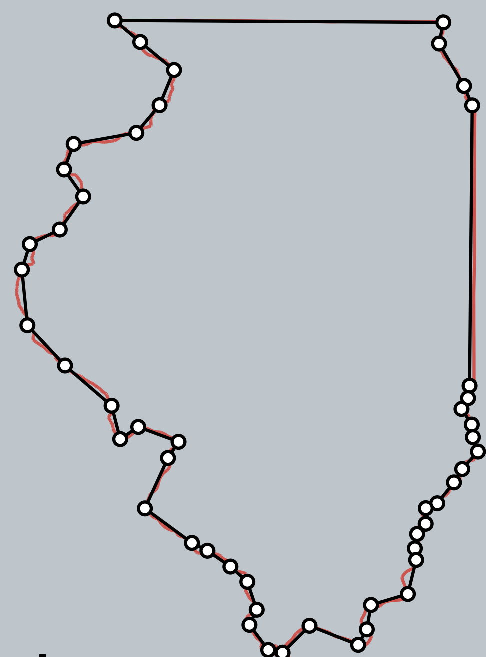
Surface Approximation for Computer Graphics



Michael Garland
University of Illinois at Urbana–Champaign

The Situation

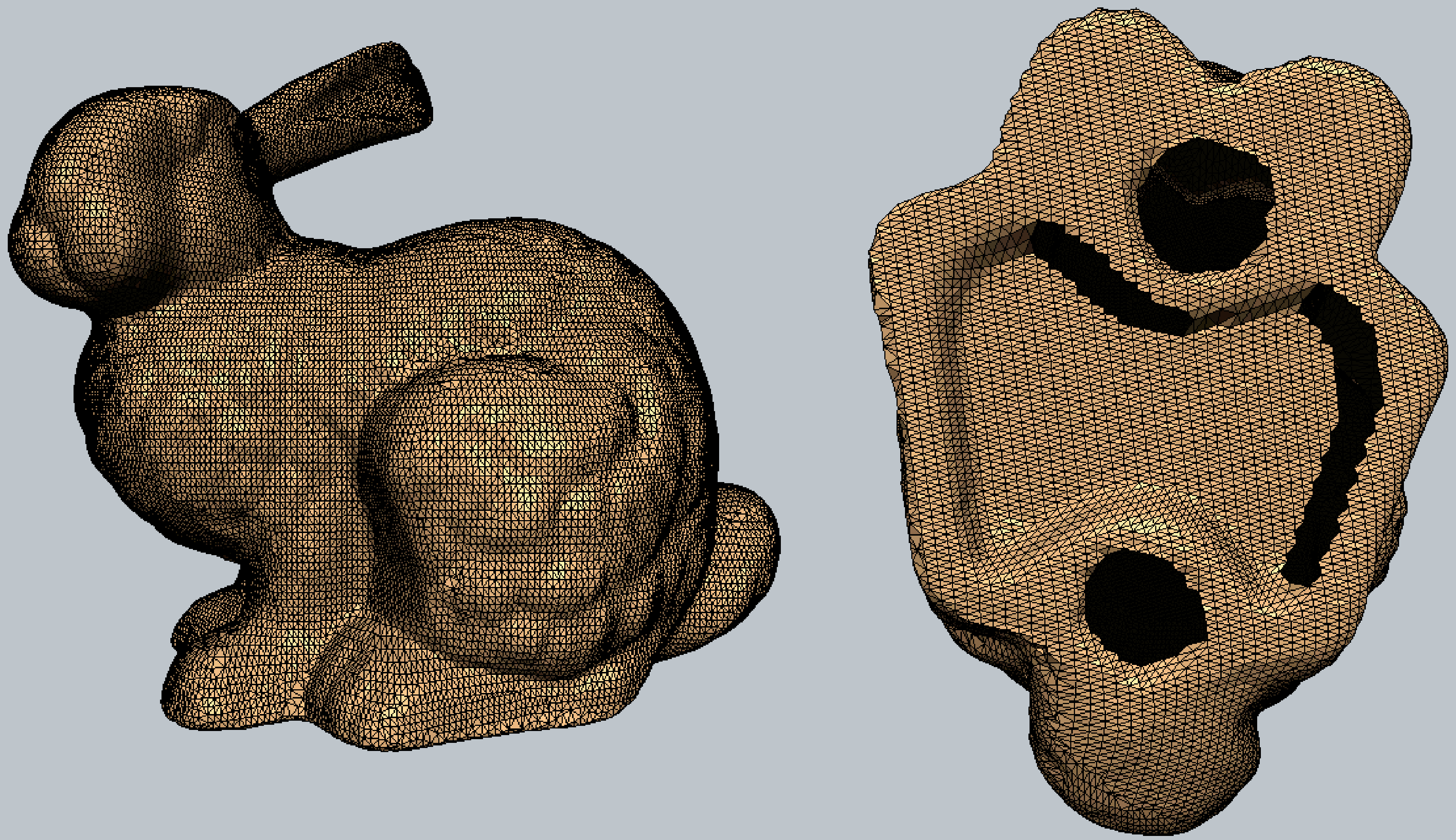
- Resources are limited
  - disk, RAM, GPU, CPU, dpi, ...
- Many meshes are “bad”
  - too many triangles
  - triangles that poorly sample the surface

Our World-View

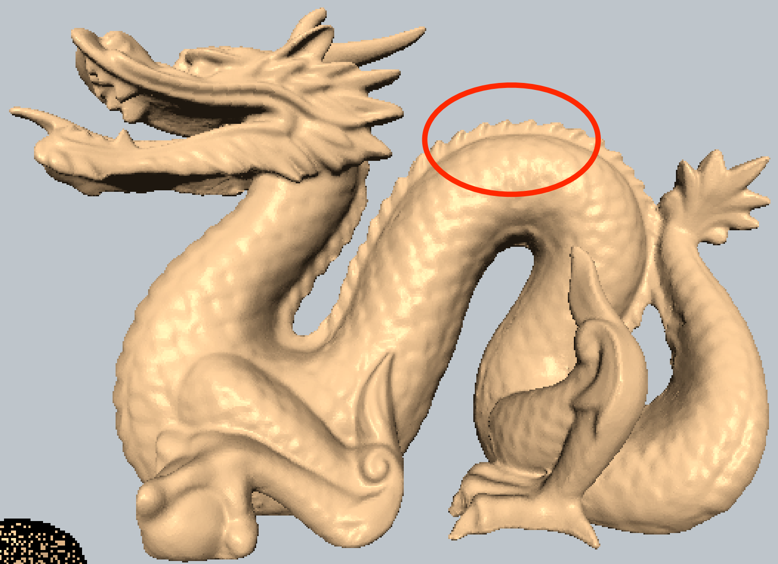
- *Interactive display* is the ultimate application
    - want something like 30 frames per second
- Polygons much smaller than a pixel don't really matter

