Development of a mask editor for lithography simulations using ORCAN.

Integration of Sthamas3D into ORCAN.

Cristian Tota,

Daniel Vizman,

VION-Software

ORCAN Workshop, April 26.-27. 2005, Erlangen

Mask Editor for Lithography Simulations

Technology Simulation - Lithography, Fraunhofer Institute, Erlangen
Dr. Andreas Erdmann

Requirements:

- 2D Editing
  - Rectangles, Layers, Groups, Materials
- Visualization
- Basic operations
  - Translation, rotation, deformation
- Save/load mask

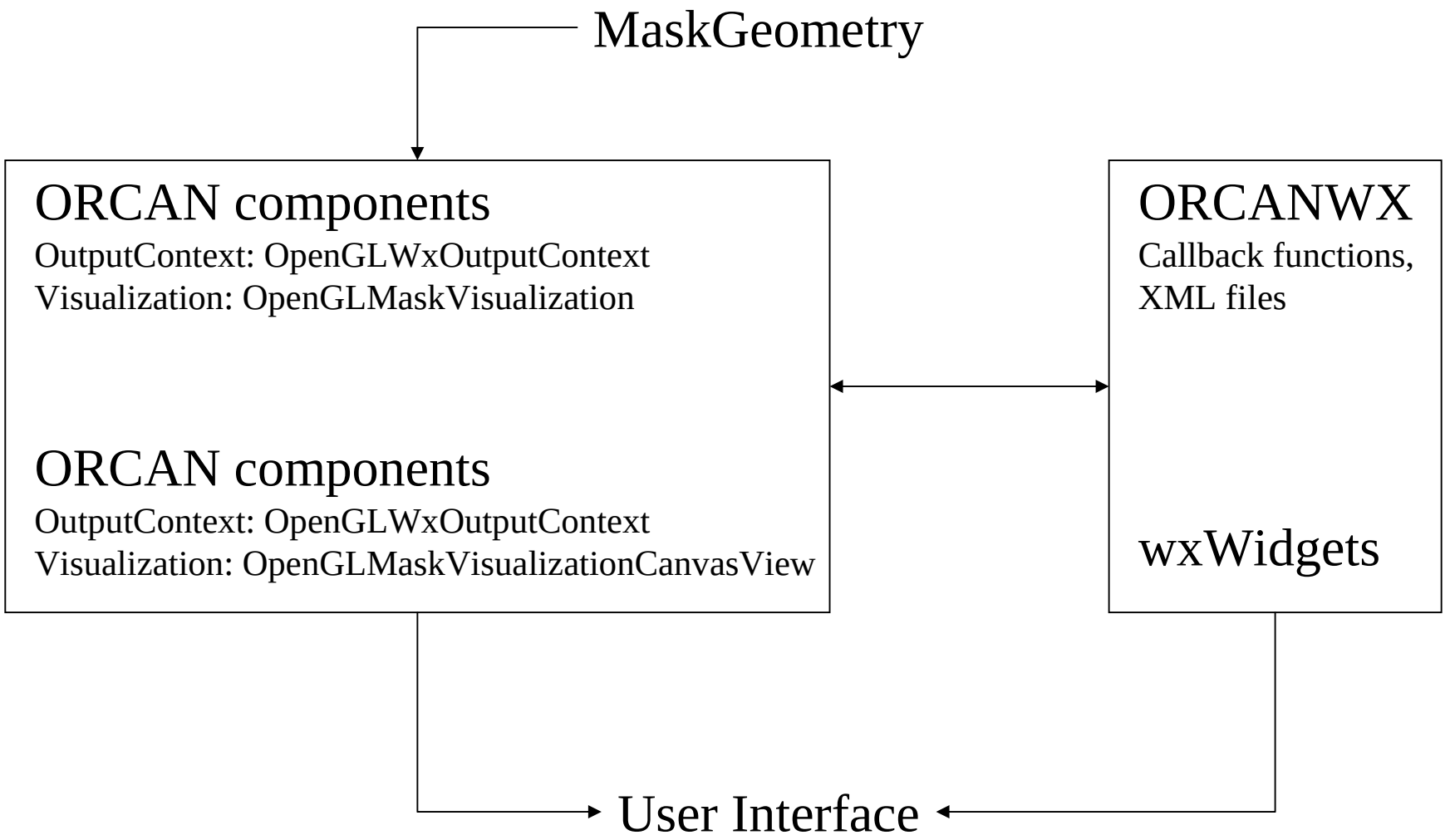
ORCAN features used:

- OutputContex
- Visualization
  - Interfaces used:
    - Default
    - Interactive
    - MaskEditor
- Graphical controls
    - XML definitions
    - PropertyMaps

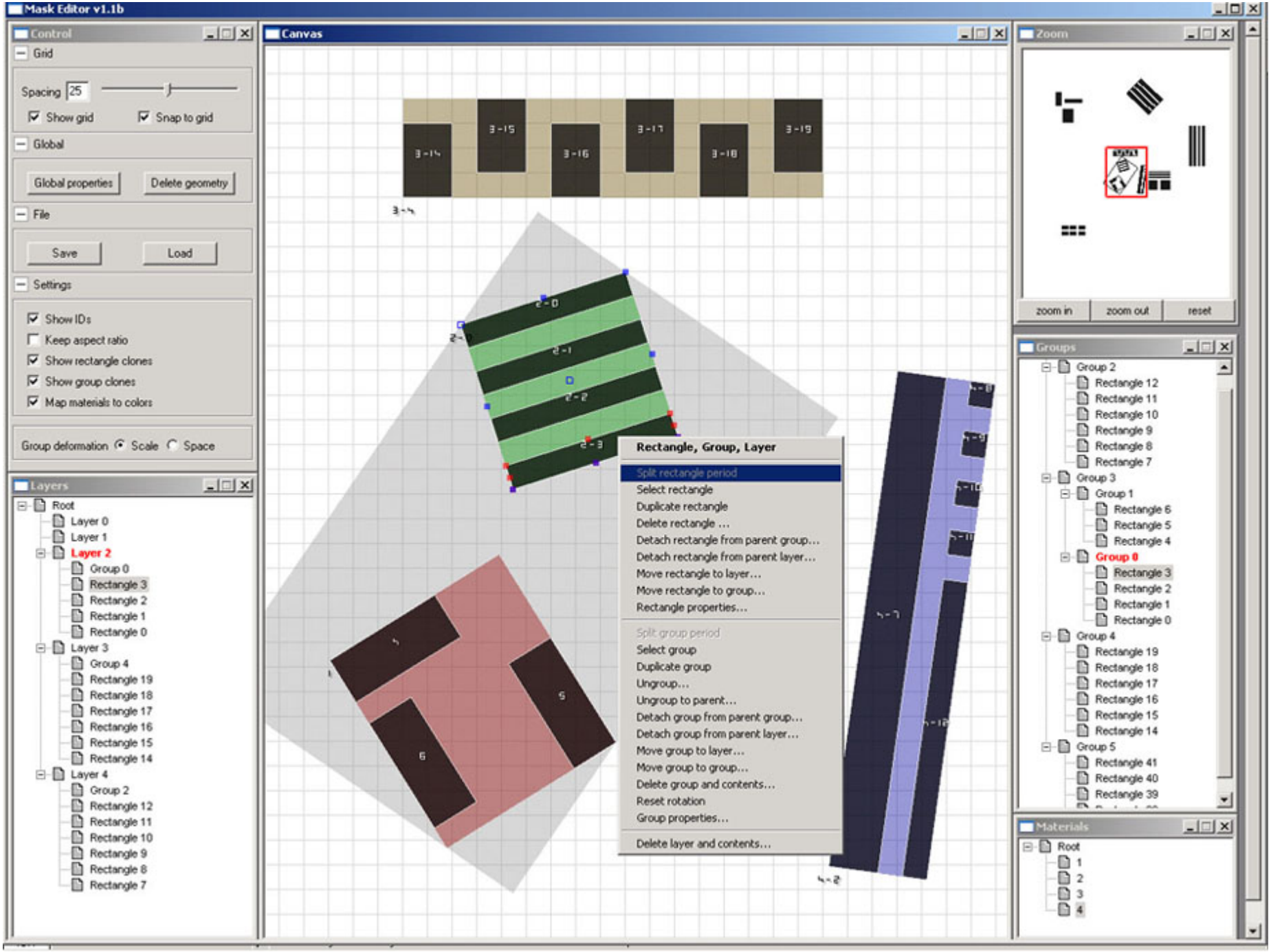
Editing, Rendering

Property Dialogs

General structure:



MaskEditor screenshot



