Descripteur d'environnement machine : ENVIMA

Résumé :

On présente dans ce document les fonctions permettant de récupérer les valeurs entières ou réelles caractéristiques dépendantes de la plate-forme utilisée. Ont été ajoutées quelques constantes mathématiques et certains paramètres liés à l'utilisation de la mémoire et des fichiers.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Table des matières

1 Présentation générale et justification....3
1.1 Paramètres arithmétiques....4
1.1.1 Représentation des nombres entiers....4
1.1.2 Représentation des nombres réels....4
1.1.3 Représentation des logiques....5
1.2 Valeurs spéciales....5
1.3 Paramètre lié à l'utilisation de la mémoire centrale....5
1.4 Paramètres d'utilisation des fichiers....5
1.5 Constantes mathématiques....5
2 Les fonctions fournies et leur utilisation....6
2.1 Généralité....6
2.2 Type logique....6
2.3 Type entier....6
2.4 Type réel....7
2.5 Paramètres d'utilisation des fichiers....7
2.6 Affectation de constantes binaires....7
2.7 Affectation de constantes mathématiques ou physiques....7
2.8 Valeurs particulières....7
2.9 Utilitaire....8
2.10 Précautions d'emploi....8
3 Bibliographie....9
4 Annexe : Portabilité et Efficacité....10
5 Annexe : Résultat du sous-programme impvem....11

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

1 Présentation générale et justification

La définition des types real et double precision par la norme FORTRAN 77 ne permet pas de réaliser des logiciels scientifiques portables, avec unicité de source et des performances numériques comparables [bib1], [bib2]. FORTRAN 90 intègre la notion de longueur lors de la déclaration des différents types mais malheureusement ceci n'apporte aucune assurance quant à la précision réelle utilisée, cette dernière dépendant de l'implémentation du compilateur.

La norme IEEE-P745 [3] définit les bornes de précision binaire mais n'est pas appliquée par tous :

type	chiffres significatifs	
	toujours	parfois
simple précision	6	7 ou 8
double précision	15	16
simple étendue	>9	-
double étendue	>18	-

Il est donc nécessaire de suppléer aux imperfections de la norme FORTRAN 77, qui ne fixe pas de règles de représentation des nombres, bien qu'il existe des algorithmes appropriés permettant de déterminer certains paramètres dynamiquement.

Cette version d'ENVIMA est issue d'une précédente réalisation datant de 1990 qui s'est révélée trop foisonnante, de nombreuses fonctions n'ayant jamais été utilisées dans *Code_Aster*. Leur réalisation en FORTRAN a été remplacée par un code écrit en C permettant de les regrouper avec l'ensemble des fonctions présentant des adhérences aux machines et/ou aux systèmes d'exploitation. À cette occasion, les fonctions de type complexe dont l'usage était marginal, et qui présentent une réelle difficulté de portabilité le type complexe existant en FORTRAN mais pas en C, ont été purement supprimées.

Le progiciel ENVIMA regroupe plusieurs fonctions sans arguments, de nom –EM, qui permettent d'accéder, à partir de n'importe quelle routine FORTRAN, aux paramètres nécessaires caractérisant la machine sur laquelle le traitement est réalisé.

Les paramètres sont définis statiquement dans chaque version du progiciel pour :
- les entiers : longueur, valeurs extrémales ;
- les logiques : longueur ;
- les flottants : base du système de représentation, longueur de la mantisse, précision relative, valeurs extrémales représentables ;
- les constantes : constantes particulières (π , NaN, ..) ;
- les fichiers : tailles limites (liées aux contraintes d'exploitation des machines).

Quatre groupes de fonctions sont disponibles :
- paramètres arithmétiques (définition des nombres),
- paramètres pour l'utilisation de la mémoire centrale,
- paramètres pour l'utilisation des mémoires auxiliaires (fichiers),
- valeurs de constantes.

