

# DevMtg 2019-02-06

## Developers Meeting on Weds, February 6, 2019

### 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)

### Agenda

#### Quick Reminders

*Friendly reminders of upcoming meetings, discussions etc*

- [DSpace 7 Working Group \(2016-2023\)](#): Next meeting is Thurs, Feb 7 at 15:00 UTC
- [DSpace 7 Entities Working Group \(2018-19\)](#): Next meeting on Tues, Feb 19 at 16:00 UTC
  - Last meeting notes at [2019-02-05 DSpace 7 Entities WG Meeting](#)
- [DSpace Developer Show and Tell Meetings](#): On hold until interesting topics arise.

#### 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 (from [DSpace 7 Working Group \(2016-2023\)](#))
2. (Ongoing Topic) [DSpace 6.x Status Updates](#) for this week
  - a. 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. [Upgrading Solr Server for DSpace \(Mark H. Wood\)](#)
  - a. PR <https://github.com/DSpace/DSpace/pull/2058>
4. [DSpace Backend as One Webapp \(Tim Donohue\)](#)
  - a. PR: <https://github.com/DSpace/DSpace/pull/2265> (PR is in a reviewable state. SWORDv1 and SWORDv2 are merged into "Spring REST" webapp, with basic Integration Tests to prove both work)
5. [DSpace Docker and Cloud Deployment Goals \(old\) \(Terrence W Brady\)](#)
  - a. Simplify invocation by using multiple fragments, auto load content on startup
    - i. <https://github.com/DSpace-Labs/DSpace-Docker-Images/pull/68>
    - ii. Summary page: [https://github.com/DSpace-Labs/DSpace-Docker-Images/blob/master/helper\\_cmds/docker-compose-files/dspace-compose/ComposeFiles.md](https://github.com/DSpace-Labs/DSpace-Docker-Images/blob/master/helper_cmds/docker-compose-files/dspace-compose/ComposeFiles.md)
  - b. Speed up Docker builds
    - i. <https://github.com/DSpace/DSpace/pull/2307>
  - c. Add Docker build/push to Travis
    - i. This make sense to consider after 2307 is merged
    - ii. <https://github.com/DSpace/DSpace/pull/2308>
6. Brainstorms / ideas (Any quick updates to report?)
  - a. (On Hold, pending Steering/Leadership approval) Follow-up on "DSpace Top GitHub Contributors" site (Tim Donohue): <https://tdonohue.github.io/top-contributors/>
  - b. [Bulk Operations Support Enhancements](#) (from Mark H. Wood)
  - c. [Curation System Needs](#) (from Mark H. Wood)
    - i. PR 2180 improves reporting. Ready for review.
7. Tickets, Pull Requests or Email threads/discussions requiring more attention? (Please feel free to add any you wish to discuss under this topic)
  - a. Quick Win PRs: <https://github.com/DSpace/DSpace/pulls?q=is%3Aopen+review%3Aapproved+label%3A%22quick+win%22>

#### 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/master/dspace-6\\_x/dspace-api/src/main/java/org/dspace/core/HibernateDBConnection.java](https://github.com/DSpace/DSpace/blob/master/dspace-6_x/dspace-api/src/main/java/org/dspace/core/HibernateDBConnection.java)

- c. Bulk operations, such as loading batches of items or doing mass updates, have another issue: transaction size and lifetime. Operating on 1 000 000 items in a single transaction can cause enormous cache bloat, or even exhaust the heap.
- i. Bulk loading should be broken down by committing a modestly-sized batch and opening a new transaction at frequent intervals. (A consequence of this design is that the operation must leave enough information to restart it without re-adding work already committed, should the operation fail or be prematurely terminated by the user. The SAF importer is a good example.)
  - ii. Mass updates need two different transaction lifetimes: a query which generates the list of objects on which to operate, which lasts throughout the update; and the update queries, which should be committed frequently as above. This requires *two* transactions, so that the updates can be committed without ending the long-running query that tells us what to update.

## 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

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

Tim Donohue [8:56 AM]

@here: in ~4 minutes is our DSpace DevMtg in this channel. The agenda for today can be found at <https://wiki.duraspace.org/display/DSPACE/DevMtg+2019-02-06>

@here It's DSpace DevMtg time. Agenda above. Let's do a quick roll call of who is available today

Terry Brady [9:02 AM]

hello

Mark Wood [9:03 AM]

Hi

Tim Donohue [9:03 AM]

Good morning Terry & Mark. At least we have a trio :wink:

We'll go ahead and get started here. If needed, we can wrap up this meeting early.

