G4VR: A Virtual Reality Extension for GEANT4

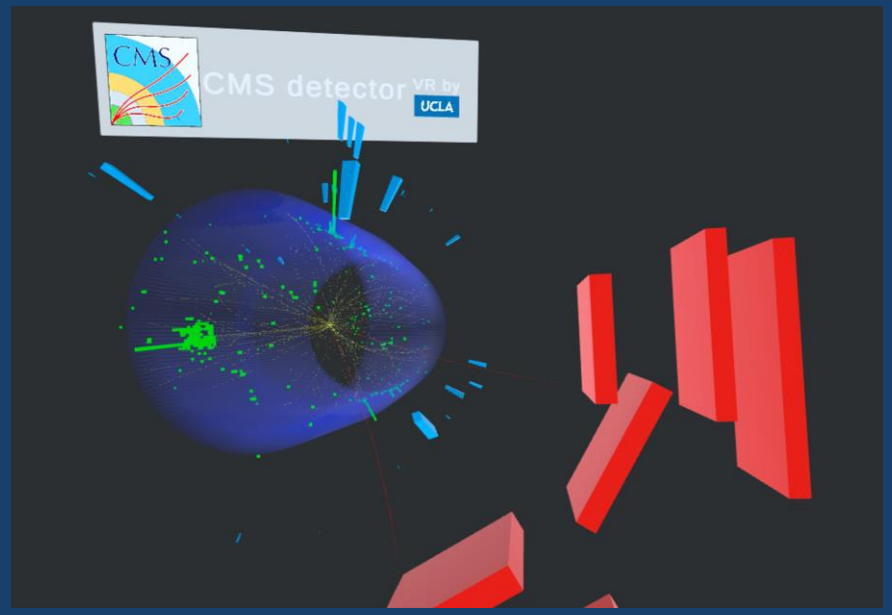
Benjamin Jobilal[1], Stewart Boogert[2], Muhammed Ansar Iqbal[1], Jay Hauser[1]

[1] University of California Los Angeles (UCLA)

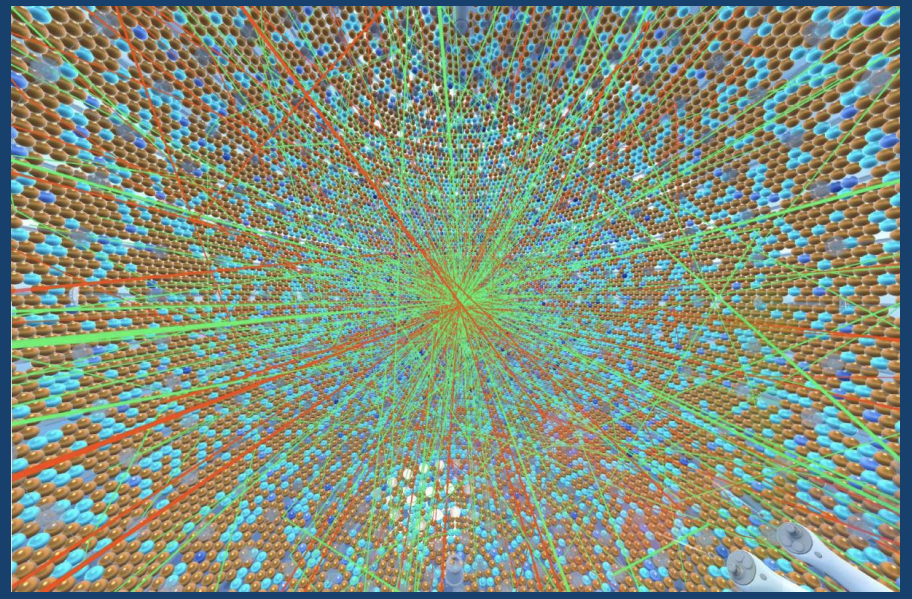
[2] The University of Manchester

Virtual Reality in Science

- Virtual Reality (VR) applications in science allow for a more instinctive and immersive understanding of scientific phenomena.
  - Some efforts have already been made in this direction with the Belle II and JUNO experiments [1][2].
  - The UCLA Physics VR lab has produced a range of applications in science, most notably VR SPY for CMS [3].
- VR devices such as the Oculus Quest Series and Apple Vision Pro are scaling → potential for scientific applications!



