

Geometric Modeling

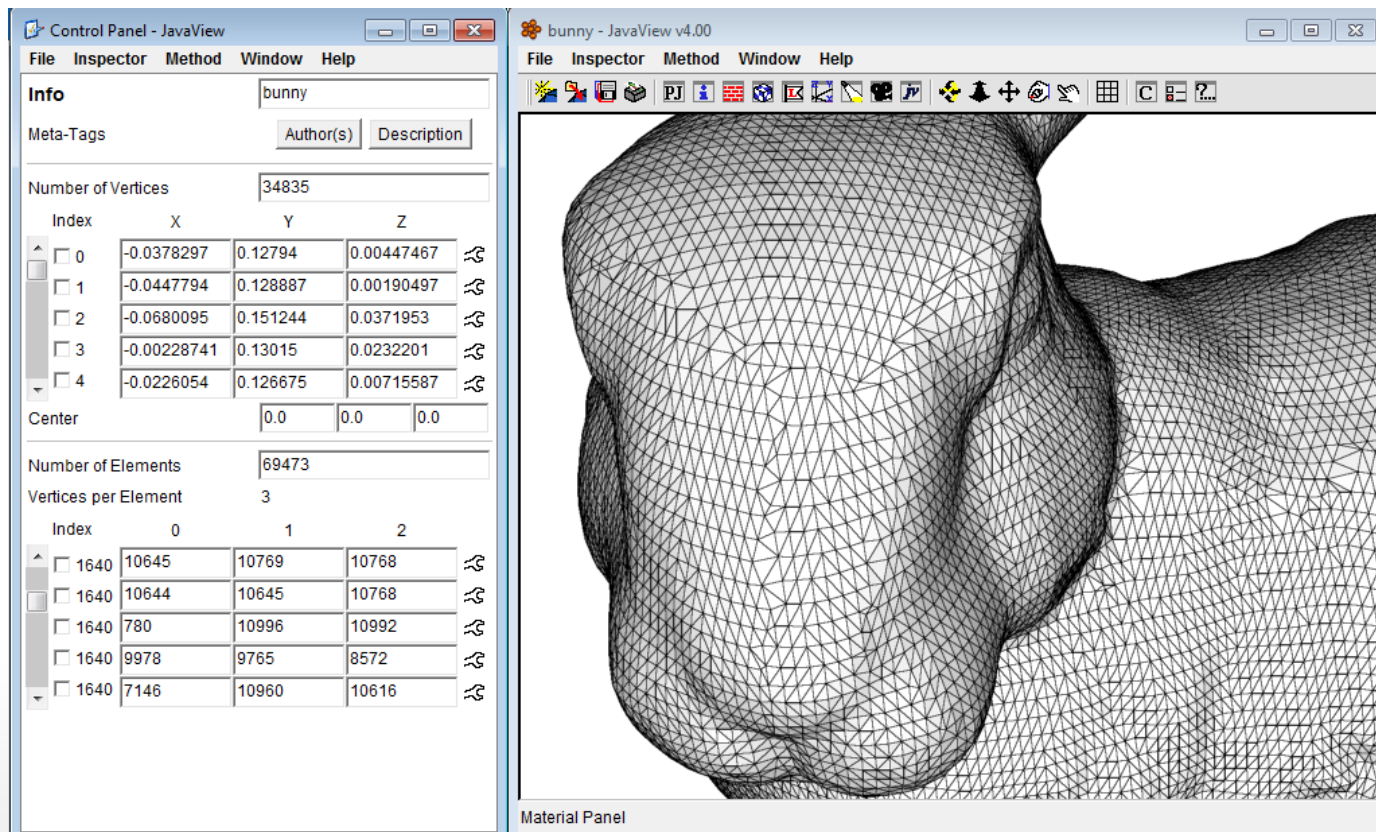
2015

Surface Meshes

Triangle Meshes

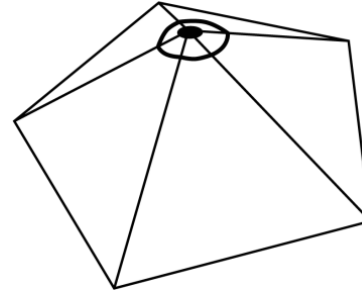
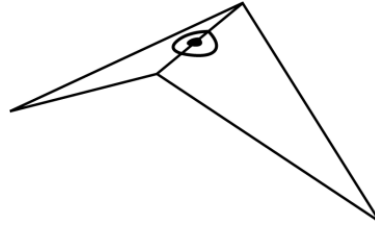
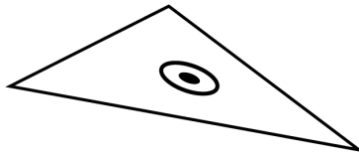
Representation of a triangle mesh in \mathbb{R}^3

- Vertices: a finite list $\{v_1, \dots, v_n\}$ of points in \mathbb{R}^3
- Faces: a list of triples, e.g. $\{\{2,34,7\}, \dots, \{14,7,5\}\}$



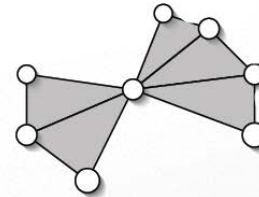
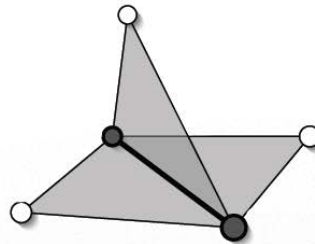
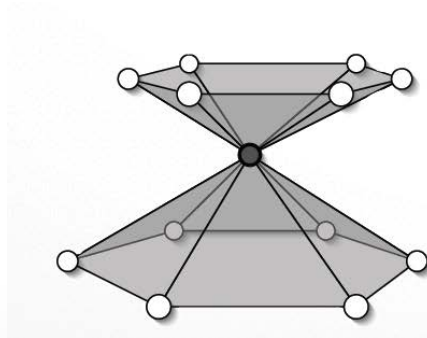
Surfaces

Surface: All points are locally disks (except boundaries)

