A Quick Guide on XML Geometries and Voxels

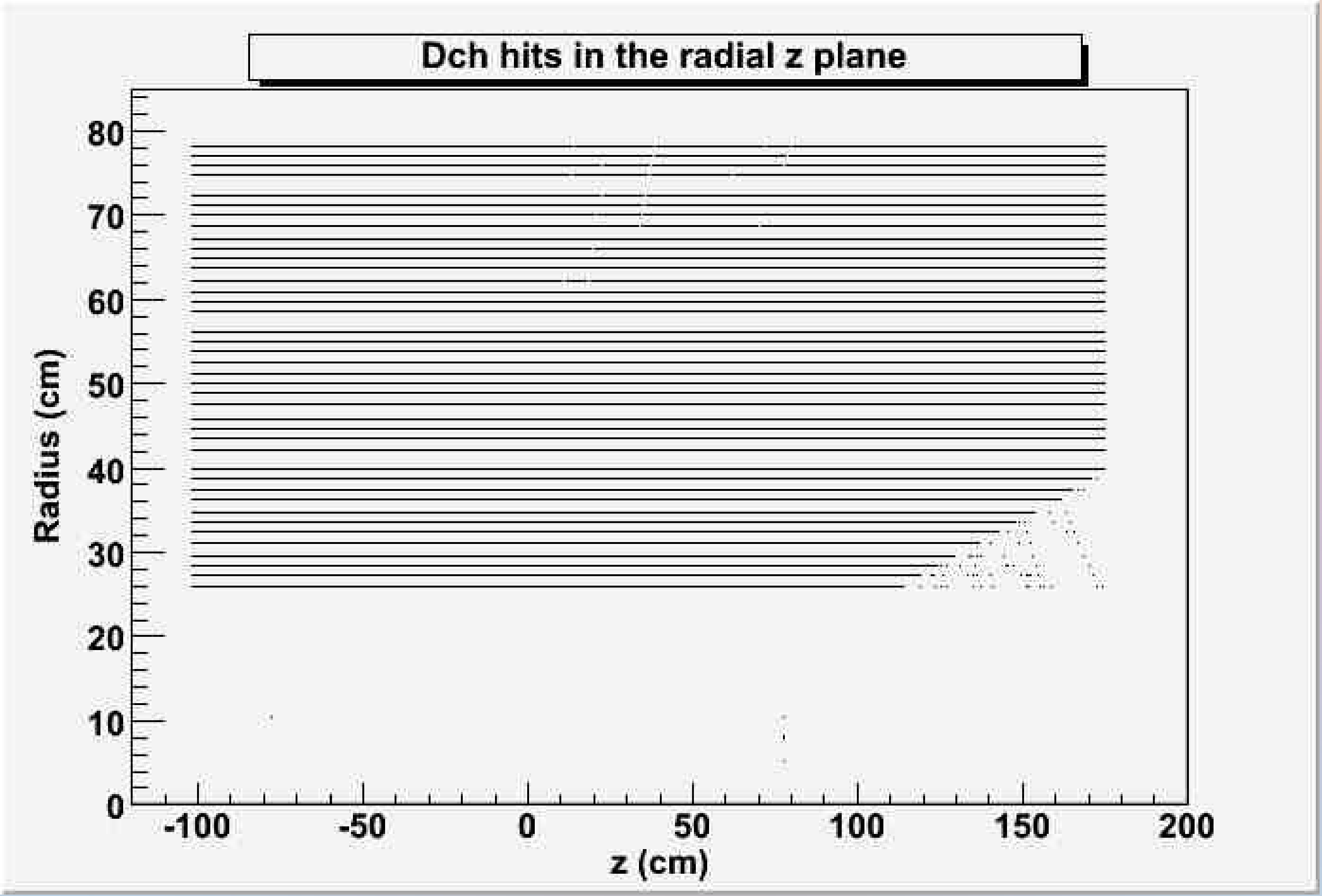
September 28, 2010

McGill

Darren Swersky,
McGill University

