
Notice de recette de Salome-Meca

Résumé :

Cette documentation décrit la méthode pour effectuer la recette d'une installation de Salome-Meca sur une station locale. Il peut s'agir de la version Calibre diffusée en interne EDF ou de la version LGPL diffusée en externe. On y décrit plus précisément le lancement des tests fournis avec les versions de *Code_Aster* embarquées et des tests des différents outils de la plate-forme Salome-Meca.

Ce document a pour objectif :

- de formaliser une démarche de recette de la plate-forme Salome-Meca,
- de permettre à un Bureau d'Études de vérifier selon les mêmes critères qu'EDF la bonne installation d'une version de Salome-Meca,
- de permettre au donneur d'ordre EDF de contrôler le bon fonctionnement d'une version installée localement chez un prestataire si elle a été utilisée pour les calculs finaux.

1 Lancement des tests associés au solveur Code_Aster

Il est nécessaire d'avoir procédé à l'installation de la plate-forme Salome-Meca en utilisant les instructions disponibles sur le site www.code-aster.org après avoir téléchargé la version ad-hoc.

La procédure de recette décrite ci-dessous a été validée pour les versions de Salome-Meca à partir de la version **2016**. Il s'agit de lancer les tests embarqués dans la(les) version(s) du solveur *Code_Aster* embarqué(es). Concernant la version 2016, on dispose des versions 12.6.0 (*stable*) et 13.2.0 (*testing*). À titre d'information, l'exécution des 3362 tests associés à la version *stable* demande environ 1H45 sur une station de type HP P700 (32 Go de RAM, 24 cœurs, Intel(R) Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz) sous système d'exploitation 64 bits Calibre 9 (variante interne EDF de la Debian 8 Jessie). pour les différents logiciels embarqués spécifiquement pour la plate-forme

Salome intègre un script permettant de lancer des tests. Ce script peut-être utilisé pour lancer en automatique l'ensemble de la base de tests livrée et d'effectuer un bilan qu'il est facile de comparer à celui figurant dans la fiche qualité de la version d'exploitation (fascicule A0 du manuel A de la documentation *Code_Aster*). D'éventuels écarts par rapport à la liste des tests mentionnés comme s'arrêtant en erreur doivent être signalés à l'équipe projet R&D afin qu'elle se prononce.

La commande qui permet de lancer la procédure de recette est par exemple :

```
salome test -I ASTER_12 -j 8
```

ou bien

```
salome test -I ASTER_13 -j 12
```

Le nombre d'occurrence (8 ou 12) doit être adapté à la plate-forme utilisée et au nombre de cœurs disponibles.

En fin d'exécution le bilan est affiché dans le terminal :

```
...
3360/3362 Test #7687: ASTER_12.6.0_zzzz367a ..... Passed 17.67 sec
3361/3362 Test #7681: ASTER_12.6.0_zzzz364a ..... Passed 68.67 sec
3362/3362 Test #7623: ASTER_12.6.0_zzzz337b ..... Passed 146.09 sec

99% tests passed, 42 tests failed out of 3362

Label Time Summary:
ASTER_12.6.0                = 48657.85 sec
ASTER_12.6.0,SMECA_INTEGR   = 5.82 sec

Total Test time (real) = 6301.69 sec

The following tests FAILED:
 4465 - ASTER_12.6.0_erreu06a (Failed)
 4783 - ASTER_12.6.0_plexu01a (Failed)
 4784 - ASTER_12.6.0_plexu02a (Failed)
 4785 - ASTER_12.6.0_plexu02b (Failed)
 4786 - ASTER_12.6.0_plexu03a (Failed)
 4787 - ASTER_12.6.0_plexu03b (Failed)
 4788 - ASTER_12.6.0_plexu03c (Failed)
 4789 - ASTER_12.6.0_plexu04a (Failed)
 4790 - ASTER_12.6.0_plexu05a (Failed)
 4791 - ASTER_12.6.0_plexu06a (Failed)
 4792 - ASTER_12.6.0_plexu07a (Failed)
 4793 - ASTER_12.6.0_plexu08a (Failed)
 4794 - ASTER_12.6.0_plexu08b (Failed)
 4795 - ASTER_12.6.0_plexu08c (Failed)
 4796 - ASTER_12.6.0_plexu08d (Failed)
 4797 - ASTER_12.6.0_plexu08e (Failed)
 4798 - ASTER_12.6.0_plexu09a (Failed)
 4799 - ASTER_12.6.0_plexu09b (Failed)
 4800 - ASTER_12.6.0_plexu10a (Failed)
 4801 - ASTER_12.6.0_plexu10b (Failed)
 4803 - ASTER_12.6.0_plexu10d (Failed)
```

```
4804 - ASTER_12.6.0_plexu10e (Failed)
4805 - ASTER_12.6.0_plexu10f (Failed)
6916 - ASTER_12.6.0_supv002a (Failed)
7469 - ASTER_12.6.0_zzzz218a (Failed)
7470 - ASTER_12.6.0_zzzz218b (Failed)
7471 - ASTER_12.6.0_zzzz218c (Failed)
7652 - ASTER_12.6.0_zzzz354a (Failed)
7653 - ASTER_12.6.0_zzzz354b (Failed)
7654 - ASTER_12.6.0_zzzz354c (Failed)
7655 - ASTER_12.6.0_zzzz354d (Failed)
7656 - ASTER_12.6.0_zzzz354e (Failed)
7657 - ASTER_12.6.0_zzzz354f (Failed)
7658 - ASTER_12.6.0_zzzz354g (Failed)
7659 - ASTER_12.6.0_zzzz355a (Failed)
7660 - ASTER_12.6.0_zzzz355b (Failed)
7661 - ASTER_12.6.0_zzzz355c (Failed)
7662 - ASTER_12.6.0_zzzz355d (Failed)
7663 - ASTER_12.6.0_zzzz355e (Failed)
7664 - ASTER_12.6.0_zzzz355f (Failed)
7665 - ASTER_12.6.0_zzzz355g (Failed)
7666 - ASTER_12.6.0_zzzz355h (Failed)
Errors while running CTest
```

