Selector: AGATA+${ancillary} analysis

D. Brugnara

M. Sedlak

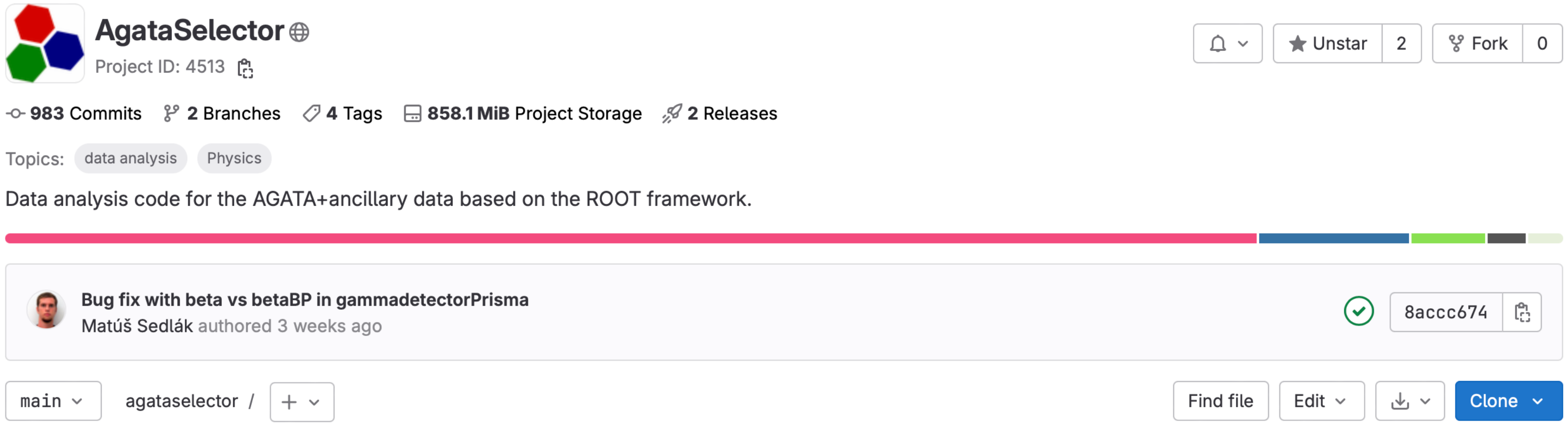
Outline

- Installation and introduction
- Setup of the analysis
- How to add histograms and personalize the analysis
- Detector specific analysis:
    - Prisma
    - Spider
    - Dante
    - LaBr
    - Euclides
- Optimization
- Reading raw ancillary data and building ancillary events
- Other useful scripts

Installation and introduction

Installation procedure

- The repository can be found here: https://baltig.infn.it/gamma/agataselector



AgataSelector

Project ID: 4513

983 Commits 2 Branches 4 Tags 858.1 MiB Project Storage 2 Releases

Topics: data analysis Physics

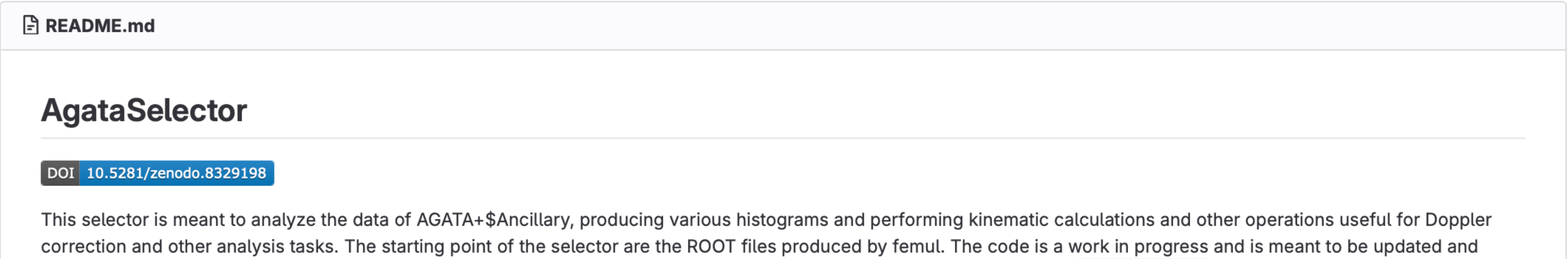
Data analysis code for the AGATA+ancillary data based on the ROOT framework.

Bug fix with beta vs betaBP in gammadetectorPrisma

Matúš Sedlák authored 3 weeks ago

8accc674

- Some info and instructions are contained in the README.md file



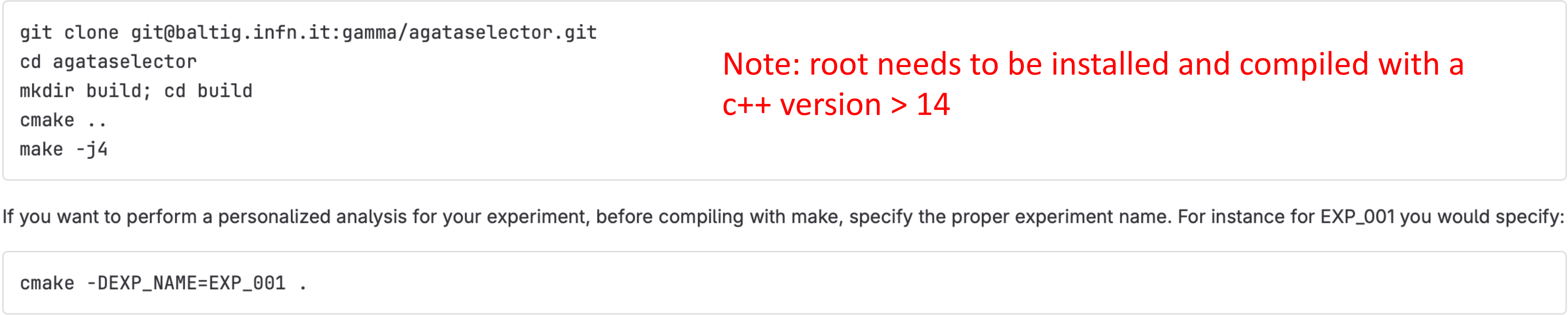
README.md

AgataSelector

DOI 10.5281/zenodo.8329198

This selector is meant to analyze the data of AGATA+$Ancillary, producing various histograms and performing kinematic calculations and other operations useful for Doppler correction and other analysis tasks. The starting point of the selector are the ROOT files produced by femul. The code is a work in progress and is meant to be updated and

Installation procedure

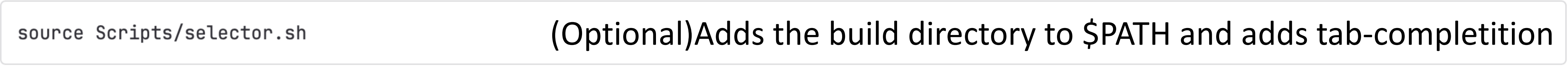


```
git clone git@baltig.infn.it:gamma/agataselector.git
cd agataselector
mkdir build; cd build
cmake ..
make -j4
```

Note: root needs to be installed and compiled with a c++ version > 14

If you want to perform a personalized analysis for your experiment, before compiling with make, specify the proper experiment name. For instance for EXP_001 you would specify:

```
cmake -DEXP_NAME=EXP_001 .
```



```
source Scripts/selector.sh
```

(Optional)Adds the build directory to $PATH and adds tab-completition

Known troubles and how to solve them

- Make sure to compile root with at least c++14:

```
- DCMAKE_CXX_STANDARD=17 (or 14 if you prefer)
```

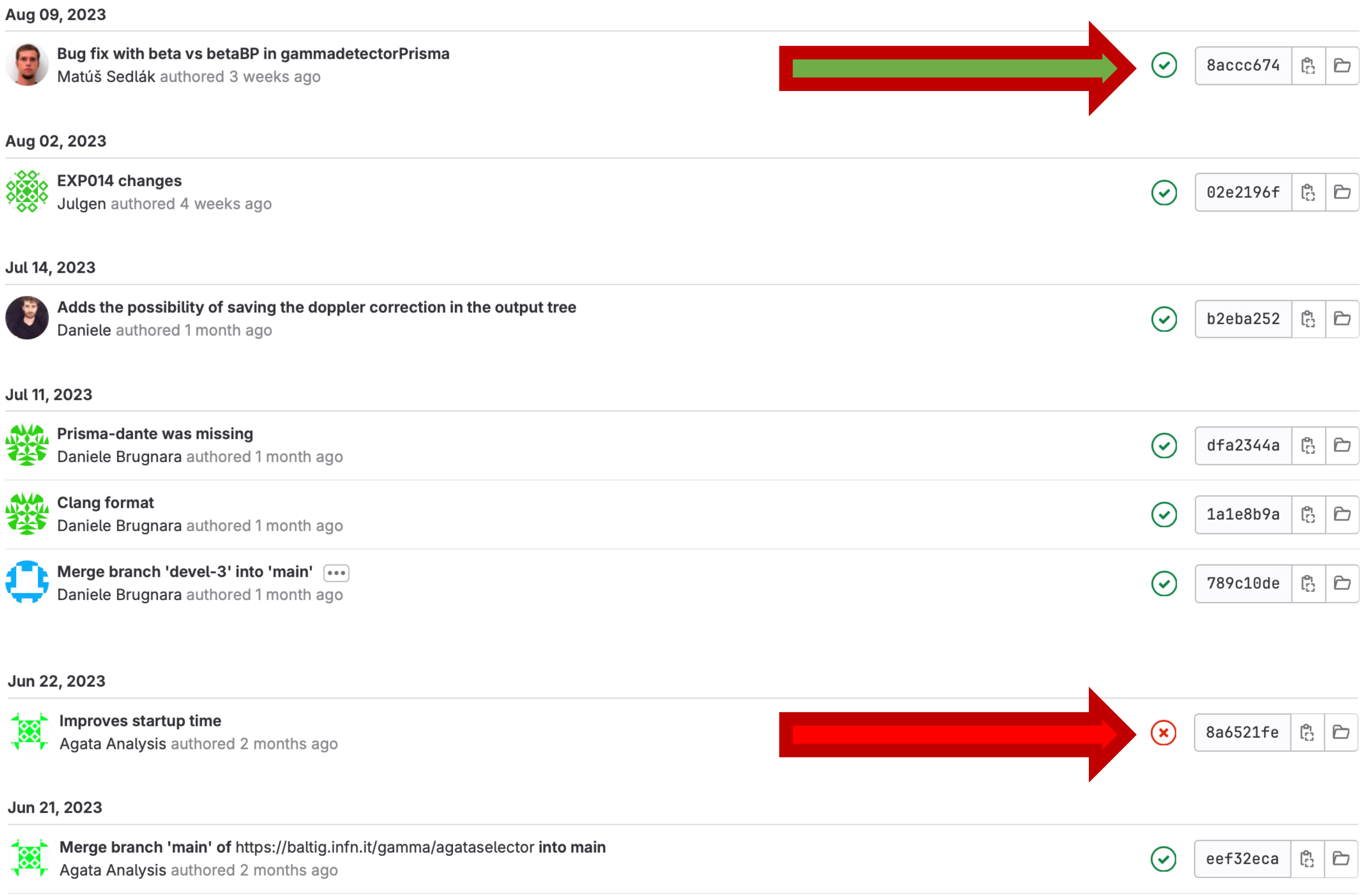
- On mac a compilation error, dependent on root, might appear. This is a known bug and requires root to be compiled with the built-in version of nlohmann:

```
-Dbuiltin_nlohmannjson=ON
```

- There is an issue with libPrisma that has the same name of femul. In this case you need to source Scripts/selector.sh

Tests

- At each commit the code is tested.
- The test include:
  - Does the code build properly?
  - Additionally does the prisma-agata analysis work?
  - At last, two spectra (DC, DCBP) are fitted for fully identified isotopes
  - The test passes if integral, mean and sigma of the peaks of interest is within the parameters
- This helps us in controlling that nothing was broken in the process
- An X marked commit means that the test was not passed



Aug 09, 2023

Bug fix with beta vs betaBP in gammadetectorPrisma
Matúš Sedlák authored 3 weeks ago ✓ 8accc674

Aug 02, 2023

EXP014 changes
Julgen authored 4 weeks ago ✓ 02e2196f

Jul 14, 2023

Adds the possibility of saving the doppler correction in the output tree
Daniele authored 1 month ago ✓ b2eba252

Jul 11, 2023

Prisma-dante was missing
Daniele Brugnara authored 1 month ago ✓ dfa2344a

Clang format
Daniele Brugnara authored 1 month ago ✓ 1a1e8b9a

Merge branch 'devel-3' into 'main' •••
Daniele Brugnara authored 1 month ago ✓ 789c10de

Jun 22, 2023

Improves startup time
Agata Analysis authored 2 months ago ✗ 8a6521fe

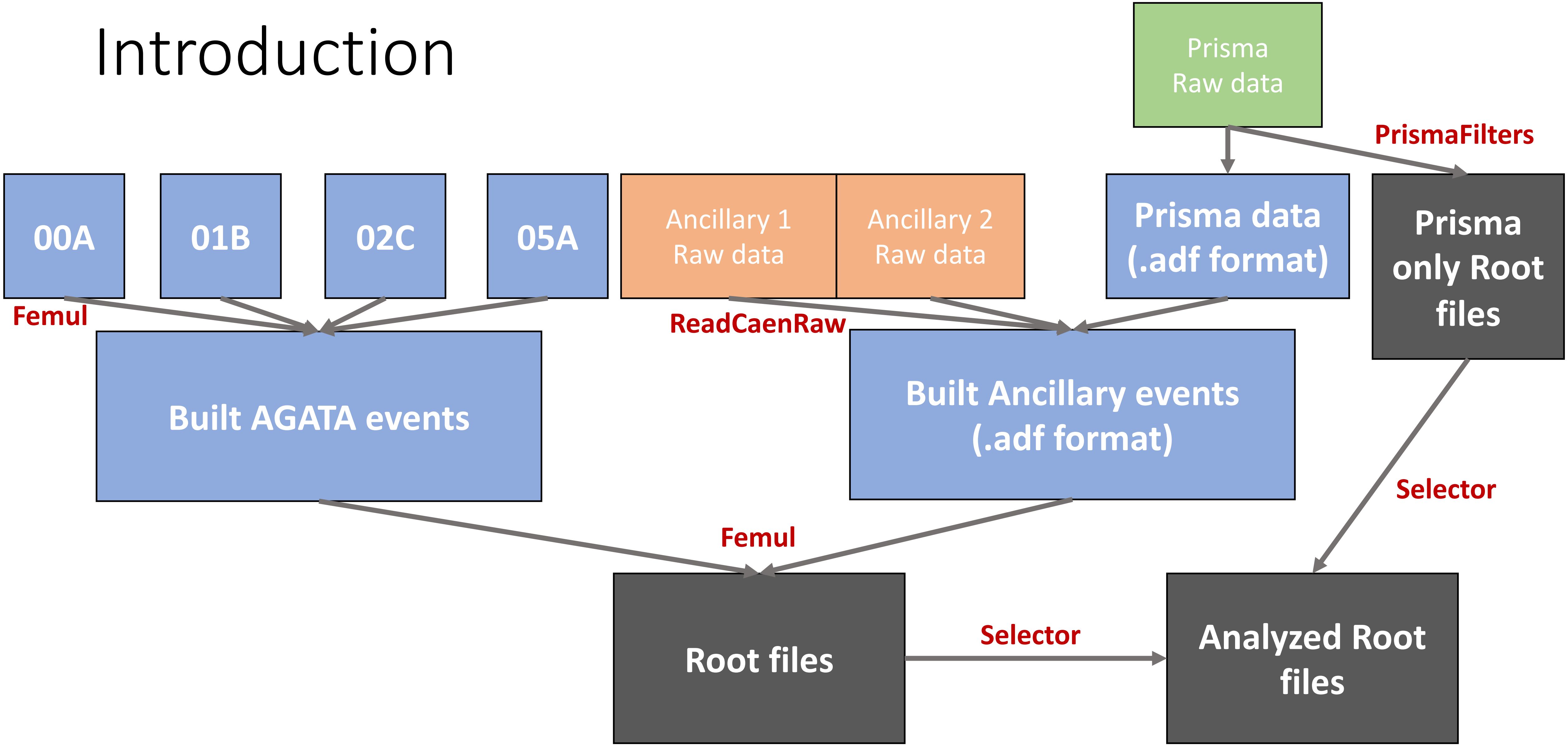
Jun 21, 2023

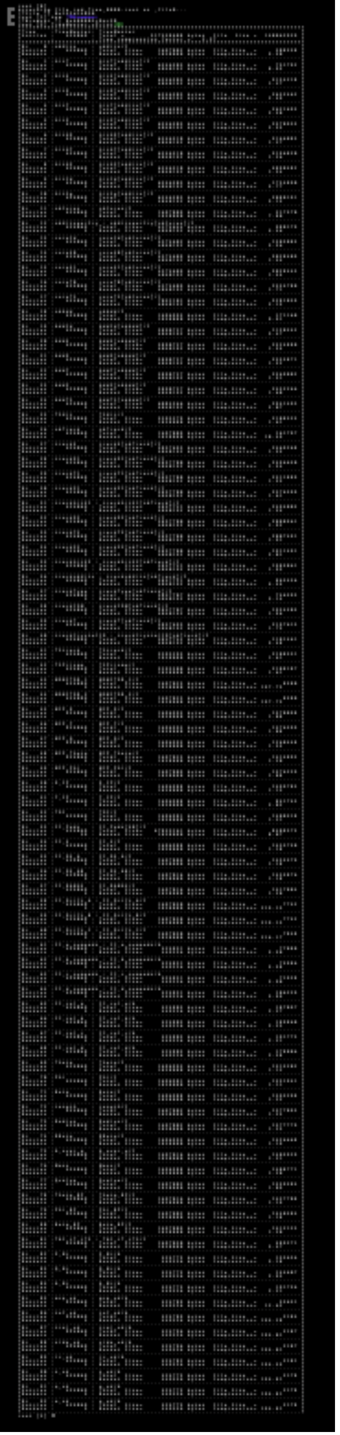
Merge branch 'main' of https://baltig.infn.it/gamma/agataselector **into main**
Agata Analysis authored 2 months ago ✓ eef32eca

Introduction

- Femul produces root files
- The selector represents the last step of the analysis procedure, where the coincidences between different detectors are analyzed
- It allows to generate histograms (and possibly also other root files) with high-level analyzed data
- Can perform optimization procedures to improve on its parameters
- Can place gates and select data based on cuts or intervals
- It can be made user (or experiment) specific to produce ad-hoc spectra
- **You are meant to modify it at your will as it represents a starting point for the analysis**

Introduction





The starting point

- Femul produces a root file containing MANY leaves
- The analysis procedure is common to all experiments and there is little benefit of repeating the same steps over and over
- The code was created for a quick near-line analysis and has since evolved with more refinements with full analysis capabilities
- This also means that sometimes some changes/improvements are made and could break backward compatibility (for instance for the configuration file). Since the program is more stable now it does not happen often
- We try to log on the CHANGELOG.md significant changes in different versions
- Also checkout the README.md file in the root directory.
- Other README.md files are located in the Conf folder, explaining the meaning of the files.

The output

- The selector produces ROOT files containing:
    - Histograms (each analysis is contained within a folder):
        - Single detector analysis
        - Coincidence analysis
    - TTrees of:
        - High level data of a single detector. For instance, Spider provides things such as excitation energy or angles
        - Doppler correction based on the analysis of the agata+ancillary coincidence
- Generally, TTrees take up a considerable amount of disk space and are not very useful since histograms provide the high-level analysis already

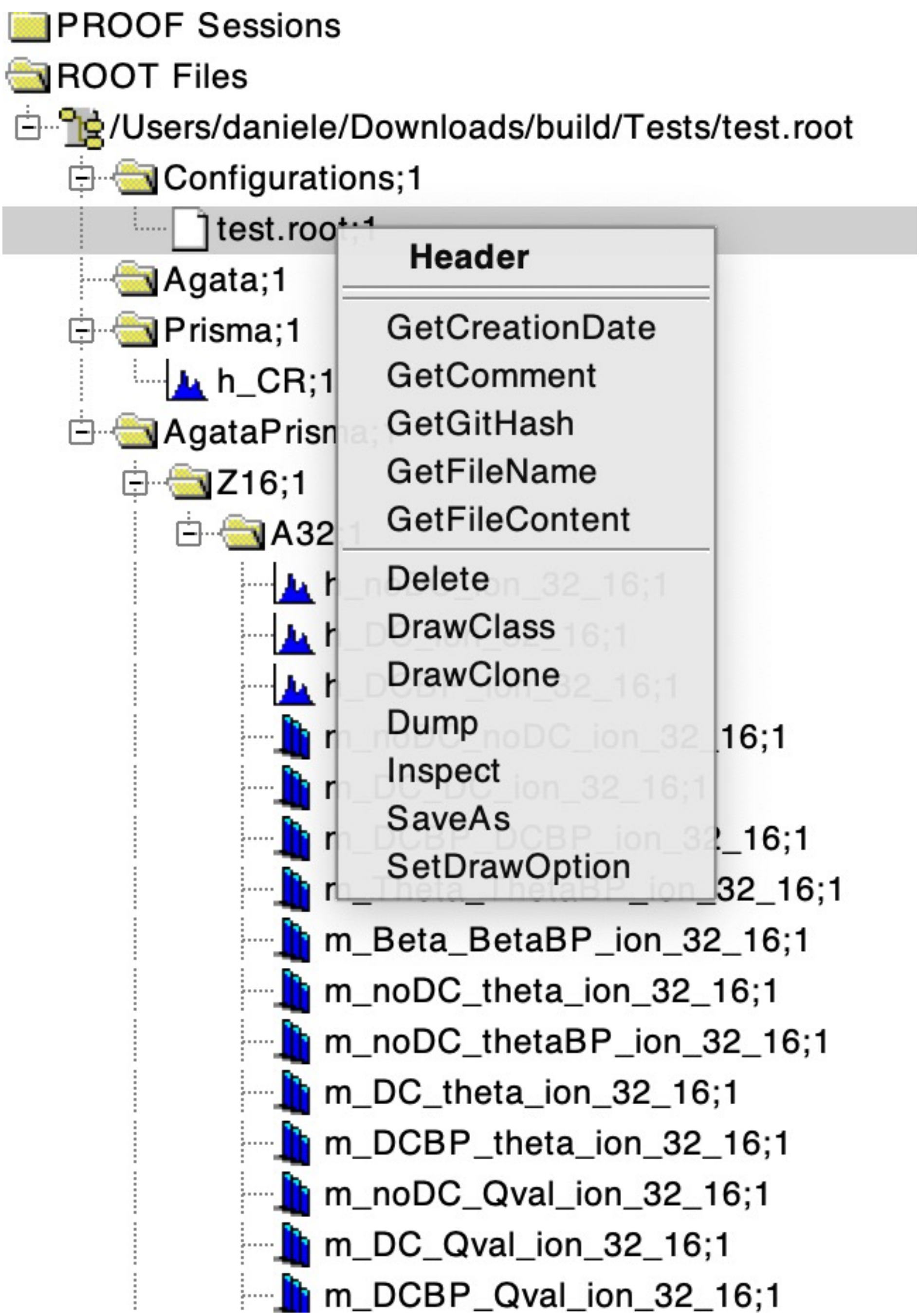
The output

- For each input file an output file is produced in the output folder. The files of each run are added in a single file called sum_xxx-yyy.root, where xxx is the run number and yyy is the number of input files.
- Different runs can be also added with - -sum_all
- The partial files can be discarded with the option - -rm_partial
- Additionally to the “regular” output it is possible to personalize the analysis with the UserSelector which will in turn produce additional output in the User folder.

