

Potential improvements & automations related to data management

Technical Interchange Meeting
2025-01-22

Riccardo Di Maio (CERN), Mario Lassnig (CERN),
Dimitrios Christidis (CERN), Fabio Luchetti (UW)



- Most ATLAS-specific logic has been moved into the policy package
 - `pfn2lfn`, non-deterministic PFN, scope extraction algorithm, permission, schema
- Improve policy package deployment
 - Right now, policy package files are being individually added as secrets
 - Working on fixing this by either deploying the package in an init pod or creating a custom image with the policy package installed as a Python package
 - CMS uses the custom image approach, see [here](#)
- Migrate logic from patches to policy package where needed
 - Some patches might require extending the scope of what Rucio policy packages can do
 - E.g. the existing [tape_metadata.patch](#), which appends metadata to the URL
 - Other patches should instead be implemented in core Rucio code (e.g. [collection_replicas.patch](#))
- Define tape metadata templates for datatypes other than RAW
 - Currently, tape metadata is only enabled for transfers of RAW data
 - We can extend this to other datatypes, if they should be stored on tape (EVNT, AOD, DAOD)

- Moving ATLAS tests to ATLAS VO folder ([#7335](#))
- ATLAS-specific database tables in the DB ([#7336](#))
- Extract lifetime model exceptions into ATLAS policy package ([#7337](#))
- Remove ATLAS hack from replica logic ([#7338](#))
- Move ATLAS-specific BB8 scripts out of core daemon code ([#7339](#))
- Move `atlas_move` RSE Decommissioner profile into ATLAS policy package ([#7340](#))
- Move `policy_filter`-wrapped functions to ATLAS policy package ([#7345](#))
- Remove `ATLAS_SITE_NAME` env variable check when detecting client location ([#7346](#))
- Remove ATLAS-specific space usage endpoint via `gsiftp` protocol ([#7347](#))
- Overhaul Auditor daemon (particularly its ATLAS-specific parts) ([#3437](#))

- Replication Policy on the Grid (RPG)
 - Similar to subscriptions that trigger after a certain amount of days
 - These are currently configured [here](#)
 - Official replication policies from <https://twiki.cern.ch/twiki/bin/view/Atlas/ReplicationPolicy2024>
 - This could be something that is integrated into the core Rucio logic (i.e. when a subscription is created, you could set `min_frozen_time`, `max_age`... for when the logic should be triggered)
 - Relevant issue: [#4508](#)
- Subscription editing can be inconsistent in Rucio UI/CLI
 - E.g. certain special characters can break the parsing on the Rucio UI

Database dumps

- Rucio shovels the majority of its Oracle database contents to HDFS daily
 - Custom *sqoop* scripts, one each per table triggered by cron
 - *sqoop* is obsolete and not updated for many years
 - Effort by Luca Canali (CERN IT) to migrate to Spark-based retrieval a few years ago
 - With the departure of Thomas Beermann we have no one to look into this and put it into production
 - Mario spending a few minutes every other month to keep the existing machinery running and within HDFS quota
 - Two great advantages
 - We have an almost complete "off-site" backup of the Rucio database
 - We can use these database dumps to do calculations that are prohibitively expensive with Oracle/SQL

- The system is extremely brittle

- Depends on fixed Java version because of provided CERN IT Hadoop cluster
- Local Java regularly gets messed up by "other" CERN IT central updates
- Hadoop docker images can run into version conflicts with local downstream reporting tools

```

2021-04-26 03:16 /user/rucio01/dumps/2021-04-26/bad_replicas
2021-04-26 03:22 /user/rucio01/dumps/2021-04-26/collection_replicas
2021-04-26 06:45 /user/rucio01/dumps/2021-04-26/contents
2021-04-26 19:34 /user/rucio01/dumps/2021-04-26/contents_history
2021-04-26 04:25 /user/rucio01/dumps/2021-04-26/dids
2021-04-26 02:09 /user/rucio01/dumps/2021-04-26/dslocks
2021-04-27 05:12 /user/rucio01/dumps/2021-04-26/messages_history
2021-04-26 08:22 /user/rucio01/dumps/2021-04-26/replicas
2021-04-26 02:03 /user/rucio01/dumps/2021-04-26/rse_metadata
2021-04-26 02:03 /user/rucio01/dumps/2021-04-26/rse_protocols
2021-04-26 02:01 /user/rucio01/dumps/2021-04-26/rses
2021-04-26 02:11 /user/rucio01/dumps/2021-04-26/rules
  
```

- It's extremely powerful, but beware who ye enter here...

