The TRICLADE application

Dr. François Letierce

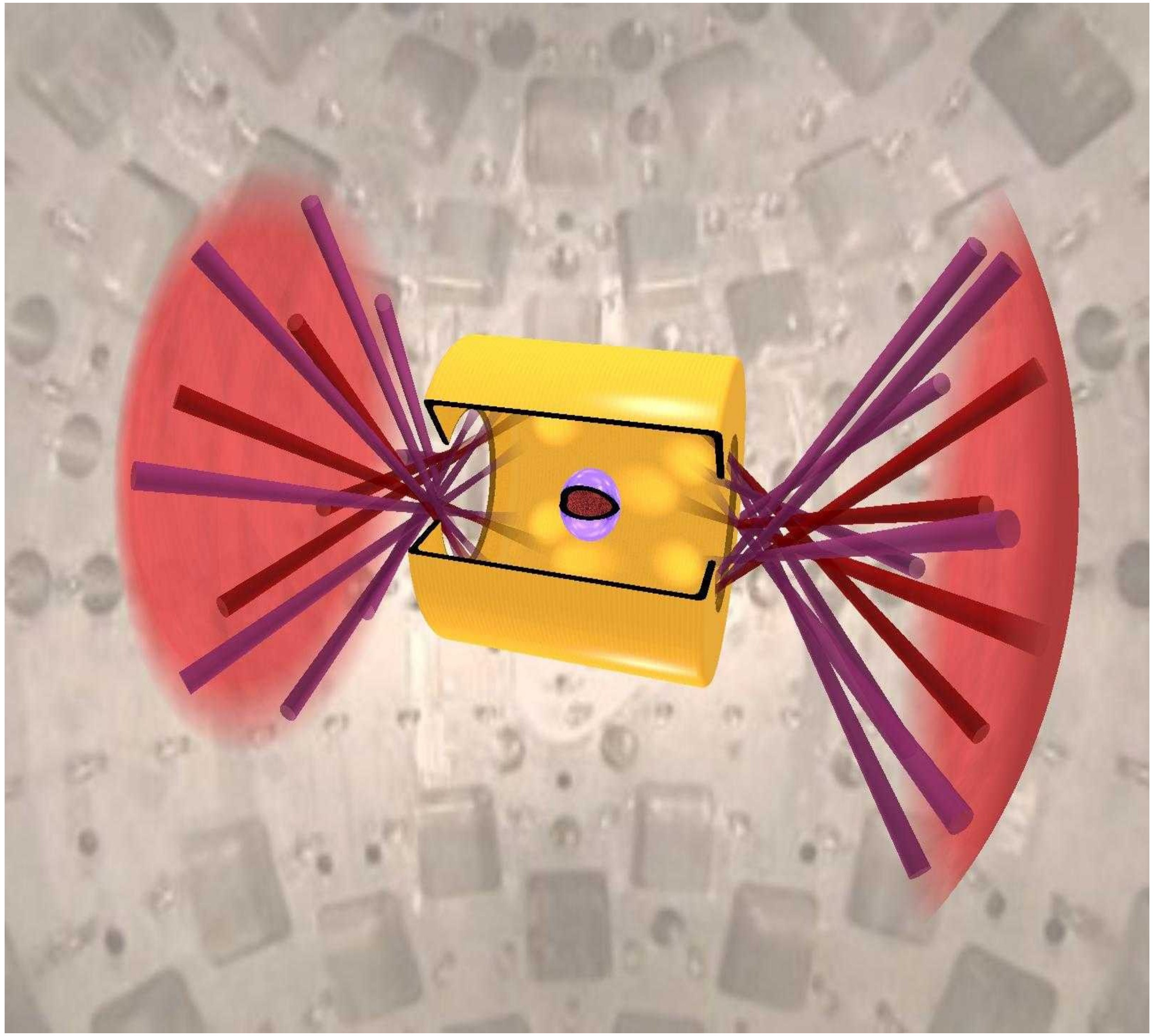
- *CExA Kick Off – 2023.09.19*

Introduction

<u>Turbulent mixing</u>

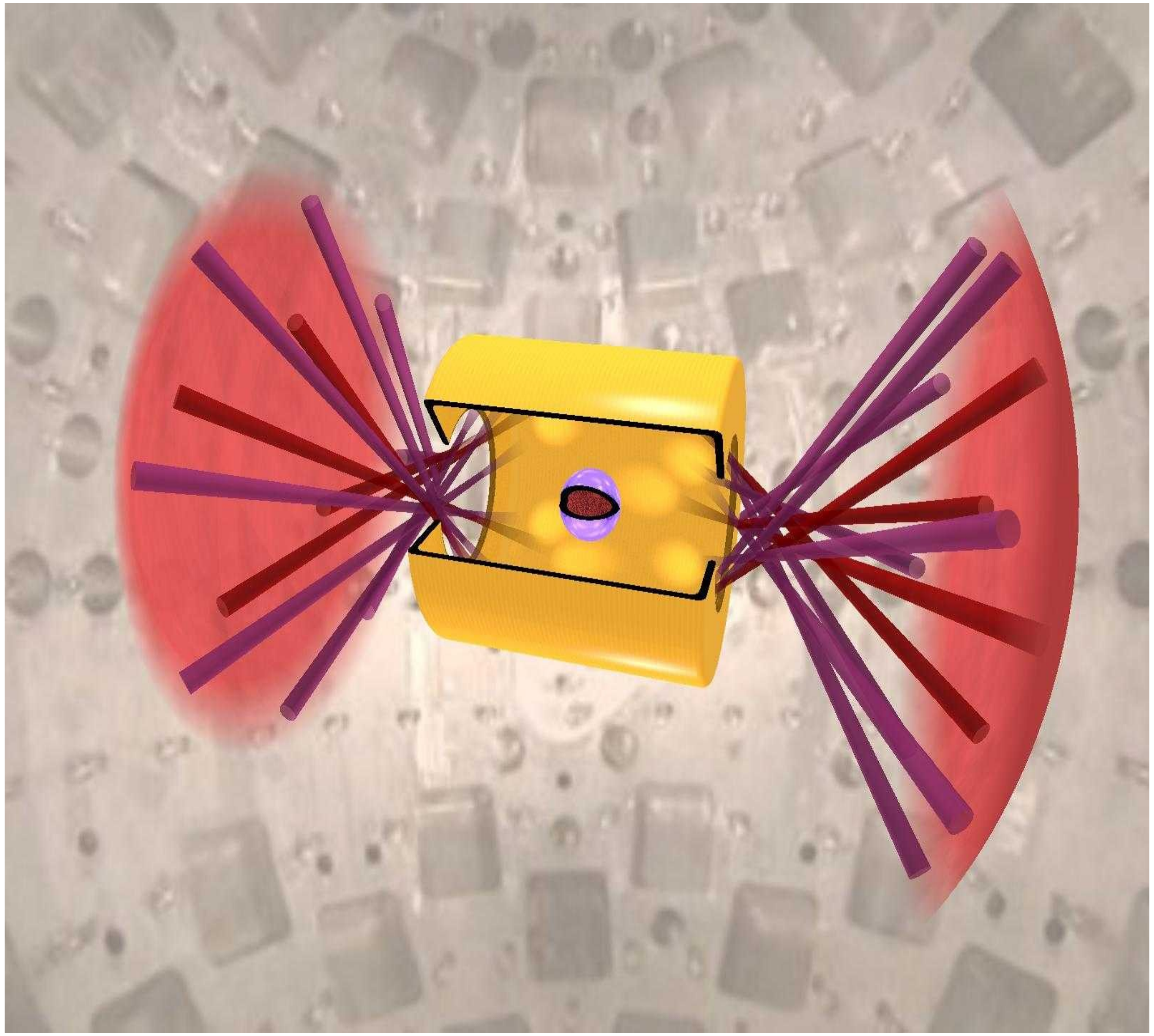
- Found in fields of interest to the CEA:
  - Astrophysics ;
  - Geophysics ;
  - Inertial Confinement Fusion ;
  - Etc.

- Very complex problem :
  - Intrinsically 3D ;
  - Multi-scale.



