2019-10-24 DSpace 7 Working Group Meeting

Date

24 Oct 2019 from 14:00-15:00 UTC

As of Oct 31, 2019 this meeting is switching to 15:00 UTC (as daylight savings ends in Europe on Oct 27). This means the local time will be unchanged for European attendees.

In North America, as daylight savings ends one week later (on Nov 3), attendees will attend this meeting at 11:00am EDT on Oct 31, but as of November 7 it will switch back to 10:00am EST.

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

- More connection options available at DSpace Meeting Room

Agenda

- (15 mins) 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. (Topic #1)
    2. Reviewing & scoping tasks flagged as "NEEDS MORE INFO" in Development Planning Spreadsheet
        a. This is necessary to prepare for the next phase of the DSpace 7 Estimation Process
- (15 mins) Planning for next week
    - Assigning PRs for Review
    - Next tasks from Development Planning Spreadsheet

Attendees

- Art Lowel (Atmire)
- Andrea Bollini (4Science)
- Tim Donohue
- Lieven Droogmans
- Mark H. Wood
- Giuseppe Digilio (4Science)
- Ben Bosman
- Chris Wilper
- Paulo Graça
- Dimitris Pierrakos
- Julius Gruber
- Laura Henze

Current Work

Legend for status icons

⭐ = Highest Priority tasks (please prioritize these reviews/tasks over others). These are tasks with lots of dependencies

❌ = review done, changes were requested or bugs found.

✅ = review done, approved.

⚠️ = review done, merge conflict or other minor changes requests

1 APPROVAL = pull request only requires a single approval to merge. This is generally reserved for PRs which are either smaller, obvious, and /or bug fixes with tests to prove they work.

Tickets / PRs In Progress

1. (Angular) Adding Accessibility via Travis CI https://github.com/DSpace/dspace-angular/pull/356 (work in progress) (*Lower priority*)
2. ⚠️ (Angular Bug) https://github.com/DSpace/dspace-angular/issues/368 (Art Lowel (Atmire))
3. (REST Contract) Edit Homepage news: https://github.com/DSpace/Rest7Contract/pull/45 (Ben Bosman - has outstanding questions/comments) (*Lower priority*)
4. (REST) DS-4043: Revisit the security layer of the submission (work in progress) Andrea Bollini (4Science)
5. (REST) Pagination bug with withdrawn items: https://github.com/DSpace/DSpace/pull/2406 (Dimitris Pierrakos , Ben Bosman - Feedback provided)
6. (REST Contract) ResourcePolicy endpoint - coming before next meeting

PRs Needing Review

1. (REST Contract) Group and eperson management: https://github.com/DSpace/Rest7Contract/pull/41 (Tim Donohue - feedback provided, Andrea Bollini (4Science) - ⚠️ feedback provided)
2. **(NEW)** (REST Contract) bundle DELETE contract https://github.com/DSpace/Rest7Contract/pull/86   `1 APPROVAL`   (✅ Tim Donohue)
3. (REST) Scripts and processes endpoint https://github.com/DSpace/DSpace/pull/2529 FEEDBACK HAS BEEN PROCESSED (Tim Donohue, Dimitris Pierrakos) - Was reviewed by Ben Bosman internally.
4. (REST) DS-4337 implement bitstream-bitstreamformat relation endpoints https://github.com/DSpace/DSpace/pull/2503 (✅ Tim Donohue ,✅ Andrea Bollini (4Science), ✅ Ben Bosman)
5. (REST) Ds 4317 bundles in rest https://github.com/DSpace/DSpace/pull/2548 (Ben Bosman, Tim Donohue, Chris Wilper)
6. (REST) REST Projections "proof of concept": https://github.com/DSpace/DSpace/pull/2547 (**Early Reviews welcome from all.**)
7. (REST) Ds 4358 tests in modules https://github.com/DSpace/DSpace/pull/2553   `1 APPROVAL`   (Tim Donohue , **NEEDS SECOND REVIEWER**)
8. **(NEW)** (REST) CRUD on Collection & Community logo https://github.com/DSpace/DSpace/pull/2562 (Chris Wilper, Tim Donohue)
9. (Angular) Shibboleth integration support: https://github.com/DSpace/dspace-angular/pull/429 (Giuseppe Digilio (4Science) reviewed again **fixed error with yarn start**, Fernando FCT/FCCN, Paulo Graça - feedback provided)
10. (Angular) forceBypassCache should be removed from the RequestService: https://github.com/DSpace/dspace-angular/pull/468 (✅Art Lowel (Atmire) - approved again, Giuseppe Digilio (4Science))
11. (Angular) Routing by handle and uuid: https://github.com/DSpace/dspace-angular/pull/490 (Art Lowel (Atmire) - provided feedback, Giuseppe Digilio (4Science), Tim Donohue , Andrea Bollini (4Science) - might be able to help with pid endpoint)
12. (Angular) Tracking stats from the UI https://github.com/DSpace/dspace-angular/pull/495 (Tim Donohue , **NEEDS SECOND REVIEWER**)
13. (Angular) Refactor object lists https://github.com/DSpace/dspace-angular/pull/497 (Giuseppe Digilio (4Science), Tim Donohue)
14. (Backend) dspace.bat file: https://github.com/DSpace/DSpace/pull/2544   `1 APPROVAL`   (Tim Donohue, Alexander Sulfrian, Chris Wilper)

PRs Merged this week!

