Opérateur CALC_EUROPLEXUS

1 But

Cet opérateur permet de préparer d'une étude Europlexus depuis un fichier de commandes *Code_Aster*.

La commande CALC_EUROPLEXUS permet de définir une étude Europlexus pour une exécution ultérieure.
Les fichiers de commandes Europlexus sont préparés à partir des arguments des concepts Aster.

Il n'y a pas d'objet résultat produit. Les fichiers d'entrée pour Europlexus sont écrits dans le répertoire REPE_OUT.

Remarque

Il est nécessaire de préciser le mot-clé DEBUG=_F(HIST_ETAPE='OUI') dans DEBUT/POURSUITE pour utiliser cette macro-commande.

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

2 Syntaxe

```
evol = CALC_EUROPLEXUS (

        ♦  /  NOM_CAS       =      nom,                    [TXM]
                            =      'study',                [DEFAUT]

        ♦  /  MODELE        =      mo,                     [modele]
              CARA_ELEM     =      carac,                  [cara_elem]
              CHAM_MATER    =      chmat,                  [cham_mater]

           /  ETAT_INIT =   _F (
                 ♦  RESULTAT         = evolno              [evol_noli]
                 ◊  CONTRAINTE       =  /   'NON'          [DEFAUT]
                                        /   ' OUI '        [TXM]
                 # si CONTRAINTE = 'OUI' :
                 ◊  VARI_INT         =  /   'NON'          [DEFAUT]
                                        /   'OUI'          [TXM]
                 ◊  VITESSE          =  /   'NON'          [DEFAUT]
                                        /   'OUI'          [TXM]
                 # si CONTRAINTE = 'NON' :
                 ◊  NITER            =  /   1              [DEFAUT]
                                        /   niter          [I]
                 ◊  EQUILIBRE        =  /   'OUI'          [DEFAUT]
                                        /   'NON'          [TXM]
                       ),
        ♦  COMPORTEMENT =   _F (
                 ♦  RELATION         = 'ELAS'              [DEFAUT]
                                     = 'GLRC_DAMAGE'       [TXM]
                                     = 'VMIS_ISOT_TRAC'    [TXM]
                                     = 'VMIS_JOHN_COOK '   [TXM]
                                     = 'BPEL_FROT'         [TXM]
                 ♦  GROUP_MA         = l_grma              [l_gr_GROUP_MA]

        ◊  FONC_PARASOL =   _F (
                 |       ♦  NFKT          = nfkt           [fonction_sdaster]
                         ♦  NFKR          = nfkr           [fonction_sdaster]
                 |       ♦  NFAT          = nfat           [fonction_sdaster]
                         ◊  NFAR          = nfar           [fonction_sdaster]
                 ♦  GROUP_MA      = gma                   [l_gr_GROUP_MA]
                           ),

        ♦  EXCIT         =   _F (
                 ♦  CHARGE    =      cho,                 [char_meca]
                 ♦  FONC_MULT =      fi,                  [fonction/formule]
                              ),

        ♦  CALCUL    =   _F (
              ♦  /  TYPE_DISCRETISATION =  'AUTO',         [DEFAUT]
                       ♦  CSTAB        =  /  0.3,         [DEFAUT]
                                          /  cstab,       [R]

                 /  TYPE_DISCRETISATION=   'UTIL',         [TXM]
                       ♦  PASFIX       =  pasfix,         [R]
              ♦  INST_INIT       =  tini,                 [R]
              ♦  INST_FIN        =  tfin,                 [R]
              ◊  NMAX            =  nmax,                 [R]
                           ),
```


