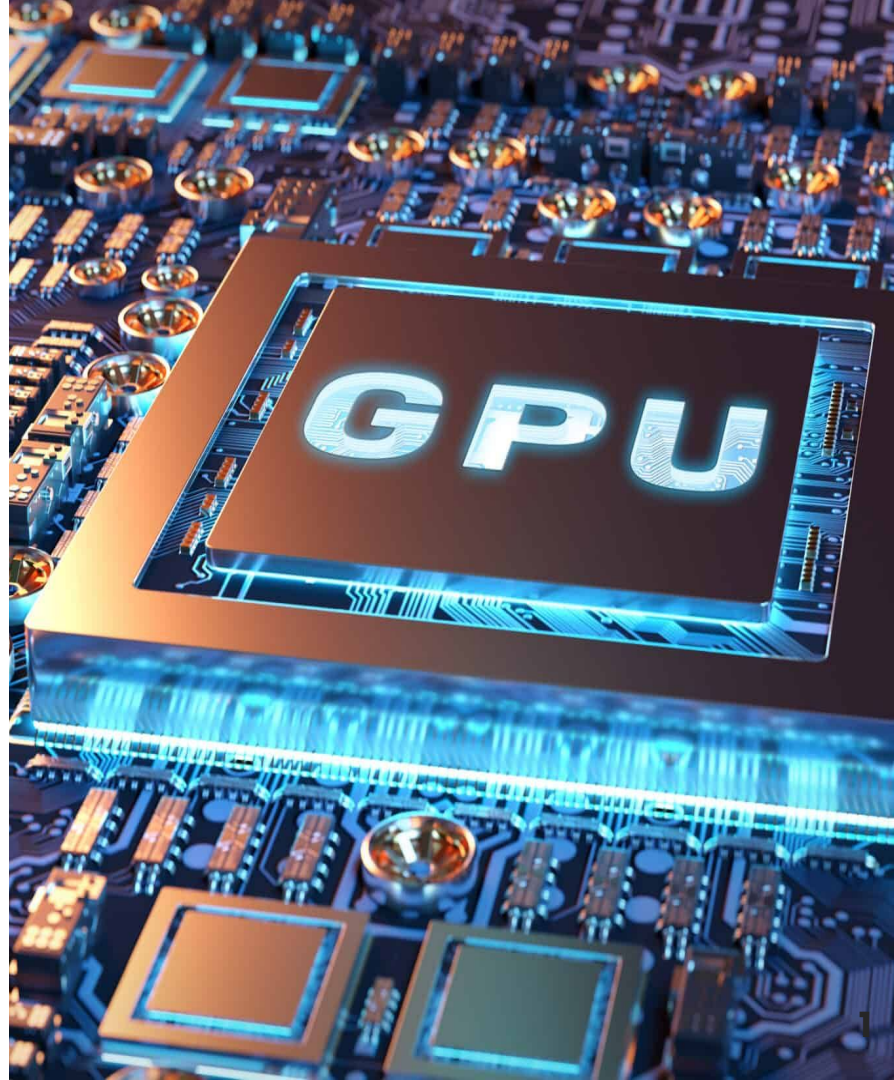




# **CExA “moonshot” project Computing at Exascale with Accelerators at CEA**

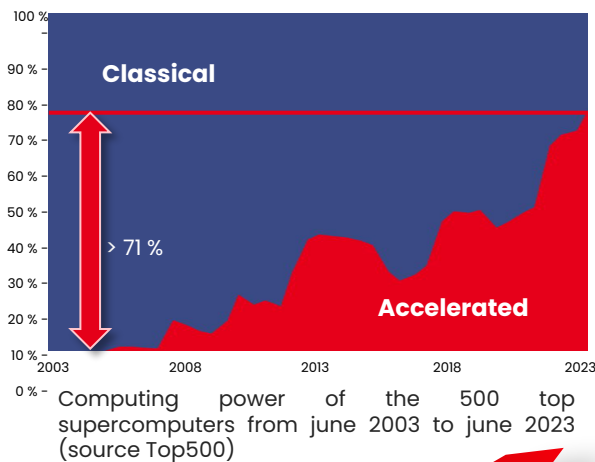
Software catalyst for GPU computing



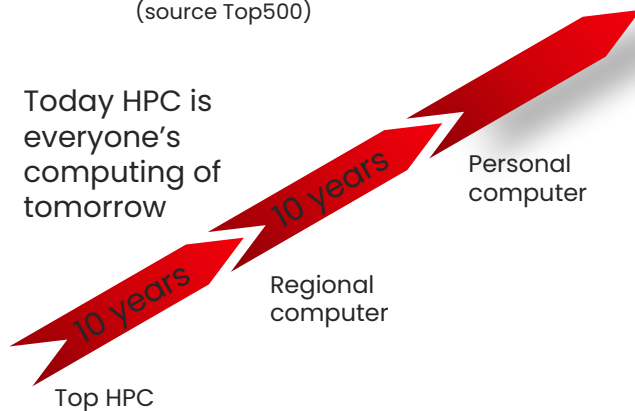
*Kokkos user group – 12 December 2023  
Julien Bigot & CExA team*

# Context

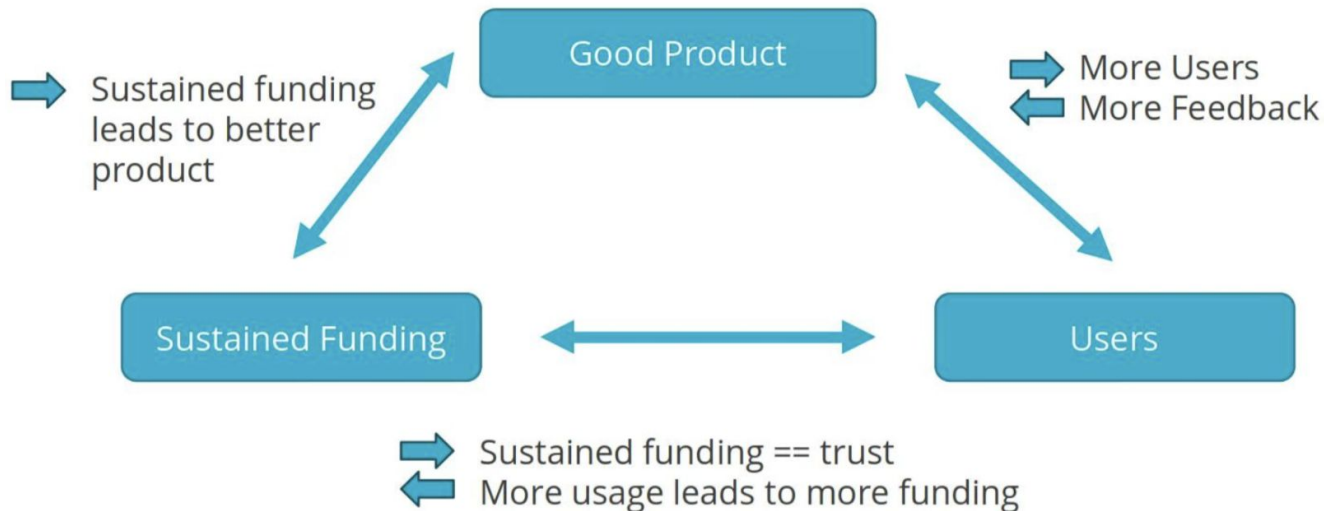
- **CEA**: French Atomic Energy Commission (French DoE)
  - Around **20k researchers**, 9 research centers all over France
  - Organized in 4 divisions: military applications (DAM), energies (DES), fundamental research (DRF) & technological developments (DRT)
  - HPC is a tool largely used all over CEA, source **competitvity**
- We just entered the **Exascale** era, that means **GPU**
  - European pre-Exascale systems: Mix of AMD & Nvidia
  - First Exascale machines planned in Europe for 2024/2025
    - Jupiter machine at Jülich => Nvidia & Rhea
    - Jules Vernes, French machine at **CEA/TGCC** (open)
  - Need to re-develop applications with **Performance portability**
- GPU middleware: **software catalysts**
  - France and Europe: great research but no production tool
  - App developers are sitting on Buridan's ass
- A **need** for a long-term sustainable solution
  - **Adapted** to our hardware and software specificities
  - **Trust** in the roadmap



