Utilisation des SD_RESULTAT

Résumé :

La SD_RESULTAT [D4.06.08]: n'est accessible qu'à travers les routines décrites dans ce document.

La routine RSCRSD doit être complétée pour créer de nouveaux types de SD_RESULTAT.

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

Table des matières

1 Liste des Routines....3
2 Routines de création, destruction, agrandissement....5
3 Routines d'accès à partir des mots clés de la commande Aster....5
4 Routines d'accès aux structures SD_RESULTAT....7
5 Routine interpolation....10
6 Routines d'existence....10
7 Exemple....12

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

1 Liste des Routines

Routines de Création, Destruction, Agrandissement	
RSCRSD	Créer une structure SD_RESULTAT
RSDLSD	Détruire une structure SD_RESULTAT
RSRUSD	Détruire tous les champs au numéro d'ordre IORDR d'une structure SD_RESULTAT
RSAGSD	Agrandir une structure SD_RESULTAT

Routines d'accès à partir des mots clés de la commande Aster	
RSUTNU	Récupération à partir des mots clés de la commande, des numéros d'ordre dans une structure SD_RESULTAT
RSTRAN	Récupération a partir des mots clés de la commande pour une structure Résultat tran_gene

Routines d'accès aux structures SD_RESULTAT	
RSORAC	Récupération à partir d'une variable d'accès, du ou des numéros d'ordre d'une structure SD_RESULTAT
RSEXCH	Récupération a partir d'un nom symbolique et d'un numéro d'ordre, du nom d'un champ dans une structure SD_RESULTAT.
RSNOCH	Noter à partir d'un nom symbolique et d'un numéro d'ordre, le nom d'un champ dans une structure SD_RESULTAT.
RSADPA	Récupération à partir d'un numéro d'ordre et d'une liste de nom(s) de paramètre(s) ou de variable(s) d'accès, des adresses JEVEUX d'une structure SD_RESULTAT
RSNOPA	Récupération des noms et du nombre de paramètres dans une structure SD_RESULTAT ainsi que les noms et le nombre de variables d'accès. Ces noms sont stockés dans un vecteur JEVEUX de nom nomjv.
RSUTNC	Récupération à partir d'un nom symbolique, des champs notés et de leurs numéros d'ordre associés dans une structure SD_RESULTAT .
RSUTOR	Récupération à partir d'un champ, du numéro d'ordre et éventuellement du nom symbolique dans une structure SD_RESULTAT.

Routine d'interpolation	
RSINCH	Interpolation pour un nom symbolique et une variable d'accès, d'un champ dans une structure SD_RESULTAT

Routines d'existence	
RSEXIS	Vérification de l'existence d'une structure SD_RESULTAT.
RSEXPA	Vérification de l'existence d'un nom de paramètre ou d'un nom de variable d'accès dans une structure SD_RESULTAT
RSVPAR	Vérification de l'existence d'un nom de paramètre et de sa valeur pour un numéro d'ordre

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

	donné dans une structure SD_RESULTAT

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

2 Routines de création, destruction, agrandissement

CALL RSCRSD (nomsd, typesd, nbordr)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	typesd	K16	type de la structure de données RESULTAT
IN	nbordr	I	nombre de numéro d'ordre

Créer une structure SD_RESULTAT (si cette structure existe déjà, on la détruit). La modification de cette routine permet de créer de nouveaux types de SD_RESULTAT ou d'enrichir ceux existants.

CALL RSDLSD (nomsd)

IN	nomsd	K8	nom de la structure de données RESULTAT

Détruire une structure SD_RESULTAT. Cette routine ne détruit pas les champs répertoriés par la SD_RESULTAT.

CALL RSRUSD (nomsd, iordr)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	iordr	I	numéro d'ordre

Détruire dans la structure SD_RESULTAT les champs à partir d'un numéro d'ordre donné.

CALL RSAGSD (nomsd, nbordr)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	nbordr	I	nouvelle taille du RESULTAT (la taille est doublée si NBORDR = 0)

Agrandir une structure SD_RESULTAT.

