Reunion hGATE
March 27th 2012

Optical Imaging Module:
status report from SHFJ

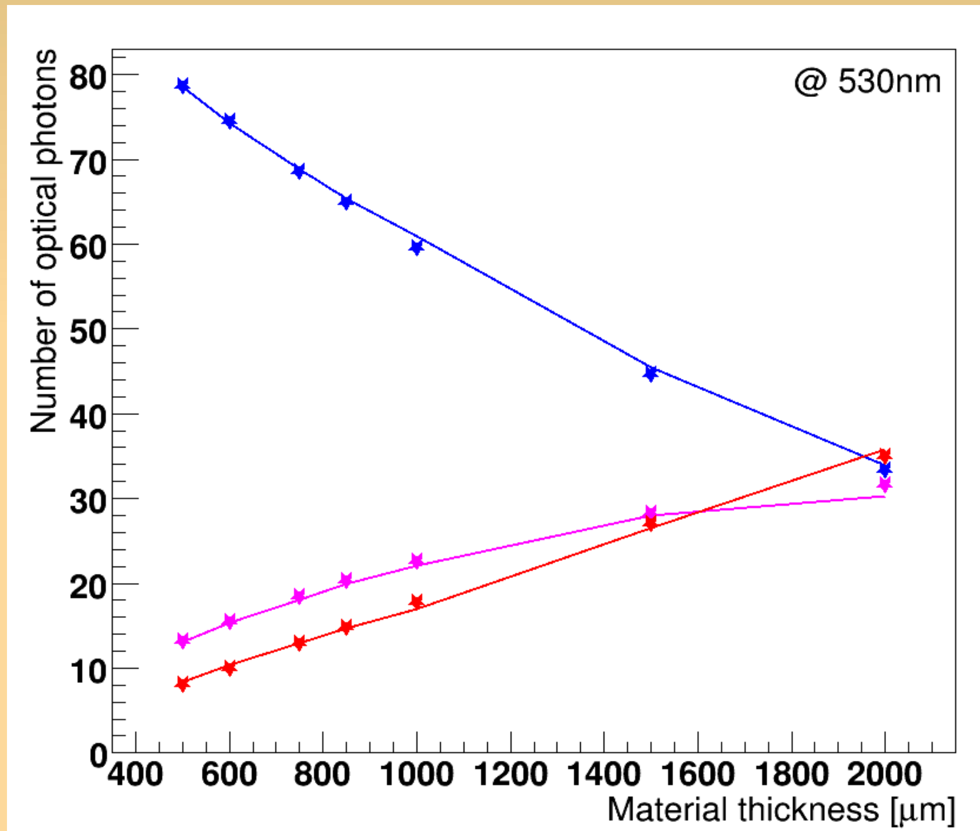
V. Cuplov, Sebastien Jan
SHFJ, Orsay

Outline

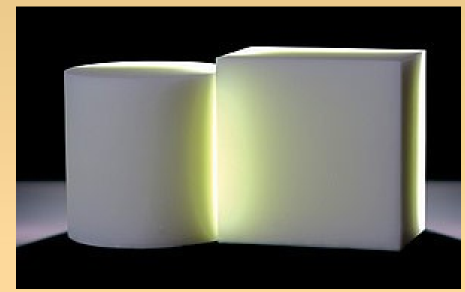
- Comparison between Gate and MCML
- Fluorescence process in Gate (New)
- Optical imaging *cuda* code status and plan:
  - Mie scattering
  - Optical processes at boundary
- Conclusion/Plan

Comparison between Gate and MCML

In Monte Carlo for Multi-Layered media (MCML), each layer has its own optical properties of absorption, scattering, anisotropy, and refractive index.



