

Documentation - draft



# Contents



## Part I

# Library overview



## Chapter 1

# Compilation: CMake





## Chapter 2

# Preprocessing: Salome (or Gambit)



## Chapter 3

### Development: Femus



## Chapter 4

# Postprocessing: Paraview



# Chapter 5

## Libraries involved

### 5.1 Externally installed

#### 5.1.1 MPI

There are direct MPI calls

#### 5.1.2 Petsc

##### 5.1.2.1 Metis/Parmetis

#### 5.1.3 Slepc

#### 5.1.4 HDF5

#### 5.1.5 Boost

For certain little tasks

### 5.2 Internally shipped

#### 5.2.1 b64

For encoding in output VTK files

#### 5.2.2 Adept

For automatic differentiation

#### 5.2.3 json

For input info





## Part II

# Compiling environment: CMake



## Chapter 6

# Advantages

For instance, you can use the targets automatically generated by CMake in the makefiles to compile the library in different modes

You can find for example ways to only do the PREPROCESSING of some target, so that you can look at some issue in the preprocessed code.

I needed this once to find about a global variable that was declared in SOME INCLUDE FILE that I had no idea about. Thanks to looking at the preprocessed code, I was able to identify the location of the declaration of this global variable.

This declaration was making my code compile even though I did not expect that it would compile.



## Chapter 7

**cmake vs. ccmake vs.  
cmake-gui**



## Part III

# Application: Creation





## 7.1 How to SETUP a NEW APPLICATION (will do a script probably)

- Create a folder in the ./applications directory with your main function and your src/header files
  - Write the Cmakelists.txt file for it
  - Add the reference to the application in the Cmakelists.txt of the package library
- open cmake-gui and run configure and generate
- now go to your application folder and run make to generate your executable



## Part IV

# Mesh Generation: Salome



# Chapter 8

## Features

- SUPPORT for DIFFERENT SHAPE ELEMENTS (quad, hex, tri, tet, ...)
  - SUPPORT for HYBRID MESHES (different elements on the same mesh)
  - SUPPORT for BIQUADRATIC ELEMENTS
  - SUPPORT for SCRIPTING
  - Possibility to define Data or Initial/Boundary conditions on your Mesh
  - An ACTIVE COMMUNITY of USERS
  - Potentially, Salome is also available to be *compiled* on your computer
  - Salome also contain its own version of Paraview

### 8.1 Modules

#### 8.1.1 SHAPER

This is a new module that in the long run is supposed to replace the role of the GEOM module.

At the beginning you start with a "Partset" that contains 7 elements (Origin, axes, ...) that cannot be deleted

##### 8.1.1.1 perform drawing operations in the Shaper module

To draw stuff in Shaper, first create a Part. Part 1 is created by default.

##### 8.1.1.2 Start - Points

To start setting points from scratch, do them with

Construction - Point

- Construction allows to put precise information

Don't do them with Vertex

- If you built a Point as a Construction in the Partset and you want to REMOVE it, you have to click on "Partset" and say "Activate". Then you can

right-click and finally "Delete" is active. Then, Deactivate again to continue drawing

#### **8.1.1.3 Start - Sketch**

- Sketch is for drawing things "imprecisely", as if it was a real sketch, without specifying precise coordinates and so on.

Then you apply constraints like coincidence of points, parallelism, horizontality etc. and you will have your precise shape eventually.

When you do a Sketch, remove the Interaction Style Switch (white mouse icon)

- Pay attention sometimes when you Dump study, it seems like you may lose the Shaper part!!!

#### **8.1.1.4 Export to Geom**

- do "Export to Geom" if you want to then do the meshing.

**8.1.1.4.1 Results** Notice: things that are exported to Geom are those that appear under the "Results" of a Part. If you have no object in the Results, nothing will be exported to Geometry.

### **8.1.2 GEOMETRY**

#### **8.1.3 MESH**

#### **8.1.4 PARAVIS**

It is Paraview with additional features, especially for MED files

## Chapter 9

# Installation

### 9.1 Issues

The first time you run Salome after installation, you may have it asking for opening service...

The "Failed to narrow the root naming context" is an error displayed when there is no entry in `/etc/hosts` corresponding to your hostname. So, you need to add the output of the command `"hostname"` to that file!





## Chapter 10

# How to save/open your work

### 10.1 Study HDF file

### 10.2 Python script

#### 10.2.1 When you REMOVE things, such as Mesh Elements, or Groups, the script may complain at loading. Then it is not hard to fix it

You go in the script and you see where it says “some objects were removed”. Remove them manually.



# Chapter 11

## How to run

You can run salome in GUI mode or also in terminal mode

### 11.1 GUI mode

```
salome start study.hdf  
salome start study.py
```

### 11.2 TUI mode

```
salome start -t study.hdf  
salome start -t study.py
```

### 11.3 Run salome from terminal if one salome instance is already running

```
Do first  
salome start it &  
Then do  
salome shell file.py
```



## Chapter 12

# Procedure: Geometry and Mesh

### 12.1 General ideas

#### 12.1.1 Parametrize your operations

Use the NoteBook tab to define variables. You can then use them both in the Geometry and in the Mesh modules.

