Open Data at CERN: Preserve to Reuse

Tibor Šimko
CERN

UNIGE Open Science Class visit at CERN
November 30th 2022

https://indico.cern.ch/event/1224257/
CERN Open Data portal
CERN Open Data portal

- launched in November 2014
- rich content
  - collision and simulated datasets for research
  - derived datasets for education
  - configuration files and documentation
  - virtual machines and container images
  - software tools and analysis examples
- total size in November 2022
  - over 15’000 bibliographic records
  - over 1’000’000 files
  - over 3 petabytes

Developed by CERN in close collaboration with Experiments

https://opendata.cern.ch
Education-oriented use cases

Interactive event display and histogramming for derived datasets
Research-oriented use cases

Run CernVM Virtual Machines

Run research-grade analysis examples

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

Johann Ni, Zuzana Golmar, Achim, Bin Anuar, Aliq Abdulle.

Cite as: Johann, Ni, Zuzana; Golmar, Achim; Bin Anuar, Aliq Abdulle. (2017). Higgs-to-four-lepton analysis example using 2011-2012 data.
CERN Open Data Portal. DOI:10.7483/OPENDATA.CMS-JKBXJ842

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:1207.7236.

The published reference plot which is being approximated in this example is https://inspirehep.net/record/1243836/files/427_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 note and of the meaning of this reference plot which can be reached via training.

Use with

The example runs on the Open Data and with the Linux OS.

©tiborsimko
Enables independent theoretical research

Over thirty papers citing CMS open data

Searches, QCD jet studies, Machine Learning...

...that the CMS collaboration cites!
LHC collaboration data preservation and open access policies

Restricted data → embargo period (5 years) → open data

ALICE data preservation strategy

LHCb Collaboration Data Preservation and Open Access Policies

ALICE data access policy

May 1st 2014

ATLAS Data Access Policy

Introduction

ATLAS has developed a robust framework for open access to publication data. This document specifies the open access policies for the ATLAS Collaboration and provides guidelines for ATLAS data access. The policies are based on the ATLAS Collaboration’s commitment to maximizing the impact of the ATLAS dataset by enabling data to be freely accessed and used.

Policies for Different Data Levels

There are three levels of data that the ATLAS Collaboration manages: restricted, embargoed, and open. The level of access is determined by the level of embargo on the data.

Restricted data: data that is embargoed for a specific period of time (5 years).

Embargoed data: data that is embargoed for an unspecified period of time.

Open data: data that is available immediately.

CERN data preservation, re-use, and open access policy

CERN adopts a unique approach to data preservation, re-use, and open access. Data that is generated by the CERN European Laboratory for Particle Physics is preserved and made available for re-use and open access. This approach is based on the CERN Open Access Policy, which sets out the principles and guidelines for the preservation, re-use, and open access of CERN data.

Abstract

This document presents the CERN Data Access Policy. This policy is consistent with the CERN Open Access Policy Statement of Intent (10 June 2014).
A FAIRy tale
The FAIR Guiding Principles for scientific data management and stewardship

Mark D. Wilkinson, Michel Dumontier, [...] Barend Mons

Scientific Data 3, Article number: 160018 (2016) | Cite this article
180k Accesses | 2306 Citations | 1797 Altmetric | Metrics

An Addendum to this article was published on 19 March 2019

Abstract

There is an urgent need to improve the infrastructure supporting the reuse of scholarly data. A diverse set of stakeholders—representing academia, industry, funding agencies, and scholarly publishers—have come together to design and jointly endorse a concise and measureable set of principles that we refer to as the FAIR Data Principles. The intent is that these may act as a guideline for those wishing to enhance the reusability of their data holdings. Distinct from peer initiatives that focus on the human scholar, the FAIR Principles put specific emphasis on enhancing the ability of machines to automatically find and use the data, in addition to supporting its reuse by individuals. This Comment is the first formal publication of the FAIR Principles, and includes the rationale behind them, and some exemplar implementations in the community.
The first step in (re)using data is to find them. Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services.

