Migrating Repository Metadata and Users: The Harvard DRS2 Project

Andrea Goethals, Franziska Frey, Robin Wendler, Chris Vicary and Spencer McEwen; Harvard University; Cambridge, MA (USA)

Abstract

The Digital Repository Service (DRS), first launched in 2000, is Harvard Library's long-term preservation and access repository. The DRS is central to the Library's discovery, access and management infrastructure; and vital to digitization, reformatting and collection management workflows throughout the university.

In 2008 the Library began the DRS2 project - a multi-year repository enhancement project to update to the latest technologies and digital preservation standards and practices; and to provide curators, collection managers and repository staff with significantly enhanced tools. This paper describes early findings of the last stage of the DRS2 project - the migration of the repository metadata for over 46 million files into the newly enhanced DRS, and the facilitated transition of DRS users to learning and adopting the new repository concepts and tools.

Background

The Digital Repository Service (DRS)

The Digital Repository Service (DRS) is Harvard Library's long-term preservation and access repository. The first incarnation of the DRS was made possible through Harvard's Library Digital Initiative (LDI) program [1] which provided the funds, beginning in 1998, to enable the Library to collect digital material. It not only funded the development of technical infrastructure including the DRS, but it also funded the hiring of specialists and 49 internal grants to build digital collections. The DRS went into production in 2000 as the preservation back end for these digital collections, made accessible to users through integrated discovery and delivery platforms.

In the decade that followed, 55 Harvard libraries, archives and museums grew to use the DRS, as shown in Figure 1. In addition, the DRS became integrated with the workflows and tools used by the image and audio reformatting labs at the university, and the ecosystem of systems and tools used by the Library for DRS ingest, management and access, as shown in Figure 2.

Within the DRS, primarily administrative and technical metadata was stored in an Oracle database for all content, and additionally written to METS files [2] for some types of content. Most of the metadata schemas were custom, as there were not standard preservation schemas when the DRS was first developed. The data model was very simple - everything was modeled at the file-level but relationship metadata could be traced to access or report on related content.

Over the years enhancements were made to the DRS infrastructure to support both digitized and born-digital content in image, audio, text, geospatial, document and web formats. In 2008 the DRS project, a large-scale effort to plan the next-generation DRS, began.

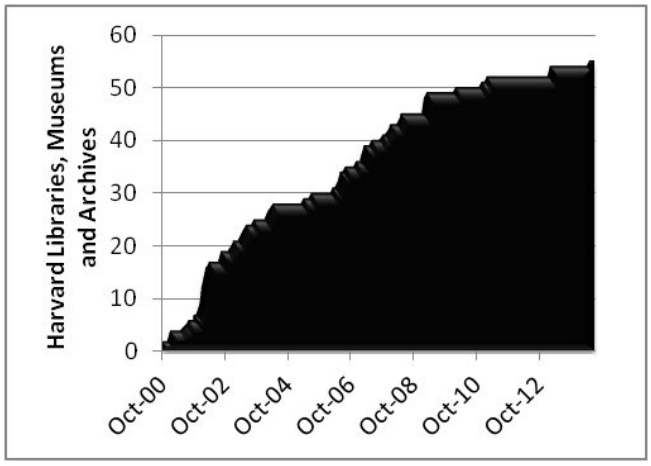

Figure 1. Growth in the number of Harvard libraries, archives and museums depositing and managing digital material in the DRS from 2000-2013.

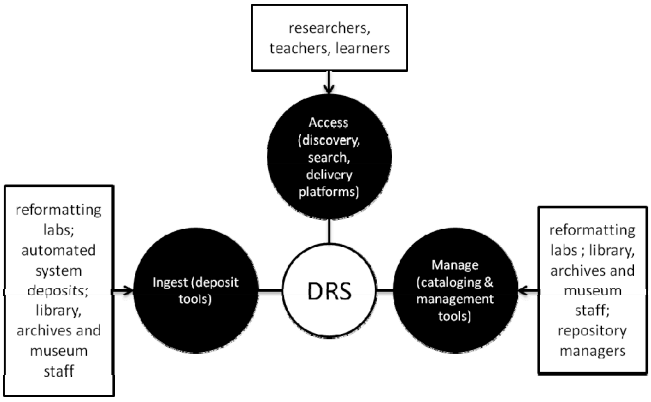

Figure 2. The DRS is central to the Library's infrastructure for collecting, managing and accessing digital collections, and has many different types of users.

