The VRE

The Virtual Research Environment

ET EIB workshop - Aachen March 10 2024

Today's Agenda

- VRE context
  - ESCAPE & EOSC Future project
  - Scientific context
  - Computing challenges

- VRE components
  - AAI layer
  - User Interface and Notebook service
  - Distributed storage solution: ESCAPE Data Lake infrastructure
  - Distribute storage solutions.

- Demo

Computing challenges become global

Upcoming experiments and Research Infrastructures facing a **change of scale in computing needs**, eg. the HL-LHC at CERN but also in Astronomy and Nuclear Physics.

- New generation of detectors, antennas and telescopes producing big data volumes, requiring massive processing power and support large user communities

Strong commonalities in the main distributed computing areas, eg. Data Management and Access, storage orchestration, Identity management, user analysis, etc.

European Strategy for Particle Physics

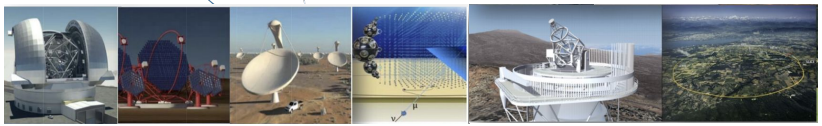
*"**Large-scale data-intensive software** and **computing infrastructures** are an **essential** ingredient to **particle physics research programmes**. The community faces major challenges in this area, notably with a view to the HL-LHC. [...] The community must vigorously pursue **common, coordinated R&D efforts in collaboration with other fields of science and industry,** to **develop software and computing infrastructures** that exploit recent advances in information technology and data science. Further development of internal policies on **open data and data preservation should be encouraged**, and an adequate level of resources invested in their implementation."*

2020 UPDATE OF THE EUROPEAN STRATEGY FOR PARTICLE PHYSICS

ESCAPE project

ESFRI RIs

- Address RI's needs in Data Management, Access and Analysis for **Astro-particle, Radio-astronomy, Gravitational Waves, Cosmology** and **Particle Physics.**
- Provide a fully working **common data infrastructure** "The ESCAPE Data Lake" to test novel data management tools and models, giving the opportunity to influence and steer its development.
- Expand **collaborations** and foster involvement with other Scientific Communities. Maintain and strengthen collaborations with related EC initiatives and projects.
- ESCAPE finished Jan '23 and become an open collaboration [link]



Data centres

ESCAPE and the VRE

- Work done on scientific (re)analyses has been extended in the context of the EU funded **EOSC-Future project** to establish an analysis platform (a *Virtual Research Environment*) for the community.
- Being developed in collaboration and for the scientific community based on **two concrete use cases**, "Science projects":
  - Dark Matter
  - Extreme Universe
- Aim:
  - Create a **proof of concepts** (PoC) to **demonstrate** the **utility** of the **solutions and technologies developed**
  - **Integrate** them into the **EOSC landscape**, as a prototype of an ESCAPE Cell in EOSC.

Common solutions developed within ESCAPE, being integrated in the VRE:

- Data Lake infrastructure
- Software repository
- Analysis platform tools
- Integration of the IVOA
- Outreach and citizen science

EOSC

The ambition of the European Open Science Cloud (EOSC) is to provide European researchers, innovators, companies and citizens with **a federated and open multi-disciplinary environment** where they can **publish, find** and **reuse data**, tools and services **for research, innovation and educational purposes**.

- The five science clusters in Europe are actively collaborating and joining efforts in these common goals

panosc
photon and neutron open science cloud

FAIR ENVRI

SSHOC
social sciences & humanities open cloud

ESCAPE
European Science Cluster of Astronomy & Particle physics ESFRI research Infrastructures

EOSC-Life

VRE challenges

Can we make use of a all these heterogeneous scientific solutions from an unique "entry point"?

How can we provide CERN-born and HEP-specific technologies to the wider communities?

Why do we want to do this ?

- Coordinate efforts to **share with** and **learn from** different communities
  - more people means more manpower & knowledge
- **Combine approaches** towards **common challenges**

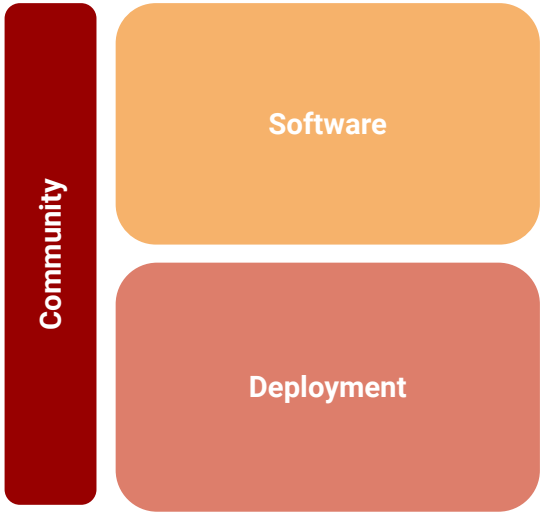
How do we do this?

- **Open Science** initiatives
- **FAIR** principles
- **Data preservation** and **reproducible research/science**

Findable
Accessible
Interoperable
Reusable

Virtual Research Environment

The VRE is an **open source analysis platform** where researchers from various scientific communities have access to all the **digital content** (software, data, computational power) needed to to develop, share and reproduce an **end-to-end scientific result** in compliance with **FAIR** principles.



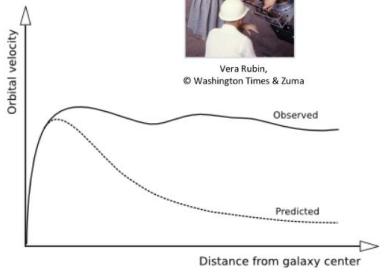
Use case

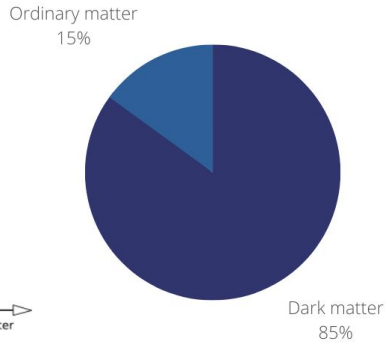
Dark Matter EOSC-Future Science Project

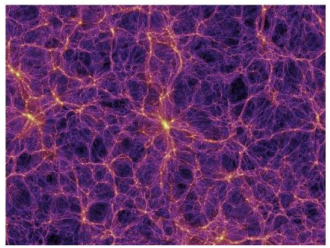
➔ *Galaxy rotation curves* --> a larger amount of gravitational mass is expected to exist in the universe.

➔ It does not interact with the electromagnetic field and *cannot therefore be seen.*

➔ Many DM candidates. Many experiments target the problem. Many different research approaches.







Dark Matter EOSC-Future Science Project

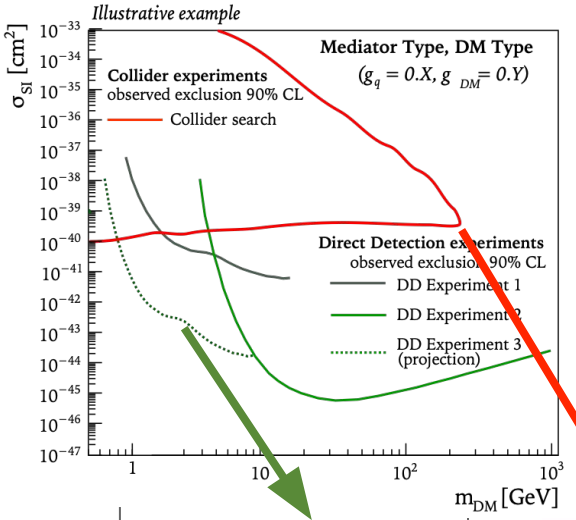
It is the perfect example for a use case, as both HEP and Astrophysics are researching limits of Dark Matter.

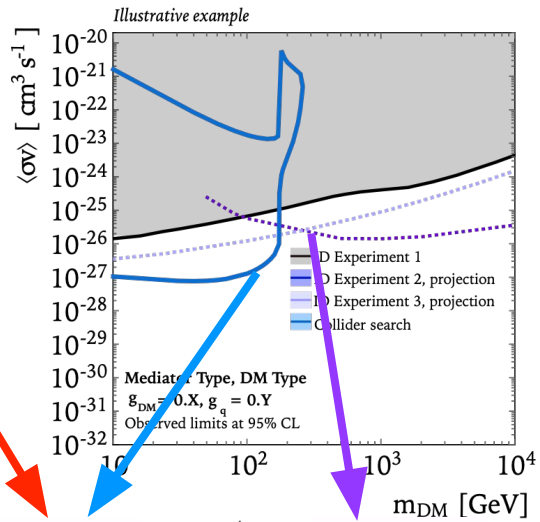
- They need similar infrastructures
- It would be useful to share software and expertise on how to use it

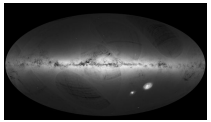
The VRE aim is to provide:

- An **infrastructure** where to run an end-to-end analysis in one place
- A solution to **preserve** the analysis steps in case of re-use
- An **interdisciplinary open science** example from a **bottom-up** effort

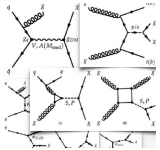
Combined DM plots



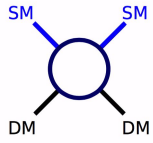




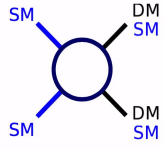
Astrophysics



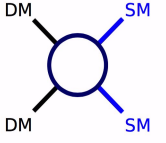
Theory



Direct Detection



Colliders



Indirect Detection

INFN LAPP CERN CCIN2P3 cnrs DARKSIDE ATLAS EXPERIMENT CMS KM3NeT cta cherenkov telescope array Fermi Gamma-ray Space Telescope

