

---

## Description of the file of grid of Code\_Aster

---

### Summary:

This file makes it possible to describe a grid of *Code\_Aster*. It can be built by an automatic interface between a maillor and the code, but can also be written with the hand. It contains:

- a possible title,
- a list of nodes with their coordinates,
- lists of meshes, each mesh is described by the list of the nodes which defines its topology, and its orientation,
- groups of nodes and groups of meshes.

### Note:

*A concept of the type `grid`, once read on the file of grid, can be enriched using operators by Code\_Aster by information which cannot be currently described in the file of grid. This relates to in particular the static under-structuring. It is of more possible, now, to create groups of nodes or meshes under study.*

## Tables of content

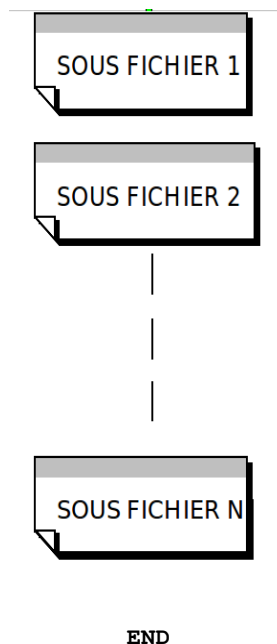
---

1 Structure of the file of grid.....	3
2 Structure of under file.....	4
2.1 The heading.....	5
2.2 List of information.....	5
2.3 END of subfile.....	5
3 List of the keywords (standard of under files).....	6
4 The description of the meshes.....	7
5 List of the formats of description.....	13
5.1 Correspondence Item - Standard FORTRAN.....	13
5.2 FORMAT of reading associated with the keyword.....	13
5.3 Repetition of the Format.....	13
6 Examples of under file.....	15
6.1 File resulting from interface IDEAS-Aster.....	15
6.2 Example illustrating the possible formats of writing.....	15
6.3 Example illustrating the assignment by groups made up.....	16

## 1 Structure of the file of grid

The file of grid *Aster* first line until the first occurrence of a line is read begin with the word **END**. This keyword is obligatory.

The file of grid *Aster* consists of a succession of under files of grid, independent.



The file of **grid** *Aster* is a file with format, within the meaning of FORTRAN:

- the length of line is limited to 80 characters (any information starting from column 81 is simply ignored);
- white character ' ' and the comma', 'are only the recognized separators,
- the character % is a character reserved to indicate the beginning, until the end of the line, of a comment,
- the numerical data are interpreted in free format:

1 = 1. = 1.0 = 1.E+0 = .1E1 = 10.D-1

The file **grid** must contain at least:

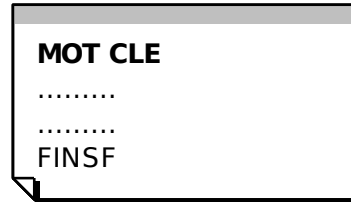
- a subfile of coordinates of nodes `COOR_2D` or `COOR_3D`, defined by 2 or 3 real numbers in a single orthonormal Cartesian reference mark.
- a subfile by type of mesh used.

It can contain, moreover, and that is recommended, a subfile `TITLE`. It is practically essential to define, at the time of the grid, the subfiles of groups of meshes `GROUP_MA` and of the subfiles of groups of nodes `GROUP_NO`.

It will be retained that the assignment of a kind of finite element, can be done only on one mesh (or exceptionally on a node for the discrete elements) described in the file grid. The loadings or the boundary conditions are affected on meshes or nodes according to the cases.

