

Data Lake as a Service

It's a lake.. but for data.. presented as a service

Muhammad Aditya Hilmy

CERN Openlab Summer Student 2021

ESCAPE WP2 Fortnightly Meeting, 16 Sep 2021

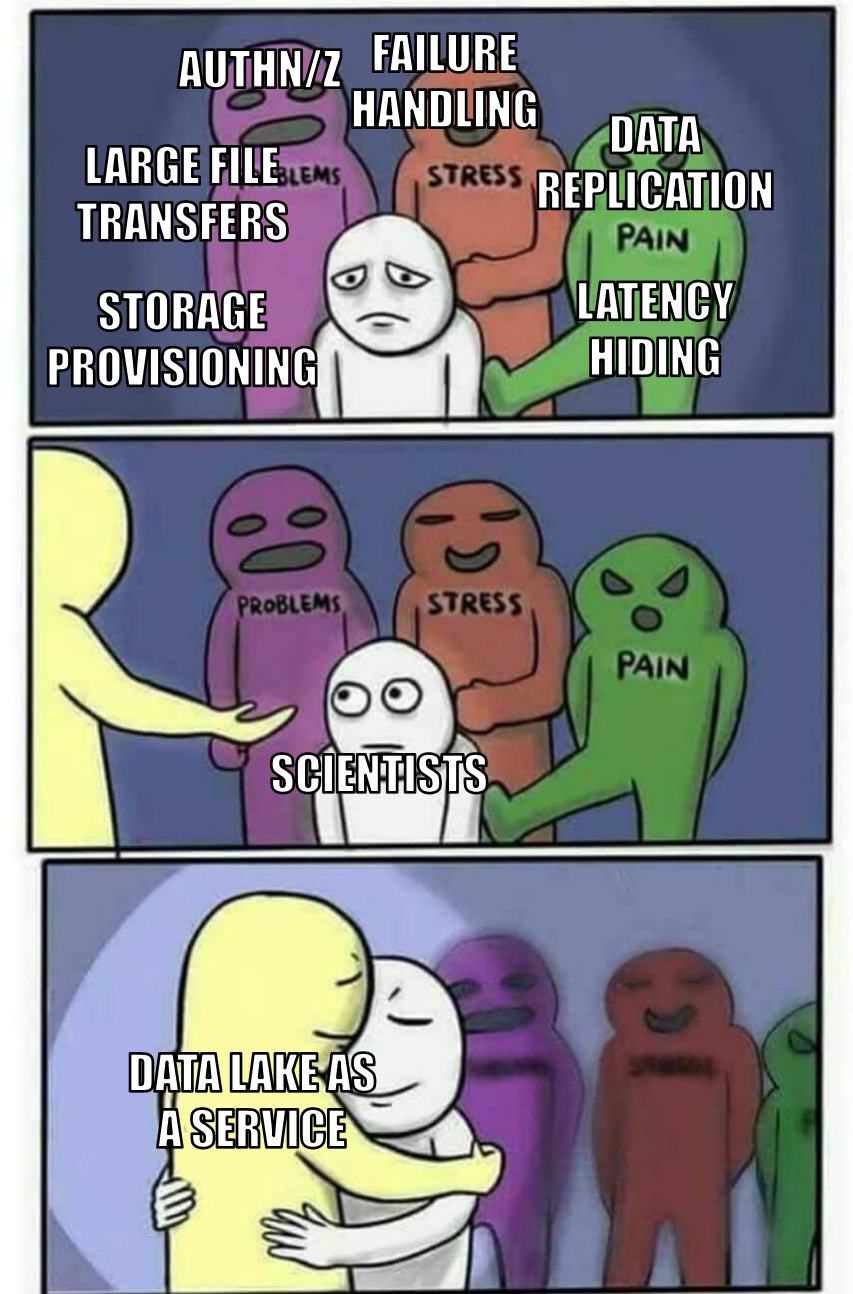
A bit of context

- We will have HL-LHC and other experiments coming online.
- Data volume expected to increase to exabyte scale.
- We need to think about how to store and manage the data.
- The Data Lake is a place where experiments can 'dump' their data.
- ...and scientists can 'fish' data from.
- The challenge: making sure the scientists can 'fish' easily.



