

FINAL REPORT

3D Visualization of Muon g-2 data and simulation

Summer student:

Marco Di Bella

Supervisor:

Adam Lyon

Contents

1	Introduction	1
2	Muon g-2 experiment	2
2.1	The importance of g	2
2.2	Why muons?	3
2.3	How Muon g-2 works	4
3	The <i>Art</i> framework	6
3.1	What is a framework	6
4	Visualization	8
4.1	Paraview	8
4.2	Event display	9
4.2.1	SimSteps2VTK	9
4.2.2	SimXtalHits2VTK	11
4.2.3	SimStrawHit2VTK	11
4.3	Interesting event	13
4.3.1	Event 1 (Fig. 11)	13
4.3.2	Event 2(Fig. 13)	14
4.3.3	Event 3(Fig. 14)	15
5	Conclusion	16

1 Introduction

The experiment Muon g-2 is a precision measurement of the anomalous magnetic moment of the muon. This experiment is an upgrade of the previous BNL experiment at Brookhaven National Laboratory and his goal is to obtain a precision of 140 ppb.

The Muon g-2 is important because of the discrepancy between Standard Model predictions and Brookhaven muon experiment values that is more than three sigma. What this experiment will do is to refuse the claim or confirm it with a confidence level exceeding a discovery threshold.

It is going to take the first data in the start of the 2017.

A brief description of experimental apparatus and main detectors will be reported.

The main task of my job here in Fermilab was to learn how the framework used by the experiment works and to successfully connect this framework with a 3D visualization program called Paraview in order to obtain real-time simulation event visualization.

The issue about this connection was that this two elements "talks" different languages.

My job was to write programs in C++ to translate the language in something usable by Paraview. In this way I realized to obtain event display for the *Geant Steps*, for the *Calorimeter Hits* and lastly the *Straw Tracker Hits*.

Pictures of simulation events visualization will be shown.

2 Muon g-2 experiment

2.1 The importance of g

Any charged particle in a magnetic field, whose spin is equal to 1/2, has a precession frequency equal to:

$$\omega_s = g \frac{eB}{2mc} \quad (1)$$

where g is the gyromagnetic factor.

This factor was studied since it was predicted by Dirac's theory to be equal to 2 for an elementary fermion. The really first experiment values agrees with the theory but when in the 1948 Kush and Foley did a precise measurement they found that the experimental value differs fractionally from 2 by a 0.1%. This difference is the anomalous magnetic moment, called a and defined as:

$$a = \frac{g-2}{2} \quad (2)$$

Kush and Foley's experiment gives a gyromagnetic factor for the electron equal to:

$$g_e = 2.00238(6)$$

This difference was explained supposing that virtual particles continually fluctuate in and out the vacuum. The physicist Schwinger took one look at the problem and solved it adding a term of 1st order electron self-interaction. In this way the theoretical values become:

$$g_e^{theo} = 2 \left(1 + \frac{\alpha}{2\pi} \right) \simeq 2.00232.$$

The contribution added by Schwinger is shown in Figure (1b).

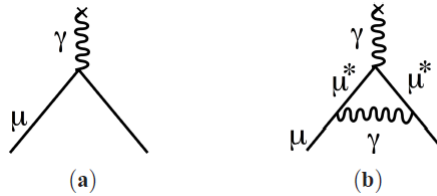


Figure 1: Feynman's diagrams a) for a theory with $g=2$ b) with the 1st order electron self-interaction

Nowadays the experimental value has much more significant figures than before and every one of these came from different contributions. From the QED contribution added by Schwinger to the QCD, Weak and something else that we are still searching for. The Muon g-2 experiment is working to find some discrepancy between the Standard Model contributions and what fluctuates in and out of the vacuum. In Figure (2) it is possible to see the precision of the past measurement and the goal for Fermilab's experiment.

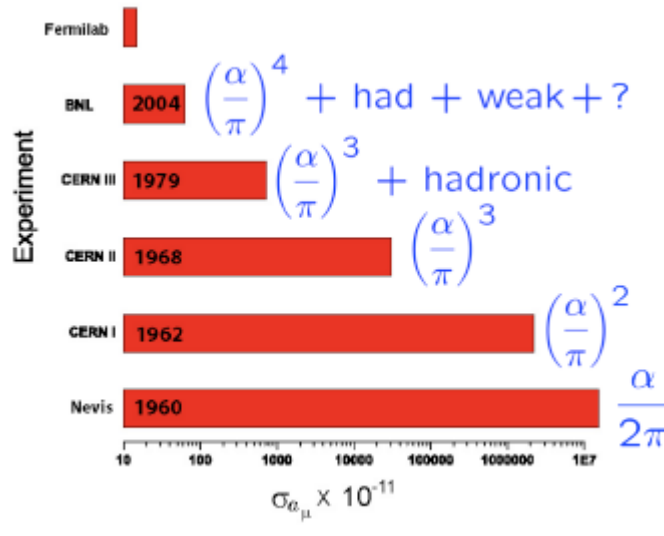


Figure 2: Precision for the previous experiment compared with Fermilab's one and contribution from the vacuum

2.2 Why muons?

Muons are an elementary particles with similar interaction to electrons but 200 times more massive. They are heavy enough to use them experimentally and also they are light enough to be produced copiously.

