

Search for computational workflow synergies in reproducible research data analyses in particle physics and life sciences

Tibor Šimko
CERN
Geneva, Switzerland
tibor.simko@cern.ch

Kyle Cranmer
New York University
New York, USA
kyle.cranmer@nyu.edu

Michael R. Crusoe
Common Workflow Language
Vilnius, Lithuania
mrc@commonwl.org

Lukas Heinrich
New York University
New York, USA
lukas.heinrich@cern.ch

Anton Khodak
Wellcome Sanger Institute
Hinxton, UK
anton.khodak@sanger.ac.uk

Dinos Kousidis
CERN
Geneva, Switzerland
dinos.kousidis@cern.ch

Diego Rodríguez
CERN
Geneva, Switzerland
diego.rodriguez@cern.ch



Usage

1

Structure your analysis

```
version: 0.3.0
inputs:
  files:
    - code/mycode.py
    - inputs/mydata.csv
  parameters:
    myparameter: myvalue
workflow:
  type: cwl
  file: workflow/myworkflow.cwl
  outputs:
    files:
      - results/myplot.png
```

2

Select a REANA cluster...

...or install your own

```
# Install REANA client
$ export REANA_SERVER_URL=https://reana.cern.ch/
$ export REANA_ACCESS_TOKEN=XXXXXXXX
$ reana-client create -n my-analysis
$ export REANA_WORKON=my-analysis
$ reana-client upload /code /data
$ reana-client upload /code /data
$ reana-client start
$ reana-client status
$ reana-client list
$ reana-client download results/plot.png
```

3

Run your analysis

```
# Install REANA client
$ mvirtualenv reana-client
$ pip install reana-client
$ reana-client create -n my-analysis
$ export REANA_WORKON=my-analysis
$ reana-client upload /code /data
$ reana-client upload /code /data
$ reana-client start
$ reana-client status
$ reana-client list
$ reana-client download results/plot.png
```

