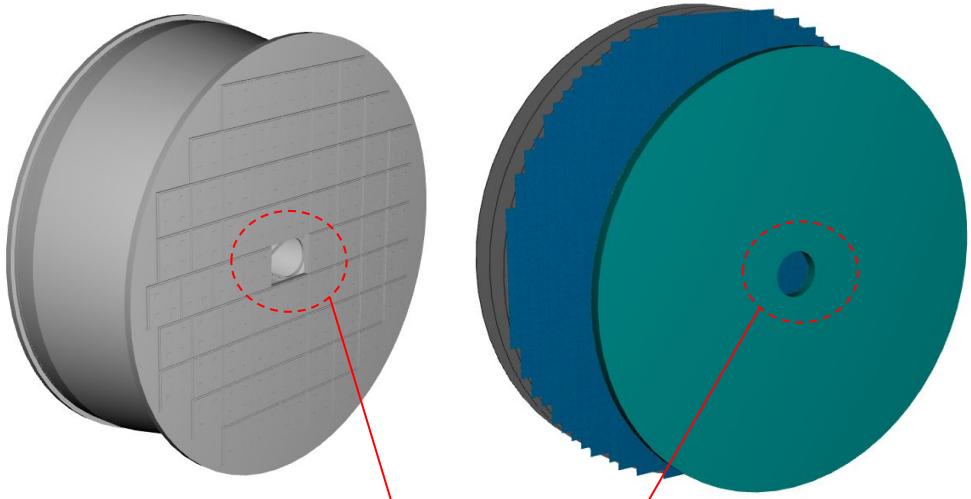
pfRICH Implementation in DD4Hep

Nov 12, 2023

Problem, Consueses and Plan (Old Slides)



CAD import

Standalone Geant4 import

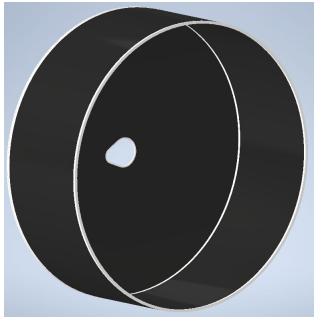
Beampipe implementation

Currently:

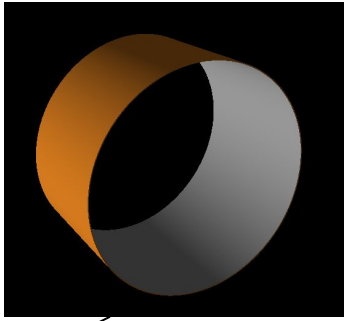
- **Two implementations:**
  - CAD import
  - Standalone Geant4 import
- **Problem:**
  - Neither implementations work within the ePIC DD4Hep framework
  - CAD import: significantly increase the overhead of the simulation
  - Standalone: doesn't remain non-standard beam pipe features
  - Both are not ideal
- **Consueses:**
  - A new implementation is needed
  - The new pfRICH shouldn't increase the simulation overhead
  - The new implementation should preserve the critical features from the CAD (such as the beam pipe geometry)

Ambitious plan, how do we do this?

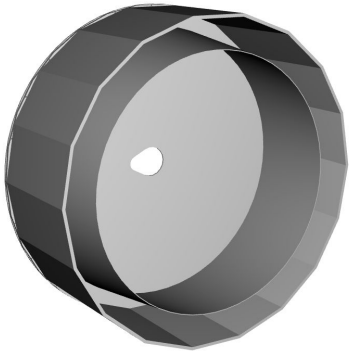
Tentative Plan (Old Slides)



pfRICH outer shell CAD GDML

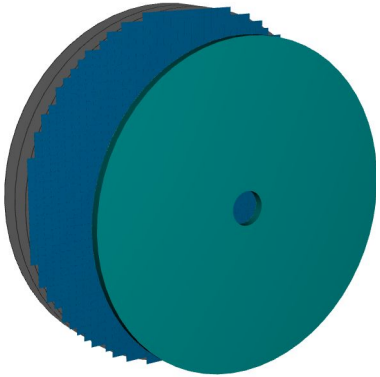


Geant4 Object xml



Step 1: Setting up a hybrid (CAD import-xml description) base xml description in DD4Hep