An other important aspect of these particles is that a beam of muons can be produced polarized. We can collide a proton beam with a light-element target and then the pions produced in this way will decay in a muon and a neutrino. Because of the fixed helicity of neutrinos also the muons have to be polarized. This means that when the muon will decay into the sample the positron obtained will be emitted in the direction of the muon's spin. In this way we can control the behaviour of the muon' spin in the sample.

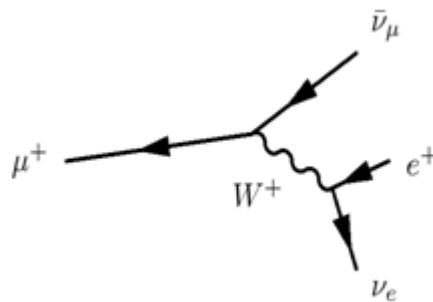
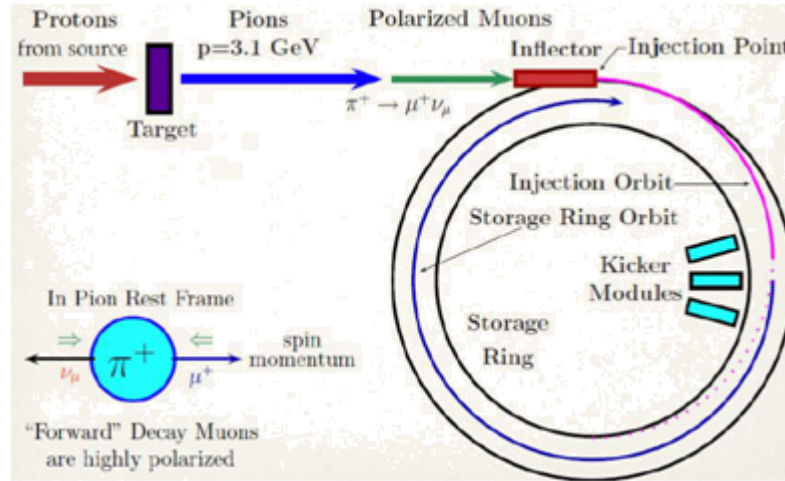


Figure 3: The anti-muon three body decay

2.3 How Muon g-2 works

The Muon g-2 is the "son" of the previous experiment at Brookhaven. The ring is the same and it was moved from there to Fermilab in a very interesting trip. It took 35 days and 2 special vehicles to bring it to Chicago. Except from the ring the other detectors are new and much more precise than the previous ones. They will take more than 20 times the statistics acquired in three runs at Brookhaven and the precision will increase of a factor 4 respect to BNL one.

The path of the particles is the following: custom short "bunches" of protons will be formed in the Recycler and delivered one at time to the existing antiproton target. A 3.1 GeV positive pion beam will be directed out of the target and along a 290 m beam line. Most of the pions will decay along the way and forward-going, highly polarised, muons will be captured in the line. The muons are directed around the antiproton accumulator complex and then back toward the target region along a parallel transfer line. The resulting pure muon beam is injected in the storage ring.



Muons are kept into the ring with a magnetic field \vec{B} which will effect the particles in two different ways. The first effect is to permit the spiraling of particles inside the ring with a so called cyclotron frequency, ω_c defined in this way:

$$\omega_c = \frac{eB}{\gamma mc} \quad (3)$$

The second effect is the Larmor precession of its spin vector due to torque provided by the magnetic field, with a spin frequency given by:

$$\omega_s = \frac{g_\mu eB}{2mc} + (1 - \gamma) \frac{eB}{\gamma mc} \quad (4)$$

The spin frequency relative to the Cyclotron frequency is the *anomalous precession frequency*, ω_a :

$$\omega_a = \omega_s - \omega_c = \left(\frac{g-2}{2} \right) \frac{eB}{mc} \quad (5)$$

In the experiment is needed also to vertical focus the beam and for this reason we need an electric field \vec{E} . In this way the anomalous precession frequency equation becomes more difficult unless the muon momentum is chosen to be equal to $p = 3.14 \text{ GeV}/c$. With such a neglect the equation is:

$$\vec{\omega}_a = -a_\mu \frac{e\vec{B}}{mc} \quad (6)$$

where a_μ is the *anomalous magnetic moment*.

The final key element of the Muon g-2 experiment is about the parity violation of the muon three body decay and this means that we can track the muon spin behaviour with the correlation between this and the directions in the muon rest frame of the decay electron/positron.

3 The *Art* framework

3.1 What is a framework

A framework is a universal, reusable software environment that provides informatics infrastructures built for the user to easily use it. It has to be user friendly because everyone has to learn it before it can be used. There are pros and cons about the use of a framework.

