# Liste des routines utilitaires de Code_Aster

**Résumé :**

Nous donnons dans ce document une liste d'environ 500 utilitaires de Code_Aster. Pour chacun d'eux, on donne une très brève description de sa fonction.

Pour en savoir plus sur la fonction des utilitaires, il faut consulter l'entête de la subroutine (fortran en général) qui lui correspond.



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

# Table des Matières

1 Mode d'emploi....4
2 Liste des paquets....4
3 Liste des utilitaires....6
3.1 CARA_ELEM....6
3.2 CHAMP....6
3.3 CHARGE....8
3.4 COMPOR....8
3.5 DEBUG et ERREUR....9
3.6 DIVERS....10
3.7 ELT_COQUE....10
3.8 ELT_ISO....11
3.9 ELT_POUTRE....12
3.10 ELT_TOUS....13
3.11 ENVIMA....13
3.12 FICHIER....14
3.13 FONCTION....14
3.14 GRANDEUR....15
3.15 INITEL....16
3.16 JEVEUX....16
3.17 LIGREL....17
3.18 MAILLAGE....18
3.19 MATERIAU....19
3.20 MATR_ASSE....21
3.21 MESSAGE....22
3.22 MEMOIRE....22
3.23 MPI....22
3.24 MPLEIN....23
3.25 NUME_DDL....23
3.26 PREPOST....24
3.27 REPERE....24
3.28 Resuelem....25
3.29 RESULTAT....25
3.30 RUPTURE....26
3.31 SD....27
3.32 SUPERVISEUR....27
3.33 TABLE....28
3.34 TITRE....28
3.35 TYPE_FORTRAN....28



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

3.36 VARI_COM.....30



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

# 1 Mode d'emploi

Ce document doit permettre aux développeurs de nouvelles fonctionnalités dans le Code_Aster de répondre à la question suivante :

“existe-t-il dans le code une routine qui fasse ce dont j’ai besoin ? “.

La réponse (si elle est positive) sera le nom de cette routine ainsi qu’une brève description de celle-ci (2 lignes de français). Pour utiliser cette routine avec profit, il faudra soit consulter d’autres documents (D5 ou D6), soit si ces routines ne sont pas documentées (cas assez fréquent), consulter le texte (et les commentaires) de son source.

Pour pouvoir parcourir rapidement la liste de ces utilitaires, nous les avons classées en “paquets”. On peut parfois retrouver une même routine dans plusieurs paquets. Les paquets sont associés aux types des objets manipulés par les routines. Par exemple, une routine de résolution d’un système assemblé se retrouvera dans les paquets : matr_asse et cham_no. Les objets manipulés dans ces routines ne sont pas toujours des Structures de Données Aster ; on peut aussi manipuler des variables fortran : scalaires ou tableaux. Nous définissons dans le tableau du paragraphe suivant les “types” d’objets considérés par la suite.

Comment faire vivre ce document ? Ce document utile (je l’espère) à la communauté des développeurs doit vivre par eux. L’auteur leur demande donc de lui communiquer (par mél si possible) leurs remarques : Quels sont les utilitaires à ajouter à ce document (nouveaux ou oubliés) ? Quels sont les utilitaires dont la fonction n’est pas assez claire (ou erronée) ? Quels sont les utilitaires à supprimer de la liste (supprimés du code ou à déconseiller) ?

# 2 Liste des paquets

| PAQUET | définition |
|---|---|
| BLAS/LAPACK | Ce “paquet” a été supprimé. Les bibliothèques BLAS et LAPACK sont maintenant des “pré-supposés” pour Code_Aster. http://www.netlib.org/lapack/lug/lapack_lug.html |
| CARA_ELEM | SD cara_elem |
| CHAMP | Champ : cham_no, cham_elem, carte, cham_no_s, cham_elem_s |
| CHARGE | SD char_meca, SD char_acou, SD char_ther, SD char_cine, SD liste_rela, SD liste_charge |
| COMPOR | lois de comportement |
| DEBUG | “debogage”, routines d’aide pour les développeurs |
| DIVERS | date, heure, mesure des temps d’exécution, ... |
| ELT_COQUE | éléments finis de type “coque” |
| ELT_ ISO | éléments finis “iso-paramétriques” |
| ELT_POUTRE | éléments finis de type “POUTRE” ou “tuyau” |
| ELT_TOUS | éléments finis “général” (ce qui est manipulé par les routines TEXXXX) |
| ENVIMA | Constantes dépendant de la machine d’exécution : IRIX, SOLARIS, ... |
| FICHIER | fichiers d’entrée / sortie |
| FONCTION | Fonction : FUNCTION de Fortran, SD fonction, SD nappe |
| GRANDEUR | grandeur physique : noms des composantes, entiers codés, ... |
| JEVEUX | Objets JEVEUX : objets simples ou collections |



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

| | |
|---|---|
| INITEL | Initialisation des ELREFE. |
| LIGREL | SD ligrel |
| MAILLAGE | SD maillage |
| MATERIAU | Caractéristiques d'un matériau : SD mater, SD cham_mater, SD materc (matériau "codé utilisé dans les routines TEXXXX) |
| MATR_ASSE | SD matr_asse, SD solveur |
| MEMOIRE | Accès aux différents paramètres permettant de mesurer la mémoire consommée : valeurs données par le système (VmPeak, VmSize, etc.), compteurs actualisés par le gestionnaire de mémoire JEVEUX (limite mémoire, cumul des allocations dynamiques, cumul des objets utilisés, etc.), consommation du superviseur. |
| MESSAGE | messages d'erreur ou d'alarme (UTMESS) ou messages de type "INFO" (mot clé INFO des commandes) |
| MPI | parallélisme MPI |
| MPLEIN | matrice "pleine" ; i.e. tableaux Fortran à 2 indices |
| NUME_DDL | Numérotation des inconnues d'un système linéaire : SD nume_ddl, SD prof_chno, SD nume_equa, SD stockage |
| PREPOST | "pré" et "post" traitement graphiques : GIBI, IDEAS |
| REPERE | changement de repère, rotation |
| resuelem | Matrices (ou vecteurs) élémentaires : SD resuelem, SD vect_elem, SD matr_elem |
| RESULTAT | SD resultat |
| RUPTURE | mécanique de la rupture |
| SD | Structure de données du Code_Aster (i.e. ensemble d'objets JEVEUX) |
| SUPERVISEUR | communication des commandes avec le superviseur : routines GETXXX |
| TABLE | SD table |
| TITRE | titre (ou sous-titre) associé à une structure de données |
| TYPE_FORTRAN | Types Fortran : REAL, COMPLEX, INTEGER, CHARACTER et vecteurs Fortran de ces types |
| VARI_COM | SD vari_com |

**Tableau 2-1**



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

# 3 Liste des utilitaires

## 3.1 CARA_ELEM

| | |
|---|---|
| RECUDE | récupération du diamètre extérieur d'une structure tubulaire a partir des données fournies par un concept de type cara_elem |

**Tableau 3.1-1**

## 3.2 CHAMP

| | |
|---|---|
| ALCART | allouer une SD carte [D6.10.01] |
| ALCHML | créer une SD cham_elem "vierge" |
| ASASVE | assembler les vecteurs élémentaires provenant des charges |
| ASCAVC | produire le second membre du à des charges cinématiques |
| ASCOVA | combiner des vecteurs assemblés |
| ASSVEC | Assembler des vecteurs élémentaires pour en faire un second membre (SD cham_no) |
| BARYCH | combinaison linaire de cham_no ou de cham_elem |
| CALCUL | faire les calculs élémentaires correspondant à une OPTION sur les éléments d'une SD ligrel. |
| CALVCI | Calcul du second membre du à des charges cinématiques |
| CARCES | transformer une SD carte en une SD cham_elem_s |
| CARCOMP | comparer deux cartes |
| CELFPG | récupérer la liste des noms des familles de PG d'un cham_elem (ELGA) |
| CELCES | transformer une SD cham_elem en SD cham_elem_s |
| CELVER | vérifier qu'une SD cham_elem possède certaines propriétés |
| CESCAR | transformer un cham_elem en carte |
| CESCES | changer la discrétisation d'un cham_elem_s (ELNO/CART/ELGA) |
| CESCNS | transformer une SD cham_elem_s en une SD cham_no_s |
| CESCRE | créer une SD cham_elem_s |
| CESCRM | créer un cham_elem_s en prenant un autre cham_elem_s comme modèle |
| CESEXI | tester l'existence d'une CMP d'un point d'une maille d'une SD cham_elem_s |
| CESRED | "réduire" une SD cham_elem_s sur une liste de mailles et/ou une liste de CMPS. |
| CHLIGR | convertir un cham_elem en un autre cham_elem sur un autre ligrel. |
| CHPCHD | changer le support géométrique d'un champ (NOEU/ELNO/ELGA/CART) |
| CHPNUA | transformer un cham_no en une SD nuage pour pouvoir le projeter sur un autre maillage (méthode 'NUAG_DEG_0/1') |
| CHSFUS | fusionner plusieurs SD cham_elem_s (ou SD cham_no_s) (par addition ou surcharge des CMPS) |
| CHSUT1 | modifier les noms de la grandeur et des CMPS d'un cham_no_s ou d'un cham_elem_s |



