Hands on: modifying the detector geometry

Lorenzo Sestini
INFN-Padova

On behalf of the Muon Collider Detector and Physics group

MuCol: training on the detector design and physics performance tool - CERN- 6/7/2023

Modify the detector geometry

- In this hands-on you will learn **how to modify the detector geometry**. This is the basic knowledge to implement a new detector (instructions at https://mcdwiki.docs.cern.ch/tutorial/advanced/geometry/)

- Remember: the simulation is performed with ***ddsim***, that is based on DD4HEP

- The **main XML file** that described the geometry is located in `$MUCOLL_GEO`

```
[lsestini@lxplus905 mucoll-tutorial-2023]$ ls $MUCOLL_GEO
/cvmfs/muoncollider.cern.ch/release/2.8-patch2/linux-almalinux9-x86_64/gcc-11.3.1/lcgeo-0.18.1-qp36lz4ohtjqltkvtvkirsyjkshsvg57/share/lcgeo/compact/MuColl/MuColl_v1/MuColl_v1.xml
[lsestini@lxplus905 mucoll-tutorial-2023]$ ls $(dirname $MUCOLL_GEO)
BeamCal_o1_v01_01.xml             ECalPlug_o1_v01_02.xml           InnerTracker_o2_v06_01.xml  Nozzle_10deg_v0.xml              Solenoid_o1_v01_01.xml    YokeEndcap_o1_v01_01.xml
BeamInstrumentation_o1_v01_01.xml HCalBarrel_o1_v01_01.xml         LumiCal_o1_v01_01.xml       OuterTrackerBarrelModuleDown.xml TrackerDiskModuleIn.xml   elements.xml
Beampipe_o1_v01_02.xml            HCalEndcap_o1_v01_01.xml         MuColl_v1.xml               OuterTrackerBarrelModuleUp.xml   TrackerDiskModuleOut.xml  materials.xml
ECalBarrel_o2_v01_02.xml          InnerTrackerBarrelModuleDown.xml MuColl_v1_mod.xml           OuterTracker_o2_v06_01.xml       Vertex_o2_v06_01.xml      text_description.md
ECalEndcap_o2_v01_02.xml          InnerTrackerBarrelModuleUp.xml   MuColl_v1_vis.xml           README.md                        YokeBarrel_o1_v01_01.xml
```

- `MuColl_v1.xml` contains a list of includes that define specific subdetectors. It contains also a list of constants that are used through the geometry files

Detector description

`MuColl_v1.xml`

```
<include ref="Beampipe_o1_v01_02.xml"/>
<include ref="Nozzle_10deg_v0.xml"/>
<!-- <include ref="BeamInstrumentation_o1_v01_01.xml"/> -->

<include ref="Vertex_o2_v06_01.xml"/>

<include ref="InnerTracker_o2_v06_01.xml"/>
<include ref="OuterTracker_o2_v06_01.xml"/>

<include ref="ECalBarrel_o2_v01_02.xml"/>
<include ref="ECalEndcap_o2_v01_02.xml"/>

<include ref="HCalBarrel_o1_v01_01.xml"/>
<include ref="HCalEndcap_o1_v01_01.xml"/>

<include ref="Solenoid_o1_v01_01.xml"/>

<include ref="YokeBarrel_o1_v01_01.xml"/>
<include ref="YokeEndcap_o1_v01_01.xml"/>

<!-- <include ref="LumiCal_o1_v01_01.xml"/> -->

<!-- <include ref="BeamCal_o1_v01_01.xml"/> -->
```

You can comment some include to exclude a detector and speed up the simulation

`Vertex_o2_v06_01.xml`

```
<constant name="VertexEndcap_offset" value="3*mm"/>
<constant name="VertexEndcap_rmax"   value="Vertex_outer_radius - VertexEndcap_offset"/>
<constant name="VertexEndcap_zmin"   value="70*mm"/>
<constant name="VertexEndcap_zmax"   value="310*mm"/>
<constant name="VertexEndcap_z1"     value="80*mm"/>
<constant name="VertexEndcap_z2"     value="120*mm"/>
<constant name="VertexEndcap_z3"     value="200*mm"/>
<constant name="VertexEndcap_z4"     value="280*mm"/>
<constant name="VertexEndcapModules" value="16"/>
<constant name="VertexEndcap_rmin1"  value="22*mm + VertexEndcap_offset"/>
<constant name="VertexEndcap_rmin2"  value="28*mm + VertexEndcap_offset"/>
<constant name="VertexEndcap_rmin3"  value="35*mm + VertexEndcap_offset"/>
<constant name="VertexEndcap_rmin4"  value="50*mm + VertexEndcap_offset"/>
<constant name="VertexEndcapOverlap" value="1*mm"/>
```

As an example

`<constant name="VertexEndcap_z2" value="120*mm"/>`

defines the position of the 2nd Vertex Endcap disk

Pay attention: do not introduce overlaps!!

