5.07.2023

Digitisation and Reconstruction Hands on

MuCol: training on the detector design and physics performance tool

L. Buonincontri

INFN Padova, University of Padova

Simulation and reconstruction chain

- Simulation step ⇨ done:

  Output file: ~/work/mucoll-tutorial-2023/sim_hbb/Hbb_out.slcio

Now we will se how to:

- Run the **digitisation** step: converting energy deposits in the detector to realistic detector signals;
- Run the **reconstruction** step: individual particles and higher-level objects using dedicated algorithms;

Digitisation and reconstruction

- These stages are executed under the Marlin framework (Modular Analysis and Reconstruction for the LINear Collider)
- Every computing task (digitisation, tracks reconstruction etc..) is implemented as a **processor**

- In order to run digitisation and reconstruction steps a XML steering file have to be built with:
  - A list of processors (and their order) that you want to execute
  - Named parameters (string, float, int - single and arrays) for every processor as well as for the global scope

- You can produce an example of steering file with a list of all the available processors and all the available parameters with:

```
Marlin -x > steering_sample.xml
```

Digitization and reconstruction steering files

The digitisation and reconstruction processes are configured as two independent stages with individual steering files, which will be executed one after another:

- **`mucoll-benchmarks/digitisation/marlin/digi_steer.xml`**
  for the digitisation step

- **`mucoll-benchmarks/reconstruction/marlin/reco_steer.xml`**
  for the reconstruction step

Scheme of a Marlin steering file

Three main sections in the Marlin .xml steering file

1) Execute section (ordered list of processors to be executed)

```
<execute>
<processor name="MyAIDAProcessor"/>
<processor name="MyTestProcessor"/>
<processor name="MyLCIOOutputProcessor"/>
</execute>
```

2) Global section (global settings)

```
<global>
<parameter name="LCIOInputFiles"> input.slcio </parameter>
<parameter name="MaxRecordNumber" value="1000" />
</global>
```

3) Processor section (processor configuration)

```
<processor name="MyLCIOOutputProcessor" type="LCIOOutputProcessor">
<parameter name="LCIOOutputFile" type="string"> Output_DST.slcio </parameter>
<parameter name="DropCollectionTypes" type="StringVec">
SimCalorimeterHit
SimTrackerHit
</parameter>
<parameter name="LCIOWriteMode" type="string" value="WRITE_NEW"/>
<parameter name="SplitFileSizekB" type="int">1048576 </parameter>
<parameter name="Verbosity" type="string">WARNING </parameter>
</processor>
```

Processors type for digitisation

mucoll-benchmarks/digitisation/marlin/digi_steer.xml

InitializeDD4hep

- Geometry initialization

DDPlanarDigiProcessor

- Tracker: gaussian smearing of SIM hits positions and times, time window selection

DDCaloDigi

- Calorimeters: simple digitisation with an energy calibration constant, an energy threshold and a selection time window
- Muon detectors: simple digitisation with an energy calibration constant and an energy threshold

LCIOOutProcessor

- Fill a .slcio output file with collections

```xml
<!-- ========== TrackerDigitisation.xml ========== -->
<processor name="VXDBarrelDigitiser"/>
<processor name="VXDEndcapDigitiser"/>
<processor name="ITBarrelDigitiser"/>
<processor name="ITEndcapDigitiser"/>
<processor name="OTBarrelDigitiser"/>
<processor name="OTEndcapDigitiser"/>

<!-- ========== CaloDigitisation.xml ============ -->
<processor name="CaloDigitiser"/>
<!-- ========== MuonDigitisation.xml ============ -->
<processor name="MuonDigitiser"/>

<!-- ========== Output ========== -->
<processor name="LCIOWriter_all"/>
<processor name="LCIOWriter_light"/>
```

mucoll-benchmarks/digitisation/marlin/digi_steer.xml

Execute section: it contains the **name** of processors to be run for digitisation

```xml
<global>
  <parameter name="LCIOInputFiles">input.slcio</parameter>
  <!-- Limit the number of processed records -->
  <parameter name="MaxRecordNumber" value="-1" />
  <parameter name="SkipNEvents" value="0" />
  <parameter name="SupressCheck" value="false" />
  <parameter name="Verbosity" options="DEBUG0-9,MESSAGE0-9,WARNING0-9,ERROR0-9,SILENT">MESSAGE </parameter>
  <parameter name="RandomSeed" value="1234567890" />
</global>
```