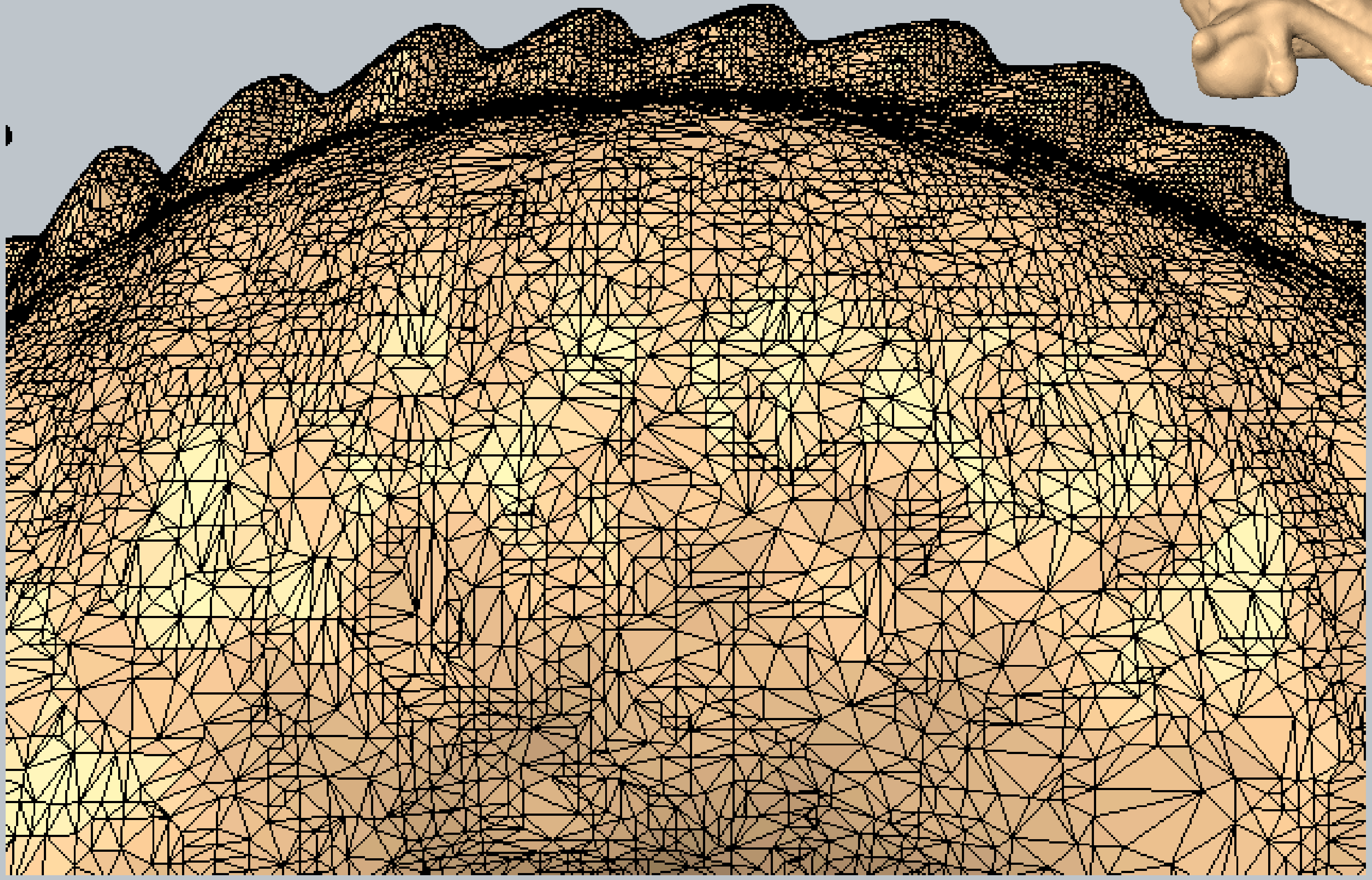
The Problem

Blindly Regular Meshes

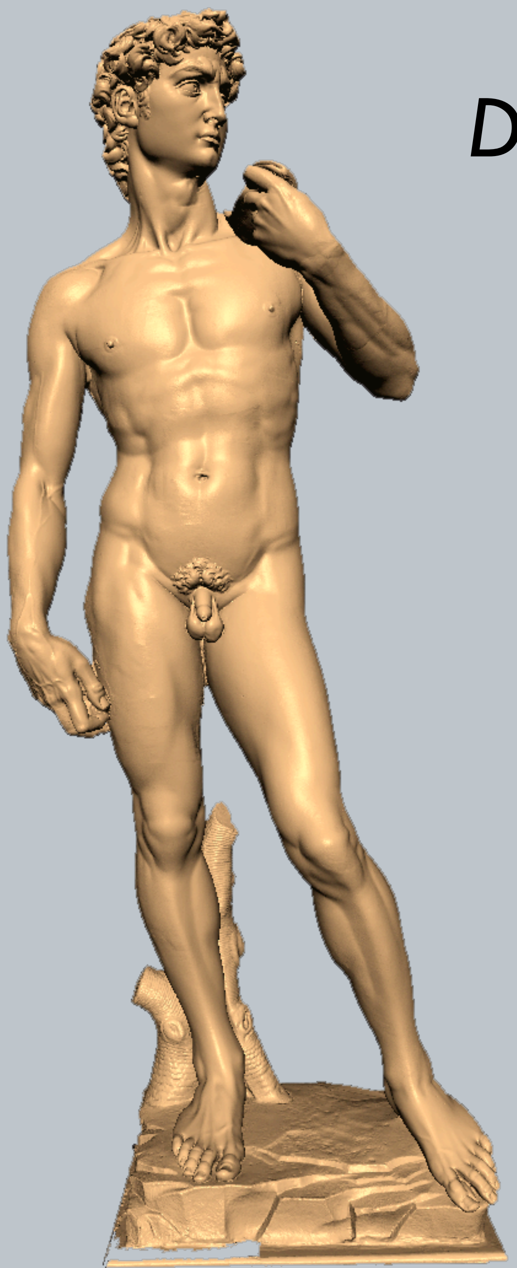


Irregular Meshes





Massive Meshes

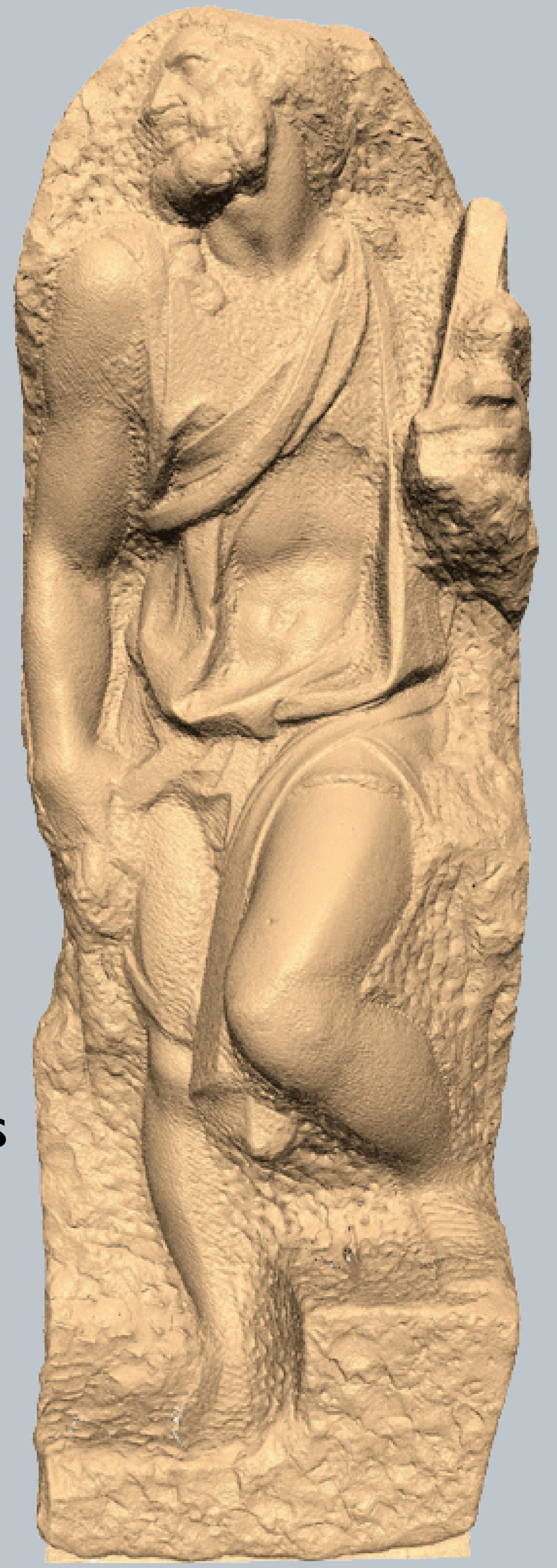


David

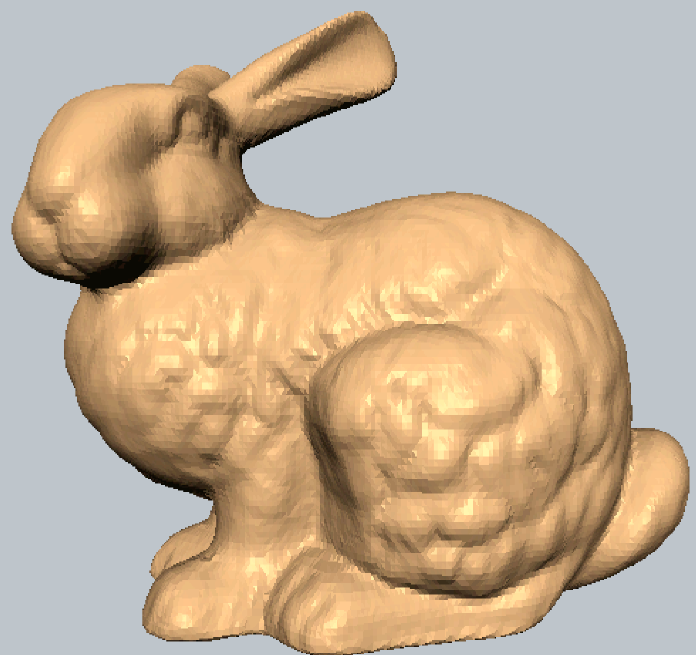
57 million triangles
1.4 Gigabytes
1 mm resolution

St. Matthew

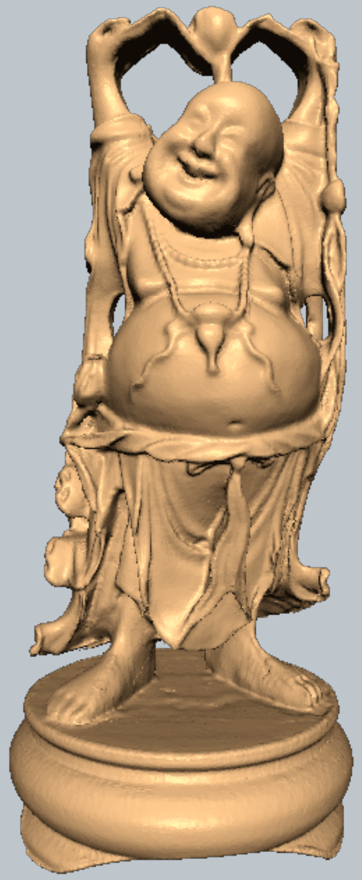
300 million triangles
6.5 Gigabytes
¼ mm resolution



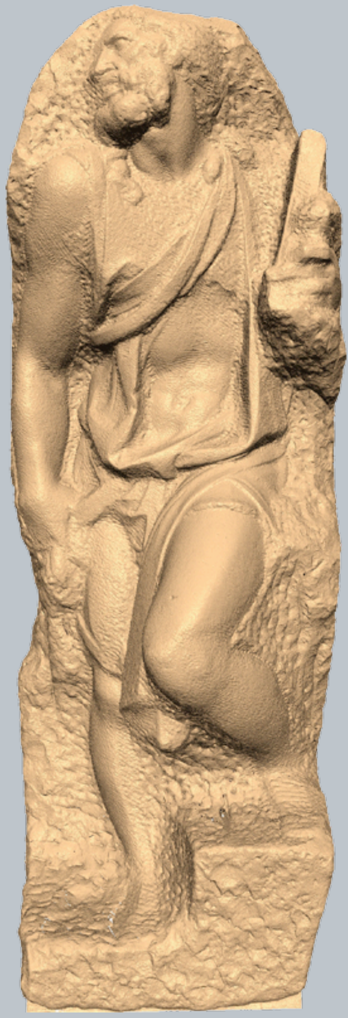
Historical Aside: What is Big?