VR SPY [3]



JUNO VR [2]

The VR Toolkit

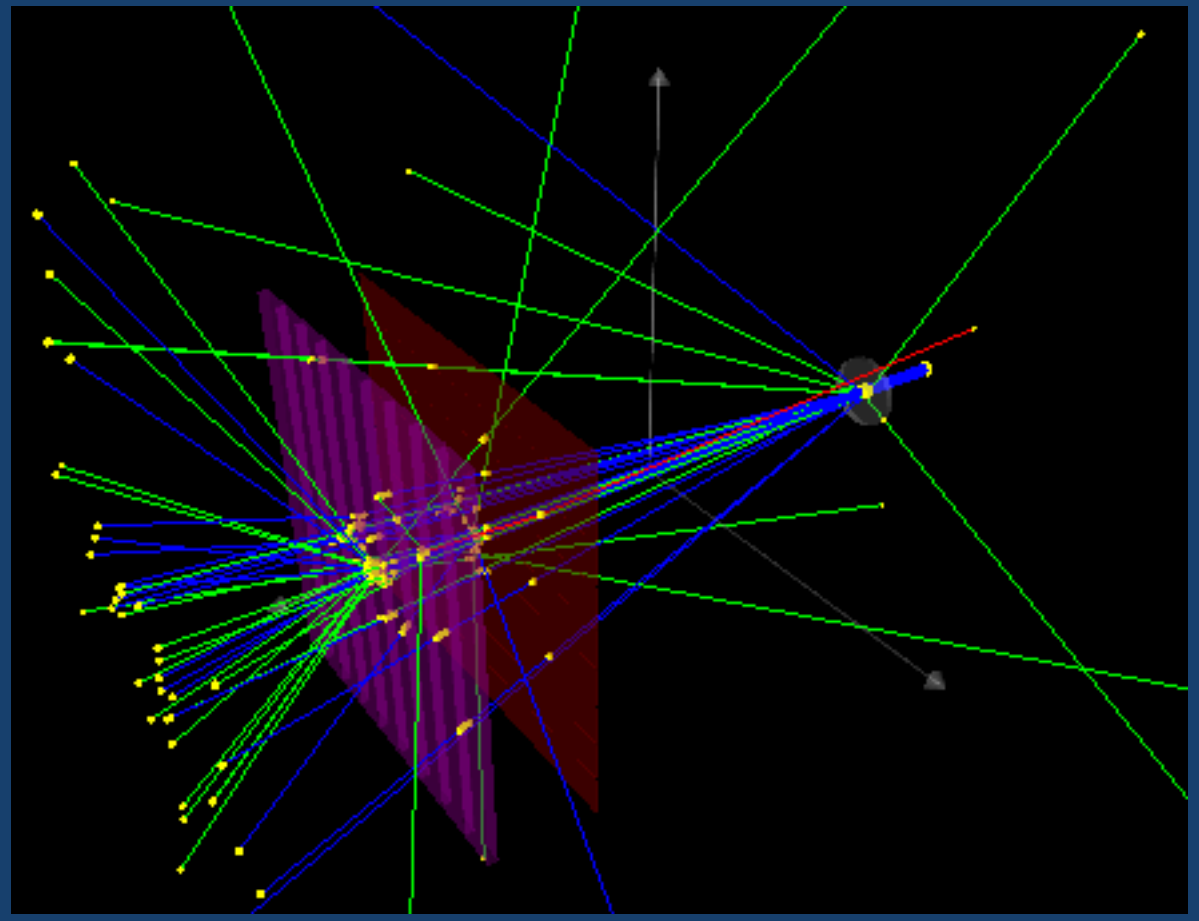
- The UCLA Physics VR Lab develops VR apps (APKs) in Unity – a game development platform with VR features.
  - We use Meta Oculus integration packages to integrate VR features.
- To use any APK, the user must have an Oculus headset (Quest 2+) and controllers.

Oculus Quest 3

Oculus Quest 2

A Short Introduction to GEANT4

- A widely used Monte-Carlo package to simulate the passage of particles through matter [4].
- It has a wide range of applications in science, medicine, engineering, etc. For example, all LHC experiments use GEANT4!
- Users can define their own geometries, radiation sources, materials, simulate them, and examine the results of simulations with in-built GUI and data analysis tools.



From GEANT4 to G4VR

- The UCLA Physics VR lab has, over the last year, developed **G4VR – a GEANT4 VR event display**.
- G4VR can be run on any Oculus-based headset and produces the GEANT4 scene given the experiment geometries and tracks (assets).
- **Idea**:
  I. Allow users to run a standard GEANT4 experiment with a novel XR driver.
  II. XR driver interfaces with the VR headset to allow asset transfer.
  III. Data is received by G4VR and processed to produce the scene.

