

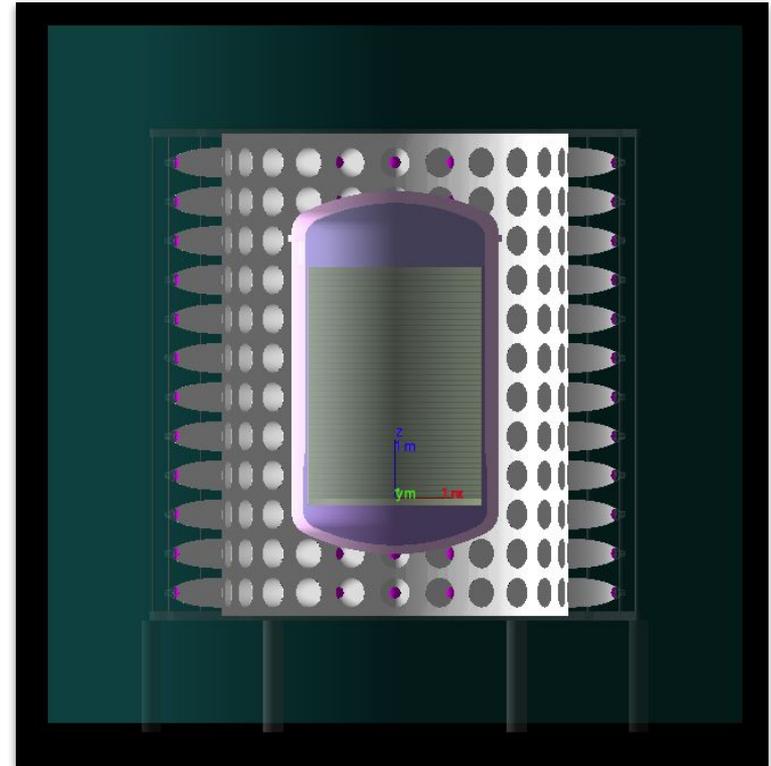
Intro to XLZD Sandbox



THE UNIVERSITY
of EDINBURGH

Sam Woodford
04/02/26

- XLZD Sandbox is a Geant4-based simulation framework
- Initially designed for quick and easy testing of different geometry configurations for a large dark matter detector
- Now adopted as the main simulation framework for the collaboration
- Has since been used for multiple analyses:
 - Validation of CAD models from engineers
 - Veto efficiency studies in the outer detector
 - Light collection efficiency studies
 - Estimation of backgrounds to physics searches
 - Etc.



Building the framework using the Edinburgh PPE server is very simple:

- Start a ssh session: `ssh <username>@phcomputepe01.ph.ed.ac.uk`
- Enter an image using singularity: `singularity shell -B /cvmfs docker://xlzdg3/xlzd_sandbox:rocky9`
- Clone the code from git: `git clone git@gitlab.com:XLZD-UK/xlzd-sandbox.git`
- Set up the environment: `cd xlzd-sandbox/; source setup.sh`
- Build: `make`

[Git link](#)

- To run a simulation, generally only need two files: a detector file (`.json`), and a physics macro (`.mac`)

```
{
  "outerDetector": {
    "Height": 12,
    "Radius": 6,
    "SteelShell": {
      "Status": true,
      "Thickness": 0.05
    },
    "GdPercent": 0.1,
    "Medium_options" : {
      "0" : "No LAB",
      "1" : "WbLS with 1% concentration of LAB",
      "2" : "WbLS with 5% concentration of LAB",
      "3" : "WbLS with 10% concentration of LAB"
    },
    "WbLS_option": 0,
    "Medium":{
      "DopedLABGd": false,
      "DopedWater": true
    },
    "FMT": {
      "Type_options" : {
        "0" : "8 inch",
        "1" : "10 inch",
        "2" : "20 inch"
      },
      "Type": 0,
      "Status": true,
      ...
    }
  }
}
```

```
# XLZD Manager options
/XLZD/setGeometry detector.json
/XLZD/outfile neutronsOCV.root
/XLZD/useOpticalPhysics true
/XLZD/Stepping false
/XLZD/ICVBottomShape elliptical
/XLZD/TD28030998 true
/XLZD/TD28030998/JSON true

# Initialise run
/run/initialize

# Set particle type
/gps/particle neutron

/gps/ene/type Arb
/gps/hist/type arb
/control/execute ../Ti-U-n.mac
/gps/hist/inter Lin

/gps/ang/type iso
/gps/pos/type Volume
/gps/pos/shape Sphere
/gps/pos/radius 9 m
/gps/pos/confine OCV_proper

# Run number of particles
/run/beamOn 1000000
```

