2016-08-05 - Import - Export Planning Meeting

Time/Place

- Time: 3:00pm Eastern Time US
- Dial-in Number: (712) 775-7035
  - Participant Code: 479307#
  - International numbers: Conference Call Information

Attendees

- Nick Ruest
- Joshua Westgard
- David Wilcox
- A. Soroka
- Michael Durbin
- Bethany Seeger
- Mark Jordan
- Justin Simpson
- Yinlin Chen
- Benjamin Armintor

Related

- Design - Import - Export

Agenda

1. Finish: Initial Requirements
   a. Two flavours of import/export effort; RDF and binaries vs BagIt Bags
2. Workplan and timelines
   a. Sprint scheduling
      i. August 8-19, 2016
   b. Developers
      i. Nick Ruest
      ii. Esmé Cowles
      iii. Michael Durbin (Maybe)
      iv. A. Soroka (Maybe)
   c. Testing and validation
      i. Michael Durbin
      ii. Youn Noh
      iii. Bethany Seeger (Maybe)
      iv. Nick Ruest
      v. Joshua Westgard
      vi. Yinlin Chen
      vii. Justin Simpson
   d. Documentation
      i. Youn Noh
      ii. Nick Ruest
      iii. Joshua Westgard

Minutes

Workplan and Timelines

- Sprint 1 changed to start August 29 due to people being unavailable for the 8th

  Developers
  - Nick
  - Esmé - Nick will follow up
  - Mike
  - Ben

  Testing/validation
  - Josh
  - Mike
  - Youn- Nick will follow up
  - Nick
  - Yinlin
Requirements

External systems

1. Support import from and export to a TBD list of external systems.
   a. Already discussed
   b. Add Archivematica - Justin
   c. Add LOCKSS - Mark J

General

1. Support transacting in Fedora-compliant RDF
   a. What do we mean by Fedora-compliant?
   b. Strike unless/until someone clarifies the meaning
2. Support allowing the option to include Binaries
   a. No issues
3. Support references from exported resources to other exported resources
   a. Important but challenging requirement
   b. Is this restricted to subtree resources?
      i. Need clarification on this point
4. Support transacting in BagIt bags
   a. No issues
5. Support import into a non-existing Fedora container
   a. Split into 2 requirements
      i. With same relative path
      ii. With a different relative path
   b. Merge 5, 6, and 7?
6. Support import into an existing, empty Fedora container
   a. Same issues as 5
7. Support import into an existing, non-empty Fedora container with various policies: add, overwrite, delete, version, skip
   a. Same issues as 5
8. Support export of resource versions
   a. No issues
9. Support import of resource versions
   a. No issues
10. Support export of resource and its "members" based on the ldp:contains predicate
    a. No issues
11. Support export of resource and its "members" based on a user-provided membership predicate
    a. No issues
12. Support recursive RDF insert/updates with LDP Indirect Container specified POST (and PUT / PATCH?) (ref: FCREPO-2042)
    a. Non-RESTful - impacting many resources by interacting with one
    b. Could be done client-side using transactions
    c. Makes more sense with PATCH than POST or PUT
    d. Can we put this aside for now pending further discussion?
    e. Not a priority for the first sprint
13. Consider import/export performance as is possible under the assumption that this work is done via the REST interface
    a. This is not a requirement, just a recommendation
    b. We would need numbers to work against in order to turn this into a requirement
    c. Needs more discussion

Round-Tripping

1. Support preservation of dates during round-tripping
   a. No issues
2. Support preservation of version snapshots during round-tripping
   a. No issues
   b. Will be challenging to implement
3. The URIs of the round-tripped resources must be the same as the original URIs
   a. No issues
   b. Note: within Fedora’s existing capabilities
4. Support lossless round-tripping. (ie, if you export a resource, delete that resource and import there is no difference from if you had never performed any of those operations).
   a. No issues
   b. No difference = no semantic difference
   c. There would still be an audit trail

Bag-It

1. The structure and scope of accepted and produced BagIt bags must be configurable (resource)
1. This is client-side - a script will pull the appropriate profile and assemble a bag
2. Unambiguously support linking between resources within a bag, and from resources in the bag to resources outside the bag
   a. Is there already an assumed requirement that bags contain multiple resources?
      i. We should start with single resource bags as an initial goal
      ii. This feature will not be very useful to some people if we cannot support multi-resource bags
   b. Needs more discussion at the beginning of the sprint (and during the sprint)

* Nick will send out a Doodle poll to nail down a call on the 29th for the beginning of the sprint