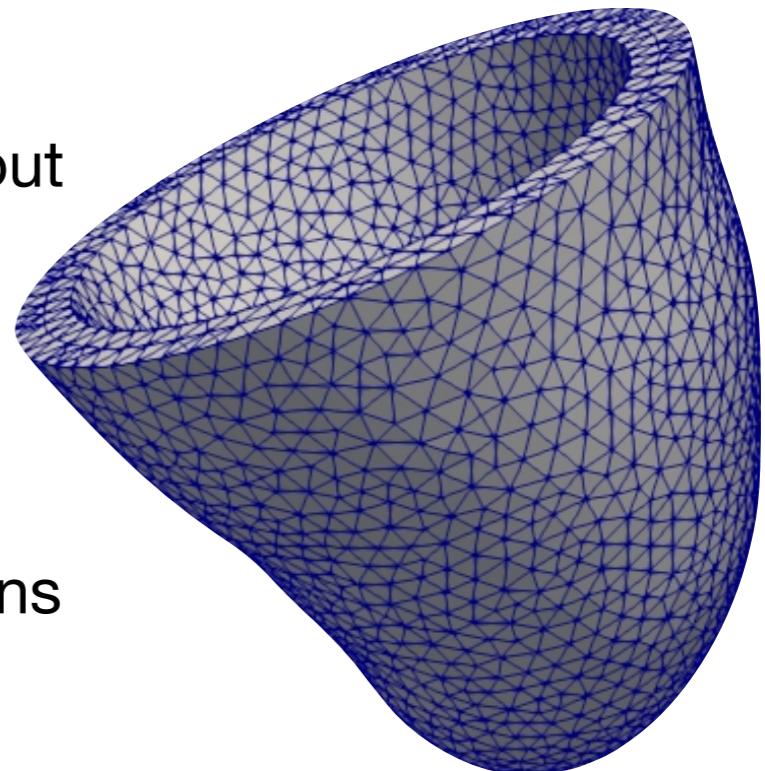


Remesh

Input

.vtk, .vtp (or even .xmf and .h5) with no information about boundary flags



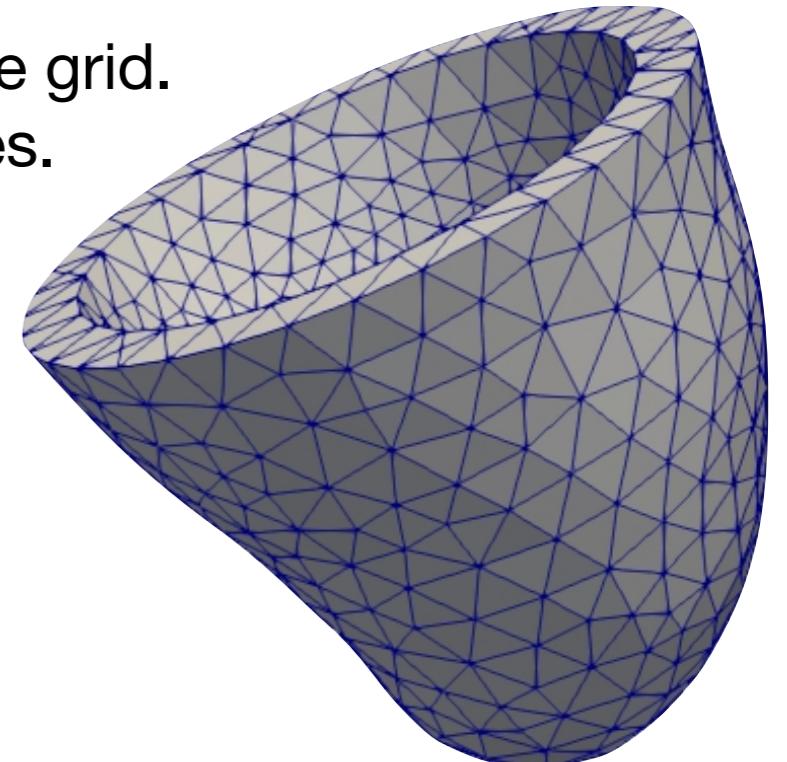
Output

Volume mesh file (.mesh for LifeV) suitable for simulations

When?

e.g. You have geometries but you want to coarsen/refine the grid.

e.g. You received have simulations results but no geometries.



