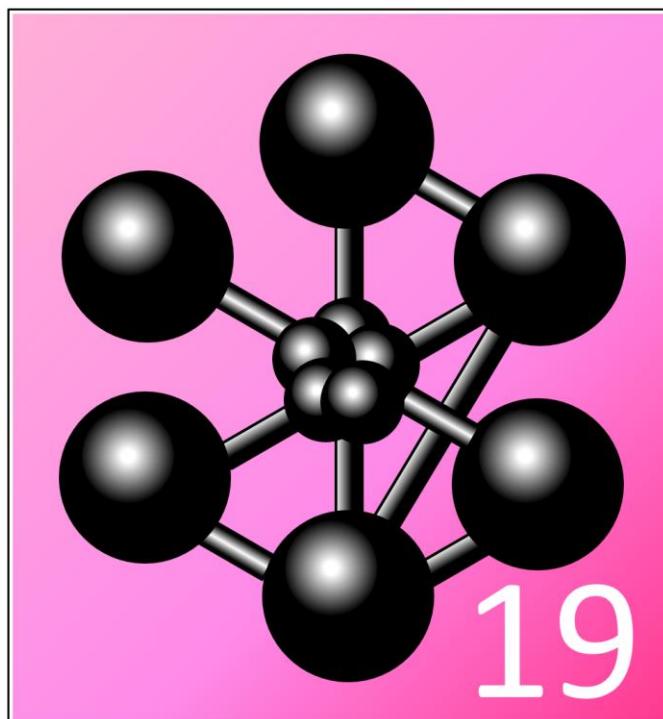


Cast3M 2019

Note de fabrication de Cast3M 2019



Cast3M est un logiciel de calcul par la méthode des éléments finis pour la mécanique des structures et des fluides. Cast3M est développé au Département de Modélisation des Systèmes et Structures (DM2S) de la Direction de l'Énergie Nucléaire du Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA).

Le développement de Cast3M entre dans le cadre d'une activité de recherche dans le domaine de la mécanique dont le but est de définir un instrument de haut niveau, pouvant servir de support pour la conception, le dimensionnement et l'analyse de structures et de composants.

Dans cette optique, Cast3M intègre non seulement les processus de résolution (solveur) mais également les fonctions de construction du modèle (pré-processeur) et d'exploitation des résultats (post-traitement). Cast3M est un logiciel « boîte à outils » qui permet à l'utilisateur de développer des fonctions répondant à ses propres besoins.

Cast3M est notamment utilisé dans le secteur de l'énergie nucléaire, comme outil de simulation ou comme plateforme de développement d'applications spécialisées. En particulier, Cast3M est utilisé par l'Institut de Radioprotection et de Sûreté Nucléaire (IRSN) dans le cadre des analyses de sûreté des installations nucléaires françaises.

SOMMAIRE

ASSURANCE QUALITÉ CAST3M	1
1. PRÉSENTATION DE CAST3M 2019	7
2. PLATEFORMES DE PRODUCTION DE CAST3M	8
2.1 PC – LINUX (32 BITS).....	8
2.2 PC – LINUX (64 BITS).....	8
2.3 PC – WINDOWS (32 BITS)	8
2.4 PC – WINDOWS (64 BITS)	8
2.5 MAC – MAC OS X (64 BITS).....	8
3. ÉLABORATION DE LA VERSION 2019 DE CAST3M	9
3.1 OBJET	9
3.2 ÉTAPES DE L'ÉLABORATION DE LA VERSION.....	9
3.2.1 <i>Phase 1 : le 31/12 de l'année « N-1 »</i>	9
3.2.2 <i>Phase 2 : du 01/01 au 28/02 de l'année « N »</i>	13
3.2.3 <i>Phase 3 : du 01/03 au 31/03 de l'année « N »</i>	14
3.2.3.1 <i>Compilateurs GNU Linux 64-bits</i>	14
3.2.3.2 <i>Compilateurs GNU Linux 32-bits</i>	14
3.2.3.3 <i>Compilateurs GNU Windows 64-bits</i>	15
3.2.3.4 <i>Compilateurs GNU Windows 32-bits</i>	16
3.2.3.5 <i>Compilateurs GNU MAC OSX 64 bits</i>	17
3.2.3.6 <i>Librairies externes</i>	17
3.2.3.7 <i>Traducteur Esope vers FORTRAN77.....</i>	18
3.2.3.8 <i>Compilation de Cast3M</i>	18
3.3 SCHÉMA DE PRINCIPE DE LA PRÉPARATION DES VERSIONS ANNUELLES DE CAST3M.....	20
ANNEXE A. SCRIPT REPERCUTER.SH	21
ANNEXE B. SCRIPT ENVIRONNEMENT_CAST3M19.BAT.....	24
ANNEXE C. SCRIPTS COMPILCAST19.BAT ET COMPILCAST19	25
ANNEXE D. SCRIPTS ESSAICAST19.BAT ET ESSAICAST19	33
ANNEXE E. SCRIPTS CAST_UTIL19.BAT ET CAST_UTIL19.....	39
ANNEXE F. SCRIPTS CASTEM19.BAT ET CASTEM19	41
ANNEXE G. DOCUMENTATION CAST3M	50
ANNEXE H. TRAÇABILITÉ	52

1. PRÉSENTATION DE CAST3M 2019

Cast3M est un logiciel développé au Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA) qui a pour objet la résolution d'équations aux dérivées partielles par la méthode des éléments finis.

Les domaines d'applications sont la mécanique des structures, la mécanique des fluides, la thermique et la magnétostatique.

En mécanique des structures, le logiciel permet la résolution de problèmes métier tels que la plasticité, le flambage, le fluage, l'analyse sismique, la thermoplastique, la mécanique de la rupture, le post-flambage, la ruine des structures. Les structures étudiées sont 2D ou 3D et de nombreuses lois de comportement de matériaux sont implémentées.

En magnétostatique, les possibilités sont les analyses linéaires d'un champ magnétique en 2D ou 3D, les analyses non linéaires pour des matériaux avec des caractéristiques dépendant du champ magnétique, le calcul du champ de Biot et de Savart, et en électrostatique les calculs des potentiels scalaire et vecteur.

En mécanique des fluides, de nombreux modèles physiques sont disponibles, notamment des modèles d'écoulement (écoulements incompressibles ou dilatables, écoulements à faible nombre de Mach, écoulements compressibles, écoulements multi-espèces réactifs ou non, modèles de turbulence, diphasique homogène équilibré ou diphasique bi-fluide), des modèles homogénéisés (Navier-Stockes en milieu chargé, équations d'énergie), des modèles de combustion (cinétique Arrhenius, modèles EBU ou corrélations, modèles de recombinateur catalytique), et des modèles de condensation (condensation en paroi (Chilton-Colburn), condensation en masse).

Cast3M est un code muni d'un langage interprété de mise en données appelé GIBIANE.

L'utilisateur développe des jeux de données GIBIANE appelant des opérateurs qui agissent sur des opérandes dans le but de créer un résultat. Cast3M peut être considéré comme une boîte à outils comprenant plus de 500 opérateurs mis à la disposition des utilisateurs. Il comprend notamment des fonctionnalités de maillage et de post-traitement.

Cast3M est disponible sous 2 licences : licence « éducation et recherche » et licence « industrielle ».

- La licence « éducation et recherche » est réservée aux organismes de recherche, aux enseignants ainsi qu'aux étudiants. Elle est gratuite et se décline en version « *utilisateur* » ou en version « *développeur* ». Pour les versions développeur, un exécutable Esope est fourni avec l'exécutable de Cast3M dans le but de traduire les programmes Esope vers des programmes en fortran 77.
- La licence « industrielle » est, quant à elle, payante et ne se décline qu'en version « *utilisateur* ».

2. PLATEFORMES DE PRODUCTION DE CAST3M

Les plates-formes sur lesquelles est fabriquée la version annuelle de Cast3M sont les suivantes :

2.1 PC – LINUX (32 BITS)

Plateforme de compilation et de test :

Modèle de système	: Mandriva Linux release 2009.0-i586
Type de processeur	: x86_64 Intel® Xeon™ 3,20GHz (2 cœurs)
Mémoire Vive	: 2Go

2.2 PC – LINUX (64 BITS)

Plateforme de compilation et de test :

Modèle de système	: Mandriva Linux release 2009.0-x86_64
Type de processeur	: x86_64 Intel® Xeon™ 3,20GHz (2 cœurs)
Mémoire Vive	: 2Go

2.3 PC – WINDOWS (32 BITS)

Plateforme de compilation (Cross Compilation) :

Modèle de système	: Windows 7 Entreprise, Service Pack 1
Type de processeur	: Intel® core™ 2 Duo CPU E8500, 2 CPU, 3,16GHz
Mémoire Vive	: 8Go

Plateforme de test :

Modèle de système	: Windows XP Professionnel 2002, Service Pack 3
Type de processeur	: Intel® pentium™ 4 2 CPU, 3,00GHz
Mémoire Vive	: 1,5Go (1024Mo + 512Mo)

2.4 PC – WINDOWS (64 BITS)

Plateforme de compilation et de test :

Modèle de système	: Windows 7 Entreprise, Service Pack 1
Type de processeur	: Intel® core™ 2 Duo CPU E8500, 2 CPU, 3,16GHz
Mémoire Vive	: 8Go

2.5 MAC – MAC OS X (64 BITS)

Plateforme de compilation et de test :

Modèle de système	: MAC OS Mojave 10.14
Type de processeur	: Intel® core™ i5, 2 CPU, 2,6GHz
Mémoire Vive	: 8Go

3. ÉLABORATION DE LA VERSION 2019 DE CAST3M

3.1 OBJET

L'objectif est de produire une version annuelle de Cast3M en vue d'une large diffusion par téléchargement, notamment sur le site internet (<http://www-cast3m.cea.fr>), ainsi que ses programmes d'installation automatisés pour différentes plates-formes informatiques :

- Windows (32/64 bits)
- Linux (32/64 bits)
- MAC OSX (64 bits)

3.2 ÉTAPES DE L'ÉLABORATION DE LA VERSION

Pour produire la version de l'année « N » de Cast3M, les actions suivantes sont réalisées par ordre chronologique.

3.2.1 Phase 1 : le 31/12 de l'année « N-1 »

Cette phase consiste à saisir la version de développement de Cast3M sur le réseau du SEMT et comprend des sources C, des sources ESOPE, des procédures, des notices, des cas-tests et un fichier d'erreur GIBI.ERREUR.

L'ensemble des manipulations décrites pour la préparation de la version de l'année N de Cast3M est organisé au sein d'un répertoire portant comme nom l'année de la version de Cast3M à préparer (ex : 2019). La Figure 1 décrit l'arborescence de ce répertoire. Le contenu de chacun de ces répertoires est détaillé dans les paragraphes qui suivent.

