Ecalpro:
A tutorial for future inter-calibrators

Luca Perniè

11.03.15

Instructions

- Consult the twiki:
  https://twiki.cern.ch/twiki/bin/viewauth/CMS/EcalPro

- Very simple instructions:

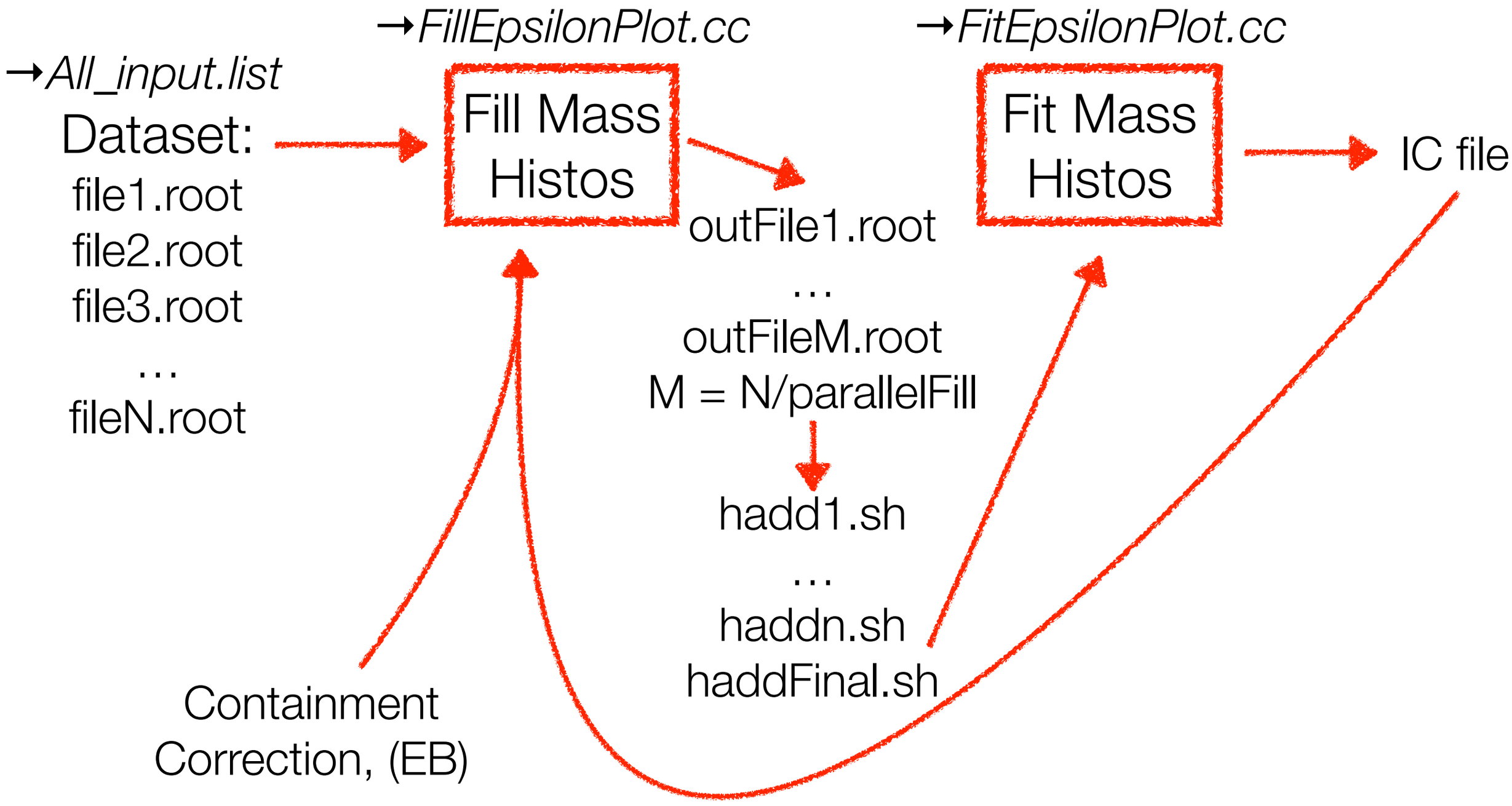
```
mkdir -d ECALpro
cd ECALpro
cmsrel CMSSW_7_4_X
cd CMSSW_7_4_X/src
cmsenv
git clone git@github.com:lpernie/ECALpro.git CalibCode
scram b -j 12
cd CalibCode /submit
chmod +x submitCalibration.py resubmitCalibration.py
```

