

# ILD Software and Analysis Meeting

## APRIL inclusion in DDMarlinPandora

Tanguy Pasquier (speaker), Gérald Grenier, Imad Laktineh, Bo Li, Rémi Été

IP2I/Univ Lyon 1

August 28, 2024

# Context

- Working on SDHCAL calibration and corrections for ILD option 2 to prepare for APRIL PFA implementation
  - Detector model : ILD\_12\_v02 with Videau geometry and SDHCAL
  - Files on the grid at : `/ilc/user/g/ggrenier/prod/v02-02-03/`
- Development split in 3 packages

**SDHCALContent** for ILD option 2/SDHCAL calibration, corrections, ... [▶ Git repo](#)

**APRILContent** for APRIL PFA algorithms [▶ Git repo](#)

**DDMarlinPandora** Created new version of DDMarlinPandora [▶ Git repo](#)

- **Issue** : Need to include SDHCALContent and APRIL in DDMarlinPandora

# Contents

## SDHCALContent

- Have computed new sets of calibration constants for standard SDHCAL semi-digital energy reconstruction.
- New formulas to reconstruct energy with SDHCAL are explored.
- A M1 internship has looked into corrections for ILD option 2 barrel inter-module gaps.

## APRILContent

- Design of the APRIL split-cluster procedure is underway
- Timing implementation has started

## DDMarlinPandora

- Included SDHCALContent and APRIL in DDMarlinPandora

# SDHCALContent in DDMarlinPandora

- Compilation options : `cmake -C $ILCSOFT/ILCSOFT.cmake -DPANDORA_MONITORING=ON -DUSE_SDHCALCONTENT=ON -DSDHCALContent_DIR=/absolute/path/to/SDHCALContent ..`
- Using preprocessor tag (`#ifdef`) to include `SDHCALContent.h` header and register `SDHCALContent` plugins in `DDPandoraPFANewProcessor`
- Allows user to use the `SDHCAL` related plugins and corrections with `PandoraPFA` or `APRIL PFA`

# APRIL in DDMarlinPandora

- Compilation options : `cmake -C $ILCSOFT/ILCSOft.cmake -DPANDORA_MONITORING=ON -DUSE_APRIContent=ON -DAPRIContent_DIR=/absolute/path/to/APRIContent -Dmlpack_DIR=/absolute/path/to/mlpack/INSTALL ..`
- APRIContent uses mlpack nearest neighbours algorithms
- Possible to include SDHCALContent and APRIContent by combining both compilation options
- Using preprocessor tag (`#ifdef`) to include `APRIContent.h` header and register APRIL algorithms in `DDPandoraPFANewProcessor`
- Created a `DDCaloHitCreator::ChooseFactory()` to run APRIL with the right hit factory

# New Marlin parameter

- Created a "UseAPRIL" Marlin parameter :
  - 1 In the PFA xml file you use, in the section for the DDMarlinProcessor add :

```
<parameter name="UseAPRIL" type="bool">>false</parameter>
```
  - 2 To use Marlin with APRIL, add the following option when calling it :

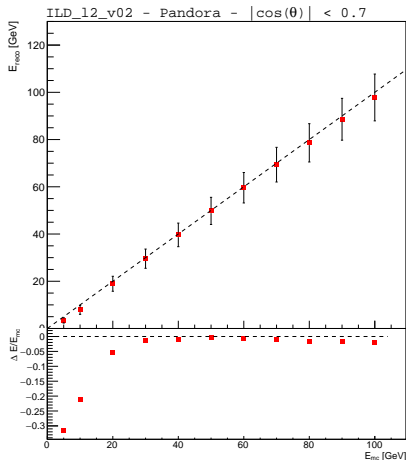
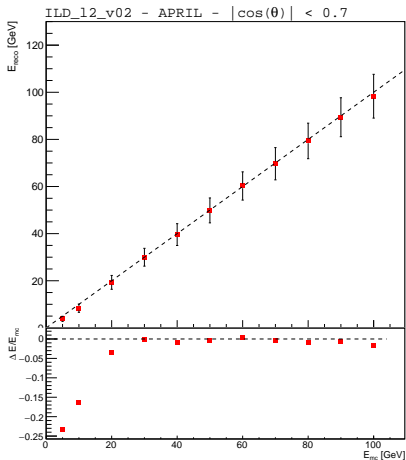
```
--MyDDMarlinPandora.UseAPRIL="true"
```
- UseAPRIL is on false by default
- If you compiled with APRILContent options, just switch the parameter on or off to choose between PandoraPFA and APRIL

# Issue with LCContent and APRILContent

- The Pandora `LCContent::RegisterBasicPlugins` method registers
  - AHCAL energy corrections,
  - PID plugins,
  - `PseudoLayerPlugin` and `ShowerProfilePlugin`.
- `PseudoLayerPlugin` and `ShowerProfilePlugin` can only be set once, preventing APRIL to use AHCAL energy corrections or Pandora PIDs.
- Returns `STATUS_CODE_ALREADY_INITIALIZED` if tried
- Ideal would be to split the LCContent plugins registrations
- **Temporary fix** : Created a `DDPandoraPFANewProcessor::PandoraHack` method that registers LCContent corrections and PID plugins
- Works, but not really elegant