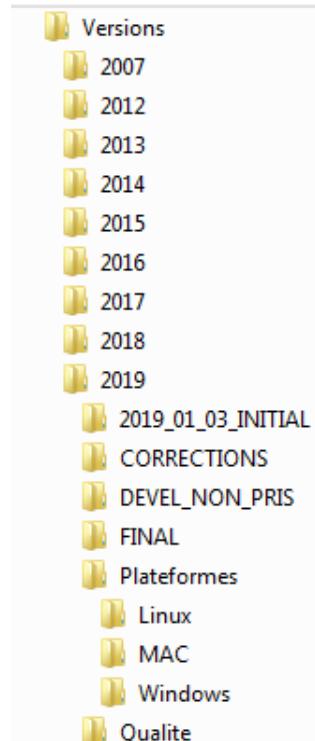


Figure 1. Arborescence des répertoires pour la préparation de Cast3M 2019

- Le répertoire 2019_01_03_INITIAL contient la saisie de la version de développement de Cast3M vierge de toutes corrections. Ce répertoire n'est en aucun cas modifié au cours des opérations suivantes (voir §3.2.2 et §3.2.3). Le contenu de ce répertoire est constitué de fichiers et de dossiers à récupérer sur le disque /u2/castem/ du réseau du SEMT accessible depuis le serveur semtpc0 ou semt2. La liste des répertoires à récupérer est donnée sur la Figure 2.

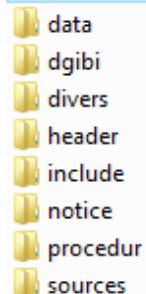


Figure 2. Liste des répertoires à récupérer dans /u2/castem/ sur le réseau du SEMT

- dgibi : cas tests de la version de développement de Cast3M.
- divers : fichiers de données externes nécessaires pour l'exécution de la base des cas tests.
- header : fichiers d'entêtes.
- include : comprend l'ensemble des includes ESOPE nécessaires.
- notice : comprend l'ensemble des notices disponibles dans Cast3M.
- procedur : comprend l'ensemble des procédures disponible dans Cast3M.
- data : répertoire supplémentaire créé pour y placer le fichier d'erreur de Cast3M à récupérer sur le réseau du SEMT à l'adresse suivante : /u2/castem/CAST3M.ERREUR
- sources : répertoire supplémentaire créé pour y placer l'ensemble des sources .c, .eso, .h. Ces fichiers sont extraits sous un environnement linux dans le shell script bash à l'aide de la commande suivante :

```
arc -eon castem.arc '*.*'
```

COMMANDÉ D'EXTRACTION DES SOURCES

- **Note** : Afin d'optimiser la taille qu'occupe Cast3M sur le disque mais également la taille des paquets d'installation, les caractères « espaces blancs » inutiles en fin de ligne sont retirés.
 - o **12%** de gain de place pour les procédures et notices.
 - o **15%** de gain de place pour les sources.
- Le répertoire FINAL (voir Figure 1) contient l'ensemble des répertoires communs à toutes les plates-formes mis à jour de toutes les corrections d'anomalies.

- Le répertoire CORRECTIONS contient l'ensemble des répertoires faisant état des corrections d'anomalies survenues durant la phase 2 (Voir §3.2.2). Les fichiers impactés par ces corrections sont consultables sur le site Cast3M dans l'onglet Anomalies/Consulter (<http://www-cast3m.cea.fr/index.php?page=anomalies>) ou dans le fichier /u2/castem/hist.hist sur le réseau SEMT.
 - La lecture des fiches d'anomalies permet de distinguer entre les corrections d'anomalie et les nouveaux développements intervenus durant cette période.
 - Seuls les fichiers impactés par une correction d'anomalie sont pris en compte.
 - Pour chacune des anomalies récupérées, un sous-dossier est créé. Son nom correspond à la date de la correction de l'anomalie au format AAAA_MM_JJ.
 - Chacun de ces dossiers contient les fichiers impactés qui peuvent être de plusieurs types :
 - .eso,
 - .INC,
 - .c,
 - .h,
 - .dgibi,
 - .procedur,
 - GIBI.ERREUR.
- Le répertoire Plateformes contient les répertoires et fichiers nécessaires à la fabrication de Cast3M sur les 3 plates-formes supportées qui sont Linux, MAC et Windows.
 - Pour chacune des plates-formes un répertoire Dev est créé et contient les sous-dossiers présentés sur la Figure 3. Le script Repercuter.sh génère automatiquement l'arborescence complète à sa première utilisation et cela pour toutes les plates-formes ; voir Annexe A.
 - Le répertoire dgibi_make_version contient l'ensemble des jeux de données de Cast3M et servira à exécuter la base des cas-tests (Voir §3.2.3).
 - Un répertoire GCC contient l'ensemble des outils de développement GNU.

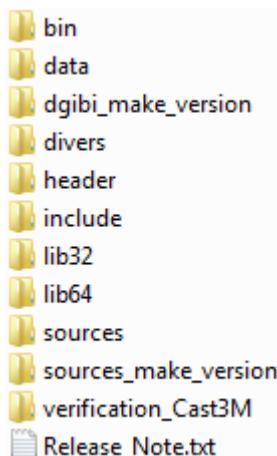


Figure 3. Arborescence du répertoire Dev de toutes les plates-formes

- Le répertoire `bin` contient tous les scripts, exécutables et fichiers d'aide. Il faut reprendre les scripts et binaires de la dernière version stable de Cast3M dont la liste est la suivante :

- <code>ftnchek-32bits</code>	
- <code>ftnchek-64bits</code>	
- <code>cast_evol19</code>	
- <code>cast_UTIL19</code>	
- <code>castem19</code>	
- <code>compilcast19</code>	
- <code>essaicast19</code>	
- <code>environnement_Cast3M19</code>	(Windows seulement)
- <code>LOGO_ASCII_19.txt</code>	(UNIX & Windows différents)
- <code>liste_rlwrap19</code>	(UNIX seulement)
- <code>rlwrap_32</code>	(UNIX seulement)
- <code>rlwrap_64</code>	(UNIX seulement)

Chaque script est renommé avec l'année en cours dans leur nom pour assurer la cohabitation des versions successives.

- Le répertoire `sources_make_version` (arborescence présentée sur la Figure 4) contient les fichiers nécessaires pour compiler les sources des différentes architectures (32-bits ou 64-bits).

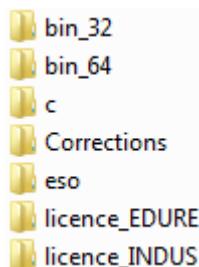


Figure 4. Arborescence du répertoire `sources_make_version`

- Les répertoires `c` et `eso` contiennent respectivement les sources C et Esope de Cast3M dans leur version initiale.
- Le répertoire `Corrections` contient toutes les sources C et Esope ayant fait l'objet de corrections d'anomalie.
- Les répertoires `bin_32` et `bin_64` contiennent toutes les sources compilées ainsi que les sources FORTRAN 77 (`.f`) issues de la traduction des sources Esope. Cela permet de conserver séparément les architectures 32-bits et 64-bits.
- Les répertoires `licence_EDURE` et `licence_INDUS` contiennent respectivement les sources spécifiques permettant de différencier la version « Éducation & Recherche » de la version « Industrielle » de Cast3M.

- Sources spécifiques pour la licence « Éducation & Recherche »*

- `defdat.eso`
 - `fin.eso`
 - `journa.eso`
 - `pilot.eso`
 - `verdat.c`

- Sources spécifiques pour la licence « Industrielle »*

- `pilot.eso`
 - `journa.eso`
 - `fin.eso`
 - `perm.c`

3.2.2 Phase 2 : du 01/01 au 28/02 de l'année « N »

Cette phase consiste à répercuter les corrections apportées dans la version du jour de Cast3M vers les différentes plates-formes.

L'ensemble des évolutions de Cast3M est répertorié dans le fichier `/u2/castem/hist.hist` sur le réseau SEMT et est consultable en ligne sur le site Cast3M (<http://www-cast3m.cea.fr/index.php?page=anomalies>). La vérification des évolutions versées est effectuée quotidiennement durant cette période. Seuls les fichiers affectés par les corrections d'anomalies sont intégrés à la nouvelle version de Cast3M, les nouveaux développements sont omis.

Pour récupérer des corrections, un répertoire dont le nom correspond au jour de versement des corrections (au format `AAAA_MM_JJ`) est créé dans le répertoire `Corrections`. Les fichiers affectés y sont copiés. Ces corrections peuvent concerner les fichiers de plusieurs types : `.eso`, `.c`, `.dgibi`, `.procedur`, `.notice`, `GIBI.ERREUR`, `.INC` ou `.h`.

- les fichiers `.eso`, `.c` et `.h`, sont à extraire de l'archive `/u2/castem/castem.arc`
- les fichiers `.dgibi` sont à récupérer dans le répertoire `/u2/castem/dgibi`
- les fichiers `.procedur` sont à récupérer dans le répertoire `/u2/castem/procedur`
- les fichiers `.notice` sont à récupérer dans le répertoire `/u2/castem/notice`
- les fichiers `.INC` sont à récupérer dans le répertoire `/u2/castem/include`

La répercussion de ces fichiers corrigés dans tous les répertoires de chacune des distributions est réalisée par le script `Repercuter.sh` (voir Annexe A) avec comme arguments le ou les répertoires contenant les corrections d'anomalies par ordre chronologique.

- Lorsque des sources (`.c` ou `.eso`) ou includes (`.INC` ou `.h`) sont présents, les exécutables et librairies sont automatiquement supprimés dans chacune des plates-formes (garde-fou obligeant à les générer à nouveau).
- Lorsque des fichiers `.notice` ou `.procedur` sont présents, les fichiers d'accès direct `CAST3M.MASTER` et `CAST3M.PROC` sont automatiquement supprimés dans chacune des plates-formes (garde-fou obligeant à les générer à nouveau).

3.2.3 Phase 3 : du 01/03 au 31/03 de l'année « N »

Cette phase consiste à porter Cast3M sur l'ensemble des plates-formes supportées (Windows 32/64-bits, LINUX 32/64-bits et MAC OSX 64-bits).

Afin de réaliser cette tâche, il faut disposer d'une suite de compilateurs. Ensuite, il sera nécessaire de compiler quelques librairies externes à Cast3M avant de compiler Cast3M à proprement parler.

