

# respuestas

February 2, 2021

## 1 PREGUNTAS

Pregunta: ¿qué es el Hub de Docker? ¿cuál es la diferencia entre una imagen y un contenedor? Un poco de ayuda.

Respuesta: El Hub docker es una colección de imágenes online. Una imagen es el sistema de archivos y configuración de nuestra aplicación que se usa para crear contenedores. Un contenedor es una ejecución de una instancia de una imagen, los contenedores ejecutan las aplicaciones reales e incluyen la aplicación y todas sus dependencias

```
echo ip$(ifconfig eth1 | grep "inet " | awk -F'[: ]+' '{ print $4 }' | sed 's/\./-/g')-$SESSION_ID
```

Pregunta: ¿qué hace el comando anterior? ¿De qué subcomandos se compone y qué papel juegan (ifconfig, grep, awk y sed)?

*Respuesta:* el comando ifconfig devuelve la configuración de la red >eth1 Link encap:Ethernet HWaddr 02:42:AC:12:00:0F inet addr:172.18.0.15 Bcast:0.0.0.0 Mask:255.255.0.0 UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:1353 errors:0 dropped:0 overruns:0 frame:0 TX packets:1261 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:0 RX bytes:165434 (161.5 KiB) TX bytes:364403 (355.8 KiB)

de todo el texto devuelto por el comando el comando grep filtra por las líneas que contienen la palabra “inet” >inet addr:172.18.0.15 Bcast:0.0.0.0 Mask:255.255.0.0

awk se usa para procesar datos basados en texto en este caso se usa para imprimir la cuarta columna del texto filtrado por el grep. >172.18.0.15

sed sustituye los caracteres “.” por “-” >172-18-0-15

por último se añade delante “ip” y detrás el ID de sesión: >ip172-18-0-15-c0akij434gag009jgri0

Preguntas: ¿Qué es un fichero ppk? ¿Qué es una clave RSA público/privada (fichero key)?

Respuesta: Es un fichero que se usa para configurar conexiones seguras sin password. Se genera una clave privada que se copia en el servidor y los clientes se copia la pública que se usa para validar el usuario

Pregunta: ¿cómo se llama a este tipo de haz de partículas?

*Respuesta:*

Pregunta: ¿Cómo es posible que veamos figuras 3D en una página web? ¿Qué estándares, alternativas, consorcios e instituciones están implicados? ¿Qué alternativa estamos usando en este ejercicio

(pista) y qué gran centro de investigación en matemáticas estuvo implicado (pista)?

Respuesta: Es posible porque existen diferentes estándares como WebGPU o WebGL que han creado APIs para trabajar con 3D, En el caso del WebGPU está implicado el W3C GPU for the Web Community Group con ingenieros de Apple, Mozilla, Microsoft, Google y otros. En el caso de WebGL está desarrollado por Khronos WebGL Working Group fundado por las compañías ATI Technologies, Discreet, Evans & Sutherland, Intel Corporation, NVIDIA, Silicon Graphics (SGI), y Sun Microsystems.

En el caso de esta práctica estamos usando X3d que forma parte del Web3D consortium

Pregunta: ¿cuánto tiempo puede estar serpenteando un fotón que nace en el centro del Sol hasta llegar a la retina de cualquiera de tus ojos?

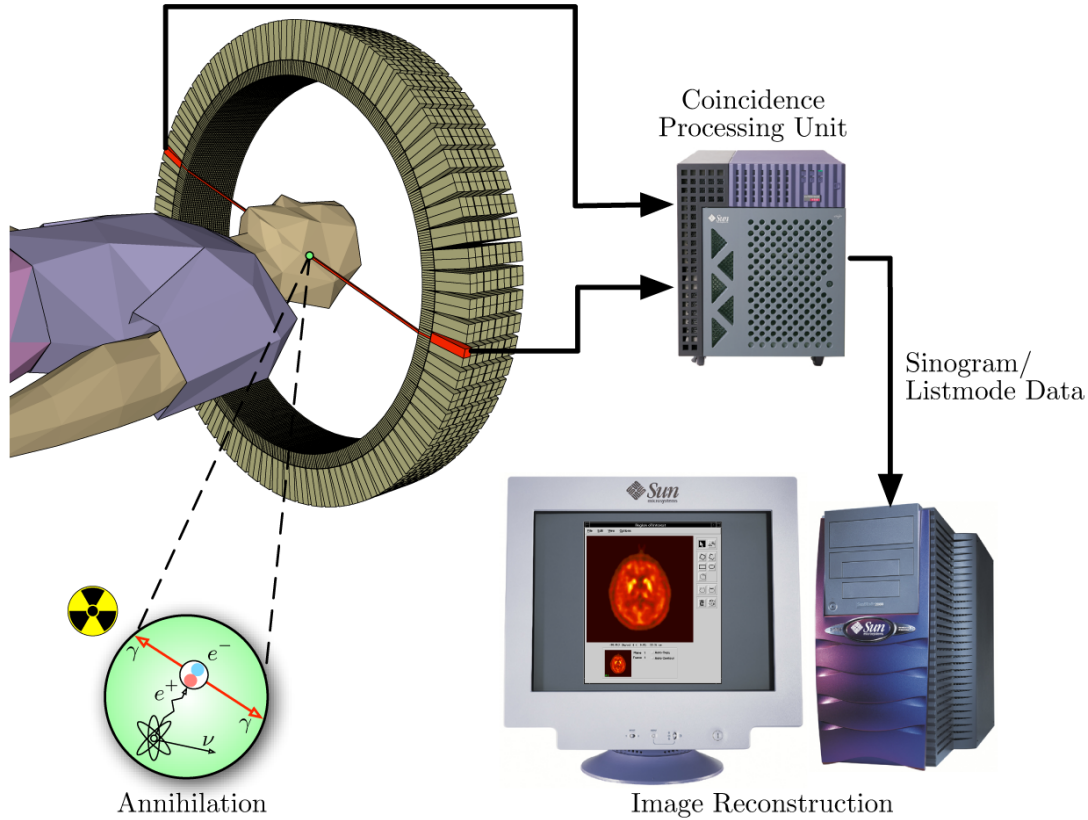
*Respuesta:* Se estima que un fotón tarda entre 10.000 y 170.000 años en llegar desde el centro a la superficie del sol y 8 minutos más desde la superficie del sol hasta nuestro ojo (izquierdo o derecho :wink: )

Pregunta: ¿qué es un positrón? ¿Quién postuló su existencia teórica y cuándo y cómo se descubrió experimentalmente?

Respuesta: es una partícula elemental , antipartícula del electrón. Forma parte de la antimateria y su carga es de  $1e$ . Fue predicha por Paul Dirac en 1928 y descubierta por Carl David Anderson en 1932 al fotografiar las huellas de rayos cósmicos en una cámara de niebla.

Pregunta: ¿qué relación tiene la modalidad radiológica PET con la aniquilación de pares electrón-positrón? ¿Cómo se trasladan positrones hasta el interior de las células cancerígenas?

