Communication with the Supervisor of execution: routines GETXXX

Summary:

This document describes the operation of the Supervisor briefly, it details the requests of the operators to the Supervisor concerning the contents of the orders user (use of the routines GETVXX, GETRES and GETFAC) or of the formal description of an order in the catalogue (use of the routines GETMXX). Examples of use of these routines are treated.

Contents

1 Operation summary of the Supervisor.....3
2 Communication enters the operators and the Supervisor.....3
2.1 Description of the routines users.....3
2.1.1 Arguments of the routines getvxx.....3
2.1.2 Treatment of the detected errors.....4
2.1.3 List of the routines getvxx.....4
2.1.4 getltx to obtain the lengths of the chains of a parameter of the character string type.....5
2.1.5 GETRES to obtain information on the result of an order.....5
2.1.6 GETFAC to obtain the number of occurrences of a keyword factor.....6
2.1.7 GETTCO to obtain the type attached to a concept.....6
2.1.8 GCUCON for to test the existence of a concept in the command set.....7
3 Routines of access to the catalogue of the orders (routines getmxx).....7
3.1 Description of the routines.....7
3.1.1 GETMAT for the list of the keywords ratios control employee.....7
3.1.2 GETMJM to obtain relative information with the description of a keyword factor in the order user current.....8
3.1.3 GETEXM to obtain the composition of a keyword factor in an order.....8

1 Operation summary of the Supervisor

What is this it is?

The supervisor is a whole of routines FORTRAN, C and modules python. One thus names the part of the software which manages the execution of *Code_Aster* upstream of routines FORTRAN principal partners with the orders (`opxxxx`) as well as the routines of request of information on the command file since FORTRAN (`getxxx`).

Role of the supervisor:

Its principal functions are:
- reading of the catalogues of orders,
- reading of **command set** provided by the user, the syntactic checking of this one (coherence with the catalogues of order, obligatory keywords…), the construction of the python objects associated with the command set itself, the orders (stages), the keywords,
- the opening of the management system of the memory capacity (`JEVEUX`),
- execution of the macro-orders: enrichment of the python object "command set" of the orders produced by these macros,
- the treatment orders by ordering of the command set: with each stage treatment of the pending order - called **current order** - start the execution of an operator FORTRAN whose name is specified in the catalogue of the order by the attribute " `op` ",
- supply of a set of functions described in this document which make it possible to count the values associated with the keyword with the orders and if required to recover these values with the request.

The call to the operators, since the Supervisor, is done by subroutines of the type `op0xxx`.
These subroutines do not have an argument.

2 Communication enters the operators and the Supervisor

The operators make requests with the Supervisor concerning the contents of the orders of the user.

For that they have a set of routines giving access to them the arguments keywords:

- Request of access to the values of the keywords:

  A set of subroutines specific to each known type of the supervisor is available:

`getvis`	Recovery of whole values
`getvr8`	Recovery of actual values
`getvc8`	Recovery of complex values
`getvid`	Recovery of a name of concept
`getvtx`	Recovery of values texts (character string)
`getltx`	Recovery lengths of the values texts

- Request of access to the result:
  The subroutine `getres` allows to obtain the name user of the result as well as the name of the associated type known by the supervisor.

- Request relating to the composition of the orders in the command file:

`getfac`	Recovery amongst occurrence of a keyword factor
`gettco`	Recovery of the type of a concept.
`gcucon`	Presence of a concept in the command set

2.1 Description of the routines users

2.1.1 Arguments of the routines `getvxx`

Routines getvxx allow to recover the values provided in a simple keyword located under a keyword factor, in the current order.
The signature of the routines is thus (only the type of the turned over values changes):

```
    cal getvxx (motfac, keyword, iocc, nbval, vect, scal, nbret, isdefault)
```

motfac is the name of the keyword factor, “if the simple keyword is not under a keyword factor.
keyword is the name of the simple keyword.
The following arguments are optional.
iocc is the number of occurrence of the keyword factor.
nbval is the number of values to be written to the maximum in vect.
vect is the vector in which will be written the values read.
scal is the variable receptacle when one wants to read only one value.
nbret is the number of values actually read if nbret is positive or null. If nbret is negative, there were more values than nbval (example: the user informs 3 values, if nbval=2, one will have nbret=-3 and the first 2 values are affected in the vector vect).
isdefault 1 is worth if the value were indicated by the user, 0 if it is the defect of the catalogue.