→ The only command you really need

git clone https://github.com/lpernie/ECALpro CalibCode

The Process

- Inter-calibration is an iterative process:



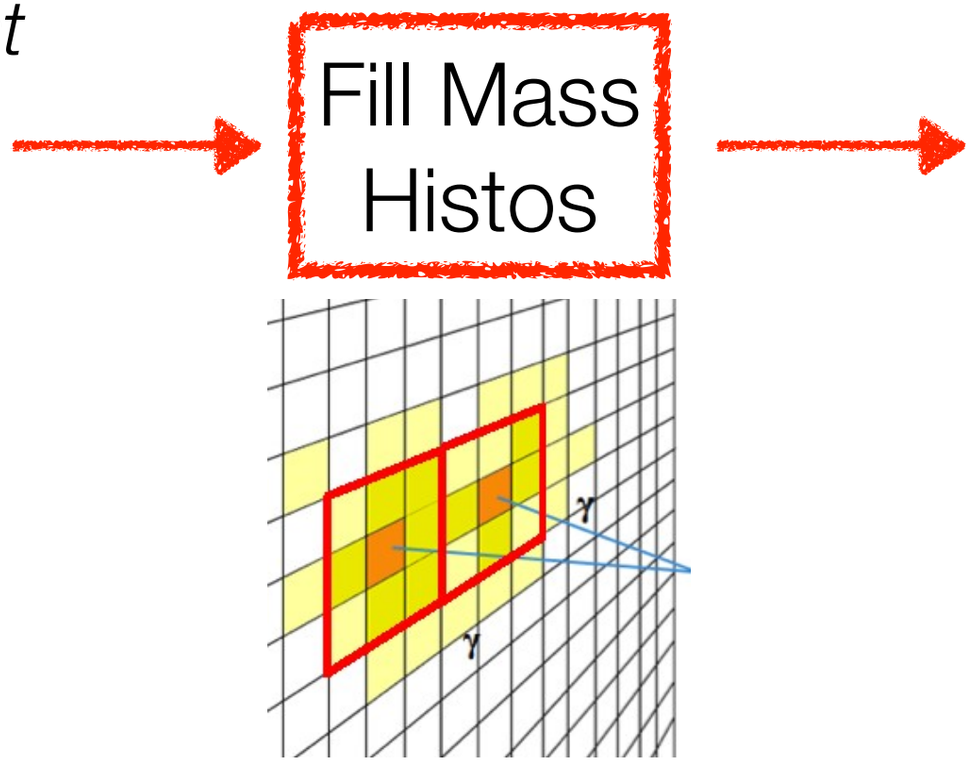
FillEpsilonPlot

❖It produces an Inv. Mass histos for each EB/EE crystal

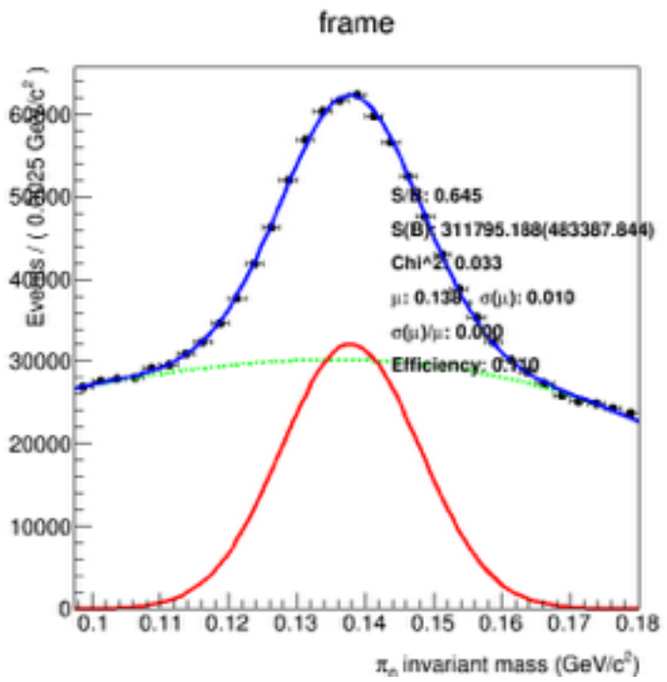
→*FillEpsilonPlot.cc*

→*All_input.list*

Dataset:
file1.root
file2.root
file3.root
…
fileN.root



From the RecHits List:
→ Create Seed
→ Create 3x3 Clusters*
→ Apply prev. ICs
→ Create combinatorial π^{0*}
→ Apply Cont. Corr. (CC)
→ Filling histograms