Cons:

- Infrastructure is hidden behind the scenes from you
- Your ideas may not be included
- You have to trust a system you didn't write
- You miss out on the fun of writing super-cool complicated C++ code

Pros:

- You can concentrate on physics code
- Your C++ is pretty easy (you are using a complicated system, not writing it)
- You get to miss out having to maintain the complicated C++ code (yay!)
- You can use code from others and share yours with others
- You can get services for free (e.g. data handling)

Taking in account this pros and cons the Muon g-2 experiment decided to use a framework, and to be precise the *Art* framework for different reason:

1. A "lite" forked version of the CMS framework
2. Supplies all expected framework services as well as links between data objects (Ptr's and Assn's)
3. Used by many Fermilab Intensity Frontier Experiments (NOvA, Muon g-2, Mu2e, MicroBoone, LBNE) and some others (e.g. DS50)
4. Written by SCD/CET department
5. Currently being adapted for multi-processing and DAQ

To make this more clear it is shown in the Fig. (4) an example of infrastructure.

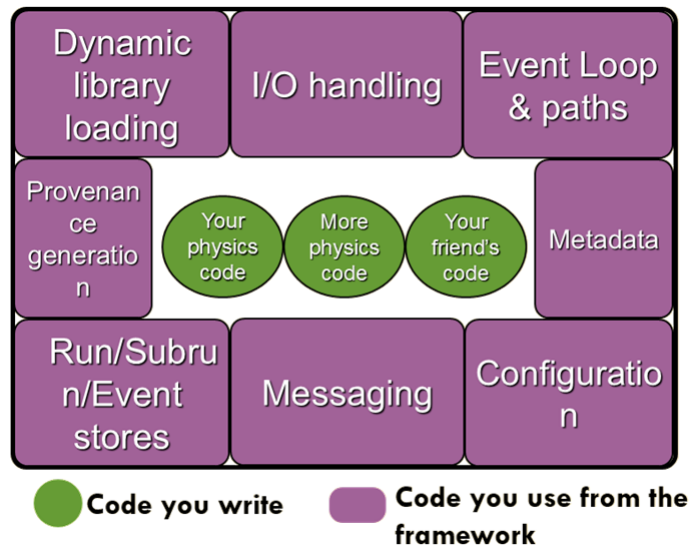


Figure 4: The infrastructure provided by the framework

As it is possible to see the most of the code is already done and in this way the physicist can think just about the "physics code" and nothing more.

The *Art* framework is modular, this means that you write modules that piece together. There are different types of modules such as:

- **Input source:** this module is a source for data. E.g. a ROOT file or an Empty one for the start of simulated data
- **Producer:** this module create new event data from scratch or by running algorithms on existing data
- **Filters:** like producers, but can stop running of downstream modules
- **Analyser:** which cannot save to event. For, e.g. diagnostics plots
- **Output:** Writes data to output file (ROOT). Can specify conditions and have many files

All these modules can just read data from the event, so they should not communicate with each other, except through the event. A picture of how it works is attached in Fig. (5)

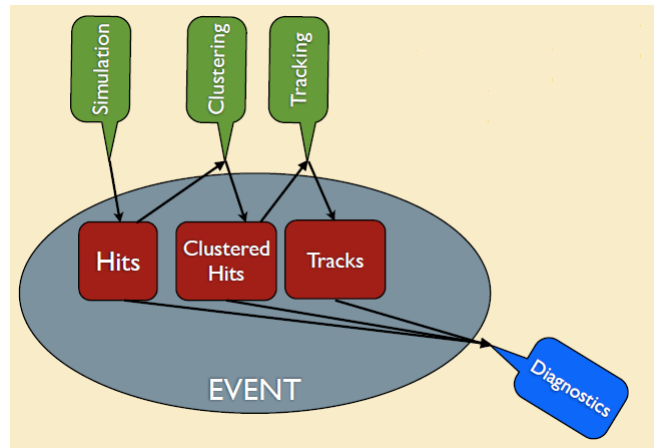


Figure 5: Every module can only talk to the Event and not to each others

4 Visualization

My project was about visualization for simulation events, I used a program called Paraview to 3D visualize event generated from Muon g-2 simulation. The main part of my work was about writing C++ codes in order to connect the *Art* framework with Paraview. Paraview is a program used by other important scientific institution such as the Los Alamos National Laboratory and is really easy to work with. The only problem is that Paraview talks a different language from the one used by *Art* and because of this we need specific modules for translating the Event language into a VTK Multi-BlockDataSet, the kind of object Paraview understands. I have worked with different modules, starting from the steps muon does into the ring, implementing the calorimeter hits and at the end visualizing crystal hits. Here I will show some pictures of the program itself and some screen-shoot of the event display.

4.1 Paraview

The visualization program used in my work is an open-source, multi-platform data analysis and visualization application called Paraview. ParaView users can quickly build visualizations to analyze their data using qualitative and quantitative techniques. The data exploration can be done interactively in 3D or programmatically using Paraview's batch processing capabilities. ParaView was developed to analyze extremely large datasets using distributed memory computing resources. It can be run on supercomputers to analyze datasets of petascale size as well as on laptops for smaller data, has become an integral tool in many national laboratories, universities and industry, and has won several awards related to high performance computation. The main page of the program is here reported in Fig. (6).