A next-generation dark matter detector may take many forms...

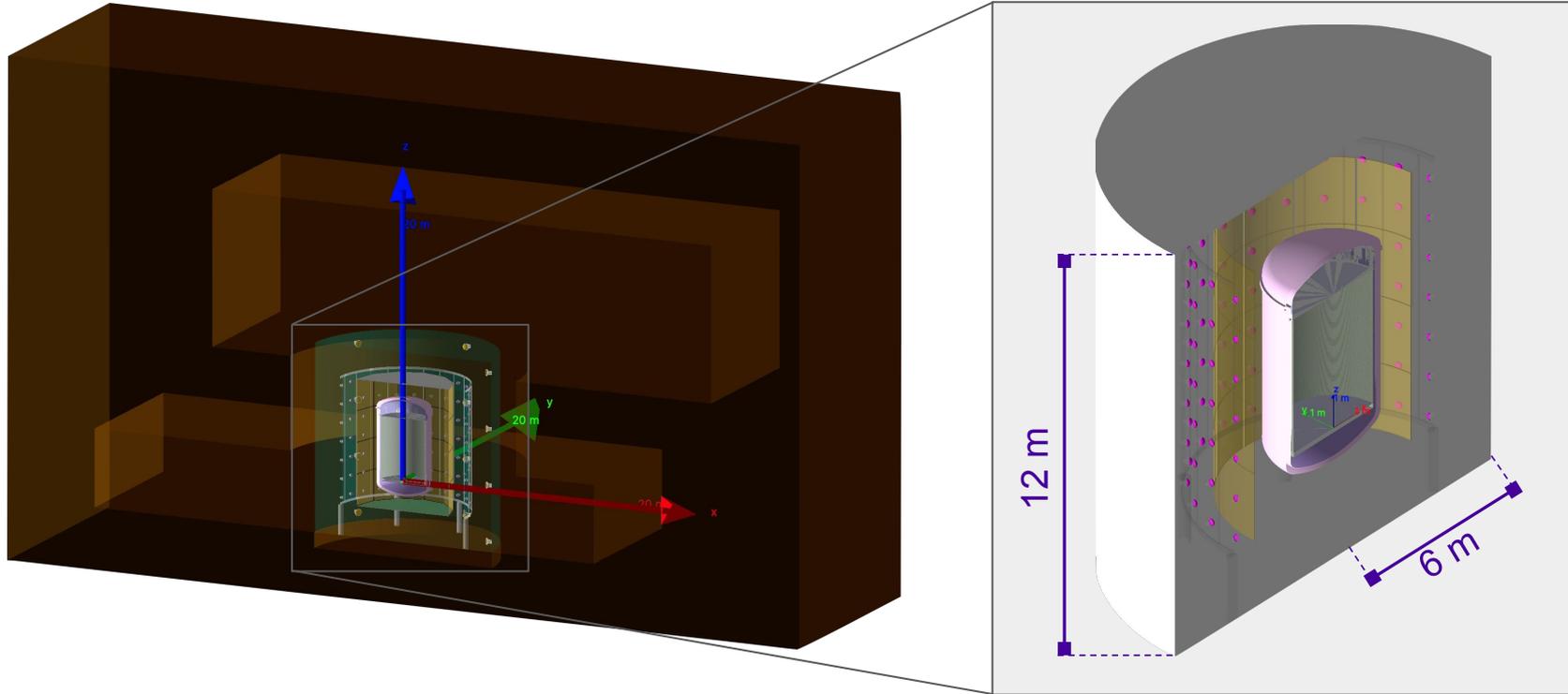
Basic Essentials

- Centrepiece: liquid xenon TPC
 - Instrumented by PMTs in arrays at the top and bottom
- Outer detector shield/veto
 - Large volume surrounding the TPC to veto neutrons and gammas that may have coincident signals with the TPC, instrumented by larger PMTs