Global section: input file containing the simulated hits that have to be digitised, set the number of events you want to reconstruct

```xml
<!-- Including processor definitions from external files -->
<include ref="subconfigs/DD4hep.xml"/>
<include ref="subconfigs/Overlay.xml"/>
<include ref="subconfigs/TrackerDigitisation.xml"/>
<include ref="subconfigs/CaloDigitisation.xml"/>
<include ref="subconfigs/MuonDigitisation.xml"/>
<include ref="subconfigs/VertexDoubleLayerFiltering.xml"/>
```

Processor section: it contains the link to xml files where these processors are defined and definition of output processor (next slide)

Definition of the output processor

mucoll-benchmarks/digitisation/marlin/digi_steer.xml

```xml
<!-- LCIO output: keep all collections -->
<processor name="LCIOWriter_all" type="LCIOOutputProcessor">
  <parameter name="LCIOOutputFile" type="string"> out.slcio </parameter>
  <parameter name="FullSubsetCollections" type="StringVec">  </parameter>
  <parameter name="DropCollectionTypes" type="StringVec">  </parameter>
  <parameter name="DropCollectionNames" type="StringVec">  </parameter>
  <parameter name="KeepCollectionNames" type="StringVec">  </parameter>
  <parameter name="LCIOWriteMode" type="string" value="WRITE_NEW"/>
  <parameter name="Verbosity" type="string">WARNING </parameter>
  <!-- <parameter name="SplitFileSizekB" type="int">996147 </parameter> -->
</processor>
```

All the collections are saved in out.slcio: Monte Carlo particles, simulated hits and digitized hits

```xml
<!-- LCIO output: keep only collections relevant for analysis -->
<processor name="LCIOWriter_light" type="LCIOOutputProcessor">
  <parameter name="LCIOOutputFile" type="string"> out_light.slcio </parameter>
  <parameter name="FullSubsetCollections" type="StringVec"> </parameter>
  <!-- Removing SimHits, MCParticles and all the relation info -->
  <parameter name="DropCollectionTypes" type="StringVec">
    SimTrackerHit
    SimCalorimeterHit
    LCRelation
  </parameter>
  <parameter name="DropCollectionNames" type="StringVec">
    MCParticle
  </parameter>
  <parameter name="KeepCollectionNames" type="StringVec">  </parameter>
  <parameter name="LCIOWriteMode" type="string" value="WRITE_NEW"/>
  <parameter name="Verbosity" type="string">WARNING </parameter>
</processor>
```

Some collections are dropped in out_light.slcio: only the digitized hits are kept

