Task force report: RawCalorimeterHit modification

Taikan Suehara
(Kyushu University)

Task force members

- Chair (DAQ): T. Suehara
- SiW-ECAL: A. Irles
- SciCAL: J. Kvasnicka, K. Braggi, W. Ootani
- SDHCAL: G. Grenier
- Software: D. Jeans

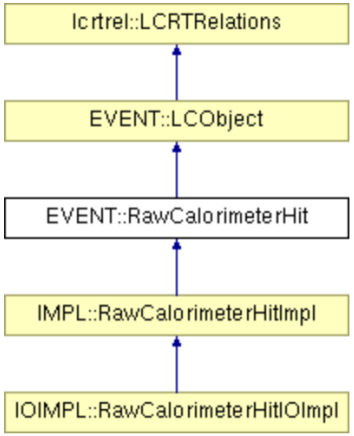
Current RawCalorimeterHit

EVENT::RawCalorimeterHit Class Reference

The generic calorimeter hit for real data (or simulation thereof). More...

#include <pre-generated/EVENT/RawCalorimeterHit.h>

Inheritance diagram for EVENT::RawCalorimeterHit:



List of all members.

Public Types

typedef RawCalorimeterHit	**lcobject_type** Useful typedef for template programming with **LCIO**.

Public Member Functions

virtual	**~RawCalorimeterHit** () Destructor.
virtual int	**getCellID0** () const =0 Returns the detector specific (geometrical) cell id.
virtual int	**getCellID1** () const =0 Returns the second detector specific (geometrical) cell id.
virtual int	**getAmplitude** () const =0 Returns the amplitude of the hit in ADC counts.
virtual int	**getTimeStamp** () const =0 Returns a time stamp for the hit.

Simple object having 4 integers

- CellID0
- CellID1
- Amplitude
- TimeStamp

Not enough?

RawCalorimeterHit for EDM4hep

https://github.com/key4hep/EDM4hep/blob/master/edm4hep.yaml#L249

```
#-------------  RawCalorimeterHit
edm4hep::RawCalorimeterHit:
  Description: "Raw calorimeter hit"
  Author : "F.Gaede, DESY"
  Members:
    - uint64_t cellID   //detector specific (geometrical) cell id.
    - int32_t amplitude              //amplitude of the hit in ADC counts.
    - int32_t timeStamp              //time stamp for the hit.
```

Possible new structure

Data	Current	Proposed	Comment
Cell ID	4 / 8 bytes	8 bytes	
Amplitude	4 bytes	4 (or 8) bytes	High/low gain Currently 4 bytes are enough
Timestamp	4 bytes	8 bytes	For both BXID and hi-reso TDC
Flag	-	4 bytes	Gain, Trig bits

Not all data bytes are needed for all subsystems

- Relying compression – study from next page
- Separate by 32-bit (or 16-bit) variables?
  - Amplitude1, 2, Timestamp1, 2

A test in EDM4hep framework

Thanks to T. Madlener

```
#-------------  RawCalorimeterHit as defined in EDM4hep
calice::EDM4hepRawCaloHit:
  Description: "Raw calorimeter hit from EDM4hep"
  Author : "F.Gaede, DESY"
  Members:
    - uint64_t cellID   //detector specific (geometrical) cell id.
    - int32_t amplitude               //amplitude of the hit in ADC counts.
    - int32_t timeStamp               //time stamp for the hit.

#-------------- RawCalorimeterHit as defined in LCIO
calice::LCIORawCaloHit:
  Description: "Raw calorimeter hit from LCIO with only a 32 bit cellID"
  Author: "F.Gaede, DESY"
  Members:
    - int32_t cellID0      // cellID0
    - int32_t amplitude    // amplitude of the hit in ADC counts
    - int32_t timeStamp    // time stamp for the hit

# -------------- RawCalorimeterHit with only 64 bit fields
calice::RawCaloHit64Bit:
  Description: "Raw calorimeter hit with 3 64 bit fields, where the last is split into 4 16 bit ones"
  Author: "T. Madlener, DESY"
  Members:
    - uint64_t cellID                   // the cellID
    - uint64_t timeStamp                // the time stamp(s) of the hit
    - std::array<uint16_t, 4> amplitude // the amplitude
```

What have been tested

- Filling random data and compare the size
  1. 32 x 3 bits structure, 32 x 3 bits random: 100%
  2. 64 x 3 bits structure, 32 x 3 bits random: 124%
  3. 64 x 3 bits structure, 64 x 3 bits random: 191%
- Another test
  1. 32 x 2 + 64 x 1 structure, fully filled: 100%
  2. 64 x 3 structure, 32 x 2 + 64 x 1 filled: 115%
- Conclusion / comment
  - Compression works, but bigger structure still have overhead
    → minimize common structure
  - Random data has worse compression
    → may have bigger overhead with real data

Proposal

Data	Current	Proposed	Comment
Cell ID	4 / 8 bytes	8 bytes	
Amplitude	4 bytes	2 x 2 bytes	High/low gain
Timestamp	4 bytes	4 x 2 bytes	For both BXID and hi-reso TDC
Misc	-	4 bytes	Gain, Trig bits
Total	12/16 bytes	24 bytes	

- Questions
  - ~25% overhead tolerated?
    - Note this is not the very raw data
  - Are we OK with 4 bytes of amplitude?
  - Do we need additional variables?

Status / Plans

- Comments from TB (today)
  - Need to check if it's consistent with Klaus
  - Maybe a usage of "3 points" capture of spectrum
    - Misc can be used for 3rd point
  - Should not wait DRD6 to be formed
- Comments from TF / collaboration
- Agreement on final structure
- Implementation
  - In LCIO
  - In Edm4hep