SAND2008-1881C

# Mesh Generation and Parallel Mesh Generation for Biomedical Applications

**Jason Shepherd**
**Philippe Pebay**
**Michael Stephenson**

**March 2008**

# Outline

- **Motivation**
- **Three projects**
  - SCIRun / BioMesh3D
    - **Callahan, M., Cole, M., Shepherd, J., Stinstra, J., Johnson, C., "BioMesh3D: A Meshing Pipeline for Biomedical Computing," accepted to a special biomedical issue of *Engineering with Computers.***
  - Dissertation and ongoing research
    - **J.F. Shepherd, "Topologic and Geometric Constraint-Based Hexahedral Mesh Generation," Doctoral Dissertation, University of Utah, 2007.**
  - SNL's pCAMAL
    - **Pebay, P., Stephenson, M.B., Fortier, L., Owen, S., Melander, D., "pCAMAL: An Embarrassingly Parallel Hexahedral Mesh Generator," Proceedings, 16th International Meshing Roundtable, Oct. 2007.**
- **Conclusion**

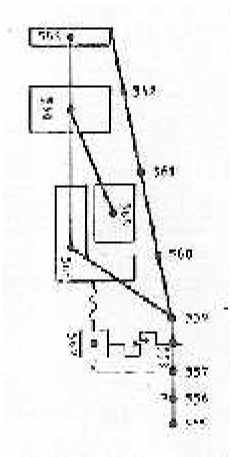
# Motivation

"Ironically, as numerical analysis is applied to larger and more complex problems, non-numerical issues play a larger role. Mesh generation is an excellent example of this phenomenon. Solving current problems in structural mechanics or fluid dynamics with finite difference of finite element methods *depends on the construction of high-quality meshes of surfaces and volumes*. *Geometric design and construction of these meshes are typically much more time-consuming than the simulations that are performed with them*."

- John Guckenheimer, "Numerical Computation in the Information Age" in June 1998 issue of SIAM News.

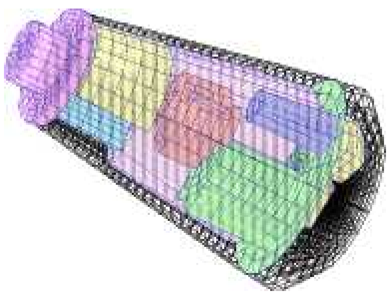
# Capacity and Resolution

**ca. 1988**



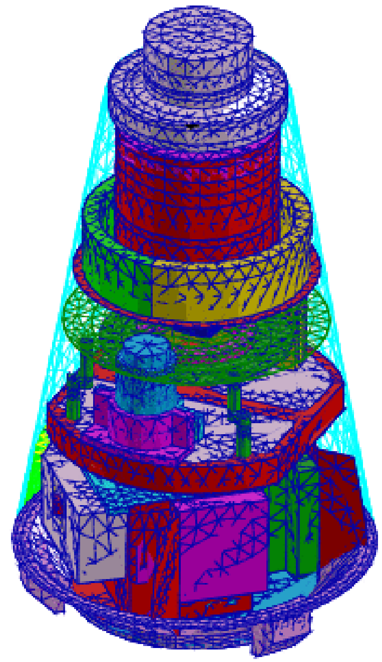
**200 dof**
**Shellshock 2D**

**ca. 1995**



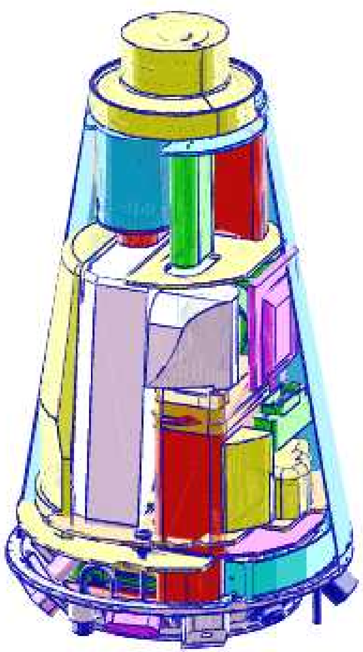
**40,000 dof**
**NASTRAN**

**ca. 1998**



**800K dof**
**MP Salinas**

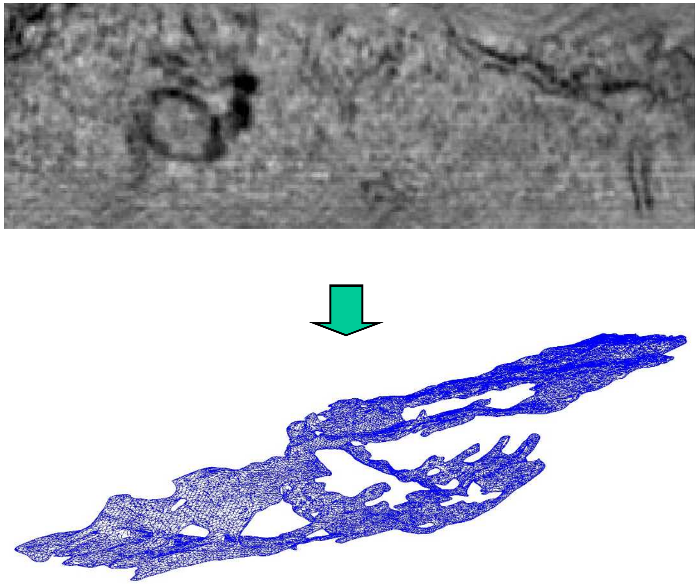
**ca. 2000**



**>10M dof**
**MP Salinas**

# Capacity and Resolution

**ca. 2002**

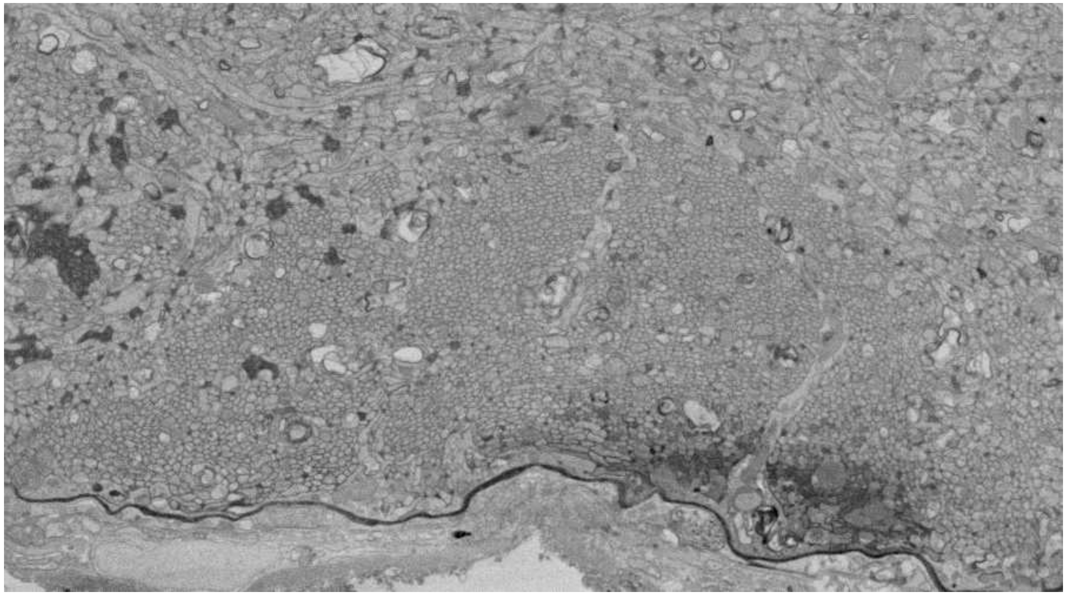


**Endoplasmic Reticulum**

**(courtesy of Bridget Wilson, et al.**

**University of New Mexico)**

**2007-2008?**



**Neural Fiber Bundles (Zebra Fish)**

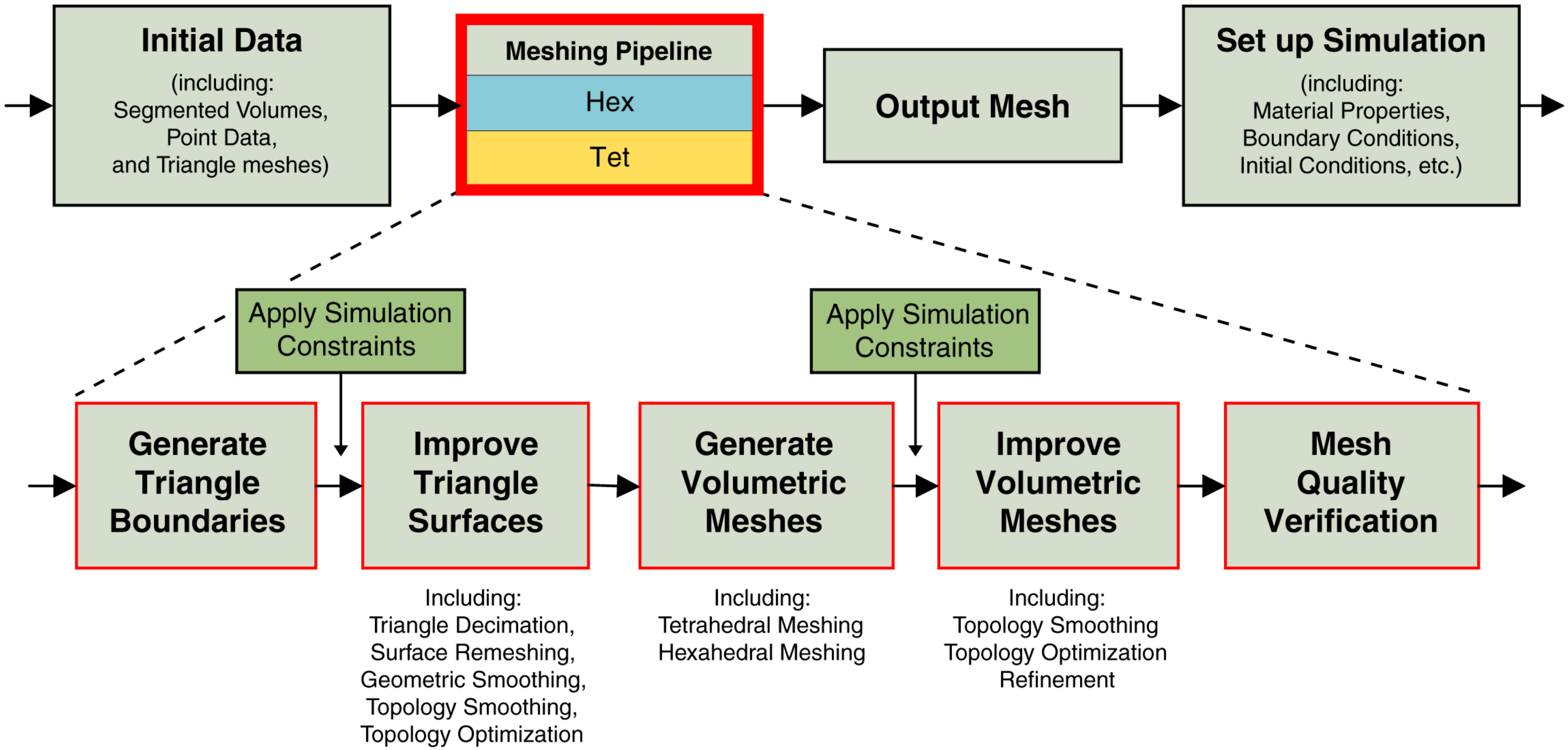
**(courtesy of Liz Jurrus & Chi-Bin Chien, University of Utah and**