Future development:

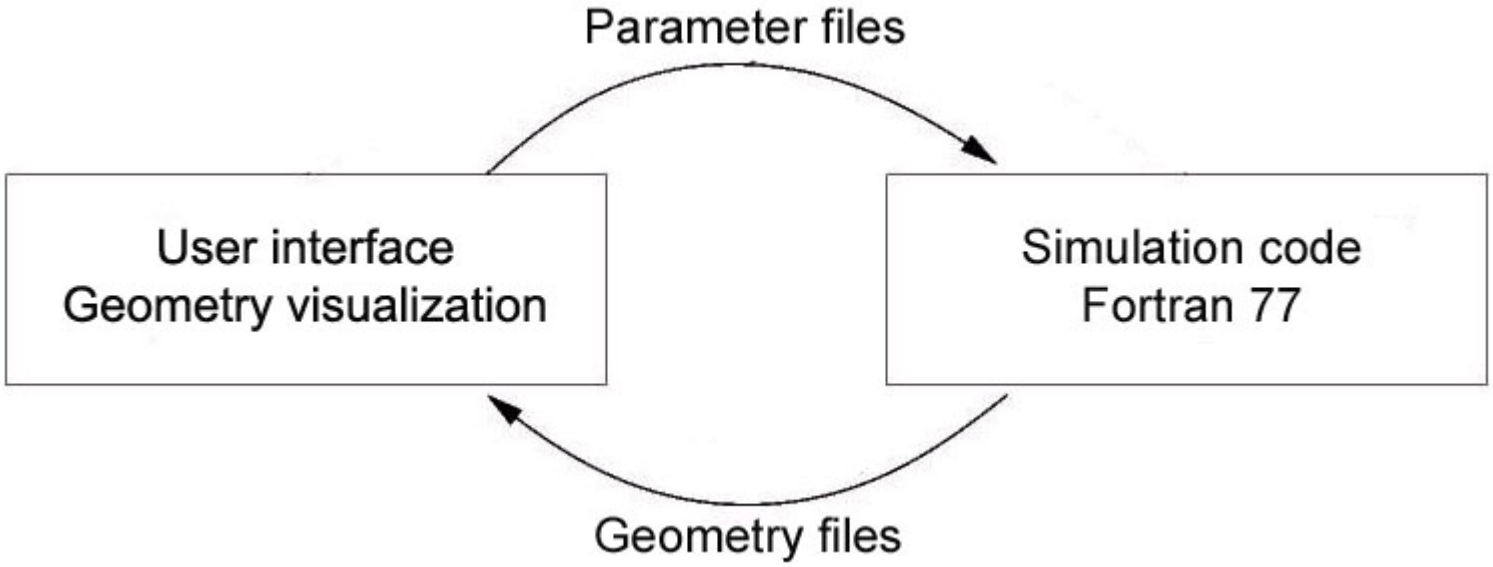
- Extension to 3D
- New mask objects
- Save/load mask using ORCAN
  - Define new components
    - E.g. Mask, MaskReader, MaskWriter
  - Create implementations
    - E.g. MAMLMaskReader, MAMLMaskWriter,
    - GDSIIMaskReader, GDSIIMaskWriter

