

Opérateur CREA_RESU

1 But

Créer ou enrichir une structure de données `resultat` à partir de champs aux nœuds. Affectation possible des champs aux nœuds pour différents numéros d'ordre.

L'affectation par l'intermédiaire d'un `cham_no` de fonction produit par CREA_CHAMP [U4.72.04] s'effectue en évaluant chaque fonction à l'aide du paramètre représentant le temps fourni sous les mots clés `LIST_INST` ou `INST`.

Le concept produit par cet opérateur est, pour le moment, de type `evol_elas`, `evol_noli`, `evol_ther`, `mult_elas`, `fourier_elas`, `fourier_ther`, `evol_varc` ou `evol_char`.

De plus, trois fonctionnalités particulières sont accessibles dans cet opérateur :

- la création d'un concept de type `evol_char` par affectation de champ ou une formule analytique ;
- la création d'un concept `resultat` simulant la réorganisation des assemblages combustibles ;
- la projection d'un transitoire thermique 1D sur un maillage axisymétrique 3D.

2 Syntaxe

```
resu [resultat] = CREA_RESU (
```

```
    ◊ reuse = resu,
```

```
    ◆ OPERATION = / 'AFFE',
                  / 'ECLA_PG',
                  / 'PERM_CHAM',
                  / 'PROL_RTZ',
                  / 'PREP_VRC1',
                  / 'PREP_VRC2',
                  / 'ASSE',
```

Construction d'un résultat par affectations ou évaluations successives
de cham_no : (OPERATION : 'AFFE')

```
    ◆ TYPE_RESU          = 'MULT_ELAS' ,
    ◆ NOM_CHAM           = 'DEPL',
    ◆ AFFE = _F (
        ◆ CHAM_GD        = chno,          [cham_no]
        ◊ NOM_CAS        = nomc,          [Kn]
        ◊ MODELE         = mo,            [modele]
        ◊ CHAM_MATER      = chmat,         [cham_mater]
        ◊ CARA_ELEM       = carac,         [cara_elem]
    ),
```

```
    ◆ TYPE_RESU          = / 'EVOL_ELAS',
                          / 'EVOL_NOLI',
    ◆ NOM_CHAM           = 'DEPL',
    ◆ AFFE = _F (
        ◆ CHAM_GD        = chno,          [cham_no]
        ◊ MODELE         = mo,            [modele]
        ◊ CHAM_MATER      = chmat,         [cham_mater]
        ◊ CARA_ELEM       = carac,         [cara_elem]
        ◆ / INST         = linst,         [l_R8]
        / LIST_INST      = litps,         [listr8]
        ◊ NUME_INIT       = numi,         [I]
        ◊ NUME_FIN        = numf,         [I]
        ◊ PRECISION       = /prec,        [R]
                          / 0.0,          [DEFAULT]
        ◊ CRITERE         = / 'RELATIF',  [DEFAULT]
                          / 'ABSOLU',
    ),
```

```
    ◆ TYPE_RESU          = 'FOURIER_ELAS',
    ◆ NOM_CHAM           = 'DEPL',
    ◆ AFFE = _F (
        ◆ CHAM_GD        = chno,          [cham_no]
        ◊ MODELE         = mo,            [modele]
        ◊ CHAM_MATER      = chmat,         [cham_mater]
        ◊ CARA_ELEM       = carac,         [cara_elem]
        ◊ NUME_MODE       = num,          [I]
        ◊ TYPE_MODE       = / 'SYME',     [DEFAULT]
                          / 'ANTI',
                          / 'TOUS',
    ),
```

```

♦ TYPE_RESU = 'FOURIER_THER',
♦ NOM_CHAM = 'TEMP',
♦ AFPE = _F (
    ♦ CHAM_GD = chno, [cham_no]
    ♦ MODELE = mo, [modele]
    ♦ CHAM_MATER = chmat, [cham_mater]
    ♦ CARA_ELEM = carac, [cara_elem]
    ♦ NUME_MODE = num, [I]
    ♦ TYPE_MODE = / 'SYME', [DEFAULT]
                  / 'ANTI',
                  / 'TOUS',
                ),

♦ TYPE_RESU = 'EVOL_THER',
♦ NOM_CHAM = / 'TEMP',
              / 'HYDR_ELGA',
♦ AFPE = _F (
    ♦ CHAM_GD = chno, [cham_no]
    ♦ MODELE = mo, [modele]
    ♦ CHAM_MATER = chmat, [cham_mater]
    ♦ CARA_ELEM = carac, [cara_elem]
    ♦ / INST = linst, [l_R8]
      / LIST_INST = litps, [listr8]
    ♦ NUME_INIT = numi, [I]
    ♦ NUME_FIN = numf, [I]
    ♦ PRECISION = / prec, [R]
                  / 0.0, [DEFAULT]
    ♦ CRITERE = / 'RELATIF', [DEFAULT]
                / 'ABSOLU',
              ),

♦ TYPE_RESU = 'EVOL_VARC',
♦ NOM_CHAM = 'IRRA',
♦ AFPE = _F (
    ♦ CHAM_GD = chno, [cham_no]
    ♦ MODELE = mo, [modele]
    ♦ CHAM_MATER = chmat, [cham_mater]
    ♦ CARA_ELEM = carac, [cara_elem]
    ♦ / INST = linst, [l_R8]
      / LIST_INST = litps, [listr8]
    ♦ NUME_INIT = numi, [I]
    ♦ NUME_FIN = numf, [I]
    ♦ PRECISION = / prec, [R]
                  / 0.0, [DEFAULT]
    ♦ CRITERE = / 'RELATIF', [DEFAULT]
                / 'ABSOLU',
              ),

♦ TYPE_RESU = 'MODE_MECA',
♦ NOM_CHAM = 'DEPl',
♦ MATR_A = ma, [matr_asse_depl_r]
♦ MATR_B = mb, [matr_asse_depl_r]
♦ AFPE = _F (
    ♦ CHAM_GD = chno, [cham_no]
    ♦ MODELE = mo, [modele]
    ♦ CHAM_MATER = chmat, [cham_mater]
    ♦ CARA_ELEM = carac, [cara_elem]
    ♦ FREQ = freq, [l_R8]
    ♦ NUME_MODE = numo, [I]
                ),

