EDF R&D

FLUID DYNAMICS, POWER GENERATION AND ENVIRONMENT DEPARTMENT
SINGLE PHASE THERMAL-HYDRAULICS GROUP

6, QUAI WATIER
F-78401 CHATOU CEDEX

TEL: 33 1 30 87 75 40
FAX: 33 1 30 87 79 16

MARCH 2024

code_saturne documentation

code_saturne version 8.0 tutorial: full domain

contact: saturne-support@edf.fr

http://code-saturne.org/

TABLE OF CONTENTS

I Introduction 5

1 Introduction . . . 6

1.1 code_saturne short presentation . . . 6

1.2 About this document . . . 6

1.3 code_saturne copyright informations . . . 6

II Full domain 7

1 Study description . . . 8

1.1 Objective . . . 8

1.2 Description of the configuration . . . 8

1.3 Characteristics . . . 8

1.4 Mesh characteristics . . . 9

1.5 Summary of the different calculations . . . 9

2 CASE 1: Passive scalar with various boundary conditions and output management 10

2.1 Calculation options . . . 10

2.2 Initial and boundary conditions . . . 11

2.3 Parameters . . . 12

2.4 Output management . . . 13

2.5 Results . . . 13

3 CASE 2: Time dependent boundary conditions and variable fluid density . . . 16

3.1 Calculation options . . . 16

3.2 Initial and boundary conditions . . . 16

3.3 Variable Density . . . 17

3.4 Parameters . . . 17

3.5 Output management . . . 18

3.6 Calculation restart . . . 18

3.7 Results . . . 18

4 CASE 2 (BIS): Time dependent boundary conditions using "Time Tables" management . . . 23

4.1 Calculation options . . . 23

4.2 Initial and boundary conditions . . . 23

5 CASE 3: Head losses and parallelism . . . 25

5.1 Calculation options . . . 25

5.2 Initial and boundary conditions . . . 25

5.3 Variable Density . . . 26

5.4 Head losses . . . 26

5.5 Parameters . . . 27

5.6 Output management . . . 27

5.7 Results . . . 27

III Step by step solution **30**

1 Solution for CASE1 . . . 31

1.1 Mesh tab . . . 31

1.2 Calculation features tab . . . 35

1.3 Volume conditions tab . . . 38

1.4 Boundary conditions tab . . . 41

1.5 Time settings tab . . . 45

1.6 Numerical Parameters tab . . . 47

1.7 Postprocessing tab . . . 49

1.8 Run calculation . . . 54

2 Solution for CASE2 . . . 55

2.1 Volume tab . . . 55

2.2 Boundary condition tab . . . 57

2.3 Postprocessing . . . 59

2.4 Time settings . . . 59

3 Solution for CASE3 . . . 63

3.1 Mesh tab . . . 63

3.2 Volume conditions tab . . . 64

3.3 Time settings . . . 64

3.4 Run calculation . . . 66

Part I

Introduction

1 Introduction

1.1 code_saturne short presentation

code_saturne is a system designed to solve the Navier-Stokes equations in the cases of 2D, 2D axisymmetric or 3D flows. Its main module is designed for the simulation of flows which may be steady or unsteady, laminar or turbulent, incompressible or potentially dilatable, isothermal or not. Scalars and turbulent fluctuations of scalars can be taken into account. The code includes specific modules, referred to as "specific physics", for the treatment of lagrangian particle tracking, semi-transparent radiative transfer, gas, pulverized coal and heavy fuel oil combustion, electricity effects (Joule effect and electric arcs) and compressible flows. code_saturne relies on a finite volume discretization and allows the use of various mesh types which may be hybrid (containing several kinds of elements) and may have structural non-conformities (hanging nodes).

1.2 About this document

The present document is a tutorial for code_saturne version 8.0. It presents three simple test cases and guides the future code_saturne user step by step into the preparation and the computation of the cases.

The test case directories, containing the necessary meshes and data are available in the `saturne-tutorials` git base.

This tutorial focuses on the procedure and the preparation of the code_saturne computations. For more elements on the structure of the code and the definition of the different variables, it is higly recommended to refer to the user manual.

1.3 code_saturne copyright informations

Part II

Full domain

1 Study description

1.1 Objective

The aim of this case is to tackle the merging of initially separate meshes into a single fluid domain. The questions of mesh joining and hanging nodes will be addressed. The test case will then be used to present more complex calculations, with time dependent variables and boundary conditions.

1.2 Description of the configuration

The fluid domain is composed of three separate meshes, very roughly representing elements of a nuclear pressurized water reactor vessel:

- the downcomer
- the vessel's bottom
- the lower core plate and core

Figure II.1 represents the complete domain. The flow circulates from the top left horizontal junction to the right vertical outlet.

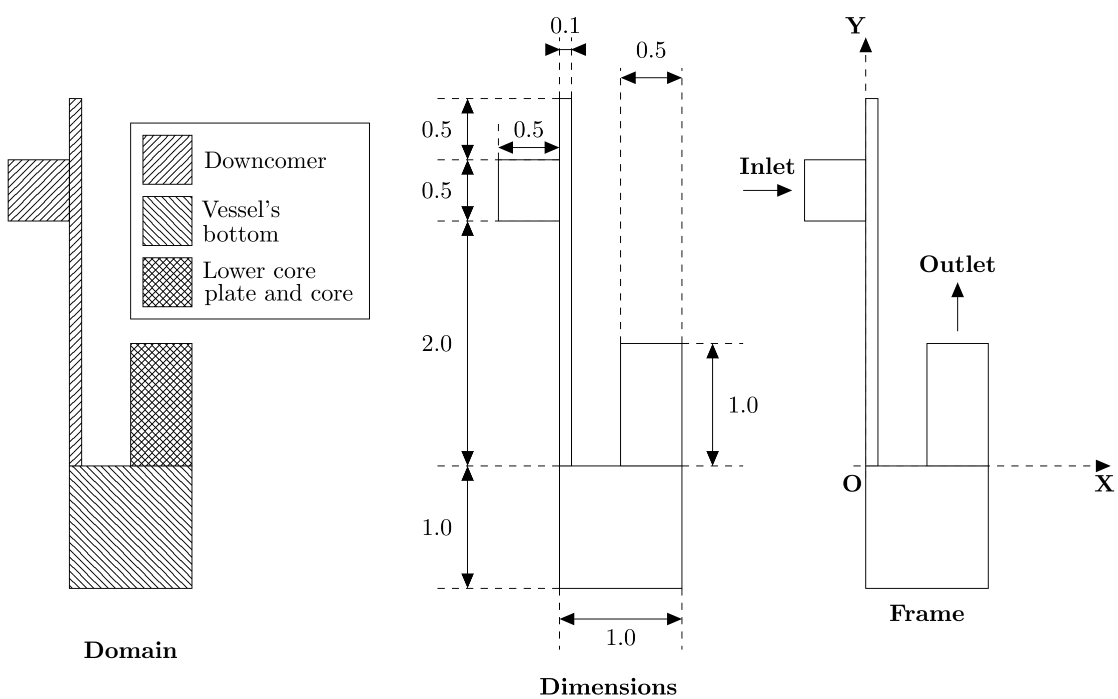


Figure II.1: Geometry of the complete domain

1.3 Characteristics

Characteristics of the geometry and the flow:

Height of downcomer	$H = 3.00\ m$
Thickness of downcomer	$E_d = 0.10\ m$
Diameter of the inlet cold branch	$D_b = 0.50\ m$
Height of vessel's bottom	$H_{fc} = 1.00\ m$
Width of vessel's bottom	$l_{fc} = 1.00\ m$
Height of core above the lower core plate	$H_{pic} = 1.00\ m$
Width of core above the lower core plate	$l_{pic} = 0.50\ m$
Inlet velocity of fluid	$V = 1\ m.s^{-1}$

Table II.1: Characteristics of the geometry and the flow

Physical characteristics of fluid:

The initial water temperature in the domain is equal to 20°C. The inlet temperature of water in the cold branch is 300°C. Water characteristics are considered constant[1] and their values taken at 300°C and $150 \times 10^5\ Pa$, except density which is considered variable in `case2` and `case3`:

- density: $\rho = 725.735\ kg.m^{-3}$
- dynamic viscosity: $\mu = 0.895 \times 10^{-4}\ kg.m^{-1}.s^{-1} = 8.951 \times 10^{-5}\ Pa.s$
- heat capacity: $C_p = 5\,483\ J.kg^{-1}.°\mathrm{C}^{-1}$
- thermal conductivity $= 0.02495\ W.m^{-1}.K^{-1}$

1.4 Mesh characteristics

Figure II.2 shows a global view of the mesh and some details of the joining zones, to show that code_saturne can deal with hanging nodes. This mesh is composed of 1 650 cells, which is very small compared to those used in real studies. This is a deliberate choice so that tutorial calculations run fast.

Type: block structured mesh

Coordinates system: cartesian, origin on the edge of the main pipe at the outlet level, on the nozzle side (figure II.2)

Meshing strategy: load existing meshes and join them with the Preprocessor of code_saturne (in order to deal with hanging nodes)

Groups names: see figure II.3. Note that the naming of the boundary faces for walls uses the first letter of the 3 parts (D for Downcomer, V for Vessel's bottom and C for lower Core plate). In the same way, all groups used for symmetry or fusion end up respectively with `Sym` and `Fuse`.

1.5 Summary of the different calculations

Three cases will be studied with this geometry. The following table gives a summary of their different characteristics.

Remark: In this case, you must add three meshes which have to be joined. In order to join the three meshes, you must add a selection criteria in the box **Selection criteria** under the **Preprocessing** sub-folder. In this case, only faces in groups DFuse, VFuse and CFuse are liable to be joined (different group names can be entered on a single line, separated by comma).

[1] Which makes temperature a passive scalar ... but it is only for simplification purposes.

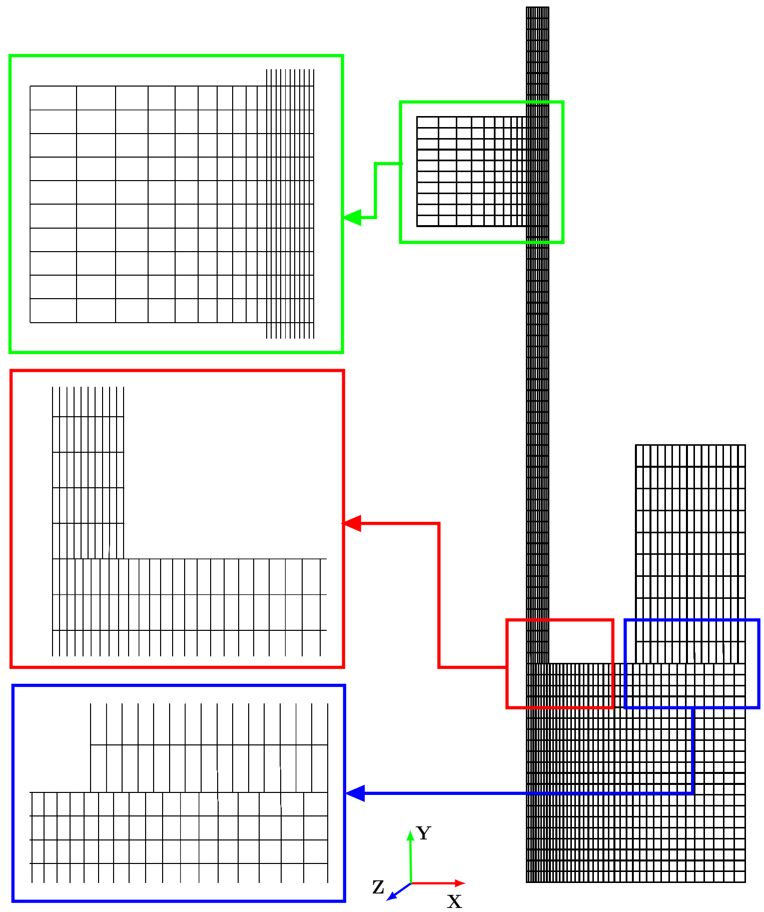


Figure II.2: View of the full domain mesh with zoom on the joining regions

CASE	Characteristics
CASE 1	Unsteady flow, additionnal passive scalar, output management
CASE 2	Same as case 1 with time dependent boundary conditions, fluid density depending on the temperature and calculation restart
CASE 2 (BIS)	Same as case 2 with time dependent boundary conditions, but using "Time tables" management as boundary condition
CASE 3	Same as case 2 with head losses and parallelism

Table II.2: Summary of the different calculations

You can verify the quality of your mesh by running a **Mesh quality criteria only** computation, which you can access through **Execution mode** in the **Mesh** heading.

2 CASE 1: Passive scalar with various boundary conditions and output management

2.1 Calculation options

Some options are similar to those of the `simple_junction` tutorial:

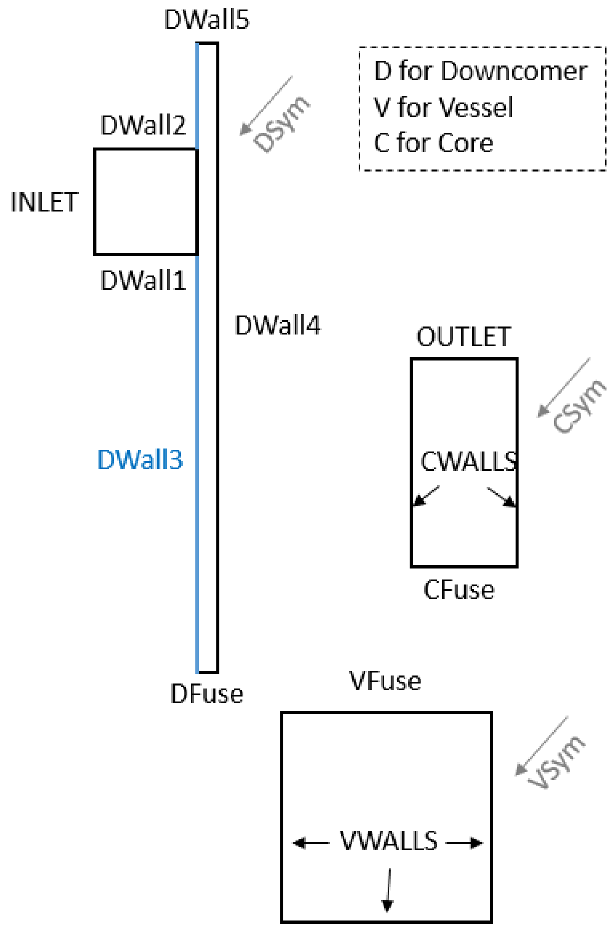


Figure II.3: Groups of the boundary faces

→ Turbulence model: $k - \varepsilon$ LP (Linear Production)

→ Temperature activated with no gravity (acts like a passive scalar)

→ Physical properties: uniform and constant

The additional options are:

→ Flow type: unsteady flow

→ Time step: uniform and constant

→ One additional passive scalar named `tracer` [2], with diffusion coefficient 8.55 ($\times 10^{-5}$ $m^2.s^{-1}$)

→ Management of monitoring points

2.2 Initial and boundary conditions

→ Initialization: 20°C for temperature
10 for the passive scalar `tracer`

[2] It could correspond to a tracer concentration for instance.

The boundary conditions are defined in the user interface and depend on the boundary zone.

- **Flow inlet**: Dirichlet condition, an inlet velocity of 1 $m.s^{-1}$, an inlet temperature of 300°C and an inlet value of 200 for the passive scalar are imposed
- **Outlet**: default value
- **Walls**: velocity, pressure and thermal scalar: default value
  passive scalar: different conditions depending on the group and geometric parameters (see table below)

In order to test the ability to specify boundary condition regions in the Graphical Interface, various conditions will be imposed for the passive scalar `tracer`, as specified in the following table:

Wall	Nature	Value
wall_1	Imposed value (Dirichlet)	0
wall_2	Imposed value (Dirichlet)	5
wall_3	Imposed value (Dirichlet)	0
wall_4	Imposed value (Dirichlet)	25
wall_5	Imposed value (Dirichlet)	320
wall_6	Imposed value (Dirichlet)	40

The **wall_1** to **wall_6** regions are defined as follows, through group names and geometric localization:

Label	Names of Groups and geometric parameters
wall_1	VFuse and $0.1 \leqslant x$ and $x \leqslant 0.5$
wall_2	DWALL1 or DWALL2
wall_3	DWALL3 or DWALL5 or VWalls
wall_4	DWALL4 and $y > 1$
wall_5	DWALL4 and $y \leqslant 1$
wall_6	CWalls

Figure II.3 shows the group's name used for boundary conditions and table II.3 defines the correspondance between list of groups and the type of boundary condition to use.

Groups	Conditions
INLET	Inlet
OUTLET	Outlet
DWALL1 DWALL2 DWALL3 DWALL4 DWALL5 VWalls CWalls	Wall
DSym VSym CSym	Symmetry

Table II.3: Boundary faces groups and associated conditions

2.3 Parameters

All parameters necessary to this study can be defined through the Graphical Interface without using any user source files.

In order to join the separate meshes into a single domain, groups DFuse, VFuse and CFuse will have to be joined through the Graphical Interface.

Calculation control parameters	
Pressure-Velocity coupling	SIMPLEC algorithm
Number of iterations	300
Reference time step	0.05
Output period for post-processing files	2

2.4 Output management

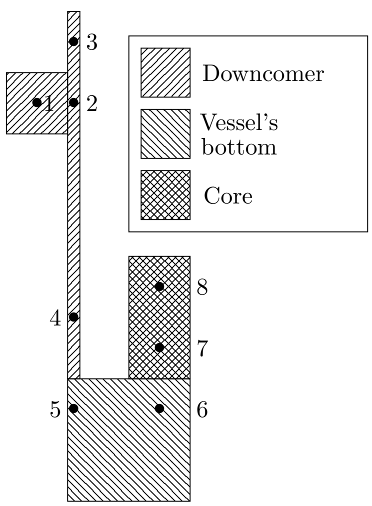
In this case, different aspects of output management will be addressed.

By default, in the Graphical Interface, all variables are set to appear in the listing, the post-processing and the chronological records. This default choice can be modified by the user.

In this case, the **Pressure**, the **Turbulent energy** and the **Dissipation** will be removed from the listing file.

The **Courant number** (CFL) and **Fourier number** will be removed from the post-processing results[3].

Eventually, probes will be defined for chronological records, following the data given in figure II.4. Then the **total pressure** will be deactivated for all probes.



Probe n^o.	x (m)	y (m)	z (m)
1	-0.25	2.25	0
2	0.05	2.25	0
3	0.05	2.75	0
4	0.05	0.5	0
5	0.05	-0.25	0
6	0.75	-0.25	0
7	0.75	0.25	0
8	0.75	0.75	0

Figure II.4: Position and coordinates of probes in the full domain

In addition the domain boundary will be post-processed. This allows to check the boundary conditions, and especially that of the passive scalar.

2.5 Results

Figure II.5 shows the boundary domain colored by the passive scalar boundary conditions. The different regions of boundary conditions defined earlier can be checked.

Figure II.6 presents results obtained at different times of the calculation. They were plotted from the post-processing files, with ParaView.

[3]This can be very useful to save some disk space if some variables are of no interest, as post-processing files can be large.

