



Club U MFront

Retour d'expérience de l'intégration de MFront dans Code_Aster

CEA Cadarache
6 février 2015

Jean-Michel Proix,
Nicolas Sellenet,
Mathieu Courtois,
François Hamon
Thomas De Soza,
Josselin Delmas

EDF-R&D



Plan

- **Contexte à EDF**
- **2013 – 2015 : l’histoire d’une intégration**
 - Janvier 2013 : la rencontre
 - Septembre 2013 : l’évaluation
 - Novembre 2013 – juin 2014 : en avant vers l’Open-Source
 - Juillet - septembre 2014 : du prototype à l’intégration transparente dans Code_Aster
 - Octobre 2014 : la mise en Open-Source
 - Décembre 2014 : version 12.3 intégrant MFront
- **Perspectives et besoins**

Contexte à EDF

Plusieurs projets engagés dans MFront

- **Une dizaine de projets impliqués dans des domaines variés**
 - « Intégrateurs codes » : PSM, 3M
 - « Transverse » : LoCo2
 - « Métiers » : ADELAHYD (barrages), PERFORMII (micro-méca), OMARISI-2016 (GC séisme), CIWAP2 (GC), CORIOLIS (corrosion sous contraintes), Crayon combustible

- **Différents codes ou outils**
 - Code_Aster
 - Cyrano3
 - CADEEX (Mtest)

Contexte à EDF

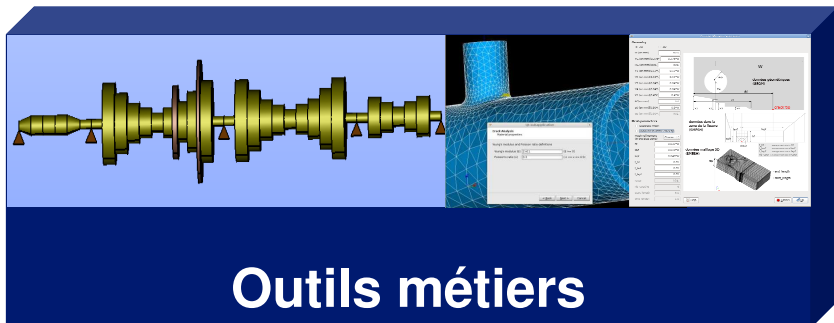
Code_Aster et Salome-Meca



- CAO
- Maillage
- Composition de schémas de calcul
- Post-traitements graphiques et mathématiques



- *Code_Aster* : mécanique généraliste implicite
- Europlexus : dynamique rapide



- Mécanique de la rupture
- Machines tournantes
- Conduites forcées
- Reconstruction de champs de température

Janvier 2013

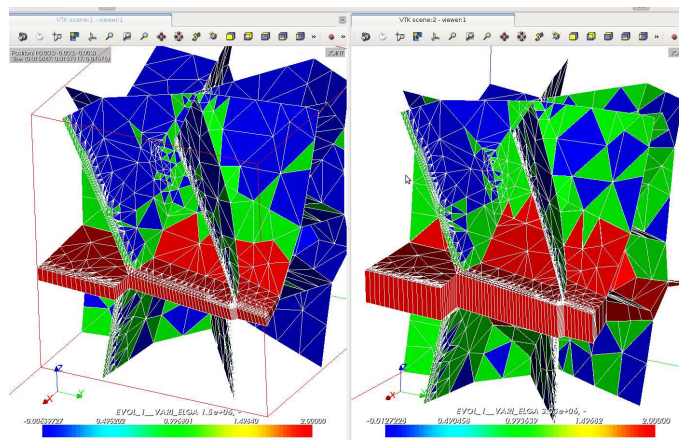
La rencontre

- **Rencontre des équipes PLEIADES CEA et Code_Aster EDF sur la thématique de l'intégration des lois de comportement**
 - Découverte de MFront après un premier contact avec T. Helfer
 - Partage d'expérience sur l'intégration numérique des lois de comportement

Septembre 2013

L'évaluation

- Prise en main de MFront mis à disposition gracieusement par le CEA
- Implémentation dans Code_Aster d'une interface pour MFront
- Écriture d'une note technique en commun
 - Collaboration bénéfique pour les deux parties (amélioration de MFront, benchmark pour Code_Aster)
 - Mise en évidence de la robustesse et des performances de MFront (jusqu'à deux fois plus rapide que l'implémentation native Aster sur certains cas)
 - Facilite la mise au point de l'intégration (écriture facilitée de la matrice jacobienne analytique pour les lois complexes → gain en performances)



Fissuration complète dans un calcul de corrosion sous contraintes; 10h45 CPU avec jacobienne analytique vs. 43h CPU avec jacobienne numérique (implémentation Aster originale)

