When a Repository Is Not Enough

» Redesigning a Digital Ecosystem to Serve Scholarly Communication

BY ROBIN R. SEWELL, SARAH POTVIN, PAULINE MELGOZA, JAMES SILAS CREEL, JEREMY T. HUFF, GREGORY T. BAILEY, JOHN BONDURANT, SEAN BUCKNER, ANTON R. DUPLESSISL, LISA FURUBOTTEN, JULIE A. MOSBO BALLESTRO, IAN W. MUISE, AND BRIAN J. WRIGHT

INTRODUCTION

Over the past decade, under the banners of scholarly communication, special collections, collection development, and digital scholarship, academic libraries have taken on greater responsibilities for collecting, publishing, and preserving a range of digital assets. These include the creative and scholarly output of libraries' host institutions, digitized or born- digital general collections holdings, and a diverse array of materials from special collections and archives.

Broadly, "digital asset management consists of management tasks and decisions surround- ing the ingestion, annotation, cataloguing, storage, retrieval, and distribution" of image, multimedia, and text files (Wikipedia, 2017). These tasks rely on digital asset management systems (DAMS) that are either commercial products, open-source systems (out-of-the-box or customized), or entirely homegrown to meet diverse needs. DAMS differ in their ap- proach to functions and offer a range of associated capabilities.

The infrastructure for managing an academic library's digital assets might include DAMS oriented toward scholarly publication needs, deployed in the form of institutional or data repositories, or archival and special collections needs with modules aimed at display and exhibition. Base infrastructure requirements for scholarly publication have expanded as libraries' scholarly communication programs have extended to inculcate openness across the entire research life cycle, encompassing or allying with digital scholarship and digital humanities, data curation and management, library publishing, and evaluation metrics.

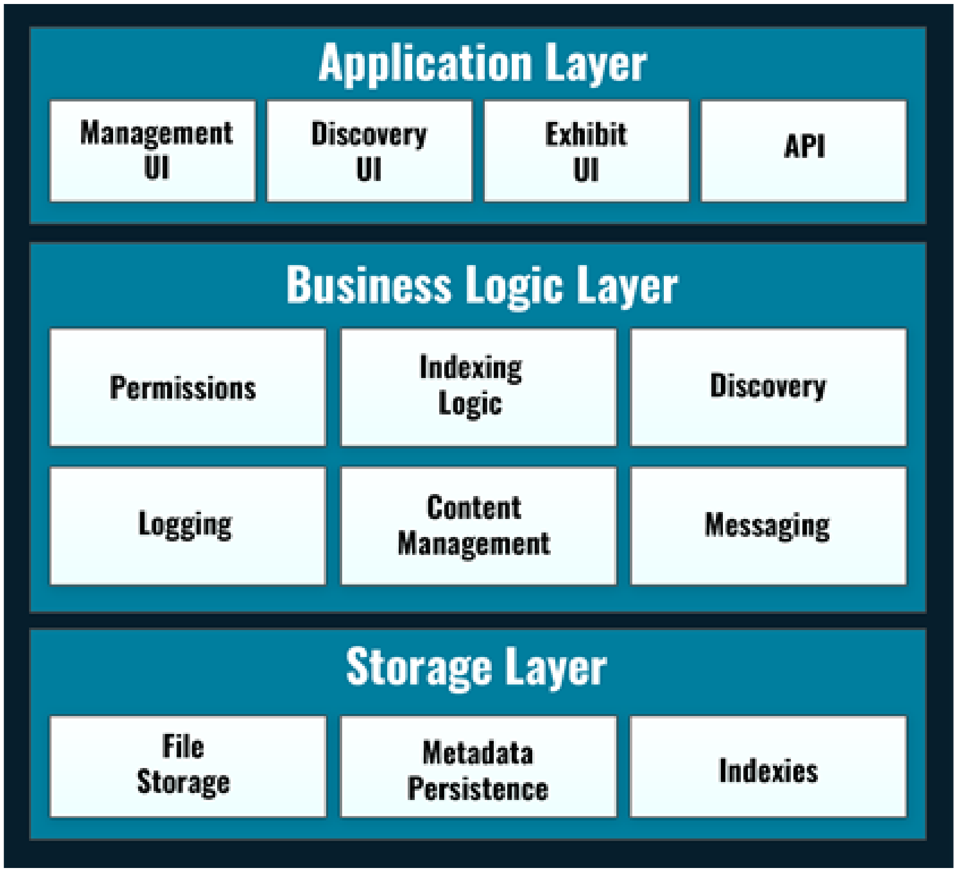

Figure 1. Generic DAMS implementation

At our institution, an increased range of scholarly communication and digital scholarship publishing services had been shoehorned into an institutional repository infrastructure: the libraries' existing DAMS, DSpace, had been in use since 2004 (Maslov, Mikeal, & Leggett, 2009). "Scholarly communication" in the library had been broadly construed to include much of the library's digital collection development and management, with an emphasis on open access. Although the libraries had a long-standing commitment to the open source DSpace community, and while recent upgrades had further enhanced the capabilities of DSpace, its flat metadata structure and prescriptive data modeling made representation of complex objects difficult to achieve. Outside tools, such as book readers, were incorporated to fulfill the libraries' display and online exhibit needs. Efforts over the years to integrate these tools into the DSpace interface had become unsustainable, with these integrated com- ponents requiring extensive mending and rebuilding with every DSpace upgrade. Increased interest in storage and display of streaming video content, geographic information system (GIS) data, and 3D specimens could not be accommodated by the DSpace version then in use (DSpace 4.0).1 In short, a tool designed for sharing preprints was not ultimately well suited for managing preservation workflows and

» **The Digital Asset Management Task Force (TF) was created in August 2014 and charged with investigating and making recommendations for a solution or solutions that would en- able the libraries to store, display, and preserve new forms of university information and research, including digital scholarship, special collections, and archives.**

complex digital library holdings.

