Operator DEFI_GROUP

1 Drank

To define in an existing mesh, new nodes groups or of meshes. This can facilitate the definition of new loci for inputs or postprocessings.

To create new groups, one uses topological, logical or geometrical criteria.

Of mesh type modify a data structure , squelette or grid.

Contents

2 Syntax

```
my (mesh) =DEFI_GROUP          (

          ◊reuse  = / my,
                    /gr.,
          ♦   | MAILLAGE =ma      ,                    /         [mesh]
                                                     /         [squelette]
              | GRILL =gr      ,                     /         [grid]

          ♦   |  DETR_GROUP_MA =_F    (
              ♦NOM    =lgma                  ),            [l_group_ma]
              |  DETR_GROUP_NO =_F    (
              ♦NOM    =lgno                  ),            [l_group_no]


              |  CREA_GROUP_MA =   (_F (
              ♦NOM    =gma                   ,               [identifier]
              ◊TYPE_MAILLE=/             "TOUT"           [DEFAULT]
                                       /"3D" /"2D" /"1D"
                                       /"SEG2" /"TRIA3"/"QUAD4"
                                       /"QUAD8"/ … /"PYRAM13"
              ♦/MAILLE=lmail                      ,
[l_maille]
                  /TOUT         =  ' OUI'   ,
                  /INTERSEC     =lgma    ,                [l_group_ma]
                  /UNION        =lgma    ,                [l_group_ma]
                  /DIFFE        =lgma    ,                [l_group_ma]
                  /GROUP_MA     =gma         ,               [group_ma]
                     /NUME_INIT    =/nuini         ,          [I]
                                    /1        ,          [DEFAULT]
                        NUME_FIN  =nufin       ,          [I]

                     /POSITION   =     "INIT"   ,
                                    /  "FIN"  ,
                                    /  "MILIEU",

                  /OPTION       =  ' FACE_NORMALE" ,
                     ♦/ANGL_NAUT         = (has, b) ,       [l_R]
                        /VECT_NORMALE = (X, there, Z) ,       [l_R]
                     ◊ANGL_PREC     = has   ,               [R]
                                    / 0.5,              [DEFAULT]
                     ◊VERI_SIGNE   = / "NON",
                                    / "OUI',            [DEFAULT]

                  /OPTION       =  ' SPHERE",
                     ♦/POINT       = (X, there, Z),              [l_R]
                        /NOEUD_CENTRE = No,           [node]
                        /GROUP_NO_CENTRE  = grno,         [group_no]
                     ♦RAYON  = R,                     [R]

                  /OPTION       =  ' CYLINDRE",
                     ♦/POINT       = (X, there, Z),              [l_R]
                        /NOEUD_CENTRE = No,          [node]
                        /GROUP_NO_CENTRE  = grno,         [group_no]
                     ♦RAYON  = R,                     [R]
                     ♦/ANGL_NAUT         = (has, b),          [l_R]
                        /VECT_NORMALE = (X, there, Z),       [l_R]
```

```
                        /OPTION       =  ' TAPE”,
                           ♦/POINT       = (X, there, Z),              [l_R]
                              /NOEUD_CENTRE = No,            [node]
                              /GROUP_NO_CENTRE  = grno,          [group_no]
                           ♦/ANGL_NAUT        = (has, b),          [l_R]
                              /VECT_NORMALE = (X, there, Z),       [l_R]
                           ♦DIST   = D,                           [R]

                        /OPTION       =  ' APPUI”,
                           ♦/GROUP_NO       = lgno      ,          [l_group_no]
                              /NOEUD  = lno,                  [l_noeud]
                           ♦TYPE_APPUI    =  “AU_MOINS_UN”
                                          /“TOUT”
                                          /“SOMMET”
                                          /“MAJORITE”
                        /OPTION       =  ' FISS_XFEM”,
                           ♦FISSURE   = (fiss1, fiss2,…),
      [l_fiss_xfem]
                           ♦TYPE_GROUP    =  “XFEM”
                                          /“HEAVISIDE”
                                          /“CRACKTIP”
                                          /“MIXTE”
                                          /“FISSUREE”
                        ),),


                  |  CREA_GROUP_NO =  (_F (
                     /♦NOM      =gno                  ,             [identifier]
                        ♦/NOEUD=lnoeu                ,             [l_noeud]
                           /INTERSEC     =lgno    ,              [l_group_no]
                           /UNION        =lgno    ,              [l_group_no]
                           /DIFFE        =lgno    ,              [l_group_no]
                           /GROUP_NO     =gno        ,              [group_no]
                              /NUME_INIT       =/nuini        ,       [I]
                                              /1        ,       [DEFAULT]
                                 NUME_FIN=nufin            ,       [I]
                              /POSITION     =     “INIT”   ,
                                            /  “FIN”  ,
                                            /  “MILIEU”,
                           /OPTION       =  ' ENV_SPHERE”,
                              ♦/POINT       = (X, there, Z),           [l_R]
                                 /NOEUD_CENTRE = No,         [node]
                                 /GROUP_NO_CENTRE  = grno,      [group_no]
                              ♦RAYON  = R,                   [R]
                              ♦PRECISION    = eps ,             [R]
                           /OPTION      =   ' ENV_CYLINDRE”,
                              ♦/POINT       = (X, there, Z),           [l_R]
                                 /NOEUD_CENTRE = No,         [node]
                                 /GROUP_NO_CENTRE  = grno,      [group_no]
                              ♦RAYON  = R,                   [R]
                              ♦/ ANGL_NAUT   = (has, b),         [l_R]
                                 /VECT_NORMALE= (X, there, Z),     [l_R]
