

The Semantics of Metadata: Avalon Media System and the Move to RDF

The Avalon Media System (Avalon) provides access and management for digital audio and video collections in libraries and archives. The open source project is led by the libraries of Indiana University Bloomington and Northwestern University and is funded in part by grants from The Andrew W. Mellon Foundation and Institute of Museum and Library Services.

Avalon is based on the Samvera Community (formerly Hydra Project) software stack and uses Fedora as the digital repository back end. The Avalon project team is in the process of migrating digital repositories from Fedora 3 to Fedora 4 and incorporating metadata statements using the Resource Description Framework (RDF) instead of XML files accompanying the digital objects in the repository. The Avalon team has worked on the migration path for technical metadata and is now working on the migration paths for structural metadata (PCDM) and descriptive metadata (from MODS XML to RDF). This paper covers the decisions made to begin using RDF for software development and offers a window into how Semantic Web technology functions in the real world.

By Juliet L. Hardesty and Jennifer B. Young

Introduction

[Avalon Media System](#) (Avalon) is “an open source system for managing and providing access to large collections of digital audio and video.” (The Project [updated 2017]) In addition to playback capabilities and audio and video file management, Avalon is a search and discovery interface. Through metadata, audio and video items can be discovered and accessed based on people involved, subjects, dates, and genres along with any keyword-indexed content from titles, abstracts, and notes. This metadata has been managed as [MODS XML](#), a bibliographic XML standard that offers a more human-readable and shareable version of the MARC library catalog record, and is a reflection of the libraries and archives focus of Avalon Media System as a software product. Technical, structural, and some administrative metadata have also been managed using XML but not based on any specific XML standard. The XML created and used for these functional areas has been customized and used internally by Avalon with the Samvera Community (formerly the Hydra Project) software stack using the [Fedora Commons digital repository](#), version 3.x. [1]

Avalon’s move to update to the latest version of the Fedora digital repository software, version 4.x, means a significant shift in working with metadata. Fedora 4 is a major reworking of the digital repository software, particularly regarding increased Linked Data capabilities. Instead of XML files containing different kinds of metadata information associated with a binary file (the “digital object”), [RDF properties](#) are stored together with the digital object (technically still in a text file associated with the digital object but available more directly through the repository software than in the previous environment). These properties can be a mix of ontologies that are needed to express necessary information about a digital object and they are stored together with the properties Fedora tracks on its own about digital objects and relationships between objects. This significant change means multiple ontologies can be used including local ontologies. Choices have to be made on how to use more complex ontologies that involve hierarchical RDF statements (where the object of a subject->predicate->object statement is an entire subject->predicate->object statement) since that can mean creating and managing objects in the repository that are not actual digital objects but rather a concept, like a subject or a person.

For Avalon the goal is to continue providing the same functionality and metadata as this shift occurs from XML to RDF. The following describes the process so far and the plans to move forward and enhance Avalon’s metadata capabilities through the use of semantic web technologies. Migrating Avalon metadata from Fedora 3 to Fedora 4 and from XML to RDF is part of upgrading Avalon versions so changes described here that have already occurred for technical and structural metadata and changes described here that will occur for descriptive metadata will be part of going through the Avalon upgrade process.