1995
70,000 triangles

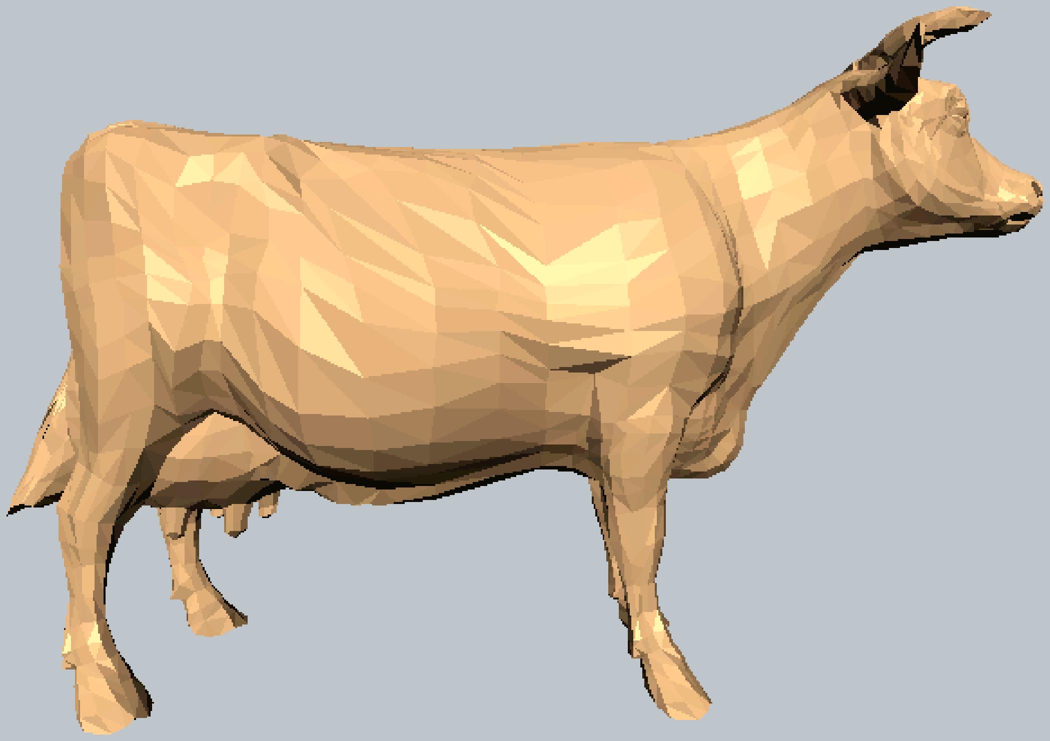


1997
1 million triangles

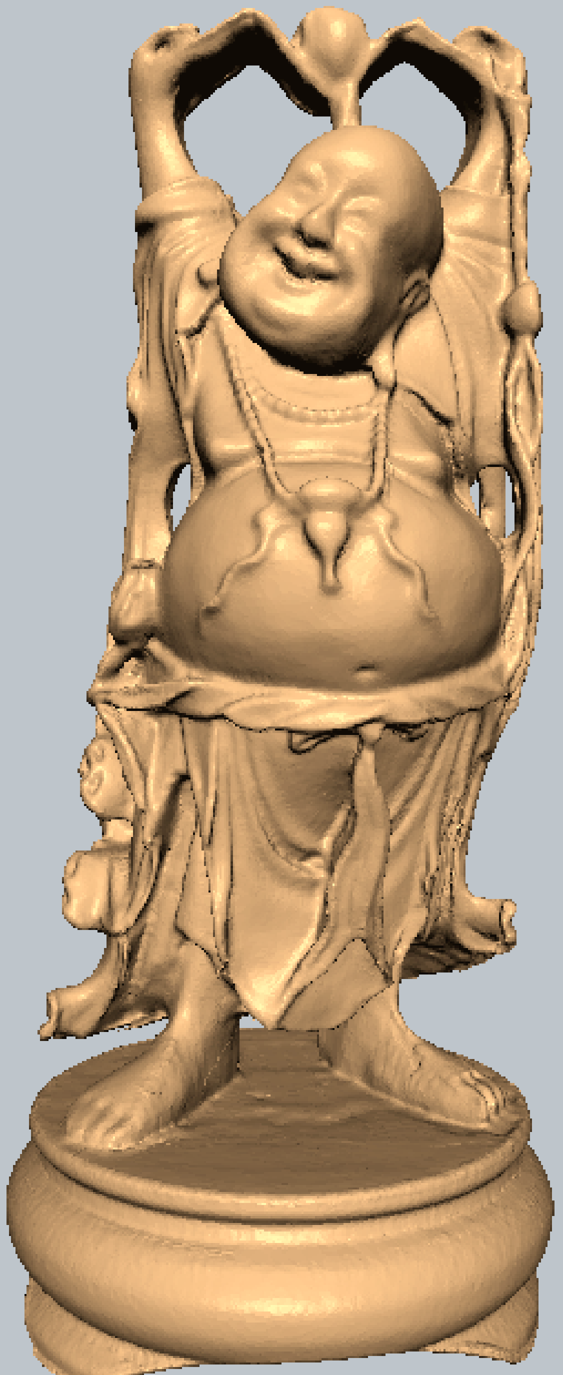


2000
300 million triangles

Bad Topology

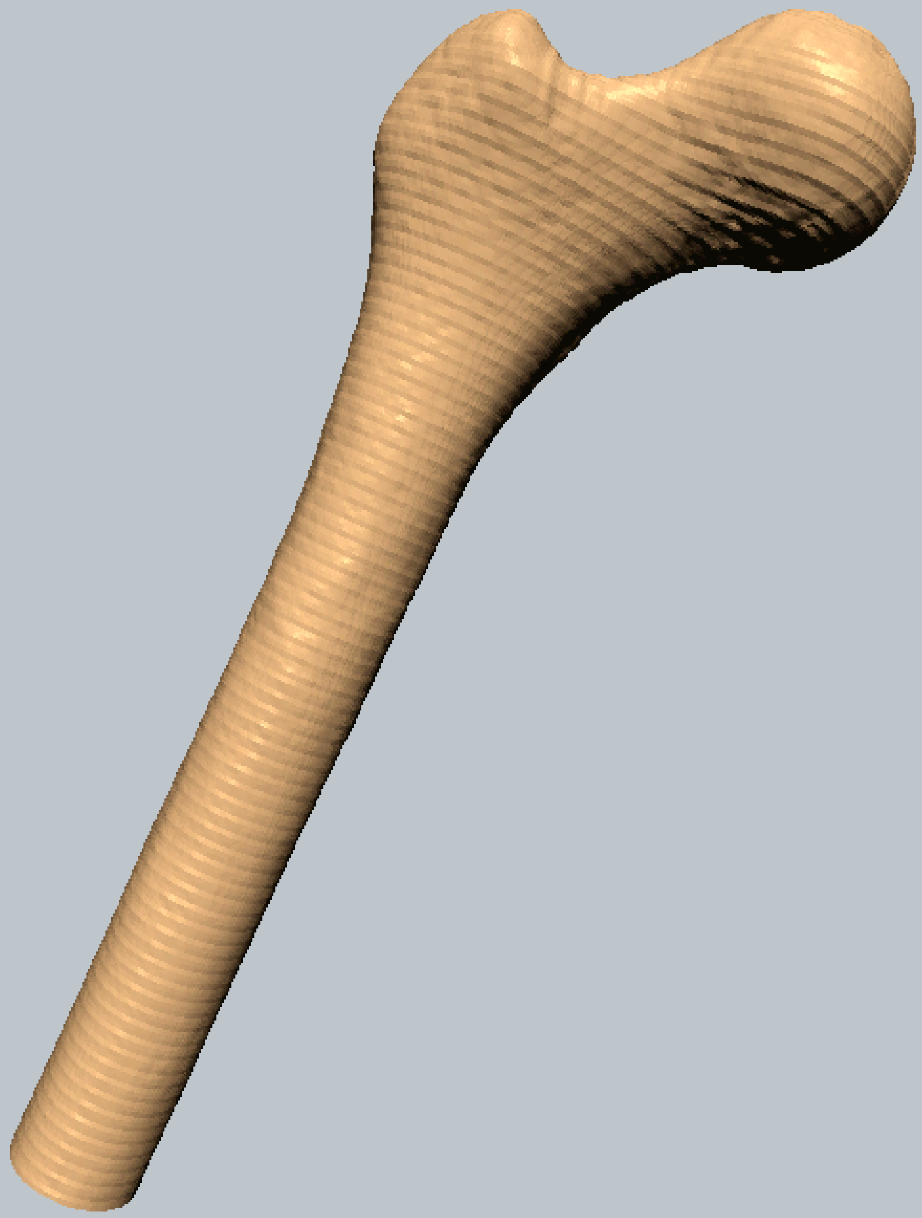


Not *quite* manifold



Artificially high genus

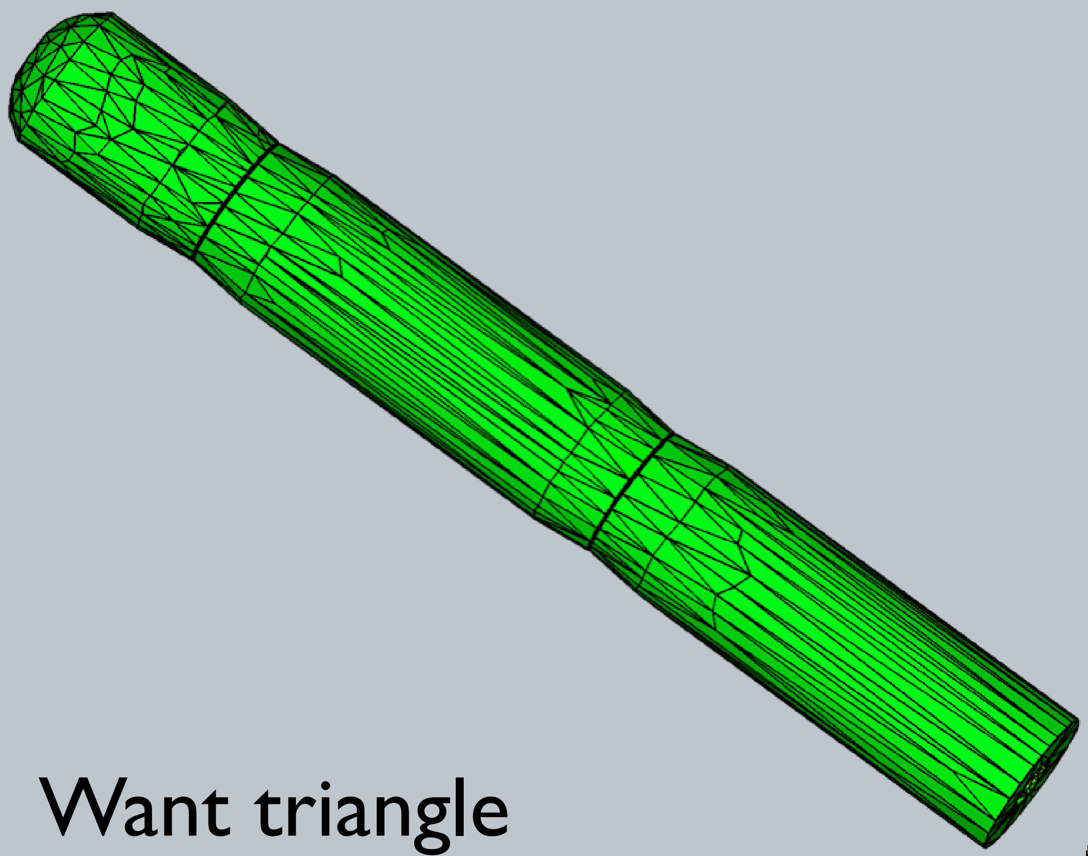
Noise & Grid Artifacts



What do we want?

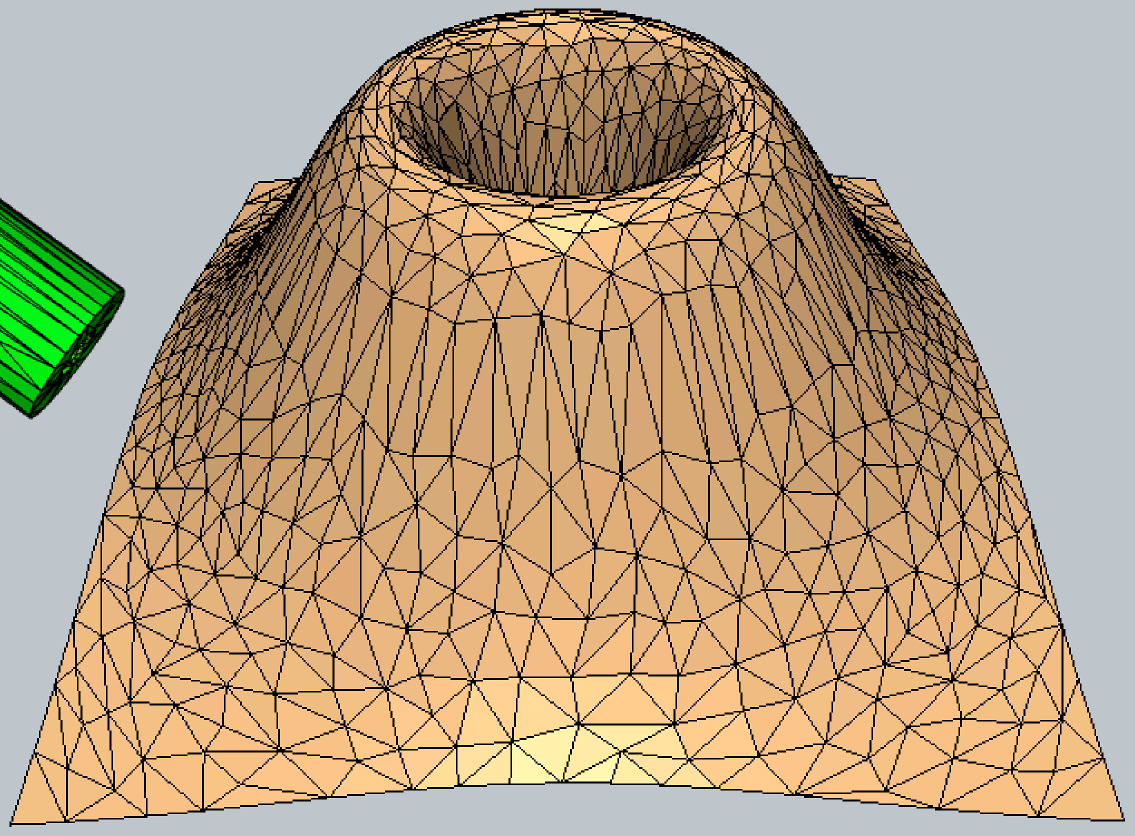
- Only as many triangles as really necessary
- Tessellation that adapts to geometry

Curvature adaptation



Want triangle aspect ratio of:

$$\rho = \sqrt{\left|\frac{\kappa_2}{\kappa_1}\right|}$$



Hierarchical Surfaces

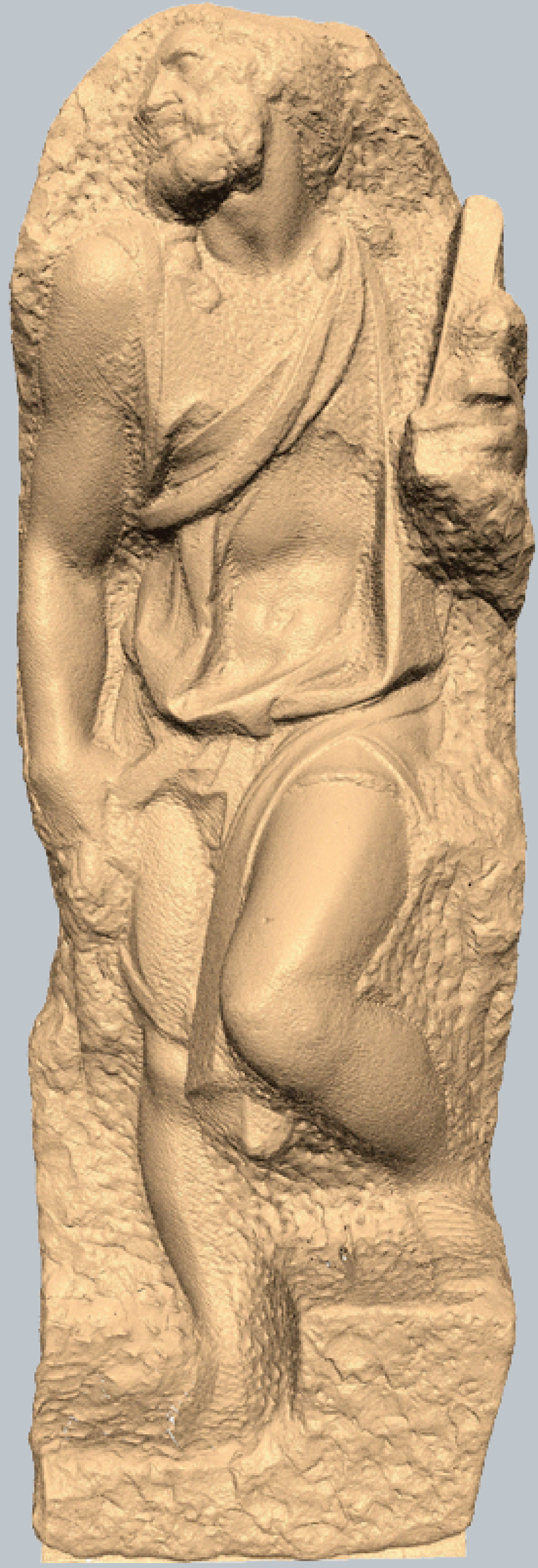
172,974 vertices

20,000 vertices

5000 vertices

Out-of-core Processing

- 6.5 GB of raw data
- Impractical to load into RAM

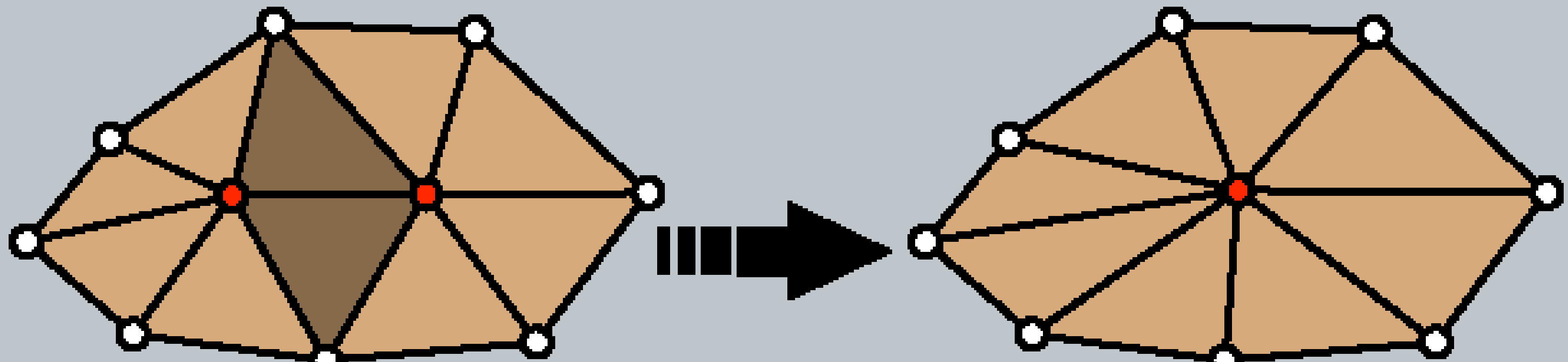


Surface Simplification

Simplification Framework

Greedy iterative contraction

1. rank all edges by cost metric
2. contract minimum cost edge
3. update edge costs



How to measure Cost?

- Want to allow points to move over surface
  - measure deviation from surface
  - this is expensive
- An approximation would require points to remain close to the local tangent planes

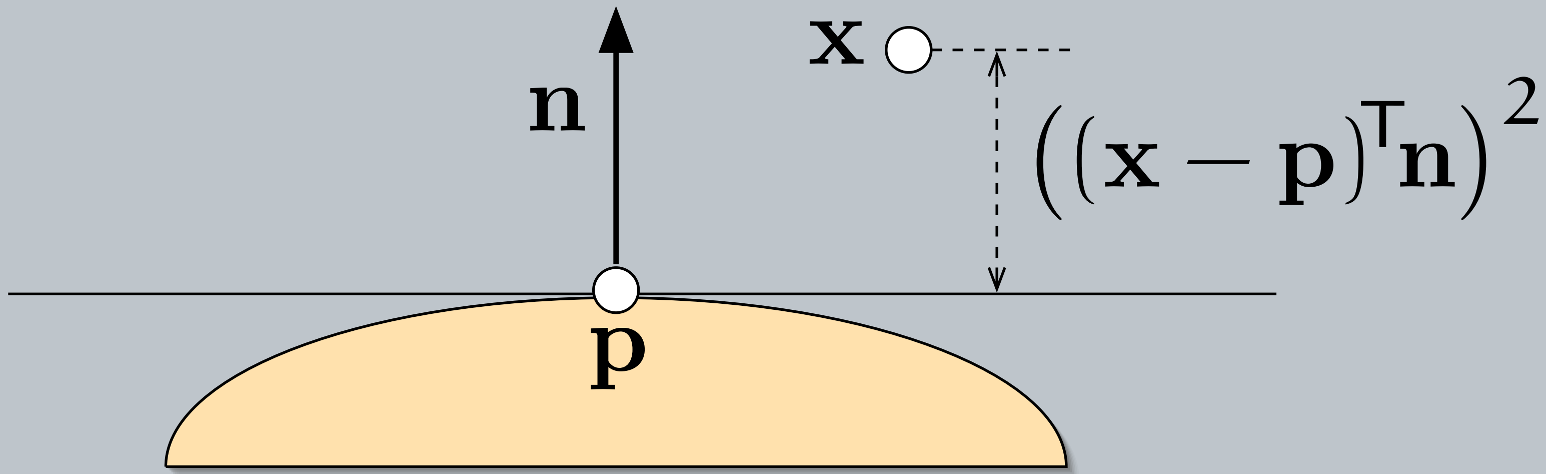
Quadric Error Metric

Given a tangent plane, we can define a *quadric*

$$Q = (\mathbf{A}, \mathbf{b}, c) = (\mathbf{n}\mathbf{n}^T, -\mathbf{A}\mathbf{p}, \mathbf{p}^T\mathbf{A}\mathbf{p})$$