The DRS2 Project

The DRS2 project was motivated by advances that had been made in technology, digital preservation standards and best practices, as well as user expectations in the decade after the DRS was put into production. Digital preservation systems and services that are not kept current risk providing insufficient preservation services, ultimately leading to lost content, inability to take advantage of community resources such as format registries, failure to pass repository audits, and dissatisfied users.

While enhancements and new applications had been added to the DRS infrastructure over time, the metadata foundation in

particular was in need of an overhaul. The DRS had been based on custom schemas and models because the building blocks of preservation repositories such as the OAIS reference model [3], PREMIS preservation metadata [4], METS wrappers and community-standard technical metadata schemas did not yet exist. Altering the repository to make use of these data modeling and metadata advances was a very large undertaking that had to be spread out over multiple years. For example, changing the file-based DRS data model to be consistent with the object-based PREMIS data model produced a ripple effect of changes that had to be made to the repository database and every application that processed DRS content during ingest or after it was already in the repository.

The project began in the fall of 2008 with an assessment of third-party repository software resulting in a decision to continue enhancing the DRS software that had been developed in-house. From 2009-2013 the architecture was redesigned, software enhanced, and exposed to Library staff as beta releases for testing and training. In 2013 the focus changed to planning and executing the last stage of the DRS2 project - the migration of metadata and users to the new version of the DRS, which is the focus of this paper.

DRS2 Migration

The DRS2 migration has two interrelated components - one technical (the metadata migration) and the other organizational (transitioning of users to use the new DRS). Technically, the metadata migration was necessary to transform the repository metadata to the new object-based data model, database schema, METS metadata packaging format and schemas.

DRS "users" (depositors, curators and collection managers) needed to be transitioned to use the enhanced repository tools. Because of the large number of DRS users as compared to the small number of DRS support staff, the user transitioning had to be phased to spread out training and support needs. This was considered one of the most important parts of the DRS2 project to get right because the enhanced repository had been anticipated for years by DRS users and a successful rollout was critical for customer satisfaction, staff morale, and to minimize operational disruptions.

The Challenge

This would not be a simple metadata conversion, such as a mapping from one schema to another. The DRS contains over 46 million files. To conform to the new PREMIS-based data model, these files needed to be grouped into logical objects (e.g. digitized books or sets of derivative images), and METS object descriptor files needed to be generated for each object.

The source of the metadata to populate the METS object descriptor files came from several different locations: an Oracle database, metadata embedded within the files themselves, and for some content, legacy METS files and/or catalog records and finding aids. The FITS tool [5] would be used to process all the files, identify and validate file formats and to extract metadata from within the files.

The DRS contains content in many different formats which required different migration logic and format-specific technical metadata. Some of these formats posed additional complexities. For example, there are instances in the DRS of very large page-turned objects that have thousands of files. Special code had to be written to handle them.

Because of these complexities and the large number of files in the DRS, all of this processing would take years if it were not done as efficiently as possible. As an illustration, if the processing took 1 second per file and the processing was done as a linear process, the migration would take over 530 days. A process running this long would be too disruptive for everyone involved. The goal was to complete the migration in under a year.

Fifty-five Harvard libraries, archives, museums and reformatting labs deposit and/or manage content in the DRS. Each of these units needed to learn the new DRS concepts and tools and change their deposit and management workflows. The goal was to transition these users over to the new DRS in the least disruptive way and without overwhelming DRS support staff. Once a unit's metadata had been migrated, it could only be managed in the new DRS. Users did not want to have to work in the old and new systems simultaneously so it was important to minimize the time that each unit had DRS content in both the old and new systems. In this way the timing of migrating a unit's metadata had to be coupled with the unit's readiness to switch over to use the new tools.

Analysis

To come up with the migration plan, three parallel tracks of analysis were performed - repository content, metadata and user analysis.

Content Analysis

The primary goal of the content analysis was to develop algorithms for building objects from files during the migration. Prior to the analysis "content models" or object types had been designed and documented, so algorithms for building objects were needed for fifteen different content model types. Some examples of content models include still image, audio, document and web harvest objects. The content model-specific algorithms consist of a series of steps that describe which metadata to use to select files as starting points, how to identify related files that should be part of the same object, and which metadata to associate with different object components. Because most of this metadata is stored in an Oracle database, the language of these algorithms is primarily SQL.

