

## Impressions directed by the keyword INFORMATION orders (package INFXXX )

---

### Summary:

This document indicates how the developers must carry out the impressions directed by the keyword INFORMATION orders.

## Contents

---

<a href="#">1 Introduction.....</a>	<a href="#">3</a>
<a href="#">2 How to transmit a message of information.....</a>	<a href="#">3</a>
<a href="#">3 To protect itself from an excess of messages.....</a>	<a href="#">4</a>
<a href="#">4 Operation of the utilities 'INFXXX'.....</a>	<a href="#">4</a>
<a href="#">5 Example of use LIRE_MAILLAGE (INFO=2).....</a>	<a href="#">5</a>

## 1 Introduction

The information directed by the keyword `INFORMATION` orders are messages of information concerning the course of calculation. They are contractual (for level 1) and, normally, they are documented in the documentation of use.

The more detailed description of than is (or what is not) one `INFORMATION` is made in the document [D2.07.01], in short:

- one `INFORMATION` is neither a result, nor an alarm...
- the keyword `INFORMATION` can take 2 values: 1 or 2; the defect is 1,
- the impressions are made by `WRITE` and not of `U2MES* <I>` [D6.04.01]
- `INFORMATION` are written on the file `MESSAGE`

Impressions being made by `WRITE` formatted, the only problems to be regulated are:

- on which logical unit?
- how to respect the choice of the user: `INFORMATION = 1 or 2 ?`

This been the object of the paragraph §2 “How to transmit a message of information”.

Another problem occurs when an order does not want a message printed systematically by a utility routine. This been the object of the paragraph §3 “to protect itself from an excess of messages”.

## 2 How to transmit a message of information

- The routine `INFMAJ` is used to inform the “package” `INFXXX` level of impressions asked by the user,
- the routine `INFNIV` is used to recover the level of impression asked by the user as well as the logical unit of the file ‘`MESSAGE`’.
- the programmer of an order will thus make:
  - to put the keyword `INFORMATION` in the catalogue of the order: 2 possible values 1 and 2, value by default:1
  - in the routine `OP000I`

```
CAL GETVIS (`` , `INFORMATION' , 0,1,1, NIV, IBID)
CAL INFMAJ (NIV)
```
  - in a routine wanting to print `INFORMATION`:
    - `CAL INFNIV (IFM, NIV)`
    - for `INFORMATION` of level 1 (contractual):

```
IF (NIV.GE.1) WRITE (IFM,...)
```
    - for `INFORMATION` of level 2:

```
IF (NIV.EQ.2) WRITE (IFM,...)
```

Note:

*One could think that them `INFORMATION` of level 1 can be written without protecting itself by ( `IF (NIV.GE.1) ...`) because these impressions start by default. One will see with the §3 “protecting oneself from an excess of messages”, why they should be protected.  
To protect itself from the orders which do not have the keyword `INFORMATION` in their catalogue, the supervisor updates it `COMMON` front each order: `CAL INFMAJ (1)` .*

## 3 To protect itself from an excess of messages

The person in charge of an order is responsible for the impressions of level 1 of this order. He must document them and arrange himself so that the file message remains readable.

A problem arises when an order called on a utility printing of `INFORMATION` of level 1 that the person in charge of the order judges irrelevant (or too many... for example in a loop). It is necessary to give to the order the means "of making conceal" the utility. For that, one has the 2 routines:

- `INFMUE ()` to make the code "DUMB" until new order,
- `INFBAV ()` to give the code in "TALKATIVE" mode.

These 2 routines are effective only if the level of impression required is 1: one cannot prevent the code from "speaking" if `INFORMATION = 2`.

Example:

The order `OP000I` known as the routine B does not want to only hear

```
SUBROUTINE OP000I
CAL INFMAJ
...
CAL WITH (...)
...
CAL INFMUE ()
CAL B (...)
CAL INFBAV ()
...
END
```

Note:

- *This system is not perfect because it only makes it possible to stop without understanding the flood of the impressions. It does not allow for example, to prohibit B from speaking while leaving C (called by B) speech!*
- *So that `INFMUE/INFBAV` that is to say effective, it is necessary that the routines that one wants to make conceal programmed their impressions of level 1 in the following way*  

```
IF (NIV.GE.1)
WRITE...
```
- *Routines `INFMUE` and `INFBAV` modify (temporarily) the level of impression (`NIV`). But within the routine where they are called, it cannot obviously modify the value of the local variable `NIV`. In the example above, if the routine `OP000I` wants to print `INFORMATION` after the call to `INFMUE`, it must "update" the variable `NIV` by calling the routine again `INFNIV`.*