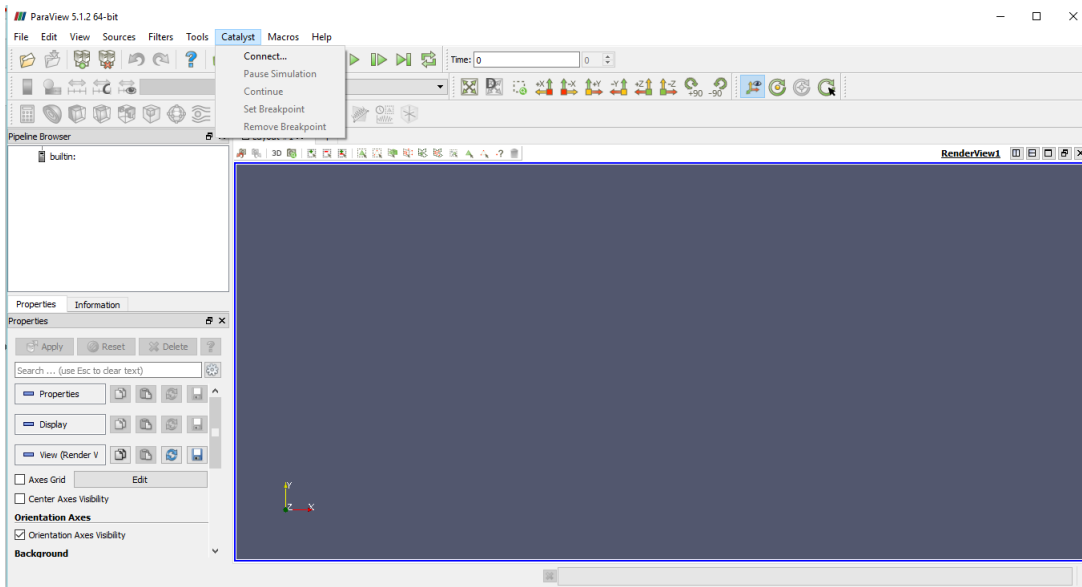


Figure 6: The main page of Paraview

It is important to highlight the button "Connect..." under "Catalyst" which allows you to connect the program with a server and visualize in real time the simulation on the grey screen you can see in the picture.

4.2 Event display

In this section I will report in chronological order the modules I worked on and for everyone there will be pictures of the final visualization. Every module ended with -2VTK because like I said before the physical modules were already on the framework and the ones I wrote have the task to translate the data from *Art* language into a VTK object that can be read by Paraview. A flow chart with an example of the simulation stream is in Fig. (7).

4.2.1 SimSteps2VTK

The first module I worked on is the one that use Geant4 to describe the particles' path into the detector and to represent all the physics interaction with the \vec{B} field, the decay and also with the ring itself. The peculiarity of this modules is that it uses both points and cells for entirely describing a particle's path. The points has information about the global time, the momentum and the spin of the particle, instead the cells contains information such as the energy lost in the step between two subsequent points, the time between them and the pdg number of the particle. So when we do the visualization for this module what appears on the screen is the muon running into the ring until it decays into a positron and a neutrino (not visible). The positron then can interact with the ring or can radiate a photon for bremsstrahlung. In Fig. (8) the SimSteps2VTK

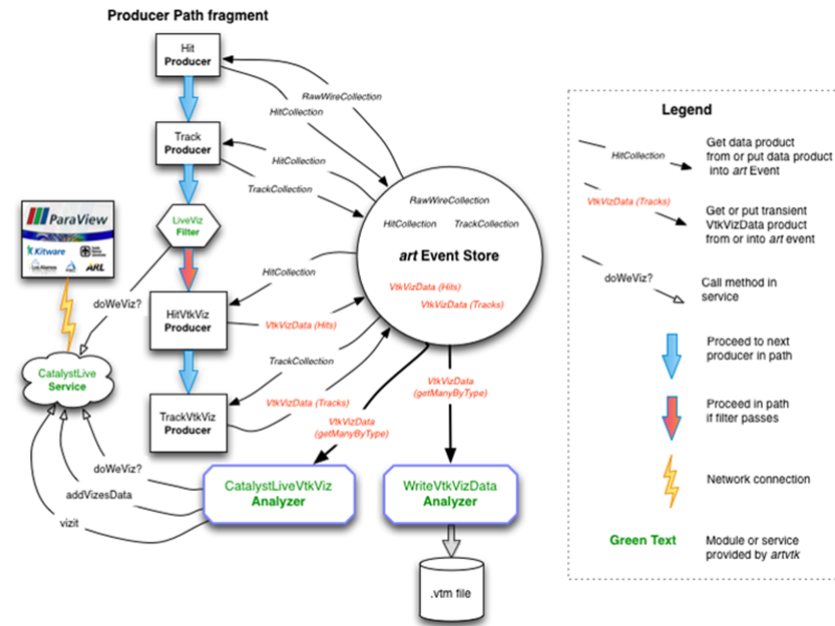


Figure 7: An usual flow chart for visualization

module is connected with Paraview.

