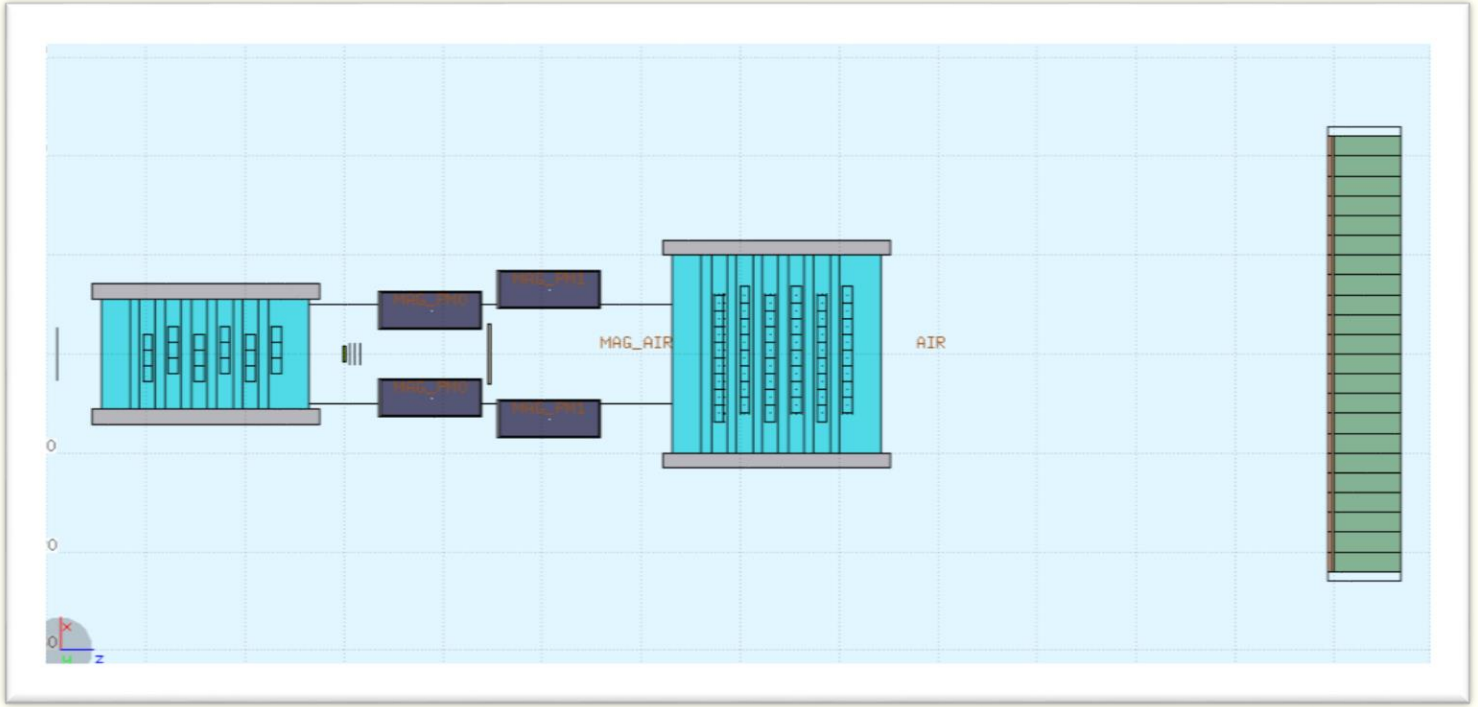
SIMULATION STATUS

FOOT Collaboration Meeting
Napoli, 25-26 May 2017

S.M. Valle and G. Battistoni



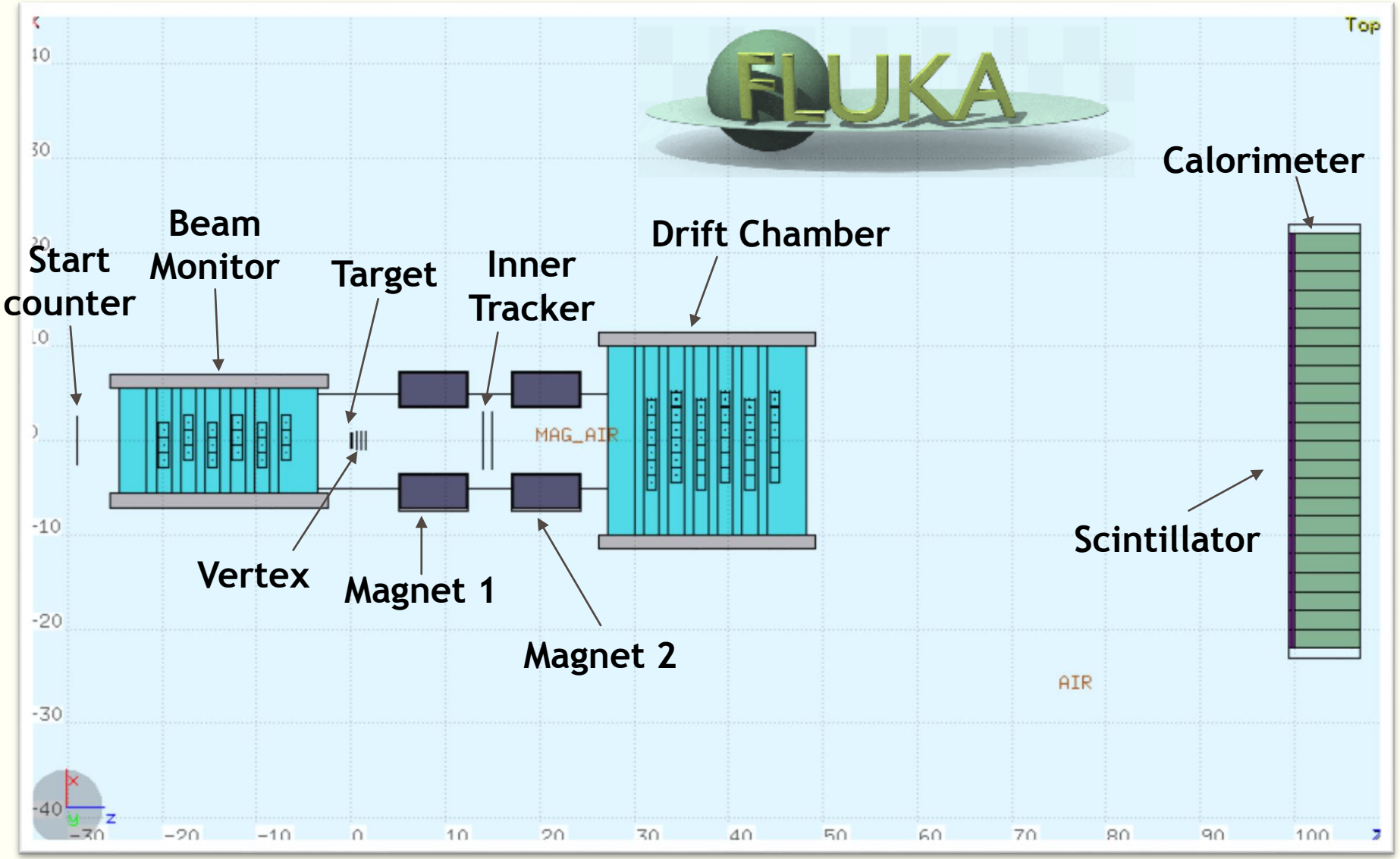
Current status of the FOOT FLUKA simulation