- Preserve CAD feature, serve as bridge between CAD and Geant4



Step 2: Rewrite/improve PFRICH_geo.cc, to parse the gdml geometry and merge the existing detector components (regular shaped)

Where we are

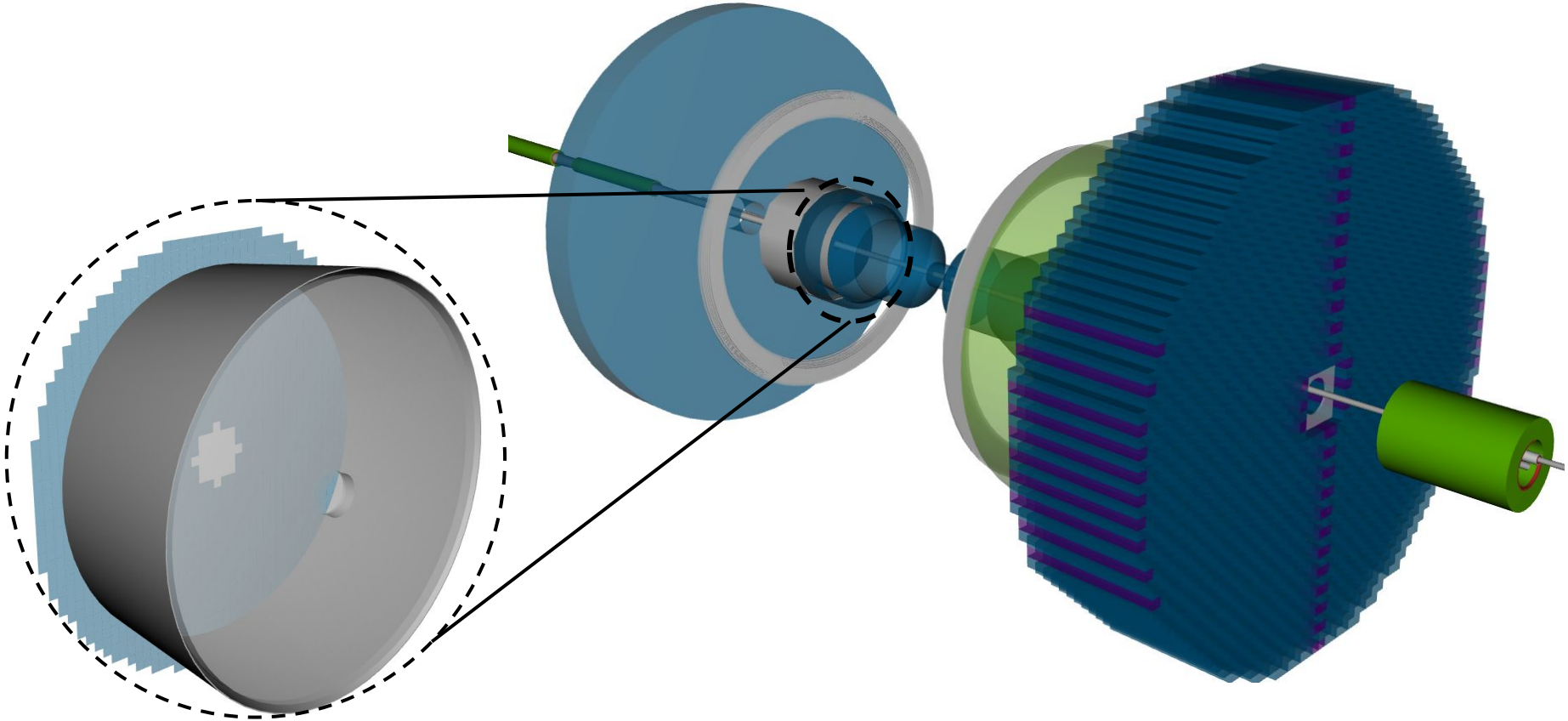
Step 3: Testing
GDML outer shell will be outside of the Cherenkov volume, and will be it will be excluded if it is identified to increase the simulation overhead

