October 19 - 25, 2024

CHEP 2024

Conference on Computing in High Energy and Nuclear Physics

GEOMETRY SIMPLIFICATION METHODS FOR VIRTUAL REALITY APPLICATIONS

Roger JONES

Lancaster University ***For:***

SHARMAZANASHVILI Alexander

Giorgi Mirziashvili

Kote TSUTSKIRIDZE

Ela ABRAMOVI

Avtandil KHELASHVILI

Aleskander ALIKHANOV

Georgian Technical University

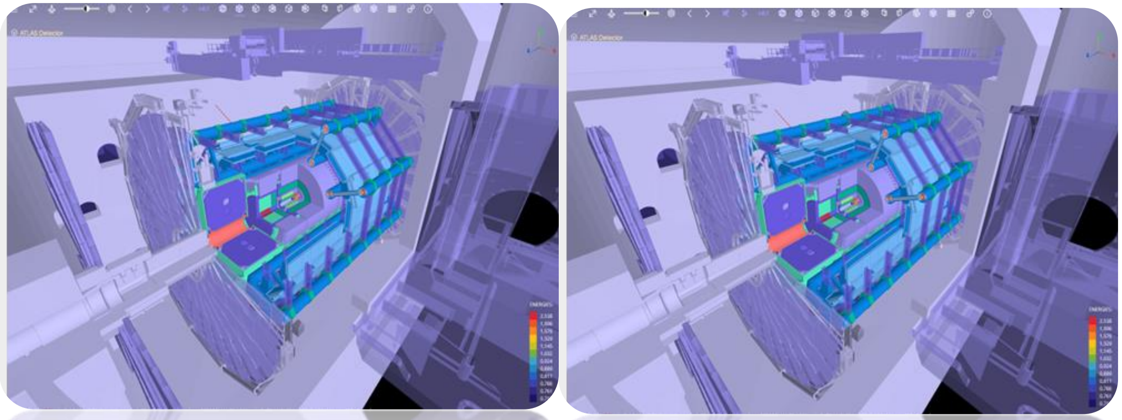
https://indico.cern.ch/event/1338689/

I. Challenges of VR Applications

Requirements

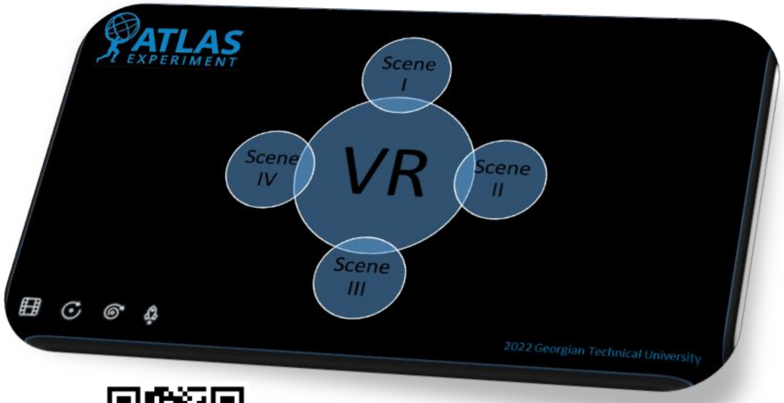
The Virtual Reality (VR) applications should be

- extensive
- easily accessible
- compatible with most hardware and operating systems
- simple to use
- with a well-developed user framework
- open source



About the ATLAS VR Application

- Tracer-VR is the Virtual Reality browser-based application for the virtual tours at the ATLAS Point 1
- Virtual Reality headsets necessary for the application are Google Cardboard class units
- The application puts moderate load on the GPU and to ensure good quality of the 3D scenes and smooth movement on the average power phones



Google Chrome is recommended

Challenge

- High-quality visualization scenes cause problems in application performance
- The Quality and Performance are antagonistic parameters of the VR

