Implementation and Development Call 20140424

Calls are held every Thursday at 1 pm eastern time – convert to your time at http://www.thetimezoneconverter.com

Announcements

- **News Release: Layne Johnson is the New VIVO Project Director**
- Ontology Working Group: next call is Thursday, May 1 at noon EDT/ 9 am PDT
    - addressing persistent VIVO URIs and related issues
    - today's call was focused on datasets and dataset ontologies, including options to engage with a W3C working group active in that area
        - The Health Care Life Sciences working group new draft standard for dataset description
        - We're trying to think how VIVO can contribute
            - Submitting use cases on how we represent datasets in VIVO
                - Patrick West: DCAT ontology is what we're using at RPI for the Deep Carbon project. Also have our own DCO ontology that we'd be happy to share. Integrating with CKAN for data flow and management.
            - Making people aware that VIVOs will have extensive contextual data around datasets
        - Also discussion of how VIVO relates to ORCID iDs and the distinctions between an ORCID iD, a VIVO URI, and the set of statements about a person in a VIVO
- Apps & Tools Working Group: next call is April 29 at 1 pm EDT
    - Find links to YouTube videos of all previous calls there
    - Alexandre Rademaker (IBM Brazil) and getting specialized data into RDF and then into VIVO
    - Apps and Tools workshop at the conference. Looking for participants to do demos -- looking for the best ways to create, use, visualize VIVO data and would love to have additional authors and help
- VIVO Bootcamp at ELAG 2014 (June 10 at University of Bath, United Kingdom)
    - Violeta Ilik from Texas A&M will be there
    - the organizers (including Violeta) are looking for another representative from the VIVO community

Upcoming Activities

- First annual survey of VIVO sites:
    - Status: 3 surveys have been completed
    - See Paul's email to the lists: What does your site care about? Participate in the 2014 VIVO Annual Survey
    - The survey form is live: https://www.formstack.com/forms/?1704676-gIMWlsYzom
    - A presentation proposal has been submitted to the conference and in order to have time to compile and analyze the data, responses are requested by May 29
- Next themed weekly call topic – May 8th: VIVO End User Documentation – **Help files or documentation for VIVO editors**
    - for supporting end users who will be editing in VIVO, whether individually or as proxy editors for a whole department (requested by University of Florida) -- can we make templated documentation where individual sites can put in tips or extra information? Rumor has it there's a good document underway -- can this be shared on GitHub using GitHub pages? (Alex: GitHub being used by non-coders for local political campaigns and other collaborative writing projects)
    - Format of themed calls could be a mix of presentations and working
    - Note that we are seeking volunteer facilitators for each themed call -- let Alex, Paul, or Jon know directly or via the listserv(s)

Site Updates

- Site updates during next week's call

Theme: Performance Part 1

- **Follow ongoing activity at Additional Performance Tips**
- Attendees: Brown, Cornell, Duke, EPA, IBM/FGV, Scripps, Smithsonian, Symplectic, Weill-Cornell
- Part 1 of a discussion around VIVO Implementation Performance concerns: concepts, common performance concerns, VIVO components related to performance, benchmarking (will merge Paul and Alex's Google doc into shared call doc)
    - trying to address deficiencies in existing documentation while trying to get more knowledge out there
    - a lot of knowledge is out there but is likely fragmented -- one person may know Tomcat configuration or Java garbage collection options while someone at another site may have expertise in MySQL configuration
    - we also need to have a better collective understanding of benchmarking performance, as well as of how to instrument a VIVO and its support software (Apache, Tomcat, MySQL) to detect performance problems
- Existing wiki docs related to performance:
    - https://wiki.duraspace.org/display/VIVO/MySQL+configuration,+tuning,+and+troubleshooting
    - https://wiki.duraspace.org/display/VIVO/Use+HTTP+caching+to+improve+performance
- Show of hands (encouraged group discussion at start of call): has your site struggled with performance? In what regard?
    - Alex R: Data ingest may have different issues -- if we have benchmarks for a specific use case such as ingesting 500 publications
        - was not sure whether the time it took him was the expected time, and that's why benchmarks would be helpful
        - would like a benchmark on how long it takes to ingest a certain number of n-triples (vs. RDF/XML), say per 100 *new* triples
            - Jim & Jon: this can depend on how many triples are already in the data model -- if you already have 1M triples in the store, time to ingest n new triples may be longer than for an empty store
        - should we have a canonical dataset with 100 people, 500 publications, 250 grants, etc. that could be used for testing a successful installation

                - could set it up on your machine and compare performance behavior against test results with the same data on other people's installations
                - how big would the test dataset have to be to provide meaningful answers?
                    - some processes like re-inferencing seem to get slow
                - some data exists in the VIVO sourceforge site that could be used to begin a canonical data set: http://iweb.dl.sourceforge.net/project/vivo/Data%20Ingest/
            - it would also be helpful to have a benchmark Amazon image that could be used for testing comparisons
                - as well as to develop a recommendation on minimum hardware performance requirements
                - UF is the only one we know is running a production VIVO on AWS, but using the Amazon persistence solution instead of MySQL?
            - Paul: do sites notice that performance degrades linearly as you add VIVO data?
                - Jon: work in small chunks when doing data ingest, e.g. to insert 40k triples, insert them 5k triples at a time with some delay (30 seconds) in between batches
                - Jon: ideally we will upgrade JENA libraries to latest for VIVO 1.7 -- this will require removing dependence on JENA RDB
                - AlexR: how close to VIVO code being completely decoupled from JENA, e.g. any gaps would be seen as a bug and not a feature/enhancement request? Jon: started with VIVO 1.5, but not clear how extensively this has been tested by VIVO implementations. VIVO 1.5. has gone with an RDF API so in theory VIVO could work with any triple store that works with standard SPARQL commands. Jim: yes but our application-specific data is tied a little more closely to Jena
                    - so you'd have to still run Jena RDB and MySQL as well as the triple store for your primary content, until VIVO 1.7
            - Ted: a test data set would be very helpful, and would be interested in assembling this (see link shared above). Jon: this would also be very helpful for refactoring/re-engineering search functionality in VIVO -- to allow for change while reducing risk of breaking existing results.
            - Patrick: performance of large profile pages, e.g. over 1,500 publications on a Duke faculty page taking over a minute on average to respond
                - Jon: at some point you have to ask yourself, where else on the web would you want to store that much information?
                - Patrick: yes, I think lazy loading might make sense
                - Jon: the Sakai project concluded that if you're really serious about performance and want sub-second load times, you would have to use caching; NYPL never runs any pages from Oracle directly -- how much of 1.6 caching work would support this?
    - AlexV: performance can mean different things to different sites
        - how long it takes for a visualization to render
        - how long it takes for a search to return through the web interface
        - how long it takes a SPARQL query to return
        - how long it takes a large page to render (e.g., a person with 800 - 1500 publications)
        - how long it takes to regenerate the search index or to re-inference the whole of a VIVO
        - how long it takes to generate an export of RDF data (or whether it ever completes)
        - how long it takes to generate a CV using the Digital Vita Docs or Mike Conlon's new CV/biosketch tool
- What are the signs sites have noticed that a VIVO instance is performing poorly? (see notes above)
    - Reinferencing takes several days
    - Large profile take more than 10 seconds to load
        - Should we refactor the code to do more lazy loading -- who really needs to see a list of 800 publications unless specifically requested?
    - Harvester is slow to execute
- Most common causes for performance issues? (maybe an opportunity for Jon/Jim sharing stories?)
- VIVO application and web/persistence stack components with performance consideration
    - Apache itself - I would be surprised if this were the problem but that doesn't mean there can be some issues such as the number of processes. The issue is comparing this to the number of processes Tomcat uses. We've seen this turn ugly if Tomcat is seeing fewer connections...
    - MySQL
- What affects performance - size of db, number of triples?
- Internal: insufficient resources like memory, MySQL configuration (cache size)
- External: bots not minding robots.txt
    - What software upgrades (or software libraries) have a material effect on performance?
    - New Jena libraries likely have improvements to SDB but RDB has been phased out so VIVO 1.7 has to include effort to migrate user accounts to TDB
    - older vs newer version of Apache or an Apache library (and how this is affected by your version of Linux)
    - Tomcat, Solr, MySQL, OS (Linux, Windows), Java coding performance idioms vs JVM configuration
- Benchmarking (and Monitoring?) tools
    - Difference between performance benchmarking and performance monitoring
    - How much is slow performance related to load vs. slow code or large amounts of data -- does your VIVO degrade significantly under load or is it more or less the same speed regardless of load (within normal bounds, at least)?
        - If even tiny pages are slow (multiple seconds), there may be a more systemic problem
    - How much is performance related to editing?
    - Ted uses jMeter, which is appropriate for load testing
- Where do you look to see where there might be a performance problem?
    - Load on front end (Apache) versus mid-tier Tomcat/VIVO app versus Solr and database?
    - Apache Benchmark or other tools? Generally Apache is the least likely culprit, but can't rule it out
        - the number of processes in comparison to the number of processes (simultaneous requests) vs. that Tomcat is permitting
            - the Apache connector to Tomcat should be looked at, including the number of threads available to both Apache and Tomcat (uLimit) -- the number should increase, not decrease, as you go down the stack so that the lower tier doesn't become a bottleneck
    - Solr monitoring https://wiki.apache.org/solr/SolrMonitoring
    - http://stackoverflow.com/questions/1754427/jmeter-alternative
        - A load testing option
        - JMeter is a tool for testing and benchmarking websites
        - can create a test plan and adjust the number of simultaneous users to simulate being hit with a large load
        - let Ted know if you want to share ideas or test plans