Biomimic optical phantom

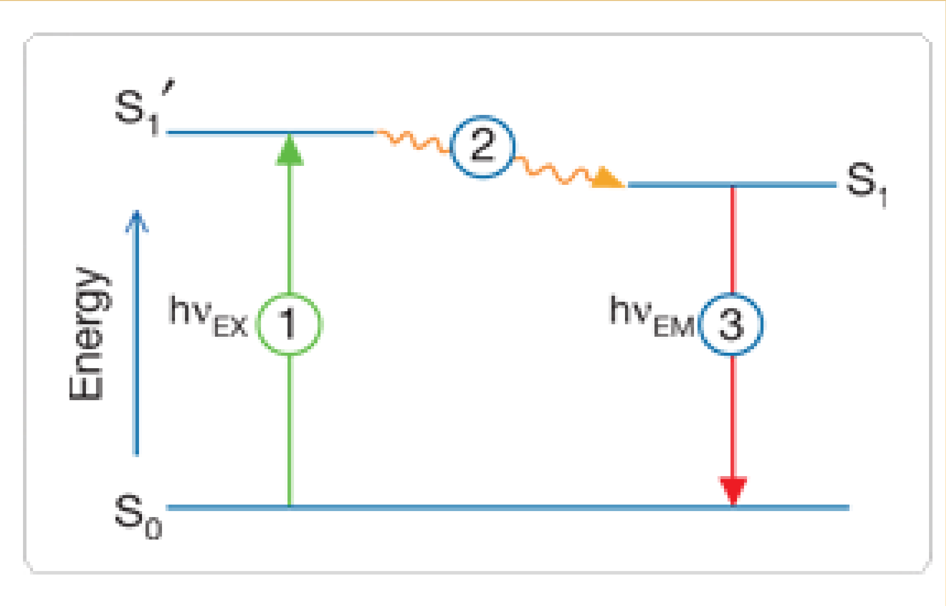


Excellent comparison between both codes!

Visible Light Fluorescence

Optical photons energy [1-4]eV or wavelength [800-300]nm.

1) Photon from external source is Absorbed by the fluorophore: Excited state



3) Photon is emitted (lower energy – longer λ than excitation photon)

2) Lifetime of the excited state is 1-10 ns – the fluorophore interacts with environment: Relaxed excided state

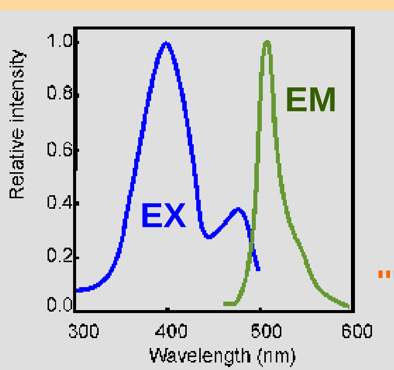
Simulation of the Green Fluorescence Protein with Gate

Wave Length Shifting (Geant4)

- GateOpticalWLSPB with ProcessName "OpticalWLS"
- Inherits from G4OpWLS class.

/gate/physics/addProcess OpticalAbsorption
/gate/physics/addProcess OpticalRayleigh
/gate/physics/addProcess OpticalBoundary
/gate/physics/addProcess OpticalMie (Feb. 16th 2012)
/gate/physics/addProcess OpticalWLS (New! Soon in SVN)

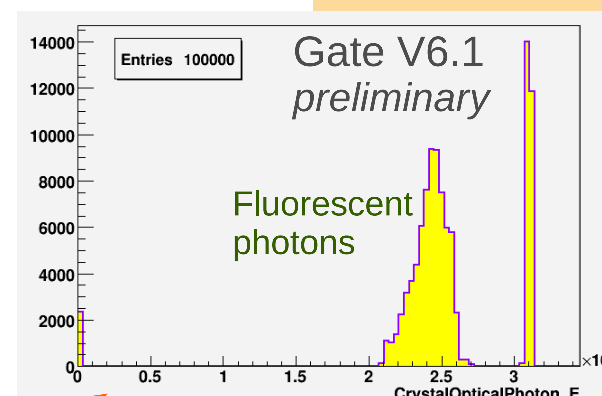
WLS fibers collect light produced in HEP hadronic calorimeters. They absorb blue and re-emit green so that as much light reaches PMTs.



GFP excitation and emission spectra from litterature

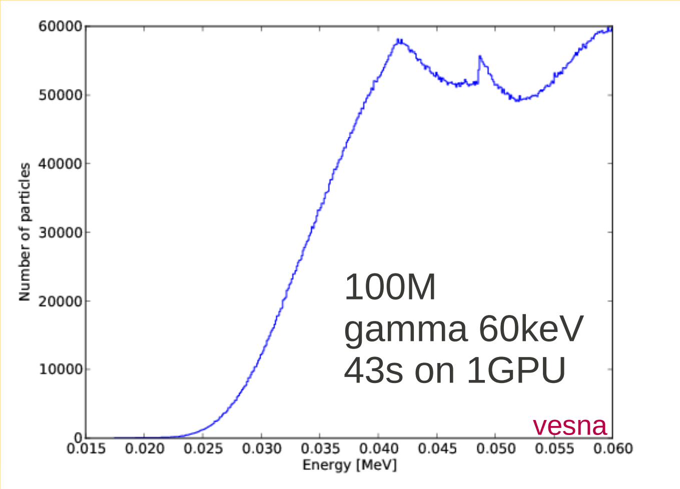
```
<propertyvector name="WLSABSLENGTH" unit="mm" energyunit="eV">
   <ve energy="3.9" value="50"/>
   <ve energy="3.6" value="50"/>
   <ve energy="3.4" value="50"/>
   <ve energy="3.2" value="50"/>
   <ve energy="3.11" value="0.4"/>
   <ve energy="3.1" value="0.4"/>
   <ve energy="3.09" value="0.4"/>
   <ve energy="2.9" value="50"/>
   <ve energy="2.8" value="50"/>
   <ve energy="2.7" value="50"/>
   <ve energy="2.6" value="50"/>
   <ve energy="2.5" value="50"/>
   <ve energy="2.4" value="50"/>
   <ve energy="2.3" value="50"/>
   <ve energy="2.2" value="50"/>
   <ve energy="2.1" value="50"/>
   <ve energy="2.1" value="50"/>
</propertyvector>
```