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

```
◊  AMORTISSEMENT    =   _F (
      ♦  TYPE_AMOR     =   'QUASI_STATIQUE',       [DEFAUT]
      ♦  FREQUENCE     =   freq,                   [R]
      ♦  COEF_AMOR     =   amor,                   [R]
      ◊  INST_DEB_AMOR=    instd,                  [R]
      ◊  INST_FIN_AMOR=    instf,                  [R]
                           ),
◊  OBSERVATION      =   _F (
         ♦  NOM_CHAM  =   /  'DEPL'                  [DEFAUT]
                          /  'VITE'                  [TXM]
                          /  'ACCE'                  [TXM]
                          /  'SIEF_ELGA'             [TXM]
                          /  'EPSI_ELGA'             [TXM]
                          /  'VARI_ELGA'             [TXM]

         ♦  /  PAS_INST  =  pas_inst                 [R]
            /  PAS_NBRE  =  pas_nbre                 [I]
            /  INST      =  l_inst                   [listr8]
            /  NUME_ORDRE = l_nume                   [l_I]

         ◊  /  GROUP_NO     =  grno                  [l_gr_noeud]
            /  TOUT_GROUP_NO=  'OUI'                 [TXM]

         ◊  /  GROUP_MA     =  grma                  [l_gr_GROUP_MA]
            /  TOUT_GROUP_MA=  'OUI'                 [TXM]

                           ),

♦  ARCHIVAGE =   _F (
         ♦  /  PAS_INST  =  pinst                    [R]
            /  PAS_NBRE  =  pnbre                    [I]
            /  INST      =  l_inst                   [listr8]
            /  NUME_ORDRE =  l_nume                  [l_I]
                  )

◊  COURBE =   _F (
         ◊  UNITE_ALIT    =  ualit                   [I]
         ◊  NOM_CHAM      =  ncham                   [TXM]
         ◊  NOM_CMP       =  ncmp                    [TXM]
         ◊  /  GROUP_NO   =  noeud                   [l_gr_noeud]
            /  GROUP_MA   =  grma                    [l_gr_maille]
               ◊  NUM_GAUSS = n                      [I]
         ♦  /  PAS_INST_COURBE     =  picourbe       [R]
            /  PAS_NBRE_COURBE     =  pncourbe       [I]
            /  INST_COURBE         =  l_inst         [listr8]
            /  NUME_ORDRE_COURBE   =  l_nume         [l_I]
            ),

◊  DOMAINES  =   _F (
         ◊  GROUP_MA  =  dom_gma                     [l_gr_maille]
         ◊  IDENTIFIANT  =  dom_id                   [I]
               ),
◊  INTERFACES    =   _F (
            ◊  GROUP_MA_1    =  int_gma1            [l_gr_maille]
            ◊  GROUP_MA_2    =  int_gma2            [l_gr_maille]
            ◊  TOLE   =  tole                       [R]
            ◊  IDENT_DOMAINE_1  =  int_dom1         [I]
            ◊  IDENT_DOMAINE_2  =  int_dom2         [I]
                  ),
```


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

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


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

3 Restrictions d'usage

CALC_EUROPLEXUS définit une étude Europlexus. Pour cela, il utilise, non seulement les concepts, mais aussi les mots-clés renseignés dans certaines des commandes précédentes (AFFE_CARA_ELEM, AFFE_CHAR_MECA …).
Pour cela, il est nécessaire d'ajouter le mot-clé DEBUG=_F(HIST_ETAPE='OUI') dans DEBUT ou POURSUITE .

Pour la même raison, il est obligatoire que ces commandes « parentes » soient définies dans le jeu de commandes au même niveau que CALC_EUROPLEXUS (et non dans un INCLUDE par exemple).

Enfin, les affectations dans ces mêmes commandes doivent être faites par groupes de mailles (mots-clés GROUP_MA*) et non par mailles.

4 Opérandes

4.1 Opérande NOM_CAS

Les fichiers d'entrée pour Europlexus sont écrits dans REPE_OUT dans le répertoire temporaire de l'exécution. Pour récupérer ce répertoire en fin de calcul, il faut l'indiquer dans AsterStudy avec *Set-up directories* ou dans astk avec le type repe en sortie.

Pour préparer plusieurs mises en donnée, les noms de fichiers sont construits à partir de ce mot-clé : commandes_ + NOM_CAS.
Par défaut, NOM_CAS vaut study.