Measuring the squared distance of point to the plane

$$Q(\mathbf{x}) = \mathbf{x}^T\mathbf{A}\mathbf{x} + 2\mathbf{b}^T\mathbf{x} + c$$



Quadric Error Metric

Sum of quadrics represents a set of planes

$$\sum_i \left((\mathbf{x}-\mathbf{p}_i)^\mathsf{T}\mathbf{n}_i\right)^2 = \sum_i Q_i(\mathbf{x}) = \left(\sum_i Q_i\right)(\mathbf{x})$$

Any set of vertices has an associated quadric

- sum over faces with corner in set
- find representative minimizing error

$$\nabla Q(\mathbf{x}^\star) = 0 \qquad \mathbf{A}\mathbf{x}^\star = -\mathbf{b}$$

Algorithm Overview

- *Initialization*
  - Compute quadric for each vertex from planes of surrounding faces
- *Iterative Contraction*
  - Repeatedly contract the edge of least cost
  - Add quadrics when contracting edges

Sample Results

172,974 vertices

20,000 vertices

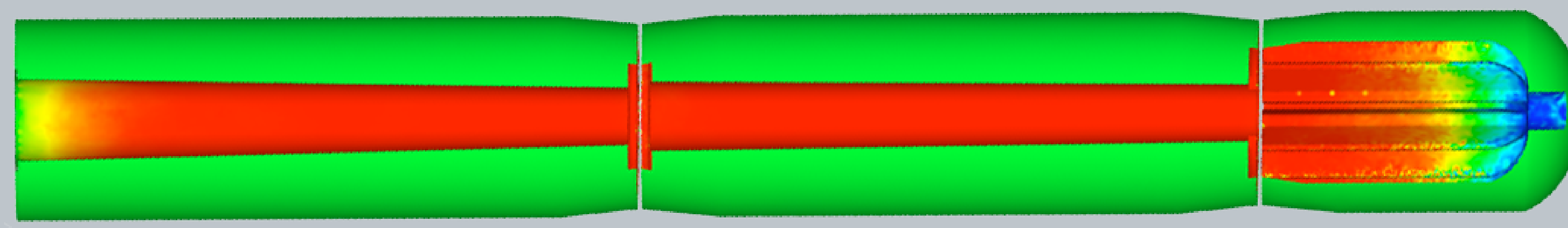
5000 vertices

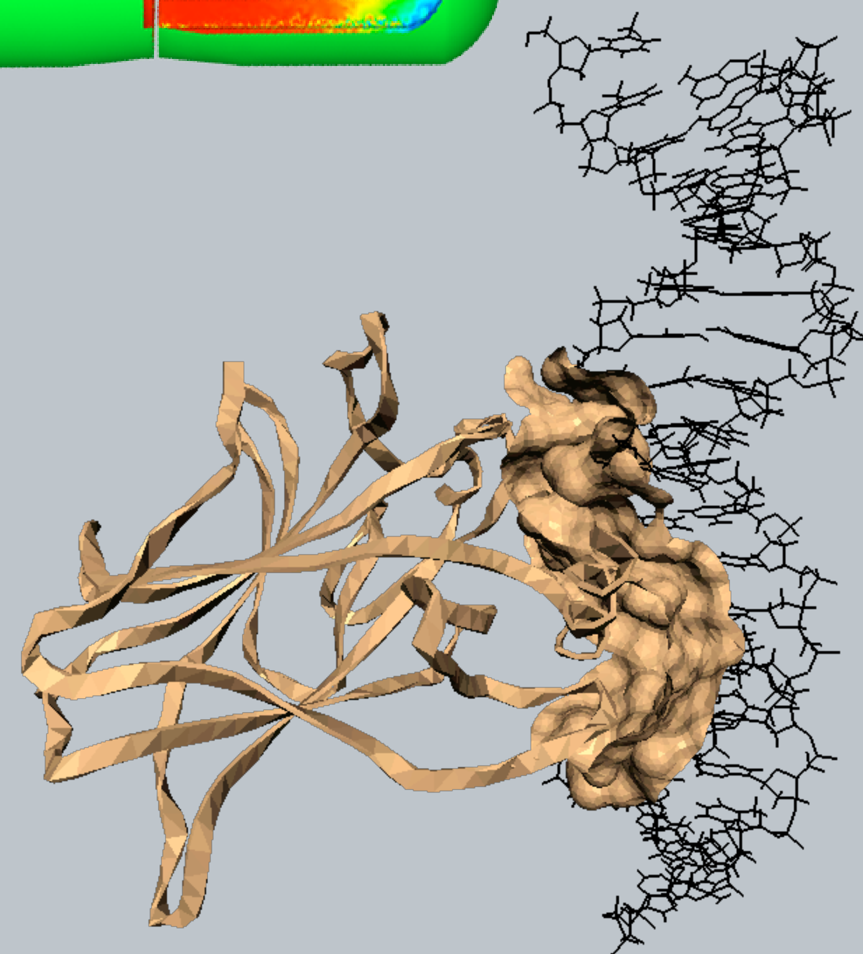
Less than 13 seconds on 1GHz Pentium III

Demo

Generalized Simplification

Beyond Triangulated Surfaces





Quadric Error Metric

Any *d*-simplex has a tangent space of dimension *d* with orthonormal tangent vectors $\mathbf{e}_1, \ldots, \mathbf{e}_d$

Squared distance of a point to this tangent space:

$$(\mathbf{x} - \mathbf{p})^{\mathsf{T}} \left(\mathbf{I} - \sum_{i=1}^{d} \mathbf{e}_i \mathbf{e}_i^{\mathsf{T}} \right) (\mathbf{x} - \mathbf{p})$$

Quadric Error Metric

Squared distance of a point to this tangent space:

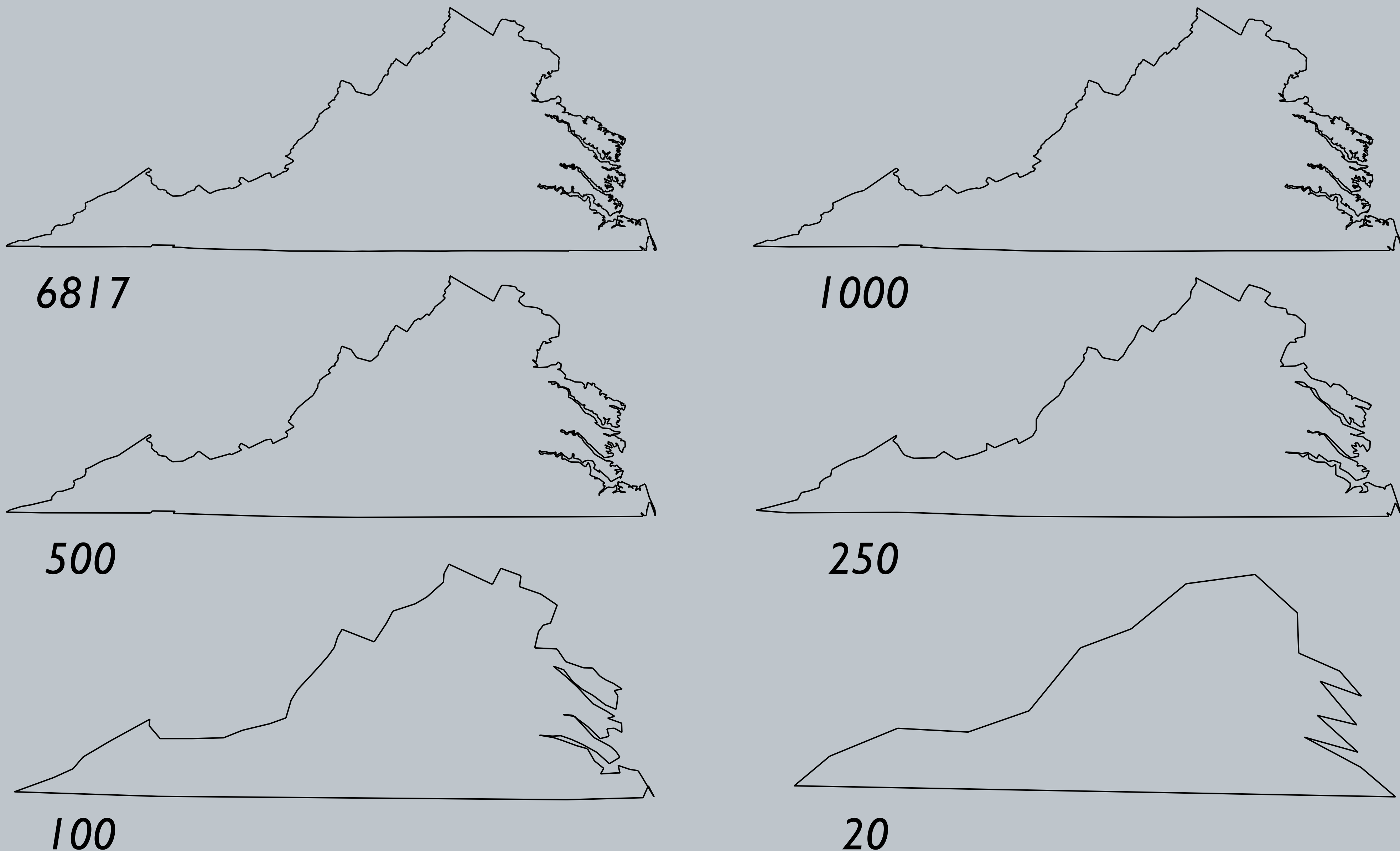
$$Q(\mathbf{x}) = \mathbf{x}^{\mathsf{T}}\mathbf{A}\mathbf{x} + 2\mathbf{b}^{\mathsf{T}}\mathbf{x} + c$$

where

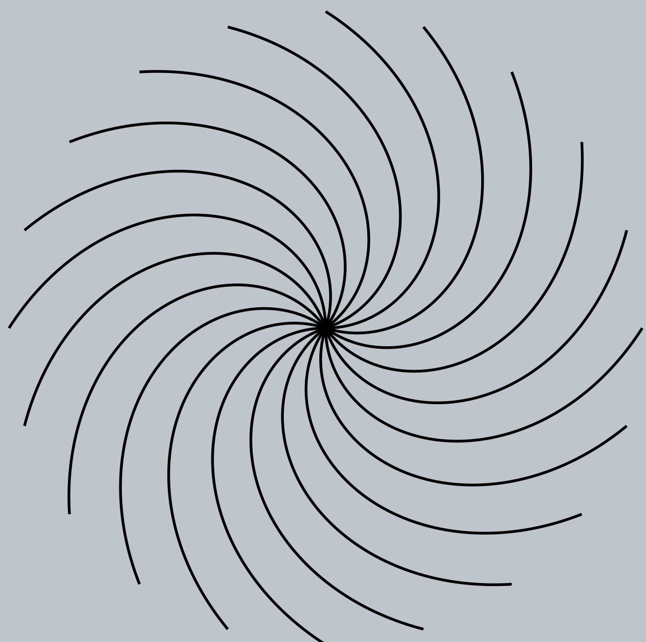
$$\mathbf{A} = \mathbf{I} - \sum_{i=1}^{d} \mathbf{e}_i \mathbf{e}_i^{\mathsf{T}} \qquad \mathbf{b} = -\mathbf{A}\mathbf{b} \qquad c = \mathbf{p}^{\mathsf{T}}\mathbf{A}\mathbf{p}$$

