DevMtg 2018-08-08

Developers Meeting on Weds, August 8, 2018

Today's Meeting Times

- DSpace Developers Meeting / Backlog Hour: 15:00 UTC in #duraspace IRC or #dev-mtg Slack channel (these two channels sync all conversations)
  - Please note that all meetings are publicly logged

Our IRC logging bot has been blocked from Freenode (as of July 27).

Discussion logs are no longer available at http://irclogs.duraspace.org/. As our current IRC log bot (based on PircBot) is unmaintained and doesn't align with Freenode policies (around requiring SASL authentication), Tim has reached out to https://botbot.me/ to see if they could log our #duraspace IRC channel. In the meantime, full logs of meeting discussions will be copied into the Wiki notes below.

Agenda

Quick Reminders

Friendly reminders of upcoming meetings, discussions etc

- DSpace 7 Working Group (2016-2023): Next meeting is tomorrow, Thurs, August 9 at 14:00 UTC
- DSpace Entities Working Group (2017-18): *This Working Group will be replaced by DSpace 7 Entities Working Group (2018-19)*
- DSpace Developer Show and Tell Meetings: Next Show & Tell will be August 28 at 15:00 UTC. The topic will be DSpace + Docker

Discussion Topics

If you have a topic you'd like to have added to the agenda, please just add it.

1. (*Ongoing Topic*) DSpace 7 Status Updates for this week.
    a. DSpace 7 Working Group (2016-2023) is where the work is taking place
    b. DSpace 7 Dev Status spreadsheet: https://docs.google.com/spreadsheets/d/18brPF7cZy_UKyj97Ta44UJg5Z8OwJGi7PLoPJVz-g3g/edit#gid=0
2. *(Ongoing Topic)* DSpace 6.x Status Updates for this week
    a. **Master ports from 6.3 are COMPLETE!. A huge thanks to Terrence W Brady , as he did the vast majority of the ports!**
        i. https://docs.google.com/spreadsheets/d/1X-Zk56gz-wg6p7JaiuBzzUquqOvwwx_-o_ZDDvGPSQU/edit?usp=sharing
    b. 6.4 will surely happen at some point, but no definitive plan or schedule at this time. Please continue to help move forward / merge PRs into the dspace-6.x branch, and we can continue to monitor when a 6.4 release makes sense.
3. DSpace and Docker
    a. Tutorial: https://dspace-labs.github.io/DSpace-Docker-Images/
    b. PRs for DSpace 4, 5, 6, 7 - see DSpace and Docker link above
4. Discussion topics / half-baked ideas (*Anything more to touch on with these?*)
    a. Bulk Operations Support Enhancements (from Mark H. Wood)
        i. Better support for bulk operations (in database layer), so that business logic doesn't need to know so much about the database layer. Specifically, perhaps a way to pass a callback into the database layer, to be applied iteratively to the results of a query.
        ii. Then, the database layer can handle batching, transaction boundaries, and other things that it should know about, and the business logic won't have to deal with them.
        iii. This is the result of thinking about a recent -tech posting from a site with half a million objects that needed checksum processing.
        iv. (This is almost an extension of the tabled topic below regarding DSpace Database Access, but a bit more specific in trying to simplify/improve upon how bulk operations are handled)
    b. Curation System Needs (from Terrence W Brady)
5. How to encourage / credit folks who do Code Reviews? (Tim Donohue)
    a. We have a lot of open PRs. As we know, the process for reviewing is very ad-hoc, sometimes encounters delays. If we can find ways to encourage/empower folks (even non-Committers if they know Java / Angular well) to do code reviews & be credited publicly...maybe we can speed up this process?
    b. Other brainstorms welcome!
6. Tickets, Pull Requests or Email threads/discussions requiring more attention? (*Please feel free to add any you wish to discuss under this topic*)

Tabled Topics

These topics are ones we've touched on in the past and likely need to revisit (with other interested parties). If a topic below is of interest to you, say something and we'll promote it to an agenda topic!

