Structures de données sd_mater et mater_code

Résumé :

On décrit ici les 3 structures de données construites à partir de la description et de la valeur des différents paramètres associés aux comportements d'un matériau (compor, mater, mater_code).

Les trois structures de données sont présentées :

- les SD en accès par nom : mater et compor
- la SD en accès par adresse : mater_code (qui remplace les précédentes dans les routines TExxxx).

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

Table des matières

1 Généralités....3
2 La structure de données mater....4
2.1 Contenu de la structure de données mater....5
2.2 La structure de données compor....5
2.3 Contenu de la structure de données compor....5
2.4 Liste de valeurs réelles ou fonctions....6
3 La structure de données mater_code....6
3.1 mater_code....7
4 Exemple....9
4.1 Fichier de commandes....9
4.2 Impression de la structure de données....9

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

1 Généralités

Le matériau apparaît dans de nombreuses commandes et intervient dans la plupart des routines de calcul des termes élémentaires (TExxxx).

En effet, et contrairement à d'autres codes, on distingue la notion de « matériau », défini par la commande DEFI_MATERIAU, et pouvant être archivé sous forme de catalogue, (voir la commande INCLUDE_MATERIAU [U4.43.02]) de la notion de comportement utilisé lors d'un calcul donné. Le matériau se compose de valeurs de paramètres associés à un certain nombre de modèles de comportements. En pratique ces paramètres sont définis par l'intermédiaire de mots-clés simples, sous des mots-clés facteurs de la commande DEFI_MATERIAU. Ces mots-clés facteurs (par exemple ELAS, ECRO_LINE, CIN1_CHAB, LEMAITRE) permettent de définir tout ou partie des paramètres nécessaires à un modèle de comportement. Par exemple, le comportement (choisi par l'utilisateur dans STAT_NON_LINE ou DYNA_NON_LINE sous le mot-clé facteur COMPORTEMENT) VMIS_CINE_LINE utilisera les paramètres matériau définis sous les mots-clés facteur ELAS et ECRO_LINE de DEFI_MATERIAU. Le comportement VISC_CIN1_CHAB utilisera ceux définis par ELAS, CIN1_CHAB et éventuellement LEMAITRE.

Par abus de langage, dans ce document, on appellera « comportement » un mot-clé facteur de DEFI_MATERIAU.

De plus, certaines commandes globales (MECA_STATIQUE, THER_LINEAIRE,...) ne permettent pas à l'utilisateur de choisir le comportement utilisé pour le calcul. Il faut donc que le calcul (par exemple de la matrice de rigidité dans les exemples précédents) puisse se faire sans ambiguïté. C'est pourquoi il est nécessaire que certains comportements de même nature (ELAS_xxx, THER_xx) soient uniques dans le matériau. Par convention de langage dans ce document, on dira que l'ensemble des comportements de même nature (même préfixe : ELAS, THER) appartiennent au même « phénomène ». Certains « comportements » peuvent donc être classés par « phénomène » dans lequel ils s'excluent mutuellement.

La structure de données associée au matériau contient le nom et les valeurs associées aux paramètres décrivant chaque comportement. Les paramètres peuvent être de type réel, complexe ou fonction. Au sein de cette structure de données on accède aux valeurs des paramètres par nom.

Pour des raisons de performances du code (notamment dans le cas des comportements non-linéaires), le matériau codé a été introduit. La structure de données associée au matériau codé est temporaire, elle contient les adresses des différents objets constituant la structure de données mater. L'accès aux valeurs des paramètres ne nécessite plus dans ce cas une mise en mémoire des objets JEVEUX et une recherche par nom à chaque fois que l'on utilise le matériau. L'accès par nom aux paramètres demeurent néanmoins.

Ces deux structures de données s'appuient sur des structures de type fonction (fonction constante, fonction d'un paramètre ou nappe) et des objets simples JEVEUX de type vecteur.

La création d'une structure de données fonction (.&&RDEP) préfixée par le nom du matériau permet de disposer ensuite d'un espace permanent de mémoire nécessaire à l'interpolation des nappes définissant les courbes de traction dépendant de la température. Cette structure de données est créée sur la base GLOBALE afin d'être échangée entre les différentes commandes du code.