Simultaneous to our local realization, an international community effort to forge a collec- tive "DSpace Vision" for the platform emphasized the need to "focus on the fundamen- tals of the modern 'Institutional Repository' use case." This community-supported vision pledged that DSpace "will be designed in such a way that it can be easily/quickly configured to integrate with new and future tools/services in the larger digital scholarship 'ecosystem'" (Donohue, 2014, DSpace 3-5 Year Vision Statement).

In our library, these twin promises—of a closer emphasis on institutional repository functionality (rather than broad digital library or asset management design) and the potential to integrate with other systems—positioned DSpace as a likely component among other tools fulfilling our diverse digital needs. These factors, coupled with strategic hiring that forged a cross-unit emphasis on digital collection-building and preservation, prompted a reevalu- ation of our strategy of using DSpace for any and all university-generated open access content. We needed to look for a more robust, complementary DAMS to meet the existing and projected needs of the university.

The Digital Asset Management Task Force (TF) was created in August 2014 and charged with investigating and making recommendations for a solution or solutions that would en- able the libraries to store, display, and preserve new forms of university information and re- search, including digital scholarship, special collections, and archives. The TF was instruct- ed to evaluate DAMS products and identify an optimal solution. Our scope was limited to evaluating the suitability of existing commercial and open-source DAMS—evaluation of exhibit layer software and development of workflows and policies would be undertaken by separate task forces. Perhaps unusual in the charge was the instruction that our recommen- dation should attempt to report but not weigh cost. Members of the TF were drawn from all areas of the library, including user services, special collections, cataloging, digital initia- tives (the libraries' IT group), medical libraries, preservation, and scholarly communication. There were special challenges associated with reviewing open source software and with com- paring commercial and open source DAMS costs and capabilities. As Woods and Guliani (2005) argue, open source software is difficult to evaluate. Commercial software vendors invest in marketing and communicating functionality and benefits in ways that open source communities do not; open source tools must be assessed through installation and testing. While commercial solutions may come with a specific price tag, open source costs are more elusive and tied to local IT staffing. Woods and Guliani (2005) observe,

> *With an open source program it is far more likely that an IT department will have to solve an integration or customization problem on its own. It's hard to generalize about whether this is a strength or a weakness of open source. . . .*
>
> *Anything can be done with open source, so the barrier to creating the optimal system for supporting a business process is often lower. (pp. 73-74).*

This article presents two models: (1) a process for identifying, selecting, and evaluating open-source and commercial DAMS; and (2) a "digital asset management ecosystem" (DAME) approach to technical infrastructure that comprises a distributed, linked set of open source platforms.

LITERATURE REVIEW

The TF scoured DAMS articles and case studies from academic institutions to (1) generate an exhaustive list of current commercial and open source system digital asset systems and analyze and adapt DAMS needs assessment and selection processes. This environmental scan provided the basis of our assessment methodology.

Finding the elusive perfect DAMS fit requires both an analysis of institutional need—per- taining to content and collections, users, and administration—and available tools. Some reports bundled these analyses; others were oriented toward either needs or tools. The prac- tical literature revolves around two main scenarios: institutions starting from scratch and those that have outgrown their current DAMS and are looking to migrate to a new system or systems with increased functionality. The University of Utah's exceptional report and webinar document their robust review process, criteria, and a DAMS scoring model, ulti- mately recommending a migration from CONTENTdm to Hydra (Masood & Neatrour, 2014). We adapted their model for our own testing. Stein and Thompson (2015) provide a metareview of DAMS migration studies in their analysis of motivations, observing a ten- dency of institutions to move from proprietary to open source systems ("primarily Islan- dora, Hydra/Fedora, and DSpace") (section 4.2, para. 3). Michigan State's analysis unfolds in an environment without a "comprehensive, campus-wide digital preservation strategy" or institutional repository (Schmidt, Ghering, & Nicholson, 2011, p. 106). Their "digital curation planning project to explore and evaluate existing digital content and curation practices" (p. 110) issued in the early stages of identifying digital content and developing policies and procedures, and focused as it was on assessment, included a detailed survey that was sent to out to their campus. The National Library of Medicine's report evalu- ates 10 commercial systems and open source software programs (NLM Digital Repository Evaluation Selection Working Group, 2008). Of particular interest to us was the in-depth test of the final three systems and NLM's selection of Fedora on the basis of its flexibility, active development community, and open source code. A