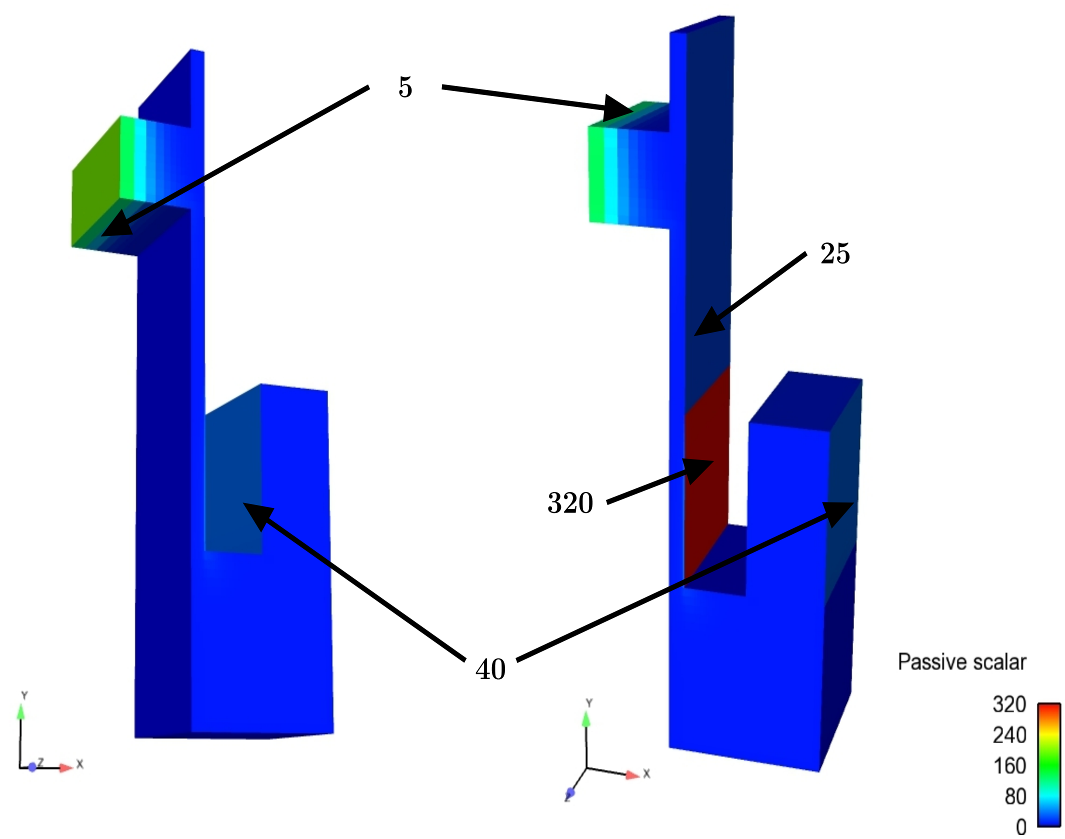


Figure II.5: View of the boundary domain colored by the scalar2 variable - Case 1

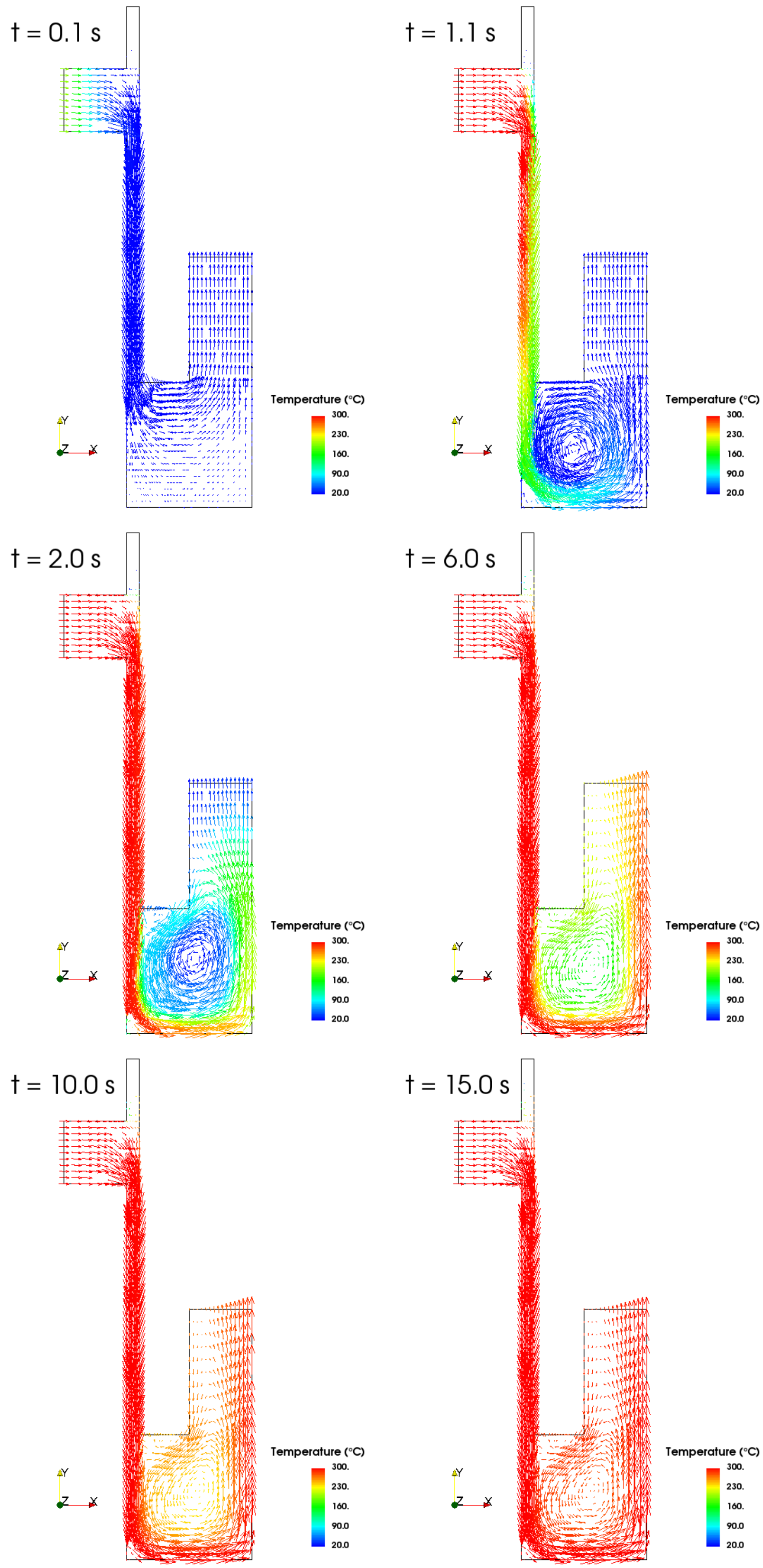


Figure II.6: Water velocity field colored by temperature at different time steps - Case 1

3 CASE 2: Time dependent boundary conditions and variable fluid density

In this case some boundary conditions will be time dependent and some physical characteristics of the fluid will be dependent on the temperature.

Remark: You can copy your `case1` in order to make the `case2`:

```
$ code_saturne create --copy-from case1 case2
```

3.1 Calculation options

The options for this case are the same as in `case1`, except for the variable fluid density and the gravity magnitude:

→ Flow type: unsteady flow

→ Time step: uniform and constant

→ Turbulence model: $k-\varepsilon$ LP (Linear Production)

→ Temperature is activated (no more passive scalar as gravity will be non zero)

→ One passive scalar `tracer`

→ Physical properties: uniform and constant (except density)

→ Management of monitoring points

3.2 Initial and boundary conditions

→ Initialization: 20°C for temperature
10 for the passive scalar `tracer`

The boundary conditions are defined in the Graphical User Interface (GUI) and depend on the boundary zone. The time dependence of the temperature boundary condition can also be setted directly in the GUI.

- **Flow inlet**: Dirichlet condition, an inlet velocity of 1 $m.s^{-1}$, a time dependent inlet temperature and a value of 200 for the passive scalar `tracer` are imposed;
- **Outlet**: default value;
- **Walls**: velocity, pressure and thermal scalar: default value
passive scalar `tracer`: different conditions depending on the groups and geometric parameters (see table below).

The boundary condition at the inlet is now variable in time, following the law:

$$\begin{cases} T = 20 + 100t & \text{for } 0 \leqslant t \leqslant 3.8 \\ T = 400 & \text{for } t > 3.8 \end{cases} \tag{II.1}$$

where T is the temperature in °C and t is the time in seconds (s).

The formula can be setted directly in the GUI using the **Prescribed value (user law)** option.

The boundary conditions for the passive scalar are identical as those in `case1`, as specified in the following table:

Wall	Nature	Value
wall_1	Imposed value (Dirichlet)	0
wall_2	Imposed value (Dirichlet)	5
wall_3	Imposed value (Dirichlet)	0
wall_4	Imposed value (Dirichlet)	25
wall_5	Imposed value (Dirichlet)	320
wall_6	Imposed value (Dirichlet)	40

The **wall_1** to **wall_6** regions are defined as follows, through names of groups and geometric localization:

Label	Name of groups and geometric parameters
wall_1	VFuse and $0.1 \leqslant x$ and $x \leqslant 0.5$
wall_2	DWALL1 or DWALL2
wall_3	DWALL3 or DWALL5 or VWalls
wall_4	DWALL4 and $y > 1$
wall_5	DWALL4 and $y \leqslant 1$
wall_6	CWalls

Figure II.3 shows the group's name used for boundary conditions and table II.4 defines the correspondance between list of groups and the type of boundary condition to use.

Groups	Conditions
INLET	Inlet
OUTLET	Outlet
DWALL1 DWALL2 DWALL3 DWALL4 DWALL5 VWalls CWalls	Wall
VFuse for $0.1 \leq x \leq 0.5$	Wall
DSym VSym CSym	Symmetry

Table II.4: Boundary faces groups and associated references

3.3 Variable Density

In this case the density is a function of the temperature. The variation law is defined in the Graphical User Interface. The expression is:

$$\rho = T(AT + B) + C \tag{II.2}$$

where ρ is the density, T is the temperature, $A = -4.0668\times10^{-3}$, $B = -5.0754\times10^{-2}$ and $C = 1\,000.9$.

In order for the variable density to have an effect on the flow, gravity must be set to a non-zero value. $\underline{g} = -9.81\underline{e}_y$ will be specified in the Graphical Interface.

Remark:
The temperature is **temperature** in the user expression. Don't forget **;** at the end of the expression.

3.4 Parameters

The calculation parameters are identical as those in `case1`.

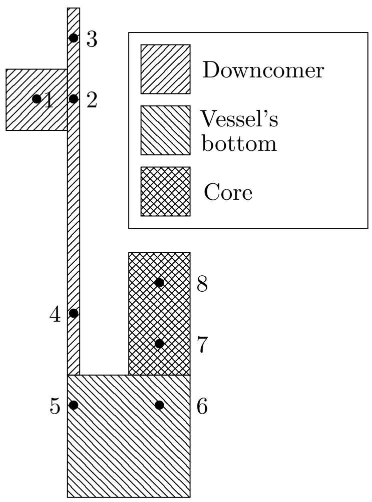
All the parameters necessary to this study can be defined through the Graphical Interface.

In order to join the separate meshes into a single domain, groups DFuse, VFuse and CFuse will have to be joined through the Graphical Interface.

Parameters of calculation control	
Number of iterations	300
Reference time step	0.05
Output period for post-processing files	2

3.5 Output management

The output management is the same as in `case1`, except that a nineth monitoring point will be added, just at the entry, to monitor the temperature evolution at inlet.



Probe n^o.	x (m)	y (m)	z (m)
1	-0.25	2.25	0
2	0.05	2.25	0
3	0.05	2.75	0
4	0.05	0.50	0
5	0.05	-0.25	0
6	0.75	-0.25	0
7	0.75	0.25	0
8	0.75	0.75	0
9	-0.5	2.25	0

Figure II.7: Position and coordinates of probes in the full domain

In this case, the **Pressure**, the **Tubulent Energy** and the **Dissipation** will be removed from the listing file.

The **Courant number** (CFL) and **Fourier number** will be removed from the post-processing results[4].

Eventually, probes will be defined for chronological records, following the data given in figure II.7. Then the **total_pressure** will be deactivated from all probes.

In addition the domain boundary will be post-processed. This allows to check the boundary conditions, and especially that of the temperature and passive scalar.

3.6 Calculation restart

After the first run, the calculation will be continued for another 400 time steps. The calculation restart is managed through the Graphical Interface.

3.7 Results

Figure II.8 shows the time evolution of temperature recorded on each monitoring probe. Note that the .csv files obtained for each case, were concatenated to plot the evolution of temperature over the entire period.

Figure II.9 shows the velocity fields colored by temperature in the first run of calculation.

Figure II.10 shows the velocity fields in the second calculation (restart of the first one).

[4]This can be very useful to save some disk space if some variables are of no interest, as post-processing files can be

large.

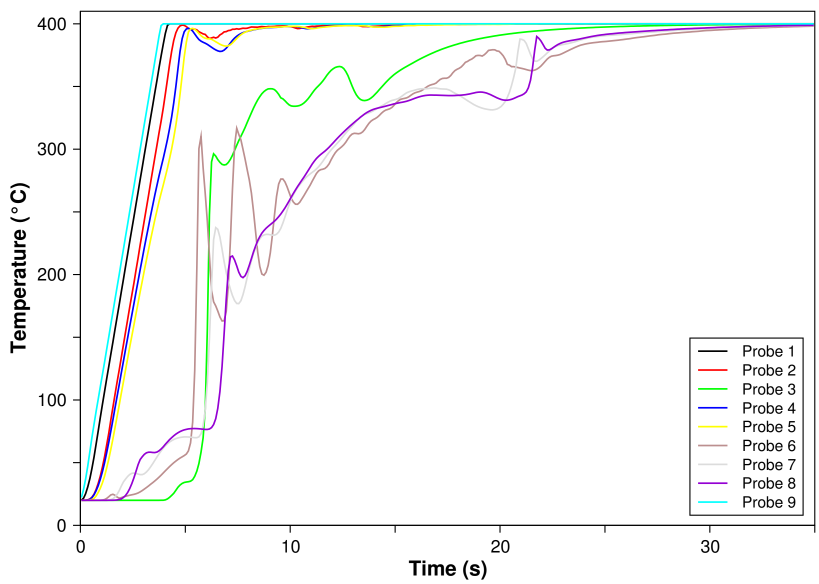


Figure II.8: Time evolution of temperature at monitoring probes - Case 2

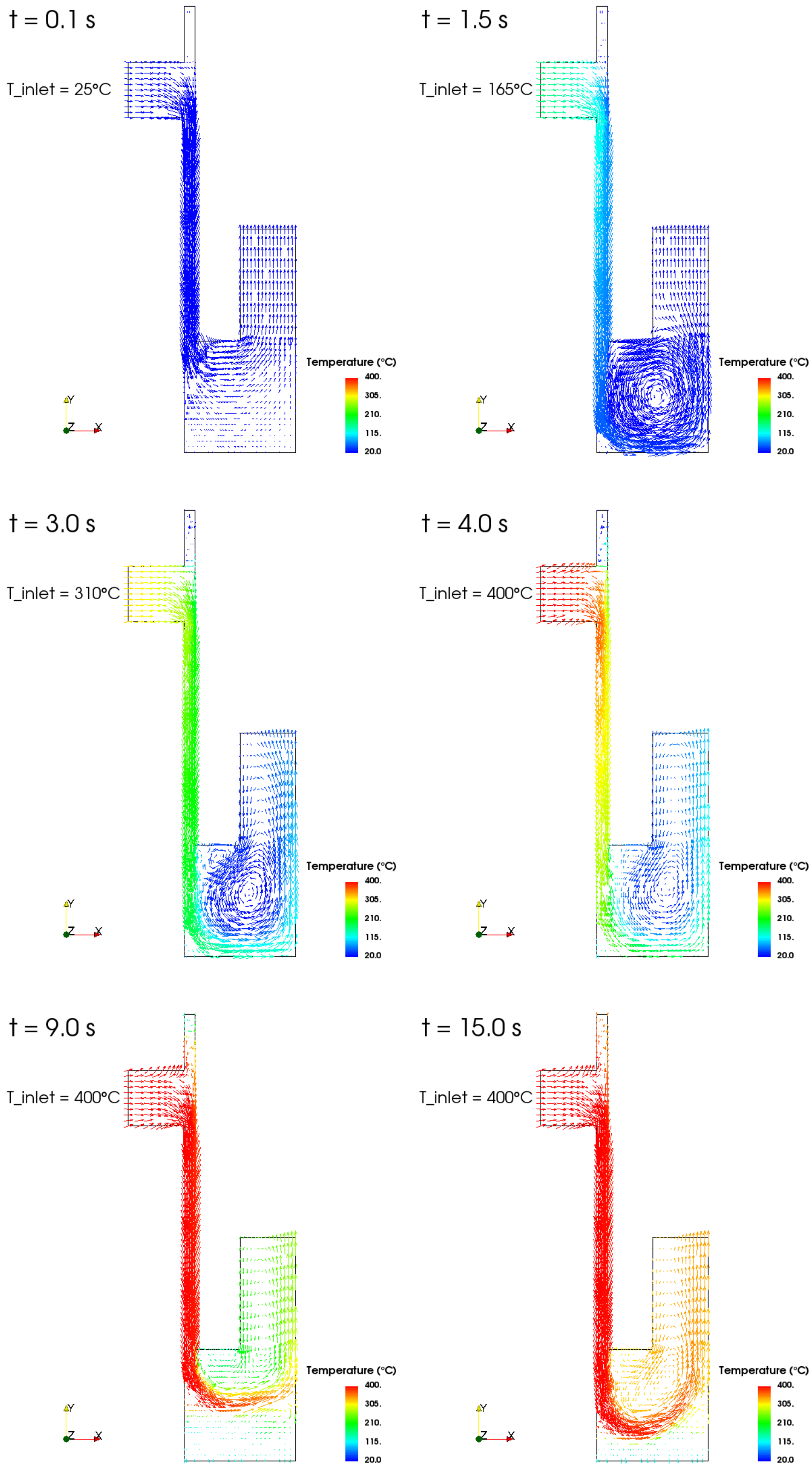


Figure II.9: Water velocity field colored by temperature and inlet temperature value at different time steps (first calculation) - Case 2

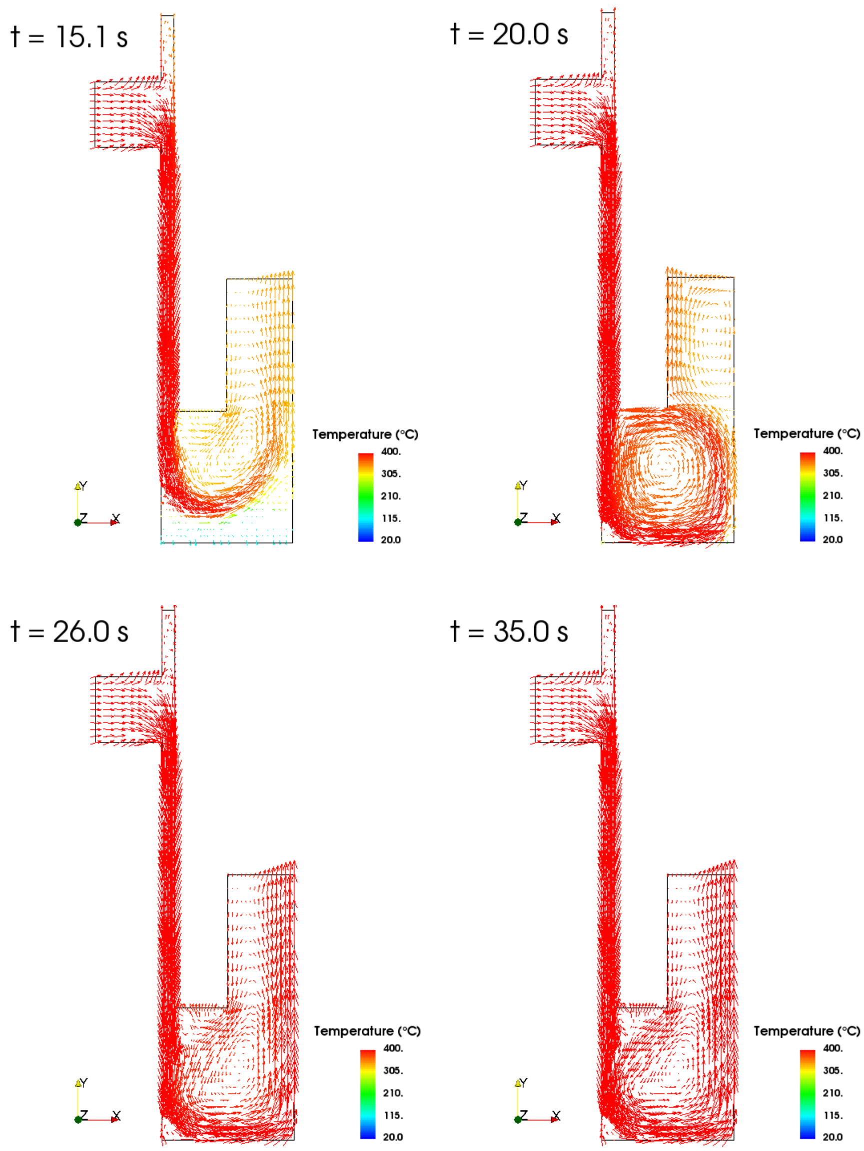


Figure II.10: Water velocity field colored by temperature at different time steps (second calculation) - Case 2

4 CASE 2 (BIS): Time dependent boundary conditions using "Time Tables" management

In this case the previous variable boundary condition for the temperature will no more be defined using analytical function of time but rather using an imput parameter file that give the law to be imposed. We thus will use the "Time tables" item in the GUI.

Remark: You can copy your `case2` in order to make the `case2_bis`:

```
$ code_saturne create case2_bis --copy-from=case2
```

4.1 Calculation options

The options for this case are the same as in `case2`.

4.2 Initial and boundary conditions

Initialization values are kept identical to case 2.

The boundary conditions are defined in the Graphical User Interface (GUI) and depend on the boundary zone. All boundary conditions are kept identical, except for the temperature at the inlet which will be set directly in the GUI using "Time tables" rather than using the analytical function of time.

