Operator `AFFE_MATERIAU`

1 Drank

To assign materials to geometrical zones of a mesh.

To define the command variables (temperature, space, hydration, drying, corrosion,…) for numerical computations.

Product a data structure of the cham_mater `type`.

2 Syntax

```
chm [cham_mater] = AFFE_MATERIAU

   (  ♦MAILLAGE  = my  ,                  /            [mesh]
                                          /            [squelette]
      ◊MODELE       = Mo   ,                               [model]

      # assignment of the name of the material  :
      ♦AFFE   =   (_F (
                        ♦/TOUT          =  `OUI' ,
                           /  | GROUP_MA =lgma    ,       [l_gr_maille]
                              | NET   =lma        ,       [l_maille]

                        ♦MATER  =/mat        ,            [to subdue]
                                   /l_mat ,               [l_mater]
                  ),),

      # assignment of the command variables:
      ♦AFFE_VARC    =   (_F (
                        ◊/TOUT          =  `OUI' ,      [DEFAULT]
                           /  |  NET   =lma       ,       [l_maille]
                              |  GROUP_MA =lgma   ,       [l_gr_maille]

                        ♦NOM_VARC  =      "TEMP",
                                        /  "GEOM",
                                        /  "CORR",
                                        /  "EPSA",
                                        /  "HYDR",
                                        /  "IRRA",
                                        /  "M_ACIER",
                                        /  "M_ZIRC",
                                        /  "NEUT1",
                                        /  "NEUT2",
                                        /  "PTOT",
                                        /  "DIVU",
                                        /  "SECH",

                        ◊CHAM_GD=chvarc                        [field]
                        ◊EVOL=evovarc                        [evol_sdaster]
                               ◊NOM_CHAM    = nosymb,          [TXM]
                              ◊FONC_INST    = finst,      [function]
                           ◊PROL_DROITE  =/"EXCLUDED",    [DEFAULT]
                                          /"CONSTANT",
                                          /"LINEAIRE",
                           ◊PROL_GAUCHE  =/"EXCLUDED",    [DEFAULT]
                                          /"CONSTANT",
                                          /"LINEAIRE",
                        # If NOM_VARC = "TEMP" (or "SECH"):
                           ♦VALE_REF  = vref,             [R]

                        # "hidden" key keys:
                        VARC_TEMP = _F (...),
                        VARC_GEOM = _F (...),
                        VARC_HYDR = _F (...),
                        VARC_SECH = _F (...),
                        …
```

```
                ),),

          # assignment of the “behavior” of the multifibre beams  :
           ◊  AFFE_COMPOR =    (_F (
                          ♦/TOUT         =  `OUI' ,       [DEFAULT]
                             /  |  GROUP_MA =lgma      ,    [l_gr_maille]
                                |  NET    =lma         ,    [l_maille]

                          ♦COMPOR    =compor         ,          [compor]
                ),),

          ◊INFO   =/1                                ,          [DEFAULT]
                                    /2
       )
```

3 General information

This command is used to affect the material characteristics on the finite elements of the model (even if they are meshes mesh which is really affected). These material characteristics are defined in the materials which one affects on meshes (key word MATER). Each material contains a certain number of parameters (Young modulus, density,…). These parameters can be related to certain variables. We will call these variables of the “command variables”.

Currently, the command variables used are:

- the temperature,
- space,
- the hydration,
- drying,
- the metallurgical phases,
- the irradiation,
- corrosion,
- …

All these command variables must be affected with command AFFE_MATERIAU (key word AFFE_VARC).

In the case of model for multifibre beams, it is also necessary to affect in this command the “behaviors” defined with command DEFI_COMPOR/MULTI_FIBER [U4.43.06].

4 Operands

4.1 MAILLAGE

♦MAILLAGE = my,

Name of the mesh (or the squelette) which one wants to affect by characteristics of material.

Note:

> *The operation of assignment is the same one for meshes of a squelette as for meshes of a mesh. In the continuation of the document, one will always say mesh to simplify.*
>
> *When one affects materials on meshes of a squelette, it is that one wants to calculate stresses (for example) on meshes of postprocessing (coarser).*

4.2 MODEL

◊MODELE = Mo,

Name of the model. This argument (optional) is only used to check that the meshes affected ones in the command are part of the model well.

4.3 Key word factor AFFE

the key word factor AFFE makes it possible to affect various materials on “pieces” of the mesh.

