

EIC Event Display

Phoenix based event display for EPIC

Dmitry Romanov

Event display requirements

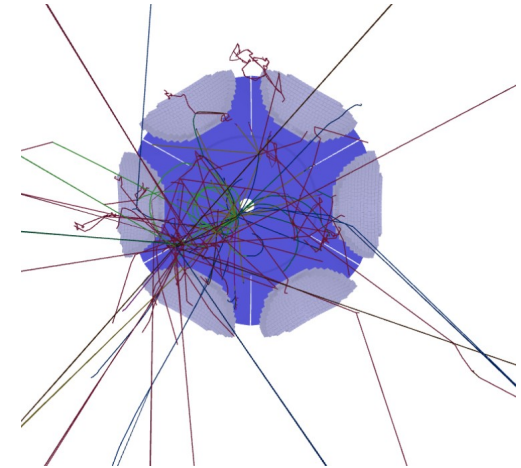
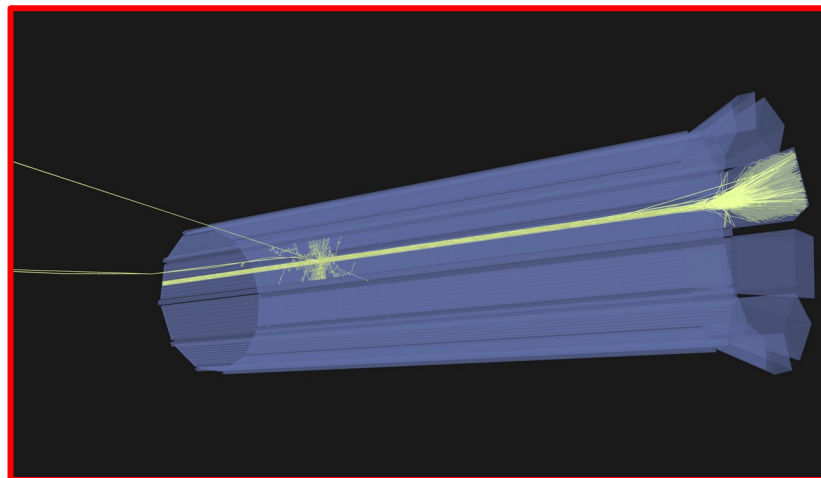
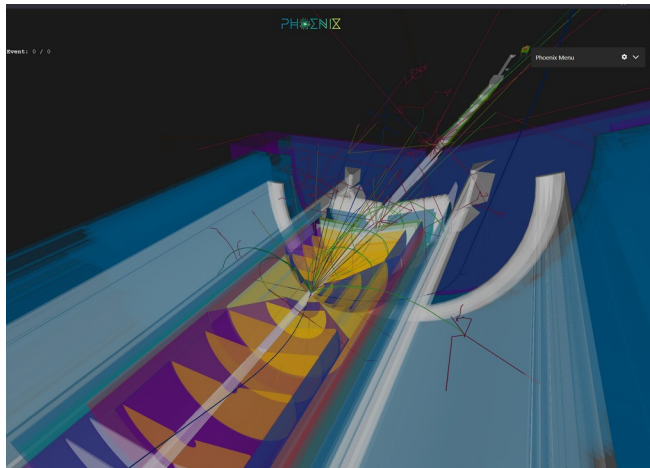
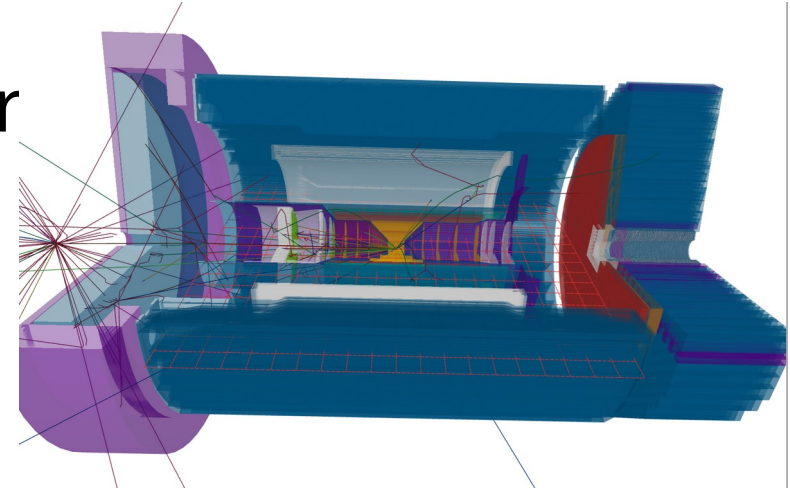
[Requirements for event display](#)

Two primary purposes:

1. Outreach through enhanced visualization of detector and physics data
2. (!) **Detailed debugging of simulation and reconstruction**

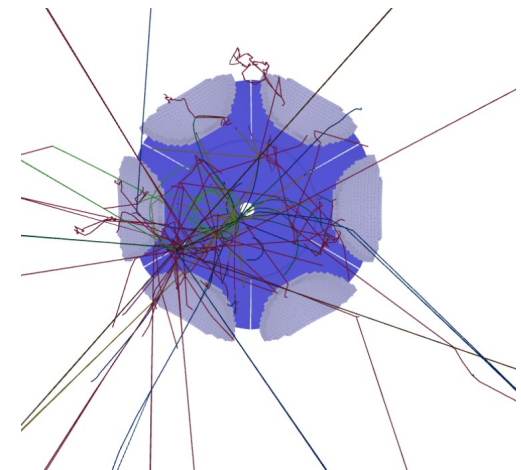
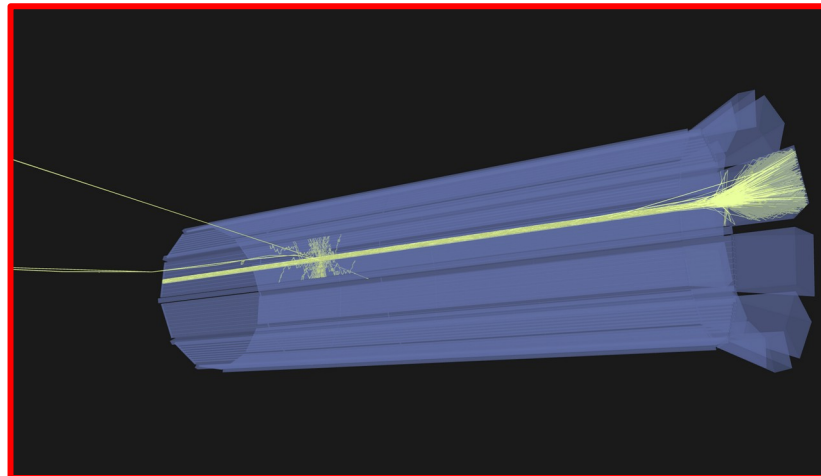
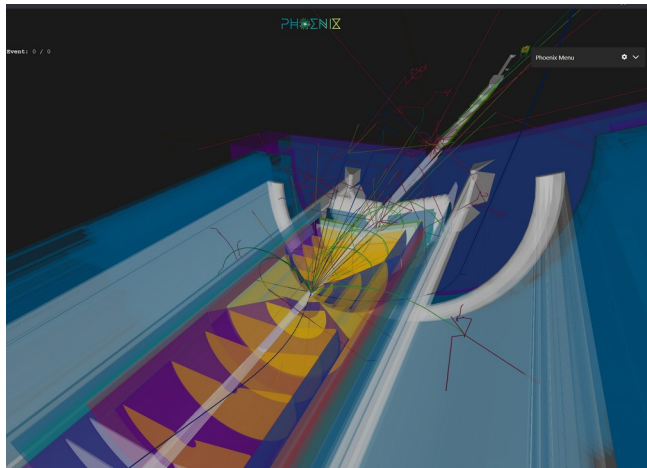
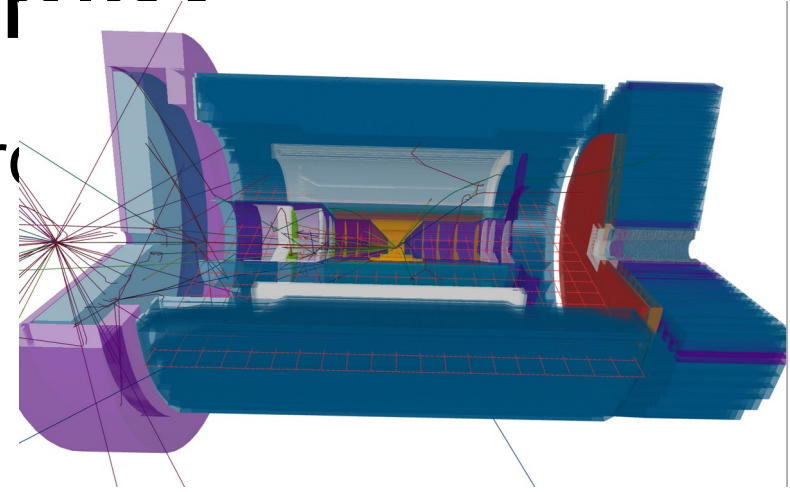
What is there so far

