

DevMtg 2012-05-09 - 3.0 Planning

- [DSpace 3.0 Planning Meeting on Weds, May 9, 2012](#)
 - Join via Skype + IRC backchannel
 - [Connection Instructions](#)
 - [Agenda - 3.0 Discussion & Planning](#)
 - [Meeting Notes](#)
 - [Action Items](#)

DSpace 3.0 Planning Meeting on Weds, May 9, 2012

Special Topic Meeting via Skype

 This meeting will be a 3.0 "Special Topic" Meeting that will take place via Skype (audio only) or phone. We also recommend logging in to [#duraspace IRC](#), as it will be utilized as a "backchannel" (to share links as needed).

Join via Skype + IRC backchannel

- **When:** This call will begin at 20:00UTC (the same time as our weekly DSpace Developers Meeting). We will try to keep discussion to one hour.
- **Where:**
 - Call in via Skype or Phone line - See [#Connection Instructions](#) below
 - In addition, [#duraspace IRC](#) will be utilized as a discussion "backchannel" – to share links to current discussion topics/ideas, and potentially even take notes.

Connection Instructions

You can call in via either Skype or via your Phone. **This call will be audio-only.** We will also use the [#duraspace IRC](#) as a discussion "backchannel" (to share links, etc.)

Please figure out your mute/unmute settings in advance of the call! As there may be quite a few people on the call, we will ask that you try to mute your line when you are not speaking (as this will ensure we minimize any background noise on the line).

Via Phone:

- Dial-in Number: (805) 399-1200
 - Participant Code: 929807#

Via Skype:

1. Search and Add Contact: "freeconferencecallhd.8053991200"
 - Copy and paste freeconferencecallhd.8053991200 (the entire name) into your contact search area
 - add as a contact
2. After freeconferencecallhd.8053991200 is listed in your contact list, simply call that contact.
3. Enter the participant code through the Skype keypad (not by just typing in the numbers on your keyboard): 929807# (you will prompted to add the # sign).
 - Mac Users: The Skype Keypad can be found at the top of your call window (look for the little phone keypad icon).
 - Windows Users: The Skype Keypad is unfortunately hidden under the Call -> Show Dial Pad menu
 - Linux Users: The Skype Keypad can be found at the bottom of your call window (look for the little phone keypad icon).

More info on Skype connections also available at: <http://www.freeconferencecallhd.com/skypeinstructions.html>

Agenda - 3.0 Discussion & Planning

Discussion Topics

1. [3.0 Release](#) Planning
 - Brainstorming out new features or design for 3.0, timeline/schedule, etc.

Potential 3.0 Features/Discussions?

1. Brainstorming Features/Changes for 3.0. Can we work towards development teams around any of these projects?
 - Larger DSpace projects which seem to have a lot of recent support:
 - Moving towards a Common "Business Logic" / Business Services API. (I.e. avoiding duplication of business logic in all UIs)
 - [Metadata For All](#) (i.e. metadata on all objects), SubTopic: Getting us up-to-date how we use Dublin Core / DCMI. (Support from DCAT)
 - [@Mire Resource Level Metadata \(Pull Request\)](#)
 - Various features from the [Dryad Project](#) (based on DSpace)? May or may not include the following.
 - [Item-level Versioning](#)
 - [Support for additional External Identifier Services](#) (e.g. DOI, etc.)
 - UMICH and @Mire [Advanced Embargo Support](#)
2. Other DSpace projects receiving mention recently
 - a. [Migrate Search and Browse to DSpace Discovery](#)
 - b. "Change which UIs come out-of-the-box" (and which are optionally installed later)

- c. [Move configurations to DB or similar](#) (so they can be managed/edited from Admin UI)
- d. [Context Guided Ingest](#) (or Item Type based submissions)
- e. Other [Development Proposals](#)

Meeting Notes

These notes are what Tim was able to "glean" from the interesting conversation. Please feel free to enhance or correct any mistakes as you see fit.

