# The Virtual Monte-Carlo, status and applications

R.Brun[1], F.Carminati[1], A.Fasso[2], E.Futo[1], A.Gheata[1,3], M.Gheata[3], P.Hristov[1], I.Hrivnacova[4], A.Morsch[1]

1 – CERN, 2 – SLAC, 3 – ISS Bucharest, 4 – IPN Orsay

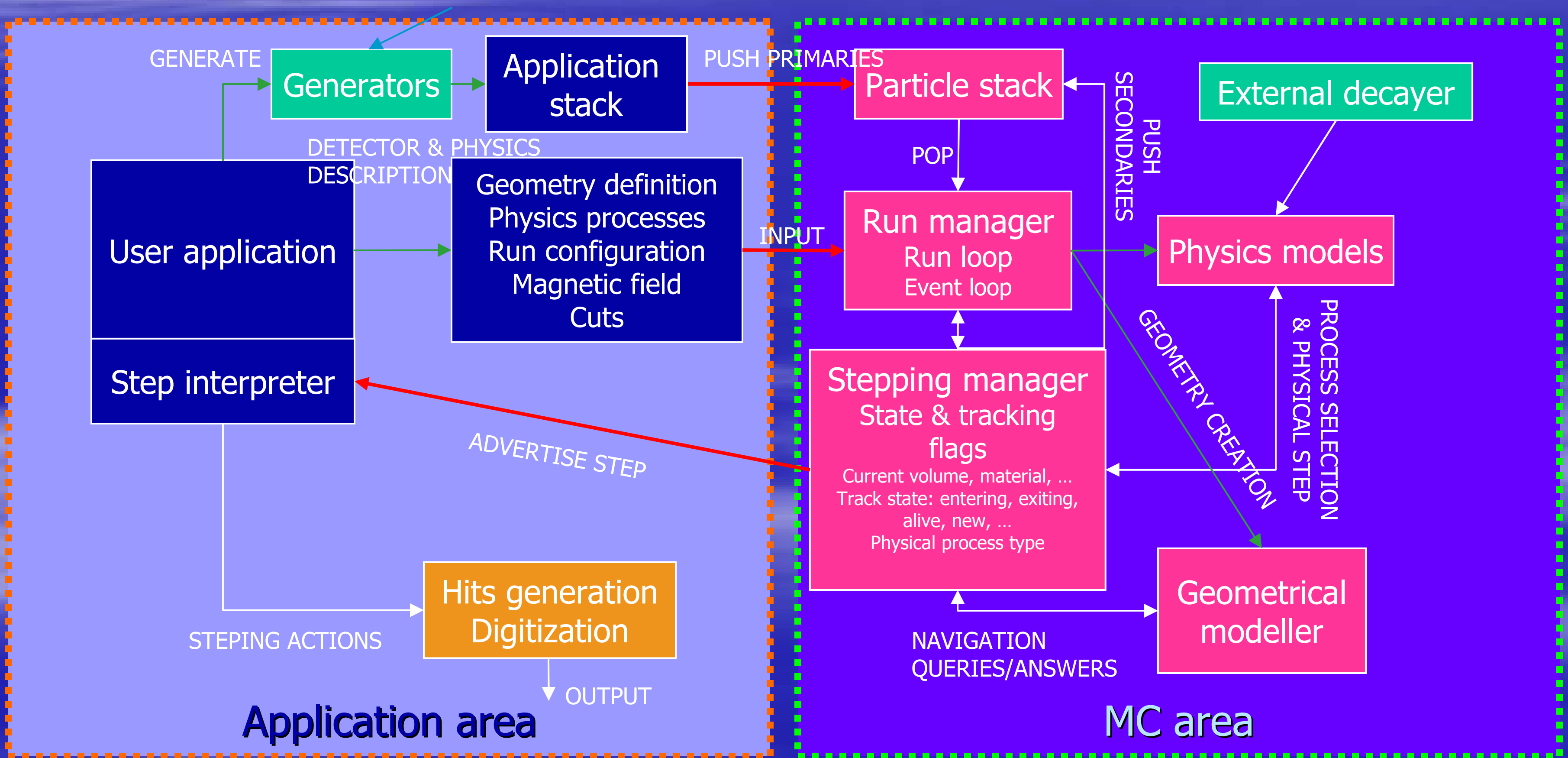
Presented by: A.Gheata

# Outline

- Concept and current structure
- Geometry and VMC
- Status of the interfaces to specific MC's
- Validation tests