3.2.3.1 Compilateurs GNU Linux 64-bits

Configuration de la suite de compilateurs GNU pour **Linux 64-bits** :

```
Using built-in specs.

COLLECT_GCC=../GCC/GCC-x86_64/bin/gcc-8.2.0
COLLECT_LTO_WRAPPER=/mnt/extra2/izpack/install/GCC/GCC-
x86_64/bin/../libexec/gcc/x86_64-pc-linux-gnu/8.2.0/lto-wrapper
Target: x86_64-pc-linux-gnu

Configured with: ../gcc-8.2.0/configure --
prefix=/home/kk2000/Document/CAST3M19/gcc/prefix/ --disable-bootstrap --disable-
libquadmath --disable-libquadmath-support --program-suffix=-8.2.0 --with-
gmp=/home/kk2000/Document/CAST3M19/gcc/prefix/ --with-
mpfr=/home/kk2000/Document/CAST3M19/gcc/prefix/ --with-
mpc=/home/kk2000/Document/CAST3M19/gcc/prefix/ --with-
isl=/home/kk2000/Document/CAST3M19/gcc/prefix/ CC=/usr/local/bin/gcc
CXX=/usr/local/bin/g++ AR=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/ar
AS=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/as
LD=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/ld
NM=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/nm
RANLIB=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/ranlib
STRIP=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/strip
OBJCOPY=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/objcopy
OBJDUMP=/home/kk2000/Document/CAST3M19/gcc/prefix/bin/objdump --enable-
languages=c,c++,fortran,lto --no-create --no-recursion

Thread model: posix
gcc version 8.2.0 (GCC)
```

3.2.3.2 Compilateurs GNU Linux 32-bits

Configuration de la suite de compilateurs GNU pour **Linux 32-bits** :

```
Cible : i686-pc-linux-gnu
Configuré avec: ../configure --enable-bootstrap --disable-libquadmath --disable-
libquadmath-support --enable-languages=c,c++,fortran --program-suffix=-6.3.0 --
with-mpc=/home/kk2000/Tools/mpc-1.0.3 --with-mpfr=/home/kk2000/Tools/mpfr-3.1.5 --
with-gmp=/home/kk2000/Tools/gmp-6.1.2 --with-isl=/home/kk2000/Tools/isl-0.18
Modèle de thread: posix
gcc version 6.3.0 (GCC)
```

3.2.3.3 Compilateurs GNU Windows 64-bits

On utilise **MinGW x86_64-8.1.0-release-posix-seh-rt_v6-rev0** qui contient la suite de compilateurs GNU précompilée et utilisable nativement sous Windows. Ce paquet est téléchargeable sur le site : <https://sourceforge.net/projects/mingw-w64/files/>

Configuration de la suite de compilateurs GNU pour **Windows 64-bits** :

```
Using built-in specs.

COLLECT_GCC=gcc

COLLECT_LTO_WRAPPER=C:/Cast3M/PCW/MinGW/x86_64-8.1.0-release-posix-seh-rt_v6-
rev0/mingw64/bin/../libexec/gcc/x86_64-w64-mingw32/8.

1.0/lto-wrapper.exe

Target: x86_64-w64-mingw32

Configured with: ../../src/gcc-8.1.0/configure --host=x86_64-w64-mingw32 --
build=x86_64-w64-mingw32 --target=x86_64-w64-mingw32 --prefix=/mingw64 --with-
sysroot=/c/mingw810/x86_64-810-posix-seh-rt_v6-rev0/mingw64 --enable-shared --
enable-static --disable-multilib --enable-languages=c,c++,fortran,lto --enable-
libstdcxx-time=yes --enable-threads=posix --enable-libgomp --enable-libatomic --
enable-lto --enable-graphite --enable-checking=release --enable-fully-dynamic-
string --enable-version-specific-runtime-libs --disable-libstdcxx-pch --disable-
libstdcxx-debug --enable-bootstrap --disable-rpath --disable-win32-registry --
disable-nls --disable-werror --disable-symvers --with-gnu-as --with-gnu-ld --
with-arch=nocona --with-tune=core2 --with-libiconv --with-system-zlib --with-
gmp=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-
mpfr=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-
mpc=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-
isl=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-
pkgversion='x86_64-posix-seh-rev0, Built by MinGW-W64 project' --with-
bugurl=https://sourceforge.net/projects/mingw-w64 CFLAGS='-O2 -pipe -fno-ident -
I/c/mingw810/x86_64-810-posix-seh-rt_v6-rev0/mingw64/opt/include -
I/c/mingw810/prerequisites/x86_64-zlib-static/include -
I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include' CXXFLAGS='-O2 -
pipe -fno-ident -I/c/mingw810/x86_64-810-posix-seh-rt_v6-
rev0/mingw64/opt/include -I/c/mingw810/prerequisites/x86_64-zlib-static/include -
-I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include' CPPFLAGS=' -
I/c/mingw810/x86_64-810-posix-seh-rt_v6-rev0/mingw64/opt/include -
I/c/mingw810/prerequisites/x86_64-zlib-static/include -
I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include' LDFLAGS=' -pipe -
fno-ident -L/c/mingw810/x86_64-810-posix-seh-rt_v6-rev0/mingw64/opt/lib -
L/c/mingw810/prerequisites/x86_64-zlib-static/lib -
L/c/mingw810/prerequisites/x86_64-w64-mingw32-static/lib ' 

Thread model: posix
gcc version 8.1.0 (x86_64-posix-seh-rev0, Built by MinGW-W64 project)
```

3.2.3.4 Compilateurs GNU Windows 32-bits

On utilise **MinGW** i686-6.4.0-release-posix-dwarf-rt_v5-rev0 qui contient la suite de compilateurs GNU précompilée et utilisable nativement sous Windows. Ce paquet est téléchargeable sur le site : <https://sourceforge.net/projects/mingw-w64/files/>

Configuration de la suite de compilateurs GNU pour **Windows 32-bits** :

```
Using built-in specs.

COLLECT_GCC=gcc

COLLECT_LTO_WRAPPER=C:/Cast3M/PCW/MinGW/i686-6.4.0-release-posix-dwarf-rt_v5-
rev0/mingw32/bin/../libexec/gcc/i686-w64-mingw32/6.4.

0/lto-wrapper.exe

Target: i686-w64-mingw32

Configured with: ../../src/gcc-6.4.0/configure --host=i686-w64-mingw32 --
build=i686-w64-mingw32 --target=i686-w64-mingw32 --prefix=/mingw32 --with-
sysroot=/c/mingw640/i686-640-posix-dwarf-rt_v5-rev0/mingw32 --enable-shared --
enable-static --disable-multilib --enable-languages=c,c++,fortran,lto --enable-
libstdcxx-time=yes --enable-threads=posix --enable-libgomp --enable-libatomic --
enable-lto --enable-graphite --enable-checking=release --enable-fuzzy-dynamic-
string --enable-version-specific-runtime-libs --enable-libstdcxx-fsyste-
ts=yes --disable-sjlj-exceptions --with-dwarf2 --disable-libstdcxx-pch --
disable-libstdcxx-debug --enable-bootstrap --disable-rpath --disable-win32-
registry --disable-nls --disable-werror --disable-symvers --with-gnu-as --with-
gnu-ld --with-arch=i686 --with-tune=generic --with-libiconv --with-system-zlib -
-with-gmp=/c/mingw640/prerequisites/i686-w64-mingw32-static --with-
mpfr=/c/mingw640/prerequisites/i686-w64-mingw32-static --with-
mpc=/c/mingw640/prerequisites/i686-w64-mingw32-static --with-
isl=/c/mingw640/prerequisites/i686-w64-mingw32-static --with-pkgversion='i686-
posix-dwarf-rev0, Built by MinGW-W64 project' --with-
bugurl=https://sourceforge.net/projects/mingw-w64 CFLAGS='-O2 -pipe -fno-ident -
I/c/mingw640/i686-640-posix-dwarf-rt_v5-rev0/mingw32/opt/include -
I/c/mingw640/prerequisites/i686-zlib-static/include -
I/c/mingw640/prerequisites/i686-w64-mingw32-static/include' CXXFLAGS=' -fno-ident -
-I/c/mingw640/i686-640-posix-dwarf-rt_v5-rev0/mingw32/opt/include -
I/c/mingw640/prerequisites/i686-zlib-static/include -
I/c/mingw640/prerequisites/i686-w64-mingw32-static/include' CPPFLAGS=' -
-I/c/mingw640/i686-640-posix-dwarf-rt_v5-rev0/mingw32/opt/include -
I/c/mingw640/prerequisites/i686-zlib-static/include -
I/c/mingw640/prerequisites/i686-w64-mingw32-static/include' LDFLAGS=' -fno-ident -
-L/c/mingw640/i686-640-posix-dwarf-rt_v5-rev0/mingw32/opt/lib -
L/c/mingw640/prerequisites/i686-zlib-static/lib -
L/c/mingw640/prerequisites/i686-w64-mingw32-static/lib -Wl,--large-address-
aware'

Thread model: posix
gcc version 6.4.0 (i686-posix-dwarf-rev0, Built by MinGW-W64 project)
```

3.2.3.5 Compilateurs GNU MAC OSX 64 bits

Configuration de la suite de compilateurs GNU pour **MAC OSX 64-bits** :

```
Utilisation des specs internes.

COLLECT_GCC=gcc-8.2.0

COLLECT_LTO_WRAPPER=/Users/kk2000/Documents/CAST3M19/Dev/GCC/GCC-
x86_64/bin/..../libexec/gcc/x86_64-apple-darwin18.0.0/8.2.0/lto-wrapper

Cible : x86_64-apple-darwin18.0.0

Configuré avec: ../gcc-8.2.0/configure --
prefix=/Users/kk2000/Documents/CAST3M19/gcc/prefix --enable-
language=c,c++,fortran --program-suffix=-8.2.0 --with-
gmp=/Users/kk2000/Documents/CAST3M19/gcc/prefix --with-
mpfr=/Users/kk2000/Documents/CAST3M19/gcc/prefix --with-
mpc=/Users/kk2000/Documents/CAST3M19/gcc/prefix --with-
isl=/Users/kk2000/Documents/CAST3M19/gcc/prefix : (reconfigured) ../gcc-
8.2.0/configure --prefix=/Users/kk2000/Documents/CAST3M19/gcc/prefix --enable-
bootstrap --enable-language=c,c++,fortran --program-suffix=-8.2.0 --with-
gmp=/Users/kk2000/Documents/CAST3M19/gcc/prefix --with-
mpfr=/Users/kk2000/Documents/CAST3M19/gcc/prefix --with-
mpc=/Users/kk2000/Documents/CAST3M19/gcc/prefix --with-
isl=/Users/kk2000/Documents/CAST3M19/gcc/prefix

Modèle de thread: posix
gcc version 8.2.0 (GCC)
```

Remarques spécifiques à MAC OSX :

- Il est nécessaire de disposer des outils de développement qui peuvent être installés grâce à la commande `sudo xcode-select --install` (nécessité d'être connecté en tant que 'admin-local'). Il est également nécessaire de disposer d'une installation fonctionnelle de X11 (XQuartz), pour le fonctionnement des tracés X et OpenGL ;
- La version de `rlwrap` compilée à partir des sources comporte des anomalies (par exemple : impossibilité d'effacer la saisie lors de l'écriture d'une ligne de commande dans le terminal), la version livrée utilise donc la version installable avec `homebrew` (installable avec la commande `/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`) ;
- Les versions de `ar` et de `ranlib` utilisées par défaut étaient les versions GNU, il a été nécessaire de renommer ces dernières (situées dans `/usr/local/bin`) en « `gnu_ar` » et « `gnu_ranlib` » ainsi que de créer des liens symboliques vers les versions de `ar` et de `ranlib` fournies par `Xcode`, qui se trouvent dans `/usr/bin`. Sans cette manipulation, `make` s'interrompait lors de la construction de certaines librairies et il était impossible de construire celles-ci.

