Operator REST_REDUIT_COMPLET

The goal of the operator is to rebuild the solutions on a complete model starting from a scale model.

The operator rebuilds one `evol_ther` or one `evol_noli` starting from an empirical base (see [U4.67.01], Operator `DEFI_BASE_REDUITE`) and of the result of a reduced calculation.
Contents

1 Syntax .................................................................................................................................................. 3
2 Operands .............................................................................................................................................. 4
   2.1 OperandS PHENOMENON and MODEL ......................................................................................... 4
   2.2 OperandS BASE_PRIMAL ............................................................................................................... 4
   2.3 Operand REST_DUAL .................................................................................................................... 4
      2.3.1 Operand BASE_DUAL ............................................................................................................ 4
      2.3.2 Operand CORR_COMPLET .................................................................................................... 4
      2.3.3 Operand GROUP_NO_INTERF ............................................................................................... 4
   2.4 Operand RESULTAT_REDUIT ........................................................................................................ 4
      2.4.1 Operand TABL_COOR_REDUIT ............................................................................................. 5
   2.5 Structure of output data ................................................................................................................ 5
1 Syntax

```plaintext
evol = REST_REDUIT_COMPLET (  
  ♦ PHENOMENON = /'MECHANICAL' /'THERMAL'     [DEFECT]  
  ♦ MODEL = model                     [modele_sdaster]  
  ♦ RESULTAT_REDUIT = base2,           [resultat_sdaster]  
  ♦ BASE_PRIMAL = baseprim,            [mode_empI]  
  ♦ REST_DUAL = /'NOT', /'YES',         [DEFECT]  
     # if REST_DUAL=' OUI'  
  ♦ BASE_AL = base dual,               [mode_empi]  
  ♦ GROUP_NO_INTERF = grno,            [grno]  
     ♦ CORR_COMPLET = /'NOT' /'YES'     [DEFECT]  
          ♦ TABL_COOR_REDUIT = tabl_coor, [table]  
          ♦ TITLE = title,          [1_Kn]  
          ♦ INFORMATION = /1,       [DEFECT]  
                          /2,  
  )
```

Warning: The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.

Copyright 2020 EDF R&D - Licensed under the terms of the GNU FDL (http://www.gnu.org/copyleft/fdl.html)
2 Operands

2.1 OperandS PHENOMENON and MODEL

◊ PHENOMENON = /'MECHANICAL'/ [DEFECT]
/'THERMAL'/

Type of treated phenomenon: mechanics or thermics. The choice of the phenomenon will typify the structure of output data: evol_ther for THERMICS or evol_noli for MECHANICS.

◊ MODEL = model [modele_sdaster]

NRom of the model on which will be rebuilt the structure of data result.

2.2 OperandS BASE_PRIMAL

◊ BASE_PRIMAL = baseprim, [mode_empi]

The primal base will use to rebuild the primal fields: DEPL for mechanics and THER for thermics. It is necessary that this base was built on the model given by the keyword MODEL.

2.3 Operand REST_DUAL

◊ REST_DUAL = /'NOT'/, [DEFECT]
/'YES'/,

By defaults, the duaux fields are not restored (REST_DUAL=' NON'). Keywords BASE_DUAL and GROUP_NO_INTERF are necessary to restore the duaux fields.

2.3.1 Operand BASE_DUAL

◊ BASE_AL = base dual, [mode_empi]

The dual base is necessary to rebuild the duaux fields: SIEF_NOEU for mechanics and FLUX_NOEU for thermics. It is necessary that this base was built on the model given by the keyword MODEL.

2.3.2 Operand CORR_COMPLET

◊ CORR_COMPLET = /'NOT'/ [DEFECT]
/'YES'/

When one did a calculation with correction finite element (see [U4.51.03]), the use of this keyword makes it possible to improve quality of the fields.

2.3.3 Operand GROUP_NO_INTERF

◊ GROUP_NO_INTERF = grno, [grno]

Group of nodes defining the interface the reduced field and the rest of the model.

2.4 Operand RESULTAT_REDUIT

◊ RESULTAT_REDUIT = base2, [resultat_sdaster]
This keyword gives the structure of data result (evol_ther or evol_noli) who comes from the calculation reduced with THER_NON_LINE or STAT_NON_LINE. This structure of data contains all the necessary information to rebuild the results on model given by the keyword MODEL.

2.4.1Operand TABL_COOR_REDUIT

◊ TABL_COOR_REDUIT = tabl_coor ,

Lorsqu'on carry out a gappy-POD by REST_REDUIT_COMPLET, it is necessary to have the reduced coordinates of calculation. These coordinates are stored in a structure of data table of name 'COOR_REDUIT' who is attached to structure of data result. One can recover it via the operator RECU_TABLE. For example:

```plaintext
coorredp=RECU_TABLE (CO=resul, NOM_TABLE='COOR_REDUIT')
```

But if you recover the empirical base previously calculated by an operator like LIRE_RESU (in particular with format MED), this table is not available. The operator TABL_COOR_REDUIT thus allows to give it to REST_REDUIT_COMPLET.

It is thus necessary to have saved this table upstream at the same time as the non-linear result (by one IMPR_TABLE), then to recover it (by one LIRE_TABLE) to give it to REST_REDUIT_COMPLET.

2.5 Structure of output data

The structure of output data is one sd_resultat standard of code_aster. It is complete for a resumption of calculation in thermics, on the other hand, in mechanics, it will miss the field of internal variables (VARI_ELGA). It will thus be necessary to supplement calculation with ETAT_INIT/VARI in STAT_NON_LINE.