Example of processors definition files

mucoll-benchmarks/digitisation/marlin/subconfigs/TrackerDigitisation.xml

```
<processor name="ITBarrelDigitiser" type="DDPlanarDigiProcessor">
  <parameter name="SubDetectorName" type="string"> InnerTrackers </parameter>
  <!--whether hits are 1D strip hits-->
  <parameter name="IsStrip" type="bool"> false </parameter>
  <!--resolution in direction of u-->
  <parameter name="ResolutionU" type="float"> 0.007 </parameter>
  <!--resolution in direction of v-->
  <parameter name="ResolutionV" type="float"> 0.09 </parameter>
  <!--Name of the Input SimTrackerHit collection-->
  <parameter name="SimTrackHitCollectionName" type="string" lcioInType="SimTrackerHit"> InnerTrackerBarrelCollection </parameter>
  <!--Name of TrackerHit -> SimTrackHit relation collection-->
  <parameter name="SimTrkHitRelCollection" type="string" lcioOutType="LCRelation"> ITBarrelHitsRelations </parameter>
  <!--Name of the TrackerHit output collection-->
  <parameter name="TrackerHitCollectionName" type="string" lcioOutType="TrackerHitPlane"> ITBarrelHits </parameter>
  <!--resolution in time-->
  <parameter name="ResolutionT" type="FloatVec"> 0.06  </parameter>
  <!--resolution in direction of u - either one per layer or one for all layers -->
  <parameter name="UseTimeWindow" type="bool"> true </parameter>
  <!--Correct hit times for propagation: radial distance/c-->
  <parameter name="CorrectTimesForPropagation" type="bool" value="true"/>
  <!--Lower bound of the time window [ns]-->
  <parameter name="TimeWindowMin" type="float"> -0.18 </parameter>
  <!--Upper bound of the time window [ns]-->
  <parameter name="TimeWindowMax" type="float"> 0.3 </parameter>
  <!--verbosity level of this processor ("DEBUG0-4,MESSAGE0-4,WARNING0-4,ERROR0-4,SILENT")-->
  <parameter name="Verbosity" type="string"> WARNING </parameter>
</processor>
```

Inner Tracker barrel digitisation, parameters of the processor can be changed

HANDS ON 1

Let's try to run the digitisation steering file

- Create a directory for digitisation

```
mkdir digi_Hbb && cd digi_Hbb
```

- Then run digitization:

```
Marlin ../mucoll-benchmarks/digitisation/marlin/digi_steer.xml \
--global.LCIOInputFiles="../sim_Hbb/Hbb_out.slcio" \
--DD4hep.DD4hepXMLFile="$MUCOLL_GEO"
```

After successful execution of the Marlin process it will produce three output files:

- output_digi.slcio - contains all the collections produced by the executed processors
- output_digi_light.slcio - contains a subset of output collections, which are relevant for later analysis
- histograms.root - contains some diagnostics plots and trees

Anajob out_light.slcio

Check which collections are contained in the .slcio files:

anajob output_digi.slcio

anajob output_digi_light.slcio

```
RUN: -1          EVENT: 9          DETECTOR: MuColl_v1
---------------------------------------------------------------------------
COLLECTION NAME                   COLLECTION TYPE              # OF ELEMENTS
===========================================================================
ECALBarrelHits                    CalorimeterHit                          96
ECALEndcapHits                    CalorimeterHit                         155
HCALBarrelHits                    CalorimeterHit                          37
HCALEndcapHits                    CalorimeterHit                         130
HCALOtherHits                     CalorimeterHit                           2
ITBarrelHits                      TrackerHitPlane                         74
ITEndcapHits                      TrackerHitPlane                        103
MuonHits                          CalorimeterHit                          13
OTBarrelHits                      TrackerHitPlane                         31
OTEndcapHits                      TrackerHitPlane                         79
VXDBarrelHits                     TrackerHitPlane                        109
VXDEndcapHits                     TrackerHitPlane                        164
---------------------------------------------------------------------------
```

Processors in the reconstruction steering file

mucoll-benchmarks/reconstruction/marlin/reco_steer.xml

TRACK RECONSTRUCTION:

ACTSSeededCKFTrackingProc
ACTSDuplicateRemoval
RefitFinal

- Combinatorial Kalman Filter algorithm implemented using ACTS library. Takes in input digitized tracker hits to build reconstructed tracks, remove duplicates and apply quality requirements on tracks.

arXiv:2106.13593

```
<!-- ========== TrackReconstruction.xml ========== -->
<processor name="CKFTracking"/>
<processor name="TrackDeduplication"/>
<processor name="TrackRefit"/>

<!-- =============== PFOReconstruction.xml ================= -->
<processor name="DDMarlinPandora" />

<!-- =============== PFOSelection.xml ================= -->
<group name="PFOSelection" />

<!-- =============== VertexJet.xml ================ -->
<processor name="FastJetProcessor" />
```

```
<!-- Including processor definitions from external files -->
<include ref="subconfigs/DD4hep.xml"/>
<include ref="subconfigs/TrackReconstruction.xml"/>
<include ref="subconfigs/PFOReconstruction.xml"/>
<include ref="subconfigs/PFOSelection.xml"/>
<include ref="subconfigs/VertexJet.xml"/>
```

