# Macro-command MACRO_EXPANS

## 1 Drank

macro-command MACRO_EXPANS makes it possible to carry out the expansion of experimental data on a digital model from a base of expansion. It consists of the succession of operators EXTR_MODE, PROJ_MESU_MODAL, REST_GENE_PHYS, and PROJ_CHAMP.

# 2 Syntax

```
MACRO_EXPANS (

        ♦ MODELE_CALCUL = _F (♦ = modelnum MODELS,        [modele_sdaster]

                              ♦ BASE = bases,
[mode_meca]

                              ◊ NUME_MODE  = numerical
[l_I]
                              ◊ NUME_ORDRE = numord                [l_I]
                             )

        ♦ MODELE_MESURE = _F ( ♦ MODELS = modelexp,
[modele_sdaster]

                              ♦ MESURE = my,                /  [mode_meca]
                                                             /[dyna_harmo]

                              ◊ NOM_CHAM =/"DEPL"               [DEFAULT]
                                          /"QUICKLY"
                                          /"ACCE"
                                          /"SIEF_NOEU"
                                          /"EPSI_NOEU"

                              ◊ NUME_MODE = numerical
[l_I]
                              ◊ NUME_ORDRE = numord                [l_I]
                             )

        ◊ RESOLUTION = _F (◊ METHODE =/"READ"
[DEFAULT]
                                      /"SVD"

        # If METHODE = "SVD" then :
                           ◊ EPS =/0.0                          [DEFAULT]
                                  /eps                             [R]

                           ◊ REGUL =/"NON"                      [DEFAULT]
                                    /"NORM_MIN"
                                    /"TIK_RELA"

        # If   REGUL! = "NON" then :
                           ◊/COEF_PONDER = /0.                  [DEFAULT]
                                           /W                      [l_R]
                             /COEF_PONDER_F = w_f            [l_fonction]
                           ),

        ◊ NUME_DDL = num_ddl,                                    [nume_ddl]

        ◊ RESU_NX = res_nx,                                     [mode_meca]

        ◊ RESU_EX = res_ex,                                  /  [mode_meca]
                                                            /[dyna_harmo]
```

```
◊ RESU_ET = res_et,                                  /  [mode_meca]
                                                     /[dyna_harmo]

◊ RESU_RD = res_rd,                                  /  [mode_meca]
                                                     /[dyna_harmo]

)
```

# 3 Operands

## 3.1 Key word MODELE_CALCUL

♦ MODELE_CALCUL

Factor key word gathering all the key words relating to the base of expansion, in general obtained by computation (from where the name).

### 3.1.1 Key word MODELS

♦ MODELS = modelnum
*modele_sdaster* indicating the model on which one will extend measurement

### 3.1.2 Key word BASE

♦ BASE = bases
*as a basis* mode_meca being used for the expansion.
The base should not have vectors colinéaires, and the number of modes used must be lower than the number of DDL of measurement (preferably, nmodes << nmes) or else, the system to be solved is under-given, which can lead to a fatal error, and a stop of the code.

### 3.1.3 Key word NUME_ORDRE/NUMÉRIQUE_MODE

Lists sequence numbers or modal positions of the modes which one wishes to use for the expansion.

## 3.2 Key word MODELE_MESURE

♦ MODELE_MESURE

Factor key word gathering all the key words relating to the experimental base that one wishes to extend

### 3.2.1 Key word MODELS

♦ MODELS = modelexp
*modele_sdaster* indicating the model associated with the experimental mesh. The knowledge of the nodes is in general enough to determine an experimental mesh. The model associated can be then defined in the following way:

```
MODELEXP = AFFE_MODELE (MAILLAGE = MAIEXP,
                        AFFE = _F (GROUP_MA = "SENSORS",
                                   PHENOMENE = "MECHANICAL",
                                   MODELISATION = "DIS_T",),);

CAREXP = AFFE_CARA_ELEM (= MODELEXP, DISCRET
                         MODELS = _F (GROUP_MA = "SENSORS",
                                      LOCATES = "GLOBAL",
                                      CARA = "K_T_D_N",
                                      VALE = (100.0, 100.0, 100.0,),
                         ),);
```

The value of the stiffness given is arbitrary, it is not useful in computation.

NB: to use operator PROJ_CHAMP in the macro one, one needs to generate a nume_ddl associated with this mesh. For that, it is moreover necessary to assign one material to the model, to calculate the elementary matrixes (stiffness for example) and to create classification with NUME_DDL.

### 3.2.2 Key word MESURE

♦ MESURE = my
*dyna_harmo* or *mode_meca* to be extended. These data are in general imported of result of measurement (file .unv) with operator LIRE_RESU.

### 3.2.3 Key word NUME_ORDRE/NUMÉRIQUE_MODE

Lists integers. Allows to select the modes which one wishes to extend.

### 3.2.4 Key word NOM_CHAM

♦ . NOM_CHAM = "DEPL"…
Experimental quantity to extend.

## 3.3 Key word RESOLUTION

Two techniques of resolution are proposed for the inverse problems: SVD truncated, method READ.
For the SVD, one can choose to truncate the singular values smallest to improve conditioning of the problem (choice of "eps"), or to use a regularization of the Tikhonov type.
One will be able to refer to the documents [U4.73.01] (Doc. of PROJ_MESU_MODAL) and [R6.03.01] (Doc. of reference on decomposition in singular values).

## 3.4 Key word NUME_DDL

Makes it possible to force classification to be used for operator PROJ_CHAMP. For more accuracy, to refer to the documentation of PROJ_CHAMP [U4.72.05].

## 3.5 Key words RESU_XX

key words RESU_XX allow pre-délcarer names of the outgoing concepts:

- RESU_NX is the truncation of the numerical base (key word BASE under factor key word the MODELE_CALCUL) to the modes chosen in NUME_MODE,
- RESU_EX is the truncation of the experimental base (key word MESURE under factor key word the MODELE_MESURE) to the modes chosen in NUME_MODE,
- RESU_ET is result of the expansion,
- RESU_RD is the reprojection on the model experimental wide basis: it is intéressaant to check if the reprojection of result wide is comparable to the initial experimental data.