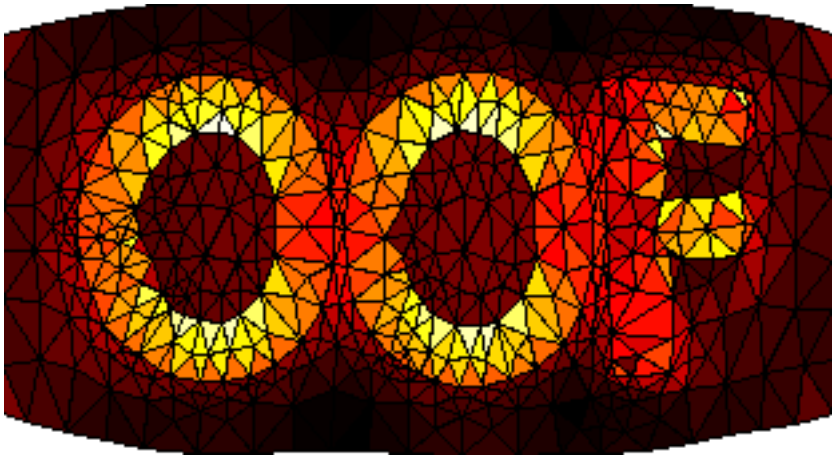


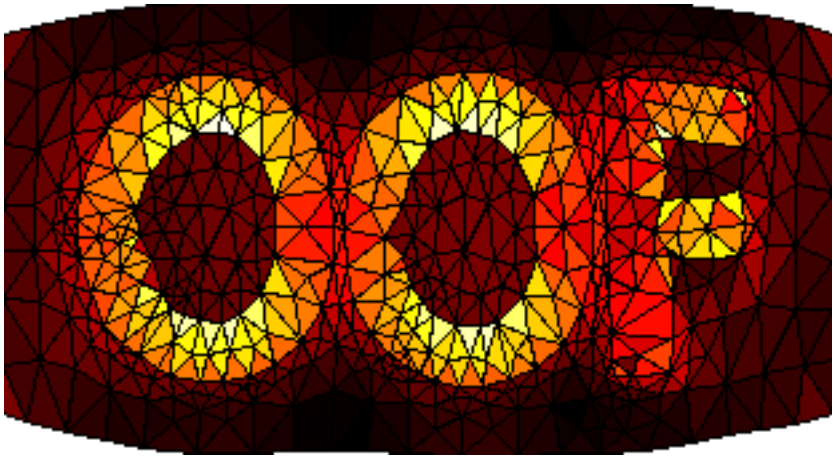
Overview & Philosophy

Steve Langer	NIST ITL
Ed Fuller, Andy Roosen	NIST MSEL
Craig Carter	MIT Mat. Sci. & Eng.
Andrew Reid	MIT/NIST guest researcher
Edwin Garcia	MIT grad student
Rajesh Raghavan	MIT postdoc
Mark Locatelli	NRC postdoc, Ceramics
Nita Parekh	CTCMS/UConn postdoc
Daniel Vlacich	Montgomery Blair HS

NIST CENTENNIAL 1901-2001

INDUSTRY WEEK IW

1999 Technology of the year



Organizations

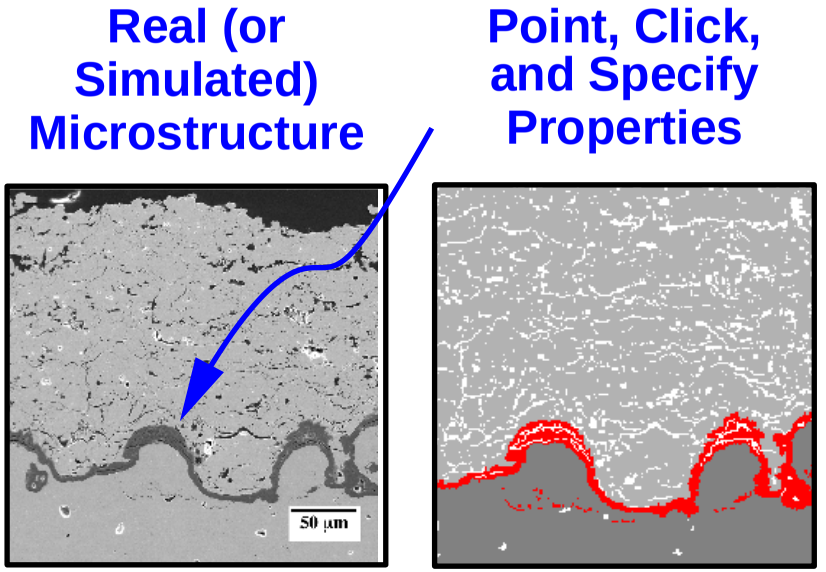
- **NIST** Center for Theoretical & Computational Computer Science
- **NIST** Materials Science & Engineering Laboratory
- **NIST** Information Technology Laboratory
- **MIT** Department of Materials Science & Engineering

$ **US DOE** Advanced Turbine Systems

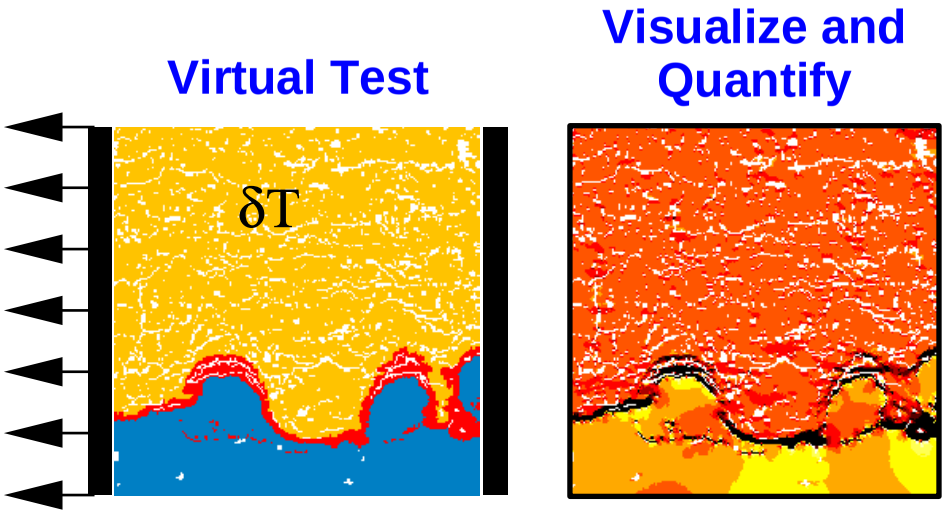
Goals of the OOF Project

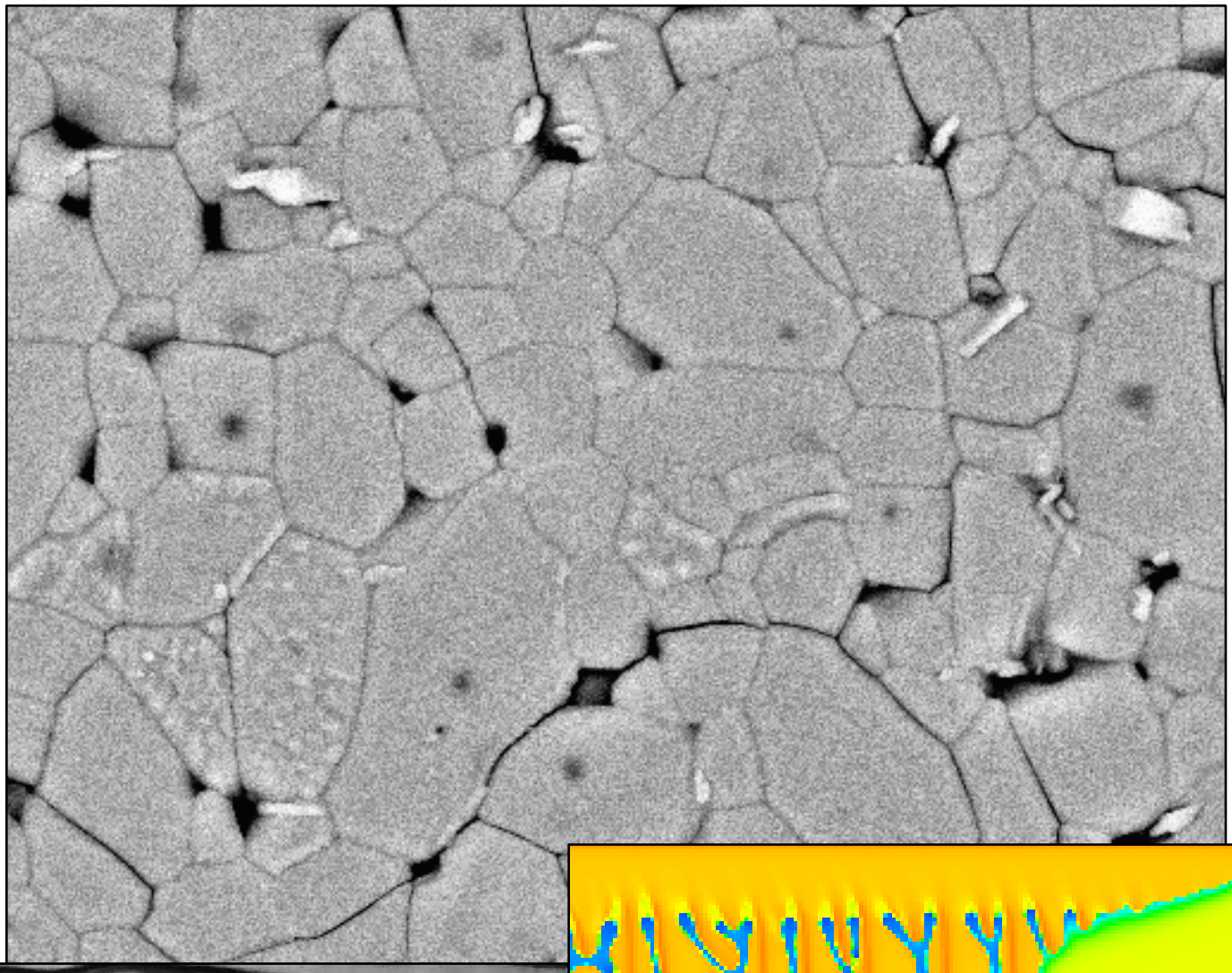
- Easy–to–use tools for analyzing the effects of microstructure on material properties.
- Input from micrographs and simulations, using all available microstructural data. No mean–field approximations.
- User defined constitutive relations.
- Convenient “what if?” experiments.
- Extensibility.
- Generality.
- We are successful if OOF is used in unanticipated ways.

