Building a Dark Archive in the Sunshine State: A Case Study

Priscilla Caplan
Florida Center for Library Automation
Gainesville, Florida, USA

Abstract

The Florida Center for Library Automation (FCLA) is implementing the FCLA Digital Archive as a preservation repository for the use of the libraries of the eleven state universities of Florida. The FCLA Digital Archive is built upon DAITSS software, which is still in development at FCLA. DAITSS is designed to be a "dark archive," dedicated exclusively to ensuring the long-term viability, integrity, and renderability of archived content.

DAITSS software follows the Open Archives Information System (OAIS) model in the implementation and functional separation of Ingest, Data Management, Archival Storage and Dissemination. It supports two levels of preservation, bit-level and full, and implements full preservation using the active strategies of formation normalization and forward format migration. It obtains and maintains detailed technical metadata, tracks complex relationships among objects, and documents digital provenance.

DAITSS is currently being used by the FCLA Digital Archive for ingest only. Other functions are still being designed and programmed, and are expected to be completed in calendar 2005. When completed DAITSS will be made freely available for use by the cultural heritage community.

Figure 1. DAITSS logo

Introduction

The Florida Center for Library Automation (FCLA) was established to provide centralized automation services for the essential needs of the libraries of the ten (now eleven) public universities of Florida. FCLA services include maintaining a public catalog and library management system shared by the state universities, supporting consortial purchasing of electronic resources, and maintaining applications and infrastructure to help the libraries manage their digital content.

Since the libraries started building digital collections of their own materials in the mid-1990s, there was an assumption that FCLA would provide a preservation "archive" for some of their digital files. Initially, these were mostly TIFF masters of locally digitized books and photographs, for which FCLA provided secure central storage, backup, and data management services. However, as the graduate schools began allowing or requiring electronic dissertations (ETDs), FCLA began accepting copies of ETDs consisting of PDFs and associated files in a variety of image, sound and video formats. The libraries are responsible for the long-term availability of these ETDs, and the library directors were concerned that more would be required for these materials than secure storage.

In 2002 FCLA applied for and received a grant from the Institute of Museum and Library Services to build a preservation repository based on the active preservation strategies of format normalization and forward migration. The FCLA Digital Archive began taking a limited amount of materials for ingest in early 2005 and will expand to full capacity later this year. The software underlying the FCLA Digital Archive is called DAITSS (Dark Archive In The Sunshine State). The section "Current Status and Future Plans" describes the current state of software development; elsewhere in this paper the application is described as though complete.

DAITSS Overview

Two defining qualities of DAITSS are 1) it is a "dark archive" and 2) it is designed exclusively as a preservation repository. Each of these requires some explanation.

Preservation is useless if the preserved materials are inaccessible, and most preservation repository systems are designed to allow some real-time online access to materials by authorized users. There are two models for this: either the repository system itself has a presentation interface, or the system delivers sets of files to an external presentation interface.

DAITSS will disseminate packages of content on request to authorized users, but there is no real-time online access. The model of providing a presentation interface to archived materials was simply infeasible. The eleven state universities of Florida are largely independent and they employ many different applications for digital content management and access. Some are establishing institutional repositories, some have implemented content management systems, and several are using “digital library” systems from various third-party vendors. It would not have been economically or technically possible to replicate all of these presentation systems within the preservation repository application.

It would have been possible to implement the second model, and to deliver content in real-time on request to distributed presentation interfaces. However, this would have required the libraries to store both preservation masters and presentation (service) copies in the preservation repository. Since the FCLA Digital Archive is expected to become a fee-based cost-recovery operation, the libraries were not enthusiastic about storing service copies, especially for large files.

Instead, we adopted a model that allows any system to be a front end to the preservation repository. The repository itself is “dark” in the sense that content requested by the libraries may be delivered hours or even days later. The universities and their libraries can store, manage, and provide online retrieval and presentation services for their own content however they choose. The FCLA Digital Archive is responsible for implementing preservation strategies to ensure that objects stored in the preservation repository are usable at any time. The libraries are responsible for deciding what content to copy to the FCLA Digital Archive for centralized preservation services.

If, for example, a library has an oral history website, it may have a master AIFF file and derivative RM, MP3 and QuickTime streaming versions of each interview. The library may decide to send only the AIFF files to the FCLA Digital Archive. If in the future another streaming format is needed, the library can retrieve the AIFF file back from the Archive, and generate another derivative. If AIFF itself threatens to become obsolete, the FCLA Digital Archive will do a forward migration of the stored AIFF file to a reasonable successor format. In this way, the libraries retain complete control of their access and presentation systems, while responsibility for preservation decisions is shared between the libraries and the Archive.