3.2.3.6 Librairies externes

Toutes les librairies nécessaires doivent être compilées avec la version de GCC utilisée pour Cast3M. On utilise toujours la librairie statique (fichier « .a ») lorsque cela est possible, renommée en ajoutant l'architecture (32 ou 64 bits) et placée dans le répertoire `lib32/` ou `lib64/` correspondant du répertoire Cast3M de chaque plateforme. Ces librairies sont :

- **XDR** – Uniquement pour Linux. Récupéré à l'aide de la commande `urpmi libtirpc-devel`. L'installation dépend de l'architecture :
 - `/usr/lib64/libtirpc.a` (Linux-x86_64)
 - `/usr/lib/libtirpc.a` (Linux-i586)
- **FXDR** - La version 32 bits est la version disponible sur http://meteora.ucsd.edu/~pierce/fxdr_home_page.html et la version 64 bits est une version modifiée disponible sur le réseau du SEMT ;

- **JPEG** - Prendre la dernière version disponible sur <http://www.iijg.org/> au 03/01/2019. La version utilisée pour Cast3M 2019 est la v9c ;
- **ZLIB** - Prendre la dernière version disponible sur <https://www.zlib.net/> au 03/01/2019. La version utilisée pour Cast3M 2019 est la 1.2.11 ;
- **HDF5** - Prendre la version compatible avec MED, sur <https://support.hdfgroup.org/ftp/HDF5/>. La version utilisée pour Cast3M 2019 est la 1.8.14 ;
- **MED** - Prendre une version compatible avec Cast3M. La version utilisée pour Cast3M 2019 est la 3.3.1 ;
- **FreeGLUT** - Seulement pour Linux et Windows puisque XQuartz fournit une émulation de GLUT. La version utilisée pour Cast3M 2019 est la 3.0.0. Disponible sur <http://freeglut.sourceforge.net/> ;
- **OpenMPI** - Prendre la dernière version disponible sur <https://www.open-mpi.org/> au 03/01/2019. La version utilisée pour Cast3M 2019 est la 4.0.0. Indisponible pour Windows et non fonctionnelle sur MAC OSX. On utilise la version dynamique de MPI pour permettre son éventuel remplacement sans avoir à refaire l'édition des liens (Application au TGCC) ;
- **ESOPE_GEMAT** : La dernière version du gestionnaire de mémoire ESOPE est disponible sur le réseau du SEMT : /u2/esope/esope2018.10.

3.2.3.7 Traducteur Esope vers FORTRAN77

Les sources du traducteur Esope doivent être compilées avec la version de GCC utilisée pour Cast3M, pour cela il est nécessaire de disposer d'un traducteur Esope fonctionnel (repris de l'année précédente).

- Compilation des sources avec la commande `compilcast19 -ESOPE *.eso *.c`
- Edition des liens avec la commande `essaicast19 -ESOPE`

Afin de vérifier que l'exécutable généré est fonctionnel, on effectue l'opération une deuxième fois mais avec le traducteur fraîchement créé (« bootstrap »). La comparaison de l'exécutable généré et de l'exécutable utilisé pour le générer (ils doivent être identiques) permet de s'assurer que la traduction puis la compilation donnent le même résultat.

3.2.3.8 Compilation de Cast3M

Les sources (.eso et .c) sont compilées à l'aide de la commande `compilcast19 -f *.eso *.c` (voir Annexe C) :

- En cas d'erreur de traduction, un fichier .lst portant le préfixe de la source est généré. Une analyse préliminaire de l'erreur permettra d'amorcer la discussion avec les développeurs de Cast3M afin que la correction appropriée soit apportée ;
- En cas d'erreur de compilation, un fichier .log portant le préfixe de la source est généré. Une analyse préliminaire de l'erreur permettra d'amorcer la discussion avec les développeurs de Cast3M afin que la correction appropriée soit apportée ;
- Toute erreur doit être signalée par l'émission d'une fiche d'anomalie dans l'atelier logiciel de Cast3M : dial20 (sur semt2). Celle-ci comprendra le nom de la source, la ou les plates-formes ainsi que l'architecture en question.

Les fichiers .o et .f obtenus sont déplacés respectivement dans les répertoires bin_32 ou bin_64, où sera effectuée l'édition des liens à l'aide du script essaicast19 (voir Annexe D), selon l'architecture en cours de compilation :

- Lors de la toute première tentative de faire l'édition des liens, ni l'exécutable ni la librairie de Cast3M n'existent. Cette commande les génère et les place automatiquement au bon endroit ;
- Durant l'édition des liens, il se peut que certaines erreurs surviennent. Le cas échéant un fichier link_cast_64_19.log est généré et contient les messages d'erreurs. Les plus classiques sont listées ci-dessous :
 - undefined reference to `flush_`
➤ Mauvais depmac.eso
 - undefined reference to `std::ios_base::Init::~Init()`

- Ajouter la librairie standard c++ dans les directives : -lstdc++
- undefined reference to `crt1.o'
- Ajouter le chemin « système » où se trouve l'objet crt1.o dans la variable d'environnement LIBRARY_PATH

Une fois le portage effectué, Cast3M est vérifié et validé. Ceci consiste à exécuter l'ensemble des cas-tests du répertoire dgibi_make_version et ce pour toutes les plates-formes et toutes les architectures supportées. La commande castem19 -test (voir Annexe F) permet d'effectuer cette manipulation :

- En cas d'erreur d'exécution d'un cas test, un fichier .err portant le préfixe du cas test est généré. Une analyse préliminaire de l'erreur permettra d'amorcer la discussion avec les développeurs de Cast3M afin que la correction appropriée soit apportée ;
- Toute erreur doit être signalée par l'émission d'une fiche d'anomalie dans l'atelier logiciel de Cast3M : dial20 (sur semt2). Celle-ci comprendra le nom du cas-test, la plate-forme ainsi que l'architecture en question.

3.3 SCHÉMA DE PRINCIPE DE LA PRÉPARATION DES VERSIONS ANNUELLES DE CAST3M

La Figure 5 schématisise les points précédents et met en évidence la manière dont est gérée l'élaboration d'une version annuelle de Cast3M.