To define a "Time table" click on the **Calculation environement** item and select the **Time tables** menu. From here simply load the file `temperature_data.csv` as follows :

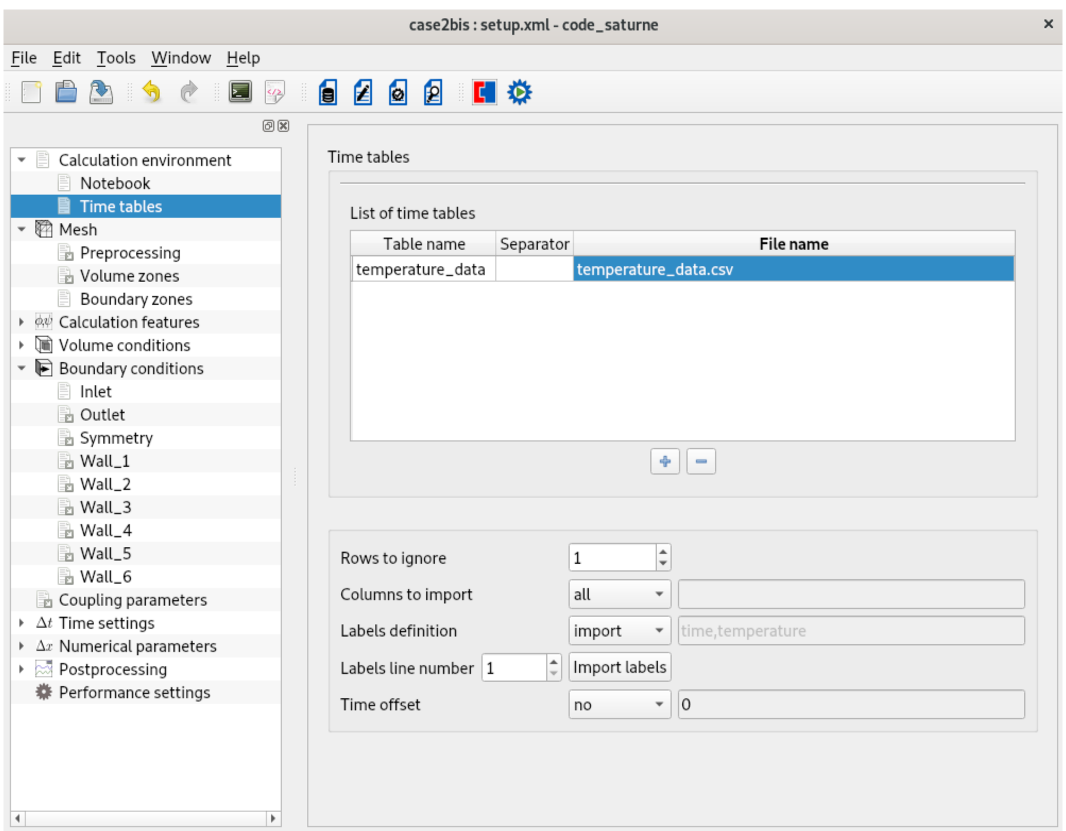


Figure II.11: View of the "Time table" menu - Case 2 (bis)

Once it has been properly loaded, you can defined the temperature at the inlet boundary condition as follows :

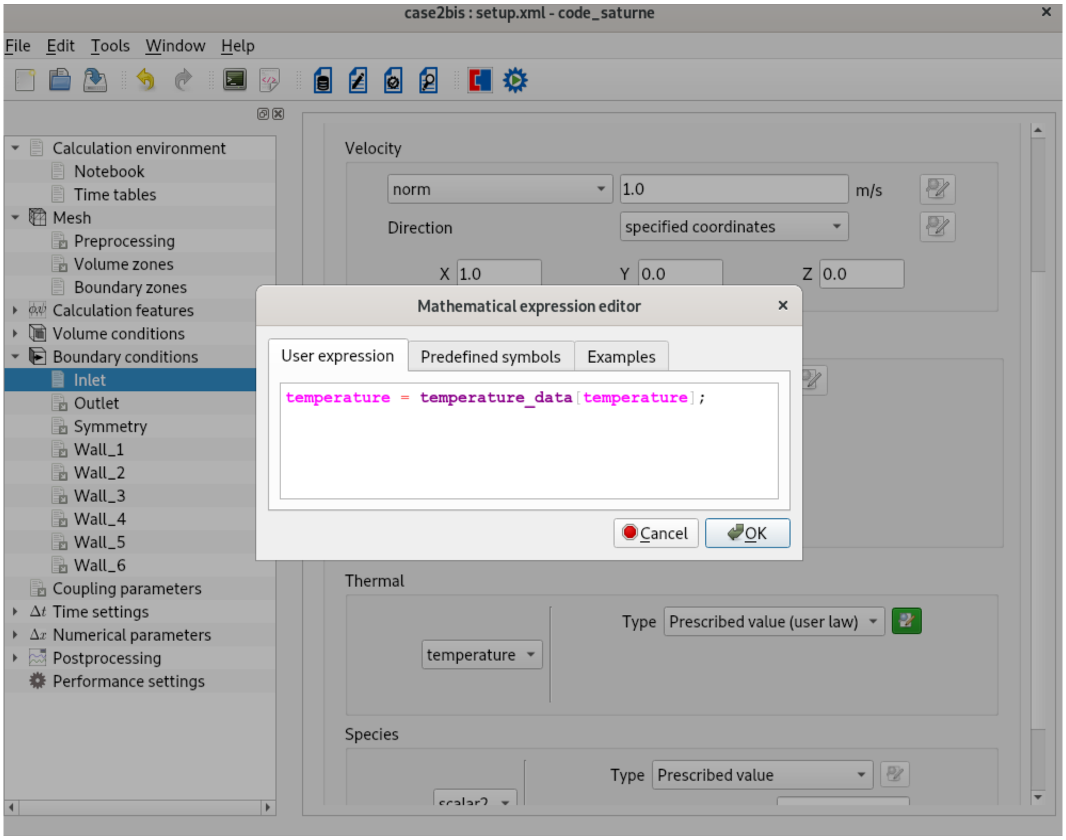


Figure II.12: View of the boundary condition definition for the temperature - Case 2 (bis)

This will ensure that the temperature follows what is defined in the file data. The interpolation between two points in the file is directly made by code_saturne.

5 CASE 3: Head losses and parallelism

This case will be run in parallel on two processors and head losses will be used to simulate the presence of an obstacle in the flow.

5.1 Calculation options

The options for this case are the same as in `case2`:

→ Flow type: unsteady flow

→ Time step: uniform and constant

→ Turbulence model: $k - \varepsilon$ LP (Linear Production)

→ Temperature is activated (no more passive scalar as gravity will be non zero)

→ One passive scalar `tracer` with diffusion coefficient 8.55 ($\times 10^{-5}$ $m^2.s^{-1}$)

→ Physical properties: uniform and constant (except density)

→ Management of monitoring points

5.2 Initial and boundary conditions

→ Initialization: 20°C for temperature
10 for the passive scalar `tracer`

The boundary conditions are defined in the user interface and depend on the boundary zone.

- **Flow inlet**: Dirichlet condition, an inlet velocity of 1 $m.s^{-1}$ and a time dependent inlet temperature and a value of 200 for the passive scalar `tracer` are imposed
- **Outlet**: default value
- **Walls**: velocity, pressure and thermal scalar: default value
passive scalar `tracer`: different conditions depending on the groups and geometric parameters (see table below).

The boundary conditions for the passive scalar are identical as those in `case2`, as specified in the following table:

Wall	Nature	Value
wall_1	Imposed value (Dirichlet)	0
wall_2	Imposed value (Dirichlet)	5
wall_3	Imposed value (Dirichlet)	0
wall_4	Imposed value (Dirichlet)	25
wall_5	Imposed value (Dirichlet)	320
wall_6	Imposed value (Dirichlet)	40

The **wall_1** to **wall_6** regions are defined as follows, through names of groups and geometric localization:

Figure II.3 shows the group's name used for boundary conditions and table II.4 defines the correspondance between list of groups and the type of boundary condition to use.

Label	Name of groups and geometric parameters
wall_1	VFuse and $0.1 \leqslant x$ and $x \leqslant 0.5$
wall_2	DWALL1 or DWALL2
wall_3	DWALL3 or DWALL5 or VWalls
wall_4	DWALL4 and $y > 1$
wall_5	DWALL4 and $y \leqslant 1$
wall_6	CWalls

Groups	Conditions
INLET	Inlet
OUTLET	Outlet
DWALL1 DWALL2 DWALL3 DWALL4 DWALL5 VWalls CWalls	Wall
VFuse for $0.1 \leq x \leq 0.5$	Wall
DSym VSym CSym	Symmetry

Table II.5: Boundary faces groups and associated references

5.3 Variable Density

The law for the variable density is identical as that in `case2`.

5.4 Head losses

To simulate the presence of an obstacle 0.20 (m) large and 0.5 (m) high in the vessel, a zone of head losses will be created in the domain (fig II.13).

The head losses zone is located between the coordinates $x = 0.2$ (m) and $x = 0.4$ (m), and $y = -0.75$ (m) and $y = -0.25$ (m).

The head losses coefficient to apply is $K_{11} = K_{22} = K_{33} = 10^4 = \frac{1}{2}\ \alpha_{11} = \frac{1}{2}\ \alpha_{22} = \frac{1}{2}\ \alpha_{33}$ and is isotropic.

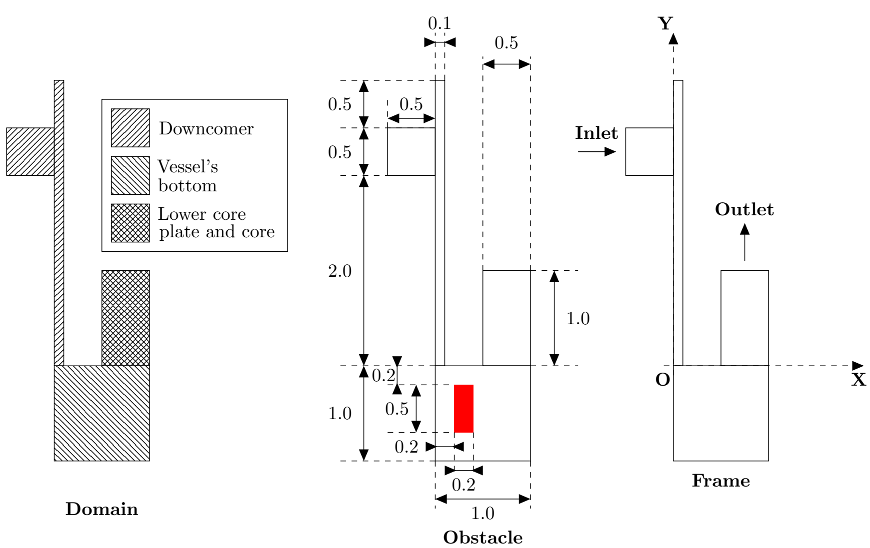


Figure II.13: Full domain geometry with the obstacle

5.5 Parameters

All the parameters necessary to this study can be defined through the Graphical Interface.

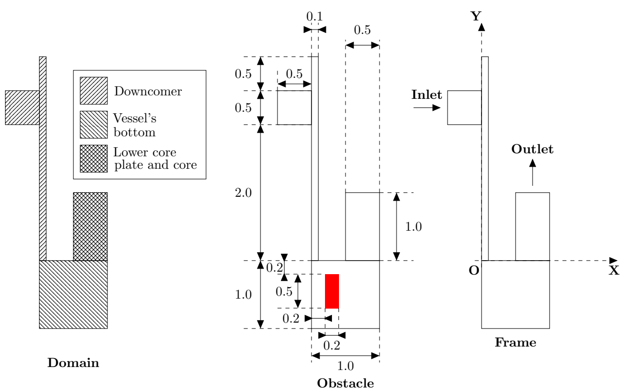
Parameters of calculation control	
Number of iterations	900
Reference time step	0.01
Output period for post-processing files	2
The calculation will be run in parallel	2 procs.

In order to join the separate meshes into a single domain, groups DFuse, VFuse and CFuse will have to be joined through the Graphical Interface.

Note that the time step has been reduced because of the head losses: the pressure step is more difficult to be solved in presence of head losses.

5.6 Output management

The output management is the same as in `case2`.



Probe n^o.	x (m)	y (m)	z (m)
1	-0.25	2.25	0
2	0.05	2.25	0
3	0.05	2.75	0
4	0.05	0.5	0
5	0.05	-0.25	0
6	0.75	-0.25	0
7	0.75	0.25	0
8	0.75	0.75	0
9	-0.5	2.25	0

Figure II.14: Position and coordinates of probes in the full domain

In this case, the **Pressure**, the **Tubulent Energy** and the **Dissipation** will be removed from the listing file.

The **Courant number** (CFL) and **Fourier number** will be removed from the post-processing results[5].

Eventually, probes will be defined for chronological records, following the data given in Figure II.4. Then the **total pressure** will be deactivated from all probes.

In addition the domain boundary will be post-processed. This allows to check the boundary conditions, and especially that of the temperature and passive scalar.

5.7 Results

Figure II.15 shows the evolution of the spatial average of the temperature. This information can be post-processed in ParaView with the filter **Plot Data Over Time**.

Figure II.16 shows velocity fields colored by temperature. The effect of the head loss modeling the obstacle is clearly visible.

[5] This can be very useful to save some disk space if some variables are of no interest, as post-processing files can be large.

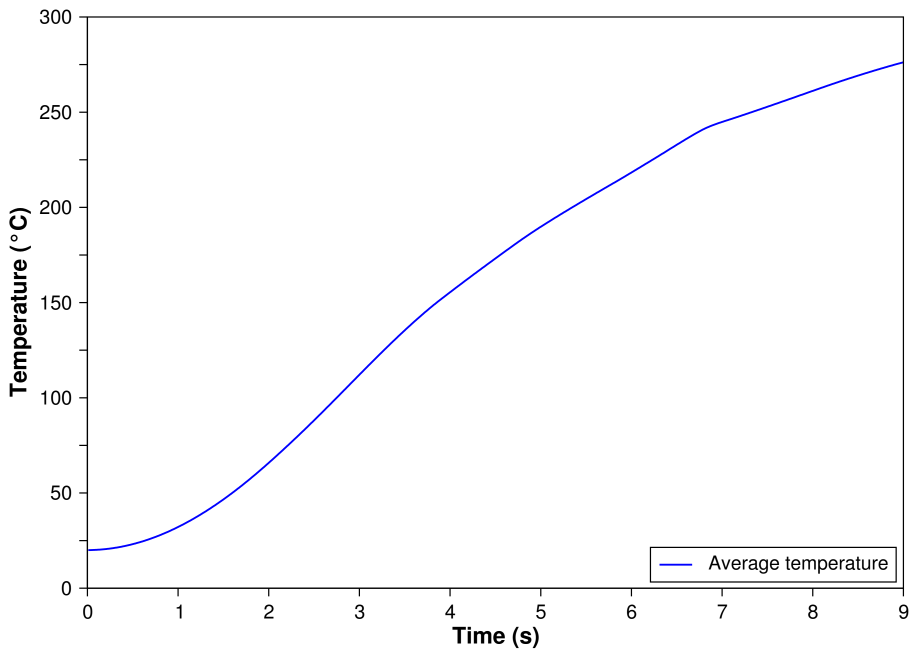


Figure II.15: Evolution of the spatial average of the temperature as a function of time - Case 3

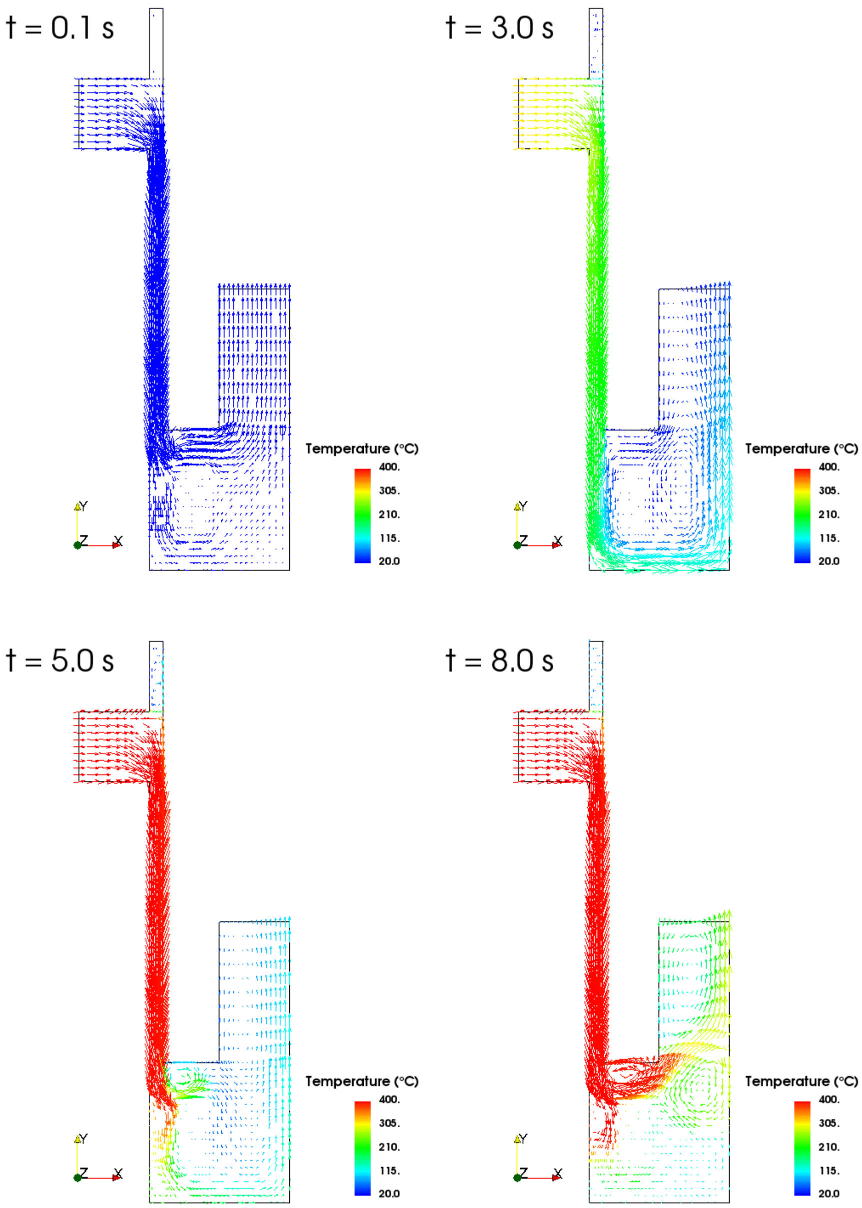


Figure II.16: Water velocity field colored by temperature - Case 3

Part III

Step by step solution

1 Solution for CASE1

1.1 Mesh tab

This case corresponds to a new study, in which there will be three calculation cases (cases 1, 2 and 3). We can create one case in a single `code_saturne create` command and additional cases can be added later. To test this functionality, first create the study directory, with case subdirectory 🗀 `case1`, as below:

```
$ code_saturne create -s full_domain -c case1
$ cd full_domain
```

Go to the 🗀 `DATA` directory in 🗀 `case1`, open a new case and select the meshes to use.

Click on the heading **Mesh** then add three meshes which have to be joined as shown below.

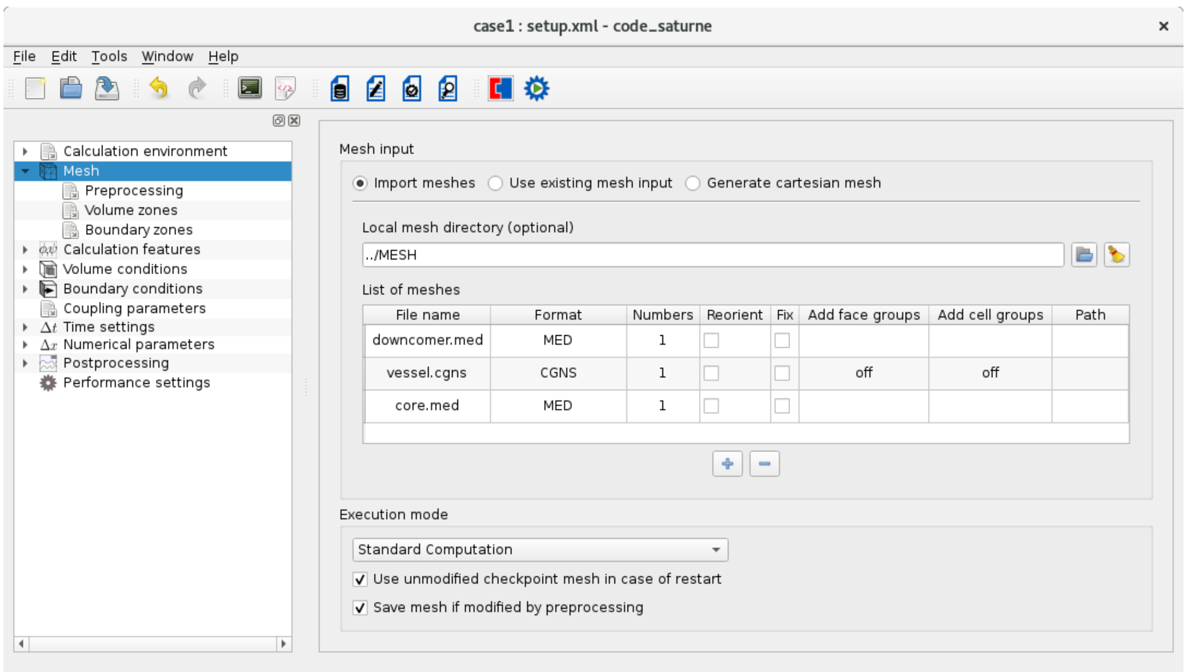


Figure III.1: Meshes: list of meshes

Preprocessing In order to join the three meshes, you must add a selection criteria in the box **Selection criteria** of the **Face joining (optional)** item under the **Preprocessing** sub-folder. In this case, only faces in groups DFuse, VFuse and CFuse can be joined (different names of groups can be entered on a single line, separated by comma).

Click on the [+] icon to enter the list of groups to be joined in the **Face joining (optional)** item.

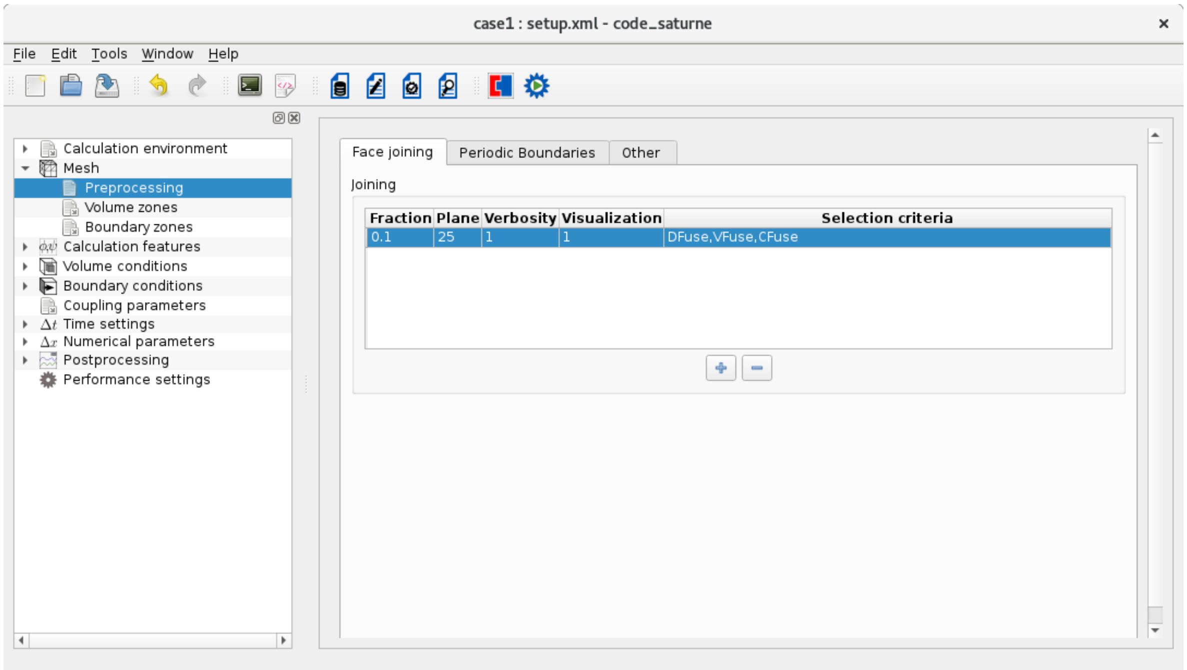


Figure III.2: Join a mesh

Boundary zones The procedure is the same as in `simple_junction`, but the groups are differents. Note that groups DFuse and CFuse have completely disappeared in the joining process (they are now internal faces and are not considered as boundaries), while some boundary faces of the group VFuse remain.

