

Descriptor of environment machine: ENVIMA

Summary:

One presents in this document the functions allowing to recover the characteristic values whole or real dependent on the platform used. Were added some constant mathematics and certain parameters related to the core use and files.

Contents

1	General presentation and justification.....	3
1.1	Arithmetic parameters.....	4
1.1.1	Representation of the integers.....	4
1.1.2	Representation of the real numbers.....	4
1.1.3	Representation of logics.....	5
1.2	Special values.....	5
1.3	Parameter related to the core use central.....	5
1.4	Parameters of use of the files.....	5
1.5	Constant mathematics.....	5
2	Provided functions and their use.....	6
2.1	General information.....	6
2.2	Logical type.....	6
2.3	Whole type.....	6
2.4	Real type.....	7
2.5	Parameters of use of the files.....	7
2.6	Assignment of binary constants.....	7
2.7	Assignment of constant mathematics or physics.....	7
2.8	Particular values.....	8
2.9	Utility.....	8
2.10	Precautions for use.....	8
3	Bibliography.....	9
4	Appendix: Portability and Effectiveness.....	10
5	Appendix: Result of the subroutine impvem.....	11

1 General presentation and justification

The definition of the types `real` and `double precision` by the standard `FORTRAN 77` does not allow to carry out portable scientific software, with unicity of source and of the comparable digital performances [bib1], [bib2]. `FORTRAN 90` integrate the concept length at the time of the declaration of the various types but unfortunately this does not bring any insurance as for the real precision used, the latter depending on the implementation of the compiler.

Standard IEEE-P745 [3] defines the terminals of binary precision but is not applied by all:

type	significant figures	
	always	sometimes
single precision	6	7 or 8
double precision	15	16
simple extent	>9	-
double extent	>18	-

It is thus necessary to compensate for the imperfections of the standard `FORTRAN 77`, which does not fix rules of representation of the numbers, although there exist suitable algorithms making it possible to determine certain parameters dynamically.

This version of `ENVIMA` is resulting from a preceding realization going back to 1990 which appeared too plentiful, from many functions not having never been used in `Code_Aster`. Their realization in `FORTRAN` was replaced by a code written out of C allowing to gather them with the whole of the functions presenting of adherences to the machines and/or the operating systems. On this occasion, the functions of the type complexes of which the use was marginal, and who present a real difficulty of portability the existing complex type in `FORTRAN` but not out of C, were purely removed.

The software package `ENVIMA` gather several functions without arguments, of name – `EM`, which makes it possible to reach, starting from any routine `FORTRAN`, with the parameters necessary characterizing the machine on which the treatment is carried out.

The parameters are statically defined in each version of the software package for:

- entières: length, values extreme;
- logics: length;
- the floating ones: base system of representation, length of the mantissa, relative precision, representable extreme values;
- constants: particular constants (π , Not.);
- files: limiting sizes (related to constraints of exploitation of the machines).

Four groups of functions are available:

- arithmetic parameters (definition of the numbers),
- parameters for the core use central,
- parameters for the core use auxiliary (files),
- values of constants.

Some definitions:

- **Unit of addressing** : Each manufacturer defines for a machine a mode of addressing of information in memory; the measuring unit of this address is the unit of addressing: the word was used formerly on certain platforms, it is now the byte on most workstations containing processors x86.
- **Length** : Chaque type of variable is characterized by a length of representation out of machine; this one can to be measured out of bits, bytes or units of addressing. The manager of memory `JEVEUX`

used in *Code_Aster* this information at the time of the definition of the attribute of the type requires of objects created.

1.1 Arithmetic parameters

1.1.1 Representation of the integers

Four parameters are available for the variables of the whole type standard (`integer`):

- the length of an entirety measured out of bits, bytes, or units of addressing;
- the maximum number of significant figures to represent the number into decimal;
- the representable maximum value i.e. greatest positive entirety i such as all the entireties of the interval $[-i, +i]$ are represented by the whole type;
- range defined by the greatest entirety i such as:
 - $-i$ that is to say exact,
 - for formula $-i < i_a, i_b < +i$ the operation formulates $|i_a \oplus i_b|$ with formula $\oplus \in \{+, -, *\}$: that is to say exact and formula does not exceed i in absolute value.

1.1.2 Representation of the real numbers

The definition of the standard real type in the standard `FORTRAN 77` does not allow to carry out portable software with unicity of source and comparable digital performances. To achieve this goal, we chose to use in the whole of *Code_Aster* the type except standard `REAL*8`, admitted by very many compilers and which leads to the closest representations (64 bits on any punt forms).

