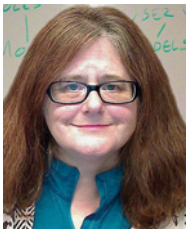
Tools and workflows for bit-level digital preservation strategy at the University of Cincinnati

Received (in revised form): 4th October, 2018

Nathan Tallman
is Digital Preservation Librarian at Penn State University Libraries. He co-chairs the National Digital Stewardship Alliance Infrastructure Interest Group and the Academic Preservation Trust Bagging Interest Group. He was the Digital Content Strategist for the University of Cincinnati Libraries from 2014 to 2017.

Penn State University Libraries, 402 Pattee Library, University Park, PA 16802, USA
Tel: +1 814 865 0860;
E-mail: nathan.tallman@psu.edu



Linda Newman
was Head of Digital Collections and Repositories at University of Cincinnati Libraries, where she oversaw digital preservation strategy. She has been an active participant in the Academic Preservation Trust, including serving on its governing board. She retired with emeritus status from the University of Cincinnati at the end of September 2018.

University of Cincinnati Libraries, PO Box 210033, Cincinnati, OH 45221, USA
Tel: +1 513 556 1424;
E-mail: newmanld@ucmail.uc.edu

Abstract Digital preservation is a complex, highly technical activity. Large institutions with many collections in multiple repositories (or repository-like systems) may find it challenging to corral digital collections, preservation masters and metadata into coherent archival information packages for preservation storage. This paper describes the tools and workflows for a bit-level preservation strategy for digital content at the University of Cincinnati Libraries and lessons learned on the way.

KEYWORDS: digital preservation, preservation storage, distributed preservation, workflows, lessons learned

INTRODUCTION

The University of Cincinnati chose the Academic Preservation Trust (APTrust) for preservation storage in 2013, in the midst of changes to the local digital repository infrastructure. With the attitude that any preservation is better than no preservation, and as content in APTrust can be updated, the authors did not want to wait for the ideal infrastructure and workflows to be in place before sending content. By discussing the University of Cincinnati Libraries' (UC Libraries) experience in establishing workflows for bit-level digital preservation, the authors hope the lessons learned from their experience will help others avoid some of the pitfalls associated with this process.

Experience suggests that getting started, even with minimal workflows, can present many challenges. Corralling digital collections, preservation masters and metadata into coherent archival information packages (AIPs)[1] is easier said than done. The data that typically comprises AIPs is

often scattered in repositories, file systems and people's heads. While every institution's workflows will look slightly different, there are nonetheless commonalities for achieving a bit-level preservation strategy that are applicable to any institution.

BACKGROUND AND CONTEXT

The University of Cincinnati Libraries (UC Libraries) began digitising collections in the late 1990s. Since the mid-2000s, born-digital content has also been collected. This content is made available via three digital repositories, using three different applications — Samvera, DSpace and LUNA, and some content is delivered through websites. Most of these collections do not include the preservation master files, which are stored separately on an Isilon file system. The Samvera-based institutional repository, Scholar@UC, allows faculty, staff and select students, to self-deposit research objects. The repositories based on DSpace and LUNA are largely for digitised cultural heritage content and digital archives owned by UC Libraries. UC Libraries also hosts an instance of Open Journal Systems (OJS) for online journals.

Work is underway to build a Samvera-based repository to replace DSpace and LUNA. The migration process has the potential to unite access and preservation master digital files. Storing all digital content in a Samvera repository will facilitate active management and serialisation to bags for transfer to preservation storage environments. In the meantime, it is still necessary to achieve bit-level preservation.

ROLES AND RESPONSIBILITIES

UC Libraries has not, to date, had dedicated staff for digital preservation. Responsibilities have been distributed among the Digital Collections and Repositories department and the Archives and Rare Books Library, with support from Library IT and IT@UC. The Digital Collections and Repositories department, as headed by Linda Newman, has been the operational home for moving content from repositories to APTrust. Nathan Tallman, while at the University of Cincinnati, completed most operational activities in coordination with Digital Repository Developer, Glen Horton and Digital Archivist, Eira Tansey. Eira Tansey stewards born-digital content not yet in a repository.