##### 12.1.1.1 Parametrize Geometry with Shaper Parameters

This is the new way to go

##### 12.1.1.2 Parametrize Mesh with NoteBook

##### 12.1.1.3 Beware of not using “Update study” because it often crashes... or at least save the Study and the Script before!

### 12.1.2 Synergy between Geometry and Mesh

#### 12.1.2.1 Do your Geometry in a way that you have Geometrical Objects for the upcoming Groups in the Mesh

In that way, you will be able to construct “Groups on Geometry”, which is flexible when you change your parameters in the Notebook

##### 12.1.2.1.1 You need the Geometrical Objects for the Boundary conditions

##### 12.1.2.1.2 You may need the Geometrical Objects for Volume regions as well

### **12.1.3 Try to do Geometry operations as opposed to Mesh operations**

#### **12.1.3.1 Compound (in the Geometry) instead of CompoundMesh**

Then you will do Submeshes

You will have DOUBLE ENTITIES in the interfaces

You then have to remove them manually

- Merge nodes
- Remove inner edges

#### **12.1.3.2 Where to do certain operations**

**12.1.3.2.1 ROTATIONS in Geometry or Mesh?** If you have to rotate, do not do that in the Geometry, because then the Mesh will not follow.

If you do it in the Mesh, the Geometry does not follow either.

#### **12.1.3.3 SPLIT EDGES/FACES - Partition, Explode**

The tool for this is: Partition then Explode. At first it seems like with Partition no split happens, but then if you do Explode you will get all the pieces It seems like Cut is not the right tool for splitting

Pay attention to the difference between Common and Intersection

**12.1.3.3.1 When you mesh a Partition object, the Mesh module automatically generates Groups of Faces... Later you may use them, they are automatically Groups on Geometry!**

## Chapter 13

# Procedure: Non-hybrid meshes

### 13.1 1D

#### 13.1.1 Edges

### 13.2 2D

#### 13.2.1 Triangles

#### 13.2.2 Quadrangles

##### 13.2.2.1 Geometry

Create faces (Draw points and lines if needed, or create Faces directly)

- Extract edges from a face (Explode)

##### 13.2.2.2 Mesh

Create Mesh object (only initialize)

Create SubMesh of Mesh over the wanted SubGeometry

- Hypothesis: Propagation of 1D Hypothesis

In Mesh:

- 2D: Quadrangle Mapping

1D: Wire Discretization, or another SubMesh with Propagation in the other direction (this would overwrite Wire Discretization)

##### 13.2.2.2.1 Create Mesh by pieces: CompoundMesh (deprecated) –

Convert EACH mesh to the order you need (linear/quadratic/biquadratic)

You have to do this operation BEFORE creating a CompoundMesh, because the compound mesh does not have an underlying geometry to use

You have to do ONE BY ONE, you cannot do by selecting all meshes at once!!!

- Build CompoundMesh to merge the meshes
- ! Check the Boundary of the CompoundMesh!!!
- ! When you do a Compound, the double nodes are MERGED, but the inner edges or inner faces are not REMOVED! So, you have to remove them manually!!!

- ! You need to remove the boundary elements of the previous separate pieces that are now inside the new domain!

To do so you do "Remove elements" and you set a filter "Free borders - Not": inner boundary elements are non-free borders

Unfortunately you don't see this operation as a new object in the Mesh menu

- ! Rename it to "Mesh" + something!!!

- AAA: I found how to do the "quadrangle-mapping" mesh of a face where one side has multiple edges coming from Explode!!! I just do the same, but I specify "Composite wire discretization" instead of "Wire discretization"!!!

### 13.2.2.3 Divided Disk

Divided Disk or Divided Cylinder do exactly what you are looking for!!!

Partition a semisphere using a Divided Disk!

Beware: it is essential that a vertex of the inner square is aligned with the "meridian arc" that is intrinsic in the generation of a Sphere surface!

### 13.2.2.4 From scratch

For a circle, just do a square in the middle, then divide the circumference into 4 parts, join the angles of the squares with the vertices of these parts and you are done.

For the semisphere, if you try the same it doesn't work, because the faces that are constructed do not belong to the original semisphere...

So the idea seems to be NOT the following: - make curved edges on the semisphere - use them to create curved faces So, try to avoid starting from the Edges and instead keep yourself at the level of Faces. Then only at the end you will Explode to get the Edges (that you may need later for submeshes)

Also, try not to cut too much, because if you do so you may have multiple edges on one side and then it is not so easy to mesh...

The other way to go is to directly create subfaces by Partition of the semisphere!

First you split, then you join...

