Opérateur EXEC_LOGICIEL

1 But

Le but de cet opérateur est d'exécuter un logiciel externe depuis code_aster sur la machine locale. Cela peut être n'importe quel programme ou script exécutable.
Une première variante est d'exécuter un mailleur au lieu d'un programme quelconque afin de produire un fichier de maillage. Dans ce cas, on démarre SALOME ou GMSH en lui passant un fichier de données en argument et on récupère ensuite un fichier de maillage.
La deuxième variante consiste à faire appel à une session SALOME déjà ouverte sur la machine locale ou une machine distante et à y exécuter un script (de visualisation par exemple).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Syntaxe

```
EXEC_LOGICIEL (
# ou si MAILLAGE est présent
mail = EXEC_LOGICIEL (                                          [maillage]

    ◊  |  LOGICIEL  =  nom_exe,                                 [TX]
          ◊  ARGUMENT  =  (arg1, arg2,… )                       [l_TX]
          ◊  SHELL     =  /  'NON',                             [DEFAUT]
                          /  'OUI',

       |  /  MAILLAGE  =  _F (
             ♦  FORMAT    =  /  'GMSH',                        [TX]
                             /  'SALOME'
             ◊  UNITE_GEOM   =  /  igeom,                       [I]
                                =  /  16                        [DEFAUT]
             ),

       |  /  SALOME    =  _F (
             ♦  CHEMIN_SCRIPT   =  script,                      [TX]

             ◊   MACHINE        =   machine ,                       [TX]
                 # si MACHINE est renseigné
                 ♦ UTILISATEUR  =  utilisateur,                 [TX]
             ◊  PORT            =  /  port,                     [I]
                                   /   2810,                     [DEFAUT]
             ◊  FICHIERS_ENTREE     =  l_arg ,                  [l_TX]
             ◊  FICHIERS_SORTIE     =  l_arg ,                  [l_TX]
             ◊  ♦  NOM_PARA         =  l_arg ,                  [l_TX]
                ♦  VALE             =  l_arg ,                  [l_TX]
             ),

       ◊  CODE_RETOUR_MAXI =  /  icode,                         [I]
                              /  0,                             [DEFAUT]
       ◊  INFO   =  /  1  ,                                     [I]
                    /  2  ,                                     [DEFAUT]
)
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Opérandes

3.1 Opérande LOGICIEL

◊ LOGICIEL = nom_exe

Nom de la commande ou de l'exécutable à appeler. C'est une chaîne de caractères (entre '), il faut préciser le chemin complet pour atteindre l'exécutable.
Cet opérande peut également être utilisé pour surcharger la commande par défaut lors de la création d'un maillage ou de l'appel de SALOME.

3.2 Mot clé ARGUMENT

◊ ARGUMENT = l_arg

Permet de définir la liste des arguments passés à l'exécutable. À chaque élément de la liste correspond un argument fourni à l'exécutable. Si un argument contient un espace, l'exécutable récupérera une chaîne de caractères contenant l'espace (voir mot-clé SHELL).
Ce mot clé est obligatoire dans le cas où l'on crée un maillage au format SALOME. Dans ce cas, il faut fournir un argument unique qui est le nom du fichier qui est produit par le script de maillage.

3.3 Mot clé SHELL

◊ SHELL = 'OUI'/'NON'

La bonne pratique consiste à exécuter un programme (mot-clé LOGICIEL) avec une liste d'arguments (mot-clé ARGUMENT). On laisse alors SHELL à sa valeur par défaut qui vaut 'NON'.
Parfois, on souhaite exécuter non pas un programme auquel on passe des arguments mais une ligne de commande complète (par exemple commençant par un test if). Dans ce cas, il faut utiliser SHELL='NON'. Le mot-clé LOGICIEL et les éventuels ARGUMENT[s] supplémentaires sont joints avec un espace pour composer la ligne de commande complète.
Voir la documentation du module subprocess de Python pour les écueils de SHELL='NON' (False).

3.4 Mot clé MAILLAGE

◊ MAILLAGE = _F (

Permet de produire un maillage en appelant directement depuis le fichier de commandes l'un des outils suivants : GMSH ou SALOME. Le maillage mail est retourné par l'opérateur.
Par défaut, la commande pour exécuter le mailleur est définie dans le fichier d'installation xxx/share/aster/external_programs.js.
On peut utiliser LOGICIEL (et ARGUMENT/SHELL) pour appeler un autre exécutable.

3.4.1 Opérande FORMAT

/ FORMAT = 'GMSH'

Création d'un maillage au format GMSH.

/ FORMAT = 'SALOME'

Création d'un maillage au format SALOME, ou plus généralement un fichier au format MED.
Dans ce cas le mot clé ARGUMENT est obligatoire et contient le nom du fichier MED produit par le script.

3.4.2 Opérande UNITE_GEOM

UNITE_GEOM = igeom

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Numéro d'unité logique associé au fichier de données utilisé pour créer le maillage.