An advantage of being “dark” is the DAITSS software is relieved of the responsibility of providing access and presentation services. Also, since the assumption is that it will be used as a “back-end” to other systems, there are no facilities for creating or collecting content, generating bibliographic descriptions, or similar functionality often provided in other applications such as institutional repositories or digital library systems. DAITSS has the single goal of ensuring the long-term viability, integrity and renderability of ingested content, and it is designed from the start to implement active preservation strategies towards this end.

DAITSS Architecture

DAITSS implements the functional model of the Open Archival Information System reference model (OAIS). Like the OAIS, DAITSS provides the five functions of Ingest, Data Management, Archival Storage, Administration, and Access.

Ingest

In the OAIS model, the agent providing the information to be preserved (the Producer) delivers metadata and content data files together in a Submission Information package (SIP). In the DAITSS implementation, a SIP consists of metadata in the form of a METS document and one or more content data files. The SIP is processed by Ingest, which populates the archive management database, constructs an Archival Information Package (AIP), and passes the AIP to storage through a generic storage interface. (Ingest is described in more detail below.)

Data Management

DAITSS uses relational database tables implemented in MySQL to control its own repository management functions, to record preservation metadata, and to provide billing and reporting capability. The data tables are populated and updated by Ingest. Some events such as dissemination and fixity checks are recorded post-ingest by other services.

Archival Storage

DAITSS contains a generic interface for which implementations can be written for any specific storage system. The Storage Interface specifies a minimal set of required behaviors that are likely to be supported by all storage devices. In the FCLA Digital Archive, a TSMStorage implementation interacts with a Tivoli Storage Management system using a robotic tape library, but another implementation could be substituted. Once committed to Archival Storage, an AIP cannot be changed in any way – to correct, replace, or migrate the content, the data must be disseminated and re-ingested. The Storage Maintenance service ensures that stored masters remain fixed and readable.

Administration

DAITSS supports administration of the preservation repository through interfaces for updating configuration files and profiles.

Access

The Dissemination service in DAITSS is equivalent to the Access function in the OAIS model. Dissemination verifies that requests are authentic, copies content from

Archival Storage, and creates a Dissemination Information Package (DIP). The DIP contains by default both the originally submitted content and the most usable version of the content. Dissemination does not remove any content from the repository.

Withdrawal

The Withdrawal service, which is not explicitly an OAIS function, deletes an AIP from the repository. Only a complete AIP can be withdrawn.

DAITSS manages information about digital objects at three levels: the Intellectual Entity, the Data File, and the Bitstream. An Intellectual Entity is defined as a coherent set of content that is described and used as a unit, for example a book or map. The boundaries of an Intellectual Entity are subjective and up to the Producer; in different instances it might be a web page or a website, a serial issue or a serial volume. Each SIP is assumed to contain at least one representation of a single Intellectual Entity, and each AIP must contain all of the Data Files necessary to render the Intellectual Entity to a user.

A Data File is a single named digital file, such as a PDF, TIFF or XML file. A Bitstream is a sequence of bits embedded within a Data File, that has meaningful attributes for preservation purposes. Data Files and Bitstreams are implemented as hierarchical sets of Java classes. For example, in the Data File hierarchy an XML File is a subclass of Markup File which is a subclass of Text File which is a subclass of Data File. Attributes are inherited from each higher level. Similarly an XML Stream is a subclass of Text Stream which is a subclass of Bitstream.

Technical metadata is associated with both Data File and Bitstream objects. In the case of an image file, for example, characteristics such as size and file type are properties of the Data File and detailed technical characteristics such as bits per sample and color space are properties of the Bitstream.

Preservation Functionality

DAITSS supports two levels of preservation, “bit-level” and “full.” Bit-level preservation preserves a file exactly as it was submitted, and includes steps to ensure the viability and integrity of the file. An implementation-specific number of master copies of each submitted file are created, and each master is independently stored, backed up, and refreshed when necessary. In the FCLA Digital Archive implementtation, the storage system is not Web-accessible and security is controlled by password protection and restricted physical access to machines allowed to communicate with the storage server. Stored files are continuously monitored for fixity and viability.