Now:

- Fuse doesn't work with non-planar faces... - Glue Edges is to put together coincident edges, but not to remove them - It seems like the right option is Sewing, then Union faces, then Explode to get the edges for the submeshes - The only problem is that now one face does not have 4 edges, but one side can



be made of multiple edges, and that creates problems in the mapping algorithm. So, it's all about making 1 edge

## 13.3 3D

### 13.3.1 When you do a 3d mesh, for Element Types not only will you have Volumes and Faces, but also the Edges of the “Bounding Box”

You can leave the edges where they are, you don't need to remove them, because in 3d only Volumes and Faces will be read from the code

### 13.3.2 Hexahedra

#### 13.3.2.1 Parallelepiped with 3 arbitrary mesh discretizations in the 3 directions

I cannot create a Submesh of a Submesh

- Geometry

- Create Volumes

- j Extract Faces from a Volume

- j Extract Edges from 2 faces in order to do Submesh of 3 edges in the x,y,z directions

**13.3.2.1.1 Hexahedron (i,j,k)** Global Algorithms: 3d: Hexahedron (i,j,k)  
2d: Quadrangle mapping 1d: Wire discretization - Here sometimes it gives trouble if you don't specify any "Number of Segments"...

Do 3 submeshes on 3 Edges, and on each of them establish a "Propagate hypothesis on opposite edges"

#### 13.3.2.1.2 Extrusion 3d Box

- Explode the Faces (so that you can select the boundary conditions)

- Explode the Edges of the Faces

### 13.3.2.2 Geometry of Extrusion type

#### 13.3.2.2.1 BETTER: Do 2d Geometry, Extrude 2d Geometry, then 3d Mesh with Algorithm “Extrusion 3d”

**13.3.2.2.2 WORSE: Do 2d Geometry, then 2d Mesh, then Extrude the Mesh** - Do extrusion only of faces to avoid generation of extra edges

- "Scale factors" refer to the TRANSVERSAL direction, not the orthogonal one

**13.3.2.3 HOW TO SHARE SUBMESHES ON CONTIGUOUS FACES**

When you explode adjacent faces, the edges are called differently... we should identify them

**13.3.3 Tetrahedra****13.3.3.1 Algorithms****13.3.4 Wedges**

## Chapter 14

# Procedure: hybrid meshes

I am not sure that the available algorithms allow HYBRID meshes.

The only possibility that I see is

- to create Sub-Meshes of a Mesh;
- to have only one type of geometric element in each Sub-Mesh



## Chapter 15

# Procedure: Convert to biquadratic (for quad/hex) or to quadratic (for tri/tet)

This is because we need those nodes to construct quadratic/biquadratic dofs

Do we need to do it now or can we postpone it right before exporting the MED file?

I think it is the same

### 15.1 It can be done only on a Mesh or Sub-mesh



## Chapter 16

# Procedure: Mesh operations (avoid, whenever possible)

It is better to avoid them because they do not have underlying Geometrical Objects anymore

### 16.0.1 Compound Mesh

**16.0.1.1** If you do Compound, do the “Groups on Geometry” before doing Compound, i.e., on the partial meshes

In fact a Compound does not have an associated Geometry to it

### 16.0.2 Translate Mesh

Copy Mesh of course does not do the trick, you need “Translation”

“Offset” of a mesh does not work for Quadratic mesh

“Translation” of a mesh works with “Create a new mesh” and “Copy groups” creates a translated copy

### 16.0.3 Stretch elements

“Scale transform” does the trick

### 16.0.4 Move nodes

There is a command called “Move node”, perhaps it helps avoiding to remesh everything if I go back and change the Geometry instead...





## Chapter 17

# Procedure: Do Groups of Mesh elements

### 17.0.0.1 Can I define more than one Group on the same element?

Yes, you can. It will be a bit more complicated to handle the reading, though. In fact, when Groups of elements are intersection, the naming inside the MED file changes

If there are groups that intersect, then Salome - when exporting to MED - will define intersecting families.

If there are elements belonging to 2 groups, they will define a FAM with its own index and the name of the 2 groups

So, the elements belonging only to 1 group will have a FAM with only one group

So, FAM are generated based on WHAT INTERSECTIONS BETWEEN GROUPS there are.

All of this is made in such a way that every element will have *only one* FAM number in the FAM field of the elements.

### 17.0.0.2 Can I select faces INSIDE the domain, not at the boundary???

The MED format exports only the BOUNDARY ELEMENTS... How can I export INNER FACE ELEMENTS as well? Well, when I draw a mesh by doing Submeshes and making a Compound, there are still elements from the interfaces of the boundaries...

We could have some of the inner edges, coming from the result of drawing subpieces

Perhaps what we can do is to not remove the inner faces used to build the submeshes.

Then, I have to change the way I check that ALL boundaries have been implemented, because now I will have additional edges in the list of all edges

that are NOT boundary edges... Basically, to check if an element is on the boundary I would just have to count how many volume elements it belongs to (I am excluding a slit domain with this)

If an edge belongs only to 1 volume element, then it is a Boundary face.