Making 'data fishing' easier

- The Data Lake has a lot of moving parts.
- The goal of the service is to abstract the complexities of the Data Lake from the scientists.
- This way, scientists can focus their time on doing science instead of data procurement.





<https://youtu.be/pRLI3fhuWc>

Feature Highlights

- Multiple notebook environment selection
- Rucio data browser (with scope browser and wildcard search)
- “Add to shopping cart” for data catalogue
 - DID is attached as a metadata in the Notebook file
- Injects a variable containing local file path, ready to be used
- Direct file upload to Rucio
- Scratch space for large files (EOS FUSE mount)
 - Files older than two days old are deleted automatically
 - TPC file upload using scratch space



RUCIO

Upload AUTHORS.md to Rucio

Please make sure that the necessary credentials are configured. You can see the upload status on the Rucio sidebar.

Destination RSE Expression:

Lifetime (in seconds):

Scope:
 ▼

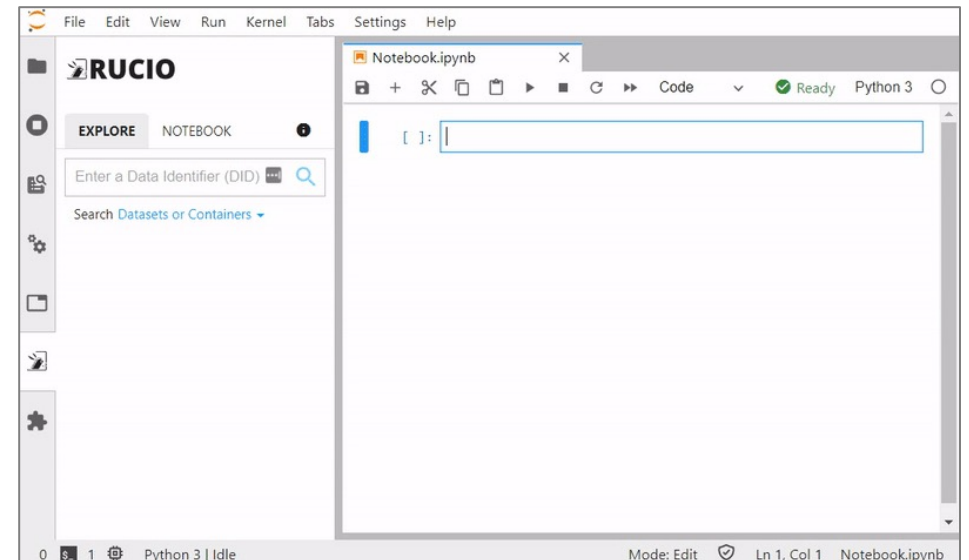
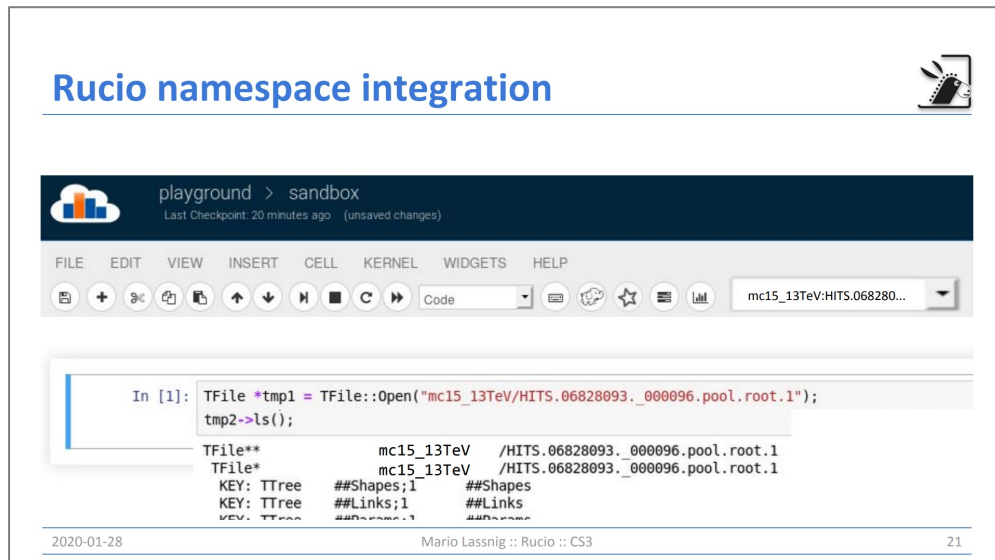
```
jovyan@jupyter-muhilmy:~$ datalake upload --help
Usage: datalake upload [OPTIONS] [PATHS]...

Upload specified files within the scratch space to Rucio

Options:
  -s, --scope TEXT           Scope of the uploaded files [required]
  -p, --prefix TEXT          Prefix of the DID name
  -t, --lifetime INTEGER     Replication rule lifetime in days (Use 0 for
                             indefinite lifetime)
  --rse TEXT                 RSE expression for the replication rule
  --delay-deletion            If false, the file in /scratch will be deleted
                             immediately after a successful upload
  --help                     Show this message and exit.
jovyan@jupyter-muhilmy:~$
```