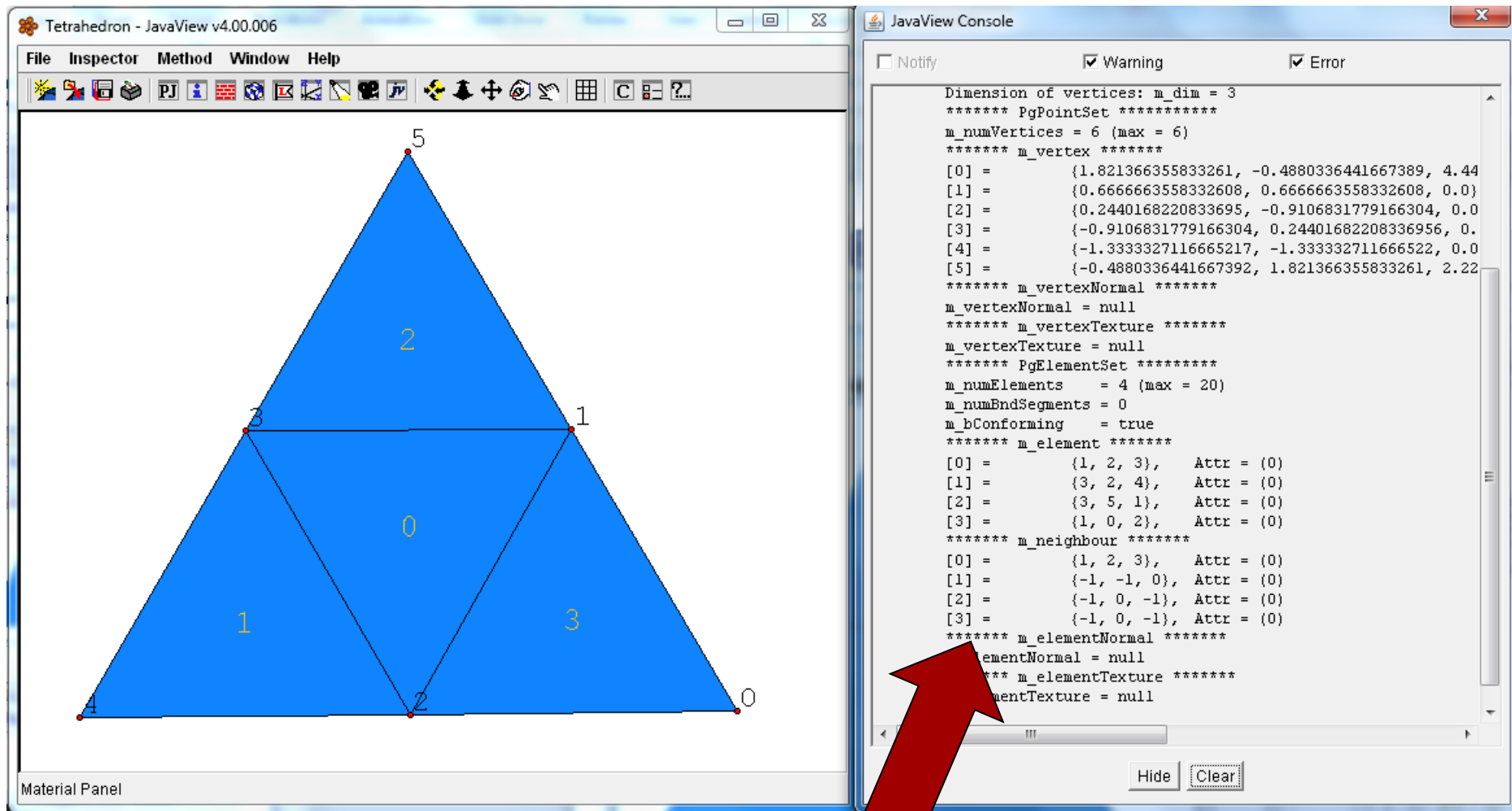


- Required by many algorithms

Examples: Non-manifold



Local Neighborhoods



The screenshot displays the Tetrahedron - JavaView v4.00.006 application. The main window shows a blue tetrahedron mesh with vertices labeled 0 through 5. The mesh is composed of four triangles: a central triangle (0, 1, 2) and three outer triangles (0, 1, 3), (0, 2, 3), and (1, 3, 5). The console window on the right shows the following output:

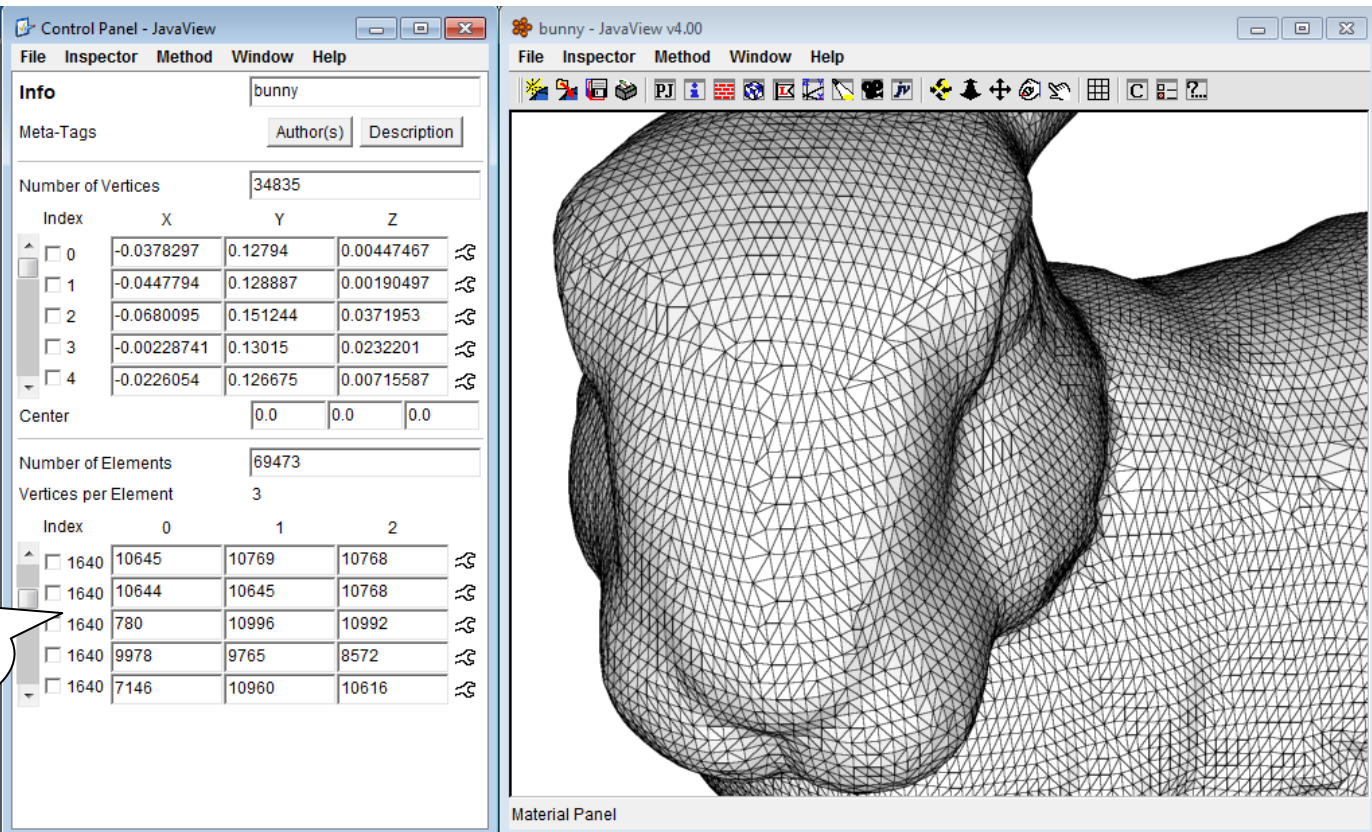
```
Dimension of vertices: m_dim = 3
***** PgPointSet *****
m_numVertices = 6 (max = 6)
***** m_vertex *****
[0] = {1.821366355833261, -0.4880336441667389, 4.44}
[1] = {0.6666663558332608, 0.6666663558332608, 0.0}
[2] = {0.2440168220833695, -0.9106831779166304, 0.0}
[3] = {-0.9106831779166304, 0.24401682208336956, 0.0}
[4] = {-1.3333327116665217, -1.333332711666522, 0.0}
[5] = {-0.4880336441667392, 1.821366355833261, 2.22}
***** m_vertexNormal *****
m_vertexNormal = null
***** m_vertexTexture *****
m_vertexTexture = null
***** PgElementSet *****
m_numElements = 4 (max = 20)
m_numBndSegments = 0
m_bConforming = true
***** m_element *****
[0] = {1, 2, 3}, Attr = (0)
[1] = {3, 2, 4}, Attr = (0)
[2] = {3, 5, 1}, Attr = (0)
[3] = {1, 0, 2}, Attr = (0)
***** m_neighbour *****
[0] = {1, 2, 3}, Attr = (0)
[1] = {-1, -1, 0}, Attr = (0)
[2] = {-1, 0, -1}, Attr = (0)
[3] = {-1, 0, -1}, Attr = (0)
***** m_elementNormal *****
m_elementNormal = null
***** m_elementTexture *****
m_elementTexture = null
```

A red arrow points from the console output to the central triangle (0, 1, 2) in the mesh.

Index of the triangle opposite to the vertex (with the same position in the array) is stored and -1 for boundaries

Topology

Topology



The screenshot displays the JavaView interface. The left pane, titled 'Control Panel - JavaView', shows the 'bunny' model's properties. The 'Info' section lists 'Number of Vertices' as 34835 and 'Number of Elements' as 69473. The 'Vertices per Element' is set to 3. Below this, a table lists the first five vertices of the face array, each with an index and X, Y, and Z coordinates. A speech bubble points to this table with the text 'Topology!'. The right pane, titled 'bunny - JavaView v4.00', shows a 3D wireframe view of the bunny model, illustrating the surface defined by the topology.

Index	X	Y	Z
0	-0.0378297	0.12794	0.00447467
1	-0.0447794	0.128887	0.00190497
2	-0.0680095	0.151244	0.0371953
3	-0.00228741	0.13015	0.0232201
4	-0.0226054	0.126675	0.00715587

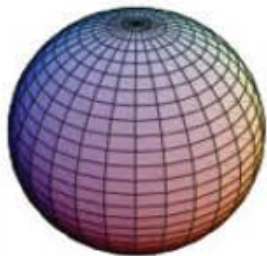
Topology!

The face array contains information about the surface that is independent of the choice of vertex positions

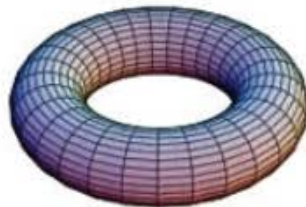
Topological Classification

Genus (compact, orientable surfaces)

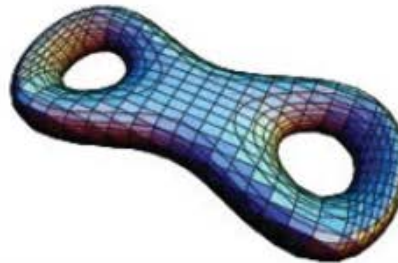
- Half the maximal number of closed paths that do not disconnect the surface
- Intuition: Number of holes (or handles)



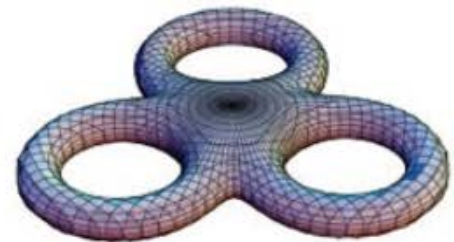
Genus 0



Genus 1

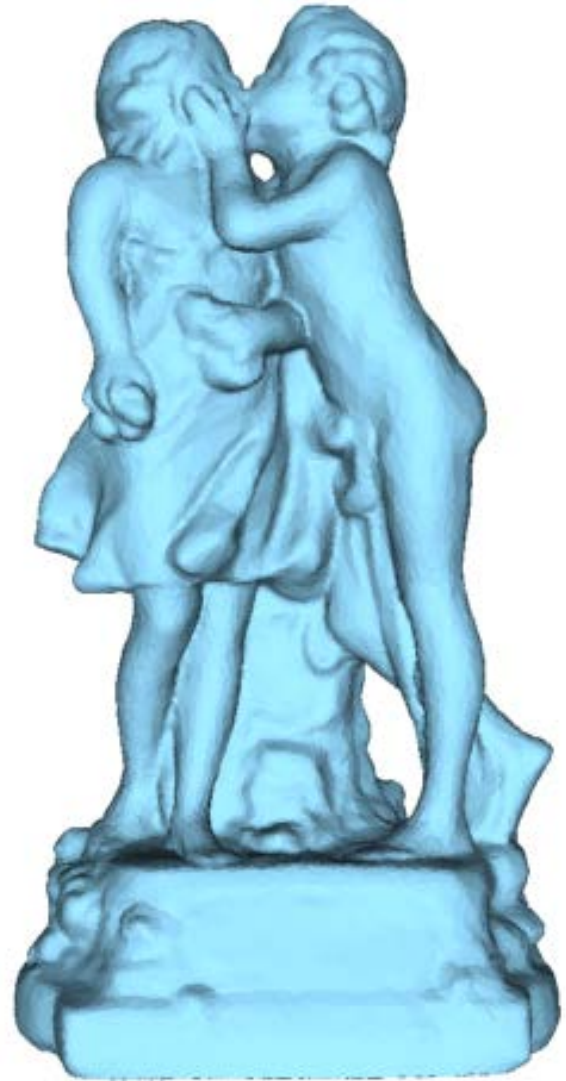
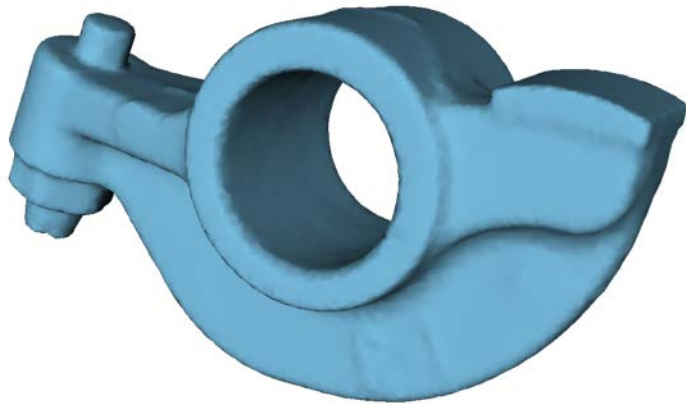
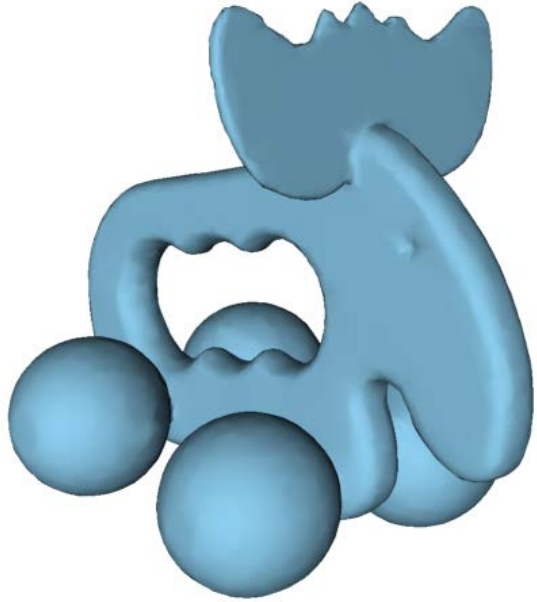


Genus 2



Genus 3

What is the Genus of...



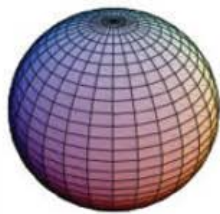
Euler Characteristic

For a triangle (or polygonal) mesh (without boundary)
Euler's formula

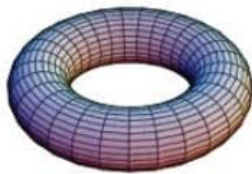
$$V - E + F = 2(1 - g)$$

relates the number of vertices V , edges E , faces F ,
and the **genus** g

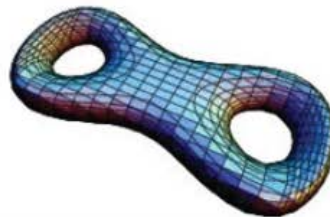
The term $2(1 - g)$ is called the **Euler characteristic** X