# Some results

- Samples of single KLong reconstructed with SDHCAL angular corrections
- Linear reconstruction





# Summary and outlook

## Summary

- **DDMarlinPandora update**

- DDPandoraPFANewProcessor can run Pandora and APRIL
- APRILContent and SDHCALContent run in our DDMarlinPandora version
- Possibility to combine APRIL + LCContent plugins or Pandora + SDHCALContent plugins
- SDHCAL DDMarlinPandora ready to be propagated to iLCSoft (pull request done)

## Outlook

- **SDHCAL Calibration**

- Calibration in ILD\_12\_v02 better than before
- Still some work to do to perfect it
- Issue with lcgeo/k4geo Videau geometry recently found : to be investigated

- **APRILContent**

- Design of the APRIL split-cluster procedure has started
- Timing implementation in APRIL has started

# Backup

# LCContent plugins registration

```
pandora::StatusCode LCContent::RegisterBasicPlugins(const pandora::Pandora &pandora)
{
    LC_ENERGY_CORRECTION_LIST(PANDORA_REGISTER_ENERGY_CORRECTION);
    LC_PARTICLE_ID_LIST(PANDORA_REGISTER_PARTICLE_ID);

    PANDORA_RETURN_RESULT_IF(pandora::STATUS_CODE_SUCCESS, !=, PandoraApi::SetPseudoLayerPlugin(pandora, new lc_content::LCPseudoLayerPlugin));
    PANDORA_RETURN_RESULT_IF(pandora::STATUS_CODE_SUCCESS, !=, PandoraApi::SetShowerProfilePlugin(pandora, new lc_content::LCShowerProfilePlugin));

    return pandora::STATUS_CODE_SUCCESS;
}
```

# Temporary fix

- Created a "PandoraHack" method in DDPandoraPFANewProcessor that registers LCContent corrections and PID plugins

```
pandora::StatusCode PandoraHack(const pandora::Pandora &pandora)
{
    PandoraApi::RegisterEnergyCorrectionPlugin(pandora, "CleanClusters", pandora::HADRONIC, new lc_content::LCEnergyCorrectionPlugins::CleanCluster);
    PandoraApi::RegisterEnergyCorrectionPlugin(pandora, "ScaleHotHadrons", pandora::HADRONIC, new lc_content::LCEnergyCorrectionPlugins::ScaleHotHadrons);
    PandoraApi::RegisterEnergyCorrectionPlugin(pandora, "MuonCoilCorrection", pandora::HADRONIC, new lc_content::LCEnergyCorrectionPlugins::MuonCoilCorrection);

    PandoraApi::RegisterParticleIdPlugin(pandora, "LCemShowerId", new lc_content::LCParticleIdPlugins::LCemShowerId);
    PandoraApi::RegisterParticleIdPlugin(pandora, "LCPhotonId", new lc_content::LCParticleIdPlugins::LCPhotonId);
    PandoraApi::RegisterParticleIdPlugin(pandora, "LCElectronId", new lc_content::LCParticleIdPlugins::LCElectronId);
    PandoraApi::RegisterParticleIdPlugin(pandora, "LCMuonId", new lc_content::LCParticleIdPlugins::LCMuonId);

    return pandora::STATUS_CODE_SUCCESS;
}
```

# lcgeo/k4geo issue : DEBUG

```

auto staveDetElement = m_volumeManager.lookupDetElement( pCaloHit->getCellID0() );
dd4hep::Position local1(0.0, 0.0, 0.0);
dd4hep::Position local2(normalVector[0],normalVector[1],normalVector[2]);
dd4hep::Position global1(0.0, 0.0, 0.0);
dd4hep::Position global2(0.0, 0.0, 0.0);
staveDetElement.nominal().localToWorld( local1, global1 );
staveDetElement.nominal().localToWorld( local2, global2 );
dd4hep::Position normal( global2-global1 );

streamlog_out(DEBUG6) << "  detelement: " << staveDetElement.name()
    << "  parent: " << staveDetElement.parent().name()
    << "  grandparent: " << staveDetElement.parent().parent().name()
    << "  cellID: " << pCaloHit->getCellID0()
    << "  PhiLoc:" << atan2( global1.y(), global1.x() )*180/M_PI
    << "  PhiNor:" << atan2( normal.y(), normal.x() )*180/M_PI
    << " normal vector "
    << std::setw(15) << normal.x()
    << std::setw(15) << normal.y()
    << std::setw(15) << normal.z()
    << std::endl;

```

# lcgeo/k4geo : output

When running Marlin the code above produces the following kind of output (hit position dependent) :

## 1. When using ILD geometry ILD\_I5\_v02

```
[ DEBUG6 "MyDDMarlinPandora"] detelement: stave5 parent: HcalBarrel grandparent: world cellID: 9704278 PhiLoc:0  
PhiNor:-180 normal vector -1 -3.33067e-16 0
```

## 2. When using ILD geometry ILD\_I2\_v02

```
[ DEBUG6 "MyDDMarlinPandora"] detelement: HcalBarrel parent: world grandparent: cellID: 64095606 PhiLoc:0  
PhiNor:0 normal vector 0 0 1
```