying the current and future work to be undertaken. This will be a full certification, conducted by outside auditors who are not part of the Chronopolis project.

## References


i Song, S. and JaJa, J. ACE: A Novel Software Platform to Ensure the Integrity of Long Term Archives. in Archiving 2007. 2007: IS&T.

ii Smorul, M. and JaJa, J. A Case Study in Distributed Collection Monitoring and Auditing Using the Audit Control Environment (ACE). in Archiving 2009. 2009: IS&T.

iii Crockford, Douglas (May 28, 2009). "Introducing JSON". json.org. http://json.org.

iv BagIt Specification: http://www.cdlib.org/inside/diglib/bagit/bagitspec.html

v Library of Congress Transfer Tools: http://sourceforge.net/projects/loc-xferutils/

vi NDIIPP Partners' Meeting Report: http://www.digitalpreservation.gov/news/2009/20090702news_article_partnersmeeting.html

vii Storage Resource Broker: http://www.sdsc.edu/srb/index.php/Main_Page

viii Storage Resource Broker Interfaces: http://www.sdsc.edu/srb/index.php/FAQ#Interfaces_and_Tools

ix iRODS: https://www.irods.org/

x Replication Monitor: https://wiki.umiacs.umd.edu/adapt/index.php/Replication:Replication_Monitor_2.0

xi Auditing Control Environment: https://wiki.umiacs.umd.edu/adapt/index.php/Ace:Main

xii TRAC Certification: http://www.crl.edu/content.asp?l1=13&l2=58&l3=162&l4=91