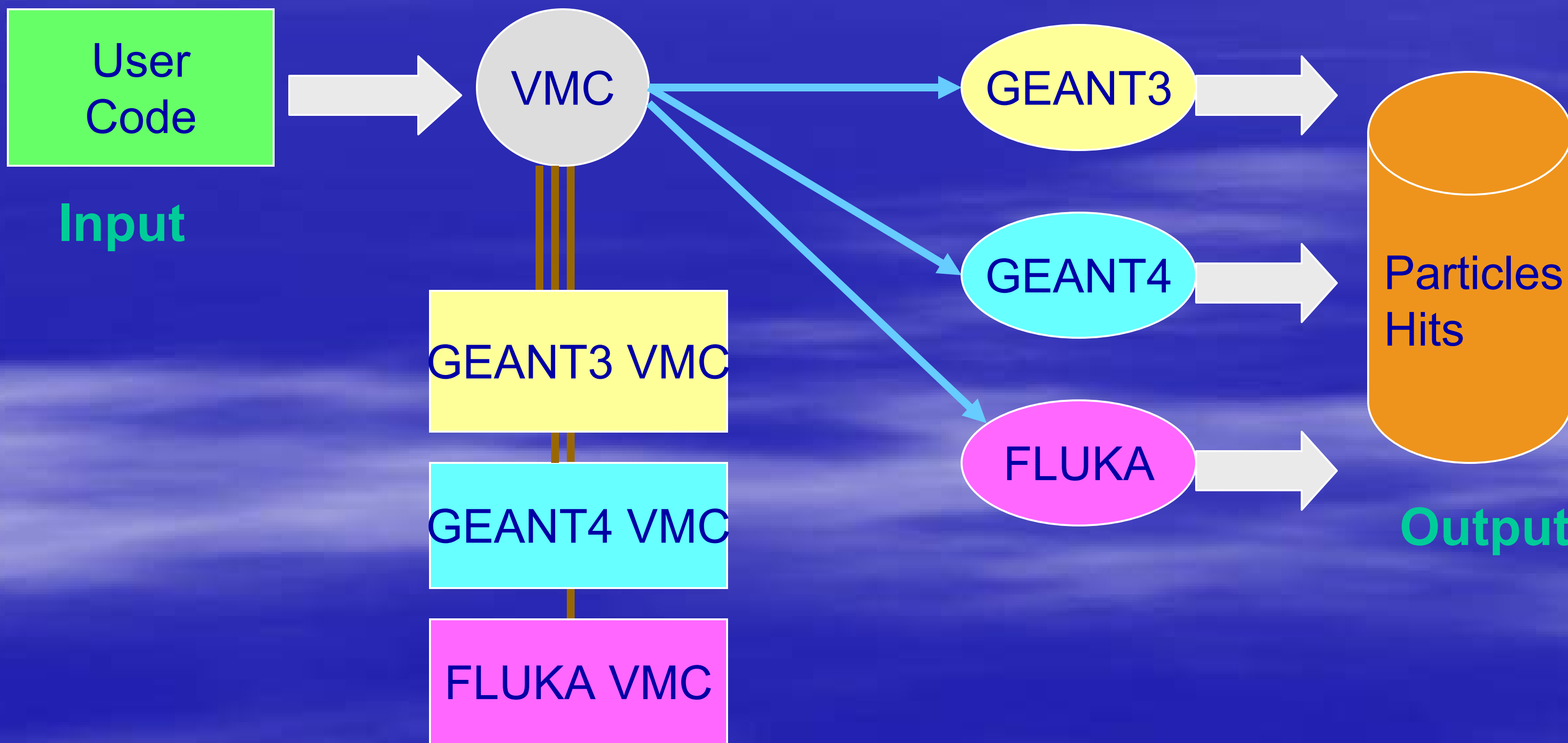
# A classical transport MC application

- Experiment/MC – tailored application



# VMC concept

- Transport MC transparent to the user application (more details in the VMC presentation at CHEP03)

