DSpace Futures REST API Meeting Notes

Attendees:

- Virginia Tech (Zhiwu Xie, Keith Gilbertson)
- MIT (Richard Rodgers, Sean Thomas)
- Harvard (Reinhard Engels, Bill McKinney)
- @mire (Bram Luyten)
- Georgetown (Terry Brady)
- U of Cincinnati (Linda Newman)
- DuraSpace (Tim Donohue, Jonathan Markow, Valorie Hollister)
- Ohio State U (Maureen Walsh, Melanie)
- U of Manchester
- U of Delaware (Mark Grabowski)

Tim's Overview

- Overview of DSpace Futures
  - In Oct / Nov, we had a series of meetings asking sponsors, service providers, Committers & DCAT members for their feedback on DSpace & ideas on how the platform could improve
  - Out of those meetings, we published a public report of the discussions (in January). To DuraSpace, it seemed like there were three topics that kept coming up as being of general interest. So, these next meetings (including this one) are to take the next step and open these discussions to the public.
  - REST API
  - Hydra + DSpace
  - Metadata Improvements

The PURPOSE of each of these meetings is NOT a deep technical dive, or to solve each problem in one meeting. Rather, these are "coordination calls". The goal of each call is to talk through how we can work to get these initiatives "kickstarted". So, this is an opportunity to talk through needs / motivations, but also to try and organize our common resources and perhaps even organize a project team (if we have enough resources available).

REST API interests / background.

- From the discussions, the main interest around a REST API was to provide a "Repository Abstraction Layer" for DSpace
- There were many reasons why people felt this could be important:
  - May allow for development of non-Java based, agile UIs (perhaps based on rapid development frameworks like Ruby on Rails or PHP or whatever). May even allow for a "connection" point to a Hydra interface
  - May allow DSpace to start to become even more "modular". Third party products/services could hook into this layer as a "connection point" to DSpace. Less need to code everything in Java or try to get DSpace to "do it all" out-of-the-box.
- As we noted in the email, there are currently two (known) stable implementations of a REST API out there (in GitHub)
  - One by Hedtek: [https://github.com/hedtek/dspace-rest](https://github.com/hedtek/dspace-rest)
  - One by Wijiti: [https://github.com/wijiti/dspace-rest-api](https://github.com/wijiti/dspace-rest-api)
- The gap that we now face is one of "lack of resources". Either of these APIs could be reasonable to adopt. It may even be reasonable to adopt both (or a combo): one Read-only and one Writeable (requiring authentication).
- We'd like to understand needs, motivation, what specific interests are, identify resources available to join project. Is this still a high priority effort?

Use Cases / Discussion

- Univ of Delaware - Mark Grabowski
  - Use case: tie resources in DSpace to other interfaces, need to reuse/repurpose information into other services. Currently having to "scrape" DSpace for info
- Reinhard - Harvard
  - Use case: creating Javascript widgets that auto updates researcher pages, have simple ones in place, but think there could be more done to help market the repo
  - Use case: want to create analysis/statistical tools that can suck out info from DSpace easily. They'd rather have an API, then needing to pull out data via SQL or similar
- Bill McKinney - Harvard
  - Currently playing with Wijiti API. Dipping their toes in the water and trying it out. Using for a Javascript widget.
  - One technical comment: They want the API to use Item Handles. Currently Wijiti only uses Item IDs. Harvard has customized it to allow for references via Item Handles.
- Manchester
  - creating external user interface based on a Ruby on Rails projects - we need to provide accurate usage stats for views and downloads - we want to be able to report in DSpace at the end of the day
- Zhiwu - Virginia Tech
  - Use case: want to layer extra UI on top of DSpace
  - willing to contribute resources for development
  - Would anticipate heavy loads on a REST API, so it needs to be able to scale well (may need some caching?)
- Linda Newman - Univ of Cincinnati
  - All above use case ideas sound intersteing
  - Use case: allow content contributors to embed content on their own website
  - RSS feed doesn't allow them to embed content in their website
  - Use case: could open up ways to leverage content and allow for more sexy UIs
  - can same REST API work for both XML and JSP?: per Tim, yes, REST API would talk to underlying infrastructure (e.g. database) and not be reliant on current UIs

Overview of Managed Projects & examples from Fedora Futures (Jonathan Markow)
How we’ve organized projects for Fedora Futures projects - Jonathan
work on projects comes from community volunteers
several from Fedora community identified the need for a sustained effort to move Fedora forward
large projects for DSpace might also be approached this way (lots of new contributors and new features) - more focused projects, more focused goals - get commitments from several institutions for dev time, at least 1/2 an FTE for an extended period of time, plus a tech lead and a project manager
Tim: getting resources on larger projects are challenging - want to see if we can get something going for a REST API - maybe even have something in the next release
Jonathan: joint work on API will be more useful to the community than just one institution contributing it -
Tim: good way to get involved, be a leader and have a bit more control over what is in DSpace
Jonathan: not proposing a separate project - work must be done in consultation with the current committers - process must be open - not override any agreements about who is able to commit code to DSpace - must follow standard procedures

More Discussion / Use Cases

- Reinhard - Harvard
  - from a dev perspective we’d love to work on things together - sharing our things and working with others
- Terry Brady - Georgetown
  - how do building APIs alter the plan for existing DSpace UIs?
  - Tim: not going to effect track of DSpace - REST API would be a separate interface, like SWORD interface. Current UIs (JSP, XML) would still exist, a new interface might become another option through this work, we will not be abandoning JSP/XML UI, might decide releasing UI updates separately at some point (though it's hard to speculate)
- Zhiwu - Virginia Tech
  - Existing RSS Feed functionality and other interfaces (even OAI-PMH or similar) could be rewritten eventually to use REST API
    - Tim: Yes, that could be something we'd think about in the future. Maybe not initially, but it's a definite possibility.
- Reinhard - Harvard
  - Does Fedora have a REST API - is it something to "inspire us".
    - Tim: yes, that's what Hydra uses to communicate with Fedora
    - Jonathan - API is a little too full featured, a lot of stuff that never gets exercised, trying to figure out what they should include in the Fedora rework - might start to deprecate some things - might be worth talking to Fedora folks
    - Fedora may have encountered similar issues (in performance, etc). Jonathan said yea, performance is a main concern right now with Fedora Futures.
  - Fedora might have suggestions about what Java libraries, handles, etc. we should use
  - Tim: should mtn communication with Fedora - might be common problems to solve in common ways
- Linda - Cincinnati
  - Another use case - grapple with managing data - one of possible uses would be APIs that do data visualization or data mining - so the repository can be queried
- Zhiwu - Virginia Tech
  - trying to layer data from LOC - would be a good use of REST API for DSpace
- Richard - MIT
  - might be good to have place to gather some of these use cases - especially for those who can't participate in a joint project - should set aside wiki space
  - Tim: will publish notes and use cases on wiki - invite others to add
- Bram - @mire
  - Use Case: integration with other web platforms - see clear convergence to a few major platforms like Drupal - are there specific systems that it would be important to intergrate with?
  - Several folks said they are interested in seeing DSpace + Drupal integrations via a REST API
    - Reinhard - Harvard
    - Zhiwu - Virginia Tech
    - Perhaps this is a good initial use case / test case for a REST API?

Next steps

- Individuals / institutions should express their interest in being a part of a DSpace REST API project team
- Email Tim, Val or Jonathan
- Next step is to organize effort and to figure out what combined resources are available
- DuraSpace's role will be "advisory". We can help coordinate calls, but we need institutions / individuals to do the "on the ground" work to get this REST API implemented.