Interface to Unity for High Energy Physics detectors visualization

Tianzi Song[1], Kaixuan Huang[1], Yumei Zhang[1], Zhengyun You[1]
[1]Sun Yat-sen University, China

ACAT 2024
22nd International Workshop on Advanced Computing Analysis Techniques in Physics Research

Abstract: The visualization process of the detector is one of the important problems in high energy physics (HEP) software. At present, the description of detectors in HEP is complicated. Industry professional visualization platforms have the most advanced visualization capabilities and technologies, which can help to achieve the visualization of detectors. The work is to find an automated interface to efficiently convert all detector descriptions from HEP experiments directly to 3D models in Unity. This work has great potential to play an auxiliary role in detector design, HEP offline software development, physical analysis, and other parts of HEP experiments, and it also provides a good foundation for future research such as event display.

Introduction

The visualization of detectors is an integral aspect throughout the entire process of HEP experiments. We have significant demands for detector visualization in various aspects including detector design, detector assembly and commissioning, experiment operation and maintenance, data quality monitoring, simulation and reconstruction, as well as physics analysis. Moreover, detector visualization implies the possibility of achieving event display, which may hold significant implications for physics analysis. The HEP Software Foundation Community White Paper Working Group has also outlined guidelines for research directions in visualization.

Advantages of visualization technology from industry

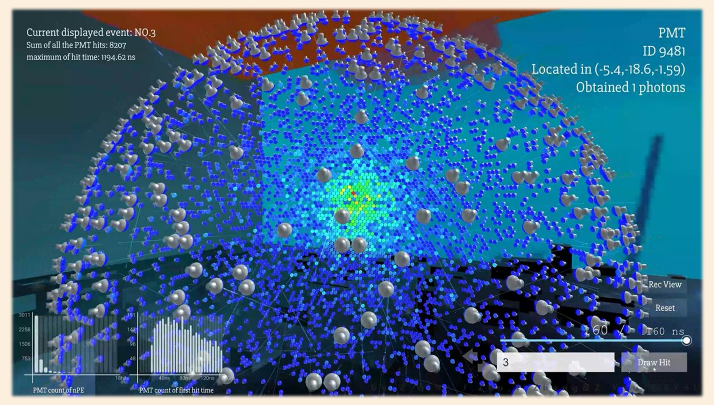
Currently, there are several software and platforms for detector visualization in HEP experiment field. But in comparison, visualization technology from industry has more evident and numerous advantages.

Unity is a professional video and game production engine, which can help to visualize HEP detectors.

- Professional 3D software.
- Provide access to VR or AR.
- Supports more than 20 platforms.

Although several HEP experiments have made targeted visualization software, such as ELAINA for JUNO and CAMELIA for ATLAS, based on Unity, we hope to complete the HEP experiment detector universal visualization interface.

Game developed by Unity



JUNO event display - ELAINA

Detector Description

HEP experiment detectors are usually large-scale, complex and precise detection instruments, and their internal detection units often need to be optimized and upgraded. As a result, if we want to implement the detector visualization based on Unity, it is almost impossible to reconstruct a complete and accurate model for each HEP detector in Unity. This means we need to import the existing detector description into unity for visualization.

Detector description in HEP now:

- **GDML**
- **ROOT**
- **DD4hep**
- **Geant4**

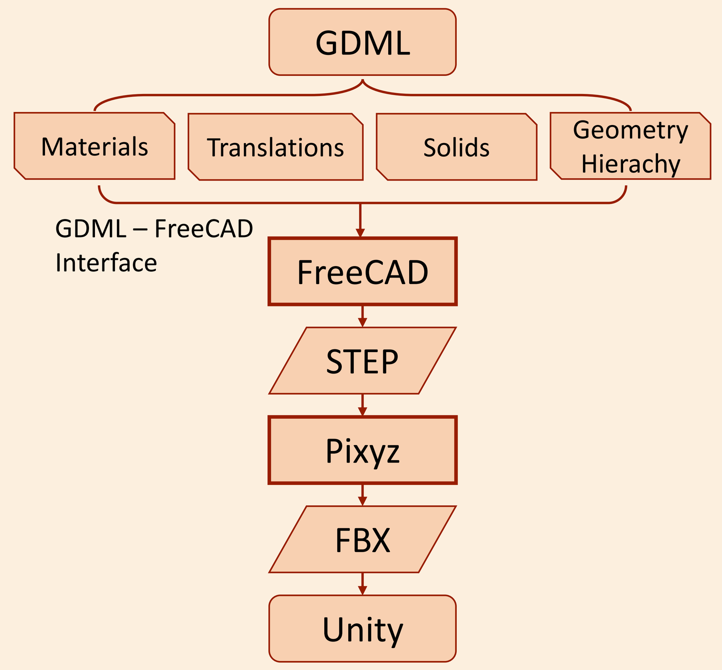
None of them can be directly imported into Unity

Develop a method for automatic detector transformation!

The method should work for all detectors and all formats, while keeping consistency.

Methodologies

- **A feasible transformation method provided by other works:**



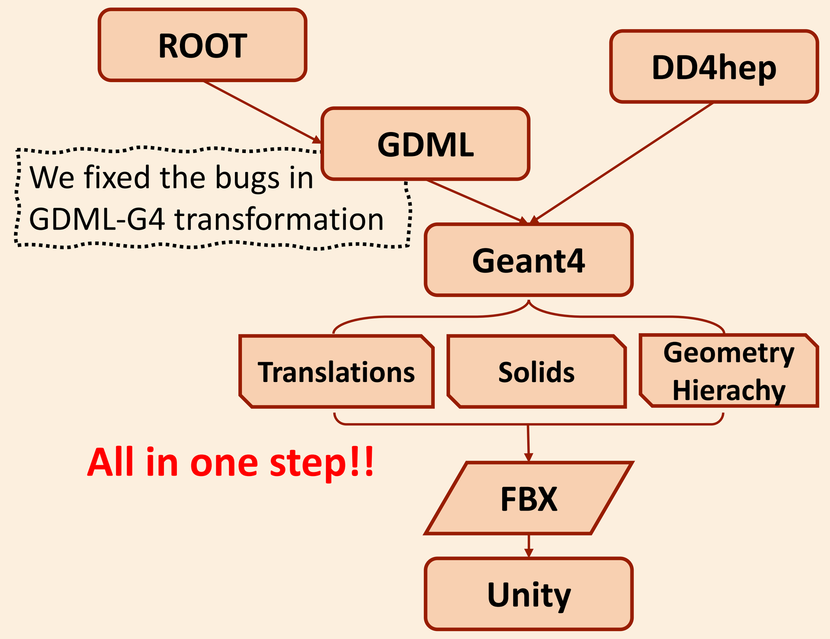
Result of previous work:

- Maintain the unique identifier of each detector unit.
- Provides richer visualization properties.

Disadvantage of previous work:

- Too complicated and time consuming.

