Method for HEP detector description and visualization in Unity

Tianzi Song（宋天资） Kaixuan Huang（黄凯旋）
Yumei Zhang（张玉美） Zhengyun You（尤郑昀）

中山大學
SUN YAT-SEN UNIVERSITY

songtz@mail2.sysu.edu.cn

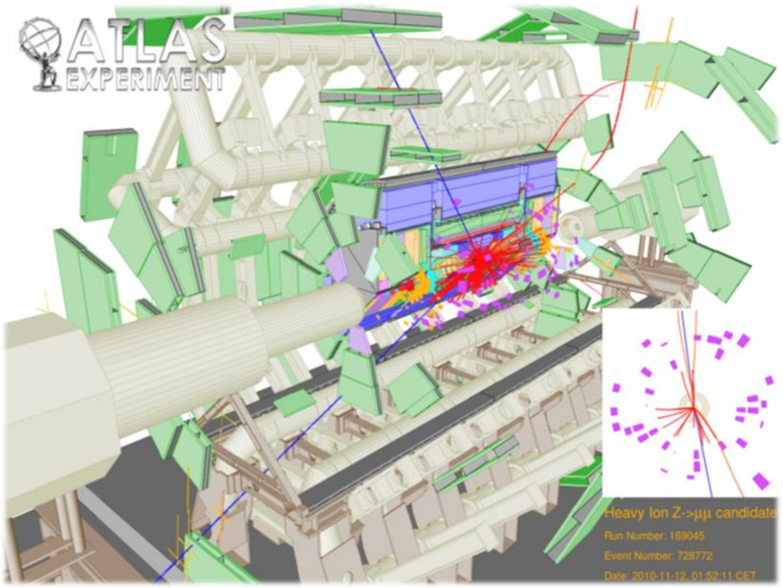
2023/6/11

Outline

1. Introduction
2. Methodologies
3. Visualization in Unity
4. Further applications
5. Summary

Visualization requirements for HEP experiments

- Detector design
- Detector construction & assembly
- Detector commissioning
- Experiment operation & maintenance
- Data quality monitoring
- Simulation & reconstruction
- Event display
- Physics analysis
- Education
- Outreach



mp-ph] 26 Nov 2018

HEP Software Foundation Community White Paper Working Group – Visualization

HEP Software Foundation: Matthew Bellis[a,b] Riccardo Maria Bianchi[c,1] Sebastien Binet[d] Ciril Bohak[e] Benjamin Couturier[f] Hadrien Grasland[g] Oliver Gutsche[h] Sergey Linev[i] Alex Martyniuk[j] Thomas McCauley[k,1] Edward Moyse[l] Alja Mrak Tadel[m] Mark Neubauer[n] Jeremi Niedziela[f] Leo Piilonen[p] Jim Pivarski[q] Martin Ritter[r] Tai Sakuma[s] Matevz Tadel[m] Barthélémy von Haller[f] Ilija Vukotic[t] Ben Waugh[j]

Visualization Technology from Industry

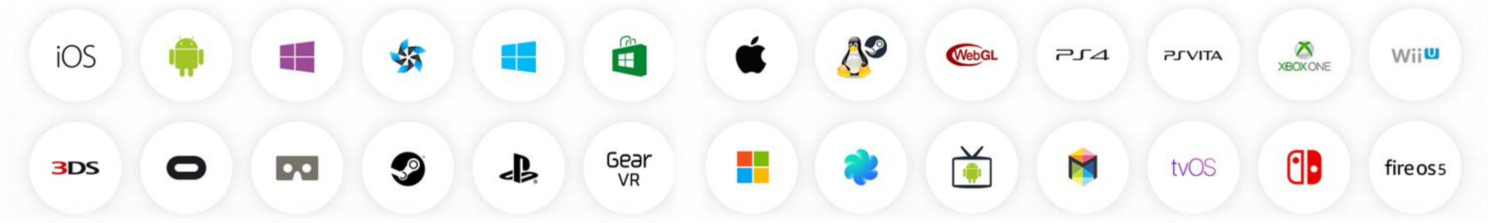
- More advanced visualization techniques
- Creation for game, film, video, education, art, industrial design and training
- Professional, stable and long-term support from industry community
- Software – Platform – System – Hardware seamless integration
- Outstanding platforms such as Unity, UnReal

Digital Twins in Unity

Rendering in UnReal

Unity : A powerful visualization software

- Unity is a professional video and game production engine
- Advantage:
  - Professional **3D software**.
  - Provide access to VR or AR.
  - Supports more than 20 platforms.

