Evaluating Your DPN Metadata Approach

Prepared by the DPN Preservation Metadata Standards Working Group
Completed July 27, 2017

When determining a metadata approach for DPN deposits, institutions will want to keep in mind the unique context of DPN's preservation services. Because of DPN's long-term commitment to ensure the security of institutions' digital assets, there could potentially be a scenario in which your institution may not be the organization that ultimately recovers the data in the case of catastrophic data loss or institutional collapse. In this case, a data professional without any accompanying institutional knowledge may be the one to rescue the content; it is extremely important, then, to have a metadata approach that is clear and self-evident. Additionally, data may need to be recovered far into the future, when technology could look very different in unpredictable ways.

With this context in mind, DPN's Preservation Metadata Standards Working Group has prepared this document to help guide members through a number of questions that could facilitate the development of a sound approach to preserving your institution's data. Where possible, we have tried to accommodate a range of risk management perspectives to empower members to make decisions that most closely align with their own institutional needs.

What information is needed to understand and contextualize an object?

Descriptive Metadata

Descriptive metadata describes an object for the purposes of identification and discovery. Dublin Core, MODS and VRAcore are common standards used for descriptive metadata. You'll want to review your object-level metadata for inclusion in DPN, keeping in mind the factors that make metadata for DPN unique. Are there any fields which are only relevant for local systems? Do your records need to be enhanced in any way?

Structural Metadata

Structural metadata describes relationships between objects. If this object is part of a structured or sequential set (ie a page in a book), you'll need a way to express this. The Structural Map elements of the METS standard is often used for expressing hierarchical relationships, such as page order or parent/child relationships. The "relationship" element of PREMIS can be used to express version relationships, such as migrations or normalizations.

What information is needed to understand and contextualize a collection?

Descriptive Metadata

A collection-level descriptive record is a great way to give contextual information to a digital collection. DCMI's Dublin Core Collection Description Task Group has created a Collections Application Profile that illustrates mandatory and optional descriptive information for a collection-level record. This DCMI model could easily be adapted for other schemas, such as MODS.

Structural Metadata

As has previously been mentioned, structural metadata is essential for understanding the relationship between objects. As the relationships between objects accrue, one can begin to see collection structure emerge. Collection structure is often stored in the Structural Map in the METS standard. We have also seen a few member organizations rely on EAD finding aids as records of collection structure. Something to think about regarding the structural information of collections is determining whether your institution's preservation approach requires machine-actionable structural information to be preserved with the related bitstreams. Not packaging machine-actionable structural metadata with objects does create a risk of losing the capability of representing digital objects according to the original order of the analog collection.

How do I connect/relate objects to a collection? What are some different approaches?

Objects in a collection could be deposited together.

Perhaps the most certain way to ensure assets remain connected and related to their parent collection is by packaging all assets from a collection together, along with structural metadata indicating each discrete object and its positioning within the collection. (One might also package at a series or sub-series level.) This might translate over to maintaining directory structures in born digital ingests through the use of disk imaging and/or "directory printing." For digitized assets, one might use structural information (such as the structMap in METS) to indicate the archival arrangement of the original items. The degree to how faithful the digital organization should be to the analog physical organization of the reformatted items is debatable; the decisions around this could possibly be based on how your organization views reformatting---whether you are reformatting mainly for preservation, mainly for access, or for both. Often, ingests of born-digital materials from one specific collection might require multiple ingests. Ensuring that collection-level metadata is associated with each ingest would be ideal in this scenario.

Individual objects could be deposited.

If objects from a collection are deposited individually, one would want to enter information into a field that clearly indicates its parent collection. For digitized materials, this option might be explored more often with institutions who are less concerned about digitally preserving the original order of the

analog collection and are instead more focused on preserving individual objects. For born digital ingests, this might indicate multiple ingests are required due to large file sizes.

Both collections and individual objects could be deposited.

This might occur due to several factors: preserving both born digital and digitized assets could lead to a mix of collections and objects, workflows might indicate the need to preserve both the original collection order (created within/by the preservation repository) and individual object metadata (created within/by a system other than the preservation repository), or institutions might preserve assets as they are created through patron digitization requests or other collection/object variant workflows. In any situation, if there are unique distinctions between the purpose of the collection ingests vs. the object ingests (say, the collection package is a structured package of all the preservation masters, and individual object packages contain object-level descriptive metadata for access files that were derived from the preservation collection), appropriate distinctions and connections should be maintained. For instance, you may use metadata to indicate that one package is a preservation master and the other an access object, but you might also use metadata to indicate that the access package contains a derivative of a file located in the preservation package. Other circumstances may not be as complicated; in short, be sure to form consistent practices and guidelines that take into consideration how another entity might approach the data 20+ years from now. What connections and/or distinctions are necessary to make sense of your preservation storage?

How are versions connected/related to one another?

PREMIS

The "relationship" element of PREMIS can be used to express version relationships, such as migrations or normalizations.