                              ♦PRECISION    = eps,               [R]
                           /OPTION       =  ' PLANE”,
                              ♦/POINT       = (X, there, Z),           [l_R]
                                 /NOEUD_CENTRE = No,         [node]
```

```
                    /GROUP_NO_CENTRE  = grno,        [group_no]
                  ♦/ ANGL_NAUT    = (has, b),        [l_R]
                    /VECT_NORMALE= (X, there, Z),      [l_R]
                  ♦PRECISION     = eps,              [R]

                /OPTION       =  ' SEGM_DROI_ORDO",
                  ♦/NOEUD              = lno,        [l_noeud]
                    /GROUP_NO      = gno2  ,        [group_no]
                  ♦/NOEUD_ORIG       = noA  ,        [node]
                    /GROUP_NO_ORIG    = gnoA   ,       [group_no]
                  ♦/NOEUD_EXTR       = noB  ,        [node]
                    /GROUP_NO_EXTR    = gnoB   ,       [group_no]
                  ♦PRECISION    =prec ,             [R]
                  ♦CRITERE     =       "RELATIF"   ,
                                 /  "ABSOLU"  ,

                /OPTION       =  ' NOEUD_ORDO",
                  ♦GROUP_MA  =gmaAB       ,          [group_ma]
                  ◊/NOEUD_ORIG       = noA  ,        [node]
                    /GROUP_NO_ORIG    = gnoA   ,       [group_no]
                  ◊/NOEUD_EXTR       = noB  ,        [node]
                    /GROUP_NO_EXTR    = gnoB   ,       [group_no]
                  ◊VECT_ORIE      = (vx, vy, [vz]),    [l_R]

                /OPTION       =  ' TUNNEL",
                  ♦/TOUT             = "OUI'
                    /  |  GROUP_MA     = lgma ,    [l_group_ma]
                       |  NET       = lmai ,    [l_maille]
                  ♦/MAILLE_AXE       = noA  ,        [l_maille]
                    /GROUP_MA_AXE = gnoA   ,        [l_group_ma]
                  ♦/NOEUD_ORIG       = noA  ,        [node]
                    /GROUP_NO_ORIG    = gnoA   ,       [group_no]
                  ♦RAYON           = R  ,           [R]
                  ◊LONGUEUR         = long  ,        [R]

                /OPTION       =  ' INCLUSION",
                  ♦GROUP_MA      = lgma  ,          [l_group_ma]
                  ♦CAS_FIGURE       =    /"2D"
                                         /"3D"
                                         /"2.5D"
                  ◊DISTANCE_MAX = distma         [R]
                  ♦GROUP_MA_INCL    = lgma_inc  ,    [l_group_ma]
                  ◊MAILLAGE_INCL    = ma_inc ,       [mesh]


                /OPTION       =  ' INTERVALLE_VALE",
                  ♦CHAM_GD      = chno   ,          [cham_no]
                  ♦NOM_CMP      = cmp,              [TXM]
                  ♦VALE         = (vmin, vmax)      [R]

              /OPTION       =  ' FISS_XFEM',
                  ♦FISSURE   = (fiss1, fiss2,…),       [l_fiss_xfem]
                  ♦TYPE_GROUP    =  "XFEM"
                                   /"HEAVISIDE"
                                    /"CRACKTIP"
                                   /"MIXTE"
                                   /"ZONE_MAJ"
                                   /"TORE"
                  #Si  TYPE_GROUP = "TORE":
```

```
                    ♦RAYON_TORE    = R,                    [R]

                    /GROUP_MA   = lgma,       [l_identificator]
                       ◊NOM           = lgno,             [l_group_no]
                       ◊CRIT_NOEUD    =  "TOUS"   ,       [DEFAULT]
                                        /"SOMMET",
                                        /"MILIEU",
                                        /"CENTER",

                       /TOUT_GROUP_MA   : "OUI',
                    ),),

                 ◊ALARME=/           "OUI',                      [DEFAULT]
                             /   "NON",

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

Standard of result:

So MESH: maillagealors : mesh: squelettesquelette
If GRILL : grillealors : roast

3 Operands

3.1 General information on the operands

This command in the same way treats the concepts of mesh type or squelette. In the continuation one will use the vocabulary “mesh”.

This command makes it possible to define new mesh groups (or nodes groups) in an existing mesh: one enriches the mesh my.

The definition of a new group can be done in several ways:

- in extension: key keys NETS or NOEUD,
- by Boolean operation on existing groups: intersection (INTERSEC), meeting (UNION) or difference (DIFFE),
- according to a geometrical criterion: meshes whose node belongs to a given sphere,…
- for the nodes groups, by referring to existing mesh groups. The nodes group thus defined meshes **contains** all the nodes of mesh group origin (key keys TOUT_GROUP_MA and GROUP_MA).

The operator treats initially key word CREA_GROUP_MA so that one can make use of the mesh groups thus defined in key word CREA_GROUP_NO.

For each occurrence of a key word CREA_GROUP_MA (_NO) one defines a new named group (key word NOM). This new group can then be re-used in the following occurrences to define new groups by intersection, meeting,…

key keys DETR_GROUP_MA and DETR_GROUP_NO make it possible “to destroy” mesh groups or nodes. Meshes and the nodes of these groups are not removed, they are only the definitions of the groups which are unobtrusive. These keywords are useful for example in the loops python when one wants to create a group with each iteration of the loop: one starts by destroying this group then one recreates it under the same name. That avoids changing name of group to each iteration.

3.2 Operands MAILLAGE and GRILL

♦ | MAILLAGE = my

my is the name of the mesh which one wants “to enrich”.

| GRILL = gr.

gr. is the name of auxiliary grid which one wants “to enrich”.

3.3 Key keys DETR_GROUP_MA and DETR_GROUP_NO

These two keys key factor make it possible to remove the definition of mesh groups or nodes. These keywords are sometimes necessary because the code stops in fatal error if one tries to create a group whose name is already used. It is necessary to destroy the group before being able to re-use its name. The behavior of the two key keys is similar and we will speak here only about DETR_GROUP_MA.

Syntax:

```
DETR_GROUP_MA=_F (NOM= (gm1, gm2,…)),
```

The key word factor DETR_GROUP_MA is a priori répétable but it is never necessary because key word NOM makes it possible to indicate a list of names of groups to be destroyed (gm1, gm2,…).

It is important to know that all the occurrences of key word DETR_GROUP_MA are treated **before** those of key word CREA_GROUP_NO because the purpose of this key word is to be able to re-use the destroyed name. It also should be known that the destruction of a non-existent group does not involve any alarm message. These choices make it possible for example to make in a loop python:

```
for I in arranges (N):
    DEFI_GROUPE (reuse=MA, MAILLAGE=MA,
                 DETR_GROUP_MA=_F (NOM ("GM1",),
                 CREA_GROUP_MA=_F (NOM=' GM1',…
```

At the time of the first iteration, group "GM1" does not exist, one requires his destruction but no alarm message is transmitted.

Note:

As the destruction takes place at the beginning of the command, it is impossible to modify a group by making only one call with DEFI_GROUP . For example, one cannot make "enlarge" (in a loop) a group by adding a small group (b1) to him:

```
for I in arranges (N):
    b1=nouveau group…
    DEFI_GROUP (reuse=MA, MAILLAGE=MA,
    CREA_GROUP_MA=_F (NOM=' tout', UNION= ("all", "b1"),),)
```

to do that, DEFI_GROUP should be called twice :

```
for I in arranges (N):
    b1=nouveau group…
    DEFI_GROUP (reuse=MA, MAILLAGE=MA,
       DETR_GROUP_MA=_F (NOM=' tout2'),
       CREA_GROUP_MA=_F (NOM=' tout2', UNION= ("all", "b1"),),)
    DEFI_GROUP (reuse=MA, MAILLAGE=MA,
       DETR_GROUP_MA=_F (NOM=' tout'),
       CREA_GROUP_MA=_F (NOM=' tout', UNION= ("tout2", "b1"),),)
```

3.4 Key word CREA_GROUP_MA

| CREA_GROUP_MA

an occurrence of this key word factor makes it possible to define a new mesh group.

3.4.1 Operand NOM

♦NOM = gma

One gives here the name (with "quotes") of the new mesh group.

3.4.2 Operand TYPE_MAILLE

```
◊TYPE_MAILLE=/      "TOUT"/"3D"/"2D"/"1D"   (DEFAUT=' TOUT')
                 /"SEG2"/"TRIA3"/"QUAD4"/…/"PYRAM13"
```

This key word makes it possible to filter meshes that one will put in the new mesh group. By default, it does not have there a filter, but if the user writes for example: TYPE_MAILLE=' 2D', the group created will contain only the meshes surface ones.
The user can also filter the group to be created for a kind of mesh individual (TRIA3, HEXA27, …). All the types of meshes (POI1, SEG2, SEG3, SEG4, …, PYRAM13) are authorized.

Examples :
CREA_GROUP_MA=_F (NOM=' VOLUM', "TOUT=' OUI', TYPE_MAILLE=' 3D')
makes it possible to create the group of all the meshes voluminal ones (HEXA, PENTA,...) mesh.

CREA_GROUP_MA=_F (NOM=' VOLH27', "GROUP_MA=' GMA1", TYPE_MAILLE=' HEXA27')
makes it possible meshes to create the group of all of type "HEXA27" contained in the GROUP_MA. GMA1.

3.4.3 Operand NETS

/MAILLE = lmail

This key word makes it possible to define the mesh group in extension: the list of the component is meshes given.

3.4.4 Operand TOUT

/TOUT = "OUI'

This key word makes it possible to define a group containing all meshes mesh.

3.4.5 Operand INTERSEC

/INTERSEC = (gma1, gma2, gma3,…),

The new mesh group will be obtained by taking all meshes gma1 which also belong to gma2, gma3,…. The order of meshes remains that of gma1.

3.4.6 Operand UNION

/UNION = (gma1, gma2, gma3,…)

The new mesh group will be obtained by taking all meshes gma1, then by adding meshes gma2 which do not belong to gma1, then those of gma3 which belong neither to gma1 nor with gma2, etc

3.4.7 Operand DIFFE

/DIFFE = (gma1, gma2, gma3,…)

The new mesh group will be obtained by taking all meshes gma1 which do not belong to the other groups of the list. The order of meshes remains that of gma1.

3.4.8 Sub-group of an existing group: key words GROUP_MA / POSITION / NUME_INIT / NUME_FIN

One can meshes create a new group of mesh by selecting some existing group.

1st possibility:

One creates a group **of only one mesh** by specifying by key word POSITION the required mesh.

Example:

```
CREA_GROUP_MA = _F (   GROUP_MA = G1,    POSITION = "INIT",  NOM = G1I)
```

group G1I contains the 1st mesh of the G1 group.

2nd possibility:

One creates a group containing meshes lain between the rows nuini and nufin (included) in an existing group.

Example:

```
CREA_GROUP_MA=_F (GROUP_MA = G1,  NUME_INIT = 3   ,   NUME_FIN = 7,
NOM = G1P)
```

group G1P contains meshes the 3,4,5,…, 7 of G1.

Caution:

> *These keywords use the notion **of order** of meshes in a mesh group. This order is often unknown to the user. It can depend on the preprocessor. It is the order of meshes at the time of the definition of the GROUP_MA in the Aster mesh file.*

3.4.9 Operand OPTION = "FACE_NORMALE"

/OPTION = "FACE_NORMALE"

This option makes it possible meshes to define a GROUP_MA made up of surface whose norm is parallel to the direction of the vector defined by its components if one uses key word VECT_NORMALE or with that of the first vector of the new base defined by the change of reference due to the nautical angles.

In 3D, it is supposed that the meshes surface ones are plane facets. They are of type TRIA3, TRIA6, QUAD4, QUAD8 or QUAD9. If one calls $\boldsymbol{X}_1$ $\boldsymbol{X}_2$, and the $\boldsymbol{X}_3$ vectors position of the first three nodes tops of the element, the norm is determined by the cross product: $(\boldsymbol{X}_2-\boldsymbol{X}_1)\wedge(\boldsymbol{X}_3-\boldsymbol{X}_1)$.

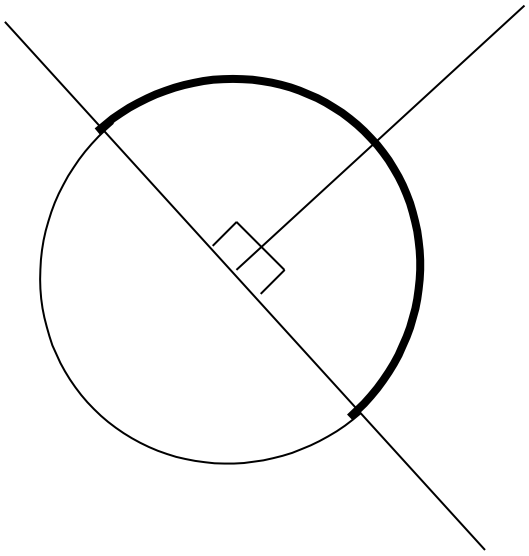
In 2D, it is supposed that the meshes surface ones are right segments. They are of type SEG2 or SEG3. If one and the $\boldsymbol{X}_1$ calls $\boldsymbol{X}_2$ vectors position of the two nodes ends of the element, the norm is defined by $(\boldsymbol{X}_2-\boldsymbol{X}_1)\wedge z$ where z is the unit vector perpendicular to the plane and where one has affected 0. like third component with $(\boldsymbol{X}_2-\boldsymbol{X}_1)$.

Note:

A mesh "facet" will be retained if its norm is colinéaire with the normal vector defined by VECT_NORMALE . This condition must be checked except for a certain accuracy (key word ANGL_PREC).

When one chooses an ANGL_PREC (for example 30. degrees), one defines in fact the group of meshes whose norm belongs to the cone of axis VECT_NORMALE and point angle ANGL_PREC .

This can be used (for example) to gather meshes half wraps spherical (ANGL_PREC = 90.).



3.4.9.1 Operand ANGL_NAUT

♦/ANGL_NAUT =aen 2D
(has, B) in 3D

the nautical angles (has, b) defined in degrees, are the angles making it possible to pass from the total reference of definition of the coordinates of the nodes to a reference whose first vector indicates the direction according to which the norm of meshes surface is directed that one wishes to recover.

For the definition of the nautical angles, to see operator AFFE_CARA_ELEM [U4.42.01] operand ORIENTATION.

3.4.9.2 Operand VECT_NORMALE

♦/VECT_NORMALE = (X, there) in 2D
(X, there, Z) in 3D

coordinates X, there, Z are those giving the direction according to which the norm of meshes surface is directed that one wishes to recover.

3.4.9.3 Operand ANGL_PREC

◊ANGL_PREC =a

It is the tolerance, in degrees, that one accepts on the angle formed by the vector provided by the user and the normal vector to the surface element to affirm that these two vectors have the same direction.

The value by default of has is 0.5 degree.

3.4.9.4 Operand VERI_SIGNE

◊VERI_SIGNE = "NON",
/ "OUI', [DEFAULT]

If one "NON" assigns the value to VERI_SIGNE, the GROUP_MA will be made up by meshes surface whose norm is parallel to the vector given by the user.

If one affects the value "OUI', the GROUP_MA will be made up by meshes surface whose norm is parallel and has the same directional sense as the vector given by the user.

The default value is "OUI'.

3.4.10 Operand OPTION = "SPHERE"

/OPTION = "SPHERE"

This option makes it possible to define a GROUP_MA made up of meshes whose at least node belongs to a sphere (a circle in 2D) defined by its center and its.

3.4.10.1 Operand POINT

♦/POINT = (X, there) in 2D
(X, there, Z) in 3D

X Z are there the coordinates of the center of the sphere.

3.4.10.2 Operand /NOEUD_CENTRE /GROUP_NO_CENTRE

♦/NOEUD_CENTER= No
/ GROUP_NO_CENTRE= grno

These two keywords make it possible to indicate which is the node coinciding with the center of the sphere.

3.4.10.3 Operand RADIUS

♦RAYON = R

R is the radius of the sphere (circle in 2D).

3.4.11 Operand OPTION = "CYLINDRE"

/OPTION = "CYLINDRE"

This option makes it possible to define a GROUP_MA made up of meshes whose at least node belongs to a cylinder defined by its axis and its radius.

The axis is defined by a vector and a point pertaining to this axis. This option has meaning only in 3D.

3.4.11.1 Operand POINT

♦/POINT = (X, there, Z)

X Z are there the punctual coordinates located on the axis of the cylinder.

3.4.11.2 Operand /NOEUD_CENTRE /GROUP_NO_CENTRE

♦/NOEUD_CENTER= No
/ GROUP_NO_CENTRE= grno

These two keywords make it possible to indicate a node located on the axis of the cylinder.

