```
<volume name="TargetPositionOptics1_logical">
  <materialref ref="G4_Galactic"/>
  <solidref ref="TargetPosition_solid"/>
  <physvol>
    <volumeref ref="TargetCFoilUS_logical"/>
    <position z="-30*cm"/>
  </physvol>
  <physvol>
    <volumeref ref="TargetCFoilDS_logical"/>
    <position z="+60*cm"/>
  </physvol>
  <auxiliary auxtype="TargetSystem" auxvalue="Optics1"/>
</volume>
```

Optics1 US Optics1 DS

25 x 25 x 1 mm dimensions of the C foil

This is how it has been set up in the targetLadder.gdml file.

And then we declare the sampling volume.

```
/remoll/geometry/absolute_position targetLadder (0,560,0)
/remoll/target/mother Optics1
/remoll/target/volume DSC
/remoll/target/print
```

Does it mean that other C foil (that has not been declared as the sampling volume) is physically removed from its location or it just mean that the generator will produce an electron at the sampling C foil location but the other C foil is still there and contributing towards the energy loss?

Sampling Types

(defined in remollVEventGen.cc and remollBeamTarget.cc source files)

kNoTargetVolume

This one is called when we are using the beam generator

AllTargetVolumes / kActiveTargetVolume

For all the other generators.

For the beam generator the electron will just traverse whatever comes its way.

But for the physics generators we need to define the vertex location of where the primary electron will be produced from. That is done within the remollBeamTarget.cc file using the SampleVertex() function. And depending upon the sampling type a primary electron will be produced from one of the C foils (kActiveTargetVolume) or from both the foils (AllTargetVolumes) and the corresponding energy losses will also be taken into account.

```
// Get absolute position
fMotherTargetAbsolutePosition = fTargetMothers[fActiveTargetMother].first->GetTranslation().z() - 4500;

for (auto it =  fTargetVolumes[fActiveTargetMother].begin();
          it != fTargetVolumes[fActiveTargetMother].end();
          it++) {

    // Try to cast the target volume into its tubs solid
    G4VPhysicalVolume* physvol = (*it).first;
    G4LogicalVolume* volume = physvol->GetLogicalVolume();
    G4Material* material = volume->GetMaterial();
    G4VSolid* solid = volume->GetSolid();

    G4Tubs* tubs = dynamic_cast<G4Tubs*>(solid);
    G4Box* box = dynamic_cast<G4Box*>(solid);

    // Assume everything is non-nested tubes
    if ((tubs == nullptr) && (box == nullptr)) {
        G4cerr << "ERROR:  " << __PRETTY_FUNCTION__ << " line " << __LINE__ <<
            ":  Target volume " << volume->GetName() << " not made of G4Tubs or G4Box" << G4endl;
        exit(1);
    }

    G4double z_half_length = 0;
    if (tubs != nullptr) z_half_length = tubs->GetZHalfLength();
    if (box != nullptr)  z_half_length = box->GetZHalfLength();

    fTotalTargetEffectiveLength += 2.0 * z_half_length * material->GetDensity();

    if (it == fTargetVolumes[fActiveTargetMother].begin() + fActiveTargetVolume) {

        fActiveTargetEffectiveLength = 2.0 * z_half_length * material->GetDensity();
    }
}
```

To account for the relative offset of the mother and the global coordinate systems.

A piece of code from remollBeamTarget.cc file.

The quantity of interest. This defines the z extent to which the primary electron will be produced.

Once we have defined the active target effective length.

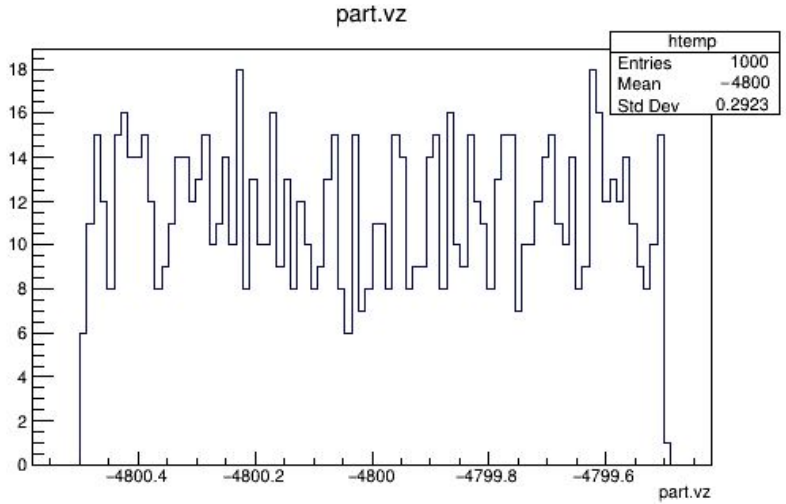
```
// Sample raster x and y positions on target
// (assumed independent of z position)
G4double rasx = G4RandFlat::shoot(fX0 - fRasterX/2.0, fX0 + fRasterX/2.0);
G4double rasy = G4RandFlat::shoot(fY0 - fRasterY/2.0, fY0 + fRasterY/2.0);

// Sample where along target weighted by density (which roughly corresponds to A
// or the number of electrons, which is probably good enough for this

// Update if needed
if (fUpdateNeeded) UpdateInfo();
```

```
// Figure out how far along the target we got
G4double total_effective_length = 0;
switch (sampling_type) {
    case kActiveTargetVolume:
        total_effective_length = fActiveTargetEffectiveLength;
        break;

    case kAllTargetVolumes:
        total_effective_length = fTotalTargetEffectiveLength;
        break;

    case kNoTargetVolume:
        // nothing to do, just avoid compilation warning
        break;
}
G4double effective_position = G4RandFlat::shoot(0.0, total_effective_length);
```