- **A new method provided by this work: (develop based on HSF Geometry Writer)**



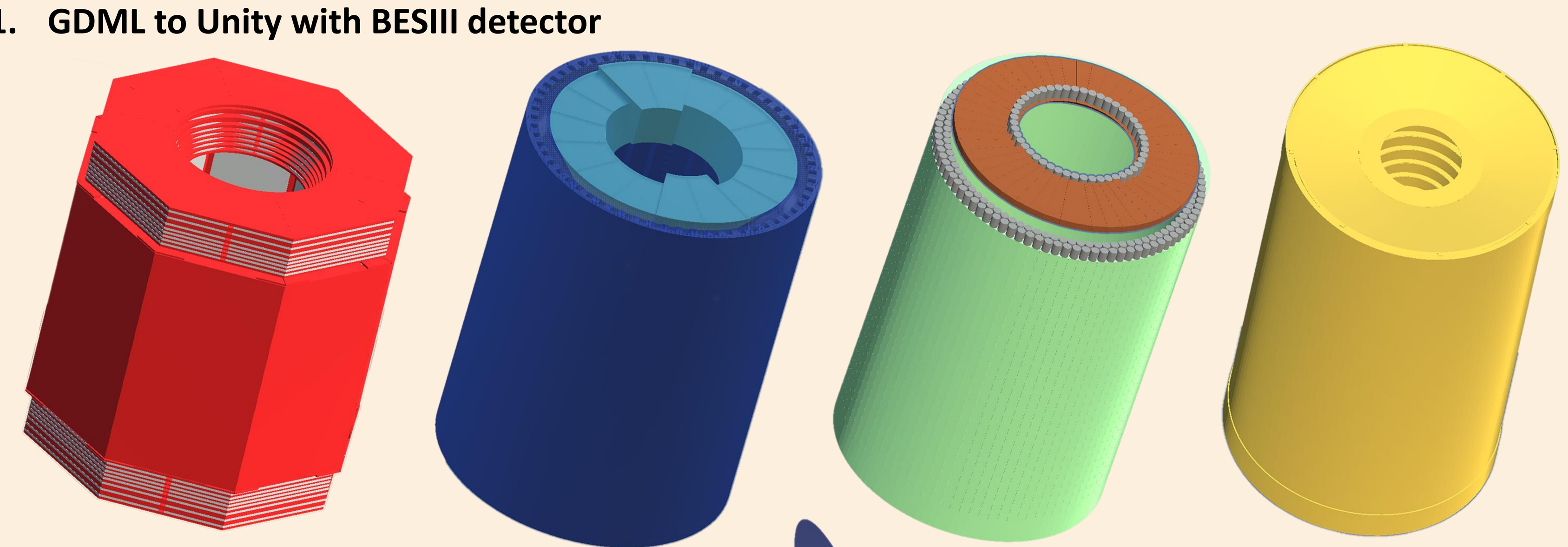
Improvement of this work:

- Maintain the unique identifier.
- Fine tuning of configuration to solve crash caused by complicated geometry.
- Support self-defined shapes and geometry classes
- The steps are easier and faster.
- Is able to assist all four detector descriptions.
- Running in Geant4, it's totally free

Visualization in Unity

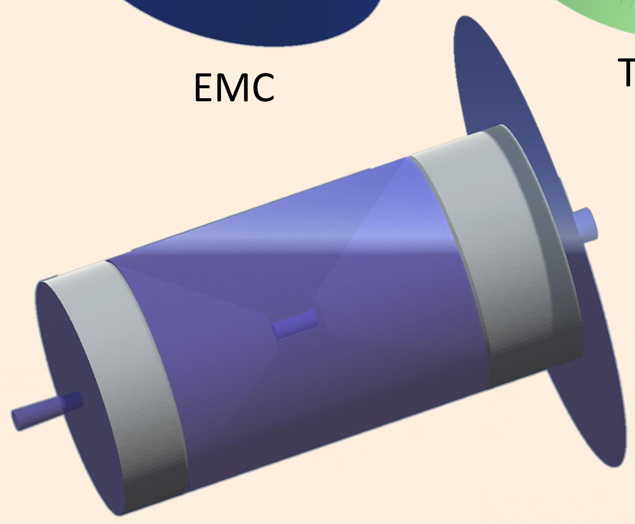
Currently, the interface provided in this work is capable of converting all four detector descriptions into FBX-formatted files, which can be directly imported into Unity for display.