```

/ # Construction d'un concept de type EVOL_CHAR par affectation ou évaluation

d'un cham_no

```

♦ TYPE_RESU                = 'EVOL_CHAR',
♦ NOM_CHAM                 = 'PRES',
♦ AFFE = _F (
    ♦ CHAM_GD              = chno,          [cham_no]
    ◇ MODELE               = mo,            [modele]
    ◇ CHAM_MATER           = chmat,         [cham_mater]
    ♦ / ♦ INST             = linst,         [l_R8]
    / ♦ LIST_INST          = litps,         [listr8]
        ◇ NUME_INIT        = numi,         [I]
        ◇ NUME_FIN         = numf,         [I]
    ◇ PRECISION            = / prec,        [R]
        / 0.0,              [DEFAULT]
    ◇ CRITERE              = / 'RELATIF',   [DEFAULT]
        / 'ABSOLU',
),

```

/ # Construction d'un résultat sur un maillage éclaté pour visualisation ou
post-traitement (OPERATION : 'ECLA_PG')

```

♦ TYPE_RESU                = / 'EVOL_ELAS',
                          / 'EVOL_NOLI',
                          / 'EVOL_THER',

♦ ECLA_PG= _F (           voir [U4.44.14]           ),

```

/ # Construction d'un résultat dédié aux assemblages combustibles
(OPERATION : 'PERM_CHAM')

```

♦ TYPE_RESU                = 'EVOL_NOLI',

♦ NOM_CHAM                 = | 'DEPl' ,
                          | 'SIEF_ELGA',
                          | 'VARI_ELGA',

♦ RESU_INIT                = resu_2,         [evol_noli]
◇ INST_INIT               = tf,             [R]
◇ PRECISION                = / prec,
                          / 1.0E-6,         [DEFAULT]
◇ CRITERE                  = / 'ABSOLU',
                          / 'RELATIF',

♦ MAILLAGE_INIT            = ma_1,           [maillage]
♦ RESU_FINAL               = resu,           [evol_noli]
♦ MAILLAGE_FINAL           = mo_2,           [maillage]
♦ PERM_CHAM = _F (
    ♦ GROUP_MA_FINAL        = gma_2,         [gr_ma]
    ♦ GROUP_MA_INIT         = gma_1,         [gr_ma]
    ♦ TRAN                   = tx,ty,tz),    [l_R]
    ◇ PRECISION              = / prec ,
                          / 1.0E-3,         [DEFAULT]
),