3.4.11.3 Operand RADIUS

♦RAYON = R

R is the radius of the cylinder.

3.4.11.4 Operand ANGL_NAUT

♦/ANGL_NAUT = (has, b)

the nautical angles has, B defined in degrees, are the angles making it possible to pass from the total reference of definition of the coordinates of the nodes to a reference whose first vector indicates the direction of the axis of the cylinder.

For the definition of the nautical angles to see operator AFFE_CARA_ELEM [U4.42.01] operand ORIENTATION.

3.4.11.5 Operand VECT_NORMALE

♦/VECT_NORMALE = (X, there, Z)

X Z are there the coordinates of a vector directing the axis of the cylinder.

3.4.12 Operand OPTION = "TAPE"

/OPTION = "TAPE"

This option makes it possible to define a GROUP_MA made up of meshes whose at least node belongs to a "tape" defined by plane "a medium" (a line in 2D) and the half-width on both sides of this plane.

The plane is defined by a normal vector in this plane and a point belonging to him.

3.4.12.1 Operand POINT

♦/POINT = (X, there) in 2D
(X, there, Z) in 3D

X Z are there the punctual coordinates belonging to plane "medium" of the tape.

3.4.12.2 Operand / NOEUD_CENTRE /GROUP_NO_CENTRE

♦/NOEUD_CENTER= No
/ GROUP_NO_CENTRE= grno

These two keys key make it possible to define belonging to plane “a medium” of the tape.

3.4.12.3 Operand ANGL_NAUT

♦/ANGL_NAUT =aen 2D
(. B has) in 3D

the nautical angles has, B defined in degrees, are the angles making it possible to pass from the total reference of definition of the coordinates of the nodes to a reference whose first vector is orthogonal in plane “the medium” of the tape.

For the definition of the nautical angles, to see operator AFFE_CARA_ELEM [U4.42.01] operand ORIENTATION.

3.4.12.4 Operand VECT_NORMALE

♦/VECT_NORMALE = (X, there) in 2D
(X, there, Z) in 3D

X there and Z are the components of a vector perpendicular to plane “the medium” of the tape.

3.4.12.5 Operand DIST

♦DIST =d

D is the half-width of the tape.

3.4.13 Operand OPTION = “APPUI”

This option makes it possible to recover the group of meshes of which some nodes belong to all the nodes specified by the key keys NOEUD and GROUP_NO.

♦/ GROUP_NO=l_gno
. /NOEUD=l_no

These 2 keywords make it possible to define the list of the nodes which will be used as bearing with meshes.
Let us call lno1 this list.

♦TYPE_APPUI=
/“TOUT” : the mesh will be retained if **all** its nodes belong to lno1
/“SOMMET” : the mesh will be retained if **all** its nodes “ **top**” belong to lno1
/“AU_MOINS_UN” : the mesh will be retained **at least one** of its nodes belongs to lno1
/“MAJORITE” : the mesh will be retained so **more half** of its nodes belong to lno1

3.4.14 Operand OPTION = “FISS_XFEM”

This option makes it possible to recover the group of meshes of X-FEM type specified by key keys TYPE_GROUP.

♦FISSURE= (fiss1, fiss2,…)

♦TYPE_GROUP=
/“HEAVISIDE” : the mesh will be retained if it is of Heaviside type
/“CRACKTIP” : the mesh will be retained if it is of type Ace-tip
/“MIXTE” : the mesh will be retained if it is of Mixed type (Heaviside and Ace-tip)

/"XFEM" : the mesh will be retained if it is of Heaviside type, Mixed Ace-tip or
/"FISSUREE" : the mesh will be retained if **all** its nodes are nouveau riches

For a precise definition of the notions of Heaviside mesh and mesh Ace-tip, to see R7.02.12, §3.2.5.

3.5 Key word CREA_GROUP_NO

| CREA_GROUP_NO

an occurrence of this key word factor makes it possible to define a new nodes group (for key keys GROUP_MA and TOUT_GROUP_MA, one creates several nodes groups "of a blow").

3.5.1 Operand NOM

/♦NOM = gno

One gives here the name (with "quotes") of the new nodes group.

3.5.2 Operand NOEUD

/NOEUD = lnoeu

This key word makes it possible to define the nodes group in extension: one gives the list of the nodes the component.

3.5.3 Operand INTERSEC

/INTERSEC = (gno1, gno2, gno3,…)

The new nodes group will be obtained by taking all the nodes of gno1 which also belong to gno2, gno3,…. The order of the nodes remains that of gno1.

3.5.4 Operand UNION

/UNION = (gno1, gno2, gno3,…)

The new nodes group will be obtained by taking all the nodes of gno1, then by adding the nodes of gno2 which do not belong to gno1, then those of gno3 which belong neither to gno1 nor with gno2, etc

3.5.5 Operand DIFFE

/DIFFE = (gno1, gno2, gno3,…)

The new nodes group will be obtained by taking all the nodes of gno1 which do not belong to the other groups of the list. The order of the nodes remains that of gno1.

3.5.6 Under group of an existing group: key words GROUP_NO / POSITION / NUME_INIT / NUME_FIN

One can some nodes create a new group of node by selecting existing group.

1st possibility:

One creates a group **of only one node** by specifying by key word POSITION the required node.

Example:

```
CREA_GROUP_NO = _F (GROUP_NO = G1,    POSITION = "INIT",  NOM = G1I)
```

group G1I contains the 1st node of the G1 group.

2nd possibility:

One creates a group containing the nodes ranging between the rows nuini and nufin (included) in an existing group.

Example:

```
   CREA_GROUP_NO=_F (GROUP_NO = G1,  NUME_INIT = 3NUMÉRIQUE_FIN    = 7, NOM
= G1P)
```

group G1P contains node 3,4,5,…, 7 of G1.

Caution:

*These keywords use the notion **of order** of the nodes in a nodes group. This order is often unknown to the user. It can depend on the preprocessor. It is the order of the nodes at the time of the definition of the GROUP_NO in the Aster mesh file.*

3.5.7 Operand OPTION = "ENV_SPHERE"

/OPTION = "ENV_SPHERE"

This option makes it possible to define a GROUP_NO made up of the nodes located on the envelope of a sphere except for an accuracy given.

3.5.7.1 Operand POINT

♦/POINT = (X, there) , in 2D
(X, there, Z), in 3D

X Z are there the coordinates of the center of the sphere.

3.5.7.2 Operand /NOEUD_CENTRE /GROUP_NO_CENTRE

♦/NOEUD_CENTER= No
/ GROUP_NO_CENTRE= grno

These two keywords make it possible to define the node coinciding with the center of the sphere.

3.5.7.3 Operand RADIUS

♦RAYON =r

R is the radius of the sphere.

3.5.7.4 Operand accuracy

◊PRECISION =eps

eps is the tolerance with which one the belonging defines one node in the envelope of the sphere. This tolerance is to be taken with the following meaning:

if d is the distance from a node in the center of the sphere, it is said that this node belongs to the group if:

$$|d-r| \leq \text{eps}$$

3.5.8 Operand OPTION = "ENV_CYLINDRE"

/OPTION = ' ENV_CYLINDRE'

This option makes it possible to define a GROUP_NO made up of nodes located on the envelope of a cylinder except for an accuracy given.

This option has meaning only in 3D.

3.5.8.1 Operand POINT

♦/POINT = (X, there, Z)

X Z are there the punctual coordinates pertaining to the axis of the cylinder.

3.5.8.2 Operand /NOEUD_CENTRE /GROUP_NO_CENTRE

♦/NOEUD_CENTER= No
/ GROUP_NO_CENTRE= grno

These two keywords make it possible to define a node pertaining to the axis of the cylinder.

3.5.8.3 Operand RADIUS

♦RAYON =r

R is the radius of the cylinder.

3.5.8.4 Operand ANGL_NAUT

♦/ANGL_NAUT = (has . b)

the nautical angles has, B defined in degrees, are the angles making it possible to pass from the total reference of definition of the coordinates of the nodes to a reference whose first vector indicates the direction of the axis of the cylinder.

For the definition of the nautical angles, to see operator AFFE_CARA_ELEM [U4.42.01] operand ORIENTATION.

3.5.8.5 Operand VECT_NORMALE

♦/VECT_NORMALE = (X, there, Z)

X Z are there the coordinates of a vector directing the axis of the cylinder.

3.5.8.6 Operand accuracy

◊PRECISION =eps

eps is the tolerance with which one the belonging defines one node in the cylinder clothing.

This tolerance is to be taken with the following meaning:

if d the distance from the point indicates running to the axis of the cylinder, it is said that the point running belongs to the cylinder clothing if:

$$|d-r| \leq \text{eps}$$

3.5.9 Operand OPTION = "PLANE"

This option makes it possible to define a GROUP_NO made up of nodes located on a line (in 2D) or in a plane (in 3D) except for an accuracy given.

3.5.9.1 Operand POINT

♦/POINT = (X, there) , in 2D
(X, there, Z), in 3D

X Z are there the punctual coordinates belonging to plane (with the right).

3.5.9.2 Operand /NOEUD_CENTRE /GROUP_NO_CENTRE

♦/NOEUD_CENTER= No
/ GROUP_NO_CENTRE= grno

These 2 keywords make it possible to define a node belonging to plane (with the right).

3.5.9.3 Operand ANGL_NAUT

♦/ANGL_NAUT =a , in 2D

(has , b), in 3D

the nautical angles has, B defined in degrees, are the angles making it possible to pass from the total reference of definition of the coordinates of the nodes to a reference whose first vector is orthogonal in plane "the medium" of the tape.

For the definition of the nautical angles, to see operator AFFE_CARA_ELEM [U4.42.01] operand ORIENTATION.

3.5.9.4 Operand VECT_NORMALE

♦/VECT_NORMALE = (X, there) , in 2D
(X, there, Z), in 3D

X there and Z are the components of a vector perpendicular to the plane (with the right).

3.5.9.5 Operand accuracy

◊PRECISION =eps

eps is the tolerance with which one the belonging defines node in the plane (or with the right).

This tolerance is to be taken with the following meaning:

if d indicates the distance from the node to the plane (or the right), it is said that this node belongs to this plane (or on this line) if:

$$|d| \leq \text{eps}$$

3.5.10 Operand OPTION = "SEGM_DROI_ORDO"

This option is used to order a set of nodes roughly located on a line segment AB.

♦/NOEUD= lno2,
/GROUP_NO = gno2,

One defines all the nodes which one wants to order.

♦/NOEUD_ORIG= noA , ♦/NOEUD_EXTR= noB ,
/GROUP_NO_ORIG = gnoA , /GROUP_NO_EXTR= gnoB ,

One defines the nodes A and B, origin and end of the segment AB.

♦PRECISION =prec ,
♦CRITERE = "RELATIF" ,
/ "ABSOLU" ,

These two arguments are parapets, they are used to check that the nodes which one seeks to order (lno2 or gno2) are well on the segment AB. If the variation of a node with AB is higher than prec the code stops in fatal error.
If the selected criterion is "RELATIF", the distance from a node with AB will be divided by the length from AB.

3.5.11 Operand OPTION = "NOEUD_ORDO"

This option is used to create an ordered **group_no** containing the nodes of a set of meshes formed by segments (SEG2, SEG3 or SEG4). The group of these meshes must form line continuous. Line can be "open" (with 2 ends) or "closed" (it is then a simple loop).

♦GROUP_MA =gmaAB

Name of the group_ma which one wants to order the nodes.
Meshes of gmaAB must form line open.

◊/NOEUD_ORIG= noA , ◊/NOEUD_EXTR= noB ,
/GROUP_NO_ORIG = gnoA , /GROUP_NO_EXTR= gnoB ,

the keywords make it possible to define the nodes A and B, origin and end of line AB.

The node A will be numbered in first, then one makes use of the topology of meshes of gmaAB to number the nodes gradually.

If the node A is not provided by the user, the program will choose like node “origin”, the first node of gmaAB which belongs only to only one mesh segment. The origin is thus arbitrary: the program could just as easily have fallen on the other end.

It is checked that the last numbered node is well B (if this one is given).

3.5.11.1 Case of the closed lines

If line is a loop, one cannot determine its ends automatically. To define the origin of the curvilinear abscisses, the user **is obliged** to inform the nodes origin and end. It is necessary that these 2 nodes are identical.

To direct one line closed, one cannot make use of the knowledge of the node origin (since it is identical to ending node). If it wishes it, the user can then inform key word VECT_ORIE (2 or 3 cordonnées according to the dimension of space). One will choose as meaning of path of the loop, the mesh of gmaAB which touches the node origin and which minimizes the angle with the vector provides by VECT_ORIE.

3.5.12 Operand OPTION = "TUNNEL"

This option is used to create the formed group_no by the nodes located inside a "tunnel" which one provides the axis and the radius. The nodes selected will be those whose distance to the axis is lower than the radius.

The axis of the "tunnel" is defined by the meshes linear ones provided via key keys MAILLE_AXE and GROUP_MA_AXE.

The axis of the tunnel must have a "origin" defined by key keys NOEUD_ORIG and GROUP_NO_ORIG.

The key word RADIUS is used to define the "radius" of the tunnel.

One can limit the tunnel by giving his length by key word LONGUEUR. This length is measured from the origin of the tunnel.

The nodes candidates to be part of the tunnel those carried by are meshes defined by the keywords: TOUT=' OUI', GROUP_MA and MESH.

3.5.13 Operand OPTION = "INCLUSION"