- DD4Hep plugin saving tracks to phoenix for
- Hits in EDM4Hep (json, + root in work)
- Load current geometries
- Can handle far-forward
- Geometry optimization (being in progress)



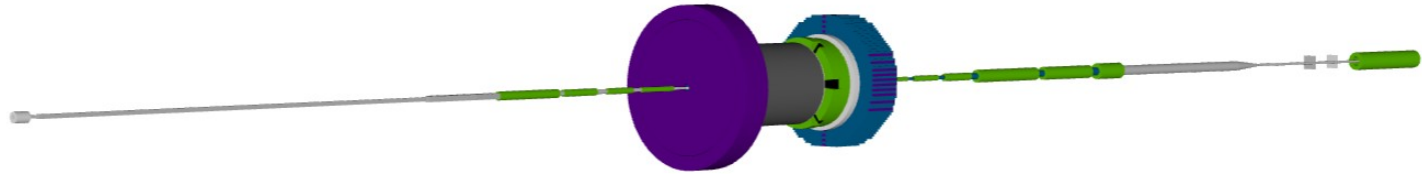
EIC Phoenix based event display

- Together with HSF Phoenix event display group
- <https://github.com/eic/firebird>
- Static website using GH pages
- <https://eic.github.io/firebird>



Geometry time!

Geometry



Task Manager

Type a name, publisher, or PID to search

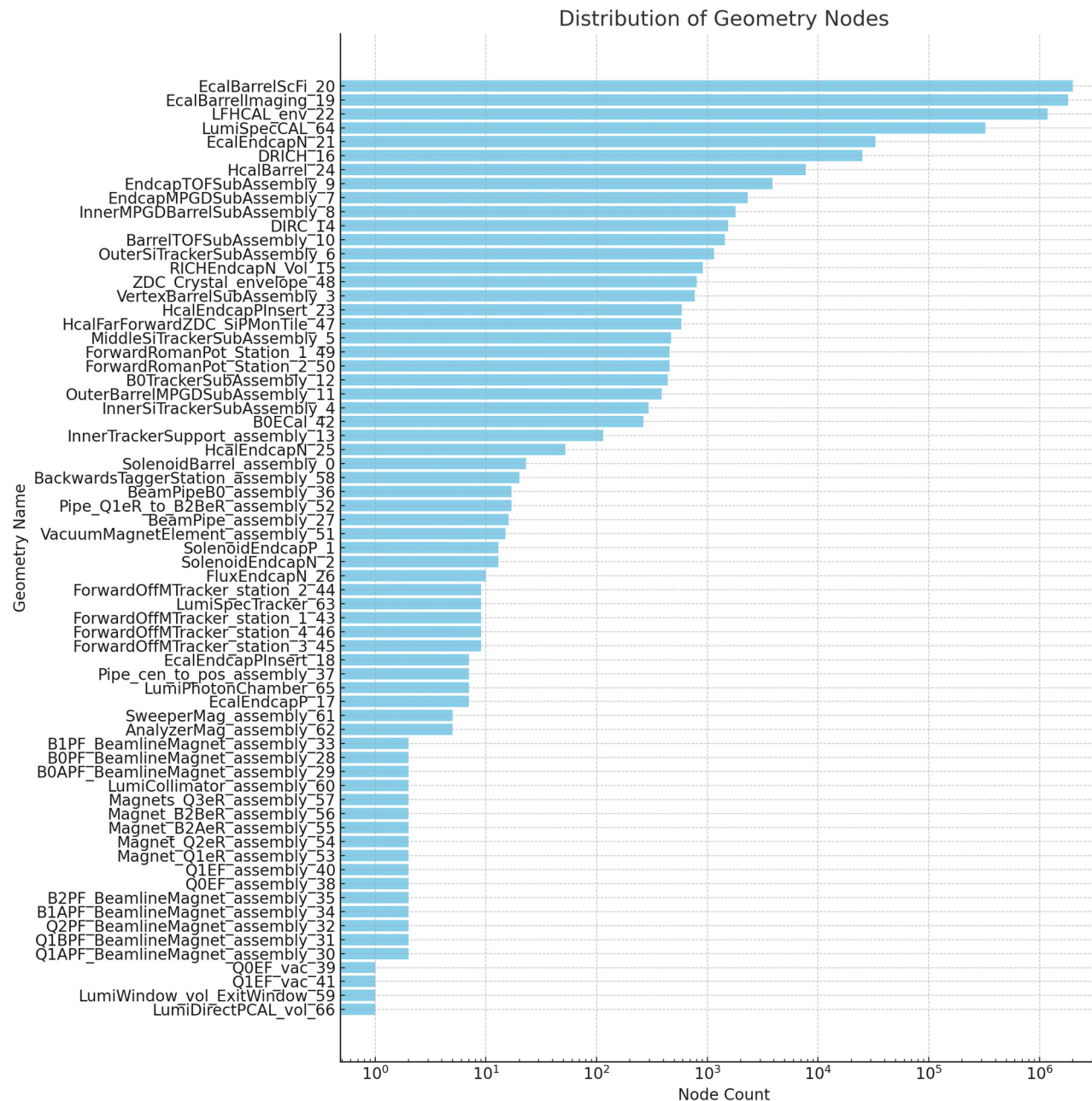
Processes

Name	Status	PID	CPU	Memory
> Vivaldi (63)			15.3%	20,574.8 MB
> WebStorm (15)			0.1%	5,287.1 MB
> Google Chrome (8)			6.3%	4,434.8 MB
Node.js JavaScript Runtime		9340	0%	2,851.3 MB
> PyCharm (13)			0%	2,069.0 MB
VmmemWSL		43080	0.1%	891.5 MB



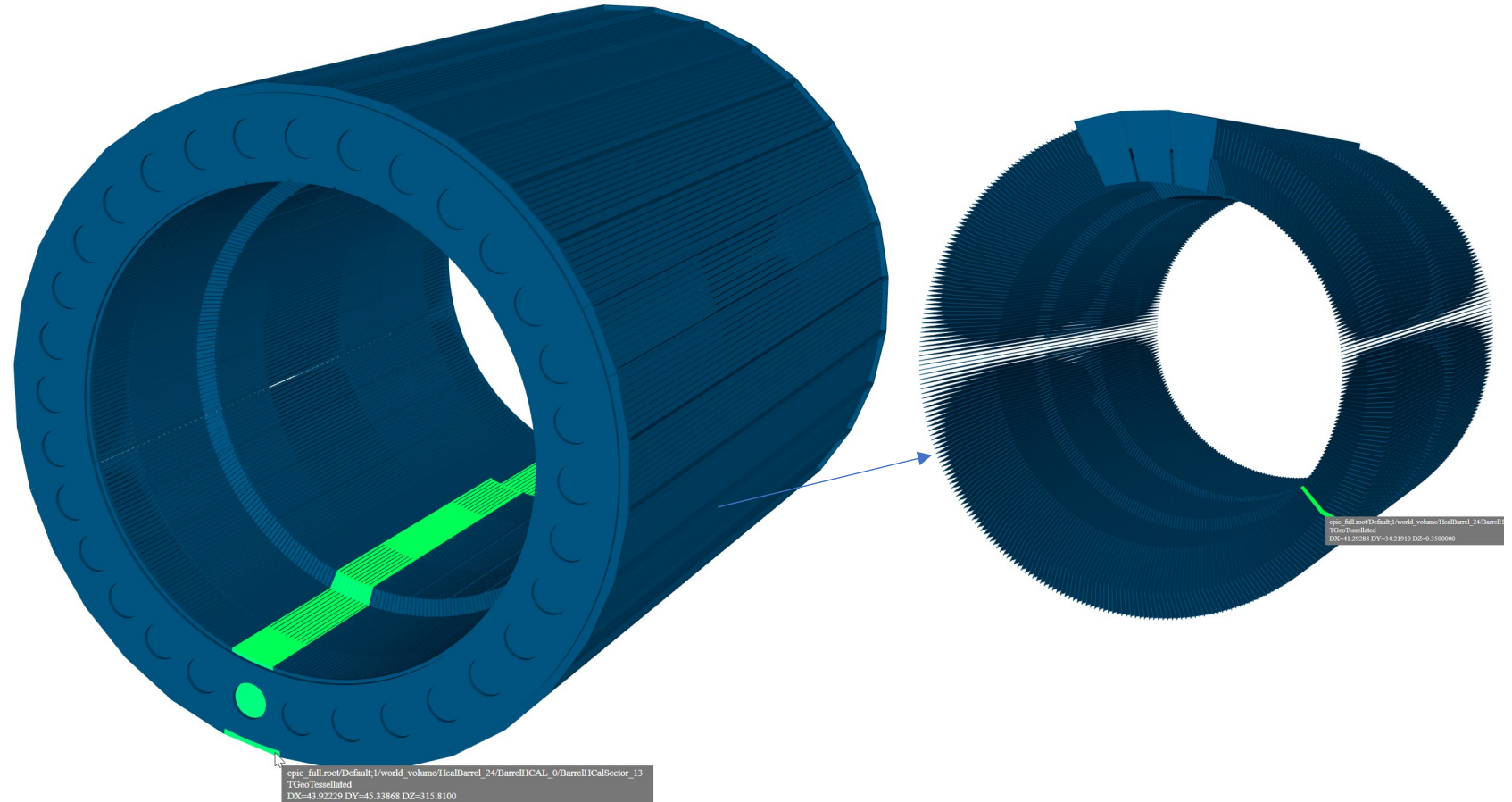
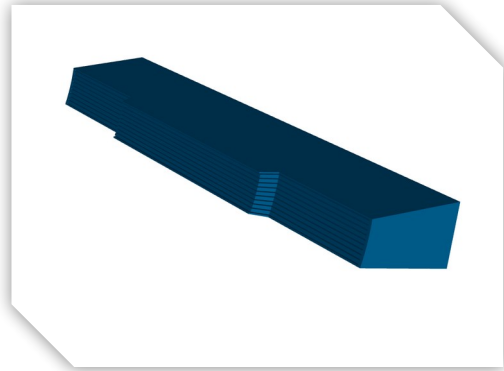
Geometry Node

