

---

# TRUST/TrioCFD 1.7.4 user's training session

---

# TRUST/TrioCFD 1.7.4 training sessions

---

- Users training session:
  - 1<sup>st</sup> day: TRUST/TrioCFD presentation & practices,
  - 2<sup>nd</sup> day: Automated test case with TRUST presentation & practices, mesh with tools (Xprepro, Salomé or Gmsh) presentation & practices, TRUST/TrioCFD practices.
- Developer training session:
  - 1<sup>st</sup> day: Basic OOC concepts used in TRUST, practices, Baltik project, Exploring the Kernel modules, How to debug TRUST,
  - 2<sup>nd</sup> day: Managing input/output files with TRUST classes, How to parallelize in TRUST, TRUST test coverage, TRUST coding rules
- Custom user session of one day to help on a specific problem

Céline CAPITAINÉ (CS) - Marthe ROUX (CS)  
TRUST/TrioCFD support team  
[triu@cea.fr](mailto:triu@cea.fr)

# Table of contents

---

- TRUST/TrioCFD historic..... p4
- Modeling flow with TRUST/TrioCFD..... p9
- Examples of performed calculations..... p16
- Models, schemes, numerical methods.....p21
- Data files & calculation.....p49
- Command lines..... p80
- Parallel calculation..... p87
- Mesh generators: Internal tools & Salomé & Gmsh... p109
- Automating validation test case .....p125
- TRUST/TrioCFD support.....p143
- Examples of data files.....p146
- Recommendations..... p167

# Table of contents

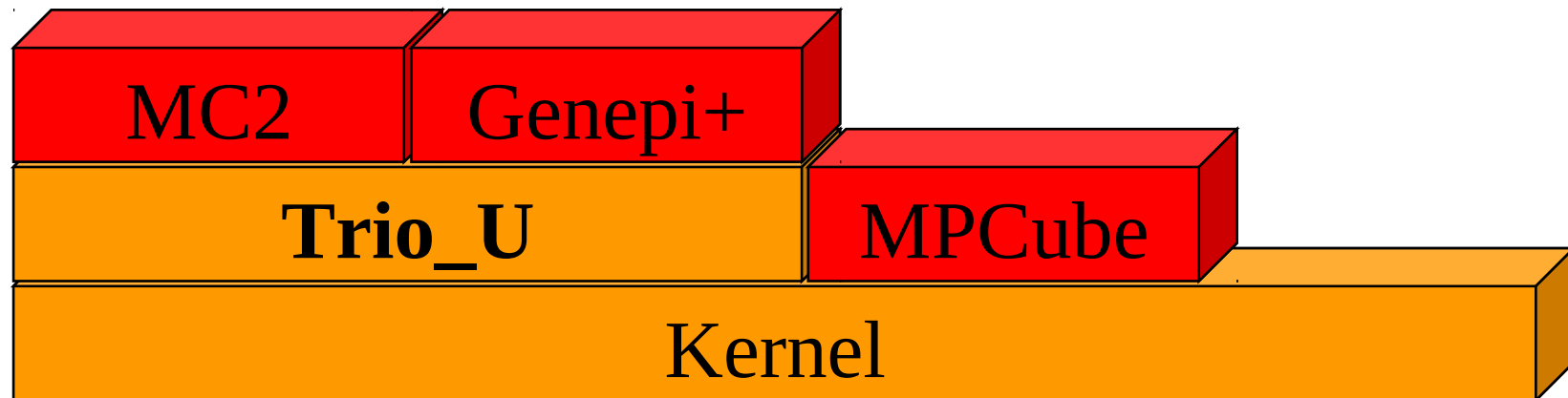
---

- **TRUST/TrioCFD historic**
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Salomé/Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations

# TRUST/TrioCFD historic (1/4)

---

- Trio\_U code: CFD code for incompressible monophasic / diphasic flow
- Developed at the CEA/DEN/DANS/DM2S/STMF service
  - TRUST project leader : [gauthier.fauchet@cea.fr](mailto:gauthier.fauchet@cea.fr)
  - TrioCFD project leader: [anne.burbeau@cea.fr](mailto:anne.burbeau@cea.fr)
- Trio\_U, a software brick used by other CEA applications:



## TRUST/TrioCFD historic (2/4)

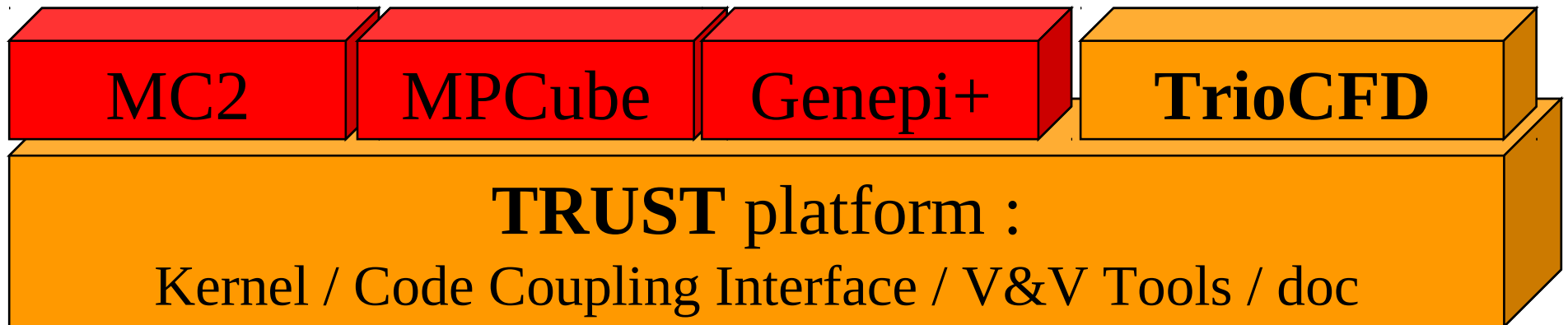
---

- 1994: start of the project Trio\_U
- 01/1997 : v1.0 (VDF only)
- 06/1998 : v1.1 (VEF version)
- 04/2000 : v1.2 (parallel version)
- 07/2001 : v1.3 (radiation model)
- 11/2002 : v1.4 (new LES turbulence models)
- 02/2006 : v1.5 (VDF/VEF Front Tracking)
- 10/2009 : v1.6 (data structure revamped)
- 06/2015 : v1.7 (cut into TRUST and TrioCFD  
+ switch to open source)

# TRUST/TrioCFD historic (3/4)

Since version 1.7.1, Trio\_U is divided in two parts:

- a **BALTIK** project named **TrioCFD** based on new platform named **TRUST**,  
$$\text{Trio\_U} = \text{TRUST} + \text{TrioCFD (FT, Radiation, LES, zoom)}$$
- Where :
  - **TRUST**: “**TRio\_U Software for Thermohydraulics**”,
  - **BALTIK**: “**Build an Application Linked to Trio\_U Kernel**”.



- The kernel contains the equations, space discretizations, numerical schemes, parallelism...

# TRUST/TrioCFD historic (4/4)

---

- Main CEA goals:
  - R&D platform for fluid mechanics
  - To advocate LES turbulence models or RANS-LES coupling for nuclear safety studies
  - Codes coupling (fluid, structure, neutronic...)
  - Alternative to commercial CFD software
  - To base on advanced technology (C++, COO, //...)



# Table of contents

---

- TRUST/TrioCFD historic
- **Modeling flow with TRUST/TrioCFD**
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations

# Modeling flow with TRUST/TrioCFD (1/6)

## Incompressible single phase flow

- Laminar or Turbulent flow
- Navier Stokes with or without energy equation
- Incompressible fluid or with low variation for volumic mass
  - Boussinesq hypothesis:  
 $\rho = \rho(T) \sim \rho_0 - \beta(T - T_0)$
  - Quasi-compressible model:  
 $\rho = \rho(P, T)$  for low mach numbers

$$\nabla \cdot \vec{u} = 0$$

$$\frac{\partial \vec{u}}{\partial t} + \nabla \cdot (\vec{u} \otimes \vec{u}) = \nabla \cdot (\nu \nabla \vec{u}) - \nabla P^*$$

$$\frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T = \nabla \cdot (\alpha \nabla T) + \frac{Q}{\rho C_p}$$

$$P^* = \frac{P}{\rho} + g z$$

# Modeling flow with TRUST/TrioCFD (2/6)

## Description of the Quasi Compressible model used

- Accounts for space and time variations of density due to high changes of temperature:

$$\text{Ideal gas law: } \rho(\vec{x}, t) = \frac{P_0(t)}{RT(\vec{x}, t)}$$

- Filters acoustic waves in order to avoid too small time-step:

$$\text{Total pressure: } P(\vec{x}, t) = P_0(t) + P_1(\vec{x}, t) \quad \text{with } P_1 \approx M^2 P_0 \text{ and } M = \text{Mach} \ll 1$$

$$\text{Thermodynamic pressure : } P_0(t)$$

$$\text{Hydrodynamic pressure : } P_1(x, t)$$

- Set of equations solved:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0$$

$$\frac{\partial (\rho \vec{u})}{\partial t} + \nabla \cdot (\rho \vec{u} \vec{u}) = \nabla \cdot (\mu \nabla \vec{u}) - \nabla P_1^* - \rho \vec{g}$$

$$\rho C_p \left( \frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T \right) = \nabla \cdot (\lambda \nabla T) + \frac{dP_0}{dt} + Q$$

$$P_0 = \rho R T$$

$$P_1^* = P_1 + \frac{2}{3} \mu \operatorname{div}(\vec{u})$$

# Modeling flow with TRUST/TrioCFD (3/6)

- Heat exchange
  - Conduction
  - Radiation in transparent medium
  - Radiation in semi-transparent medium
- Transport of passive scalars  $\frac{\partial C_i}{\partial t} + \vec{u} \nabla C_i = \text{Div} (D_i \nabla C_i)$
- Porous Media
  - Surface or volume porosities
  - Singular or regular pressure loss

# Modeling flow with TRUST/TrioCFD (4/6)

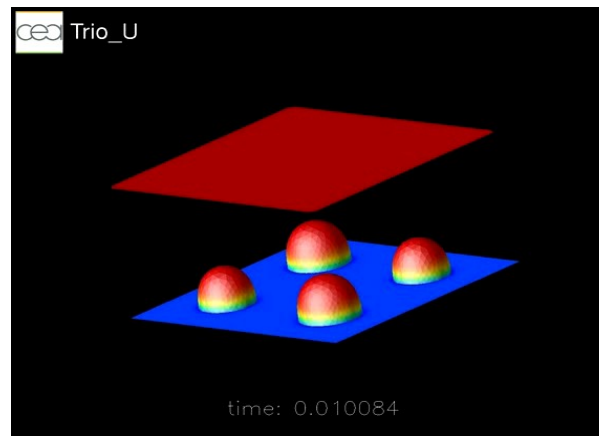
---

- Particles transport model:
  - One way coupling
    - Particle motion affected by the flow
  - Two way coupling
    - As above but particle disturbances also affect the flow
  - Possible to convert droplet/bubble below a given size into particles during a Front Tracking calculation

# Modeling flow with TRUST/TrioCFD (5/6)

## Front tracking model

- Two phases flow
  - Eulerian mesh where Navier Stokes equations are solved
  - Lagrangian moving mesh for the interface locations
  - Coalescence or breakup models for bubbles and drops



Boiling bubbles

- Can be declined in TrioCFD to use an Immersed Boundary Method using IBC (Immersed Boundary Conditions)

# Modeling flow with TRUST/TrioCFD (6/6)

---

- 2D calculation
  - Plane, Cartesian coordinates (x,y)
  - Axi-symmetric, coordinates (r,z) (VDF only, k-eps OK)
- 3D calculation
  - Cartesian coordinates (x,y,z)
- Transient flow calculation calculated by:
  - Explicit, semi-implicit or implicit time schemes
- Steady state calculation obtained:
  - By convergence of the transient flow

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- **Examples of performed calculations**
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations



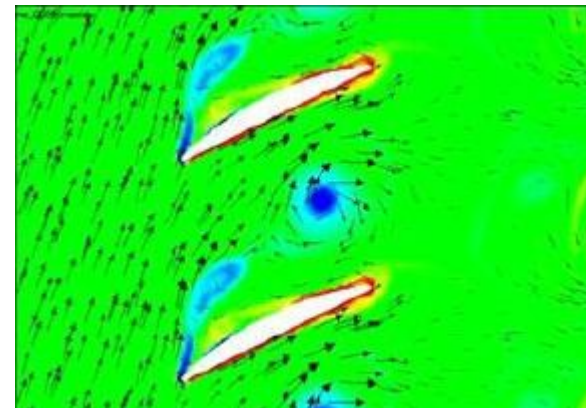
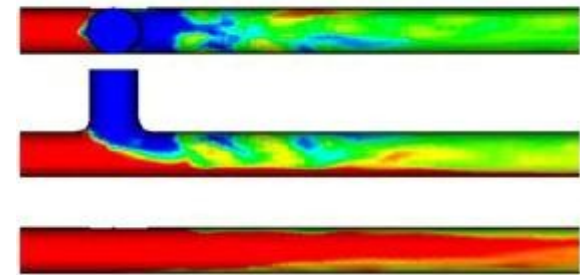
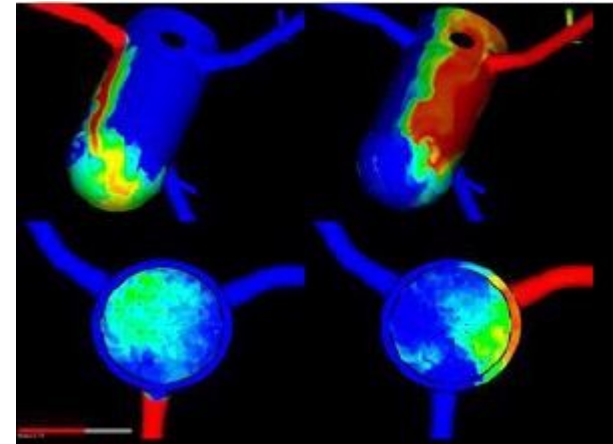
# Examples of performed calculations (1/2)

---

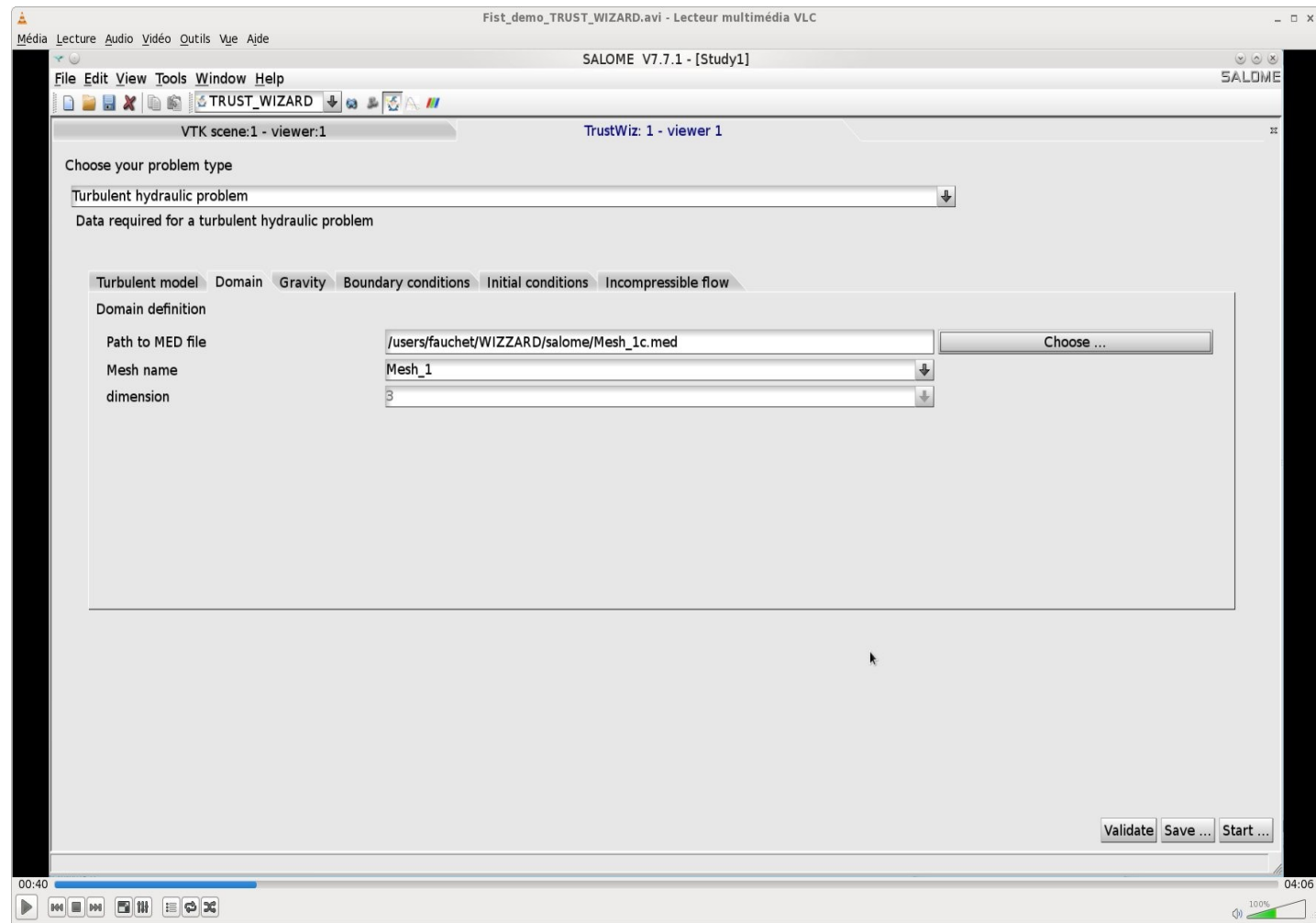
- Academic calculations
  - Plane channel with conduction coupling at the wall
  - Flow around obstacle
  - Pipe flow
  - Impinging jet
  - Isotropic turbulence

# Examples of performed calculations (2/2)

- Industrial calculations
  - Various studies about the core of a reactor
  - Thermal stress in a T-shaped mixing pipe
  - Natural convection in a storage room of waste
  - Atmospheric dispersion (polluting or radio-nucleid)
  - Compressor blades in GFR (Gas Fast Reactor)



# IHM TRUST



# Practice

---

## Using TRUST PLOT2D tools:

- > source /home/triou/env\_TRUST\_X.Y.Z.sh
- > echo \$TRUST\_ROOT
- > mkdir -p ~/test/yourname
- > cd ~/test/yourname
- > trust -copy Obstacle
- > cd Obstacle
- > trust -evol Obstacle
  - “Start computation!”: Wait until 100%
  - Visualize a probe: select a probe and click on “Plot”
  - Visualize a field with VisIt: “Visualisation”

# Table of contents

---

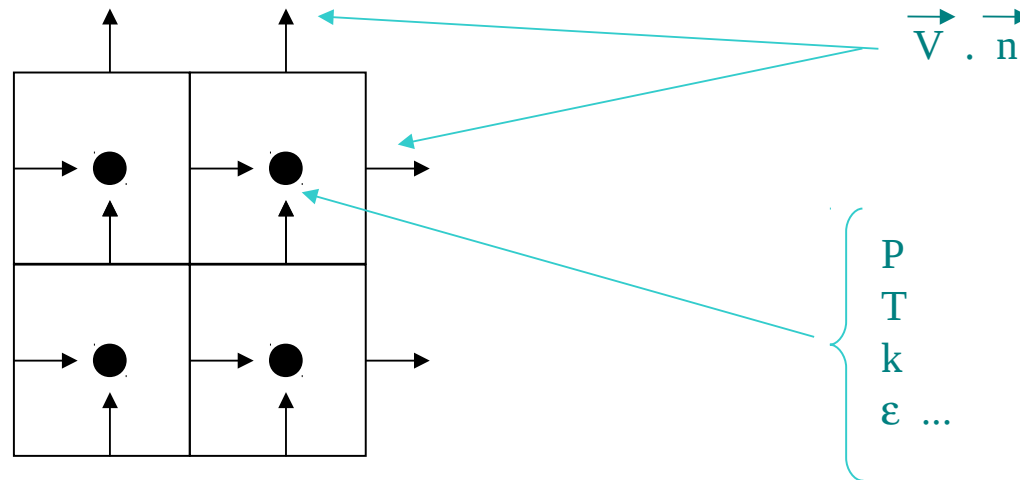
- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- **Models, schemes, numerical methods**
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations

# Models, schemes, numerical methods (1/6)

- **Discretizations (VDF/VEF).....p22**
- Time and space schemes.....p28
- Boundaries conditions.....p35
- Source terms.....p39
- Solvers for linear systems.....p43
- Turbulence models.....p47

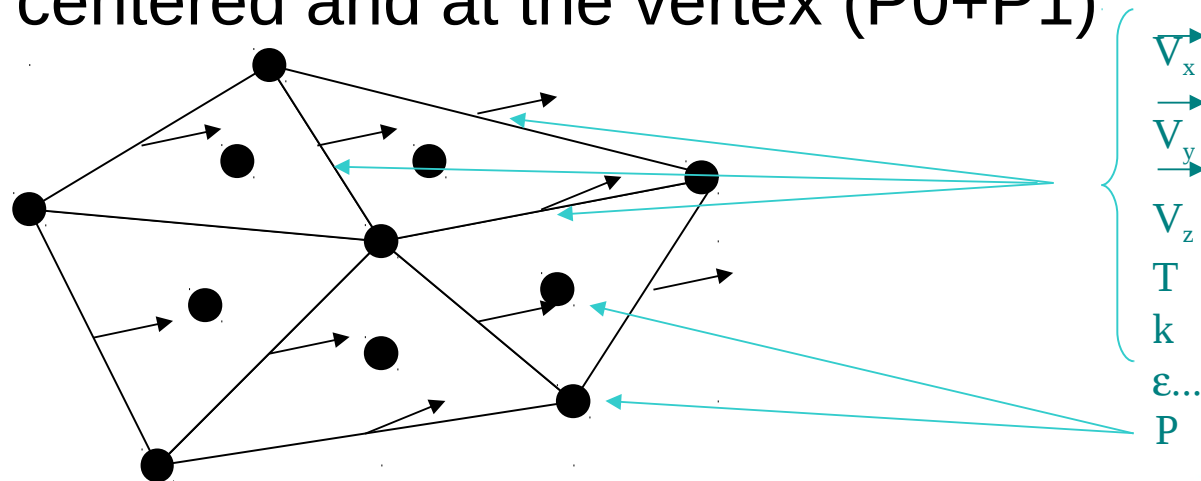
# Discretizations (1/5)

- **Finite Differences Volumes (VDF)**
  - Structured meshing ; velocity and pressure are shifted
  - Scalar unknown is mesh centred
  - Velocity is normal at the face



## Discretizations (2/5)

- **Finite Elements Volumes (VEF)**
  - Unstructured meshing triangles (2D) or tetrahedrons (3D)
  - Unknown fields are face centered (P1NC)
  - Physical characteristics are cell centered
  - Pressure :
    - Mesh centered and at the vertex (P0+P1)

