INRIA Sophia Antipolis, Asclepios Research Project

« Design of Mesh Topologies In SOFA »

Hervé DELINGETTE

Brina Goyette

SOFA

Geometry vs Topology

- A mesh is composed of :
  - A set of DOFs (Degrees of Freedom) , e.g. positions of each node
  - A description of how those DOFS are connected, e.g. edges, triangles, tetrahedra

Geometry Description

Topology Description

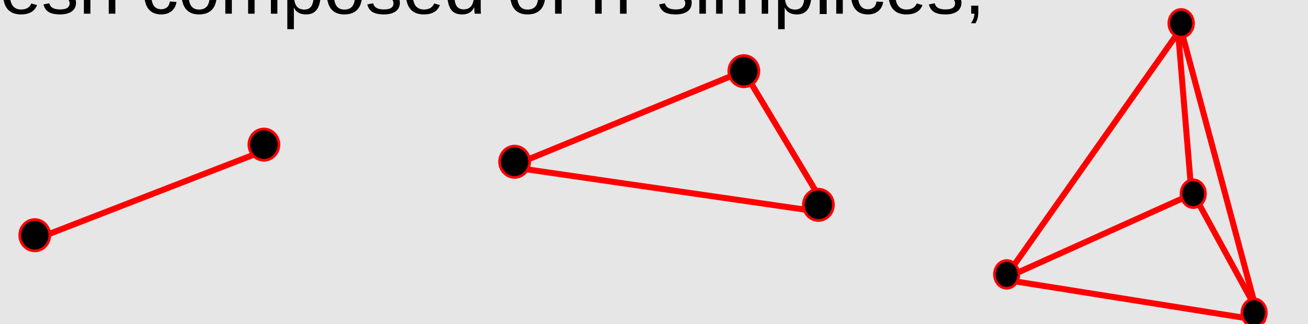
- Topology description is independent of geometry

SOFA

Example of Topologies

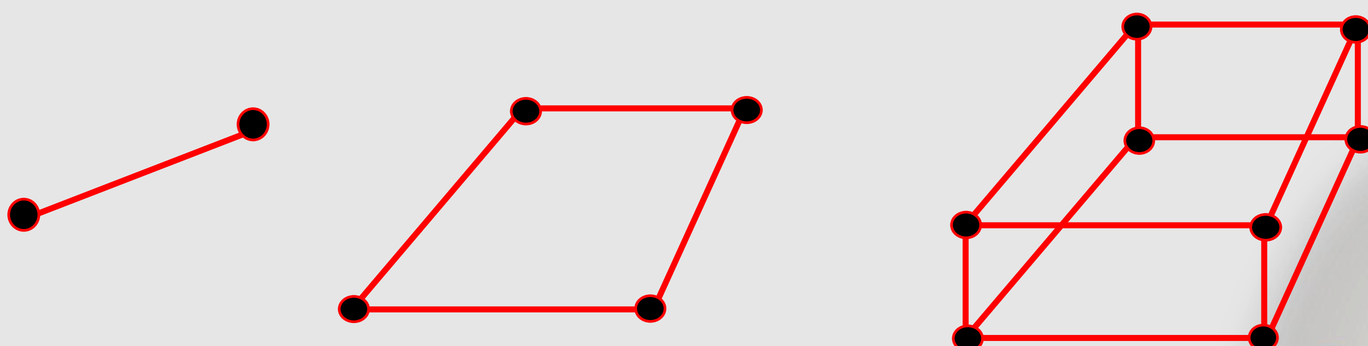
- 2 common mesh topologies :

