

**Manuel d'Utilisation**  
**Fascicule U4.7- : Opérations sur les résultats et les champs**  
**Document : U4.72.04**

## Opérateur CREA\_CHAMP

---

### 1 But

---

Créer un champ de type `cham_no`, `carte` ou `cham_elem`. On peut créer un champ :

- par affectation de valeurs sur des nœuds ou des mailles,
- en assemblant (et / ou combinant) des morceaux de champs existants,
- en évaluant les fonctions d'un champ de fonctions pour en faire un champ de réels,
- en modifiant la représentation géométrique d'un champ (passage nœuds <-> points de Gauss par exemple),
- en extrayant un champ d'une SD de type `SD_RESULTAT` (`evol_ther`, `evol_noli`, `mode_meca`, ...).

On peut aussi se servir de cette commande pour combiner linéairement plusieurs champs (fonctionnalité également possible avec les commandes `COMB_CHAM_NO` et `COMB_CHAM_ELEM`), mais on peut faire des combinaisons plus "compliquées" : multiplication de champs ...

Produit une SD de type `cham_no_*` ou `carte_*` ou `cham_elem_*`

## 2 Syntaxe

```

ch2 [*] = CREA_CHAMP (

    ♦ TYPE_CHAM =          / 'NOEU_xxx' ,
                           / 'CART_xxx' ,
                           / 'ELNO_xxx' ,
                           / 'ELGA_xxx' ,
                           / 'ELEM_xxx' ,

    # pour imposer la numérotation des ddls du CHAM_NO résultat :
    ◇ /  NUME_DDL = nu ,                      [nume_ddl]
      /  CHAM_NO = cnonu ,                    [cham_no_*]

    ◇ TITRE = titre ,                        [l_Kn]
    ◇ INFO  = /1 ,                          [DEFAULT]
              /2 ,

    / OPERATION = 'AFFE' ,
    # =====

    ♦ /  MAILLAGE = ma ,                      [maillage]

      /  MODELE   = mo ,                      [modele]
      ◇ OPTION = option ,                    [Kn]
      ◇ PROL_ZERO = / 'NON' ,                [DEFAULT]
                  / 'OUI' ,

    ♦ AFFE= (_F ( ♦ / TOUT = 'OUI' ,
                  /  GROUP_MA = grma ,      [l_grma]
                  /  MAILLE  = maille ,     [l_maille]
                  /  GROUP_NO = grno ,      [l_grno]
                  /  NOEUD   = noeud ,      [l_noeud]

                  ♦ NOM_CMP  = lcmp ,        [l_K8]
                  ♦ /  VALE   = lvaler ,     [l_R]
                  /  VALE_I = lvalei ,       [l_I]
                  /  VALE_C = lvalec ,       [l_C]
                  /  VALE_F = lvalef ,       [l_fonction]
                  /                                     [l_formule]

                  ),),

```

Titre :       Opérateur CREA\_CHAMP  
Auteur(s) :   J. PELLET

Date :       31/01/06  
Clé :   U4.72.04-D1 Page :   3/16

```

/  OPERATION =                'ASSE' ,
#  =====

♦  /  MAILLAGE =                ma ,                [maillage]
    /  MODELE  =                mo ,                [modele]
    ◊  OPTION  = option                [Kn]
    ◊  PROL_ZERO =                /  'NON' ,                [DEFAULT]
                                   /  'OUI' ,

♦  ASSE= (_F ( ♦  /  TOUT      = 'OUI' ,
                /  |  GROUP_MA = grma ,                [l_grma]
                |  MAILLE    = maille ,                [l_maille]
                |  GROUP_NO  = grno ,                [l_grno]
                |  NOEUD     = noeud ,                [l_noeud]

                ♦  CHAM_GD    = chl ,                [champ_*]

                ◊  ♦  NOM_CMP    = lcmp ,                [l_K8]
                ◊  NOM_CMP_RESU = lcmp_resu,                [l_K8]

                ◊  /  CUMUL    =                'NON' ,                [DEFAULT]
                /  CUMUL    =                'OUI' ,
                ◊  COEF_R    = /                1. ,                [DEFAULT]
                                   /                coef ,                [R]

                ),),

/  OPERATION =                'EVAL' ,
#  =====

♦  CHAMP_F    =  ch_fonc ,                /                [cham_no_NEUT_F]
                                   /                [cham_elem_NEUT_F]

♦  CHAMP_PARA=  l_ch_para ,                [l_champ_*]

/  OPERATION =                'DISC' ,
#  =====

#  si le champ résultat est un CHAM_ELEM :
◊  MODELE    =                mo ,                [modele]
◊  OPTION    = option                [Kn]
◊  PROL_ZERO =                /  'NON' ,                [DEFAULT]
                                   /  'OUI' ,

♦  CHAM_GD    =                chl ,                [champ_*]

/  OPERATION =                'EXTR' ,
#  =====

#  extraction du champ de géométrie d'un maillage :
/  ♦  MAILLAGE =  ma ,                [maillage]
    ♦  NOM_CHAM =  'GEOMETRIE' ,

#  extraction dans une table :
/  ♦  TABLE    =  tabl ,                [table]

#  extraction des "level set" d'une SD fin_xfem :
/  ♦  FISSURE    = fxfem ,                [fin_xfem]
    ♦  NOM_CHAM  = /  'LTNO' ,
                                   /  'LNNO' ,
                                   /  'GRLTNO' ,
                                   /  'GRLNNO' ,

