

2016-02-18 - 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)
- Dial-in Number: (712) 775-7035
 - Participant Code: 479307#
 - International numbers: [Conference Call Information](#)
 - Web Access: <https://www.freeconferencecallhd.com/wp-content/themes/responsive/flashphone/flash-phone.php>
- 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

- [Bethany Seeger](#)
- [A. Soroka](#)
- [Yinlin Chen](#)
- [Esme Cowles](#)
- [Jared Whiklo](#)
- [Andy Wagner](#)
- [David Wilcox](#)
- [James R. Griffin III](#) ★
- [Andrew Woods](#)
- [tamsin johnson](#)
- [Benjamin Armintor](#)
- [Namibia Bahulekar](#)
- [Allen Flynn](#)
- [Katherine Lynch](#)

Agenda

1. [\[redacted\]](#) ms for RdfStream interface configuration. (Pull Request) Unable to locate Jira server for this macro. It may be due to Application Link
2. [\[redacted\]](#) nd organization
3. Removing /fcr:nodetypes endpoint (Pull Request) Unable to locate Jira server for this macro. It may be due to Application Link configuration.
4. Moving away from LevelDB to... MySQL? Postgres? other?
 - a. Is there some context for this? Problems with LevelDB? Are there tickets documenting why we no longer like it?
5. Fedora Specification updates
 - a. [Messaging SPI](#)
 - b. [Atomic Batch Operations](#) - name? BatchOps?
 - c. [CRUD](#)
 - d. [Resource Versioning](#) (A. Soroka will start work on this at the top of the coming week)
 - e. [Binary Fixity Checking](#)
 - f. [Authorization](#)
6. Recent test results
 - a. [Unknown User \(bbpennel\): PCDM](#)
 - b. [Esmé Cowles: MySQL vs. LevelDB](#)
7. ...
8. Status of "in-flight" tickets

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.

Ticket Summaries

1. Please squash a bug!

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.

2. 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.

3. Tickets created 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.

Minutes

1. Using Java 8 streams for RdfStream interface

Soroka

- Addressing the pull request (PR) as it stands...should we merge or replace it with something more ideal?
- No one is suggesting not merging the PR
- Does Coburn have time to finesse it?

Coburn

- (Providing the background for the issue)
- Current implementation of Fedora Commons (fcrepo) extensively uses Guava iterators
- Allows one to do lazy processing...functional idioms for writing codes
- Java 8 allows one to use core streams library, deprecating the need for Guava
- Best to use core packages rather than rely upon Guava
- Addressing the #getTriples function:
 - The function returns an iterator (must, hence, be changed to a stream, as specified above)
 - However, accepted by the function are name of implemented Classes
 - These correspond to approximately 8 sets of triples which could be requested (e. g. membership, versioning, fixity...)
 - As a consequence, this introduces a hard dependency on ModeShape implementation of fcrepo
 - In turn, this precludes any further abstraction, inhibiting the implementation of non-ModeShape fcrepo
- Hence, this PR introduces an *enum*
- Covers all of the cases currently in case with the ModeShape implementation and prefer headers in REST API
- As an *enum*, it doesn't allow for any extension of these values
- An idea proposed by Soroka is, rather than using the enum, use an interface or set of interfaces which can be passed in

Soroka

- Is there time to find a more ideal solution to *enum* now?
- Or, is it viable to avoid merging now (merging the PR without an alternative to the *enum* solution requires that work be thrown away later)

Coburn:

- The PR is quite large
- 1/10th of the entire code base; Rebasing it is complete hell
- To keep iterating on this in order to add additional functionality while merging other PR's into the code base presents other problems
- Specifically, the task of managing the PR becomes increasingly difficult
- Definitely should remove *enum*, but advocates merging the PR as it is
- Then, replace *enum* immediately after

Soroka:

- Why would this approach prevent us from not using the enum at all and taking the time now to refactor the PR?

Coburn:

- He wouldn't have the time to refactor the PR with the preferred solution

Soroka:

- We all agree that this must be redesigned
- Not suggesting that this is a blocker
- Not volunteering to refactor the *enum*

Woods:

- Why bundle in the change for getTriples into the PR for this ticket?

