SDHCAL Digitizer

- Current version is working with ilcsoft v01-19-04
  - ~~Not anymore with ilcsoft v01-19-05-pre03 (bug in dd4hep::BitField64 class)~~
    - ~~need temporary fix by Rémi until this issue is resolved~~

- Changes since last pull request :
  - Separate the digitizer into 2 processors :
    - 1st digitizer only does the 'digitization' itself → just apply the thresholds
    - 2nd digitizer transforms the thresholds (1,2,3) into energy values for the linear formula : $E = \alpha*N_1 + \beta*N_2 + \gamma*N_3$
    - By proceeding this way we don't have to relaunch the digitization each time (which is quite slow) just to test an other energy calibration
  - Removed the mokka/gear part as requested
  - + some cleaning

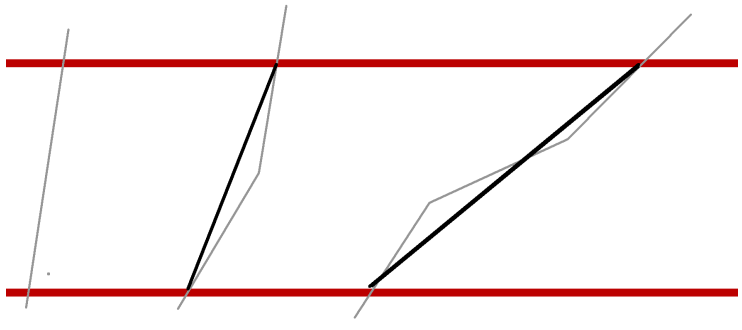
- We will do some checks with the v01-19-05 version and then we can pull request the digitizer and the updated steering file

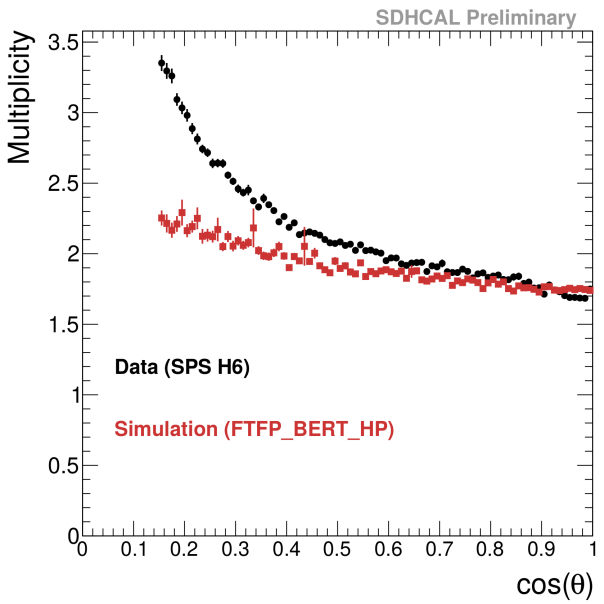
SDHCAL Digitizer – angle correction

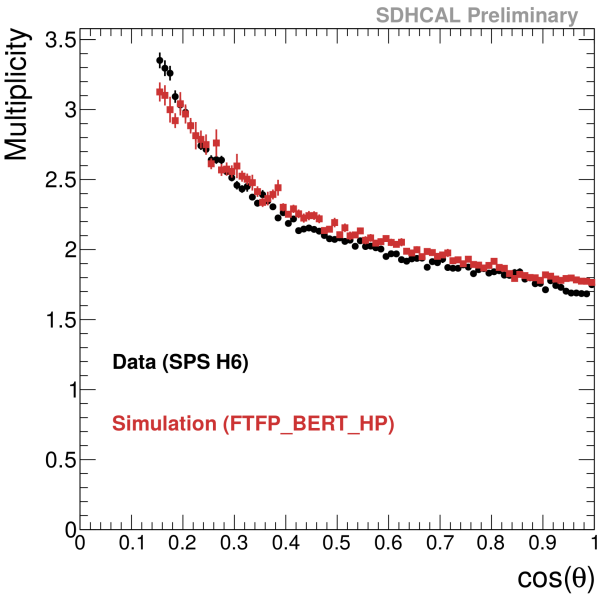
- We have our own standalone GEANT4 simulation to simulate the SDHCAL prototype
- In this simulation we have an extra procedure in the digitization process
- We apply some correction to the induced charge depending on the step angle
  - Greatly improves the multiplicity reproduction

- For doing the same procedure in ILD simulation we need some extra informations :
  - In the simulation itself we link the steps belonging to the same particle together and we register it as a single MCContribution with
    - position = (leavingPos + entrancePos)/2
    - length = (leavingPos – entrancePos).mag()
  - (I don't know what is done in DD4hep...)

  - Then to apply the correction we use the length of this MCContribution







SDHCAL Digitizer – angle correction

- We have no way to store this length in the current SimCalorimeterHit format, so we write it in a separate LCGenericObject collection
- It would be nice to have some space in the SimCalorimeterHit to write the length of each MCContribution
- Without the extra step length information, digitizer angular correction can't be implemented in the full ILD simulation/reconstruction.