**Winfried Denk, Max Planck Institute for Medical Research)**

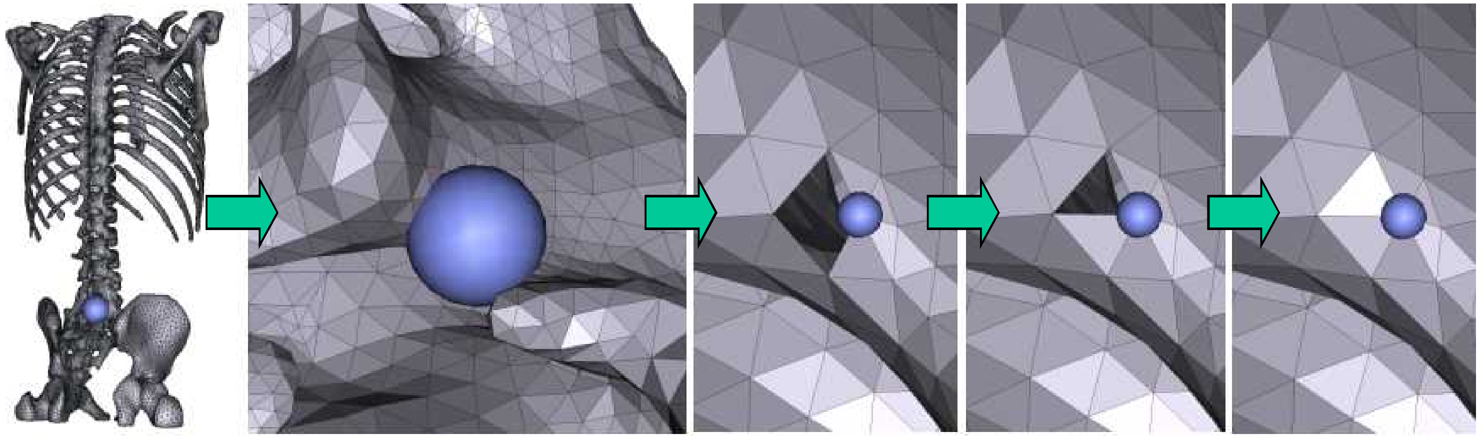
# BioMesh3D

- **Goals:**
  - Develop a suite of tools (pipeline) for efficiently generating meshes for biomedical simulation
  - Meshes must have reasonable quality for simulation
  - Tools should be available for general public release (open-source)
  - Pipeline should be expandable to new tools/techniques.
  - Easy-to-use, flexibility