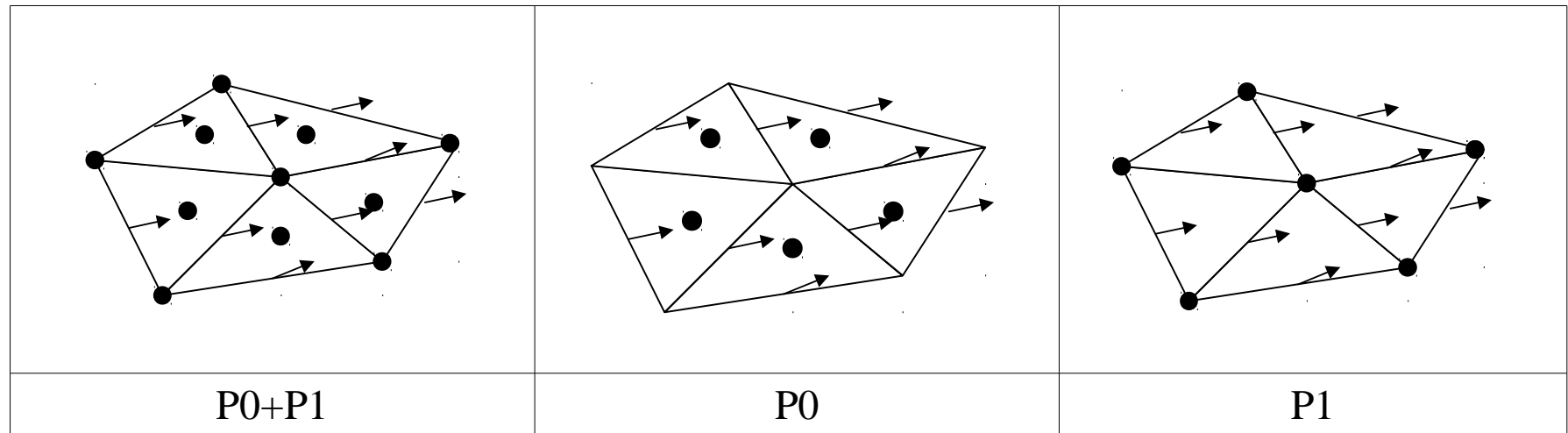




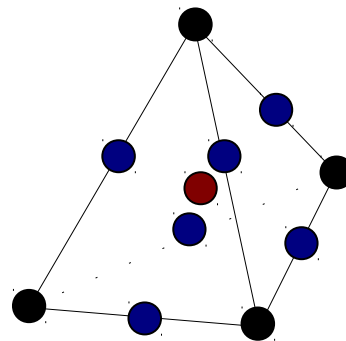
# Discretizations (3/5)

- **Finite Elements Volumes (VEF)**

- By default, P0+P1 for pressure but less/more pressure nodes is possible:



Plus in 3D: P0+P1+Pa



11 pressure nodes per tetra:

-1 in center (P0)

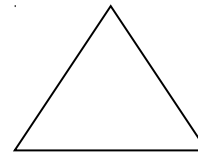
-4 on vertexes (P1)

-6 on edges (Pa)

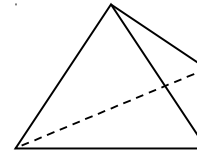
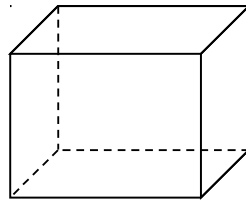
-> **Useful** for flow with a strong source term & a low velocity field where P0+P1 pressure gradient P0+P1 has trouble to match the source term

# Discretizations (4/5)

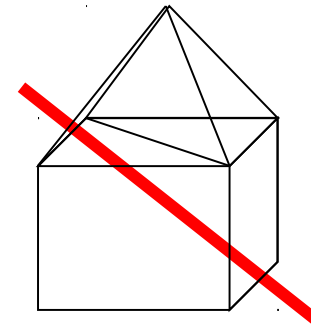
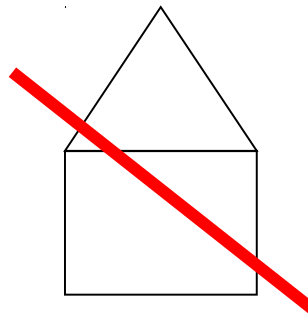
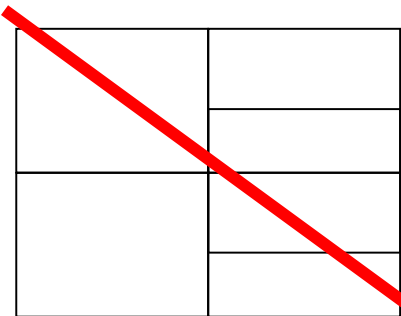
- Kind of mesh permitted
  - Quadrangular or triangular meshing for 2D cases



- Hexahedral or tetrahedral meshing for 3D cases

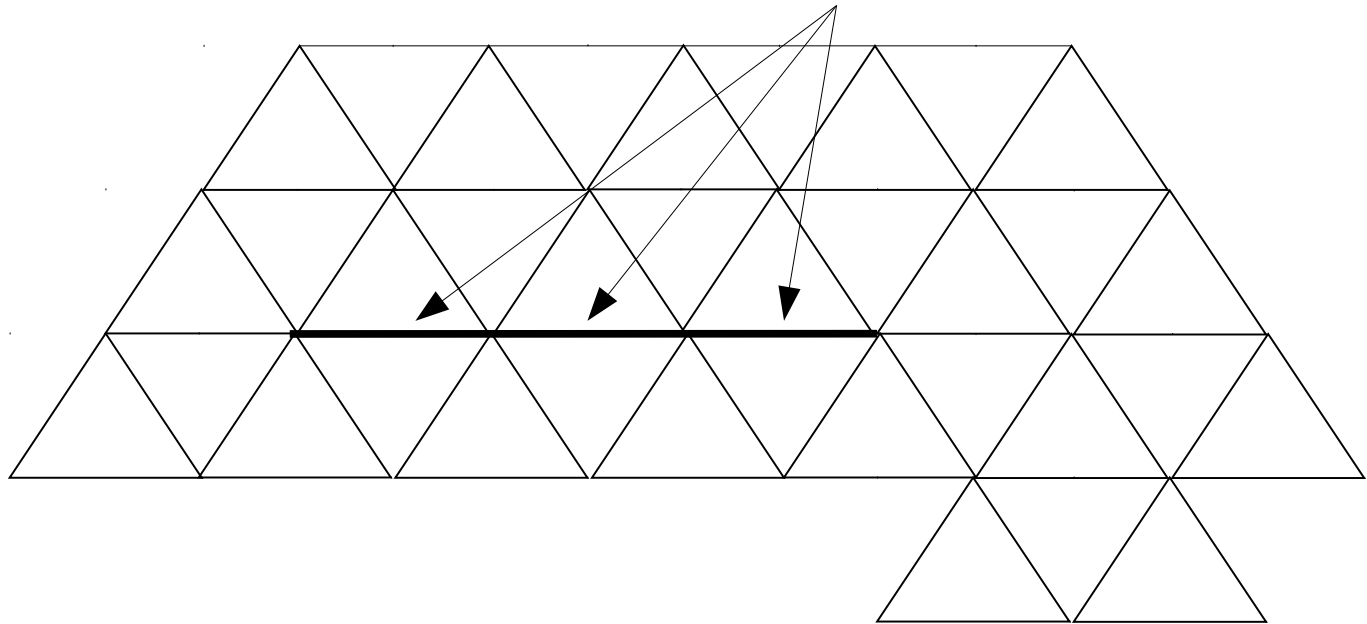


- Non standard or hybrid meshing not supported



# Discretizations (5/5)

- Kind of mesh permitted
  - Internal boundaries partially supported
    - with ICEM, spit faces in two and define boundaries
    - TRUST will differentiate the two faces



# Models, schemes, numerical methods (2/6)

- Discretizations (VDF/VEF)
- **Time and space schemes**
- Boundaries conditions
- Source terms
- Solvers for linear systems
- Turbulence models

# Some time and space schemes (1/6)

- Explicit time schemes:

$$\frac{\partial I^{n+1}}{\partial t} + \vec{u}^n \nabla I^n = \text{Div}(\alpha \nabla I^n)$$

- Euler explicit (order 1)
- Runge Kutta (order 2 or 3)

- Semi-implicit time scheme:

$$\frac{\partial I^{n+1}}{\partial t} + \vec{u}^n \nabla I^n = \text{Div}(\alpha \nabla I^{n+1})$$

- Euler semi-implicit (diffusion implicit)

- Implicit time schemes (not unconditionally stable):

$$\frac{\partial I^{n+1}}{\partial t} + \vec{u}^{n+1} \nabla I^{n+1} = \text{Div}(\alpha \nabla I^{n+1})$$

- Implicite, PISO, SIMPLE (dynamic time step)

# Some time and space schemes (2/6)

- Explicit time schemes:

- For Navier-Stokes equations and incompressible flows:

$$(1) \nabla \cdot \vec{u} = 0$$

$$(2) \frac{\partial \vec{u}}{\partial t} = \frac{-1}{\rho} \nabla P - \nabla \cdot (\vec{u} \otimes \vec{u}) + \nabla \cdot (\nu \nabla \vec{u})$$

Convective term  
Diffusive term

- Equation (2) leads to:  $u^{n+1} = u^n + \Delta t \left( -\frac{1}{\rho} \nabla P - \nabla \cdot (\vec{u} \otimes \vec{u}) + \nabla \cdot (\nu \nabla \vec{u}) \right)$
- We set:  $u^t = u^n + \Delta t \left( -\nabla \cdot (\vec{u} \otimes \vec{u}) + \nabla \cdot (\nu \nabla \vec{u}) \right)$
- Using (1) into (2):

$$0 = \nabla \cdot \left( \rho \frac{\partial \vec{u}}{\partial t} \right) = \nabla \cdot \left( -\nabla P - \rho \nabla \cdot (\vec{u} \otimes \vec{u}) + \rho \nabla \cdot (\nu \nabla \vec{u}) \right)$$

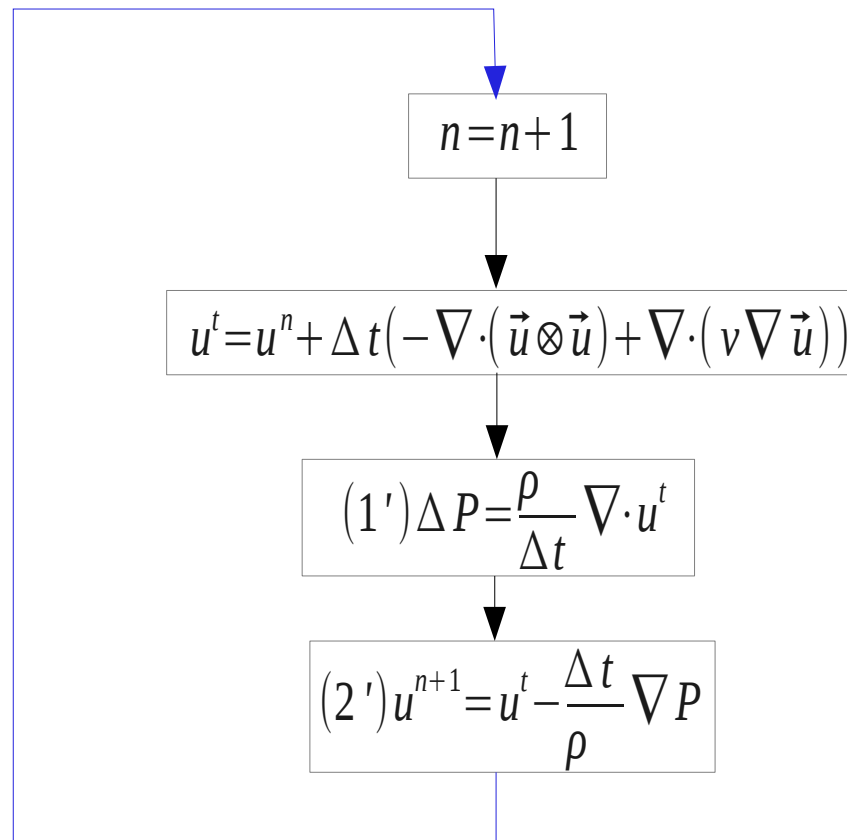
- We get:

$$(1') \Delta P = \frac{\rho}{\Delta t} \nabla \cdot u^t$$

- And (2) becomes:  $(2') u^{n+1} = u^t - \frac{\Delta t}{\rho} \nabla P$

# Some time and space schemes (3/6)

- Explicite scheme: projection method



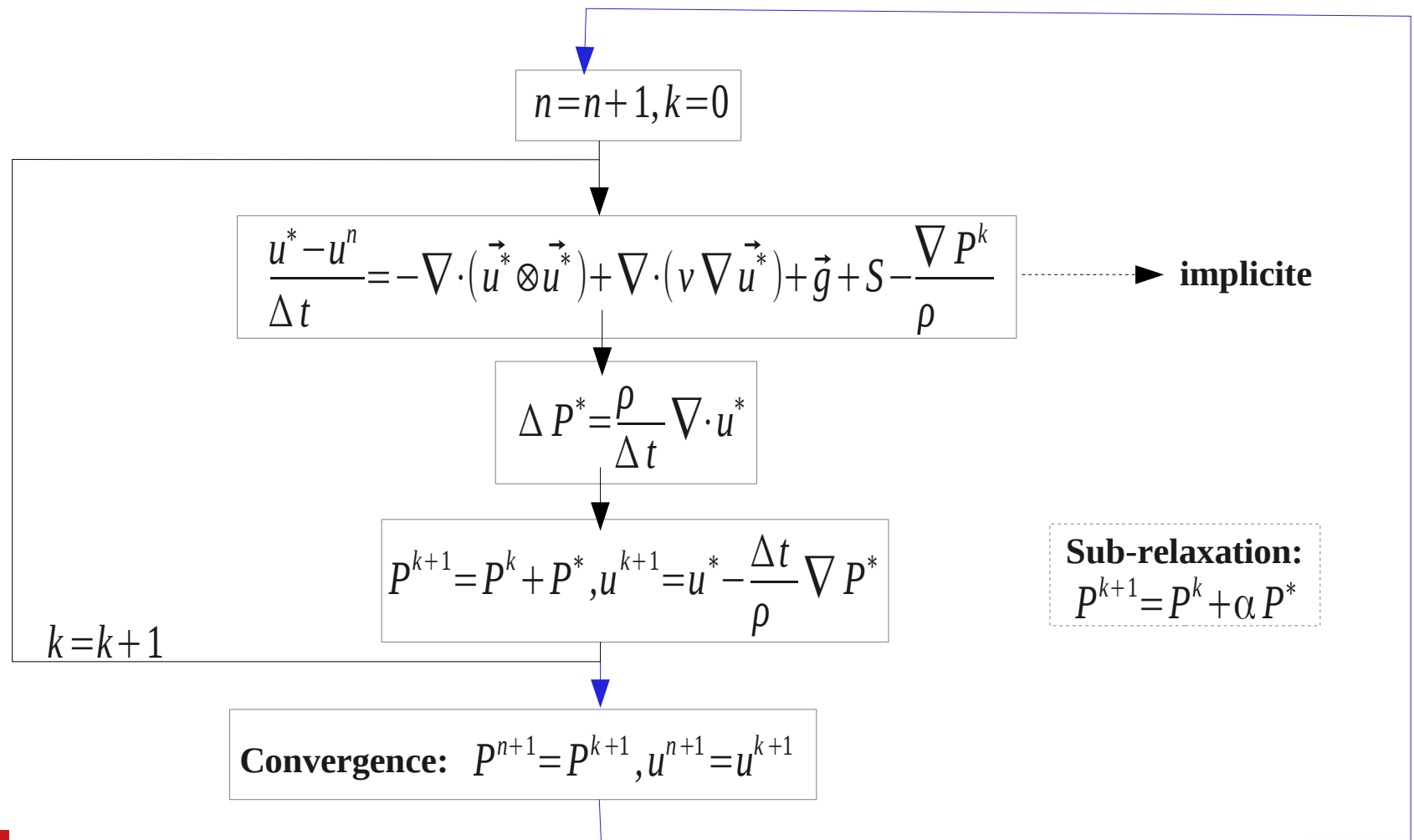
Integrate with out pressure

Poisson's equation

Projection on incompressible fields

# Some time and space schemes (4/6)

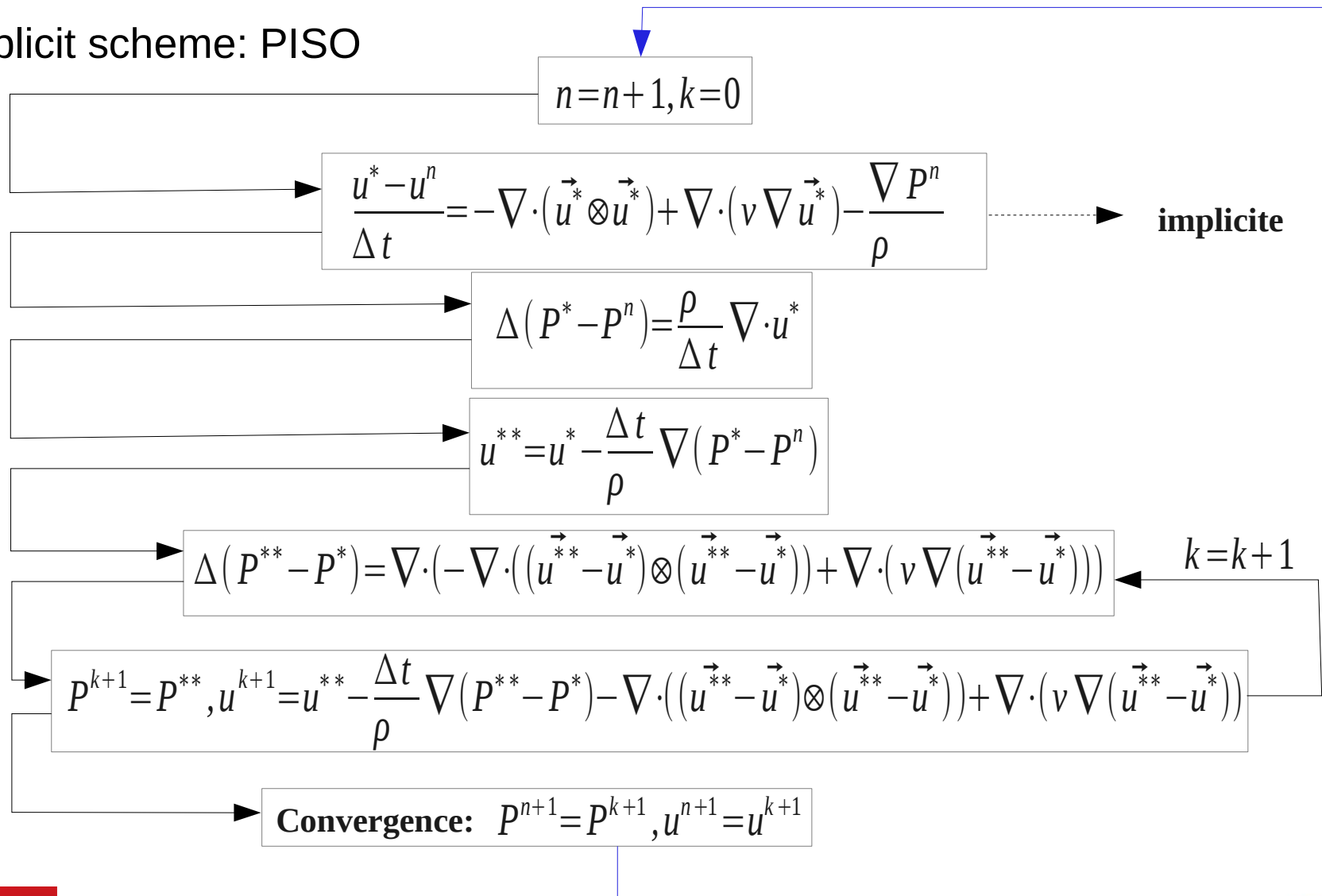
- Semi-implicite scheme: SIMPLE





# Some time and space schemes (5/6)

- Implicit scheme: PISO



# Some time and space schemes (6/6)

---

- convection schemes with VDF discretization
  - Quick (order 2-3)
  - Centre (order 2 or 4) « centered »
  - Amont (order 1) « upwind »
- convection schemes with VEF discretization
  - EF\_stab (order 2) « centered stabilized »
  - Muscl (order 2) « quick like »
  - Amont (order 1) « upwind »

# Models, schemes, numerical methods (3/6)

- Discretizations (VDF/VEF)
- Time and space schemes
- **Boundaries conditions**
- Source terms
- Solvers for linear systems
- Turbulence models

# Boundaries conditions (1/3)

---

Available BC for momentum equation

- Wall:
  - No slip ( $u=0$ )
  - Slipping at the wall ( $u.n=0$ )
  - Imposed tangential velocity
- Fluid boundary:
  - Imposed velocity
  - Imposed pressure or pressure gradient
  - Free outlet
  - Periodic

## Boundaries conditions (2/3)

---

### Available BC for energy equation

- Wall:
  - Imposed temperature
  - Adiabatic or imposed flux
  - Imposed exchange (coefficient)
  - Contact resistance possible between two walls
- Fluid boundary:
  - Imposed temperature
  - No flux
  - Periodic

# Boundaries conditions (3/3)

---

- Boundary condition values may be:
  - Uniform on the boundary
  - Space dependent
  - Time dependent
  - Read in a file

# Models, schemes, numerical methods (4/6)

- Discretizations (VDF/VEF)
- Time and space schemes
- Boundaries conditions
- **Source terms**
- Solvers for linear systems
- Turbulence models

# Source terms (1/3)

---

- Navier Stokes equation:

- Boussinesq

$$S = \rho_0 g \beta (T - T_0)$$

- Useful for small variation of volumic mass

- Flow rate

$$S = Q_m$$

- Pressure loss

$$S = -0.5 \rho C_f U |U| / D$$

- Regular pressure loss (Blasius or  $C_f$  given by the user)

- Periodic channel

$$S = Q_m$$

- Useful to keep constant flow rate into a periodic channel

- ...



## Source terms (2/3)

- Navier Stokes equation:

Calculation into a non Galilean referential  $R'$ . Coriolis and inertial forces, the user specifies:

- Acceleration and velocity of  $R'$  referential into the Galilean referential  $R$
- $\Omega$ ,  $d\Omega/dt$  : rotation and its derivative term into the  $R'$  referential
- A centre of the rotation of  $R'$  into  $R$  with the coordinates given into the  $R'$  referential

$$\vec{F}_{ie} = -m \vec{a}_e = -m (\vec{a}(A)_{(R)} + \left( \frac{d\vec{\Omega}_{(R'/R)}}{dt} \right)_{(R)} \wedge A\vec{M} + \vec{\Omega}_{(R'/R)} \wedge (\vec{\Omega}_{(R'/R)} \wedge A\vec{M}))$$

$$\vec{F}_{ic} = -m \vec{a}_c = -m 2\vec{\Omega}_{(R'/R)} \wedge \vec{v}_r$$

## Source terms (3/3)

---

- Energy equation:

- Volumic heat power

$$S=P$$

- For example into a solid media

- ....