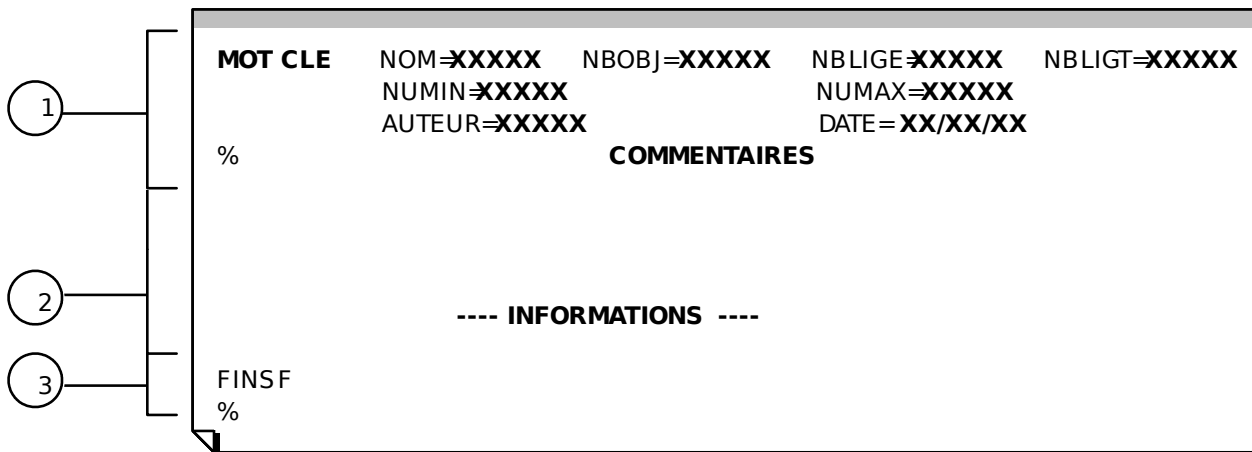
## 2 Structure of under file

Under file starts with one **keyword** and is finished by the imposed keyword **FINSF** :



This description is sufficient for Aster. The most complete description is the following one:

- each under file comprises 3 parts:
  - 1) the heading of under file,
  - 2) contents of information,
  - 3) the end mark of under file (**FINSF**).



- **WORD KEY** and **FINSF** (end of under file) are obligatory because they delimit under file inside the file of grid.
- **NAME** is optional. When the keyword is **GROUP\_MA** or **GROUP\_NO**, if **NAME** is absent then the first channel of character met after these **KEYWORDS** it will form **NAME GROUP\_MA** or of **GROUP\_NO**.

By convention any character string which follows the characters % is interpreted like a comment.

## 2.1 The heading

The role of the heading is to provide total information on the contents of under file:

**KEYWORD** : besides its role of delimiter of under file it informs about the nature of information which will follow. It must be in 1st line of under file **and tallied on the left**.

**NAME** : it is used to indicate a set of information. It must be in 1st line of the heading of under file. It is optional.

all the other keywords are currently ignored by the operator `LIRE_MAILLAGE` [U4.21.01].

## 2.2 List of information

They are the information resulting from a maillor and written in free format.

Information which this file must contain defines of the classical entities of the finite element method:

- **nodes** : points defined by one **name** and by theirs **Cartesian coordinates** in space **2D** or **3D**,
- **meshs** : topological figures **named** plane or voluminal (not, segment, triangle, quadrangle, tetrahedron,...), defined by the ordered list of the nodes which constitute them and to which will be able to apply various types of finite elements, boundary conditions or loadings. The order of the nodes directs the mesh. This orientation must be scrupulously established because it is in particular used to lay down the direction of the loadings (pressure,...).

One can also define groups of these topological entities.

- **groups of nodes** : named lists of names of nodes,
- **groups of meshs** : named lists of names of meshs.

Systematically, when one has to refer to nodes (or meshs), one will be able to refer to groups of nodes (or groups of meshs).

The interfaces with the maillors generate such groups automatically (according to criteria specific to the maillor).

These groups facilitate the assignment of the material characteristics on the grid, of the boundary conditions or the impression of the results. Indeed, at the time of such operations one refers to the nodes or the meshs, which it is interesting to describe by group.

The operator of *Code\_Aster* `DEFI_GROUP` [U4.22.01] allows to create groups a posteriori on any concept of the type `grid` already read.

It will be noted, that all the entities handled in the code (nodes, meshs, groups of nodes, groups of meshs) are **named** and usable constantly by their name (**8 characters** to the maximum for the meshs and the nodes, **24 characters** maximum for `GROUP_MA` and `GROUP_NO`).

The "breakage" of the names is preserved: for example, `'GR_AB'` is different from `'gr_ab'`. The classification of the entities is never clarified: it is used for only in-house to point on the values of the various associated variables.

## 2.3 END of subfile

It is reserved' and obligatory word the `'FINSE'` who indicates the end of under file, which begin with one `WORD KEY`.

