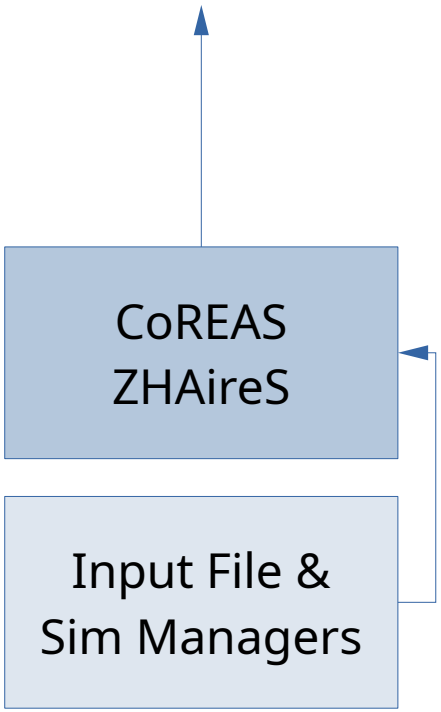
How is the simulation chain ia organized

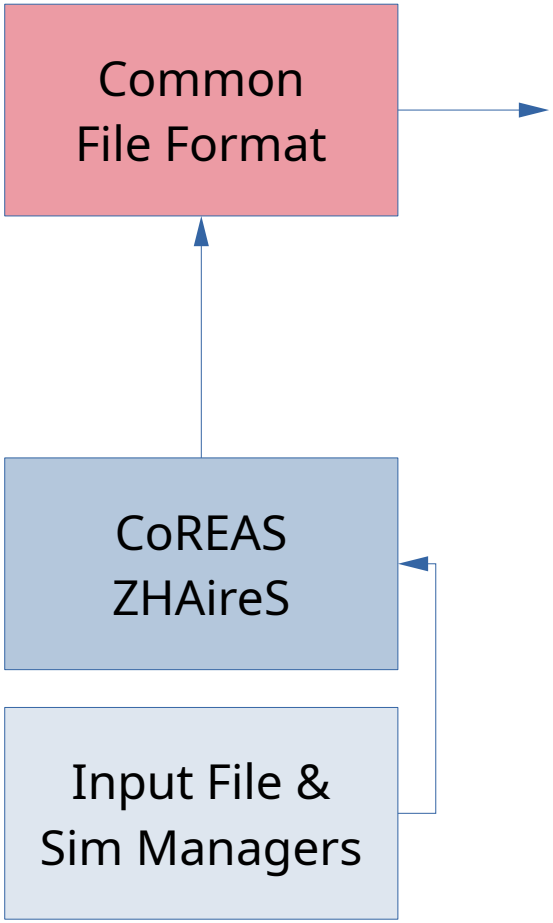
M. Tueros
Instituto de Fisica La Plata, Argentina
CONICET/UNLP

