Updates to GeGeDe for nu-opticks

Any anyone else that wants them

Brett Viren and Claude the Bot

April 13, 2026

Overview

- Add support for **optical properties**.
  - Added a **full G4 app and viz** as a test.
- Test out proper GDML **sensitive detector** marking.
- Fixed some bugs, added some docs.

Optical properties for bulk matter in a GeGeDe builder

```
# converts wavelength to proper energy units
from gegede.util import wl

# Glass with dispersive index of refraction
geom.matter.Molecule("Glass", ..., properties = [
    ("RINDEX", [
        (wl(700), 1.570),   # red
        (wl(620), 1.579),   # orange
        (wl(580), 1.585),   # yellow
        (wl(530), 1.596),   # green
        (wl(490), 1.609),   # cyan
        (wl(450), 1.626),   # blue
        (wl(400), 1.655),   # violet
    ])])
```

Optical properties for surface in a GeGeDe builder

```
ELO = Q(2.034, "eV"); EHI = Q(4.136, "eV"); EPEAK = Q(3.06, "eV")
geom.surfaces.OpticalSurface(
    "pmt_surface", # <-- referenced by name
    model="unified", finish="polished", type="dielectric_metal",
    value=1.0,
    properties = [
        # fully absorbing
        ("REFLECTIVITY", [(ELO, 0.0),  (EHI, 0.0)]),
        ("EFFICIENCY",   [(ELO, 0.05), (EPEAK, 0.25), (EHI, 0.08)]),
    ])
# SkinSurface covers all physical instances of pmt_vol automatically
geom.surfaces.SkinSurface("pmt_skin",
                          surface="pmt_surface", volume="pmt_vol")
# BorderSurface applies to a directed interface between two placements
geom.surfaces.BorderSurface("air_steel_border",
                            surface="steel_air_surface",
                            physvol1="air_in_steel",
                            physvol2="air_in_steel")
```

gegede/example/optical/ - a G4 app & GeGeDe geometry

Steel cube box, water, glass prism and PMTs (also glass).

- Steel has REFLECTIVITY, PMT has (zero) REFLECTIVITY and EFFICIENCY
- Glass and water has RINDEX, water has absorption/scatter.

Sensitive detectors

The PMT is a **sensitive detector** defined in GeGeDe like:

```
pmt_vol = geom.structure.Volume(
            "pmt_vol",
            material = "pmt_glass",
            shape    = "pmt_hemi",
            params   = [("SensDet", "PMTDetector")])
```

And exports to GDML as:

```
<volume name="pmt_vol">
  <materialref ref="pmt_glass"/>
  <solidref ref="pmt_hemi"/>
  <auxiliary auxtype="SensDet" auxvalue="PMTDetector"/>
</volume>
```

Warning: `"PMTDetector"` requires **collusion** between GeGeDe and C++/G4 codeed strings! Is this better than how LArSoft requires certain spellings of volume names?

Build and run three jobs

```
cmake -S g4app -B g4app/build \
      -DGeant4_DIR=/path/to/geant4/lib/cmake/Geant4
cmake --build g4app/build

uv run gegede -o optical_prism.gdml optical.cfg

./g4app/build/optical_g4 macros/run.mac optical_prism.gdml
```

`run.mac` collect hits on PMTs.

`run_traj.mac` collect photon trajectories.

`dsotm.mac` Dark Side of the Moon

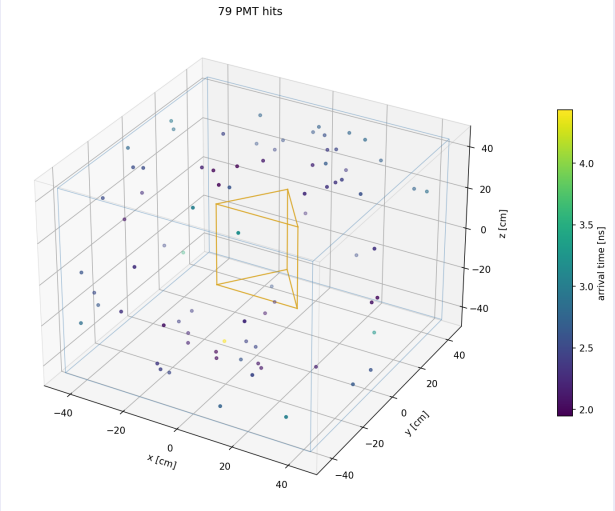
Run 1: PMT Hits Isotropic Spectrum

Macro (`macros/run.mac`)

