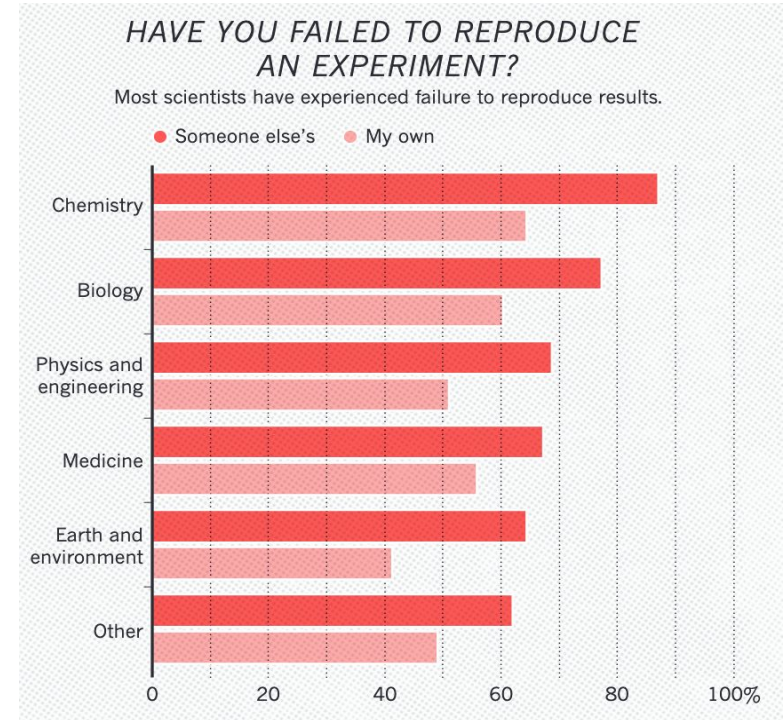
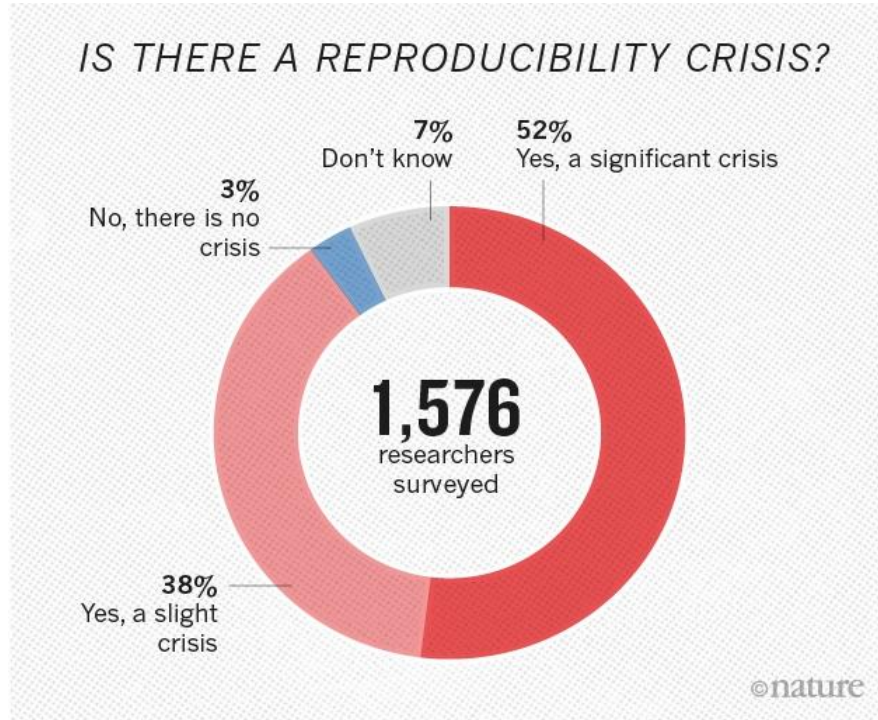


