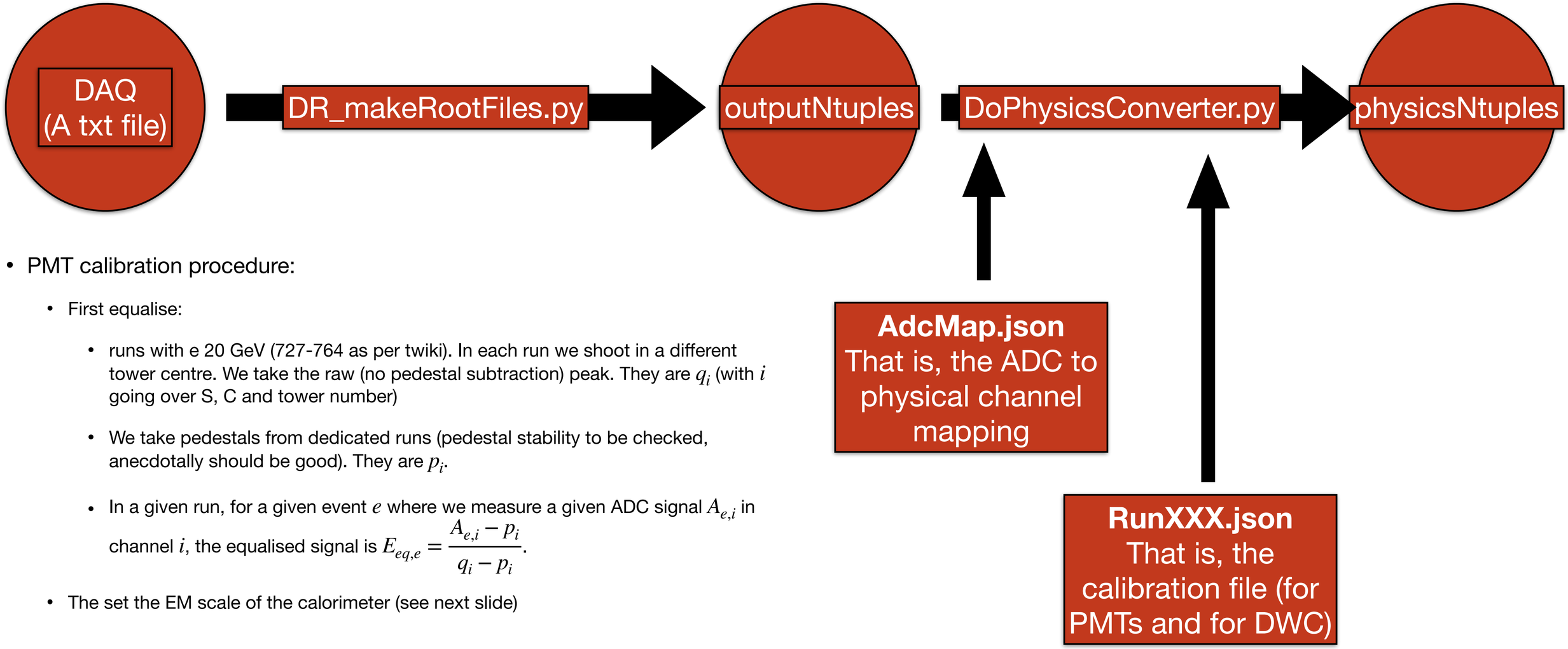
ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

TB2024 - A first look

Iacopo Vivarelli - Internal HiDRa meeting - 11/9/2024 - all results are EARLY DAYS - do not use for public talks

Reconstruction chain



- PMT calibration procedure:
  - First equalise:
    - runs with e 20 GeV (727-764 as per twiki). In each run we shoot in a different tower centre. We take the raw (no pedestal subtraction) peak. They are q_i (with i going over S, C and tower number)
    - We take pedestals from dedicated runs (pedestal stability to be checked, anecdotally should be good). They are p_i.
    - In a given run, for a given event e where we measure a given ADC signal $A_{e,i}$ in channel i, the equalised signal is $E_{eq,e} = \frac{A_{e,i} - p_i}{q_i - p_i}$.
  - The set the EM scale of the calorimeter (see next slide)