- New geometry **versions numbering**
- Some **geometry changes**
- New (approximated) **magnetic field map**
- **Improved tracking** in magnetic field
- **Emulsions FLUKA simulations** are fully handled by the Napoli team *(see Adele's talk)*

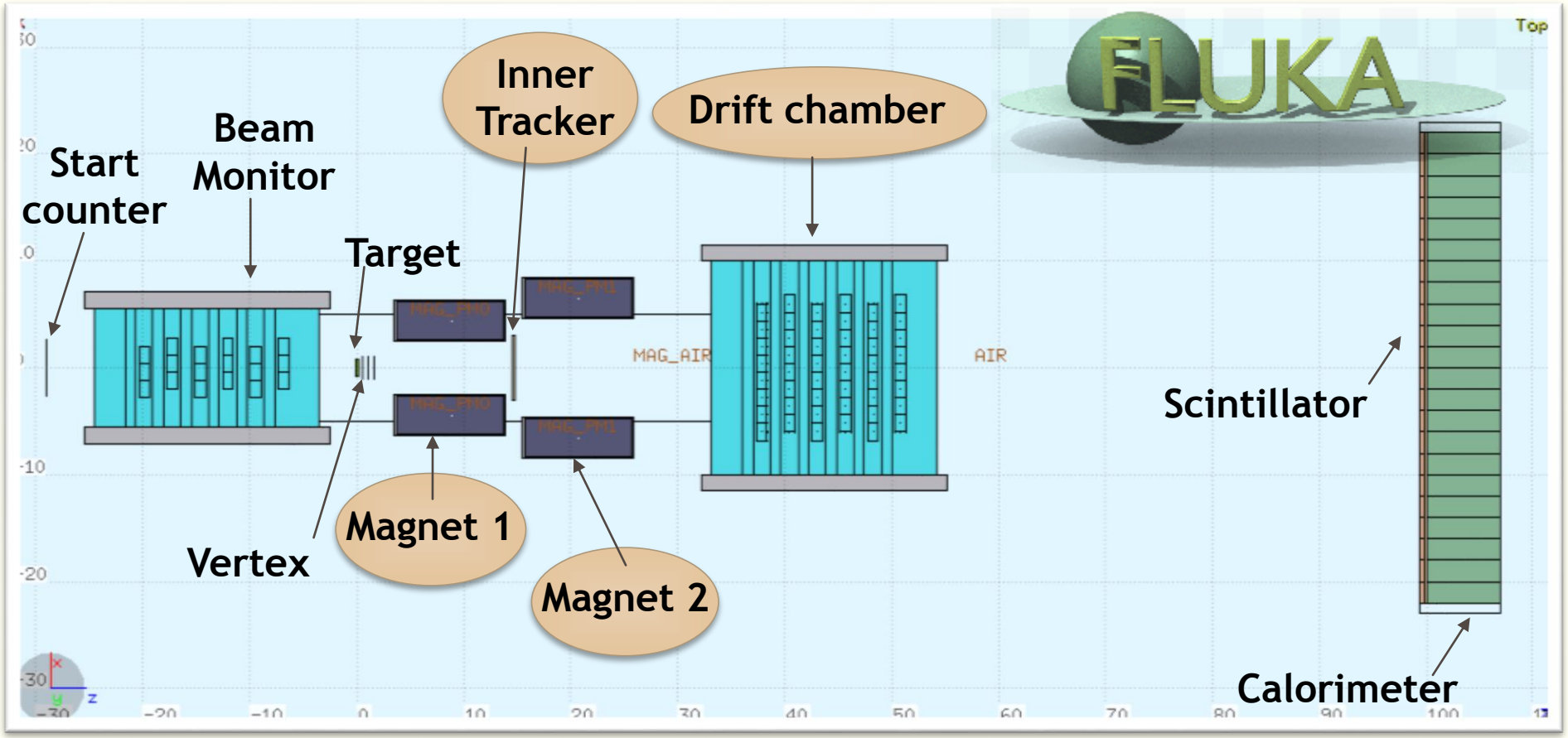
The new simulations have been used for studies on:

- **Inverse kinematics** and **target thickness** *(see Antonia's talk)*
  Global Tracking and **Momentum reconstruction** *(see Matteo's talk)*
- **Calorimeter** performances *(see Michela's talk)*
- **Beam Monitor** performances
- Design of the **Drift Chamber** *(see Francesco's talk)*
- Detector expected performances on **A and Z identification** *(see Roberto's talk)*