One can give the following impression of the representation machine of the floating numbers:

$$x = \sigma B^E \sum_{k=1}^N x(k) B^{-k} \text{ where } x \text{ indicate a real number,}$$

σ the sign,

B the base of representation (2 most of the time),

E the exhibitor ($E_{min} \leq E < E_{max}$),

N the number of digits allocated with the mantissa.

This representation imposes obviously $0 < x(1) < B$ and $0 < x(i) < B$ for $1 < i < N$. It is thus noted that two distinct realities x_1 and x_2 the representation above is written with the same exhibitor E can differ at least from B^N . When the exhibitor differs from a unit, the difference between two realities is at least of B^{1-N} .

Values B, N, E_{min} and E_{max} were introduced into the software and can be recovered by the adequate function.

It is then easy to define the following characteristic values: smallest positive reality: $B^{E_{min}-1}$, greatest positive reality: $B^{E_{max}}(1 - B^N)$, the smallest relative increment: B^{-N} , the greatest relative increment: B^{1-N}

The parameters available are:

- the length out of bits, bytes or units of addressing;
- the maximum number of significant figures to represent the number into decimal;
- the base of representation B of the floating numbers;
- the length of the mantissa;
- the relative precision is such as any reality other than 1.0 is not represented by: formula $1.0 - \varepsilon_1 < 1.0 < 1.0 + \varepsilon_2$ with $\varepsilon_1 = (1/b)\varepsilon_2$
- representable positive extreme values: maximum (overflow) and minimum (underflow);
- the range is defined by greatest reality such as if $\varepsilon_1, \varepsilon_2, \varepsilon_3$ are about the relative precision formulates ε_1 :

- formula $-x$ that is to say correctly representable by formula $-x(1 \pm \varepsilon_1)$;
- for formula $1/x < |a|, |b| < x$ the operation formulates $a \oplus b$ with $\oplus \in \{+, -, *, /\}$ is such as formula $1/x < |a \oplus b| < x$ is correctly represented by formula $a(1 \pm \varepsilon_2) \oplus b(1 \pm \varepsilon_3)$.

1.1.3 Representation of logics

Only one parameter is necessary:

- the length in bytes.

1.2 Special values

The actual value `Not` (Not has Number) can be obtained on the machines supporting arithmetic `IEEE`. Historically on waiters `CRAY` the value used was `UNDEF`, it could apply to floating but too with the entireties. This value is used in certain cases (search for bug) to reset the zones memory associated with the objects managed by `JEVEUX`.

The assignment of variables by the value `Not` then allows to detect their use in floating operations because it immediately causes a stop of the code with emission of a signal (handler) which can be recovered.

1.3 Parameter related to the core use central

The length in bytes of the unit of addressing of the machine is the only accessible parameter.

1.4 Parameters of use of the files

Two parameters are available:

- maximum size in bytes of a file,
- maximum size in bytes of the whole of the open files.

They were introduced only because as of constraints related to the exploitation of shared resources (on the waiter centralized aster, limitation of the temporary space of files associated with a batch processing) and are used in the management of the files of binary direct access of the manager of memory.

1.5 Constant mathematics

A set of universal constants (optimal in the type requested) is provided to the user.

These constants are (currently):

- values of π and of 2π ,
- the value of the absolute zero for the temperature,
- parameters of conversion radian/degree and degree/radian.

2 Provided functions and their use

2.1 General information

To help with the use of these functions, one endeavoured to codify the name of the functions starting from three components:

- nature GG size:
 - LB length out of bits,
 - LO length in bytes,
 - MY maximum positive value,
 - SEMI minimal positive value,
 - GA range of values.
- The type All size to which it applies:
 - IS INTEGER (entirety),
 - LS LOGICAL (logic),
 - R8 REAL*8 (reality),
 - C8 COMPLEX*8 (complex).
- The suffix EM (Environment Machine).

For the parameters length (l_) the functions are of whole type. The names of function are form: GGTEM.

For the parameters whose value depends on the type, the type of the function will be that of the required value. In this case the names of function are form: TTGGEM.

Since these functions are not part of the intrinsic functions of the language, it is essential to declare their type in each unit of program user.

