22 October 2014

Present

<table>
<thead>
<tr>
<th>Institution</th>
<th>Attendees</th>
</tr>
</thead>
<tbody>
<tr>
<td>University of Hull</td>
<td>Chris Awre, Richard Green, Simon Lamb, Diane Leeson</td>
</tr>
<tr>
<td>LSE</td>
<td>Neil Stewart, Andrew Amato (plus colleagues attending for parts)</td>
</tr>
<tr>
<td>Lancaster University</td>
<td>Masud Khokhar, Adrian Albin-Clarke, Hardy Schwamm</td>
</tr>
<tr>
<td>University of Durham</td>
<td>Matthew Phillips, Paul Dixon</td>
</tr>
<tr>
<td>University of Oxford</td>
<td>Anusha Ranganathan</td>
</tr>
<tr>
<td>University of York</td>
<td>Frank Feng</td>
</tr>
</tbody>
</table>

The meeting covered a variety of topics selected prior to the day itself by those attending. Notes of discussions under each topic are listed.

State of the HydraSphere

CA gave a presentation on the current state of the HydraSphere, using slides from the Hydra Connect conference put together by Tom Cramer and used with permission.

Installation

Institutions reported on their experiences of installing Hydra. Key points from this were:

- A combination of the Dive into Hydra and Rails for Zombies tutorials had worked well for Lancaster in installing HyHull. The dependency on Ruby 1.9.3 within this had been noted and adapted to, given that Ruby is currently at 2.1.3. (this is due to the timing of the HyHull work in 2013).
- Additional sources of information that have proved useful at Hull include:
  - Ruby on Rails Tutorial (Learn Web Development with Rails) - https://www.railstutorial.org/book - This goes through a creating sample application from start to finish using external tech like Git, Heroku, along with going through Test driven approaches to add new features.
  - Codeschool has a free elective course (in addition to Rails for Zombies) which is a little bit more “question and answers” - https://www.codeschool.com/paths/ruby
  - Railsbridge arranges Ruby-based events, but also has resources to get up and running - http://docs.railsbridge.org/docs/
  - Oxford had had success with installing suifa. The main issue had been working with Ruby itself, there had been no issues with Hydra itself.
  - There was some discussion on the use of Ruby Version Manager (RVM). Use of .ruby-version within this had been found to be very useful.
  - York had queries about adding local customisations within Hydra. This still remains dependent on what type of customisation has been made. Changes to core components should be avoided, though this is less of an issue now as they are much more stable. Otherwise, Hydra is designed to be adapted.
  - LSE had encountered gem dependency issues, and had found it often easier to build from scratch and create a new gemfile. The use of the gemfile.lock file had proved useful at Hull, as this provides stability and a fixed point of reference. It was also noted that upgrades to gems since going to Hydra 6 had gone very smoothly.
  - Passenger was advocated as a Ruby web server.

Metadata, content models and sets

Discussion under this heading was, with no surprise, widespread. Core points raised in the discussion included:

- Primary descriptive metadata within Hydra implementations has tended to be MODS.
- Use of other schemas has been explored to help manage other types of metadata. For example, the Avalon media Hydra head is exploring use of PBCore for structural metadata, whilst using parts of PREMIS has also been discussed (though not yet widely implemented) for preservation and event metadata.
- Hull has implemented UKETD_DC within HyHull, automatically generating this from MODS on the fly using a stylesheet (a similar mapping enables plain DC outputs for harvesting). Additional mappings being developed are one from MODS to RIOXX (once this profile has been finalised) for open access compliance and from MODS to RIF-CS (for use with the proposed research data registry work at DCC). These mappings would be shared as they are completed.
- The N8 is developing a metadata profile for research data, which is based on the profile produced by the Research Data @ Essex project at the University of Essex (which informed the Recollect EPrints plugin available).
- Hydra is moving more towards use of RDF. Hence, is this more sustainable? Oxford uses RDF by default and has done for some time. It then transforms this into MODS as required.
  - This highlighted the debate (in which there are no clear answers) as to whether metadata formats should be primarily viewed as database storage formats or simply transfer formats.
  - Oxford are also making good use of the questioning_authority gem, allowing them to apply local vocabularies as well as LCSH. The gem works at one level of hierarchy only at the moment, but its functionality is still valuable.
  - Questions arose about whether it may be possible to use multiple, overlapping vocabularies, and whether and how it may be possible to hold people authority objects with different IDs.
• Rights metadata was also discussed. Oxford like the Admin Policy Object (APO) approach - where objects link to separate objects declaring rights statements rather than hold their own information: this allows much richer rights information to be applied, but does have the disadvantage that the rights information is not with the object at all times. They are currently looking at WAC (rights vocabulary within RDF). The issue of rights in RDF was also a topic of discussion at Hydra Connect #2.

