# Conversion from GEANT4 hits into EDM hits

@ last meeting

G. Tassielli – L. Lavezzi

## STARTING POINT

**DCH + SVX + PSHW + (CALO)**
standalone code

- Simulation: GEANT4
- Analysis: ROME

Dependencies:
- GCC 6.3.0
- ROOT 6.14.06
- GEANT4-10.4.3 with GDML
- CLHEP 2.4.0.0

**KEY4HEP STACK**
*/cvmfs/sw.hsf.org/key4hep/setup.sh*

- Simulation: GEANT4
- ~~Analysis: ROME~~

Dependencies:
- GCC 8.3.0
- ROOT 6.22.06
- GEANT4-10.4.7
- CLHEP 2.4.4.0

**WHAT HAPPENED**

- At the beginning there were attempts to find a match between the two setups
- It was decided to adopt to setup everything to depend **only on kay4hep stack**
- The compilation was moved **from Makefile to Cmake** → authomatic find(packages)
- Now **the simulation of the standalone compiles under key4hep** stack
- Everything is on git:
  https://github.com/lialavezzi/IDEA/tree/master/DriftChamberPLUSVertex

**NEXT TO DO**

- Actual conversion of the data model

- **THANKS** to Iacopo Vivarelli for the help!

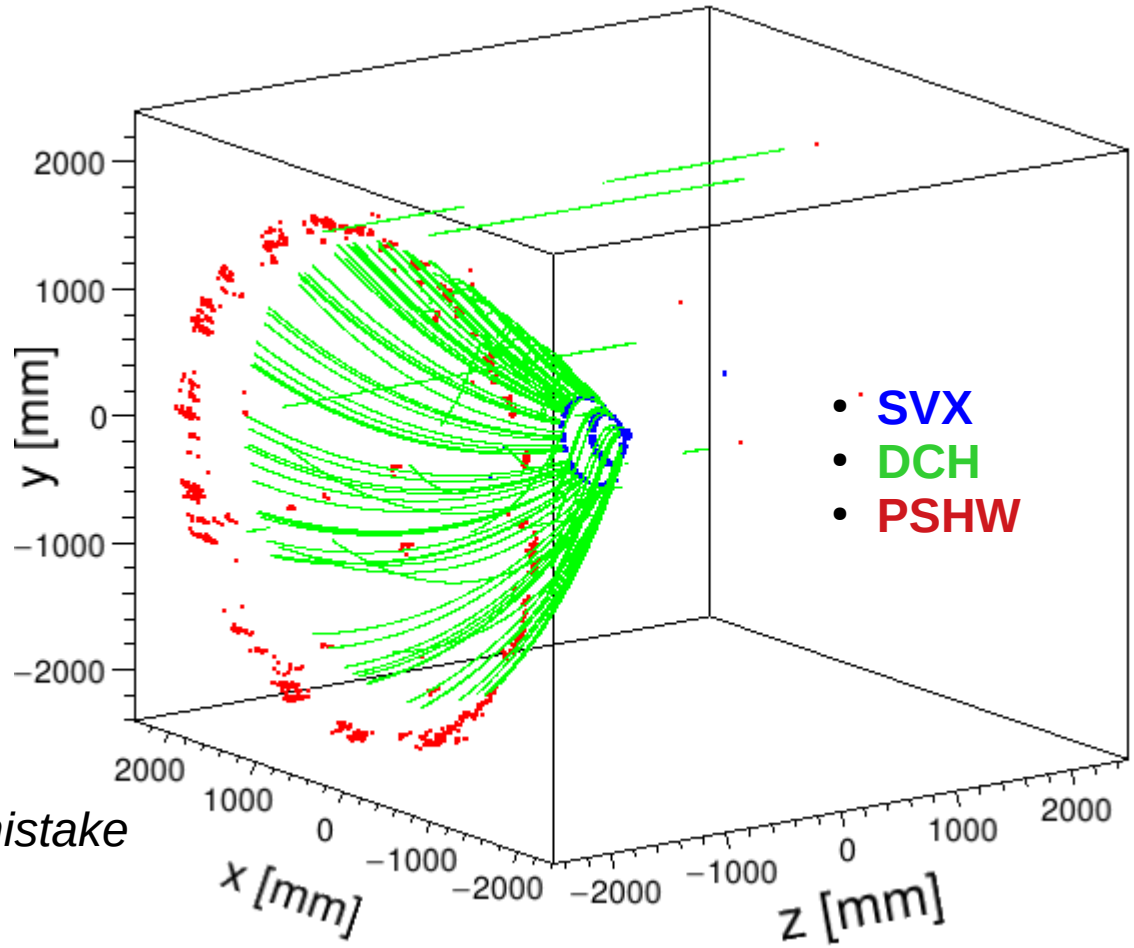
## ACTUAL CONVERSION

- the class **convertHits** has been written by duplicating the existing class **readHits**
- **readHits** translates the GEANT4 hits to the format useful as input for ROME analysis
- **convertHits** takes GEANT4 hits as well and translates them to EDM4Hep model
- it is compiled via CMake, hence on the FCC stack @ lxplus
- the EDM4Hep hit is at the moment the SimTrackerHit defined in: https://github.com/key4hep/EDM4hep/blob/master/edm4hep.yaml