- An alternative - by hand - could be to use “Add Quadratic Edge”... but it’s very tedious

- The solution of keeping the inner edges is constrained to the fact that I have to draw thinking of those inner edges “ahead of time”. For instance, assume you want to perform integration over a face that was not used to draw your mesh... Then you should draw your mesh using that face from the very beginning...

- The third solution, in case we cannot create groups of inner edges graphically, is to revert back to using if’s on the coordinates...

- Another way that is graphical consists in defining the Group of Volume Elements whose Boundary you want to identify, and then do “Create boundary elements” on that Group

### 17.0.1 Do “Group on Geometry” or “Group on Filter” instead of “Standalone group”, whenever possible

The flexible way to define these groups is NOT by selecting the IDs but by doing either “Group on geometry” or “Group on filter”

- You can use Filters to define Groups of Mesh Elements

## 17.1 Requirements for Groups: ALL

### 17.1.1 Name

must be named `Group_X_Y`, where X is the flag given to the object and Y is the type of boundary condition (0 = Dirichlet, 1 = Neumann).

What about the groups that are not for boundary conditions? In that case, first is “flag”, second is “type”, where “type” will have a different meaning

In order to enforce Boundary Conditions, Material Properties, and potentially other things too, we have to define Groups of Edges, or Faces, or Volumes (also Groups of Nodes for the extremes of a 1d domain)

### 17.1.2 Give one and only one group for each element

That’s because we have not implemented reading when multiple groups are defined on the same element

## **17.2 Requirements for Groups: BOUNDARY CONDITIONS**

### **17.2.1 Remove inner boundary elements**

## **17.3 Requirements for Groups: OTHER PROPERTIES**

You can specify geometrical regions



## Chapter 18

# Procedure: Run some Mesh controls

Before exporting the mesh file, it would be wise to check that there are

- no Free Nodes or Orphan Nodes (nodes that don't belong to any element)
- no Double Nodes
- no Inner Edges (only Free Borders, i.e., edges that are at the boundary of the domain)



## Chapter 19

# Procedure: Export MED file

Make sure you remove the check “Automatically detect space dimension”.

That’s because we always read all 3D coordinates. That can handle cases like 2d manifold in 3d space.





## Chapter 20

### Questions

- 20.1 What is the difference between SubMeshes and Groups? Are Submeshes exported in the MED file?
- 20.2 When I remove a Group, can I ask to remove also the elements of that group from the Mesh???
- 20.3 When you do Extrusion, it also generates the Groups of Volumes from Groups of Faces! It puts a suffix

extruded



- 20.4 When you remove inner edges in a compound mesh and then you dump the script and reload it, the mesh is corrupted! On the other hand, if instead of dumping the script I save the study it is NOT corrupted! So, I will stick with saving the study for now.
- 20.5 Other times, you dump the script and the Shaper is gone!! It is a current bug! Up to now, you need to activate Shaper before any dumping. Sorry.
- 20.6 Operations in the Mesh such as "Convert to/from quadratic", "Remove elements" and others are not shown as stages in the Mesh menu in Object Browser
- 20.7 Intersection vs. Common: the first does it with Edges, the second does it with Volumes, basically...
- 20.8 To remove Extra Edges on a Face, you first have to create an auxiliary volume, and then do Repair-; Remove extra edges from that volume, then that face will be fixed! Another trick is to put the Face in a Compound. There it works!
- 20.9 When you convert a mesh to quadratic/biquadratic, do it on a Mesh that is ATTACHED to a Geometry, because it is with the geometry that new nodes are added! Otherwise nothing happens!
- 20.10 What is the difference between Regular Faces and Free Faces in the Mesh algorithms? For instance, the Disk only shows algorithms for Free faces...
- 20.11 Is it possible to merge items obtained by

Geometry object you will then mesh. I can try this to double check.

## Part V

# Application: Implementation, main stages



- Follow the examples





# Chapter 21

## Main blocks

- Mesh generation
  - Mesh Partitioning for parallel (Metis)
  - Solution on a Mesh
  - Problem with Solution unknown (certain unknowns, certain not)
  - Solution: Boundary conditions
  - Solution: Initial conditions
  - Space discretization (and quadrature)
  - Time discretization (i.e. quadrature in time)
  - System of a Problem (Nonlinear loop, Linear Solver, Preconditioner, etc.)
  - Final output
  - MultilevelMesh
  - MultilevelSolution(MultilevelMesh)
  - MultilevelProblem(MultilevelSolution)
  - System
  - MultilevelProblem.add(System)
  - System.solve() –, with this process, the MultilevelSolution (which is accessed by System through MultilevelProblem) is updated



# Chapter 22

## Init

### 22.1 Parallel initialization

If you want to retrieve  
the number of processors  
or the current processor  
you can do so from any class that inherit `ParallelObject`

### 22.2 Input parser

We have some interface to parse:

- the command line
- an input file

### 22.3 File handling



## Chapter 23

### Init: Example



# Chapter 24

## Problem

### 24.1 MultilevelProblem

Everything in the Assemble function (and in other parts too) is accessed through this class

Il problema deve essere dotato dei dati CONDIVISIBILI tra tutte le EQUAZIONI:  
- parametri fisici / di visualizzazione - puntatori alle equazioni - puntatori ai punti di gauss per l'integrazione - puntatore al MESH - NUMERO DI LIVELLI del multigriglia

Io ci aggiungerei anche - MGpar, parametri del MG disponibili per tutte le equazioni.