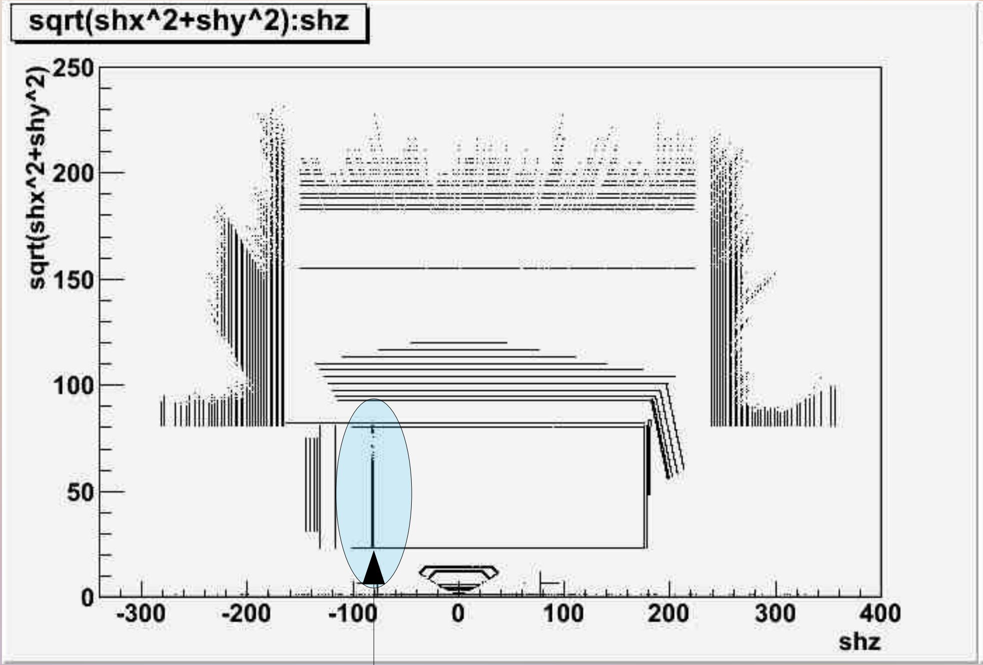
Bugs in previous versions:

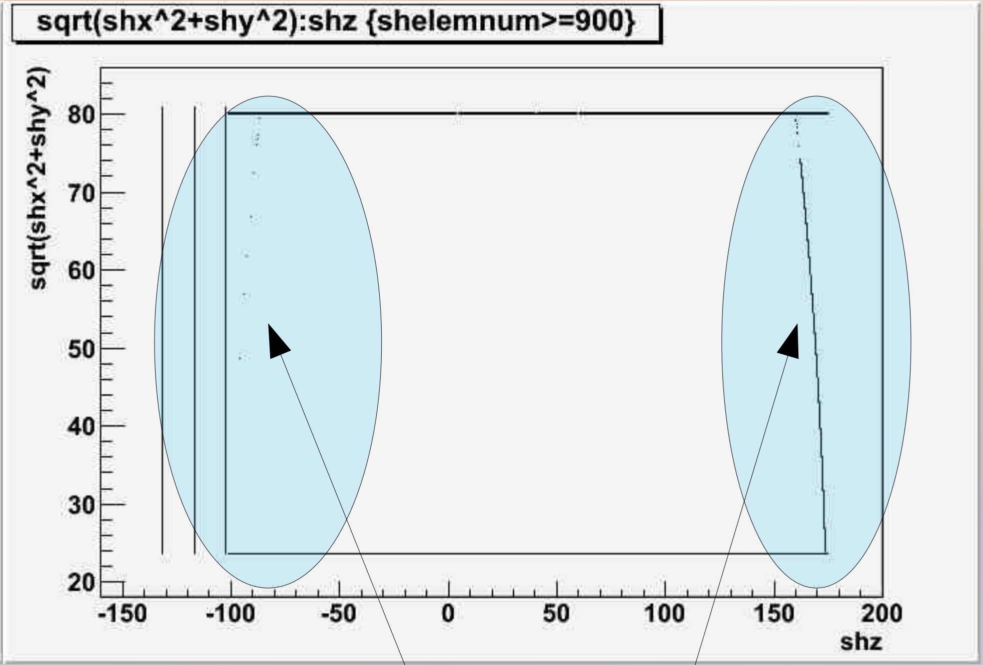
V0.2.0: The “Corners Bug”



