Interface of the file of grid GIBI with *Code_Aster*

Summary:

The use of GIBI (maillor of CASTEM 2000) can be done on workstation Unix/Linux according to the versions available.

One describes here the principle of the interface which makes it possible to use in *Code_Aster* a file resulting from CASTEM 2000.

This interface is put in work in *Aster* by the order PRE_GIBI [U7.01.11].

1 Use of GIBI

CASTEM 2000 is a code of analysis of the structures by finite elements, which has a set of features dedicated to the grid in 2D and 3D. The whole of these functions of grid is known under name GIBI.

In the continuation of this document, one will name GIBI, the whole of the functions of grid as well as the whole of the orders for visualization and the analysis of the results.

GIBI is available on a certain number of platforms Unix/Linux. It is a version diffused by EDF R & D /AMA within the framework of a contract with the ECA, it is available in download on the site http://www.code-aster.org.

The memory allocated by GIBI is of 20 megawords by default. For certain applications it is necessary to increase this size memory. With this intention, the variable should be defined ESOPE_PARAM in the following way:

```
setenv ESOPE_PARAM 'ESOPE=x000000, NTRK=1024, LTRK=1048576'
```

or according to Shell used:

```
export ESOPE_PARAM=' ESOPE=x000000, NTRK=1024, LTRK=1048576'
```

where X is the number of million words which one wishes.

The ECA produces a new version of GIBI each year. Before being commissioned on the centralized waiter and to be diffused, this version is tested by the team codes on the whole of the tests *Aster*. Thus the “standard” version is currently GIBI version 2000, launched in interactive by the order gibi2000.x.

2 Documentation on the use of GIBI

The documentation of GIBI can be obtained by carrying out GIBI with the data file according to:

```
NOTE;
END;
```

One recovers then in the listing of work, the list of the chapters corresponding to all the operators of GIBI.

GIBI has a documentation in line which makes it possible to have the syntax of each operator by the order:

```
INFORMATION name of the order;
```

but also the possibility of knowing the whole of the operators dedicated to the grid, gathered by features (operators of creation of points, linear entities, surface entities and voluminal entities…) by the order:

```
INFORMATION DEBU;
```

This information is also available since a browser Internet (the site depends on the local installation).

3 The Council of use of GIBI for *Aster*

- Not to forget that each order ends in one "; ",
- *Code_Aster* can read again the files created by GIBI having level 3,4,5,6,8,9,10,11 or 13. This level is related to the change of format,
- to think that it is not possible to impose the passage of the grid by a point or a line not being reproduced on external contour. To envisage the geometry division consequently,
- to think that GIBI directs with its own conventions the various meshs. Elements can be thus "turned over" (from where negative jacobiens and pressures to back!). In GIBI, the user has the operators OPPOSITE and TO DIRECT for modification orientation of elements. One will refer advantageously to the order MODI_MAILLAGE ([U4.23.04], keywords ORIE_PEAU_2D/3D) to intervene on the orientation of the meshs in *Aster*,
- the operator PRE_GIBI suppose that GIBI was carried out without error, i.e. the second line of the file .mgib is "LEVEL LEVEL ERROR 0 DIMENSION…"
  In the contrary case, one carries out, nevertheless the interface, by emitting an alarm,
- the file containing grid GIBI must be produced by the order "TO SAVE FORMAT grid"; it is the only format read again by the interface PRE_GIBI.

4 Interface with Aster

The interface with *Aster* is activated by the order PRE_GIBI [U7.01.11].

5 Entities of grid GIBI and Aster

Maillor GIBI handles typified and named objects (cf Doc. of use of GIBI).

The types used are:

- constants whole, real, text,…,
- POINTS,
- GRIDS (together of meshs).