❖A maximum of 18 crystals contribute to a single mass value:
A mass value is given to all the crystals that contribute to such π^0, but weighted for the energy fraction they carried

*Selection is applied

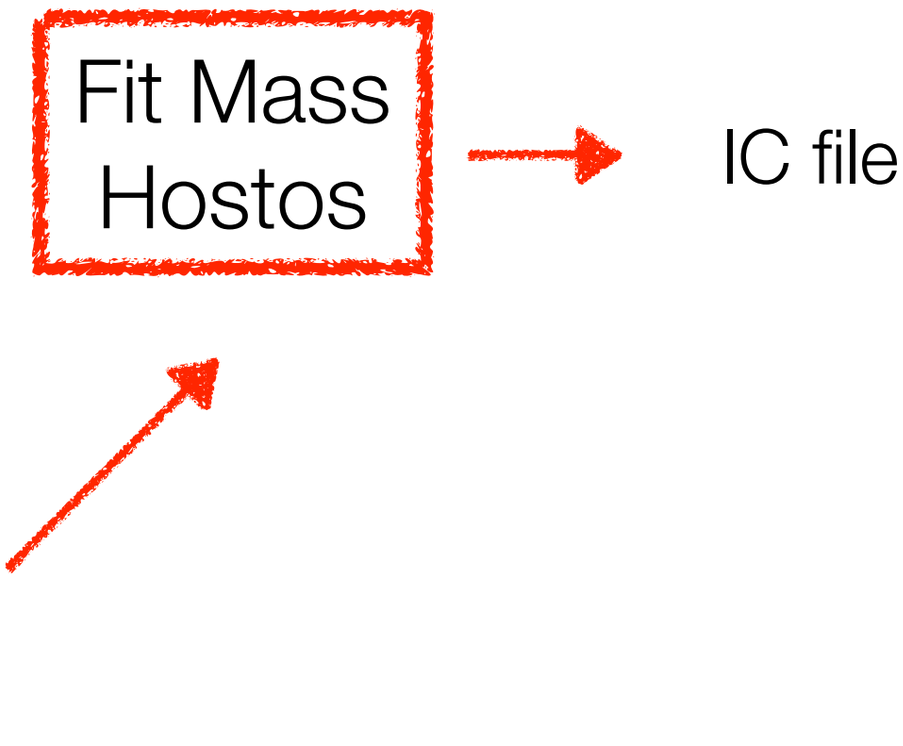
FitEpsilonPlot

- The Fit is done for each histogram and it provide the relative IC

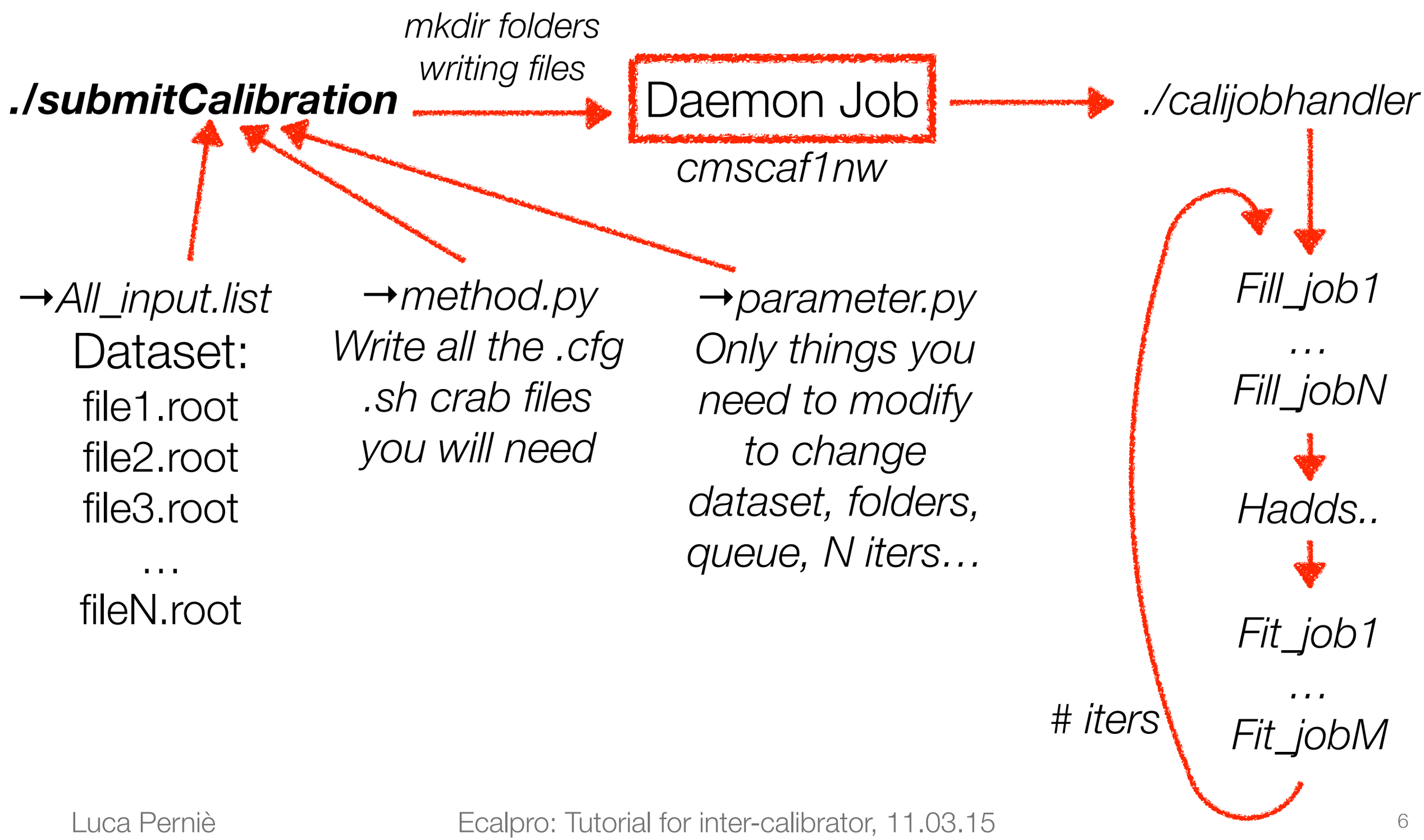
→*FitEpsilonPlot.cc*

Background has a very different shape depending on the conditions, the geometric region, the selection etc…
Needs for a stable Fit:

- Iterative
- Chebyshev + Gauss



The tools (batch submission)



What you get from git-hub

- After getting the code:

CalibCode →
- CalibTools
- DumpCaloGeometry
- EgammaObjects
- FillEpsilonPlot → Fill the Mass histos
- FitEpsilonPlot → Fit the Mass hosts
- GBRTrain → MVA Cont. Corr
- ReadGT → Now we read it from cmssw
- json → WORKING Directory
- submit ↓

submitCalibration.py → THE command. Launch the demon, write files and mkdir.
→ parameter.py → parameters (the main file you will modify)
→ method.py → Allo to write the cfg/sh/crab files your demon will use
calibJobHandler.py → The demon that will do all the work (fill hadd fit fill add…)
resubmitCalibration.py → To resubmit an IC if it break at iter N.

The tools (crab submission)

mkdir folders
writing files

./submitCalibration → *Instructions:*

→*All_input.list*
Dataset:
file1.root
file2.root
file3.root
…
fileN.root

→*method.py*
Write all the .cfg
.sh crab files
you will need

→*parameter.py*
Only things you
need to modify
to change
dataset, folders,
queue, N iters…

They will tell you:

- How to find/run the Crab files
- How to launch hadd and the fit after CRAB

Extras:

- Calibcode/submit/Utilities
- Calibcode/submit/AfterCalibration/

DeadXtals → Rem. dead xtal (OLD)
Optimization → To optimize the sel.
TestConvergence → Are the IC converging?
WorkOnIC → Produce the abs. IC

ThePerfectBashScript.sh

It will do useful things for you, as remove root file not need after the hadd etc…

Purify_List.py

Given a file.list of input file and a jsonfile it will remove useless files to have a reduced list with mainly goo events

To do list

- Run on DIGI
- Finalize the off-line optimization of the selection
- Make the code available to run on different T2
- Perform a full inter-calibration on MC

Backup