Quelques définitions :
- **Unité d'adressage** : Chaque constructeur définit pour une machine un mode d'adressage des informations en mémoire; l'unité de mesure de cette adresse est l'unité d'adressage : le mot était utilisé autrefois sur certaines plates-formes, c'est maintenant l'octet sur la plupart des stations de travail à base de processeurs x86.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- **Longueur** : Chaque type de variable est caractérisé par une longueur de représentation en machine ; celle-ci peut être mesurée en bits, en octets ou en unités d'adressage. Le gestionnaire de mémoire JEVEUX utilisé dans *Code_Aster* requiert cette information lors de la définition de l'attribut de type des objets créés.

1.1 Paramètres arithmétiques

1.1.1 Représentation des nombres entiers

Quatre paramètres sont disponibles pour les variables de type entier standard (integer) :
- la longueur d'un entier mesurée en bits, en octets, ou en unités d'adressage ;
- le nombre maximum de chiffres significatifs pour représenter le nombre en décimal ;
- la valeur maximale représentable c'est à dire le plus grand entier positif i tel que tous les entiers de l'intervalle $[-i,+i]$ soient représentés par le type entier ;
- la gamme définie par le plus grand entier i tel que :
  - $-i$ soit exact,
  - pour formule $-i<i_a,i_b<+i$ l'opération formule $|i_a \oplus i_b|$ avec formule $\oplus \in \{+,-,*\}$: soit exacte et n'excède pas formule i en valeur absolue.

1.1.2 Représentation des nombres réels

La définition du type réel standard dans la norme FORTRAN 77 ne permet pas de réaliser des logiciels portables avec unicité de source et performances numériques comparables. Pour atteindre cet objectif, nous avons choisi d'utiliser dans l'ensemble de *Code_Aster* le type hors norme REAL*8, admis par de très nombreux compilateurs et qui conduit aux représentations les plus voisines (64 bits sur toute plate forme).

On peut donner l'image suivante de la représentation machine des nombres flottants :

$x=\sigma\, B^E \sum_{k=1}^{N} x(k)\, B^{-k}$ où x désigne un nombre réel,

σ le signe,

B la base de représentation (2 la plupart du temps),

E l'exposant ($E_{min} \le E < E_{max}$),

N le nombre de digits alloués à la mantisse.

Cette représentation impose évidemment $0<x(1)<B$ et $0<x(i)<B$ pour $1<i<N$. On constate donc que deux réels distincts x_1 et x_2 dont la représentation ci-dessus s'écrit avec le même exposant E peuvent différer au minimum de B^N . Lorsque l'exposant diffère d'une unité, la différence entre les deux réels est au minimum de B^{1-N} .
Les valeurs B, N, E_{min} et E_{max} ont été introduites dans le logiciel et peuvent être récupérées par la fonction adéquate.

Il est alors aisé de définir les valeurs caractéristiques suivantes : le plus petit réel positif : $B^{E_{min}-1}$, le plus grand réel positif : $B^{E_{max}}(1-B^N)$, le plus petit incrément relatif : B^{-N} , le plus grand incrément relatif : B^{1-N}

Les paramètres disponibles sont :
- la longueur en bits, en octets ou en unités d'adressage ;
- le nombre maximum de chiffres significatifs pour représenter le nombre en décimal ;
- la base de représentation B des nombres flottants ;
- la longueur de la mantisse ;
- la précision relative est telle qu'aucun réel autre que 1.0 ne soit représenté par : formule $1.0-\varepsilon_1<1.0<1.0+\varepsilon_2$ avec $\varepsilon_1=(1/b)\varepsilon_2$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- les valeurs extrémales positives représentables: maximum (overflow) et minimum (underflow) ;
- la gamme est définie par le plus grand réel tel que si $\varepsilon_1, \varepsilon_2, \varepsilon_3$ sont de l'ordre de la précision relative formule ε_1 :
  - formule $-x$ soit correctement représentable par formule $-x(1.\pm\varepsilon_1)$;
  - pour formule $1/x<|a|,|b|<x$ l'opération formule $a\oplus b$ avec $\oplus\in\{+,-,*,/\}$ est telle que formule $1/x<|a\oplus b|<x$ est correctement représentée par formule $a(1\pm\varepsilon_2)\oplus b(1\pm\varepsilon_3)$.