The implementation of each step is discussed in the following slides.

I. XR Driver

- Facilitate data transfer to VR → develop a new vis driver for GEANT4.
  - Named “XR” to accommodate expansion to AR in the future.
- The driver performs three functions:
  1. **Collects experiment geometries** → writes them to GLB binaries (fast, resource efficient 3D asset format).
  2. **Collects track data (steps, positions, times, energies, etc.)** by accessing scene trajectories → writes them to CSV.
  3. **Launches a local server** → assets are posted here, and G4VR collects them from this address.

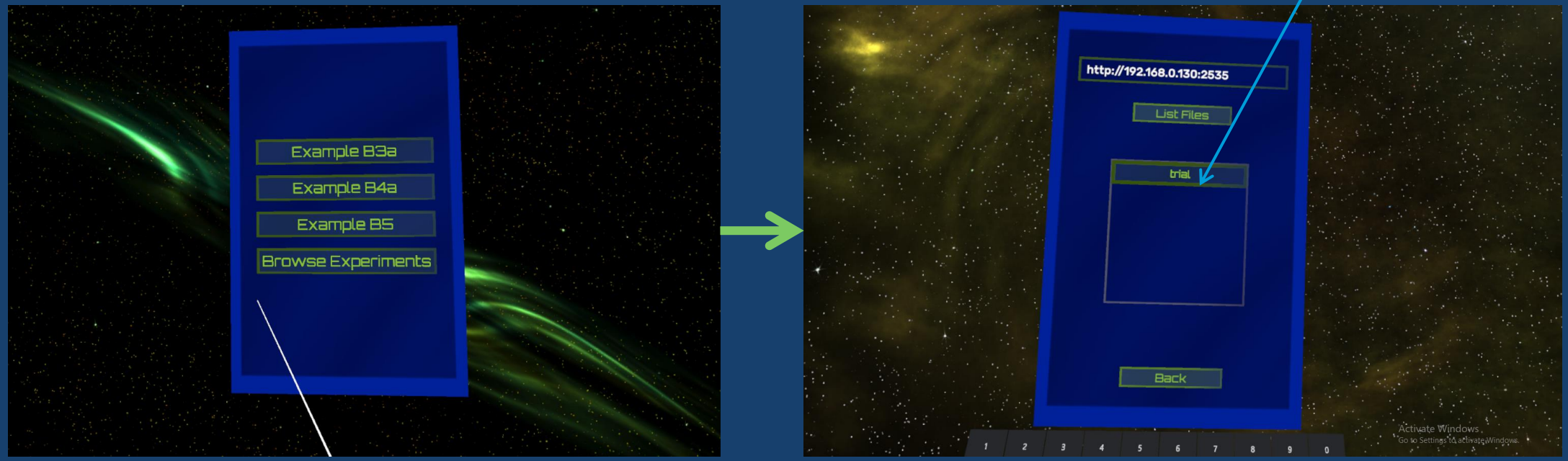
I. XR Driver

- From the user's point of view, the use of the driver is simplified:
  - Execute the GEANT4 program.
  - Enter /vis/open Xr in the GEANT4 console.
  - Run the desired number of runs.
- The server is set up to store the data of each run separately so that the user can switch between the results of each run in VR.

II. Asset Transfer

- During runtime, G4VR is set up to read from a specific input URL and import the experiment data from that address.
  - A UI is enabled that the user can interact with to enter the address and select the experiment.