Create the inlet, outlet and symmetry boundary zones with the following groups:

```
- Inlet    :  ''INLET''
- Outlet   :  ''OUTLET''
- Symmetry:  ''DSym or VSym or CSym''
```

In this case, different conditions are applied for the walls. Separate corresponding wall boundary regions must therefore be created, following the data in the following table.

Label	Nature	Selection criteria
Wall_1	Wall	VFuse and $0.1 \leqslant x$ and $x \leqslant 0.5$
Wall_2	Wall	DWALL1 or DWALL2
Wall_3	Wall	DWALL3 or DWALL5 or VWalls
Wall_4	Wall	DWALL4 and y>1
Wall_5	Wall	DWALL4 and y$\leqslant$1
Wall_6	Wall	CWalls

The **Wall_1** region combines group's name and geometrical criteria. The associated character string

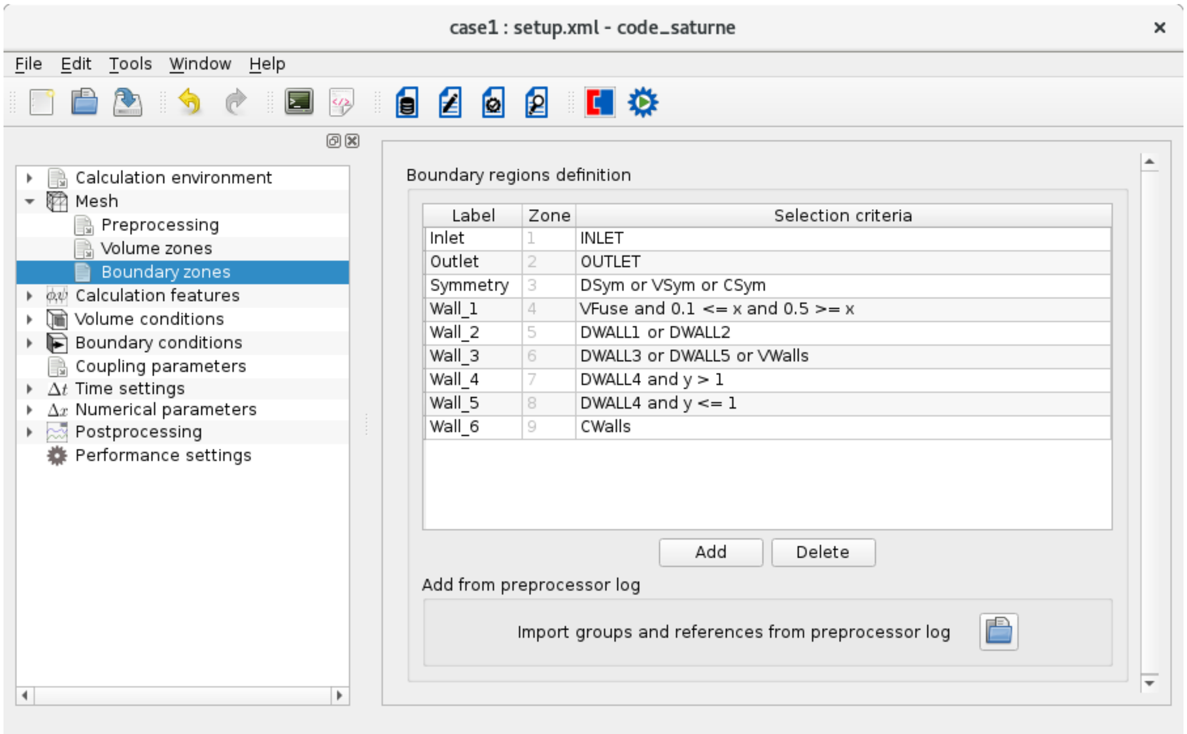


Figure III.3: Creation of the boundary zones

to enter in the **Selection criteria** box [1] is as follows:

```
''VFuse and 0.1 <= x and 0.5 >= x''
```

Define the other wall boundary zones. The faces of the group `DWALL4` have to be divided in two separate zones, based on a geometrical criterium on y.

[1]Note that, due to the joining process, there are in fact no boundary faces of group VFuse with x coordinate outside the [0.1;0.5] interval. The geometrical criterium is therefore not necessary. It is presented here to show the capabilities of the face selection module.

Mesh quality verification You can now verify the quality of your mesh by running a **Mesh quality criteria only** Computation. To do so, go back to the heading **Mesh** and change the **Execution mode** to **Mesh quality criteria only**. calculation.

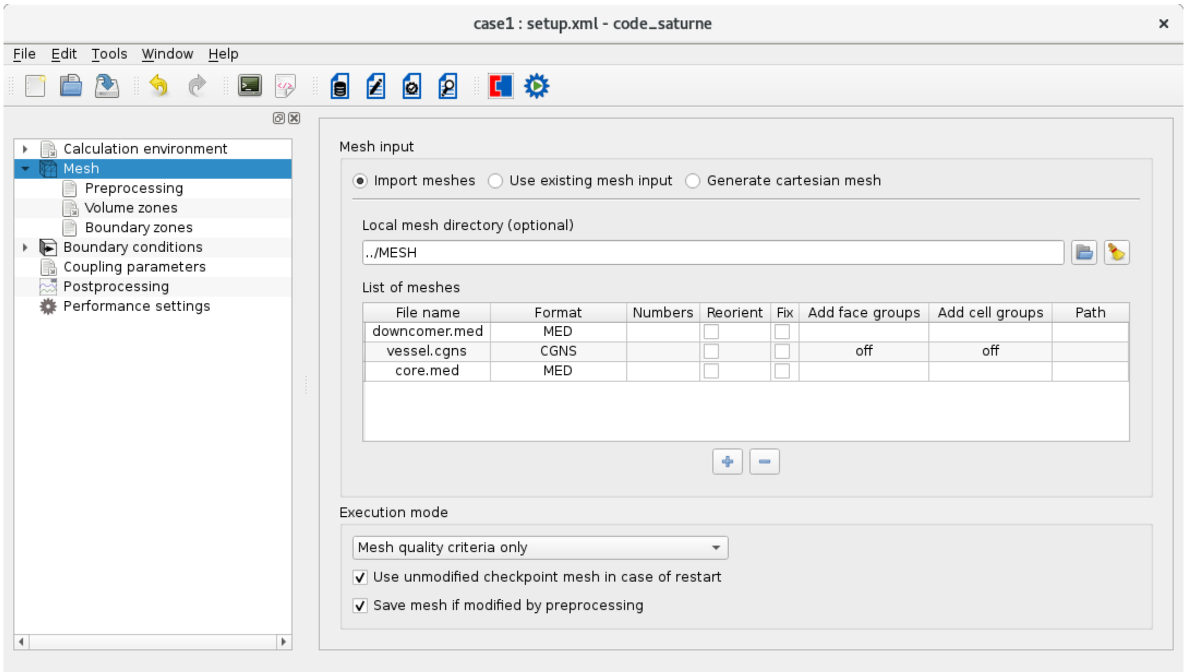


Figure III.4: Mesh quality criteria

Click on the  icon to run the mesh checking computation. Once this is done you can open the file entitled "listing" located in the fresh RESU sub-folder. You will find information related to the orthogonality, meshes types, etc.

1.2 Calculation features tab

Now, go back to the **Standard Computation** mode and click on the **Calculation features** heading. The heading **Calculation features** is identical to `simple_junction`.

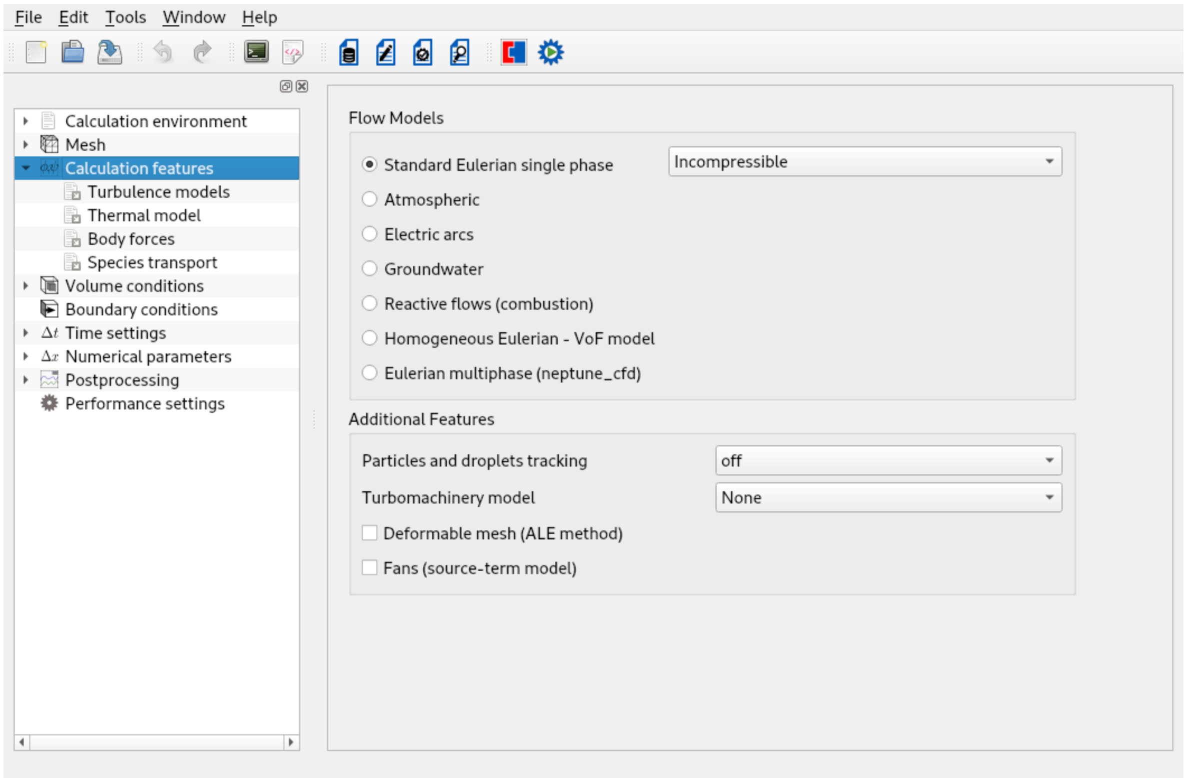


Figure III.5: Thermophysical models/Calculation features: unsteady flow

Turbulence models Set a k-ε Production turbulence model with a velocity scale reference value of **1.0**(m/s). See III.6

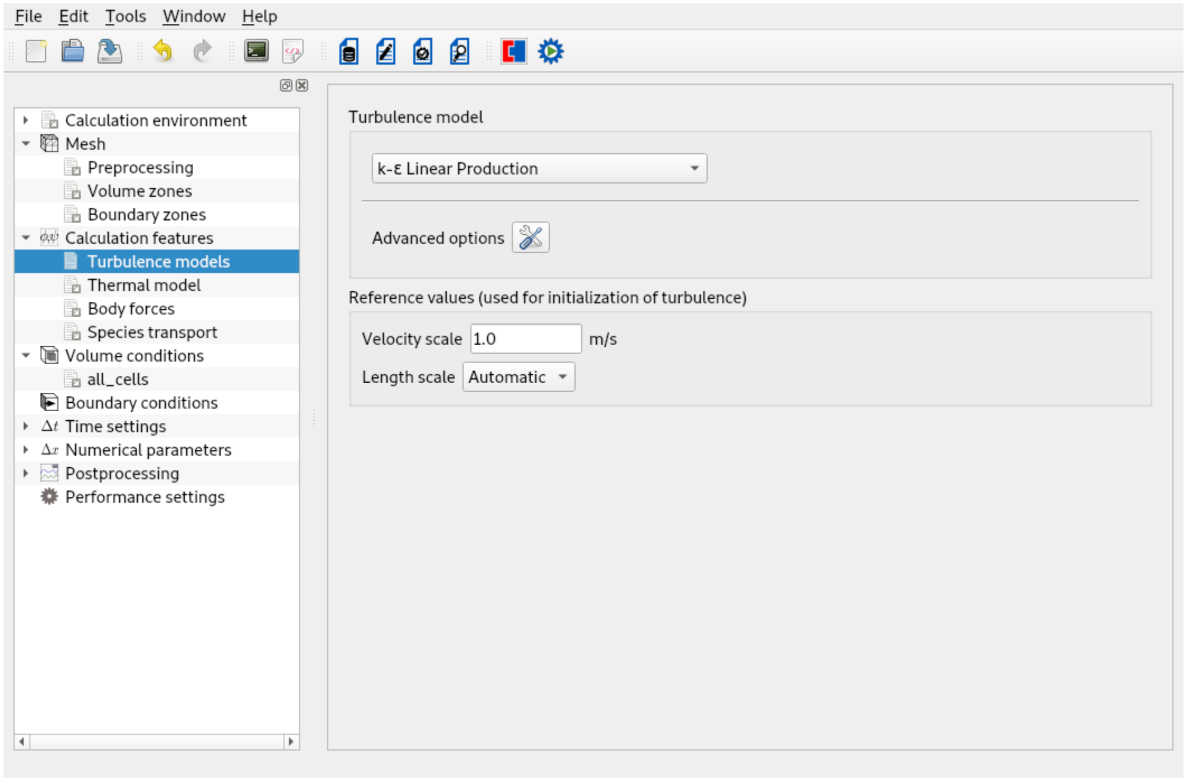


Figure III.6: Turbulence Model

Species transport To add an additional scalar, click on the [+] icon in the **Species transport** item under the **Calculation features** tab. You can rename the scalar by clicking on it. Here you can add : **scalar2**

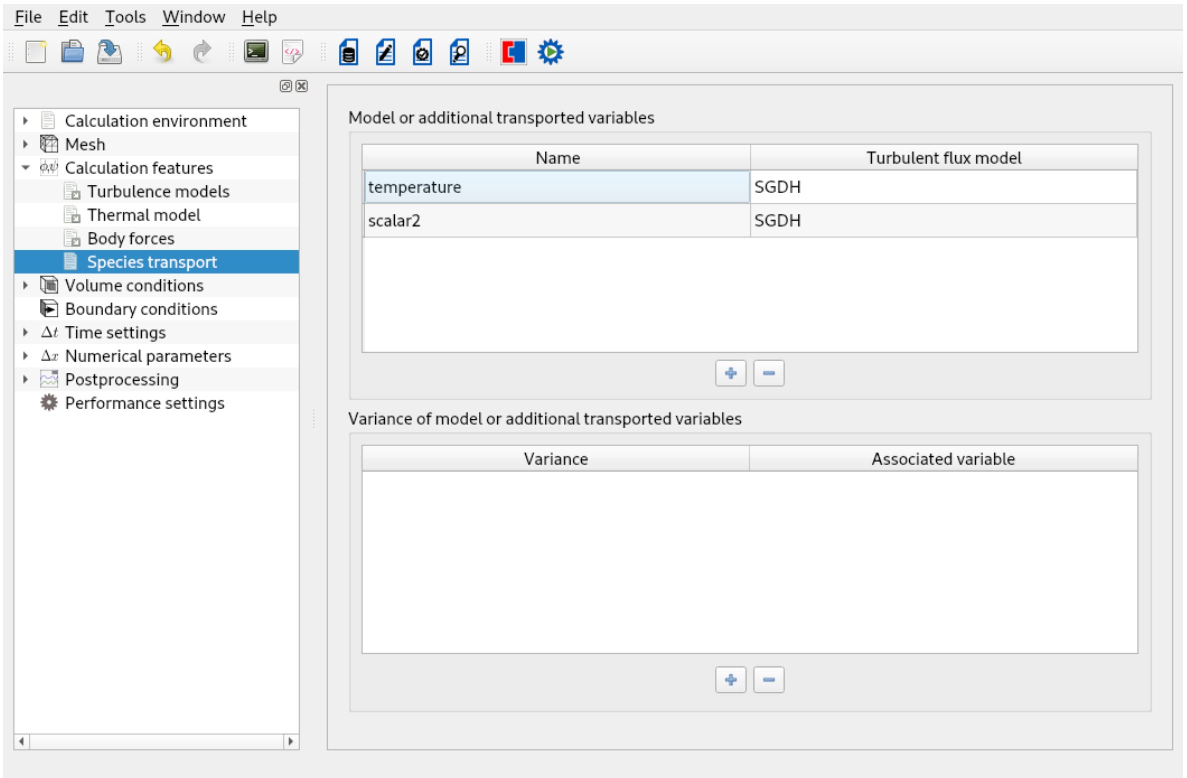


Figure III.7: Additional scalar

1.3 Volume conditions tab

Before setting fluid properties do not forget to tick all parameters you need otherwise you won't be able to see all tab options later. Tick : **Initialization** and **Physical properties**

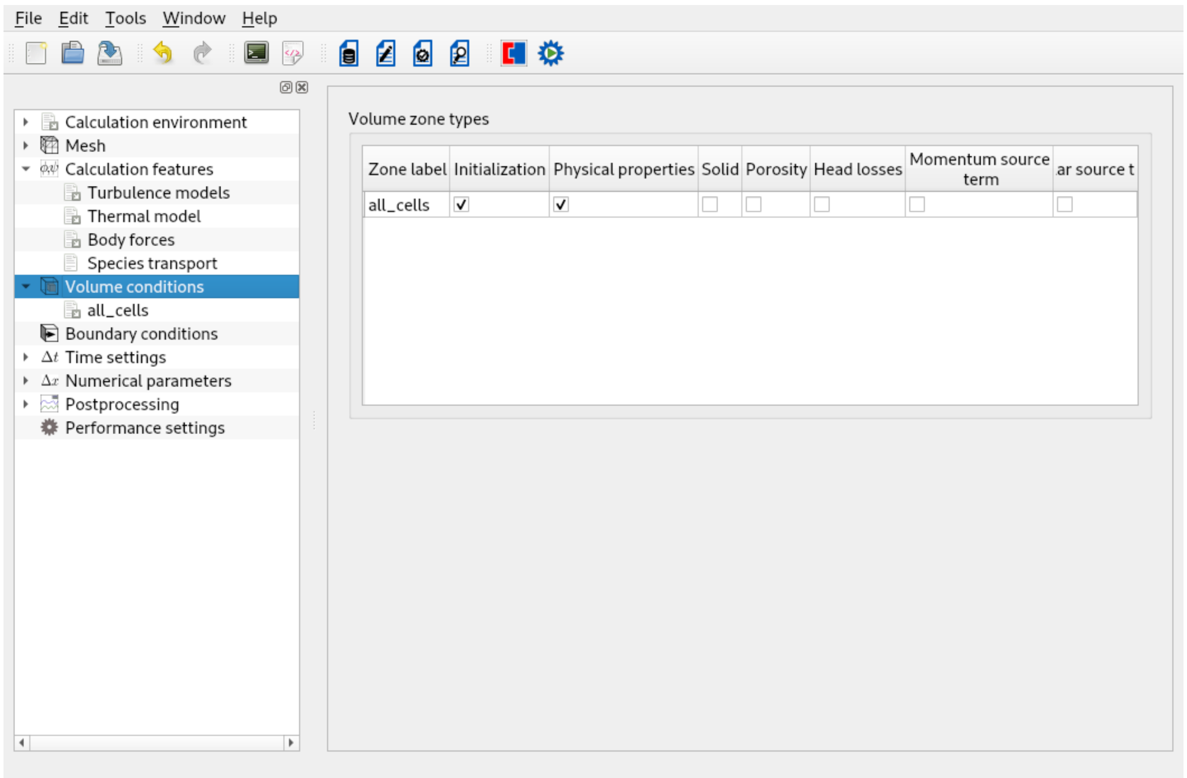


Figure III.8: Additional scalar

Physical properties The heading **Physical properties** is identical to `simple_junction`, except for the new scalar. Here, we can specify the diffusion coefficient of this new scalar.

- $\rho \quad = 725.735 \ kg.m^{-3}$
- $\mu \quad = 0.895 \times 10^{-4} \ kg.m^{-1}.s^{-1}$
- $C_p \quad = 5\,483 \ J.kg^{-1}.K^{-1}$
- $\lambda \quad = 0.02495 \ W.m^{-1}.K^{-1}$

Click on the scalar name to highlight it, then enter the value in the box. In this case, the species diffusion coefficient value is **8.55** ($\times 10^{-5} \ m^2.s^{-1}$) for the **scalar2** to solve.