- Complete reengineering necessary

- Next step is producing the reports, done via Spark
 - Popularity reports, Locks per RSE, Consistency datasets, New replicas per RSE
 - Lost files, Suspicious files, Global accounting, Replicas per RSE
- Same brittle Java/Spark local/docker mechanism with an added twist
 - Many of the reports need to be exposed via HTTP
 - For Hammercloud and many other tools that site-admins wrote
 - Custom Java Servlets/WebApps that stream content directly from HDFS
 - Custom reverse proxy in front to serve content through CERN Outer Perimeter Firewall
- This "report knowledge" is lost to Rucio
 - Any changes/updates to the reports are practically infeasible, e.g., "unique replicas per RSE"
 - Arcane knowledge needed to compile&deploy these servlets
 - Need to think about ways how to request&serve this via Rucio itself

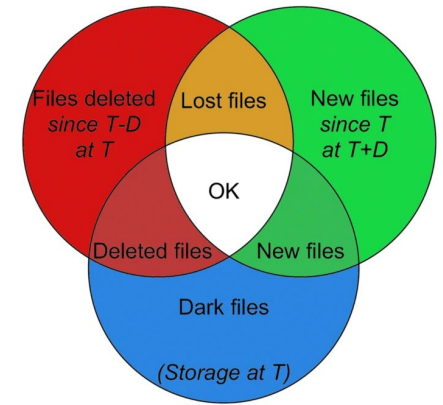
- Crucial not only for ADC but also S&C in general
- Global Accounting
 - Spark3 (better memory efficiency) Scala job to aggregate the data, once a week
 - Spark2 (java compatibility) Scala job to send OpenSearch instance
 - Custom python code that sends it to ActiveMQ where it's ingested by MONIT
- Site Accounting
 - Computed by Oracle job and extracted by Python job to a local CEPH volume
 - Sent with a Python job from CEPH volume to ActiveMQ where it's ingested by MONIT
- Yellowgreen plots
 - Additional accounting reports produced by yet another custom python implementation
- We regularly have problems with these scripts failing
 - Random machine updates, broken dumps, broken reports, ... usually found by CRC
 - Extremely complex to recreate and/or repair
 - Broken bins cannot be repaired by us, requires SNOW tickets to CERN IT
 - Repainting bins necessary when the whole chain fails
 - Need to copy previous days over
 - Knowledge that this was done gets lost in time, then usually triggers questions a year later

- Site-to-site based measurements
 - <https://atlas-rucio-network-metrics.cern.ch/metrics.json>
 - Network throughput, distance, bytes/files sent, bytes/files queued
 - Constructed with separate but chained crons
 - Custom python to prepare aggregate values for PanDA
 - Sent via ActiveMQ to OpenSearch
- Used by PanDA for scheduling

- Vast collection of custom things
 - Python scripts, Scala jobs, Java jobs, cronjobs, Spark jobs, ActiveMQ, OpenSearch
- Now and then we get request for additional features
 - e.g., easy way to get unique replicas per RSE?
- Extending the existing machineries is a fool's errand
- New way to generate reports needed that is customisable also by users
 - Prototype from Luca Canali for database extraction and report generation
<https://gitlab.cern.ch/atlas-adc-ddm/prototypes/spark-rucio-db-import/-/pipelines>
 - Reports are archived and exposed directly on a http accessible eosatlas endpoint

- So far, mostly a long, redundant, manual operation supervised by DDM Ops
 - Some data can be deleted, some has to be copied elsewhere
 - Often lead to discover bad replicas, handled on a case-by-case base
 - Plus the configuration bits (CRIC)
- Rucio rse-decommissioner daemon
 - Can use generic or community specific (ATLAS) profiles to define “what to do” with the data registered on a set of RSEs
 - Decommissioning RSE tagged via attributes
 - Might suspend the operations until an admin/operator validates the progress

- **Eventual consistency between the Rucio catalogue and the sites storages**
 - Relies on the periodic availability of updated storages namespace dumps
 - And on the Rucio catalogue dumps
- **Rucio rse-auditor daemon**
 - Hard-coded ATLAS specific parts
 - Does not scale well (memory bound)
 - Lacks a helm/k8s/flux resource definition, now deployed on a dedicated VM (puppet) that needs to be maintained
 - Relevant issue: [#3437](#) , community effort: [consistency-enforcement](#)
 - Now object of a Qualification Task



End