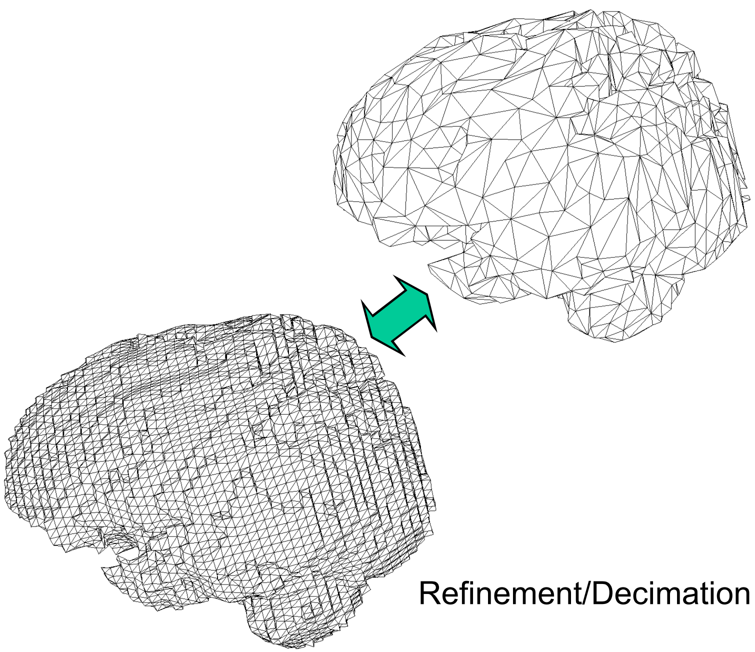
# BioMesh3D pipeline



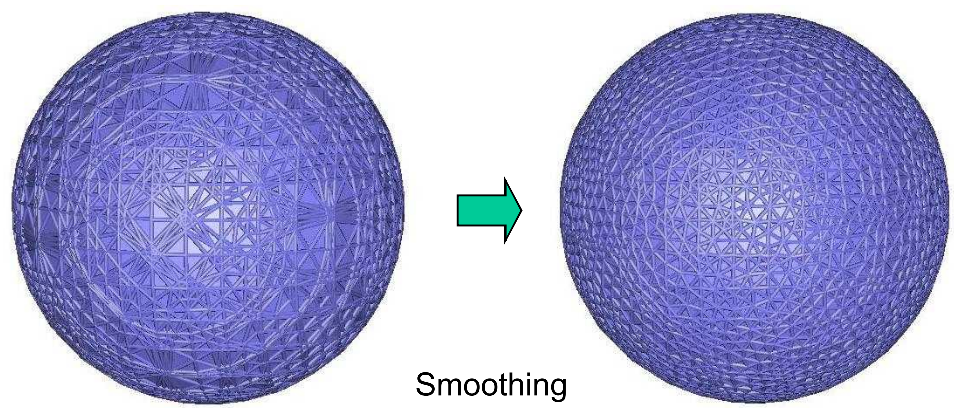
# Pipeline tools



Mesh Editing



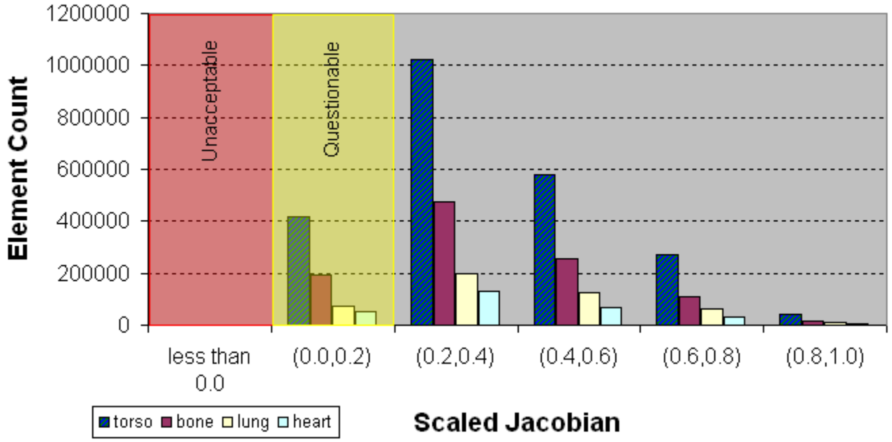
Refinement/Decimation

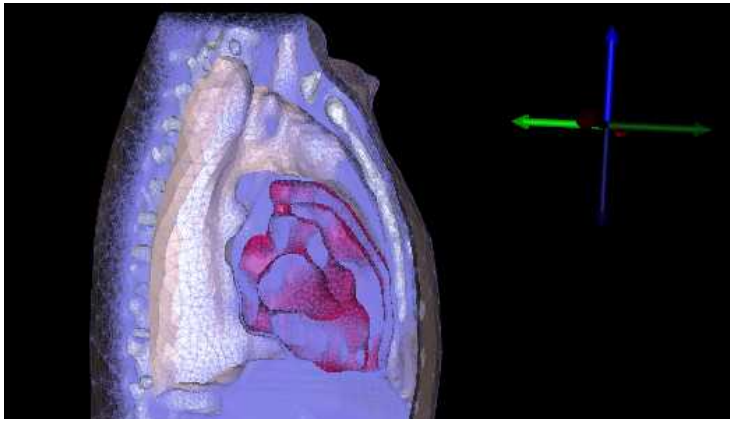


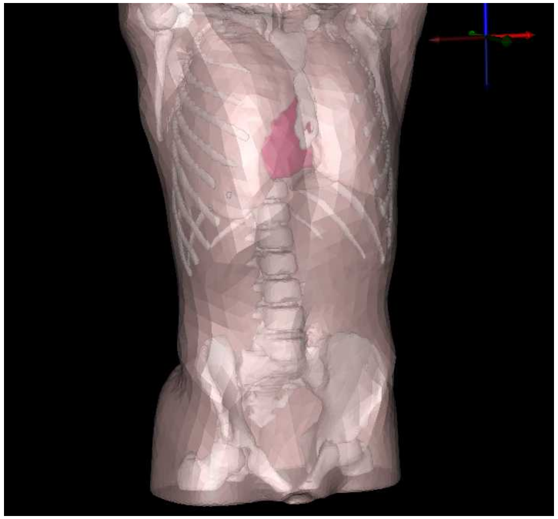
Smoothing

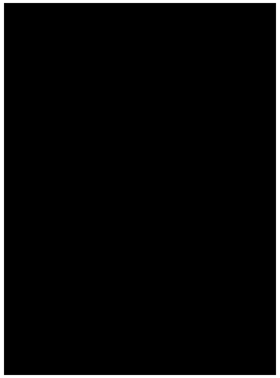
# Example – Pediatric Torso

**Element Quality Distribution for Pediatric Torso Model using TetGen**

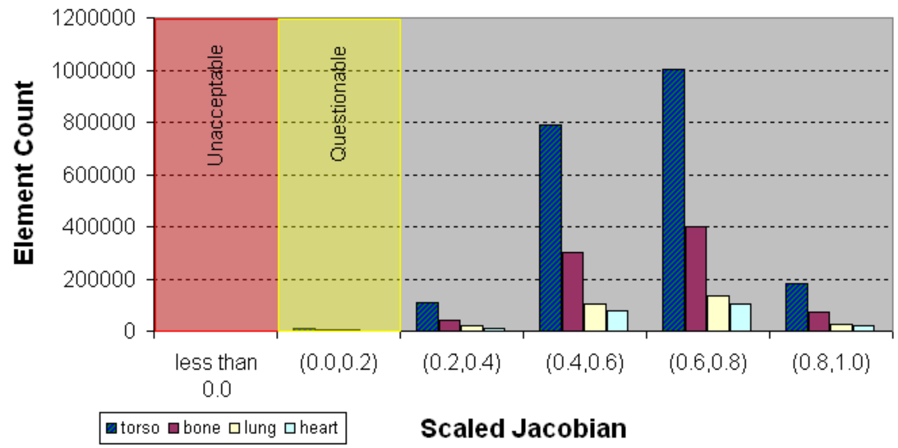


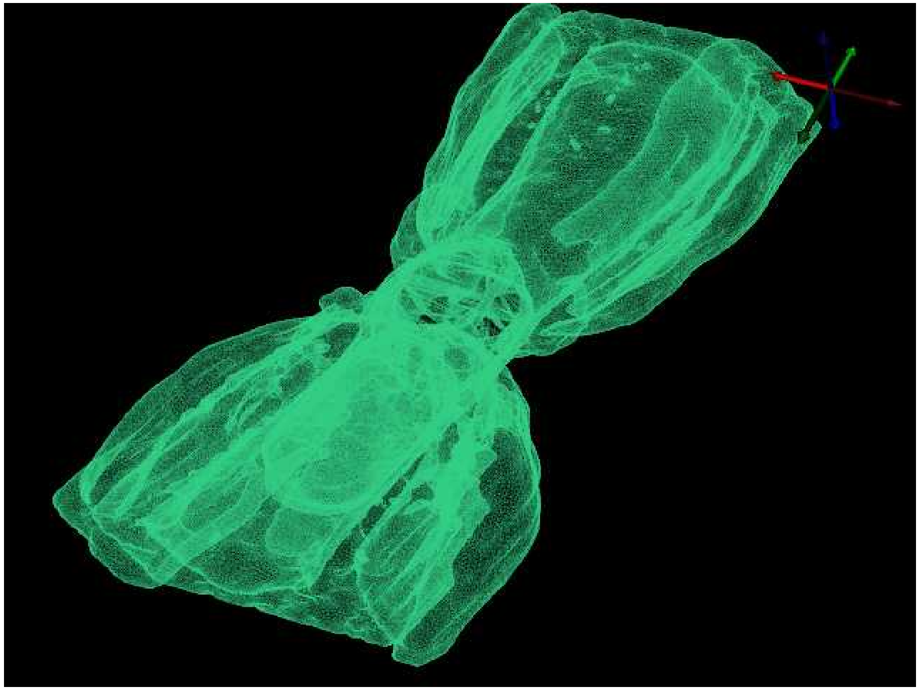




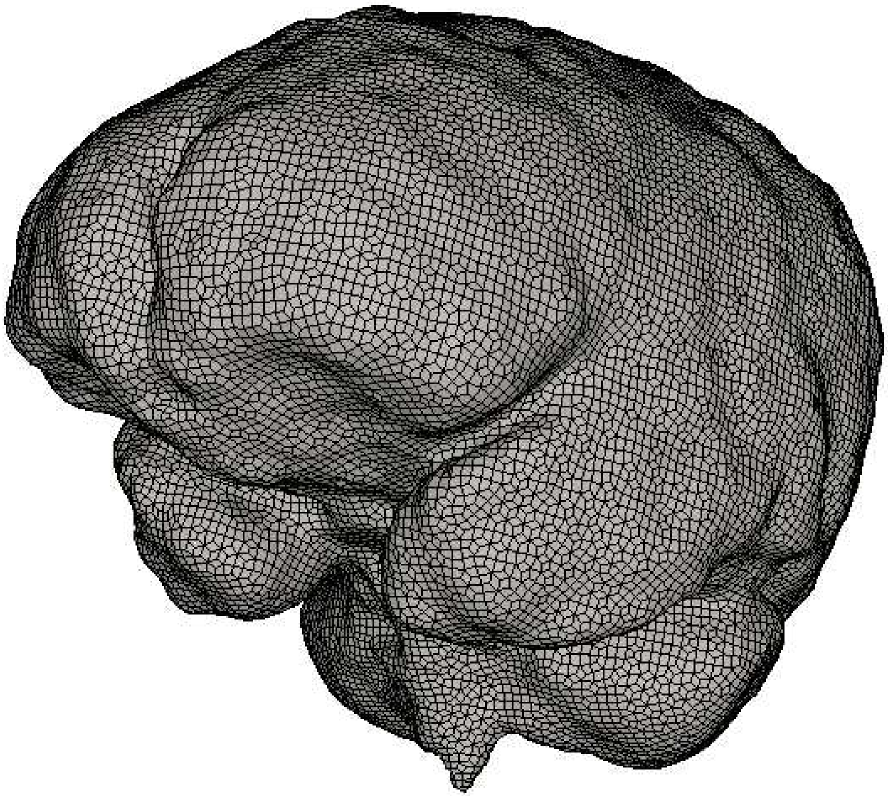
**Element Quality Distribution for Pediatric Torso Model Using GHS3D (CAMAL)**



# Pipeline Examples



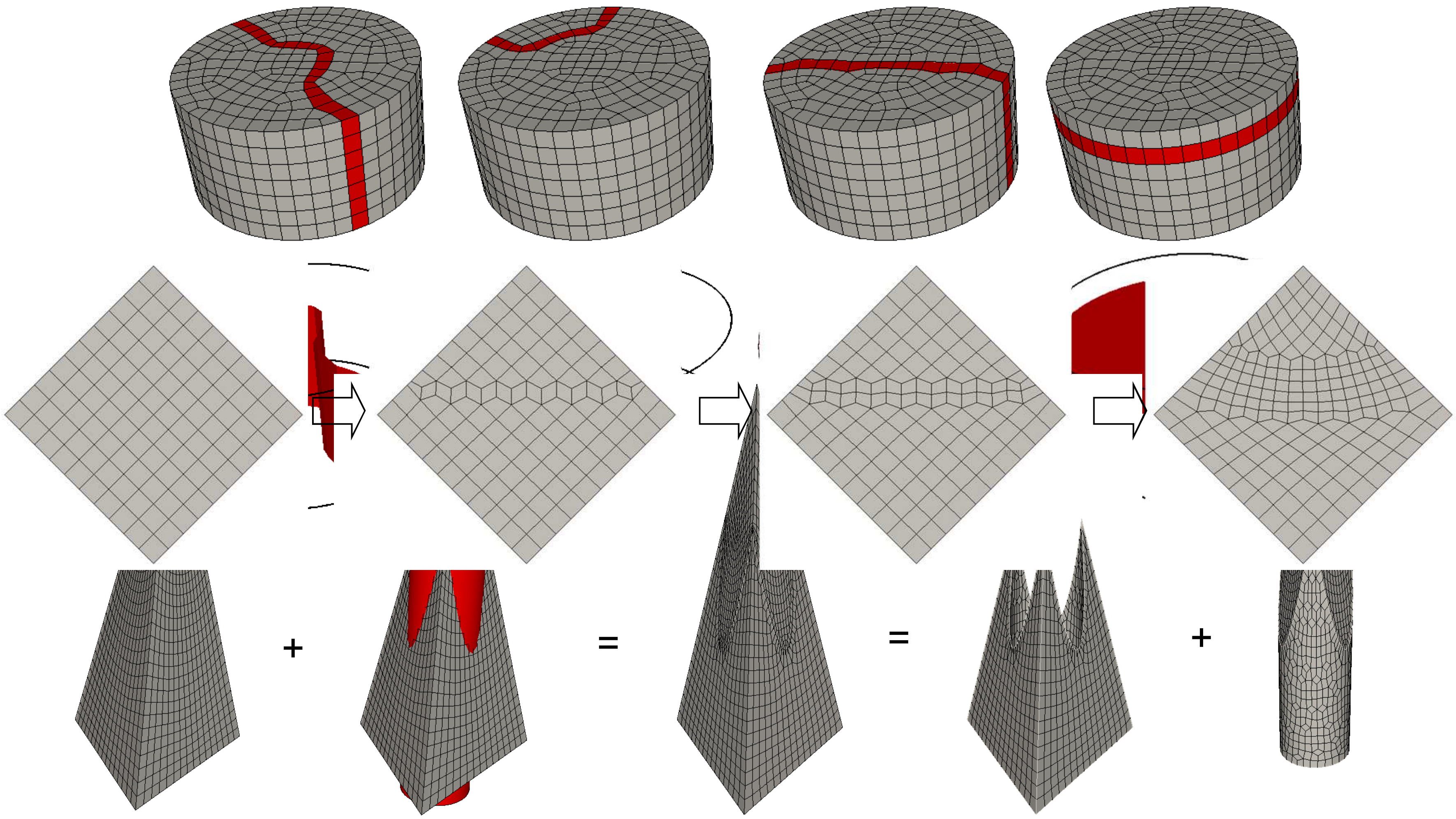
Mesh generated by Marty Cole, UofU
Model courtesy of Ellisman, et al., UCSD (NMCIR and Cell-Centered DataBase (CCDB))