Node Count	Geometry Name
2,007,793	EcalBarrelScFi
1,816,561	EcalBarrelImaging
1,185,962	LFHCAL
324,163	LumiSpecCAL
32,872	EcalEndcapN
25,211	DRICH
7,716	HcalBarrel
3,860	EndcapTOFSubAssembly
2,311	EndcapMPGDSUBAssembly
1,795	InnerMPGDBarrelSubAssembly
1,537	DIRC
1,443	BarrelTOFSubAssembly
1,151	OuterSiTrackerSubAssembly
910	RICHEndcapN
802	ZDC_Crystal
773	VertexBarrelSubAssembly
588	HcalEndcapPInsert
579	HcalFarForwardZDC_SiPMonTile
471	MiddleSiTrackerSubAssembly
455	ForwardRomanPot_Station_1



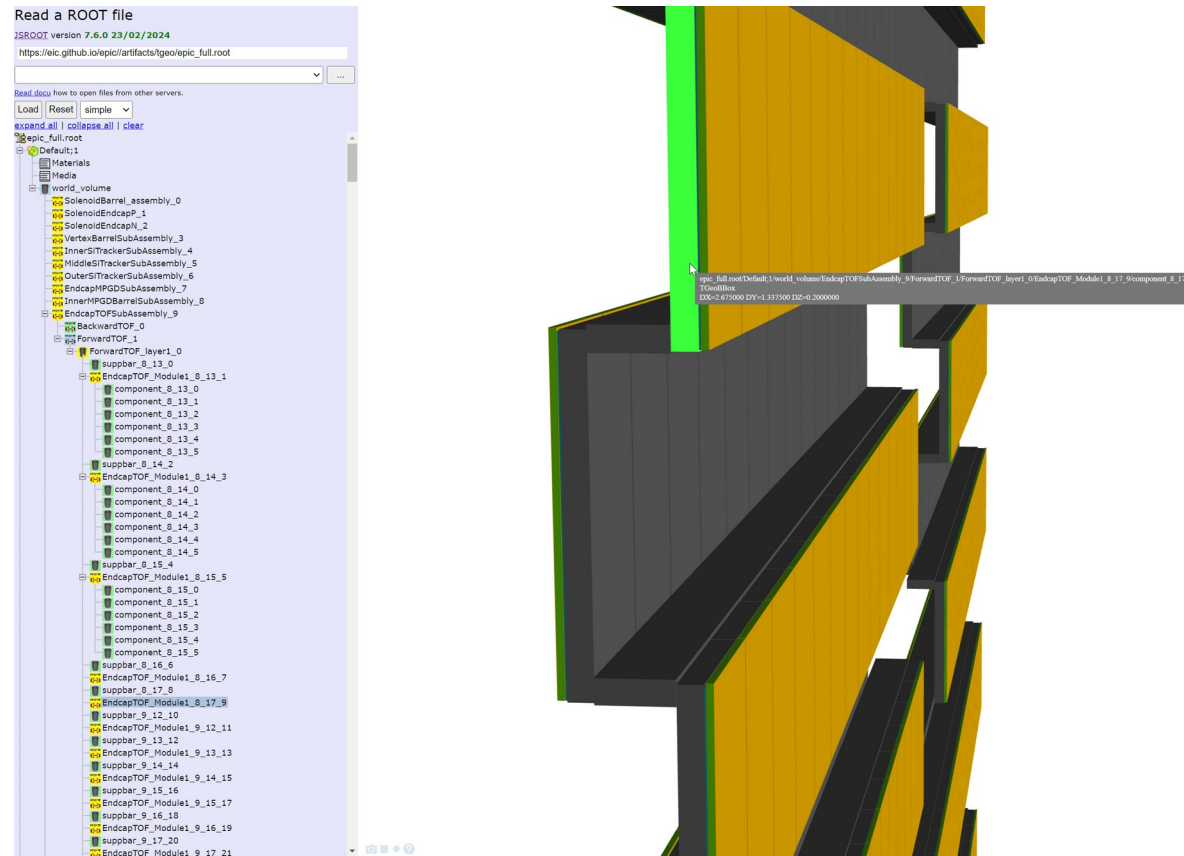
Hcal Barrel

- Small number of parts, but active use of tessellated objects

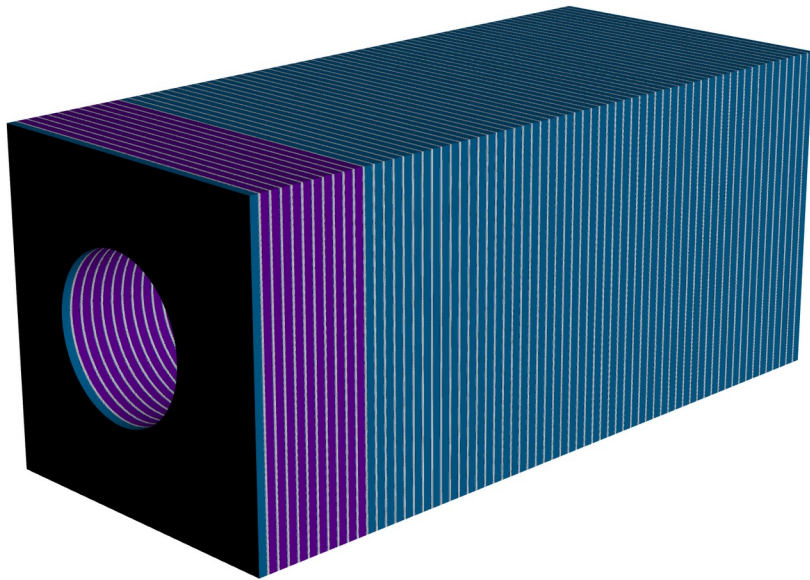


Inconsistent naming

- Many geometry elements named assuming no one ever will use names

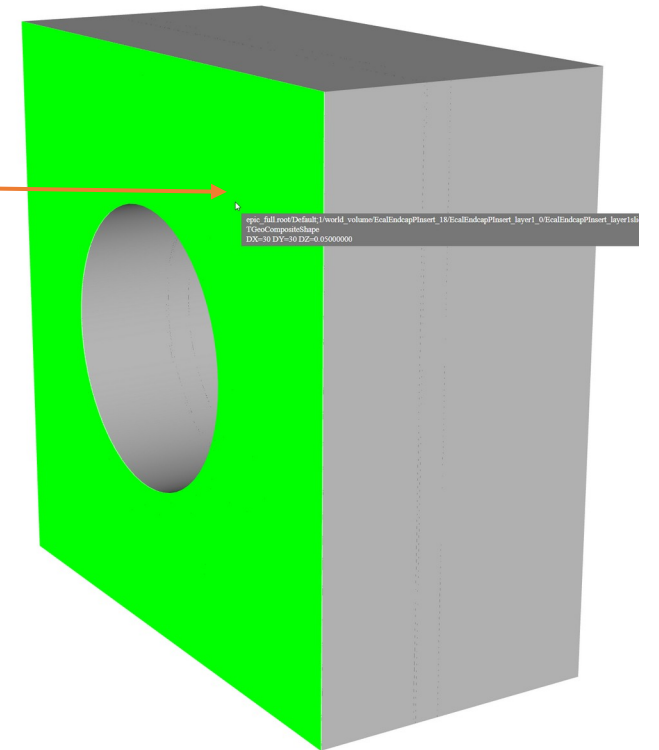


Same name? EcalEndcapPInsert



```
Read a ROOT file
ISRQOJ version 7.6.0 23/02/2024
https://eic.github.io/epic/artifacts/geolepic_full_root

Read .docu how to open files from other servers.
Load | Reset | simple v
expand all | collapse all | clear
+ OuterSTTrackerSubAssembly_6
+ EndcapMPGDSSubAssembly_7
+ InnerMPGDSBarrelSubAssembly_8
+ EndcapTOPSubAssembly_9
+ BarrelTOPSubAssembly_10
+ OuterBarrelMPGDSSubAssembly_11
+ BOTrackerSubAssembly_12
+ InnerTrackerSupport_assembly_13
+ DIRC_14
+ RICHEndcapN_Vol_15
+ DRICH_16
+ EcalEndcapP_17
+ EcalEndcapPInsert_18
+ EcalBarrelImaging_19
+ EcalBarrelScF_20
+ EcalEndcapN_21
+ LHCAL_env_22