1.1.3 Représentation des logiques

Un seul paramètre est nécessaire :
- la longueur en octets.

1.2 Valeurs spéciales

La valeur réelle `NaN` (Not a Number) peut être obtenue sur les machines supportant une arithmétique `IEEE`. Historiquement sur les serveurs CRAY la valeur utilisée était `UNDEF`, elle pouvait s'appliquer aux flottants mais aussi aux entiers. Cette valeur est utilisée dans certains cas (recherche de bug) pour réinitialiser les zones mémoire associées aux objets gérés par `JEVEUX`.
L'affectation de variables par la valeur `NaN` permet ensuite de détecter leur usage dans des opérations flottantes car elle provoque immédiatement un arrêt du code avec émission d'un signal (handler) qui peut être récupéré.

1.3 Paramètre lié à l'utilisation de la mémoire centrale

La longueur en octets de l'unité d'adressage de la machine est le seul paramètre accessible.

1.4 Paramètres d'utilisation des fichiers

Deux paramètres sont disponibles :
- la taille maximum en octets d'un fichier,
- la taille maximum en octets de l'ensemble des fichiers ouverts.

Ils ont été introduits uniquement en raison des contraintes liées à l'exploitation de ressources partagées (sur le serveur centralisé aster, limitation de l'espace temporaire de fichiers associé à un travail batch) et sont utilisés dans la gestion des fichiers d'accès direct binaire du gestionnaire de mémoire.

1.5 Constantes mathématiques

Un jeu de constantes universelles (optimales dans le type demandé) est fourni à l'utilisateur.
Ces constantes sont (actuellement) :
- les valeurs de π et de 2π,
- la valeur du zéro absolu pour la température,
- les paramètres de conversion radian/degré et degré/radian.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Les fonctions fournies et leur utilisation

2.1 Généralité

Pour aider à l'utilisation de ces fonctions, on s'est efforcé de codifier le nom des fonctions à partir de trois composantes :

- la nature GG de la grandeur :
  - LB longueur en bits,
  - LO longueur en octets,
  - MA valeur positive maximale,
  - MI valeur positive minimale,
  - GA gamme de valeurs.

- Le type TT de la grandeur à laquelle elle s'applique :
  - IS INTEGER (entier),
  - LS LOGICAL (logique),
  - R8 REAL*8 (réel),
  - C8 COMPLEX*8 (complexe).

- Le suffixe EM (Environnement Machine).

Pour les paramètres de longueur (l_) les fonctions sont de type entier. Les noms de fonction sont de la forme : GGTTEM.

Pour les paramètres dont la valeur dépend du type, le type de la fonction sera celui de la valeur recherchée. Dans ce cas les noms de fonction sont de la forme : TTGGEM.

Puisque ces fonctions ne font pas partie des fonctions intrinsèques du langage, il est indispensable de déclarer leur type dans chaque unité de programme utilisatrice.

2.2 Type logique

- Type logique standard (LOGICAL)

Type de la fonction	Paramètre fourni	Fonction ENVIMA	Plate-forme 64 bits	Plate-forme 32 bits
INTEGER	Longueur en octets	**LOLSEM**	8	4

2.3 Type entier

- Type entier standard (INTEGER)

Type de la fonction	Paramètre fourni	Fonction ENVIMA	Plate-forme 64 bits	Plate-forme 32 bits
INTEGER	Longueur en octets	**LOISEM**	8	4
INTEGER	Valeur maximale	**ISMAEM**	formule $2^{63}-1$	formule $2^{31}-1$ 2147483647

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2.4 Type réel

- Type réel standard (REAL*8)

Type de la fonction	Paramètre fourni	Fonction ENVIMA	Plate-forme 64 bits	Plate-forme 32 bits
REAL*8	Précision relative	**R8PREM**	2.220D-16	2.220D-16
REAL*8	Valeur maximale	**R8MAEM**	1.798D+308	1.798D+308
REAL*8	Valeur minimale	**R8MIEM**	2.225D-308	2.225D-308
REAL*8	Gamme	**R8GAEM**	1.340D+154	1.340D+154

2.5 Paramètres d'utilisation des fichiers

Ces paramètres sont utilisés en raisons des contraintes d'exploitation sur les serveurs dont les ressources sont partagées. Les bases (fichiers glob.i et vola.i) sont en général créées sur un espace partagé limité (/tmp).