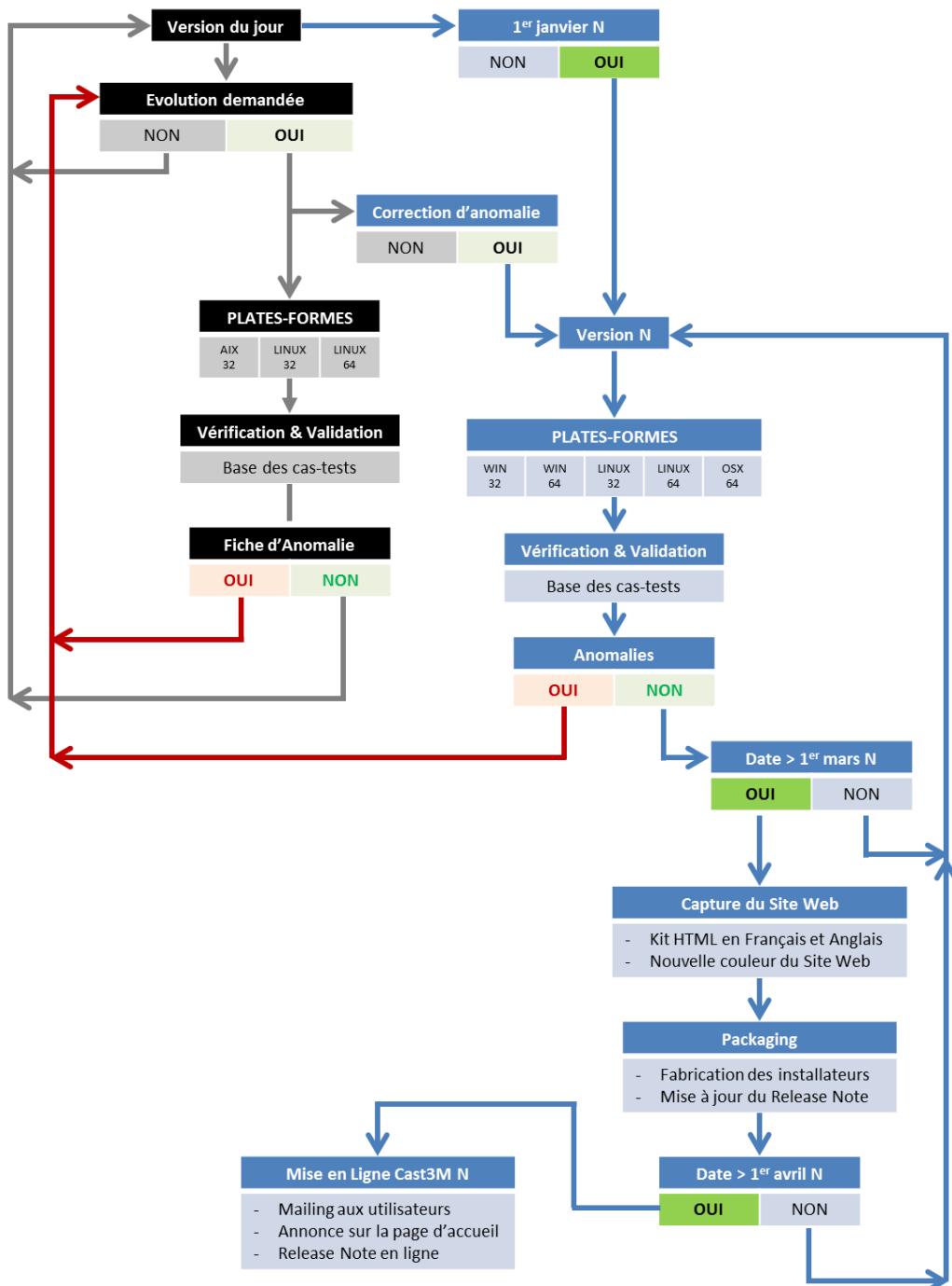


Figure 5. Organigramme de la préparation des versions annuelles de Cast3M



Annexe A. Script Repercuter.sh

```
#!/bin/bash

#####
# Fonctionnalites du script Repercuter.sh:
#
# Ce Script prend en argument une liste de Répertoire contenant des #
# evolutions de Cast3M (.dgibi, .procedur, .notice, .eso, .INC, GIBI,ERREUR) #
# et répercute ces évolutions dans les répertoires de toutes les plateformes #
#####
#- Quelques initialisations:
# - Choix de la version
# -
#
#2- Creation de l arborescence complete de processus de developpement Cast3M pour
# chaque plateforme:
# - Linux
# - MAC
# - Windows
#
#3- Test sur les arguments d'entrée
# S'il s'agit d'un repertoire non vide, on trie par type
l'ensemble de son contenu
# et on stocke dans des listes
#
#4- Pour chacun des fichiers presents dans le repertoire, on retire les espaces inutiles
# en fin de ligne
#
#5- Si dans le repertoire il existe des fichiers .dgibi:
# - on remplace les chemins /u2/castem/divers par ../divers
#           /u2/castem/divers par ../divers
# - on copie les .dgibi dans les repertoires suivants:
#   ${LISTE_DGI_1} ${REP_FINAL}/dgibi/
#   ${LISTE_DGI_1} ${REP_LINUX}/dgibi/
#   ${LISTE_DGI_1} ${REP_WINDOWS}/dgibi/
#   ${LISTE_DGI_1} ${REP_MAC}/dgibi/
#
#####
# Initialisations diverses
VERSION="19"

REP_DISQUE=/media/Support_Cast3M
REP_BASE=${REP_DISQUE}/Cast3M/Versions/20${VERSION}

REP_CORR=$PWD
REP_FINAL=${REP_BASE}/FINAL

REP_LINUX=${REP_BASE}/Plateformes/Linux/Dev
REP_MAC=${REP_BASE}/Plateformes/MAC/Dev
REP_WINDOWS=${REP_BASE}/Plateformes/Windows/Dev

source ${REP_LINUX}/bin/environnement_Cast3M_${VERSION}

CLEAN_SRC="FAUX"
CLEAN_NOT="FAUX"
CLEAN_PRO="FAUX"

# Test de l'arborescence Complète pour la Répercution des fichiers
if [ ! -d ${REP_FINAL} ] ; then
    mkdir ${REP_FINAL}
fi
if [ ! -d ${REP_FINAL}/data ] ; then
    mkdir ${REP_FINAL}/data
fi
if [ ! -d ${REP_FINAL}/dgibi ] ; then
    mkdir ${REP_FINAL}/dgibi
fi
if [ ! -d ${REP_FINAL}/divers ] ; then
    mkdir ${REP_FINAL}/divers
fi
if [ ! -d ${REP_FINAL}/doc ] ; then
    mkdir ${REP_FINAL}/doc
fi
if [ ! -d ${REP_FINAL}/header ] ; then
    mkdir ${REP_FINAL}/header
fi
if [ ! -d ${REP_FINAL}/include ] ; then
    mkdir ${REP_FINAL}/include
fi
if [ ! -d ${REP_FINAL}/include/c ] ; then
    mkdir ${REP_FINAL}/include/c
fi
if [ ! -d ${REP_FINAL}/include/eso ] ; then
    mkdir ${REP_FINAL}/include/eso
fi
if [ ! -d ${REP_FINAL}/notice ] ; then
    mkdir ${REP_FINAL}/notice
fi
if [ ! -d ${REP_FINAL}/procedur ] ; then
    mkdir ${REP_FINAL}/procedur
fi
if [ ! -d ${REP_FINAL}/sources ] ; then
    mkdir ${REP_FINAL}/sources
fi
if [ ! -d ${REP_FINAL}/verification_Cast3M ] ; then
    mkdir ${REP_FINAL}/verification_Cast3M
fi
if [ ! -d ${REP_BASE} ] ; then
    mkdir ${REP_BASE}
fi
if [ ! -d ${REP_BASE}/Plateformes ] ; then
    mkdir ${REP_BASE}/Plateformes
fi
if [ ! -d ${REP_BASE}/Plateformes/Linux ] ; then
    mkdir ${REP_BASE}/Plateformes/Linux
fi
if [ ! -d ${REP_BASE}/Plateformes/Windows ] ; then
    mkdir ${REP_BASE}/Plateformes/Windows
fi
if [ ! -d ${REP_BASE}/Plateformes/MAC ] ; then
    mkdir ${REP_BASE}/Plateformes/MAC
fi
```

```
# Linux
REP_OS=${REP_LINUX}
if [ ! -d ${REP_OS} ] ; then
    mkdir ${REP_OS}
fi
if [ ! -d ${REP_OS}/bin ] ; then
    mkdir ${REP_OS}/bin
fi
if [ ! -d ${REP_OS}/data ] ; then
    mkdir ${REP_OS}/data
fi
if [ ! -d ${REP_OS}/dgibi ] ; then
    mkdir ${REP_OS}/dgibi
fi
if [ ! -d ${REP_OS}/divers ] ; then
    mkdir ${REP_OS}/divers
fi
if [ ! -d ${REP_OS}/header ] ; then
    mkdir ${REP_OS}/header
fi
if [ ! -d ${REP_OS}/include ] ; then
    mkdir ${REP_OS}/include
fi
if [ ! -d ${REP_OS}/include/c ] ; then
    mkdir ${REP_OS}/include/c
fi
if [ ! -d ${REP_OS}/include/eso ] ; then
    mkdir ${REP_OS}/include/eso
fi
if [ ! -d ${REP_OS}/lib32 ] ; then
    mkdir ${REP_OS}/lib32
fi
if [ ! -d ${REP_OS}/lib64 ] ; then
    mkdir ${REP_OS}/lib64
fi
if [ ! -d ${REP_OS}/sources_make_version ] ; then
    mkdir ${REP_OS}/sources_make_version
fi
if [ ! -d ${REP_OS}/sources_make_version/bin_32 ] ; then
    mkdir ${REP_OS}/sources_make_version/bin_32
fi
if [ ! -d ${REP_OS}/sources_make_version/bin_64 ] ; then
    mkdir ${REP_OS}/sources_make_version/bin_64
fi
if [ ! -d ${REP_OS}/sources_make_version/bin_64 ] ; then
    mkdir ${REP_OS}/sources_make_version/bin_64
fi
if [ ! -d ${REP_OS}/sources_make_version/c ] ; then
    mkdir ${REP_OS}/sources_make_version/c
fi
if [ ! -d ${REP_OS}/sources_make_version/eso ] ; then
    mkdir ${REP_OS}/sources_make_version/eso
fi
if [ ! -d ${REP_OS}/sources_make_version/Corrections ] ; then
    mkdir ${REP_OS}/sources_make_version/Corrections
fi
if [ ! -d ${REP_OS}/sources_make_version/licence_EDURE ] ; then
    mkdir ${REP_OS}/sources_make_version/licence_EDURE
fi
if [ ! -d ${REP_OS}/sources_make_version/licence_INDUS ] ; then
    mkdir ${REP_OS}/sources_make_version/licence_INDUS
fi
if [ ! -d ${REP_OS}/verification_Cast3M ] ; then
    mkdir ${REP_OS}/verification_Cast3M
fi

# MAC
REP_OS=${REP_MAC}
if [ ! -d ${REP_OS} ] ; then
    mkdir ${REP_OS}
fi
if [ ! -d ${REP_OS}/bin ] ; then
    mkdir ${REP_OS}/bin
fi
if [ ! -d ${REP_OS}/data ] ; then
    mkdir ${REP_OS}/data
fi
if [ ! -d ${REP_OS}/dgibi ] ; then
    mkdir ${REP_OS}/dgibi
fi
if [ ! -d ${REP_OS}/divers ] ; then
    mkdir ${REP_OS}/divers
fi
if [ ! -d ${REP_OS}/header ] ; then
    mkdir ${REP_OS}/header
fi
if [ ! -d ${REP_OS}/include ] ; then
    mkdir ${REP_OS}/include
fi
if [ ! -d ${REP_OS}/include/c ] ; then
    mkdir ${REP_OS}/include/c
fi
if [ ! -d ${REP_OS}/include/eso ] ; then
    mkdir ${REP_OS}/include/eso
fi
if [ ! -d ${REP_OS}/lib32 ] ; then
    mkdir ${REP_OS}/lib32
fi
if [ ! -d ${REP_OS}/lib64 ] ; then
    mkdir ${REP_OS}/lib64
fi
if [ ! -d ${REP_OS}/sources_make_version ] ; then
    mkdir ${REP_OS}/sources_make_version
fi
if [ ! -d ${REP_OS}/sources_make_version/bin_32 ] ; then
    mkdir ${REP_OS}/sources_make_version/bin_32
fi
if [ ! -d ${REP_OS}/sources_make_version/bin_64 ] ; then
    mkdir ${REP_OS}/sources_make_version/bin_64
fi
if [ ! -d ${REP_OS}/sources_make_version/bin_64 ] ; then
    mkdir ${REP_OS}/sources_make_version/bin_64
fi
if [ ! -d ${REP_OS}/sources_make_version/c ] ; then
    mkdir ${REP_OS}/sources_make_version/c
fi
if [ ! -d ${REP_OS}/sources_make_version/eso ] ; then
    mkdir ${REP_OS}/sources_make_version/eso
fi
if [ ! -d ${REP_OS}/sources_make_version/Corrections ] ; then
    mkdir ${REP_OS}/sources_make_version/Corrections
fi
if [ ! -d ${REP_OS}/sources_make_version/licence_EDURE ] ; then
    mkdir ${REP_OS}/sources_make_version/licence_EDURE
fi
if [ ! -d ${REP_OS}/sources_make_version/licence_INDUS ] ; then
    mkdir ${REP_OS}/sources_make_version/licence_INDUS
fi
if [ ! -d ${REP_OS}/verification_Cast3M ] ; then
    mkdir ${REP_OS}/verification_Cast3M
fi

# Windows
REP_OS=${REP_WINDOWS}
if [ ! -d ${REP_OS} ] ; then
    mkdir ${REP_OS}
fi
if [ ! -d ${REP_OS}/bin ] ; then
    mkdir ${REP_OS}/bin
fi
if [ ! -d ${REP_OS}/data ] ; then
    mkdir ${REP_OS}/data
fi
if [ ! -d ${REP_OS}/dgibi ] ; then
    mkdir ${REP_OS}/dgibi
fi
if [ ! -d ${REP_OS}/divers ] ; then
    mkdir ${REP_OS}/divers
fi
if [ ! -d ${REP_OS}/header ] ; then
    mkdir ${REP_OS}/header
fi
if [ ! -d ${REP_OS}/include ] ; then
    mkdir ${REP_OS}/include
fi
```

```

if [ ! -d ${REP_OS}/include/c ] ; then
  mkdir ${REP_OS}/include/c ; fi
if [ ! -d ${REP_OS}/include/eso ] ; then
  mkdir ${REP_OS}/include/eso ; fi
if [ ! -d ${REP_OS}/lib32 ] ; then
  mkdir ${REP_OS}/lib32 ; fi
if [ ! -d ${REP_OS}/lib64 ] ; then
  mkdir ${REP_OS}/lib64 ; fi
if [ ! -d ${REP_OS}/sources_make_version ] ; then
  mkdir ${REP_OS}/sources_make_version ; fi
if [ ! -d ${REP_OS}/sources_make_version/bin_32 ] ; then
  mkdir ${REP_OS}/sources_make_version/bin_32 ; fi
if [ ! -d ${REP_OS}/sources_make_version/bin_64 ] ; then
  mkdir ${REP_OS}/sources_make_version/bin_64 ; fi
if [ ! -d ${REP_OS}/sources_make_version/c ] ; then
  mkdir ${REP_OS}/sources_make_version/c ; fi
if [ ! -d ${REP_OS}/sources_make_version/eso ] ; then
  mkdir ${REP_OS}/sources_make_version/eso ; fi
if [ ! -d ${REP_OS}/sources_make_version/Corrections ] ; then
  mkdir ${REP_OS}/sources_make_version/Corrections ; fi
if [ ! -d ${REP_OS}/sources_make_version/licence_EDURE ] ; then
  mkdir ${REP_OS}/sources_make_version/licence_EDURE ; fi
if [ ! -d ${REP_OS}/sources_make_version/licence_INDUS ] ; then
  mkdir ${REP_OS}/sources_make_version/licence_INDUS ; fi
if [ ! -d ${REP_OS}/verification_Cast3M ] ; then
  mkdir ${REP_OS}/verification_Cast3M ; fi
if [ `uname -m` == "x86_64" ] ; then
  BIT="64"
else
  BIT="32"
fi

# 1er Repercuter avec arguments ?
if [ "$(ls -A ${REP_LINUX}/sources_make_version/eso)" == "" ] && [ "$(ls -A ${REP_LINUX}/sources_make_version/bin_${BIT})" == "" ] ; then
  Repercuter1="VRAI"
  #On placera les .eso dans ${REP...}/sources_make_version/eso/*
  #On evolera (cast_evol) les repertoire eso/ et c/
else
  Repercuter1="FAUX"
  #On placera les .eso dans
${REP...}/sources_make_version/Corrections/*
  #On evolera (cast_evol) le repertoire Corrections/
fi

# Teste le nombre d'arguments ## d'entrée
COMPT_DGI_1=0
if [ "$#" == "0" ] ; then
  # Cas du nombre d'arguments nul
  echo
  echo " Donnez une liste de répertoire contenant des
évolutions de Cast3M "
  echo " à répercuter sur les plateformes suivantes "
  echo " -linux (32/64 -bits)"
  echo " -Windows (32/64 -bits)"
  echo " -MAC OSX ( 64 -bits)"
  echo

else
  # Si le nombre d'arguments n'est pas nul, alors on les
  parcourt #@
  for rep in $@ ; do
    # Initialisation des listes dans chacun des répertoires
    LISTE_DGI_1=""
    LISTE_PRO_1=""
    LISTE_NOT_1=""
    LISTE_ESO_1=""
    LISTE_INC_1=""
    LISTE_C_1=""
    LISTE_H_1=""
    LISTE_DIVERS_1=""

    # Boucle sur les fichiers de chacun des répertoires pour
    # remplir les listes de fichiers à évoluer
    for j in $(find ${REP_CORR}/${rep} -type f) ; do
      i=$(basename ${j})
      # Retire les espaces blancs en fin de ligne inutiles
      if [ `which sed` != "" ] ; then
        sed 's/[[:blank:]]*$/` ${j}`> ${j}_TMP
        mv ${j}_TMP ${j}
      fi

      #Cas particulier pour le jeux de donnees "acqulata.dgibi"
      if [ ${i} == acqulata.dgibi ] ; then
        LISTE_DIVERS_1="${LISTE_DIVERS_1} ${j}"
        sed s\?/u2/castem/dgibi/\?..\divers/\?g ${j} > ${j}.$$
        mv ${j}.$$ ${j}
      fi

      case ${i} in
        "basename ${i}.dgibi` .dgibi")
        LISTE_DGI_1="${LISTE_DGI_1} ${j}";
        ``basename ${i}.procedur` .procedur")
        LISTE_PRO_1="${LISTE_PRO_1} ${j}";
        ``basename ${i}.notice` .notice")
        LISTE_NOT_1="${LISTE_NOT_1} ${j}";
        ``basename ${i}.eso` .eso")
        LISTE_ESO_1="${LISTE_ESO_1} ${j}";
        ``basename ${i}.INC` .INC")
        LISTE_INC_1="${LISTE_INC_1} ${j}";
        ``basename ${i}.c` .c")
        LISTE_C_1="${LISTE_C_1} ${j}";
        ``basename ${i}.h` .h")
        LISTE_H_1="${LISTE_H_1} ${j}";
      esac
    done
  done
done

# Copie des listes dans les répertoires de chacune des
plateformes
cd ${rep}
echo "REPERTOIRE : ${rep}"
# Traitement des .dgibi
if [ ! "${LISTE_DGI_1}" == "" ] ; then
  ls -1 ${LISTE_DGI_1}
  # les chemins /u2/castem/divers sont remplacés par
../divers
  # /u/castem/divers sont remplacés par
../divers
  for i in ${LISTE_DGI_1} ; do
    sed s\?/u2/castem/divers\?..\divers\?g ${j} > ${j}.$$
    mv ${j}.$$ ${j}

    sed s\?/u/castem/divers\?..\divers\?g ${j} > ${j}.$$
    mv ${j}.$$ ${j}

    sed s\?/u2/castem/dgibi\?..\divers\?g ${j} > ${j}.$$
    mv ${j}.$$ ${j}
  done

# Repertoire Commun à toutes les Plateformes
cp -f ${LISTE_DGI_1} ${REP_FINAL}/dgibi/
# Linux
cp -f ${LISTE_DGI_1} ${REP_LINUX}/dgibi/
# Windows
cp -f ${LISTE_DGI_1} ${REP_WINDOWS}/dgibi/
# MAC
cp -f ${LISTE_DGI_1} ${REP_MAC}/dgibi/
fi

# Traitement des .procedur
if [ ! "${LISTE_PRO_1}" == "" ] ; then
  ls -1 ${LISTE_PRO_1}
  CLEAN_PRO="VRAI"
  # Repertoire Commun à toutes les Plateformes
  cp -f ${LISTE_PRO_1} ${REP_FINAL}/procedur/
fi

# Traitement des .notice
if [ ! "${LISTE_NOT_1}" == "" ] ; then
  ls -1 ${LISTE_NOT_1}
  CLEAN_NOT="VRAI"
  # Repertoire Commun à toutes les Plateformes
  cp -f ${LISTE_NOT_1} ${REP_FINAL}/notice/
fi

# Traitement des .eso
if [ ! "${LISTE_ESO_1}" == "" ] ; then
  ls -1 ${LISTE_ESO_1}
  CLEAN_SRC="VRAI"
  # Repertoire Commun à toutes les Plateformes
  cp -f ${LISTE_ESO_1} ${REP_FINAL}/sources/
  # Ou doit-on placer les .eso ?
  if [ "${Repercuter1}" == "VRAI" ] ; then
    eso_rep=eso/
  else
    eso_rep=Corrections/
  fi

  # Linux
  cp -f ${LISTE_ESO_1} ${REP_LINUX}/sources_make_version/${eso_rep}
  # Windows
  cp -f ${LISTE_ESO_1} ${REP_WINDOWS}/sources_make_version/${eso_rep}
  # MAC
  cp -f ${LISTE_ESO_1} ${REP_MAC}/sources_make_version/${eso_rep}
fi

# Traitement des .c
if [ ! "${LISTE_C_1}" == "" ] ; then
  ls -1 ${LISTE_C_1}
  CLEAN_SRC="VRAI"
  # Repertoire Commun à toutes les Plateformes
  # Les sources .c ne sont pas fournies dans les
  executables
  # Linux
  cp -f ${LISTE_C_1} ${REP_LINUX}/sources_make_version/c/
  # Windows
  cp -f ${LISTE_C_1} ${REP_WINDOWS}/sources_make_version/c/
  # MAC
  cp -f ${LISTE_C_1} ${REP_MAC}/sources_make_version/c/
fi

# Traitement des .INC
if [ ! "${LISTE_INC_1}" == "" ] ; then
  ls -1 ${LISTE_INC_1}
  CLEAN_SRC="VRAI"
  # Repertoire Commun à toutes les Plateformes
  cp -f ${LISTE_INC_1} ${REP_FINAL}/include/eso/
  # Linux
  cp -f ${LISTE_INC_1} ${REP_LINUX}/include/eso/
  # Windows