+ HcalEndcapPInsert_23
+ HcalBarrel_24
+ HcalEndcapN_25
+ FluxEndcapN_26
+ BeamPipe_assembly_27
+ BOPF_BeamlineMagnet_assembly_28
+ BOAPF_BeamlineMagnet_assembly_29
+ Q1APF_BeamlineMagnet_assembly_30
+ Q1BPF_BeamlineMagnet_assembly_31
+ Q2PF_BeamlineMagnet_assembly_32
+ B1PF_BeamlineMagnet_assembly_33
+ B1APF_BeamlineMagnet_assembly_34
+ B2PF_BeamlineMagnet_assembly_35
+ BeamPipeB0_assembly_36
+ Pipe_cen_to_pos_assembly_37
+ QDEF_assembly_38
+ QDEF_vac_39
+ Q1EF_assembly_40
+ Q1EF_vac_41
+ BOEcal_42
+ ForwardOHMTracker_station_1_43
+ ForwardOHMTracker_station_2_44
+ ForwardOHMTracker_station_3_45
+ ForwardOHMTracker_station_4_46
+ HcalFarForwardZDC_SiPMTile_47
+ ZDC_Crystal_envelope_48
+ ForwardRomanPot_Station_1_49
+ ForwardRomanPot_Station_2_50
+ VacuumMagnetElement_assembly_51
+ Pipe_Q1eR_to_B2BaR_assembly_52
+ Magnet_Q1eR_assembly_53
+ Magnet_Q2eR_assembly_54
+ Magnet_B2aR_assembly_55
+ Magnet_B2BaR_assembly_56
+ Magnets_Q3aR_assembly_57
+ BackwardsTaggerStation_assembly_58
+ LumiWindow_vol_ExitWindow_59
+ LumiCollimator_assembly_60
+ SniperMag_assembly_61
+ AnalyzerMag_assembly_62
```



Performance Inner Detector

epic_inner_detector.root

(All numbers are [ms])

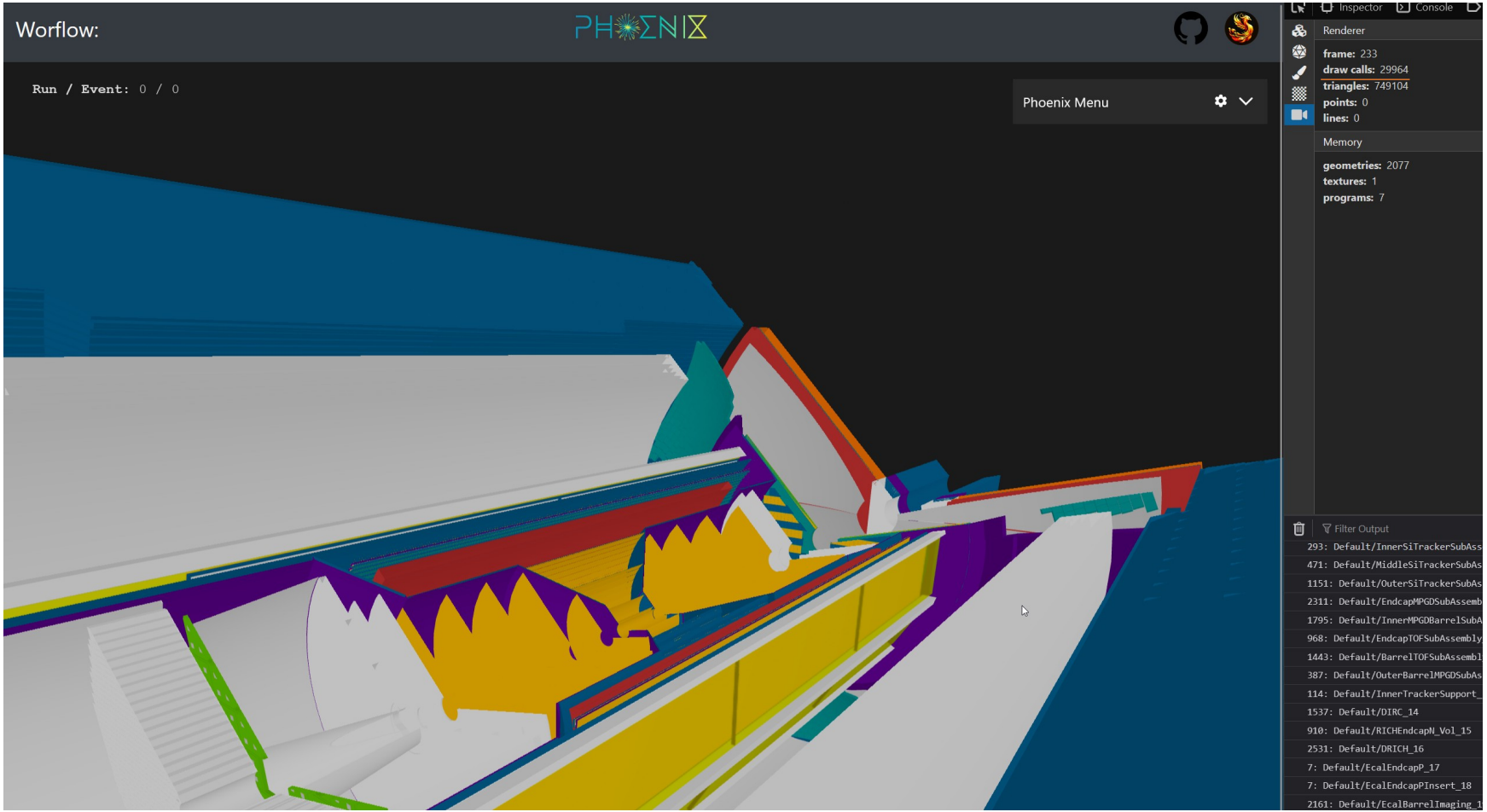
- Open root file: 678
- Reading geometry from file: 993
- Go over all nodes: 132
- Build geometry (for WebGL): 2002
- Convert to JSon geometry (Phoenix): 20

Total visible nodes 15902 numfaces 412466

So now it works fine?