The order PRE_GIBI causes:

- to write points (GIBI) in the form of nodes (*Aster*) : names of the nodes *Aster* are form: N_i where i is number GIBI of the corresponding point,
- to write the meshs (*Aster*) contained in grids (GIBI): names of the meshs *Aster* are form: M_j where j is the sequence number of mesh GIBI in the file of result GIBI,
- to write GROUP_MA (*Aster*) corresponding to grids (GIBI) and same names,

  Note:

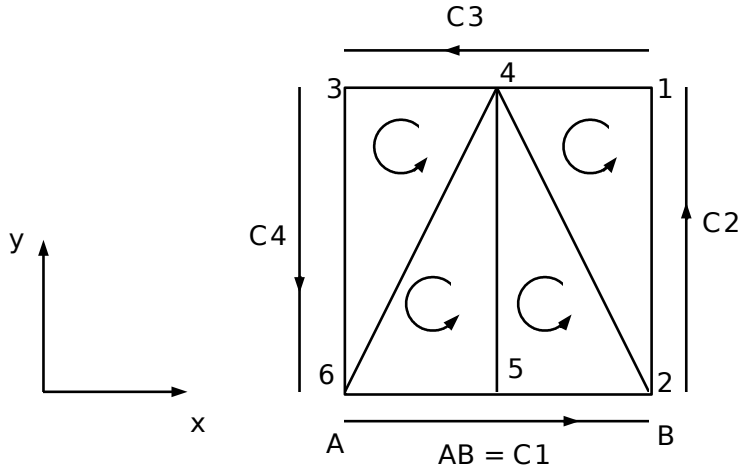
  > *The groups of meshs contain only the meshs defining grid GIBI indeed. They do not contain the meshs of the objects referred in this grid (edges for example).*

- to write GROUP_NO (*Aster*) reduced to only one nodes and whose names are the names of the points of GIBI.

It is noticed that PRE_GIBI do not generate GROUP_NO containing several nodes. What would be sometimes very useful to impose boundary conditions on all the nodes of an edge. With this intention, the user has in Aster the order DEFI_GROUP who allows to create groups of nodes starting from groups of meshs.

The keyword CREA_GROUP_NO of the operator DEFI_GROUP [U4.22.01] allows to circumvent this obstacle.

6 Examples

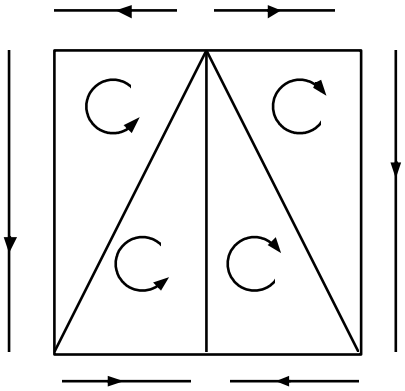


- for GIBI: A and B are named points.
  One thus finds in the grid *Aster* GROUP_NO names A and B
- for GIBI: AB , $CARRE$, $C1$,..., $C4$ are named grids.
  One thus finds in the grid *Aster* GROUP_MA names: AB , $CARRE$, $C1$,..., $C4$

  Note:

  > *The interface writes only once possibly identical meshs (what arrives sometimes with the operator " AND "GIBI). This has as a consequence that the names of the meshs M_j can have a "classification with holes". One should not worry some.*

- The orientation of the meshs is that given by GIBI. This can have importance for certain changes: pressure,…
  It is necessary to be conscious owing to the fact that the operators of grid of GIBI direct the elements (with rules specified in documentation). Certain elements can be "turned over" compared to others. Another command set GIBI could, for example to generate the grid:



6.1 Notice concerning the order POIN GIBI

This order can produce results of the different type:

- if this order makes it possible to recover only one point,

  for example:

  ```
  Pa = louse   POIN   PROC   With   ;
  ```

  or

  ```
  Pa = louse   POIN   INITIAL   ;
  ```

  then PA is a named point GIBI.

  For *Aster*, there will be one GROUP_NO of name PA who contains only one node.

- if this order makes it possible to recover several points:

  for example:

  ```
  Pa = louse   POIN   RIGHT  With   B   1.0D-2 ;
  ```

  then PA is a named grid GIBI:

  This grid consists of meshs POI1.

  For *Aster*, there will be one GROUP_MA of name PA who contains several POI1. To use this object as GROUP_NO, the keyword factor will have to be used CREA_GROUP_NO order DEFI_GROUP.