CHOOSING A PRESERVATION STORAGE PARTNER

This paper does not provide an analysis of preservation storage providers as there are many options available with varying levels of service for different needs.[2] Preservation storage is a critical aspect of bit-level digital preservation[3] and is different from regular data storage; Schaefer *et al*. have created preservation storage criteria that articulate these differences.[4] The University of Cincinnati had a strong organisational interest in using APTrust for preservation storage.

APTrust is a consortium of cultural heritage institutions working together to develop and achieve digital preservation strategies. It is managed and operated by the University of Virginia and relies heavily on member collaboration for shared governance. Through working and interest groups and twice-yearly meetings, members collaborate to find community solutions to local problems. APTrust is both a deposit location and a replicating node for the Digital Preservation Network (DPN). The initial focus of APTrust has been a stable bit-level preservation storage repository with diverse geo-spatial redundancy and fixity.[5]

Digital content is sent to APTrust via BagIt[6] bags, which are arranged into intellectual objects. After bags are received and validated, two instances are created for every file, one on live spinning disk and another in nearline storage,[7] both in Amazon Web Services (AWS), and with geographic distance between the two instances. Each

instance is replicated twice for a total of three copies in each instance, making a total of six copies for each file. Fixity audits occur every 90 days on the primary spinning disk instance; if corruption is found, a valid copy is used to restore corrupted copies. Event metadata is captured during each step of the process and stored in the system.[8]

One feature of APTrust that has proven to be critical is the ability to delete intellectual objects. This is not a common feature in preservation repositories as they are designed to store content both securely and indefinitely. When an intellectual object is deleted, a record of the object is retained in the system, as well as all associated event metadata; a new event is added to reflect the deletion. Processing of unorganised (or less than ideally organised) content takes time. As with archival processing of physical materials, this may happen iteratively over many years. A preservation storage environment that allows deletion makes it possible to update stored collections as they are iteratively processed.

TOOLS FOR IMPLEMENTING WORKFLOWS FOR TRANSFERRING CONTENT TO APTRUST

There are many ways and means to process digital content for preservation storage. Each institution will develop its own toolkit based on local infrastructure and capacity. UC Libraries leverages the following open-source tools and software for managing digital content, contributing code back to the community as appropriate:

- *Bagit-Python* is a command-line tool and Python library for creating and modifying bags.[9] It is maintained by the Library of Congress (LoC) and has near parity with bagit-java,[10] the primary bagging software maintained by LoC for creating and modifying bags. Bagit-Python is missing the ability to split bags. This was originally a problem for UC Libraries; however, APTrust went from a 250 GB maximum bag size to a 5 TB maximum bag in 2017, obviating the need to split large bags into parts. UC Libraries chose bagit-Python as it had more local expertise with Python and the command line interface made it easy to integrate into bash scripts; by contrast, bagit-java has no command line interface as it is only a library that must be called from a java program.
- *APTrust Partner Tools* are command-line binaries written in Go using the official AWS S3 library.[11] They are maintained by APTrust and may be used to monitor the progress of ingest, validate bags to the APTrust specification, upload bags, list and delete the contents of AWS transfer buckets, and to download restored bags. These tools may be incorporated into any script or application. Alternatively, many things can also be accomplished using the APTrust Member API.[12] UC Libraries uses the partner tools for sending content and the Member API for monitoring ingest.
- *Aws-cli* is a Python-based command-line tool for interacting with AWS.[13] It is used primarily for uploading bags and listing bucket contents. Use of this tool has not been necessary since the APTrust Partner Tools were released, but some local scripts that have not yet been updated may still make use of aws-cli.
- *DSpace* is an all-purpose digital repository used by many institutions. It has a built-in feature that exports collections and objects as zipped, pre-bagged AIP packages that include a Metadata Encoding and Transmission Standard (METS) XML wrapping all metadata.[14] These packages conform to the BagIt specification but need additional data to conform to the APTrust bag specification. UC Libraries has used DSpace for many years and originally chose it because it was hosted through a consortium. The consortium has since ended this service and UC Libraries' DSpace instance is now hosted at the University of Cincinnati.