## 3 List of the keywords (standard of under files)

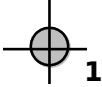
TITLE	Description of the file grid in documentary matter. The operators of interface with mailleurs build a specific title starting from information to each mailleur
/ COOR_3D	Description of the nodes by three space coordinates: <ul style="list-style-type: none"><li>• name of the node described (or number interpreted like a name),</li><li>• 3 coordinates,</li></ul>
/ COOR_2D	Description of the nodes by 2 coordinates : <ul style="list-style-type: none"><li>• name of the node described (or number interpreted like a name),</li><li>• 2 coordinates,</li></ul>
/ POI1	points )
/ SEG2	/ SEG3 / SEG4 segments )
/ TRIA3	/ TRIA6 / TRIA7 triangles )
/ QUAD4	/ QUAD8 / QUAD9 quadrangles )
/ HEXA8	/ HEXA20 / HEXA27 hexahedrons )
/ PENTA6	/ PENTA15 / PENTA18 pentahedrons )
/ TETRA4	/ TETRA10 tetrahedrons )
/ PYRAM5	/ PYRAM13 pyramids )
	<b>connectivity</b>
	<b>meshs</b>
GROUP_MA	Group of meshes (attention the name is obligatory)
GROUP_NO	Group of nodes (attention the name is obligatory)


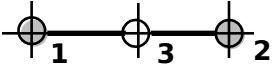
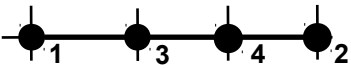
### Note:

- When a keyword was read, there remains interpreted until the appearance of *FINSF*.
- After one *FINSF*, any keyword not recognized (typing error) is ignored and a message is transmitted.
- *COOR\_2D* and *COOR\_3D* are mutually exclusive.
- The presence of a frame of reference in universal file IDEAS causes a stop of the IDEAS/Aster interface.

## 4 The description of the meshes

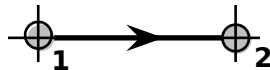
Conventions of description of the topology of the meshes in *Aster* are summarized here. The use of the meshes is given here as an indication, it is necessary to refer to specific documentation to each order.

Mesh associated with a <b>node</b> (2D or 3D)	
POI1	

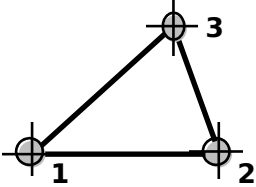
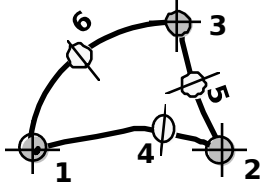
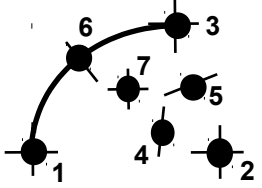
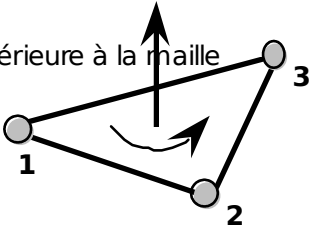
Mesh associated with a <b>segment</b> (2D or 3D)	
SEG2	
SEG3	
SEG4	

### Orientation :

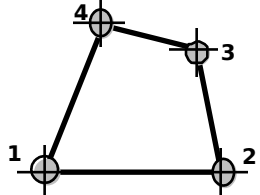
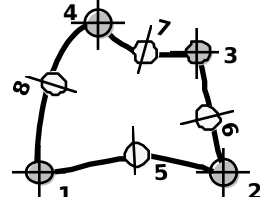
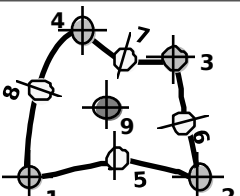
Down the direction is laid by the order of the nodes tops



This orientation is possibly supplemented by characteristics given according to the type of affected finite element on the mesh by the keyword factor `ORIENTATION` of the operator `AFFE_CARA_ELEM` [U4.42.01].

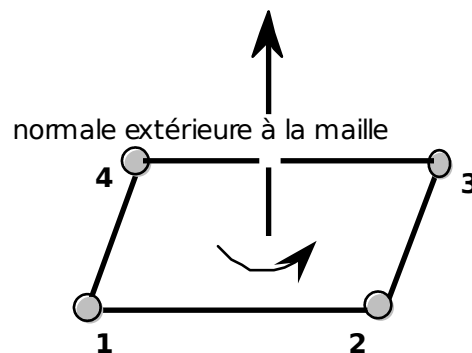
Mesh associated with a triangle (2D or 3D)	
TRIA3	
TRIA6	
TRIA7	
<p><b>Orientation:</b></p> <p>Down the direction is laid by the order of the nodes tops which are given to define the direct trigonometrical direction.</p> <div style="text-align: center;">  </div> <p>This orientation is possibly supplemented by characteristics given to the elements by the operator <code>AFFE_CARA_ELEM</code> [U4.42.01], in particular for the elements of hull by the keyword factor <code>HULL</code>.</p>	