Il faut ensuite comparer le résultat du passage des tests avec la fiche qualité de la version associée (cf. par exemple [A0.02.80] « Fiche Qualité de la version d'exploitation de *Code_Aster* : version 12 »). Avec une installation Salome-Meca standard sous Calibre9, les fichiers associés à l'exécution des cas tests sont conservés sous le répertoire `/local100/tmp/nni_resutestYY.Y.Y` où *nni* est le compte utilisateur sur la station Calibre9 et *YY.Y.Y* est la version de *Code_Aster* considérée.

Il est possible de relancer un unique test en utilisant la commande suivante :

```
salome test -R "ASTER_YY.Y.Y_XXXX"
```

avec *YY.Y.Y* la version de *Code_Aster* considérée et *XXXX* le nom du test, par exemple :

```
salome test -R "ASTER_13.2.0_erreu06a"
```

2 Lancement des tests associés à la plate-forme Salome-Meca

Il est possible de lancer les tests des différents outils de la plate-forme Salome-Meca en utilisant la commande suivante :

```
./salome test -L SMECA
```

Cette commande lancera les tests fournis pour les différents logiciels embarqués spécifiquement pour la plate-forme, dont une sélection limitée des tests des versions du solveur code_aster. Elle produit par exemple pour la version 2017 la sortie suivante :

```
Test project /home/I27518/salome_meca/appli_V2017/bin/salome/test
  Start 829: EDYOS_edynos.tests.unittest_entity_form.ComplexEntityFormTC
1/197 Test #829: EDYOS_edynos.tests.unittest_entity_form.ComplexEntityFormTC .....
Passed    6.35 sec
...
  Start 918: EDYOS_edynos.tests.unittest_validation.PapaTehdCalculationTC
90/197 Test #918: EDYOS_edynos.tests.unittest_validation.PapaTehdCalculationTC .....
Passed    2.96 sec
  Start 919: MT_MT_sdll124a
91/197 Test #919: MT_MT_sdll124a .....
Passed   11.06 sec
...
114/197 Test #942: MT_MT_sdn1Ba
.....***Failed 1163.64 sec
...
  Start 961: MT_MT_sdll126f
133/197 Test #961: MT_MT_sdll126f .....
Passed    8.43 sec
  Start 962: OMA_coude
134/197 Test #962: OMA_coude .....
Passed   15.17 sec
  Start 963: OMA_tube
135/197 Test #963: OMA_tube .....
Passed   22.51 sec
  Start 964: ORT_ort.data_struct.TestStudyData
136/197 Test #964: ORT_ort.data_struct.TestStudyData .....
Passed    0.47 sec
...
  Start 971: ORT_ort.engine.aster.TestAsterStudy
143/197 Test #971: ORT_ort.engine.aster.TestAsterStudy .....
Passed   11.92 sec
  Start 972: DHRC_dhrc.studydata.TestStudyData.test01_basic
144/197 Test #972: DHRC_dhrc.studydata.TestStudyData.test01_basic .....
Passed    0.10 sec
...
  Start 981: DHRC_dhrc.engine.post.TestParametersIdentification.test01_basic
153/197 Test #981: DHRC_dhrc.engine.post.TestParametersIdentification.test01_basic .....
Passed   20.66 sec
  Start 982: SMECA_UTILS_test01
154/197 Test #982: SMECA_UTILS_test01 .....
Passed   16.37 sec
