PROPOSED PDS-RECO1 FOR STANDARD RECONSTRUCTION

SBND Winter Collaboration Meeting

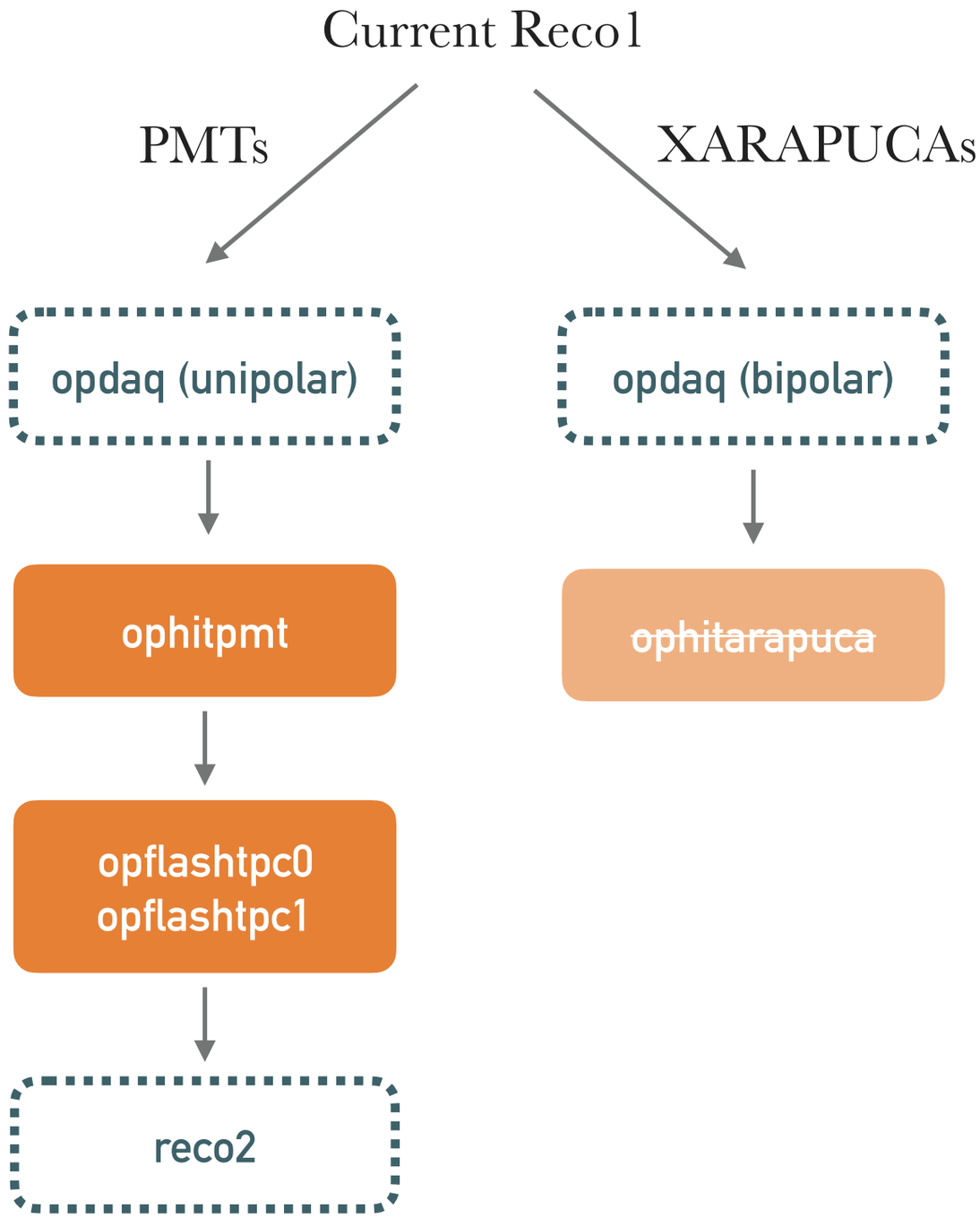
07/12/22

Francisco Javier Nicolás Arnaldos

fjnicolas@ugr.es

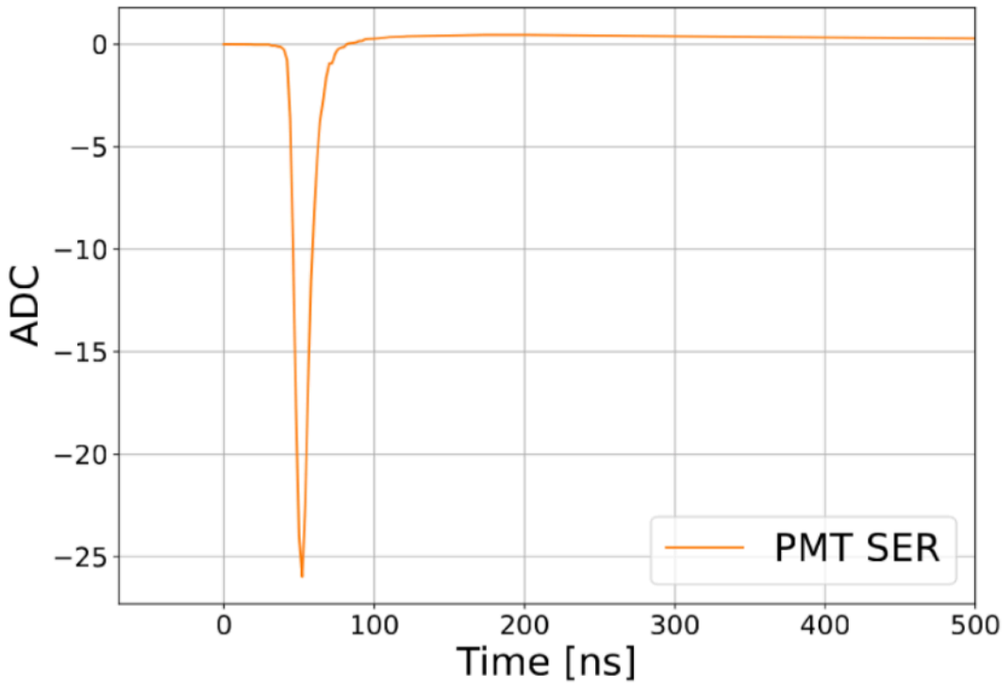
PDS RECO1: LONG STORY SHORT

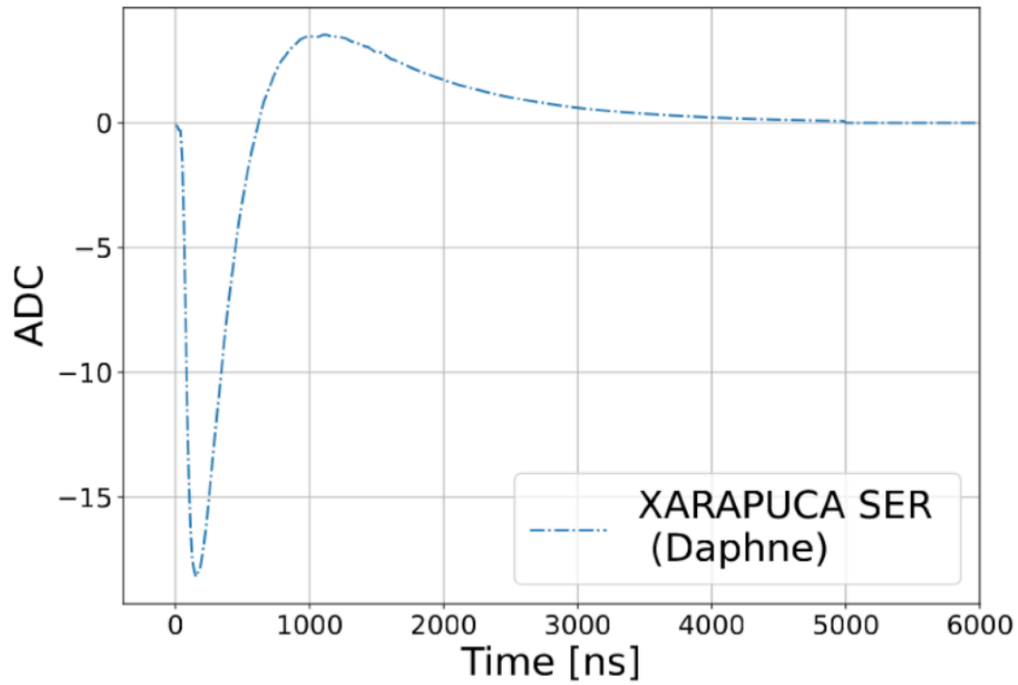
- Back in 2020: first PDS reconstruction modules landed in sbndcode (work done by M.Tutto and others, 19169)
- Reco1 divided in two stages
  - Pulse finding: produces **OpHit** objects
  - Clustering among different photon detectors (PDs): produces **OpFlash** objects
- Using unipolar (not realistic) PMT SER
- Two different PD technologies: reco1 runs twice, one per subsystem
- This has been our default PDS reco1 in the past MC productions



PDS RECO1: LONG STORY SHORT

- Our signals will look different to what we've been using thus far: PDs are AC coupled (bipolar)





- A dedicated reconstruction to treat the overshoot issue was required
  - Overshoot makes output charge reconstruction challenging
- Proposed solution for both subsystems: signal **deconvolution** before hit finding (in a similar fashion to the TPC)
  - PMTs: 26422, 26791, 25323
  - XARAPUCAs (Rodrigo): 26809, 25332, 24900

PDS RECO1: LONG STORY SHORT

- Full reco1 chain including realistic digitization ready (after sbndcode PR301) for both PMT & XARAPUCA systems!
