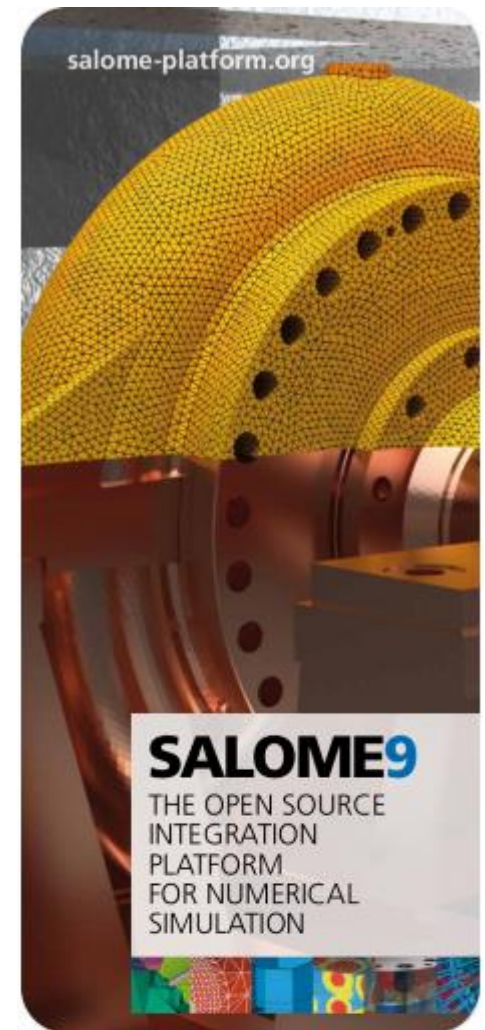




MEDCOUPLING, a short introduction

Paul RASCLE
September 20, 2019





MEDCoupling : input / output data manipulations

- ▶ Field-Physics simulation studies require manipulations of
 - ▶ *Mesh* = spatial discretization of a geometric domain
 - ▶ *Fields* = physical data / boundary conditions on the above discretization
- ▶ **Data movement** : Read, write meshes and fields
 - ▶ From / to files
 - ▶ Process data in memory between process (exchange, extract, duplicate...)
- ▶ **Data analysis** : extract / gather information
- ▶ **Data conversion** : exchanges between simulation codes
 - ▶ Interpolation, projection, conversion of format...
- ▶ **Data optimization** : for simulation codes
 - ▶ Renumbering
 - ▶ Partition for parallel codes
 - ▶ ...

