

## Opérateur DEFI\_BASE\_MODALÉ

---

### 1 But

---

Le but de l'opérateur est de définir :

- la base d'une sous-structuration dynamique, quand on a découpé la structure complète en sous-structures ;
- ou la base d'une recombinaison modale, quand on traite directement la structure complète.

La base modale obtenue par cet opérateur est du type : 'CLASSIQUE' si la base modale est composée de modes propres dynamiques et des déformées statiques calculées par l'opérateur à partir d'un concept de type `interf_dyna_clas` produit par DEFI\_INTERF\_DYNA [U4.64.01]. L'option `DIAG_MASS` permet de recalculer une numérotation pour les modes statiques de telle sorte que la matrice de masse soit diagonale.

La base est de type `RITZ` dans les autres cas.

L'opérateur produit un concept de type `mode_meca`.

## Table des Matières

1 But.....	1
2 Syntaxe.....	3
3 Opérandes.....	4
3.1 Mot-clé facteur CLASSIQUE.....	4
3.1.1 Opérande INTERF_DYNA.....	4
3.1.2 Opérandes MODE_MECA / NMAX_MODE.....	4
3.2 Mot-clé facteur RITZ.....	4
3.2.1 Mot-clé MODE_MECA.....	4
3.2.2 Mot-clé MODE_INTF.....	5
3.2.3 Mot-clé BASE_MODAL.....	5
3.2.4 Opérande NMAX_MODE.....	5
3.2.5 Opérande INTERF_DYNA.....	5
3.2.6 Opérande NUME_REF.....	5
3.2.7 Opérande LIST_AMOR.....	5
3.2.8 Opérande ORTHO.....	5
3.2.9 Mot-clé MATRICE.....	5
3.3 Mot-clé facteur DIAG_MASS.....	5
3.3.1 Mot-clé MODE_MECA.....	6
3.3.2 Mot-clé MODE_STAT.....	6
3.4 Mot-clé facteur ORTHO_BASE.....	6
3.4.1 Mot-clé BASE.....	6
3.4.2 Mot-clé MATRICE.....	6
3.5 Mot-clé facteur SOLVEUR.....	6
3.6 Opérande TITRE.....	6
3.7 Opérande INFO.....	6
4 Exemples.....	6
4.1 Sous-structuration dynamique.....	7
4.2 Enrichissement d'une base de modes propres avec des modes statiques.....	7

## 2 Syntaxe

```
bamo [mode_meca] = DEFI_BASE_MODALÉ (

♦ / CLASSIQUE = _F(♦ INTERF_DYNA= intdyn, [interf_dyna_clas]
                  ♦ MODE_MECA = modes, [l_mode_meca]
                  ♦ NMAX_MODE = nbmodes, [l_I]
                  ),

/ RITZ = _F(♦ / MODE_MECA = modes [l_mode_meca]
            / MODE_INTF = modeintf, / [mode_meca]
            / BASE_MODALÉ= bamo, [mult_elas]
            / BASE_MODALÉ= bamo, [mode_meca]
            ♦ NMAX_MODE = nbmodes, [l_I]
            )

/ DIAG_MASS = _F(♦ MODE_MECA = modes, [l_mode_meca]
                 ♦ MODE_STAT = modesta, [mode_meca]
                 ),

/ ORTHO_BASE = _F(♦ BASE = modes, / [mode_meca]
                  ♦ MATRICE = matrice, / [mult_elas]
                  ♦ MATRICE = matrice, [matr_asse_*]
                  ),

♦ INTERF_DYNA = intdyn, [interf_dyna_clas]

♦ NUME_REF = numddl, [nume_ddl]

♦ LIST_AMOR = listamor, [l_R8]

♦ ORTHO = / 'OUI'
          / 'NON' [DEFAULT]
♦ MATRICE = matrice, [matr_asse_*]

♦ SOLVEUR =_F( voir document [U4.50.01])

♦ TITRE = titre, [l_Kn]

♦ INFO = / 1, [DEFAULT]
         / 2,

)
```

## 3 Opérandes

---

### 3.1 Mot-clé facteur `CLASSIQUE`

- ◆ / `CLASSIQUE = _F(`  
Mot-clé facteur pour la définition d'une base modale de type '`CLASSIQUE`', quand on réalise un calcul par sous-structuration dynamique.

#### 3.1.1 Opérande `INTERF_DYNA`

- ◆ `INTERF_DYNA = intdyn`  
Nom du concept de type `interf_dyna_clas` produit par `DEFI_INTERF_DYNA` [U4.64.01].

L'opérateur calcule les déformées statiques correspondant aux différentes interfaces définies, en s'appuyant sur la numérotation utilisée pour le calcul des modes propres.

#### 3.1.2 Opérandes `MODE_MECA` / `NMAX_MODE`

- ◇ `MODE_MECA = modes`  
Liste des concepts de type `mode_meca` contenant les modes propres de la structure.
- ◇ `NMAX_MODE = nbmodes`  
Liste d'entiers donnant, pour chaque concept `mode_meca`, le nombre maximal de modes propres à retenir. A partir de la  $i^{\text{ème}}$  base modale de la liste `modes`, on retient alors les modes correspondant aux `nbmodes[i]` fréquences les plus basses.

En général, la longueur de la liste d'entiers, renseignée sous le mot-clé `NMAX_MODE`, doit correspondre à celle de la liste des bases modales (mot-clé `MODE_MECA`). La seule exception à cette règle est quand une unique valeur est donnée dans `NMAX_MODE`. Dans ce cas, la valeur de `NMAX_MODE` est utilisée pour chacune des bases modales.

Si `NMAX_MODE` n'est pas renseigné, on prend par défaut tous les modes de chacune de bases modales données sous le mot-clé `MODE_MECA`.

### 3.2 Mot-clé facteur `RITZ`

- ◆ / `RITZ = _F(`  
Mot-clé facteur permettant :
  - dans le cas d'un calcul par sous-structuration, de construire une base modale de sous-structure de type '`RITZ`' ;
  - dans le cas d'un calcul directement avec la structure complète, de construire une base modale enrichie (par exemple : des modes propres de vibration, qu'on enrichit de modes statiques associés à des forces extérieures).

La base est constituée à partir de 2 occurrences du mot-clé `RITZ` (une avec le mot-clé `MODE_MECA` ou `BASE_MODAL`, une autre avec le mot-clé `MODE_INTF`).

#### 3.2.1 Mot-clé `MODE_MECA`

Nom du concept de type `mode_meca` contenant les modes propres de vibration de la structure, ou de la sous-structure dans le cas d'un calcul par sous-structuration.  
On peut donner une liste de `mode_meca` obtenus pour la même structure (avec des conditions aux limites différentes par exemple).

## 3.2.2 Mot-clé `MODE_INTF`

Nom du concept de type `mode_meca` (produit par `CALC_MODES` [U4.52.02] ou par `MODE_STATIQUE` [U4.52.14]) ou `mult_elas` (produit par `MACRO_ELAS_MULT` [U4.51.02]) contenant :

- dans le cas d'un calcul par sous-structuration : les modes qu'on veut utiliser comme modes d'interface de la sous-structure ;
- dans le cas d'un calcul directement sur la structure complète : des modes qui enrichissent la base de modes propres de vibration donnée sous le mot-clé `MODE_MECA` précédent. Cela peut par exemple être des déformées statiques associées à des forces extérieures (force imposée connue ; force de choc associée à un obstacle ; etc.).

## 3.2.3 Mot-clé `BASE_MODAL`

Nom de concept de type `mode_meca` produit par un appel précédent de l'opérateur de `DEFI_BASE_MODAL` [U4.64.02]. Il ne peut être entré que lors de la première occurrence du mot-clé `RITZ`. La seconde occurrence du mot-clé `RITZ` contiendra alors obligatoirement le mot-clé `MODE_INTF`. Le nom de concept `mode_meca` résultat de l'opérateur peut être différent de celui-ci ou identique (il est alors réentrant).

## 3.2.4 Opérande `NMAX_MODE`

Nombre de modes à retenir dans les modes dynamiques (ou statiques) donnés par un des mots-clés précédents sous l'occurrence du mot-clé `RITZ`. Si on renseigne une liste de `mode_meca`, il faut donner soit une liste de la même taille pour les nombres de modes à retenir, soit une valeur unique (dans ce cas, cette valeur est appliquée à tous les concepts `mode_meca`).

Si `NMAX_MODE` n'est pas renseigné, on prend par défaut tous les modes.

## 3.2.5 Opérande `INTERF_DYNA`

Interface dynamique de la sous-structure (à renseigner éventuellement et seulement si l'on utilise le mot-clé facteur '`RITZ`').

## 3.2.6 Opérande `NUME_REF`

Numérotation de référence sur laquelle tous les champs de déplacement (modes dynamiques et statiques) constituant la base de '`RITZ`' seront réordonnés.

## 3.2.7 Opérande `LIST_AMOR`

Liste des amortissements modaux que l'utilisateur peut fournir pour enrichir les modes déclarés sous le mot-clé `MODE_MECA`. Ça revient à ajouter des amortissements réduits pour ces modes mêmes si à l'origine ils sont des modes réels. Cette option est utile pour simuler des résultats expérimentaux.

## 3.2.8 Opérande `ORTHO`

Opérande permettant de faire une orthonormalisation de la base de Ritz (à renseigner si on souhaite cette ré-orthonormalisation et seulement si l'on utilise le mot-clé facteur '`RITZ`'). Cette orthonormalisation est faite avec un algorithme de type Gram-Schmidt itératif (IGS) suivant la version de Kahan-Parlett.

## 3.2.9 Mot-clé `MATRICE`

Nom du concept de type `matr_asse_*` qui sera pris en compte pour les produits scalaires lors de la réorthonormalisation de la base de `RITZ`. C'est un mot-clé obligatoire si `ORTHO='OUI'`.

## 3.3 Mot-clé facteur `DIAG_MASS`

♦ / `DIAG_MASS = _F(`

Mot-clé permettant de recalculer les modes statiques en éliminant la contribution dynamique et en procédant à une orthogonalisation de Graam-Schmidt.

### 3.3.1 Mot-clé **MODE\_MECA**

Nom du concept de type `mode_meca` contenant les modes propres dynamiques de la sous-structure traitée.

### 3.3.2 Mot-clé **MODE\_STAT**

Nom du concept de type `mode_meca` produit par l'opérateur `MODE_STATIQUE` [U4.52.14] qui contient les modes statiques.

## 3.4 Mot-clé facteur **ORTHO\_BASE**

◆ / `ORTHO_BASE = _F(`

Mot-clé permettant d'orthonormaliser les modes d'une base Cette orthonormalisation est faite avec un algorithme de type Graam-Schmidt itératif (IGS) suivant la version de Kahan-Parlett.

### 3.4.1 Mot-clé **BASE**

Nom du concept de type `mode_meca` ou `mult_elas` contenant des modes propres dynamiques.

### 3.4.2 Mot-clé **MATRICE**

Nom du concept de type `matr_asse_*` qui contient sera pris en compte pour les produits scalaires lors de la réorthogonalisation.

## 3.5 Mot-clé facteur **SOLVEUR**

◇ `SOLVEUR = _F(...)`

Ce mot-clé facteur est facultatif : il permet de choisir un autre solveur de résolution de système. Dans le cas de cette commande, la syntaxe du mot-clé est restreinte à deux méthodes: on peut choisir entre la méthode par défaut, `MULT_FRONT`, et les méthodes `LDLT` ou `MUMPS`. La syntaxe étant commune à plusieurs commandes, veuillez consulter le manuel [U4.50.01].

## 3.6 Opérande **TITRE**

◇ `TITRE = titre`

Titre du concept créé.

## 3.7 Opérande **INFO**

◇ `INFO =`

Niveau des informations fournies dans le fichier `MESSAGE` :

- 1 pas d'impression ;
- 2 écriture des généralités (concepts amont, type de base).

# 4 Exemples

## 4.1 Sous-structuration dynamique

Les différentes modélisations du cas-test SDLS106 présentent la mise en œuvre complète d'un calcul par sous-structuration, avec différentes méthodes (Craig-Bampton, Mc Neal, maillages compatibles ou non, etc.).

## 4.2 Enrichissement d'une base de modes propres avec des modes statiques

```
# calcul des matrices assemblées et de la numérotation des DDL
...

# calcul des modes propres de vibration
modesvib = CALC_MODES( MATR_RIGI=K_ASSE,
                      MATR_MASS=M_ASSE,
                      CALC_FREQ=_F( NMAX_FREQ = nb_freq) ,
                      ) ;

# application d'une force extérieure
force = AFFE_CHAR_MECA( MODELE=modele,
                      FORCE_NODALE=_F( NOEUD = 'N11',
                                      FX=800., FY = -1000.)) ;

# calcul de la déformée statique associée à cette force extérieure
modestat = MODE_STATIQUE( MATR_RIGI=K_ASSE,
                          MATR_MASS=M_ASSE,
                          FORCE_NODALE=_F( NOEUD='N11',
                                          AVEC_CMP=('DX','DY') ),
                          ) ;

# construction de la base modale enrichie
basemode = DEFI_BASE_MODALÉ( RITZ =( _F( MODE_MECA=modesvib ),
                                   _F( MODE_INTF=modestat ) ),
                             ),
          NUME_REF=NUMDDL,
          ) ;
```

Le cas-test SDNL104c présente la mise en œuvre complète de cette technique.

Un autre exemple est donné par le cas-test SDNL301a, où les déformées statiques qui enrichissent la base des modes propres, sont celles liées à des forces extérieures engendrées par des chocs.