Tools

GMSH

Remesh

1. Create surface mesh

Command line:

```
gmsh -2 -optimize initialSolid.vtk -o initialSolid_surf.stl
```

Alternatively, open gmsh, open the .vtk and “Save as”

2. Reclassify

Load .stl in GMSH → Mesh → Reclassify2D → Adjust parameters → Save (.msh file)

3. Remesh

In a .geo script (see remeshSolid.geo):

- Merge .msh file (the one created after the reclassify)
- create topology
- create compound elements (the lines and surfaces you want to be remeshed)
- create volume
- create Physical Surfaces (the ones that you want to be saved in the final mesh)

Run the script from command line:

```
gmsh -3 -optimize remeshSolid.geo -o remeshSolid.msh
```

Alternatively, open gmsh, open the .geo script → Mesh → 3D → Optimize 3D

4. Conversion from .msh to .mesh

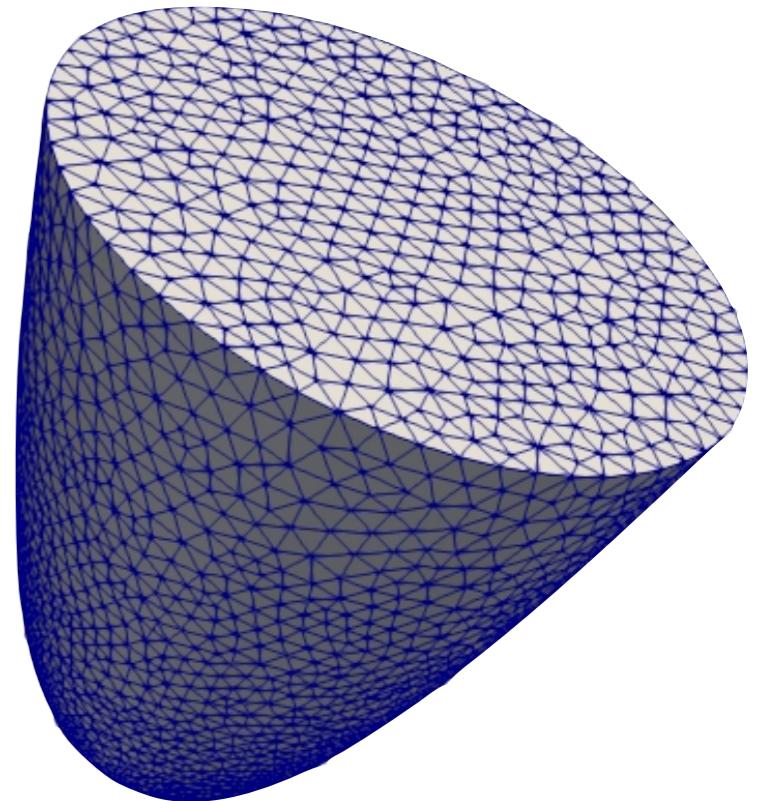
Use gmsh2medit.py from command line:

```
./gmsh2medit.py remeshSolid.msh remeshSolid.mesh
```

Add Boundary Flag

Input

Any surface or volume mesh where you want to add a boundary flag on a subregion (of your choice)