Full preservation is intended to ensure renderability as well, and includes bit-level treatment of the SIP as submitted as well as the active preservation strategies of format normalization and format migration. Full preservation is available only for files in formats for which an action plan has been developed and implemented in DAITSS. An action plan is a human-readable document which specifies how the format will be treated in the short-term and in the longer-term, and when the plan will be reviewed. Treatment may include “normalization” (creating a version of the file in a different and possibly more preservation-friendly format) and/or forward migration (creating a version in a format considered to be a successor format).

One basic premise behind DAITSS development is that we know so little about digital preservation at this time, it is wise to err on the side of caution. Therefore redundancy is built into the system wherever possible. Multiple master copies of each file are stored, two different message digests (SHA-1 and MD5) are calculated for each file, and all metadata is stored in both the system management database tables and as an XML document in the AIP. In the same vein, normalized versions of files are created whenever possible. A submitted PDF file, for example, will be normalized into one or more page-image TIFF files. Both the submitted version and the normalized version will be carried into the future through forward migration. Normalization therefore creates a second path for ensuring long-term usability; if one path reaches a dead end (100 years in the future there is no usable successor to PDF and its successor formats) there is a chance that the second path may still be viable.

To implement an action plan, Ingest must be able to identify the format and to build a Data File object in that format. The Data File object must know how to validate itself, to extract its own technical metadata, and if appropriate to create normalized and/or migrated versions which are themselves Data File objects. This may involve writing new Data File and Bitstream classes, or adding methods to existing classes. Format identification and validation are done with algorithms similar to JHOVE (hul.harvard.edu/jhove/) except that validation errors are documented and some are tolerated.

Although forward migration is implemented on ingest, most files do not require migration at the time they are submitted. If a migration later becomes necessary, the AIP must be disseminated and re-ingested. This leaves the decision whether to do mass migration or migration on request up to the repository management. To do a mass migration, all files of a particular format would be identified through the reporting (Data Management) service and the AIPs containing them would be disseminated and re-ingested. To do migration on request, any Intellectual Entity requested by an authorized user would be disseminated, re-ingested, and again disseminated, this time to the requester.

DAITSS is designed with the goal of providing a usable version of any Intellectual Entity in the preservation repository at any point in time. To that end, it must be able to identify all Data Files making up the Intellectual Entity (including normalized and migrated versions) and to understand the relationships among them. It must also be able to demonstrate digital provenance by documenting all actions of the repository system in relation to the stored objects. Management database tables record relationships and events in detail. As with all other metadata, this information is copied into an XML document and stored with the content data objects in the AIP.

Ingest Procedures

Most of the preservation functionality in the DAITSS system occurs in the Ingest service. Ingest will process any SIP provided to it in a specific input directory. In the FCLA Digital Archive implementation, participating libraries FTP packages to a designated FTP directory. The packages are examined by a program which may or may not do some preprocessing and then copies the packages to the DAITSS input directory.

Ingest validates the SIP, normalizes and/or migrates files, records metadata, and builds the AIP. Looking at this in more detail, for every incoming SIP, Ingest performs the functions described below.

The SIP descriptor (which must be a METS document) is identified and validated against the METS schemas. Any descriptive or technical metadata contained in the METS file is extracted for later use. The Producer is identified along with any processing information optionally supplied by the Producer, which together are used to determine the level of preservation treatment to be accorded to the files in the package, and reporting and billing profiles.

Every file in the SIP is checked for viruses and the SIP is rejected if a virus is found. If the SIP descriptor included a checksum (message digest) for the file, this is verified and any difference is reported. The format of the file is identified, and a Data File object is created for the file.

The Data File object validates itself against an internally stored profile. Technical metadata is extracted from the file and compared with submitted technical metadata; conflicts are reported and resolved. Bitstreams are identified and Bitstream objects including detailed technical metadata are created.

If the SIP does not include all files considered necessary to ensure the long-term usability of the Intellectual Entity, Ingest attempts to harvest the missing files. This is done, for example, if an XML file in the SIP references an external schema, DTD or stylesheet not included in the package. When one or more external files are downloaded, a new SIP descriptor is created.

Files in the SIP are evaluated for preservation treatment according to their file format and parameters supplied by the Producer. Migrated and then normalized versions are created where necessary. The sequence ensures that if a file is migrated to a format that requires normalization, the normalized form of the migrated file is created.

Technical metadata, relationship metadata and event metadata are recorded as all of the above functions are performed.