Today HPC is everyone's computing of tomorrow



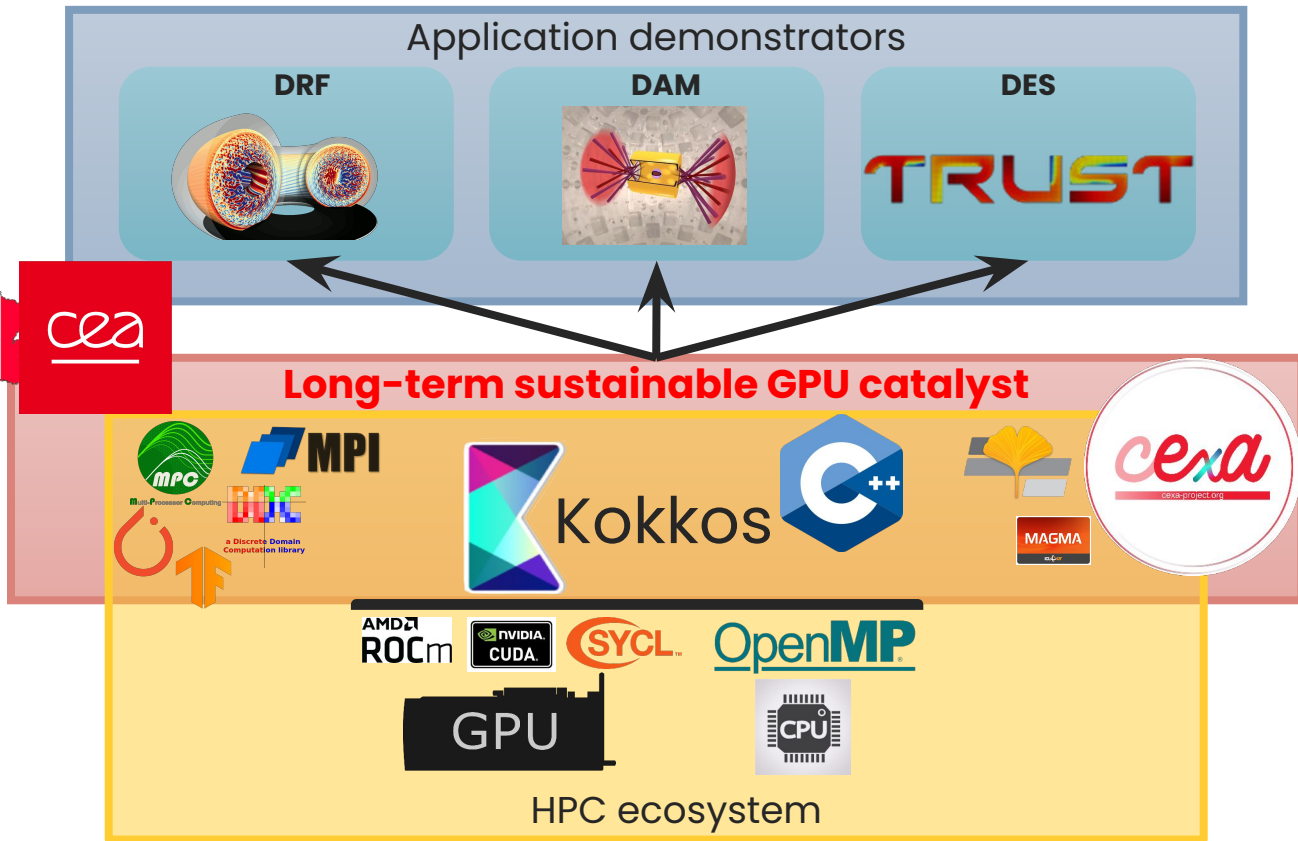
# Sustainment: a self-reinforcing circle



**There is strength in numbers: collaboration on core product good for everyone!**

© Christian Trott &  
Damien Lebrun Grandie

# The project



Disseminate and offer training in CEA and at large

Adapt application demonstrators

Provide a long-term sustainable software catalyst for GPU computing

# CExA in short

“**adopt and adapt**” strategy based on  Kokkos

- Kokkos : a **strong technical basis**
  - A software architecture ready for the future
  - Mature, free, libre, and open-source
  - An **independent foundation** to own the product
    - Under the Linux Foundation
  - A **standardisation** effort in **ISO C++**
    - A **stepping stone** one step ahead toward **parallel C++**
- Some **adaptations required**
  - For European **hardware**
    - There is no real hardware sovereignty without software sovereignty
  - For **applications** from CEA, France and Europe
    - Take our specificities into account

## Adaptation to our architectures

- Efficient memory transfers
- **Multi-architectures compilation support**
- Rhea processor support

## Simplification of deployment on our computers

- **Continuous integration on our computers**
- **Installation on the national computers**
- **Interfacing with MPI**

## Hardware specificities

## Interface with external tools

- **Interface with JAX, Pytorch, Tensorflow**
- **Batched linear solvers**
- **FFT, splines, ...**
- **ONNX compilation**

## Adding new features

- **Support for named dimensions**
- **Handle numerical precision and stability questions**