An analysis workflow

Data Collection

Experimental data generated at site

Data processing

Data is processed for analysis

Analysis

Run with preserved, versioned software

Result interpretation

Results are combined and compared across workflow

Result preservation

Of data, software, environment and computational steps for later reuse and share

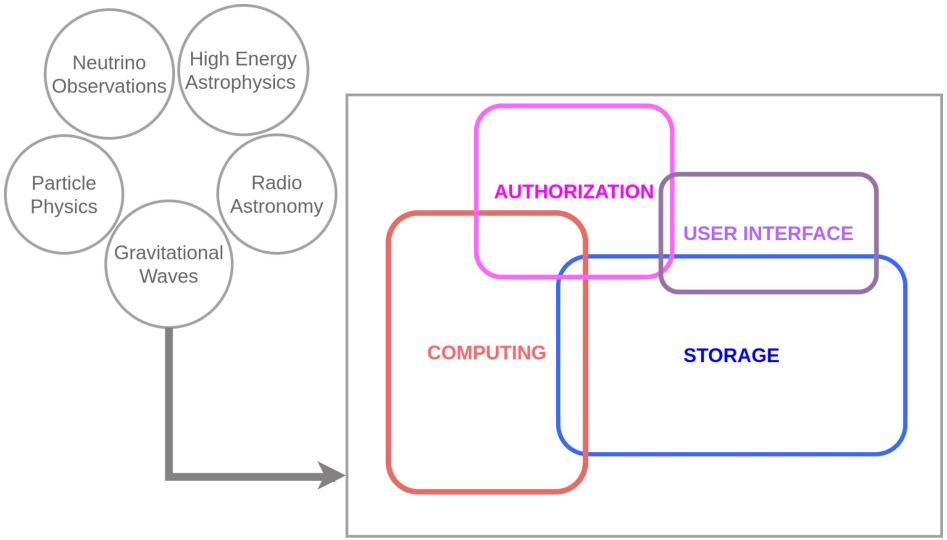
ESFRI RIs

Data Centres

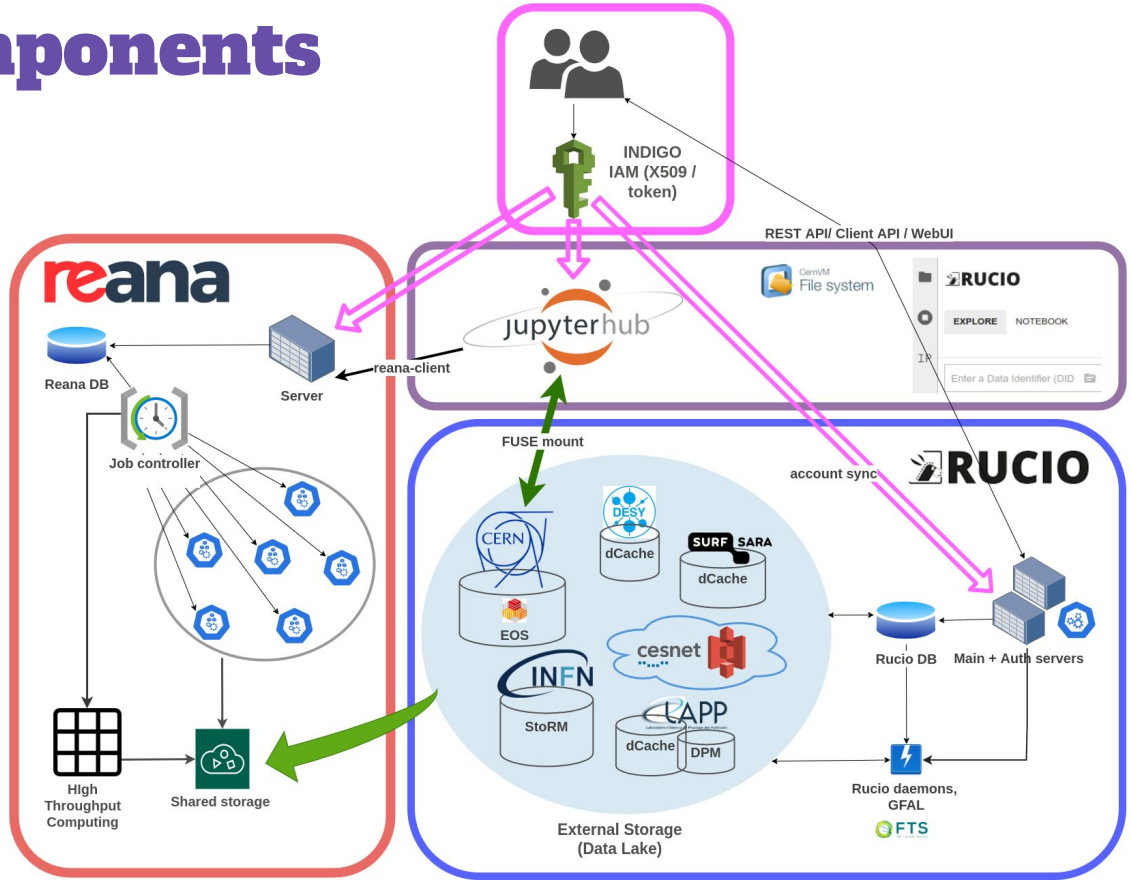
VRE components

VRE components

1. federated **Authentication and Authorization** layer
2. federated distributed **storage** solution
3. **Notebook service** with customizable containerised environments
4. **computing** cluster supplying the processing power

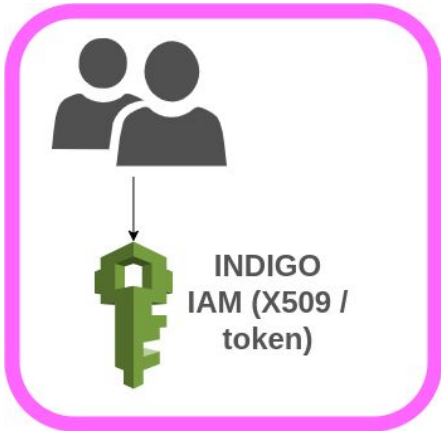


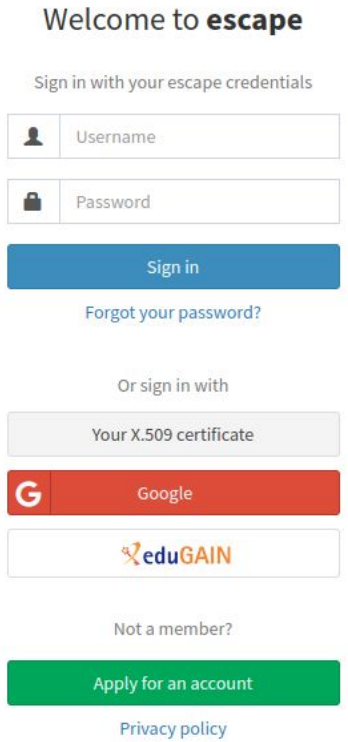
VRE components



1. Authentication and Authorisation

- INDIGO Identity and Access Management (IAM) service
  - adopted by WLCG for token usage
- supports authentication via
  - EduGAIN
  - OIDC tokens
  - X.509 certificates/Virtual Organization Membership Service (VOMS) attribute provisioning services
- Instance deployed on a K8s cluster at INFN-CNAF Bologna
- Rucio and Jupyterhub support IAM
  - Reana's IAM support is under active development





https://iam-escape.cloud.cnaf.infn.it/login

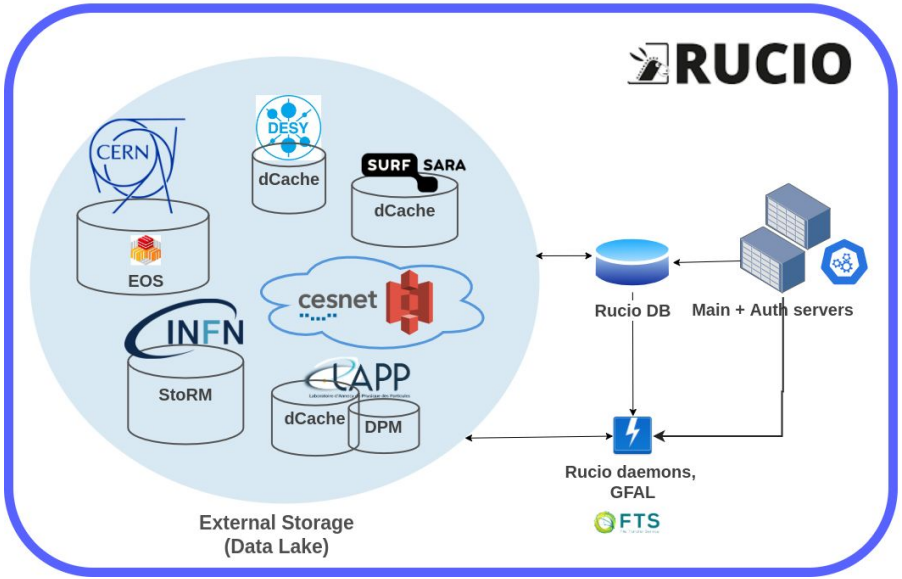
2. Storage: Rucio

https://rucio.cern.ch/

Rucio is an **open-source data management project** initially developed by the ATLAS experiment to manage large volumes of data.
It is now used by various CERN and non-CERN communities.

VRE Rucio instance:

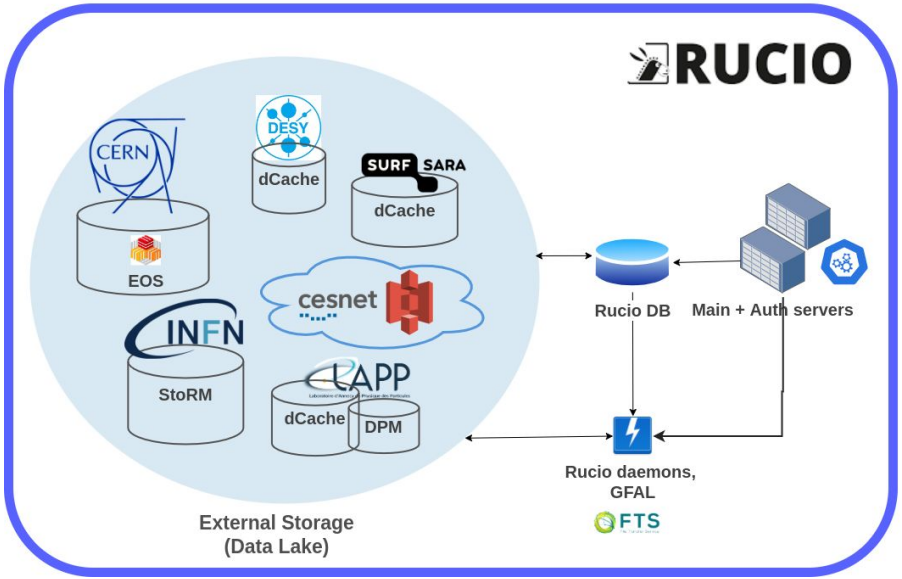
1. **cloud infrastructure**
   a. Rucio servers, daemons and webUI
   b. manages API requests, user authentication, data upload, access, download and replication
2. central relational **database**
   a. backup services in case of major disruptions
3. **Rucio Storage Elements (RSEs)**
   a. Petabyte-scale
   b. Distributed, managed by partner institutions
   c. Support for various storage technologies: **EOS, StoRM, dCache, DPM, XRootD**



2. Storage: Data Lake

The **Data Lake is a policy-driven, reliable, distribute**d data infrastructure able to deliver data **on-demand** at **low latency** to all types of processing facilities.