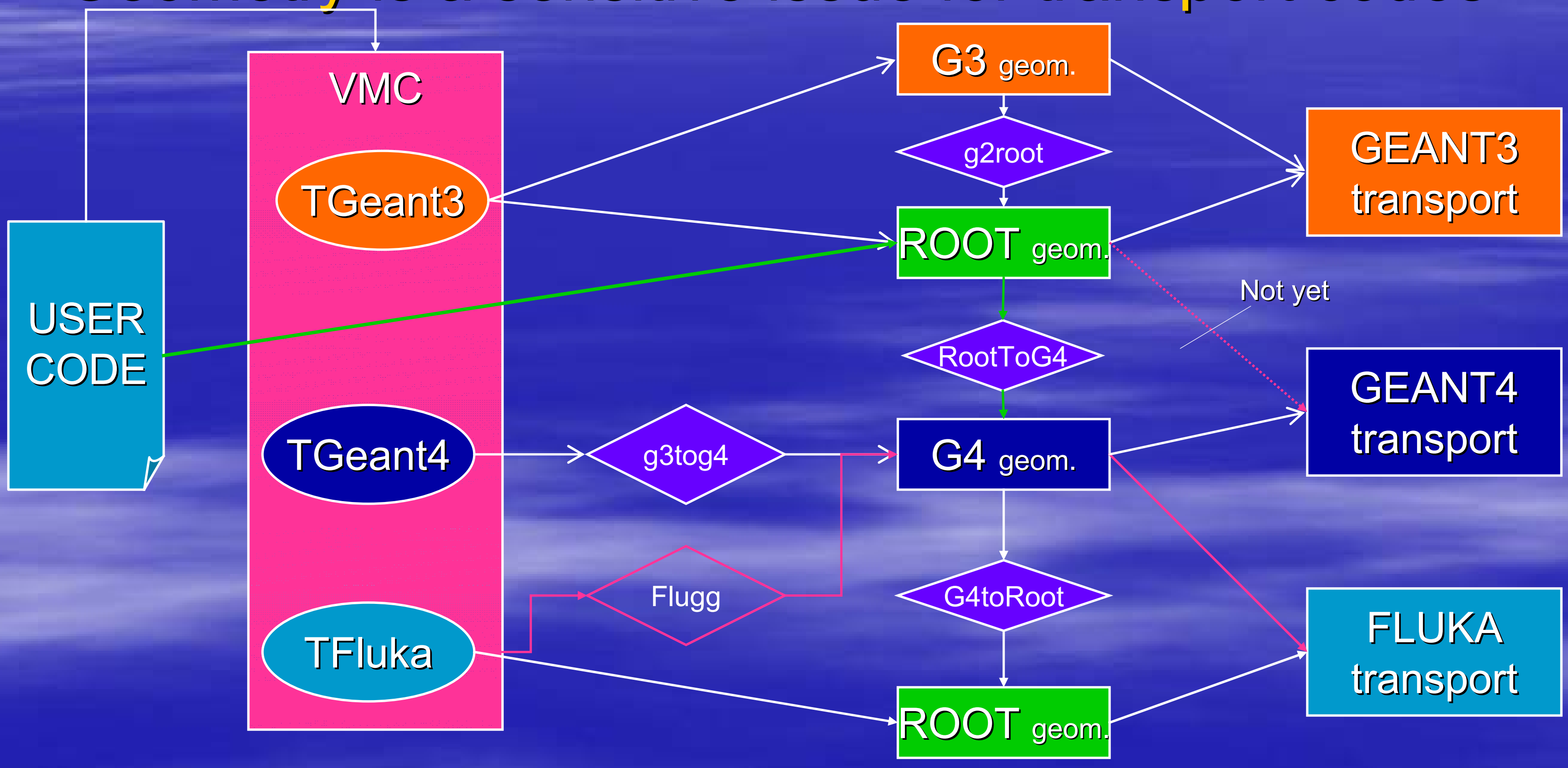


# Benefits

- Running different transport MC's from the same application
- Using the same geometry description
  - Even the same geometry engine – TGeo (not yet the case for G4)
- No need to customize detector response, physics configuration and cuts (partially) or input/output according the transport code

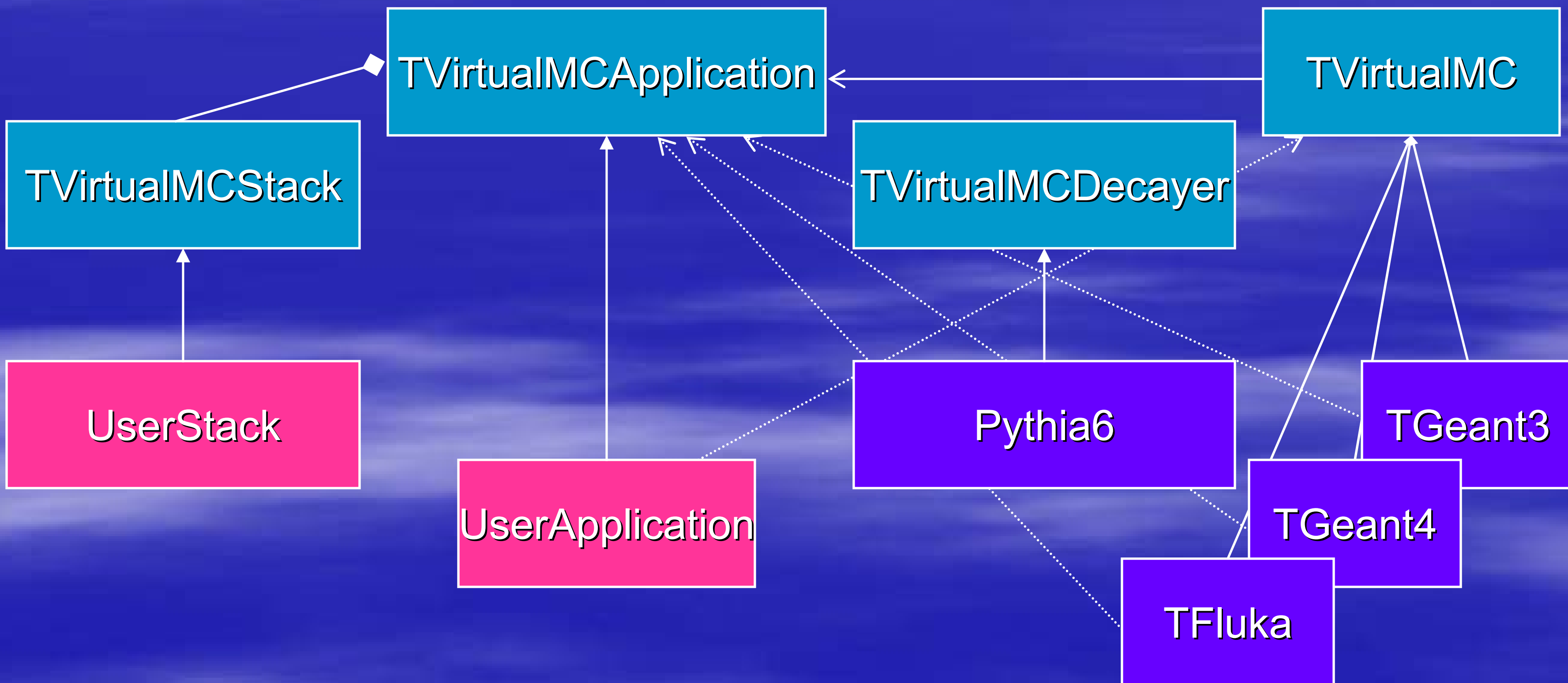
# Geometry description and representation