```

```

cp -f ${LISTE_INC_1} ${REP_WINDOWS}/include/eso/
# MAC
cp -f ${LISTE_INC_1} ${REP_MAC}/include/eso/
fi

# Traitement des .h
if [ ! "${LISTE_H_1}" == "" ]; then
  ls -1 ${LISTE_H_1}
  CLEAN_SRC="VRAI"
# Repertoire Commun à toutes les Plateformes
cp -f ${LISTE_H_1} ${REP_FINAL}/include/c/
# Linux
cp -f ${LISTE_H_1} ${REP_LINUX}/include/c/
# Windows
cp -f ${LISTE_H_1} ${REP_WINDOWS}/include/c/
# MAC
cp -f ${LISTE_H_1} ${REP_MAC}/include/c/
fi

# Traitement de GIBI.ERREUR
if [ -f GIBI.ERREUR ]; then
  ls -1 GIBI.ERREUR
# Repertoire Commun à toutes les Plateformes
cp -f GIBI.ERREUR ${REP_FINAL}/data/
# Linux
cp -f GIBI.ERREUR ${REP_LINUX}/data/
# Windows
cp -f GIBI.ERREUR ${REP_WINDOWS}/data/
# MAC
cp -f GIBI.ERREUR ${REP_MAC}/data/
fi

# Traitement de ce qu'il reste : les divers
if [ ! "${LISTE_DIVERS_1}" == "" ]; then
# Repertoire Commun à toutes les Plateformes
cp -f ${LISTE_DIVERS_1} ${REP_FINAL}/divers/
# Linux
cp -f ${LISTE_DIVERS_1} ${REP_LINUX}/divers/
# Windows
cp -f ${LISTE_DIVERS_1} ${REP_WINDOWS}/divers/
# MAC
cp -f ${LISTE_DIVERS_1} ${REP_MAC}/divers/
fi

cd ..
done

if [ "${CLEAN_PRO}" == "VRAI" ]; then
#Recreation CAST3M.PROC
cd ${REP_FINAL}/procedur
which cast_UTILS${VERSION}>/dev/null 2>&1
if [ "$?" == "0" ]; then
  cast_UTILS${VERSION}
  cp -f UTILPROC ${REP_LINUX}/data/CAST3M.PROC
  cp -f UTILPROC ${REP_WINDOWS}/data/CAST3M.PROC
  cp -f UTILPROC ${REP_MAC}/data/CAST3M.PROC
  mv UTILPROC ${REP_FINAL}/data/CAST3M.PROC
fi
fi

if [ "${CLEAN_NOT}" == "VRAI" ]; then
#Recreation CAST3M.MASTER
cd ${REP_FINAL}/notice

