

MSLIB Fortran 90

CS SI

Nomenclature : **M-MU-0-120-CIS**

Edition : 01 Date: 03/08/1998

Révision: 01 Date: 10/04/2002

Volume Z

Utilitaires de traitement des codes retour

Rédigé par : Guylaine PRAT	le : CS (SI/Espace/MS/EC)	
Validé par : Guylaine PRAT Marie-Claude RIBIS (ingénieur qualité)	le : CS (SI/Espace/MS/EC) CS (SI/Espace)	
Pour application : Eric LE DÉ (chef de projet Cnes)	le : CNES (DTS/MID/MS/CMS)	

DIFFUSION INTERNE CNES**Observations**

Voir la note nomenclaturée M-NT-0-18-CN:
"Liste de diffusion de la documentation utilisateur MSLIB".

DIFFUSION EXTERNE CNES**Observations**

Voir la note nomenclaturée M-NT-0-18-CN:
"Liste de diffusion de la documentation utilisateur MSLIB".

BORDEREAU D'INDEXATION**CONFIDENTIALITE :** NC**MOTS-CLES :****TITRE :** Volume Z - Utilitaires de traitement des codes retour**AUTEUR :** Guylaine PRAT**RESUME :**

Ce document rassemble les notices d'utilisation des routines du thème "Utilitaires de traitement des codes retour".

SITUATION DU DOCUMENT : Création**VOLUME :****PAGES :** 15**PLANCHES :****FIGURES :****LANGUES :** F**CONTRAT :** Marché 870/96/Cnes/0720 BC 62 Lot 6**SYSTEME HOTE :** Frame6/MSLIB

MODIFICATION

ETAT DOCUMENT				PAGES REVISEES	
ED.	REV.	DATE	REFERENCE ORIGINE (pour chaque édition)	ETAT PAGE *	NUMERO DES PAGES
01	00	03/08/98	M-MU-0-120-CIS Rédacteur : G. Prat	M	Création
01	01	10/04/02	M-MU-0-120-CIS Rédacteur : G. Prat		§ Code retour des routines (modification de libellés)

* I = Inséré

S = Supprimé

M = Modifié

Sommaire

Présentation du thème Z : *page 1*

Liste des routines du thème Z : *voir pages suivantes du sommaire.*

Liste des routines du thème Z:

mz_numero_routine : *page 2*

“Cette routine donne le nom et le rôle d’une routine, identifiée par son numéro d’identification (ce numéro est contenu dans le champ %routine du code retour)”.

mz_val_code_retour : *page 6*

“Cette routine donne la signification du champ %valeur d’un code retour”.

Présentation du thème Z

Le thème “*Utilitaires de traitement des codes retour*” regroupe une série de routines qui permettent de traiter les champs “%valeur” et “%routine” des codes retour des routines de la MSLIB Fortran 90.

Rappel:

- le champ “%valeur” est prévu pour permettre d’effectuer un diagnostic sur les sorties de la routine,
- quant au champ “%routine”, son rôle est de fournir la valeur du numéro d’identification de la routine.

Avant d’utiliser les routines de ce thème, nous vous recommandons de lire les chapitres concernant “les paramètres de la MSLIB Fortran 90” et la “gestion des anomalies” du volume 3 de la documentation utilisateur: “*Caractéristiques principales et conventions d’utilisation de la MSLIB Fortran 90*”.

Routine *mz_numero_routine*

Identification

“Cette routine donne le nom et le rôle d’une **routine**, identifiée par son **numéro** d’identification (ce numéro est contenu dans le champ `%routine` du code retour)”.

Rôle

Pour une routine MSLIB, identifiée par son numéro d’identification, cette routine donne le nom et le rôle succinct associés.

Avant d’utiliser cette routine, nous vous recommandons de lire les chapitres concernant “les paramètres de la MSLIB Fortran 90” et la “gestion des anomalies” du volume 3 de la documentation utilisateur: “*Caractéristiques principales et conventions d’utilisation de la MSLIB Fortran 90*”.

Séquence d’appel

(voir explications dans le volume 3)

call *mz_numero_routine* (*routine*, *nom*, *identification*, *code_retour*)

Description des arguments

(voir explications dans le volume 3)

• Entrées obligatoires

integer	routine	numéro d’identification de la routine
---------	----------------	---------------------------------------

• Sorties obligatoires

character(len=pm_nom_routine)	nom	nom de la routine
character(len=pm_identification_routine)	identification	rôle succinct de la routine
tm_code_retour	code_retour	

Conditions sur les arguments

- La chaîne **nom** est dimensionnée à l’aide du parameter MSLIB: *pm_nom_routine*. La déclaration peut donc se faire de la sorte:

```
character(len=pm_nom_routine) :: nom
```

- La chaîne **identification** est dimensionnée à l’aide du parameter MSLIB: *pm_identification_routine*. La déclaration peut donc se faire de la sorte:

```
character(len=pm_identification_routine) :: identification
```


Notes d'utilisation

- Compte tenu du surdimensionnement nécessaire de la longueur des chaînes de caractères **nom** et **identification**, nous recommandons l'utilisation de la fonction intrinsèque fortran 90 *trim*, après l'appel à *mz_numero_routine*. Il suffit de faire, par exemple:

`trim(nom)`

pour obtenir la chaîne **nom** débarrassée de tous les caractères blancs en fin de chaîne.