Table 1. DAMS considered for evaluation

DAMS	License	Developed by	Website
ArchivalWare[a]	Proprietary	PTFS	http://www.archivalware.net
CONTENTdm[a,b]	Proprietary	OCLC	http://www.contentdm.org
Cumulus	Proprietary	Canto	https://www.canto.com/cumulus/
DAITSS[a]	Open Source	FCLA	http://daitss.fcla.edu
DigiTool[a,b]	Proprietary	ExLibris	http://www.exlibrisgroup.com/category/ DigiToolOverview
DSpace 5[a,b]	Open Source	DuraSpace	http://www.dspace.org
Eprints[a]	Open Source	U. of Southampton	http://www.eprints.org/uk/
Fedora[a,b]	Open Source	DuraSpace	http://www.fedora-commons.org
Greenstone	Open Source	New Zealand Digital Library Project	http://www.greenstone.org
Hydrab	Open Source	DuraSpace	http://projecthydra.org
Invenio	Open Source	Invenio Software (CERN)	http://invenio-software.org
Islandoraa	Open Source	DuraSpace	http://islandora.ca
Keystone DLSc	Open Source	Index Data	http://www.indexdata.com
KORA	Open Source	Michigan State Univ.	http://kora.matrix.msu.edu
Luna	Proprietary	Luna Imaging	http://www.lunaimaging.com/ software
MDIDd	Open Source	James Madison Univ.	https://mdid.cit.jmu.edu
Mnesysb	Proprietary	Naoned Systemes	http://www.mnesys.fr
Nuxeo	Hybrid	Nuxeo	http://www.nuxeo.com
Omekab	Open Source (exhibit only)	George Mason Univ.	http://omeka.org
ORIOAIb	Open Source	National Consortium	http://www.ori-oai.org
ResourceSpace	Hybrid	ResourceSpace	www.resourcespace.org
Shared Shelf	Open Source	ARTSTOR	http://www.artstor.org/sharedshelf
TACTICa	Hybrid	Southpaw	http://www.southpawtech.com/tactic
VITALd	Proprietary	VTLS	http://www.iii.com/products/vital
XTFb	Open source	CDL	http://xtf.cdlib.org
Yoolib	Proprietary	Amanager	http://my.yoolib.com/demo

Note: These DAMS existed at the time of testing. Some may no longer be available.
[a]NLM Digital Repository Evaluation Selection Working Group (2008). [b]Andro, Asselin, and Maison- neuve (2012). [c]Schmidt et al. (2011). [d]Masood et al. (2014).

2016 article from the University of Houston's (UH) DAMS Implementation Task Force discusses their needs assessment, systems evaluation, and testing (Wu, Thompson, Vacek, Watkins, & Weidner, 2016); while the UH article was published subsequent to the TF's review and recommendation, conversations with our colleagues at UH improved our approach.

Related literature offered insight into the emergent ecosystem model of digital asset man- agement. A University of California (UC) System report details "technical and philosophi- cal" goals for DAMS development, emphasizing modularity, principles of service-oriented architecture, and the selection of "best of breed components with open source tendencies that have broad adoption and community support" (Grappone, Fleming, Hetzner, Perry, & Tingle, 2013, p. 3, 2). In service to their goal of implementing a "progressive model for a system wide DAMS," the UC System selected Nuxeo, an open source product with vendor support, as an immediate, interim solution (p. 2). A recent article on UH's implementation delves further into their workflows and DAMS architecture—"an ecosystem of modular components" —deployed and developed to support access and preservation of the libraries' digitized cultural heritage holdings (Weidner et al., 2017, Bayou City DAMS Ecosystem, para. 1).

Given the rapid changes in digital asset management design and approaches, and the nature of the systems dominant in cultural heritage institutions (whether open source or com- mercial, these solutions engender robust communities of practice), published reports and articles represent a small fraction of what might be described as relevant scholarly analysis and frameworks. These reports and articles are foundational, situate our work in a larger context, and provide adaptable models of assessment. But in constructing a fuller under- standing of assessment approaches and system options, the TF additionally benefitted from myriad conference presentations, hallway conversations, shared internal documentation, and phone calls with colleagues at other institutions invested in digital asset management needs and systems assessment.

DESCRIPTION OF PROCESS

The task force installed products and evaluated their capabilities against a task-based rubric of essential features, gleaned from internal library and university needs assessments, using representative test collections. Here, we include lessons learned that will help libraries un- dertaking similar evaluation processes to serve scholarly communication and digital service needs. We also discuss obstacles encountered in our attempt to test complex systems that may not have all necessary features available in their vanilla, out-of-the box implementation. Both our findings, detailed in data supplementary to this article, and relevant assess- ment tools have been deposited in the Texas Data Repository.

PRELIMINARY WORK

Needs Assessment

Criteria for the DAMS and the selection of testing document types were informed by two needs assessments. The first was an internal library needs assessment conducted by the li- braries' scholarly communication unit in 2013 through informal meetings with each library unit or department to assess current interest and potential projects. The second, more for- mal needs assessment was a campus-wide survey initiated in response to interest expressed by other university units in the creation of a campus-wide asset management solution (IRB 2017-0744). Several units on campus had either brought up a DAMS or were looking into bringing one up to manage their locally created digital assets. Business cases—such as managing university marketing departments' assets—were included alongside university- wide scholarly communication and research needs. Campus entities voiced an interest in a campus-wide system, designed and hosted by the library, which would serve everyone's needs. The TF charge was adjusted to meet the broader scope, and members were added from outside the libraries. A survey was created and sent to representatives at the various campus units to gauge interest and gather data on current and future needs, including space, files types, preservation requirements, and access restrictions.[2]

Identification and Review of Available DAMS

The TF began by conducting literature reviews and environmental scans, as discussed above in the literature review, to investigate current digital asset systems and review DAMS need/selec- tion assessment processes at a number of peer institutions. Consultations with and documen- tation provided by the University of Utah, University of Houston, and Penn State University were particularly helpful, as were reports out of Michigan State University, the University of California, and the National Institutes of Health. Based on this research, the TF was able to scope beyond the commercial and open-source systems most familiar to libraries, generating a list of 25 possible systems (Table 1). Members of the TF reviewed each of these systems in depth to identify the license type (open source, proprietary, or hybrid), the organizations re- sponsible for development and management, the institutions that used the systems, the pres- ence or absence of an active development community, and additional anecdotal information from articles, case studies, or conversations with users of the systems. DAMS were eliminated from consideration based on lack of community support, lack of active development, or ab- sence of an English-language interface or a North American user community **(see Table 1)**.