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

| | |
|---|---|
| CMPCHA | fournir le nombre, le nom des composantes d'un champ et la correspondance "composante champ⇔composante grandeur" |
| CNOCNS | transformer une SD cham_no en SD cham_no_s |
| CNOMAX | Calcule le max de la norme L2 du déplacement DX DY DZ pour un cham_no_depl_R. |
| CNSCES | transformer une SD cham_no_s en une SD cham_elem_s |
| CNSCNO | transformer une SD cham_no_s en SD cham_no |
| CNSCRE | créer une SD cham_no_s |
| CNSPRJ | projeter un cham_no_s sur une autre maillage |
| CNSRED | "réduire" une SD cham_no_s sur une liste de nœuds et/ou une liste de CMPS. |
| CRCHNO | création d'une SD cham_no |
| CRCNCT | créer un cham_no constant sur tous les nœuds d'un maillage. |
| CSMBGG | calcul de la contribution au second membre des ddl imposés lorsqu'ils sont traités par élimination (SD char_cine) |
| IRCH19 | imprimer un champ (cham_no ou cham_elem) |
| IRCHMD | impression d'un champ dans un fichier MED (peu d'arguments) |
| MAJOUR | mise à jour d'un champ de déplacement suite à un incrément en tenant compte d'éventuelles grandes rotations |
| MCCONL | tenir compte du conditionnement des Lagrange sur le second membre |
| MCMULT | effectue le produit d'une matrice par N vecteurs (si complexe) |
| MECACT | créer 1 SD carte constante [D6.10.01] |
| MECARA | récupération du nom des champs dans un cara_elem |
| MEGEOM | récupération du champ de géométrie dans 1 modèle ou 1 liste de charges |
| MEMAXM | extraire le "max" ou le "min" d'une CMP sur un ensemble d'éléments d'un cham_elem/ELEM |
| MEMOY | calculer la moyenne (pondérée) d'une CMP sur un ensemble d'éléments d'un cham_elem |
| MESOMM | faire la somme (sur les mailles d'un maillage) des valeurs d'un cham_elem |
| MRCONL | tenir compte du conditionnement des termes de Lagrange sur le second membre |
| MRMULT | effectue le produit d'une matrice par N vecteurs (cas réel) |
| NOCART | noter un couple (entitée_affectée, grandeur) dans une SD carte [D6.10.01] |
| NUACHP | transformer une SD nuage en un cham_no |
| PJ2DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJEFPR (cas 2D) |
| PJ3DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJEFPR (cas 3D) |
| PJ4DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJEFPR (cas 2,5D) |
| PJ6DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJEFPR (cas 1,5D) |
| PJEFPR | projeter un cham_no sur un autre maillage (méthode 'COLOCATION') |
| PRONUA | projeter une SD nuage sur un autre maillage (méthode 'NUAG_DEG_0/1') |
| SDCHGD | changer le type R / C d'un cham_no ou d'un cham_elem. |
| TECART | "terminer" une SD carte : gérer une "surcharge fine" des CMPS affectées [D6.10.01] |
| UTCH19 | extraire une valeur (CMP) d'une SD cham_elem |
| UTCHDL | récupérer le numéro d'une CMP dans une SD cham_elem |



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

| | |
|---|---|
| UTNCMP | récupère le nombre et le noms des CMPS d'un champ |
| VTAXPY | effectuer l'opération $Y = a.X + Y$ sur tous les degrés de liberté de deux cham_no |
| VLAXPY | effectuer l'opération $Y = a.X + Y$ sur les degrés de liberté de Lagrange de deux cham_no |
| VTCMBL | Combinaison linéaire de cham_no |
| VTCOPY | Copie des valeurs d'1 cham_no dans un autre cham_no ayant éventuellement une autre numérotation. |
| VTCREA | Allocation d'un cham_no |
| VTCREB | Allocation d'un cham_no |
| VTCREM | Allocation d'un cham_no |
| VTGPLD | ajoute un champ de déplacement à un champ de géométrie : $X2 = X1 + a.U$ où $a$ est un réel |
| ZERLAG | mettre à zéro les DDLS de Lagrange dans une SD cham_no |
| ZEROSD | détermine si un champ est complètement "vierge" (0.) |

**Tableau 3.2-1**

## 3.3 CHARGE

| | |
|---|---|
| AFLRCH | écrire dans une charge les relations linéaires d'une SD liste_rela |
| AFRELA | écrire une relation linéaire dans une SD liste_rela |
| ASCAVC | produire le second membre du à des charges cinématiques |
| ASSCHC | modifier une matr_asse pour tenir compte de l'élimination des ddls contraints par des SD char_cine |
| CALVCI | Calcul du second membre du à des charges cinématiques |
| COCHRE | vérifie sur une liste de charges la présence d'une seule charge répartie |
| CORICH | gérer un éventuel lien entre un champ et une charge pour pouvoir lui appliquer plus tard une "FONC_MULT" |
| CSMBGG | calcul de la contribution au second membre des ddl imposés lorsqu'ils sont traités par élimination (SD char_cine) |
| MEDOM1 | saisie et vérification de la cohérence des données mécaniques du problème |
| MEDOME | saisie et vérification de la cohérence des données mécaniques du problème |
| MEGEOM | récupération du champ de géométrie dans 1 modèle ou 1 liste de charges |
| NMDOME | Vérification des données d'un problème non-linéaire mécanique |

**Tableau 3.3-1**

## 3.4 COMPOR

| | |
|---|---|
| EXICP | Renvoie .TRUE. si au moins une maille d'un maillage ou d'une liste de mailles donnée en entrée est associée à un élément-fini modélisé en contraintes planes |
| LCDEVI | calcule le déviateur d'un tenseur d'ordre 3 |



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

| | |
|---|---|
| LCDIMA | calcule la différence de 2 matrices pleines carrées |
| LCDIVE | calcule la différence de 2 vecteurs de réels |
| LCEQVE | copie d'un vecteur de réels |
| LCEQVN | copie d'un vecteur de réels |
| LCHYDR | calcule la partie sphérique d'un tenseur |
| LCINMA | initialisation d'une matrice carrée |
| LCINVE | initialisation d'un vecteur réel |
| LCIV2E | calcul du second invariant d'un tenseur de déformation |
| LCIV2S | calcul du second invariant d'un tenseur de contrainte |
| LCNRTE | norme du second invariant d'un tenseur de déformation |
| LCNRTS | norme du second invariant d'un tenseur de contrainte |
| LCOPIL | opérateur de souplesse pour un comportement élastique linéaire |
| LCOPLI | opérateur de rigidité pour un comportement élastique linéaire |
| LCPRMM | produit de 2 matrices carrées |
| LCPRMV | produit matrice carrée * vecteur |
| LCPRSC | produit scalaire de 2 vecteurs |
| LCPRSM | multiplie une matrice carrée par un scalaire |
| LCPRSV | multiplie un vecteur par un scalaire |
| LCPRTE | produit tensoriel de 2 vecteurs |
| LCQEQV | teste l'égalité de 2 vecteurs |
| LCSOMA | calcule la somme de 2 matrices carrées |
| LCSOVE | calcule la somme de 2 vecteurs |
| MATINV | inverse les matrices de dimension inférieure ou égale à 3 |
| NMDORC | Traitement du mot clé facteur COMPORTEMENT |

**Tableau 3.4-1**

## 3.5 DEBUG et ERREUR

| | |
|---|---|
| Note : | Pour comparer l'exécution de 2 versions du code donnant des résultats différents (par exemple debug et nodebug), on peut déclencher des impressions très utiles et pas trop volumineuses en positionnant DBG=.TRUE. dans la routine calcul.f |
| imptou | imprimer sur listing la "signature" de tous les objets JEVEUX présents sur une base |
| jeimpm | imprime la segmentation de la mémoire [D6.02.01] |
| jeimpr | impression du répertoire d'une ou plusieurs classes [D6.02.01] |
| jeprat | impression des objets système ou des objets attribut de collection [D6.02.01] |
| jeundf | mettre à "undef" un objet JEVEUX |
| jxveri | teste la cohérence de la segmentation mémoire de JEVEUX [D6.02.01] |
| dbgobj | Imprimer dans un fichiers 5 nombres caractérisant un objet JEVEUX : contenu + certains attributs |
| uttcpu | mesure le temps CPU (user et système) consommé entre 2 instructions [D6.01.0] |



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

| | |
|---|---|
| memver | vérifier qu'il n'y a pas de "fuite" mémoire dans un programme fortran |
| memres | retourne la valeur "max" de la zone mémoire que l'on peut encore allouer |
| mempid | retourne les valeurs de VmData et VmSize de la mémoire utilisée par le processus |
| cheksd | vérifie la cohérence d'une structure de donnée |
| irchmd | Impression d'un champ dans un fichier MED |
| ASSERT | Macro (en majuscules) déclenchant un message d'erreur si la condition en argument n'est pas vérifiée. Prendre l'habitude de faire ASSERT(i.eq.0) et non if (i.ne.0) ASSERT(.false.). Le message contient le texte de la condition ainsi que le nom du fichier et le numéro de ligne où cela s'est produit. |
| Autres macros | Plusieurs autres macros ont été définies (dans assert.h) afin de faciliter l'écriture de la vérification des arguments optionnels.<br>On a repris une syntaxe proche des *règles* utilisées dans le catalogue de commande.<br>Acceptant un nombre fixe d'arguments, elles ont des variantes pour 2, 3 ou 4 arguments. Par exemple : UN_PARMI2 ou ENSEMBLE4.<br>Les règles retournent un booléen que l'on peut utiliser dans un ASSERT ou bien pour construire une règle plus compliquée. |
| absent | L'inverse de present(). Retourne .true. si l'argument optionnel n'est pas présent. |
| UN_PARMI2<br>UN_PARMI3<br>UN_PARMI4 | Retourne .true. si un et seul argument parmi ceux fournis est présent. **Exactement 1**. |
| AU_MOINS_UN2<br>AU_MOINS_UN3<br>AU_MOINS_UN4 | Retourne .true. si un ou plusieurs arguments parmi ceux fournis sont présents.<br>**De 1 à N.** |
| EXCLUS2<br>EXCLUS3<br>EXCLUS4 | Retourne .true. si au plus un argument parmi ceux fournis est présent.<br>**0 ou 1.** |
| ENSEMBLE2<br>ENSEMBLE3<br>ENSEMBLE4 | Retourne .true. si aucun ou tous les arguments fournis sont présents.<br>**0 ou N.** |

**Tableau 3.5-1**

## 3.6 DIVERS

| | |
|---|---|
| JJMMAA | écriture du nom de l'auteur et de la date de création de ce fichier |
| UTTCPU | mesure le temps CPU (user et système) consommé entre 2 instructions [D6.01.03] |

**Tableau 3.6-1**

## 3.7 ELT_COQUE

| | |
|---|---|
| CQ3D2D | calcul des coordonnées 2D d'un triangle ou d'un quadrangle à partir de ses coordonnées 3D passage dans le repère du plan du triangle ou du quadrangle avec teta=angle entre l'axe X et le cote A1A2 |
| DKQBF | matrice B au point qsi eta pour l'élément DKQ |
| DKTBF | matrice B au point qsi eta pour l'élément DKT |
| DSQBFA | matrice BFA au point qsi eta pour l'élément DSQ |



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

| | |
|---|---|
| DSQBFB | matrice BFB au point qsi eta pour l'élément DSQ |
| DSQCIS | matrices BCB et BCA au point qsi eta pour l'élément DSQ |
| DSQDIS | matrice AN du cisaillement pour l'élément DSQ |
| DSTBFA | matrice BFA au point qsi eta pour l'élément DST |
| DSTBFB | matrice BFB au point qsi eta pour l'élément DST |
| DSTCIS | Matrices BCA et AN du cisaillement pour l'élément DST |
| DXBSIG | calcul des forces internes B*SIGMA aux nœuds de l'élément dues au champ de contraintes SIGMA défini aux points d'intégration pour les éléments : DST, DKT, DSQ, DKQ et Q4G |
| DXEFGT | efforts généralisés d'origine thermique aux points d'intégration pour les éléments COQUE et DST, DKT, DSQ, DKQ et Q4G |
| DXEFRO | passage des efforts ou déformations généralisées du repère intrinsèque de l'élément au repère local de la COQUE |
| DXMATE | calcul des matrices de rigidité de flexion, membrane , couplage membrane-flexion et cisaillement pour un matériau isotrope ou multicouche |
| DXMATH | calcul des matrices de rigidité de flexion, membrane , couplage membrane-flexion et cisaillement pour un matériau isotrope ou multicouche |
| DXQBM | matrice BM membrane au point qsi eta pour éléments DKQ et DSQ |
| DXQPGL | construction de la matrice de passage global –> local pour une maille triangle DKQ ou DSQ |
| DXREPE | calcul des matrices T1VE et T2VE de passage d'une matrice du repère de la variété au repère élément et T2VE inverse de T2EV pour toutes les options de post traitement COQUE |
| DXROEP | Récupération de la masse volumique du matériau et épaisseur de la plaque |
| DXSIRO | passage des contraintes ou déformations du repère intrinsèque de l'élément au repère local de la COQUE |
| DXTBM | matrice BM en membrane pour les éléments DKT et DST |
| DXTPGL | construction de la matrice de passage global –> local pour une maille triangle DKT ou DST |
| GQUAD4 | grandeurs géométriques sur le QUAD4 |
| GTRIA3 | paramétrage des éléments DKT (TRIA3) |
| JQUAD4 | jacobien à un point sur le QUAD4 |
| Q4GBC | matrice BC au point qsi eta pour l'élément Q4G |
| COQREP | calcul de la matrice de passage du repère de l'élément au repère donné en argument. Le repère est caractérisé par 2 angles. |

**Tableau 3.7-1**

## 3.8 ELT_ISO

| | |
|---|---|
| BMATMC | calculer la matrice B reliant les déformations du premier ordre aux déplacements pour un point d'intégration |
| BSIGMC | calculer les forces internes B*sigma aux nœuds de l'élément |
| BTDBMC | calculer le produit Bt*D*B donnant la matrice de rigidité élémentaire |
| CONNEC | initialisation des éléments iso-P2 |
| DFDM1D | calcul des dérivées des fonctions de forme par rapport a un élément courant en un point de gauss |



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

| | |
|---|---|
| | pour les éléments 1D |
| DFDM2D | calcul des dérivées des fonctions de forme par rapport a un élément courant en un point de gauss pour les éléments 2D |
| DFDM3D | calcul des dérivées des fonctions de forme par rapport a un élément courant en un point de gauss pour les éléments 3D |
| DMATMC | calcul de la matrice de HOOKE pour les éléments isoparamétriques pour des matériaux isotrope, orthotrope et isotrope transverse |
| DPFCH3 | calcul des dérivées des fonctions de forme par rapport a un élément courant en un point de gauss pour les éléments 3D non isoparamétriques |
| EPSIMC | construction du vecteur des déformations initiales définies en chaque point d'intégration a partir des données utilisateur pour l'élément courant |
| EPSTMC | calcul des déformations thermiques pour les éléments isoparamétriques |
| EPSVMC | calcul des déformations mécaniques (i.e. eps_totales - eps_thermiques) aux points d'intégration pour les éléments isoparamétriques |
| SUBACV | calcul de la base contra-variante (dimension 3) |
| SUMETR | calcul du tenseur métrique (2×2) et de son jacobien |
| VFF2DN | calcule la normale et le poids d'un point de Gauss d'un élément SEG en 2D |
| VFF3D | calcule le poids d'un point de Gauss d'un élément SEG en 3D. |
| DFDMIP | calcul des dérivées des fonctions de forme et du jacobien 2D,AXI,3D |
| NMGEOM | calcul des éléments cinématiques en un point de Gauss (éventuellement en grande transformations) |
| NMMABU | calcul de la matrice B (DEPS = B.DU) |
| NMEPSI | calcul des déformations cinématiques 2D,AXI,3D,GRAND |
| NMEPSB | calcul des déformations régularisées et leurs gradients 2D,3D |

**Tableau 3.8-1**

## 3.9 ELT_POUTRE

| | |
|---|---|
| CARCOU | récupérer la géométrie des éléments tuyau (coude) |
| DEELPO | récupération du diamètre extérieur d'un élément de POUTRE |
| FUN1 | calcul de l'aire ou de la constante de torsion équivalente d'une POUTRE droite à section variable sous l'hypothèse de variation linéaire des coordonnées |
| FUN2 | calcule le moment d'inertie équivalent d'une POUTRE droite à section variable sous l'hypothèse de variation linéaire des coordonnées |
| GDFINT | pour un élément de POUTRE en grand déplacement, calcule la contribution du point de gauss numéro KP aux forces internes |
| GDJRG0 | pour un élément de POUTRE en grand déplacement, calcule, aux points de gauss, le jacobien et la matrice de rotation des axes principaux d'inertie en position de référence, par rapport aux axes de coordonnées généraux |
| GDMB | pour un élément de POUTRE en grand déplacement, calcule la contribution du déplacement du nœud ne à la matrice de déformation B au point de gauss KP |
| JPD1FF | calcul des fonctions de forme de déformations généralisées pour l'élément POUTRE 6 ddl à 3 |



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

| | |
|---|---|
| | points de gauss |
| JSD1FF | calcul des fonctions de forme de déformations généralisées pour l'élément POUTRE 7 ddl à 3 points de gauss |
| POEFGR | calcul du vecteur élémentaire effort généralisé réel, pour les éléments de POUTRE d'Euler et de Timoshenko |
| POMASS | calcule la matrice de masse des éléments de POUTRE |
| PORIGI | calcule la matrice de rigidité des éléments de POUTRE |
| POUEX7 | Traitement de l'excentricité des éléments de POUTRE |
| PTENCI | Calcule l'énergie cinétique pour les éléments de POUTRE, DISCRET et BARRE |
| PTENPO | Calcule l'énergie de déformation pour les éléments de POUTRE, DISCRET et BARRE |
| PTKA01 | calcule la matrice de raideur de l'élément de POUTRE droite à section constante |
| PTKA02 | calcule la matrice de raideur de l'élément de POUTRE droite à section variable |
| PTKA10 | calcule la matrice de raideur de l'élément de POUTRE courbe |
| PTKA21 | calcule la matrice de raideur de l'élément de POUTRE droite à section constante à 7 ddl par nœud |
| PTMA01 | calcule la matrice de masse de l'élément de POUTRE droite |
| PTMA10 | calcule la matrice de masse de l'élément de POUTRE courbe |

**Tableau 3.9-1**

## 3.10 ELT_TOUS

| | |
|---|---|
| TEATTR | pour récupérer la valeur d'un attribut associé à un TYPE_ELEMENT |
| LTEATT | pour tester si un attribut a une valeur donnée (sur-couche de TEATTR) |
| FGEQUI | calcul des grandeurs équivalentes en contrainte et déformation |
| ISELLI | Renvoie TRUE si le TYPE_ELEMENT est linéaire |
| ISMALI | Renvoie TRUE si le TYPE_MAILLE est linéaire |
| JEVECH | Récupérer l'adresse du champ local correspondant à un paramètre |
| ELREFE_INFO | récupérer les adresses des tableaux contenant les valeurs des fonctions de forme (et de leurs dérivées) sur une famille de points d'intégration (+ dimensions, matrice de passage Gauss → Nœud) |
| PPGAN2 | passage des valeurs aux points de gauss aux valeurs aux nœuds sommets et aux nœuds milieux par valeur moyenne |
| UTELVF | récupérer les valeurs des fonctions de forme sur une famille de points d'intégration quand on n'est pas dans une routine de calcul élémentaire. (Sinon, il faut utiliser ELREFE_INFO). |
| TECACH | récupérer les caractéristiques d'un champ_local : adresse, longueur, ... |
| TECAEL | récupérer les caractéristiques d'un élément_fini : nom de la maille associée, ... |

**Tableau 3.10-1**

## 3.11 ENVIMA



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

| | |
|---|---|
| ISMAEM | l'entier maximal possible [D6.01.01] |
| R8DEPI | la valeur réelle 2*PI [D6.01.01] |
| R8DGRD | Conversion degré / radian [D6.01.01] |
| R8GAEM | gamme : nombre tel que gamme**2 soit représentable en machine [D6.01.01] |
| R8MAEM | réel le plus grand [D6.01.01] |
| R8MIEM | réel le plus petit [D6.01.01] |
| R8NNEM | Un réel NaN [D6.01.01] |
| R8PI | Retourne la valeur réelle PI. [D6.01.01] |
| R8PREM | précision relative des nombres réels [D6.01.01] |
| R8RDDG | Conversion radian / degré [D6.01.01] |
| R8VIDE | la valeur d'un réel "impossible" (peut servir à tester si un réel a été affecté ou non) [D6.01.01] |

**Tableau 3.11-1**

## 3.12 FICHIER

| | |
|---|---|
| ULDEFI | définit l'association unité logique - nom local (FICHIER) - nom fichier (NOM_SYSTEME), fait appel à ULOPEN pour les fichiers ASCII |
| ULOPEN | effectue l'association, l'open fortran et le positionnement pour les fichiers ASCII |
| ULCLOS | effectue la libération et le "close" pour les fichiers ASCII |
| ULPOSI | se positionne (NEW, OLD, APPEND) dans le fichier de type ASCII (en FORTRAN il n'est malheureusement pas possible de se positionner lors de l'OPEN et les extensions à la norme ne sont pas toujours admises sur toutes les plates-formes) |
| ULINIT | initialise la structure de données stockée dans les communs |
| ULIMPR | imprime le contenu de la structure de données |
| ULISOP | renvoie un entier non nul si l'unité logique a été affectée et si le fichier associé est ouvert. Le nom local est aussi renvoyé. |
| ULNUME | renvoie un numéro d'unité logique inutilisé entre 1 et 99 |
| IUNIFI | La routine IUNIFI destinée à récupérer le numéro d'unité logique associé à un nom local (FICHIER) est conservée pour assurer la compatibilité, mais s'appuie maintenant sur les nouvelles structures de données. |

**Tableau 3.12-1**

## 3.13 FONCTION

| | |
|---|---|
| nota : | lorsque l'on a besoin d'une fonction "nulle" (par exemple comme fonction par défaut dans des commandes AFFE_XXXX_F), on peut toujours utiliser la fonction "&FOZERO" créée par la routine debut.f et donc disponible à tout instant. |
| FOATTR | surcharge les attributs d'un concept de type fonction |
| FOC1MA | calculer des maxima d'un concept de type fonction |
| FOCRCH | récupération d'une fonction dans une structure tran_gene pour un nœud de choc |



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

| | |
|---|---|
| FOCSTE | création d'un concept de type fonction constante |
| FODERI | obtention de la valeur de la fonction et de sa dérivée pour une fonction linéaire par morceau |
| FOEC2F | écriture des couples (paramètre, résultat) d'un concept de type fonction |
| FOEC2N | écriture des valeurs (paramètre, fonction) d'une nappe |
| FOIMPR | impression d'un concept de type fonction sur un fichier |
| FOINT0 | remise a zéro du common utilisé par la routine foint2 |
| FOINTE | évaluer une fonction (i.e. calcul de $f(x,y,z,...)$ ) |
| FOINTN | interpolation dans les nappes |
| FOINTR | interpolation-extrapolation de toute une fonction |
| FOLOCX | recherche de la place de x dans le vecteur ordonné croissant |
| FONBPA | récupérer la liste des noms des paramètres d'une SD fonction |
| FOPRO1 | récupérer les prolongements et type d'interpolation d'un concept de type fonction |
| FOZERO | créer une fonction nulle |
| TBEXFO | extraire une fonction d'une table en désignant 2 colonnes en vis-à-vis. [D6.06.01] |
| UTTRIF | tri d'une fonction par abscisses croissantes |
| ZERODI | Résolution d'une fonction réelle scalaire par dichotomie (une itération : nécessite de gérer soi-même l'algorithme itératif) |
| ZEROD2 | Résolution d'une fonction réelle scalaire par dichotomie avec traitement des dérivées (une itération : nécessite de gérer soi-même l'algorithme itératif) |
| ZEROCO | Résolution d'une fonction réelle scalaire par une méthode de corde (une itération : nécessite de gérer soi-même l'algorithme itératif) |
| ZEROG2 | Résolution d'une fonction réelle scalaire par approximations P2 (une itération : nécessite de gérer soi-même l'algorithme itératif) |
| ZEROFR | Résolution d'une fonction réelle scalaire avec recherche possible d'intervalle initial: routine générale, qui renvoie vers ZEROFC, ZEROFO, ZEROF2, ZEROFB |
| ZEROFC | Résolution d'une fonction réelle scalaire par une méthode corde (sécante) |
| ZEROFO | Résolution d'une fonction réelle scalaire par une méthode corde (sécante) combinée avec une dichotomie conditionnelle (méthode de Dekker) |
| ZEROF2 | Résolution d'une fonction réelle scalaire par une méthode corde (sécante) combinée avec une dichotomie toutes les 3 itérations (variante de la méthode de Dekker) |
| ZEROFB | Résolution d'une fonction réelle scalaire par la méthode de Brent |
| ZEROP2 | Résolution d'un polynôme de degré 2 |
| ZEROP3 | Résolution d'un polynôme de degré 3 |
| ZEROPN | Résolution d'un polynôme de degré n par la méthode « Companion Matrix Polynomial » |

**Tableau 3.13-1**

## 3.14 GRANDEUR

| | |
|---|---|
| DEC2PN | décoder un entier codé en base 2 |
| DGMODE | trouver le descripteur grandeur associé à un mode local de carte, cham_no, ou cham_elem, sous forme "iden" |



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

| | |
|---|---|
| DIGDEL | Récupérer le nombre de scalaires représentant la grandeur pour un mode local |
| EXISDG | décoder un entier codé |
| IPOSDG | rend la position d'1 composante dans un descripteur grandeur DG |
| IRCCMP | trouver le nombre et les noms des composantes d'une liste présente dans une grandeur |
| ISCODE | coder un entier codé sur les 30 premières puissances de 2 ( pas de puissance 0) |
| ISDECO | décoder un entier codé sur les 30 premières puissances de 2 (pas de décodage sur puissance 0) |
| ISGECO | gérer l'addition ou la soustraction des deux entier codés sur les 7 premières puissances entières de 2 |
| NBCMP | retourne le nombre d'entiers codés pour une grandeur |
| NBEC | retourne le nombre d'entiers codés pour une grandeur |
| SCALAI | retourne le type d'une grandeur : réel, entier, character,.. |
| VERIGD | vérifier la cohérence d'une liste de CMPS d'une grandeur |
| UTCMP1<br>UTCMP2<br>UTCMP3 | Récupérer des infos derrière un mot clé NOM_CMP |

**Tableau 3.14-1**

## 3.15 INITEL

| | |
|---|---|
| ELRACA | dimensions diverses d'un ELREFE |
| ELRAGA | description des familles d'intégration d'un ELREFE |
| ELRFVF | fonctions de formes d'un ELREFE |
| ELRFDF | dérivées des fonctions de formes d'un ELREFE |

**Tableau 3.15-1**

## 3.16 JEVEUX

| | |
|---|---|
| CHLICI | vérifier qu'une chaîne de caractères est licite au sens de JEVEUX |
| IMPTOU | imprimer sur listing la "signature" de tous les objets JEVEUX présents sur une base |
| JACOPO | recopier un morceau d'objet JEVEUX dans un autre |
| JEAGCO | Recopier une collection JEVEUX dans une autre plus grande |
| JECCTA | « Retasser » une collection contiguë qui a été surdimensionnée lors de sa création (attribut LONT) |
| JECREC | créer une collection JEVEUX [D6.02.01] |
| JECREO | créer un objet simple JEVEUX [D6.02.01] |
| JECROC | déclarer un nouvel objet dans une collection (ou dans un répertoire de nom) [D6.02.01] |
| JEDEMA | décrémente la marque et libère les objets marqués [D6.02.01] |
| JEDETC | destruction d'un ensemble d'objets JEVEUX [D6.02.01] |
| JEDETR | détruire un objet JEVEUX (simple ou collection) [D6.02.01] |
| JEDISP | renvoie dans un tableau les longueurs max disponibles [D6.02.01] |



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

| | |
|---|---|
| JEDUPC | dupliquer un ensemble d'objets JEVEUX [D6.02.01] |
| JEDUPO | dupliquer 1 objet JEVEUX [D6.02.01] |
| JEDUP1 | dupliquer 1 objet JEVEUX sans se préoccuper de son existence |
| JEECRA | affectation d'un attribut d'un objet JEVEUX [D6.02.01] |
| JEEXIN | teste l'existence d'un objet JEVEUX [D6.02.01] |
| JEIMPA | imprimer les attributs d'un objet JEVEUX [D6.02.01] |
| JEIMPM | imprime la segmentation de la mémoire [D6.02.01] |
| JEINFO | retourne des informations sur l'utilisation de la mémoire JEVEUX |
| JEIMPO | imprimer un objet JEVEUX [D6.02.01] |
| JEIMPR | impression du répertoire d'une ou plusieurs classes [D6.02.01] |
| JELIBE | libérer un objet JEVEUX de la mémoire [D6.02.01] |
| JELIRA | consulter un attribut d'un objet JEVEUX [D6.02.01] |
| JELSTC | retrouver les noms des objets dont le nom contient une chaîne de caractères donnée, présents sur une base JEVEUX [D6.02.01] |
| JEMARQ | incrémente la marque courante [D6.02.01] |
| JENONU | renvoie le numéro associe a un nom (hash-coding JEVEUX) [D6.02.01] |
| JENUNO | renvoie le nom associe a un numéro (hash-coding JEVEUX) [D6.02.01] |
| JEPRAT | impression des objets système ou des objets attribut de collection [D6.02.01] |
| JERAZO | remise a zéro du segment de valeurs associé a un objet JEVEUX [D6.02.01] |
| JEUNDF | mettre à "undef" un objet JEVEUX |
| JEVEUO | récupérer un pointeur sur un objet JEVEUX [D6.02.01] |
| JEVEUT | récupère un objet en mémoire de façon permanente (marque = -1) [D6.02.01] |
| JEXATR | récupération des longueurs cumulées des objets d'une collection contiguë [D6.02.01] |
| JEXNOM | fonction d'accès aux objets des collections nommées (ou des pointeurs de noms) [D6.02.01] |
| JEXNUM | fonction d'accès aux objets des collections numérotées (ou des pointeurs de noms) [D6.02.01] |
| JUVECA | agrandissement d'un objet simple JEVEUX [D6.02.01] |
| JXVERI | teste la cohérence de la segmentation mémoire de JEVEUX [D6.02.01] |
| TSTOBJ | Récupérer 5 nombres caractérisant un objet JEVEUX : contenu + certains attributs |
| WKVECT | créer un objet JEVEUX de type vecteur [D6.02.01] |

**Tableau 3.16-1**

## 3.17 LIGREL

| | |
|---|---|
| ADALIG | réorganiser les grels d'une SD ligrel pour qu'ils aient des tailles adaptées à la gestion de la mémoire. |
| CALCUL | faire les calculs élémentaires correspondant à une OPTION sur les éléments d'une SD ligrel. |
| CHLIGR | convertir un cham_elem en un autre cham_elem sur un autre ligrel. |
| EXLIM1 EXLIM2 | Création d'un ligrel à partir d'une liste de mailles |



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

| | |
|---|---|
| EXLIM3<br>EXLIMA | |
| INITEL | initialiser les types éléments présents dans le ligrel et créer les objets .PRNM et/ou .PRNS du ligrel |
| LIGLMA | Extraire d'un ligrel, la liste des mailles affectées ainsi que la liste de leurs type_elem. |
| LGPHMO | Créer un ligrel sur toutes les mailles d'un maillage |
| NBELEM | retourne le nombre d'éléments d'un GREL d'une SD ligrel |
| NBGREL | retourne le nombre de GREL d'une SD ligrel |
| NOLIGR | Ajouter des éléments finis dans une SD ligrel |
| TYPELE | Déterminer le type des éléments finis d'un GREL d'une SD ligrel |

**Tableau 3.17-1**

## 3.18 MAILLAGE

| | |
|---|---|
| CESRED | "réduire" une SD cham_elem_s sur une liste de mailles et/ou une liste de CMPS. |
| CHPNUA | transformer un cham_no en une SD nuage pour pouvoir le projeter sur un autre maillage (méthode 'NUAG_DEG_0/1') |
| CNCINV | construction de la table de connectivité inverse d'une SD maillage |
| CNSPRJ | projeter un cham_no_s sur une autre maillage |
| CNSRED | "réduire" une SD cham_no_s sur une liste de nœuds et/ou une liste de CMPS. |
| COOR_BARY | Calcule les coordonnées barycentrique d'un point dans un simplexe défini par 2, 3 ou 4 nœuds (1D/2D/3D) |
| CPCLMA | Réalise la recopie des collections .GROUPEMA et .GROUPENO d'une sd_maillage |
| CRLINU | transforme une liste de noms de nœuds en une liste de numéros de mailles tardives pour NOCART |
| EXLIM1 | Création d'un LIGREL à partir d'une liste de mailles |
| EXLIMA | Création d'un LIGREL à partir d'une liste de mailles |
| EXMANO | extraction des numéros des mailles de type SEG2 dont l'une des extrémités est un nœud de numéro donné |
| GETVEM | vérifier la cohérence d'une liste d'entités du maillage donnée par l'utilisateur |
| GETCARA_LISNO | Détermine la dimension géométrique d'une liste de nœuds : 0/1/2/3 |
| GMGNRE | remplir la liste de nœud sous-jacente à la liste de maille |
| MEGEOM | récupération du champ de géométrie dans 1 modèle ou 1 liste de charges |
| NUACHP | transformer une SD nuage en un cham_no |
| PACOAP | trier 2 listes de nœuds de manière à mettre en vis a vis les nœuds des 2 listes |
| PACOOR | donner la liste des coordonnées des nœuds d'une maille |
| DIS2NO | Calculer la distance entre 2 nœuds du maillage |
| PADIST | Calculer la distance entre 2 points |
| PANBNO | calculer le nombre de nœuds sommets, de nœuds d'arêtes, de nœuds intérieurs d'une maille d'un type donné |



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

| | |
|---|---|
| PJ2DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJXXPR (cas 2D) |
| PJ3DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJXXPR (cas 3D) |
| PJ4DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJXXPR (cas 2,5D) |
| PJ6DCO | créer une SD corresp_2_mailla pour pouvoir utiliser PJXXPR (cas 1,5D) |
| PJXXPR | projeter les champs d'une sd_resultat sur un autre maillage |
| PJXXCH | projeter 1 champ sur un autre maillage |
| PRONUA | projeter une SD nuage sur un autre maillage (méthode 'NUAG_DEG_0/1') |
| RELIEM | récupérer la liste des nœuds (ou des mailles) donnés derrière des mots clés. |
| UMALMA | établir la liste (sans doublons) des mailles d'une liste de GROUP_MA |
| UMALNO | établir la liste (sans doublons) des nœuds d'une liste de mailles |
| UTMAMO | établir la liste dans un objet jeveux (sans doublons) des mailles affectées par le modèle |
| UTMAM2 | établir la liste dans un vecteur (sans doublons) des mailles affectées par le modèle |
| UTFLMD | filtrer une liste de maille dans un objet jeveux d'après leur dimension ou leur type |
| UTFLM2 | filtrer une liste de maille dans un vecteur d'après leur dimension ou leur type |
| UTNONO | renvoie le 1er nœud (ou la 1ère maille) d'un GROUP_NO (ou d'un GROUP_MA) |
| VERIMA | vérifier la cohérence d'une liste d'entités du maillage donnée par l'utilisateur |
| VTGPLD | ajoute un champ de déplacement à un champ de géométrie : $X2 = X1 + U$ |

**Tableau 3.18-1**

## 3.19 MATERIAU

| | |
|---|---|
| MATELA | récupération des valeurs de E, NU, ALPHA dans un matériau |
| RCADMA | récupération des composantes métallurgiques d'un matériau |
| RCCOMA | obtention du comportement complet d'un matériau |
| RCCOME | obtention du comportement complet d'un matériau |
| RCFODE | obtention de la valeur de la fonction et de sa dérivée pour une fonction de la température linéaire par morceau |
| RCFONC | interpolation sur une fonction de type R(P) |
| RCMFMC | création de la carte du matériau codé à partir du cham_mater |
| RCPARE | vérification de la présence d'une caractéristique dans un comportement donné |
| RCTRAC | détermination du module de Young et de la fonction d'écrouissage à partir de la courbe de traction d'un matériau donné |
| RCVADA | obtention de la valeur des coefficients du matériau et de leurs dérivées par rapport a la température |
| RCVALA | obtention de la valeur d'un paramètre réel d'un élément d'une relation de comportement d'un matériau donné, à partir d'une adresse du matériau codé en donnant explicitement la liste des variables de commande dont peuvent dépendre les fonctions du matériau. |
| RCVALB | obtention de la valeur d'un paramètre réel d'un élément d'une relation de comportement d'un matériau donné, à partir de la désignation du point de Gauss (et sous-point). |
| RCVALC | obtention d'un paramètre complexe d'un élément d'une relation de comportement d'un matériau |



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

| | |
|---|---|
| | donné |
| RCVALE | obtention de la valeur d'un paramètre réel d'un élément d'une relation de comportement d'un matériau donné, à partir d'un nom du matériau codé |
| RCVALT | Récupérer TOUS les paramètres matériau sous un même mot clé facteur |
| RCADLV | Récupérer les paramètres matériau correspondant à un mot clé de type « liste ». Par exemple pour les comportements DHRC et UMAT. |

**Tableau 3.19-1**



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

## 3.20 MATR_ASSE

| | |
|---|---|
| ATASMO | construction d'une SD matr_asse par calcul du produit : At * A où A est une matrice rectangulaire |
| AJLAGR | ajoute les lagrange dans la matrice de masse à partir de la matrice de raideur |
| ASMATR | assembler des matrices élémentaires dans une matrice assemblée |
| ASSCHC | modifier une matr_asse pour tenir compte de l'élimination des ddls contraints par des SD char_cine |
| CONLAG | récupérer le coefficient de conditionnement des lagrange d'une matrice assemblée |
| COPMAT | copie d'une matr_asse dans une matrice pleine |
| CRESOL | Créer une SD solveur |
| CRSOLV | Créer une SD solveur par "défaut" pour la méthode LDLT |
| DETLSP | Détruire l'instance MUMPS du préconditionneur LDLT _SP |
| ECHMAT | calculer les extrema (et leur moyenne arithmétique) des valeurs absolues des termes non nuls de la diagonale de la matrice (en dehors des termes correspondant aux Lagrange) |
| EXTDIA | extraction de la diagonale d'une matrice |
| FLEXIB | calculer la matrice de flexibilité résiduelle associée a un problème cyclique avec interface Mac Neal ou aucun |
| JACOBI | résolution du problème réduit aux valeurs propres par la décomposition de Jacobi généralisée |
| MATIDE | modification des termes d'une matrice, suivant une liste de DDL spécifiés, pour la rendre inversible |
| MCCONL | tenir compte du conditionnement des Lagrange sur le second membre |
| MCMULT | effectue le produit d'une matrice par N vecteurs (si complexe) |
| MRCONL | tenir compte du conditionnement des termes de Lagrange sur le second membre |
| MRMULT | effectue le produit d'une matrice par N vecteurs (cas réel) |
| MTCMBL | combinaison linéaire de matrices |
| MTCONL | combinaison linéaire du conditionnement des Lagranges des matrices |
| MTCOPY | recopie les valeurs de la matrice dans une autre matrice |
| MTDEFS | définition de la structure d'une matrice |
| MTDSC2 | récupération de l'adresse d'un objet d'une SD matr_asse |
| MTDSCR | allocation / désallocation des descripteurs d'une SD matr_asse |
| MTEXIS | vérifier l'existence d'une matrice |
| PCLDLT | pré conditionnement d'une matr_asse en vue de l'utilisation du solveur GCPC (LDLT_INC) |
| PCMUMP | pré conditionnement d'une matr_asse en vue de l'utilisation des solveurs GCPC ou PETSC (LDLT_SP) |
| PRERES | factoriser une matr_asse (LDLT/MULT_FRONT)ou fabriquer une matrice de pré conditionnement (GCPC ou PETSC) |
| RESGRA | résolution par une méthode de gradient conjugue (GCPC) pour une matrice stockée 'MORSE' |
| RESOUD | Résolution d'un système linéaire |



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

**Tableau 3.20-1**

## 3.21 MESSAGE

| | |
|---|---|
| infbav | mettre le mécanisme INFO en mode bavard [D6.04.01] |
| infmaj | mise à jour pour le mot clé INFO [D6.04.01] |
| infmue | mettre le mécanisme INFO en mode muet [D6.04.01] |
| infniv | Renvoi le niveau d'impression et l'unité logique d'impression [D6.04.01] |
| utmess | imprimer un message d'erreur, d'alarme ou d'information avec, optionnellement, des paramètres de type chaîne de caractères, entiers ou réels (vecteur ou scalaire). On peut également fournir un numéro d'exception pour émettre une exception particulière. |
| onerrf | Pour gérer le comportement en cas d'erreur <F> : abort ou exception |

**Tableau 3.21-1**

## 3.22 MEMOIRE

| | |
|---|---|
| utgtme | Renvoie les valeurs des différents compteurs associés à la consommation de la mémoire |
| utptme | Permet de positionner certaines valeurs associées à la consommation de la mémoire (mémoire totale allouée à l'exécution, mémoire consommée par les solveurs externes, etc.) |

**Tableau 3.22-1**

## 3.23 MPI

| | |
|---|---|
| asmpi_info | Retourne le rang du processus et/ou le nombre de processus associé à un communicateur. |
| asmpi_comm dans aster_mpi.c | Fonction d'interrogation sur les communicateurs MPI : on peut récupérer le communicateur global ou courant, affecter le communicateur courant, en supprimer un. |
| asmpi_split_comm dans aster_mpi.c | Permet de créer des sous-communicateurs. |
| asmpi_barrier | Place une barrière sur un communicateur ou le communicateur courant. |
| asmpi_comm_vect | Effectue une communication de type MPI_ALLREDUCE, MPI_BCAST, MPI_REDUCE pour un vecteur (ou un scalaire) de type entier, réel ou complexe. Elle s'effectue sur le communicateur courant. |
| asmpi_comm_jev | Effectue une communication de type MPI_ALLREDUCE, MPI_BCAST, MPI_REDUCE pour un objet JEVEUX. Elle s'effectue sur le communicateur courant. |
| asmpi_comm_point | Effectue une communication point à point MPI_SEND, MPI_RECV d'un vecteur (ou scalaire) de type entier ou réel. |
| asmpi_status, asmpi_check, asmpi_warn | Fonctions permettant de vérifier que les communications MPI se passent bien : avant d'initier une communication globale (asmpi_barrier, asmpi_comm_vect, asmpi_comm_jev), on vérifie en effectuant une communication non bloquante entre |



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

| | |
|---|---|
| | le processus 0 et chacun des autres que tous les processeurs sont au rendez-vous (dans le delai de 20 % du temps restant). Sinon on émet un message d'erreur et on interrompt le calcul. |
| comatr.f | Faire les communications MPI nécessaires pour que tous les processeurs (associés au communicateur courant) se transfèrent mutuellement certaines colonnes d'une matrice de I/R/C. On peut aussi faire la même chose sur les lignes de la matrice transposée. |
| sdmpic.f | Faire les communications MPI nécessaires pour “compléter” une SD (i.e la rendre 'MPI_COMPLET'). |

**Tableau 3.23-1**

## 3.24 MPLEIN

| | |
|---|---|
| AMPPR | ajouter une matrice pleine réelle à une matrice pleine réelle |
| COPMAT | copie d’une MATR_ASSE dans une matrice pleine |
| MAVEC | passage matrice pleine (m*m) > demi-matrice colonne vecteur(n) |
| MGAUSS | résolution par la méthode de Gauss d’un système linéaire |
| PMAT | produit de matrices carrées |
| PMAVEC | produit matrice carrée pleine par un vecteur |
| PMPPR | produit de deux matrices stockées pleines avec prise en compte de transposition par l’intermédiaire d’indicateur |
| PRMAMA | produit de matrices pleines rectangulaires |
| PROMAT | produit de deux matrices pleines |
| UTBTAB | fait le produit de matrices pleines : BT * A * B |
| VECMA | transforme une matrice symétrique (triangulaire) en une matrice carrée |

**Tableau 3.24-1**

## 3.25 NUME_DDL

| | |
|---|---|
| CHEDDL | chercher le rang d’un ddl à partir de son type et du nœud |
| CRPRNO | création et allocation d’une structure prof_chno |
| NUDLG2 | Créer un objet permettant d'apparier les couples de coefficients de Lagrange correspondants aux relations linéaires dualisées. |
| NUMERO | Créer une SD nume_ddl |
| POSDDL | donne le numéro du ddl associé au nœud et à sa composante |
| PTEDDL | récupérer les numéros d’équation correspondant à certains noms de CMPS |
| PTEEQU | créer l’objet .DEEQ d’une SD prof_chno |
| RGNDAS | retrouver le nom du nœud et la composante correspondant à un numéro d’équation dans un système assemblé |



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

**Tableau 3.25-1**

## 3.26 PREPOST

| | |
|---|---|
| ECRTES | écriture de l'en tête d'un dataset SUPERTAB |
| GICOOR | créer la collection qui donne la permutation des nœuds des mailles (ASTER-> GIBI) |
| INISTB | initialisation des noms des mailles ASTER-TRIFOU en fonction du code graphique I-DEAS 4.0 |
| IRADHS | adhérences IDEAS |
| IRGAGS | recherche des grandeurs IDEAS présentent dans une grandeur |

**Tableau 3.26-1**

## 3.27 REPERE

| | |
|---|---|
| ANGVX | Calcule les 2 angles nautiques à partir d'un vecteur |
| ANTISY | calcule une matrice de rotation dans R3 |
| CANOR2 | calcule la normale à un SEG2 (en 2D) |
| CANOR3 | calcule la normale à un TRIA3 (en 3D) |
| CANORM | calculer la normale à une maille en un nœud avec ou sans normalisation de ce vecteur |
| CHGREP | Changement de repère: local global et vice-versa |
| CHMALG | passage du repère local au repère global des matrices élémentaires |
| CQ3D2D | calcul des coordonnées 2D d'un triangle ou d'un quadrangle à partir de ses coordonnées 3D passage dans le repère du plan du triangle ou du quadrangle avec téta=angle entre l'axe X et le cote A1A2 |
| CTETGD | calcul de la matrice téta permettant de passer des ddl de l'interface droite à ceux de l'interface gauche |
| GLO LOC | changement de repère pour un système dynamique modal |
| INTET0 | calculer la matrice de rotation pour DX,DY,DZ,DRX,DRY et DRZ |
| LOCGLO | passage du repère local au repère global pour un système dynamique modal |
| MAROTA | calcule la matrice de rotation correspondant au vecteur rotation |
| MATPGL | Construction de la matrice de passage global local |
| MATRO2 | calcul de la matrice rotation pour une POUTRE courbe |
| MATROT | calcul de la matrice rotation pour une POUTRE droite |
| MUDIRX | calcule les cosinus directeurs de la matrice de passage du repère de l'élément au repère de référence ainsi que les 3 directions normées du repère de l'élément |
| ORIEN2 | orientation d'un trièdre défini par 3 points |
| ORTREP | récupération des données Utilisateur définissant le repère d'orthotropie relatif à l'élément courant |
| PROJMG | passage BASE_MODALE → repère physique |
| REFLTH | calcule le passage des termes de conductivité du repère de référence au repère de l'élément |
| UTPSGL | passage Global → Local pour une matrice élémentaire symétrique (triangulaire) |



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

| | |
|---|---|
| UTPSLG | passage Local → Global pour une matrice élémentaire symétrique (triangulaire) |
| UTPVGL | passage Global → Local pour un vecteur |
| UTPVLG | passage Local → Global pour un vecteur |

**Tableau 3.27-1**

## 3.28 Resuelem

| | |
|---|---|
| ASASMA | assembler les matrices élémentaires de rigidité et de Dirichlet |
| ASASVE | assembler les vecteurs élémentaires provenant des charges |
| ASMATR | assembler des matrices élémentaires dans une matrice assemblée |
| ASSVEC | Assembler des vecteurs élémentaires pour en faire un second membre (SD cham_no) |
| CALCUL | faire les calculs élémentaires correspondant à une OPTION sur les éléments d'une SD ligrel. |
| CESVAR | créer une SD cham_elem_s (DCEL_I) permettant d'étendre les cham_elem (VARI_R) calculés par la routine CALCUL. |
| MEAMME | calcul des matrices élémentaires d' AMOR_MECA ou RIGI_MECA_HYST |
| MEDIME | calcul des matrices élémentaires des éléments de Lagrange (mécanique) |
| MEDITH | calcul des matrices élémentaires des éléments de Lagrange (thermique) |
| MEMAME | calcul des matrices élémentaires de MASS_MECA |
| MEMARE | créer et initier l'objet .REFE_RESU des SD matr_elem (ou SD vect_elem) |
| MERIME | calcul des matrices élémentaires de RIGI_MECA (élastique) |
| MERIMO | calcul des matrices élémentaires des éléments du modèle et des termes élémentaires du résidu (STAT_NON_LINE) |
| MERITH | calcul des matrices élémentaires de RIGI_THER |
| TYPMAT | déterminer si un matr_elem contient des matrices élémentaires non-symétriques |
| REDETR | détruire les resuelem nuls présents dans un matr_elem (pb du doublon matrices symétriques, non symétriques) |

**Tableau 3.28-1**

## 3.29 RESULTAT

| | |
|---|---|
| BMNODI | récupérer les déformées d'interface dans une SD base_modale |
| CTETGD | calcul de la matrice téta permettant de passer des ddl de l'interface droite à ceux de l'interface gauche |
| DCAPNO | récupérer l'adresse d'un .VALE d'un cham_no a partir de son type et de numéro d'ordre dans un résultat composé |
| DYARCH | saisie du mot clé facteur ARCHIVAGE (dans une SD resultat) |
| EXTMOD | extraire d'un concept mode_meca la déformée pour un ou plusieurs ddl. Les lagranges sont supprimes. |
| FOCRCH | récupération d'une fonction dans une structure tran_gene pour un nœud de choc |



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

| | |
|---|---|
| IMBAMO | imprimer les résultats relatifs a la base modale |
| IRECRI | écriture d'une structure de données résultat sur un fichier |
| IRPARA | impression des paramètres d'une structure de données résultat |
| IRPARB | détermination / vérification des paramètres d'une structure de données résultat |
| IRTITR | impression du titre d'une SD resultat |
| NDARCH | archivage des déplacements, vitesses, accélérations, contraintes |
| PROJMG | passage base_modale → repère physique |
| RSADPA | récupération des adresses JEVEUX des paramètres de calcul ou des variables d'accès d'une structure de données résultat pour le numéro d'ordre donné et pour la liste de variables de noms symboliques [D6.05.01] |
| RSAGSD | redimensionnement d'une structure de données résultat [D6.05.01] |
| RSBARY | Interpoler un champ entre 2 instants d'un SD resultat [D6.05.01] |
| RSCRSD | Création d'une structure de données résultat [D6.05.01] |
| RSEXCH | récupération du nom du champ d'une structure de données résultat [D6.05.01] |
| RSEXIS | Existence d'une structure de données résultat [D6.05.01] |
| RSEXPA | Existence d'un paramètre (ou d'une variable d'accès) dans une structure de données résultat [D6.05.01] |
| RSINCH | Interpolation d'un champ d'une structure de données résultat [D6.05.01] |
| RSINDI | trouver un réel (ou un complexe) dans une liste de paramètres d'une SD resultat [D6.05.01] |
| RSINFO | impression (sur listing) de la structure d'une SD resultat [D6.05.01] |
| RSMENA | "menage" (suppression des objets inutiles) dans une SD resultat |
| RSNOCH | Noter un champ dans la structure de données résultat [D6.05.01] |
| RSNOPA | récupération du nombre de variables d'accès et du nombre de paramètres ainsi que de leur noms d'une structure de données résultat [D6.05.01] |
| RSORAC | Récupération des numéros d'ordre d'une structure de données résultat à partir d'une variable d'accès [D6.05.01] |
| RSRUSD | Détruire les champs d'une structure de données résultat à partir d'un numéro d'ordre [D6.05.01] |
| RSUTNU | Récupération des numéros d'ordre d'une structure de données résultat à partir d'une variable d'accès [D6.05.01] |
| RSUTN2 | Comme RSUTNU, mais filtre les numéros d'ordre trouvés en vérifiant l'existence d'un champ symbolique sur ces numéros d'ordre. |
| RSUTN1 | Comme RSUTNU, mais filtre les numéros d'ordre trouvés en vérifiant l'existence d'un paramètre (ou d'une variable d'accès) sur ces numéros d'ordre. |

**Tableau 3.29-1**

## 3.30 RUPTURE

| | |
|---|---|
| DFFDIR | Retourne le vecteur de direction de propagation (1er vecteur de la base locale en fond de fissure) en un nœud |
| DFFNOR | Retourne le vecteur normal à la surface de la fissure (2ème vecteur de la base locale en fond de fissure) en un nœud |
| DFFTAN | Retourne le vecteur tangent au fond de fissure (3ème vecteur de la base locale en fond de fissure) |



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

| | |
|---|---|
| | en un nœud |
| DFFLON | Calcule une estimation de la longueur des segments du fond de fissure en un nœud du fond (uniquement en 3d) |
| GABSCU | pour chaque nœud du fond de fissure on calcule son abscisse curviligne |
| GDFONC | calcul des gradients pour le calcul du taux de restitution d'énergie en 2D |
| GDINOR | calcul de la direction du champ thêta dans le cas ou la normale au plan des lèvres figure dans la sd fond_fiss |
| GDIREC | pour chaque nœud du fond de fissure, on calcule la direction du champ thêta |
| EXIXFE | Détecte si on a affaire à une modélisation XFEM |
| XVFIMO | Détecte si une fissure X-FEM (sd_fiss_xfem) est associé à un modele |

**Tableau 3.30-1**

## 3.31 SD

| | |
|---|---|
| COPISD | dupliquer une structure de données sous un autre nom [D6.07.05] |
| DETRSD | détruire une structure de données [D6.07.05] |
| DISMOI | poser une question sur une SD [D6.07.05] |
| EXISD | Tester l'existence d'une SD [D6.07.05] |
| IMPRSD | Imprimer (lisible) une structure de données (champ, table ou matrice) [D6.07.05] |
| UTIMSD | Imprimer (dump) le contenu des objets d'une SD [D6.07.05] |
| GNOMSD | Obtenir un nom valide pour SD "cachée". |

**Tableau 3.31-1**

## 3.32 SUPERVISEUR

| | |
|---|---|
| GCNCON | obtenir le nom d'une SD (K8) qui ne soit pas en conflit avec les autres noms de SD |
| GETFAC | retourne le nombre d'occurrences d'un mot clé facteur [D6.03.01] |
| GETLTX | retourne la longueur des chaînes d'un mot clé de type 'texte' [D6.03.01] |
| GETRES | retourne le nom et le type du résultat d'une commande [D6.03.01] |
| GETTCO | retourne le type d'une SD utilisateur [D6.03.01] |
| GETVC8 | retourne la liste des arguments d'un mot clé de type 'complexe' [D6.03.01] |
| GETVID | retourne la liste des arguments d'un mot clé de type 'identificateur' [D6.03.01] |
| GETVIS | retourne la liste des arguments d'un mot clé de type 'entier' [D6.03.01] |
| GETVR8 | retourne la liste des arguments d'un mot clé de type 'réel' [D6.03.01] |
| GETVTX | retourne la liste des arguments d'un mot clé de type 'texte' [D6.03.01] |
| UTALRM | permet de masquer temporairement une alarme (puis en rétablir l'affichage) |

**Tableau 3.32-1**



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

## 3.33 TABLE

| | |
|---|---|
| TBAJLI | Ajouter une ligne à une SD table [D6.06.01] |
| TBAJPA | Ajouter des paramètres dans une SD table [D6.06.01] |
| TBAJVA | Ajouter une valeur « à la bonne place » associée à un paramètre dans une des listes typées |
| TBAJVC | Ajouter une valeur C « à la bonne place » associée à un paramètre dans une des listes typées |
| TBAJVI | Ajouter une valeur I « à la bonne place » associée à un paramètre dans une des listes typées |
| TBAJVK | Ajouter une valeur K « à la bonne place » associée à un paramètre dans une des listes typées |
| TBAJVR | Ajouter une valeur R « à la bonne place » associée à un paramètre dans une des listes typées |
| TBCRSD | créer une SD table [D6.06.01] |
| TBCRSD | déclarer une nouvelle SD table [D6.06.01] |
| TBEXFO | extraire une fonction d'une SD table en désignant 2 colonnes en vis-à-vis. [D6.06.01] |
| TBEXIP | Existence d'un paramètre dans une SD table [D6.06.01] |
| TBEXTB | Extraire une sous-table d'une SD table [D6.06.01] |
| TBEXVE | extraire un objet_JEVEUX contenant une colonne d'une SD table. [D6.06.01] |
| TBLIVA | Lecture d'une cellule d'une SD table [D6.06.01] |
| TBNULI | Renvoie le numéro d'une ligne d'une SD table [D6.06.01] |

**Tableau 3.33-1**

## 3.34 TITRE

| | |
|---|---|
| IRTITR | impression du titre d'une SD resultat |
| TITRE2 | créer un sous-titre |
| TITRE | créer un titre |

**Tableau 3.34-1**

## 3.35 TYPE_FORTRAN

| | |
|---|---|
| ALMULR | produit de n nombres réels avec test de l'overflow et de l'underflow avec cumul de valeur antérieur ou remise a zéro |
| AS_ALLOCATE AS_DEALLOCATE | Pour allouer (ou désallouer) un vecteur de I, R, C, K8, ... |
| BASE3N | calcule une base orthonormée de R3 ayant son 1er vecteur colinéaire à un vecteur donné |
| CODENT | écrit un entier dans une chaîne de caractères |
| CODREE | écrit un réel dans une chaîne de caractères |
| COMPR8 | compare deux réels entre eux à une précision donnée (en absolu ou en relatif) |
| EXTRAC | extraction dans un tableau contenant des vecteurs à des instants successifs du vecteur éventuellement interpolé à l'instant souhaité |
| FOVERF | vérification du caractère croissant des valeurs dans un vecteur |



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

| | |
|---|---|
| FREQOM | calcule la fréquence associée a la pulsation |
| GCNCO2 | obtenir une chaîne de caractère par incrémentation d'un numéro |
| GGUBS | générateur de nombres (pseudo-)aléatoires uniformément repartis entre (0,1) |
| INDIIS | retourne le rang d'un entier dans un vecteur d'entier |
| INDIK8 | retourne le rang d'un K8 dans un vecteur de K8 |
| INDK16 | retourne le rang d'un K16 dans un vecteur de K16 |
| INDK24 | retourne le rang d'un K24 dans un vecteur de K24 |
| INDK32 | retourne le rang d'un K32 dans un vecteur de K32 |
| INDK80 | retourne le rang d'un K80 dans un vecteur de K80 |
| KNDIFF | faire la différence entre 2 listes de chaînes de caractères LK3 = LK1 - LK2 |
| KNDOUB | vérifier qu'il n'y a pas de doublons dans une liste de chaînes de caractères |
| KNINCL | vérifier qu'une liste de chaînes de caractères est incluse dans une autre |
| KNINDI | retourne le rang d'un K* dans un vecteur de K* |
| LIIMPR | imprimer une liste d'entiers ou de réels |
| LSAME | teste l'égalité de 2 chaînes de caractères indépendamment de leurs casses |
| LXCAPS | met en majuscules une chaîne de caractères |
| LXLGUT | retourne la longueur utilise d'une chaîne de caractères (sans les blancs) |
| LXLIIS | décode une chaîne de caractères pour y lire un entier |
| LXSCAN | décode une chaîne de caractères en mots de différents types : entier, réel, texte, ... |
| NORMEV | norme un vecteur de R3 et retourne sa norme initiale |
| OMEGA2 | calcule la pulsation associée à la fréquence |
| ORDIS | réarrangement d'une liste d'entiers par ordre croissant |
| ORDR8 | trouver l'ordre croissant d'une liste de réels, pas de modification de l'ordre d'entrée mais détermination d'un pointeur d'ordre |
| PERMR8 | permutation circulaire des éléments d'un tableau de REAL*8 |
| PROVEC | calcul du produit vectoriel de deux vecteurs de R3 |
| PSCVEC | multiplie un vecteur de Rn par un scalaire réel |
| R8INIR | initialisation d'un vecteur de Rn |
| SOMINT | fonction (de type entier) sommant tous les termes d'un vecteur d'entiers. |
| TRI | tri (Quick Sort) d'un tableau d'entiers et répercussion sur un tableau d'entiers |
| TRIR | tri (Quick Sort) d'un tableau d'entiers et répercussion sur un tableau de réels |
| UTLISI | utilitaire d'opérations logiques sur les listes d'entiers : union, intersection, singleton |
| UTREMT | recherche un mot dans une liste de mots |
| UTTR24 | trier une liste de K24 |
| UTTRII | trier une liste d'entiers |
| UTTRIR | trier une liste de réels |
| VDIFF | calcule la différence entre 2 vecteurs : Z = X - Y |
| VECMA | transforme une matrice symétrique (triangulaire) en une matrice carrée |
| VECINC | initialisation d'un vecteur complexe à une valeur complexe donnée. |



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

| | |
|---|---|
| VECINI | initialisation d'un vecteur réel à une valeur réelle donnée. |
| VECINK | initialisation d'un vecteur de caractères à une chaîne de caractères donnée. |
| VECINT | initialisation d'un vecteur d'entiers à une valeur entière donnée. |

**Tableau 3.35-1**

## 3.36 VARI_COM

| | |
|---|---|
| VRCINS | fabrication du champ de variables de commande à un instant donné. |
| RCVARC | récupération d'une variable de commande sur un point de Gauss (dans une routine de calcul élémentaire) |
| NMVCD2 | teste si une variable de commande est présente |

**Tableau 3.36-1**



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