- Concentration equation:

- Boussinesq

$$S=\rho_0 g \beta (C-C_0)$$

- Useful to build a two miscible fluids calculation (if the volume mass is similar). Concentration  $C$  will be equal to the fraction fluid

# Models, schemes, numerical methods (5/6)

- Discretizations (VDF/VEF)
- Time and space schemes
- Boundaries conditions
- Source terms
- **Solvers for linear systems**
- Turbulence models

# Solvers for linear systems (1/3)

Linear systems	Sparse	Symmetric	Constant
Pressure linear system for incompressible flow	X	X	X
Pressure linear system for quasi compressible flow	X	X	X
Pressure linear system for diphasic flow	X	X	
Use of an implicit scheme	X		
Radiation in transparent medium			X
Radiation in semi transparent medium	X	X	X

# Solvers for linear systems (2/3)

---

- **TRUST solvers** (default choice):
  - Symmetric matrix (e.g. pressure solver)
    - GCP (Conjugate gradient with SSOR preconditioning)
  - Non symmetric matrix (e.g. implicit solver)
    - GMRES or Bi-CGSTAB with diagonal preconditioning
- Integrated **PETSc solvers** (advanced choice):
  - Symmetric matrix
    - Parallelized Cholesky:

If enough memory available AND matrix is constant, generally the best choice under 500 cores (max 1000 cores on 10e6 cells, ~2s/timestep to solve  $Ax=B$ )
    - BI-CGSTAB with block jacobi ICC(1):

The fastest if high scalability needed (>1000 cores), use GCP with block jacobi ICC(1) if BI-CGSTAB diverges. RCM ordering of the local matrix may accelerate also.
  - Non symmetric matrix
    - GMRES or BI-CGSTAB with diagonal preconditioning:

Generally faster than previous TRUST versions

# Solvers for linear systems (3/3)

---

- Iterative solvers (GC, GMRES,...)
  - Need a tolerance  $\varepsilon$  to be defined :  $\|Ax-B\| < \varepsilon$
  - Possible pitfall because it is an absolute (not a relative) value in TRUST
  - So, check the balances!
    - Exemple: Solving pressure system for an incompressible flow  $\Leftrightarrow \text{Div}(u)=0$ 
      - So, check the flow rate error in .out file
- Direct solvers (PETSc Cholesky)
  - Use it if possible

# Models, schemes, numerical methods (6/6)

- Discretizations (VDF/VEF)
- Time and space schemes
- Boundaries conditions
- Source terms
- Solvers for linear systems
- **Turbulence models**

# Turbulence models (1/1)

---

- DNS (Direct numerical simulation)
  - No model
- RANS (Reynolds averaged Navier Stokes equations)
  - 0 equation
    - Mixing length
  - 2 equations
    - Standard k- $\epsilon$
- LES (Large eddy simulation)
  - Wale [http://www.cfd-online.com/Wiki/Wall-adapting\\_local\\_eddy-viscosity\\_\(WALE\)\\_model](http://www.cfd-online.com/Wiki/Wall-adapting_local_eddy-viscosity_(WALE)_model)
  - Smagorinsky [http://www.cfd-online.com/Wiki/Smagorinsky-Lilly\\_model](http://www.cfd-online.com/Wiki/Smagorinsky-Lilly_model)
- Wall laws
  - Standard (logarithmic law)
  - TBLE (Turbulent Boundary Layer Equations)



# Table of contents

---

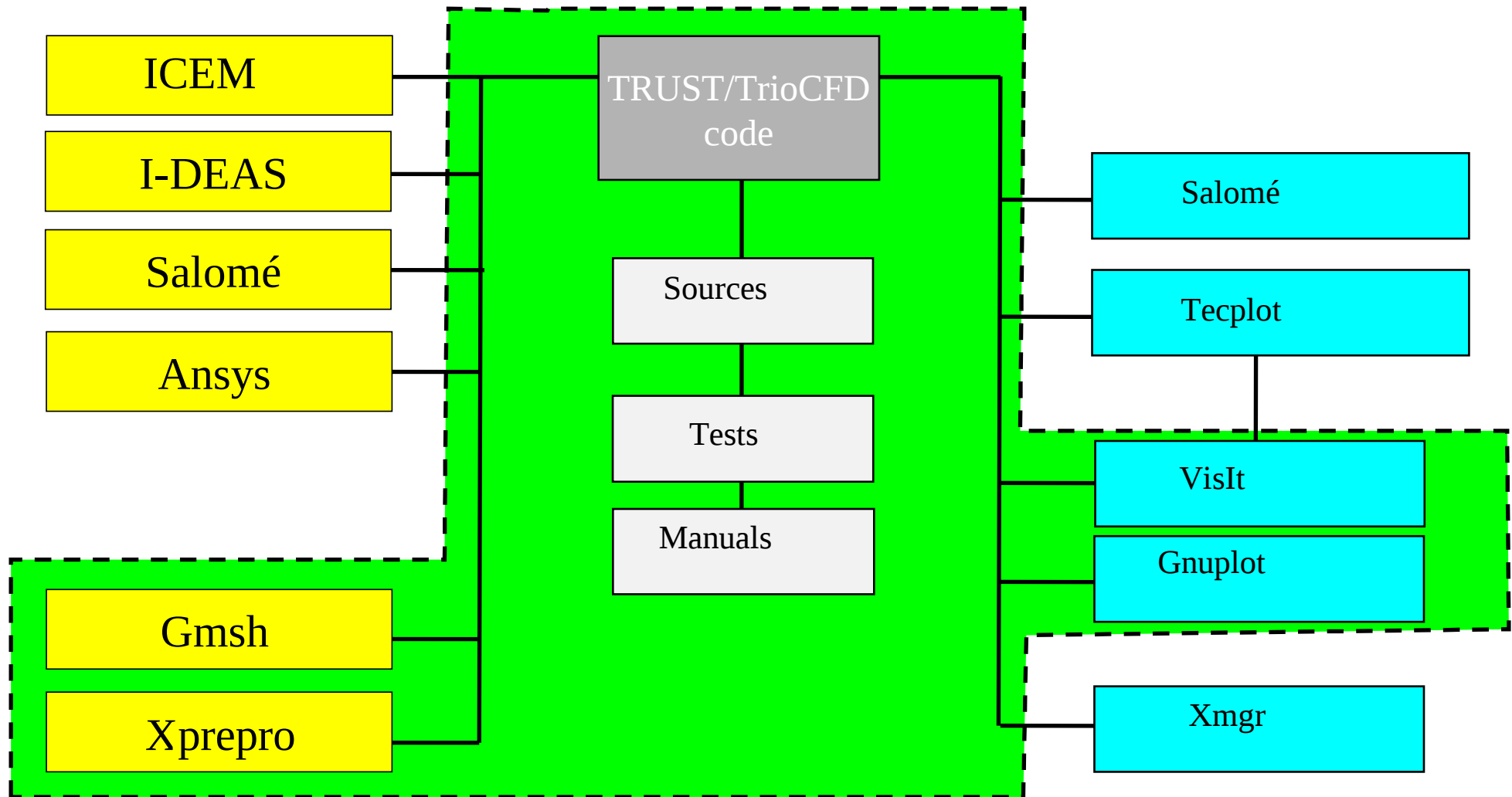
- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- **Data files & calculation**
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations

# Data files & calculation (1/5)

---

- **TRUST and tools interfaces.....p50**
- Data file description.....p52
- Operations on meshes.....p62
- Post processing description.....p64
- Output files description.....p73

# TRUST and interfaces between tools (1/1)



TRUST/TrioCFD package released and supported

# Data files & calculation (2/5)

---

- TRUST and tools interfaces
- **Data file description**
- Operations on meshes
- Post processing description
- Output files description

# Practice

---

Exercise: Obstacle.data + VisIt

(incompressible 2D flow)

# Data file description (1/8)

---

- Objects creation with keywords (Domain, Time scheme, Problem, Fluid, ...) :

# Data file objects definition #

**Domaine** my\_domain

**Schema\_Euler\_Explicite** my\_scheme

**Pb\_Thermohydraulique** my\_problem

...

## Data file description (2/8)

---

- Action on these objects with keywords:

```
Read_file my_domain meshing.geom # Read a mesh file #  
Read_file file.geo ;                # Read external instructions #  
Read my_scheme {  
    tinit 0.  
    dt_min 0.001  
    dt_max 0.002  
    dt_impr 0.001  
    .... }  
Associate my_problem my_scheme      # Association #  
Read my_problem { .... }            # Read (define) the problem #  
Solve pb                             # The problem is solved with #  
End                                   # Not necessary keyword to finish #
```

# Data file description (3/8)

## Sequential data file example

```
# Hydraulique 2D laminar with Quick scheme #  
# Dimension 2D or 3D #  
dimension 2
```

```
# Domain definition #  
Domaine dom
```

```
# BEGIN MESH #  
/* Read mesh from ICEM */  
/* Read_File_Binary dom mesh.bin */  
/* Read mesh from Salome */  
/* Read_MED dom Mesh_1 mesh.med */  
/* Create domain and mesh from TRUST */  
Read_file Obstacle.geo ;  
# END MESH #
```

```
# Discretization on hexa or tetra mesh #  
VDF ma_discretisation
```

```
# Time scheme explicit or implicit #  
Scheme_euler_explicit mon_schema  
Read mon_schema  
{
```

```
    # Time step #  
    # Initial time [s] #  
    tinit 0  
    # Min time step #  
    dt_min 5.e-3
```

```
# Max time step #  
dt_max 5.e-3 # dt_min=dt_max so dt imposed #  
    # facsec such as dt = facsec * min(dt(CFL),dt_max) ; #  
    # for explicit scheme facsec <= 1. By default facsec equals to 1 #  
    # facsec 0.5 #  
    # make the diffusion term in NS equation implicit : disable(0) or enable(1) #  
diffusion_implicit 0
```

```
# Output criteria #  
    # .out files printing period #  
dt_impr 5.e-3 # Note: small value to print at each time step #  
    # .sauv files printing period #  
dt_sauv 100  
periode_sauvegarde_securite_en_heures 23
```

```
# Stop if one of the following criteria is checked: #  
    # End time [s] #  
tmax 5.0  
    # Max number of time steps #  
    # nb_pas_dt_max 3 #  
    # Convergence threshold (see .dt_ev file) #  
seuil_statio 1.e-8
```

```
}
```



# Data file description (4/8)

*# Problem definition #*

**Pb\_hydraulique** pb

*# Physical characteristics of medium #*

**Fluide\_Incompressible** milieu

**Read** milieu

```
{
    # hydraulic problem #
    # Dynamic viscosity [kg/m/s] #
    mu Champ_Uniforme 1 3.7e-05
    # Volumic mass [kg/m3] #
    rho Champ_Uniforme 1 2
}
```

*# Association between the different objects #*

**Associate** pb dom

**Associate** pb mon\_schema

**Associate** pb milieu

**Discretize** pb ma\_discretisation

*# New domains for post-treatment #*

*# By default each boundary condition of the domain is  
already extracted with such name  
"dom"\_boundaries\_"BC" #*

*# Problem description #*

**Read** pb

{

*# hydraulic problem #*

**Navier\_Stokes\_standard**

{

*# Pressure matrix solved with #*

```
solveur_pression GCP {
    preconditioner { omega 1.500000 }
    seuil 1.000000e-06
    impr
}
```

$$\Delta P = \frac{\rho}{\Delta t} \nabla \cdot \vec{u}^t$$

*# Two operators are defined #*

```
convection { quick }
diffusion { }
```

$$\vec{u}^t = \vec{u}^n + \Delta t \left( -\nabla \cdot (\vec{u} \otimes \vec{u}) + \nabla \cdot (\nu \nabla \vec{u}) \right)$$

*# Uniform initial condition for velocity #*

```
initial_conditions {
    vitesse Champ_Uniforme 2 0. 0.
}
```

*# Boundary conditions #*

**boundary\_conditions** {

```
    Square    paroi_fixe
    Upper      symetrie
    Lower      symetrie
    Outlet     frontiere_ouverte_pression_imposee Champ_front_Uniforme 1 0.
    Inlet      frontiere_ouverte_vitesse_imposee Champ_front_Uniforme 2 1. 0.
```

}

}

...

# Data file description (5/8)

---

## List of possible keywords to define a field:

- Volume fields, keyword **Champ\_TYPE** where TYPE may be:
  - uniforme** (uniform field)
  - uniforme\_par\_morceaux** (uniform field per sub-zone)
  - fonc\_t** (uniform time dependent field)
  - fonc\_xyz** (space dependent field)
  - fonc\_txyz** (space and time dependent field)
  - fonc\_fonction** (depends on another field, analytic function)
  - fonc\_tabule** (depends on another field, tabulated function)
  - fonc\_MED** (read a MED field)
  - don\_lu** (field read in a file)
- Surface fields, keyword **Champ\_front\_TYPE** where TYPE:
  - As volume fields plus:
  - lu** (field read in a file)
  - recyclage** (field extracted from a plane or a boundary of another problem)
  - ...

# Data file description (6/8)

Formulas for a field in a data file:

**Champ\_front\_fonc\_txyz** 2 **cos**(y+x^2) t+ln(y)

**Champ\_fonc\_xyz** domain\_name 2 **tanh**(4\*y)\*(0.95+0.1\*rnd(1)) 0.

The variables which can be used are:

x,y,z : coordinates  
t : time

Constant or mathematical functions available:

PI, ABS, COS, SIN, TAN, ATAN, EXP, LN, SQRT, INT, ERF, RND(x), COSH, SINH, TANH  
NOT(x), \_AND\_, \_OR\_, \_GT\_, \_GE\_, \_LT\_, \_LE\_

You can also use the following operations:

+ : addition  
- : subtracte  
/ : division  
\* : multiplication  
% : modulo  
\$ : max  
^ : power  
< : lesser than  
> : greater than  
[ : less or equal to  
] : greater of equal to  
( ) : test if

# Data file description (7/8)

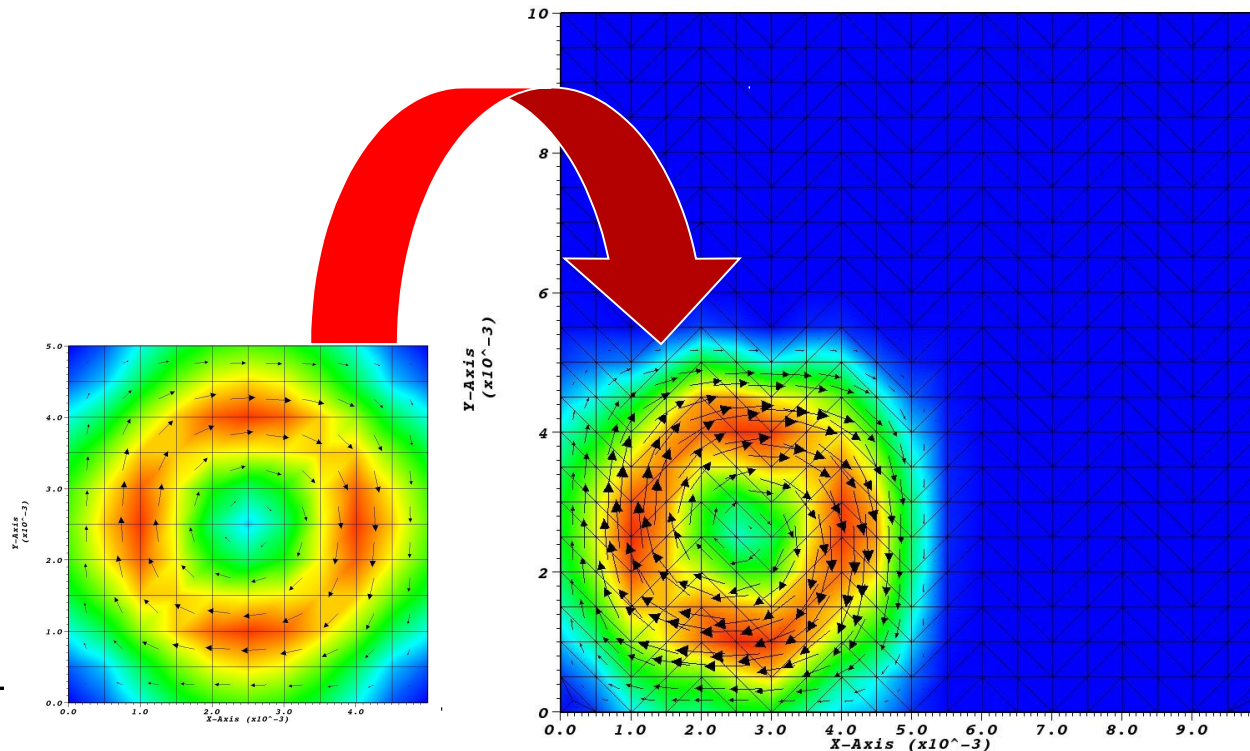
## Example of Champ\_Fonc\_MED

First calculation on a VDF mesh:

```
Postraitement { fichier VDF_field format med Champs dt_post 0.1 { vitesse elem } }
```

Second calculation on a different refined VEF mesh with initial condition from the VDF field:

```
conditions_initiales { vitesse Champ_Fonc_MED last_time VDF_field.med domain vitesse elem 0 }
```

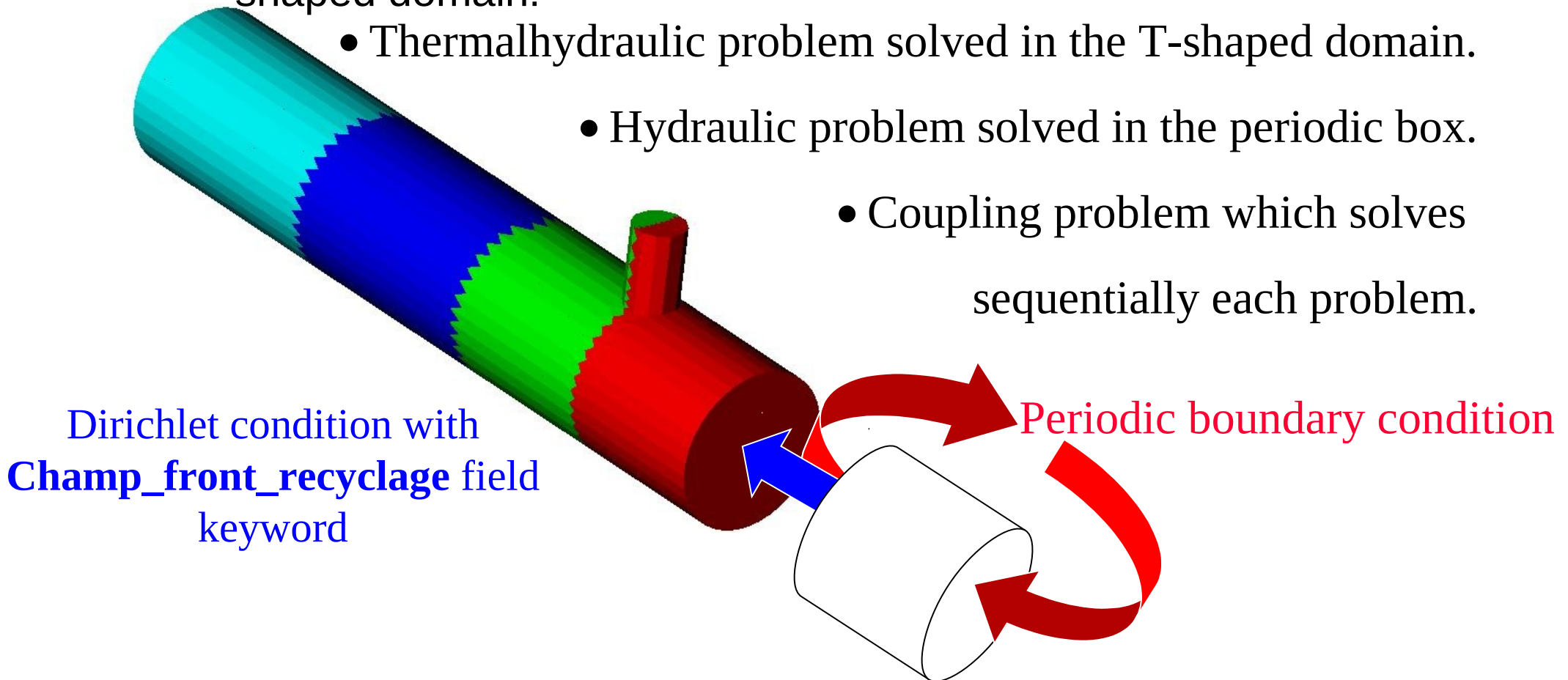


# Data file description (8/8)

## Example of TrioCFD coupled problems

Periodic box to provide a fully developed turbulent flow inlet for the T-shaped domain:

- Thermalhydraulic problem solved in the T-shaped domain.
- Hydraulic problem solved in the periodic box.
- Coupling problem which solves sequentially each problem.



# Data files & calculation (3/5)

---

- TRUST and tools interfaces
- Data file description
- **Operations on meshes**
- Post processing description
- Output files description

# Operations on meshes (2/2)

---

- Keywords exist to modify your mesh after reading it (\*.med, \*.bin, \*.geo, ...)
- List of possible keywords to adjust a mesh:
  - **Dilate** (to change the size of a mesh)
  - **Mesh** (to mesh a block or merge several meshes)
  - **Transformer** (to transform a mesh with a function)
  - **Rotation** (to rotate a mesh according to an axis)
  - **Extruder** (to extrude a 2D mesh into a 3D mesh)
  - **Trianguler/Tetraedriser** (to triangulate, to tetraedrise)
  - **Raffiner\_(an)isotrope** (to refine a mesh)
  - **RegroupeBord** (to merge or rename boundaries)
  - **Supprime\_Bord** (to suppress boundaries)
  - **Remove\_Elem** (to create holes in a VDF mesh)
  - ...

# Data files & calculation (4/5)

---

- TRUST and tools interfaces
- Data file description
- Operations on meshes
- **Post processing description**
- Output files description



# Data file description (1/8)

```
...
# Post_processing description #
/* To know domains that can be treated directly, search in .err output file: "Creating a surface domain named" */
/* To know fields that can be treated directly, search in .err output file: "Reading of fields to be postprocessed" */
Post_processing
{
    # Probes #
    Probes
    {
        # Note: periode with small value to print at each time step (necessary for spectral analysis) #
        sonde_pression      pression      periode 0.005 points 2      0.13 0.105      0.13 0.115
        sonde_vitesse       vitesse       periode 0.005 points 2      0.14 0.105      0.14 0.115
        sonde_vit      nodes vitesse      periode 0.005 segment 22 0.14 0.0      0.14 0.22
        sonde_P      pression      periode 0.01 plan 23 11 0.01 0.005      0.91 0.005      0.01 0.21
        sonde_Pmoy      Moyenne_pression      periode 0.005 points 2      0.13 0.105      0.13 0.115
        sonde_Pect      Ecart_type_pression      periode 0.005 points 2      0.13 0.105      0.13 0.115
    }

    # Fields #
    format lml # lata for VisIt tool #
    fields dt_post 1. # Note: Warning to memory space if dt_post too small #
    {
        pression elem
        pression som
        vitesse elem
        vitesse som
    }
}
```

# Data file description (2/8)

```
# Statistical fields #
Statistiques dt_post 1.
{
    t_deb 1. t_fin 5.
    moyenne vitesse
    ecart_type vitesse
    moyenne pression
    ecart_type pression
}

# Saving and restarting process #
/* sauvegarde_simple binaire datafile.sauv */ # Note: last time step only saved #
/* resume_last_time binaire datafile.sauv */
}

# The problem is solved with #
Solve pb
# Not necessary keyword to finish #
End
```

# Post processing description (3/8)

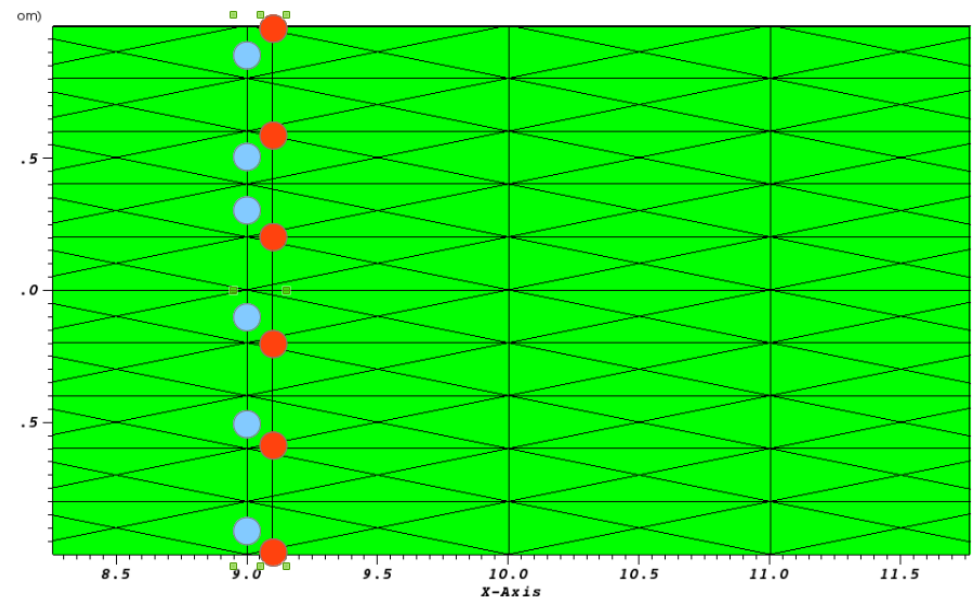
## Probes : "Nodes" option

"Nodes" option moves the probes to the nearest faces, but take care of stretched meshes!

```
sonde_vit nodes vitesse periode 0.005 segment 22 0.14 0.0 0.14 0.22
```

In the following 2D example, the initial probes in **red** are defined along a segment from boundary to boundary, but when applying option "**nodes**" it moves the probes (in **blue**) to the nearest face and sometimes near the boundary the nearest face **IS NOT** a boundary face.