- We propose to include it in the standard reco fhicls to start using by default
  - You can follow the updates in sbndcode PR309:
  - Merge intended in the next ~1/2 releases

Updated PDS reco1 #309

Draft **fjnicolas** wants to merge 5 commits into `develop` from `feature/pdsdeco_workflow`

Conversation 0 | Commits 5 | Checks 0 | Files changed 11

fjnicolas commented 5 minutes ago • edited — Member

This PR updates the PDS reco1 workflow used in our standard fhicls. Summary of the changes:

- Switch to the measured SER for PMTs
- Move to the refactored PDS reconstruction workflow including signal deconvolution
- Turns on by default the updates to the OpFlash t0 included in Update for OpFlash t0 #289

Discussed in docdb#

Assignees

fjnicolas

rodralva

Labels

breaking change

UPDATES @ DETSIM

- Switch to the PMT test-bench SER
- Changes to DetSim configuration fhicl for PMTs:

sbndcode/OpDetSim/digi_pmt_sbnd.fcl

```
-   PMTSinglePEmodel:     false   # false for ideal PMT response, true for test bench measured response
-   PMTDataFile:          "OpDetSim/digi_pmt_sbnd.root" # located in sbnd_data
33  +   PMTSinglePEmodel:     true   # false for ideal PMT response, true for test bench measured response
34  +   PMTDataFile:          "OpDetSim/digi_pmt_sbnd_v2int0.root" # located in sbnd_data
```

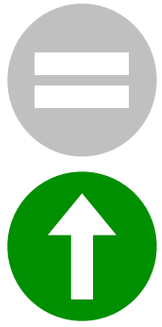
- XARAPUCAs are already using the bipolar SER:

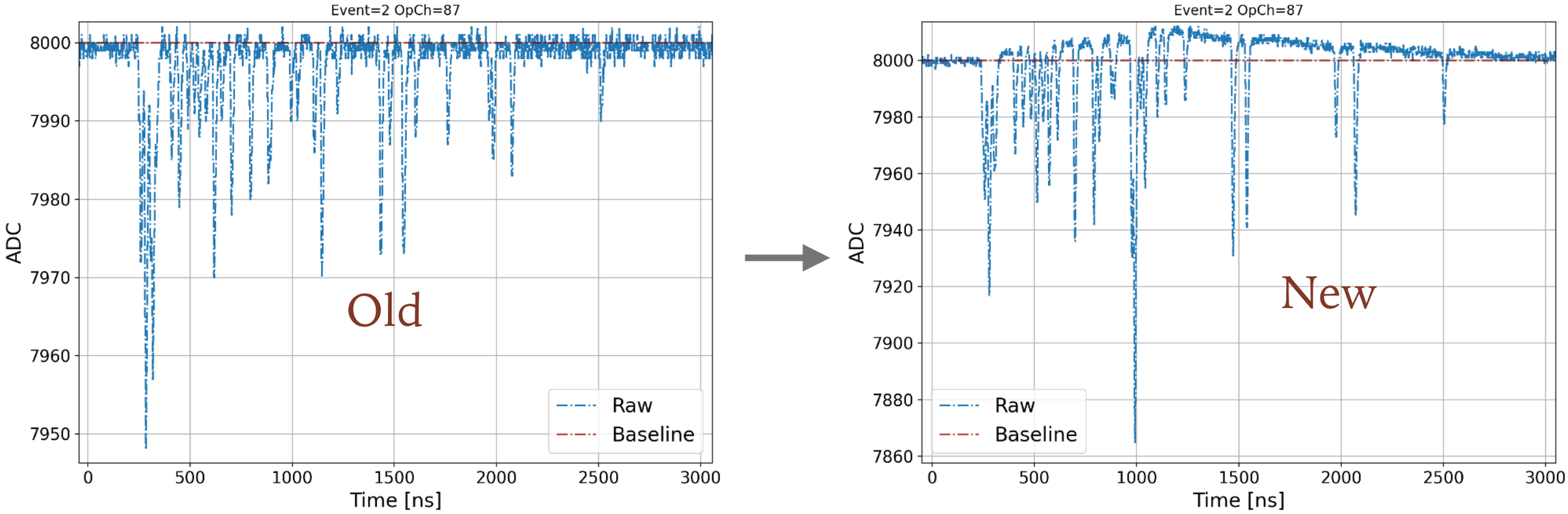
sbndcode/OpDetSim/digi_arapuca_sbnd.fcl

```
XArapucaVISEff:           0.014   #XArapuca VIS efficiency (taking into account 70% mesh transparency 0.02*0.7)
DecayTXArapucaVIS:        8.5     # decay time of EJ280 in ns
ArapucaDataFile:          "OpDetSim/digi_arapuca_sbnd.root" # located in sbnd_data
ArapucaSinglePEmodel:     true    # false for ideal response true for response from XTDBoard cold tests
DaphneFrequency:          80.0    #in MHz. Frequency of the Daphne Readouts
```

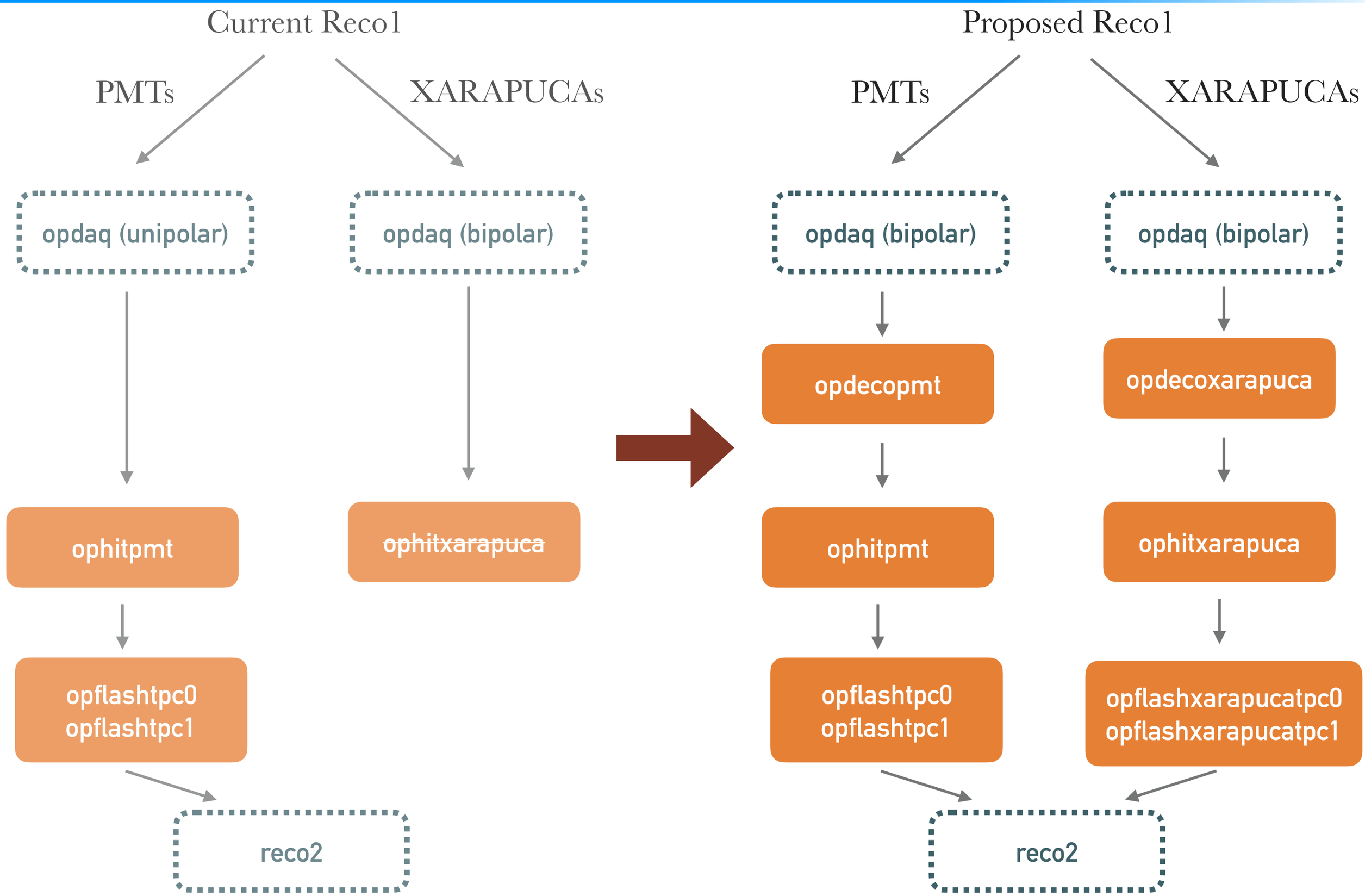
UPDATES @ DETSIM

- What changes should we expect **@DetSim**?
- No changes in memory/CPU/file size
- Different PMT waveforms (closer to data)





UPDATES @ RECO1



UPDATES @ RECO1

- Refactored reco fhicl:

sbndcode/JobConfigurations/base/reco_sbnd.fcl

```
 96  +   ### optical deconvolution
 97  +   opdecopmt:           @local::SBNDOpDeconvolutionPMT
 98  +   opdecoxarapuca:      @local::SBNDOpDeconvolutionXARAPUCA
 99  +
100      ### optical hit finders
101      # ophit:                 @local::sbnd_hit_finder
     -   ophitpmt:              @local::sbnd_ophit_finder_pmt
     -   ophitarapuca:         @local::sbnd_ophit_finder_arapuca
102  +   ophitpmt:              @local::SBNDDecoOpHitFinderPMT
103  +   ophitdecoxarapuca:     @local::SBNDDecoOpHitFinderXArapuca
104
105      ### flash finders
106      # opflash:               @local::sbnd_opflash
     -   opflashtpc0:           @local::SBNDSimpleFlashTPC0
     -   opflashtpc1:           @local::SBNDSimpleFlashTPC1
107  +   opflashtpc0:           @local::SBNDDecoSimpleFlashTPC0
108  +   opflashtpc1:           @local::SBNDDecoSimpleFlashTPC1
109  +
110  +   # opflash(arapucas):              @local::sbnd_opflash
111  +   opflashtpc0xarapuca:          @local::SBNDDecoSimpleFlashTPC0Arapuca
112  +   opflashtpc1xarapuca:          @local::SBNDDecoSimpleFlashTPC1Arapuca
```