1. Mesh composed of n-simplices,



Edge　Triangle　Tetrahedron

2. Meshes composed of n-cube

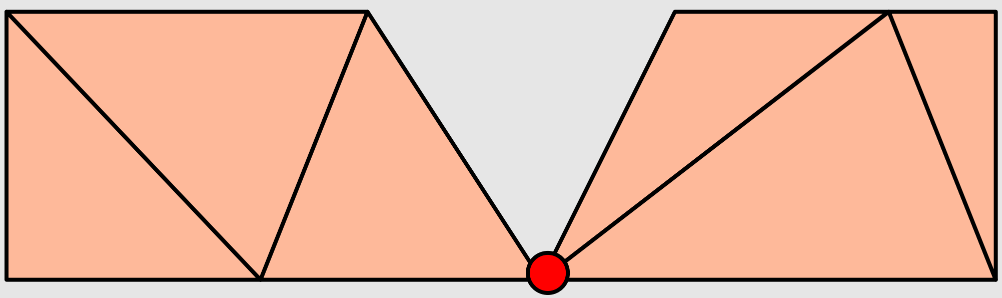


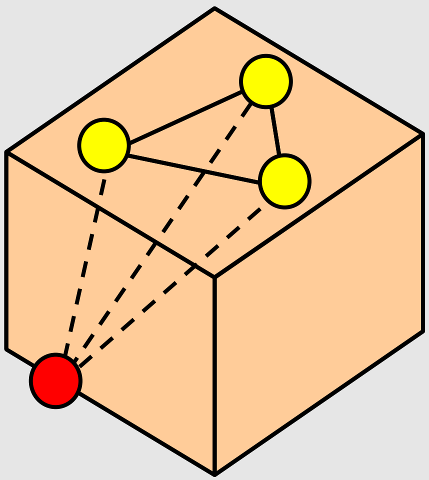
Edge　Quad　Hexahedro

SOFA

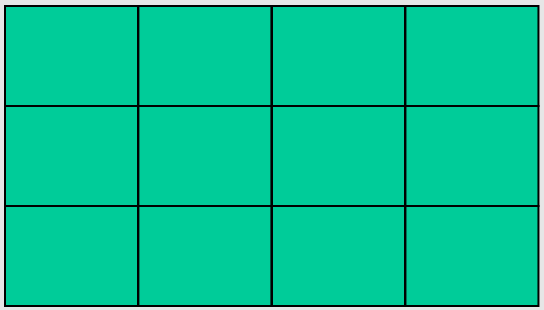
Topology description

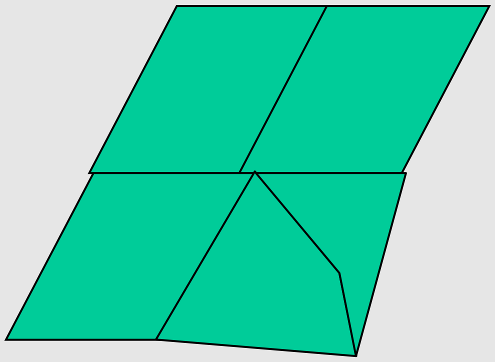
- Far more complex classification :
  - Conformal vs Manifold meshes





  - Structured vs Unstructured grid





SOFA

Mesh Geometry

Where are mesh vertices located in space ?

(position of each vertex)

Mesh Topology

How are mesh vertices connected to each other ?

(edges, triangles, tetrahedron, …)

COMPUTATIONAL MESH

- **Mesh Visualization**
- **Collision Detection**
- **Mechanical Modeling (deformation)**
- **Haptic Rendering**
- **Description of Scalar Fields (temperature, electric potential, …) or Vectorial Fields (speed, fiber orientation, …)**