Finally, the AIP is created. Unwanted, duplicate and global files are eliminated. (Global files are files which occur in SIPs so frequently – for example the METS schema – that they are stored only once by the system.) At least two message digests using different algorithms are created for all remaining files. All metadata pertaining to the Intellectual Entity, Data File objects and Bitstream objects are formatted into XML according to a local METS extension schema, and a METS format AIP descriptor is created. Some files are compressed in a non-proprietary and lossless compression scheme. The entire AIP, including the AIP descriptor and all content data files, is then written to storage through the Archival Storage interface.

The number of copies to write is configurable. The FCLA Digital Archive is configured to write three copies of the AIP – two local and one remote. Each of these copies is considered a "master," in the sense that all files are independently addressable by DAITSS. (In many storage management systems, backup copies are assigned names by the system and can only be accessed through the recovery facilities of the system.) Storage maintenance utilities such as those that verify the integrity of files will record a successful event only if all three masters pass the test.

When writes to storage are completed for all copies of the AIP, the DAITSS data management tables are updated. Any critical error up to this point will cause processing for the entire SIP to be backed out and the Producer to be notified. SIP processing continues until the input directory is exhausted.

Current Status and Future Plans

At the time of this writing only the Ingest function is complete and operational. The other functions are expected to be coded within the next twelve months. FCLA staff are currently negotiating formal service agreements with Producers (the libraries of the public university system) and exercising Ingest on a limited set of materials (TIFF masters of aerial photographs). Procedural documentation for the FCLA Digital Archive is still being drafted.

Mechanisms for instituting cost-recovery billing will be included in the DAITSS Data Management (reporting) service, but the FCLA Digital Archive does not plan to bill for services until 2006 at the earliest. The directors of the eleven state university libraries constitute the Advisory Board of the FCLA Digital Archive and will decide when to start charging and what billing algorithms to use.

FCLA is holding discussions with a small number of institutions about possible partnership arrangements during 2005. The partner sites would implement DAITSS at their own institutions and provide feedback on where the software needs to be generalized or improved. Some sites may be co-developers of the remaining functions. When all functions are complete, we plan to release DAITSS 1.0 as freely available source code.

As a software application, DAITSS is being built with open source distribution in mind. The code is written in Java and developed under Linux. The database used is MySQL, and care was taken to allow other relational databases to be substituted. Some third-party software is included for specific functions such as virus-checking, but an effort was made to use freely available components whenever possible.

Two rather substantial enhancements to DAITSS are already known to be required. First, we want to add support for digital signatures so that incoming SIPs and user requests can be authenticated. Second we want to make the metadata in DAITSS management tables fully compliant with the anticipated PREMIS preservation metadata specification. We have applied for grant funding to test the exchange of PREMIS-compliant information packages between the FCLA Digital Archive and another operational digital archive.

Conclusion

Many of the applications that libraries and their parent institutions are depending on for digital preservation were initially designed to help capture and describe digital information. For example, the DSpace institutional repository system was first released with substantial functionality to handle the needs of different communities of authors and depositors, but with only bit-level preservation functionality. Many of these applications, including DSpace, are now being redesigned to support active preservation strategies.

DAITSS can be seen as one of a second generation of preservation repositories designed from the start to ensure the long-term renderability of archived objects. DAITSS features that support active preservation include: code and metadata to implement format normalization and format migration; code and metadata to maintain complex relationships among various versions of bitstreams, files, and intellectual objects; and code and metadata to track the detailed digital provenance of each stored object.

As digital preservation is a relatively new field, we believe firmly in the principle of “hedging our bets,” and DAITSS has built-in redundancy wherever possible. We also believe that the ability to experiment is necessary, so DAITSS allows much flexibility in how format migrations are performed, how many intermediate versions are retained, and what is included in a Dissemination Information Package.

Finally, we believe that there is no one true model for digital preservation, and that software applications implementing many different models should be available for the community to learn from. We hope that DAITSS will prove useful to some institutions as one tool for the digital preservation of some of their resources.

For more information on the FCLA Digital Archive and the DAITSS application, see the FCLA Digital Archive home page at www.fcla.edu/digitalArchive.

Biography

Priscilla Caplan received her B.S. from Harvard University in 1974 and her M.L.S from the University of North Carolina at Chapel Hill in 1978. She worked in library systems at Harvard and the University of Chicago before coming to the Florida Center for Library Automation as Assistant Director for Digital Library Services in 1999. She currently co-chairs the OCLC/RLG Working Group on Preservation Metadata: Implementation Strategies (PREMIS).