Since several version it is possible to visualize the .son files containing the probes in VisIt. It provides you a point MESH which is the localization of the probes. We will try in the 1.6.9 version to help user by improving the messages in the .err file when probes are moved according to the "nodes" option and also to run VisIt on .son files directly from the triou script.



So, if you want extreme probes of the segment on the boundaries, try to move slightly the segment.

# Post processing description (4/8)

## TRUST results export

2D/3D results files are readable:

- Either directly by :
  - VisIt (use *lata* format in the data file “**format lata**”)\*
  - Salomé (use *med* format in the data file “**format med**”):
    - to post-process with Salomé: use ParaVis module → New study → File → Open Paraview File,
    - to open a mesh in Salomé: use Mesh module → New study → File → Import → MED File
- Or:
  - Read with VisIt and then export (VTK format, field by field): Paraview
  - Using lata2dx script (lata\_to\_med, lata\_to\_case): for example Tecplot

1D results files by :

- Gnuplot, XmGrace, Excel

# Post processing description (5/8)

- Possible basic post processed fields

- Equation unknowns (velocity, pressure, temperature,...)
- Physical characteristics (dynamic viscosity, thermal conductivity,...)
- Model fields (turbulent viscosity, friction velocity,...)

The complete list of keywords for fields which could be post processed for the current calculation is printed into the .err file :

**Reading of fields to be postprocessed**

Milieu\_base : 1 **masse\_volumique**

Fluide\_Incompressible : 2 **viscosite\_cinematique viscosite\_dynamique**

Equation\_base : 1 **volume\_maille**

Operateur\_base : 0

Navier\_Stokes\_std : 16 **divergence\_U gradient\_pressionY gradient\_pressionX gradient\_pression pression\_pa pression\_vitesseY vitesseX vitesse taux\_cisaillement courant\_maille reynolds\_maille y\_plus porosite\_volumique critere\_Q vorticite**

- Possible advanced post processing fields

- Can be created in the data file with the keyword **Definition\_champs**
- 3 examples given:
  - Monitoring extreme values of a field
  - Averaging a field on a boundary
  - Error between Trio\_U and an analytical solution

# Post processing description (6/8)

## Maximal value of a field

```
Read pb {  
  ....  
  Post_processing {  
    Definition_champs {  
      # Creation of the 0D field: maximal temperature of the domain #  
      temperature_max Reduction_0D {  
        methode max  
        source refChamp { Pb_champ pb temperature }  
      }  
    }  
    Probes {  
      # Print max(temperature) into the datafile_TMAX.son file #  
      tmax temperature_max periode 0.01 point 1 0. 0. 0.  
    }  
    Fields dt_post 1.0 { ... }  
  }  
}
```

# Post processing description (7/8)

## Averaging a field on a surface

**Dimension** 3

**Domaine** dom # 3D domain with a boundary named wall #

...

**Domaine** surface # 2D domain object created for use by the Extraction keyword or use **DomainName\_boundaries\_BoundaryName** in the new field #

**Read** problem { ...

**Post\_processing** {

**Definition\_champs** {

*# Creation of the 0D field: mean temperature on the boundary wall #*

        wall\_mean\_temperature **Reduction\_0D** {

**methode** moyenne

**source** Extraction {

**domaine** surface **nom\_frontiere** wall **methode** trace

**source** refChamp { **Pb\_champ** problem **temperature** }

            }

        }

    }

*# Print into the datafile\_TWALL.son file #*

**Probes** { twall wall\_mean\_temperature **periode** 0.01 **point** 1 0. 0. 0. }

**Fields** dt\_post 1.0 { ... }



# Post processing description (8/8)

## Calculating an error between fields

```
Post_processing {
  Definition_champs { # Creation of the 3D field: error #
    error Transformation {
      methode formule expression 1 vit-sol
      localisation elem
      sources {
        refChamp { Pb_champ problem vitesse nom_source vit } ,
        Transformation { methode vecteur expression 3 x*y x+y nom_source sol }
      }
    }
    # Calculate the L2 norm of the error as 0D field #
    error_norm Reduction_0D { methode norme_L2 source_reference error }
    } # Print into the datafile_ERROR_NORM.son file #
    Probes { sonde_error_norm error_norm periode 0.01 point 1 0. 0. 0. }
    format LATA Fields dt_post 1.0 {
      error # Post process the error field #
      velocity
    }
  }
}
```



# Data files & calculation (5/5)

---

- TRUST and tools interfaces
- Data file description
- Operations on meshes
- Post processing description
- **Output files description**

# Output files description (1/6)

## Saving process

- Unknowns (velocity, temperature,...) are saved in:
  - one .xyz file
  - one or several (parallel calculation) .sauv files
- By default, saving process in .sauv happens during the calculation:
  - At the start and at the end
  - Periodically (each 23 hours of CPU with **tcpumax** keyword)
  - But, user may also specify a time physical period (**dt\_sauv** keyword)
- By default, saving process in .xyz file happens during the calculation:
  - At the end
  - But, user may enable it with the specific keyword "**EcritureLectureSpecial 0**"
- By default, there is a default name for the .sauv files (testcase\_000n.sauv), the format is binary, and the files are appended during successive saves but user can change the behaviour with the keywords:

**sauvegarde|sauvegarde\_simple binaire|xyz filename.sauv|filename.xyz**

with:

- **sauvegarde\_simple** : the file is deleted before the save
- **xyz**: the .xyz file is written instead of the .sauv files

# Output files description (2/6)

---

## Restarting process

Restarting the calculation is possible :

- Either from *.sauv* file(s) (one file per process)
  - > Necessary to restart the calculation with the same number of equations on the same number of processes
- Or from a *.xyz* file
  - > Possible to restart a calculation by changing the number of equations solved
  - > Possible to restart with a different number of processes

The mandatory syntax in the data file is:

**reprise binaire|xyz** *filename.sauv|filename.xyz*

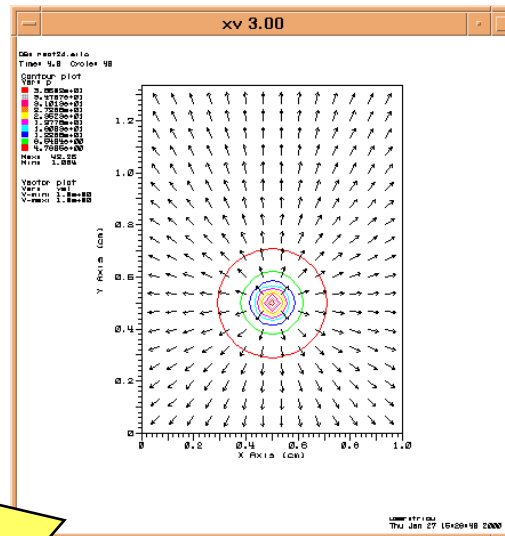
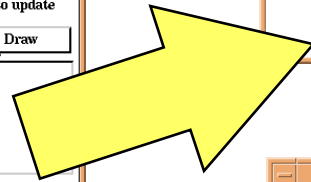
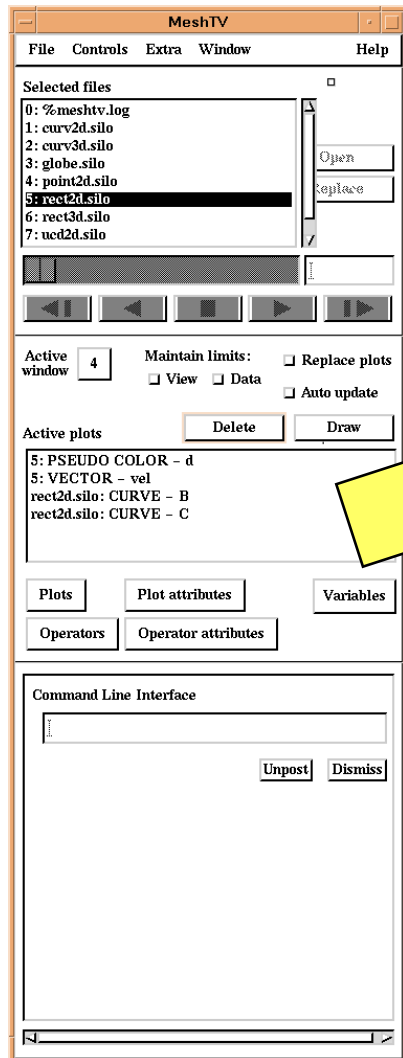
# Output files description (3/6)

## TRUST files summary

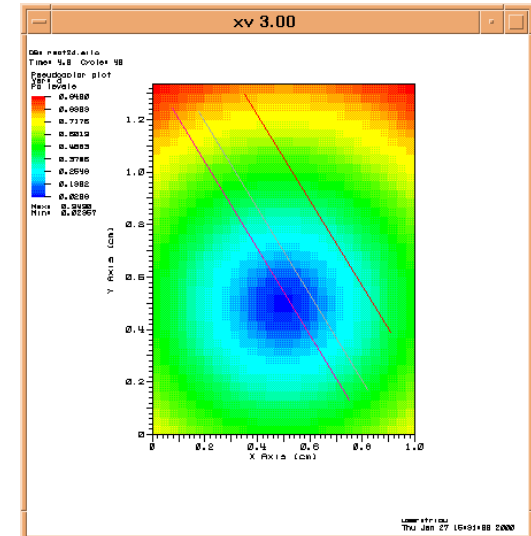
- Input:
  - Data file: .data
  - Meshing: .geom (or .bin)
  - Instructions file: .geo
  - Sub zones: .ssz
  - Sub domains: .Zones
- Output :
  - 2D/3D results: .lata (or .med)
  - 1D results: .son
  - Saving-restart: .sauv ou .xyz
  - Listing (physical infos): .out
  - Listing (warnings&errors&domain infos): .err
  - Listing of boundary fluxes: \*.out
  - Canal\_perio outputs: \*"BCname"
  - CPU performances: .TU
  - Time steps, facsec, equation residuals: .dt\_ev
  - Stop file (0 or 1): .stop

# Output files description (4/6)

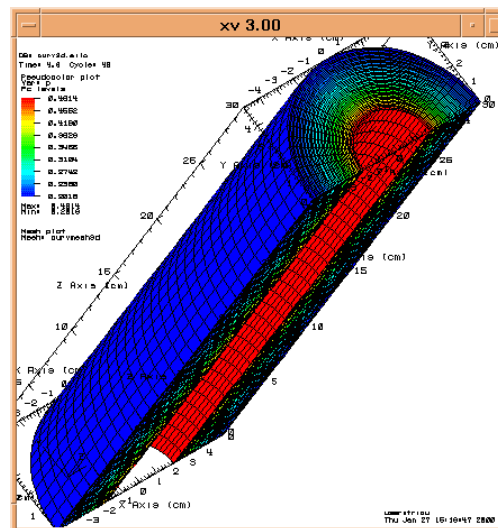
## TRUST with VisIt



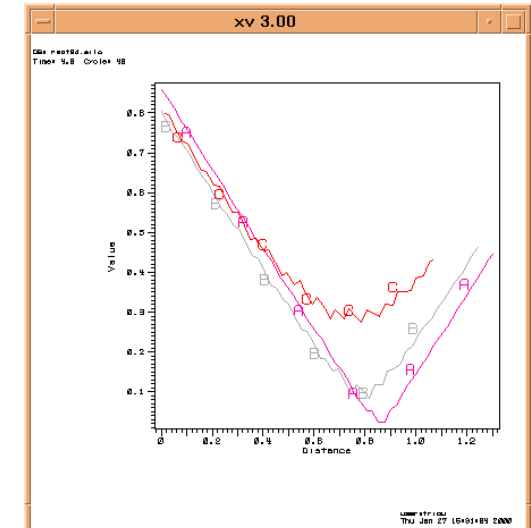
Vectors/Scalars



2D Slice



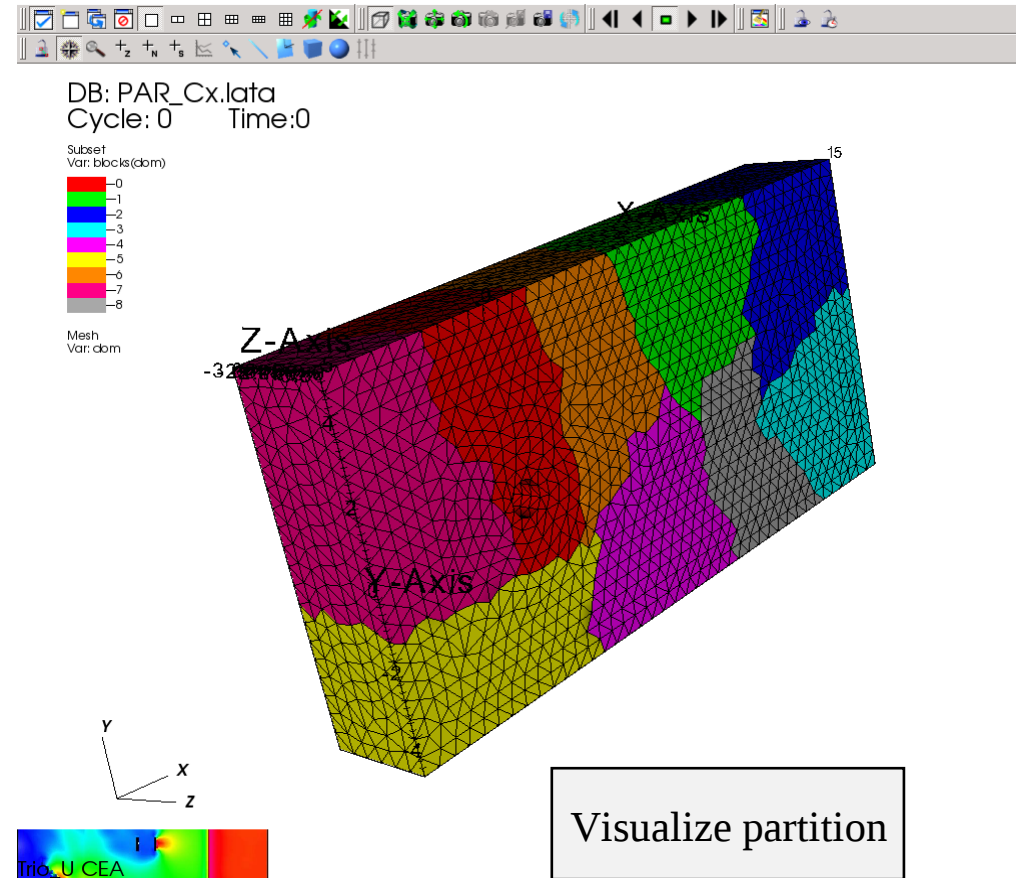
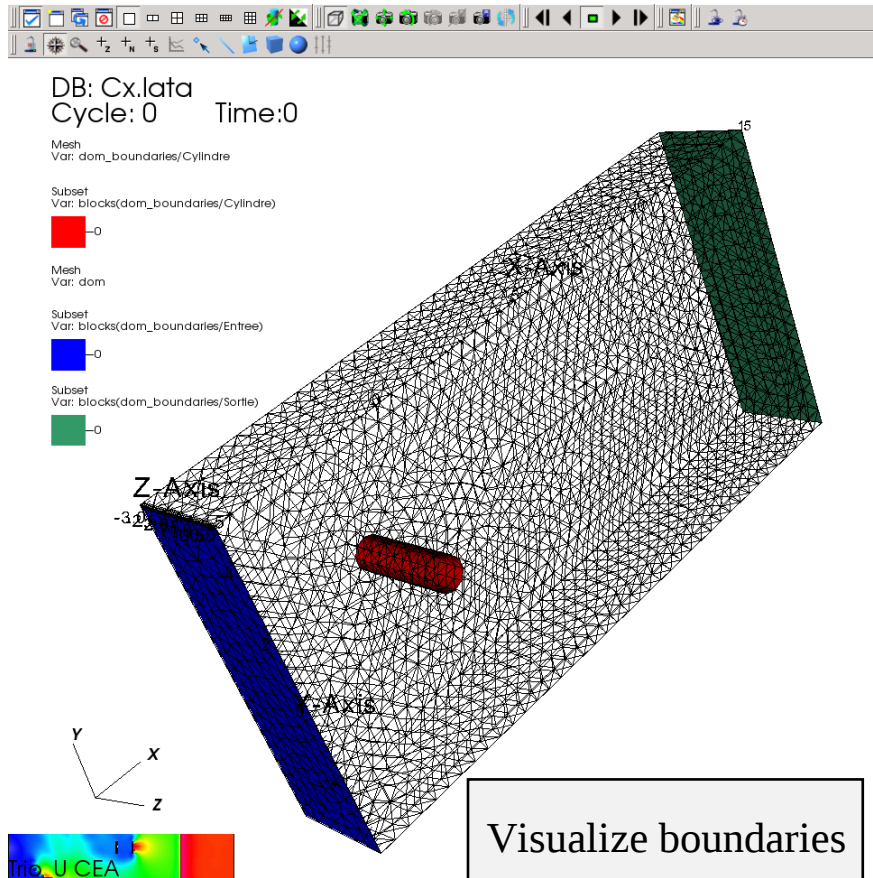
3D View



1D Cut

# Output files description (5/6)

## TRUST with VisIt



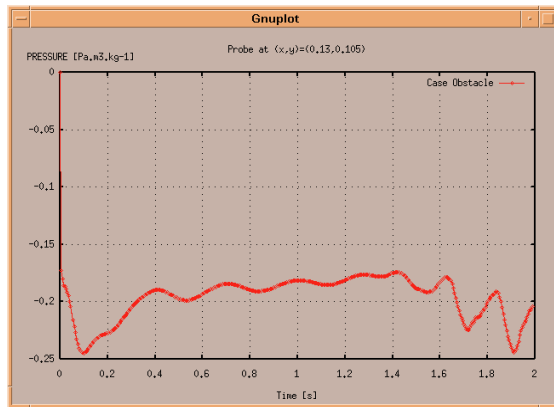
For more informations and to download manuals see :

<https://wci.llnl.gov/codes/visit/manuals.html>

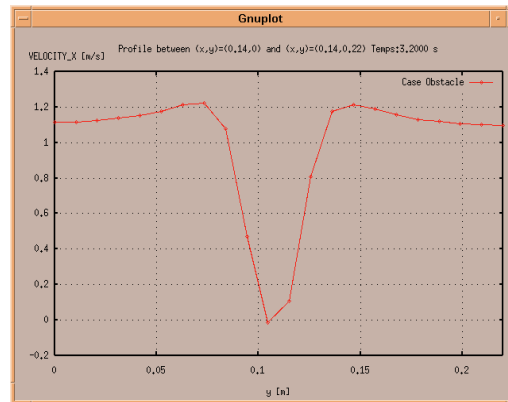
# Output files description (6/6)

## TRUST with gnuplot

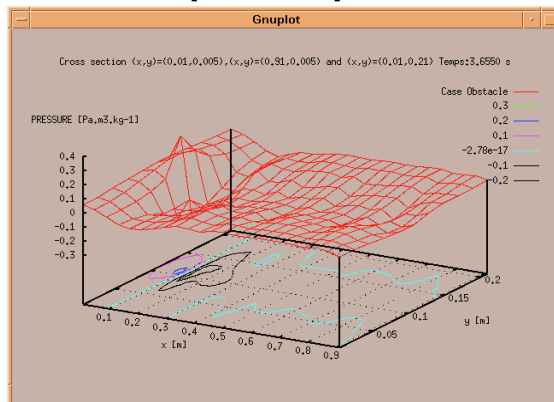
Real-time display of calculated quantities:



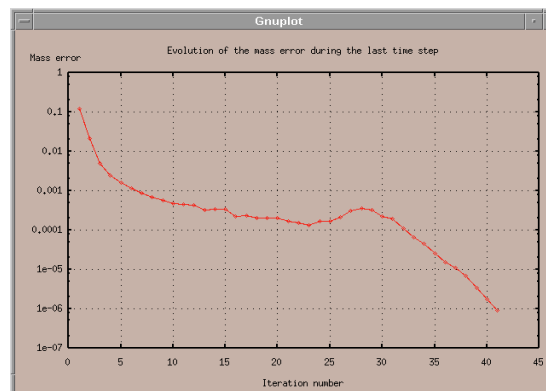
One point probe



Probes segment



Probes plane



Convergence

Instantaneous or averaged value and also, flux balance at the boundaries like:

- flow rate
- pressure forces
- viscous forces
- heat flux
- ...