Hands-on: moving a vertex detector sensor

1. Prepare a local directory and files with the geometry:

```
cp -r $(dirname $MUCOLL_GEO) MuColl_v1_mod1

mv ./MuColl_v1_mod1/MuColl_v1.xml ./MuColl_v1_mod1/MuColl_v1_mod1.xml
```

2. Modify the position of the 2nd endcap disk in `./MuColl_v1_mod1/Vertex_o2_v06_01.xml`

`<constant name="VertexEndcap_z2" value="150*mm"/>` from 120 mm to 150 mm

3. Run the simulation with the new geometry:

```
ddsim --steeringFile mucoll-benchmarks/simulation/ilcsoft/steer_baseline.py \
--compactFile=./MuColl_v1_mod1/MuColl_v1_mod1.xml --outputFile=./mumu_H_bb_mod1.slcio
```

4. Visualize the simulated events with

```
ced2go -d ./MuColl_v1_mod1/MuColl_v1_mod1.xml ./mumu_H_bb_mod1.slcio
```

Changing the calorimeter technology

`materials.xml`

```
<material name="beam" state="gas">
    <D unit="g/cm3" value="1.7e-14"/>
    <fraction n="0.363" ref="H"/>
    <fraction n="0.363" ref="N"/>
    <fraction n="0.117" ref="C"/>
    <fraction n="0.157" ref="O"/>
</material>

<material formula="Fe" name="Iron" state="solid" >
    <D type="density" unit="g/cm3" value="7.874" />
    <composite n="1" ref="Fe" />
</material>
```

It contains the definition of materials

`ECalBarrel_o2_v01_02.xml`

```
<layer repeat="40" vis="ECalLayerVis">
    <slice material = "TungstenDens24" thickness = "1.90*mm" vis="ECalAbsorberVis" radiator="yes"/>
    <slice material = "G10"            thickness = "0.15*mm" vis="InvisibleNoDaughters"/>
    <slice material = "GroundOrHVMix"  thickness = "0.10*mm" vis="ECalAbsorberVis"/>
    <slice material = "Silicon"        thickness = "0.50*mm" sensitive="yes" limits="cal_limits" vis="ECalSensitiveVis"/>
    <slice material = "Air"            thickness = "0.10*mm" vis="InvisibleNoDaughters"/>
    <slice material = "siPCBMix"       thickness = "1.30*mm" vis="ECalAbsorberVis"/>
    <slice material = "Air"            thickness = "0.25*mm" vis="InvisibleNoDaughters"/>
    <slice material = "G10"            thickness = "0.75*mm" vis="InvisibleNoDaughters"/>
</layer>
```

Definition of the ECAL barrel layers, in this case repeated 40 times

`sensitive="yes" limits="cal_limits"` define the sensitive layer

Hands-on: changing the calorimeter technology

1. Add a material (PbF_2) in `./MuColl_v1_mod1/materials.xml`

```
<material name="LeadDiflourite">
        <D type="density" value="7.77" unit="g/cm3"/>
        <composite n="1" ref="PB"/>
        <composite n="2" ref="F"/>
    </material>
```

2. Change the layer definition in `./MuColl_v1_mod1/ECalBarrel_o2_v01_02.xml` with

```
<layer repeat="5" vis="ECalLayerVis">
                <slice material = "LeadDiflourite" thickness = "40*mm" sensitive="yes" limits="cal_limits" vis="ECalSensitiveVis"/>
                <slice material = "Silicon"        thickness = "1*mm" vis="ECalAbsorberVis"/>
                <slice material = "siPCBMix"       thickness = "3*mm" vis="ECalAbsorberVis"/>
                <slice material = "Air"            thickness = "1*mm" vis="InvisibleNoDaughters"/>
</layer>
```