- F1. (Meta)data are assigned a globally unique and persistent identifier
- F2. Data are described with rich metadata (defined by R1 below)
- F3. Metadata clearly and explicitly include the identifier of the data they describe
- F4. (Meta)data are registered or indexed in a searchable resource
Each dataset is identified by a “record ID” and optionally minted with a DOI
Dataset information: rich context description & machine readability

Dataset information: rich context description & machine readability

How were these data selected?

There are four categories of triggers in the Mu dataset (with significant overlaps):
- 70% inclusive single muon triggers with varying trigger pT threshold 3.5, 7.5, 11, 13, 15, 17, 19, 21 GeV plus a few with loosened quality cuts.
- 20% isolated single muon triggers with varying trigger pT threshold 9, 11, 13, 15, 17 GeV.
- 10% inclusive dimuon triggers with varying trigger pT threshold 3.5 GeV plus one Z->mmu trigger with loosened quality cuts.
- 20% combinations of muon triggers with various pT thresholds 3.5, 7, 8, 9, 11 GeV with some EM/emu/gamma or hadronic energy deposit with thresholds 6-100 GeV.

How were these data validated?

During data-taking all the runs recorded by CMS are certified as good for physics analysis if all subdetectors, trigger, lumi and physics objects (tracking, electrons, muon, photon, jet and MET) show the expected performance. Certification is based first on the offline streamer evaluation and later on the feedback provided by detector and Physics Object Group experts. Based on the above information, which is stored in a specific database called Run Registry, the Data Quality Monitoring group verifies the consistency of the certification and prepares a pool file of certified runs to be used for physics analysis. For each reprocessing of the raw data, the above mentioned steps are repeated. For more information see:

CMS data-quality monitoring: Systems and experiences
The CMS Data Quality Monitoring software experience and future improvements
The CMS data quality monitoring software: experience and future prospects

How can you use these data?

You can access these data through the CMS Virtual Machine. See the instructions for setting up the Virtual Machine and getting started.

How to install the CMS Virtual Machine
Getting started with CMS open data

https://opendata.cern.ch/record/14  http://doi.org/10.7483/OPENDATA.CMS.B8MR.C4A2

Rich curated context information on data selection, validation, and use
## CMS Guide to the CMS condition database

This page explains the use of global tags and the condition database with the CMS Open Data.

A Global Tag is a coherent collection of records of additional data needed by the reconstruction and analysis software. The Global Tag is defined for each data-taking period, separately for collision and simulated data.

These records are stored in the condition database. Condition data include non-event-related information (Alignment, Calibration, Temperature, etc.) and parameters for the simulation/reconstruction/analysis software. For CMS Open Data, the condition data are provided as SQLite files in the `/cmfs/cms-opendata-conddb.cern.ch` directory, which is accessible through the CMS Open Data VM. Note that when using CMS Open Data docker images, connecting to this area with the command `process.GlobalTag.connect = cms.string('@string')` in the job configuration file is not required.

Most physics objects in the CMS Open Data are already calibrated and ready-to-use, and no additional corrections are needed other than selection and identification criteria, which will be applied in the analysis code. Therefore, simple analyses do not need to access the condition database. Examples of such analyses are the di-muon spectrum example or the Higgs analysis example.

However, access to the condition database is necessary, for example, for jet energy corrections and trigger configuration information. Examples of such analyses are for the PAT object production or the top quark pair production.

Note that when you need to access the condition database, the first time you run the job on the CMS Open Data VM, it will download the condition data from the `/cmfs` area. It will take time (an example run of a 10 Mbps link took 45 mins), but it will only happen once as the files will be cached on your VM. The job will not produce any output during this time, but you can check the ongoing processes with the command `top` and you can monitor the progress of reading the condition data to the local cache with the command `df`.

The Global Tags for condition data are different for different types of data taking. Below, the instructions are given for proton-proton and heavy-ion data.