## Software specificities



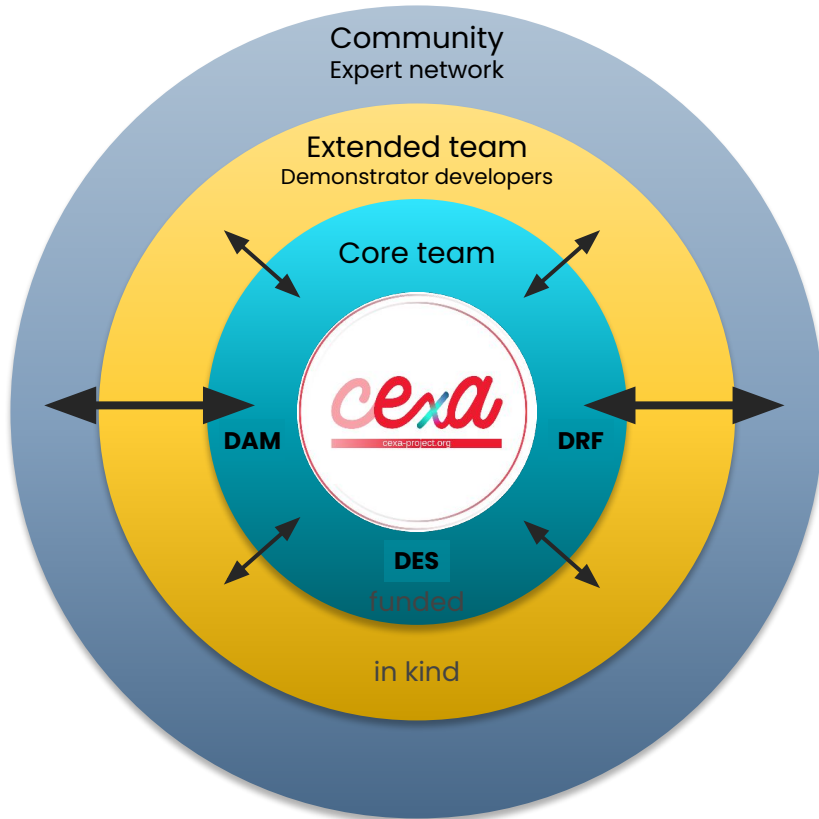
# HIGH PERFORMANCE SOFTWARE FOUNDATION



Tentative Founding members

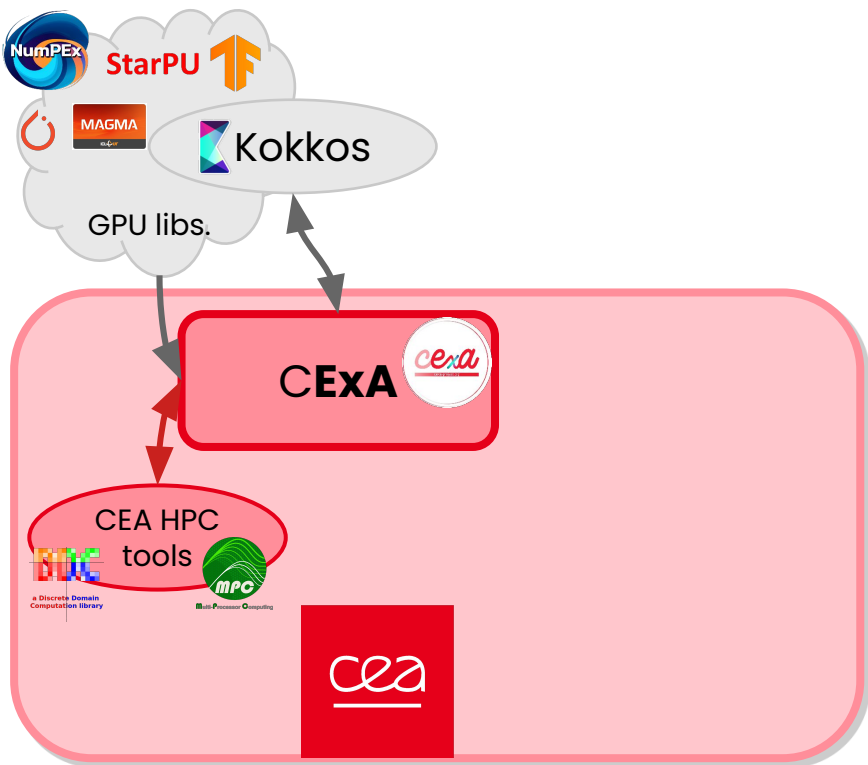


# Project organization



- **Core team**
  - Management, implementation and dissemination
  - 8 permanent researchers from all over CEA
  - 4 recrutements done, looking for 2 more
    - 1 as a permanent researcher !
  - Funding for 3 more hire expected next year
- **Extended team**
  - Demonstrator developers
    - Not funded
    - Find their interest in the participation
  - 3 new demonstrators to be selected next year
- **Community**
  - Federation of an expert network
  - Co-design of CExA:
    - Identification of needs
    - Usage of CExA in applications
  - Priority target for dissemination
  - Sustainability of the work

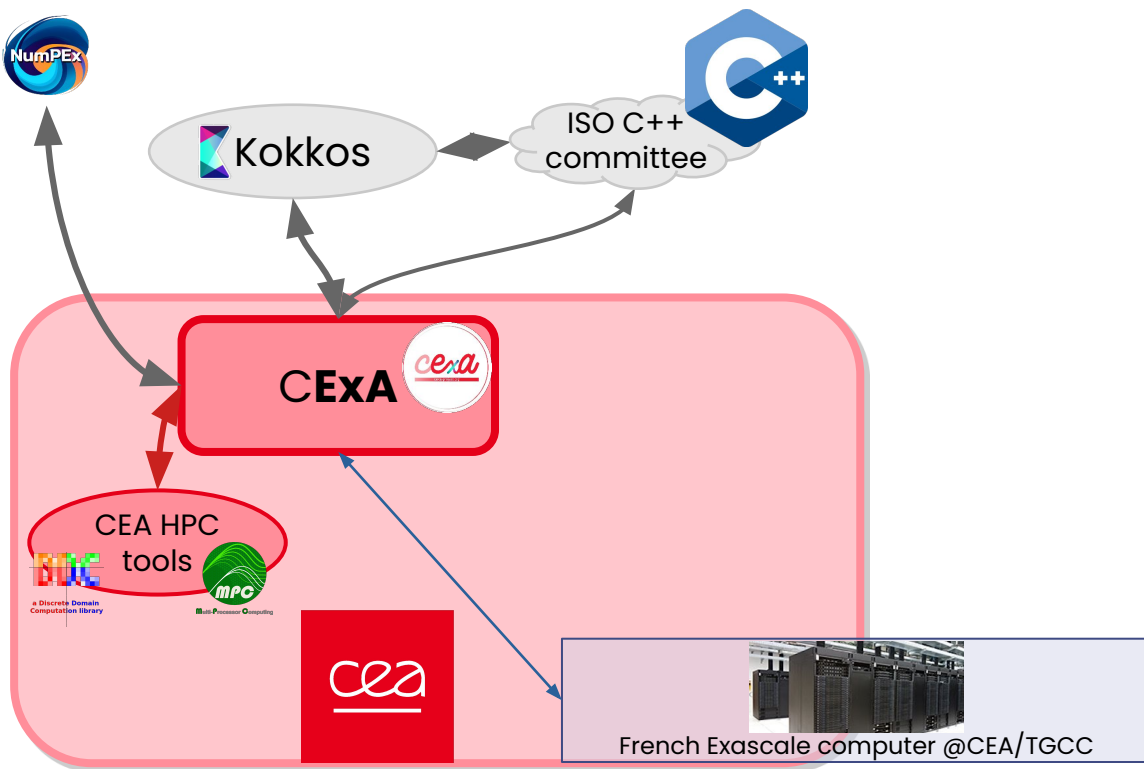
# CExA ecosystem: the upstream



- Kokkos library
  - Our main upstream
  - Kokkos-core but also the whole Ecosystem
- Existing CEA HPC libraries
  - MPC, DDC, Arcane, etc.
  - Integration and exchanges
- Existing GPU libraries
  - Tensorflow, Pytorch, MAGMA, etc.
  - Interfacing thanks to the free, libre, open-source strategy
  - Together with NumPEX PEPR

