

Manuel de Descriptif Informatique
Fascicule D6.05 : -
Document D6.05.01

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.

Table des matières

1	Routines de création, destruction, agrandissement	3
2	Routines d'accès à partir des mots clés de la commande Aster	4
3	Routines d'accès aux structures SD_RESULTAT	5
4	Routine interpolation	8
5	Routines d'existence	8
6	Exemple	10

1 Routines de création, destruction, agrandissement

Remarque :

Qu'est-ce qu'une structure de donnée SD_RESULTAT, qu'est-ce qu'un numéro d'ordre, un paramètre, une variable d'accès ... ? les réponses à ces questions sont dans [D3.04.01] et [D4.06.08].

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.

2 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_THETA, 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_THETA). Dans ce dernier cas l'argument mot clé spécifié dans les routines ci-dessous doit être ''.

CALL RSUTNU	(nomsd, motcle, iocc, knum, nbordr, jordr, prec, crit,tout, ilinst, ilfreq, 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 pour écrire la liste des numéros
(out)	nbordr	I	nombre de numéros d'ordre trouvés
(out)	jordr	I	adresse JEVEUX du vecteur ZI
(in)	prec	R8	précision demandée
(in)	crit	K8	critère demandé
(in)	itout	I	accès de la commande par le mot clé "TOUT"
(in)	ilinst	I	accès de la commande par le mot clé "LIST_INST"
(in)	ilfreq	I	accès de la commande par le mot clé "LIST_FREQ"
(out)	iret	I	code retour, = 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, jinst, jrang, 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)	jinst	I	adresse JEVEUX du vecteur ZR des instants
(out)	jrang	I	adresse JEVEUX du vecteur ZI des numéros
(out)	iret	I	code retour, = 0 : ok

Récupération a 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.

3 Routines d'accès aux structures SD_RESULTAT

CALL RSORAC			(nomsd, accès, ival, rval, kval, cval, prec, crit, nutrou, ndim, nbtrou)
(in)	nomsd	K8	nom de la structure de données RESULTAT
(in)	accès	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 accès = 'LONUTI' : récupération de la longueur utile du .ORDR
si accès = 'LONMAX' : récupération de la longueur maxi du .ORDR
si accès = 'DERNIER' : récupération du dernier numéro d'ordre
si accès = 'PREMIER' : récupération du premier numéro d'ordre
si accès = 'TOUT_ORDRE' : récupération de tous les numéros d'ordre.

CALL RSEXCH			(nomsd, nomsy, iordr, chextr, iret)
(in)	nomsd	K8	nom de la structure de données 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 existe. >0 : le champ n'existe pas. 1) IORDR est inférieur au max autorisé : 100 : le nom symbolique est licite. 101 : le nom symbolique est interdit. 1) IORDR est supérieur au max autorisé : 110 : le nom symbolique est licite. 111 : le nom symbolique est interdit.

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

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

CALL RSNOCH		(nomsd, nomsy, iordr, chextr)	
(in)	nomsd	K8	nom de la structure de données RESULTAT
(in)	nomsy	K16	nom symbolique
(in)	iordr	I	numéro d'ordre
(in)	chextr	K19	nom du champ "extrait" sur lequel on veut pointer.

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

Convention :

chextr = '' : le nom du champ est celui construit par la routine,
chextr ≠ '' : le nom du champ sera celui fournit par le développeur.

Remarque :

La routine RSNOCH gère les liens (via la routine SDLIEN) entre les différentes structures 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, jpara)	
(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
(out)	jpara	I	adresse JEVEUX de nomjv

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
(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, nomsy, chextr, ndim, tnomsy, nuordr, nbtrou)	
(in)	nomsd	K8	nom de la structure de données RESULTAT
(in)	nomsy	K16	nom symbolique
(in)	chextr	K19	nom du champ à rechercher
(in)	ndim	I	dimension des tableaux
(out)	tnomsy	K16	tableau des noms symboliques
(out)	nuordr	I	tableau des numéros d'ordre
(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 champ, du numéro d'ordre et éventuellement du nom symbolique dans une structure SD_RESULTAT.

Convention :

si nomsy = '' : récupération des noms symboliques et des numéros d'ordre.

4 Routine interpolation

CALL RSINCH			(nomsd, nomsy, accès, rval, chextr, proldr, prolga, istop, iret)
(in)	nomsd	K8	nom de la structure de données RESULTAT
(in)	nomsy	K16	nom symbolique
(in)	accès	K16	nom de la variable d'accès
(in)	rval	R8	valeur réelle de la variable d'accès
(in)	chextr	K19	nom du champ à créer. (s'il existe, on le détruit)
(in)	proldr	K8	type de prolongement voulu à droite
(in)	prolga	K8	type de prolongement voulu à gauche
(in)	istop	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.
(out)	iret	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. 1) 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.

5 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.

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	= 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	= 0, ce n'est pas un paramètre = 100, la valeur du paramètre est correct = 110, la valeur du paramètre n'est pas correct

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

6 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.

Etape 1 : création de la structure SD_RESULTAT

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

Etape 2 : stockage des vecteurs propres

```
DO 10 LM = 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, ' ' )  
  ...
```

Etape 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) = ...  
...
```