Mesh generated by J. Shepherd
Model courtesy of Simon Warfield

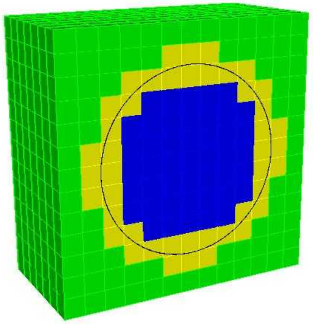
# Hexahedral mesh generation for biomedical models

# Methods – Sheet Insertion and Extraction



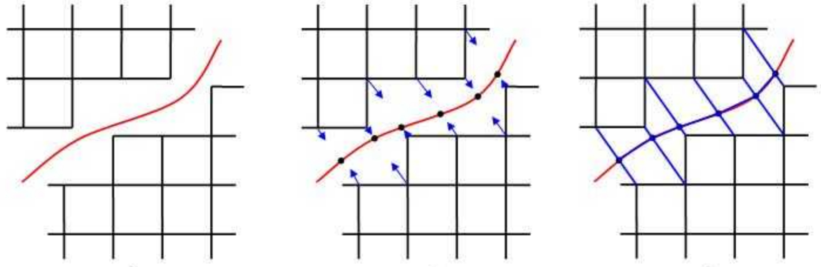
# Methods –
# Sheet Insertion (Pillowing)

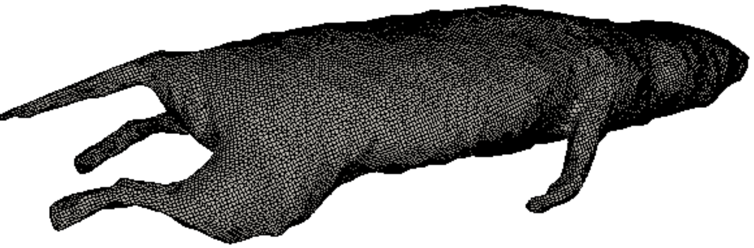
**Given a hexahedral mesh (not necessarily octree) and a triangle mesh on a manifold**

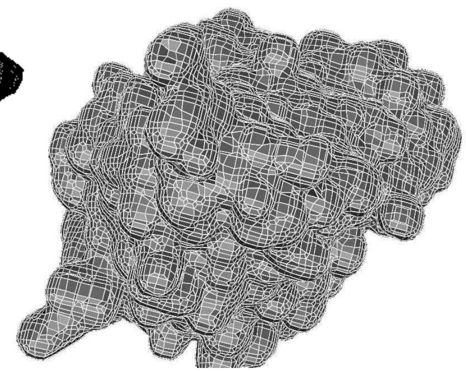


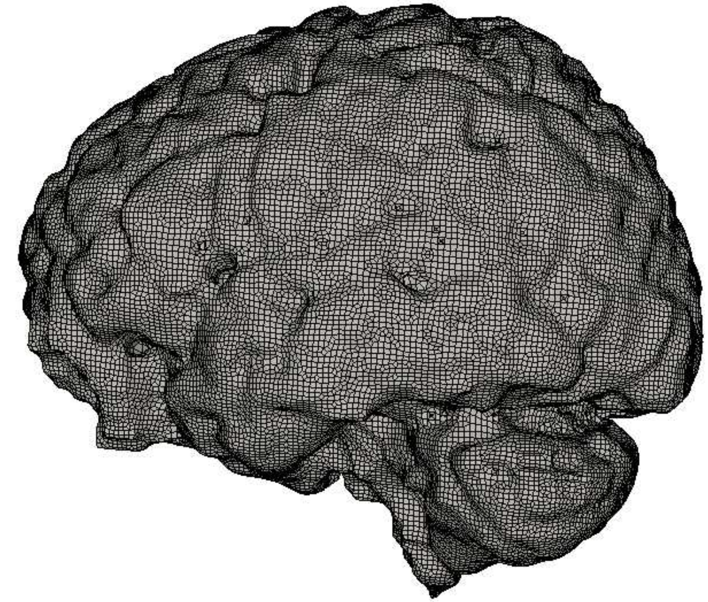
1. **Separate the hexahedra into three groups**
    1. Hexes intersected by the triangle mesh
    2. Hexes to one side of the triangle mesh (Side1), and
    3. Hexes on the opposite side of the triangle mesh (Side2).

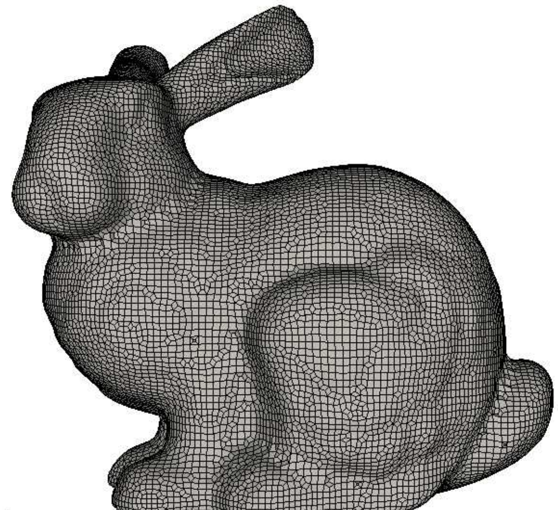
2. **Placing the intersected hexes with one of the two sides, insert two sheets of hexahedra between the resulting groups projecting the new nodes to the original triangle mesh.**