Flexible

Run many computational workflow engines



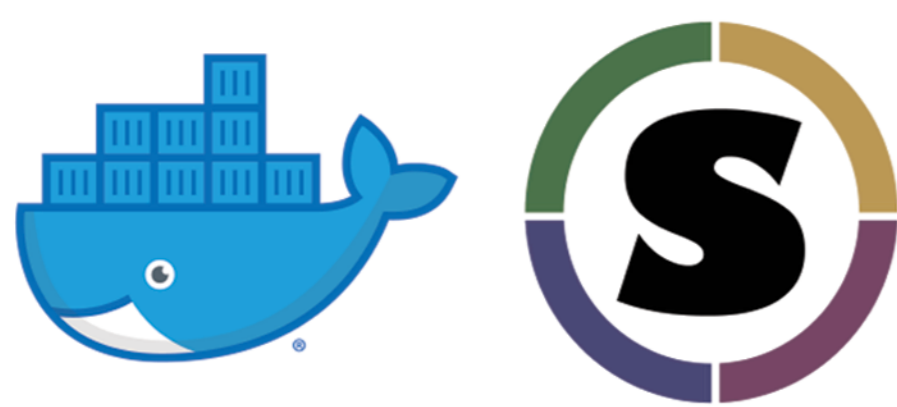
Scalable

Support for remote compute clouds



Reusable

Containerised once reuse elsewhere. Cloud-native

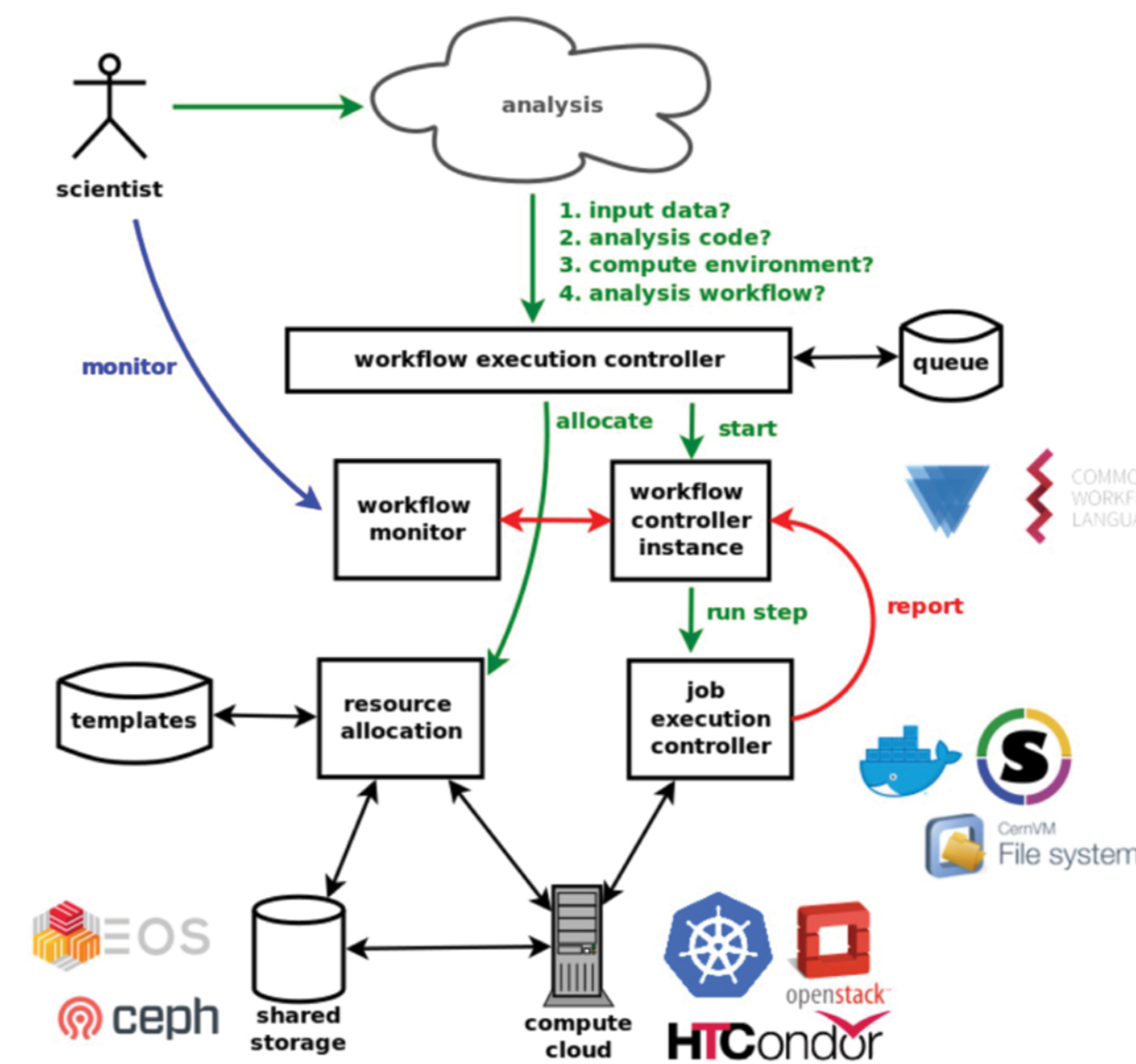


Free