**Example of simulation**

**particle**
- 100 events
- 1 electron/event
- theta in [44.5, 45.5] deg
- energy = 1 GeV

**geometry**
- Beam pipe
- SVX
- DCH
- PSHW
- *No CALO yet (sorry, I made a mistake last time saing it was there!)*
- magnetic field = 2.0 T



# OLD AND NEW HIT MODEL

**GMCG4TrackerHit**
*original* GEANT4 *hit*

```
G4int          fTrackID;
G4int          fChamberNb;
G4int          fChannelNb;
G4double       fEdep;
G4double       fNoIEdep;
G4double       fGlobalTime;
G4double       fProperTime;
G4ThreeVector  fPos;
G4ThreeVector  fPosEnding;
G4ThreeVector  fMomentum;
G4double       fStepLength;
G4String     fProcessCode;
```

**SimTrackerHit**
EDM4Hep tracker hit

```
#---------- SimTrackerHit
edm4hep::SimTrackerHit:
  Description: "Simulated tracker hit"
  Author : "F.Gaede, DESY"
  Members:
    - unsigned long long cellID      //ID of the sensor that created this hit
    - float EDep                     //energy deposited in the hit [GeV].
    - float time                     //proper time of the hit in the lab frame in [ns].
    - float pathLength               //path length of the particle in the sensitive material that res
    - int   quality                  //quality bit flag.
    - edm4hep::Vector3d position     //the hit position in [mm].
    - edm4hep::Vector3f momentum     //the 3-momentum of the particle at the hits position in [GeV]
  OneToOneRelations:
    - edm4hep::MCParticle MCParticle //MCParticle that caused the hit.

  ExtraCode :
    includes: "#include <cmath>"
    declaration: "
    int  set_bit(int val, int num, bool bitval){ return (val & ~(1<<num)) | (bitval << num); }\n
    void setOverlay(bool val) { setQuality( set_bit( getQuality() , BITOverlay , val ) ) ;   }\n
    void setProducedBySecondary(bool val) { setQuality( set_bit( getQuality() , BITProducedBySecondary , val ) ) ;   }
    "
```

- There are some variables in the GMCG4TrackerHit which are not foreseen in SimTrackerHit

# OLD AND NEW HIT MODEL

**GMCG4TrackerHit**
*original* GEANT4 *hit*

```
G4int          fTrackID;
G4int          fChamberNb;
G4int          fChannelNb;
G4double       fEdep;
G4double       fNoIEdep;
G4double       fGlobalTime;
G4double       fProperTime;
G4ThreeVector  fPos;
G4ThreeVector  fPosEnding;
G4ThreeVector  fMomentum;
G4double       fStepLength;
G4String       fProcessCode;
```

**SimTrackerHit**
EDM4Hep tracker hit

```
#---------- SimTrackerHit
edm4hep::SimTrackerHit:
  Description: "Simulated tracker hit"
  Author : "F.Gaede, DESY"
  Members:
    - unsigned long long cellID      //ID of the sensor that created this hit
    - float EDep                     //energy deposited in the hit [GeV].
    - float time                     //proper time of the hit in the lab frame in [ns].
    - float pathLength               //path length of the particle in the sensitive material that res
    - int   quality                  //quality bit flag.
    - edm4hep::Vector3d position     //the hit position in [mm].
    - edm4hep::Vector3f momentum     //the 3-momentum of the particle at the hits position in [GeV]
  OneToOneRelations:
    - edm4hep::MCParticle MCParticle //MCParticle that caused the hit.

  ExtraCode :
    includes: "#include <cmath>"
    declaration: "
    int  set_bit(int val, int num, bool bitval){ return (val & ~(1<<num)) | (bitval << num); }\n
    void setOverlay(bool val) { setQuality( set_bit( getQuality() , BITOverlay , val ) ) ;   }\n
    void setProducedBySecondary(bool val) { setQuality( set_bit( getQuality() , BITProducedBySecondary , val ) ) ;   }
    "
```