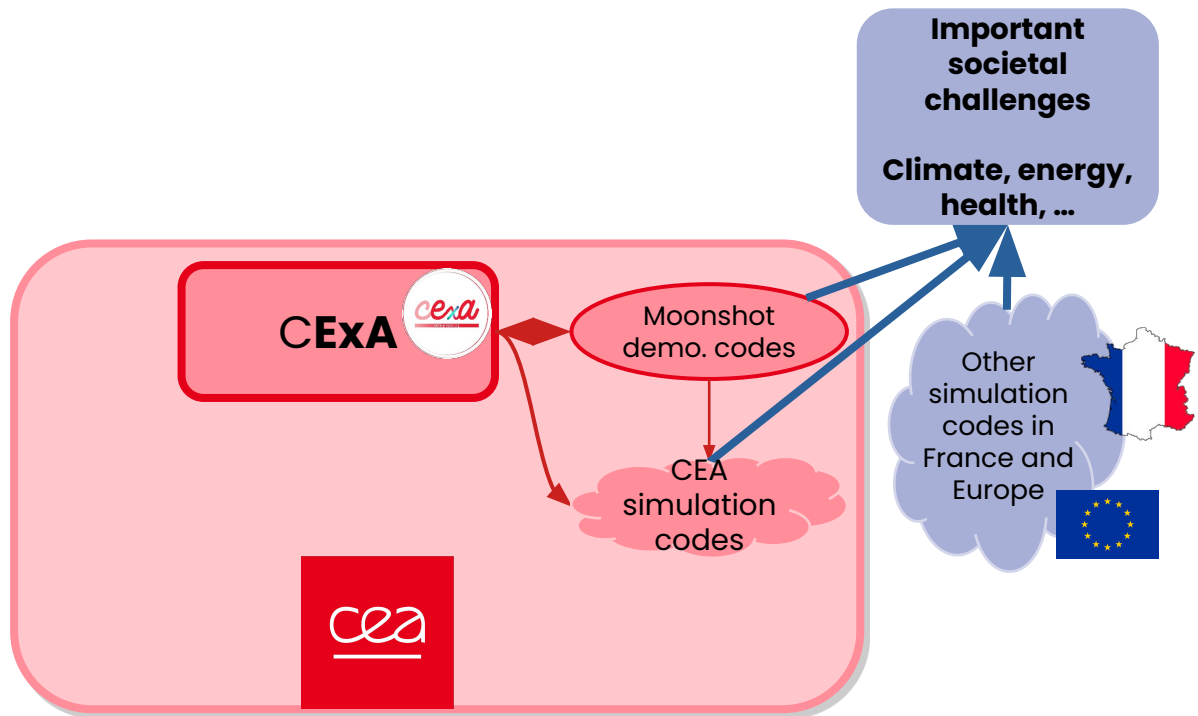


# CExA ecosystem: our partners



- Kokkos & ISO C++ committee
  - A strong relationship
  - Standardisation
    - through Kokkos
  - Normalisation for sustainability of developments
- Jules Vernes project (Exascale France)
  - Strong links established with GENCI, TGCC, and NumPEX
  - Call for proposal soon
    - Including **CExA** requirements
  - Answer in 2024
    - Selection of the target architecture
  - Delivery in 2025/26
    - **CExA** ready for production

# CExA ecosystem: the downstream



- Two stage downstream
  - The acceleration stage ⇒ applications
  - The final stage ⇒ societal challenges
- Integrated demonstrators
  - Co-development
  - Training of the teams
  - Impact in important domains
- CEA applications
  - Training, hackathons, gain of experience
  - A clear choice ⇒ knock-on effect
  - Creation of a community
- FR and EU community
  - Visibility et place of CEA

# The Triclade demonstrator (DAM)

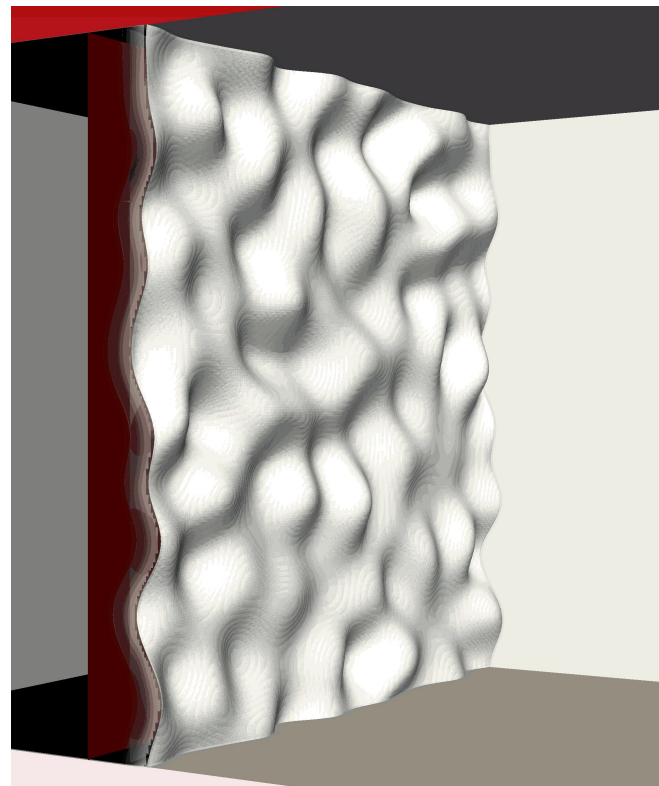
Turbulent mixing

Found in many fields of interest of the CEA:

- Astrophysics ;
- Geophysics ;
- Inertial Confinement Fusion ;
- Etc.

Very complex problem :

- Intrinsically 3D ;
- Multi-scale.



# Code information

- C++
- Not really modern though...
- $\approx 100\,000$  Lines of Code
- MPI domain decomposition
- Modular design
  - 1 module  $\approx$  1 numerical scheme
- Depends on
  - Very few external libraries: MPI & FFTW
  - Lots of internal libraries for code environment



**FFTW**

**MPI**

# Porting Triclade to GPU

Goal: explore the Kokkos way

Triclade GPU port was decided

Impacted modules are roughly 10 000 LoC



Regardless of the CExA initiative

Focusing on currently most use features

+ yet to be discovered dependencies...