Reproducibility

- The output files contain the parameters used to generate it:
  - The entire selector.conf
  - The git hash
  - The date of creation
- This means that the analysis can be reproduced simply by printing the selector.conf used for this specific file and checking out the correct hash
- It is also citable with a DOI:

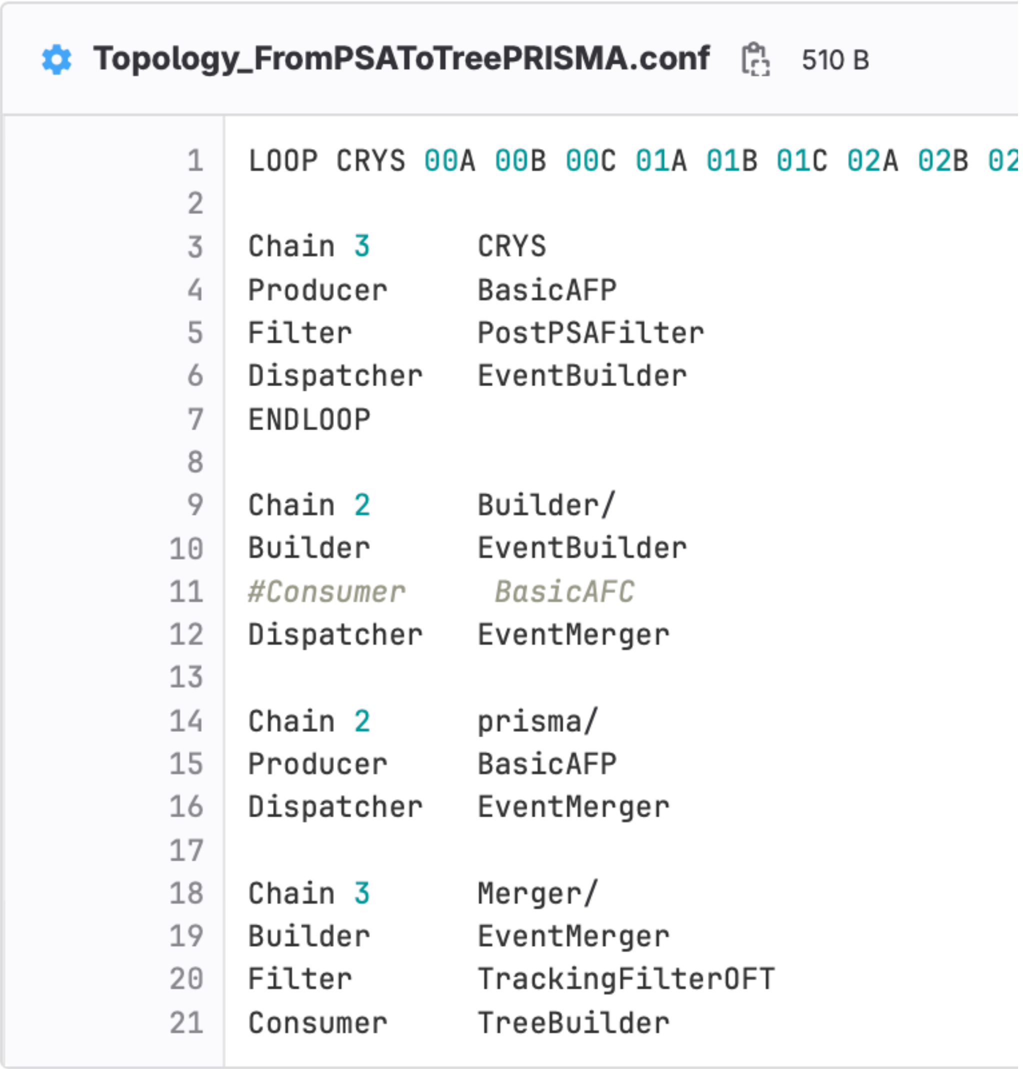
DOI 10.5281/zenodo.8329198



Running the replay

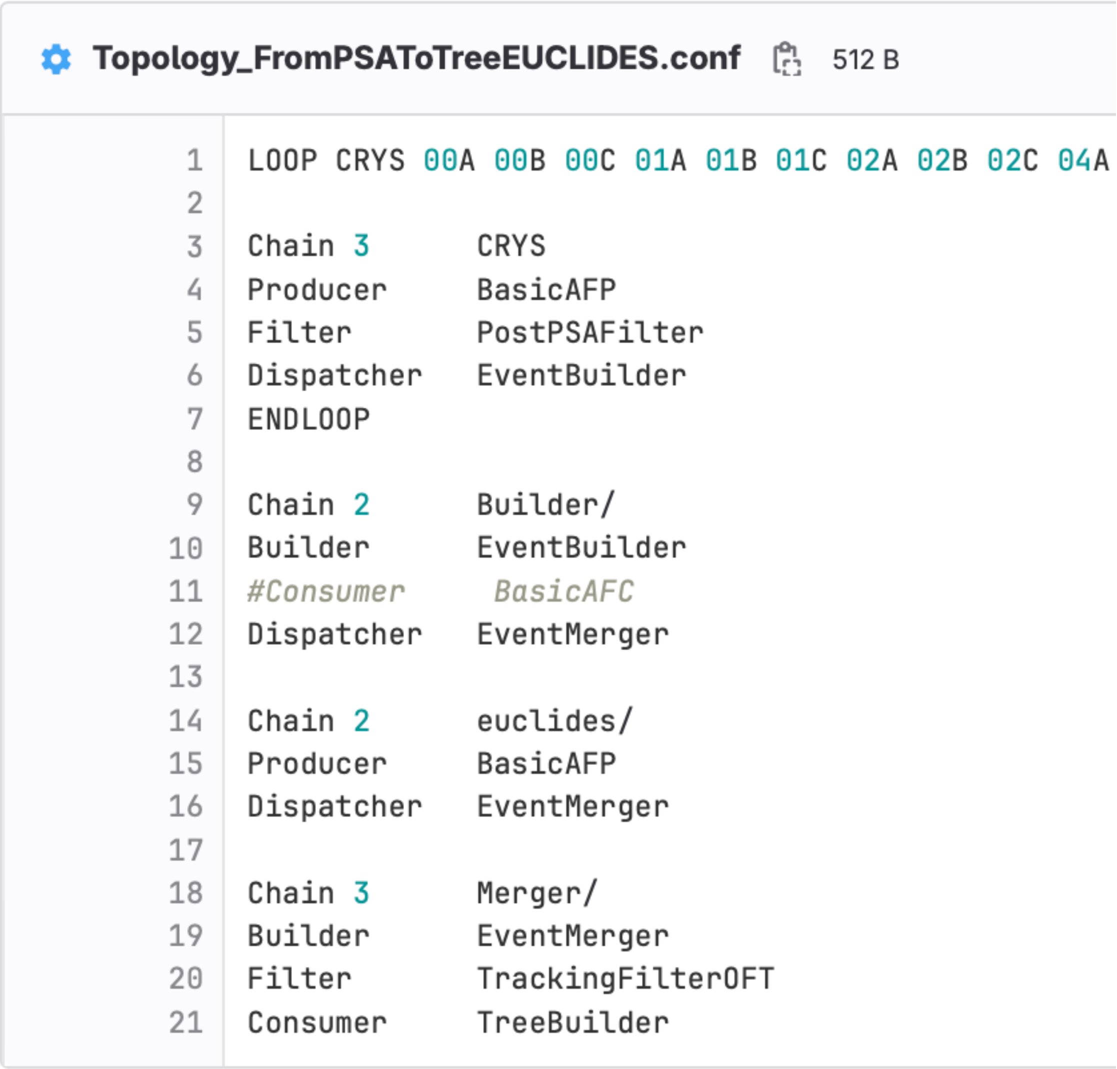
- Although the replay.py script is present in agataselector/Scripts/Replay/, it is mainly meant for the nearline users. It can, in principle be adapted but is not very useful
- In this case on the femul side of things there is not much to do except setting the **appropriate time windows to build events**, the rest is done in the selector
- As a consequence, once Agata is time aligned and calibrated the replay will be run once

The Topology



Topology_FromPSAToTreePRISMA.conf 510 B

```
LOOP CRYS 00A 00B 00C 01A 01B 01C 02A 02B 02

Chain 3      CRYS
Producer     BasicAFP
Filter       PostPSAFilter
Dispatcher   EventBuilder
ENDLOOP

Chain 2      Builder/
Builder      EventBuilder
#Consumer     BasicAFC
Dispatcher   EventMerger

Chain 2      prisma/
Producer     BasicAFP
Dispatcher   EventMerger

Chain 3      Merger/
Builder      EventMerger
Filter       TrackingFilterOFT
Consumer     TreeBuilder
```

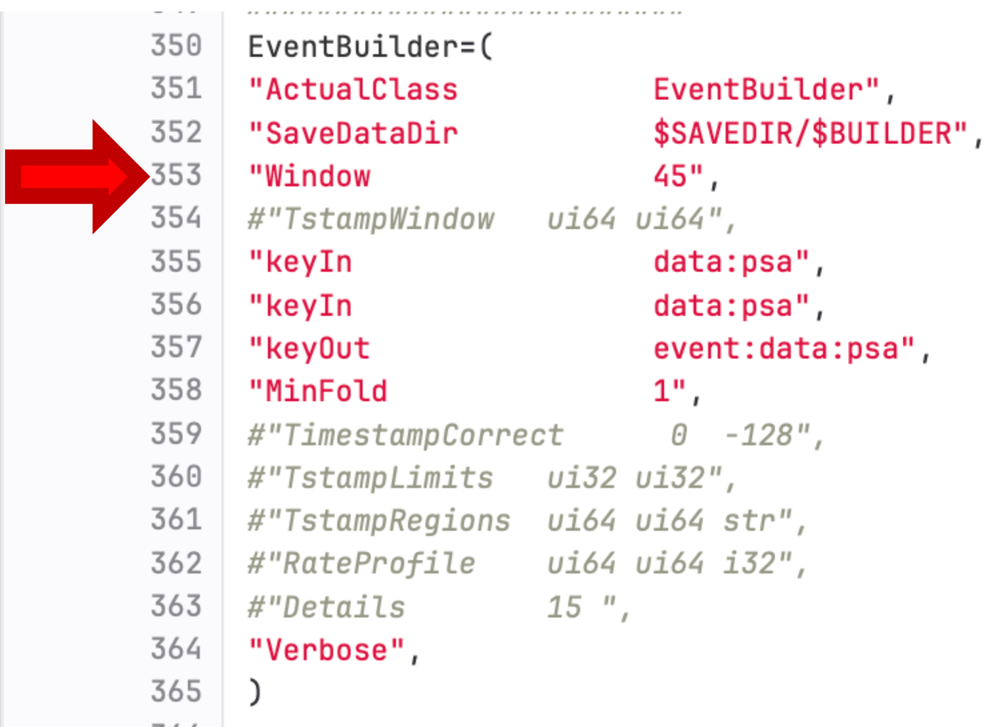


Topology_FromPSAToTreeEUCLIDES.conf 512 B

```
LOOP CRYS 00A 00B 00C 01A 01B 01C 02A 02B 02C 04A

Chain 3      CRYS
Producer     BasicAFP
Filter       PostPSAFilter
Dispatcher   EventBuilder
ENDLOOP

Chain 2      Builder/
Builder      EventBuilder
#Consumer     BasicAFC
Dispatcher   EventMerger

Chain 2      euclides/
Producer     BasicAFP
Dispatcher   EventMerger

Chain 3      Merger/
Builder      EventMerger
Filter       TrackingFilterOFT
Consumer     TreeBuilder
```

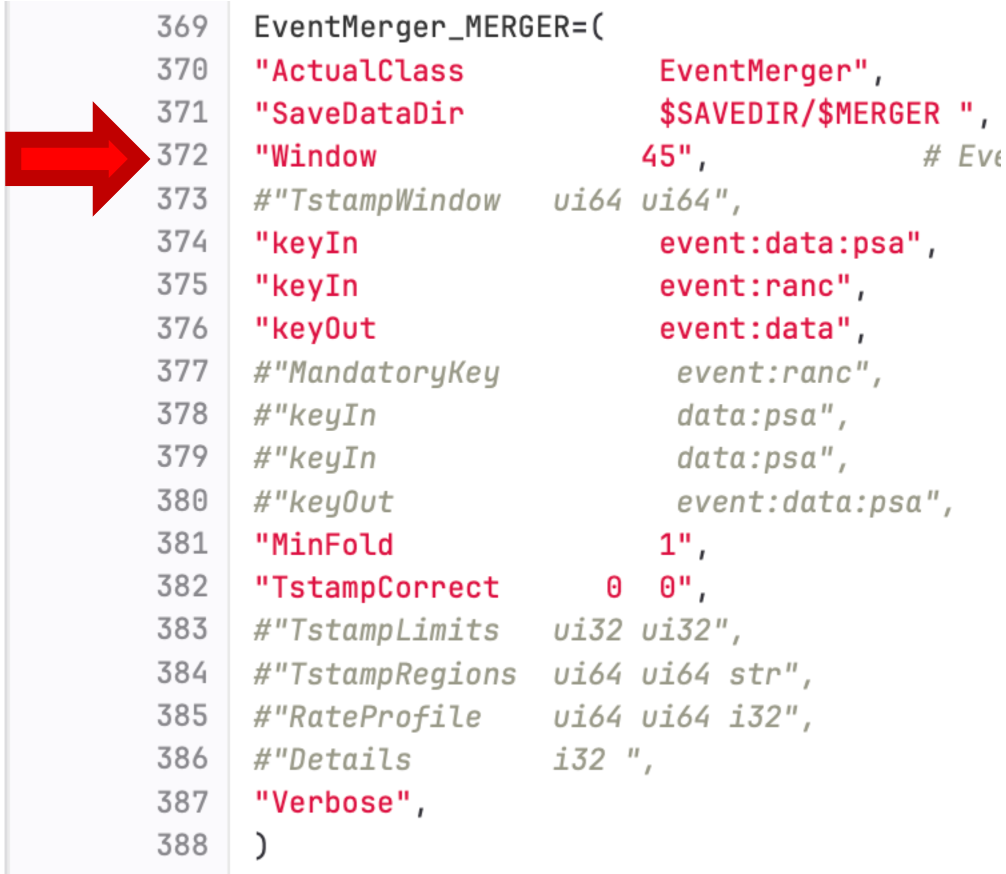
The gen_conf.py

- The event builder builds agata events



```
EventBuilder=(
"ActualClass           EventBuilder",
"SaveDataDir           $SAVEDIR/$BUILDER",
"Window                45",
#"TstampWindow    ui64 ui64",
"keyIn                 data:psa",
"keyIn                 data:psa",
"keyOut                event:data:psa",
"MinFold               1",
#"TimestampCorrect      0   -128",
#"TstampLimits    ui32 ui32",
#"TstampRegions   ui64 ui64 str",
#"RateProfile     ui64 ui64 i32",
#"Details         15 ",
"Verbose",
)
```

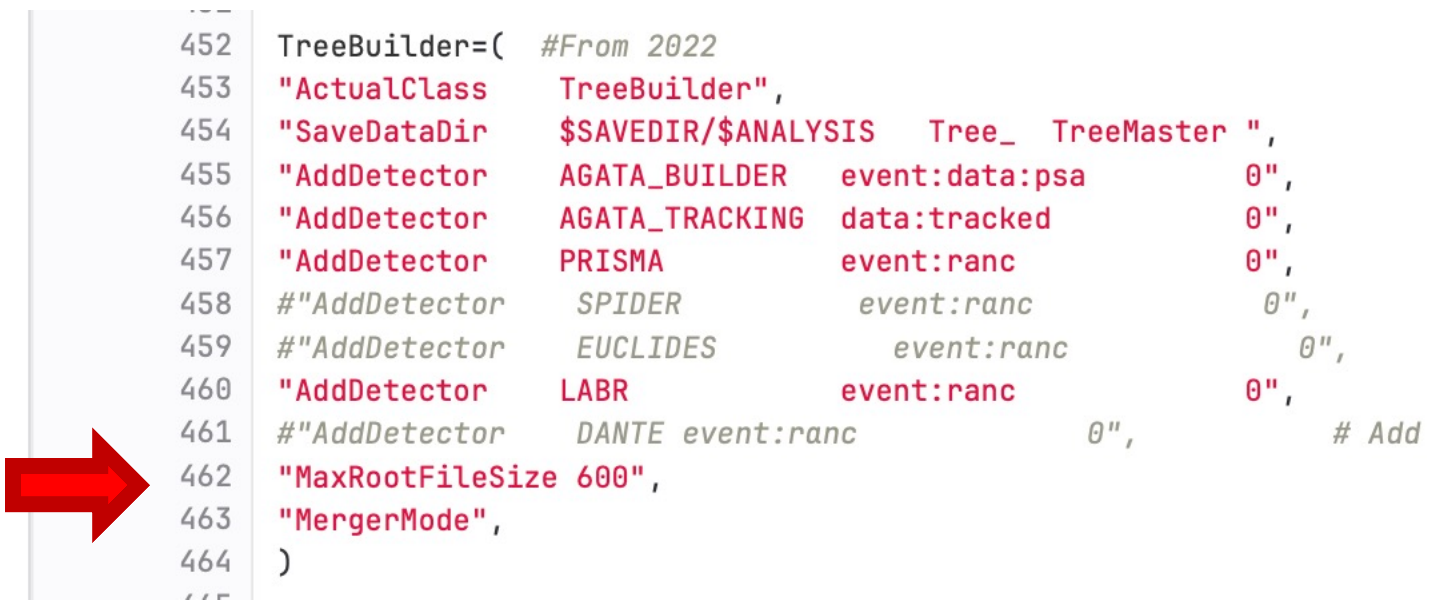
- The event merger builds agata+ancillary events. The ancillary events need to be already built



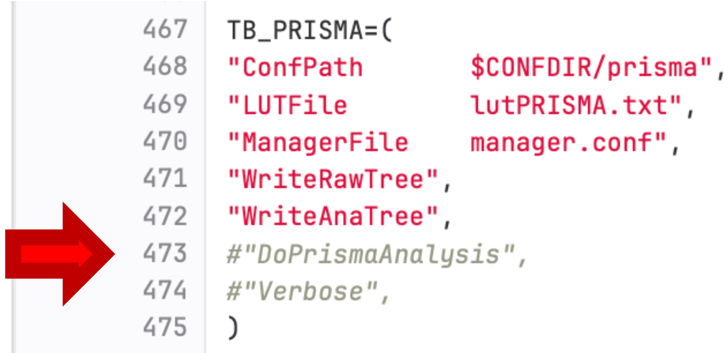
```
EventMerger_MERGER=(
"ActualClass           EventMerger",
"SaveDataDir           $SAVEDIR/$MERGER ",
"Window               45",            # Eve
#"TstampWindow    ui64 ui64",
"keyIn                 event:data:psa",
"keyIn                 event:ranc",
"keyOut                event:data",
#"MandatoryKey          event:ranc",
#"keyIn                 data:psa",
#"keyIn                 data:psa",
#"keyOut                event:data:psa",
"MinFold               1",
"TstampCorrect      0  0",
#"TstampLimits    ui32 ui32",
#"TstampRegions   ui64 ui64 str",
#"RateProfile     ui64 ui64 i32",
#"Details         i32 ",
"Verbose",
)
```

The gen_conf.py

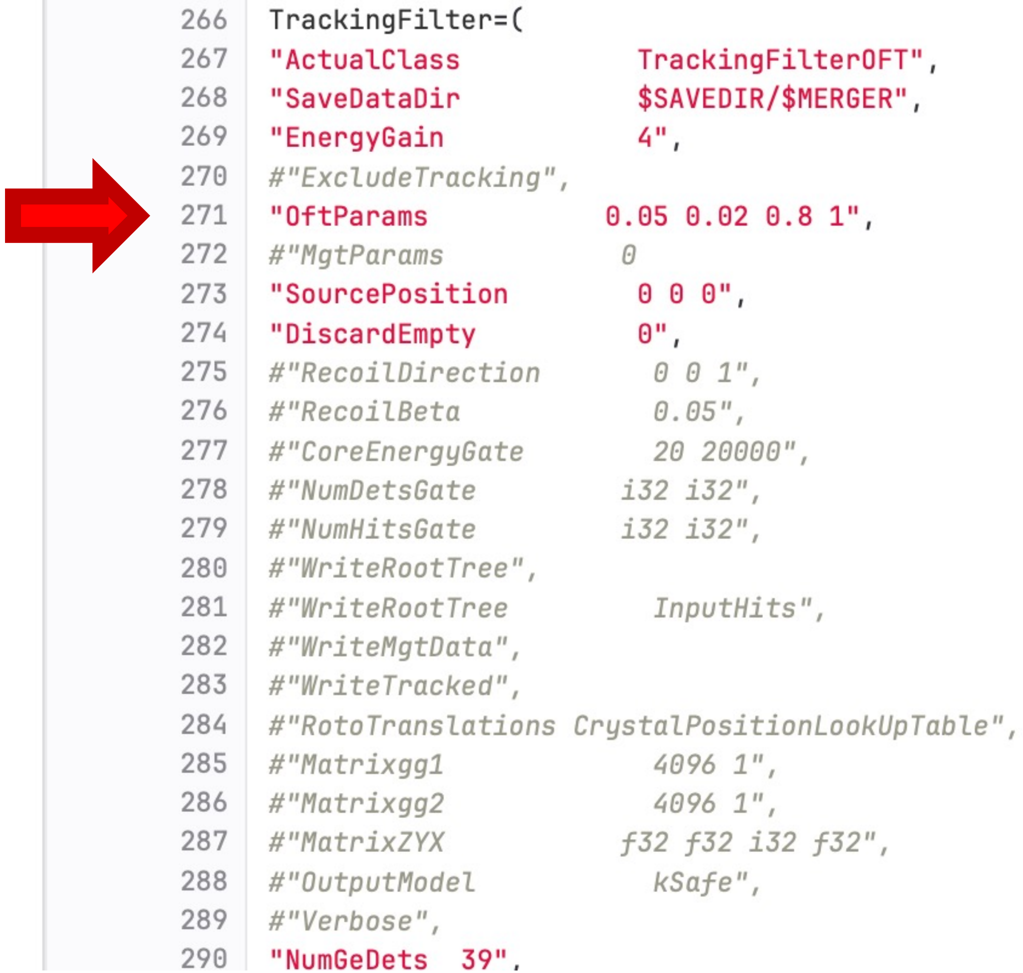
- The tracking and TreeBuilder are the last steps



```
TreeBuilder=(  #From 2022
"ActualClass     TreeBuilder",
"SaveDataDir     $SAVEDIR/$ANALYSIS   Tree_  TreeMaster ",
"AddDetector     AGATA_BUILDER    event:data:psa          0",
"AddDetector     AGATA_TRACKING   data:tracked            0",
"AddDetector     PRISMA           event:ranc              0",
#"AddDetector     SPIDER           event:ranc              0",
#"AddDetector     EUCLIDES           event:ranc              0",
"AddDetector     LABR             event:ranc              0",
#"AddDetector     DANTE event:ranc             0",          # Add
"MaxRootFileSize 600",
"MergerMode",
)
```



```
TB_PRISMA=(
"ConfPath        $CONFDIR/prisma",
"LUTFile         lutPRISMA.txt",
"ManagerFile     manager.conf",
"WriteRawTree",
"WriteAnaTree",
#"DoPrismaAnalysis",
#"Verbose",
)
```



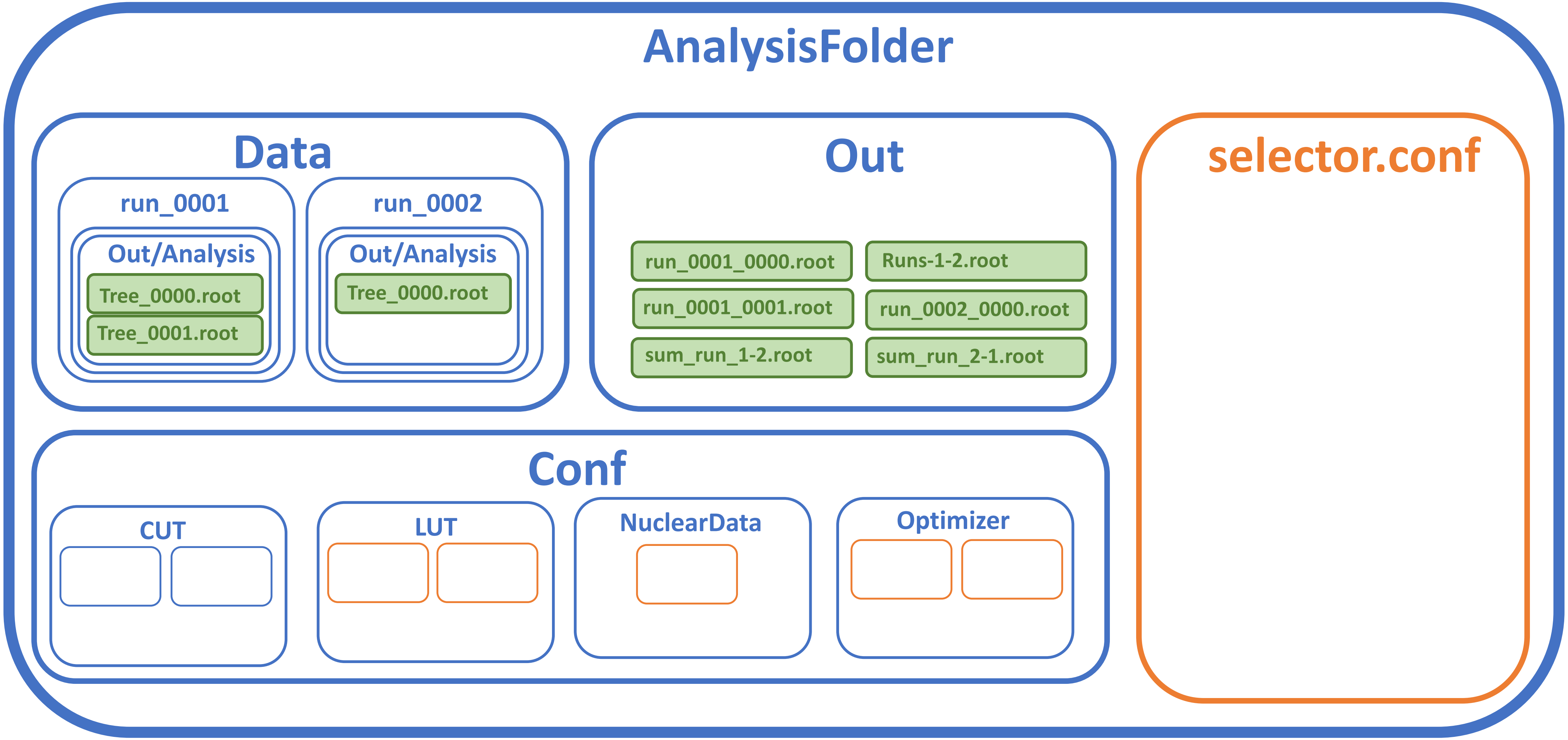
```
TrackingFilter=(
"ActualClass             TrackingFilterOFT",
"SaveDataDir             $SAVEDIR/$MERGER",
"EnergyGain              4",
#"ExcludeTracking",
"OftParams             0.05 0.02 0.8 1",
#"MgtParams              0
"SourcePosition          0 0 0",
"DiscardEmpty            0",
#"RecoilDirection         0 0 1",
#"RecoilBeta              0.05",
#"CoreEnergyGate          20 20000",
#"NumDetsGate            i32 i32",
#"NumHitsGate            i32 i32",
#"WriteRootTree",
#"WriteRootTree           InputHits",
#"WriteMgtData",
#"WriteTracked",
#"RotoTranslations CrystalPositionLookUpTable",
#"Matrixgg1               4096 1",
#"Matrixgg2               4096 1",
#"MatrixZYX             f32 f32 i32 f32",
#"OutputModel             kSafe",
#"Verbose",
"NumGeDets   39",
```

Common issues

- Prisma needs a configuration folder containing gates and calibrations generated for the Prisma analysis. Check that the gates are loaded at the start of femul (they should flash in green and yellow).
- The folder, according to the nearline gen_conf.py should be in the Conf/Prisma folder. After running the gen_conf.py, the manager.conf file should point to the correct path of the conf of the specific replay
- If the prisma branches are empty check the DoPrismaAnalysis parameter in the gen_conf.py
- In some cases the coincidence peak will not be present, we will handle the situation later on

Setup of the analysis

Setup of the analysis folder

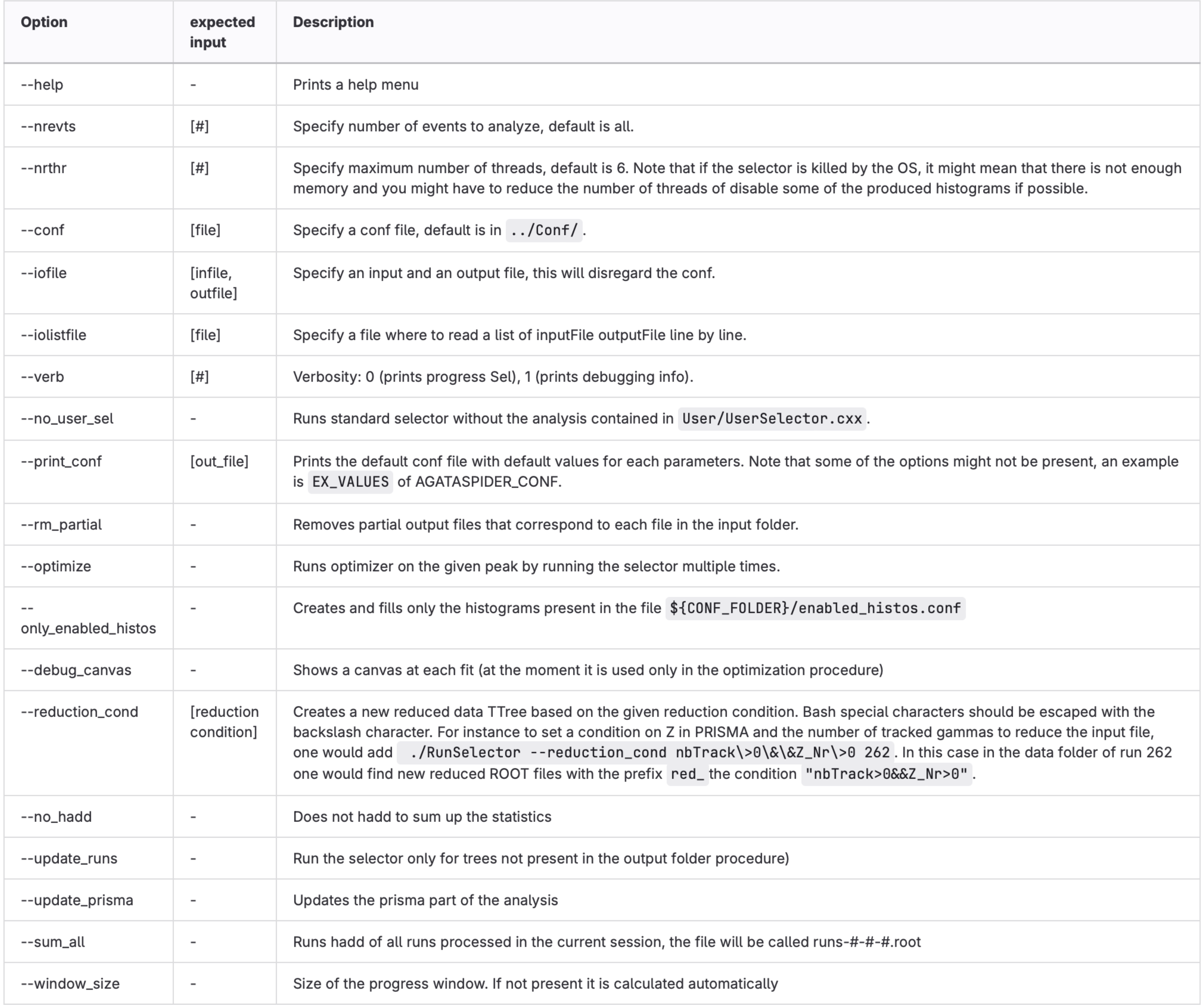


Generate/retrieve the default selector.conf

- In all experiments, a nearline analysis was performed, it is a good starting point for the offline analysis.
- To “recall” the analysis you should have compiled the selector with che cmake option “-DEXP_NAME=EXP_###”. This will compile the UserSelector that you can find in agataselector/User/EXP/EXP_###
- The configuration folder adapted for your experiment is agataselector/User/EXP/EXP_###/Conf
- The selector.conf is in agataselector/User/EXP/EXP_###/selector.conf (you might have multiple versions if the setup was changed during the experiment)