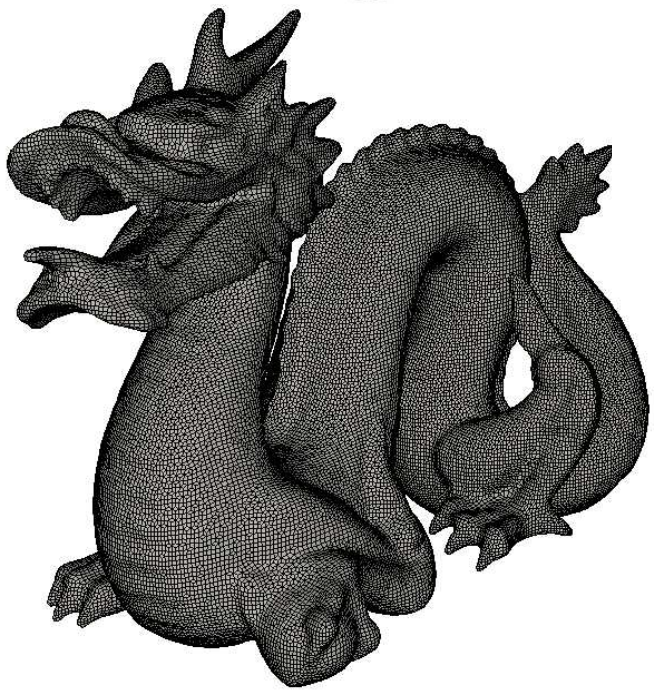


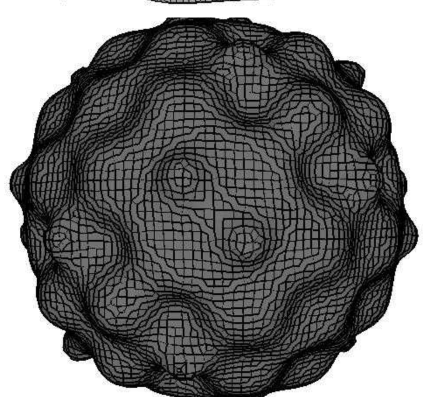


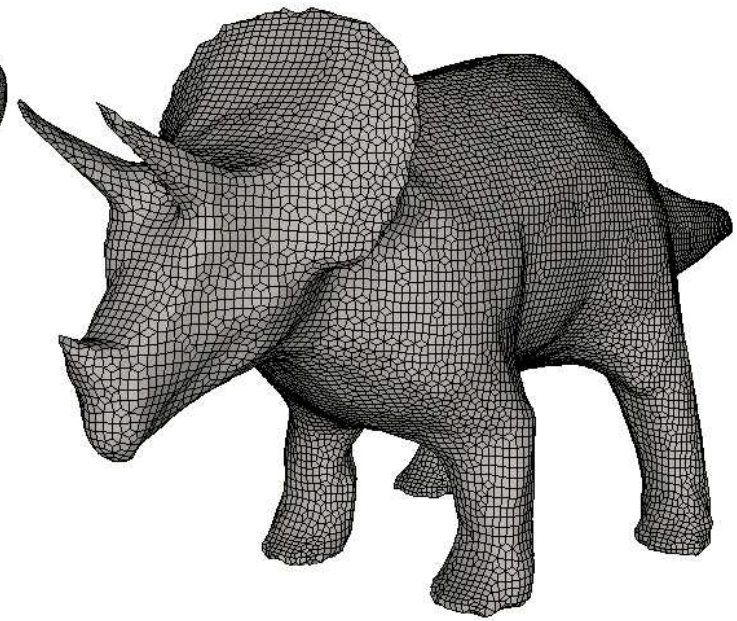






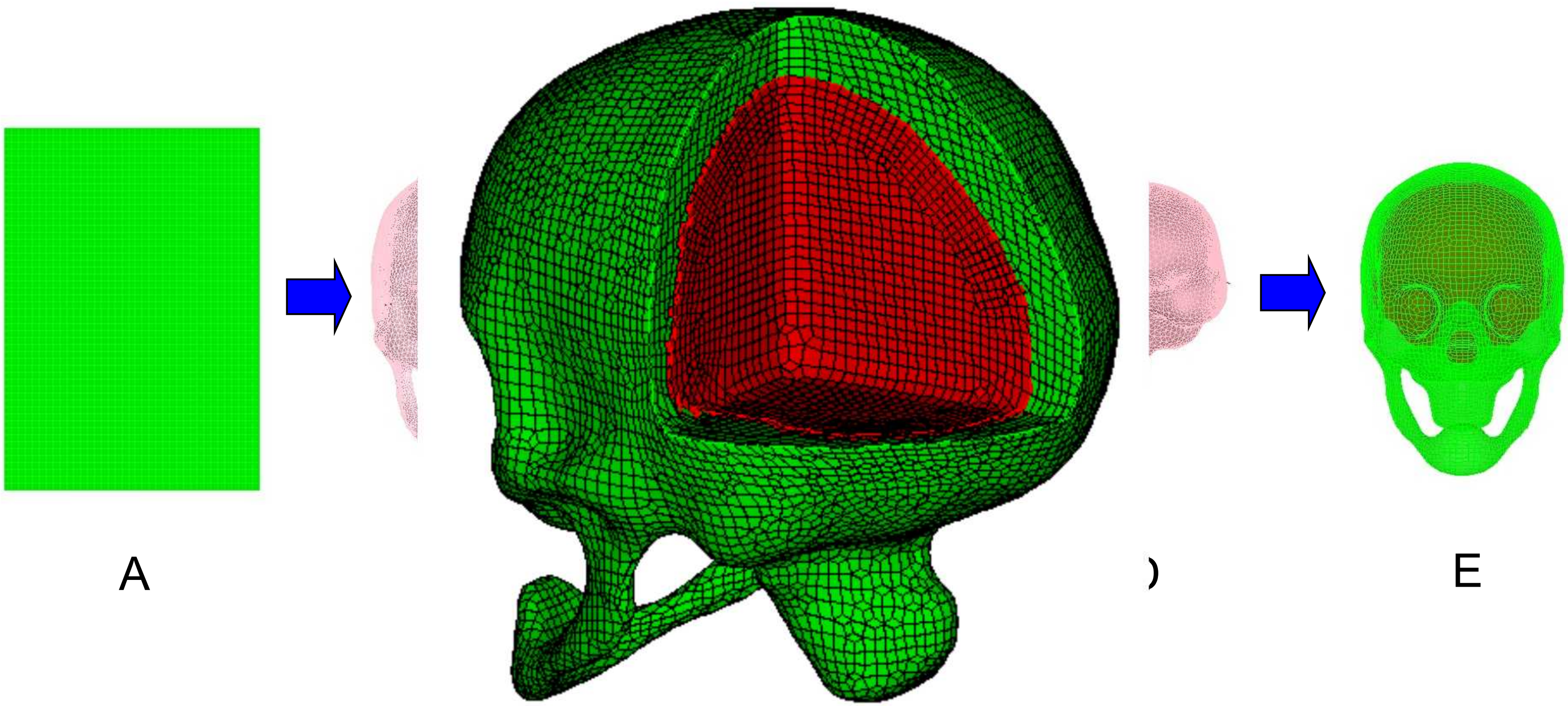






-Mouse model is courtesy of Jeroen Stinstra of the SCI Institute at the University of Utah
-Bumpy Sphere model is provided courtesy of mpii by the AIM@SHAPE Shape Repository
-Brain and Hand Models are provided courtesy of INRIA by the AIM@SHAPE Shape Repository

# Multi-surface
# Hexahedral Mesh Generation

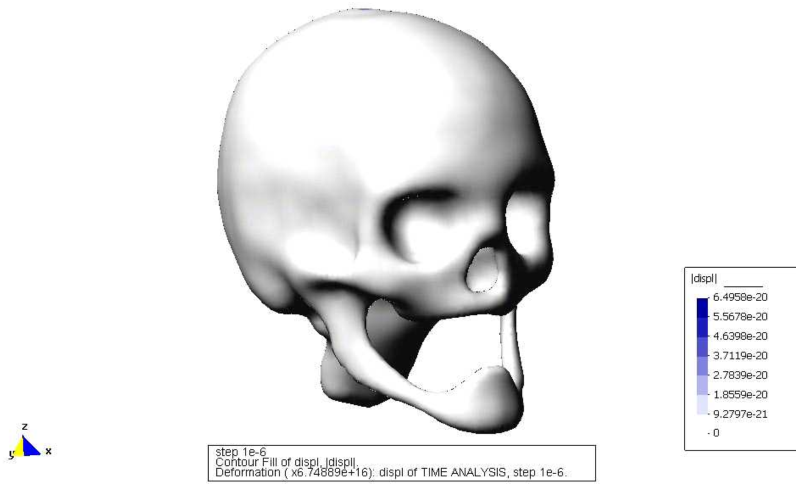



-Original model is provided courtesy of Inria by the AIM@SHAPE Shape Repository

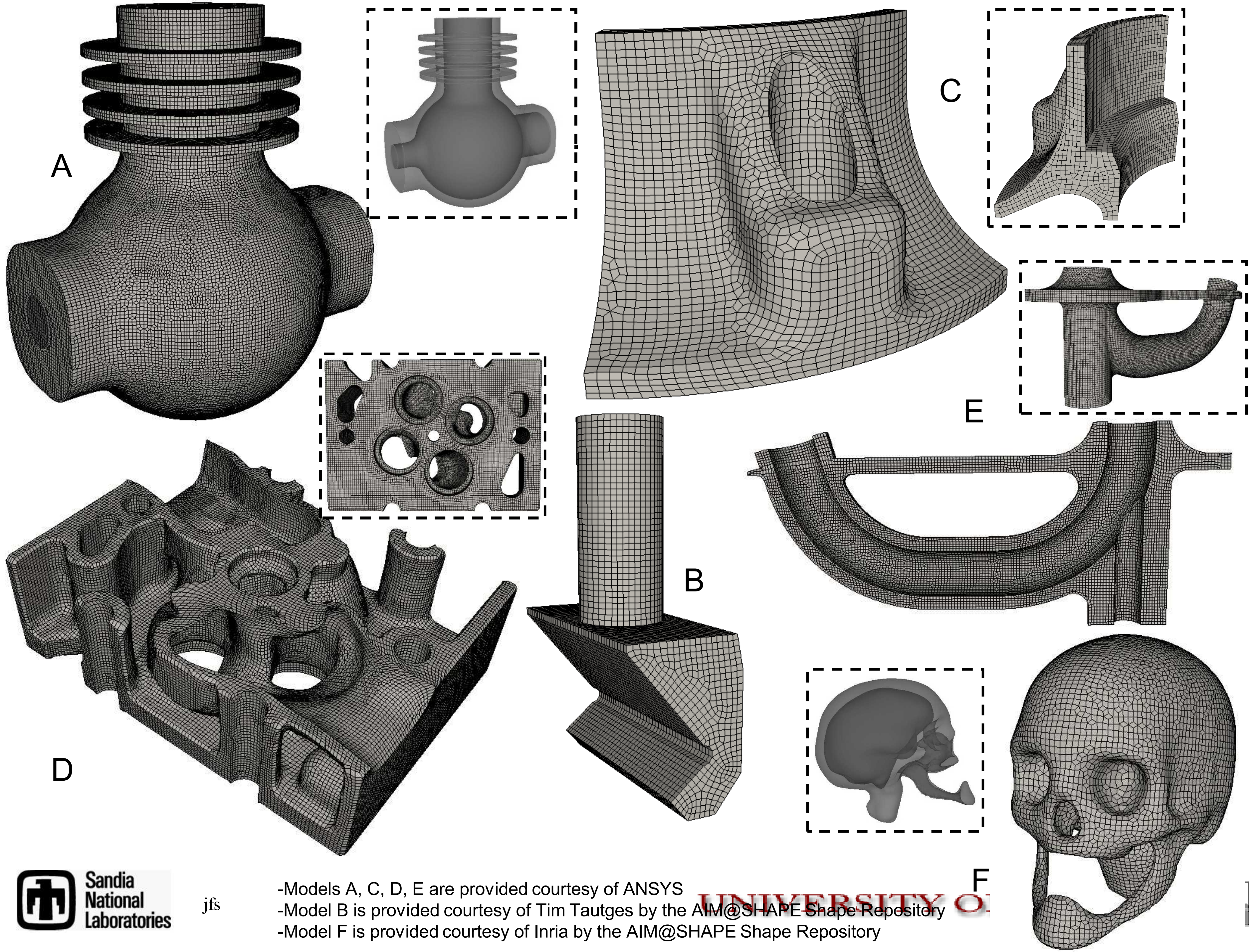
# Multi-surface Hexahedral Mesh Generation

- **Example (skull) -**


Skull bone shown in blue
Cranial cavity shown in magenta




Impact analysis courtesy of Dr. Marco Stupazzini, Department fuer Geo- und Umweltwissenschaften Sektion Geophysik Ludwig-Maximilians-Universitaet Theresienstrasse 41