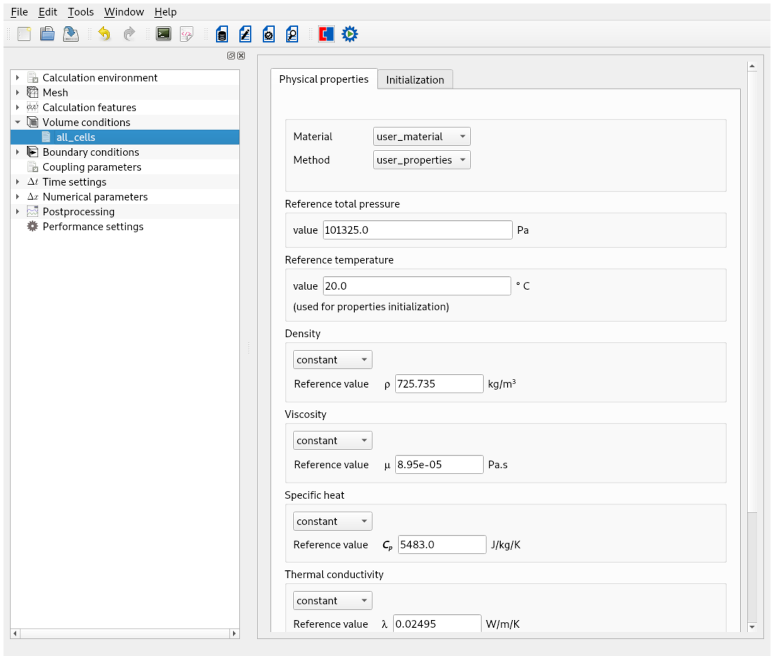


Figure III.9: Fluid properties : part1

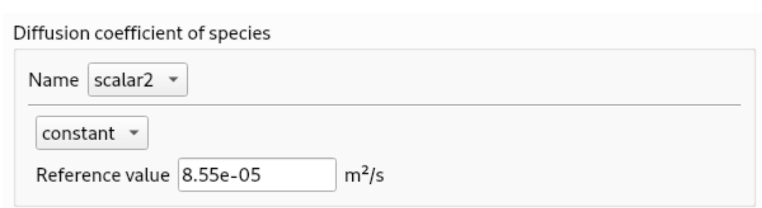


Figure III.10: Fluid properties : part 2 - scalar2 settings

Initialization To initialize variables at the instant $t = 0$ s, go to the **Initialization** item under the heading **Volume conditions**.

Here the velocity, the thermal scalar and the turbulence can be initialized. In this case, the default values to be set are: zero velocity (default), an initial temperature of **20**°C and a turbulence level based on a reference velocity of **1** ($m.s^{-1}$) (default). You must also initialize the `scalar2` species at **10**°C.

Specific zones can be defined with different initializations. In this case, only the default **all_cells** is used.
Note: To set parameters you need to click on the green icon  next to the selected field.

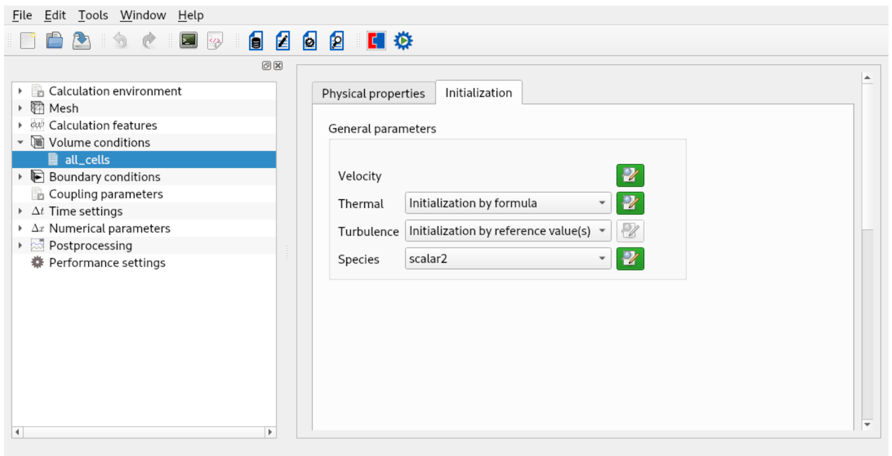


Figure III.11: Initialization

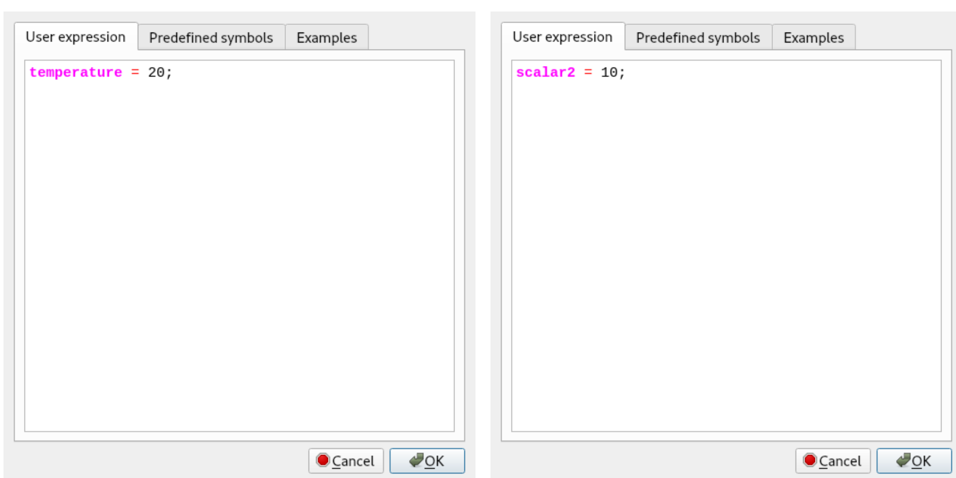


Figure III.12: Initialization: temperature and scalar2

1.4 Boundary conditions tab

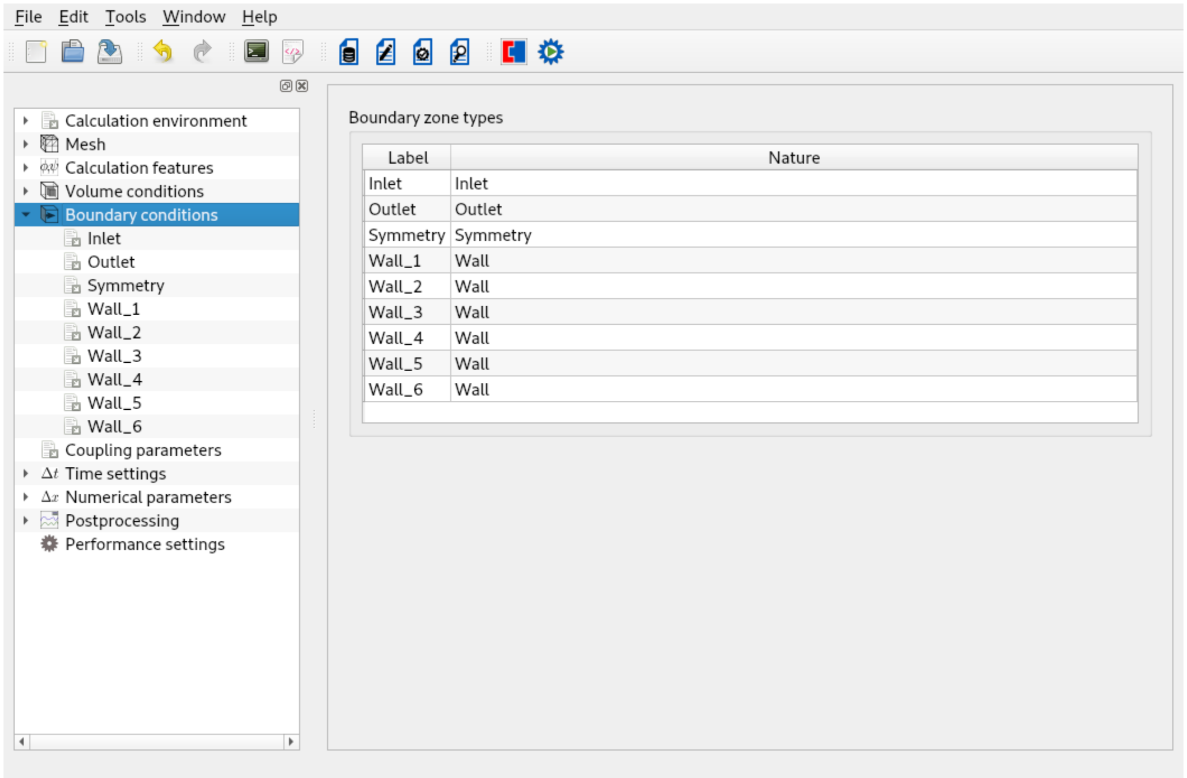


Figure III.13: Boundary conditions : nature

The dynamic boundary conditions are the same as in `simple_junction` for the inlet, and there are still no sliding walls.

– Inlet:

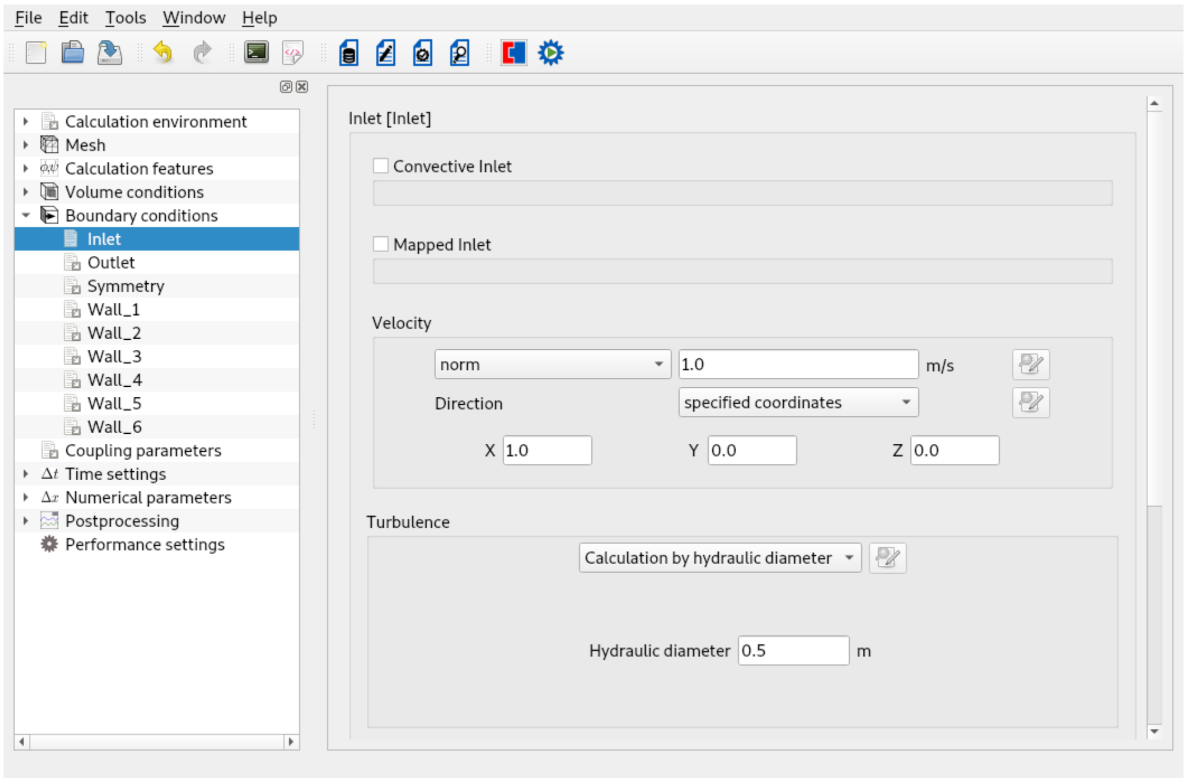


Figure III.14: Dynamic variables boundary: inlet - part1

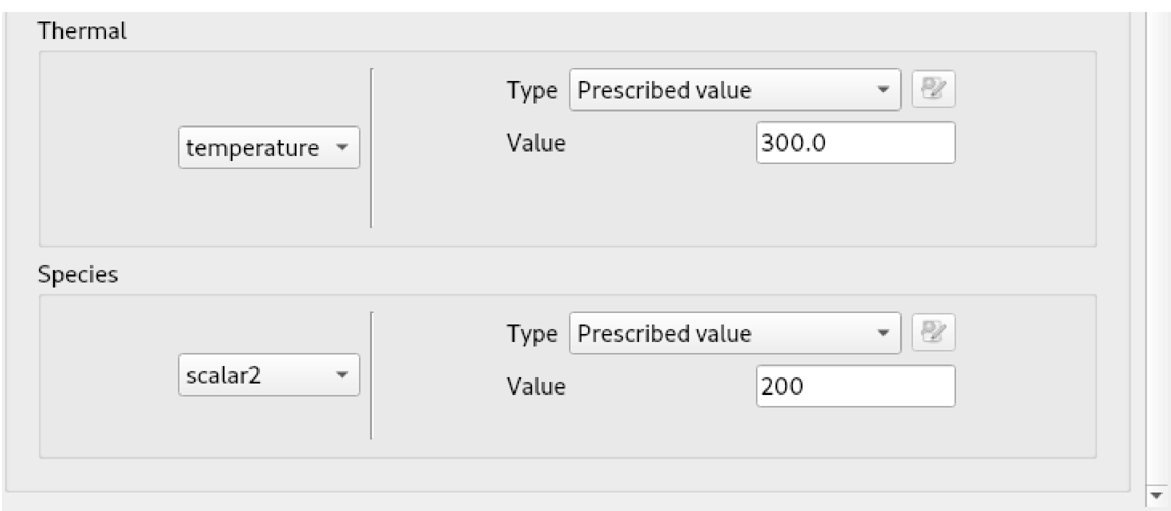


Figure III.15: Dynamic variables boundary: inlet - part2

– Outlet:

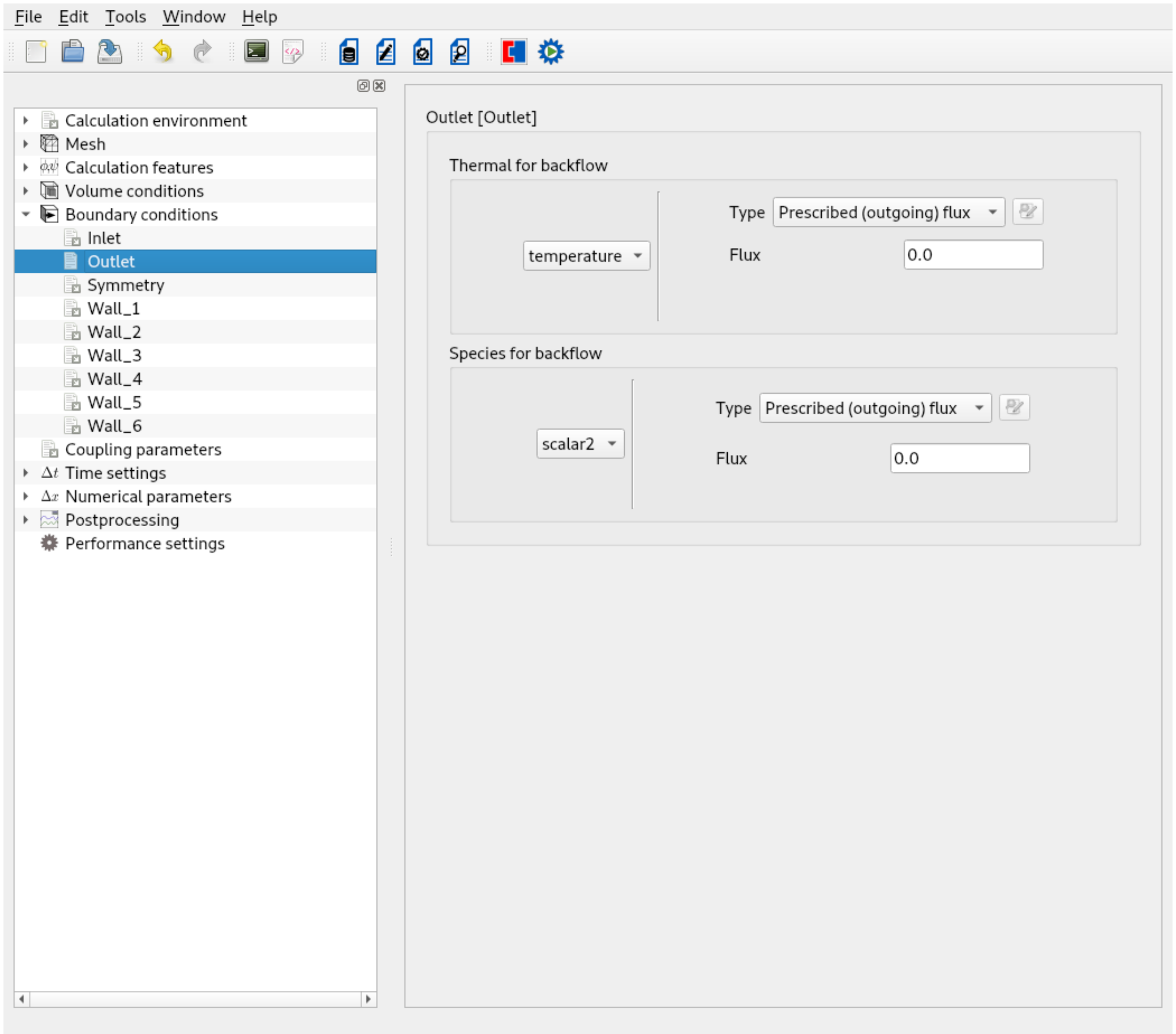


Figure III.16: Dynamic variables boundary: outlet

– Wall:

To configure the scalar boundary conditions on the walls, select individually each wall in the **Boundary conditions** item.

On all the walls, a default homogeneous **Prescribed flux** is set for temperature, and **Prescribed value** is specified for the passive scalar for each wall, named `scalar2`, according to the following table:

Wall	Nature	`Scalar2` value
Wall_1	Prescribed value (Dirichlet)	0
Wall_2	Prescribed value (Dirichlet)	5
Wall_3	Prescribed value (Dirichlet)	0
Wall_4	Prescribed value (Dirichlet)	25
Wall_5	Prescribed value (Dirichlet)	320
Wall_6	Prescribed value (Dirichlet)	40

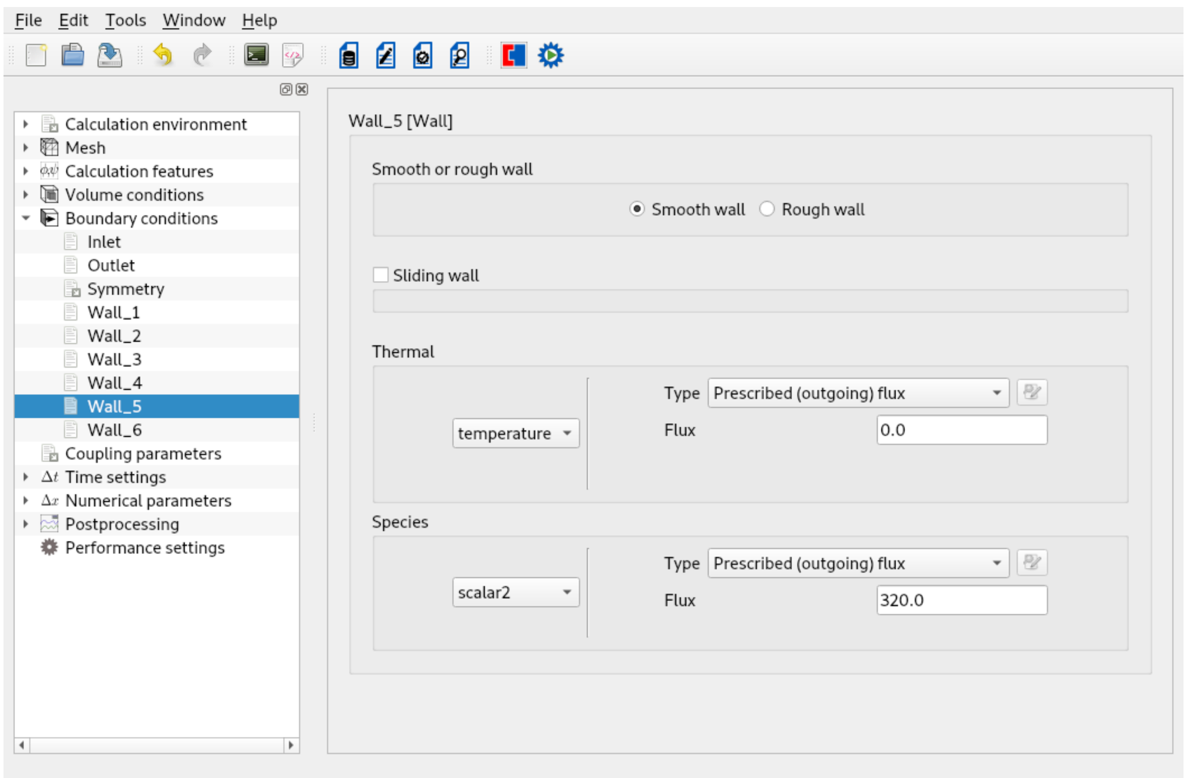


Figure III.17: Scalars boundaries: wall_5

1.5 Time settings tab

Some calculation parameters need to be defined now. Go to the **Time settings** heading. In our case the time step is **Constant** and the **Velocity-Pressure algorithm** is **SIMPLEC**. Set the number of iterations to **300** and the reference time step to **0.05** (s).