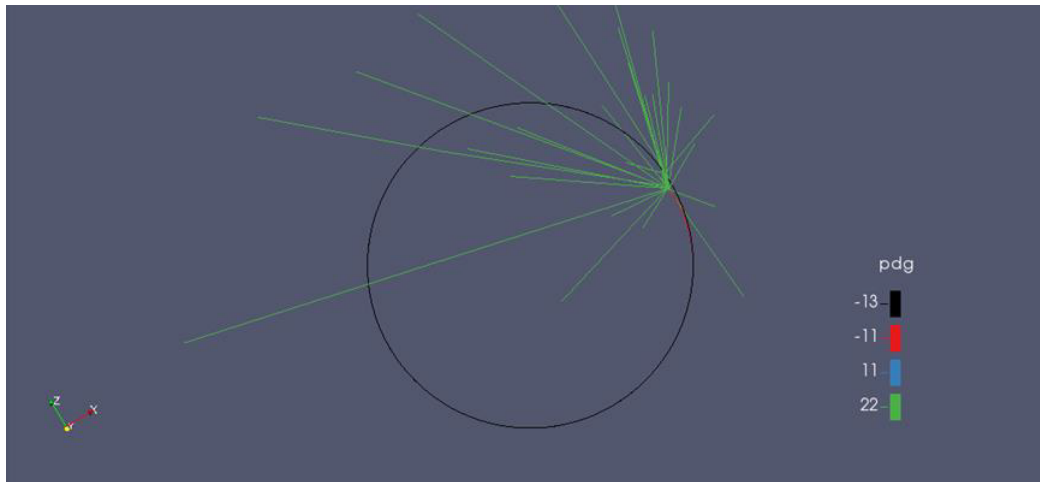


Figure 8: Visualization for SimSteps2VTK

Particles are identified with the Monte Carlo number with the rules that negative numbers correspond to anti-particles. So:

- 11= electrons

- 22= photons
- -11= positrons
- -13= positive muons

It is possible to see the muon (in black) running around the ring and then decaying in a positron (red) which will start a shower of photons (green) and some electrons (blue).

4.2.2 SimXtalHits2VTK

The work on this module was in two parts, at the beginning I worked on a module called CaloHits then on his final version called SimXtalHits. The difference between the two modules is that the first one just registers the energy deposited into the calorimeter in a point placed at the centre of the calorimeter. The final version instead has points at the centre of every crystal hit by the particle. For every point we have information like the particle name, the energy, the hit time and of course information about the position of the crystal hit. In Fig. (9) is shown the event with the SimXtalHits2VTK module (white dots) and the SimSteps2VTK module active.

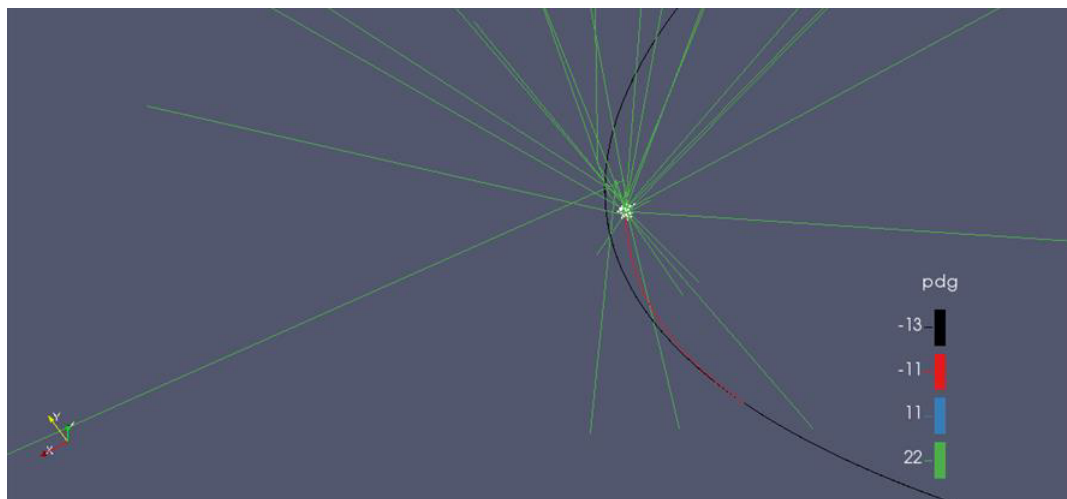


Figure 9: Visualization for SimXtalHits2VTK

The Monte Carlo number are the same explained before. With this zoom in it is easier to see the positron and then the crystal hit.

4.2.3 SimStrawHit2VTK

The last part of my work was to implement the hits on the tracker's straws. This module contains particle, time and straw information in dots like SimSteps2VTK. So the visualization of this module will be really similar to the previous one but it is possible to notice that this dots will be divided by precise gap, that is the gap between different

straw planes and trackers. Image in Fig. (10) is what looks like the final visualization event project with all the module active.

