

# Connecting REANA and the CERN-VRE: a JupyterLab extension middleware

*Rubén Pérez Mercado*

Enrique García García, Giovanni Guerrieri

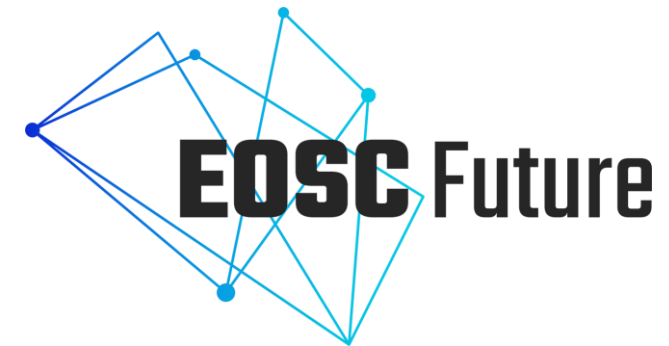
IT-GOV-ENG

August 2024

# Introduction

## The Virtual Research Environment (VRE):

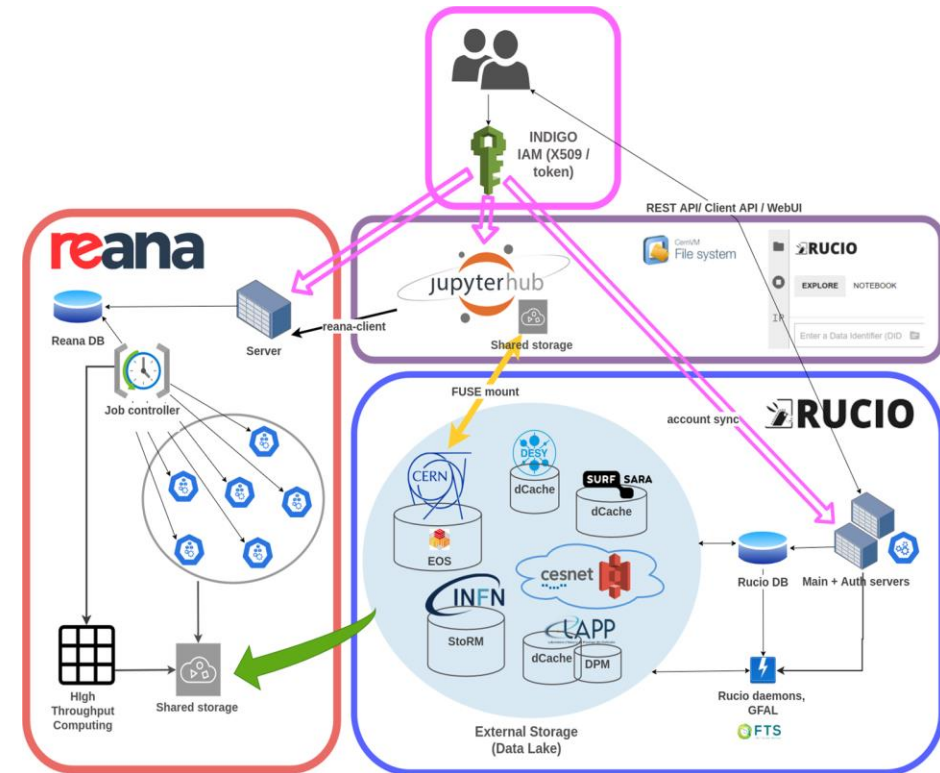
- Collaborative analysis platform where researchers can develop and share end-to-end workflows.
- Developed at CERN within the ESCAPE and EOSC (European Open Science Cloud) Future projects.
- Showcases how disciplines ranging from HEP to Astrophysics could benefit from the usage of common technologies.



# Introduction

## The Virtual Research Environment (VRE):

- **Four main components:**
  - A federated Authentication and Authorization layer.
  - A federated distributed storage solution (the Data Lake) with a Data Management framework (Rucio).
  - An enhanced **notebook interface**.
  - A computing cluster with a re-analysis software (**Reana**).

