

Structure of Data sd_ligrel and sd_modele

Summary:

The structure DE given sd_modele represents the result of the assignment of finite elements on the meshes of a grid.

A structure of data sd_ligrel is used to list groups of finite elements in the same way standard.

Note: sd_modele is not the only structure of data being able to contain one sd_ligrel.

Contents

1 Structures of data in a few words.....	3
2 Tree structures.....	4
3 Contents of the objects JEVEUX sd_ligrel.....	5
3.1 Object '.LGRF'.....	5
3.2 Object '.NBNO'.....	5
3.3 Object '.PRNM'.....	5
3.4 Object '.LIEL'.....	5
3.5 Object '.REPE'.....	6
3.6 Object '.NVGE'.....	6
3.7 Object '.SSSA'.....	6
3.8 Object '.NEMA'.....	7
3.9 Object '.PRNS'.....	7
3.10 Object '.LGNS'.....	8
3.11 Notice on the redundant objects of sd_ligrel.....	8
4 Contents of the objects JEVEUX sd_modele.....	10
4.1 Object '.MESH'.....	10
4.2 Object '.LEFT'.....	10
4.3 Object '.XFEM'.....	10
5 Examples.....	11
5.1 SD sd_modele.....	11
5.2 SD sd_ligrel (load, double lagrange).....	11

1 Structures of data in a few words

In short:

- one `sd_ligrel` contains a set of finite elements and/or static substructures. A finite element being the formed couple of a mesh (mesh of the grid or “late” mesh) and of a kind of finite element (`type_elem`),
- one `sd_ligrel` can contain static substructures: “activation” of super-meshes of `sd_maillage` [D4.06.01 §2]. If one `sd_ligrel` does not contain finite elements, then it must contain substructures.
- the meshes supporting the finite elements can be meshes of the grid or meshes additional (or late),
- one `sd_modele` one contains `sd_ligrel` ; but it can also exist one `sd_ligrel` in other SD; for example in one `sd_char_meca` [D4.06.04],
- to allow the parallelism of elementary calculations and the assemblies, one arranges oneself so that one can, in general, “to go up” of one `sd_ligrel` until one `sd_modele` (which contains `sd_partition`).
- in one `sd_ligrel`, a mesh of the grid can carry only one finite element to more (the object `.REPE`),
- with one `sd_ligrel` is associated one PHENOMENON and only one: ‘MECHANICS’, ‘THERMICS’,...
- with each PHENOMENON is associated a particular `mode_local`: ‘DDL_MECA’, ‘DDL_THER’ or ‘DDL_ACOU’. This `mode_local` determines (via the catalogue of `type_element`) ddls of the finite elements of `sd_ligrel` (objects `.PRNM` and `.PRNS`),
- one `sd_ligrel` (like one `sd_modele`) is always associated with one `sd_maillage`.

2 Tree structures

```
sd_ligrel      (K19)      .: =record

(O)   \.NBNO' :      OJB          S   V   I
(O)   \.LGRF' :      OJB          S   V   K8      long=3
(O)   \.PRNM' :      OJB          S   V   I

| % if the sd_ligrel contains finite elements
(O)   \.LIEL' :      OJB          XC  V   I      NAKED ()

(F) % if the sd_ligrel contains elements on meshes of      grid:
      \.REPE' :      OJB          S   V   I

(F) % if the sd_ligrel contains elements on meshes
late:
      \.NEMA' :      OJB          XC  V   I      NAKED ()

(F) % if the sd_ligrel contains late nodes:
      \.PRNS' :      OJB          S   V   I
      \.LGNS' :      OJB          S   V   I

| % if the sd_ligrel contains static substructures
(O)   \.SSSA:      OJB          S   V   I

| % if the sd_ligrel contains elements needing the vicinity
(O)   \.NVGE:      OJB          S   V   K16 (long=1)

sd_modele     (K8)      .: =record

(O)   \.MODELE' :      sd_ligrel

(F) % if it sd_modele contains finite elements:
      \.MAILLE' :      OJB          S   V   I

(F) % if it sd_modele comes from the order MODI_MODELE_XFEM:
      \. $VIDE' :      sd_modele_XFEM

(F) % if the user wishes parallel elementary calculations:
      \.PARTIT' :      OJB          S   V   K8
```

3 Contents of the objects JEVEUX sd_ligrel

3.1 Object ``.LGRF'`

``.LGRF'` : S V K8 (long=3)

V (1) : name of the grid associated with sd_ligrel.