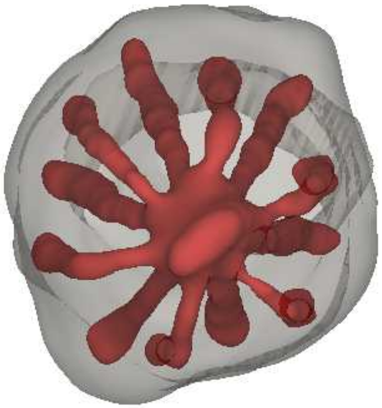


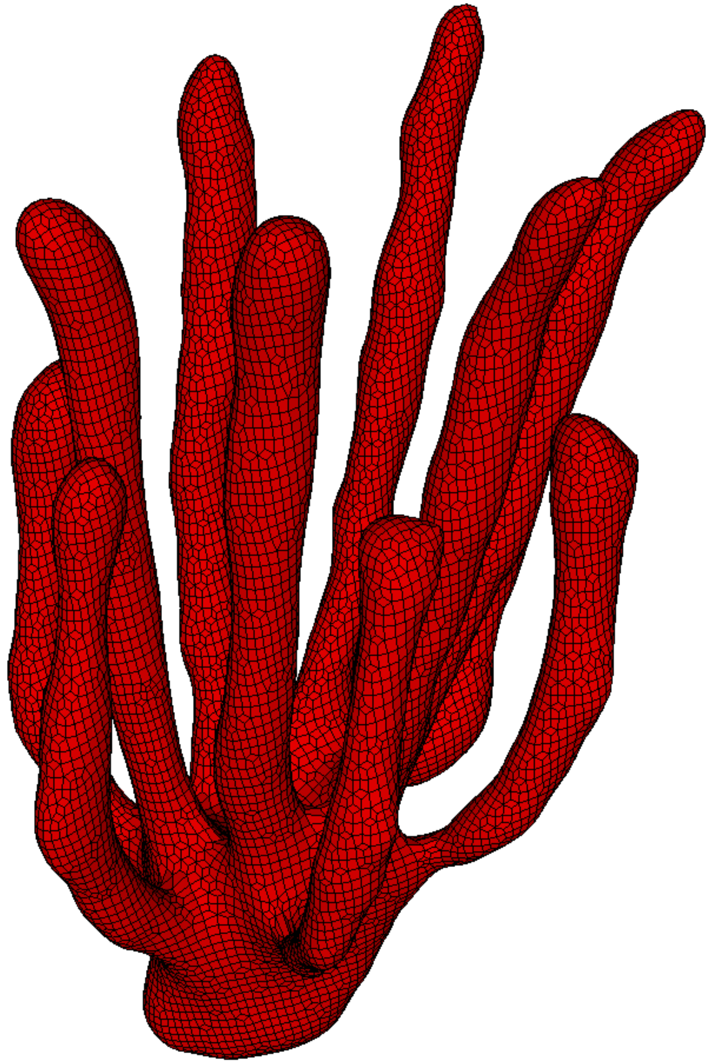

Sandia National Laboratories

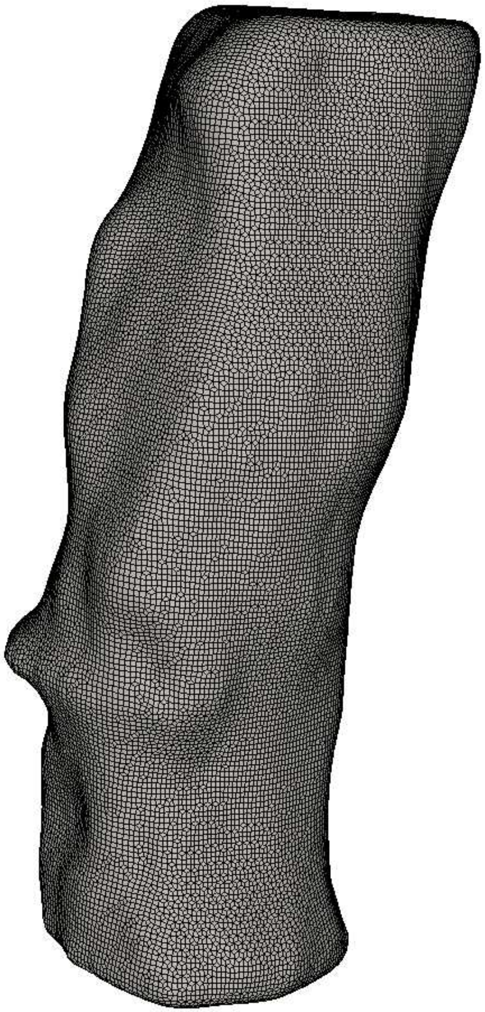
jfs

-Models A, C, D, E are provided courtesy of ANSYS
-Model B is provided courtesy of Tim Tautges by the AIM@SHAPE Shape Repository
-Model F is provided courtesy of Inria by the AIM@SHAPE Shape Repository

# Multi-surface Hexahedral Mesh Generation







Created for S. Shontz's IVC Collaboration with F. Lynch, M.D. (PSU Hershey Medical Center), M. Singer (LLNL), S. Sastry (PSU), and N. Voshell (PSU)

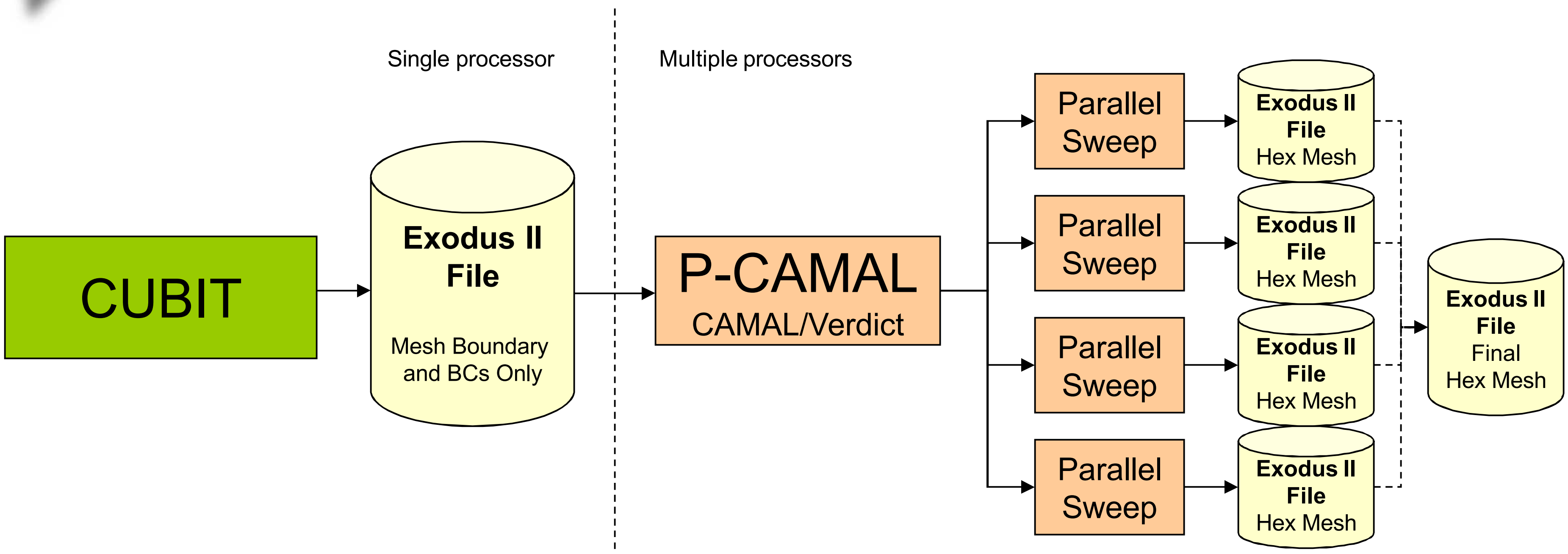
# Parallel Mesh Generation

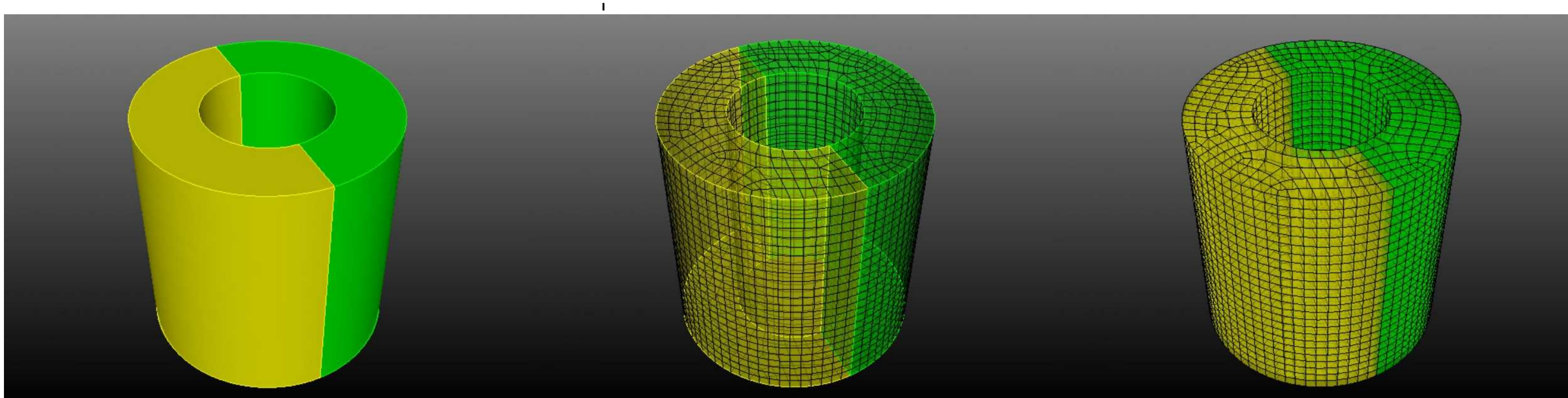
# The Problem

- **National laboratories need very large meshes**
  - Currently 10+ million unstructured hex elements
  - Future 100 million to 1 billion unstructured elements

- **Cannot generate large mesh on user workstation**
  - Limited by memory and processor power
  - Limited by size of file exported to analysis software