PMT calibration procedure

- PMT calibration procedure:
  - First equalise:
    - runs with e 20 GeV (727-764 as per twiki). In each run we shoot in a different tower centre. We take the raw (no pedestal subtraction) peak. They are q_i (with i going over S, C and tower number)
    - We take pedestals from dedicated runs (pedestal stability to be checked, anecdotally should be good). They are p_i.
    - In a given run, for a given event e where we measure a given ADC signal $A_{e,i}$ in channel i, the equalised signal is $E_{eq,e} = \frac{A_{e,i} - p_i}{q_i - p_i}$.
  - The set the EM scale of the calorimeter scale (using run 746 - e 20 GeV in T00):
    - Separately for S and C, compute the energy in the central part of the calorimeter ($E_{R0} + E_{R1} + E_{R2}$, should have negligible electron contamination)
    - Define a calibration factor as $\mathrm{adcToGeV} = 20\ \mathrm{GeV}/(\mathrm{E_{R0}} + \mathrm{E_{R1}} + \mathrm{E_{R2}})$.
  - Finally, take into account shifts in HV:
    - Build a list of runs taken with lower voltages in T00.
    - Compute, separately for S and C, an additional factor as $E_{T00\ in\ run\ 746}/E_{T00\ in\ run\ 766}$

The three steps in the code

- The adcToGeV factors

```
506  void Event::calibratePMT(PMTCalibration& pmtcalibration, EventOut* evout, Long64_t entry){
507
508    static float adcToPhysS = 20./1.2617; // Second attempt to bring the calorimeter to the electromagnetic scale. Number obtained using second equalisation cycle (20 GeV electrons)
509
510    static float adcToPhysC = 20./1.3396; // Second attempt to bring the calorimeter to the electromagnetic scale. Number obtained using second equalisation cycle (20 GeV electrons)
```

- Taking into account the different HVs

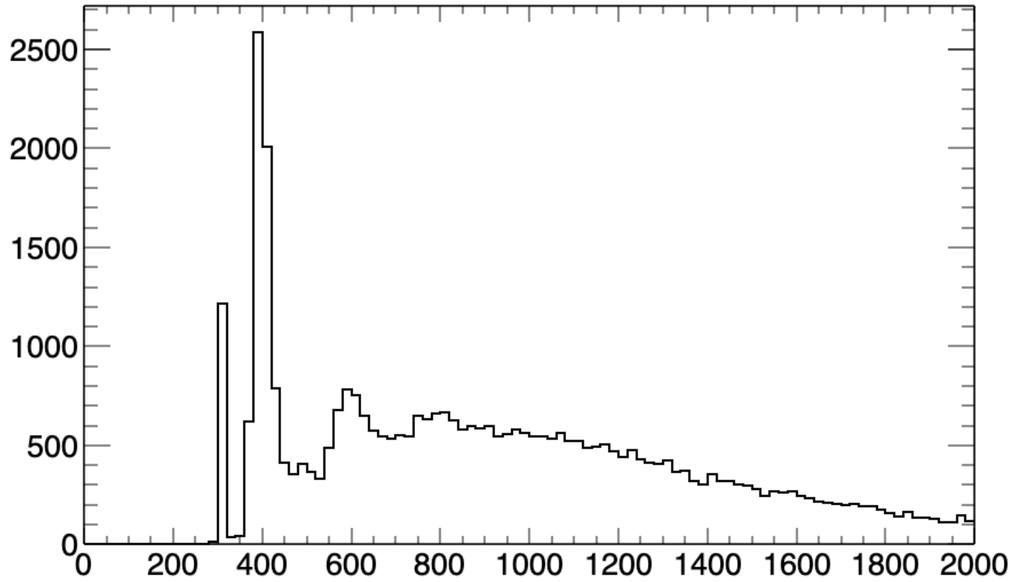
```
512
513    /* These numbers are used to take into account the change in HV in tower 0 in some runs*/
514    static float correctT00_S = -1.;
515    static float correctT00_C = -1.;
516
517    if (correctT00_S < 0){ // check that we haven't yet tested whether this run should be corrected or not
518      correctT00_S = 1.;
519      correctT00_C = 1.;
520      std::vector<unsigned int> runs_tobecorrected = {766,767,772,774,775,776,777,778,779,780,781,782,783,784,786,792,793,794,796,797};
521      for (unsigned int run_tc : runs_tobecorrected){
522        if (run_number == run_tc){
523          correctT00_S = 15.37/5.75; // Ratio of the peak position in run 746 and in run 766 (766 before applying this calibration
524          correctT00_C = 14.9/2.88; // Ratio of the peak position in run 746 and in run 766 (766 before applying this calibration
525          std::cout << "This run was taken with the new HV. The response in T00 will be rescaled" << std::endl;
526          std::cout << "TS00 response will be multiplied by " << correctT00_S << std::endl;
527          std::cout << "TC00 response will be multiplied by " << correctT00_C << std::endl;
528        }
529      }
530    }
531
```