In general, the selector is being constantly updated with bug fixes and improvements. This could mean that some features could have been added and some changes need to be applied to the selector.conf if you have pulled from origin. In order to get access to all options you can print the default configuration file with the option RunSelector - -print_conf my_selector.conf. Each conf parameter will contain a comment with some infos on the effect of the parameter

Options



Option	expected input	Description
--help	-	Prints a help menu
--nrevts	[#]	Specify number of events to analyze, default is all.
--nrthr	[#]	Specify maximum number of threads, default is 6. Note that if the selector is killed by the OS, it might mean that there is not enough memory and you might have to reduce the number of threads of disable some of the produced histograms if possible.
--conf	[file]	Specify a conf file, default is in `../Conf/`.
--iofile	[infile, outfile]	Specify an input and an output file, this will disregard the conf.
--iolistfile	[file]	Specify a file where to read a list of inputFile outputFile line by line.
--verb	[#]	Verbosity: 0 (prints progress Sel), 1 (prints debugging info).
--no_user_sel	-	Runs standard selector without the analysis contained in `User/UserSelector.cxx`.
--print_conf	[out_file]	Prints the default conf file with default values for each parameters. Note that some of the options might not be present, an example is `EX_VALUES` of AGATASPIDER_CONF.
--rm_partial	-	Removes partial output files that correspond to each file in the input folder.
--optimize	-	Runs optimizer on the given peak by running the selector multiple times.
--only_enabled_histos	-	Creates and fills only the histograms present in the file `${CONF_FOLDER}/enabled_histos.conf`
--debug_canvas	-	Shows a canvas at each fit (at the moment it is used only in the optimization procedure)
--reduction_cond	[reduction condition]	Creates a new reduced data TTree based on the given reduction condition. Bash special characters should be escaped with the backslash character. For instance to set a condition on Z in PRISMA and the number of tracked gammas to reduce the input file, one would add `./RunSelector --reduction_cond nbTrack\>0\&\&Z_Nr\>0 262`. In this case in the data folder of run 262 one would find new reduced ROOT files with the prefix `red_` the condition `"nbTrack>0&&Z_Nr>0"`.
--no_hadd	-	Does not hadd to sum up the statistics
--update_runs	-	Run the selector only for trees not present in the output folder procedure)
--update_prisma	-	Updates the prisma part of the analysis
--sum_all	-	Runs hadd of all runs processed in the current session, the file will be called runs-#-#-#.root
--window_size	-	Size of the progress window. If not present it is calculated automatically

Agata leaves

- In general, the leaves contain the following information for:
  - Single hit (within a segment)
  - Single core
  - Addback (nearest cores)
  - Tracking

Leaf name	Data type	Content
nb	int	Number of gammas/interactions
id	int[nb]	Id of the core/segment
Energy	float[nb]	Energy of the gamma/hit
TS	unsigned long/unsigned long[nb]	Lowest timestamp/array of all triggered channels
(G)X/(G)Y/(G)Z	float[nb]	Position of the hit/first interaction
T	float[nb]	Cfd time, needs to be added to TS

Agata leaves

- Some leaves are more specific

Leaf name	Data type	Content
trackX2/trackY2/trackZ2	float[nb]	Position of the second interaction of the gamma (for polarization analysis puroposes)
hitX/hitY/hitZ	float[nb]	Position of a hit in the crystal frame of reference
trackFOM	float[nb]	Figure of merit of tracking
trackType	float[nb]	Compton/photoelectric/pair production

Other ancillaries (aka CAEN digitizers)

- Caen digitizers provide a common input data as a consequence each of these detectors (Euclides, Spider, Dante, Labr, …) require a lookup table (LUT) that assigns to board+channel a given signal that is used in the analysis.

Leaf name	Data type	Content
nb	int	Number of channels in an event
Channel	int[nb]	Channel that has triggered
Board	int[nb]	Board that has triggered
TS	unsigned long	Lowest timestamp of all triggered channels
TSHit	unsigned long[nb]	Timestamp of the single hit
Time	float[nb]	Interpolated time, needs to be added to TSHit
Energy	float[nb]	Energy of trapezoid (PHA) of Qlong (PSD)
QShort	float[nb]	Short integration (PSD only)

General remarks

- Although the naming scheme for histograms can be messy there are some rules:
  - h_??? Means TH1D, g_?? Means TGraph, m_??? Means TH2D and TH3D
  - **DC means Doppler corrected for whatever is detected directly by the ancillary**
  - DCBP means Doppler corrected for the undetected binary partner of the reaction
  - Similarly ThetaBP would mean the reconstructed angle of the binary partner
- Otherwise, one has to read the histogram title

Structure of the selector.conf

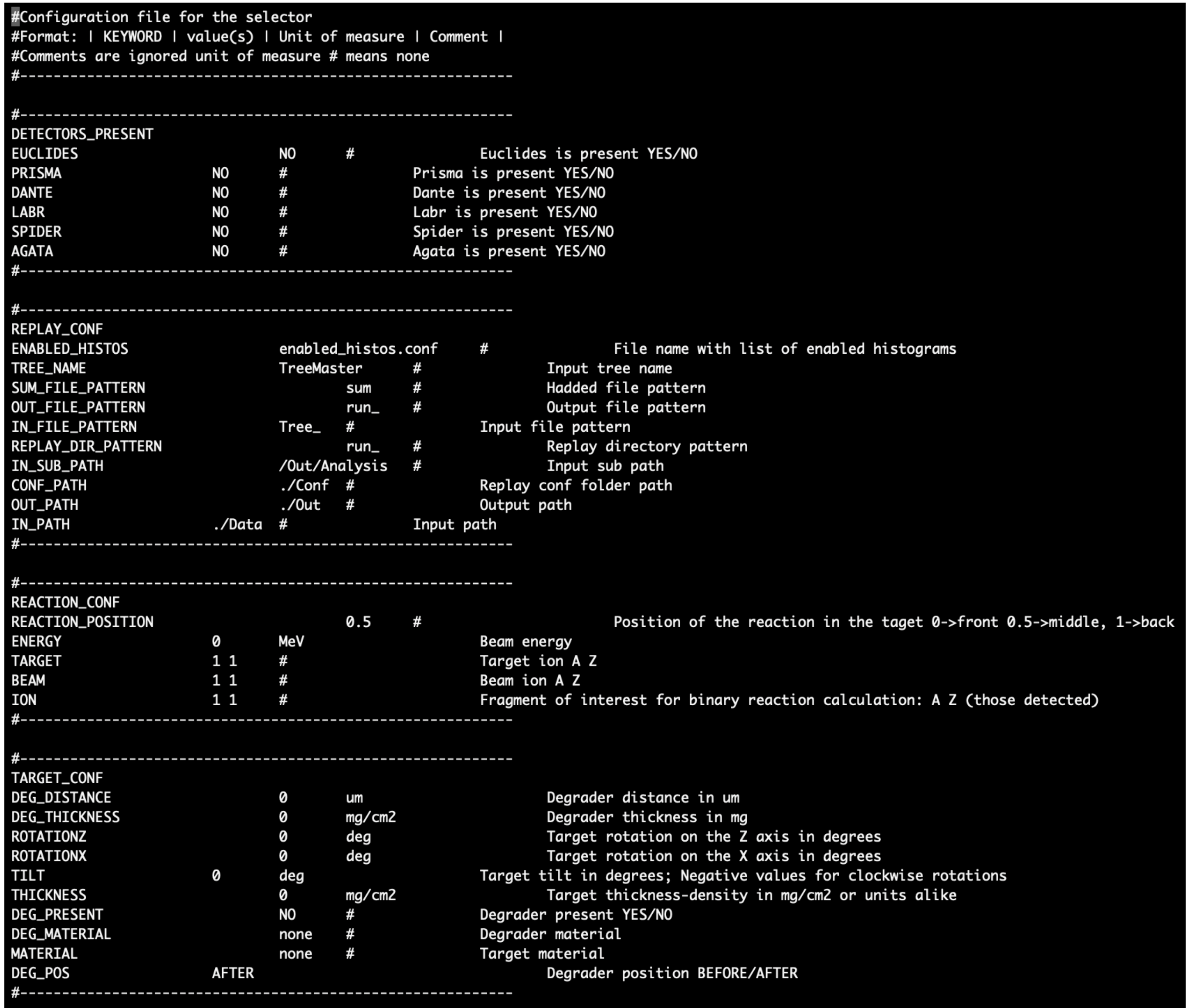
KEYWORD | value(s) | unit of measure | comment

Detectors considered in the analysis

Configuration of the folders, the file patterns, and the TTree names

Configuration of the reaction, multiple ions of interest can be added

Target thickness and rotations, used for energy loss calculaitons. The presence of a degrader before or after the target is also possible.

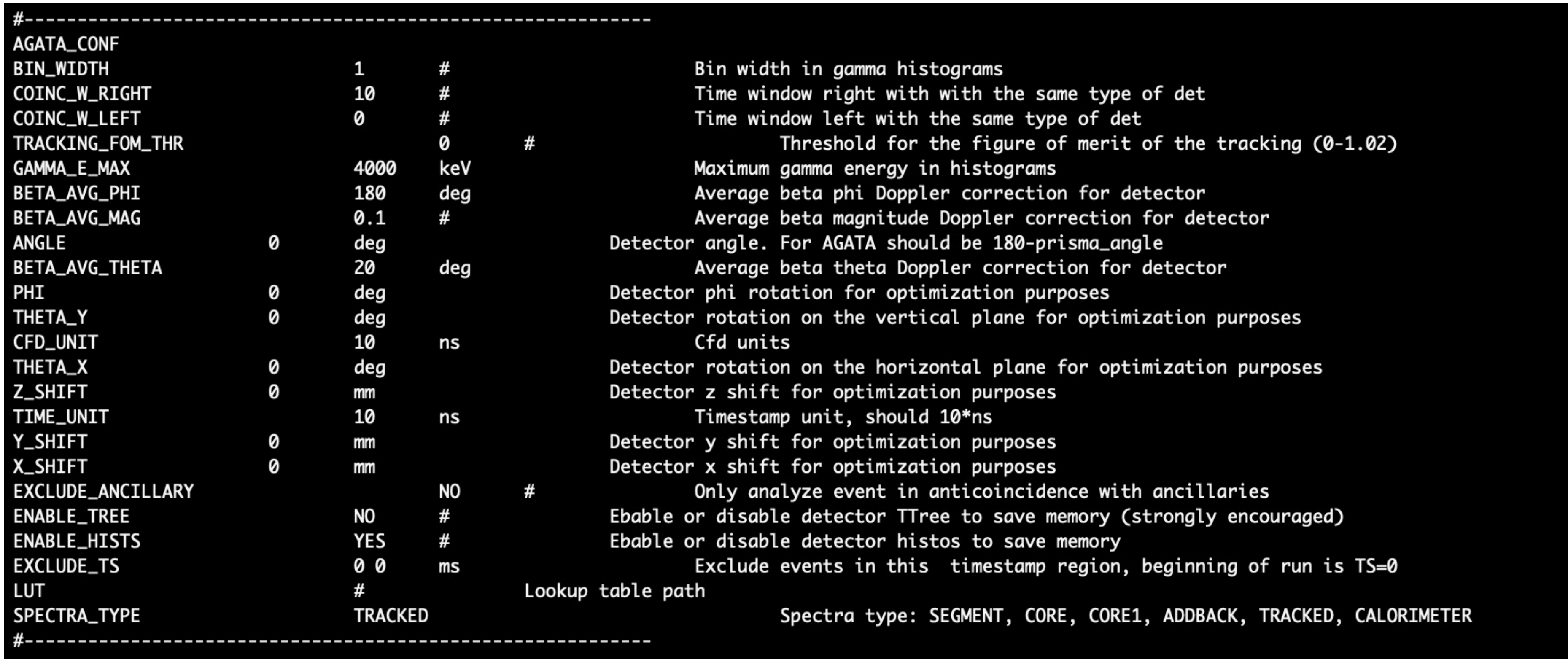


```
#Configuration file for the selector
#Format: | KEYWORD | value(s) | Unit of measure | Comment |
#Comments are ignored unit of measure # means none
#-----------------------------------------------------------

#-----------------------------------------------------------
DETECTORS_PRESENT
EUCLIDES                        NO      #               Euclides is present YES/NO
PRISMA                  NO      #               Prisma is present YES/NO
DANTE                   NO      #               Dante is present YES/NO
LABR                    NO      #               Labr is present YES/NO
SPIDER                  NO      #               Spider is present YES/NO
AGATA                   NO      #               Agata is present YES/NO
#-----------------------------------------------------------

#-----------------------------------------------------------
REPLAY_CONF
ENABLED_HISTOS                  enabled_histos.conf     #               File name with list of enabled histograms
TREE_NAME                       TreeMaster      #               Input tree name
SUM_FILE_PATTERN                        sum     #               Hadded file pattern
OUT_FILE_PATTERN                        run_    #               Output file pattern
IN_FILE_PATTERN                 Tree_   #               Input file pattern
REPLAY_DIR_PATTERN                      run_    #               Replay directory pattern
IN_SUB_PATH                     /Out/Analysis   #               Input sub path
CONF_PATH                       ./Conf  #               Replay conf folder path
OUT_PATH                        ./Out   #               Output path
IN_PATH                 ./Data  #               Input path
#-----------------------------------------------------------

#-----------------------------------------------------------
REACTION_CONF
REACTION_POSITION                       0.5     #                       Position of the reaction in the taget 0->front 0.5->middle, 1->back
ENERGY                  0       MeV                     Beam energy
TARGET                  1 1     #                       Target ion A Z
BEAM                    1 1     #                       Beam ion A Z
ION                     1 1     #                       Fragment of interest for binary reaction calculation: A Z (those detected)
#-----------------------------------------------------------

#-----------------------------------------------------------
TARGET_CONF
DEG_DISTANCE                    0       um                      Degrader distance in um
DEG_THICKNESS                   0       mg/cm2                  Degrader thickness in mg
ROTATIONZ                       0       deg                     Target rotation on the Z axis in degrees
ROTATIONX                       0       deg                     Target rotation on the X axis in degrees
TILT                    0       deg                     Target tilt in degrees; Negative values for clockwise rotations
THICKNESS                       0       mg/cm2                  Target thickness-density in mg/cm2 or units alike
DEG_PRESENT                     NO      #               Degrader present YES/NO
DEG_MATERIAL                    none    #               Degrader material
MATERIAL                        none    #               Target material
DEG_POS                 AFTER                                   Degrader position BEFORE/AFTER
#-----------------------------------------------------------
```

Example of Detector Conf: Agata

KEYWORD | value(s) | unit of measure | comment

- Enable/disable histograms or TTrees
- Speficy LUT (if necessary)
- Set some global angles and psitions
- Sets parameters of the histograms such as bin width
- Sets other detector-dependent parameters

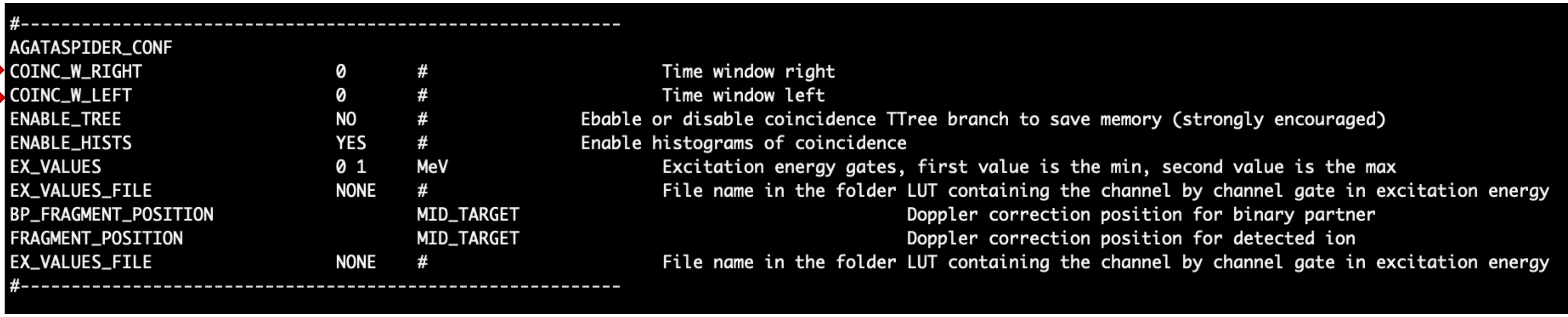


```
#-----------------------------------------------------------
AGATA_CONF
BIN_WIDTH                       1       #                       Bin width in gamma histograms
COINC_W_RIGHT                   10      #                       Time window right with with the same type of det
COINC_W_LEFT                    0       #                       Time window left with the same type of det
TRACKING_FOM_THR                        0       #                       Threshold for the figure of merit of the tracking (0-1.02)
GAMMA_E_MAX                     4000    keV                     Maximum gamma energy in histograms
BETA_AVG_PHI                    180     deg                     Average beta phi Doppler correction for detector
BETA_AVG_MAG                    0.1     #                       Average beta magnitude Doppler correction for detector
ANGLE                   0       deg                     Detector angle. For AGATA should be 180-prisma_angle
BETA_AVG_THETA                  20      deg                     Average beta theta Doppler correction for detector
PHI                     0       deg                     Detector phi rotation for optimization purposes
THETA_Y                 0       deg                     Detector rotation on the vertical plane for optimization purposes
CFD_UNIT                        10      ns                      Cfd units
THETA_X                 0       deg                     Detector rotation on the horizontal plane for optimization purposes
Z_SHIFT                 0       mm                      Detector z shift for optimization purposes
TIME_UNIT                       10      ns                      Timestamp unit, should 10*ns
Y_SHIFT                 0       mm                      Detector y shift for optimization purposes
X_SHIFT                 0       mm                      Detector x shift for optimization purposes
EXCLUDE_ANCILLARY                       NO      #                       Only analyze event in anticoincidence with ancillaries
ENABLE_TREE                     NO      #               Ebable or disable detector TTree to save memory (strongly encouraged)
ENABLE_HISTS                    YES     #               Ebable or disable detector histos to save memory
EXCLUDE_TS                      0 0     ms                      Exclude events in this  timestamp region, beginning of run is TS=0
LUT                             #               Lookup table path
SPECTRA_TYPE                    TRACKED                                 Spectra type: SEGMENT, CORE, CORE1, ADDBACK, TRACKED, CALORIMETER
#-----------------------------------------------------------
```

Example of Coincidence Conf: Agata+Spider

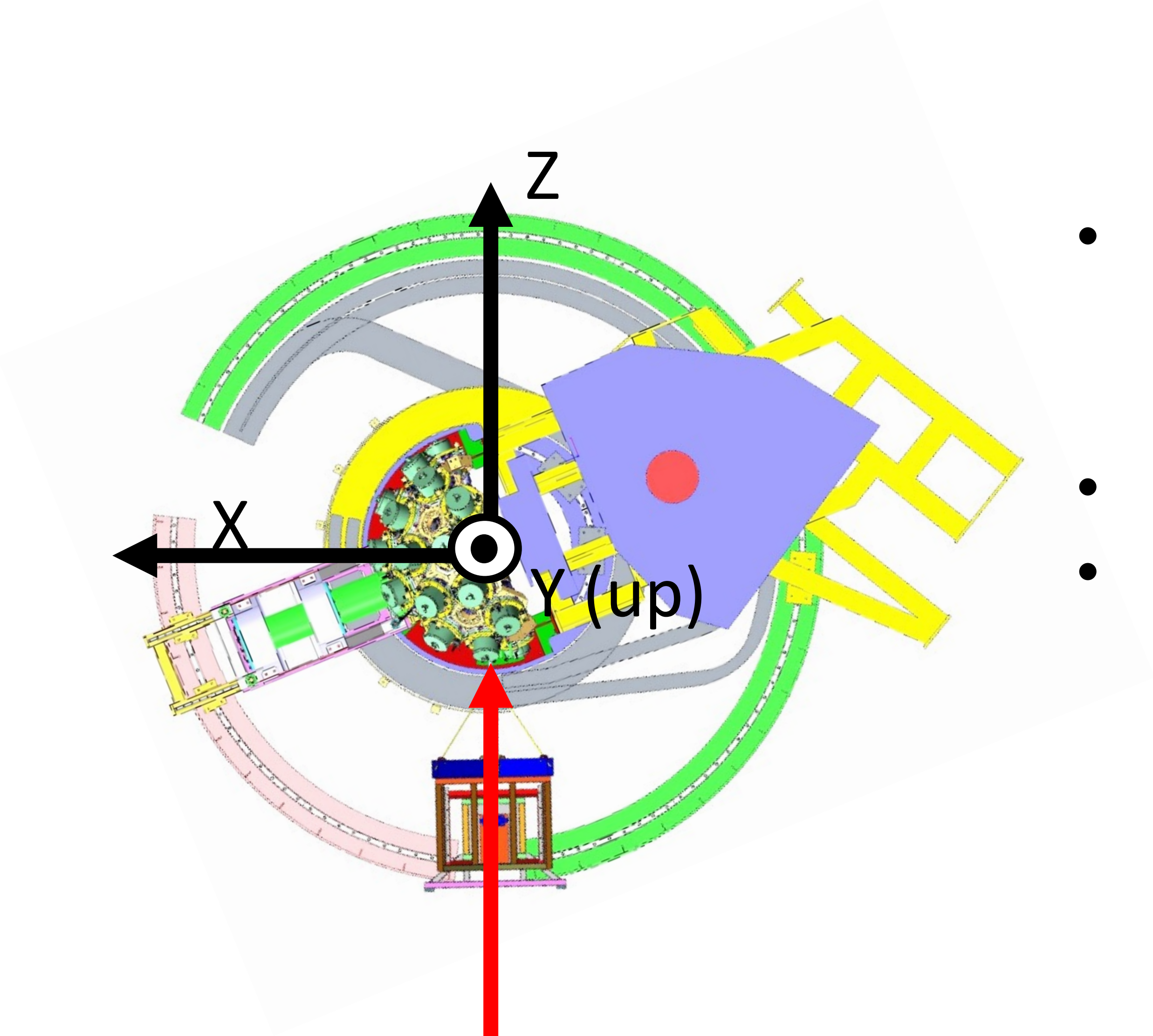
KEYWORD | value(s) | unit of measure | comment

- Set the coincidence window based on the peak position
- Enable histograms and trees
- Set Doppler correction position
- Other detector-dependent parameters



```
#------------------------------------------------------------
AGATASPIDER_CONF
COINC_W_RIGHT                   0       #                        Time window right
COINC_W_LEFT                    0       #                        Time window left
ENABLE_TREE                     NO      #                Ebable or disable coincidence TTree branch to save memory (strongly encouraged)
ENABLE_HISTS                    YES     #                Enable histograms of coincidence
EX_VALUES                       0 1     MeV                      Excitation energy gates, first value is the min, second value is the max
EX_VALUES_FILE                  NONE    #                        File name in the folder LUT containing the channel by channel gate in excitation energy
BP_FRAGMENT_POSITION                    MID_TARGET                                       Doppler correction position for binary partner
FRAGMENT_POSITION                       MID_TARGET                                       Doppler correction position for detected ion
EX_VALUES_FILE                  NONE    #                        File name in the folder LUT containing the channel by channel gate in excitation energy
#------------------------------------------------------------
```

Frame of reference



- The frame of reference with respect to the beam is necessary for reaction calculations
- It is common to all detectors.
- Z points in the direction of the beam and Y points upwards

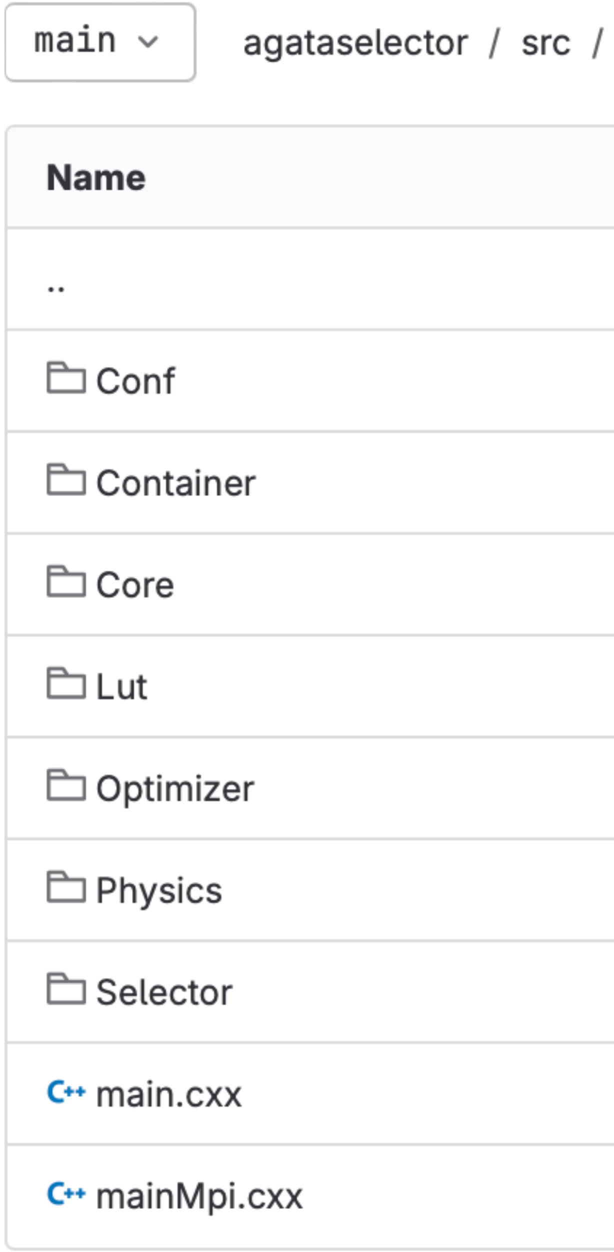
Energy loss corrections

- The energy loss corrections for the target are calculated for the kinematics reconstruction and the doppler correction. The kinematics reconstruction is done in at the reaction point specified in the REACTION_CONF ([0-1]).
- They are calculated **only if the eloss table is present** under Conf/EnergyLoss with the naming scheme given by the keyword MATERIAL/DEGRADER_MATERIAL under TARGET_CONF (check with - -verb 2).
- They need to be calculated with SRIM, and can be generated with the Script under Scripts/Srim