Patrick Trottier [9:05 AM]

Hello

Tim Donohue [9:05 AM]

On the DSpace 7 side of things, I'll note that development is still highly active, but not major updates.

Anyone wanting a latest status is welcome to either lurk on meetings, or check meeting notes.

Currently we could use more code reviewers, and the list of tickets we concentrate on are in the weekly meeting agenda. This week's agenda (for Thurs) is at <https://wiki.duraspace.org/display/DSPACE/2019-02-07+DSpace+7+Working+Group+Meeting> (Check out the "Current Work" section for latest work)

Hi @Patrick Trottier welcome back!

So, I don't have specific updates to share on the DSpace 7 side, and there's a lot of overlap between attendance in that meeting & who is here today. But, if there's any questions/comments, please feel free to ask

Terry Brady [9:08 AM]

I completed my 2 reviews for DSpace 7 although I am waiting for merge conflicts to be resolved in order to test. Is there a yellow checkmark I could apply to note that status.

Tim Donohue [9:11 AM]

@terrywbrady: you are talking about in the DSpace 7 meeting agenda's "Current Work" section? If there's merge conflicts or requests for changes, I think we need to put it back in "In Progress" and reassign to the developer / team. I haven't had time myself to look at that "Current Work" section to reorg for tomorrow's meeting yet.

The red/green icons were added by Andrea, I see. I'm not sure if there's a yellow one, we'd have to look at the icons that come with Confluence.

So, I'll take a look at this between meetings today...this new format for our DSpace 7 Meetings is still very much a work-in-progress (as you know), so it's possible this needs a bit of cleanup to make it easier to note the exact status.

Terry Brady [9:14 AM]

Sounds good. I see that one of the PR's was updated since yesterday, so I should be able to complete my testing.

Tim Donohue [9:15 AM]

Good. Yes, in general, I'd say ping the developer in the GitHub PR to let them know about the merge conflict. Often, they'll get it fixed quickly...unfortunately, though, GitHub doesn't notify anyone of merge conflicts, so it's always good to ping people when discovered

Moving along for now...

On the DSpace 6.x side, I don't have any updates here either. This is still a standing topic on our agenda,

but it's really waiting on someone to "take the reins" and bring the next 6.x release forward. Right now we're simply in a holding pattern until we find someone with time  
Anyone have other questions/comments though on 6.x before we move along?  
No one typing, so I'll assume not. Let's move along  
Next up is our weekly update on Solr Upgrade status :wink: <https://wiki.duraspace.org/display/DSPACE/Upgrading+Solr+Server+for+DSpace>  
Any updates this week to share, @mwood?

Mark Wood [9:19 AM]

I've begun looking at schema updates. @terrywbrady told me where to look for the changes he's been making. It looks like the actual schema changes are basically just replacing deprecated/obsolete quantifier fieldTypes with the \*pointField types.  
The authority schema is nice and small, and I've cleaned out remaining unused gunk and will push the changes up for inspection. The others are very cluttered and will take more work.

Tim Donohue [9:21 AM]

Ok, yes, working schema by schema seems fine. Is there 4 schemas in all? (trying to recall)

Mark Wood [9:22 AM]

Yes: authority, oai, search, statistics.

Tim Donohue [9:23 AM]

Sounds good. I'd say maybe add a checklist of all 4 schemas to the PR description, and that gives us a good sense of what to expect is done. Hopefully we can get an early "sanity" review after the first 1-2 schemas are completed, while you continue to work on the larger ones.

Mark Wood [9:24 AM]

The oai and search cores can be recreated if we make changes that require that, but we probably need to dump /reload the authority core and definitely the statistics core.

Tim Donohue [9:26 AM]

Right, that'd be part of the migration process. I know we already have a dump/reload option for the statistics core. We could do something similar for the authority core (if it doesn't already exist)

Mark Wood [9:26 AM]

Ah, that's for updates, not for fresh\_install, so, later.

Tim Donohue [9:26 AM]

Exactly, let's do fresh\_install first. That way we can then test out ideas for upgrades/migrate

Mark Wood [9:27 AM]

Indexes \*will\* need rebuilding, since we are changing fieldtype classes.

Tim Donohue [9:27 AM]

So, it sounds like next steps are to post the updates to the first 1-2 schemas. Then, maybe @terrywbrady or I (or both) could give it a quick look to ensure it all looks reasonable (I suspect it should).

Mark Wood [9:28 AM]

OK

Tim Donohue [9:29 AM]

Testing might prove difficult until the "search" schema is ready to go...as I think you need that to work right to even get to "statistics" or "authority". But, we can at least do a code review

Mark Wood [9:29 AM]

One thing I've done in authority is to strip out most of the commentary about Solr. I think that our schemas should comment what DSpace is doing, and if you want to know about Solr you should read the stock sample schemas and the Solr doco.

