PreProcess

Application to be run on rawdata from the MUTOMCA apparatus, consisting of two drift-tube (DT) modules and two CMS superlayers (SL) (a.k.a. Phi).

PreProcess: environment

The same environment provided by the guide is suitable.

PreProcess: downloading and building the code

The source code for the PreProcess is stored in Baltig.

Since the source code of the DTFitter can be downloaded only by the authorized developers, credentials for Baltig are necessary.

The simplest way is using RSA keys, the public key must be uploaded into Baltig and the private key must be saved into ${HOME}/.ssh/id_rsa

Once git client has been correctly configured the commands are:

```
git clone https://baltig.infn.it/muontomography/PreProcess.git
cd PreProcess
```

The commands to build the code are:

```
cmake3 <path-to-source>
make
```


<path-to-source> is the path to the CMakeLists.txt file in PreProcess directory, e.g.: /home/centos/PreProcess

The executable is found in the directory **run/**

PreProcess: running the code

The Preprocess application can be used for different purposes, that can be specified as an argument when launching the program.

occupancy

Display the occupancy of the four detectors and save the histograms in an output root file.

writeNoiseFile

By means of an interactive function, write the noisy channels in an output text file in the following format:

```
[detector] [board ID] [channel ID] [SW layer ID] [SW tube/cell ID]
```

where [detector] could be *DET0*, *DET1*, *SL0*, or *SL1*. This file can be given as an input noise file to the **DTfitter** application.

noise

Display the histograms of the raw times read from the noisy channels written in the noise file and save the histograms in an output root file.

t0

Calculate t0 of each event and save it in an output text file. The output file has the following format:

```
[event ID] [t0 info] [t0 DET 0 in ns] [t0 DET 1 in ns]
```

where [t0 info] is set to *ext* if a mean timer is present in that event, otherwise to *min* if the minimum of the drift times of that event is used. This file can be given as an input t0 file to the **DTfitter** application. Also, the histograms related to the mean-timer formulas are shown and save in an output root file.

monitor

Show an event-per-event display of the hits in the four detector, in the local reference frame of the detectors or in a global reference frame centered with the cask.

Preliminary settings: the configuration file

Copy the template configuration file **config-example.ini** from the **PreProcess/utils/** directory to the **PreProcess/run/**directory and rename it **config.ini**. Now you can modify the **config.ini** file in your **PreProcess/run/** directory and set all the parameters needed to run the code.

- Input files:

Variable	Type	Description
rawDirName	*string*	path of the input file directory
rawFileNameTubes_1	*string*	name of the input file for DET 0 (file 1_)
rawFileNamePhi_3	*string*	name of the input file for SL 0 (file 3_)
rawFileNameTubes_2	*string*	name of the input file for DET 1 (file 2_)
rawFileNamePhi_4	*string*	name of the input file for SL 1 (file 4_)
maxEventNumber	*int*	maximum number of events to be read

- Output files:

Variable	Type	Description
outputDirName	*string*	path of the output file directory
noiseFileName	*string*	name of the output file with noisy channels
t0FileName	*string*	name of the output file with t0
rootFileName	*string*	name of the output root file with histograms

- Debugging:

Variable	Type	Description
debug	*bool*	set to 1 to dump debug messages

- Monitor: these variables are only relevant when running the *Monitor*:

Variable	Type	Description
global	*bool*	Display hits in the gloabl (1) or local (0) reference frame
start	*uint*	First event number to be displayed
det0Rot	*float*	DET 0 rotation angle wrt an arbitrary reference [deg]
det1Rot	*float*	DET 1 rotation angle wrt an arbitrary reference [deg]
det0Dist	*float*	distance of DET 0 from Castor center [cm]
det1Dist	*float*	distance of DET 1 from Castor center [cm]
det0Z	*float*	z coordinate of DET 0 center [cm]
det1Z	*float*	z coordinate of DET 1 center [cm]

How to run

```
cd run
./runPreProcess <analysis>
```

<analysis> may be: occupancy, writeNoiseFile, noise, t0, monitor.