### 24.2 Quadrature

You put it here so that many systems can share it

Clearly, if you have terms with different polynomial degrees, you pick for the quadrature the ones that guarantees exactness for the highest-degree polynomial

### 24.3 FE Abstract Families (with precomputed Quadrature evaluations)





## Chapter 25

### Problem: Example



# Chapter 26

## Mesh

If you choose LINEAR for the MESH DISCRETIZATION, it only works in SERIAL! It could actually work if the MESH was LINEAR...

For now, the Mesh must be BI/TRI-QUADRATIC. Could it be Tri7, or does it have to be Tri6??? We have to test.

### 26.1 MultilevelMesh

MLMesh contains a vector of Mesh, each of which contains the Sol of the topology

EraseCoarseLevels: it shifts the level vector, so if you start with 5 levels and you want to remove the first 3, it shifts 4 and 5 to the 1st and 2nd position

### 26.2 Mesh

#### 26.2.1 Elem

The Elem class contains the **list** of all elements

Every Mesh object has an Elem object

The Elem list is initialized in different manners at different levels:

- at the COARSE level, it is constructed in the Read function of MEDIO or GambitIO

- for all other levels, it is constructed in the RefineMesh function in MeshRefinement, with another constructor

```
//          7-----14-----6
//          /|              /|
//          / |             / |
//          15 |      25      13 |
//          / 19      22  /  18
//          /  |             /  |
```

```
// 4-----12-----5 |
// | 23 | 26 | 21 |
// | 3-----10-----2
// | / | /
// 16 / 20 17 /
// | 11 24 | 9
// | / | /
// | / | /
// 0-----8-----1
```

```
//      3
//     /\
//    /\
//   /\
//  9  |  8
// /\  |  \
// /\  |  \
// 2-----5-----1
//  \  |  /
//   \  |  /
//    \  |  /
//     \  |  /
//      0
```

```
//      5
//     /\
//    /\
//   /\
//  11 | 10
// /\  |  \
// /\  |  \
// 3-----9-----4
// | 17 | 16 |
// | 2 |
// | / \ |
// 12 / 15 \ 13
// | 8 7 |
```

```
//  | /      \ |
//  | /      \|
//  | /      \|
//  0-----6-----1
```

```
//      3-----6-----2
//      |           |
//      |           |
//      7       8       5
//      |           |
//      |           |
//      0-----4-----1
```

```
//      2
//      | \
//      | \
//      5   4
//      |   6 \
//      |       \
//      0-----3-----1
```

```
//
// 0-----2-----1
//
```

### 26.2.2 Boundary mesh

Whenever you want to retrieve the Boundary of a Mesh, you need to loop over each mesh element and then loop over its *faces*.

If the FaceElementIndex is  $< 0$ , then it is a *boundary* face.

Otherwise, it is equal to the volume element number corresponding to that face!

FaceElementIndex is initialized to  $-1$  at first.

When the FaceElementIndex is stored, the flag that is on the mesh file is made NEGATIVE and then we subtract 1

### 26.2.3 Interface mesh

If I wanted some **interface** faces, not on the boundary, I should set their FaceElementIndex to be  $> 0$ . How does this affect the rest of the code?

## 26.3 Geometric Elements

## 26.4 MeshInput

These mesh files also set FLAGS to faces

### 26.4.1 From function (MeshGeneration)

### 26.4.2 Gambit IO

### 26.4.3 MED IO

In the mesh file we can specify Groups of Volume Elements as well as Groups of Face Elements.

## 26.5 MeshPartitioning

### 26.5.0.1 Elem and Node splitting

In a FE code you have you must do a partition of Elements and Nodes among the processes.

Elements only belong to one process

Nodes belong to one process only, but then they are also Ghosted in the other processes that “own” them

### 26.5.1 MeshMetisPartitioning

The Metis library is responsible for that

The Elements are reordered, so that

the 1st range goes to proc 0, the 2nd range goes to proc 1, and so on...

### 26.5.2 MeshASMPartitioning

## 26.6 MeshRefinement

This object is used in the MultilevelMesh class, to construct the various levels

## Chapter 27

# Mesh: Example

27.1 From file or from function

27.2 Refine or not, erase coarse levels or not

27.3 Partitioning, Serial vs. parallel

27.4 Printing





## Chapter 28

# Solution (Quantities, Equations, Operators) on a Mesh

### 28.1 Approximation Theory VS Solution of a PDE System

If you want, you can just interpolate a given function

Otherwise, if you want to solve a PDE system, you first have to define all the Unknowns as Solution objects.