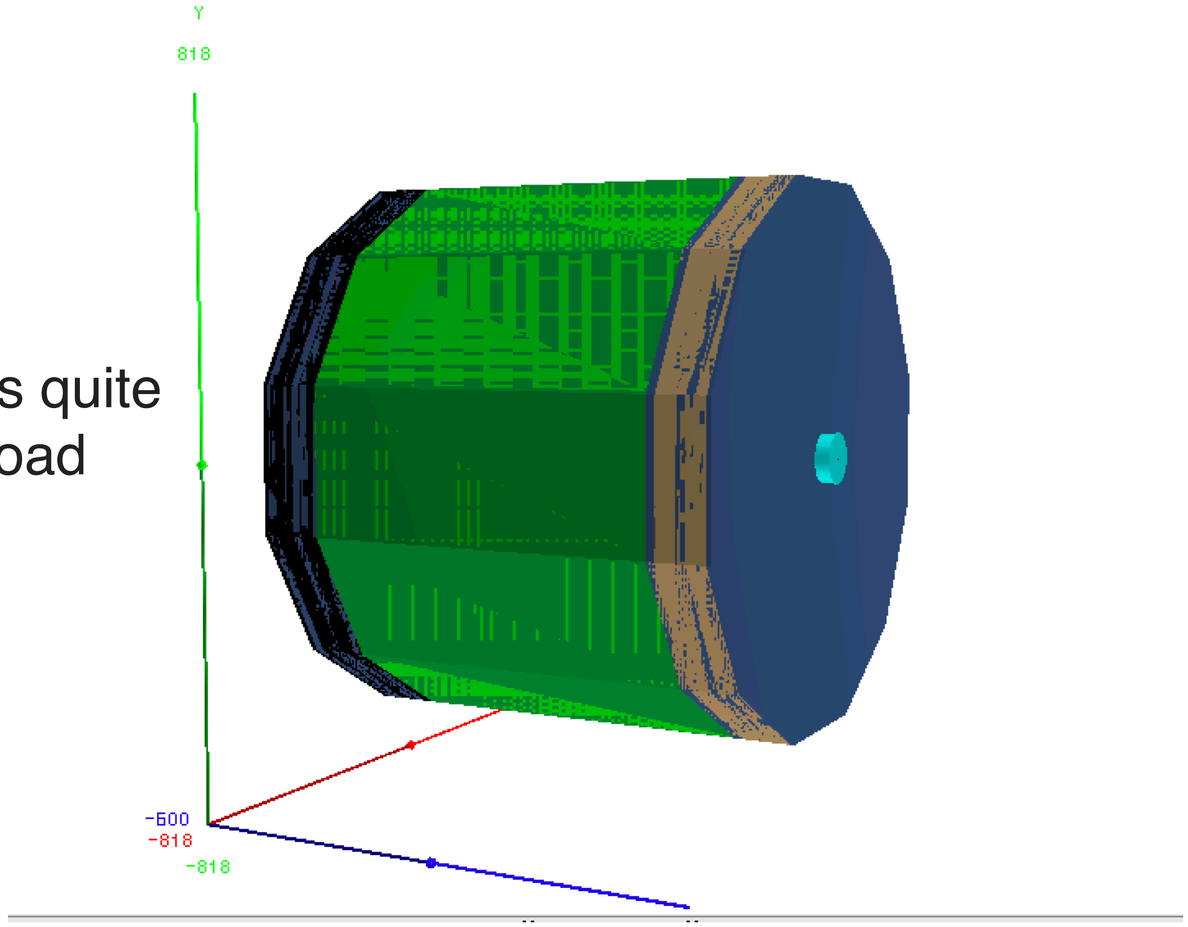
3. Run the simulation with the new geometry and visualize the events (as in previous exercise)

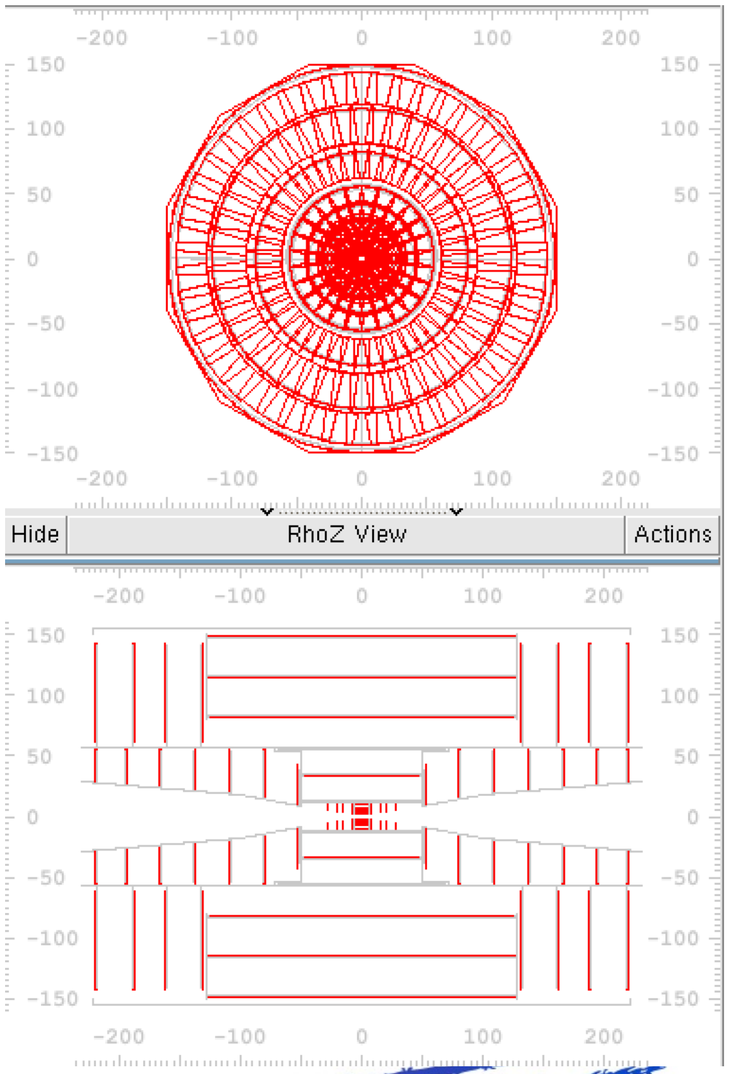
Hands-on: changing the calorimeter technology