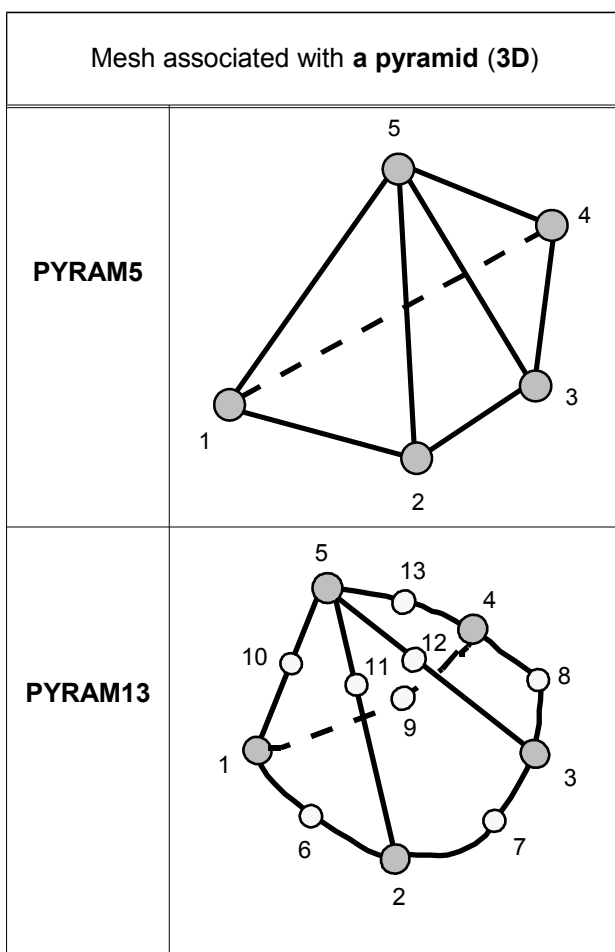
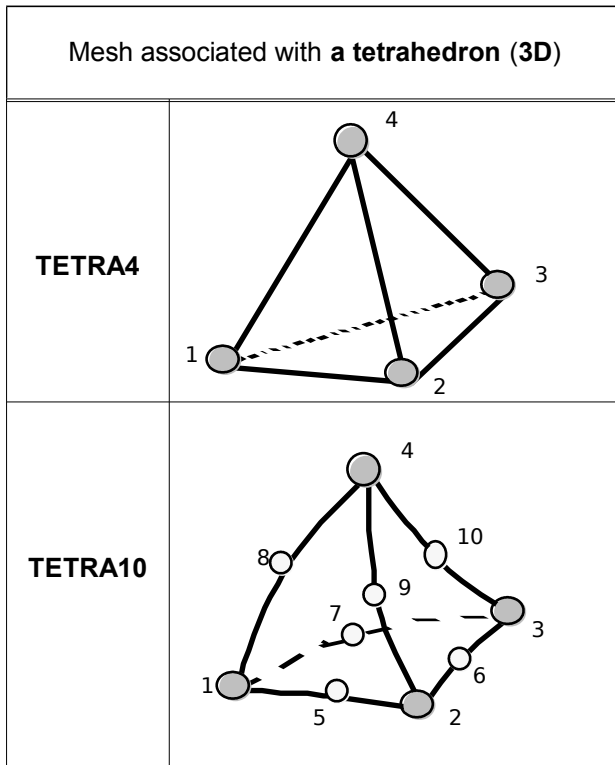
Mesh associated with a quadrangle (2D or 3D)	
QUAD4	
QUAD8	
QUAD9	

### Orientation:

Down the direction is laid by the order of the nodes tops which are given to define the direct trigonometrical direction



This orientation is possibly supplemented by characteristics given to the elements by the operator `AFFE_CARA_ELEM` [U4.42.01], in particular for the elements of hull by the keyword factor `HULL`.



Mesh associated with a pentahedron (3D)	
<b>PENTA6</b>	
<b>PENTA15</b>	
<b>PENTA18</b>	

Mesh associated with a hexahedron (3D)	
<b>HEXA8</b>	
<b>HEXA20</b>	
<b>HEXA27</b>	

## 5 List of the formats of description

We will describe the format of description of information by a continuation of items.