```

Titre :        *Opérateur CREA\_CHAMP*  
Auteur(s) :   **J. PELLET**

Date :        31/01/06  
Clé :    U4.72.04-D1    Page :    4/16

```
#  extraction d'un champ d'une SD_RESULTAT :
/      ♦  RESULTAT =   resu ,
      ♦  NOM_CHAM =   /   'ACCE' ,
                      /   ... (cf. [§7.1]),

♦  SENSIBILITE =      /   theta,                [theta_geom]
                      /   para ,                [para_sensi]

♦  /  #  Sélection d'un numéro d'ordre dans la SD_RESULTAT
      /  NUME_ORDRE   =   nuordr ,                [I]
      /  NUME_MODE    =   numode ,                [I]
      /  NOEUD_CMP    =   (noeud,cmp) ,          [l_K8]
      /  NOM_CAS      =   nocas ,                [Kn]
      /  ANGL         =   α ,                    [R]
      /  ♦  /  INST    =   inst ,                [R]
      /      /  FREQ    =   freq ,                [R]
      ♦  |  PRECISION =   /  prec ,                [R]
                      /  1.D-3,                [DEFAULT]
      |  CRITERE     =   /  'RELATIF' , [DEFAULT]
                      /  'ABSOLU' ,
      |  INTERPOL    =   /  'NON' , [DEFAULT]
                      /  'LIN' ,

/  #  Calcul d'un champ contenant les "extrema" d'une
SD_RESULTAT

♦  TYPE_MAXI =      /   'MAXI' ,
                  /   'MINI' ,
                  /   'MAXI_ABS' ,
                  /   'MINI_ABS' ,
                  /   'NORM_TRAN' ,

♦  TYPE_RESU =      /   'VALE' ,                [DEFAULT]
                  /   'INST' ,

♦  /  TOUT_ORDRE = 'OUI' ,                [DEFAULT]
      /  LIST_INST = linst ,                [listr8]
      /  LIST_FREQ = lfreq ,                [listr8]
      ♦  |  PRECISION = /  prec ,                [R]
                      /  1.0E-3,                [DEFAULT]
      |  CRITERE     = /  'ABSOLU' ,
                      /  'RELATIF' , [DEFAULT]

);

Si      TYPE_CHAM    =      'NOEU_TEMP_R'      alors      [*] =  CHAM_NO_TEMP_R
                        'NOEU_DEPL_R'          CHAM_NO_DEPL_R
                        ...
                        'CART_TEMP_R'          CARTE_TEMP_R
                        'CART_DEPL_R'          CARTE_DEPL_R
                        ...
                        'ELNO_DEPL_R'          CHAM_ELEM_DEPL_R
                        ...
                        'ELGA_DEPL_R'          CHAM_ELEM_DEPL_R
                        ...
```

## 3 Opérandes

### 3.1 Opérandes généraux

#### 3.1.1 Opérande TYPE\_CHAM, affectation d'un type au champ résultat

Ce mot clé (obligatoire) sert d'abord à typer (au sens du superviseur d'Aster) le champ résultat de la commande. Il est formé de 2 "mots" reliés par un "blanc souligné" (  ) :

TYPE\_CHAM= 'XXXX\_GD' où :

XXXX=	/	'NOEU'	Champ aux nœuds	(CHAM_NO_*)
	/	'CART'	Champ constant par maille	(CARTE_*)
	/	'ELNO'	Champ par éléments aux nœuds	(CHAM_ELEM_*)
	/	'ELGA'	Champ par éléments aux points de Gauss	(CHAM_ELEM_*)
	/	'ELEM'	Champ constant par élément	(CHAM_ELEM_*)
GD=	/	'DEPL_R'	déplacement	
	/	'SIEF_R'	contrainte	
	/	'TEMP_R'	température	
	/	'FLUX_R'	flux	
	/	...		

Le type du champ résultat est entièrement déduit de cette information donnée par l'utilisateur. Par exemple :

```
TYPE_CHAM= 'NOEU_DEPL_R'      ->  CHAM_NO_DEPL_R
TYPE_CHAM= 'CART_SIEF_R'      ->  CARTE_SIEF_R
TYPE_CHAM= 'ELNO_EPSI_R'      ->  CHAM_ELEM_EPSI_R
TYPE_CHAM= 'ELGA_VARI_R'      ->  CHAM_ELEM_VARI_R
```

Ce mot clé sert aussi à préciser (pour la commande) quelle doit être la nature du champ voulu en résultat. Il est indispensable pour les opérations 'AFFE', 'ASSE' et 'DISC'.

Exemples :

```
OPERATION= 'AFFE' + TYPE_CHAM= 'CART_DEPL_R'  => une CARTE de DEPL_R.
OPERATION= 'ASSE' + TYPE_CHAM= 'NOEU_EPSI_R'  => un CHAM_NO de EPSI_R.
OPERATION= 'DISC' + TYPE_CHAM= 'NOEU_SIEF_R'  => un CHAM_NO de SIEF_R.
```

Il n'y a que deux opérations pour lesquelles ce mot clé est une contrainte inutile (mais obligatoire !) pour l'utilisateur (OPERATION='EVAL' et OPERATION='EXTR') car pour ces deux opérations, la nature du champ résultat est imposée par le choix de l'opération.

Le renseignement du mot clé TYPE\_CHAM est (malheureusement) fastidieux pour l'OPERATION='EXTR'. Il se déduit du mot clé NOM\_CHAM. La correspondance est donnée dans [§3.7.1].

#### Remarque importante

*La possibilité de créer des `cham_elem` de n'importe quelle grandeur est conditionnée par l'état de développement (informatique) des types d'éléments finis du modèle. Tout n'est pas encore possible ; par exemple, pour créer un `cham_elem` de `FLUX_R` sur un modèle contenant les éléments `DKT`, il faut que cet élément fini ait prévu de le faire (ce qui n'est pas le cas aujourd'hui).*

*On ne peut donner ici la liste précise des grandeurs permises pour chaque type d'élément fini. On se contentera de dire approximativement :*

- *pour les éléments iso-paramétriques de mécanique, sont permises les grandeurs : `GEOM_R`, `INST_R`, `NEUT_R`, `NEUT_F`, `EPSI_R`, `SIEF_R`, `VARI_R`, `DOMMAG` et `HYDR_R`,*
- *pour les éléments iso-paramétriques de thermique, sont permises les grandeurs : `GEOM_R`, `INST_R`, `NEUT_R` et `NEUT_F`.*

### 3.1.2 Mots clé NUME\_DDL et CHAM\_NO

Ces deux mots clé permettent d'imposer une numérotation pour le champ résultat (si celui-ci est un CHAM\_NO). On donne via ces mots clé un "modèle" de numérotation pour le champ résultat.

Si on donne NUME\_DDL= nu, on prendra comme numérotation celle de nu. Cette possibilité n'est valable que pour les champs de déplacements (phénomène 'MECANIQUE' ) ou pour des champs de température (phénomène 'THERMIQUE' ) ou des champs de pression acoustique (phénomène 'ACOUSTIQUE').

Si on donne CHAM\_NO= chno, on prendra la numération de chno.

#### Remarque sur l'espace disque utilisé :

Ces 2 mots clés permettent parfois d'économiser beaucoup de place sur la base "Globale". Quand par exemple, on extrait de nombreux *cham\_no* d'une *SD\_RESULTAT*, si on n'utilise pas l'un de ces mots clés, on duplique le profil du champ pour chacun d'eux. Si on utilise l'un de ces 2 mots clés, tous ces champs s'appuieront sur le profil contenu dans *chno* (ou *nu*).

