EPJ Web of Conferences **337**, 01268 (2025) https://doi.org/10.1051/epjconf/202533701268
CHEP 2024

Experiences from the CBM collaboration: CAD to ROOT conversion for Detector Geometries

Simon Neuhaus[1,*], *Mehulkumar* Shiroya[2,3,**], *Omveer* Singh[2,***], and *Patrick* Dahm[3,****]

[1]Bergische Universität Wuppertal, 42119 Wuppertal, Germany
[2]Goethe-Universität Frankfurt, 60438 Frankfurt am Main, Germany
[3]GSI - Helmholtzzentrum für Schwerionenforschung, 64291 Darmstadt, Germany.

Abstract. Fully automated conversion from CAD geometries directly into their ROOT geometry equivalents is a topic of wide interest in particle physics experiment communities for some time. Tessellation of the surface of an intricate geometry is a powerful approach towards this goal, by potentially providing a shared geometrical representation with very good convergence even for the case of complex geometries. However, using tessellated geometries also requires significant computational effort for particle tracking inside and through tessellated objects.

In this paper, we first discuss the experiment and the methodology involved in tessellation and conversion. We report on the application and first experience of using two different software approaches. The two tools, VecGeom and TGeoArbN, were used for simulation of the same tessellated subdetector component. Our observations in this simulation with respect to obtained results and simulation speed are reported along with our general observation about the handling of these tools.

1 Introduction

The Compressed Baryonic Matter (CBM) experiment is being installed to use heavy-ion beams from the SIS100 synchrotron of the newly constructed Facility for AntiProton and Ion Research (FAIR) which is adjacent to and uses facilities of the long-established Helmholtzzentrum für Schwerionenforschung (GSI) in Darmstadt, Germany. Research goals of the CBM collaboration are varied. To the first instant, they relate to the topic of strongly interacting hadronic matter with initial investigations focused on determination of the phase transitions occurring in QCD which reveal themselves by varying the parameters of baryonic chemical potential (μ_B) and temperature (T). Fig. 1 shows the detector subssytems of the CBM experiment as they will appear after installation. Data collection is expected to commence in early 2028, focusing on collisions at beam energies ranging from $\sqrt{s_{NN}} = 2.9$ GeV to 4.9 GeV.[1]

*e-mail: sneuhaus@uni-wuppertal.de
**e-mail: m.shiroya@gsi.de
***e-mail: omvir.ch@gmail.com
****e-mail: p.dahm@gsi.de

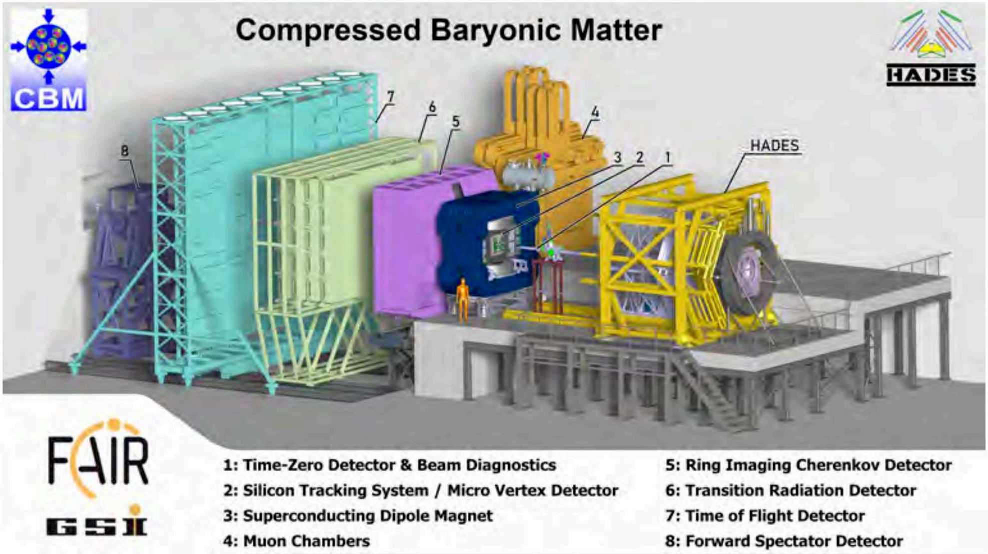

Figure 1. Experimental Setup of the CBM Experiment with all sub-detector system and HADES.

