Macro-commande ASSEMBLAGE

1 But

Calculer une ou plusieurs matrices assemblées (rigidité, masse, amortissement …) ainsi que un ou plusieurs vecteurs assemblés.

Cette commande est recommandée pour créer les concepts nécessaires aux calculs dynamiques, qu'il s'agisse des matrices ou des vecteurs. Elle évite de fastidieux appels aux commandes "de base" : CALC_MATR_ELEM [U4.41.01], NUME_DDL [U4.61.01], ASSE_MATRICE [U4.61.22], CALC_VECT_ELEM [U4.61.02], ASSE_VECTEUR [U4.61.23].

Produit une ou plusieurs structures de données de type matr_asse_* et cham_no plus, éventuellement, une structure de données de type nume_ddl.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Syntaxe

```
ASSEMBLAGE    (
♦  MODELE     =  mo ,                                   [modele]
◊  CHAM_MATER    =  chmater,                            [cham_mater]
◊  CARA_ELEM =  carac  ,                                [cara_elem]
◊  CHARGE     =  l_char ,                          /    [l_char_meca]
                                                   /    [l_char_ther]
                                                   /    [l_char_acou]
   ◊  INST       =  /  tps,                             [R]
                    /  0. ,                             [DEFAUT]
◊  CHAR_CINE     =  l_chci ,                            [l_char_cine]
♦  NUME_DDL      =  nu  ,                               [nume_ddl]
```

♦ # assemblage des matrices :

```
   /  MATR_ASSE = (_F  (
      ♦  MATRICE    =  matas  ,                        [matr_asse_*]
      ♦  OPTION     =
                    # phénomène mécanique :
                       /  'RIGI_MECA' ,
                          ◊  MODE_FOURIER =  /  nh ,   [I]
                                             /  0  ,   [DEFAUT]
                       /  'RIGI_GEOM' ,
                          ♦  SIEF_ELGA =        sig,   [cham_elem_SIEF_R]
                          ◊  MODE_FOURIER =  /  nh ,   [I]
                                             /  0  ,   [DEFAUT]
                       /  'RIGI_ROTA' ,
                       /  'IMPE_MECA' ,
                       /  'ONDE_FLUI' ,
                       /  'RIGI_MECA_HYST' ,
                       /  'RIGI_FLUI_STRU' ,
                       /  'AMOR_MECA' ,
                       /  'MASS_FLUI_STRU' ,
                       /  'MASS_MECA' ,
                       /  'MASS_MECA_DIAG' ,
                    # Gyroscopie:
                        / 'MECA_GYRO' ,
                       /  'RIGI_GYRO' ,
                          ◊  GROUP_MA =   grma

                    # phénomène thermique :
                       /  'RIGI_THER' ,
                          ◊  MODE_FOURIER =  /  nh ,   [I]
                                             /  0  ,   [DEFAUT]
                       /  'RIGI_THER_CONV' ,
                       /  'RIGI_THER_CONV_D' ,

                    # phénomène acoustique :
                       /  'RIGI_ACOU' ,
                       /  'MASS_ACOU' ,
                       /  'AMOR_ACOU' ,
                    ),),
```

assemblage des vecteurs :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
/   VECT_ASSE = (_F  (
      ♦  VECTEUR   =  vectas ,                          [cham_no]
      ♦  OPTION    =
                # phénomène mécanique :
                      /  'CHAR_MECA' ,
                         ◊  CHARGE    =  l_char_v, /   [l_char_meca]
                         ◊  MODE_FOURIER =  /  nh  ,   [I]
                                            /  0   ,   [DEFAUT]
                # phénomène thermique :
                      /  'CHAR_THER' ,
                         ◊  CHARGE    =  l_char_v, /   [l_char_ther]

                # phénomène acoustique :
                      /  'CHAR_ACOU' ,
                         ◊  CHARGE    =  l_char_v, /   [l_char_acou]
                   ),),
◊  TITRE     =  titr   ,                                [l_Kn]

◊  INFO      =  /1                                      [DEFAUT]
                /2
             )
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Opérandes

3.1 Concepts créés par la macro-commande

Cette macro-commande crée plusieurs concepts typés :

- une matr_asse_* à chaque occurrence du mot clé facteur MATR_ASSE (mot clé MATRICE),
- une cham_no à chaque occurrence du mot clé facteur VECT_ASSE (mot clé VECTEUR),
- éventuellement un nume_ddl (mot clé NUME_DDL) si celui-ci n'existe pas au préalable.