- *Jq* is a JavaScript Object Notation (JSON) processor used to parse JSON returned by the APTrust Member API. *Xsltproc* is an XML processor used to parse XML data returned by the DSpace packager. Both of these tools parse data from other tools in the chain and were chosen for their simplicity.
- *Red Hat Enterprise Linux* is the operating system on the virtual server where bagging takes place. These tools are combined in a set of locally developed *bash* scripts called *Tricerabagger* that convert directories into APTrust bags and can optionally transfer them to APTrust.[15] Red Hat Enterprise Linux was chosen because it is the preferred Linux distribution of University of Cincinnati IT@UC (enterprise information technology that hosts the virtual server). Bash is the default shell (command line interpreter) used by Red Hat Enterprise Linux.
- Nathan Tallman has updated these tools at his new institution, Penn State University Libraries. psuBagger now leverages the APTrust partner tools for transferring bags. In addition, variables that used to be hard coded can now be passed as arguments.[16]
- *Samvera* (formerly known as Hydra) is an open-source digital repository framework that is used by many academic libraries. It commonly combines *Fedora*, an open source digital repository, and *Blacklight*, an open source catalogue and discovery tool. UC Libraries chose Samvera for its extensibility, developer community and open source nature.[17]

WORKFLOWS

While some of the details change, the overall workflow for sending content to APTrust is the same, as illustrated in Figure 1:

1. Content must be selected for preservation storage.[18]
2. Content is exported from a repository and/or staged on a Linux filesystem.
3. If available, descriptive metadata is parsed from XML and stored or supplied as a variable.
4. The bagging process uses the variables to create an APTrust compliant Bagit Bag.
5. Aws-cli and the partner tools are used to send content to APTrust and monitor the ingest process.

DSpace collections have the most specialised workflow. First, selected collections are exported from DSpace as zipped AIP bags. The bags must then be unzipped and restructured for further processing. Next, xsltproc extracts a description which is passed to bagit-Python when it creates the bag. Xsltproc then extracts a title from the bag and uses it to create an APTrust-specific tag file, aptrust-info.txt. The bag is then serialised to an uncompressed tar file and is ready to be transferred to APTrust.

While the script, Tricerabagger, could doubtlessly be improved on in several ways, it has one glaring weakness. Technically speaking, the bags could be called invalid because the tag manifests are not updated when adding aptrust-info.txt and updating bag-info.txt to include the bag title. Because the tag files are edited/created after the bag has been created, the checksum for bag-info.txt will not match the tag manifest and aptrust-info.txt will not be listed in the manifest. The bagit-Python tool used by Tricerabagger does not have this ability to re-generate the tag manifest. The bagit-java library does have this feature, however, and custom java scripts may be written to re-generate the tag manifest. APTrust does not validate the tag manifests, only the content manifest. As the checksums in the content manifest are validated against the actual content when APTrust receives the bag, UC Libraries feels confident that fixity of the content is maintained. While the tag manifest would ideally be re-generated, this is not a problem for sending content and it was decided that it was more important to send content to preservation storage than it was to

Figure 1: Basic workflow for sending content to APTrust

have the perfect workflow. It is always possible to iterate and improve the process later.