A humble beginning

- Started as an idea presented on CS3 2020 by the Rucio team [1]
- Developed “Rucio JupyterLab Extension” as a part of Google Summer of Code 2020 [2]
- Deployed the extension as “Data Lake as a Service” as a part of Openlab Online (🙄) Summer Students Programme 2021

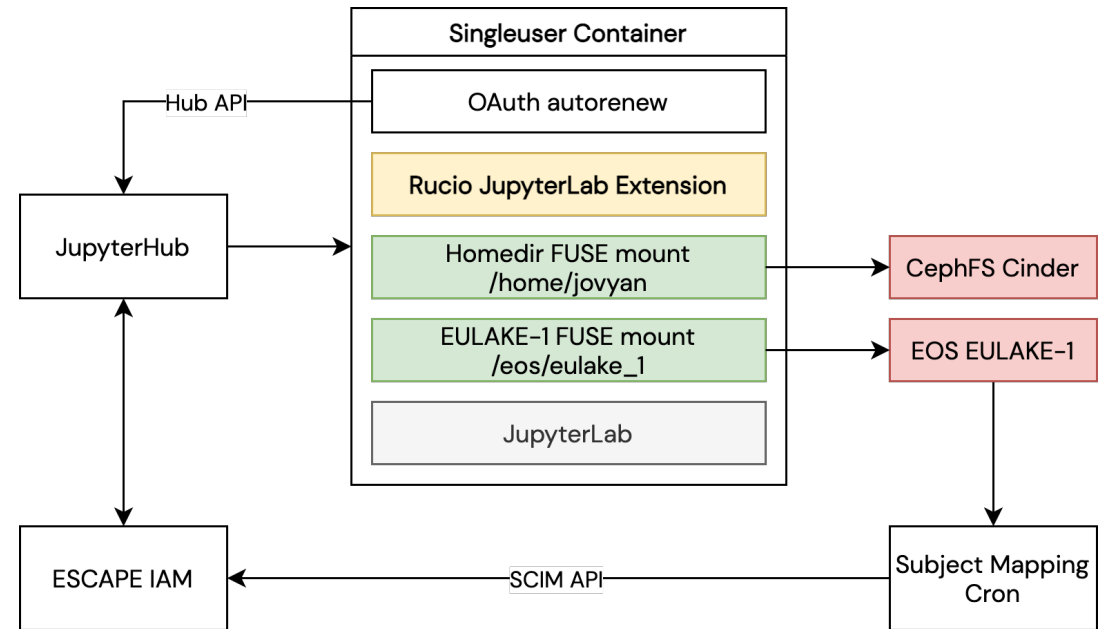


[1] <https://indico.cern.ch/event/854707/contributions/3680520/>

[2] https://hepsoftwarefoundation.org/gsoc/2020/proposal_SWAN_RUCIO_integration.html