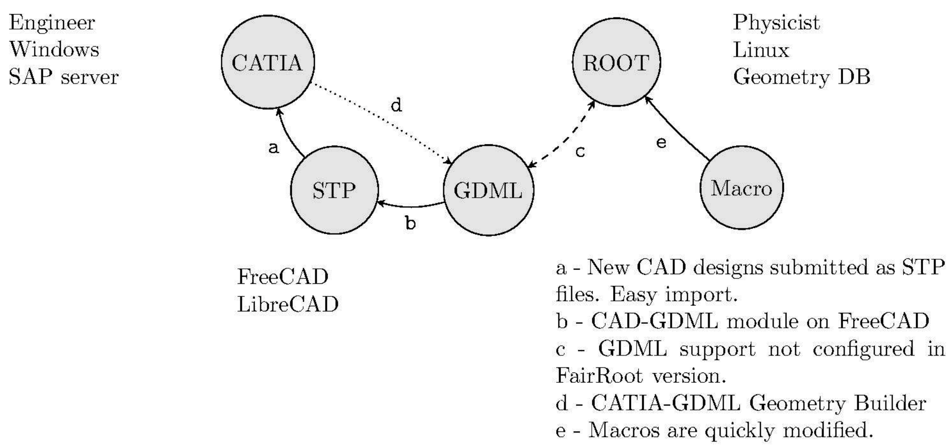

Figure 2. Outline of the current status of the geometry building from CATIA CAD models to ROOT geometries.

Since the CBM experiment will be operational in a low-beam energy regime, it requires an accurate representation of the simulation geometry to effectively analyze and understand the secondary particle background created during the particle collision. Software toolkits, such as ROOT and GEANT, are widely used to model detector geometries by describing shapes, composite structures, and their placement. However, the intricate details of detector geometries, as specified in CAD models, in the simulation geometries using the current toolkits pose significant challenges. To overcome the challenges, advancements have been made within those software toolkits. Techniques such as converting CAD models to the Geometry

Description Markup Language (GDML)[2] or using the Standard Triangle Language (STL) file format directly from CAD models with tessellated solids/shapes offer a robust solution to accurately describe complex detector geometries for simulation purposes. In this context tessellation describes the process of replacing a 3D shape with an unbroken and overlap free 2D triangle mesh of the surface. The figure 2 outlines the current status of the geometry building from CATIA CAD models to ROOT geometries. The motivation for this procedure is first and foremost, to reduce the workload on the designers of simulation geometries when iterations on structures are needed, as these geometries can quickly become very complex. Using a mostly automated approach enables rapid and precise feedback from physicists to engineers, helping to minimize secondary particles and reduce the noise background in measurements. In the section 2.2 and 2.3, there are two alternative ways of the CAD-to-ROOT method described. We have adapted the beam-cross simulation geometry in the RICH detector and performed the simulation comparison as described in the section 4.

2 Methods

In physics simulation, detector modelling and simulation play a crucial role in designing detector systems by evaluating their response using particle transport model libraries such as GEANT and FLUKA. Traditionally, particle transport and simulation studies in High Energy Physics (HEP) experiments rely on geometry constructed using a classical approach, primarily based on primitive solids from ROOT/GEANT. However, with advancements in technology, an alternative method has emerged, allowing to import the CAD based geometry model into the ROOT/GEANT. Currently, there are two approaches enabling this process that are discussed in details in the following sections.

2.1 TGeoTessellation

Tessellated solid-based geometry using ROOT can be implemented for the simulation using `TGDMLParser` after conversion of the Standard for the Exchange of Product Data (STEP) file into the GDML file format using CAD toolkits such as `Free-CAD, CATIA`, etc. Tessellated-based geometry cannot be used directly in ROOT for physics simulations if the ROOT version is below 6.32, as it lacks navigation functionality for tessellated solids and contains bugs. TGeoNavigator, a default navigator in ROOT, failed to trace the true shape and boundaries accurately. Consequently, these tessellated solids behave as simple boxes, following the characteristics of their parent class.[1] To ensure proper navigation functionality, ROOT must be compiled with Vectorized Geometry (VecGeom). Further details about VecGeom are discussed in section 2.2.

2.2 Solution A - Vectorized Geometry (VecGeom)

VecGeom[3, 4] is a geometry modeller library under development as a part of GEANT-V, focusing on the optimum efficient usage of the Single Instruction Multiple Data (SIMD) and Single Instruction Multiple Threads (SIMT) in heavily multi-threaded frameworks.[5]

ROOT supports navigation functionality for tessellated solids only if it is compiled with the VecGeom package[2], if it is using the VecGeom converter during geometry modelling and, if the shapes are written into the ROOT file. The converter[3] essentially transforms all

[1] https://root.cern.ch/doc/master/classTGeoTessellated.html
[2] https://gitlab.cern.ch/VecGeom/VecGeom
[3] https://root.cern/doc/master/tessellatedNav_8C.html

the shapes in each `TGeoVolume` (including tessellated) into VGShapes as defined within the ROOT, `TGeoVGShape` class.[4] This integration facilitates navigation capabilities for the tessellated shapes within the ROOT framework.