Coburn:

- In thoroughly addressing the ticket...it became apparent that all of the implementing classes in ModeShape impl. would need to be rewritten
- Ideally, these would be separate pull requests, but they aren't

Woods:

- Two different things bundles into a single PR for this ticket
 - Migrating way from homegrown iterator (addressed using Java 8 core)
 - Mechanism for identifying the triples desired for underlying repository
- First goal is accomplished

Armintor:

- Preferred that this not be released with the *enum*
- But, it is harder to merge later on, and best to get everyone on the same base

Woods:

- enum Might still be a blocker for the next release

Soroka:

- Agrees, merge and consider the *enum* issue to be a blocker

Woods:

- Consistent amount of changes into the code base
- Concerted effort to avoid introducing breaking changes for HTTP API (and other API) levels
- Aiming for a 4.5.1 release
- Could be value in this...relates to third agenda item
- Removing an endpoint (deprecation, breaking change)
- Good to get a "point" release out which alerts the community to this
- Should enum remain a blocker for a point release?
- Less than a month required for addressing the *enum* question?

Soroka:

- This is not a part of the public API
- Hence, can wait to resolve enum issue until this affects a component of the public API
- No need to block a "point" release
- Just desires to set a time limit to rectify this problem
- People will want to implement the API
- This will still block these efforts

Armintor:

- Doesn't see a reason to block a "point" release for this issue

Woods:

- Will write the ticket
- Refactor the enum approach
- Make it a high priority, try to address this immediately and jointly
- Enable alternate implementations to then be written

2. Package Naming and organization

Coburn

- Somewhat related to agenda item #1
- Number of new classes and interfaces in kernel API (4)
- 3 are in the base level org.fcrepo.kernel.api
- 1 is an implementation api.rdf
- Uncertain of a good location for these...do some constitute an implementation?
- What are these packages inside and outside of the kernel API?
- Generally speaking, you should avoid cyclical dependencies between packages
- e. g. The api.exception package references code in the api package, which itself references code in the api.exception package
- Usually not the best practice
- Raises the larger question of...what are these packages?
- RDF package has one class within it
- More inclined to have have fewer packages
- Other approaches prefer more specific package names
- Circular dependencies are also really bad in the ModeShape module

Soroka:

- ModeShape is a monolith
- Discussions have been had regarding similar

Coburn:

- Few or no circular dependencies are found within the HTTP modules

Woods:

- Intensive assessment of modules and the packages within each module

Coburn:

- Proposes a Google Doc for this discussion

Soroka:

- Sonar will detect loops, doesn't indicate how best to restructure the packages

3. Removing the FCR nodetypes

Coburn:

- fcr:nodetypes endpoint is undiscoverable by an LDP client
- The endpoint describes all of the RDF classes, includes all of the JCR hierarchies
- Most of the time, repository resources shouldn't need to know anything about the JCR hierarchies
- No strong argument to retain this endpoint
- PR to remove it

Woods:

- Few likely know about this endpoint, fewer probably use it
- Yet, this would still constitute a breaking change
- Should alert community
- Add a warning header to this endpoint indicating that this is to be removed
- Perhaps adopt a policy to ensure that these deprecation warnings are issued
- Further, specify a term of time

Esme:

- Best to have deprecation
- Ideally, header should have the time frame
- Not just a generic warning, but specify a date for the removal

Coburn:

- PR which was merged adding the warning header doesn't specify a date or particular release

Woods:

- Prefers to have a date, but would easier

Esme:

- Concrete predictions require that this be addressed within the release plan

Coburn:

- E-mailed the list 2 weeks ago
- <https://groups.google.com/forum/#!msg/fedora-tech/1Gfsln0Ugug/lcoIP3DBCQAJ>
- 4.5.1 deprecation release should issue the warning
- 4.6.0 deprecation release should no longer feature the endpoint

- Note that this information is not in the header for the current PR

Esme:

- Most won't see the deprecation release until there is a "point" release and they upgrade
- Suggests that there should be a deprecation warning, released in a "point" release
- Then, others have the opportunity to take some action, introducing the breaking change in the succeeding "point release"

Woods:

- (Queries the community for period of time)

Coburn:

- Perhaps distinguish between core features people are using and those not likely being used by many
- For features being actively used, 6 - 12 months

Johnson:

- Several months might be a good guideline
- But, far less time might be fine for core features which aren't used
- Key is to effectively use version numbers
- Major releases should be well organized and with the proper notes

Woods:

- Before making a breaking change, identify deprecation within a header message
- Ideally, target release where the deprecation
- There will be cases where this might not be possible (sticks around for a number of releases)

Esme:

- Typically wait 2 months between releases for certain architectural changes already
- Good practice to specify that this is removed in 4.6.0...being that specific would be the most helpful
- Avoid specifying a date and missing this deadline makes it less predictable

Woods:

- The ticket for this already exists, any should be in the position to add a PR
- <https://jira.duraspace.org/browse/FCREPO-1892>

4. Moving from LevelDB (ModeShape-specific storage for objects)

Woods:

- By default, use LevelDB
- Can now use MySQL in code base
- Esme's PR offers integration for PostgreSQL
- Corruption issues for LevelDB have been identified in at least one e-mail thread
- Bulk ingest with an "out of memory error"
- Tomcat hangs, must be restarted
- Scripts from Muhammad from U. Maryland
- Works for some in identifying corruption in the LevelDB

Esme:

- Part of the ModeShape move away from Infinispan seems to be to move towards a RDBMS
- Try to align ourselves now by preparing to move towards an approach which leverages these

Woods:

- When ModeShape 5 is released, JDBC supports PostgreSQL
- Migration would still be required
- What is required in moving from LevelDB to MySQL or PostgreSQL
- Fedora 4 offers a *backup & restore/JCR export* feature
 - Not ideal, won't show up in Fedora specification, but still there
- Yinlin successfully tested a LevelDB to MySQL migration
- Esme started performance testing (against LevelDB and MySQL)
- Might we change or suggest that LevelDB be avoided?
- How hard to push on JDBC backing for Fedora 4?
- Should we wait until the ModeShape 5 release?

Esme:

- Looks like ModeShape 5 is going to be released within 1-2 months
- Not a long time to wait...confusing to offer this support
- Then, introduce the new migration
- But, there are parties within corrupted repositories right now which much have this addressed

Woods:

- Also, parties looking to just start might be best working without LevelDB

Esme:

- Advocates using a "point" release in which MySQL and PostgreSQL (or both) are supported
- ModeShape 5 would then trigger a major release for Fedora

Woods:

- Proposes that parties not be encourage to migrate prematurely (given the upcoming release of ModeShape 5)

Esme:

- Agreed, this should start the conversation, but avoid forcing anyone to migrate before

5. Fedora Specification Updates

Woods:

- 6 documents
- Sections of Fedora specification
- All being drafted (and in various states)
- Call from involved persons to produce a summary

Messaging API

Coburn

- Finished drafting
- Invites comments

Atomic Batch Operations

Whiklo

- https://docs.google.com/document/d/1lj4lFomcOJuOiWptZPyhP_wBRtxbqBP3_Rdw1eKmCIM/edit
- As this was essentially an initial attempt, would appreciate someone to hack it apart
- Still would prefer to iterate and refine the document

Authorization

Flynn:

- Want to watch how others are starting the process
- Intend to have something substantial for the next meeting

Bahulekar:

- Questions relating to WebACL specification compliance

Woods:

- WebACL spec. leaves some room for interpretation
- Best to tighten up the ambiguities which are there

CRUD

Johnson:

- Complete from the perspective of the immediate adjustments to LDP
- Open question about how to handle PUT creation
- There was pretty heavy discussion on this
- Following the conclusion to this discussion, these points must be addressed
 - Any possible content restrictions
 - Which triples are allowed

- Any formal specification of the prefer headers (or other fcrepo specific headers)
- None of these are featured

Armintor

- So far, writing into 2 sections
- First section addressed specifications in LDP which need to be refined
- Second section addressed unspecified in the LDP which need to be specified outright
- Invite feedback in order to gauge the navigability of the document
- Use sections to align with the LDP spec. section numbers
- Need to resolve comments on the document before this can be addressed