30 Nov 2016 Geometry

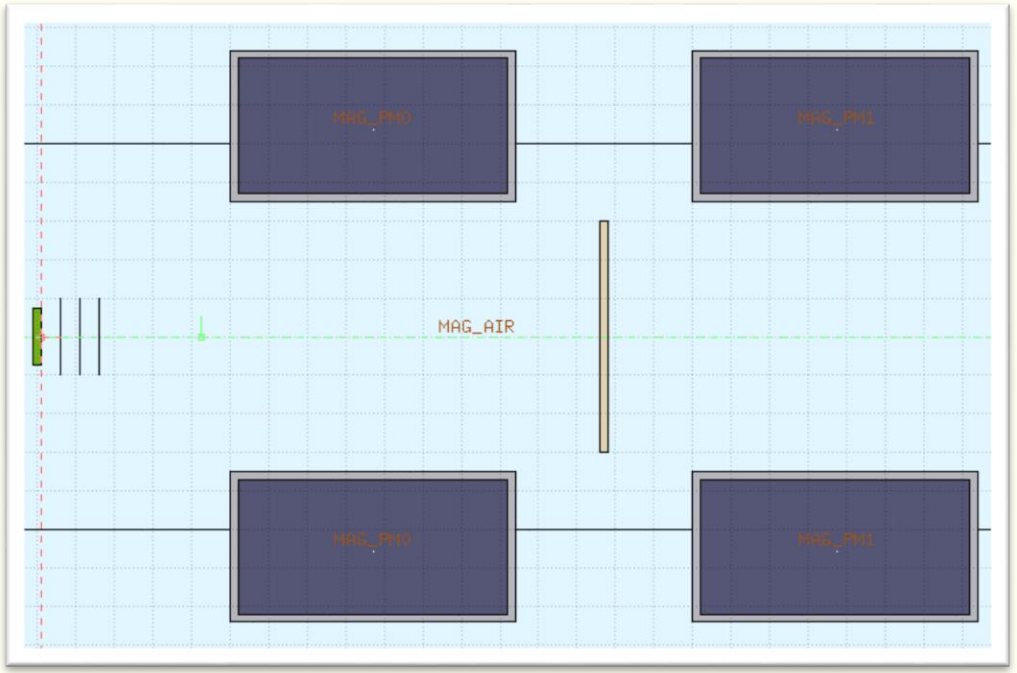


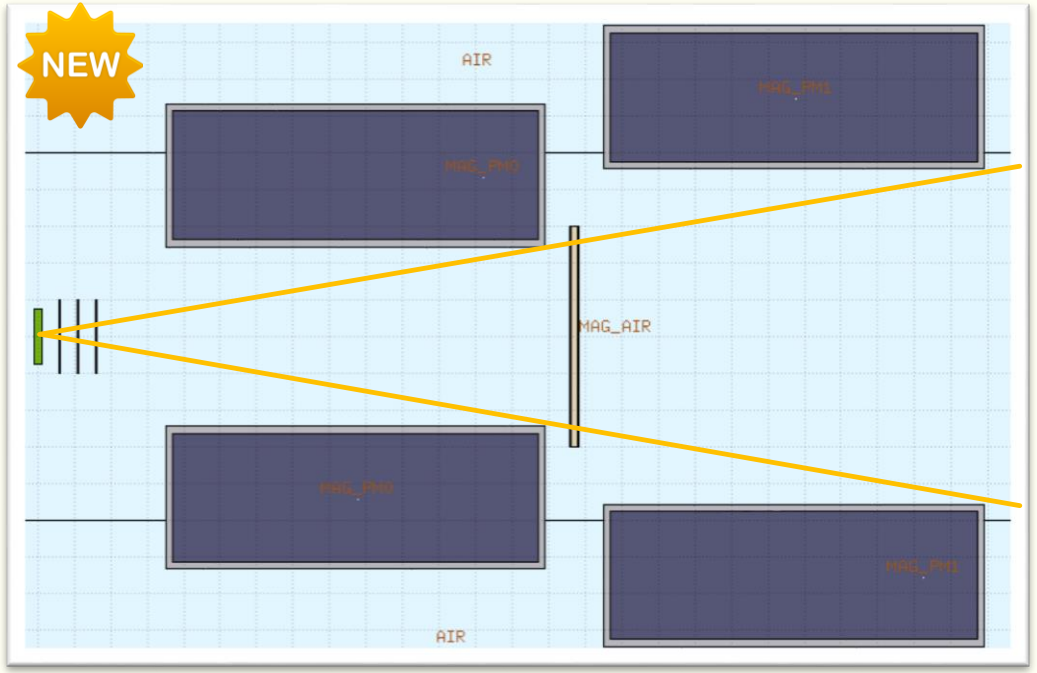
Back to the Present: V10.2 Geometry



Geometry widely tested in the past few months

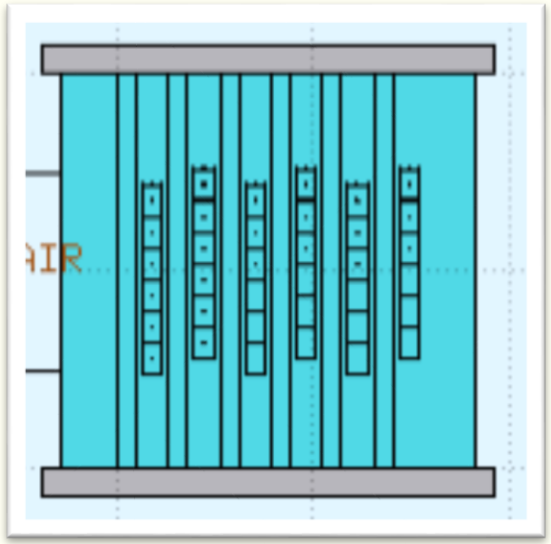
V10.2 - Magnets





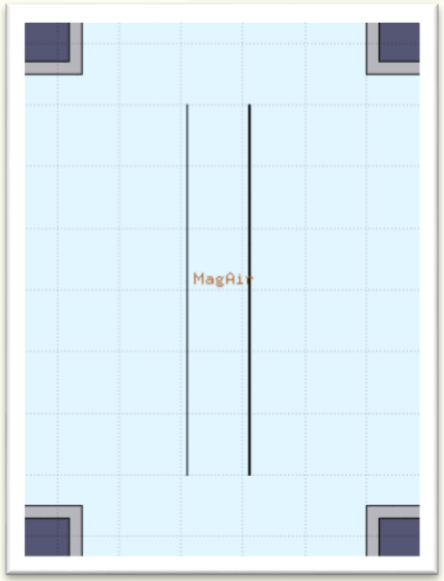
- **Increased longitudinal dimension** from 7cm to 10cm to increase the magnetic region and improve momentum reconstruction accuracy
- Magnet1 and Magnet2 have **different gap dimensions** to cover a 10° solid angle from target (R1~2.5cm and R2~4.8cm) → reduced cost?

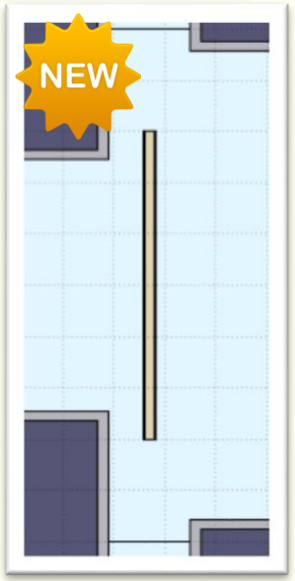
V10.2 - Drift Chamber & Inner Tracker



Drift Chamber

- Maintained cell dimensions
- **Increased cell number per layer** from 6 to 8 to cover a 10° solid angle from target
- **Reduced mylar windows thickness** to a more realistic value (100μm→25μm)