3 Routines d'accès à partir des mots clés de la commande Aster

L'accès via le langage de commandes aux données répertoriées dans les SD_RESULTAT est possible par les mêmes mots clés quel que soit la commande (IMPR_RESU, CALC_G, etc …).

La description de ces mots clés (TOUT_ORDRE, NUME_ORDRE, etc …) se trouve par exemple en [U4.71.01].

Pour certaines commandes (IMPR_RESU, par exemple) ces mots clés sont sous un mot clé facteur (RESU) pour d'autres non (CALC_G). Dans ce dernier cas l'argument mot clé spécifié dans les routines ci-dessous doit être ' ' .

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

CALL RSUTNU (nomsd, motcle, iocc, knum, nbordr, prec, crit, iret)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	motcle	K*	mot clé facteur de la commande
IN	iocc	I	numéro d'occurrence du mot clé facteur
IN	knum	K19	nom du vecteur JEVEUX ZI pour écrire la liste des numéros
OUT	nbordr	I	nombre de numéros d'ordre trouvés
IN	prec	R8	précision demandée
IN	crit	K8	critère de recherche
OUT	iret	I	code retour d'erreur, = 0 : ok

Récupération à partir des mots clés de la commande, des numéros d'ordre dans une structure SD_RESULTAT.

Remarque :

La routine RSUTNU ci-dessus récupère la liste des numéros d'ordre possibles de la SD_RESULTAT , éventuellement filtrée par les mots clés du langage de commande (TOUT='OUI' , LIST_INST=...). Il n'est pas sûr pour autant, que pour tous ces numéros d'ordre, tous les champs (NOM_CHAM) ou tous les paramètres existent vraiment. Lorsque l'on veut récupérer la liste des numéros d'ordre pour lesquels un champ (ou un paramètre) existe vraiment, il faut utiliser les routines RSUTN1 ou RSUTN2 .

CALL RSTRAN (interp, nomsd, motcle, iocc, kinst, krang, nbinst, iret)

IN	interp	K4	type d'interpolation souhaité
IN	nomsd	K8	nom de la structure tran_gene
IN	motcle	K16	mot clé facteur de la commande
IN	iocc	I	numéro d'occurrence du mot clé facteur
IN	kinst	K19	nom du vecteur JEVEUX pour écrire la liste des instants
IN	krang	K19	nom du vecteur JEVEUX pour écrire la liste des numéros
OUT	nbinst	I	nombre d'instants lus
OUT	iret	I	code retour, = 0 : ok

Récupération à partir des mots clés de la commande pour une structure Résultat tran_gene, suivant le type d'interpolation :

- des instants (INTERP = 'LIN' , 'LOG'), données utilisateur,
- des instants et leurs numéros d'ordre associés (INTERP = 'NON'), instants de calcul du tran_gene .

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

4 Routines d'accès aux structures SD_RESULTAT

CALL RSORAC (nomsd, acces, ival, rval, kval, cval, prec, crit, nutrou, ndim, nbtrou)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	acces	K16	nom de la variable d'accès.
IN	ival	I	valeur de la variable d'accès si entière
IN	rval	R8	valeur de la variable d'accès si réelle
IN	kval	K..	valeur de la variable d'accès si caractère
IN	cval	C16	valeur de la variable d'accès si complexe
IN	prec	R8	précision demandée
IN	crit	K8	critère demandé
IN	ndim	I	dimension de la liste nutrou.
OUT	nutrou	I	liste des numéros d'ordre trouvés.
OUT	nbtrou	I	nombre de numéros d'ordre trouvés. si le nombre trouvé nbtrou est supérieur à ndim, alors on rend nbtrou = -nbtrou

Récupération à partir d'une variable d'accès, du ou des numéros d'ordre d'une structure SD_RESULTAT.

Convention :
- si acces = 'LONUTI' : récupération de la longueur utile du .ORDR
- si acces = 'LONMAX' : récupération de la longueur maxi du .ORDR
- si acces = 'DERNIER' : récupération du dernier numéro d'ordre
- si acces = 'PREMIER' : récupération du premier numéro d'ordre
- si acces = 'TOUT_ORDRE' : récupération de tous les numéros d'ordre.

