

2018-09-25 - VIVO Development IG

Date

25 Sep 2018

Call-in Information

Time: 11:00 am, Eastern Time (New York, GMT-04:00)

To join the online meeting:

- Go to: <https://duraspace.zoom.us/j/823948749>
- Or iPhone one-tap :
 - US: +14086380968,,823948749# or +16468769923,,823948749#
- Or Telephone:
 - Dial(for higher quality, dial a number based on your current location):
 - US: +1 408 638 0968 or +1 646 876 9923 or +1 669 900 6833
 - Meeting ID: 823 948 749
- International numbers available: https://duraspace.zoom.us/join?m=Qy8de-kt6W4fMMDQCAV_3qfH1W-lxAo5

Slack

- <https://vivo-project.slack.com>
 - Self-register at: <https://goo.gl/forms/JxQFkut4TYj4Ehww1>

Development Process

- See <https://github.com/vivo-project/VIVO/wiki/Development-Processes>

Attendees

[blocked URL](#) Indicating note-taker

1. [Andrew Woods](#)
2. [Graham Triggs](#)
3. [Mike Conlon](#)
4. [Kitio Fofack](#)
5. [Jim Blake](#)
6. [Tim Worrall](#)
7. [Huda Khan](#) [blocked URL](#)
8. [Don Elsborg](#)
9. [Brian Lowe](#)

Agenda

1. [Sprint update](#)
 - a. i18n
 - b. Externalized Search
 - c. ABox / TBox
2. 1.10 bugs (1.10.1?)

type	key	summary	assignee	reporter	priority	status	resolution	created	updated	due
------	-----	---------	----------	----------	----------	--------	------------	---------	---------	-----

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

3. Demo and walk-through of:
4. Recent mailing list topics:
 - a. [Add Handle Support in vivo?](#)
 - b. [Problems with Startup of VIVO 1.9](#)
 - c. [FreeMarker template error: When calling macro "getItemType", required parameter "type"](#)

- d. [Removal of schema.org in individual--foaf-person.ftl](#)
- e. [Adding External Vocabularies](#)
- 5. Active ticket
- a.

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

 (Kitio Fofack - where does

Notes

[Draft notes in Google-Doc](#)

• Advanced Role Management demo (Graham Triggs)

- [Recording.mp4](#)
 - Using discrete permissions for properties: set of checkboxes per set of permissions:
 - Display, update, publish: with options to select any of: site admin, curator, editor, self editor, and public
 - Display, update, publish: key permissions tied to roles earlier
 - Original configuration had permissions (such as different types of management) but did not cover property-specific display/update/publish permissions
 - Permission sets defined for display, update, publish, and then these are tied to particular classes, e.g.
 - displayPermissions:Public (where displayPermissions is a prefix) has predicate 'auth:forEntity' with several URIs as objects including both properties (data and object) and classes
 - Mike: VIVO is positive permissioning system, i.e. if permission is not declared, you cannot do anything
 - Graham: this permission scheme is additive. Before, we defined restrictions and if no restriction defined, it was available. In this scheme, if permission set does not exist, you do not have permissions, and you can add to what permissions are available
 - Before, (some of the?) permissions info was saved in display model, but in this scheme, these permission definitions are now all saved in the accounts graph. Permission implementation now completely orthogonal to the rest of the application.
 - Don: reason that permission_entities file in auth/firsttime instead of everytime?
 - Put it in firsttime since being loaded into graph in config triplestore, which means when you edit the data, the edits are stored back in the graph. When changes are made, they are persistent and the original is not reloaded when you restart the application.
 - Could technically have the files in everytime and maintain your changes in the filegraph, but this is a wider discussion than just permissions.
 - Mike: graph management issues related to this work.
 - Auth migrator Java class in pull request
 - If upgrading VIVO, will take permissions already configured for existing installation and convert to equivalent in advanced role management scheme
 - Mike: these authorizations are functional. Quite beautiful.
 - Can this be used to setup editor for history department so they can only edit the things that belong to the history department?
 - Graham: not really something that could be implemented by this scheme
 - Mike: Because a connected graph and there is no history department (i.e. as a separate entity)
 - Graham: but equivalence to role of self-editor. As self-editor, can edit my own information including information that has been declared as connected to me. Already part of existing release.
 - Self-editor policy which uses self-editor relationship, and checks if this entity is related to the self-editor.
 - Permissions and policies play into each other.
 - Jim: some good decisions here.
 - Question about self-editing: policy = if you can tell me what it means to have data related to you (by specifically saying that you are the author or PI or some other relationship), the relationship can be put into policy.
 - Mike: Could thus define what it means to be a department editor (analogous to the self-editor policy)
 - Jim: if you add a new property, who is authorized to say who can set permissions for that property? Only root account?
 - Graham: in theory, possible by anyone who can edit the ontology. If you can get into the ontology editor, you can set permissions using the editor.
 - Graham: Haven't gone through existing ontology files (annotation files) to remove the old implementation and that will need to be removed. May also need to update the permission entities configuration. This is the right point though to share this code and have people see what it is doing.
 - Andrew: the process for creating roles?
 - In permission_config file, pick existing permission set (e.g. auth:Editor) and copy over and then change permissions
 - E.g. auth:Pub_editor with auth:hasPermission displayPermission:pub_editor and updatePermission:pub_editor and publishPermission:pub_editor
 - Also create permission set individuals displayPermission:pub_editor, updatePermission:pub_editor, and publishPermission:pub_editor
 - Jim: If you were to then re-maven install and restart tomcat, you would see pub editor as a possibility for property permissions
 - Graham: should work
 - Permission entities in VIVO level, and permission config in Vitro level
 - Andrew: is there anything that ensures that the permission configuration stays in synch with the ontology?
 - Mike: Not right now but could perhaps have some validation
 - Jim: sounds like a smoke test
 - Mike: could check if entity references in permission entities does exist in the ontology
 - Andrew: TO DO: will create ticket around validation/smoke test
- Andrew: Migration tool. Would be useful to have a quick review.
 - Graham: startup listeners: includes block of migration classes
 - Added AuthMigrator class
 - Looks up old roles and maps to new permission scheme
 - Would require more work to make this more general but might be quite complex
 - Andrew: large pull request
 - Question: do you have thoughts about how to expedite getting this in? The larger the pull request, the more possibility for merge conflicts
 - Helpful to get it in relatively soon

- Dedicated day where people could work on it together?
- Ralph: Would be happy to do it in group after this week (and/or after next week)
- Jim: appropriate method. Approach would have been just by inspecting code but idea of getting trial use case, hitting as many corner cases, and seeing how code responds - sounds good.
- Mike: Documentation hat on. Fair amount here is new. Immediately went to idea that there might be design doc that describes how permissions and policies actually work. What could you do with them from a design point of view. Which would lead to admin document (how to define permissions) and developer document (e.g. if you want to define a new policy, you would need to do the following development things). Lot to work on.
- Jim: documentation is a big task but important to clarify what functionalities are in VIVO.
- Andrew: an abbreviated sprint (e.g. a week)
 - If we could get documentation in place, we could test that documentation in the process of implementing use cases and try out functionality
 - (there was this whole bit about chickens and eggs but I doubt I could have captured it well...)
- Graham: 2 pull requests: in Vitro and VIVO. had to re-open VIVO one.
 - Mike: require 1.10 fresh install (develop 1.10), and then try out the pull requests?
 - Graham: yes
- Next steps: documentation, review

Previous Actions

- [Alex Viggio](#) will bring news of Elasticsearch instead of Solr up with Product Evolution. Might there be consequences for the September sprint.