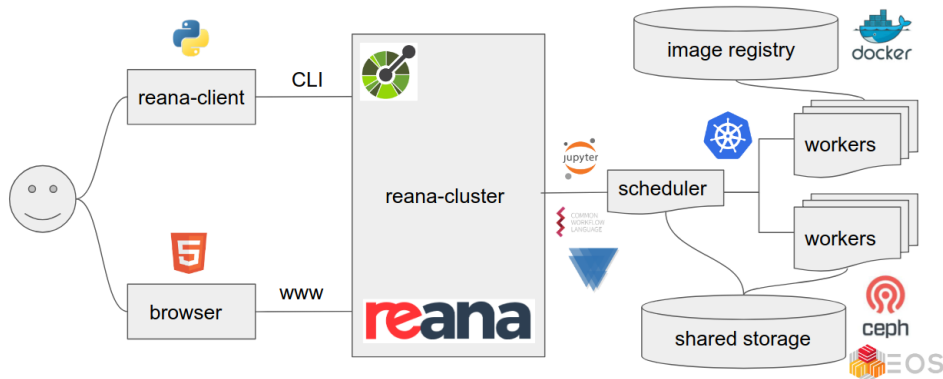


**VRE-hub:** <https://vre-hub.github.io>

# Introduction

## REANA:

- REANA is a **reproducible analysis** platform that allows scientists to run containerised data analysis pipelines on remote compute clouds.
- Analysis are defined in a YAML file, describing the inputs, workflow steps and the outputs.



Reana: <https://docs.reana.io/>

reana



## Your workflows

Refreshed at 14:52:54 UTC

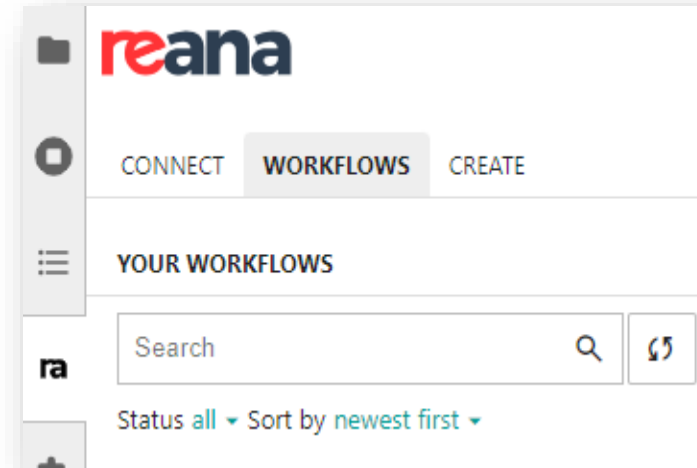
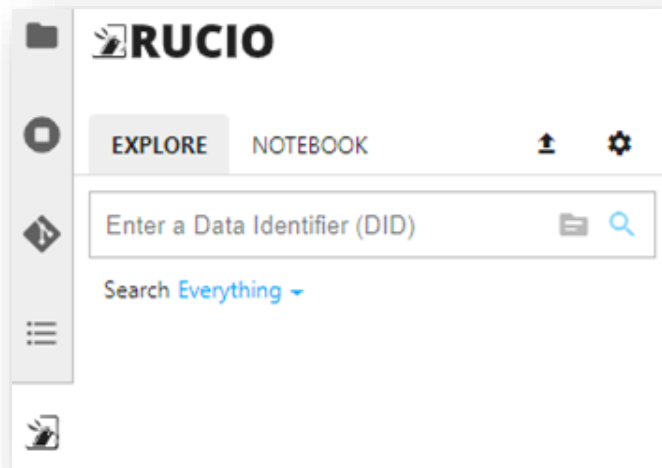
Search...	
Status	Latest first
✔ roofit #26 Finished an hour ago	finished in 22 seconds step 2/2
✔ roofit #25 Finished 3 hours ago	finished in 19 seconds step 2/2
✔ roofit #24 Finished 5 hours ago	finished in 22 seconds step 2/2
✔ roofit #23 Finished 5 hours ago	finished in 22 seconds step 2/2
✔ roofit #22 Finished 5 hours ago	finished in 21 seconds step 2/2

« < 1 2 3 4 5 ... 8 > »

Reana-UI: <https://github.com/reanahub/reana-ui>

# Motivation

- In order to use Reana in our virtual environment we need to know how to use **reana-client** and check the progress of the workflows with **reana-ui**.
- As we did for Rucio, we aim to integrate REANA and its functionalities into the JupyterLab environment.