SOFA

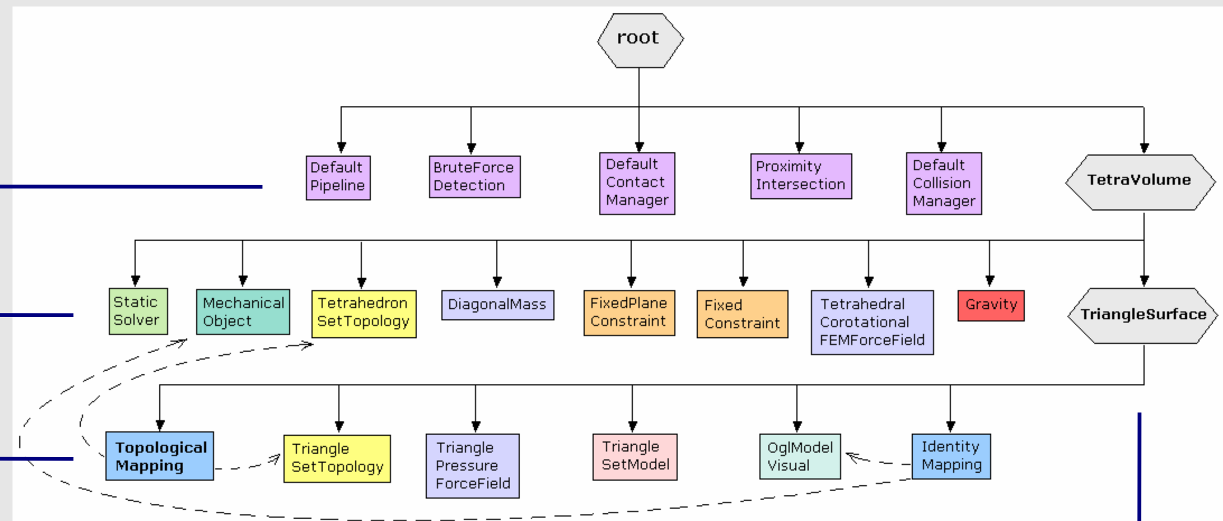
Open source C++ platform SOFA :

“ Real-time Modeling of Deformable Structures for Medical Simulation and Planning ”

Simulation Tree gathering software Components acting on meshes

Multiple software Components associated to one or several objects

- ***Solvers***
- ***Degrees of Freedom*** **(vertex positions)**
- ***Topological Description***
- ***Mechanical Force Fields***
- ***Mappings***
- ***Mass***
- ***Constraints***



Hierarchical levels of modeling the scene objects

Behavior Model*, *Collision Model*, *Visual Model

Information flow is carried on by Visitors

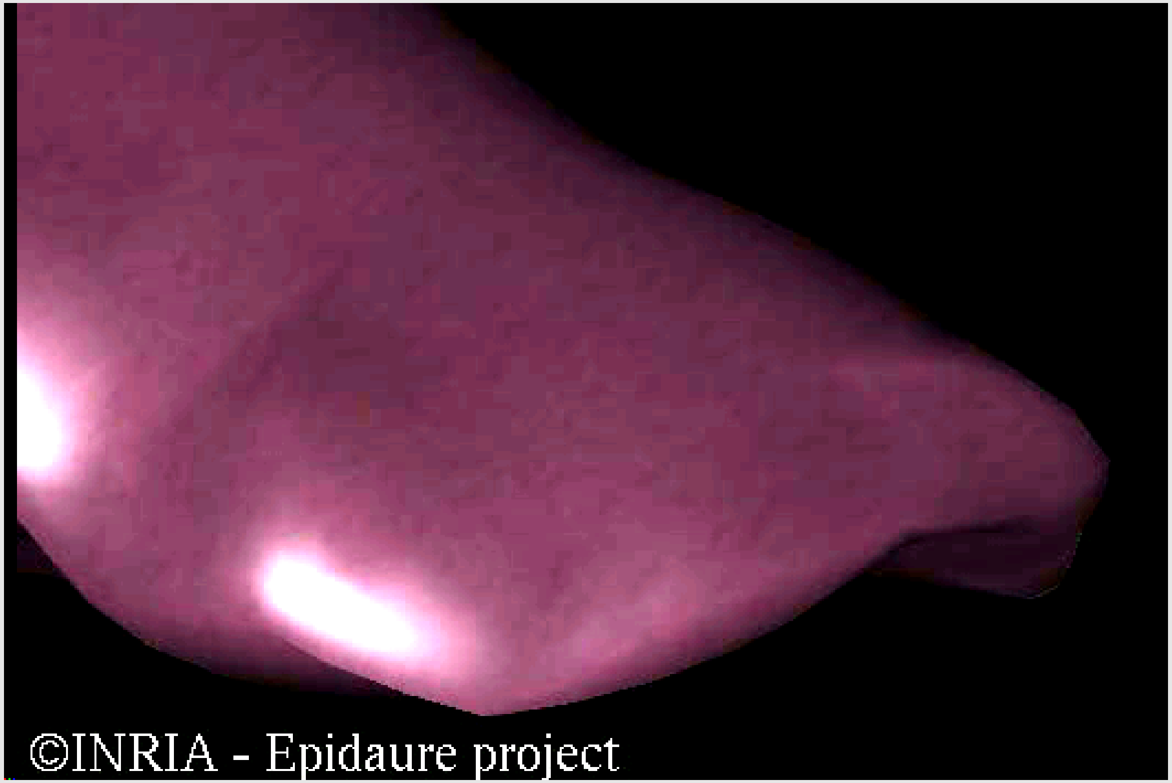
Visitors **traverse the *Simulation Tree* to propagate spatial positions ⇩ top down and forces ⇧ bottom up.**