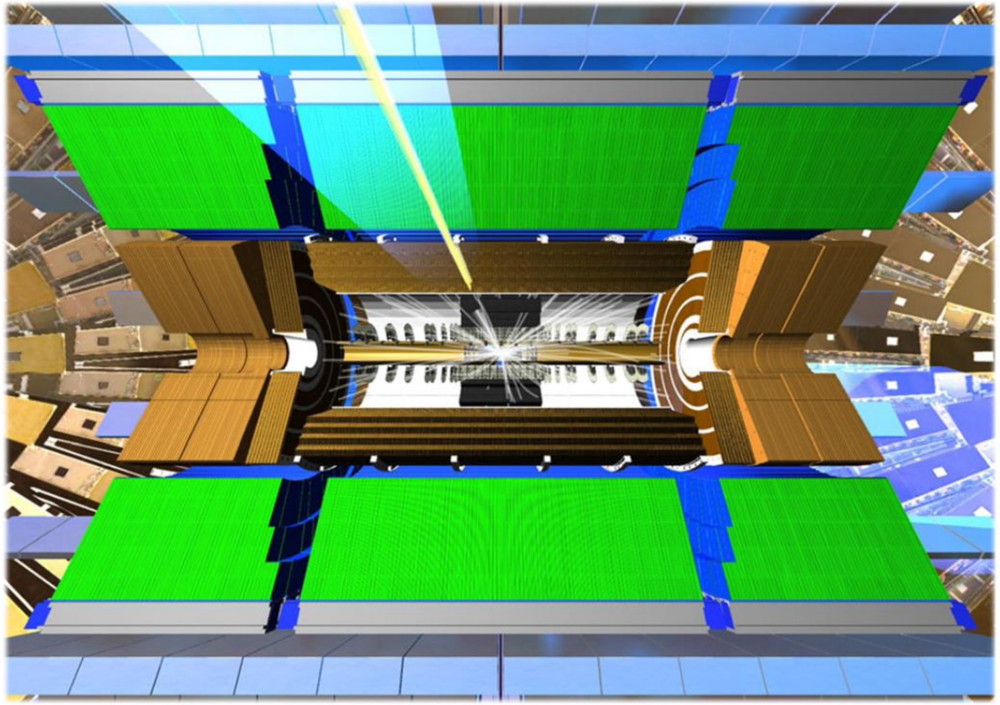


Games developed by unity

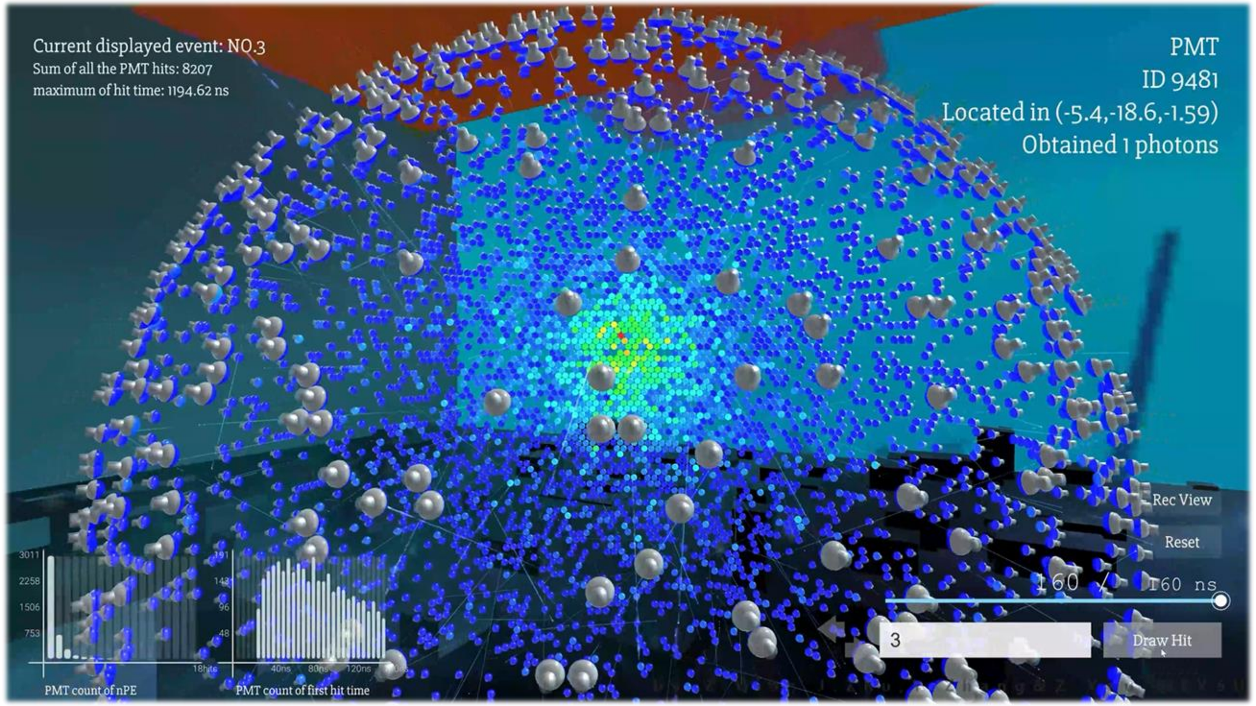
VR

Application of Unity in HEP

ATLAS event display - CAMELIA



JUNO event display - ELAINA

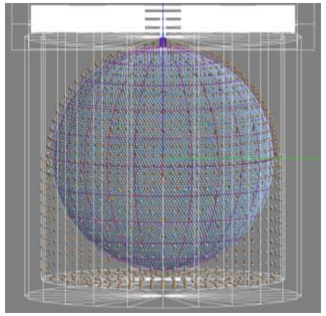


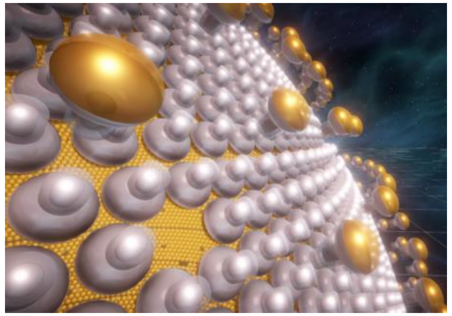
Outline

1. Introduction
2. Methodologies
3. Visualization in Unity
4. Further applications
5. Summary

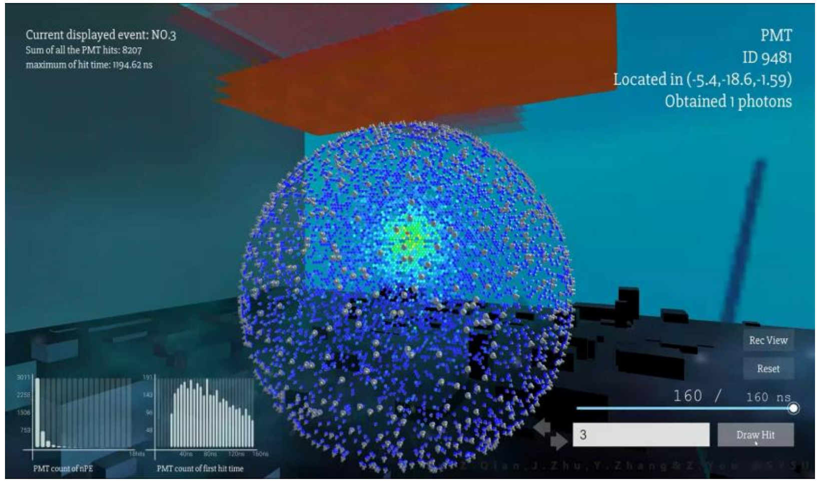
Development based on Unity

- Detector geometry transformation
  - A method to transform different formats of detector description into Unity

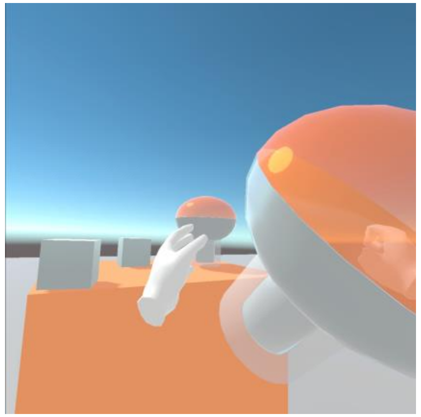




- Application based on Unity
  - Event display, detector optimization, …

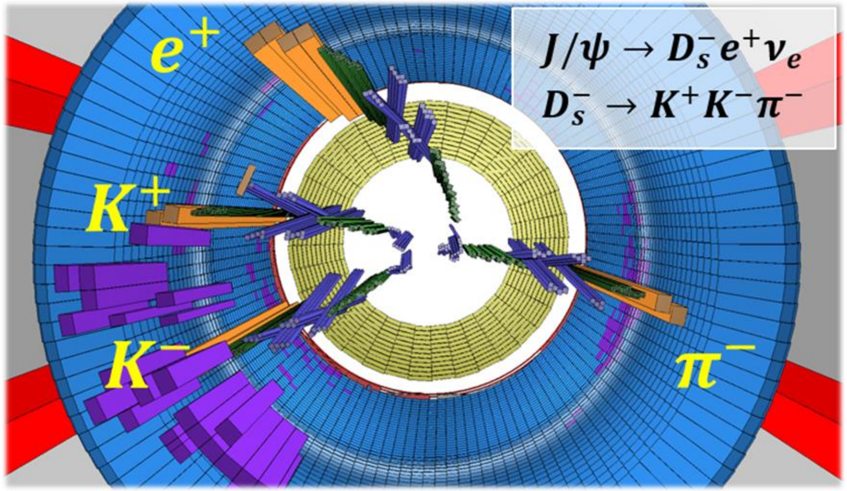


- Extension
  - Virtual Reality, outreach, …



Why do we need to transform detector?

- HEP experiments are usually large-scale scientific apparatuses with complicated detector geometry



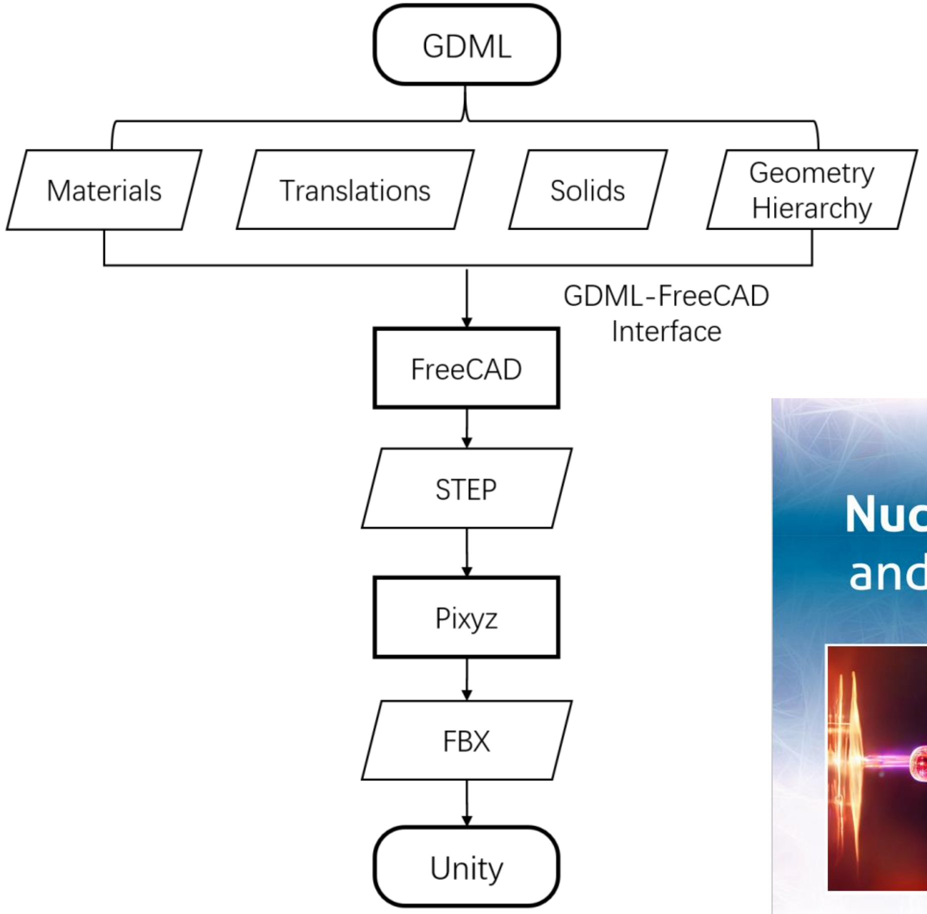
- Different detector description:
  - GDML
  - ROOT
  - DD4hep
  - Geant4

- However, none can be directly imported into Unity

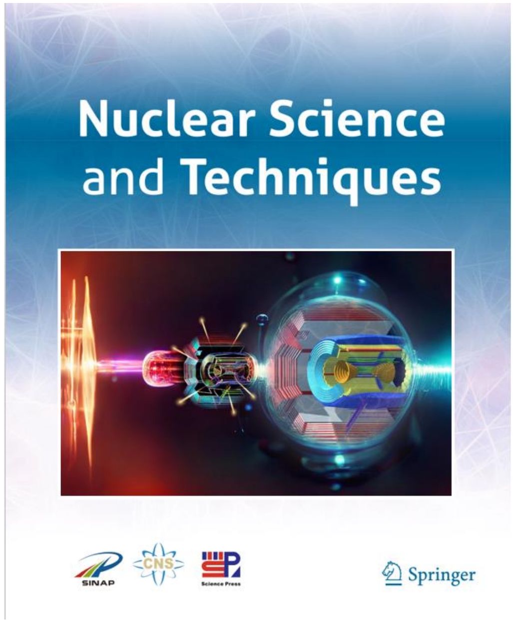
Develop a method for automatic detector transformation!

Works for all detectors, all formats, while keeping consistency

A feasible transformation method

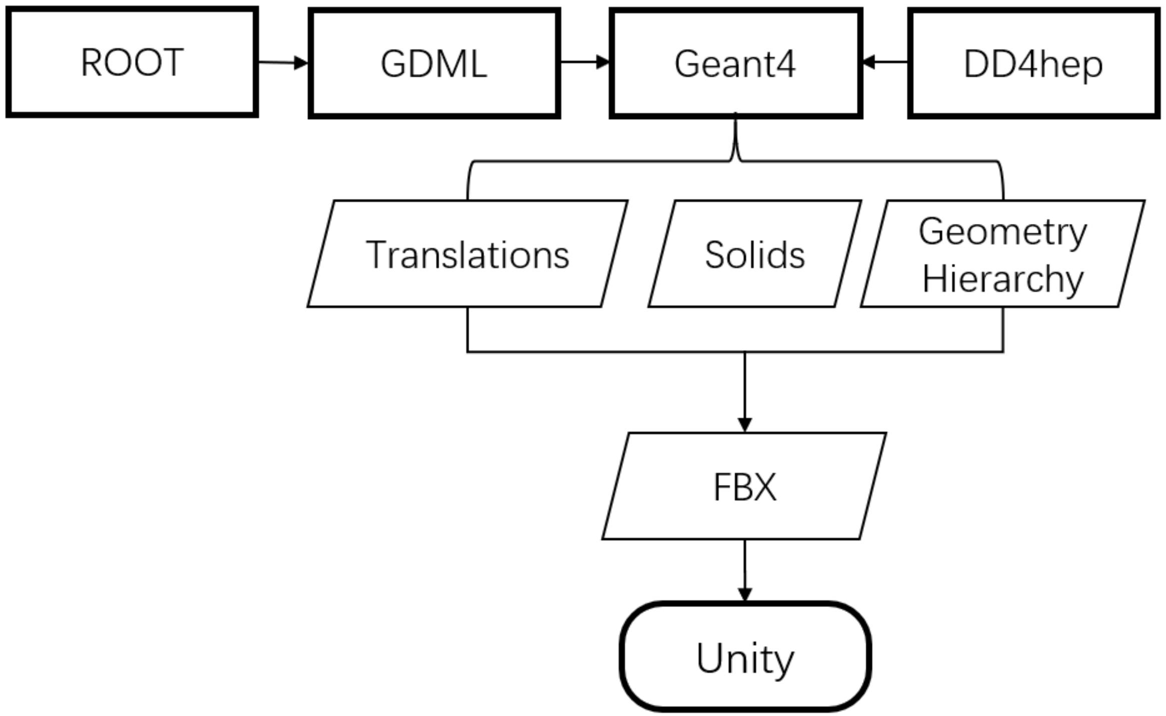


NUCL SCI TECH (2022) 33:142



- Using the GDML–FreeCAD interface, FreeCAD, and Pixyz software.
- Maintain the unique identifier of each detector unit.
- Provides richer visualization properties.
- The method is feasible, however, too complicated and time consuming.

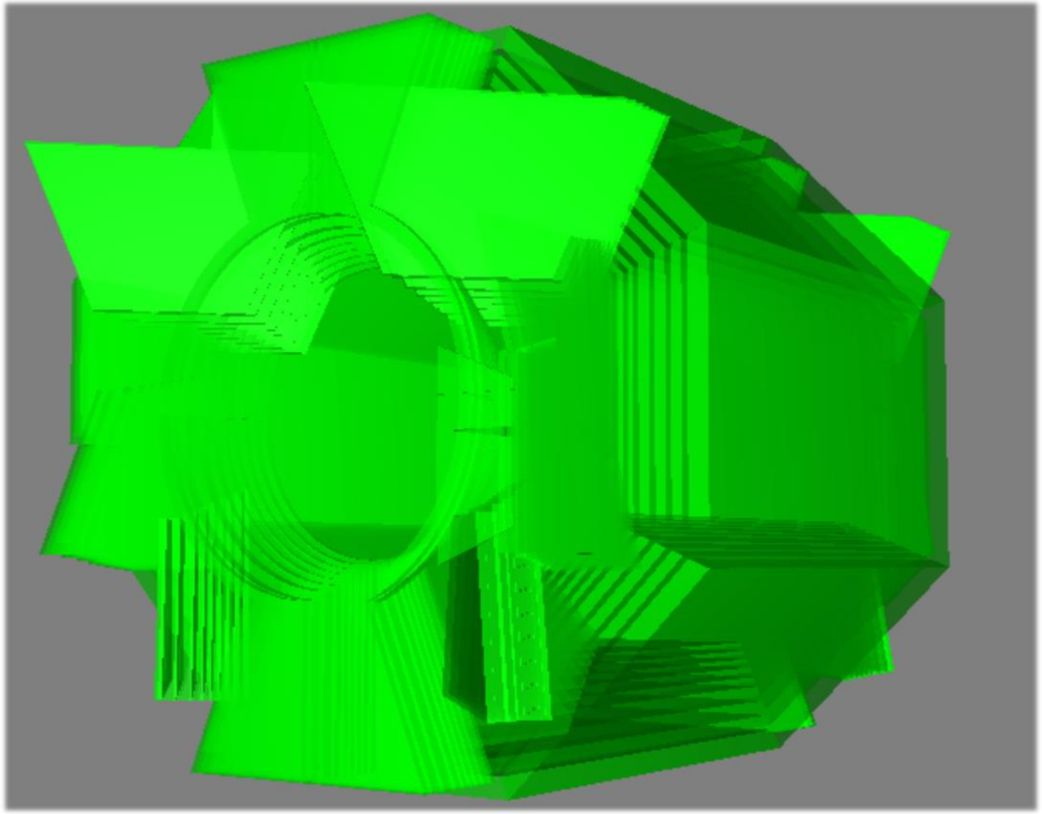
A new method under development



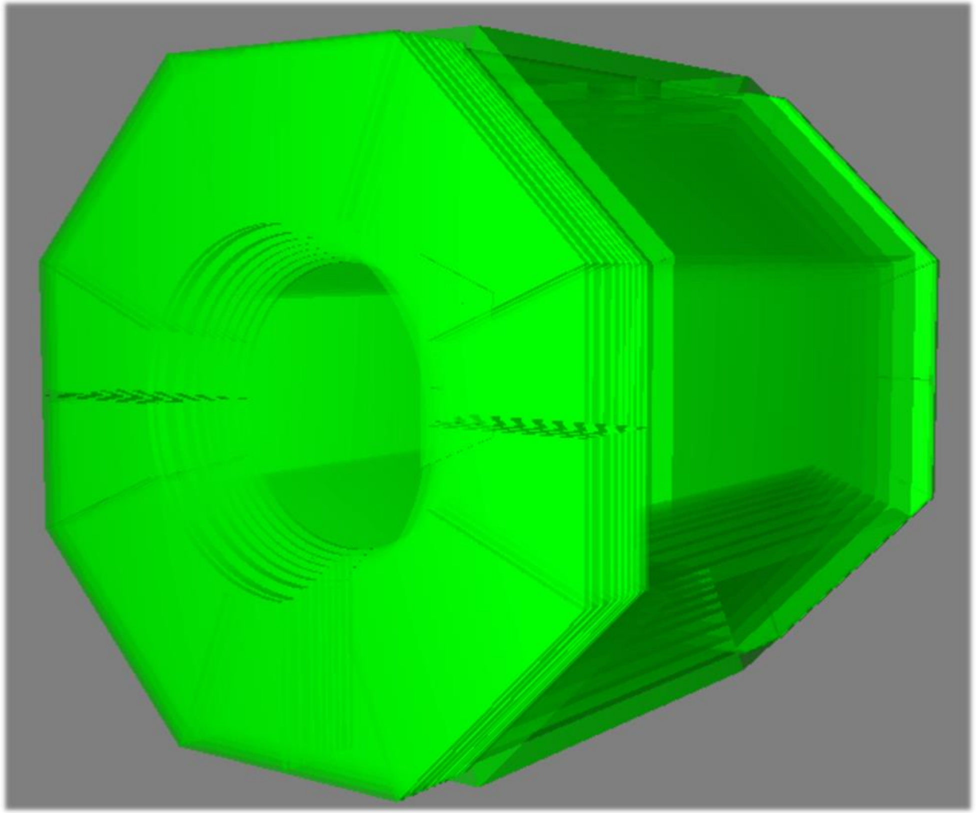
Complete all the transformation processes in one step.

- Develop based on HSF Geometry Writer, which has been used in Belle II
- Convert Geant4 detector description and write it into FBX format
- Update with several features
  - Fine tuning of configuration to solve crash caused by complicated geometry
  - Support self-defined shapes and geometry classes
  - Fix bugs in GDML-G4 transformation

Example of interface update in GDML-G4



Before the modification of Boolean operation



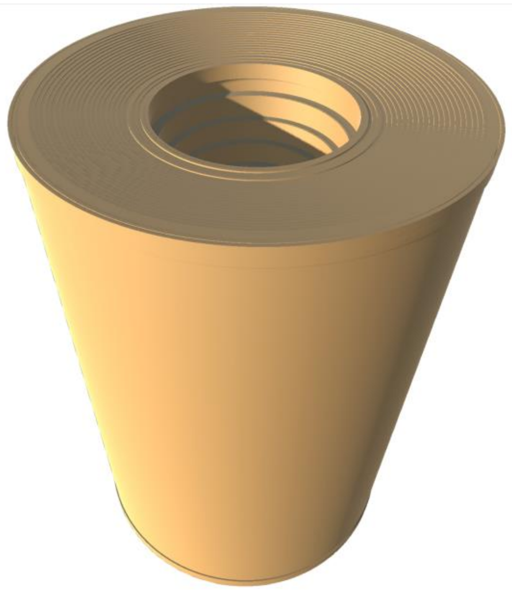
After the modification of Boolean operation

Outline

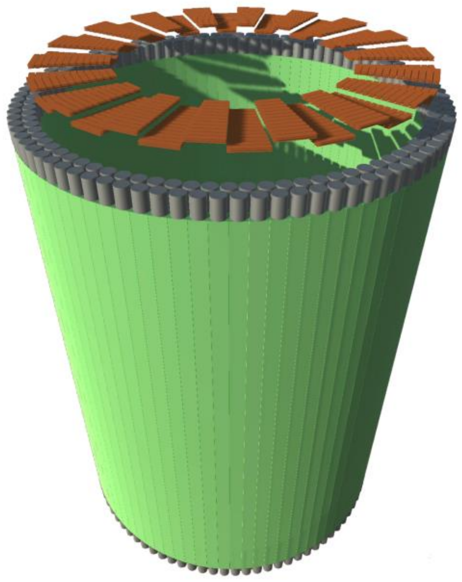
1. Introduction
2. Methodologies
3. Visualization in Unity
4. Further applications
5. Summary

GDML to Unity with BESIII detector

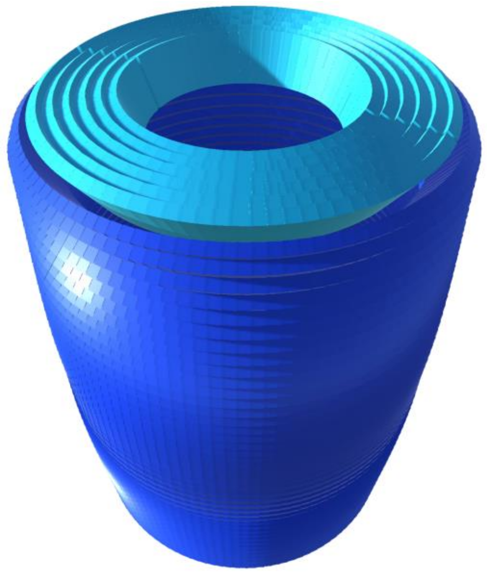
- ➢The BESIII detector description with GDML
- ➢Transformed to FBX and displayed in Unity