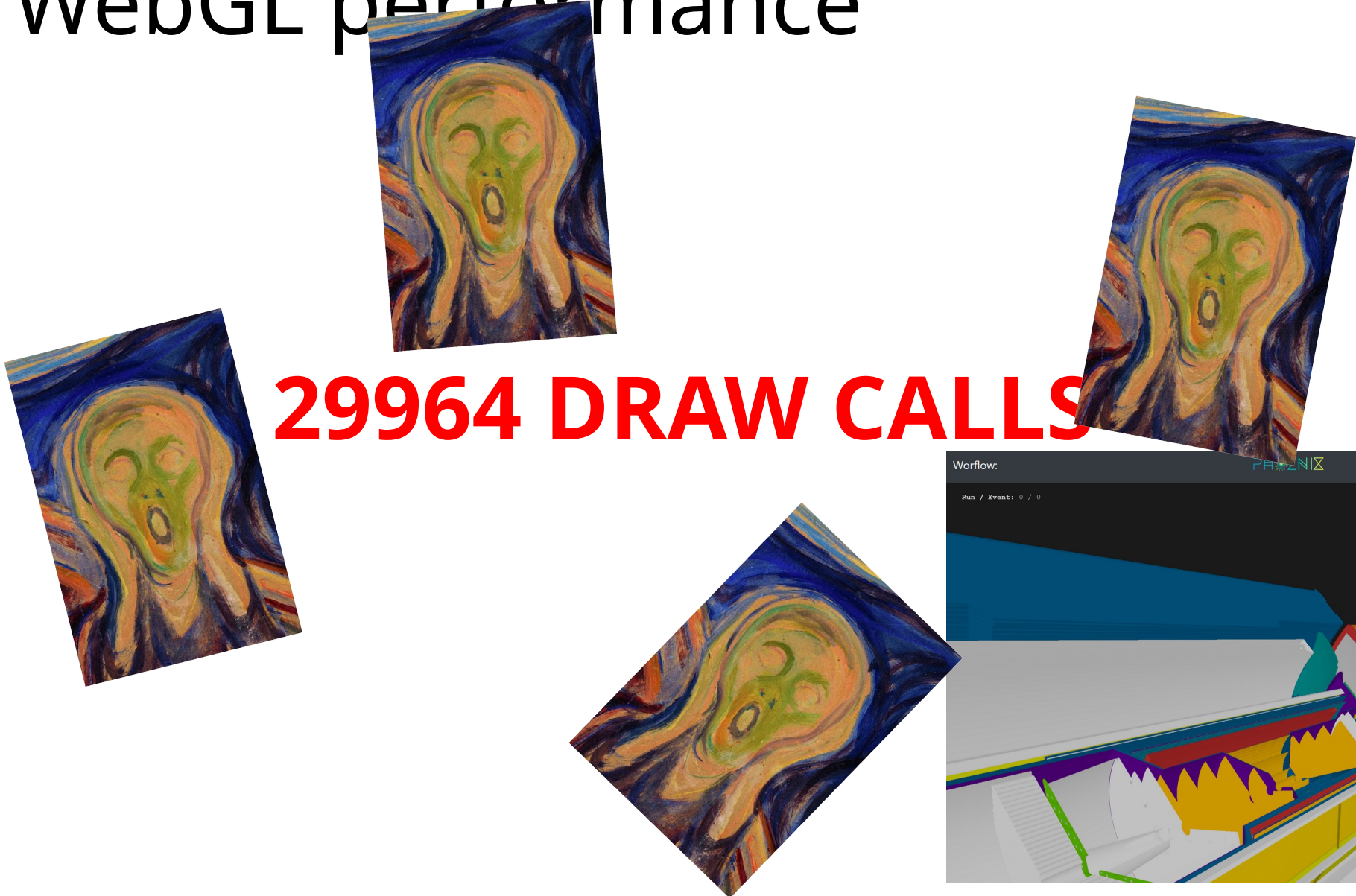
WebGL performance

29964 DRAW CALLS



WebGL performance

29964 DRAW CALLS



Workflow: PHOENIX

Run / Event: 0 / 0

Phoenix Menu

Inspector Console

Renderer

- Frame: 223
- Draw calls: 29964
- Triangles: 749104
- Points: 0
- Lines: 0

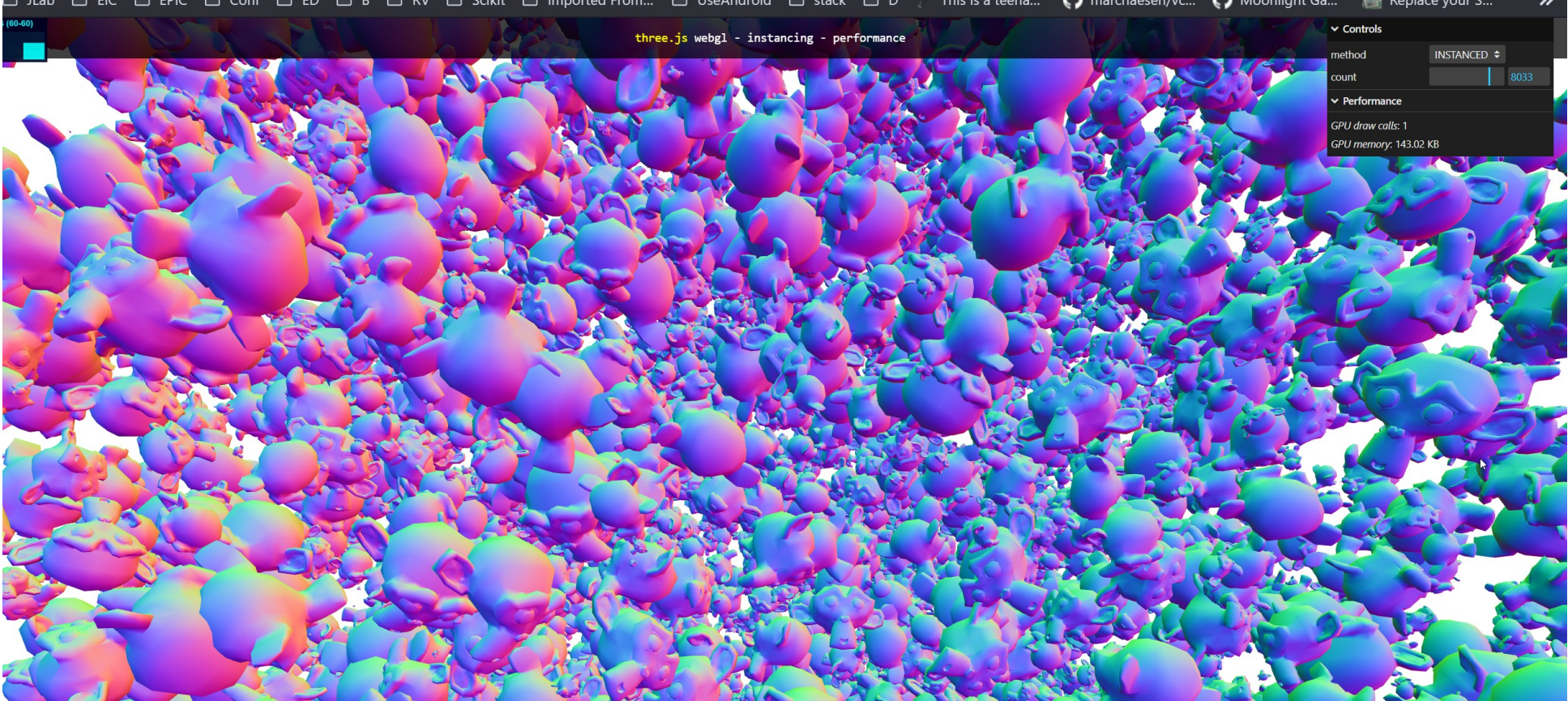
Memory

- geometries: 2077
- textures: 1
- programs: 7

Filter Output

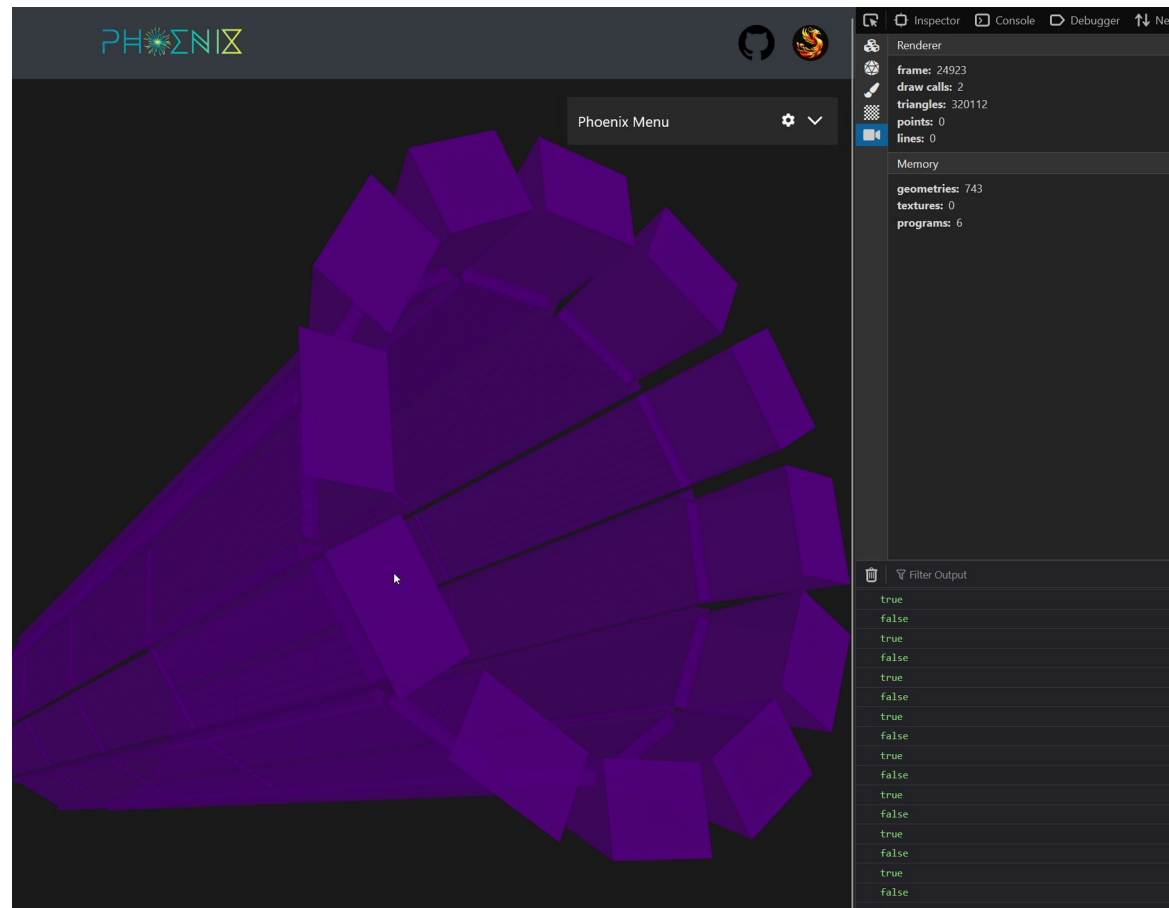
- 293: Default/InnerS1TrackerSubAs
- 472: Default/InnerS1TrackerSubAs
- 1152: Default/OuterS1TrackerSubAs
- 2111: Default/InnerS1TrackerSubAs
- 1795: Default/InnerS1TrackerSubAs
- 968: Default/InnerS1TrackerSubAs
- 1443: Default/InnerS1TrackerSubAs