Besides developing these object-building algorithms, the content analysis also revealed dependencies between different types of objects that needed to be taken into consideration when designing the order for sequencing the migration. For example some of the auxiliary content models like target image objects and color profile objects needed to be migrated before the still image objects that pointed to them through relationship metadata. As another example, the page-turned objects needed to be migrated before the still image objects because of the lack of metadata to definitively differentiate between standalone images and images that were part of a page-turned object. Fortunately because the METS files of page-turned objects referenced their page images, by migrating those first the remaining images could be correctly identified as being stand-alone still images.

Lastly, the content analysis revealed anomalies within the metadata that could be cleaned up before or after the migration. For example relationships between files were found that did not make sense, e.g. target image files that are described as being target image files of other target files. Orphaned file anomalies were also found, for example audio delivery files that were not related to any audio archival files. As another example objects

were found that had been merged into themselves. Most of these anomalies could be explained by errors made by DRS users when supplying metadata during deposit or when editing metadata using the DRS management applications. The DRS provides a great deal of functionality to depositors, curators and repository staff to manage metadata but this analysis showed that there needs to be more automated checking in place to make sure that metadata remains consistent. It was educational to learn that in cases where the workflow was completely automated, i.e. there was no manual input, there were also no anomalies found.

Metadata Analysis

The goals of the metadata analysis track were to make sure that each piece of existing metadata either had a home in the new DRS or that the key stakeholders agreed that it was no longer needed. Because much of the deposit of content into the DRS was centralized in reformatting labs, the heads of those labs were considered key stakeholders, along with individuals within Harvard libraries and archives who actively managed their DRS content and took a keen interest in any decisions related to the migration of their metadata.

Finding corresponding elements for the existing DRS metadata in the new DRS metadata schemas turned out to be the easiest part of the analysis. However, some metadata elements did not map cleanly. These were largely narrative fields such as processing enhancements, history, producer, and capture system that had been defined initially but over the years had been used by depositors in non-standard ways. Stakeholders had differing opinions about the utility of these fields. Some thought that not enough time had passed to know whether this metadata would prove valuable in the future, but the prevalent opinion was that the level of effort needed to capture some of the metadata did not match its value. Because this metadata had been recorded in inconsistent ways, its value was limited. It could not be used for data mining or reporting and it was doubtful that a human would want to look at this metadata for preservation planning.

Some of the more challenging tasks were making sure the metadata generated from automated tools that would be run during the migration, especially FITS, was well-understood, and determining how much descriptive metadata would be copied into the DRS during the migration, in which cases and from which sources.

The metadata analyst formed a small metadata working group to discuss how best to take advantage of the migration to add descriptive metadata to the object descriptors in an automated way from existing catalogs. They met several times over a two month period. They decided that metadata would be imported for still image and page-turned objects from the central MARC catalog, encoded archival finding aids, the image catalog and possibly from the geospatial catalog. This descriptive metadata would be encoded as MODS metadata in the object descriptors. The primary purpose would be to meet the preservation need of knowing what the object is in the repository, but the metadata can also be used to provide labels and captions in various delivery environments. The migration design had to ensure that the imported metadata was at the appropriate level of description and that the metadata sources were queried in the correct order to get the most appropriate metadata.

User Analysis

In parallel to the content and metadata analysis, DRS users were assessed to figure out a best sequencing order for switching them over to use the new DRS. DRS users included Harvard reformatting labs (which act as depositing agents for many Harvard units), DRS content owners, and systems that deposit automatically to the DRS.

Several factors were examined for these users: how actively they deposit or edit metadata as evident in system logs, and how much training they had on the new DRS concepts and tools by participating in training classes and/or by helping with beta testing. It was determined that two types of users warranted particular attention - “high-volume active users” and the reformatting labs.

The high-volume active users were the Harvard libraries, archives and museums that had very large volumes of content in the DRS and had deposited content to the DRS within the last year. The migration process needed to minimize disruption especially for these units. A survey was sent out to this group to find out how often they deposited content to the DRS themselves vs. used the reformatting labs as deposit agents. They were also asked how prepared they felt to use the new tools and if they had concerns about switching over to use the new tools. Since this group was very engaged with the DRS, they would be able to provide timely feedback on any problems encountered during the migration so they would make good candidates to switch over first.