SOFA

Topological Changes

Changing topology is key for medical simulation

Modifying topology entails huge impact for all aspects of the simulation (visual, mechanical, collision detection,..)



SOFA

Problem Position

“ How to ensure visual, mechanical, haptic and collision behavior

stay valid and consistent upon any topological change ? ”

SOFA

Static and Dynamic Topologies

Static Topology

- No changes throughout the simulation
- MeshTopology Component

Dynamic Topology

- Can change connections
- Requires changes to propogate throughout scene graph

BaseMeshTopology Component – common interface to both

SOFA

Implementation choices

- **Efficient storage of information into simple arrays**

  elements renumbering if topological changes

  time to update data structures only depends on the number of modified elements

- **Mesh related data not centralized but stored in the software *Components* and spread out in the *Simulation Tree***

- **Update of data structures transparent to the user through the propagation of topological events**

SOFA

Container Data Structures

- Force Fields, Constrains, Mapping may require to store information for each topological item (point, edge,…)
- Defined 2 container classes that handle topological changes
  - PointData<MyType>, EdgeData<MyType> are arrays (same as std::vector) of item of type MyType
  - PointSubset, EdgeSubset are arrays of points or edges
  - There are used-defined functions that are called when an item is created or destroyed

SOFA

Container Data Structures

- Those container data structures are “aware” of topological changes.
- User only provide callback functions to handle :
  - Destruction of a topological item
  - Creation of a topological item