Experiment named “trial” can be loaded



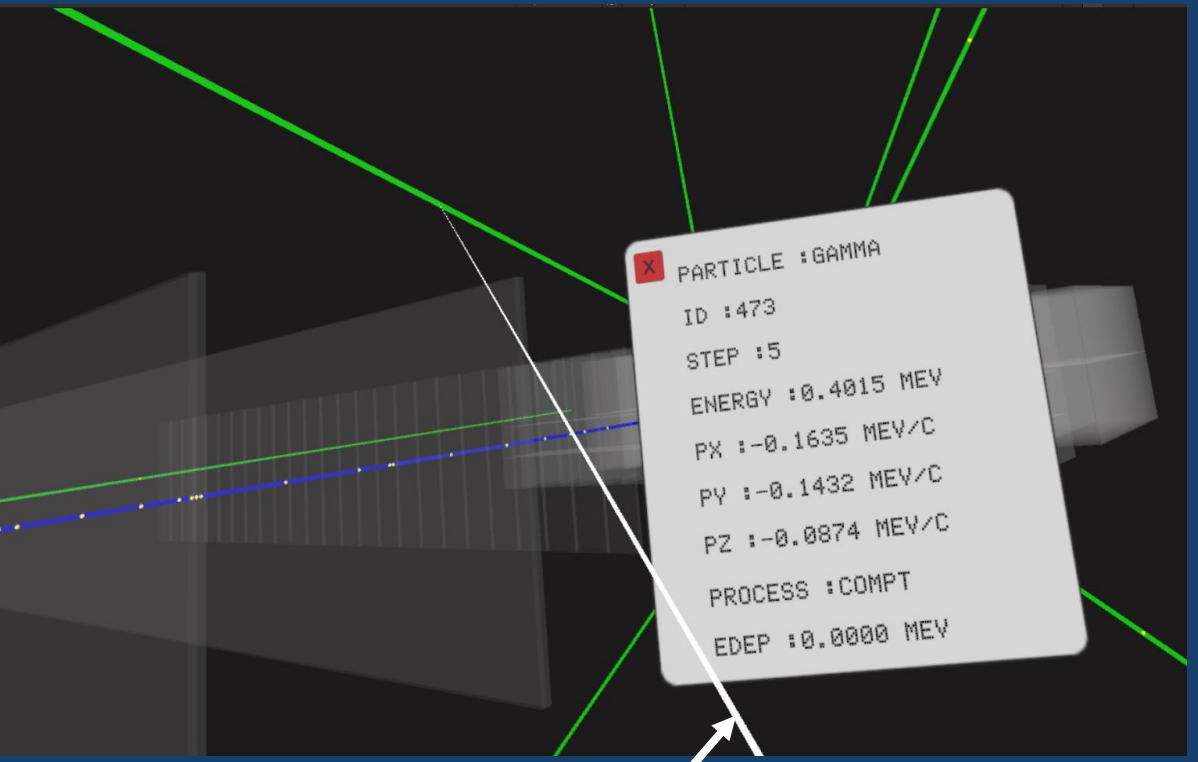
III. Event Display

- From the CSV → reconstruct the tracks.
- From the GLB → instantiate the geometries.
  - G4VR is developed using **Unity**, which supports GLB parsing to GameObjects (Unity scene objects).

- The event display has a range of functionalities useful for analysis:
  1. Track Analysis
  2. Energy Deposit Analysis
  3. Time Evolution

