



PIC  
port d'informació  
científica



**RUCIO**

# Rucio JupyterLab Extension status

7th Rucio Community Workshop, 1st October 2024

*Francesc Torradeflot, Enrique Garcia, Giovanni Guerrieri*



# Outline



- Introduction
- Overview
  - Features
  - Architecture
- Current status:
  - Latest developments
  - Future work
- Use cases
- Demo



# Introduction



The Rucio JupyterLab Extension was developed by Muhammad Aditya Hilmy in 2020 in the context of a Google Summer of Code internship. He is still (by far) the main contributor to the project.

It was adopted as an official Rucio component in 2024 and is now maintained by Francesc Torradeflot and Enrique Garcia.

Its main purpose was to provide the scientists an easier access to the data, by bridging the gap between the analysis tool (JupyterLab - SWAN) and the data lake (Rucio).



# Overview



The screenshot displays the RUCIO JupyterLab interface. The top menu bar includes File, Edit, View, Run, Kernel, Tabs, Settings, and Help. The left sidebar shows the RUCIO logo, navigation tabs for EXPLORE and NOTEBOOK, and search results for 'test:dataset1'. The main notebook area, titled 'Untitled.ipynb', shows a Python 3 (ipykernel) environment with the following code cells:

```
[1]: ds1
```

```
[1]: [/tmp/rucio_xrd1/test/80/25/file1, /tmp/rucio_xrd1/test/f3/14/file2]
```

```
[ ]:
```

The bottom status bar indicates 'Simple' mode, 'Python 3 (ipykernel) | Idle', and 'Mode: Edit Ln 1, Col 1 Untitled.ipynb 1'.