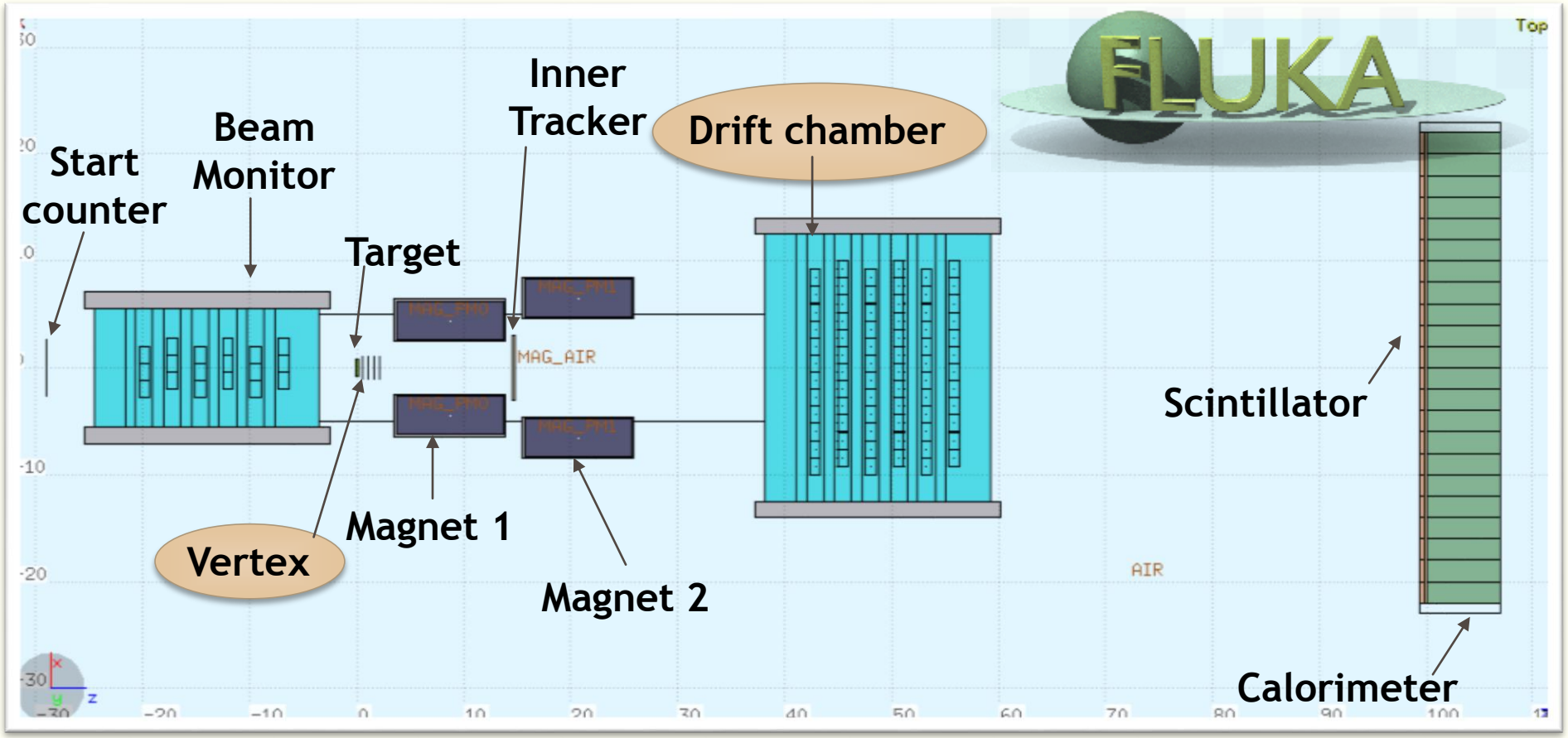




Inner Tracker

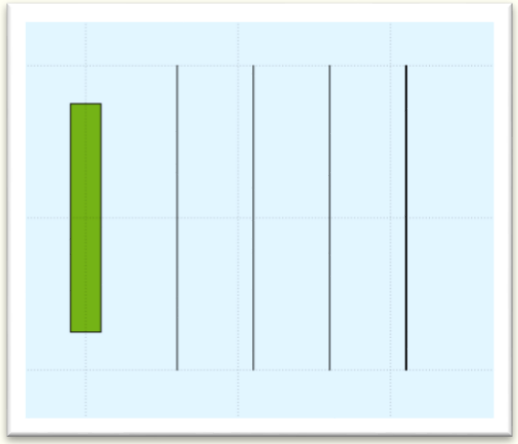
- PLUME geometry:
  - 2 sensitive **50μm Si layers**
  - 2mm SiC foam to separate Si layers
  - Other electronics layers
- Still an approximated layout: should be **staggered MIMOSA28** strips *(see Eleuterio's talk)*