### 3.1.3 Opérande TITRE = titr

Titre que l'on veut donner au champ résultat [U4.03.01].

### 3.1.4 Opérande INFO = /1 /2

INFO = 1  
Aucune impression.

INFO = 2  
Impression sur le fichier 'MESSAGE' du champ résultat.

### 3.2 Opérande OPERATION = /'AFFE' /'ASSE' /'EVAL' /'DISC' /'EXTR'

Cet opérande sert à choisir le "mode" de fabrication du champ résultat. On peut créer un champ :

- par affectation de valeurs sur des nœuds ou des mailles (OPERATION='AFFE'),
- en assemblant des morceaux de champs définis sur des morceaux de maillages (OPERATION='ASSE'),
- en modifiant la représentation géométrique (discrétisation) d'un champ (passage nœuds <-> points de Gauss par exemple) (OPERATION='DISC'),
- en extrayant un champ d'une SD de type SD\_RESULTAT (evol\_ther, evol\_noli, mode\_meca, ...) (OPERATION='EXTR').
- en combinant linéairement des champs (OPERATION='ASSE'),
- en "combinant" (multiplication, exponentielle, ...) des champs (OPERATION='EVAL'),
- en évaluant les fonctions d'un champ de fonctions pour en faire un champ de réels (OPERATION='EVAL'),

### 3.3 Opérandes pour OPERATION = 'AFFE'

Cette opération permet d'affecter des valeurs (réelles, entières, complexes ou fonction) sur des entités géométriques (nœuds ou mailles) d'un maillage.

La grandeur associée au champ est implicitement donnée par le mot clé TYPE\_CHAM (ci-dessus).

### 3.3.1 Opérande MAILLAGE = mailla

Lorsque l'on crée un CHAM\_NO ou une CARTE, il faut préciser sur quel maillage s'appuiera ce champ. Pour cela, on utilise le mot clé MAILLAGE.

**3.3.2 Opérandes MODELE = modele OPTION = option PROL\_ZERO = /'OUI' /'NON'**

Lorsque l'on crée un CHAM\_ELEM, il faut préciser sur quels éléments finis sera défini le champ. Pour cela, on utilise le mot clé MODELE.

Pour décrire la structure d'un CHAM\_ELEM, il ne suffit pas de donner (via le MODELE) un type d'élément pour chaque maille, car un type d'élément peut connaître plusieurs "formes" de CHAM\_ELEM pour une grandeur donnée. Pour créer le champ souhaité, l'utilisateur peut utiliser le mot clé OPTION. Si par exemple, il écrit : `X= CREA_CHAMP(... MODELE=mo, OPTION='SIEF_ELGA_DEPL', ...)`, le champ créé par CREA\_CHAMP aura la même forme que s'il avait été calculé par CALC\_ELEM / `OPTION='SIEF_ELGA_DEPL'`. Si l'utilisateur n'emploie pas le mot clé OPTION, CREA\_CHAMP choisira (s'il le peut) une forme par défaut.

Dans un CHAM\_ELEM, tous les éléments d'un même type (même type de maille et même modélisation) doivent obligatoirement posséder les mêmes composantes. Si le champ résultat est incomplètement décrit (si l'on n'a pas affecté toutes les mailles), il faut autoriser (éventuellement) la commande CREA\_CHAMP à "prolonger" le champ par "zéro" là où il n'a pas été décrit. C'est l'objet du mot clé PROL\_ZERO.

Si PROL\_ZERO = 'NON' (défaut)

Si des éléments sont incomplets, on s'arrête en erreur fatale.

Si PROL\_ZERO = 'OUI'

On prolonge le champ incomplètement décrit par des "0."

**3.3.3 Mot clé facteur AFFE**

Les autres opérandes sont regroupés sous le mot clé facteur AFFE. Ce mot clé est répétable. Le principe de surcharge est appliqué entre les différentes occurrences du mot clé AFFE : si une entité géométrique est affectée plusieurs fois, la dernière affectation l'emporte.

**3.3.4 Opérandes TOUT= 'OUI', GROUP\_MA, GROUP\_NO, MAILLE et NOEUD**

Les entités géométriques que l'on veut affecter sont données par les opérandes TOUT='OUI', GROUP\_MA, GROUP\_NO, MAILLE et NOEUD.

Si TYPE\_CHAM='NOEU\_xxx', on affecte des nœuds ; l'utilisation des opérandes GROUP\_MA et MAILLE est possible et signifie que l'on affecte tous les nœuds des mailles spécifiées.

Si TYPE\_CHAM:'EL..\_xxx' (ou 'CART\_xxx'), on affecte des mailles ; l'utilisation des opérandes GROUP\_NO et NOEUD est alors interdite.

**3.3.5 Opérande NOM\_CMP**

Les noms des composantes que l'on veut affecter sont donnés par l'opérande NOM\_CMP.

**3.3.6 Opérandes VALE, VALE\_I, VALE\_C ou VALE\_F**

Les valeurs à affecter sont données par les opérandes VALE, VALE\_I, VALE\_C ou VALE\_F selon la nature (réel, entier, complexe, fonction (ou formule) ) des composantes de la grandeur (DEPL\_R : réel, DEPL\_C : complexe, TEMP\_F : fonction / formule , ...).

**3.3.7 Exemples**

Création d'un champ aux nœuds de déplacement. On veut imposer la numérotation du champ (celle de cnomod) :

```
DEPL1 = CREA_CHAMP( OPERATION= 'AFFE',
                    TYPE_CHAM='NOEU_DEPL_R' , MAILLAGE = MA , CHAM_NO= CNOMOD,
                    AFFE= ( _F(TOUT='OUI' , NOM_CMP=('DX','DY','DZ') , VALE=(0.0.0.)),
                          _F ( GROUP_MA=(GM1,GM2) , NOM_CMP= 'DX' , VALE= 3.5d-2),
                          _F ( NOEUD=(n5,n7,n9) , NOM_CMP= 'DY' , VALE= 1.6d-2),
                    )
)
```

Création d'une carte de température (fonctions) :

```
TEMPF = CREA_CHAMP( OPERATION= 'AFFE',  
                    TYPE_CHAM= 'CART_TEMP_F' , MAILLAGE = MA,  
                    AFFE=( _F( TOUT='OUI' , NOM_CMP=( 'TEMP' ), VALE_F= F1 ),  
                           _F( GROUP_MA=( GM1, GM2 ), NOM_CMP=( 'TEMP' ), VALE_F= F2 ), )  
)
```

## 3.4 Opérandes pour OPERATION = 'ASSE'

### 3.4.1 Généralités

Cet opérateur "assemble" des "morceaux de champs" pour en fabriquer un nouveau. Chaque occurrence du mot clé ASSE définit un morceau de champ. On appelle un morceau de champ, la restriction d'un champ existant (carte / cham\_no ou cham\_elem) sur un ensemble d'entités géométriques (mailles ou nœuds) et sur un ensemble de composantes.

