

2020-07-30 DSpace 7 Working Group Meeting

- [Date](#)
- [Agenda](#)
- [Attendees](#)
- [7.0 Release Goals](#)
- [Current Work](#)
 - [Project Board](#)
 - [Delayed / Needs Discussion](#)
- [Notes](#)

Date

30 Jul 2020 from [14:00-15:00 UTC](#)

Location: <https://lyrasis.zoom.us/my/dspace> (Meeting ID: 502 527 3040).

- More connection options available at [DSpace Meeting Room](#)

Beta 4 Sprint : Ongoing



- Beta 4 active development
- [DSpace 7 Project Boards](#)
- View PRs assigned to you for review/testing: <https://github.com/pulls/review-requested>

Agenda

- **(BEFORE MEETING IN #dev-sprint)** Developer Stand Up - Developers give brief updates on their effort (or their team's effort).
 - Update/see "Current Work" section below based on your status. *Please feel free to update prior to meeting.*
 - Please highlight any new work (needing reviews/testing), any blockers (for you), and any discussion topics you may have.
- (30 mins) General Discussion Topics
 1. **(20 mins) Submission form conceptual design:** Originally designed to only touch WorkspaceItem and WorkflowItem, the recent changes in [PR#541](#) now touch the Item object (and cause issues for other PRs). [Giuseppe Digilio \(4Science\)](#) added a ticket to describe the problem & provide a proposed solution: <https://github.com/DSpace/dspace-angular/issues/818>
 - a. Do we agree that the submission forms should ONLY touch WorkspaceItem and WorkflowItem? (NOTE: This is similar to behaviors in DSpace v6, where submission forms never interacted with the Item directly, but always worked through these "wrapper" objects)
 - b. Any additional ideas/concepts we need to clarify in Submission process design? (This is an opportunity to document our assumptions/design for this Submission process, so that we can work from a common idea.)
 2. **(10 mins) Security issues in Processes REST endpoint:** Some CLI flags should never be usable via the REST API (e.g. `--eperson`). We should find a way to either disable these flags automatically via the REST API, or perhaps a way to configure (per script) which flags are "CLI-only".
 - a. This same issue has already appeared in multiple scripts (see links below) and for multiple CLI flags. Unfortunately, some CLI flags become a **security issue** when available on REST, but they are **not** a security issue via CLI.
 - b. See this initial issue around `metadata-import`: <https://github.com/DSpace/DSpace/issues/2822>
 - c. The Curation Tasks PR has the same security issue, as it also allows the `--eperson` flag: <https://github.com/DSpace/DSpace/pull/2820>
 - d. Another problematic flag is the ability to specify an output file location, e.g. in `metadata-export`: <https://github.com/DSpace/DSpace/issues/2821>
 3. Additional topics welcome. Please add by Tues, July 28th.
- (30 mins) Planning for next week
 - Review of our [Beta 4 Project Board](#)

Attendees

- [Art Lowel \(Atmire\)](#)
- [Andrea Bollini \(4Science\)](#) - On holiday July 27-Aug 7
- [Tim Donohue](#) - Upcoming holiday Aug 10-18
- [Lieven Droogmans](#) - Upcoming holiday Aug 3 - Aug 7
- [Giuseppe Digilio \(4Science\)](#) - Upcoming holiday July 31-Aug 21
- [Ben Bosman](#)
- [Paulo Graça](#) - out between Aug 15-31
- [Heather Greer Klein](#)

7.0 Release Goals

These resources define the prioritization and general schedule we are working towards

- [DSpace 7 Release Goals](#) : overview of goals/timelines & beta release process
- [DSpace 7 Release Plan spreadsheet](#): our planning spreadsheet which details which features are scheduled for each Beta release.