True of any *d*-simplex in any Euclidean *n*-space

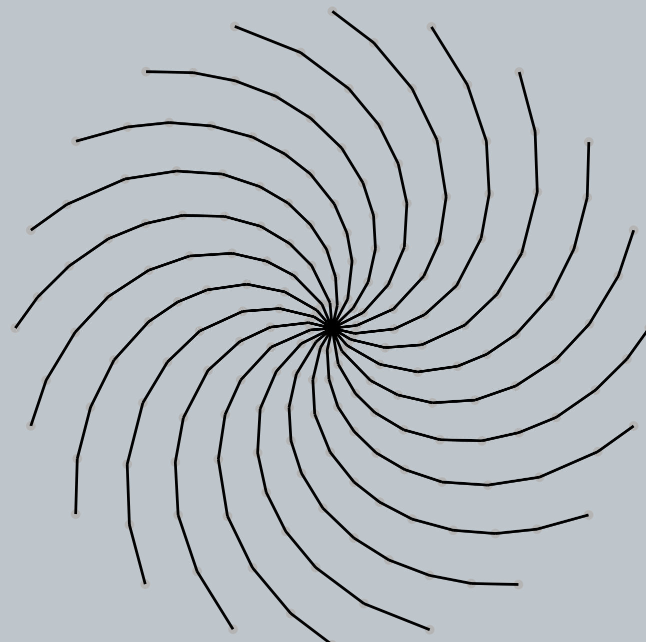
Plane Curves



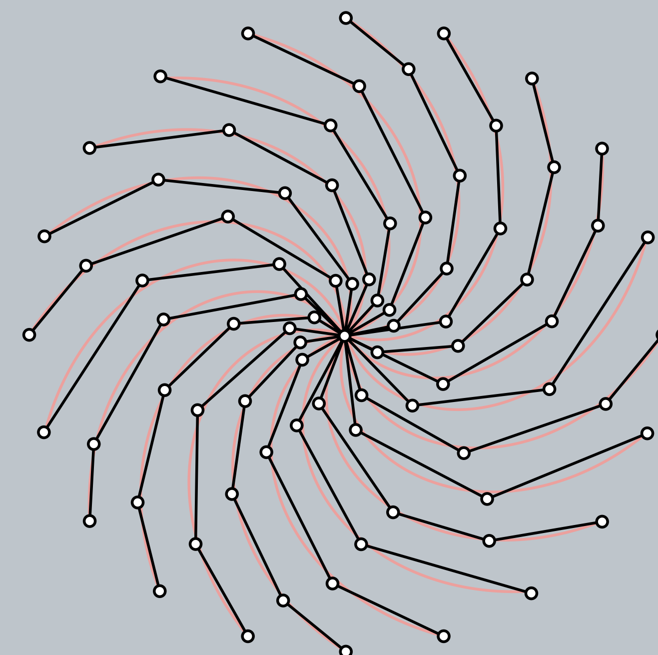
Non-manifold Plane Curve with Boundary



20,001 vertices

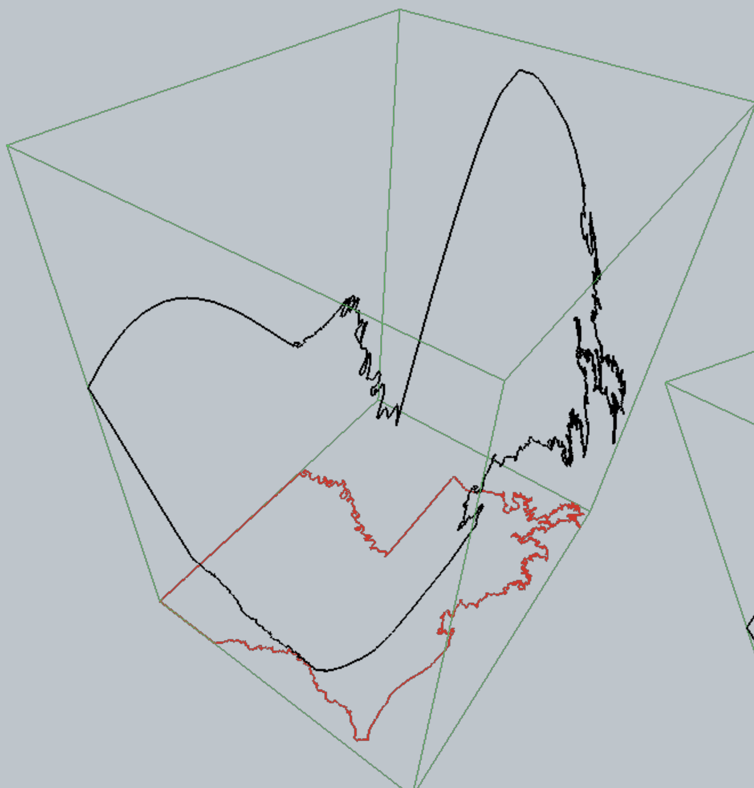


201 vertices



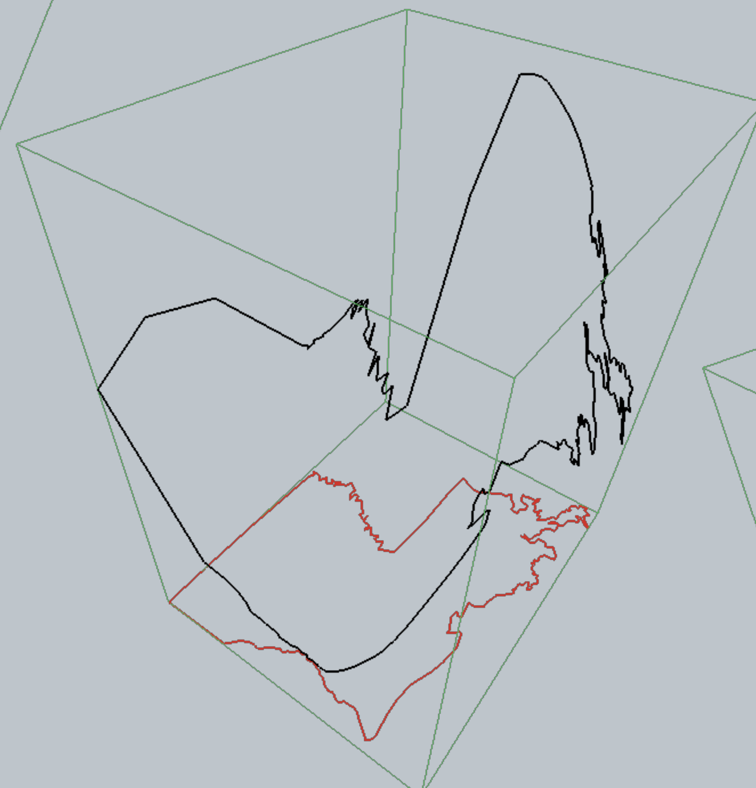
81 vertices

Space Curves

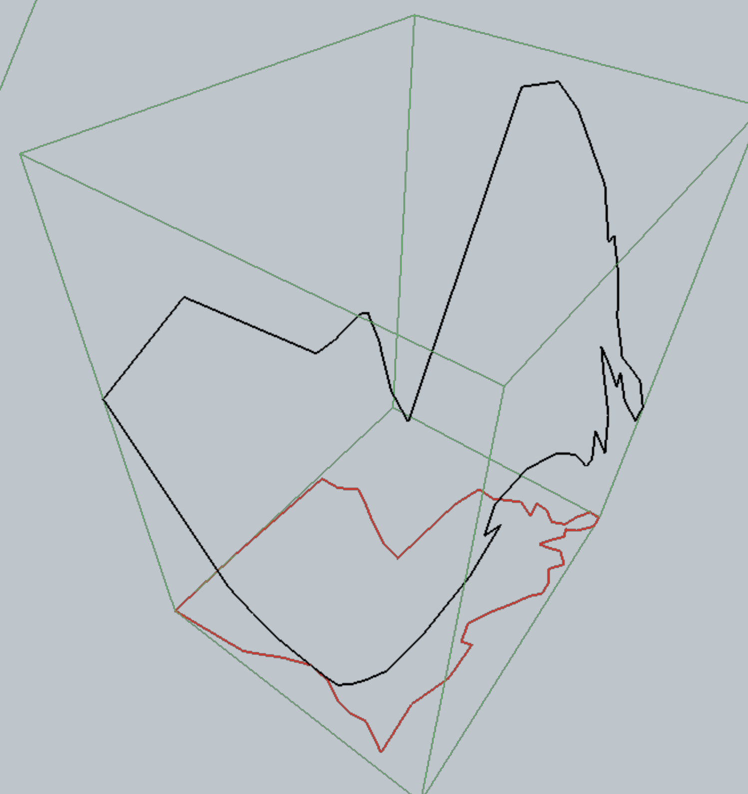


Original
8234 vertices

250 vertices

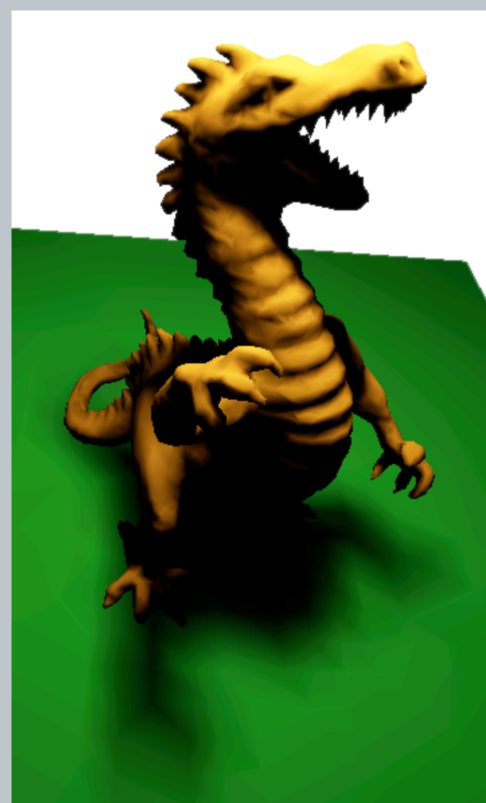


50 vertices



Surfaces with Attributes

50,000

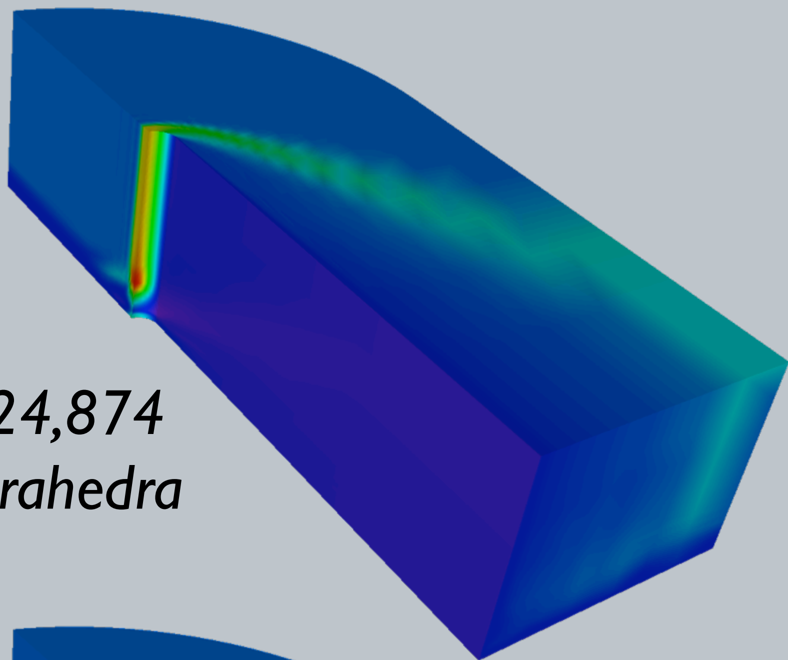


10,000