How to add histograms and personalize the analysis

The source code

- Conf: parameters that can be read from the selector.conf
- Container: classes that represent the analyzed detector
- Core: things unrelated to the analysis
- Lut: classes that read the lookup tables of various detectors
- Optimizer: fitter class and minimizer for optimization
- Physics: nuclear data class, energy loss and reaction calculator
- Selector: analysis code

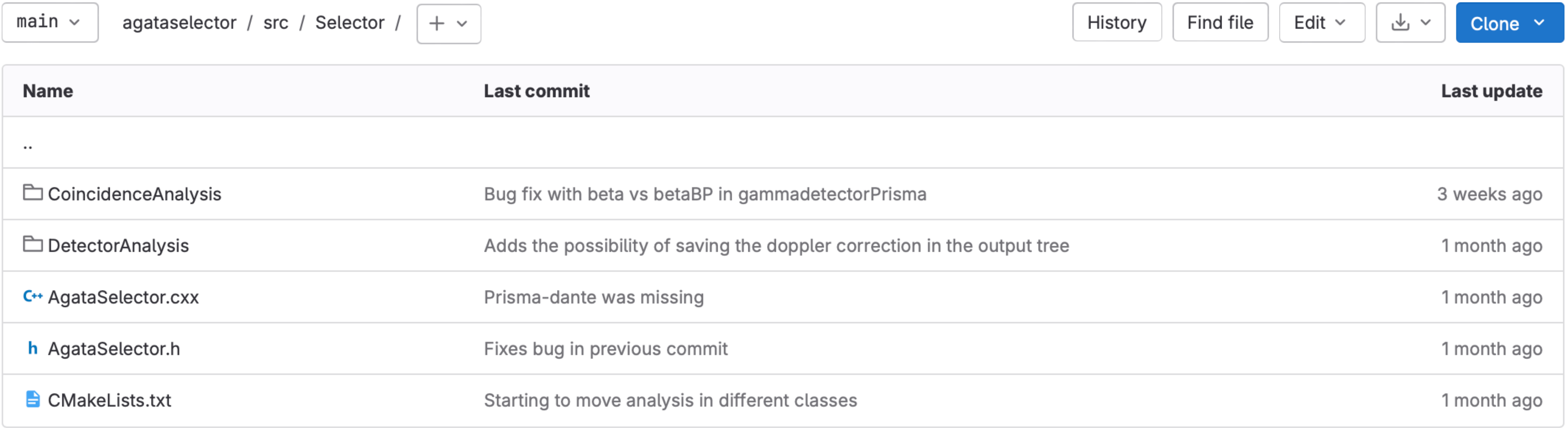


main ⌄ agataselector / src /

Name
..
Conf
Container
Core
Lut
Optimizer
Physics
Selector
main.cxx
mainMpi.cxx

The source code

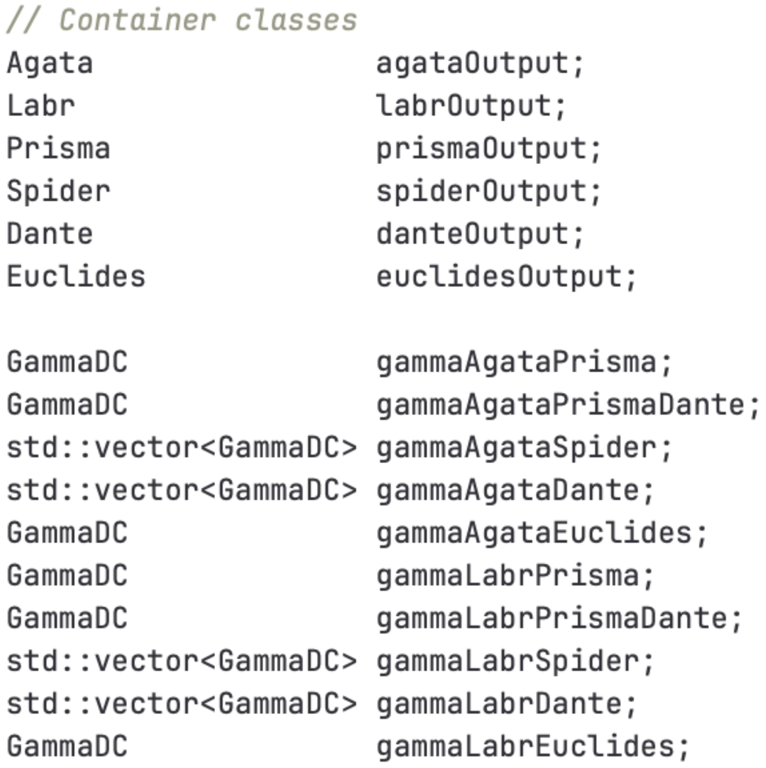
- The analysis is contained in src/Selector
- In the newest version, the detector analysis is separated from the coincidence analysis



Name	Last commit	Last update
..		
CoincidenceAnalysis	Bug fix with beta vs betaBP in gammadetectorPrisma	3 weeks ago
DetectorAnalysis	Adds the possibility of saving the doppler correction in the output tree	1 month ago
AgataSelector.cxx	Prisma-dante was missing	1 month ago
AgataSelector.h	Fixes bug in previous commit	1 month ago
CMakeLists.txt	Starting to move analysis in different classes	1 month ago

The analyzed data

- The ”Analyzed” data is saved in classes that can be accessed in later steps of the analysis.
- The doppler correction is also saved for each detector that can provide it
- These classes are the ones that can be saved also in root files
- They are kept in the directory: src/Container/



```
// Container classes
Agata                agataOutput;
Labr                 labrOutput;
Prisma               prismaOutput;
Spider               spiderOutput;
Dante                danteOutput;
Euclides             euclidesOutput;

GammaDC              gammaAgataPrisma;
GammaDC              gammaAgataPrismaDante;
std::vector<GammaDC> gammaAgataSpider;
std::vector<GammaDC> gammaAgataDante;
GammaDC              gammaAgataEuclides;
GammaDC              gammaLabrPrisma;
GammaDC              gammaLabrPrismaDante;
std::vector<GammaDC> gammaLabrSpider;
std::vector<GammaDC> gammaLabrDante;
GammaDC              gammaLabrEuclides;
```

Basic steps of the analysis procedure

- Each analysis is associated to a different folder and all have some steps:

At the start

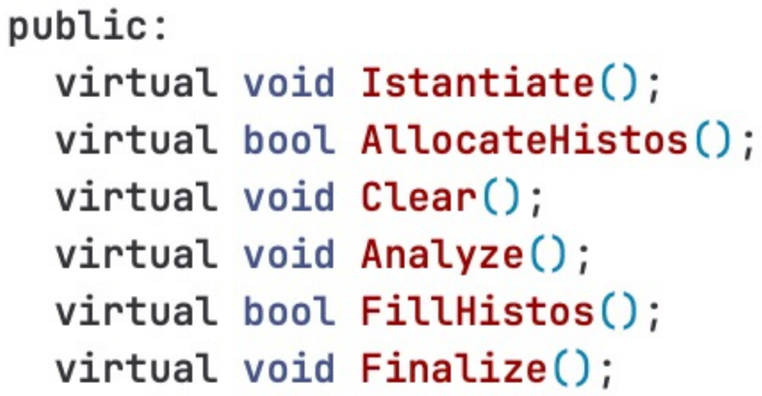
  - Istantiate->Associates to a detector the correct input
  - AllocateHistos->Allocates memory for the histograms if they are enabled

Every event

  - Clear -> Clears the containers of each detector
  - Analyze ->Fills the container based on the analysis
  - FillHistos -> Fills the histograms with the data in the containers

At the end

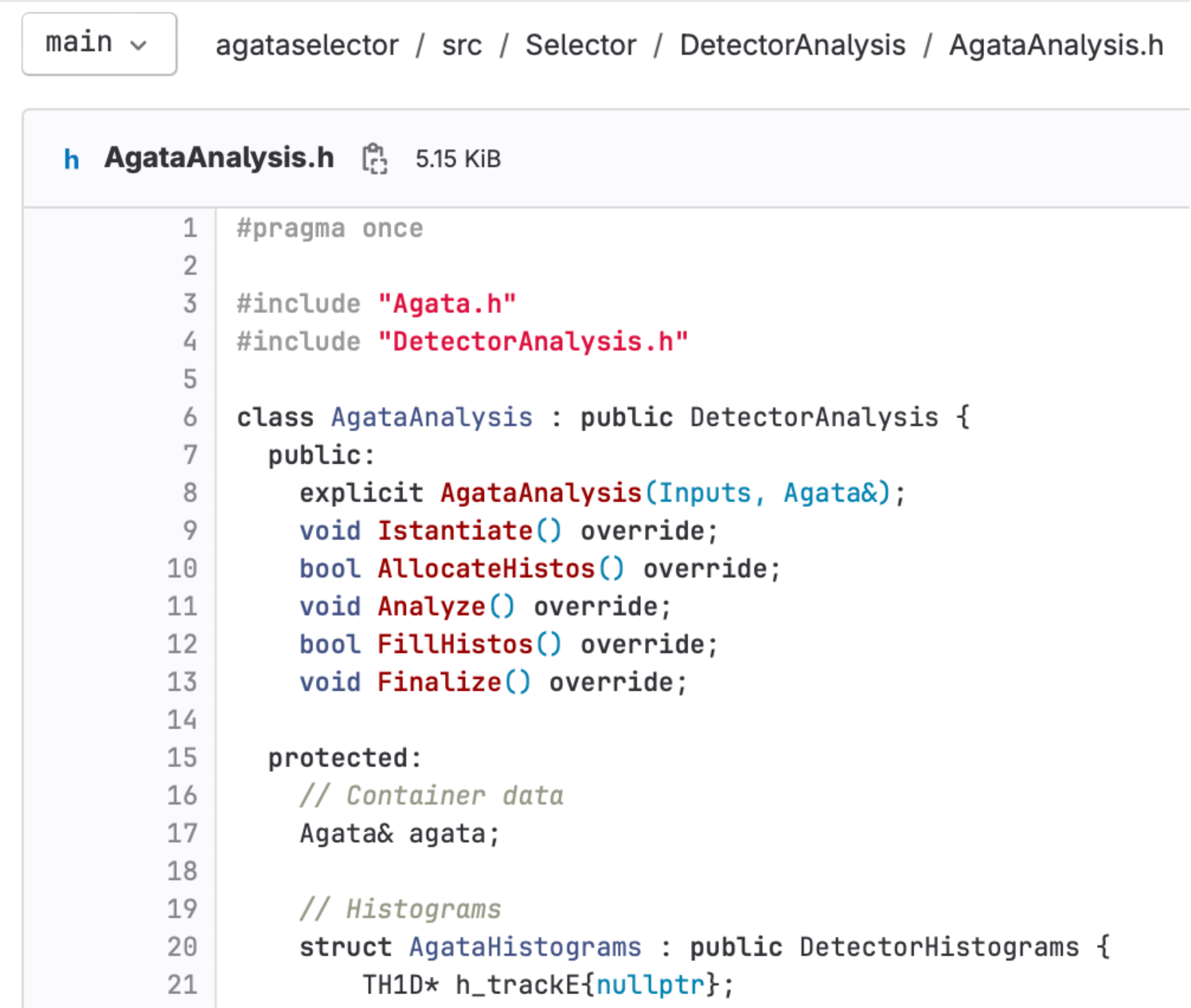
  - Finalize -> Performs operations on the final histogram



```
public:
  virtual void Istantiate();
  virtual bool AllocateHistos();
  virtual void Clear();
  virtual void Analyze();
  virtual bool FillHistos();
  virtual void Finalize();
```

How to add a histogram

- If you want to add an histogram you have to declare it in the correct analysis class' header
- The histogram are placed in a struct that correspond to the analysis folder
- Substructs and subfolders are also present



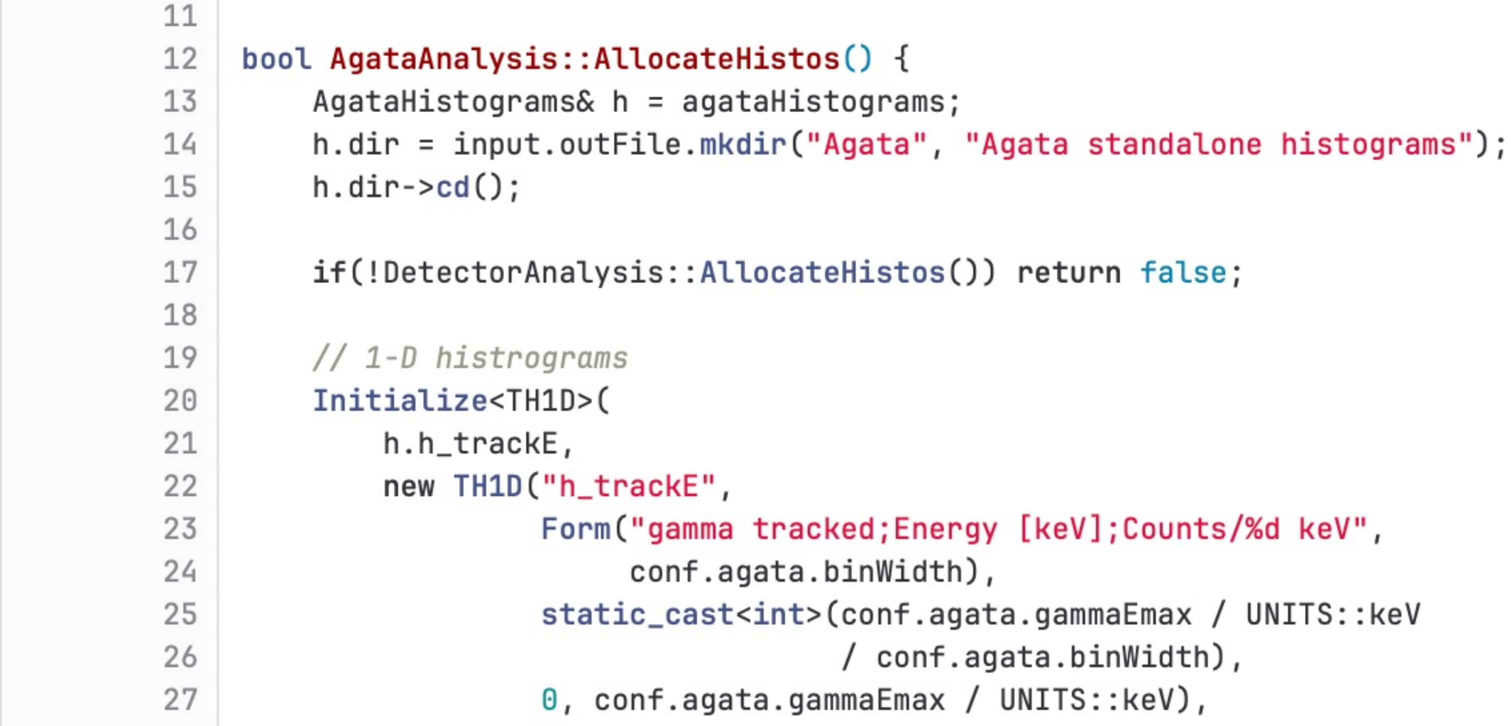
agataselector / src / Selector / DetectorAnalysis / AgataAnalysis.h

h AgataAnalysis.h 5.15 KiB

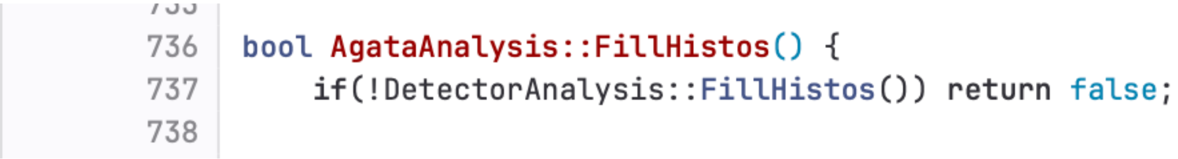
```
#pragma once

#include "Agata.h"
#include "DetectorAnalysis.h"

class AgataAnalysis : public DetectorAnalysis {
 public:
  explicit AgataAnalysis(Inputs, Agata&);
  void Istantiate() override;
  bool AllocateHistos() override;
  void Analyze() override;
  bool FillHistos() override;
  void Finalize() override;

 protected:
  // Container data
  Agata& agata;

  // Histograms
  struct AgataHistograms : public DetectorHistograms {
    TH1D* h_trackE{nullptr};
```

How to add a histogram

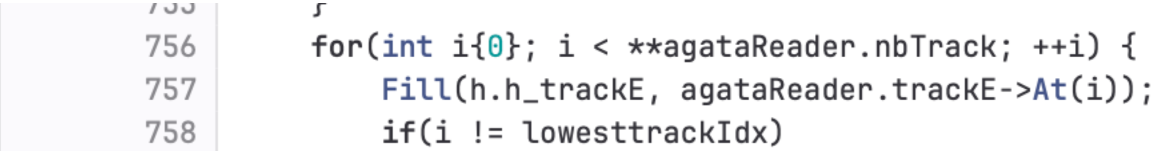
- Use the specific functions that ensure functionalities such as the detection of not enabled histograms:
    - **Allocate the memory with Initialize<TH*D>(…)**
    - **Fill with the Fill(…) function**



```cpp

bool AgataAnalysis::AllocateHistos() {
    AgataHistograms& h = agataHistograms;
    h.dir = input.outFile.mkdir("Agata", "Agata standalone histograms");
    h.dir->cd();

    if(!DetectorAnalysis::AllocateHistos()) return false;

    // 1-D histrograms
    Initialize<TH1D>(
        h.h_trackE,
        new TH1D("h_trackE",
                  Form("gamma tracked;Energy [keV];Counts/%d keV",
                       conf.agata.binWidth),
                  static_cast<int>(conf.agata.gammaEmax / UNITS::keV
                                   / conf.agata.binWidth),
                  0, conf.agata.gammaEmax / UNITS::keV),
```



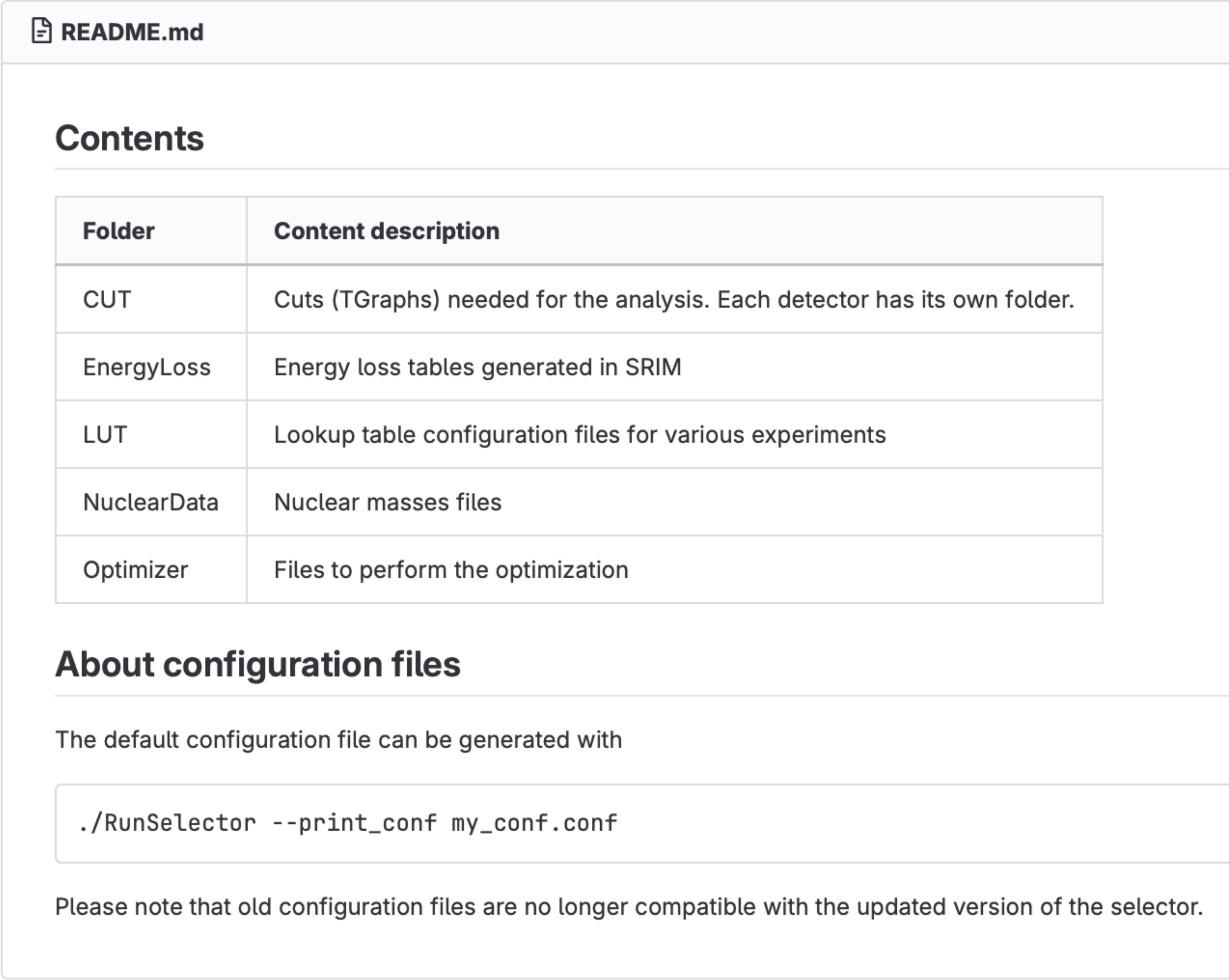
```cpp
bool AgataAnalysis::FillHistos() {
    if(!DetectorAnalysis::FillHistos()) return false;

```



```cpp
    for(int i{0}; i < **agataReader.nbTrack; ++i) {
        Fill(h.h_trackE, agataReader.trackE->At(i));
        if(i != lowesttrackIdx)
```

The Conf folder

- Contains the parameters used by the selector



README.md

Contents

Folder	Content description
CUT	Cuts (TGraphs) needed for the analysis. Each detector has its own folder.
EnergyLoss	Energy loss tables generated in SRIM
LUT	Lookup table configuration files for various experiments
NuclearData	Nuclear masses files
Optimizer	Files to perform the optimization

About configuration files

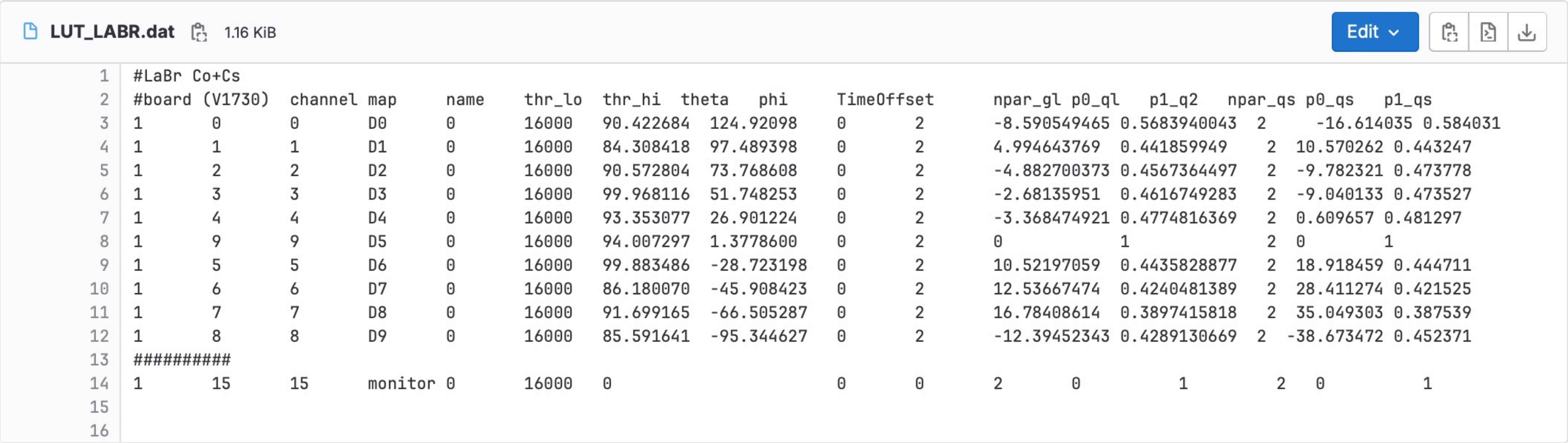
The default configuration file can be generated with

```
./RunSelector --print_conf my_conf.conf
```

Please note that old configuration files are no longer compatible with the updated version of the selector.