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

2 La structure de données mater

Le nom du « phénomène » est la racine du nom du « comportement ». Dans les commande où l'utilisateur ne choisit pas le modèle de comportement, il faut savoir trouver sans ambiguïté les caractéristiques matérielles nécessaires, par exemple, le calcul des matrices de rigidité à travers la commande MECA_STATIQUE ne sait pas faire la distinction entre ELAS et ELAS_ORTHO, c'est pourquoi ces deux comportements s'excluent. En revanche, un matériau peut contenir les phénomènes ELAS et THER. Pour les comportements non-linéaires mécaniques, l'unicité n'est pas nécessaire, car l'utilisateur choisit une relation de comportement au niveau de la commande globale (STAT_NON_LINE).

Par exemple, les comportements ELAS, ELAS_FLUI, ELAS_ISTR, ELAS_ORTH, ELAS_THM, ELAS_COQUE constituent le phénomène ELAS.

La structure de données mater est composée :

- de plusieurs phénomènes ,
- éventuellement d'une fonction &&RDEP pour les comportements TRACTION et META_TRACTION.

```
mater  (K8) ::= record
    '.MATERIAU.NOMRC'   : OJB S V K32
     % Comportement Elastiques Généraux
     |  /   '.ELAS      '          : COMPOR
        /   '.ELAS_FLUI '          : COMPOR
        /   '.ELAS_ISTR '          : COMPOR
        /   '.ELAS_ORTH '          : COMPOR
        /   '.ELAS_THM  '          : COMPOR
        /   '.ELAS_COQUE'          : COMPOR

    % Comportements Mécaniques Non Linéaires Généraux
    |       '.TRACTION  '          : COMPOR
            '.&&RDEP'             : FONCTION
    |       '.ECRO_LINE '          : COMPOR
    |       '.PRAGER    '          : COMPOR
    |       '.CIN1_CHAB '          : COMPOR
    |       '.CIN2_CHAB '          : COMPOR
    |       '.TAHERI    '          : COMPOR
    |   /   '.LEMAITRE  '          : COMPOR
    |       '.NORTON_HOFF'         : COMPOR

    % Comportements liés à l'endommagement et la rupture
    ...
```

Remarque :

La structure de données mater ne contient pas d'information sur la forme des lois de comportement : élasticité, Lemaitre, etc … Ces dernières n'existent pas sous forme de structures de données, mais uniquement en "dur" dans le FORTRAN.

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

2.1 Contenu de la structure de données mater

```
.MATERIAU.NOMRC :       S   V   K32
```

Vecteur de type CHARACTER*16 dimensionné au moins au nombre de comportements présents lors de la définition du matériau (LONUTI est le nombre de comportements présents).

Contient le nom des comportements affectés dans la commande DEFI_MATERIAU ou DEFI_COMPOSITE (sans _FO).

La commande DEFI_COMPOSITE y stocke le nom du comportement ELAS_COQMU ou THER_COQMU, une chaîne blanche et le nom de chaque matériau pour chaque couche. L'attribut LONUTI de cet objet est récupéré dans différentes routines pour obtenir le nombre de couches.

```
.&&RDEP :               FONCTION
```

N'est présente que si les comportements TRACTION et META_TRACTION figurent dans le matériau. C'est une fonction d'une variable dimensionnée au nombre maximum de points des fonctions composant les courbes de traction dépendant de la température.

2.2 La structure de données compor

On définit tout d'abord une structure de données liée à chaque comportement pouvant définir le matériau. Un comportement est un ensemble de paramètres nommés (K16) associés à une valeur. Si la valeur est un K16 le paramètre est associé à un objet (par exemple une fonction).

```
compor (K19) ::= record

   '.VALK' :       OBJ S V K16
   '.VALR' :       OBJ S V R
   '.VALC' :       OBJ S V C
   '. ORDR ' :     OBJ S V K16
   '.KORD' :       OBJ S V I
```

Le nom complet de la structure de données compor (K19) est constitué par le nom utilisateur du matériau (K8) suivi de la chaîne ".CPT." suivi de la valeur de l'indice de rangement dans l'objet .MATERIAU.NOMRC.