Step 4: Writing a geometry cross-checking software (between Geant4 vs CAD)

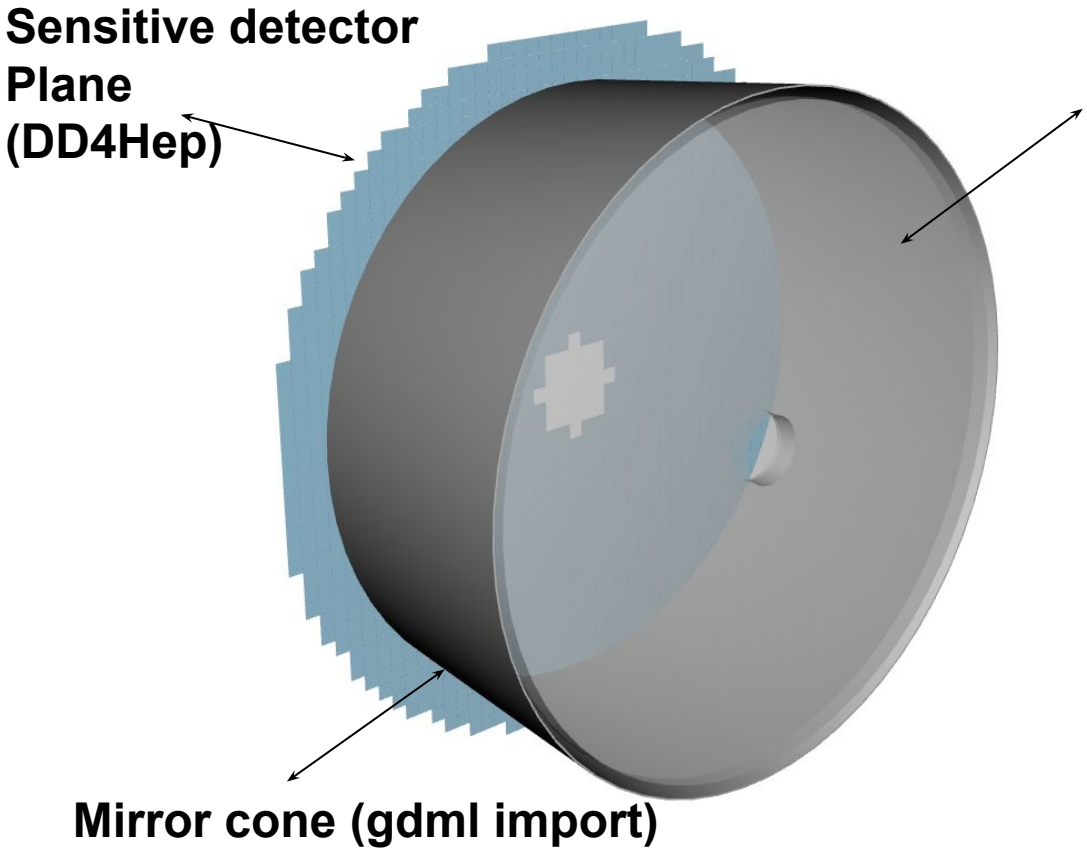
And

Reconstruction

Update of pfRICH in DD4Hep

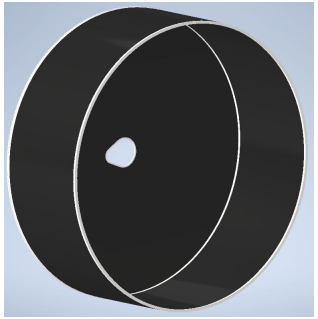


Update of pfRICH in DD4Hep

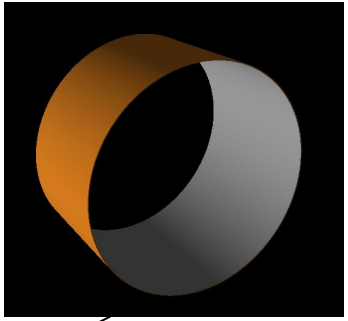


- **New update:**
  - Mirror Cone
  - Aerogel + Filter Plane
  - Detector Plane

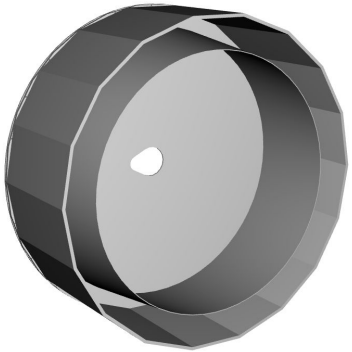
Tentative Plan



pfRICH outer shell CAD GDML

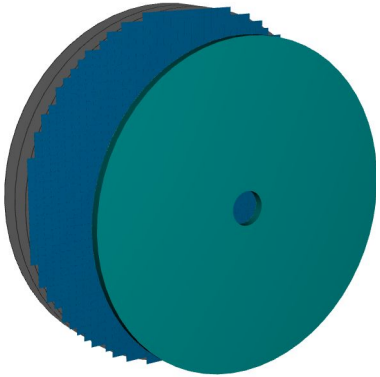


Geant4 Object xml



Step 1: Setting up a hybrid (CAD import-xml description) base xml description in DD4Hep