```

```

which cast_UTILS${VERSION}>/dev/null 2>&1
if [ "$?" == "0" ]; then
  cast_UTILS${VERSION}
  cp -f UTILNOTI ${REP_LINUX}/data/CAST3M.MASTER
  cp -f UTILNOTI ${REP_WINDOWS}/data/CAST3M.MASTER
  cp -f UTILNOTI ${REP_MAC}/data/CAST3M.MASTER
  mv UTILNOTI ${REP_FINAL}/data/CAST3M.MASTER
fi

if [ "${CLEAN_SRC}" == "VRAI" ]; then
# Linux
#Que doit-on evoluer ?
if [ "${Repercuter1}" == "VRAI" ]; then
  cast_evol_rep="${REP_LINUX}/sources_make_version/eso/
${REP_LINUX}/sources_make_version/c/"
else
  cast_evol_rep="${REP_LINUX}/sources_make_version/Corrections/"
fi
rm -f ${REP_LINUX}/bin/bin_Cast3M *
rm -f ${REP_LINUX}/lib32/libcastem_DEVEL_32.a
rm -f ${REP_LINUX}/lib32/libcastem_INDUS_32.a
rm -f ${REP_LINUX}/lib64/libcastem_DEVEL_64.a
rm -f ${REP_LINUX}/lib64/libcastem_INDUS_64.a
if [ "${PLATEFORME}" == "Linux" ]; then
  which cast_evol${VERSION}>/dev/null 2>&1
  if [ "$?" == "0" ]; then
    cast_evol${VERSION} ${cast_evol_rep}
  fi
fi

# Windows
rm -f ${REP_WINDOWS}/bin/bin_Cast3M *
rm -f ${REP_WINDOWS}/lib32/libcastem_DEVEL_32.a
rm -f ${REP_WINDOWS}/lib32/libcastem_INDUS_32.a
rm -f ${REP_WINDOWS}/lib64/libcastem_DEVEL_64.a
rm -f ${REP_WINDOWS}/lib64/libcastem_INDUS_64.a
# MAC
#Que doit-on evoluer ?
if [ "${Repercuter1}" == "VRAI" ]; then
  cast_evol_rep="${REP_MAC}/sources_make_version/eso/
${REP_MAC}/sources_make_version/c/"
else
  cast_evol_rep="${REP_MAC}/sources_make_version/Corrections/"
fi
rm -f ${REP_MAC}/bin/bin_Cast3M *
rm -f ${REP_MAC}/lib32/libcastem_DEVEL_32.a
rm -f ${REP_MAC}/lib32/libcastem_INDUS_32.a
rm -f ${REP_MAC}/lib64/libcastem_DEVEL_64.a
rm -f ${REP_MAC}/lib64/libcastem_INDUS_64.a
if [ "${PLATEFORME}" == "MAC" ]; then
  which cast_evol${VERSION}>/dev/null 2>&1
  if [ "$?" == "0" ]; then
    cast_evol${VERSION} ${cast_evol_rep}
  fi
fi
fi

```

Annexe B. Script environnement_Cast3M19.bat

```
@ECHO OFF
SET CASTEM_VERSION=%XXANNEEXX%
SET CASTEM_REVISION=%XXREVISIONXX%
SET SCRIPTREP=%~dp0

REM Espaces remplaçant l'année dans les ECHO
SET SPY=""
IF "%CASTEM_VERSION%"=="" (
    SET SPY=" "
)
REM : Retrait des doubles cotes
SET SPY=%SPY:~1,-1%

SET LIGNEUP=x-----
SET LIGNEDOWN=-----c
SET LIGNEMIDDLE='-----□

REM retrait de "\bin\" a la fin du chemin
SET CASTEMREP=%SCRIPTREP:~-0,-5%
SET ESOPEREP=%CASTEMREP%\..\Esope

IF "%BIT%"=="64" (
    GOTO SUITE_1
) ELSE IF "%BIT%"=="32" (
    GOTO SUITE_1)

IF DEFINED ProgramFiles(x86) (
    SET /A BIT=64
) ELSE (
    SET /A BIT=32)

REM Balise de suite pour tenir compte de la définition de BIT
:SUITE_1
SET CASTEM_PLATEFORME=WINDOWS-%BIT%bits
IF NOT EXIST "%CASTEMREP%\MinGW" GOTO SUITE_2
IF %BIT% EQU 64 (
    SET MINGWPATH=%CASTEMREP%\MinGW\x86_64-8.1.0-release-posix-seh-rt_v6-rev0\mingw64\bin"
    SET TFEHOME=%CASTEMREP%\MFRONT\mfront-x86_64"
) ELSE (
    SET MINGWPATH=%CASTEMREP%\MinGW\i686-6.4.0-release-posix-dwarf-rt_v5-rev0\mingw32\bin"
    SET TFEHOME=%CASTEMREP%\MFRONT\mfront-i686")
SET CMAKEPATH=%CASTEMREP%\..\CMAKE\cmake-3.13.0-rc1-win64-x64\bin"

REM retrait des doubles cotes de part et d'autre des variables suivantes
SET MINGWPATH=%MINGWPATH:~1,-1%
SET CMAKEPATH=%CMAKEPATH:~1,-1%

REM Ajout au PATH du chemin vers les compilateurs
SET PATH=%MINGWPATH%;%CASTEMREP%\lib%BIT%;%CMAKEPATH%;%TFEHOME%\bin;%PATH%

REM Pour MFRONT (TFEHOME est obligatoire !!!)
SET TFEHOME=%TFEHOME:~1,-1%
SET AR=%MINGWPATH%\ar.exe"
SET AS=%MINGWPATH%\as.exe"
SET CC=%MINGWPATH%\gcc.exe"
SET CXX=%MINGWPATH%\g++.exe"
SET FC=%MINGWPATH%\gfortran.exe"
SET MAKE=%MINGWPATH%\mingw32-make.exe"

REM définition du titre de la fenêtre de commande
TITLE Environnement de Développement MinGW - %BIT%bits
EXIT /B 0

:SUITE_2
IF NOT EXIST "%CASTEMREP%\licence" GOTO SUITE_3
TITLE Environnement Industriel - %BIT%bits
EXIT /B 0

:SUITE_3
TITLE Environnement Utilisateur - %BIT%bits
EXIT /B 0
```

Annexe C. Scripts compilcast19.bat et compilcast19

a. Script compilcast19.bat

```

@ECHO OFF
SETLOCAL ENABLEDELAYEDEXPANSION

SET SCRIPT_REP=%~dp0

REM Chargement de l'environnement (Laisser %CASTEM_VERSION%, il est
remplace automatiquement par InstallJammer)
CALL %SCRIPT_REP:~0,-1%\environnement_Cast3M%XXANNEEXX%

REM Options de compilation FORTRAN
SET F_INIT=-finit-real=snan -finit-integer=-2147483648 -finit-
logical=false -finit-character=32
REM SET F_INIT=-finit-real=snan -finit-integer=2147483647 -finit-
logical=false -finit-character=32
REM SET F_INIT=-finit-real=snan -finit-integer=-9223372036854775808 -
finit-logical=false -finit-character=32
REM SET F_INIT=-finit-real=snan -finit-integer=9223372036854775807 -
finit-logical=false -finit-character=32
SET F_COMMON_OPTIONS=-ftracer -fgcse-sm -fgcse-las -fopt-info -
freciprocal-math -ffree-loop-distribution -fassociative-math -frename-
registers -fno-trapping-math -fno-signaling-nans
SET F_MANDATORY_OPTIONS=-ffpe-summary=invalid,zero,overflow -fopenmp -
frecursive -pthread -mthreads -mfpmath=sse -mtune=generic -fno-signed-
zeros -fno-sign-zero -fno-aggressive-loop-optimizations
SET COMP_OPTIM=-O5

REM Detection si WIN32 ou WIN64-bits ou si BIT est defini
IF %BIT% EQU 64 (
    SET F_DEFAULT=-fdefault-integer-8 -fdefault-real-8 -fdefault-double-8
    SET F_ARCH_OPTIONS=
) ELSE (
    SET F_DEFAULT=
    SET F_ARCH_OPTIONS=-march=pentium4 -msse -fno-tree-dse -fno-tree-fre -
fno-tree-pre -fno-tree-dominator-opts -fno-tree-loop-vectorize -fno-
predictive-commoning -fno-caller-saves)

SET F_OPTIONS=%F_MANDATORY_OPTIONS% %F_DEFAULT% %F_INIT%
%F_COMMON_OPTIONS% %F_ARCH_OPTIONS%

REM Definition de l'executable Esope et des parametres de traduction
SET esopX_Local=%esopX_%BIT%_%CASTEM_VERSION%.exe
IF EXIST %esopX_Local% (
    SET ESOPEXEC=%esopX_Local%
) ELSE (
    SET
ESOPEXEC="%CASTEM_REP%\bin\bin_esope_Win_%BIT%_%CASTEM_VERSION%.exe")
REM retrait des doubles cotes
SET ESOPEXEC=!ESOPEXEC:~1,-1!

SET ESOPE_INC=%CASTEM_REP%\include\eso
SET ESOPE_PARAM=NORME=TRADUCTEUR,FORT=WIN%BIT%,ESOPE=100Mo

REM Initialisations :
SET FORTRAN_FILE=
SET LOG_FILE=
SET LISTE_SRC=
SET PAUSEFIN=
SET AIDE=
SET HELP=
SET PARALLELE_COMP=FAUX
SET SUFFIX1=
SET DELETE_BALISE=FAUX
SET FILE_LIST=
SET NOM_BALISE=
SET EXIT_CMD=/B
SET /A ERROLEV=0
SET /A NBCPU=1

REM Lecture des Arguments d'entree
SET OPTIONS_SCRIPT=
:DEBUT_LECTURE
SET ArgNAME1=%1
REM affichage de l'aide en Francais
IF "%ArgNAME1%"=="--aide" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)
IF "%ArgNAME1%"=="/?" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)
REM affichage de l'aide en Anglais
IF "%ArgNAME1%"=="--help" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)
IF "%ArgNAME1%"=="-h" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)
REM compilation sans optimisation du code
IF "%ArgNAME1:~0,2%"=="-O" (
    SET COMP_OPTIM=%ArgNAME1%
    SET F_OPTIONS=%F_MANDATORY_OPTIONS% %F_DEFAULT% %F_INIT%
    %F_ARCH_OPTIONS%
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    SHIFT
    GOTO DEBUT_LECTURE
)

REM compilation avec l'option CONTROLE d'ESOPE
IF "%ArgNAME1%"=="-c" (
    SET ESOPE_PARAM=!ESOPE_PARAM!,CONTROL
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    SET SUFFIX1=_C
    SHIFT
    GOTO DEBUT_LECTURE
)

REM activation du mode DEBUG
IF "%ArgNAME1%"=="-d" (
    SET COMP_OPTIM=-g -O0
    SET F_OPTIONS=-frange-check -fbacktrace -ftrapping-math -fsignaling-
nans -ffpe-trap=invalid,zero,overflow -ftrapv -fcheck=array-
temps,do,pointer %F_MANDATORY_OPTIONS% %F_DEFAULT% %F_INIT%
%F_ARCH_OPTIONS%
    SET FORTRANFILE=VRAI
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    SET SUFFIX1=_d
    SHIFT
    GOTO DEBUT_LECTURE
)

REM CONTROLE et DEBUG
IF "%ArgNAME1%"=="-cd" (
    SET COMP_OPTIM=-g -O0
    SET F_OPTIONS=-frange-check -fbacktrace -ftrapping-math -fsignaling-
nans -ffpe-trap=invalid,zero,overflow -ftrapv -fcheck=array-
temps,do,pointer %F_MANDATORY_OPTIONS% %F_INIT% %F_DEFAULT%
%F_ARCH_OPTIONS%
    SET FORTRANFILE=VRAI
    SET ESOPE_PARAM=!ESOPE_PARAM!,CONTROL
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    SET SUFFIX1=_cd
    SHIFT
    GOTO DEBUT_LECTURE
)

REM compilation en conservant le fichier FORTRAN77
IF "%ArgNAME1%"=="-f" (
    SET FORTRANFILE=VRAI
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    SHIFT
    GOTO DEBUT_LECTURE
)

REM compilation en conservant le fichier .log
IF "%ArgNAME1%"=="-log" (
    SET LOGFILE=VRAI
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    SHIFT
    GOTO DEBUT_LECTURE
)

REM recuperation de l'assembleur
IF "%ArgNAME1%"=="-S" (
    SET FORTRANFILE=VRAI
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    SET F_OPTIONS!=!F_OPTIONS! -S -g -fverbose-asn
    SHIFT
    GOTO DEBUT_LECTURE
)

REM compilation du traducteur ESOPE
IF "%ArgNAME1%"=="-ESOPE" (
    SET ESOPE_INC=%ESOPEXEC%\include
    SET OPTIONS_SCRIPT=!OPTIONS_SCRIPT! %ArgNAME1%
    IF EXIST %esopX_Local% (
        SET ESOPEXEC=%esopX_Local%
    ) ELSE (
        SET
ESOPEXEC="%CASTEM_REP%\bin\bin_esope_Win_%BIT%_%CASTEM_VERSION%.exe")
    REM retrait des doubles cotes
    SET ESOPEXEC=!ESOPEXEC:~1,-1!

    SHIFT
    GOTO DEBUT_LECTURE
)

REM compilation en parallele
IF "%ArgNAME1:~0,2%"=="-P" (
    IF "%ArgNAME1:~2,2%"==""
        ECHO.
        ECHO %LIGNEUP%
        ECHO Il manque le nombre de processus apres -P. Exemple : -P4
        ECHO %LINEDOWN%
)