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- **Command lines**
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations



# Command lines (1/3)

---

- TRUST environment initialization:  
**source \$TRUST\_ROOT/env\_TRUST.sh**
- To run a TRUST calculation with the trust script:
  - Sequential run:  
***trust datafile***
  - Parallel run on N CPUs:
    - Partitioned mesh partitioned should be created sequentially, then interactively:  
***trust datafile N***
    - Or to run on a batch-queuing system, add this line into the submission file :  
***mpirun -np N \$exec datafile N***
- To redirect into output and error files, after the command line, add:  
***trust .... 1>datafile.out 2>datafile.err***

# Command lines (2/3)

---

- To copy a data file from the test database:  
**trust -copy** datafile
- To visualize the mesh and its boundaries used by a data file:  
**trust -mesh** datafile[.data]
- To edit interactively (change/add schemes, solvers...) a data file with:  
**EditData** datafile.data
- Check a data file without running TRUST:  
**VerifData** datafile.data
- To identify all the data sets from the no regression data base which contain some specified keywords (word1 word2...wordn). The identified data set are listed in the file 'liste\_cherche'.  
**cherche.ksh** [-reference\_only] word1 word2 ...wordn
- To monitor only your calculation:  
**trust -probes** [datafile[.data]]

# Command lines (3/3)

---

- To run VisIt with a LATA results file:  
**visit -o datafile.lata &**
- To clean your calcul directory:  
**trust -clean**
- To open the PDF documentation (User's manual):  
**trust -doc**
- To browse some useful resources (PDF manuals, test cases, keywords, C++ classes,...) :  
**trust -index**
- In all cases, the TRUST binary may be changed by the **\$exec** variable and by default, `exec=$TRUST_ROOT/exec/TRUST_mpi_opt`

# Practice

---

- > source /home/triou/env\_TRUST\_X.Y.Z.sh
- > echo \$TRUST\_ROOT
- > cd ~/test/yourname/Obstacle
- > nedit Obstacle.data &
  - Change the domain name to “truc” instead of “dom” at line 7
  - Save and close the file
- > VerifData Obstacle.data
  - ERROR !
  - Modify it to the previous value.
- > VerifData Obstacle.data
  - OK

# Practice

---

- > EditData Obstacle.data &
  - Expand “pb”, “postraitement”
  - Change “lml” to “lata” into “format”
  - Close the window and say yes into the terminal.
- > VerifData Obstacle.data
  - OK

# Practice

---

## Exercise: Heat exchange VDF/VEF exercise

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- **Parallel calculation**
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations

# Parallel calculation (1/3)

---

- **Parallel calculation description..p88**
- Parallel calculation on clusters..... p97
- Visualization with VisIt..... p105



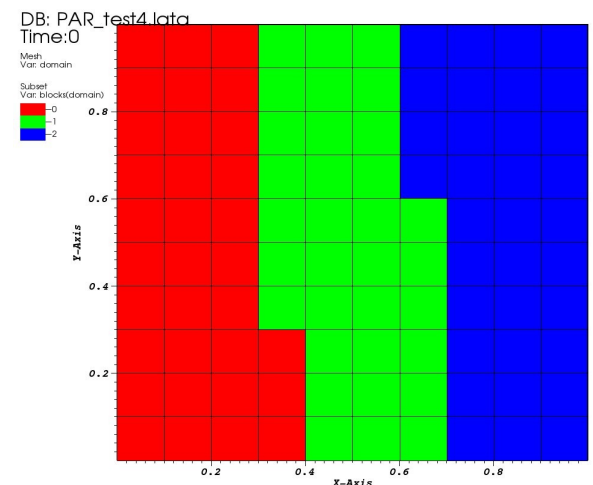
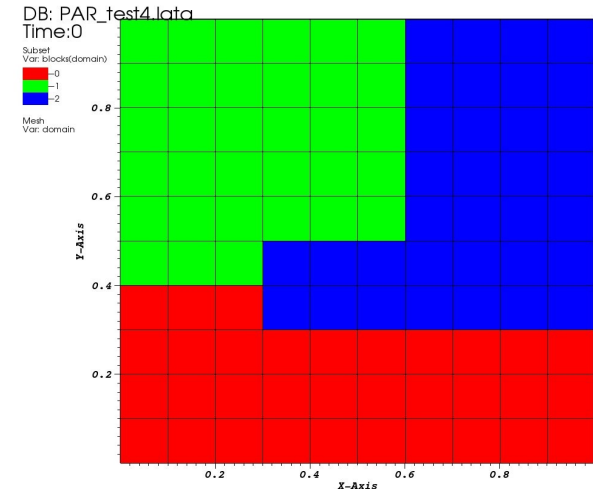
# Parallel calculation description (1/5)

---

- SPMD model (Single Program Multiple Data)
- Messages exchange by MPI (Message Passing Interface)
- From PC to massively // computer, with shared or distributed memory

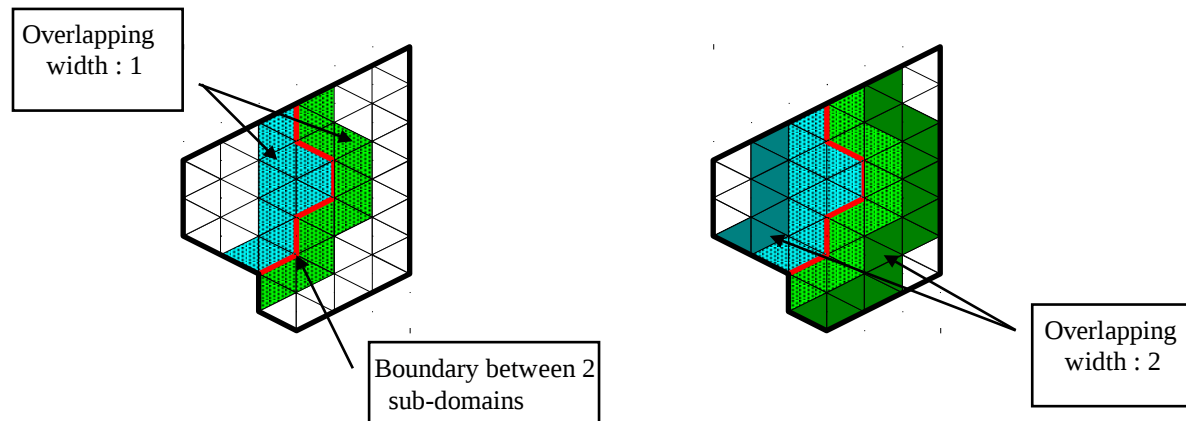
# Parallel calculation description (2/5)

- Domain partitioning tools:
  - Metis
  - Tranche “band partitioning”
- Performances are partition dependent:
  - Same number of cells by sub-domain
  - To minimize the joints length (boundaries between sub-domains)
- If possible, use 20000-30000 cells per process.



# Parallel calculation description (3/5)

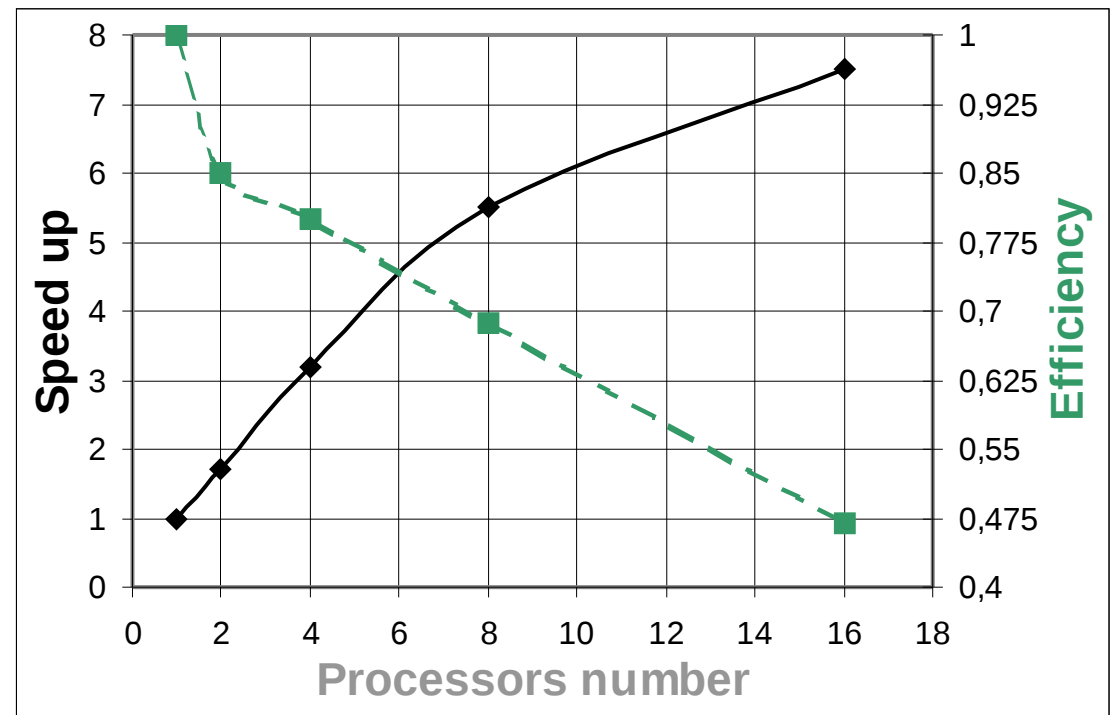
- Definition of overlapping width value
  - Number of vertexes or elements on the remote sub-domain known by the local sub-domain
  - Specified by the users during partitioning task
  - This value depends on the space scheme orders:
    - 1 if 1-2<sup>nd</sup> order
    - 2 if 3-4<sup>th</sup> order
  - In practice, use 2 except if you use **only** upwind schemes



# Parallel calculation description (4/5)

- Performance examples :
  - PC linux cluster (Scali network):

Processor number	Speed Up= $\frac{\text{seq\_time}}{\text{par\_time}}$	Efficiency= $\frac{\text{Speed\_Up}}{n \text{ b\_procs}}$
1	1	1
2	1.7	0.86
4	3.2	0.80
8	5.5	0.69
16	7.5	0.47

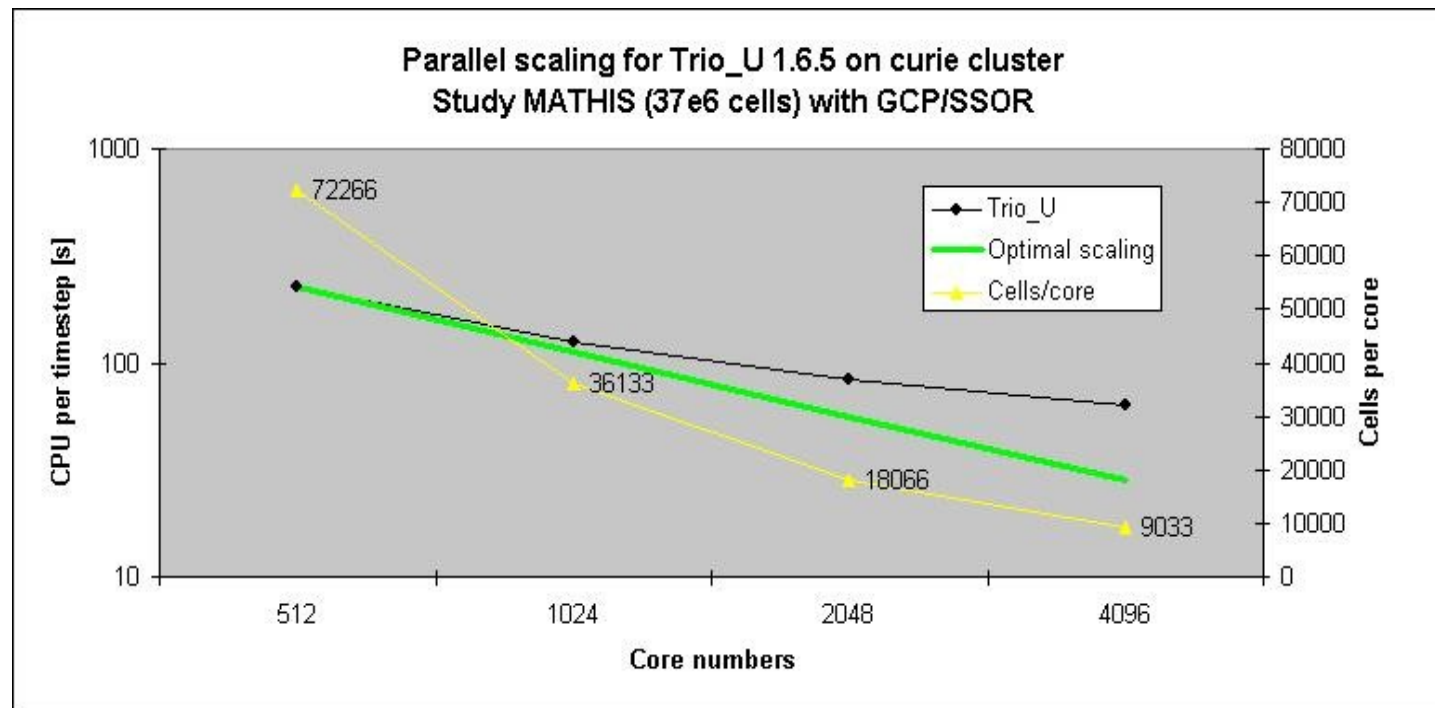


Cf datafile\*.TU files

# Parallel calculation description (5/5)

Some advices:

- Choose a number of elements per process between 20000 and 30000 for optimal performances.
- Below 20000 elements/process, TRUST parallel efficiency may dramatically decreases. Example on curie:



Scaling of curie from 72000 cells/core to 9000 cells/core

# Parallel calculation description (5/5)

---

⇒ To run a parallel calculation, you must do two runs:

- \* **the first one**, to partitioning and create your 'n' sub-domains,

- \* **the second one**, to read your 'n' sub-domains and run the calculation on 'n' processors.

# Data file description (3/8)

---

## Parallel data file example for the first run

```
dimension 2

# Domain definition #
Domaine dom

# BEGIN MESH #
Read_file Obstacle.geo ;
# END MESH #

# BEGIN PARTITION #
Partition dom
{
    /* Choose Nb_parts so to have ~ 25000 cells per processor */
    Partition_tool metis { nb_parts 2 }
    Larg_joint 2
    zones_name DOM
}
End
# END PARTITION #
```

# Data file description (3/8)

## Parallel data file example for the second run

```
dimension 2

Domaine dom

# BEGIN SCATTER #
Scatter DOM.Zones dom
# END SCATTER #

VDF ma_discretisation

Scheme_euler_explicit mon_schema
Read mon_schema { ... }

Pb_hydraulique pb

Fluide_Incompressible milieu
Read milieu { ... }

Associate pb dom
Associate pb mon_schema
Associate pb milieu
Discretize pb ma_discretisation

Read pb
{
    Navier_Stokes_standard { ... }
    Post_processing { ... }
}
Solve pb
End
```



# Parallel calculation (2/3)

---

- Parallel calculation description
- **Parallel calculation on clusters**
- Visualization with VisIt

# Parallel calculation on clusters (1/7)

---

- To connect to:
  - CEA/DM2S service cluster: **callisto** (~1 000 cores)
  - CEA/Marcoule cluster: **ceres2** (~700 cores)
  - CEA/Cadarache cluster: **mezel** (~600 cores)
  - CEA/CCRT cluster: **cobalt** (~40 000 cores)
  - CEA/TGCC cluster: **curie-ccrt** (~90 000 cores)
  - CINES cluster: **occigen** (~50 000 cores)
- Ask for a login:
  - callisto: access for CEA only
  - ceres2: access for CEA only
  - mezel: access for CEA only
  - CCRT: <http://www-ccrt.cea.fr>
  - TGCC: <http://www-hpc.cea.fr>
  - occigen: <http://www.cines.fr>
- Once you have the login, connect from your PC to:
  - `ssh -X login@callisto-login1` or `ssh -X login@callisto-login2`
  - `ssh -X login@ceres2`
  - `ssh -X login@mezel`
  - `ssh -X login@name.ccc.cea.fr` (name=curie-ccrt | cobalt )
  - `ssh -X login@occigen.cines.fr`

# Parallel calculation on clusters (2/7)

- **TRUST/TrioCFD versions located:**
  - At CEA, on PCs or on callisto:  
by “sourcing”:
    - `source /home/triou/env_TRUST_X.Y.Z.sh`
    - `source /home/triou/env_TrioCFD_X.Y.Z.sh`
  - On clusters :
    - mezel ROOT=/soft/mezel/TRIO
    - ceres2 ROOT=/softs/trio\_u
    - cobalt & curie-ccrt ROOT=/ccc/cont002/home/den/triou
    - occigen ROOT=/opt/software/applications/trio\_u
- by « sourcing »:
  - `source $ROOT/env_TRUST-X.Y.Z.sh`
  - `source $ROOT/env_TrioCFD-X.Y.Z.sh`
- You can add this command line or create **alias** in your `~/.profile` or `~/.bashrc` file:  
`alias TRUST_XYZ='source /home/triou/env_TRUST_X.Y.Z.sh'`
- **Check your environment**, after you reconnect to the cluster, look at TRUST\_ROOT variable:  
`echo $TRUST_ROOT`  
`echo $exec`

# Parallel calculation on clusters (3/7)

---

- First sequential run **to partition a mesh**: interactive run (but very time-limited)  
***trust datafile***
- Second parallel interactive run: (but very time-limited), for example to check your datafile:  
***trust datafile nb\_processes***
- **To use the batch queuing system** (for a long time-limited run), you need to create first a submission file (named sub\_file):  
***trust -create\_sub\_file datafile nb\_processes***
- Then, you submit the job:  
for callisto, mezel, ceres2 & occigen: **sbatch sub\_file**  
for cobalt & curie-ccrt: **ccc\_msub sub\_file**

# Parallel calculation on clusters (4/7)

---

- Don't forget to read the cluster documentation before running your jobs, to be aware of its particularities.
- Before you submit the job, you can edit and change the values of the submission file **sub\_file**
- The submission file describes:
  - The job name
  - The number of cores required
  - The default output files
  - The CPU time required (the CPU value selects implicitly a queue)
  - The location of the TRUST study
  - The TRUST parallel command line

# Parallel calculation on clusters (5/7)

- Example on callisto

```
#SBATCH -J name_of_the_job
#SBATCH -p slim
#SBATCH --qos=normal
#SBATCH -t 2880
#SBATCH -o myjob.%J.o
#SBATCH -e myjob.%J.e
#SBATCH -n 2
cd $SLURM_BRIDGE_DIR
srun -n $SLURM_NTASKS $exec datafile $SLURM_NTASKS 1>jdd.out 2>jdd.err
```

- Example on CCRT/TGCC

```
#MSUB -r job_name
#MSUB -q standard
#MSUB -Q normal
#MSUB -T 86400
#MSUB -o myjob.%J.o
#MSUB -e myjob.%J.e
#MSUB -E "--no-requeue"
#MSUB -n nb_procs
# On cluster curie, add also your project (e.g. Genden):
#MSUB -A genden
cd $BRIDGE_MSUB_PWD
ccc_mprun -n $BRIDGE_MSUB_NPROC $exec datafile $BRIDGE_MSUB_NPROC 1>datafile.out 2>datafile.err
```

- Example on occigen

```
#SBATCH -J name_of_the_job
#SBATCH -t 24:00:00
#SBATCH -o myjob.%j.o
#SBATCH -e myjob.%j.e
#SBATCH --constraint=BDW28
#SBATCH --exclusive
#SBATCH -n 2
#SBATCH -N 1
cd $SLURM_SUBMIT_DIR
srun --mpi=pmi2 -K1 --resv-ports -n $SLURM_NTASKS $exec datafile $SLURM_NTASKS 1>jdd.out 2>jdd.err
```

# Parallel calculation on clusters (6/7)

---

- Description of partitions for each cluster:
  - callisto, mezel, ceres2 & occigen: ***sinfo***
  - cobalt & curie-ccrt: ***ccc\_mpinfo***
- Description of queues for each cluster:
  - callisto, mezel, ceres2 & occigen: ***sacctmgr list qos***
  - cobalt & curie-ccrt: ***ccc\_mqinfo***
- List of jobs and their state:
  - cobalt & curie-ccrt: ***ccc\_mpp -u your\_login***
  - others: ***squeue -u your\_login*** or ***squeue -j job\_number***
- Kill a job (the job\_number is given by the previous command):
  - cobalt & curie-ccrt ***ccc\_mdel job\_number***
  - others: ***scancel job\_number***

# Parallel calculation on clusters (6/7)

---

- Complet informations for each cluster:
  - callisto: from cluster, “[evince /cm/shared/docs/callisto.pdf](#)”
  - mezel: see <https://www-linuxcad.intra.cea.fr/doku.php>
  - ceres2: see <https://www-linuxmar.intra.cea.fr/dokuwiki/doku.php>
  - cobalt: see <https://www-tgcc.ccc.cea.fr/docs/cobalt.info.html>
  - curie-ccrt: see <https://www-tgcc.ccc.cea.fr/docs/curie.info.html>
  - occigen: see <https://www.cines.fr/calcul/materiels/occigen>  
and <https://www.cines.fr/tt6-comment-choisir-la-partition-doccigen>
- Space disc on each cluster:
  - \$HOME limited space for source code & binaries, backup
  - \$SCRATCHDIR large space for calculation datas, outputs, no backup
  - \$STOREDIR space for data archiving, file's number limited, backup



# Parallel calculation (3/3)

---

- Parallel calculation description
- Parallel calculation on clusters
- **Visualization with VisIt**

# Visualization with VisIt (1/2)

---

## On clusters

- If available, use a **HPCDrive** deported session on CCRT/TGCC clusters to run VisIt without network slowness
  - Available on curie cluster: <https://visu-tgcc.ccc.cea.fr/HPCDrive/home>
  - Available on cobalt cluster: <https://visu-cobalt.ccc.cea.fr/HPCDrive/home>
  - Ask us the HPCDrive user manual
- Or the client/server mode
  - See the following description VisIt (**callisto**)
  - Unhappily, this mode DOES NOT work with CCRT/TGCC clusters
  - Fine tuning of a critical option: Rendering->Advanced->Auto (2000KPolys)
- Or local mode
  - Copy the LATA results from the cluster to your PC and run the parallel version of VisIt

# Visualization with VisIt (2/2)

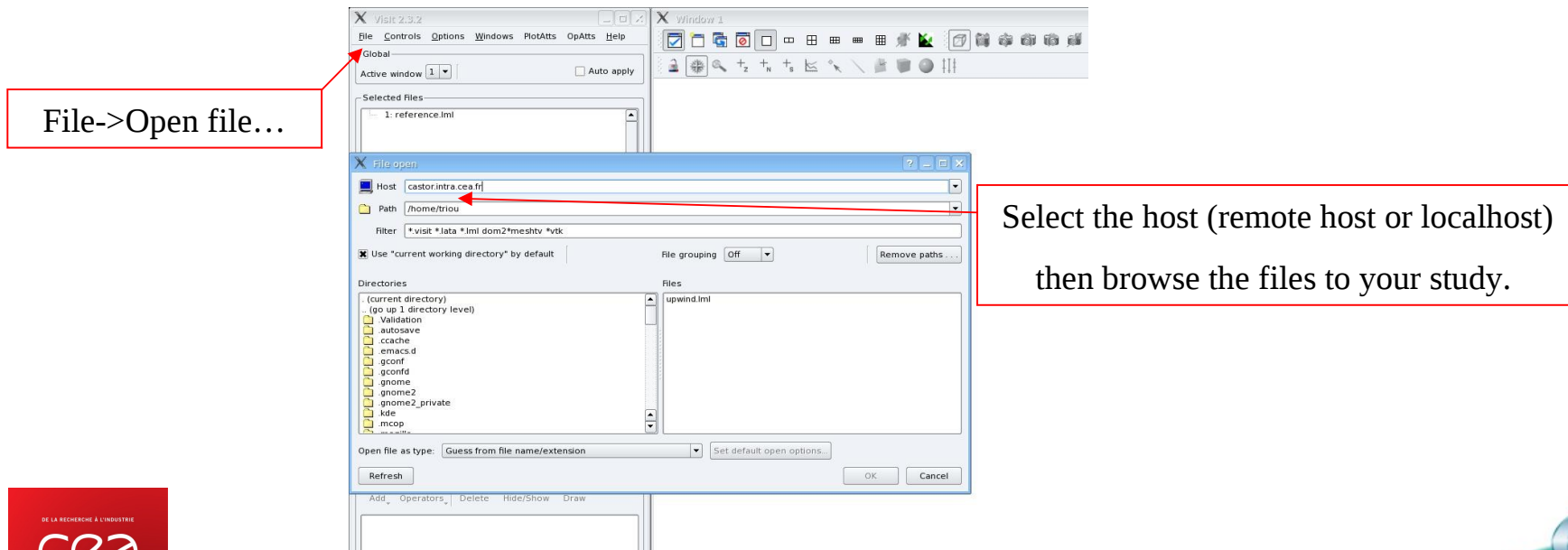
## Recent features with VisIt

-The TRUST install builds a parallel version of VisIt:

```
visit -np 8 -o results.lata
```

-Client/server mode available by default for some clusters (callisto)

- You run the TRUST calculation on the cluster
- You visualize with VisIt your results from your Linux/Windows PC without data copy and/or network slowness



# Practice

---

Exercise: Obstacle.data //

Exercise: Calculation on callisto

Exercise: Turbulent flow on a 3D step

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- **Mesh generators: Internal tools & Salomé & Gmsh**
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- Recommendations

# Mesh generators (1/3)

---

- **Possible meshing tools.....p110**
- TRUST internal mesh tool.....p114
- Salomé.....p116
- Gmsh..... p118
- Read a med file in TRUST.....p122

# Possible meshing tools (1/3)

---

- TRUST internal mesh tool
  - Used by keywords in the data file
  - Limited to simple geometry (assembling of rectangle in 2D or blocks in 3D)
- Xprepro (only VDF)
  - External tool created by the Trio\_U team
  - To create a complex geometry but with a regular hexahedral mesh
  - cf \$TRUST\_ROOT/doc/TRUST/Manuel\_Xprepro.pdf
- Or use of a mesh generator tool linked with TRUST:
  - Salomé
  - Gmsh

# Possible meshing tools (2/3)

---

- Mesh generator tools :
  - ICEM (Ansys) generates a file at Trio\_U format
  - TRUST reads .unv files from I-DEAS\*
  - TRUST reads 2D/3D meshes from old tools of Fluent (Gambit/TGrid)
  - TRUST reads « .med » meshes from Salomé or Gmsh
- Form factors (view factors for the radiation model):
  - Link between Ansys and TRUST

\*:tetrahedral meshing only



# Possible meshing tools (3/3)

- Presentation of:
  - TRUST internal mesh tool,
  - Salomé,
  - Gmsh.
- Exercise with **Salomé** or **Gmsh** according to your needs

