On-Demand Distributed Computing Workflow for Physics Analysis at the CMS Experiment

Diyaselis Delgado[1], Kati Lassila-Perini[2], Clemens Lange[3]

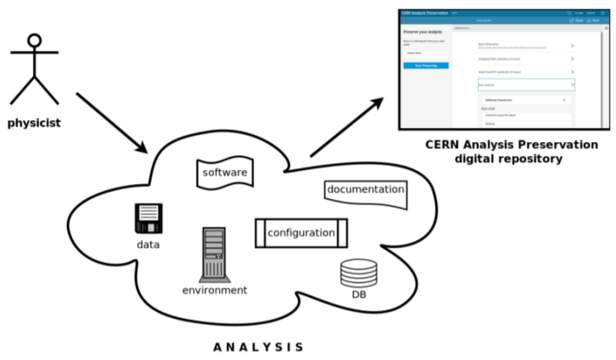
[1] *University of Puerto Rico Mayaguez,* [2] *Helsinki Institute of Physics,* [3] *European Organization for Nuclear Research*

Introduction

The CMS experiment is a strong contributor to the CERN Open Data Portal (CODP) where it publishes research level data together with environment, software and instructions on how to use it. This project takes part in developing and testing simplified examples of open data while providing a connection to analysis preservation and a new reproducible research data analysis platform (ReANA).

CERN Analysis Preservation (CAP)

- The CAP Framework [2] is a central platform for the four LHC collaborations
- Developed to address the need for the long-term preservation of the data analysis process

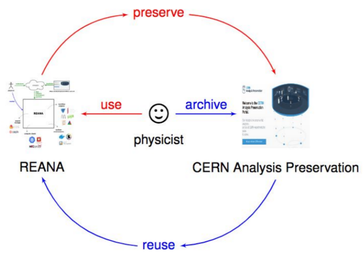


- Capturing knowledge and assets of individual physics analyses in a digital repository in view of facilitating their future reuse.

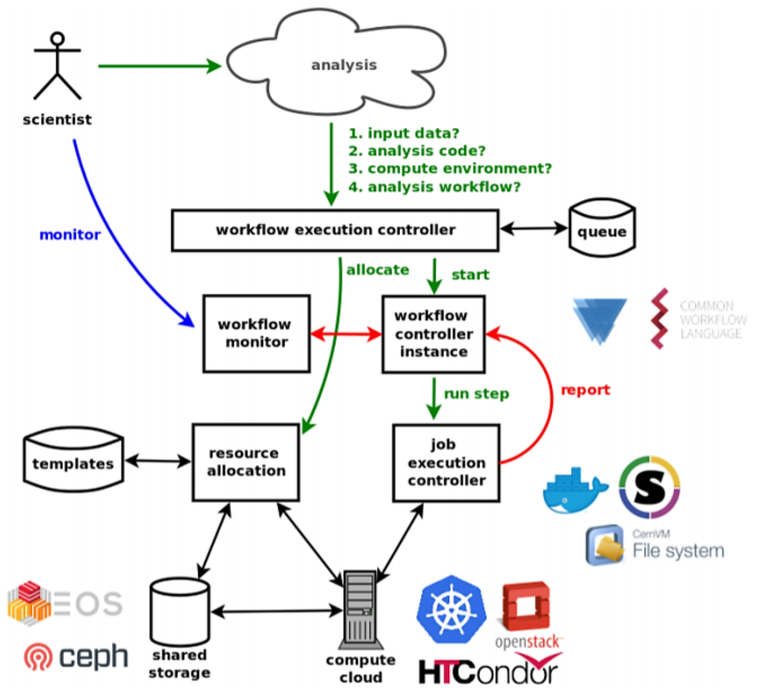
ReANA Platform

reana is a reusable and reproducible research data analysis platform. [3]

- Structures input data, analysis code, containerized environments and computational workflows
- Supports multiple scenarios, such as: computing clouds, running environments, resource orchestration tools, workflow engines, and shared storage systems
- ReANA is applicable to any scientific discipline



CAP and ReANA Integration



- Building system to instantiate preserved analysis on the cloud

Higgs-to-four-lepton analysis example using 2011-2012 data

- Simplified reimplementation of the Higgs discovery [1]
- Uses CMS Open Data inputs and available CMS Software environment
- ReANA, captures the analysis workflow and runs the commands to obtain the output

input.yml

```
BuildFile:
  class: File
  path: HiggsExample20112012/HiggsDemoAnalyzer/BuildFile.xml
HiggsDemoAnalyzerGit:
  class: File
  path: HiggsExample20112012/HiggsDemoAnalyzer/src/HiggsDemoAnalyzerGit.cc
Cert_190456-208686_8TeV_22Jan2013ReReco_Collisions12_JSON:
  class: File
  path: HiggsExample20112012/datasets/Cert_190456-208686_8TeV_22Jan2013ReReco_Collisions12_JSON.txt
demoanalyzer_cfg_level3MC:
  class: File
  path: HiggsExample20112012/Level3/demoanalyzer_cfg_level3MC.py
demoanalyzer_cfg_level3data:
  class: File
  path: HiggsExample20112012/Level3/demoanalyzer_cfg_level3data.py
M4Lnormdatall_lvl3:
  class: File
  path: HiggsExample20112012/Level3/M4Lnormdatall_lvl3.cc
```

workflow.cwl

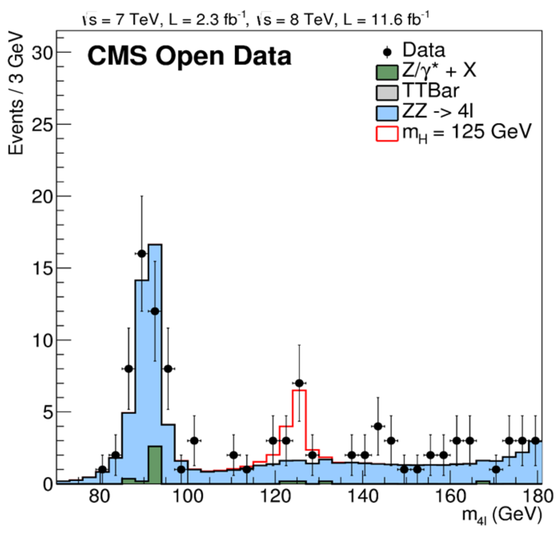
```
steps:
  step1:
    run: step1.cwl
    in:
      BuildFile: BuildFile
      HiggsDemoAnalyzerGit: HiggsDemoAnalyzerGit
#     events: events
    out: [output_txt]
```

steps.cwl

```
arguments:
  - prefix: -c
    valueFrom: |
      echo "::: Setting up CMS environment...";
      source /opt/cms/cmsset_default.sh;
      echo "::: Setting up CMS environment... [done]";
      scramv1 project CMSSW_5_3_32;
      cd CMSSW_5_3_32/src || exit;
      eval `scramv1 runtime -sh`;
      ls;
      # git clone git://github.com/cms-opendata-analyses/HiggsExample20112012.git || exit
      # scram b || exit
      # cd HiggsExample20112012/Level3 || exit
      # cmsRun {{inputs.parameters.command}} || exit
      # cp *.root /mnt/vol
      # ls -l /mnt/vol
```

reana.yaml

```
version: 0.2.0
code:
  files:
    - code/demoanalyzer_cfg_level3MC.py
    - code/demoanalyzer_cfg_level3data.py
    - code/M4Lnormdatall_lvl3.cc
    - code/HiggsDemoAnalyzerGit.cc
inputs:
  parameters:
    input: workflow/cwl/input.yml
outputs:
  files:
    - outputs/mass4l_combined_user3.pdf
environments:
  - type: docker
    image:   clelange/cmssw:5_3_32
workflow:
  type: cwl
  file: workflow/cwl/workflow.cwl
```



Running the analysis on ReANA

- This CERN project interprets and executes workflows.
- In addition to the existing code, only a set of descriptive workflow files is needed.
- Installing the REANA command-line client

```
$ mkvirtualenv reana-client -p /usr/bin/python2.7
$ pip install reana-client

$ export REANA_SERVER_URL=http://reana.cern.ch
$ eval $(reana-cluster env)
```

- Test client-to-server connection and create a new workflow

```
$ reana-client ping
Server is running.
$ reana-client workflow create
workflow.1
$ reana-client workflow status -w workflow.1
NAME       RUN_NUMBER   ID
workflow   1            91797125-012c-498d-8a92-b4f7e3598513
$ export REANA_WORKON=workflow.1
```

- Upload code and start the workflow

CODP Validation Examples

- Validation examples prove that a result can be obtained with the legacy/open data tools
- For example, a validation codes for 2010 datasets [5] consists of:

  (1) Setting up the CMS environment

  (2) Compiling and Running the Analysis

  (3) Producing a result (e.g. a comparison plot for the dimuon mass spectrum)

Structure your analysis

```
version: 0.2.0
code:
  files:
  - code/mycode.py
inputs:
  files:
  - inputs/mydata.csv
  parameters:
    myparameter: myvalue
environments:
  - type: docker
    image: johndoe/mypython:1.0
workflow:
  type: cwl
  file: workflow/myworkflow.cwl
outputs:
  files:
  -outputs/myplot.png
```

Select a REANA cluster...

```
$ export
REANA_SERVER_URL=https://reana.cern.ch/
```

...or install your own

```
# install kubectl 1.9.1 and minikube
0.23.0
$ sudo dpkg -i kubectl*.deb minikube*.deb
$ minikube start --kubernetes-
version="v1.6.4"
# install reana-cluster utility
$ mkvirtualenv reana-cluster
$ pip install reana-cluster
# deploy new cluster and check progress
$ reana-cluster init
$ reana-cluster status
# set environment variables for reana-
client
$ eval $(reana-cluster env)
```