## 4 Operation of the utilities 'INFXXX'

The purpose of this paragraph is only to help to understand (if necessary) the use of these routines:

- `COMMON/INF001/NIVUTI, NIVPGM, UNIT`
  - `INTEGER NIVUTI:` level requested by the user: 1 or 2
  - `INTEGER NIVPGM:` level accessible to the programmer: 0.1, or 2
  - `INTEGER UNIT:` logical unit of the file 'MESSAGE'
- `COMMON INF001` is used only by routines `INFXXX`

```
SUBROUTINE INFMAJ ()
COMMON/INF001/...
```

*Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.*

Copyright 2021 EDF R&D - Licensed under the terms of the GNU FDL (<http://www.gnu.org/copyleft/fdl.html>)

```
GETVIS (nomCMD, 'I', 'INFORMATION', ..., NIV)  
NIVUTI = NIV  
NIVPGM = NIV  
UNIT = IUNIFI ('MESSAGE')  
END
```

```
SUBROUTINE INFNIV (IFM, NIV)  
COMMON/INF001/...  
INF = UNIT  
NIV = NIVPGM  
END
```

```
SUBROUTINE INFMUE ()  
COMMON/INF001/...  
IF (NIVUTI.EQ.1) NIVPGM = 0  
END
```

```
SUBROUTINE INFBAV ()  
COMMON/INF001/...  
NIVPGM = NIVUTI  
END
```

## 5 Example of use **LIRE\_MALLAGE** (INFO=2)

### Command file **INFORMATION = 1**

```
MY = LIRE_MALLAGE ()
```

### File of message

```
MA=LIRE_MALLAGE (INFO=1,  
                 VERI_MAIL=_F (APLAT=1.E-3, VERIF=' OUI'),  
                 UNITE=20,  
                 FORMAT=' ASTER',  
                 );
```

```
===== CHECKING OF THE GRID =====
```

```
----- GRID MY          - LEVEL IMPRESSIONS 1 -----
```

```
MA-22-AVRI-2008 12:25: 18
```

MANY NODES	4	
MANY MESHS	5	
	SEG2	4
	QUAD4	1
MANY GROUPS OF NODES	1	
	LOW	2
MANY GROUPS OF MESHS	1	
	CONTOUR	4

### Command file **INFORMATION = 2**

Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.

Copyright 2021 EDF R&D - Licensed under the terms of the GNU FDL (<http://www.gnu.org/copyleft/fdl.html>)

MY = LIRE\_MAILLAGE (INFO=2)

## File of message

```
MA=LIRE_MAILLAGE (INFO=2,  
                  VERI_MAIL=_F (APLAT=1.E-3, VERIF=' OUI'),  
                  UNITE=20,  
                  FORMAT=' ASTER',  
                  );
```

===== CHECKING OF THE GRID =====

----- GRID MY - LEVEL IMPRESSIONS 2 -----

MA-22-AVRI-2008 12:28: 11

MANY NODES	4		
MANY MESHS	5		
	SEG2	4	
	QUAD4	1	
MANY GROUPS OF NODES	1		
	LOW	2	
MANY GROUPS OF MESHES	1		
	CONTOUR	4	

----- LIST OF THE NODES -----

1	NO1	0.00000D+00	0.00000D+00
2	NO2	0.10000D+01	0.00000D+00
3	NO3	0.10000D+01	0.10000D+01
4	NO4	0.00000D+00	0.10000D+01

----- LIST OF THE MESHES -----

1	SQUARE	QUAD4	1	2	3	4
2	LEFT	SEG2	1	4		
3	RIGHT-HAND	SIDE	SEG2	2	3	
4	HIGH	SEG2	3	4		
5	LOW	SEG2	1	2		

----- LIST OF THE GROUPS OF NODES -----

1	LOW	2	1	2
---	-----	---	---	---

----- LIST OF THE GROUPS OF MESHES -----

1	CONTOUR	4	2	3	4	5
---	---------	---	---	---	---	---