- Preserve CAD feature, serve as bridge between CAD and Geant4



Step 2: Rewrite/improve PFRICH_geo.cc, to parse the gdml geometry and merge the existing detector components (regular shaped)

Done !

Step 3: Testing
GDML outer shell will be outside of the Cherenkov volume, and will be it will be excluded if it is identified to increase the simulation overhead

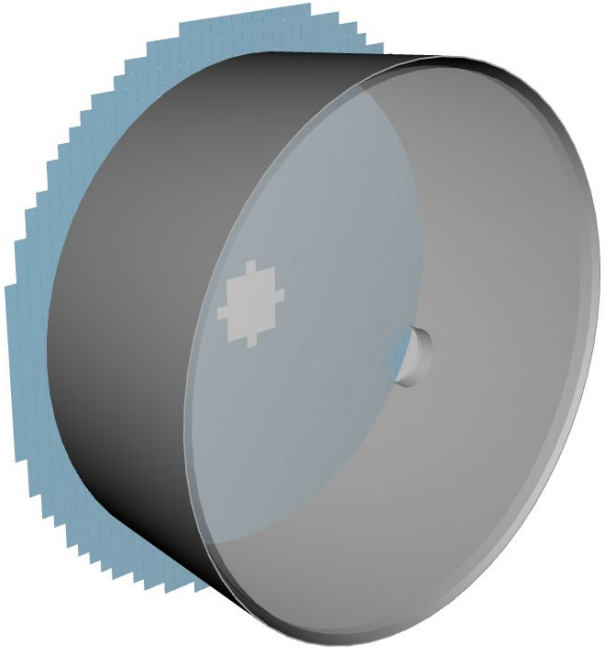
Where we are

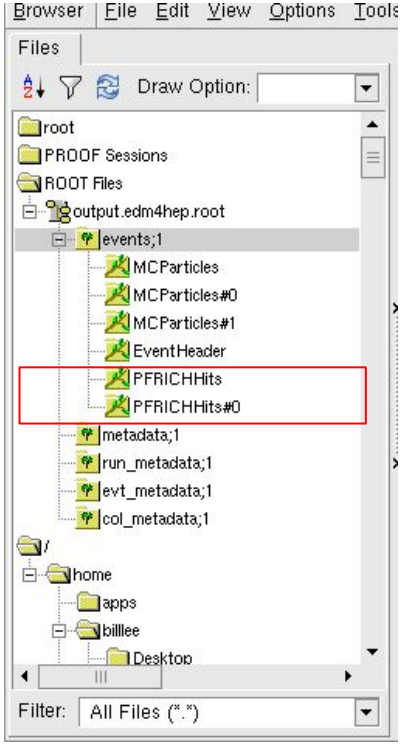
Step 4: Writing a geometry cross-checking software (between Geant4 vs CAD)