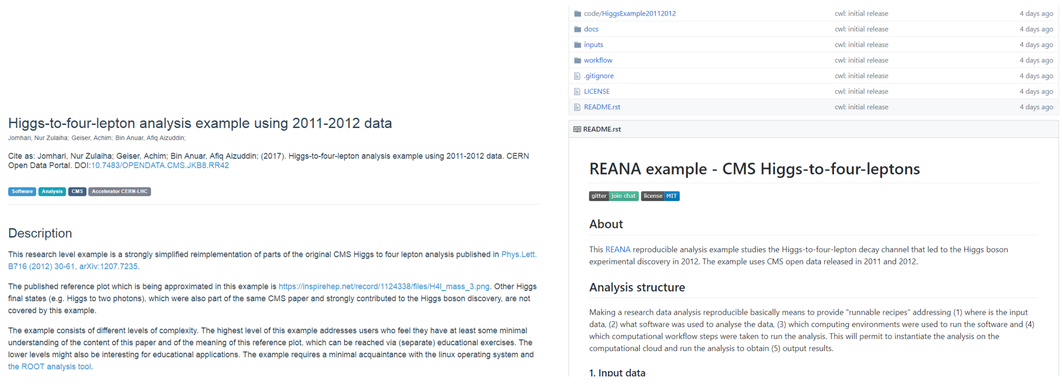
Run your analysis

```
# install reana-client
$ mkvirtualenv reana-client -p
/usr/bin/python2.7
$ pip install reana-client
$ reana-client ping
# create new workflow
$ export REANA_WORKON=$(reana-client
workflow create)
# upload runtime code and inputs
$ reana-client code upload ./code/*
$ reana-client inputs upload ./inputs/*
# start workflow and check progress
$ reana-client workflow start
$ reana-client workflow status
# download outputs
$ reana-client outputs list
$ reana-client outputs download myplot.png
```

- Demonstrated setup allows to rerun CMS Open Data analyses (e.g. confirm that instructions are correct and run large-scale analyses relatively fast)

Working Example on the Platform

The Higgs-to-four-lepton analysis example has been implemented in ReANA. This CMS Open Data example is up and running. [4]



Higgs-to-four-lepton analysis example using 2011-2012 data

Jomhari, Nur Zulaiha; Geiser, Achim; Bin Anuar, Afiq Alzuddin;

Cite as: Jomhari, Nur Zulaiha; Geiser, Achim; Bin Anuar, Afiq Alzuddin; (2017). Higgs-to-four-lepton analysis example using 2011-2012 data. CERN Open Data Portal. DOI:10.7483/OPENDATA.CMS.JKB8.RR42

Software Analysis CMS Accelerator CERN LHC

Description

This research level example is a strongly simplified reimplementation of parts of the original CMS Higgs to four lepton analysis published in Phys.Lett. B716 (2012) 30-61, arXiv:1207.7235.

The published reference plot which is being approximated in this example is https://inspirehep.net/record/1124338/files/H4l_mass_3.png. Other Higgs final states (e.g. Higgs to two photons), which were also part of the same CMS paper and strongly contributed to the Higgs boson discovery, are not covered by this example.

The example consists of different levels of complexity. The highest level of this example addresses users who feel they have at least some minimal understanding of the content of this paper and of the meaning of this reference plot, which can be reached via (separate) educational exercises. The lower levels might also be interesting for educational applications. The example requires a minimal acquaintance with the linux operating system and the ROOT analysis tool.

code/HiggsExample20112012 cwl: initial release 4 days ago
docs cwl: initial release 4 days ago
inputs cwl: initial release 4 days ago
workflow cwl: initial release 4 days ago
.gitignore cwl: initial release 4 days ago
LICENSE cwl: initial release 4 days ago
README.rst cwl: initial release 4 days ago

README.rst

REANA example - CMS Higgs-to-four-leptons

gitter Join chat License MIT

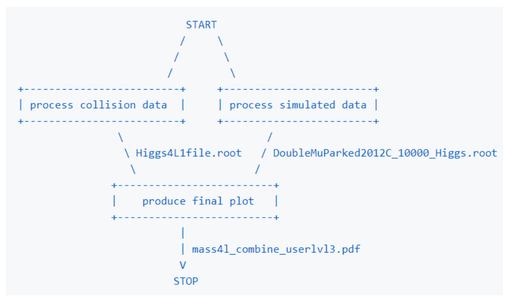
About

This REANA reproducible analysis example studies the Higgs-to-four-lepton decay channel that led to the Higgs boson experimental discovery in 2012. The example uses CMS open data released in 2011 and 2012.

Analysis structure

Making a research data analysis reproducible basically means to provide "runnable recipes" addressing (1) where is the input data, (2) what software was used to analyse the data, (3) which computing environments were used to run the software and (4) which computational workflow steps were taken to run the analysis. This will permit to instantiate the analysis on the computational cloud and run the analysis to obtain (5) output results.

1. Inout data



References

[1] http://doi.org/10.7483/OPENDATA.CMS.JKB8.RR42
[2] https://analysispreservation.cern.ch/
[3] http://www.reana.io/
[4] https://github.com/reanahub/reana-demo-cms-h4l
[5] http://opendata.cern.ch/search?page=1&size=20&q=validation%20cms&subtype=Validation