Samvera content was pragmatically bagged as a single directory when Scholar@UC was based on Samvera code on top of Fedora 3. The directory structure in Fedora 3 allowed the entire repository to be bagged as a single bag. A similar script was used as the previous workflow, but instead of extracting metadata, it was necessary to change variables in the script manually. This tactic worked while the institutional repository was small and until Scholar@UC was upgraded to Fedora 4. This workflow is changing for the new instance of Scholar@UC, which uses Fedora 4. In this case, the Fedora import/export tools will be used to create APTrust valid bags for each work object. This introduces the necessity to track when individual works have been sent to APTrust, which new works have not been sent, and which works already sent to APTrust require update or deletion. (It is not UC Libraries' intention to preserve content that faculty delete from the self-submission repository.) Bagging the entire repository as a single bag avoids grappling with those issues.

Other content generally follows the *file system workflow*. This is true for LUNA, OJS and other content not stored in a repository. They differ in the first step, which is to gather the content. LUNA does not include preservation masters, so metadata is exported and saved with the preservation masters. OJS content is copied from the web server, as well as a database dump. (OJS has the ability to deposit content into a Community LOCKSS Network run by the Public Knowledge Project, the maintainers of OJS. UC Libraries should explore this as a preservation option for open access journals.) Content has been exported by collection from these systems, and a manual system is maintained for tracking which collections have been preserved.

LUNA, OJS, or other content is gathered into a directory for Tricerabagger to run against. The script variables are updated for the specific content each time the script is run. Tricerabagger creates the bag, adds the APTrust-specific files, creates the tar file for the bag, and transfers it to APTrust.

To minimise the amount of storage space that would need to be set aside for content on its way to APTrust, UC Libraries created a shared storage area, using campus IT's Isilon technology, that could be addressed by multiple servers. Scripts that exported content from the Luna, DSpace, OJS or Samvera users would use the same area of physical disk as the destination. This allowed the bash scripts and APTrust tools to be installed on a single server, and it avoided mass transfers of content from one server to local disk and then backup to another area of Isilon storage. Such mass transfers can themselves introduce the possibility of disk corruption, so it is advisable to design a workflow that minimises the local transfer of files prior to upload to the preservation storage network.

LESSONS LEARNED

UC Libraries has learned many lessons from the experience of selecting tools and establishing workflows for bit-level digital preservation. Most, if not all, of these lessons apply to any institution seeking to do the same. These lessons can be grouped into three categories: collaboration, technology, and systemising and automation.

Collaboration

Collaboration is the key to realisation

Digital preservation is a highly collaborative process by nature, especially at an institution with distributed responsibility. This can be likened to a traditional preservation department where collection curators work with preservation specialists to determine priorities for treatment. Preservationists should be careful about their role in selection and provide the tools, knowledge and evaluative criteria for curators to make selection decisions.[19]

Digital preservation is also a highly technical process

Just as a traditional preservation department has strong relationships with facilities, so strong relationships with IT and infrastructure departments are also essential. As will be discussed in the following paragraphs, numerous technical hurdles can interfere with processing materials through a digital preservation workflow. Having the system administrator, network technicians and storage specialists working together can make or break a programme.

Digital preservation can be a complex and costly endeavour

Having a community of fellow practitioners willing to discuss shared issues can be helpful for problem solving. While not quite benchmarking, it also allows one to be better informed of what colleagues are working on and finding new potential areas for collaboration. Working together can also help share the financial burden. UC Libraries chose APTrust, but other options and different price points are also available. There may also be special pricing for consortia.

Plan ahead and plan with colleagues

Digital preservation is collaborative because it affects many stakeholders. Involve them in the planning and keep them in the loop. Be agile because the facts may change and it is necessary to inspect and adapt the plan to continue making progress.

Collaboration takes time

When input or action is needed from another department, meetings or approvals may ensue. Be sure to allow enough time, especially if a protracted decision-making process is likely.

Technology

Infrastructure has limitations

If moving large amounts of content through a network, there are bound to be issues. Cabling, switches, firewalls, bandwidth and network cards can all get in the way (especially if taken offline for an upgrade). Talk to network administrators and technicians, tell them the current plan, and determine whether there will be any barriers. Moving data on a slow network can take time and, if not planned for, deadlines may be missed.