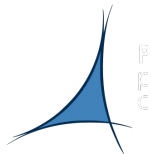


# Overview - Features

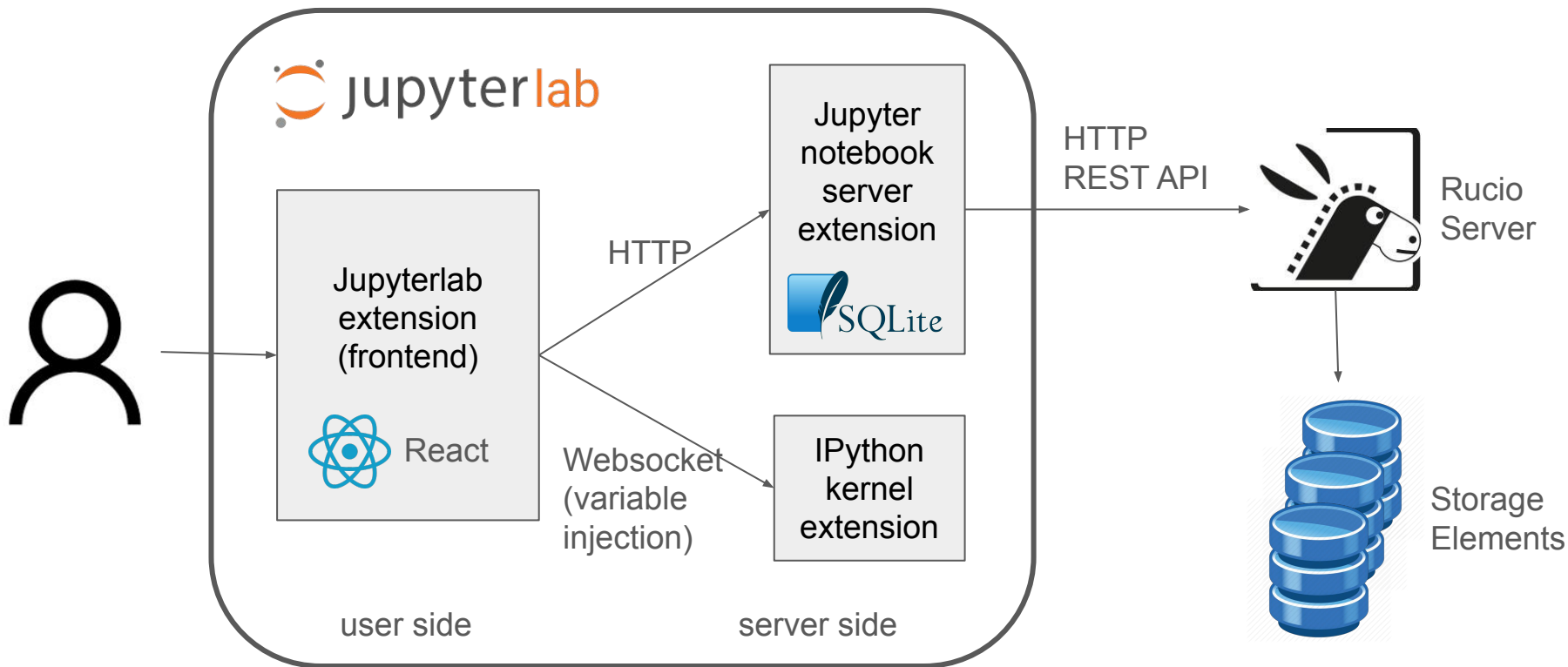


- Browse Rucio data from the Lab sidebar
- Replicate data with just one click
- Resolves file path automagically
- Inject path to notebook as a variable
- Supports three methods of authentication (currently)
  - Username & Password
  - X.509 User Certificate (or Proxy)
  - OIDC tokens (with limitations)
- Supports two modes of operation:
  - **Replica mode:** uses network-attached storage as a Rucio Storage Element (RSE), utilizes Rucio's file transfer capability.
  - **Download mode:** downloads data directly to the user's directory using Rucio clients.

Content from Muhammad's [Rucio-SWAN Integration Project slides](#)



# Overview - Architecture





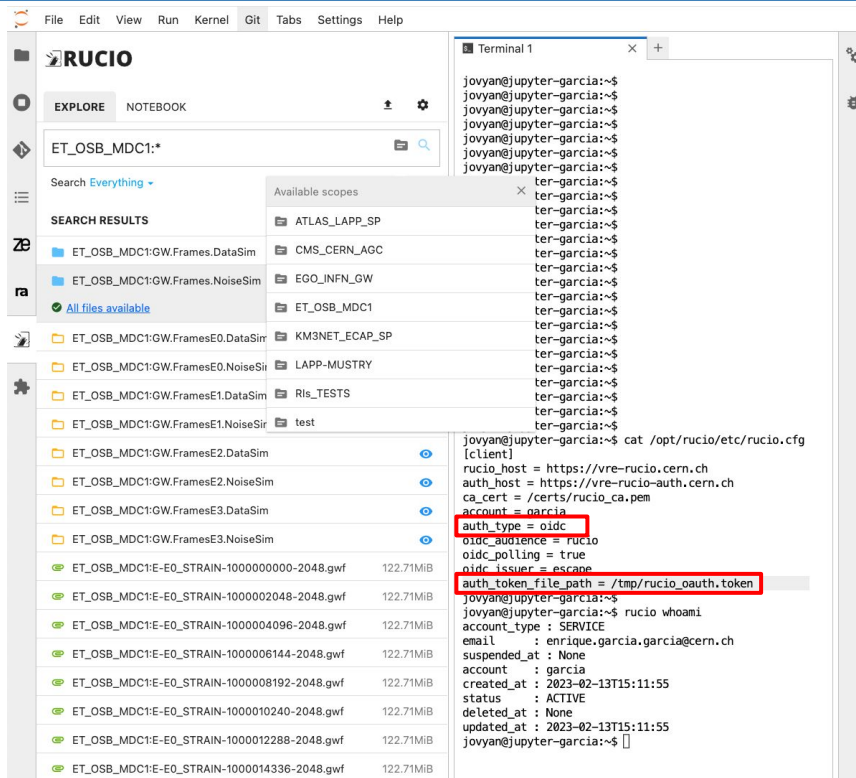
# Current Status



- Latest developments
  - [#27](#) Migration from JupyterLab v3 to v4 (Done)
  - [#29](#) Creation of a test environment (relying on Rucio's test env) (Done)
  - [#36](#) Apply filters in DID search. (credit to Georgy Skorobogatov) (In Review)
- Future work
  - [#35](#) Include docker image publication in CI/CD
  - [#39](#) Improve handling of errors in Rucio REST API calls



- CERN VRE is an analysis facility developed within ESCAPE project
  - First environment hosting the extension since it's development.
- Extension connected to ESCAPE Rucio Instance (deployed @CERN)
  - Interacts with an EOS instance
    - Mounted in the jlab localhost
    - Configured as a RSE
  - Uses OIDC tokens as auth method
  - Tested with 1.28.0 → 1.30.0 releases
  - Release v34 raised some problems with OIDC authentication
    - X509 proxy works fine, though.
- Extension is currently being integrated in the CERN/SWAN framework.



