

Structure de données FORMAT_IDEAS

Résumé:

On décrit ici la structure de données `FORMAT_IDEAS` . Cette SD est utilisée lors de l'exécution de la commande `LIRE_RESU` , elle permet de repérer et d'extraire du fichier universel « unv » d' IDEAS , les résultats souhaités par l'utilisateur.

Table des matières

1 Généralités.....	3
2 Arborescence.....	3
3 Contenu des objets.....	3
3.1 'FID_NOM' : S V K16.....	3
3.2 'FID_NUM' : S V I.....	3
3.3 'FID_PAR' : S V I.....	4
3.4 'FID_LOC' : S V I.....	4
3.5 'FID_CMP' : S V K8.....	5
3.6 'FID_NBC' : S V I.....	5
4 Structure des datasets.....	5
5 Valeurs par défaut.....	7
6 Exemples.....	7

1 Généralités

L'utilisateur définit dans la commande LIRE_RESU les informations (TYPE_RESU, NOM_CHAMP, INST, ...) qui caractérisent le résultat qu'il souhaite extraire du fichier universel d 'IDEAS . Ces renseignements ne sont pas exploitables directement, il est nécessaire de les convertir au format du fichier "universel".

La structure de donnée FORMAT_IDEAS contient tous les critères de recherche au format "universel", permettant d'extraire les résultats souhaités par l'utilisateur. Si l'utilisateur ne spécifie pas dans la syntaxe de la commande LIRE_RESU , le mot clé facteur FORMAT_IDEAS , la structure de donnée FORMAT_IDEAS est initialisée par défaut [U4.26.03]. Dans le cas contraire, l'utilisateur définit ses propres critères de recherche.

2 Arborescence

```
FORMAT_IDEAS (K16) ::= record
  ♦ '.FID_NOM' : OJB S V K16 Long = nbnoch
  ♦ '.FID_NUM' : OJB S V I Long = nbnoch
  ♦ '.FID_PAR' : OJB S V I Long = nbnoch*800
  ♦ '.FID_LOC' : OJB S V I Long = nbnoch*10
  ♦ '.FID_CMP' : OJB S V K8 Long = nbnoch*1000
  ♦ '.FID_NBC' : OJB S V I Long = nbnoch
```

3 Contenu des objets

Convention : nbnoch = nombre de champs à lire

3.1 '.FID_NOM' : S V K16

Cet objet contient le nom des champs à lire.

Pour $i=1$, nbnoch

→ $v(i)$: nom du $i^{\text{ème}}$ champ à lire

ex : 'DEPL', 'VITE', ..., 'SIEF_ELNO'

3.2 '.FID_NUM' : S V I

Cet objet contient pour chacun des champs à lire le numéro du dataset associé.

Pour $i=1$, nbnoch

→ $v(i)$: numéro du dataset associé au $i^{\text{ème}}$ champ à lire

ex : 55, 57, 2414

3.3 '.FID_PAR' : S V I

Cet objet contient pour chacun des champs à lire les caractéristiques de l'entête du dataset recherché. Cet entête est composé au maximum de 20 "records" constitué de 40 "fields" chacun.

V (1)	1 ^{er} field du record 1 du champ 1
...	
V (48)	8 ^{ème} field du record 2 du champ 1
...	
V (800)	40 ^{ème} field du record 20 du champ 1
...	
V (6401)	1 ^{er} field du record 1 du champ 9
...	
V (7200)	40 ^{ème} field du record 20 du champ 9
...	
$v((ich-1)*800+(irec-1)*40+ifield)$	valeur associée au champ ich située à l'enregistrement irec et pour le champ ifield

3.4 '.FID_LOC' : S V I

Cet objet contient pour chaque champ, 5 couples de valeurs entières permettant de localiser à l'intérieur du dataset, le numero d'ordre, l'instant, la fréquence ... La première valeur indique le n° de l'enregistrement où est stockée l'information et la deuxième valeur indique sa position.

v (1)	= N° de l'enregistrement	Numéro d'ordre	
v (2)	= Position		
v (3)	= N° de l'enregistrement	Instant	Champ n°1
v (4)	= Position		
v (5)	= N° de l'enregistrement	Fréquence	
v (6)	= Position		
v (7)	= N° de l'enregistrement	Nume_mode	
v (8)	= Position		
v (9)	= N° de l'enregistrement	Mass_gene	
v (10)	= Position		
v (11)	= N° de l'enregistrement	Numéro d'ordre	
v (13)	= Position		
v (13)	= N° de l'enregistrement	Instant	Champ n°2
v (14)	= Position		
...			