Containerised data analysis workflows using **reana**

Marco Donadoni - CERN
CSC 2022, Krakow

Reproducibility crisis

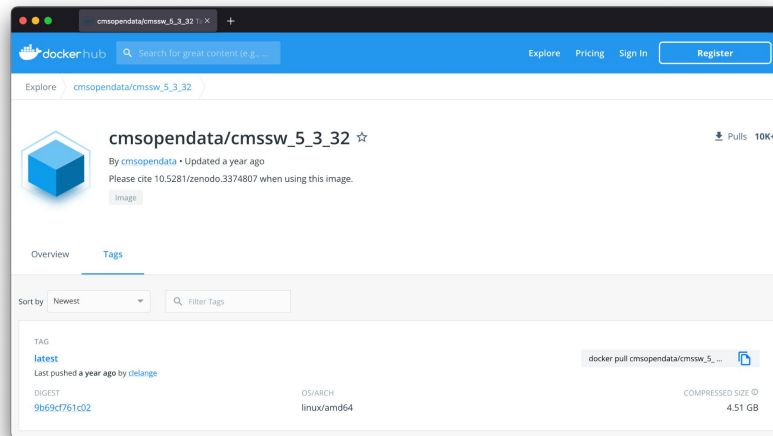


Containers

- Container images encapsulate computational environments that are independent from the hosting system
- Container images have everything needed by the data analysis
- Data analyses can be run in an isolated environment
 - ... on your machine
 - ... on a remote cloud
 - ... by someone else

```
Dockerfile (~/.cern/csc) - VIM
1 FROM centos:7
2
3 RUN yum update -y && \
4     yum install -y epel-release && \
5     yum install -y \
6         root-6.24.06 \
7         python3-3.6.8
8
9 COPY requirements.txt .
10
11 RUN pip3 install --no-cache-dir -r requirements.txt
```

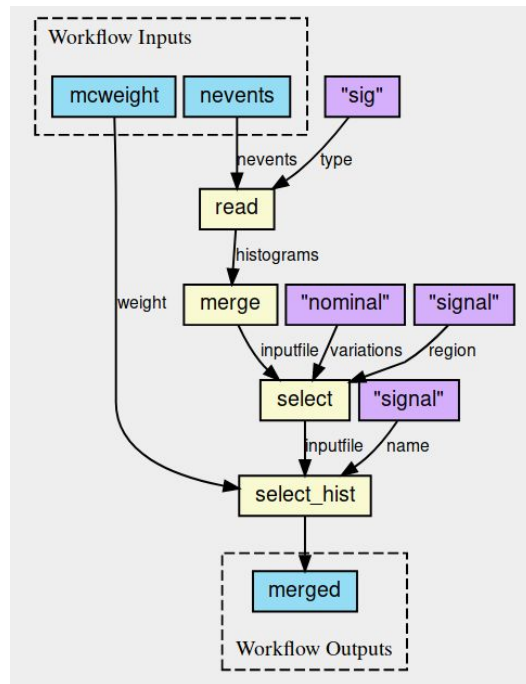
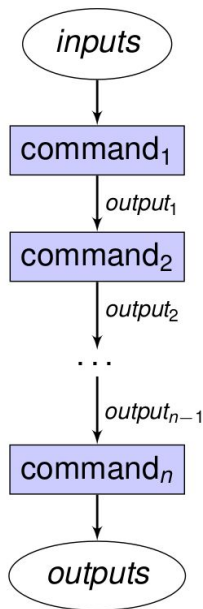
Dockerfile example



Containerised CMS software framework

Workflows

- A workflow is a recipe explaining how to produce results from input data
- What is needed?
 - Input data
 - Code
 - Computing environment
 - Workflow steps
- Steps can be described by means of declarative workflow languages (CWL, Snakemake, Yadage, ...)
- Each step can use (possibly) different container images



What is REANA?