Respuesta: Cuando los positrones se encuentran con los electrones se emiten en dirección opuesta dos fotones que son capturados y permiten localizar la zona afectada. Como se muestra en la siguiente imagen:



los positrones se administran en forma de un radiofármaco de vida media ultracorta administrado a través de una inyección intravenosa.

## 2 Simulaciones

### 2.1 Instalación

He optado por instalar Docker localmente, he descargado el instalador desde <https://hub.docker.com/editions/community/docker-ce-desktop-windows/>

Una vez instalado desde VS Code creo un proyecto y dentro de la carpeta copio los ficheros `simulation.py` y `wrl2html.py` que descargo de <https://raw.githubusercontent.com/pammacdotnet/FFRepo/master/>

desde el terminal PowerShell de VS Code ejecuto: `>docker pull pammacdotnet/g4lpwd`

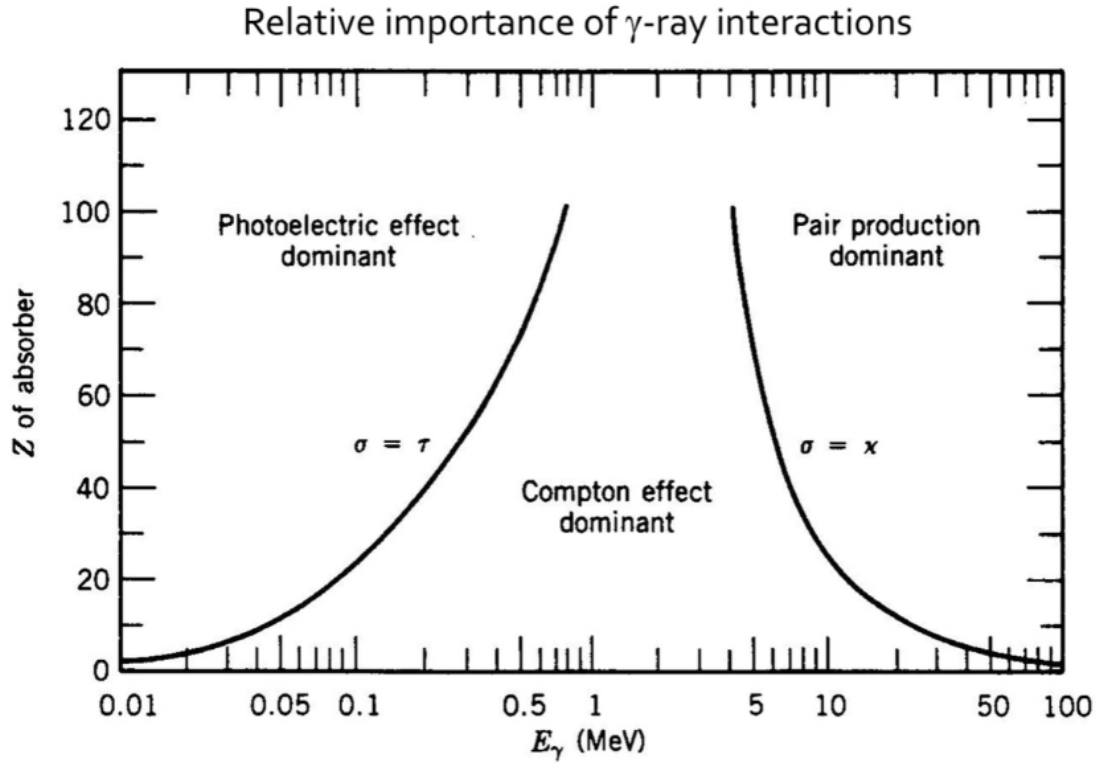
para ejecutar cada simulación ejecuto el comando: `>docker run -v ${PWD}:/root pammacdotnet/g4lpwd`

esto genera un fichero llamado `simulation.wrl` con los resultados de la simulación que se puede transformar en un html ejecutando el comando: `>./wrl2html.py -wrl simulation.wrl -html simulation.html`

Para la visualización de los ficheros wrl he optado por la instalación del programa paraview que he descargado de: <https://www.paraview.org/>

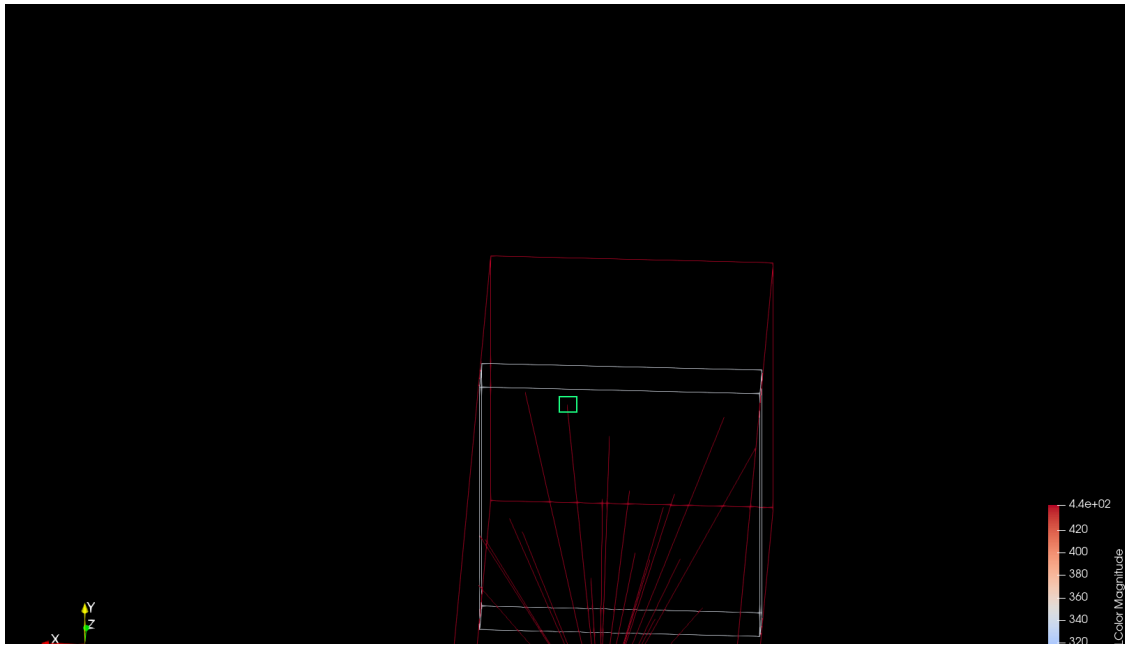
## 2.2 Efecto Fotoeléctrico

Para simular el efecto foto eléctrico usamos una energía de 0.01 MeV ya que en esas intensidades predomina el efecto fotoeléctrico según el grafico:

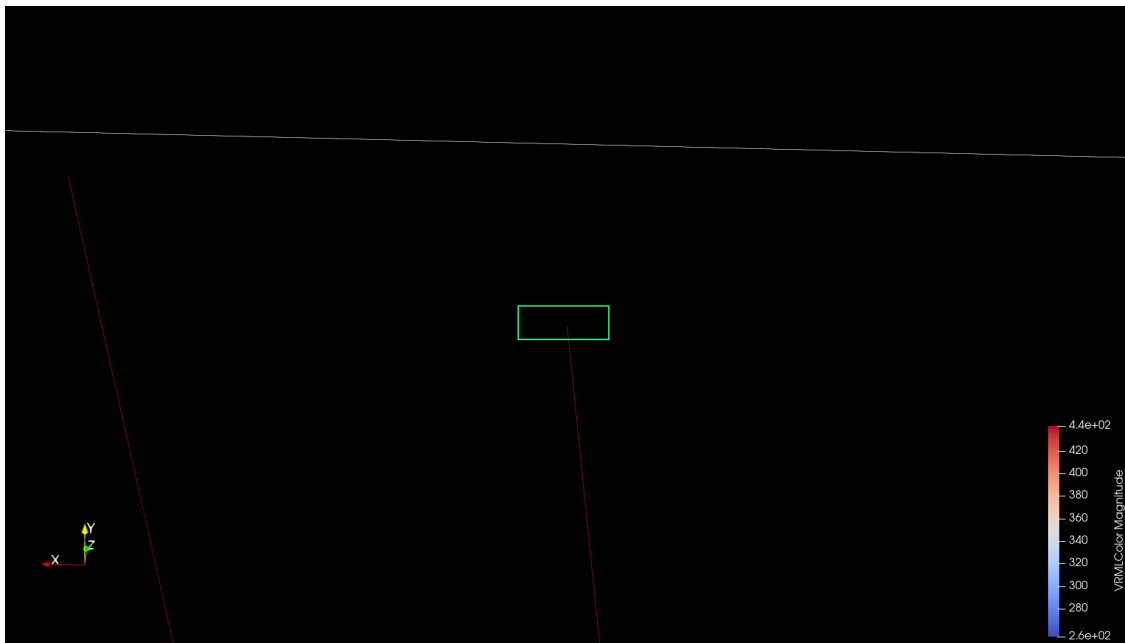


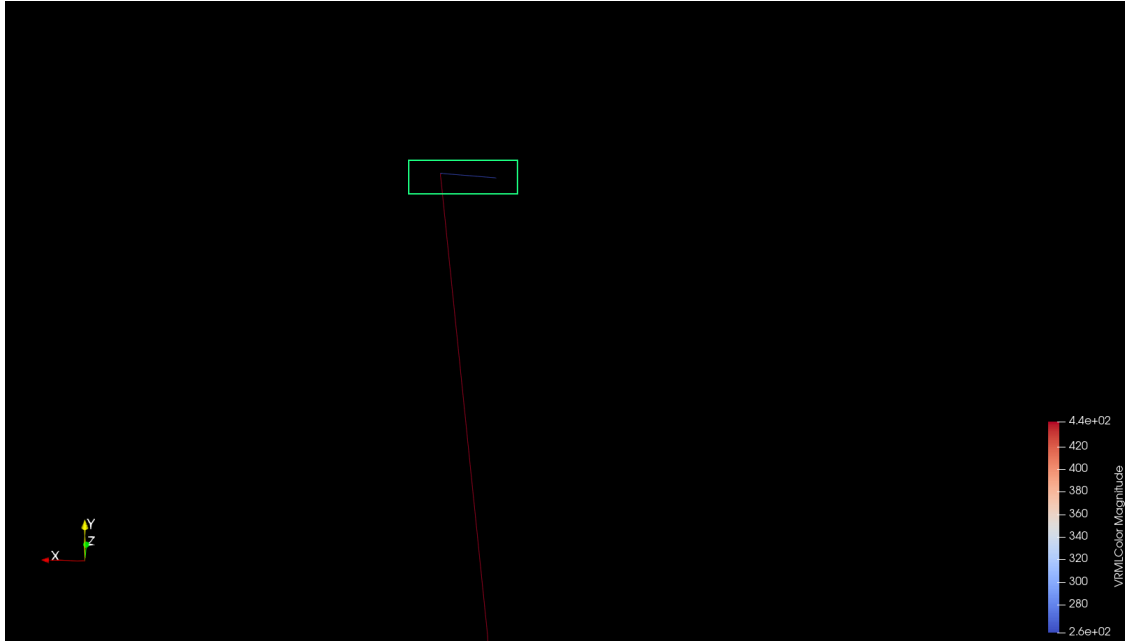
para ellos ejecutaremos la simulación: `>docker run -v ${PWD}:/root pammacdotnet/g4lpwd -energy=0.01 -wrl=./simulacion/fotoelectrico.wrl`

esto crea dentro de la carpeta simulación un fichero fotoelectrico.wrl que abro con el programa paraview