- Main Topic: 3.0 Release. We have a very preliminary schedule up at [DSpace Release 3.0 Notes](#)
- Richard sent some brainstorms to Committers (see [2012-05 Reflections & Brainstorms](#)) around whether we want to do any radical/significant re-architecting of DSpace for 3.0 (or start any work in parallel to 3.0). Tim attempts to summarize them (likely badly).
- Some concerns that Cocoon (technology under XMLUI) is not very well supported anymore. Do we need to look towards a new UI? Is XMLUI work sustainable if the technology underneath isn't really moving forward?
 - Likely if we stayed on XMLUI, we'd need to get more involved in Cocoon to help push it forward more & help them release Cocoon 3
 - Obviously, this does **NOT** mean we are stopping any support of XMLUI. XMLUI will continue to be supported fully in 3.0. This is more of a long term question as to whether we need to begin looking towards a new UI.
- MarkD mentions some concerns about any "rapid re-architecting" project from "mistakes" made in past, especially what we learned from the 2.0 Prototype process & the DAO Prototype (before 1.6).
- Any larger re-architecting would need to involve more committers (cannot fall on a sub-set of committers or it may not achieve wide acceptance)
- 3.0 would need to be scoped properly, no matter what it is. Cannot do everything all at once. (Sands)
- MarkD wants more info on suggested "radical re-architecting" changes from Richard
- Richard explains thought processes behind larger changes like Metadata on All Objects (high priority feature, which may require large changes)
 - Metadata on All Objects would require large scale changes in DB/API, and also major UI level changes (as most every layer of UI would be affected by newly available metadata). Do we want to make those massive changes to XMLUI? Especially if we are already wondering if ongoing Cocoon support is questionable.
- Does Admin UI even need to be the same as the main browsing/searching UI? (Mark Wood) Could we scope any project so that it just deals with **part** of the UI.
- WebMVC project does this / same with XMLUI – several not neither were "feature complete" at the initial time of release.
- Do we really want to be deciding on or supporting multiple interfaces centrally (as a Committers team), or do we want to concentrate more on "business logic" / REST / maybe one UI? (MarkD) Our resources are already a bit strapped.
- Should we look at what we have "mostly done" in terms of UIs? – i.e. put more work into [WebMVC](#) (Robin) or [SkylightUI](#) (Tim)
- MarkD: One concern, going back to Async Release discussions - pulling stuff "out of trunk" was seen as a bad thing. Limitations to achieve that had to do with community/committer feedback (most wanted a centralized "Trunk"). How does this affect UI discussions? (May need to either pull some UIs out of trunk or make it easier to allow others to build & support their own third-party UIs outside of "trunk"). Note, "Trunk" is now "master" in GitHub
- Andrea: Concerns about too large of changes. Don't want to scare off less technical folks, but want to also keep technical folks interested. 3.0 doesn't need to be any bigger than 1.7->1.8. Some things may be difficult to do gradually.
- If new releases have clear features/functionality, that helps convince people to upgrade to more recent versions. (Andrea)
- Upgrade interests are definitely driven by feature sets. (MarkD)
- Incremental Development works best for us (RobinT) - only exception is possibly Manakin. But, we could have multiple approaches at once (some folks working on architecture reworking, while others do incremental work in 3.0)
- Need to adopt a longer view (3.0 obviously won't be perfect and cannot do everything at once). Need to stick to timed releases - Andrea
- A large rearchitecting project may not be "shiny" enough yet for others organizations to contribute to - MarkD
- Could potentially leverage Spring more to not be "bound" to specific UI technologies - MarkD
- We all admit there's stuff in DSpace that "gets in the way" of evolution at times
- We need to support the building of interfaces, but need not support all interfaces ourselves (at least not centrally) - MarkW
- Throw out "parity" across UIs? Just support "parity" at a Java level (Business Logic) / REST API? - Peter
- Alternative UIs should go on GitHub, so we can all pitch in as we need & want (MarkW)
- Richard's work towards re-architecting DSpace is all in GitHub in his "MDS" project. <https://github.com/richardrogers/mds>
- MarkD : likewise @mire's work on Metadata on all objects is in GitHub as well. <https://github.com/DSpace/DSpace/pull/12>
- Tim points at Business Logic / REST API really needing some stabilizing & support if we really want to support others building new UIs
- Tim wonders out-loud if we can have rearchitecture & new feature work going on simultaneously if we can all agree on a "middle layer" (likely REST API or Java Business Logic). I.e. if we decide that everything needs to "bubble up" via REST API, then rearchitecting could happen under that (as long as it doesn't break other main UIs)
- MarkW - we can "break stuff" in terms of API compatibility, if we want in 3.0. But, we just don't want it to be difficult to upgrade to.
- Some discussion in the end on how we want to use GitHub & Pull Requests. Tim mentions a thread on dspace-devel: <http://www.mail-archive.com/dspace-devel@lists.sourceforge.net/msg08702.html>
- Questions on our new GitHub workflow.
 - Still want to be tracking things via JIRA (as JIRA is our "history" of what changed in DSpace in each version).
 - We may need to play around more with Pull Requests and see what "works for us". Worst that can happen is that we need to roll back mistakes in GitHub
 - **TIM TODO:** Also want to ask Fedora/Hydra folks how they manage JIRA + GitHub. What does their workflow look like, and are GitHub changes/commits/pull-requests auto tracked in JIRA?

Tim's 3.0 Discussion Summary:

1. There are some concerns about the long-term 'viability' of Cocoon (which XMLUI is based on). It has served us well, but it doesn't seem to be as supported as it once was. This makes us all wonder if it's time to encourage new UI development or if we want to take a more active role in Cocoon's future. (NOTE: Currently nothing specific is in the works, though we do have several recent UI projects in: [WebMVC](#) and [SkylightUI](#).)
2. Sounds like most of us prefer the "gradual evolution" model that we've had with DSpace, as it lets us stick to our release schedules easier. So, 3.0 should be a gradual evolution from 1.8, just like 1.8 was a gradual evolution from 1.7.
 - That being said, we can and should make DB changes or API changes as needed to support that evolution
3. Sounds like none of us are against others starting to do re-architecture work with DSpace. However, there are obviously concerns that the work needs to be done openly so that we get buy-in from all. This work could happen parallel to 3.0.
4. We still need more discussions on what exactly will be in 3.0.
 - These should happen via email lists (dspace-devel).
 - We should **all** start to add our brainstorms in there. All brainstorms welcome – no need to have a full-fledged proposal.

- Next week we can discuss this more via IRC

Action Items

Action items coming out of this meeting go here, if any.

- TODO - Update the [ReleaseCoordinating](#) page to better outline Release Processes
 - Add "Update this document with lessons learned" as the final task of any release.