Members of the libraries' IT unit further evaluated 17 of the most promising DAMS against a matrix of features to determine if the DAMS would be compatible with other programs, programming languages, and software used throughout the libraries (e.g., Java-based) and were likely to be successfully integrated into a networked digital asset ecosystem model. IT evaluated each system using a six-point Likert scale (0 least, 5 most) on existing institutional knowledge, application programming interface (API), discovery (ability to search within the DAMS), documentation, community health (size and activity of support community), and development health (ongoing development and new versions). The IT matrix is available in the supplemental data.

SELECTED DAMS

Based on the TF evaluation results and the IT matrix scores, the TF selected four systems for testing. Two systems—Islandora and Hydra with Sufia and Blacklight (Hydra/Sufia)—were open source, and two were hybrid commercial and open source options— ResourceSpace and Nuxeo. DSpace, the libraries' current DAMS, was added to the test group as a delta, the minimum standard of functionality. In order to serve as a minimum standard and not be given an unfair advantage, DSpace 5.5 was also tested as an out-of- the-box deployment without any of the enhancements and customizations found in the libraries' DSpace 4 instance.

Our nascent digital ecosystem approach, described in greater detail later in this article, opened up the possibility of modular development and supplementing DSpace with ad- ditional tools and services that, deployed in the distributed service architecture, might bridge functionality gaps and extend DSpace's capabilities. Every system selected boasted a robust API, broad adoption, strong community support, and the ability to function as either a modular component in a DAME or a standalone DAMS (including a range of functionality and support for user management, display, indexing and discovery, built in statistics, etc.).

EVALUATION PROCESS

The TF decided to pilot each system individually and sequentially with a common rubric using multiple predetermined sets of sample content containing various types of files based on use cases gathered from the campus needs assessment survey as well as library-based needs. The test bed included simple files, metadata files, complex related objects, and AV content.

Initial Rubric & Scoring

The initial rubric (long rubric or LR) consisted of over 200 tasks of varying complexity. The rubric tasks or functions were grouped into the following eight sections: Inputting and Structuring Content, User Management, Ticket/Request/Workflow, Statistics and Report- ing, Discovery, Relational Linking, Presentation, and External Systems.

The TF recognized that in most cases, the Relational Linking and External Systems catego- ries would require research to determine feasibility of implementation rather than actual testing; however, these categories contained important components for existing and future scholarly communication needs—including Archivematica, Shibboleth, ORCID, VIVO, and Plum Analytics integrations in our larger scholarly communication ecosystem—and warranted investigation and scoring.

Each TF member was assigned multiple subsections of the rubric to test across all systems using the defined test collections. Task assignments were based on TF member experi- ence and expertise, and each task was tested by two or three TF members. Members were instructed to limit testing and evaluation to 20 minutes per task and grade each task on ease of completion using a scale of 0 (low score, not possible) to 3 (high score, easily com- pleted). If a task could not be completed using our implementation, additional investiga- tion was

Table 2. Summary of Long Rubric and Short Rubric scores

	Long Rubric (717 points)		Short Rubric (48 points)	
DAMS	Total Points	Percentage	Total Points	Percentage
DSpace	321	44.77	27	56.25
Islandora	263	36.68	28	58.33
Hydra/Sufia	306	42.68	18	37.50
Nuxeo	—	—	43	89.58
ResourceSpace	—	—	32	66.67

conducted using DAMS documentation and other sources to determine if the task was feasible with additional configuration or development. A score of "C" was used to denote that the task was possible with configuration or local or community development, and a "T" (time out) denoted that a solution was not found within the allotted 20 minutes. Notes were gathered in the spreadsheet to help members testing the same task communicate with each other and keep track of research to help determine if a task could be configured.

Deployment

The systems were developed, deployed, and tested, generally in one-month intervals, in the following order: DSpace, Islandora, Hydra/Sufia, Nuxeo, and ResourceSpace. For the first three pilots, library IT developed a sandbox/test environment for TF members, providing them with accounts/logins and technical support, when necessary. However, because of technical problems with our local deployment of Islandora, many tests were performed on a sandbox hosted by Islandora3 rather than our local test instance. TF members completed all assigned sets of tasks using the rubric and the predetermined sets of sample content.

Obstacles

The TF encountered two obstacles after testing was completed on DSpace, Islandora, and Hydra/Sufia. The TF was under pressure to wrap up testing as it neared the end of its second year. This deadline limited the ability of the TF to deploy the open source versions of Nuxeo and ResourceSpace. In addition, during a conference call with the UC Digital Library (UCDL) group, the TF learned that key components of Nuxeo's functionality were available only through the vendor's subscription service, and not included in the open source version. UCDL also discussed the potential for steep escalation in annual fees associated with the vendor-based solution. Although the TF was charged with evaluating DAMS without considering cost, this news raised concerns for Nuxeo's viability as a DAMS candidate.