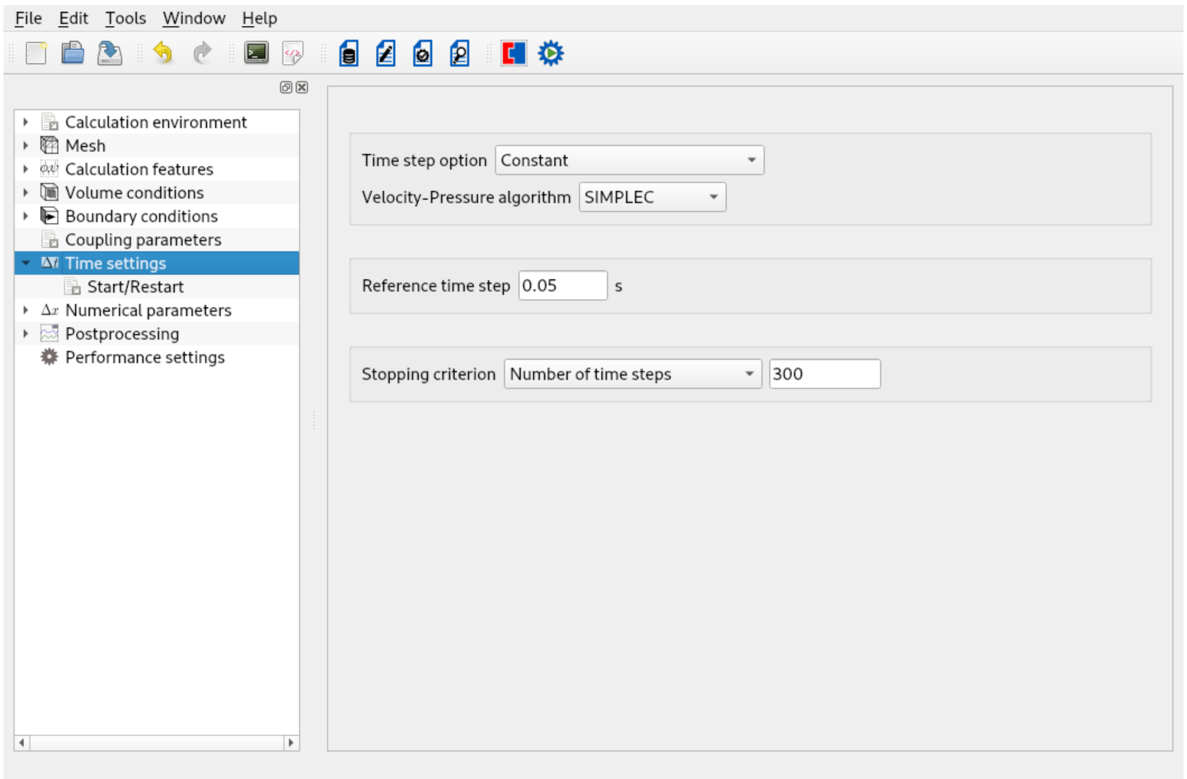


Figure III.18: Time step setting

Numerical parameters are the same as in `simple_junction`.

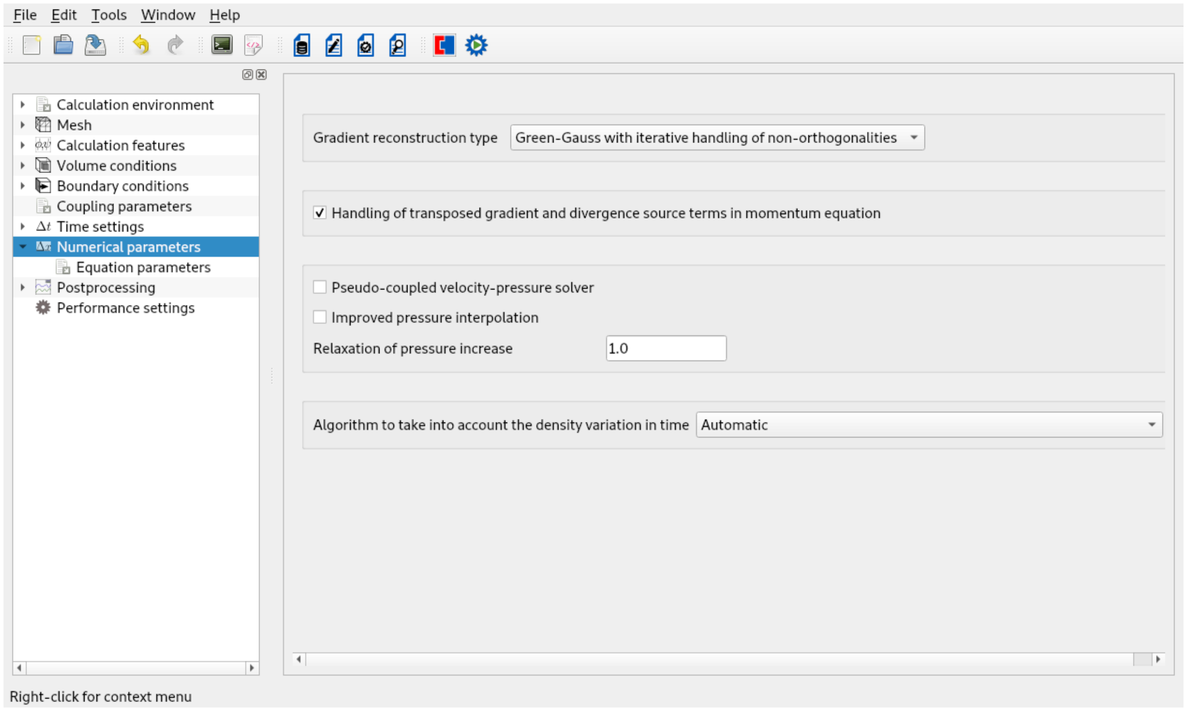


Figure III.19: Numerical parameters

1.6 Numerical Parameters tab

Go to the **Equation parameters** item under the heading **Numerical parameters**. You can define the maximum and minimum value for the `temperature` and for the `scalar2` scalars.

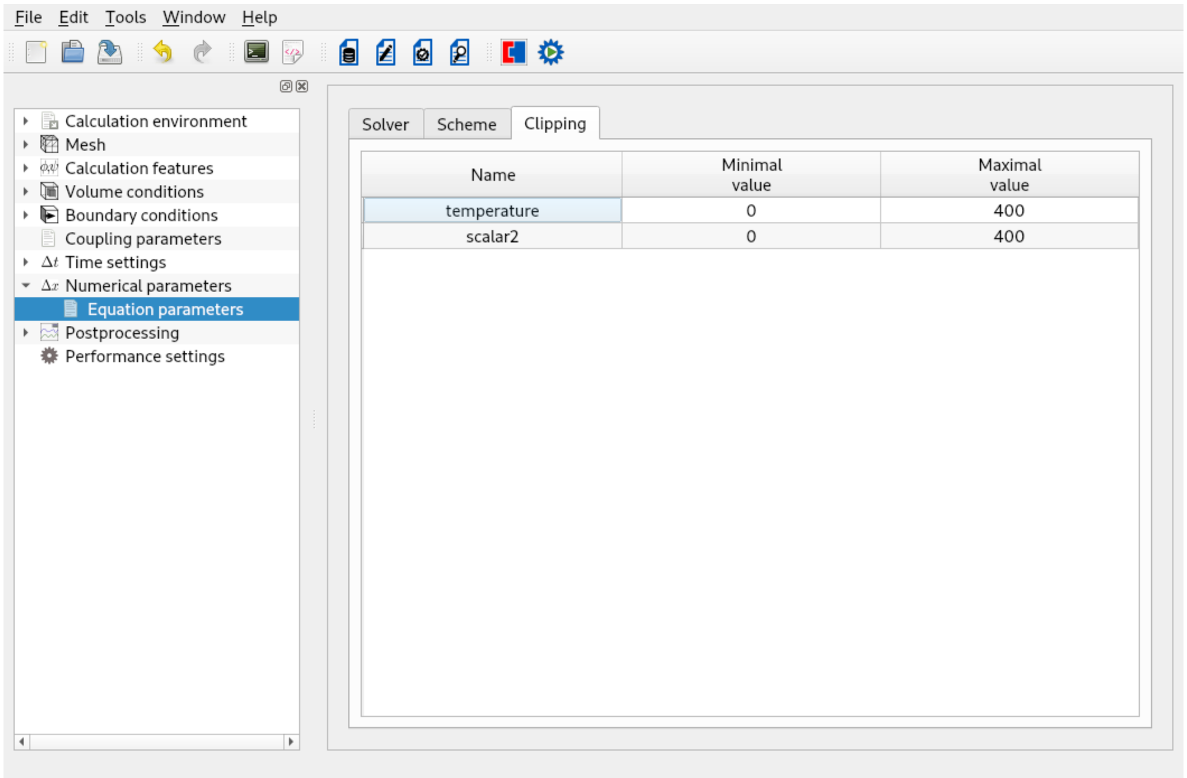


Figure III.20: Clipping

Go to the **Postprocessing** heading to set the output parameters. In the **Output Control** tab, keep the default value for the output listing frequency.

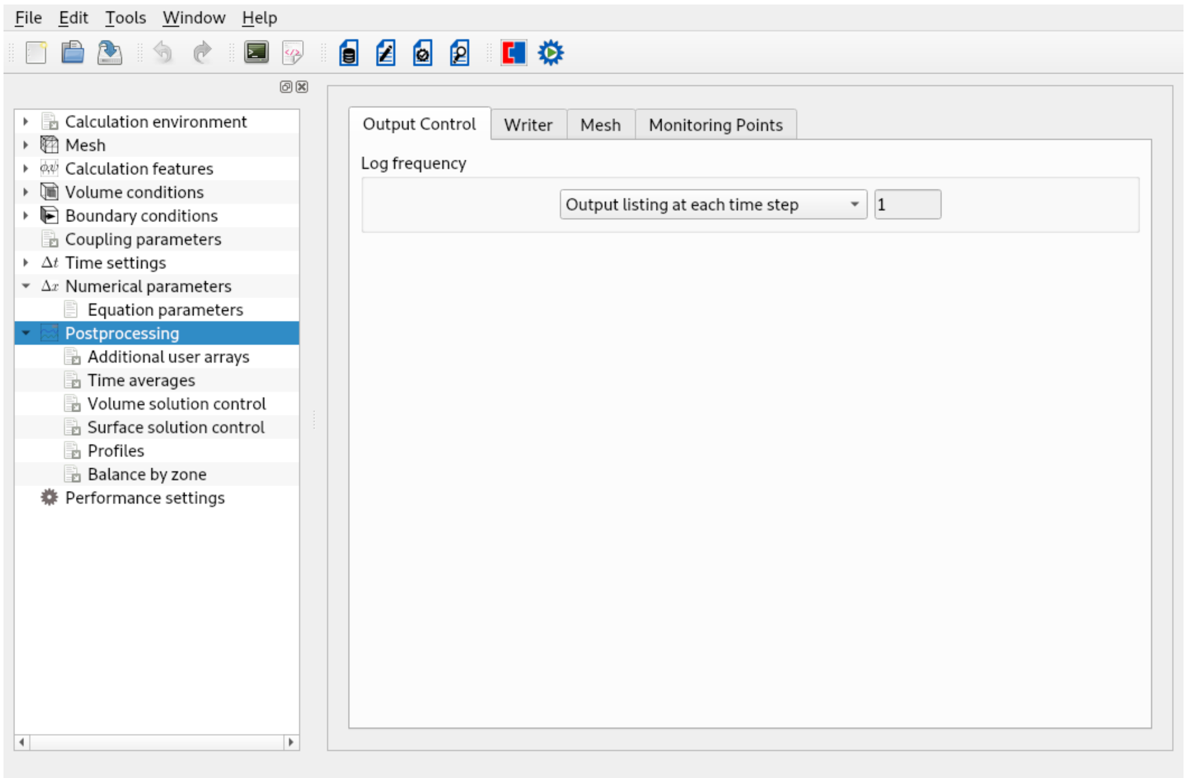


Figure III.21: Output control: log frequency

1.7 Postprocessing tab

For the post-processing, go to the **Writer** item and click on **results**.

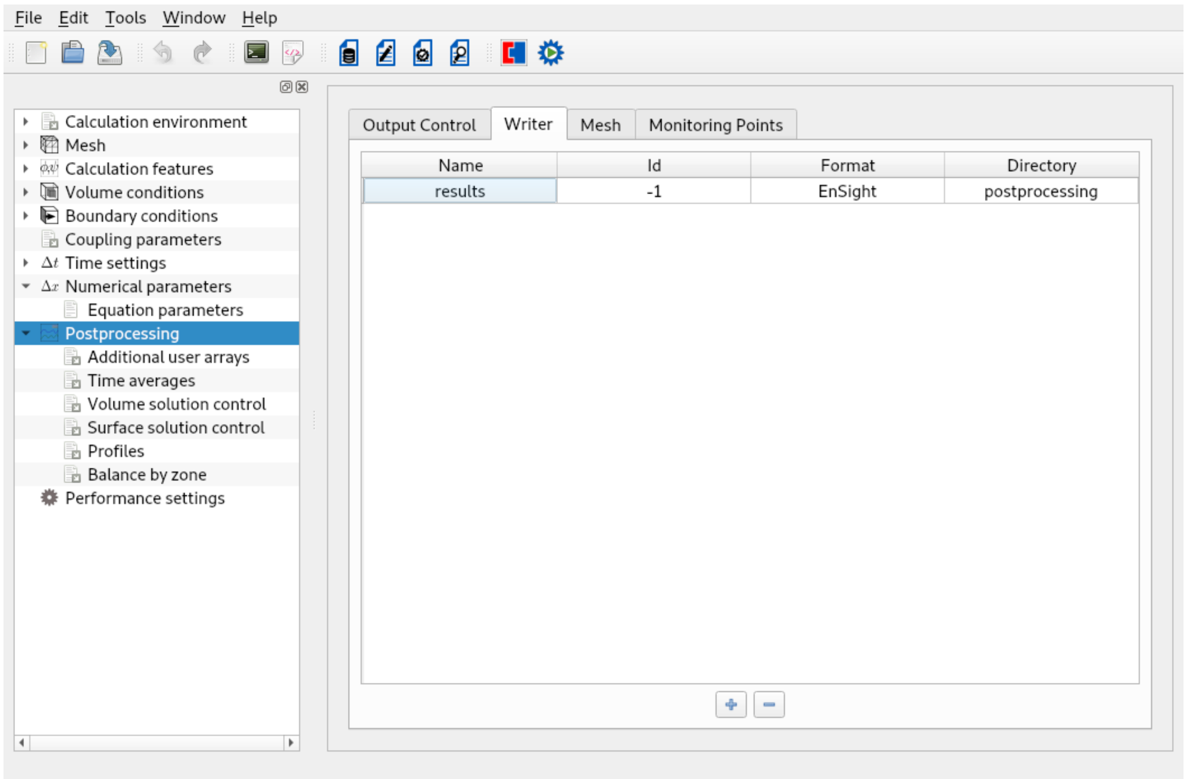


Figure III.22: Output control: writer

Now you can select the third option in the **Frequency** (**output every 'n' time steps**) item and set the value of **n** to **2**.

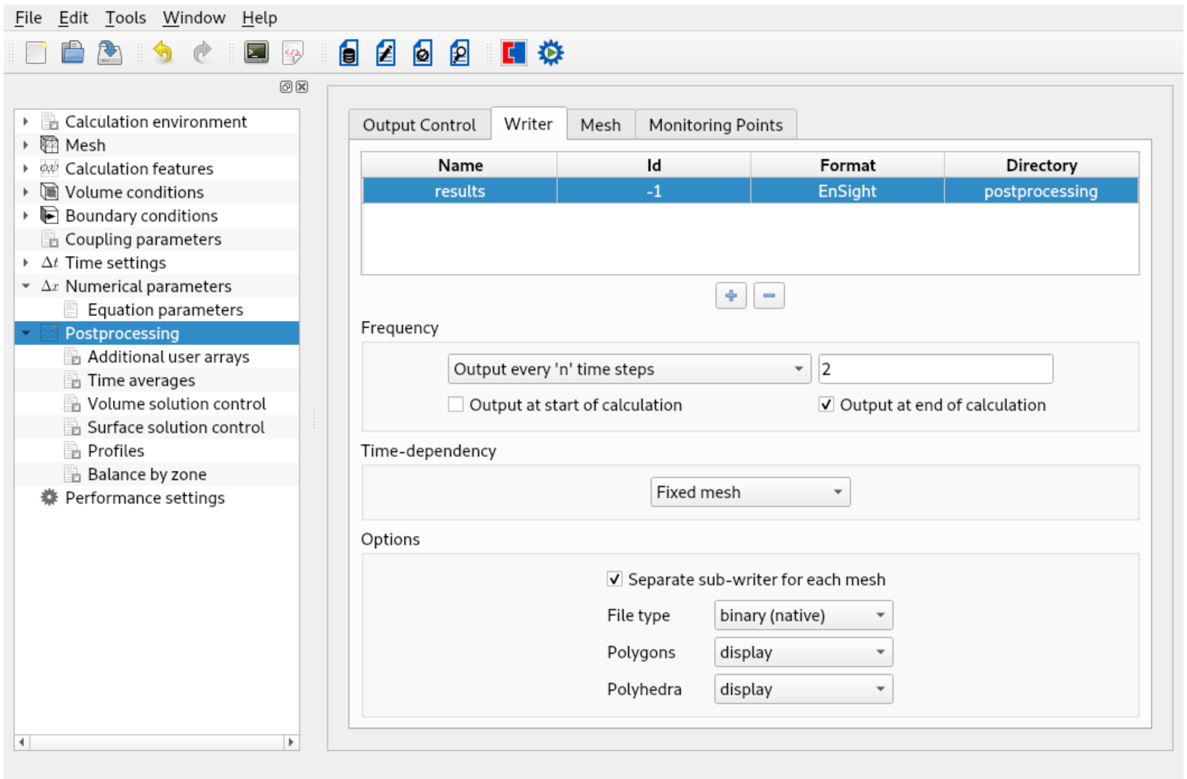


Figure III.23: Output control: results

You can also choose the format. In our case, we choose the EnSight format.

In this case, chronological records on specified monitoring probes are needed. To define the probes, click on the **Monitoring Points** tab. Click on [+] and enter the coordinates of the monitoring points you want to define.

Repeat the procedure for the other probes. The coordinates of the probes are indicated in the following table (the z coordinate is always 0):

Probe n^o.	x (m)	y (m)	z (m)
1	-0.25	2.25	0.0
2	0.05	2.25	0.0
3	0.05	2.75	0.0
4	0.05	0.50	0.0
5	0.05	-0.25	0.0
6	0.75	-0.25	0.0
7	0.75	0.25	0.0
8	0.75	0.75	0.0

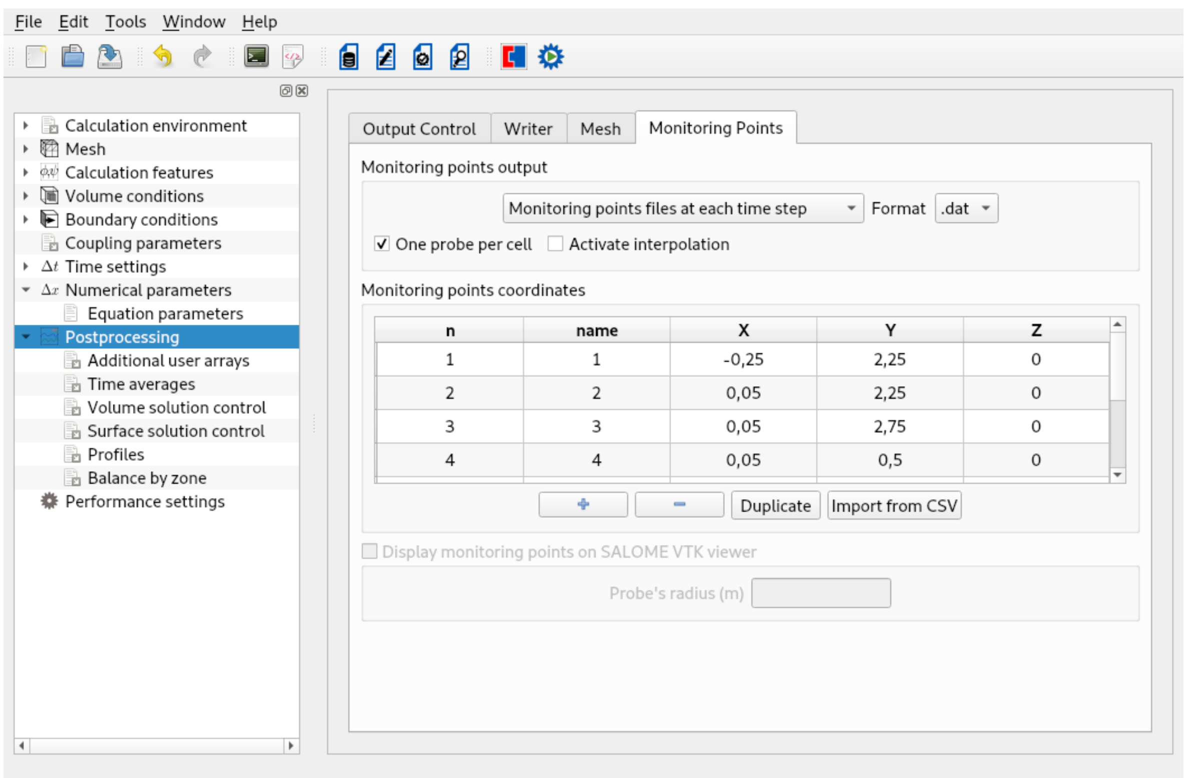


Figure III.24: Output control: monitoring points

Remember to save the `xml` file regularly.

Volume solution control Go to the **Volume solution control** item to define which variables will appear in the listing, the post-processing and the chronological records.

Uncheck the boxes in front of the **Pressure**, **k** and **epsilon** variables, in the **Print in listing** column. Information on these three variables will not appear in the output listing anymore.

