

2022-02-24 DSpace 7 Working Group Meeting

- [Date](#)
- [Agenda](#)
- [Attendees](#)
- [Current Work](#)
 - [Project Board](#)
 - [New Feature development process for 7.3](#)
 - [Issue Triage process for 7.3](#)
 - [On Hold Topics](#)
- [Notes](#)


Date

24 Feb 2022 from [15:00-16:00 UTC](#)

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

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

7.3 Release Plan

 Release Schedule (*TENTATIVE, may need revising based on OR2022 plans*):

- **Thursday, Apr 28 (PR Creation Deadline):** All new feature (or larger) PRs should be created by this date. (Smaller bug fixes are welcome anytime)
- **Thursday, May 19 (Review/Test Deadline):** All code reviewers or testers should submit their feedback by this date. Code reviews must be constructive in nature, with resolution suggestions. Any code reviews submitted AFTER this date will be considered *non-blocking* reviews. This means feedback received after Jan 20 is *optional* to address (unless the team or PR developer decides it is required).
- **Friday, May 27 (PR Merge Deadline):** All new feature PRs should be merged by this date. (Bug fixes can still get in, as long as they are small or important)
- **Week of May 30:** Internal / Early release goal. If possible, we'd like to release 7.3 in late May or first week of June.
- **Monday, June 6:** Public Release Deadline. 7.3 *must be* announced/released by this date.

Ongoing/Weekly tasks:

- Tackle/Claim issues on [7.3 board](#) (starting with "high priority")
- Review/test all PRs assigned to you for review/testing: <https://github.com/pulls/review-requested> (*Prioritize reviews of "high priority" PRs first*)

Likely 7.3 major new features (tentative):

1. (TIER 1) (Admin) Ability to [preview Batch Metadata changes during import of CSV](#), similar to 6.x. In 7.x, it is possible to import a CSV from the Admin Toolbar, but you are not shown a preview of pending changes. ([UI ticket #782](#) , [REST ticket #2849](#))
2. (TIER 1) (Workflow) Preview an item during workflow approval to allow for easier accepting/rejecting, similar to 6.x ([UI ticket #772](#))
3. (TIER 1) (Admin) Ability to export metadata (to CSV) from search results, similar to 6.x. ([REST Ticket #3129](#))
4. (TIER 2) (NEW) Advanced ORCID integration port from DSpace-CRIS to DSpace ([Ticket #8157](#))
5. (TIER 2) (NEW) Migrate additional "Live-Import" sources from DSpace-CRIS to DSpace: ([Ticket #3359](#))
6. (TIER 2) (NEW) Versioning support for Entities: ([UI ticket #1312](#), [REST ticket #7937](#))
7. (TIER 2) (General) Support for hierarchical controlled vocabularies in Search interfaces, similar to 6.x ([UI ticket #815](#), [REST ticket #2870](#))
8. (TIER 2) (Submission) SHERPA/RoMEO integration in Submission UI, similar to 6.x ([UI ticket #769](#), [REST ticket #2840](#))
9. (TIER 2) (Admin) Administrative Control Panel (similar to 6.x XMLUI) ([UI ticket #801](#), [REST ticket #2877](#))
10. (TIER 2) (NEW) Signposting support, aligning with recommendations from the COAR Next Generation Repositories Report ([UI ticket #811](#), [REST ticket #2881](#))
11. Upgrading to Angular 12 or 13 ([UI ticket #1432](#)), and UI performance improvements (e.g. [UI ticket #741](#) and [UI ticket #1357](#))

Agenda

- (30 mins) General Discussion Topics
 1. [DSpace 7 at OR2022](#) - What talk(s) are we submitting for OR2022?
 2. Review tentative 7.3 Release Schedule above. This is based on a new 4-month release window, but could conflict with OR2022 plans.
 3. Any [assigned 7.3 work](#) needing early discussion?
 - a. **Additional feedback on Administrative Control Panel:** DEMO of third-party webapp for [Spring Boot Actuators](#). Could we use this webapp as our Administrative Control Panel, or should we build our own UI in the Angular UI?
 - i. Actuator PR: <https://github.com/DSpace/DSpace/pull/8166>
 - b. **Additional feedback on DSpace 7 Versioning with Entities:** https://docs.google.com/document/d/1E7_zUse5ojl_N9hks53GS3NGboXKkxF6ud1lPmcDP8/edit#
 - i. Feedback from last week was the high level design looks good. All agree.
 - ii. Asked [Ben Bosman](#) for a new diagram to outline the use case of two Publications linked via a relationship where Publication A is the review of Publication B. If Publication B is then versioned into B2, it may be important for Publication A to somehow retain an "original" link to Publication B (as that was the version which was reviewed), even if B2 is now the latest version.
 4. (Other Topics?)
- (30 mins) Planning for next week
 - Review the [Backlog Board](#) - Are there any tickets here stuck in the "Triage" column? We'd like to keep this column as small as possible.
 - Review the [7.3 Project Board](#) - Assign tickets to developers & assign PRs to reviewers.