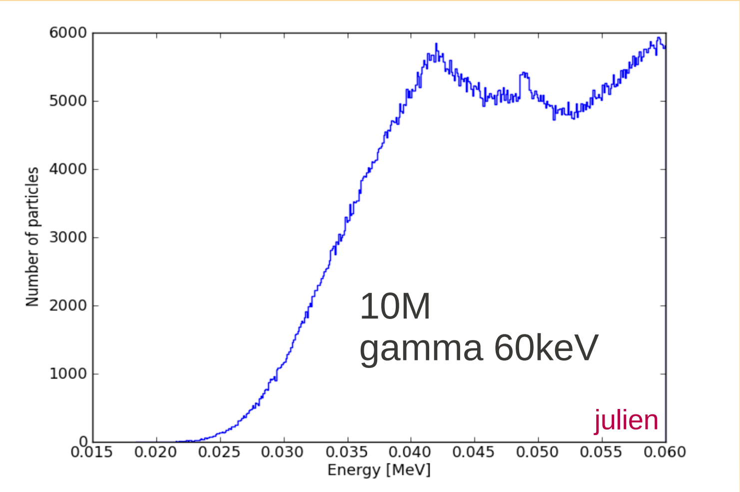
```
</propertyvector>
<propertyvector name="WLSCOMPONENT" energyunit="eV">
   <ve energy="3.9" value="0.0"/>
   <ve energy="3.6" value="0.0"/>
   <ve energy="3.4" value="0.0"/>
   <ve energy="3.2" value="0.0"/>
   <ve energy="3.1" value="0.0"/>
   <ve energy="2.9" value="0.0"/>
   <ve energy="2.8" value="0.0"/>
   <ve energy="2.7" value="0.0"/>
   <ve energy="2.6" value="4"/>
   <ve energy="2.5" value="75"/>
   <ve energy="2.4" value="50"/>
   <ve energy="2.35" value="35"/>
   <ve energy="2.3" value="25"/>
   <ve energy="2.25" value="20"/>
   <ve energy="2.2" value="10"/>
   <ve energy="2.1" value="4"/>
   <ve energy="2.1" value="2"/>
```



hgate cuda code:
testing the machinery at CCRT

- Many thanks to Julien Bert who provided me with a package: hgate_v03_livermore_standard
- Water box with an isotropic 60keV gamma source placed at the center. Physics is Rayleigh, Compton and PhotoElectric effect.
- Testing the code out-of-the-box on the Cesium (Nvidia 1.3) machine at CCRT*: successful!
- Reproducing the Energy spectrum of simulated gammas:





* Centre de Calcul Recherche et technologie - CEA

Mie scattering cuda code: status and plan

- Recipe for the Implementation of the Mie Scattering process
  - Table of scattering length and anisotropy (mc_cst.cu).
  - Choose a new deflection angle (θ) for the photon propagation according to the Henyey-Greenstein approximation (mc_fun.cu):

$$p(\cos\theta) = \frac{1 - g^2}{2x(1 + g^2 - 2g\cos\theta)^{3/2}}$$

g: anisotropy

  - Calculation of the mean free path is done in the woodcock tracing function (mc_fun.cu)

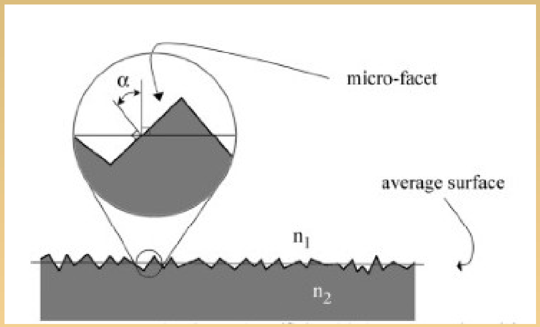
$- \log \mu(E)$ with $\mu(E) = 1/CS(E)$ → In order to follow the existing code implementation, we might actually create a Table of CS instead of μ.