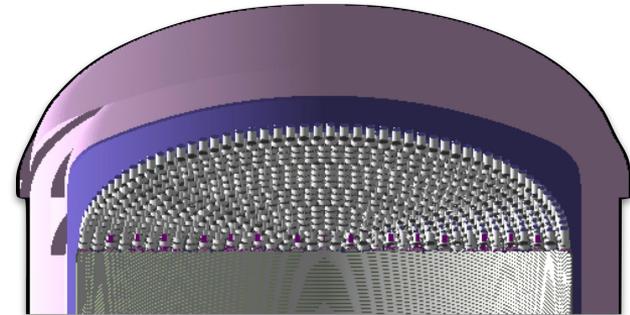
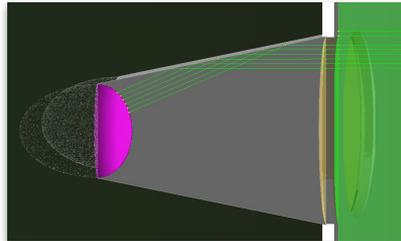
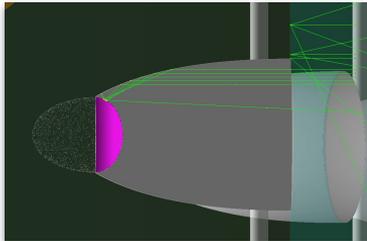
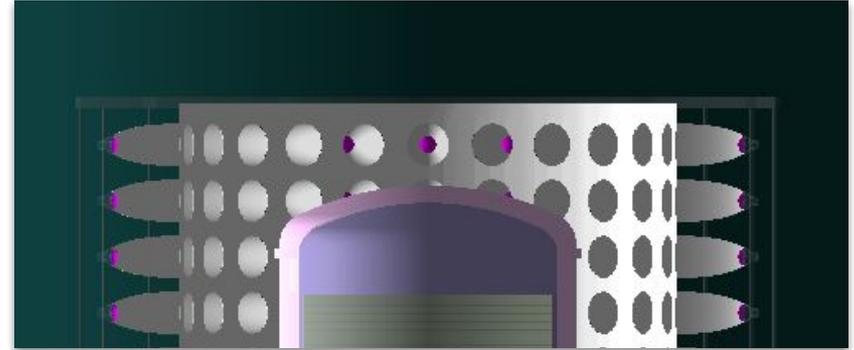
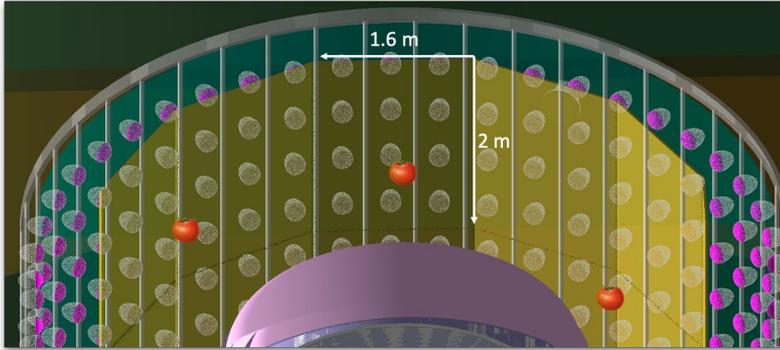
Variables

- Volume sizes: how much LXe can we use as the target volume?
- Fill media: shall the OD be filled with doped water or liquid scintillator?
- Site location: which lab should be used to host the detector?
- Instrumentation: how many PMTs shall we use?
- Backgrounds: how do detector materials contribute to signal background?
- Etc.