Because Nuxeo could only be evaluated through a demonstration by the vendor, and be- cause of time constraints, the TF created a short rubric (SR) to assess Nuxeo and the re- maining DAMS for testing, ResourceSpace. The SR consisted of 24 criteria with a possible score of yes, no, or partial. A partial score was used to indicate that the feature was not currently implemented, but could be implemented without too much difficulty, or was currently available but lacking in some desired components.

Reconciling Rubrics

Having three systems graded using the granular, task-based LR and two evaluated using the criteria/feature-based SR made comparisons between the systems problematic. After exploring the possibility of mapping scores between the long and short rubrics, the task force ultimately rescored DSpace, Islandora, and Hydra/Sufia using the SR, to provide a consistent method of comparison with Nuxeo and ResourceSpace. The LR remained useful for comparing DSpace, Islandora, and Hydra/Sufia.

EVALUATION SUMMARY AND RECOMMENDATIONS

Rubric Scores

The LR and SR rubric scores were converted to numeric values to facilitate comparisons of the DAMS. Each LR task had a maximum value of 3 points. Assigned numeric scores were taken at value, each C score had a value of 1, and each T score a value of 0. The final LR task score was the average of the individual member scores. SR yes, no, and partial feature scores were assigned point values (yes = 2 points, partial = 1 point, no = 0 points). **Table 2** shows the summary scores for both the LR and SR (detailed results are available in the supplemental data).

That none of the DAMS evaluated using the LR achieved 50% of the total possible points may be an indication that our LR rubric was overly ambitious or that the DAMS were resistant to out-of-the-box testing. The SR scores for the two commercial products, Nuxeo and ResourceSpace, were much higher than the SR scores for the three open source DAMS (DSpace, Islandora, and Hydra/Sufia). The differences in scores could indicate the presence of capabilities that are more fully developed in the commercial applications, but must be configured or developed in open source systems. While Nuxeo is the clear winner based on the SR evaluation, the cost made it a less attractive solution.

Qualitative Impressions

Our evaluation allowed us to quantify and visualize each DAMS' ability to provide needed functionality and to potentially complement the libraries' existing DSpace instance with an eye toward the implementation of a DAME and the ability to expand to meet growing campus-wide scholarly communication needs. At the end of the evaluation process, the TF members provided their overall impressions of the DAMS gather during testing.

Islandora

The TF had mixed results and feelings about Islandora, including its reliance on Drupal as an interface and the inability to authenticate and set granular permissions. It also did not score as well as Hydra/Sufia on the LR, and had a higher number of configuration and time-out scores than either DSpace or Hydra/Sufia.

Hydra/Sufia

While Hydra/Sufia is backed by a large and engaged community and had an intuitive and well-designed user interface, the tested version of Sufi—Sufia 6—also lacked metadata versioning and the ability to authenticate and set granular permissions.

Nuxeo

Nuxeo scored well on the SR, had many of the desired features, and would have enabled rapid deployment of a DAMS; however, these positive aspects were outweighed by the ongoing and potentially increasing cost of the vendor model that includes Nuxeo Studio.

ResourceSpace

ResourceSpace was of interest because it was in use by some campus groups and would have facilitated content sharing. Unfortunately, ResourceSpace did not support structured meta- data, which is an essential feature in a DAME component to supplement DSpace.

Fedora 4

At the end of the review process, TF members found that they appreciated the functional- ity provided by Fedora, which underlies both Islandora and Hydra/Sufia. While it is not a full freestanding DAMS, it provided access to many desired features, including support for complex and hierarchical metadata, linked data capabilities, and the ability to function well as a component of the DAME. Fedora's strengths include the following:

- Has a robust development community, under the umbrella of DuraSpace (with some possibility of integration with VIVO and DSpace)
- Forms the basis of several popular open-source DAMS, including Hydra and Islandora
- Is a flexible object model that is complementary to DSpace's more constrained model
- Implements the Linked Data Platform W3C recommendation with support for RDF expression
- Has built-in durability functionality
- Implementation draws on local strengths with Java development

Fedora's weaknesses include the following:

- Requires a significant investment of developer time and support, potentially in addition to the contracting of support teams like the Data Curation Experts group4
- Requires community investment to gain fluency (including attending Fedora users group meetings and Fedora Camps)

The TF identified several ways to extend DSpace functionality that would allow it to serve as an interim solution while Fedora 4 and the DAME are implemented. Video capability of our current DSpace could be extended by installing new video streaming tools developed at Virginia Tech. The need for completely private deposits, not visible to anyone, would be handled by the use of private status in DSpace, depositing those items directly in Archive- matica, or bringing up another instance of DSpace for dark storage.

LESSONS LEARNED

In our search for a DAMS that was just right, we faced challenges in designing testing protocols and encountered technical options with complex, multifaceted implications. Our extensive research and testing also positioned us to discover system functionalities outside of our initial set of use cases and needs.