LUT

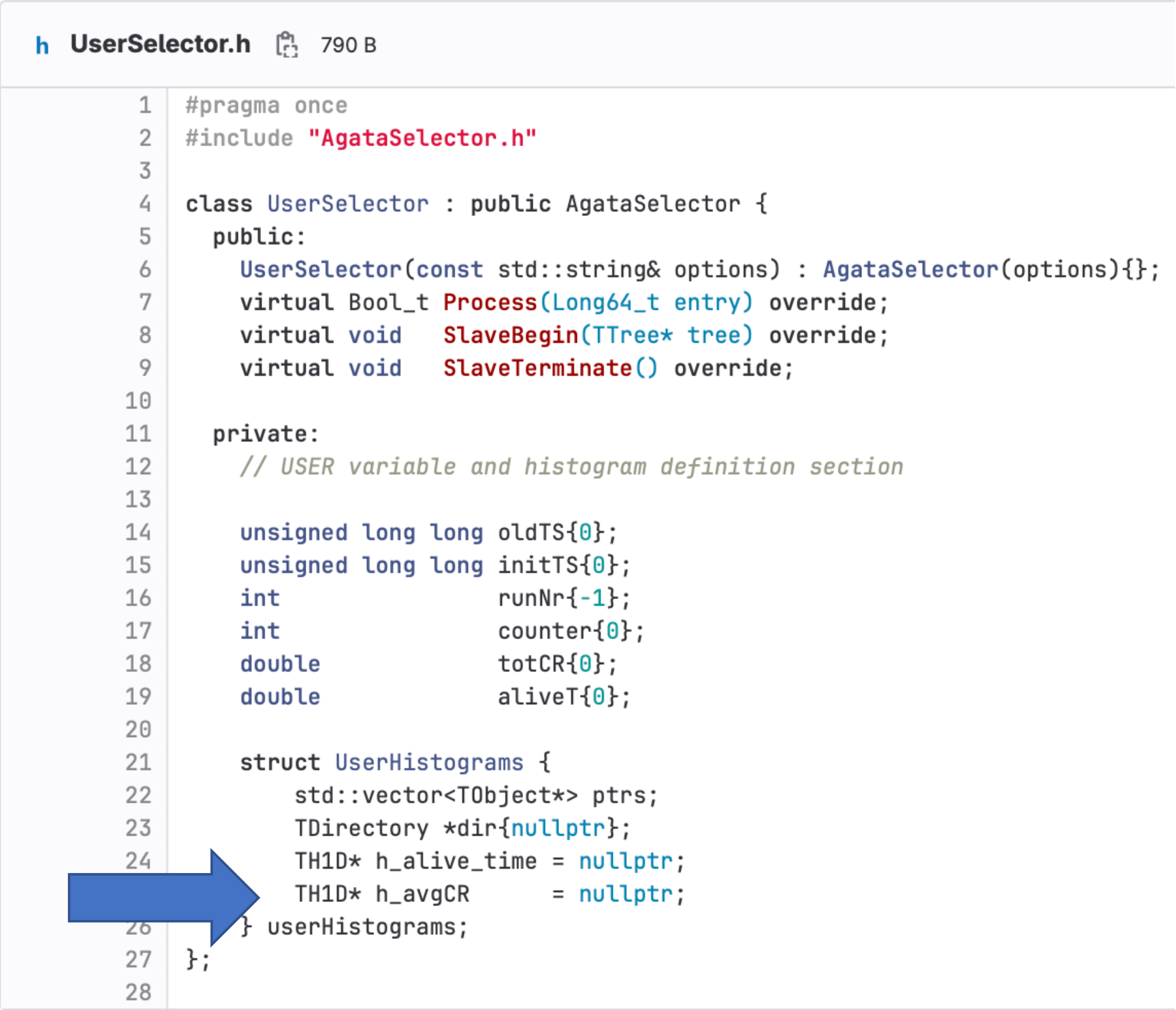
- The default LUTs can be found in User/EXP/Template/Conf/LUT/.
- The name of a channel+board combination is important for the analysis
- Generally, they allow to add an energy threshold (low, high), a time offset for alignment, and a N-degree polynomial calibration
- The remaining parameters are detector dependent and include angles or positions in space



```
#LaBr Co+Cs
#board (V1730)  channel map     name    thr_lo  thr_hi  theta   phi     TimeOffset      npar_gl p0_ql   p1_q2   npar_qs p0_qs   p1_qs
1       0       0       D0      0       16000   90.422684  124.92098    0       2       -8.590549465 0.5683940043  2     -16.614035 0.584031
1       1       1       D1      0       16000   84.308418  97.489398    0       2       4.994643769  0.441859949    2  10.570262 0.443247
1       2       2       D2      0       16000   90.572804  73.768608    0       2       -4.882700373 0.4567364497   2  -9.782321 0.473778
1       3       3       D3      0       16000   99.968116  51.748253    0       2       -2.68135951  0.4616749283   2  -9.040133 0.473527
1       4       4       D4      0       16000   93.353077  26.901224    0       2       -3.368474921 0.4774816369   2  0.609657 0.481297
1       9       9       D5      0       16000   94.007297  1.3778600    0       2       0            1              2  0        1
1       5       5       D6      0       16000   99.883486  -28.723198   0       2       10.52197059  0.4435828877   2  18.918459 0.444711
1       6       6       D7      0       16000   86.180070  -45.908423   0       2       12.53667474  0.4240481389   2  28.411274 0.421525
1       7       7       D8      0       16000   91.699165  -66.505287   0       2       16.78408614  0.3897415818   2  35.049303 0.387539
1       8       8       D9      0       16000   85.591641  -95.344627   0       2       -12.39452343 0.4289130669  2  -38.673472 0.452371
##########
1       15      15      monitor 0       16000   0                       0       0       2       0          1         2   0          1
```

The UserSelector

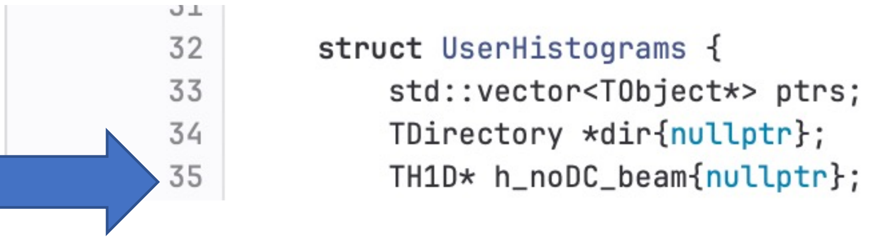
- If a part of the analysis is of general interest, it should be added to the regular part of the code under src/Selector
- However, in many cases some things are experiment-specific and can be handled by the UserSelector
- Histograms can be added to the struct in the header



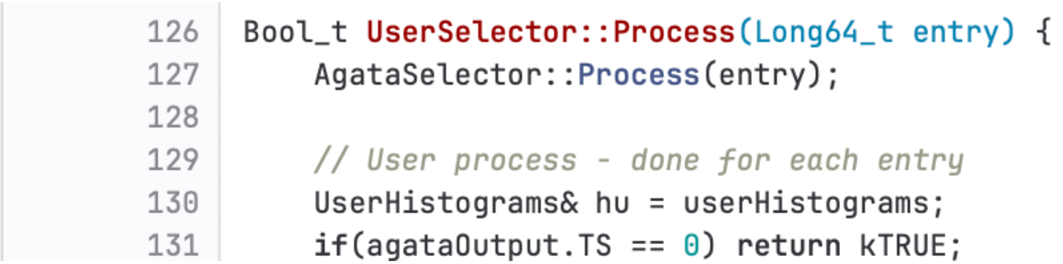
```cpp
#pragma once
#include "AgataSelector.h"

class UserSelector : public AgataSelector {
  public:
    UserSelector(const std::string& options) : AgataSelector(options){};
    virtual Bool_t Process(Long64_t entry) override;
    virtual void   SlaveBegin(TTree* tree) override;
    virtual void   SlaveTerminate() override;

  private:
    // USER variable and histogram definition section

    unsigned long long oldTS{0};
    unsigned long long initTS{0};
    int                runNr{-1};
    int                counter{0};
    double             totCR{0};
    double             aliveT{0};

    struct UserHistograms {
        std::vector<TObject*> ptrs;
        TDirectory *dir{nullptr};
        TH1D* h_alive_time = nullptr;
        TH1D* h_avgCR      = nullptr;
    } userHistograms;
};
```

The UserSelector

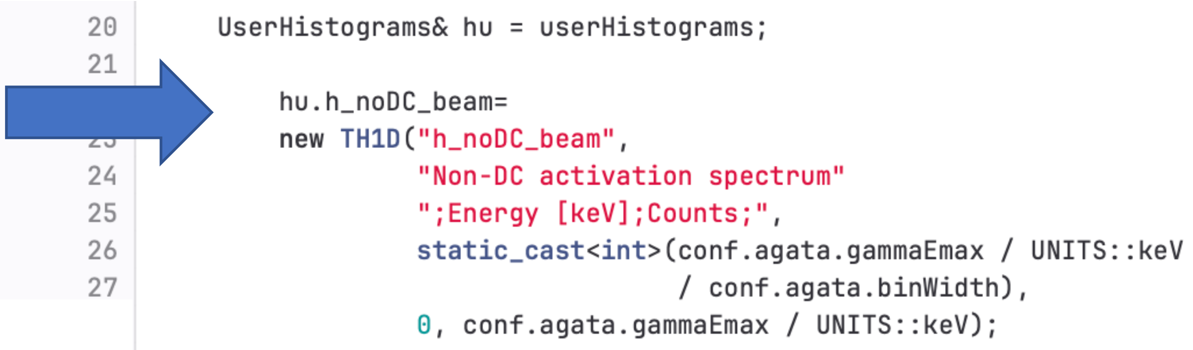
- Memory is allocated in the SlaveBegin
- Histograms are filled in the Process function
- Example from EXP_019



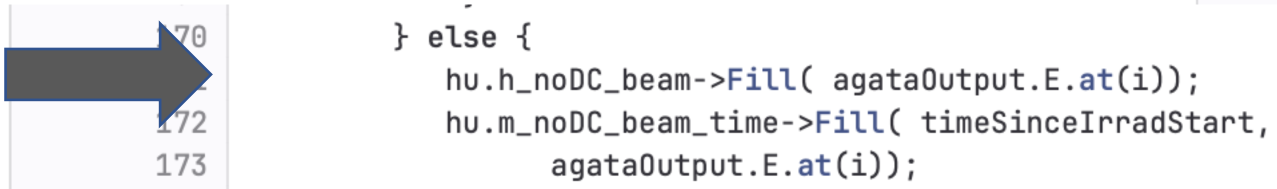
```
struct UserHistograms {
    std::vector<TObject*> ptrs;
    TDirectory *dir{nullptr};
    TH1D* h_noDC_beam{nullptr};
```



```
Bool_t UserSelector::Process(Long64_t entry) {
    AgataSelector::Process(entry);

    // User process - done for each entry
    UserHistograms& hu = userHistograms;
    if(agataOutput.TS == 0) return kTRUE;
```

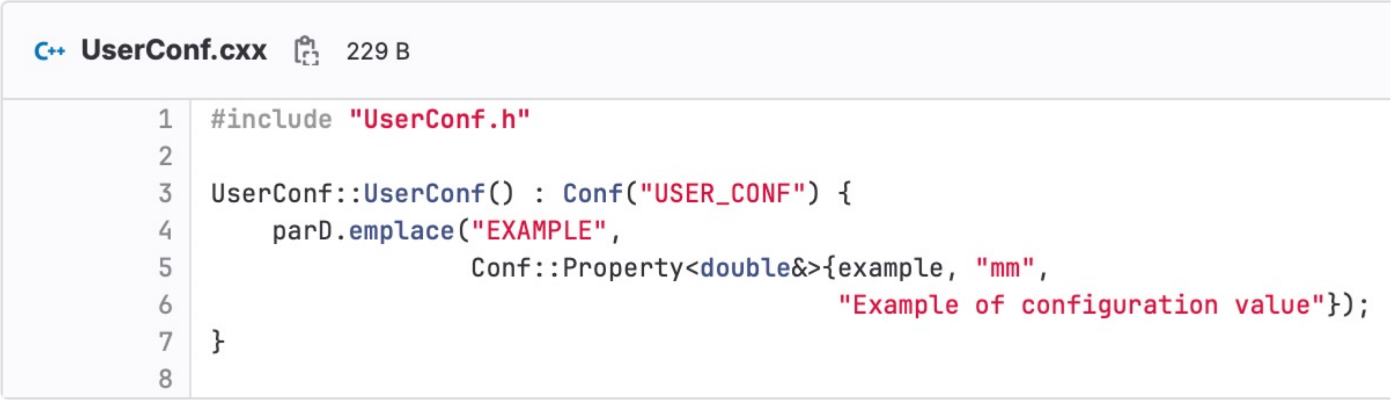


```
UserHistograms& hu = userHistograms;

    hu.h_noDC_beam=
    new TH1D("h_noDC_beam",
             "Non-DC activation spectrum"
             ";Energy [keV];Counts;",
             static_cast<int>(conf.agata.gammaEmax / UNITS::keV
                                / conf.agata.binWidth),
             0, conf.agata.gammaEmax / UNITS::keV);
```



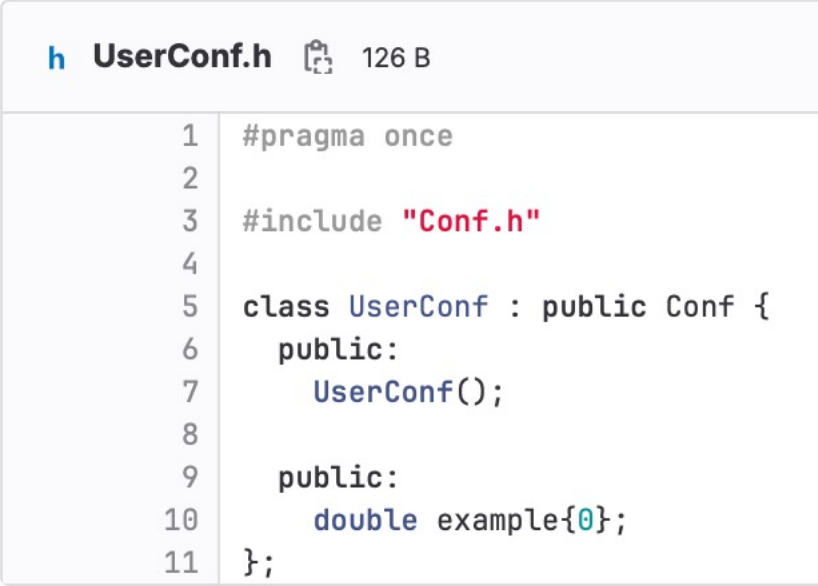
```
        } else {
            hu.h_noDC_beam->Fill( agataOutput.E.at(i));
            hu.m_noDC_beam_time->Fill( timeSinceIrradStart,
                    agataOutput.E.at(i));
```

The UserConf



```cpp
#include "UserConf.h"

UserConf::UserConf() : Conf("USER_CONF") {
    parD.emplace("EXAMPLE",
                 Conf::Property<double&>{example, "mm",
                                         "Example of configuration value"});
}

```

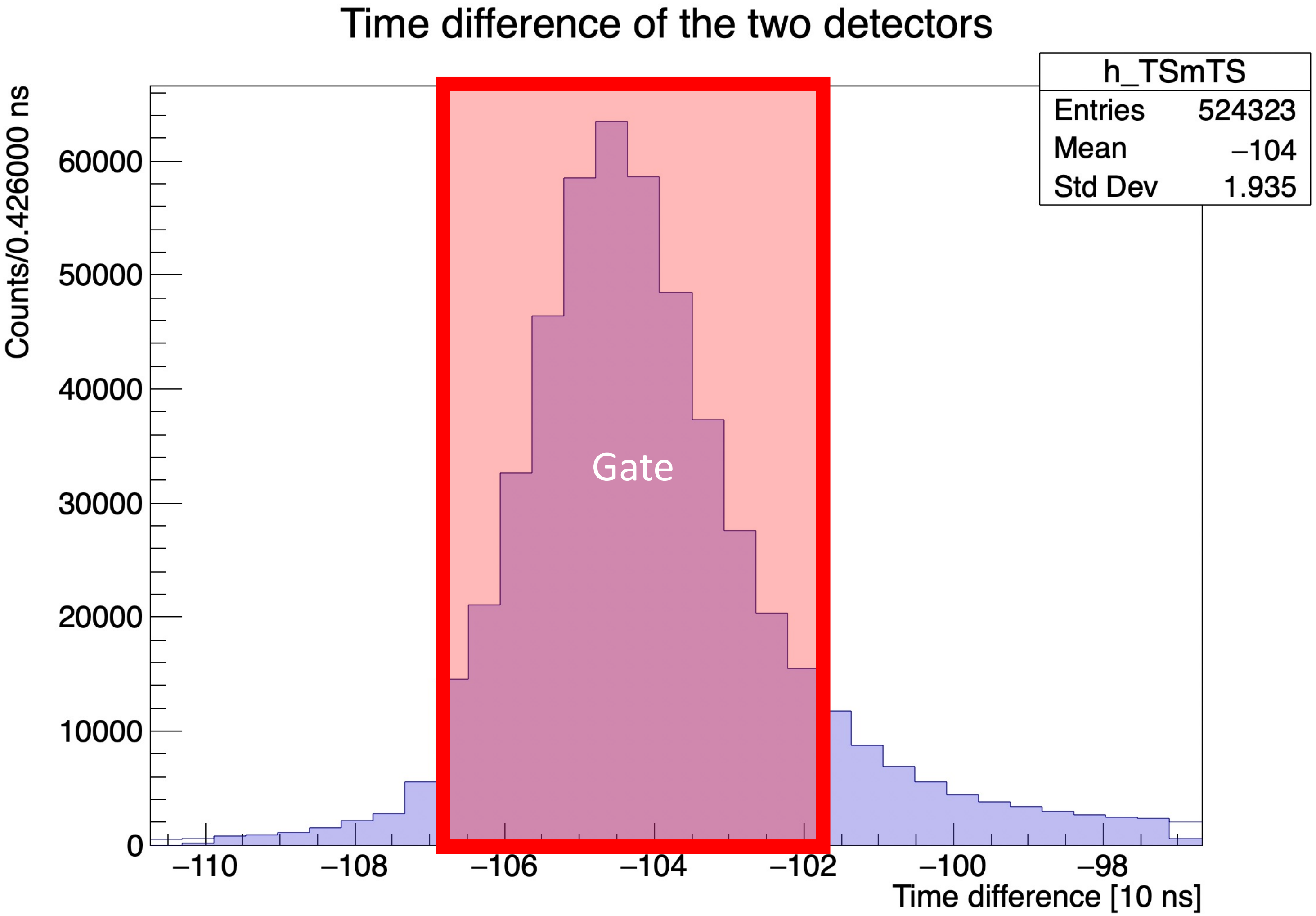
- It adds the possibility of reading custom parameters from the selector.conf
- You can declare a parameter in the .h and read it in the .cxx
- In this case the parameter to read is a double (parD), other types are also present such as strings (parS)
- Checkout src/Conf/Conf.h for other types



```cpp
#pragma once

#include "Conf.h"

class UserConf : public Conf {
  public:
    UserConf();

  public:
    double example{0};
};
```

Time coincidences

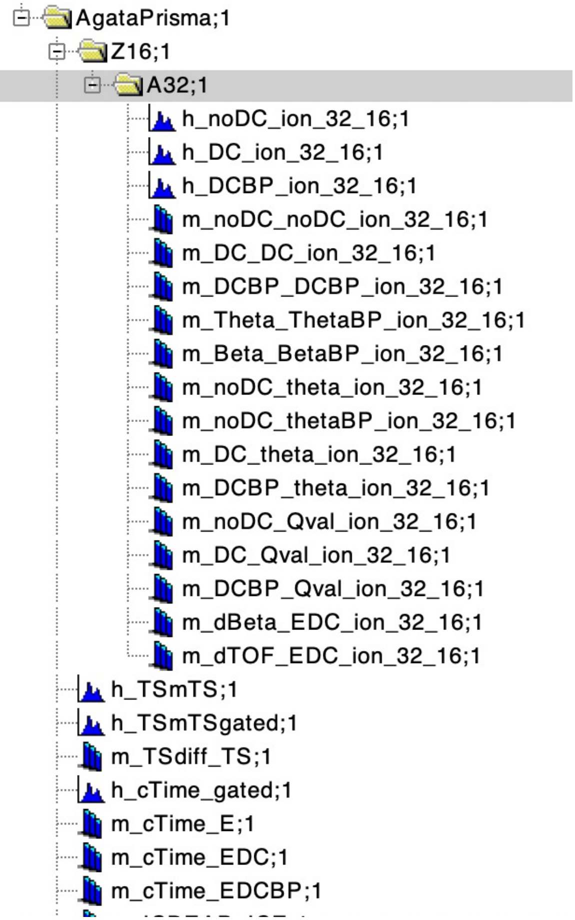
- All ancillary detectors need to be time-gated
- All coincidences will have a time difference histogram that is used to select the gate in the ***_CONF
- Some detectors such as Euclides, Agata have an internal time gate
- All time gates are set with the parameter:
  - COINC_W_LEFT
  - COINC_W_RIGHT



Detector specific analysis

Example: Agata-Prisma analysis

- Ion-gated histograms can be found in the Z##/A## folder.
- Important histograms include:
  - The DC(BP)_Qval which allows to gate on the total excitation energy of the system
  - The DC(BP,noDC)_Theta(BP) that allows to check the Doppler correction as a function of the angles
  - The various gamma-gamma matrices



```
AgataPrisma;1
  Z16;1
    A32;1
      h_noDC_ion_32_16;1
      h_DC_ion_32_16;1
      h_DCBP_ion_32_16;1
      m_noDC_noDC_ion_32_16;1
      m_DC_DC_ion_32_16;1
      m_DCBP_DCBP_ion_32_16;1
      m_Theta_ThetaBP_ion_32_16;1
      m_Beta_BetaBP_ion_32_16;1
      m_noDC_theta_ion_32_16;1
      m_noDC_thetaBP_ion_32_16;1
      m_DC_theta_ion_32_16;1
      m_DCBP_theta_ion_32_16;1
      m_noDC_Qval_ion_32_16;1
      m_DC_Qval_ion_32_16;1
      m_DCBP_Qval_ion_32_16;1
      m_dBeta_EDC_ion_32_16;1
      m_dTOF_EDC_ion_32_16;1
    h_TSmTS;1
    h_TSmTSgated;1
    m_TSdiff_TS;1
    h_cTime_gated;1
    m_cTime_E;1
    m_cTime_EDC;1
    m_cTime_EDCBP;1
```

Prisma

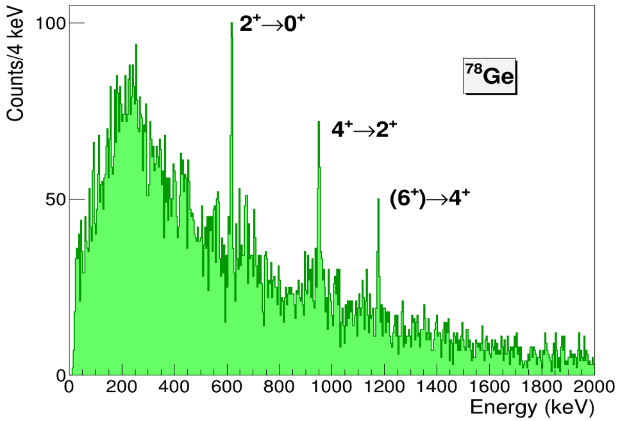
- The analysis of Prisma is more complex with respect to the other ancillaries and is mostly performed
- The selector can produce the histograms for the analysis on which one can set gates
- Refer to Elia Pilotto’s presentation

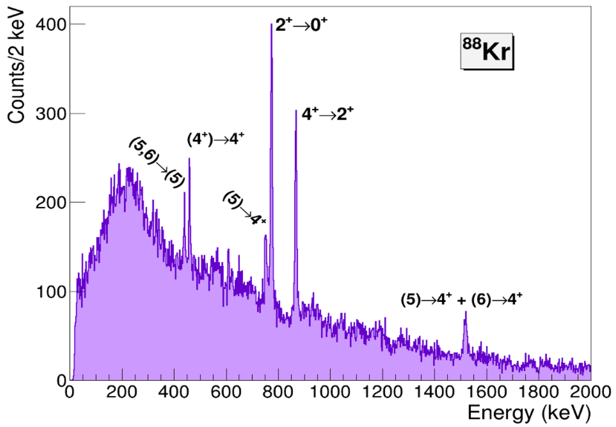
Prisma

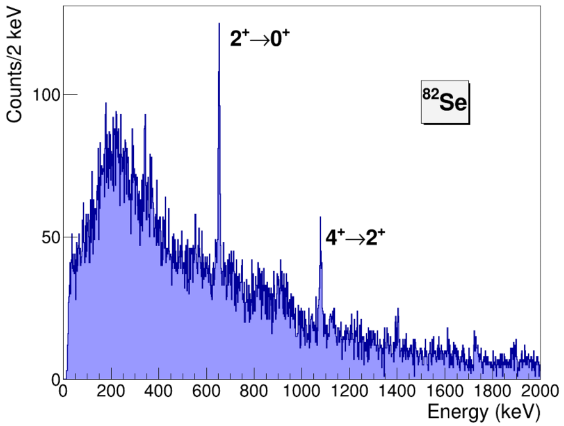
- Broken IC or PPAC channels can be disabled
- The TOF offset can be set in the selector to optimize the Doppler correction without disrupting the identification
- One can require or discard some parameters such as TOF_OK, IC_OK to perform the analysis
- Cuts in Z can be placed in Conf/CUT/Prisma/IC/ to produce histograms in coincidence
- In AGATAPRISMA_CONF it is possible to set EX_VALUES to gate on specific values of TKEL and generate additional histograms