- New PMT/XARAPUCA deconvolution stage
- We use the module in sbndcode/OpDetReco/OpDeconvolution

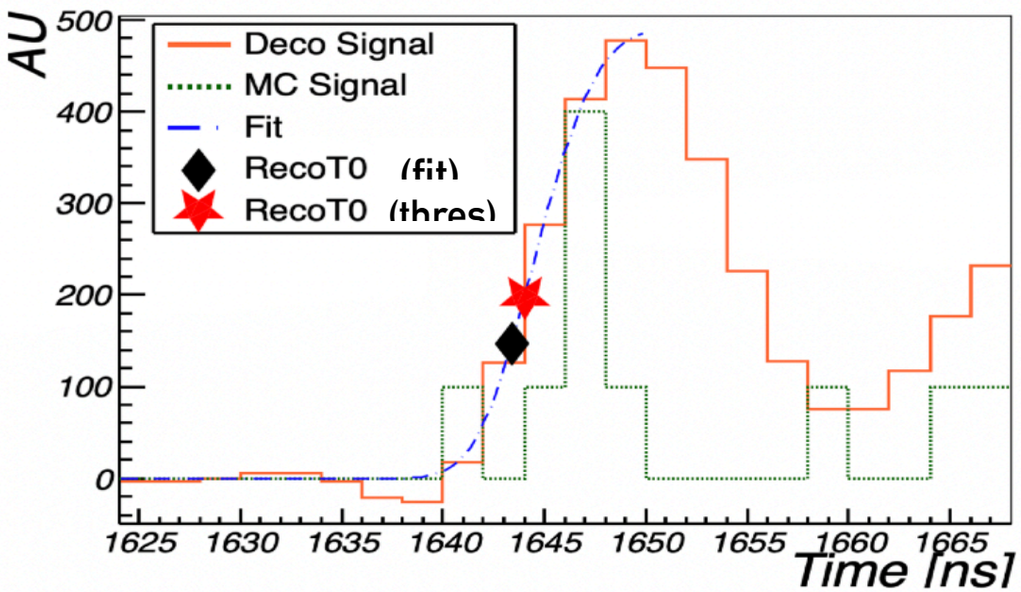
- OpHit finder tuned for the deconvolved signals
- We use the same algorithms from larana with updated parameters

- Updated OpFlash finder
- Includes recent improvements to the time resolution (next slide)

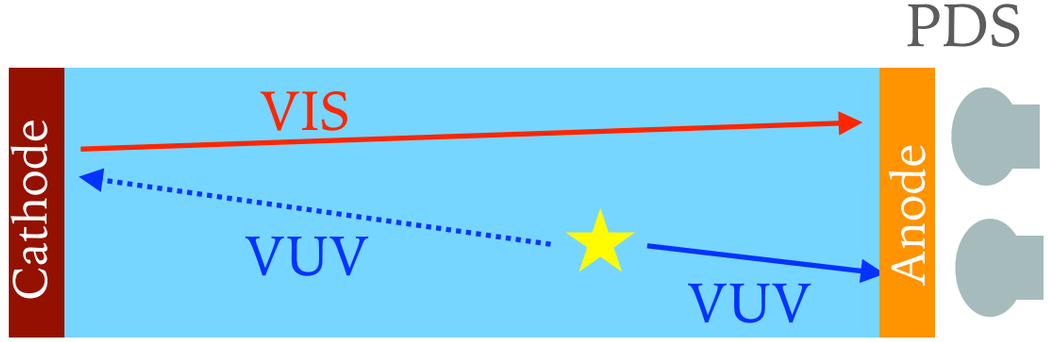
- New OpFlash for XARAPUCAs

UPDATES @ RECO1

- Recent updates to the OpFlash t_0 reconstruction include corrections to:
  - Waveform rise time
  - t_0 estimation algorithm
  - Scintillation photon time of flight



- Time resolution ~2 ns
- These changes were presented in the last CM and merged in sbndcode PR289



- These corrections are turned on in the proposed OpFlash configuration

UPDATES @ RECO1

- What changes should we expect in the **artroot files**? 🟰
  - Deconvolved waveforms stored in raw::OpDetWaveform data products
  - These are fed into the OpHit finder algorithm and then dropped
  - No impact on artroot file size

Reco1	opflashtpc1xarapuca		art::Assns<recob::OpHit,recob::OpFlash,void>	209
Reco1	rns		std::vector<art::RNGsnapshot>	0
Reco1	opdecopmt		std::vector<raw::OpDetWaveform>	?
Reco1	TriggerResults		art::TriggerResults	1
Reco1	opflashtpc0xarapuca		std::vector<recob::OpFlash>	0
Reco1	opflashtpc1		std::vector<recob::OpFlash>	2
Reco1	caldata		std::vector<recob::Wire>	11224
Reco1	opflashtpc0		std::vector<recob::OpFlash>	1
Reco1	opflashtpc0		art::Assns<recob::OpHit,recob::OpFlash,void>	94
Reco1	gaushit		art::Assns<recob::Wire,recob::Hit,void>	112
Reco1	opflashtpc1		art::Assns<recob::OpHit,recob::OpFlash,void>	1727
Reco1	gaushit		std::vector<recob::Hit>	112
Reco1	opflashtpc0xarapuca		art::Assns<recob::OpHit,recob::OpFlash,void>	0
Reco1	opdecoxarapuca		std::vector<raw::OpDetWaveform>	?
Reco1	opflashtpc1xarapuca		std::vector<recob::OpFlash>	1
Reco1	caldata		art::Assns<raw::RawDigit,recob::Wire,void>	11224
Reco1	ophitxarapuca		std::vector<recob::OpHit>	241
Reco1	gaushitTruthMatch		art::Assns<simb::MCParticle,recob::Hit,anab::BackTrackerHitMatchingData>	52
Reco1	ophitpmt		std::vector<recob::OpHit>	2068