Processors in the reconstruction steering file

mucoll-benchmarks/reconstruction/marlin/reco_steer.xml

PARTICLE RECONSTRUCTION:

DDPandoraPFANewProcessor
CLICPfoSelector

- Takes in input reconstructed tracks and calorimeter digitized hits.
- Uses Pandora Particle Flow Algorithm (arXiv:1308.4537) to recognize different patterns of hits released by different particle types

```
<!-- ========== TrackReconstruction.xml ========== -->
<processor name="CKFTracking"/>
<processor name="TrackDeduplication"/>
<processor name="TrackRefit"/>

<!-- =============== PFOReconstruction.xml ================= -->
<processor name="DDMarlinPandora" />

<!-- =============== PFOSelection.xml ================= -->
<group name="PFOSelection" />

<!-- =============== VertexJet.xml ================ -->
<processor name="FastJetProcessor" />
```

```
<!-- Including processor definitions from external files -->
<include ref="subconfigs/DD4hep.xml"/>
<include ref="subconfigs/TrackReconstruction.xml"/>
<include ref="subconfigs/PFOReconstruction.xml"/>
<include ref="subconfigs/PFOSelection.xml"/>
<include ref="subconfigs/VertexJet.xml"/>
```

Processors in the reconstruction steering file

`mucoll-benchmarks/reconstruction/marlin/reco_steer.xml`

JETS RECONSTRUCTION:

FastJetProcessor

- Takes in input reconstructed particles
- Uses kt-algorithm to reconstruct jets with cone 0.7

```xml
<!-- ========== TrackReconstruction.xml ========== -->
<processor name="CKFTracking"/>
<processor name="TrackDeduplication"/>
<processor name="TrackRefit"/>

<!-- =============== PFOReconstruction.xml ================= -->
<processor name="DDMarlinPandora" />

<!-- =============== PFOSelection.xml ================= -->
<group name="PFOSelection" />

<!-- =============== VertexJet.xml ================ -->
<processor name="FastJetProcessor" />
```

```xml
<!-- Including processor definitions from external files -->
<include ref="subconfigs/DD4hep.xml"/>
<include ref="subconfigs/TrackReconstruction.xml"/>
<include ref="subconfigs/PFOReconstruction.xml"/>
<include ref="subconfigs/PFOSelection.xml"/>
<include ref="subconfigs/VertexJet.xml"/>
```

HANDS ON 2

Run the reconstruction steering file:

- Create a directory for digitisation

```
mkdir reco_Hbb && cd reco_Hbb
```

- Create a link to the Pandora configuration files:

```
ln -s ../mucoll-benchmarks/reconstruction/marlin/PandoraSettings ./
```

- Determine location of ActsTracking processor that contains relevant parts of the default MuColl_v1 geometry converted to the format required by ACTS track-reconstruction framework

```
ACTS_PATH=$(echo $MARLIN_DLL | tr ':' '\n' | grep actstracking | sed "s:/lib.*::")
```

- Then run the reconstruction

```
Marlin ../mucoll-benchmarks/reconstruction/marlin/reco_steer.xml \
--global.LCIOInputFiles="../digi_Hbb/output_digi.slcio" \
--CKFTracking.MatFile="${ACTS_PATH}/share/ACTSTracking/data/material-maps.json" \
--CKFTracking.TGeoFile="${ACTS_PATH}/share/ACTSTracking/data/MuColl_v1.root" \
--DD4hep.DD4hepXMLFile="$MUCOLL_GEO"
```

