

salome_meca : plateforme de simulation de mécanique pour les études et la recherche

M. Abbas¹, G. Drouet¹, G. Ferté¹, N. Pignet¹

¹ EDF R&D, Département Electrotechnique et Mécanique des Structures, boulevard Gaspard Monge, 91120 Palaiseau, {mickael.abbas, guillaume.drouet,, guilhem.ferte, nicolas.pignet}@edf.fr

Résumé — salome_meca est une plate-forme dédiée à la simulation en mécanique. Son ambition est de fournir un environnement logiciel permettant de réaliser la chaîne complète de calcul CAO, maillage, calcul, post-traitement et visualisation des résultats.

Mots clefs — code_aster, salome, salome_meca, simulation numérique, open source.

1. Le solveur code_aster



é des installations mécaniques et de génie civil d'EDF nécessitent tation, de réparation ou de remplacement par des modélisations non linéaires en mécanique. Ainsi, au-delà des fonctionnalités standards d'un logiciel généraliste de simulation aux éléments finis en mécanique, code_aster capitalise les modèles numériques issus des travaux de recherche spécifiques à ces enjeux, assurant ainsi la maîtrise de leur mise en œuvre et surtout leur transfert rapide aux études d'ingénierie.

L'ambition de code_aster est donc double :

- mettre un logiciel de simulation puissant, robuste et stable à disposition des études d'expertise dans un contexte de développement et de diffusion sous assurance de la qualité ;
- accueillir et capitaliser les modèles en mécanique numérique produits par EDF R&D.

En lien avec les deux objectifs cités, développer son propre code garantit la capitalisation de la R&D et son transfert rapide vers l'ingénierie, difficile à obtenir avec un code commercial. Cette R&D est spécifique car EDF est exploitante, et non manufacturière de ses matériels, en particulier nucléaires, dont elle doit justifier la durée de vie, économiquement et réglementairement.

Ainsi, code_aster intègre des modèles numériques spécifiques :

- pour simuler le vieillissement des matériaux et des structures : fatigue, endommagement, mécanique de la rupture, milieux poreux, etc. ;
- pour les calculs propres aux problématiques de l'exploitation nucléaire : interaction sol-structure et séisme, calculs réglementaires, modélisation des assemblages combustibles, etc.

Depuis 2001, code_aster est diffusé comme *logiciel libre* (GNU general public licence) avec les objectifs suivants : améliorer la qualité et le référencement du code par la démultiplication de son usage ; être un vecteur de nos partenariats dans le domaine de la simulation. Le logiciel et son code source, mais aussi les cas tests et toute la documentation, sont librement accessibles à tous [1]. Après 30 années, code_aster est un succès mondial de diffusion et fédère un réseau d'utilisateurs par le

forum du site web.

A partir de la version 15.4, sortie en juillet 2021, code_aster accueille deux évolutions majeures d'architecture en plus des fonctionnalités supplémentaires (nouveaux modèles, nouveaux éléments finis).

L'interface d'accès principal utilise un langage de commande en Python depuis 2001. Désormais, code_aster devient un module Python classique, permettant une plus grande interaction entre le solveur et l'utilisateur. Les objets bas-niveau (Fortran) sont accessibles de manière sûre dans l'espace Python, sans recopie mémoire. De nombreuses fonctions d'accès sont écrites en Python, ce qui permet un maquettage plus fin.

```
import code_aster
import numpy as np

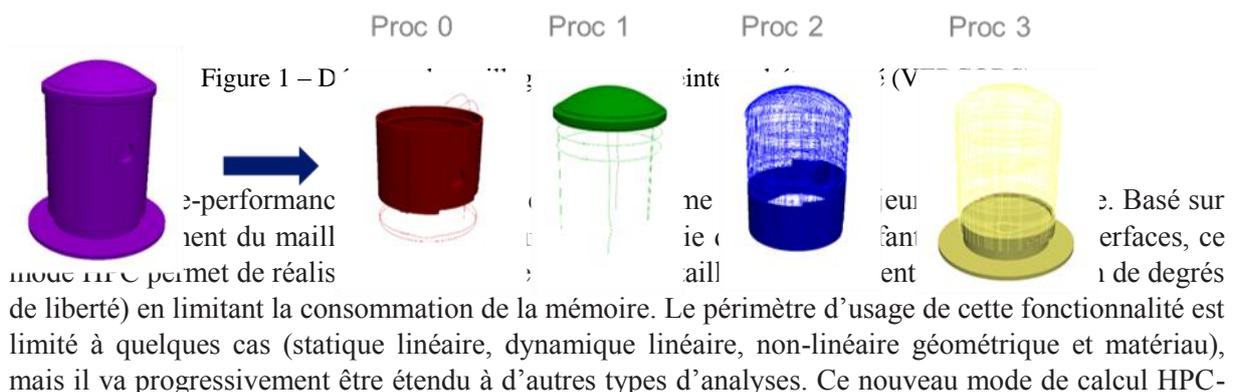
code_aster.init("--test")
test=code_aster.TestCase()

# Creation of the mesh
mesh = code_aster.Mesh()
mesh.readMedFile("zzzz255a.mmed")

# Creation of the model
model = code_aster.Model(mesh)
model.addModelingOnGroupOfCells(code_aster.Physics.Mechanics,
                                code_aster.Modelings.Tridimensional,"ALL")

model.build()
```