Mesh generator	Salomé	Gmsh	Other (ICEM,...)
Availability	Free	Free	License
TRUST discretization	VEF	VEF	VEF
TRUST package	No	Yes	No
TRUST tutorial	Yes	Yes	No
Support	<a href="mailto:support-salome@cea.fr">support-salome@cea.fr</a>	<a href="mailto:gmsh@geuz.org">gmsh@geuz.org</a>	

## Mesh generators (2/3)

---

- Possible meshing tools
- **TRUST internal mesh tool**
- Salomé
- Gmsh
- Read a med file in TRUST

# TRUST internal mesh tool

**Mailler dom {**

**Pave Left**

```
{
  Origine 0. 0.
  Nombre_de_Noeuds 6 23
  Longueurs 0.1 0.22
}
{
  bord Inlet X = 0. 0. <= Y <= 0.22
  bord Upper Y = 0.22 0. <= X <= 0.1
  bord Lower Y = 0. 0. <= X <= 0.1
  bord Square X = 0.1 0.1 <= Y <= 0.12
},
```

**Pave Up**

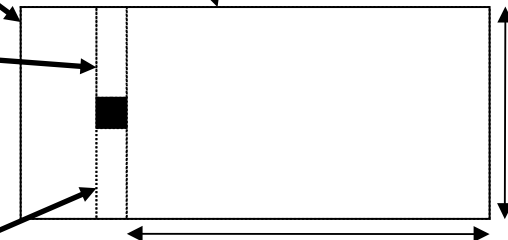
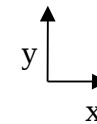
```
{
  Origine 0.1 0.12
  Nombre_de_Noeuds 2 11
  Longueurs 0.02 0.1
}
{
  bord Upper Y = 0.22 0.1 <= X <= 0.12
  bord Square Y = 0.12 0.1 <= X <= 0.12
},
```

**Pave Bottom**

```
{
  Origine 0.1 0.
  Nombre_de_Noeuds 2 11
  Longueurs 0.02 0.1
}
{
  bord Lower Y = 0. 0.1 <= X <= 0.12
  bord Square Y = 0.1 0.1 <= X <= 0.12
},
```

**Pave Right**

```
{
  Origine 0.12 0.
  Nombre_de_Noeuds 41 23
  Longueurs 0.8 0.22
}
{
  bord Outlet X = 0.92 0. <= Y <= 0.22
  bord Lower Y = 0. 0.12 <= X <= 0.92
  bord Upper Y = 0.22 0.12 <= X <= 0.92
  bord Square X = 0.12 0.1 <= Y <= 0.12
}
```



(Ox,Oy) Lx Nx

**Origine:** Block origin Ox Oy [Oz]

**Nombre\_de\_noeuds:** Nodes number Nx Ny [Nz]

**Longueurs:** Lengths Lx Ly [Lz]

**Bord:** Boundary

Ly Ny

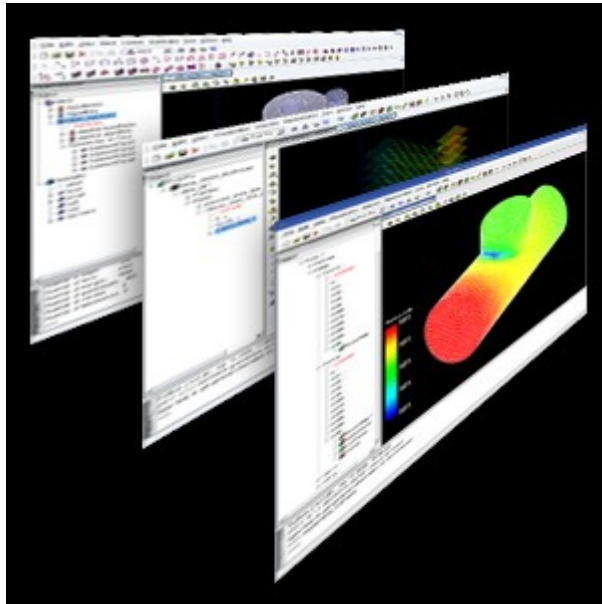
## Mesh generators (2/3)

---

- Possible meshing tools
- TRUST internal mesh tool
- **Salomé**
- Gmsh
- Read a med file in TRUST

# Salomé (1/1)

---



**Salomé:** An OpenSource platform (CEA, EDF, OpenCascade,...) which provides a 2D/3D element mesh generator. It is available here for download:

<http://www.salome-platform.org>

- > **Salomé** is not provided in the TRUST package
- > To have a full training session, or receive support for install or use, see:

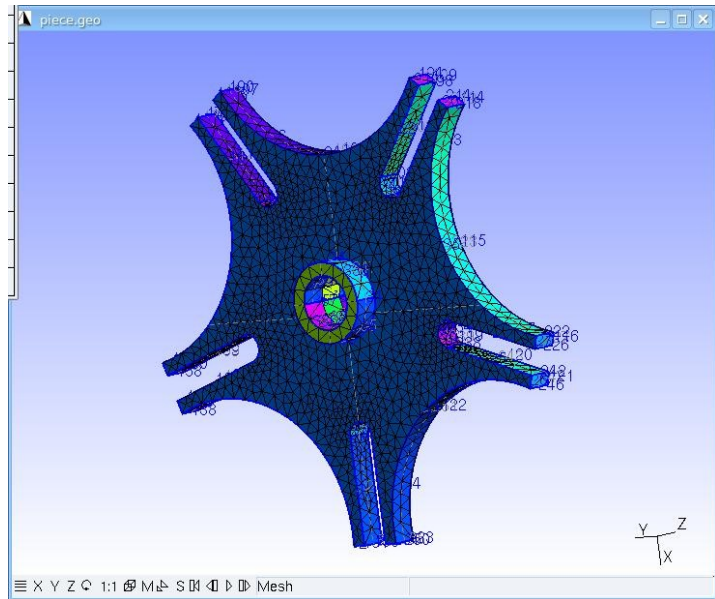
<http://www.salome-platform.org/service-and-support>  
[support-salome@cea.fr](mailto:support-salome@cea.fr)

# Mesh generators (3/3)

---

- Possible meshing tools
- TRUST internal mesh tool
- Salomé
- **Gmsh**
- Read a med file in TRUST

# Gmsh (1/3)



**Gmsh:** A 2D/3D finite element mesh generator available here:

<http://www.geuz.org/gmsh>

-> The documentation is here:

<http://geuz.org/gmsh/doc/texinfo/gmsh.html>

-> Gmsh is downloaded and built during the TRUST install. There are tutorials and examples under:

`$TRUST_ROOT/exec/gmsh/share/doc/gmsh/tutorial`

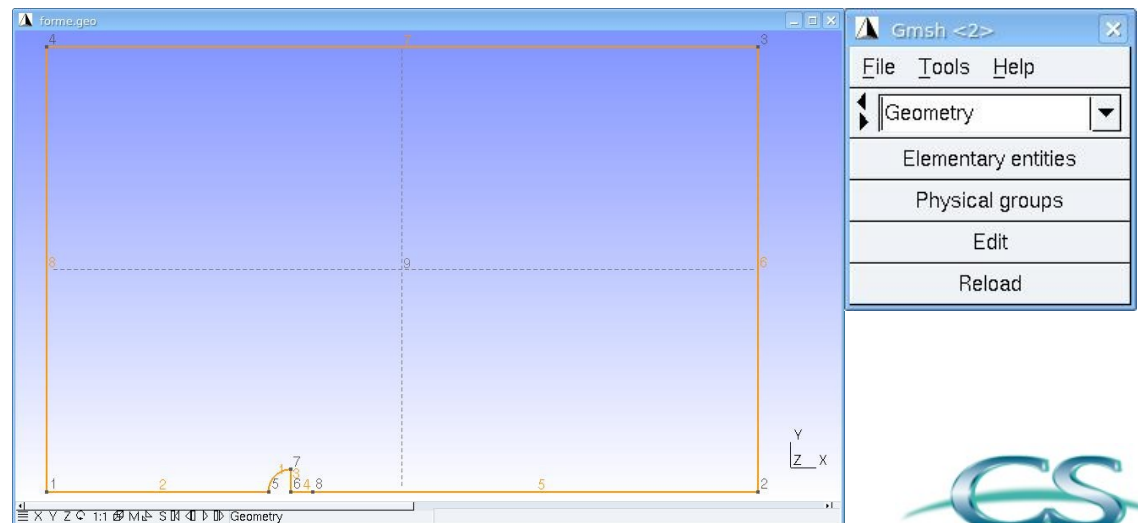
`$TRUST_ROOT/exec/gmsh/share/doc/gmsh/demos`

-> Support on Gmsh at [gmsh@geuz.org](mailto:gmsh@geuz.org)

Best is to start from a .geo file of one of the previous examples. Run Gmsh with:

**gmsh** *file.geo*

Edit and change your .geo file and use the **Reload** button to update the geometry visualization.



## Example of .geo file

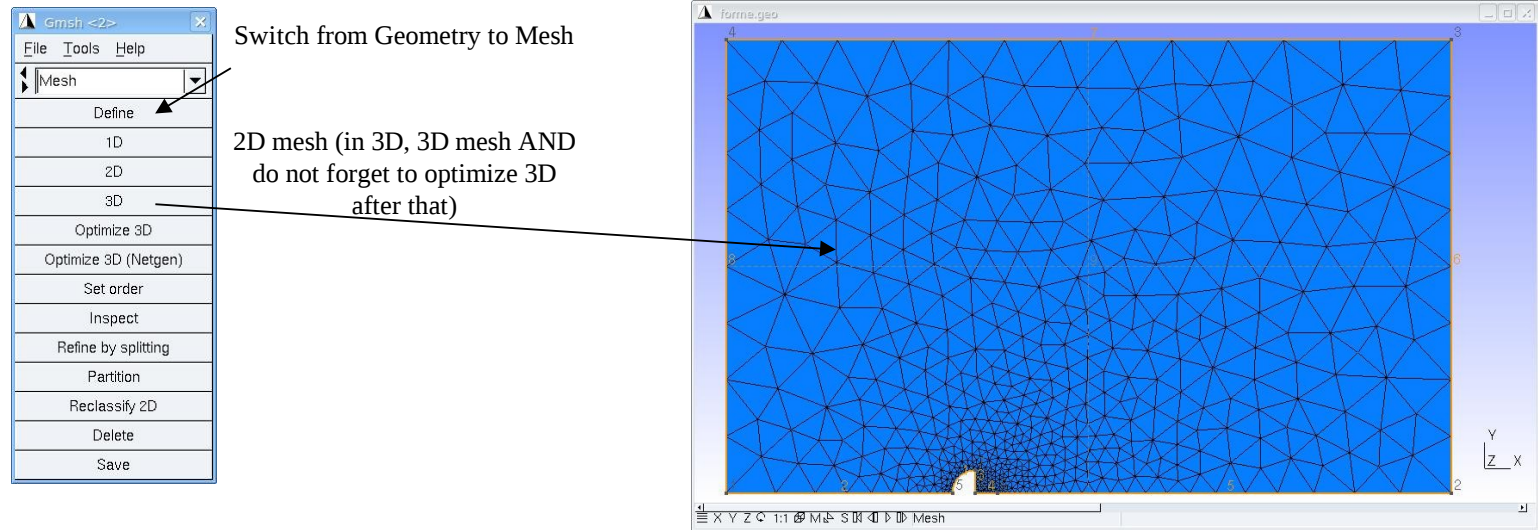
```
// Naming the boundaries is MANDATORY
// and it is thanks to the
// Physical Line (use Lines or Circle to
define it)
// DO NOT USE LINE LOOPS !!!!
Physical Line("Shape") = {1,3};
Physical Line("Axis") = {2,4,5};
Physical Line("Outlet") = {6};
Physical Line("Top") = {7};
Physical Line("Inlet") = {8};

// A lineloop is a loop on several lines
// for defining/orienting a surface
// Use negative lines to reverse the
// orientation of the line
Line Loop(1) = {2,1,3,4,5,-6,7,8};
// The surface will use the lineloop
Plane Surface(1) = {1};
// Naming the domain is MANDATORY
Physical Surface("domain") = {1};
```





# Gmsh (3/3)



Then, export the mesh to a MED format file (File->Save As, format MED) and DO NOT select “Save All” because points could be saved.

Important: Check that your mesh is created with the command (if nothing appears, you forgot to name boundaries and/or the domain with the Physical keywords):

**gmsh file.med**

# Mesh generators (3/3)

---

- Possible meshing tools
- TRUST internal mesh tool
- Salomé
- Gmsh
- **Read a med file in TRUST**

# Gmsh (3/3)

---

\* To import a .med mesh file, add in the TRUST datafile:

**Dimension** 2

**Domaine** dom

*# By default, the mesh name is the name of the file, so there mesh\_name=file #*

**Read\_MED family\_names\_from\_group\_names** dom mesh\_name *file.med*

\* If you have created several domains into the same Gmsh mesh, you will add cause the different domains in the Gmsh mesh file are seen as subzones.

**Domain** second\_dom

**Create\_domain\_from\_sous\_zone** { **domain\_final** second\_domain **par\_sous\_zone** sub\_zone\_name **domaine\_init** dom }

\* Possibility to **create a TRUST data file by opening a .med mesh**:

> **trust -wiz**

⇒ in the choice of domain set your .med file.

The wizzard will automatically find your boundary names,...

# Practice

---

## Exercise: Meshing tools

### VEF calculation with TRUST

- Run Salomé exercise in the tutorial
- Run Gmsh exercise in the tutorial

*Run Xprepro exercise in the tutorial if one is interested by a VDF calculation with TRUST*

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- **Automating validation test case**
- TRUST/TrioCFD support
- Examples of data files
- Recommendations

# Automating validation test case (1/15)

---

- What is an automated test case ?
- How to generate an automated test case?
- How to run an automated test case?

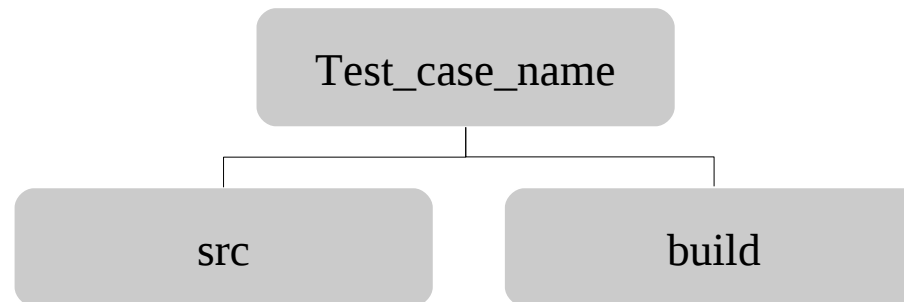
# Automating validation test case (2/15)

---

- **Verification of the non-regression** is done by running **.prm automated tests cases.**
- This is a **tool to compare TRUSTs results** and experimental data and/or analytical solutions.
- Finally it creates a **report in PDF format** containing:
  - » figures (images or gnuplot plots)
  - » tables with results
  - » visualizations (built by VisIt tool)
  - » etc...
- Useful to quickly validate a new TRUST/TrioCFD version or to compare different versions of the code

# Automating validation test case (3/15)

- **How to generate an automated test case?**
  - Create a directory for example Test\_case\_name and under:
    - src directory will contain the elements to build the test case
  - TRUST will create a build directory which will contain the results of the building run



- Example of a study report



# Automating validation test case (4/15)

---

- **How to generate an automated test case?**
  - In the src directory, you will create a:
    - **.prm** file (mandatory)
      - It contains the automated test case description
    - **prepare** script (optional)
      - It will build one (several) data file(s) in the build sub-directories
    - **pre\_run** script (optional)
      - It will do operations BEFORE calculations begin, for example the mesh partition before a parallel calculation
    - **post\_run** script (optional)
      - It will extract relevant information from the raw result files AFTER the calculation finishes (run python scripts,...)
    - other optional files/directories necessary to build the test case
      - data file, mesh file, experimental data, images,...

# Automating validation test case (5/15)

- Example of a **prm** file and his **pdf**:

## Parameters {

**Title** "Check Boussinesq source term in VEF"

**Author** "Marthe Roux (CS) from initial work by Ulrich Bieder (CEA)"

**Testcase** . diffusion

**Testcase** . convection\_diffusion

...

}

## 2 CALCULATION DOMAIN

Check Boussinesq source term in VEF

## 1 Introduction

Validation made by : Marthe Roux (CS) from initial work by Ulrich Bieder (CEA).  
Report generated 13/12/2016.

### 1.1 Description

### 1.2 Parameters TRUST

- Version TRUST :
- Version Trio\_U from out: /data/tmpletr/triou/installations/TRUST\_1.7.4/FED\_18\_64/TrioCFD<sub>i</sub> (1.7.4)

### 1.3 Test cases

- ./diffusion.data :
- ./convection\_diffusion.data :

# Automating validation test case (6/15)

- Example of a **prm** file and his **pdf**:

**Chapter {**

**Title** "Discretization"

**Description** "The calculation domain is meshed in a pure tetrahedral grid. The two discretisation methods described the nexts parts are analysed."

**Figure {**

**Title** "2D discretisation"

**description** "P1NC Velocity localization is on the centre of the faces, enlight by the square symbol  $\blacksquare$ ."

**description** "P0 Pressure localization is on the centre of the element, enlight by the circle symbol  $\bullet$ ."

**description** "P1 Pressure localization is on the vertices, enlight by the circle symbol  $\bullet$ ."

**Width** 5cm

**picture** triangle.pdf

**}**

...

**}**

## 5 Discretization

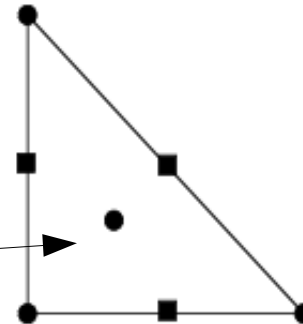
The calculation domain is meshed in a pure tetrahedral grid. The two discretisation methods described the nexts parts are analysed.

### 5.1 2D discretisation

P1NC Velocity localization is on the centre of the faces, enlight by the square symbol  $\blacksquare$ .

P0 Pressure localization is on the centre of the element, enlight by the circle symbol  $\bullet$ .

P1 Pressure localization is on the vertices, enlight by the circle symbol  $\bullet$ .



# Automating validation test case (7/15)

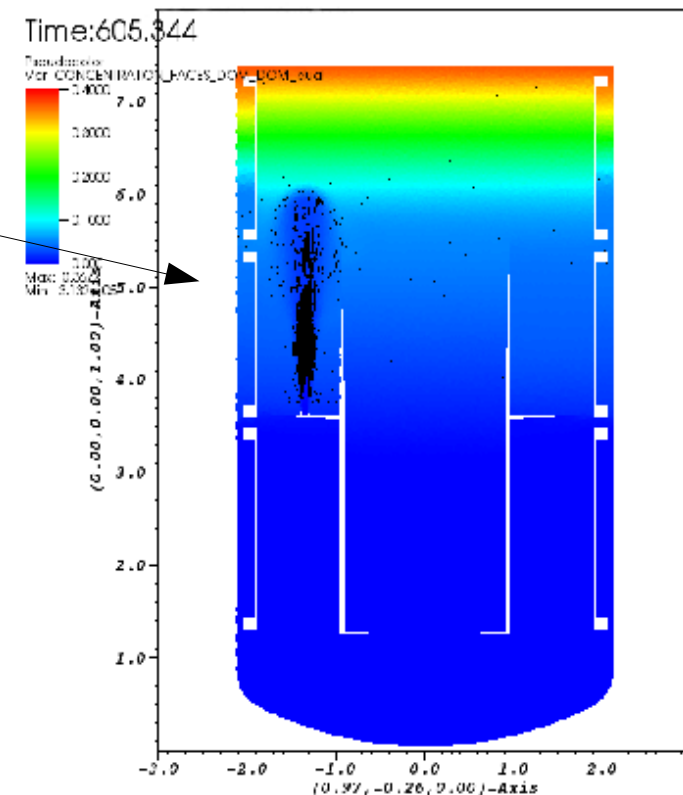
- Example of a **prm** file and his **pdf**:

```
Chapter {  
  Title "Results"  
  Description "The erosion of the helium layer by the jet is shown in  
Figure 3 (Trio_U-CFD result). ..."  
  Description "\latex_(\vspace{0.5cm})\textbf{Figure 3: erosion of  
the helium layer by an air jet (Fr = 1)} ($\theta=135$)\latex_)"  
  Visu {  
    Width 16cm  
    pseudocolor_with_range calculs/results.lata  
      DOM_DOM_dual CONCENTRATION FACES 0. 0.4  
    operator slice2D 0 0 0 -0.258819 -0.965926 0.  
    BlackVector_with_nb calculs/results.lata  
      DOM_DOM_dual VITESSE FACES 3000 1.4  
    operator clip_1plane 0 0 3.75 0 0 -1.  
    operator slice2D 0 0 0 -0.258819 -0.965926 0.  
    zoom2D -3. 3. 0. 8.  
  }  
}
```

## 5 Results

The erosion of the helium layer by the jet is shown in Figure 3 (Trio\_U-CFD result). We can see the low penetration of the jet into the stratified layer of light gas accumulated at the top and the transport of helium down to the edge of the impact zone of the jet.