4.2 Opérandes MODELE / CHAM_MATER / CARA_ELEM

```
♦  /  MODELE       =     mo,
      CARA_ELEM    =     carac,
      CHAM_MATER   =     chmat,
```

Ces mots-clés permettent de renseigner :

- le nom du modèle (mo) dont les éléments font l'objet du calcul mécanique. Seuls les calculs mécaniques sont autorisés, et pour les modélisations Q4GG, BARRE (section GENERALE dans AFFE_CARA_ELEM), POU_D_E (section rectangulaire et circulaire dans AFFE_CARA_ELEM), 3D (maille HEXA8 et TETRA4), 3D_SI (maille HEXA8), DIS_T et DIS_TR.
- le nom du champ de matériau (chmat) affecté sur le maillage. Attention, tous les groupes de mailles du modèle doivent être associées à un matériau.
- le nom des caractéristiques (carac) des éléments de coque, poutre, tuyau, barre, câble, et éléments discrets affectés sur le modèle mo . Cet opérande n'est pas obligatoire si aucun élément du modèle n'en a besoin (ex : 3D).

4.3 Mot clé ETAT_INIT

```
♦  /  ETAT_INIT    = _F(
      ♦  RESULTAT     =     resu,
      ◊  CONTRAINTE
      ◊  VARI_INT
      ◊  VITESSE
      ◊  NITER
      ◊  EQUILIBRE
```


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

Ce mot-clé permet à Europlexus de débuter le calcul à partir d'un état initial issu d'un concept résultat de Code_Aster. Les champs fournis à Europlexus sont ceux correspondant au dernier instant de calcul du concept résultat donné.

Le champ de déplacement est transmis dans tout les cas. Le champ de vitesse peut être transmis dans le cas d'un état initial dynamique (VITESSE = 'OUI'), mais même dans ce cas, le champ d'accélération n'est pas transmis car il est inutile à EPX qui le calcule automatiquement.

Remarque :

On récupère le modèle, le champ de matériau et les caractéristiques élémentaires du concept résultat.

4.3.1 Opérande RESULTAT

♦ RESULTAT

Concept résultat fournissant le maillage et les champs de déplacements et de contraintes qui serviront d'état initial à Europlexus.

4.3.2 Opérande CONTRAINTE

◊ CONTRAINTE

Si CONTRAINTE ='OUI', le champ de contraintes SIEF_ELGA fait partie de l'état initial, sinon seul le champ de déplacement est donné, les contraintes initiales sont alors calculées à partir des déplacements par Europlexus.

Seules les modélisations BARRE, Q4GG , 3D et 3D_SI sont compatibles avec CONTRAINTE ='OUI'. Contrairement aux autres éléments, pour les 3D_SI , les modèles Code_Aster et EPX n'étant pas tout à fait les mêmes, un état initial équilibré dans Code_Aster ne le sera pas dans EPX. La fonctionnalité est tout de même activer mais il est nécessaire dans ce cas d'activer le mot-clé EQUI .

4.3.3 Opérande VARI_INT

◊ VARI_INT

Si VARI_INT ='OUI', un champ de variables internes (VARI_ELGA) fait partie de l'état initial. Ce champ n'est pas nul uniquement sur les mailles affectées d'un comportement pour lequel une transformation des variables internes de Code_Aster vers EPX est développée. Des messages d'alarmes préviennent lorsque des variables internes mises à zéro sont envoyées à EPX et quand le champ de variables internes est nul sur toutes les mailles du modèle (ce dernier précisant alors que l'utilisation de l'opérande est sans effet).
Cet opérande n'est disponible que si CONTRAINTE = 'OUI'.

4.3.4 Opérande VITESSE

◊ VITESSE

Si VITESSE ='OUI', le champ de vitesse (VITE) fait partie de l'état initial qui est donc dynamique.
Cet opérande n'est disponible que si CONTRAINTE = 'OUI'.

