

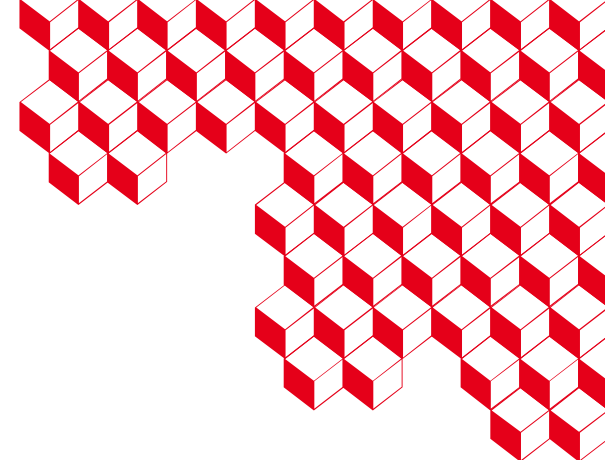


Rencontres Arcane

4^{ème} édition

Feuille de route 2025 – 2028

Avenant n°7 de la collaboration CEA/IFPEN



Sommaire détaillé

- Avenant n°7 2025 - 2028
 - Génie Logiciel
 - Faciliter les usages
 - Nouveaux usages
- Collaborations
 - IFPEN
 - CEA

Génie Logiciel

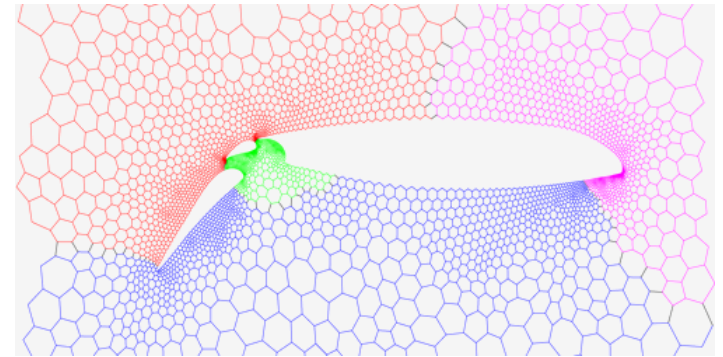
- **Évolutivité** et **maintenabilité** de la plateforme
 - Réorganisation/refonte d'anciennes parties de code
 - Support de nouvelles fonctionnalités du C++ 20/23
- **Intégration continue** et tests
 - Maintien et évolution des environnements de tests vers le HPC
 - multi-gpu NVIDIA AMD Intel
 - Tests de performance
 - Maintien de la couverture de tests (amélioration si possible...)



1 ■ Faciliter les usages

Faciliter les usages

- Simulation de **grande taille** (1/2)
- Faciliter la lecture des maillages
 - Partitionnement à la volée du maillage avec un partitionneur simple
 - Raffinement uniforme d'un maillage initial grossier
 - Intégration de nouveaux partitionneurs
 - KaHIP Université Karlsruhe ou COUPE thèse CEA
 - Partitionnement de plusieurs maillages
 - Optimisation des structures de données de maillages
 - connectivités compressées pour les maillages et les groupes
 - Support du format MSH dans le partitionneur externe parallèle



[KaHIP] Exemple de partitionnement avec KaHIP

[KaHIP] <https://kahip.github.io/>

[COUPE] <https://github.com/LIHPC-Computational-Geometry/coupe>

Faciliter les usages

- Simulation de **grande taille** 2/2
 - Besoin d'un format d'écriture similaire à VTKHDF[1]
 - Plusieurs directions possibles pour résoudre les difficultés d'export
 - Gestion de fusion de maillage pour permettre de fusionner les sous-domaines
 - Support direct des matériaux dans le post-traitement
 - Réaliser des post-traitements pendant le calcul
 - Réaliser des post-traitements à posteriori
 - Réduire le nombre de sorties intermédiaires du code de calcul
 - Éventuellement rejouer des unités de calcul du code

[1] https://docs.vtk.org/en/latest/design_documents/VTKFileFormats.html

Faciliter les usages

- Rendre la plate-forme plus modulaire
 - Accès de l'API GPU via Arccore (sans inclure tout Arcane)
 - Possibilité d'utiliser Arcane sous forme de bibliothèque pour gérer le maillage et les connectivités associées
 - Possibilité de ne pas compiler certaines composantes
 - Gestion des constituants (matériaux et milieux)
 - Implémentation des maillages (AMR, cartésiens et polyédriques)
- Réduire les dépendances obligatoires
 - Supprimer l'utilisation de la GLIB
- Définir précisément l'API publique disponible pour les utilisateurs

Faciliter les usages

- Optimisation et diffusion des outils HPC
 - Enrichir l'API GPU d'Arcane
 - Réaliser un back-end utilisant le standard SYCL 2020 permettant d'adresser tous types d'accélérateurs
 - Implémentation du parallélisme hiérarchique
 - Implémentation de structures de données tel que les tables de hachage
 - Implémentation d'un support des algorithmes avancées en mode multi-thread (Via TBB)
 - Branchement algèbre linéaire Alien sur GPU
 - Utilisation de l'API GPU d'Arcane
 - Branchement solveurs GPU
 - Indexation 64 bits d'Alien

Faciliter les usages

- Optimisation et diffusion des outils HPC
 - Lien avec NumPEx
 - Post doc permettant de comparer l'API GPU Arcane à Kokkos
 - Utilisation des lois ArcGeoSim dans l'application ShArc
 - Dépôt d'un AAP ou ShArc sera un démonstrateur
- Documentation et diffusion de la plateforme
 - Ajout d'exemples et didacticiels si besoin
 - Choix des langues dans la documentation
 - Animation d'une communauté Arcane ?
 - Ouverture d'un serveur Discord (en cours de construction...)
 - <https://discord.gg/tqdu7U2UwH>