- Geometry is a sensitive issue for transport codes



# Structure of VMC

- Communication allowed only via virtual interface



# Status

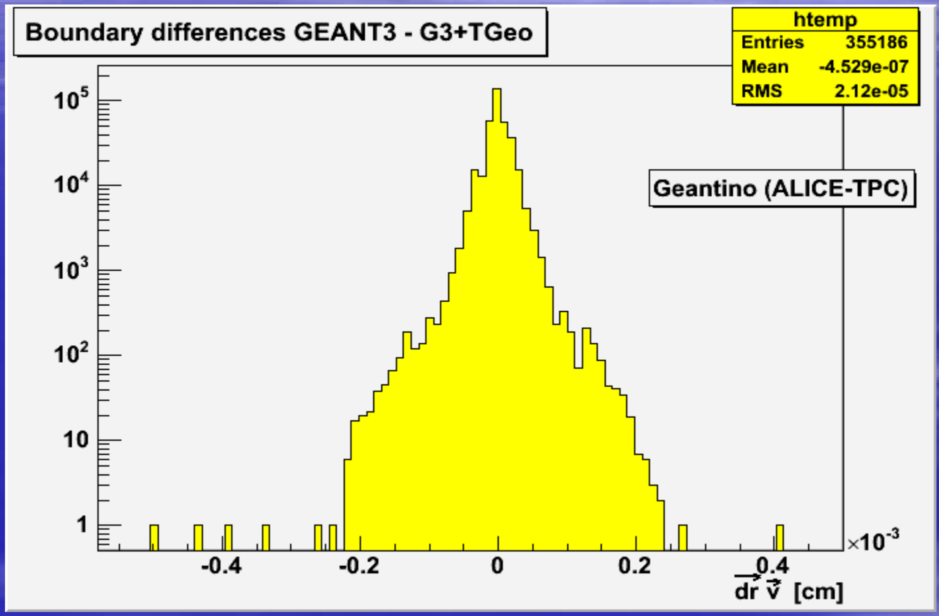
- Consolidation of old interfaces (GEANT3 VMC, GEANT4 VMC) and further development of the new one (FLUKA VMC)
- Additions (mostly user demands):
  - Added new functions to enable a user to define ions and own particles
    - TVirtualMC::DefineIon(…)
    - TVirtualMC::DefineParticle(...)
    - TVirtualMCApplication::AddParticles()
  - Added function to enable a user to abort run
    - TVirtualMC::StopRun()
  - Other changes
    - more meaningful names in TVirtualMCStack functions
    - added Bool return value to TVirtualMC functions which could fail:
      - SetCut(), SetProcess(), DefineParticle(), DefineIon()
- No major changes in the user interface
  - Most efforts directed during last year to development of TFluka
  - Geometry converters RootToG4 and G4ToRoot (see "The Virtual Geometry Model" by I.Hrivnacova)

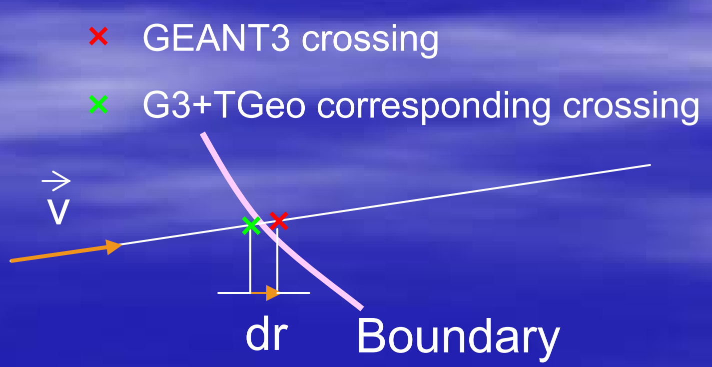
# Geant3 VMC

- In production in ALICE since quite a while
  - Implementation very stable : negligible changes since last year
  - Just few corrections in the version *TGeant3+TGeo*
- Ongoing extensive tests for TGeo vs. G3 native geometry validation
  - Version using ROOT *TGeo* navigation planned to enter production next year

# Validation of G3 with TGeo navigation

- Comparisons at hits level done in ALICE
  - No relevant differences found
    - See "A geometrical modeler for HEP" presentation at CHEP03
  - More detailed analysis at detector module level to be done
- Ongoing step-by-step comparisons
  - Already done for simple geometries
  - Geantinos shot in partial ALICE geometry
    - Differences at the expected level
    - Fine tuning of TGeo behavior when tracking close to boundaries
    - Analysis of possible rounding errors
    - Identify all possible remaining hidden problems in the interface or in TGeo itself
- Feed-back from other people that already started this kind of comparisons is welcome !