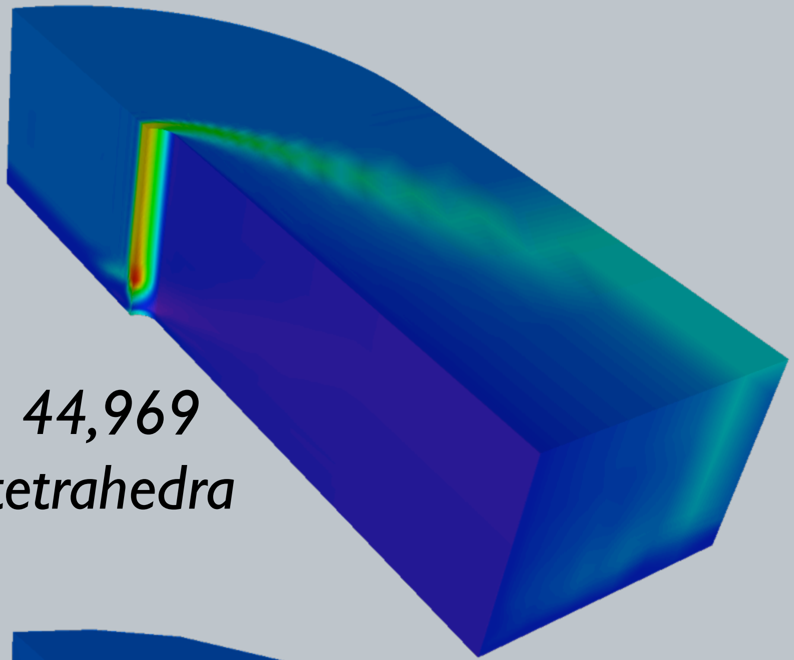
3000

1500

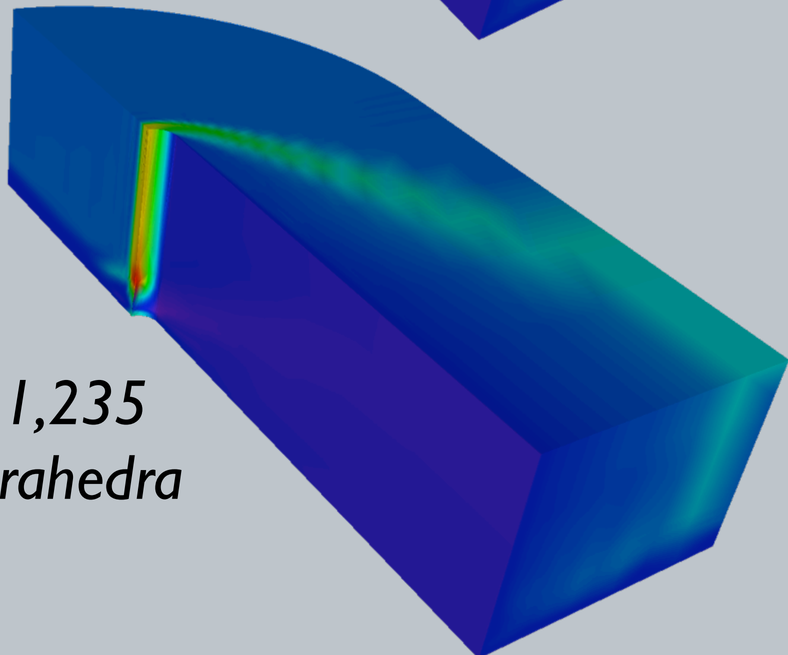
Blunt Fin CFD Data



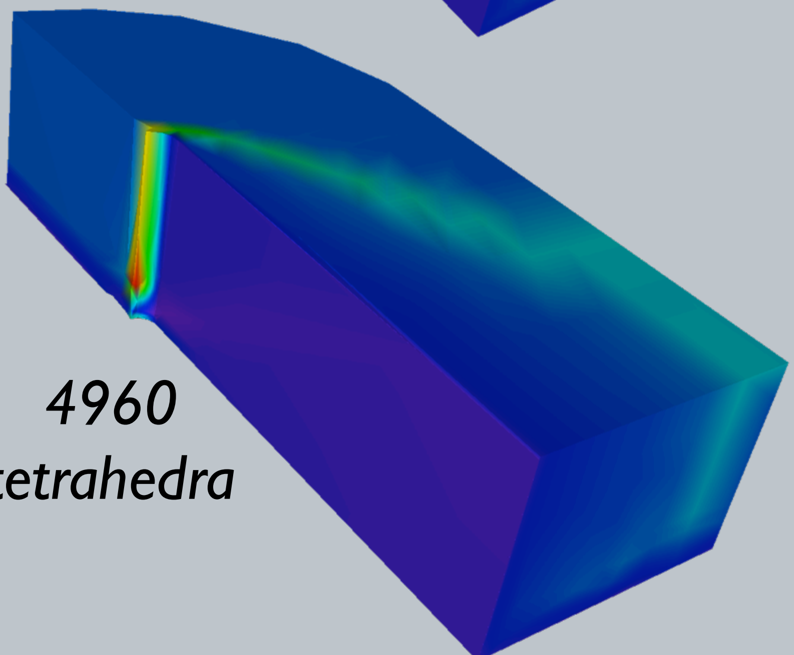
224,874 tetrahedra



44,969 tetrahedra

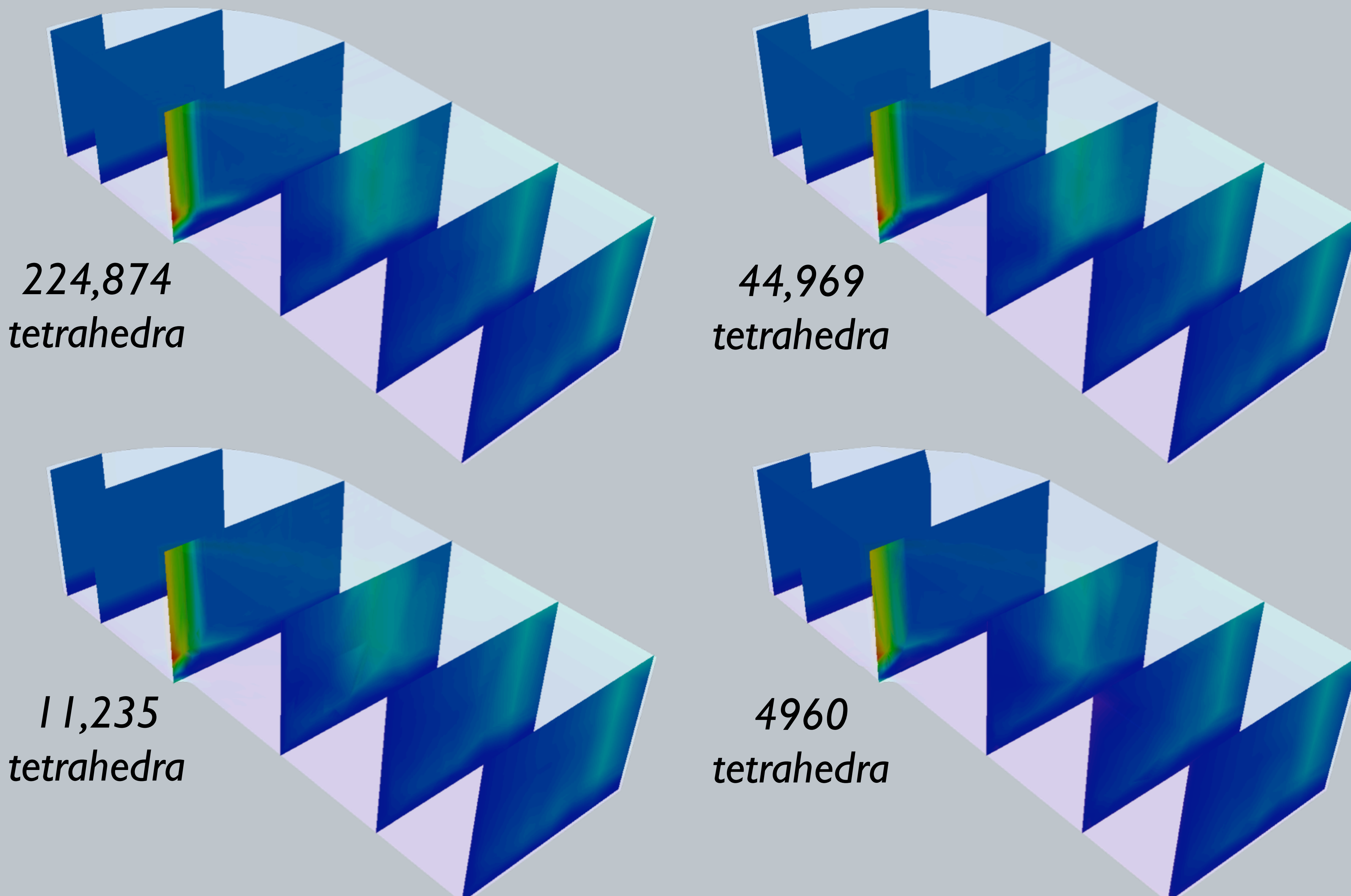


11,235 tetrahedra

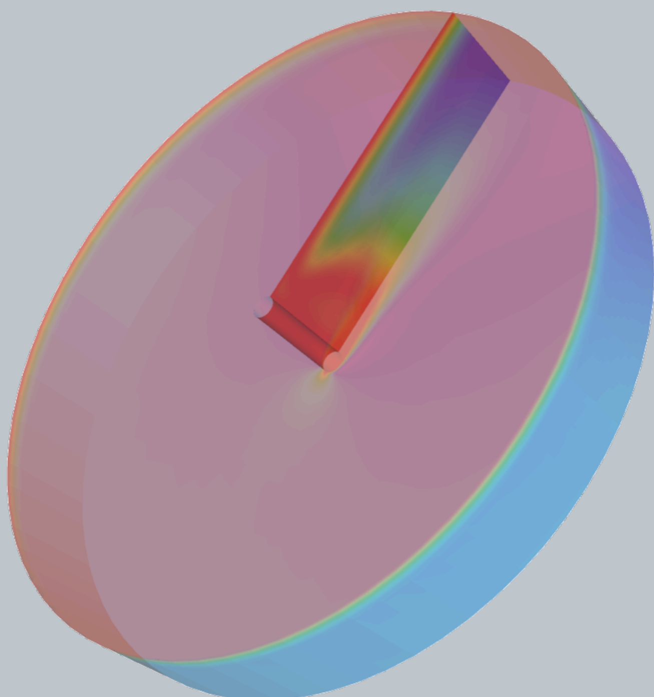


4960 tetrahedra

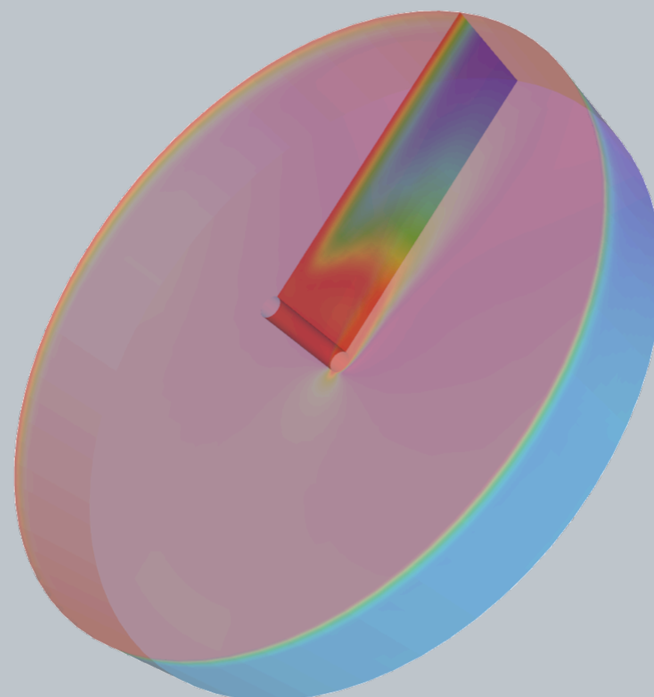
Blunt Fin CFD Data



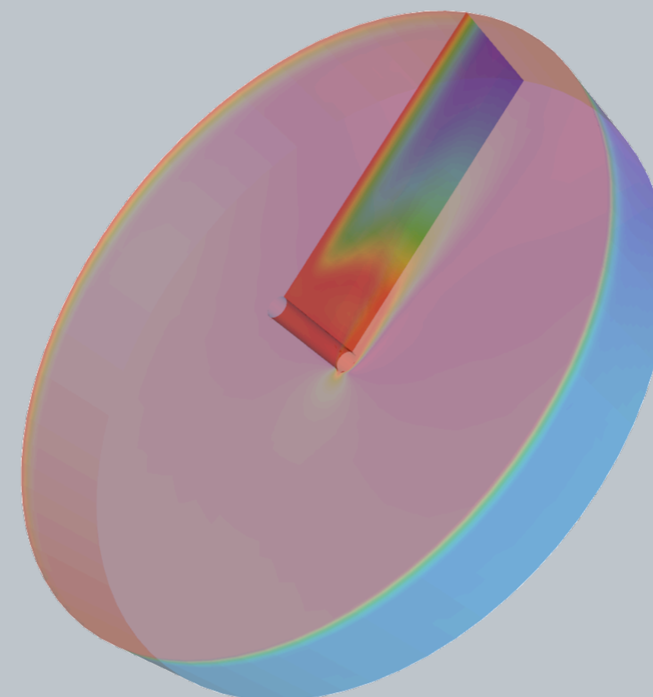
Liquid Oxygen Post



616,050 tetrahedra

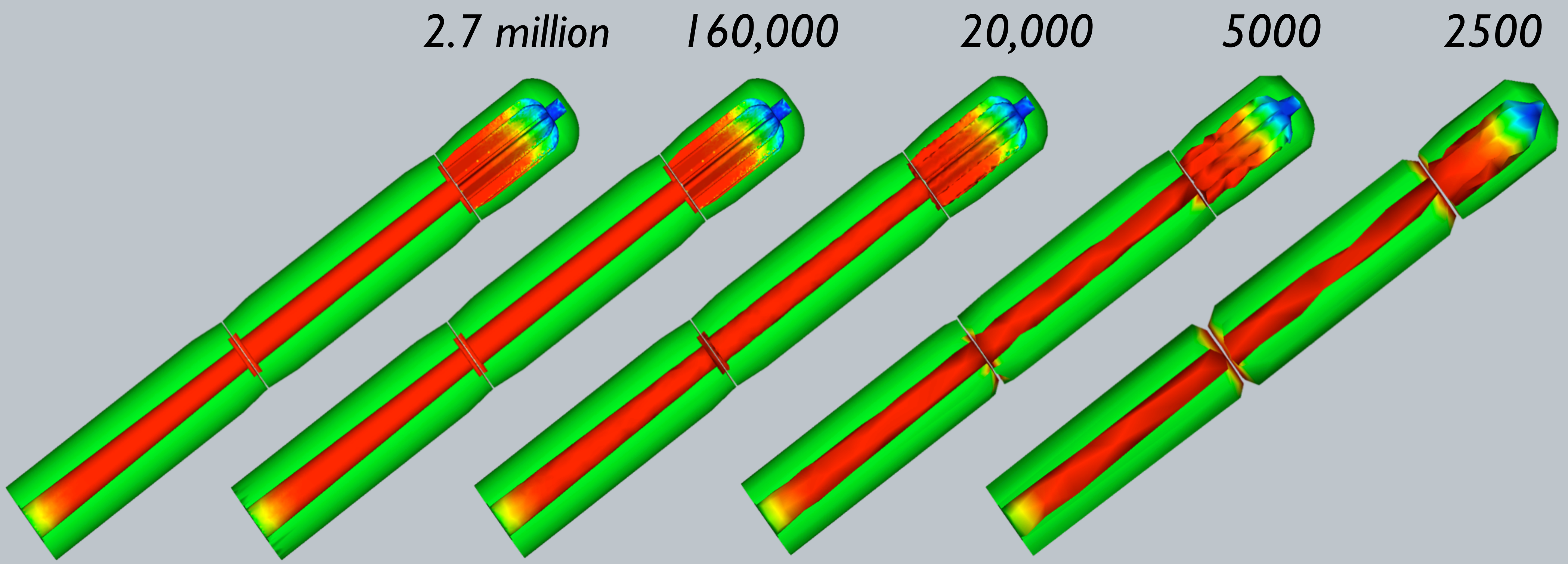


144,000 tetrahedra



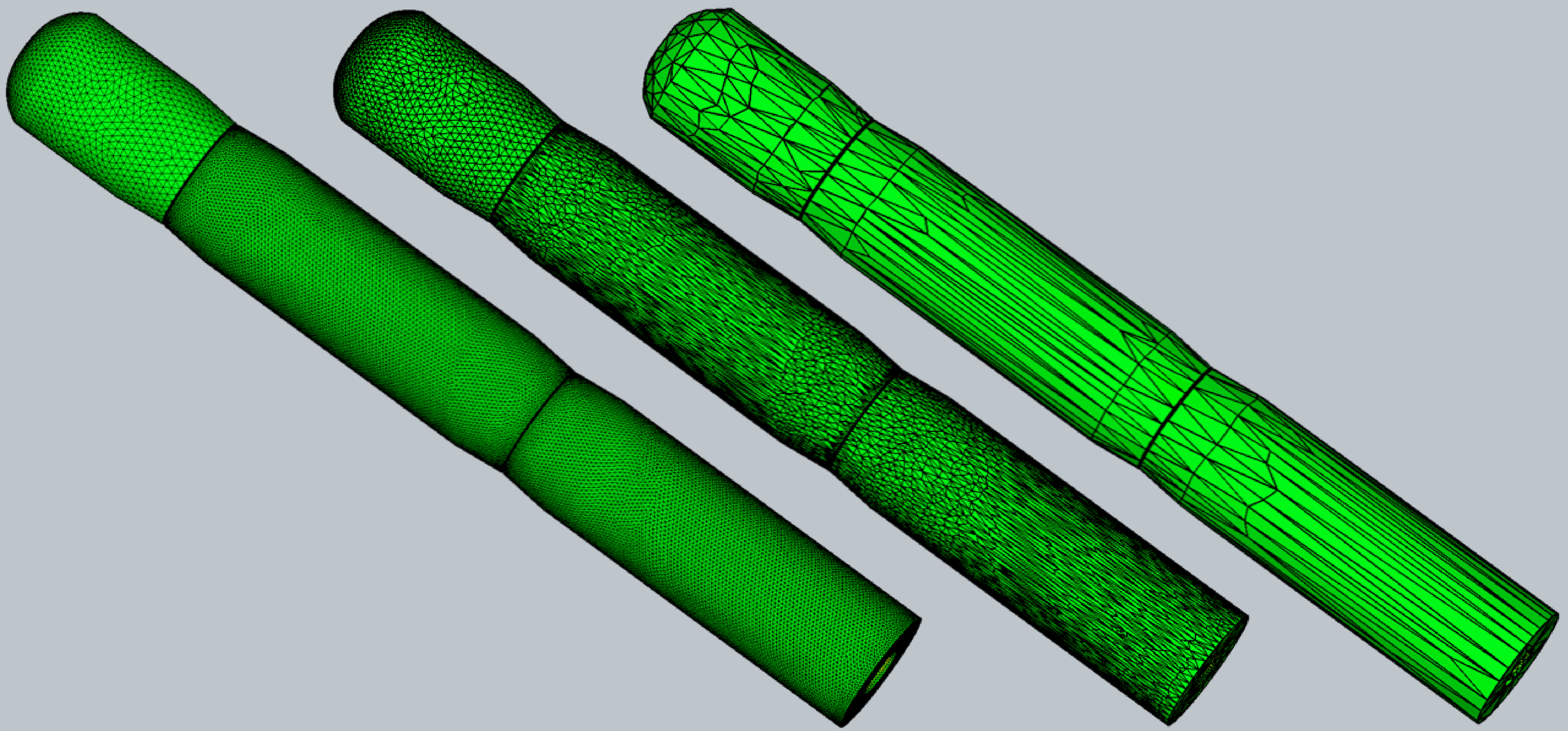