2.3 Contenu de la structure de données compor

```
VALK :          S   V   K16
```

Vecteur de type CHARACTER*16 dimensionné à 2 fois le nombre maximum de paramètres, 2xNBPAR (il faut pouvoir stocker tous les noms de paramètres et tous les noms de fonctions, si tous les paramètres sont associés à des fonctions) pour la loi de comportement considérée. Par exemple E et NU sont des paramètres du comportement élastique ELAS.

Contient dans l'ordre :

- les noms des paramètres associés à des valeurs réelles,
- les noms des paramètres associés à des valeurs complexes,
- les noms des paramètres associés à des tables,
- les noms des paramètres associés à des fonctions,
- les noms des paramètres associés à des liste de valeurs réelles,
- les noms des paramètres associés à des liste de fonctions,
- le noms des objets (tables, fonctions, listes).

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

LONUTI est le nombre de l'ensemble des paramètres et objets stockés dans le vecteur.

La commande DEFI_COMPOSITE remplit cet objet avec des noms de paramètres associés aux coefficients homogénéisés ainsi que pour chaque couche. Une valeur réelle est stockée pour chacun.

.VALR : S V R

Vecteur de type REAL*8 dimensionné au nombre maximum de paramètres (NBPAR) pour la loi de comportement considérée. LONUTI fournit le nombre de paramètres réels.
Contient les valeurs associées aux paramètres réels.

.VALC : S V C

Vecteur de type COMPLEX*16 dimensionné au nombre maximum de paramètres (NBPAR) pour la loi de comportement considérée. LONUTI fournit le nombre de paramètres complexes.
Contient les valeurs associées aux paramètres complexes.
Attention : on stocke d'abord autant de 0 que de paramètres réels, puis les LONUTI valeurs complexes.

.ORDR : S V K16

Vecteur de type CHARACTER*16 dimensionné au nombre de valeurs de ORDRE_PARAM quand il existe pour le comportement.
C ontient le noms de paramètres tel que fourni par ORDRE_PARAM .

.KORD : S V I

Vecteur de type INTEGER dimensionné à 2 + 4 x NBPAR.
Contient :

- Le nombre de valeurs de .ORDR, aussi celui de ORDRE_PARAM.
- Le nombre de paramètres définis pour le comportement (nombre de réels, de complexes et autres objets).
- Pour chaque paramètre défini, sa position dans le vecteur .ORDR.
- Pour chaque paramètre défini, sa position dans le vecteur .VALK si c'est un réel, 0 sinon.
- Pour chaque paramètre défini, sa position dans le vecteur .VALK si c'est un complexe, 0 sinon.
- Pour chaque paramètre défini, sa position dans le vecteur .VALK si ce n'est ni un réel, ni un complexe, 0 sinon.

2.4 Liste de valeurs réelles ou fonctions

Quelques paramètres matériau peuvent contenir plus d'une valeur réelle ou plus d'une fonction.
Dans ce cas, un vecteur de REAL*8 ou de CHARACTER*8 est créé.

```
lisv_r8 (K24) ::= record
      '.LISV_R8'       : OBJ S V R
lisv_fo (K24) ::= record
      '.LISV_FO'       : OBJ S V K8
```

Le nom de ce vecteur est constitué par le nom utilisateur du matériau (K8) suivi de ".000000i" (sur 7 chiffres) suivi de .LISV_R8 pour des réels, .LISV_FO pour des fonctions.
Ce nom est renseigné dans .VALK du comportement comme pour une fonction, table, etc.

3 La structure de données mater_code

C'est une structure de données temporaire (créée sur la base VOLATILE) contenant les adresses mémoire des objets JEVEUX constituant un matériau (SD MATER).

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

```
mater_code  (K19) ::= record
      '.CODI'       : OBJ S V I
```

Le nom de cette structure de donnée est indexé sur le nom spécifié par le développeur et sur l'occurrence du matériau dans le cham_mater.

3.1 mater_code

.CODI : S V I

Vecteur de type INTEGER dont la dimension dépend du nombre de comportements décrits dans la structures de données MATER. Objet JEVEUX temporaire créé sur la base VOLATILE.