The reformatting labs were also key to the migration plan in a couple of ways. They provide deposit and management services to a large number of DRS content owners, so we had to make sure that they would be able to continue to provide these services during and after the migration. Also since they were expert users of DRS tools, they could help test the tools and help field questions from other units switching over to the new DRS.

Key High-Level Requirements

In addition to the content, metadata and user analyses a core set of requirements to inform the migration plan and design were developed. The first was that the migration process had to be flexible and iterative. It needed to allow for mistakes in human analysis or software logic and be able to recover from them. Learning from Portico’s migration [6], the design needed to expect the unexpected. It was planned from the beginning that when problems are found it should be possible to rerun the migration on particular sets or kinds of content. Similarly, the old DRS database would be archived to the new DRS as a way of preserving the earliest incarnation of the repository metadata.

Because of the large amount of content in the DRS it was a given that the migration would not be short. For this reason the old and new applications would have to co-exist for a period of time until the migration was complete. The management applications were enhanced to prevent changes to metadata for files in the migration pipeline. The delivery applications were enhanced to be able to deliver content from both DRS versions and automatically retrieve metadata from the new DRS for migrated content, permitting uninterrupted use of the content by researchers during the migration.

The last key requirement was that all the identifiers (persistent URNs and Oracle IDs of the files) needed to remain valid post-migration. Keeping the same Oracle IDs made it easier to compare metadata between the two DRS databases during migration testing and verification, and kept us from having to change any of the delivery URLs that the URNs resolved to. We

were able to keep the same Oracle IDs for the files by starting the database sequence in the new database at a very large number so newly assigned IDs would not clash with the files needing migration.

The Plan

A DRS Advisory Group that included representatives of DRS depositors and content owners was established to provide guidance and help with the rollout phase of the DRS2 project. Members of this group helped lead meetings to explain and get feedback on the metadata migration and user adoption plan. This group updated the larger Library community regularly on the status of the project and migration primarily through the use of the Library's email newsletter.

The results of the separate analyses were combined into a single migration plan, designed to address both the technical and human factors described earlier. The migration would be conducted in five "tiers" which mapped to different kinds of DRS content (See Table 1). It took into consideration the technical need to build, test and optimize the migration software on simpler objects before migrating more complex objects (e.g. objects without relationships before objects with potentially many relationships), as well as dependencies between different kinds of content (e.g. color profiles had to be migrated before the still images referencing the color profiles).

Table 1: The overall migration order was based on content type

Tier 1	Text methodology, Color profiles, Target images, ESRI world files, PDF documents
Tier 2	Page-turned objects, Still images
Tier 3	Audio, Audio playlists
Tier 4	Web harvests, Opaque containers
Tier 5	Google-scanned books, Biomedical images

Within each tier, the metadata for all DRS content owners would be migrated at the same time, except for Tier 2. The content associated with this tier - page-turned objects and still images - needed to be treated differently for a couple reasons. This metadata, along with structure in the case of page-turned objects, was actively managed by many DRS content owners and depositors. They typically changed descriptive metadata such as display labels and image captions, or merged page-turned objects into a single presentation for series or multi-volume monographs. Also in terms of numbers this content vastly dominated all other content in the DRS. This migration tier would take longer than any other, and because this content was frequently managed, it would have been very disruptive to migrate it across all owner codes at one time. Because the two versions of the DRS use different data models and metadata schemas there are specific management applications tied to each. A key goal in the migration plan was to minimize the amount of time that library staff needed to use the old and new management tools at the same time. For this reason the Tier 2 part of the migration would be performed on the content of one Harvard library, archive or museum at a time to minimize the amount of time any particular unit had to use both versions of the management applications while their content was being migrated.

A few "pioneers" were helped to switch over to the new DRS soon after the new software and hardware were in place in October 2013. These early users gave the support team an opportunity to develop the process for facilitating the switch-over process and helped uncover some technical bugs in the new tools. For the bulk of the depositors and content owners however they were asked to hold off using the new DRS until after their unit's Tier 2 migration. This allowed the support team to focus their attentions on a smaller number of units at one time.

Technical Design

The migration application was architected and developed as three modular components (selection, descriptor-building and ingest) intended to be run in parallel as shown in Figure 3. A key piece of this architecture is Oracle Advanced Queuing (AQ) [7], a message-based technology built on top of the Oracle database software. Oracle AQ made possible the needed parallelism to perform the migration in less time.