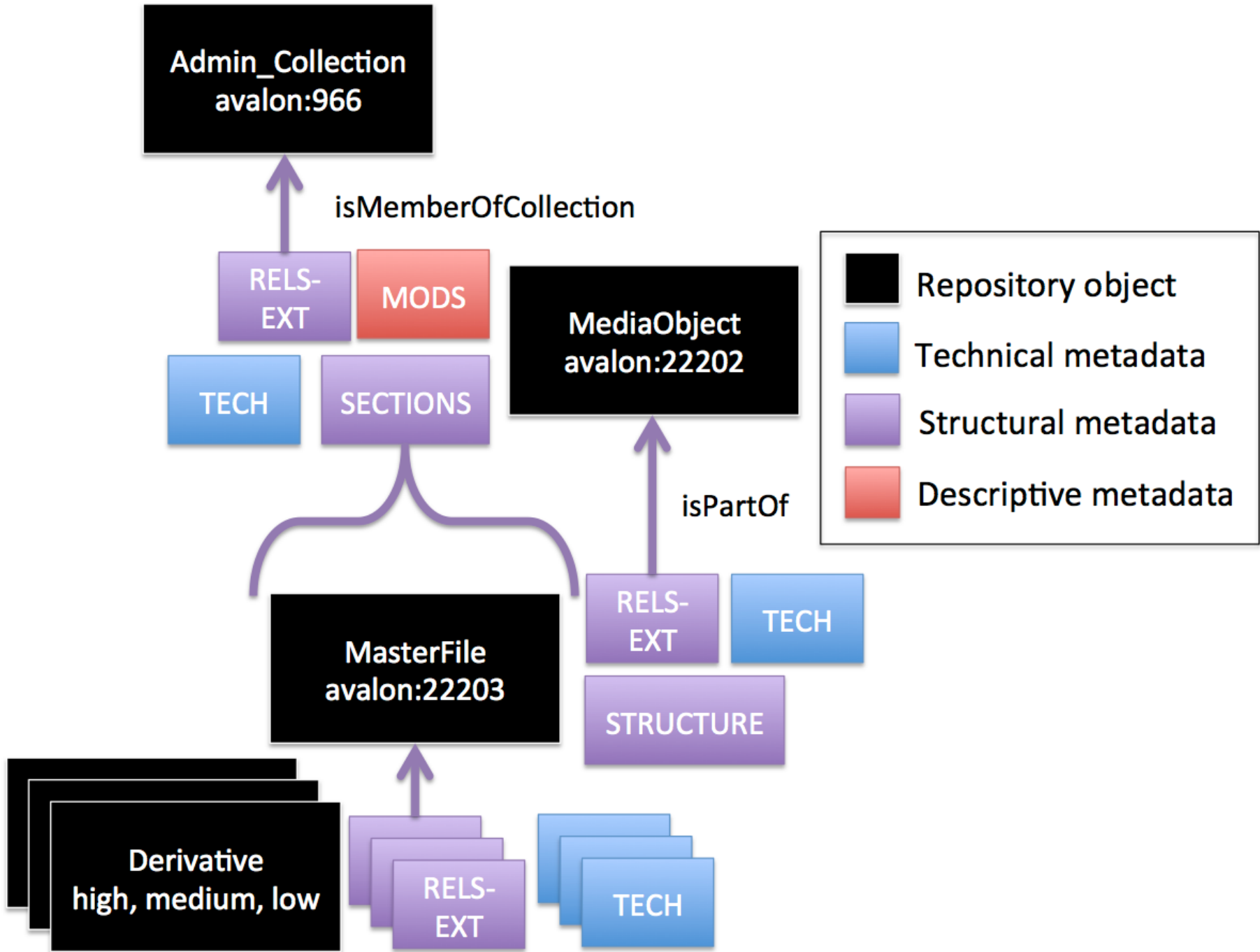


Figure 1. Diagram representing technical, structural, and descriptive metadata on Avalon objects in Fedora 3. Each blue, purple, and red box represents a separate XML file accompanying the repository object.

Avalon Technical and Structural Metadata in Fedora 4

As part of the work that has already occurred for [Avalon Release 6](#), Avalon now works with Fedora 4. Technical metadata is now being stored as RDF properties on Avalon objects instead of in an XML file. Where custom XML defined file location, size, format, duration, checksum, and date digitized and was only usable within the context of Avalon, that information is now handled as RDF properties defined using established ontologies for those properties ([EBUCore](#), [PREMIS](#), and Library of Congress identifier types) along with a few locally-defined properties for internal use.

Figures 2-6 show the process of storing technical and structural metadata as RDF properties instead of flat XML.

```
1 <fields>
2 <file_format>Moving image</file_format>
3 <file_location>/path/to/file/avalon_7089-1797.mov</file_location>
4 <duration>43099</duration>
5 <display_aspect_ratio>1.3333333333333333</display_aspect_ratio>
6 <original_frame_size>720x486</original_frame_size>
7 <poster_offset>8353</poster_offset>
8 <thumbnail_offset>8353</thumbnail_offset>
9 <file_size>735569993</file_size>
10 <date_digitized>2014-12-08T19:03:59Z</date_digitized>
11 </fields>
```

Figure 2. A Fedora 3 example of Avalon technical metadata for a video object as custom XML. This is represented as the blue “TECH” boxes in Figure 1.

```
@prefix fedoraObject: <http://[LocalFedoraRepository]/>.
@prefix ldap: <http://www.w3.org/ns/ldap#>.
@prefix fedora: <http://fedora.info/definitions/v4/repository#>.
@prefix ebucores: <http://www.ebu.ch/metadata/ontologies/ebucore/ebucore#>.
@prefix avalon: <http://avalonmediasystem.org/rdf/vocab/master_file#>.
@prefix premis: <http://www.loc.gov/premis/rdf/v1#>.

fedoraObject:90/19/s2/51/9019s251h a ldap:RDFSource, ldap:Container, fedora:Container, fedora:Resource;
    ebucores:aspectRatio "1.3333333333333333";
    ebucores:duration "43099";
avalon:posterOffset "8353";
    avalon:thumbnailOffset "8353";
    ebucores:fileSize "735569993";
    ebucores:filename "file:///path/to/file/avalon_7089-1797.mov";
    ebucores:height 486;
    ebucores:width 720;
ebucores:locator "/path/to/file/avalon_7089-1797.mov";
    premis:hasFormatName "Moving image";
    ebucores:dateCreated "2014-12-08T19:03:59Z".
```

Figure 3. The same technical metadata as in Figure 2 rendered as RDF properties in Fedora 4 Avalon. Please note that the ontology namespace prefixes are provided for easier reading.

These technical properties can be mapped from Avalon's custom XML to simple RDF statements and do not require a modification in value ranges (i.e., properties that are defined with text values in XML can be mapped to properties that still use text values, also called literals, in RDF). Many of these values (duration, aspect ratio, height, width) are audio and video specific so EBUCore from the European Broadcasting Union was a useful ontology to express those properties. Mapping recommendations for more [general technical metadata properties](#) have been made within the Samvera Community for common properties such as filename and file size and those recommendations were followed as closely as possible.

