

## Checking material with tools in and DD4hep

There are several tools (material scans, material checks) in DD4hep, Geant4 and ROOT too check the material. In the context of tracking surfaces and the material averaged and associated to them, the AidaTT package includes additional tools in its examples. You may need to build the examples by doing (in a working, initialized ILCSoft release):

```
cd $AIDATT;
cd build;
make examples;
cd examples
ls
```

Of particular interest is `check_materials` which prints all the information of encountered surfaces along a  $(\theta, \phi)$  track. It can be used to compare against the output from `materialScan` that is available in `$PATH` when DD4hep is initialized in your ILCSoft release.

## Checking orientation of surfaces

First try to understand the output of `check_materials`:

```
$AIDATT/build/examples/check_materials $lcgeo_DIR/CLIC/compact/CLIC_o2_v04/CLIC_o2_v04.xml 30 0
```

The command above will have a long printout including the geometry initialization, the intersection with surfaces, the type of surface, the u,v,n vectors, etc.

Should you want to filter out most of the information and keep say the u,v,n vectors for the endcaps, use `grep` and `awk` for example like so:

```
$AIDATT/build/examples/check_materials $lcgeo_DIR/CLIC/compact/CLIC_o2_v04/CLIC_o2_v04.xml 30 0 2
```

When firing at an angle  $\phi = 0$  ( $zx$  plane) and  $\theta = 30$  in the positive  $z$  direction, the  $u$   $v$   $n$  vectors in the endcap should be approximately:

```
u(1,0,0) <-> r (x for this angle)
v(0,1,0) <-> rPhi (y for this angle)
n(0,0,1) <-> z
```

 The convention above yields a right-handed coordinate system in the positive  $z$  direction in the endcap, with  $u$  along the "bad" measurement direction (with the worse resolution typically for strip detectors in the  $r$ -direction). We need to make sure that it makes sense and how it looks in the negative  $z$ -direction. The barrel is somehow more straightforward:  $n$  points outwards,  $v$  along  $z$  and  $u$  is  $r\Phi$ .

## Comparing material from averaging to material scan

 You may find that it's easier to disable parts of the detector model when performing such cross-checks. Just comment-out every subdetector include from the compact xml and gradually add them when performing the checks.

Use first the `check_materials` command which will give you the intersections with all the surfaces (by convention, set at the middle of the sensitive or support volume) along a given straight line from the IP. For example, let's perform a scan along 40 degrees:

```
$AIDATTbuild/examples/check_materials $lcgeo_DIR/CLIC/compact/CLIC_o2_v04/CLIC_o2_v04.xml 40 0
```

after a lot of printout, the last few lines should be:

```
found intersection at s = 138.577, ( 138.577, -9.60183e-09, 165.15 ) - [ phi: -6.92887e-11 ,
  id: 0 type : sensitive[0] helper[1] plane[1] cylinder[0] cone[0] parallelToZ[0] orthogonalToZ[0]
  u : ( 1, 0, 0 ) - [ phi: 0 , rho: 1 ] [ theta: 1.5708 , r: 1 ] v : ( 0, 1, 0 ) - [ phi: 0 , rho: 1 ]
  inner material : CarbonFiber, A: 11.9678, Z: 6.00233, density: 1.5, radiationLength: 28.2108
  outerMaterial : CarbonFiber, A: 11.9678, Z: 6.00233, density: 1.5, radiationLength: 28.2108
X0 : 0.013882 - total : 0.1641
```

Some things to notice is the intersection point with the last surface ( **138.577, -9.60183e-09, 165.15** ) and the total number of interaction lengths (**total : 0.1641**). This particular surface is a helper surface at the center of a support structure to provide material properties to the tracking algorithms. We will now check against the TGeo material scan (**materialScan**) which takes as arguments a compact xml file and the components of a starting and end point. Naturally, the starting point should be the IP (0,0,0). The end point is **not** the intersection point from above since it's a point on the surface installed in the middle of the support volume. Instead the scan needs to include the material **after the surface** till the end of the module, as defined by the **outer thickness** associated with the surface. In this case we notice **OuterMaterial: . . . . . thickness: 0.15**, i.e. 0.15 cm. We correct the intersection point by adding 0.15 to its z component and  $\tan(40) * 0.15$  to its x component. The full material scan command should therefore be:

```
materialScan CLIC/compact/CLIC_o2_v04/CLIC_o2_v04.xml 0 0 0 138.7 0 165.30
```

We immediately see at the last line of the printout:

```
+-----+
+ Material scan between: x_0 = ( 0.00, 0.00, 0.00) [cm] and x_1 = ( 138.70, 0.00, 165.30) [cm]
+-----+
| \ Material          Atomic          Radiation   Interaction   Path
| Num. \ Name          Number/Z      Mass/A      Density      Length      Length      Thickness   Length
| Layer \              [g/mole]    [g/cm3]     [cm]         [cm]         [cm]         [cm]         [cm]
+-----+
| 1 beam              5      9.370      0.0000      2.80097e+15  2.07517e+15  4.574       4.57
| ...
+-----+
| 0 Average Material  7      13.391     0.0289      1262.5909   2733.2137   215.782    215.78
+-----+
```

that the total number of radiation lengths is 0.170904 which differs from the value obtained from the surfaces, 0.1641. It seems like this discrepancy can be attributed to the Air between the modules: at the time of writing, our models do not install auxilliary surfaces to describe the air between the modules, nor do they adjust the thicknesses of the surfaces to include the air in their material averaging. It is therefore understood that the unaccounted air will contribute to a discrepancy of approximately 1%X0 for a full path length of 3 m. One way to verify the contribution of air is to use `grep` and `awk` to filter out the air thicknesses and calculate their sum in terms of X0:

```
materialScan CLIC/compact/CLIC_o2_v04/CLIC_o2_v04.xml 0 0 0 138.7 0 165.30 |grep Air|awk 'BEGIN{sum=0;}
```

which yields:

```
Air with thickness: 0.156
Air with thickness: 1.696
Air with thickness: 1.851
Air with thickness: 7.888
Air with thickness: 0.031
Air with thickness: 17.735
Air with thickness: 1.349
Air with thickness: 46.244
Air with thickness: 5.309
Air with thickness: 1.072
Air with thickness: 38.894
```

## DD4hepSurfaces < CLIC < TWiki

Air with thickness: 0.982  
Air with thickness: 43.598  
Air with thickness: 1.387  
Air with thickness: 0.196  
Air with thickness: 0.196  
Air with thickness: 37.285  
Air with thickness: 0.865  
Air with thickness: 0.196  
Number of X0 due to air: 0.00680337

We therefore see that the air is contributing to approximately 0.7%X0. The material scan, corrected for the air is now 0.170904-0.00680337 which yields 0.16410063 in perfect agreement with the value from check\_materials.

-- NikiforosNikiforou - 2015-12-04

---

This topic: CLIC > DD4hepSurfaces

Topic revision: r3 - 2015-12-10 - NikiforosNikiforou



Copyright &© 2008-2022 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.  
or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback