# Structure of data sd_interf_dyna_clas

**Summary:**

One describes the structure of data here produced by the order DEFI_INTERF_DYNA.
It defines the interfaces associated with a given macronutrient.
A macronutrient can comprise several interfaces.

# Contents

1 General information....3
2 Tree structure of the Structure of Data....3
3 Contents of the objects JEVEUX....3
3.1 General information....3
3.2 Object .IDC_REFE....3
3.3 Object.IDC_DESC....4
3.4 Object .IDC_NOMS....4
3.5 Object.IDC_TYPE....4
3.6 Object .IDC_LINO....4
3.7 Object .IDC_DDAC....4
3.8 Object .IDC_DY_FREQ....4
3.9 Object.IDC_DEFO....5

# 1 General information

The structure of data sd_interf_dyna_clas is exclusively produced by the order DEFI_INTERF_DYNA . This one defines the interfaces of a structure for the modal recombination or the dynamic under-structuring by modal synthesis.

# 2 Tree structure of the Structure of Data

sd_interf_dyna_clas (K8)

- '.IDC_DDAC' : OJB XD V I NUM () VARI
- '.IDC_DEFO' : OJB S V I
- '.IDC_DESC' : OJB S V I LONG=5
- '.IDC_LINO' : OJB XD V I NUM () VARI
- '.IDC_NOMS' : OJB S NR K8
- '.IDC_REFE' : OJB S V K24 LONG=3
- '.IDC_TYPE' : OJB S V K8
- '.IDC_DY_FREQ' : OJB S V R LONG=1

# 3 Contents of the objects JEVEUX

## 3.1 General information

One sd_interf_dyna_clas contains one or more interfaces ( nb_intf ) named.

Each interface contains one or more nodes. Each node of interface carries a certain number of ddls (component of the size DEPL_R ). For each node of interface, the user chooses a subset of the ddls carried by this node: they are the “active” ddls of the node.

For each ddl active of sd_interf_dyna_clas , one will calculate (later in DEFI_BASE_MODALE ) a static deformation.

One is interested in the size DEPL_R .

## 3.2 Object .IDC_REFE

'.IDC_REFE' : S V K24 LONG = 3

| | |
|---|---|
| V (1) | Name of sd_maillage |
| V (2) | Name of sd_nume_ddl |
| V (3) | vacuum |

## 3.3 Object.IDC_DESC

'.IDC_DESC' : S V I LENGTH = 5

| | |
|---|---|
| V (1) | 1 |
| V (2) | Many coded entireties necessary to the size DEPL_R ( nbec ) |
| V (3) | Component count maximum for the size DEPL_R |
| V (4) | Number of the size DEPL_R in the catalogue of the sizes |
| V (5) | Many static deformations to calculate ( nb_def ) |

## 3.4 Object .IDC_NOMS

'.IDC_NOMS' : S NR LONG K8 = nb_intf

It is the pointer of names giving the correspondence number of the interface ↔ name of the interface

V (I): name of the interface number I

## 3.5 Object.IDC_TYPE

'.IDC_TYPE' : S V LONG K8 = nb_intf

V (1 with nb_intf) : type of the interface ( ' CRAIGB ', ' MNEAL ', ' CB_HARMO ' or ' NONE ')

## 3.6 Object .IDC_LINO

'.IDC_LINO' : XD V I NUM () VARI NB_OBJ = nb_intf

This collection comprises a number of elements equal to the numbers of interface ( nb_intf ).
That is to say V I ème object of the collection
V as a dimension the number of nodes of the interface number I have ( nbno ).

V (1 with nbno ) : number (in the grid) of the nodes of the interface number I

## 3.7 Object .IDC_DDAC

'.IDC_DDAC' : XD V I NUM () VARI NB_OBJ = nb_intf

This collection comprises a number of elements equal to the numbers of interface ( nb_intf )
That is to say V I ème object of the collection.
V as a dimension the number of nodes of the interface number I have ( nbno ) multiplied by the number of coded entirety necessary to the description of the size DEPL_R ( nbec ).

V (1 with nbno*nbec) : list of the coded entireties describing the active ddls of the interface number I

## 3.8 Object .IDC_DY_FREQ

'.IDC_DY_FREQ' : S V R LONG = 1

V (1) : Value of the frequency used for the calculation of the harmonic constrained modes

## 3.9 Object.IDC_DEFO

'.IDC_DEFO' : S V I LENGTH = (2+nbec) *nbnot

This object describes them ( nbnot ) nodes of sd_interf_dyna_clas and their ddls active. It describes also the classification of the static deformations (which one will calculate later) associated with the active ddls with sd_interf_dyna_clas .

The number of nodes of sd_interf_dyna_clas ( nbnot ) is possibly lower than the composing sum of the numbers of nodes of the various interfaces sd_interf_dyna_clas because interfaces of the same type ( ' CRAIGB ',... ) can have joint nodes (which "will then be amalgamated" in sd_interf_dyna_clas ).

Each node of sd_interf_dyna_clas a type has ' MNEAL ', ' CRAIGB ', ' CB_HARMO ', ' NONE '.

Nodes of sd_interf_dyna_clas are gathered per packages in the same way standard. These nodes are arranged in the order: ' MNEAL ', ' CRAIGB ', ' CB _ HARMO ', ' NONE '.

That is to say nmn , ncb , ncbh , nau : numbers of nodes of sd_interf_dyna_clas types: ' MNEAL ', ' CRAIGB ', ' CB_HARMO ', ' NONE '.

That is to say: nbnot = nmn + ncb + ncbh + nau

The vector . IDC_DEFO is made of 3 "blocks":

Numbers (in the grid) of the nodes:
V (ino) : number of ino ème node of sd_interf_dyna_clas

Numbers of the 1 eras deformations carried by each node:
V (nbnot+ino) : number of the 1 era deformation of suddenly of sd_interf_dyna_clas

This storage block is used very little in the code. One makes use of it (temporarily) in the order DEFI_INTERF_DYNA who creates the SD. Once the SD created, the final contents of this block is used only in the routine bmnodi.f ( REST_GENE_PHYS , MODE_ITER_CYCL )

Coded entireties describing the active ddls of the nodes:

V (2*nbnot+ nbec* (ino-1): 2*nbnot+ nbec* (ino)) : coded entireties describing the active ddls of the ino ème node of sd_interf_dyna_clas