1. Management of database connections for DSpace going forward (7.0 and beyond). What behavior is ideal? Also see notes at DSpace Database Access

a. In DSpace 5, each "Context" established a new DB connection. Context then committed or aborted the connection after it was done (based on results of that request). Context could also be shared between methods if a single transaction needed to perform actions across multiple methods.
b. In DSpace 6, Hibernate manages the DB connection pool. Each **thread** grabs a Connection from the pool. This means two Context objects could use the same Connection (if they are in the same thread). In other words, code can no longer assume each `new Context()` is treated as a new database transaction.
    i. Should we be making use of `SessionFactory.openSession()` for READ-ONLY Contexts (or any change of Context state) to ensure we are creating a new Connection (and not simply modifying the state of an existing one)? Currently we always use `SessionFactory.getCurrentSession()` in HibernateDBConnection, which doesn't guarantee a new connection: https://github.com/DSpace/DSpace/blob/dspace-6_x/dspace-api/src/main/java/org/dspace/core/HibernateDBConnection.java

Ticket Summaries

1. Help us test / code review! These are tickets needing code review/testing and flagged for a future release (ordered by release & priority)

key	summary	type	created	updated	assignee	reporter	priority	status	fixversions

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

2. Newly created tickets this week:

key	summary	type	created	assignee	reporter	priority	status

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

3. Old, unresolved tickets with activity this week:

key	summary	type	created	updated	assignee	reporter	priority	status

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

4. Tickets resolved this week:

key	summary	type	created	assignee	reporter	priority	status	resolution

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

5. Tickets requiring review. This is the JIRA Backlog of "Received" tickets:

key	summary	type	created	updated	assignee	reporter	priority

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

Meeting Notes

Meeting Transcript (IRC Bot is not working)

- ~~IRC Transcript is available at - http://irclogs.duraspace.org/index.php?date=2018-08-08~~

Log from #dev-mtg Slack (All times are CDT)

```
Tim Donohue [9:49 AM]
@here: Reminder that our DSpace DevMtg starts at the top of the hour (a little over 10mins from now).  Agenda
is at https://wiki.duraspace.org/display/DSPACE/DevMtg+2018-08-08

Tim Donohue [10:00 AM]
@here: It's DevMtg time.  Agenda is above :point_up: .  Let's do a quick roll-call to see who's able to join us
today.

Mark Wood [10:01 AM]
Hi.

Alexander Sulfrian [10:01 AM]
Hi

Tim Donohue [10:02 AM]
Looks like we are a small group today :wink:  But, we may as well get started. Hopefully others are lurking or
coming online shortly

Terry Brady [10:03 AM]
hello

Tim Donohue [10:03 AM]
We'll jump into quick updates... on the DSpace 7 side, not much to say.  The summer months have been a bit slow
(as folks take vacations/holidays).  But, our next weekly meeting is tomorrow at 14UTC
And, as always, if you want to get a sense of the "current status" of DSpace 7, we've been trying to keep this
Dev Planning spreadsheet up-to-date: https://docs.google.com/spreadsheets/d
/18brPF7cZy_UKyj97Ta44UJg5Z8OwJGi7PLoPJVz-g3g/edit#gid=0
On the DSpace 6.x front, I'm very happy to report that we've cleared the queue of `port to master` PRs from
GitHub (these were PRs that were merged/released in 6.3, but hadn't yet been merged into `master`).   Most of
the heavy lifting was done by @terrywbrady :clap:
That's it for the 6.x updates though.
I'll pause here for a moment to see if there are any questions/comments on 7.x or 6.x updates?

Alexander Sulfrian [10:07 AM]
I have seen, that there are two massive pull requests from 4science for DSpace7. One with 63k added lines and
one with 11k added lines. It will be not an easy task to review, becaus it is mostly everything in one commit. :
confused:

Tim Donohue [10:09 AM]
@sulfrian: those were expected PRs. They let us know in advance (in DSpace 7 meetings) that these were coming.
The work in those PRs was demoed at OR2018 (and in this DSpace 7 update video: https://youtu.be/yKnos2jTdSQ).
But, yes, there is a lot of code to review/test
Unfortunately, the effort there was not able to easily come back into `master` in small pieces...which is ideal.
I plan to still help review/test those PRs myself.
I can definitely understand though that it's frustrating to see such large PRs.  It does go against our best
practices for PRs, but we (the DSpace 7 team) had already decided to make an exception for this scenario

Alexander Sulfrian [10:13 AM]
Ok, I only would like to encourage people to make pull requests for smaller steps in the future.
(Or at least have multiple smaller commits in a bigger pull request.)

Tim Donohue [10:14 AM]
```

