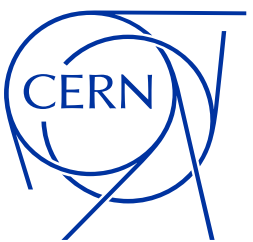
Rucio DC24 Retrospective

Dimitrios Christidis
for the Rucio Team



Summary

- The communities using Rucio reported overall success!
  - Rucio proved to be able to scale and meet the demands of the Data Challenge
  - Increasing the number of component instances and/or threads is a manual endeavour
- Focus on the areas of improvement
  - The injection method is somewhat artificial and exasperated some of the observed issues (e.g. short replication rule lifetime)

Transfer Submission to FTS

- Following an FTS outage, ATLAS reported poor submission performance
  - The top of the queue was dominated by transfer requests of expired replication rules
  - Contention between Rucio daemons
- Tracked in issue #6505
  - ATLAS applied a primitive patch directly on production

Handling of Expired Replication Rules

- ATLAS reported inability to delete expired replication rules on large datasets in their early stages of replication
  - In one case, more than 24 hours
  - Contention between Rucio daemons
- No significant effect on overall performance
- Tracked in issue #6511

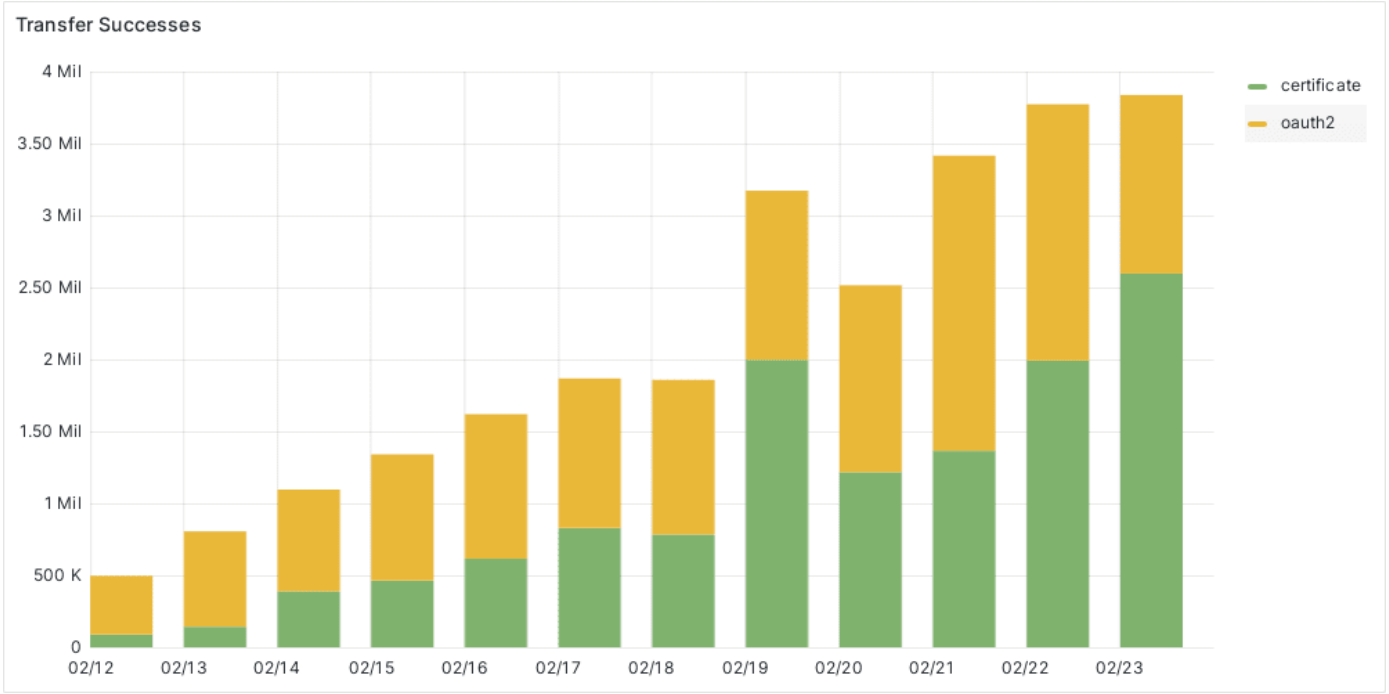
Deletion Overlap on Slower Sites

- CMS reported poor deletion performance at some sites
  - Underlying issue is the rate of deletion at the sites themselves
  - But a design in Rucio does not handle this well, leading to multiple threads working on the same files
  - This hinders performance even further
- Affected dataset reuse and storage occupancy
- Tracked in issue #6512

Tokens (1)

- Both the Rucio and FTS consider their current implementations a ‘technology preview’
  - Insufficient time for thorough testing prior to the Data Challenge
  - Too many choices, too many open questions
  - Lacking guidance from a person or body that oversees the token effort across all projects and can offer concrete advice on the development
- Despite this, and the fact that it was a secondary goal, it was without a doubt a success!

Tokens (2)



Use of tokens over certificates (ATLAS & CMS, ‘Data Challenge’ activity). [Source]

Tokens (3)

- Under the agreed model, did not observe any issues with IAM
  - It would be useful to invest in some monitoring
- RSE-wide *storage.modify* tokens are a cause for concern
  - Acceptable only for the short term
  - Token leaks happened already; they will happen again
  - Reattempt the file-specific destination token test
  - Attempt a hybrid model with RSE-wide *storage.create* on first attempts, then file-specific *storage.modify* on retries

Questions?