2.3 Solution B - TGeoArbN

TGeoArbN[6] is another tool for including tessellation in the ROOT geometry description for GEANT-based detector simulation. It is actively developed at the University of Bonn as part of the NRW-FAIR network activities, primarily for the PANDA experiment. TGeoArbN provides its own independent particle propagation / navigation routines for the tessellated objects, which is needed for particle physics detector simulation. Depending on this independent navigation implementation, tessellation is provided both for GEANT3 and GEANT4-based simulation frameworks. TGeoArbN can be easily included into an already existing ROOT- or FAIRROOT-based simulation framework, where tessellated volumes using TGeoArbN can be handled like any other TGeoVolume from ROOT. This has the benefit of a simple possibility to add them to larger detector geometries and the disadvantage of the need of an own input (STL) file form the CAD model for each different material.

In order to reduce the computational effort and to speed up the simulation time, TGeoArbN includes an optional Octree-based partitioning scheme for subdividing the geometrical object. The purpose of Octree is to avoid looping over all triangles of the mesh by using a 3D version of a binary decision tree. Octree splits the cuboids of the surrounding box in eight smaller same sized cuboids and recursively repeats this for each of the cuboids until a given depth or until they are considered as empty. A subcuboid counts as empty if the number of mesh elements inside is below a predefined number.

3 Experience from CBM

Within the CBM collaboration, tessellation is starting to be routinely used by different subdetector groups, applying different approaches and different concepts.

The Cherenkov detector (RICH)[7] group started to use TGeoArbN-based tessellation for selected geometries only. At present, the aim here is to apply tessellation mainly in the initial design phase to speed up the iteration cycle and to switch to the classical implementation finally for faster simulation speed. Others, like the Silicon Tracking detector (STS)[8], Transition radiation detector (TRD)[9], or BeamMonitor (BMON) detector groups currently rely on VecGeom-based tessellation, either to run full simulations (STS) or for visualization purpose only (TRD, BMON). For certain complex geometries in STS, the goal is to use the tessellation-based implementation also as final solution. Distinct from other detector groups the BMON group uses tessellation to describe the full detector and not only parts of it.

Having the same geometry implemented both based on tessellation and based on classical approach provides a good opportunity for cross-checking the final version and to quantify the impact of simplifications of the geometry. Relying solely on tessellation on the other hand can significantly reduce the effort needed for precisely implementing the geometry, and helps to assure optimal agreement between CAD- and GEANT simulation geometries. However, this would have the drawback of a runtime increasing of the simulations compared to the use of a classical approach in the final simulations.

[4] https://root.cern/doc/master/dir_f659b5a9ca37b079a242c72a02a19916.html

Figure 3. Picture of the Beamcross which is used for the comparing of the different methods for tessellation.

4 Comparison

The following section focuses on a comparison of TGeoArbN and VecGeom.

For proper navigation in a simulation using a tessellated solid, ROOT must be compiled with the VecGeom library package. TGeoArbN, in contrast to this, can be easily used on top of an already existing, and maybe shared, ROOT installation. This simplifies the first integration steps.

For both approaches, VecGeom and TGeoArbN, setting up a new geometry starts with importing a STL mesh file from the CAD software. TGeoArbN can load the mesh directly to create a volume in ROOT. VecGeom, in comparison, needs some intermediate steps.

This comparative study of both approaches uses UrQMD simulations for 8 A GeV/c Au-Au central collisions with 10000 events each, which applies tessellation for a single detector element, a beam pipe support structure, which has a mesh with 2274 triangles and is visible in figure 3, at position in z- or beam-direction from 310 cm to 360 cm regarding the center of magnet as origin of the coordinate system. In total, results of four simulations, differing only in the geometry building of the tessellated volume, are compared. One simulation uses VecGeom for the tessellated beam pipe support, two simulations use TGeoArbN for it, one with and one without Octree, and the fourth simulation leaves this volume completely out. The simulation without the beam pipe support serves for evaluation of the impact of the tessellated volume in comparison with the other simulations.