V (2) : name of sd_modele giving access one sd_partition (parallelism of elementary calculations). If this information is missing, one cannot parallel elementary calculations (nor assemblies).

V (3) : is worth ```` (in the case general) or `'LAG1'` in the case of a ligrel of load for which the user asked to use simple Lagrange.

The attribute `'DOCU'` of this object contains: `'MECA'/'THER'/'ACOU'`. This information makes it possible to determine the ddls carried by the nodes (objects `.PRNM` and `.PRNS`)

3.2 Object ``.NBNO'`

``.NBNO'` : S V I (long=1)

Contient the number of late nodes of sd_ligrel

3.3 Object `' .PRNM'`

`' .PRNM'` : S V I

This object describes the degrees of freedom carried by the nodes of sd_ligrel. He takes account of the degrees of freedom brought by the finite elements and of those brought by the substructures.

That is to say:

$V = .PRNM$

nbno = many nodes of the grid

nec = many coded entireties necessary to the fundamental size associated with sd_ligrel

nec = (number of CMP (fundamental size)/ 30) +1

for I = 1, nbno :

V (nec* (i-1) +1)	1 ^{er} coded entirety of the node I
V (nec* (i-1) +2)	2 ^{ème} coded entirety of the node I
...	
V (nec* (i-1) +nec)	last coded entirety of the node I

3.4 Object ``.LIEL'`

```
\.LIEL' : XC V I NAKED ()
```

The collection `.LIEL` contains the numbers of the supporting meshes of the finite elements. The elements are gathered by groups of elements in the same way standard (`GREL`) [D0.04.02]. An object of the collection corresponds to one `GREL`.

That is to say I^{ème} `GREL SD_LIGREL`

```
V = .LIEL (I)  
N = LONG (V) = 1+nbre meshes of Ième GREL
```

V (1)	number of the mesh associated with the 1 ^{er} element of <code>GREL</code>
V (2)	number of the mesh associated with the 2 ^{ème} element of <code>GREL</code>
...	
V (n-1)	number of the mesh associated with the last element with <code>GREL</code>
V (N)	number of the type of element associated with <code>GREL I</code> (object <code>&CATA.TE.NOMTE [D4.04.01]</code>)

Important conventions :

- if the mesh is a mesh of the grid its number is stored such as it is.
- if the mesh is a late mesh, its number is stored with the minus sign (cf object `.NEMA`),
- the number of `GREL` of one `sd_ligrel` is worth: `NUITOC (.LIEL')` (Attention: the collection can be oversize: `NMAXOC ≥ NUTIOC`)

3.5 Object `\.REPE'`

```
\.REPE' : S V I
```

That is to say

```
V = \.REPE'  
nbma = many meshes of the grid associated with sd_ligrel LONG (V) = 2*nbma
```

```
for I = 1, nbma  
  V (2 (i-1) +1) : number of GREL associated with the mesh I grid  
  V (2 (i-1) +2) : position in GREL mesh I grid
```

This object is I" 'opposite" of L' object `.LIEL` concerning the meshes of the grid

If I am a nonaffected mesh: $V (2 (i-1) +1) = V (2 (i-1) +2) = 0$

3.6 Object `'NVGE'`

```
\.NVGE' : S V K16 (long=1)
```

V (1): name of `sd_voisinage` affected to `ligrel`

3.7 Object `'SSSA'`

```
\.SSSA' : S V I
```

That is to say:

```
V = \.SSSA'
```

nb_sm = many super-meshs of the grid

```
LENGTH (V) = nb_sm + 3
```

V (i=1, nbsm)	/ 1 if the super-mesh I is affected ("active" under-structuring) / 0 if not
V (nbsm+1)	many super-meshs of the grid
V (nbsm+2)	many active substructures
V (nbsm+3)	many nodes of Lagrange of the grid

Note:

V (nb_sm+1) and V (nb_sm+3) do not have to be used any more, information existing already in the grid (object . TITHE).