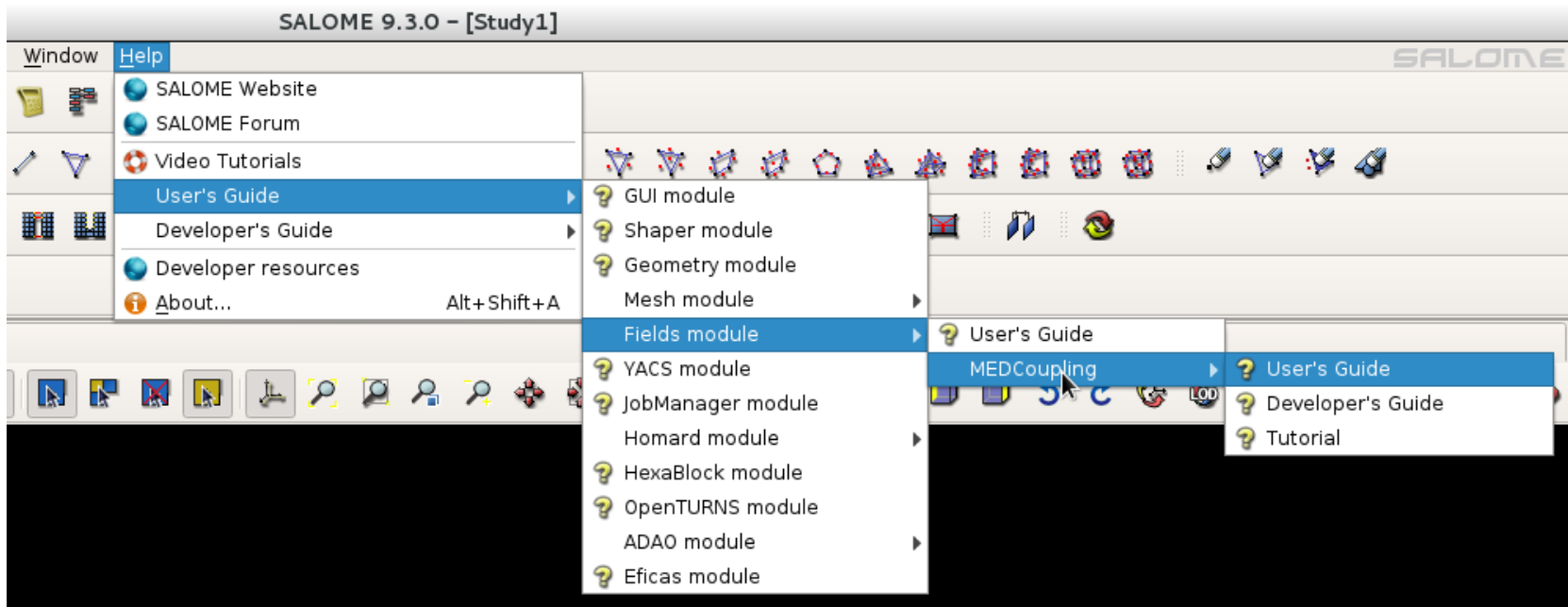


MEDCOUPLING : A powerful set of libraries

- ▶ C++ and Python interfaces
- ▶ C++
 - ▶ Similarity with VTK code structures
 - ▶ Ease the interaction with VTK (ParaView plugins)
- ▶ Python (swig from C++)
 - ▶ DataArrays : behave as Numpy arrays
 - ▶ Easy conversion, with zero copy