Structural metadata in Avalon with Fedora 3 used RDF/XML to express connections between objects based on the [RELS-EXT relationships](#) that Fedora defines.

```

1 | <rdf:RDF xmlns:ns0="http://projecthydra.org/ns/relations#"
2 |   xmlns:ns1="info:fedora/fedora-system:def/model#"
3 |   xmlns:ns2="info:fedora/fedora-system:def/relations-external#"
4 |   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
5 |   <rdf:Description rdf:about="info:fedora/avalon:22203">
6 |     <ns0:hasModelVersion>R3</ns0:hasModelVersion>
7 |     <ns1:hasModel rdf:resource="info:fedora/afmodel:MasterFile"/>
8 |     <ns2:isPartOf rdf:resource="info:fedora/avalon:22202"/>
9 |   </rdf:Description>
10| </rdf:RDF>

```

Figure 4. The RELS-EXT relationship in Fedora 3 as RDF/XML for an Avalon MasterFile object (avalon:22203). This connects the MasterFile to the Avalon MediaObject (avalon:22202) that contains the MasterFile. This is represented as the purple “RELS-EXT” box on the MasterFile and the “isPartOf” arrow in Figure 1.

```

1 | <rdf:RDF xmlns:ns0="http://projecthydra.org/ns/relations#"
2 |   xmlns:ns1="info:fedora/fedora-system:def/model#"
3 |   xmlns:ns2="info:fedora/fedora-system:def/relations-external#"
4 |   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
5 |   <rdf:Description rdf:about="info:fedora/avalon:22202">
6 |     <ns0:hasModelVersion>R4</ns0:hasModelVersion>
7 |     <ns0:isGovernedBy rdf:resource="info:fedora/avalon:966"/>
8 |     <ns1:hasModel rdf:resource="info:fedora/afmodel:MediaObject"/>
9 |     <ns2:isMemberOfCollection rdf:resource="info:fedora/avalon:966"/>
10|   </rdf:Description>
11| </rdf:RDF>

```

Figure 5. The RELS-EXT relationship defined on the Avalon MediaObject (avalon:22202) connecting it to an Avalon Admin_Collection (avalon:966) in Fedora 3 as RDF/XML. This is represented as the purple “RELS-EXT” box on the MediaObject and the “isMemberOfCollection” arrow in Figure 1.

```

1 | <rdf:RDF xmlns:ns0="info:fedora/fedora-system:def/model#"
2 |   xmlns:ns1="http://projecthydra.org/ns/relations#"
3 |   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
4 |   <rdf:Description rdf:about="info:fedora/avalon:966">
5 |     <ns1:hasModelVersion>R3</ns1:hasModelVersion>
6 |     <ns0:hasModel rdf:resource="info:fedora/afmodel:Admin_Collection"/>
7 |   </rdf:Description>
8 | </rdf:RDF>

```

Figure 6. The Admin_Collection (avalon:966) is defined as not being part of anything else – it is the top level.

The relationships defined in Avalon by Fedora 3 start from the lowest level of the object – the digital file – and point up to the highest level – the collection. The collection does not contain any structural definition of the objects contained within it, and the objects also do not contain any structural definition of the files that make up those objects. At the MediaObject level Avalon creates custom XML to define the files, or sections, that make up a single Avalon item so that those are available more easily and have a specified order when accessing the MediaObject:

```

1 | <fields>

```

```

2 | <section_pid>avalon:22203</section_pid>
3 | <section_pid>avalon:22204</section_pid>
4 | <section_pid>avalon:22205</section_pid>
5 | </fields>

```

Figure 7. Sections XML metadata on an Avalon MediaObject in Fedora 3. This is represented as the purple “SECTIONS” box in Figure 1.

In Fedora 4 those RELS-EXT relationships remain as RDF properties on the MasterFile and the MediaObject, but they are no longer stored as RDF/XML (as shown in Figure 1) and are augmented by other ontologies that allow for more and bidirectional structural definitions (see Figure 8). A MasterFile is still part of a MediaObject, which is still a member of an Admin_Collection using those RELS-EXT RDF properties. Dublin Core’s *hasPart* property is used on the MediaObject to identify the MasterFile objects that are part of that MediaObject. Additionally, the MediaObject is able to use what is called a *list_source* object to define an ordered list of objects that serve as proxies for MasterFiles contained within the MediaObject and places them in a certain order (important for tracks on an album or acts in movie). The [Linked Data Platform \(LDP\)](#) vocabulary and the [Internet Assigned Numbers Authority’s \(IANA\)](#) ordering properties are both ontologies that Fedora 4 can use to establish these connections from a higher level like the MediaObject down to the MasterFiles contained within instead of only the structural direction from lowest level upwards.

```

@prefix fedoraObject <http://[LocalFedoraRepository]/>.
@prefix fedora: <http://fedora.info/definitions/v4/repository#>.
@prefix ldp: <http://www.w3.org/ns/ldp#>.
@prefix avalonMigration: <http://avalonmediasystem.org/ns/migration#>.
@prefix fedoraModel: <info:fedora/fedora-system:def/model#>.
@prefix dcterms: <http://purl.org/dc/terms/>.
@prefix fedoraRELSEXT: <info:fedora/fedora-system:def/relations-external#>
@prefix iana: <http://www.iana.org/assignments/relation/>.

fedoraObject:tq/57/nr/06/tq57nr067
  a fedora:Container, ldp:RDFSource, ldp:Container, fedora:Resource;
  avalonMigration:migratedFrom"avalon:22202";
  fedoraModel:hasModel"MediaObject" ;
  dcterms:hasPart fedoraObject:dr/26/xx/45/dr26xx45j, fedoraObject:nv/93/52/90/nv9352905, fedoraObject:2v/23/vt/45/2v23vt451;
  fedoraRELSEXT:isMemberOfCollection fedoraObject:dj/52/w4/73/dj52w4734;

  ldp:contains <http://[FedoraAggregation]/tq/57/nr/06/tq57nr067/list_source>;
  iana:first <http://[FedoraObjectProxy]/tq/57/nr/06/tq57nr067/list_source#g284616160>;
  iana:last <http://[FedoraObjectProxy]/tq/57/nr/06/tq57nr067/list_source#g309531160>.

```

Figure 8. Structural RDF properties on an Avalon MediaObject in Fedora 4 showing that it has three parts (Avalon sections or MasterFiles).

