

# Rucio Coding Camp 2019

---



# Thank you!

- Thank you for attending the second Rucio Coding Camp!
- 21 physical participants
  - 9 ATLAS
  - 5 CMS
  - 2 SKA
  - 1 DIRAC
  - 1 DUNE
  - 1 ESCAPE
  - 1 Multi-VO Rucio
  - 1 VIRGO





# Organizational

---

- Agenda
  - Tuesday (1pm-6pm) Discussion & Planning (~2h), Coding
  - Wednesday (9am-6pm) Coding
  - Thursday (9am-6pm) Coding
- Rooms
  - Today and tomorrow: This conference room on second floor ([774/2-058](tel:7742058))
  - Thursday: Conference room on first floor ([774/1-079](tel:7741079))
- Snacks
  - Drinks & Snacks in the room, please help yourself
  - Coffee downstairs in the kitchen or Cafeteria
- Lunch
  - Either at CERN (Restaurant 3 [menu](#))
  - Or we get: Pizza ([Pizza Fast Meyrin](#)), Kebab, etc. and eat it downstairs in the kitchen



# Organizational

---

- Dinner
  - Wednesday, 7pm at [Restaurant Bois Joly](#), Crozet, France (Same as last year)  
1131 Route de télécabine  
01170 Crozet  
France
- Transportation
  - Will optimize cars and people for journey to/from Preveysin and to/from the dinner
  - There are also two shuttles leaving from Building 874 (CCC) across the street which go to the Meyrin main building (500)
    - [Shuttle 2](#)
    - [Shuttle 5](#)
- Questions?



# Discussions

---

- Discussion time!
- I prepared a few topics I would like to discuss, please feel free to bring up additional topics
- Notes



# Documentation

---

- Documentation needs improvement
  - Some parts are outdated
  - I am not sure if the structure still fits our needs
  - Specifically operator-related parts need major updates
    - Different step-by-step installations of Rucio
- Recipes and FAQs
  - Lots of questions get asked very often; Answers are clearly beneficial for the entire community
    - Collecting them and updating the documentation works in theory, but in practice it is just not happening
  - → Community driven way to ask questions and easily collect answers?
    - WIKI? Or some kind of service?
- Looking for people to invest some time to improve documentation
  - Don't have to be Rucio experts, just interface with them; A fresh look on the documentation is very helpful



# Handling of LTS patches

---

- From 1.21 onwards we will have our first LTS release to maintain
  - Until ~June 2021 we need to provide security and critical patches for this release → `lts-patches`
- New helper script `tools/create-lts-patch-branch.sh`
  - Will create a branch based on the `1.21-lts` branch on upstream
  - Three PRs will be created
    - `<dev>` → `1.21-lts`, `<dev>` → `master`, `<dev>` → `next`
    - Merging against `master` and `next` might have conflicts
  - Version tags for the LTS release line will be done in its respective branch
- New LTS label for issues
  - Please suggest if an Issue is critical/security related and should be included in the LTS release line
- Issue template will get a field for affected versions
- If you plan to do a bug-fix, please be ready to be asked to do this as an `lts-patch`



# web.py

---

- [webpy](#) (web framework) is the backbone of our REST interface
- Not really maintained
  - Although more recently there are some movements
  - 0.40 was released two weeks ago
- Several new WSGI based frameworks are much better maintained
  - [Flask](#) already used in our codebase to do auto-generation of REST commands
  - Also enables us to use other community tools ([Swagger](#), ...)
  - More likely to see new features AND security fixes in these packages
- But
  - Major restructuring needed due to the different exception handling with webpy
    - Exception in streaming APIs are handled very differently





# Metadata

---

- Multiple “backends”
  - Fixed rows in dids table - addressed by `core/did.py:set_metadata()`
  - JSON column - addressed by `core/did.py:add_did_meta()`
- Should be unified to one core interface
- Address the right source in the backend
- Potentially add possibility to have additional sources
  - External metadata catalogs?



# Dependencies

---

- Lately we are having an increased amount of issues with dependencies
  - Especially with the py2 → py3 migration
    - E.g. `fts3-rest-API` package
- Some of our dependencies are not adequately maintained
- Currently
  - 21 core dependencies (pip-requires)
  - 15 client dependencies (pip-requires-clients)
  - 19 testing dependencies (pip-requires-test)
  - 6 extras (via `setup.py`)
- We should try to slim down and remove dependencies which are not strictly needed
- Adding a new dependency should be very well motivated



# 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>