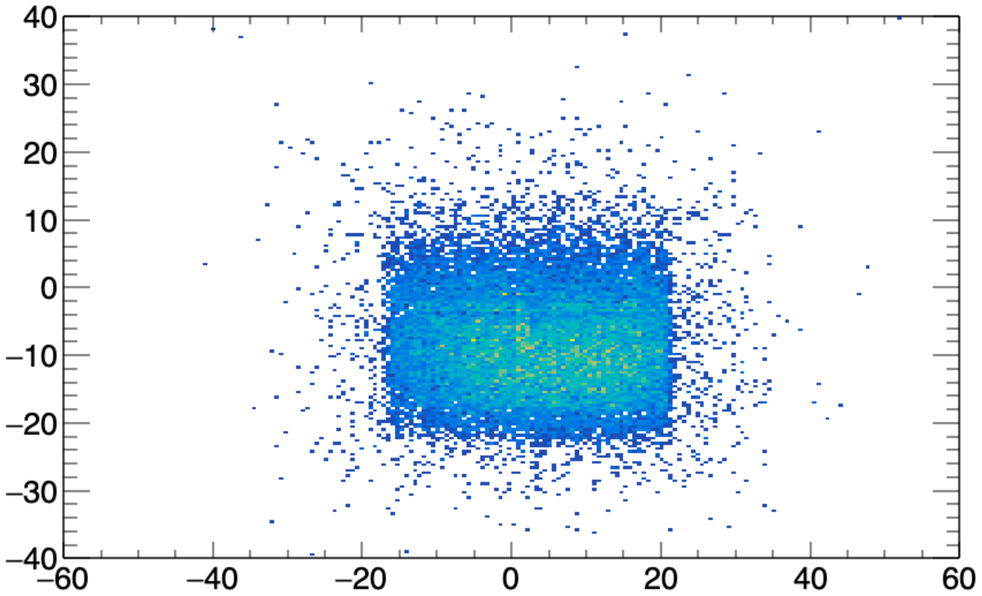
Finally, the calibration

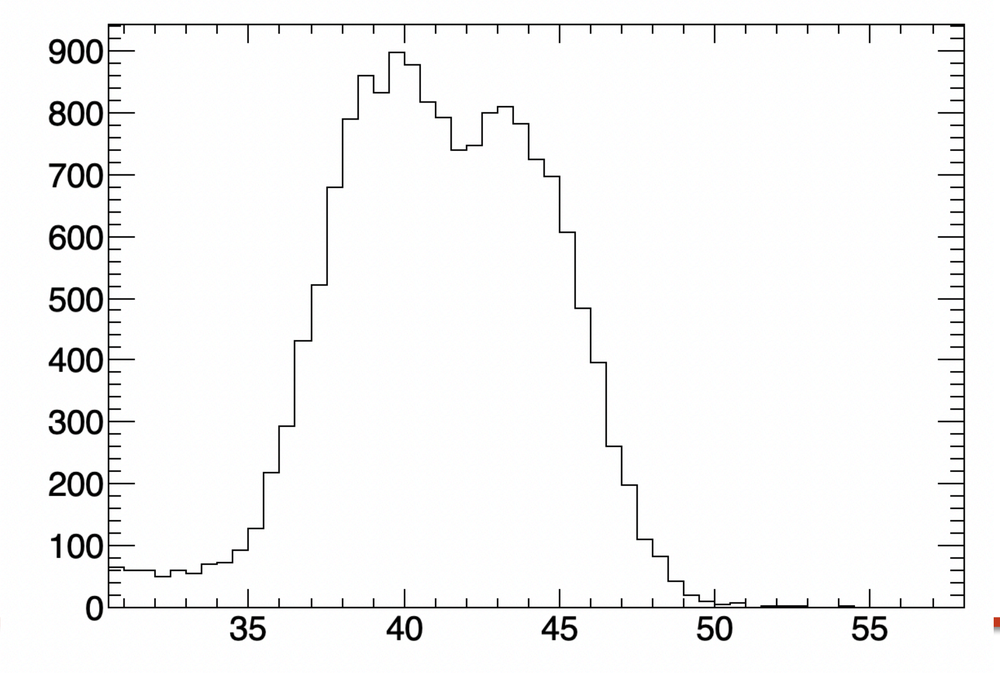
```
533       if (entry < 0){ // Then use the pedestals and peaks from file
534         for (auto it = this->channel.begin(); it != this->channel.end(); ++it){
535           std::string key = it->first;
536
537           // check if the key is available in the PMTcalibration map. If it isn't, this is an ancillary. Skip for the moment
538
539           if (pmtcalibration.PMTped.find(key) != pmtcalibration.PMTped.end())
540             {
541               static float adcToPhys;
542               if (key.find("TS") != std::string::npos){
543                 adcToPhys = adcToPhysS;
544               } else if (key.find("TC") != std::string::npos){
545                 adcToPhys = adcToPhysC;
546               }
547               if (key == "TS00"){
548                 this->channel_calibrated[key] = adcToPhys*correctT00_S*((float(this->channel[key])) - pmtcalibration.PMTped[key])/(pmtcalibration.PMTpk[key] - pmtcalibration.PMTped[key]);
549               } else if (key == "TC00"){
550                 this->channel_calibrated[key] = adcToPhys*correctT00_C*((float(this->channel[key])) - pmtcalibration.PMTped[key])/(pmtcalibration.PMTpk[key] - pmtcalibration.PMTped[key]);
551               } else {
552                 this->channel_calibrated[key] = adcToPhys*((float(this->channel[key])) - pmtcalibration.PMTped[key])/(pmtcalibration.PMTpk[key] - pmtcalibration.PMTped[key]);
553               }
554             }
555         }
556
```