Des paramètres sont associés au matériau codé

LMAT : nombre de paramètres associés au comportement
LFCT : nombre de paramètres associés aux concepts (fonctions, tables)
LSUP : nombre de paramètres supplémentaires (fonctions &&RDEP, &&MZP)

Ce vecteur est de longueur 2 + NBCM*LMAT + NBCO*LFCT + NBT*LSUP où

NBCM : nombre de comportements présents dans le matériau
NBCO : nombre de concepts (fonctions, tables) présents dans le matériau
NBT : nombre de courbes de traction présentes dans le matériau

Ce vecteur contient les adresses mémoire des objets composant la structure de données MATER.

CODI(1) :	Nombre de matériaux différents (n)
CODI(2) :	Indice dans CODI du premier matériau.
CODI(3) :	Indice dans CODI du deuxième matériau (le cas échéant)
…..	….
CODI(n+1) :	Indice dans CODI du nième matériau (le cas échéant)
CODI(n+2) :	adresse de .MATERIAU.NOMRC.(pour le premier matériau)
CODI(n+3) :	NBCM nombre de comportements présents dans le matériau.
CODI(n+2+1:n+2+NBCM) :	pointeur du Kième comportement dans CODI, pour K=1 à NBCM

pour chaque Kième comportement du matériau 1

soit ipi = CODI(n+2+K)

CODI(n+2+k) :	ipi, pointeur du Kième comportement
CODI(ipi) :	nombre de paramètres associés à des réels.
CODI(ipi+1) :	nombre de paramètres associés à des complexes.
CODI(ipi+2) :	nombre de paramètres associés à des concepts (fonctions, tables)
CODI(ipi+3) :	adresse mémoire de l'objet .VALK.
CODI(ipi+4) :	adresse mémoire de l'objet .VALR.
CODI(ipi+5) :	adresse mémoire de l'objet .VALC.

pour le Lième concept de type table associée à un paramètre du Kième comportement,
soit ipif = ipi+LMAT-1

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

```
CODI(ipif+LFCT*(L-1))   : la table est transformée en une liste de réels (LIST_R8)
                          adresse mémoire de l'objet .VALE du LIST_R8
CODI(ipif+LFCT*(L-1)+1) : 0
CODI(ipif+LFCT*(L-1)+2) : 0
CODI(ipif+LFCT*(L-1)+3) : ISNNEM()
CODI(ipif+LFCT*(L-1)+4) : ISNNEM()
CODI(ipif+LFCT*(L-1)+5) : ISNNEM()
CODI(ipif+LFCT*(L-1)+6) : ISNNEM()
CODI(ipif+LFCT*(L-1)+7) : ISNNEM()
CODI(ipif+LFCT*(L-1)+8) : ISNNEM()
```

pour le Lième concept de type fonction associée à un paramètre du Kième comportement, soit ipif = ipi+LMAT-1

```
CODI(ipif+LFCT*(L-1))   : nombre de points de la fonction associée.
CODI(ipif+LFCT*(L-1)+1) : adresse mémoire de l'objet .PROL.
CODI(ipif+LFCT*(L-1)+2) : adresse mémoire de l'objet .VALE.
CODI(ipif+LFCT*(L-1)+3) : adresse mémoire du pointeur de longueur pour une nappe.
CODI(ipif+LFCT*(L-1)+4) : adresse mémoire de l'objet .PARA pour une nappe
CODI(ipif+LFCT*(L-1)+5) : attribut LONUTI de l'objet .PARA pour une nappe.
CODI(ipif+LFCT*(L-1)+6) : pointeur dans CODI pour les courbes de traction &&RDEP ou pour les assemblages combustibles &&MZP
CODI(ipif+LFCT*(L-1)+7) : sauvegarde de l'indice de l'intervalle d'interpolation.
CODI(ipif+LFCT*(L-1)+8) : sauvegarde d'un indice de recherche (équation non linéaire en thermique).
```

soit ipifc = CODI(ipif+LFCT*(L-1)+6)

```
CODI(ipifc)   : adresse mémoire de l'objet &&MZP.PROL.
CODI(ipifc+1) : adresse mémoire de l'objet &&MZP.VALE.
```

ou bien

```
CODI(ipifc)   : adresse mémoire de l'objet &&RDEP.PROL.
CODI(ipifc+1) : adresse mémoire de l'objet &&RDEP.VALE.
```