72,000 tetrahedra

Titan IV Rocket

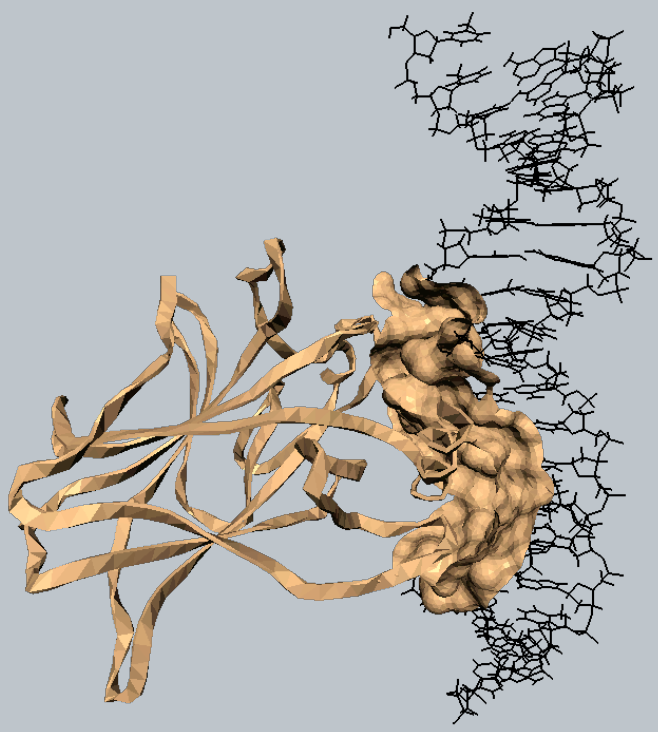


3.5 minutes

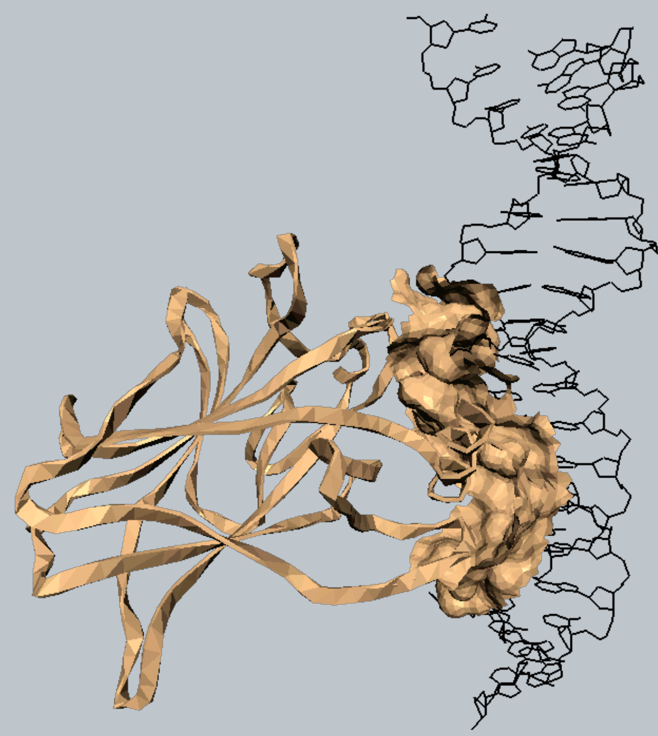
Titan IV Boundary



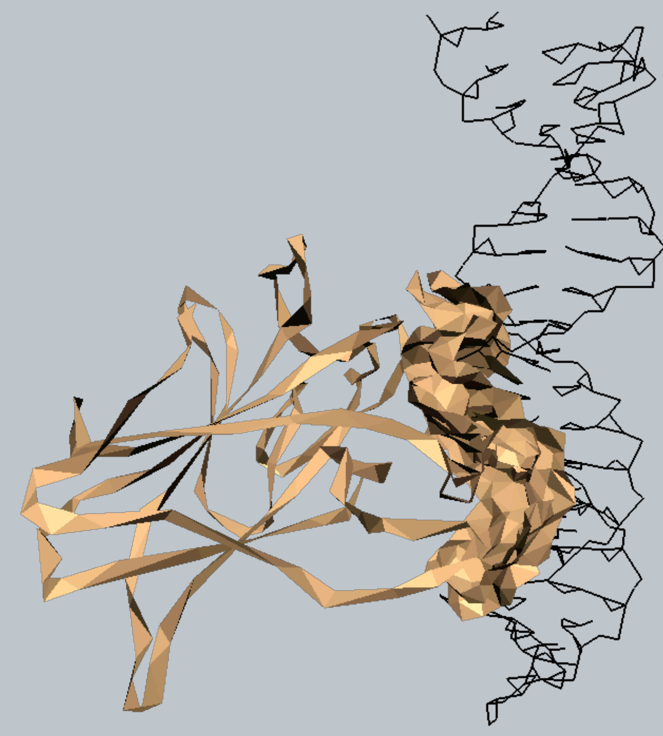
Mixed Complexes



10,710 vertices



5000 vertices



1000 vertices

Massive Meshes

Overall Approach

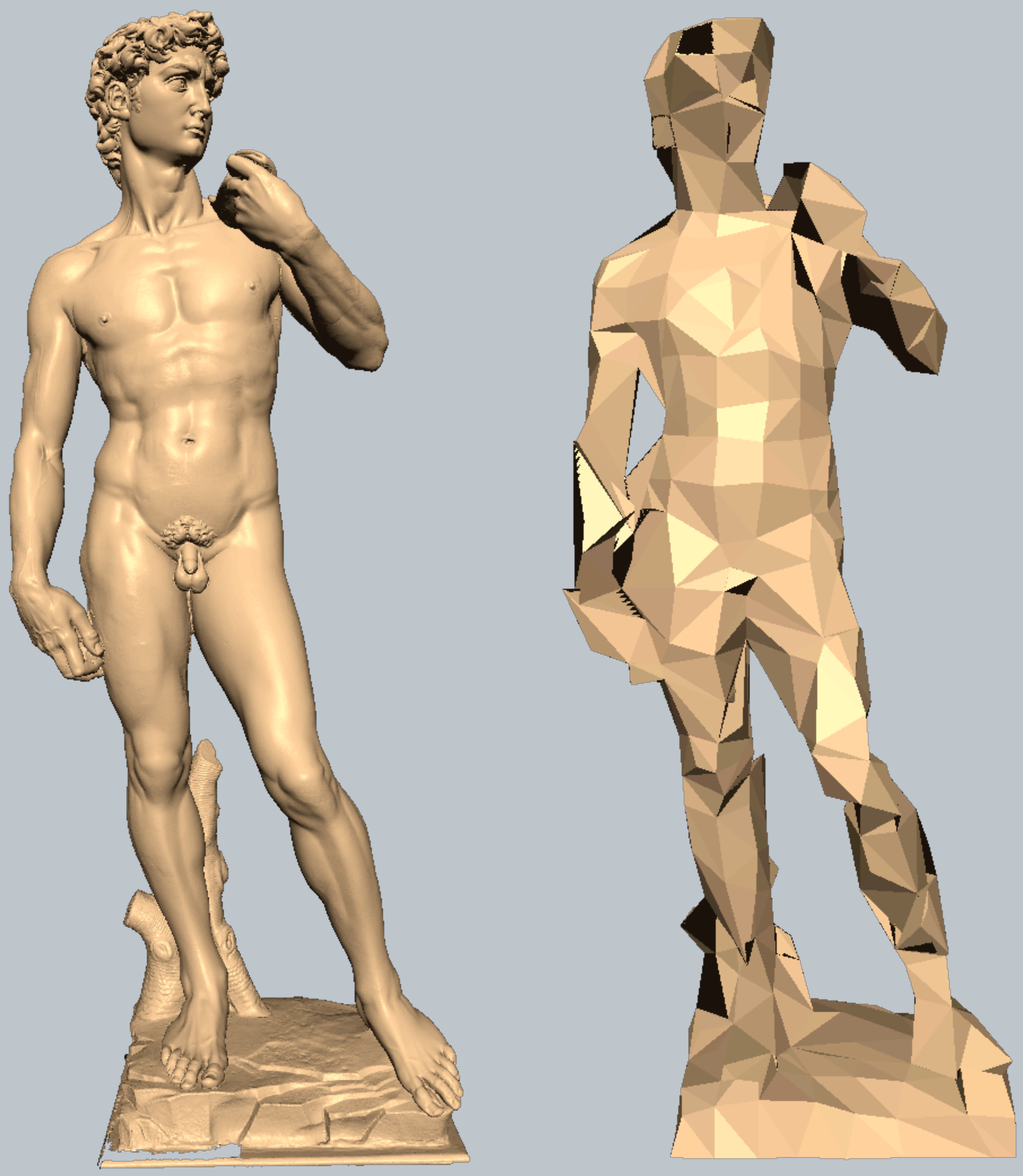
- Memory usage independent of input size
- Single linear scan of input data

Out-of-Core Clustering

- Single pass spatial clustering
  - partition space into cells
  - merge all vertices within a cell