  Start 983: AC_acseisme.engine.functions.TestFunctions.test01a
155/197 Test #983: AC_acseisme.engine.functions.TestFunctions.test01a .....
Passed    6.06 sec
...
  Start 993: AC_acseisme.engine.classes.TestClasses.test_rangee
165/197 Test #993: AC_acseisme.engine.classes.TestClasses.test_rangee .....
Passed    6.26 sec
  Start 994: CABRI
166/197 Test #994: CABRI .....
Passed   16.45 sec
  Start 995: CT_test0001
167/197 Test #995: CT_test0001 .....
Passed   10.70 sec
...
  Start 998: CT_test0004
170/197 Test #998: CT_test0004 .....
Passed    9.79 sec
  Start 999: CT_test0005
```

```
171/197 Test #999: CT_test0005 .....
Passed 8.75 sec
Start 1000: CT_test0006
172/197 Test #1000: CT_test0006 .....
Passed 11.97 sec
Start 1001: CT_test0007
173/197 Test #1001: CT_test0007 .....
Passed 9.39 sec
Start 1002: CT_perfe02a
174/197 Test #1002: CT_perfe02a .....
***Failed 10.33 sec
Start 1003: CT_ssnv108a
175/197 Test #1003: CT_ssnv108a .....
***Failed 9.61 sec
Start 1004: CF_Test01
176/197 Test #1004: CF_Test01 .....
Passed 27.05 sec
Start 1005: CF_Test02
177/197 Test #1005: CF_Test02 .....
Passed 18.29 sec
Start 1006: MAC3_MONO
178/197 Test #1006: MAC3_MONO .....
Passed 10.20 sec

...
Start 1010: MAC3_N4
182/197 Test #1010: MAC3_N4 .....
Passed 191.35 sec
Start 1210: ASTER_12.8.0_forma01c
183/197 Test #1210: ASTER_12.8.0_forma01c .....
Passed 6.73 sec

...
Start 4154: ASTER_12.8.0_zzzz218a
189/197 Test #4154: ASTER_12.8.0_zzzz218a .....
Passed 23.68 sec
Start 4559: ASTER_13.4.0_forma01c
190/197 Test #4559: ASTER_13.4.0_forma01c .....
Passed 3.89 sec

...
Start 7925: ASTER_13.4.0_zzzz401a
197/197 Test #7925: ASTER_13.4.0_zzzz401a .....
Passed 2.74 sec
```

98% tests passed, 4 tests failed out of 197

Label Time Summary:

ASTER_12.8.0,SMECA_INTEGR	= 68.13 sec
ASTER_13.4.0,SMECA_INTEGR	= 76.84 sec
SMECA	= 16.37 sec
SMECA_AC	= 132.12 sec
SMECA_CABRI	= 16.45 sec
SMECA_CF	= 45.33 sec
SMECA_CT	= 89.29 sec
SMECA_DHRC,SMECA_INTEGR	= 21.77 sec
SMECA_EDYOS	= 297.86 sec
SMECA_INTEGR	= 611.24 sec
SMECA_MAC3	= 593.49 sec
SMECA_MT	= 4687.32 sec
SMECA_MT,SMECA_INTEGR	= 1208.90 sec
SMECA_OMA	= 37.69 sec
SMECA_ORT	= 13.89 sec
SMECA_UTILS	= 16.37 sec

Total Test time (real) = 7383.53 sec

The following tests FAILED:

- 942 - MT_MT_sdn1Ba (Failed)
- 943 - MT_MT_sdn1Bb (Failed)
- 1002 - CT_perfe02a (Failed)
- 1003 - CT_ssnv108a (Failed)

Errors while running CTest

3 Cas-tests de validation graphique de la plate-forme Salome-Meca

Il n'existe pour l'heure pas de procédure de lancement automatique des cas-tests de validation graphique de Salome-Meca. La procédure consiste à tester manuellement l'installation en suivant les instructions des manuels contenus dans les fascicules SV1, SV2 et SV3 de la documentation Salome-Meca.