---

### Proton-proton data

**For 2010 collision data**, the global tag available in the `/cmfs` area is FT_R_42_V10A. When using the "CMS-OpenData-1.1.2" VM or a higher version, it is recommended reading the condition data from there. First, set the symbolic links:

```
ln -s /cmfs/cms-opendata-conddb.cern.ch/FT_R_42_V10A FT_R_42_V10A
ln -s /cmfs/cms-opendata-conddb.cern.ch/FT_R_42_V10A.db FT_R_42_V10A.db
```

Then, define the correct set of condition data by mentioning the Global Tag in the configuration file of the job.

```
#globaltag
process.GlobalTag.connect = cms.string('sqlite_file:/cmfs/cms-opendata-conddb.cern.ch/FT_R_42_V10A.db')
```

Note that this only works in the "CMS-OpenData-1.1.2" or a higher version of the 2010 CMS Open Data VM.

**For 2010 Montecarlo data**, the global tag is START42_V17B. To access the condition database, first, set the symbolic links:

```
ln -s /cmfs/cms-opendata-conddb.cern.ch/START42_V17B START42_V17B
ln -s /cmfs/cms-opendata-conddb.cern.ch/START42_V17B.db START42_V17B.db
```

Then, define the correct set of condition data by mentioning the Global Tag in the configuration file of the job.

```
#globaltag for 2010 MC
process.GlobalTag.connect = cms.string('sqlite_file:/cmfs/cms-opendata-conddb.cern.ch/START42_V17B.db')
#process.GlobalTag.globaltag = 'START42_V17B::ALL'
```

Note that this only works in the "CMS-OpenData-1.1.2" or a higher version of the 2010 CMS Open Data VM.

**For 2011 collision data**, the global tag is FT_53_LV5_AN1. To access the condition database, first, set the symbolic links:

```
ln -s /cmfs/cms-opendata-conddb.cern.ch/FT_53_LV5_AN1_RUNA FT_53_LV5_AN1
ln -s /cmfs/cms-opendata-conddb.cern.ch/FT_53_LV5_AN1_RUNA.db FT_53_LV5_AN1_RUNA.db
```

Make sure the `/cmfs-opendata-conddb.cern.ch` directory has actually expanded in your VM. One way of doing this is executing:

```
ls -l
ls -l /cmfs/
```

---

A detailed guide to CMS global tags and condition database
Information discovery

Faceted search interface

Corresponding REST API
FAIR principles: A is for Accessible

Once the user finds the required data, she/he needs to know how can they be accessed, possibly including authentication and authorisation.

- A1. (Meta)data are retrievable by their identifier using a standardised communications protocol
- A1.1 The protocol is open, free, and universally implementable
- A1.2 The protocol allows for an authentication and authorisation procedure, where necessary
- A2. Metadata are accessible, even when the data are no longer available
Downloading content

Browsing and downloading files manually

Automated cernopendata-client
Command-line client

```
/tmp $ cernopendata-client --help
Usage: cernopendata-client [OPTIONS] COMMAND [ARGS]...

Command-line client for interacting with CERN Open Data portal.

Options:
  --help    Show this message and exit.

Commands:
  download-files    Download data files belonging to a record.
  get-file-locations    Get a list of data file locations of a record.
  get-metadata    Get metadata content of a record.
  list-directory    List contents of a EOSPUBLIC Open Data directory.
  verify-files    Verify downloaded data file integrity.
  version    Return cernopendata-client version.
```

```
/tmp $ cernopendata-client get-metadata --doi 10.7483/OPENDATA.CMS.ZVT8.MZNY
...
  "formats": ["wodsimg", "root"],
  "number_events": 38125269,
  "number_files": 26958,
  "size": 956028252603
```

```
/tmp $ cernopendata-client get-file-locations --recid 5500
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/BuildFile.xml
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/HiggsDemoAnalyzer.cc
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/list_indexfile.txt
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/M4NormDatafile.cc
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/M4NormDatafile_v13.cc
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/demonaalyzer_cfg_level1MC.py
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/demonaalyzer_cfg_level1MC.py
http://opendata.cern.ch/oes/open_data/cms/software/HiggsExample20120212/demonaalyzer_cfg_level1MC.py
```

```
/tmp $ cernopendata-client download-files --recid 5500 --verify
  Downloading file 1 of 11
    -> File: /5500/BuildFile.xml
    -> Progress: 0/0 KIB (100%)
  == Verifying file BuildFile.xml...
    -> Expected size 385, found 305
    -> Expected checksum adler32:ff63668a, found adler32:ff63668a
  == Downloading file 2 of 11
    -> File: /5500/HiggsDemoAnalyzer.cc
    -> Progress: 82/82 KIB (100%)
  == Verifying file HiggsDemoAnalyzer.cc...
    -> Expected size 83781, found 83781
    -> Expected checksum adler32:2f05f06, found adler32:2f05f06
  == Downloading file 3 of 11
    -> File: /5500/list_indexfile.txt
    -> Progress: 0/0 KIB (100%)
  == Verifying file list_indexfile.txt...
    -> Expected size 1665, found 1665
    -> Expected checksum adler32:46a997fc, found adler32:46a997fc
  == Downloading file 4 of 11
    -> File: /5500/M4NormDatafile.cc
    -> Progress: 14/14 KIB (100%)
  == Verifying file M4NormDatafile.cc...
    -> Expected size 14943, found 14943
    -> Expected checksum adler32:af301992, found adler32:af301992
```

```
@tiborsimko
```