Novembre 2013 – juin 2014

En avant vers l'Open-Source

■ Intérêts pour EDF à disposer de MFront dans un cadre Open-Source

□ Productivité

- Gains de temps pour l'écriture de comportements, plus besoin de compétences informatiques spécifiques (environnement Aster, Fortran)
- Performances de l'implémentation MFront par comparaison à des lois existantes dans Aster

□ Préparation de l'avenir

- Grand nombre de méthodes d'intégration disponibles (par exemple Broyden, Runge-Kutta 5/4)
- Capacité à devenir co-développeur de MFront pour l'adapter à nos besoins spécifiques (besoin non couvert par les bibliothèques fermées type UMAT ou ZMAT)

□ Facilité d'accès aux lois de comportement (ouverture plus large que dans un code)

- Plus besoin d'avoir recours à *Code_Aster* pour des applications « point matériel »

□ Standardisation des lois de comportement pour mutualiser leur utilisation à EDF

- Substitution de lois présentes « en dur » dans *Code_Aster*
- Exemple d'utilisations possibles : SALOME/ADAO pour identification de paramètres, projets « matériaux » à EDF

■ Discussion d'une mise en libre entre les programmes CEA et EDF

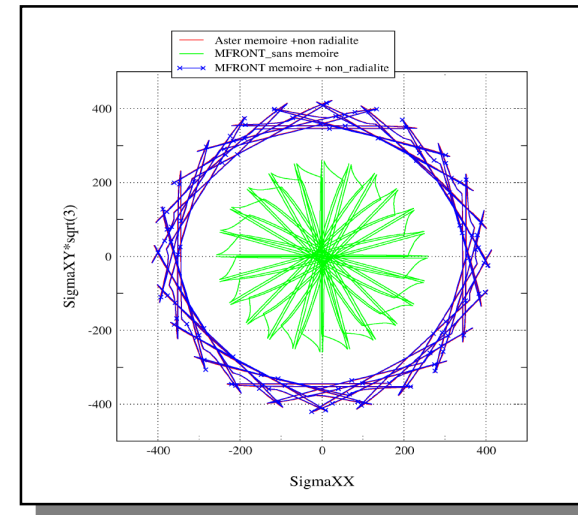
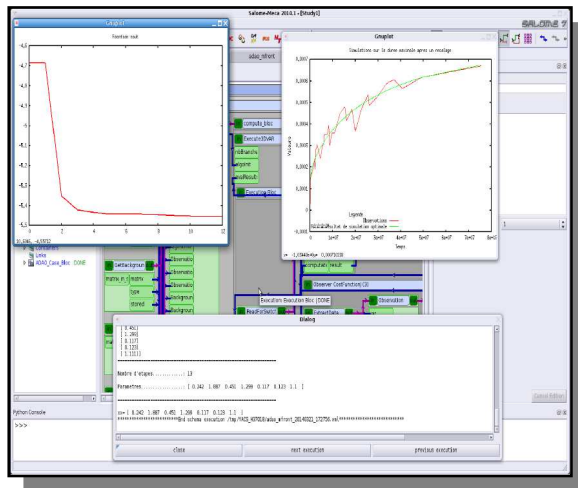
Novembre 2013 – juin 2014

En avant vers l'Open-Source (2)

- Quelques exemples des performances obtenues tirés de « *Écriture de lois de comportement avec mfront : tutoriel* », T. Helfer, J.M. Proix

TABLE 1 – comparaison des temps de calcul

éprouvette entaillée 2D axi	MFRONT impl J prog	MFRONT impl J num	MFRONT expl RK4/5	Aster impl J prog	Aster impl J num	Aster expl RK2/1
Nb de pas de temps	601	601	4011	601	601	4011
Nb itérations Newton	1818	1810	17493	2479	1832	17495
Temps CPU	3mn5s	3mn19s	32mn21s	8mn40s	8mn53s	46mn37s



Juillet 2014 – septembre 2014

Du prototype à l'intégration transparente

- **Développement d'une interface dédiée pour Aster dans MFront**
 - Modifications mineures de l'interface déjà mise à disposition par T. Helfer
 - Suppression de beaucoup de paramètres (par rapport au standard UMAT)
 - Passage du type de modélisation à MFront (3D, Axisymmetrical, ...)
 - Interrogation de la loi de comportement pour connaître les *ExternalStateVariables*
 - Récupération côté Aster des *InternalStateVariables*
- **Premier étape côté Aster : prototypage**
 - Syntaxe côté Aster : définition d'un matériau « UMAT », Appel de MFront dans un opérateur de calcul

```
MAT = DEFI_MATERIAU( # DruckPragEcroLin.mfront
                    UMAT = _F( C1 = 5800.0E6,
                               C2 = 0.3,
                               C3 = 0.328,
                               C4 = 0,
                               C5 = 2.11E6,
                               C6 = 0,
                               NB_VALE=6, ), )
```

```
U1 = STAT_NON_LINE( # (...)
                    COMPOTEMENT=_F( RELATION = 'MFRONT',
                                       NOM_ROUTINE='asterdruckpragecrolin',
                                       LIBRAIRIE='druckprag.so', ) # (...)
                    ),
```