- 387: Default/InnerS1TrackerSubAs
- 114: Default/InnerS1TrackerSupport
- 1537: Default/InnerS1TrackerSupport
- 918: Default/InnerS1TrackerSupport
- 2531: Default/InnerS1TrackerSupport
- 7: Default/InnerS1TrackerSupport
- 2161: Default/InnerS1TrackerSupport

What is good drawcalls number?



Can we do the same?

- Kind of but...



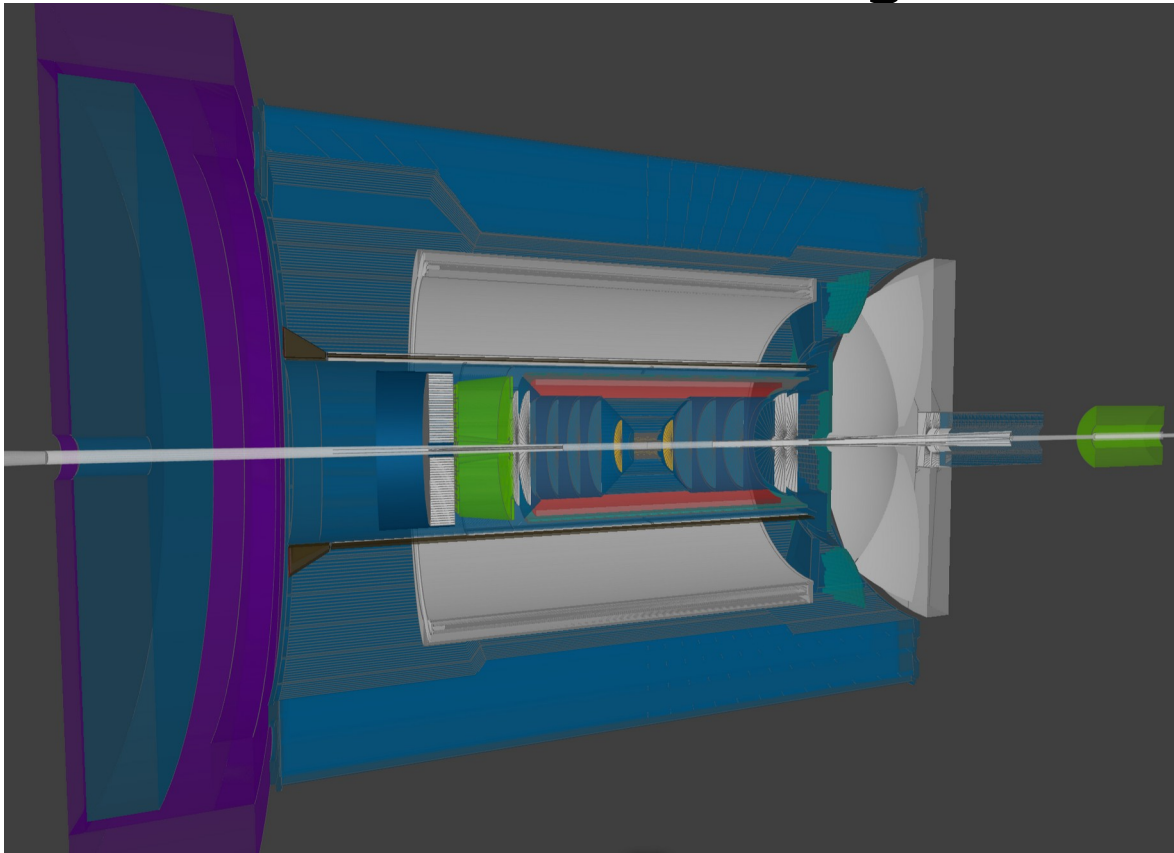
Data processing chains

- Geometry pipeline



Current results

- ~100 Drawcalls and 1000 geometries



Renderer

frame: 5352

draw calls: 116

triangles: 799026

points: 0

lines: 307594

Memory

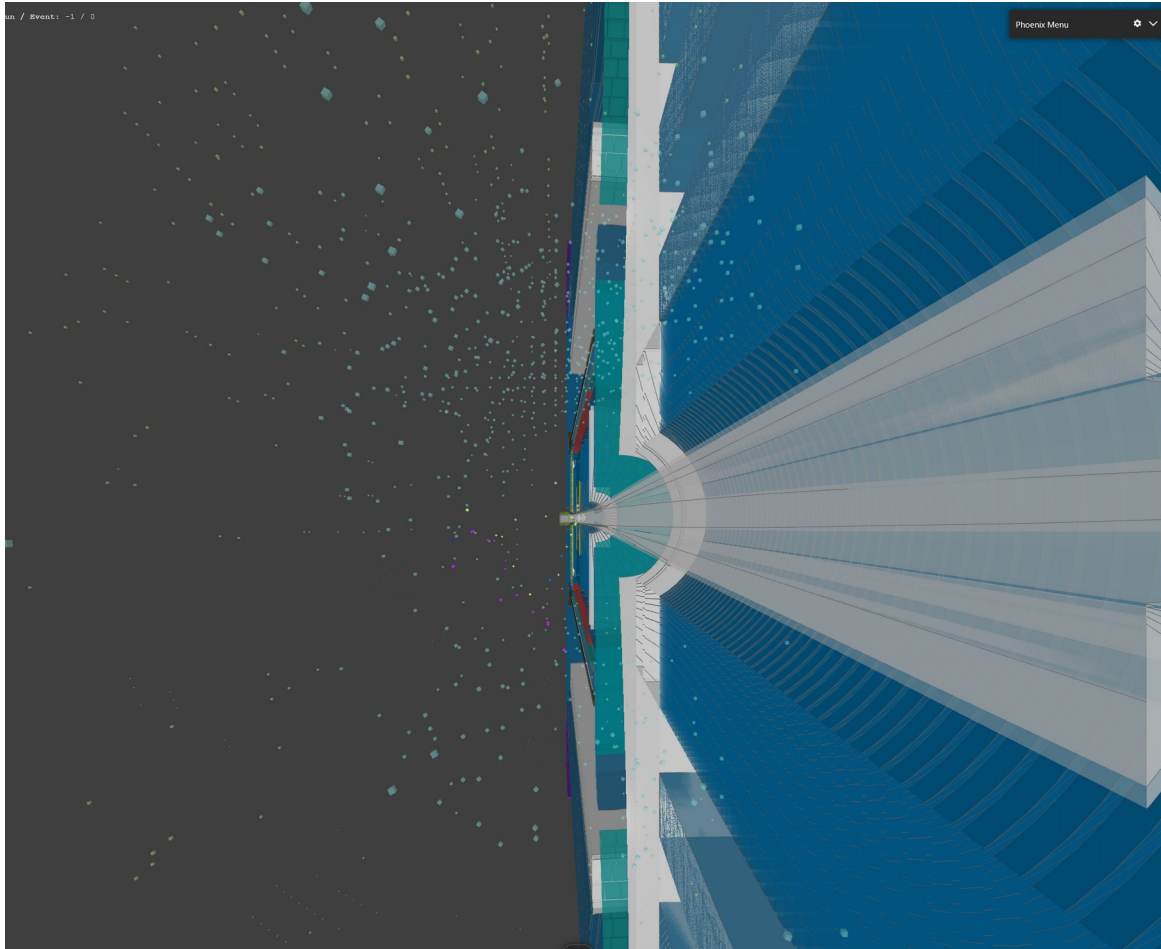
geometries: 752






textures: 0

programs: 10

What about hits?