- La valeur de **routine** est fournie par:

`routine = code_retour%routine`

en sortie de n'importe quelle routine de la bibliothèque.

Références documentaires

Sans objet.

Code retour (voir explications dans le volume 3)

pm_OK (0) : Retour normal.

pm_err_numero_routine_inconnu (-1807) : Cette valeur de numéro de routine est inconnue dans la bibliothèque.

Exemple n°1 en Fortran 90 portable (voir explications dans le volume 3)

```
program THEME_Z_1

  use mslib

  integer                                :: ROUTINE
  character(len=pm_nom_routine)          :: NOM
  character(len=pm_identification_routine) :: IDENTIFICATION
  type(tm_code_retour)                   :: CODE_RETOUR

  ROUTINE = pm_num_mt_topo_E_car_sgd
  ! "pm_num_mt_topo_E_car_sgd" est le parameter MSLIB qui contient
  ! le numero d'identification de la routine mt_topo_E_car_sgd

  call mz_numero_routine(ROUTINE, NOM, IDENTIFICATION, CODE_RETOUR)

  ! appel a la routine utilisateur d'ecriture des resultats
  call WRITE_RESULTATS (NOM, IDENTIFICATION, CODE_RETOUR)

end program THEME_Z_1
```

Résultats attendus:

NOM = mt_topo_E_car_sgd

IDENTIFICATION = dans un repere TOPOcentrique Est (convention axe Ox vers l'Est), passage des coordonnees CARtesiennes aux coordonnees Site/Gisement/Distance.

CODE_RETOUR%valeur = 0

CODE_RETOUR%routine = 1002

Exemple n°2 en Fortran 90 portable

 (voir explications dans le volume 3)

Cet exemple montre une utilisation standard de la routine *mz_numero_routine*, en couplage avec l'appel à une routine de la bibliothèque (ici nous avons choisi: *mt_topo_E_car_sgd*).

```
program THEME_Z_2

  use mslib

  ! Declarations liees a l'appel de la routine mt_topo_E_car_sgd
  real(pm_reel), dimension(3)          :: POS_CAR
  type(tm_sgd)                         :: POS_SGD
  type(tm_code_retour)                 :: CODE_RETOUR

  ! Declarations liees a l'appel de la routine mz_numero_routine
  character(len=pm_nom_routine)        :: NOM
  character(len=pm_identification_routine) :: IDENTIFICATION
  type(tm_code_retour)                 :: RETOUR_MZ

  ! Initialisations pour l'appel a mt_topo_E_car_sgd
  POS_CAR(1) = 65._pm_reel
  POS_CAR(2) = 100._pm_reel
  POS_CAR(3) = 35._pm_reel

  ! Appel a la routine de calculs mt_topo_E_car_sgd
  call mt_topo_E_car_sgd (POS_CAR, POS_SGD, CODE_RETOUR)

  ! Appel a la routine mz_numero_routine
  call mz_numero_routine(CODE_RETOUR%routine,          &
                        NOM, IDENTIFICATION, RETOUR_MZ)

  ! Appel a la routine utilisateur d'ecriture de l'entree et des
  ! resultats de mz_numero_routine
  call WRITE_RESULTATS (CODE_RETOUR%routine,          &
                        NOM, IDENTIFICATION, RETOUR_MZ)

end program THEME_Z_2
```

Résultats attendus:

Valeur initialisée lors de l'appel à mt_topo_E_car_sgd :

CODE_RETOUR%routine = 1003

Valeurs initialisées lors de l'appel à mz_numero_routine :

NOM = mt_topo_E_car_sgd

IDENTIFICATION = dans un repere TOPOcentrique Est (convention axe Ox vers l'Est),
passage des coordonnees CARtesiennes aux coordonnees Site/Gise-
ment/Distance.

RETOUR_MZ%valeur = 0

RETOUR_MZ%routine = 1002

Routine mz_val_code_retour

Identification

“Cette routine donne la signification du champ **%valeur** d’un **code retour**”.

Rôle

Cette routine donne la signification du champ “%valeur” d’un code retour. Selon le cas, cette signification peut être physique, mathématique ou informatique.

Avant d’utiliser cette routine, nous vous recommandons de lire les chapitres concernant “les paramètres de la MSLIB Fortran 90” et la “gestion des anomalies” du volume 3 de la documentation utilisateur: “*Caractéristiques principales et conventions d’utilisation de la MSLIB Fortran 90*”.