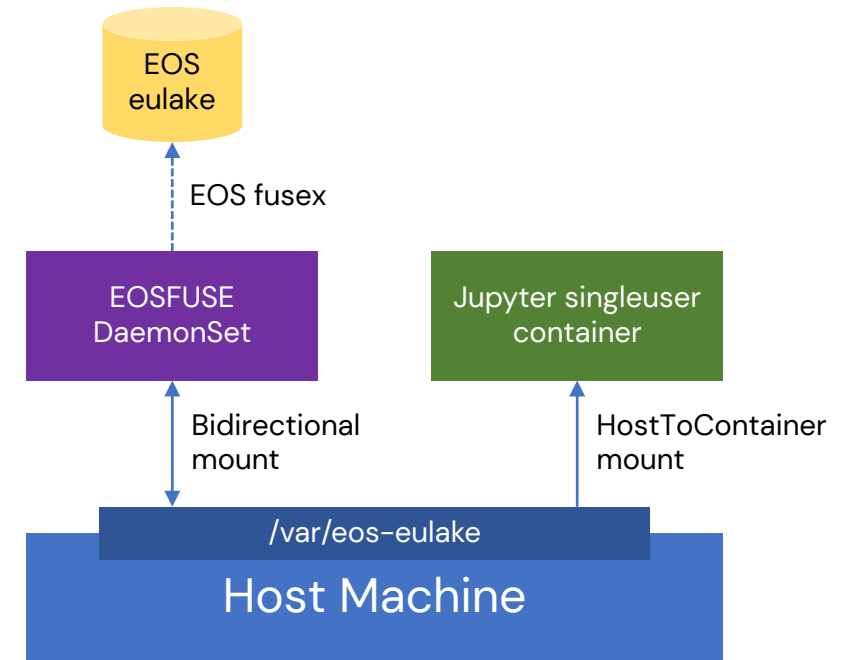
Deployment

- Deployed in Kubernetes @ CERN Openstack, using Zero-to-JupyterHub Helm chart.
 - <https://escape-notebook.cern.ch>
- OAuth authentication using ESCAPE IAM.
 - X509 and Userpass are still supported
- Uses [Rucio JupyterLab Extension](#) in Replica mode (i.e. TPC to local storage)
 - Connected to ESCAPE Data Lake (escape-rucio.cern.ch)
 - Automatically preconfigured to use OIDC authentication
 - Has a FUSE mount to EULAKE-1 RSE (EOS)
 - Making files available means creating a replication rule to move files to EULAKE-1
 - Download mode is still possible, if configured



FUSE mount to EOS eulake

- There are two FUSE mounts to the same EOS instance:
 - `/eos/eulake_1` → `/eos/eulake/tests/rucio_test/eulake_1`
 - `/scratch` → `/eos/eulake/tests/jupyter-scratch`
- FUSE mount is implemented using k8s DaemonSet, mounting to a folder in the host, with Bidirectional mount propagation
- Singleuser containers bind to the mount folder, with HostToContainer mount propagation
- Uses OAuth2 authentication
 - ESCAPE IAM user is mapped to EOS user using crons