Warning

> *isdefault is not relevant if the simple keyword (or the keyword factor which contains it) is under a block.*

To know how much values were indicated and thus to dimension the vector vect correctly, one makes usually a call of this type with nbval=0:

```
    cal getvxx (motfac, keyword, iocc, nbval=0, nbret=nbv)
```

allowance of a table of size – nbv, then:

```
    cal getvxx (motfac, keyword, iocc, nbval=-nbv, vect=array)
```

2.1.2 Treatment of the detected errors

The principal detectable errors with the supervision are:

1) the command file does not contain the required simple keyword:
   - first case: the catalogue of order proposes a value by default for the keyword: this value is turned over by the routine,
   - second case: there is no value by default but the statute of the parameter is “optional” (‘F’): the routine turns over nbret=0,
   - third case: the parameter is obligatory the treatment is stopped with phase 1 of the analysis of the command file (see Operation summary of the Supervisor, page 1).
2) A value invalidates for iocc involve the interruption of the treatment.
3) One makes a request on a keyword factor or a simple keyword which do not exist in the catalogue of the order (typing error).

2.1.3 List of the routines getvxx

getvc8 : reading of a parameter of the type ‘It, complex numbers. vect and scal are declared complex (kind=8).

getvis : reading of a parameter of the type ‘I’, integers. vect and scal are declared integer.

getvr8 : reading of a parameter of the type ‘R’, real numbers. vect and scal are declared real (kind=8).

getvtx : reading of a parameter of the type ‘TXM’, character strings. vect and scal are declared character (len=*).

getvid : reading of a parameter of the type 'CO', concepts. vect and scal are declared character (len=*).

2.1.4 getltx to obtain the lengths of the chains of a parameter of the character string type

Goal

From the current order, name of a simple keyword located under a keyword factor - current order - which one provides the name or directly located under the current order, the routine GETLTX turn over in a table of entireties the exact length of each character string attached to the simple keyword in the command file.

The exact length of a character string is the exact number of significant natures (not white) composing the value attached to the simple keyword.

Use

In a routine FORTRAN, GETLTX requires the following arguments:

MOTFAC	Name of the keyword factor, of the current order, in which one seeks the simple keyword. White space if the simple keyword KEYWORD is located directly under the current order.
KEYWORD	Name of the simple keyword, in which one seeks the values. The simple keyword can be located under a keyword current ratio control, in a current valve block, under a keyword factor of a current valve block or directly under the current order. In the first case, MOTFAC does not have to be white.
IOCC	If the keyword factor appears several times, in the current order, IOCC indicate the number of occurrence for which the user searches the simple keyword; if there is only one occurrence, IOCC must be equal to 1. The number of occurrences of the keyword factor can be obtained by using the routine GETFAC (see page 1). If the keyword factor is white, the argument IOCC is ignored.
IARG	0 if the value were indicated by the user, 1 is worth if it is the value by default defined in the catalogue.
MXVAL	Maximum number of turned over values, otherwise-known as cuts table ISVAL (this number can be dynamically given to see p. 1)
ISVAL	Table containing them NBVAL length of the character strings turned over; if the simple keyword is not found in the command set, table FORTRAN ISAL is not modified.
NBVAL	Effective number of turned over values (=0, if the values were not found in the command set), if the number of values present in the command file is higher than MXVAL, the list of the turned over values is truncated with MXVAL values and NBVAL is turned over with the negative value – MXVAL.

2.1.5 GETRES to obtain information on the result of an order

Goal

The routine GETRES, turns over in character strings, information concerning current order:

- the name user of the turned over concept (if there is one of them),
- the type of the turned over concept (if there is one of them),
- the name of the order.

