

2020-10-13 - VIVO Development IG

Date

13 Oct 2020

Call-in Information

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

To join the online meeting:

- Go to: <https://lyrasis.zoom.us/my/vivo1>
- One tap mobile:
 - US: +16699006833,,9358074182# or +19292056099,,9358074182#
- Or Telephone:
 - US: +1 669 900 6833 or +1 929 205 6099 or 877 853 5257
 - Meeting ID: 935 807 4182
- International numbers available: <https://zoom.us/j/aeANHanzED>

Slack

- <https://vivo-project.slack.com>
 - Self-register at: <http://bit.ly/vivo-slack>

Attendees

★ Indicating note-taker

1. [William Welling](#)
2. [Andrew Woods](#)
3. [Alexander \(Sacha\) Jerabek](#)
4. [Benjamin Gross](#)
5. [Huda Khan](#) ★
6. [Ralph O'Flinn](#)
7. [Brian Lowe](#)
8. [Nicolas Dickner](#)
9. [Michel Héon](#)

Agenda

1. i18n - Ne
 - a.
 - b.
 - c.
 - d.
 - e. Unable to locate Jira server for this macro. It may be due to Application Link configuration.
2. Solr conf
 - a. Unable to locate Jira server for this macro. It may be due to Application Link configuration.
3. Moving [Scholars closer to core](#) - [next steps](#)
 - a. [SelectQueryDocumentModifier](#)
 - b. Entities: Collection, Concept, Document, Organization, Person, Process, and Relationship
 - c. Configuring Solr
4. Moving [Data Ingest Task Force](#) forward

Future topics

1. [Vitro JMS messaging approaches](#) - redux
 - a. Which architectural pattern should we take?
 - b. What should the body of the messages be?
2. [Renaming of 'master' branch?](#) ([ZDNet](#), [BBC](#))

- a. [Guidance](#) from GitHub
 - b. DSpace has done it
 - c. Fedora has done it
 - d. Samvera is doing it
3. Incremental
- a.
 - b. Unable to locate Jira server for this macro. It may be due to Application Link configuration.
 - c.
 - i. Unable to locate Jira server for this macro. It may be due to Application Link configuration.

Tickets

1. Status of In-Review tickets

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.										

Notes

[Draft notes in Google-Doc](#)

- Style guide discussion
 - Haven't turned on many of these rules yet.
 - Impression of any check styles? Concerns around code quality?
 - William: Code style checks include rules on code complexity. Good to limit number of conditionals/nesting/number of arguments in functions
 - Biggest challenge: tracking which encapsulated classes are being called from which functions
 - Andrew: Significant step to refactor code
 - Set on runtime
 - William: Dependency injection would take care of it
 - Make sure no other class has an explicit understanding of another class and invokes things by that knowledge
 - Brian: As a priority, eliminating bad practices such as setting static variables in one method and calling from another.
 - Thinking about old code we could eliminate. Layers of objects - beans - objects returned by daos.
 - Parallel way of accessing data that is also accessed via direct SPARQL in other places
 - Would be good to not maintain both of these approaches. One consistent way of getting at that data
 - Andrew: Challenge with refactoring (especially updates that would be removing code): we don't necessarily know what the effects are because we don't have the tests in place to ensure that we haven't broken things
 - Building comprehensive test suite from the ground up
 - Need to transition from conversation to action (not to quote Elvis)
 - Challenge may be the legacy nature of the code base
 - Andrew: New project based on subset of what VIVO does - a lightweight project. Need to identify which aspects of VIVO we would want to target for a ground up application because putting in new VIVO in place of existing VIVO - VIVO as it exists does a lot and a new lightweight version would not do all the things the old one did
 - William: Sets of features etc.?
 - Andrew: Spreadsheets exist
 - Brian: In early days of NIH project, the point was made that RDF is the API. In retrospect, RDF/linked data crawling not the way front-end development proceeds. Need a layer in between to translate between the two that has a nice well-designed API. Code development done in the context of NIH grant - operating under a different philosophy.
 - Andrew: Would the VIVO scholars approach be an appropriate way forward? Taking a slice of VIVO functionality (profiles) and starting afresh
 - Brian: General idea of having a fresh application that is focusing on a subset of functionality: good but insufficient. If only profile pages result, why bother creating semantic data? Could just show pages displaying lists.
 - To justify semantic graph underlying the application, there has to be a way to take advantage of the graph.
 - The main application has a few pre-packaged ways: co-author diagrams, etc.
 - Either we decide we see value in using a graph that goes beyond display of information which we could do using other means, or ...
 - Michel: The feature not used enough is the VIVO ontology itself. Much work has been done to develop the VIVO ontology and because this ontology can be reused outside the software. A real vocabulary of education and researcher profiles. In terms of linked open data, value for exposing the ontology and making a community that can work with this ontology.
 - In Europe, using VIVO ontology but have to add more concepts/classes to be used for the French institutions. Same issue for Quebec/Canadian institutions.
 - More flexibility of making change inside the ontology and reflect the change on the visual part of VIVO
 - Automatic data extraction - auto-extracting who are the authors, what are the abstracts, etc. Extracting the ontology on this document where we can have actual data set of VIVO

- Michel: The big nice feature of graph: interconnection. If using a graph inside one instance of VIVO, it's just another database. The real power is interconnecting graphs.
 - Unique identifier in one instance of VIVO. Just to manage identifiers. Another instance that is representing what is happening inside a local organization. Query across multiple instances - where each instance is dedicated to a specific task.
- William (from chat)
 - <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/GSPARQL.pdf>
 - <https://www.w3.org/Data/events/data-ws-2019/assets/position/Ruben%20Taelman.pdf>
- Huda said some stuff
 - Linked data application curse:
 - Practical limitations of open-source/free technologies
 - Knowledge graph benefits: integrating across heterogeneous data sources and providing the ability to reason across/"AI magic" the resulting data
 - We haven't progressed very far on either of the benefit sides yet - part of this may just be the amount of work it takes to just set up a semantic system that enables editing and display because plugging in real-time/web systems requires more scalable/faster infrastructure than normally available in the "free" stack
- Andrew: Documenting the clarity of the vision. What does new/ideal VIVO look like? On the other hand, understanding why people use VIVO right now and like it?
 - Incrementally making the application better in support of people who like what they have
- Brian: As Michel said, interconnected systems really would be a plus
 - Identifiers that could be reused and linked
 - Instead of relying on systems like scopus etc./large commercial data vendors, would have the ability to use and reuse the data we model and make available
 - Those use cases seem to have disappeared but may be interesting to revisit
- Andrew: Ideal vivo may have less to do with the app itself but more about realizing the power of the VIVO ontology
 - Maybe talking about building something new based on what we currently know

Actions

-