This bidirectional structure will be further enhanced as Avalon implements the Portland Common Data Model (PCDM). PCDM is a domain model meant to provide an interoperable framework for working with repository objects. Its current focus is providing structural metadata and access controls for better interoperability between repository applications. (Portland Common Data Model [updated 2016 Aug 9]) Fedora 4 does not automatically apply PCDM properties and relationships to objects, however, in addition to LDP Containers and Indirect Containers, PCDM allows the definition of Collections, Objects, and Files, providing another option for representing the structure of Avalon objects and collections of those objects.

Derivatives, or compressed audio and video files that can be streamed online at different bandwidths, produced off of each MasterFile are also stored as objects in Fedora with high and medium qualities created for audio and high, medium, and low qualities created for video. Each derivative is connected back to its MasterFile object using a RELS-EXT relationship defined on the Derivative object only, so there is no bidirectional connection. This single relationship maps over to an RDF statement on the Derivative object in Fedora 4. Custom XML in Fedora 3 defined technical properties like encoding qualities and file locations. Where possible in Fedora 4 EBUCore properties were used to map this information to RDF and all values were mapped as literals (strings of text).

Collections and Units have come across from Fedora 3 to Fedora 4 but the mapping is not ideal. At the MediaObject level, the same RELS-EXT RDF property is used to connect to the collection, which is its own repository object and can be referenced using a URI.

```

@prefix fedoraModel: <info:fedora/fedora-system:def/model#>.
@prefix loc: <http://id.loc.gov/vocabulary/identifiers/>.
@prefix dcterms: <http://purl.org/dc/terms/>.
@prefix bf: <http://bibframe.org/vocab/>.

<fedoraObject:dj/52/w4/73/dj52w4734> a ldp:Container, fedora:Resource, fedora:Container, ldp:RDFSsource;
  fedoraModel:hasModel "Admin::Collection";
  loc:local "avalon:966";
  dcterms:title "JSOM Performance";
  bf:heldBy "Music Library".

```

Figure 9. RDF statements stored on a collection object in Fedora 4.

The least ideal portion of Figure 9’s mapping is the handling of Avalon units. Units are the top level of structure within Avalon but collections are the top level of objects in the repository (see Figure 1). So units are used for grouping and access, but not handled as repository objects. We can still manage units this way in Fedora 4, but to map them to explain what they are semantically in RDF means they need to be defined by a property at the collection level that shows the collection is contained in, is part of, or is held by the unit. The level of unit in Avalon can be equivalent to something that is identifiable as a thing in an authorized controlled vocabulary, such as LCNAF or VIAF (for example, Indiana University School of Music or Northwestern University Archives). A property with that sort of semantic meaning but requiring a URI value will work, which is why the Fedora 4 mapping is set to “heldBy” from the Bibframe ontology. Units in Avalon are based on hard-coded configurations at this point and changes are still needed to change the value to a URI for Fedora storage. Additionally, this Bibframe property is still pointing to the [1.0 version of the ontology](#), which has been replaced by [Bibframe 2.0](#). This mapping is still considered a work in progress.

One structural definition that neither Fedora 3 nor Fedora 4 provide for is defining structure within a single digitized file (within one MasterFile). Audio and video files often have multiple pieces of content on a single digital file (multiple songs, multiple acts of an opera or performance, etc.).

```

1 <Item label="Tape 1">
2   <Span begin="0:0:0" end="0:1:0" label="Track 1. [ambience]"/>
3   <Span begin="0:1:0" end="0:16:19" label="Track 2. Ludwig van Beethoven, Twelve Variations on a Theme from Judas Maccabaeus (Handel)">
4     <Div label="Johannes Brahms, Sonata in F major, Op. 99">
5       <Span begin="0:16:19" end="0:23:10" label="Track 3. I. Allegro vivace"/>
6     </Div>
7 </Item>

```

Figure 10. Custom structural XML defining structure within a MasterFile object, used in Fedora 3 and Fedora 4. This is represented as the purple “STRUCTURE” box in Figure 1.

To enable this structural definition Avalon uses another set of custom XML that is stored in both Fedora 3 and Fedora 4 as an XML file with the MasterFile object, using the same custom XML. There are no RDF properties that can express the needed structure with layers of labels, start times, and end times defined within a single file. This custom XML is closely based on XML used for the same purpose in the Variations Digital Music Library System, a digital audio and score access system developed at Indiana University that in many ways is the predecessor to Avalon Media System.

Mapping from XML to RDF for technical metadata has not resulted in much that takes advantage of Linked Data, but we do now use common ontologies for the properties describing that technical metadata instead of customized XML that could only be understood and used in the context of Avalon. Our structural metadata makes the same use of Linked Data relationships as it did before through Fedora’s RELS-EXT connections from files up to objects up to collections, but it is also expanded to allow connections in the opposite direction. An external triplestore might not require such explicit property statements if a [semantic reasoner](#) is available, but it is helpful in Fedora 4 to provide statements at the object level connecting to the files that make up the object. Hard-coded units have the potential for semantic mapping but the implementation is not yet complete.