# Two-Stage Solution



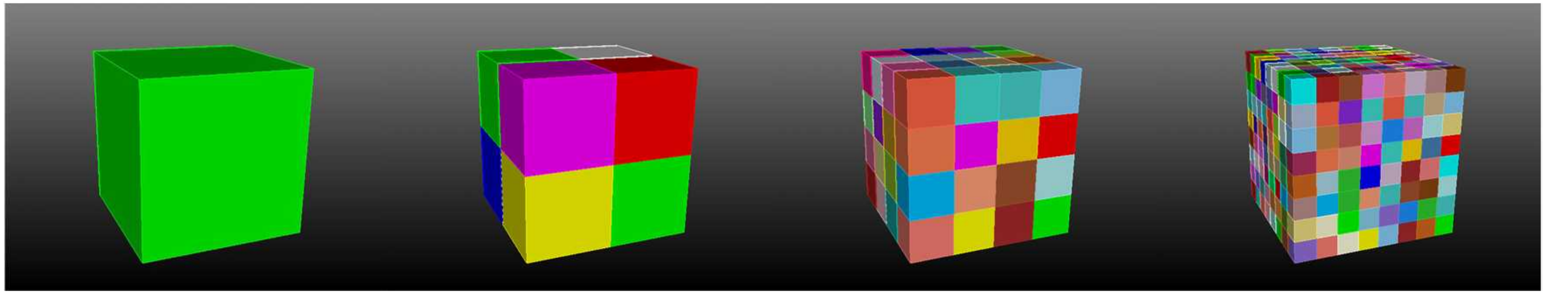



# Test Environment

- **SNL Catalyst Cluster**
  - 128 dual core Xeon processors (3.06 GHz)
  - 2 GB or memory per processor
  - Gigabit ethernet
  - 4X Infiniband high-speed network
  - Linux 2.6.17.11 kernel
  - TORQUE resource manager (batch scheduling)

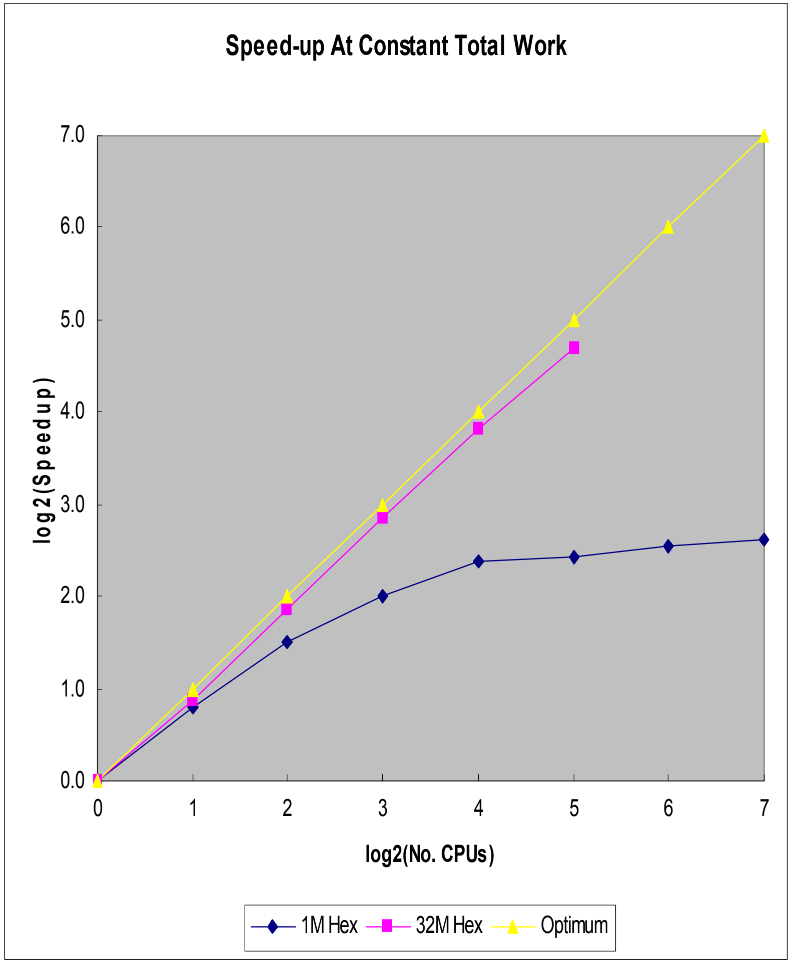
# Similar Cubes

- **Subdivide cube along axes**
  – Uniform subdivisions shown
  – Other subdivision schemes give rectangular solids

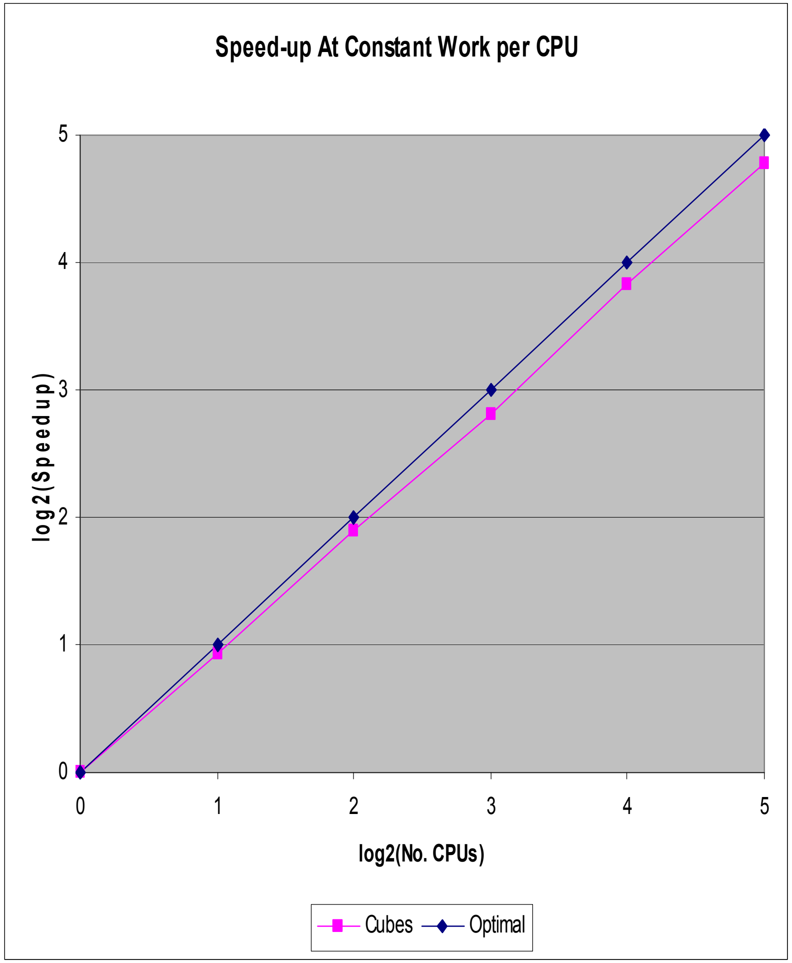


# Similar Cubes

- **1M hexes**
  - 1024 cubes of 1000 hexes
  - Overhead dominates
- **32M hexes**
  - 32 cubes of 100,000 hexes