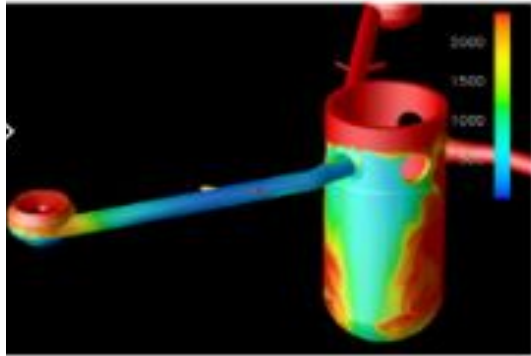
**kokkos**

# The TRUST/TrioCFD demonstrator (DES)

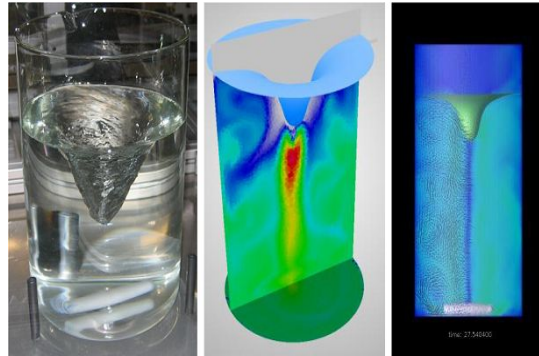
**TRUST** A thermohydraulic platform

**TrioCFD** A TRUST-based application dedicated to CFD

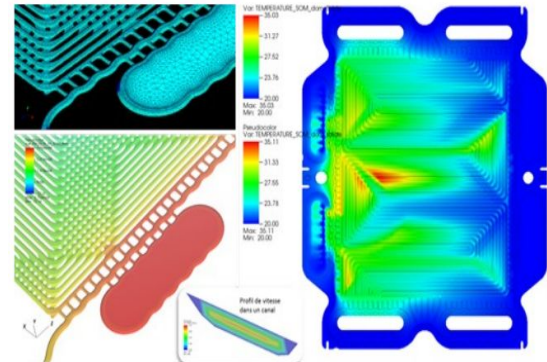
- Fluid mechanics (low/incompressible, mono/multiphase, interface follow)
- For multiple domains



PWR Reactor



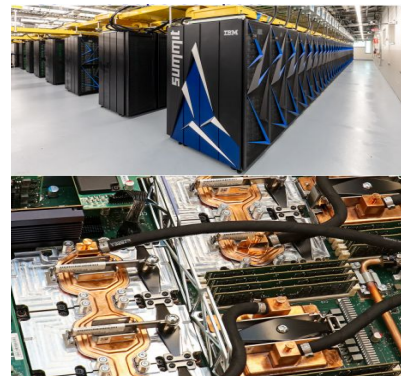
Vortex mixer



Fuel cell

- C++, MPI, OpenSource <https://github.com/cea-trust-platform>
- Many other applications based on TRUST: FLICA5, SympyToTRUST, CATHARE3D, Trio-IJK, TrioMC , GENEPI+, PAREX+

# TRUST/TrioCFD roadmap for GPU computing



2014

- **First** use of GPU in TRUST (**PETSc**)
  - Single node GPU, limited to one solver (GMRES/Jacobi)

2020

- Test **AmgX**, Nvidia GPU library
  - Multi-node GPU, more solvers available (CG/Multigrid)

- Port TRUST on **ARM** architecture

2021

- Add **AmgX** library (Nvidia) to TRUST (1.8.3)
- Nvidia Hackathon participation
  - Challenge TRUST team to evaluate **OpenACC** approach (parallel pragma directives)

2022

- First study with a GPU **partial** accelerated **TrioCFD** (Jean-Zay)
- Partial port on AMD GPU with OpenMP on Adastral (GENCI contract)

2023

- First run with a GPU **fully** accelerated TRUST (Topaze)

2024

- Enable **CExA** (**Kokkos** framework at CEA) in TRUST/TrioCFD

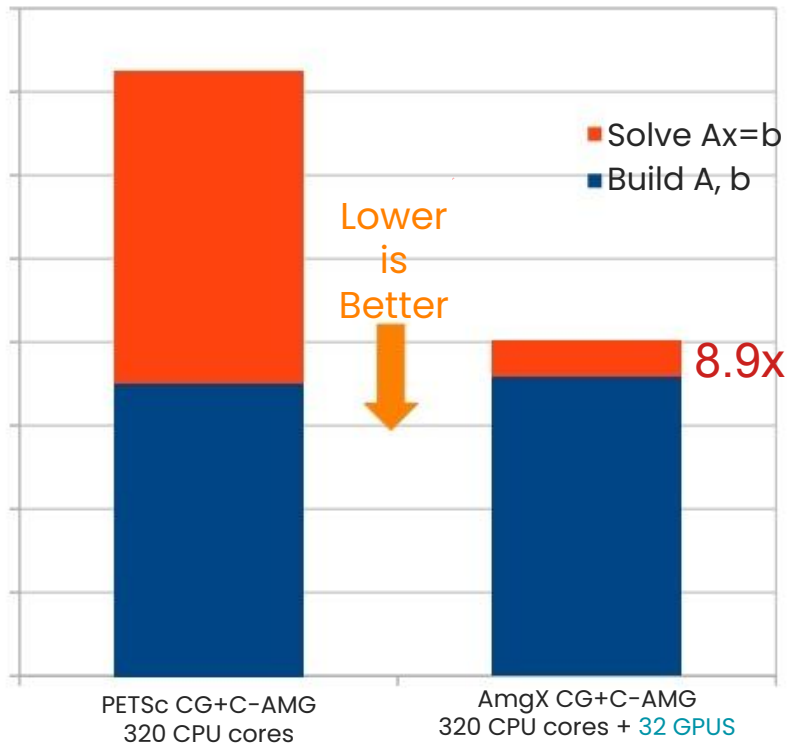
2025

- French exascale supercomputer (**ARM** CPU? / AMD/Intel/Nvidia GPU?)

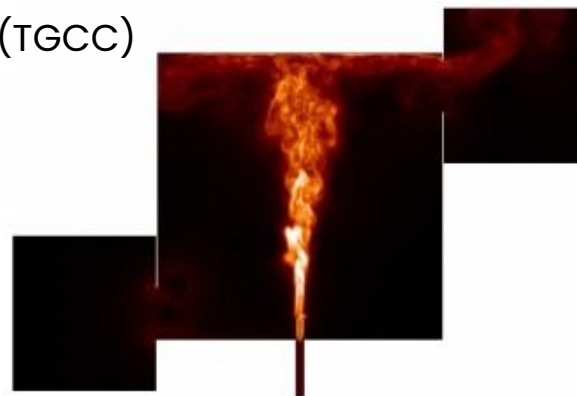


# Experimenting with AmgX solver

DNS simulation (TrioCFD 1.8.3) on Irene Joliot cluster (TGCC)



**1.8x** acceleration for the simulation



- Mini-GAMELAN geometry
- Structured mesh (VDF)
- 80M cells (250K/core)
- Unsteady DNS
- GC + C-AMG solver
- 50% time into solver

# Current state of TRUST/TrioCFD porting to GPU

- Matrix assembly ported w. OpenMP-target
  - 70 kernels ported
- TRUST platform adapted for GPU
  - Automate CPU↔GPU transfers
  - Launch Kernel
- Most data remain on GPU
  - 90% on GPU 5% copies H→D & D→H (mostly IO on CPU)
- MPI device-to-device communications tested and **required** for performance
  - Not very reliable (installation, machine specific aspects are an issue)

The OpenMP model is **not well suited** for C++, compilers sometimes fail

Going **with the community**, **sharing** solutions is critical: Kokkos with CEXA!