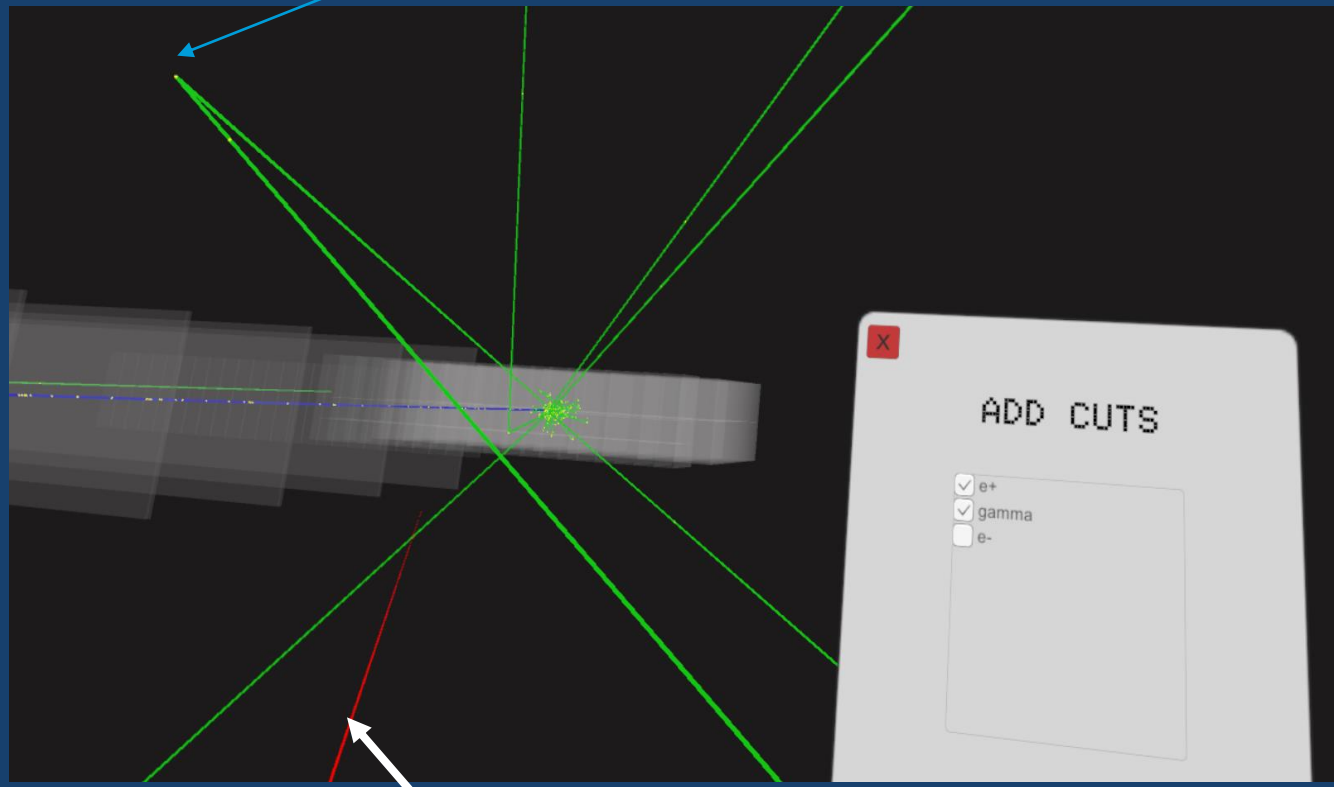
III. Event Display – Track Analysis

- Interaction with tracks to get track data
- Employ filters/cuts on specific particles