2.2 Logical type

- Standard logical type (LOGICAL)

Type of the function	Provided parameter	Function ENVIMA	Platform 64 bits	Platform 32 bits
INTEGER	Length in bytes	LOLSEM	8	4

2.3 Whole type

- Standard whole type (INTEGER)

Type of the function	Provided parameter	Function ENVIMA	Platform 64 bits	Platform 32 bits
INTEGER	Length in bytes	LOISEM	8	4
INTEGER	Maximum value	ISMAEM	formula $2^{63}-1$	formula $2^{31}-1$ 2147483647

2.4 Real type

- Standard real type (REAL*8)

Type of the function	Provided parameter	Function ENVIMA	Platform 64 bits	Platform 32 bits
REAL*8	Relative precision	R8PREM	2.220D-16	2.220D-16
REAL*8	Maximum value	R8MAEM	1.798D+308	1.798D+308
REAL*8	Minimal value	R8MIEM	2.225D-308	2.225D-308
REAL*8	Range	R8GAEM	1.340D+154	1.340D+154

2.5 Parameters of use of the files

These parameters are used in reasons of constraints of exploitation on the waiters whose resources are shared. The bases (files `glob.i` and `vola.i`) are in general created on a limited shared space (`/tmp`).

Type of the function	Provided parameter	Function ENVIMA	Platform 64 bits	Platform 32 bits
INTEGER	Maximum length of a file in bytes	LOFIEM	12582912	2048000
INTEGER	Cumulated size of the files in bytes	MOFIEM	50331648	50331648

2.6 Assignment of binary constants

For the construction of masks on the numerical information, it is necessary to reach the weights of the bits quickly. Those are accessible by the whole function `ISPBE`M (number of the bit).

$$\text{ISPBE}M(4) = 00000000000000000000000000000001000 = 2^{4-1} = 8$$

2.7 Assignment of constant mathematics or physics

The particular constants are accessible starting from the type from the required variable:

Type of the function	Provided parameter	Function ENVIMA	Value
REAL*8	π	R8PI	3.141592653589793D+00
REAL*8	2π	R8DEPI	6.283185307179586D+00
REAL*8	Conversion radian/degree	R8DGRD	1.745329251994330D-02
REAL*8	Conversion degree/radian	R8RDDG	5.729577951308232D+01
REAL*8	Absolute zero (temperature)	R8T0	273.15

2.8 Particular values

These values are available on all the machines having arithmetic IEEE.

Type of the function	Provided parameter	Function ENVIMA	Value
INTEGER	Value aggressive an error at the time of a floating operation	ISNNEM	9223372036854775807.
REAL*8	Value aggressive an error at the time of a floating operation	R8NNEM	Not
REAL*8	Value of initialization of the structures of data	R8VIDE	1.797693134862316+308

2.9 Utility

Subroutine FORTRAN `IMPVEM` allows to publish overall the various values characteristic of the processing machine on the file of logical number of unit `IUL`. An example is given in appendix.

`CAL IMPVEM (IUL)` : print all the characteristics of the standard types.

2.10 Precautions for use

Since these functions are not part of the intrinsic functions of the language, it is essential to declare their type in each unit of program user. In this case the names of function are `TTGGEM` with `ALL` type of the function and `GG` nature of the size (cf. example of use).

Note:

The values concerning the entireties are in the whole implicit type (functions starting with a letter L or N).

Functions `ENVIMA` are available in the library `C utilitai` of `Code_Aster`, and are pressed for the majority on the standard bookstores `C` : files of heading `<limits.h>` and `<float.h>` [bib4].

By definition any operation of bearing of a subset of `Code_Aster` on other materials requires the preliminary bearing of `ENVIMA`. The use of the standard headings notably limits a research deepened in the documentation of the manufacturer.

3 Bibliography

- [1] Standard CODYW.J., Floating Not Parameters, Model and - Relationship between numerical computations and programming languages, (REID J.K. ED) North-Holland Amsterdam, 1982.
- [2] FORD B., Parameterization of Environment for Transportable Numerical trans Software ACM. of Mathematical Software. Flight 4, n°2 June 1978 pp 100-103.
- [3] I.E.E.E., Standard for floating-point arithmetic, Standard ANSI/IEEE 754-1985, 1985.
- [4] KERNIGHAN B.W., RITCHIE D.M., the language C - C ANSI, Masson Prentice Hall, 1992.

4 Appendix: Portability and Effectiveness

If software written with ENVIMA are portable, ENVIMA is not portable: it requires a rewriting by using the directives of the pre processor C.

Rules of use of the functions of ENVIMA are the same ones as for any call of function which must be done out of internal loops.

The effectiveness is increased with regard to the homogenisation and the portability of the code which sees disappearing from all the routines of the instructions (or from DATED) specific to the machine.

The effectiveness is increased with regard to the digital resolution thanks to the values which as well as possible define the precision, the maximum and the minimum, the range of use.