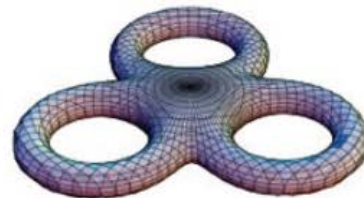
Genus 0



Genus 1

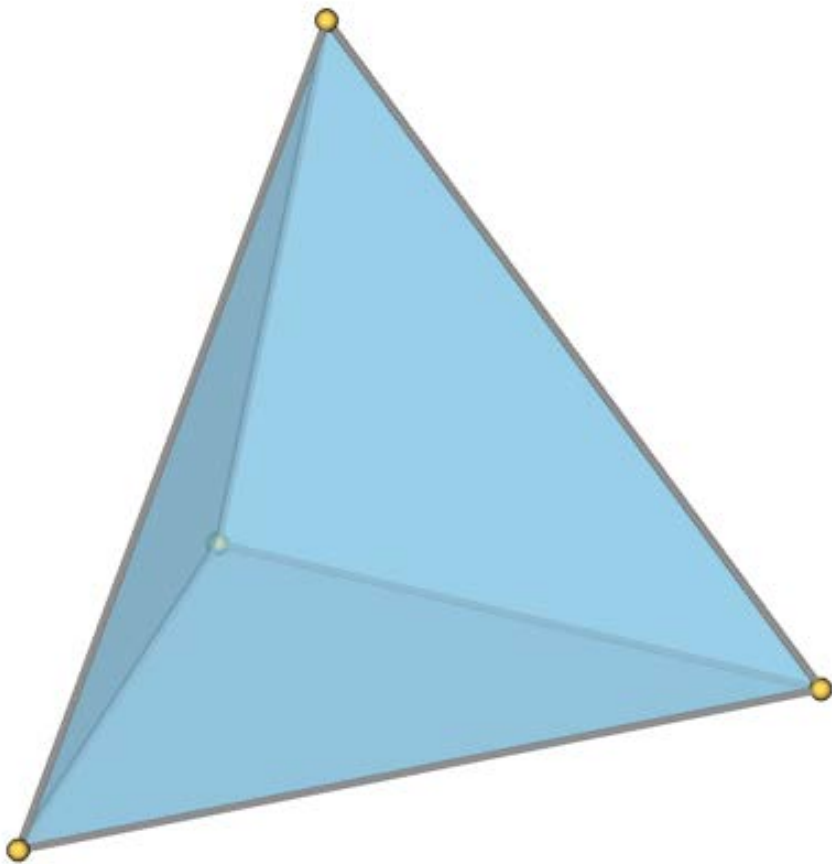


Genus 2



Genus 3

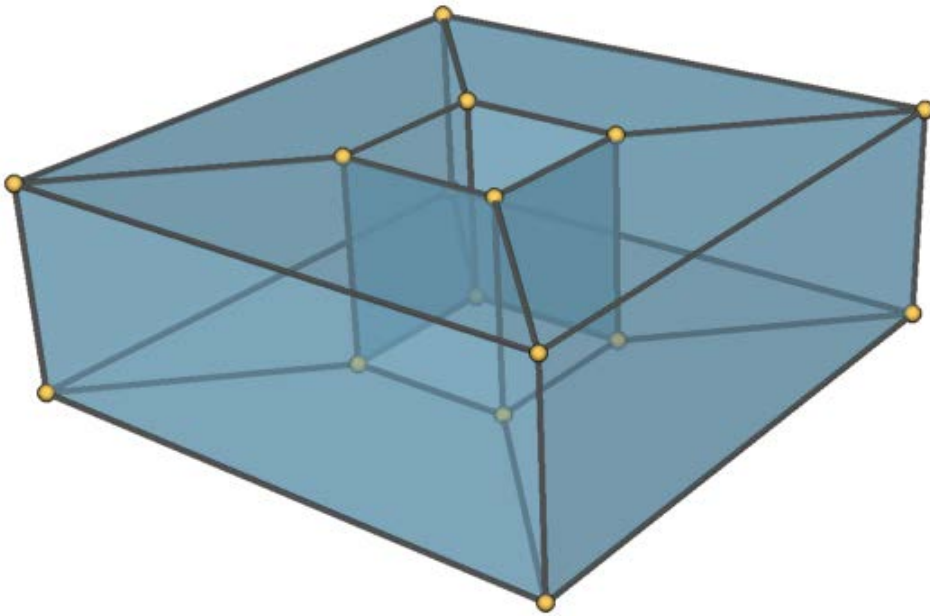
Example: Genus 0



$$V - E + F = 2(1 - g)$$

$$4 - 6 + 4 = 2(1 - 0)$$

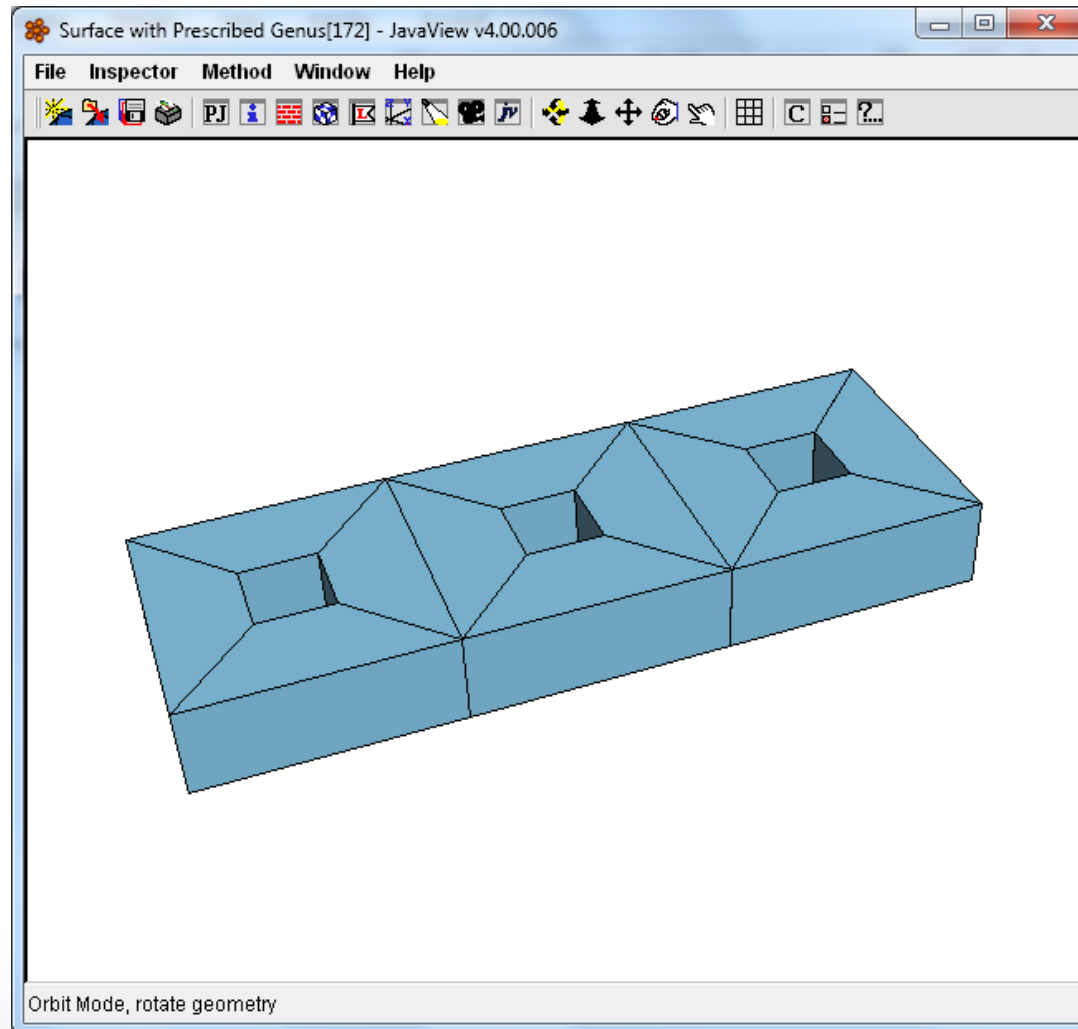
Example: Genus 1



$$V - E + F = 2(1 - g)$$

$$16 - 32 + 16 = 2(1 - 1)$$

Genus Builder



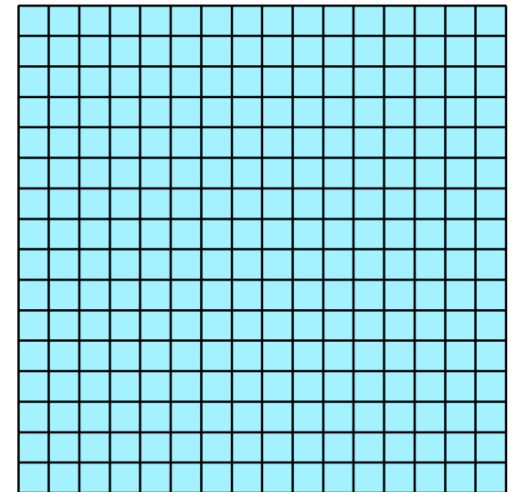