Il y a un principe de surcharge des occurrences du mot clé ASSE si les morceaux se recouvrent les uns les autres.

Actuellement, on peut fabriquer :

- un cham\_no en assemblant des morceaux de cham\_no.
- un cham\_elem en assemblant des morceaux de cham\_elem et/ou de cartes.

L'opération 'ASSE' permet également de changer la grandeur associée à un champ ; par exemple transformer un champ de déformation (EPSI\_R) en champ de contraintes (SIEF\_R). Pour cela il faut utiliser les mots clé NOM\_CMP et NOM\_CMP\_RESU.

L'assemblage des morceaux de champs peut se faire en cumulant les morceaux (mots clé CUMUL et COEF\_R). Cela permet d'utiliser cette commande pour faire des combinaisons linéaires de CHAM\_NO ou de CHAM\_ELEM.

### 3.4.2 Opérandes MAILLAGE, MODELE et PROL\_ZERO

Même usage que pour OPERATION= 'AFFE' [§3.3.1] et [§3.3.2].

### 3.4.3 Opérandes pour le mot clé facteur ASSE

Chaque occurrence du mot clé facteur ASSE permet de définir un morceau de champ que l'on assemble dans le champ résultat.

#### 3.4.3.1 Opérande CHAM\_GD = ch1

ch1 est le champ (existant) avec lequel on veut fabriquer un morceau de champ.

#### 3.4.3.2 Opérandes TOUT='OUI' , GROUP\_MA, GROUP\_NO, MAILLE et NOEUD

Ces opérandes servent à définir la restriction géométrique du champ ch1. Si ch1 est un CHAM\_NO, on peut utiliser tous ces opérandes. Si ch1 est un CHAM\_ELEM (ou une CARTE), on ne peut pas utiliser les opérandes GROUP\_NO et NOEUD.

#### 3.4.3.3 Opérandes NOM\_CMP et NOM\_CMP\_RESU

L'opérande NOM\_CMP sert à définir les composantes sur lesquelles on veut restreindre le champ ch1. Si NOM\_CMP est absent, on prend toutes les composantes de ch1.

L'opérande NOM\_CMP\_RESU sert à renommer (si on le souhaite) les composantes de ch1. Si NOM\_CMP\_RESU est fourni, NOM\_CMP doit l'être aussi et les deux listes en correspondance doivent être de même longueur.

Exemple 1 : transformer un champ de EPSI\_R en champ de VARI\_R

```
CHVARI=CREA_CHAMP( OPERATION='ASSE', TYPE_CHAM='ELGA_VARI_R',  
                  MODELE=MO,  
                  ASSE=_F( CHAM=CHEPSI , TOUT='OUI',  
                           NOM_CMP = ('EPXX', 'EPYY'),  
                           NOM_CMP_RESU= ('V3', 'V1'), ) )
```



Titre :        *Opérateur CREA\_CHAMP*  
 Auteur(s) :   *J. PELLET*

Date :        31/01/06  
 Clé :        U4.72.04-D1 Page :    9/16

Exemple 2 : permuter les cmps SIXX et SIYY d'un champ de SIEF\_R  
 CHS2=CREA\_CHAMP(OPERATION='ASSE', TYPE='NOEU\_SIEF\_R',  
   MAILLAGE= MA ,  
   ASSE=\_F(CHAMP=CHS1, TOUT='OUI',  
     NOM\_CMP =('SIXX','SIYY'),  
     NOM\_CMP\_RESU=('SIYY','SIXX',) , ))

#### 3.4.3.4 Opérandes CUMUL et COEF\_R

L'opérande CUMUL='OUI' veut dire que les valeurs de l'occurrence concernée seront additionnées aux éventuelles valeurs déjà existantes.

Si CUMUL='NON', la valeur affectée remplace la valeur éventuellement déjà présente (CUMUL='OUI' est invalide pour les champs de "texte" (k8/k16, ...) bien entendu).

L'opérande COEF\_R = *coefr* permet la multiplication du morceau de champ par le coefficient réel *coefr* avant de l'assembler au champ résultat.

Exemple :

Fabriquer le cham\_elem :  $ch3 = 2. * ch1 + 3. * ch2$

```
CH3= CREA_CHAMP( OPERATION= 'ASSE',
  MODELE = MO , TYPE_CHAM = 'ELGA_EPSI_R',
  ASSE = (_F( CHAM_GD = CH1 , TOUT = 'OUI',
    CUMUL='OUI', COEF_R = 2.),
    _F( CHAM_GD = CH2 , TOUT = 'OUI',
      CUMUL='OUI', COEF_R = 3.)),
)
```

### 3.4.4 Exemples

Exemple 1

Fabriquer un cham\_no de température en extrayant un champ déjà calculé (dans un evol\_ther) et en le redéfinissant (à 25. degrés) sur le groupe de mailles soudur1.

```
CH1= CREA_CHAMP ( OPERATION= 'EXTR' ,TYPE_CHAM='NOEU_TEMP_R',
  RESULTAT= EVOTH , NOM_CHAM= 'TEMP', INST = 12.)
CH2= CREA_CHAMP (OPERATION = 'AFFE', TYPE_CHAM='NOEU_TEMP_R',
  MAILLAGE =MA,
  AFFE=_F( TOUT = 'OUI' ,NOM_CMP = 'TEMP',VALE = 25.))

CH3= CREA_CHAMP(OPERATION = 'ASSE',
  MAILLAGE = MA,TYPE_CHAM = 'NOEU_TEMP_R',
  ASSE = (_F( CHAM_GD = CH1 , TOUT = 'OUI',),
    _F( CHAM_GD = CH2 ,GROUP_MA = SOUDUR1 )),
)
```

Exemple 2 :

Fabriquer un cham\_elem de VARI\_R (pour l'utiliser en tant qu'état initial pour stat\_non\_line) en récupérant les variables internes (6 et 8) d'une loi de comportement pour en faire les variables 1 et 2 de la (nouvelle) loi de comportement qui sera utilisée dans le stat\_non\_line à venir.

```
CH1= CREA_CHAMP ( OPERATION= 'EXTR', TYPE_CHAM='ELGA_VARI_R',
  RESULTAT= STNL, NOM_CHAM= 'VARI_ELGA', INST = 4.)

CH2= CREA_CHAMP( OPERATION= 'ASSE',
  MODELE = MO , TYPE_CHAM = 'ELGA_VARI_R',
  ASSE = _F( CHAM_GD = CH1 , TOUT = 'OUI',
    NOM_CMP = ( 'V6', 'V8'),
    NOM_CMP_RESU = ( 'V1', 'V2'), ))
```

### 3.5 Opérandes pour OPERATION = 'EVAL' ,

Cette opération sert à transformer un champ de fonctions en champs de réels en évaluant les fonctions du champ de fonctions.