Introduction

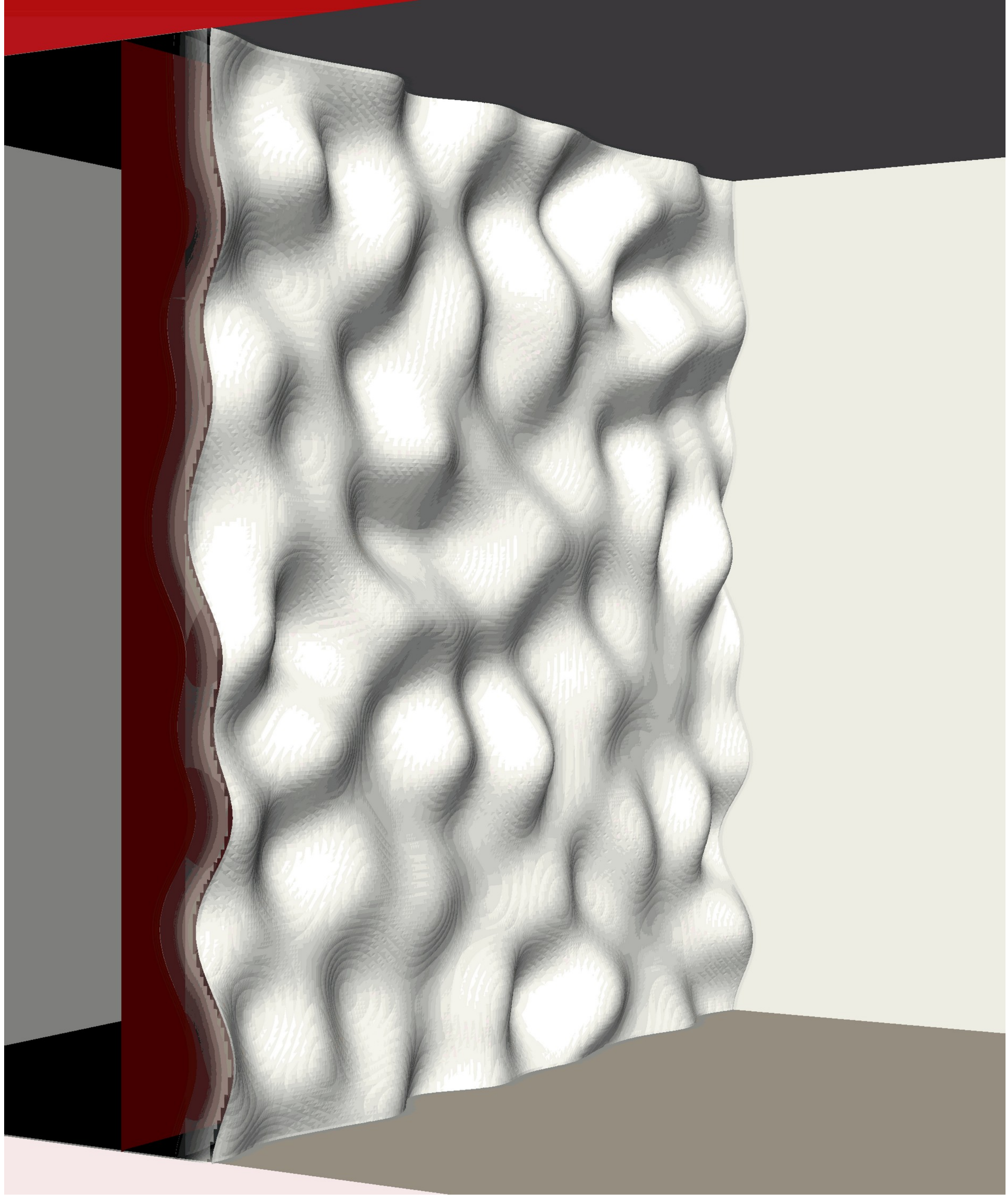
- Direct study in production codes is impossible:
  - Multi-physics ;
  - Highly complex geometries, ...
- Dedicated turbulence models:
  - Derived from theoretical concepts ;
  - Effects to be integrated into these codes ;
  - Calibrated and validated comparing results from experiments or numerical simulation.
- **TRICLADE** code serve as a pivot in this approach.



Some Context

- Study of Turbulent Mixing Zone:
  - Created and developed at fluids interface ;
  - From shock, expansion, acceleration, …
  - Dynamic and structure not fully understood.