Network storage can be friend or foe

If any local infrastructure relies on network storage, talk to its administrators. If accessing the same content from Windows and Mac desktops, as well as a mounted network file storage for command-line access, file permissions may become a problem. It may be necessary to have root-level administrative privileges because of file permissions. Be sure to talk to the system administrator about this as early as possible as root-level privileges are usually restricted.

Good file management is a practice, not a onetime activity

In terms of file management, it pays to be proactive. If there is no intellectual control of digital collections, one cannot expect to steward them properly. If master files are kept separate from access copies, be sure to have

records on where they are and how they relate to the digital objects in a collection. A well-organised file and directory structure with a solid file-naming convention can help future colleagues interpret the materials easier and make future ingests into other repositories easier. Another thing to watch out for, in terms of file names and the content itself, is character encoding.[20] Some systems will expect UTF-8 character encoding, but content, especially legacy content, can use other character encoding standards, such as Latin 1. Mismatched character encoding can lead to errors and loss of content.

Bagging choices matter

Any directory can be turned into a bag. What is in that directory, ie to what level it is bagged, has downstream implications. Think about what a future colleague will do with this content. If an entire repository is bagged in one go, it might be a huge bag and hard to disentangle if the software is no longer in use. It may also present challenges to processing, transferring and ingestion into a preservation system. In the case of restoration, it would be necessary to download the entire repository bag for a single work. If an entire collection is bagged, it makes it easy to send and keeps the collection together, but consider how metadata will be updated or individual files may be replaced if content in the collection changes. If bagging individual work objects, how will they be related to other members of the same collection? Are deleted work objects being preserved, and if not, how are deletions being tracked? What will the process be to restore an entire collection if all the works are individual objects? Archival material generally needs to be contextualised in a collection for best understanding.

Ideally, AIPs are self-contained packages that describe the content and its properties, structured for long-term preservation storage. Because preservation storage is expensive, it might be decided to include only the preservation master files. However, if the AIPs are also intended to be used in disaster recover situations, this changes the nature of the contents. It may be necessary to include derivative files so that the packages can quickly be ingested into a digital repository. Consider whether the bag names or identifiers will need to follow a schema to identify source repository or collection, providing context for recovery.

Systematising and automation

Make the preservation workflow into routine practice

Collection-by-collection efforts can be sporadic and time-consuming. In the initial stages of APTrust, as bagging standards evolved and limits such as bag size were addressed, throughput was not paramount. Now that standards have been agreed, however, it is clear that if it is to realise the routine preservation of all content, per the agreed schedule, the university will need additional tools to export, bag, upload and track with less human intervention. However, this level of automation may not be possible until there is a reduction in the number of platform sources.

When possible, automate

Hand-crafting artisanal bags may be necessary when getting started. This allows one to get the content to a safe environment and become familiar with the process. However, it is much more sustainable to automate the process.[21] If the local repository or other systems can transfer content automatically, focus can be shifted to getting content into repositories in the first place. Automation should be guided by an algorithm that uses local selection criteria, but do not let the algorithm be the final arbiter of selection decisions; algorithms are meant to assist rather than limit a curator's ability to make collection management decisions such as selection for

digital preservation. Curators should be able to override or otherwise alter the default preservation workflow choice determined by the algorithm.

Test all assumptions, test restorations, test everything

Digital preservation is playing the long game and high confidence in one's preservation storage environment is essential. Anywhere that assumptions are made they need to be tested. Test restoring content from preservation storage — is the result what was expected? Can the bag be validated? Can the AIP be ingested into the local repository? Has the preservation storage met its service agreement promises? If problems are encountered, how are they resolved?