Quality-vs-Performance

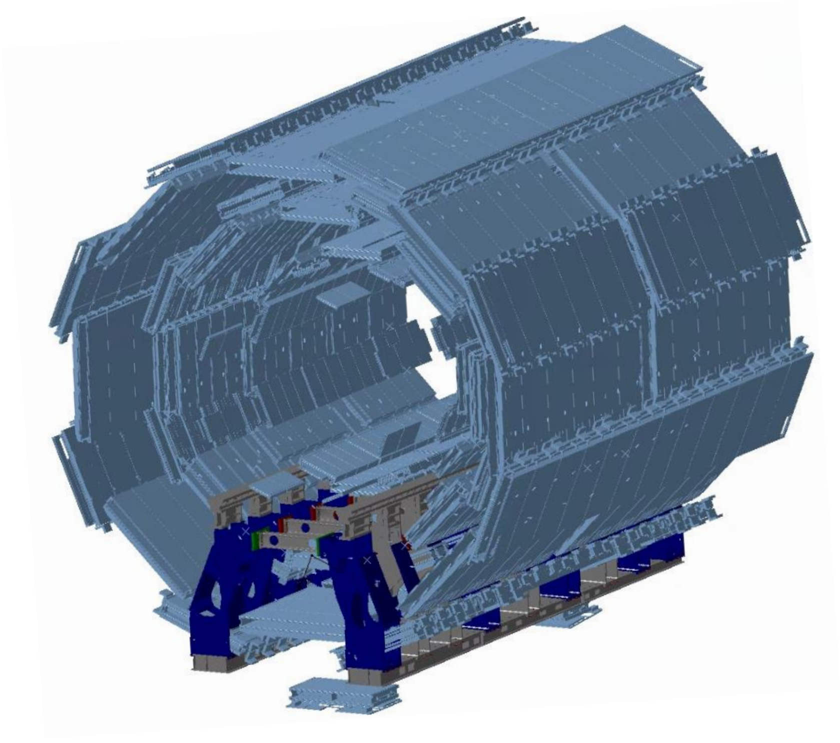
- The major challenge of VR is to find a balance between the performance of applications and the quality of the 3D scenes

➢ Geometry should be simplified to achieve a balance

Challenge

- Case Study : Investigation of Scene #2 of the Application

Scene #2 : The ATLAS Barrel Muon Chambers Assembly with Feet



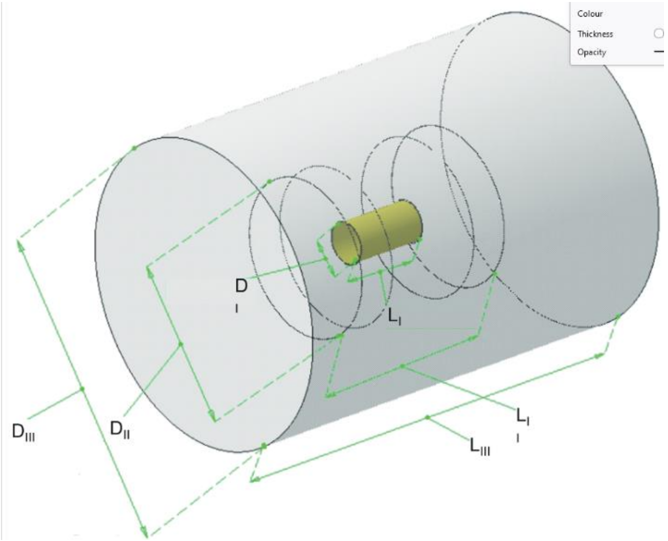
Frames per Second : 7
Triangles : 61 million
Load time : 17.210 sec
Draw calls : 340

- According to META Draw Calls value of more than 200 significantly increases the GPU load and kills application performance
- ➢ Geometry should be Simplified

II. Geometry Simplification Steps

STEP#01: PARAMETERS DEFINITION

- R_{sc} - the screen resolution. VR applications run on smartphones. Therefore, the screen resolution is expected 360 x 640 pixels



- D_{min}/L_{min} the minimal diameter and the minimum length of the components. Decide which components are not essential to visualize because of their size. All components with dimensions less than D_{min}/L_{min} will be deleted from the geometry
- Z_{max} - maximal zooming rate coefficient of the scene describes the maximal scale of the visualization of the components. Z_{max} value should be defined according to R_{sc} and D_{min}/L_{min} values