4.3.5 Opérande NITER

Si CONTRAINTE ='NON', cet opérande indique à Europlexus en combien d'étapes (pas de temps) recalculer les contraintes à partir du déplacement. Lors de ce calcul le déplacement initial donné est

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

considéré comme un déplacement imposé. À l'étape $i=1,\ldots,niter$, le déplacement imposé est multiplié par $\frac{i}{niter}$.

4.3.6 Opérande EQUILIBRE

◊ EQUILIBRE

Lorsqu'un résultat est transféré d'un code de calcul à un autre, il arrive souvent qu'un état équilibré dans le premier code ne le soit plus tout à fait dans le second. Pour éviter cela, Europlexus dispose d'une fonctionnalité pour équilibrer parfaitement un état initial en ajoutant ce qu'il faut aux forces externes. Cette fonctionnalité sera activée si on donne EQUILIBRE='OUI' (valeur par défaut).

4.4 Mot clé COMPORTEMENT

```
♦ COMPORTEMENT =   _F  (
                ♦ RELATION        =  'ELAS'              [DEFAUT]
                                  =  'GLRC_DAMAGE'       [TMX]
                                  =  'VMIS_ISOT_TRAC'    [TMX]
                                  =  'VMIS_JOHN_COOK'    [TMX]
                                  =  'BPEL_FROT'         [TMX]
                ♦   GROUP_MA      =  l_grma              [l_gr_GROUP_MA]
```

Sur le modèle des opérateurs STAT_NON_LINE et DYNA_NON_LINE, le mot-clé COMPORTEMENT permet d'affecter un comportement aux groupes de mailles modélisés dans le calcul.

Les seuls comportements disponibles sont 'ELAS', 'GLRC_DAMAGE', 'VMIS_ISOT_TRAC', 'VMIS_JOHN_COOK' et 'BPEL_FROT'. Ils sont renseignés par le mot-clé RELATION.

4.5 Mot clé FONC_PARASOL

♦ FONC_PARASOL

Ce mot clé permet de définir les fonctions de raideur, et d'amortissement, de translation et de rotation des supports élastiques définies dans les mots-clé facteurs RIGI_PARASOL et DISCRETS de la commande AFFE_CARA_ELEM.

Il est possible de combiner à la fois un tapis de ressorts de sol à un tapis d'amortisseurs (type dashpots). Dans ce cadre, on peut mélanger les descriptions : par exemple coupler des ressorts K_TR_D_N à 6 composantes, dont les raideurs de rotation, avec des amortisseurs de type A_T_D_N . Dans Europlexus, les coefficients d'amortissement en rotation valent alors implicitement 0.

Bien entendu, les descriptions homogènes au niveau des degrés de liberté , de type K_TR_D_N avec A_TR_D_N ou K_T_D_N avec A_T_D_N sont aussi autorisées. Dans tous les cas, les arguments qui suivent (NFKT , NFKR , NFAT et NFAR) doivent être spécifiés en cohérence avec les degrés de liberté de raideur et d'amortissement.

4.5.1 Opérande NFKT

♦ NFKT

Permet de définir la fonction de translation des raideurs suivants les axes globaux.

4.5.2 Opérande NFKR

♦ NFKR

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

Permet de définir la fonction de rotation des raideurs suivants les axes globaux.

4.5.3 Opérande NFAT

♦ NFAT

Permet de définir la fonction de translation des amortissements suivants les axes globaux.

4.5.4 Opérande NFAR

♦ NFAR

Permet de définir la fonction de rotation des amortissements suivants les axes globaux.

4.5.5 Opérande GROUP_MA

♦ GROUP_MA

Renseigne les groupes de mailles auxquelles vont être affectées les différentes fonctions déclarées dans l'occurrence du mot-clé facteur.