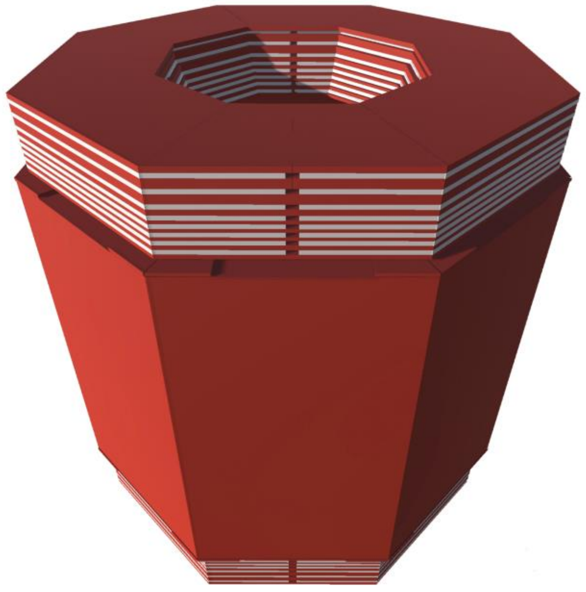
MDC



TOF

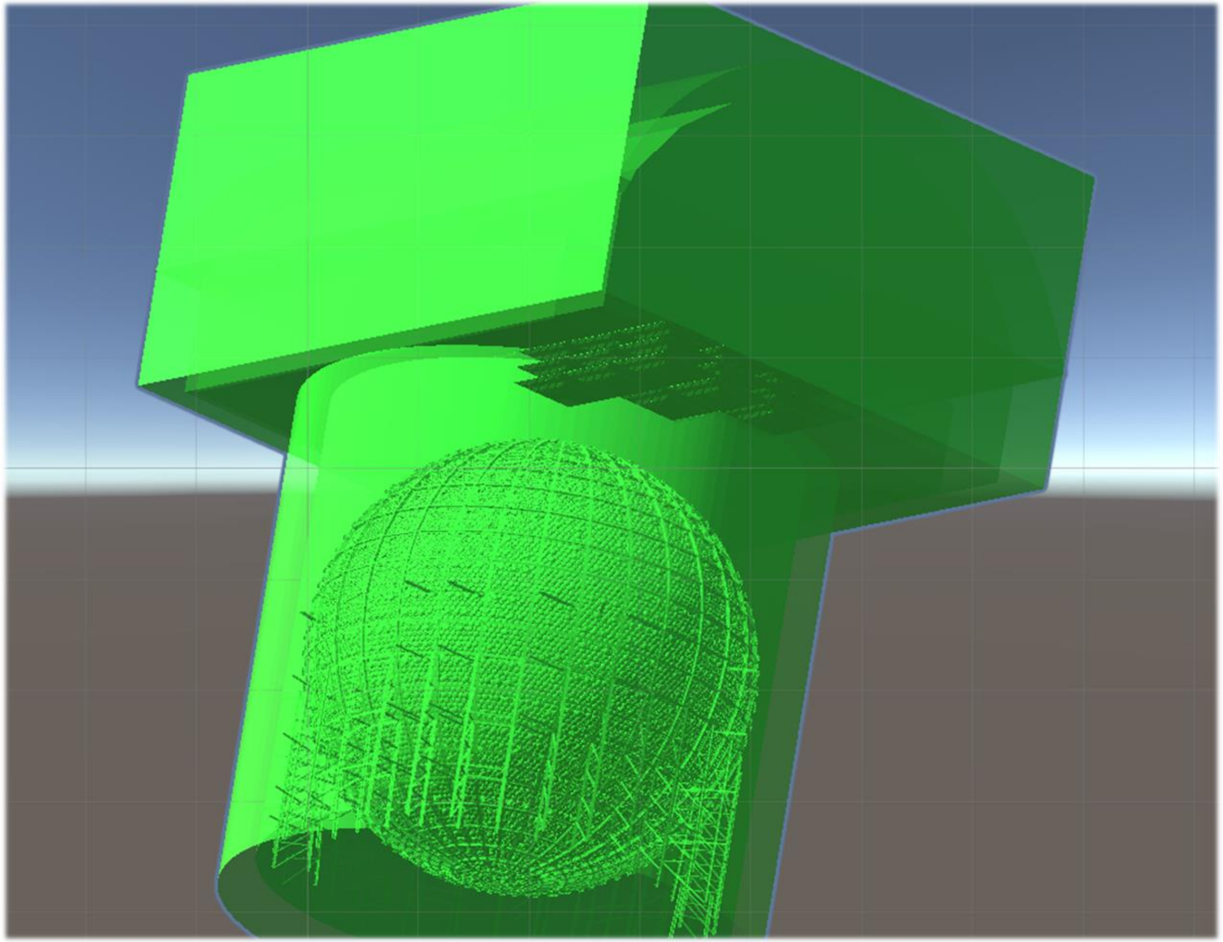


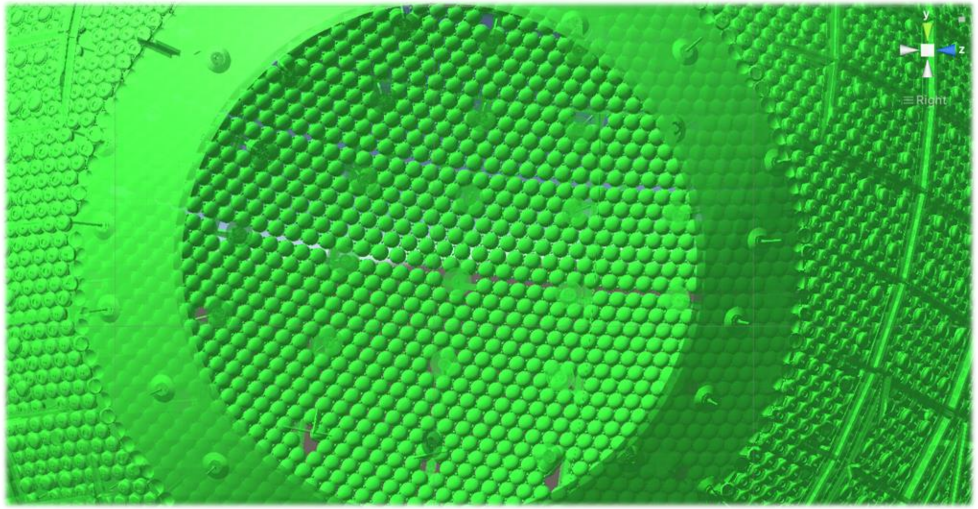
EMC



MUC

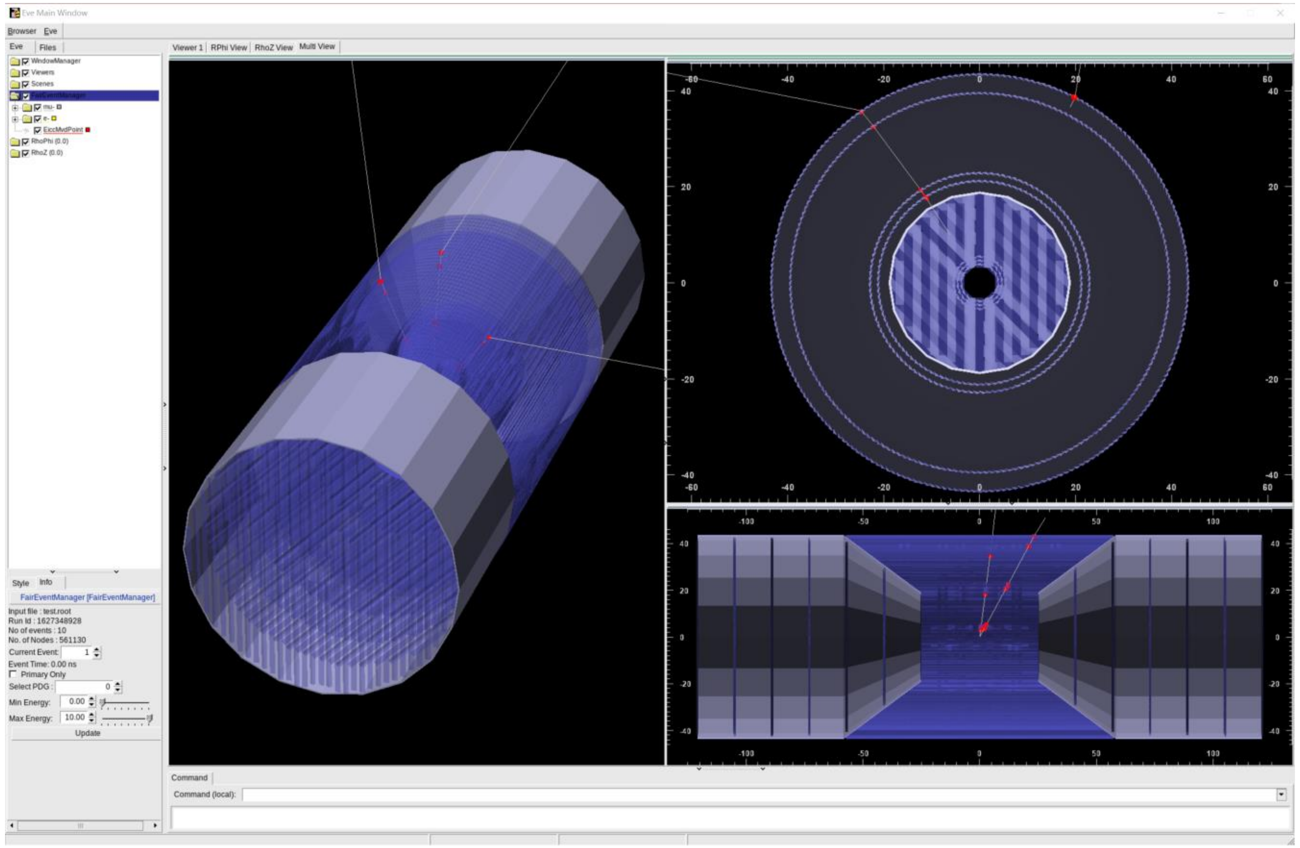
Geant4 to Unity with JUNO detector

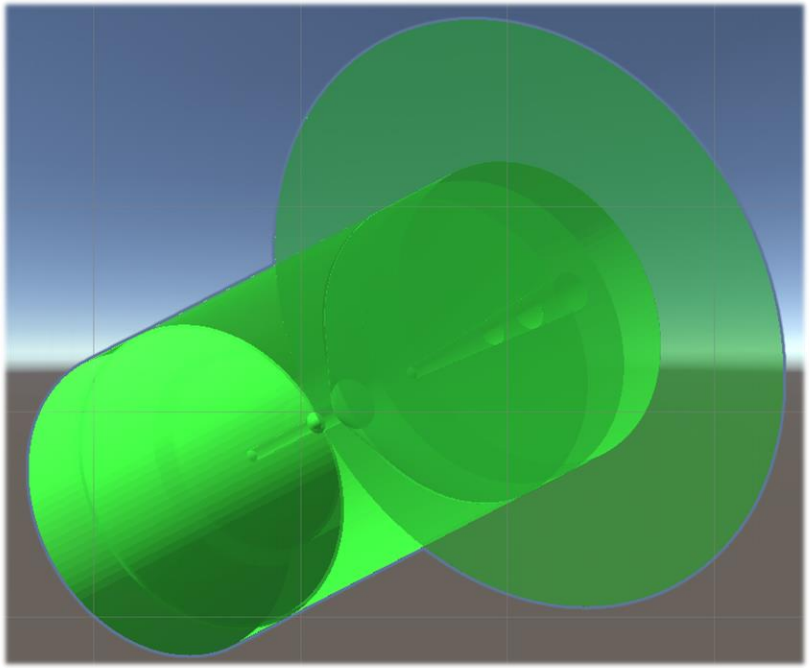




- 3D view of JUNO with FBX transformed from Geant4
- Overall view of JUNO and details in part of Central Detector