Hits rendering is not optimized and in TODO



Renderer	
	frame: 2376
	draw calls: 17327
	triangles: 981722
	points: 69
	lines: 281672
Memory	
	geometries: 23879
	textures: 14
	programs: 10



Future plans

User local CLI (visioning)

- Vision:

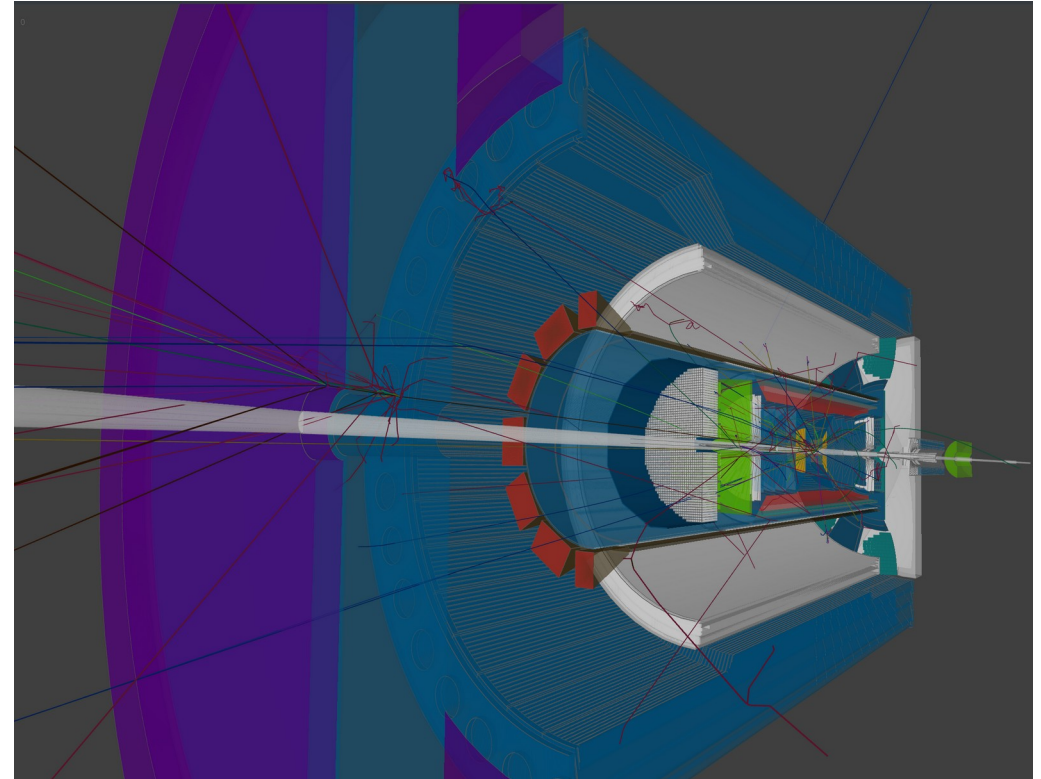
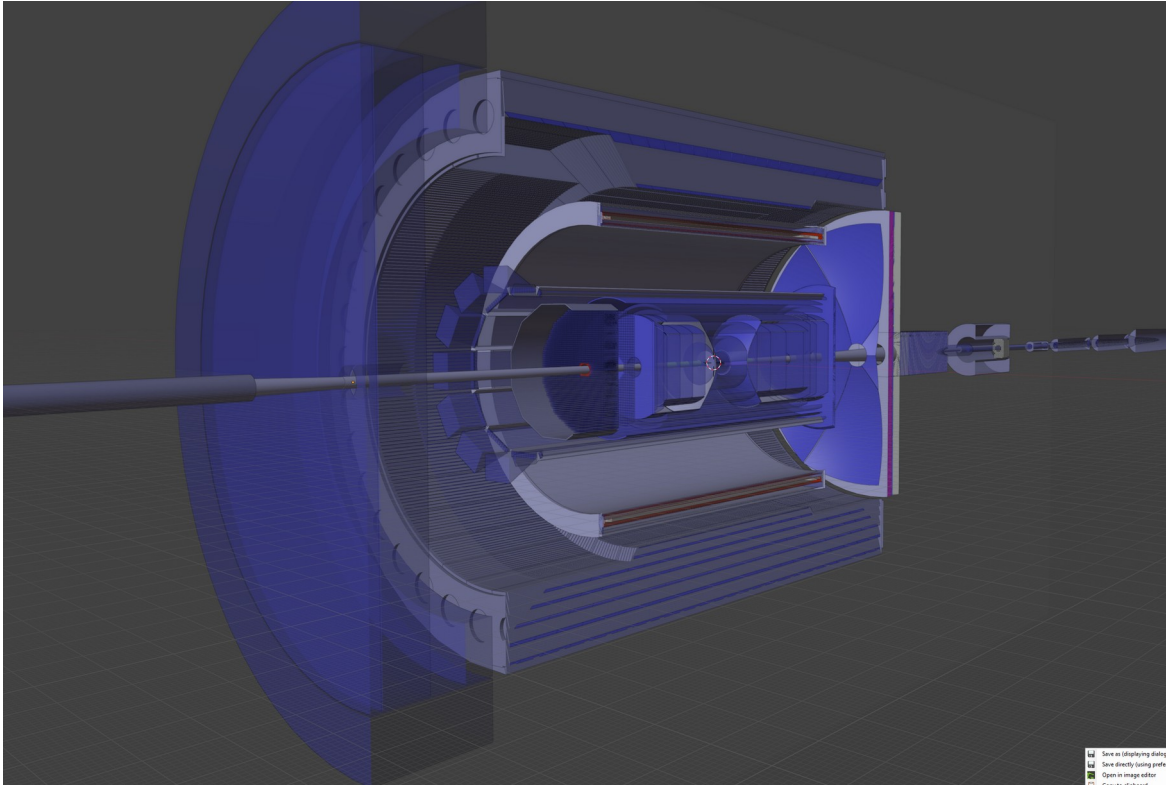
```
pip install pyrobird
```

```
> fbd display -g geometry -I input1 ...
```

```
(base) romanov@SmallBrave:/mnt/c/eic/firebird$ pip install pyrobird
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pyrobird in /home/romanov/.local/lib/python3.10/site-packages (0.0.5)
Requirement already satisfied: click in /home/romanov/.local/lib/python3.10/site-packages (from pyrobird) (8.1.7)

[notice] A new release of pip is available: 23.0 -> 24.0
[notice] To update, run: /usr/bin/python3 -m pip install --upgrade pip
(base) romanov@SmallBrave:/mnt/c/eic/firebird$ fbd display --geo='epic_full.root' -i sim.edm4hep.root -i reco.edm4eic.root
```