Dropped

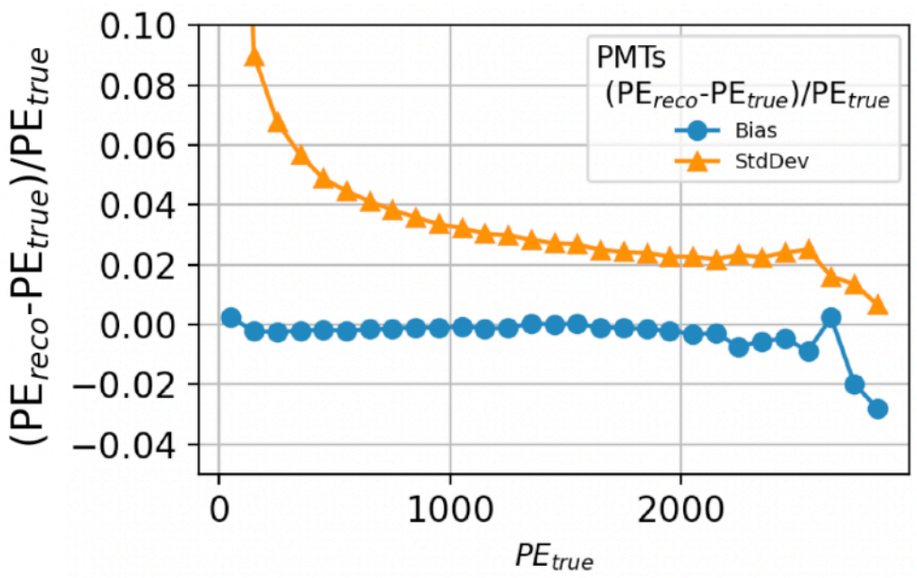
New

New

Dropped

UPDATES @ RECO1

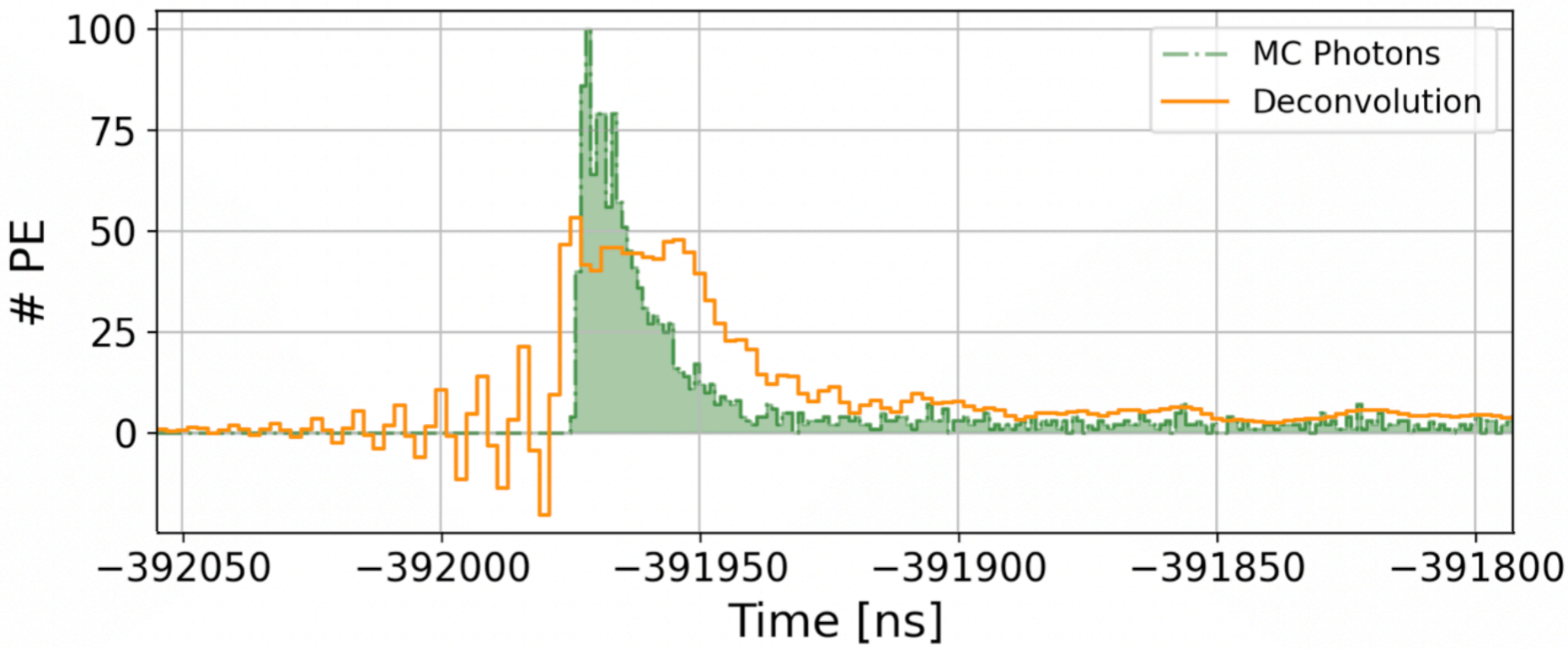
- What changes should we expect in the **light calorimetry**?
- PMTs:
  - Similar/identical performance to what we currently have using the ideal SER*, but…
  - using **<u>waveforms/reconstruction tools closer to what we'll have when data arrives</u>**



  - Caveat: saturated signals (next slide)
  - Similar Reco2 performance expected. Nevertheless a verification/optimization would be convenient.
- New: working OpHit objects for the XARAPUCAs