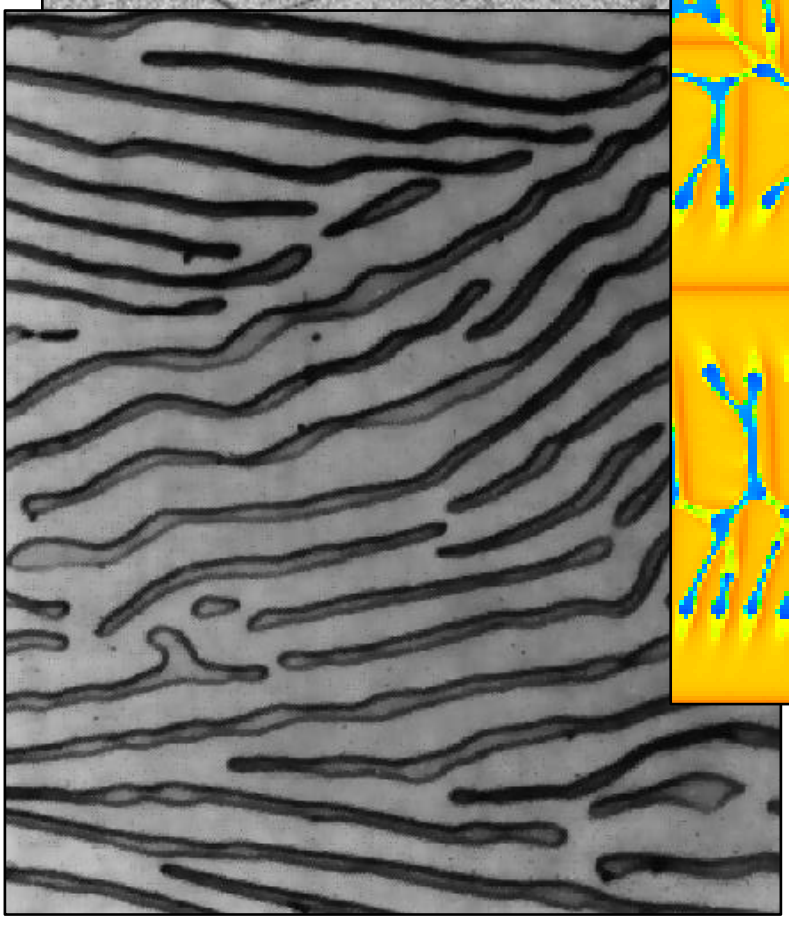
ppm2oof: a tool to convert an image or a micrograph of a heterogeneous, multi–component material into a finite element mesh with constitutive properties specified by the user.

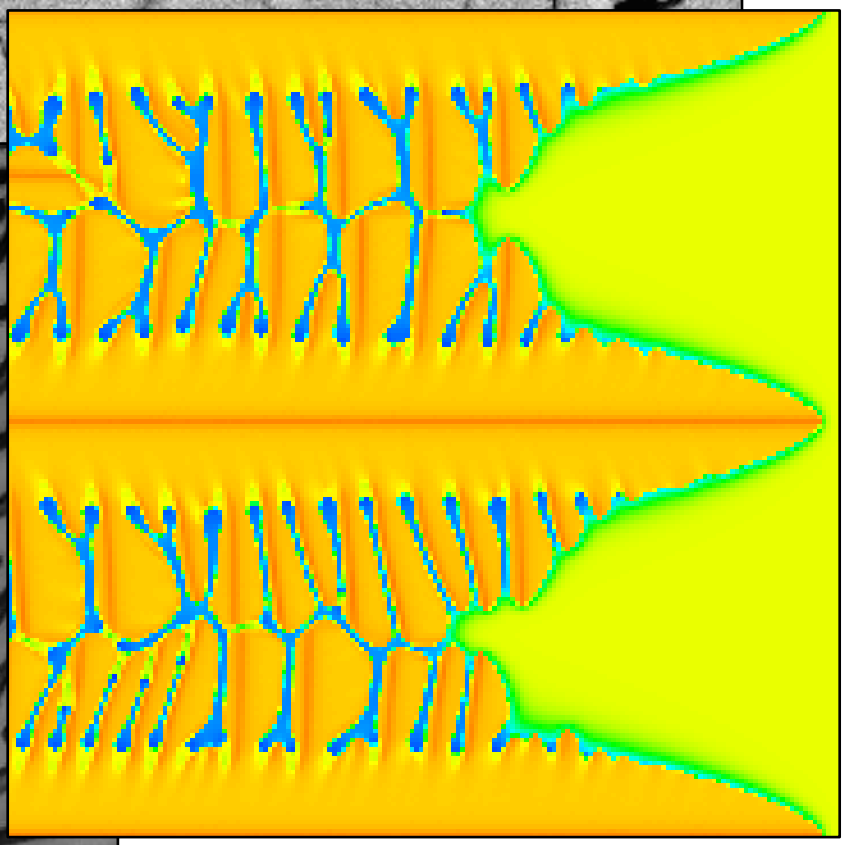


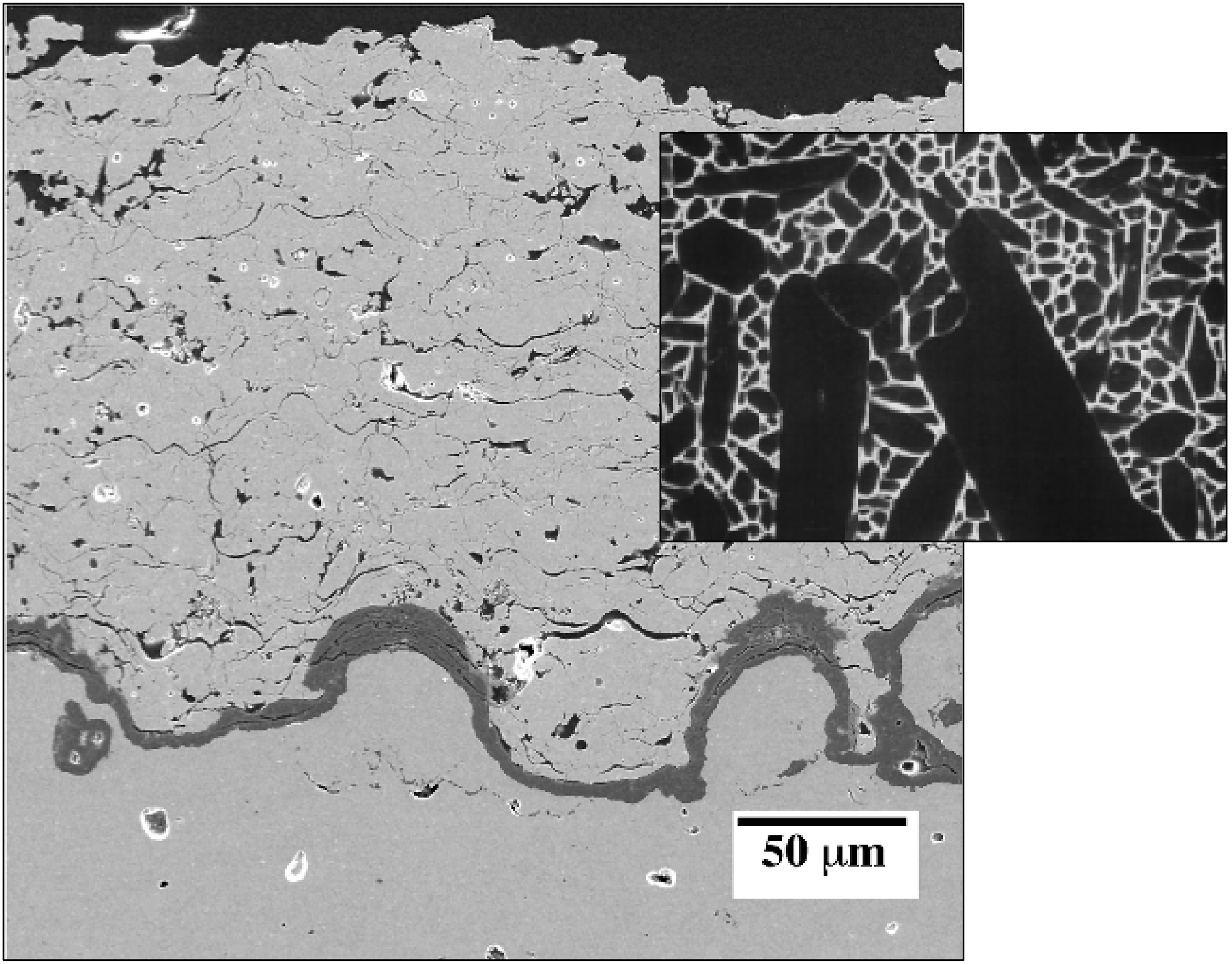
oof: a tool to test physical properties and to investigate the influence of microstructure on macroscopic behavior, via finite element analysis.