Soit par exemple la séquence :

```
...
# nu n'est pas un concept encore existant.

ASSEMBLAGE    (MODELE = mo , ...
                 NUME_DDL = CO( " nu " ),
                 MATR_ASSE = ( _F( MATRICE= CO("K"), OPTION='RIGI_MECA'),
                 VECT_ASSE = ( _F( VECTEUR= CO("F"), OPTION='CHAR_MECA'))

ASSEMBLAGE    (MODELE = mo , ...
                 NUME_DDL = nu,
                 MATR_ASSE = (_F(MATRICE=CO( "M") ,OPTION='MASS_MECA'),
                              _F(MATRICE=CO( "A") ,OPTION='AMOR_MECA'),
),)
```

le 1er appel à ASSEMBLAGE crée 1 MATR_ASSE : K
1 VECT_ASSE : F
1 NUME_DDL : nu
le 2ème appel à ASSEMBLAGE crée 2 MATR_ASSE : M et A

Remarque :

Lors du 2 ème appel, le concept nume_ddl est fourni (il existe, il a été créé par le 1 er appel). Dans ce cas, il est supposé convenir aux 2 matr_asse à créer ; il est donc important que le nume_ddl soit d'abord calculé pour une option de type " RIGIDITE " pour contenir les nœuds de Lagrange des charges de blocages (cf. CALC_MATR_ELEM [U4.61.01]).

Dans l'exemple précédent, les 3 matr_asse K , M et A et le vecteur F s'appuient sur la même numérotation des inconnues (nu). C'est une condition nécessaire pour les algorithmes de recherche de modes propres.

3.2 Opérandes MODELE / CHAM_MATER / CARA_ELEM / INST

♦ MODELE = mo

Cet opérande sert à indiquer les éléments pour lesquels doivent être effectués les calculs élémentaires : on rappelle que les éléments finis sont pour la plupart définis dans le modèle.

Il y a quelques exceptions :

1) Les éléments de dualisation des conditions de DIRICHLET, c'est-à-dire les éléments permettant d'imposer des conditions sur les degrés de liberté de déplacement en mécanique, les degrés de liberté de température en thermique et les degrés de liberté de pression en acoustique.
2) Les éléments de chargement nodal, d'échange thermique entre parois, ...

Ces éléments sont définis dans les concepts de type char_meca, char_ther ou char_acou.

On doit donc fournir l'argument l_char pour le calcul des matrices assemblées de rigidité : RIGI_MECA, RIGI_THER, RIGI_ACOU, RIGI_MECA_HYST et RIGI_THER_CONV(_D).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

◊ CHAM_MATER = chmater

Nom du champ de matériau où sont définies les caractéristiques des matériaux des éléments.

Cet argument est presque toujours nécessaire.

En pratique, on peut s'en passer :

- pour les éléments discrets dont les matrices élémentaires sont définies dans le concept cara_elem. Voir AFFE_CARA_ELEM [U4.42.01],
- pour le calcul des rigidités dues à la dualisation des conditions aux limites.

◊ CARA_ELEM = carac

Les caractéristiques élémentaires carac sont nécessaires s'il existe dans le modèle des éléments de structure (poutre, coque ou éléments discrets) ou des éléments de milieu continu non isotropes.

◊ INST = tps

L'argument tps est utilisé lorsque les caractéristiques matérielles ou les chargements dépendent du temps. Un cas assez fréquent est celui d'un matériau mécanique dépendant de la température qui elle même dépend du temps.