Geometry Simplification Steps

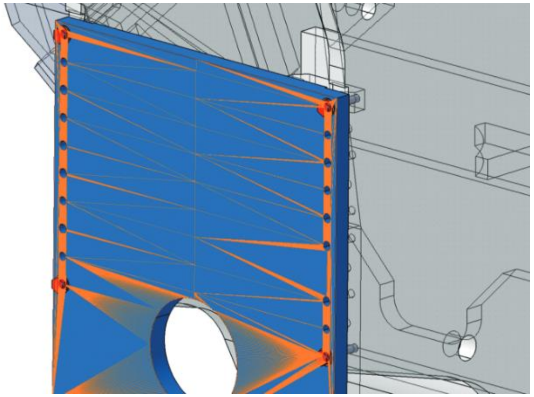
STEP#02: REMOVAL OF THE COMPONENTS

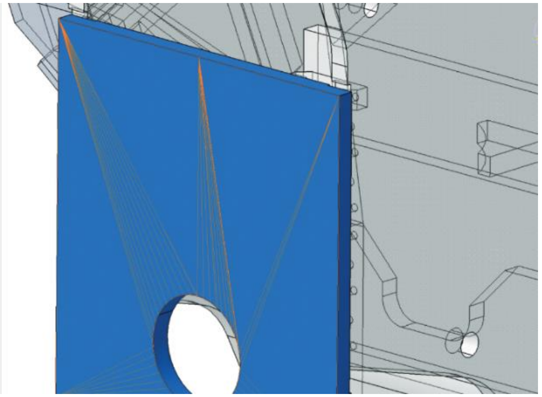
- Removal of the components with dimensions less than D_{min}/L_{min}. Because D_{min}/L_{min} sets the minimal dimensions of the components to ensure their visibility in the scene, all components less than those values should be deleted

STEP#03: REMOVAL OF THE PARTS

- The ‘As-built’ geometry of the detector contains components with different purposes – facilities, services, mechanical, support and access structures, and others
- Depending on the visualisation applications’ purpose, some categories can and should be deleted

STEP#04: REMOVAL OF THE HOLES

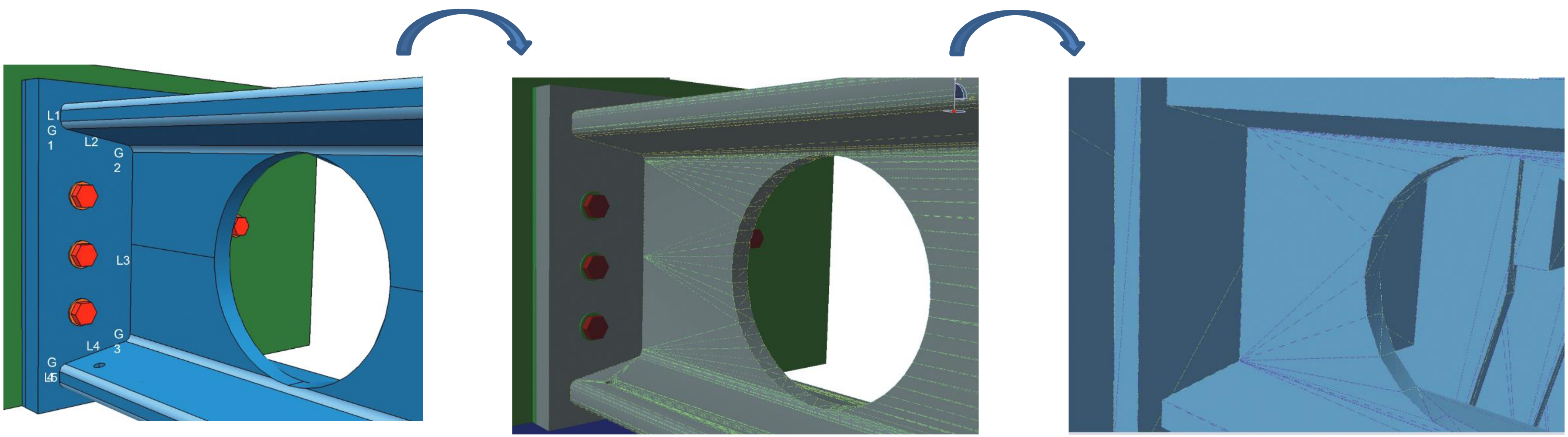




- Each hole in the geometry is a set of lines with several vertices. Triangles are created for all the vertices
- Therefore, it is necessary to remove holes from the geometry description as much as possible. Only exceptional holes, describing essential functionality or facility structure features should remain
- Usually, detector facilities contain <u>many</u> holes, and their removal is a critical step in the simplification of the geometry

STEP#05: TRANSFORMATION OF THE ROUNDED PROFILES

- Transformation of rounded connections to straight-line connections. The ‘As-built’ geometry of the detector has many construction elements for mechanical engineering purposes
- The galettes and chamfers are usually used as a standard constructive joining element. They have to be replaced by straight-line connections