OAuth2 in EOS FUSE mount

- In the singleuser container:
 - JWT is stored in a file in the following format:
 - `oauth2:<jwt>:<token-introspection-endpoint>`
 - Example: `oauth2:eyJ...:iam-escape.cloud.cnaf.infn.it/userinfo`
 - Note: token introspection endpoint doesn't have the "https://" part
 - The token file must have at most 0600 permission
 - An environment variable needs to be set:
 - `OAuth2_TOKEN=FILE:/path/to/token/file`
- In the EOSFUSE DaemonSet container:
 - EOS FUSEx daemon (eosxd) needs to be configured for SSS authentication
 - SSS keytab needs to be present

Docs: <https://eos-docs.web.cern.ch/using/oauth2.html>

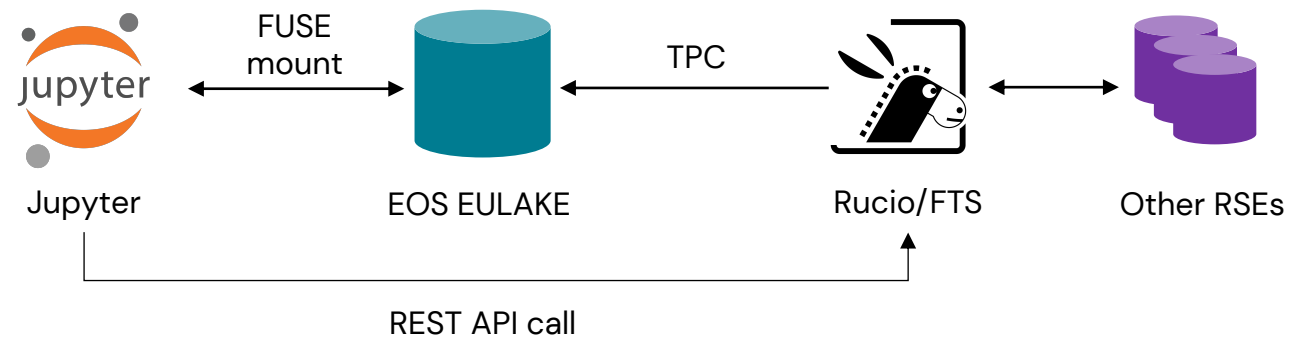
Singleuser container setup

Some things need to happen:

- OAuth token exchange (eos-eulake and rucio)
 - Uses a modified version of SWAN's KeyCloakAuthenticator
- Enable token autorenewal
 - Uses [swanoauthrenew](#)
- Write token files to /tmp
- Set OAUTH2_TOKEN env for EOS authentication
- Write rucio.cfg file

Making files available

- Replica mode: uses Third Party Copy (TPC)
- EULAKE-1 is a Rucio Storage Element and is FUSE-mounted to /eos/eulake_1
- When “Make Available” is clicked:
 - The extension creates a replication rule to move requested files into EULAKE-1
 - Lifetime is set to 7 days (configurable by service admins)
 - Rucio will move the files to EULAKE-1
 - Once the replication status is OK, the extension translates the Physical File Name into local path
 - `root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/file` → `/eos/eulake_1/file`
 - File is accessible as if it were local



Uploading files in scratch (..technically EOS eulake)

When “datalake upload” is run:

- The script translates local path to full Physical File Name:
 - /scratch/muhiomy/file → root://eoseulake.cern.ch:1094//eos/eulake/tests/jupyter-scratch/muhiomy/file
- The file in scratch is added to the Rucio replica catalogue
- A replication rule is created to move the files from scratch space to a destination storage
- Rucio will move the files to the destination storage
- When the replication status is OK, Rucio will delete the file in scratch
- A cron job will run every 24h to delete files (and folders) older than 2 days old that might not be in the Rucio catalogue

Integration with XCache

- When running in Download mode, having a caching layer would be useful
- XCache URL must be registered in Rucio
- Rucio will prepend the XCache path if the client site name != RSE site name