- ensures data security, quality, access
- Distributed and heterogeneous storage
- File **transfer** with CERN's File Transfer Service (**FTS3**)
- File **upload and download** with CERN's grid file access library (**GFAL2**)
- Support for multiple protocols (**gridFTP, HTTP(S), XRoot, S3**)

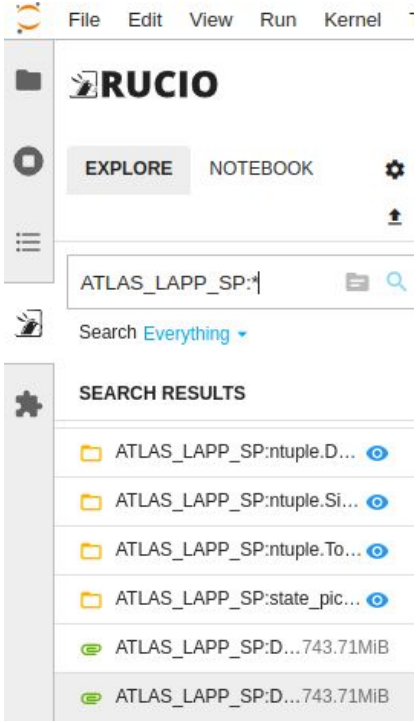


3. Notebook service

Allows users to easily interact with the underlying infrastructure by leveraging **containerised environments**. Used for **preliminary analysis.**

Rucio plug-in:

- Hides complexity of Data Lake – allows browsing Rucio database
- Performs user authentication to Rucio instance via **tokens**
- Allows **data replication** from any storage element to back-end storage element
- Allows to assign a parameter to the data and easily **import it in the notebook**



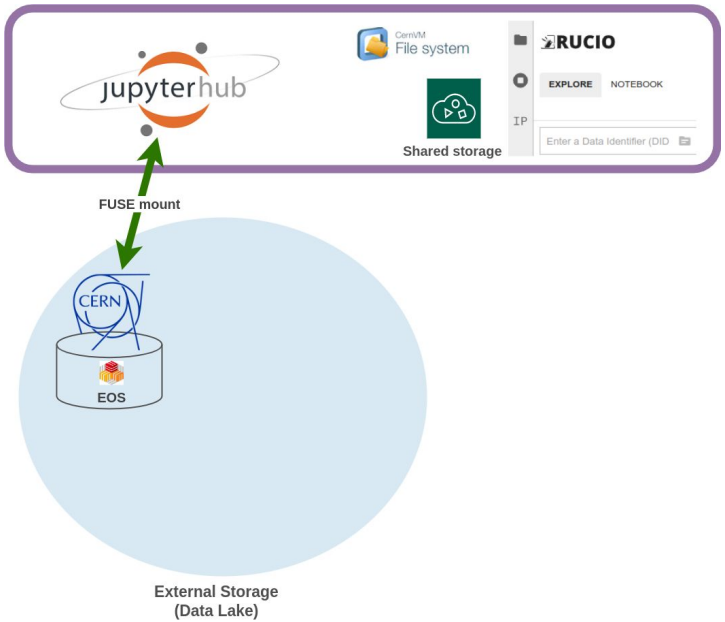
3. Notebook service

External Storage:

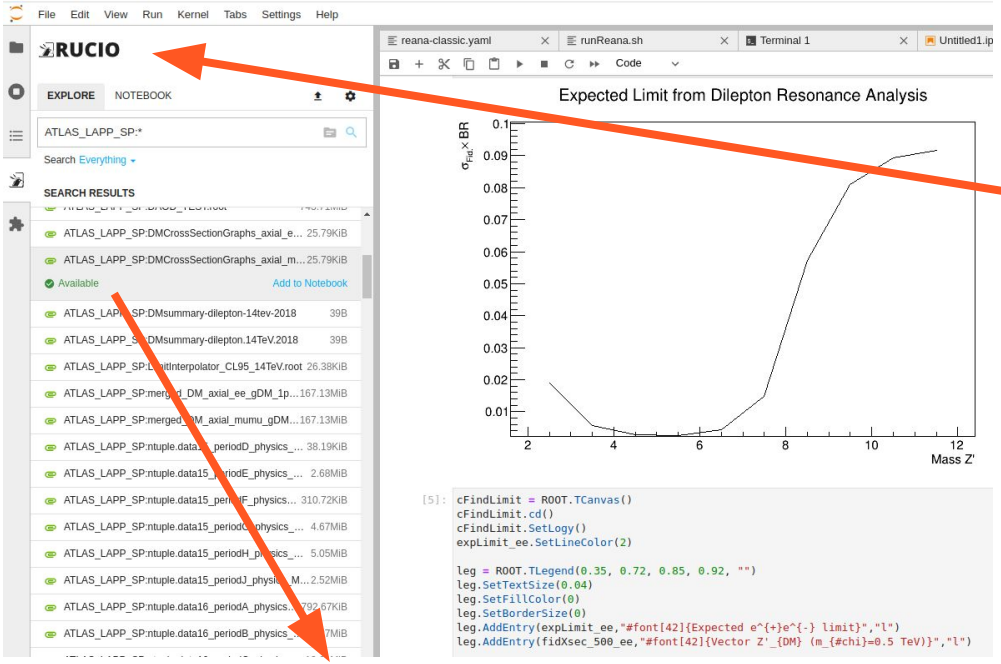
- EOS storage element (RSE) of half petabyte **FUSE mounted** on the JH node
- Data **replication** from the Rucio Data Lake to here when requested for analysis

Other attributes:

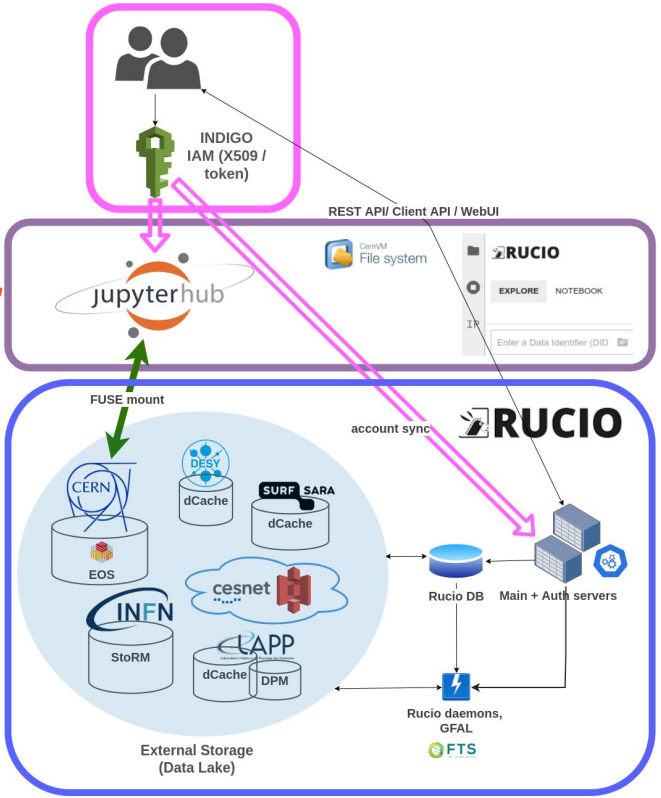
- 800GB **CephFS volume**, to install software, provided as shared, temporary storage solution
- Rucio and Reana libraries installed – interact as a client
- **CERN Virtual Machine FS (CVMFS)** accessible from terminal



3. User Interface: Jupyterhub



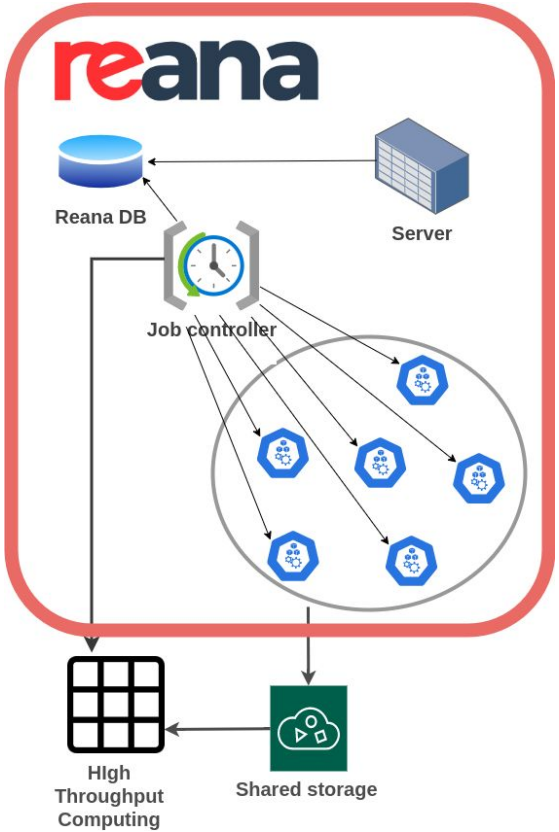
Available on FUSE mounted Rucio Storage Element



4. Computing: Reana

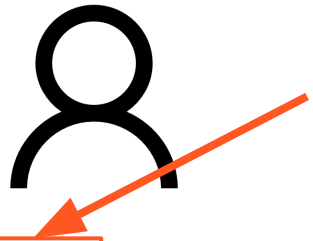
Reana is a **reproducible analysis** project developed at CERN, to make the preservation of an analysis seamless.

- Used for heavier analyses, it allows workflow distribution on various **computing back-ends**
  - **K8s (default)**
  - **HTCondor**
  - **Slurm**
- Intuitive declarative programming approach (`reana.yaml` file)
- Supports workflow engines (CWL, Snakemake, Yadage)
- Installed on the VRE cloud infrastructure via Helm



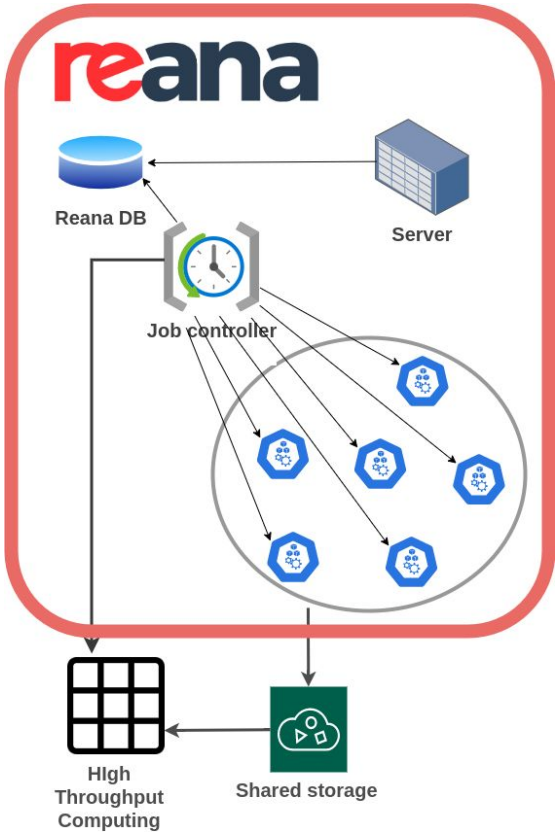
https://reanahub.io/

4. Computing: Reana



fetched locally

1. input data and parameters
2. code
3. computing environments
4. computational steps