1000 isotropic photons from the center; no trajectory storage.

- Output: `hits.csv` (~80 detections, ~8 % detection rate).
- Visualise: `g4app/python/plot_hits.py`

Result



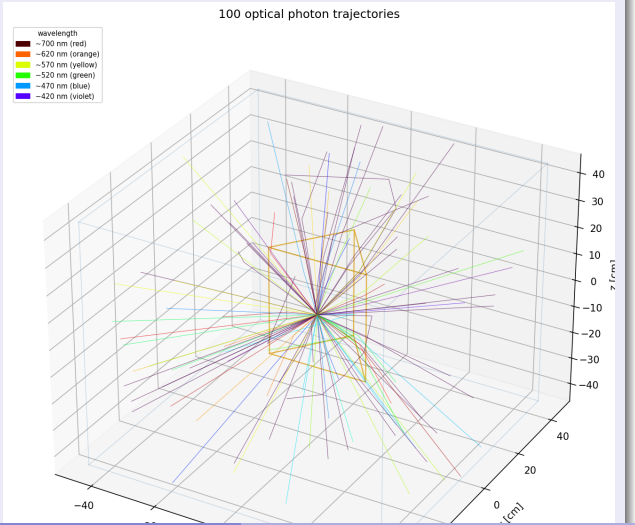
Run 2: Trajectories - Isotropic Spectrum

Macro (`macros/run_traj.mac`)

50 photons; every step point recorded.

- Output: `hits.csv` + `trajectories.csv`.
- Visualise: `g4app/python/plot_traj.py`

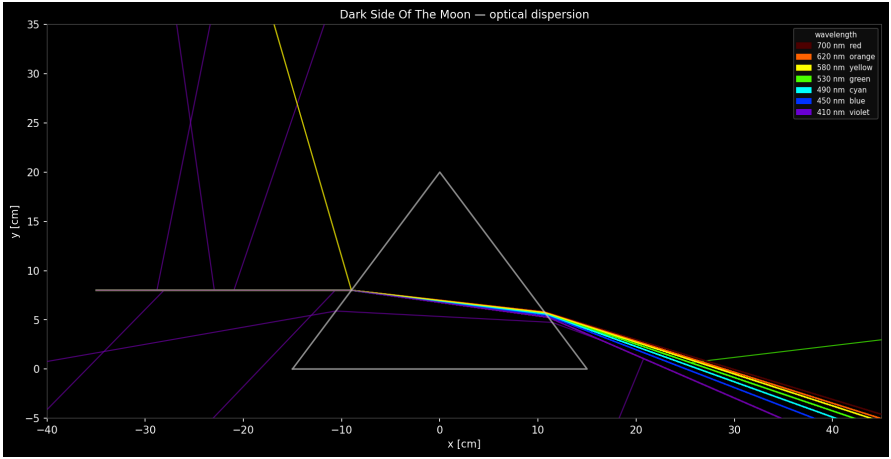
Result



Run 3: Dark Side Of The Moon - Dispersion Demo

Parallel “white” beam enters prism left face; dispersive glass separates colors.

- Visualise: `g4app/python/plot_traj.py --view dsotm`



Caveat

Planning by Claude Opus and implementing by Sonnet.

- 4 hours, 31M tokens, $21.41.

Directed and checked by me.

- Drank coffee, read email, watched Emacs videos on youtube.

FIN