Descriptive Metadata and Avalon

Our analysis of descriptive metadata began by looking at the mapping recommendations from the Samvera [MODS and RDF Descriptive Metadata Subgroup](#). Avalon's current move to Fedora 4 has kept our descriptive metadata stored as a flat MODS XML file without RDF properties. Figure 1 represents this XML file as the red "MODS" box on the MediaObject. Changing this MODS XML to RDF is another step in the overall metadata migration process. In preparation for allowing our descriptive metadata to be stored as RDF properties with the digital object we created a spreadsheet mapping our current MODS XML elements in comparison to the MODS and RDF Subgroup's recommendations. [2] While a [MODSRDF ontology](#) from the Library of Congress does exist, it is still a work in progress and often results in complex hierarchical RDF (where the object of one RDF statement is another RDF statement). This complexity would make storing descriptive metadata in Fedora about a single digital object more difficult than considering a use of mixed ontologies to express the same meaning of the MODS XML elements as simple RDF statements. Another consideration is that Avalon MODS does not currently allow for reference identifiers to be easily added to our fields. This reality means for migration purposes using a single ontology would likely require more reconciliation of text values over to URI values. Considering properties that can allow for text values means more flexibility in how we are mapping and thus involves a variety of ontologies.

The Samvera Community is developing a new "open-source Hydra-powered [sic] repository front end" called [Hyrax](#). Hyrax is the merger of two Samvera gems – [Sufia](#) and [CurationConcerns](#). While Avalon only allows for audio and video content to be deposited, Hyrax is meant to be customizable to allow for other types of content as needed. Current Avalon development is moving towards integration with Hyrax, which is built on Fedora 4 and uses RDF properties for descriptive metadata. Another step in mapping our descriptive metadata to RDF then was to consider the RDF properties that Hyrax is using. However, our analysis of Hyrax's generic type of work and descriptive metadata mapping shows that only 48%, or 12 out of 25 of Avalon's descriptive metadata MODS XML elements match. [3]

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <mods xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.loc.gov/mods/v3" xsi:schemaLocation="http://www.loc.gov
3   <titleInfo usage="primary">
4     <title>Freshwater jellyfish</title>
5   </titleInfo>
6   <name type="personal" usage="primary">
7     <namePart>Indiana University, Bloomington. Audio-Visual Center</namePart>
8     <role>
9       <roleTerm authority="marcrelator" type="code">cre</roleTerm>
10      <roleTerm authority="marcrelator" type="text">Creator</roleTerm>
11    </role>
12  </name>
13  <name type="personal">
14    <namePart>Flaten, Clarence M., 1910-1974</namePart>
15    <role>
16      <roleTerm authority="marcrelator" type="code">ctb</roleTerm>
17      <roleTerm authority="marcrelator" type="text">Contributor</roleTerm>
18    </role>
19  </name>
20  <name type="personal">
21    <namePart>Vuke, George J.</namePart>
22    <role>
23      <roleTerm authority="marcrelator" type="code">ctb</roleTerm>
24      <roleTerm authority="marcrelator" type="text">Contributor</roleTerm>
25    </role>
26  </name>
27  <name type="personal">
28    <namePart>Lytle, Charles F.</namePart>
29    <role>
30      <roleTerm authority="marcrelator" type="code">ctb</roleTerm>
31      <roleTerm authority="marcrelator" type="text">Contributor</roleTerm>
32    </role>
33  </name>
34  <name type="personal">
```



```
35     <namePart>Indiana University, Bloomington. Audio-Visual Center</namePart>
36     <role>
37         <roleTerm authority="marcrelator" type="code">ctb</roleTerm>
38         <roleTerm authority="marcrelator" type="text">Contributor</roleTerm>
39     </role>
40 </name>
41 <typeOfResource>moving image</typeOfResource>
42 <genre>Nature films.</genre>
43 <genre>Science films.</genre>
44 <genre>Educational films.</genre>
45 <genre>Nonfiction films.</genre>
46 <genre>Short films.</genre>
47 <originInfo>
48     <dateIssued encoding="edtf">1978</dateIssued>
49     <publisher>Indiana University Audio-Visual Center</publisher>
50 </originInfo>
51 <language>
52     <languageTerm type="code">eng</languageTerm>
53     <languageTerm type="text">English</languageTerm>
54 </language>
55 <physicalDescription>
56     <internetMediaType>video/quicktime</internetMediaType>
57 </physicalDescription>
58 <abstract>Examines the life cycle of the freshwater jellyfish and describes its habitats, collection procedures, feeding behavior
59 <note type="statement of responsibility" altRepGroup="00">produced by Indiana University Audio-Visual Center.</note>
60 <note type="creation/production credits">Producers, Clarence M. Flaten, George Vuke; consultant, Charles Lytle.</note>
61 <note type="additional physical form">Also issued as videocassette.</note>
62 <subject>
63     <topic>Craspedacusta sowerbyi</topic>
64 </subject>
65 <relatedItem type="original">
66     <physicalDescription>
67
68
69 <form authority="marccategory">motion picture</form>
70
71
72 <form authority="marcsmd">film reel</form>
73
74
75 <form type="media" authority="rdamedia">projected</form>
76
77
78 <form type="carrier" authority="rdacarrrier">film reel</form>
79
80
81 <extent>1 film reel (13 min.) : sound, color ; 16 mm</extent>
82 </physicalDescription>
83     <identifier type="lccn">79700435</identifier>
84     <identifier type="oclc">ocm05892454</identifier>
85     <identifier type="oclc">5892454</identifier>
86     <identifier type="local">7830578</identifier>
87 </relatedItem>
88 <location>
89     <url displayLabel="Digitized version" usage="primary display">http://purl.dlib.indiana.edu/iudl/media/653702wt32</url>
90 </location>
91 </location>
```