```

/ # Projection d'un transitoire 1D sur un maillage axisymétrique
(OPERATION = 'PROL_RTZ')

```

♦ TYPE_RESU                = 'EVOL_THER'

♦ PROL_RTZ= _F (
    ♦ MAILLAGE_FINAL        = ma_3D,         [maillage]
    ♦ TABLE                = post_1D,       [table]
    ◇ / INST                = inst,          [R]
    / LIST_INST             = linst,         [l_R]
    ◇ PRECISION              = / prec,

```

```

                                / 1.0E-6,      [DEFAULT]
◇ CRITERE                      = / 'ABSOLU',
                                / 'RELATIF', [DEFAULT]
◇ PROL_DROITE                  = / 'EXCLU',      [DEFAULT]
                                / 'LINEAIRE',
◇ PROL_GAUCHE                  = / 'EXCLU',      [DEFAULT]
                                / 'LINEAIRE',
                                / 'CONSTANT',

◇ REPERE                       = 'CYLINDRIQUE',
◇ ORIGINE                      = (ori1,ori2,ori3),  [1_R]
◇ AXE_Z                        = (axe1,axe2,axe3),  [1_R]
                                ),

/ # Construction d'un résultat de type EVOL_THER pour calculer la
# température dans les couches des coques de type multicouche à partir
# d'un champ de fonctions du temps et de l'espace (épaisseur)
# (OPERATION : 'PREP_VRC1')
    ◇ TYPE_RESU                = 'EVOL_THER'
    ◇ PREP_VRC1 = _F (
        ◇ CHAM_GD              = chno,          [cham_no]
        ◇ MODELE                = mo,           [modele]
        ◇ CARA_ELEM             = carac,        [cara_elem]
        ◇ INST                  = inst,         [1_R8]
    ),

/ # Construction d'un résultat de type EVOL_THER pour calculer la
# température dans les couches des coques multicouche à partir d'un
# evol_ther "coque" contenant TEMP/TEMP_INF/TEMP_SUP
# (OPERATION : 'PREP_VRC2')
    ◇ TYPE_RESU                = 'EVOL_THER'
    ◇ PREP_VRC2 = _F (
        ◇ EVOL_THER             = evol,         [evol_ther]
        ◇ MODELE                = mo,           [modele]
        ◇ CARA_ELEM             = carac,        [cara_elem]
    ),

/ # Création par assemblage de structures de données résultat evol_ther :
# (OPERATION : 'ASSE')
    ◇ TYPE_RESU                = 'EVOL_THER'
    ◇ ASSE = _F (
        ◇ RESULTAT              = evol,         [evol_ther]
        ◇ TRANSLATION           = / tr,         [R]
                                / ~~~         [DEFAULT]
    ),
)

Si TYPE_RESU : 'MULT_ELAS'      alors resu de type mult_elas
Si TYPE_RESU : 'FOURIER_ELAS'   alors resu de type fourier_elas
Si TYPE_RESU : 'FOURIER_THER'   alors resu de type fourier_ther
Si TYPE_RESU : 'EVOL_THER'      alors resu de type evol_ther
Si TYPE_RESU : 'EVOL_VARC'      alors resu de type evol_varc
Si TYPE_RESU : 'EVOL_ELAS'      alors resu de type evol_elas
Si TYPE_RESU : 'EVOL_NOLI'      alors resu de type evol_noli
Si TYPE_RESU : 'EVOL_CHAR'      alors resu de type evol_char

```

3 Opérandes

3.1 Opérande OPERATION

- ♦ OPERATION = définit le type d'opération à effectuer avec cet opérateur :

'AFFE'	: création d'une structure de données résultat à partir de champs,
'ECLA_PG'	: création d'une structure de données sur un maillage éclaté pour visualisation,
'PERM_CHAM'	: réorganisation des assemblages combustibles,
'PROL_RTZ'	: prolongement d'un champ 1D sur une structure axisymétrique,
'PREP_VRC1'	: calcul de la température dans les couches d'une coque en partant d'une température TEMP= f(EPAIS,INST),
'PREP_VRC2'	: calcul de la température dans les couches d'une coque en partant d'une température calculée par aster avec un modèle de coques (TEMP/TEMP_INF/TEMP_SUP),
'ASSE'	: création d'une structure de données résultat à partir de plusieurs structures de données résultat mises bout à bout.