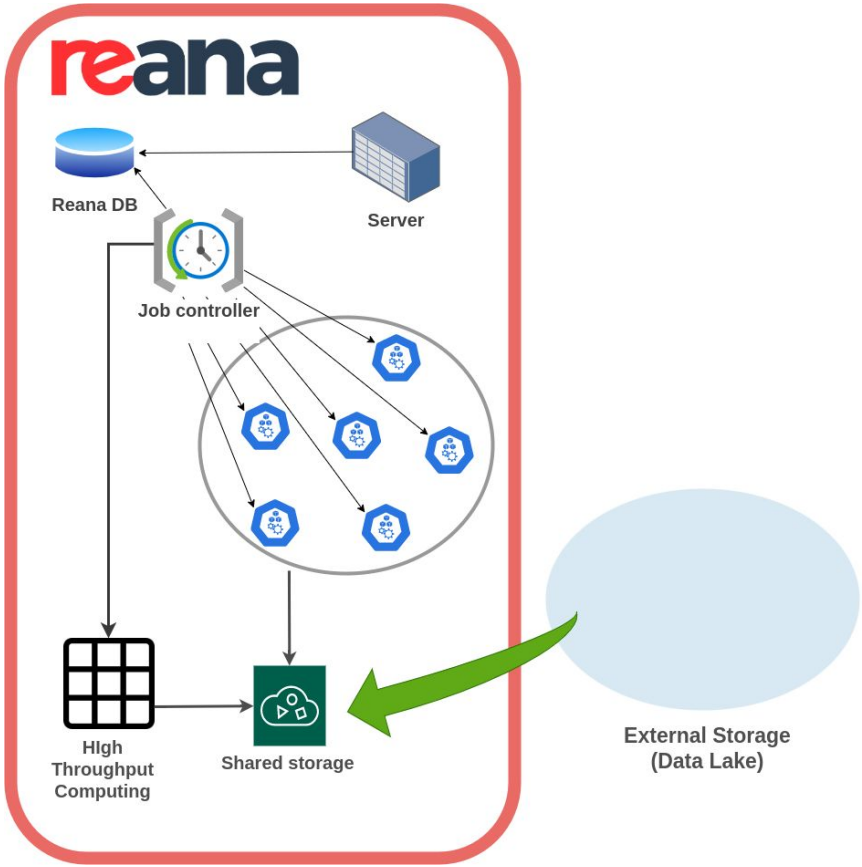


Reana + Rucio

Summer project

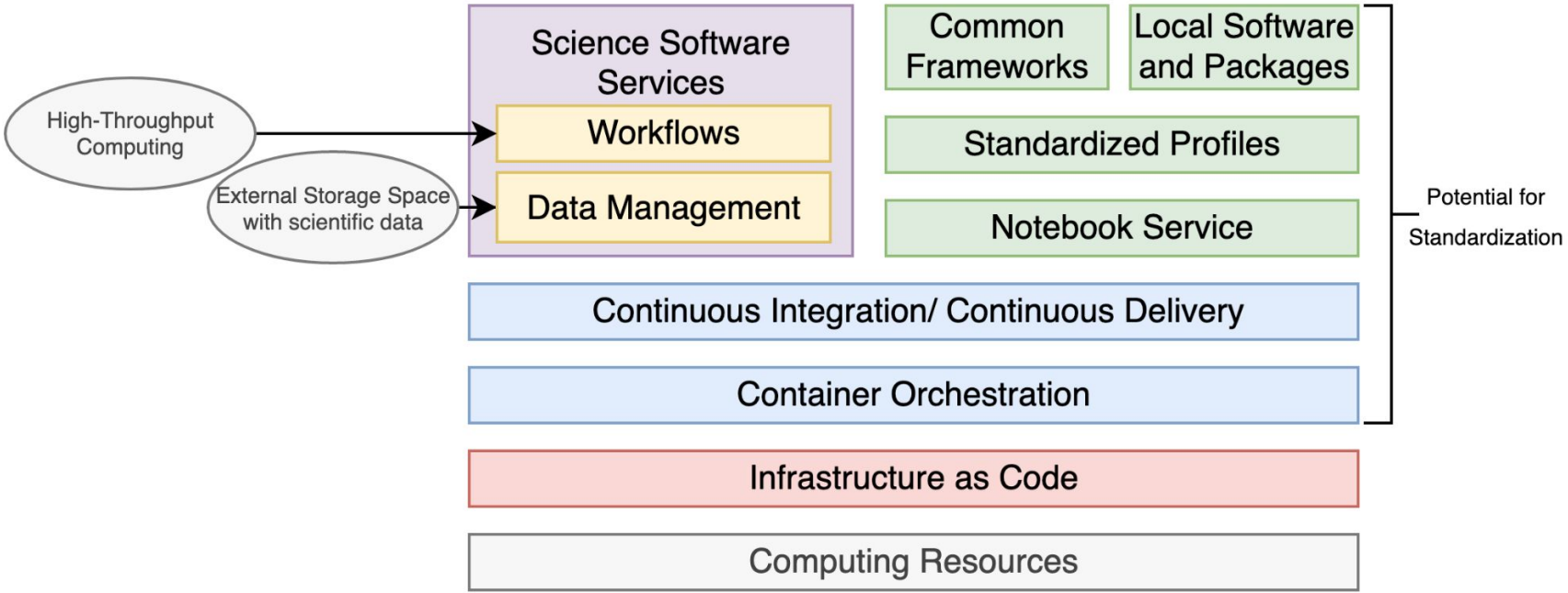
Implementation of **direct download** from Rucio Data Lake to the Reana shared storage via a side-car container that allows authentication to the VRE Rucio instance.

➜ the analysis can be reproduced fully and **independently from local storage.**



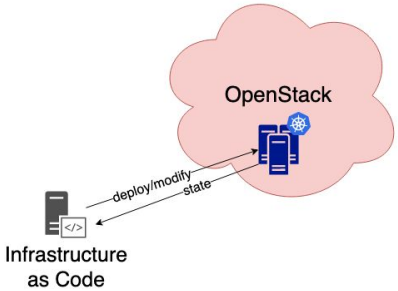
Technical Aspects & Deployment

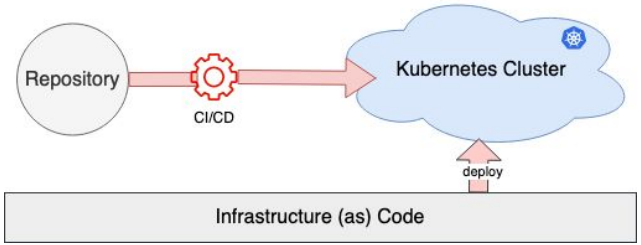
IT Architecture Overview

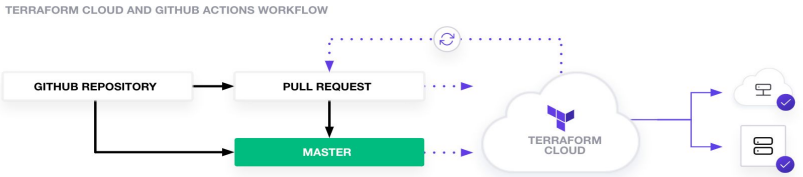


IT Architecture Overview

- Resources provided by CERN OpenStack
  - Computing - Container orchestrator (K8s)
  - Storage
  - Networking
- Management of Infrastructure
  - Terraform - Infrastructure as code (IaC)
  - Services installed via Helm Charts
  - Infrastructure keep updated via GitHub Actions (+ Gitlab CI/CD)
    - Migration to a single repository
  - Sealed secrets
  - CERN Data Base On Demand (DBOD)



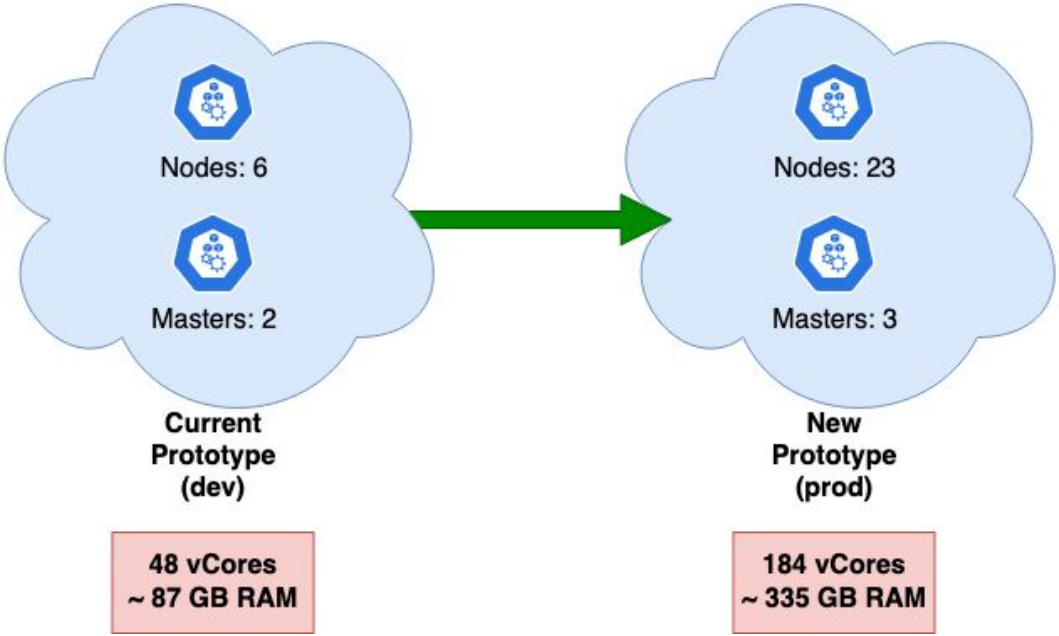




Source: https://developer.hashicorp.com/terraform/tutorials/automation/github-actions

The Cluster(s)

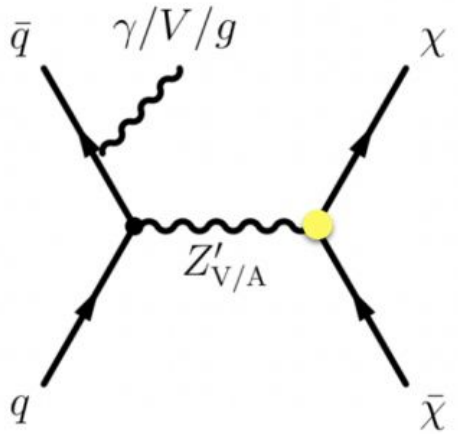
- One Prototype cluster currently in use with **~220 registered users**
- Improved and more powerful cluster under development