Type de la fonction	Paramètre fourni	Fonction ENVIMA	Plate-forme 64 bits	Plate-forme 32 bits
INTEGER	Longueur maximum d'un fichier en octets	**LOFIEM**	12582912	2048000
INTEGER	Taille cumulée des fichiers en octets	**MOFIEM**	50331648	50331648

2.6 Affectation de constantes binaires

Pour la construction de masques sur les informations numériques, il est nécessaire d'accéder rapidement aux poids des bits. Ceux-ci sont accessibles par la fonction entière ISPBEM (numéro du bit).

ISPBEM(4)= 00000000000000000000000000001000 = $2^{4-1} = 8$

2.7 Affectation de constantes mathématiques ou physiques

Les constantes particulières sont accessibles à partir du type de la variable demandée :

Type de la fonction	Paramètre fourni	Fonction ENVIMA	Valeur
REAL*8	π	**R8PI**	3.141592653589793D+00
REAL*8	2π	**R8DEPI**	6.283185307179586D+00
REAL*8	Conversion radian/degré	**R8DGRD**	1.745329251994330D-02
REAL*8	Conversion degré/radian	**R8RDDG**	5.729577951308232D+01
REAL*8	Zéro absolu (température)	**R8T0**	273.15

2.8 Valeurs particulières

Ces valeurs sont disponibles sur toutes les machines possédant une arithmétique IEEE.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Type de la fonction	Paramètre fourni	Fonction ENVIMA	Valeur
INTEGER	Valeur provocant une erreur lors d'une opération flottante	**ISNNEM**	9223372036854775807.
REAL*8	Valeur provocant une erreur lors d'une opération flottante	**R8NNEM**	NaN
REAL*8	Valeur d'initialisation des structures de données	**R8VIDE**	1.797693134862316+308

2.9 Utilitaire

Le sous-programme Fortran IMPVEM permet d'éditer globalement les différentes valeurs caractéristiques de la machine de traitement sur le fichier de numéro d'unité logique IUL. Un exemple est donné en annexe.

CALL IMPVEM (IUL) : imprime toutes les caractéristiques des types standards.

2.10 Précautions d'emploi

Puisque ces fonctions ne font pas partie des fonctions intrinsèques du langage, il est indispensable de déclarer leur type dans chaque unité de programme utilisatrice. Dans ce cas les noms de fonction sont TTGGEM avec TT type de la fonction et GG nature de la grandeur (*cf.* exemple d'utilisation).

Remarque :

Les valeurs concernant les entiers sont dans le type implicite entier (fonctions commençant par une lettre L- ou N-).

Les fonctions ENVIMA sont disponibles dans la bibliothèque C utilitai de *Code_Aster*, et s'appuient pour la plupart sur les librairies standard C : fichiers d'en-tête <limits.h> et <float.h> [bib4].

Par définition toute opération de portage d'un sous ensemble de Code_Aster sur d'autres matériels nécessite le portage préalable d'ENVIMA. L'usage des en-têtes standard limite notablement une recherche approfondie dans la documentation du constructeur.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Bibliographie

[1] CODYW.J., Floating Point Parameters, Model and Standards - Relationship between numerical computations and programming languages,(REID J.K. Ed) North-Holland Amsterdam, 1982.

[2] FORD B., Parameterization of Environment for Transportable Numerical Software ACM trans. of Mathematical Software. Vol 4, n°2 June 1978 pp 100-103.

[3] I.E.E.E., Standard for floating-point arithmetic, ANSI/IEEE Std 754-1985, 1985.