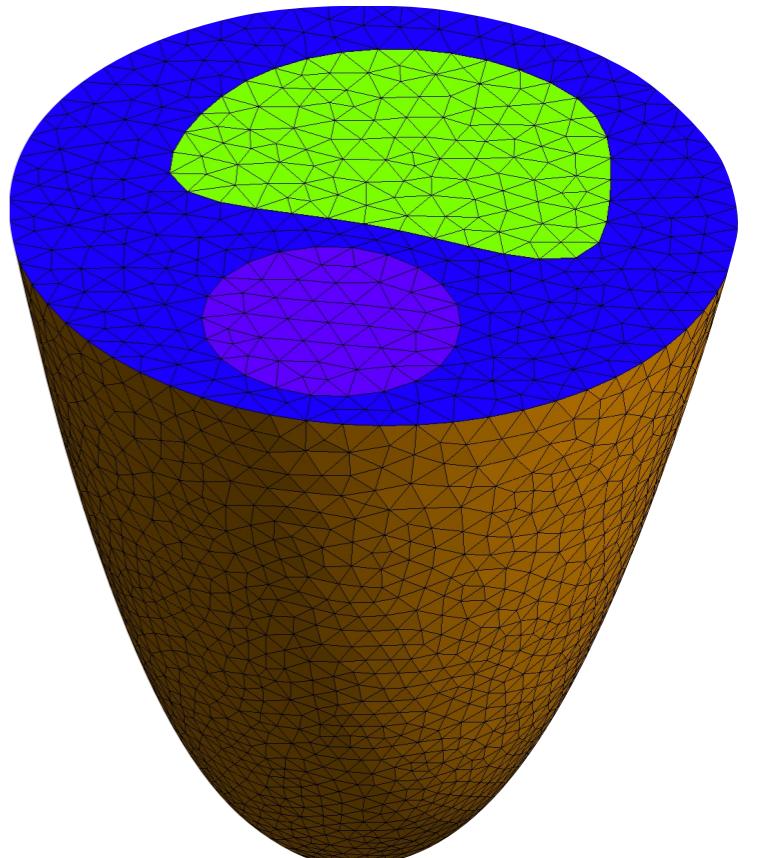


Output

Volume mesh file (.mesh for LifeV) suitable for simulations

Purpose

you want to create **new subregions** and you need to
remesh



Tools

Paraview

GMSH

Add Boundary Flag

1. Create the new boundaries

In this example we use Paraview to create the new boundaries but you can use other tools of your preference!

The new boundaries should be **coherent** with the initial **geometry** (they should actually lie on the boundary of your domain) but it is not required that they follow the mesh. The mesh will be recomputed such that the new boundaries are respected.

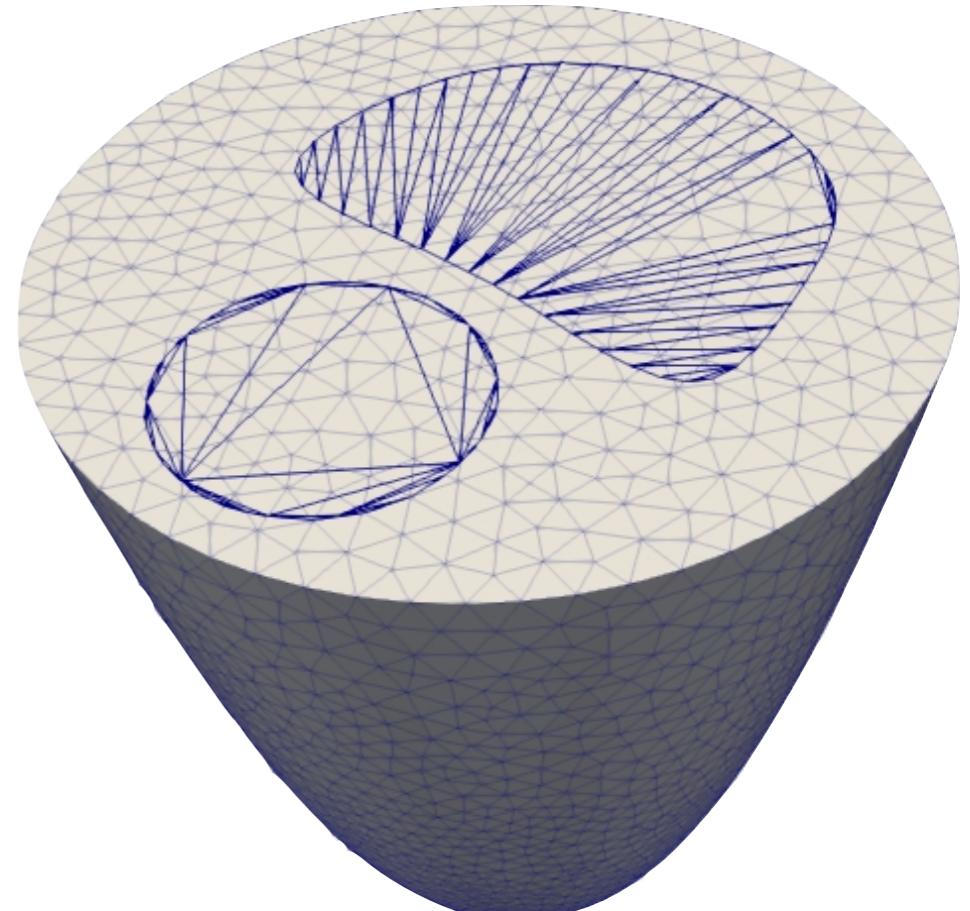
- Download `fluid.vtk` and `valves.py`
- Open `valves.py` with a text editor and change the path of `fluid.vtk` file
- Then Open Paraview
- In Paraview: Tools → Python shell → run script → `valves.py`

