Minutes of the HTTP REST TAPE API meeting (28th of January 2022)

Goal of the meeting

- Discuss the need of a disk lifetime on the STAGE request
- Discuss the UNPIN functionality
- Discuss the FILEINFO JSON response message structure

Discussions

The user will be allowed to provide a *disk lifetime* attribute for each file he will stage

It was discussed and agreed that a user will be able to provide a *disk lifetime* for each file he submits for staging.
The aim is to allow StoRM and dCache to have a hint when to UNPIN files from the disk cache after the file has been retrieved from tape.
This parameter will be **optional** and will anyway be ignored by EOSCTA as it does not support this functionality (no PIN concept).
If this parameter is not provided for dCache and StoRM endpoints, a default disk lifetime (configured on their side) will be applied.

This attribute will be part of the common schema and will be named *disk_lifetime*. It must be a non-zero positive integer.

The JSON message a user will have to send in order to stage files from tape will look like:

```
{
  "files": [
      {
         "path": "/eos/atlas/test/test.txt",
         "disk_lifetime":3600, //optional
         "targeted_metadata":
         {
           "eosctaatlas.cern.ch:8080": {
              "activity": "atlas_t0_daq"
           }
         }
      },
      {
         "path": "/eos/atlas/test/test2.txt",
         "targeted_metadata":
         {
           "eosctaatlas.cern.ch:8080": {
              "activity": "atlas_t0_daq"
           }
         }
      }
   ]
}
```

The UNPIN functionality will be renamed RELEASE

As EOSCTA does not have any concept of PIN/UNPIN, and dCache and StoRM do not have an EVICT concept. We decided to change the name of this functionality to RELEASE.

Should the RELEASE be part of the STAGE resource of the API?

Initially, the UNPIN (and now RELEASE) functionality was supposed to be a specific resource of our common REST API allowing a user to tell the storage provider behind the API that he does not need this file to be on disk anymore.

As dCache and StoRM do not allow a user to UNPIN the file for every user, it was decided in previous meetings that the UNPIN functionality would accept a STAGE request id.

EOSCTA does not need a STAGE request id and may want to allow the user to RELEASE files that were not previously staged via the API.

EOSCTA, dCache and StoRM will have internal discussion and report on the next meeting to answer the question whether the RELEASE functionality be part of the STAGE resource of the API or replace the current UNPIN resource.

Should the RELEASE update the state of the files belonging to a STAGE bulk-request?

The question was raised whether the state of the files belonging to a STAGE request needed to be updated in the case a user requested for their releasing?

EOSCTA, dCache and StoRM will have internal discussion and report about this on the next meeting.

Next meeting Friday the 4th of February 2022

- RELEASE functionality part of the STAGE resource?
- Update of the state of the files after releasing?
- Continue the discussion about the schema of the JSON messages exchanged between the client and the REST API: FILEINFO response.