1. **GDML to Unity with BESIII detector**



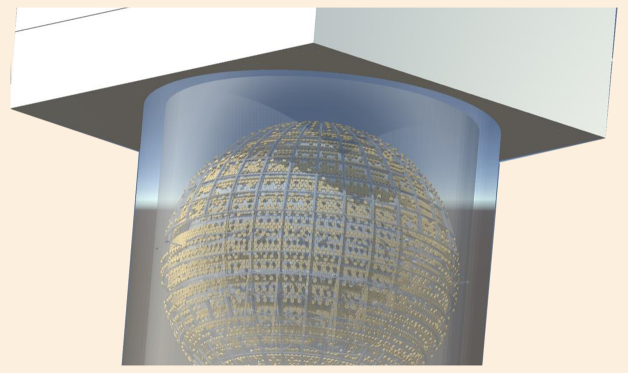
MUC EMC TOF MDC

2. **ROOT to Unity with EicC detector**

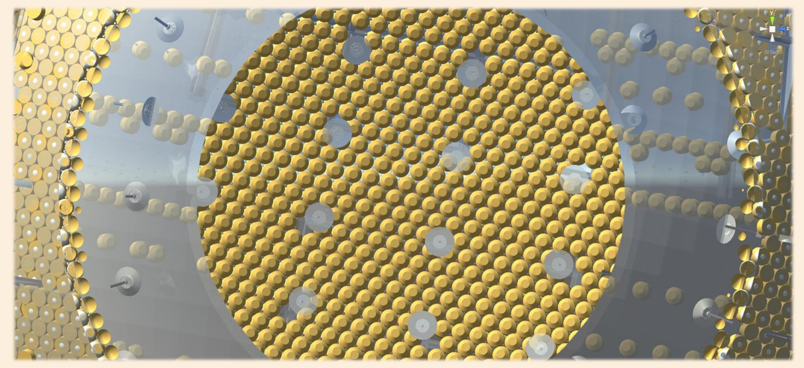


Visualization in Unity

3. **Geant4 to Unity with JUNO detector**



side view from outside of JUNO detector

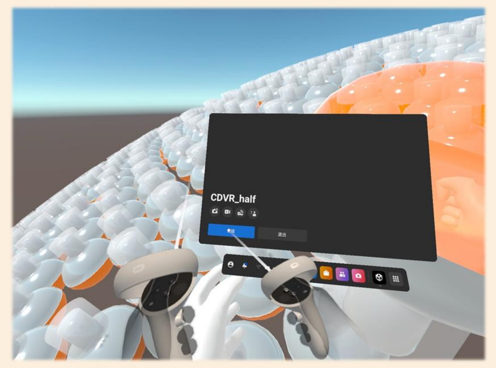


Inner view of JUNO central detector

Further applications

With the FBX files converted from four descriptions, we can visualize detectors in Unity directly, which means we can develop more technology based on Unity. Which is promising for us in the future:

- **Virtual Reality (VR) / Augmented Reality (AR)**

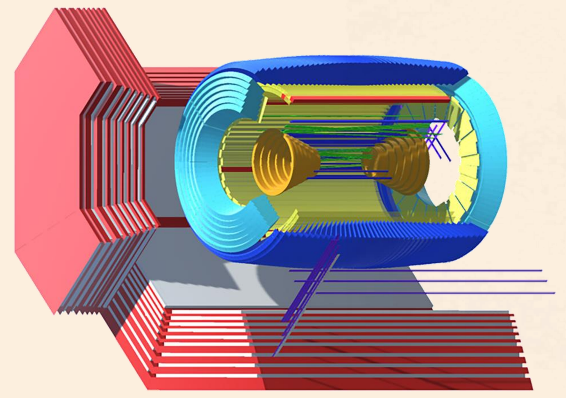


JUNO detector in VR

Unity provides a direct interface to AR or VR, where we can upgrade to more and richer interactive content. This content can be done based on hardware such as HTC Vive, Oculus Quest2, Apple Vision Pro and so on.

The figure on the left top shows how the JUNO detector behaves in VR interactions, which can assist in the design, assembly and operational monitoring of the detector.

Oculus Quest2



- **Event display**

The right figure is the event display shown in Besvis, which is based on ROOT. In the future, we can develop event display software for various detectors based on Unity more conveniently. In addition, we can develop real-time event display, 3D example display video production technology, which can further enrich our operation monitoring and physics analysis.

Reference

[1] HEP Software Foundation Community White Paper Working Group---Visualization[J]. Bellis M, et al., arXiv:1811.10309, 2018.

[2] Method for detector description transformation to Unity and application in BESIII[J]. Huang K X, et al., Nuclear Science and Techniques, 2022, 33(11): 142.

[3] Ric-bianchi, tpmccauley, et al., (2018) Visualization[source code].https://github.com/HSF/Visualization.

Poster: Tianzi SONG

Email: songtz@mail2.sysu.edu.cn