- Paid (by DSpace project) developers must keep in mind priority. If new "high" or "medium" priority tickets come in, developers should move effort off of "low" priority tasks.
- Volunteer developers are allowed to work on tickets regardless of priority, but ideally will review code in priority order.

Attendees

- Art Lowel (Atmire)
- Andrea Bollini (4Science)
- Tim Donohue
- Lieven Droogmans
- Giuseppe Digilio (4Science)
- Ben Bosman
- Paulo Graça
- Mark H. Wood

Current Work

Project Board

DSpace 7.3 Project Board: <https://github.com/orgs/DSpace/projects/16>

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")

New Feature development process for 7.3

- **For brand new UI features**, at a minimum, the UI ticket should contain a description of how the feature will be implemented
 - If the UI feature involves entirely new User Interface interactions or components, *we recommend mockups or links to examples elsewhere on the web*. (If it's useful, you can create a Wiki page and use the [Balsamiq wireframes](#) plugin in our wiki)
 - Feature design should be made publicly known (i.e. in a meeting) to other Developers. **Comments/suggestions must be accepted for TWO WEEKS, or until consensus is achieved (whichever comes first)**. After that, silence is assumed to be consent to move forward with development as designed. (The team may decide to extend this two week deadline on a case by case basis, but only before the two week period has passed. After two weeks, the design will move forward as-is.)
 - This does mean that if a UI feature is later found to have design/usability flaws, those flaws will need to be noted in a bug ticket (to ensure we don't repeat them in other features) and fixed in follow-up work.
- **For brand new REST features (i.e. new endpoints or major changes to endpoints)**, at a minimum we need a REST Contract prior to development.
 - REST Contract should be made publicly known (i.e. in a meeting) to other Developers. **Comments/suggestions must be accepted for TWO WEEKS, or until consensus is achieved (whichever comes first)**. After that, silence is assumed to be consent to move forward with development. (The team may decide to extend this two week deadline on a case by case basis, but only before the two week period has passed. After two weeks, the design will move forward as-is.)
 - This does mean that some REST features may need future improvement if the initial design is found to later have RESTful design flaws. Such flaws will need to be noted in a bug ticket (to ensure we don't repeat them in other features) and fixed in follow-up work.
 - *REST API Backwards Compatibility support*
 - During 7.x development, we REQUIRE backwards compatibility in the REST API layer between any sequential 7.x releases. This means that the 7.1 REST API must be backwards compatible with 7.0, and 7.2 must be compatible with 7.1, etc.
 - However, deprecation of endpoints is allowed, and multi-step 7.x releases may involve breaking changes (but those breaking changes must be deprecated first & documented in Release Notes). This means that it's allowable for the 7.2 release to have changes which are incompatible with the 7.0 release, provided they were first deprecated in 7.1. Similarly, 7.3 might have breaking changes from either 7.1 or 7.0, provided they were deprecated first.
 - After 7.x development, no breaking changes are allowed in minor releases. They can only appear in major releases (e.g. 7.x8.0 or 8.x9.0 may include breaking changes).