Le champ de fonctions est obligatoirement un champ de la grandeur 'NEUT\_F' et le champ résultat sera toujours un champ de 'NEUT\_R'. Ce champ pourra être transformé en champ d'une autre grandeur quelconque en faisant appel une seconde fois à la commande CREA\_CHAMP / OPERATION='ASSE'.

Un exemple de l'usage de l'opération 'EVAL' est donné dans le document [U2.01.09] "Définition analytique d'un champ de contraintes ..."

#### 3.5.1 Opérande CHAM\_F= chf

chf est le nom du champ de fonctions à évaluer (CHAM\_NO, CARTE ou CHAM\_ELEM).

#### 3.5.2 Opérande CHAM\_PARA= l\_chpara

l\_chpara est la liste des champs "paramètres" pour l'évaluation des fonctions. Tous les champs de l\_chpara doivent être discrétisés de la même façon que chf. Par exemple, si chf est un CHAM\_ELEM/ELGA, il faut que tous les champs de l\_chpara soient aussi des CHAM\_ELEM/ELGA.

Il faut que la liste des champs paramètres soit suffisante pour permettre l'évaluation de toutes les fonctions référencées dans chf.

#### 3.5.3 Exemples

##### 3.5.3.1 Exemple 1

On veut créer un cham\_elem (SIEF\_R) aux points de Gauss dont les composantes soient des fonctions analytiques de la géométrie et du temps. Pour cet exemple, on suppose que l'on a déjà fabriqué deux champs aux points de Gauss CHGEOMG : champ de géométrie et CHINSTG : champ d'instant.

```

RHO=1000.
G=10.
KP=3.
SIZZ = FORMULE (NOM_PARA = 'Z', VALE = 'RHO*G*Z')
SIXX = FORMULE (NOM_PARA = ('Z','INST'), VALE = 'KP*SIZZ(Z)+3.*INST')

# Affectation des fonctions :
# -----
SIG1=CREA_CHAMP(OPERATION='AFFE', TYPE_CHAM='ELGA_NEUT_F',
  MODELE=MO , PROL_ZERO='OUI',
  AFFE=_F(TOUT='OUI', NOM_CMP=('X1','X2'),
    VALE_F=('SIXX','SIZZ') ))

# Evaluation des fonctions :
# -----
SIG2= CREA_CHAMP(OPERATION='EVAL',TYPE_CHAM='ELGA_NEUT_R',
  MODELE=MO ,CHAM_F=SIG1 , CHAM_PARA=( CHGEOMG,CHINSTG) )

# transformation du champ de NEUT_R en SIEF_R :
# -----
SIG3=CREA_CHAMP(OPERATION='ASSE', TYPE_CHAM='ELGA_SIEF_R',
  MODELE=MO , PROL_ZERO='OUI',
  ASSE=_F(TOUT='OUI', CHAM_GD=SIG2,
    NOM_CMP= ('X1','X2') ,
    NOM_CMP_RESU=('SIXX','SIZZ') ,
  ))

```

### 3.5.3.2 Exemple 2

On veut calculer un champ de température aux nœuds (CH3) contenant le produit de 2 autres champs aux nœuds de température (CH1 et CH2)

```
# 1) transformation des cham_no_TEMP_R (CH1 et CH2) en cham_no_NEUT_R:
# -----
CH1N=CREA_CHAMP( OPERATION='ASSE', TYPE_CHAM='NOEU_NEUT_R', MAILLAGE=MA,
                 ASSE=_F( TOUT = 'OUI', CHAM_GD = CH1,
                 NOM_CMP = ('TEMP',), NOM_CMP_RESU = ('X1',)), )

CH2N=CREA_CHAMP( OPERATION='ASSE', TYPE_CHAM='NOEU_NEUT_R', MAILLAGE=MA,
                 ASSE=_F( TOUT = 'OUI', CHAM_GD = CH2,
                 NOM_CMP = ('TEMP',), NOM_CMP_RESU = ('X2',)), )

# 2) multiplication    CH3N = CH1N * CH2N :
# -----
FMULT = FORMULE(NOM_PARA = ('X1', 'X2'), VALE = 'X1*X2')

CHFMU=CREA_CHAMP( OPERATION='AFFE', TYPE_CHAM='NOEU_NEUT_F', MAILLAGE=MA,
                 AFFE=_F( TOUT = 'OUI', NOM_CMP = 'X3', VALE_F = FMULT) )

CH3N=CREA_CHAMP( OPERATION='EVAL', TYPE_CHAM='NOEU_NEUT_R',
                 CHAM_F=CHFMU, CHAM_PARA=(CH1N,CH2N,) )

# 3) transformation du cham_no_NEUT_R (CH3N) en cham_no_TEMP_R:
# -----
CH3=CREA_CHAMP( OPERATION='ASSE', TYPE_CHAM='NOEU_TEMP_R', MAILLAGE=MA,
                 ASSE=_F( TOUT = 'OUI', CHAM_GD = CH3N,
                 NOM_CMP = ('X3',), NOM_CMP_RESU = ('TEMP',)), )
```

### 3.6 Opérandes pour OPERATION = 'DISC'

Cette opération sert à modifier la "discrétisation" d'un champ existant. Par exemple, transformer un champ aux nœuds en champs aux points de Gauss. Les deux champs (donnée et résultat) sont évidemment associés à la même grandeur.

La discrétisation voulue par l'utilisateur pour son champ résultat est indiquée par le mot clé TYPE\_CHAM.