Uncheck the boxes in front of the **CourantNb** and **FourierNb** variables in the **Post-processing** column. These variables will be removed from the post-processing results.

Uncheck the box in front of the **total_pressure** variable in the **Monitoring** column. No chronological record will be created for this variable.

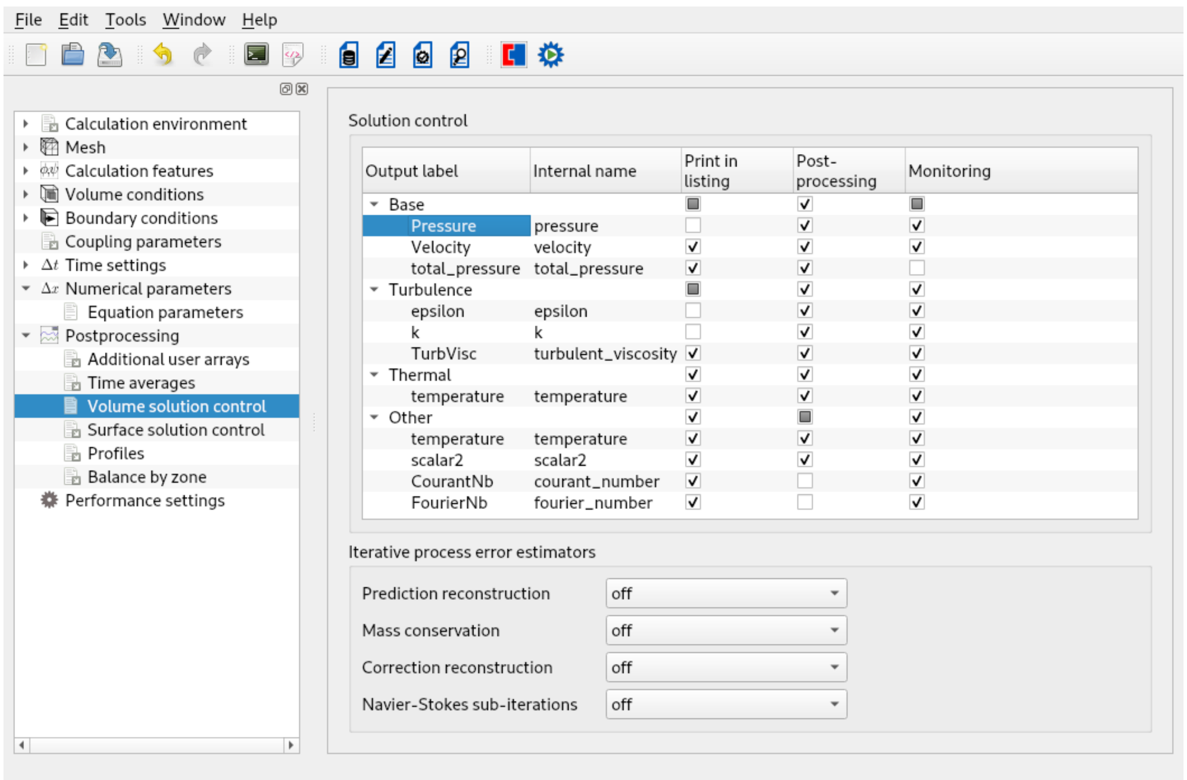


Figure III.25: Solution control: output configuration

After completing the interface, before running the calculation, some user routines need to be modified to post-process the additional transported scalars on the domain boundary (only boundary temperature post-processing can be dealt with in the GUI).

Go to the folder 🗁 `SRC/REFERENCE` and copy `cs_user_parameters.c` in the 🗁 `SRC` directory.

In this case `cs_user_parameters.c` is used to add boundary values for all scalars as follows :

```
void
cs_user_parameters(cs_domain_t *domain)
{
  /* Add boundary values for all scalars */
  {
    int n_fields = cs_field_n_fields();
    for (int f_id = 0; f_id < n_fields; f_id++) {
      cs_field_t *f = cs_field_by_id(f_id);
      if (f->type & CS_FIELD_VARIABLE)
        cs_parameters_add_boundary_values(f);
    }
  }
}
```

1.8 Run calculation

To prepare the launch script and, on certain architectures, launch the calculation, click on the icon in the menu bar and a new window will appear as shown below:

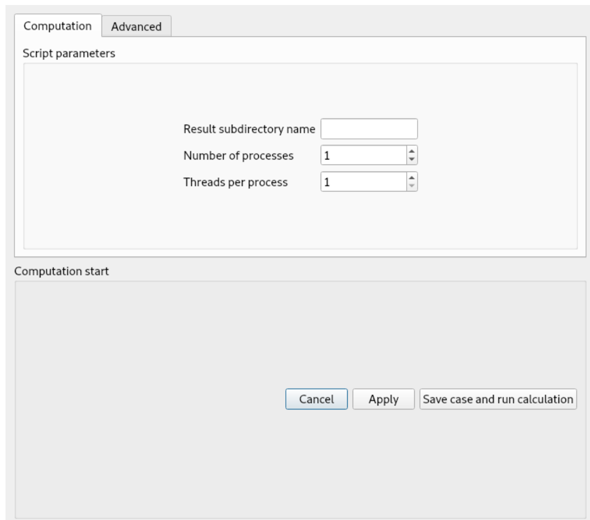


Figure III.26: Prepare batch calculation: computer selection

Run the calculation by clicking on the **Save case and run calculation** button:

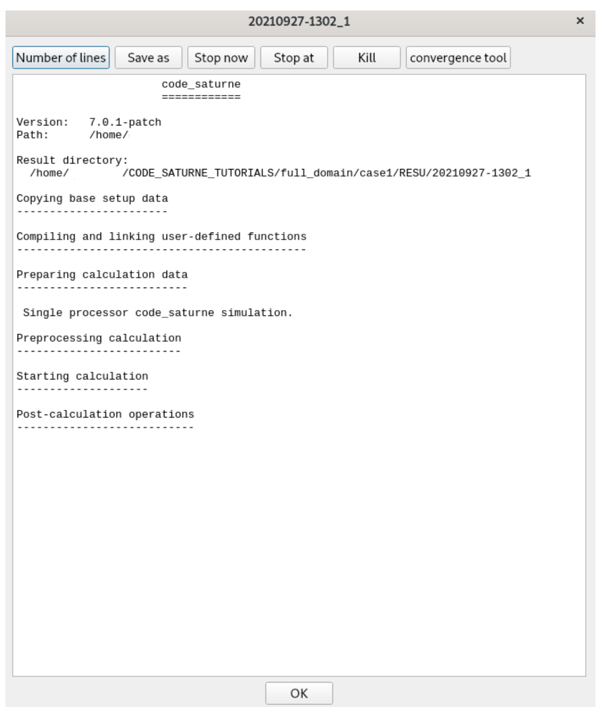


Figure III.27: Run

2 Solution for CASE2

Only a few elements are different from `case1`.

2.1 Volume tab

In this case the density becomes variable. Go to the **Fluid properties** heading and change the nature of the density from **constant** to **user law**.

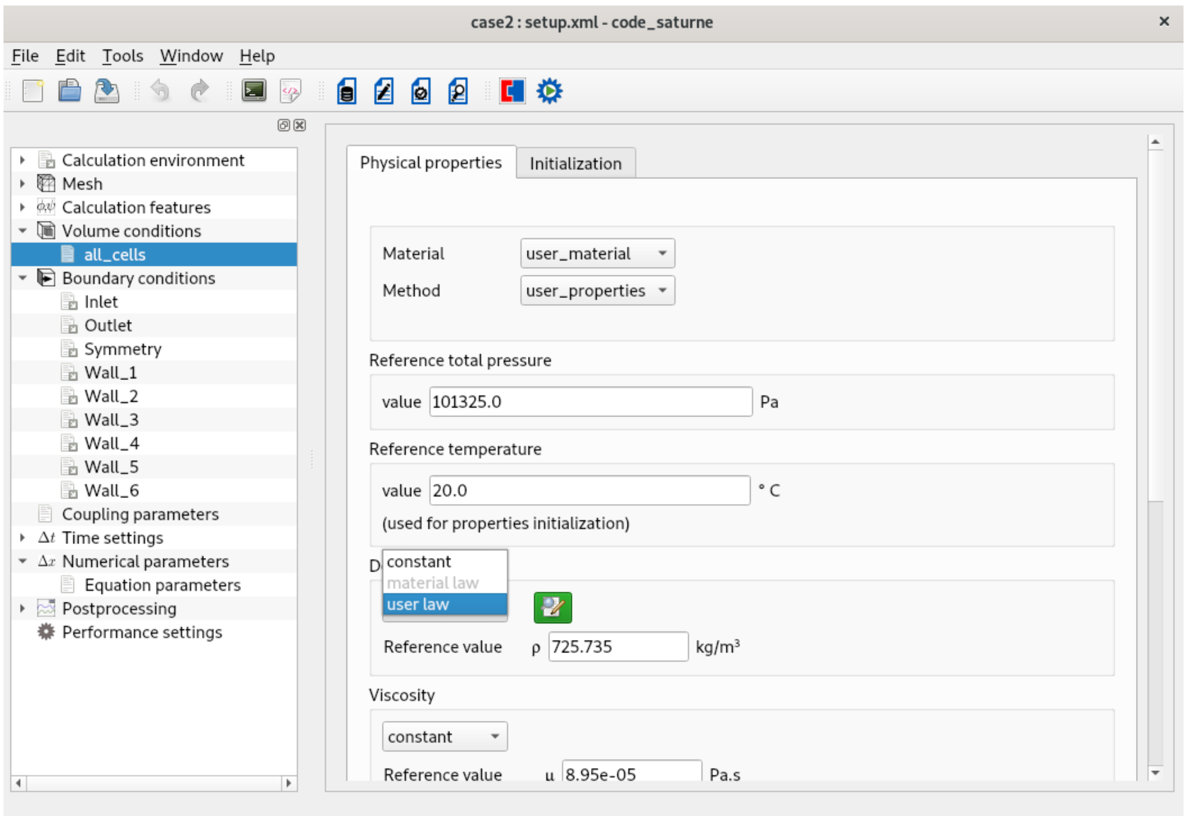


Figure III.28: Fluid properties: variable density

The user law of the density is defined as follows in the code_saturne (GUI):

```
density = temperature * ( -4.0668E-03*temperature - 5.0754E-02 ) + 1000.9;
```

Click on the highlighted icon and define the user law in the window that pops up. Follow the format used in the **Examples** tab.

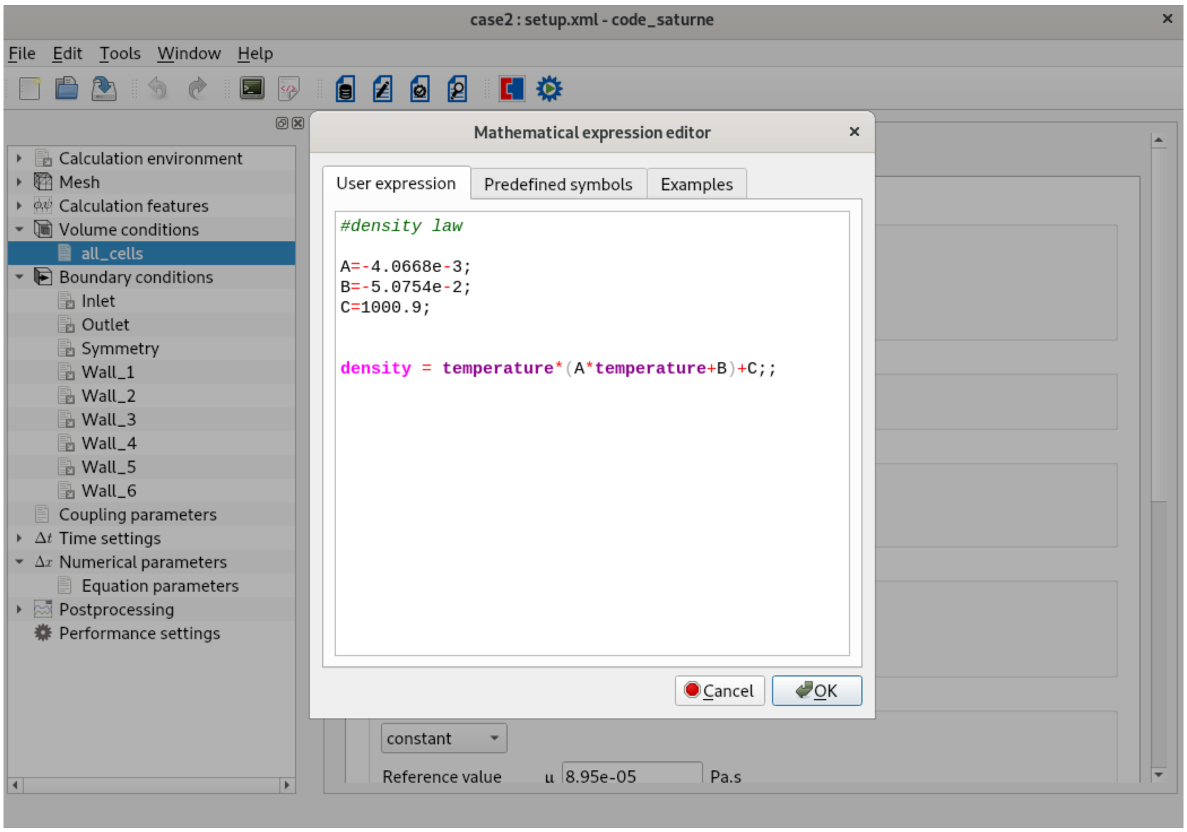


Figure III.29: Fluid properties/Variable density: user expression

Body forces As the density is variable, the influence of gravity has to be considered. Go to **Body forces** item under **Calculation features** heading and set the value of each component of the gravity vector.

```
g_x = 0.0 g_y = -9.81 g_z = 0.0
```

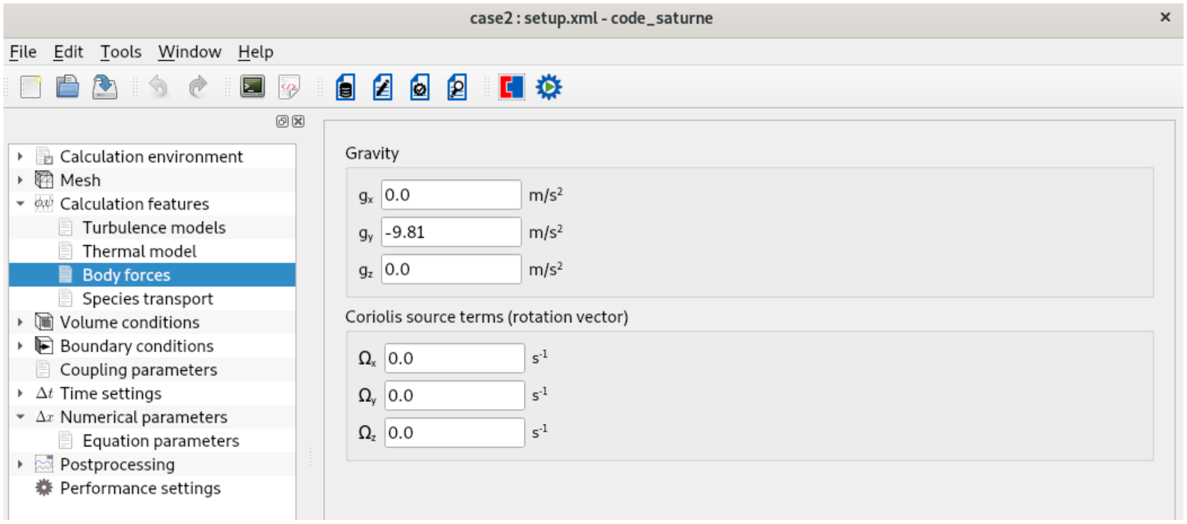


Figure III.30: Fluid properties: gravity

2.2 Boundary condition tab

In this case, the boundary condition at the inlet also becomes variable in time. Go to the **Boundary conditions** heading, select **Inlet** in the list and change the nature of the temperature from **Prescribed value** to **Prescribed value (user law)**.

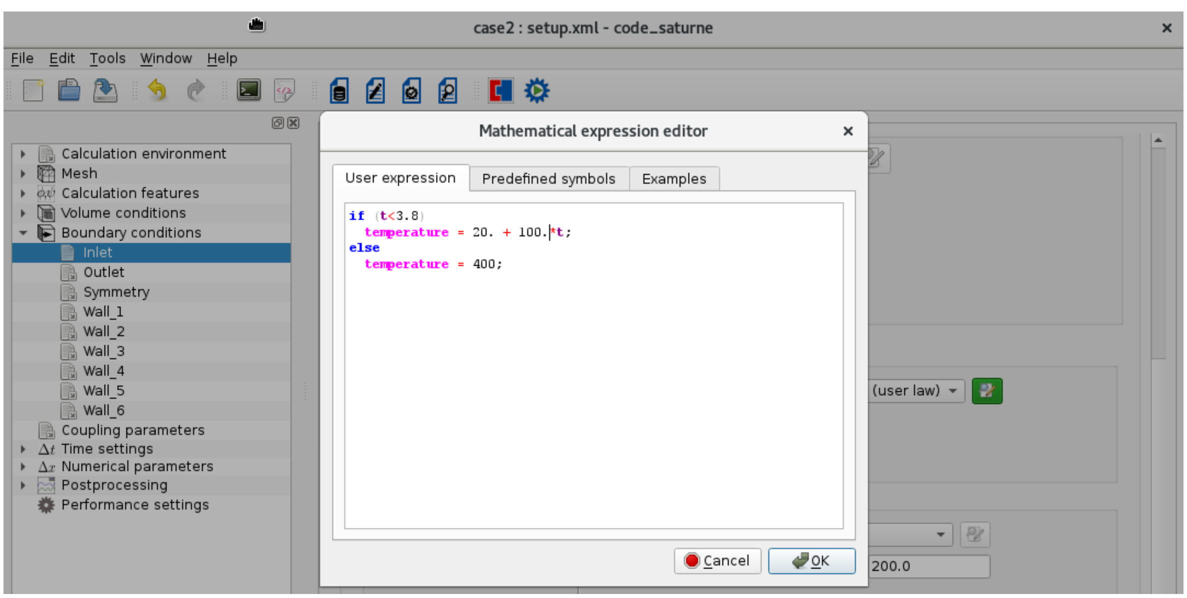


Figure III.31: User law for the inlet boundary conditions

The user law of the boundary temperature for the inlet is defined as follows in the code_saturne (GUI):

```
if (t<3.8)
temperature = 20.  + 100.*t;
else
temperature = 400.;
```

Click on the highlighted icon and define the user law in the window that pops up. Follow the format used in the **Examples** tab.

2.3 Postprocessing

Add a monitoring point close to the entry boundary condition in the **Monitoring Points** tab under the **Postprocessing** heading.

Probe	x (m)	y (m)	z (m)
9	-0.5	2.25	0.0

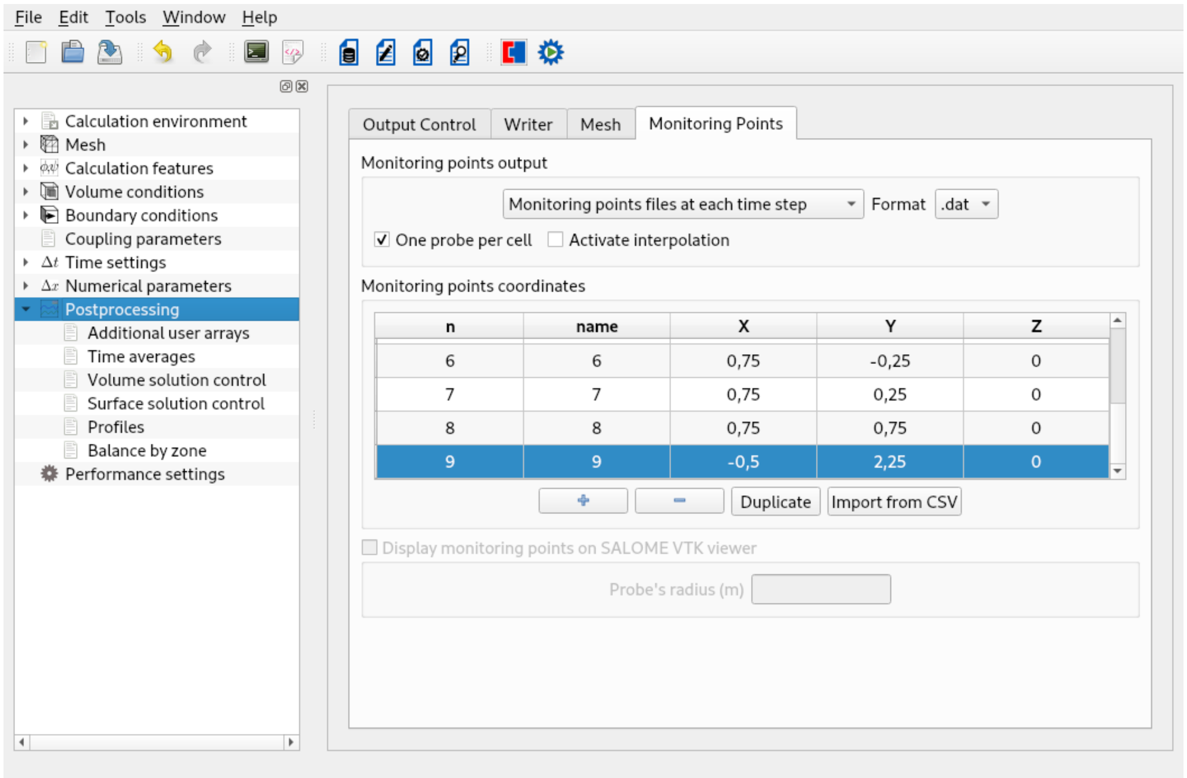


Figure III.32: New monitoring probe

Run the calculation as explained in `case1`.