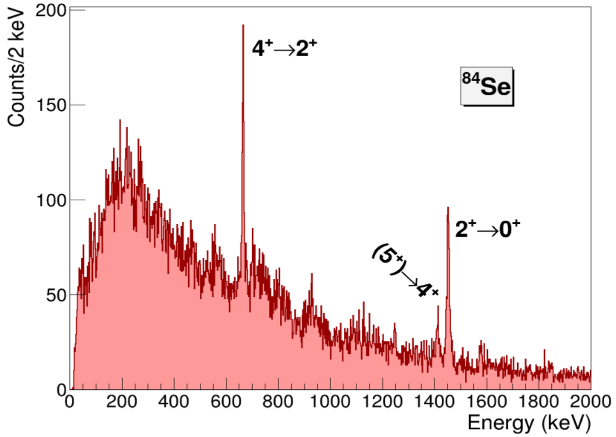
Prisma

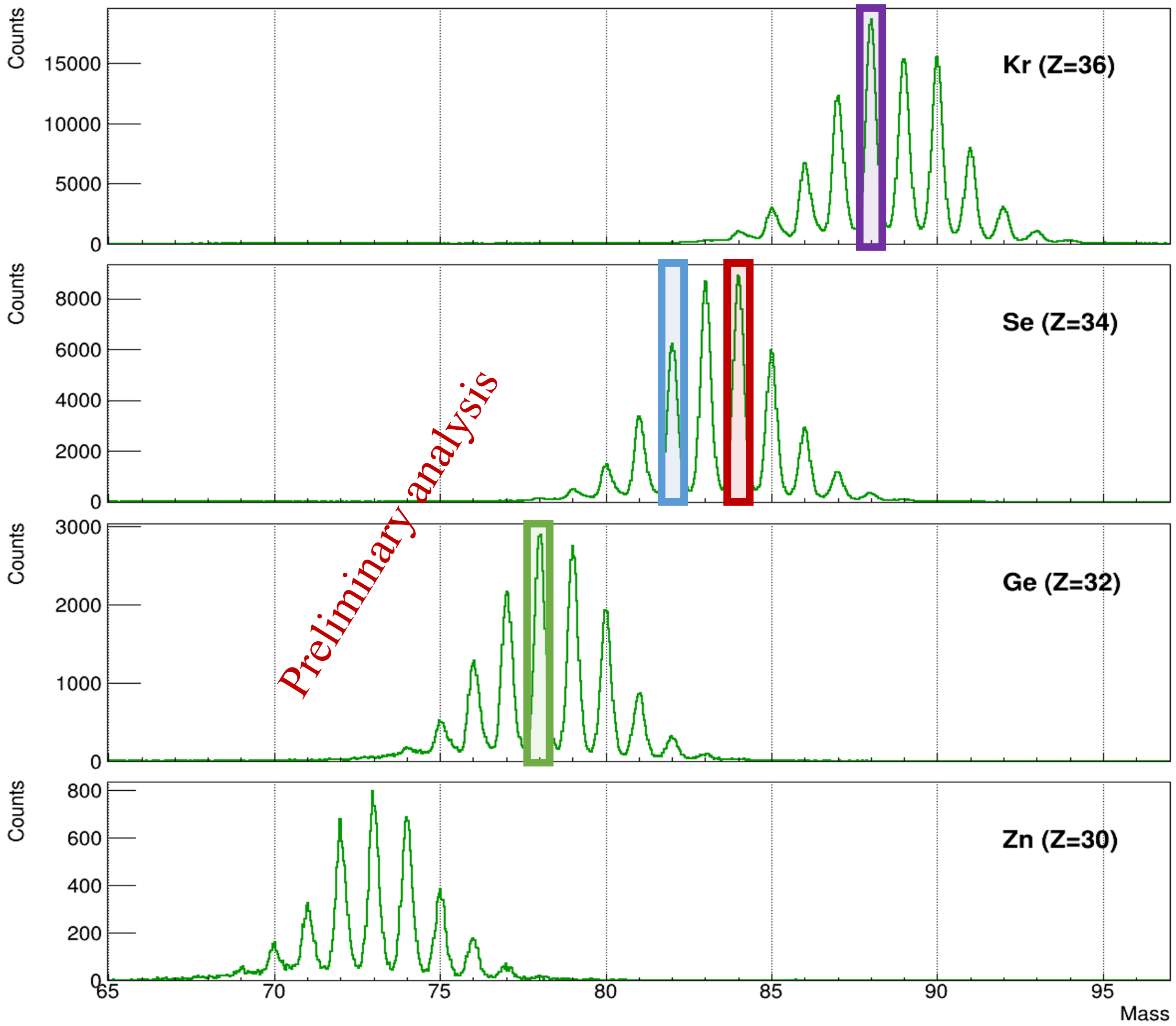
- Example of identification









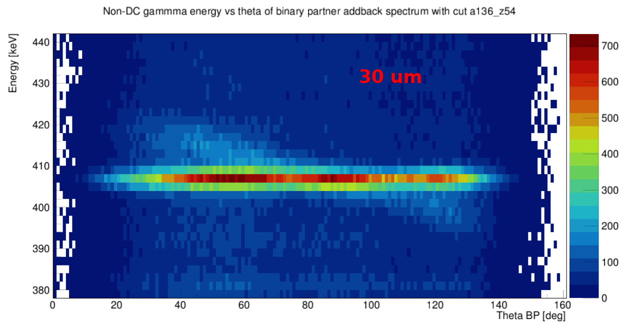


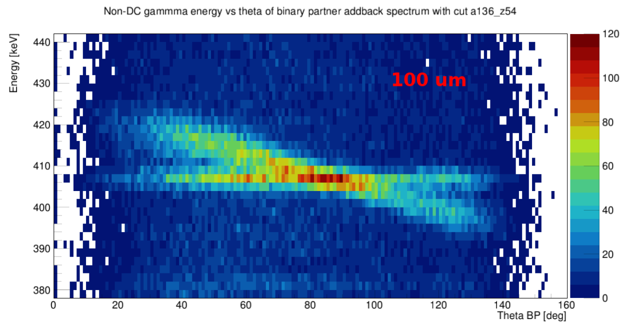
Preliminary analysis

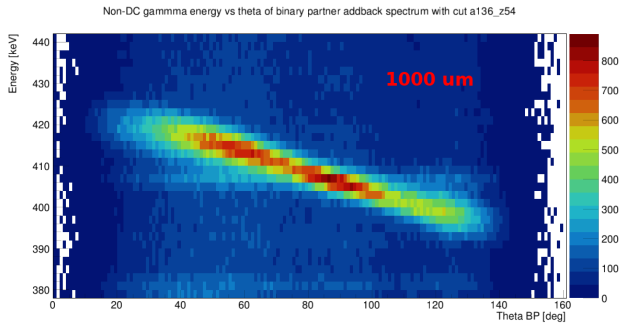
F. Angelini PhD Thesis

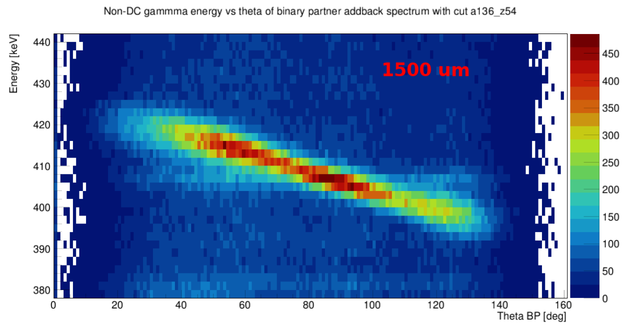
Prisma example

- Prisma provides the possibility of a fine kinematics reconstruction
- The Q-value matrices are often a very powerful tool
- The angle reconstruction is also great

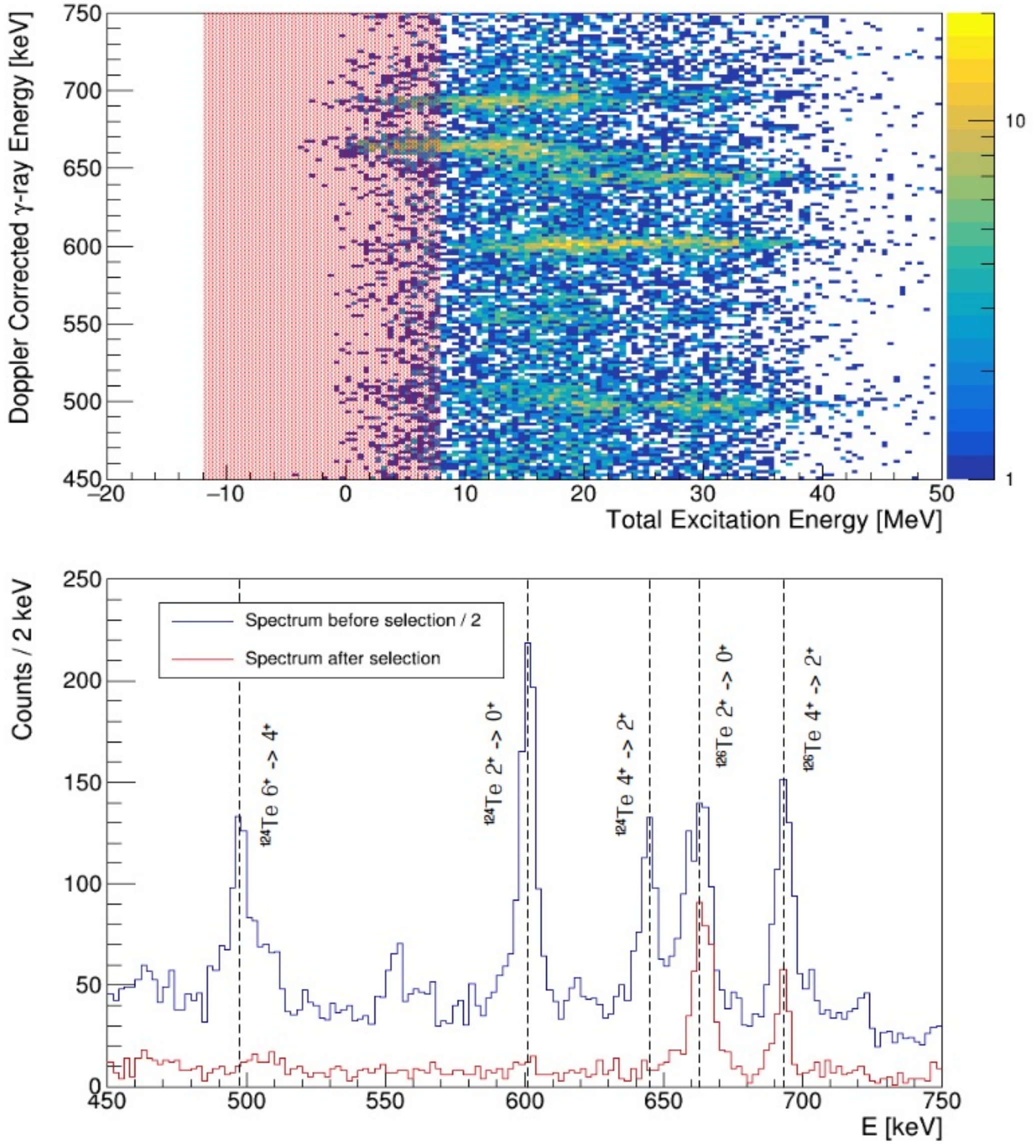








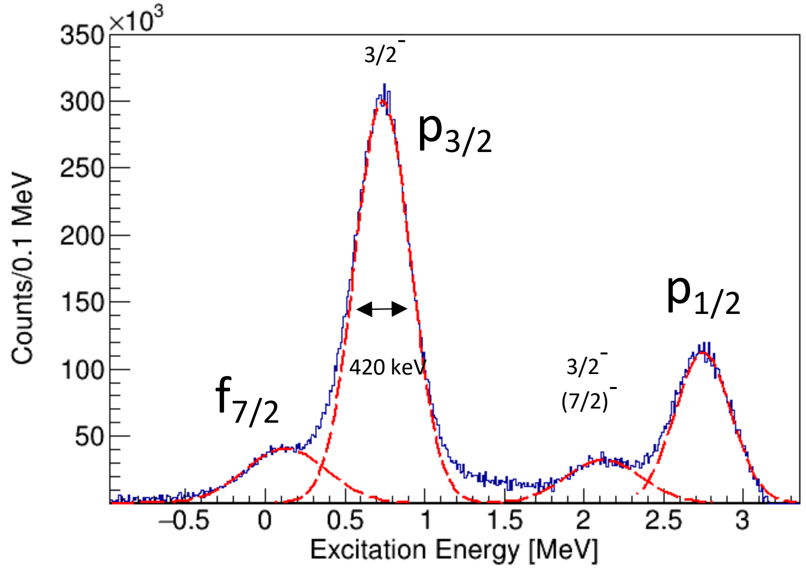
E. Pilotto Master Thesis

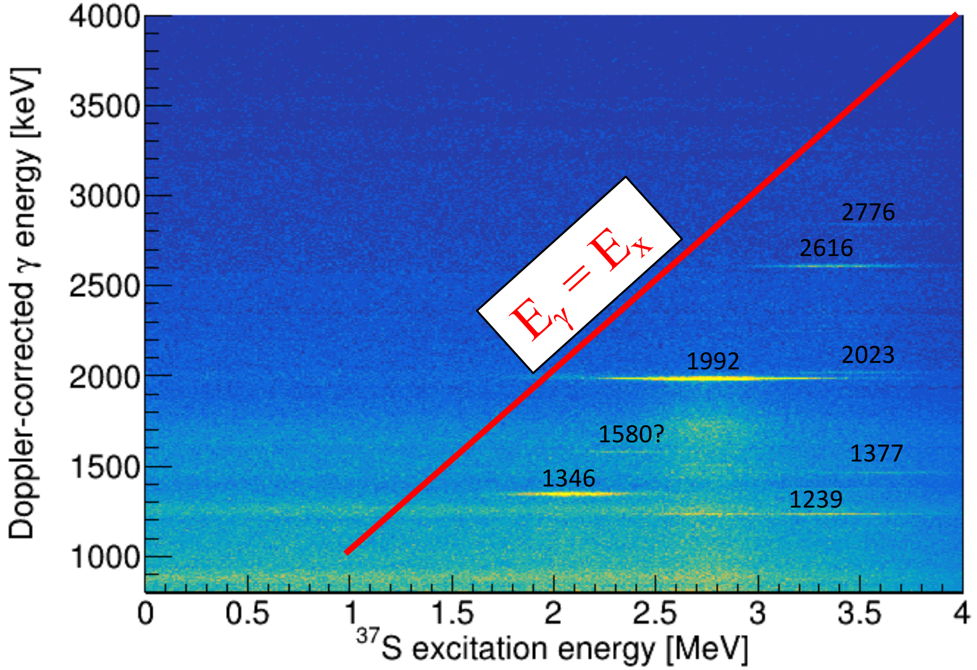


Spider

- The first step
- In some cases, the excitation energy can be very helpful
- The EX_VALUES keyword allows to generate histograms gated on the right value

L. Zago PhD Thesis



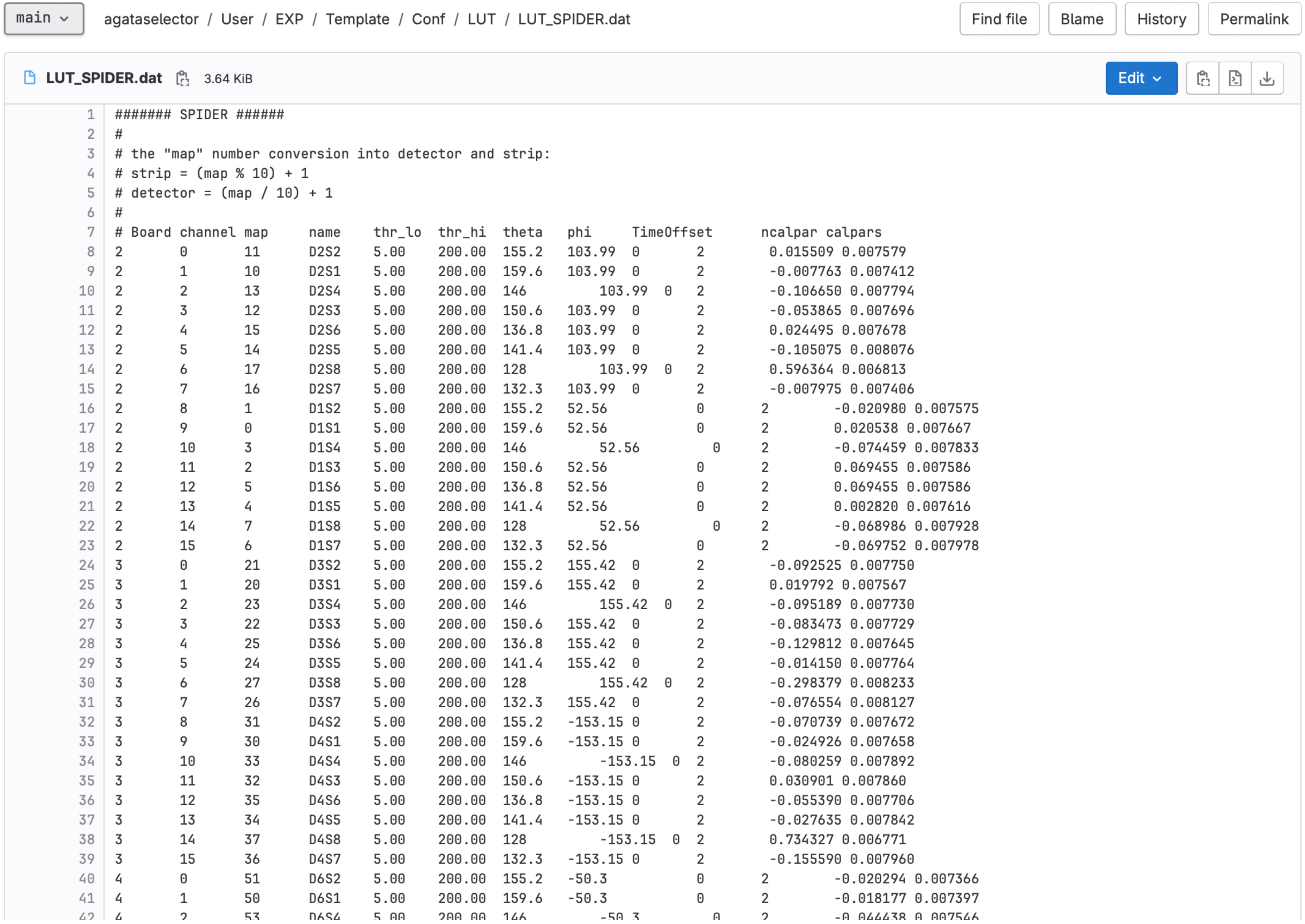


Spider

- Additionally, it is possible to gate on a gamma-ray to generate additional histograms in coincidence with it such as additional gamma-gamma matrices with the keyword GAMMA_GATE of AGATASPIDER_CONF
- Kinematic line TCuts can be placed in the Conf/CUT/SPIDER/ThetaLabELab folder
- To extract the optimal results it is possible to tune theta and phi of each spider channel to optimize the Doppler correction. This feature is under construction and testing and can be compiled running cmake with the option –DSPIDER_ANGCAL=On

Spider

- Detector dependent parameters:
  - Theta
  - Phi
- Channel names are not important but are helpful for the user


main
agataselector / User / EXP / Template / Conf / LUT / LUT_SPIDER.dat
Find file
Blame
History
Permalink
LUT_SPIDER.dat
3.64 KiB
Edit

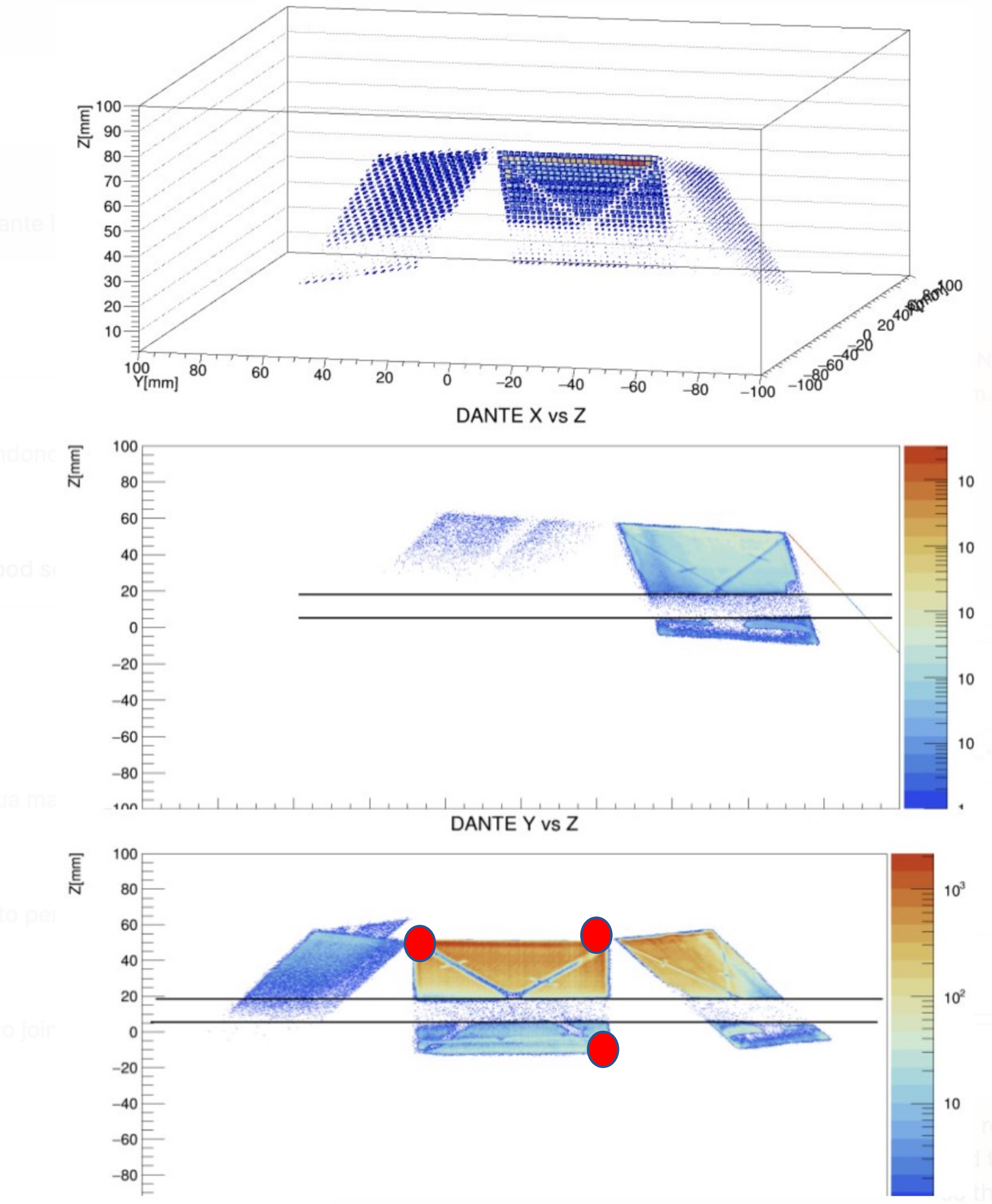
```
####### SPIDER ######
#
# the "map" number conversion into detector and strip:
# strip = (map % 10) + 1
# detector = (map / 10) + 1
#
# Board channel map     name    thr_lo  thr_hi  theta   phi     TimeOffset      ncalpar calpars
2       0       11      D2S2    5.00    200.00  155.2   103.99  0       2         0.015509 0.007579
2       1       10      D2S1    5.00    200.00  159.6   103.99  0       2         -0.007763 0.007412
2       2       13      D2S4    5.00    200.00  146         103.99  0   2         -0.106650 0.007794
2       3       12      D2S3    5.00    200.00  150.6   103.99  0       2         -0.053865 0.007696
2       4       15      D2S6    5.00    200.00  136.8   103.99  0       2         0.024495 0.007678
2       5       14      D2S5    5.00    200.00  141.4   103.99  0       2         -0.105075 0.008076
2       6       17      D2S8    5.00    200.00  128         103.99  0   2         0.596364 0.006813
2       7       16      D2S7    5.00    200.00  132.3   103.99  0       2         -0.007975 0.007406
2       8       1       D1S2    5.00    200.00  155.2   52.56           0       2        -0.020980 0.007575
2       9       0       D1S1    5.00    200.00  159.6   52.56           0       2        0.020538 0.007667
2       10      3       D1S4    5.00    200.00  146         52.56         0     2        -0.074459 0.007833
2       11      2       D1S3    5.00    200.00  150.6   52.56           0       2        0.069455 0.007586
2       12      5       D1S6    5.00    200.00  136.8   52.56           0       2        0.069455 0.007586
2       13      4       D1S5    5.00    200.00  141.4   52.56           0       2        0.002820 0.007616
2       14      7       D1S8    5.00    200.00  128         52.56         0     2        -0.068986 0.007928
2       15      6       D1S7    5.00    200.00  132.3   52.56           0       2        -0.069752 0.007978
3       0       21      D3S2    5.00    200.00  155.2   155.42  0       2         -0.092525 0.007750
3       1       20      D3S1    5.00    200.00  159.6   155.42  0       2         0.019792 0.007567
3       2       23      D3S4    5.00    200.00  146         155.42  0   2         -0.095189 0.007730
3       3       22      D3S3    5.00    200.00  150.6   155.42  0       2         -0.083473 0.007729
3       4       25      D3S6    5.00    200.00  136.8   155.42  0       2         -0.129812 0.007645
3       5       24      D3S5    5.00    200.00  141.4   155.42  0       2         -0.014150 0.007764
3       6       27      D3S8    5.00    200.00  128         155.42  0   2         -0.298379 0.008233
3       7       26      D3S7    5.00    200.00  132.3   155.42  0       2         -0.076554 0.008127
3       8       31      D4S2    5.00    200.00  155.2   -153.15 0       2         -0.070739 0.007672
3       9       30      D4S1    5.00    200.00  159.6   -153.15 0       2         -0.024926 0.007658
3       10      33      D4S4    5.00    200.00  146         -153.15   0 2         -0.080259 0.007892
3       11      32      D4S3    5.00    200.00  150.6   -153.15 0       2         0.030901 0.007860
3       12      35      D4S6    5.00    200.00  136.8   -153.15 0       2         -0.055390 0.007706
3       13      34      D4S5    5.00    200.00  141.4   -153.15 0       2         -0.027635 0.007842
3       14      37      D4S8    5.00    200.00  128         -153.15   0 2         0.734327 0.006771
3       15      36      D4S7    5.00    200.00  132.3   -153.15 0       2         -0.155590 0.007960
4       0       51      D6S2    5.00    200.00  155.2   -50.3           0       2        -0.020294 0.007366
4       1       50      D6S1    5.00    200.00  159.6   -50.3           0       2        -0.018177 0.007397
4       2       53      D6S4    5.00    200.00  146         -50.3         0     2        -0.044438 0.007546
```

Dante

- In general, a “perfect” Dante event should contain at least 3 events, corresponding to x, y (TACS) and T (cfd logic signal)
- Additionally, a TAC can be placed between Dante and Prisma, this is also handled by the analysis
- This does not happen all the time and the selector should handle this, some options are present in the selector.conf file
- The spatial calibration is performed by selecting the (x,y) points of the extremities of the
- The analysis should be expanded and improved for Dante

Dante

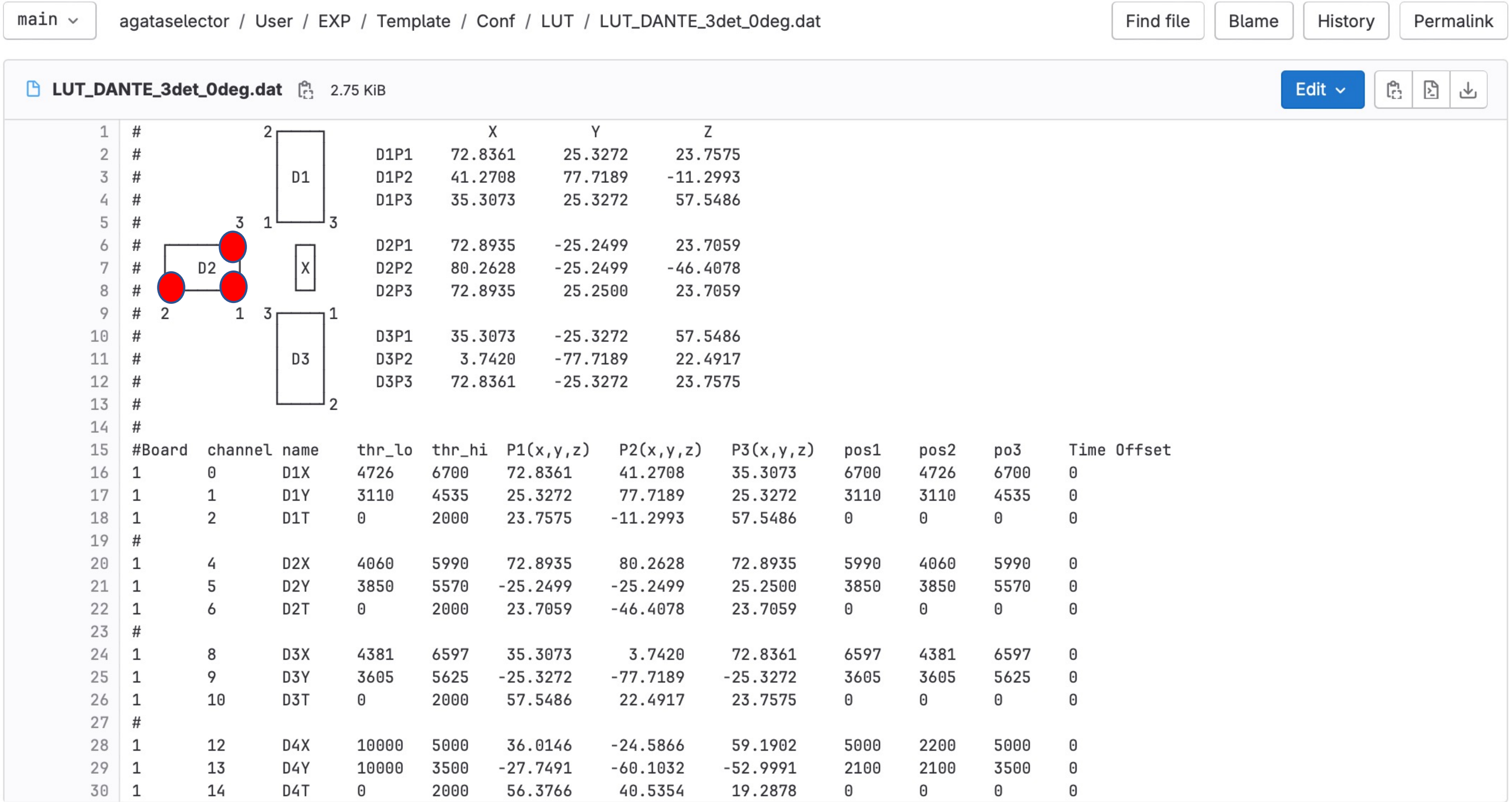
- The position is used to refine the Doppler correction
- It is possible to set gates in Conf/Cuts/PrismaDante/TOF_TKEL
- In this case of the triple coincidence AGATA-PRISMA-DANTE it is necessary to set two time gates: agata-prisma and agata-dante