Seuls 7 cas de figure sont traités actuellement par la commande :

- |                               |                     |
|-------------------------------|---------------------|
| 1) passage : CARTE            | -> CHAM_ELEM / ELNO |
| 2) passage : CARTE            | -> CHAM_ELEM / ELGA |
| 3) passage : CARTE            | -> CHAM_ELEM / ELEM |
| 4) passage : CARTE            | -> CHAM_NO          |
| 5) passage : CHAM_NO          | -> CHAM_ELEM / ELNO |
| 6) passage : CHAM_NO          | -> CHAM_ELEM / ELGA |
| 7) passage : CHAM_ELEM / ELNO | -> CHAM_NO          |

Pour les cas de figure 1, 2 et 3, on duplique la valeur affectée à la maille (CARTE) sur tous les points de Gauss (ou les nœuds) de cette maille.

Pour le cas de figure 5, le passage des valeurs des nœuds aux points de Gauss se fait en utilisant les fonctions de forme des éléments finis du modèle.

Pour le cas de figure 6, le passage aux valeurs des nœuds du maillage se fait par moyenne arithmétique des valeurs portées par les nœuds des éléments concourants.

#### 3.6.1 Opérande CHAM\_GD= ch1

ch1 est le champ dont on veut modifier la "discrétisation".

#### 3.6.2 mots clé MODELE et PROL\_ZERO

Même usage que pour OPERATION = 'AFFE' [3.3.2]

#### 3.6.3 Exemple

```
% CHXG = CHAMP DE GEOMETRIE AUX POINTS DE GAUSS :  
% -----  
CHXN =CREA_CHAMP(OPERATION='EXTR', TYPE_CHAM='NOEU_GEOM_R',  
                  NOM_CHAM='GEOMETRIE', MAILLAGE=MA )  
CHXG= CREA_CHAMP(OPERATION='DISC', TYPE_CHAM='ELGA_GEOM_R',  
                  MODELE=MO , CHAM_GD= CHXN )
```

### 3.7 Opérandes pour OPERATION = 'EXTR'

Cette opération sert en général à extraire un champ d'une SD de type `resultat_sdaster`.

Il y a 3 exceptions :

- on peut extraire le champ de géométrie des nœuds d'un maillage. Il faut alors utiliser les mots clés : `MAILLAGE = ma`, `NOM_CHAM = 'GEOMETRIE'` et `TYPE_CHAM = 'NOEU_GEOM_R'`.
- on peut créer un champ en extrayant d'une table les valeurs correspondant à des paramètres de noms pré-établis : `MAILLE`, `NOEUD`, `POINT`, `SOUS_POINT`, noms des composantes.
- on peut extraire les "level set" associées à une fissure XFEM. Il faut alors utiliser les mots clés :  
`FISSURE = fiss_xfem`, `NOM_CHAM =       / 'LTNO'       / 'LNNO'`  
`/ 'GRLTNO'   / 'GRLNNO'`  
et `TYPE_CHAM = 'NOEU_NEUT_R'`  
  
`'LTNO'       level set tangente`  
`'LNNO'       level set normale`  
`'GRLTNO'     gradient de la level set tangente`  
`'GRLNNO'     gradient de la level set normale`

#### 3.7.1 Typage du champ résultat, mot clé TYPE\_CHAM

Le mot clé `TYPE_CHAM` (obligatoire) [§3.2] doit être renseigné. Mis à part le cas de l'extraction dans une SD de type `DYNA_HARMO`, le mot clé `TYPE_CHAM` se déduit du nom symbolique du champ extrait (`NOM_CHAM`). Le tableau ci-dessous donne la correspondance entre ces deux mots clés.

NOM_CHAM	TYPE_CHAM	NOM_CHAM	TYPE_CHAM
'ACCE'	'NOEU_DEPL_R'	'INTE_ELNO_ACTI'	'ELNO_INTE_R'
'ACCE_ABSOLU'	'NOEU_DEPL_R'	'INTE_ELNO_REAC'	'ELNO_INTE_R'
'DCHA_ELGA_SIGM'	'ELGA_VARI_R'	'INTE_NOEU_ACTI'	'NOEU_INTE_R'
'DCHA_ELNO_SIGM'	'ELNO_VARI_R'	'INTE_NOEU_REAC'	'NOEU_INTE_R'
'DEGE_ELNO_DEPL'	'ELNO_EPSI_R'	'IRRA'	'NOEU_IRRA_R'
'DEPL'	'NOEU_DEPL_R'	'META_ELGA_TEMP'	'ELGA_VARI_R'
'DEPL_ABSOLU'	'NOEU_DEPL_R'	'META_ELNO_TEMP'	'ELNO_VARI_R'
'ECIN_ELEM_DEPL'	'ELEM_ENER_R'	'META_NOEU_TEMP'	'NOEU_VAR2_R'
'EFGE_ELNO_CART'	'ELNO_SIEF_R'	'PRES_ELNO_DBEL'	'ELNO_DBEL_R'
'EFGE_ELNO_DEPL'	'ELNO_SIEF_R'	'PRES_ELNO_IMAG'	'ELNO_PRES_R'
'EFGE_NOEU_CART'	'NOEU_SIEF_R'	'PRES_ELNO_REEL'	'ELNO_PRES_R'
'EFGE_NOEU_DEPL'	'NOEU_SIEF_R'	'PRES_NOEU_DBEL'	'NOEU_DBEL_R'
'ENDO_ELNO_SIGA'	'ELNO_SIEF_R'	'PRES_NOEU_IMAG'	'NOEU_PRES_R'
'ENDO_ELNO_SIGM'	'ELNO_SIEF_R'	'PRES_NOEU_REEL'	'NOEU_PRES_R'
'ENDO_ELNO_SINO'	'ELNO_SIEF_R'	'RADI_ELGA_SIGM'	'ELGA_VARI_R'
'EPME_ELGA_DEPL'	'ELGA_EPSI_R'	'RADI_ELNO_SIGM'	'ELNO_VARI_R'
'EPME_ELNO_DEPL'	'ELNO_EPSI_R'	'REAC_NODA'	'NOEU_DEPL_R'
'EPME_ELNO_DPGE'	'ELNO_EPSI_R'	'SIEF_ELGA'	'ELGA_SIEF_R'
'EPMG_ELGA_DEPL'	'ELGA_EPSI_R'	'SIEF_ELGA_DEPL'	'ELGA_SIEF_R'
'EPMG_ELNO_DEPL'	'ELNO_EPSI_R'	'SIEF_ELNO'	'ELNO_SIEF_R'
'EPOT_ELEM_DEPL'	'ELEM_ENER_R'	'SIEF_ELNO_ELGA'	'ELNO_SIEF_R'
'EPSA_ELNO'	'ELNO_EPSI_R'	'SIGM_ELNO_CART'	'ELNO_SIEF_R'
'EPSG_ELGA_DEPL'	'ELGA_EPSI_R'	'SIGM_ELNO_DEPL'	'ELNO_SIEF_R'
'EPSG_ELNO_DEPL'	'ELNO_EPSI_R'	'SIGM_ELNO_SIEF'	'ELNO_SIEF_R'
'EPSI_ELGA_DEPL'	'ELGA_EPSI_R'	'SIGM_ELNO_VARI'	'ELNO_SIEF_R'
'EPSI_ELNO_DEPL'	'ELNO_EPSI_R'	'SIGM_NOEU_CART'	'NOEU_SIEF_R'
'EPSI_NOEU_DEPL'	'NOEU_SIEF_R'	'SIGM_NOEU_DEPL'	'NOEU_SIEF_R'
'EPSP_ELGA'	'ELGA_EPSI_R'	'SIGM_NOEU_SIEF'	'NOEU_SIEF_R'
'EPSP_ELNO'	'ELNO_EPSI_R'	'SIGM_NOEU_VARI'	'NOEU_SIEF_R'
'EQUI_ELGA_EPSI'	'ELGA_SIEF_R'	'SIGM_NOZ1_ELGA'	'NOEU_SIEF_R'
'EQUI_ELGA_SIGM'	'ELGA_SIEF_R'	'SIGM_NOZ2_ELGA'	'NOEU_SIEF_R'
'EQUI_ELNO_EPSI'	'ELNO_SIEF_R'	'SIPO_ELNO_DEPL'	'ELNO_SIEF_R'
'EQUI_ELNO_SIGM'	'ELNO_SIEF_R'	'SIPO_ELNO_SIEF'	'ELNO_SIEF_R'
'EQUI_NOEU_EPSI'	'NOEU_SIEF_R'	'SIPO_NOEU_SIEF'	'NOEU_SIEF_R'
'EQUI_NOEU_SIGM'	'NOEU_SIEF_R'	'SIRE_ELNO_DEPL'	'ELNO_SIEF_R'
'ERRE_ELEM_NOZ1'	'ELEM_ERREUR'	'SOUR_ELGA_ELEC'	'ELGA_SOUR_R'
'ERRE_ELEM_NOZ2'	'ELEM_ERREUR'	'TEMP'	'NOEU_TEMP_R'
'ERRE_ELGA_NORE'	'ELGA_ERREUR'	'THETA'	'NOEU_DEPL_R'