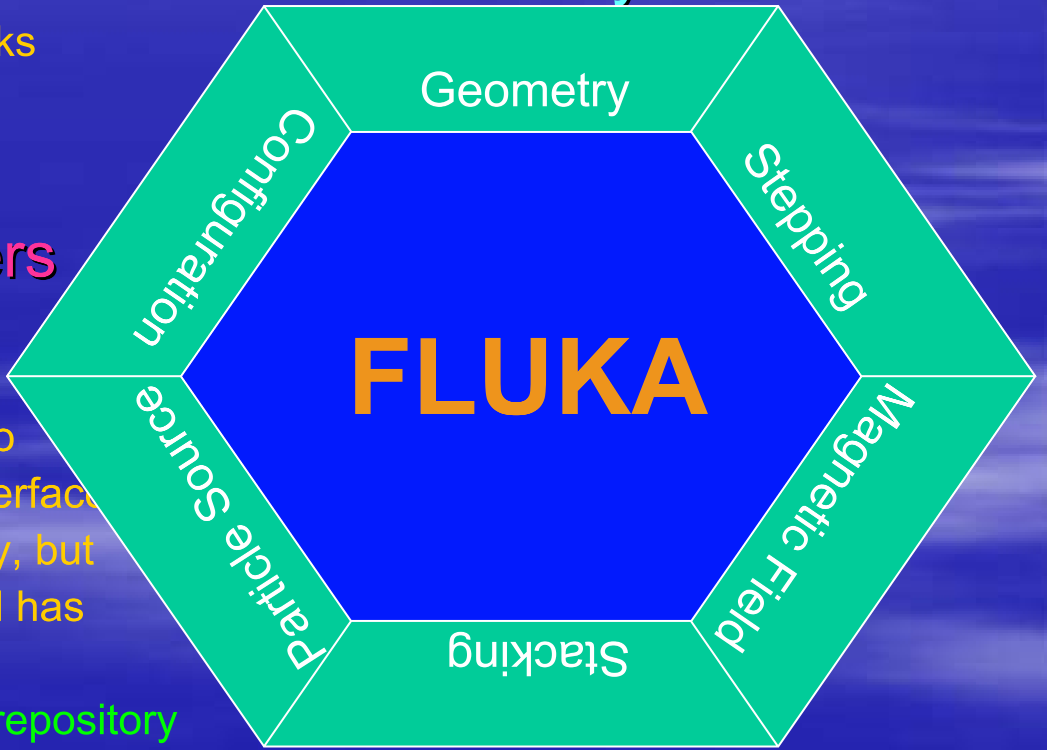



# Geant4 VMC

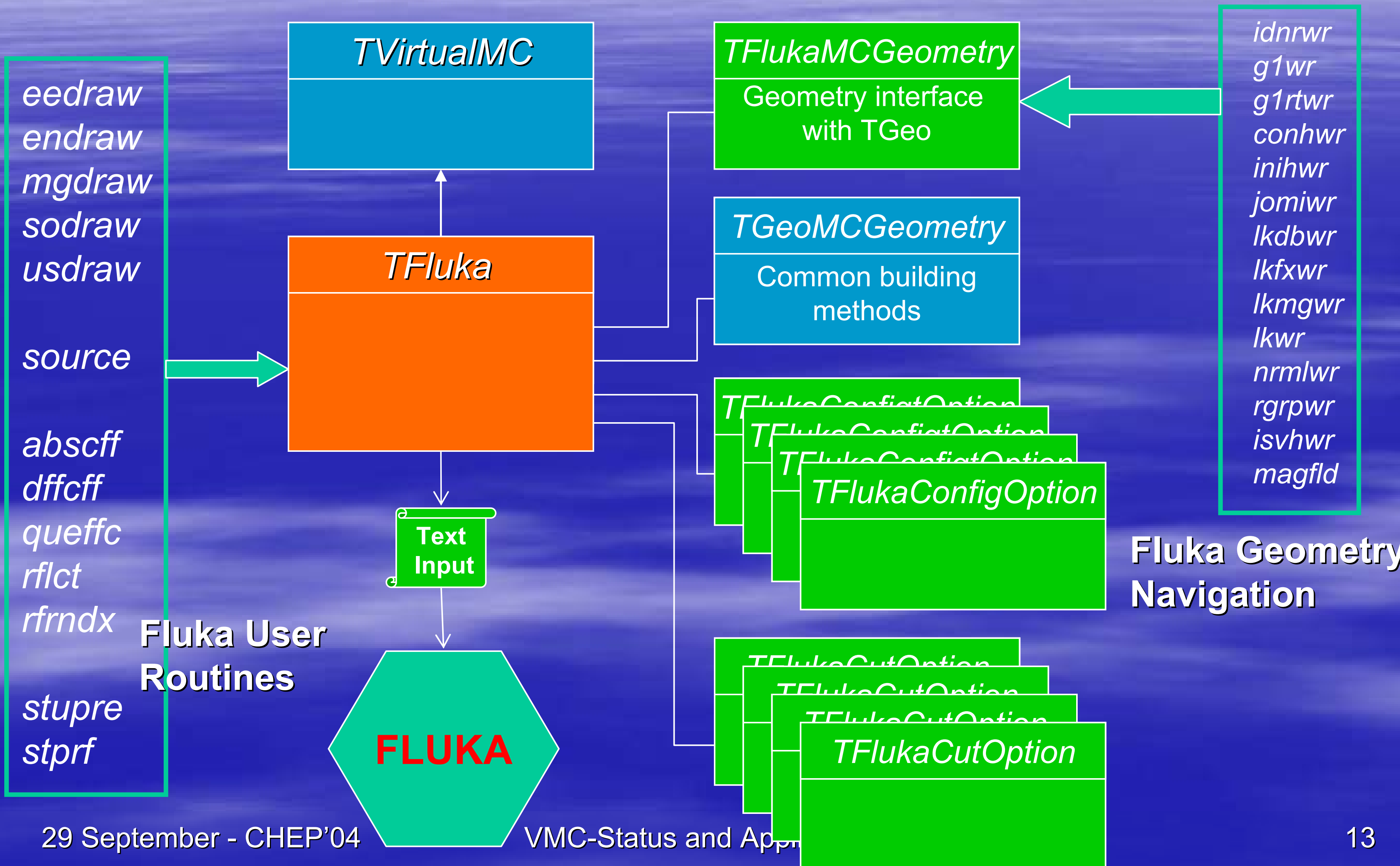
- In maintenance:
  - Updates for new Geant4 releases
  - Tagged with each Root/Geant4 release that requires backward-incompatible changes
- New features :
  - User-defined physics list enabled
  - Implements all new methods added in TVirtualMC
  - Consolidation of roottog4 converter:
    - support for positions with reflection
    - support for composite shapes
  - Extension of g4toxml converter for GDML scheme