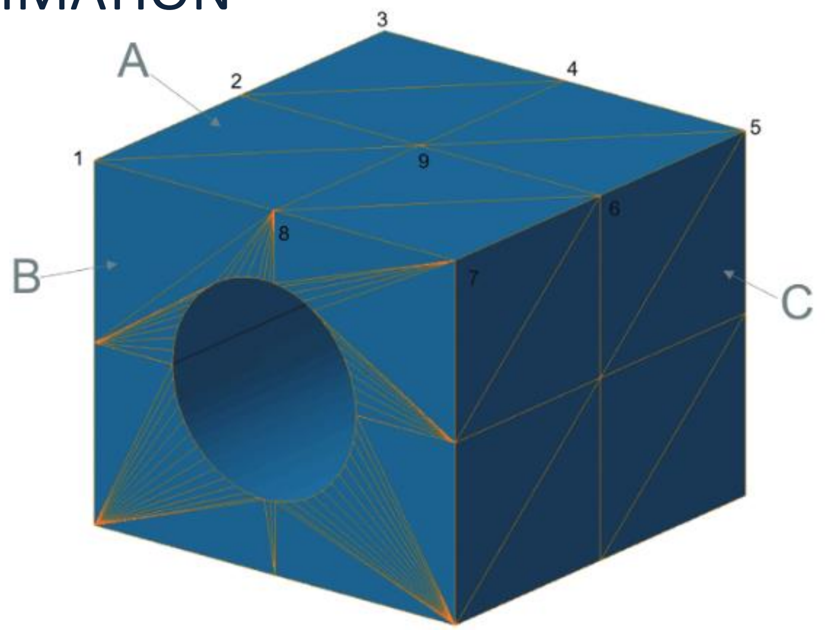


STEP#06: TRANSFORMATION OF THE COMPLEX SURFACES

- The ATLAS detector is a symmetrical structure. There are complex components built from a number of similar subcomponents
- Those subcomponents should be replaced by one generic geometry

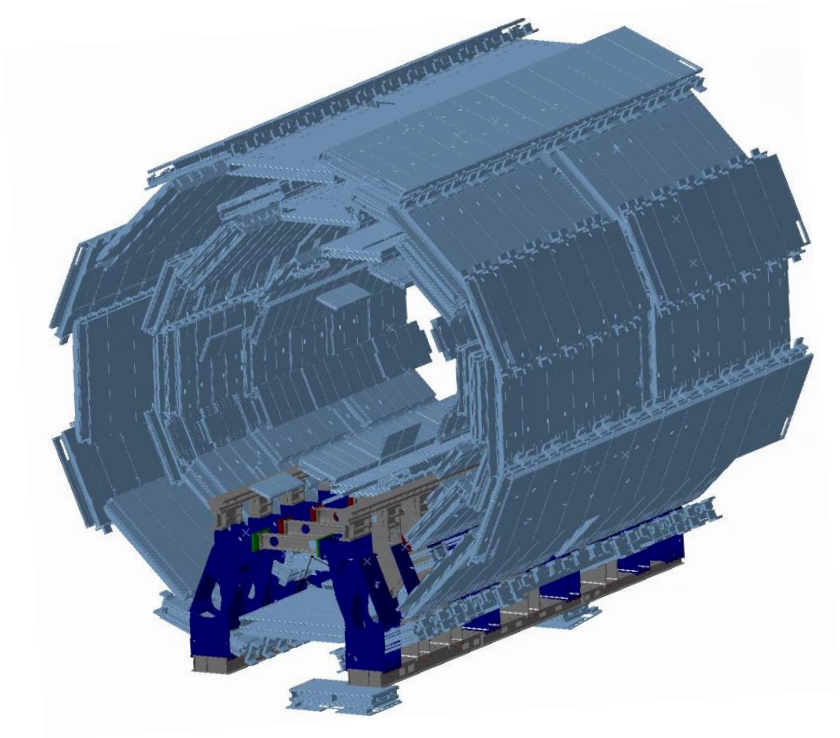
STEP#07: SELECTION OF THE APPROXIMATION

- The approximation value dramatically impacts the number of triangles in the scene and the quality of the visualization scene
- The critical parameter for choosing the length is the Z_{max} defined in STEP#01