An opportunity to identify where you can contribute

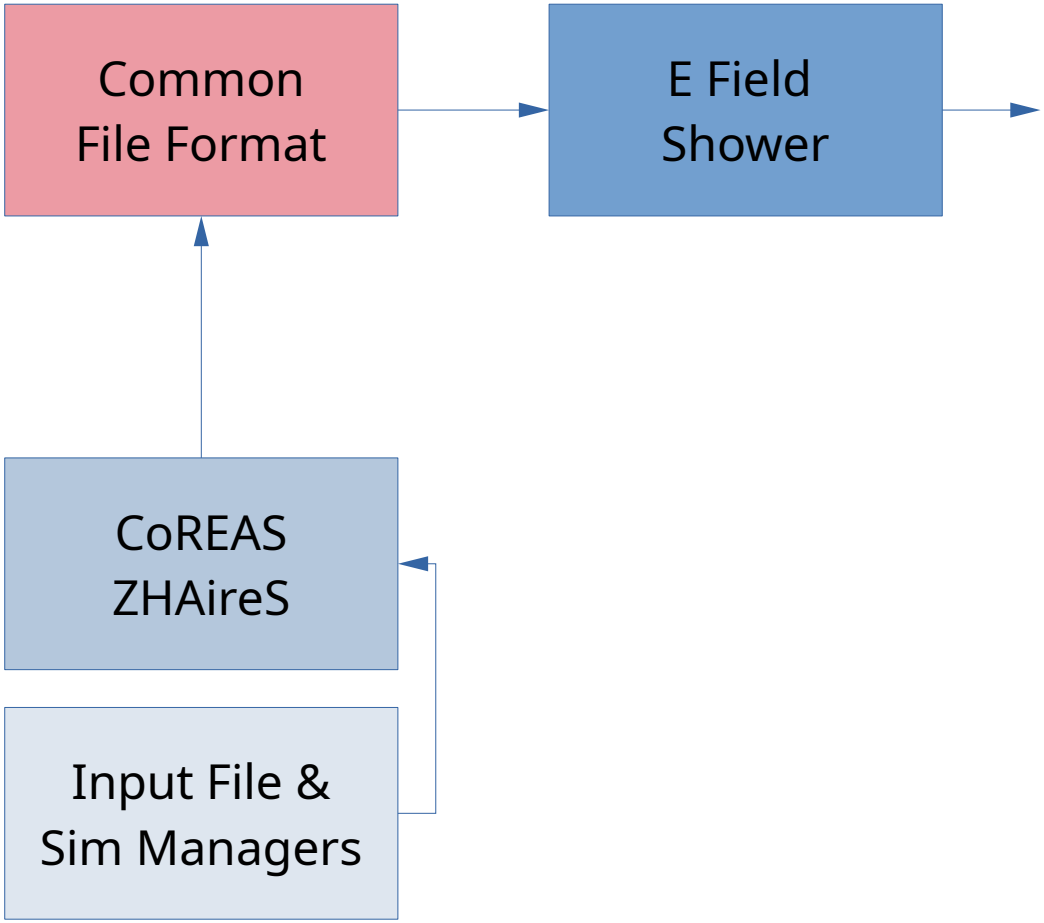
How is the simulation pipe organized



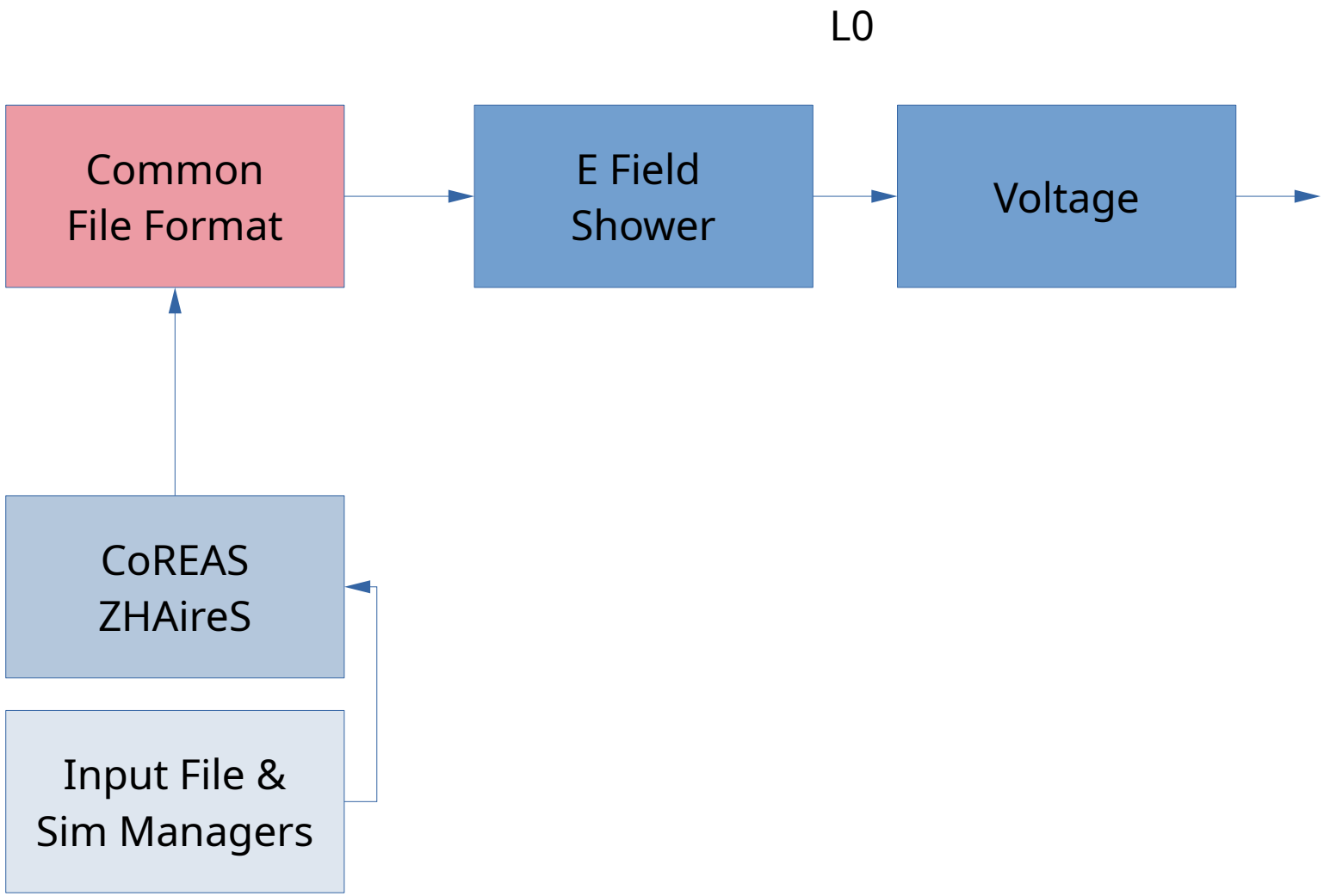
How is the simulation pipe organized



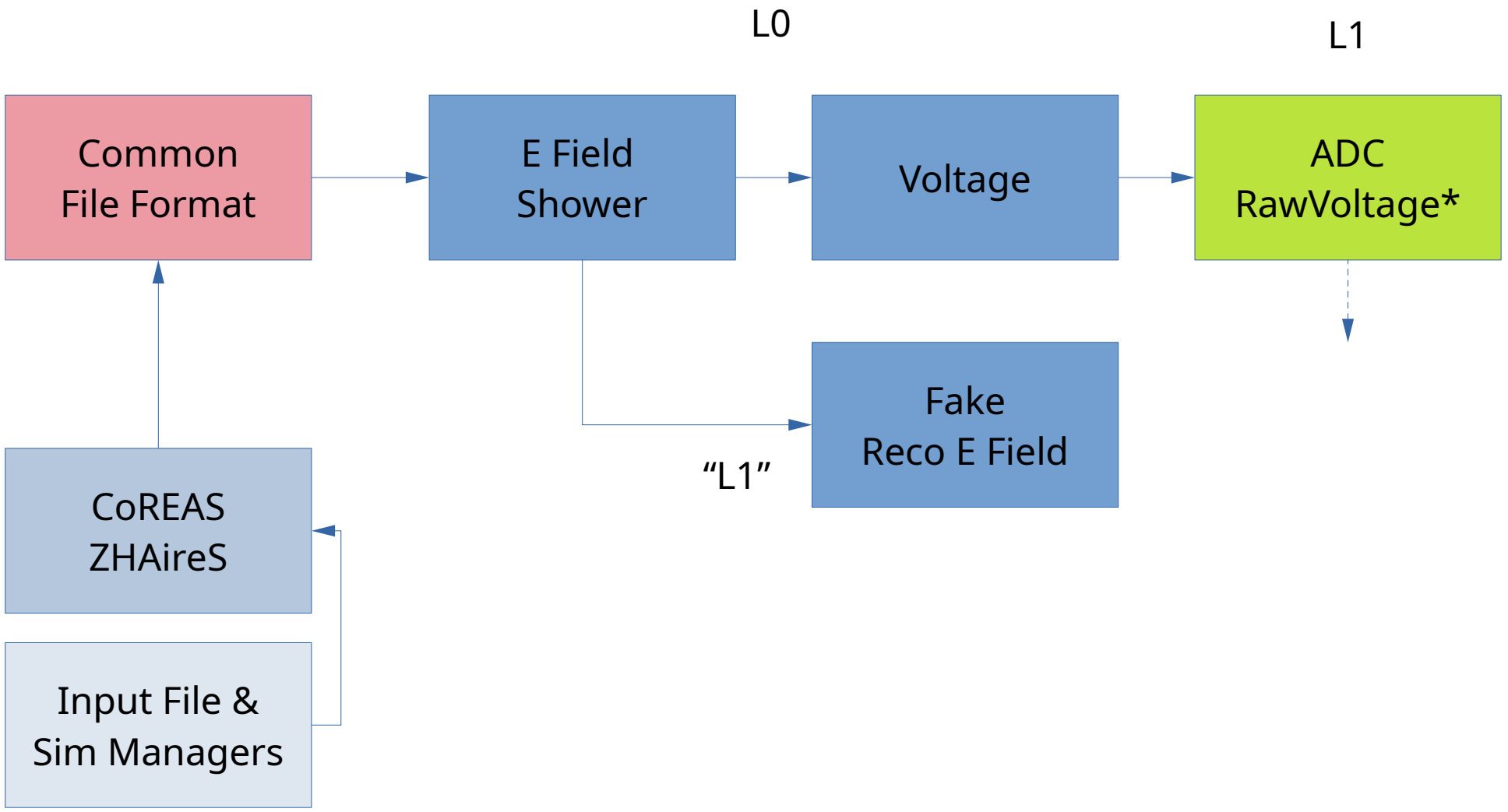
How is the simulation pipe organized



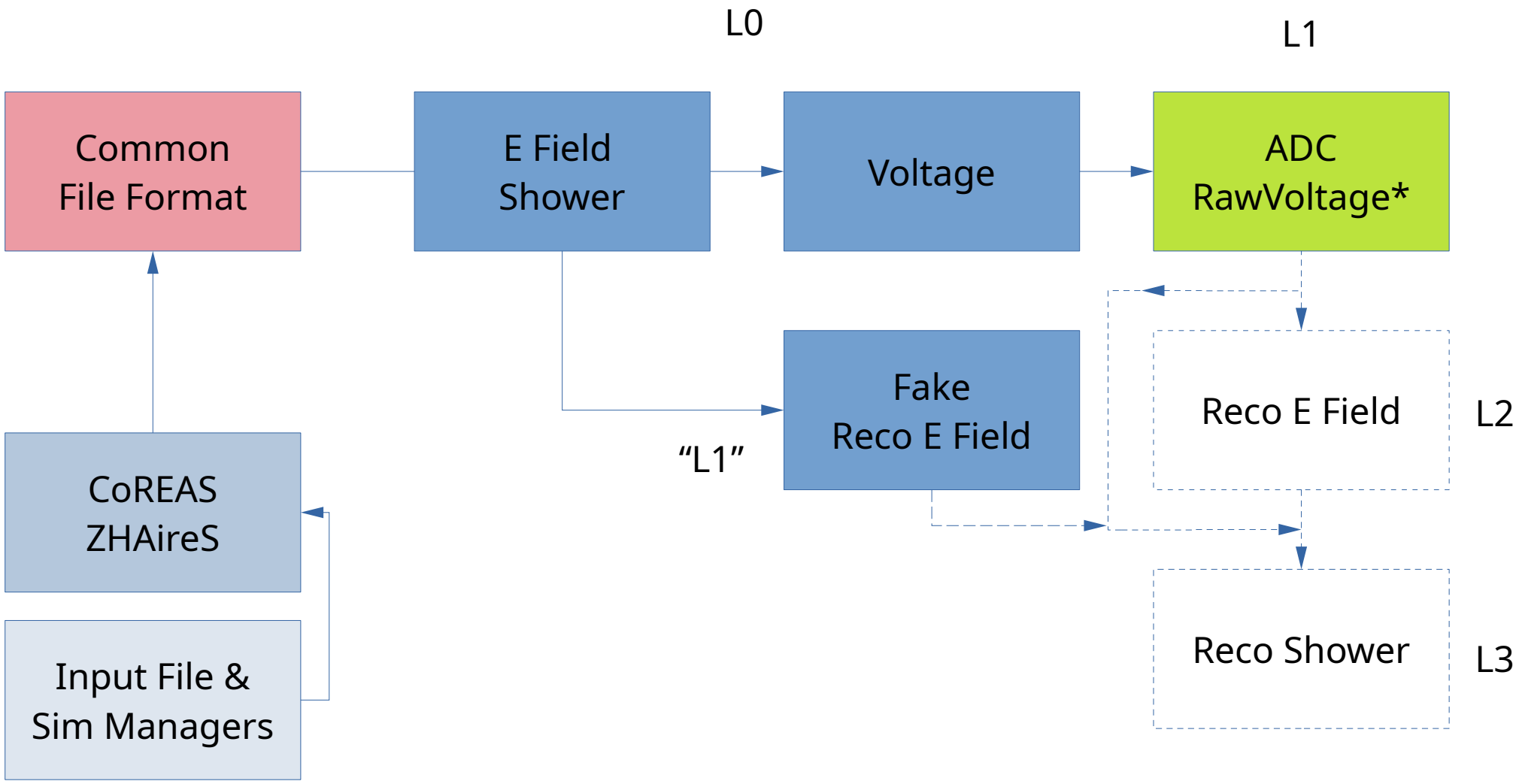
How is the simulation pipe organized



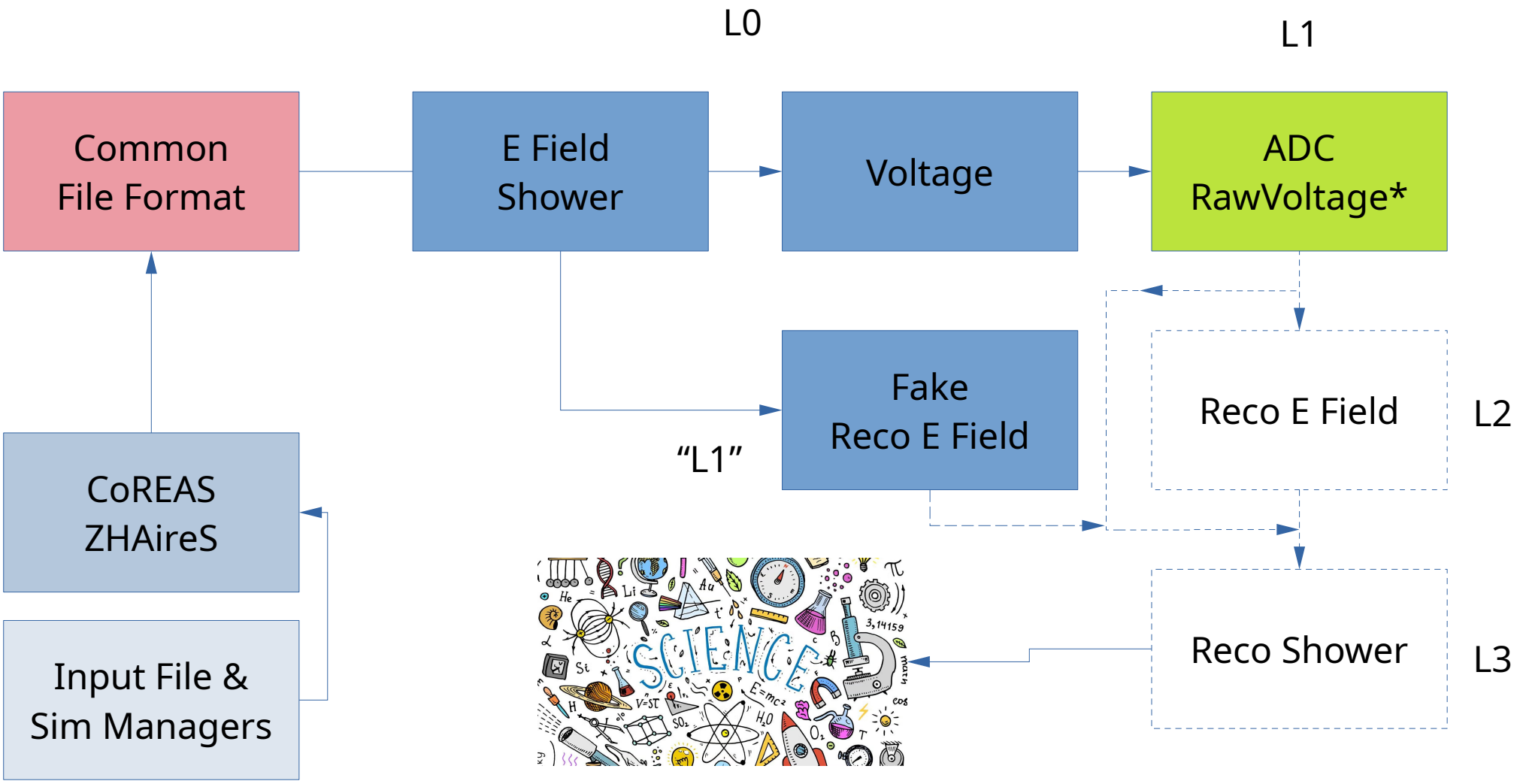
How is the simulation pipe organized