```

92     <url access="object in context">http://purl.dlib.indiana.edu/iudl/media/653702wt32</url>
93 </location>
94 <recordInfo>
95     <descriptionStandard>rda</descriptionStandard>
96     <recordContentSource authority="marcorg">Indiana Univ. Audio-Visual Center.</recordContentSource>
97     <recordCreationDate encoding="edtf">790718</recordCreationDate>
98     <recordChangeDate encoding="iso8601">2015-10-04T09:05:13-04:00</recordChangeDate>
99     <recordOrigin>Converted from MARCXML to MODS version 3.5 using MARC21slim2MODS3-5.xsl (Revision 1.106 2014/12/19), custom
100     <languageOfCataloging>
101         <languageTerm authority="iso639-2b" type="code">eng</languageTerm>
102     </languageOfCataloging>
103     <recordIdentifier source="local">7830578</recordIdentifier>
104     <recordIdentifier source="Fedora">avalon:6696</recordIdentifier>
105 </recordInfo>
106 </mods>

```

Figure 11. Example of a MODS XML record stored with an Avalon item in Fedora 3. It does not represent every possible MODS element that Avalon records, but offers a representation of the XML elements with which we are working.

We will now provide details on evaluations and decision points for Avalon’s descriptive metadata fields as we consider taking them from MODS XML and making them RDF properties stored with digital objects in Fedora 4. We want to have the ability to use URIs as much as possible instead of literal text to allow our metadata to be used as Linked Data.

Names

@prefix dc: <http://purl.org/dc/elements/1.1/> .

@prefix relator: <http://id.loc.gov/vocabulary/relators/> .

| Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|--|---------------------|----------------|-----------------|
| name@usage="primary"/namePart (role/roleTerm set to "Creator") | relator:cre | dc:creator | dc:creator |
| name/namePart (role/roleTerm set to "Contributor") | relator:ctb | dc:contributor | dc:contributor |

In order to use the MODS and RDF Subgroup mapping recommendation for names, we would either have to resolve the names for Creator and Contributor fields to URIs or create local URIs for names. Unfortunately many of the names entered by just IU and Northwestern as examples will not easily be resolved so we made the decision to deviate from the recommended MODS and RDF Subgroup mapping of relator:cre and relator:ctb to dc:creator and dc:contributor to allow for text values. Hyrax also maps Creator to dc:creator and Contributor to dc:contributor.

Dates

@prefix dcterms: <http://purl.org/dc/terms/> .

| Date Fields | Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|------------------|--|---------------------|-----------------|-----------------|
| Publication Date | originInfo/issueDate | dcterms:issued | dcterms:created | dcterms:issued |
| Creation Date | originInfo/dateCreated@encoding="edtf" | dcterms:created | NONE | dcterms:created |

Avalon currently uses `originInfo/dateIssued` within MODS to store the publication date for an item. The MODS and RDF Subgroup recommends mapping to `dcterms:issued` while Hyrax uses `dcterms:created`. We will have to change how our dates are stored as well as update documentation and the sorting of the facets if we change to using Hyrax's mapping of `dcterms:created`. We are likely to try matching this to the MODS and RDF Subgroup recommendation of `dcterms:issued` for Publication Date and `dcterms:created` for Creation Date.

Language

@prefix dc: <<http://purl.org/dc/elements/1.1/>> .

@prefix dcterms: <<http://purl.org/dc/terms/>> .

| Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|-----------------------|---------------------|-------------|------------------|
| language/languageTerm | dcterms:language | dc:language | dcterms:language |

Hyrax uses `dc:language` (from the DC element 1.1 set) while the MODS and RDF Subgroup recommends mapping to `dcterms:language`. Both recommend use of a controlled vocabulary for the value but `dcterms:language` specifies that the value needs to be of the `LinguisticSystem` class, so a URI value instead of a text value. The MODS and RDF Subgroup is mapping these to [ISO639-2](#) values and Avalon uses [MARC Language codes](#). The result is that Avalon can migrate this content to a URI value (like <http://id.loc.gov/vocabulary/languages/eng>) so `dcterms:language` will work best for this mapping.

Physical Description

@prefix bibframe: <<http://bibframe.org/vocab/>> .

| Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|--|---------------------|-------|-----------------|
| relatedItem@type="original"/physicalDescription/extent | bibframe:extent | NONE | bibframe:extent |

The MODS and RDF Subgroup is mapping to the [“extent” property from Bibframe 1.0](#) to allow for text values ([Bibframe 2.0’s “extent” property](#) requires a URI value). Another potential property that allows for the use of text values is [Bibframe Lite’s “extent.”](#) Hyrax does not offer a field for physical description so this will be an addition or customization for Avalon. There will have to be a decision made as to which property will be better to use, although there is no question that these extent values will have to remain as text.

Related Items URL/Label

@prefix rdfs: <<http://www.w3.org/2000/01/rdf-schema#>> .

| Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|---------------------------------------|---------------------|--------------|-----------------|
| relatedItem@displayLabel/location/url | NONE | rdfs:seeAlso | rdfs:seeAlso |
| relatedItem@displayLabel | NONE | NONE | rdfs:label |

The MODS and RDF Subgroup does not have a mapping for this particular case of a related item URL. Hyrax uses `rdfs:seeAlso` and the value will always be a URI so that mapping can work for a related item link.

The text label for the link is a more difficult case to handle. In Avalon MODS XML, this text is encoded as a `displayLabel` attribute on `<relatedItem>` so this mapping is not straightforward. The `rdfs:label` property is probably not specific enough to identify a label for the related item URL but using `rdfs:label` with `rdfs:seeAlso` ties these pieces