- **TRICLADE**:
  - Turbulent binary mixing in a highly compressible environment
  - Navier-Stokes equations
  - Structured Cartesian Mesh
  - « Shock-capturing » numerical schemes



Some Context

- **TRICLADE:**
  - Wave Propagation" of order 5 in time and space: **WP5**
  - *[R.J. LeVeque. Finite Volume Method for Hyperbolic Problems. Cambridge texts in applied mathematics. Cambridge University Press, Cambridge, 2002]*
  - *[V. Daru and C. Tenaud. High order one-step monotonicity-preserving schemes for unsteady compressible flow calculations. Journal of Computational Physics, 193 :563–594, 2004]*
  - MUSCL of order 5 in time and 3 in space: **M5**
  - *[K. Kim and C. Kim. Accurate, efficient and monotonic numerical methods for multi-dimensional compressible flows. Part II : multi-dimensionnal limiting process. J. Comput. Phys., 208 :570–615, 2005]*
  - Both explicit
  - Boundary conditions = ghost cells beyond physical domain

Collaboration

- **TRICLADE:**
  - Developed and used in-use at CEA
  - Nationnaly with the **ISAE**
  - Internationnaly in the **ϴ-Group collaboration**
  - Numerous research papers published

Code information

- C++
  - Not really modern though…
  - ≈ 100 000 Lines of Code
  - Modular design
    - 1 module ≈ 1 numerical scheme
  - Depends on
    - Very little external libraries: MPI et FFTW
    - Lots of internal libraries for code environment

Code information

- Internal libraries:
  - Initial states management
  - Complex fields modifier
    - Can re-use them as inputs
  - I/O:
    - Own format,
    - Checkpoint / restart,
    - Pre / post-processing

- Scripts :
  - Help configure, run, etc.

Code information

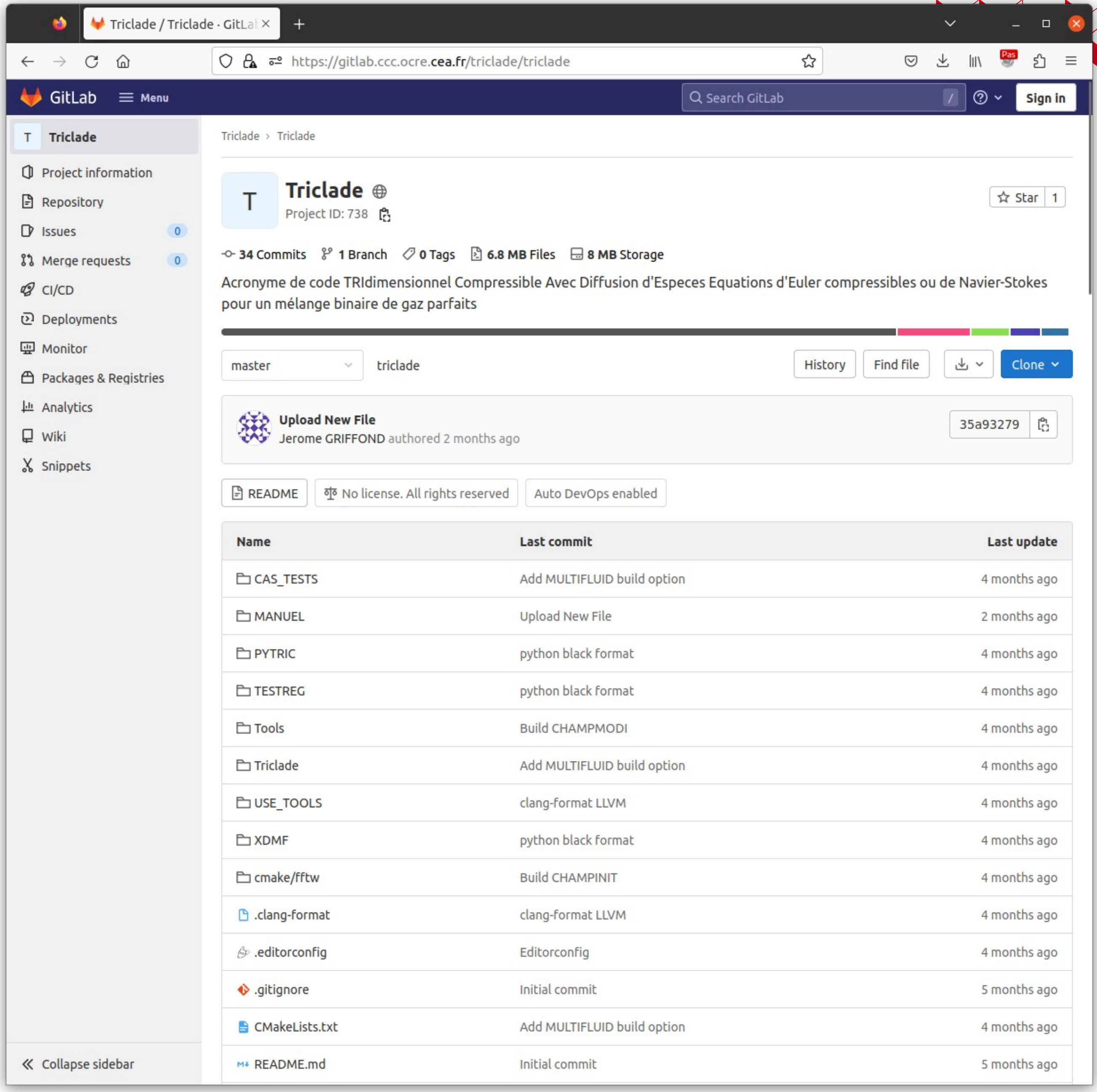
- Open Source soon ?
  - CEA Internal GitLab