Iterate to success

Information professionals can get caught up in the quest for perfection. If fortunate enough to be able to replace objects in preservation storage, do not wait for the ideal workflow and file management — send what is on disk as soon as possible. Any preservation is better than no preservation; if local storage becomes corrupt from a software bug, preservation copies will be necessary. As you are able to process digital content, replace quick-and-dirty bags with better ones later. (The strategy may switch from a collection-level to object-level approach after the initial campaign of transferring content for initial deposit.)

CONCLUSION

Digital preservation can be a daunting challenge. It is complex, it is expensive and it is time-consuming. However, it is also essential that we responsibly steward cultural heritage content for future generations. Solid planning, with built-in flexibility and good relationships, can make or break workflows. Finding the ideal workflow for one's institution and identifying barriers will provide assurance that with the content in hand, bit-level digital preservation can be achieved. Having confidence in workflows allows energy to be better focused on other strategic and operational priorities and facilitates the development of a strong digital preservation programme.

ACKNOWLEDGMENTS

The authors would like to thank their co-presenters, Salwa Ismail (Head of Library Information Technology, Georgetown University), Suzanne Chase (Head of Digital Services Unit, Georgetown University) and Joe Carrano (Digital Archivist, Massachusetts Institute of Technology) from their National Digital Stewardship Alliance Digital Preservation 2017 presentation, on which this paper is based.[22]

References

1. Consultative Committee for Space Data Systems (2012) 'Reference Model for an Open Archival Information System (OAIS)', Report No. CCSDS 650.0-M-2, pp. 4–36, available at: https://public.ccsds.org/pubs/650x0m2.pdf (accessed 4th January, 2017).
2. A group of digital preservation service providers and community members has created a matrix to compare services that may be useful to those still considering options. At the time of writing, it was still a work in progress and not ready to be released. For more information, contact Bradley Daigle at the University of Virginia.
3. Wright, R., Miller, A. and Addis, M. (2009) 'View of the significance of storage in the "cost of risk" of digital preservation', *International Journal of Digital Curation*, Vol. 4, No. 3, pp. 104–122.
4. Schaefer, S., McGovern, N., Goethals, A., Zierau, E., Truman, G., Zwaard, K., Mandelbaum, J. and Knight, S. (2017) 'Preservation Criteria, Version 2', available at: https://osf.io/sjc6u/ (accessed 4th June, 2018).
5. Diamond, A. (2017) 'APTrust 2.0', presentation at APTrust Spring Meeting, 12th April, available at: https://docs.google.com/presentation/d/1I4WRjYizyO2L_GzsPjXGvD_F1Vct3yfH-iQTVYFjNzc/edit?usp=sharing (accessed 4th June, 2018).
6. The BagIt specification describes a way of packaging files for transfer. The content is saved in a data directory, with metadata saved in text files in the root directory of the bag. One of the metadata files is a manifest of all the files contained in the

data directory with a checksum for each, usually MD5 or SHA-256. The manifest makes it easy to verify bit-integrity of the content, either manually using programs like Bagger (https://github.com/LibraryOfCongress/bagger), or automated using the command line tools mentioned elsewhere in this paper. For more information, see Kunze, J., Scancella, J., Adams, C., Madden, L. and Littman, J. (2018) 'The BagIt File Packaging Format (V1.0)', Internet Engineering Task Force, available at: https://tools.ietf.org/html/draft-kunze-bagit-17 (accessed 1st October, 2018).