@sulfrian: yes, that's our usual policy & I agree completely. This likely will be one of the few scenarios where we'd make an exception for DSpace 7 -- and part of the issue is that this submission/workflow feature was developed as part of a separate project, and has now been "gifted" to DSpace 7. So, it's hard to "split the code" back up into small PRs
But, thanks for noting your concern here. I share that concern
Moving back to our agenda now. This was on the agenda for last week, and I wasn't sure if @terrywbrady wanted to touch on it more. DSpace + Docker: https://wiki.duraspace.org/display/DSPACE/DSpace+and+Docker

Terry Brady [10:17 AM]
The Dockerfile has been merged on 6x. There are 3 variants that need merging as well: 4x, 5x, 7x.
#2137 - 4x (build.properties)
#2136 - 5x (build.properties) + Mirage2 build
#2135 - 7x (local.cfg) + webapps
Once those are merged, I want to re-visit granting build rights to DockerHub.
Tim, perhaps you and I could meet and step through the rights granting process on GitHub. A trigger is inserted that will rebuild Docker images anytime our branches are udpated.

Tim Donohue [10:19 AM]
@terrywbrady: is it worth looking at DockerHub permissions more immediately (e.g. getting 6.x working as you want)?
Sure, I can meet to look at the rights granting process in GitHub as needed

Terry Brady [10:19 AM]
If you have time today, lets do it. Otherwise, we can do it when I am back in the office on the 20th.

Tim Donohue [10:20 AM]
I should have time today...pretty much anytime after this meeting is free

Terry Brady [10:20 AM]
For those new to Docker, I added some videos to the tutorial pages: https://dspace-labs.github.io/DSpace-Docker-Images/
Docker for DSpace Testing and Development
DSpace-Docker-Images
Optimize Your DSpace Development Processes using Docker
Great. I will catch you after this meeting.
The DSpace Dev Show and Tell on Aug 28 will focus on Docker: https://wiki.duraspace.org/display/DSPACE/Dev+Show+and+Tell+-+Aug+28%2C+2018+-+1500UTC+-+DSpace+On+DockerHub

Tim Donohue [10:22 AM]
Definitely looking forward to the next Dev Show & Tell on this!
Ok, anything else to mention on this topic? Should we move along?

Terry Brady [10:22 AM]
I am fine to move along

Tim Donohue [10:22 AM]
Ok, next up... under #4 on the agenda are a few brainstorms / discussion topics.
First, we had past discussion on an idea from @mwood about "Bulk Operations Support Enhancements". Did you have more to discussion on this @mwood? Are we nearing a wiki page or ticket to describe the idea?

Mark Wood [10:24 AM]
Bulk operations has been discussed a bit, and I haven't revisited it yet. Thanks for reminding me that I need to flesh this out a bit.
Curation System Needs was starved by the other discussion, so maybe we could talk about that today?

Tim Donohue [10:25 AM]
Ideally, we should minimally move this into a Wiki page or similar...it seems worth tracking (and possibly not just in this weekly agenda)

Mark Wood [10:25 AM]
OK, I'll see what I can work up.

Tim Donohue [10:25 AM]
Sure, we can move on to Curation System though, if you'd rather.
This one already has a wiki page of brainstorms: https://wiki.duraspace.org/display/~terrywbrady/Curation+System+Needs

Terry Brady [10:26 AM]
When I first started working in DSpace, I needed to develop some simple extensions - mostly reporting stuff.
Curation tasks seemed like an easy way to develop and deploy a simple add-on.
But, there are some limitations to the current curation process. It does not take parameters (other than a

scope handle) and it does not really persist output.
I *think* that many CLI and Admin functions could be reduced to curation tasks if the input/output issues around curation were resolved.

Mark Wood [10:28 AM]
I recall something like parameters for tasks, but it's really hard to find any information about them.

Terry Brady [10:28 AM]
As we moved to the REST7 api, it will become more complicated to make features available both to (1)the Angular UI and (2) the CLI interface. Perhaps curation could solve this.
The only params I remember for curation are (1)write text output to STDOUT and (2)run immediately vs queue for later

Mark Wood [10:30 AM]
They're called Task Properties.

Tim Donohue [10:30 AM]
Most curation params are actually *configuration*
(so, it's accurate to say you cannot pass params on commandline or similar)

Mark Wood [10:30 AM]
It's not quite the same thing, but it does allow configuring the same task code to be run in more than one way.

Alexander Sulfrian [10:31 AM]
Yes, task properties are a workaround for missing parameters.

Terry Brady [10:31 AM]
That is good to know.

Mark Wood [10:31 AM]
I can see that properties may not be flexible enough.