# Reana JupyterLab extension

## Main features

Connect to any Reana server

List workflows

Interact with a workflow

Create workflow from the environment

## UI Requirements

Responsive

User-friendly



# Reana JupyterLab extension

## Main features

Connect to any Reana server ✓

List workflows

Interact with a workflow

Create workflow from the environment

## UI Requirements

Responsive

User-friendly

The screenshot shows the Reana JupyterLab extension interface. At the top left is the Reana logo. Below it are navigation tabs: 'CONNECT' (highlighted), 'WORKFLOWS', and 'CREATE'. A hamburger menu icon is visible. The main content area is titled 'CONNECT TO REANA'. It contains a 'Server Name' input field with the value 'https://reana-vre.cern.ch', an 'Access Token' input field with masked characters, and a 'Connect' button.

# Reana JupyterLab extension

## Main features

Connect to any Reana server ✓

List workflows ✓

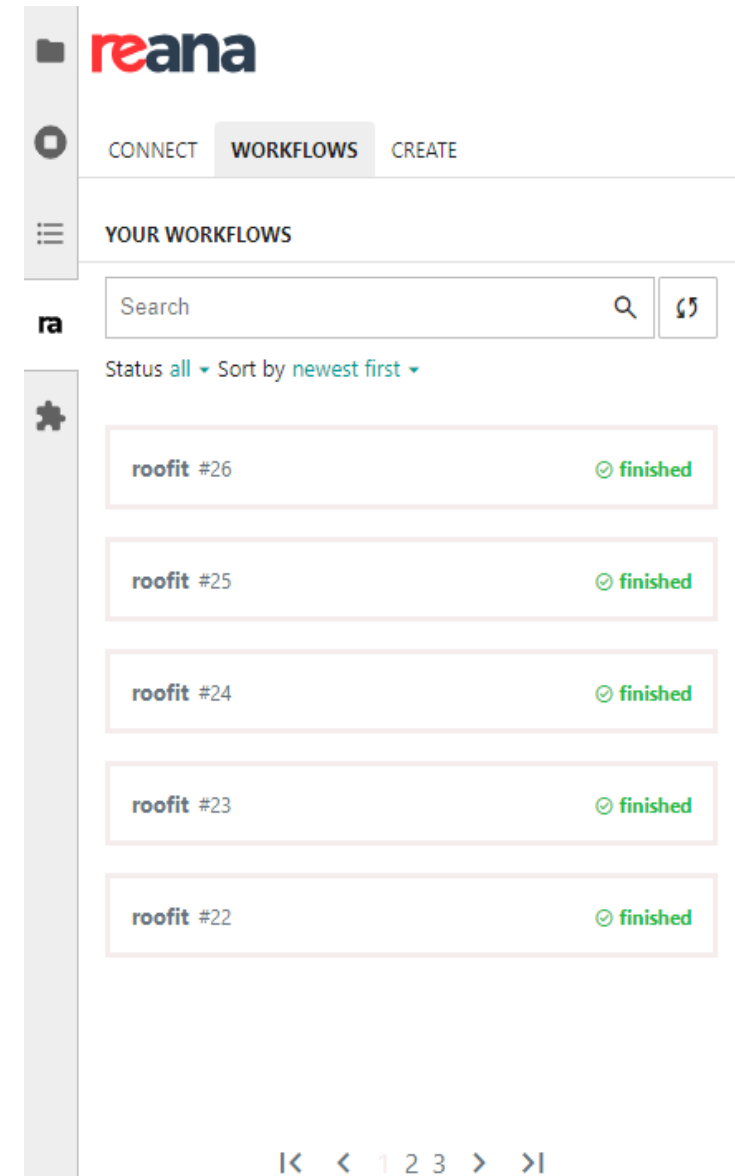
Interact with a workflow

Create workflow from the environment

## UI Requirements

Responsive

User-friendly



The screenshot shows the Reana web interface. At the top, there's a navigation bar with 'CONNECT', 'WORKFLOWS', and 'CREATE' buttons. Below that, a section titled 'YOUR WORKFLOWS' contains a search bar, a status filter set to 'all', and a sort option set to 'newest first'. A list of workflows is displayed, each with a name (e.g., 'roofit #26') and a status icon indicating it is 'finished'. The interface is clean and modern, with a light gray background and blue accents.



# Reana JupyterLab extension

## Main features

Connect to any Reana server ✓

List workflows ✓

Interact with a workflow ✓

Create workflow from the environment

## UI Requirements

Responsive

User-friendly

reana

CONNECT WORKFLOWS CREATE

←

roofit #26 finished in 22 seconds step 2/2  
Finished: 5/8/2024, 13:41:50

ENGINE LOGS JOB LOGS WORKSPACE SPECIFICATION

Step gendata

finished in 5 seconds Kubernetes docker.io/reanahub/reana-env-root6:6.18.04

```
$ mkdir -p results && root -b -q 'code/gendata.C(20000,\"results/data.root\")'
```

job: :