Then, you may also define OTHER Solutions, as auxiliary fields, and so on

So, the Solutions are the “WARM-UP” to ADD A SYSTEM to the PROBLEM

Solution is like an INTERMEDIATE STEP between MESH and SYSTEM

### 28.2 Solution

Every Solution object contains ALL variables

It should actually be called a MultiVariableSolution (multiple in variables, not in levels)

Every Solution object has multiple Numeric Vectors

Sol

SolOld

Eps

Res

## 28.3 MultilevelSolution

MLSol contains a vector of Sol

Only MultilevelSolution has a Writer object. This writer will write from the FINEST level available.

Solution does not have a Writer.

Similarly, MultilevelMesh has a Writer, but not Mesh.

However, currently the Writer in MultilevelMesh does not work.

- The values of variables are taken from the MultilevelSolution object..  
That's where the absolute values are stored, otherwise it is only about delta x in the nonlinear loops

## 28.4 MultilevelSolution: Boundary conditions

They are set through a function pointer.

They are associated to the Solution and not to the Equation.

In this way you can set BCs also for Solutions that are not Unknowns of an Equation

### 28.4.1 Dirichlet

- Dirichlet boundary conditions are set AFTER the ASSEMBLY process, by setting a Dirichlet row to zero and putting 1 in the diagonal, and by setting the corresponding Residual to ZERO at those rows

### 28.4.2 Neumann

They are set by implementing a boundary integral

## 28.5 MultilevelSolution: Initial conditions

28.5.0.1 Check that boundary values and initial values at the boundary are CONSISTENT

## 28.6 Output: Writer

## Chapter 29

### Solution: Example



## Chapter 30

# System(s) (of a Problem)

After the MGSolve or MLSolve or solve function is called, the MultilevelSolution vector is updated and ready to be printed to file...

But where exactly is it updated? Also, the print routines only print at the FINE level, so is MultilevelSolution updated only at the FINE level or at ALL levels???

Ok, what happens is that the MultilevelSolution object contains a VECTOR (based on number of levels) of Solution objects, and THIS is the vector that the Writer uses need to access if I want to retrieve my values!!!

System: the System class has a

std::vector of Mesh pointers and a

std::vector of Solution pointers,

and it also has pointers to MultilevelSolution and MultilevelMesh...

MultilevelMesh, in turn, has a std::vector of Mesh pointers in it;

MultilevelSolution, in turn, has a std::vector of Solution pointers in it

- Every Mesh should be SINGLE-LEVEL, but it has a \_ProjCoarseToFine object, so it is not really single-level...

### 30.1 Assemble function of each System

The purpose of the assemble function is to fill Jacobian and Residual.

Then, one solves for the Error of the Unknowns, and that error goes to update the Unknowns in the MLSolution container.

In fact, ultimately the Print function acts on the MLSolution components, so the difference between having a System or not is that the System solves for the Errors and then uses those errors to update the MLSolution.

The systems are practically always BLOCK-type. They in fact may involve multiple scalar unknowns and therefore you have a block structure.

(Well, unless you wanted to sort them dof by dof instead...)

### 30.1.1 Element Residual and Jacobian

The block way

```
vector < vector < double > > Res(n_unknowns);    //every row, every dof ind
vector < vector < vector < double > > > Jac(n_unknowns); //every row, every column,
```

## 30.2 Solvers and preconditioners

### 30.2.1 Linear Equation Solver (child of Linear Equation)

This is to handle  $Ae = -r$

The LinearImplicitSystem has a vector of LinearEquationSolver objects (one for each mesh level)

Every LinearEquationSolver has a Solver (which somehow tends to be called Smoother often times) and a Preconditioner

### 30.2.2 Sparsity pattern

When you have a matrix to store, you need to define a sparsity pattern

The classes where this is taken care of are

LinearImplicitSystem, in the init() function

LinearEquation, in the InitPde function

### 30.2.3 ML vs. MG solver

In MG you pass the Smoother, while in ML you don't.

## Chapter 31

System(s) (of a Problem):  
Example





## Part VI

# Application: Run



### 31.1 How to run in parallel with threads instead of cores

"nproc" gives the number of threads. If your CPU has 2 threads per core, "mpirun -n " works up to nproc/2

To overcome this, do

"mpirun -map-by socket:OVERSUBSCRIBE -n ..."

### 31.2 How to schedule a suite of runs

- Script

### 31.3 How to change the mesh file of an application

- Put the mesh file in the input/ directory
- Set the filename in the main file

### 31.4 When you change the input file in the repo folder (NOT IN THE BINARY FOLDER) then you have to RERUN CMAKE !!!

### 31.5 How to RESTART a run of an application

- THE WHOLE RUN is in a single "time" DIRECTORY... so to restart we will EXCLUSIVELY READ from THAT DIRECTORY

- Set the folder from which to restart in the file run\_to\_restart\_from
- set the ITERATION NUMBER in the "initial\_step" variable in the configuration file IN THE FOLDER OF THE CONSIDERED RUN!
- set the other parameters like "nsteps"
- We must be very careful.
- We must change ALMOST NOTHING
- Do not change NUMBER OF PROCESSORS
- Do not change NUMBER OF LEVELS
- Do not change the OUTPUT FOLDERNAME (of course...)