XLZD Geometry



A wide range of detector features may be modelled by modifying the detector file only:



In order to simulate physics, the simulation needs to know what physics to include

- **Physics lists** contains the details of how a particle(s) will interact

Why isn't there just 'include physics'?

- A model that describes what a particle does/can do is unlikely to perfectly describe it at all energies. So there are alternative physics models
- If your simulation only cares about the decay of a particular isotope and not what future interactions happen, why would you bother including things that aren't that?

There are broadly 2 types of physics lists

1. Reference
 - a. These are ones that come with Geant4
2. Custom (user defined)
 - a. User defined parameters for interactions

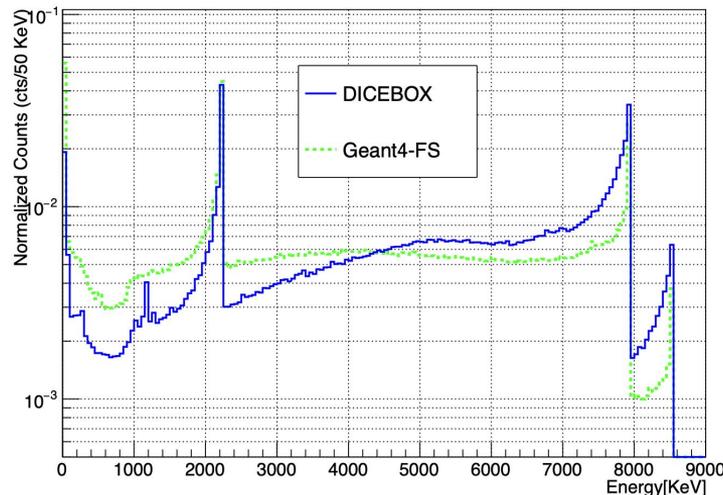
In the simulation we use a mixture of both

Physics Lists: Example



- When a neutron is captured on Gd, the simulation needs to know the nature of the gammas (multiplicity & energy) that are emitted
- Use `G4ParticleHPCaptureFS_Gd` instead of the reference class
 - Experience from LZ informs us G4 doesn't handle Gd deexcitation well (at least pre-G4.11.0)
 - Using information from DICEBOX, we have a series of files which contain more accurate gamma information

Example from LZ



Material Properties



- A crucial feature of the Geant4 program is the definition of material properties. Material properties govern the generation and propagation of optical photons within a volume.
- In Geant4, this is commonly handled by a G4MaterialPropertiesTable (MPT). Properties in general may be independent of photon energy, or may be expressed as a function of the photon's energy.

Example from XLZDMaterials: GdLS

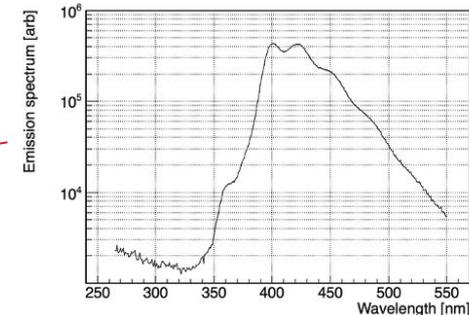
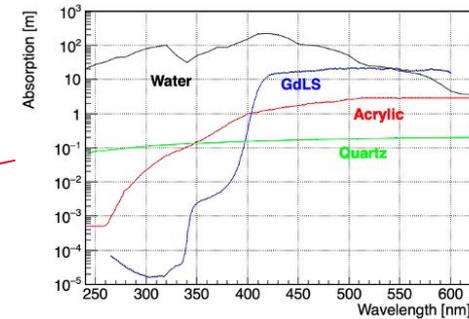
Refractive index