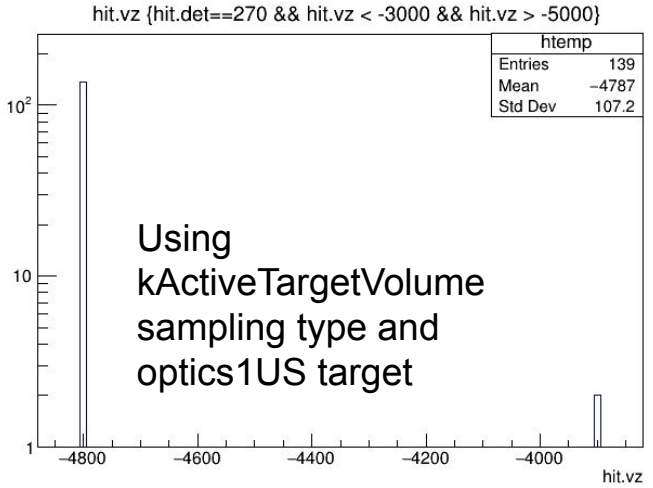
Depending upon the sampling type and the effective length a particle is created.

The event rates depend upon the particular generator being used. (differences in the form factors)

It also means that declaring a volume as the active target volume only defines the point of origin for the primary electron it does not mean that the other C foil (not the sampling volume) is physically removed from its place.

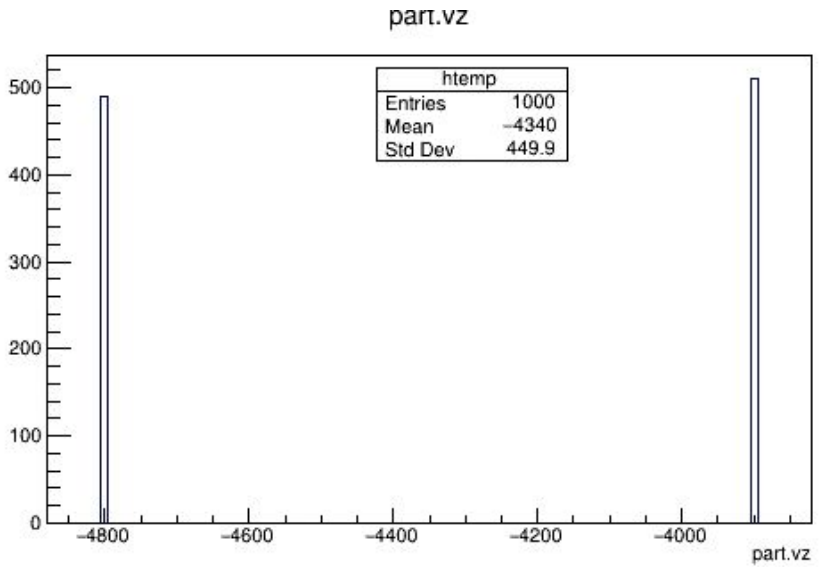


Using a C12 generator on Optics1US target. The vz distribution of primaries (as expected)



Vertex distribution for all particles on the sieve plane. As can be seen there is a peak at -3900 mm as well (optics1 DS)

Now by using the C12 generator but with the kAllTargetVolume sampling type (in remollVEventGen.cc source file).