3.5 '.FID_CMP' : S V K8

Cet objet contient pour chaque champ les composantes de la grandeur à lire.

v (1)	DX	
v (2)	DY	
v (3)	DZ	Champ n°1
v (4)	DRX	
v (5)	DRY	
v (6)	DRZ	
v (1001)	EPXX	
v (1002)	EPYY	
v (1003)	EPZZ	Champ n°2
v (1004)	EPXY	
v (1005)	EPXZ	
v (1006)	EPYZ	
...		
...		

3.6 '.FID_NBC' : S V I

Cet objet contient pour chaque champ le nombre de composantes à lire.

4 Structure des datasets

La structure générale des datasets résultats 55 , 57 et 2414 exploités par la commande LIRE_RESU est composée de 2 parties :

- Partie A : entête contenant des informations générales,
- Partie B : contient les valeurs.

	-1	Partie A
	55 %VALEURS AUX NOEUDS	
Record 1 →	ASTER 5.01.00 CONCEPT TEMPE CALCULE - CHAMP AUX NOEUDS DE ...	
Record 2 →	CHAMP AUX NOEUDS DE NOM SYMBOLIQUE TEMP - TEMP	
Record 3 →	ASTER 5.01.00 CONCEPT TEMPE CALCULE LE 18/12/98 A 15:19:49 DE ...	
Record 4 →	CHAMP AUX NOEUDS DE NOM SYMBOLIQUE TEMP	
Record 5 →	NUMERO D'ORDRE: 0 INST: 0.000000E+00	
Record 6 →	2 4 1 5 2 1	
Record 7 →	2 1 1 0	
Record 8 →	0.000000E+00	
	1 % NOEUD N1	Partie B

	205 % NOEUD N205	
	1.000000E+02	
	-1	

Figure 4-a : Dataset 55 (exemple)

-1						Partie A
57 \$VALEURS AUX NOEUDS DES ELEMENTS						
Record 1 →	ASTER 3.05.30 CONCEPT 0 CALCULE LE - CHAMP PAR ELEMENT AUX ...					
Record 2 →	CHAMP PAR ELEMENT AUX NOEUDS DE NOM SYMBOLIQUE VARI_ELNO_ELGA - ...					
Record 3 →	ASTER 3.05.30 CONCEPT 0 CALCULE LE 29/12/95 A 09:56:55 DE TYPE ...					
Record 4 →	CHAMP PAR ELEMENT AUX NOEUDS DE NOM SYMBOLIQUE VARI_ELNO_ELGA					
Record 5 →						
Record 6 →	1	4	3	0	2	6
Record 7 →	2	1	1	1		
Record 8 →	0.15000E+02					
						Partie B
	1	1	0	6	\$ MAILLE MA2	
	2.07919E-05	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.

	2.07919E-05	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.
	-1					

Figure 4-b: Dataset 57 (exemple)

Figure 4-c: Dataset 2414 (exemple)

-1								Partie A
2414								
Record 1 →	1							
Record 2 →	B.C. 1, TEMPERATURE_1, LOAD SET 1							
Record 3 →	1							
Record 4 →	/users/lebonv/SGI/Code_Aster/TPLS100B/tplsr100_coque.mf1							
Record 5 →	MODEL_SOLUTION_SOLVE							
Record 6 →	LOAD SET 1							
Record 7 →	Analysis time was		06-Jan-99	11:11:25				
Record 8 →	NONE							
Record 9 →	2	1	1	5	2	1		
Record 10 →	-10	0	1	1	1	0	0	
Record 11 →	2	0						
Record 12 →	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.		
Record 13 →	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.		
								Partie B
	1							
	-6.10352E-06							

	205							
	-6.10352E-06							
	-1							

5 Valeurs par défaut

Les valeurs par défaut stockées dans la structure de données `FORMAT_IDEAS` sont présentées dans le document d'utilisation [U2.26.03] .