## Part VII

# Visualization: Paraview



## Chapter 32

# Selections

If you want to select only some cells based on a Threshold value, use the Threshold filter

If you want to select only some cells interactively, do Interactive Cells mode on

If you want the Edges, do Extract Edges first

What if I want to extract all the Faces? Should I split each element, and then Extract Surface for each element??

You can also use Edit -> Find data to establish Selection criteria





# Chapter 33

## Plots

- plot in the x and y directions
  - plot along the axis
  - warp in z direction
  - warp in y direction ("boat")
  - glyphs
  - streamlines



## Chapter 34

# COMPARE/DIFF RESULTS in paraview

Can I compare results in paraview? Well, you should have the same geometry, with the same mesh. Also, if you want the same numbering, you should have the SAME NUMBER OF LEVELS and the SAME NUMBER OF PROCESSORS...

You could make a diff of the hdf5 files, but if the NODE ORDERING is different, then you're gonna have even the same values but in different order! So you have to use the SAME LEVS and SAME PROCS, unless paraview has a GEOMETRIC DIFF, based not on the numbering but on the COORDINATE POSITIONS!!!

I found this script for Programmable filter <http://rbfzone.blogspot.com/2011/06/difference-two-datasets-in-paraview.html> Still it doesn't work to me with different files, but in the same file it works

```
# Get the two inputs A = self.GetInputDataObject(0, 0) B = self.GetInputDataObject(0,
1) s0 = A.GetPointData().GetScalars('Becontx') s1 = B.GetPointData().GetScalars('Becontx_LEVEL2')
out = vtk.vtkDoubleArray() out.SetName('Set Signed Err') out2 = vtk.vtkDoubleArray()
out2.SetName('Set Unsigned Err') out3 = vtk.vtkDoubleArray() out3.SetName('Set
Rel Err') for i in xrange(A.GetNumberOfPoints()): val1 = s0.GetValue(i) val2 =
s1.GetValue(i) val_err = val1 - val2; out.InsertNextValue(val_err) out2.InsertNextValue(abs(val_err))
out3.InsertNextValue(abs(val_err)/abs(val1)) # Initialize the output and add
the labels array output = self.GetOutput() output.ShallowCopy(B) output.GetPointData().AddArray(out)
output.GetPointData().AddArray(out2) output.GetPointData().AddArray(out3)
```



## Chapter 35

# Warp of Cell Data

Use the following filters

- Shrink, with a factor of 0.9999, to get disconnected nodes
- Cell Data to Point Data
- Warp



## Part VIII

**File formats (could be both  
for Input and for Output):  
HDF5**





## Chapter 36

# HDF5

- How can I DIFF two HDF5 files?!?  
Yes there is a utility, called "h5diff"!



## Chapter 37

# XDMF

A way to read HDF5 files from within Paraview



## Chapter 38

# HDFView

Follow the instruction in the HDFVIEW website

At the end of the build process, you should have an `hdfview.sh` script inside the `bin` directory of your build. You must open that script and change the `INSTALLDIR` variable with the path of this build (by default, `/usr/local` is in it)

Another alternative to view the content of an HDF5 file is to use the `'h5dump'` utility shipped with HDF5 (installed through PETSc, for instance)

You can change the names of the Datasets from HDFVIEW



## Part IX

# Code maintenance





## Chapter 39

# Programming practices

- Use .hpp for header files and .cpp for source files
  - Use `std::cout`, do not use `printf` for output.
  - If a function must be performed by only one processor (processor 0), then write an `ifproc` INSIDE the FUNCTION (not outside).
  - Divide the includes in FEMuS includes, C++ includes, other external libraries' includes.
    - Always put include guards in header files.
    - Try to put the includes EXPLICITLY WHERE THEY HAVE TO BE, and not thinking that somewhere some include will lead me to the file i need...
    - In the SCRIPTS, almost all the variables have a `FM_` or `FEMUS_` prefix. The variables without this prefix are like this because they are defined for external packages For instance, `PETSC_DIR` and `PETSC_ARCH` are used to COMPILE PETSC.
    - when you create a NEW HEADER, make sure you change the `INCLUDE GUARD`. Then, make sure you add the new files to git.
    - when you change the name of a file, header or source, recompile all the applications and make sure that everything compiles correctly.
    - when you start a file, put the source code among the "namespace femus"
    - data encapsulation, please
    - it is good to create small functions: you isolate tasks, you define interfaces so you can highlight what is "const" and what is not, you can avoid code duplication, etc.

