Virtual Research Environment goes live:
Big data shared across scientific analyses

Agisilaos Kounelis

Supervisors: Elena Gazzarrini, Alba Vendrell Moya, Xavier Espinal Curull
Contributor: Tibor Šimko

September 15, 2022
Openlab Summer Student Lightning Talks
DARK MATTER

LHC
Neutrino Observations
Indirect Detection Methods

EXTREME UNIVERSE

Compact objects
Active Galactic Nuclei
Multimessenger Astronomy

EOSC-FUTURE

COMPLETE COMMON INFRASTRUCTURE

PLATFORM to
develop, deploy,
expose and preserve
scientific workflows

WEB SERVICES

COMPUTE CLUSTER

different
data needs

different
methodologies

Openlab Summer Student Lightning Talks - 15.09.2022 - Agisilaos Kounelis
Virtual Research Environment

Generation and simulation of events
Experimental DATA from ESCAPE experiments

Data processing
Including reconstruction and calibration path

Analysis
Including background subtraction

Interpretation of results
COMBINATION of results from different experiments
COMPARISON of results with other searches

ESCAPE's Data Lake running on a K8s cluster

WebUI: DataLake-as-a-Service
Platform to reproduce workflows running on various computing backends

Openlab Summer Student Lightning Talks - 15.09.2022 - Agisilaos Kounelis
Problem

Manage plenty of Data spread across facilities at multiple institutions

Analyze and Extract useful conclusions from big data

Save Results back into data facilities
**What?**
- CERN's service that manages large volumes of data spread across facilities

**Why?**
- Centralises the control of all data
- Highly scalable and modular
- Orchestrates data sharing

---

**What?**
- CERN's re-analysis platform allowing researchers to run remote analyses.

**Why?**
- powerful - takes advantage of declarative (.yaml files) instead of imperative
- portability - thanks to recent advances of cloud container technology
Existing Solution

- **RSEs**
  - Download Data
  - Upload Data

- **Local Storage**
  - Start Analysis
  - Fetch Result Data

- **Workspace**
Existing Solution - Problems

But, what if:

- Not enough available storage for the data
- Not good enough internet connection
- Someone else wants to execute the same workflow
Implemented Solution

Local Storage

Start Analysis

Download Data

Upload Data

Workspace

RUCIO

RSEs

Openlab Summer Student Lightning Talks - 15.09.2022 - Agisilaos Kounelis
Implemented Solution

- **RSEs**
- **Local Storage**
- **Download Data**
- **Upload Data**
- **Start Analysis**

Keywords:
- Implemented Solution
- Local Storage
- Download Data
- Upload Data
- Start Analysis

---

Openlab Summer Student Lightning Talks - 15.09.2022 - Agisilaos Kounelis
Contributions

<table>
<thead>
<tr>
<th>Rucio Sidecar container</th>
<th>REANA components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Docker environments</td>
<td>Documentation</td>
</tr>
</tbody>
</table>

Openlab Summer Student Lightning Talks - 15.09.2022 - Agisilaos Kounelis
Contributions

- Rucio Sidecar container
- REANA components
- Docker environments
- Documentation
SIDECAR container

- Main Job
- Rucio Auth Sidecar
**SIDECAR container**

- Provide the appropriate configuration for Rucio
- Rucio authentication methods: OIDC, VOMS, x509, GSS, SSH key exchange, username/password, ...
- Selected **VOMS** authentication as it is already provided by REANA.

1. Add secrets needed

   ```
   $ reana-client secrets-add --file userkey.pem --file usercert.pem --env VOMSPROXY_PASS=XXXXXXXX --env VONAME=escape --env RUCIO_USERNAME=akouneli
   ```

2. Set `voms_proxy` & `rucio` to **true**

   ```
   steps:
   - name: fetchdata
     voms_proxy: true
     rucio: true
   environment: 'reanahub/reana-env-rucioclient:latest'
   commands:
     - rucio get agis_test:fitdata.C agis_test:gendata.C
   ```

3. Start of the workflow initializes the sidecar container
SIDECAR container

- Provide the appropriate configuration for Rucio
- Rucio authentication methods: OIDC, VOMS, x509, GSS, SSH key exchange, username/password, ...
- Selected VOMS authentication as it is already provided by REANA.