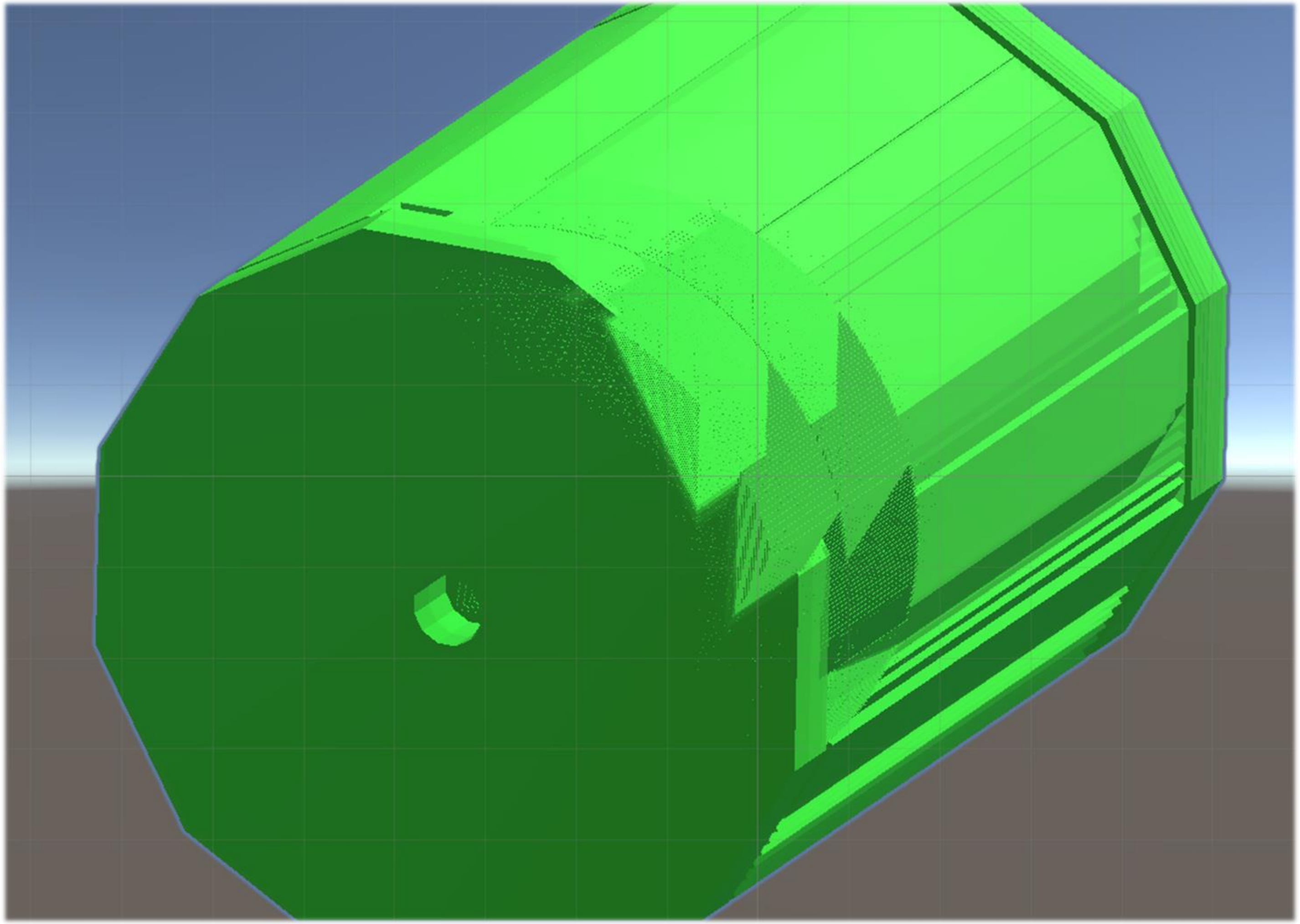
ROOT to Unity with EicC detector





- ➢EicC detector geometry in FairROOT
- ➢Converted to FBX from ROOT

DD4hep to Unity with CEPC

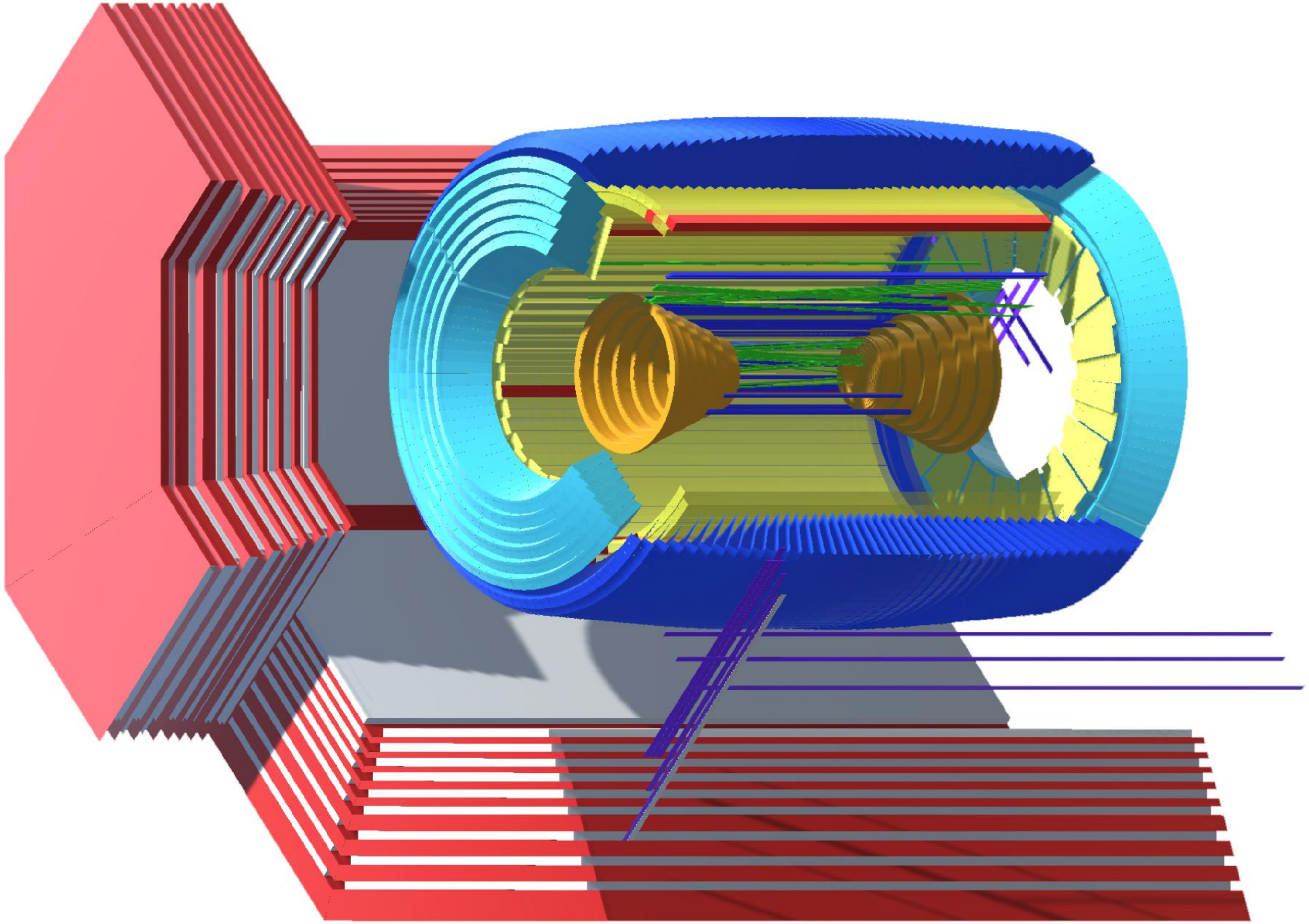


- CEPC detector description based on DD4hep, configured with XML
- Several sub-detectors have been successfully converted into FBX and displayed in Unity
- Still some failure for complex sub-detectors due to naming duplication