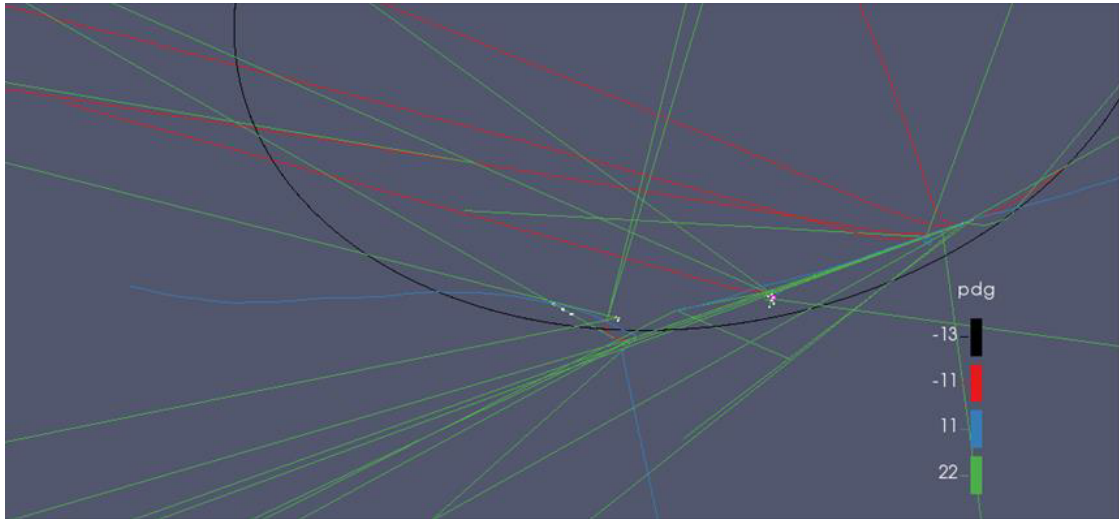


Figure 10: Visualization for SimStrawHit2VTK

In this picture is possible to see the XtalHits in the two groups of points on the right and the StrawHit on the left with precise gaps between them. The particles are the usual: muon running around the ring then decaying in a proton and then from this comes photons and electron. On closer inspection it is possible to see that the straws were activated by a photon (green). In the experiment this will not happen because it is needed a charged particle to activate a gas base detector but right now the simulation is not extremely precise and then the tracker is activated by every kind of particle passing throughout its physical position.

4.3 Interesting event

In this section of the paper my task is to highlight the importance of visualization for a fast comprehension of the simulation events. All the pictures has the g-2 ring for a much more easier comprehension.

4.3.1 Event 1 (Fig. 11)

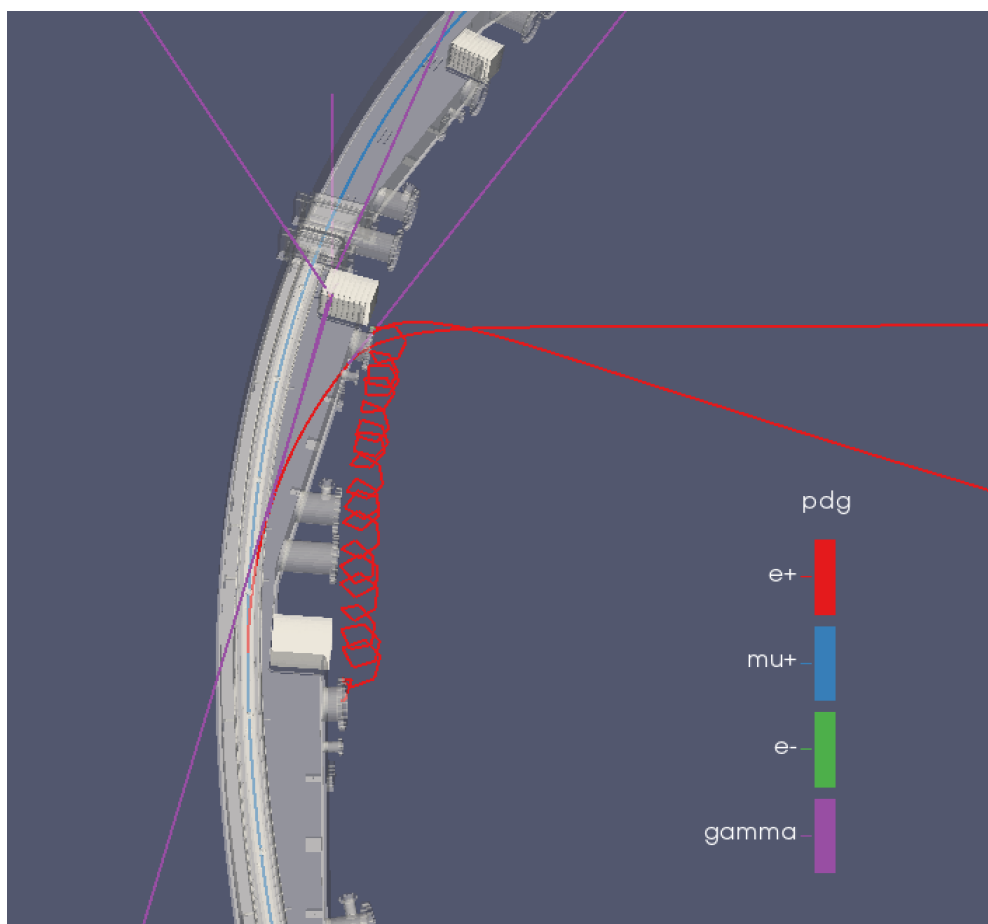


Figure 11: Overview of the first event

What happened in this simulated event is that a muon decayed near a calorimeter station but the positron created is few energetic to reach the calorimeter and because of the magnetic field it exit from the ring before reaching the detector. When the positron hits wall it produces photon which converts to an electron and a spiraling positron. All these interactions are more visible in Fig. (12).

