



CTA Smart writing

Conceptual proposition

Eric Cano on behalf of the CTA team

The problem

Files tagging by dataset name

User interface

Tape system optimization

Possible bonus features

What are we trying to solve?

- Datasets are always read whole
- Tape systems not dataset-aware during write
 - Files scattered over tapes \Rightarrow more read mounts
 - Files interleaved with others within tape
 - \Rightarrow drive spends time positioning on reads

Files tagging by dataset name

- Per-file property
- Type = string
 - Can we define a length cap?
- Only rely on comparison (no ordering, ranking...)

User interface

- On write, per file tagging
 - Has to go through Rucio/FTS/EOS/CTA
- Back tagging of existing files (several scenarios)
 - Executed as a one-off, we could have rule based update script
 - More general: provide get/set operation per file and leave it to the user

Tape system optimizations

- Write optimization
 - Divide archive queue in per-dataset sub-queue
 - Make write mounts stick to a dataset (until it is drained)
 - \Rightarrow Contiguous files, zero positioning on read
 - Possibly cap the per-dataset parallel writes
 - \Rightarrow Soft-limiting the spreading over tapes
- Repack/defrag
 - Repack can then write in an optimized manner (defrag)
 - Repack input (which files to read) could be dataset driven instead of tape driven
 - If extra read mount cost bearable
 - Will have to take into account tape level constraints as well (will it be worth the complexity?)

Possible bonus features

- Multi-level tagging, allowing to better choose the *next* dataset in a mount
- Retrieve by dataset (implies big changes in whole data transfer chain, and possibly hairy error handling)

