Updates from INFN-T1

Lucia Morganti, Daniele Cesini

CNAF-STORM for the ESCAPE data-lake

Two StoRM WebDAV endpoints are configured for ESCAPE (*xfer.cr.cnaf.infn.it*)

- used in production by ATLAS
- both VOMS AuthN/Z and tokens are supported
- the dedicated storage area is /escape
- files are currently served from /storage/gpfs_atlas/escape path; we proposed to switch to /storage/gpfs_escape
 - <u>https://jira.skatelescope.org/browse/EDLK-112</u>
 - waiting for feedback from ESCAPE about the actual need of moving existing data

CNAF-STORM for the ESCAPE data-lake

- 20 TB disk space are available for ESCAPE in *gpfs_escape*
- Max iNode: previously 100k (on *gpfs_atlas*), now it is not set, which should be better for ESCAPE
- We will create a tape buffer in *gpfs_escape* and provide 10 TB of "simulated" tape storage, to be accessed with a dedicated StoRM endpoint *storm-escape.cr.cnaf.infn.it*
 - simulated tape area will be created on the same filesystem to verify workflow and data layout/ directory structure created on tape system
 - we prefer to avoid too complex directory tree structures on tape

XCache for the ESCAPE data-lake

The considered scenario (see <u>wikipage</u> from Diego Ciangottini):

- one endpoint acting as a remote custodial site (origin) running an xrootd server exposing an HTTP/WebDAV service
- another endpoint working as an XCache instance fetching and storing data from the custodial site (HTTPs access)
- capability-based AuthN/Z model managed with Escape IAM access tokens



XCache for the ESCAPE data-lake

- Both origin and XCache run in two Docker images on the same server *xs-404.cr.cnaf.infn.it*, which is a CMS StoRM WebDAV server
- The origin exposes a dedicated fileset in CMS fs *gpfs_tsm_cms* @CNAF
- XCache uses a dedicated fileset in a fs *gpfs_cache* hosting caches.
- The workflow has been tested with simple exercises
- Currently waiting for Rucio integration, so to test the workflow for embargoed data