4.3.1 Notice concerning computations of fracture mechanics

In general, the material characteristics must be known on the finite elements modelling the “matter”: “voluminal” elements (or of structure). The finite elements of “skin” are there to apply boundary conditions and do not have to know the material properties of the subjacent matter. An exception exists for the computation of option CALC_K_G of operators . For these computations, the finite elements modelling the lips of crack must be affected by the same material as the "voluminal" elements subjacent.

4.3.2 Operands TOUT=' OUI', GROUP_MA, TOUT

NETS the key key, GROUP_MA and MESH makes it possible to indicate the group of meshes which will be affected.

If a mesh appears explicitly (or implicitly) in several occurrences of factor key word the AFFE, the rule of overload is observed: it is the last assignment which precedes [U2.01.08].

4.3.3 Operand MATER

♦MATER = mat,

Name of the material which one wants to affect.

In the general case, each mesh is affected only by one material. Sometimes, it is necessary to indicate a list of materials when the nonlinear structural mechanics behavior is obtained by the command DEFI_COMPOR [U4.43.06].

4.4 Key word AFFE_VARC

This key word factor makes it possible to affect fields of **command variables** on meshes of the mesh. This key word can be repeated. It is necessary to use several occurrences d'AFFE_VARC to be able to affect several different command variables. But one can also use several occurrences for only one variable. For example, on a mixed model (3D + beams), one can affect like temperature:
1 evol_ther calculated on the elements 3D
1 field of temperature (constant in time) on the beam elements.

A command variable is a scalar (real) which can influence the material behavior via the parameters which are functions (for example a parameter of DEFI_MATERIAU/ELAS_FO). A command variable is a field known **before** computation. This field can be variable in time.

The command variables were introduced above all for mechanical computations. The command variable most usual for mechanical computations is the temperature.

For thermal computations, it is possible to use one (or several) command variables, but that relates to only some very rare parameters:

```
SECH_NAPPE      /FONCTION
THER_FO         /LAMBDA
THER_FO         /RHO_CP
THER_COQUE_FO   /COND_LMM, COND_TMM,…
```

4.4.1 Operand NOM_VARC

♦NOM_VARC = nomvarc,

Name of the command variable which one wants to affect (TEMP, GEOM, IRRA, CORR, HYDR, SECH,…).

Meaning and role of the various variables:

TEMP	temperature
GEOM	directions of space
CORR	corrosion of steels
EPSA	unelastic strain
HYDR	hydration of concrete
IRRA	Irradiation
M_ACIER	metallurgical phases of metallurgical
steel	M_ZIRC phases of “
neutral	” zircaloy NEUT1 variable 1: the material coefficients of the materials according to a parameter “user” (see example 3 below) allows to vary
NEUT2	“neutral” variable 2 (as NEUT1)
SECH	drying of concrete
PTOT	Stagnation pressure of fluid in THM (chained resolution)
DIVU	Strain voluminal in THM (chained resolution)

Certain command variables are scalars. Others are “vectors” including several scalar components.
One gives in the table below the name of the components of command variables

TEMP	TEMP, TEMP_MIL, TEMP_SUP, TEMP_INF
GEOM	X, Y, Z
CORR	CORR
EPSA	EPSAXX, EPSAYY, EPSAZZ, EPSAXY, EPSAXZ, EPSAYZ
HYDR	HYDR
IRRA	IRRA
M_ACIER	FERRITE, PEARLITE, PBAINITE, PMARTENS, TAUSTE, TRANSF, TACIER
M_ZIRC	ALPHPUR, ALPHBETA, TZIRC, TEMPS
NEUT1	NEUT1
NEUT2	NEUT2
SECH	SECH
PTOT	PTOT
DIVU	DIVU

The command variable IRRA corresponds to a fluence, i.e. the intégale in the time of a neutron flux. It is used by several constitutive laws, in specific units:

- it must be expressed in DPA (displacement per atom) for model IRRAD3M (cf [R5.03.23]);
- it must be expressed in $10^{20} n/cm^2$ for models VISC_IRRA_LOG, GRAN_IRRA_LOG, LEMAITRE_IRRA (cf [R5.03.09]).

4.4.2 Operands TOUT=' OUI', GROUP_MA, NETS

These key keys make it possible to indicate meshes zone to be affected.

4.4.3 Operand CHAM_GD

This key word makes it possible to associate with the command variable nomvarc the field chvarc. This field is a field of realities (not functions). It is thus independent of time and will be used throughout transient computations.
If the values of the command variable are dependant on time, it is necessary to use key word EVOL (see below). Cham_elem ELGA are authorized only if they result from operator PROJ_CHAMP/ METHODE = "SOUS_POINT", to assign the values to the subpoints of Gauss of the elements at subpoints.

