RUcore
Rutgers Community Repository

Multiple archival data streams and directory uploads:

Dealing with complex multi-archival digital assets paired with simplified presentation formats

Description

This document describes best practices for handling digital assets which contain multiple files as a directory package. Such files can include research datasets, or complex born-digital objects which were derived from multiple source files of various types, nested in multiple subdirectories.

This document also specifies the criteria for sorting and uploading sets of multiple files from a directory object upload, such that they are processed by RUcore in a predictable fashion.

Specification

Multiple Archival Datastreams; One Presentation Datastream

It has been specified that complex objects be handled in WMS by offering the capability to upload an entire directory, and all files and subdirectories recursively contained within, as ARCH datastreams. Although such complex directory (DIR) objects may contain multiple files, the presentation versions of the same object may at times be a single presentation datastream, multiplexed from the DIR archival master objects.

Thus far, most of the scenarios encountered for these types of complex archival objects culminate in a final or near-final rendered object, from which a presentation format is derived that can be presented to users. This rendered object can serve as a derivative master file (DMASTER) in the existing paradigm of RUcore and how WMS handles files. Insomuch as WMS expects at times to verify a relationship between master files and presentation datastreams, we can be reasonably certain that there will consistently be a 1:1 ratio between such rendered, near-final DMASTER files when they are present, and corresponding presentation files for each presentation type. In consideration of this, WMS should be configured to follow a general rubric:

Table 1: Rubric for file handling criteria and relationship ratios

Object Type	Course of action if object contains Master files only	Course of action if object contains both Master and DMASTERs
Map, Still Image, 3DObject	**WMS should maintain 1:1 ratio** of ARCH datastream files to files per presentation stream.	**WMS should maintain 1:1 ratio** of DMASTER files to objects per presentation stream. A 1:1 ratio should not be enforced between presentation and ARCH files, however.
Book, Document, ETD, Faculty Submission, Periodical	There should be **no expectation of 1:1** parity between ARCH/DMASTER files, and presentation files. Example: Multiple TIFF files can be converged into one or more PDFs, or could be used to create multiple JPGS or JP2s in a page-turner presentation, or both.	There should be **no expectation of 1:1** parity between ARCH/DMASTER files, and presentation files. Example: Multiple TIFF files can be converged into one or more PDFs, or could be used to create multiple JPGS or JP2s in a page-turner presentation, or both.
Research Dataset	Behavior can vary from one object to the next. Although currently encountered datasets have a 1:1 relationship, we should prepare for the possibility of there being a disparity in future datasets.	Will need to be reviewed at the point we encounter datasets with a multi:1 relationship.
EAD	N/A: Dynamic rendering of presentation format from source data.	N/A: Dynamic rendering of presentation format from source data.
Moving Image and Sound	If master file-only objects should be digital surrogates in a **single file** (i.e. WAV, MOV, AVI, M4V), accompanied by a presentation format (i.e. mp3, mp4), **maintain a 1:1 ratio** between ARCH and presentation files.	Usually these are born-digital objects, with multiple source clips and application data (e.g. prproj, avi, mov, mts, m4v, wav, m4a, jpg and others). Multiple-file ARCHs (a DIR object) will be preserved as a directory upload, and a rendered DMASTER will be created that is a high-bitrate version of the final product (i.e. AVI, WAV, M4V, MP4, MOV). This will be accompanied by a presentation format (i.e. mp3, mp4). **1:1 ratio will be enforced between the DMASTER and the presentation files**, but not against the ARCH datastreams.
Dark Archive Object	Dark Archives contain objects which have only archival datastreams but no presentation objects. This will result in an N:0 relationship, where N can be any number of ARCH datastreams.	Dark Archives contain objects which have only archival datastreams but no presentation objects. This will result in an N:0 relationship, where N can be any number of the datastreams.

File Sorting

An option for the user to select a consistent, predictable criteria for sorting multiple files by filename during a DIR object upload is necessary. A standardized method of sorting, known as the Unicode Collation Algorithm (UCA), Unicode Technical Standard #10,[1] exists for this purpose and is adhered to by modern versions of PHP and MySQL.

By default, MySQL adheres to UCA in sort operations, when the UTF-8 character set is specified for a database.[2] The same is true of PHP when the locale is set to a UTF-8 localization, such as en_US.UTF-8 UTF-8.[3]

To ensure consistency and predictability in handling of multiple ARCH files, WMS should offer an option in the File Upload dialog for to have files in a DIR upload sorted. When selected, WMS should process files sorted by filename, in accordance with the UCA specification.