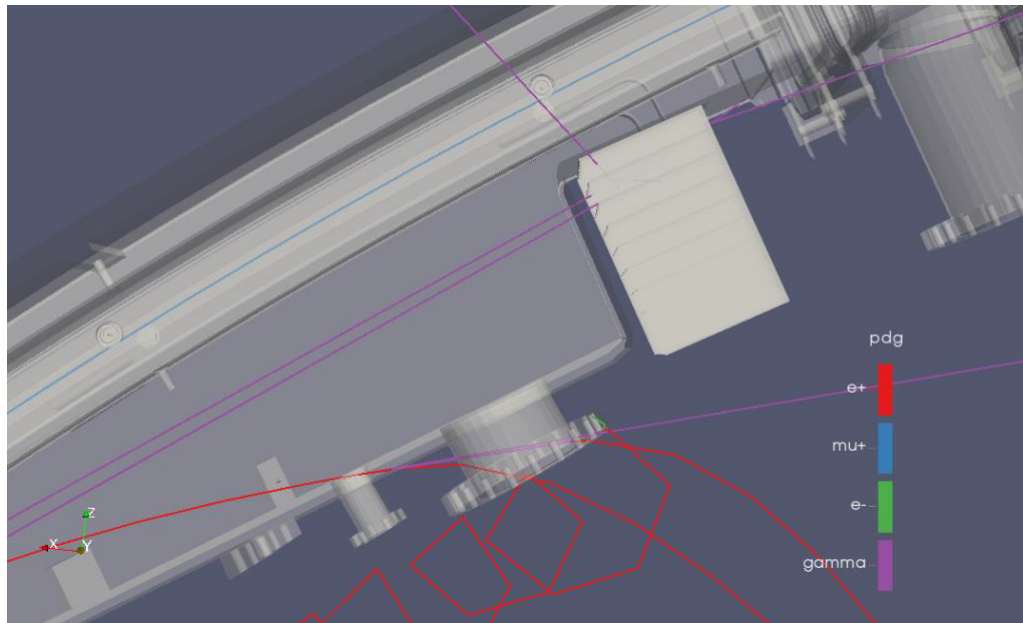


Figure 12: Zoom in of the first event

4.3.2 Event 2 (Fig. 13)

In the second event reported we focus on the crystal hits that are represented by the spheres and there is a scale of colour concerning the energy deposited in the hits. The first hit is where the photon loses more energy and then it creates a small shower of light photons.

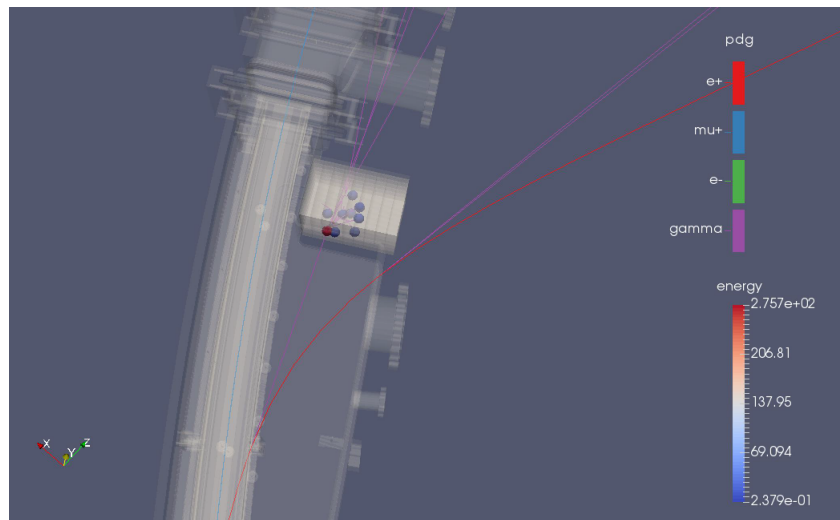


Figure 13: Overview of the second event

4.3.3 Event 3 (Fig. 14)

The third and last event is similar to the previous one but in this case is the positron to interact in the calorimeter and because it is heavier it creates a bigger photon shower