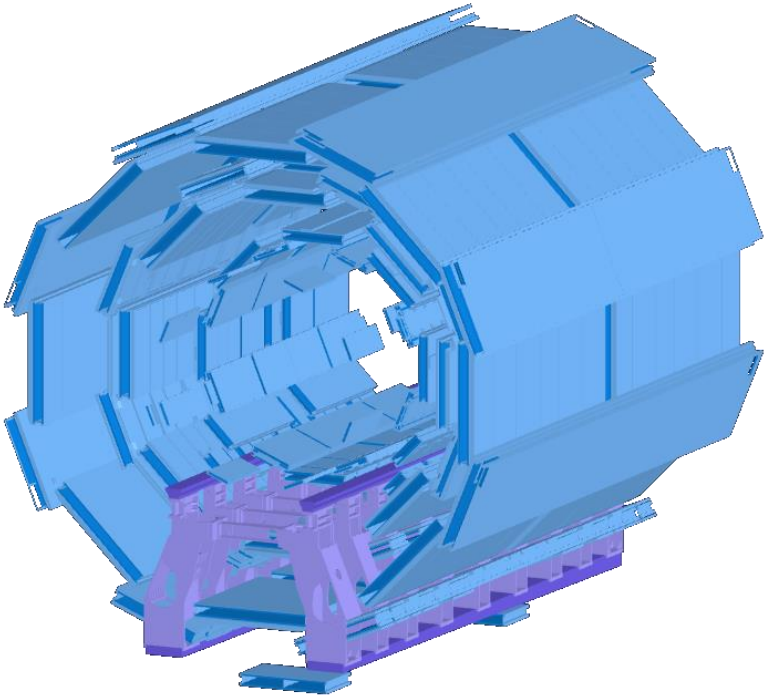
Results of the Simplification of the Scene#02

Source Geometry



Frame per Second : 7
Triangles : 61 million
Load time : 17.210 sec
Draw calls : 340

Simplified Geometry



Frame per Second : 59
Triangles : 0.015 million
Load time : 1.210 sec
Draw calls : 87

III. Geometry Development Life Cycle

Geometry Development Life Cycle

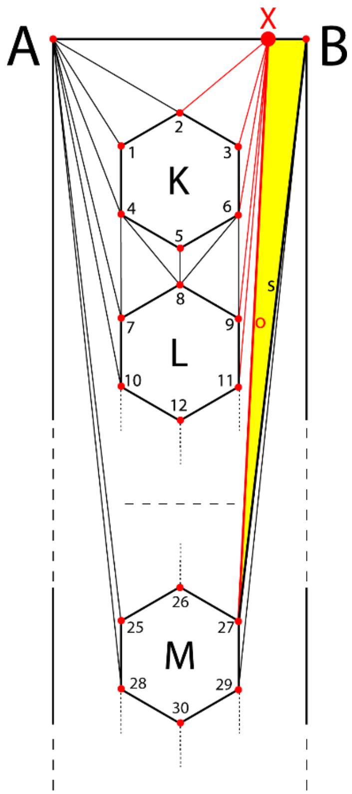
- The as-built geometry of the ATLAS detector is stored in the CATIA engineering database in the form of solids
- VR application is built based on the WebGL Three.js gaming engine which uses GLB geometries based on the facets
- The blender application is used for CATIA to GLB transformation.
- However, direct export/import from CATIA-2-Blender is almost impossible for the ATLAS geometry because of its complexity. Reading the *wrl* file generated by CATIA for Scene#02 takes about 5 hours and ends with a crash because of the memory-overloaded

➢ Therefore, intermediate geometry transformations are needed based on the auxiliary applications. We used KeyShot to do this.

Geometry Development Life Cycle

Challenges of using Intermediate transformations

- The appearance of the extra faces or vertices on the geometry that were not present in the original geometry
- Geometry Transformation with KeyShot foresees:
  1. Initial CAD import - the application reads the precise mathematical parameters
  2. Tessellation - mathematical surfaces converting into a mesh of polygons (usually triangles)
  3. Exporting the Mesh - materials, textures, and other scene data adding to mesh and exporting to the target format
- Tessellation is the stage where the extra vertices (like X point) are generated



Geometry Development Life Cycle

- Therefore, in addition to the simplification steps already described, the so-called "instant clean" must be implemented in Blender
- This follows the steps below:

1. Removing Duplicate Vertices
2. Deleting Unused Materials
3. Simplifying Mesh Geometry
4. Fixing Normals
5. Removing Loose Geometry
6. Cleaning Up Mesh Topology

Conclusion

1. High-quality VR applications need development of the 3D scenes with simplified geometries

2. Geometry simplification implies 7 consecutive steps with the removal of the triangles from the description

3. Geometry development life cycle needs intermediate transformations with auxiliary applications

4. As well as the simplification steps, instant cleaning is needed for the removal of the "artificial" vertices