# TRUST on GPU



TRUST  
on GPU

- A **small** subset of TRUST ported to GPU
  - **Enough** for some demonstration & benchmarking computation

# TRUST on GPU

Goal: Incremental porting of a critical platform to GPU

TRUST  
on GPU

TRUST

TrioC  
FD

Triol  
JK

Flica5

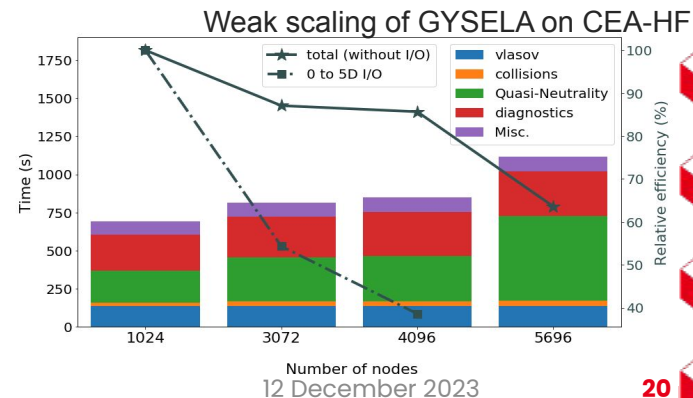
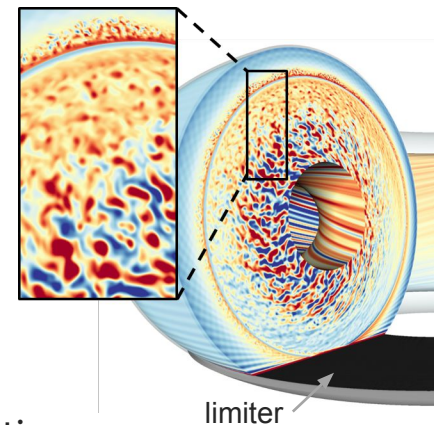
Batte  
ries

Cat  
har  
e3  
D

- A **small** subset of TRUST ported to GPU
  - **Enough** for some demonstration & benchmarking computation
- **But:**
  - Many TRUST elements missing: meshes, numerical schemes, models, most physics operators , ...
  - No actual TRUST-based application ported
  - Still a long road ahead

# The GyselaX++ demonstrator (DRF)

- Gyrokinetic codes require **state-of-the-art HPC techniques** and must run efficiently on **several thousand processors**
  - **Non-linear 5D simulations** (3D in space + 2D in velocity)
  - + multi-scale problem in space and time
- Even **more resources** required when modelling **both core & edge** plasmas like GYSELA
- GYSELA = Fortran 90 code with **hybrid MPI/OpenMP** parallelisation optimized **up to ~1.5M threads**
  - Relative efficiency of **85%** on **more than 1M threads**
  - and **63%** on **~1.5M threads** on CEA-HF (AMD EPYC 7763)
- Intensive use of petascale resources:
  - **~ 150 millions of hours / year**
  - GENCI + PRACE + HPC Fusion resources





# How to prepare GYSELA to HPC exascale architectures ?

□ Huge efforts of optimization and porting during EoCoE-II




Target: 3 different architectures in the top 20

*Porting in 2021-2022 via CEA-RIKEN collaboration and GENCI support with ATOS*

*Porting in 2022-2023 with HPE and EOLEN in the frame of ADASTRA Contrat de Progrès at CINES and with SCITAS-EPFL in the frame of EUROfusion Advanced Computing Hub*

*May 2022: Opportunity to run during « Grand Challenge » campaign*

Rank	 NOVEMBER 2022	Cores	Rpeak (Pflop/s)
2	<b>Fugaku</b> – A64FX 48C, Fujitsu - RIKEN Center for Computational Science – <b>Japan</b>	7,630,848	537.21
11	<b>Adastr</b> a – HPE Cray, <b>AMD Instinct MI250X</b> - GENCI-CINES – <b>France</b>	319,072	46.10
20	<b>CEA-HF</b> – BullSequana XH2000, <b>AMD EPYC 7763</b> - CEA – <b>France</b>	810,240	23.24

**Impossible without HPC experts**

■ Operator refactoring (collisions, sources) + Performance optimization at node level (vectorization, blocking, asynchronous MPI communications) □ **Gain > 70%**



Good performance on the 3 architectures with same Fortran code via OpenMP directives

□ Not feasible without rewriting, duplication of most of the kernels

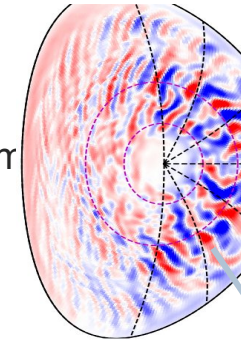
# Roadmap for GyselaX++ towards exascale

## □ Why do we choose to rewrite GYSELA ?

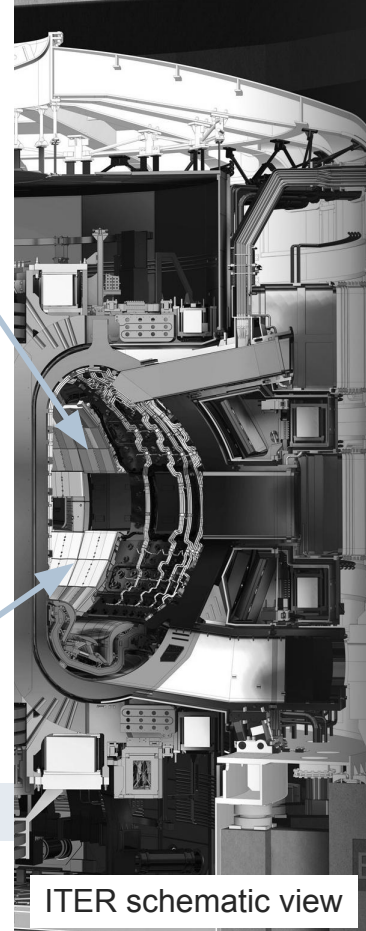
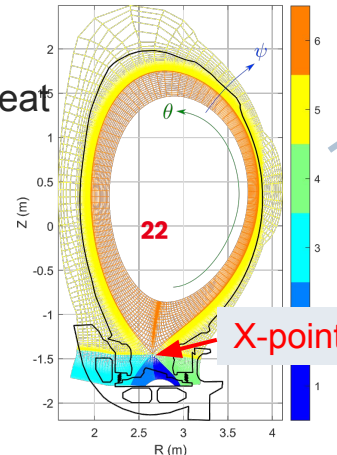
- 20 years-old code written in Fortran with hybrid MPI/OpenMP parallelism
- Unique code for both CPU (AMD milan or ARM-A64FX) and GPU with OpenMP directives is NOT optimal
  - extremely difficult to optimize on all architectures.
- Non-equidistant mesh mandatory for core-edge-SOL turbulence simulations
  - Modifying splines in GYSELA = rewrite most of the kernels
- X-point geometry
  - Development of new semi-Lagrangian scheme required to treat multipatches

