

2019-08-29 - Fedora Tech Meeting

Time/Place

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

- Time: 11:00am Eastern Daylight Time US (UTC-4)
- Audio/Video Conference Link: <https://lyrasis.zoom.us/my/fedora>
 - Dial-in:
 - +1 408 638 0968
 - +1 646 876 9923
 - +1 669 900 6833
 - Meeting ID:
 - 812 835 3771
- Join fedora-project.slack.com on the "tech" channel

Attendees

Part 1:

1. [Danny Bernstein](#)
2. [Peter Winckles](#) ★
3. [Andrew Woods](#) (out)
4. [David Wilcox](#)
5. [Peter Eichman](#)
6. [Joshua Westgard](#) (out)
7. [Jared Whiklo](#)
8. [Bethany Seeger](#)
9. [Youn Noh](#)
10. [Thomas Bernhart](#) (might have to leave early)
11. [Ben Cail](#)
12. [Rosie Le Faive](#)
13. [Daniel Lamb](#)
14. [Aaron Birkland](#)

Part 2:

Agenda

1. Announcements
2. [Sprint Planning](#)
 - a. [6.0 Architecture Review](#)
 - b. Coming to consensus on:
 - i. Definition of "rebuild"
create all necessary indexes, caches, or other ancillary data structures derived from *persisted content* (OCFL + *unversioned content*)
 - c. unversioned content
 - i. What use cases must we support?
 1. Automatic versioning of staged content after X minutes/hours/days/weeks?
 2. Permanently unversioned content
 3. ?
 - ii. Issues to gain consensus on:
 1. Are we agreed that unversioned content will live outside OCFL Storage Root?
 2. ?
 - d. Multi-object transaction implementation approach?
 - e. [Transaction Sidecar Spec](#) Update
 - f. Major Areas of Work
 - i. Design/Development
 1. Interface Definition
 - a. Persistence API
 - b. ?
 2. OCFL Client Development
 - a. OCFL Java API
 - b. OCFL Java Client Implementation
 3. Transactions
 - ii. Documentation
 - iii. Testing
 - iv. Import/Export/Migration
 - v. ?
 - g. Sprint Goals
3. Status on organizing a Fedora documentation review
4. Introduction to running Fedora with Valkyrie
5. Applying a digital preservation framework (e.g. [NDSA Levels of Digital Preservation](#)) to Fedora 6

- 6. Update on [Fedora 6 Pilots](#)
- 7. Status
 - a. API Test Suite PRs
 - i. <https://github.com/fcrepo/Fedora-API-Test-Suite/pulls>
 - b. Minimal 4 5 migration needs testing and code review:
 - i. <https://github.com/fcrepo4-exts/fcrepo-upgrade-utils/pull/17>
- 8. Your topic here...

Tickets

- 1. In Review

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.										

- 2. 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.										

- 3. 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.										

- 4. 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.										

Notes

Part I

- 1. Announcements:
 - a. Good, well attended meeting in Switzerland. Interest in Fedora 6.
- 2. Value prop of using Valkyrie with/without Fedora
 - a. Fedora 4 & 5 about standardization
 - i. Some performance issues
 - b. Fedora 6 adding more preservation minded features
 - i. Transparent, human-readable filesystem
 - ii. Rebuildability from FS

- c. Why better to use Fedora over Preservica?
 - i. Preservica not active storage
- d. Fedora offers standard middleware integrations: notifications, robust api
- 3. Definition of Fedora rebuild
 - a. Ability to recreate everything needed to run Fedora based on the files on disk (Daniel's exact words should be added here)
- 4. Support unversioned content?
 - a. Some use Fedora to author content over a period of time. They want to version when the content is ready and not have a version of every change.
 - i. Clean version set
 - ii. Reduce bloat on disk
 - b. Fedora currently offers the ability to mutate objects and create fixed versions. In OCFL, everything is immutable. How to reconcile this, and maintain Fedora functionality.
 - i. Store everything in OCFL and make version logical structures on top of this
 - ii. Store only immutable content in OCFL
 - 1. Mutable content stored within OCFL's 'deposit' directory
 - 2. Fedora maintains mutable content
 - iii. Automatically periodically "flush" staged changes to OCFL
 - 1. OCFL versions would not be meaningful – requires logical versions
 - c. There's a possibility that an object could never have a version. Is this okay?
 - d. Auto-versioning should be toggle-able
 - e. Islandora controls versioning – auto-versioning would be off by default
 - f. Where does unversioned content live?
 - i. OCFL 'deposit' directory is intended for short-lived version creation and not long-term storage
 - ii. OCFL 'deposit' directory space is out of spec and is up to the individual library to use it as it will
 - g. Currently, you can delete version markers, but this is not possible if you store version markers in OCFL. Perhaps store version markers outside of OCFL?
 - i. All changes stored in OCFL
 - ii. Fedora managed version file that maps OCFL versions to logical versions. This mapping is outside of the OCFL object.
 - h. Pros about versioning every change in OCFL
 - i. Easier to implement
 - ii. Easier to reason about
 - iii. Rebuildability is easier
 - i. Cons about versioning every change in OCFL
 - i. Binary bloat
 - ii. Inventory bloat
 - j. OCFL versions are not presented in the API only Fedora versions are

Part II

- 1. What is "auto-versioning"?
 - a. On close of Fedora transaction an OCFL version is created – this happens regardless of whether or not auto-versioning is enabled
 - b. Auto-versioning on: Fedora tag automatically created for every OCFL version
 - c. Auto-versioning off: No Fedora tag created
 - d. Absent a tag file that describes Fedora versions, OCFL versions are exposed
 - e. Counter-prop: No auto-versioning. All Fedora versions must be manual.
 - f. Cannot support tagging old OCFL versions with Fedora versions
 - g. Concerns about exposing OCFL versions through Fedora APIs – don't want Fedora to be tied to OCFL
 - h. How does this work when importing from old versions of Fedora?
 - i. Fedora 4 & 5: can the versions be replayed?
 - ii. Fedora 4 5: Export 4, run transform, Import 5
 - iii. Fedora 5 6: Export 5, run transform that produces OCFL, point Fedora 6 to OCFL
 - iv. Fedora 3 6: Conversion of Fedora 3 files on disk to OCFL, point Fedora 6 to OCFL
- 2. Transactions
 - a. Transactions at the object level are do-able, but across multiple objects is trickier. May not need to implement multi object transactions.
 - b. Bound to an archival group. Don't allow multi object transactions across different archival groups.
 - c. Transaction endpoint exposed for every OCFL object
 - d. Cannot make changes to objects out of scope of original transaction.
 - e. How do you open a transaction for a new OCFL object?
 - i. Create an empty object, open a transaction, update object, close transaction – would not remove the empty object on rollback
 - f. Why limit transactions to just one object?
 - i. Multi-object OCFL transactions are hard to implement
 - g. Something in the import/export group for mapping to archival groups

Actions

- [Aaron Birkland](#) to look explore notion of OCFL client with database as authoritative metadata source + asynchronous writing of the inventory.json file
- [Peter Eichman](#) and maybe [Ben Pennell](#) to make recommendations re transaction side car specification.
- [Andrew Woods](#) will look into java 11 transition