Integration of Sthamas3D into ORCAN

Crystal Growth, Fraunhofer Institute, Erlangen
Dr. Jochen Friedrich, Dr. Thomas Jung

Sthamas3D

- 3D Simulation of convective heat and mass transport
- Block-structured grids
- Standalone executable (Fortran code)



Requirements:

- Load Sthamas3D geometry
  - Has its own geometry format
- Control execution
  - User needs to run make and executable
- Edit parameters
  - In text files
- Structured grid generator (one block)
  - Has its own, but very specific

Sthamas3D in ORCAN

ORCAN features used:

- VolMesh, VolMeshReader → Geometry loader

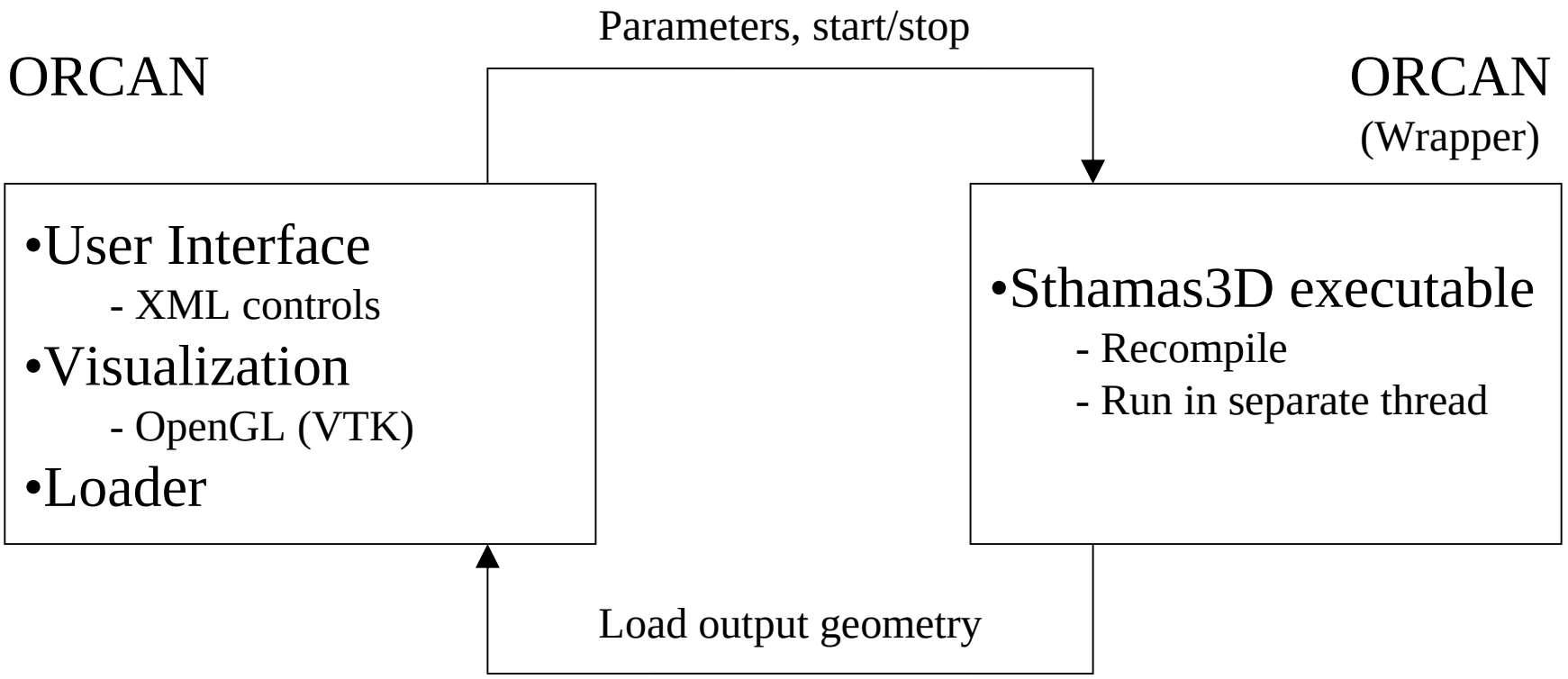
  VolMeshSthamas3DReader

- ExternalToVolMesh → C++ wrapper

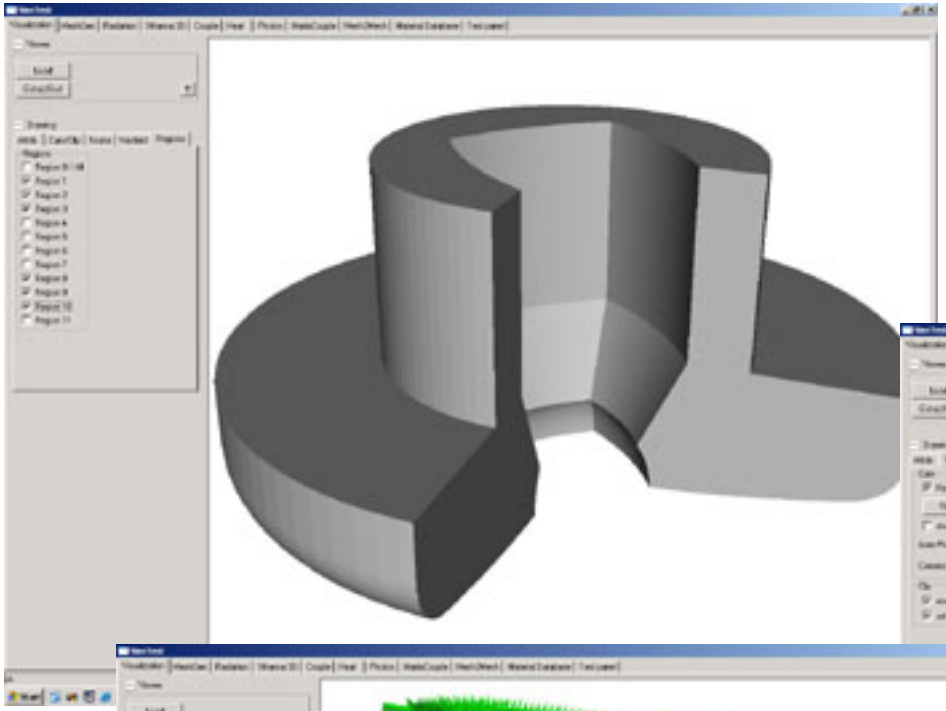
  ExternalToVolMeshSthamas3D,
  ControlExternal