Séquence d’appel

(voir explications dans le volume 3)

call mz_val_code_retour (valeur, signification, code_retour)

Description des arguments

(voir explications dans le volume 3)

• Entrées obligatoires

integer	valeur	valeur du code retour (obtenu via le champ %valeur de tout code retour).
---------	---------------	--

• Sorties obligatoires

character(len=pm_signification_code_retour)	signification	signification du champ %valeur.
tm_code_retour	code_retour	

Conditions sur les arguments

- La chaîne **signification** est dimensionnée à l’aide du parameter MSLIB: *pm_signification_code_retour*. La déclaration peut donc se faire de la sorte:

```
character(len=pm_signification_code_retour) :: signification
```

Notes d'utilisation

- Compte tenu du surdimensionnement nécessaire de la longueur de la chaîne de caractères **signification**, nous recommandons l'utilisation de la fonction intrinsèque fortran 90 *trim*, après l'appel à *mz_val_code_retour*. Il suffit de faire, par exemple:

```
trim(signification)
```

pour obtenir la chaîne **signification** débarrassée de tous les caractères blancs en fin de chaîne.
- La valeur de **valeur** est fournie par:

```
valeur = code_retour%valeur
```

en sortie de n'importe quelle routine de la bibliothèque.

Références documentaires

Sans objet.

Code retour (voir explications dans le volume 3)

pm_OK (0) : Retour normal.

pm_err_val_code_retour_inconnu (-1806) : Cette valeur de code retour est inconnue dans la bibliothèque.

Exemple n°1 en Fortran 90 portable (voir explications dans le volume 3)

```
program THEME_Z_1

  use mslib

  integer :: VALEUR
  character (len=pm_signification_code_retour) :: SIGNIFICATION
  type(tm_code_retour) :: CODE_RETOUT

  VALEUR = pm_err_para_option

  ! "pm_err_para_option" est un parameter MSLIB utilisé pour la
  ! gestion des anomalies. A chaque fois qu'il manque une entrée
  ! optionnelle pour calculer une sortie optionnelle, la
  ! bibliothèque MSLIB utilise ce parameter. La liste des parameter
```

! utilisé pour la gestion des anomalies est indiquée au niveau du
! paragraphe "Code retour" de la notice d'utilisation de chaque
! routine.

```
call mz_val_code_retour ( VALEUR, SIGNIFICATION, CODE_RETOUR)
```

```
! appel a la routine utilisateur d'ecriture des resultats  
call WRITE_RESULTATS (SIGNIFICATION,CODE_RETOUR)
```

```
end program THEME_Z_1
```

Résultats attendus:

SIGNIFICATION = Compte tenu des sorties optionnelles demandées, il manque des entrees optionnelles.

CODE_RETOUR%valeur = 0
CODE_RETOUR%routine = 1001

Exemple n°2 en Fortran 90 portable

 (voir explications dans le volume 3)

Cet exemple montre une utilisation standard de la routine *mz_val_code_retour*, en couplage avec l'appel à une routine de la bibliothèque (ici nous avons choisi: *mt_topo_E_car_sgd*).

```
program THEME_Z_2  
  
  use mslib  
  
  ! Declarations liees a l'appel de la routine mt_topo_E_car_sgd  
  real(pm_reel), dimension(3) :: POS_CAR  
  type(tm_sgd)                :: POS_SGD, VIT_SGD  
  type(tm_code_retour)         :: CODE_RETOUR  
  
  ! Declarations liees a l'appel de la routine mz_val_code_retour  
  character (len=pm_signification_code_retour) :: SIGNIFICATION  
  type(tm_code_retour)                        :: RETOUR_MZ  
  
  ! Initialisations pour l'appel a mt_topo_E_car_sgd  
  POS_CAR(1) = 65._pm_reel  
  POS_CAR(2) = 100._pm_reel  
  POS_CAR(3) = 35._pm_reel
```

```
! Appel a la routine de calculs mt_topo_E_car_sgd
  call mt_topo_E_car_sgd (POS_CAR, POS_SGD, CODE_RETOUR,      &
                        vit_sgd = VIT_SGD)

! Appel a la routine mz_numero_routine
  call mz_val_code_retour (CODE_RETOUR%valeur,              &
                        SIGNIFICATION, RETOUR_MZ)

! Appel a la routine utilisateur d'ecriture de l'entree et des
! resultats de mz_val_code_retour
  call WRITE_RESULTATS (CODE_RETOUR%valeur,                &
                        SIGNIFICATION, RETOUR_MZ)
end program THEME_Z_2
```

Résultats attendus:

Valeur initialisée lors de l'appel à mt_topo_E_car_sgd :

CODE_RETOUR%valeur = -1801

Valeurs initialisées lors de l'appel à mz_numero_routine :

SIGNIFICATION = Compte tenu des sorties optionnelles demandées, il manque des
entrees optionnelles.

RETOUR_MZ%valeur = 0

RETOUR_MZ%routine = 1001