HANDS ON 2

- Check which collections are contained in the .slcio files

```
anajob output_reco.slcio

anajob output_reco_light.slcio
```

```
RUN: -1          EVENT: 9          DETECTOR: MuColl_v1
--------------------------------------------------------------------------
COLLECTION NAME                    COLLECTION TYPE             # OF ELEMENTS
==========================================================================
AllTracks                          Track                                  69
ECALBarrelHits                     CalorimeterHit                         96
ECALEndcapHits                     CalorimeterHit                        155
HCALBarrelHits                     CalorimeterHit                         37
HCALEndcapHits                     CalorimeterHit                        130
HCALOtherHits                      CalorimeterHit                          2
JetOut                             ReconstructedParticle                   3
MuonHits                           CalorimeterHit                         13
PandoraClusters                    Cluster                                10
PandoraPFOs                        ReconstructedParticle                  20
PandoraStartVertices               Vertex                                 20
SeedTracks                         Track                                  69
SiTracks                           Track                                  26
SiTracks_Refitted                  Track                                  25
--------------------------------------------------------------------------
```

HANDS ON 2

- Access to reconstructed particles information with

dumpevent output_reco.slcio -1 1 > file.txt

```
-------------- print out of Track collection ---------------

 flag:  0x0
    LCIO::TRBIT_HITS : 0

 [   id   ] |    type    |     d0     |  phi      | omega     |     z0      | tan lambda|
hi2    |  ndf
------------|-----------|----------|----------|----------|-----------|-----------|
------
 [00002344] | 00000000 |-1.52e+00 |-3.85e-01 |+6.64e-04 |-1.221e+01 |-4.041e-01 |
```