3.8 Object '.NEMA'

```
\.NEMA` : XC V I NAKED
```

This collection describes the late meshes of sd_ligrel .

It exists an object of collection by late mesh.

Caution :

This collection is perhaps oversize. The real number of late meshes is obtained by NUTIOC (.NEMA) .

That is to say:

```
nbmas = many late meshes of sd_ligrel  
nbmas = NUTIOC (.NEMA)
```

```
for I = 1, nbmas
```

```
V = .NEMA (I)
```

```
N = LONG (V) = (many nodes of the mesh) + 1
```

V (L)	number of the 1 ^{er} node of the mesh I
V (2)	number of the 2 ^{eme} node of the mesh I
...	
V (n-1)	number of the last node of the mesh I
V (N)	number of the type of the mesh I

If the number of a node is negative, it is the opposite of the number of a late node of sd_ligrel (cf object. NBNO).

3.9 Object '.PRNS'

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 2021 EDF R&D - Licensed under the terms of the GNU FDL (<http://www.gnu.org/copyleft/fdl.html>)

`\.PRNS` : S V I

This object describes the ddls carried by the late nodes of `sd_ligrel`.

That is to say:

V = `.PRNS`
`nbnos` = many late nodes of `sd_ligrel`
`nec` = many coded entireties necessary to the fundamental size associated with `sd_ligrel`
`nec` = (number of CMP (fundamental size)/ 30) + 1

for I = 1, `nbnos`:

V (nec* (i-1) +1)	1 ^{er} coded entirety of the node I
V (nec* (i-1) +2)	2 ^{ème} coded entirety of the node I
...	
V (nec* (i-1) +nec)	last coded entirety of the node I

“The small” piece of `.PRNS` concerning the node I is what one calls a Descriptor-Size [D4.06.05].

3.10 Object ‘.LGNS’

`\.LGNS` : S V I

The length of this vector is higher than the number of late nodes of `sd_ligrel` (the object on - is dimensioned).

V (INO) : number indicating how the late node of Lagrange INO must be numbered (see `sd_num_dld`).

V (INO) : 0	the node INO is not a node of Lagrange
V (INO) : +1	the node INO is a node of Lagrange of the type “1”. It must be numbered before the physical ddls that it constrained.
V (INO) : - 2	the node INO is a node of Lagrange of the type “2”. It must be numbered after the physical ddls that it constrained.
V (INO) : - 1	the node INO is a node of Lagrange of the type “1”. It must be numbered after the physical ddls that it constrained.
V (INO) : +2	the node INO is a node of Lagrange of the type “2”. It must be numbered before the physical ddls that it constrained.

3.11 Notice on the redundant objects of `sd_ligrel`

Certain objects of `sd_ligrel` can result from other objects. The purpose of this redundant information is to accelerate certain treatments (from the point of view of time CPU).

The object.REPE can result from.LIEL Objects.PRNM and.PRNS can result from.LIEL.SSSA and.LGRF

The routine `cormgi.f` allows to calculate.REPE The routine `initel.f` allows to calculate.PRNM and.PRNS.

4 Contents of the objects JEVEUX sd_modele

4.1 Object '.MESH'

``.MESH` : S V I`

That is to say $V = \text{`.MAILLE`}$

$\text{LENGTH}(V) = \text{many meshes of the grid} = \text{nbma}$

for $I = 1, \text{nbma}$

$V(I)$: number of the type of element carried by the mesh I
(= 0 if the mesh is not affected by a finite element)

4.2 Object '.LEFT'

``.LEFT` : S V long K8 = 1`

$V(1)$: name of `sd_partition` describing the parallelism of elementary calculations.