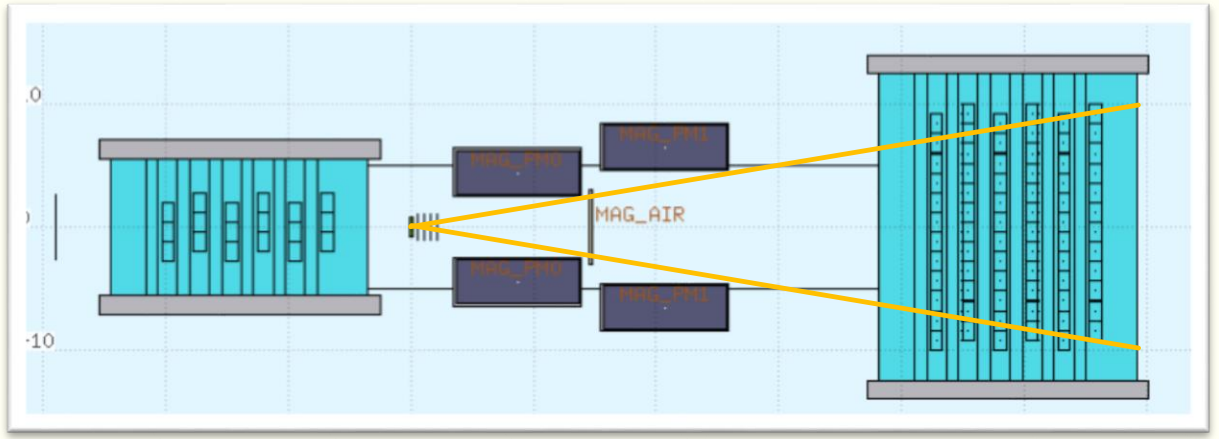
V11.2 Geometry



This is our new geometry proposal

V11.2 – Vertex & Drift Chamber





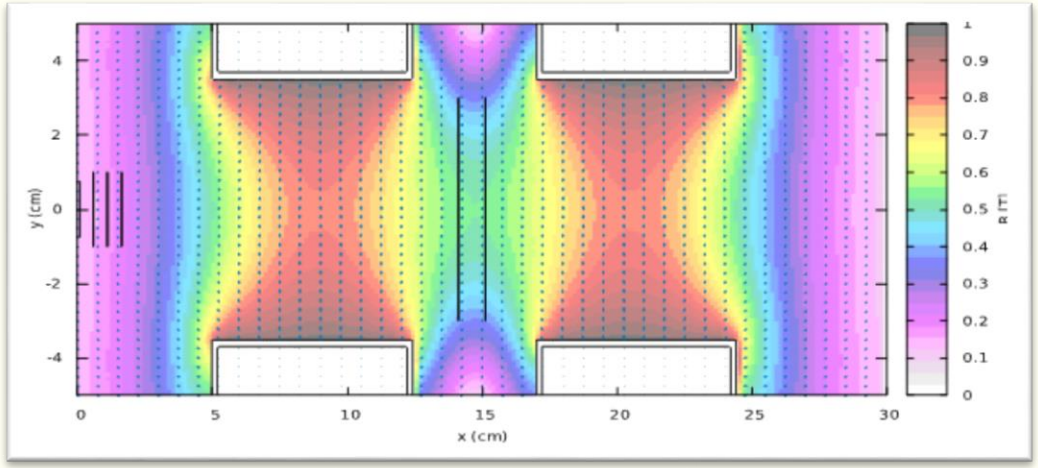
Vertex

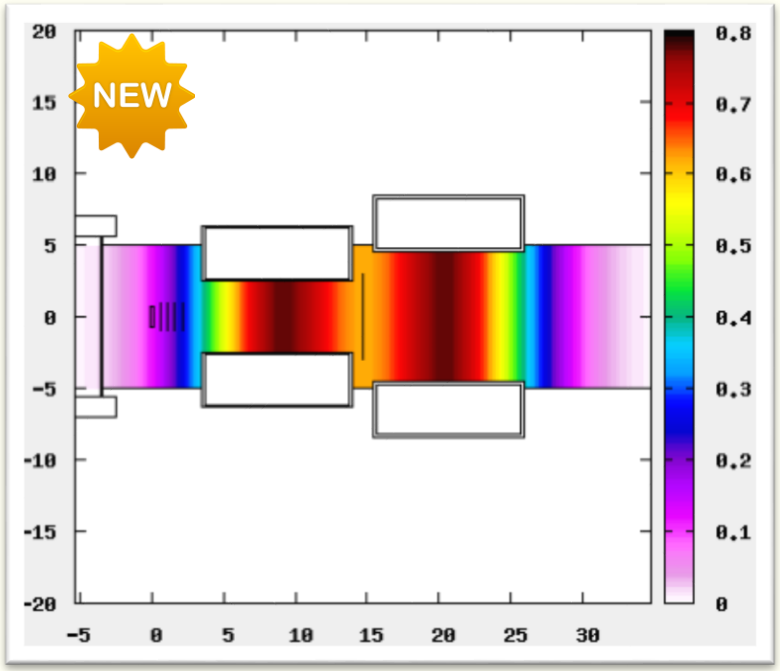
- Maintained thickness
- Maintained distances from target and between layers
- Added **one more layer** to improve tracking resolution