- There are some variables in the GMCG4TrackerHit which are not foreseen in SimTrackerHit
- e.g.:
  - to identify the element which is firing two integers are used instead of one
  - the ProcessCode string is used to identify the type of process in the step

**How to solve this?**

# HOW TO “SOLVE” THE HIIT

**GMCG4TrackerHit**
*original* GEANT4 *hit*

```
G4int           fTrackID;
G4int           fChamberNb;
G4int           fChannelNb;
G4double        fEdep;
G4double        fNoIEdep;
G4double        fGlobalTime;
G4double        fProperTime;
G4ThreeVector   fPos;
G4ThreeVector   fPosEnding;
G4ThreeVector   fMomentum;
G4double        fStepLength;
G4String     fProcessCode;
```

- Sent an email to Gerri Ganis and Clement Helsens
  ... who CC-ed Valentin Volkl and Thomas Madlener (EDM4Hep experts)
  … who CC-ed Frank Gaede and Andre Sailer (EDMHep experts)
  → started a discussion:
- My idea was to define a custom sim hit for the IDEA trackers
- Their idea is to widen the SimTrackerHit definition, since EDM4Hep is still In progress, so they are open to changes if it is needed
- They suggested to define an index containing chamberNb and channelNb together: is it possible?
- Can the ProcessCode be an integer instead of a string?
- A presentation/discussion of the case was requested for next EDM4Hep online meeting on 6th April
  https://indico.cern.ch/category/11461/
- I accepted to present this, but we should decide what information are needed and what to ask for/discuss

# DO WE WANT TO CONVERT THE TRACKS?

If we decide to translate to EDM4Hep also the tracks we have to match also the descriptions in the standalone and in EDM4Hep and compile possible requests for the EDM4Hep meeting

Moreover we might have to port the compilation of ROME to the FCC stack as we did for the simulation with GEANT4

***ROME reconstructed track***
*analyzer/GMC/xml/RecoTracks.xml*

**EDM4Hep reconstructed track**
*edm4hep.yaml*

| | | |
|---|---|---|
| Int_t | TrkID | Track ID |
| Double_t | x0,y0,z0 | Track Vertex x,y,z coord |
| Double_t | err_x0, err_y0,err_z0 | Error On the Track Vertex x,y,z coord |
| Double_t | theta | polar angle w.r.t. the beam direction |
| Double_t | err_theta | Error on the polar angle w.r.t. the beam direction |
| Double_t | phi | azimuthal angle in the X-Y plane |
| Double_t | err_phi | Error on the azimuthal angle in the X-Y plane |
| Double_t | Momentum | Track Momentum |
| Double_t | Err_Momentum | Error on Track Momentum |
| TVector3 | mom | Fitted Track Momentum |
| TMatrixDSym | cov | Covariant Matrix of fitted Tracks |
| Int_t | hitindex | Index of the hit in DCHHitRecResult folder |
| Int_t | detid | detector id of hit: 0 - DCH, 1 - SVX |
| Bool_t | Skipped | Flag for hits removed from the fit |
| TVector3 | StateVector | State vectors of the track |
| Int_t | nhits | number of hits |
| Int_t | ngoodhits | number of good hits |
| Int_t | nhitsdch | number of DCH hits |
| Int_t | ngoodhitsdch | number of good DCH hits |
| Int_t | nhitssvx | number of SVX hits |
| Int_t | ngoodhitssvx | number of good SVX hits |
| Int_t | nhitspshw | number of PSHW hits |
| Int_t | ngoodhitspshw | number of good PSHW hits |
| Double_t | chi2 | chi2 of the track |
| Int_t | dof | Degrees of freedom |
| Bool_t | IsFitted | Flag indicating if the track has been already fitted |

```
#-------- Track
 edm4hep::Track:
   Description: "Reconstructed track"
   Author : "F.Gaede, DESY"
   Members:
     - int type                        //flagword that defines the type of track.Bits 16-31 are used internally
     - float chi2                      //Chi^2 of the track fit
     - int ndf                         //number of degrees of freedom of the track fit
     - float dEdx                      //dEdx of the track.
     - float dEdxError                 //error of dEdx.
     - float radiusOfInnermostHit      //radius of the innermost hit that has been used in the track fit
   VectorMembers:
     - int subDetectorHitNumbers       //number of hits in particular subdetectors.Check/set collection variable TrackSubdetectorNames for decoding the indice
     - edm4hep::TrackState trackStates //track states
   OneToManyRelations:
     - edm4hep::TrackerHit trackerHits //hits that have been used to create this track
     - edm4hep::Track tracks           //tracks (segments) that have been combined to create this track
```