# Fluka VMC

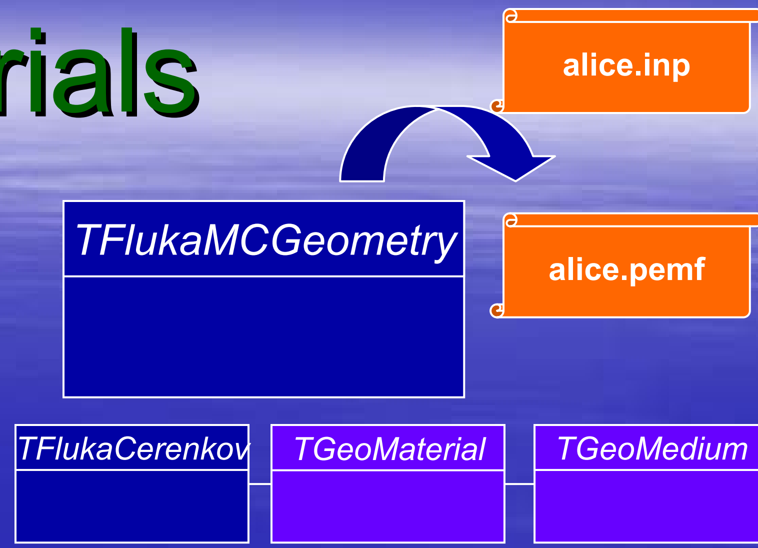
- User/Transport code interface
  - Done in traditional FORTRAN frameworks by:
    - User routines / common blocks
    - Text files (input cards)
  - TFluka hides this communication from users
    - It follows VMC interface design
    - Major efforts done this year to develop and consolidate the interface
    - Still in *AliRoot* CVS repository, but can be separately compiled and has no dependency to ALICE code
    - To be moved in ROOT CVS repository



# Class structure



# Geometry and materials



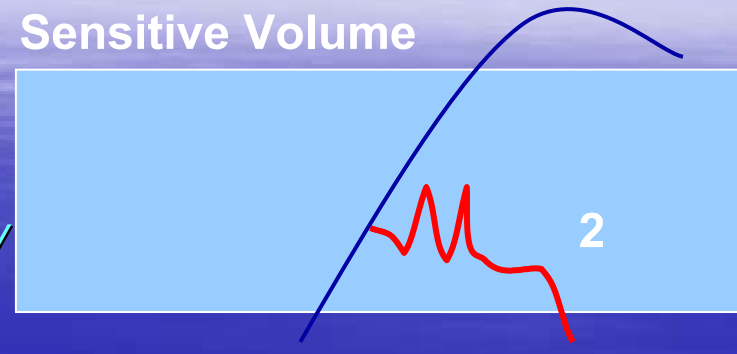

- Before : FLUGG + GEANT4
  - Unstable / many crashes very hard to debug
- Now : TGeo interface
  - Allowing navigation in full ALICE geometry
  - Preliminary validation steps done
  - Providing input “cards” for materials/regions
- Material definition and assignment via FLUKA input cards
  - MATERIAL, COMPOUND, LOW-MAT and ASSIGNMAT
  - Materials defined by effective Z not accepted
- Automatic /on-demand generation of PEMF file needed for simulation of electromagnetic processes
  - Best possible home-made solution not requiring manual intervention, but definitely not as good as fully automation of the process directly by FLUKA
- TFlukaCerenkov to store optical properties
- Magnetic field integrated to VMC scheme
  - Activation per region possible, but not yet implemented

# Configuration

- Physics and cuts configuration used via standard FLUKA formatted input file.
  - User cuts and process switches are stored at initialization stage as arrays of objects. These are finally appended to FLUKA input
- Solutions and open problems :
  - Primary particles retrieved from *TVirtualMCStack*
  - Secondaries just monitored, except particles put on stack by user code
    - Extra constraint to user stack to check for un-handled particles before tracking a new primary
  - Most switches/cuts available in VMC implemented
    - HADR=0 implemented through THRESHOLD input card
    - DCAY=0 not implemented
    - Kinetic energy cut for n, charged hadrons, muons not yet possible material-by-material

# Stepping

- Fluka user routines rewritten in C++
  - Calling *TVirtualMCApplication::Stepping to notify the step*
    - current track ID, current region, process ID, ...
  - Several routines for step notification
    - Energy deposition
    - Interaction
    - Normal step
    - Boundary crossing
    - Special events are notified to user code:
      - Double step for exiting and entering
      - Track resumed after secondary tracking
    - Special actions:
      - Cerenkov quantum efficiency handled at detection level



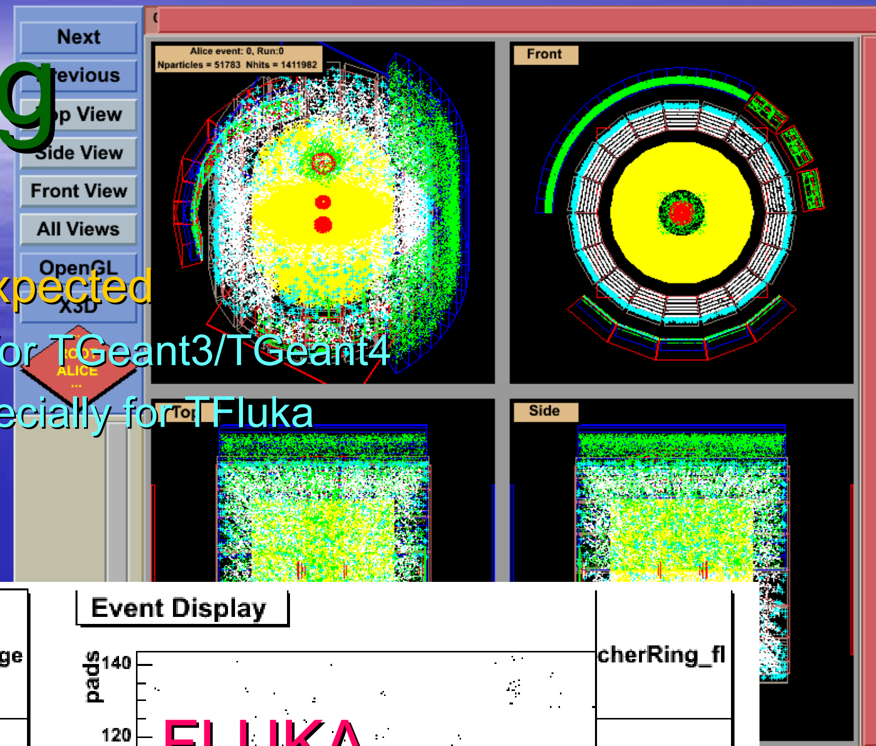

| Geant 3 | FLUKA |
|---|---|
| 1: entering | 1: entering |
| 1: exiting | 1: disappeared |
| 2: entering | 2: entering |
| 2: exiting | 2: exiting |
| | 1: entering |
| | 1:exiting |
| **2 hits** | **3 hits** |

- Special feature: different tracking sequence may influence number of hits, but not digits. Signaled anyway to user code.