Ce mot clé permet de guider l'utilisateur lors de la construction du fichier de commande à l'aide de l'outil eficas.

La structure de données résultat est réentrante et pour OPERATION = 'AFFE' les champs existants peuvent être remplacés suivant la valeurs de la variable d'accès INST en utilisant les valeurs renseignées derrière les mots clés PRECISION et CRITERE . Quand il y a remplacement d'un champ existant, le code émet un message d'alarme, sinon les champs sont stockés à la fin de la structure de données.

3.2 Opérande TYPE_RESU

- ♦ TYPE_RESU

Type de la structure de données résultat créée.

3.3 Opérande NOM_CHAM

- ♦ NOM_CHAM

Nom symbolique de la grandeur affectée.

3.4 Mot clé CHAM_GD

3.4.1 Opérande CHAM_NO

- ♦ CHAM_NO = chno

chno est :

- 1) soit un cham_no de fonction créé par la commande CREA_CHAMP [U4.72.04] et dans ce cas on évalue pour chaque nœud la fonction et pour chaque instant défini derrière LIST_INST ou INST on crée un cham_no de réels,
- 2) soit un cham_no de réels créé par la commande CRE_CHAMP (mot d'AFFE ou EXTR) et ce champ est dupliqué autant de fois que la liste d'instant définie derrière LIST_INST ou INST le nécessite.

3.4.2 Opérandes MODELE, CHAM_MATER, CARA_ELEM

Ces opérandes facultatifs sont utilisés pour permettre le remplissage des structures de données résultat. Ce remplissage est indispensable dans le cas où la commande CREA_RESU est appelée par MACRO_ELAS_MULT pour utiliser ensuite les commandes de post-traitement qui vont rechercher cette information dans la SD.

- ♦ MODELE = mo,

Nom du modèle dont les éléments font l'objet du calcul.

◇ CHAM_MATER = chmat,
Nom du champ de matériau.

◇ CARA_ELEM = carac,
Nom des caractéristiques des éléments structuraux (poutre, coque, discret, ...) s'ils sont utilisés dans le modèle. Lorsque OPERATION prend la valeur PREP_VRC1 ou PREP_VRC2, on y récupère les composantes EPAIS et COQU_NCOU.

3.4.3 Opérandes LIST_INST / NUME_INIT / NUME_FIN

◆ LIST_INST = litps
Liste de réels produite par DEFI_LIST_REEL [U4.34.01].

◇ NUME_INIT = nuini

◇ NUME_FIN = nufin

Les instants de calcul sont ceux définis dans le concept litps pris entre le nuini et le nufin numéro d'instant. En l'absence du mot clé NUME_FIN, c'est la taille de la liste de réels qui est prise en compte.

3.4.4 Opérandes INST

◆ INST = linst

Liste de réels : liste des instants pour lesquels le cham_no de fonction sera évalué, ou bien le cham_no de réels sera affecté.

Remarque :

Le numéro d'ordre créé dans le concept resultat est soit récupéré à partir de la valeur de la variable d'accès INST lorsque elle est présente, soit affecté à la valeur maximum immédiatement supérieure.

3.4.5 Opérandes PRECISION / CRITERE

Ces opérandes permettent d'affiner l'accès par variables d'accès réelles du temps.

| PRECISION = / prec [R]
/ 0.0 ou 1.0D-3 ou 1.0D-6 [DEFAULT]

Ce mot clé permet d'indiquer que l'on recherche tous les champs dont l'instant (respectivement la fréquence) se trouve dans l'intervalle "inst ± prec" (Cf. CRITERE).

Dans le cas où OPERATION = 'AFFE', la valeur par défaut prec est fixée à 0.0 pour éviter d'écraser un champ dont la valeur de l'instant est proche de celui que l'on traite. l'instant fourni ne sert pas à récupérer un champ dans la structure de données, c'est un attribut qu'il faut associer au champ que l'on stocke. En général, les champs que l'on stocke correspondent tous à des instants différents.