together, so even though the namespace is from a more general-use ontology it can work in the Avalon/Samvera context. There is still a problem, however, if there is more than one related item; which rdfs:label goes with which rdfs:seeAlso URL? Since related item links are really just links and not Linked Data objects themselves, there is no guarantee of being able to find a property with a text label at the resolved URL for use in Avalon. Hyrax only shows the URL of the link so it might be best, though not ideal, to migrate only the URL and not the label text in Fedora 4 with RDF. To maintain our current Avalon functionality would mean possibly creating empty objects in Fedora 4 to manage a text label for related item links.

Genre

@prefix edm: <http://www.europeana.eu/schemas/edm/> .

| Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|-------------|---------------------|-------|-----------------|
| genre | edm:hasType | NONE | edm:hasType |

The MODS and RDF Subgroup recommends mapping genres to edm:hasType, which requires a URI for a value. To use this mapping will require Avalon to reconcile any values stored in the Genre field in Fedora 3 before migrating to Fedora 4. This is worth it to try reconciling. If text values fail to reconcile to URI values, they would need to be updated so they could reconcile before migration could complete.

Subjects

@prefix bf2: <http://id.loc.gov/ontologies/bibframe/> .

@prefix dc: <http://purl.org/dc/elements/1.1/> .

@prefix dcterms: <http://purl.org/dc/terms/> .

@prefix foaf: <http://xmlns.com/foaf/0.1/> .

| Subject Type | Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|--------------|--------------------|----------------------|-----------------|----------------------|
| Topical | subject/topic | dcterms:subject | dc:subject | dc:subject |
| Geographic | subject/geographic | dcterms:spatial | foaf:based_near | dcterms:spatial |
| Temporal | subject/temporal | bf2:temporalCoverage | NONE | bf2:temporalCoverage |

Topical Subjects: Ideally we would like to use the MODS and RDF Subgroup recommendation of dcterms:subject and require URIs, but that would mean external reconciliation, which is probably not realistic due the current practice of not requiring a particular controlled vocabulary. Thus, subjects have been entered more like keywords. We would need to use the text-based dc:subject to make migration easier from this field.

Geographic Subjects: Similarly to topical subjects, we do not require a controlled vocabulary for geographic subjects. This field is much more likely to be reconcilable against a location vocabulary like the Thesaurus of Geographic Names, so we think external reconciliation would be easier to manage. This will allow Avalon to have URIs available for the value and use the MODS and RDF Subgroup recommendation of dcterms:spatial. Ideally we would also be able to provide a report of records that did not reconcile post-migration. Hyrax maps Geographic Subject to foaf:based_near as Location. The GeoNames web service is supposed to be behind this field, but regardless of entering text or a URI, only text is saved in Fedora. Work is happening to add controlled vocabulary selection to the [Questioning Authority](#) gem behind this feature, so even though this functionality is not working in Hyrax out of the box, it is in progress.

Temporal Subjects: bf2:temporalCoverage takes a literal value. We ask for [Extended Date Time Format \(EDTF\)](#) values in this field but we have not enforced this in anyway. Temporal subjects are not indexed for date faceting so the EDTF format is not required for this field to be migrated to the Bibframe 2.0 property.

Terms of Use

@prefix dc: <http://purl.org/dc/elements/1.1/> .

@prefix edm: <http://www.europeana.eu/schemas/edm/> .

| Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|---|---------------------|------------|--------------------------|
| accessCondition@type="use and reproduction" | dc:rights | edm:rights | dc:rights and edm:rights |

This field contains text statements regarding rights instead of a URI value like that from Rightsstatement.org. A separate field for a Rights URI should be offered as part of the migration process and would make these items more externally shareable to places like the Digital Public Library of America and Europeana. But for the text values that need to migrate, dc:rights can take a text value. This might be an example of using both the MODS and RDF Subgroup recommendation and the Hyrax property as separate RDF statements on the digital object.

Statement of Responsibility

@prefix skos: <http://www.w3.org/2004/02/skos/core#> .

| Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|---|---------------------|-------|-----------------|
| note@type="statement of responsibility" | skos:note | NONE | skos:note |

MODS records have notes elements of varying types. The MODS and RDF Subgroup's simple mapping uses the skos:note property and appends the "type" attribute to the beginning of the text value of the note as a label. It is likely that Avalon will follow this recommendation to bring over the statement of responsibility as well as other types of notes that are offered.

Notes

@prefix skos: <http://www.w3.org/2004/02/skos/core#> .

| Note Fields | Avalon MODS | MODS & RDF Subgroup | Hyrax | Avalon MODS/RDF |
|-------------|-------------|---------------------|-------|-----------------|
| Note | note | skos:note | NONE | skos:note |
| Note Type | note@type | skos:note | NONE | skos:note |

If there is a "type" attribute for any note then that value becomes a label added to the note text with a colon: "[@type value]: [note text]"

Identifiers

@prefix dc: <http://purl.org/dc/elements/1.1/> .

@prefix edm: <http://www.europeana.eu/schemas/edm/> .

@prefix loc: <http://id.loc.gov/vocabulary/identifiers/> .

@prefix owl: <http://www.w3.org/2002/07/owl#> .

@prefix prov: <http://www.w3.org/ns/prov#> .

| | | | | |
|--|---|---------------|---------------|---------------------|
| Identifier Fields | MODS & RDF SubgroupHyraxAvalon MODS/RDF | | | |
| avalon:pid | avalon:pid (migration only) | NONE | NONE | prov:wasDerivedFrom |
| Permalink (MediaObject and MasterFile) | location/url@access="object in context" | edm:isShownAt | dc:identifier | dc:identifier |
| Bibliographic ID | relatedItem@type="original"/identifier | loc:local | NONE | owl:sameAs |
| Other Identifier | relatedItem@type="original"/identifier | NONE | NONE | loc:local[19] |