CALL RSEXCH (kstop, nomsd, nomsy, iordr, chextr, iret)

IN	kstop	K1	'F' / ' ' Si 'F' et iret > 0 => erreur fatale pour l'utilisateur.
IN	nomsd	K8	nom de la structure de donnes RESULTAT
IN	nomsy	K16	nom symbolique
IN	iordr	I	numéro d'ordre
OUT	chextr	K19	nom du champ extrait
OUT	iret	I	code retour = 0 : le champ est « possible » et il existe. =100 : le champ est « possible » mais n'existe pas = 101: le nom symbolique est interdit. = 110 : La sd_resultat est « pleine ». Il faut l'aggrandir. = 102 : le numéro d'ordre IORDR n'est pas permis

Récupération à partir d'un nom symbolique et d'un numéro d'ordre, du nom d'un champ dans une structure SD_RESULTAT.

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

Cette routine gère les champs dans la structure SD_RESULTAT et fait appel a la routine RSUTCH.

CALL RSNOCH (nomsd, nomsy, iordr)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	nomsy	K16	nom symbolique
IN	iordr	I	numéro d'ordre

Noter (=stocker) à partir d'un nom symbolique et d'un numéro d'ordre, le nom d'un champ dans une structure SD_RESULTAT.

CALL RSADPA (nomsd, cel, npara, lpara, iordr, itype, ljeveu, ctype)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	cel	K1	condition d'accès aux paramètres : 'L' : lecture, 'E' : écriture.
IN	npara	I	nombre de paramètres cherchés
IN	lpara	K16	liste des noms des paramètres
IN	iordr	I	numéro d'ordre
IN	itype	I	code indiquant que l'on désire le type = 0 pas de type ? 0 on fournit le type
OUT	ljeveu	I	liste des adresses JEVEUX dans ZI, ZR, ...
OUT	ctype	K4	liste des types des adresses JEVEUX I entier, R réel, C complexe, 'K8' 'K16' 'K24' 'K32' 'K80' caractère.

Récupération à partir d'un numéro d'ordre et d'une liste de nom(s) de paramètre(s) ou de variable(s) d'accès, des adresses JEVEUX d'une structure SD_RESULTAT.

CALL RSNOPA (nomsd, icode, nomjv, nbacc, nbpara)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	nomjv	K19	nom du vecteur JEVEUX pour écrire la liste des noms
IN	icode	I	= 0, variables d'accès seules = 1, paramètres seuls = 2, variables d'accès et paramètres
OUT	nbacc	I	nombre de variables d'accès
OUT	nbpara	I	nombre de paramètres

Récupération des noms et du nombre de paramètres dans une structure SD_RESULTAT ainsi que les noms et le nombre de variables d'accès. Ces noms sont stockés dans un vecteur JEVEUX de nom nomjv.

CALL RSUTNC (nomsd, nomsy, ndim, nomch, nuordr, nbtrou)

IN	nomsd	K8	nom de la structure de données RESULTAT

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

IN	nomsy	K16	nom symbolique
IN	ndim	I	dimension des tableaux
OUT	nomch	K16	tableau des noms de champs
OUT	nuordr	I	tableau des numéros d'ordre des champs trouvés
OUT	nbtrou	I	nombre de champs trouvés si le nombre trouvé nbtrou est supérieur à ndim, alors on rend nbtrou = -nbtrou

Récupération à partir d'un nom symbolique, des champs notés et de leurs numéros d'ordre associés dans une structure SD_RESULTAT.

CALL RSUTOR (nomsd, chextr, nomsy, nuordr, iret)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	chextr	K19	nom du champ à rechercher
OUT	nomsy	K16	nom symbolique
OUT	nuordr	I	numéro d'ordre
OUT	iret	I	code retour: =1, le champ a été trouvé, =0 sinon

Récupération à partir d'un champ, du numéro d'ordre et éventuellement du nom symbolique dans une structure SD_RESULTAT.