Downloading metadata information

Downloading and verifying files
Working with large datasets

A dataset of 9.6 TB size, 26'958 files

Downloading file locations

Downloading one file for inspection
The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.

▶ I1. (Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.
▶ I2. (Meta)data use vocabularies that follow FAIR principles
▶ I3. (Meta)data include qualified references to other (meta)data
Data formats and vocabularies

A variety of data formats
ROOT as a community standard

No formal vocabularies
Some common classification

Filter by file type

<table>
<thead>
<tr>
<th>Filter</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>115</td>
</tr>
<tr>
<td>aod</td>
<td>849</td>
</tr>
<tr>
<td>aodsim</td>
<td>11</td>
</tr>
<tr>
<td>cc</td>
<td>933</td>
</tr>
<tr>
<td>csv</td>
<td>1</td>
</tr>
<tr>
<td>docx</td>
<td>1</td>
</tr>
<tr>
<td>fevtdebughl</td>
<td>1</td>
</tr>
<tr>
<td>gen-sim</td>
<td>95</td>
</tr>
<tr>
<td>gen-sim-digi-raw</td>
<td>1</td>
</tr>
<tr>
<td>gen-sim-reco</td>
<td>1</td>
</tr>
<tr>
<td>gz</td>
<td>1</td>
</tr>
<tr>
<td>h5</td>
<td>4</td>
</tr>
<tr>
<td>html</td>
<td>1</td>
</tr>
<tr>
<td>ig</td>
<td>1</td>
</tr>
<tr>
<td>ipynb</td>
<td>1</td>
</tr>
<tr>
<td>jpg</td>
<td>12</td>
</tr>
<tr>
<td>json</td>
<td>1</td>
</tr>
<tr>
<td>m4v</td>
<td>1</td>
</tr>
<tr>
<td>minlaodsim</td>
<td>1</td>
</tr>
<tr>
<td>nanoaod</td>
<td>1</td>
</tr>
<tr>
<td>ova</td>
<td>1</td>
</tr>
<tr>
<td>pdf</td>
<td>1</td>
</tr>
<tr>
<td>png</td>
<td>1</td>
</tr>
<tr>
<td>py</td>
<td>1</td>
</tr>
<tr>
<td>raw</td>
<td>1</td>
</tr>
<tr>
<td>reco</td>
<td>1</td>
</tr>
<tr>
<td>root</td>
<td>1</td>
</tr>
<tr>
<td>tar</td>
<td>1</td>
</tr>
<tr>
<td>tar.gz</td>
<td>1</td>
</tr>
<tr>
<td>txt</td>
<td>1</td>
</tr>
<tr>
<td>xls</td>
<td>1</td>
</tr>
<tr>
<td>xml</td>
<td>1</td>
</tr>
<tr>
<td>zip</td>
<td>1</td>
</tr>
</tbody>
</table>