Terry Brady [9:30 AM]

<https://github.com/DSpace-Labs/DSpace-Docker-Images/pull/65> illustrates how the new solr configuration could be tested with Docker. Once the schema stuff is part of the DSpace code base, I will re-work this PR to provide it as an option for testing the changes.

Tim Donohue [9:30 AM]

@mwood: I like that direction. We should have comments obviously on what our custom fields are for, etc. (which may take time to figure out). But, I agree, we don't need all the stock comments that come in the Solr schema by default (especially those for features we don't use)

Mark Wood [9:30 AM]

I would like to whack out that big comment at the top, but it contains Apache's copyright claim and I'm not

sure how much we have to modify before the file becomes *\*ours\**.

Terry Brady [9:30 AM]

Are you creating schemas or are you using the API to tell what is needed in the schema?

Mark Wood [9:31 AM]

I'm just updating the schema.xml.

Terry Brady [9:31 AM]

At one time, I think either you or Tim recommended using the API to build up the schema.

Tim Donohue [9:32 AM]

We talked about the Solr Schema API being an option for the future, but as no one has experience with it (yet), we might be better off working with the "schema.xml" for 7.0.

Mark Wood [9:32 AM]

I could be energetic and look it up, but I'll be lazy and ask: does that API depend on cloud mode? We haven't decided whether and in what cases we will require cloud mode. Some of the APIs aren't available in legacy mode.

Terry Brady [9:33 AM]

Here is the API structure to build the authority schema: <https://github.com/DSpace-Labs/DSpace-Docker-Images/blob/1b137e478f5bcf32a9e6ca345a56768a7d00a82c/dockerfiles/dspace-solr/schema-mods/authority-schema.json>

Mark Wood [9:33 AM]

Consider also that, if the people who run Solr are not the people who run DSpace, one might not be *\*allowed\** to submit schemas via API.

Terry Brady [9:34 AM]

I had initially created schema files and I think one of you suggested using the API.

Tim Donohue [9:36 AM]

FWIW, here's a StackOverflow question on using "schema.xml" (manual updates) vs "managed-schema" (Schema API): <https://stackoverflow.com/a/41833677/3750035>

There are noted benefits to the Schema API listed there, but the big drawback listed is having a *\*version controlled schema\**, which is something we kinda need with DSpace (we need a schema for DSpace 7 vs 8 vs 9) So, while I'm OK with us looking at the Schema API further (to see if this answer is wrong or misleading), I'm under the impression that we may want to stick with the "schema.xml" way of doing things ...*\*at least for DSpace 7\**

So, my assumption is we'll not use the Schema API yet. If someone comes up with a strong, thought out argument for using Schema API, we can talk it through. But, right now, I suggest we move forward with schema.xml (edited)

Mark Wood [9:39 AM]

The version control argument looks right. Once you let Solr manage the schema, the schema is whatever Solr was last told it should be, and that knowledge only exists in the cores.

Tim Donohue [9:40 AM]

@mwood: I don't know if we have any notes about the Schema API in the wiki page you started, but maybe it's worth capturing *\*why\** we are moving forward with schema.xml (at least for now). The main point being that we think we need a version controlled schema.

Mark Wood [9:40 AM]

I will add that.

Tim Donohue [9:40 AM]

at least that will document what we currently know...if we learn more, we can change those notes Thanks!

Any other thoughts/questions/comments on Solr Upgrade?

Mark Wood [9:42 AM]

Just that I try to keep in front of me the idea that, with a separately maintained Solr, we may have to ask others to manage our cores for us in some ways.

Tim Donohue [9:43 AM]

Yes, I think that's an accurate note. Personally, though, I don't yet know what that "means", and I feel we'd discover a lot more about what that "means" once we have the fresh\_install process working. But, yes, this will be quite different, and need a lot of explanation and/or documentation

Mark Wood [9:44 AM]

Yes. So far what I'm sure of is that one will have to gather up the empty cores provided, and convey them to

the Solr instance. That could be as simple as 'cp -r' or it could mean putting them on a USB stick and asking the Solr admin. to install them pretty please.  
And tell me where the URIs should point.  
Essentially we have two databases now and must deal with two DBAs. In the simple case, you are both DBAs.

Tim Donohue [9:46 AM]

So, I think this is all good to be thinking about, but I wonder if it's best to write down brainstorm / rough notes of things we need to consider, so that we can revisit in greater detail once we have a "fresh\_install" (that we can test ideas against)?

