2011-01-13 - Fedora-DuraCloud Selective Backup Discussion

Attendees

Steve DiDomenico, Bill Parod, Julie Patton, Claire Stewart, Ben Armintor

Fedora Flexport Utility

http://cwilper.github.com/fcrepo-misc/fcrepo-flexport/index.html

- a shared capability across Fedora/DuraCloud integration, but also for Fedora users as well
- currently at requirements gathering phase
- allows users to be specific about what you want to get out of repository and get content external datastream pointers and grab content
- utility will run at command line and give opportunity to be specific about objects to be exported from fedora and what you want to do with the datastreams
- can give pid patterns (asterisk option), can give sparql query to run against resource index
- selective restore capability as well: only ideas from Chris, tool may be able to copy selected pids, copy from duracloud and replace in fedora

Meeting Minutes

Steve - has written custom script that is currently working (does not need to continue testing)... for export... but useful for Fedora testing in future. would recommend, appears as if you ahve to list out individual datastream id, will be flexible enough to work across all content types (what if no datastream)

Chris - filter section, if any of them match, then the action will be taken, but if you don't specify the id... it will do it with everything (an implicit 'and' in there). selective restore would be very useful in production environment! (library staff log into DuraCloud and be able to restore an object by clicking a button)... where they see the value of DuraCloud, for DuraCloud to recognize Fedora objects is where they see the value of DuraCloud services (seeing Fedora objects and then be able to restore them)

Chris - DuraCloud having Fedora-aware capabilities

Bill - when you are doing a restore, how do you handle external datastreams?

Chris - objects consist of solely managed content... but how to handle 'e' types? don't have an idea for that, but would be out of band with typical Fedora import (would need to know where and how to import)

Bill - would would a Fedora object look like coming back, as well as datastream? would it be xml? could it recognize that it was external?

Chris - there is something with a similar set of routing rules, was thinking self-contained Fedora objects

Steve - perhaps some sort of configuration file

Chris - working on content handler and content resolver. content handler configured to your specific institution and would know what to do with Fedora object. this type of abstraction might work

Bill - are you seeing a vocabulary emerge, like an api, or a configuration tool?

Chris - thoughts have been simple configuration file, but can see more sophisticated cases, using a message bus would be more appropriate

Bill - messaging capability, enabling in workflow

Chris - having messaging capability, is that an immediate requirement

Bill - use it now for backend ingest/preprocessing (message-based). in case of e datastreams on remote system... maybe just need connect information and appropriate set up on remote host, but maybe would like to break into that workflow (premis events, etc., variety of other concerns, indexing, etc.)

Chris - would be curious about vocabulary to meet environment's needs

Ben - were thinking more that they have datastreams wnat to indicate is a preservation datastream and replicate to multiple storage locations. import /export used existing fedora messaging api and it went poorly. akubra plug in?

Chris - yes, one does exist. integration at low level was not simple. need to make asynchronous assumption, content may need to be sent to multiple storage locations.

Ben - also nice would be inverse, what if there was a way instead for fedora to run in duracloud and storage to be in duracloud and replicated storage to be your local storage. ways to leverage duracloud to have less IT overhead. one option is have duracloud replicating on local storage (with listener). hadn't thought about using external/chronical utility to do import/exports and has advantage of preserving object's context. something feels a little jalopy-ish.

Chris - simple import/export will get job done, but will double storage required

Ben - yes, but also administrative overhead on top of doubling storage. some of the real cost barriers are storage and duplicated storage that is not fulfilling preservation function is expensive and bad. reason moved away from original messaging approach to clustering fedora.

Chris - have you thought about restore, even in reverse scenario? how do you recover from scenario when storage is corrupted in either place? import /export too expensive?

Steve - in our case was test system, which would be wanted anyway. but can see it being an issue. in terms of performance, not sure if grabbing content from fedora or from file systems on external storage.

Chris - what to do on simple restore

Steve - perhaps having it be a little more involved process, but still have DuraCloud recognize objects and datastreams

Ben - asynchronous storage, have fedora modify object, perhaps in rels-int

Chris - we have built in assumption that fedora objects is that content is handled by lower level storage. but exposing those locations (duracloud and local), would help make this work better

Ben - has this been discussed with Dan Davis? samfs system presented as file system, but rule-based decisions on where content is put (and some is only on tape).

Chris - not real need from community on this, so probably fedora 4.0 issue, api level, fedora return appropriate http response for content not available yet.

Chris - wrap up. flexport will help in selecting subset of repo for testing purposes. but big thing that is needed is roundtrip and having some sort of hosted interface (here's the fedora object i want to restore, fix it in my repository).

Steve - flexport will be great, but restore will be most useful to Northwestern