Dante

- Detector dependent parameters:
  - P1, P2, P3
  - pos1, pos2, pos3

- Channel names distinguish X, Y, T and TOF


main ⌄
agataselector / User / EXP / Template / Conf / LUT / LUT_DANTE_3det_0deg.dat
Find file
Blame
History
Permalink
LUT_DANTE_3det_0deg.dat
2.75 KiB
Edit ⌄

```
#              2 ┌────┐                 X          Y            Z
#                │    │    D1P1    72.8361    25.3272     23.7575
#                │ D1 │    D1P2    41.2708    77.7189    -11.2993
#                │    │    D1P3    35.3073    25.3272     57.5486
#           3  1 └────┘ 3
#   ┌───────┐     ┌─┐      D2P1    72.8935   -25.2499     23.7059
#   │  D2   │     │X│      D2P2    80.2628   -25.2499    -46.4078
#   └───────┘     └─┘      D2P3    72.8935    25.2500     23.7059
#  2        1  3 ┌────┐ 1
#                │    │    D3P1    35.3073   -25.3272     57.5486
#                │ D3 │    D3P2     3.7420   -77.7189     22.4917
#                │    │    D3P3    72.8361   -25.3272     23.7575
#                └────┘ 2
#
#Board  channel name    thr_lo  thr_hi  P1(x,y,z)   P2(x,y,z)   P3(x,y,z)   pos1    pos2    po3     Time Offset
1       0       D1X     4726    6700    72.8361     41.2708     35.3073     6700    4726    6700    0
1       1       D1Y     3110    4535    25.3272     77.7189     25.3272     3110    3110    4535    0
1       2       D1T     0       2000    23.7575    -11.2993     57.5486     0       0       0       0
#
1       4       D2X     4060    5990    72.8935     80.2628     72.8935     5990    4060    5990    0
1       5       D2Y     3850    5570   -25.2499    -25.2499     25.2500     3850    3850    5570    0
1       6       D2T     0       2000    23.7059    -46.4078     23.7059     0       0       0       0
#
1       8       D3X     4381    6597    35.3073      3.7420     72.8361     6597    4381    6597    0
1       9       D3Y     3605    5625   -25.3272    -77.7189    -25.3272     3605    3605    5625    0
1       10      D3T     0       2000    57.5486     22.4917     23.7575     0       0       0       0
#
1       12      D4X     10000   5000    36.0146    -24.5866     59.1902     5000    2200    5000    0
1       13      D4Y     10000   3500   -27.7491    -60.1032    -52.9991     2100    2100    3500    0
1       14      D4T     0       2000    56.3766     40.5354     19.2878     0       0       0       0
```

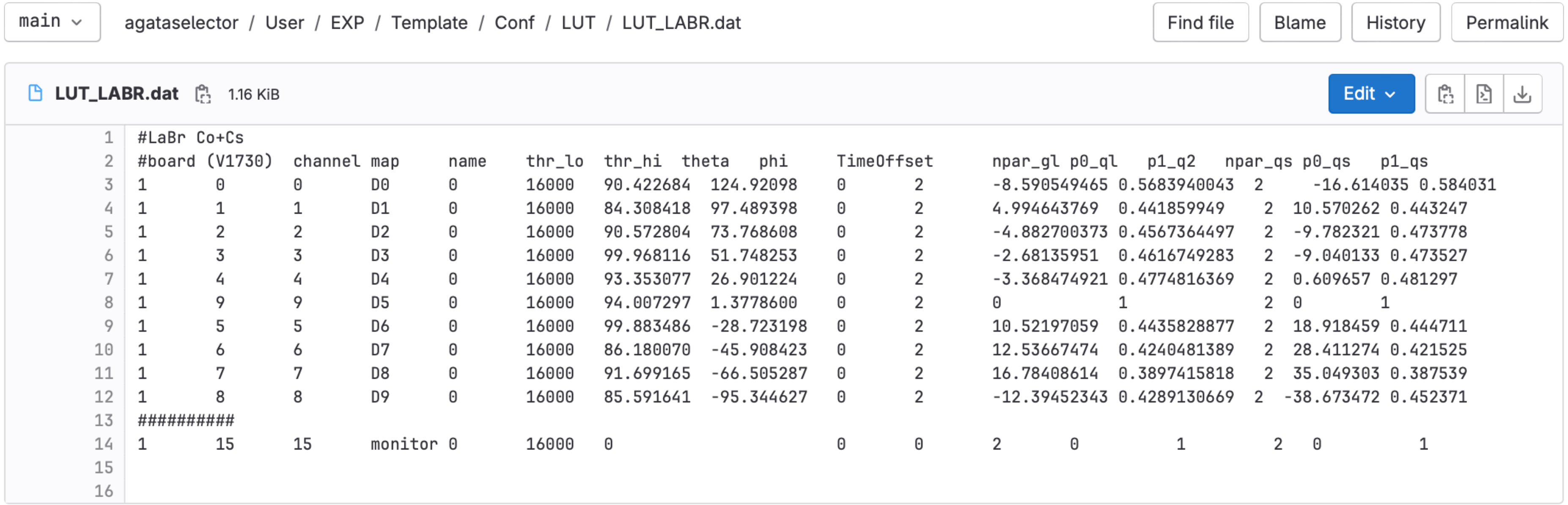
The lookup table also performs the 3D position reconstruction of DANTE, mapping 2D points (pos1, pos2, pos2) to 3D points (P1, P2, P3)

LaBr

- They share the same base class of Agata: GammaDetector
- As a consequence, the analysis of coincidences with Agata is exactly the same, so you can perform the same analysis as for Agata.
- In some cases, they were acquired without external trigger, meaning that they will have a lot of data. In this case you can use the --labr_slave option of ReadCaenRaw and/or the mandatoryKey of femul to process their data only if it is in coincidence with other ancillaries in the first case or femul in the second.
- In the case of experiments with Prisma, we discovered that they are strongly affected by the magnetic field despite the shielding so they require a calibration when the magnet was on

Labr

- Detector dependent parameters:
    - Theta
    - Phi



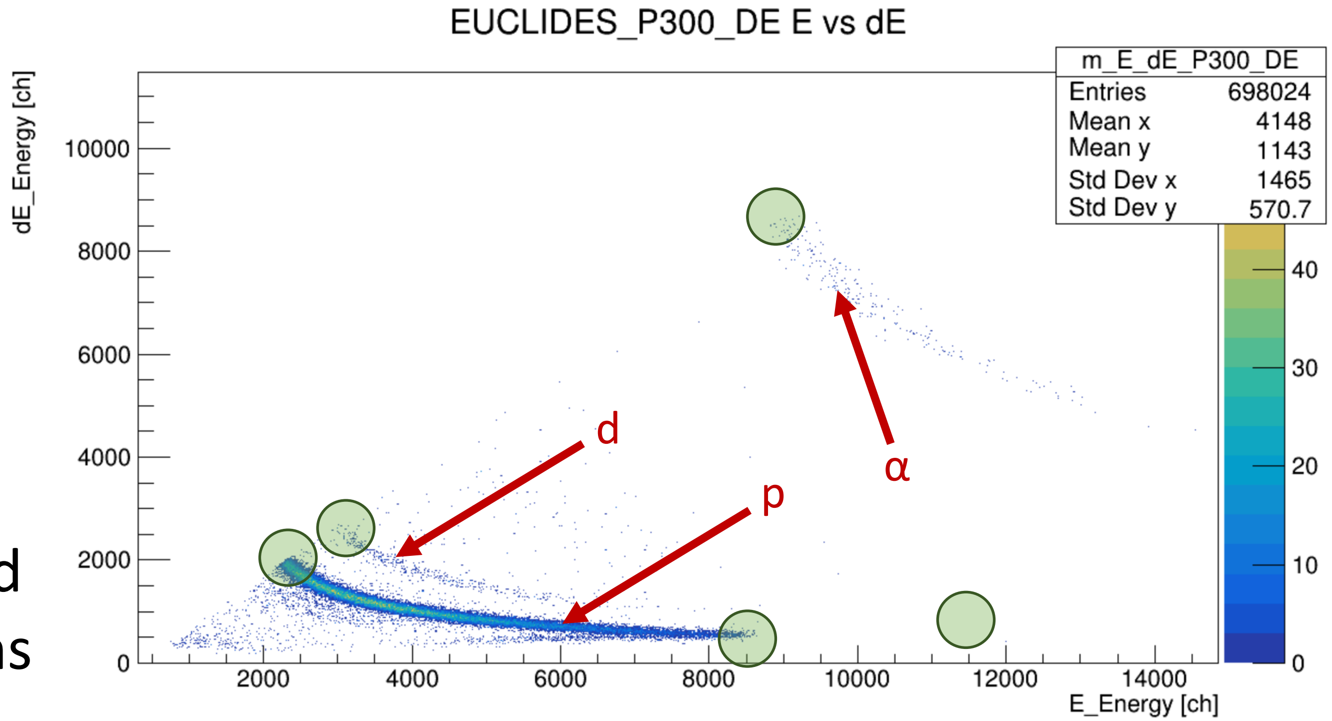
agataselector / User / EXP / Template / Conf / LUT / LUT_LABR.dat

LUT_LABR.dat 1.16 KiB

```
#LaBr Co+Cs
#board (V1730)  channel map     name    thr_lo  thr_hi  theta   phi     TimeOffset      npar_gl p0_ql   p1_q2   npar_qs p0_qs   p1_qs
1       0       0       D0      0       16000   90.422684  124.92098    0       2       -8.590549465 0.5683940043  2     -16.614035 0.584031
1       1       1       D1      0       16000   84.308418  97.489398    0       2       4.994643769  0.441859949    2  10.570262 0.443247
1       2       2       D2      0       16000   90.572804  73.768608    0       2       -4.882700373 0.4567364497   2  -9.782321 0.473778
1       3       3       D3      0       16000   99.968116  51.748253    0       2       -2.68135951  0.4616749283   2  -9.040133 0.473527
1       4       4       D4      0       16000   93.353077  26.901224    0       2       -3.368474921 0.4774816369   2  0.609657 0.481297
1       9       9       D5      0       16000   94.007297  1.3778600    0       2       0            1              2  0        1
1       5       5       D6      0       16000   99.883486  -28.723198   0       2       10.52197059  0.4435828877   2  18.918459 0.444711
1       6       6       D7      0       16000   86.180070  -45.908423   0       2       12.53667474  0.4240481389   2  28.411274 0.421525
1       7       7       D8      0       16000   91.699165  -66.505287   0       2       16.78408614  0.3897415818   2  35.049303 0.387539
1       8       8       D9      0       16000   85.591641  -95.344627   0       2       -12.39452343 0.4289130669  2  -38.673472 0.452371
##########
1       15      15      monitor 0       16000   0                       0       0       2       0          1         2   0          1


```

Euclides

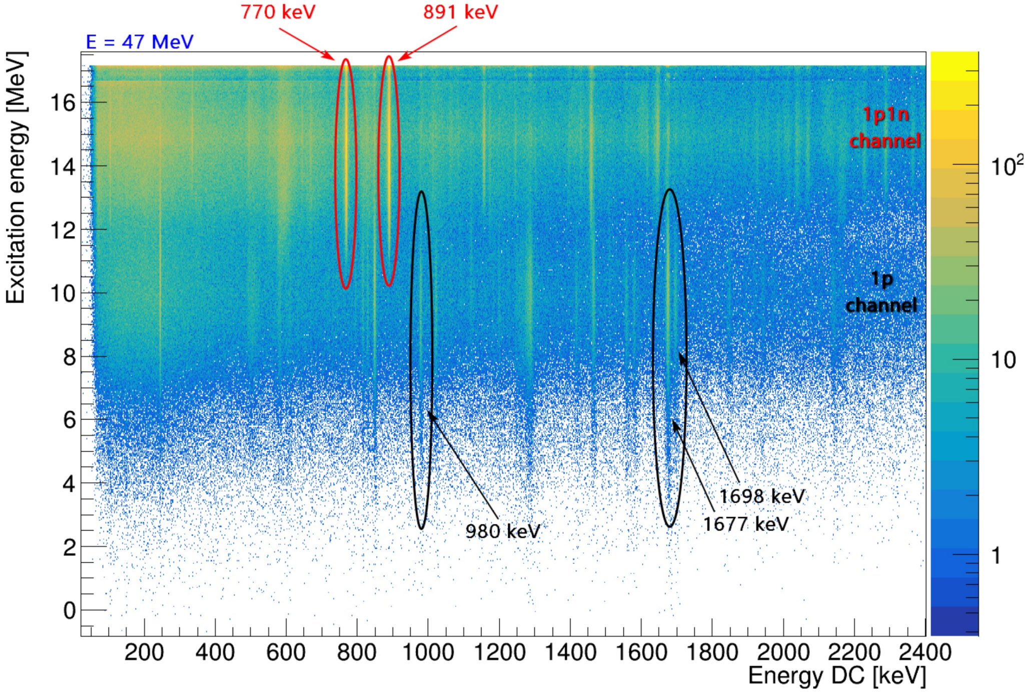
- Set gates for all telescopes in Conf/CUT/EUCLIDES/EdE/
- The naming scheme to adopt should be z1_m1_mapnr for while for alphas it should be z2_m2_mapnr
- Check the time alignment
- Calibrate with alpha run or with punch trough points 🟢
- In this case the reaction of interest could be a Nbody reaction. In this case the ions of interest need to be specified with:
  IONS A1 Z1 A2 Z2 A3 Z3 END Comment

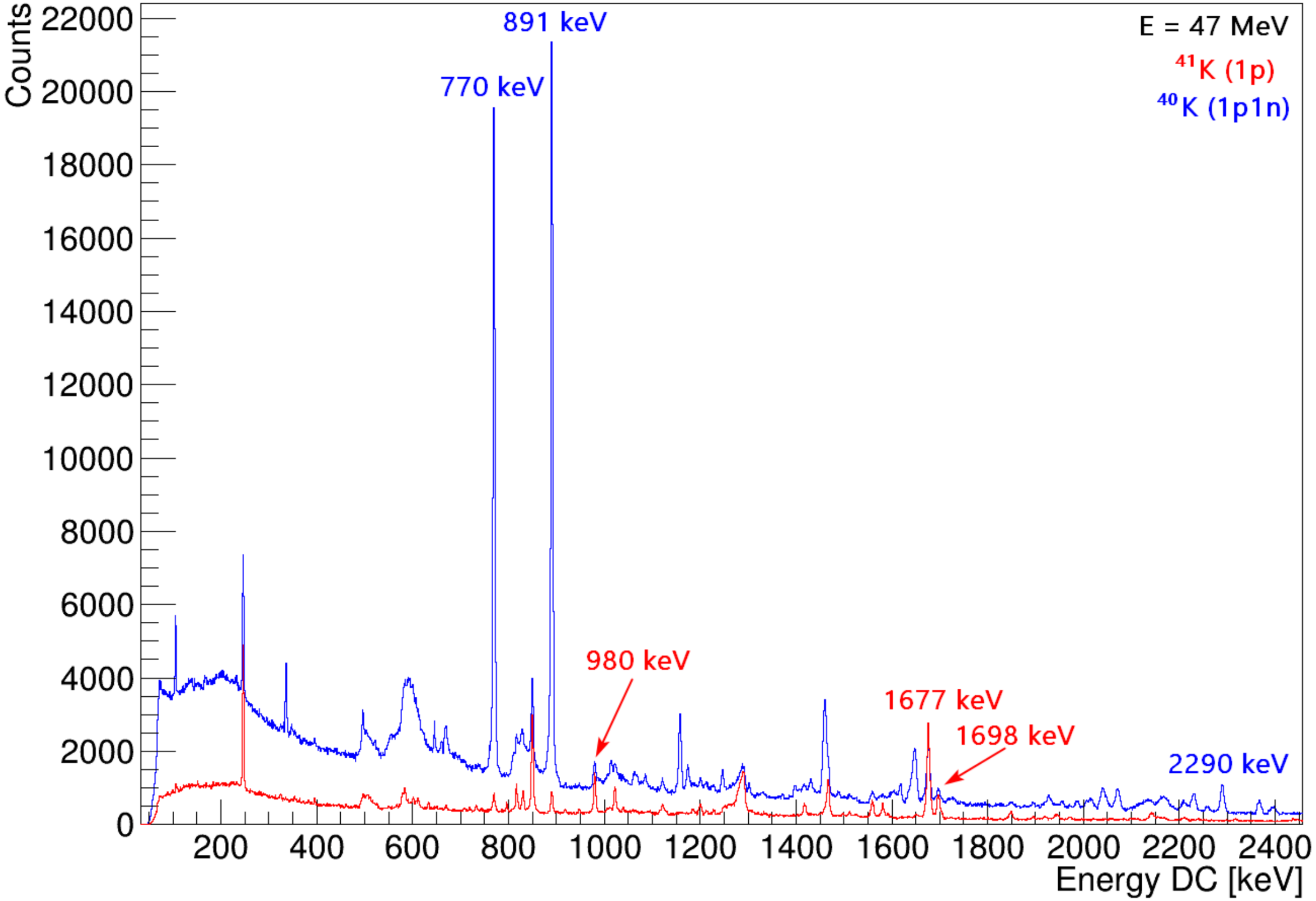


Mirco Del Fabbro PhD Thesis

Mirco Del Fabbro PhD Thesis

Euclides

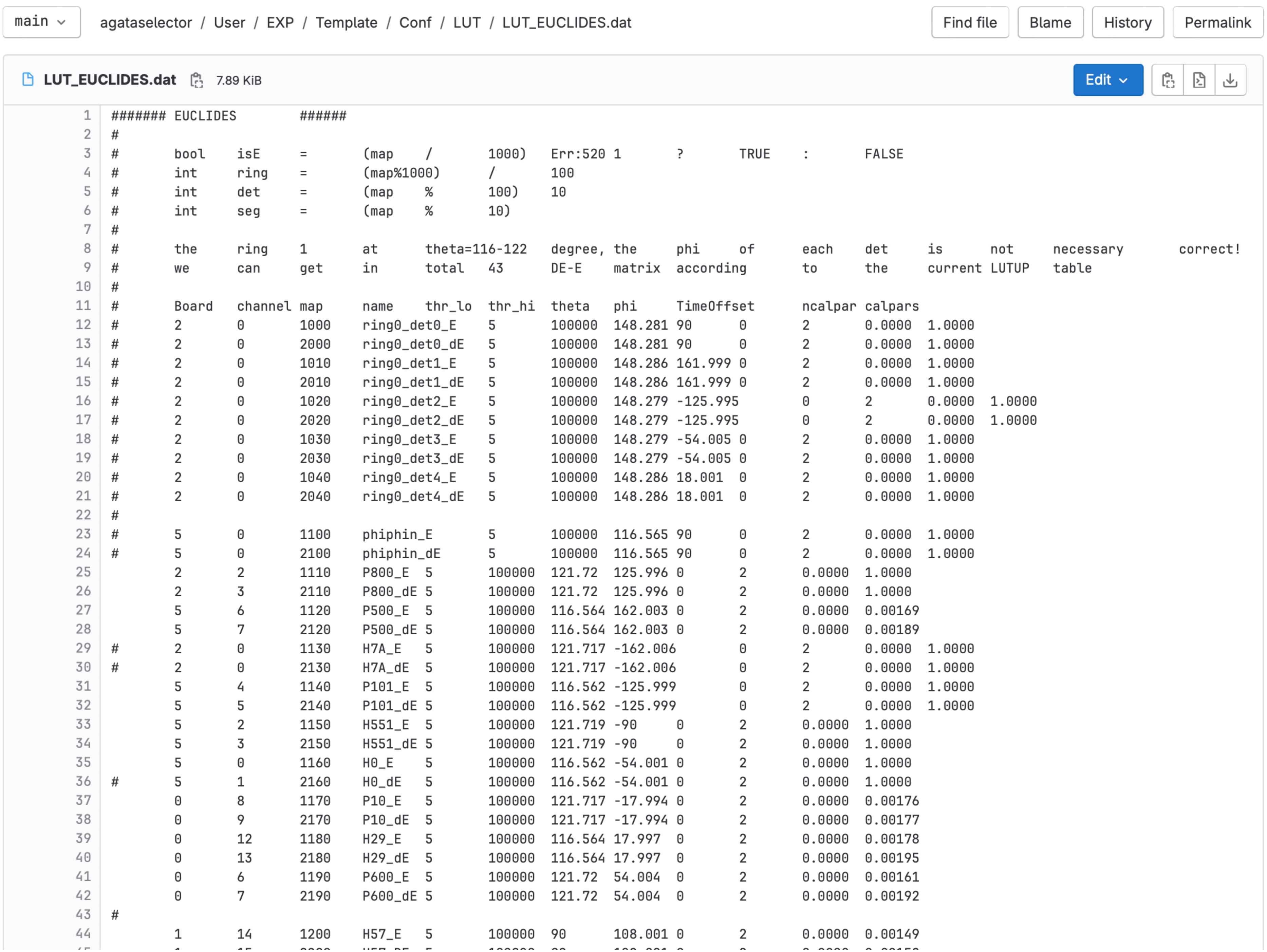




- Constructing a “rough” compound system excitation energy it is possible to discriminate not only protons, deutrons and alphas but also the 1p1n channel from the 1p channel (as an example)

Euclides

- Detector dependent parameters:
  - Theta
  - Phi
- Channel names distinguish E and dE

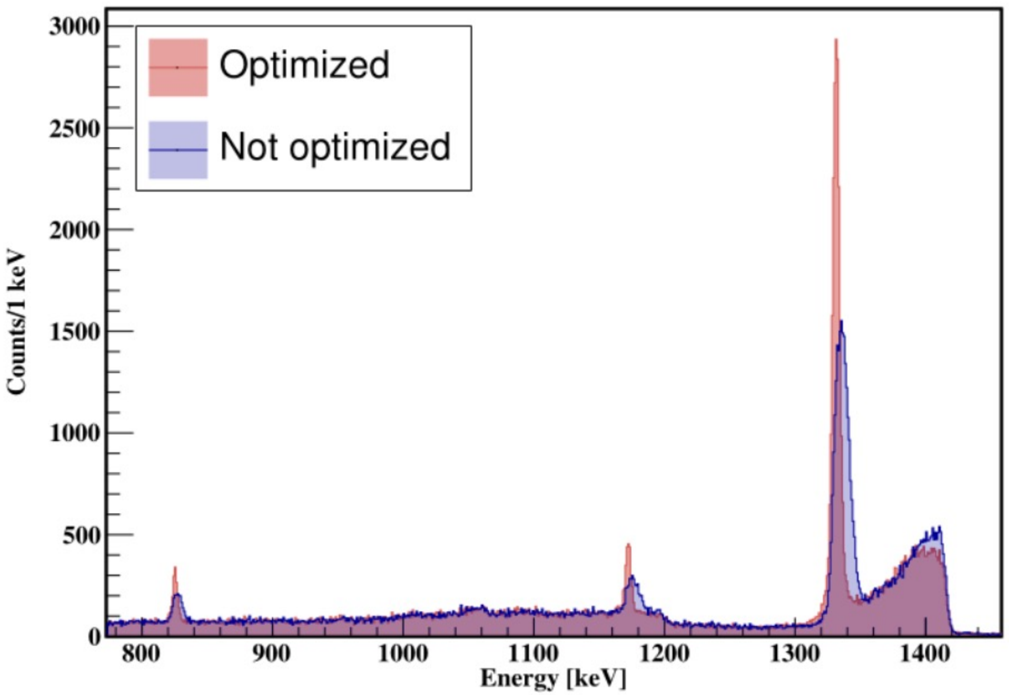


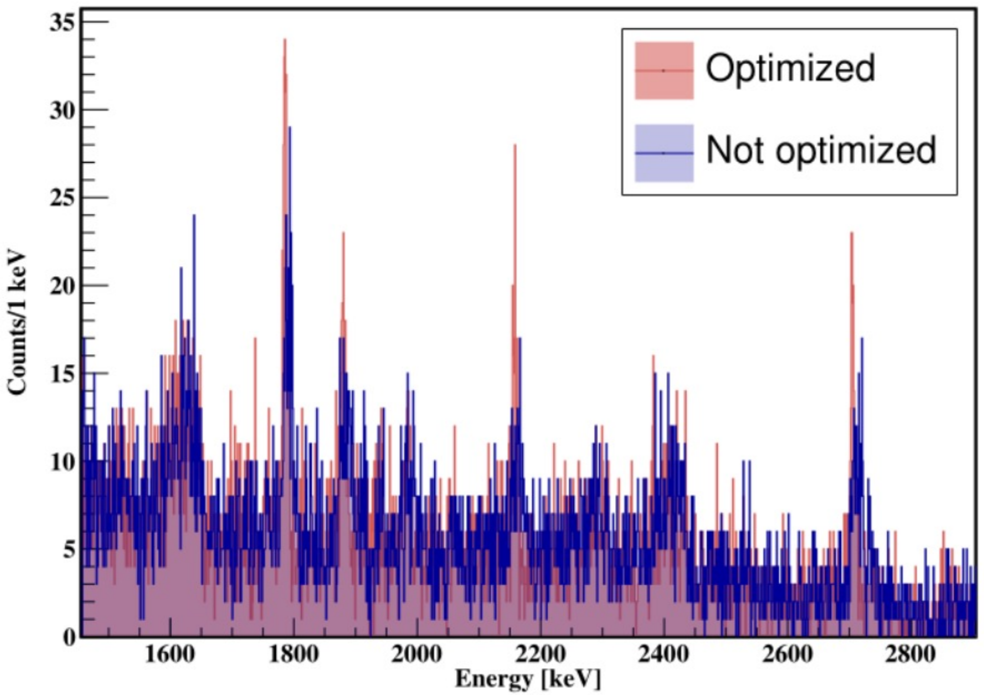
main
agataselector / User / EXP / Template / Conf / LUT / LUT_EUCLIDES.dat
Find file
Blame
History
Permalink
LUT_EUCLIDES.dat
7.89 KiB
Edit

```
####### EUCLIDES        ######
#
#       bool    isE     =       (map    /       1000)   Err:520 1       ?       TRUE    :       FALSE
#       int     ring    =       (map%1000)      /       100
#       int     det     =       (map    %       100)    10
#       int     seg     =       (map    %       10)
#
#       the     ring    1       at      theta=116-122   degree, the     phi     of      each    det     is      not     necessary       correct!
#       we      can     get     in      total   43      DE-E    matrix  according       to      the     current LUTUP   table
#
#       Board   channel map     name    thr_lo  thr_hi  theta   phi     TimeOffset      ncalpar calpars
#       2       0       1000    ring0_det0_E    5       100000  148.281 90      0       2       0.0000  1.0000
#       2       0       2000    ring0_det0_dE   5       100000  148.281 90      0       2       0.0000  1.0000
#       2       0       1010    ring0_det1_E    5       100000  148.286 161.999 0       2       0.0000  1.0000
#       2       0       2010    ring0_det1_dE   5       100000  148.286 161.999 0       2       0.0000  1.0000
#       2       0       1020    ring0_det2_E    5       100000  148.279 -125.995        0       2       0.0000  1.0000
#       2       0       2020    ring0_det2_dE   5       100000  148.279 -125.995        0       2       0.0000  1.0000
#       2       0       1030    ring0_det3_E    5       100000  148.279 -54.005 0       2       0.0000  1.0000
#       2       0       2030    ring0_det3_dE   5       100000  148.279 -54.005 0       2       0.0000  1.0000
#       2       0       1040    ring0_det4_E    5       100000  148.286 18.001  0       2       0.0000  1.0000
#       2       0       2040    ring0_det4_dE   5       100000  148.286 18.001  0       2       0.0000  1.0000
#
#       5       0       1100    phiphin_E       5       100000  116.565 90      0       2       0.0000  1.0000
#       5       0       2100    phiphin_dE      5       100000  116.565 90      0       2       0.0000  1.0000
        2       2       1110    P800_E  5       100000  121.72  125.996 0       2       0.0000  1.0000
        2       3       2110    P800_dE 5       100000  121.72  125.996 0       2       0.0000  1.0000
        5       6       1120    P500_E  5       100000  116.564 162.003 0       2       0.0000  0.00169
        5       7       2120    P500_dE 5       100000  116.564 162.003 0       2       0.0000  0.00189
#       2       0       1130    H7A_E   5       100000  121.717 -162.006        0       2       0.0000  1.0000
#       2       0       2130    H7A_dE  5       100000  121.717 -162.006        0       2       0.0000  1.0000
        5       4       1140    P101_E  5       100000  116.562 -125.999        0       2       0.0000  1.0000
        5       5       2140    P101_dE 5       100000  116.562 -125.999        0       2       0.0000  1.0000
        5       2       1150    H551_E  5       100000  121.719 -90     0       2       0.0000  1.0000
        5       3       2150    H551_dE 5       100000  121.719 -90     0       2       0.0000  1.0000
        5       0       1160    H0_E    5       100000  116.562 -54.001 0       2       0.0000  1.0000
#       5       1       2160    H0_dE   5       100000  116.562 -54.001 0       2       0.0000  1.0000
        0       8       1170    P10_E   5       100000  121.717 -17.994 0       2       0.0000  0.00176
        0       9       2170    P10_dE  5       100000  121.717 -17.994 0       2       0.0000  0.00177
        0       12      1180    H29_E   5       100000  116.564 17.997  0       2       0.0000  0.00178
        0       13      2180    H29_dE  5       100000  116.564 17.997  0       2       0.0000  0.00195
        0       6       1190    P600_E  5       100000  121.72  54.004  0       2       0.0000  0.00161
        0       7       2190    P600_dE 5       100000  121.72  54.004  0       2       0.0000  0.00192
#
        1       14      1200    H57_E   5       100000  90      108.001 0       2       0.0000  0.00149
```