### 5.1 Correspondence Item - Standard FORTRAN

```
NOM_DE_NOEUD
  CHARACTER*8
NOM_DE_MAILLE
  CHARACTER*8
Coordinate
  REAL*8
LIGNE_DE_TEXTE
  CHARACTER*80
```

### 5.2 FORMAT of reading associated with the keyword

#### FORMATS recognized by Aster

#### keyword

(1*NOM_DE_NOEUD, 3*COORDONNÉE)	COOR_3D
(1*NOM_DE_NOEUD, 2*COORDONNÉE)	COOR_2D
(1*NOM_DE_MAILLE, 1*NOM_DE_NOEUD)	POI1
(1*NOM_DE_MAILLE, 2*NOM_DE_NOEUD)	SEG2
(1*NOM_DE_MAILLE, 3*NOM_DE_NOEUD)	SEG3, TRIA3
(1*NOM_DE_MAILLE, 4*NOM_DE_NOEUD)	QUAD4, TETRA4, SEG4
(1*NOM_DE_MAILLE, 5*NOM_DE_NOEUD)	PYRAM5
(1*NOM_DE_MAILLE, 6*NOM_DE_NOEUD)	TRIA6, PENTA6
(1*NOM_DE_MAILLE, 7*NOM_DE_NOEUD)	TRIA7
(1*NOM_DE_MAILLE, 8*NOM_DE_NOEUD)	QUAD8, HEXA8
(1*NOM_DE_MAILLE, 9*NOM_DE_NOEUD)	QUAD9
(1*NOM_DE_MAILLE, 10*NOM_DE_NOEUD)	TETRA10
(1*NOM_DE_MAILLE, 13*NOM_DE_NOEUD)	PYRAM13
(1*NOM_DE_MAILLE, 15*NOM_DE_NOEUD)	PENTA15
(1*NOM_DE_MAILLE, 18*NOM_DE_NOEUD)	PENTA18
(1*NOM_DE_MAILLE, 20*NOM_DE_NOEUD)	HEXA20
(1*NOM_DE_MAILLE, 27*NOM_DE_NOEUD)	HEXA27
(1*NOM_DE_NOEUD)	GROUP_NO
(1*NOM_DE_MAILLE)	GROUP_MA
(LIGNE_DE_TEXTE)	TITLE

### 5.3 Repetition of the Format

- Any repetition of the format must start at the beginning of line.

Examples:

Is **ILLICIT** :

```
COOR_2D
N1 1.      2.      N2 1.
3.
FINSF
```

Is **LICIT** :

```
COOR_2D
N1 1.
2.
N2 1.
```

3.  
FINSE

## 6 Examples of under file

Examples of under files describing the space coordinates of 5 nodes expressed in the basic Cartesian total reference mark. Here the contents of information is the same one for the 2 pennies files, it is the mode of representation which differs.

### 6.1 File resulting from interface IDEAS-Aster

```
COOR_3D  NOM=MAILLAGE1      NBOBJ=5      NBLIGE=5      NBLIGT=12
          NUMIN=4           NUMAX=14
          AUTEUR=INTERF_ST/TF      DATE= 11/27/89
%        XMAX=10.0        YMAX=20.0        ZMAX=0.0
%        XMIN=0.0        YMIN=0.0        ZMIN=0.0

%FORMAT= (1*NOM_DE_NOEUD, 3*COORDONNÉE)
NO4      0.0      0.0      0.0
NO7      5.0      0.0      0.0
NO8      5.0      5.0      0.0
NO10     10.0     20.0     0.0
NO14     5.0      10.0     0.0

FINSF
```