Titre : *Opérateur CREA\_CHAMP*  
Auteur(s) : **J. PELLET**

Date : 31/01/06  
Clé : U4.72.04-D1 Page : 14/16

'ERRE_ELNO_ELGA'	'ELNO_ERREUR'	'VALE_NCOU_MAXI'	'ELEM_SPMA_R'
'FLUX_ELGA_TEMP'	'ELGA_FLUX_R'	'VARI_ELGA'	'ELGA_VARI_R'
'FLUX_ELNO_TEMP'	'ELNO_FLUX_R'	'VARI_ELNO'	'ELNO_VARI_R'
'FLUX_NOEU_TEMP'	'NOEU_FLUX_R'	'VARI_ELNO_ELGA'	'ELNO_VARI_R'
'FORC_NODA'	'NOEU_DEPL_R'	'VARI_NOEU_ELGA'	'NOEU_VAR2_R'
'GEOMETRIE'	'NOEU_GEOM_R'	'VITE'	'NOEU_DEPL_R'
'GRAD_ELGA_THETA'	'ELGA_G_DEPL_R'	'VITE_ABSOLU'	'NOEU_DEPL_R'
'GRAD_ELNO_ELGA'	'ELNO_G_DEPL_R'	'VNOR_ELEM_DEPL'	'ELEM_VNOR_C'
'GRAD_NOEU_THETA'	'NOEU_VAR2_R'		
'HYDR_ELGA'	'ELGA_HYDR_R'		

Pour les DYN\_HARMO, NOM\_CHAM peut prendre trois valeurs : 'DEPL', 'VITE' et 'ACCE'. Dans les trois cas, le type du champ résultat est un CHAM\_NO\_DEPL\_C et il faut donc renseigner : TYPE\_CHAM= 'NOEU\_DEPL\_C'.

### 3.7.2 Opérande TABLE

◆ TABLE = tabl

Nom de concept table contenant les valeurs à stocker dans le champ. Les noms des paramètres de la table doivent respecter certaines règles.

Si le champ à produire est un champ aux nœuds 'NOEU\_XXXX', on doit trouver comme noms des paramètres de la table :

NOEUD + des noms de CMPS parmi les noms des CMPS de la grandeur XXXX.

Le paramètre NOEUD contient le nom du nœud.

Si le champ à produire est un champ aux éléments 'ELNO\_XXXX', on doit trouver comme noms des paramètres de la table :

MAILLE, POINT + des noms de CMPS parmi les noms des CMPS de la grandeur XXXX.

Le paramètre MAILLE contient le nom de la maille.

Le paramètre POINT contient le numéro du nœud dans la connectivité de la maille.

Si le champ à produire est un champ aux nœuds 'ELGA\_XXXX', on doit trouver comme noms des paramètres de la table :

MAILLE, POINT, SOUS\_POINT + des noms de CMPS parmi les noms des CMPS de la grandeur XXXX.

Le paramètre MAILLE contient le nom de la maille.

Le paramètre POINT contient le numéro du point de Gauss dans la maille.

Le paramètre SOUS\_POINT (nécessaire uniquement pour les champs à "sous-points") contient le numéro du sous-point de Gauss dans la point de Gauss.

Si le champ à produire est un champ aux nœuds 'ELEM\_XXXX', on doit trouver comme noms des paramètres de la table :

MAILLE + des noms de CMPS parmi les noms des CMPS de la grandeur XXXX.

### 3.7.3 Opérande RESULTAT

◆ RESULTAT = resu

Nom de concept resultat dans lequel on veut récupérer un champ.

### 3.7.4 Opérande NOM\_CHAM

◆ NOM\_CHAM

Ce mot clé précise le nom symbolique du champ à extraire [U4.71.00].

### 3.7.5 Opérande SENSIBILITE = para

Ce mot clé signifie que l'on ne souhaite pas extraire le véritable champ de la SD\_RESULTAT, mais sa dérivée par rapport au paramètre sensible para.