The screenshot shows the RUCIO web interface with a search for 'ET\_OSB\_MDC1:\*' and a list of search results. A terminal window is open, showing the output of the 'cat /opt/rucio/etc/rucio.cfg' command. The configuration file content is as follows:

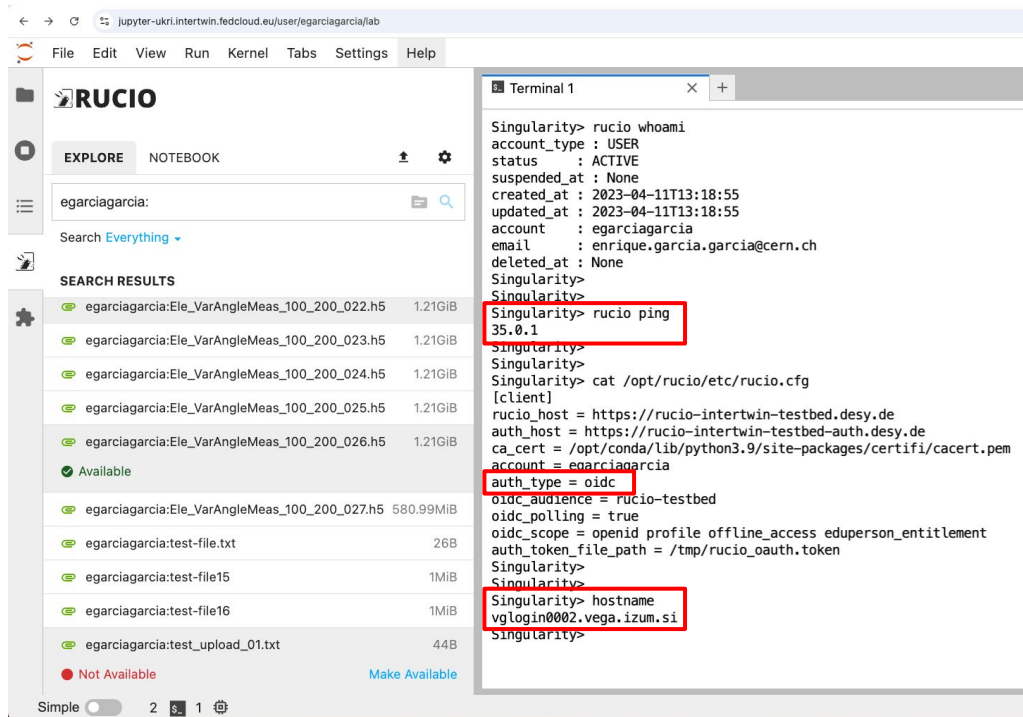
```

jovyan@upyter-garcia:~$ cat /opt/rucio/etc/rucio.cfg
[client]
rucio_host = https://vre-rucio.cern.ch
auth_host = https://vre-rucio-auth.cern.ch
ca_cert = /certs/rucio_ca.pem
account = garcia
auth_type = oidc
oidc_audience = rucio
oidc_polling = true
oidc_issuer = escape
auth_token_file_path = /tmp/rucio_outh.token
jovyan@upyter-garcia:~$ rucio whoami
account : SERVICE
email   : enrique.garcia.garcia@cern.ch
suspended_at : None
account : garcia
created_at : 2023-02-13T15:11:55
status  : ACTIVE
deleted_at : None
updated_at : 2023-02-13T15:11:55
jovyan@upyter-garcia:~$
  
```





- EU project building a prototype of a Digital Twin Engine
  - Data infrastructure based on the ESCAPE Data Lake (RUCIO + FTS + Federated AAI)
  - Rucio instance deployed @ DESY
- Rucio extension available @ VEGA HPC Center
  - Uses OIDC as auth method
  - Interacts with a dCache instance
    - Volume mounted on VEGA
    - Configured also as a RSE



The screenshot shows a JupyterLab interface with a sidebar on the left displaying the RUCIO logo and search results. The main area shows a terminal window with the following output:

```

Singularity> rucio whoami
account_type : USER
status       : ACTIVE
suspended_at : None
created_at   : 2023-04-11T13:18:55
updated_at   : 2023-04-11T13:18:55
account      : egarciagarcia
email        : enrique.garcia.garcia@cern.ch
deleted_at   : None
Singularity>
Singularity> rucio ping
35.0.1
Singularity>
Singularity> cat /opt/rucio/etc/rucio.cfg
[client]
rucio_host = https://rucio-intertwin-testbed.desy.de
auth_host = https://rucio-intertwin-testbed-auth.desy.de
ca_cert = /opt/conda/lib/python3.9/site-packages/certifi/cacert.pem
account = egarciagarcia
auth_type = oidc
oidc_audience = rucio-testbed
oidc_polling = true
oidc_scope = openid profile offline_access eduperson_entitlement
auth_token_file_path = /tmp/rucio_oauth.token
Singularity>
Singularity> hostname
vglogin0002.vega.izum.si
Singularity>
    
```



# Thank you