

Refactored CRT Update

Richie Diurba

David Rivera

Changes

- Very preliminary work
- I modified the perl and shell scripts used for the Legacy gdml files and repurposed them to generate the refactored gdml files
 - The scripts have the added convenience of loops; no need to manually edit the gdml
- Richie rewrote the section that generates the CRT elements
 - No "_" allowed in larg4 (see legacy version on the right ->)

```
<volume name="volAuxDetSensitive_CRT_Paddle_U1_1">
  <materialref ref="Polystyrene"/>
  <solidref ref="CRT_Paddle"/>
</volume>
<volume name="volAuxDetSensitive_CRT_Paddle_U1_2">
  <materialref ref="Polystyrene"/>
  <solidref ref="CRT_Paddle"/>
</volume>
<volume name="volAuxDetSensitive_CRT_Paddle_U1_3">
  <materialref ref="Polystyrene"/>
  <solidref ref="CRT_Paddle"/>
</volume>
<volume name="volAuxDetSensitive_CRT_Paddle_U1_4">
  <materialref ref="Polystyrene"/>
  <solidref ref="CRT_Paddle"/>
</volume>
<volume name="volAuxDetSensitive_CRT_Paddle_U1_5">
  <materialref ref="Polystyrene"/>
  <solidref ref="CRT_Paddle"/>
</volume>
<volume name="volAuxDetSensitive_CRT_Paddle_U1_6">
  <materialref ref="Polystyrene"/>
  <solidref ref="CRT_Paddle"/>
</volume>
```

Geometry

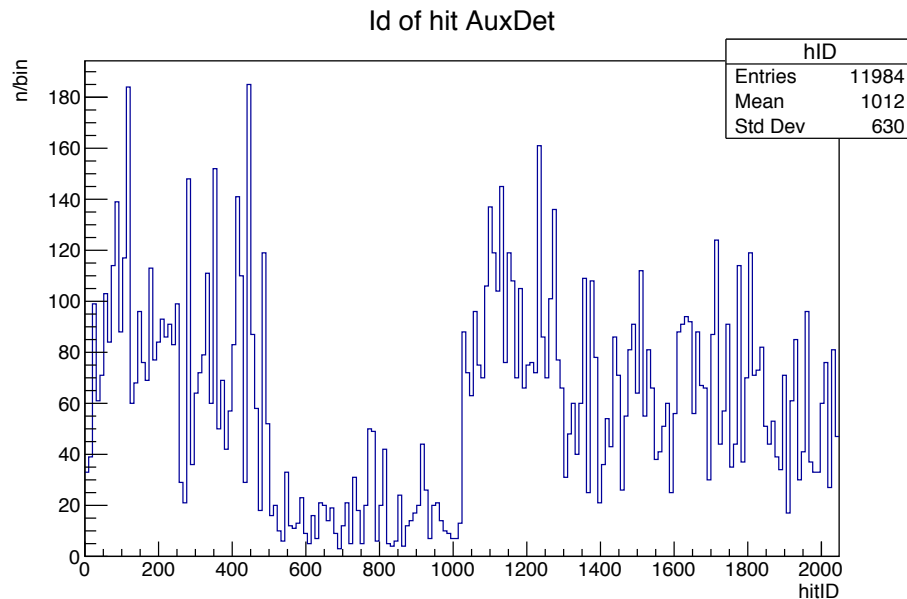
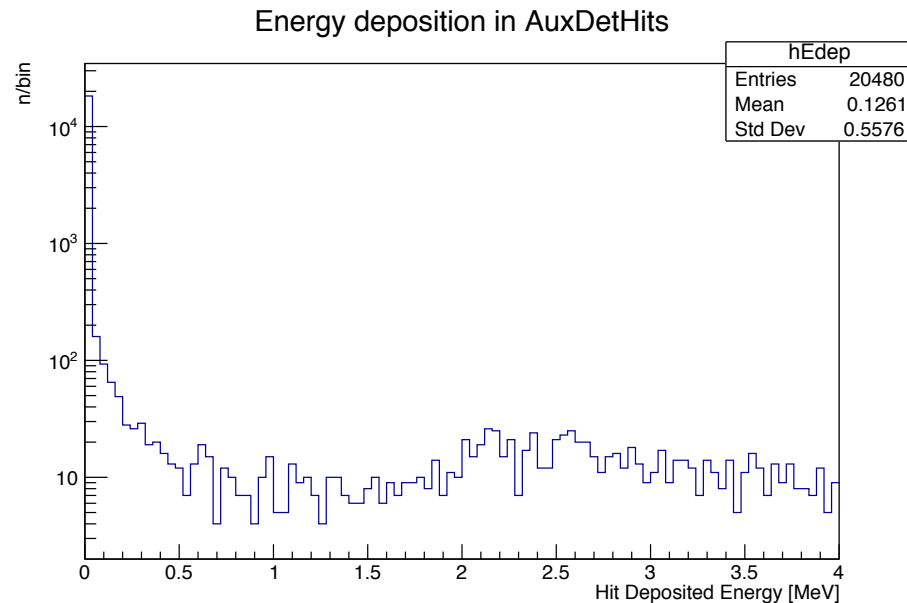
- As with the active Lar Volume, the CRT paddles are declared as *sensitive detectors* (**SensDet** of type **AuxDet**)
 - This allows for Geant to carry out the hit/energy deposition calculations
- GDML requires a few additions and a refactored naming convention
- (32modules x 64 strips/module) = 2048 strips
- Each strip is instantiated with a unique **copy number**

```
<volume name="volAuxDetSensitiveCRTPaddle">
  <materialref ref="Polystyrene"/>
  <solidref ref="CRTPaddle"/>
  <auxiliary auxtype="SensDet" auxvalue="AuxDet"/>
  <auxiliary auxtype="Solid" auxvalue="True"/>
</volume>
<volume name="volAuxDetCRTModuleU1">
  <materialref ref="Air"/>
  <solidref ref="CRTModule"/>
  <physvol name="CRTPaddle1U1" copynumber="1">
    <volumeref ref="volAuxDetSensitiveCRTPaddle"/>
    <position name="posCRTPaddleSensitive1U1" unit="cm"
      x="-78.75"
      y="0.5"
      z="0"/>
    <rotationref ref="rIdentity"/>
  </physvol>
  <physvol name="CRTPaddle33U1" copynumber="33">
    <volumeref ref="volAuxDetSensitiveCRTPaddle"/>
    <position name="posCRTPaddleSensitive33U1" unit="cm"
      x="-76.25"
      y="-0.5"
      z="0"/>
    <rotationref ref="rIdentity"/>
  </physvol>
  <physvol name="CRTPaddle2U1" copynumber="2">
    <volumeref ref="volAuxDetSensitiveCRTPaddle"/>
    <position name="posCRTPaddleSensitive2U1" unit="cm"
      x="-73.75"
      y="0.5"
      z="0"/>
    <rotationref ref="rIdentity"/>
  </physvol>
</volume>
```