EVALUATION CHALLENGES

A core goal was the generation of data on DAMS under consideration as the basis for an evidence-driven decision. We knew from experience that advertised features—even in community-supported open source systems—didn't always function as promised. The TF developed a set of requirements formed around a community needs assessment, designed an extensive task-based testing protocol with multiple testers (as the basis for establishing and accounting for reliability), and supplemented task-based testing with research-based test- ing and unstructured interviews with current users of the systems under consideration. But the DAMS themselves, each of which included constantly evolving features, complicated this robust protocol. By deploying out-of-the-box versions of DAMS, we may have been inadequately attentive to features that had not yet been folded into core code. Additionally, despite our investment in testing, the TF was aware that task- and research-based inquiries were potentially inadequate substitutes for community embeddedness: in short, owing to incomplete documentation and distributed user networks for these products, it was impos- sible to get a full picture of capabilities simply through research and testing. Our selection of Fedora and preferencing of an ecosystem model (described below) served, to some extent, to compensate for the barriers to a total evaluation of current and potential functionality: by emphasizing components over an all-in-one system, we have broken down some of the po- tential complexity of the latter in favor of more easily evaluated and certainly more closely scoped elements.

DEPLOYMENT TRADE-OFFS

During our review and testing, the TF observed a trade-off between ease of deployment and flexibility. While all-in-one, out-of-the-box systems enable rapid deployment and mini- mal investment in IT personnel time, their ease of use is accompanied by inflexible data models and approaches that limit their functionality and make them cumbersome to use. Conversely, the systems with the greatest flexibility and range of functionality require con- siderable IT time to deploy and near-constant maintenance. Additionally, we observed the potential necessity of configurations that curtail the flexibility of DAMS in order to frame a more usable, interoperable platform: for example, highly flexible Fedora implementations often employ relatively prescribed data modeling or rigid administrative interfaces that limit range of use. This observation affirms the design principle of a "Flexibility-Usability Trad- eoff," which dictates that "flexibility has real costs in terms of complexity, usability, time, and money" (Lidwell, Holden, Butler, & Elam, 2010, p. 102).

Task Force Size and Composition

The TF was a large committee with representatives from across the libraries and two individ- uals from university units. While it is essential to have feedback from the represented units, a smaller, more focused group that interacted periodically with library units and interested outside parties would have been more agile than the large committee. The LR evaluation process was time intensive, and having a task force composed of individuals with dedicated time set aside for the evaluation rather than having TF duties added to already busy schedules and heavy workloads would have helped move the evaluation process forward more rapidly. The inclusion and active participation of a member of the libraries' IT team was crucial to the evaluation process and provided needed insight into the potential of a DAME and the possibility of integrating DAMS with other library software systems. However, this may have introduced some bias or path dependency, as existing IT strengths and skills sets were considered in the DAMS selection process.

Rubrics

Testing with two different rubrics was not an ideal situation. The LR provided a lot of in- formation, but may have been overly complex. It was time consuming to test and score each DAMS. The detailed testing and configuration notations used in the LR turned out to not be as helpful when the information was consolidated to create a final score. The LR did reveal is- sues that would not have been discovered with SR—issues with metadata handling, version- ing, and multipart objects. However, the changing

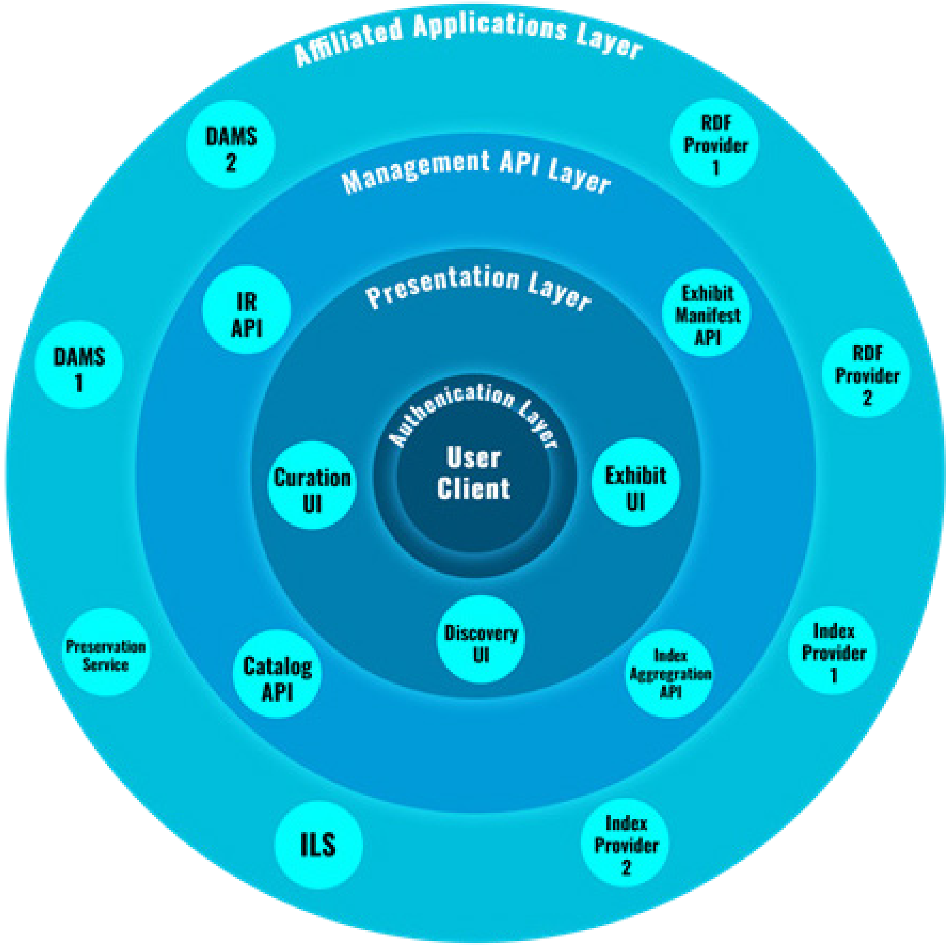

Figure 2: Vision for DAMS service integration

nature of the open source systems tested made complete functional deployment of out-of-the box systems challenging. For example, complications with Islandora's deployment led to testing on the existing Islandora sandbox rather than our own implementation, and local knowledge of DSpace informed us that out- of-the-box required adding SWORD to enable file upload functionality.[5]