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

4 Exemple

4.1 Fichier de commandes

Les commandes ci-dessous permettent de définir 3 lois de comportement : élasticité, plasticité avec courbe de traction dépendant de la température et thermique linéaire.

```
DEBUT()

#
# donnees de modelisation
#
F_E = DEFI_FONCTION ( NOM_PARA    = 'TEMP',
                      PROL_DROITE = 'LINEAIRE',
                      PROL_GAUCHE = 'LINEAIRE',
                      VALE = (  0.,   200.E+03,
                               50.,   198.E+03,  ),
                    )
#
F_NU = DEFI_CONSTANTE ( VALE = 0.3      )
#
F_AL = DEFI_CONSTANTE ( VALE = 10.E+06  )
#
FCT1 = DEFI_FONCTION ( NOM_PARA    = 'EPSI',
                       PROL_DROITE = 'LINEAIRE',
                       PROL_GAUCHE = 'LINEAIRE',
                       VALE = (  0.200E-02,   400.,
                                 0.400E-02,   500.,  ),
                     )
#
FCT2 = DEFI_FONCTION ( NOM_PARA    = 'EPSI',
                       PROL_DROITE = 'LINEAIRE',
                       PROL_GAUCHE = 'LINEAIRE',
                       VALE = (  0.100E-02,   200.,
                                 0.300E-02,   300.,  ),
                     )
#
CTRACB = DEFI_NAPPE ( NOM_PARA    = 'TEMP',
                      PROL_DROITE = 'LINEAIRE',
                      PROL_GAUCHE = 'LINEAIRE',
                      PARA     = (  0.,     50.,   ),
                      FONCTION = (  FCT1,   FCT2,  ),
                    )
#
# materiau isotrope
#
MAT = DEFI_MATERIAU (  THER     =_F( RHO_CP = 0.0E-03,  LAMBDA = 1.0E-03, ),
                       ELAS_FO  =_F( E = F_E,  NU = F_NU,  ALPHA =
                                     F_AL,TEMP_DEF_ALPHA=20.0, ),
                       TRACTION =_F( SIGM = CTRACB, ),
                    )
#
IMPR_CO ( CONCEPT =_F( NOM= MAT ), ATTRIBUT='OUI', CONTENU='OUI' )
```

4.2 Impression de la structure de données

```
IMPRESSION DES ATTRIBUTS DES OBJETS TROUVES :
------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007 .TCO<
CLAS G
GENR V
```


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

```
TYPE K
LTYP 24
DOCU
DATE 0
LONMAX 1
LONUTI 1
LONO 1
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.&&RDEP .PROL<
CLAS G
GENR V
TYPE K
LTYP 24
DOCU
DATE 0
LONMAX 6
LONUTI 6
LONO 6
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.&&RDEP .VALE<
CLAS G
GENR V
TYPE R
LTYP 8
DOCU
DATE 0
LONMAX 8
LONUTI 8
LONO 8
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.&&RDEP ._TCO<
CLAS G
GENR V
TYPE K
LTYP 24
DOCU
```


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

```
DATE 0
LONMAX 1
LONUTI 1
LONO 1
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000001.VALC<
CLAS G
GENR V
TYPE C
LTYP 16
DOCU
DATE 0
LONMAX 2
LONUTI 0
LONO 2
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000001.VALK<
CLAS G
GENR V
TYPE K
LTYP 16
DOCU
DATE 0
LONMAX 4
LONUTI 2
LONO 4
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000001.VALR<
CLAS G
GENR V
TYPE R
LTYP 8
DOCU
DATE 0
LONMAX 2
LONUTI 2
```


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

```
LONO 2
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000001._TCO<
CLAS G
GENR V
TYPE K
LTYP 24
DOCU
DATE 0
LONMAX 1
LONUTI 1
LONO 1
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000002.VALC<
CLAS G
GENR V
TYPE C
LTYP 16
DOCU
DATE 0
LONMAX 8
LONUTI 0
LONO 8
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000002.VALK<
CLAS G
GENR V
TYPE K
LTYP 16
DOCU
DATE 0
LONMAX 16
LONUTI 11
LONO 16
IADM ************
IADD 0
```