Bugs in previous versions:

V0.2.4: Hit detection problems



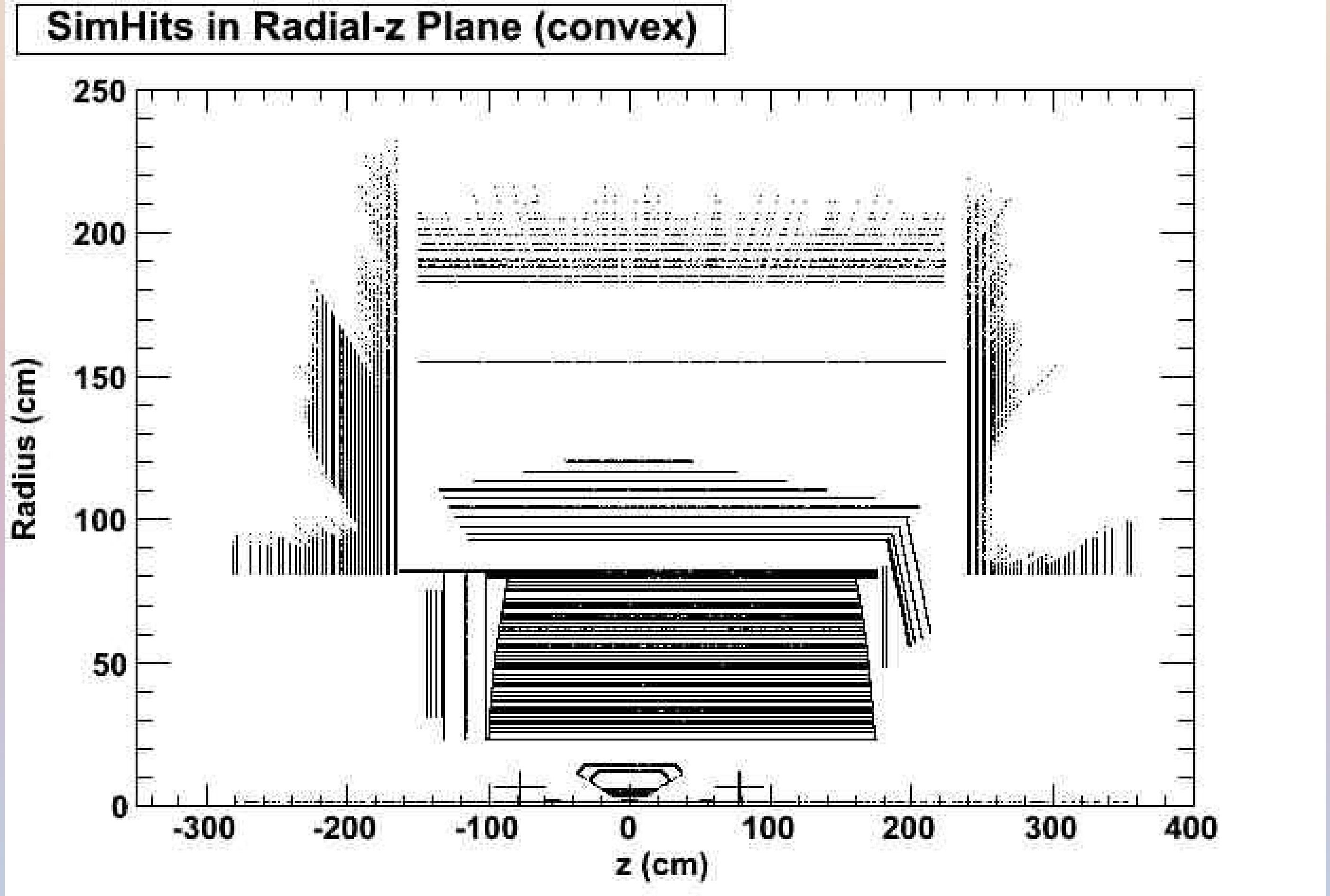


Test with no wires

Missing hits on endplates (this example had wires, not drawn)

With new voxel fix:

It works great!



Voxel example: Baseline geometry

```
<cyl name="DchInnerCyl" id="901" zmin="wirezmin" zmax="wirezmax" radius="23.60" thick="0.1000" mat="dch-CFiber" />
<cyl name="dch-Wires" id="801" zmin="wirezmin" zmax="wirezmax" radius="26.05" thick="0.0120" mat="dch-Wires" gap="0.985" />
<cyl name="dch-He-Ibu" id="701" zmin="wirezmin" zmax="wirezmax" radius="26.06" thick="1.4300" mat="dch-He-Ibu" meas="Axial" />
<cyl name="dch-Wires" id="802" zmin="wirezmin" zmax="wirezmax" radius="27.24" thick="0.0120" mat="dch-Wires" gap="0.986" />
<cyl name="dch-He-Ibu" id="702" zmin="wirezmin" zmax="wirezmax" radius="27.25" thick="1.4300" mat="dch-He-Ibu" meas="Axial" />
<cyl name="dch-Wires" id="803" zmin="wirezmin" zmax="wirezmax" radius="28.43" thick="0.0120" mat="dch-Wires" gap="0.987" />
<cyl name="dch-He-Ibu" id="703" zmin="wirezmin" zmax="wirezmax" radius="28.44" thick="1.4300" mat="dch-He-Ibu" meas="Axial" />