Filter by category

<table>
<thead>
<tr>
<th>Category</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>B physics and Quarkonia</td>
<td>16</td>
</tr>
<tr>
<td>Exotica</td>
<td>18</td>
</tr>
<tr>
<td>Gravitons</td>
<td>1</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>1</td>
</tr>
<tr>
<td>Higgs Physics</td>
<td>28</td>
</tr>
<tr>
<td>Beyond Standard Model</td>
<td>17</td>
</tr>
<tr>
<td>Standard Model</td>
<td>19</td>
</tr>
<tr>
<td>Physics Modelling</td>
<td>18</td>
</tr>
<tr>
<td>Standard Model Physics</td>
<td>338</td>
</tr>
<tr>
<td>Drell-Yan</td>
<td>62</td>
</tr>
<tr>
<td>ElectroWeak</td>
<td>276</td>
</tr>
<tr>
<td>Forward and Small-x QCD Physics</td>
<td>2</td>
</tr>
<tr>
<td>Minimum Bias</td>
<td>1</td>
</tr>
<tr>
<td>QCD</td>
<td>134</td>
</tr>
<tr>
<td>Top physics</td>
<td>497</td>
</tr>
<tr>
<td>Supersymmetry</td>
<td>3</td>
</tr>
</tbody>
</table>
Data semantics

Dataset variables coming with detailed semantics description
Interoperability via accompanying examples

Dimuon spectrum analysis example

H → ττ analysis example
FAIR principles: R is for Reusable

The ultimate goal of FAIR is to optimise the reuse of data. To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.

- R1. (Meta)data are richly described with a plurality of accurate and relevant attributes
- R1.1. (Meta)data are released with a clear and accessible data usage license
- R1.2. (Meta)data are associated with detailed provenance
- R1.3. (Meta)data meet domain-relevant community standards
Simulated dataset BulkGravTohTohhbhhbb_narrow_M-4500_13TeV-madgraph in MINIAODSIM format for 2016 collision data

/CmsGravTohTohhbhhbb_narrow_M-4500_13TeV-madgraph/RunIISummer16MiniAODv2-
PUMoriond17_80X_mcRun2_asymptotic_2016_Tranche4_v6-v1/MINIADODSIM, CMS Collaboration


How were these data generated?

These data were generated in several steps (see also CMS Monte Carlo production overview):

Step LHE
Release: CMSSW_7_1_16
Output dataset: /BulkGravTohTohhbhhbb_narrow_M-4500_13TeV-madgraph/RunIIWinter15swmLHE-MCRUN2_71_V1-
V1/LHE
Note: To get the exact generator parameters, please see Finding the generator parameters.

Step SIM
Release: CMSSW_7_1_20
Configuration file for SIM (link)
Output dataset: /BulkGravTohTohhbhhbb_narrow_M-4500_13TeV-madgraph/RunIISummer15GS-MCRUN2_71_V1-
V1/GEN-SIM

Step HLT RECO
Release: CMSSW_R_0_31

Data records come with full provenance information

- full capture of data generation steps
- full capture of compute environments
- full capture of configuration files
- full capture of production scripts
Data provenance is part of the standard JSON metadata

```
"methodology": {
  "description": "These data were generated in several steps (see also <a href="/docs/cms-mc-production-overview">CMS Data Production Overview</a>)
  "steps": [
    {
      "configuration_files": [
        {
          "script": "/!/bin/bash\nsource /cvmfs/cms.cern.ch/cmsset_default.sh\nexport SCRAM_ARCH=slc5_amd64_gcc462\nif [ -r
          "title": "Production script"
        },
        {
          "title": "Generator parameters",
          "url": "https://cms-pdmv.cern.ch/mcm/public/restapi/requests/get_fragment/HIG-Summer12-02276"
        },
        {
          "cms_confdb_id": "a97a2f6c22dfba999c0131657a81efcfd",
          "process": "SIM",
          "title": "Configuration file"
        }
      ],
      "generators": [
        "pythia6"
      ],
      "global_tag": "START53_V7C::All",
      "output_dataset": "/BH_10To18_TuneZ2star_8TeV_pythia6_tautaua/Summer12-START53_V7C-v1/GEN-SIM",
      "release": "CMSSW_5_3_13",
      "type": "SIM"
    }
  ],
  "configuration_files": []
}
```

CERN Open Data portal standardises CMS DAS/McM information
Can we reproduce published data samples?