The histogram in the left part of figure 4 shows the 2D-distribution of the xy-coordinates of the conversion vertices with z-coordinates in the region of the added geometry for the simulation with VecGeom. The right histogram in figure 4 shows the similar distribution for the geometry with the TGeoArbN-based tessellated Beamcross. In both approaches an increase of conversion vertices reflecting the shape of the cross is clearly visible.

The plot in figure 5 compares the runtimes of the transport step for the four different simulations, running on the GSI cluster Virgo3. The runtime for the geometry without the Beamcross is included for comparison and could be regarded as an offset caused by the simulation of the classically implemented detector parts. The fastest simulation with the Beamcross is the one using the VecGeom with an increase of 7.6 min compared to baseline simulation. The simulation with the geometry which used TGeoArbN with the build-in functionality of Octree, takes 10 min (29 %) longer then the one using VecGeom. The TGeoArbN-based simulation without Octree needs more than three times the time to run.

For a quantitative comparison of both methods, VecGeom and TGeoArbN, figure 6 shows the ratio of the number of conversion vertices as function of coordinate in beam direction in the left plot and in the transversal xy-plane in the right plot. The ratio in both projections is largely compatible with one, proving the good general agreement of both approaches. Addi-

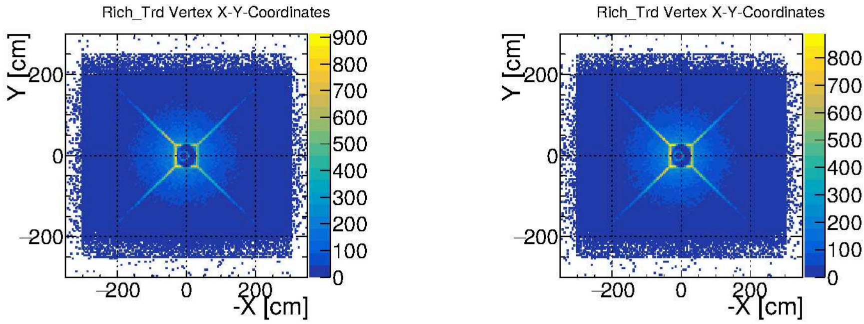

Figure 4. Plot of the xy-coordinate of the starting vertex from MC tracks with z-coordinate in the region of the Beamcross. Left for the simulation using VecGeom and right for the one using TGeoArbN.

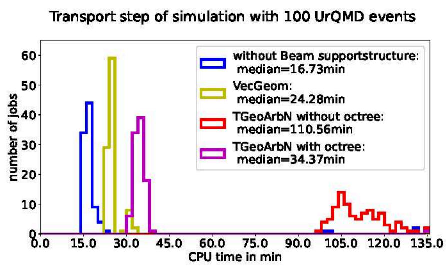

Figure 5. Runtime for the transport step of the simulations using the GSI cluster Virgo3.

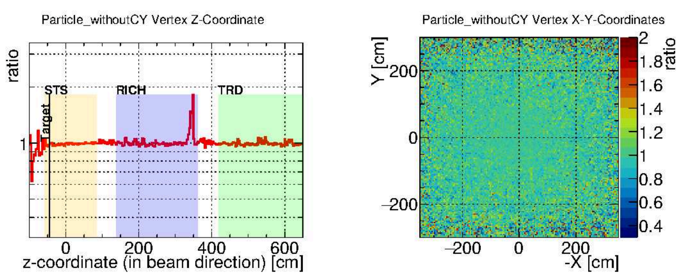

Figure 6. Ratio of the distribution of the starting vertex from MC tracks between simulation with Beamcross using TGeoArbN (without Octree) and using VecGeom.

tionally to statistical fluctuations, especially in the region before the target with low statistics, there is at the beginning of the tessellated structure, around $z = 350\,\text{cm}$, a distinct peak in ratio with values up to 1.8. Zooming into the z-range of this peak, the corresponding transversal xy-projection in figure 7 clearly reflects the shape of the beam support structure. This reveals some differences in the VecGeom and TGeoArbN-based tessellation results. The reason for this difference, which only appears at the upstream side of the Beamcross, is still not understood. A simple shift between both geometries has already been excluded.

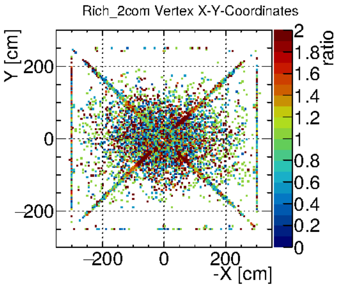

Figure 7. Ratio of the distribution of the xy-coordinate from starting vertex from MC tracks with z-coordinate in region of the peak in figure 6 between simulation with Beamcross using TGeoArbN (without Octree) and using VecGeom.