### 39.1 How to add source files to the library

- NOW PAY ATTENTION TO WHAT HAPPENS WHEN YOU ADD SOURCE FILES to the LIBRARY (not to the application)! You have to update the LIST in the `src/` `CMake` file!!! (use `ls -l`)
  - now go to the binary directory and type `make` to generate the library



# Chapter 40

## Git

- First golden rule: using the manual never hurts.
  - Set name and email in your computer:  
`git config --global user.name "Name Surname" git config --global user.email "name.surname@example.com"`  
To check what configuration you obtained, do  
`git config --list`  
(this shows you more than you see with `"git config -e"` because it also reads other git configuration files, such as `$HOME/.gitconfig`)
  - To contribute: Create a personal github account Create a fork of the femus repository in your github account Clone your fork in your computer Work in your computer (do branches, commit changes, ...) Push your branches back to your fork Send a pull request to the main femus repository The maintainers will decide what to do with the pull request and possibly it will be merged to master Periodically, sync the master in the fork with the master in the main femus repository

### 40.1 WORKFLOW for updating the master in the FORK from the master in the MAIN REPO

In github:

- Go in the MAIN REPO
- Click on "Pull requests" (on the right)
- Click on "New pull request" (green button)
- Click on "compare across forks"
- The "base fork" is going to be the FORK master branch
- The "head fork" is going to be the MAIN master branch
- Click on "create pull request"
- Add some title for it

Make sure that the branches can be AUTOMATICALLY MERGED (otherwise you have to solve the conflicts using command line...)

Click on "Merge pull request" (you find it by scrolling towards the bottom of the page)

Click on "Confirm merge"

From command line:

To be added

## Chapter 41

# Debugging, parallel

Put yourself in the simplest case

First of all, check that the coarse mesh has enough elements: if the coarse mesh only has 1 element, then Metis does not work

Then, try to only run Mesh and Printing, by commenting the solution of your System. See how the Metis partitions are printed.

Fine, it works.

### 41.0.0.1 Use valgrind!!!

You could run valgrind in parallel

```
mpiexec -n 2 valgrind myapp
```

### 41.0.0.2 Command

```
mpiexec -n 2 myapp -start_in_debugger
```

### 41.0.0.3 Command, better with kdbg!

```
mpiexec -n 2 kdbg myapp
```

Then you have to click on Run in all windows

### 41.0.0.4 Segfaults

If you use -info it tells you a lot of things about PETSC calls. -log\_trace, -info, -log\_summary

Finding a segmentation fault shouldn't be too difficult with valgrind or gdb (see also this link: <http://www.cprogramming.com/debugging/segfaults.html>)

Di solito guardi uno dei puntatori nella riga in cui si blocca e magari e' un puntatore non inizializzato, oppure e' un puntatore a NULL, oppure e' un puntatore che e' stato deallocato, oppure sto dereferenziando un puntatore al di fuori del suo range, oppure qualcosa altro ma poco...

Per quanto riguarda le PETSC puo' essere molto lungo riuscire a trovare un segmentation fault, perche' dovresti andare a vedere DENTRO LE STRUTTURE (Mat, Vec sono strutture di C) e guardare di quegli elementi qual e' quello che causa il segmentation fault.

Per fare questo conviene USARE LE PETSC in DEBUG MODE e vedere quale output ti danno; cosi' vedi le chiamate all'interno della libreria petsc.

#### 41.0.0.5 Parallel numbers different from serial numbers

That means that the matrices and rhs are filled differently in serial and parallel

We then must print either the Element Matrix and Rhs or the Global Matrix and Rhs and see

#### 41.0.0.6 How to visualize residuals, errors, matrices

- 1) Either you print them to file as NumericVector
  - 2) or you print them in Paraview as Solution objects
  - 1) If you print as NumericVectors you don't really see the subdivision between levels, processors, variables, so one should do some print\_solution for that
- Does a Solution have the same dofmap structure of the underlying NumericVector?

Not exactly.

In a Solution there is a vector of NumericVectors, one for each scalar unknown

In the LINEAR EQUATION, there is a single NumericVector with all pieces attached.

The dofmap structure in the Linear Equation is constructed based on the list of Solutions that are Unknowns

# Chapter 42

## Documentation

Add `BUILD_DOCUMENTATION` in Cmake

We have a configuration file `DoxyfileFemuslib.in` that is used by Cmake to generate `DoxyfileFemuslib`

This last file is going to be the input to `doxygen`

However, we don't need to know exactly where these files are.

We just need to go in the build directory and type "make Docs"

For each member datum or member function, Doxygen takes the comments in the form `/** ... */`

or in the form `/// ...`

both from the `.hpp` and from the `.cpp` files

Inside the comments that are processed by Doxygen, one can use

- `@todo ...`, to generate a Todo list page
- `@deprecated ...`, to generate a Deprecated list page
- `@test`, to generate a Test list page
- `@bug`, to generate a Bug list page

- Doxygen could even generate a Latex output if `GENERATE_LATEX` is set to yes in the configuration file!

- Doxygen could even generate man pages if `GENERATE_MAN` is set