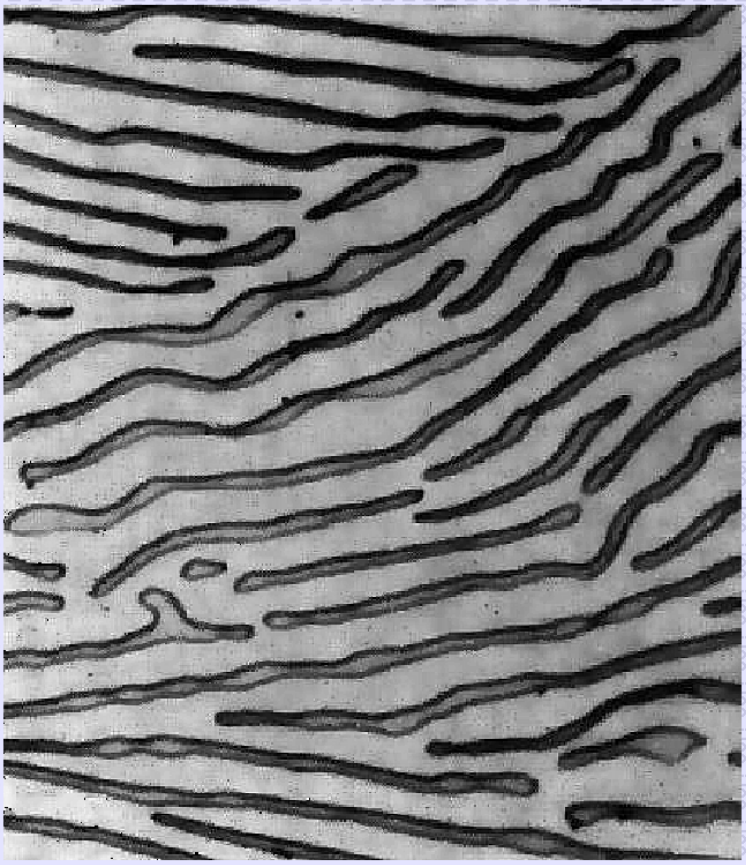


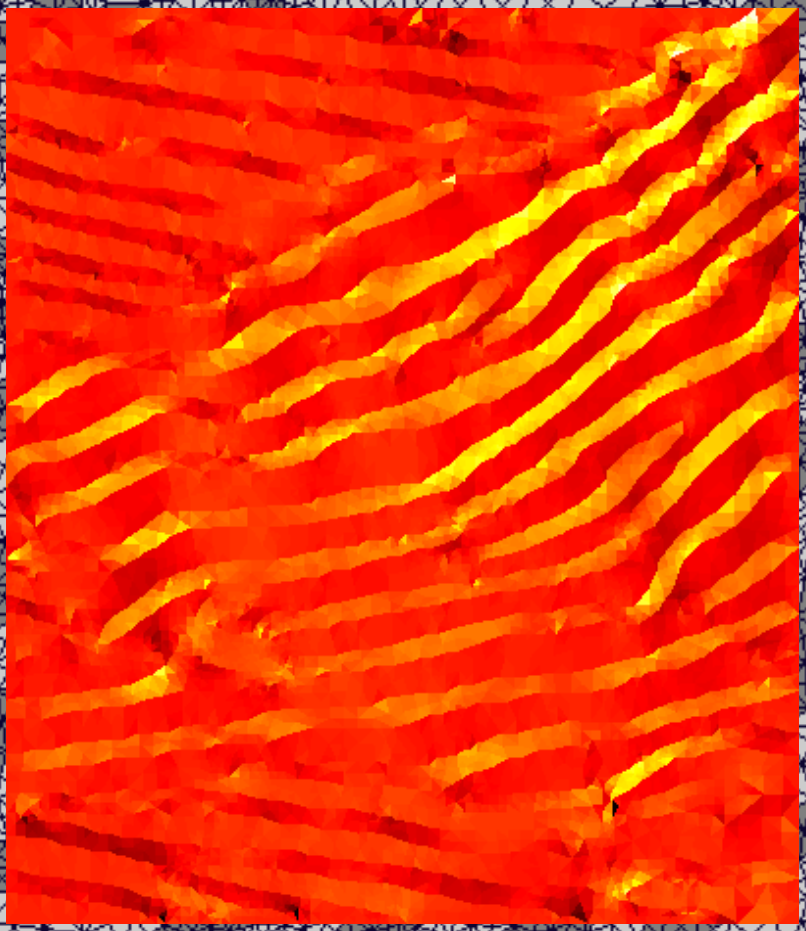


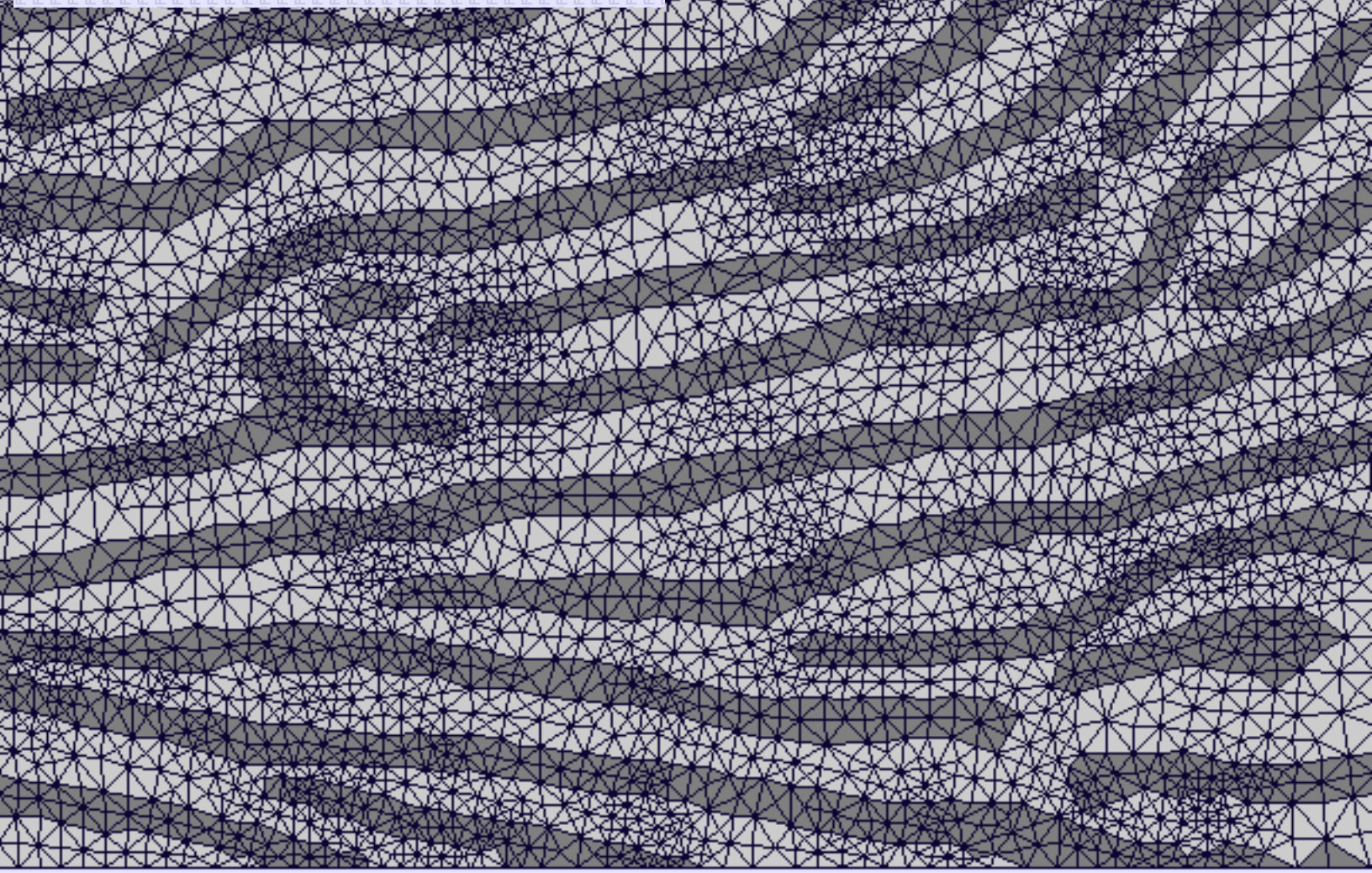




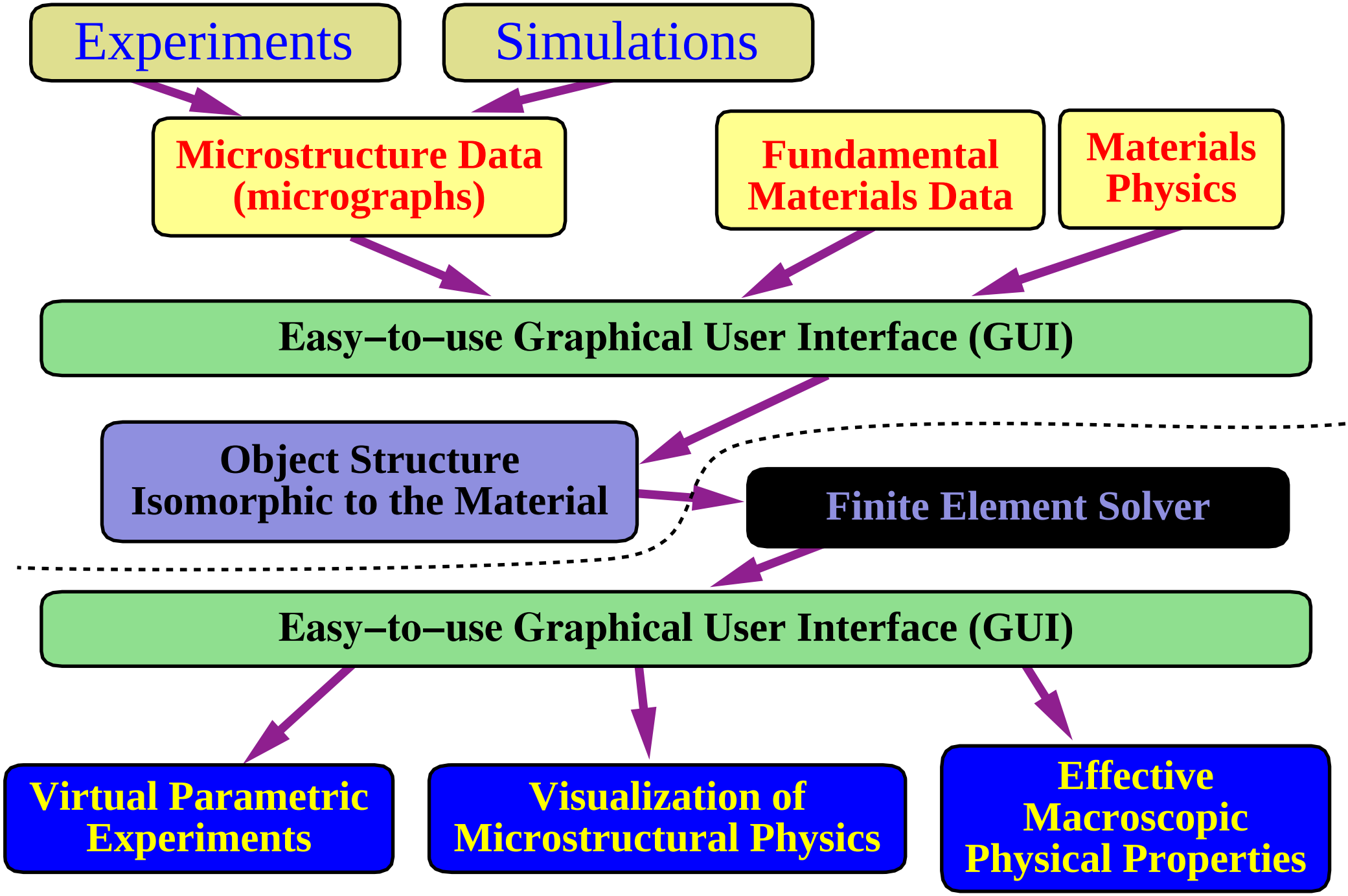

50 μm







Building a Microstructural Model

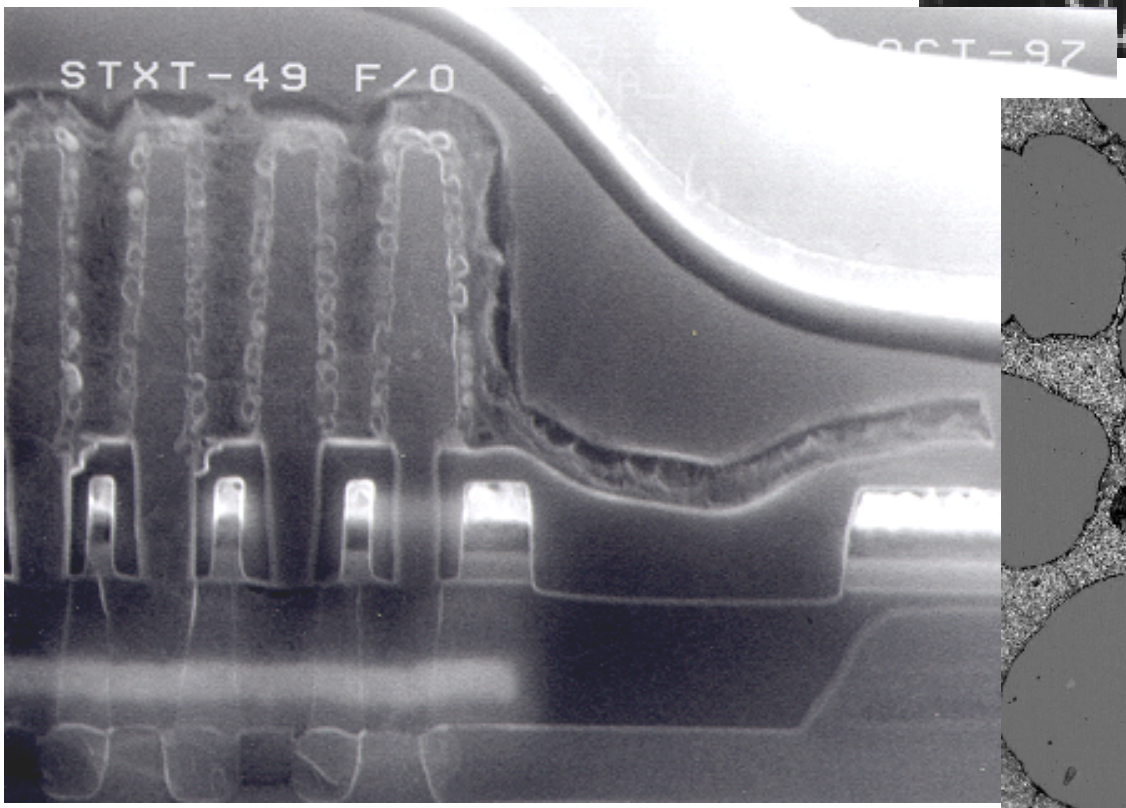


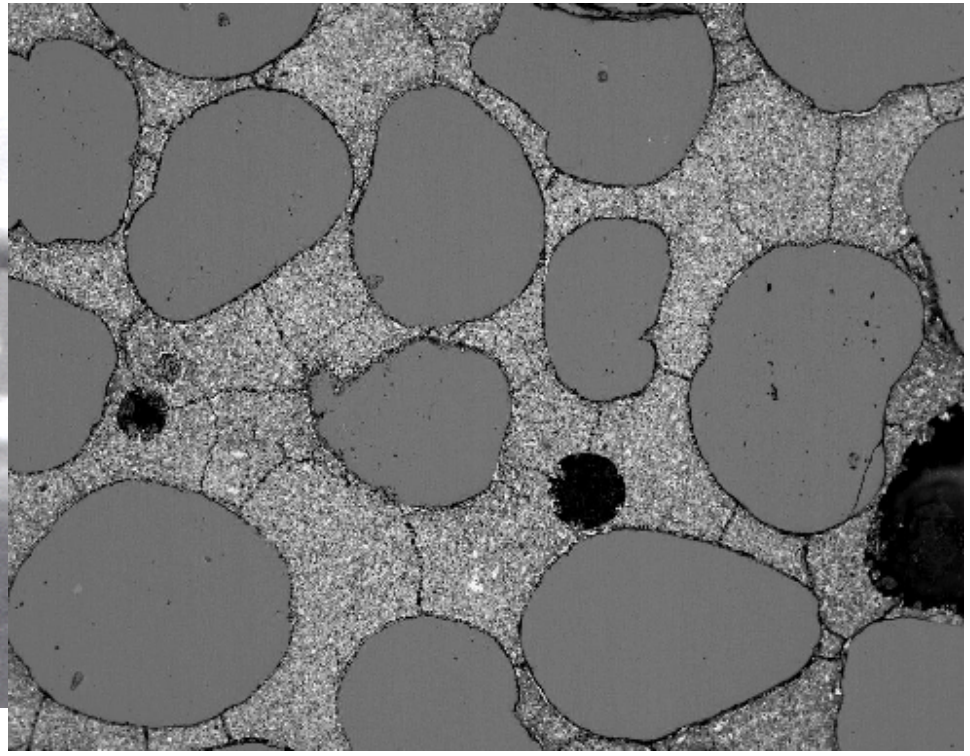
Why OOF?

- Commercial finite element packages work best for large–scale systems with regularly shaped components.

Why OOF?

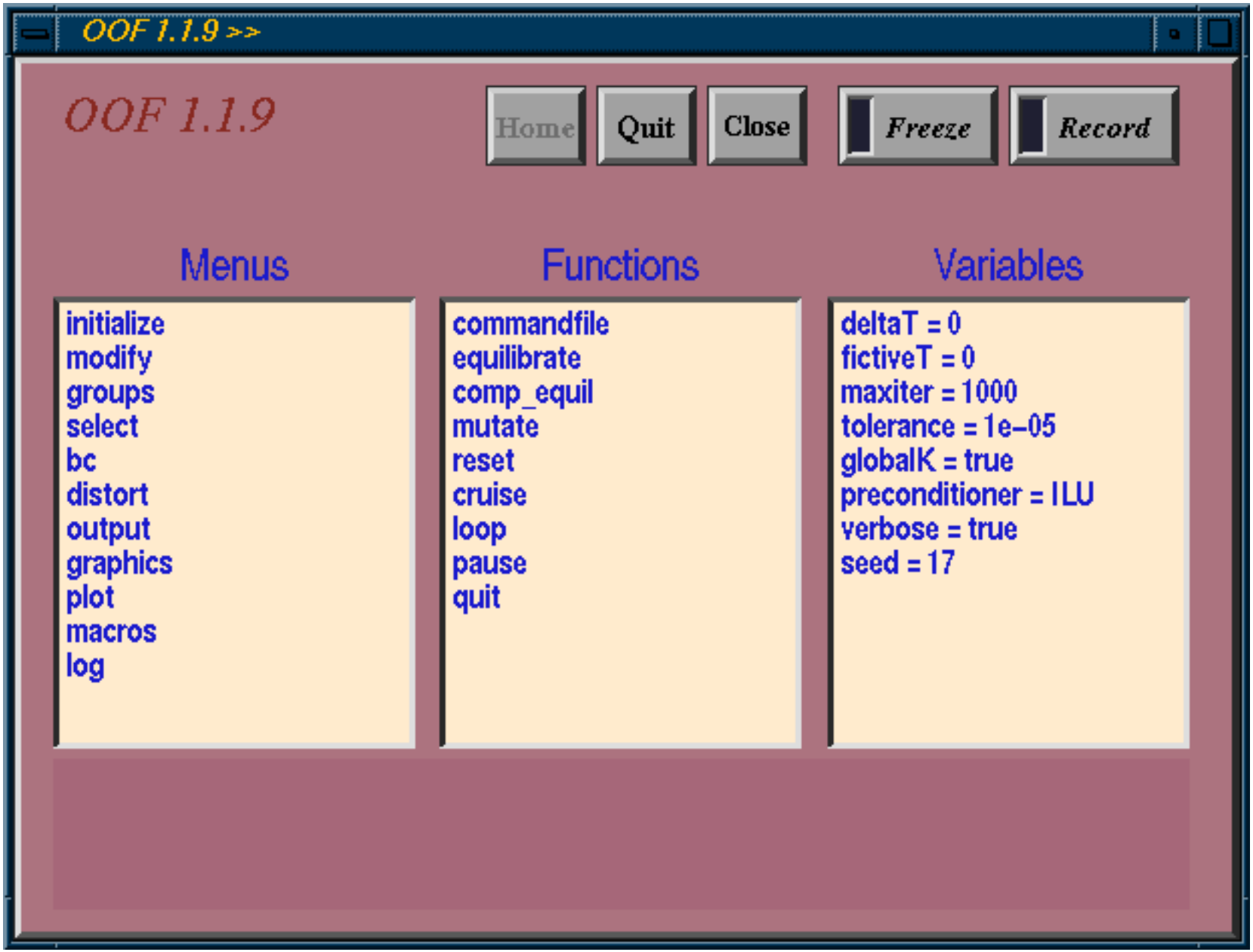
- Materials systems are small–scale and disordered.





Why OOF?

- OOF is designed for materials scientists
- OOF is easy to use.



Current Capabilities

PPM2OOF:

- Modify images.
- Select pixels.
- Assign material properties to pixels.
- Create unstructured 2–dimensional triangular mesh.
- Adapt mesh to material boundaries.
    - refine elements.
    - move nodes.
    - swap edges.
- Define element and pixel groups.
- Define “active areas” to localize operations.

Current Capabilities

OOF:

- Solve unstructured two–dimensional triangular mesh.
- Linear elasticity with thermal expansion.
- Thermal conductivity.
- Any crystal symmetry:
  - isotropic.
  - hexagonal.
  - etc.
  - new crystal symmetries added easily .
- Plane stress or plane strain.
- Simple models of fracture.
- Designer elements.
- Boundary conditions:
  - fixed displacement, temperature.
  - free.
  - constrained motion.
  - any combination of the above.

Current Capabilities

OOF, continued:

- Distortions:
  - applied displacement, temperature.
  - applied force, heat flux.
- Mesh creation:
  - Unstructured mesh from ppm2oof.
  - Generate uniform mesh.
  - Modify material properties of existing mesh.
- Output
  - Maps of stress, strain, temperature, energy density, etc.
  - Statistics of stress, strain, etc for whole mesh or element group
  - Plot of stress, strain, etc along a cross section.
  - Stress, strain, etc at selected elements.
  - Forces at selected nodes.