REANA helps researchers run declarative workflows in containerised clouds

- multiple container technologies (Docker, Singularity)
- multiple workflow systems (CWL, Snakemake, Yadage)
- multiple compute backends (HTCondor, Kubernetes, Slurm)
- multiple shared storage platforms (Ceph, EOS, NFS)



Analysis Example

Input data/params & Code

Workflow steps

Environment

Results

```
reana.yaml (~/reana/src/reana-demo-root6-roofit) - VIM
1  version: 0.6.0
   inputs:
3   files:
4     - code/gendata.C
5     - code/fitdata.C
6   parameters:
7     events: 20000
8     data: results/data.root
9     plot: results/plot.png
10  workflow:
11    type: serial
12    specification:
13      steps:
14        - name: gendata
15          environment: 'reanahub/reana-env-root6:6.18.04'
16          kubernetes_memory_limit: '256Mi'
17          commands:
18            - mkdir -p results && root -b -q 'code/gendata.C(${events},${data})'
19        - name: fitdata
20          environment: 'reanahub/reana-env-root6:6.18.04'
21          kubernetes_memory_limit: '256Mi'
22          commands:
23            - root -b -q 'code/fitdata.C("${data}","${plot}")'
24      outputs:
25        files:
26          - results/plot.png
~
1,1 All
```

CLI client

```
$ tree
```

```
├── LICENSE
├── README.rst
├── code
│   ├── fitdata.C
│   └── gendata.C
├── docs
│   └── plot.png
└── reana.yaml
```

```
2 directories, 6 files
```

```
madonado@mb-pro-2020: ~/reana/src/reana-demo-root6-roofit

$ reana-client run -w reana-demo-root6-roofit
==> Creating a workflow...
==> Verifying REANA specification file... /Users/madonado/reana/src/reana-demo-root6-roofit/reana.yaml
    -> 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.
reana-demo-root6-roofit.1
==> Uploading files...
==> Detected .gitignore file. Some files might get ignored.
==> SUCCESS: File /code/gendata.C was successfully uploaded.
==> SUCCESS: File /code/fitdata.C was successfully uploaded.
==> Starting workflow...
==> SUCCESS: reana-demo-root6-roofit.1 has been queued
```

```
$ reana-client status -w reana-demo-root6-roofit
```

NAME	RUN_NUMBER	CREATED	STARTED	STATUS	PROGRESS
reana-demo-root6-roofit	1	2022-09-03T14:36:28	2022-09-03T14:36:40	running	0/2

```
$ reana-client status -w reana-demo-root6-roofit
```

NAME	RUN_NUMBER	CREATED	STARTED	ENDED	STATUS	PROGRESS
reana-demo-root6-roofit	1	2022-09-03T14:36:28	2022-09-03T14:36:40	2022-09-03T14:36:59	finished	2/2

```
$ reana-client download -w reana-demo-root6-roofit
==> SUCCESS: File results/plot.png downloaded to /Users/madonado/reana/src/reana-demo-root6-roofit.
```