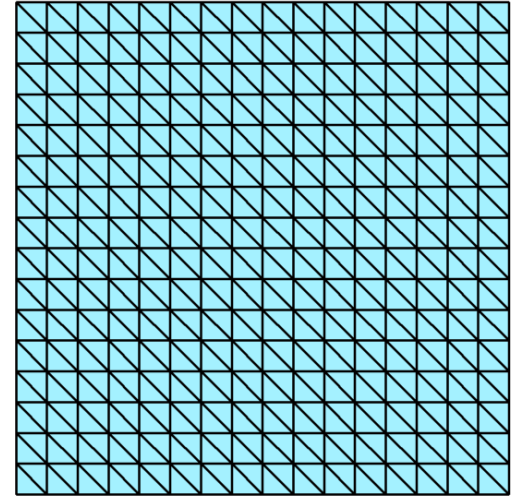
Average Valences

Valence of a vertex

- Number of adjacent edges

Average valence (for large number of vertices)

- Triangle meshes: 6
- Quad meshes: 4



Average Valences

Why?

- For triangles: $3F = 2E$ (every triangle has 3 edges and every edge is in 2 triangles)

- Euler Formula

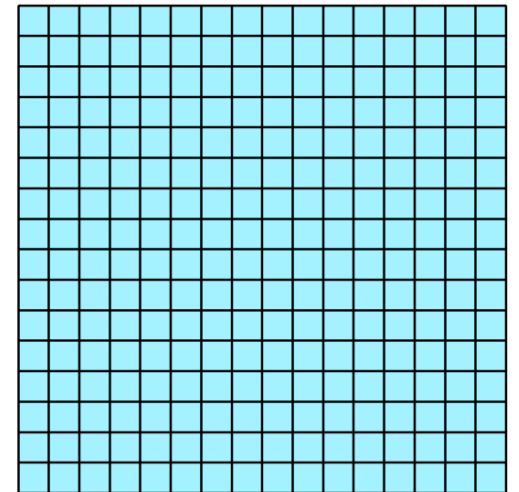
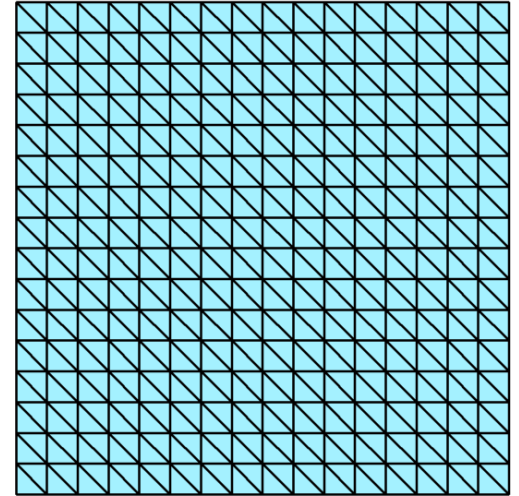
$$V + F - E = V + 2E/3 - E = 2 - 2g$$