```
-----  
| Welcome to ROOT 6.18/04                https://root.cern |  
|                                     (c) 1995-2019, The ROOT Team |  
| Built for linuxx86_64gcc on Jan 08 2020, 14:10:00 |  
| From tags/v6-18-04@v6-18-04 |  
| Try '.help', '.demo', '.license', '.credits', '.quit'/.q' |  
-----
```

Processing code/gendata.C(20000,\"results/data.root\")...

```
@[1mRooFit v3.60 -- Developed by Wouter Verkerke and David Kirkby@[0m  
Copyright (C) 2000-2013 NIKHEF, University of California &  
All rights reserved, please read http://roofit.sourceforge.
```

```
[#1] INFO:ObjectHandling -- RooWorkspace::import(w) importing RooAddPdf::mc  
[#1] INFO:ObjectHandling -- RooWorkspace::import(w) importing RooChebychev:  
[#1] INFO:ObjectHandling -- RooWorkspace::import(w) importing RooRealVar::>  
[#1] INFO:ObjectHandling -- RooWorkspace::import(w) importing RooRealVar::>  
[#1] INFO:ObjectHandling -- RooWorkspace::import(w) importing RooRealVar::>
```

# Reana JupyterLab extension

## Main features

Connect to any Reana server ✓

List workflows ✓

Interact with a workflow ✓

Create workflow from the environment ✓

## UI Requirements

Responsive

User-friendly

The screenshot displays the Reana JupyterLab extension interface. At the top, there are navigation tabs for 'CONNECT', 'WORKFLOWS', and 'CREATE'. Below this is a section titled 'CREATE A REANA WORKFLOW'. It includes a 'Workflow Name' input field with the value 'roofit' and a 'YAML File' input field with the value 'reana-demo-root6-roofit/reana.yaml'. There are 'Validate' and 'Create & Run' buttons. Below the input fields is an 'OUTPUT' section showing a terminal log with the following content:

```
==> Verifying REANA specification file... /home/ruben/UGR/jupyterlab-extension-tutoria
-> SUCCESS: Valid REANA specification file.
==> Verifying REANA specification parameters...
-> SUCCESS: REANA specification parameters appear valid.
==> Verifying workflow parameters and commands...
-> SUCCESS: Workflow parameters and commands appear valid.
==> Verifying dangerous workflow operations...
-> SUCCESS: Workflow operations appear valid.
```