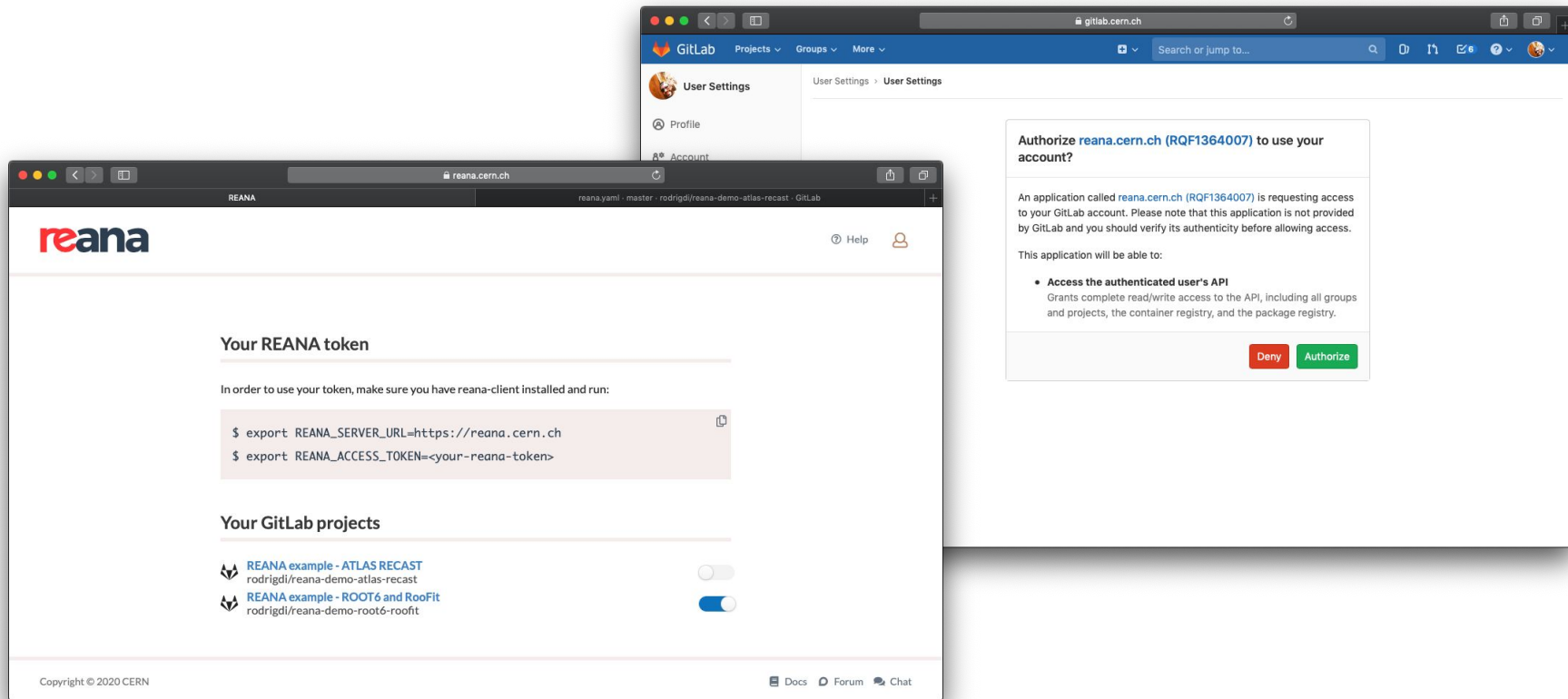

Web interface

The screenshot displays the REANA web interface within a browser window. The interface features the REANA logo in the top left, a 'Help' link and a user profile icon in the top right, and a navigation bar with links for 'Engine logs', 'Job logs', 'Workspace' (which is the active tab), and 'Specification'. Below the navigation bar is a search input field. The main content area shows a job titled 'reana-demo-root6-roofit #2' with a status of 'finished' in green, indicating it completed in 15 seconds at step 2/2. A green progress bar is visible below the job status. The 'Workspace' tab displays a table of files in the workspace.

Name	Modified	Size
results/data.root	2022-05-30T12:50:59	154453
results/plot.png	2022-05-30T12:51:06	15450
code/gendata.C	2022-05-30T12:50:46	1937
code/fitdata.C	2022-05-30T12:50:46	1648

At the bottom of the interface, there are links for 'Docs', 'Forum', and 'Cluster Health'.

... and many other features such as GitLab integration



Conclusions

- Containerization helps in encapsulating computational environments
- Declarative workflow languages describe the analysis computation logic
- data + code + environment + workflow → reproducible analyses!
- REANA aims at helping researchers
 - execute complex data analyses on remote compute clouds
 - organise data analyses to facilitate preservation and reuse

