

Structure of data sd_matr_elem and sd_vect_elem

Summary:

Contents

1 Structures of Data in a few words.....	3
2 Tree structure.....	3
3 Contents of the objects JEVEUX.....	3
3.1 Object '.RERR'.....	3
3.2 Object '.RELR'.....	4
3.3 Object '.RELC'.....	4
4 Examples.....	5
4.1 sd_matr_elem.....	5
4.2 sd_vect_elem.....	5

1 Structures of Data in a few words

One sd_matr_elem represent a set of elementary matrices.
One sd_vect_elem represent a set of elementary vectors.

One can "assemble" sd_matr_elem to obtain one sd_matr_asse.
One can "assemble" sd_vect_elem to obtain one sd_cham_no.

Practically, them sd_matr_elem and them sd_vect_elem consist of a set of sd_resuelem [D4.06.05]. One sd_resuelem being the whole of the matrices (or vectors) elementary correspondents with the elements of one sd_ligrel.

One sd_matr_elem (or one sd_vect_elem) can not contain any sd_resuelem. That can arrive if it sd_modele contains only static substructures.

2 Tree structure

```
sd_matr_elem (K19):: =record  
  
  (O)   \.RERR':  OBJ S   V K24 long=5  
  (F)   \.RELR':  OBJ S   V K24 long=*
```

```
sd_vect_elem (K19):: =record  
  
  (O)   \.RERR':  OBJ S   V K24 long=5  
  (F)   \.RELR':  OBJ S   V K24 long=5  
  (F)   \.RELC':  OBJ XC  V I   NO
```

3 Contents of the objects JEVEUX

3.1 Object \.RERR'

```
\.RERR': S V K24 long=5
```

That is to say $V = \text{\.RERR}'$

V (1)	name of sd_modele subjacent
V (2)	name of the attached on-option: 'RIGI_MECA', 'MASS_THER', 'CHAR_MECA', ...
V (3)	/ 'OUI_SOUS_STRUC' / 'NON_SOUS_STRUC'
V (4)	Name of sd_cham_mater subjacent with sd_matr_elem (or sd_vect_elem).
V (5)	Name of sd_cara_elem subjacent with sd_matr_elem (or sd_vect_elem).

V (3) =/ 'OUI_SOUS_STRUC' / 'NON_SOUS_STRUC' : Indicate if the elementary terms (matrices or vectors) of the static substructures are to be taken into account (or not).

For example, for one `sd_matr_elem` of type `'RIGI_MECA'` who would relate to only blockings of `ddl`s with dualisation, it is necessary to be unaware of the substructures. If not, during the assembly, one would be likely to double the rigidity of the substructures.

Notice :

`'OUI_SOUS_STRUC'` does not want to say that the model has active substructures inevitably. But if it has some, they will be taken into account.

The object `.RERR` is obtained by calling the routine `memare.f`.

3.2 Object `'RELR'`

`'RELR'` : S V K24

This object contains the list of `sd_resuelem` composing it `sd_matr_elem` (or it `sd_vect_elem`).

This object does not exist if the model contains only substructures (and not ordinary finite elements).

That is to say $V = \text{'RELR'}$,

$V(I) (1:19)$: name of $I^{\text{ème}}$ `sd_resuelem` `sd_matr_elem` (or of `sd_vect_elem`).

Caution :

For `sd_vect_elem`, $V(I)$ can be the name of one `sd_cham_no` (and not of one `resuelem`) when this second member comes from a load of the type `AFFE_CHAR_MECA/VECT_ASSE`

Note:

The number of `sd_resuelem` is obtained by `'LONUTI'` object `.RELR`.

The utility `reajre.f` allows to store them `resuelem` in `sd_matr_elem` (or `sd_vect_elem`).

3.3 Object `'RELC'`

`'RELC'` : XC V I NO ()

This object exists only if the grid contains super-meshes. This collection is named by the loading cases indicated by the user in the order `CALC_VECT_ELEM`. All objects of this collection have the same length.

That is to say `nomcas` such a loading case,

$V = \text{'RELC'}(nomcas)$.

$LENGTH(V) = nbmas =$ many super-meshes of the subjacent grid

for $I = 1, nbmas$

$V(I)$:

/ 1 if the super-mesh I is active for the loading `nomcas`

/ 0 if the super-mesh I is not active for the loading `nomcas`

4 Examples

4.1 sd_matr_elem

```
MATELE_1=CALC_MATR_ELEM (  MODELE=MODELE_1,      CHARGE=CHARGE_1,  
                           CARA_ELEM=CARAC_1,  
                           CHAM_MATER=CH_MAT_1, OPTION=' RIGI_MECA')
```

give:

```
-----  
SEGMENT IMPRESSION OF VALUES >MATELE_1          .RELR      <  
  
  1 - >MATELE_1.ME001          <>MATELE_1.ME002      <  
  3 - >                          <                          <  
-----  
SEGMENT IMPRESSION OF VALUES >MATELE_1          .RERR      <  
  
  1 - >MODELE_1                <>RIGI_MECA          <  
  3 - >OUI_SOUS_STRUC          <>CH_MAT_1.MATE_CODE   <  
  5 - >CARAC_1                  <                          <
```

4.2 sd_vect_elem

```
VECELE_1=CALC_VECT_ELEM (  CHARGE=CHARGE_1,  OPTION=' CHAR_MECA')
```

give:

```
-----  
SEGMENT IMPRESSION OF VALUES >VECELE_1          .RELR      <  
  
  1 - >VECELE_1.VE001          <>VECELE_1.VE002      <  
  3 - >                          <>                          <  
  5 - >                          <>                          <  
  7 - >                          <>                          <  
  9 - >                          <>                          <  
-----  
SEGMENT IMPRESSION OF VALUES >VECELE_1          .RERR      <  
  
  1 - >MODELE_1                <>CHAR_MECA          <  
  3 - >NON_SOUS_STRUC          <>                          <  
  5 - >                          <                          <
```