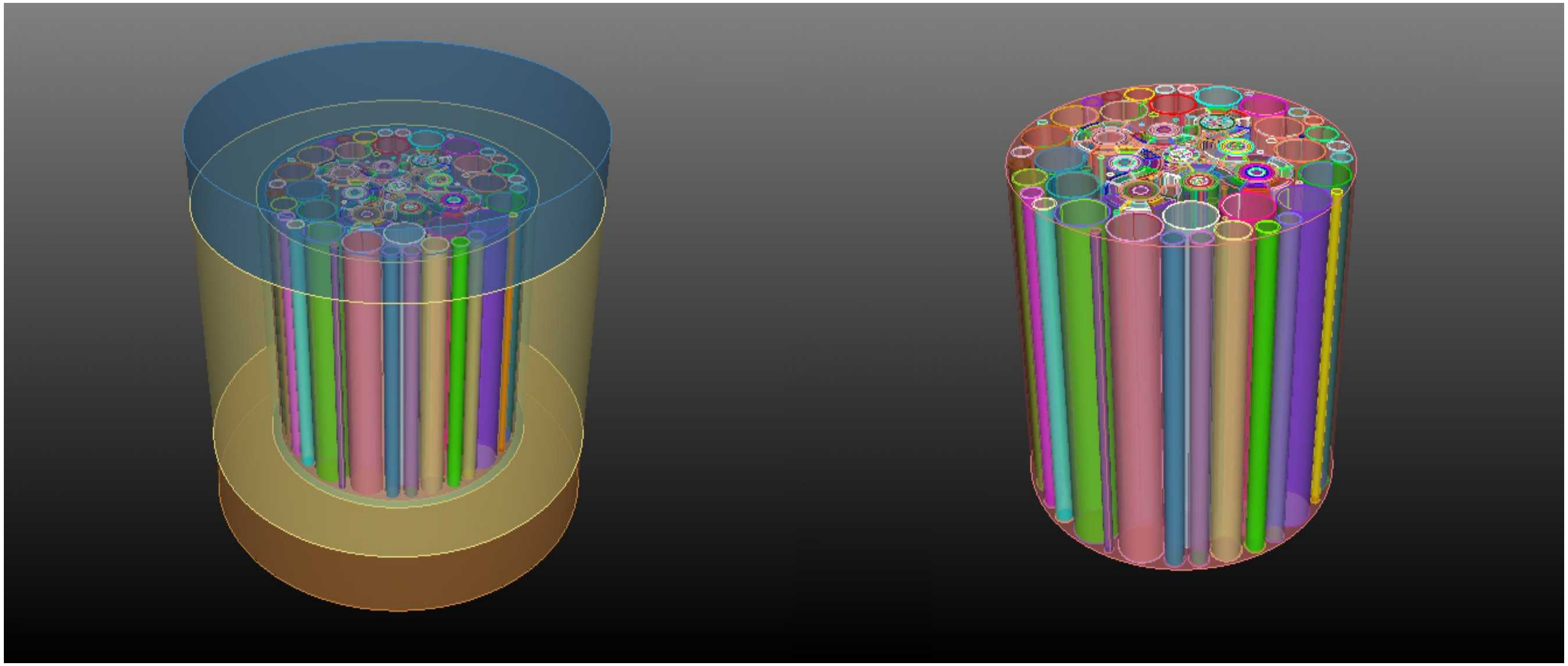


# Similar Cubes

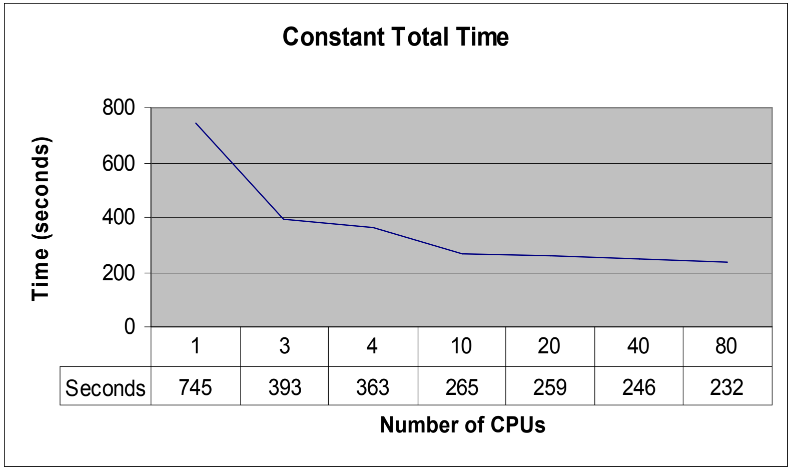



- **1,000,000 hexes/processor**
  - 1M to 32 M hexes

# INL Reactor Core Model

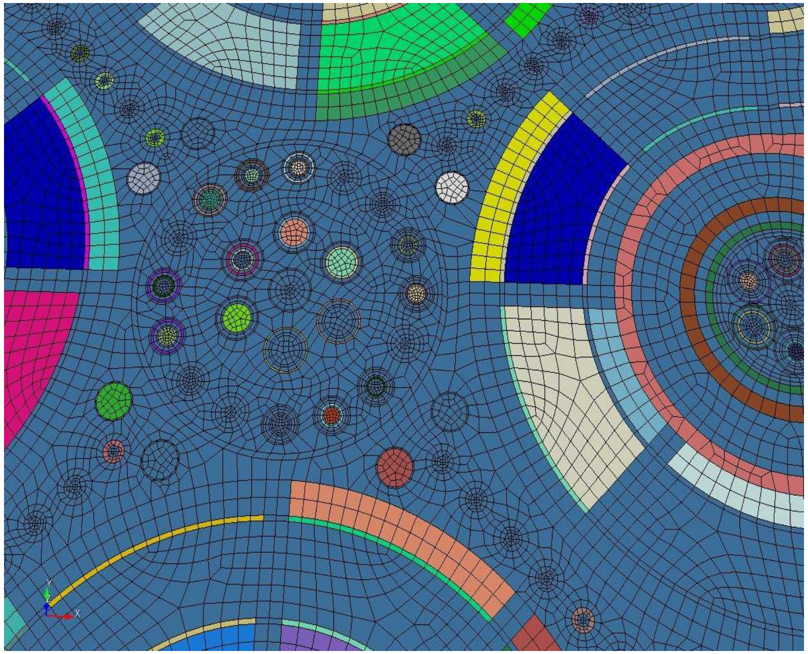


# INL Reactor Model



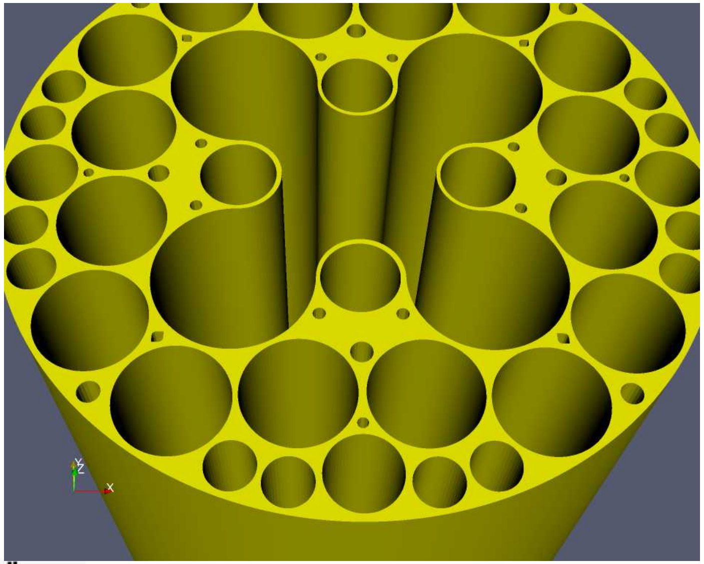

| Number of CPUs | 1 | 3 | 4 | 10 | 20 | 40 | 80 |
|---|---|---|---|---|---|---|---|
| Seconds | 745 | 393 | 363 | 265 | 259 | 246 | 232 |

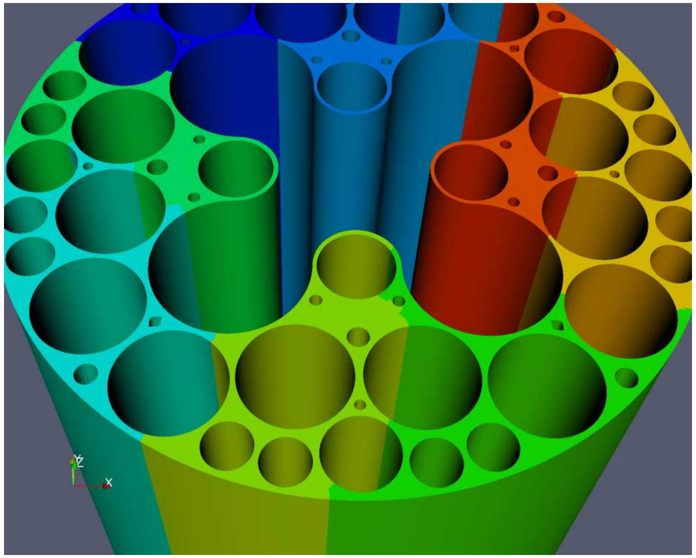
10.5 million elements



# Load Balancing

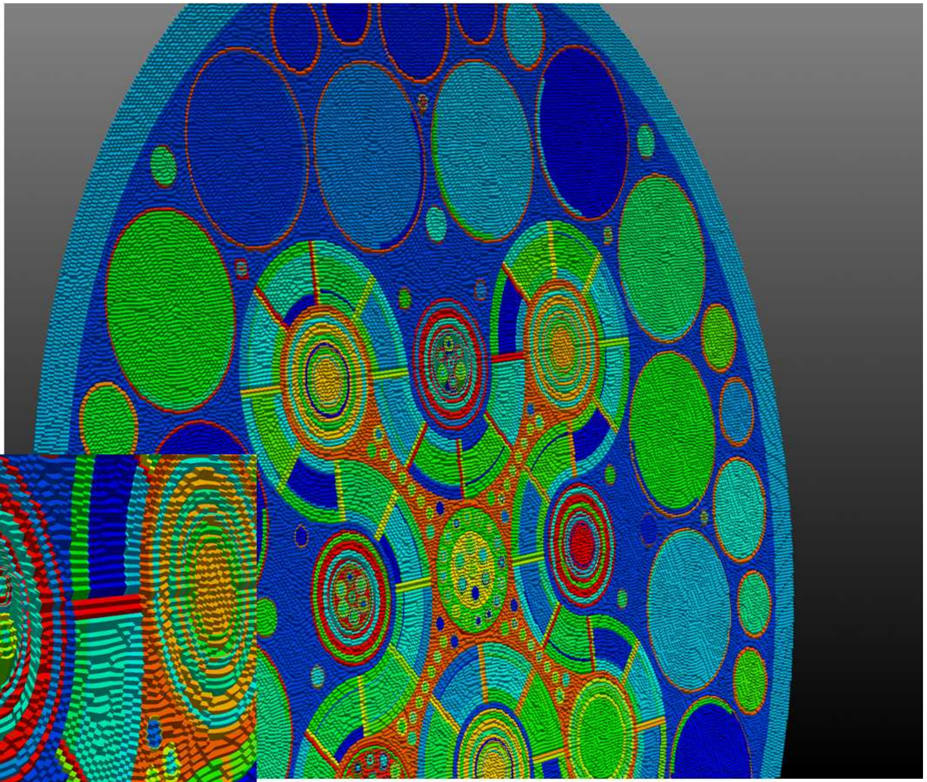
- **Source surface partitioning**
  - 1.5 million hex elements
  - Single volume: 54 sec. to mesh on one processor
  - 8 sub-volumes: 15 sec. to mesh on two processors

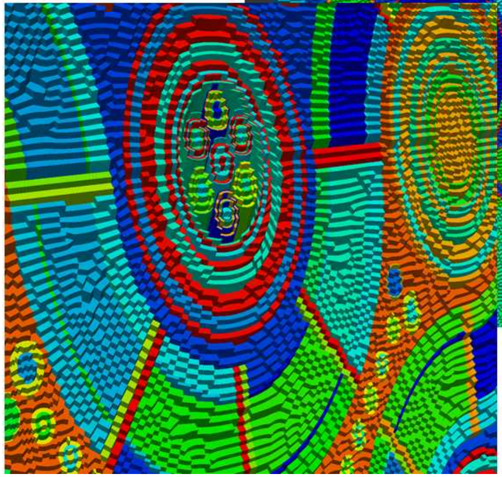




# Parallel Hex Meshing

- **Big models**
  - 33.4 million hexes
  - Too big for serial Cubit
  - Too big for most desktop workstations





Mesh generated by Philippe Pebay (SNL), Mike Stephenson (MBS&A)
Model courtesy of Scott Lucas, Glen Hansen (INEL)

# Conclusions

- **BioMesh3D**
- **Hexahedral mesh generation for biomedical models**
- **Parallel mesh generation**
  - pCAMAL is scalable for large models
  - Serial decomposition controls overall mesh time
  - Future work: surface meshing in parallel