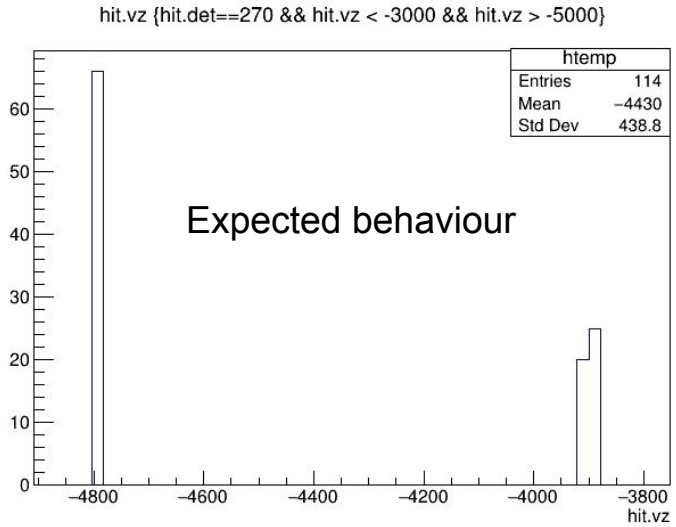
Again 1000 events were simulated but now with a different sampling and as expected now the primaries are equally likely to be produced at both C foil locations.

/remoll/target/volume → This becomes irrelevant now.

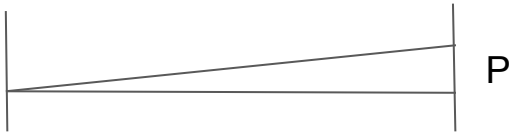
With the beam generator we just need to define the target ladder position

/remoll/target/mother
/remoll/target/volume

I don't think these are relevant with beam generator. The simulation ran without any errors without declaring these.



Vertex distribution of all particles hitting the sieve virtual plane. Two peaks at the two C foil locations.



For theta = 0.1 deg P = 0.157 mm
For theta = 2 deg P = 3.143 mm

Does it change anything for us?

It does change things for us. The way the source code has been written, even if I declare for example the DS C foil as the sampling volume the energy loss due to the US C foil will be taken into account.

```
switch (sampling_type) {
    case kActiveTargetVolume:
        if (it == fTargetVolumes[fActiveTargetMother].begin() + fActiveTargetVolume) {
            // This is the active volume, and we only sample here
            found_active_volume = true;
            actual_position_in_volume = effective_position/material->GetDensity();
            // but we still want cumulative radiation lengths of part of the volume
            cumulative_radiation_length += actual_position_in_volume/material->GetRadlen();
        } else {
            // but we still want cumulative radiation lengths of all of the volume
            cumulative_radiation_length += effective_length/material->GetDensity()/material->GetRadlen();
        }
        break;
```

Look at how the cumulative radiation length has been defined

```
Initialized tree.
--> Event 0 starts.
0.222742
remoll: Some detectors have been explicitly disabled in macros.
remoll: To disable/enable detectors, use the following syntax:
remoll:   /remoll/SD/print_all
remoll:   /remoll/SD/enable_all
remoll:   /remoll/SD/disable_all
remoll:   /remoll/SD/enable 5530
remoll:   /remoll/SD/disable 5530
--> Event 1 starts.
0.48290164877337
--> Event 2 starts.
0.18305572550082
--> Event 3 starts.
0.72874827674615
--> Event 4 starts.
0.074081374065093
--> Event 5 starts.
0.47372208602182
--> Event 6 starts.
0.86662274469281
--> Event 7 starts.
0.94524809063687
--> Event 8 starts.
0.045791445900345
--> Event 9 starts.
0.26647140832453
### Run 0 ended (3.55s).
Writing output to o_remoll_C12_all.root ... done
Graphics systems deleted.
Visualization Manager deleting...
 Running time[s]: 103.17
```

When I use the US C foil with elastic C12 generator and simulate 10 events, I print out the radiation lengths (in units of mm) that are taken into account while calculating the energy loss. Each C foil is 1 mm in z and as can be seen above in all the 10 events we are getting a number < 1. It is expected since now I am using the US C foil.

```
Initialized tree.
--> Event 0 starts.
1.45288
remoll: Some detectors have been explicitly disabled in macros.
remoll: To disable/enable detectors, use the following syntax:
remoll:   /remoll/SD/print_all
remoll:   /remoll/SD/enable_all
remoll:   /remoll/SD/disable_all
remoll:   /remoll/SD/enable 5530
remoll:   /remoll/SD/disable 5530
--> Event 1 starts.
1.3164391846328
--> Event 2 starts.
1.9115885652532
--> Event 3 starts.
1.2900543265793
--> Event 4 starts.
1.0298765239057
--> Event 5 starts.
1.8747303533486
--> Event 6 starts.
1.392107132653
--> Event 7 starts.
1.0751198384491
--> Event 8 starts.
1.0240288022529
--> Event 9 starts.
1.8617041546814
### Run 0 ended (4.16s).
Writing output to o_remoll_C12_all.root ... done
Graphics systems deleted.
Visualization Manager deleting...
 Running time[s]: 98.88
```

When I use the DS C foil with elastic C12 generator and simulate 10 events, I print out the radiation lengths (in units of mm) that are taken into account while calculating the energy loss. Each C foil is 1 mm in z and as can be seen above in all the 10 events we are getting a number > 1.