Better visual



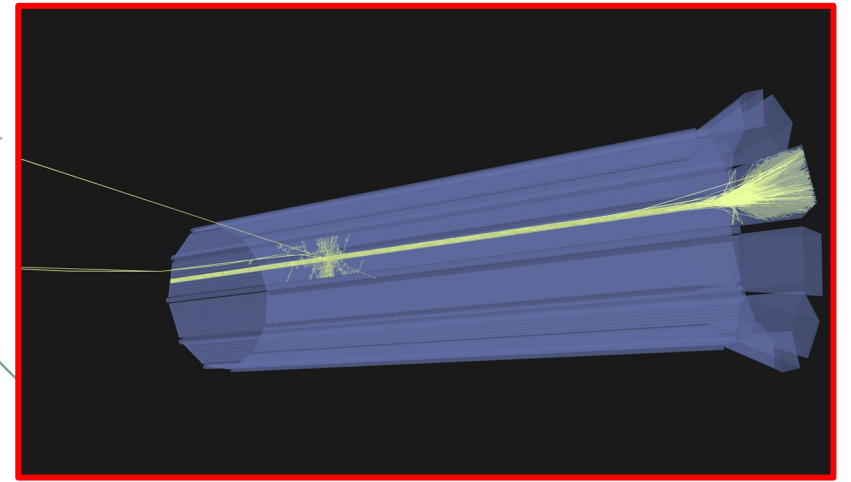
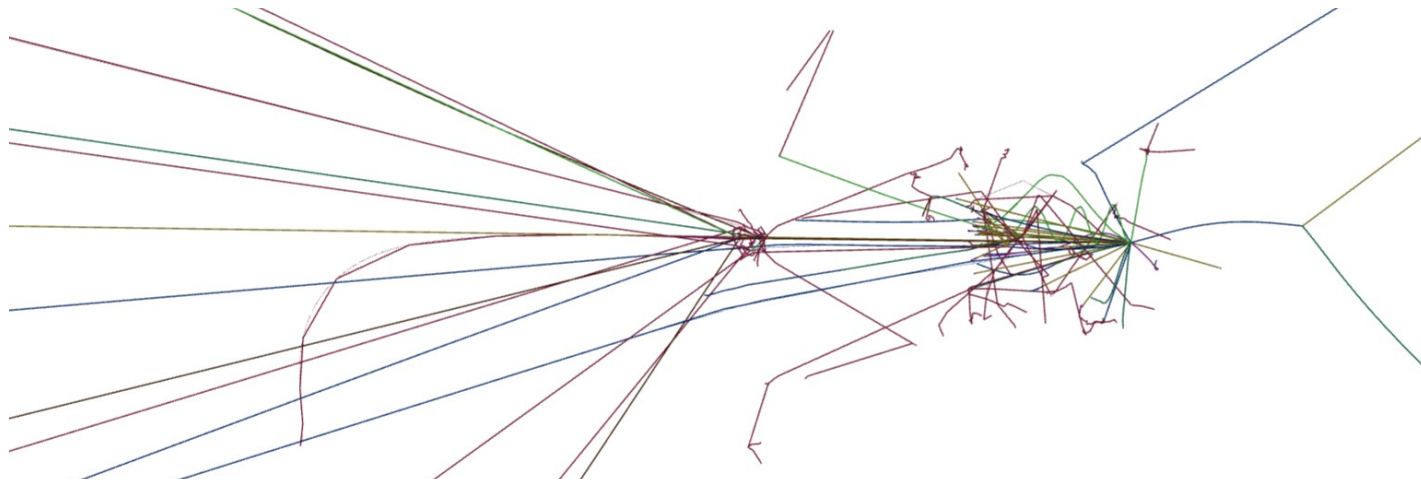
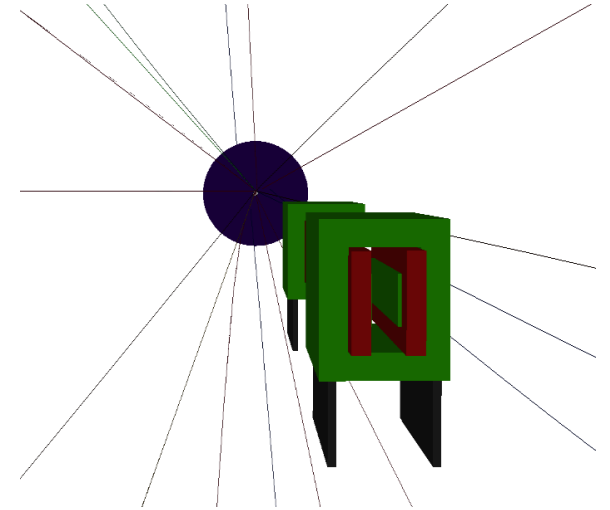
Geometry rules, community contributions

- All geometry manipulations done by set of rules
- Anticipated detector groups to contribute
- No javascript coding for rules (as it seems now)

```
99      {
100     ↗     namePattern: "*/EcalEndcapN*",
101     ↗     editRules: [
102     ↗       {pattern: "*/crystal*", prune:PruneRuleActions.RemoveSiblings},
103     ↗     ]
104     },
105     {
106     ↗     namePattern: "*/HcalEndcap*",
107     ↗     editRules: [
108     ↗       {pattern: "*/layer?slice1?", prune:PruneRuleActions.RemoveSiblings},
109     ↗     ]
110     },
111     {
112     ↗     namePattern: "*/HcalBarrel*",
113     ↗     editRules: [
114     ↗       {pattern: "*/Tile*", prune:PruneRuleActions.Remove},
115     ↗       {pattern: "*/ChimneyTile*", prune:PruneRuleActions.Remove},
116     ↗     ]
117     },
118     {
119     ↗     namePattern: "*/EndcapTOF*",
120     ↗     editRules: [
121     ↗       {pattern: "*/suppbar*", prune:PruneRuleActions.Remove},
122     ↗       {pattern: "*/component*3", prune:PruneRuleActions.RemoveSiblings},
123     ↗     ]
124     }
125   }
```

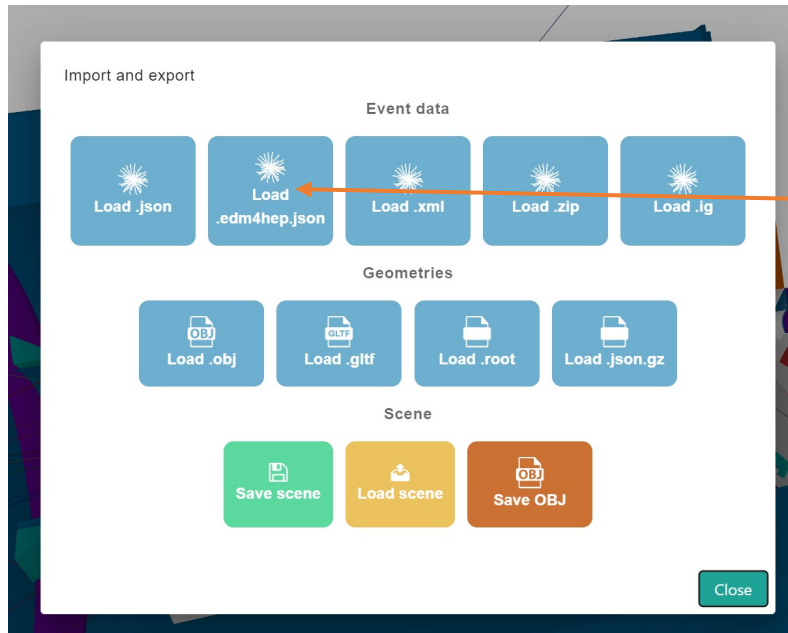
DD4HEP integration

- Using non-intrusive DD4Hep plugin
- Currently is a proof of concept
- Vision / possibilities:
 - Full “what is happening” (easily GBs for event)
 - Cuts by region/energy/vertex/detector, ...
 - Integration with npsim
 - Write to edm4hep
 - Far forward debug



Import export

- Load edm4hep from root (not json as of now)
- Load edm4eic (planned by June 2024)
- Load ACTS reconstructed tracks and info



```
(base) romanov@SmallBrave:~$ edm4hep2json --help
Usage: edm4hep2json [olenfvh] FILEPATH
  -o/--out-file          output file path
                        default: "?edm4hep.root" --> ".edm4hep.json"
  -l/--coll-list        comma separated list of collections to be converted
  -e/--events           comma separated list of events to be processed
  -n/--nevents         maximal number of events to be processed
  -f/--frame-name      input frame name
                        default: "events"
  -v/--verbose         be more verbose
  -h/--help           show this help message
```

Deliverables June

Deliverables will depend on community. Core plans:

- **Software perspective:**

- Initial integration with campaigns
- Improve on modularity, making more convenient for groups to work on their parts of interest
- Improve on cli part, make convenient for users and in batch operation
- Implement particular event objects if available (recoil electron, jets, connected flows)
- **Acts data source**
- **EDM4EIC data source**

- **Users' perspective:**

- Make tutorials on how to contribute to event display software
- Collect and analyze feedback from the community to focus on further priorities
- Can select runs and events from the latest simulation campaign reconstructed files (more on planning on the next page)

More on planning and deliverables

- [Implementation plan for event display on google docs](#)
(written internally 2 months ago but is actual)
- [Requirements for event display](#)