Ce nouveau mode d'interaction avec le code complète le mode « historique », en effet les anciens jeux de commande sont toujours utilisables sans modifications. Il permet naturellement d'importer des modules Python de toute nature, y compris numpy (de nombreux opérateurs d'accès aux objets de code_aster peuvent exporter dans ce format). Ces fonctions d'accès sont également disponibles dans la partie C++ du code qui a été ajoutée pour protéger l'accès aux données et permettre le wrapping du cœur Fortran avec le Python.



massif complète les possibilités HPC déjà disponibles dans code_aster.

2. Salomé

EDF R&D développe depuis plusieurs années des codes de calculs dans le domaine de la simulation (code_aster, Code_Saturne, Syrthes, etc.). Ces codes de calculs sont très souvent stratégiques, porteurs d'une très forte innovation technologique par les phénomènes physiques qu'ils permettent de simuler et sont l'objet d'investissements lourds.

Salome propose un environnement utilisateur interactif (Pré/Post Processeur) unifié, indispensable à l'utilisation de ces codes dans l'ingénierie, en facilitant la phase de construction du jeu de données et limitant le coût d'entrée de nouveaux utilisateurs. De plus, grâce à son architecture modulaire, la plateforme Salome facilite l'interopérabilité entre la CAO et les codes de calculs et l'implémentation des couplages entre les codes de calculs dans un environnement hétérogène distribué. Ceci permet de répondre aux exigences de couplages multi-physiques.

La plate-forme Salome, développée dans le cadre d'un consortium, est diffusée sous une licence OpenSource LGPL [2]. Le choix du modèle OpenSource permet aux développeurs de codes de calculs de se concentrer sur leur cœur de métier (la physique et les méthodes numériques) et d'optimiser, par le système de la mutualisation, l'investissement requis pour disposer d'un environnement interactif efficace.

3 La plateforme salome_meca



me plate-forme dédiée à la simulation en mécanique construite sur base Salome. fournir un environnement logiciel permettant de réaliser la chaîne complète de calcul en mécanique, post-traitement et visualisation des résultats.

La plate-forme salome_meca 2021 (diffusée en juillet 2021) permettra de réunir dans un environnement intégré et unique :

1. modules de **CAO** et **maillage** de salome ;
2. **code_aster**, solveur de mécanique implicite ;
3. **AsterStudy**, l'interface graphique nouvelle génération de code_aster ;
4. **Open TURNS**, logiciel permettant de traiter les incertitudes et de réaliser des études de sensibilité des paramètres du calcul [4] ;
5. **ADAO**, bibliothèque d'algorithmes mathématiques d'assimilation de données permettant d'établir des scénarios de recalage de paramètre avec les solveurs de la plate-forme ;
6. **Paravis**, module de visualisation basé sur le moteur graphique parallèle de Paraview [5] ;

Références

[1] : Site officiel de code_aster, www.code-aster.org

[2] : Site officiel de Salome, www.salome-platform.org

[4] : Site officiel d'Open TURNS, trac.openturns.org

[5] : Site officiel de Paraview, www.paraview.org