Operator TO FACTORIZE

1 Goal

To factorize an assembled matrix or **to manufacture a matrix of pre-conditioning**.

The matrix produced by this operator is provided to the operator TO SOLVE[U4.55.02] to solve the linear systems.

This operator allows:
1. That is to say to factorize the matrix assembled (in a product of two triangular matrices) for the direct methods,
2. That is to say to build a matrix of pre conditioning for the iterative methods with an aim of accelerating convergence at the time of the resolution.

This order is an obligatory precondition before calling the ordering of resolution TO SOLVE.

Product (or enriches) a structure of data of the type matr_asse.

2 Syntax

```
chechmate [matr_asse_*] =  TO FACTORIZE

          (  # Obligatory (except if GCPC + LDLT_INC: prohibited then)
             ◊  reuse  =  chechmate,

             ♦  MATR_ASSE =  chechmate ,
                 # If method MULT_FRONT ,MUMPS ,LDLT :
                                          /        [matr_asse_DEPL_R]
                                          /        [matr_asse_DEPL_C]
                                          /        [matr_asse_TEMP_R]
                                          /        [matr_asse_TEMP_C]
                                          /        [matr_asse_PRES_R]
                                          /        [matr_asse_PRES_C]

                 # If method GCPC  or PETSC :
                                          /        [matr_asse_DEPL_R]
                                          /        [matr_asse_TEMP_R]
                                          /        [matr_asse_PRES_R]

                 # If method MULT_FRONT :
                   ◊  STOP_SINGULIER   =      /   'YES', [DEFECT]
                                              /   'NOT'.
                   ◊  RENUM  =                /   'MONGREL',   [DEFECT]
                                              /   'MDA',
                                              /   'MANDELEVIUM'
                   ◊  NPREC  =  /  nprec  ,           [I]
                                /  8.                 [DEFECT]

                 # If method MUMPS:  [DEFECT]
                   ◊  RENUM=          /   'CAR', [DEFECT]
                                      /   'AMD',
                                      /   'MFA',
                                      /   'QAMD',
                                      /   'PORD',
                                      /   'MONGREL',
                                     /    'SCOTCH TAPE'
                    ◊  STOP_SINGULIER   =  /   'YES'  ,  [DEFECT]
                                           /   'NOT'  ,
                    ◊  TYPE_RESOL       =  /   'CAR'  ,  [DEFECT]
                                           /   'NONSYM',
                                           /   'SYMDEF',
                                           /   'SYMGEM'.
                    ◊  PCENT_PIVOT      =  /   20  ,     [DEFECT]
                                           /  pcpiv.    [R]
                    ◊  PRETREATMENTS=  /   'CAR'  ,  [DEFECT]
                                           /   'WITHOUT'.
                    ◊  ELIM_LAGR       =   /   'LAGR2'   ,  [DEFECT]
                                           /   'NOT'.
                    ◊  GESTION_MEMOIRE  =  /   'IN_CORE' ,[DEFECT]
                                           /   'OUT_OF_CORE',
                                           /   'EVAL'.
#Parameters limited to certain versions of MUMPS
                ◊  ACCELERATION     =      /'CAR',          [DEFECT]
                                         /  /'FR',
```

```
                                        /   /'FR+',
                                        /   /'LR',
                                        /   /'LR+'

                  ◊ LOW_RANK_SEUIL=      / 0.0                    [DEFECT]
                                         / lr_seuil   [R]
                # If method GCPC or PETSC :
                  ◊ / PRE_COND    = /   'LDLT_INC',               [DEFECT]
                          ◊  NIVE_REMPLISSAGE =  /  0,             [DEFECT]
                                                 /  niv [ I ]
                          ◊  FILLING       = /  1. ,               [DEFECT]
                                                 /  Cr.       [R]
                     /  PRE_COND   = /   'LDLT_SP',
                                     /   'LDLT_DP',
                          ◊  PCENT_PIVOT     =  /  20,             [DEFECT]
                                                 /  pcent [R]
                          ◊  REAC_PRECOND    =  /  30,             [DEFECT]
                                                 /  reactionary, [I]
                          ◊  GESTION_MEMOIRE =  /  'CAR',    [DEFECT]
                                                  / 'IN_CORE',
                          ◊  LOW_RANK_SEUIL  =  /  1.E-08        [DEFECT]
                                                 /   lr_seuil [R]
                          ◊   RENUM    =        /   ' WITHOUT '
[DEFECT]
                                                 /   ' PARMETIS '
                  ◊ / PRE_COND    =  / 'JACOBI',
                                     /  'SOR',
                # If method LDLT :
                        ◊  STOP_SINGULIER  =  /  'YES' ,  [DEFECT]
                                              / 'NOT' ,
                        ◊  NPREC           =  /  nprec .  [I]
                                              /  8     ,  [DEFECT]
                        ◊  /  BLOC_DEBUT   =  comic ,        [I]
                           /  DDL_DEBUT    =  dd  ,          [I]
                        ◊  /  BLOC_FIN     =  bf ,           [I]
                           /  DDL_FIN      =  df. ,          [I]

                  ◊  TITLE  =  title  ,                      [l_K80]

                  ◊  INFORMATION  =  /  1  ,                 [DEFECT]
                                     /  2  .

              )


if MATR_ASSE:    [matr_asse_DEPL_R]  then   [*]    - >DEPL_R
                    [matr_asse_DEPL_C]                 DEPL_C
                    [matr_asse_TEMP_R]                 TEMP_R
                    [matr_asse_TEMP_C]                 TEMP_C
                    [matr_asse_PRES_R]                 PRES_R
                    [matr_asse_PRES_C]                 PRES_C
```

3 Operands

The choice of the method of resolution is made as a preliminary in the order NUME_DDL (keyword METHOD).