1. ✅ (REST Contract) Collection logo https://github.com/DSpace/Rest7Contract/pull/84   `1 APPROVAL`
2. ✅ (REST Contract) collection item template https://github.com/DSpace/Rest7Contract/pull/85   `1 APPROVAL`
3. ✅ (REST) Remove unnecessary dspace.restUrl and fix default configuration: https://github.com/DSpace/DSpace/pull/2554
4. ✅ (REST) Authority control bugfixes https://github.com/DSpace/DSpace/pull/2528

Blocked

1. (Blocked PRs go here)

Delayed / Needs Discussion

1. **Managing Authorization info in Angular UI**: How to pass Authorization rights (i.e. logged in user's access rights) from REST API to Angular? See for example: https://github.com/DSpace/dspace-angular/issues/393
   a. Can this be achieved via passed HAL "_links" (e.g. the existence of an "edit" link in REST response means you must have Edit rights)?
   b. In July 25 meeting, we noted this probably cannot be resolved with just one simple solution. May need to look at different options for different scenarios
      i. Also likely to need to store/cache a user's Groups in UI layer, as some areas (e.g. Administrative) require knowledge of user group m...
2. **REST API Projections:** _Unable to locate Jira server for this macro. It may be due to Application Link configuration._
   a. Early work beg...l, this approach seems like a good direction, need volunteers to move it forward.
      i. https://github.com/DSpace/Rest7Contract/issues/2 (discussion resumed by Andrea Bollini (4Science) could be relevant for the projection)
3. Initial Performance Testing from Chris.
   a. https://cwilper.github.io/dspace-perftest/
4. (REST Contract) Edit Homepage News: https://github.com/DSpace/Rest7Contract/pull/45
   a. Delayed until after Preview release. 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.
5. Concurrency in DSpace 7 (or 8). What do we want to do when multiple editors are editing the same object? Needs further analysis regarding implementation details

a. We've decided (in meeting on March 7, 2019) to use ETags to implement concurrency. REST Contract notes on ETags: https://github.com/DSpace/Rest7Contract#etags--conditional-headers
b. ETags only update of the two fields match. If someone edits first, your edit would fail and you would get a fail response (422?)
c. ETags seems to have broader support in other REST APIs. Recommended also by both Art and Andrea.

Notes

- Discussion of managing Authorization in DSpace 7.
  - Some specific use cases / needs require multiple authorization checks. For example, Moving an Item requires permissions on both the Item itself and the Collection you are moving it to.
  - Ideally, we all agree that the Angular UI should know as little about resource policies / authorization as it needs to. Longer term, it'd be great to be able to essentially ask the REST API "can I move this item to this collection?" and get a response which lets the Angular UI know whether to display that "move" button. However, that sort of logic will require backend refactoring of the Authorization system...and it'd be out of scope for DSpace 7.
  - For DSpace 7, we'll need to use an approach similar to DSpace 6...where unfortunately the UI will need to have more "understanding" of which permissions to check, and in some scenarios it might need to do multiple queries....e.g. moving an Item may require two queries... first, can this user edit the Item? second, can this user Add to the Collection?
  - Andrea notes we may be able to use HTTP OPTIONS method to do some of these requests. It's purpose is to return what HTTP actions you can perform (so, it'll say whether you have permissions to GET, POST, PUT, PATCH or DELETE). Some examples: https://codedestine.com/rest-options-restful-web-services/
  - Ben notes that sounds good, but in some scenarios there's not an easy "mapping" of an HTTP action to a permission level. For example, withdrawing an item or moving an item don't map very well necessarily.
  - There's also the problem of permission inheritance. Admins don't have explicit ResourcePolicy permissions on all objects, but inherit implicit permissions based on being a member of the Administrator Group. This implicit/explicit permission checking is performed in our AuthorizeService: https://github.com/DSpace/DSpace/blob/master/dspace-api/src/main/java/org/dspace/authorize/AuthorizeServiceImpl.java
  - Should we be creating an endpoint that simply maps to the functionality of AuthorizeService?
    - Andrea seems to agree here but was calling this the "resourcepolicy" endpoint
    - Tim notes that naming is confusing, because ResourcePolicy is a separate object with a separate Service....and those only detail **explicit** permissions, and not **implicit** ones. https://github.com/DSpace/DSpace/blob/master/dspace-api/src/main/java/org/dspace/authorize/ResourcePolicyServiceImpl.java
    - AuthorizeService is what we'd want to use to avoid rewriting the implicit permission logic there. It uses the ResourcePolicy service to gather explicit permissions as well.
  - **Next Steps: Andrea will draft a REST Contract for next week**
    - **Atmire will provide some example use cases for that REST Contract to take into account**
    - We will analyze this again next week to see how best to implement on REST Side
      - Lieven notes that simplifying the REST side can sometimes make things harder on the Angular side. Tim agrees, and notes that next week's discussion will be about finding the right balance for DSpace 7.
- Meeting ended.