For example starting from the order:

```
mail=LIRE_MAILLAGE ()
```

The call to GETRES

```
CAL GETRES ( NOMRES, CONCEP, NOMCMD )
```

turn over:
NOMRES with the value 'e-mail'
CONCEP with the value 'GRID'
NOMCMD with the value 'LIRE_MAILLAGE'

Use

In a routine FORTRAN, GETRES requires the following arguments:

NOMRES	Name given by the user to the result produced by the current order. The breakage of this name (tiny or capital) is preserved. If the name of the result appearing in the file is too long compared to the capacity of the variable NOMRES, the supervisor truncates the chain returned to adapt it in keeping with the variable. If the current order is a procedure - i.e. an operation without result - NOMRES is put at white.
CONCEP	Type of result NOMRES ; this information is defined by the developer of the order in the catalogue of the orders (keyword sd_prod) If the current order is a procedure - i.e. an operation without result - CONCEP is put at white.
NOMCMD	Name of the current order (its text without its arguments).

2.1.6 GETFAC to obtain the number of occurrences of a keyword factor

Goal

The routine GETFAC turn over the number of times where this keyword factor appears in the current order.

Use

In a routine FORTRAN, GETFAC requires the following arguments:

MOTFAC	Name of the keyword factor, the current order.
IOCC	Many occurrences of the keyword factor.

2.1.7 GETTCO to obtain the type attached to a concept

Goal

From the name of a concept, result produced by an order, GETTCO turn over the name of its type. One can of course consult GETTCO on concepts recovered like arguments of simple keyword, not necessarily produced by the current order.

Use

In a routine FORTRAN, GETTCO requires the following arguments:

NOMCO	Name of the concept
TYPECO	Name of the type attached in the name of the concept

2.1.8 GCUCON for to test the existence of a concept in the command set

Goal

Checking of the existence of the couple (resul, concep) in the results produced by the preceding stages.

Use

In a routine FORTRAN, GCUCON requires the following arguments:

RESUL	Name of the concept
CONCEP	Name of the type attached in the name of the concept
IRET	Code return: >0 so present, 0 so absent

3 Routines of access to the catalogue of the orders (routines getmxx)

The operators can make requests with the Supervisor in connection with the formal description of an order in its catalogue. Returned information does not relate to anything the particular use which is made of these orders in the data file transmitted to the execution.

The following whole of subroutines is available.

GETMAT recovery of the list of the keywords factors of an order, as described in its catalogue;

GETMJM recovery of general information of a keyword factor;

GETEXM Function indicating if a keyword, simple or factor, is present in the catalogue of the current order.

Notice general:

In the event of error on the name of the order or the operator, one stops the execution of the program.

3.1 Description of the routines

3.1.1 GETMAT for the list of the keywords ratios control employee

Goal

By using the catalogue of the orders – and not the command file user –, the routine GETMAT turn over following information concerning the catalogue of the current order:

- the full number of keywords factors described by the catalogue,
- the list of the names of these keywords factors.

```
CAL GETMAT (NBTMCL , LMOCLE )
```

turn over:

`NBTMCL=180` the number of keywords factors under the order
`LMOCLE= ('ELAS', 'ELAS_FO',…)` the list of character strings containing their names

Use

This routine is only employed by the operator `DEFI_MATERIAU` in order to recover in FORTRAN the list of materials permitted by the catalogue.

3.1.2 GETMJM to obtain relative information with the description of a keyword factor in the order user current

Goal

The routine `GETMJM` turn over information – since the command file of the user – concerning k - ième occurrence of the keyword factor whose name passed in argument:

- the list of the simple keywords of the keyword factor requested,
- for each simple keyword, the type of the expected argument (real, whole…).

Caution :

> *k allows to traverse the whole of the keywords factors but there is no of significant nature of the keywords factors in the order. In particular, the name of the keyword factor turned over for the value k is not the name of k - ième keyword factor such as writing by the user. Moreover, all the conditions of block are supposed to be checked.*

3.1.3 GETEXM to obtain the composition of a keyword factor in an order

Goal

By using the catalogue of the orders, the function `GETEXM` indicate if a simple keyword is present in the catalogue of the current order.

Use

Example of call to know if the catalogue of the current order contains the keyword `MCSIMP` :

```
INFORMER = GETEXM ('' , 'MCSIMP' )
```

Example of call to know if the catalogue of the current order contains the keyword `MCSIMP` under the keyword factor `MCFACT` :

```
INFORMER = GETEXM ('MCFACT' , 'MCSIMP' )
```