<cyl name="dch-Wires" id="804" zmin="wirezmin" zmax="wirezmax" radius="29.62" thick="0.0120" mat="dch-Wires" gap="0.987" />
<cyl name="dch-He-Ibu" id="704" zmin="wirezmin" zmax="wirezmax" radius="29.63" thick="1.4300" mat="dch-He-Ibu" meas="Axial" />
<cyl name="dch-Wires" id="805" zmin="wirezmin" zmax="wirezmax" radius="31.25" thick="0.0120" mat="dch-Wires" gap="0.986" />
<cyl name="dch-He-Ibu" id="705" zmin="wirezmin" zmax="wirezmax" radius="31.26" thick="1.4300" mat="dch-He-Ibu" meas="Stereo+"/>
<cyl name="dch-Wires" id="806" zmin="wirezmin" zmax="wirezmax" radius="32.44" thick="0.0120" mat="dch-Wires" gap="0.986" />
<cyl name="dch-He-Ibu" id="706" zmin="wirezmin" zmax="wirezmax" radius="32.45" thick="1.4300" mat="dch-He-Ibu" meas="Stereo+"/>
<cyl name="dch-Wires" id="807" zmin="wirezmin" zmax="wirezmax" radius="33.63" thick="0.0120" mat="dch-Wires" gap="0.987" />
<cyl name="dch-He-Ibu" id="707" zmin="wirezmin" zmax="wirezmax" radius="33.64" thick="1.4300" mat="dch-He-Ibu" meas="Stereo+"/>
```