Current Capabilities

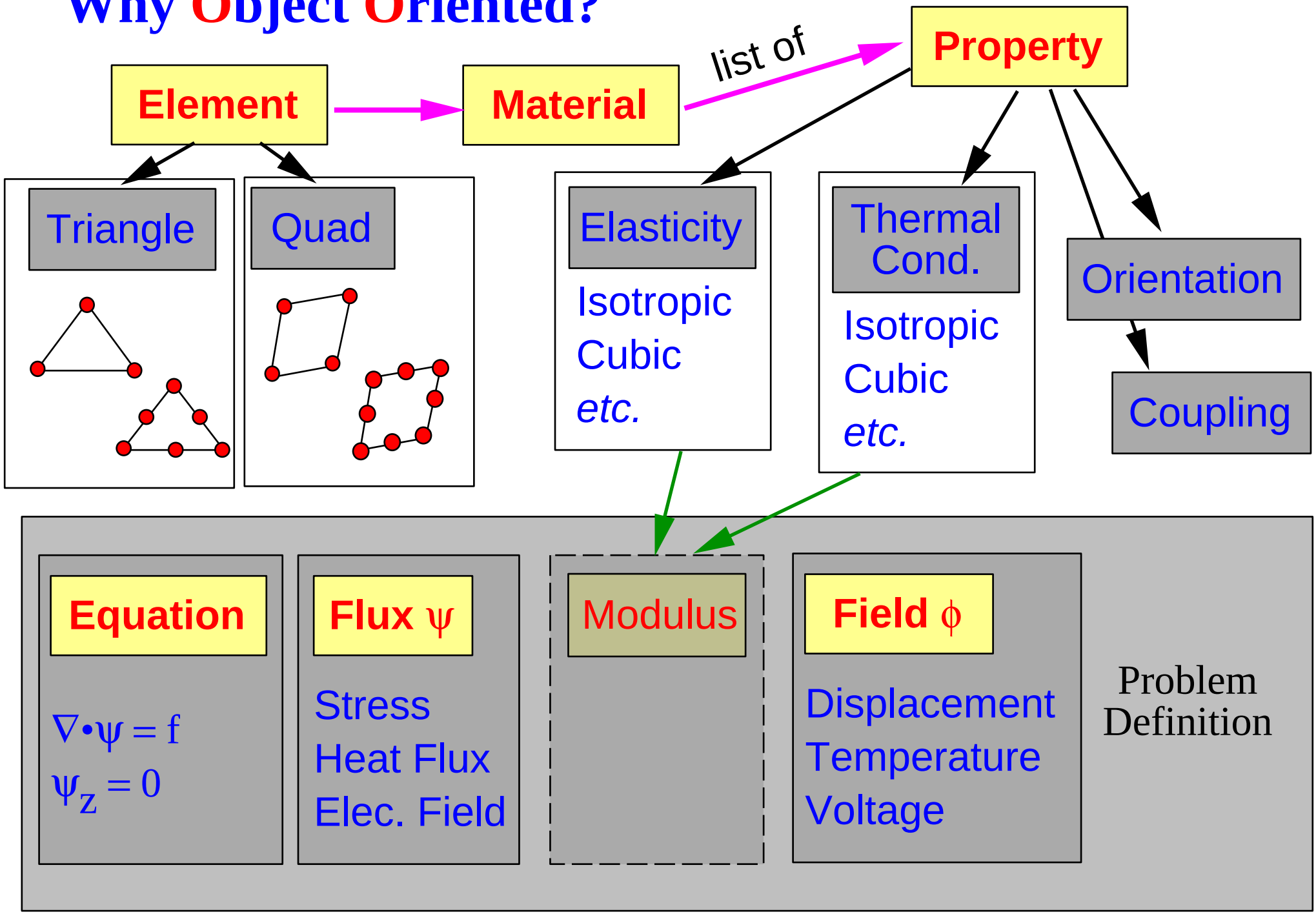
OOF & PPM2OOF:

- Controlled from graphical user interface or script.
  - menu driven.
  - commands in tree of submenus.
- Run on SGI, Sun, Alpha, and PC (Linux).

OOF2ABAQUS:

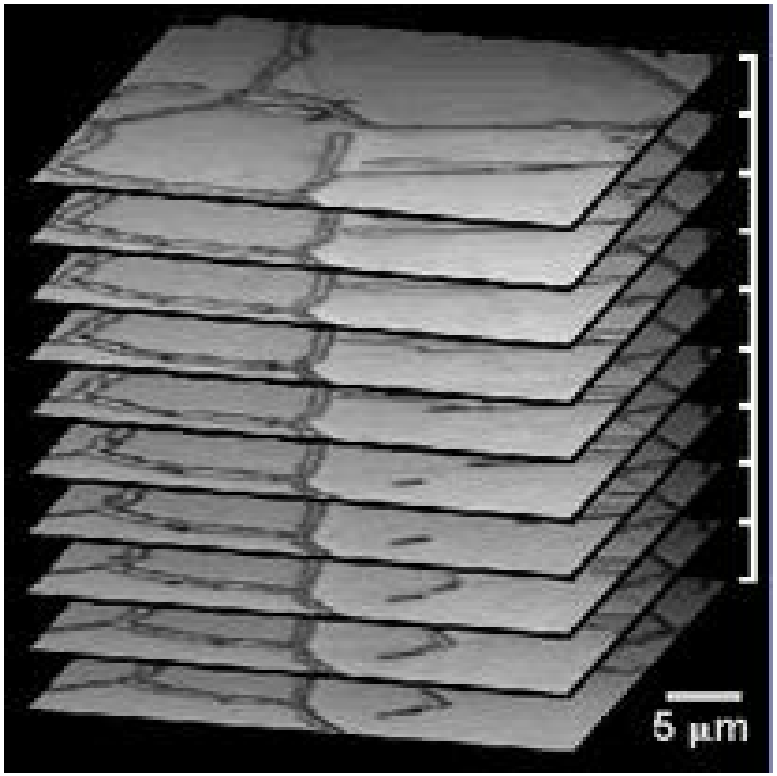
- Converts OOF files into ABAQUS format
- Exports geometrical information
  (Nodes, elements, node groups, element groups)
- Can write 3–node or 6–node triangular elements
- Material specification is done in ABAQUS

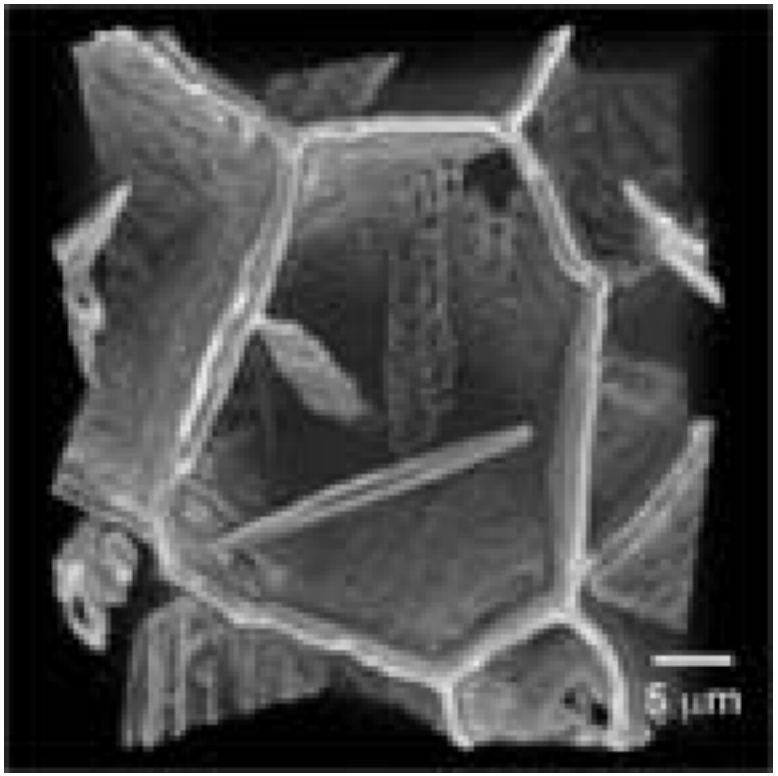
Why Object Oriented?



Why 2D?

- There aren’t many three dimensional micrographs.
- There are lots of two dimensional micrographs.
- Two dimensions is easier.
- Three dimensions will come later...



