2016-06-24 Performance - Scale meeting

Time/Place

- Time: 2:00pm Eastern Time US (UTC-5)
- Dial-in Number: (712) 775-7035
    - Participant Code: 479307#
    - International numbers: Conference Call Information
    - Web Access: https://www.freeconferencecallhd.com/wp-content/themes/responsive/flashphone/flash-phone.php

Attendees

- Nick Ruest
- Andrew Woods
- David Wilcox ⭐
- Esme Cowles

Agenda

1. Performance and scale hackathon
2. ModeShape 5 issue
3. JMeter tests with MySQL
4. Test results
5. Request on root resource with a million children fails to respond

Minutes

Hackathon

- University of Michigan can fund a hack house for performance and scale work
- Can hopefully schedule for this Fall, but the calendar is getting close
- Maybe in Ann Arbor?
    - Nick can probably go
    - Esme’s calendar is already pretty full

ModeShape 5 issue

- https://jira.duraspace.org/browse/FCREPO-2060
- Related to PostgreSQL 9.3
    - Not seeing an issue with later versions
    - No issues with MySQL
    - Probably should just recommend using PostgreSQL 9.4+
        - Esme will add this to: https://wiki.duraspace.org/display/FEDORA4x/Configuring+JDBC+Object+Store
        - Will also provide some detailed example commands for PostgreSQL and MySQL since we are pushing people in that direction
- Ingest seems to be about 15% faster with Mode5 compared with Mode4

JMeter tests with MySQL

- https://gist.github.com/ruebot/413ca1552ed2690654d3b1db7b347a0d
- Seems to be a null-pointer exception somewhere
- Tested with Mode4
- Nick will test with Mode5

Test results

- Esme has done test #4 thoroughly
    - Will do test #2 with Mode5
    - Has not done test #1 due to lack of disk space
    - Currently we only have tests for #1-4
- Andrew will write test #5 and Esme will provide the file
- Esme will run test #3 until running out of disk space
    - PostgreSQL and MySQL
- Need to do some analysis of the test results
    - Are these results adequate for the kinds of use cases people actually have?
    - Can we push Fedora to its absolute limit in terms of how many resources it can contain?

Request on root resource with a million children fails to respond

- Failing to get a count of all child resources
- We can probably work around this by disabling the count
- A better solution would be to store the state, but this is could be tricky
- Or we could stop counting after x children (e.g. there are 1000+ children)
    - We do something similar in the HTML UI - we only list the first 100 links
    - You can already do this in a REST request by adding a limit
    - Andrew will implement a fix
- Is there a real use case for knowing the number of children?