etc.

```
    <config>
      <sect name="Drift_Tracking">
        <param name="rbounds"    type="vector"       >
          23 25 27 28 29 31 32 33 34 36
          37 38 39 42 43 44 45 47 48 50
                           51 52 53 54 56 58 59 60 62 63
                           64 66 67 68 69 71 72 74 75 77
                           78 79 80.5 81 </param>
        <!-- no phi segmentation needed, as all elements are perfectly
cylindrical -->
        <param name="nphivoxels"    type="vector"        >
          1 1 1 1 1 1 1 1 1 1
          1 1 1 1 1 1 1 1 1 1
          1 1 1 1 1 1 1 1 1 1
          1 1 1 1 1 1 1 1 1 1
          1 1 1 </param>
        <param name="zbounds"   type="vector"        > -132 -117
-103 -102 -101 174 176 178 </param>
      </sect>
    </config>
```

Space radial voxels so there are ~2 elements or less per voxel (mostly used for cylinders)

For n radial voxels, this has n-1 entries

Voxel example: Baseline geometry

```
 <ring name="DchEndplate" id="903" z="wirezmin" lowradius="23.6" hiradius="80.9" thick="0.80" mat="dch-CFiber" />
<ring name="DchEndplate" id="904" z="wirezmax" lowradius="23.6" hiradius="80.9" thick="0.80" mat="dch-CFiber" />
<ring name="DchEndplate" id="602" z="177.6" lowradius="23.6" hiradius="80.9" thick="0.10" mat="dch-Aluminum" />
<ring name="DchElectronics" id="603" z="-102.5" lowradius="23.6" hiradius="80.9" thick="0.10" mat="dch-Aluminum" />
<ring name="DchElectronics" id="604" z="-116.5" lowradius="23.6" hiradius="80.9" thick="0.30" mat="dch-Aluminum" />
<ring name="DchElectronics" id="605" z="-131.5" lowradius="23.6" hiradius="80.9" thick="0.40" mat="dch-Aluminum" />
```

etc.

```
        <config>
           <sect name="Drift_Tracking">
              <param name="rbounds"    type="vector"        >
                 23 25 27 28 29 31 32 33 34 36
                 37 38 39 42 43 44 45 47 48 50
                                    51 52 53 54 56 58 59 60 62 63
                                    64 66 67 68 69 71 72 74 75 77
                                    78 79 80.5 81 </param>
              <!-- no phi segmentation needed, as all elements are perfectly
  cylindrical -->
              <param name="nphivoxels"    type="vector"         >
                 1 1 1 1 1 1 1 1 1 1
                 1 1 1 1 1 1 1 1 1 1
                 1 1 1 1 1 1 1 1 1 1
                 1 1 1 1 1 1 1 1 1 1
                 1 1 1 </param>
              <param name="zbounds"   type="vector"        > -132 -117
  -103 -102 -101 174 176 178 </param>
           </sect>
        </config>
```

Space z-voxels so there are ~2 elements or less per voxel (mostly used for rings), avoid empty voxels (waste of computing time)

Loop example: Convex geometry

```
<define>
        <variable name="loopvar" type="int"></variable>
        <formula name="rightplate_conebottom_z" par0="loopvar" formula="220.8333*cos(0.107072+[0]*0.0131812)-45.9333"></formula>
        <formula name="rightplate_conetop_z" par0="loopvar" formula="220.8333*cos(0.107072+([0]+1)*0.0131812)-45.9333"></formula>
        <formula name="leftplate_conebottom_z" par0="loopvar" formula="-220.8333*cos(0.107072+[0]*0.0131812)+119.3333"></formula>
        <formula name="leftplate_conetop_z" par0="loopvar" formula="-220.8333*cos(0.107072+([0]+1)*0.0131812)+119.3333"></formula>
        <formula name="bothplates_conebottom_r" par0="loopvar" formula="220.8333*sin(0.107072+[0]*0.0131812)"></formula>
        <formula name="bothplates_conetop_r" par0="loopvar" formula="220.8333*sin(0.107072+([0]+1)*0.0131812)"></formula>
        <formula name="rightplate_id" par0="loopvar" formula="903+[0]"></formula>
        <formula name="leftplate_id" par0="loopvar" formula="923+[0]"></formula>
</define>
```

```
<loop variable="loopvar" count="20" startvalue="0" stepvalue="1">

<cone name="DchEndCone" id="rightplate_id" rho1="bothplates_conebottom_r"z1="rightplate_conebottom_z"rho2="bothplates_conetop_r"
z2="rightplate_conetop_z" thick="0.50" mat="dch-CFiber" />

</loop>

<loop variable="loopvar" count="20" startvalue="0" stepvalue="1">

<cone name="DchEndCone" id="leftplate_id" rho1="bothplates_conebottom_r" z1="leftplate_conebottom_z" rho2="bothplates_conetop_r"
z2="leftplate_conetop_z" thick="0.50" mat="dch-CFiber" />

</loop>
```