Tim Donohue [10:32 AM]
In any case, I agree that Curation Tasks are limited...especially in output format. And that they take input more from configuration (instead of params)

Terry Brady [10:32 AM]
I remembered my brainstorming on this as a possible way to address @mwood’s bulk operations needs.

Tim Donohue [10:34 AM]
Regarding Curation Task output, I think the most logical extension there would be to support JSON output. To bring Curation Tasks to the REST API would require either the current text output (embedded in JSON) or straight JSON output
I don't see as much usefulness though to HTML or XML output...as our REST API speaks entirely JSON, and the DSpace 7 UI can always format that output into HTML

Mark Wood [10:35 AM]
Tasks have been rather free to write anything they like. Other than wrapping strings to make them legal JSON, it may take a lot of work (and cramp a lot of style) to structure the output.

Terry Brady [10:35 AM]
That makes sense. We might also want to generate some html fragments as reports.

Alexander Sulfrian [10:36 AM]
Would be good if the UI/reporting after running a curation task on multiple items can be improved.

Mark Wood [10:36 AM]
If we get structured stuff out, it can be transformed to HTML or anything else.

Tim Donohue [10:36 AM]
@sulfrian: yes, I think that's caused mostly by the text-based output format. It's hard to display plain text in a UI in a "pretty way"

Terry Brady [10:37 AM]
Sounds good. As long as link-like things can be written out, the format is less important.

Alexander Sulfrian [10:37 AM]
@tdonohue Currently only the result of the last item is displayed. That's a bit unexpected for users.

Tim Donohue [10:39 AM]
@sulfrian: I think that's a result of the lack of persistence of the output....you get the full output on

commandline (as it's written out as each item is processed). In the UI though, it's hard to write output during processing without Javascript/dynamic output.
If we persisted the output, we could load it all together and provide a full view of the output (in the UI)
Or, with the Angular UI, we might be able to build the dynamics here a bit better than the current UIs

Mark Wood [10:41 AM]
Sounds like the first thing we need is to replace setResult(String) with addResult(String). On the console it just writes to the console; in a webapp. it is accumulated somewhere, or fed out via AJAX or whatever. (edited)

Terry Brady [10:41 AM]
It could be useful to share some of this discussion in the DSpace7 meeting to see if this approach could make any of that development work easier...

Tim Donohue [10:41 AM]
This is a good discussion to have now though, as it's not a feature that is enabled/built yet in DSpace 7.
I agree these ideas should be documented somewhere for DSpace 7 team. I'm not sure it should go into a DSpace 7 meeting yet though, until we are ready to work on it. But, we could create a ticket for DSpace 7 REST API to discuss implementation ideas

Mark Wood [10:44 AM]
Likewise Curator.getResult() should return List<String>.

Tim Donohue [10:45 AM]
The reality here though is that it's highly likely we *won't* be able to rebuild or heavily enhance Curation Tasks in DSpace 7 (we just don't have the time to redesign everything)....but, if there are minor enhancements necessary to help make it work better for REST / Angular, those could/should happen
And it sounds like we've identified at least a few minor enhancements.... namely persisting the output (or feeding via AJAX like streams)....and possibly looking towards a JSON output format

Terry Brady [10:46 AM]
(I need to step away for 5 min. I will rejoin you all in a moment)

Tim Donohue [10:46 AM]
No worries
Ok, so it sounds like we are wrapping up this discussion. I think the task here is to create a ticket (or two) on implementation ideas for the DSpace 7 team

Mark Wood [10:47 AM]
I'm not sure we can do much more than {"string", "string"...}

Tim Donohue [10:48 AM]
@mwood: we may not be able to. I'm uncertain as well...but I think we should be able to "stream" updates to the Angular UI (to allow it to "persist" output at least until the task completes)
I can write that up in an Angular UI ticket as an idea/brainstorm (and link in this discussion)

Terry Brady [10:49 AM]
I am back...

Mark Wood [10:50 AM]
Curation really doesn't have much structure other than "ran task on object 1; ran task on object 2...."

Terry Brady [10:51 AM]
In some instances, we will want simple feedback from curation (a message) and in some instances we will want feedback persisted (a report that requires follow-up action). It would be nice to have a curation system option that could do either.