Simpler to rewrite main kernels in modern C++ from scratch

□ GyselaX++ code



GYSELA  
D-Shape  
geometry

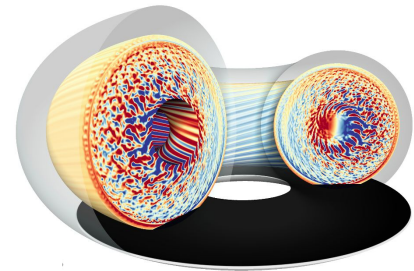
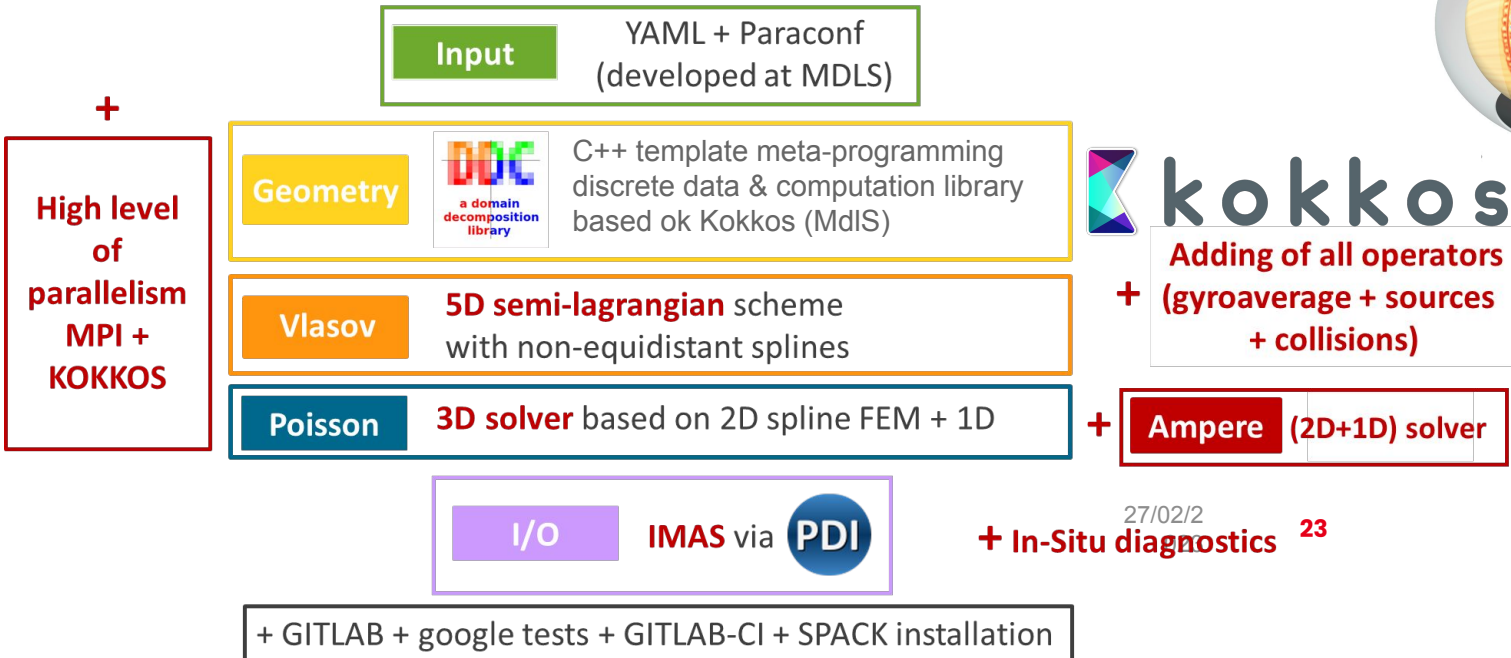