NOTE: We would love to hear more about how you are versioning your data. We would appreciate any information regarding this so we can provide more comprehensive guidance to DPN members!

How do I ensure that metadata records are connected to associated objects and collections? What are some different approaches?

All metadata resides in one record.

This is the most conservative and safest option, as a canonical metadata file packaged with the assets it describes is ideal for long-term preservation. Metadata stored within one record in a preservation context is often stored in a METS file. These files are usually produced by a preservation system or application upon processing and/or ingest. A METS record can contain descriptive records at both the collection level and item-level, thus connecting the objects to the collection not only by proximity, but in the data as well.

Metadata resides in multiple records/files.

If your processes require multiple metadata records/files, it would be prudent to create or repurpose a unique identifier (such as an ARK) to link the files together to associate them with a specific access/preservation package, if the metadata files are not stored with the object it describes. This could be drilled down further to include one identifier connecting the metadata to the access object it describes, as well as an identifier linking the access item's descriptive metadata to a preservation instance.

DPN also allows for optional tag directories within its bags, which could serve as a place to store any number of metadata files associated with a deposit's bitstreams. Members may place these directories at the "top level" of the DPN SIP to contain specific types of metadata records. If you are creating bags for submission, you may consider this option when thinking about how to best preserve your locally-created metadata. Metadata files may also be included with the digital objects located in the "data" directory. Regardless of your approach, it is best to have all metadata associated with a specific object included in the deposit, if at all possible.

How do I ensure the authenticity of an object?

Ensure it via the bag manifest/checksum created by DPN.

DPN architecture uses checksums to ensure that the files they receive remain the same. While this doesn't 100% ensure that the depositing institution gets back the same file they uploaded, it does verify that the file remained the same once ingested.

Generate checksums before ingesting objects into DPN.

Tools to do this could range from a command line tool like md5 to bagger/Bagit to Archivematica. Checksums would then be verified / compared against the original after retrieving the object to prove chain of custody and verify authenticity.

In addition to tracking/creating checksums, complete chain of custody documentation is an important indicator of authenticity.

This could include logs of checksum files that date from the initial transfer or creation of the digital file.

How do I distinguish original objects from migrated versions of the originals?

As has been previously mentioned, the "relationship" element of PREMIS can be used to express version relationships, such as migrations or normalizations. A tool like Archivematica maintains a format policy registry which can indicate if normalization needs to occur to a specific file type; in this scenario, Archivematica keeps the original file and also normalizes it to a designated preservation file format. The distinction between the two files is recorded both in the metadata produced by the system and the structure of the AIP.

If you don't have a tool like Archivematica, placing original files in one directory and normalized/migrated files in another can keep the distinctions clear. In this scenario, you might want to name the normalized/migrated file the same as the original file (except for the file extension, of course); this will indicate the linkage between the two files. In the descriptive record for the normalized/migrated file, you might indicate that it was normalized/migrated from the original.

If data must be migrated upon retrieval, how do I ensure that the essential characteristics of the original are known?

Policy determination of essential characteristics.

It is not possible to create an identical copy through format migration - some aspect(s) will always be different. Since the significant aspects of digital objects will vary depending on format, content, context, and other factors, the determination of which characteristics are essential for maintaining authenticity of and access to a given set of materials should be conducted by the holding repository. The object's content, appearance, structure, context, and behavior can be ranked as essential or non-essential and may be treated as such during the transformation process.

Development/adoption of migration pathways.

Establish a set of migration pathways that will allow a particular format to meet the criteria as defined in the policy. This should take into account not only the characteristics to be preserved, but factors such as the target format (widely adopted open formats are best), software tools to perform the migration, pre- or post-processing of the digital material, quality checks of file conversions and required metadata, as well as a process to document these changes in the digital object's lifecycle. Benefits and costs of migration should be taken into account.

Retention of the original, as well as the migrated copy.

In many cases, this would be the best practice for long-term digital preservation. Advances in transformation techniques or emulation availability may allow for improved access to the original in the future.

Documentation of the original with technical metadata.

Capturing the technical details of the original should be an integral part of migration and should be included as an initial component of the migration pathway. Tools such as Harvard's File Information Tool Set (FITS) can help with this process.

What about formats whose essential characteristics I might see as challenging to capture/understand?

Complex objects

Complex objects can be difficult to represent in metadata, description, or migration. For example, databases may lose behavior characteristics (or even structure and context) on conversion to open formats; photo editing files will lose layers (i.e., content and context) on conversion to flat image files.

Objects created using proprietary software

Migration options may simply not exist for many formats created with proprietary software. In some cases, a visual representation may be all that is available. If access to the software is not available (for example, due to cost restrictions), the content itself may likewise be inaccessible.

Objects created with obsolete software

Although options are improving (such as emulation), files that were created with software that has become obsolete may not be accessible without reproducing the original technology.