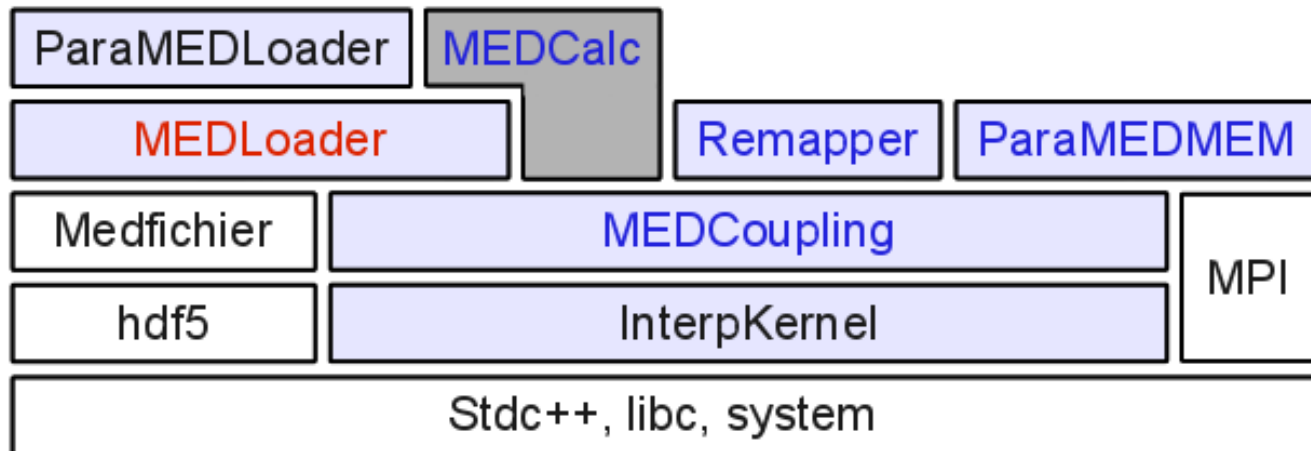
Medcoupling : documentation

- ▶ With SALOME 9.3 binaries:
 - ▶ User's guide, developer's guide, Tutorial



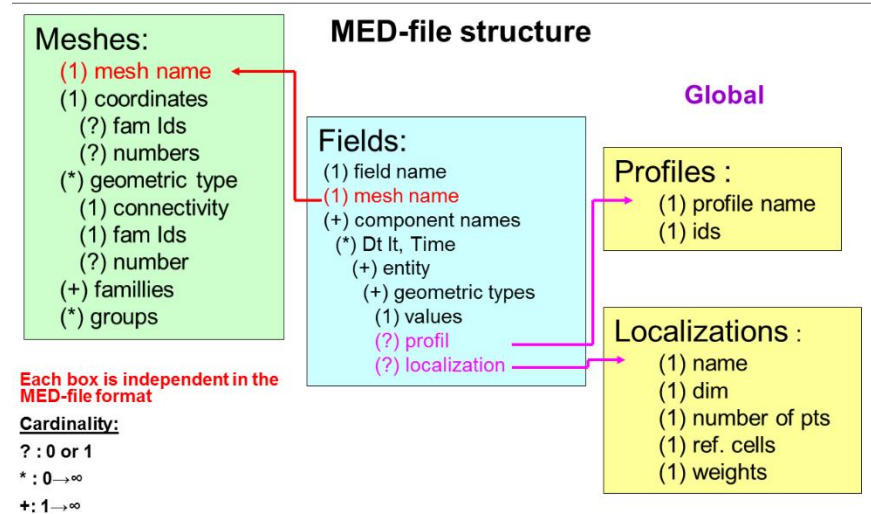
MEDCoupling Architecture

- ▶ **MEDCoupling** :data structures used for cross process exchange of meshes and fields.
- ▶ **MEDLoader** and **ParaMEDLoader** : I/O functions to the MED file format
 - ▶ sequential and parallel processing
 - ▶ built on top of the MED-file library.
- ▶ interpolation tools : mathematical structures and algorithms for interpolation and localization.
 - ▶ InterpKernel, Remapper and ParaMEDMEM (Remapper with parallel processing).



MED file structure

- ▶ several meshes
 - ▶ Several fields
 - ▶ With/without time steps
 - ▶ With/without Profile
(if on a part of the mesh)
 - ▶ ...
- ▶ MED file documentation with the binaries
 - ▶ In French ☹️
- ▶ A lot of methods in MEDLoader depending on the complexity of MED file
 - ▶ Several ways to do something
 - ▶ Reading: get information on content, simple methods for simple structures (optional arguments)
 - ▶ ...





Example: Reading a field

```
import MEDLoader as ml
import numpy as np
# first way to get a mesh
source = ml.MEDFileData("Demo1.med")
meshsrc=source.getMeshes()
msrcm=meshsrc[0]
m0=msrcm[0]
# 2nd way to get a mesh (for the first one)
m1 = ml.ReadMeshFromFile("Demo1.med")
# explore fields names
fieldsNames = ml.GetAllFieldNames("Demo1.med")
# triplets (iteration, order, time) on first field
itf1 = ml.GetAllFieldIterations("Demo1.med", fieldsNames[0])
# access to a timestep with iteration, order
f1 = ml.ReadField("Demo1.med", fieldsNames[0], 1, 1)
# use numpy (no copy)
npf1 = f1.getArray().toNumPyArray()
```



Exercise: field transfer from one mesh to another

- ▶ Two meshes provided : Fine.med, Coarse.med
 - ▶ Read the meshes, read field on Fine.med: "MeasureOfMesh_Fine"
 - ▶ The Field represents the cell volume
 - ▶ Some exploration : min, max, cells id in a certain range...
 - ▶ Field transfer from fine to coarse
 - ▶ Extensive data conservation
 - ▶ Both fields are constant on cell: P0P0
 - ▶ Compare the integral of field on both meshes
 - ▶ Write a new file with coarse mesh and field
 - ▶ Retrieve Skin cells, write a new mesh with 2D Cells...
- ▶ Get SlovenieMEDCoupling.py script
 - ▶ Execute it step by step in embedded Python console of SALOME
 - ▶ Examine objects...



PARTNERSHIP FOR ADVANCED COMPUTING IN EUROPE

Thank you !