3.3 Opérandes CHARGE

◊ CHARGE = l_char

Cet opérande a plusieurs fonctions distinctes :

1) préciser les éléments pour lesquels sont faits les calculs élémentaires de rigidité (conditions de DIRICHLET),
2) pour les options 'RIGI_THER_CONV(_D)' : donner la valeur de la vitesse de convection,
3) pour l'option 'IMPE_MECA' : donner la valeur de l'impédance acoustique des mailles du bord,
4) pour l'option 'ONDE_FLUI' : donner la valeur de la pression de l'onde incidente,
5) pour l'option 'RIGI_ROTA' : donner la valeur de la rotation imposée au modèle.

Remarque :

Lors d'un appel à la macro-commande pour l'assemblage des vecteurs dans le cas où le concept nu est entrant (sous le mot-clé NUME_DDL), la valeur du mot clé CHARGE l_char doit être identique à celle utilisée pour la création de nu.

Soit par exemple la séquence suivante :

```
...
# nu n'est pas un concept encore existant.

ASSEMBLAGE    (MODELE = mo , CHARGE = (char_1,char_2),...
               NUME_DDL = CO( " nu " ),
               MATR_ASSE = ( _F( MATRICE= CO("K"), OPTION='RIGI_MECA'))

# nu existe et est un concept entrant.

ASSEMBLAGE    (MODELE = mo , CHARGE = (char_1,char_2),...
               NUME_DDL = nu,
               VECT_ASSE = (   _F(VECTEUR=CO( "F") ,OPTION='CHAR_MECA',
                                  CHARGE=CHAR_F),),)
```

3.4 Opérande CHAR_CINE

◊ CHAR_CINE = lchci

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Cet opérande permet de ne pas "dualiser" certaines conditions aux limites de Dirichlet (blocages). Ceux-ci seront éliminés.

3.5 Opérande NUME_DDL

♦ NUME_DDL = nu

L'utilisateur de la macro doit toujours donner un nom de concept derrière le mot clé NUME_DDL. Si le concept n'existe pas, il est créé.

Si le concept existe, il est utilisé pour numéroter les matrices.

3.6 Mot clé MATR_ASSE

Ce mot clé facteur permet de demander le calcul d'une matrice assemblée (à chaque occurrence).

Le nom de la matrice assemblée est donné par l'opérande MATRICE et la "nature" de la matrice est définie par l'opérande **OPTION**.

Les options possibles : 'RIGI_MECA', 'MASS_MECA', … sont décrites dans [U4.61.01].

Le choix de l'option détermine le type de la matr_asse_* résultat [U4.61.01].

3.6.1 Opérandes particulières à certaines options

3.6.1.1 Opérande MODE_FOURIER

◊ MODE_FOURIER = nh

Entier positif ou nul indiquant l'harmonique de FOURIER sur laquelle on calcule les matrices.

Par défaut : nh = 0

3.6.1.2 Opérande SIEF_ELGA (option 'RIGI_GEOM')

♦ SIEF_ELGA = sig

Le champ de contraintes sig donné pour le calcul de l'option 'RIGI_GEOM' doit avoir été calculé avec l'option 'SIEF_ELGA' (champ de contraintes aux points de GAUSS des éléments) (cf. commandes CALC_CHAM_ELEM [U4.81.03] ou CALC_CHAMP [U4.81.04]).

3.6.1.3 Opérande GROUP_MA (options 'MECA_GYRO' et 'RIGI_GYRO')

◊ GROUP_MA = grma

On peut restreindre le calcul des options 'MECA_GYRO' et 'RIGI_GYRO' sur les mailles spécifiées par GROUP_MA .

3.7 Mot clé VECT_ASSE

Ce mot clé facteur permet de demander le calcul d'un vecteur assemblé (à chaque occurrence).

Le nom du vecteur assemblée est donné par l'opérande VECTEUR.

Dans les options possibles de calcul élémentaire et d'assemblage de vecteurs, seules certaines possibilités parmi toutes celles proposées dans les commandes CALC_VECT_ELEM et ASSE_VECTEUR sont proposées ici: 'CHAR_MECA' pour les phénomènes mécaniques, 'CHAR_THER' et 'CHAR_ACOU' pour les phénomènes thermiques et acoustiques respectivement.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

L'ensemble d'options et opérandes associées sont décrites dans [u4.61.02] et [u4.61.23].