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

5 Routine interpolation

CALL RSINCH (nomsd, nomsy, acces, rval, chextr, proldr, prolga, istop, base, iret)

nomsd	IN	K8	nom de la structure de données RESULTAT
nomsy	IN	K16	nom symbolique
acces	IN	K16	nom de la variable d'accès
rval	IN	R8	valeur réelle de la variable d'accès
chextr	IN	K19	nom du champ a créer. (s'il existe, on le détruit)
proldr	IN	K8	type de prolongement voulu à droite : 'CONSTANT', 'LINEAIRE', 'EXCLU'
prolga	IN	K8	type de prolongement voulu à gauche : 'CONSTANT', 'LINEAIRE', 'EXCLU'
istop	IN	I	en cas d'erreur d'interpolation : = 0, n'écrit pas de message, ne fait pas stop. = 1, écrit message, ne fait pas stop. = 2, écrit message, fait stop.
base	IN	K8	base du champ créé
iret	OUT	I	code retour : 1) le champ est calculé : = 0, le champ est interpolé entre 2 valeurs. = 1, le champ est prolongé à gauche. = 2, le champ est prolongé à droite. 2) le champ n'est pas calculé : = 10, il n'existe aucun champ pour l'interpolation. = 11, le prolongement à gauche interdit. = 12, le prolongement à droite interdit. = 20, la variable d'accès est illicite.

Interpolation pour un nom symbolique et une variable d'accès, d'un champ dans une structure SD_RESULTAT

6 Routines d'existence

CALL RSEXIS (nomsd, iret)

IN	nomsd	K8	nom de la structure de données RESULTAT
OUT	iret	I	code retour : = 0, la structure existe ; ? 0, la structure n'existe pas.

Vérification de l'existence d'une structure SD_RESULTAT.

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

CALL RSEXPA (nomsd, icode, nompar, iret)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	icode	I	= 0, variable d'accès = 1, paramètre = 2, variable d'accès ou paramètre
IN	nompar	K16	nom du paramètre ou de la variable d'accès
OUT	iret	I	code retour: = 0, le nom n'existe pas; = 100, le nom existe.

Vérification de l'existence d'un nom de paramètre ou d'un nom de variable d'accès dans une structure SD_RESULTAT.

CALL RSVPAR (nomsd, iordr, nompar, ipar, rpar, kpar, iret)

IN	nomsd	K8	nom de la structure de données RESULTAT
IN	iordr	I	numéro d'ordre
IN	nompar	K16	nom du paramètre à vérifier
IN	ipar	I	valeur du paramètre si entière
IN	rpar	R8	valeur du paramètre si réelle
IN	kpar	K..	valeur du paramètre si caractère
OUT	iret	I	code retour: = 0, ce n'est pas un paramètre = 100, la valeur du paramètre est correcte = 110, la valeur du paramètre n'est pas correcte

Vérification de l'existence d'un nom de paramètre et de sa valeur pour un numéro d'ordre donné dans une structure SD_RESULTAT.

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

7 Exemple

Prenons l'exemple du stockage des modes de vibration mécanique (mode_meca), on a calculé 5 modes et on les stocke dans une structure SD_RESULTAT.

Étape 1 : création de la structure SD_RESULTAT

```
NBMODE = 5
CALL RSCRSD ( MODES, 'MODE_MECA', NBMODE )
```

Étape 2 : stockage des vecteurs propres

```
DO 10 IM = 1, NBMODE
...
CALL RSEXCH (' ', MODES, 'DEPL', IM, CHAMNO, IER )
...
CALL JEVEUO(CHAMNO//'.VALE','E',LVALE)
DO 12 IEQ = 1, NEQ
  ZR(LVALE+IEQ-1) = ...
12  CONTINUE
CALL RSNOCH ( MODES, 'DEPL', IM)
...
```

Étape 3 : stockage des variables d'accès et paramètres modaux

```
CALL RSADPA ( MODES, 'E' , 1, 'FREQ' , IM, 0, LFREQ, K8B )
ZR(LFREQ) = ...
CALL RSADPA ( MODES, 'E', 1, 'NUME_MODE', IM, 0, LNUME, K8B )
ZI(LNUME) = ...
CALL RSADPA ( MODES, 'E', 1, 'NORME' , IM, 0, LNORM, K8B )
ZK24(LNORM) = ...
...
```


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