Juillet 2014 – septembre 2014

Du prototype à l'intégration transparente (2)

- **Deuxième étape : ajout de comportements MFront de manière transparente**
 - Intégrés dans le source de Code_Aster
 - Compilés à l'installation de Code_Aster
 - Ces comportements peuvent remplacer des comportements présents dans Code_Aster
- **Syntaxe beaucoup plus lisible pour l'utilisateur**

```
MATF=DEFI_MATERIAU (
    Chaboche=_F(YoungModulus = 145200,
                PoissonRatio = 0.3,
                R_inf = 151.,
                R_0 = 87.,
                b = 2.3,
                k = 0.43,
                w = 6.09,
                C_inf_0 = 187.*341.,
                C_inf_1 = 29.*17184.,
                g_0_0 = 341,
                g_0_1 = 17184,
                a_inf = 1.,),)

    SOLNL=SIMU_POINT_MAT (
        MATER=MATF,
        COMPORTEMENT=_F(RELATION='Chaboche',
                        RESI_INTE_RELA=1e-12,),
        # (...) )
```

Octobre 2014

La mise en Open-Source

- **Actée en juillet, effective en octobre**
- **Licence libre**
 - A permis d'éviter la remise en cause de la politique de diffusion libre de *Code_Aster*
- **Accès au dépôt SVN**
 - A facilité la restitution des développements EDF dans l'interface Aster de MFront

Décembre 2014

Version 12.3 de Code_Aster intégrant MFfront

- Merci à Thomas Helfer, Renaud Masson, Bruno Michel et Vincent Marelle pour avoir rendu cela possible



Code_Aster

Accueil du site Présentation Documentation Formations Forum Téléchargement Support ProNet

Français ▼

Téléchargement

- Code_Aster
- Mise à jour de Code_Aster
- Salome-Meca
- Pré-requis
- Outils
- Licences
- FAQ - Installation

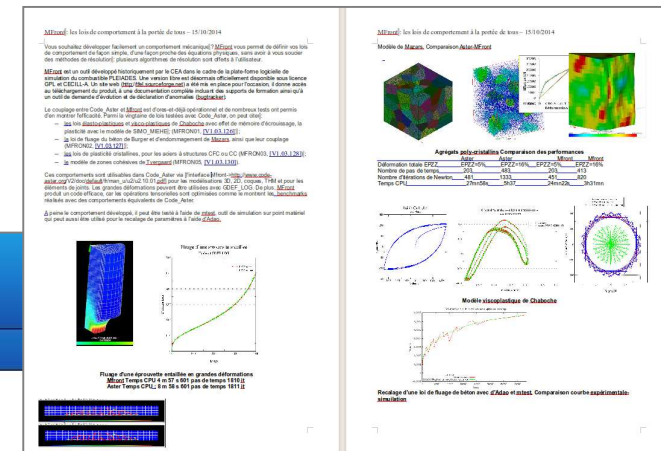
Téléchargement

Derniers téléchargements

Les paquets `aster-full` (`Code_Aster_stable` et `Code_Aster_testing`) contiennent tous les pré-requis nécessaires sauf Python qui doit être installé avant `Code_Aster`.

Pour l'installation des mises à jour pour `Code_Aster_unstable`, consultez la page [Version unstable](#) de la section Support.

Vous devez consulter et accepter les termes de la licence avant de télécharger un produit.



Perspectives concernant MFront dans Code_Aster

- **Conversion de lois de Code_Aster intégrées par un algorithme vers MFront**
 - Facilitation de la maintenance
 - Amélioration de la robustesse (plus d'algorithmes de résolution)
 - Simplification de certains modèles
- **Utilisation de MFront pour les lois élastiques non-linéaires (hyper-élasticité)**

Besoins et intérêts pour de nouvelles fonctionnalités

- **Intégration de la non-localité dans MFront**

- Pour les lois d'endommagement
- Gradient de variables internes ou de déformation en entrée

- **Lois écrites en efforts généralisés (lois globales pour le GC sous séisme)**

- Permettrait à plus long-terme l'intégration dans Europlexus pour le chaînage avec Code_Aster (même loi, même implémentation)

Besoins pour la qualification et la pérennisation de l'intégration

- **AQ**

- Politique de versionnement (version stable, version de dév.)
- Release notes, fiche qualité (si comportements directement utilisés)

- **Cadre contractuel ou partenarial**

- Pour restitution dans le dépôt officiel
- Pour définir les orientations scientifiques et techniques en matière de développement