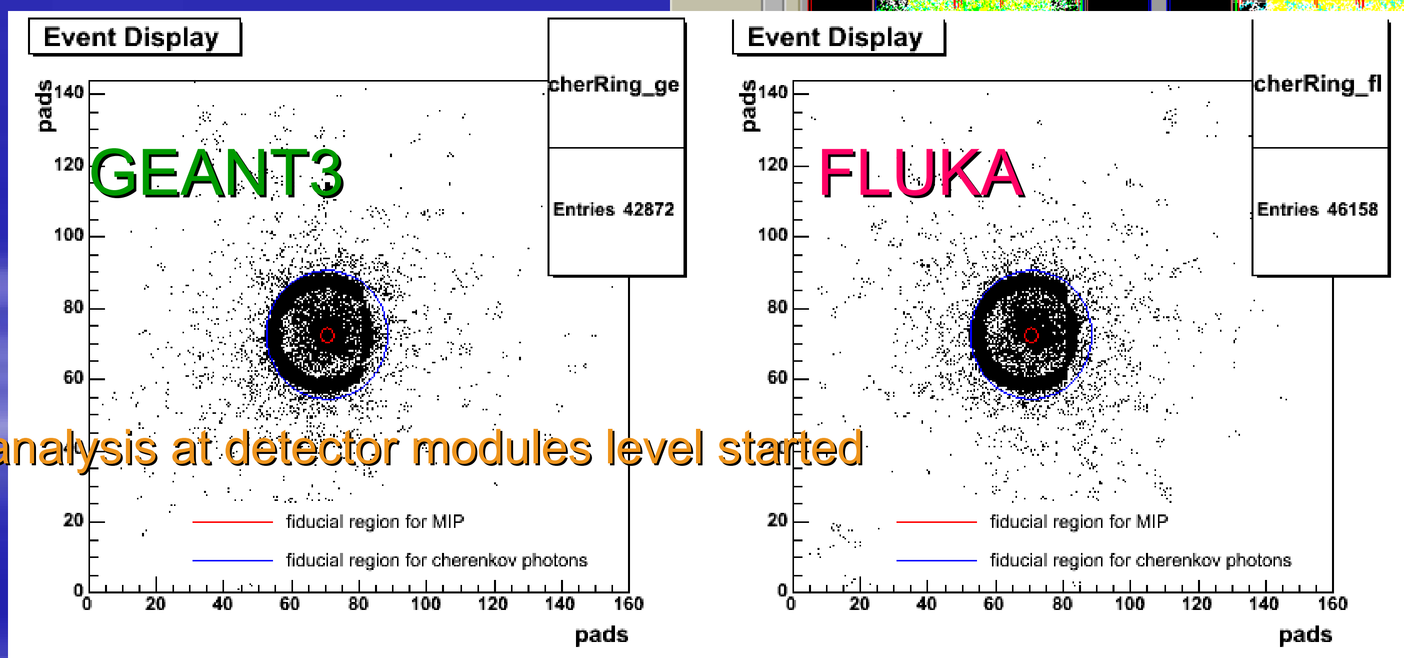
# Stepping actions, stacking

- *StopTrack()* implemented for tracks in magnetic field and Cerenkov photons
- *TVirtualMCStack* acts as observer and mirrors FLUKA private stack using "hook wrappers" for :
  - Particles from electromagnetic interactions
  - Particles from hadronic interactions
  - Cerenkov photons
  - Exception: Particles produced by user during stepping. Example: Feed-back photons
- For some processes (bremsstrahlung, delta-electrons, ...), FLUKA interrupts the tracking of the mother particle in order to follow the secondary first.
  - New method *TVirtualMC::SecondariesAreOrdered()* used to communicate order of secondaries to *TVirtualMCStack*.
- Interfacing with *TVirtualMCDecayer* not yet ready

# TFluka / G3 testing



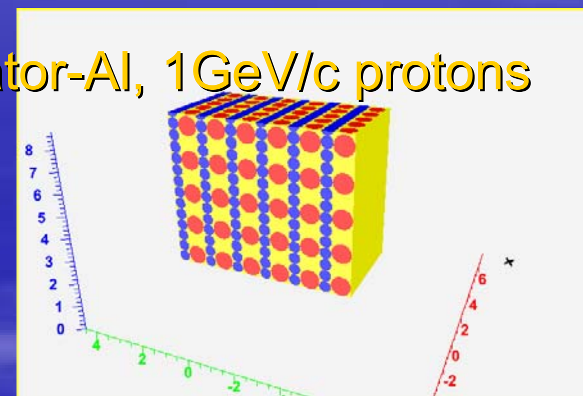

- Behavior of AliRoot running TFluka as expected
  - Hits/digits produced in the same way as for TGeant3/TGeant4
  - No changes needed in the application specially for TFluka
    - 1 exception: stack reordering