Result shown in slide 4

Loop example: IP beam shield

```
<define>
        <variable name="loopvar1" type="int"></variable>
        <variable name="loopvar2" type="int"></variable>

        <!-- Define inner cones of fwd beamstopper (coneA) -->

        <formula name="fwd_beamstopperconeA_z1" par0="loopvar1" formula="5.0+[0]*1.0"></formula>
        <formula name="fwd_beamstopperconeA_z2" par0="loopvar1" formula="5.0+(1+[0])*1.0"></formula>
        <formula name="fwd_beamstopperconeA_r1" par0="loopvar1" par1="loopvar2"
                    formula="(1.1+0.03571*[0])*(8-[1])*0.125+(1.2+0.09286*[0])*[1]*0.125"></formula>
        <formula name="fwd_beamstopperconeA_r2" par0="loopvar1" par1="loopvar2"
                    formula="(1.1+0.03571*(1+[0]))*(8-[1])*0.125+(1.2+0.09286*(1+[0]))*[1]*0.125"></formula>
        <formula name="fwd_beamstopperconeA_thick" par0="loopvar1"
                    formula="((1.2+0.09286*(0.5+[0]))-(1.1+0.03571*(0.5+[0])))*0.125"></formula>
        <formula name="fwd_beamstopperconeA_id" par0="loopvar1" par1="loopvar2" formula="1001+[0]*9+[1]"></formula>
        <formula name="fwd_beamstopperringA_r1" par0="loopvar1" formula="1.1+0.03571*[0]"></formula>
        <formula name="fwd_beamstopperringA_r2" par0="loopvar1" formula="1.2+0.09286*[0]"></formula>
        <formula name="fwd_beamstopperringA_id" par0="loopvar1"formula="1127+[0]"></formula>

        <!-- Define middle cones of fwd beamstopper (coneB) -->

        <formula name="fwd_beamstopperconeB_z1" par0="loopvar1" formula="19.0+[0]*1.0"></formula>
        <formula name="fwd_beamstopperconeB_z2" par0="loopvar1" formula="19.0+(1+[0])*1.0"></formula>
        <formula name="fwd_beamstopperconeB_r1" par0="loopvar1" par1="loopvar2"
                    formula="(1.6+0.02449*[0])*(8-[1])*0.125+(2.5+0.44898*[0])*[1]*0.125"></formula>
        <formula name="fwd_beamstopperconeB_r2" par0="loopvar1" par1="loopvar2"
                    formula="(1.6+0.02449*(1+[0]))*(8-[1])*0.125+(2.5+0.44898*(1+[0]))*[1]*0.125"></formula>
        <formula name="fwd_beamstopperconeB_thick" par0="loopvar1"
                    formula="((2.5+0.44898*(0.5+[0]))-(1.6+0.02449*(0.5+[0])))*0.125"></formula>
        <formula name="fwd_beamstopperconeB_id" par0="loopvar1"par1="loopvar2" formula="1201+[0]*9+[1]"></formula>
        <formula name="fwd_beamstopperconeB_tip_r1" par0="loopvar2" formula="2.18776*(8-[0])*0.125+13.27552*[0]*0.125"></formula>
        <formula name="fwd_beamstopperconeB_tip_r2" par0="loopvar2"         formula="2.2*(8-[0])*0.125+13.5*[0]*0.125"></formula>
        <formula name="fwd_beamstopperconeB_tip_id" par0="loopvar2" formula="1417+[0]"></formula>
        <formula name="fwd_beamstopperringB_r1" par0="loopvar1" formula="1.6+0.02449*[0]"></formula>
        <formula name="fwd_beamstopperringB_r2" par0="loopvar1" formula="2.5+0.44898*[0]"></formula>
        <formula name="fwd_beamstopperringB_id" par0="loopvar1" formula="1426+[0]"></formula>
```