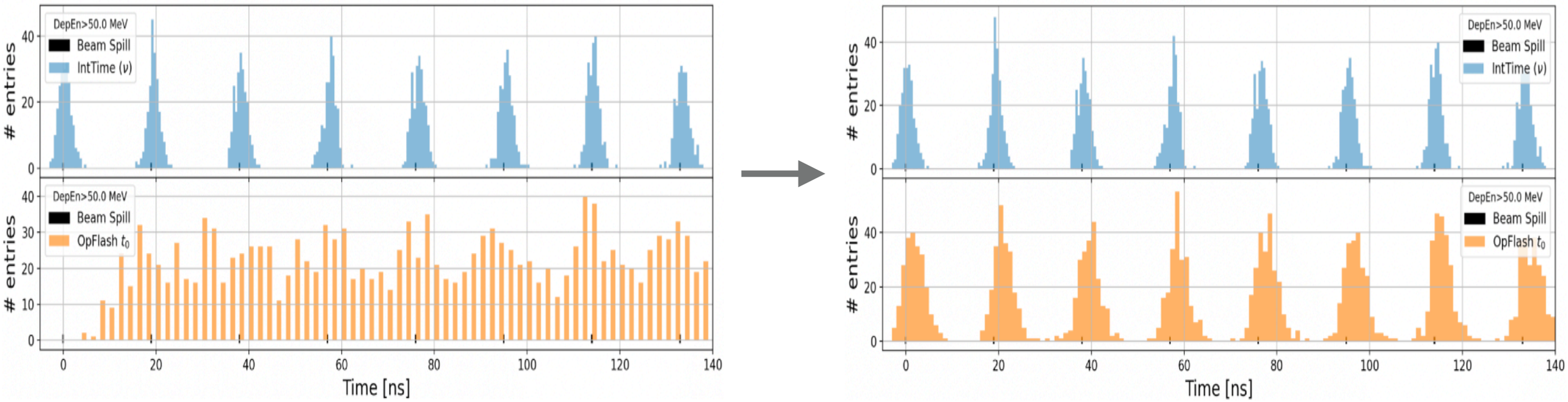
UPDATES @ RECO1

- A dedicated reconstruction would be needed for the saturated signals
  - Deconvolution doesn't make sense in a truncated signal
- Fake OpHits reconstructed due to the waving introduced by the deconvolution
- These signals <u>will not be included</u> in the OpFlash reconstruction
  - Including them worsen the t_0 estimation



UPDATES @ RECO1

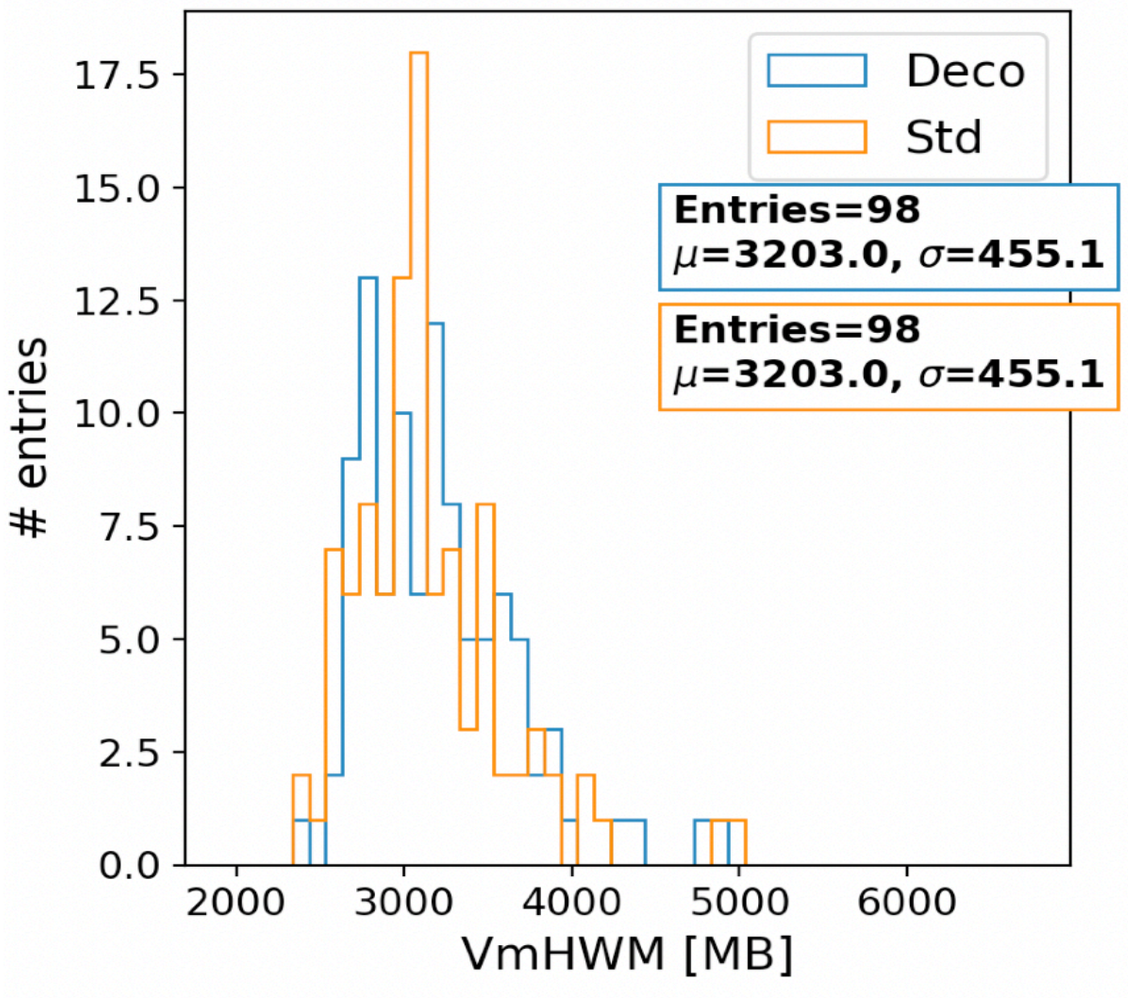
- What changes should we expect in the **OpFlash objects**?
- PMTs:
  - Same reco efficiency and #PE resolution
  - Improved time resolution!

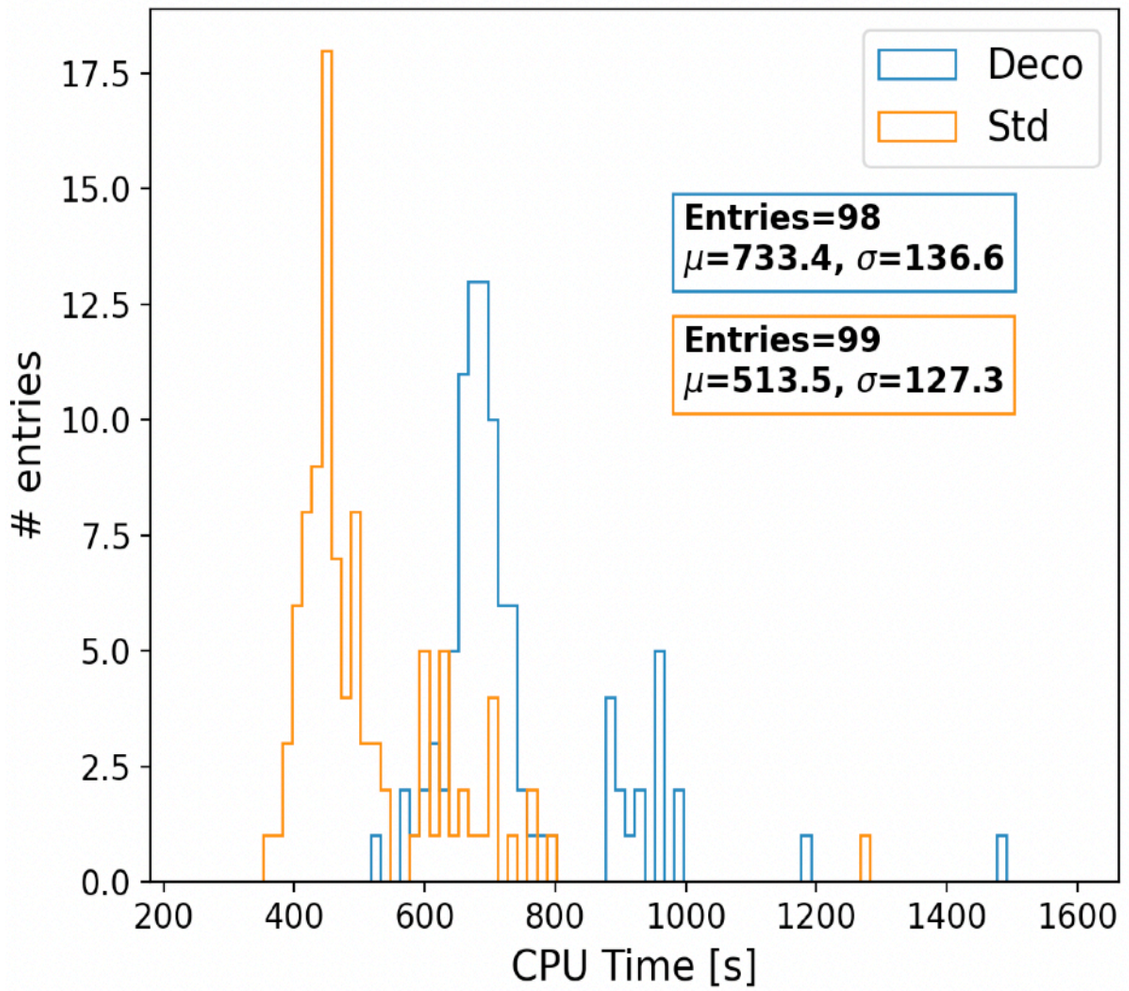


- New: OpFlash for the XARAPUCAs

UPDATES @ RECO1

- What changes should we expect in the **memory usage/cpu usage**?
  - Same memory usage
  - Cost: cpu time as we are also deconvolving the PDS system
- Workflow validated in past productions: MCP2021Cv2_prodoverlay_corsika_cosmics_proton_genie_nu_spill_gsimple-configh-v1_tpc





* Sample nu+cosmics with 20ev/job

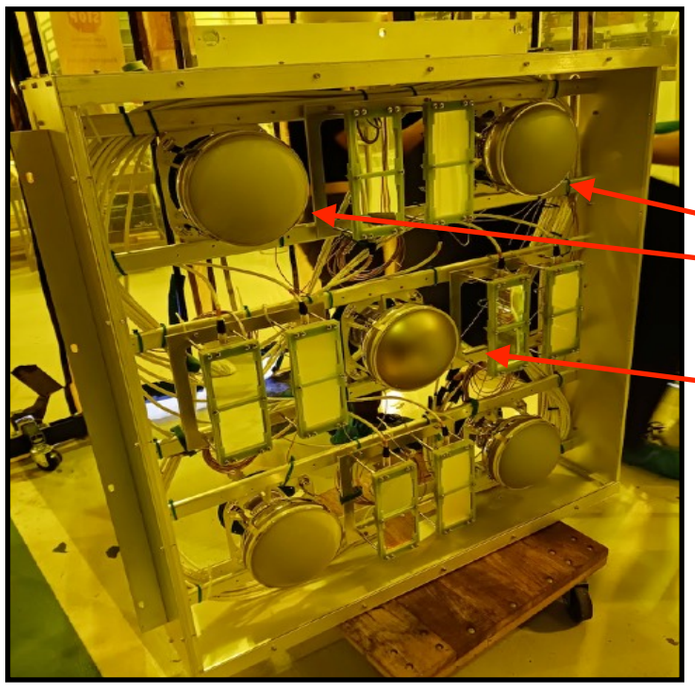
PMT DETECTION EFFICIENCIES

- We'll be also updating the PMT detection efficiencies
- Currently we simulate two detection efficiencies, distinguishing by light component (VUV or visible)

sbndcode/OpDetSim/digi_pmt_sbnd.fcl

```
31     QEDirect:                  0.03    #PMT quantum efficiency for direct (VUV) light
32     QERefl:                    0.03    #PMT quantum efficiency for reflected (TPB converted) light
```

- Changes in feature/fnicolas_pmtqe
- Adds the feature to distinguish by light component and by PMT flavor



```
PMTCoatedVUVEff:        0.12
PMTCoatedVISEff:        0.14
PMTUncoatedEff:         0.19
```