protodune_v7_refactored_nowires.gdml

Initial tests

- Used simple analyzer in larg4
 - CheckAuxDetHit_module.cc
- 10 corsika events with ~700 primaries each
- Verified that the hitIDs made sense



TODO

- Carefully verify the mapping
 - Strip IDs must match Legacy numbering scheme
- The AuxDet hitID corresponds to the copy number
- AuxDetSimChannel objects need to be filled

TODO (continued)

- Only `sim::AuxDetHit` objects are currently stored
- We will need to define a producer of `sim::AuxDetSimChannel` objects
- Suggestion: start with the legacy module: `larsim/LArG4/AuxDetReadout`

PRINCIPAL TYPE: Event				
PROCESS NAME	MODULE LABEL..	PRODUCT INSTANCE NAME.....	DATA PRODUCT TYPE.....	..SIZE
CorsikaGen..	rns.....	std::vector<art::RNGsnapshot>.....2
CorsikaGen..	generator.....	std::vector<simb::MCTruth>.....1
CorsikaGen..	TriggerResults	art::TriggerResults.....-
G4.....	largeant.....	LArG4DetectorServicevolAuxDetSensitiveCRTPaddle	std::vector<sim::AuxDetHit>.....	...968
G4.....	elecDrift.....	std::vector<sim::SimChannel>.....	.12461
G4.....	rns.....	std::vector<art::RNGsnapshot>.....2
G4.....	TriggerResults	art::TriggerResults.....-
G4.....	largeant.....	std::vector<simb::MCParticle>.....	617752
G4.....	largeant.....	LArG4DetectorServicevolTPCActive.....	std::vector<sim::SimEnergyDeposit>.....	793422
G4.....	largeant.....	art::Assns<simb::MCTruth,simb::MCParticle,void>	617752
G4.....	largeant.....	LArG4DetectorServicevolTPCActiveOuter.....	std::vector<sim::SimEnergyDeposit>.....	.17699