The SR criteria meant that DAMS did not need to be fully deployed to be evaluated. Giv- en the resistance of the DAMS to local testing, it was, in some ways, a better to fit to have a short impressionistic rubric rather than a long, task-oriented one. It would also be useful in a situation where it was not feasible to deploy a DAMS for detailed testing, like the TF encountered with Nuxeo, or in a setting with insufficient IT capabilities to bring up and test a DAMS. However, testing with the SR alone would not have revealed the strengths of Fedora underlying two of the open source systems (Islandora and Hydra/Sufia), and the opportunity to explore Fedora as a DAME component would have been missed.

A DIGITAL "ECOSYSTEM" TO SERVE THE SCHOLARLY COMMUNICATION ECOLOGY

As we evaluated the variety of candidate DAMS, it became apparent that no single system could meet the diversity of library and campus needs by itself. The technician on the com- mittee proposed that we consider solutions involving an ecosystem of many services that could communicate while separating concerns among storage, preservation, and access needs. This realization suggested a new designation for the architecture: the Digital Asset Management Ecosystem (DAME).

Having settled on a DAME as the most appealing solution, the evaluation was refocused on interoperability and complementary functionality of components of the scholarly com- munication ecology. In this view, a given DAMS plays a central role of storing items and metadata and providing interfaces to enable access and administration.

The technical documentation of various DAMS presents a variety of architectural dia- grams, but they all share certain core architectural layers and features. For the purposes of exposition, we can call these the "Application," "Business Logic," and "Storage" layers. The Application layer of a DAMS provides user interfaces for management, discovery, and exhibition. It also exposes APIs for use by third-party applications. The Business Logic layer mediates access from the Application layer to the Storage layer where content and metadata are hosted. The Business Logic layer handles such roles as event logging, event messaging, indexing of content, and access permissions. The Storage layer is the ultimate repository of content (files) and metadata. It typically manifests as a file system and data- base, but variations and adjuncts such as Solr indexes, RDF triplestores, and cloud storage are possible. A generic representation of such an application can be seen in **Figure 1**.

A DAME will exhibit a layered architecture like a DAMS, but the DAME differs in that its component parts are discrete applications and services. These applications and services are distributed across the layers of the DAME. Its various components (including DAMS) all participate with the DAME in a modular fashion. In general, a DAME will incorporate the following layers:

- Authentication
- Presentation
- Management API
- Affiliated Applications

The distribution of such applications and services across these layers is depicted in **Figure 2**.

A DAMS, which includes its own Application, Business Logic, and Storage layers will par- ticipate in the DAME as one of the Affiliated Applications. Other systems such as preserva- tion services, an integrated library system (ILS), or scholarly tools, such as VIVO, would participate in this capacity as well. The crucial element of a DAMS or other Affiliated Ap- plication that enables participation in the DAME is the API portion of its Application layer. All major DAMS provide APIs to accommodate this role.

The Authentication layer provides a single point of entry to mediate user access to the DAME. In this way, an institution can rely on a single authentication regime (such as Shib- boleth SSO) and avoid the need for users to maintain separate logins for myriad applications.

The Presentation layer houses the user-facing applications where authenticated user can manage, curate, discover, browse, and otherwise access the DAME's content and metadata. Presentation layer applications can be custom built or off-the-shelf third-party user inter-faces. The requirement here is that they be coupled with the Management API layer, which will mediate access to affiliated services such as DAMS.

The Management API of the DAME will provide the Presentation layer with access to the underlying Affiliated Applications by communicating with their APIs. In this way, the Management API layer provides a single route of communication for the UIs in the Pre- sentation layer to interact with the various Affiliated Applications. Insofar as the DAME's Management API is coded to interact with different DAMS, the DAME can be repository agnostic; that is, if a decision is made to change out a DAMS, the rest of the DAME will not require any updates or changes. Furthermore, with a Management API in place, multiple DAMS, preservation services, ILS, or third-party APIs (for maps, weather, etc.), can be ag- gregated and homogenized for use by Presentation layer applications.

A major design consideration to help incorporate off-the-shelf third-party software in the Presentation layer and in the Affiliated Application layer is the utilization of standard pro- tocols and formats between interfaces. For example, Solr is a widely adopted indexing tool with a well-defined API—the Management API layer can provide a pass-through for Solr indexes to accommodate a wide variety of open-source discovery applications in the Pre- sentation layer. The International Image Interoperability Framework (IIIF) is another API specification that can support many important use cases when incorporated into the Man- agement API layer.