haciendo zoom podemos observar cómo al final se produce el efecto fotoeléctrico



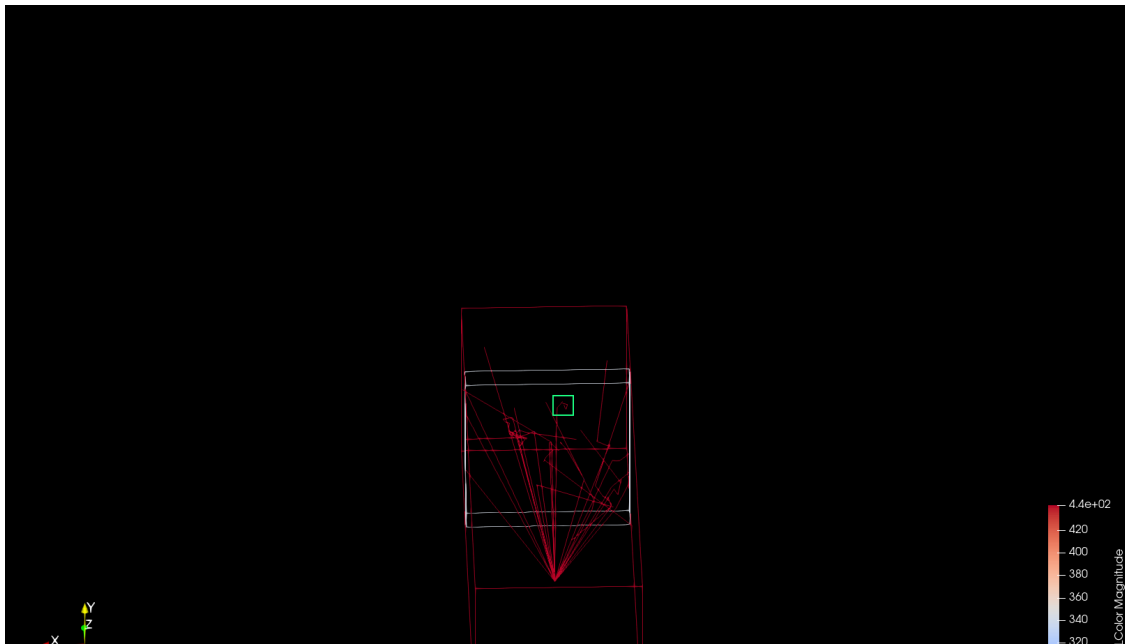


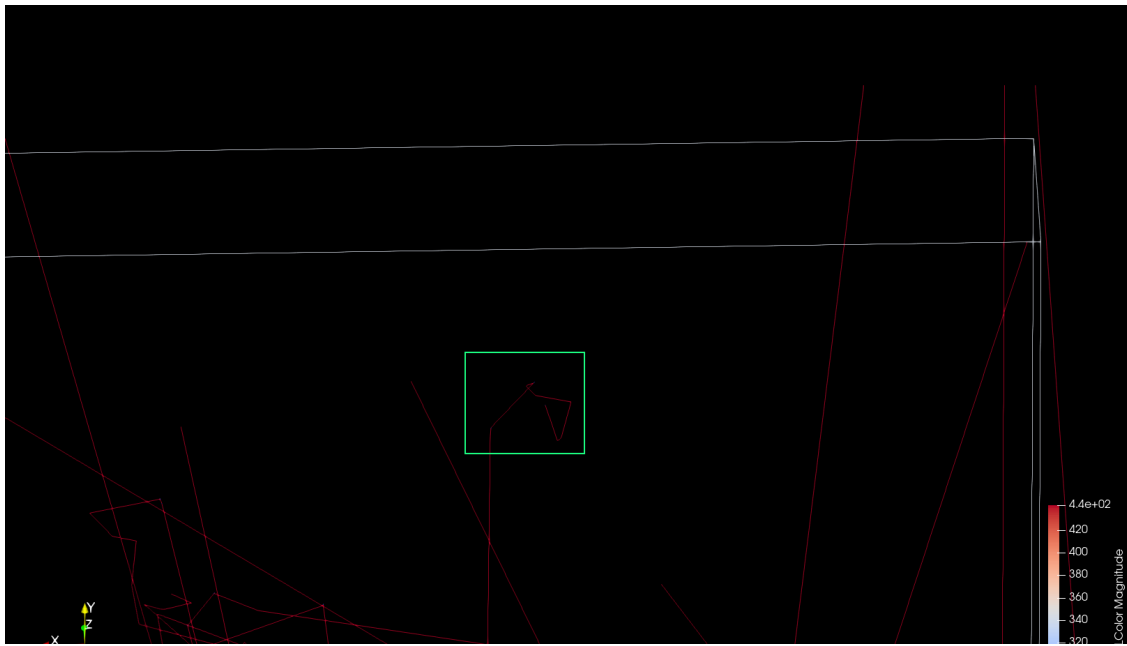
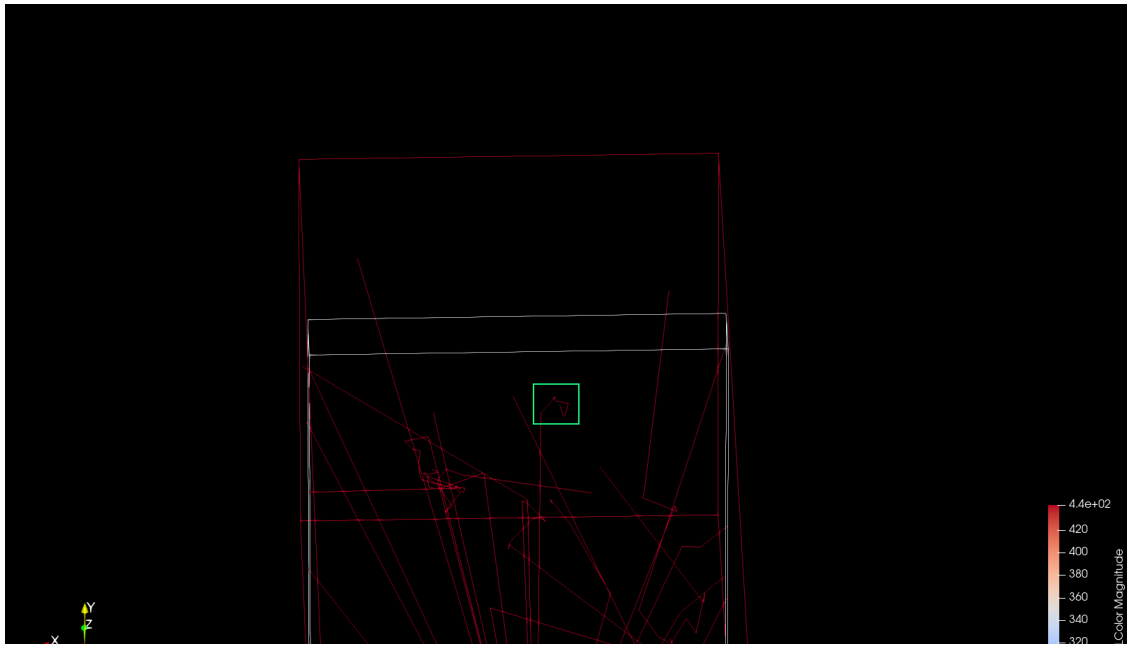
### 2.3 Efecto Compton

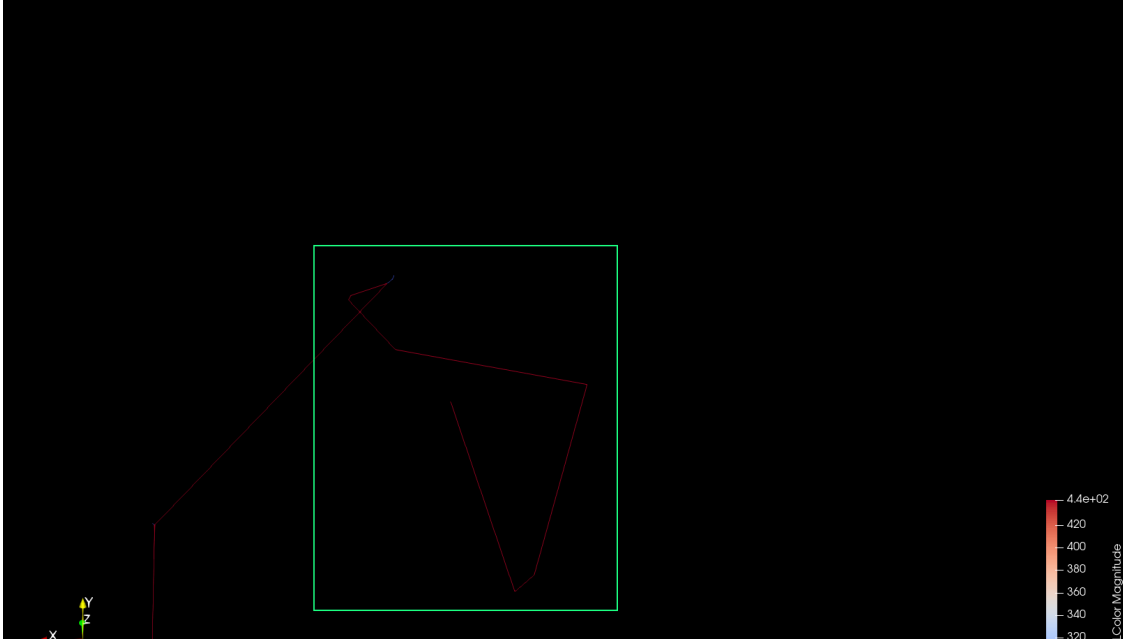
para comprobar el efecto Compton subimos la energía a 1 MeV y ejecutamos el siguiente comando:

```
docker run -v ${PWD}:/root pammacdotnet/g4lpwd --energy=1 --wrl=./simulacion/Compton.wrl
```