7. Nearline storage is used here to describe a tier of storage below live spinning disk, but above a traditional tape library. It is offline, in the sense that it is not immediately available via a URL, but can be requested and, after a short waiting period, then become available via a URL. AWS does not specify which technology is used in its service; it may be tape or another low-performance storage technology.
8. For a fuller, more technical description of what happens to bagged digital content after it is received by APTrust, and the APTrust ingest and ongoing processes to maintain fixity, see APTrust (n.d.) 'Definition of AIP', APTrust Member Wiki, available at: https://wiki.aptrust.org/Definition_of_AIP#Transforming_SIPs_into_AIPs (accessed 4th June, 2018).
9. Library of Congress (2017) 'bagit-python: Work with BagIt packages from python', available at: https://github.com/LibraryOfCongress/bagit-python (accessed 4th June, 2018).
10. Library of Congress (2018) 'bagit-java: Java library to support the BagIt specification', available at: https://github.com/LibraryOfCongress/bagit-java (accessed 4th June, 2018)
11. APTrust Member Wiki (2017) 'Academic Preservation Trust. Partner Tools', available at: https://wiki.aptrust.org/Partner_Tools (accessed 4th June, 2018).
12. APTrust Member Wiki (2017) 'Academic Preservation Trust. Member API', available at: https://wiki.aptrust.org/Member_API (accessed 4th June, 2018).
13. Amazon Web Services (2018) 'aws-cli: Universal Command Line Interface for Amazon Web Services', available at: https://github.com/aws/aws-cli (accessed 4th June, 2018).
14. DuraSpace Wiki (n.d.) 'DSpace AIP Format — DSpace 1.8 Documentation', available at: https://wiki.duraspace.org/display/DSDOC18/DSpace+AIP+Format#DSpaceAIPFormat-MetadatainMETS (accessed 4th June, 2018).
15. Tallman, N. (2017) 'tricerabagger: Utilities for bagging AIPs produced by DSpace 1.8.2 to the APTrust bag specification', University of Cincinnati Libraries, available at: https://github.com/uclibs/tricerabagger (accessed 4th June, 2018).
16. Tallman, N. (2018) 'psuBagger', Penn State University, available at: https://git.psu.edu/digipres/psuBagger (accessed 26th February, 2018).
17. Samvera (http://samvera.org/) is an open-source digital repository framework developed and supported by the library, archive and museum community. As a framework, it is not presently a turn-key solution that can simply be installed and run, but recent work on the Hyku project has sought to create such a solution (https://github.com/samvera-labs/hyku). Additional documentation is also available (http://samvera.github.io/). Fedora, an open-source repository platform, is a common component of Samvera repositories. Fedora provides the underlying repository features, including metadata and file storage, and a linked data endpoint (https://duraspace.org/fedora/). Blacklight is an open-source discovery interface that provides catalogue functionality in Samvera repositories and has many plugins to extent functionality beyond traditional catalogue features, such as geographic-based discovery and online exhibitions (http://projectblacklight.org/).
18. The Digital Preservation Network has partnered with AV Preserve to create a curriculum on digital preservation workflows that includes selection, processing, roles and responsibilities, programme building and sustainability, which may be helpful when designing local workflows. Digital Preservation Network (2017) 'Digital Preservation Workflow Curriculum Now Available', available at: https://dpn.org/news/2017-07-31-digital-preservation-workflow-curriculum-now-available (accessed 4th June, 2018).
19. For additional guidance on selecting for digital preservation see: Tallman, N. and Work, L. (2018) 'Approaching appraisal: guidelines and criteria to select for digital preservation', paper presented at the International Conference on Digital Preservation, Boston, MA, 24th–27th September, available at: https://osf.io/8y6dc/ (accessed 27th September, 2018); Ravenwood, C., Matthews, G. and Muir, A. (2013) 'Selection of digital material for preservation in libraries', *Journal of Librarianship and Information Science*, Vol. 45, No. 4, pp. 294–308.
20. Giaretta, D. (2011) 'Advanced Digital Preservation', pp. 78–79, Springer-Verlag, Berlin.
21. Becker, C., Faria, L. and Duretec, K. (2014) 'Scalable decision support for digital preservation', *OCLC Systems & Services: International Digital Library Perspectives,* Vol. 30, No. 4, pp. 284–249.
22. DLF Forum (2017) '#r1e: Trials and tribulations of moving forward with digital preservation workflows and strategies', available at: https://dlfforum2017.sched.com/event/BztW/r1e-trials-and-tribulations-of-moving-forward-with-digital-preservation-workflows-and-strategies (accessed 4th June, 2018).