https://gitlab.ccc.ocre.cea.fr/triclade/triclade

- Makefile ↔ Cmake

- Documentation
  - User manual
  - Developer manual

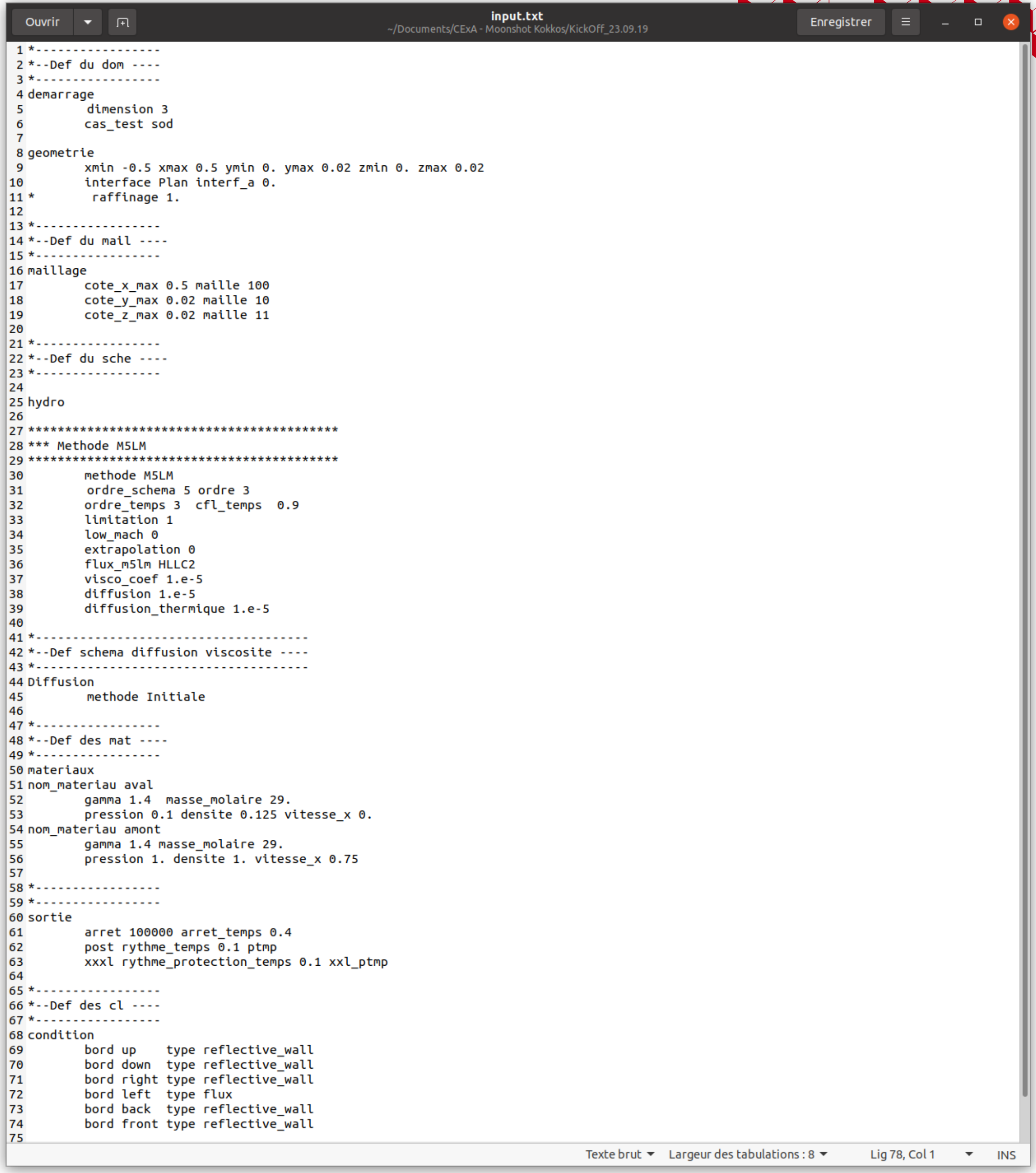
- A few examples



Code information

- Input files
  - Old plain text style...
  - Mesh (dimension, length, refinement, etc.)
  - Numerical scheme
  - Solver methods (time and space orders, etc.)
  - Boundary conditions
  - Material definitions
  - Other complex parameters
  - I/O, etc.

- Leads to beautiful outputs !

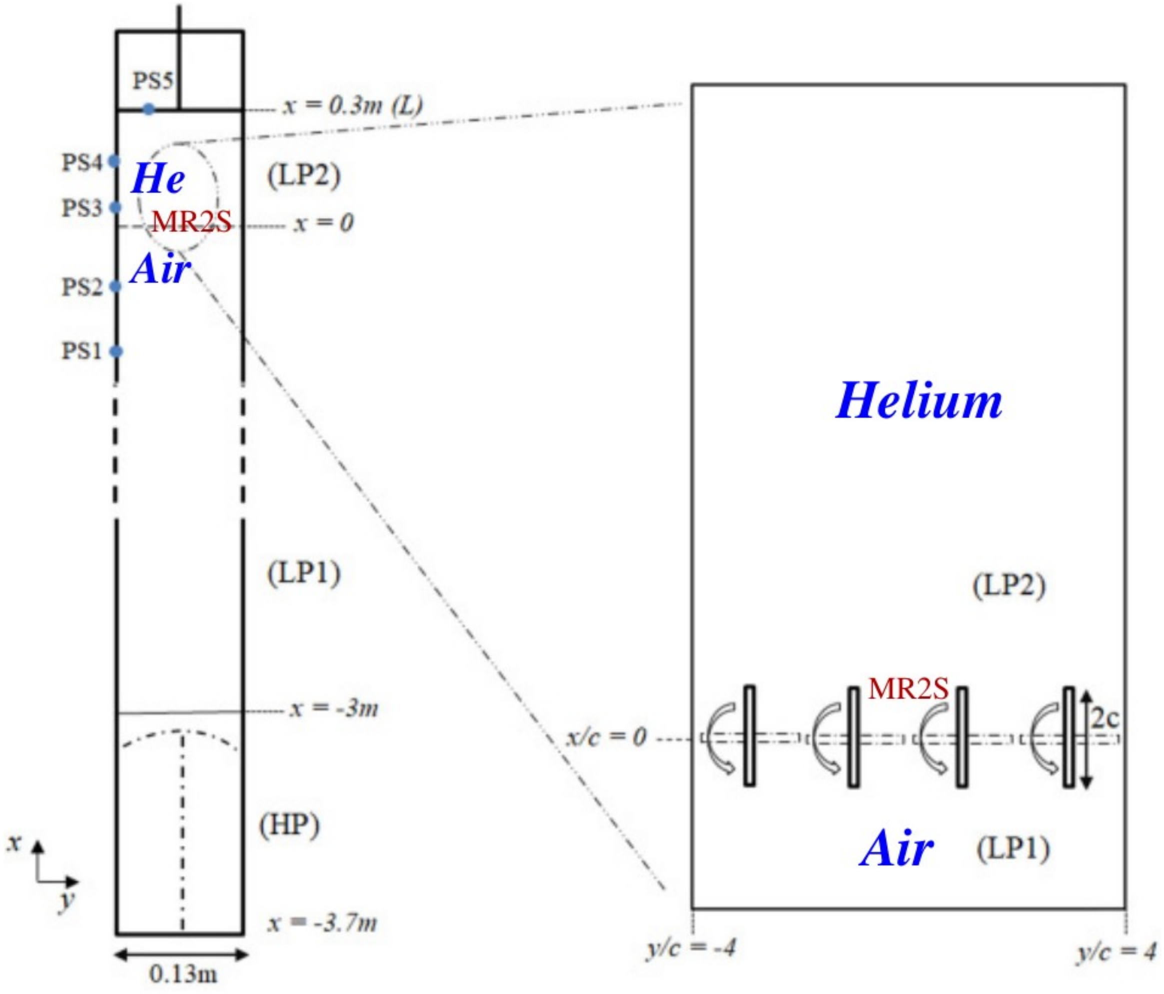



```
*----------------
*--Def du dom ----
*----------------
demarrage
        dimension 3
        cas_test sod

geometrie
        xmin -0.5 xmax 0.5 ymin 0. ymax 0.02 zmin 0. zmax 0.02
        interface Plan interf_a 0.
*        raffinage 1.

*----------------
*--Def du mail ----
*----------------
maillage
        cote_x_max 0.5 maille 100
        cote_y_max 0.02 maille 10
        cote_z_max 0.02 maille 11

*----------------
*--Def du sche ----
*----------------

hydro

*******************************************
*** Methode M5LM
*******************************************
        methode M5LM
        ordre_schema 5 ordre 3
        ordre_temps 3  cfl_temps  0.9
        limitation 1
        low_mach 0
        extrapolation 0
        flux_m5lm HLLC2
        visco_coef 1.e-5
        diffusion 1.e-5
        diffusion_thermique 1.e-5

*--------------------------------------
*--Def schema diffusion viscosite ----
*--------------------------------------
Diffusion
        methode Initiale

*----------------
*--Def des mat ----
*----------------
materiaux
nom_materiau aval
        gamma 1.4  masse_molaire 29.
        pression 0.1 densite 0.125 vitesse_x 0.
nom_materiau amont
        gamma 1.4 masse_molaire 29.
        pression 1. densite 1. vitesse_x 0.75

*----------------
*----------------
sortie
        arret 100000 arret_temps 0.4
        post rythme_temps 0.1 ptmp
        xxxl rythme_protection_temps 0.1 xxl_ptmp

*----------------
*--Def des cl ----
*----------------
condition
        bord up    type reflective_wall
        bord down  type reflective_wall
        bord right type reflective_wall
        bord left  type flux
        bord back  type reflective_wall
        bord front type reflective_wall