Remarques :

- Si des caractéristiques élémentaires sont déclarées sur un groupe de mailles par le mot-clé DISCRET de AFFE_CARA_ELEM alors ce groupe de mailles doit nécessairement figurer dans cet opérande pour une des occurrences de FONC_PARASOL.
- Si des ressorts de sol sont déclarés sur un groupe de mailles par l'opérande GROUP_MA_POI1 du mot-clé RIGI_PARASOL de AFFE_CARA_ELEM alors ce groupe de mailles doit nécessairement figurer dans cet opérande pour une des occurrences de FONC_PARASOL.

4.6 Mot clé EXCIT

♦ EXCIT

Ce mot clé facteur permet de définir une charge à chaque occurrence. Ces charges sont issues de l'opérateur AFFE_CHAR_MECA.

4.6.1 Opérande CHARGE

♦ CHARGE

Les types de charges prises en compte par CALC_EUROPLEXUS sont les suivantes :

- DDL_IMPO : il permet de déclarer des déplacements imposés. Si ce sont des blocages, il est préférable de ne pas associer de fonction multiplicatrice à ce chargement. Ainsi les informations seront traduites par la liaison « BLOQ » d'EPX, ce qui permet de traiter tous les ddls dans une même occurrence. Si une fonction multiplicatrice est associée alors le chargement sera traduit par la liaison « DEPL » d'EPX, il peut alors n'y avoir qu'un seul mot-clé par occurrence parmi DX, DY, DZ, DRX, DRY, DRZ. Pour traduire un déplacement imposé autre qu'un blocage, il est obligatoire d'associer une fonction multiplicatrice au chargement.
- FORCE_COQUE/PRES : pression sur une coque, on doit obligatoirement associer à ce type de charge à un coefficient multiplicateur (mot-clé FONC_MULT). Les autres mots-clé de FORCE_COQUE ne sont pas autorisés.
- RELA_CINE_BP : relations cinématiques issues de la macro-commande DEFI_CABLE_BP, définit des liaisons entre degrés de liberté de béton et de câbles.
- PRES_REP/PRES : pression sur des faces d'éléments 3D, on doit obligatoirement associer à ce type de charge à un coefficient multiplicateur (mot-clé FONC_MULT). Les autres mots-clés de PRES_REP ne sont pas autorisés. **Attention, dans Code_Aster ce chargement est autorisé sur**

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

les coques mais cette utilisation dans CALC_EUROPLEXUS conduira à une erreur du calcul EPX.

- FORCE_NODALE : permet de déclarer des forces nodales de la même manière que pour un calcul avec les autres opérateurs de Code_Aster. Comme pour DDL_IMPO, on ne peut déclarer qu'un seul mot-clé parmi FX, FY, FZ, MX, MY et MZ. On doit obligatoirement associer à ce type de charge à un coefficient multiplicateur (mot-clé FONC_MULT).
- LIAISON_MAIL : pour que CALC_EUROPLEXUS puisse traiter cette liaison, il est impératif de spécifier LIAISON_EPX='OUI' à AFFE_CHAR_MECA.

4.6.2 Opérande FONC_MULT

◊ FONC_MULT

Fonction du temps multiplicatrice du chargement défini par l'occurrence actuelle du mot clé facteur EXCIT.

4.7 Mot clé CALCUL

♦ CALCUL

Ce mot clé facteur permet de choisir les paramètres de calcul à utiliser.

4.7.1 Opérande TYPE_DISCRETISATION

♦ TYPE_DISCRETISATION = 'AUTO', [DEFAUT]
'UTIL',

Permet de choisir entre une discrétisation automatique en temps ('AUTO', il faut alors préciser CSTAB) et une discrétisation imposée par l'utilisateur ('UTIL', il faut alors préciser PASFIX). INST_INI et INST_FIN précisent le temps initial et final du calcul.

4.7.2 Opérande CSTAB

♦ CSTAB = / 0.3, [DEFAUT]
/ cstab,

Coefficient de sécurité pris au cours du pas de temps de stabilité estimé (c'est-à-dire critique) pour chaque élément. La valeur par défaut est de $0,3$.