        - Paul -- sees patterns of progressive worsening -- sinusoidal patterns
            - sometimes it goes down, but usually not at times you would think it has peak load
            - Richard - can be hard to see... the Duke search index re-runs every night, and backups are going on
                - A search re-index can take 5-6 hours at night but completed in 90 minutes at 8 am
                - Duke has two load-balanced servers running VIVO with a 3rd server for database, solr, and administrative editing functions -- has diagrams, but thinks it might be on wiki already?
- Quick and easy way to monitor performance?
    - Google Analytics will record some basic performance in the Behavior > Site Speed section
        - very easy to get it turned on when you set up your VIVO
        - might be useful across multiple VIVOs as an easy way to get initial comparisons
        - shows average page load times -- 8 seconds for Weill Cornell when look by week or month
        - Alex -- can you drill in further to see more than the average value?
- Benchmarking against other VIVOs?
- New Relic performance tracking on the Java VM
    - Harry (Weill Cornell): in recent testing had about 2000 connections calling the common pool data source -- servlet constantly making new connections for each query -- is that normal?
        - Jim - no, it should be getting connections from the pool and releasing them once it's finished
        - Harry - were there changes in that from 1.5 to 1.6?
        - Jon -- 2,000 connections seems unhealthy
        - Harry -- Just came back from New Relic training
    - looking at the IndividualController -- we know that it is slow -- making all the queries is normal, but within it, each query to the quad is getting a new connection --
        - is it creating one or grabbing one out of the pool? which issues the request for the new connection?
        - Brian Lowe probably has the most intimate knowledge of what's happening with connections at that level -- whether they would ever time out
    - Alex -- how is New Relic to use? looks like it supports Java, Python, Ruby
        - Harry -- hook in the JAR for New Relic when you start up Tomcat
        - monitors the processes
        - there's a plugin to connect to MySQL as well
            - can see the queries being launched and how they're created
- Any differences with type of instance: cloud vs. dedicated vs. virtual
    - e.g., separation of Tomcat and MySQL on different servers
- What difference would the use of alternative triple stores make?
    - Sesame
    - Virtuoso
    - Allegrograph (note: not free at scale)
    - OWLIM (has unusual support for reasoning, but also a licensing fee)
    - others (4Store, MarkLogic)
- Future performance topics (calls #2 or #3):
    - How to troubleshoot performance issues
- Throttling crawler/bot access (Florida, Indiana)
- Load balancing / Multi-headed Tomcat (Ted)
- Caching in 1.6+ (could be a separate call)
- MySQL tuning (again, could be a separate call)
- Inefficient in-page queries. What are some examples of slow vs. fast in-page queries? (Jim, Brian, Tim?)
- Which alternatives to existing libraries (Apache, Solr, Jena, Tomcat, etc.) are most worth exploring?
- What role does Apache's mod_mem_cache play?
- JavaScript optimization
- Drastic approaches for improving performance: new triple store, using Solr to store more information…

Notable list traffic

- **Project displaying issue on person's page**
- **ingest problem: things disappear after, recompute**
- **sometimes when you log in you get a bunch of status warnings**

See the vivo-dev-all archive and vivo-imp-issues archive for complete email threads

Call-in Information

- Date: Every Thursday, no end date
- Time: 1:00 pm, Eastern Daylight Time (New York, GMT-04:00)
- Meeting Number: 641 825 891

To join the online meeting

- Go to https://cornell.webex.com/cornell/e.php?AT=WMI&EventID=167096322&RT=MiM2
- If requested, enter your name and email address.
- Click "Join".

1. Call in to the meeting:

1-855-244-8681 (Call-in toll-free number (US/Canada))

1-650-479-3207 (Call-in toll number (US/Canada))

2. Enter the access code:

641 825 891 #

3. Enter your Attendee ID:

8173 #