

# Rucio status

---

[Martin Barisits](#)

on behalf of the Rucio team



# Rucio Coding Camp 2019

- 3 day Coding Camp at CERN
  - 15-17 October
  - [Indico](#)
  - 20 registrations
    - 11 Non-ATLAS
- Agenda
  - Discussion, Brainstorming and technical planning
  - Development
    - [Plan](#) (established developers)
    - Introduction for new developers





# 3<sup>rd</sup> Rucio Community Workshop

- 1<sup>st</sup>: [CERN](#), Switzerland
- 2<sup>nd</sup>: [University of Oslo](#), Norway
- The 3<sup>rd</sup> Rucio Community Workshop will be held in the **United States** at **Fermilab**
- Dates are being finalized right now but most likely week of **23-27 March 2020**
- 2, maybe 3 days workshop
- Will be announced very soon!





# Release plan

---

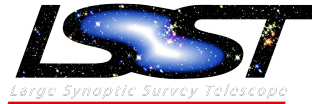
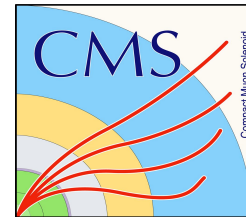
- Three feature releases in 2019
  - **1.19** “Fantastic Donkey” **February 2019** ✓
  - **1.20 LTS** “Wonder Donkey” **June 2019** ✓
  - **1.21** “Donkeys of the Galaxy” **October 2019**
- Feature releases in 2020 (proposal)
  - **1.22** “?” **February 2020**
  - **1.23 LTS** “?” **June 2020**
  - **1.24** “?” **November 2020**
- Will continue to produce release candidates for all feature releases



# Community



Science & Technology  
Facilities Council





# ATLAS Kubernetes usage

---

- Big benefits not only for deployment and scaling but also for isolation of the software due to full control of the environment with containers
- However, we have little experience in running **production** K8s deployments at CERN
  - There are errors (Networking, Routing, ...) we just don't have on our current deployment
  - With current configuration K8s httpd servers seem slower than OpenStack
  - **Need to understand and gain experience!**
- Plan
  - Run full integration cluster on K8s by end of **October 2019**
    - Multiple httpd servers, all daemons
  - Step-wise move of production cluster to K8s during **Q1 2020**
    - Daemon by daemon, having half the workload coverage on OpenStack, half on K8s
    - Full switch to K8s during **Q2 2020**
- Future: Auto-scaling of Rucio components based on workload conditions





# Developments in the last months

---

- Global Quotas → 1.21
  - Current quota system only allows quotas for accounts on a single RSE
  - Global quotas will allow the definition of a quota for an account on an RSE expression
    - E.g. 100TB for account barisits combined on all SCRATCHDISKS
  - Benefits
    - GROUPDISK migration to DATADISK
    - Abuse of scratchdisks by some users
- Source throttling in the conveyor → 1.21
  - Currently only destination throttling
  - For data carousel workflow
- SSO Login to Rucio WebUI → already in 1.20
  - Part of Google Summer of Code
  - Needs to be deployed and integrated



# Developments in the last months

---

- Reaper 2.0 → already in 1.20
  - Current reaper (Deletion daemon) has “practical” scalability issues requiring permanent manual interventions for it to scale properly
    - Reaper daemons are configured to work on a set of RSEs, but workload conditions change constantly
    - Critical milestone for Run3 and Blocker for Run4
  - All workers essentially work on the entire workload, thus a worker will only be idle if all workers are idle
    - Protection for maximum number of threads to delete on an RSE (and hostname) are in place
  - Currently using it to delete on all Tier-3 RSEs
    - Runs stable and performs well, some improvements with selection queries identified





# Developments in the last months

---

- Rucio Multi-VO preparation → already in 1.20
  - Development done by STFC
  - Current Rucio instances can only support a single VO
  - Multi-VO mode will allow to manage multiple VOs with a single instance
  - Designed in a way which is fully transparent to the Rucio core
    - Changes on REST/API level to encode VO selection in scope/account strings
    - Rucio core stays largely unchanged due to this
    - 8k+ lines of code changed; Had only a few, very minor bugs due to this 👍
  - Fully functional Multi-VO Rucio probably with 1.22
- New docker dev environment → already in 1.20
  - Full containerized dev environment with FTS, multiple xrootd storage endpoints
  - Plan is to use this as a basis for full-stack integration tests with travis



# Developments in the last months

---

- Globus Online as a transfertool in Rucio → 1.21
  - Development by BNL for Light source experiment and other usecases
  - Working Prototype implementation, will be merged into Rucio codebase soon
  - Will need testing/commissioning effort!
- Rucio mover (Storm protocol support, pcaches, ...) → already in 1.20
  - See Mario's talk
  - Some improvements needed in error handling and propagation
- CRIC data import → already in 1.20
  - RSEs, accounts
  - Distances to be added soon
  - Currently being tested by Rucio ESCAPE instance
- Lots of bug fixes → in 1.20



# In development

---

- **Generic Multi-Hop support → prototype planned for 1.21**
  - Multi-Hop needed if no distance between SRC and DST or no matching protocols
    - E.g. CTA there is no distance between CTA buffer and the world (Only EOS disk connected)
  - Generic Multi-Hop will cover all these cases
  - Needs to be implemented carefully, since current transfer workflows must not be impacted
- **Token-based authentication → 1.21**
  - See Jaroslav's talk for details
  - Extends authentication to OpenID Connect
  - Development done with funding from XDC (H2020), Prototype testing with IAM on ESCAPE instance
  - Workflow of tokens for TPC and storage interaction discussed by DOMA TPC



# In development

---

- VO Config and general configuration → 1.21
  - VO configurations (permission, policies, schemas) currently part of core repository
    - Will be extracted and VOs will build their own “policy packages”. E.g. atlas-rucio-policy which is installed alongside rucio
  - Over the years a large mixture of configuration options accumulated
    - Cleanup, re-structurization and documentation needed
- Metadata → 1.21/1.22
  - Currently three different interfaces & backends to store metadata: hard-coded columns, generic json columns, key/value map (deprecated)
  - Will be refactored to transparently address all backends with one interface
  - Future:
    - Integrate additional external backends to store metadata



# In development

---

- QoS → iterative, usable probably with 1.23
  - Implement storage Quality of Service in Rucio
  - Users express QoS needs for the replication rules
  - Rucio communicates QoS needs to storage
  - Transition data from one QoS area to another
  - Development will done by ESCAPE project, aligned with DOMA QoS
- Lots of other smaller developments



# Summary

---

- Rucio running stable
- Potential scalability issues for Run3/Run4 being addressed now
- Community is getting bigger
- Many ongoing and new developments
  - Focus on stability
    - Priority is to keep the system stable
    - New developments must not negatively impact the existing system
  - Sustainability
    - New features need to be maintained
    - Trying to highlight this to new developers
- [Rucio Coding Camp 2019](#)



# More information

---

Website



<http://rucio.cern.ch>

Documentation



<https://rucio.readthedocs.io>

Repository



<https://github.com/rucio/>

Images



<https://hub.docker.com/r/rucio/>

Online support



<https://rucio.slack.com/messages/#support/>

Developer contact



[rucio-dev@cern.ch](mailto:rucio-dev@cern.ch)

Publications



<https://rucio.cern.ch/publications.html>

Twitter



<https://twitter.com/RucioData>