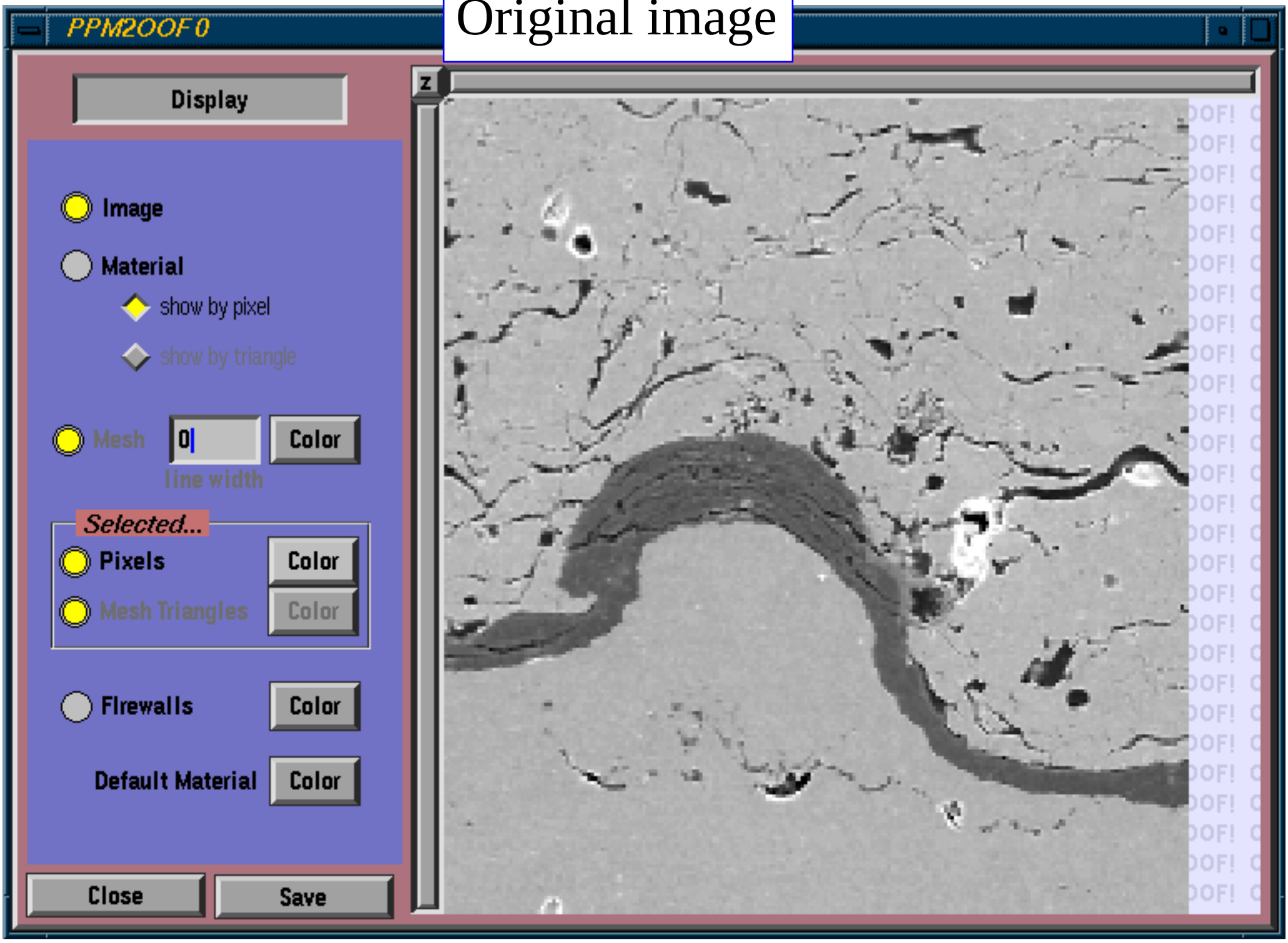


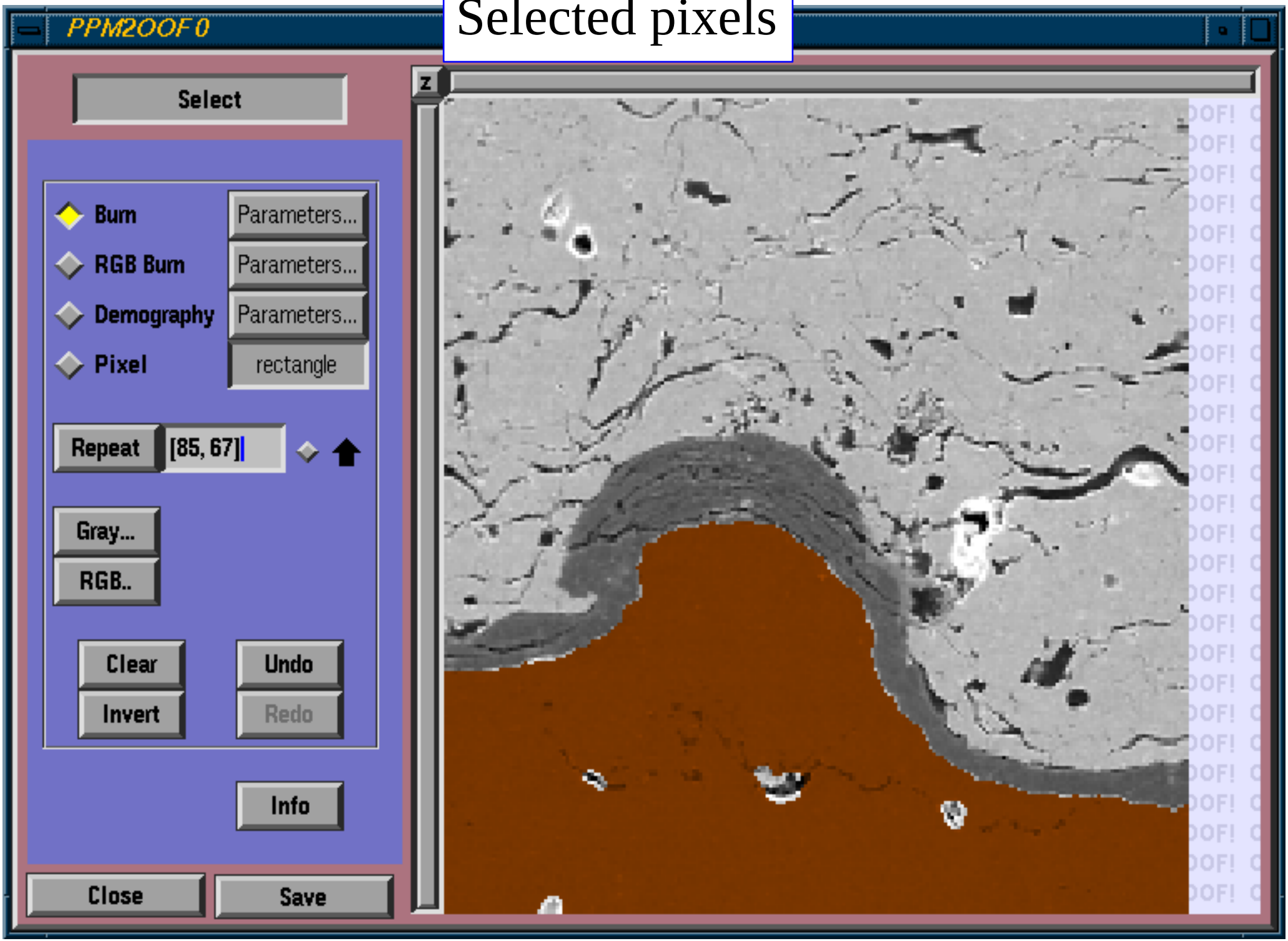
G. Spanos, NRL

http://mstd.nrl.navy.mil/6320/6324/3dmicrostructures.html

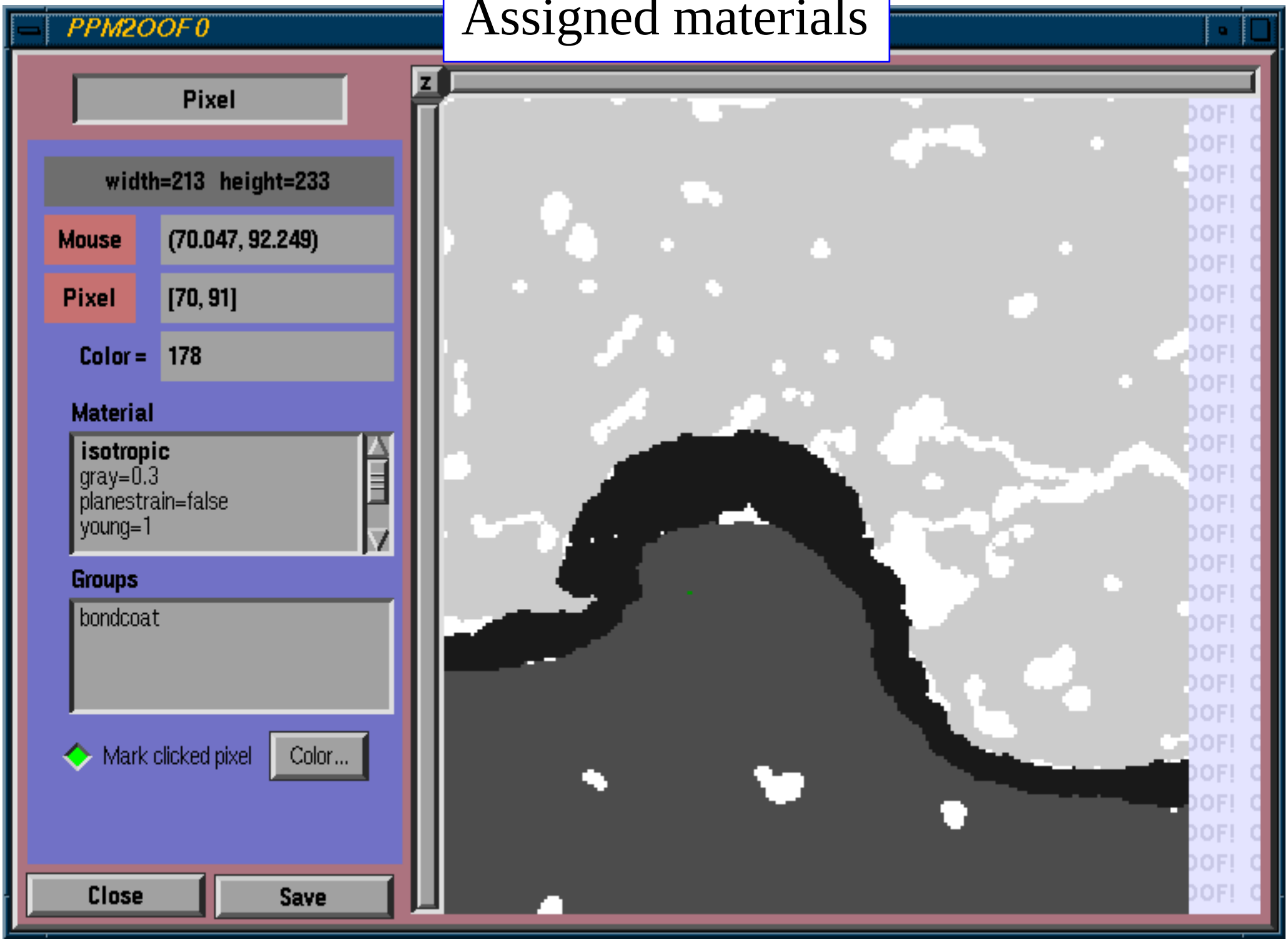
PPM2OOF in a nutshell

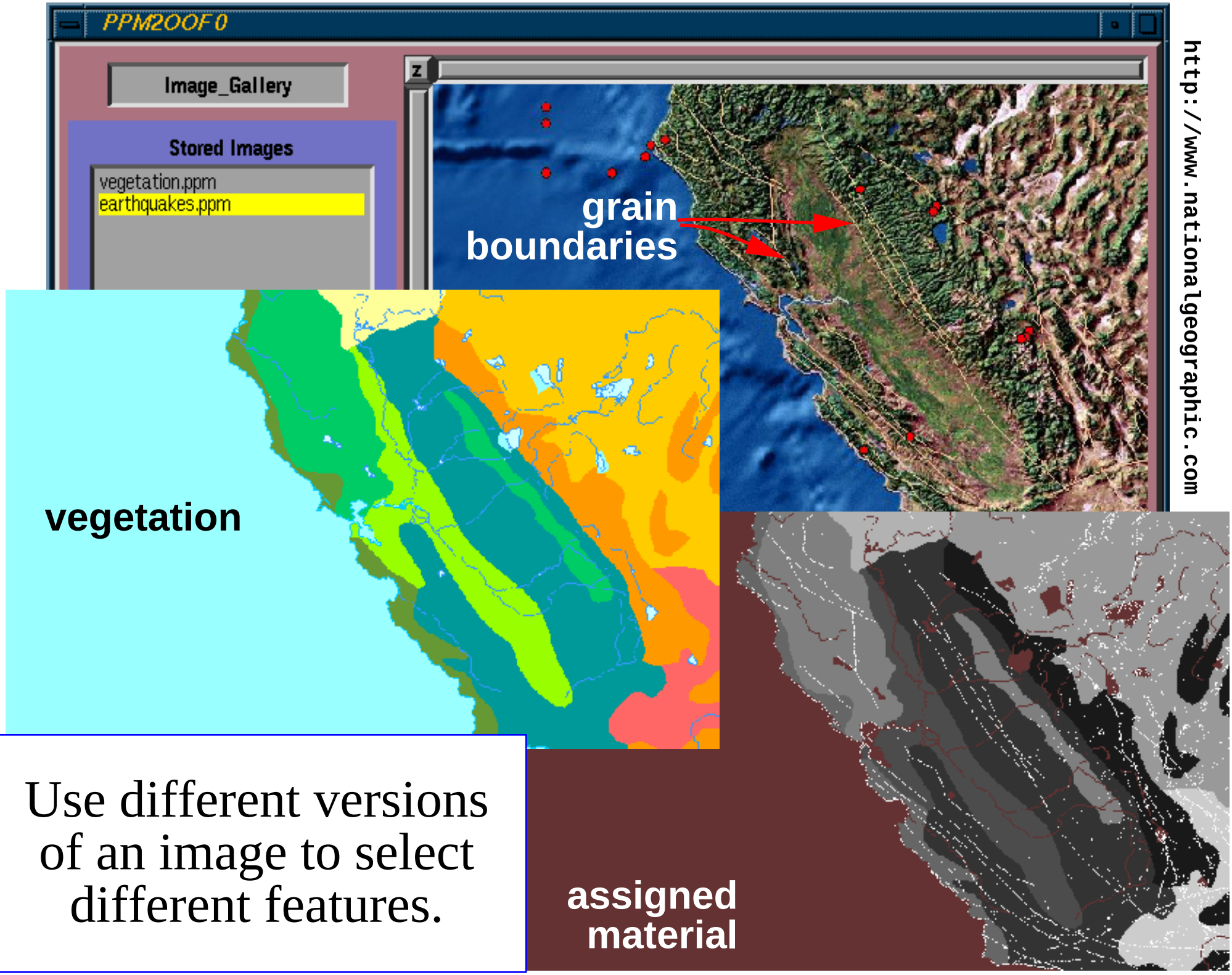
1. Select pixels in the image
2. Assign material properties to groups of pixels
3. Create a mesh
4. Adapt and refine the mesh


Original image
PPM2OOF0
Display
Image
Material
show by pixel
show by triangle
Mesh
0
Color
line width
Selected...
Pixels
Color
Mesh Triangles
Color
Firewalls
Color
Default Material
Color
Close
Save
z


Selected pixels
PPM2OOF 0
Select
Burn
Parameters...
RGB Burn
Parameters...
Demography
Parameters...
Pixel