Se puede observar el efecto Compton en las siguientes imagenes





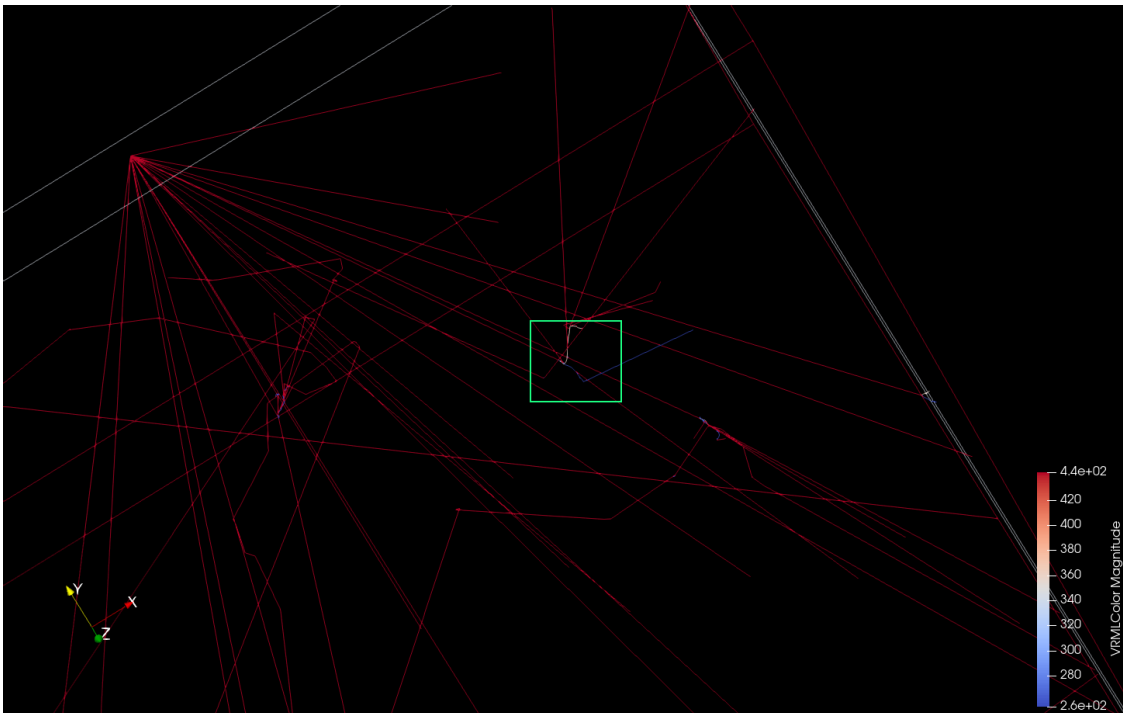


## 2.4 Producción de pares

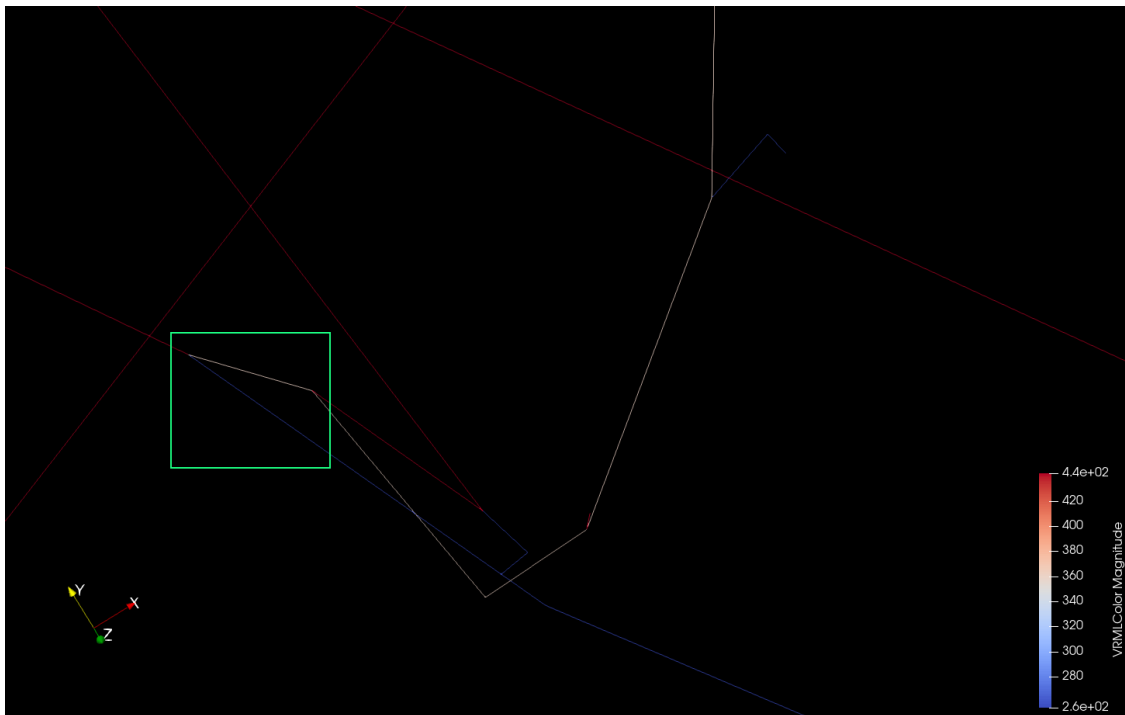
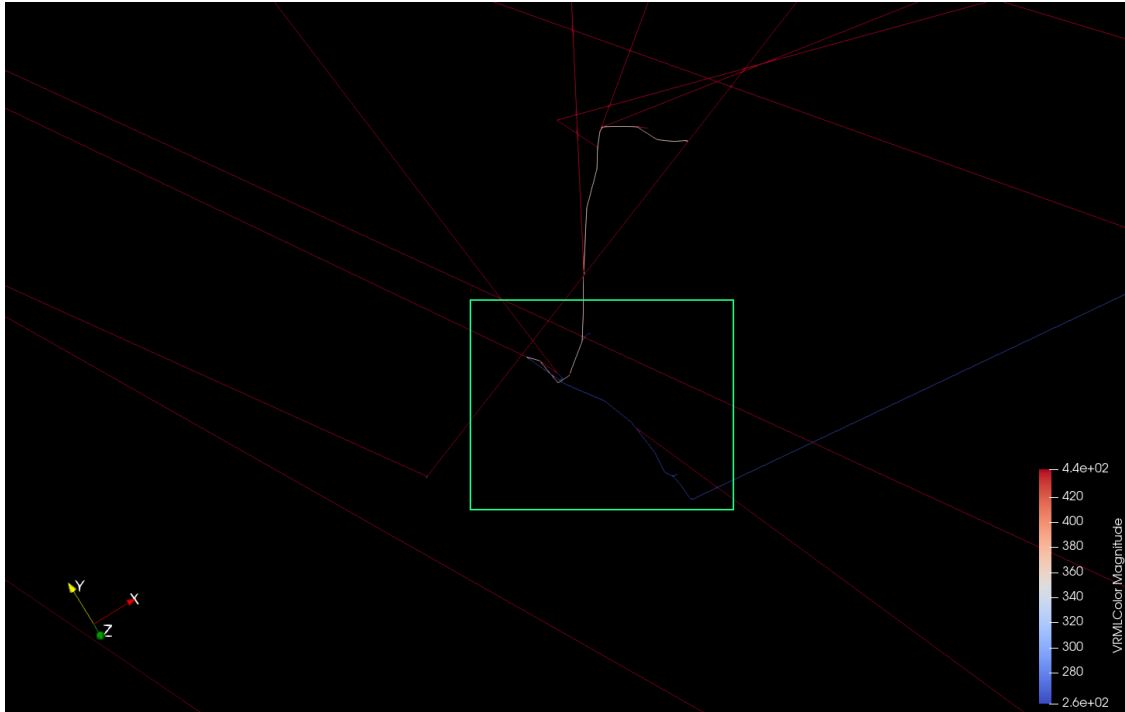
Para la producción de pares seguimos aumentando la energía, ejecutamos el comando:

```
docker run -v ${PWD}:/root pammacdotnet/g4lpwd --energy=40 --wrl=./simulacion/Pares.wrl
```