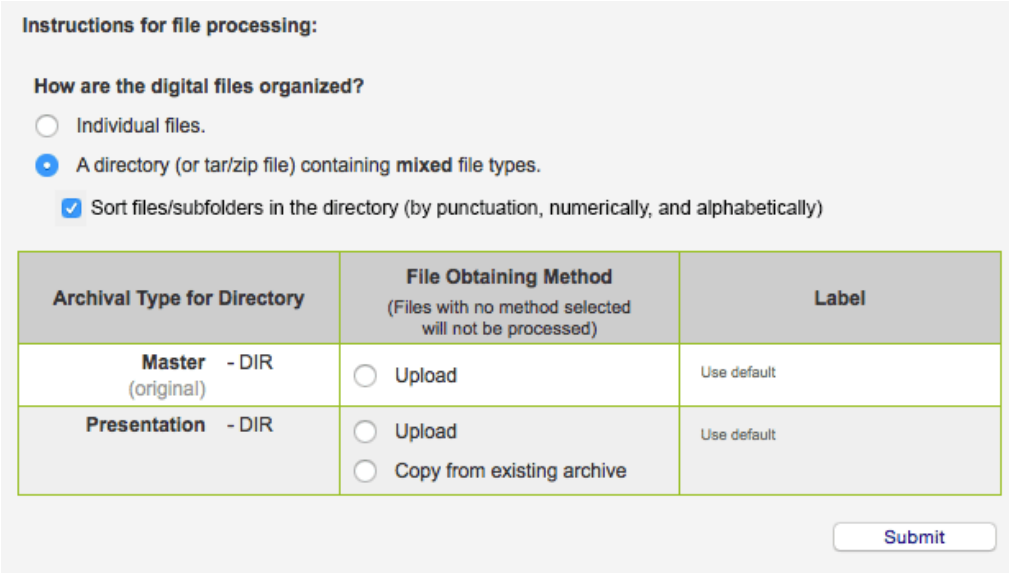

Figure 1: Proposed user-optional dialog to request that files and subfolders in a DIR upload be sorted.

Background Information

Complex Multi-File Master objects

There have been recent questions raised in the development of RUcore regarding a proliferation of object types that previously expected to be consistently simple in nature. For these object types, current and past implementations of digital repositories, including RUcore and the Workflow Management System, have consistently operated on a previously-valid assumption that most digital assets intended for preservation maintained a simplified, single-file structure. These object types, particularly those of the digital surrogate variety, have usually consisted of a structure where a single file constituted the entire object intended for preservation. Consequently, WMS (in most cases) expects to have at least one presentation format generated for each individual archival datastream in a digital object.

This paradigm is shifting, however. An increasing number of born digital objects, as well as some special situations involving some digitization projects, means that we are increasingly encountering digital assets with complex archival master structures, which consist of multiple files of varying types, and whose relative context and position to other related master files must be considered. Such complex masters are increasingly the norm in the research datasets as well.

[1] Davis, M. and Whistler, K. (2014, June 10) "Unicode Technical Standard #10: Unicode Collation Algorithm, Revision 30" Retrieved 2014-12-15 from http://www.unicode.org/reports/tr10/

[2] See collation chart at http://collation-charts.org/mysql60/mysql604.ascii_general_ci.html

[3] An example implementation is the `Collator` class, available in PHP starting with version 5.3.0. See http://unicode-programming.readthedocs.org/en/latest/sorting/php/ for an example.

Example Scenarios for Complex Master Files

- **Roman Coins collection:** Objects in this collection are small in physical size, and require multiple shots to get both in-detail images for presentation, and images with a color calibration target for effective color accuracy. Ultimately, both images are stored to meet preservation requirements, but only the in-detail shots are used to create presentation datastreams.

  As a result, a least two Master images are created for each image in a presentation datastream, but only one of those streams is used to create the presentation that the user will see.

- **Maps collection:** Some Maps may be sufficiently large that the object must be digitized in sections, and the final product "stitched" together to create the final digital image. Due to variations in digital stitching methods, it is necessary to preserve the original section images as well as the stitched object, the latter from which a presentation stream will be created.

- **Born digital Sound and Moving Images:** Most born-digital sound and video objects are created from multiple source elements (e.g. movie clips, sound files, photos, graphics) arranged in a directory, and paired with an XML or similar data file that is read by editing software to render the final product, typically a single WAV, MP3, AAC, MP4, MOV or AVI file.

Legacy complex object types

It should be noted that from the inception of RUcore, complex object types have existed. Some examples include books and other multi-page documents, where a number of multiple scanned TIFFs are stored as archival masters, but the resulting presentation format is a single PDF file. This document does not propose any change to the handling of such object types, but will codify their current treatment for reference purposes.

Additional Notes:

- We have not encountered enough research datasets to characterize typical behavior. A better formula for handling these types of objects may be forthcoming as we gain additional experience.
- Born digital objects (moving image and audio) – For born digital master with multiple object types and files in directories, there would be a DMASTER rendered copy. This permits an archival-quality datastream to be available that is fully rendered and doesn't require the source editing software for a preservationist to view the high quality content. So, enforce the 1:1 ratio… not with the ARCH master files, but with DMASTER.

  In the case where no DMASTER is uploaded, the presumption is that this is because the archival master is already a single file (and AVI or M4v for instance) that came from an already-rendered analog source. In such a case, we can do 1:1 between ARCH master and presentation streams.

File Sorting Issues

At times, the correct WMS pipeline generation of a presentation object, such as a "page-turner" presentation of a yearbook object from an uploaded directory of multiple TIFF and JPG files, depends on the source files being processed in a specific order, usually sorted numerically and alphabetically. Inconsistencies have been encountered in not only how WMS sorts such files, but also how different desktop operating systems (e.g. Mac; different versions of Windows from XP, 7 and 8.x; and even different flavors of Linux) sort the same group of files. In each case, an apparently arbitrary and unstandardized method for sorting is implemented. The end result is that occasionally, unpredictable and erroneous results can occur in the finished presentation stream of an object, such as the display of pages in a book in an incorrect order.

By standardizing to a set criteria for sorting files by filename, users and developers can be reasonably certain that a set of files, properly named, will be processed and ultimately retrieved and displayed in a predictable order that can reduce the probably of such errors.

The Unicode Collation Algorithm defines a standardized, but customizable method to compare and sort strings of characters. These comparisons can then be used to sort text in any writing system and language that can be represented with Unicode. With such a standard already established and used by some of the key infrastructure components in RUcore, it appears logical to adhere to UCA for purposes of consistent sorting of files by name and numeric order.