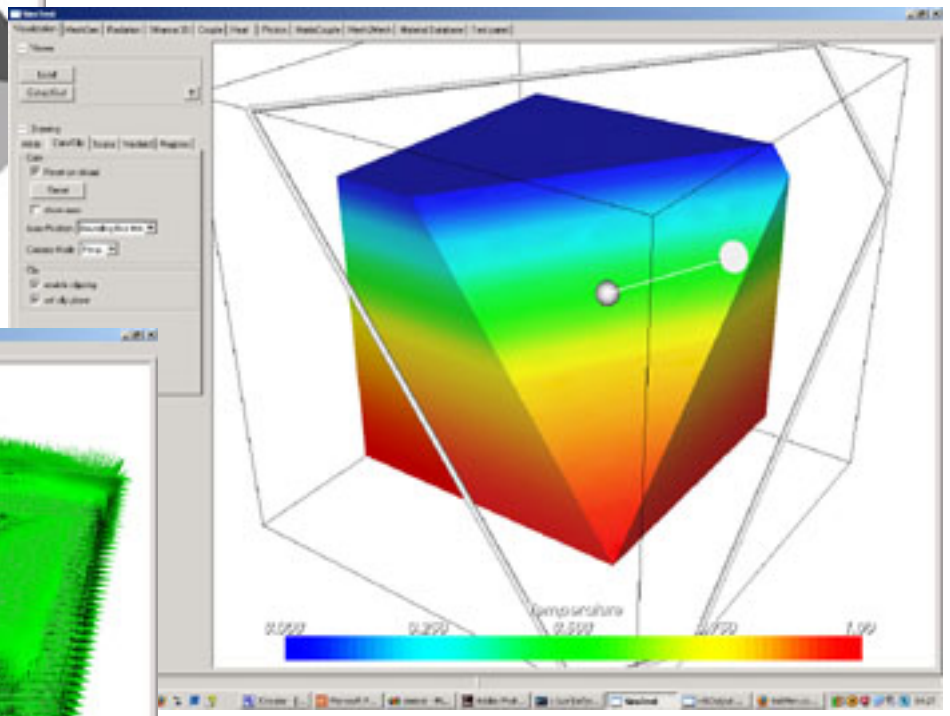
- XML-defined controls → User interface

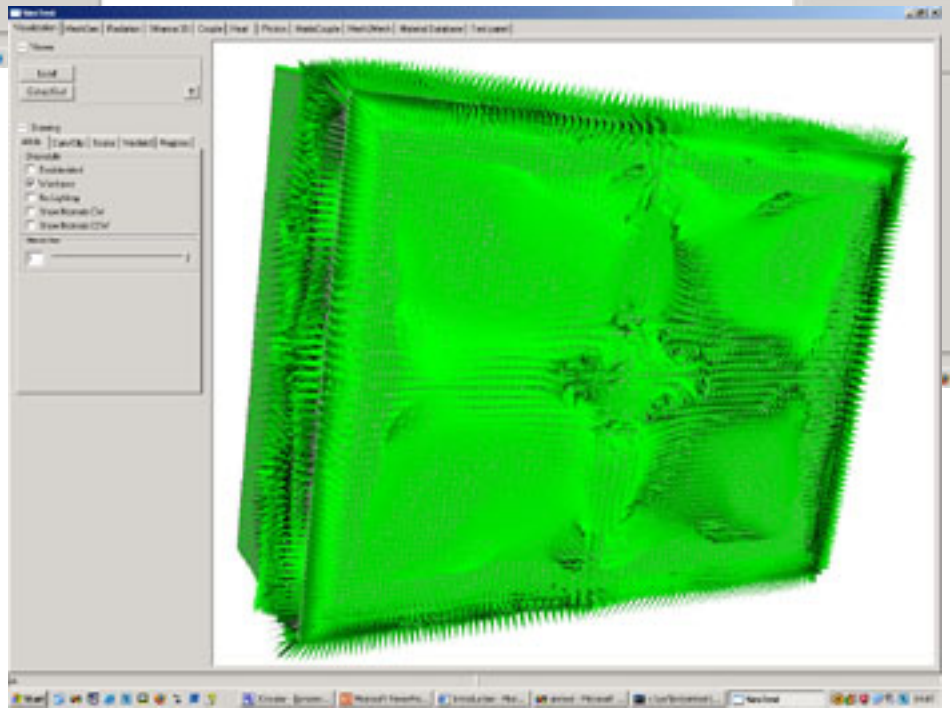
Integration structure



Screenshots of Sthamas3D geometry displayed in Simtest







Structured Grid Generator

Mathematically:

- Surface mesh, one block
  - Isomorph to a cube
- 6 faces
- Structured surface grids
  - for each face
- Volume mesh
  - Transfinite interpolation

Implementation:

Implementation of VolMeshGen component:
(volume mesh generator)

- Input
  - Surface mesh (SurfMesh reference)
- Output
  - Volume mesh (VolMesh reference)
- Execute
  - Generate structured grid

Usage sequence:

- Read a geometry (GeometryReader)
- Generate surface mesh (SurfMeshGen)
- Set input (SurfMesh)
- Set output (VolMesh)
- Generate (Execute)

Future:

- Algorithm for finding blocks in a volume