• The use of structural sets and display sets at Hull was described. Structural sets provide internal structure, allowing behind the scenes management and also a route to assign rights to groups of objects (which inherit the rights assigned at the set level). Display sets are more ad hoc, and are intended to provide flexible groupings of objects that can be displayed as collections, facilitating navigation. Structural sets and display sets do not necessarily have a one-to-one mapping in terms of what is in them.

• The ability to control rights through sets relates to the need to better manage embargoes to meet the terms of the HEFCE REF open access policy. The work carried out within Worthwhile for Case Western Reserve University's Hydra head may have some use here.

Authentication and authorisation

The range of options for authentication and authorisation is broad, and a number of options were discussed.

• HyHull makes use of CAS, and applies a CAS module for Devise. (Note: since the meeting ICTD at Hull has announced the intention to move from CAS to Ping Federate, a commercial product.)

• Devise is flexible, and can also be used to support OAuth and OpenID.

• There was some debate on the degree to which it is possible to enable datastream-level access control (as can be achieved through York's current implementation of XACML. This is primarily enabled through separate objects within Hydra (using an appropriate hierarchy), though some Hydra heads may have tackled this further (e.g., the DIL head at Northwestern, which manages images in different resolutions). Datastream restrictions can be defined, but additional code would be required to implement these in rights metadata.

Data upload and delivery

A particular size-related issue had arisen when loading data into Hydra heads. Other related issues followed from this.

• Is there a 2GB limit in Rubydora? This was a Fedora 3 issue. Fedora 4 is better at big objects, and doesn't use Rubydora

• Large objects need to be ingested via the Fedora route, where a choice can be made as to whether to reference the content and treat it as Fedora managed content. Use of this route can take place behind a common ingest interface, so end-users would not know a different mechanism was being used.

• WGBH and Avalon, working with media files, have extensive experience of ingesting large objects.

• Discussion on preservation focused on how Archivematica might be used with Hydra. This will be explored separately.

• The browse_everything gem can be used to ingest from external sources. Lancaster would like to make use of it to ingest from Box.com (as will Hull in due course once this service is set up).

• Batch ingest - there are a number of scenarios here, primarily one-to-many (many objects, one metadata record attached to each) and many-to-many (many objects, each with their own metadata). LSE do the latter via Fedora. Lancaster use a Python script they have developed and Selenium Webdriver to carry out batch ingest into HyHull.

• What should be done with existing Fedora objects in converting them to Hydra? Should conversion wait for Fedora 4 (to avoid multiple conversions)? This is an open question. Current guidance from DuraSpace is to wait for Fedora 4.1 towards the end of 2015 to migrate from existing Fedora implementations. New implementations are recommended to go with Fedora 4.0 (now available). The choice relates to local timetables. Fedora 4 will support XML (so a conversion to RDF is not mandatory). What Fedora 3.x support will there be? This is an open question for DuraSpace and the community to address.

• Page turners - A RIIF gem had been developed and could be the basis for a Hydra page-turner. (Note, since the meeting discussion on the lists has highlighted the benefit of simply making objects IIIF compatible and then exploiting one of the emerging IIIF object viewers.)

• Geoblacklight was described - more on this will emerge as work progresses at Stanford.

• PDFviewer, a Mozilla pdf.js viewer, can be used to display PDF files, or the browser defaults can be used.

The meeting at this point split into two groups. Notes below are from the manager meeting on the future development of the Hydra UK group.

Hydra UK

A range of topics were discussed.

• Membership
  • Those sites using Hydra in Ireland will be invited to attend (as has been the case for the Fedora UK&I User Group).

• Advocacy
  • Two institutions who attended the Hydra Europe symposium in April 2014 are still interested in Hydra, but are addressing this over time in relation to other developments.
  • A welcome message will be sent to new adopters of Hydra across Europe (as was done for Ghent just prior to this meeting).
  • What conferences might we able to talk about Hydra at?

• Organisation
  • A hydra-uk google group will be set up to facilitate practical arrangements. Queries on software or other community matters should be directed to hydra-tech or hydra-community so they are shared widely.

• Funding
  • The Jisc Research at Risk programme may offer some opportunities to develop Hydra within the UK. Ideas are welcome.