Tim Donohue [10:52 AM]
@terrywbrady: I think I agree. I think we need to separate here what is "doable in DSpace 7" versus what is likely "future enhancements"
I suspect the doable in DSpace 7 is more about taking the current system & making sure the UI is better (i.e. streaming results to the UI, so that it can display them all in a nice format)

Mark Wood [10:53 AM]
An AbstractCurationTask has several getXXXProperty() methods, and these could be extended with "dynamic" properties that are taken from the request rather than configuration. I think tasks could easily not care how their properties were set.

Tim Donohue [10:53 AM]
Future enhancements could include a bigger overall to find a place to persist reports (more permanently), etc

Mark Wood [10:54 AM]

Commandline task runs can already save reports wherever they like. Where would a GUI run usefully save reports? Probably just build a document that can be saved by the browser.

Tim Donohue [10:55 AM]
@mwood: if a report were saved in a semi-structured format on the backend, then the front-end should be able to transform it into JSON (for the UI) in the same way that it would do so for a "live" task.
But, I think that's likely out-of-scope for DSpace 7 timelines...nonetheless, it's worth thinking about / brainstorming in a Wiki page for future enhancements

Terry Brady [10:56 AM]
Since curation pushes some tasks to a background queue, there may not always be a UI to persist.

Mark Wood [10:56 AM]
The question is: where does it go? Format as JSON or whatever you wish, then make it an "attachment". I forget the details, but I've done that before.

Terry Brady [10:56 AM]
(Tim, no need to repeat the caution about DSpace 7 scope)

Mark Wood [10:57 AM]
Background: good point. We'd need a place to store reports, then.
I think the original idea may have been "just log the details." There's some special support for logging. But the logs are a junkpile already....

Terry Brady [10:58 AM]
I like that the notion of foreground/background is already there. A developer does not need to decide which approach to use until execution time.

Tim Donohue [10:58 AM]
@mwood: To be honest, if the output is plain text...it could be a plain text file (like a log file). If the REST API knows to read each line of that file and "stream" to the Angular UI, it could look very similar to what would be streamed in "live" output
But, that's just if the output remains plain text. We also could define a more structured format for the output (JSON or similar)
I think there's promise here on incremental improvement to this.... first, in DSpace 7, get the full results "streamable" to the UI (so they all can be displayed, just like you'd see in STDOUT). Then, in future, find a place to archive those results on backend, and "stream" from archived location

Pablo Prieto [11:00 AM]
Hi all

Tim Donohue [11:01 AM]
Hi @Pablo Prieto

Mark Wood [11:01 AM]
If you're going to trigger a curation run in the GUI, walk away, and review the results later *in the GUI* then we'll need to issue a "run identifier" that you can copy/paste, save, and copy/paste later to retrieve the results.

Tim Donohue [11:02 AM]
In any case, I'm realizing we are now at the top of the hour. I think the next steps here are to (1) create a implementation brainstorm ticket for DSpace 7... (2) Update the wiki page proposal (for future enhancements) with some of these ideas
I can create the DSpace 7 ticket (I think it's likely an Angular ticket initially, until we get a better handle on what would need to happen in REST API)

Terry Brady [11:03 AM]
I will capture these notes on the wiki page. I probably will not have time to organize them.

Mark Wood [11:05 AM]
I'd just like to say: don't GUIfy task output too quickly. Let the UI render it; retain any structure somewhat abstractly. I tend to want to build pipelines in scripts rather than sit there and drive everything manually.

Tim Donohue [11:05 AM]
Ok, so let's wrap up the meeting for today then. Thanks for joining the discussion today, all! The next DevMtg is Weds, Aug 15 at 20UTC.
@mwood: I think that'll likely happen naturally. I don't think we'll have time to change task output in DSpace 7, TBH. So, output format changes may need to wait. But, I think, with Angular, we should be able to "stream" task output to the UI...so that the Angular UI output looks more like STDOUT output.

Terry Brady [11:06 AM]
@tdonohue, do you want to meet and look at the Git/DockerHub build process?
Notes captured: https://wiki.duraspace.org/display/~terrywbrady/Curation+System+Needs

Tim Donohue [11:07 AM]
@terrywbrady give me ~5 mins. Then, yes, sure. We can always move over to #dev though

Mark Wood [11:07 AM]
I guess I'm saying: let the REST code turn this stuff into JSON. I'll shut up now.

Tim Donohue [11:08 AM]
@mwood: yes, we're saying the same thing. I was noting that I think that's all we'll have time for in DSpace 7...so, it's highly likely in DSpace 7, the REST code will simply turn the current output into JSON