4.4.4 OpérandesEVOL , NOM_CHAM, FONC_INST, PROL_DROITE, PROL_GAUCHE

These keys key make it possible to associate with the command variable nomvarc the transient evovarc. Key word NOM_CHAM makes it possible to indicate the symbolic name of the fields of SD_résultat to be used. By default, the code chooses:

NOM_VARC	*NOM_CHAM*
TEMP	"TEMP"
GEOM	"GEOM"
CORR	"CORR"
EPSA	"EPSA_ELNO"
HYDR	"HYDR_ELNO"
IRRA	"IRRA"
M_ACIER	"META_ELNO"
M_ZIRC	"META_ELNO"
NEUT1	"NEUT"
NEUT2	"NEUT"
SECH	"TEMP"
PTOT	"DEPL"
DIVU	"EPSI"

the fields are real fields (neither complexes, nor functions).
Cham_elem ELGA are authorized only if they result from operator PROJ_CHAMP/ METHODE = "SOUS_POINT", to assign the values to the subpoints of Gauss of the elements at subpoints.

The key word FONC_INST = finst makes it possible to define a function (time) which is used as correspondence between the "time" of the evolution evovarc (t_evo) and the "time" of later computation (t_calc). The function can be a simple "translation" (to take account owing to the fact that the mechanical beginning of times of computation is different from the time of the beginning of thermal computation, but one can make more complicated, for example to impose a mechanical loading (thermal thermal expansion) "cyclic" by calculating one cycle of temperature. One will be able to consult the case test zzzz223a to illustrate the use of this key word.

Caution: The function finst is that which transforms t_calc into t_evo : t_evo = finst (t_calc)

key keys PROL_GAUCHE and PROL_DROITE makes it possible to specify if one can use the transient evovarc before time "min" of the transient (PROL_GAUCHE) and/or after time "max" of transient (PROL_DROITE).
The value "EXCLUDED" will cause an error if one seeks to use the transient apart from his field.
Value "CONSTANT" prolongs the transient by the values at time "min" (or "max").
Value "LINEAIRE" linearly prolongs the transient starting from the 2 first (or the last) points of the transient.

4.4.5 Operand VALE_REF

This key word makes it possible the command variable to define a value of "reference" for nomvarc when this one needs a value of reference.
Currently, only two command variables require a value of reference: "TEMP" and "SECH". For these two variables, key word VALE_REF is compulsory. For the other variables, this key word is prohibited.
For the command variable "TEMP" in the case of the shells, the reference temperature is supposed to be the same one for the 3 components. This is why it is returned only once.

```
♦VALE_REF  = Tref    (or c0)        [R]
```

4.4.5.1 Reference temperature (T_{ref}):

The reference temperature T_{ref} introduced behind key word VALE_REF is the temperature for which there is no thermal strain (cf [R4.08.01]).

If the thermal coefficient of thermal expansion α (of which the value is introduced into command DEFI_MATERIAU [U4.43.01]) does not depend on the temperature: $\varepsilon^{th}(T)=\alpha(T-T_{ref})$.

If the thermal coefficient of thermal expansion depends on the temperature the mathematical statement allowing the computation of the thermal strain differs according to the specification from the thermal coefficient of thermal expansion in command DEFI_MATERIAU :

- the values of the thermal coefficient of thermal expansion (introduced into DEFI_MATERIAU) were determined by tests of dilatometry carried out with the temperature T_{ref}.
  In this case, key word TEMP_DEF_ALPHA should not be specified in command DEFI_MATERIAU and the thermal strain is calculated by the statement:

  $$\varepsilon^{th}(T)=\alpha(T)(T-T_{ref}) \text{ and } \varepsilon^{th}(T_{ref})=0$$

  where $\alpha(T)$ is well informed under key word ALPHA (or ALPHA_*) in DEFI_MATERIAU.
- the values of the thermal coefficient of thermal expansion are determined by tests of dilatometry which took place with a temperature T_{def} different from the reference temperature T_{ref}.
  It is then necessary to carry out a change of reference in the computation of the thermal strain [R4.08.01].

  $$\varepsilon^{th}(T)=\varepsilon_m^{th}(T)-\varepsilon_m^{th}(T_{ref})$$

  wh er e ε_m^{th} is the measured thermal strain (definite compared to the temperature T_{def}),

  ε^{th} is the calculated thermal strain (definite compared to the temperature T_{ref}).

  The temperature T_{def} is indicated under key word TEMP_DEF_ALPHA in DEFI_MATERIAU, and the values of the coefficient of thermal expansion (definite compared to the temperature T_{def}) are indicated under key word ALPHA or (ALPHA_*) in DEFI_MATERIAU.