2 ■ Nouveaux usages

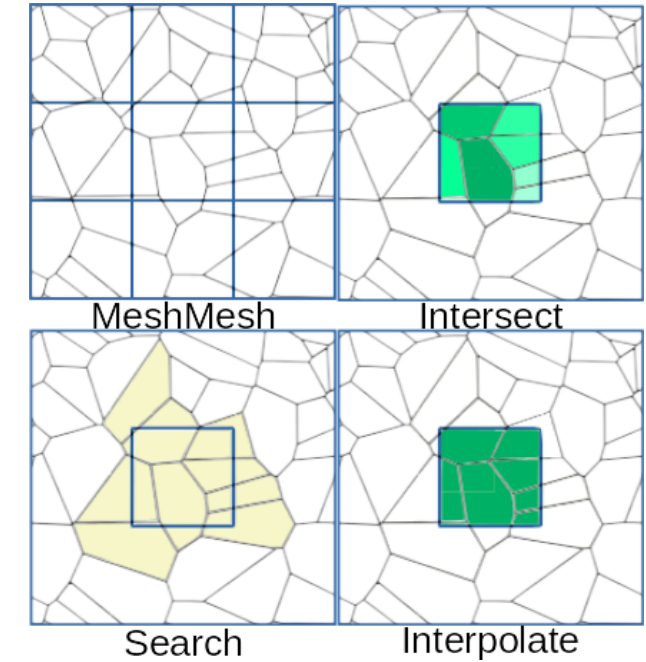
Nouveaux usages

- Nouvelles API : **IA**
 - Proposer une API pour utilisation de modèles IA et inférence
 - Pouvoir appeler depuis Arcane des bibliothèques IA en python
 - Ajouter nouveaux types `BFloat16`, `Float16` et `Float32`
 - variables et échanges de message post-traitement, protection/reprises
- Nouvelles API : **Éléments Finis**
 - Définir et implémenter une API pour les éléments finis
 - Supporter les éléments de maillage d'ordre élevé
 - Intégration d'Alien dans cette API



Nouveaux usages

- Nouvelles API : **Couplage et projections**
 - Proposer l'écriture d'une API de couplage de codes au sein d'Arcane
 - Proposer une première implémentation en utilisant l'outil preCICE [2]
 - Développements d'algorithmes de projection spécifiques si besoin



[1] Projection d'un maillage source structuré vers maillage cible non structuré

 preCICE



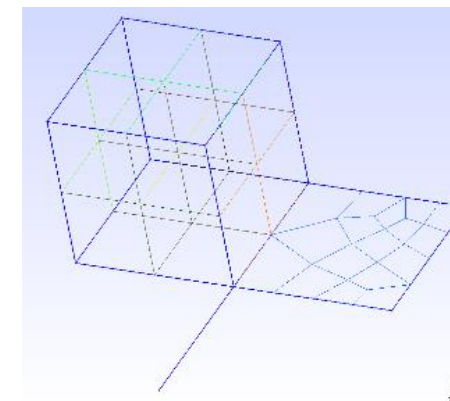
3 ■ Collaborations

Collaborations IFPEN

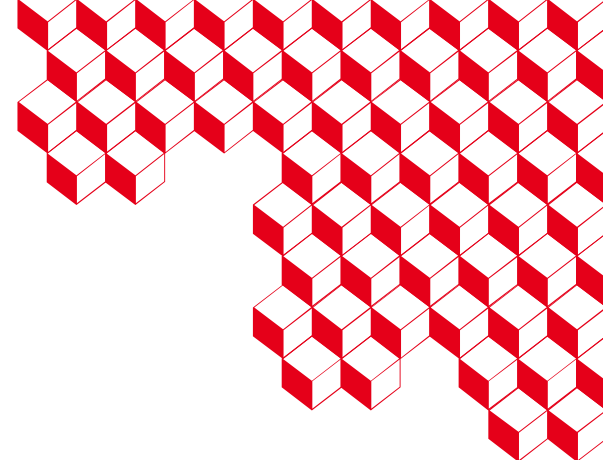
- Contrat progrès avec le **CINES** pour le portage GPU d'une application Géoscience (ArcaDES)
- Collaborations à venir via **NumPEX**
 - Post-doctorat avec la **Maison de la Simulation** lois et GPU
 - Montage d'un projet ANR : **CEA** (MdIS, DES), **CECI** (CNRS/CERFACS), **ONERA**, **INRIA**
- **Thèse** avec l'**INRIA** inférence pour le calcul HPC
 - Conceptualisation et design d'un framework d'inférence de DNNs dédié à la simulation massivement parallèle sur des architectures Exascale

Collaborations CEA

- Collaboration avec le CEA/DES
 - Gestion des éléments finis : **ArcaneFem**
 - Support des maillages non-manifold pour le code **Manta [1]**
 - Gestion des mailles de dimensions différentes dans un même maillage
 - Support des informations de périodicité
 - Participation à deux programmes transverses de compétence
 - PTC-RAMI : raffinement de maillage lors de l'initialisation
 - PTC-EXAQUAKE: simulation de séisme



[1] Olivier Jamond, Nicolas Lelong, Axel Fourmont, Joffrey Bluthé, Matthieu Breuze, et al.. MANTA : un code HPC généraliste pour la simulation de problèmes complexes en mécanique. CSMA 2022 15ème Colloque National en Calcul des Structures, May 2022, Giens, France.
<https://cea.hal.science/hal-03688160/>



**Merci de
votre
attention**

<https://github.com/arcaneframework/framework>
<https://discord.gg/tqdu7U2UwH>