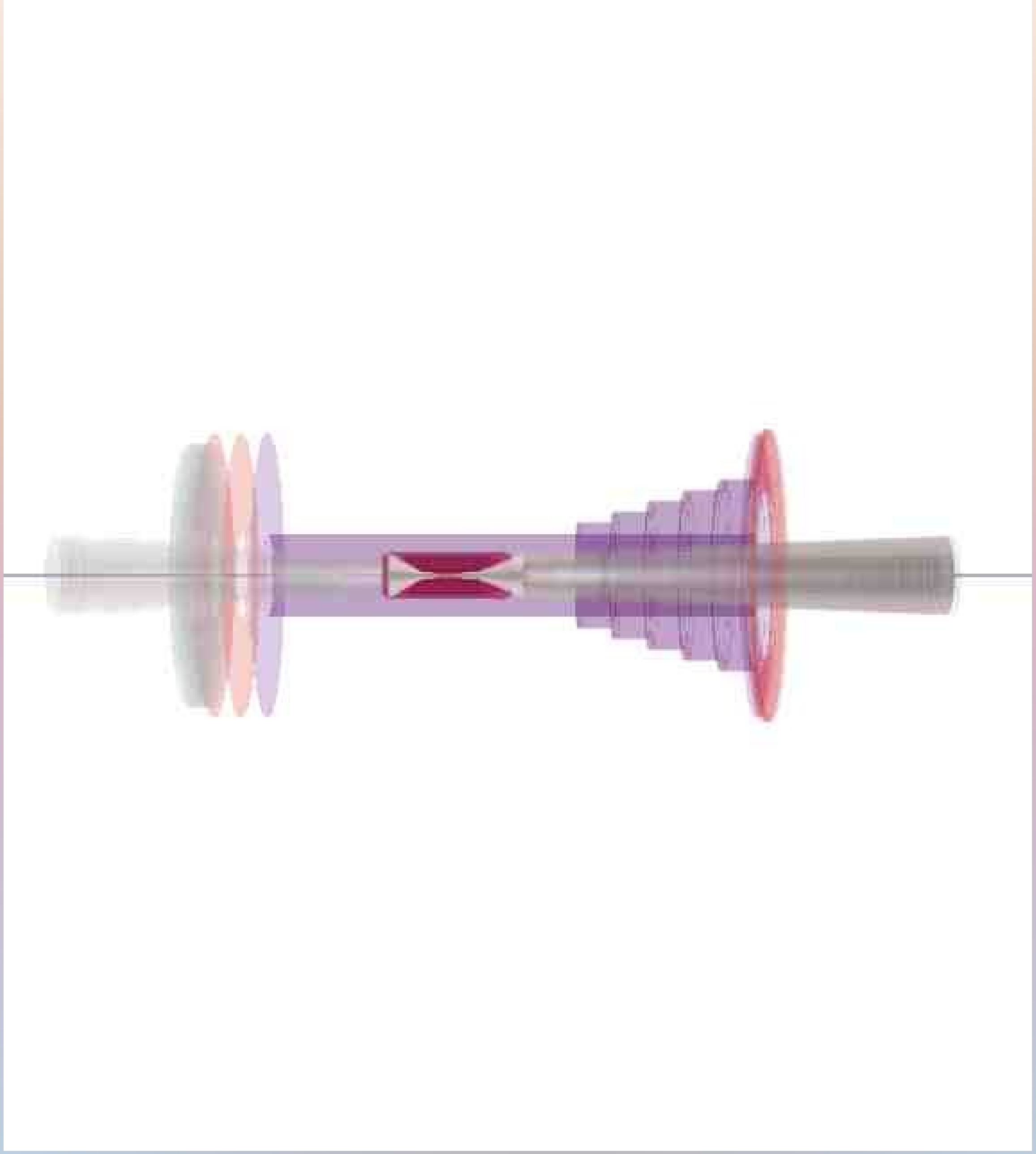
etc.