- Solving for E

$$E = 3(V - 2 + 2g)$$

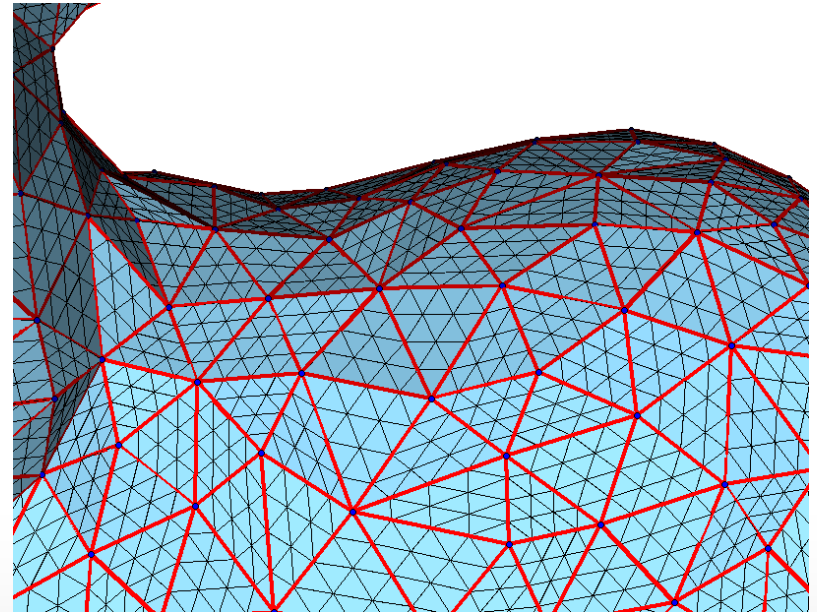
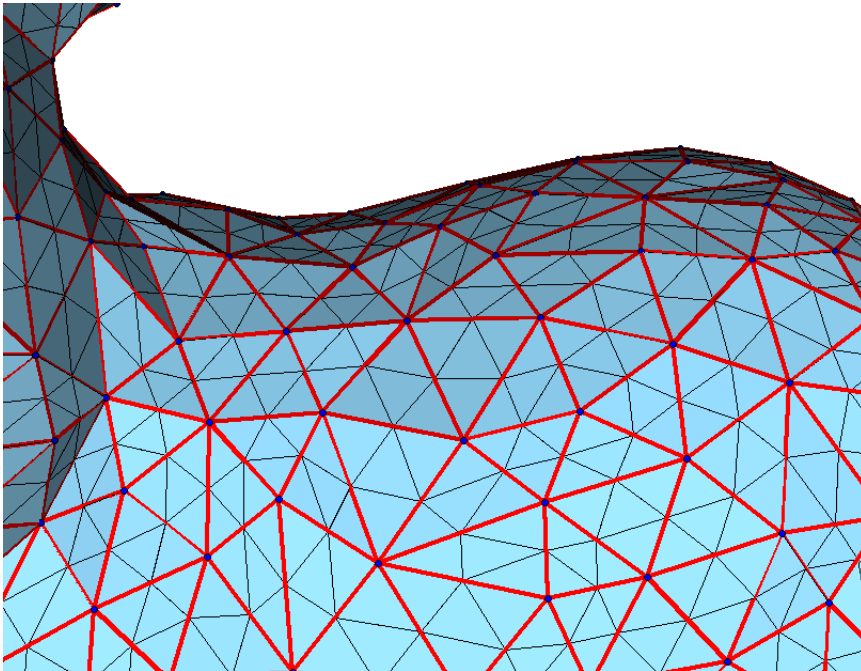
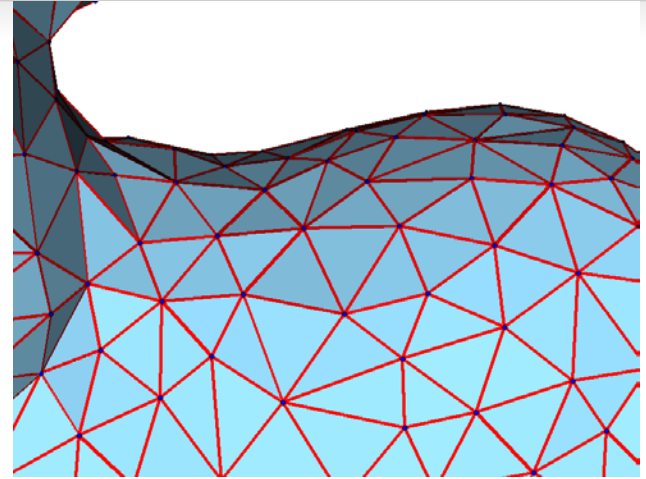
- For large V and small g , we get

$$\frac{2E}{V} = \frac{6(V - 2 + 2g)}{V} \approx 6$$



Example: One-to-Four Refinement

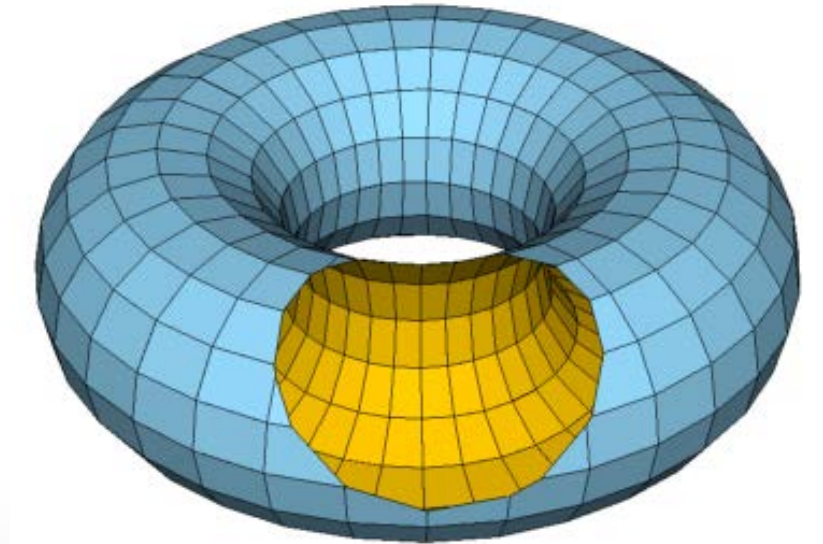
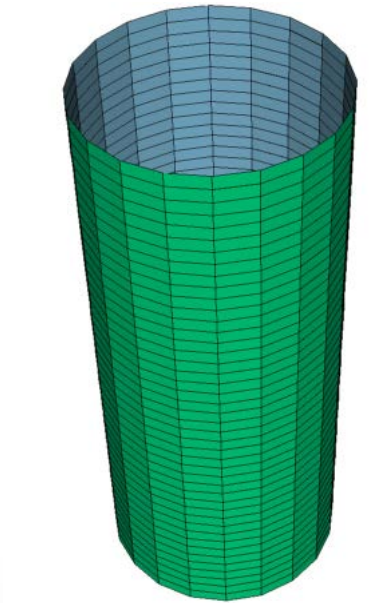
All new vertices have valence 6



Orientable Surfaces

Orientable

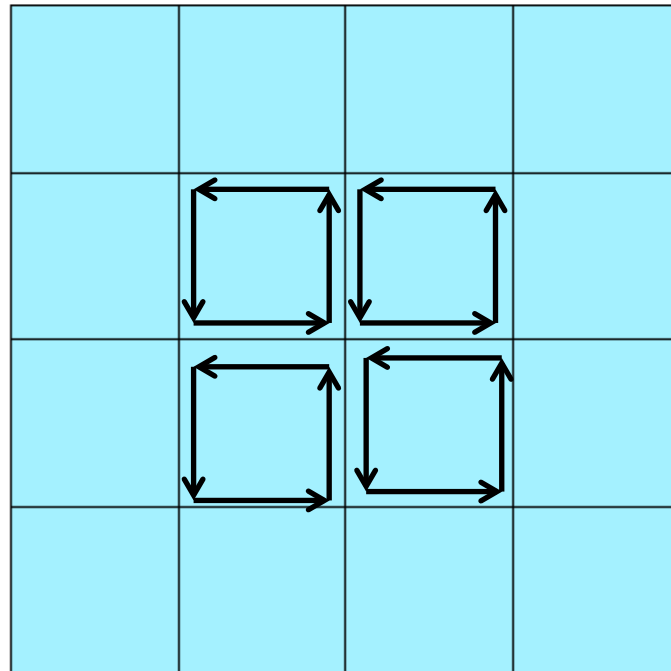
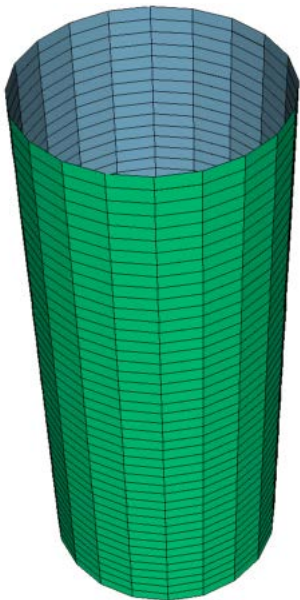
- Surface has a consistent normal field
- “Can color the inside and outside with different colors”