rectangle
Repeat
[85, 67]
Gray...
RGB..
Clear
Undo
Invert
Redo
Info
Close
Save
z

Assigned materials

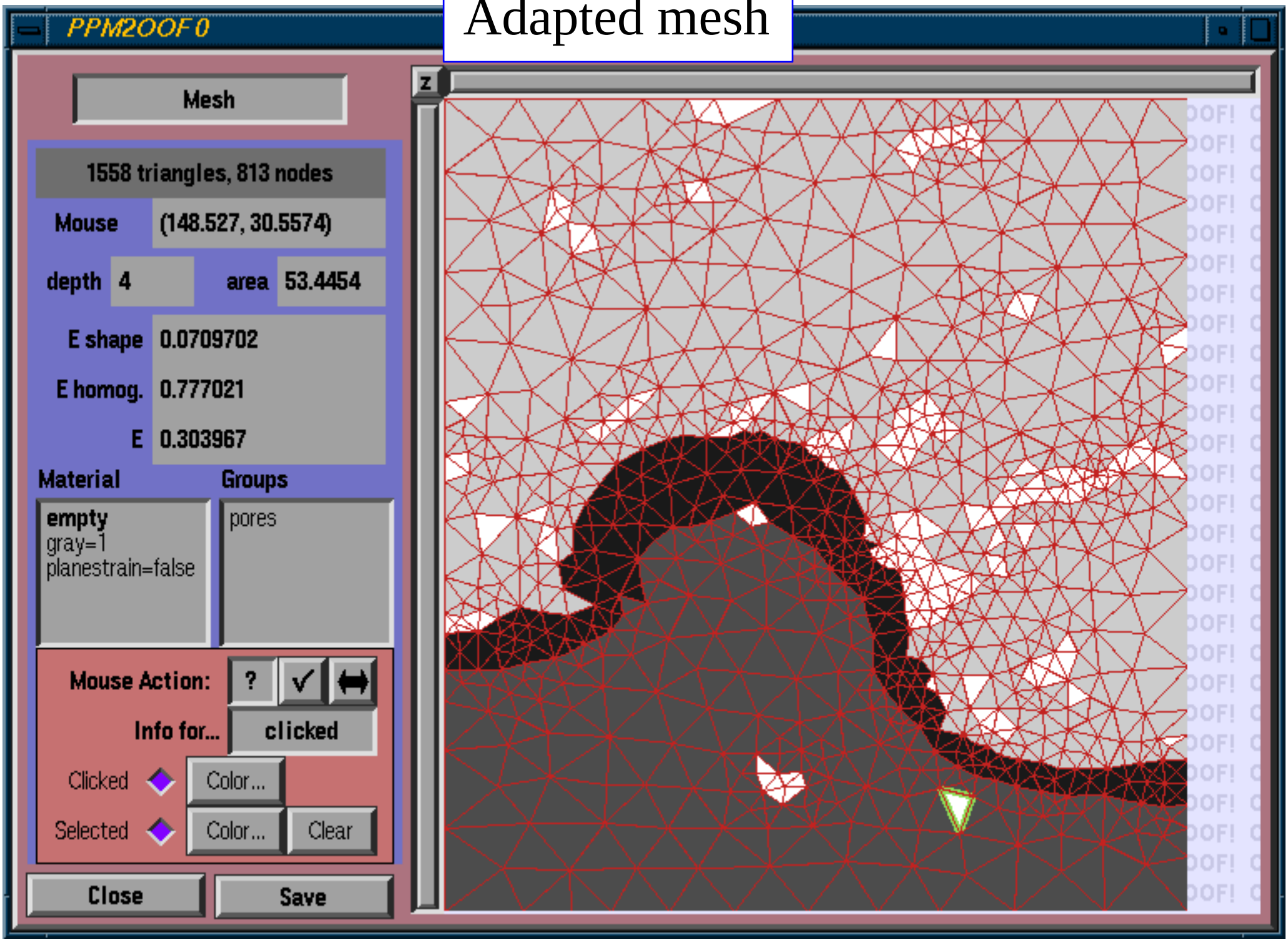




Use different versions of an image to select different features.

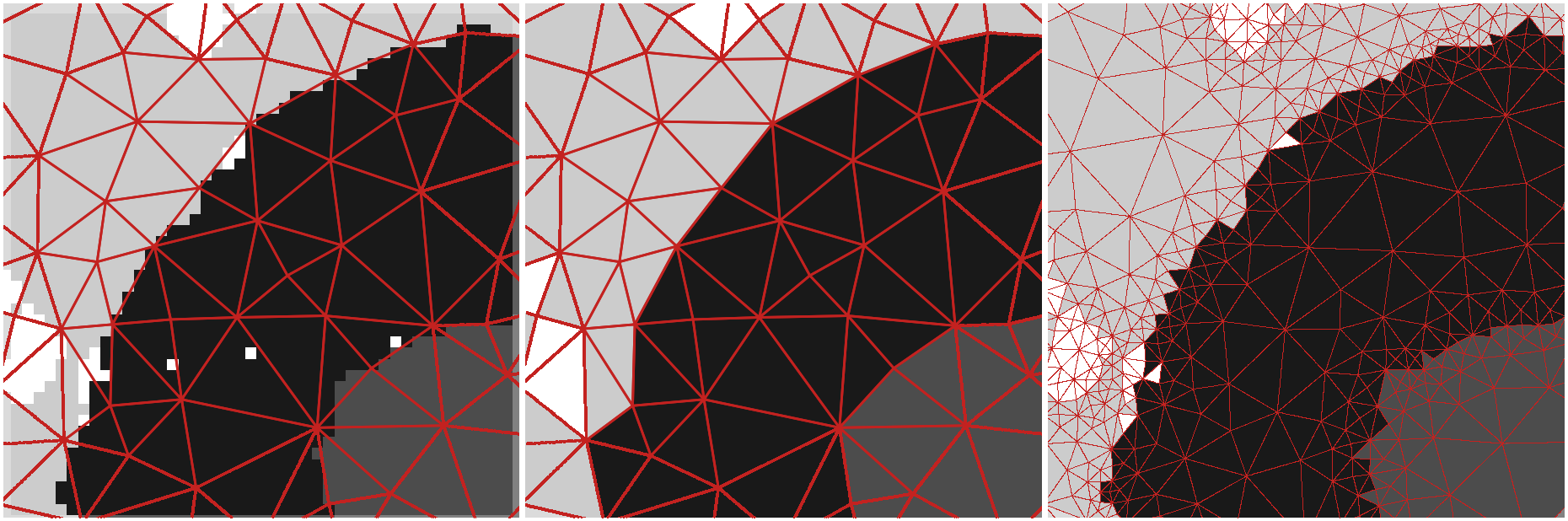
Uniform mesh

Adapted mesh



Rules for Creating Meshes from Images

- Remember:
  - Image is an approximation to the material.
  - Assigned materials are an approximation to the image.
  - Mesh is an approximation to the assigned materials.
- Choose which features to resolve.
- Don’t refine too far!



oops…

Adapting a mesh to material boundaries

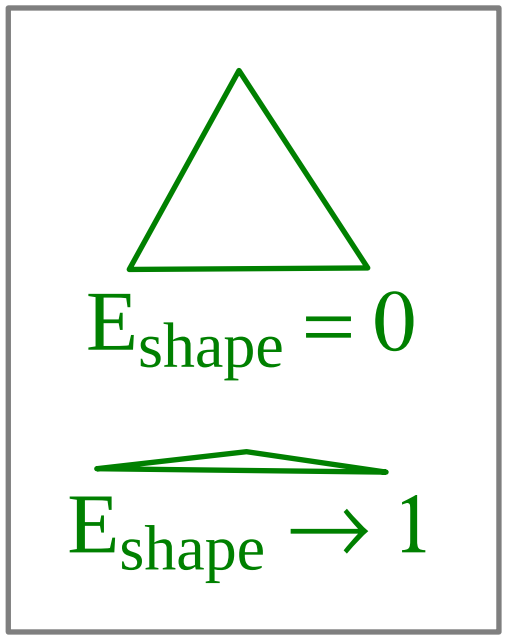
$$E = (1-\alpha)\, E_{shape} + \alpha\, E_{homogeneity}$$