Remarque :

L'option 'FORC_NODA' et la prise en compte d'un modèle contenant des sous-structures sous l'option 'CHAR_MECA', souvent utilisée dans le cas de la statique, ne sont pas disponibles dans cette macro-commande.

3.7.1 Opérandes particulières à certaines options

3.7.1.1 Opérande MODE_FOURIER

◊ MODE_FOURIER = nh

Entier positif ou nul indiquant l'harmonique de FOURIER sur laquelle on calcule les vecteurs dans le cas des phénomènes mécaniques.

Par défaut : nh = 0

3.7.1.2 Opérande CHARGE

◊ CHARGE = l_char_v

Cette opérande permet d'accéder aux informations de chargement spécifiques à chaque vecteur. Les valeurs de l_char_v complètent (par addition) les charges l_char spécifiées sous le mot-clé CHARGE en dehors du mot clé VECT_ASSE.

Remarque :

Les doublons dans les charges données dans l_char et dans l_char v sont interdits.

Exemple pour le traitement des chargements :

```
ASSEMBLAGE    (MODELE = mo , …
              CHARGE = (char _1,char_2) ,
              NUME_DDL = CO( " nu " ),
              MATR_ASSE = ( _F( MATRICE= CO("K"), OPTION='RIGI_MECA'),
              VECT_ASSE = ( _F( VECTEUR= CO("F1"),
                                OPTION='CHAR_MECA',CHARGE=char_A ),
                            _F( VECTEUR= CO("F2"),
                                OPTION='CHAR_MECA',CHARGE=char_B ),
                            _F( VECTEUR= CO("F3"),OPTION='CHAR_MECA'),)
```

Dans l'exemple ci-dessus, les charges assemblées pour chaque vecteur sont :

Vecteur	Charges assemblées
F1	char_1, char_2, char_A
F2	char_1, char_2, char_B
F3	char_1, char_2

3.8 Opérande TITRE

◊ TITRE

Titre que l'on veut donner au résultat [U4.03.01].

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

4 Exemples

4.1 Calcul de modes propres mécaniques

```
ASSEMBLAGE    (  MODELE = mo   ,      CHARGE = ch_blocage ,
                 CHAM_MATER = chmat  ,  CARA_ELEM = carac ,
                 NUME_DDL = CO( "nu")    ,
       MATR_ASSE = (_F  (MATRICE = CO( "K") , OPTION = 'RIGI_MECA'),
                     _F  (MATRICE = CO( "M") , OPTION = 'MASS_MECA'),),)

modes = CALC_MODES ( MATR_RIGI = K,
                     MATR_MASS = M,
                     ...)
```

4.2 Calcul de modes de flambement d'Euler

`# chsief` est le champ de contraintes lié au chargement

```
ASSEMBLAGE    (  MODELE = mo   ,      CHARGE = ch_blocage ,
                 CHAM_MATER = chmat  ,  CARA_ELEM = carac ,
                 NUME_DDL = CO( "nu") ,
        MATR_ASSE = (  _F(MATRICE= CO("K"), OPTION='RIGI_MECA'),
                       _F(MATRICE= CO( "KG"), OPTION='RIGI_GEOM',
                       SIEF_ELGA=chsief),),)

modflamb = CALC_MODES ( MATR_RIGI = K,
                        MATR_MASS = KG,
                        …)
```

4.3 Calcul de réponse forcée

```
ASSEMBLAGE    (  MODELE = mo   ,      CHARGE = ch_blocage ,
                 CHAM_MATER = chmat  ,  CARA_ELEM = carac ,
                 NUME_DDL = CO( "nu")    ,
       MATR_ASSE = (_F  (MATRICE = CO( "K") , OPTION = 'RIGI_MECA'),
                     _F  (MATRICE = CO( "M") , OPTION = 'MASS_MECA'),),
       VECT_ASSE = ( _F( VECTEUR= CO("F"),
                       OPTION='CHAR_MECA',CHARGE=char_A ),)

reponse = DYNA_VIBRA ( MATR_RIGI = K,   MATR_MASS = M,
                       EXCIT = _F( VECT_ASSE = F, ...)...)
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)