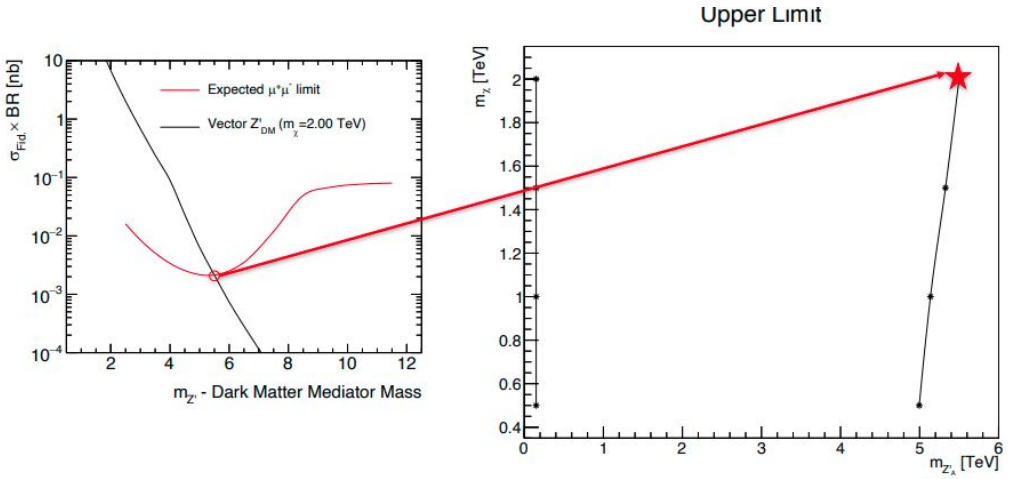


Demo

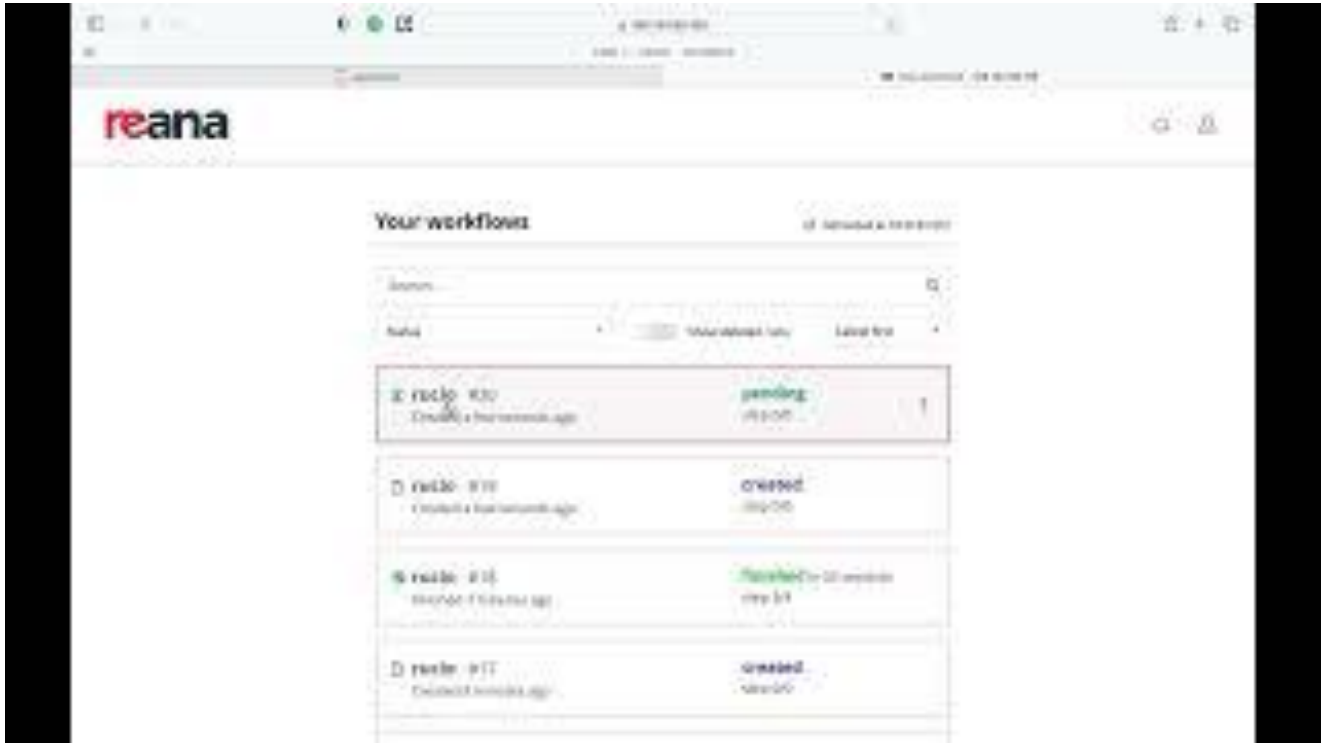
Dark Matter Dilepton Reinterpretation analysis

To set **limits** on High-Luminosity LHC constraints on **Z' → χχ** and project them to 14 TeV to compute the fiducial cross-sections in **lower mass regions.**





Demo Video



We invite you to try the analysis platform!

The current prototype is available here:

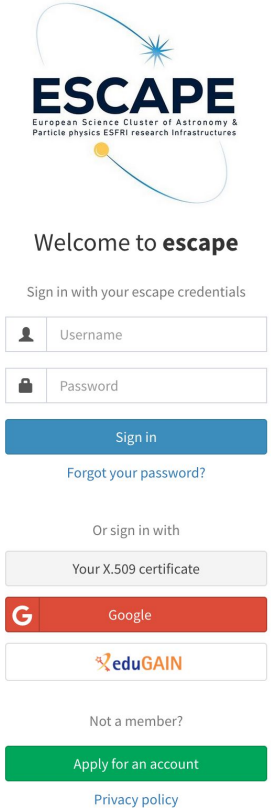
https://escape-notebook.cern.ch

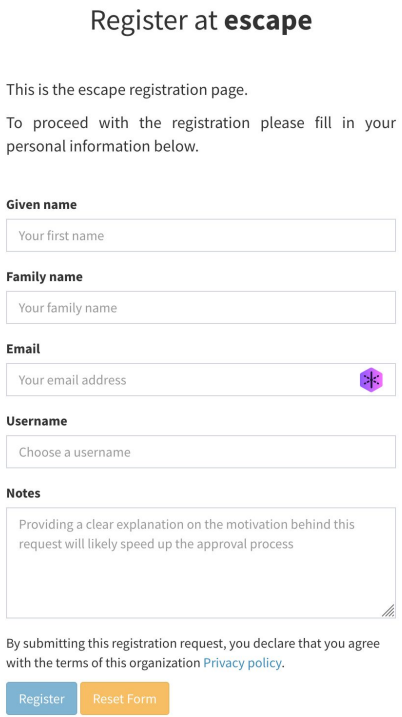
together with the documentation here:

https://datalake-rucio.docs.cern.ch/

Register a new account with "*ET-EIB workshop*" in the *Notes* field.

It is still under development and subject to change!





In summary the VRE is..

.. an analysis **platform as a whole**, build from the bottom up! It **integrates important scientific software**, tools, and packages and offers a **common entry point with the same authentication** for all components. The **deployment is kept simple and is extensively documented** so it can be used by other institutes as a **blueprint**.

The platform is built **CERN independent**, with shared **European resources** and access to **data from multiple experiments also outside of HEP**. As a European effort, it is **accessible to people from other institutes** as well.

Thank you for your attention!

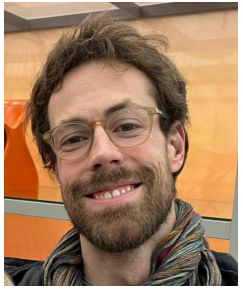
Where to find us: *CERN Meyrin site, room 513-1-014*

Contact: *vre-admin@cern.ch*

VRE team

Elena Gazzarrini
CERN Fellow
-
King's College London
Solid state physics background

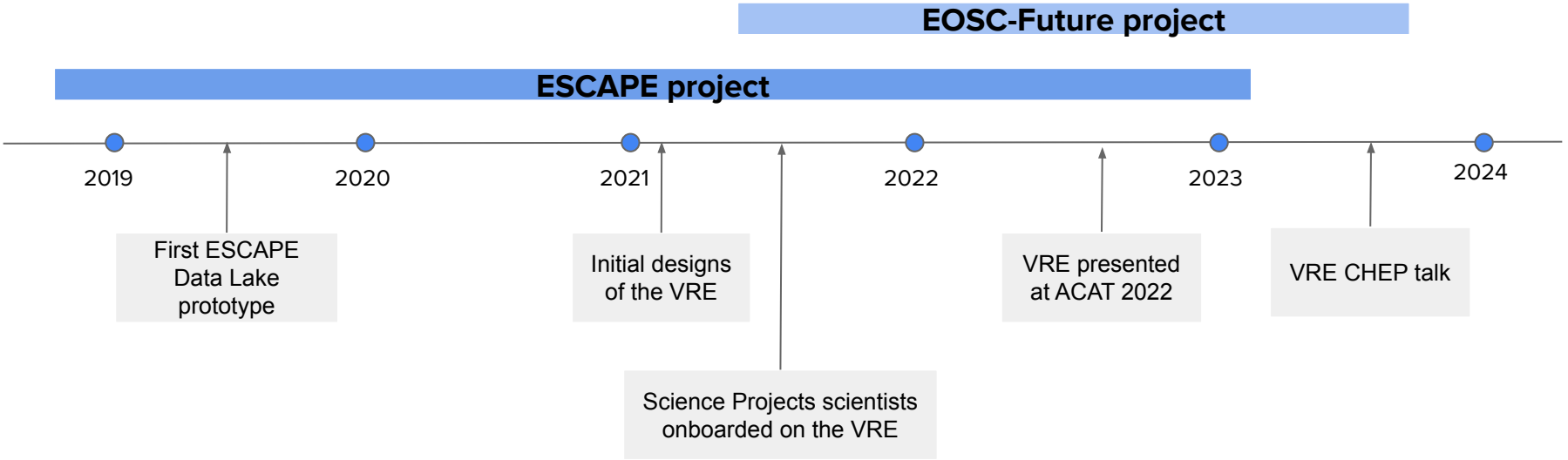
Domenic Gosein
Technical Student
-
Mannheim University of Applied sciences
Computer Scientist



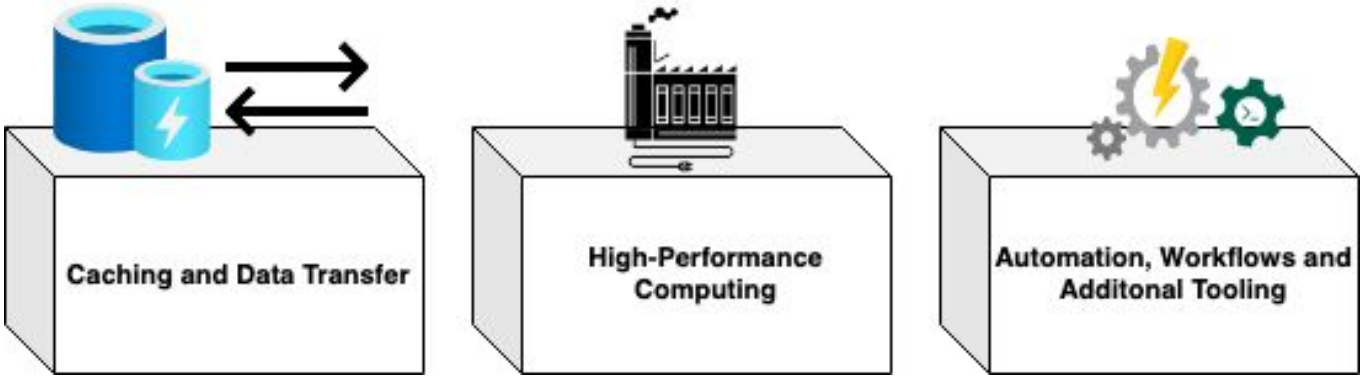
Enrique Garcia
CERN Fellow
-
Univ. Autónoma de Madrid
Astrophysicist and software developer background