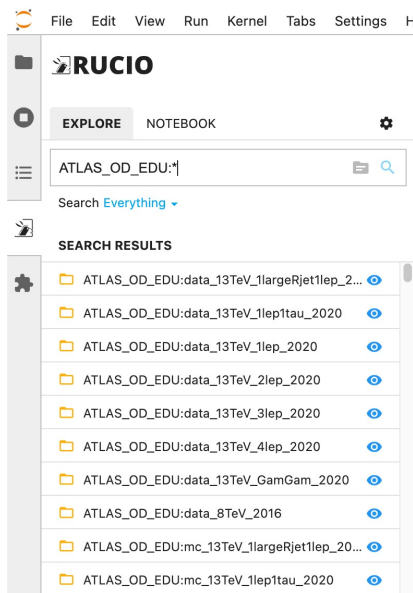
```
jovyan@jupyter-muhilmy:~$ SITE_NAME=xcache_test rucio list-file-replicas ATLAS_LAPP_JEZEQUEL:data.root --protocol root
```

```
+-----+
| RSE: REPLICA                                     |
+-----+
| EULAKE-1: root://xcache-redirector.cern.ch//root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/bd/8f/data.root |
| ALPAMED-DPM: root://xcache-redirector.cern.ch//root://lapp-testse01.in2p3.fr:1094//dpm/in2p3.fr/home/escape/rucio/lapp_dpm/ATLAS_LAPP_JEZEQUEL/bd/8f/data.root |
+-----+
```

- Challenge: XCache needs to be configured to accept all RSEs in the Data Lake as origin
 - We cannot allow all origins, since that would be problematic for AuthN/Z
- Solution: A cron job that populates /etc/xrootd/Authfile using entries from Rucio

Use Cases

- Data discovery and access
- Submitting jobs to external service (remote computing)
 - Users can use the convenience of the extension to browse data in Rucio and access the file PFN directly from the notebook code.

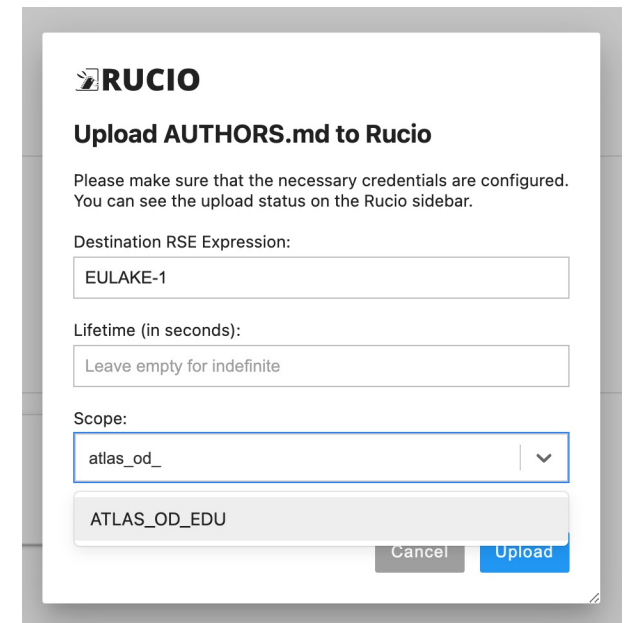
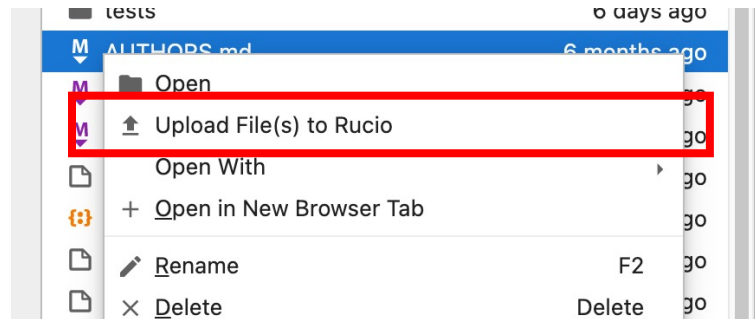


```
[3]: for item in hy_20:
      print(item.pfn)

root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/6f/98/data_A.GamGam.root
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/f1/3a/data_B.GamGam.root
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/45/95/data_C.GamGam.root
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/73/e3/data_D.GamGam.root
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/6d/aa/mc_341081.tH125_gamgam.GamGam.root.1
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/1b/95/mc_343981.ggH125_gamgam.GamGam.root.1
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/ff/c7/mc_345041.VBFH125_gamgam.GamGam.root.1
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/13/b8/mc_345318.WpH125J_Wincl_gamgam.GamGam.root.1
root://eoseulake.cern.ch:1094//eos/eulake/tests/rucio_test/eulake_1/ATLAS_LAPP_JEZEQUEL/76/fd/mc_345319.ZH125J_Zincl_gamgam.GamGam.root
```