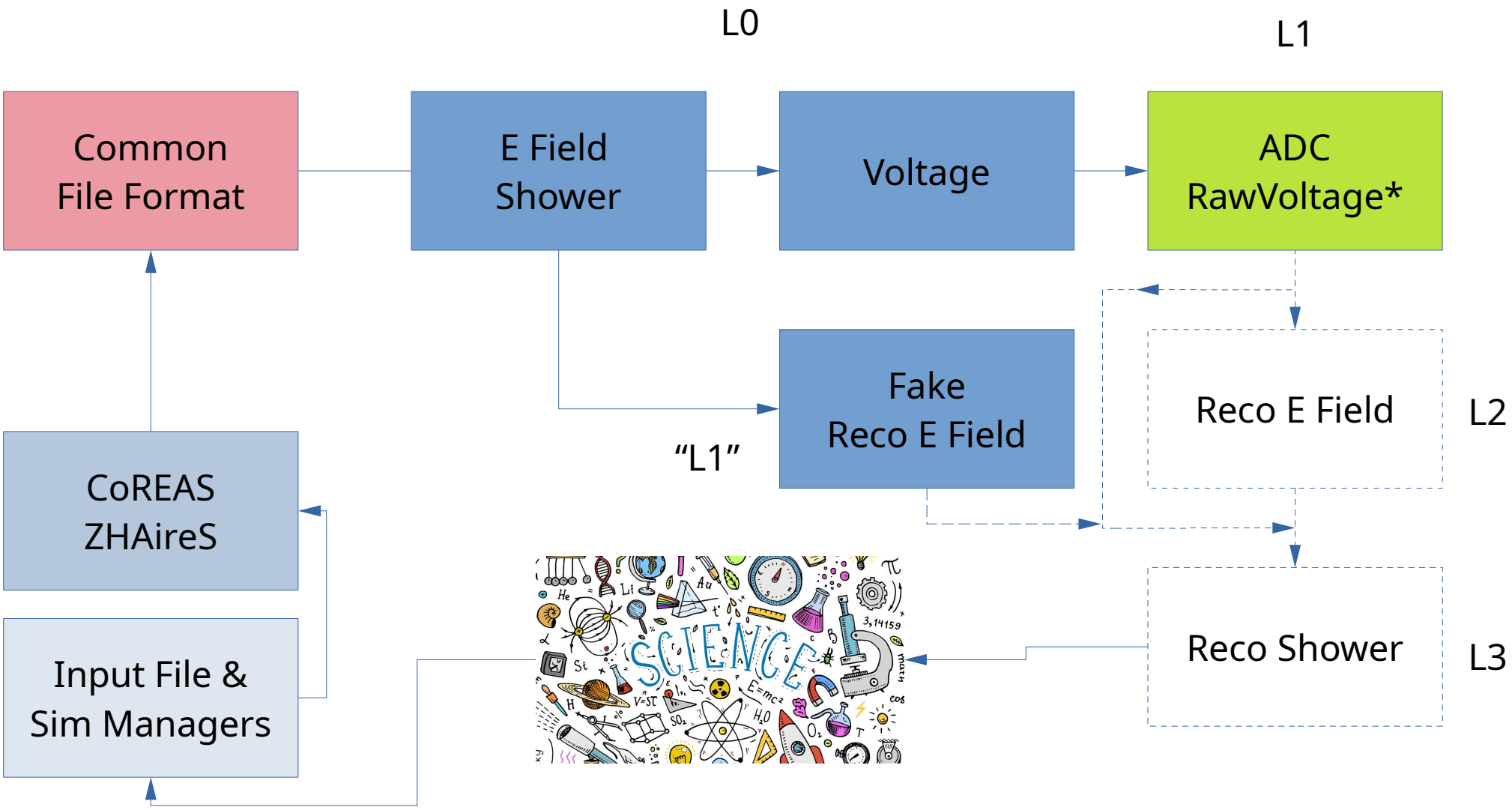
How is the simulation pipe organized



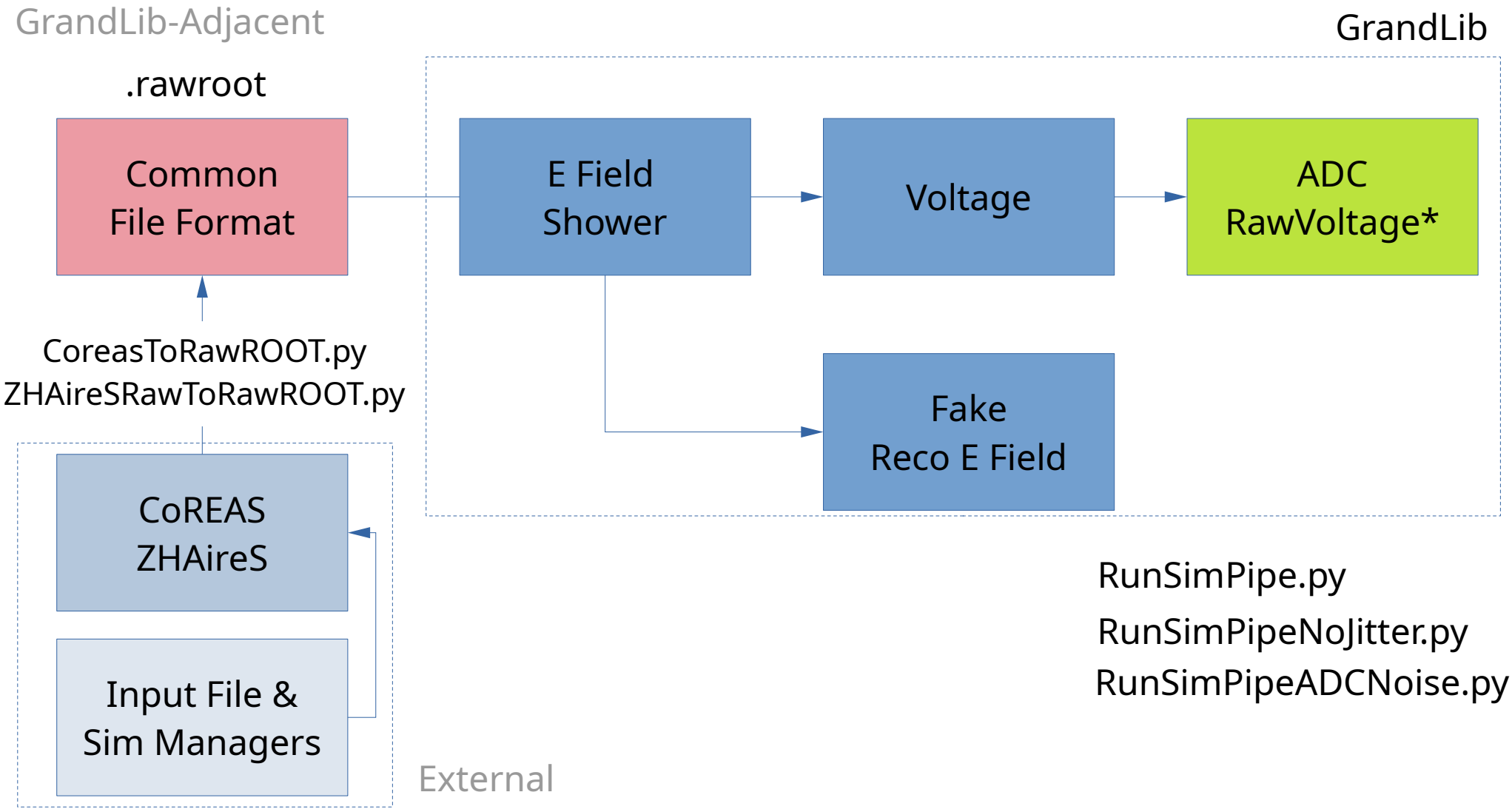
How is the simulation pipe organized



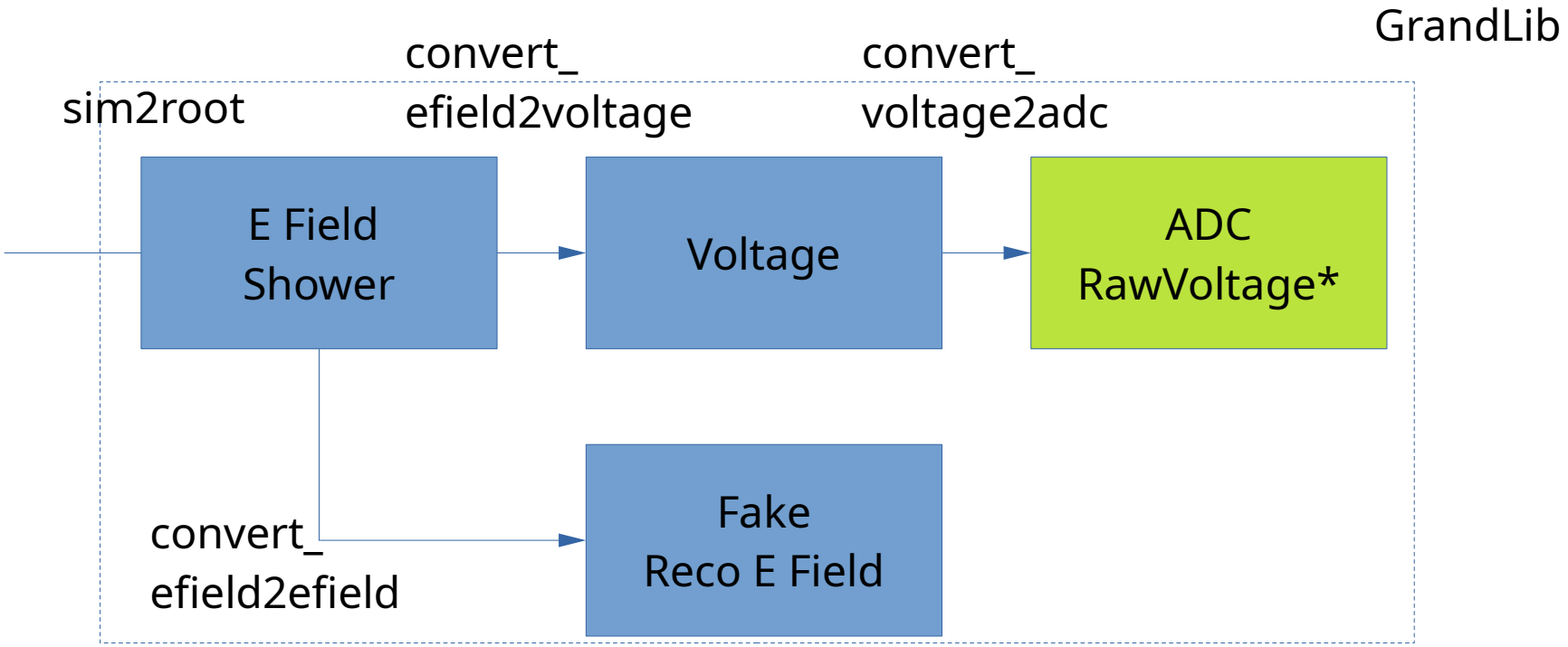
How is the simulation pipe organized



How is the simulation pipe organized



How RunSimPipe works



The order of the scripts is rigid!. Root files cannot be updated, only regenerated

How sim2root works

```
sim2root.py INPUTDIR  --verbose=warning