NEXT STEPS

Our analysis of 25 systems allows us to confidently assert that no one digital asset man- agement product will meet even a fairly standard set of library and campus needs without extensive customization. Needs will evolve and change over time, as will technological ca- pabilities, necessitating an endless quest for a better system and incurring continuing over- head in personnel time and equipment costs to discover, evaluate, deploy, and migrate new systems. The DAME model, built as it is around the addition, replacement, and removal of components, does not negate the need for ongoing investment and adjustment but rather anticipates it. Ultimately, research and scholarly communication functionality took prec- edent over campus business needs. The libraries have moved forward in implementing the DAME architecture described in this article, with Fedora and DSpace serving core storage and management roles. The flexible nature of the DAME architecture, and our ambition to position services and tools for persisting complex digital objects in the context of myriad other scholarly communication services and tools, has guided the growth of an even broader digital library approach at our institution. ■

ABOUT THE AUTHORS: The authors are all with Texas A&M University. Robin R. Sewell is a web & electronic access librarian. Sarah Potvin is a digital scholarship librarian. Pauline Melgoza is a science & engineering librarian. James Silas Creel is a software applications developer. Jeremy T. Huff is a software applications developer. Gregory T. Bailey is a university archivist & Clements curator. John Bondurant is a digital archivist. Sean Buckner is a digital preservation librarian. Anton R. duPlessis is the Mexican colonial collection curator. Lisa Furubotten is the director of cataloging & metadata. Julie A. Mosbo Ballestro is the director of preservation. Ian W. Muise is a preservation librarian. Brian J. Wright is a senior audio visual specialist, research communications.

REFERENCES

Andro, M., Asselin, E., & Maisonneuve, M. (2012). Digital libraries: Comparison of 10 software. *Library Collections, Acquisitions, and Technical Services,* 36(3), 79–83. https://doi.org/10.1016/j.lcats.2012.05.002 Digital asset management. (n.d.) In Wikipedia. Retrieved October 5, 2017, from https://en.wikipedia.org/wiki/Digital_asset_management

Donohue, T. (2014). DSpace 2013 vision document (DuraSpace Wiki). Retrieved October 5, 2017, from https://wiki.duraspace.org/display/DSPACE/DSpace+2013+Vision+Document

Gilbert, H., & Mobley, T. (2013). Breaking up with CONTENTdm: Why and how one institution took the leap to open source. *Code4Lib Journal,* 20. Retrieved from http://journal.code4lib.org/articles/8327

Grappone, T., Fleming, D., Hetzner, E., Perry, S., & Tingle, B. (2013). Next-Generation Technical Services (NGTS) POT 1 LT 1C: Proposed model for systemwide digital asset management system (DAMS) with discovery and display (final report). Retrieved from http://libraries.universityofcalifornia.edu/groups/files/POT1_LT1C_finalreport_08Feb2012.pdf

Lidwell, W., Holden, K., Butler, J., & Elam, K. (2010). *Universal principles of design: 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design* (2nd ed.). Beverly, MA: Rockport Publishers.

Maslov, A., Mikeal, A., & Leggett, J. (2009). Cooperation or control? Web 2.0 and the digital library. *JODI: Journal of Digital Information,* 10(1), 2. Retrieved from https://journals.tdl.org/jodi/index.php/jodi/article/view/271

Masood, K., & Neatrour, A. (2014). Digital asset management system options: Report of the University of Utah libraries DAM Review Task Force. Retrieved November 1, 2017

NLM Digital Repository Evaluation Selection Working Group. (2008). Recommendations on NLM digital repository software. Retrieved from https://www.nlm.nih.gov/digitalrepository/DRESWG-Report.pdf

Schmidt, L., Ghering, C., & Nicholson, S. (2011). Digital curation planning at Michigan State University. *Library Resources & Technical Services,* 55(2), 104–118. https://doi.org/10.5860/lrts.55n2.104

Stein, A., & Thompson, S. (2015). Taking control: Identifying motivations for migrating library digital asset management systems. *D-Lib Magazine,* 21(9/10), 5–5. https://doi.org/10.1045/september2015-stein

Weidner, A., Watkins, S., Scott, B., Krewer, D., Washington, A., & Richardson, M. (2017). Outside the box: Building a digital asset management ecosystem for preservation and access. *Code4Lib Journal,* 36. Retrieved from http://journal.code4lib.org/articles/12342

Wikipedia. (2017). Digital asset management. Retrieved October 5, 2017, from https://en.wikipedia.org/wiki/Digital_asset_management

Woods, D., & Guliani, G. (2005). *Open source for the enterprise: Managing risks, reaping rewards.* (1st ed.). Sebastopol, CA: O'Reilly Media.

Wu, A., Thompson, S., Vacek, R., Watkins, S., & Weidner, A. (2016). Hitting the road towards a greater digital destination: Evaluating and testing DAMS at University of Houston libraries. *Information Technology & Libraries,* 35(2), 5–18. https://doi.org/10.6017/ital.v35i2.9152

Footnotes

[1] There are compelling arguments against incorporating video playback functionality in DSpace, given institutional configurations that pose technical challenges. For more on this issue, and Virginia Tech's work to implement video playback in their DSpace instance, see Gilbert and Mobley (2013).

[2] See supplemental data, https://doi.org/10.18738/T8/A21OLT

[3] https://sandbox.islandora.ca

[4] https://curationexperts.com/

[5] https://wiki.duraspace.org/display/DSDOC6x/SWORDv1+Server