Use Cases (2)

- Data preparation and processing
 - Use the service to preprocess data, and once done, upload it back to the Data Lake.
- Data preservation
 - Use the service to produce data and reupload them to the Data Lake

A screenshot of the RUCIO web interface showing the 'Upload AUTHORS.md to Rucio' dialog box. The dialog includes the RUCIO logo, a title, and instructions. It contains three input fields: 'Destination RSE Expression' with the value 'EULAKE-1', 'Lifetime (in seconds)' with the value 'Leave empty for indefinite', and 'Scope' with a dropdown menu showing 'atlas_od_'. Below the dropdown is a button labeled 'ATLAS_OD_EDU'. At the bottom right are 'Cancel' and 'Upload' buttons.

RUCIO

Upload AUTHORS.md to Rucio

Please make sure that the necessary credentials are configured. You can see the upload status on the Rucio sidebar.

Destination RSE Expression:

Lifetime (in seconds):

Scope:

▼

ATLAS_OD_EDU

Cancel Upload

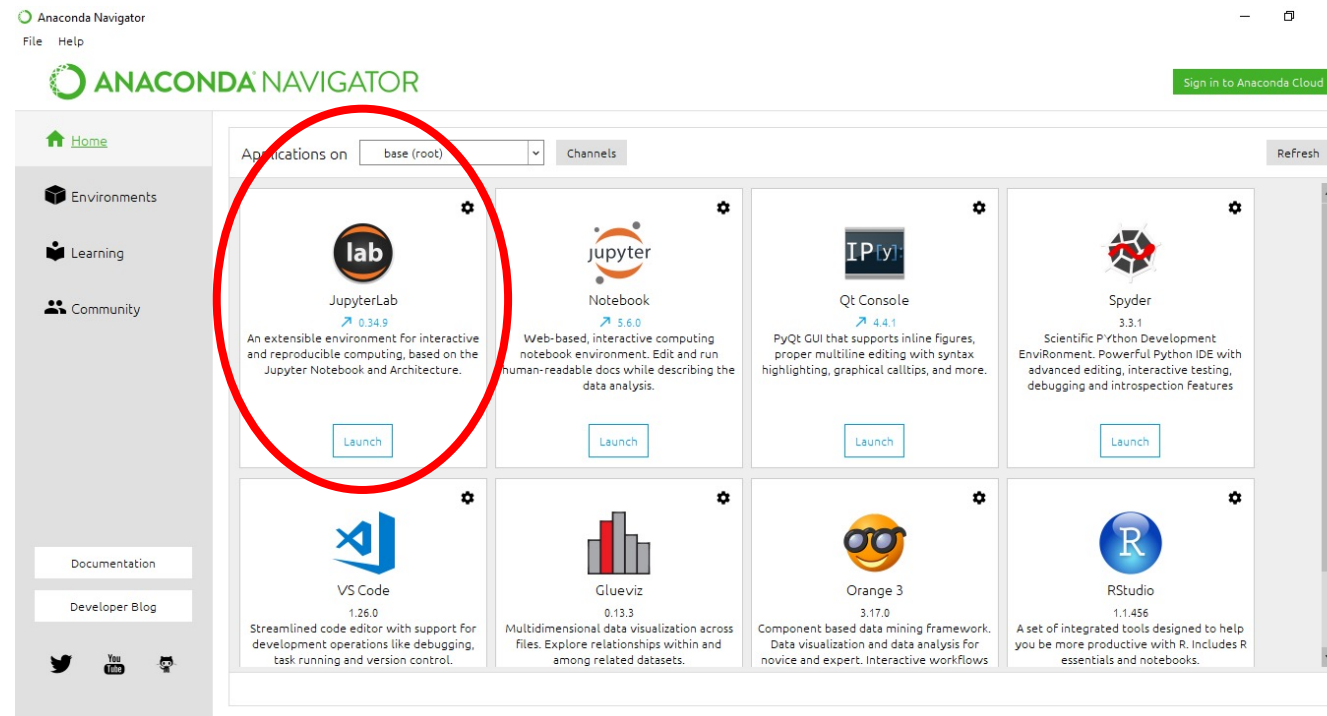
Future Developments

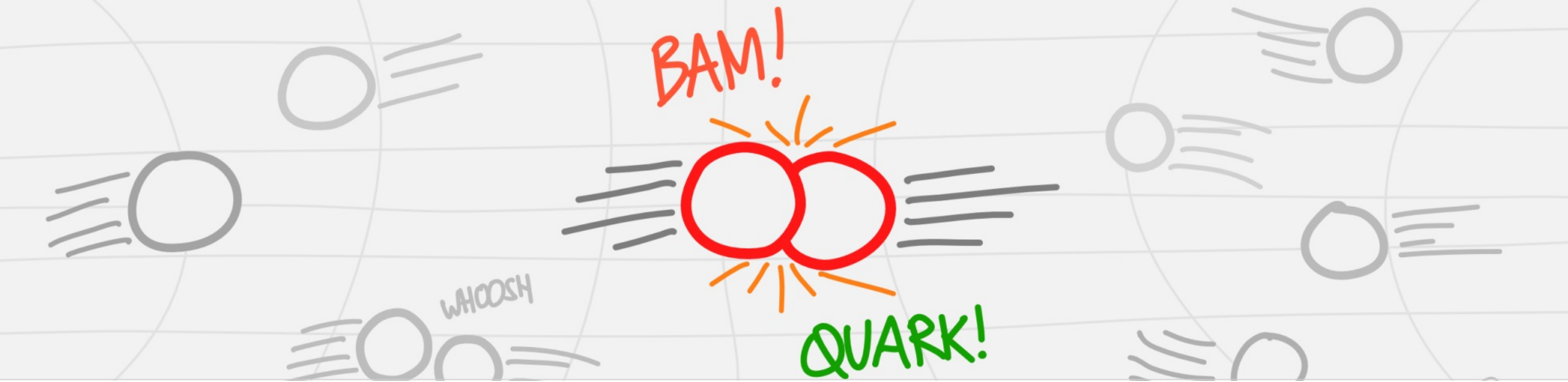
- More kernel compatibility
 - Currently, only Python is supported
- Token-support for direct download and upload
 - OIDC integration ongoing to all remaining ESCAPE RSEs.
- Integration with content delivery and caching layer
 - XCache can be integrated to allow faster file download
 - Will be completely transparent from the user PoV
 - Successfully tested at small scale
- Integration with SWAN
 - Extension was evaluated on a SWAN instance, was working out-of-the-box
 - SWAN migration to JupyterLab 3 is in progress

Desktop Data Lake as a Service

An installable package that gives the possibility to connect to the Data Lake seamlessly.

(A preconfigured JupyterLab installation in Anaconda Navigator could be an option)





Thank you.

 Muhammad Aditya Hilmy

 mhilmy@hey.com

 didithilmy