Orientable Surfaces

Orientable

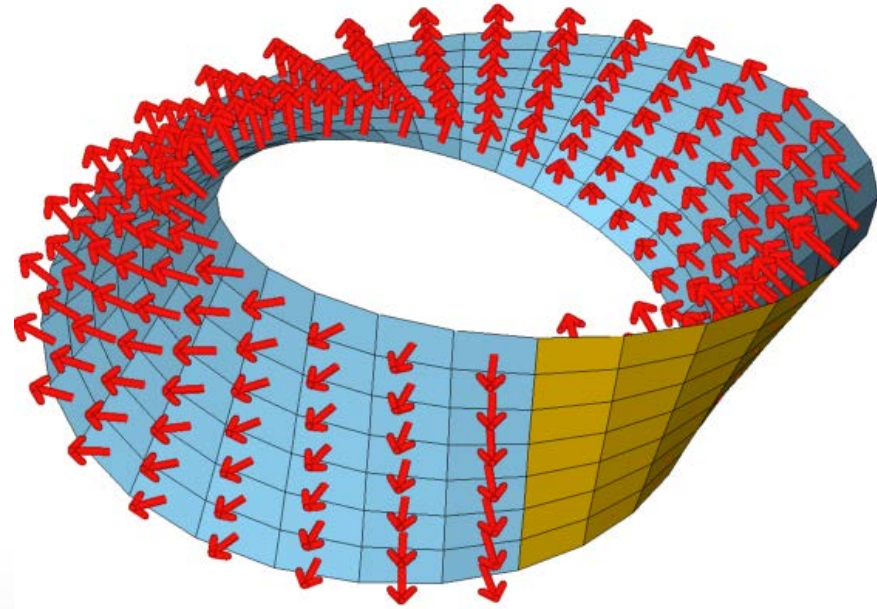
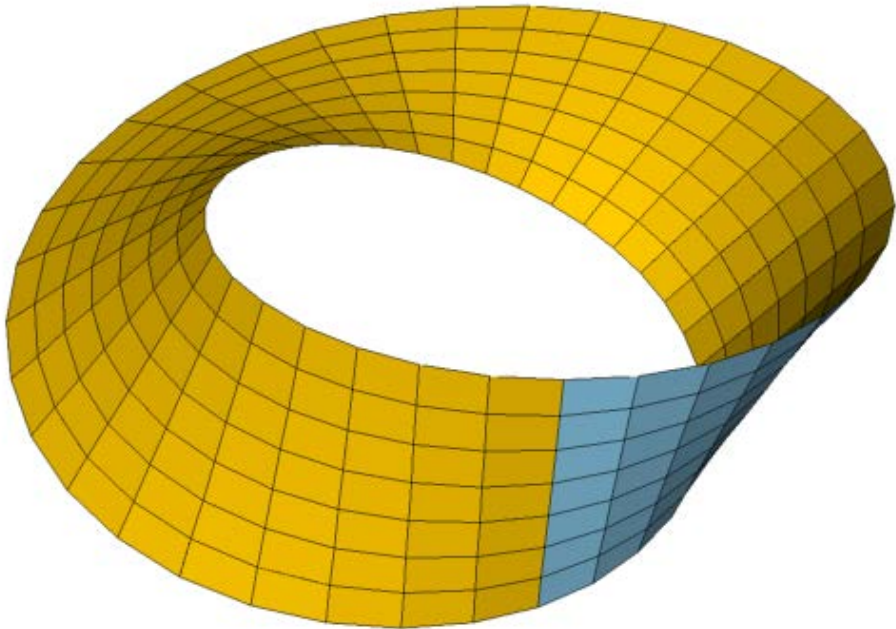
- Can be decided from the face array (vertex positions not needed)
 - Can all edges can be oriented consistently?



Non-Orientable Surfaces

Möbius Strip

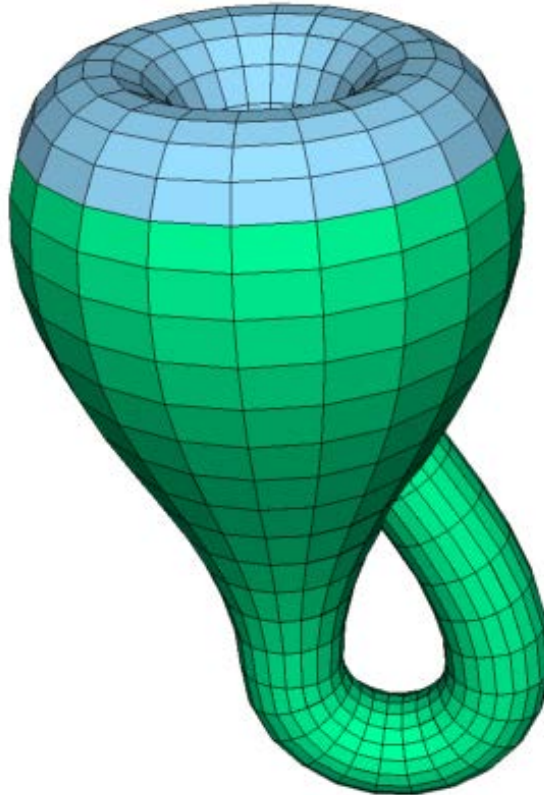
- Dangerous, algorithms may crash
- Example: How do you compute vertex normals?



Non-Orientable Surfaces

Klein Bottle (Klein's surface)

- Euler characteristic: $X = 0$



Costa-Hoffman-Meeks Surface

Orientable?