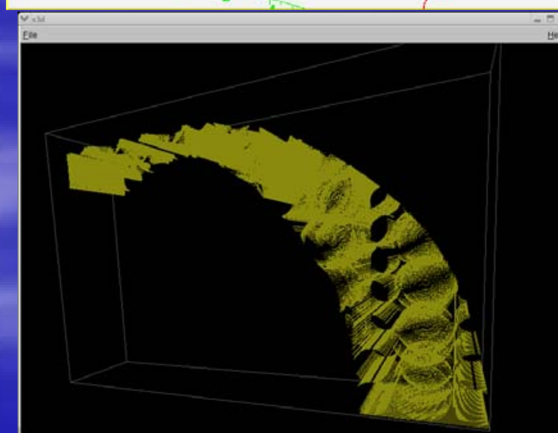



- Detailed analysis at detector modules level started

# Geometry interface validation

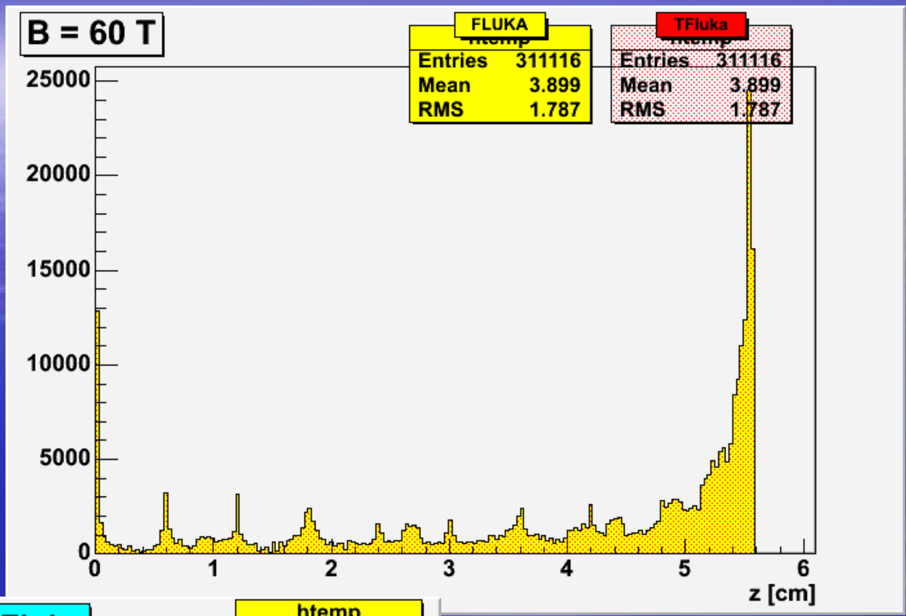
- Comparison between FLUKA native stepping versus TFluka, for a simple example
- Original setup: calorimeter sandwich Pb-Scintillator-Al, 1GeV/c protons in magnetic field
  - Vacuum put everywhere : pure geometry testing
  - Geometry reproduced identically with TGeo
    - Boolean compositions (150 components)
    - Simple application as the ones from VMC examples
    - Hits collected at boundary crossings: x,y,z, track ID, region number
- Matching procedure track-by-track
  - Computation of the distance between matching points : $dr = (\mathbf{P}_{TFluka} - \mathbf{P}_{FLUKA}) \cdot \mathbf{v}$

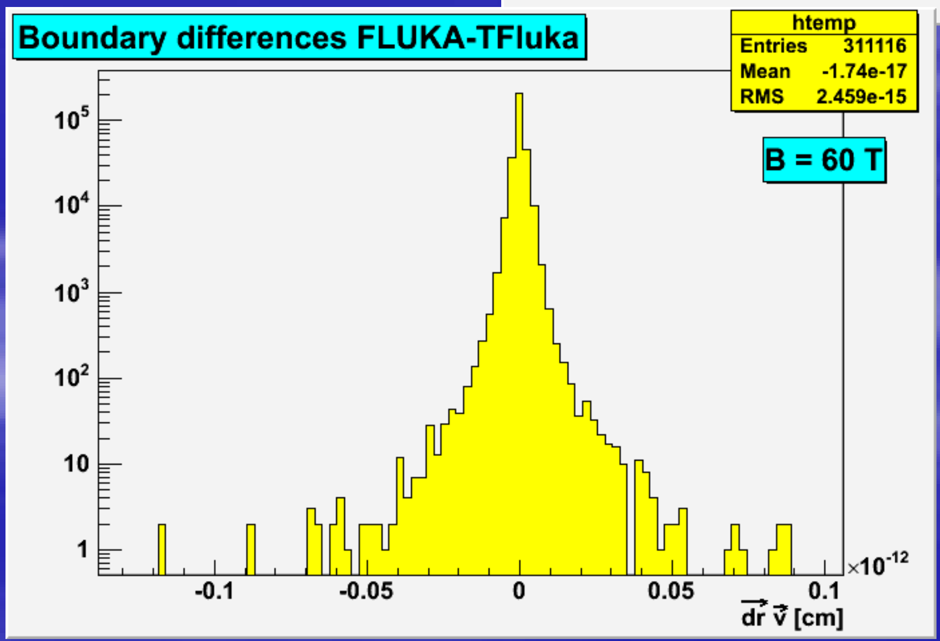




# Results

- All crossing points matching
  - Same boundary sequence
- No magnetic field
- B = 60 T
  - Helped fixing few bugs first
- Very encouraging since differences are at double-precision error level as expected



# Conclusions

- Most efforts this year focused to the Fluka VMC part
  - Geometry interface with TGeo modeller
    - Tests done against FLUKA native geometry
  - Several developments related to configuration settings, materials, stepping and stacking
  - Ready for detailed validation
  - To be moved to ROOT CVS as all other interfaces
- New features developed on user requests
- Geometry converters ROOT-G4 developed
  - The only alternative at this moment for ROOT-based geometry descriptions until a solution will be found for G4+TGeo navigation
- Continuous maintenance work