Hypatia content and layout for Sets and Items

Here is a summary of expected datastreams and relationships to be created for the Summer 2011 phase of Hypatia deployment.

Datastream	Set	Item	Child of atomistic object
identityMetadata	Has <objectType>set</objectType>	Has <objectType>item</objectType>	Minimal, to record objectType objectLabel and uuid
descMetadata	EAD to MODS mapping, with <mods:typeOfResource collection="yes"/>	EAD to MODS mapping or FTK to MODS mapping	n/a
sourceMetadata	n/a	optional	n/a
provenanceMetadata	n/a	Defer	n/a
technicalMetadata	n/a	Defer; part of object creation, validation and preservation workflows.	possible during object assembly until integrated into parent item record; not part of final Hypatia objects
contentMetadata	n/a; "content" is defined by the set's membership	Using Stanford schema	possible during object assembly until integrated into parent item record; not part of final Hypatia objects
rightsMetadata	Public template	Public template	n/a
RELS-EXT	hydra:hasModel=implicitSet	*fedora:hasModel*=hydra:genericParent *fedora:hasModel*=hydra:commonMetadata *hydra:isGovernedBy* relationship to an Admin Policy object *fedora:isMemberOf* relationship to any immediate aggregate set of which this object is a member *fedora:isMemberOfCollection* relationship to top "collection" set	*fedora:hasModel*=hydra:genericContent ... or ... *isPartOf* relationship to parent item
content	n/a	n/a, that is to say, Hypatia fully embraces the atomistic model, separating the parent metadata-bearing object from child content-bearing object(s), even in cases off single-file content. This is not a given across Hydra in general, but a simplifying assumption for early Hypatia.	For Hypatia we can use both the simple genericContent model where each child object bears a single file, or the compoundContent model where multiple content files in the child are in content01, content02, etc. Where the content file is an image, the staticImage model may be appropriate.

Issues

A few Stanford-centric features may not be of relevance ot of interest to other Hypatia partners. What accommodation, if any, is needed for Stanford vs generic software?

- Stanford identityMetadata datastream
- Stanford-specific extensions to contentMetadata, e.g., shelve, publish and preserve directives
- Stanford has all content in its externally managed "Digital Stacks"
  - Child objects have no *content* datastreams (and in some cases may be obviated)
  - contentMetadata <location> URLs will point to stacks delivery services
  - Shared demo Hypatia objects need to be locally managed and self-sufficient, without reliance on Stanford environment

Sharing

Method of contributing/sharing of objects to be determined by the practitioners along the lines of whatever works. Current thinking is that FOXML is the best lowest common denominator, or perhaps working directly with a dev hypatia repository where possible.

identityMetadata

Stanford-specific. Will appear in Stanford produced fixtures and demo objects; not required otherwise.

```
<identityMetadata>
  <objectId>druid:zz923jk342</objectId>
  <objectType>set | item</objectType>
  <objectLabel>EAD collection ...</objectLabel>
  <objectCreator>DOR</objectCreator>
  <sourceId source="spec">M1437</sourceId>
  <otherId name="uuid">7f3da130-7b02-11de-8a39-0800200c9a66</otherId>
  <agreementId>druid:yh72ms9133</agreementId>
  <tag>Hypatia : Gould</tag>
</identityMetadata>
```

descMetadata

Per Hypatia EAD conversion analysis

sourceMetadata

This may be supplied as part of creating the object to meet repository preservation requirements for separately recording information about the source media. Stanford is eamining the Xanadu and Gould output for information that might be placed here.

provenanceMetadata

What history is to be captured as provenance to complete the full ingest story for a repository is a workflow/preservation consideration, not applicable to the first round of shared Hypatia demo objects.

technicalMetadata

The digitization of the artifacts carrying born digital materials produces one or more collateral files that describe the physical media and their contents. It is tempting (and not incorrect) to consider these files are forms of technical metadata. But similar to the way photos of a hard drive or diskette label might be considered metadata yet are treated as content to be delivered to the user, we believe bit-images and content analysis files such as those produced by the Forensics Toolkit (FTK) are in the realm of user-delivered content, to be shared as part of the understanding of the digital object. The technicalMetadata stream serves a slightly different purpose. Here is where we would place repository workflow output relevant to the characterization and validation of content files. For example, in technicalMetadata there might be JHOVE or similar output concerning the .jpg, .dd and .txt or .xml files associated with the object.

contentMetadata

See references to Stanford schema and Hull's version at the Hydra content models page. Sample contentMetadata XML is also online as part of the description of the Xanadu EAD and Hypatia fixture objects .

rightsMetadata

First round has publicly accessible objects only, using the following rightsMetadata:

```
<rightsMetadata>
  <access type="discover">
    <machine>
      <world/>
    </machine>
  </access>
  <access type="read">
    <machine>
     <world/>
    </machine>
  </access>
</rightsMetadata>
```

RELS-EXT

Discussion continues on richer RDF relationships that might benefit the Hypatia application. These might be expressed as hypatia relationships rather then generic hydra one, or may lead to useful global considerations for Hydra set relationships. Here is an example based on what we know now (mid-summer 2011):

```
<rdf:RDF xmlns:fedora-model="info:fedora/fedora-system:def/model#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:rel="info:fedora
/fedora-system:def/relations-external#" xmlns:hydra="http://projecthydra.org/ns/relations#">
 <rdf:Description rdf:about="info:fedora/hypatia:fixture_xanadu_drive1">
   <fedora-model:hasModel rdf:resource="info:fedora/hydra:commonMetadata"/>
   <hydra:isGovernedBy rdf:resource="info:fedora/hypatia:fixture_xanadu_apo"/>
   <isMemberOfCollection xmlns="info:fedora/fedora-system:def/relations-external#" rdf:resource="info:fedora/hypatia:fixture_xanadu_collection"/>
   <isMemberOf xmlns="info:fedora/fedora-system:def/relations-external#" rdf:resource="info:fedora/hypatia:fixture_xanadu_borndigital"/>  </rdf:
Description>
</rdf:RDF>
```