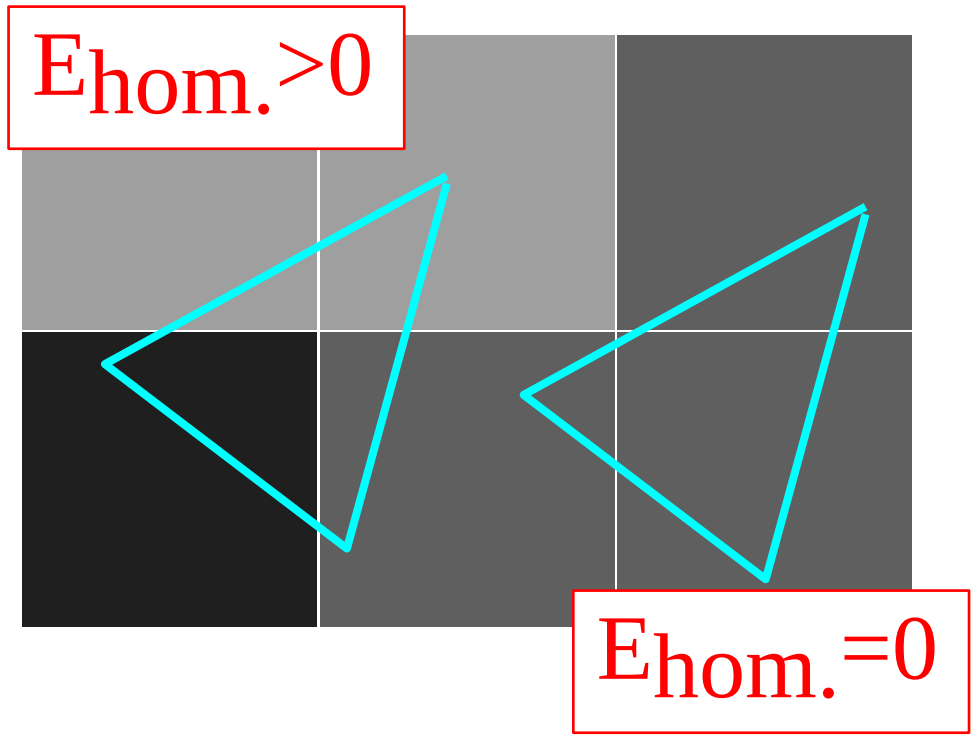
$$E_{shape} = 1 - \frac{36}{\sqrt{3}} \frac{A}{L^2}$$

$$E_{homog.} = \prod_{i=1,N} [(1 - a_i)/(1-1/N)]$$

a_i = fractional area of pixel type i

N = total number of pixel types





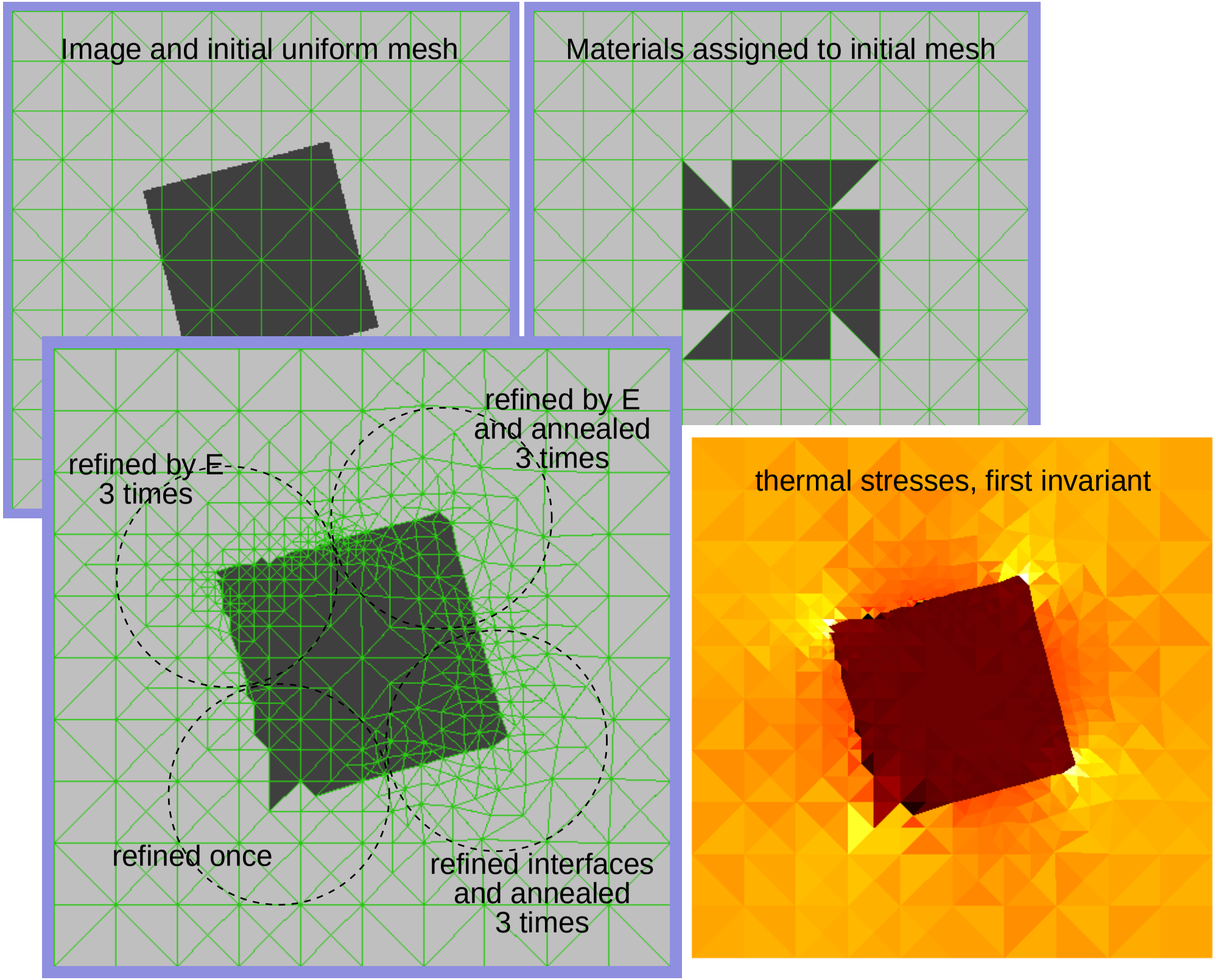
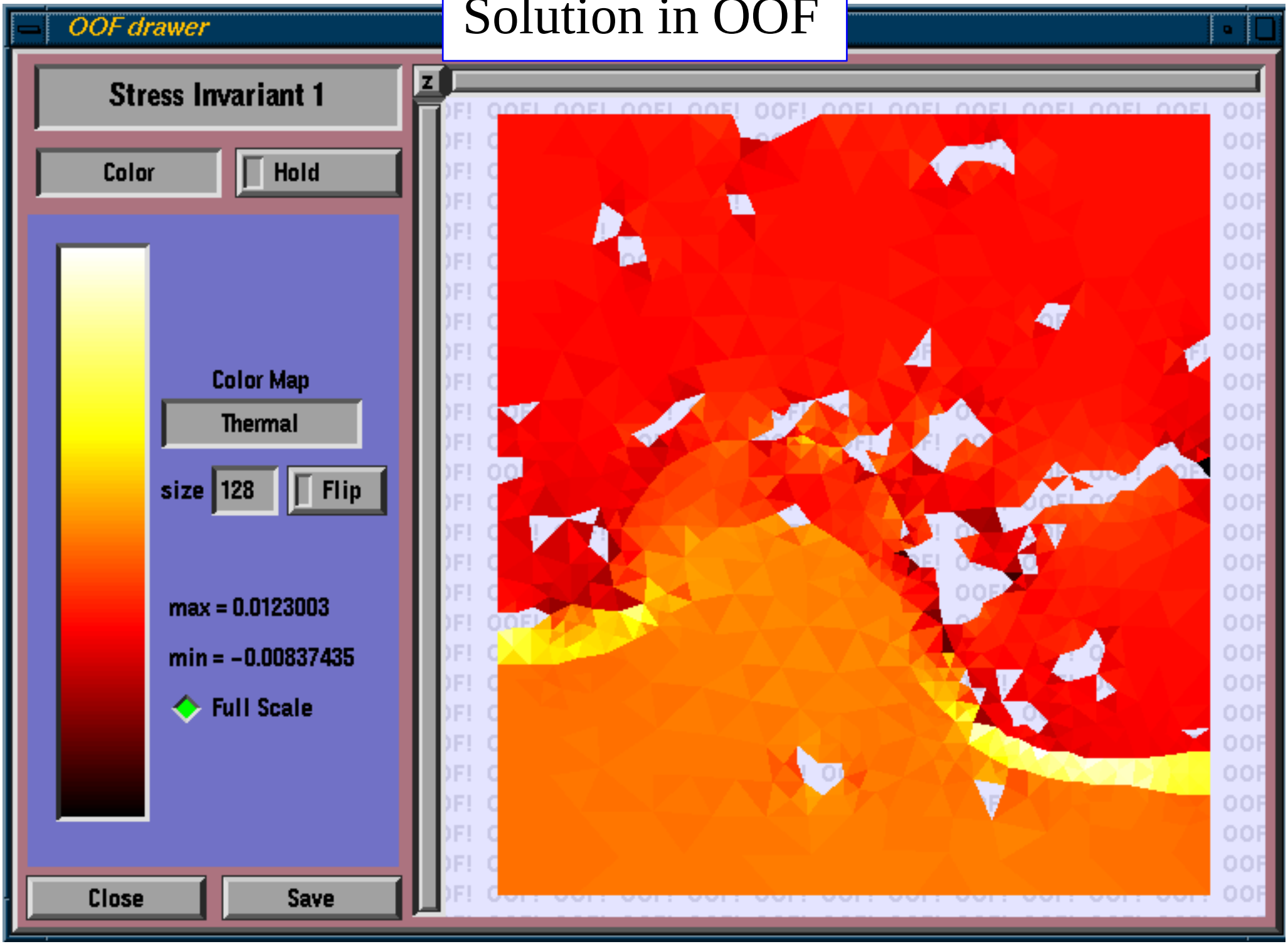


Image and initial uniform mesh
Materials assigned to initial mesh
refined by E
3 times
refined by E
and annealed
3 times
refined once
refined interfaces
and annealed
3 times
thermal stresses, first invariant

Solution in OOF



http://www.ctcms.nist.gov/oof/

- Download
  - Source code
  - Compiled binaries

    SGI, Sun, Alpha, Intel (Linux)
  - Auxilliary programs
  - Manual (html, postscript) & Tutorials
  - Old newsletters

- Join our Mailing List!