Back-up slides

Roadmap and timeline



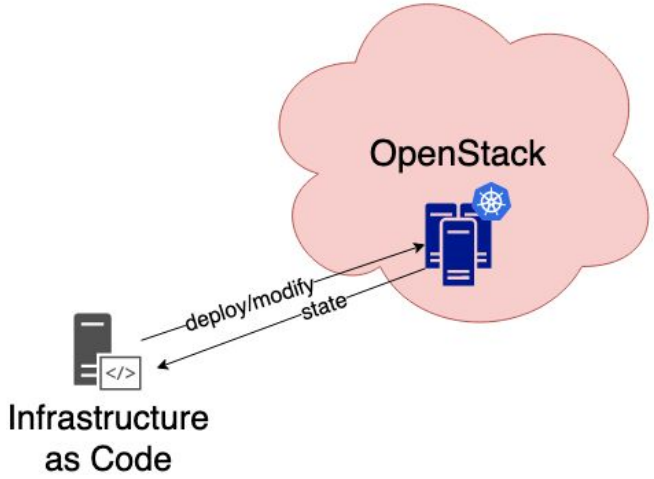
Future Development



Infrastructure as Code

Infrastructure as Code: **terraform**

```
resource "openstack_containerinfra_cluster_v1" "openstack_cluster" {
  name               = var.cluster-name
  cluster_template_id = data.openstack_containerinfra_clustertemplate_v1.cluster_template.id # 22a4c77f-cfe3-47bb-8006-31d02375a3f3
  master_count       = 3
  node_count         = 23
  keypair            = var.cluster-keypair-name
  merge_labels       = true
  flavor             = "m2.xlarge"
  master_flavor      = "m2.large"
  labels = {
    cern_enabled             = "true"
    cvmfs_enabled            = "true"
    cvmfs_storage_driver     = "true"
    eos_enabled              = "true"
    monitoring_enabled       = "true"
    metrics_server_enabled   = "true"
    ingress_controller       = "nginx"
    logging_producer         = var.logging-producer
    logging_installer        = "helm"
    logging_include_internal = "true"
    grafana_admin_passwd     = "admin"
    keystone_auth_enabled    = "true"
    auto_scaling_enabled     = "true"
    min_node_count           = "4"
    max_node_count           = "23"
  }
  provisioner "local-exec" {
    command = "sh ../../scripts/post_cluster_setup.sh"
    environment = {
      cluster = var.cluster-name
    }
  }
}

resource "openstack_sharedfilesystem_share_v2" "share_1_reana" {
  name        = var.reana-share-name
  description = "Share for reana"
  share_proto = "CEPHFS"
  size        = 1000
  share_type  = var.cephfs-type
}
```




```
    required_providers {
      openstack = {
        source  = "terraform-provider-openstack/openstack"
        version = "1.49.0"
      }
      kubernetes = {
        source  = "hashicorp/kubernetes"
        version = "2.16.0"
      }
      helm = {
        source  = "hashicorp/helm"
        version = "2.8.0"
      }
    }
  }
```

Cluster Modification with Scripts

```bash
#!/bin/bash

# post setup cluster configuration --> alternatively also use tf with k8s provider to label resources and get nodes through kubernetes_resource data

# get cluster config for kubectl
$(openstack coe cluster config $cluster)
export KUBECONFIG="config"

# extract node prefix
NODE_PREFIX=$(kubectl get nodes -l magnum.openstack.org/role=worker --sort-by .metadata.name -o name | head -n 1)
NODE_PREFIX=${NODE_PREFIX%-0}
echo $NODE_PREFIX

openstack loadbalancer set --description "vre-rucio.cern.ch" $LB_ID_MAIN
openstack loadbalancer set --description "vre-rucio-auth.cern.ch" $LB_ID_AUTH

## set reana HA node labels
kubectl label "${NODE_PREFIX}-3" reana.io/system=infrastructure
kubectl label "${NODE_PREFIX}-4" reana.io/system=runtimebatch
kubectl label "${NODE_PREFIX}-5" reana.io/system=runtimejobs
kubectl label "${NODE_PREFIX}-6" reana.io/system=runtimesessions
```

```
    provisioner "local-exec" {
      command = "sh ../../scripts/post_cluster_setup.sh"
      environment = {
        cluster = var.cluster-name
      }
    }
```

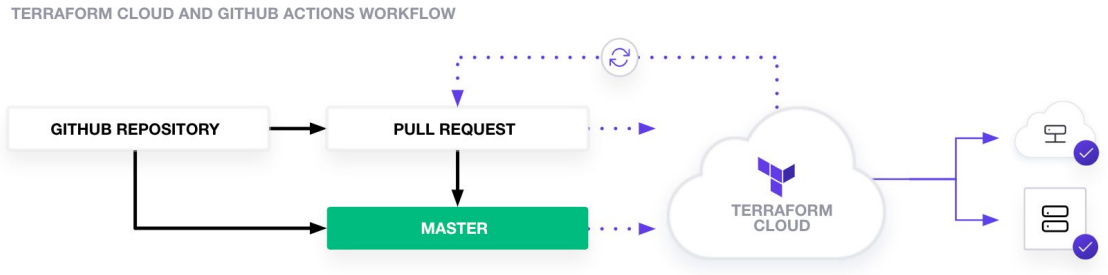
Terraform with Helm and Kubernetes

```
73  resource "helm_release" "jupyterhub-chart" {
74    name       = "jhub-${var.resource-suffix}"
75    repository = "https://jupyterhub.github.io/helm-chart/"
76    chart      = "jupyterhub"
77    version    = "2.0.0"
78    namespace  = var.ns-jupyterhub
79
80    values = [
81      "${file("jhub/config.yaml")}"
82    ]
83
84    set {
85      name  = "hub.db.url"
86      value = data.kubernetes_secret_v1.jhub_db_secret.data.dbconnectstring
87    }
88    set {
89      name  = "hub.config.GenericOAuthenticator.client_id"
90      value = data.kubernetes_secret_v1.jhub_iam_secret.data.client_id
91    }
92    set {
93      name  = "hub.config.GenericOAuthenticator.client_secret"
94      value = data.kubernetes_secret_v1.jhub_iam_secret.data.client_secret
95    }
96  }
```

```
68  resource "kubernetes_storage_class_v1" "sc_manila-meyrin-cephfs" {
69    metadata {
70      name = "manila-meyrin-cephfs" # ref.: https://kubernetes.docs.cern.ch/docs/storage/fileshares/
71    }
72    storage_provisioner  = "cephfs.manila.csi.openstack.org"
73    reclaim_policy       = "Delete"
74    allow_volume_expansion = true
75    parameters = {
76      type                                                = "Meyrin CephFS" # ref.: https://clouddocs.web.cern.ch
    /file_shares/share_types.html
77      "csi.storage.k8s.io/provisioner-secret-name"            = "os-trustee"
78      "csi.storage.k8s.io/provisioner-secret-namespace"       = "kube-system"
79      "csi.storage.k8s.io/controller-expand-secret-name"      = "os-trustee"
80      "csi.storage.k8s.io/controller-expand-secret-namespace" = "kube-system"
81      "csi.storage.k8s.io/node-stage-secret-name"             = "os-trustee"
82      "csi.storage.k8s.io/node-stage-secret-namespace"        = "kube-system"
83      "csi.storage.k8s.io/node-publish-secret-name"           = "os-trustee"
84      "csi.storage.k8s.io/node-publish-secret-namespace"      = "kube-system"
85    }
86  }
```

Terraform Automation with GH Actions

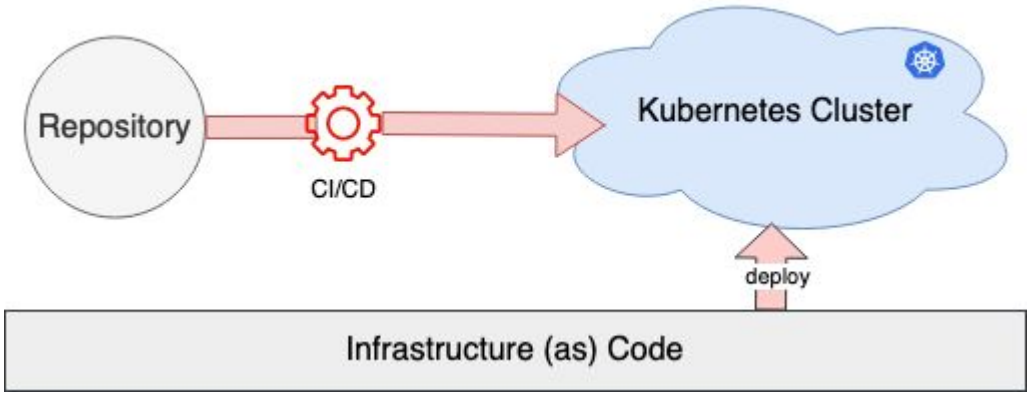
```yaml
terraform:
  name: "Terraform"
  runs-on: ubuntu-latest
  permissions:
    pull-requests: write
  steps:
    - name: Checkout
      uses: actions/checkout@v3

    - name: Setup Terraform
      uses: hashicorp/setup-terraform@v1
      with:
        # terraform_version: 0.13.0:
        cli_config_credentials_token: ${{ secrets.TF_API_TOKEN }}

    - name: Terraform Format
      id: fmt
      run: terraform fmt -check

    - name: Terraform Init
      id: init
      run: terraform init

    - name: Terraform Validate
      id: validate
      run: terraform validate -no-color

    - name: Terraform Plan
      id: plan
      if: github.event_name == 'pull_request'
      run: terraform plan -no-color -input=false
      continue-on-error: true
```



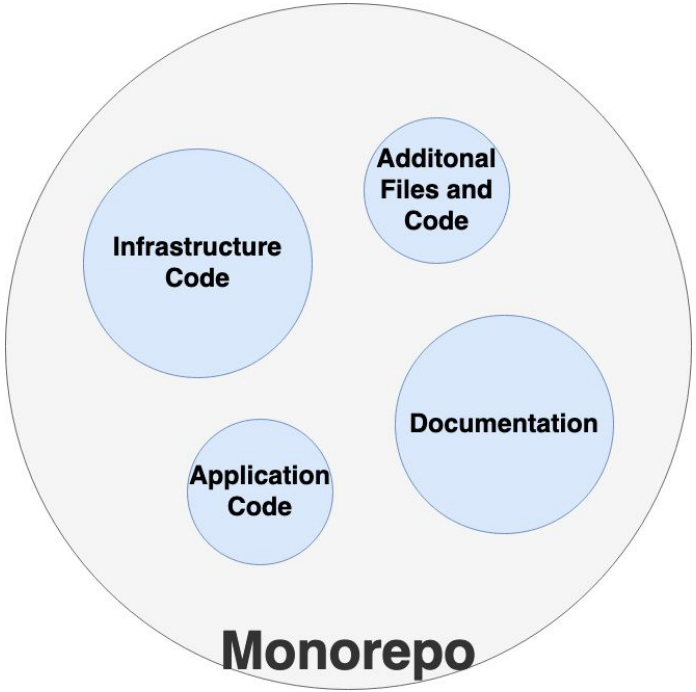
Source: https://developer.hashicorp.com/terraform/tutorials/automation/github-actions

Additional CI/CD

- IaC for cloud resources
- GitOps Tools additionally for K8s Manifests and Applications
- GitHub Actions for other workflows like the image registry



Code Management and Documentation



main · 3 branches · 0 tags · Go to file · Add file · Code

egazzarr Merge pull request #58 from cern-vre/rucio-dev … 23c4428 2 weeks ago 191 commits

.github/workflows	Update merge-check.yml	2 weeks ago
iac	necessary 'main' of https://github.com/cern-vre/cern-vre into rucio-...	2 weeks ago
.gitignore	change secrets dir, mod gitignore and add tmp dir	last month
README.md	Update README.md	2 weeks ago
cern-vre-logo.png	update logo and rm (#28)	last month

README.md

This repository is still under construction

Merge Check passing

CERN VRE

Designed with freelogodesign

The CERN Virtual Research Environment.

Please find the technical documentation for this repository here CERN VRE Wiki.

Home

egazzarr edited this page last week · 14 revisions

Edit · New page

CERN VRE Technical Documentation

The following is the technical documentation of the CERN Virtual Research Environment, including CERN-specific matters, which are marked accordingly.

TL;TR

In order to create or interact with this cluster you'll need: `kubectl`, `terraform`, `helm` and `kubeseal` installed locally.

Further, you'll need the OpenStack RC configuration and the kubeconfig (once the cluster has been created).