Code writing (Mie scattering in water) + validation against Gate CPU by the end of April.

Reminder: G4Surface concept

Choose the Surface finish: polished or ground

no surface needed
only 2 media Rindex



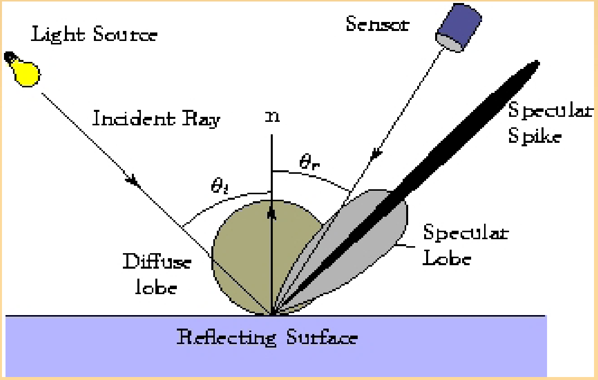

```
<surface name="MySurface" type="dielectric_dielectric" sigmaalpha="20.0" finish="ground">
<propertiestable>
    <propertyvector name="SPECULARLOBECONSTANT" energyunit="eV">
      <ve energy="2.34" value="0.0"/>
      <ve energy="1.97" value="0.0"/>
    </propertyvector>
      <propertyvector name="SPECULARSPIKECONSTANT" energyunit="eV">
      <ve energy="2.34" value="0."/>
      <ve energy="1.97" value="0."/>
    </propertyvector>
    <propertyvector name="BACKSCATTERCONSTANT" energyunit="eV">
      <ve energy="2.34" value="0."/>
      <ve energy="1.97" value="0."/>
    </propertyvector>
<!--      <propertyvector name="REFLECTIVITY" energyunit="eV">
      <ve energy="2.34" value="1.0"/>
      <ve energy="1.97" value="1.0"/>
    </propertyvector>-->
  </propertiestable>
</surface>
```

Reflection about the normal of a microfacet

Reflection about the AVERAGE surface normal

Exact back-scatter

Example of 100% Lambertian Reflection



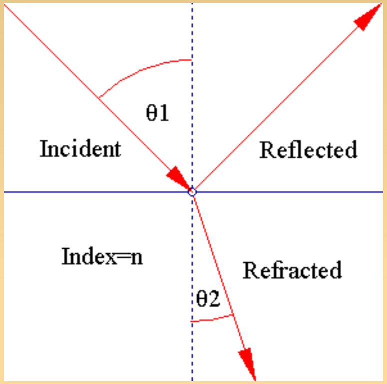
If not 100% Lambertian (diffuse), you can simulate specular and ultimate back-scatter.

"Not all rough surfaces are perfect lambertian reflectors, but it is a good approximation if surface characteristics are unknown." (Geant4 Developper guide).

Question: Can we bypass the G4surface concept for processes at boundary in the cuda implementation?

Processes at Boundary: cuda implementation plan

- **Fresnel partial transmission and reflection equations.**
- Compute Fresnel Reflectance (incident and transmit Rindex, cosine of incident and transmit angle) using Snell's law:



$$n_i \sin \theta_i = n_t \sin \theta_t$$

$$R(\alpha_i) = \begin{cases} \dfrac{(n_t - n_i)^2}{(n_t + n_i)^2}, & \text{if } \alpha_i = 0 \quad \text{normal incident} \\ \dfrac{1}{2}\left[\dfrac{\sin^2(\alpha_i - \alpha_t)}{\sin^2(\alpha_i + \alpha_t)} + \dfrac{\tan^2(\alpha_i - \alpha_t)}{\tan^2(\alpha_i + \alpha_t)}\right], & \text{if } 0 < \alpha_i < \sin^{-1}\left(\dfrac{n_i}{n_t}\right) \\ 1, & \text{very slant} \quad \text{if } \sin^{-1}\left(\dfrac{n_i}{n_t}\right) < \alpha_i < \dfrac{\pi}{2} \end{cases}$$

- Decide whether the photon will be transmitted or reflected on the boundary using $R(\alpha_i)$:
  - Total internal reflection (reflection)
  - Partially transmitted (refraction plus reflection)

Implementation: Rindex Table – Fresnel equations – give new direction for photon propagation.