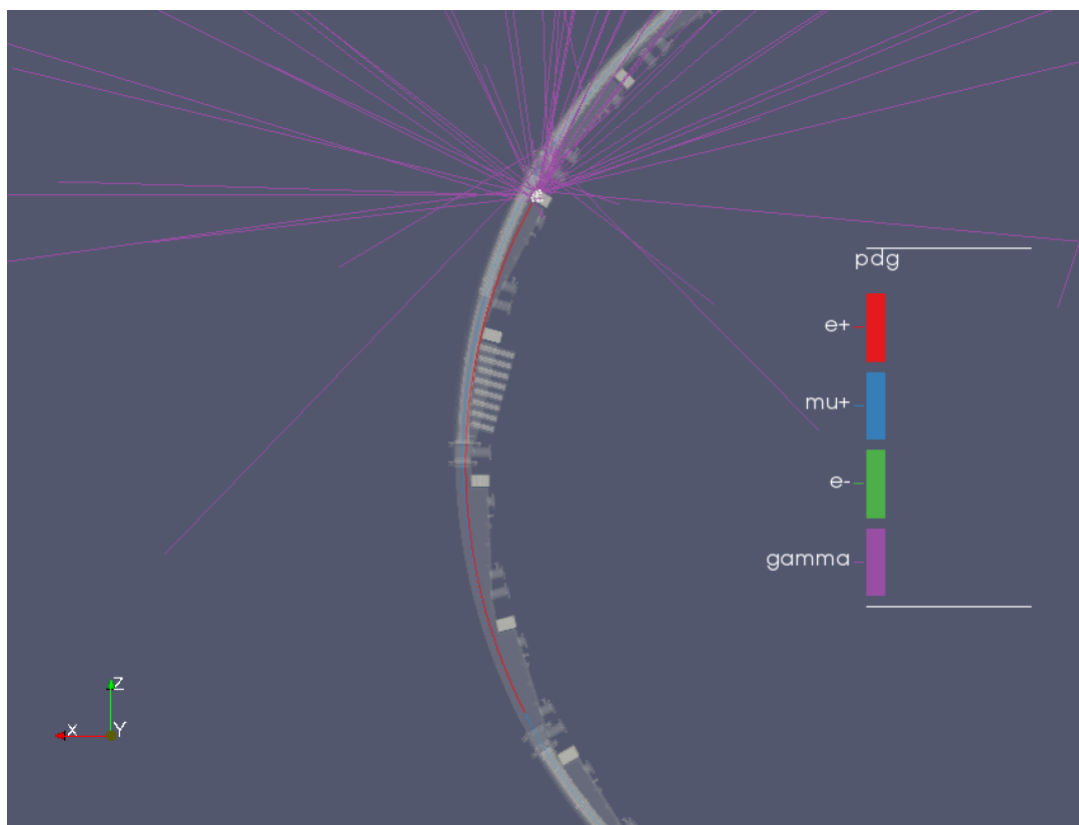


Figure 14: Overview of the third event

In the zoom in Fig. (15) it is possible to see that almost all the calorimeters' crystal are activated (white dots) by the shower.

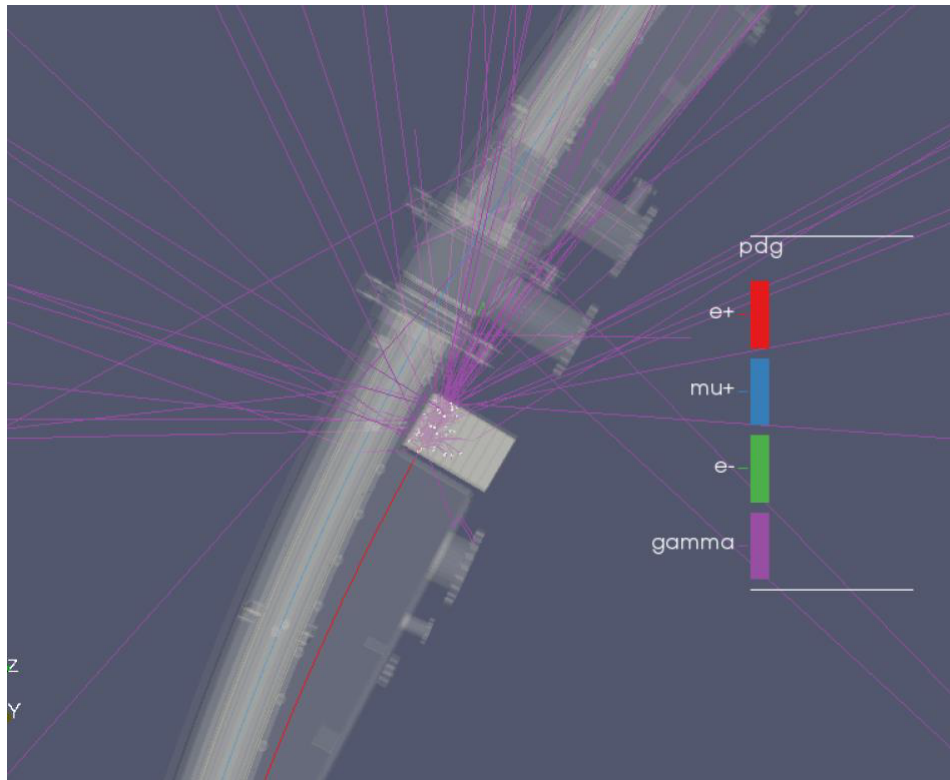


Figure 15: Overview of the third event

5 Conclusion

In order to draw the appropriate conclusion I can say that:

- The first part of the work was concluded excellently learning about the framework in general, their use, pros and cons and then I learned why the experiment Muon g-2 chosen specific the *Art* framework to work with.
- Then I started learning about the actual physical simulation used by Muon g-2, all the work already done and what has to be done after this summer school period.
- I have also learned how Paraview, an open-source visualization program, works and how to connect it with the Catalyst option to an on-line server.
- Writing the codes necessary to translate the actual simulation output in a VTK object I have improved my computational programming skill.
- Combining all this work at the end I achieved to collect some nice pictures and explain some strange stuff happening in the simulation. Visualization tool-kit

was useful to solve a Geant4 problem where an electron remains stuck in the calorimeter.

To conclude all my work right now is on the Muon g-2 server and is usable by everyone in the team. The visualization is as much powerful as was expected. A lot of people in the team already asked for visualization help for their simulations.