4.7.3 Opérande PASFIX

♦ PASFIX

Ce paramètre est un raccourci permettant d'assigner un pas de temps utilisateur fixe. Doit être utilisé en conjonction de TYPE_DISCRETISATION = UTIL.

4.7.4 Opérande INST_INIT

♦ INST_INIT

Instant initial du calcul. Dans le cas d'un calcul de reprise, ce paramètre est ignoré (il peut cependant être laissé), car la nouvelle valeur du temps initial est lue dans le fichier de reprise.

4.7.5 Opérande INST_FIN

♦ INST_FIN

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

Instant final du calcul.

4.7.6 Opérande NMAX

◊ NMAX
Nombre maximum de pas de calcul. La valeur pas défaut est : 1000000 .

4.8 Mot clé AMORTISSEMENT

♦ AMORTISSEMENT

Ce mot clé facteur permet de d'ajouter de l'amortissement au calcul EPX.

4.8.1 Opérande TYPE_AMOR

♦ TYPE_AMOR = 'QUASI_STATIQUE', [DEFAUT]

Permet de choisir le type d'amortissement souhaité. Pour le moment seul la valeur 'QUASI_STATIQUE' est disponible.

4.8.2 Opérande FREQUENCE

♦ FREQUENCE

Définit la fréquence d'amortissement.

4.8.3 Opérande COEF_AMOR

♦ COEF_AMOR

Définit le coefficient d'amortissement.

4.8.4 Opérandes INST_DEB_AMOR et INST_FIN_AMOR

◊ INST_DEB_AMOR
◊ INST_FIN_AMOR

Ces deux mots-clés permettent de donner l'instant à partir duquel l'amortissement doit être activé et celui à partir duquel il doit cesser. Ces deux mots-clés doivent être renseignés ensemble, s'ils ne le sont pas l'amortissement sera actif tout au long du calcul.

4.9 Mot clé OBSERVATION

Active l'impression dans le fichier de sortie « listing » d'EPX des champs souhaités sur les entités (nœuds ou éléments) demandés par l'utilisateur.

4.9.1 Opérande NOM_CHAM

◊ NOM_CHAM

Renseigne le nom des champs à écrire dans le listing.

4.9.2 Opérandes PAS_INST / PAS_NBRE / INST / NUME_ORDRE

◊ / PAS_INST
/ PAS_NBRE
/ INST

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

/ NUME_ORDRE

Détermine la liste des instants pour lesquels on souhaite l'affichage :
- par fréquence définie par l'intervalle de temps : PAS_INST ;
- par nombre de pas de temps : PAS_NBRE ;
- par une liste d'instants : INST ;
- par une liste de numéro d'ordre : NUME_ORDRE .

Les quatre mots-clés peuvent être renseignés simultanément. La liste des instants sélectionnés sera alors l'union des instants définis par les différents mots-clés.

4.9.3 Opérandes GROUP_NO / TOUT_GROUP_NO

◊ / GROUP_NO
/ TOUT_GROUP_NO

Détermine les nœuds pour lesquels on souhaite visualiser les informations :
- À certains nœuds, à travers la liste définie dans un groupe : GROUP_NO ;
- Pour tous les groupes de nœuds du maillage : TOUT_GROUP_NO.

4.9.4 Opérandes GROUP_MA / TOUT_GROUP_MA

◊ / GROUP_MA
/ TOUT_GROUP_MAILLE

Détermine les mailles pour lesquelles on souhaite visualiser les informations :
- Sur certaines mailles, à travers la liste définie dans un groupe : GROUP_MA ;
- Pour tous les groupes de mailles du maillage : TOUT_GROUP_MA.

4.10 Mot clé ARCHIVAGE

Permet de sélectionner les instants pour lesquels on souhaite que les résultats soient archivés dans le fichier au format MED écrit par EPX. C'est à partir de ce fichier qu'est reconstruit le résultat Aster. La récupération de ces informations est assez coûteuse, il est conseillé de n'archiver que les instants strictement nécessaires au post-traitement ou à la poursuite du calcul.