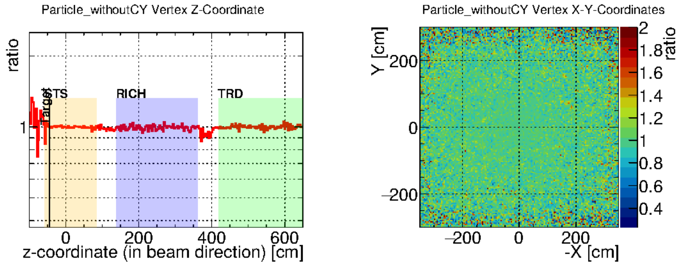

Figure 8. Ratio of the distribution of the starting vertex from MC tracks between simulation with Beamcross using TGeoArbN with and without Octree.

The ratio of the conversion vertex distribution between simulations with Beamcross using TGeoArbN with and without Octree, shown in figure 8, proved, that the reduction of the runtime, caused by using Octree, does not change the results of this simulation.

Future work is needed, to understand the reason for the observed difference between the simulation using VecGeom and using TGeoArbN. A working hypothesis is that the peak seen in figure 6 is caused by some surface effects in at least one of both methods. An initial comparison of conversion density inside a simple steel plate, implemented both using TGeoArbN-based tessellation and using a classical TBox, did not reveal any (significant) differences.

5 Conclusion

Tessellation is a very encouraging concept to improve the geometry building in ROOT from an already existing CAD model. Two tools aiming to make tessellated meshes usable in GEANT-based particle physics simulations are tested. A comparison between both tools, VecGeom and TGeoArbN, revealed some minor, not yet understood differences in the results. The simulation using VecGeom is faster but the application of TGeoArbN is easier to use. Some future studies, to understand the observed difference and to find out which simulation is more realistic, are needed to decide which tool to prefer. Because of the significant increase of the simulation runtime by adding single tessellated volumes to the geometry it is recommended to use tessellated shapes only in the development and replace them with simple ROOT geometries after setting the final design.

Acknowledgement

Work supported by “Netzwerke 2021”, an initiative of the Ministry of Culture and Science of the State of Northrhine Westphalia, and GSI.

TGeoArbN is a development of the PANDA group at university Bonn, made available to us as part of the joint NRW-FAIR network activities.

References

[1] P. Senger, *"CBM/FAIR capabilities for charm and dilepton studies"*, 5th International Workshop on Critical Point and Onset of Deconfinement - CPOD 2009, June 08 - 12 2009. https://pos.sissa.it/071/042/pdf

[2] R. Chytracek, J. McCormick, W. Pokorski, G. Santin, *"Geometry Description Markup Language for Physics Simulation and Analysis Applications"*, IEEE Trans. Nucl. Sci., Vol. 53, Issue: 5, Part 2, 2892-2896.

[3] J. Apostolakis, et al., *“Towards a high performance geometry library for particle-detector simulations.”*, Journal of Physics: Conference Series. Vol. 608. No. 1. IOP Publishing, (2015)

[4] S. Wenzel, J. Apostolakis, and G. Cosmo, *“A VecGeom navigator plugin for Geant4”*, EPJ Web of Conferences 245, 02024

[5] Amadio, G., Ananya, A., Apostolakis, J. et al., *"GeantV"*, Comput Softw Big Sci 5, 3 (2021). https://doi.org/10.1007/s41781-020-00048-6

[6] B. Salisbury, C. Schmidt, T. Stockmanns, U. Thoma, *"Towards an automatic geometry conversion: TGeoArbN - a ROOT compatible triangle mesh geometry implementation"*, CHEP’2024 proceedings (2025)

[7] S. Neuhaus, C. Pauly, and K.-H. Kampert, *“Applying TGeoArbN based tessellation in the CBM geometry description”*, CBM Progress Report 2023 pg. 168-169

[8] O. Singh, M. Shiroya, F. Uhlig, V. Friese, D. Emschermann, E. Clerkin, A. Toia, C. Blume, L. Wahmes, and O. Vasylev, *“Modelling of simulation geometries using Tessellated Shapes with the Vectorized Geometry (VecGeom) package”*, CBM Progress Report 2023 pg. 166-167

[9] C. Blume, P. Kähler, A. Meyer-Ahrens, O. Singh, and L. Wahmes, *“Towards an optimized TRD geometry”*, CBM Progress Report 2023 pg. 105

[10] E. Clerkin and P. Dahm, *“A review of changes to technical and simulation subsystem geometries”*, CBM Progress Report 2023 pg. 155-158