SingleElectron primary dataset sample in RAW format from RunA of 2011 (from /SingleElectron/Run2011A-v1/RAW)


Description
A sample from SingleElectron primary dataset in RAW format from RunA of 2011. Run range [161224,163286].

This dataset contains selected runs from 2011 RunA. The list of validated lumi sections, which must be applied to all analyses on events reconstructed from these data, can be found in

CMS list of validated runs Cert_160404-180252_7TeV_ReRecoNov08_Collisions11_JSON.txt

Dataset characteristics
2064298 events. 116 files. 424.3 GB in total.

How can you use these data?
These data are in RAW format and not directly usable in analysis. The reconstructed data reprocessed from these RAW data are included in the data of this record. The reconstruction step can be repeated with the configuration file below and the resulting AOD has been confirmed to be identical with the original one with comparison code available in Validation code to plot basic physics objects from AOD

RAW

AOD
Replicating original computing environments

ATLAS collaboration
https://hub.docker.com/r/atlas/analysisbase/tags

CMS collaboration
https://gitlab.cern.ch/cms-cloud/cmssw-docker

Docker and Singularity images help to encapsulate the computing environment
The Effects of FreeSurfer Version, Workstation Type, and Macintosh Operating System Version on Anatomical Volume and Cortical Thickness Measurements

Ed H. B. M. Gronenschild, Petra Habets, Heidi I. L. Jacobs, Ron Mengelers, Nico Rozendaal, Jim van Os, Machteld Marcelis

Published: June 1, 2012 • DOI: 10.1371/journal.pone.0038234

Does it really matter?

Software changes (Freesurfer 4.3.1, 4.5.0, 5.0.0): 8.8±6.6% (volume) and 2.8±1.3% (thickness)
Operating system changes (macOS 10.5, 10.6): about factor two smaller
Reprocessing workflow: computational recipe

3. Workflow

The workflow can be logically divided into several parts:

0. Upload all files.
   Some files cannot be generated at run time and need to be uploaded.
   
1. Fix the CMS SW environment variables manually.
   First, we have to set up the environment variables accordingly for the CMS SW. Although this is done in the docker image, reana overrides them and they need to be reset. This is done by invoking the cms entrypoint.sh script commands.
   
   See also this issue.

   $ source /opt/cms/cmsset_default.sh
   $ scramv1 project CMSSW CMSSW_9_3_32
   $ cd CMSSW_9_3_32/src
   $ eval 'scramv1 runtime -sh'

2. Create the specific CMS path.
   CMS specific data analysis framework requires two directory levels. See also this issue.

   $ mkdir Reconstruction && cd Reconstruction
   $ mkdir Validation && cd Validation

3. Create the reconstruction file.
   See also this repo.

   $ cmsDriver.py reco -- RAWDIGIT,L1Reco,RECO,USER:EventFilter/HCALRawsToDigit/HCALiselnsbheelffilter2012.cf

4. Adjust the reconstruction file to the specific data file.
   Although generated using parameters, the reconstruction file still requires changes.

   $ sed -i 's/Configuration/Alice.GlobalTag import GlobalTag(process.GlobalTag.connect = cms.string"'
   $ sed -i 's/# Other statements/Configuration.Alice.GlobalTag import GlobalTag/g' reco.cmsdriver.py

5. Link the CMSFW files.
   The is commands are explicitly needed to make sure that the cms-opendata-condb.cern.ch directory has actually expanded in the image, according to this guide. See also this issue.

   $ ln -sf /cvmfs/cms-opendata-condb.cern.ch/T1_UH_2019_AN1
   $ ln -sf /cvmfs/cms-opendata-condb.cern.ch/T1_UH_2019_AN1
   $ ln -sf /cvmfs/cms-opendata-condb.cern.ch/T1_UH_2019_AN1
   $ ln -sf /cvmfs/cms-opendata-condb.cern.ch/T1_UH_2019_AN1
   $ ln -s -i /cvmfs/

6. Run the reconstruction.
   At this point all environment variables and files should be proper.

   $ cmsRun reco.cmsdriver.py

7. Adjust project structure for validation.
   Copy the required files for the next steps.

   $ mkdir src
   $ cp .../.../.../.../AliceObjectHistos.cc .
   $ cp .../.../.../.../demoanalyzers.cfg .
   $ cp .../.../.../.../demoanalyzers.cfg .