In the selection phase an Oracle database is queried to assemble the files into objects and a unique ID is written to the database for each object. In the descriptor-building phase, metadata is gathered from many sources, including running the files through FITS, and a METS XML object descriptor is written for the file. In the ingest phase, the metadata is written to a new Oracle database schema, additional metadata is written to the object descriptor, the descriptor metadata is written to a Solr index and the descriptor is written to preservation storage to be replicated.

The migration code uses a multithreaded model. Worker threads managed by a main execution thread listen indefinitely for messages to be added to the queues. The main execution thread is able to shutdown the migration application cleanly when a developer or system administrator executes a kill command [8] to terminate the migration for any reason.

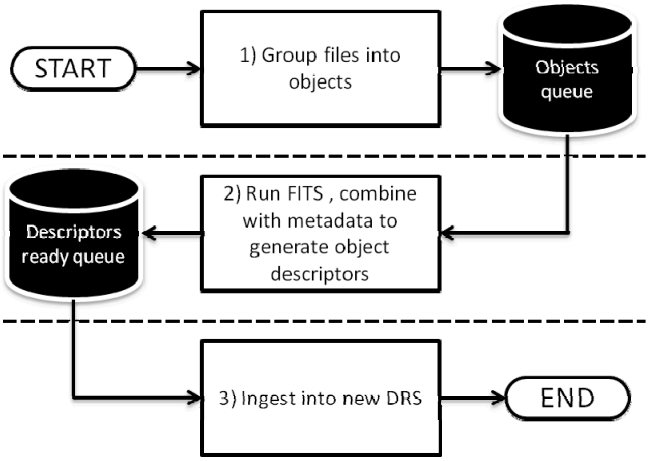

Figure 3: The core of the metadata migration has three steps ("selection phase", descriptor-building phase", and "ingest phase") that are run in parallel using 2 Oracle AQ queues storing file and object identifiers ready to move on to the next stage.

The migration application was designed to be portable so that it could be run on any hardware. Each of the select, descriptor-building and ingest components can be run on multiple machines, and on each machine multiple threads can run. The thread count for each of the three components is separately configurable. The intention was to test the migration in different configurations to find an optimal thread and machine count for each of the three components.

New fields were added to the old DRS database to track the migration status of each file and any errors that occurred. An element in the new DRS administrative metadata, "administrative flags", was found to be especially useful for the migration. This element had been originally designed to record events that require repository staff confirmation or intervention, for example to document a detected virus or social security number. For the migration it was used to note cases where automatically determined metadata such as MIME media-types did not match what had previously been recorded for a file. These could indicate incorrect metadata and warrant attention in the future.

Results

The migration software was tested end-to-end and several bugs were found. One of the largest bugs found was that multiple instances of the FITS tool were not able to run concurrently within the same JVM because of static variables used by FITS and DROID [9], a tool wrapped by FITS. After this and other bugs were fixed the migration benchmarking began in December 2013.

First, a small test of 57 PDF documents was conducted on a test machine. It was already known that the selection phase for this content was very fast (less than a second for all of the PDFs together) so this test focused on different thread counts for the descriptor-building and ingest phases. The results, shown in Table 2, show that the ingest component can keep up with the descriptor-building component, in fact it appears to be throttled by the descriptor-building component. This was not a surprise, but confirmed that the maximum load balancing should be done for the descriptor-building component.

Table 2: Timing for different thread counts (TC) for different migration components (Sel = Selection, Des = Descriptor-building, Ing = Ingest)

Sel TC	Des TC	Ing TC	Sel time	Des time	Ing time	Total time
1	3	3	1 s	23 s	23 s	24 s
1	10	3	1 s	16 s	16 s	17 s
1	10	10	1 s	13 s	13 s	14 s

In a follow-up test, again on the same 57 PDF documents, thread counts of 5 and 10 were tested for the descriptor-building phase. A more powerful machine with 16 cores and 64 GB memory was used this time. The results, shown in Table 3, show that doubling the thread count from 5 to 10 did not make a noticeable difference.