Dans le cas très rare où l'utilisateur souhaiterait écraser l'un des champs contenu dans la structure de données, il devra utiliser le mot clé PRECISION. Un message d'alarme indique alors le nom des champs concernés avec leurs instants de stockage, et la précision fournie par l'utilisateur:

| CRITERE = / 'RELATIF' [DEFAULT]
/ 'ABSOLU'

'RELATIF' : l'intervalle de recherche est : [inst (1 - prec), inst (1 + prec)]

'ABSOLU' : l'intervalle de recherche est : [inst - prec, inst + prec].

3.4.6 Opérandes NUME_MODE / TYPE_MODE

◇ NUME_MODE = num

Entier désignant le numéro de l'harmonique de Fourier du champ stocké dans un concept de type fourier_elas.

```
◇ TYPE_MODE = / 'SYME'  
               / 'ANTI'  
               / 'TOUS'
```

Définit le type du mode de Fourier stocké.

'SYME' : harmonique symétrique
'ANTI' : harmonique antisymétrique
'TOUS' : harmonique symétrique et antisymétrique

3.4.7 Opérande NOM_CAS

◆ NOM_CAS = nomc

Chaîne de caractères définissant la variable d'accès du champ stocké dans un concept de type mult_elas.

3.4.8 Opérandes NUME_MODE/FREQ

◇
◇ NUME_MODE = num
Entier désignant le numéro du mode dans le cas TYPE_RESU='MODE_MECA'.

◇ FREQ = freq
Valeur de la fréquence.

Remarque :

| l'utilisateur doit indiquer NUME_MODE et FREQ pour chacun des
| champs

3.4.9 Opérandes MATR_A/MATR_B

◇ MATR_A = ma
Matrice de rigidité correspondant au champs stockés dans le cas TYPE_RESU='MODE_MECA'.

◇ MATR_B = mb
Matrice de masse correspondant au champs stockés.

4 Opérandes associés aux champs aux points d'intégration

4.1 Mot clé ECLA_PG

Il est déconseillé d'utiliser directement la commande CREA_RESU, on préférera se reporter à la macro-commande, MACR_ECLA_PG (Voir [U4.44.14]).

5 Opérandes associés aux assemblages combustibles

5.1 Opérandes RESU_INIT

◆ RESU_INIT = rinit
Nom de la SD evol_noli contenant les champs à transférer sur le nouveau maillage.

5.2 Opérandes INST_INIT / PRECISION/CRITERE

◆ INST_INIT = iinit
Instant caractérisant dans la SD evol_noli indiquée sous RESU_INIT, les champs à transférer sur l'autre maillage. Par défaut, le dernier instant archivé est sélectionné

♦ PRECISION = prec

Précision utilisée pour rechercher l'instant spécifié par INST_INIT dans la SD evol_noli associée à RESU_INIT.

♦ CRITERE = / 'RELATIF' [DEFAULT]
/ 'ABSOLU'

Critère utilisé pour rechercher l'instant spécifié par INST_INIT dans la SD evol_noli associée à RESU_INIT.

5.3 Opérandes MAILLAGE_INIT

♦ MAILLAGE_INIT = maillagei

Nom du maillage sur lequel a été définie la SD evol_noli indiquée sous RESU_INIT.

5.4 Opérandes RESU_FINAL

♦ RESU_FINAL = resu

Nom de la SD evol_noli définie sur le nouveau maillage sur lequel seront transférés les champs. C'est aussi dans ce cas le nom du concept sortant de la commande CREA_RESU. La structure de données resu doit exister (elle aura été créée par exemple par la commande STAT_NON_LINE) et ne doit contenir qu'un seul numéro d'ordre.

5.5 Opérandes MAILLAGE_FINAL

♦ MAILLAGE_FINAL = mailfin

Nom de la structure de données maillage créée sur le nouveau maillage sur lequel seront transférés les champs.

5.6 Mot clé PERM_CHAM

5.6.1 Opérandes GROUP_MA_FINAL

♦ GROUP_MA_FINAL = gma_2

Nom du groupe de mailles du MAILLAGE_FINAL, lieu où les champs sont transférés dans RESU_FINAL.