The detector geometry can be inspected in details with

```
teveDisplay -compact ./MuColl_v1_mod1/MuColl_v1_mod1.xml
```

However this command takes quite a long time to load





And now … hands on!!

Backup

Detector

hadronic calorimeter

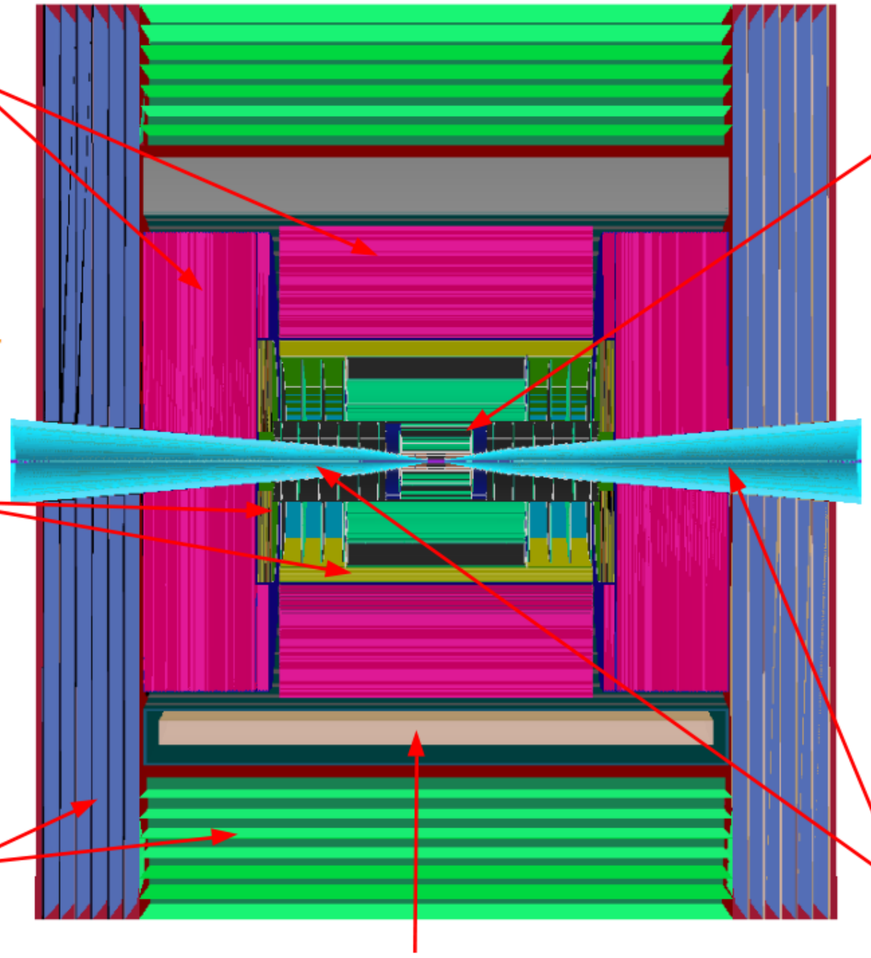
- 60 layers of 19-mm steel absorber + plastic scintillating tiles;
- 30x30 mm^2 cell size;
- 7.5 λ_I.

electromagnetic calorimeter

- 40 layers of 1.9-mm W absorber + silicon pad sensors;
- 5x5 mm^2 cell granularity;
- 22 X_0 + 1 λ_I.

muon detectors

- 7-barrel, 6-endcap RPC layers interleaved in the magnet's iron yoke;
- 30x30 mm^2 cell size.



superconducting solenoid (3.57T)

tracking system

- Vertex Detector:
  - double-sensor layers (4 barrel cylinders and 4+4 endcap disks);
  - 25x25 μm^2 pixel Si sensors.
- Inner Tracker:
  - 3 barrel layers and 7+7 endcap disks;
  - 50 µm x 1 mm macro-pixel Si sensors.
- Outer Tracker:
  - 3 barrel layers and 4+4 endcap disks;
  - 50 µm x 10 mm micro-strip Si sensors.

shielding nozzles

- Tungsten cones + borated polyethylene cladding.