5 Appendix: Result of the subroutine `impvem`

The following impressions are the result of under program `impvem`, they can be easily obtained by using the command file `Code_Aster` according to:

```
BEGINNING (DEBUG=_F (ENVIMA=' TEST'));  
END ();
```

See the file `result case test ZZZZ103`.

The file `result` produced on a platform `Calibre6 (fort.8)` contains the various characteristic values printed in a decimal and hexadecimal format.

```
-----  
---- ENVIMA VERSION 97 MULTI MACHINES -----  
-----  
----- LENGTH OUT OF BITS  
  
LBISEM ENTIRETY INTEGER 64  
  
----- LENGTH IN BYTES  
  
LOLSEM LOGIC LOGICAL 8  
LOISEM ENTIRETY INTEGER 8  
LOR8EM REALITY REAL*8 8  
LOC8EM COMPLEX COMPLEX*16 16  
  
----- LENGTH AND SIZE OF FILE  
  
LOFIEM IN BYTES 12582912  
MOFIEM IN BYTES 50331648  
  
----- STANDARD ENTIRETY  
  
ISMAEM ENTIRETY INTEGER 9223372036854775807 7FFFFFFFFFFFFFFF  
ISNNEM ENTIRETY INTEGER 9223372036854775807 7FFFFFFFFFFFFFFF  
  
----- REAL*8  
  
FORMAT OF IMPRESSION OF FLOATING (1X, A, 1PD24.15, 2X, Z16)  
  
R8BAEM BASE NUMERATION REAL*8 2.000000000000000D+00 4000000000000000  
R8PREM RELATIVE PRECISION REAL*8 2.220446049250313D-16 3CB0000000000000  
R8MAEM MAXIMUM REAL*8 1.797693134862316+308 7FFFFFFFFFFFFFFF  
R8MIEM MINIMAL REAL*8 2.225073858507201-308 1000000000000000  
R8GAEM RANGE REAL*8 1.340780792994259+154 5FFFFFFFFFFFFFFF  
R8NNEM NOT WITH NUMBER REAL*8 Not  
FFF7FFFFFFFF0000  
R8VIDE VACUUM REAL*8 1.797693134862316+308  
7FFFFFFFFFFFFFFF  
RMIREM B ** - T 1.110223024625157D-16  
3CA0000000000000  
RMAREM B ** (1-T) 2.220446049250313D-16  
3CB0000000000000  
RMINEM B ** (EMIN-1) 2.225073858507201-308  
1000000000000000  
RMAXEM B ** EMAX (1-B ** (- T)) 8.988465674311579+307  
7FDFFFFFFFFFFFFFFF
```

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>)

----- WEIGHT OF BITS 1 WITH LBIS
ISPBE M BITS NUMBER FROM RIGHT TO LEFT

1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128
9	256
10	512
11	1024
12	2048
13	4096
14	8192
15	16384
16	32768
17	65536
18	131072
19	262144
20	524288
21	1048576
22	2097152
23	4194304
24	8388608
25	16777216
26	33554432
27	67108864
28	134217728
29	268435456
30	536870912
31	1073741824
32	2147483648
33	4294967296
34	8589934592
35	17179869184
36	34359738368
37	68719476736
38	137438953472
39	274877906944
40	549755813888
41	1099511627776
42	2199023255552
43	4398046511104
44	8796093022208
45	17592186044416
46	35184372088832
47	70368744177664
48	140737488355328
49	281474976710656
50	562949953421312
51	1125899906842624
52	2251799813685248
53	4503599627370496
54	9007199254740992
55	18014398509481984
56	36028797018963968

Code_Aster

Version
default

Titre : Descripteur d'environnement machine : ENVIMA
Responsable : LEFEBVRE Jean-Pierre

Date : 10/06/2011 Page : 13/13
Clé : D6.01.01 Révision :
441ec035fc0a

57 72057594037927936
58 144115188075855872
59 288230376151711744
60 576460752303423488
61 1152921504606846976
62 2305843009213693952
63 4611686018427387904

----- PARTICULAR VALUES PI, DEPI, ...

R8PI	REALITY	REAL*8	3.141592653589793D+00
400921FB54442D18			
R8DEPI	REALITY	REAL*8	6.283185307179586D+00
401921FB54442D18			
R8DGRD	REALITY	REAL*8	1.745329251994330D-02
3F91DF46A2529D39			
R8RDDG	REALITY	REAL*8	5.729577951308232D+01
404CA5DC1A63C1F8			

---- FINE TEST ENVIMA MULTI MACHINES -----

OK