Repository Structure

The repo is designed to be a Monorepo, containing all relevant files to this project:

- `.github` : Workflows
- `adm` additonal configuration files
- `iac` : Infrastructure as Code
  - `scripts` : Scripts used for the infrastructure
  - `secrets` : sealed secrets
    - `tmp_local_secrets` : not tracked dir for local secrets to work with the scripts
  - `tf` : Terraform files
    - `cluster` : Cluster main files
    - `modules` : Terraform modules used in the cluster

CERN VRE Technical Documentation ©CERN 2023

Pages

Find a page...

Home
- CERN VRE Technical Documentation
- TL;TR
- Repository Structure

Components

Developer Setup

Infrastructure

Add a custom sidebar

Clone this wiki locally

https://github.com/cern-vre/cer

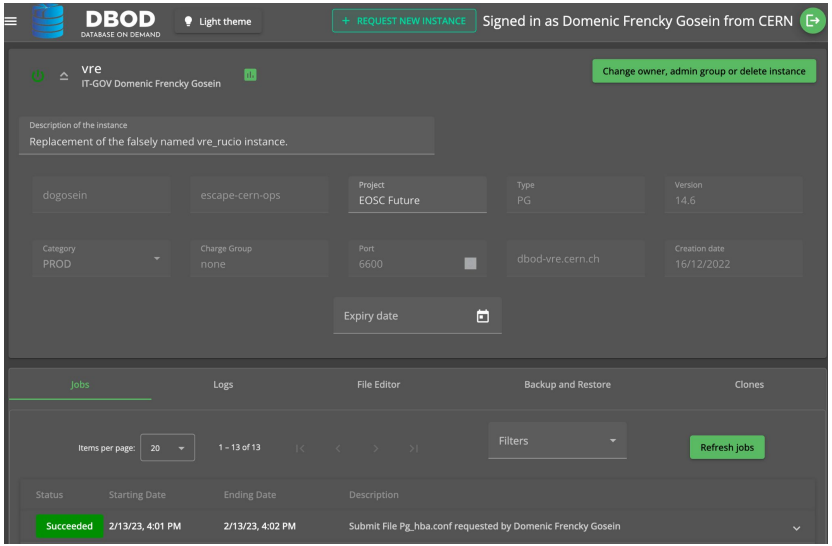
Secrets Management: Sealed Secrets

- Controller installed in the cluster
  - Has the key
  - Creates the K8s secret
- Client side utility to encrypt values and create the custom resource “SealedSecret”

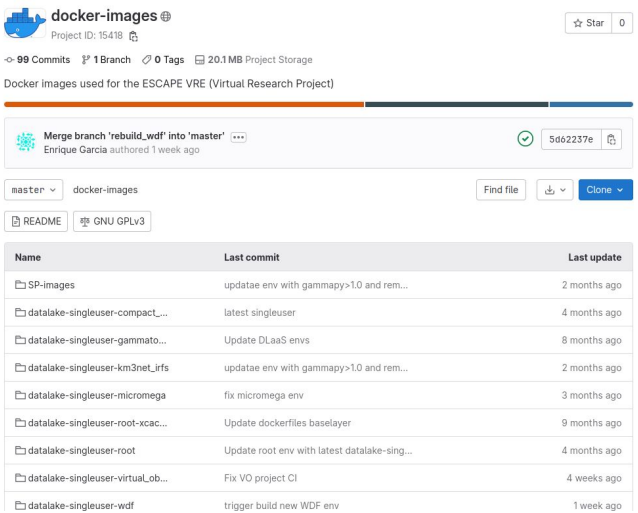
```yaml
apiVersion: bitnami.com/v1alpha1
kind: SealedSecret
metadata:
  creationTimestamp: null
  name: jhub-cvre-iam-secrets
  namespace: jhub
spec:
  encryptedData:
    client_id: AgB40MgVJ6LSkkXsfBSQ9Ple5tiyXVqUu76sc39bEzwUXvhjZzrkdvlW0X8bzMaX66mKwwesla0CNLjO0rjLBdZ8cAgVBX/Kpxo7in/DPfRJIk
/RAOo88X8C2GtcQzV6vKXTgRM+ZOadx+cRJMLAoGmg5jSaYmR7NlCy532pEtJ3FWZ81mqqCAF9+S1oOLukGdwF+7ihz/bVEVu0pfOkjAP9oDdl9PBXvl9lhe6Udj5
/q3cEbotoCuWvWOJH1+ygfWeDLps3rc
/ah613FZYjf+HmekOfiPOfg1xXzxHrq8dvSnFkLaVe3YNyni5CJVutkooGjxZs9n6Ucw4XufxGFSf3Ho6mGpkrsMmX9XlYoao4A1ItLA6c46YFCv2nLaDpRTAfgzH
/pNYyNrPERYNgFXoUyC/+aTGcfzliY75f0F0Hkp+hxuvqkcZhifesE+lxgxLtThLJjGDLpIWcDMMuhiDcbF9vaCwe6cQjQEF8BG6SWZ+2dxXM8T1Yzk2cWs/XqJlp
/eO49smycjlHsYkebMB0KOWpx0opSXhjAm8BLgL7vaPKgbkZfTE19Ks9xZZnCZPeiIvXfuKmwAOTbvbOe8GILasSBJjQ/Y8Yd+78TgSsxDH8eOIQF/JCdZUDemXLz
    client_secret: AgBmvm2fe9eJ79Pn1+NEVadUhjAxdVy2TuPVk25Tz9vrK1uuQ/euMCmhoZjzsE5lYWc46AXv
/d05aE052RB8ZErIH4jEJzEK2c+b+Lb3guO4Nneh7wkG1hsVYHxiTKZomM71CjE3vud7fR1+YoDEc0kdZCiHI8Ygz+g6Y7yQvzAZSAhc75h0GakJQNrksmzLP92vs
/v4TdploHWSFZ6PWJyJOXTPrEBQgOGFjaywKAowrR+iqm8pE0Q4NTFxHXFY5eptVu4Mv74d8vc6wXATHkreO0c5pw8nVornXh9gOnjXXkpwcDL0PKgxQ9uBg1Ny6j
/kQL1jo7psyz21UX5pCcbVeWlvR6zjGYj8VP3syd0XuvqY4N5tkaqF3US8F9MAQZzLVhLnneoprWbCMCQVP+e3mPO5kLij5Uf1HlhSZfj7ekRjL0MgmhmAiF9iunr
/31nHTo1FDyuXV6ZmYHfvRzjP2cj5jeRvyo2GW+ChFCK5jnW6JdFanUoH9h5UK3LOle04wfH73cBUtt6sA4V52DCPFFmJ8YtYL0DNXcIEp7lR+dJM4
/V3VMAeladZrr4ywM40nA5boU3t23gCVlq31ODR6ia8NtCi+DvEtCX/S8H/MN0WN8ch6o7pZFV/BEc9XINrPjvpSDPC/SjNf27zzqIVR
/DwrwhPA6uTMkurhrkFPu5+AFcbQ3FuMycJCB9aqWwVzhn1Cg/EQoxM0iHMpMUmeEpk9vu6T7nbEzTVwRIYYiPnKs0CoGO7kkjTVmXpyQVD+Dvpo9PdWnZn1r9YN2
  template:
    metadata:
      creationTimestamp: null
      name: jhub-cvre-iam-secrets
      namespace: jhub

