

## Structure de données sd\_liste\_rela

---

Résumé :

## Table des matières

---

1 Généralités.....	3
2 Arborescence.....	3
3 Contenu des objets.....	3
3.1 Objet ".RLNR".....	3
3.2 Objet ".RLTC".....	4
3.3 Objet ".RLTV".....	4
3.4 Objet ".RLNT".....	4
3.5 Objet ".RLPO".....	4
3.6 Objet ".RLCO".....	4
3.7 Objet ".RLNO".....	4
3.8 Objet ".RLDD".....	4
3.9 Objet ".RLBE".....	5
3.10 Objet ".RLSU".....	5

## 1 Généralités

La structure de données `sd_liste_rela` est une structure de données "volatile" qui sert dans les opérateurs `AFFE_CHAR_XXXX` de structure intermédiaire entre les données de l'utilisateur (les relations linéaires à dualiser) et leur traduction en éléments finis de Lagrange (dans le `LIGREL` de `CHARGE`) et en `CARTES` contenant les coefficients des relations.

En gros, l'idée est la suivante : on stocke les relations linéaires au fur et à mesure (une par une) dans une `sd_liste_rela` (routine `AFRELA`) puis on "verse" la `sd_liste_rela` dans la `sd_char_xxx` à la fin de la commande (routine `AFLRCH`).

**Remarque :**

*on peut créer plusieurs `sd_liste_rela` au sein d'une même commande `AFFE_CHAR_XXX`. Il faut alors "verser" chacune de ces SD dans la `sd_char_xxx`. Cela permet de rendre indépendantes les différentes routines qui gèrent des relations linéaires : `caddli`, `cafaci`, ...*

## 2 Arborescence

```
sd_liste_rela (K19) ::=record
  (o) '.RLCO' : OJB S V R/C
  (o) '.RLBE' : OJB S V R/C/K24
  (o) '.RLDD' : OJB S V K8
  (o) '.RLNO' : OJB S V K8
  (o) '.RLNT' : OJB S V I
  (o) '.RLPO' : OJB S V I
  (o) '.RLSU' : OJB S V I
  (o) '.RLTC' : OJB S E K8
  (o) '.RLTV' : OJB S E K8
  (o) '.RLNR' : OJB S E I
```

## 3 Contenu des objets

**Notations:**

- `nb_rela` : nombre de relations stockées dans la `sd_liste_rela`.
- `nb_coef_lrel` : nombre de coefficients d'une relation linéaire (partie "gauche" de l'équation). Les coefficients peuvent être : réels ou complexes.
- `nb_coef_tot` : nombre total de coefficients de l'ensemble des relations linéaires stockées dans la `sd_liste_rela`.  
$$\text{nb\_coef\_tot} = \text{nb\_coef\_lrel}(1) + \text{nb\_coef\_lrel}(2) + \dots$$
- `coef_impo` : valeur imposée pour la relation linéaire (partie "droite" de l'équation ou "second membre"). La valeur imposée peut être : réelle, complexe ou fonction.

### 3.1 Objet ".RLNR"

Cet objet est de longueur 1.

`RLNR(1)` : nombre de relations linéaires `nb_rela`.

## 3.2 Objet ".RLTC"

Cet objet est de longueur 1.

RLTC(1) : type des coefficients des relations 'REEL' ou 'COMP'.

## 3.3 Objet ".RLTV"

Cet objet est de longueur 1.

RLTV(1) : type des seconds membres des relations 'REEL', 'COMP' ou 'FONC'.

## 3.4 Objet ".RLNT"

Cet objet est de longueur nb\_rela.

Il contient les nombres de coefficients pour chaque relation.

RLNT(irela) : nombre de coefficients de la relation irela = nb\_coef\_1rel(irela).

## 3.5 Objet ".RLPO"

Cet objet est de longueur nb\_rela.

Il permet de "pointer" (pour une relation donnée) dans les objets .RLCO, .RLNO et .RLDD.

Ces 3 objets ont la même dimension et on y pointe de la même façon.

RLPO(irela) : adresse dans .RLCO (par exemple) du dernier coefficient de la relation irela.

RLCO(RLPO(irela) - RLNT(irela) + 1) est le 1<sup>er</sup> terme de la relation irela.

## 3.6 Objet ".RLCO"

Cet objet est de longueur nb\_coef\_tot.

Il contient les coefficients (R ou C) des équations.

## 3.7 Objet ".RLNO"

Cet objet est de longueur nb\_coef\_tot.

Il contient les noms des nœuds impliqués dans les relations.

## 3.8 Objet ".RLDD"

Cet objet est de longueur nb\_coef\_tot.

Il contient les noms des CMPS impliquées dans les relations.

## 3.9 Objet ".RLBE"

Cet objet est de longueur `nb_rela`.

Il contient les seconds membres des relations.

`RLBE(irela)` : second membre de la relation `irela` = `coef_impo(irela)`.

## 3.10 Objet ".RLSU"

Cet objet est de longueur `nb_rela`. C'est un indicateur pour dire si les relations linéaires doivent être prises en compte (ou non).

En effet, avant de "verser" les relations linéaires dans la `sd_charge`, on examine si certaines relations ne sont pas données en plusieurs exemplaires (doublons).

```
RLSU(irela) :  
  / 0 -> la relation irela est à prendre en compte.  
  / 1 -> la relation irela est le doublon d'une relation précédente.  
          Il ne faut pas la prendre en compte
```