[4] KERNIGHAN B.W., RITCHIE D.M., Le langage C - C ANSI, Masson Prentice Hall,1992.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

4 Annexe : Portabilité et Efficacité

Si les logiciels écrits avec ENVIMA sont portables, ENVIMA n'est pas portable : il nécessite une réécriture en utilisant les directives du pré processeur C.

Les règles d'utilisation des fonctions d'ENVIMA sont les mêmes que pour tout appel de fonction qui doit se faire hors des boucles internes.

L'efficacité est accrue en ce qui concerne l'homogénéisation et la portabilité du code qui voit disparaître de toutes les routines des instructions (ou des DATA) spécifiques à la machine.

L'efficacité est accrue en ce qui concerne la résolution numérique grâce aux valeurs qui définissent au mieux la précision, le maximum et le minimum, la gamme d'utilisation.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5 Annexe : Résultat du sous-programme impvem

Les impressions suivantes sont le résultat du sous programme impvem, elles peuvent être facilement obtenues en utilisant le fichier de commandes *Code_Aster* suivant :

```
DEBUT(DEBUG=_F(ENVIMA='TEST'));
FIN();
```

Voir le fichier resultat du cas test ZZZZ103.

Le fichier résultat produit sur une plate-forme Calibre6 (fort.8) contient les différentes valeurs caractéristiques imprimées dans un format décimal et héxa-décimal.

```
-------------------------------------------------
---- ENVIMA VERSION 97 MULTI MACHINES    --------
-------------------------------------------------
----- LONGUEUR EN BITS

 LBISEM  ENTIER   INTEGER              64

 ----- LONGUEUR EN OCTETS

 LOLSEM  LOGIQUE  LOGICAL               8
 LOISEM  ENTIER   INTEGER               8
 LOR8EM  REEL     REAL*8                8
 LOC8EM  COMPLEXE COMPLEX*16           16

 ----- LONGUEUR ET TAILLE DE FICHIER

 LOFIEM  EN OCTETS                    12582912
 MOFIEM  EN OCTETS                    50331648

 ----- ENTIER STANDARD

 ISMAEM  ENTIER   INTEGER             9223372036854775807  7FFFFFFFFFFFFFFF
 ISNNEM  ENTIER   INTEGER             9223372036854775807  7FFFFFFFFFFFFFFF

 ----- REAL*8


 FORMAT D'IMPRESSION DES FLOTTANTS (1X,A,1PD24.15,2X,Z16)

 R8BAEM  BASE NUMERATION    REAL*8       2.000000000000000D+00  4000000000000000
 R8PREM  PRECISION RELATIVE REAL*8       2.220446049250313D-16  3CB0000000000000
 R8MAEM  MAXIMAL            REAL*8       1.797693134862316+308  7FEFFFFFFFFFFFFF
 R8MIEM  MINIMAL            REAL*8       2.225073858507201-308    10000000000000
 R8GAEM  GAMME              REAL*8       1.340780792994259+154  5FEFFFFFFFFFFFFE
 R8NNEM  NOT A NUMBER       REAL*8      NaN                     FFF7FFFFFFFF0000
 R8VIDE  VIDE               REAL*8       1.797693134862316+308  7FEFFFFFFFFFFFFF
 RMIREM  B**-T                           1.110223024625157D-16  3CA0000000000000
 RMAREM  B**(1-T)                        2.220446049250313D-16  3CB0000000000000
 RMINEM  B**(EMIN-1)                     2.225073858507201-308    10000000000000
 RMAXEM  B**EMAX(1-B**(-T))              8.988465674311579+307  7FDFFFFFFFFFFFFF

 ----- POIDS DES BITS 1 A LBIS
 ISPBEM  BITS NUMEROTES DE DROITE A GAUCHE

   1                    1
   2                    2
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

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

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
  63   4611686018427387904

 ----- VALEURS PARTICULIERES  PI, DEPI, ...

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

-------------------------------------------------

---- FIN TEST ENVIMA MULTI MACHINES -------------

   OK

-------------------------------------------------
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)