# Reana JupyterLab extension

## Main features

Connect to any Reana server ✓

List workflows ✓

Interact with a workflow ✓

Create workflow from the environment ✓

## UI Requirements

Responsive ✓

User-friendly

The screenshot displays the Reana JupyterLab extension interface. At the top, there are navigation tabs: 'CONNECT', 'WORKFLOWS', and 'CREATE'. Below these is a section titled 'CREATE A REANA WORKFLOW'. It contains a 'Workflow Name' input field with the value 'roofit' and a 'YAML File' input field with the value 'reana-demo-root6-roofit/reana.yaml'. There are 'Validate' and 'Create & Run' buttons. Below the form is an 'OUTPUT' section showing a terminal log with the following text:

```
==> Verifying REANA specification file... /home/ruben/UGR/jupyterlab-extension-tutoria
-> SUCCESS: Valid REANA specification file.
==> Verifying REANA specification parameters...
-> SUCCESS: REANA specification parameters appear valid.
==> Verifying workflow parameters and commands...
-> SUCCESS: Workflow parameters and commands appear valid.
==> Verifying dangerous workflow operations...
-> SUCCESS: Workflow operations appear valid.
```

# Reana JupyterLab extension

## Main features

Connect to any Reana server ✓

List workflows ✓

Interact with a workflow ✓

Create workflow from the environment ✓

## UI Requirements

Responsive ✓

User-friendly ✓

The screenshot displays the Reana JupyterLab extension interface. At the top, there are navigation tabs: 'CONNECT', 'WORKFLOWS', and 'CREATE'. Below these is a section titled 'CREATE A REANA WORKFLOW'. It contains a 'Workflow Name' input field with the value 'roofit' and a 'YAML File' input field with the value 'reana-demo-root6-roofit/reana.yaml'. There are 'Validate' and 'Create & Run' buttons. Below the form is an 'OUTPUT' section showing a terminal log with the following content:

```
==> Verifying REANA specification file... /home/ruben/UGR/jupyterlab-extension-tutoria
-> SUCCESS: Valid REANA specification file.
==> Verifying REANA specification parameters...
-> SUCCESS: REANA specification parameters appear valid.
==> Verifying workflow parameters and commands...
-> SUCCESS: Workflow parameters and commands appear valid.
==> Verifying dangerous workflow operations...
-> SUCCESS: Workflow operations appear valid.
```

# Reana JupyterLab extension

- **Frontend:**
  - Jupyter lab extension
  - Based on [jupyterlab/extension-template](https://github.com/jupyterlab/extension-template)
  - React, Typescript, CSS, HTML
- **Backend:**
  - Jupyter server extension
  - Python
- **Connection to REANA API:**
  - reana-client (CLI)
  - Reana REST API



## Docker image and Python package available!

```
docker pull ghcr.io/vre-hub/reana-jupyterlab-extension:latest  
pip install reana-jupyterlab
```

# Future Work

**DOCUMENTATION**

**UNIT TESTS**

**MEETINGS**

**MAINTENANCE**

# Thank you!

Questions?



[ruben.perez.mercado@cern.ch](mailto:ruben.perez.mercado@cern.ch)



[rubenpermerc@gmail.com](mailto:rubenpermerc@gmail.com)



[linkedin.com/in/rubenperezmercado](https://www.linkedin.com/in/rubenperezmercado)



[github.com/vre-hub/reana-jupyterlab-extension](https://github.com/vre-hub/reana-jupyterlab-extension)





# Bibliography

1. Gazzarrini, E., Garcia, E., Gosein, D., Moya, A. V., Kounelis, A., & Espinal, X. (2023). The Virtual Research Environment: towards a comprehensive analysis platform. *arXiv preprint arXiv:2305.10166*.
2. Gazzarrini, E., Garcia, E. G., Gosein, D., & Espinal, X. (2024). The Virtual Research Environment: A multi-science analysis platform. In *EPJ Web of Conferences* (Vol. 295, p. 08023). EDP Sciences.
3. Šimko, T., Heinrich, L., Hirvonsalo, H., Kousidis, D., & Rodríguez, D. (2019). REANA: A system for reusable research data analyses. In *EPJ Web of Conferences* (Vol. 214, p. 06034). EDP Sciences.