Se puede observar en las siguientes imagenes:

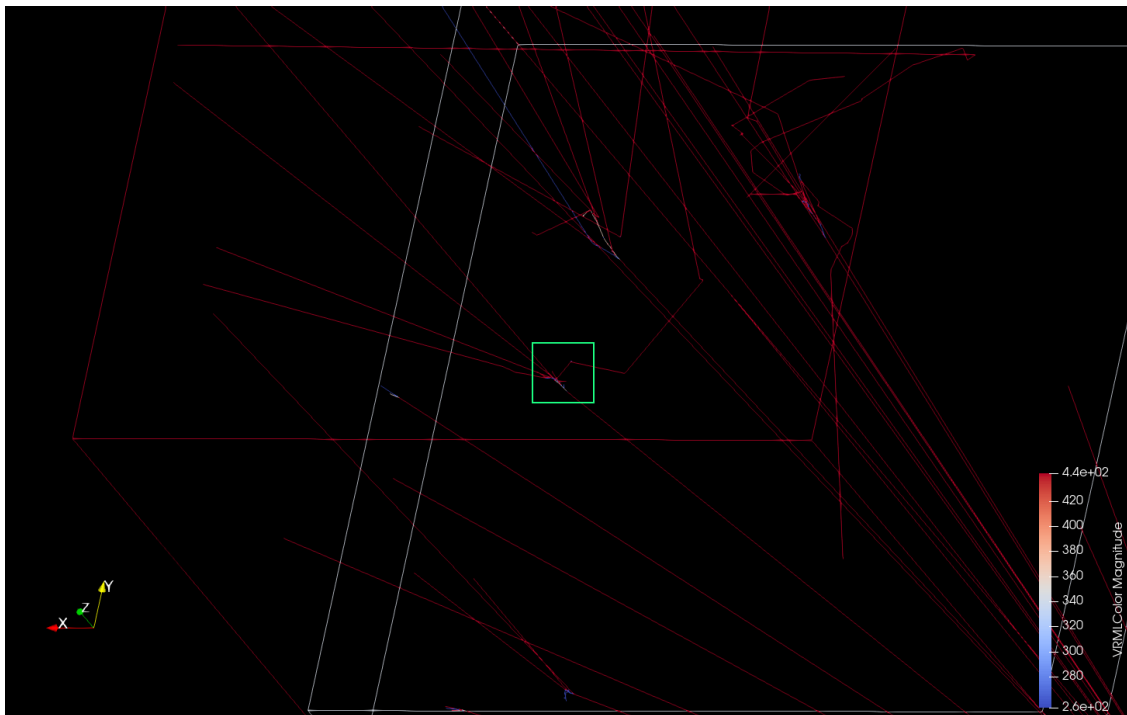
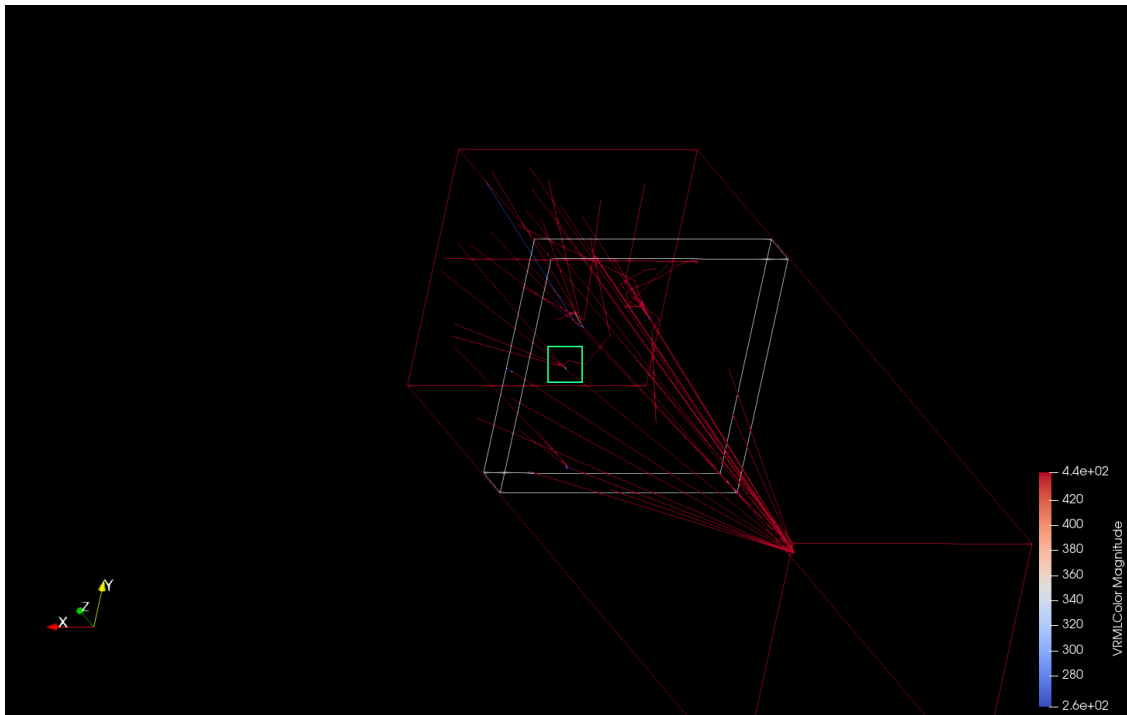