Notice concerning the THM:

In a modelization THM, the temperature is not a command variable. However, the coefficient of thermal expansion can be a function of the temperature. To be able to transform curved alpha (T) (TEMP_DEF_ALPHA - > TEMP_REF), a reference temperature is necessary. This is why, contrary so that known as syntax, it is possible to inform NOM_VARC=' TEMP' + VALE_REF without informing key keys CHAM_GD or EVOL.
But this use must be held with modelizations THM.

4.4.5.2 Drying of reference (c_0):

c_0 represent the water content initial of the concrete. The user must provide this number when he does a mechanical computation (MECA_STATIQUE or STAT_NON_LINE) with a loading of type drying.

c0 must be given in the same units as “drying” (AFFE_MATERIAU/AFFE_VARC= _F (NOM_VARC=' SECH'…) for example in L/m^3). This unit must be coherent with parameter DEFI_MATERIAU/ELAS_FO/K_DESSIC.

A this water content initial, the shrinkage of desiccation is null since:
`EPS_rd = K_DESSIC(c0 - c).`

4.4.6 Keywords "hidden" for the assignment of the command variables

We saw how the user can affect fields (isolated or coming from `sd_resultat`) like command variable for his later computations.
But the command variables are scalars named and the associated fields also of the components named. The problem is to associate each command variable with a component of the field.

There exists as much of keyword factor "hidden" than of command variables allowing these associations. These keywords "are hidden" because they have values by default (see the table below). One should use them only when one wishes to do something of a little "special". For example:

- to use variables `NEUT_1` or `NEUT_2`,
- to make pass a field of temperature for a field of corrosion,
- …

Let us explain these keywords factors on two examples:

The user carried out a "thermal" computation whose solution is actually an evolution evo1 whose fields contain drying. In this `evol_ther`, fields called `TEMP` carry a component called also `TEMP`.

The user who wishes to use such fields as command variable `SECH` will be able to write:

```
CHMAT=AFFE_MATERIAU (…
       VAR_SECH=_F (NOM_VARC=' SECH', GRANDEUR=' TEMP_R',
                                                 CMP_VARC=' SECH', CMP_GD='
TEMP'))
```

What one can translate by: "the fields which I wish to affect as command variable "`SECH`" are quantity `TEMP_R` and the component to be used is "`TEMP`".

When a command variable is a "vector" having several scalar components, for example the variable `M_ACIER` which has 7 components ("`PFERRITE`",…, "`TACIER`"). The user can write:

```
CHMAT=AFFE_MATERIAU (…
       VAR_M_ACIER=_F (NOM_VARC=' M_ACIER', GRANDEUR=' VARI_R',
                       CMP_VARC= ("PFERRITE", "PEARLITE",…, "TACIER"),
                       CMP_GD= ("V1", "V2",…, "V7")))
```

What wants to say: "the fields which I wish to affect as command variable "`M_ACIER`" are quantity `VARI_R` and the correspondence of the components to be used is: `("PFERRITE", "V1"), ("TACIER", "V7").`

The hidden keywords have like values by default:

NOM_VARC	***QUANTITY***	***CMP_VARC***	***CMP_GD***
TEMP	TEMP_R	TEMP	TEMP
GEOM	GEOM_R	X	X
		Y	Y
		Z	Z
SECH	TEMP_R	SECH	TEMP
HYDR	HYDR_R	HYDR	HYDR

CORR	CORR_R	CORR	CORR
IRRA	IRRA_R	IRRA	IRRA
NEUT1	NEUT_R	NEUT1	X1
NEUT2	NEUT_R	NEUT2	X1
EPSA	EPSI_R	EPSAXX	EPXX
		EPSAYY	EPYY
		EPSAZZ	EPZZ
		EPSAXY	EPXY
		EPSAXZ	EPXZ
		EPSAXZ	EPXZ
M_ACIER	VARI_R	PFERRITE	V1
		PEARLITE	V2
		PBAINITE	V3
		PMARTENS	V4
		TAUSTE	V5
		TRANSF	V6
		TACIER	V7
M_ZIRC	VARI_R	ALPHPUR	V1
		ALPHBETA	V2
		TZIRC	V3
		TEMPS	V4
PTOT	DEPL_R	PTOT	PTOT
DIVU	EPSI_R	DIVU	DIVU

4.5 Key word factor AFFE_COMPOR

This key word factor allows D" to affect the "behavior multifibre" of the multifibre beam elements.