What can go wrong?

- The peak position at the different stages can depend quite a bit on the electron selection.
- Everywhere we used PShower > 550 (which is fine).
- DWC are a bit more critical, need to study their impact a bit better.

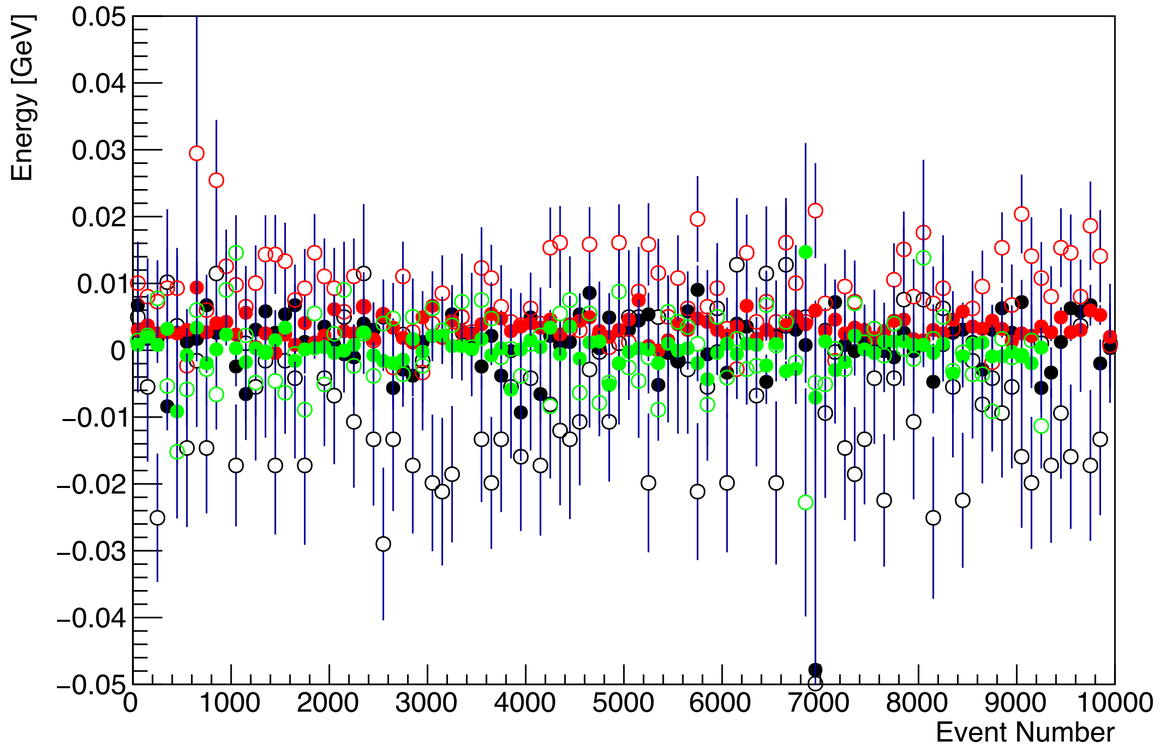






Pedestal stability

- Took Run 782 (in black, new HV), Run 713 (in red, old HV), Run 770 (in green, old HV). Plotted pedestal value Vs EventID. Full marker is TS00, empty marker is TC00.



Fibre attenuation length

- Used runs 804-816 to c compute the fibre attenuation length
  - Work done before last calibration, ignore absolute scales.
  - The energy is R0 + R1 + R2.
  - Bad result
  - But why essentially the same result for C and S?????