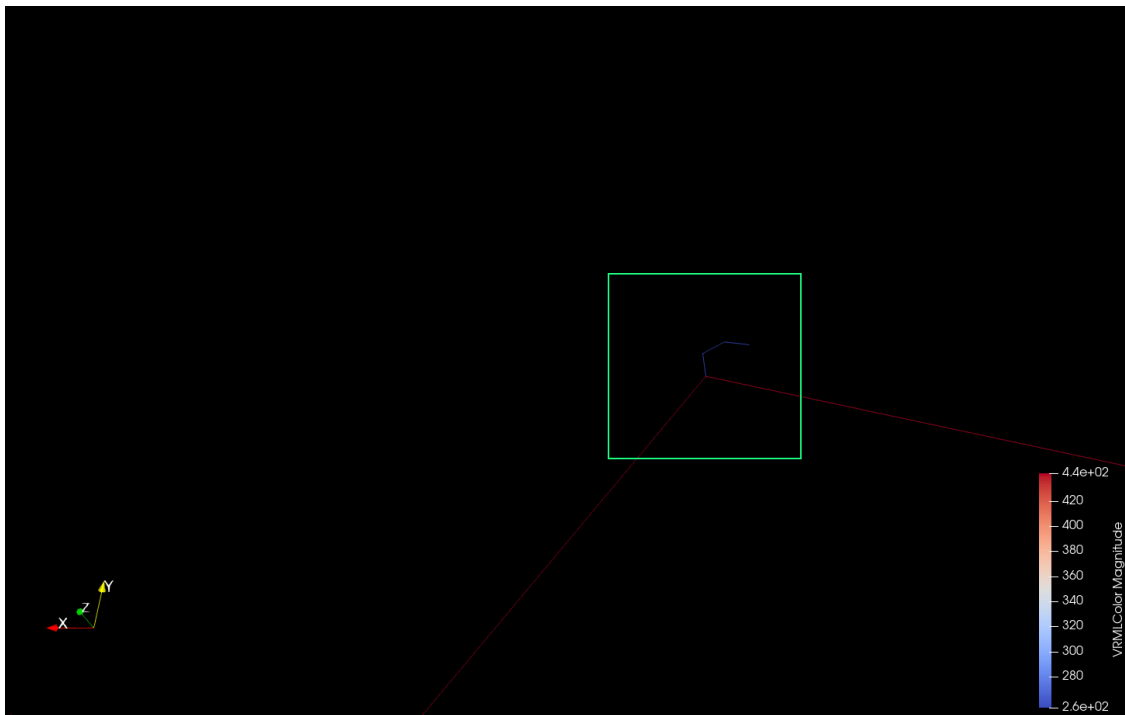
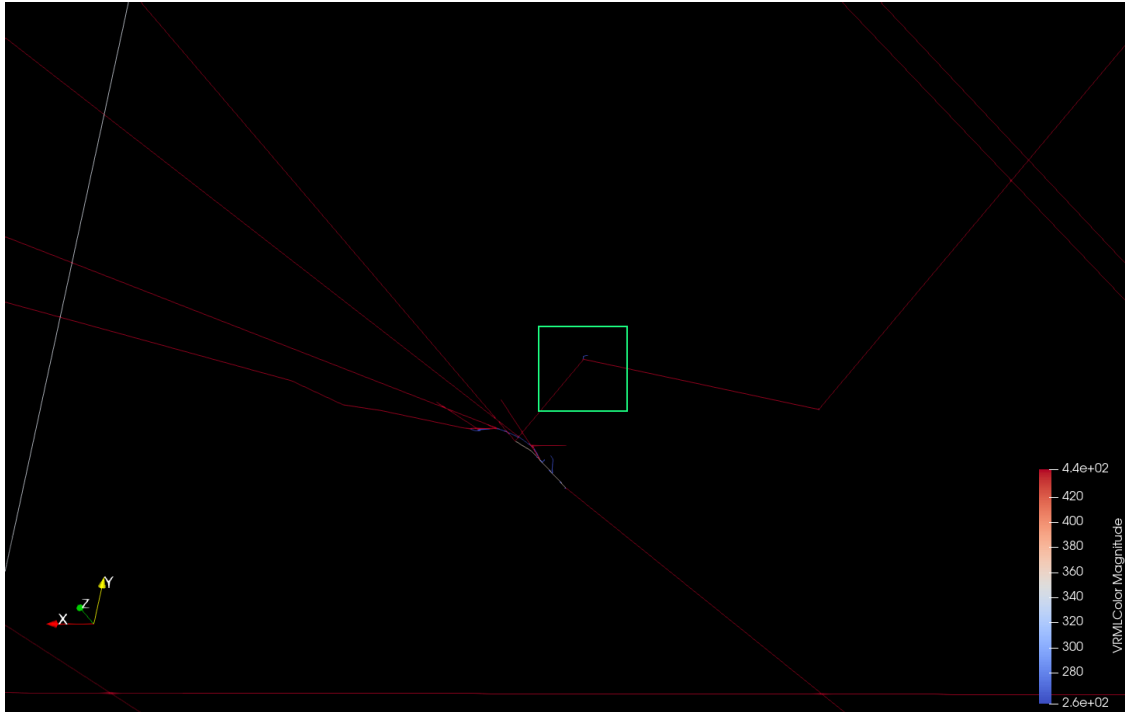




## 2.5 Ionización

En la misma simulacion que la produccion de pares se puede observar la Ionizacion:

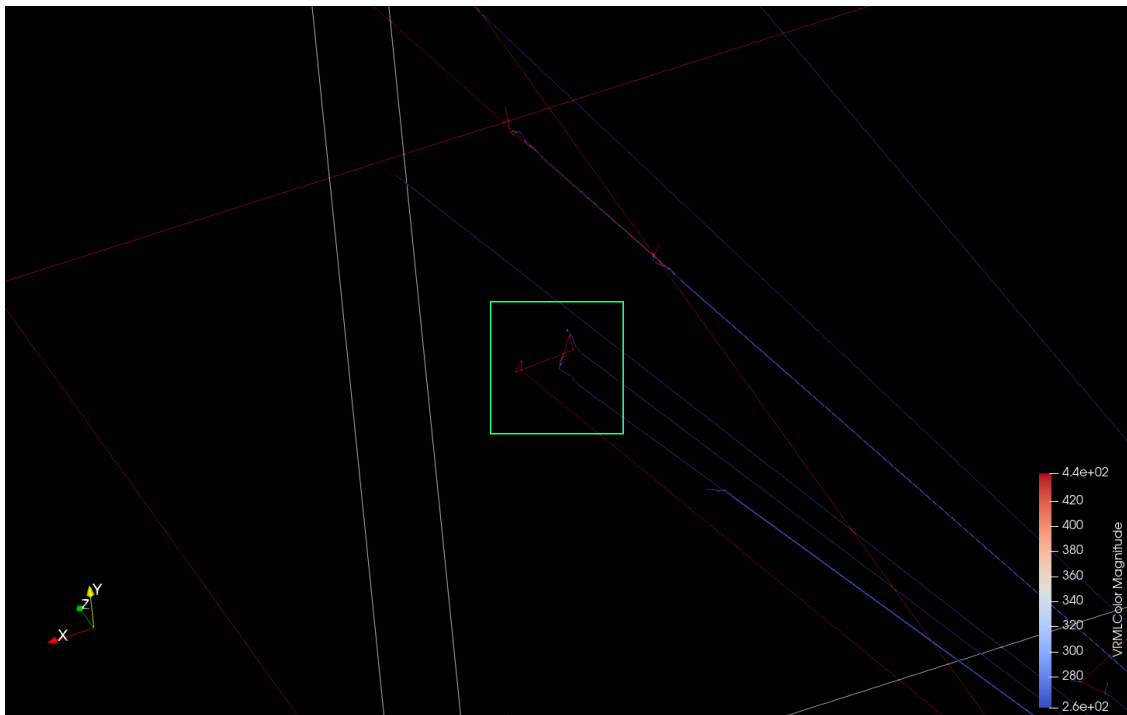
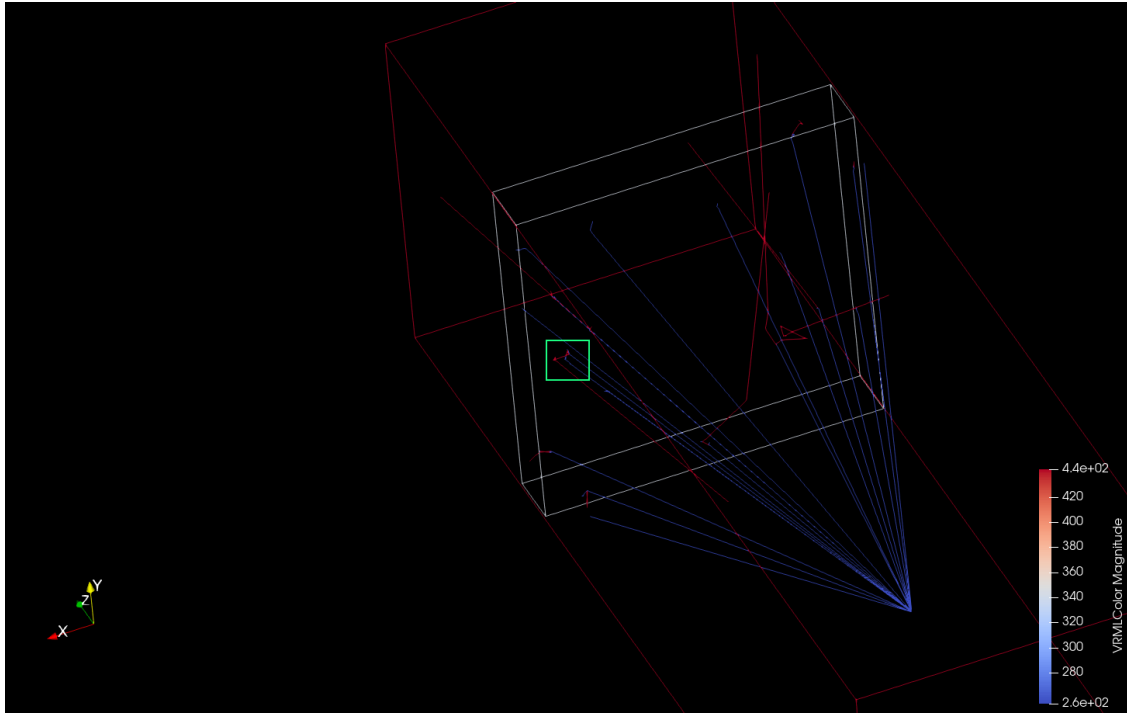


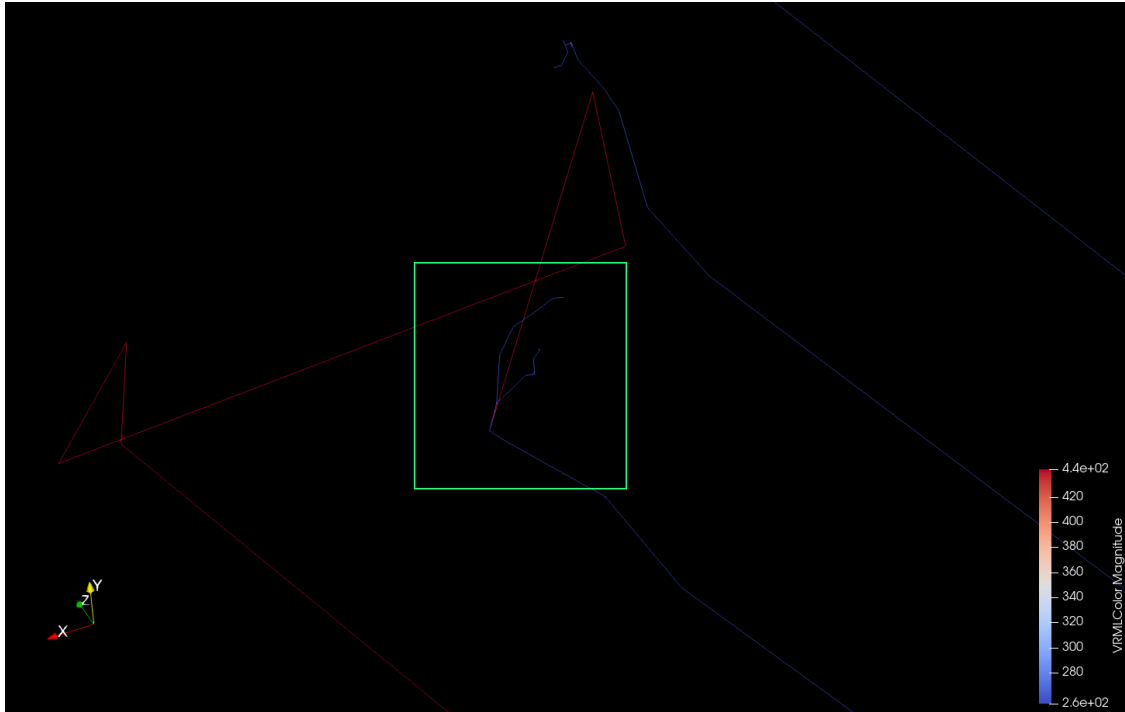


## 2.6 Bremsstrahlung

Para la siguiente simulacion cambiamos a electrones y ejecutamos:

```
docker run -v ${PWD}:/root pammacdotnet/g4lpwd --energy=10 --
wrl=./simulacion/Bremsstrahlung.wrl --type=e-
```





## 2.7 Aniquilación de pares

Para la aniquilacion cambiamos el tipo de particulas a positrones:

```
docker run -v ${PWD}:/root pammacdotnet/g4lpwd --energy=0.1 --  
wrl=./simulacion/Aniquilacion.wrl --type=e+
```

y podemos observar la aniquilacion de pares en las siguientes imagenes:

