Operators AFFE_CHAR_MECA and AFFE_CHAR_MECA_F

1 Drank

To affect loadings and boundary conditions on a mechanical model.

- For AFFE_CHAR_MECA, the affected values do not depend on any parameter and are defined by actual values.
- For AFFE_CHAR_MECA_F, the affected values are function of one or more parameters as a whole {INST, X, Y, Z}.

  These functions must be in particular defined beforehand by the call to one of the operators:

  - DEFI_CONSTANTE [U4.31.01],
  - DEFI_FONCTION [U4.31.02],
  - DEFI_NAPPE [U4.31.03],
  - CALC_FONC_INTERP [U4.32.01].

Contents

2 general Syntax

```
CH  [char_meca] = AFFE_CHAR_MECA

          (  ♦  MODELS =mo    ,                            [model]

          ♦  | VERI_NORM=/       "YES",                    [DEFAULT]
                            /    "NON",
             | LIAISON_XFEM=/    "NON",                    [DEFAULT]
                            /    "YES",
             | EVOL_CHAR=evch                              [evol_char]
             | ROTATION=          (Omega, rear, Br, Cr)       [l_R]
              | PRE_SIGM =        sigm/                    [carte_sdaster]
                                                        /  [cham_elem]
             | PESANTEUR=_F       (see key word  PESANTEUR        [§ 4.6])
             | DDL_IMPO=_F        (see key word     DDL_IMPO        [§ 4.9])
             | FACE_IMPO= _F      (see key word     FACE_IMPO       [§
4.10])
             | ARETE_IMPO= _F     (see key word     ARETE_IMPO      [§
4.45])
             | LIAISON_DDL=_F     (see key word     LIAISON_DDL     [§
4.11])
             | LIAISON_OBLIQUE=_F (see key word  LIAISON_OBLIQUE  [§ 4.12])
             | LIAISON_GROUP=_F   (see key word     LIAISON_GROUP   [§
4.13])
             | LIAISON_MAIL=_F    (see key word     LIAISON_MAIL    [§
4.14])
             | LIAISON_CYCL=_F    (see key word     LIAISON_CYCL    [§
4.15])
             | FORCE_NODALE=_F    (see key word     FORCE_NODALE    [§
4.16])
             | LIAISON_SOLIDE=_F  (see key word     LIAISON_SOLIDE  [§
4.17])
             | LIAISON_ELEM=_F    (see key word     LIAISON_ELEM    [§
4.18])
             | LIAISON_UNIF=_F    (see key word     LIAISON_UNIF    [§
4.19])
             | LIAISON_CHAMNO=_F  (see key word     LIAISON_CHAMNO  [§
4.20])
             | CHAMNO_IMPO=_F     (see key word     CHAMNO_IMPO     [§
4.21])
             | LIAISON_INTERF=_F  (see key word     LIAISON_INTERF  [§
4.22])
             | VECT_ASSE=_F       (see key word  VECT_ASSE        [§ 4.23])
             | FORCE_SOL=_F       (see key word  FORCE_SOL        [§ 4.24])
continuum  | FORCE_FACE=_F      (see       mot-cléFORCE_FACE      [§ 4.25])
             | FORCE_ARETE=_F     (see       mot-cléFORCE_ARETE     [§
4.26])
             | FORCE_CONTOUR=_F   (see       mot-cléFORCE_CONTOUR      [§
4.27])
             | FORCE_INTERNE=_F   (see       mot-cléFORCE_INTERN       [§
4.28])
             | PRES_REP=_F        (see       mot-cléPRES_REP        [§
4.29])
             | EFFE_FOND=_F       (see       mot-cléEFFE_FOND          [§
4.30])
              | PRE_EPSI=_F           ( see key word      PRE_EPSI        [§
4.31])
beam shell   | FORCE_POUTRE=_F    (see       mot-cléFORCE_POUTRE       [§
4.32])
             | DDL_POUTRE =_F     (see       mot-cléDDL_POUTRE      [§
4.33])
             | FORCE_TUYAU=_F     (see       mot-cléFORCE_TUYAU     [§
4.34])
```

```
                | FORCE_COQUE=_F      (see      mot-cléFORCE_COQUE      [§ 4.35])
                | LIAISON_COQUE=_F    (see      mot-cléLIAISON_COQUE    [§ 4.36])
concrete        | RELA_CINE_BP=_F     (see      mot-cléRELA_CINE_BP     [§ 4.37])
electromechanical | FORCE_ELEC=_F     (see      mot-cléFORCE_ELEC       [§ 4.38])
                | INTE_ELEC=_F        (see      mot-cléINTE_ELEC        [§ 4.39])
acoustic        | IMPE_FACE=_F        (see      mot-cléIMPE_FACE        [§ 4.40])
                | VITE_FACE=_F        (see      mot-cléVITE_FACE        [§ 4.41])
                | ONDE_FLUI=_F        (see      mot-cléONDE_FLUI        [§ 4.42])
                | ONDE_PLANE=_F       (see      mot-cléONDE_PLANE       [§ 4.43])
thermo-hydro    | FLUX_THM_REP=_F     (see      mot-cléFLUX_THM_REP     [§ 4.44])

            ◊INFO   =          1  ,                              [DEFAULT]
                              /2  ,
      )
```

```
CH   [char_meca] = AFFE_CHAR_MECA_F

      (   ♦MODELE=mo        ,                          [model]
          ♦   | DDL_IMPO=_F         (see   mot-cléDDL_IMPO         [§ 4.9])
              | FACE_IMPO=_F        (see   mot-cléFACE_IMPO        [§ 4.10])
              | LIAISON_DDL=_F      (see   mot-cléLIAISON_DDL      [§ 4.11])
              | LIAISON_OBLIQUE=_F  (see   mot-cléLIAISON_OBLIQUE  [§ 4.12])
              | LIAISON_GROUP=_F    (see   mot-cléLIAISON_GROUP    [§ 4.13])
              | FORCE_NODALE=_F     (see   mot-cléFORCE_NODALE     [§ 4.17])
              | LIAISON_SOLIDE=_F   (see   mot-cléLIAISON_SOLIDE   [§ 4.18])
              | LIAISON_UNIF=_F     (see   mot-cléLIAISON_UNIF     [§ 4.20])
continuum     | FORCE_FACE=_F       (see   mot-cléFORCE_FACE       [§ 4.25])
              | FORCE_ARETE=_F      (see   mot-cléFORCE_ARETE      [§ 4.26])
              | FORCE_CONTOUR=_F    (see   mot-cléFORCE_CONTOUR    [§ 4.27])
              | FORCE_INTERNE=_F    (see   mot-cléFORCE_INTERN     [§ 4.28])
              | PRES_REP=_F         (see   mot-cléPRES_REP         [§ 4.29])
              | EFFE_FOND=_F        (see   mot-cléEFFE_FOND        [§ 4.30])
              | PRE_EPSI=_F         ( see key word PRE_EPSI        [§ 4.31])
beam shell    | FORCE_POUTRE=_F     (see   mot-cléFORCE_POUTRE     [§ 4.32])
              | FORCE_TUYAU=_F      (see   mot-cléFORCE_TUYAU      [§ 4.34])
              | FORCE_COQUE=_F      (see   mot-cléFORCE_COQUE      [§ 4.35])
              | LIAISON_COQUE=_F    (see   mot-cléLIAISON_COQUE    [§ 4.36])
acoustic      | IMPE_FACE=_F        (see   mot-cléIMPE_FACE        [§ 4.40])
              | VITE_FACE=_F        (see   mot-cléVITE_FACE        [§ 4.41])
              | ONDE_PLANE=_F       (see   mot-cléONDE_PLANE       [§ 4.43])
              | FLUX_THM_REP=_F     (see   mot-cléFLUX_THM_REP     [§ 4.44])
              | VERI_NORM=/    "YES",                              [DEFAULT]
                          /    "NON",
      )
```

3 possible

Messages d'erreur General information related to command AFFE_CHAR_MECA

It arrives sometimes that an ordering of mechanical computation (MECA_STATIQUE, STAT_NON_LINE,…) during stop in fatal error the computation of the second elementary members due to the loadings defined in the AFFE_CHAR_MECA_xx commands. When the code stops during these elementary computations, important information of the error message is the name of the computation option required by the code.

The name of this option is in general unknown to the user and it is thus difficult for him to understand the message.

In the table below, one gives in with respect to the names of the computation options, the name of the command and of the key word factor which make it possible to activate this option.

Elementary computation option	Orders	Key word factor
CHAR_MECA_EPSI_F	AFFE_CHAR_MECA_F	PRE_EPSI
CHAR_MECA_EPSI_R	AFFE_CHAR_MECA	PRE_EPSI
CHAR_MECA_FF1D1D	AFFE_CHAR_MECA_F	FORCE_POUTRE
CHAR_MECA_FF1D2D	AFFE_CHAR_MECA_F	FORCE_CONTOUR
CHAR_MECA_FF1D3D	AFFE_CHAR_MECA_F	FORCE_ARETE
CHAR_MECA_FF2D2D	AFFE_CHAR_MECA_F	FORCE_INTERNE
CHAR_MECA_FF2D3D	AFFE_CHAR_MECA_F	FORCE_FACE
CHAR_MECA_FF3D3D	AFFE_CHAR_MECA_F	FORCE_INTERNE
CHAR_MECA_FFCO2D	AFFE_CHAR_MECA_F	FORCE_COQUE
CHAR_MECA_FFCO3D	AFFE_CHAR_MECA_F	FORCE_COQUE
CHAR_MECA_FLUX_F	AFFE_CHAR_MECA_F	FLUX_THM_REP
CHAR_MECA_FLUX_R	AFFE_CHAR_MECA	FLUX_THM_REP
CHAR_MECA_FORC_F	AFFE_CHAR_MECA_F	FORCE_NODALE
CHAR_MECA_FORC_R	AFFE_CHAR_MECA	FORCE_NODALE
CHAR_MECA_FR1D1D	AFFE_CHAR_MECA	FORCE_POUTRE
CHAR_MECA_FR1D2D	AFFE_CHAR_MECA_F	FORCE_CONTOUR
CHAR_MECA_FR1D3D	AFFE_CHAR_MECA	FORCE_ARETE
CHAR_MECA_FR2D2D	AFFE_CHAR_MECA	FORCE_INTERNE
CHAR_MECA_FR2D3D	AFFE_CHAR_MECA	FORCE_FACE
CHAR_MECA_FR3D3D	AFFE_CHAR_MECA	FORCE_INTERNE
CHAR_MECA_FRCO2D	AFFE_CHAR_MECA	FORCE_COQUE
CHAR_MECA_FRCO3D	AFFE_CHAR_MECA	FORCE_COQUE
CHAR_MECA_FRELEC	AFFE_CHAR_MECA	FORCE_ELEC
CHAR_MECA_PESA_R	AFFE_CHAR_MECA	PESANTEUR
CHAR_MECA_PRES_F	AFFE_CHAR_MECA_F	PRES_REP
CHAR_MECA_PRES_R	AFFE_CHAR_MECA	PRES_REP
CHAR_MECA_ROTA_R	AFFE_CHAR_MECA_F	ROTATION

4 Operands

4.1 General information on the operands

4.1.1 Two categories of operands

the operands under a key word factor are of two forms:

- operands specifying the geometrical entities on which the loadings are affected (key words `GROUP_NO`, `GROUP_MA`, etc…). The arguments of these operands are identical for the two operators,
- the operands specifying the affected values (`DX`, `DY`, etc…). The meaning of these operands is the same one for the two operators. The arguments of these operands are all of the real type for operator `AFFE_CHAR_MECA` and of the standard `function` (created in particular by one of operators `DEFI_FONCTION`, `DEFI_NAPPE` or `DEFI_CONSTANTE`) for operator `AFFE_CHAR_MECA_F`.
  This is true near with an exception: the argument of `COEF_MULT` for the key word factor `LIAISON_DDL` in `AFFE_CHAR_MECA_F` is obligatorily of real type.

We will thus not distinguish in this document, except fast mention of the opposite, two operators `AFFE_CHAR_MECA` and `AFFE_CHAR_MECA_F`.

4.1.2 Designation of the topological entities of assignment of the loadings

In a general way, the entities on which values must be affected are defined:

- by node and in this case:
  - maybe by operand `GROUP_NO` allowing to introduce a list of nodes groups: let us note that in certain cases a group of node should contain one node,
  - that is to say by the operand `NOEUD` allowing to introduce one nodes list.
- by mesh and in this case:
  - either by `GROUP_MA` allowing to introduce a list of mesh groups,
  - or by `MESH` allowing to introduce a list of meshes.

4.1.3 Regulate of overload

to define the field of assignment most simply possible, one uses **the rule of overload** defined in the document “Rules of overload” [U1.03.00] :
when various occurrences of the same key word factor exist, it is the last assignment which precedes.

The keywords different factors always cumulate.

If for example, the made user:

```
FORCE_FACE (GROUP_MA=' G1', FX=12.)
PRES_REP  (GROUP_MA=' G1', PRES=13.)
```

and if the norm for $G1$ is directed according to X,

then all will occur as if one had made:

```
FORCE_FACE (GROUP_MA=' G1', FX=25.)
```

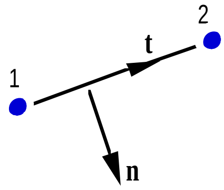
4.1.4 Structural elements, continuums

For the assignment of the distributed loadings on the elements with average average (plate - shell) or with average fiber (beam, cable, bar) the key words factors are distinct from those used for the continuums.

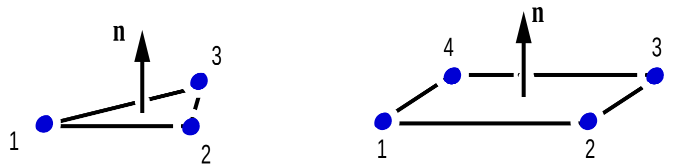
4.1.5 Norms and tangents with meshes

Normal:

- SEG2 or SEG3 in 2D (coordinated defined by COOR_2D in mesh file in the Aster format). The norm n is such as (n, t) form a direct reference, t being carried by the segment directed by the first two nodes of the segment.

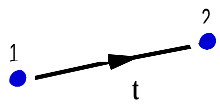


- QUAD4,..., QUAD9, TRIA3, TRIA6 in 3D (coordinated defined by COOR_3D in mesh file in the Aster format). The directional sense of the norm n is that corresponding to the direct meaning of description of the mesh.

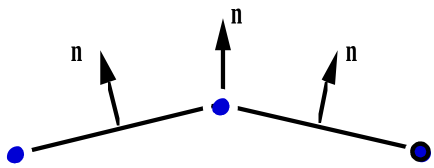


Tangents:

Can be specified only if the mesh is of type SEG2 or SEG3 in 2D. The tangent is that defined by the segment directed by its first two nodes.



If DNOR (or DTAN) are specified, the norm (or the tangent) on a node is the average of the norms or of the tangents of meshes which have this joint node (except for the curved quadratic elements where the norm is correctly calculated in any point)



4.2 Operand MODELS

```
♦MODELE=mo          ,
```

Product concept by the operator AFFE_MODELE where definite affected ones on the mesh the are element types finished.

4.3 Operand VERI_NORM

```
|   VERI_NORM=/       "YES"        [DEFAULT]
                /   "NON"
```

Checking of the directional sense of the norms to meshes surface in 3D (meshes of skin SORTED or QUAD) and linear in 2D (meshes of skin SEG). This relates to key word PRES_REP and FACE_IMPO "DNOR".
If a norm is not outgoing, there is emission of an error message fatal.

To reorientate meshes in order to have outgoing norms, operator MODI_MAILLAGE [U4.23.04]

should be used key word ORIE_PEAU_2D and ORIE_PEAU_3D.

No checking is made on the shells. To check their directional sense, one also returns to operator MODI_MAILLAGE key word ORIE_NORM_COQUE.

4.4 Operand LIAISON_XFEM (AFFE_CHAR_MECA only)

```
|   LIAISON_XFEM=/       "NON"       [DEFAULT]
                 /   "YES"
```

During a computation with the method X-FEM [R7.02.12], activation of the contact requires to add connections between the degrees of freedom of contact to observe condition LBB [R5.03.54]. These connections are automatically calculated and introduced into the load when LIAISON_XFEM=' OUI' is indicated. It is thus necessary to create an additional expenditure, as on the following example, and to use it for any computation X-FEM with contact.

```
chxfem=AFFE_CHAR_MECA           ( MODELE=modele           ,
                                     LIAISON_XFEM=' OUI',
                             )
```

4.5 Operand EVOL_CHAR (AFFE_CHAR_MECA only)

```
|   EVOL_CHAR =evch  ,
```

Loadings evolutionary in the time of the type "evol_char" produced by LIRE_RESU [U7.02.01] and containing fields of pressure, densities of volume force in 2D or 3D and densities of surface force in 2D or 3D.

4.6 Operand PESANTEUR (AFFE_CHAR_MECA only)

```
|   PESANTEUR =_F  (        ♦/GRAVITE=G              ,              [R]
                            ♦/DIRECTION=            (ap, LP, CP),    [l_R]
                            ◊/MAILLE=lma                  ,
[l_maille]
                               /GROUP_MA      =lgma ,
[l_gr_maille]
                 )
```

G represents the intensity of the field of gravity and vector DIRECTION specifies the direction and the meaning of application of the field. The loading which results from it is form:

$$\rho g \frac{(a_p \boldsymbol{i} + b_p \boldsymbol{j} + c_p \boldsymbol{k})}{\sqrt{a_p^2 + b_p^2 + c_p^2}}$$

where $(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})$ is the total cartesian coordinate system.

ρ is the definite density like characteristic of the material (see operators DEFI_MATERIAU [U4.43.01] and AFFE_MATERIAU [U4.43.03]).

By default, this field applies to all the model. It is possible to restrict it with part of the model using the key word NETS and GROUP_MA, which specify meshes to which the field applies.

Note:

It can exist U difference between the theoretical solution of computation of the weight of structure and the solution finite elements. That is due to the discretization of the problem.

In axisymmetric modelization, gravity is exerted only parallel to the axis of revolution Y.

When loading PESANTEUR is used with MECA_STATIQUE, Code_Aster calculates the forces with the nodes by means of the stiffness matrix of the element and the displacements previously calculated (option EFGE_ELNO). One thus finds well the weight of structure where the conditions of blockings are imposed.

If loading PESANTEUR is used with STAT_NON_LINE, Code_Aster Gauss points makes the sum of the nodal stresses starting from the stresses with the SIGM_ELGA. And that does not give the same thing as MECA_STATIQUE, because if one imposes, during a STAT_NON_LINE, with a node at the same time of the conditions of displacement and force (here coming from gravity), these forces is not taken into account. The only way of finding the weight of structure is:

- *To use MECA_STATIQUE*
- *During a use with STAT_NON_LINE to make so that the finite elements, on which kinematical conditions are imposed, are of a sufficiently small size so that their weight is negligible in front of that of total structure.*
- *During a use of beam elements with STAT_NON_LINE, a solution are to duplicate the nodes on which the kinematical condition is imposed and to make for example a LIAISON_DDL between the 2 nodes or to use the discrete ones.*

4.7 Operand ROTATION (AFFE_CHAR_MECA only)

```
|   ROTATION =_F   (     ♦VITESSE=omega          ,                [R]
                         ♦AXE=                (rear, Br, Cr)  ,        [l_R]
                         ◊CENTER      =    (X, there, Z),          [l_R]
                         ◊/MAILLE=lma                     ,
[l_maille]
                            /GROUP_MA      =lgma ,              [l_gr_maille]
                                              /TOUT=' OUI',
                    )
```

♦ VITESSE= Omega ,

Rotational speed

♦AXE= (rear, Br, Cr) ,

direction of the rotational axis which leads to:

$$\boldsymbol{\omega} = \omega \frac{(a_r \boldsymbol{i} + b_r \boldsymbol{j} + c_r \boldsymbol{k})}{\sqrt{a_r^2 + b_r^2 + c_r^2}}$$

The loading which results from it is: $\rho(\omega \wedge \mathbf{OM}) \wedge \omega$ where $\mathbf{O}$ is the origin of the coordinates and $\mathbf{M}$ a point running of structure with ρ definite density like characteristic of the material (see operators DEFI_MATERIAU [U4.43.01] and AFFE_MATERIAU [U4.43.03]).

```
◊CENTER    =  (X, there, Z),
```

If the center is not the origin (default), one can specify its coordinates (x, y, z).

Limitations:

- **plane modelizations:** the rotational axis must be in the direction Oz (normal direction with the plane), the center can be unspecified.
- **axisymmetric modelizations and Fourier:** the rotational axis must be in the direction Oy, the center must be the origin (if not the loading is not axisymmetric).

Notice important:

One can vary in time rotational speed by breaking up rotation in a multiplicative way between spatial loading and evolution into time $\omega(t)=\omega_0 f(t)$, then by multiplying the CHARGE by a multiplying function (key key FONC_MULT) in transient computation (DYNA_TRAN_MODAL, DYNA_LINE_TRAN, DYNA_NON_LINE). However, it is advisable to pay attention: the loading $\rho(\omega \wedge \mathbf{OM}) \wedge \omega$ being proportional to the square rotational speed $\omega(t)^2$, it is necessary to affect the square of the evolution in time $f(t)^2$, behind FONC_MULT.

4.8 Operand PRE_SIGM (AFFE_CHAR_MECA only)

```
|   PRE_SIGM =sigm   ,
```

Factor key word usable to apply a prestressing σ_{pre}. This loading makes it possible to apply average voluminal stresses, overall uniform (2D or 3D) with a voluminal field. The second calculated elementary member will be $\int_{Ve} \sigma_{pre} : \varepsilon(v^*) dV_e$.

The stress field sigm is of standard card or chamelem elga. It can come from CREA_CHAMP or be calculated in addition.

One should not confuse this prestressing with the initial stress σ_{ini} used in nonlinear, because this prestressing does not intervene directly in the statement of the constitutive law. This field of prestressings, is used like second member in the resolutions of MECA_STATIQUE and STAT_NON_LINE.

4.9 Key word DDL_IMPO

4.9.1 Drank

Factor key word usable to impose, with nodes introduced by one (at least) of the key keys: TOUT, NOEUD, GROUP_NO, MESH, GROUP_MA, one or more values of displacement (or certain associated quantities).

According to the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F).

During a computation with the method X-FEM, it is possible to impose the displacement of nodes nouveau riches. (AFFE_CHAR_MECA only). That is done in a usual way (although these nodes do not have a degree of freedom DX, DY or DZ).

Notice; if the required node is on the lips, then one imposes the condition of blocking on the nodes of the upper lips and lower.

4.9.2 Syntax

- for AFFE_CHAR_MECA

```
    |   DDL_IMPO=_F    (                  ♦/TOUT=' OUI',
                      /NOEUD        =lno     ,       [l_noeud]
                      /GROUP_NO     =lgno ,        [l_gr_noeud]
                      /MAILLE          =lma     ,       [l_maille]
                      /GROUP_MA     =lgma ,        [l_gr_maille]
                 ♦/     |  DX  =UX          ,        [R]
                        |  DY  =UY          ,        [R]
                        |  DZ  =UZ          ,        [R]
                        |  DRX =THETAX      ,    [R]
                        |  DRY =THETAY         ,    [R]
                        |  DRZ =THETAZ         ,    [R]
                        |  GRX =G       ,       [R]
                        |  PRES=p       ,       [R]
                        |  PHI =PHI         ,       [R]
                        |  TEMP=T       ,       [R]
                        |  PRE1=pr1         ,       [R]
                        |  PRE2=pr2         ,       [R]
                           …
                        |  LAGS_C=lag   ,       [R]
                        |  V11=v11      ,       [R]
                        |  V12=v12      ,       [R]
                        |  V21=v21      ,       [R]
                        |  V22=v22      ,       [R]
                        |  PRES11=pres11,    [R]
                        |  PRES12=pres12,    [R]
                        |  PRES21=pres21,    [R]
                        |  PRES22=pres22,    [R]
                      /LIAISON   =     ' ENCASTRE'
              )
```

the exhaustive list of the degrees of freedom which can be imposed is:

DX, DY, DZ, DRX, DRY, DRZ, GRX, NEAR, PHI, TEMP, PRE1, PRE2, UI2, UI3, VI2, VI3, WI2, WI3, UO2, UO3, VO2, VO3, WO2, WO3, UI4, UI5, VI4, VI5, WI4, WI5, UO4, UO5, VO4, VO5, WO4, WO5, UI6, UO6, VI6, VO6, WI6, WO6, WO, WI1, WO1, GONF, LIAISON, H1X, H1Y, H1Z, E1X, E1Y, E1Z, E2X, E2Y, E2Z, E3X, E3Y, E3Z, E4X, E4Y, E4Z, LAGS_C , V11, V12, V21, V22, PRES11, PRES12, PRES21, PRES22

- for AFFE_CHAR_MECA_F

```
  |   DDL_IMPO=_F    (                  ♦/TOUT=' OUI',
                           /NOEUD            =lno      ,        [l_noeud]
                           /GROUP_NO         =lgno ,        [l_gr_noeud])
                          /MAILLE            =lma      ,        [l_maille]
                          /GROUP_MA      =lgma ,        [l_gr_maille]
                      ♦/      |   DX=
                                   …                       [function]
                          /LIAISON       =   ' ENCASTRE'
                  )
```

4.9.3 Operands

| DDL_IMPO

All the specified values are defined in reference GLOBAL of definition of the mesh.

• DX = ux or uxf • DY = uy or uyf • DZ = uz or uzf	Value of the component of displacement in **translation** imposed on the specified nodes

Only if the specified nodes belong to discrete elements of translation - rotation, **beam** or **shell**:

• DRX = θ X or θ xf • DRY = θ there or θ yf • DRZ = θ Z or θ zf	Valeur of the component of displacement in **rotation** imposed on the specified nodes

Uniquement if the specified nodes belong to beam elements "POU_D_TG" :

• GRX = G or gf	Valeur of the warping of the beam

Only if the specified nodes belong to elements fluid or fluid structure:

• NEAR = p or acoustic	PF Pressure in the fluid (modelization "3D_FLUIDE")
• PHI = ϕ or ϕ F	Potential of displacements of the fluid (modelizations "3D_FLUIDE" and "FLUI_STRU")

Only if the specified nodes belong to surface elements free:

• DZ = uz or uzf	Displacement imposed of the free face (modelization "2D_FLUI_PESA")
• PHI = ϕ or ϕ F	Potential of displacements of the fluid (modelization "2D_FLUI_PESA")

Only if the specified nodes belong to elements THM:

• PRES= p	Pressure of interstitial fluid (modelizations "3D_JOINT_CT")
• TEMP= T	Temperature (modelizations "" with = 3D or AXIS or D_PLAN YYYY = THM or THHM or THH)
• PRE1= p1	Pression capillary or pressure of the fluid or the gas (modelizations "" with = 3D or AXIS or D_PLAN YYYY = THM or THHM or THH or HM or HHM)
• PRE2= p2	Pression of the gas (modelizations "" with = 3D or AXIS or D_PLAN YYYY = THH or THHM or HHM)

- `LH1=0` — Multiplying of lagrange hydraulic for the joined elements of type "`_JHMS`". Allows to neutralize the degrees of freedom at the edge of the joint if the solid mass of bearings is purely mechanical.

Only if the specified nodes belong to elements "`PIPE`".
These elements have 15 degrees of freedom of shell:

U : warping V, W : ovalization
I : "in plane" O : "out of planes"

Is:

- `UI2 VI2 WI2 UO2 VO2 WO2` Degrees of freedom related to the mode 2
- `UI3 VI3 WI3 UO3 VO3 WO3` Degrees of freedom related to the mode 3
- `WO WI1 WO1` Degrees of freedom of swelling and mode 1 on W

Only if the specified nodes belong to elements "`TUYAU_6M`".

- `UI4 VI4 WI4 UO4 VO4 WO4` Degrees of freedom related to the mode 4
- `UI5 VI5 WI5 UO5 VO5 WO5` Degrees of freedom related to the mode 5
- `UI6 VI6 WI6 UO6 VO6 WO6` Degrees of freedom related to mode 6

Only if the specified nodes belong to elements "`XXX_INCO`".

- `GONF` swelling

Only if the specified nodes belong to elements of regularization second gradient:

- `V11 V12 V21 V22` — Component of microscopic strain tensor
- `PRES11 PRES12 PRES21 PRES22` — Lagrange multipliers introduced for the mixed formulation

Only if the specified nodes belong to elements of regularization second gradient microphone-thermal expansion:

- `GONF` — Swelling
- `NEAR` — Multiplying of Lagrange introduces for the mixed formulation

`LIAISON = "ENCASTRE"`

Makes it possible to embed nodes directly, i.e. to force to zero the degrees of freedom of translation and rotation. The other degrees of freedom are not modified.

4.9.4 Checks and recommendations

It is checked that the specified degree of freedom exists in this node for the elements assigned in the model to meshes which contain the node.

However, if the same boundary condition is specified twice by two calls to `AFFE_CHAR_MECA` (for example, with two values of imposed displacement), that led to a matrix singular.

If it is specified twice (or more) in only one call to `AFFE_CHAR_MECA`, the rule of overload applies and an alarm message (indicating the overload) is transmitted.

4.10 Key word **`FACE_IMPO`**

4.10.1 Drank

Factor key word usable to impose, with all the nodes of a face defined by a mesh or a mesh group, one or more values of displacement (or certain associated quantities).
According to the name of the operator called, the values are provided directly (`AFFE_CHAR_MECA`) or via a concept `function` (`AFFE_CHAR_MECA_F`).

4.10.2 Syntax

- for `AFFE_CHAR_MECA`

```
      |  FACE_IMPO=_F  ( ♦/MAILLE=lma                    ,
[l_maille]
                            /GROUP_MA        =lgma ,
[l_gr_maille]
                         ◊SANS_MAILLE=lma1              ,
[l_maille]
                         ◊SANS_GROUP_MA=lgma1              ,
[l_gr_maille]
                         ◊SANS_NOEUD=lno1           ,         [l_noeud]
                              SANS_GROUP_NO=lgno1     ,          [l_gr_
node]
                         ♦/     |  DX=ux         ,                [R]
                                |  DY=uy         ,                [R]
                                |  DZ=uz         ,                [R]
                                |  DRX=          θ X,                [R]
                                |  DRY=          θ there,
[R]
                                |  DRZ=          θ Z,                [R]
                                |  GRX=g              ,           [R]
                                |  PRES=p        ,            [R]
                                |  PHI=phi            ,           [R]
                                |  TEMP=T        ,            [R]
                                |  PRE1=pr1           ,           [R]
                                |  PRE2=pr2           ,           [R]
                            /   |  DNOR=un            ,           [R]
                                |  DTAN=ut            ,           [R]
    )
```

- for `AFFE_CHAR_MECA_F`

```
      |  FACE_IMPO=_F  ( ♦/MAILLE=lma                    ,
[l_maille]
                            /GROUP_MA        =lgma ,
[l_gr_maille]
                         ◊SANS_MAILLE=lma1              ,
[l_maille]
                         ◊SANS_GROUP_MA=lgma1              ,
[l_gr_maille]
                         ◊SANS_NŒUD=lno1                ,
[l_noeud]
                              SANS_GROUP_NO=lgno1     ,          [l_gr_
node]
                         ♦/     |  DX=uxf             ,
[function]
                                |  DY=uyf             ,
[function]
                                |  DZ=uzf             ,
[function]
                                |  DRX=xf      θ  ,          [function]
```

```
                              |  DRY=yf      θ ,        [function]
                              |  DRZ=zf      θ ,        [function]
                              |  GRX=gf        ,
[function]
                              |  PRES=pf       ,
[function]
                              |  PHI=f       ϕ ,        [function]
                              |  TEMP=Tf       ,
[function]
                              |  PRE1=pr1f   ,          [function]
                              |  PRE2=pr2f   ,          [function]
                            / |  DNOR=un       ,
[function]
                              |  DTAN=ut       ,
[function])
```

4.10.3 Operands

```
◊SANS_MAILLE=lma1        ,      [l_maille]
◊SANS_GROUP_MA=lgma1     ,      [l_gr_maille]
◊SANS_NŒUD=lno1          ,      [l_noeud]
◊SANS_GROUP_NO=lgno1     ,      [l_gr_noeud]
```

Indicates that one wants to omit the nodes of the lists `lma1, lgma1, lno1, lgno1,` of the list `lma` or `lgma.`

Example:
```
FACE_IMPO = (  _F (  GROUP_MA =Gauche,
                     DX =0, DY =0),
               _F (  GROUP_MA =Haut,
                     SANS_GROUP_MA =Gauche,
                     DNOR =0),)
```

the meaning of the 2nd occurrence of `FACE_IMPO` is: “for all the nodes Top except those which belong on the left, `DNOR=0` ”.

This makes it possible not to have redundant boundary conditions.

```
♦/  |  DX  =
    |  DY  =
    |  DZ  =
    |  DRX =
    |  DRY =
    |  DRZ =
    |  GRX =
    |  PRES=
    |  PHI =
    |  TEMP=
    |  PRE1=
    |  PRE2=
```

the components, imposed on all the nodes belonging to meshes specified, are defined in **reference** `GLOBAL` of definition of the mesh.

The sides considered are made up:

- either of `TRIA3`, `TRIA6`, `QUAD4`, `QUAD8`, `QUAD9` in dimension 3,
- or of `SEG2` or `SEG3` in dimension 2 (the face is reduced on a board).

Note:

*The components of displacement in rotation `DRX, DRY, DRZ` can intervene only on nodes which belong to beam elements or of **shell** (see `DDL_IMPO` [§4.10]),*

component `GRX` on beam elements "`POU_D_TG`",

the components `NEAR` and `PHI` on elements of modelizations "`3D_FLUIDE`" and "`FLUI_STRU`", the components `DZ` and `PHI` on elements of modelization "`2D_FLUI_PESA`".
Components `TEMP`, `PRE1`, `PRE2` on elements of modelizations `THM`.

```
/  |  DNOR =
   |  DTAN =
```

the imposed components are defined according to the norm or the tangent with a mesh (**local coordinate system**).
`DNOR` : normal component (see [U4.44.01 §4.1]),
`DTAN` : tangential component (see [U4.44.01 §4.1]).

4.11 Key word LIAISON_DDL

4.11.1 Drank

Factor key word usable to define a linear relation between degrees of freedom of two or several nodes.

According to the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F).

4.11.2 Syntax

- for AFFE_CHAR_MECA

```
    LIAISON_DDL=_F (    ♦/NOEUD=lno                  ,           [l_noeud]
                           /GROUP_NO     =lgno ,            [l_gr_noeud]
                        ♦DDL=lddl        ,                        [l_K8]
                        ♦COEF_MULT=i          α ,                 [l_R]
                        ♦COEF_IMPO=           β ,                 [R]
                  )
```

- for AFFE_CHAR_MECA_F

```
    LIAISON_DDL=_F (    ♦/NOEUD          =lno              ,    [l_noeud]
                           /GROUP_NO  =        lgno,   [l_gr_noeud]
                        ♦DDL=lddl        ,                        [l_K8]
                        ♦/COEF_MULT=i                   α  ,    [l_R]
                           /COEF_MULT_FONC      =if   α  ,    [l_fonction]
                        ♦COEF_IMPO=f                    β  ,    [function]
                  )
```

4.11.3 Operands

GROUP_NO or THE NODE IS OUTSIDE THE FIELD OF DEFINITION WITH A RIGHT PROFILE OF THE EXCLU TYPE NODE: list nodes $N_i \quad (i=1,r)$ ordered in a natural way:

- in the order of the list of nodes groups, and for each nodes group, in the order of definition of the group by GROUP_NO,
- in the order of nodes list for NOEUD.

DDL : list degrees of freedom $U_i \quad (i=1,r)$ of r texts to be taken in the documentation of the simple quantities [U2.01.04]

COEF_MULT : list $\alpha_i \quad (i=1,r)$ coefficients (of real type for AFFE_CHAR_MECA **and** AFFE_CHAR_MECA_F).

COEF_MULT_FONC : list $\alpha_i \quad (i=1,r)$ coefficients of type function of the geometry only for AFFE_CHAR_MECA_F.

COEF_IMPO : coefficient β for AFFE_CHAR_MECA, function of time for AFFE_CHAR_MECA_F.

The following kinematical condition will be applied: $\sum_{i=1}^{r} \alpha_i U_i = \beta$

4.11.4 Component precautions of

4.11.4.1 use in rotation

the components of displacement in rotation `DRX, DRY, DRZ` can intervene only in combinations only **assigned** to nodes which belong to discrete elements of translation-rotation, **beam** or **shell** (see `DDL_IMPO` : cf [§4.10]).

4.11.4.2 Relation linear between the degrees of freedom of the same node

In this cas particulier, one will repeat behind the key word `NOEUD` the name of the node as many times as there are degrees of freedom in the relation. Example: to impose $U_x = U_y$ on the node $N1$, one will write:

```
LIAISON_DDL =_F   ( NOEUD     =    ("N1", "N1"),
                    DDL       =    ("DX", "DY"),
                    COEF_MULT =    (1. , -1.),
                    COEF_IMPO =0  .,                      )
```

4.11.4.3 Relation linear between nodes groups

It is important to note that to an occurrence of factor key word `the LIAISON_DDL` corresponds one and only one linear relation.

If one wants to impose the same relation between 2 nodes groups `GRN01` and `GRN02` (even node U_x displacement with node for example) **one cannot write** :

```
LIAISON_DDL = _F    ( GROUP_NO  =    ("GRNO1", "GRNO2"),
                      DDL       =    ("DX"   "DX"),
                      COEF_MULT =    (1. , -1.),
                      COEF_IMPO =0  .,                      )
```

This writing has meaning only if `GRNO1` and `GRNO2` contain each one one node. It will be necessary in the case above to clarify each linear relation, node by node, or to use `LIAISON_GROUP` [§4.14] which makes it possible to condense the writing of same linear relations between two nodes groups as screw - with - screw.

4.11.4.4 Multiplying coefficients geometry dependant

For `AFFE_CHAR_MECA_F`, one can return of the multiplying coefficients geometry dependant with `COEF_MULT_FONC`. Nevertheless, these coefficients are calculated from the initial geometry, it does not have there a possible reactualization in nonlinear.

4.12 Key word LIAISON_OBLIQUE

4.12.1 Drank

Factor key word usable to apply, with nodes or nodes groups, the same component value of displacement definite per component in an unspecified oblique coordinate system.
According to the name of the operator called, the values are provided directly (`AFFE_CHAR_MECA`) or via a concept `function` (`AFFE_CHAR_MECA_F`).

4.12.2 Syntax

- for `AFFE_CHAR_MECA`

```
|  LIAISON_OBLIQUE =_F  ( ♦/NOEUD         =no      ,      [node]
                           /GROUP_NO  =gno     ,      [gr_noeud]
                         ♦ |  DX  =UX          ,      [R]
                           |  DY  =UY          ,      [R]
                           |  DZ  =UZ          ,      [R]
                           |  DRX =X      θ  ,      [R]
                           |  DRY =Y      θ  ,      [R]
                           |  DRZ =z      θ  ,      [R]
                          ♦  ANGL_NAUT =   (α,β,γ),   [l_R]
                        )
```

- for `AFFE_CHAR_MECA_F`

```
ILIAISON_OBLIQUE  =_F   ( ♦/NOEUD         =no      ,      [node]
                           /GROUP_NO  =gno     ,      [gr_noeud]
                         ♦ |  DX  =uxf         ,
[function]
                           |  DY  =uyf         ,
[function]
                           |  DZ  =uzf         ,
[function]
                           |  DRX =xf      θ ,      [function]
                           |  DRY =yf      θ ,      [function]
                           |  DRZ =zf      θ ,      [function]
                         ♦ANGL_NAUT    =   (α,β,γ),   [l_R]
                        )
```

4.12.3 Operands

| `LIAISON_OBLIQUE`

• `DX` = `ux` or `uxf` • `DY` = `uy` or `uyf` • `DZ` = `uz` or `uzf`	Value of the component of displacement in translation in the oblique coordinate system imposed on the Only specified

nodes if the specified nodes belong to discrete elements of translation - rotation, beam or shell.

• `DRX` = θ `X` or θ `xf` • `DRY` = θ `there` or θ `yf` • `DRZ` = θ `Z` or θ `zf`	Valeur of the component of displacement in rotation in the oblique coordinate system imposed on the specified nodes

♦ANGL_NAUT = (α, β, γ),

the nautical angles (α, β, γ) defined in **degrees**, are the angles making it possible to pass from reference GLOBAL of definition of the coordinates of the nodes to an unspecified oblique coordinate system (see AFFE_CARA_ELEM [U4.42.01]).

4.12.4 Checking

One checks that the specified degree of freedom exists in this node for the elements assigned in the model to meshes which contain the node.

4.12.5 Limitation

In an occurrence of factor key word, one can introduce for time one node or one nodes group containing one node.

4.13 Key word LIAISON_GROUP

4.13.1 Drank

Factor key word usable to define the same linear relation between certain degrees of freedom of couples of nodes, these couples of nodes being obtained while putting in opposite two lists of meshes or nodes [§4.14.5].

According to the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F).

4.13.2 Syntax

- for AFFE_CHAR_MECA

```
LIAISON_GROUP=_F  ( ♦/♦/MAILLE_1=lma1                          ,
[l_maille]
                          /GROUP_MA_1      =lgma1   ,
[l_gr_maille]
                       ♦/MAILLE_2=lma2                ,
[l_maille]
                          /GROUP_MA_2      =lgma2   ,
[l_gr_maille]

                    /  ♦ /NOEUD_1          =lno1 ,   [l_noeud]
                          /GROUP_NO_1      =lgno1   ,
[l_gr_noeud]
                       ♦ /NOEUD_2          =lno2 ,   [l_noeud]
                          /GROUP_NO_2      =lgno2   ,
[l_gr_noeud]

                  ◊/SANS_NOEUD=lno                     ,
[l_noeud]
                     /SANS_GROUP_NO            =lgno ,
[l_gr_noeud]

                  ♦   DDL_1 =      |  "DX",
                                   |  "DY",
                                   |  "DZ",
                                   |  "DRX",
                                   |  "DRY",
                                   |  "DRZ",
                              /  "DNOR",
                  ♦   DDL_2 =      |  "DX",
                                   |  "DY",
                                   |  "DZ",
                                   |  "DRX",
                                   |  "DRY",
                                   |  "DRZ",
                              /  "DNOR",

                  ♦   COEF_MULT_1=         α1i  ,       [l_R]
                  ♦   COEF_MULT_2=         α2i  ,       [l_R]
                  ♦   COEF_IMPO=             β  ,         [R]
                                      ◊SOMMET=' OUI',
```

```
◊CENTER=lr ,
[l_R]
◊ANGL_NAUT=lr , [l_R]
◊TRAN=lr , [l_R]
)
```

- for AFFE_CHAR_MECA_F

```
LIAISON_GROUP=_F  ( ♦/♦/MAILLE_1=lma1                        ,
[l_maille]
                             /GROUP_MA_1       =lgm
[l_gr_maille]
                          ♦/MAILLE_2=lma2                ,    [l_maille]
                             /GROUP_MA_2       =lgma2    ,
[l_gr_maille]
                        /♦   /NOEUD_1          =lno1 ,    [l_noeud]
                             /GROUP_NO_1       =lgno1    ,    [l_gr_noeud]
                           ♦ /NOEUD_2          =lno2 ,    [l_noeud]
                             /GROUP_NO_2       =lgno2    ,    [l_gr_noeud]
                   ◊/SANS_NOEUD=lno                      ,    [l_noeud]
                      /SANS_GROUP_NO            =lgno ,    [l_gr_noeud]
                   ♦   DDL_1=/        |   ' DX',
                                   |   ' DY',
                                   |   ' DZ',
                                   |   ' DRX',
                                   |   ' DRY',
                                   |   ' DRZ',
                                /   "DNOR",
                   ♦   DDL_2=/        |   ' DX',
                                   |   ' DY',
                                   |   ' DZ',
                                   |   ' DRX',
                                   |   ' DRY',
                                   |   ' DRZ',
                                 /  "DNOR",
                   ♦   COEF_MULT_1=1i      α ,               [l_R]
                   ♦   COEF_MULT_2=2i      α ,               [l_R]
                   ♦COEF_IMPO=f            β   ,
[function]
                                   ◊SOMMET=' OUI',
                   ◊CENTER=lr                   ,                 [l_R]
                   ◊ANGL_NAUT=lr              ,                 [l_R]
                   ◊TRAN=lr                   ,                 [l_R]
                   )
```

4.13.3 Operands

/♦/GROUP_MA_1 =
/MAILLE_1 =

These operands define the first list of meshes in relation (noted Γ_1).

♦/GROUP_MA_2 =
/MAILLE_2 =

These operands define the second list of meshes in relation (noted Γ_2).

♦/GROUP_NO_1 =
/NOEUD_1 =

These operands define the first in relation nodes list.

♦/GROUP_NO_2 =
/NOEUD_2 =

These operands define the second in relation nodes list.

The two lists must have the same length.

◊/SANS_GROUP_NO =
/SANS_NOEUD =

These operands make it possible to remove list of the couples of nodes as screw - with - screw [§4.14.5] all the couples of which at least one of the nodes belongs to nodes list described by these operands.
That makes it possible factor key word to avoid the accumulation of linear relations on the same node during various repetitions of the LIAISON_GROUP, which leads most of the time to a matrix singular.

♦DDL_1 (_2) =

the argument of DDL_1 or _2 must be a list of texts taken among (DX', "DY", "DZ", "DRX", "DRY", "DRZ") or "DNOR".

♦COEF_MULT_1 (resp. COEF_MULT_2) =

List of realities dimensioned exactly with the number of degrees of freedom declared in DDL_1 (resp. DDL_2) corresponding to the multiplying coefficients of the linear relation.

♦COEF_IMPO =

Coefficient of blocking of the linear relation:

β : reality for AFFE_CHAR_MECA
βf : function for AFFE_CHAR_MECA_F

operands CENTER / ANGL_NAUT / TRAN make it possible to define a virtual transformation (rotation and/or translation) approximate Γ_1 in Γ_2 order to ensuring the bijectivity of the function opposite [§4.14.5].
The command carries out initially rotation, then the translation.

Coordinated ◊CENTER= of the center of rotation (in the total reference)
nautical ◊ANGL_NAUT= angles defining rotation (in degrees)
◊TRAN= component of vector translation

the Note:

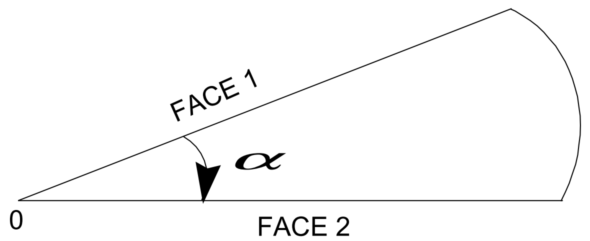
- *It is checked that the degrees of freedom specified in these operands exist for each node of the elements affected in the model to meshes which contain the node,*
- *using argument "DNOR", there is compulsory to have declared the edges using meshes and which the computation of a norm on these meshes is possible.*

◊SOMMET = ' OUI'

When meshes of edge are quadratic (thus SEG3) the use of SOMMET: "OUI' forces the algorithm of pairing to associate the tops of the SEG3 with other tops, and the mediums of the SEG3 in other mediums. In the case of fine meshes, that makes it possible in certain cases to avoid the problems of conflicts of opposite.

4.13.4 Example of use

One wants to impose a cyclic condition of repetitivity (even normal displacement) between face 1 and face 2 of the geometry below:



Let us suppose that FACE1 (respectively FACE2) is made up of the list of meshes lma1 (resp. lma2).

One wants to write the following linear relations:

$\forall N_i^1$ node of the face 1 of opposite N_i^2

$\boldsymbol{u.n}(N_i^1)=\boldsymbol{u.n}(N_i^2) \qquad \forall i=1,\ldots,$ *nbno*

where nbno is the number of nodes of face 1 (and the face 2).

The data of LIAISON_GROUP will be written:

```
LIAISON_GROUP=_F  (  MAILLE_1=lma1         ,
                     MAILLE_2=lma2         ,
                                  DDL_1=' DNOR',
                                  DDL_2=' DNOR',
                     COEF_MULT_1=1    .,
                     COEF_MULT_2=-1       .,
                     COEF_IMPO=0          ,
                     CENTER=               (X0, Y0, Z0),
                     ANGL_NAUT=            (α , 0. , 0.),
                  )
```

4.13.5 Determination of the couples of nodes in opposite

It are in the same way made that in AFFE_CHAR_THER.

Initially, one draws up the two lists of nodes to be put in opposite (IE to be paired), for each occurrence of factor key word the LIAISON_GROUP :

- for key words GROUP_NO_1 and GROUP_NO_2, they are the nodes setting up the nodes groups,
- for key words GROUP_MA_1 and GROUP_MA_2, they are the nodes of meshes setting up the mesh groups.

The redundancies being eliminated, the two lists of nodes obtained must have the same length.

The determination of the couples of nodes in opposite is done in several stages:

- for each node $N1$ of the first list, one seeks the node image $N2=f(N1)$ of the second list. If f is not injective (a node $N2$ is the image of two distinct nodes $N1$ and $N1'$), the following error message is transmitted:

  ```
  <F> <MODELISA8_85> CONFLICT BETWEEN THE FACE-TO-FACE NODES
  LE NOEUD N2 Is facing the nodes N1 AND N1'
  ```

- for each node $N2$ of the second list, one seeks the node image $N1=g(N2)$ of the first list. If g is not injective (a node $N1$ is the image of two distinct nodes $N2$ and $N2'$), the following error message is transmitted:

  ```
  <F> <MODELISA8_85> CONFLICT BETWEEN THE FACE-TO-FACE NODES
  LE NOEUD N1 Is facing the nodes N2 AND N2'
  ```

- one checks that $g=f^{-1}$, i.e. the couples obtained by the stages a) and b) are the same ones (one wants to have a bijection f between the two lists of nodes). If f is not surjective, the following error message is transmitted:

  ```
  <F> <MODELISA8_88> CONFLICT BETWEEN THE FACE-TO-FACE NODES GENERATE
  SUCCESSIVELY FROM LISTS LIST1 AND LIST2
  LE NOEUD OF the FIRST LISTE N1 IS NOT the IMAGE Of AUCUN NOEUD PAR
  CORRESPONDENCE INVERSE
  ```

For a given N node, one calls nodes list node $f(N)$ image the node of the other which carries out the minimum of distance with N. To facilitate pairing, in particular in the case of particular geometries (where the borders Γ_1 and Γ_2 could "almost" result one from the other by the composition of a translation and of a rotation), one gives the opportunity of making a virtual geometrical transformation of the first nodes group (translation and rotation before you calculate the distances (key words TRAN, CENTER and ANGL_NAUT).

For each occurrence of factor key word the LIAISON_GROUP, one thus builds the list of the new couples in opposite. When all the occurrences were swept, one removes list the couples in double.

Note:

In the couples of nodes in opposite, the order of the nodes is important. So for the first occurrence of LIAISON_GROUP , a node N belonged to the first nodes group and a node M with the second group of node, and that for the second occurrence of LIAISON_GROUP , it is the reverse, one will obtain at the conclusion of pairing the couples (N, M) and (M, N). They will not be eliminated during detection of the redundancies; on the other hand, the matrix obtained will be singular. Thus, one advises to keep same logic during the description of edges as screw - with - screw.

4.14 Key word `LIAISON_MAIL`

4.14.1 Drank

Factor key word usable to define linear relations making it possible “to restick” two “edges” of a structure.

The characteristic of this key word (compared to `LIAISON_GROUP` for example) is to make it possible to bind displacements of unconstrained nodes on the mesh. The meshes of `FACE1` and `FACE2` can be incompatible.

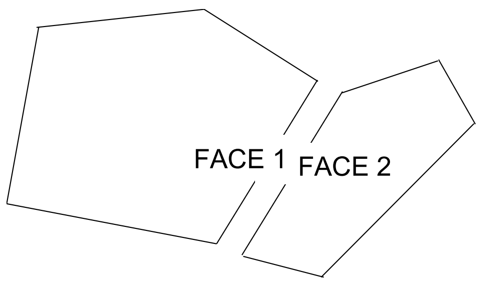
Note: The experiment showed that for computations of periodic homogenization, the results are much more precise if the 2 sides have compatible meshes (i.e the meshes of FACE `1 and` FACE `2 are` superposable modulos a isometry). Examples

: a)

a condition of periodicity (study of a cell of homogenization) b)

a cyclic condition of repetitivity c)

a condition of simple resticking In



the continuation of this paragraph, one will speak about the face “slave” (FACE2`) and` about the face “Master” (FACE1`)`.

The “resticking” of the 2 sides will be done by writing of linear relations between the degrees of freedom of the 2 sides.

Displacements of the nodes of slave face will be connected to displacements of their projections on the face Master. For each node of slave face, one will write 2 (in 2D) or 3 (in 3D) linear relations. If

FACE `1 and` FACE `2` are not geometrically confused but that there exists a isometry (rotation + translation) between the two, the user must define this isometry (that which transforms FACE `2 opposite 1).`

An application of this functionality is for example the resticking of a mesh formed by linear elements on $(P1)$ another quadratic mesh. $(P2)$ In this case it is rather advised to choose like face "slave" the quadratic face. Syntax

4.14.2 (in AFFE_CHAR_MECA only) LIAISON

```
_MAIL=_F   (◊
      TYPE_RACCORD=/         “MASSIF  ” [DEFAULT                        ]/
                          “COQUE  ”/
                          “COQUE_MASSIF  ”/
                          “MASSIF_COQUE  ” ♦

   |     GROUP_NO_ESCL=lgno2       , [
   l_gr_noeud] |NOEUD
         _ESCL=lno2         , [l_noeud                        ] |
GROUP
         _MA_ESCL=lgma2         , [                        l_gr_maille]
|NET
         _ESCL=lma2         , [l_maille                       ] ♦
   |     GROUP_MA_MAIT=lgma1       , [
   l_gr_maille] |NET
         _MAIT=lma1         , [l_maille                       ] #

   if TYPE_RACCORD = “MASSIF”: ◊
   ◊  |♦CENTER=        (xc          , yc, [zc]), [l_R              ] ♦
             ANGL_NAUT= (alpha         , [beta, gamma]), [l_R        ] |
   ♦
          TRAN=   (tx            , ty, [tz]), [l_R                ] ◊
          ♦DDL_MAIT=' DNOR'             , ♦
          DDL_ESCL=' DNOR'           , #

   if TYPE_RACCORD = “COQUE_MASSIF”: ♦
   EPAIS=epais                , [l_R                          ] CHAM
      _NORMALE=chanor        , [cham_no                       ] ◊

      ELIM_MULT=/“           NON    ”, [DEFAULT                    ]/“
                             OUI'   ,)
          Operands
```

4.14.3 Choice

4.14.3.1 of surface slave and surface Master

the principle of connection is to eliminate the degrees of freedom slaves by writing them like linear relations from the main degrees of freedom. There is a certain symmetry in the problem and one could believe that one can choose randomly who will be the Master and who will be the slave. Actually

, it is necessary to be attentive on two particular points:

- Syntax is not symmetric: side slave, the user must specify the nodes “to be welded”, whereas main side, it must give meshes. Moreover, the meshes main ones are (for time) of a topological dimension with what would be natural. For example, for a 2D mesh, surfaces to be restuck are lines, and one could expect that the meshes main ones are segments. The code expects the meshes surface ones (quadrangles and triangles). It
- is preferable (from a mechanical point of view) to choose like surface slave surface with a grid most finely. In the same way that when 2 sheets are welded, it is to better multiply the weld points. TYPE_RACCORD

4.14.3.2 This

key word makes it possible to choose the type of the linear relations which one will write to eliminate the degrees of freedom from the nodes slaves. If

- TYPE _RACCORD=' MASSIF', the nodes are supposed to carry degrees of freedom of translation (DX, DY, DZ). If the user does not specify DDL _MAIT=' DNOR', one will write (for example in 2D), 2 linear relations for each slave node: one to eliminate its "DX ", the other to eliminate its "DY ". If

- TYPE _RACCORD=' COQUE', the nodes are supposed to carry degrees of freedom of translation (DX, DY, DZ) and degrees of freedom of rotation (DRX, DRY, DRZ). One will write 6 linear relations to eliminate the 6 degrees of freedom from each slave node. If

- TYPE _RACCORD=' MASSIF_COQUE', the nodes slaves are supposed "massive" (translations: DX , DY, DZ) and the main nodes are supposed of standard "shell" (3 translations and 3 rotations).

The degrees of freedom of translation of the nodes slaves are eliminated by writing that they are equal to the translations of the "main" point in opposite. The translations of the main point are calculated as if the small segment of norm to the shell remained rigid. If

- TYPE _RACCORD=' COQUE_MASSIF', the nodes slaves are supposed of standard "shell" (6 degrees of freedom: DX , DY, DZ, DRX, DRY, DRZ) and the main nodes are supposed of "massive" type (DX, DY, DZ).

The degrees of freedom of translation of the nodes slaves are eliminated by writing that they are equal to the translations of the "main" point in opposite.
The degrees of freedom of rotation of the nodes slaves are eliminated by writing that they are equal to rotations of the "main" point in opposite. (A) Rotations of the point A are calculated starting from the translations of two other points and $A1$ located $A2$ at and $+h/2$ if $-h/2$, is h a normal vector with the shell and of which the length is the thickness of the shell (see key keys EPAIS and CHAM_NORMALE). GROUP_NO_ESCL

4.14.3.3 / NOEUD_ESCL / GROUP_MA_ESCL / MAILLE_ESCL These

key words make it possible to define all the nodes of slave face. One takes all the nodes specified by key words GROUP_NO_ESCL and NOEUD_ESCL more all the nodes carried by meshes specified by key words GROUP_MA_ESCL and MAILLE_ESCL . Note:

When

one wants to restick only normal displacements of the sides (cf key words DDL_MAIT and DDL_ESCL), it is necessary to be able to determine the normal direction of the sides. The normal direction is calculated on slave face. It is thus necessary in this case to use key words GROUP_MA_ESCL and MAILLE_ESCL with meshes of type "facets". GROUP_MA_MAIT

4.14.3.4 / MAILLE_MAIT These

key words make it possible to define the group of meshes where they with respect to the nodes of slave face will be sought. Caution:

In

3D, one should not give meshes of surface, but meshes the voluminal adjacent ones with the face. Meshes specified are "candidates" for the search of the points opposite. One can give too much of it, that is not awkward. In the same way

, in 2D, meshes the "main ones" must be surface (QUAD, SORTED) and nonlinear CENTER

4.14.3.5 / ANGL_NAUT / TRAN These

key words make it possible to define the geometrical transformation (rotation and/or translation) making it possible to pass from slave face to the face Master. If

these key words are absent, it is that the geometrical transformation is "the identity" i.e. the sides Master and slave are geometrically confused. It should be noted that

the program carries out initially rotation and then the translation. Caution: the meaning of the transformation is slave towards Master. DDL_MAIT

4.14.3.6 / DDL_ESCL If

one wants to restick only normal displacements with the sides, it is necessary to specify: DDL
`_MAIT=' DNOR'     DDL`
`_ESCL=' DNOR'     Note`

:

The normal direction is calculated on slave face (it is necessary to give meshes of facet). This normal direction is transformed by the possible rotation of the geometrical transformation to determine the normal direction on the face Master. Remarks

4.14.3.7

key word LIAISON_MAIL is made in theory to connect 2 surfaces disjoined a priori. Sometimes it is not the case and a slave node can belong to the one of meshes main. The linear relation that the problem seeks to write becomes a tautology (), $X=X$ which leads to a null pivot during factorization.

To avoid this problem, one does not write the relations connecting a slave node to his master mesh if: this

- node belongs to connectivity of the mesh
- key keys CENTER , ANGL `_TRAN`, TRAN were not used It

is necessary to be conscious that for each occurrence of LIAISON_MAIL , `one` TOUS connects the nodes slaves to meshes main even if the distances from projection are important (one emits however alarms in this case). It

would be an error to write: LIAISON_MAIL

```
= (  _F   ( GROUP_MA_ESCL=' GE', GROUP_MA_MAIT = ' GM1'), _F
                           (GROUP_MA_ESCL=' GE', GROUP_MA_MAIT = ' GM2')) by
```

thinking that the program will sort in GE the nodes close to and $\mathrm{GM1}$ those close to. $\mathrm{GM2}$ In this example, the nodes of GE will be eliminated 2 times and one can expect a problem of null pivot during factorization.

The user must write: LIAISON_MAIL

```
= _F  (GROUP_MA_ESCL=' GE', GROUP_MA_MAIT= ("GM1", "GM2")) CHAM_NORMALE
```

4.14.3.8 = chnor, EPAIS = thick These

two key keys are compulsory if TYPE_RACCORD = `"COQUE_MASSIF". Thick`
`is` the thickness of the shell on the level of connection (presumedly constant). Chnor
`is` a field at nodes which meshes contains the direction of the norm to the shell on the nodes of "main".
The field chnor `can` be obtained by the command: CHNOR

```
= CREA_CHAMP (TYPE_CHAM = "NOEU_GEOM_R", OPERATION = "NORMAL", MODELS
                        = MODEL, GROUP_MA = "GMCOQU") ELIM
```

4.14.3.9 _MULT= "OUI'/"NON" [DEFAULT] This

key word is used to solve the problem which can be posed when several surfaces adjacent slaves are restuck (i.e which have one or more common nodes). Let us imagine

for example that one writes (in 2D): LIAISON

```
_MAIL= (_F (GROUP
        _MA_ESCL=' LIGNE_AB', GROUP  _MAIT=…) _F (GROUP
        _MA_ESCL=' LIGNE_BC', GROUP  _MAIT=…) If
```

the user forces ELIM_ `MULT=' OUI', the program` will treat each occurrence of independent `LIAISON_MAIL of` way. The node, pertaining B to LINE `_AB and LINE _BC` will be eliminated 2 times and it is unfortunately probable that computation will stop during the factorization of the matrix with the message "Pivot almost no one…" because the linear relations generated by LIAISON `_MAILLE are` redundant. Most of the time

, default (ELIM _MULT=' NON') is the good choice. The only case where the user could use ELIM_ MULT=' OUI' is that of the use of the key word DDL_ESCL = ' DNOR' because so in the 2 occurrences, normal "the slaves" are not the same ones, elimination is not redundant. Key word

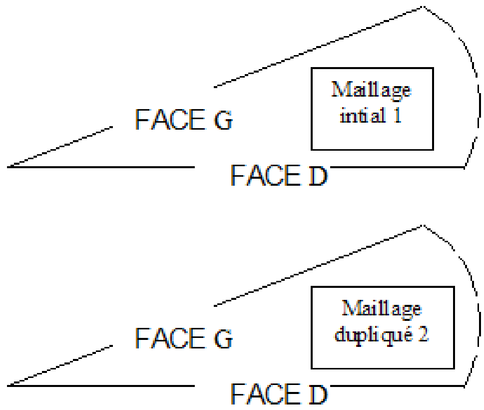
4.15 LIAISON_CYCL Drank Factor key word

4.15.1

usable to define the linear relations making it possible to impose conditions of cyclic symmetry with taking into account of a phase shift. It is mainly dedicated to being used in the restrictive frame of dynamic computation with cyclic symmetry. The characteristic

of this key word (with the image of LIAISON_MAIL) is to make it possible to bind displacements of unconstrained nodes on the mesh. The meshes of and can $FACEG$ $FACED$ be incompatible. The cyclic

condition of repetitivity applied in the frame of the dynamics is based on the method of duplication of mesh. The operator thus leaves on the postulate that the initial mesh of a sector is duplicated in two meshes identical to the image of the following figure. In



the continuation of this paragraph, one will speak about the face “slave” and the face “Master”. The “resticking” of the 2 sides will be done by writing of linear relations between the degrees of freedom of the 2 sides. Displacements

of the nodes of slave face will be connected to displacements of their projections on the face Master. For each node of slave face, one will write 2 (in 2D) or 3 (in 3D) linear relations. If and

$FACEG$ $FACED$ are not geometrically confused but that there exists a isometry (rotation + translation) between the two, the user must define this isometry (that which transforms into) $FACEG$. Note: $FACED$

An application

of this functionality is for example the resticking of a mesh formed by linear elements on another $(P1)$ *quadratic mesh. In* $(P2)$ *this case it is rather advised to choose like face “slave” the quadratic face. The statement*

of the condition of cyclic symmetry for a phase shift AND element given β and while regarding as G the interface slave is the following one: In order to

$$\begin{bmatrix} q_g^1 \\ q_g^2 \end{bmatrix} = \begin{bmatrix} \cos\beta & \sin\beta \\ -\sin\beta & \cos\beta \end{bmatrix} \begin{bmatrix} q_d^1 \\ q_d^2 \end{bmatrix}$$

write the linear relations making it possible to take into account this condition, it is necessary to give two **occurrences** of the key word factor LIAISON_CYCL : The first

- makes it possible to bind the degrees of freedom of the face of mesh G 1 with the face of the same D mesh and the face of mesh D 2. The coefficients (and) $\cos\beta$ must $\sin\beta$ be indicated by key words COEF_MAIT1 , COEF_MAIT2 . The second
- makes it possible to bind the degrees of freedom of the face of mesh G 2 with the face of the same D mesh and the face of mesh D 1. The coefficients (and) $-\sin\beta$ must $\cos\beta$ be indicated by key words COEF_MAIT1 , COEF_MAIT2 . Syntax

4.15.2 (in AFFE_CHAR_MECA only) LIAISON

```
_CYCL=_F (♦       |
    GROUP    _NO_ESCL=lgno2       ,    [l_gr_noeud                        ] |
NOEUD
         _ESCL=lno2,       [  l_noeud                          ] | GROUP
         _MA_ESCL=lgma2       ,    [l_gr_maille                      ]     |
NET
         _ESCL=lma2,       [  l_maille                         ] ♦ |
    GROUP    _MA_MAIT1=lgma1  ,    [l_gr_maille                      ]     |
NET
         _MAIT1=lma1,      [  l_maille                         ] | GROUP
         _MA_MAIT2=lgma2  ,    [l_gr_maille                      ] | NET
         _MAIT2=lma1,      [  l_maille                         ] ◊ |
    ♦  CENTER      = (xc,         yc , [zc]), [l_R]                  ♦ ANGL
          _  NAUT= (alpha        , [beta, gamma]), [l_R]          | ♦
       TRAN   = (tx,          ty , [tz]), [l_R]             ◊ |
    ♦  COEF   _  MAIT1=, [R]       α  |                  ♦
       COEF_MAIT2       =, [R ]    β  |                  ♦
       COEF_ESCL        =, [R ]    χ  ◊                  ♦
    DDL_MAIT    = ' DNOR'            , ♦ DDL_ESCL
         = ' DNOR'           ,) Operands
```

4.15.3 GROUP_NO_ESCL

4.15.4 /NOEUD_ESCL/GROUP_MA_ESCL/MAILLE_ESCL These key words

makes it possible to define all the nodes of slave face. One takes all the nodes specified by key words GROUP_NO_ESCL and NOEUD_ESCL more all the nodes carried by meshes specified by key words GROUP_MA_ESCL and MAILLE_ESCL . Note:

When

> *one wants to restick only normal displacements of the sides (cf key words DDL_MAIT and DDL_ESCL), it is necessary to be able to determine the normal direction of the sides. The normal direction is calculated on slave face. It is thus necessary in this case to use key words GROUP_MA_ESCL and MAILLE_ESCL with meshes of type "facets". GROUP_MA_MAIT1*

4.15.5 /MAILLE_MAIT1 These key words

make it possible to define the group of meshes main mesh 1 (or 2) where they with respect to the nodes of mesh 1 or 2 will slave face be sought. Caution:

In 3D

> *, one should not give meshes of surface, but meshes the voluminal adjacent ones with the face. Meshes specified are "candidates" for the search of the points opposite. One can give too much of it, that is not awkward. In the same way*

, in 2D, meshes the “main ones” must be surface (QUAD, SORTED) and nonlinear GROUP_MA_MAIT2

4.15.6 /MAILLE_MAIT2 These key words

make it possible to define the group of meshes of 1 (or 2) where they with respect to the nodes of mesh 1 or 2 will slave face be sought. Caution:

In 3D

, one should not give meshes of surface, but meshes the voluminal adjacent ones with the face. Meshes specified are "candidates" for the search of the points opposite. One can give too much of it, that is not awkward. In the same way

, in 2D, meshes the "main ones" must be surface (QUAD, `SORTED`) `and non`linear CENTER

4.15.7 /ANGL_NAUT/TRAN These key words

make it possible to define the geometrical transformation (rotation and/or translation) making it possible to pass from slave face to the face Master. If these

key words are absent, it is that the geometrical transformation is "the identity" i.e. the sides Master and slave are geometrically confused. It should be noted that

the program carries out initially rotation and then the translation. Caution: the meaning of the transformation is slave towards Master. COEF_MAIT1

4.15.8 /COEF_MAIT2/COEF_ESCL These key words

make it possible to define the coefficients of the linear relation to apply, in the case of cyclic symmetry it is the cosine and sines the angle phase shift AND element considered. These coefficients must thus be coherent with the definition of the interfaces Masters and slaves. Coefficient COEF_ESCL `makes it possible` to pass a coefficient in front of the degrees of freedom slaves. For example:

DDL_MAIT

$$\text{COEF\_ESCL}\left(q_g^1\right)=\left[\text{COEF\_MAIT1}\times\text{COEF\_MAIT2}\right]\begin{bmatrix}q_d^1\\ q_d^2\end{bmatrix}=\left[\cos\beta\cdot\sin\beta\right]\begin{bmatrix}q_d^1\\ q_d^2\end{bmatrix}$$

4.15.9/DDL_ESCL If one

wants to restick only normal displacements with the sides, it is necessary to specify: DDL_MAIT

```
= ' DNOR'      DDL_ESCL
= ' DNOR'      Note
```

: The normal

direction is calculated on slave face (it is necessary to give meshes of facet). This normal direction is transformed by the possible rotation of the geometrical transformation to determine the normal direction on the face Master. Key word

4.16 FORCE_NODALE Drank Factor key word

4.16.1

usable to apply, with nodes or nodes groups, nodal forces, definite component by component in reference GLOBAL or an oblique coordinate system defined by three nautical angles. According to

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.16.2 for

- AFFE_CHAR_MECA FORCE

```
_NODALE=_F (♦|NOEUD =lno, [l_noeud ] |
GROUP_NO
=lgno, [ l_gr_noeud ] ♦ |
FX= fx , [R] | FY=
fy , [R] | FZ=
fz , [R] | MX=
MX , [R] | MY=
my , [R] | MZ=
mz , [R] ◊ ANGL
_ NAUT= () [l_R α, β, γ ] ), for
```

- AFFE_CHAR_MECA_F FORCE

```
_NODALE=_F (♦|NOEUD =lno, [l_noeud ] |
GROUP_NO
=lgno, [ l_gr_noeud ] ♦|FX
=fxf , [function
] |FY=fyf
, [function ] |
FZ=fzf
, [function ] |
MX=mxf
, [function ] |
MY=
myf , [function
] |MZ=mzf
, [function ]
◊ANGL
_NAUT= (_f, _f , α _f) β , [l_fonction γ
]), Operands
```

4.16.3 fx, fy

, fz, MX, my, mz or fxf
, fyf, fzf, mxf, myf, mzf Values

of the components of the nodal forces applied to the specified nodes. These nodal forces will come to be superimposed on the nodal forces resulting, possibly, other loadings. Into axisymmetric, the values correspond to a sector of 1 radian (to divide the real loading by). (2π) or

α, β, γ (
_f, _f α , _f) β Lists γ

of the 3 angles, in degrees, which define the oblique coordinate system of application of the nodal forces (the last angles of the list can be omitted if they are null). The nautical angles make it possible to pass from the total reference of definition of the coordinates of the mesh to an unspecified oblique coordinate system (see operator AFFE_CARA_ELEM [U4.42 .01]). By default the angles are identically null and thus the components of forces are defined in reference GLOBAL . Key word

4.17 LIAISON_SOLIDE Drank Factor key word

4.17.1

making it possible to model an indeformable part of a structure. One imposes

linear relations between the degrees of freedom of the nodes of this indeformable part so that relative displacements between these nodes are null and one imposes possibly displacements on the values resulting from the translation and/or rotation. These nodes

are defined by the mesh groups, meshes, the nodes groups or nodes list to which they belong. Syntax

4.17.2 for

- AFFE_CHAR_MECA and AFFE_CHAR_MECA_F LIAISON_SOLIDE

```
=_F (♦/MAILLE
                    =lma, [      l_maille              ] /
GROUP_MA
                    =lgma,     [l_gr       _maille] /NOEUD
                 =lno   , [    l_noeud                ]
/GROUP_NO
                    =lgno,  [     l_gr_noeud      ]
◊NUMÉRIQUE

                 _LAGR      = "NORMAL        ", [DEFAULT
   ]/"APRES
                                     ", ◊ |
                 ♦ CENTER         = (xc, yc, [zc], [l_R]
    ♦ANGL
                    _NAUT=  (alpha, [beta, gamma]), [l_R]|
                 ♦TRAN=     ( tx, ty, [tz]), [l_R]       ◊DIST
                 _MIN=dmin    , [    R]),               Operands
```

4.17.3 ◊NUMÉRIQUE

_LAGR : If "

- NORMAL ", the two Lagrange multipliers associated with the relation will be such as the first will be located before all the terms implied in the relation and the second after, in the assembled matrix. If "
- APRES ", the two Lagrange multipliers associated with the relation will be located after all the terms implied in the relation, in the assembled matrix. This choice

  has the advantage of having an assembled matrix whose obstruction is weaker but has the disadvantage to be able to reveal a singularity in the matrix. Note:

  In a general

  way, one imposes: in 2D

  - *relations* $(nb_{ddl} \times nb_{noeud} - 3)$ *in 3D*
  - *relations* $(nb_{ddl} \times nb_{noeud} - 6)$ *where is*

- nb_{ddl} *the number of degrees of freedom per node, is*
- nb_{noeud} *the number of nodes of the list given after LIAISON_SOLIDE* `since`

a solid is determined by the position of one of its points and a reference in this point. Relations

are written by taking the vectorial formula representing a rigid body motion into small rotations: where

$$\vec{u}(M)=\vec{u}(A)+\vec{\Omega}(A)\wedge\overrightarrow{AM}$$

is A *an arbitrary node of solid. ◊CENTER*

`/ANGL_NAUT /TRAN : These` key words

make it possible to define the geometrical transformation (rotation and/or translation) making it possible to determine the displacements imposed on structure. If these

key words are absent, imposed displacements are null. It

is currently disadvised D" using key words CENTER `and ANGL_NAUT . TRAN=`

`(tx, ty, [tz]): components` of the translation imposed on structure. ◊DIST

`_ MIN: dmin This key word`

is used to define a distance (in the units of the mesh) below which one considers that the points of the mesh coincide. This distance is also used to determine points so are aligned, i.e. if they are in a cylinder of diameter lower than. By default $dmin$

, where $dmin=0.001*armin$ is $armin$ the smallest edge of the mesh. Key word

4.18 LIAISON_ELEM Drank By

4.18.1 calling

"massive part" a piece of structure modelled with isoparametric elements 3D, this factor key word makes it possible to model the connection: of

- a massive part with a part beam [R3.03.03] or a pipe section [R3.08.06], of
- a shell part with a part beam [R3.06.03] or a pipe section [R3.08.06]. This key word

also makes it possible 2D to connect edge of a structure with a beam or a discrete element. The goal

of this functionality is not to give an account of the scales length between the parts to be connected but to allow a simplification of the modelization by replacing a massive or surface part by a beam part for example. The connection

is treated by forcing linear relations between the degrees of freedom of the nodes of the junction of the two parts to be connected, without imposing superfluous relations. Syntax

4.18.2 (AFFE_CHAR_MECA only) LIAISON_ELEM

```
=_F (♦/OPTION
        = "3D_POU              ",/"3D_TUYAU
                             ",/"COQ_POU
                             ",/"COQ_TUYAU
                             ",/"PLAQ_POUT_ORTH
                             ",/"2D_
                          LOUSE  ", # beginning
   conditionoption   == "PLAQ_POUT_ORTH" ◊EXCENT
      _POUTRE=/"NON    "   ,   /"OUI'
                          ,    [DEFAULT
   ] # fine
   condition # beginning
   conditionoption   == "COQ_POU" or option
                        == "COQ_TUYAU" or option
                        == "3D_TUYAU" ◊AXE
   _POUTRE=   (X, there,       Z  ), [l_R]
       ◊CARA
   will _ELEM=cara  ,          [  cara_elem
        ] # end
   condition ♦/MAILLE
        _1=lma1,      [  l_maille
   ] /GROUP_MA_1
        =lgma1 ,   [l_gr_maille                                 ] ♦ /NOEUD_2
        =lno2,     [  l_noeud                                 ] /GROUP_NO_2
        =lgno2 ,   [l_gr_noeud                                 ] ◊NUMÉRIQUE
   _LAGR=/   "NORMAL                 ", [DEFAULT
        ]/"APRES
                             ", ◊ANGL
   _MAX=/1   ., [          DEFAULT
        ] /angl
                          , [R]),                                    Operands
```

4.18.3 of option "3D_POU " ♦OPTION

```
= "3D_POU" This
```

option makes it possible 3D to connect a massive part with a part modelled with beams of Eulerian or Timoshenko. ♦/MAILLE

```
_1= /GROUP_MA_1
= These    operands
```

define meshes surface massive part modelling the trace of the section of the beam on this massive part. These meshes must be assigned by of the finite elements sides of elements 3D before. ♦ /NOEUD_2

= /GROUP_NO_2
= These operands

define the node of the beam to be connected to the massive part. Thus if NOEUD_2 is used , one should give one node and if GROUP_NO_2 is used , one should give one group, this one containing one node. Precaution for use

: The massive

part must be with a grid with quadratic elements because the coefficients of the relations to be imposed are numerically integrated geometrical quantities. So that these integrals are evaluated correctly, it is necessary to have quadratic elements. Note:

A connection

between a massive part 3D and a beam part requires six linear relations. Operands

4.18.4 of option "2D_POU " ♦OPTION

= "2D_POU" This

option makes it possible 2D to connect a surface part to a part modelled with a beam of Eulerian or discrete. ♦/MAILLE

_1= /GROUP_MA_1
= These operands

define meshes edge of the part 2D to connect to the element 1D. ♦ /NOEUD_2

= /GROUP_NO_2
= These operands

define the node of the beam to be connected to the surface part. Thus if NOEUD_2 is used , one should give one node and if GROUP_NO_2 is used , one should give one group, this one containing one node. Precaution for use

: The surface

part must be with a grid with quadratic elements because the coefficients of the relations to be imposed are numerically integrated geometrical quantities. So that these integrals are evaluated correctly, it is necessary to have quadratic elements. Operands

4.18.5 of option "COQ_POU " This

option makes it possible to connect a part with a grid in shell with a beam part. ♦AXE

_POUTRE = Makes it possible

to define the axis of the beam to be connected, whose end is lno2 or lgno 2 (1 only node). ♦CARA

_ELEM = will cara Concept

created by the command AFFE_CARA_ELEM , containing the geometrical characteristics of the shell. ♦/MAILLE

_1 = /GROUP_MA_1
= These operands

define meshes edge of the part with a grid in shells (meshes of edge are thus SEG2 or SEG3 following the selected modelization). These meshes must be assigned by of the finite elements edge of shells before. ♦ /NOEUD_2

= /GROUP_NO_2
= These operands

define the node of the beam to be connected to the shell part. Thus if NOEUD_2 is used one should give one node, and if GROUP_NO_2 is used , one should give one group, this one containing one node. Precaution for use

: The trace

of the section of the beam on the shell part must correspond exactly to meshes of edge defined by MAILLE_1 or GROUP_MA_1 . This implies the identity of the centres of inertia, of surfaces of the sections shell and beam in opposite. Operands

4.18.6 of option "3D_TUYAU " ♦OPTION

= "3D_TUYAU", This

option makes it possible 3D to connect a massive part with a part modelled with elements PIPE . ♦AXE

_POUTRE = Defines

the axis of the pipe to be connected, whose end is only one node (lno2 or lgno 2) . ♦CARA

_ELEM = will cara Idem

[§4.19.4]. ♦/MAILLE

_1 = /GROUP_MA_1
= These operands

define meshes surface massive part modelling the trace of the section of the pipe on this massive part. These meshes must be assigned by of the finite elements sides of elements 3D before. ♦/NOEUD

_ 2 = /GROUP_NO_2
= These operands

define the node of the pipe to be connected to the massive part. Note:

A connection

between a massive part 3D and a pipe part requires six linear relations for the degrees of freedom of beam, plus a relation on the mode of swelling , plus twelve relations corresponding to the transmission of the modes of Fourier two and three of ovalization of the pipe. Operands

4.18.7 of option "COQ_TUYAU " ♦ OPTION

= "COQ_TUYAU" This

option makes it possible to connect a part with a grid in shell to a part with a grid with elements pipe. ♦AXE

_POUTRE = Makes it possible

to define the axis of the pipe to be connected, whose end is lno2 or lgno 2 (only one node). ♦CARA

_ELEM = will cara, Concept

created by the command AFFE_CARA_ELEM , containing the geometrical characteristics of the shell. ♦ /MAILLE_1

= /GROUP_MA_1
= These operands

define meshes edge of the part with a grid in shells (meshes of edge are thus SEG2 or SEG3 following the selected modelization). These meshes must be assigned by of the finite elements edge of shells before. ♦ /NOEUD_2

= /GROUP_NO_2
= These operands

define the node of the pipe to be connected to the shell part. Thus if NOEUD_2 is used one should give one node, and if GROUP_NO_2 is used , one should give one group, this one containing one node. Precaution for use

: The trace

of the section of the pipe on the shell part must correspond exactly to meshes of edge defined by MAILLE_1 or GROUP_MA_1 . This implies the identity of the centres of inertia, of surfaces of the sections shell and pipe in opposite. Consequently connections of type "bypass" are impossible. Note:

A connection

between a shell part and a pipe part requires the same linear relations as option "COQ_POU " on the degrees of freedom of beam of the element pipe besides the relations on the degrees of freedom of ovalization, warping and swelling. Operands

4.18.8 of option "PLAQ_POUT_ORTH " ♦OPTION

= "PLAQ_POUT_ORTH" This
option makes it possible to connect a part with a grid with elements TRI3 and QUA 4 (modelizations DKT, DST and DKTG) with a part modelled by a beam element or discrete. ♦/MAILLE

_1= /GROUP_MA_1
= These operands
define meshes plate which model the trace of the section of the beam on this part. These meshes must be assigned by of the finite elements plate, modelizations DKT, DST and DKTG . ♦ /NOEUD_2
= /GROUP_NO_2
= These operands
define the node to be connected to the plate. Thus if NOEUD_2 is used , one should give one node and if GROUP_NO_2 is used , one should give one group, this one containing one node. The node must carry the following degrees of freedom: dx, DY, DZ, DRX, DRY MARTINI, DRZ. ◊VERIF

_EXCENT=/ "NON " , /"OUI'
, [DEFAULT
] the node
of the beam must coincide, except for a tolerance, with the center of gravity of meshes which model the trace of this beam on slab. In the event of noncompliance with this rule, 2 behaviors are possible: if VERIF_EXCENT

- = "OUI', behavior by default, a message d'error is emitted and the code stops in fatal error. if VERIF_EXCENT
- = "NON", a message of information is emitted. This operand

makes it possible not to be obliged to position exactly the beams at the center of gravity of the trace of the section, which is not inevitably known at the time of the realization of the mesh. In the case, where this rule is not complied with, the user is informed of the distance between the node of the beam and this center of gravity either by a fatal error (VERIF_EXCENT = "OUI') or by the emission of a message of information (VERIF_EXCENT = "NON"). Note:

With

an aim of simplifying the entry of the data the following checks are not carried out: There

- is no checking which the axis of the beam is perpendicular to the plate. There
- is no checking between the computation of the mechanical characteristics (S, I,…) carried out on meshes of the trace of the section of beam and the characteristic mechanics assigned to the beam with assistance CARA_ELEM .

To make these checks would be needed that the user gives besides the node of the beam, the name of the mesh affected by the CARA_ELEM which has as an end the node of connection. In the majority of the cases this mesh is unknown of the user, it is the software of mesh which defines its name. Operand

4.18.9 ANGL_MAX ◊ ANGL_MAX

=/1., [DEFAULT] /angl
, [R] Angle

(in degree) allowing to check if meshes lists lma1 or lgma 1 have higher than norms forming an angle Eng between them. If it is the case, there is emission of an alarm message. The programming
is made in 3D case: “3D_TUYAU ”, “3D_POU ”, “PLAQ_POUT_ORTH ”. Operand

4.18.10NUME_LAGR If “

- NORMAL ”, the two Lagrange multipliers associated with the relation will be such as the first will be located before all the terms implied in the relation and the second after, in the assembled matrix. If “
- APRES ”, the two Lagrange multipliers associated with the relation will be located after all the terms implied in the relation, in the assembled matrix. This choice
  has the advantage of having an assembled matrix whose obstruction is weaker but has the disadvantage to be able to reveal a singularity in the matrix. Key word

4.19 LIAISON_UNIF Drank This

4.19.1 factor key word

makes it possible to impose the same unknown value, for a degree of freedom given, on a set of nodes. These nodes

are defined by the mesh groups, meshes, the nodes groups or nodes list to which they belong. Syntax

4.19.2 for

- AFFE_CHAR_MECA and AFFE_CHAR_MECA_F LIAISON_UNIF

```
=_F (♦/MAILLE
                              =lma, [      l_maille              ]
/GROUP_MA
                              =lgma,  [  l_gr_maille         ] /NOEUD
                           =lno   , [    l_noeud                ]
/GROUP_NO
                              =lgno,  [  l_gr_noeud          ] ♦ DDL
                         = | "DX      ", | "DY
                                      ", | "DZ
                                      ", | "DRX
                                      "  , | "DRY
                                      "  , |' DRZ
                                      ", ) Operand
```

4.19.3 ♦/MAILLE

```
/GROUP_MA
/NOEUD
/GROUP_NO
These operands
```

make it possible nodes list to define one n from which N_i one eliminated the redundancies, (for MESH and GROUP_MA , it is connectivities of meshes). ♦DDL

This

operand makes it possible to define a list of degrees of freedom with u_i texts $i=1,r$ r taken among: "DX", "DY", "DZ", "DRX", "DRY", "DRZ"

the resulting $r\times(n-1)$ kinematical conditions are: for

$$u_i(N_1)=u_i(N_k)$$

Key word $k\in\{2,\dots,n\}$ $i\in\{1,\dots,r\}$

4.20 LIAISON_CHAMNO a linear

4.20.1

relation between all the degrees of freedom present in a concept CHAM_ NO This key word Drank Factor key word usable to define can be also used to impose on structure (or a part) a given work, for a loading calculated as a preliminary with another AFFE_CHAR_MECA and leading to an assembled vector produces by ASSE_ VECTEUR [U4.61 .23]. Syntax

4.20.2 (AFFE_CHAR_MECA only) LIAISON

```
      _CHAMNO=_F (♦ CHAM_NO      =chamno         , [cham_no            ] ♦
COEF_IMPO
                             =, [R] ◊NUMÉRIQUE      β
                          _LAGR  = "NORMAL       ", [DEFAULT
   ]/"APRES
                                               ",) Operands
```

4.20.3 CHAM_NO

= Name of

the cham_ No which is used to define the linear relation. The degrees of freedom connected are all those present in the chamno . The coefficients to be applied to the degrees of freedom are the values of the chamno for these degrees of freedom. Example:

Let us suppose

that L" one has a bearing chamno on two nodes of name and respectively $N01$ $N02$ carrying the degrees of freedom "DX", "DY" and " DZ" for the N01 node and "DX", "DY", "DZ", "DRX" , "DRY" and " DRZ" for the node. Also $N02$

let us suppose that the chamno has the following values for these degrees of freedom: 2. "DX

```
"N01      1.      "DY
"N01      3.      "DZ
"N01      1.      "DX
"N02      4.      "DY
"N02      2.      "DZ
"N02      3.      "DRX
"N02      5.      "DRY
"N02      2.      "DRZ
"N02            the linear
```

relation that L" one will impose is: 2.*DX

```
  (N01) +1.*DY (N01) +3.*DZ (N01) + 1.*
DX (N02) +4.*DY (N02) +2.*DZ (N02) + 3.*
DRX (N02) +5.*DRY (N02) +2.*DRZ (N02) = COEF_IMPO  β
```

= C" is

the value of the real coefficient to the second β member of the linear relation. NUME_LAGR

= if"

- NORMAL ", the two Lagrange multipliers associated with the relation will be such as the first will be located before all the terms implied in the relation and the second after, in the assembled matrix, if"
- APRES ", the two Lagrange multipliers associated with the relation will be located after all the terms implied in the relation, in the assembled matrix. This choice

presents L" favours D" to have an assembled matrix whose L" obstruction is weaker but has the disadvantage to be able to reveal a singularity in the matrix. Key word

4.21 LIAISON _RBE3 Drank Factor key word

4.21.1

usable to define linear relations of type RBE3 between the degrees of freedom of a master node and nodes slaves. They are relations making it possible to specify the value of certain degrees of freedom of a master node as being the weighted average of certain displacements and certain rotations of nodes slaves. The produced
linear relations are such as the forces seen by the master node are distributed to the nodes slaves proportionally at their distance to the center of gravity of the nodes slaves. The possible additional weightings provided by the user can be taken into account. For more precise details, one will be able to refer to Doc. of reference [R3.03.08]. Syntax

4.21.2 (AFFE_CHAR_MECA only) LIAISON

```
        _RBE3=_F (♦/    GROUP_NO_MAIT=gno, [gr_noeud            ]/NOEUD
                              _MAIT=no, [node                   ] ♦ DDL
                           _MAIT=ddl_mait, [l_Kn]                 ♦
                           /GROUP    _NO_ESCL=lgno, [l_gr_noeud
    ]/NOEUD
                              _ESCL=lno, [l_noeud                   ] ♦ DDL
                           _ESCL= d.o.f., [l_Kn]                        ◊
COEF
                           _ESCL=formule βi                       ◊
                           NUMÉRIQUE_ LAGR=/"NORMAL      ", [DEFAULT
       ]/"APRES
                                             ",) Operands
```

4.21.3 ♦/

GROUP_NO_MAIT=gno,/NOEUD
_MAIT=no, Identification
of the master node of the linear relation. ♦ DDL

_MAIT=ddl_mait, Identification
of the degrees of freedom of the master node implied in the linear relation. One expects a list including at more the 6 entries among "DX", "DY", "DZ", "DRX" , "DRY" , "DRZ" . ♦

/GROUP _NO_ESCL=lgno,/NOEUD
_ESCL=lno, Identification
of the nodes slaves of the linear relation. ♦ DDL

_ESCL= d.o.f., Identification
of the degrees of freedom of the nodes slaves implied in the linear relation. The list must have a length equal to the number of nodes slaves Each term of the list must be a combination of the entries "DX", "DY", "DZ", "DRX" , "DRY" , "DRZ" , separated by an indent" -". ◊ COEF

_ESCL=formule β_i
of weight coefficients of the terms of the linear relation for each slave node. The list must: either

- to have the same length which the number of nodes slaves or
- to be length 1, in which case this coefficient is used for all the nodes slaves ◊

NUME_LAGR = if"

- NORMAL ", the two Lagrange multipliers associated with the relation will be such as the first will be located before all the terms implied in the relation and the second after, in the assembled matrix. if"

- APRES ″, the two Lagrange multipliers associated with the relation will be located after all the terms implied in the relation, in the assembled matrix. Example

If one
wants to create a relation of the type RBE3 between: degrees of freedom
- “DX”, “DY”, “DZ”, “DRX” of master node “NO1” ; and:

degrees of freedom
- “DX”, “DY”, “DZ” of slave node “NO2” with the weight coefficient; 0.1 degrees of freedom
- “DX”, “DY”, “DZ”, “DRX” of slave node “NO3” with the weight coefficient; 0.2 degrees of freedom
- “DX” , “DY ” , “DZ ” , “DRX ” of slave node “NO4” with the weight coefficient; one 0.3 must

write the command: LIAISON

```
_RBE3=_F (  GROUP_NO_MAIT=' NO1″, DDL_
                 MAIT= (“DX″, “DY″, “DZ″, “DRX″), GROUP
                 _NO_ESCL= (“NO2″, “NO3″, “NO4″), DDL_ESCL
                 = (“DX-DY-DZ″, “DX-DY-DZ-DRX″, “DX-DY
                           - DZ-DRX″), COEF_
                 ESCL= (0.1, 0.2,0.3),) Key word
```

4.22 CHAMNO_IMPO Drank It

4.22.1

acts by way of a light adaptation of key word LIAISON_CHAMNO of operator AFFE_CHAR_MECA . This one makes it possible to apply like coefficients of linear relation the contents of a cham_ No In the case of
key word CHAMNO_IMPO , one takes the contents of a cham_ No like second member of the linear relation. It is thus strictly equivalent to a manual procedure where one recovers the values of the cham_ No to the hand then one imposes them via DDL_IMPO . Syntax

4.22.2 (AFFE_CHAR_MECA only) CHAMNO

```
    _IMPO=_F (♦  CHAM_NO      =chamno          , [        cham_no_sdaster] ♦
COEF_MULT
                               =, [R]       β  ◊          NUMERICAL
                            _ LAGR = "NORMAL      ", [DEFAULT     ]/"APRES
                                                 ",) Operands
```

4.22.3 CHAM_NO

= Name of

the cham_ No which is used to define the specified values. COEF_MULT

= multiplying

Coefficient of the cham_ No NUME_LAGR

= if "

- NORMAL ", the two Lagrange multipliers associated with the relation will be such as the first will be located before all the terms implied in the relation and the second after, in the assembled matrix, if "
- APRES ", the two Lagrange multipliers associated with the relation will be located after all the terms implied in the relation, in the assembled matrix. This choice

has the advantage of having an assembled matrix whose obstruction is weaker but has the disadvantage to be able to reveal a singularity in the matrix. Key word

4.23 LIAISON_INTERF Factor key word Drank

4.23.1

répétable and usable with a model containing at the same time of the finite elements and of the static macro-elements condensing certain subdomains. It makes it possible and to define linear relations between the physical degrees of freedom of the interfaces of the part of model in finite elements generalized coordinates of modes of reduced representation of the motions of interface contained in certain macro-elements of static condensation. Syntax

4.23.2 (AFFE_CHAR_MECA only) LIAISON

```
        _INTERF=_F (♦MACR   _ELEM_DYNA   =macrel  , [macr_elem_dyna       ]
◊TYPE
                            _LIAISON  = "RIGIDE       ", [DEFAULT
   ]/"SOUPLE
                                                      ",) Operands
```

4.23.3 MACR_ELEM_DYNA

= Name of

the macr_ elem_dyna which is used to define the linear relations between the physical degrees of freedom of the interface between the field non-condensed modelled in finite elements and a field condensed by the macro-element and the component of the node compared to generalized coordinates of modes of motions of interface. That is necessary only when the modes of motions of interface are a reduced base of all the constrained modes corresponding each one to a mode of displacement for each physical degree of freedom of the interface. One thus generates relations of the type LIAISON_DDL whose coefficients are calculated in a transparent way for the user between the nodes of the dynamic interface of the macro-element and those associated with the base of reduction which was used to constitute the macro-element. TYPE_LIAISON

= if "

- RIGIDE ", one and the writes the relation between the physical degrees of freedom of the interface U_{Σ} component of the node compared to generalized coordinates of modes q of motions of interface in Φ the shape of simple product: . This $U_{\Sigma}=\Phi_q$ choice makes it possible to have a connection more rigid than in taking into account all the constrained modes corresponding each one to a mode of displacement for each physical degree of freedom of the interface. if "
- SOUPLE ", one and the writes the relation between the physical degrees of freedom of the interface U_{Σ} component of the node compared to generalized coordinates of modes q of motions of interface in Φ the shape of double product: . This $\Phi^T U_{\Sigma}=\Phi^T \Phi q$ choice makes it possible to have a connection more flexible than in taking into account all the constrained modes corresponding each one to a mode of displacement for each physical degree of freedom of the interface. Key word

4.24 VECT_ASSE Drank Key word

4.24.1

making it possible to assign a second member in the form of a CHAM_NO in commands STAT_NON_LINE and DYNA_NON_LINE . This CHAM_NO is transmitted to these commands via the name of the loading. Syntax

4.24.2 VECT_ASSE

```
=chamno    [cham_no_DEPL_R          ] Operand
```

4.24.3 VECT_ASSE chamno

is the name of the CHAM_NO which will serve as second member in commands STAT_NON_LINE or DYNA_NON_LINE . The mode

of use can see itself in the following way: tank

```
= AFFE_CHAR_MECA (MODELS
          =modele          , VECT_
          ASSE=chamno      ,);
                       resu
= STAT_NON_LINE (MODELS
          =modele          , EXCIT
          =_F (CHARGE          = tank),…    )
          ; Key word
```

4.25 FORCE _SOL Drank Key word

4.25.1

making it possible by means of to take into account the internal force of a field of soil the temporal evolutions of the contributions in stiffness, mass and damping of the impedance of soil. The impedance of soil extracted at initial time makes it possible to constitute by MACR_ELEM_DYNA a macro-element representing the behavior of the field of soil which one adds to the structure model. The dynamic interface of the macro-element is described either by a super-mesh of the model containing at the same time structure and this macro-element, or by a nodes group if the physical interface coincides with the modal dynamic interface. One can
also take into account, if it exists, the temporal evolution of the seismic forces, assigned to this same dynamic interface in the form of logical unit. This kind of
load is taken into account in command DYNA_NON_LINE . Syntax

4.25.2 (AFFE_CHAR_MECA only) FORCE

```
        _SOL=_F (♦|UNITE       _RESU_RIGI    =UNIRESRI   , [I] |UNITE_RESU

                              _AMOR =UNIRESAM    , [I] |UNITE
                           _RESU_MASS    =uniresma   , [I] ◊
UNITE_RESU_FORC
                            =uniresfo   , [I] ♦/
                         SUPER_MAILLE    =sup_ my , [super_maille
   ]/GROUP_NO_INTERF
                              =gnintf   , [group_no     ]) Operands
```

4.25.3 UNITE _RESU_RIGI/UNITE_RESU_AMOR/UNITE_RESU_MASS These operands

make it possible to introduce the temporal evolutions of the contributions in stiffness, mass and damping of the impedance of soil in the form of logical units. It is necessary at least that one of these operands is present. Operands