8. Run CMS scram command to fix libraries.
   Most importandy, the BuildFile.xml has to be inside the directory where the scram command is executed.

   $ scram b

9. Run the validation file.
   See also this repo

   $ cmsRun demoanalyzers.cfg
Four pillars of reusable computational research

I. Input data
What is your input data?
- input files
- input parameters

II. Analysis code
Which code analyses it?
- user code
- software frameworks

III. Computing environment
What is your environment?
- operating system
- database calls

IV. Computational recipes
Which steps did you take?
- shell commands
- notebooks and workflows
REANA: Reusable analyses

Deploy and run containerised workflows on compute clouds
REANA in a nutshell

CLI UI

Web UI

Workflow controller

Job controller

Shared storage

Kubernetes

HTCondor

Slurm

REST API

myschedd

sbatch
Computational workflows

ATLAS  http://cdsweb.cern.ch/record/2714064

Data analysis example: ATLAS displaced jet search reinterpretation

CMS  https://github.com/alintulu/reana-demo-JetMETAnalysis

Data production example: CMS jet energy resolution and corrections
Reprocessing CMS datasets on REANA

1. input parameters

2. workflow factory

3. reana.yaml

4. run by REANA platform

5. serving open data files

6. output histograms

Parametrised workflow runnable on REANA reproducible analysis platform
Preservation ⇔ Reusability

- Reusability
- Preservation
- Reuse
- Use
- Preserve
- Digital repositories
- Reusable workflows
Conclusions
Open is not enough


The solution adopted by the energy physics community is fabulous research is examples of best practices that could improve more widely. This first experience suggests that reproducibility requires going beyond open access.

On science and reproducible research has become popular goal in government, academia, and industry funding bodies. The understanding that open science and reproducible research stimulate scientific trust, accelerate follow-up projects and decrease or even eliminate. The effort to make research reproducible is of particular importance in the field of physics, where the complexity of the experiments and the need for precise measurements demand that the research findings are verified and replicated. This requires not only sharing the data, but also making the code, environment, and workflow publicly available.

Approaching reproducibility and making it happen To the major of the open science community, we first encounter a more personal viewpoint in which the reproducibility challenges seem to be inherently related to the specific projects and the individuals involved. However, when we look at the overall research landscape, it becomes clear that reproducibility is a systemic issue. The open science movement, in its efforts to improve reproducibility, has focused on the technical aspects of data sharing and publication. However, reproducibility is not just about the technical infrastructure. It is also about the culture and practices that shape the way research is done. The role of institutions and funding bodies in promoting reproducibility is crucial. They can provide incentives for researchers to invest in the development of reproducible practices and fund projects that focus on reproducibility.

Reproducible research has the potential to enhance the quality and impact of research, but it also presents challenges. There is a need for a culture shift in the scientific community, where researchers are encouraged to adopt reproducible practices and be held accountable for them. This requires a collaborative effort between researchers, institutions, and funding bodies. The open science community has a role to play in this by promoting the sharing of data, code, and workflows, and by creating incentives for reproducibility.

Table 1: Technology-related reproducible research introduced by Cambria Systems and IBM

<table>
<thead>
<tr>
<th>Type</th>
<th>Region</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Michael</td>
<td>Variables in predictive and cost evaluation</td>
</tr>
<tr>
<td>Code</td>
<td>Michael</td>
<td>High-stake experiment and replication</td>
</tr>
<tr>
<td>Environment</td>
<td>Cambria</td>
<td>Open-source software for independent use</td>
</tr>
<tr>
<td>Workflows</td>
<td>Cambria</td>
<td>Variables in predictive and cost evaluation</td>
</tr>
<tr>
<td>Transfer</td>
<td>IBM</td>
<td>Open-source software for independent use</td>
</tr>
</tbody>
</table>

The open science community has a role to play in promoting reproducibility, education, and collaboration. By working together, we can make research more accessible, reproducible, and impactful.