Issue Triage process for 7.3

- **Overview of our Triage process:**
 1. *Initial Analysis:* [Tim Donohue](#) will do a quick analysis of all issue tickets coming into our [Backlog Board](#) (this is where newly reported issues will automatically appear).
 2. *Prioritization/Assignment:* If the ticket should be considered for this release, [Tim Donohue](#) will categorize/label it (high/medium/low priority) and immediately assign to a developer to further analysis. Assignment will be based on who worked on that feature in the past.
 - a. "high priority" label = A feature is badly broken or missing/not working. These tickets must be implemented first, as ideally they *should be resolved* in the next release. (Keep in mind however that priorities may change as the release date approaches. So, it is possible that a "high priority" ticket may be rescheduled if it is a new feature that cannot fit into release timelines.)
 - b. "medium priority" label = A feature is difficult to use, but mostly works.. These tickets *might* be resolved prior to the next release (but the release will not be delayed to fix these issues).
 - c. "low priority" label = A feature has usability issues or other smaller inconveniences or a non-required feature is not working as expected. These tickets are simply "nice to have" in the next release. We'll attempt to fix them as time allows, but no guarantees are made.
 3. *Detailed Analysis:* Developers should immediately analyze assigned tickets and respond back within 1-2 days. The developer is expected to respond to [Tim Donohue](#) with the following:
 - a. Is the bug reproducible? (If the developer did not understand the bug report they may respond saying they need more information to proceed.)
 - b. Does the developer agree with the initial prioritization (high/medium/low), or do they recommend another priority?
 - c. Does the bug appear to be on the frontend/UI or backend/REST API?

- d. Does the developer have an idea of how difficult it would be to fix? Either a rough estimate, or feel free to create an immediate PR (if the bug is tiny & you have time to do so).
- e. Are you (or your team) interested in being assigned this work?
- 4. **Final Analysis:** [Tim Donohue](#) will look at the feedback from the developer, fix ticket labels & move it to the appropriate work Board. If it is moved to the [Project Board](#), then the ticket may be immediately assigned back to the developer (if they expressed an interest) to begin working on it.
 - a. If the ticket needs more info, [Tim Donohue](#) will send it back to the reporter and/or attempt to reproduce the bug himself. Once more info is provided, it may be sent back to the developer for a new "Detailed Analysis".

On Hold Topics

- Discussing Release Support now that 7.x is out
 1. The [DSpace Software Support Policy](#) notes that we support the "most recent three (3) major releases" (where a [major release is defined by the changing of the first number](#), e.g. 6.x 7.x). This would mean that 5.x, 6.x and 7.x are all supported at this time.
 2. Should we propose to Committers / Governance a change of policy to the "most recent two (2) major releases"? This would mean that we move to only supporting 6.x and 7.x.

Notes

- **Additional feedback on Administrative Control Panel:**
 - Luca demoed the Spring Boot Admin Server user interface.
 - We decided that we'd still like a very basic Angular UI for the "Control Panel". It should ONLY display the "info" and "health" actuators, which align well with the sort of basic status updates that the old DSpace XMLUI Control Panel had.
 - Anyone who wants more complex "Control Panel" functionality should install/use the Spring Boot Admin Server. We should add instructions on how to do so, as this is a very powerful tool for anyone who wants more insight into their DSpace site's performance.
 - The old XMLUI Control Panel also included a "Systemwide Alerts" section. However, this needs to be split out as a separate Ticket, as it cannot be achieved with Spring Boot Actuators. It's more an Angular UI task.
- **Additional feedback on DSpace 7 Versioning with Entities:** https://docs.google.com/document/d/1E7_zUse5ojl_N9hks53GS3NGboXKkxkF6ud1lPmcDP8/edit#
 - Ben went over updates of the new (edge) use case: two Publications linked via a relationship where Publication A is the review of Publication B. If Publication B is then versioned into B2, it may be important for Publication A to somehow retain an "original" link to Publication B (as that was the version which was reviewed), even if B2 is now the latest version.
 - Feels that the current design can be extended to support this use case easily
 - Mark W noted that this use case may need to be a separate **task**. Start simple and address the edge cases later. Tim agrees.
 - Andrea said he's not sure it's that hard to achieve this use case, but also says the approach of splitting this up is fine.
 - Updated estimate from Ben of 80-120 hours
 - All agree that the original design is good as-is. Minor feedback is still welcome on either the ticket (<https://github.com/DSpace/DSpace/issues/7937>) or the Google Doc. Development can begin.
 - All agree that the edge case can be a secondary PR later on. However, if Ben finds a way to achieve it easily, it's OK to achieve in the same PR. As of today though, it seems like it may need more analysis & therefore is not required for the first PR.
- E2E tests - Needs more discussion of what we mean by this. Will bring to next week's meeting.