Five methods are possible: 'MULT_FRONT', 'MUMPS', 'GCPC', 'PETSC' and 'LDLT'.

3.1 Keyword reuse = subdued

By preoccupation with a simplification of the programming and also to gain disk space, factorization "in place" of the matrix (use of reuse) is in general obligatory.

The exception is the couple ('GCPC','LDLT_INC'). In this case, it is interdict to use reuse.

3.2 Operand MATR_ASSE

♦ MATR_ASSE = chechmate

Name of the matrix assembled to factorize or with pre conditioning according to the method.

For the methods 'LDLT', 'MULT_FRONT' and 'MUMPS', this matrix can be real or complex, symmetrical or not. On the other hand for methods 'GCPC' and 'PETSC', this matrix must be real. For 'GCPC', the matrix must also be symmetrical.

3.3 Operands STOP_SINGULIER, NPREC, TYPE_RESOL, PCENT_PIVOT, PRETREATMENTS, GESTION_MEMOIRE, ACCELERATION, LOW_RANK_SEUIL, PRE_COND, NIVE_REMPLISSAGE, FILLING, RENUM, REAC_PRECOND and ELIM_LAGR

These keywords are described in [U4.50.01].

3.4 Operand TITLE

◊ TITLE = title

Title which one wants to give to the result [U4.02.01].

3.5 Operand INFORMATION

◊ INFORMATION =

1 : no impression

3.6 Partial factorization (method LDLT)

For the method 'LDLT', the operator allows to factorize only partially the matrix. This possibility makes it possible to factorize the matrix in several times (several work) to even modify with the flight the last lines of this factorized. Today, this functionality presents little interest except for certain

methods (known as discrete) of contact-friction where one has, intentionally, placed in the last lines of the matrix the terms concerning the nodes likely to be in contact. Thus, as iterations of pairing, the relations between these nodes changing, one erases then recomputes that these last contributions of factorized. It is a typical example where the astute use of an algorithm enough frustrates little to bring major profits (in time).

◊ / BLOC_DEBUT = comic

comic : partial factorization since comic- ième block included.

/ DDL_DEBUT = dd

dd: partial factorization since dd- ième equation included (in internal classification established by the operator NUME_DDL [U4.61.11]).

◊ / BLOC_FIN = bf

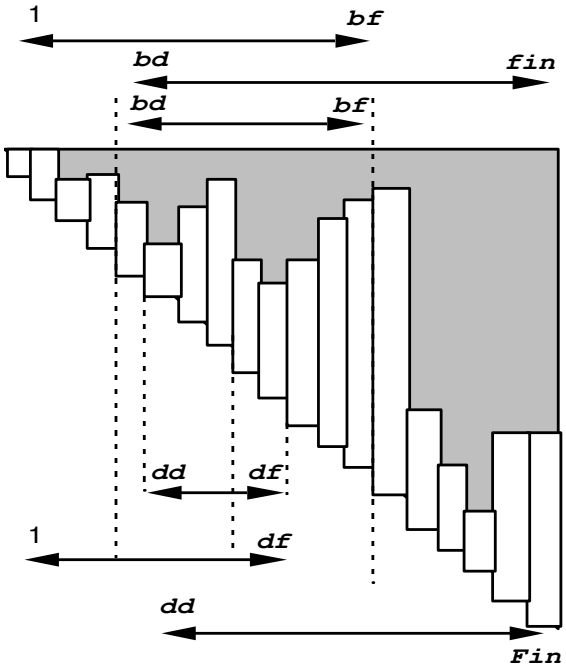
bf: partial factorization until bf- ième block included.

/ DDL_FIN = df

df: partial factorization until df- ième equation included (in internal classification established by the operator NUME_DDL [U4.61.11]).

BLOC_DEBUT and DDL_DEBUT

- in the absence of the keywords BLOC_DEBUT and DDL_DEBUT, the matrix will be factorized starting from its first line.
- if the argument comic keyword BLOC_DEBUT is negative or null, the matrix will be factorized starting from the first block. If not, one carries out a partial factorization from comic- ième block included.
- if the argument dd keyword DDL_DEBUT is negative or null, the matrix will be factorized starting from the first equation. If not, one carries out a partial factorization from dd- ième equation included.



4 Examples

See the examples in the documentation of the order TO SOLVE [U4.55.02].