Le document [U4.50.02] précise le fonctionnement de ce mot clé.

### 3.7.6 Opérandes NUME\_ORDRE / NUME\_MODE / NOM\_MODE / ... / INTERPOL

Ces mots clés servent à préciser quel est le numéro d'ordre de la SD\_RESULTAT que l'on veut extraire.

Le choix des mots clés à utiliser dépend du type de SD\_RESULTAT [U4.71.00].

#### Remarques :

- 1) Lorsque l'on utilise `INTERPOL = 'LIN'`, le champ extrait sera une interpolation entre deux champs de la SD\_RESULTAT. Cette interpolation n'a pas toujours de sens "physique" ; par exemple sur des modes propres.
- 2) Lorsque l'on utilise un accès de type "réel" (`INST` ou `FREQ`), on cherche un champ dans un intervalle donné. Si on trouve plusieurs champs dans l'intervalle, le programme s'arrête en erreur fatale.

### 3.7.7 Calcul d'un champ contenant les "extrema" d'une SD\_RESULTAT

L'idée est de créer un champ contenant en chaque point de l'espace la valeur extrême rencontrée au cours d'un transitoire (ou l'instant auquel cette valeur a été rencontrée).

Aujourd'hui ce paragraphe ne concerne que les résultats de type `evol_ther`, `evol_elas`, `evol_noli` et `dyna_trans`. Les champs sont toujours de type "réel".

On doit préciser :

- les numéros d'ordre définissant le transitoire : `TOUT_ORDRE`, `LIST_INST`,
- le type d'extrémalité choisi : `TYPE_MAXI = 'MAXI'`, ..., `'MINI_ABS'`, `'NORM_TRAN'`,
- ce que l'on veut : la valeur extrême ou l'instant où cette valeur est atteinte (`TYPE_RESU`).

#### 3.7.7.1 Opérande TYPE\_MAXI

- / 'MAXI'                   on considère le maximum atteint par les composantes au cours du transitoire,
  - / 'MAXI\_ABS'             on considère le maximum atteint par la valeur absolue des composantes au cours du transitoire,
  - / 'MINI' / 'MINI\_ABS'   idem pour les minimums,
  - / 'NORM\_TRAN'           on considère le maximum atteint par la quantité :  
 $DX**2 + DY**2 + DZ**2$ .
- pour les 4 valeurs : `'MAXI'`, ..., `'MINI_ABS'`, les composantes du champ sont traitées indépendamment les unes des autres : les valeurs extrêmes peuvent ne pas être atteintes au même instant,
  - la cinquième valeur possible : `'NORM_TRAN'` n'est possible que pour les champs de `depl_R`. En un point donné, on cherche l'instant où la norme du vecteur translation est maximum et on recopie dans le champ résultat toutes les composantes du champ à l'instant trouvé.

#### 3.7.7.2 Opérande TYPE\_RESU

- / 'VALE' ,       le champ résultat contient les valeurs extrêmes rencontrées au cours du transitoire,

#### Remarque :

Même si l'extremum a été obtenu avec une valeur absolue (`'MAXI_ABS'` ou `'MINI_ABS'`), la valeur stockée est algébrique.

- / 'INST' ,       le champ résultat contient les valeurs des instants où les valeurs extrêmes ont été rencontrées.

**Attention :** si par exemple :

```
NOM_CHAM = 'FLUX_ELNO_TEMP',  
TYPE_RESU = 'INST',
```

le champ résultat est un `cham_no_FLUX_R` qui contient des valeurs d'instants !

### 3.7.7.3 Opérandes TOUT\_ORDRE / LIST\_INST / LIST\_FREQ / PRECISION / CRITERE

Ces mots clés permettent de préciser l'étendue du transitoire à examiner.

Si TOUT\_ORDRE = 'OUI' on passe en revue tous les numéros d'ordre.

Si LIST\_INST = linst on ne considère que les instants spécifiés.

## 3.7.8 Exemples

### 3.7.8.1 Extraction d'un champ de température d'un concept resultat de type evol\_ther

```
temp10 = CREA_CHAMP ( OPERATION='EXTR',  
                     NOM_CHAM = 'TEMP' , TYPE_CHAM = 'NOEU_TEMP_R',  
                     RESULTAT = evoth , INST = 10. )
```

temp10 est le champ de température extrait du résultat evoth (de type evol\_ther) à l'instant 10.

### 3.7.8.2 Extraction d'un champ de déplacement d'un concept resultat de type mode\_meca

```
mode4 = CREA_CHAMP ( OPERATION='EXTR',  
                    NOM_CHAM = 'DEPL' , TYPE_CHAM = 'NOEU_DEPL_R',  
                    RESULTAT = modes , NUME_MODE = 4 )
```

mode4 est le 4<sup>ème</sup> mode propre du résultat modes (de type mode\_meca).

### 3.7.8.3 Extraction du champ de "température" contenant les instants où la température maximum a été atteinte au cours d'un transitoire

```
instmax = CREA_CHAMP ( OPERATION='EXTR',  
                     NOM_CHAM = 'TEMP' , TYPE_CHAM = 'NOEU_TEMP_R',  
                     RESULTAT = evoth ,  
                     TYPE_MAXI = 'MAXI' , TYPE_RESU = 'INST' )
```

### 3.7.8.4 Extraction d'un champ de contraintes dans une table

Soit le fichier (fort.81) contenant l'image de la table suivante :

```
#DEBUT_TABLE  
#TITRE CONTRAINTE `ELNO'  
MAILLE POINT SIXX SIYY SIZZ  
K8 I R R R  
M1 1 -1.632E+03 -2.553E+02 6.788E-01  
M1 2 -5.302E+03 -9.663E+01 6.018E+01  
M1 3 -3.638E+03 -1.058E+02 5.669E+01  
M2 1 5.632E+01 1.553E+02 3.788E-01  
...  
#FIN_TABLE
```

On peut extraire un champ de "contraintes" aux nœuds de cette table en faisant:

```
# lecture de la table :  
TA=LIRE_TABLE(UNITE=81, TYPE_TABLE='TABLE', SEPARATEUR=' ')  
  
# extraction du champ dans la table :  
CH=CREA_CHAMP(OPERATION='EXTR', TYPE_CHAM='ELNO_SIEF_R', TABLE='TA',  
              MODELE=MO, PROL_ZERO='OUI', OPTION='SIEF_ELNO_DEPL')
```