6 Exemples

Dans ce paragraphe, nous présentons deux exemples :

- Exemple A : utilisation des critères de recherche par défaut pour lire les résultats,
- Exemple B : utilisation des critères de recherche utilisateur pour lire les résultats.

Pour chacun de ces exemples nous donnons la syntaxe de la commande `LIRE_RESU` ainsi que le contenu de la structure de données `FORMAT_IDEAS`.

Exemple A : critères de recherche par défaut

- Syntaxe de la commande `LIRE_RESU`

```
INIT = LIRE_RESU ( MAILLAGE = m ,  
                  UNITE     = 19 ,  
                  FORMAT    = 'IDEAS',  
                  TYPE_RESU = 'EVOL_NOLI',  
                  NOM_CHAM  = ('DEPL'),  
                  INST      = 15. ,  
                  )
```

- Contenu de la SD `FORMAT_IDEAS` (valeur par défaut)

<code>FORMAT_IDEAS</code>	(1)	(2)	(3)	(4)	(5)	(6)
<code>.'FID_NOM'</code>	DEPL					
<code>.'FID_NUM'</code>	55					
<code>.'FID_PAR '(1)</code>	9999	9999	9999	9999	9999	9999
(2)	9999	9999	9999	9999	9999	9999
(3)	9999	9999	9999	9999	9999	9999
(4)	9999	9999	9999	9999	9999	9999
(5)	9999	9999	9999	9999	9999	9999
(6)	1	4	3	8	2	6
(7)	9999	9999	9999	9999	9999	9999
...						
(13)	9999	9999	9999	9999	9999	9999
...						
(40)						
...						
<code>.'FID_LOC'</code>	7	4	8	1	9999	9999
<code>.'FID_CMP'</code>	'DX'	'DY'	'DZ'	'DRX'	'DRY'	'DRZ'

Lors de la recherche du dataset, le nombre 9999 est un joker permettant d'ignorer la valeur lue dans l'entête.

Exemple B : critères de recherche défini par l'utilisateur.

- Syntaxe de la commande LIRE_RESU

```
INIT = LIRE_RESU (  MODELE      = mo  ,
                   UNITE       = 19  ,
                   FORMAT      = 'IDEAS',
                   TYPE_RESU   = 'EVOL_NOLI',
                   NOM_CHAM    = ('15') ,
                   INST        = 15.  ,
                   FORMAT_IDEAS = _F (
                       NOM_CHAM    = 'DEPL',
                       NUME_DATASET = 55,
                       RECORD_6    = (3,1,9999,4,2,3),
                       POSI_ORDRE  = (8,4),
                       POSI_INST   = (7,9999),
                       CMP         = ('DX','DY','DZ'),
                       )
                   )
```

- Contenu de la SD FORMAT_IDEAS (valeurs par défaut)

FORMAT_IDEAS	(1)	(2)	(3)	(4)	(5)	(6)
.'FID_NOM'	DEPL					
.'FID_NUM'	55					
.'FID_PAR' (1)	9999	9999	9999	9999	9999	9999
(2)	9999	9999	9999	9999	9999	9999
(3)	9999	9999	9999	9999	9999	9999
(4)	9999	9999	9999	9999	9999	9999
(5)	9999	9999	9999	9999	9999	9999
(6)	3	1	9999	4	2	3
(7)	9999	9999	9999	9999	9999	9999
...						
(13)	9999	9999	9999	9999	9999	9999
...						
(40)						
...						
.\FID_LOC'	8	4	7	1	9999	9999
.\FID_CMP'	'DX'	'DY'	'DZ'	'XXX'	'XXX'	'XXX'