2. Save surfaces in .stl

You should save all the surfaces related to the new boundaries and also the surface of the initial volume mesh.

3. Reclassify fluid.stl

Open `fluid.stl` in gmsh → reclassify → save in `.msh`



Add Boundary Flag

4. Generate new surface mesh

In a .geo script :

(See file fluidAndValves.geo)

- Set the parameters for remeshing
- Merge the surfaces (then check the flags that gmsh has assigned to each surface)
- Create new surfaces for the two valves and for the remaining part of the top surface
- Create Physical Surfaces

Run `gmsh -2 -optimize fluidAndValves.geo -o fluidAndValves.msh`

5. Generate new volume mesh

In a .geo script :

(See file fluidAndValvesVolume.geo)

- Set the parameters
- Merge the surface (fluidAndValves.msh)
- Create Surface Loop and Volume
- Create Physical Volume

Run `gmsh -3 -optimize fluidAndValvesVolume.geo -o fluidAndValvesVolume.msh`

6. Run ./gmsh2medit.py

This step is required if you need to run simulations using LifeV

Add Internal Surface

Input

Anyvolume mesh where you want to add an internal surface
(of your choice)

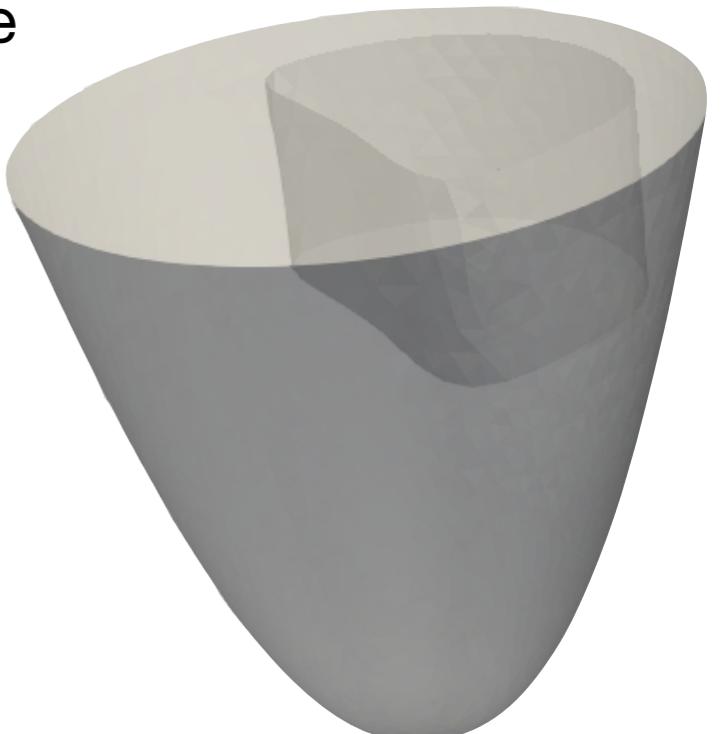


Output

Volume mesh that "fits" your inter fan surface

Purpose

You need to specify a boundary condition on the internal surface



Tools

Paraview (only to create the surface)

GMSH

Add Boundary Flag

1. Create the new boundaries

In this example we use Paraview to create the new boundaries but you can use other tools of your preference!