Figure 3: erosion of the helium layer by an air jet ( $Fr = 1$ ) ( $\theta = 135$ )

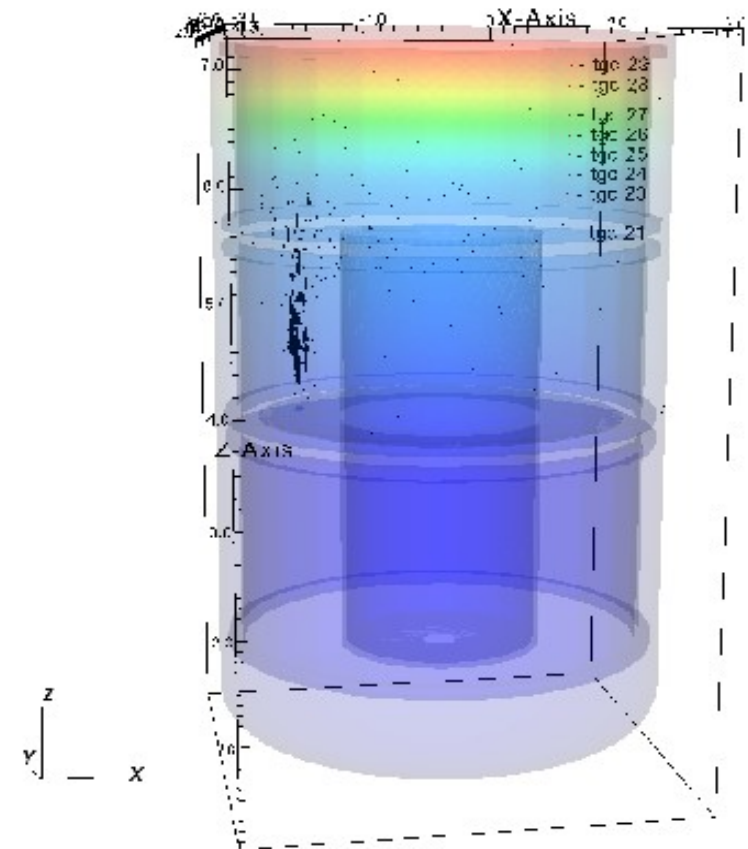


# Automating validation test case (8/15)

- Example of a **prm** file and his **pdf**:

```
Visu {  
  Width          16cm  
  pseudocolor_with_opacity  calculs/Ghost.lata  
    DOM_DOM_dual CONCENTRATION FACES 0.1  
  BlackVector_with_nb    calculs/Ghost.lata DOM_DOM_dual  
    VITESSE FACES 3000 1.4  
  
  operator      clip_1plane 0 0 3.75 0 0 -1.  
  mesh          calculs/Sondes.son MESH red  
  insertText    1.38 3.51e-01 5.40 0.02 - tgc 21  
  insertText    1.38 3.51e-01 5.80 0.02 - tgc 23  
  insertText    1.38 3.51e-01 6.00 0.02 - tgc 24  
  insertText    1.38 3.51e-01 6.20 0.02 - tgc 25  
  insertText    1.38 3.51e-01 6.40 0.02 - tgc 26  
  insertText    1.38 3.51e-01 6.60 0.02 - tgc 27  
  insertText    1.38 3.51e-01 6.90 0.02 - tgc 28  
  insertText    1.38 3.51e-01 7.10 0.02 - tgc 29  
  operator_to_all  no_databaseinfo  
  operator_to_all  no_legend  
  zoom2D         -3. 3. 0. 8.  
  up3d           0.027157 0.173858 0.984396  
  normal3D       -0.180938 -0.967639 0.17589  
}
```

Figure 4: Location of concentration sampling probes



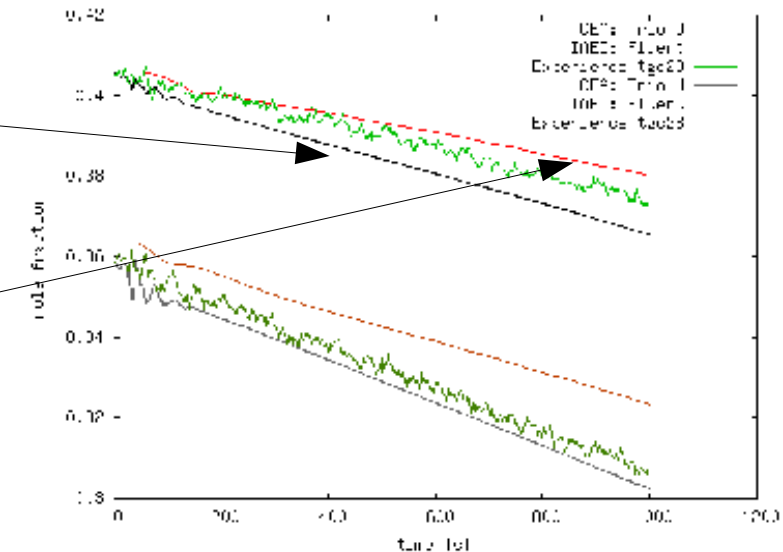
# Automating validation test case (9/15)

- Example of a **prm** file and his **pdf**:

```
Figure {  
  labelx "time [s]"  
  labely "mole fraction"  
  Width 13cm  
  Include_Description_Curves 0  
  legend top  
  Curve {  
    fichier calculs/results_SONDE_C_V7.son  
    colonnes $1  
    $22-read_value_in_file("VAR_EXPE_11","calage")  
    legende "CEA: Trio_U"  
    style lines  
    TypeLigne -1  
  }  
  Curve {  
    fichier Calculs_IAEC/calc_fluent.txt  
    colonnes $1 $11  
    legende "IAEC: Fluent"  
    style lines  
    TypeLigne 1  
  }  
}
```

## 5 RESULTS

Figure 5: Comparison of the concentration of helium measured and calculated at different heights



# Automating validation test case (10/15)

- Example of a **prm** file and his **pdf**:

```
Table {  
  Title "Physical properties"  
  nb_columns 1  
  line {  
    legend "Cinematic viscosity  $\nu$  ( $m^2/s$ )"  
    file propertiesGeometry.dat  
    nb_columns_file 5  
    columns ($1)  
  }  
  line {  
    legend "Density  $\rho$  ( $kg/m^3$ )"  
    file propertiesGeometry.dat  
    nb_columns_file 5  
    columns ($2)  
  }  
  ...  
}
```

5 DISCRETIZATION

## 4 Physical Properties

Air is used as a representative gas. Its physical properties at 20°C are given in the next table.

### 4.1 Physical properties

Cinematic viscosity $\nu$ ( $m^2/s$ )	1.511e-05
Density $\rho$ ( $kg/m^3$ )	1.205
Thermal diffusivity $\alpha$ ( $m^2/s$ )	0.0257
Heat capacity $C_p$ ( $J/(kg K)$ )	1005
Thermal expansion coefficient $\beta$ ( $1/K$ )	0.00343

# Automating validation test case (11/15)

- Example of a **prepare** script:

```
#!/bin/bash

# Loop on several convection schemes:
for scheme in "muscl ef_stab upwind"
do
    # Create a sub directory into the build directory
    mkdir $scheme

    # Go into the sub directory
    cd $scheme

    # Copy the flow.data into a new file named flow.data
    cp ../flow.data flow.data

    # Substitute into the the data file the SCHEME string by the the value of the $scheme variable
    echo -e "1,$ s?SCHEME?$scheme?g\nw" | ed flow.data

    # Create a link with pre_run and post_run script
    ln -s -f ../pre_run .
    ln -s -f ../post_run .

    # Come back to the build directory
    cd ..
done
```



# Automating validation test case (12/15)

---

- Example of a **pre\_run** script:

```
#!/bin/bash  
# Uncompress the mesh file  
gunzip -c ../Channel.msh.gz > Channel.msh  
# Partition the mesh with the make_PAR.data tool:  
make_PAR.data flow.data
```

# Automating validation test case (13/15)

- Example of **post\_run** script:

```
#!/bin/bash

# The first parameter is the name of the data file:
datafile=$1

file=${datafile%.data}

# Read the pressure drag on the 5th column of the last line (final time) of
# the pressure force file:
fp=`tail -1 $file"_pb_Force_pression.out" | awk '{print $5}'`

# Read the viscous drag:
fv=`tail -1 $file"_pb_Contrainte_visqueuse.out" | awk '{print $5}'`

# Calculate the total drag:
Drag=`echo $fp $fv | awk '{print $1+$2}'`

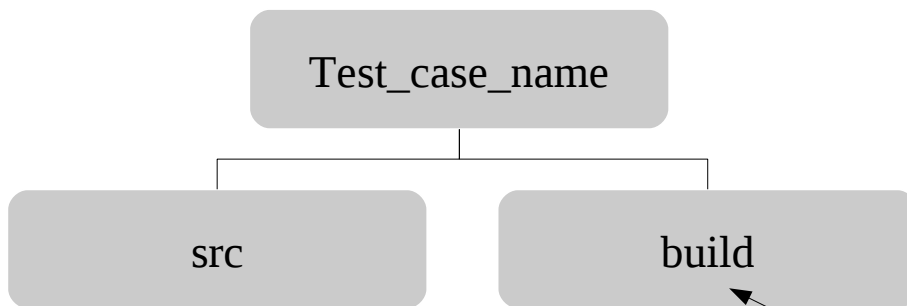
#Drag=`echo "$fp+$fv" | bc -l`

# Write the total drag into a file to be included into a table of the PDF file
echo $Drag > drag.dat

# You can also call python scripts...
```

# Automating validation test case (14/15)

- How to run an automated test case?
  - “**Run\_fiche**” command should be run either from the root directory of the test case, either in the src directory
  - All operations made by **Run\_fiche** are in the build directory:
    - » First, it runs the **prepare** script
    - » Then for each calculation:
      - runs the **pre\_run** script
      - runs the calculation
      - runs the **post\_run** script
    - » Then builds the **PDF report file**



You have access to the **latex/images/... files** in the directory:  
**build/.tmp**

# Automating validation test case (15/15)

---

- **User guide:**

\$TRUST\_ROOT/doc/TRUST/HowTo\_Validation.pdf

- **.prm syntax** documented in:

\$TRUST\_ROOT/Validation/Outils/Genere\_courbe/doc/manuel.xhtml

- **Examples** of automated verification test case:

\$TRUST\_ROOT/Validation/Rapports\_automatiques/Verification/Verification\_codage

- *To have **information** about scripts:*

*“trust -index” → “Memo scripts” link*

# Automating validation test case (15/15)

- **To identify all the data sets** from the non-regression data base which contain some specified keywords (word1 word2...wordn):  
***cherche.ksh** [-reference\_only] word1 word2 ...wordn*  
→ results in file 'liste\_cherche'
- **List of tests cases** with little explanation:  
"trust -index" → "Test cases" link
- **Tests cases ending with .jdd1, .jdd2 ...** are test cases from .prm automated test cases run on 3 time steps.
- Most of the **validation tests cases** are in TrioCFD.

# Practice

---

## Exercise with `cherche.ksh`:

```
> source /home/triou/env_TRUST_X.Y.Z.sh
> cherche.ksh Fluide_Quasi_Comp*
> more liste_cherche
> cherche.ksh *FT_Disc*
> source /home/triou/env_TrioCFD_X.Y.Z.sh
> cherche.ksh *FT_Disc*
> cherche.ksh Fluide_Quasi_Comp*
To have access to all test cases:
> source ../full_env_TrioCFD.sh
```

## Exercise: Validation form

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- **TRUST/TrioCFD support**
- Examples of data files
- Recommendations

# TRUST/TrioCFD support (1/2)

---

Subscribe to the TrioCFD newsletters (diffusion list):

- Users (1 mail/trimester) [http://saxifrage:3500/www/info/trio\\_u\\_annonces](http://saxifrage:3500/www/info/trio_u_annonces)
- Developers (1 mail/week) [http://saxifrage:3500/www/info/trio\\_u\\_dev](http://saxifrage:3500/www/info/trio_u_dev)

Download new versions from sourceforge site:

- TRUST: <http://sourceforge.net/projects/trust-platform/files/>
- TrioCFD: <http://sourceforge.net/projects/triocfd/files/>

See recent research publications related to TrioCFD project:

- Web site: <http://www-trio-u.cea.fr>

## Hot line

- **triou@cea.fr**
- Céline CAPITAINE 01 69 08 42 68 (Saclay bât 451 pièce 68B)
- Marthe ROUX 01 69 08 00 02 (Saclay bât 451 pièce 68B)



# TRUST/TrioCFD support (2/2)

---

- A release every 6 months:
  - Linux version only
  - Installed on several CEA clusters, TGCC/CCRT and CINES
  - Installation by users or TRUST/TrioCFD support
- Documentation available under \$TRUST\_ROOT/doc/TRUST directory else ask it to CEA project leaders:
  - **TRUST\_and\_TrioCFD\_presentation.pdf** (these slides)
  - **TRUST\_tutorial.pdf** TRUST/TrioCFD/Meshing exercises
  - **Models\_Equations\_TRUST.pdf** “Methodology for incompressible single phase flow”
  - **Best\_Practice\_TRUST.pdf** “Validation of Trio\_U code”
  - **TRUST\_Generic\_Guide.pdf** “User Manual TRUST/TrioCFD”
  - **HowTo\_Validation.pdf** “Organisation of TrioCFD validation data base”
  - **Manuel\_Xprepro.pdf** “User Manual Xprepro”
  - **Developer\_TRUST\_presentation.pdf** TRUST development Presentation

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- **Examples of data files**
- Recommendations

# Examples of data files (1/3)

---

- **Front Tracking calculation.....** p147
- Quasi compressible example.....p153
- Genepi+ data file.....p160

# Front Tracking calculation (1/4)

# Front Tracking calculation #

dimension 3

# Generic problem used for Front Tracking calculation #

Probleme\_FT\_Disc\_gen pb

Domaine DOM

# BEGIN MESH #

Mailler DOM

```
{
  Pave pave1
  {
    origine 0. 0. 0.
    longueurs 0.04 0.04 0.06
    nombre_de_noeuds 11 11 16

    bord paroi X = 0. 0. <= Y <= 0.04 0. <= Z <= 0.06
    bord paroi X = 0.04 0. <= Y <= 0.04 0. <= Z <= 0.06
    bord paroi Y = 0. 0. <= X <= 0.04 0. <= Z <= 0.06
    bord paroi Y = 0.04 0. <= X <= 0.04 0. <= Z <= 0.06
    bord bas Z = 0. 0. <= X <= 0.04 0. <= Y <= 0.04
    bord haut Z = 0.06 0. <= X <= 0.04 0. <= Y <= 0.04
  }
}
```

# END MESH #

# BEGIN PARTITION

Partition DOM

```
{
  Partitionneur tranche { tranches 2 1 1 }
  Larg_joint 2
  Nom_Zones DOM
}
```

End

END PARTITION #

# BEGIN READ

Scatter DOM.Zones dom

END SCATTER #

VDF dis

Schema\_Euler\_explicite sch

Read sch

```
{
  tinit 0.
  tmax 0.1
  dt_min 1.e-7
  dt_max 0.5e-2
  dt_impr 10.
  dt_sauv 100
  seuil_statio -1
}
```

# First phase: liquid #

Fluide\_Incompressible liquide

Read liquide

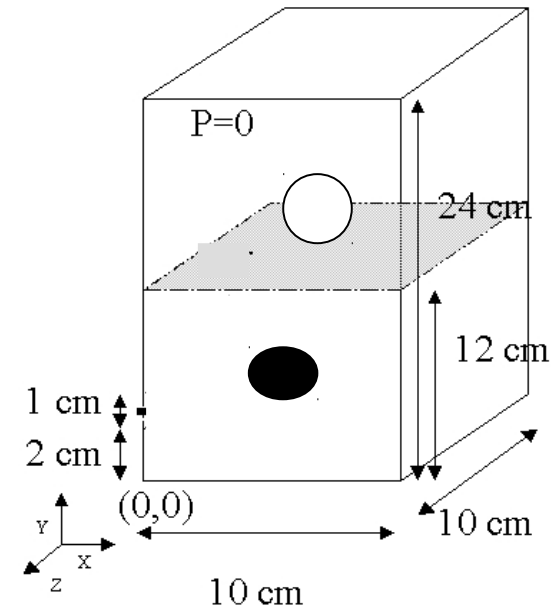
```
{
  mu Champ_Uniforme 1 0.282e-3
  rho Champ_Uniforme 1 1000.
}
```

# Second phase: gas #

Fluide\_Incompressible gaz

Read gaz

```
{
  mu Champ_Uniforme 1 0.282e-3
  rho Champ_Uniforme 1 100.
}
```



# Front Tracking calculation (2/4)

```
# Definition of the two phase media #
Fluide_Diphasique fluide
Read fluide
{
    # Give a number for each phase #
    fluide0 liquide
    fluide1 gaz
    # Surface tension #
    sigma champ_uniforme 1 0.05
}
# Add a constituent #
Constituant constituant
Read constituant
{
    diffusivite Champ_Uniforme 1 1e-6
}
# Gravity field #
Champ_Uniforme gravite
Read gravite 3 0. 0. -9.81
Associate fluide gravite

# Navier Stokes equation #
Navier_Stokes_FT_Disc hydraulique
# One equation for the two phase flow interface #
Transport_Interfaces_FT_Disc interf
# One equation for a moving body #
Transport_Interfaces_FT_Disc body
# One equation for the constituent #
Convection_Diffusion_Concentration concentration

Associate pb hydraulique
Associate pb interf
Associate pb body
Associate pb concentration
Associate pb DOM
Associate pb sch
```

```
Associate pb fluide
Associate pb constituant
Discretize pb dis
# Define the front tracking problem #
Read pb
{
    hydraulique
    {
        # Turbulence model needed and zeroed for laminar flow #
        modele_turbulence sous_maille_wale {
            Cw 0 turbulence_paro negligible
        }
        # Iterative method to solve the pressure linear system with a non-constant matrix #
        solveur_pression GCP { precondition ssor { omega 1.5 } seuil 1e-12 impr }
        convection { quick }
        diffusion { }
        conditions_initiales { vitesse champ_uniforme 3 0. 0. 0. }
        # Relation between Navier Stokes equation and interface equations #
        equation_interfaces_proprietes_fluide interf # The velocity field moves the gas-liquid interface #
        equation_interfaces_vitesse_imposee body # The body has an imposed velocity field, so moves the fluid #
        boundary_conditions
        {
            haut Sortie_libre_rho_variable champ_front_uniforme 1 0. # Outlet boundary condition for FT model #
            paroi paroi_fixe
            bas Frontiere_ouverte_vitesse_imposee champ_front_uniforme 3 0.0 0.0 0.001
        }
    }
}
```

# Front Tracking calculation (3/4)

```

interf
{
    # Definition of the transport method of the interface: velocity from the
    # Navier Stokes equation #
    methode_transport vitesse_interpolee hydraulique
    # Initial position of the water-gas interface and a drop of water #
    conditions_initiales {
        fonction z-0.03-((x-0.02)^2+(y-0.02)^2)*10 ,
        fonction ajout_phase0 (x-0.02)^2+(y-0.02)^2+(z-0.045)^2-(0.01)^2
    }
    # Options for the meshing algorithm #
    iterations_correction_volume 1
    n_iterations_distance 2
    remaillage {
        pas 0.000001 nb_iter_remaillage 1
        critere_arete 0.35 critere_remaillage 0.2
        pas_lissage 0.000001 lissage_courbure_iterations 3
        lissage_courbure_coeff -0.1 nb_iter_barycentrage 3
        relax_barycentrage 1 facteur_longueur_ideale 0.85
        nb_iter_correction_volume 3
        seuil_dvolume_residuel 1e-12
    }
    # Algorithm for the collision algorithm between interfaces #
    collisions
    {
        active juric_pour_tout
        type_remaillage Juric { source_isevaleur indicatrice }
    }
    # Boundary condition, variable contact angle is possible #
    boundary_conditions
    {
        paroi Paroi_FT_disc symetrie
        haut Paroi_FT_disc symetrie
        bas Paroi_FT_disc symetrie
    }
}

```

```

body
{
    # Initial position of the moving body #
    conditions_initiales { fonction -(((x-0.02))^2+((y-0.02)/0.6)^2+((z-
    0.02)/0.6)^2-(0.015^2)) }
    remaillage { pas 1e8 }
    boundary_conditions
    {
        haut Paroi_FT_disc symetrie
        paroi Paroi_FT_disc symetrie
        bas Paroi_FT_disc symetrie
    }
    # 2 methods to move the body: velocity(x,y,z)=f(x,y,z) or
    x(t),y(t),z(t)=f(t),g(t),h(t) #
    methode_transport vitesse_imposee
        -(y-0.02)*10 (x-0.02)*10 0.
    }
    # Constituent equation #
    concentration
    {
        diffusion { negligeable }
        convection { quick }
        conditions_initiales { concentration champ_fonc_xyz DOM 1 EXP(-((x-
        0.02)^2+(y-0.02)^2+(z-0.03)^2)/0.03^2) }
        boundary_conditions {
            haut frontiere_ouverte C_ext Champ_Front_Uniforme 1 0.
            paroi paroi # Concentration flux = 0 #
            bas paroi
        }
    }
}

```



# Front Tracking calculation (4/4)

## Postraitement

```
{
  Sondes {
    vitesse vitesse periode 1.e-7 point 1 0.02 0.02 0.03
    pression pression periode 1.e-7 point 1 0.02 0.02 0.03
    indicatrice_interf indicatrice_interf periode 1.e-7 point 1 0.02 0.02 0.03
  }
  Champs dt_post 0.05
  {
    indicatrice_interf
    concentration
  }
}
liste_postraitements
{
  # Another keywords to post process FT results #
  Postraitement_ft_lata liquid_gas
  {
    dt_post 0.05 nom_fichier liquid_gas
    format binaire print
    champs sommets { vitesse }
    champs elements
    {
      distance_interface_elem_interf
      distance_interface_elem_body
      indicatrice_interf
      pression
      concentration
      vitesse
    }
    # Post process the moving grid of the interface #
    interfaces interf {
      champs sommets { courbure vitesse }
    }
  }
}
```

```
}
Postraitement_ft_lata body
{
  dt_post 0.05
  nom_fichier body
  format binaire
  print
  interfaces body {
    champs sommets { courbure }
  }
}
}
}
Solve pb
End
```

# Practice

---

## Exercise: Tank Filling 3D



## Examples of data files (2/3)

---

- Front Tracking calculation
- **Quasi compressible example**
- Genepi+ data file

# Quasi compressible example (1/5)

TRUST coupled with  
Cathare via ICoCo

```
dimension 2
Domaine dom_fluid
Mailler dom_fluid {
  Pave Entree1 {
    Origine 0. 0.
    Nombre_de_Noeuds 4 3
    Longueurs 0.6 0.4
  }
  {
    Bord inlet1 Y = 0.    0. <= X <= 0.6
    Bord wall  X = 0.    0. <= Y <= 0.4
    Bord wall  X = 0.6   0. <= Y <= 0.4
  },
  Pave Sortie1 {
    Origine 0. 5.4
    Nombre_de_Noeuds 4 3
    Longueurs 0.6 0.4
  }
  {
    Bord outlet1 Y = 5.8   0. <= X <= 0.6
    Bord wall  X = 0.    5.4 <= Y <= 5.6
    Bord wall  X = 0.6   5.4 <= Y <= 5.8
    Bord P_imp X = 0.    5.6 <= Y <= 5.8
  },
  Pave Entree2 {
    Origine 1. 0.
    Nombre_de_Noeuds 4 3
    Longueurs 0.6 0.4
  }
}
```

```
{
  Bord inlet2 Y = 0.    1. <= X <= 1.6
  Bord wall  X = 1.    0. <= Y <= 0.4
  Bord wall  X = 1.6   0. <= Y <= 0.4
},
Pave Sortie2 {
  Origine 1. 5.4
  Nombre_de_Noeuds 4 3
  Longueurs 0.6 0.4
}
{
  Bord outlet2 Y = 5.8   1. <= X <= 1.6
  Bord wall  X = 1.    5.4 <= Y <= 5.8
  Bord wall  X = 1.6   5.4 <= Y <= 5.6
  Bord P_imp X = 1.6   5.6 <= Y <= 5.8
},
Pave Enceinte {
  Origine 0. 0.4
  Nombre_de_Noeuds 9 26
  Longueurs 1.6 5.
}
{
  Bord wall  Y = 0.4      0.6 <= X <= 1.
  Bord wall  Y = 5.4      0.6 <= X <= 1.
  Bord wall  X = 0.        0.4 <= Y <= 5.4
  Bord wall  X = 1.6       0.4 <= Y <= 5.4
}
}
Trianguler_fin dom_fluid # Triangulate the mesh #
```

# Quasi compressible example (2/5)

```

Schema_Euler_implicite sch
Read sch
{
    tinit 0.
    tmax 10.
    dt_min 1.e-6
    dt_max 0.01
    dt_impr 0.01
    dt_sauv 1000.0
    seuil_statio 1.e-6
    # Options related to implicit scheme #
    facsec 20
    facsec_max 500
    Solveur Piso
    {
        # Convergence threshold for the iterative method #
        # (by default GMRES) to solve the unsymmetric #
        # linear system #
        seuil_convergence_solveur 1.e-6
    }
}

# Definition of the a thermohydraulic problem #
# using the Quasi-compressible model #
Pb_thermohydraulique_QC pb1

# VEF discretization selected #
VEFPreP1B dis

# Gravity defined as a uniform field #
Champ_Uniforme gravite
Read gravite 2 0 -9.81

```

```

# Define the media, helium gas #
Fluide_Quasi_Compressible helium
Read helium
{
    # Pressure in Pa #
    pression 7000000.

    # Sutherland law for viscosity and conductivity #
    Sutherland mu0 1.6E-5 T0 273.15 Slambda 235. C 235.

    # Ideal gas law #
    loi_etat gaz_parfait {
        Prandtl 1.
        # Specific heat at constant pressure #
        Cp 5193.
        # Cv, specific heat at constant volume given by gamma=Cp/Cv #
        gamma 1.666
    }

    # Keyword for open flow (Neuman condition ), total pressure is constant #
    traitement_pth constant
}

# Create link between objects #
Associate helium gravite
Associate pb1 dom_fluid
Associate pb1 sch
Associate pb1 helium
Discretize pb1 dis

```

# Quasi compressible example (3/5)

*# Definition of four surface domains #*  
*# each extracted from a boundary of the problem #*

```

domaine entree1
extraire_surface
{
    domaine entree1
    probleme pb1
    avec_certains_bords 1 inlet1
}
    
```

```

domaine entree2
extraire_surface
{
    domaine entree2
    probleme pb1
    avec_certains_bords 1 inlet2
}
    
```

```

domaine sortie1
extraire_surface
{
    domaine sortie1
    probleme pb1
    avec_certains_bords 1 outlet1
}
    
```

```

domaine sortie2
extraire_surface
{
    domaine sortie2
    probleme pb1
    avec_certains_bords 1 outlet2
}
    
```