And

Reconstruction

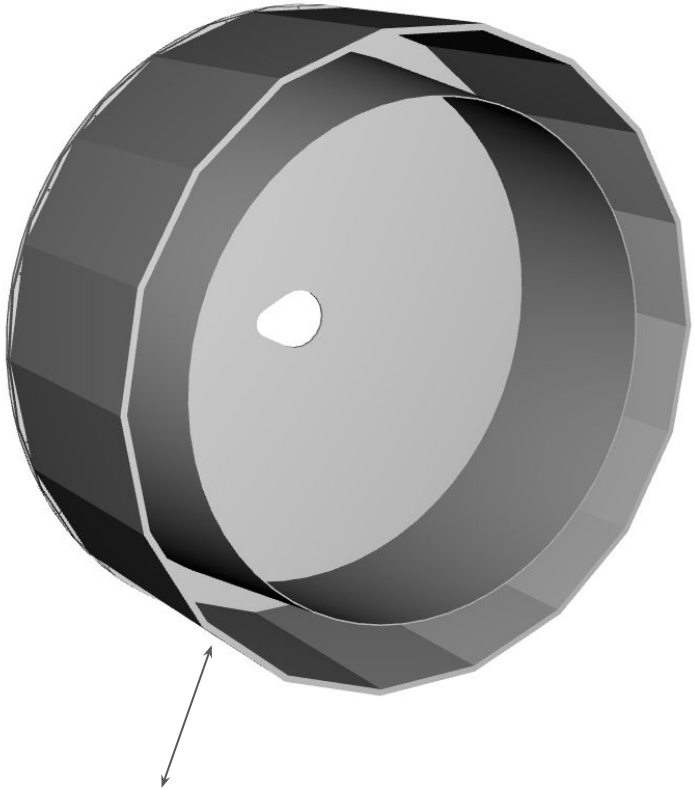
Simulation Read Out





- **The PFRICHHits works with the implementation**
- **Simulation ready !**

One Detail Regarding GDML Import



Outershell is not currently implemented

- **Issues with importing the GDML**
  - **Overlaps when importing the GDML structure and placement (our current route)**
- **Solution: important structure only**
  - Placement inside .cpp file

Parameters in the xml file

```
<!-- Actual global parameters -->
<constant name="PFRICH_length"            value="BackwardRICHRegion_length" /> <!-- vessel z-length -->
<constant name="PFRICH_zmax"              value="-BackwardRICHRegion_zmax"/>         <!-- vessel front -->
<constant name="PFRICH_zmin"              value="PFRICH_zmax + PFRICH_length"/>      <!-- vessel back -->
<constant name="PFRICH_rmin0"             value="-PFRICH_zmin * Eta3_9_tan * 0.95"/> <!-- bore radius at vessel frontplane -->
<constant name="PFRICH_rmin1"             value="-PFRICH_zmax * Eta3_9_tan * 0.85"/> <!-- bore radius at vessel backplane -->
<constant name="PFRICH_rmax"              value="BackwardPIDRegion_rmax"/>           <!-- vessel radius -->
<constant name="PFRICH_bore_slope"        value="(PFRICH_rmin1 - PFRICH_rmin0) / PFRICH_length"/> <!-- slope of bore radius -->

<!-- ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ -->

<constant name="PFRICH_proximity_gap"     value="30*cm"/>                        <!-- distance between aerogel exit plane and sensor entrance plane -->
<constant name="PFRICH_services_length"   value="15*cm"/>                        <!-- span of service materials behind the sensors -->
<constant name="PFRICH_wall_thickness"    value="0.5*cm"/>                       <!-- thickness of radial walls -->
<constant name="PFRICH_window_thickness"  value="0.1*cm"/>                       <!-- thickness of entrance and exit walls -->
<constant name="PFRICH_aerogel_thickness" value="3.0*cm"/>                       <!-- aerogel thickness -->
<constant name="PFRICH_filter_thickness"  value="0.3*mm"/>                       <!-- filter thickness -->
<constant name="PFRICH_aerogel_filter_gap" value="0.01*mm"/>                     <!-- air gap between aerogel and filter FIXME: currently a gas gap -->

<!-- FIXME: change to LAPPD(?) -->
<constant name="PFRICH_sensor_size_default"   value="25.8*mm"/>                      <!-- sensor side length (full size, with enclosure) -->
<constant name="PFRICH_sensor_size_lores"     value="PFRICH_sensor_size_default * 4"/>  <!-- lores sensor size, used for global vizualizations _only_ -->
<constant name="PFRICH_sensor_size"           value="PFRICH_sensor_size_default"/>
<constant name="PFRICH_pixel_gap"             value="0.2*mm"/>                       <!-- size of gaps between adjacent pixels AND gaps between edge pixels and se
side -->
<constant name="PFRICH_sensor_sensitive_size" value="PFRICH_sensor_size - PFRICH_pixel_gap"/> <!-- side length of photosensitive surface (to be segmented) -->
<constant name="PFRICH_sensor_thickness"      value="0.5*mm"/>
<constant name="PFRICH_num_px"                value="8"/>                            <!-- number of pixels along one side of the sensor -->
<constant name="PFRICH_pixel_size"            value="3.0*mm"/>                       <!-- a single SiPM pixel size -->
<constant name="PFRICH_pixel_pitch"           value="PFRICH_pixel_size + PFRICH_pixel_gap"/>  <!-- center-to-center distance between SiPMs in a 8x8 panel -->

<constant name="PFRICH_debug_optics"  value="0"/>
</define>
```