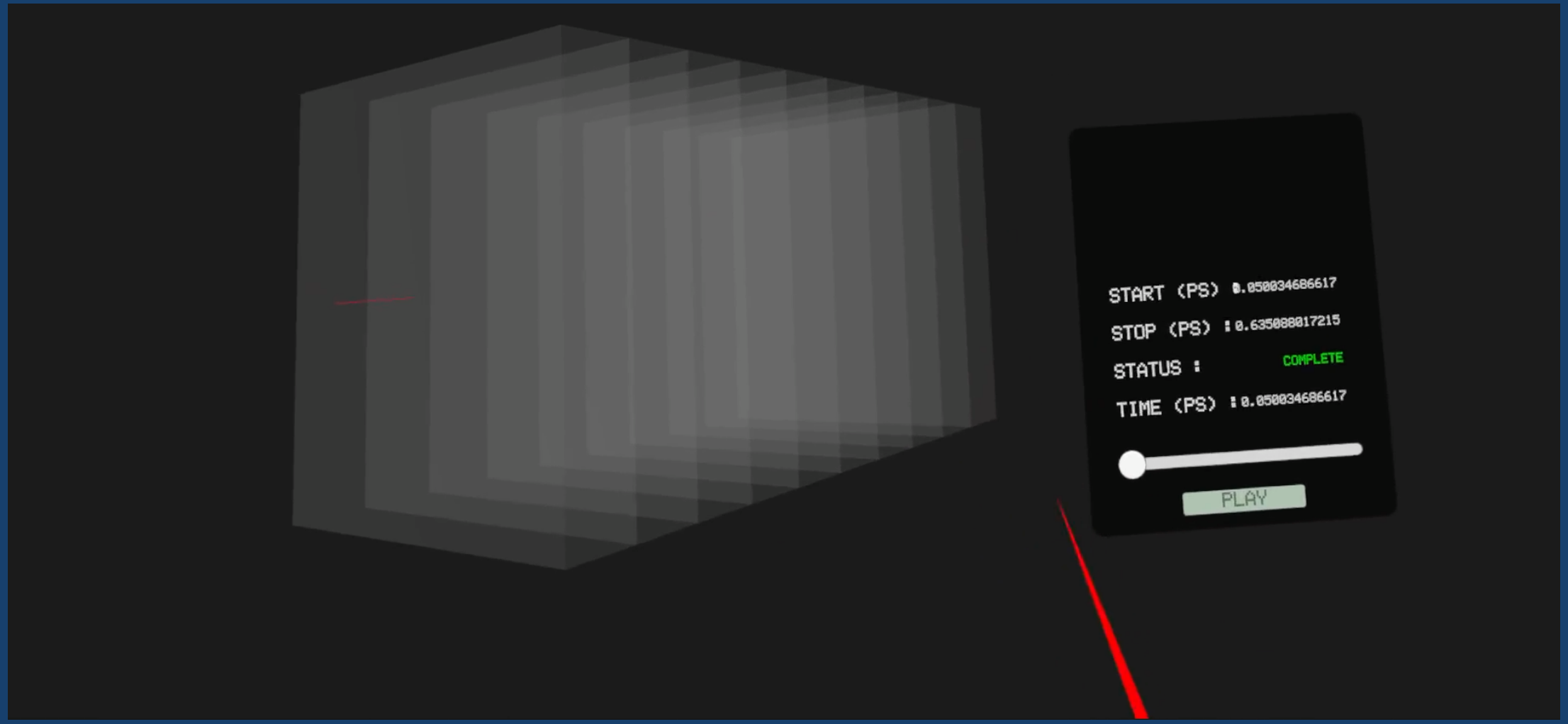
Controller ray

One would otherwise see an electron here

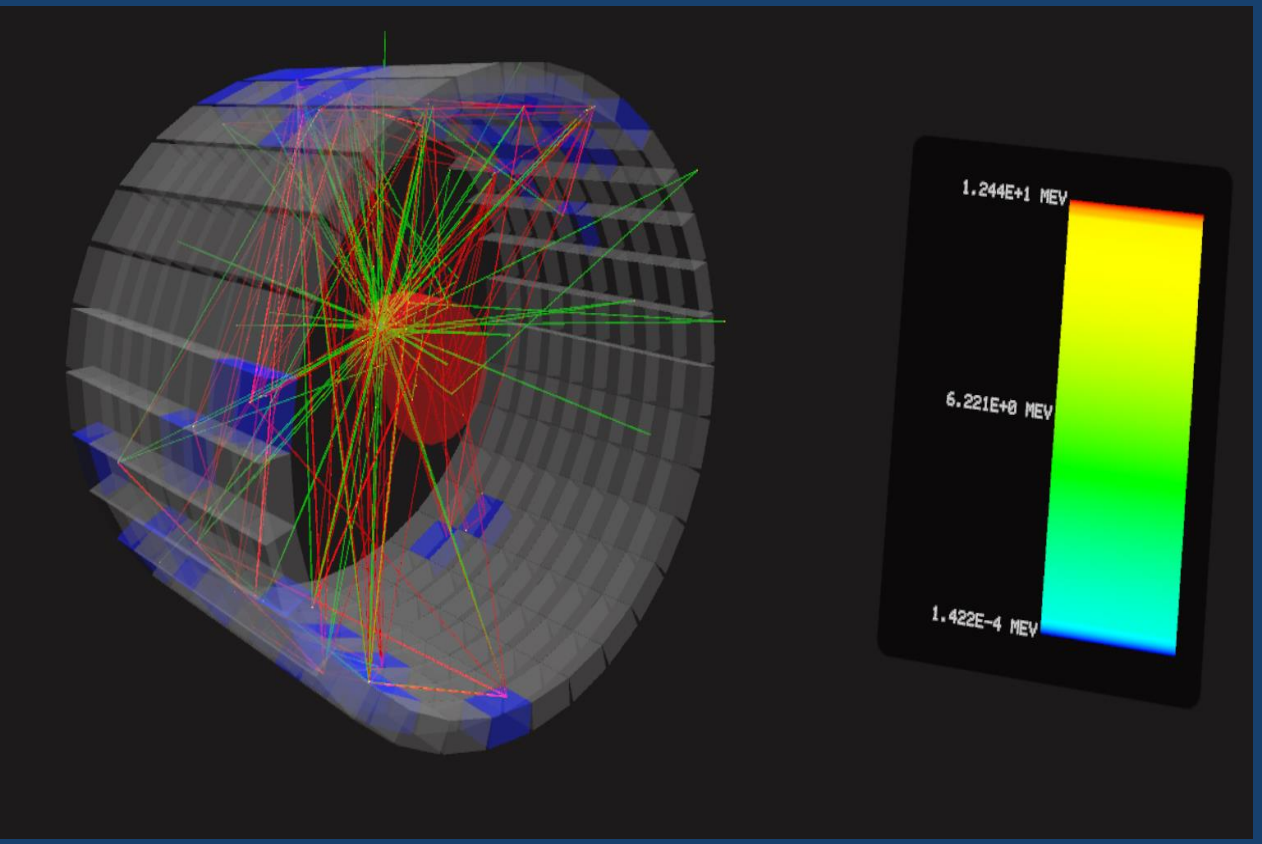


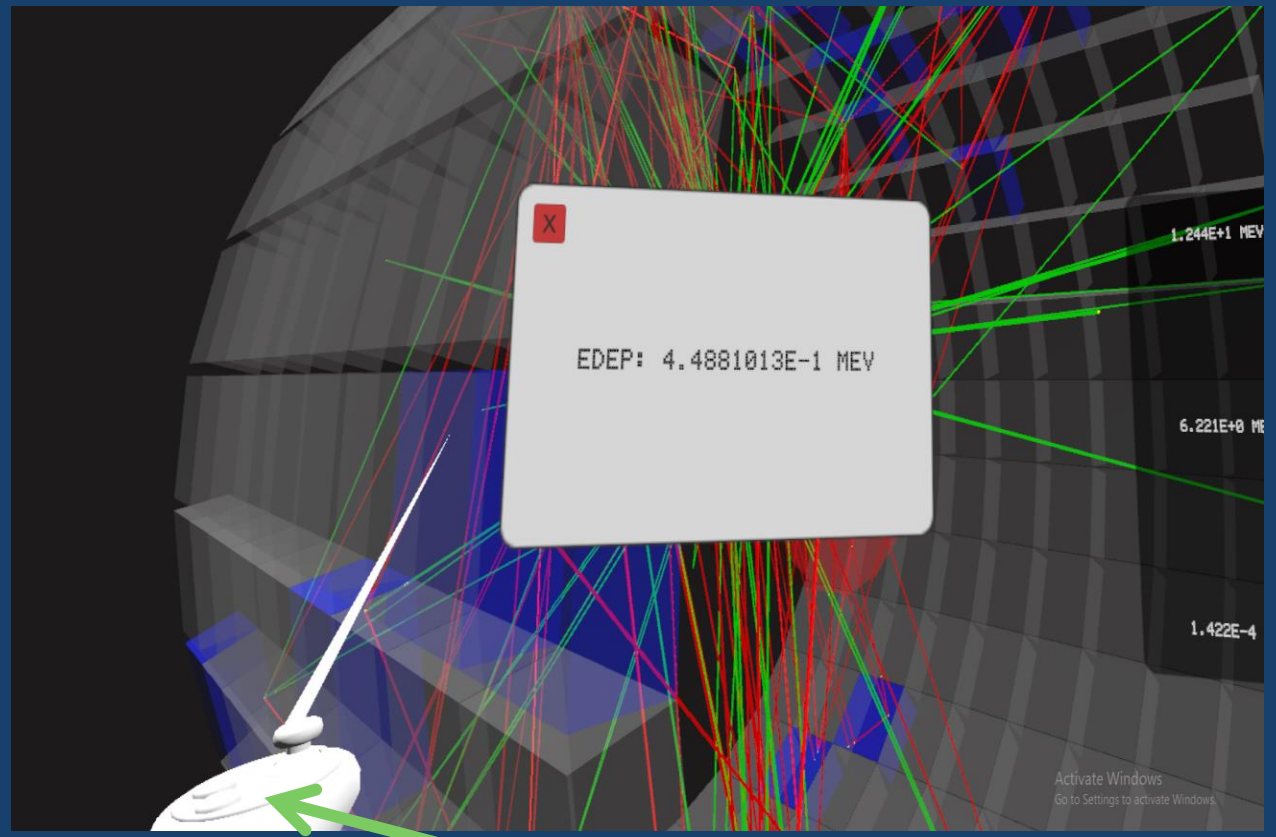
Controller ray

III. Event Display – Time Control



III. Event Display – Edep Analysis

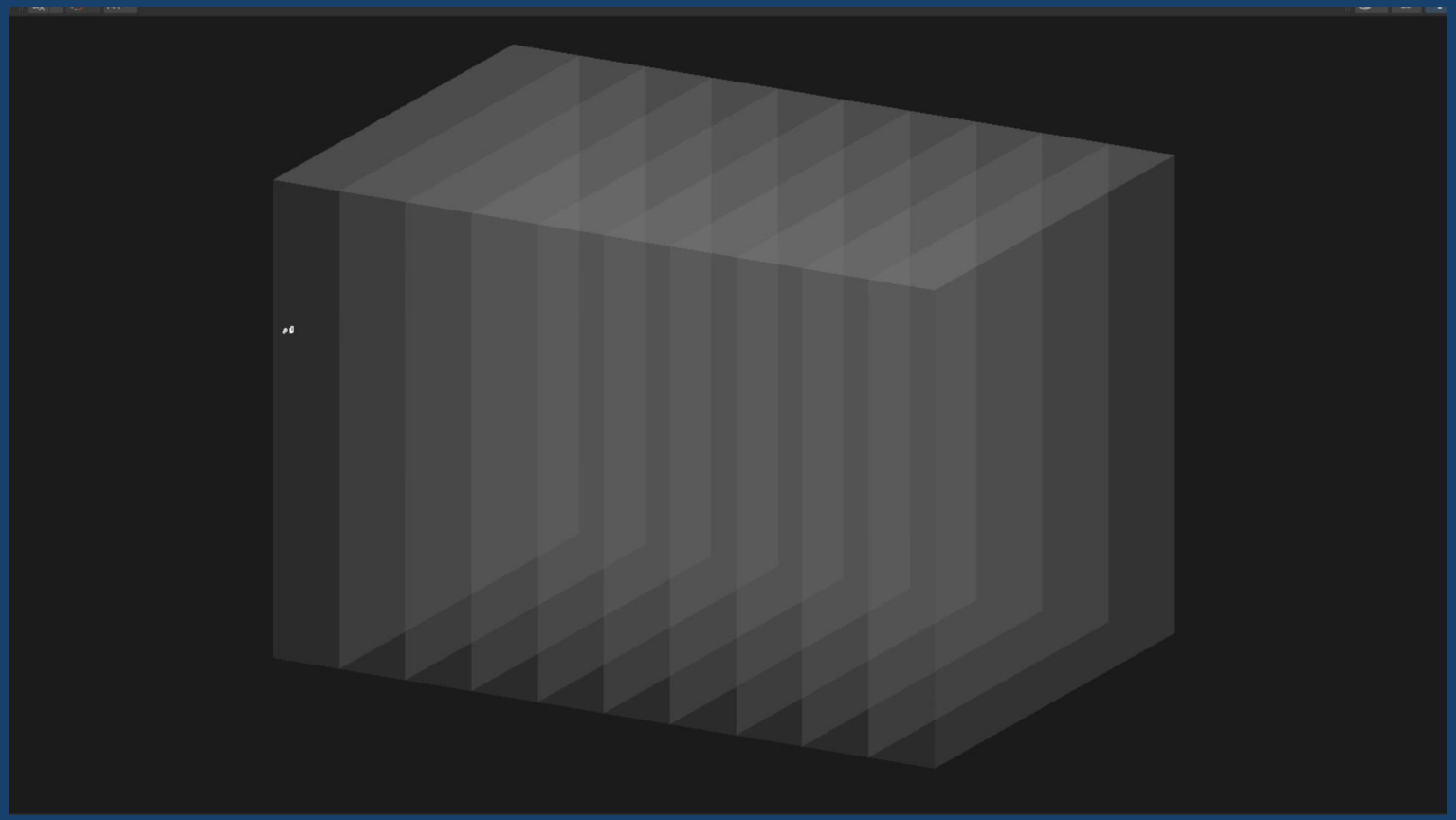




Controller selecting the crystal

III. Event Display - Movies

- An interesting feature still in development is a replication of GEANT4 movies. Below example of this in G4VR outside of the headset.



Looking Forward

- An alpha release of G4VR is set to be out early November 2025. It will contain three example scenes as well as a custom scene.
  - The custom scene can only be used once the XR driver is fully deployed with GEANT4. It currently sits on a dev branch.
  - The APK will be launched on the UCLA Physics VR website [3].
- The XR driver is limited to hosting local servers. In the future, we hope to expand this to remote servers for more flexible use.
- We hope that you can test the driver and VR app to give us feedback and suggestions!

References

[1] Piilonen, Leo, et al. “Belle2VR: An Interactive Virtual Reality Visualization of GEANT4 Event Histories.” *EPJ Web of Conferences*, vol. 245, Jan. 2020, p. 08003. https://doi.org/10.1051/epjconf/202024508003.

[2] Huang, Kai-Xuan, et al. “Unity Based Virtual Reality for Detector and Event Visualization in JUNO Experiment.” *arXiv (Cornell University)*, Sept. 2025, https://doi.org/10.48550/arxiv.2509.16285.

[3] UCLA Physics VR Lab. *UCLA Physics VR*. https://vr.physics.ucla.edu/

[4] S. Agostinelli et al., Nucl. Instrum. Meth. A 506 (2003) 250-303