Mark Wood [9:47 AM]

Yes.

Right now I'm trying to limit myself to avoidance of painting ourselves into a corner.

Tim Donohue [9:48 AM]

Once that fresh\_install is at a place where we can review it & spin it up locally, we can set aside most (or all) of a meeting to dig deeper on these questions.

Mark Wood [9:48 AM]

OK

Tim Donohue [9:48 AM]

I don't think we'll be in a corner.. We're forced into this model by Solr itself (in that we cannot bundle Solr internally at all). But that said, there likely are areas we'll be able to better smooth out along the way  
Ok, so, we are running short on time, so I'd recommend bringing these ideas to the wiki page. We'll do a deep dive in a future meeting

Mark Wood [9:50 AM]

Will do.

Tim Donohue [9:50 AM]

I'd like to move along to one more topic for today... DSpace 7 Backend as One Webapp (or merging REST v7, OAI, SWORDv1, SWORDv2, and RDF into one Spring Boot webapp)

PR: <https://github.com/DSpace/DSpace/pull/2265>

Background idea: <https://wiki.duraspace.org/display/DSPACE/DSpace+Backend+as+One+Webapp>

I wanted to bring this back up in this meeting as it's moved further along since last discussed, and the PR itself is in a reviewable state

The PR isn't 100% finished, but currently the PR merges RESTv7, SWORDv1 and SWORDv2 into one webapp (the REST v7 one) and provides very basic Integration Tests to prove the SWORD endpoints "work"

So, only the OAI and RDF webapps are left to merge...and they will be merged in a very similar way to the SWORD ones (and again I'll add ITs to prove they work)

I don't expect you all to have had time to do a deep review here, but I'd like to add this as an ongoing topic here, as it's something that I could use more eyes on. It's got support, but there are some significant conceptual changes here between how this worked in DSpace 6 vs in this PR

So, consider this more of an re-introduction to this idea. If you have thoughts/comments/questions immediately though, we have about 5mins left here.

And we can also keep this on the agenda for next week

Any thoughts/comments/questions, or any major concerns, to note today?

Mark Wood [9:55 AM]

I need to look through the code some more.

Tim Donohue [9:56 AM]

understandable. I expected more higher level thoughts/comments for today. We can dig in to specifics more in future meetings

I will note a few major changes. One: As all webapps become one, these individual "endpoints" are now configurable individually. So, for example there's a `swordv2-server.enabled` config to allow you to turn SWORD v2 on/off

Mark Wood [9:57 AM]

Nice.

Tim Donohue [9:57 AM]

There's also a `swordv2-server.path` config to let you specify which path you want SWORDv2 to respond on

Mark Wood [9:57 AM]

Also nice. OTOH we still need some way for remote users to discover our services.

Since they can move around.

We had that problem already. Lots of sites using /oai, for example, but here it's /pmh because that's the protocol.

Tim Donohue [9:58 AM]

Each webapp's web.xml is now gone (deleted). It's been replaced by a Spring Boot "Configuration" class which actually "wires" up all the Servlets / paths (in the exact same manner as a web.xml did). That's how we can dynamically configure paths & turn them on/off

@mwood: yes, good point. We could consider adding information into the HTML <head> or similar for the webapp  
But, that's something TBD at this point. I can add it for consideration on the Wiki page though

Mark Wood [9:59 AM]

Yeah, throwing everything I learned about web.xml in the trash because Spring does everything differently is my chief dislike here.

There is actually an RFC for such discovery. We'd have to work with others to get specific wellknown services defined.

Tim Donohue [10:01 AM]

At the very least, I'll add in notes that we should figure it out. I don't think this webapp/path "discovery" is going to happen in this PR, but it could happen as a followup.

Mark Wood [10:01 AM]

Of course, if we could move the SWORD users over to ResourceSync, that is already wellknown....

Tim Donohue [10:02 AM]

ResourceSync is coming, and I'd envision it being added into this same One Webapp in the same manner. Right now, it's in a PR though as a separate webapp

In any case, we are running over time here. I'd just recommend taking a closer look at this One Webapp idea, and we can make time to discuss more next week.

I'd love more feedback here in general, it is a different idea for DSpace...but, I honestly think it'll make installation a bit easier for DSpace 7. But, I welcome your feedback (positive or negative) in the coming weeks  
In any case, I need to run along now. So, we'll have to wrap this up for today. If there are agenda items for next week, feel free to pass my way

Thanks for attending & have a good rest of your week!

Mark Wood [10:05 AM]

Thanks!