1. Add secrets needed

```bash
$ reana-client secrets-add --file userkey.pem --file usercert.pem --env VOMSPROXY_PASS=XXXXXXXX
--env VONAME=escape --env RUCIO_USERNAME=akouneli
```

2. Set `voms_proxy` & `rucio` to `true`

```ini
[client]
rucio_host = https://escape-rucio.cern.ch
auth_host = https://escape-rucio-auth.cern.ch
ca_cert = /rucio_cache/CERN-bundle.pem
auth_type = x509_proxy
account = akouneli
client_x509_proxy = /vomsproxy_cache/x509up_proxy
request_retries = 3

[policy]
permission = escape
schema = escape
ln2pfn_algorithm_default = hash
support = https://github.com/rucio/rucio/issues/
support_rucio = https://github.com/rucio/rucio/issues/
```

3. Start of the workflow initializes the sidecar container.
Contributions

- Rucio Sidecar container
- REANA components
- Docker environments
- Documentation

Openlab Summer Student Lightning Talks - 15.09.2022 - Agisilaos Kounelis
REANA components

REANA-Job-Controller
- Add Configuration
- Create personalised rucio.cfg
- Copy certificate files

REANA-Commons & REANA Workflow Engines

Information Propagation:

```
"rucio": {
    "type": "boolean"
},
```
Contributions

Rucio Sidecar container

REANA components

Docker environments

Documentation
Docker environments

- Each step needs a definition of the environment
- In order to run Rucio commands, an environment with **rucio-clients** installed is required

```
- name: fetchdata
  voms_proxy: true
  rucio: true
  environment: 'projectescape/rucio-client'
  commands:
  - rucio get agis_test:fitdata.C agis_test:gendata.C
```
Docker environments

Available Options:

- Use custom environments that support Rucio commands
- Use pre-configured environments for workflows that require Rucio commands:
  - Use projectescape/rucio-client, a basic image with support for Rucio commands
  - **Sidecar container as an environment**
  - Use reanahub/reana-env-root6, encapsulated runtime execution environment for ROOT6 based data analyses
Contributions

- Rucio Sidecar container
- REANA components
- Docker environments
- Documentation

Openlab Summer Student Lightning Talks - 15.09.2022 - Agisilaos Kounelis
Rucio

About Rucio

Rucio is a project that provides services and associated libraries for allowing scientific collaborations to manage large volumes of data spread across facilities at multiple institutions and organisations. Rucio was originally developed to meet the requirements of the high-energy physics experiment ATLAS, and now is continuously extended to support the LHC experiments and other diverse scientific communities.

Rucio offers advanced features, is highly scalable, and modular. It is a data management solution that covers the needs of different communities in the scientific domain (e.g., HEP, astronomy, biology).

If your workflow needs data access to a Rucio instance, you can use the Rucio authentication technique with your REANA workflows as described in detail below.
DEMO time!

```yaml
workflow:
  type: serial
  specification:
    steps:
    - name: fetchdata
      voms_proxy: true
      rucio: true
      environment: 'projectescape/rucio-client'
      commands:
        - rucio get agis_test:fitdata.C agis_test:gendata.C
    - name: gendata
      environment: 'reanahub/reana-env-root6:latest'
      kubernetes_memory_limit: '256Mi'
      commands:
        - mkdir -p results & & root -b -q 'agis_test/gendata.C(${events},${data})'
    - name: fitdata
      environment: 'reanahub/reana-env-root6:latest'
      kubernetes_memory_limit: '256Mi'
      commands:
        - root -b -q 'agis_test/fitdata.C(${data},${plot})'
    - name: uploadedata
      voms_proxy: true
      rucio: true
      environment: 'projectescape/rucio-client'
      kubernetes_memory_limit: '256Mi'
      commands:
        - rucio upload --scope agis_test results/plot.png --rse EULAKE-1
```
**DEMO time!**

- **ENV variables**
  ```
  [root@agisvm-large secrets]# export REANA_SERVER_URL=https://localhost:30443
  [root@agisvm-large secrets]# export REANA_ACCESS_TOKEN=
  ```

- **REANA Secretes**
  ```
  (reaa) [root@agisvm-large secrets]# reana-client secrets-add --file userkey.pem --file usercert.pem --env VOMSPROXY_PASS:escake --env RUCIO_USERNAME=akoueli --overwrite
  SUCCESS: Secrets VOMSPROXY_PASS, VOMNAME, RUCIO_USERNAME, userkey.pem, usercert.pem were successfully uploaded.
  ```

- **Analysis:**
  - `create, upload start`
  ```
  (reaa) [root@agisvm-large reana-demo-root6-roofit]# ./create_upload_start.sh
  --> 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.
  --> Verifying compute backends in REANA specification file...
  --> SUCCESS: Workflow compute backends appear to be valid.
  ```

  ```
  myanalysis.106
  --> Detected .gitignore file. Some files might get ignored.
  --> SUCCESS: myanalysis is pending
  ```
DEMO time!

- **myanalysis** #106
  - Started a few seconds ago
  - Running for 16 seconds
  - Step 1/4

- Finished a few seconds ago
  - Finished in 43 seconds
  - Step 4/4
DEMO time!

Note: files matching the pattern `*` will be deleted.
Further implementations

Provide abstraction of inputs field: the stage-in and stage-out processes from external sources, such as Rucio, could be automatised.

No need for direct rucio client calls.

Enrich reana.yaml input files section to support a syntax like

```
rucio(my_instance, my_scope, my_fileID)
```
THANK YOU!