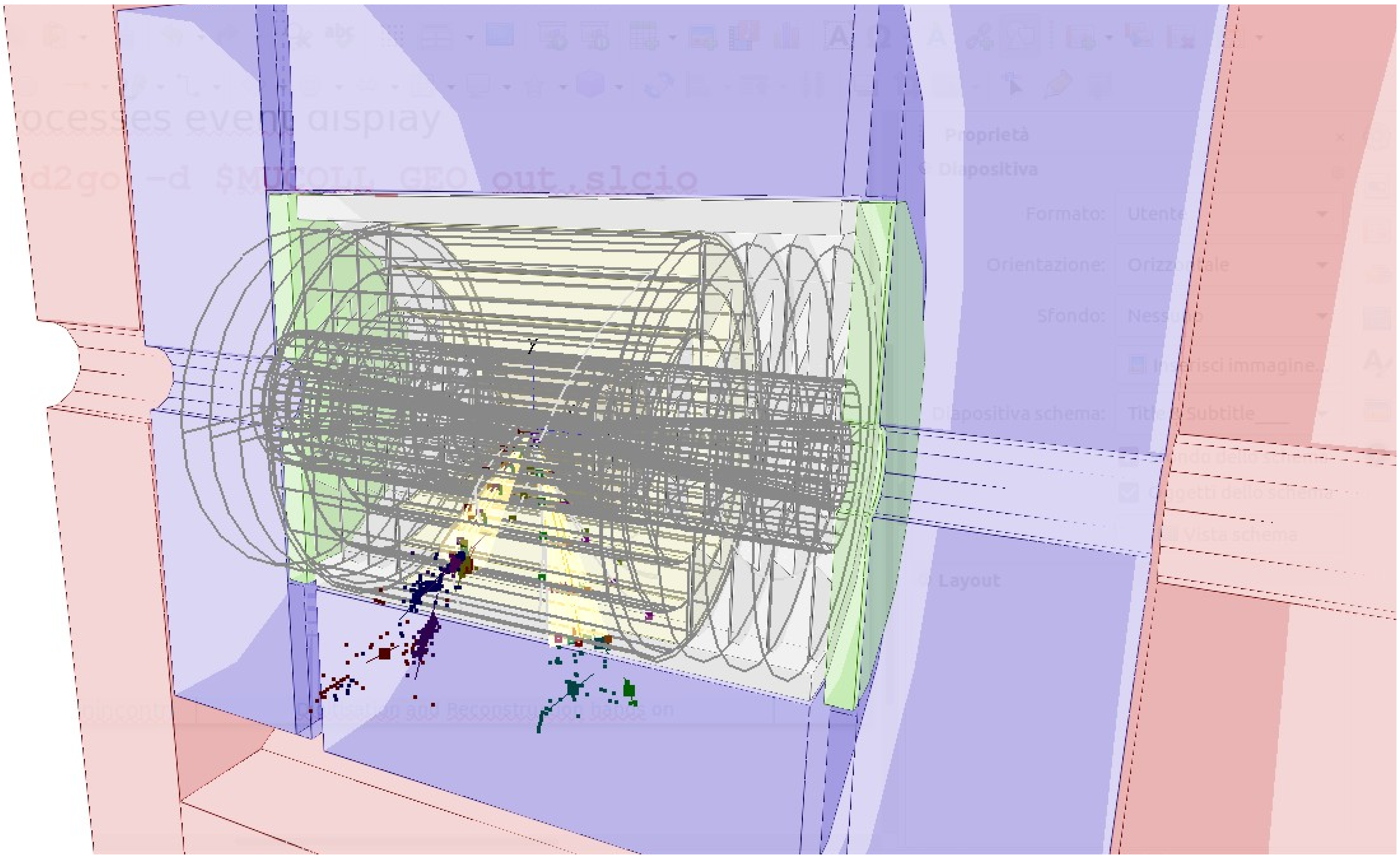
```
collection name : JetOut
parameters:

-------------- print out of ReconstructedParticle collection --------------

 flag:  0x0

[   id   ] |com|type|     momentum( px,py,pz)       | energy | mass   | charge |        position ( x,y,z)       |pidUsed|GoodnessOfPID|
------------|---|----|-------------------------------|--------|--------|--------|-------------------------------|-------|-------------|
[00000928] |  1|   0|+5.32e+01, +8.38e+01, +2.43e+01|1.02e+02|6.93e+00|0.00e+00|+0.00e+00, +0.00e+00, +0.00e+00|0000000|     0.00e+00|

   covariance( px,py,pz,E) : (0.00e+00, 0.00e+00, 0.00e+00, 0.00e+00, 0.00e+00, 0.00e+00, 0.00e+00, 0.00e+00, 0.00e+00, 0.00e+00)
   particles ( [   id   ] ):[00001030], [00001025], [00001029], [00001031], [00001019], [00001036], [00001028], [00001033]
```

- Momentum and energy of reconstructed Particles, tracks parameters and jets are saved in .slcio
- In next part of the tutorial Chiara will show you how to make ROOT ntuples!

HANDS ON 2

- Processes event display

```
ced2go -d $MUCOLL_GEO output_reco.slcio
```



BACKUP

Reconstruction output files

Useful commands:

- Number of events saved in an output file:

```
lcio_event_counter Output_REC.slcio
```

- List of collections saved in the slcio files:

```
anajob Output_REC.slcio
```

- Dump of collections' content:

```
dumpevent Output_REC.slcio
```

(more infos executing `dumpevent -h`).

Muon Collider software setup

Hands-on instructions can be found in:

1) Set up a working software environment

```
ssh ${USER}@lxplus9.cern.ch
```

2) Create a working directory that you will use for the tutorial

```
mkdir -p ~/work/mucoll-tutorial-2023
cd ~/work/mucoll-tutorial-2023
```

3) Link the environment-setup and execute the environment-setup

```
ln -s /cvmfs/muoncollider.cern.ch/release/2.8-patch2/setup.sh ./
source setup.sh
```

Muon Collider software setup

4) Download the configuration files and analysis tools:

```
git clone https://github.com/MuonColliderSoft/mucoll-benchmarks.git
```

Now you should have your working directory containing:

- mucoll-benchmarks
- setup.sh