Table 3: Comparison of timing for different thread counts (TC) for the descriptor-building phase (Sel = Selection, Des = Descriptor-building, Ing = Ingest)

Sel TC	Des TC	Ing TC	Sel time	Des time	Ing time	Total time
1	**5**	1	2 s	7 s	7 s	9 s
1	**10**	1	1 s	7 s	9 s	10 s

In a larger test, 4,736 text files and 25,447 PDF documents were put through the migration pipeline using the same powerful 16 core machine. The results confirmed that the selection phase is relatively fast, e.g. a little over a minute for all of the content. Using 5 threads running in parallel, the descriptor-building phase could on average process 35 text files per second but only 4 PDF documents per second. Looking closer at the PDF documents it was found that the descriptor-building time varied from a low of .8 PDF documents per second to a high of over 9 PDF documents per second. The difference is that some of the PDFs in the DRS are very large, e.g. PDFs of serials running thousands of pages. For this particular test the ingest portion of the migration was not run since it had already been found to be very fast in earlier tests.

Discussion

This is a case study on a very complex large-scale metadata migration. While there have been examples of other repositories migrating metadata as a result of changing their packaging format or schemas, this is the only known example of migrating metadata for many different types of content, pulling metadata from various sources including the files themselves, and while there is an active user base managing the metadata and accessing the content. In these regards this migration project is treading new ground.

Although the migration is not complete it can already be concluded based on the tests described in this paper that the migration can be completed in less than one second per file using a modular migration architecture and parallelizing the components that take longest. The development team will continue to look for additional optimizations, for example to add additional machines to the migration.

Other findings include the need to either automate more metadata contribution or put into place more validation of user-provided metadata where possible to improve the quality of the metadata. In addition, repository metadata should be periodically reviewed with key stakeholders to make sure that the value of the metadata warrants the effort to create it.

It is hoped that the lessons learned in this project could be applied to use cases in addition to mass metadata migration, for example to large-scale processes such as format migrations or mass format identification. In addition, the user-based planning and design may be of interest to those who may need to do a similar retraining or transitioning of library staff in the future.

References

[1] Harvard Library, "LDI Overview : LDI : Library Information Services", Available at http://hul.harvard.edu/ois/ldi/.

[2] METS Editorial Board, METS Metadata Encoding & Transmission Standard, Library of Congress website. Available at http://www.loc.gov/standards/mets/

[3] Consultative Committee for Space Data Systems, Reference Model for an Open Archival Information System (OAIS), Magenta Book, issue 2 (Washington, DC 2012).

[4] PREMIS Editorial Committee, PREMIS Data Dictionary for Preservation Metadata, version 2.2 (2012).

[5] Harvard Library, File Information Tool Set (FITS). Available at fitstool.org

[6] S. Morrissey, V. Cheruku, J. Meyer, M. Stoeffler, W. Howard, S. Kadirvel, Migration at Scale: A Case Study, iPRES 2012 (2012).

[7] Oracle, "Introduction to Oracle AQ", Oracle Streams Advanced Queuing User's Guide 11g Release 2. Available at http://docs.oracle.com/cd/E11882_01/server.112/e11013/aq_intro.htm

[8] The Linux Information Project, "The kill Command", (2006). Available at http://www.linfo.org/kill.html

[9] The National Archives, “File profiling tool (DROID)”, Available at http://www.nationalarchives.gov.uk/information-management/our-services/dc-file-profiling-tool.htm

Author Biography

Andrea Goethals is responsible for providing leadership in the development and operation of Harvard's digital preservation program and for the management and oversight of the Digital Repository Service (DRS), Harvard's large scale digital preservation repository.

Franziska Frey is the Malloy-Rabinowitz Preservation Librarian for the Harvard Library. She is responsible for shaping the strategic direction for preservation, conservation and digitization initiatives to ensure long-term access to all collections with the goal to create a seamless continuum for the long-term preservation of traditional collections and digital content across the Harvard Library.

Robin Wendler, Senior Metadata Analyst for Harvard Library Technology Services, specializes in metadata design, metadata standards development, digital preservation concerns, development of functional requirements for library and archive information systems, systems analysis, and workflow design.

Chris Vicary, Senior Digital Library Software Engineer at Harvard University, is the lead developer for Harvard Library's long-term digital preservation repository (the DRS) and for the multi-year DRS enhancement project (DRS2).

Spencer McEwen, Senior Software Engineer at Harvard University, previously was the lead developer for the DRS, DRS2 project and several key file processing and deposit tools that support the DRS such as the File Information Tool Set (FITS), the Object Tool Set (OTS) and Batch Builder.