3.5 Mot clé SALOME

SALOME = _F (

Permet d'exécuter un script dans une instance de SALOME qui doit déjà être lancée (Code_Aster ne lance pas SALOME dans ce cas et ne le termine pas non plus), sur la même machine que la machine d'exécution de Code_Aster ou sur une machine distante. C'est le lanceur salome qui gère le contact d'une instance distante.
Le script SALOME doit être à la syntaxe Python et être exécutable depuis SALOME via un File/*Load Script*. Il doit suivre un certain nombre de conventions d'écriture, notamment sur les variables utilisées pour les fichiers d'entrée (c'est-à-dire utilisés par le script) et les fichiers de sortie (c'est-à-dire générés par le script), car des ajustements/remplacements sont opérés avant l'exécution dans SALOME.
Par défaut, on utilise le script …/outis/salome mais on peut surcharger cette valeur en utilisant le mot-clé LOGICIEL.
En cas de connexion à une session distante, il est nécessaire d'indiquer, avec le mot-clé LOGICIEL, l'emplacement du script sur la machine distante. À défaut, on supposera qu'il est installer au même endroit que sur la machine locale (et donc que le répertoire d'installation de Code_Aster est le même).

3.5.1 Opérande CHEMIN_SCRIPT

CHEMIN_SCRIPT = '/../script.py'

Ce mot-clé permet de spécifier le chemin du script SALOME. On peut utiliser un chemin absolu (/home/user/mon-script.py) ou relatif (./fort.99 va ouvrir le fichier fort.99 contenu dans le répertoire temporaire d'exécution de Code_Aster).
Si le script a besoin de fichiers en entrée et/ou crée des fichiers de sortie, on utilise les mots-clés FICHIERS_ENTREE et FICHIERS_SORTIE. Voir les deux paragraphes ci-après.

3.5.2 Opérande MACHINE

MACHINE = machine

Ce mot-clé permet de spécifier le nom de la machine ou son adresse IP sur laquelle est ouverte la session de SALOME à contacter. Pou utiliser la machine locale, il ne faut pas utiliser ce mot-clé.

3.5.3 Opérande UTILISATEUR

UTILISATEUR = utilisateur

Ce mot-clé est obligatoire si MACHINE est renseigné. Il s'agit du nom de l'utilisateur qui a démarré l'instance SALOME sur la machine distante.

3.5.4 Opérande PORT

PORT = port

Ce mot-clé permet de spécifier le port de l'instance SALOME à laquelle on cherche à se rattacher. Ce port est affiché lors du lancement de SALOME à partir d'un terminal. Par défaut, il vaut 2810 mais il faut le vérifier !

3.5.5 Opérande FICHIERS_ENTREE

◊ FICHIERS_ENTREE = ['/../fichier_in1', '/../fichier_in2', ...]

Ce mot-clé permet de spécifier la liste des fichiers de données qui pourront être utilisés par le script SALOME (par exemple un fichier MED si le script correspond à un post-traitement).
Pour cela, le script sera modifié en remplaçant les chaînes de caractères INPUTFILE1 à INPUTFILEn par les n fichiers de FICHIERS_ENTREE. Le cas échéant, le lanceur salome se charge

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

de copier ces fichiers sur la machine distante, puis de (re)modifier le script pour y écrire le nom des fichiers après recopie.

3.5.6 Opérande FICHIERS_SORTIE

◊ FICHIERS_SORTIE = ['/../fichier_out1', '/../fichier_out2', ...]

De manière symétrique à FICHIERS_ENTREE, on peut récupérer les fichiers générés par le script SALOME. Le script produit des fichiers sous les noms OUTPUTFILE1 à OUTPUTFILEp. Ces chaînes de caractères seront remplacées par les fichiers de FICHIERS_SORTIE.
En cas de lancement distant, le lanceur salome se charge de (re)modifier le script pour y insérer les noms utilisés sur la machine distante, puis de rapatrier les fichiers produits en utilisant les noms de FICHIERS_SORTIE.

3.5.7 Opérandes NOM_PARA et VALE

◊ NOM_PARA = ['para1', 'para2', ...]
◊ VALE = ['vale1', 'vale2', ...]

Ces deux mots-clés permettent de paramétriser le script SALOME en remplaçant chaque occurrence de para1 par vale1, para2 par vale2, etc.

3.6 Opérande INFO

◊ INFO = info

Dans le cas où INFO=2, les messages provenant de la commande exécutée sont imprimés dans le fichier MESSAGE. C'est la valeur par défaut. Cela permet de conserver la trace de l'exécution.

3.7 Opérande CODE_RETOUR_MAXI

◊ CODE_RETOUR_MAXI = icode

Valeur maximum du code retour renvoyé par la commande ou le logiciel qui est tolérée pour considérer que l'exécution s'est bien déroulée. Par défaut cette valeur vaut 0, si elle est affectée à –1, le code retour de la commande est ignoré.

4 Exemples

On peut trouver des exemples d'utilisation dans le cas-test zzzz151a.
Les macro-commandes sont également de bons exemples d'utilisation des différentes possibilités, notamment STANLEY pour l'exécution de script dans une session SALOME.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)