The key key TOUT, GROUP_MA and MESH make it possible to indicate the group of meshes which will be affected.

Behind the key word COMPOR, the user will of the command indicate the name of a concept of the coming type compor DEFI_COMPOR/MULTIFIBRE.

4.5.1 Operand MATER

♦MATER = mat,

Name of the material which one wants to affect.

In the general case, each mesh is affected only by one material. Sometimes, it is necessary to indicate a list of materials when the nonlinear structural mechanics behavior is obtained by the command DEFI_COMPOR [U4.43.06].

5 Examples

Example 1: Mechanics without thermal thermal expansion

```
chmat = AFFE_MATERIAU   ( MAILLAGE = my,
        AFFE = (
          _F (TOUT = "OUI' ,               MATER = steel),
          _F (MAILLE= ("ma1", "ma2", "ma3"),    MATER = aluminum,),),
                       )
```

On the group of the mesh (except meshes: ma1, ma2, ma3) are affected the material of name steel. On meshes the ma1, ma2, ma3 are affected the material aluminum.

Example 2: Mechanics with thermal thermal expansion

Assignment on all the mesh of the material MAT whose certain parameters are functions of the temperature. Moreover the thermal coefficient of thermal expansion is defined for this material. The temporal evolution of the temperature is given via data structure result EVOTH (of evol_ther type). The reference temperature (that for which thermal expansion is null) is worth 20 degrees.

```
CHMAT = AFFE_MATERIAU (MAILLAGE = MY,
   AFFE =_F (TOUT=' OUI', MATER = MAT,),
   AFFE_VARC=_F (NOM_VARC=' TEMP', EVOL =EVOTH, VALE_REF=20. ),
   )
```

Example 3: Mechanics with thermal thermal expansion + complex modelization

In the preceding example, the thermal evolution (EVOTH) applies to all the elements of the model. But it can happen that this situation is unrealistic for certain complex modelizations. It is then necessary to repeat several times key word AFFE_VARC/NOM_VARC=' TEMP' to affect different thermal evolutions on various parts of the model.

In the following example, the model is a model 3D in which are plunged steel reinforcements. A thermal computation was carried out as a preliminary without taking account of reinforcements. It was obtained result that one called EVOTH3D. The temperature of the nodes of the elements of reinforcement is then unknown. So in addition, one is able to evaluate the temperature of reinforcements (measurements,…) and that this temperature is stored in field (TEMP_ARM), one can then calculate the thermal mechanical computation thermal expansion with the following material field:

```
       CHMAT = AFFE_MATERIAU (MAILLAGE = MY,
         AFFE =_F (…),
         AFFE_VARC= (
          _F (NOM_VARC=' TEMP', GROUP_MA=' VOLUM', EVOL =EVOTH3D, VALE_REF=20. ),
              _F   (NOM_VARC='   TEMP',   GROUP_MA='   ARMA',   CHAM_GD   =TEMP_ARM,
   VALE_REF=20. ),
       ))
```

Example 4: Mechanics with influence of the irradiation

Assignment on all the mesh of the material MAT whose certain parameters are functions of the irradiation. The temporal evolution of the irradiation is given via the SD result EVOL = FLUENC.

```
CHMAT = AFFE_MATERIAU (MAILLAGE = MY,
   AFFE =_F (TOUT=' OUI', MATER = MAT,),
   AFFE_VARC=_F (NOM_VARC=' IRRA', EVOL =FLUENC,),
   )
```

Example 5: Mechanical computation with a field of modulus Young imposed

In this example (resulting from the case test ssnv130c), one wants to illustrate the possibility of using a field of modulus Young whom one supposes known (CHYOUNG). For example, this field is read in a file (LIRE_CHAMP) or it is result of a computation.

The “trick” consists in defining a material for which the Young modulus (key word ELAS/E) is the function “identity” of variable “NEUT1” and one affects field CHYOUNG like command variable “NEUT1”.

```
CHYOUNG=…
NU_F=DEFI_CONSTANTE (VALE=0.3)
E_F  =  DEFI_FONCTION  (NOM_PARA='  NEUT1',  VALE=  (-  1.E-9,  -  1.E-9,
1.E+9,1.E+9))
MA=DEFI_MATERIAU (ELAS_FO=_F (E=E_F, NU=NU_F,),);

CM=AFFE_MATERIAU (MAILLAGE=M,
                  AFFE=_F (TOUT= “OUI', MATER= MY),
                  AFFE_VARC=_F (NOM_VARC=' NEUT1', CHAM_GD=CHYOUNG),
                  )
```