**Read** pb1

```

{
    Navier_Stokes_QC
    {
        solveur_pression petsc cholesky { } # Direct solver, pressure matrix is constant #
        convection { muscl }
        diffusion {}
        conditions_initiales { vitesse Champ_Uniforme 2 47. 0. }
        boundary_conditions {
            # Dirichlet boundary condition mass flow rate #
            # Ch_front_input : ICOCO coupling field #
            inlet1 frontiere_ouverte_rho_u_impose ch_front_input {
                nb_comp 2 nom rho_u_entree1 probleme pb1
            }
            inlet2 frontiere_ouverte_rho_u_impose ch_front_input {
                nb_comp 2 nom rho_u_entree2 probleme pb1
            }
            outlet1 frontiere_ouverte_rho_u_impose ch_front_input {
                nb_comp 2 nom rho_u_sortie1 probleme pb1
            }
            outlet2 frontiere_ouverte_rho_u_impose ch_front_input {
                nb_comp 2 nom rho_u_sortie2 probleme pb1
            }
            P_imp Frontiere_ouverte_pression_imposee
                    Champ_front_Uniforme 1 70.e5
            wall paroi_fixe
        }
    }
    # Definition of a source term (pressure loss) #
    sources {
        Perte_Charge_isotrope {
            diam_hydr champ_fonc_txyz dom_fluid 1 1.4-1.3908*(y)1.4)*(y[4.4)
            lambda 4*((16/Re)$(0.079/(Re^0.25))$0.003)
        }
    }
}
    
```

# Quasi compressible exemple (4/5)

## Convection\_Diffusion\_chaleur\_QC

```
{
    diffusion {}
    convection { muscl }
    conditions_initiales { Temperature Champ_Uniforme 1 673. }
    boundary_conditions {

        # Boundary conditions for temperature #
        # ch_front_input coupling keyword #
        inlet1 frontiere_ouverte t_ext ch_front_input {
            nb_comp 1 nom temperature_entree1 probleme pb1
        }
        inlet2 frontiere_ouverte t_ext ch_front_input {
            nb_comp 1 nom temperature_entree2 probleme pb1
        }
        outlet1 frontiere_ouverte t_ext ch_front_input {
            nb_comp 1 nom temperature_sortie1 probleme pb1
        }
        outlet2 frontiere_ouverte t_ext ch_front_input {
            nb_comp 1 nom temperature_sortie2 probleme pb
        }

        # Outlet boundary condition for temperature #
        P_imp Frontiere_ouverte T_ext champ_front_uniforme 1 700.
        # Zero heat flux #
        wall paroi_adiabatique
    }

    # Heat source term #
    sources
    {
        Puissance_thermique champ_fonc_xyz dom_fluid 1
        2400e6*2/3./10./(3*3.1416*0.7*0.7)*0.6/1.8*1.53*(y)1.4)*(y<4.4)
    }
}
```

## Postraitement

```
{
    # As usual, just notice the masse_volumique keyword
    # for the volume mass field #
    Sondes
    {
        sonde_pression1 pression periode 0.05 segment 31
        0.1 0 0.1 5.8
        sonde_pression2 pression periode 0.05 segment 31
        1.5 0 0.1 5.8
        sonde_vitesse1 vitesse periode 0.05 segment 31 0.1
        0 0.1 5.8
        sonde_vitesse2 vitesse periode 0.05 segment 31 1.5 0
        0.1 5.8
        sonde_temperature1 temperature periode 0.05 segment
        31 0.1 0 0.1 5.8
        sonde_temperature2 temperature periode 0.05 segment
        31 1.5 0 0.1 5.8
        sonde_rho1 masse_volumique periode 0.05 segment 31
        0.1 0 0.1 5.8
        sonde_rho2 masse_volumique periode 0.05 segment 31
        1.5 0 0.1 5.8
    }

    format lata
    Champs dt_post 0.05
    {
        pression elem
        pression som
        vitesse elem
        temperature elem
        temperature faces
        masse_volumique elem
    }
}
```

# Quasi compressible example (5/5)

```
Poitraitements      # List of postprocessing blocks #
{
entree1 {
    # Restrict postprocessing on surface meshes previousluy defined #
    domaine entree1
    # Keyword to create new postprocessing fields #
    Definition_champs
    {
        # Define a pressure surface field by Interpolating pressure #
        # volume field on the elements of the surface mesh #
        pressure_entree1 champ_post_interpolation
        {
            localisation elem
            domaine entree1
            source champ_post_refchamp {
                pb_champ pb1 pression
            }
        }
        # Define a temperature surface field #
        temperature_entree1 champ_post_interpolation
        {
            localisation elem
            domaine entree1
            source champ_post_refchamp {
                pb_champ pb1 temperature
            }
        }
    }
}
# Same tasks for each surface meshes #
entree2 { ... }
sortie1 { ... }
sortie2 { ... }
}
Solve pb1          # Solve the problem #
End
```

# Practice

---

## Exercise: Low Mach number flow

## Examples of data files (3/3)

---

- Front Tracking calculation
- Quasi compressible example
- **Genepi+ data file**



# Genepi+ data file (1/4)

## Old data file

```
dimension 3
ecriturelecturespecial 1 # Keyword to not save .xyz file #

# Create domains #
domaine dom
domaine MAILLAGE_GENEPI
# Read a MED file (may be composed of several meshes, a
given mesh name is necessary) #
lire_med dom MAILLAGE_GENEPI tface.med
lire_med MAILLAGE_GENEPI MAILLAGE_GENEPI maigro.med

# The keyword is used to rename boundary #
regroupebord dom Wall { WALL_01 }
regroupebord dom Entree_branche_chaude { INLET_03 }
regroupebord dom Entree_branche_froide { INLET_02 }
regroupebord dom Sortie { OUTLET_04 }

ef dis # Define the specific discretization for Genepi+ #

schema_euler_implicite sch2 # Define an implicit scheme #
Lire sch2 {
  tinit 0.0 tmax 55.0
  dt_min 0.02439 dt_max 0.02439
  dt_start dt_fixe 0.02439 # To insure the first time step is imposed #
  facsec 50000
  nb_pas_dt_max 40000
  dt_sauv 1.0 dt_impr 0.5
  seuil_statio 1e-05
  solveur implicite {
    solveur gmres # Iterative solver for the implicit linear system #
    {
      impr seuil 1e-08 # Threshold convergence for the solver #
      diag # Diagonal pre-conditioning #
    }
  }
}
```

```
# Define the gravity field #
champ_uniforme gravite
Lire gravite 3 0.0 0.0 -9.80665

# Specific Genepi+ keywords: #
fluide_melange freon
Lire freon {
  type_fluide fr_114_9_12b
  init_file dump.all
}
pb_melange pb

# Link objects #
Associer freon gravite
Associer pb dom
Associer pb sch2
Associer pb freon
discretiser pb dis

# Keyword to write the domain in a LATA file to visualization purpose #
postraiter_domaine { format lata fichier geom domaine dom }

# Keyword to define a volume porosity field and a surface porosity field #
# Here the fields are read from a MED file #
porosites_champ pb champ_fonc_med last_time betan_betae.med MAILLAGE_GENEPI
POROSITE_2 elem 0.0
```

# Genepi+ data file (2/4)

```
Lire pb {
  # Genepi+ keywords #
  navier_stokes_melange {
    # Usual keywords for Navier Stokes #
    solveur_pression gcp
    {
      precondition { omega 1.5 } seuil 1e-08
    }
    convection { generic amount }
    diffusion { option { grad_u_transpose_partout 1 } }
    conditions_initiales { vitesse champ_fonc_xyz dom 3 0 0 0 }
    conditions_limite
    {
      Wall symetrie
      Sortie frontiere_ouverte_pression_imposee champ_front_uniforme 1 880000.0
      # Boundary condition can also be read in MED files, here velocity profiles: #
      Entree_branche_chaude frontiere_ouverte_vitesse_imposee champ_front_med_test champ_fonc_med last_time maigro.med MAILLAGE_GENEPI VITESSE_3 som 0.0
      Entree_branche_froide frontiere_ouverte_vitesse_imposee champ_front_med_test champ_fonc_med last_time maigro.med MAILLAGE_GENEPI VITESSE_3 som 0.0
    }
    modelisation {
      diffusion_turbulente 1
      prandtl 0.5
      faisceau
      {
        champ_rotation champ_fonc_med last_time rotation_faisceau.med MAILLAGE_GENEPI champ_vectoriel_1 elem 0.0
        champ_presence champ_fonc_med last_time rotation_faisceau.med MAILLAGE_GENEPI champ_scalaire_2 elem 0.0
        transpose_rotation
      }
      plaques_in_file plaques.data # Definition in an external data file #
      init_file dump.all
    }
    pression_initiale champ_uniforme 1 880000
    # List of source terms : separated by comma #
    sources { source_derive , source_frottement , source_pesanteur }
  }
}
```

# Genepi+ data file (3/4)

```
enthalpie_melange # Energy equation definition #
{
  convection { generic amount }
  diffusion { option { grad_u_transpose_partout 1 } }
  conditions_initiales { enthalpie champ_uniforme 1 140000.0 }
  conditions_limites {
    Wall symetrie
    Sortie frontiere_ouverte T_ext champ_front_fonc_txyz 1 1.193e5
    # An example of space dependant boundary condition, hear an Heaviside function for enthalpy: #
    Entree_branche_chaude frontiere_ouverte_enthalpie_imposee champ_front_fonc_txyz 1 1.193e5-400*(x_gt_0.31)*(x_lt_0.32)
    Entree_branche_froide frontiere_ouverte_enthalpie_imposee champ_front_fonc_txyz 1 1.185e5+400*(x_gt_0.31)*(x_lt_0.32)
  }
  # Source terms and an example of a source term read in a file #
  sources { source_derive , source_scalaire champ_fonc_med last_time sourceHgros.med MAILLAGE_GENEPI scalar_1 elem 0.0 }
  # It is possible to change the default parameters of the implicit scheme in case of slow convergence of one equation #
  parametre_equation parametre_implicite {
    solveur gmres { impr seuil 0.0001 diag controle_residu 1 } # controle_residu is a parameter to check the residual do not increase suddenly #
  }
}
postraitement
{
  # Probes to monitor some fields, here on the cell 0 of the master process #
  sondes {
    sonde_hsat enthalpie_saturation_liquide periode 1e-06 numero_elem_sur_maitre 0
    sonde_L chaleur_latente_melange periode 1e-06 numero_elem_sur_maitre 0
  }
  format lata champs binaire dt_post 1e-0 # binaire is useless cause it is now the default for LATA output format #
  {
    pression elem
    vitesse som
    enthalpie som
    taux_de_vide_melange elem
  }
}
}
```

Resoudre pb

# Genepi+ data file (4/4)

*# File plaques.data #*

```
{
  {
    description OBSTACLE1
    champ_aire champ_fonc_med last_time frt-singulier.med MAILLAGE_GENEPI AIRE_OBSTACLE_1 elem 0.
    transpose_rotation champ_rotation champ_uniforme 9
    -1. 0. 0.
    0. 1. 0.
    0. 0. -1.
  }
  ,
  {
    description OBSTACLE2
    champ_aire champ_fonc_med last_time frt-singulier.med MAILLAGE_GENEPI AIRE_OBSTACLE_2 elem 0.
    transpose_rotation champ_rotation champ_uniforme 9
    1. -0. 0.
    0. 1. -0.
    0. 0. 1.
  }
  ,
  {
    description OBSTACLE3
    champ_aire champ_fonc_med last_time frt-singulier.med MAILLAGE_GENEPI AIRE_OBSTACLE_3 elem 0.
    transpose_rotation champ_rotation champ_uniforme 9
    0. -1. 0.
    1. 0. -0.
    0. 0. 1.
  }
}
```

# Genepi+ : PPGP example

---

- Launch PPGP:

```
> /data/tmplgls/goubioud/PPGP_1.3_FED18_64/ppgp_appli
```

- Select PPGP module.
- Click on New button to create new study.
- "Case" → "New case" → "From Standard Case".
- Choose **NR\_maxibav30\_TR411** case from list of standard cases.
- Enter "NR\_maxibav30\_TR411" name, adjust path to "/tmp/NR\_maxibav30\_TR411" and click "OK".
- Expand "PPGP", then expand "NR\_maxibav30\_TR411" object and "PreProcessing" and "Processing" sub-objects in object browser.
- Select "NR\_maxibav30\_TR411/PreProcessing", right click to call its popup menu and click "Generate meshes" button.
- Click "Compute meshes" button, calculation starts and finishes some time later successfully.
- Click "Create MED files" button.
- Check that export MED files finished.
- Click "compute intersection" button.

# Genepi+ : PPGP example

---

- Close "Generate meshes and intersections" dialog box.
- Expand "NR\_maxibav30\_TR411/PreProcessing/Obstacles", show each of them.
- Expand "NR\_maxibav30\_TR411/Processing/Processing\_1" and select "LaunchAndMonitoring".
- In property panel, click on "Start Solver" button.
- Right click to open context menu and select "Show in paravis".
- In central view, right click on the object and select "Color By" → "TEMPERATURE\_GAZ\_SOM\_SOM\_dom".
- Activate "Paravis" module by selecting "Paravis" in main toolbar.
- In PPGP toolbar, change:
  - visualised field by selecting "RHO\_U\_SOM\_dom" instead of "TEMPERATURE\_GAZ\_SOM\_SOM\_dom".
  - representation mode by selecting "Wireframe" instead of "Surface".

# Table of contents

---

- TRUST/TrioCFD historic
- Modeling flow with TRUST/TrioCFD
- Examples of performed calculations
- Models, schemes, numerical methods
- Data files & calculation
- Command lines
- Parallel calculation
- Mesh generators: Internal tools & Salomé & Gmsh
- Automating validation test case
- TRUST/TrioCFD support
- Examples of data files
- **Recommendations**

# Recommendations (1/6)

---

## cf **Best\_Practice\_TRUST.pdf**

- **For VEF meshing.....** p168
- For conditions..... p171
- For schemes..... p173
- For turbulence..... p180
- For post processing..... p186
- What if?..... p188



# Recommendations for VEF meshing (1/2)

---

- Mesh refinement
  - Minimum of 10 points between two walls
  - To avoid very small time steps, do not create small cells in high velocity regions
- Boundary layers
  - Quality of the mesh near boundary layer will be improved if 2 or 3 layers of regular cells is used near the wall
- Tetra general mesh recommendations
  - Avoid significant changes in the mesh size of two adjacent tetras. A propagation factor of 20% seems to be the upper limit (5% recommended)

# Recommendations for VEF meshing (2/2)

- Tetra general mesh recommendations:
  - Avoid the use of a great amount of obtuse cells (angles between faces greater than  $90^\circ$ ). In the contrary, non physical phenomena are observed to the diffusion operator.
    - Use any optimization tools of mesh generators to reduce too large angles
    - Check in the .err file of TRUST, the angles histogram of the mesh
    - Visualize the field (mesh quality) named LargestAngle\_elem
  - Use isotropic cells in all directions as much as possible for a 3D flow. In the case of a flow in a given direction (1D flow), the stretching of a mesh up to a hundred times in the preferred direction nevertheless doesn't much degrade the results
  - Look the Meshing paragraph of the Best Practice Guide for more advices

# Recommendations (2/6)

---

- For VEF meshing
- **For conditions**
- For schemes
- For turbulence
- For post processing
- What if?

# Recommendations for conditions (1/1)

---

- Boundary conditions

- Wall conditions for RANS model (no rule for the LES model):
  - The first discretization point should be in the logarithmic layer ( $30 < y^+ < 500$ ).  $y^+$  field may be post-processed to check this condition.
- Look for paragraph “Boundary conditions” in Best Practice Guide

- Initial conditions

- Look for paragraph “Initial conditions” of the Best Practice Guide to optimize the flow initialization in order to improve/reach faster the convergence

# Recommendations (3/6)

---

- For VEF meshing
- For conditions
- **For schemes**
- For turbulence
- For post processing
- What if?

# Recommendations for schemes (1/6)

---

## Space scheme

- Convection schemes with VEF discretization
  - EF\_stab
    - Use it for Navier Stokes and scalar equations (temperature, concentration)
    - Option  $\alpha$  is a compromise between:
      - robustness ( $\alpha$  near 1, the default value)
      - accuracy (smaller value for  $\alpha$ )
    - $\alpha=0.2$  gives the better results. When the mesh is stretched in one direction, this scheme may have some convergence issues.
    - $\alpha=1$  insures the stability and gives the same results for a lot of cases when the forced convection flow has a preferred direction.
  - Muscl
    - Use it for very perturbed or mixing flows
- Convection schemes with VDF discretization
  - Quick

# Recommendations for schemes (2/6)

---

## Explicit time scheme

Time step  $\Delta t = \text{facsec} * \Delta t_{\text{CFL}}$  (CFL condition: Courant-Friedrichs-Lewy)

- Euler explicit scheme:

$\text{facsec} \leq 1$

- Runge Kutta schemes (facsec limit may be increased):

$\text{facsec} = 2$  if RK2

$\text{facsec} = 3$  if RK3

$\text{facsec} = 4$  if RK4

# Recommendations for schemes (3/6)

---

## Implicit time scheme

### When to use it?

- To solve a steady state calculation (e.g. k- $\epsilon$  simulation)
- To reduce the time of the transient state of a calculation before switching to an explicit time scheme

**WARNING:** An unsteady calculation solved with an implicit scheme may give non physical results !

### How to use it?

- Define upper and lower limits of the dynamic time step algorithm
- Define a solver (**GMRES**, **BiCGSTAB**,...)
- Define the convergence criterias



# Recommendations for schemes (4/6)

---

## Implicit time scheme

- Dynamic time step algorithm:

Time step  $dt = \text{facsec}(t) * dt(\text{CFL condition})$

with :

$\text{facsec}(0) = \text{facsec}$  (lower limit keyword)

$\text{facsec}(t^{n+1}) = a * \text{facsec}(t^n)$

$\text{facsec}(t) \leq \text{facsec\_max}$  (upper limit keyword)

The algorithm uses  $a=1.2$  and reduces it if necessary

# Recommendations for schemes (5/6)

---

- In a first approach, use **facsec=facsec\_max** with:
  - **facsec\_max=20-30** for:
    - hydraulic calculation only
    - thermal hydraulic with forced convection and no or low coupling between velocity and temperature
  - **facsec\_max=90-100** for:
    - thermal hydraulic with forced convection and a strong coupling between velocity and temperature
  - **facsec\_max=300** for:
    - thermal hydraulic with natural convection
  - much higher value is possible for:
    - conduction calculation
- In a second approach (no convergence):
  - keep **facsec\_max** and reduce **facsec** to 5 or 10

# Recommendations for schemes (6/6)

---

- It is possible to specify a solver for each equation:
  - Momentum equations
    - **GMRES** solver generally works well
  - Scalar equations
    - **BICGSTAB/ILU** if **GMRES** converges slowly or not at all
  - Transport k- $\epsilon$  equations
    - **GMRES** solver and if no convergence (rare), solve these equations with an explicit scheme
- Convergence criteria's
  - Threshold convergence (keyword **seuil**) of the iterative solver (to have at least 3 iterations)
  - A maximal number of iterations may be specified with **nb\_it\_max** keyword (5 is a good value)
  - For coupled problems, another parameter:
    - keyword **seuil\_convergence\_implicite** should be set to 0.001(or less) for strongly coupled problems to guarantee the correct coupling of various equations as e.g. hydraulic and thermal phenomena

# Recommendations (4/6)

---

- For VEF meshing
- For conditions
- For schemes
- **For turbulence**
- For post processing
- What if?

# Recommendations for turbulence (1/5)

Which turbulence models?

## Interested in averaged quantities

- k- $\epsilon$  standard model
  - Low cost but lack of generality

## Interested in fluctuating quantities

- LES model
  - Involves a refined 3D mesh
  - Recommended in the following cases

# Recommendations for turbulence (2/5)

---

## Which turbulence models?

LES model if:

- Strongly non stationary flow
  - Access to mechanical or thermal fluctuations
  - Mixing phenomena
  - Free and impinging jets
- Flow with important secondary structures
  - Tube bends
  - Rectangular channels at reduced Reynolds numbers
- Flow with detachment and/or reattachment
  - Backward facing steps
  - Obstacles
- Flow at slow regimes
  - Natural convection

# Recommendations for turbulence (3/5)

## RANS calculation

- Model adapted to high Reynolds number
- First discretization point of the mesh should be located in the logarithmic zone ( $y^+ \sim 30$ )
- Taking care to the initial and boundary conditions for  $k$  and  $\varepsilon$  will avoid some issues and dramatically improve convergence
  - In a first approach (for example 10% turbulence rate), you can use ( $U$ , bulk velocity and  $L$ , a relevant dimension, example a pipe diameter):
$$k \sim 3/2 * (\text{turbulence\_rate} * U)^2$$
$$\varepsilon \sim k^{1.5} / L$$
- Use EF\_stab scheme for the Navier Stokes equation
- Use upwind scheme for  $k$ - $\varepsilon$  equation

# Recommendations for turbulence (4/5)

---

## LES calculation

- Time schemes:
  - High order explicit schemes like:
    - Runge Kutta order 3 (ut facsec 1.0 for LES)
    - Adams Bashforth order 2
- Convection schemes:
  - VEF:
    - EF\_stab with  $\alpha=0.2$  for Navier Stokes equation (do not use first order scheme for a LES, so if you need to set  $\alpha=1.0$  to insure convergence, then something is wrong with your mesh)
  - VDF:
    - Centre (order 4) and if unstable use facsec=0.2



# Recommendations for turbulence (5/5)

---

## LES calculation

- Use a periodic box to provide a turbulent velocity field (see page 54):
  - The length of the periodic box should be at least  $8 \cdot D_h$  ( $D_h$ : hydraulic diameter)
  - Do not stretch the mesh in the axial direction of the box
- Models
  - First approach, use Wale model, then check the turbulent energy spectra
  - If Wale is unable to dissipate sufficiently the high frequencies, use the Smagorinsky model (see “Post processing” paragraph of the Best Practice Guide)

# Recommendations (5/6)

---

- For VEF meshing
- For conditions
- For schemes
- For turbulence
- **For post processing**
- What if?

# Recommendations for post processing (1/1)

- Fields can be visualized at:
  - element, vertex, or face
- Prefer visualization without linear interpolation
- The fields visualized without linear interpolation are :
  - At elements (keyword elem)
    - pressure, turbulent viscosity,  $y^+$ , physical properties and for VDF only, temperature, concentration,  $k$ ,  $\epsilon$
  - At vertexes (keyword som)
    - pressure
  - At faces (keyword faces)
    - velocity and for VEF only, temperature, concentration,  $k$ ,  $\epsilon$
- Prefer the LATA format which can be converted to any other TRUST format

# Recommendations (6/6)

---

- For VEF meshing
- For conditions
- For schemes
- For turbulence
- For post processing
- **What if?**

# Recommendations : What if? (1/1)

---

The calculation does not converge...

- Symptoms:
  - Time-step decreases to the **dt\_min** lower limit
  - wall law does not converge (error message)
  - implicit diffusion algorithm does not converge (warning message)
  - ...
- Try:
  - Reduce the time-step (use **facsec** value 0.5 or 0.2) if using an **explicit** scheme
  - Reduce the upper limit of the time-step (reduce **facsec\_max**) if using an **implicit** scheme
  - If using the **EF\_stab** scheme, try to increase  $\alpha$  value incrementally (0.2 to 0.3 or 0.5 or 1.0) to recover stability or reduce the **facsec** (if no change, switch to muscl scheme)
  - If using the **centre** scheme, reduce the **facsec** or change your scheme (**quick** scheme)
  - Contact Trio\_U support