When a calculation is finished, code_saturne stores all the necessary elements to continue the computation in another execution, with total continuity. These elements are stored in several files, grouped in a 🗁`yyyymmdd-hhmm/checkpoint` subdirectory, in the 🗁`RESU` directory.

2.4 Time settings

Start/Restart In this case, after the first calculation is finished, a second calculation will be run, starting from the results of the first one.

Go directly on the **Start/Restart** item under the heading **Time settings**. Activate the **Checkpoint/Restart** by clicking the On box.

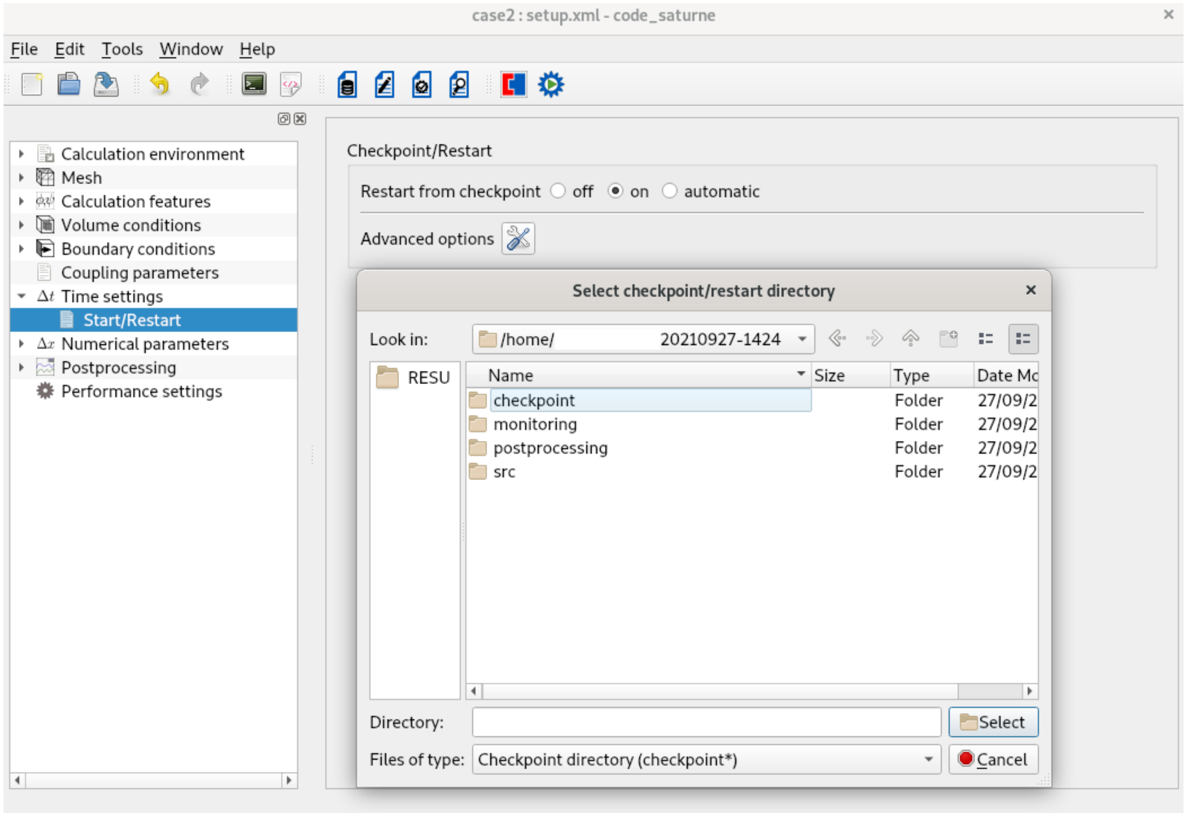


Figure III.33: Start / Restart

A window opens, with the architecture of the study sub-directories. In the 🗁`RESU` folder, click on the folder 🗁`yyyymmdd-hhmm/checkpoint` (where `yyyymmdd-hhmm` corresponds to the reference of the first calculation results). Then click on Validate.

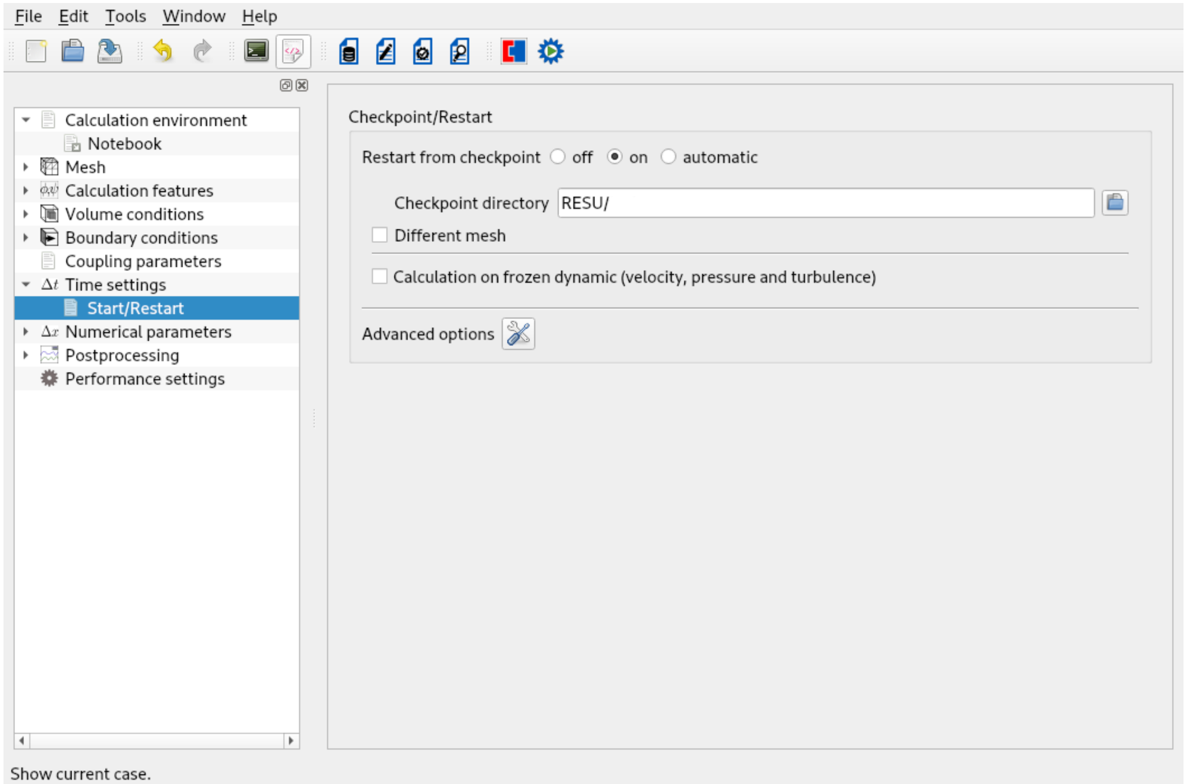


Figure III.34: Start/Restart: selection of the restart directory

Go to the **Time settings** heading and change the **Stopping criterion** from **Number of time steps** to **Additional time steps**. Set it to 400 which means another 400 iterations (the total number of iterations is 700 iterations).

Eventually, run the calculation.

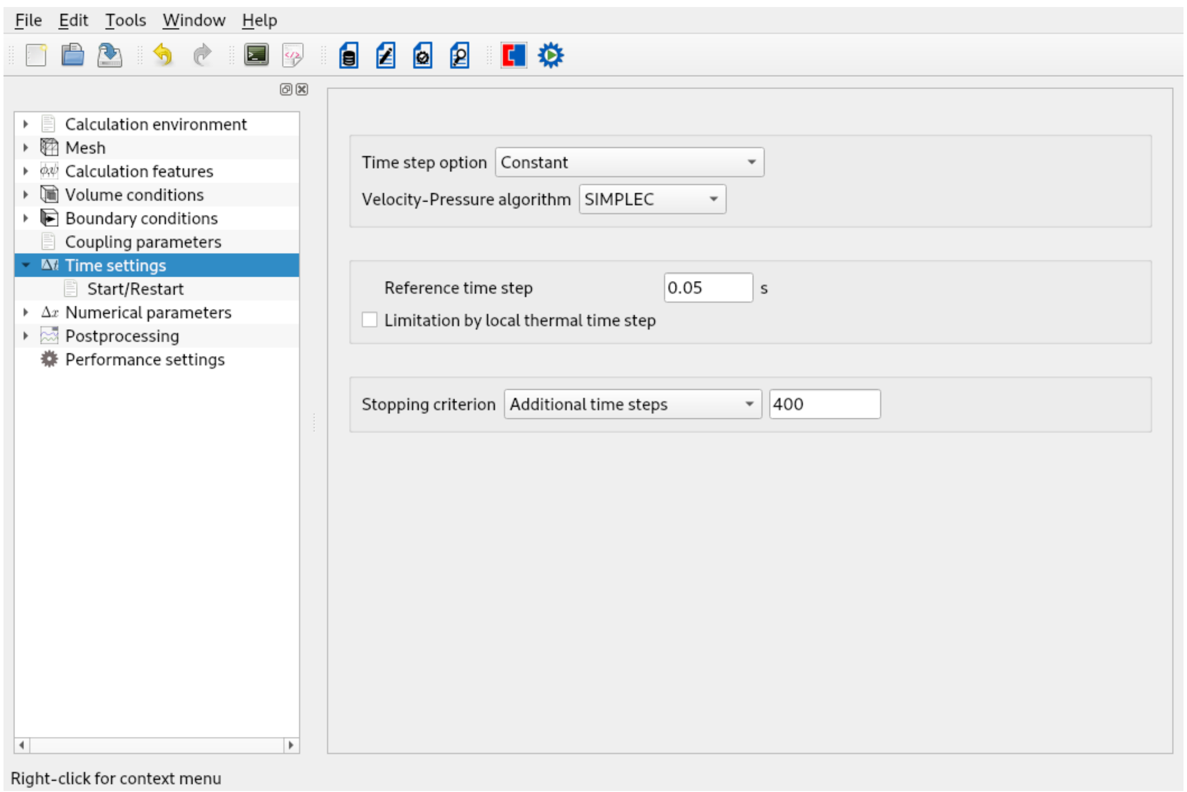


Figure III.35: Time step

3 Solution for CASE3

This case is similar to `case2`, with the following differences:

- **Step 1**: define head losses in the fluid domain,
- **Step 2**: update new Time settings,
- **Step 3**: parallel computation on 2 processors,

• Define the head losses in the Graphical User Interface (GUI)

3.1 Mesh tab

Volume zones Go to **Volume zones** folder. Click on Add to create a new volume zone. You can renamed the **Label** as losses region.
Define the limits of the head losses region in **Selection criteria**. The associated character string to enter is as below:

‘‘$x >= 0.2$ `and` $x <= 0.4$ `and` $y >= -0.75$ `and` $y <= -0.25$’’

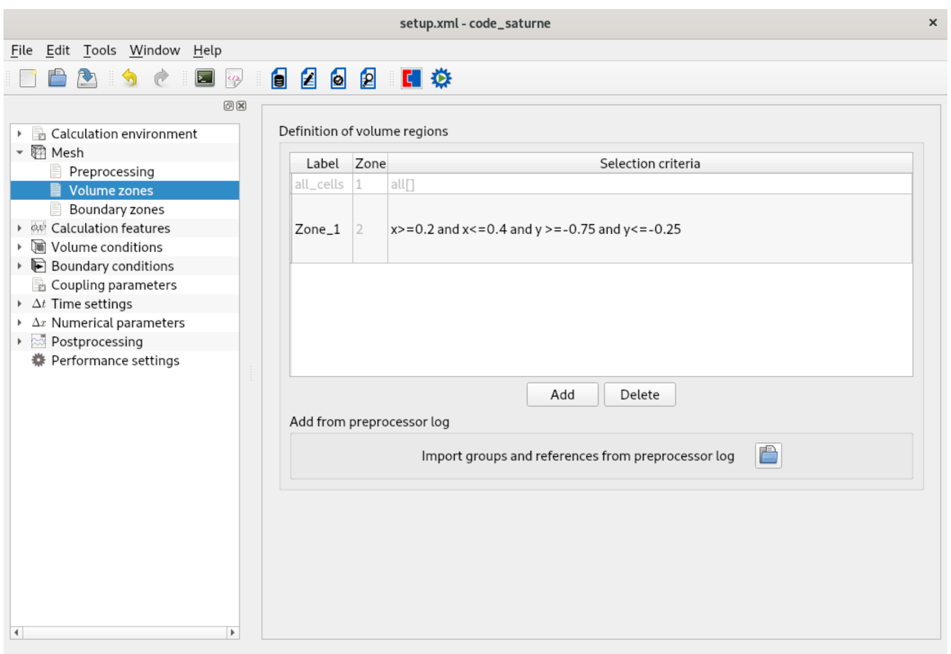


Figure III.36: Creation of the head losses region

3.2 Volume conditions tab

Head losses For this new volume you shall only tick **head losses** as follows. (see Figure III.37).

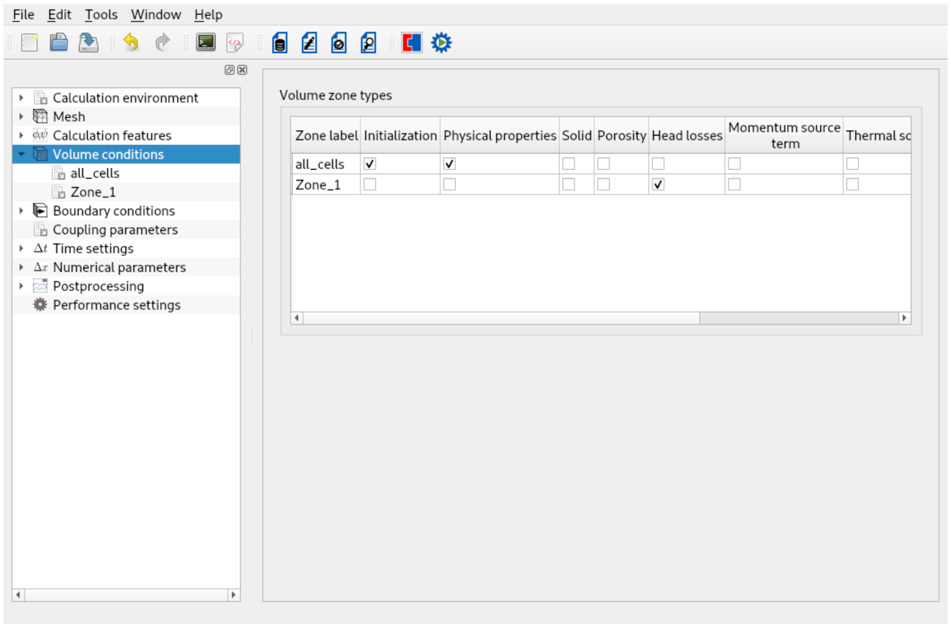


Figure III.37: **Selection criteria** of the head losses region

- Specify the head losses coefficients α_{ii}

To specify the head losses coefficients go to the **Head losses** item and select the name of the head losses volume region. In this example, the coefficient is isotropic so that we use the same value for each α_{ii}. Please note that $\alpha_{ii} = 2 \times K_{ii}$, therefore if $K_{ii} = 10^4$, $\alpha_{ii} = 2 \times 10^4$.

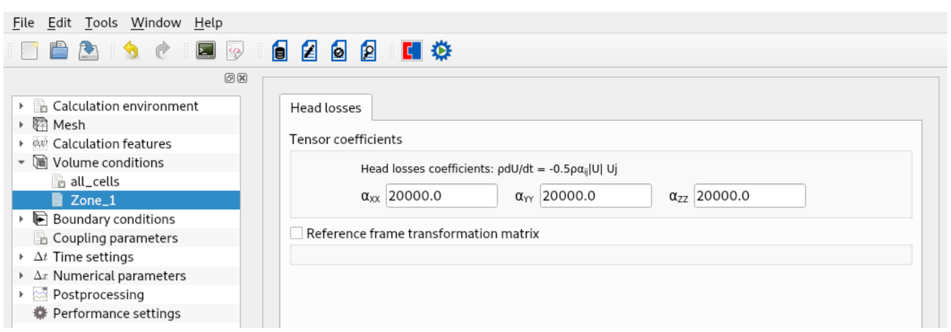


Figure III.38: Head losses coefficients

3.3 Time settings

This modification will be done in the **Calculation management** item.
To run the calculation on two processors, simply change the number of processors indicator to 2. The launch script will automatically deal with the rest.

Do not forget to set the right **Reference time step** and **Number of iterations** under the heading

Time settings as mentioned

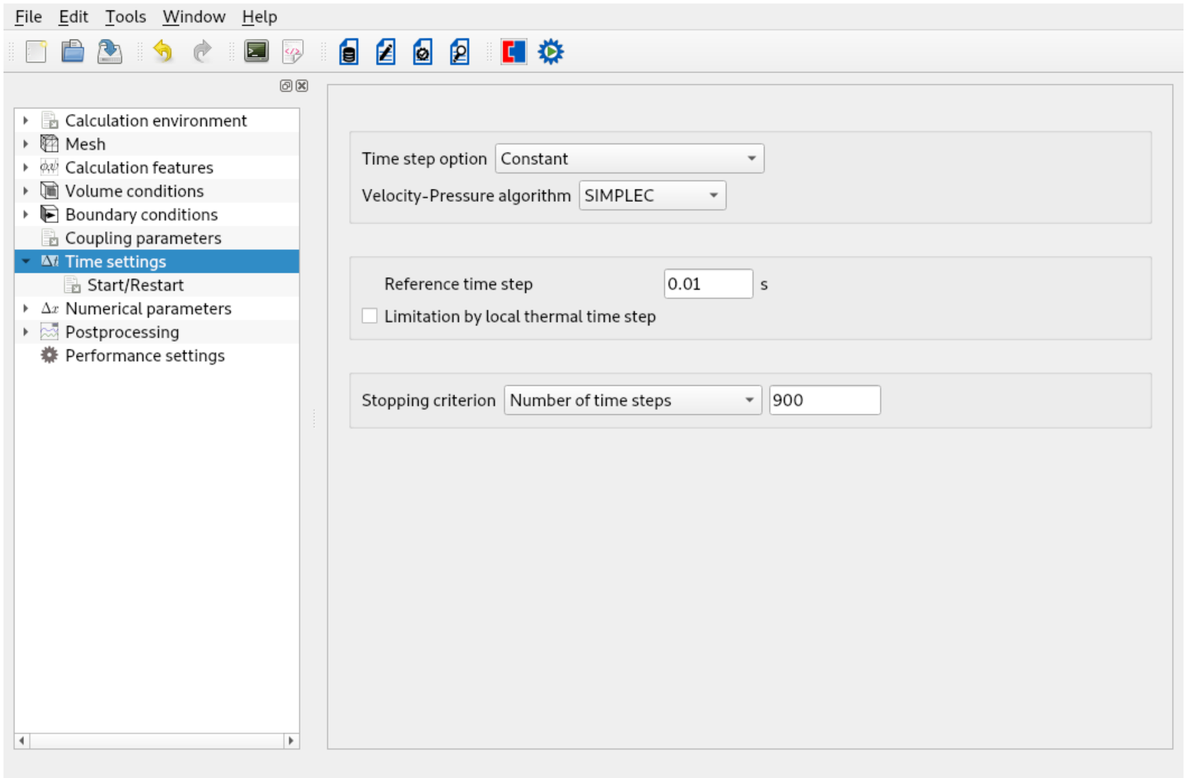


Figure III.39: Time settings

3.4 Run calculation

Verify you hardware configuration. Depending on your hardware configuration you might select either 2 processes (or more) if you either 2 threads per process.

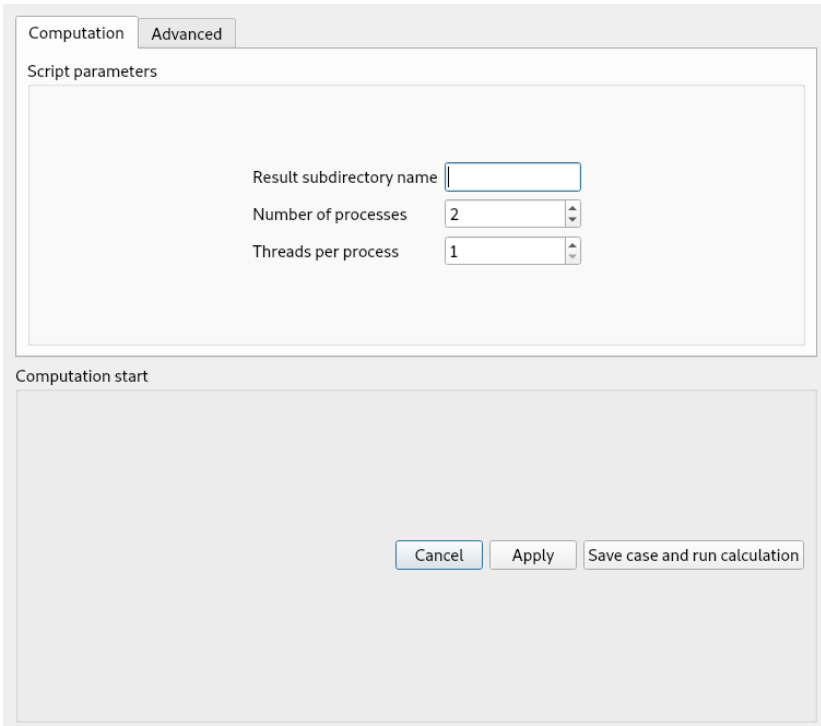


Figure III.40: Number of processors