SOFA

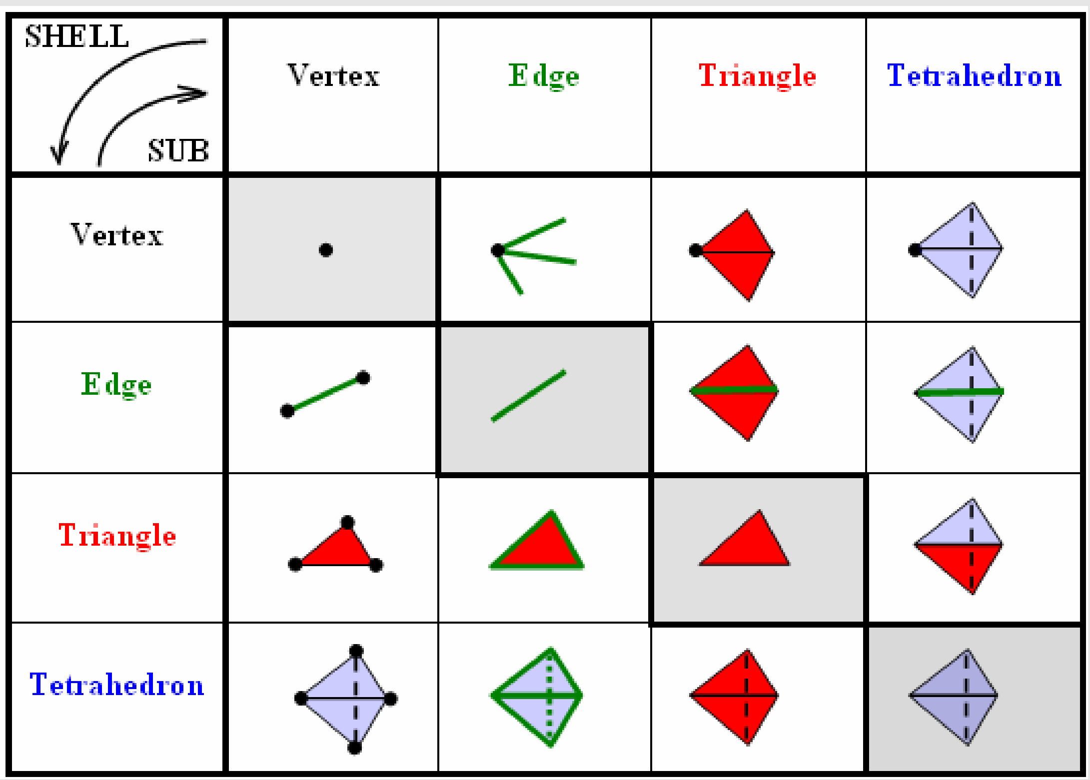
Hierarchical decomposition of meshes into k-cells

- **Edges = 1-cells**
- **Triangles, Quads = 2-cells**
- **Tetrahedron, Hexahedron = 3-cells**

