



July, 5-6th 2023
CERN

MuCol: training on detector design
and physics performance tools

Hands on: event analysis

Chiara Aimè
University of Pavia

Analysis setup

For this part of the tutorial we will also use code that is available in another repository

<https://github.com/MuonColliderSoft/MuC-Tutorial>

The repository has to be clone in the working directory

```
git clone https://github.com/MuonColliderSoft/MuC-Tutorial.git
```

Analysis methods

Output of the detector simulation and reconstruction can be analyzed using different approaches:

1. [read LCIO files using a Marlin processor](#)
2. read LCIO files using a Python script <https://mcdwiki.docs.cern.ch/tutorials/cern2023/analysis/python/>
3. converting basic objects to a ROOT tuple <https://mcdwiki.docs.cern.ch/tutorials/cern2023/analysis/histograms/>

Input file: results from the reconstruction available in this directory

```
/cvmfs/muoncollider.cern.ch/datasets/tutorial_20230705/reco_Hbb
```

Read LCIO files using a Marlin processor: LCTuple

LCTuple: configurable flat ROOT tree produced from the collections in the LCIO files

↳ based on Marlin processor - configured via an xml steering file -
in which collections to be written in the ntuple are selected

examples: <https://github.com/MuonColliderSoft/LCTuple>

LCTuple format

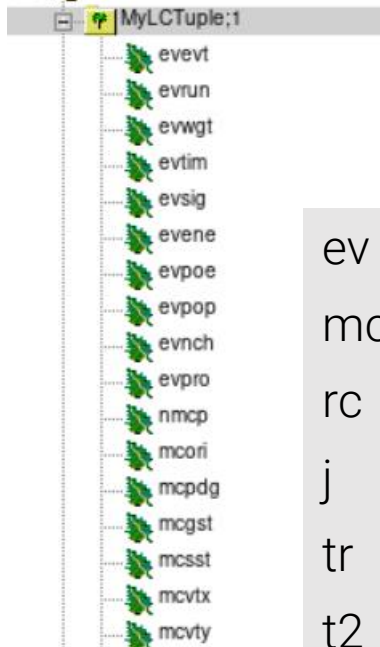
LCTuple: configurable flat ROOT tree produced from the collections in the LCIO files



each collection is saved as a different branch of the LCTuple tree

VARIABLE NAMING CONVENTION

lctuple_example.root



XXXXX
└──┬──┘
branch specific
variable

ev	global event info
mc	MonteCarlo particles
rc	reconstructed PF objects
j	jets
tr	reconstructed tracks
t2	link to the MC truth

Quick decode

<https://github.com/MuonColliderSoft/LCTuple/tree/master/src>

Useful vademecum

<https://atlaswiki.lbl.gov/analysis/muoncollider/lctuple>

LCTuple production

LCTuple: configurable flat ROOT tree produced from the collections in the LCIO files

↳ based on Marlin processor - configured via an xml steering file -
in which collections to be written in the ntuple are selected

`lctuple_steer.xml`

Essential information

1. names of the LCIO collections to be dumped into the ROOT tree

```
<parameter name="RecoParticleCollection" type="string" lcioInType="ReconstructedParticle">  
PandoraPFOs </parameter>
```

2. input filename

```
<parameter name="LCIOInputFiles"> input_file.slcio </parameter>
```

3. output filename

```
<parameter name="FileName" type="string" value="lctuple_example"/>  
<parameter name="FileType" type="string" value="«root"/>
```

LCTuple production

lctuple_steer.xml

Essential information

1. names of the LCIO collections to be dumped into the ROOT tree

```
<parameter name="RecoParticleCollection" type="string" lcioInType="ReconstructedParticle">  
PandoraPFOs </parameter>
```

2. input filename

```
<parameter name="LCIOInputFiles"> input_file.slcio </parameter>
```

3. output filename

```
<parameter name="FileName" type="string" value="lctuple_example"/>  
<parameter name="FileType" type="string" value="root"/>
```

Command to run

```
Marlin --global.LCIOInputFiles=input_file.slcio  
--MyAIDAProcessor.FileName=lctuple_example  
lctuple_steer.xml
```

LCTuple access and analysis

LCTuple is a configurable flat ROOT tree that can be analysed

1. directly with ROOT through TBrowser
2. with a ROOT macro

`lctuple_analysis.C`

```
root -l lctuple_analysis.C+
```


BACKUP

Useful tools

The default output format of the reconstruction package is an LCIO file.
It can be inspected using these commands:

1. to print the list of saved collection and their sizes

```
anajob out.slcio
```

2. to see the details of all the collection of the first event

```
dumpevent out.slcio 1 > event1.txt
```

If you are interest only in the number of events saved use

```
lcio_event_counter out.slcio
```

You can also merge or split an LCIO files

```
$ lcio_merge_files -h  
usage: lcio_merge_files <output-file>  
<input-file1> [[input-file2],...]
```

```
$ lcio_split_file -h  
usage: lcio_splitfile infilename  
outfilename sizeInBytes
```

Attention: changes in repository

Make sure to apply these two changes in order to run the analysis

1. l.68 in `lctuple_steer.xml`

```
<parameter name="JetCollection" type="string" lcioInType="ReconstructedParticle"> JetOut  
</parameter>
```