Outline

1. Introduction
2. Methodologies
3. Visualization in Unity
4. Further applications
5. Summary

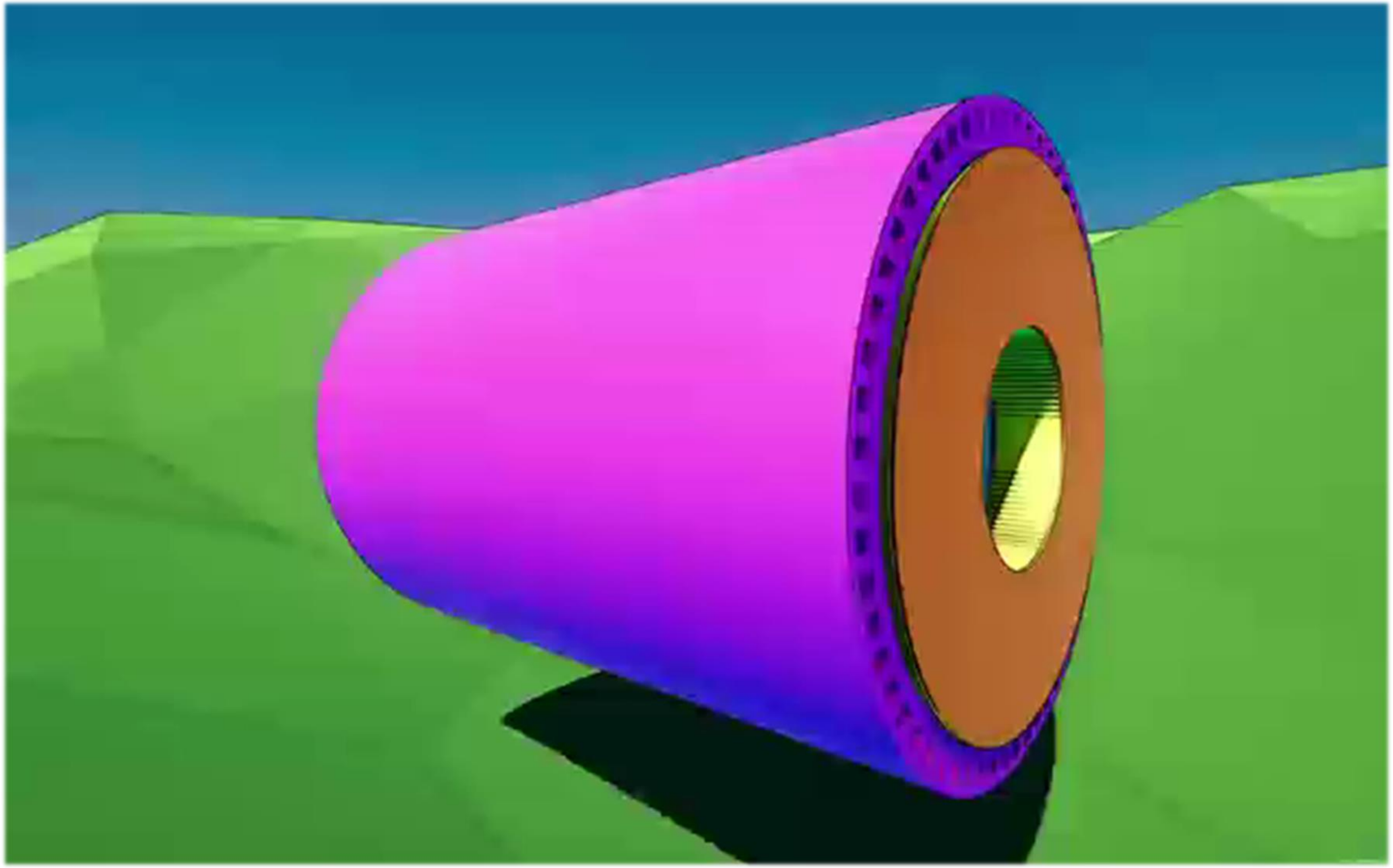
Unity based Event Display for BESIII



NUCL SCI TECH (2022) 33:142

- With the FBX file converted from GDML and implemented in unity.
- A preliminary example of event display for BESIII.
- We expect to get the full functionality for overall BESIII detector and events.