A Little messy, and requires touch up

Next step

- Configure gas volume and double check material profile
- Reflective coating of mirror cone
- Important outershell
  - Resolve the overlapping issue
- Generate photon occupancy plots
- Consolidate a list of parameters in the xml file

Backups

Update on Step 2

DD4hep_GdmlDetector.cpp

PFRICH_geo.cpp

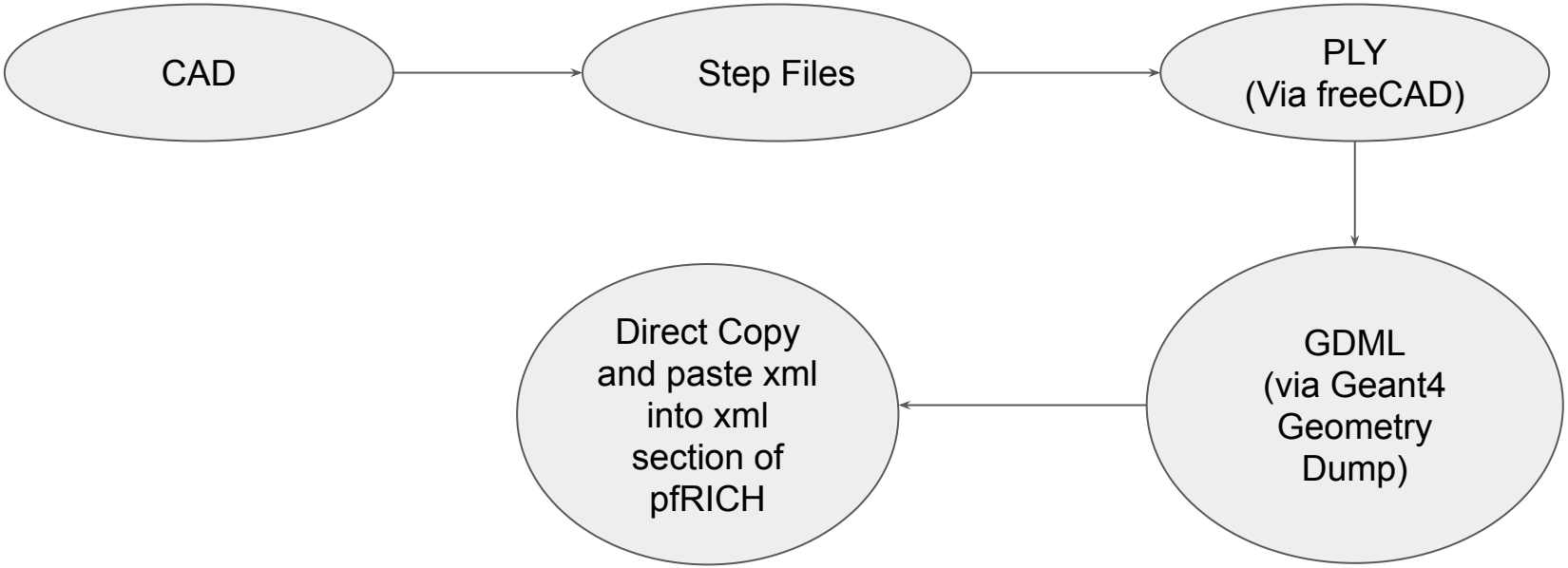
BarrelHCalCalorimeter_geo.cpp

```
// Copyright (C) 2022 Christopher Dilks, Sylvester Joosten

//----------------------------------
//  pfRICH: Proximity Focusing RICH
//  Author: C. Dilks
//----------------------------------

#include "DD4hep/DetFactoryHelper.h"
#include "DD4hep/OpticalSurfaces.h"
#include "DD4hep/Printout.h"
#include "DDRec/DetectorData.h"
#include "DDRec/Surface.h"

#include <XML/Helper.h>

// ROOT includes
#include "TGDMLParse.h"
#include "TGDMLWrite.h"
#include "TGeoElement.h"
#include "TGeoManager.h"
#include "TInterpreter.h"
#include "TUri.h"

using namespace std;
using namespace dd4hep;
using namespace dd4hep::rec;

// create the detector
static Ref_t createDetector(Detector& description, xml_h e, SensitiveDetect
{
```