6.2 Command file GIBI

```
OPTI TITHE 2 ELEM TRI 3;
*
* POINTS NAME:
*
To = 0.0   0.0  ;
B = 1.0   0.0  ;
*
* GRIDS:
*
AB = DROI 2  WITH B;
SQUARE = AB TRAN 1  (0. 1.);
C1 C2 C3 C4 = DIMENSION SQUARES    ;
*
TO SAVE FORMAT SQUARES;
END;
*
```

6.3 File of result GIBI: (as an indication)

```
RECORDING OF THE TYPE   4
 LEVEL  11 LEVEL ERROR   0 DIMENSION   2
 DENSITY 0.00000E+00
 RECORDING OF THE TYPE   7
 NUMBER INFORMATION CASTEM2000   8
 IFOUR  -1 NIFOUR   0 IFOMOD  -1 IECHO   1 IIMPI   0 IOSPI   0 ISOTYP   1
 NSDPGE     0
 RECORDING OF THE TYPE   2
 PILE NUMBER   1NBRE OBJECTS NAME        6NBRE OBJECTS       5
 AB       SQUARE    C1       C2       C3       C4
      2       1       2       3       4       5
      4       0       4       3       4
      2       3       4       5
      0       0       0       0
      1       2       3       1       3       4       2       5       3       5
       6       3
       2       0       0       2       2
       0       0
       1       2       2       5
       2       0       0       2       1
       0
       5       6
       2       0       0       2       2
       0       0
       6       3       3       4
       2       0       0       2       1
       0
       4       1
 RECORDING OF THE TYPE   2
 PILE NUMBER  32NBRE OBJECTS NAME        2NBRE OBJECTS       6
 With        B
       1       5
       6
       1       3       6       5       2       7
 RECORDING OF THE TYPE   2
 PILE NUMBER  33NBRE OBJECTS NAME        0NBRE OBJECTS       1
      21
  0.00000000000000E+00  0.00000000000000E+00  0.00000000000000E+00
  1.00000000000000E+00  0.00000000000000E+00  0.00000000000000E+00
  5.00000000000000E-01  0.00000000000000E+00  5.00000000000000E-01
  0.00000000000000E+00  1.00000000000000E+00  0.00000000000000E+00
  0.00000000000000E+00  1.00000000000000E+00  0.00000000000000E+00
  5.00000000000000E-01  1.00000000000000E+00  5.00000000000000E-01
  1.00000000000000E+00  1.00000000000000E+00  0.00000000000000E+00
 RECORDING OF THE TYPE   5
AUTOMATIC LABEL:   1
```

6.4 File of grid *Aster* product by the operator PRE_GIBI

```
TITRE
% GIBI FECIT
FINSF
%
COOR_2D
N1          0.00000000000000E+00    0.00000000000000E+00
N2          1.00000000000000E+00    0.00000000000000E+00
N3          5.00000000000000E-01    0.00000000000000E+00
N4          0.00000000000000E+00    1.00000000000000E+00
N5          0.00000000000000E+00    1.00000000000000E+00
N6          5.00000000000000E-01    1.00000000000000E+00
N7          1.00000000000000E+00    1.00000000000000E+00
FINSF
%
TRIA3
M1          N1        N3        N6
M2          N1        N6        N5
M3          N3        N2        N6
M4          N2        N7        N6
FINSF
%
SEG2
M5          N1        N3
M6          N3        N2
FINSF
%
SEG2
M7          N2        N7
FINSF
%
SEG2
M8          N7        N6
M9          N6        N5
FINSF
%
SEG2
M10         N5        N1
FINSF
%
```

```
GROUP_NO
B           N2
FINSF
%
GROUP_NO
A           N1
FINSF
%
GROUP_MA
  C4
M10
FINSF
%
GROUP_MA
  C3
M8          M9
FINSF
%
GROUP_MA
  C2
M7
FINSF
%
GROUP_MA
  C1

M5          M6
FINSF
%
GROUP_MA
  CARRE
M1          M2    M3    M4
FINSF
%
GROUP_MA
  AB
M5          M6
FINSF
%
FIN
```