Drift Chamber

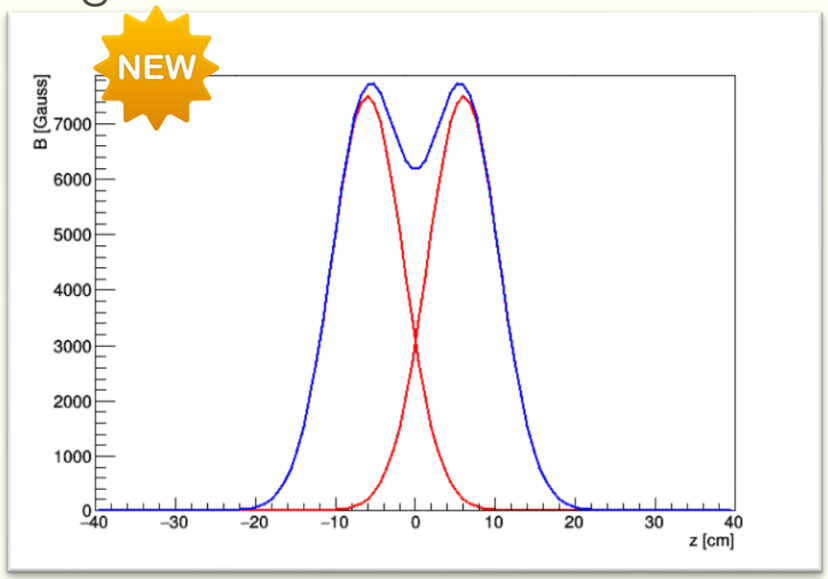
- Maintained cell dimensions
- Chamber **shifted** downstream the target to avoid fringe field and increase track separation. A 5cm only sift value has been chosen to limit the consequent increasing of electronic channels
- **Increased cell number per layer** from 8 (V.10.2) to 12 to cover a 10° solid angle from target

New Magnetic Field Map





The magnetic field map provided last year by Prof. Sanelli was calculated for the old magnets configuration (7cm long). After testing a constant magnetic field, we decided to adopt, just for evaluation purposes, an **approximated map** with $B_x=B_z=0$ and B_y **z-dependent**, calculated as a **sum of two Gaussian** magnetic field intensities that qualitatively fit the new configuration.

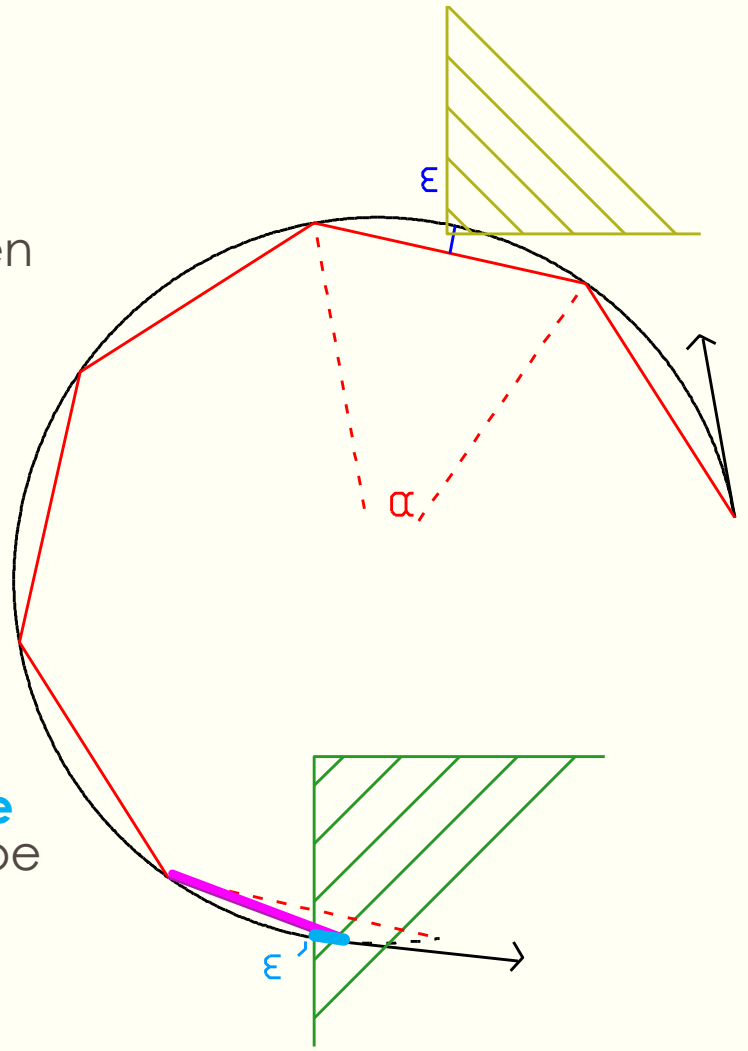


FLUKA tracking in magnetic field

When tracking particles in magnetic field, FLUKA, like many other MC codes, makes use of a different tracking algorithm since, the analytic solution for the crossing of a helix with a generic surface could be rather time consuming.

Magnetic field tracking is performed by **iterations** until a given accuracy when crossing a boundary is achieved.