Loop example: IP beam shield

```
        <!-- fwd beamstopper (coneA) constructed here -->

          <loop variable="loopvar1" count="14" startvalue="0" stepvalue="1">
                  <ring name="Fwd_Beamstopper_RingA" id="fwd_beamstopperringA_id" z="fwd_beamstopperconeA_z1"
                  lowradius="fwd_beamstopperringA_r1" hiradius="fwd_beamstopperringA_r2" thick="1.0" mat="pep-Tungsten" />

                  <loop variable="loopvar2" count="4" startvalue="0" stepvalue="1">
                  <cone name="Fwd_Beamstopper_ConeA" id="fwd_beamstopperconeA_id" rho1="fwd_beamstopperconeA_r1"
                  z1="fwd_beamstopperconeA_z1" rho2="fwd_beamstopperconeA_r2" z2="fwd_beamstopperconeA_z2"
                  thick="fwd_beamstopperconeA_thick" mat="pep-Tungsten" />
                  </loop>
<loop variable="loopvar2" count="4" startvalue="5" stepvalue="1">
<cone name="Fwd_Beamstopper_ConeA" id="fwd_beamstopperconeA_id" rho1="fwd_beamstopperconeA_r1"
z1="fwd_beamstopperconeA_z1" rho2="fwd_beamstopperconeA_r2" z2="fwd_beamstopperconeA_z2"
thick="fwd_beamstopperconeA_thick" mat="pep-Tungsten" />
</loop>

          </loop>

        <!-- fwd beamstopper (coneB) contructed here -->

          <loop variable="loopvar1" count="24" startvalue="0" stepvalue="1">
                  <ring name="Fwd_Beamstopper_RingB" id="fwd_beamstopperringB_id" z="fwd_beamstopperconeB_z1"
                  lowradius="fwd_beamstopperringB_r1" hiradius="fwd_beamstopperringB_r2" thick="1.0" mat="pep-Tungsten" />


                  <loop variable="loopvar2" count="4" startvalue="0" stepvalue="1">
                            <cone name="Fwd_Beamstopper_ConeB" id="fwd_beamstopperconeB_id" rho1="fwd_beamstopperconeB_r1"
                            z1="fwd_beamstopperconeB_z1" rho2="fwd_beamstopperconeB_r2" z2="fwd_beamstopperconeB_z2"
                            thick="fwd_beamstopperconeB_thick" mat="pep-Tungsten" />
                  </loop>
<loop variable="loopvar2" count="4" startvalue="5" stepvalue="1">
        <cone name="Fwd_Beamstopper_ConeB" id="fwd_beamstopperconeB_id" rho1="fwd_beamstopperconeB_r1"
        z1="fwd_beamstopperconeB_z1" rho2="fwd_beamstopperconeB_r2" z2="fwd_beamstopperconeB_z2"
        thick="fwd_beamstopperconeB_thick" mat="pep-Tungsten" />
</loop>

          </loop>
```

etc.

Loop example: IP beam shield- end result



Loop example: IP beam shield- end result