4.25.4 UNITE_RESU_FORC This operand

makes it possible to introduce, if it exists and in the form of logical unit, the temporal evolution of the seismic forces, assigned to the dynamic interface of the macro-element representing the behavior of the field of soil which one adds to the structure model. Operands

4.25.5 SUPER _MAILLE/GROUP_NO_INTERF These operands

make it possible to describe the dynamic interface of the macro-element representing the behavior of the field of soil which one adds to the model of structure either by a super-mesh of the model containing at the same time structure and this macro-element by key word SUPER_MAILLE , or by a nodes group by key word GROUP_NO_INTERF if the physical interface coincides with the modal dynamic interface. Example

4.25.6 of use an example

of use is provided in the case test MISS03 B [V1.10 .122]. Key word

4.26 FORCE_FACE Drank Factor key word

4.26.1

usable to apply surface **forces to** a face **(of voluminal** element) defined by one or more meshes or of the mesh groups of type triangle **or quadrangle . According to**

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.26.2 for

- AFFE_CHAR_MECA FORCE

```
_FACE=_F (♦|MAILLE=lma          , [    l_maille            ] |GROUP
                     _MA=lgma   ,    [ l_gr_maille      ] ♦|FX
                   =FX,    [R]              |        FY=
                     fy   , [R]             |        FZ=
                     fz   , [R]            )          for
```

- AFFE_CHAR_MECA_F FORCE

```
_FACE=_F (♦|MAILLE=lma          , [    l_maille            ] |GROUP
                     _MA=lgma   ,    [ l_gr_maille      ] ♦|FX
                   =fxf      , [function                   ] |FY=fyf
                          , [function                  ] |FZ=fzf
                          , [function                  ]) Operands
```

4.26.3 fx, fy

, fz fxf, fyf, fzf values	of the components in reference GLOBAL of the surface forces applied to the face. Modelizations

4.26.4 and meshes This loading

applies to the types of meshes and the following modelizations: Net

Modelization	TRIA3
, TRIA6, QUAD4 , QUAD8, QUAD9, QUAD8 , TRIA6 3D, 3D_SI	, 3D_INCO 3D_HHMD , 3D_HMD, 3D_THHD, 3D_THHMD, 3D_THMD Note

: The rule

- of remanence (see U1.03.00) applies between the various quantities which one can affect: FX, FY, … . Key word

4.27 FORCE_ARETE Drank Factor key word

4.27.1

usable to apply linear forces **, with** an edge **of voluminal** element **or** shell. This edge is defined by one or more meshes or of the mesh groups of type segment **. According to**

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.27.2 for

- AFFE_CHAR_MECA FORCE_ARETE

```
=_F (♦|MAILLE=lma              , [    l_maille              ] |GROUP
                  _MA=lgma   ,    [  l_gr_maille          ] ♦|FX
              =FX,    [R]                |             FY=
                  fy   , [R]             |             FZ=
                  fz   , [R]             |             MX=
                  MX       , [R]             |             MY=
                  my  , [R]              |             MZ=
                  mz  , [R]             )               for
```

- AFFE_CHAR_MECA_F FORCE_ARETE

```
=_F  (♦|MAILLE=lma              , [    l_maille              ] |GROUP
                  _MA=lgma   ,    [  l_gr_maille          ] ♦|FX
              =fxf       , [function                        ] |
FY=fyf
                     , [function                        ] |
FZ=fzf
                     , [function                        ] |
MX=mxf
                     , [function                        ] |
MY=
                  myf      , [function                        ] |
MZ=mzf
                     , [function                        ])
Operands
```

4.27.3 fx, fy

, fz, MX, my, mz fxf, fyf, fzf, mxf, myf, mzf: values — of the components in reference GLOBAL of the linear forces applied to the edge. Modelizations

4.27.4 and meshes This loading

applies to the types of meshes and the following modelizations: Net

Modelization	SEG2
SEG2, SEG3 DKT,	DST, Q4G 3D, 3D_SI , 3D_INCO COQUE_3D Key word

4.28 FORCE_CONTOUR Drank Factor key word

4.28.1

usable to apply linear forces **, at the edge of a field (2D,** AXIS or AXIS_FOURIER) defined by one or more meshes or of the mesh groups. According to

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.28.2 for

- AFFE_CHAR_MECA FORCE

```
_CONTOUR=_F (◊|NET =lma, [l_maille ] |
GROUP
_MA=lgma , [ l_gr_maille ] ♦|FX
=FX, [R] | FY=
fy , [R] | FZ=
fz , [R] | MX=
MX , [R] | MY=
my , [R] | MZ=
mz , [R] ) for
```

- AFFE_CHAR_MECA_F FORCE

```
_CONTOUR=_F (◊|NET =lma, [l_maille ] |
GROUP
_MA=lgma , [ l_gr_maille ] ♦|FX
=fxf , [function ] |
FY=fyf
, [function ] |
FZ=fzf
, [function ] |
MX=mxf
, [function ] |
MY=
myf , [function ] |
MZ=mzf
, [function ])
Operands
```

4.28.3 fx, fy

, fz, MX, my, mz fxf, fyf, fzf, mxf, myf, mzf values of the components in reference GLOBAL of the linear forces applied to contour. Modelizations

4.28.4 and meshes This loading

applies to the types of meshes and the following modelizations: Net

Component	Modelization	SEG2,
SEG3 C_PLAN	D_PLAN AXIS Fx, Fy	Fx, Fy Fx, Fy SEG2,

SEG3 AXIS_FOURIER	Fx (R)	, Fy (Z), Fz () Note θ

: Out of plane

, the forces are with being provided per unit of length of the mesh, into axisymmetric, the forces required are brought back to a sector of 1 radian (to divide the real loading by). Key word 2π

4.29 FORCE_INTERNE Drank Factor key word

4.29.1

usable to apply volume forces **(2D or** 3D), with a field **defined** by one or more meshes or of the mesh groups of the voluminal type **. According to**

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.29.2 for

- AFFE_CHAR_MECA FORCE

```
_INTERN=_F (♦/TOUT = ' OUI' , /|NET
=lma, [l_maille ] |
GROUP_MA
=lgma, [ l_gr_maille ] ♦|FX
=FX, [R] | FY=
fy , [R] | FZ=
fz , [R] ) for
```

- AFFE_CHAR_MECA_F FORCE

```
_INTERN=_F (♦/TOUT = ' OUI' , /|NET
=lma, [l_maille ] |
GROUP_MA
=lgma, [ l_gr_maille ] ♦|FX
=fxf , [function
] |FY=fyf
, [function ] |
FZ=fzf
, [function ])
Operands
```

4.29.3 fx, fy

, fz, fxf, fyf, fzf: values	of the components in reference GLOBAL of the volume forces applied to the field. Modelizations

4.29.4 and meshes This loading

applies to the types of meshes and the following modelizations: Net

Modelization	HEXA8
, HEXA20, HEXA27 PENTA 6, PENTA15 TETRA 4, TETRA10 PYRAM 5, PYRAM13 3D, 3D_SI	, 3D_INCO 3D_HHMD , 3D_HMD, 3D_THHD, 3D_THHMD, 3D_THMD, 3D_THHM, 3D_THM, 3D_HM, 3D_THH, 3D_HHM TRIA3
, TRIA6, QUAD4 , QUAD8, QUAD9 C_PLAN	D_PLAN AXIS AXIS_FOURIER AXIS_SI AXIS_INCO AXIS_ THHM, AXIS_HM, AXIS_THH, AXIS_HHM, AXIS_THM D_PLAN _THHM, D_PLAN_HM, D_PLAN_THH, D_PLAN_HHM, D_PLAN_THM Note:

In 2D

- *(resp 3D), the forces are with being provided per unit of area (resp volume), into axisymmetric, the forces required are brought back to a sector of 1 radian (to divide the real loading by). 2π The rule*
- of remanence (see U1.03.00) applies between the various quantities which one can affect: FX, FY,... . Key word

4.30 PRES_REP Drank Factor key word

4.30.1

usable to apply a pressure **to** a field of continuum 2D or **3D and/or** shears **with** a field of continuum 2D. According to

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.30.2 for

- AFFE_CHAR_MECA |NEAR

```
    _REP=_F    (♦/TOUT              = ' OUI'      ,  /|NET
                                   =lma,    [l_maille              ] |
GROUP_MA
                                   =lgma, [  l_gr_maille      ] | FISSURE
                                   =fiss   , [   fiss_xfem     ] ♦|NEAR
                               =P, [R      ]  |   CISA_2D
                               =T, [R  ] )              for
```

- AFFE_CHAR_MECA_F |NEAR

```
    _REP=_F    (♦/TOUT              = ' OUI'      ,  /|NET
                                   =lma,    [l_maille              ] |
GROUP_MA
                                   =lgma, [  l_gr_maille      ] | FISSURE
                                   =fiss   , [   fiss_xfem     ] ♦|NEAR
                               =Pf, [      function              ] |
CISA_2D
                               =Tf, [  function              ]) Operands
```

4.30.3 |NEAR

= P (PF) Value

of the pressure imposed P (or

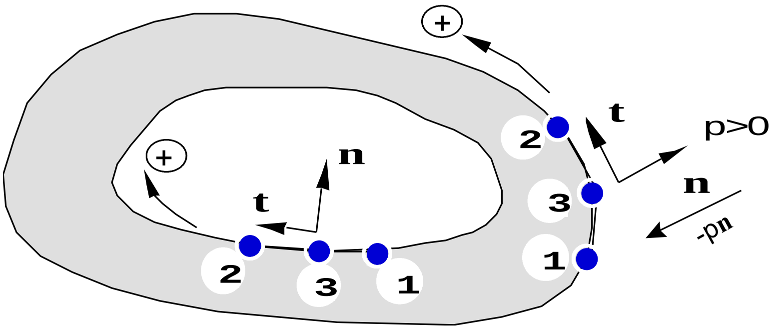
PF) is positive according to the contrary meaning of the norm to the element: that is to say the tensor σ of the stresses, the imposed loading is: . |CISA_2D $\sigma_{ij} n_i n_j = -p n_i n_j$

= T (Tf) Value

of the shears imposed T (or

Tf) is positive according to the tangent with the element. For

the definition of the norms and tangents, one will refer to the definitions given to [§4.1]. Example: | FISSURE



`=fiss , [ fiss_xfem ] the imposition`

of a pressure on the lips of a crack X-FEM `is made` specific CRACK by the key word `, since` no group of mesh corresponds to the lips. One then informs the names of the cracks (coming of the command DEFI_FISS_XFEM `[U4.82 .08]) about which` one wishes to apply the pressure. Modelizations

4.30.4 and meshes the loading

of pressure applies to the types of meshes and the following modelizations: Type

of Mesh Modelization	SEG2
SEG3 AXIS,	D_PLAN, C_PLAN, AXIS_FOURIER D_PLAN_HHM , D_PLAN_HM, D_PLAN_THHM, D_PLAN_THM SEG3
AXIS_HHM	, AXIS_HM, AXIS_THHM, AXIS_THM TRIA6
QUAD8 3D_HHM	, 3D_HM, 3D_THHM, 3D_THM TRIA3
, QUAD4 TRIA6 , QUAD8, QUAD9 3D	the loading

of shears applies to meshes and the following modelizations: Type

of Mesh Modelization	SEG2
SEG3 AXIS,	D_PLAN, C_PLAN, AXIS_FOURIER Key word

4.31 EFFE_FOND Factor key word

usable to compute: the basic effect on a branch of pipework (modelization 3D exclusively) subjected to an internal pressure. Syntax P

4.31.1 for

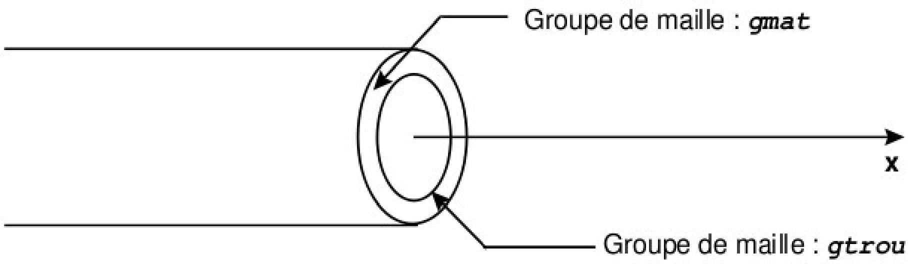
- AFFE_CHAR_MECA |EFFE_FOND

```
    =_F (♦|NET              =lma,  [l_maille                  ] |
GROUP
                         _MA=lgma   , [    l_gr_maille         ] ♦GROUP
                       _MA_INT=gtrou    ,    [l_gr_maille       ] ♦PRES
                       =p  , [R]            )             for
```

- AFFE_CHAR_MECA_F |EFFE_FOND

```
    =_F (♦|NET              =lma,  [l_maille                  ] |
GROUP
                         _MA=lgma   , [    l_gr_maille         ] ♦GROUP
                       _MA_INT=gtrou    ,    [l_gr_maille       ] ♦PRES
                       =pf , [function                         ])
Operands
```

4.31.2 ♦ |



GROUP _MA=gmat , |NET
=lma, Together

the meshes surface ones modelling the material section of pipework (gmat on the figure) where the pressure will be applied. ♦GROUP

_MA_INT=gtrou , Together

of meshes linear (SEG2 or SEG3) modelling the contour of the hole (option on the figure). The knowledge

of these meshes is necessary because one needs to calculate the area of hole. Indeed

, the force resulting (or basic effect) due to stopping from hole at the end is worth: This basic

$$F_b=\pi R_i^2 P\cdot x$$

force or effect applies to the wall of the tube (gmat). The force divided corresponding is worth: ♦PRES

$$F_p=\frac{\pi R_i^2}{\pi\left(R_e^2-R_i^i\right)}P\cdot x=P\frac{S_{trou}}{S_{mat}}\cdot x$$

: p (or PF) Pressure

interns with the pipework. One applies in fact to (F_p with $gmat$ following $p>0$ the contrary meaning of the norm to the element). Key word

4.32 PRE_EPSI Drank Factor key word

4.32.1

usable to apply a predeformation formulates ε_{pre} a loading of strain average, overall uniform applied to an element 2D, 3D or of structure. The assignment can be done on one or more meshes, one or more mesh groups or on all the elements of the model. The second

calculated elementary member will be formula $\int_{Ve} A\,\varepsilon_{pre}:\varepsilon(v*)dV_e$ formula A the elasticity tensor (recovered in the field material for all the models for which are defined the elastic characteristics). One

should not confuse this predeformation with the initial strain formulates ε_{ini} in nonlinear, because this predeformation does not intervene directly in the statement of the constitutive law. This

predeformation is usable for example to solve the elementary problems determining the elastic correctors in the basic cell (2D, 3D), in periodic homogenization. The moduli of homogenized elasticity are obtained by calculating by the operator POST_ELEM [U4.81 .22] key word ENER_POT the potential energy of elastic strain to the equilibrium starting from the correctors. But that can be useful for other applications. Syntax

4.32.2 for

- AFFE_CHAR_MECA PRE_EPSI

```
=_F (♦ /TOUT            = ' OUI'     , /|NET
                          =lma,   [l_maille                ] |
GROUP_MA
                          =lgma, [  l_gr_maille         ] ♦ |
                 EPXX    =EPSXX          ,    [R] |           EPYY
                      =  epsyy     ,    [R] |          EPZZ
                      =  epszz     ,    [R] |          EPXY
                      =  epsxy     ,    [R] |          EPXZ
                      =  epsxz     ,    [R] |          EPYZ
                      =  epsyz     ,    [R] |          EPX
                      =EPSX     ,          [ R] |            KY=
                      ky      , [R          ]    |            KZ=
                      kz      , [R          ]    |            EXX
                      =EXX  , [       R   ] |            EYY
                      =EYY  , [       R   ] |            EXY
                      =EXY  , [       R   ] |            KXX
                      =KXX  , [       R   ] |            KYY
                      =KYY  , [       R   ] |            KXY
                      =kxy  , [       R   ])              for
```

- AFFE_CHAR_MECA_F PRE_EPSI

```
=_F (♦ /TOUT            = ' OUI'     , /|NET
                          =lma,   [l_maille                ] |
GROUP_MA
                          =lgma, [  l_gr_maille         ] ♦|EPXX
                 =epsxxf                 , [function             ] |
EPYY
                     =epsyyf           , [function            ] |EPZZ
                     =epszzf           , [function            ] |EPXY
                     =epsxyf           , [function            ] |EPXZ
```

=epsxzf , [function] |EPYZ
=epsyzf , [function])

Operands

4.32.3 |EPXX

= epsxxouepsxxf |EPYY
= epsyyouepsyyfcomposantes of the tensor of the strains |EPZZ
= epszzouepszzfinitiales in reference GLOBAL |EPXY
= epsxyouepsxyf |EPXZ
= epsxzouepsxzf (in 3D only) | EPXZ
= epsxz or epsxzf Note:

For

the elements beams only: constant strain field generalized by element: |EPX

= epsx: strain according to the axis of beam

|*KY* =

ky : variation

of curvature according to the local axis y |*KZ* = $-\frac{d\,\theta_y}{dx}$

kz : variation

of curvature according to the local axis z *For* $\frac{d\,\theta_z}{dx}$

curved beams, only EPX is taken *into account currently. Emission of a message d'error fatal if the user provides KY or* KZ . For

the elements shells only: constant strain field initial by element: |EXX,

EYY , EXY: strains of membrane |KXX,
KYY , KXY: variations of curvatures Modelizations

4.32.4 and meshes This loading

applies to the types of meshes and the following modelizations: Type

of Mesh Modelization	TRIA3
, TRIA6 QUAD4 , QUAD8, QUAD9 C_PLAN	, AXIS, D_PLAN HEXA8
, HEXA20, HEXA27 PENTA 6, PENTA15 PYRAM 5, PYRAM13 TETRA 4, TETRA10 3D SEG2	
POU_D_E	, POU_D_T, POU_D_TG, POU_C_T TRIA3
, QUAD4 DKT,	DST, Q4G HEXA20
3D_SI	QUAD8
AXIS_SI	, D_PLAN_SI Key word

4.33 FORCE_POUTRE Drank Factor key word

4.33.1

usable to apply linear forces **, to** elements of type beam (POU_D_T*, POU_D_E,…) defined on all the mesh or one or more meshes or of the mesh groups. The forces are definite component by component, either in reference GLOBAL , or in the local coordinate system of the element defined by the operator AFFE_CARA_ELEM [U4.42 .01]. According to

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.33.2 for

- AFFE_CHAR_MECA FORCE_POUTRE

```
=_F (♦/TOUT = ' OUI' , /|NET
=lma, [l_maille ] |
GROUP_MA
=lgma, [ l_gr_maille ] ◊TYPE
_CHARGE=/ "FORCE ", [DEFAULT
]/"VENT
" , # if
TYPE_CHARGE = "FORCE" ♦/|
FX=fx , [R ] | FY=
fy , [R ] | FZ=
fz , [R ] | MX=
MX , [R ] | MY
=MY , [R ] | MZ
=MZ , [R ] /| N=
N, [ R] | VY=
vy , [R ] | VZ=
vz , [R ] | MT=
MT , [R ] | MFY
= mfy , [ R] | MFZ
= mfz , [ R] # if
TYPE_CHARGE = "VENT" ♦/|
FX=fx , [R ] | FY=
fy , [R ] | FZ=
fz , [R ] / | N=
N, [ R] | VY=
vy , [R ] | VZ=
vz , [R ] ) for
```

- AFFE_CHAR_MECA_F FORCE_POUTRE

```
=_F (♦/TOUT = ' OUI' , /|NET
=lma, [l_maille ] |
GROUP_MA
=lgma, [ l_gr_maille ] ◊TYPE
_CHARGE=/ "FORCE ", [DEFAULT
]/"VENT
" , # if
TYPE_CHARGE = "FORCE" ♦/|
FX=fxf , [ function ] |
FY=
fyf , [ function ] | FZ=
fzf , [ function ] | MX=
mxf , [ function ] | MY
=myf , [ function ] | MZ
=mzf , [ function ]/| N=
```

nf , [function] | VY=
vyf , [function] | VZ=
vzf , [function] | MT=
mtf , [function] |
MFY
= mfyf , [function] | MFZ
= mfzf , [function] # if