```
/OPTION       =  ' INCLUSION',
    ♦GROUP_MA      = lgma  ,          [l_group_ma]
    ♦CAS_FIGURE       =    /"2D"
                           /"3D"
                           /"2.5D"
    ◊DISTANCE_MAX = distma
    ♦GROUP_MA_INCL    = lgma_inc  ,   [l_group_ma]
    ◊MAILLAGE_INCL    = ma_inc ,      [mesh]
```

This option makes it possible to create the group of the nodes of meshes of lgma which are geometrically inside meshes of lgma_inc.
If MAILLAGE_INCL is not provided, lgma_inc is a list of GROUP_MA of the mesh which one enriches (my). If they are not GROUP_MA of ma_inc.

Key word CAS_FIGURE is compulsory, it is used to determine which are meshes lgma_inc which must be used to determine inclusion:
"2D" : one is interested only in meshes surface (SORTED and QUAD) a 2D mesh (plane XOY).
"3D" : one is interested only in meshes voluminal (TETRA, PENTA,...)
"2.5D" : one is interested only in meshes surface (SORTED and QUAD) of a mesh 3D (shell).

Key word DISTANCE_MAX is optional. It is used to give a small tolerance to determine whether a node is included in a mesh. Indeed, a node located "just" on an interface between 2 meshes, risk to be regarded as "outside" to the 2 meshes and thus like not being part of inclusion. This is why a value by default of distma is taken by the code. One chose 1% length of the smallest edge of the mesh ma_inc.
In the case of a mesh of type "shell" (2.5D), if surface is not plane, it is almost impossible that an element of facet is geometrically included in others meshes: there is almost always a variation in the "normal" direction on the surface. One will thus need in general, in this case, supply a value of distma higher than the value by default.

3.5.14 Operand OPTION = "INTERVALLE_VALE"

This option is used to create the formed group_no by the nodes whose value of a component (cmp) of a field at nodes (cham_no) lies between two values (vmin and vmax).

The field and the component which will be used to select the nodes are given by key keys CHAM_GD and NOM_CMP.

The values vmin and vmax are provided via the key word VALE.

Example:
```
DEFI_GROUP (reuse  = MAIL, MAILLAGE = MAIL,
            CREA_GROUP_NO = _F (NOM=' GN700', OPTION=' INTERVALLE_VALE',
            CHAM_GD=TEMPER, NOM_CMP=' TEMP', VALE= (700. , 800.),),);
```

GROUP_NO "GN700" will be made of all the nodes from the mesh MAIL of which the temperature in field TEMPER lies between 700. and 800.

3.5.15 Operand OPTION = "FISS_XFEM"

This option makes it possible to recover the group of the nodes of the type XFEM specified by key keys TYPE_GROUP.

♦FISSURE= (fiss1, fiss2,…)

♦TYPE_GROUP=
/"XFEM" : the node will be retained if it is an enriched node
/"HEAVISIDE" : the node will be retained if it is a node enriched by Heaviside type
/"CRACKTIP" : the node will be retained if it is a node enriched by Cracktip type
/"MIXTE" : the node will be retained if it is a node enriched by Mixed type (Heaviside and Cracktip)
/"ZONE_MAJ" : the node will be retained if it is contained in update zone of level sets. If the mesh of crack passed by the key word MAILLAGE (§ 9) , update zone coincides:
•in the absence of one auxiliary grid associated with crack, with the field of computation around the bottom,
•in the presence of one auxiliary grid associated with crack, with the field of projection between grid and mesh , independently of the field of computation used on the grid.
If a grid passed by the key word GRILL (§ 9) , update zone always coincides with the field of computation used on the grid.

/"TORE" : the node will be retained if it is contained in a torus built around the crack tip of radius given by the key word RAYON_TORE . If the localization of the field were used for the computation of crack, this option cannot be selected. In this case, the nodes group is created by means of update zone (one selects option TYPE_GROUP=' ZONE_MAJ' automatically) and the choice of the user is ignored.

For a definition specifies notions of enriched node, Heaviside node, node Ace-tip, auxiliary grid, field of computation and its localization, to see R7.02.12, §3.2.5.

3.5.16 Operand RAYON_TORE

If TYPE_GROUP=' TORE'
♦ RAYON_TORE = R,

One specifies the radius R of the torus to be used for the selection of the nodes.

3.5.17 Operands GROUP_MA and NOM

/GROUP_MA = lgma

For each mesh group of the list lgma, one creates a formed nodes group by the nodes carried by meshes of this mesh group.

◊NOM = lgno

If lgno is provided by the user, this list must be of the same length than lgma. These are the names that one wants to give to the new nodes groups.

If lgno is not provided, the nodes groups will bear the same names as the mesh groups which gave them rise.

◊CRIT_NOEUD =
/"TOUS" [DEFAULT] : all the nodes of each mesh are taken.
/"SOMMET" : one takes only the nodes "top" of meshes (i.e. ends of the edges).
/"MILIEU" : one takes only the nodes "medium" of the edges of meshes.
/"CENTER" : one takes only the nodes which is neither "top" nor "medium" it is - with - to say the nodes to the center of the facets or the voluminal elements.

3.5.18 Operand TOUT_GROUP_MA

/TOUT_GROUP_MA = "OUI'

This key word has the same meaning as the precedent, except that one creates nodes groups for **all** the existing mesh groups of the mesh.

3.5.19 Operand ALARME = "OUI' [DEFAULT] /"NON"

if ALARME = "NON", the code does not emit alarm; for example when one asks him to create a GROUP_NO and that this group is empty. The default value of this key word is "OUI'.

3.5.20 Operand INFO

if INFO = 1, one prints in the file "MESSAGE", the number of groups create and for each group, the name of the group and the number of entities the component.

if INFO = 2, one prints in the file "MESSAGE", the number of groups create and for each group, the name of the group, the number of entities the component then the list of the entities setting up the groups.

4 Examples

Example 1 (topological criteria and logics):

That is to say a my mesh containing the mesh groups already:

```
M1   m2   m3
```

and the nodes groups:

```
N1   N2   N3
```

```
my = DEFI_GROUP (reuse = my,  MAILLAGE = my,
           CREA_GROUP_MA = ( _F   (  NOM = NM1,  MESH = (MA7, MA9,…)  ),
                             _F   (  NOM = NM2,  UNION =    (M1, NM1)),
                             _F   (  NOM = Nm3,  DIFFE =    (NM2, m2)),),
           CREA_GROUP_NO = _F     (  TOUT_GROUP_MA = "OUI'),                 )

my = DEFI_GROUP (reuse = my, MAILLAGE = my,
           CREA_GROUP_MA = _F     (  NOM = NM4,  MESH = (MA7, MA11, MA13))
           CREA_GROUP_NO = (_F    (  NOM = NN1,  INTERSEC=  (NM1, N1)),
                            _F (   GROUP_MA = NM4)))
```

After these two calls to command DEFI_GROUP, the mesh contains then:

- mesh groups:
  - M1, m2, m3 (initial)
  - NM1 = (meshes: MA7, MA9,…)
  - NM2 = M1 "union" NM1
  - Nm3 = NM2 "minus" m2
  - NM4 = (MESHES: MA7, MA11, MA13)
- nodes groups:
  - N1, N2, N3 (initial)
  - M1, m2, m3, NM1, NM2, Nm3 : group_no containing the nodes of the group_ma of **same names**. These group_no is created by 1st command DEFI_GROUP.
  - NN1 = NM1 "intersection" N1
  - NM4 = (nodes of group_ma NM4)

Example 2 (geometrical criteria):

```
my = DEFI_GROUP    (reuse = my, MAILLAGE = my,
           CREA_GROUP_MA=  (_F    (NOM = facesup   ,    OPTION =   ' FACE_NORMALE',
                                  VECT_NORMALE = (0. ,  0. ,  1.)),
                           _F   (NOM = S01    ,        OPTION =   ' SPHERE',
                                  POINT = (0. ,  0. ,  0.), RADIUS = 1.),),
           CREA_GROUP_NO = (_F    (NOM = BO_S01 ,    OPTION =   ' ENV_SPHERE',
                                  POINT= (0. , 0. , 0.), RAYON=1., PRECISION=0.01),
                          _F   (NOM = S01_1  ,    GROUP_MA =S01 ),
                          _F   (NOM = S01_2  ,    OPTION = "ENV_SPHERE",
                                  POINT= (0. , 0. , 0.), RAYON=0.5,
PRECISION=0.5),),
                    )
```

After DEFI_GROUP the mesh my will contain 2 new GROUP_MA and 3 new GROUP_NO:

- facesup contains the facets whose norm is directed according to OZ (towards $Z>0$),
- S01 meshes **contains** all of which **one of the nodes** belongs to the sphere of radius 1. and centered in O (origin of the axes),
- B0_S01 is the group of the nodes which are in the vicinity of the envelope of the preceding sphere (S01),
- S01_1 is the group of all the nodes of meshes of the S01 mesh group ; caution: some nodes of this group can be outside the sphere!
- S01_2 is the group of the nodes included in the S01 sphere : $|d(M,O)-0.5|\leq 0.5$