Optimization

The optimization procedure





- Remarkable improvements are possible with the optimization but are experiment dependent.
- The selector contains a procedure to find the optimal parameters by running

  RunSelector - -optimize 2

The optimization procedure

- Checkout the “OPTIMIZER_CONF” parameters in the selector.conf
- Any parameter contained in the selector.conf that is a single float number can be optimized (target thickness, angles, calibration coefficients). Parameters are minimized simultaneously
- This can be used to improve the doppler correction or the q-value/excitation energy to the right position based on the user’s insight
- It is done with a root minimizer (multiple ones are available: Migrad, Simplex, …)

Reducing the data

- Since the selector will run multiple times, one must reduce the data to the one of interest. To do so, it is possible to apply a reduction condition on the data such as:

```
./RunSelector --reduction_cond nbTrack\>0\&\&Z_Nr\>0 262
```

- For every input file, a reduced file will be created in the folder where the original tree is, called red_TREENAME_####.root. The files can be then added to create a single file with the data of interest of few MB that will allow the selector to run with high frequency.
- The optimizer should be run on this file by changing the input file pattern in the selector.conf

Running the optimizer

- The parameter(s) to be optimized can be specified adding one (or multiple lines) in the selector.conf:

```
PARAMETER |detector|par_name|initial_value|min|max|step| ->

 PARAMETER AGATA_CONF ANGLE 31 25 25 1 deg
 PARAMETER TARGET_CONF THICKNESS 1 0.5 2 0.1 mg/cm2
```

- The “line(s)” to optimize can be specified adding one (or multiple lines) in the selector.conf :

```
TRANSITION |folder|spec_name|centroid|sigma|tail|bias ->
TRANSITION /AgataPrisma/Z16/A32 h_DC_ion_32_16 2230 4 0.1 0.5 keV
```

- The fit can be done without a tail, with a left, right, left+right and left+right symmetric tail (parameter TAIL).
- The SIGMA_WEIGHT is a parameter that goes from 0 to 1. It adds weight to the sigma in the cost function.
- The SCAN option will create a root file with the gradient scanned by the parameter, multiple scans can be performed.

Checking the results

- If the Conf/Optimizer/parameters.dat is present it will be created. If it is present the parameters for the fit (mean, sigma, tau, xmin, xmax, integral) will be read from there, otherwise the default ones will be used. You can simply remove it to start from the default ones.
- At each step some lines will be added to the Conf/Optimizer/log.txt with the current value of each parameter as well as the current cost (gradient value)
- At the end an optimized conf will be created in the folder based on the optimial value.
- The results of the fit can be checked step by step with the option:

```
-- debug_canvas
```

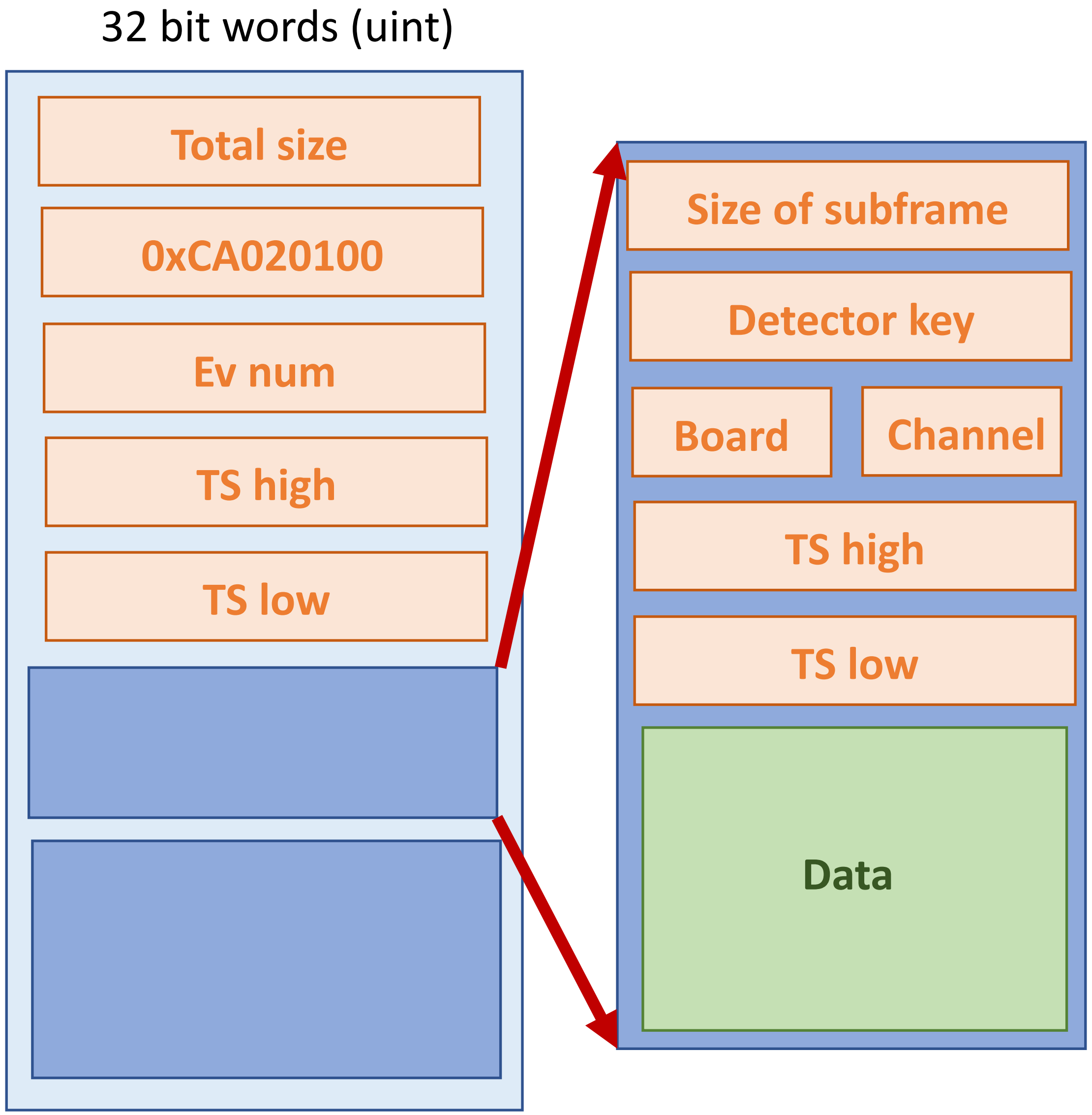
Reading raw data and building ancillary events

Generating the .adf files from the raw data

- The script to read the raw data and build ancillary events is contained in Scripts/AncMerging. To build it, run “cmake -DBUILD_SCRIPTS=On .”
- It builds events within the ancillary within a window
- It handles coincidences also with prisma+other ancillary
- It adds time offset based on the necessary delays
- It applies the correct key for each detector
- The output can then be used by femul to build Agata+ancillary events

ADF composite frame scheme for built events

- The TS is the lowest subframe's TS
- Has a key (0xCA020100) that is associated to the event merger
- Contains other ADF frames
- Can be checked with the **ListFrames** utility



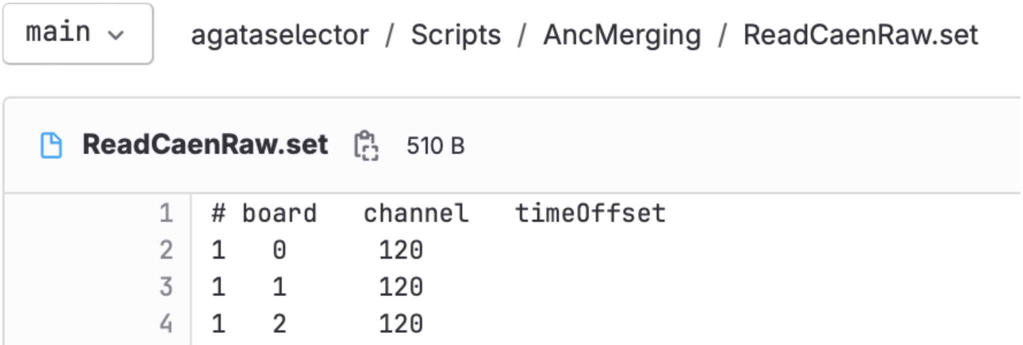
Building ancillary events

- Compile the selector with the option –DBUILD_SCRIPTS=On
- This will create an executable called ReadCaenRaw (note that boost libraries need to be installed)
- Run with

  RunCaenRaw [--labrslave] [--dante MinMultDante] [--prisma InputPrismaFileName] [--global-anc-tsoffset value] [--root file_name] OutputADFFileName

  - labrslave only adds labr events if other detectors are in coincidence
  - dante Nr only adds dante events if Nr channels are present
  - global-anc-tsoffset adds an offset to all timestamps to merge data with Agata
  - You can add a root file for debugging purposes

Building ancillary events

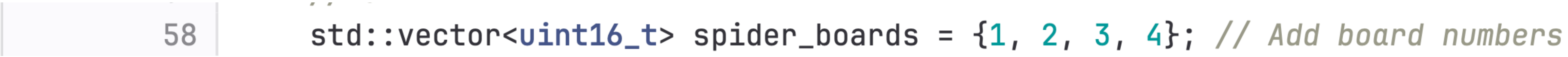


main ⌄ agataselector / Scripts / AncMerging / ReadCaenRaw.set

ReadCaenRaw.set 510 B

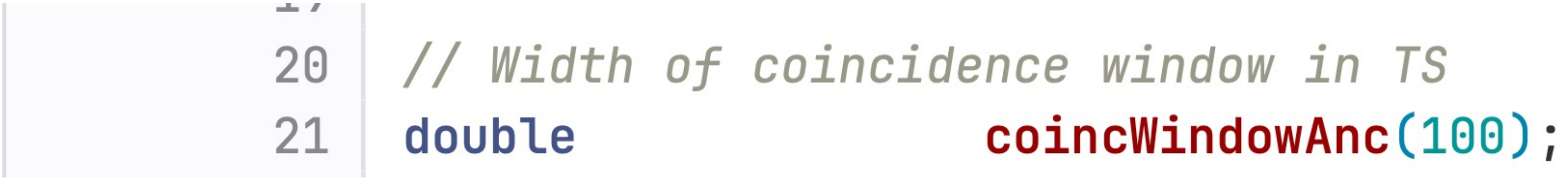
```
# board    channel    timeOffset
1    0       120
1    1       120
1    2       120
```

- The ReadCaenRaw.set file allows to set delays channel by channel, you should check the time-coincidence peak to set it up
- The boards need to be setup in ReadCaenRaw.cxx



```
    std::vector<uint16_t> spider_boards = {1, 2, 3, 4}; // Add board numbers
```

- The same file sets the window in units of 10 ns



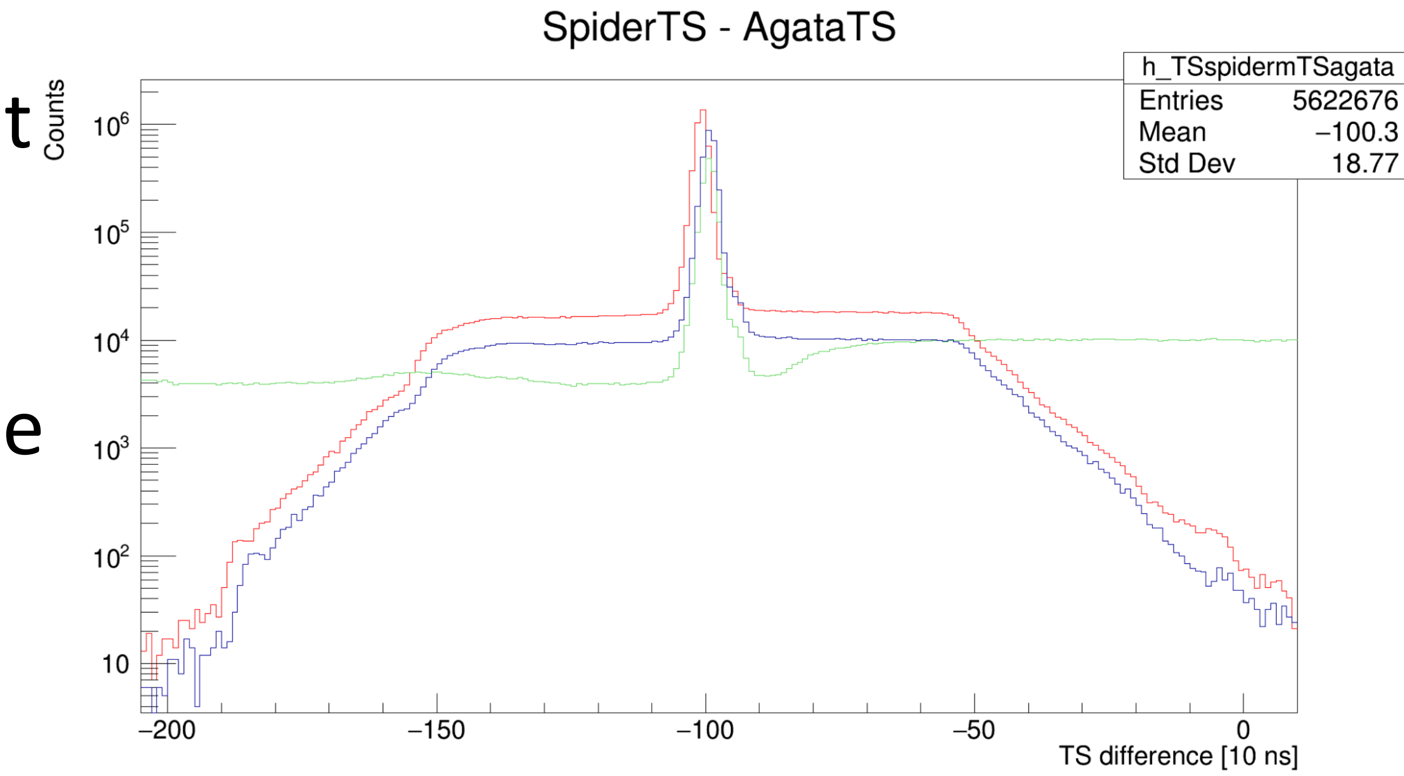
```
// Width of coincidence window in TS
double                   coincWindowAnc(100);
```

Issues with built ancillary events

- There are four possible issues that one might encounter with ancillary events built ”online”:
  1. The DAQ could have stopped building events at some point during the run, in this case one must monitor the coincidence peak over time
  2. The global offset of AGATA might not have been added, therefore there should not be a coincidence peak.
  3. The global offset applied is wrong and needs to be changed
  4. Events built by the DAQ were showing a loss of statistics that can be solved by building events offline
- **These issues are important as they cause a significant loss of statistics**

Issues with built ancillary events

- Ancillary events should be built with a 500 ns window (50 ts units)
- In reality the window should be the same of the one set in the trigger processor
- Red and blue lines represent a good coincidence peak

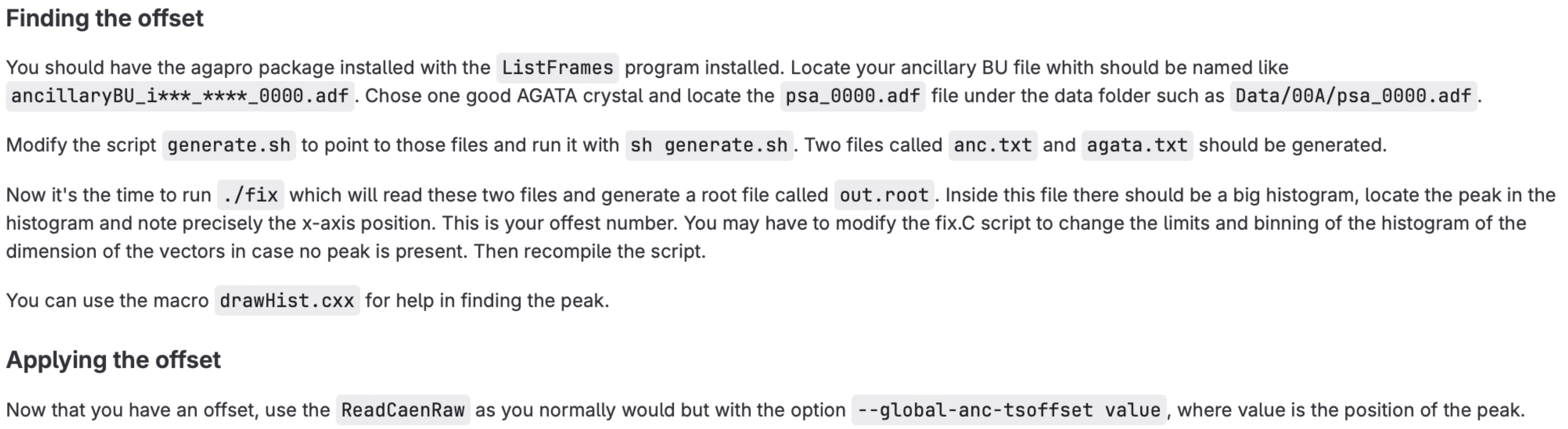


Issues flow chart

Issue	Cause	Solution	
Coincidences stop at some point	Online building problem		Run ReadCaenRaw and then femul
Loss of statistics	Online building problem		
Multiple peaks	Ancillaries or cores not aligned	Align with genconf.py or ReadCaenRaw.set	
Exponential shape	The global time offset is wrong	Find the coincidence peak as explained in Scripts/TimeOffsetPeak	
No peak	There is no global offset		

Finding the coincidence peak

- It is not straightforward and is strongly dependent on the trigger processor settings
- If LaBr are present and acquired as slaves of other ancillaries they can be used to find the right offset
- The offset can be found by randomly correlating all events as explained in Scripts/TimeOffsetFix:



Finding the offset

You should have the agapro package installed with the `ListFrames` program installed. Locate your ancillary BU file whith should be named like `ancillaryBU_i***_****_0000.adf`. Chose one good AGATA crystal and locate the `psa_0000.adf` file under the data folder such as `Data/00A/psa_0000.adf`.

Modify the script `generate.sh` to point to those files and run it with `sh generate.sh`. Two files called `anc.txt` and `agata.txt` should be generated.

Now it's the time to run `./fix` which will read these two files and generate a root file called `out.root`. Inside this file there should be a big histogram, locate the peak in the histogram and note precisely the x-axis position. This is your offest number. You may have to modify the fix.C script to change the limits and binning of the histogram of the dimension of the vectors in case no peak is present. Then recompile the script.

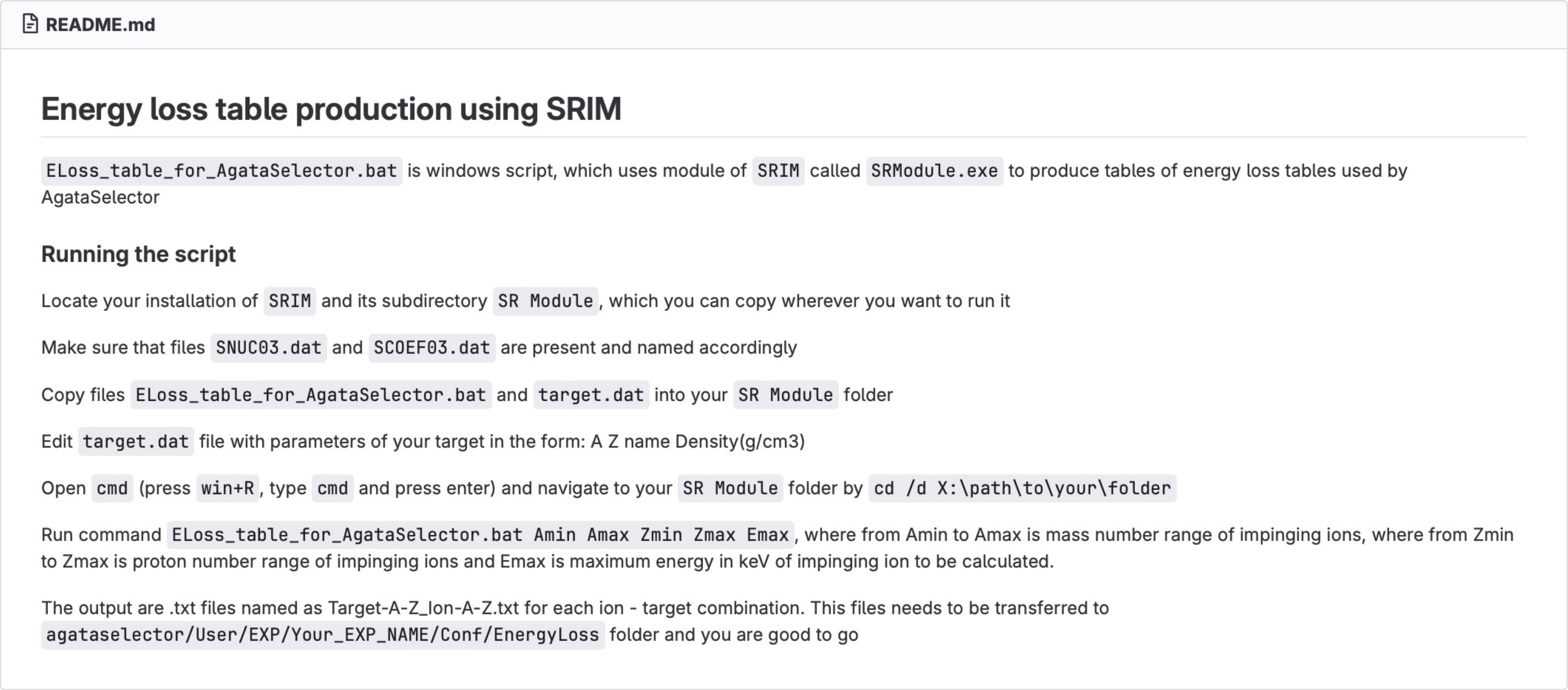
You can use the macro `drawHist.cxx` for help in finding the peak.

Applying the offset

Now that you have an offset, use the `ReadCaenRaw` as you normally would but with the option `--global-anc-tsoffset value`, where value is the position of the peak.

Other useful scripts

Generation of SRIM tables



README.md

Energy loss table production using SRIM

`ELoss_table_for_AgataSelector.bat` is windows script, which uses module of `SRIM` called `SRModule.exe` to produce tables of energy loss tables used by AgataSelector

Running the script

Locate your installation of `SRIM` and its subdirectory `SR Module`, which you can copy wherever you want to run it

Make sure that files `SNUC03.dat` and `SCOEF03.dat` are present and named accordingly

Copy files `ELoss_table_for_AgataSelector.bat` and `target.dat` into your `SR Module` folder

Edit `target.dat` file with parameters of your target in the form: A Z name Density(g/cm3)

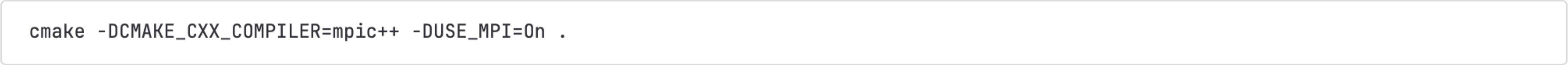
Open `cmd` (press `win+R`, type `cmd` and press enter) and navigate to your `SR Module` folder by `cd /d X:\path\to\your\folder`

Run command `ELoss_table_for_AgataSelector.bat Amin Amax Zmin Zmax Emax`, where from Amin to Amax is mass number range of impinging ions, where from Zmin to Zmax is proton number range of impinging ions and Emax is maximum energy in keV of impinging ion to be calculated.

The output are .txt files named as Target-A-Z_Ion-A-Z.txt for each ion - target combination. This files needs to be transferred to `agataselector/User/EXP/Your_EXP_NAME/Conf/EnergyLoss` folder and you are good to go

MPI

- The selector can be distributed on multiple machines if they have a common file system and boost-mpi installed.
- To enable MPI compile with:



```
cmake -DCMAKE_CXX_COMPILER=mpic++ -DUSE_MPI=On .
```

- It is also necessary to add a file called “hfile” with the ip of each machine that will contribute
- To distribute simply add the option - -distribute when launching the selector.
- Note that the machine where you launch it from will be only assigning jobs and thus will not be under load.

Perspectives and foreseen updates

- S1 (Sauron) and Oscar will be added to the analysis
- Tests should be added for other detectors
- If you find a bug or have a feature please help us, it benefits the community! You can contact us by email or open an issue on baltig: https://baltig.infn.it/gamma/agataselector/-/issues

The end.

Many people are involved in maintaining and developing the selector: Matus Sedlak, Elia Pilotto, Luca Zago, Filippo Angelini, Sara Pigliapoco, ..