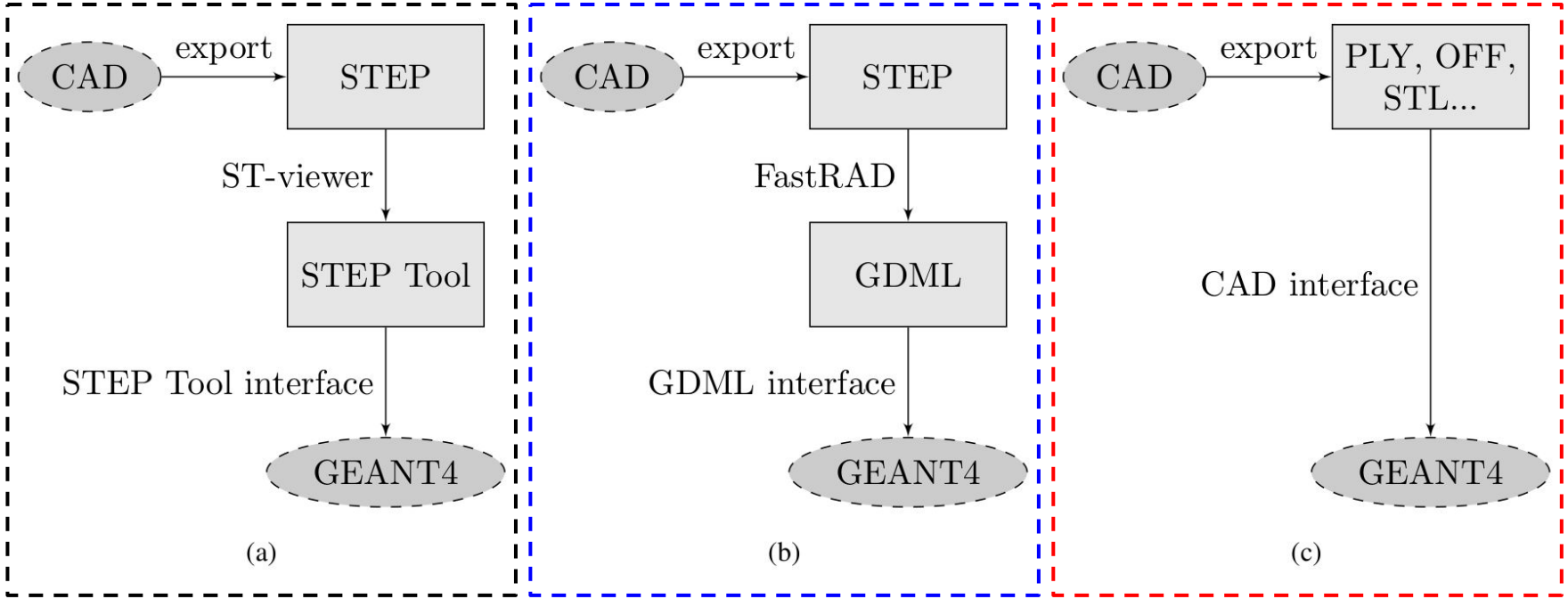
PFRICH_geo_v1.cpp

Branch:
https://github.com/billlee77/epic/tree/pfRICH

Documentation on Step 1



From CAD to Geant4 Model



Method 1

Indirect method

Method 2

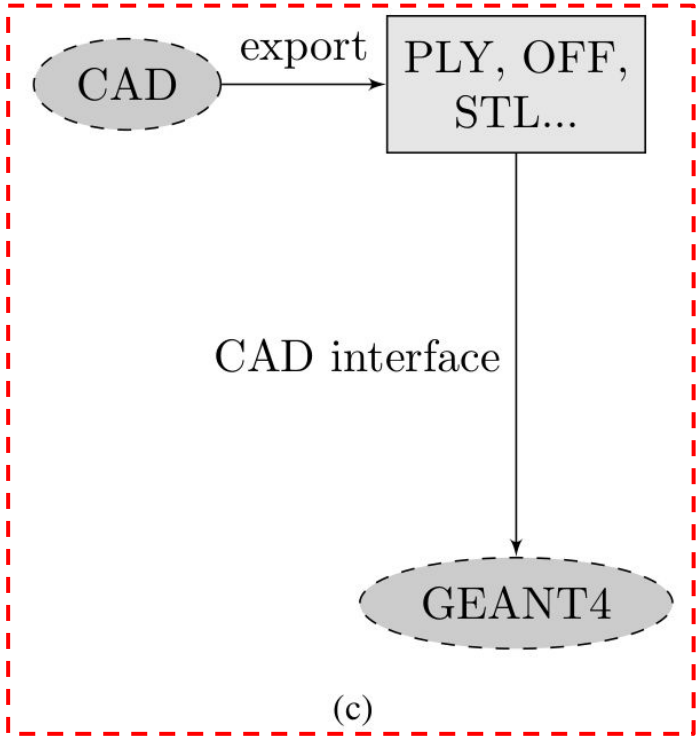
Indirect method

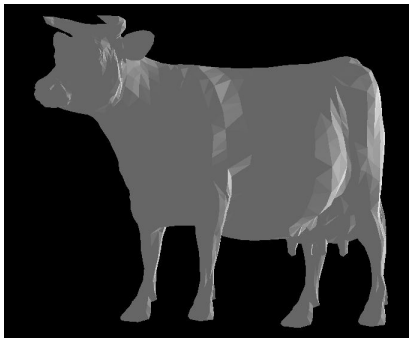
Method 3

Direct method

https://arxiv.org/pdf/1105.0963.pdf

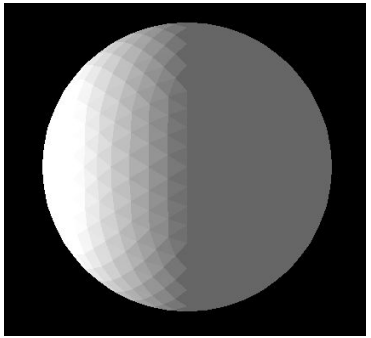
CADMesh Plugin





cow.obj

rabbit.stl



rabbit.ply

https://github.com/christopherpoole/CADMesh