5.6.2 Opérandes GROUP_MA_INIT

♦ GROUP_MA_INIT = gma_1

Nom du maillage sur lequel a été définie la SD evol_noli indiquée sous RESU_INIT.

5.6.3 Opérande TRAN

♦ TRAN = (tx,ty,tz)

Vecteur translation permettant d'obtenir géométriquement GROUP_MA_FINAL à partir de GROUP_MA_INIT. Il est nécessaire de fournir exactement 3 valeurs.

5.6.4 Opérande PRECISION

♦ PRECISION = prec

Précision absolue permettant de vérifier la bonne adéquation entre les mailles initiales et les mailles finales, par défaut la valeur est fixée à 10^{-3} .

6 Opérandes associés à la projection sur un maillage axisymétrique

6.1 Mot clé PROL_RTZ

Construction d'un transitoire thermique sur un maillage axisymétrique (3D) à partir de la donnée d'un transitoire thermique calculé sur un maillage 1D. Le transitoire 1D est donné sous la forme d'une structure de données `TABLE` issue de la commande `POST_RELEVÉ_T` possédant les paramètres suivants :

- la définition des instants ('INST'),
- les coordonnées des nœuds du maillage 1D ('COOR_X')
- la valeur des températures aux nœuds ('TEMP').

Les coordonnées de la table doivent nécessairement avoir pour origine le nœud de coordonnée 0. Les valeurs des températures peuvent éventuellement être prolongées de façon constante ou bien interpolées linéairement en fonction de la coordonnée 'COOR_X'.

6.1.1 Opérandes `MAILLAGE_FINAL`

♦ `MAILLAGE_FINAL = mailfin`

Nom du maillage sur lequel on effectue la projection, l'opérateur vérifie que le maillage est tridimensionnel.

6.1.2 Opérandes `TABLE`

♦ `TABLE = table`

Nom d'une structure de données `TABLE` issue de la commande `POST_RELEVÉ_T` contenant le transitoire thermique 1D. Les paramètres de cette table sont obligatoirement : 'INST', 'COOR_X' et 'TEMP'.

6.1.3 Opérandes `INST / LIST_INST / PRECISION / CRITERE`

♦ `INST = litps`

Liste de valeurs réelles.

♦ `LIST_INST = litps`

Liste de réels produite par `DEFI_LIST_REEL` [U4.34.01].

♦ `PRECISION = / prec [R]`
`/ 1-0D-6 [DEFAULT]`

Précision utilisée pour rechercher l'instant spécifié dans la `TABLE post_1D`.

♦ `CRITERE = / 'RELATIF',`
`/ 'ABSOLU',`

Critère utilisé pour rechercher l'instant spécifié dans la `TABLE post_1D`.

6.1.4 Opérandes `PROL_DROITE` et `PROL_GAUCHE`

La projection du transitoire est effectuée selon la coordonnée `COOR_X` considérée comme la coordonnée `r` dans le repère cylindrique du maillage 3D. On peut définir à l'aide de ces deux opérandes la façon de prolonger le champ au-delà des bornes définies par la plage de variation du paramètre 'COOR_X' dans la table.

♦ `PROL_DROITE` et `PROL_GAUCHE =`

Définissent le type de prolongement à droite (à gauche) du domaine de définition de la variable :

- 'CONSTANT' pour un prolongement avec la dernière (ou première) valeur de la fonction,
- 'LINEAIRE' pour un prolongement le long du premier segment défini (`PROL_GAUCHE`) ou du dernier segment défini (`PROL_DROITE`),
- 'EXCLU' si l'extrapolation des valeurs en dehors du domaine de définition du paramètre est interdite (dans ce cas si un calcul demande une valeur de la fonction hors du domaine de définition, le code s'arrêtera en erreur fatale).

6.1.5 Opérande REPERE/ORIGINE/AXE_Z

- ♦ REPERE = 'CYLINDRIQUE'

Le repère de travail pour projeter le transitoire est supposé cylindrique, le transitoire 1D étant considéré comme la variation radiale du champ de température. Les deux opérands suivants permettent d'effectuer un changement de repère.

- ♦ ORIGINE = (ori1,ori2,ori3)

Correspond à la position de l'origine du maillage 1D par rapport à l'origine du maillage 3D.

- ♦ AXE_Z = (axe1,axe2,axe3)

Définition de l'axe du repère cylindrique.

