OTM Chronopolis Preservation Workflow Specification

Workflows for DDP/Bridge interactions in Specification Flow Diagrams - Version 1, tailored to Chronopolis specifically.

Terminology/Assumptions:

- File: A bitstream
- Preservation Package: A set of related bitstreams (metadata or binaries) which share a unique identifier to be handed off to a distributed digital preservation repository
 - o is the unique identifier still a thing? does this relate to filegroups?
- Stage | Staging Filesystem: A filesystem used to stage data temporarily. This is coupled with a Time To Live (TTL) which tells the user how long they have to retrieve the data.
- ObjectId | FilegroupId: An id associated with a group of files which is expected to persist for the lifetime of the repository and the DDP. Can be used by a repository to reconstitute itself.

Needs:

- Delineation between ephemeral and persistent identifiers
 - ephemeral: deposit-id, delete-id, restore-id
 - persistent: filegroup-id

Send Content

- 1. Query OTM Bridge for deposits which Chronopolis needs to preserve
- a. OTM Bridge API Specification#ListDeposits
- For a given deposit: Create a package consistent with other Chronopolis packages

 This could be BagIt, OCFL, etc
 - b. Use the FileGroupId/ObjectId for the root name of the preservation package
- 3. Perform other Chronopolis tasks for the package
 - a. Generate ACE Tokens for a preservation package
 - b. Could output logging information
- 4. Notify Chronopolis Ingest that a preservation package is ready to be ingested
- 5. Wait for notification that the preservation package has been successfully preserved
- 6. Update OTM Bridge with notification that the deposit has been preserved
 - a. Needs to be specced out

Questions

(i)

- Should the different stages generate audit events? i.e. creating the package, generating tokens, notification of chronopolis, etc
 - In 2, what assumptions can be made about the data? Has the bridge:
 - done any verification of the hashes for the staged data
 what are be acid about during the listing of data?
 - ° what can be said about duplication of data? anything or nothing at all?

Delete Content

- 1. Query OTM Bridge for deletions
 - a. OTM Bridge API Specification#ListDeletes
 - b. Should be able to identify file based on the requesting OTM Bridge user
- 2. Notify Chronopolis staff about removal of the preservation package
- 3. Create tickets for removing the package
- 4. Chron staff removes packages through our deprecation process
 - a. Could be automated; might want some verification before pushing the deletion through the system
- 5. Update the status of the delete to the OTM Bridge
 - Need to spec this out

1. If a file is being removed

- a. Identify where the file is located
- b. Remove the file by ...
 - i. removing it from its filegroup/object
 - ii. OR generate a new package without said file



• Similar questions about how much we know about the request and if there's any extra validation the DDP should do

- At the very least do we know if files/objects/filegroups exist?
- I believe the discussion so far has centered around removing an entire file group (identified by an ObjectId) from the bridge is this true?
- Discussion about expectations of deletion from the system
 Chould environment and any information remain about the Object?
 - Should any information remain about the Object?
 - Audit events
 Eivity information
 - Fixity information

Retrieve Content

- 1. Query OTM Bridge for Restores to be processed
 - a. OTM Bridge API Specification#ListRestores-
- 2. Identify space for the restore to be staged on
 - a. Needs to be accessible by the OTM Gateway
 - b. What if there is insufficient space available?
 - c. Needs to guarantee that space will be available while restoring
- 3. Restaging in Chronopolis
- a. Current process
 - i. A read only mount is available which contains the preservation storage
 - ii. Symbolic links are created from the ro mount to the DuracloudVault restore area
 - b. For OTM Bridge with RO mount
 - i. Could perform a similar process and create symlinks
 - ii. Quick, but don't want to make guarantees about that mount being available (might be an object store in the future)
 - c. For OTM Bridge without RO mount
 - i. Contact Chronopolis and request the content be staged
 - ii. Could be re-staged through rsync, http, etc. Flexible.
 - d. The process which handled the deposit could potentially handle content retrieval as well
- 4. Notify the OTM Bridge that the Restore is staged and is accessible for a given TTL
- 5. Upon expiration of the TTL, remove the staged content
 - a. When does the status get updated in the Bridge?
 - b. Or does the restore cease to exist?
 - c. Can the OTM Bridge be polled for Restores passed their TTL?

Question

- · How to handle errors for insufficient space
- If individual files are requested, does the bridge handle that?
 - Restaging an entire Object could take time, might want the DDP to pull some of its own weight here as well.
 - Restaging large files which are not requested is also wasteful of staging space
- Many options available for returning content, possibly even proxying data
- Is a Restore ephemeral in the OTM Bridge?