- The true step (**black line**) is approximated by linear sub-steps.
  Sub-step length and boundary crossing iteration are governed by the required tracking precision.
- The **red line** is the path actually followed,
- The **magenta segment** is the last substep, shortened because of a boundary crossing
- The end point is ALWAYS on the true path, generally NOT exactly on the boundary, but at a distance < ε' (**light blue arc**) from the true boundary crossing. The ε' value has to be specified by the user

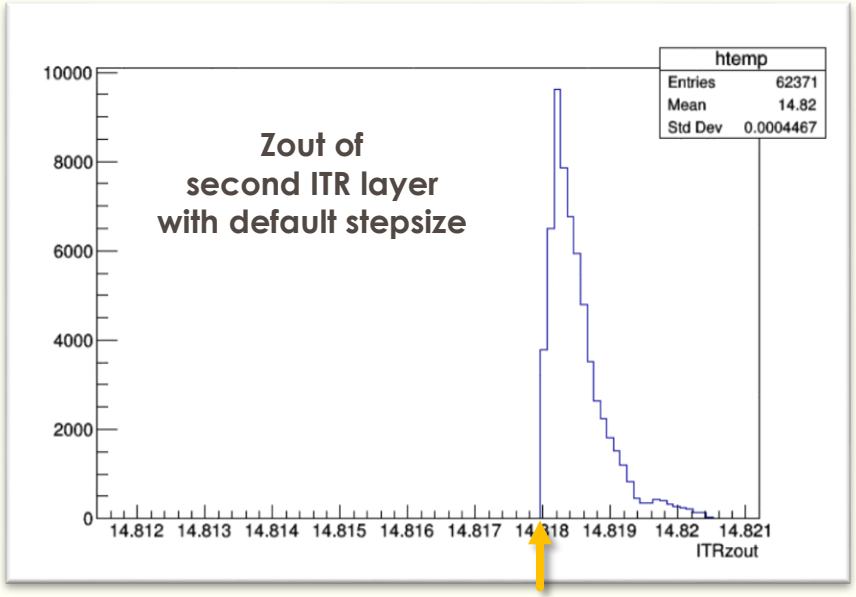


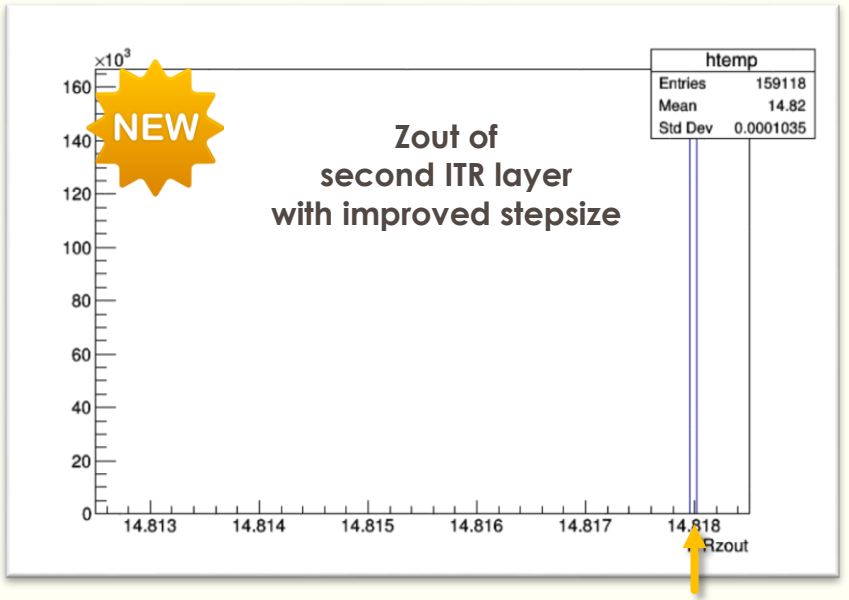
Meaningful user input is required when setting up the parameters defining the tracking accuracy.

Improved tracking in magnetic field

A default accuracy of 0.05cm is adopted in generic inputs for particles stepsize (distance between two subsequent interaction), which was used in the first FOOT simulation released versions. However, when dealing with very thin detectors (~tens or hundreds of μm) an increased accuracy must be specified.

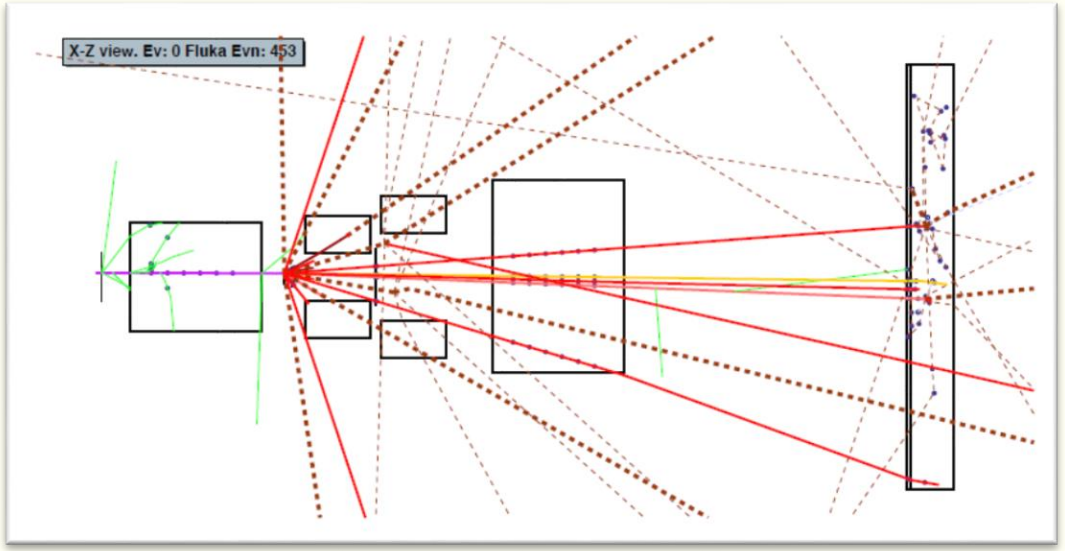
We tested different solution and we found that an accuracy of **0.1μm for stepsize** and **0.1° for the maximum angle** would provide a more suitable tracking accuracy and a sustainable CPU time consuming.

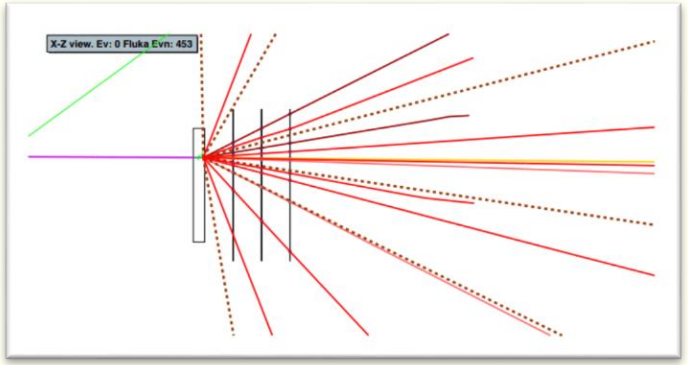




The PoorManDisplay tool

- This tool can be used to visualize particle tracks in each event
  → **event display**
- Different **color code** to distinguish particles (electrons, neutrons, ions of different charges). Solid lines for charged particles, dashed lines for neutral particles
- The event display can be restricted to each subdetector
- At present only available for V10.2 geometry





Some simulation numbers: CPU time and output dimensions

- Simulations have been performed on an Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz
- Different energies ^{16}O beams on C2H4 target have been simulated, with high (1GeV) electromagnetic thresholds
- Each simulation includes 1.E6 events

Beam energy (AMeV)	CPU mean time per event (s)	TXT output file dimension (Gb)	ROOT output file dimension (Gb)
200	0.01972 s	1,15	0,3
400	0.03259 s	1,85	0,45

Some simulation numbers: Fragments production

- ^{16}O beams @200AMeV on C2H4 target (5.E6 events)
- Particles crossing ALL the detectors are:

A/Z	1	2	3	4	5	6	7	8
1	31992							
2	14659							
3	5941	8754						
4		45856						
5		303						
6		5	3607					
7			1665	1675				
8			318	579	278			
9			52	358	1491	54		
10				4	1802	297		
11				1	119	2292		
12					19	3362	56	
13						1916	552	8
14						575	3441	201
15							4903	6014
16								668
17								
	52592	54918	5642	2617	3709	8496	8952	6891

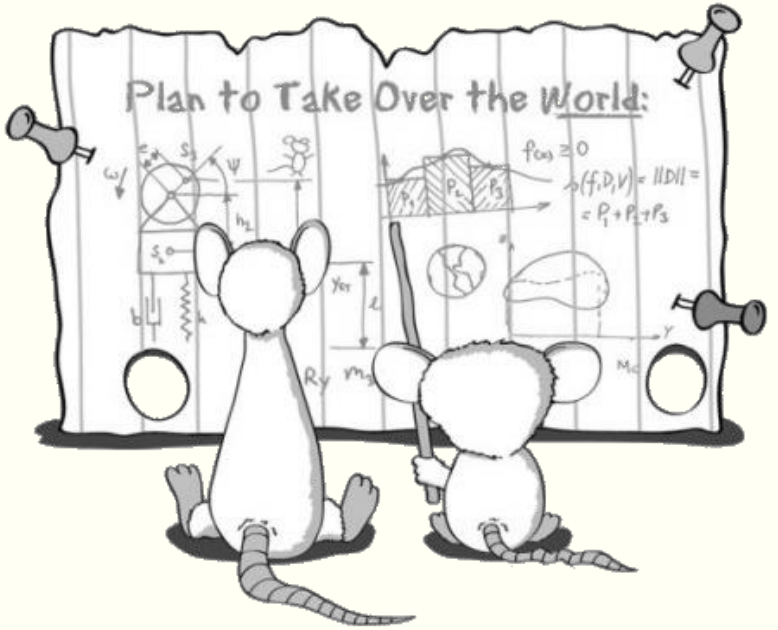
Thanks to Roberto

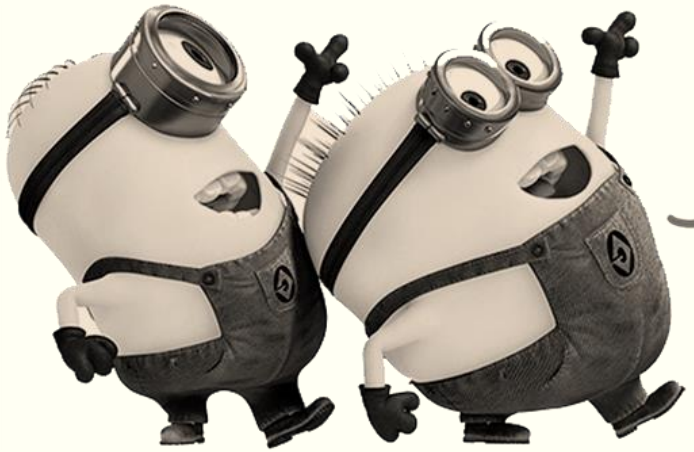
Web tools

- A complete list of the available **ROOT files on Tier3** for different geometry configurations, beams and target can be found on the **TWiki page**: http://arpg-serv.ing2.uniroma1.it/twiki/bin/view/Main/FOOTAvailableSimulation
- On Tier3 also the **configuration files** used to produce the ROOT files and inherited by the reconstruction software are stored
- All the files needed to run the simulation (inputs, FLUKA user routines and configuration files) still have to be updated in the GIT repository

To do list

- **Update the GIT repository** with the new simulation files
- Further **studies on Drift Chamber** performances and design to converge on a final configuration to be implemented in the simulation
- **Update the PoorManDisplay** software for the new geometries
- **Enlarge the team** of people capable of managing the FLUKA simulation for their own purposes





thank you!