7 Opérands associés à la préparation des variables de commande

7.1 Mots clés PREP_VRC1 et PREP_VRC2

l'évolution thermique que l'on peut associer au champ de matériau par AFFE_MATERIAU/AFFE_VARC doit être prête à être utilisée par les éléments finis du modèle mécanique. Un problème se pose pour les éléments de type coque ou tuyau qui utilisent une température variant dans l'épaisseur sur les différentes couches. Pour ces éléments, il est nécessaire de préparer le calcul de la température sur les couches en amont de la commande AFFE_MATERIAU. Pour cela, l'utilisateur doit utiliser la commande CREA_RESU avec l'une des opérations PREP_VRC1 ou PREP_VRC2 ("PREParation des VaRiables de Commande") :

- OPERATION = 'PREP_VRC1' : calcul de la température dans les couches d'une coque en partant d'une température TEMP= f(EPAIS, INST)
- OPERATION = 'PREP_VRC2' : calcul de la température dans les couches d'une coque en partant d'une température calculée par aster avec un modèle de coques (TEMP/TEMP_INF/TEMP_SUP).

7.1.1 Opérande CHAM_GD

- ♦ CHAM_GD = chgd

chgd est une carte de fonctions du temps et de l'épaisseur.

7.1.2 Opérande EVOL_THER

- ♦ EVOL_THER = evol

evo est une structure de données EVOL_THER de type « coque », c'est à dire contenant les composantes TEMP/TEMP_INF/TEMP_SUP.

8 Opérands associés à l'assemblage de SD résultat

8.1 Mot clé ASSE

Permet d'assembler plusieurs structures de données evol_ther en les mettant bout à bout en translatant la valeur du paramètre temps.

8.1.1 Opérande RESULTAT

- ♦ RESULTAT = resu

resu est une structure de données evol_ther.

8.1.2 Opérande TRANSLATION

◇ TRANSLATION = / tr, [R]
/ ~~~ [DEFAULT]

tr est la valeur réelle qui sera ajoutée à la valeur de l'attribut INST pour chaque champ de la structure de données resu avant insertion dans la structure de données résultat.

9 Exemple d'utilisation

Construction d'un transitoire thermique à partir d'une fonction :

On a défini ci-dessous les principales commandes utilisées pour construire un concept resultat de type evol_ther.

Définition d'une liste d'instants.

```
lr8=  DEFI_LIST_REEL  (  DEBUT = 0.E0,
                        INTERVALLE=( _F(JUSQU_A=5.e-3,NOMBRE=10 ),
                                       _F(JUSQU_A=5.e-2,NOMBRE= 9 ),
                                       _F(JUSQU_A=4.e-0,NOMBRE=79 ),
                                       _F(JUSQU_A=6.e-0,NOMBRE=20 ),)
                        )
```

Définition d'une fonction du paramètre 'INST'.

```
fct1 = DEFI_FONCTION  (  NOM_PARA = 'INST'
                        VALE= (    0.0,      20.0,
                                0.5,      25.0,
                                2.0,      54.0,
                                10.0,     134.0,)
                        PROL_DROIT = 'LINEAIRE',
                        PROL_GAUCHE = 'LINEAIRE',
                        )
```

Construction d'un champ au nœuds de fonction, on affecte la même fonction fct1 à l'ensemble des nœuds du maillage.

```
ch = CREA_CHAMP (    TYPE_CHAM='NOEU_TEMP_F', OPERATION= 'AFFE',
                    MAILLAGE=ma ,
                    AFFE=_F(TOUT='OUI', NOM_CMP='TEMP', VALE_F=fct1,),
                    )
```

...

Création du concept résultat TEMPE, construit à partir du champ aux nœuds de fonction ch. On se limite au numéro d'ordre 20 correspondant à la valeur 0.1. La structure de données comportera 20 numéros d'ordre de 1 à 20.

```
TEMPE = CREA_RESU  (  OPERATION = 'AFFE',
                      TYPE_RESU = 'EVOL_THER',  NOM_CHAM = 'TEMP',
                      CHAM_GD   = ( _F( CHAM_NO   = ch ,
                                         LIST_INST = lr8,
                                         NUME_FIN  = 20 ,  ),
                      )
...
FIN()
```