Absorption length

Rayleigh scattering

Emission spectrum

```
// Scintillator
dopedLABGdMat = new G4MaterialPropertiesTable ();
AddProperty (ptr: dopedLABGdMat,
             key: "RINDEX",
             PhotonEnergies: photonEnergies_ConstProp,
             PropertyValues: labRindex,
             NumEntries: NUM_PP_ConstProp);
AddProperty (ptr: dopedLABGdMat,
             key: "ABSELENGTH",
             PhotonEnergies: labAbsorptionEnergies,
             PropertyValues: labAbsorption,
             NumEntries: num_labAbs);
// AddProperty(dopedLABGdMat.get(), "ABSELENGTH", templABAbsEnergies,
//           templABAbsLength, NUM_PP_ConstProp );//temp
AddProperty (ptr: dopedLABGdMat,
             key: "RAYLEIGH",
             PhotonEnergies: lab_Rayleigh_energy,
             PropertyValues: lab_Rayleigh_length,
             NumEntries: lab_Rayleigh_entries);
AddProperty (ptr: dopedLABGdMat,
             key: "SCINTILLATIONCOMPONENT1",
             PhotonEnergies: labEmissionEnergies,
             PropertyValues: labEmission,
             NumEntries: num_labEmi);
```

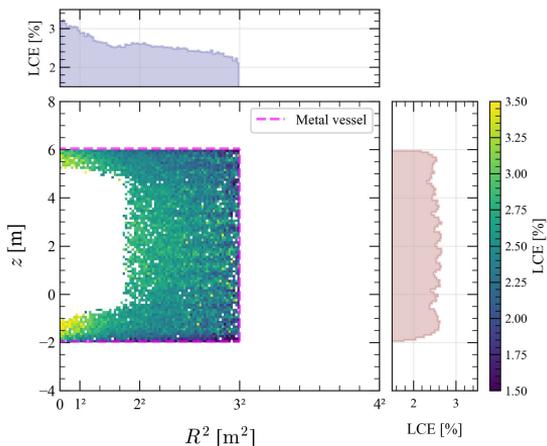


Example Studies



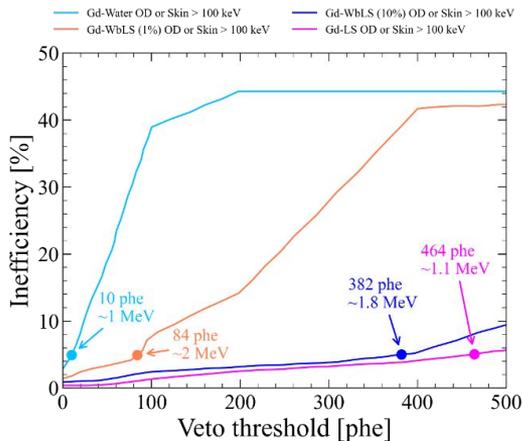
XLZD Sandbox has been used to complete a number of studies:

Light collection efficiency



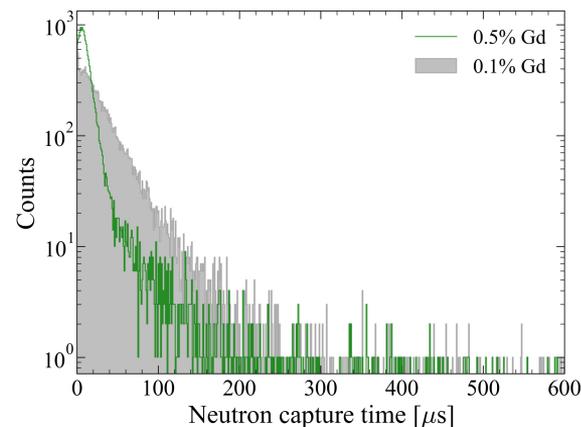
- Turn optical physics on in the macro
- Look at neutron capture position per event
- How many photons were incident on photocathodes?

Neutron veto efficiency



- Generate neutrons from detector components
- Look at signals in OD after a TPC single scatter
- Vary OD media in `detector.json` for efficiency comparison

Gd concentration studies



- Generate neutrons from detector components
- Vary Gd concentration in OD medium in `detector.json`
- Compare neutron capture times

**Thanks for
listening!**