### 6.2 Example illustrating the possible formats of writing

```
TITLE          % it is the title
THAT IT IS BEAUTIFUL MY FILE GRID  % self-satisfaction
FINSF

COOR_2D        % coordinated 2D
               % presentation sells by auction but not very advisable
NOEUD1 0      0.0
NOEUD2 1      1.D+0
NOEUD3 2.213564 2.32E+00

FINSF

SEG2          % some meshes
             % which suits me
MAILLE1 NOEUD1 NOEUD2
MAILLE3 NOEUD2 NOEUD3

FINSF

POI1
MAILLE2 NOEUD2

FINSF

GROUP_MA     % groups meshes
GROUP1 MAILLE1 MAILLE3

FINSF

GROUP_NO     NAME = GROUP2
             NOEUD1 NOEUD2

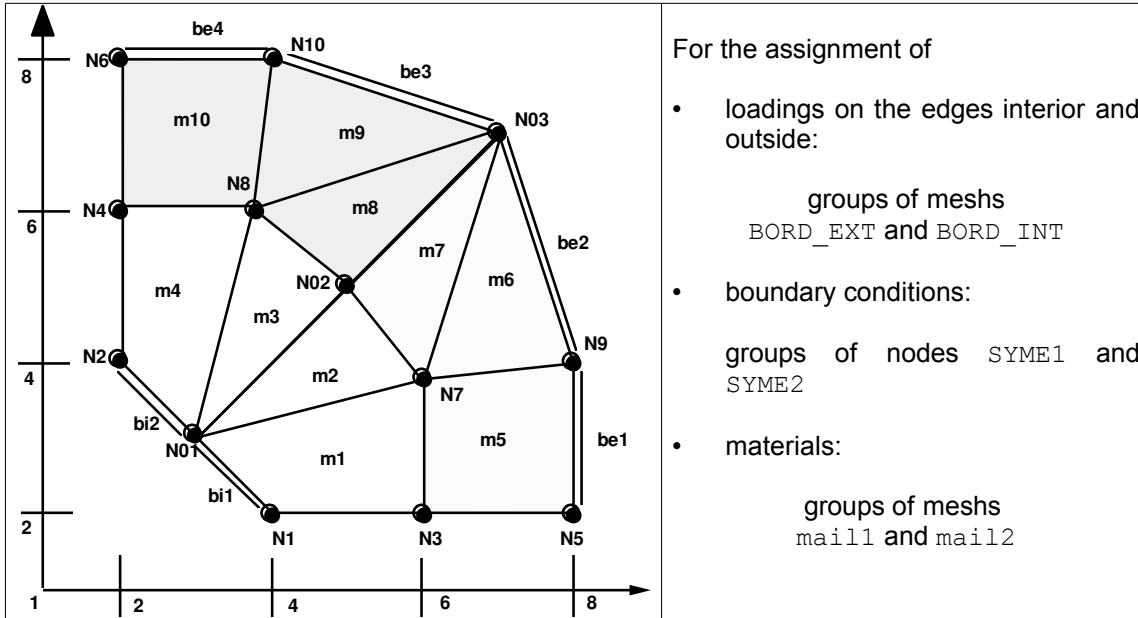
FINSF

END          % obligatory
```

## 6.3 Example illustrating the assignment by groups made up

This example is, obviously, schematic to describe the principal rules of assignment.

**Grid 2D** : a quarter of a plan field



For the assignment of

- loadings on the edges interior and outside:

groups of meshes  
BORD\_EXT and BORD\_INT

- boundary conditions:

groups of nodes SYME1 and SYME2

- materials:

groups of meshes  
mail1 and mail2

COOR\_2D

```
N1 4. 2.
N2 2. 4.
N3 6. 2.
N4 2. 6.
N5 8. 2.
N6 2. 8.
N7 6. 3.8
N8 3.8 6.
N9 8. 4.
N10 4. 8.
N01 3. 3.
N02 5. 5.
N03 7. 7.
```

FINSF

GROUP\_MA name = **BORD\_INT**  
bi1 bi2

FINSF

GROUP\_MA name = **BORD\_EXT**  
be1 be2 be3 be4

FINSF

GROUP\_NO name = **SYME1**

N1 N3 N5

FINSF

GROUP\_NO name = **SYME2**

N2 N4 N6

FINSF

GROUP\_MA name = **mail1**

m5 m6 m7

FINSF

GROUP\_MA name = **mail2**

m8 m9 m10

FINSF

TRIA3

```
m2 N01 N7 N02
m3 N01 N02 N8
m6 N7 N9 N03
m7 N7 N03 N02
m8 N8 N02 N03
m9 N8 N03 N10
```

FINSF

QUAD4

```
m1 N1 N3 N7 N01
m5 N3 N5 N9 N7
m4 N2 N01 N8 N4
m10 N4 N8 N10 N6
```

FINSF

SEG2 name = **B\_INT**

```
bi1 N1 N01
bi2 N01 N2
```

FINSF

SEG2 name = **B\_EXT**

```
be1 N5 N9
be2 N9 N03
be3 N03 N10
be4 N10 N6
```

FINSF

END



# Code\_Aster

Version  
default

---

Titre : Description du fichier de maillage de Code\_Aster  
Responsable : PELLET Jacques

Date : 06/03/2016 Page : 17/17  
Clé : U3.01.00 Révision :  
19275b5b3a91