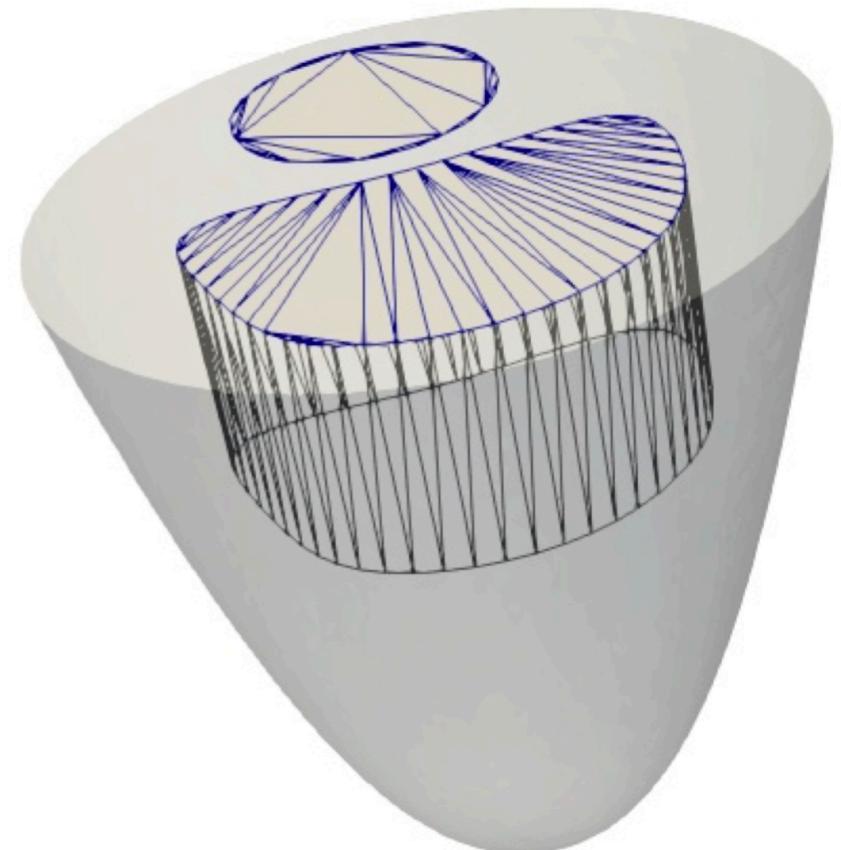
- Download `valvesExtrusion.py`
- Open `valvesExtrusion.py` with a text editor and change the path of `fluid.vtk` file
- In Paraview: Tools → Python shell → run script → `valvesExtrusion.py`

2. Save surfaces in .stl

You should save all the surfaces related to the new boundaries and also the surface of the initial volume mesh.

3. Reclassify fluid.stl

Open `fluid.stl` in gmsh → reclassify → save in `.msh`



Add Boundary Flag

4. Generate new volume mesh

In a .geo script :

(See file `fluidAndValves.geo`)

- Set the parameters for remeshing
- Merge the surfaces (then check the flags that gmsh has assigned to each surface)
- Create new surfaces for the two valves and for the remaining part of the top surface
- Create Physical Surfaces
- Create Surface Loop and Volume
- Create Physical Volume
- Add the Surface in Volume

Run `gmsh -3 -optimize fluidAndValvesVolume.geo -o fluidAndValvesVolume.msh`

5. Run `./gmsh2medit.py`

This step is required if you need to run simulations using LifeV