Video for BESIII 3D Event



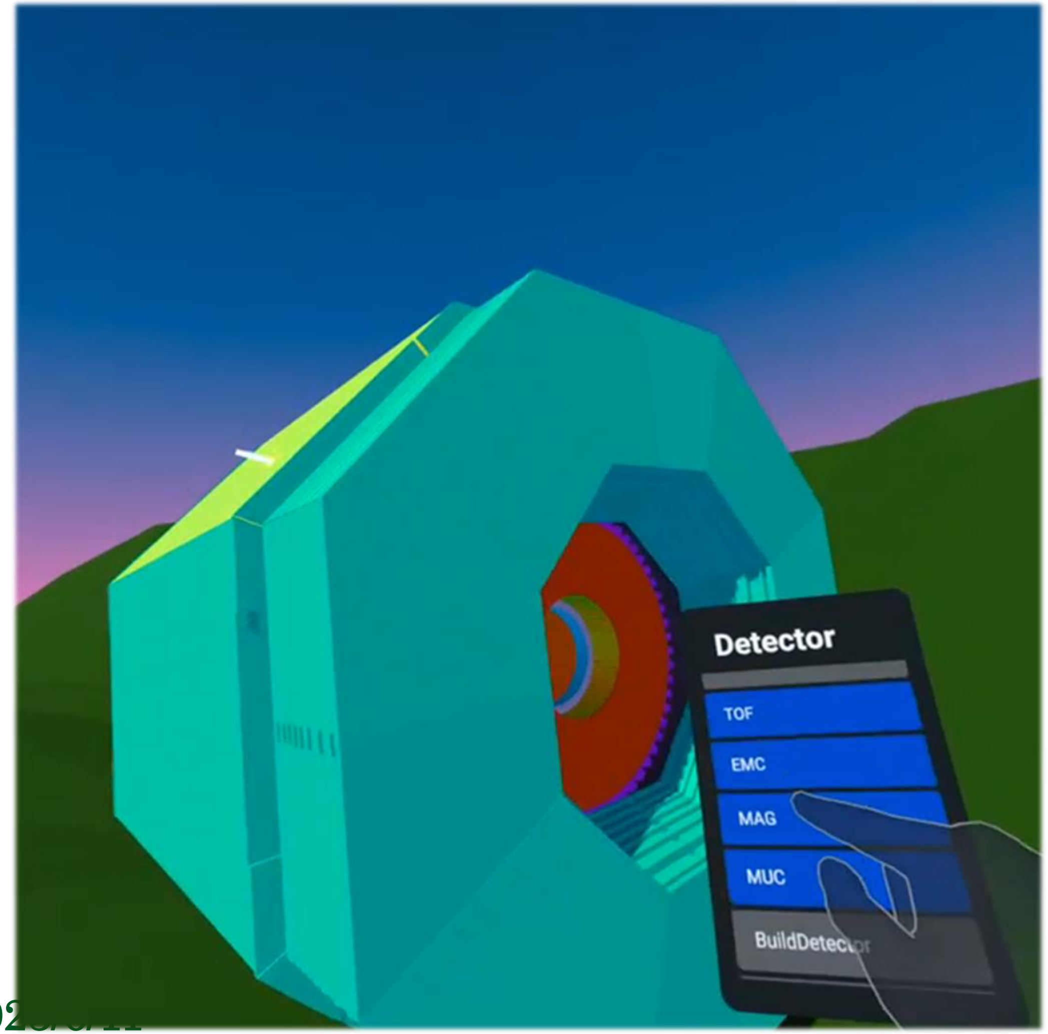
Virtual Reality

- ➢ Platform supports for XR
  - ➢ Virtual Reality
  - ➢ Augmented Reality

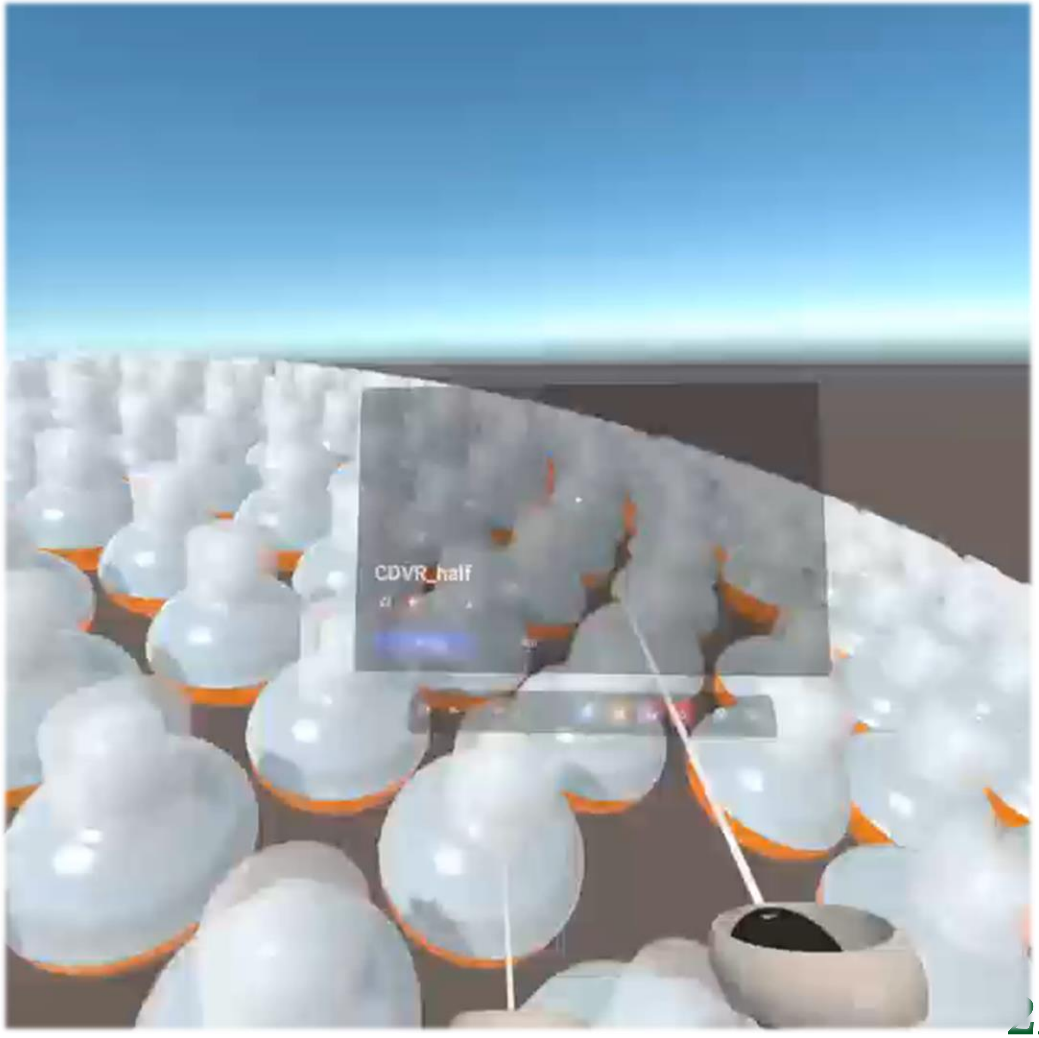
- ➢ Supports for VR/AR devices
  - ➢ HTC Vive
  - ➢ Oculus Quest 2
  - ➢ Apple Vision Pro

VR demo for BESIII & JUNO

VR for BESIII



VR for JUNO



Outline

Summary

- Detector description and visualization play important roles in HEP experiments
- A method for HEP detector geometry conversion to Unity is under development
- Promising future applications with industrial supports

Thank you for listening.

中山大學
SUN YAT-SEN UNIVERSITY