        --target_duration_us=2.048      #length of the trace
        --trigger_time_ns 550           #where the signal peak will be inside the trace
        -sl  ArrayName                  #the name of the array (site layout)
        -ef 250                         #how many events per file
        -e EXTRA                        #EXTRA thing you want in the filename
                                        sim_Xiaodushan_20221025_220000_RUN0_CD_EXTRA_0000
```

What does it do?

- Read the information on the rawroot file to:
  - adapt sim data to GRAND conventions
  - compute derived quanties not given by the rawroot
  - create L0 run, runefieldsim, rushowersim
  - create L0 showersim
  - create L0 shower
  - create L0 efield

How convert_efield2voltage works

```
convert_efield2voltage  INPUTDIR  --seed 1234 --verbose=warning
        --add_jitter_ns 5                                   #shift the trace to simulate GPS jitter
        --calibration_smearing_sigma 0.075                  #multiply trace amplitude to simulate calibration
        --no_noise                                          #if you don't want galactic noise
```

What does it do?

- Read the information on the efield file (and some others):
  - Apply the antenna response
  - Apply the RFChain
  - Add galactic noise
  - create L0 voltage

How convert_voltage2adc works

```
convert_voltage2adc  INPUTDIR  --verbose=warning
  --add_noise_from directory        #directory where ADC noise traces are (if you want ADC noise)
  --seed 1234                       # random seed
```

What does it do?

- Read the information on the voltage file (and some others):
  - Downsample to 500Mhz
  - create L1 files
  - Add ADC noise if that is what you want (pick at random traces from the noise files)
  - SOON: Create rawvoltage files, delete voltage files ?
  - NEXT: Trigger?