4.3 Object '.XFEM'

``.XFEM` : S V long K8 = 1`

$V(1)$: name of preprocessings X-FEM (pre-conditioner) necessary; this is information available in the `sd_modele_xfem`.

5 Examples

5.1 SD sd_modele

```
MOTH=AFFE_MODELE ( MAILLAGE=MAIL,
                   AFPE=_F (ALL = 'YES', MODELING = 'AXIS', PHENOMENON = 'THERMAL'))
```

product:

```
-----
SEGMENT IMPRESSION OF VALUES >MOTH      .MAILLE      <
  1 -          289          289          300          300          300
-----
IMPRESSION OF THE COLLECTION: MOTH      .MODELE      .LIEL
OBJECT IMPRESSION OF COLLECTION CONTIGUE>MOTH .MODELE      .LIEL< OC:      1
  1 -          1          2          289
OBJECT IMPRESSION OF COLLECTION CONTIGUE>MOTH .MODELE      .LIEL< OC:      2
  1 -          3          4          5          300
-----
SEGMENT IMPRESSION OF VALUES >MOTH      .MODELE      .NBNO      <
  1 -          0
-----
SEGMENT IMPRESSION OF VALUES >MOTH      .MODELE      .LGRF      <
  1 - >MAIL      <
-----
SEGMENT IMPRESSION OF VALUES >MOTH      .MODELE      .PRNM      <
  1 -          2          2          2          2          2
  6 -          2          0          0          0          0
...
 41 -          0          0          2          2          0
 46 -          0          0          0          0          0
 51 -          0          0          0          0          0
 56 -          0          0          0          0          0
 61 -          0          0          0          0          0
-----
SEGMENT IMPRESSION OF VALUES >MOTH      .MODELE      .REPE      <
  1 -          1          1          1          2          2
  6 -          1          2          2          2          3
-----
```

5.2 SD sd_ligrel (load, double lagrange)

```
CHTH=AFPE_CHAR_THER ( MODELE=MOTH,
                      TEMP_IMPO=_F (NODE = 'N4', TEMP = 100.0))
```

product:

```
-----
SEGMENT IMPRESSION OF VALUES >CHTH      .CHTH.LIGRE.LGNS      <
  1 -          1          -2          0          0          0
  6 -          0          0          0
-----
IMPRESSION OF THE COLLECTION: CHTH      .CHTH.LIGRE.LIEL
OBJECT IMPRESSION OF COLLECTION CONTIGUE>CHTH .CHTH.LIGRE.LIEL< OC:      1
  1 -          -1          92
-----
```

Code_Aster

Version
default

Titre : Structures de données sd_ligrel et sd_modele
Responsable : PELLET Jacques

Date : 22/11/2019 Page : 12/12
Clé : D4.06.02 Révision :
d6a7fa3337c8

```
SEGMENT IMPRESSION OF VALUES >CHTH .CHTH.LIGRE.NBNO <
  1 - 2
-----
IMPRESSION OF THE COLLECTION: CHTH .CHTH.LIGRE.NEMA
OBJECT IMPRESSION OF COLLECTION CONTIGUE>CHTH .CHTH.LIGRE.NEMA< OC: 1
  1 - 4 -1 -2 4
-----
SEGMENT IMPRESSION OF VALUES >CHTH .CHTH.LIGRE.LGRF <
  1 - >MAIL <
-----
SEGMENT IMPRESSION OF VALUES >CHTH .CHTH.LIGRE.PRNM <
  1 - 0 0 0 2 0
  6 - 0 0 0 0 0
...
  61 - 0 0 0
-----
SEGMENT IMPRESSION OF VALUES >CHTH .CHTH.LIGRE.PRNS <
  1 - 16 16
-----
```