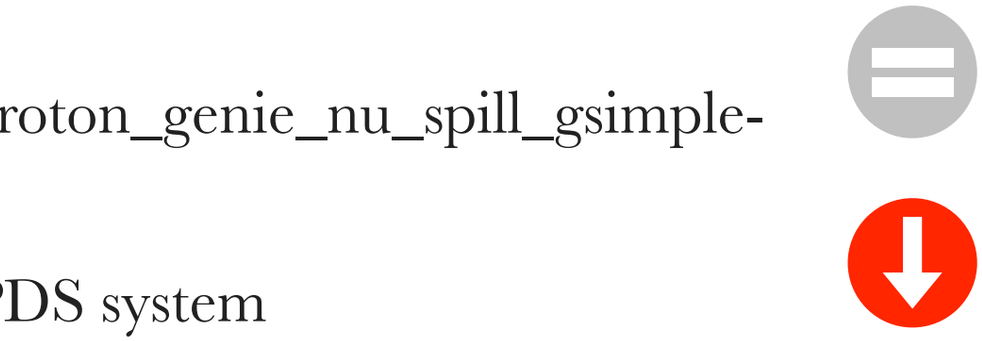
Not the final values

SUMMARY

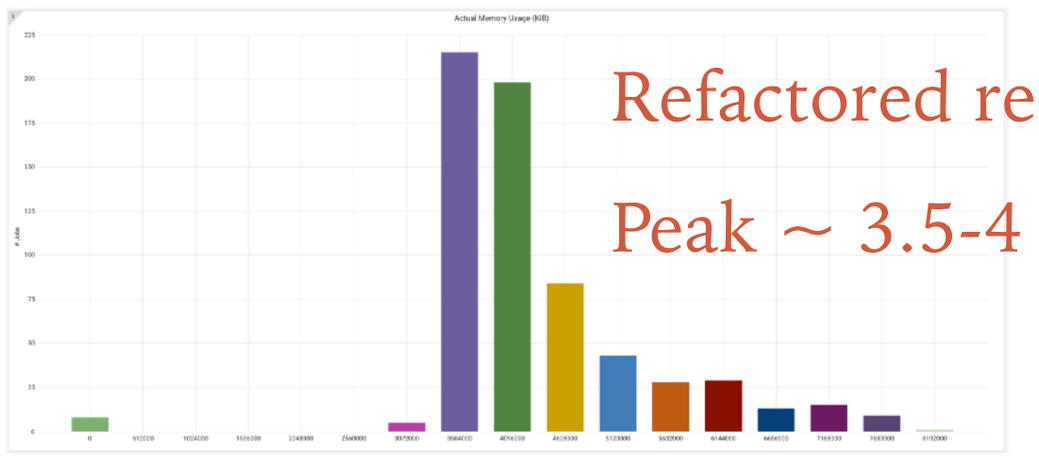
- We propose to switch to the updated reco1 stage for the PDS, including:
  - Measured SER for PMTs
  - Refactored workflow including deconvolution
    - Improved time resolution
- The PDS simulation/reconstruction paper (see db-27859) will use this workflow

UPDATES @ RECO1

- What changes should we expect in the **memory usage/cpu usage**?
  - Same memory usage (MCP2021Cv2_prodoverlay_corsika_cosmics_proton_genie_nu_spill_gsimple-configh-v1_tpc))
  - Cost: cpu time as we are also deconvolving the PDS system

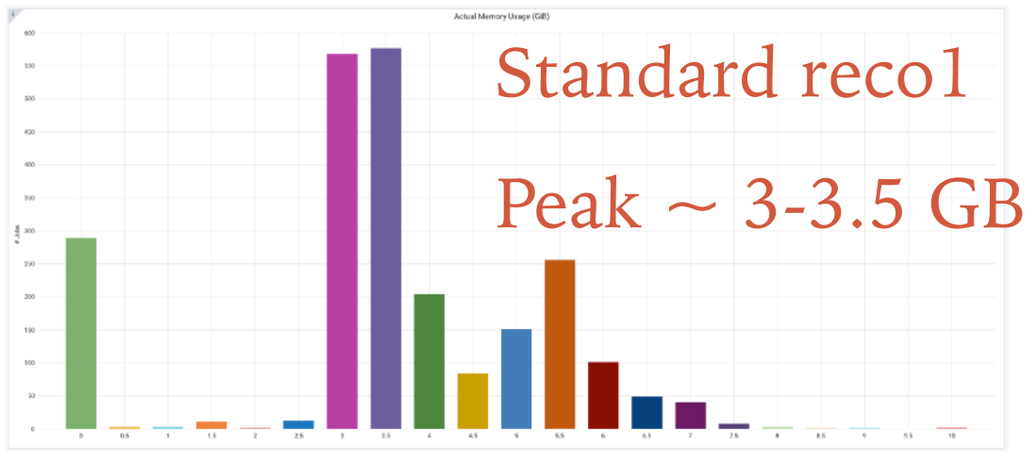


Memory Usage [GB]



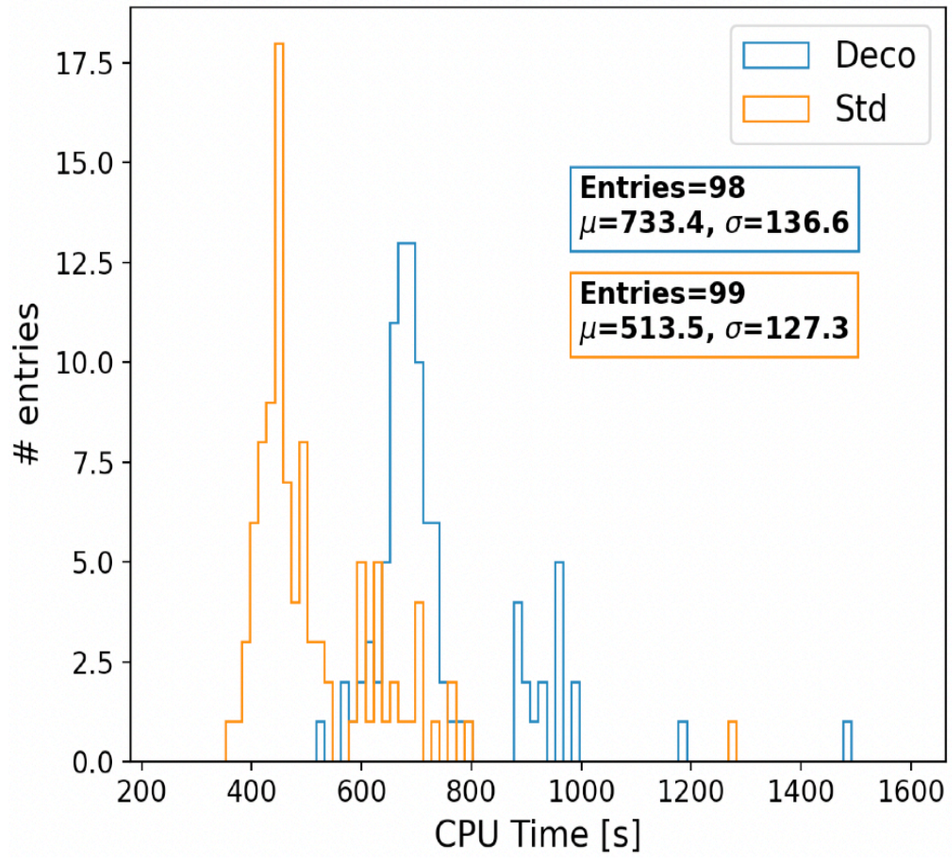
Refactored reco1

Peak ~ 3.5-4 GB



Standard reco1

Peak ~ 3-3.5 GB



* CPU time for reco1
* Sample nu+cosmics with 20ev/job