Migration of Avalon metadata from Fedora 3 to 4 means we will have new local Fedora identifiers in our Avalon instances. To not lose the Fedora 3-based unique identifiers (PIDs) we are mapping that data as a provenance property to contain a link to the Fedora 3 object. Even though it more than likely will not exist after migration completes, this fits the value expected from the PROV-O ontology and acts as a record of the path on which the digital object has traveled. This only affects migrated items.

Fedora 3 PID: avalon:19920

<Fedora 4 object> prov:wasDerivedFrom <[http://\[fedora3URL\]/avalon:19920](http://[fedora3URL]/avalon:19920)>.

Avalon has never created a unique identifier as part of the MODS XML record stored for objects. We do, however, have a Permalink field for all published items that creates a unique URL for each MediaObject and for each MasterFile. In Avalon's current release with Fedora 4, this property is mapped to a custom Samvera Community property, "hasPermalink." Hyrax uses dcterms:identifier for an optional Identifier field (defined as: "A unique handle identifying the work. An example would be a DOI for a journal article, or an ISBN or OCLC number for a book."). We are choosing to use dc:identifier as Avalon's Permalink field since we will treat the Permalink as the unique identifier and it will not be the same property as Hyrax's optional Identifier field. This is part of the proposal outlined for different Avalon identifier properties that has yet to be implemented. [4]

Avalon allows for the import of descriptive metadata by Bibliographic ID, such as through a Z39.50 web service or other configured service. The MODS and RDF Subgroup mapped this type of identifier to loc:local. However, Avalon's connection from digital objects to library catalog records mean the Avalon items are the same as the items described in the catalog, either in the same format or in a digitized format. Additionally, there are different types of identifiers that can be connected to an Avalon item (OCLC numbers, music publisher numbers, video recording numbers, local record numbers). The loc:local identifier type from the Library of Congress makes more sense in this context reserved for those identifiers that are specifically local to a collection. For those reasons owl:sameAs works better to connect a bibliographic identifier that is used to import library catalog information into Avalon, and loc:local can then be used for local identifiers that do not match with the other Library of Congress identifiers.

Conclusion

What Avalon has managed to do most effectively so far in migrating from XML to RDF is make use of properties from common ontologies and keep RDF statements simple. Technical metadata statements are producing values that are text-based instead of Linked Data URIs. Structural metadata is taking advantage of Linked Data values to connect Fedora 4 objects to each other but structure within a single file remains as custom XML. Descriptive metadata mapping shows the most possibility for engaging with Linked Data, but the transition will most likely be a process taken in steps. Languages and Related Items can definitely become Linked Data statements using URIs for values and offering potential semantic web connections. Geographic Subjects also seem likely to be reconcilable to Linked Data URIs from a controlled vocabulary, making those terms usable as Linked Data. This could also provide a way to visualize location information for mapping visualizations. Genre is another field that seems to need reconciliation (it will need a URI value) in order to be brought over as an appropriately mapped property in RDF. Beyond those fields, however, the descriptive metadata stored in Avalon is either not based on Linked Data objects, such as Terms of Use and Notes, or there are not enough requirements on the data values to make it possible to reconcile text strings to URI values, like Names and Topical Subjects.

The reasons for moving Avalon metadata to RDF are two-fold: conform to community standards and requirements of the software we are using, and the desire to enhance our metadata for better meaning and reuse through Linked Data. The results in mapping efforts so far have been mixed. Most of the data already migrated has come across

as RDF but not as Linked Data. Descriptive metadata mapping promises more possibilities with Linked Data but implementing these mappings will require a stepped progression to move away from the way MODS XML encoded values to a meaningful RDF statement that can be used to describe the digital object in Avalon.

Notes

[1] On May 24, 2017 the Hydra Project formally changed its name to the Samvera Community. <https://projecthydra.org/2017/05/24/hydras-new-name-is-samvera/>

[2] Avalon MODS comparison to MODS and RDF Subgroup mapping recommendations: <https://goo.gl/erh5jR>

[3] Avalon MODS compared to Hyrax RDF properties compared to MODS and RDF Subgroup mapping recommendations: <https://goo.gl/rk17NB/a>>

[4] Avalon Identifiers Proposal: <https://goo.gl/fDMJEE>

References

Portland Common Data Model [Internet]. [updated 2016 Aug 9]. PCDM Wiki; [cited 2017 May 25]. Available from: <https://github.com/duraspace/pcdm/wiki>

The Project [Internet]. [updated 2017]. Bloomington (IN): Avalon Media System; [cited 2017 May 25]. Available from: <http://www.avalonmediasystem.org/project>

About the Authors

Juliet L. Hardesty, jlhardes@iu.edu

Julie Hardesty is the Metadata Analyst/Librarian for the Indiana University Libraries where she manages metadata creation and use for digital library services and projects. She helps develop descriptive, structural, and technical metadata standards for the Avalon Media System with Jennifer and has been part of the project since 2012.

Jennifer B. Young, j-young2@northwestern.edu

Jennifer B. Young is the Metadata Coordinator at Northwestern University Libraries managing the creation of metadata for NUL's digital repositories. As a metadata expert for the Avalon Media System, Jennifer works with Julie to make decisions regarding all aspects of metadata. She has been with the project since 2016.

Subscribe to comments: [For this article](#) | [For all articles](#)

This work is licensed under a [Creative Commons Attribution 3.0 United States License](https://creativecommons.org/licenses/by/3.0/).