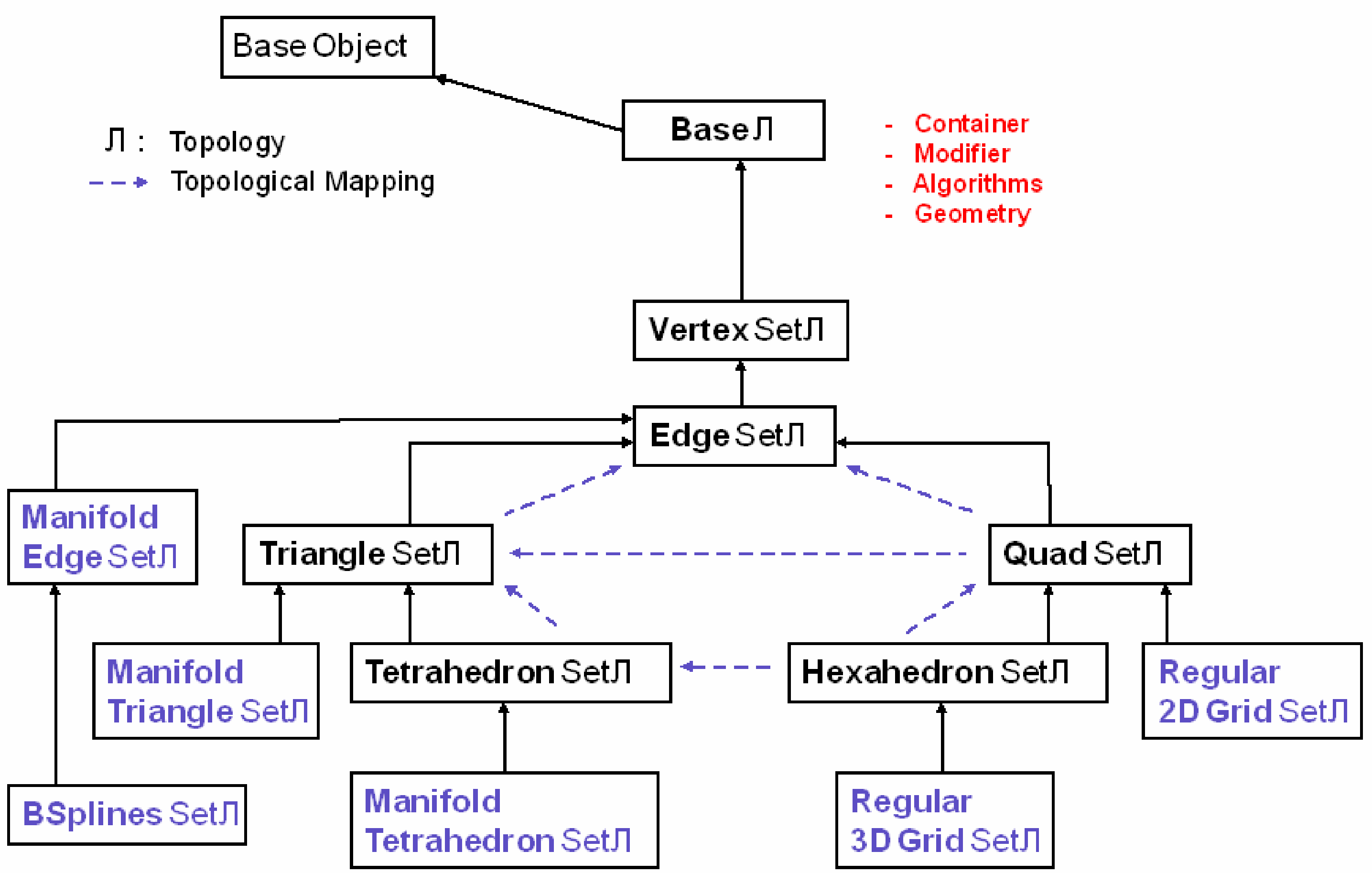
p < k

SHELL : k-cells adjacent to one p-cell

SUB : p-cells included in one k-cell



Mesh topologies structured as a Family Tree



SOFA

Topology objects composed of 3 functional members

TopologyAlgorithms

removeTriangles(l)
InciseAlongPointsList(l)
InciseAlongEdge(i)

Geometry

computeTriangleArea(i)
computeTriangleNormal(i)
computeSegmentTriangleIntersection(i)
computeIntersectedPointsList(…)

TopologyContainer

getTriangleArray()
getTriangleEdgeArray()
getTriangleVertexShell(i)
getTriangleEdgeShell(i)
getEdgeArray()
getEdgeVertexShell(i)
load()
addTrianglesWarning(l)
addTrianglesProcess(l)
removeTrianglesWarning(l)
removeTrianglesProcess(l)

SOFA

Two clinical applications

References

1. Cgal : Computational geometry algorithms library, http://www.cgal.org
2. Gipsi : General physical simulation interface, http://gipsi.case.edu
3. Opentissue, http://www.opentissue.org
4. Sofa : Simulation open framework architecture, http://www.sofa-framework.org
5. Springs, http://spring.stanford.edu
6. Agus, M., Gobbetti, E., Pintore, G., Zanetti, G., Zorcolo, A.: Real-time cataract surgery simulation for training. In: Eurographics Italian Chapter Conference, Catania, Italy, Eurographics Association (2006)
7. Allard, J., Cotin, S., Faure, F., Bensoussan, P.-J., Poyer, F., Duriez, C., Delingette, H., Grisoni, L.: Sofa – an open source framework for medical simulation. In: Medicine Meets Virtual Reality (MMVR 1915), Long Beach, USA (February 2007)
8. Forest, C., Delingette, H., Ayache, N.: Removing tetrahedra from manifold tetrahedralisation: application to real-time surgical simulation. Medical Image Analysis 9(2), 113–122 (2005)
9. Molino, N., Bao, Z., Fedkiw, R.: A virtual node algorithm for changing mesh topology during simulation. ACM Transactions on Graphics 23(3), 385–392 (2004)
10. Nesme, M., Payan, Y., Faure, F.: Efficient, physically plausible finite elements. In: Dingliana, J., Ganovelli, F. (eds.) Eurographics (short papers) (August 2005)