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

```
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000002.VALR<
CLAS G
GENR V
TYPE R
LTYP 8
DOCU
DATE 0
LONMAX 8
LONUTI 5
LONO 8
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000002._TCO<
CLAS G
GENR V
TYPE K
LTYP 24
DOCU
DATE 0
LONMAX 1
LONUTI 1
LONO 1
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000003.VALC<
CLAS G
GENR V
TYPE C
LTYP 16
DOCU
DATE 0
LONMAX 1
LONUTI 0
LONO 1
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
```


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

```
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000003.VALK<
CLAS G
GENR V
TYPE K
LTYP 16
DOCU
DATE 0
LONMAX 2
LONUTI 2
LONO 2
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000003.VALR<
CLAS G
GENR V
TYPE R
LTYP 8
DOCU
DATE 0
LONMAX 1
LONUTI 0
LONO 1
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.CPT.000003._TCO<
CLAS G
GENR V
TYPE K
LTYP 24
DOCU
DATE 0
LONMAX 1
LONUTI 1
LONO 1
IADM ************
IADD 0
LADD 0
USAGE U D
--------------------------------------------------------------------------------
JEIMPA IMPRESSION DES ATTRIBUTS DE >00000007.MATERIAU.NOMRC <
CLAS G
GENR V
```


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

```
TYPE K
LTYP 32
DOCU
DATE 0
LONMAX 6
LONUTI 3
LONO 6
IADM ************
IADD 0
LADD 0
USAGE U D
================================================================================
IMPRESSION DU CONTENU DES OBJETS TROUVES :
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007 ._TCO <
>>>>>
1 - >MATER_SDASTER <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.&&RDEP .PROL <
>>>>>
1 - >FONCTION <>LIN LIN <
3 - >EPSI <>TOUTRESU <
5 - >EE <>00000007.&&RDEP <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.&&RDEP .VALE <
>>>>>
1 - 0.00000D+00 0.00000D+00 0.00000D+00 0.00000D+00 0.00000D+00
6 - 0.00000D+00 0.00000D+00 0.00000D+00
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.&&RDEP ._TCO <
>>>>>
1 - >FONCTION_SDASTER <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000001.VALC <
>>>>>
1 - ( 0.00000D+00, 0.00000D+00)( 0.00000D+00, 0.00000D+00)
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000001.VALK <
>>>>>
1 - >RHO_CP <>LAMBDA <> <
4 - > <
--------------------------------------------------------------------------------
```


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

```
IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000001.VALR <
>>>>>

1 - 0.00000D+00 1.00000D-03
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000001._TCO <
>>>>>
1 - >MATER_PROP <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000002.VALC <
>>>>>
1 - ( 0.00000D+00, 0.00000D+00)( 0.00000D+00, 0.00000D+00)
3 - ( 0.00000D+00, 0.00000D+00)( 0.00000D+00, 0.00000D+00)
5 - ( 0.00000D+00, 0.00000D+00)( 0.00000D+00, 0.00000D+00)
7 - ( 0.00000D+00, 0.00000D+00)( 0.00000D+00, 0.00000D+00)
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000002.VALK <
>>>>>
1 - >TEMP_DEF_ALPHA <>PRECISION <>K_DESSIC <
4 - >B_ENDOGE <>COEF_AMOR <>E <
7 - >NU <>ALPHA <>00000001 <
10 - >00000002 <>00000003 <> <
13 - > <> <> <
16 - > <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000002.VALR <
>>>>>
1 - 2.00000D+01 1.00000D+00 0.00000D+00 0.00000D+00 1.00000D+00
6 - 0.00000D+00 0.00000D+00 0.00000D+00
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000002._TCO <
>>>>>
1 - >MATER_PROP <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000003.VALC <
>>>>>

1 - ( 0.00000D+00, 0.00000D+00)
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000003.VALK <
```


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

```
>>>>>
1 - >SIGM <>00000006 <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000003.VALR <
>>>>>

1 - 0.00000D+00
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.CPT.000003._TCO <
>>>>>
1 - >MATER_PROP <
--------------------------------------------------------------------------------

IMPRESSION SEGMENT DE VALEURS >00000007.MATERIAU.NOMRC <
>>>>>
1 - >THER <
2 - >ELAS <
3 - >TRACTION <
4 - > <
5 - > <
6 - > <
```


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