```

Example

ISAE shock tube : gases separation by rotative shutters

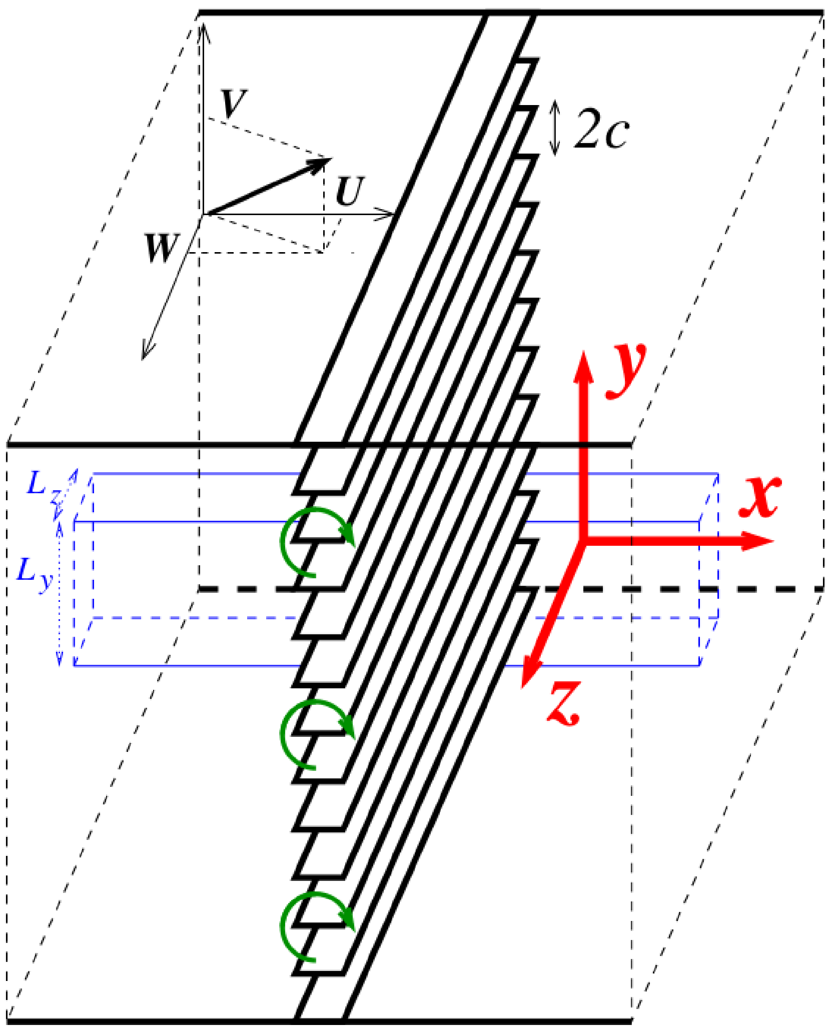
- Experimental set-up:
  - vertical shock tube :
    square cross section 13×13cm
  - interface air :
    (below LP1) / helium (above LP2)
  - incident shock waves of Mach 1.2 in air
  - adjustable end wall
  - initial gases separation :
    multiple rotating shutter system