Conclusion/Plan

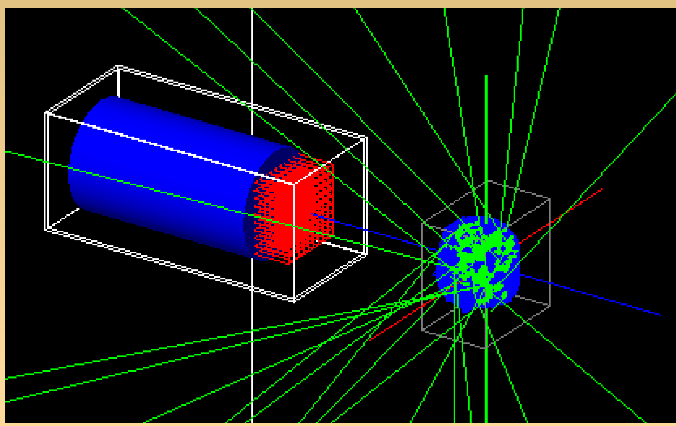
- Gate Transmittance, Reflectance and Absorbance have been validated against MCML.
- Fluorescence is now available in Gate (will be commited soon into SVN).
- We have new GPU ressources! (GENCI Project: 1.5M hours on Curie machine @ TGCC – 2 GPU Nvidia 2.0 per node).
- hgate cuda code works out-of-the-box.
- Mie scattering has been studied and its cuda implementation will *hopefully* be delivered by the end of april.
- Optical processes at boundary: hgate cuda code implementation is under study/discussion.

Backup

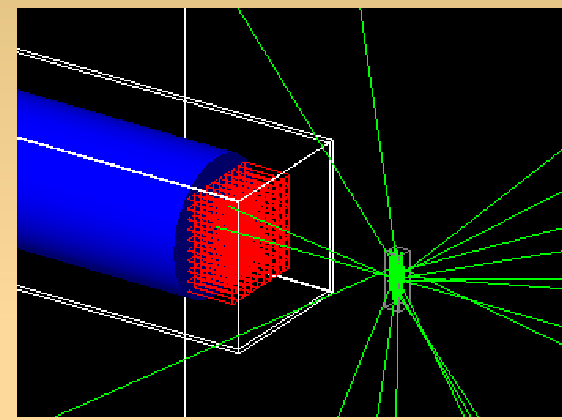
- If we have 1-2 more minutes...

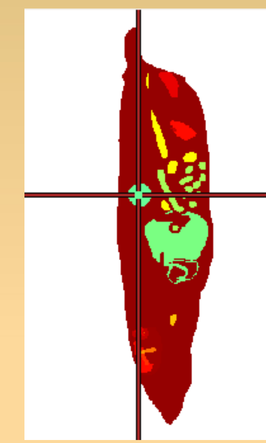
Playing with various optical imaging benchmarks

Voxelized Phantom (Air and GFP)
Voxelized Source (3.1eV optical photon)

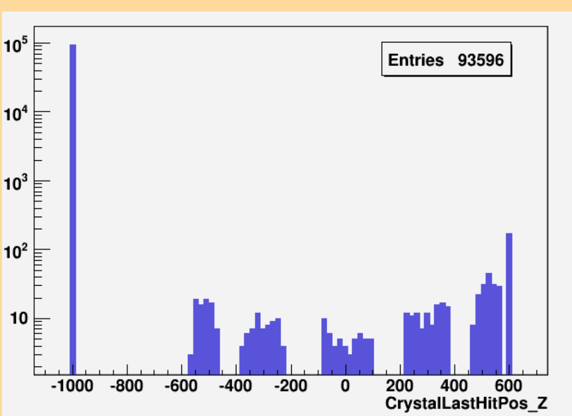


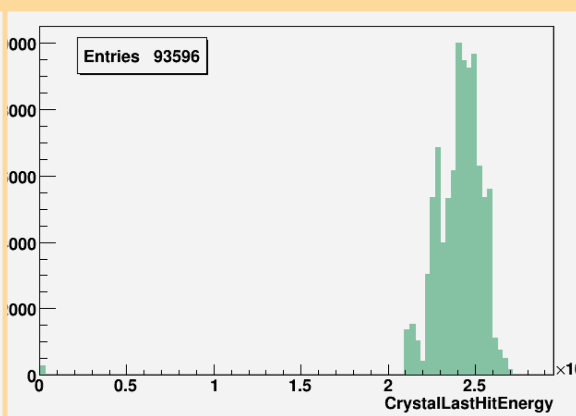
Moby Phantom with GFP sphere inserted
optical photon source (3.1eV)





Detector is a GATE Scanner System: pixel array and a PMT cylinder. It orbits around x-axis.





Testing DAQ, orbiting, projections... on multi processors at the CCRT.

Images and output files need further analysis.

GFP: Green Fluorescence Protein