Free Software. MIT licence. Made with ❤️ at CERN



Architecture



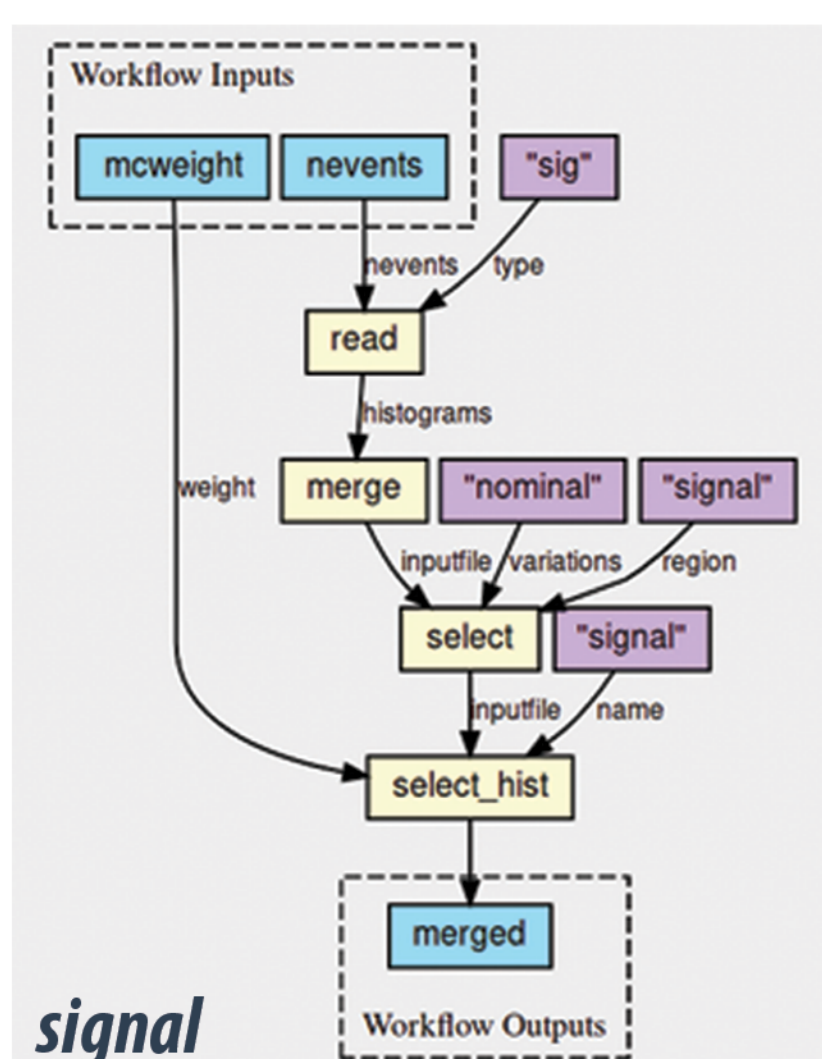
Built on micro-services



RESTful APIs

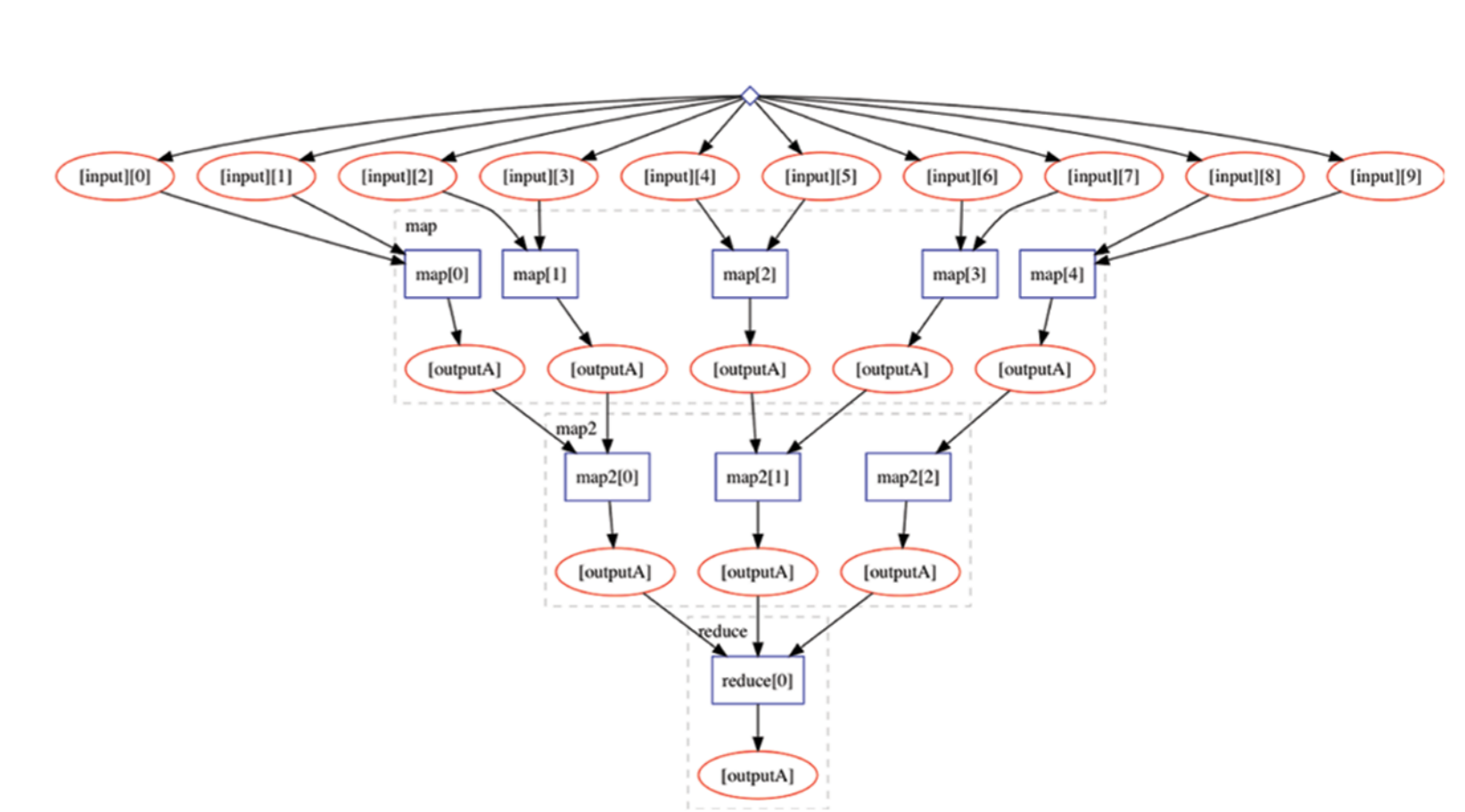
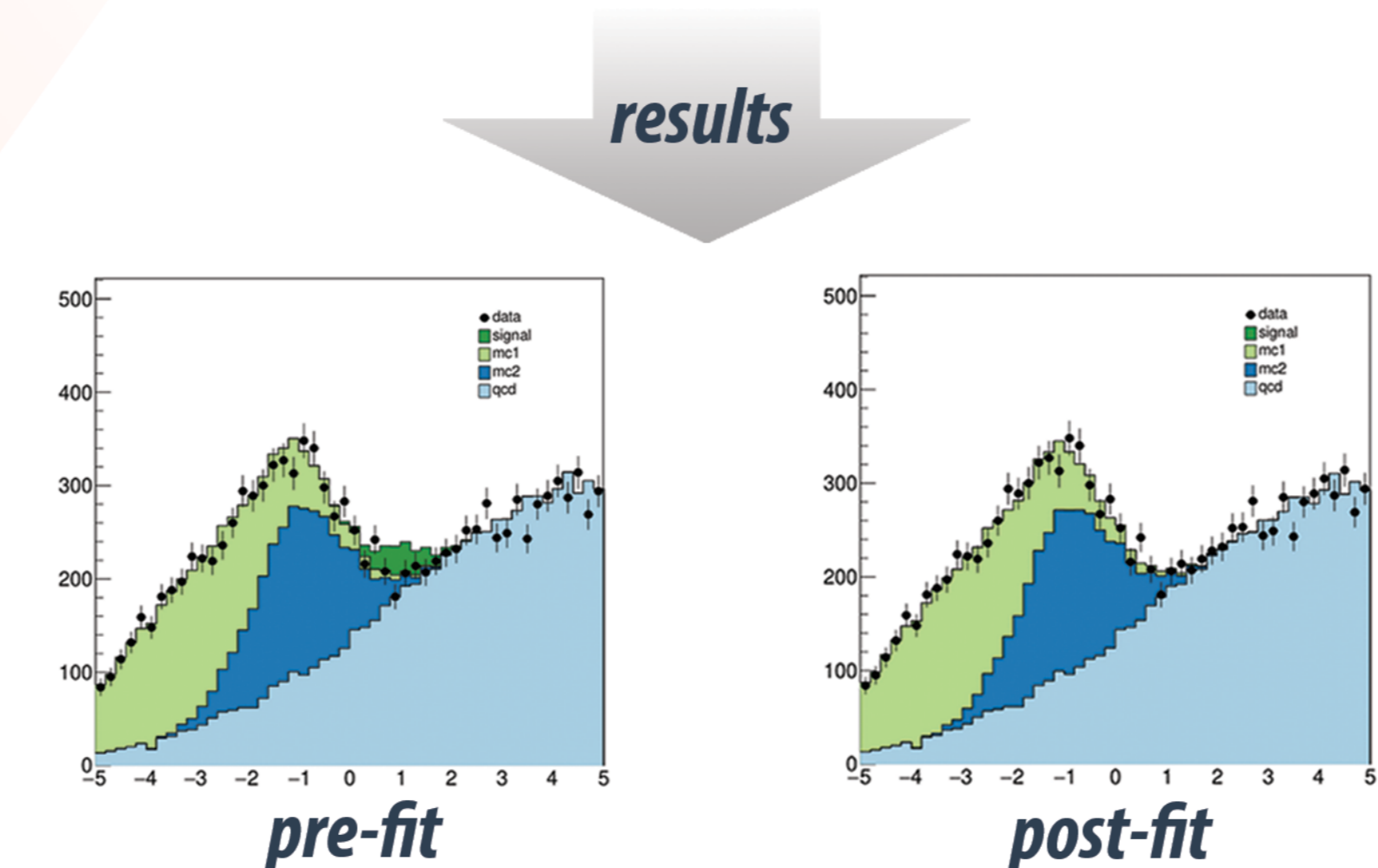
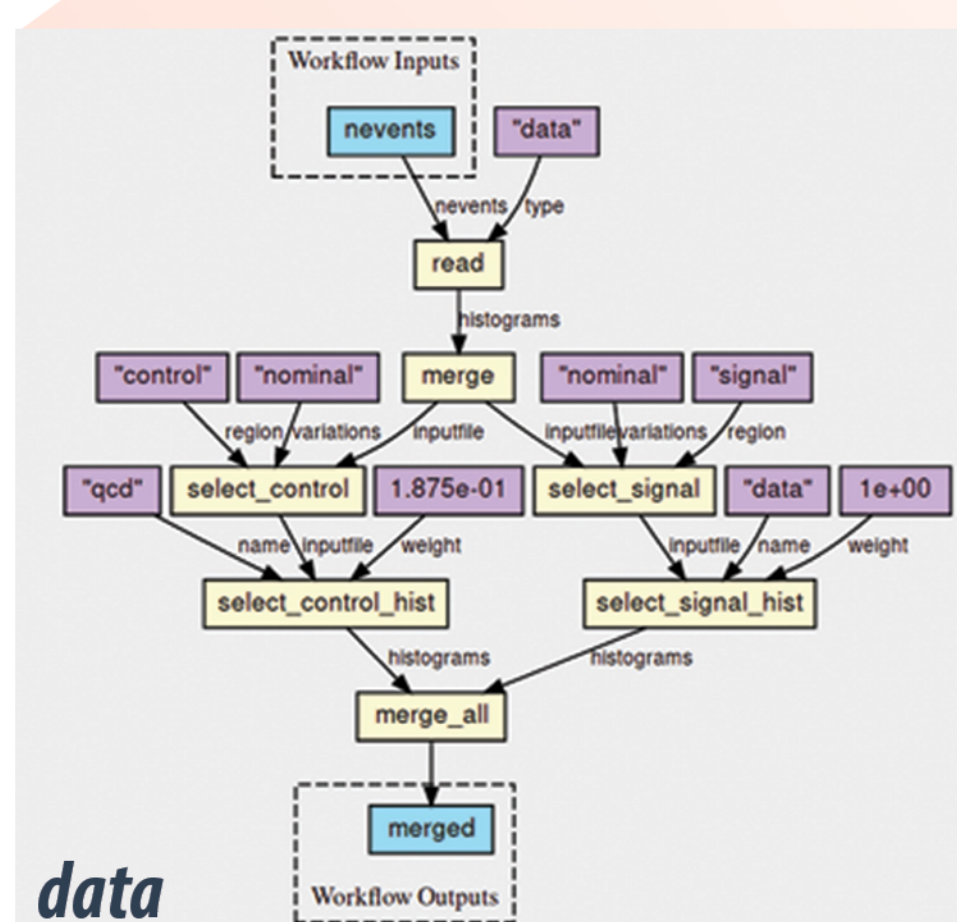
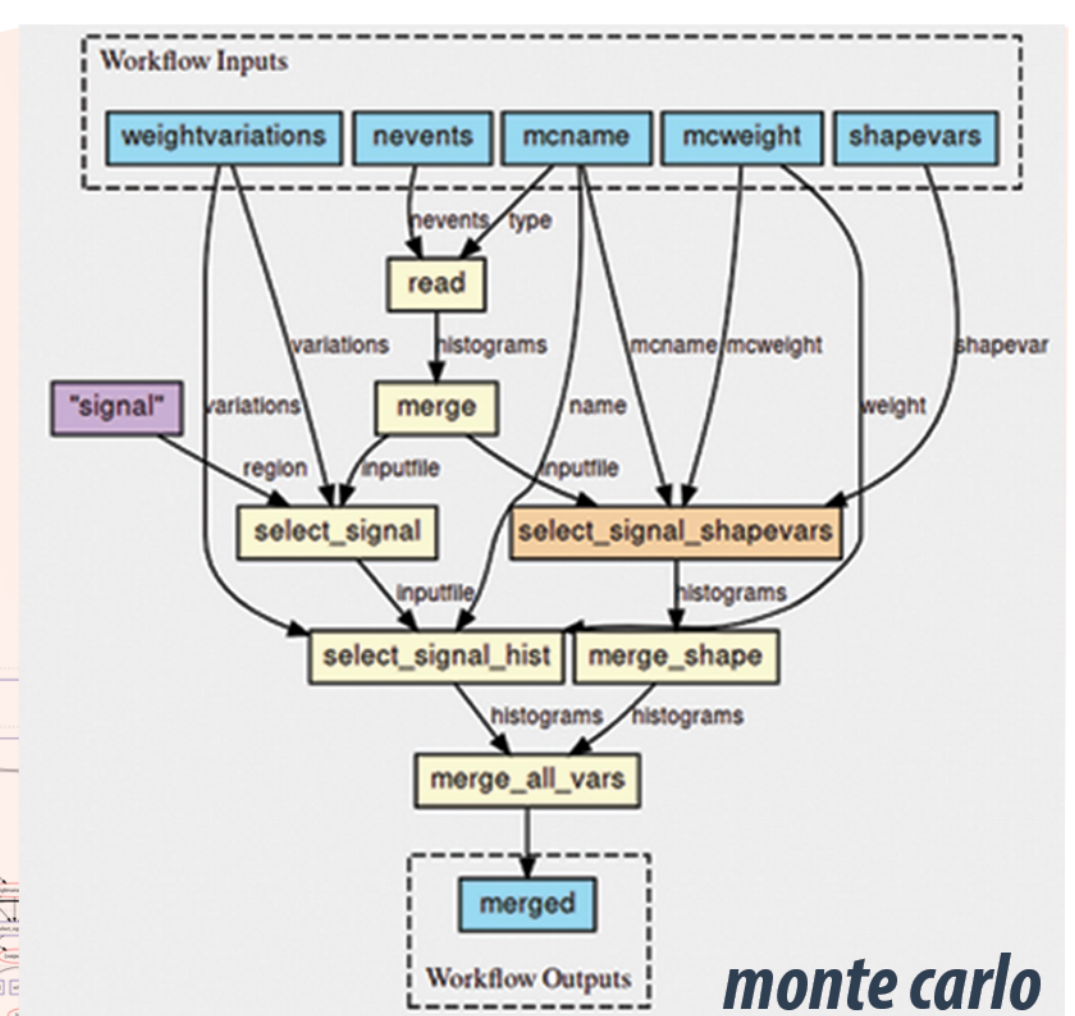
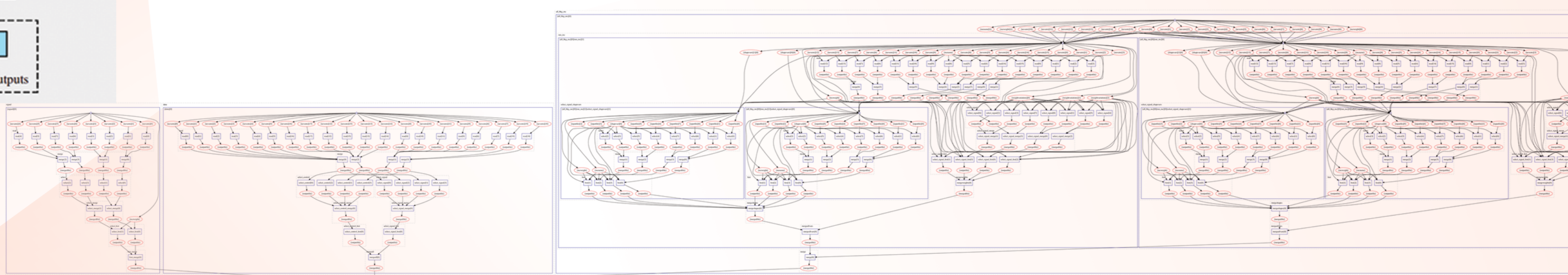


Leveraging the power of open source



Beyond Standard Model physics workflows

Expressing particle physics analysis workflows in Yadage and CWL



Conclusions

- workflow-as-a-service platform with multi engine support (CWL, Serial, Yadage)
- run complex containerised workflows on Kubernetes-orchestrated clusters
- support for diverse compute clouds (Kubernetes, OpenStack)
- direct translation of typical physics data analysis workflows to CWL
- confirmed synergies with computational practices in life sciences and other scientific disciplines

Multi-cascading map-reduce

Advanced Yadage workflow concepts require CWL scatter-gather extension