

2015-09-10 - Fedora Tech Meeting

Time/Place

This meeting is a hybrid teleconference and IRC chat. Anyone is welcome to join...here's the info:

- Time: 11:00am Eastern Daylight Time US (UTC-4)
- U.S.A/Canada toll free: 866-740-1260, participant code: 2257295
- International toll free: <http://www.readytalk.com/intl>
 - Use the above link and input 2257295 and the country you are calling from to get your country's toll-free dial-in number
 - Once on the call, enter participant code 2257295
- IRC:
 - [Join the #fcrepo chat room via Freenode Web IRC](#) (enter a unique nick)
 - Or point your IRC client to #fcrepo on irc.freenode.net

Attendees

- [Unknown User \(acoburn\)](#)
- [Unknown User \(escowles@ucsd.edu\)](#)
- [Bethany Seeger](#)
- [Andrew Woods](#)
- [Jared Whiklo](#)
- [Yinlin Chen](#)
- [Aaron Birkland](#) ★
- [Kevin S. Clarke](#)
- [James R. Griffin III](#)
- [David Wilcox](#)
- [Unknown User \(daniel-dgi\)](#)
- [Nick Ruest](#)

Agenda

1. Update on API Extension Architecture work;
 - a. More [use cases](#) added
 - b. Lots of discussion around [validation](#) use case
2. WebAC - 0.1.0
3. Next release?
4. Recent PCDM updates
 - a. <https://github.com/duraspace/pcdm/issues/23>
 - b. <https://github.com/duraspace/pcdm/pull/2>
 - i. <https://github.com/duraspace/pcdm/pull/27>
 - c. <https://github.com/duraspace/pcdm/issues/21>
5. ...
6. Please squash a bug!

type	key	summary	assignee	reporter	priority	status	resolution	created	updated	due
------	-----	---------	----------	----------	----------	--------	------------	---------	---------	-----

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

7. Tickets resolved this week:

key	summary	type	created	updated	due	assignee	reporter	priority	status	resolution
-----	---------	------	---------	---------	-----	----------	----------	----------	--------	------------

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

8. Tickets created this week:

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

Minutes

- Update on [API Extension Architecture](#) progress so far
 - New [use cases](#) are still being added, more are welcome
 - Focus has been on understanding use cases and role of architecture vs extension, this has proven to be a valuable exercise.
 - [validation](#) use case has evolved into a 'deep dive' in sketching out an extension and how it might interact with the framework
- Question from Danny (Islandora): So what is the API Extension Architecture anyway?
 - Public URIs, expose web resources in similar pattern to fcr:fixity; route requests to service/code that implements it
 - Use cases have highlighted 'filtering' pattern. Intercept and filter incoming and outgoing requests, e.g. add link headers as in [signposting](#)
- Potential Hydra and Islandora use cases
 - Danny is in the middle right now of translating nasty URLs into Friendly URLs. Is this a potential use case?
 - Yes, add it to the [use cases](#) page. The more use cases, the better the design may fit needs of community
 - Don't worry about 'is this an appropriate use case'. That question has to be asked of every use case on the page
 - Hydra has similar identifier/URI issues.
 - Whatever the identifier scheme, mapping between application objects and repository resources is not going to be 1:1, so it's good to get a handle on that.
 - Potentially Use API Extension for Identifier lookup and URI re-writing.
- Is there anything that can/should be pulled out of core and into an API extension
 - Transform, OAI, SWORD maybe
 - Still a little early to discuss this, API extension effort is not at the design stage yet; would want a working product before considering its value
 - Current extension mechanism is a little awkward, if nothing else the common interest in OSGI among various efforts could bring benefit to pluggability of existing extensions.
 - Existing patterns for async camel toolbox usage shouldn't change
- WebAC sprint ended last week
 - Quite pleased everything pulled together in the end
 - There is an implementation out there for review, please take a look
- Does it make sense to use WebAC as a guinea pig for its own independent versioning scheme?
 - Currently, everything uses Fedora version, rolled up into release
 - Doesn't make sense to bump version of a module that hasn't changed
 - WebAC will be the snowball that starts the avalanche of independently versioned modules?
 - Need for versioning matrix?
- Can XACML usage be replaced with WebAC?
 - WebAC much simpler to write and understand
 - What sort of use cases are supported by XACML but not WebAC?
 - In Islandora, XACML mostly mirrors Drupal permissions, which are simple. Looks like WebAC would probably work, and be much easier than XACML
 - Esmé looking into making a hydra-jetty branch that uses WebAC. Will make evaluation/testing easier in Hydra context
- Can we improve messaging to get out WebAC word?
 - Cross-posting?
 - Friendly reminders?
 - Maybe people from Hydra/Islandora can monitor fcrepo-community, and relay+translate messages to Hydra or Islandora communities.
 - Make the fedora messages make sense to your specific communities
- Next WebAC sprint is coming
 - There's not a lot of time to pull together feedback/goals/requirements for sprint 2
 - Given some ambiguities in the spec, it would be interesting to see how our implementation fits with people's use cases
 - Move sprint 2 backwards?
 - It would be better to have more clearly defined goals that result from the stakeholders and community having time to review it and try it out
 - Problem is scheduling time between institutions
 - [Unknown User \(acoburn\)](#), [Jared Whiklo](#) are probably OK with moving, not sure about UMD folks
 - [Andrew Woods](#) to contact UMD and check scheduling flexibility. Will contact stakeholders about their opinion on having enough time to review.
- Is it worth pushing for a release before HydraConnect? Release right after?
 - Two votes for 'no rush'
 - A big TODO would be compatibility matrix
- Is this the time to pull the plug for versioning modules with the release?
 - A few votes for 'pull the plug'
 - Has the semantic versioning question been settled?
 - Changing it now would be too confusing

- Once Fedora sorts out structural changes (cleaning out core, specifying APIs supported by core), it makes sense to move to 5 and adopt semantic versioning