```

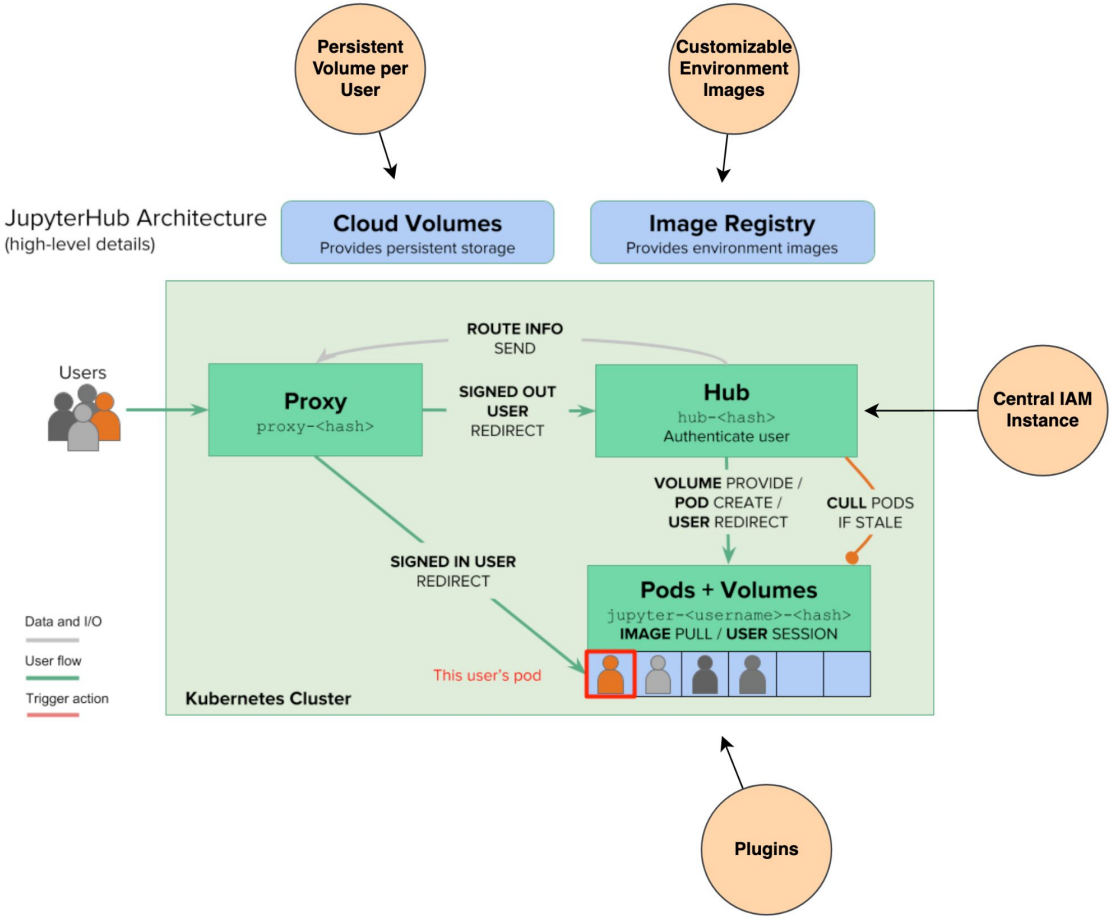
Database

- CERN Database on Demand Instance
- Postgres



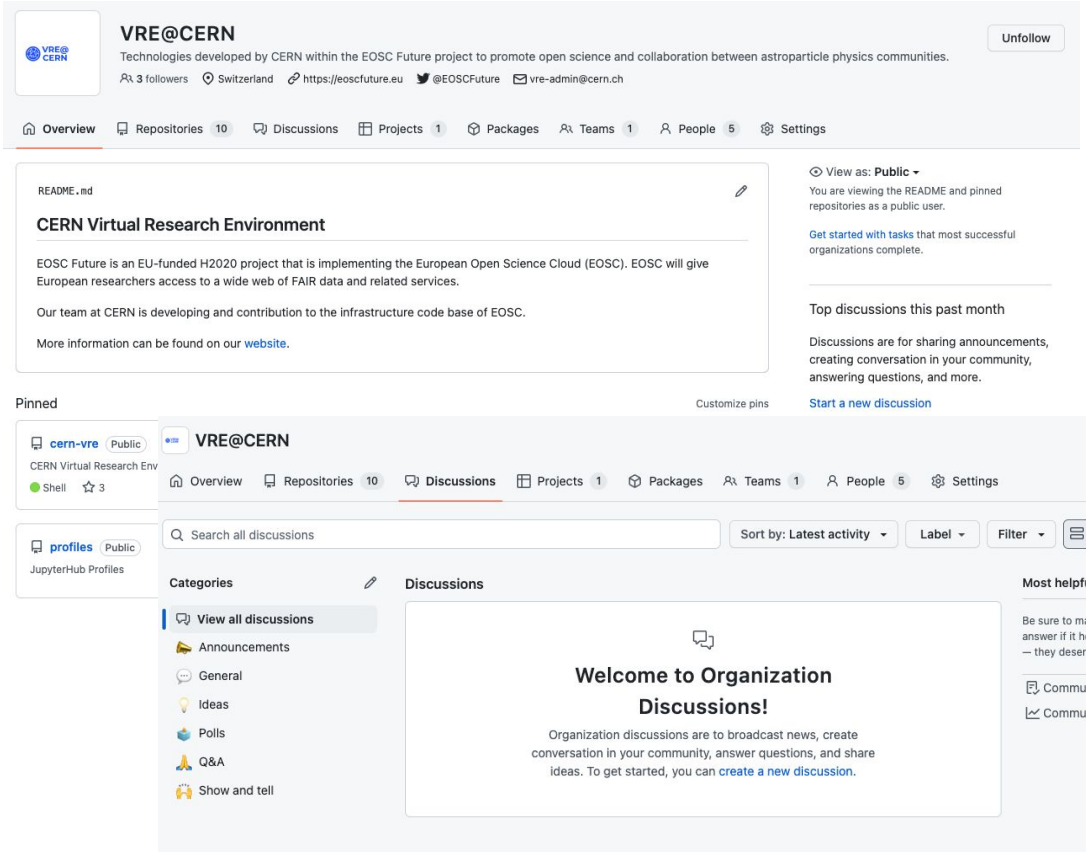
Core Notebook Service (JupyterHub)



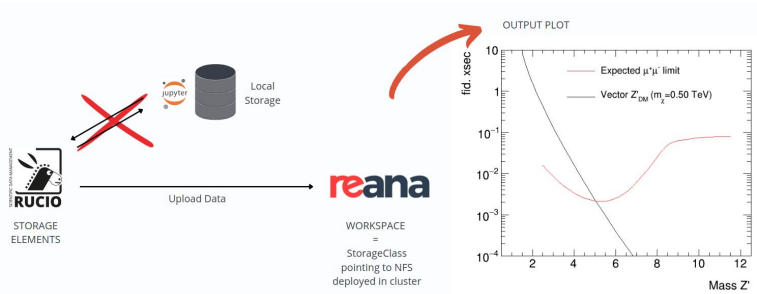


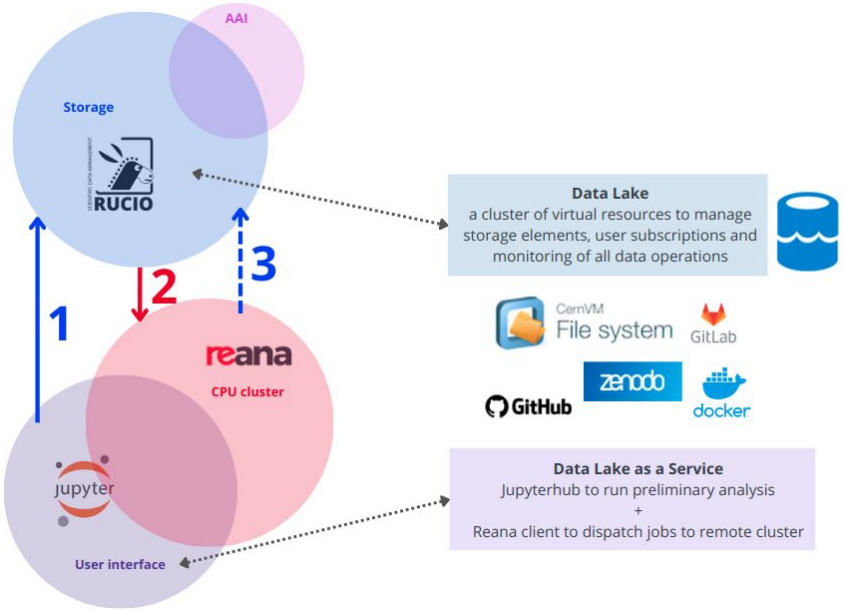
Source: https://z2jh.jupyter.org/en/latest/_images/architecture.png (modified)

Community

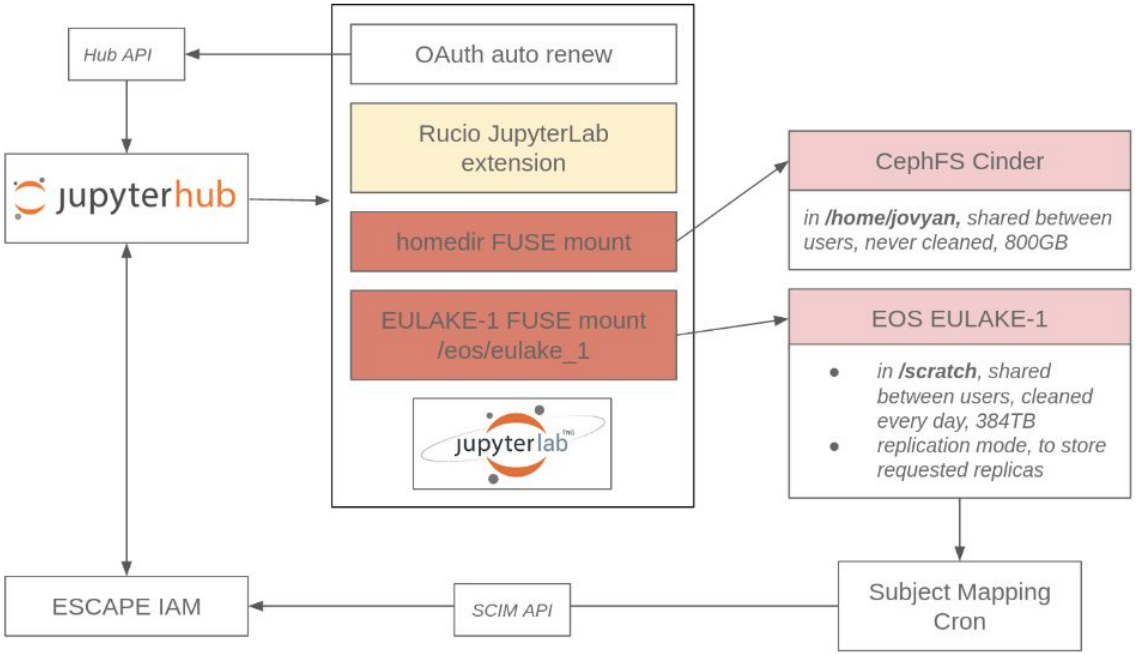
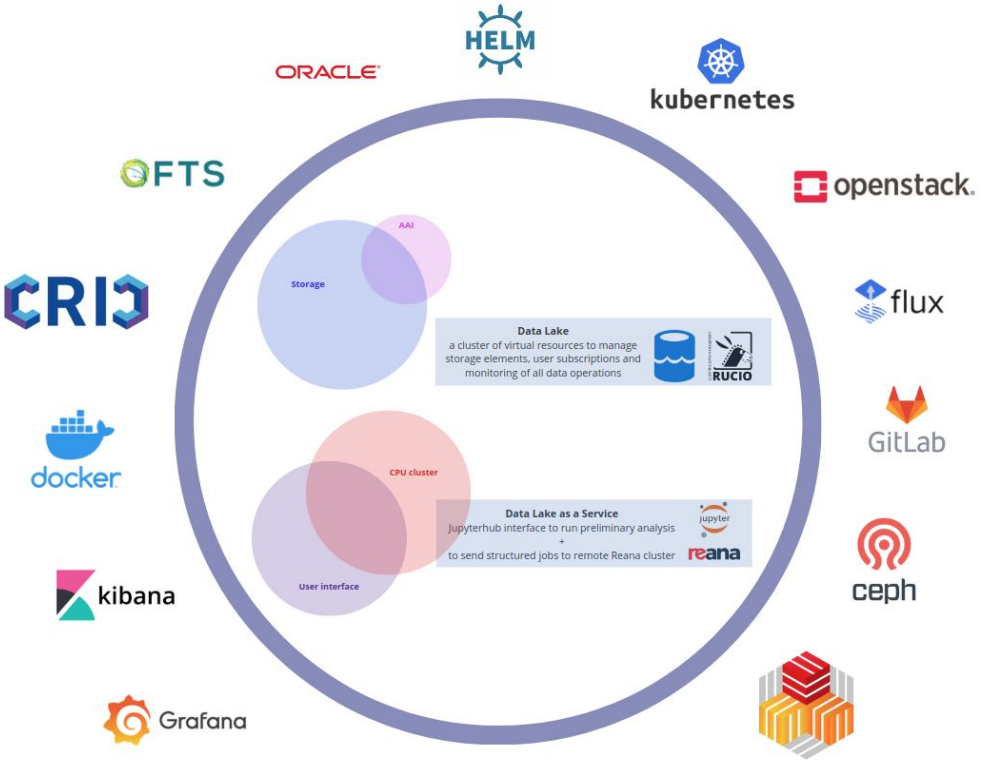

VRE@CERN
Technologies developed by CERN within the EOSC Future project to promote open science and collaboration between astroparticle physics communities.
3 followers
Switzerland
https://eoscfuture.eu
@EOSCFuture
vre-admin@cern.ch
Unfollow
Overview
Repositories 10
Discussions
Projects 1
Packages
Teams 1
People 5
Settings
README.md
CERN Virtual Research Environment
EOSC Future is an EU-funded H2020 project that is implementing the European Open Science Cloud (EOSC). EOSC will give European researchers access to a wide web of FAIR data and related services.
Our team at CERN is developing and contribution to the infrastructure code base of EOSC.
More information can be found on our website.
View as: Public
You are viewing the README and pinned repositories as a public user.
Get started with tasks that most successful organizations complete.
Top discussions this past month
Discussions are for sharing announcements, creating conversation in your community, answering questions, and more.
Start a new discussion
Pinned
Customize pins
cern-vre Public
CERN Virtual Research Env
Shell
3
profiles Public
JupyterHub Profiles
VRE@CERN
Overview
Repositories 10
Discussions
Projects 1
Packages
Teams 1
People 5
Settings
Search all discussions
Sort by: Latest activity
Label
Filter
New discussion
Categories
View all discussions
Announcements
General
Ideas
Polls
Q&A
Show and tell
Discussions
Welcome to Organization Discussions!
Organization discussions are to broadcast news, create conversation in your community, answer questions, and share ideas. To get started, you can create a new discussion.
Most helpful
Be sure to mark someone's comment as an answer if it helps you resolve your question — they deserve the credit!
Community guidelines
Community insights

Rucio-Reana: summer student project





Jupyterhub current architecture



Technologies