Example

ISAE shock tube : gases separation by rotative shutters

- Numerical simulation set-up:
  - 3D-computations after complete opening
    → motionless shutters.
  - Domain: a fraction of the chamber:
    up to the top end
  - Blades treated = slipping rigid walls
  - Euler equations for binary mixtures of ideal gases
  - 3D-cartesian grid
    resolution 0.1mm (cubic cells)
  - Doubly-periodic boundary conditions along y



Example

tacisae21ms3l11wp5fz_blanc_strio.mp4

Courtesy of:
J. Griffond, O. Soulard, D. Souffland, Y. Bury, S. Jamme, M. Rasteiro dos Santos

Triclade & HPC

Turbulence mixing problem = high complexity + multi-scale

- Need large mesh : typical size is 1 Billion cells (1024^3)
- HPC is essential
  - Code is parallel : MPI domain decomposition
    - Uses same ghost cells technique as boundary conditions
      - Consistent results for any number of domains
    - Own decomposition (3 axes)
      - Do not use FFTW3 lib decomposition (1 single axis)
        - FFT → intensively @ fields' initialization and post-processing
  - I/O = MPI-I/O
    - Same process for sequential and parallel
      - You can change domains' sizes and numbers before restarting simulation
    - Compatible with internal libraries

Porting Triclade to GPU

Triclade GPU port was decided

Impacted modules are roughly 10 000 LoC

M5
Numerical scheme

5th
Space order

3rd
Time order

few
Limiters and extrapolation methods

hllc2
flux

Regardless of the CExA initiative

Focusing on currently most use features

+ yet to be discovered dependencies...

Triclade & Kokkos

- Prior experience of porting legacy applications to GPU
- Positive experience using Kokkos
  - academic projects,
  - R&D prototypes,
  - Miniapps, ...
- No vendor dependent politic
- Performance portability
- Kokkos is the obvious choice !
  - good timing to use Triclade as a stepping stone for CExA

Battle plan

1st step: change data structures

- "Variables": multi-dimensional arrays of POD.
  - "Primitive": physical variables, lifespan of the program.
    - E.g: pressure, etc.
  - "Conservative": linked physical variables, lifespan of a class or method.
    - E.g: internal energy, etc.
- → Kokkos::View (?)

- Collection of variables
  - Looks like: double**** tab; where dimensions are [VAR][X][Y][Z]
- → Kokkos::View<double****> ?
- → std::array<Kokkos::View<double***>, VAR> ?
- → ?

Battle plan

2nd step: change compute loops

- Rewrite classical loops → parallel dispatch
  - Functors / Lambda functions
  - Use parallel_for, etc.

- Early study → possible use of hierarchical parallelism
  - Some computations seems “axis independent”
- → Kokkos::TeamEtc…

Conclusion

Triclade already fulfills its role as a demonstrator application in the CExA project, popping some interesting questions:

- Can I map variable to something more advanced than Kokkos::View
  - Properties
  - Lifespan management

- What should I use for collection of variables ?
  - Can I have memory pools for these ?
  - Can I use properties to filter variables ?
  - Can I have some batch processing ?

- This is only the beginning...

Thank you for your attention

Dr. François LETIERCE

CEA DAM Île-de-France

Bruyères-le-Châtel

91297 Arpajon cedex
France

francois.letierce@cea.fr