# Gysela-X++: towards exascale

□ Complete rewriting of the code in modern C++

5D code in modern C++ scalable on exascale architectures



**Goal: Rewrite a flagship code for GPU**

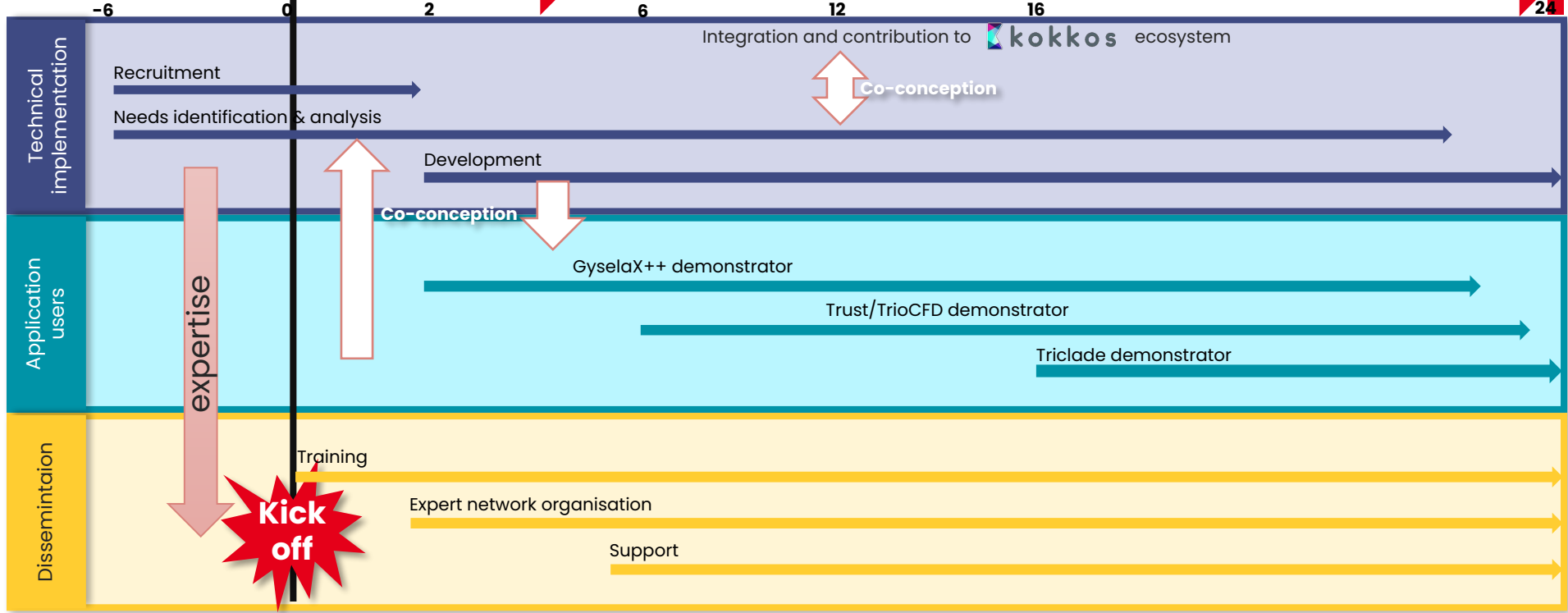
# Planning

Preliminary setup

Funding: 2 years

Start: Sept. 2023

End?: Août 2025



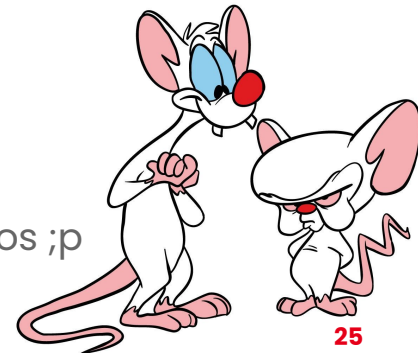
# CExA: what's going on?



**We have stickers!  
(not today)  
Come see us @ ISC**



- Training
  - First training by Christian & Damien after Kikkof
  - Another internal CEA training
  - Submitting a tutorial to ISC
- Community animation
  - A few newcomers on slack #general-fr
  - CExA virtual café every 2 weeks (maybe not the best time from the US)
  - Monthly Kokkos korner
- Working on our first features
  - Kokkos-FFT => **today 3:30 PM**
  - DDC: Discrete data & computation => **tomorrow 3:50 PM**
  - Shaman: numerical sensibility analysis => **today 4:10 PM**
- Joining the effort of bugfixing, documentation & all the fun!
- Help DoE implement the ongoing European hostile takeover of Kokkos ;p



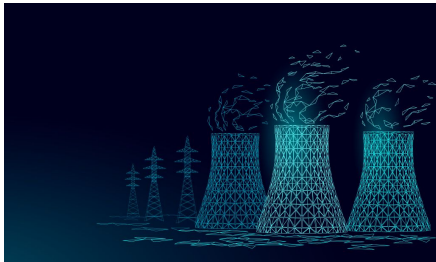
# To conclude



- A **sovereignty** tool to exploit French & EU **Exascale** supercomputers
- **Fill the value chain** of high performance computing and ensure **sustainability** of application developments

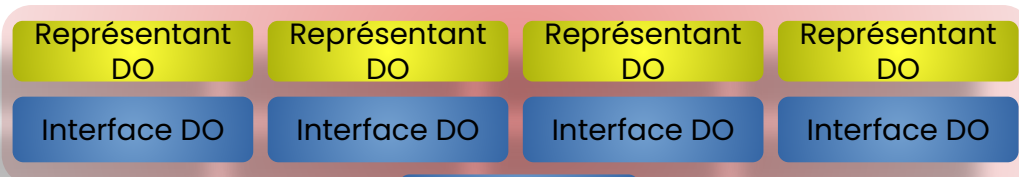


- A strong **dynamic** all over the **CEA**
- A **knock-on** effect with new **synergies** identified every weeks with code developers

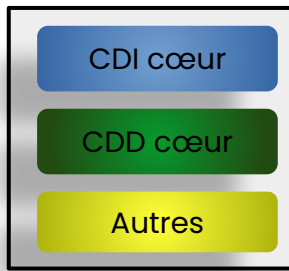


- A strong impact on the **programs of CEA** as well as on many **societal challenges**

# Notre organisation agile



Product Owner



Kokkos  
Comité C++  
Collaborations DoE  
...

Interface upstream  
Interface upstream

Référent tech.

**Équipe cœur CEXA**

Architecte

SCRUM Master

Animateur de sous-groupe  
Développeur  
Développeur  
Dev/Ops

Animateur de sous-groupe  
Développeur  
Développeur  
Dev/Ops

Animateur de sous-groupe  
Développeur  
Développeur  
Développeur

... InTheArt  
Coscinus

Animateur réseau

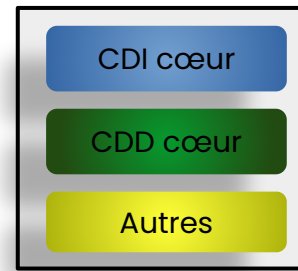
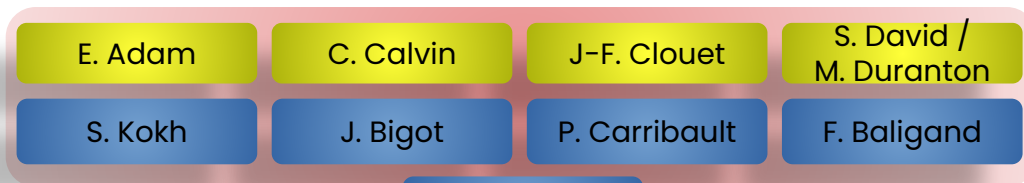
Réseau HPC@CEA

Porteur de démonstrateur  
Porteur de démonstrateur  
Porteur de démonstrateur

Porteurs de code



# L'équipe



J. Bigot

Kokkos  
Commité C++  
Collaborations DoE  
...

P. Carribault  
J. Jaeger

M. Perache

**Équipe cœur CEXA**

T. Padioleau

F. Baligand

A. Calloo

M. Lobet

C. Chevalier

R. Baron

T. Padioleau

F. Letierce

Développeur

Développeur

Développeur

Dev/Ops

Dev/Ops

Développeur

... InTheArt  
Coscinus

E. Audit

Réseau HPC@CEA

V. Grandgirard

F. Letierce

P. Ledac

Porteurs de code



# L'équipe cœur



**Julien Bigot**

*product owner*

DRF



**Ansar Calloo**

*Animateur groupe*

DES



**Mathieu Lobet**

*Animateur groupe*

DRF



**Cedric Chevalier**

*Animateur groupe*

DAM



# L'équipe cœur



**François Letierce**

développeur / porteur  
code

DAM



**Thomas Padioleau**

développeur /  
architecte

DRF



**Rémi Baron**

développeur

DES



**Yuuichi Asahi**

développeur

DRF

# L'équipe



**Marc Pérache**

*référént technique*

DAM



**Patrick Carribault**

*interface upstream*

DAM



**Julien Jaeger**

*interface upstream*

DAM



**Édouard Audit**

*animateur réseau*

DRF

# L'équipe



**Pierre Ledac**

*porteur de code*

DES



**Virginie  
Grandgirard**

*porteuse de code*

DRF



**Samuel Kokh**

*interface DO*

DES