4.10.1 Opérandes PAS_INST / PAS_NBRE / INST / NUME_ORDRE

Voir § 4.9.2 .

4.11 Mot clé COURBE

La directive EPX « SORTIE GRAP » permet, grâce à son mot-clé « COURBE », de stocker sous forme de tableau les valeurs au cours du temps de différentes grandeurs, typiquement une composante d'un champ pour un nœud ou un point de Gauss.

4.11.1 Mot clé NOM_CHAM / NOM_CMP

◊ NOM_CHAM
◊ NOM_CMP

Champ et composante choisis.

4.11.2 Opérande GROUP_NO

◊ GROUP_NO

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

Nœud dont on souhaite stocker la composante d'intérêt. Le groupe de nœuds doit contenir qu'un seul nœud.

4.11.3 Opérandes GROUP_MA et NUM_GAUSS

◊ GROUP_MA
◊ NUM_GAUSS

Maille dont on souhaite stocker la composante d'intérêt. Le groupe de maille ne doit contenir qu'une seule maille.

NUM_GAUSS indique le numéro du point de Gauss de l'élément à post-traiter. Si l'élément possède moins de points de Gauss que le numéro demandé alors EPX échouera à cette étape.

4.11.4 Opérande NOM_COURBE

Nom de la colonne concernant la présente occurrence de COURBE dans la table créée. Le nom ne doit pas faire plus de 16 caractères sans quoi il sera tronqué.

4.11.5 Opérandes PAS_INST_COURBE / PAS_NBRE_COURBE / INST_COURBE / NUME_ORDRE_COURBE

♦ / PAS_INST_COURBE
/ PAS_NBRE_COURBE
/ INST_COURBE
/ NUME_ORDRE_COURBE

Ces mots-clé définissent les instants d'archivages pour le mot-clé COURBE.
On suit la même logique qu 'au paragraphe 4.9.2.

Remarque :
Toutes les valeurs archivées sous le mot-clé COURBE partagent la même liste d'instants d'archivage.

4.12 Mot clé DOMAINES

Définit les sous domaines pour les études multi-domaines.

4.12.1 Mot clé GROUP_MA

◊ GROUP_MA

Pour chaque sous domaine il faut disposer d'un groupe de maille GROUP_MA.

4.12.2 Mot clé IDENTIFIANT

◊ IDENTIFIANT

Chaque sous domaine définit par son groupe de mailles, doit également posséder un identifiant.

4.13 Mot clé INTERFACES

◊ INTERFACES

Définit les interfaces entre les sous domaines précédemment définis.

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

4.13.1 Mot clé GROUPE_MA_1 / GROUPE_MA_2

◊ GROUPE_MA_1
◊ GROUPE_MA_2

Ces deux groupes de mailles définissent les bords des deux sous domaines en contact entre eux.

4.13.2 Mot clé TOLE

◊ TOLE

Précise la tolérance utilisée pour apparier les nœuds des bords GROUP_MA_1 et GROUP_MA_2.

4.13.3 Mot clé IDENT_DOMAINE_1 / IDENT_DOMAINE_2

◊ IDENT_DOMAINE_1
◊ IDENT_DOMAINE_2

Ces deux mots font référence aux identifiants respectifs des sous domaines, déjà précisés.

4.14 Mot clé INFO

◊ INFO = / 1 , [DEFAUT]
/ 2 ,

Permet de contrôler le niveau de message de la macro-commande.

Remarques :

• Si le mot-clé IMPR_MACRO='OUI' est présent dans la commande DEBUT , alors toutes les commandes de la macro-commande CALC_EUROPLEXUS seront imprimées dans le fichier de messages.
• Le mot-clé INFO est transmis à toutes les commandes utilisées dans la macro-commande. INFO = 2 peut donc imprimer beaucoup d'informations.
• Dans tous les cas, le fichier de commandes Europlexus est imprimé dans le fichier de messages.

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