- Usual suspects for cell decomposition
  - uniform grids, BSP trees, octrees

Grids Go Bad in the End



Our Multiphase System

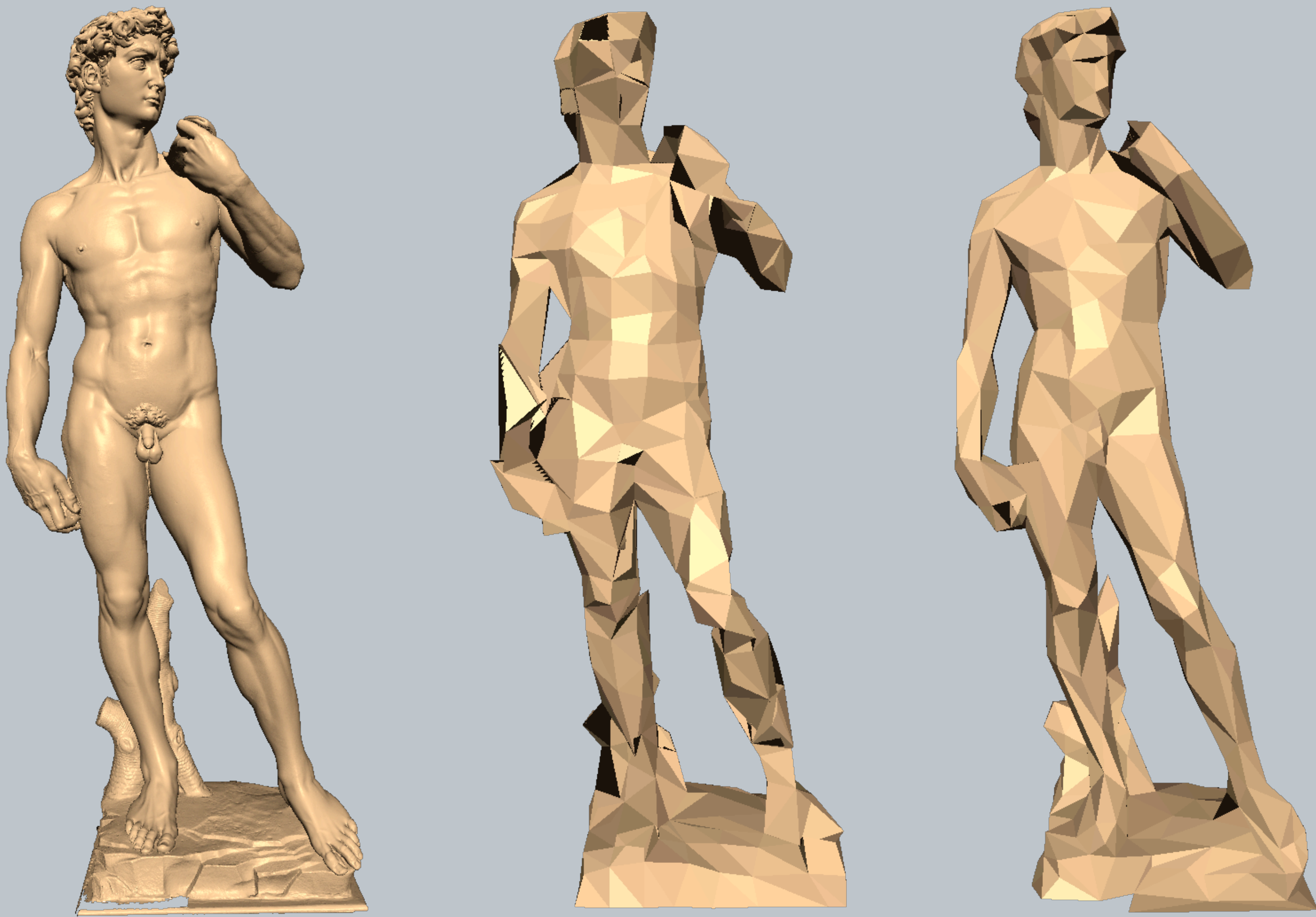
Phase I

- an initial clustering on fine grid
- accumulate quadrics for all faces in grid

Phase II

- iterative contraction
- using quadrics from Phase I

1000 Face Approximation

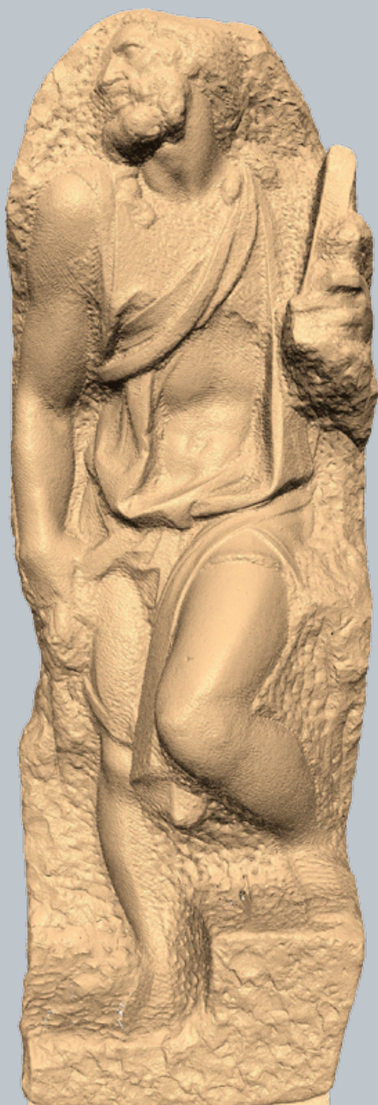


8 million faces

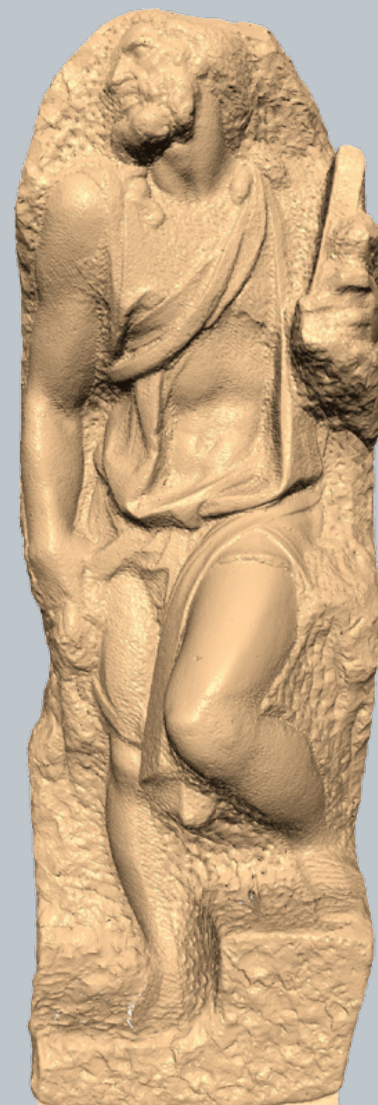
Uniform Grid
(46 seconds)

Multiphase
(65 seconds)

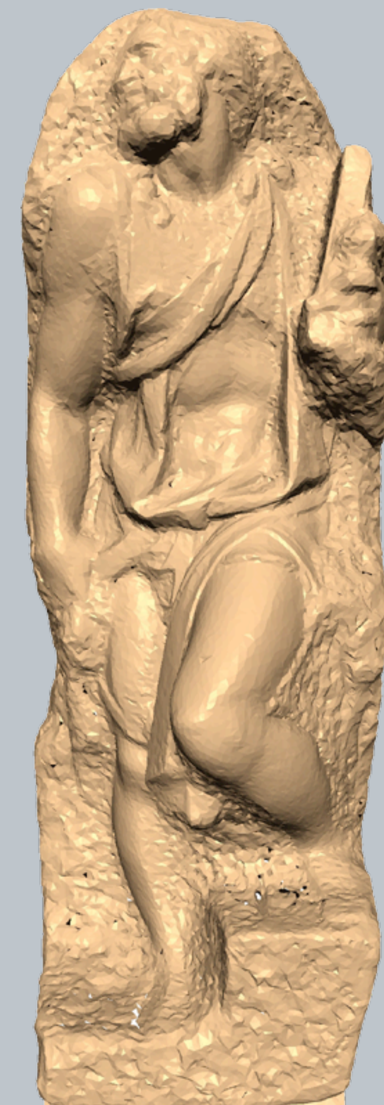
St. Matthew Sample



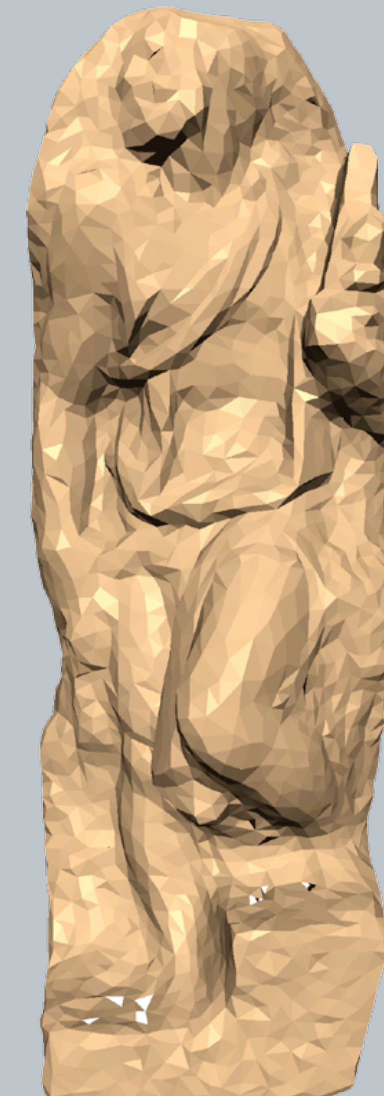
300 million



1 million



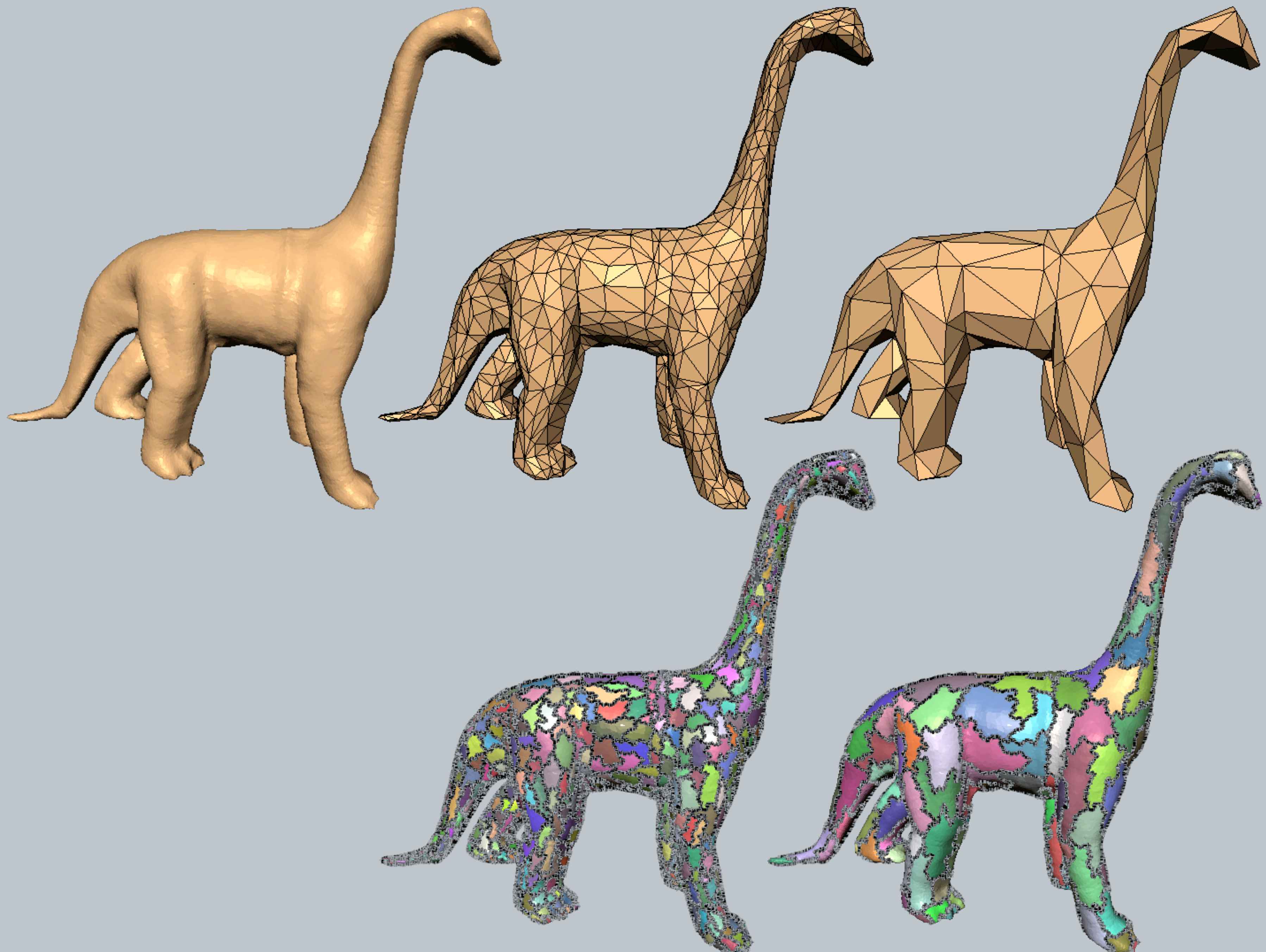
100,000



5000

What is Simplification?

Simplification is Partitioning



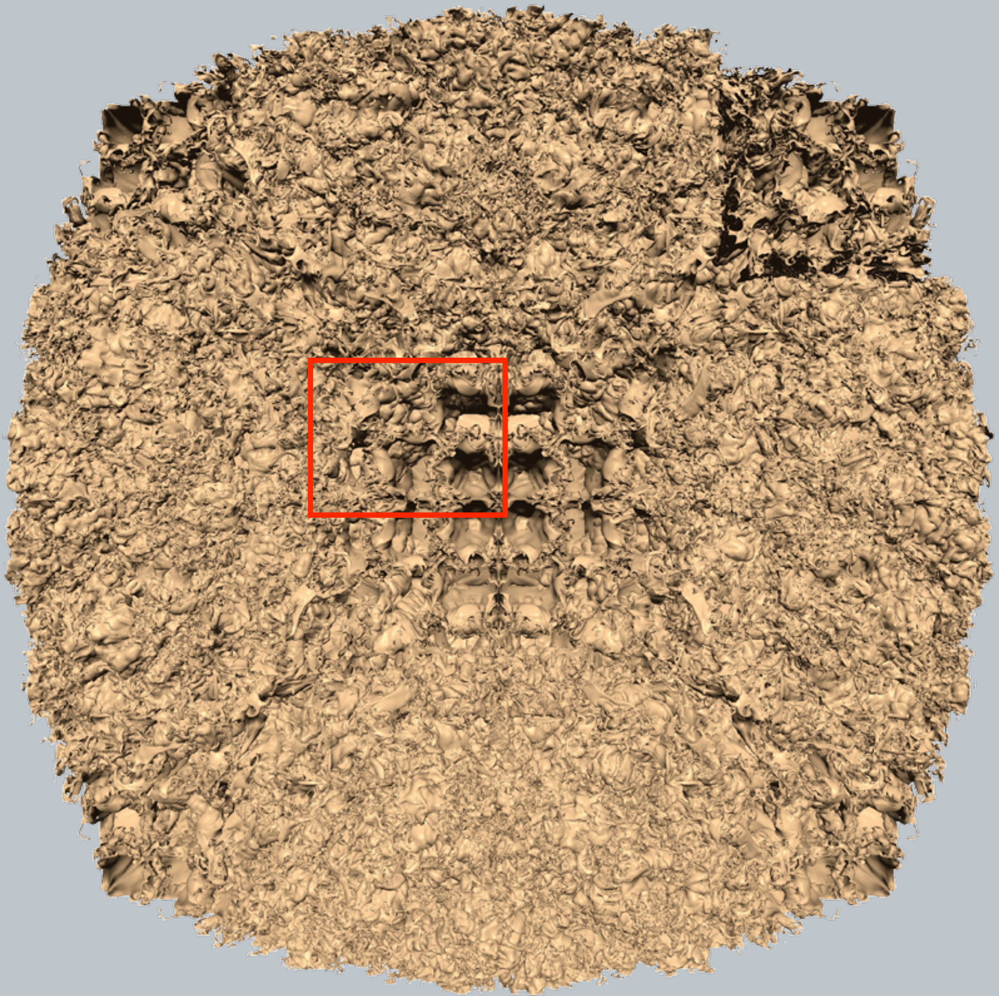
Conclusion

Summary

- Handles non-manifolds of any dimension
- Attractive blend of efficiency and quality
- Simple to implement

- But ... no (real) quality guarantees

Extreme Complexity Remains Hard



Questions?