TYPE_CHARGE = "VENT" ♦/|
FX=fx , [function
] | FY=
fy , [function] |
FZ=
fz , [function]/|
N=
N, [function] |
VY=
vy , [function] |
VZ=
vz , [function])
Operands

4.33.3 ♦/|

FX: Force according to X [R] or [function
] |FY:
Force according to Y [R] or [function
] |FZ:
Force according to Z [R] or [function
] |MX:
Following moment X [R] or [function
] |MY:
Following moment Y [R] or [function
] |MZ:
Following moment Z [R] or [function
]/|N:
Force of traction and compression [R] or [function
] |VY:
Following shears Y [R] or [function] |VZ:
Following shears Z [R] or [function] |MT:
Twisting moment [R] or [function] |
MFY:
Following bending moment Y [R] or [function] |
MFZ:
Following bending moment Z [R] or [function]
Let us note

that one must remain homogeneous factor key word in each occurrence of the FORCE_POUTRE : either all the components are defined in reference GLOBAL or all the components are defined in the reference of definition of the beam. ◊TYPE

_CHARGE=' VENT' If p

is the pressure exerted by the wind on a normal plane surface with its direction, the unit vector

$\boldsymbol{v} = (v_x, v_y, v_z)$ having the direction and the meaning velocity of the wind, the diameter

$\varnothing$ of the cable on which is exerted the wind, then

: FX =

FY = $p \varnothing v_x$

FZ = $p \varnothing v_y$

TYPE _ $p \varnothing v_z$

CHARGE=' FORCE' [DEFAULT] Case D

“an unspecified linear force. Modelizations

4.33.4 and meshes This loading

applies to the types of meshes and the following modelizations: Net

Modelization	SEG2
POU_D_T	, POU_C_T, POU_D_E , POU_D_TGM, POU_D_TG the moments

distributed can be applied only to the straight beams to constant section. Note:

the rule of remanence (see U1.03.00) applies between the various quantities which one can affect: FX, FY,… . Key word

4.34 DDL_POUTRE Drank Factor key word

4.34.1

usable to block DDL in a local coordinate system D" a beam. The local coordinate system of a beam is defined: by

- the axis determined X by the mesh to which the node belongs. The mesh is directed towards the specified node. To avoid the indetermination, it is necessary that the node to which the condition relates belongs to only one SEG. In the case or it belongs to several meshes, the user defines the mesh giving the local directional sense. by
- VECT_Y : a vector whose projection on the orthogonal level with the axis defines X the axis. The axis Y is given Z using and by X Y
- ANGL_VRIL : angle of gimlet , given in degrees, makes it possible to direct a local coordinate system around the axis. Syntax X

4.34.2 for

- AFFE_CHAR_MECA DDL_POUTRE

```
=_F (♦|NOEUD          =lno,    [l_noeud              ] | GROUP_NO
                            =lgno, [  l_gr_noeud        ] ♦|DX
                   =UX    , [  R]          |             DY
                        =   uy, [  R]          |             DZ
                        =   uz, [  R]          |             DRX
                        =   , [R] |          θx            DRY
                        =   , [R] |          θy            DRZ
                        =   , [R] #          θz             definition
of the local coordinate system ◊|NET
                            =lma,   [l_maille                 ] |
GROUP_MA
                            =lgma, [  l_gr_maille        ] ♦/ANGL
                   _VRIL      = G, [R]     /VECT_Y
                            = (V1,       V2, V3) [l_R]    ) Operands
```

4.34.3 DX =

ux DY = uy Valeur uz DRX =	of the component of displacement in translation **imposed** on the specified nodes DZ =
DRY = θ_x Value θ_y ANGL_VRIL θ_z	of the component of displacement in rotation **imposed** on the specified nodes DRZ =

= G angle of gimlet

, given in degrees, makes it possible to direct a local coordinate system around the axis. VECT_Y X

= (V1, V2, V3) vector

whose projection on the orthogonal level with the axis defines X the axis. The axis Y is given Z using and Modelizations X Y

4.34.4 and meshes This loading

S" applies to the types of meshes and the following modelizations: Net

Modelization	SEG2
POU_D_T	, POU_C_T, POU_D_TG , POU_D_E, POU_D_TGM Key word

4.35 FORCE_TUYAU Drank Factor key word

4.35.1

usable to apply a pressure to elements pipe, defined by one or more meshes or of the mesh groups. Syntax

4.35.2 AFFE_CHAR_MECA

```
•   : |FORCE

        _TUYAU=_F  (♦/TOUT          = ' OUI'     ,  /|NET
                                      =lma,   [l_maille               ] |
GROUP_MA
                                      =lgma, [  l_gr_maille         ] ♦PRES
                              =p  , [R]            )
AFFE_CHAR_MECA_F


•   : |FORCE

        _TUYAU=_F  (♦/TOUT          = ' OUI'     ,  /|NET
                                      =lma,   [l_maille               ] |
GROUP_MA
                                      =lgma, [  l_gr_maille         ] ♦PRES
                              =pf , [function                         ])
Operand
```

4.35.3 NEAR

`=p (PF)       , Value`

of the imposed pressure (real or function). is positive
p when the pressure is internal with the pipework. Modelizations

4.35.4 and meshes This loading

applies to the types of meshes and the following modelizations: Net

Modelization	SEG3,
SEG4 SEG3 "TUYAU_3M	" "TUYAU_6M " Key word

4.36 FORCE_COQUE Drank Factor key word

4.36.1

usable to apply surface forces, to elements of type shell (DKT, DST, Q4G,…) defined on all the mesh or one or more meshes or of the mesh groups. According to

the name of the operator called, the values are provided directly (AFFE_CHAR_MECA) or via a concept function (AFFE_CHAR_MECA_F). Syntax

4.36.2 for

- AFFE_CHAR_MECA FORCE_COQUE

```
=_F (♦/TOUT            = ' OUI'     ,  /|NET
                            =lma,    [l_maille             ] |
GROUP_MA
                            =lgma, [  l_gr_maille         ] ♦/|
                  FX=fx      ,    [R ]       |             FY=
                       fy  , [R   ]       |             FZ=
                       fz  , [R   ]       |             MX=
                       MX     , [R   ]       |             MY=
                       my  , [R   ]       |             PLANE
                       MZ=    mz, [  R      ]              ◊
                     = /"MOY  "   ,  /"INF
                                 ",  /"SUP
                                 ",  /"MAIL
                                 "   , [DEFAULT           ] /PRES
                    =p  , [R]          /              | F
                     1=     f1  , [R   ]       |             F2=
                        f2  , [R   ]       |             F3=
                        f3  , [R   ]       |             MF1
                        =MF   1, [   R      ] |             MF2
                        =mf   2, [   R      ])             for
```

- AFFE_CHAR_MECA_F FORCE_COQUE

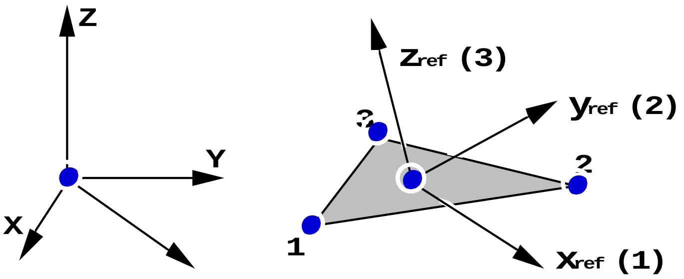
```
=_F (♦/TOUT            = ' OUI'     ,  /|NET
                            =lma,    [l_maille             ] |
GROUP_MA
                            =lgma, [  l_gr_maille         ] ♦/|
                  FX=fxf        , [ function                   ] |
FY=fyf
                            , [ function                   ] |
FZ=fzf
                            , [ function                   ] |
MX=mxf
                            , [ function                   ] |
MY=
                       myf      , [ function                   ] |
MZ=mzf
                            , [ function                   ]
◊PLAN
                    =    "MOY  "   ,  /"INF
                                 ",  /"SUP
                                 ",  /"MAIL
                                 "   , [DEFAULT           ] /PRES
                    =pf , [function                        ]/|F1=
                    f1f    , [ function                   ] |
F2=f
```

```
                2f  , [function          ] |
F3=f
                3f  , [function          ] |
MF1=
                mf1f   ,   [   function   ] |
MF2=
                mf2f   ,   [   function   ])
Operands
```

4.36.3 the operands

of FORCE_COQUE can be defined: in

- reference GLOBAL of axes , and X , Y in Z
- a reference of reference defined on each mesh or groups of mesh (reference defined on the variety); this reference is built around the norm with the shell element and of (z_{ref}) a direction (for (x_{ref}) the group of mesh) definite ANGL_REP fixes by the key word at the same time as the thickness of the shell (see factor key word COQUE operator AFFE_CARA_ELEM [U4.42 .01]). ♦/|



ANGL_REP: (45. 0.)

FX: Force according to [R] or X [function] |FY: Force according to [R] or Y [function] | FZ: Force according to [R] or Z [function] | MX: Moment of axis [R] or X [function] |MY: Moment of axis [R] or Y [function] |MZ: Moment of axis [R] or Z [function] /PRES : Normal pressure with the shell [R] or [function]/|F1: Force of membrane according to [R] or x_{ref} [function] | F2: Force of membrane according to [R] or y_{ref} [function] |F3: Following normal force [R] or z_{ref} [function] |MF1: Bending moment of axis [R] or X [function] | MF2: Bending moment of axis [R] or Y [function] Let us note

that one must remain homogeneous factor key word in each occurrence of the FORCE_COQUE : either very out of component of force in reference GLOBAL or very out of component of force in the reference of definition of the shell. The pressure

applied is positive according to the contrary meaning of the norm to the element (defined by the first 3 nodes of each mesh (cf [§4.25.3])). ◊PLAN

```
= "MOY"       ,   /"INF
           ",  /"SUP
           ",  /"MAIL
           "   , [DEFAULT          ] Makes it possible
```

to define a load vector force on the average, lower, higher level or of the mesh. If

the eccentring and d the thickness of h the shell are noted, the torsor

$(F2X , F2Y , F2Z , M2X , M2Y , M2Z)$ of the forces on the level defined by the user (excentré i.e) the torsor
$(F1X , F1Y , F1Z , M1X , M1Y , M1Z)$ of the forces in the plane of the mesh the formulas

of transition are the following ones: if the plane

- of computation is the plane of mesh: if the plane

$$F2=F1$$
$$M2=M1$$

- of computation is the excentré average average: if the plane

$$F2=F1$$
$$M2X=M1X-dx\,F1Y$$
$$M2Y=M1Y+dx\,F1X$$

- of computation is the excentré higher average: if the plane

$$F2=F1$$
$$M2X=M1X-d+\frac{h}{2}.F1Y$$
$$M2Y=M1Y+d+\frac{h}{2}.F1X$$

- of computation is the excentré lower average: /“MOY' one

$$F2=F1$$
$$M2X=M1X-d-\frac{h}{2}.F1Y$$
$$M2Y=M1Y+d-\frac{h}{2}.F1X$$

`applies`	the torsor D” forces to the excentré average average/“INF' one
`applies`	the torsor D” forces to the lower skin/“SUP' one
`applies`	the torsor D” forces to the higher skin/“MAIL' one
`applies`	the torsor D” forces to the level of the plane of the mesh

Modelizations

4.36.4 and meshes This loading

applies to the types of meshes and the following modelizations: Net Modelization

	TRIA3 QUAD4
DKT, DST	QUAD4 Q4G
SORTED	7 QUADS
9 COQUE_3D	Note:

This loading

- *is available only on one three-dimensional mesh (defined by COOR_3D* `). The rule`

- of remanence (see U1.03.00) applies between the various quantities which one can affect: FX, FY, … . Key word

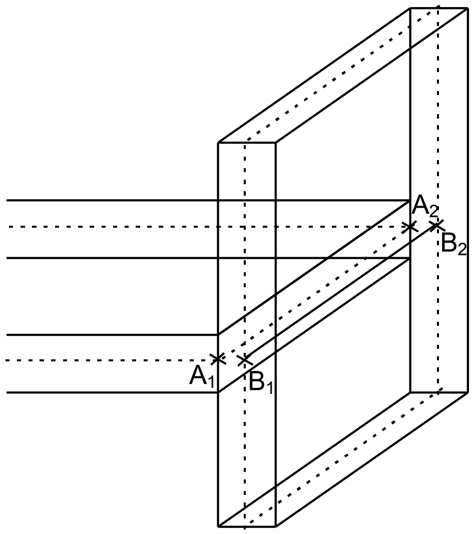
4.37 LIAISON_COQUE `Drank Factor key word`

4.37.1

making it possible to represent the connection between shells by means of linear relations. The classical approach admits that two planes with a grid in shells are cut according to a line which belongs to the mesh of structure. That has L

"disadvantage of twice counting the volume which is L" intersection of the two shells. The idea

is thus to stop the mesh of a shell perpendicular to a shell given to the level of the higher or lower skin of the latter. One represented



in features full volume with the shells and in dotted lines the average planes of these shells (which result from the mesh). The horizontal

shell stops in and the projection $A_1 A_2$ of on the average $A_1 A_2$ level of the vertical shell is (that one $B_1 B_2$ represented in full features). The link

between the 2 shells is made by connections of solid body between the nodes in with respect to the segments and. For example $A_1 A_2$ $B_1 B_2$

for the nodes and, one A_1 B_1 will write the formula (valid in small rotations): and equality

$$U(B_1) = U(A_1) + \Omega(A_1) \wedge A_1 B_1$$

of rotations: Syntax

$$\Omega(B_1) = \Omega(A_1)$$

4.37.2 for AFFE_CHAR_MECA

```
•   and AFFE_CHAR_MECA_F LIAISON_COQUE
    = _F (♦|GROUP
           _MA_1=l_gma1     , [    l_gr_maille                   ] |
MAILLE_
              1=l_ma1, [l_maille                              ] |GROUP_NO
              _   1=l_gno1, [      l_gr_noeud                    ] |
NOEUD_1
              =l_no1  , [l_noeud              ]                ♦|GROUP
           _MA_2=l_gma2     , [    l_gr_maille                   ] |
MAILLE_
              2=l_ma2, [l_maille                              ] | GROUP_NO
                 _2=l_gno2, [     l_gr_noeud                    ] |
NOEUD_
```

2= l_no2, [l_noeud] ◊NUMÉRIQUE_

LAGR=/ "NORMAL " , [DEFAULT] /"APRES"
,) Operands

4.37.3 |GROUP_MA_1

```
 | MAILLE_1
| GROUP_NO_1
 |NOEUD_1
```

Using

key words GROUP_MA_1 , MAILLE_1 , GROUP_NO_1 and NOEUD_1 , one constitutes the first nodes list (nonredundant) representing the trace of the shell perpendicular to the current shell. On our

example, they would be the nodes of the segment or the segment $B_1 B_2$. |GROUP_MA_2 $A_1 A_2$

```
 |MAILLE_2
| GROUP_NO_2
 | NOEUD_2
Using
```

key words GROUP_MA_2 , MAILLE_2 , GROUP_NO_2 and NOEUD_2 , one constitutes the second nodes list (nonredundant) pertaining to the perpendicular shell and in the nodes of the first list. Opposite is adjusted by the program according to the criterion of smaller distance. On our

example if the first list is drawn up by the nodes of, the second $A_1 A_2$ list is drawn up by the nodes of. ◊NUMÉRIQUE $B_1 B_2$ _

```
LAGR=/ "NORMAL          " ,   [DEFAULT]            /"DEFAULT
                        ",  See key word
```

LIAISON_SOLIDE [§4.19] . Important

remarks: After

1) *the key words GROUP_MA _ , MESH _ , GROUP_NO _and NOEUD _ , a node can appear several times, it is the program which is given the responsibility to eliminate the useless occurrences and thus to obtain a nonredundant list of nodes. After L*
2) *"elimination of the useless occurrences of the nodes in the two lists of nodes, these two lists must be imperatively equal length. Meshes*
3) *given after key words GROUP_MA_1 , GROUP_MA_2 , MAILLE_1 and MAILLE_2 are of meshes of edge of the type SEG2 or SEG3 of the shell elements and for which one N" does not have inevitably affected a mechanical modelization. Key word*

4.38 RELA_CINE_BP Drank This kind of

4.38.1

loading can be defined for a mechanical system including one concrete structure and its cables of prestressing. Initial profiles of tension in the cables, as well as the coefficients of the kinematic relations between the degrees of freedom of the nodes of the cables and the degrees of freedom of the nodes of concrete structure are given beforehand by the operator DEFI_CABLE _BP [U4.42.04]. The concepts cabl_precont produced by this operator bring all the necessary information to the definition of the loading. The multiple

occurrences are authorized for factor key word the RELA_CINE_BP , in order to make it possible in the same call to operator AFFE_CHAR_MECA to define the contributions of each group of cables having been the object of distinct calls to operator DEFI_CABLE _BP [U4.42.04]. With each group of cables considered, defined by a concept cabl_precont , is associated an occurrence with factor key word the RELA_CINE_BP . The loading

thus defined is then used to calculate the state of equilibrium of the group concrete structure/cables of prestressing. However, the taking into account of this kind of loading is not effective in all the operators of resolution. The loading of the type RELA_CINE_BP is recognized for time only by the operator STAT_NON_LINE [U4.51.03], option COMP_INCR exclusively . Syntax

4.38.2 (AFFE_CHAR_MECA only) RELA_CINE

```
_BP=_F (♦CABLE
                 _BP=cabl_pr   ,      [  cabl_precont          ] ◊SIGM_
                 BPEL=/ "OUI',    /   "   NON", [
                                      DEFAULT   ]            ◊RELA_
                 CINE=/ "OUI', [      DEFAULT      ]            /"NON",
                                         ◊DIST_
                 MIN=dmin  , [R]         ) Operands
```

4.38.3 ♦CABLE

_BP=cabl_pr Concept

of the cabl_precont type produces by the operator DEFI_CABLE _BP [U4.42.04]. This concept brings on the one hand the card of the initial stresses in the elements of the cables of the same group, and on the other hand the lists of the kinematic relations between the degrees of freedom of the nodes of these cables and the degrees of freedom of the nodes of concrete structure. ◊SIGM_

BPEL=/ "OUI', / " NON", [DEFAULT] Indicating

of standard text by which one specifies the taking into account of the initial stresses in the cables; the default value is "NON". In
the case "NON", only the liaisonnement kinematical one is taken into account. It is useful if one connects then STAT_NON_LINE qu "one has cables of prestressing. For the first STAT_NON_LINE it is necessary to have put "OUI', so that L" one sets up the tension in the cables. On the other hand, for the STAT_NON_LINE following , one should regard as loading only kinematical connections and thus define the loading with SIGM_BPEL = "NON", if not the tension is counted twice. Since
the restitution the macro one to put in tension the cables, the user should not need any more to make an AFFE_CHAR_MECA with SIGM_BPEL = "OUI', that should thus avoid the risks of error. ◊RELA_

CINE=/ "OUI', [DEFAULT] /"NON", Indicator

of type text by which one and the specifies the taking into account of the kinematic relations between the degrees of freedom of the nodes of the cables degrees of freedom of the nodes of concrete structure; default value “OUI'. ◊ `DIST_`
`MIN=dmin , [R] (see LIAISON_SOLIDE 4.18) Key word`

4.39 FORCE_ELEC Drank Factor key word

4.39.1

usable to apply the force of LAPLACE acting on a principal driver, due to the presence of a secondary driver right (not being based on part of Aster mesh) compared to this principal driver. In fact,

the loading defined by FORCE_ELEC has a modulus which must be multiplied by the temporal function of intensity specified by the operator DEFI_FONC _ELEC [U4.MK.10] to really represent the force of LAPLACE. The principal

driver leans on whole or part of the Aster mesh made up of linear elements in space and defined in this operator by one or more meshes, of the mesh groups or the totality of the mesh. Note:

When

the secondary driver is not rectilinear key word INTE_ELEC [§4.40] will be used. Syntax

4.39.2 FORCE_ELEC

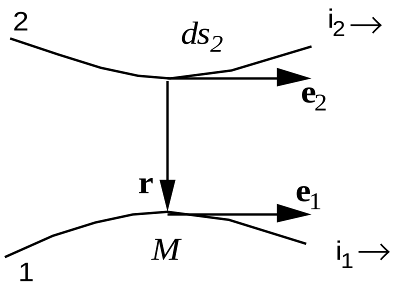
```
= _F (♦/TOUT
                    = ' OUI' ,      /|  NET
                          =lma   , [l_   mesh                  ] |
GROUP_MA
                             =lgma, [l_gr_maille                  ] ♦/|
FX
                    = fx   ,   [  R] | FY                          = fy
                          ,   [R] | FZ                        = fz
                          ,   [R]/♦POSITION
                          =    ' PARA',  ♦/TRANS
                          = (ux     , uy, uz     ,), [l_R] /DIST
=d,
                             [R ] /POINT2                        =
                             (x2 , y2, z2     ,), [l_R]/♦POSITION
                          =    ' FINI',  ♦POINT
                          1= (x1 , y1, z1        ,), [l_R] ♦POINT
                          2= (x2 , y2, z2        ,), [l_R]/♦POSITION

                          =    ' INFI' ♦POINT
                          1= (x1 , y1, z1        ,), [l_R] ♦POINT
                          2= (x2 , y2, z2        ,), [l_R]) Function
```

4.39.3 of space the function

of component space the linear density of force of LAPLACE exerted in a point of driver M 1 (principal driver) by the elements of driver 2 (secondary driver) is: In the case of

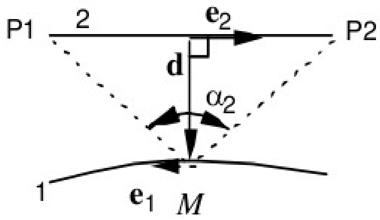
$$f(M)=\frac{e_1}{2}\wedge\int_2\frac{e_2\wedge r}{\|r\|^3}ds_2$$



avec $\|\mathbf{e}_1\| = \|\mathbf{e}_2\| = 1$

a secondary right and finished driver, this statement becomes: with

$$\boldsymbol{f}(M)=\frac{\boldsymbol{e_1}}{2}\wedge\frac{\boldsymbol{n}}{d}\left(\sin\alpha_1-\sin\alpha_2\right)$$



, In $\boldsymbol{n}=\frac{\boldsymbol{e_2}\wedge\boldsymbol{d}}{d}$ $d=\|\boldsymbol{d}\|$ $\|\boldsymbol{d}\|=1$

the typical case of the secondary driver infinite right, and tend α_1 α_2 towards, one has then $\frac{\pi}{2}$: Operands

$$\boldsymbol{f}(M)=\boldsymbol{e_1}\wedge\frac{\boldsymbol{n}}{\boldsymbol{d}}$$

4.39.4 |FORCE_ELEC

If

there are several secondary drivers infinite and parallel with the principal driver (key words COUR_PRIN and COUR_SECO in command DEFI_FONC _ELEC) one directly specifies the components of the direction of the force of LAPLACE who must be normalized to 1. /|FX = fx

, fx2 + fy2 + fz2 = 1. |FY = fy
, (fx, fy, fz) colinéaire by the strength of LAPLACE |FZ = fz
, If not,

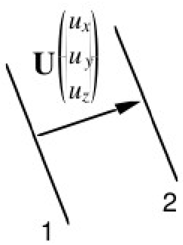
the direction of the force of LAPLACE can be defined by the position of the secondary single driver compared to the elements of the principal driver. /♦POSITION

/"PARA"

the secondary

driver is considered infinite and parallel with the principal driver. One can define his position of two ways: /TRANS:

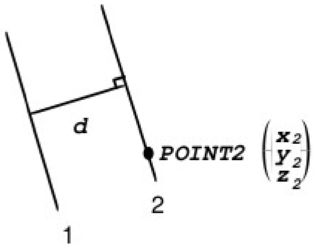
(ux uy uz) defines



$\mathbf{U}\begin{pmatrix}u_x\\u_y\\u_z\end{pmatrix}$ the translation bringing of the principal driver 1 to the

secondary driver 2 /DIST =

D, /POINT2
= (x2, y2, z2), the secondary

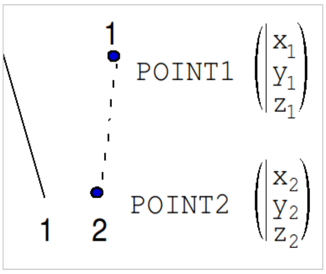


driver 2 is defined by its distance in driver 1 and one second point. /"FINI"

`the secondary`

driver is defined by two points corresponding at its ends POINT1 and `POINT2` . `POINT1` =

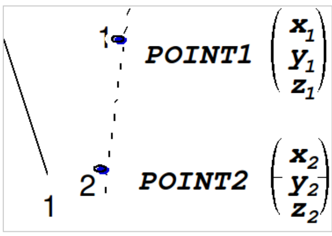
`(x1, y1, z1), POINT2 =`
`(x2, y2, z2),/"INFI"`



`the secondary`

driver is defined by two unspecified points POINT1 and `POINT2` . `POINT1` =

`(x1, y1, z1), POINT2 =`
`(x2, y2, z2), In`



both cases, it is preferable to choose POINT1 and `POINT2` such as flow circulates of POINT1 with `POINT2` . `Key word`

4.40 INTE_ELEC Drank Factor key word

4.40.1

usable to apply the force of LAPLACE acting on a principal driver, due to the presence of a secondary driver not necessarily right compared to this principal driver. In fact,

the loading defined by INTE_ELEC has a modulus which must be multiplied by the temporal function of intensity specified by the operator DEFI_FONC _ELEC [U4.MK.10] to really represent the force of LAPLACE. The principal

driver leans on part of Aster mesh made up of linear elements in space and defined in this operator by one or more meshes, of the mesh groups or the totality of the mesh. The secondary

driver is also based on part of Aster mesh made up of linear elements in the space and also specified in this operator by one or more meshes, of the mesh groups, or by a translation (or a symmetry planes) compared to the principal driver. Note:

The difference

of the use of key word INTE_ELEC compared to key word FORCE_ELEC lies in the fact that the geometry of the secondary driver can not be rectilinear and leans on part of Aster mesh only one describes here. Syntax

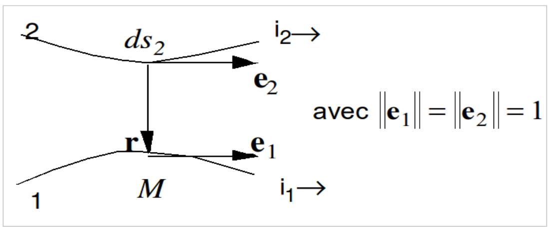
4.40.2 INTE_ELEC

```
   =_F (♦/TOUT        = '    OUI'  ,       /|  NET
                            =lma    , [l_    mesh                    ] |
GROUP_MA
                               =lgma, [l_gr_maille                   ] ♦/|NET
                2              = lma, [L    _maille                      ] |
GROUP_MA
                            2    = lgma, [l_gr_maille                    ]
/TRANS =
                         (   ux, uy, uz), [l_R] /SYME                     =
                         (   x0, y0   , z0, ux, uy, uz), [l_R]) Function
```

4.40.3 of space the function

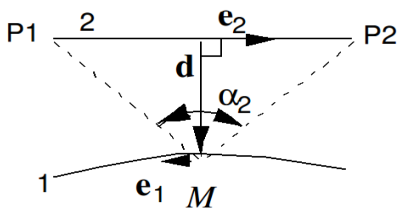
of space composing the linear density of forces of Laplace exerted in a point of driver M 1 (principal driver) by the elements of driver 2 (secondary driver) can be expressed: For each

$$f(M)=\frac{e_1}{2}\wedge\int_2 \frac{e_2\wedge r}{\|r\|^3}ds_2$$



element I of the secondary driver, one calculates his contribution from the preceding statement and one adds: with formula

$$f(M)=\sum_i \frac{e_1}{2}\wedge\frac{n}{d}(\sin\alpha_1-\sin\alpha_2)$$



$$\boldsymbol{n}=\frac{e_2\wedge\boldsymbol{d}}{d}\quad d=\|\boldsymbol{d}\|\quad \|\boldsymbol{n}\|=1$$

4.40.4 Operands/ MESH / GROUP_MA /MESH 2 /GROUP_MA 2/TRANS / SYME TOUT , MESH

, GROUP_MA:

The geometry of the principal driver defines where the loading is affected. MAILLE2, GROUP_MA2:

The geometry of the secondary driver defines. TRANS:

A translation of the principal driver defines in the secondary driver. SYME:

A symmetry compared to a plane defines (given by a point and the norm $(x_0\, y_0 z_0)$ common $(u_x\, u_y\, u_z)$ to the principal driver and the secondary driver). Key word

4.41 IMPE_FACE (" ACOUSTIC" Phenomenon) Drank IMPE_FACE

4.41.1

The factor key word makes it possible to apply an acoustic impedance, with a face defined by one or more meshes or mesh groups of type triangle or quadrangle. The values are directly given if the operator called is AFFE_CHAR_MECA ; if it is AFFE_CHAR_MECA_F , they come from a concept of type function . Syntax

4.41.2 for AFFE_CHAR_MECA

- IMPE_FACE

```
=_F (♦|NET               =lma    , [l_  mesh              ] |GROUP_MA
                         =lgma   , [l_gr_maille           ] ♦IMPE
                    =Q, [  R]) for
                AFFE_CHAR_MECA_F
```

- IMPE_FACE

```
=_F (♦|NET               =lma    , [l_  mesh              ] |GROUP_MA
                         =lgma   , [l_gr_maille           ] ♦IMPE
                    =Qf,    [function                       ])
```

Operand

4.41.3 IMPE_FACE IMPE_FACE

= Q (Qf) acoustic

Impedance applied to the face. Modelizations

4.41.4 and meshes the loading

S" applies to the types of meshes and the following modelizations: Type of

Mesh Modelization	TRIA3, SORTED
6 QUAD4, QUAD 8, QUAD9 3D_FLUIDE	Key word

4.42 VITE_FACE (" ACOUSTIC" Phenomenon) Drank VITE_FACE

4.42.1

The factor key word makes it possible to apply normal velocities, with a face defined by one or more meshes or mesh groups of type triangle or quadrangle. The values
are directly given if the operator called is AFFE_CHAR_MECA , if it is AFFE_CHAR_MECA_F , they come from a concept of type function. Syntax

4.42.2 for AFFE_CHAR_MECA

- VITE_FACE

```
=_F (♦|NET            =lma    , [l_  mesh                   ] |GROUP_MA
                      =lgma   , [l_gr_maille                ] ♦ VNOR
                   =V,   [R]) for
               AFFE_CHAR_MECA_F
```

- VITE_FACE

```
=_F (♦|NET            =lma    , [l_  mesh                   ] |GROUP_MA
                      =lgma   , [l_gr_maille                ] ♦VNOR
                  =Vf,    [function                           ])
```

Operand

4.42.3 VNOR VNOR = V

(Vf) normal

Velocity applied to the face. Modelizations

4.42.4 and meshes the loading

applies to the types of meshes and the following modelizations: Type of

Mesh Modelization	TRIA3, SORTED
6 QUAD4, QUAD 8, QUAD9 3D_FLUIDE	Key word

4.43 ONDE_PLANE Factor key word Drank

4.43.1

usable to impose a seismic loading by plane wave (only one occurrence is possible), corresponding to the loadings classically met at the time as of computations D" interaction soil-structure by the integral equations (see [R4.05.01]). Syntax

4.43.2 (AFFE_CHAR_MECA_F only) ONDE_PLANE

```
=_F (     ♦TYPE_  ONDE=ty   , [txm]           ♦DIRECTION
                     = (kx, ky, kz       ), [l_R]       ♦FONC_
                  SIGNAL=f , [function                          ] ◊|GROUP
                  _MA=lgma     , [l_gr_maille                      ] |
MAILLE=
                     lma , [l_maille                           ]) Operands
```

4.43.3 ♦TYPE_

ONDE=ty , Type of

the wave: "P" wave of compression "SV" waves
of shears "SH" waves
of shears ♦DIRECTION

= (kx, ky, kz), Direction

of the wave. ♦FONC_

SIGNAL=v , Derived

from the profile of the wave: for. $v(t)$ In harmonic $t \in [0,+\infty[$

, one plane wave elastic is characterized by its direction, its pulsation and its type (wave for P the compression waves, waves or for SV SH the waves of shears). Out of transient, the data of the pulsation, corresponding to one standing wave in time, must be replaced by the data of a profile of displacement which one will take into account the propagation in the course of time in the direction of the wave. More precisely

, one characterizes: one wave

- by the function P one wave
- by the function S With: $u(x,t)=f\left(k\cdot x-C_s t\right)\wedge k$

, unit vector

- k of direction then
- f represents the profile of the wave given according to the direction. Caution: k

it is the velocity which $v(t)=\dot{u}(t)$ the user gives in FONC_SIGNAL . ◊|GROUP

_MA=l_gr_maille , |MAILLE=
l_maille , List of

meshes of absorbing borders concerned with the introduction of the incident wave. If nothing is given, by defaults all, it is meshes of modelization ABSO which is concerned. Modelizations

4.43.4 and meshes Standard of

Mesh Modelization	MECA_FACE
_*MEPLSE2, MEPLSE3 3D_ABSO	2D_ABSO Key word

4.44 ONDE_FLUI (" ACOUSTIC" Phenomenon) Drank ONDE_FLUI

4.44.1

The factor key word `makes it possible` to apply an amplitude of pressure of sinusoidal incident wave arriving normally at a face defined by one or more meshes or mesh groups. Syntax

4.44.2 for AFFE_CHAR_MECA

- ONDE_FLUI

```
=_F (♦|NET          =lma   , [l_  mesh           ] |GROUP_MA
                    =lgma  , [l_gr_maille        ] ♦PRES
               =P, [  R]) for
        AFFE_CHAR_MECA_F
```

- Not developed

. Operand

4.44.3 NEAR NEAR = P

, Amplitude

of pressure of sinusoidal incident wave arriving normally at the face. Modelizations

4.44.4 and meshes the loading

applies to the types of meshes and the following modelizations: Type of

Mesh Modelization	TRIA3, SORTED
6 QUAD4, QUAD 8, QUAD9 3D_FLUIDE	SEG2, SEG3
2D_FLUIDE	, AXIS_FLUIDE Key word

4.45 FLUX_THM_REP Drank Factor key word

4.45.1

usable to apply to a field of continuum 2D or 3D defined by meshes or mesh groups a heat flux and/or a contribution of fluid mass (hydraulic flux). Syntax

4.45.2 for AFFE_CHAR_MECA

```
•  FLUX_THM_REP
   =_F (♦/TOUT       = '    OUI'  ,      /|  NET
                            =lma      , [l_maille                    ] |
GROUP_MA
                               =lgma   , [l_gr_maille                ] ♦|
FLUN
                         =PHIT     , [R]            |FLUN_HYDR
                               1=phie, [R]          |FLUN_HYDR
                               2=phiv [R]       )    for
                      AFFE_CHAR_MECA_F
•  FLUX_THM_REP
   =_F (♦/TOUT       = '    OUI'  ,      /|  NET
                            =lma      , [l_maille                    ] |
GROUP_MA
                               =lgma   , [l_gr_maille                ] ♦|
FLUN
                         =phiTf    , [function                       ] |
FLUN_HYDR
                               1=phief, [function                ] |
FLUN_HYDR
                               2=phivf, [function                ])
Operands
```

4.45.3 |FLUN =

phiT , Value of

heat flux with:

$$\phi_T = \lambda_T \frac{\partial T}{\partial n} + h_m^e \phi^e + h_m^v \phi^v + h_m^a \phi^a$$

: mass h_m^l enthalpy of the fluid: mass

h_m^v enthalpy of the vapor: mass

h_m^a enthalpy of the air and are

ϕ^e ϕ^v the below definite hydraulic flux |FLUN_HYDR1

= phie, Value of

hydraulic flux associated with the component water |FLUN_HYDR2

= phiv, Value of

hydraulic flux associated with the component air with:

$$\phi^e = \rho_e \left(\nabla P_e - \rho_e \boldsymbol{g} \right) \cdot \boldsymbol{n}$$

$$\phi^v = \rho_v \left(\nabla P_v - \rho_v \boldsymbol{g} \right) \cdot \boldsymbol{n}$$

: densit ρ_e of the fluid: density

y

ρ_v of the vapor: pressure

P_e of fluid (PRE1): normal

P_v flux and steam pressure (

4.45.4 PRE2) Modelizations meshes

apply to the types of meshes and the following modelizations: Type of

Mesh Modelization	SEG2 SEG3
FACE 8 D_ PLAN_YYYY	AXIS_YYYY , D_PLAN_YYYY 3D_YYYY with YYYY

= THM or THH or THHM or HM or HHM . Key word

4.46 ARETE_IMPO Drank Factor key word

4.46.1

usable to impose, with all the nodes of a voluminal edge of the elements defined by a mesh or a mesh group, one or more values of displacement (or certain associated quantities). Syntax

4.46.2 ARETE_IMPO

```
=_F (♦/MAILLE=lma             , [l_maille                          ]
/GROUP_MA
                    =lgma  , [l_gr_maille                   ] ◊SANS_
                 MAILLE=lma1      , [l_maille                ] ◊SANS_
                        GROUP_MA=lgma1, [      l_gr_maille          ] ◊SANS_
                 NOEUD=lno1       , [l_noeud      ]          SANS_GROUP
                        _NO=lgno1, [     l_gr_ node          ] ♦/|DX=
                 ux, [  R  ]    |     DY=uy                  , [
                       R]   |     DZ=uz                  , [
                       R]   |     PRES=p                 ,
                       [R  ] |PHI=phi
                       ,[  R]     |  TEMP=T           ,
                       [R  ] |PRE1=pr
                       1,   [R]   |  PRE2=pr
                       2,   [R]   /  |DTAN=ut
                    ,   [  R]) Operands
```

4.46.3 ◊SANS_

```
MAILLE=lma1      , [l_maille                ] ◊SANS_
       GROUP_MA= lgma1,  [L  _gr_maille        ] ◊SANS_
NŒUD=lno1    , [l_noeud          ]          ◊SANS_
       GROUP_NO=lgno1, [      l_gr_noeud          ] Indicates
```

that one wants to omit the nodes of the lists lma1, lgma 1, lno1, lgno1, of the list lma or lgma. Example:

ARETE_IMPO = (_F (GROUP_NO =FBas, DX =0, DY =0, DZ =0), _F (GROUP_MA =ALaterale , SANS_GROUP_NO = NBas, DTAN =10),) the meaning

of the 2nd occurrence of ARETE_IMPO is: " for all the nodes of the mesh group, except $Alateral$ for $DTAN=10$ those of the nodes group". This $Nbas$ makes it possible not to have redundant boundary conditions. ♦/|DX

= | DY = |

DZ = |
PRES= |
PHI = |
TEMP= |
PRE1= |
PRE2=
the components

, imposed on all the nodes belonging to meshes specified, are defined in reference **GLOBAL of** definition of the mesh. The edges

considered consist of SEG2 or SEG3 . Note:

The components

NEAR and PHI can intervene only on elements of modelizations "3D_FLUIDE " and "FLUI_STRU ", component PHI on elements of modelization "2D_FLUI_PESA ", components TEMP, PRE1 , PRE2 on elements of modelizations THM. /| DTAN =

the imposed

components are defined according to the tangent with a mesh (local coordinate system**).**
DTAN: tangential
component (see [U4.44.01 §4.1]).