Current Work

Project Board

DSPACE 7.0 Beta 4 Project Board: <https://github.com/orgs/DSpace/projects/3>

To quickly find PRs assigned to you for review, visit <https://github.com/pulls/review-requested> (This is also available in the GitHub header under "Pull Requests Review Requests")

Delayed / Needs Discussion

1. Finalize / approve the initial list of all authorization features which we should implement for the [/api/authz/features REST endpoint](#). This list of features should be limited to *only features which are required to enable/disable User Interface functionality*. (In other words, we can always add more features in the future. We just need to approve the list necessary for 7.0)
 - a. Review current spreadsheet (from [Andrea Bollini \(4Science\)](#)) : https://docs.google.com/spreadsheets/d/1182LcD_WqlZRbUGWpLtBw0aOMR9jhbOVB7GZqtTpR9A/edit?usp=sharing
 - i. [Art Lowel \(Atmire\)](#) : I don't see any immediate issues with the current set of features, but I would prefer a consistent naming scheme. I'd use canDoSomething for everything
 - ii. [Tim Donohue](#) added possible renames of these features based on Art's idea (see cell comments in spreadsheet). I like the "can[DoSomething]" naming scheme as well.
2. Initial Performance Testing from Chris. Needs revisiting / retesting prior to 7.0.
 - a. <https://cwilper.github.io/dspace-perftest/>
 - b. These performance tests were run *prior* to the work on "projections" (to limit the data returned by the REST API). Therefore, it is likely performance is much improved, but needs verification testing.
3. (REST Contract) Edit Homepage News: <https://github.com/DSpace/Rest7Contract/pull/45>
 - a. Delayed. General agreement (in meeting on March 21, 2019) that storing HTML in metadata fields is not really ideal behavior. Metadata (from a librarian standpoint) tends to be free of format-related markup (as that allows for easier sharing, understanding of metadata. Currently Community & Collection homepage information is HTML-based and *is stored in metadata* that is appropriate for a minor subset of information (like the title) but it is better to move large/rich text to bitstreams.
 - b. Proposal here is to consider storing HTML-based markup (for Site, Community & Collection homepages) in Bitstream(s) associated with the object in question. May allow for more CMS-lite behavior in the future
 - c. Timeline for this is uncertain. Possibly in 7 or 8. May depend on how/whether it can be scoped.

Notes

- (20 mins) Submission form conceptual design
 - **ALL agreed that the Submission process should ONLY use WorkspaceItem and/or WorkflowItem.** No features are allowed to bypass those objects and use the Item object (or endpoint).
 - WorkspaceItem & WorkflowItem are changes in DSpace 7 as they are now more than a "wrapper" object.
 - They also help us to enforce the "rules" defined in the submission process. For example, if we allowed bypassing the WorkspaceItem, then Submitters could edit any metadata already fixed in [redacted] Unable to locate Jira server for this macro. It may be due to Application Link configuration.
 - ALL agree [redacted] proposal to fix current problems sounds reasonable: <https://github.com/DSpace/dspace-angular/issues/810>
 - Effort this will take is the only area of disagreement.
 - Andrea feels the necessary REST fixes are minimal overall. Maybe just a day or two?
 - Art feels the necessary Angular fixes require a major overall. Mentioned a month of effort
 - NEXT WEEK touching base again
 - Giuseppe has a few other ideas of a possible "quick fix" to minimally get "main" branch working again. He will try these out today & create a PR.
 - Atmire team can look closer at Giuseppe's suggestions & verify their estimates around effort on Angular (and REST) side.
 - **If there is no clear solution by next week**, Tim will *revert* the problematic PR (<https://github.com/DSpace/dspace-angular/pull/541>) and reschedule it for beta5 for additional work/attention to better align to the (agreed to) Submission process design. This is a worst case scenario however, as we'd much rather resolve the issues on "main" branch and not have to (temporarily) remove this work.
- (10 mins) Security issues in Processes REST endpoint
 - **Atmire team agrees with Tim's assessment that** "Some CLI flags should never be usable via the REST API (e.g. `--eperson`). We should find a way to either disable these flags automatically via the REST API, or perhaps a way to configure (per script) which flags are 'CLI-only'.
 - Currently both the `--eperson` (in <https://github.com/DSpace/DSpace/issues/2822> and <https://github.com/DSpace/DSpace/pull/2820>) and `--file` (in <https://github.com/DSpace/DSpace/issues/2821>) flags seem to be the first ones that fit into this issue.
 - [Kevin Van de Velde \(Atmire\)](#) is currently looking at a solution & will report back to us on a possible fix.