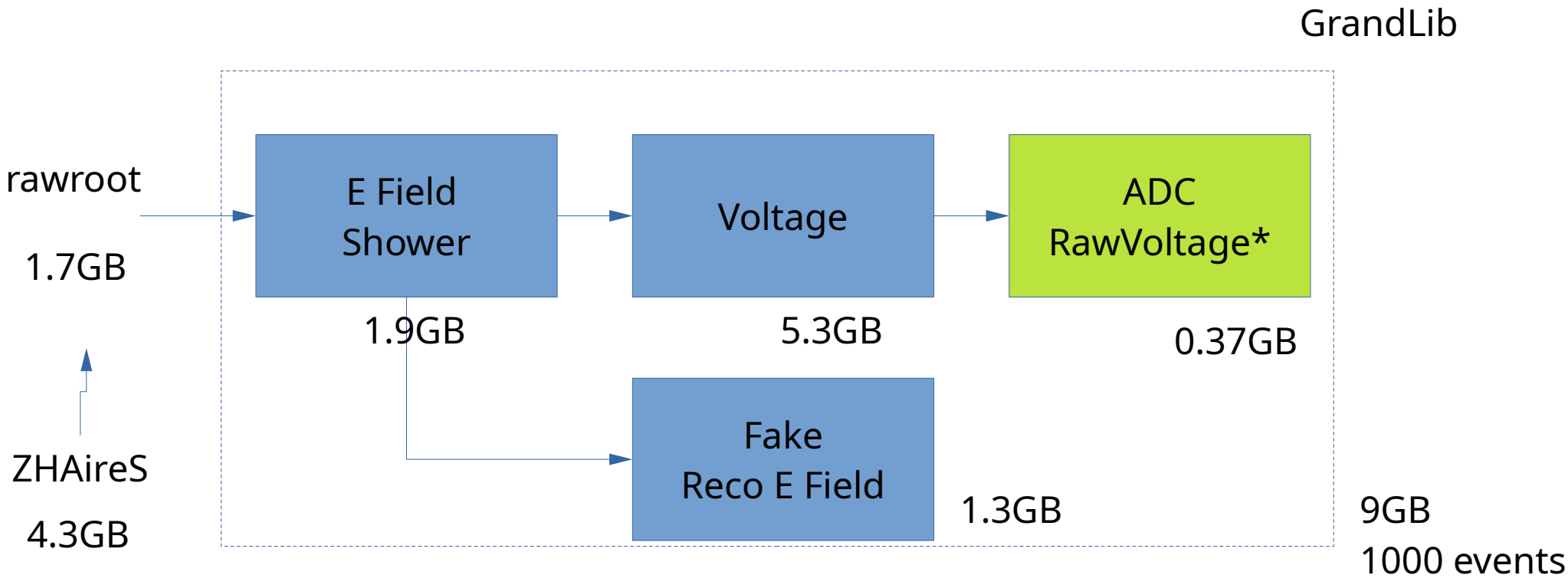
How convert_efield2efield works

```
convert_efield2efield INPUTDIR --verbose=warning
--add_noise_uVm 64                    #gaussian noise added before filtering
--add_jitter_ns 5                     #time jitter
--seed 1234                           #set random seed
--calibration_smearing_sigma 0.075    #amplitude jitter
--target_duration_us 2.048            #track length
--target_sampling_rate_mhz 500        #sampling rate
--no_filter                           #remove the filter (not in use)
```

What does it do?

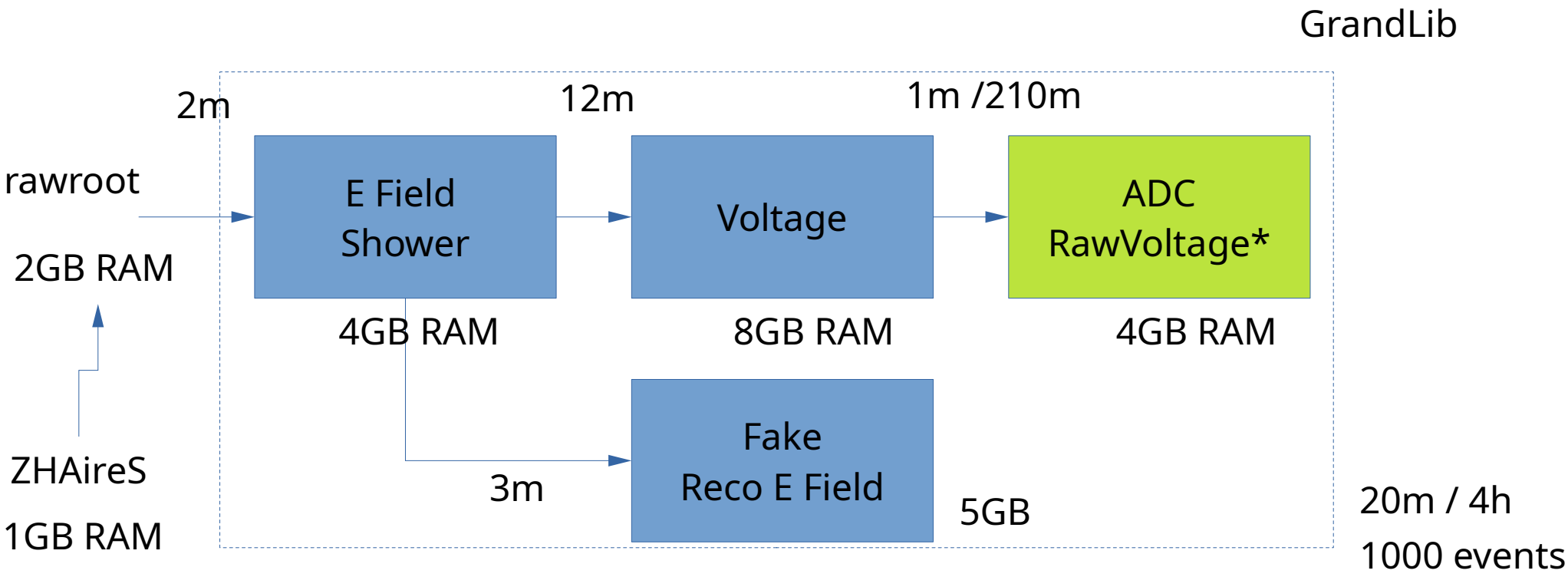
- Read the information on the efield file (and some others):
  - Add Gaussian noise
  - Elliptic Causal Filter 50-200Mhz
  - Downsample to 500Mhz
  - create L1 efield files

How Much Space it takes?



15GB stored per 1000 events! Too much!.
--Get rid of Voltage?

How Much Time and Memory it Takes?



For how RunSimPipe is done, you cannot run in parallel in the same directory
Because of Memory constraints, I run 2000 events at a time
Because I still could not get to run grandlib in lyon, run in my PC → laborur intense