```

```

        EXIT /B 1
) ELSE (
    SET /A NBCPU=%ArgNAME1:~2,2%
SHIFT
GOTO DEBUT_LECTURE
)

REM quitter le cmd.exe a la fin de la compilation
IF "%ArgNAME1%"=="-EXIT" (
    SET EXIT_CMD=
SHIFT
GOTO DEBUT_LECTURE
)

REM Nom complet donne en argument : Construction de la liste des
arguments
SET NomF2=%~1
IF NOT "!NomF2!"=="" (
    IF EXIST !NomF2! (
        IF "%NomF2:~0,15%" EQU "compilcast_PARA" (
            SET DELETE_BALISE=VRAI
            SET FILE_LIST=f
            SET NOM_BALISE=compilcast_PARA_BALISE%NomF2:~15,16%
            SET LISTE_SRC=!NomF2!
        ) ELSE (
            SET LISTE_SRC=!LISTE_SRC! !NomF2!
        )
    )
SHIFT
GOTO DEBUT_LECTURE
)

:LABEL AIDE
IF DEFINED AIDE (
REM Affiche l'aide en Francais
ECHO .
ECHO %LIGNEUP%
ECHO NOM
ECHO compilcast%CASTEM_VERSION%%SPY% : Compilation des sources
Esope et C
ECHO Site web : http://www-cast3m.cea.fr/
ECHO %LIGNEMIDDLE%
ECHO VERSION
ECHO Version du Script : %CASTEM_VERSION%%SPY%.%CASTEM_REVISION%
ECHO %LIGNEMIDDLE%
ECHO SYNTAXE
ECHO compilcast%CASTEM_VERSION%%SPY% [OPTION]...
[LISTE_FICHIERS]...
ECHO
ECHO %LIGNEMIDDLE%
ECHO DESCRIPTION
ECHO --aide : Affiche le manuel de cette commande en Francais
ECHO --help : Affiche le manuel de cette commande en Anglais
ECHO -c : Option CONTROLE de Esope [Taille des segments]
ECHO -d : Compilation en mode debug : -g -O0 activee
ECHO -cd : Option CONTROLE et mode debug
ECHO -On : Optimisations activees n=(s,g,fast,0,1,2,3)
ECHO -f : conserve le fichier fortran77 traduit depuis
l'Esope
ECHO -log : conserve la sortie du compilateur dans un fichier
.log
ECHO -Pn : Compilation des sources en parallele sur n
processus
ECHO -S : Code Assembleur de la source en sortie
ECHO -EXIT : cmd.exe sera ferme apres la compilation
ECHO -ESOPE : Compilation du TRADUCTEUR ESOPE [SEMT/LM2S]
seulement]
ECHO
ECHO %LIGNEMIDDLE%
ECHO EXEMPLES
ECHO compilcast%CASTEM_VERSION%%SPY% fichier1.eso fichier2.eso
*.c
ECHO Compile les sources normalement
ECHO
ECHO compilcast%CASTEM_VERSION%%SPY% -d f*.eso fichier2.c
ECHO Compile les sources en mode debug
ECHO
ECHO compilcast%CASTEM_VERSION%%SPY% -d -c *.eso *.c
ECHO Compile les sources en mode debug et avec le CONTROLE
d'ESOPE
ECHO
ECHO %LIGNEMIDDLE%
ECHO AUTEUR
ECHO Script ecrit par Clement BERTHINIER
ECHO %LIGNEMIDDLE%
ECHO VOIR AUSSI
ECHO Aide du Script 'castem%CASTEM_VERSION%'%SPY% :
'castem%CASTEM_VERSION%' --aide'%SPY%
ECHO Aide du Script 'essaicast%CASTEM_VERSION%'%SPY% :
'essaicast%CASTEM_VERSION%' --aide'%SPY%
ECHO
ECHO %LIGNEDOWN%
EXIT /B 0
)

:LABEL HELP
IF DEFINED HELP (
REM Affiche l'aide en Francais
ECHO .
ECHO %LIGNEUP%
ECHO NAME
ECHO compilcast%CASTEM_VERSION%%SPY% : Compilation of Esope and C
sources
ECHO Site web : http://www-cast3m.cea.fr/
ECHO %LIGNEMIDDLE%
ECHO VERSION
ECHO Script version : %CASTEM_VERSION%%SPY%.%CASTEM_REVISION%
ECHO %LIGNEMIDDLE%
ECHO DESCRIPTION
ECHO --aide : Print the manual of this script in French
ECHO --help : Print the manual of this script in English
ECHO -c : CONTROLE Option for Esope [Size of Segments]
ECHO -d : Debug mode compilation: -g -O0 activated
ECHO -cd : CONTROLE Option and debug mode
ECHO -On : Optimization activated n=(s,g,fast,0,1,2,3)
ECHO -f : Keep the fortran77 source file produced by the
conversion
ECHO -log : Keep the output of the compiler in a .log file
ECHO -Pn : Parallel compilation of sources with n process
ECHO -S : Assembler Code of the source file as output
ECHO -EXIT : cmd.exe will be closed after the compilation
ECHO -ESOPE : Compilation of ESOPE TRANSLATOR [SEMT/LM2S only]
ECHO %LIGNEMIDDLE%
ECHO EXAMPLES
ECHO compilcast%CASTEM_VERSION%%SPY% file1.eso file2.eso *.c
ECHO Compile sources normally
ECHO compilcast%CASTEM_VERSION%%SPY% -d f*.eso file2.c
ECHO Compile sources with debug mode
ECHO compilcast%CASTEM_VERSION%%SPY% -d -c *.eso *.c
ECHO Compile sources with debug mode and CONTROL for Esope
ECHO %LIGNEMIDDLE%
ECHO AUTEUR
ECHO Script written by Clement BERTHINIER
ECHO %LIGNEMIDDLE%
ECHO SEE ALSO
ECHO Manual for 'castem%CASTEM_VERSION%'%SPY% :
'castem%CASTEM_VERSION%' --help'%SPY%
ECHO Manual for 'essaicast%CASTEM_VERSION%'%SPY% :
'essaicast%CASTEM_VERSION%' --help'%SPY%
ECHO %LIGNEDOWN%
EXIT /B 0
)

REM Fin de la lecture des arguments
:FIN_LECTURE

SET SUFFIX1=
IF NOT "%INDICE_CONTROLE%%INDICE_DEBUG%"=="" (
    SET SUFFIX1=%INDICE_CONTROLE%-%INDICE_DEBUG%
)
REM Un peu de menage avant de commencer (Lance en parallele on ne
supprime pas)
IF DEFINED EXIT_CMD (
    IF EXIST "ZZZ_Erreur_Compilation%SUFFIX1%.log"      DEL
"ZZZ_Erreur_Compilation%SUFFIX1%.log">nul
    ECHO.)
REM Calcul du nombre de sources
SET /A srccompt=0
SET LISTE_SRC_PARA=!OPTIONS_SCRIPT!

REM Recopie du contenu de !FILE_LIST! dans OPT1 parce que le "FOR" ne
supporte pas les "!" pour la premiere option
SET OPT1!=!FILE_LIST!
FOR %OPT1% %i IN (!LISTE_SRC!) DO (
    SET /A srccompt+=1
)
SET /A src_restantes=%srccompt%
REM Preparation pour la traduction / compilation en parallele
REM En raison d'une limitation de la taille des variables
d'environnement en BATCH, on ne peut pas depasser 750 sources par
process en parallele
SET /A MAX_SRC_LIST=25
SET /A JOB_LOC=0
SET /A srchum=0
SET /A nbsrc=%srccompt% / %NBCPU%
IF %nbsrc% GTR %MAX_SRC_LIST% SET /A nbsrc=%MAX_SRC_LIST%
SET /A nbsrc_WAIT=%nbsrc% * %NBCPU%
SET WAITING=
REM verification du travail en parallele
IF %NBCPU% GTR 1 (
    SET PARALLELE_COMP=VRAI
) ELSE IF %NBCPU% EQU 1 (
    SET PARALLELE_COMP=FAUX
) ELSE (
    SET PARALLELE_COMP=FAUX
ECHO.
ECHO %LIGNEUP%
ECHO Le nombre de processus apres -P est invalide
ECHO %LIGNEDOWN%
EXIT /B 1
)
IF "%PARALLELE_COMP%"=="VRAI" (
    REM Decoupage en listes
    IF EXIST "compilcast_PARA*.txt"      DEL "compilcast_PARA*.txt"
)

```

```

SET /A proccour=1
SET /A procmaxi=1
FOR %OPT1% %%i IN (!LISTE_SRC!) DO (
  ECHO %%i>compilcast_PARA!proccour!.txt
  IF !proccour! GTR !procmaxi!
    SET /A procmaxi=!proccour!
)
IF !proccour! EQU %NBCPU% (
  SET /A proccour=
) ELSE (
  SET /A proccour+=1
)

REM Compilation des listes en parallele
SET /A NBCPU=%procmaxi!
FOR /L %%i IN (1, 1, !procmaxi!) DO (
  ECHO %%i>compilcast_PARA_BALISE%%i.txt
  START "COMPILE" /MIN CALL compilcast%CASTEM_VERSION% -EXIT
!OPTIONS_SCRIPT! compilcast_PARA%%i.txt
)
REM Attente de la fin de la compilation
:DEBUT_ATTENTE
SET /A nbwait=0
FOR /L %%i IN (1, 1, !procmaxi!) DO (
  IF EXIST "compilcast_PARA_BALISE%%i.txt" SET /A nbwait+=1
)
IF !nbwait! GTR 0 (
  TIMEOUT 2 /NOBREAK >nul
  GOTO DEBUT_ATTENTE
)
REM suppression des listes
DEL "compilcast_PARA*.txt"
GOTO BILAN_COMPILATION
)

REM Compilation de la liste des sources LISTE_SRC
SET /A ERROLOC=1
FOR %OPT1% %n in (!LISTE_SRC!) DO (
  SET NomF2=%n
  IF EXIST !NomF2! (
    SET /A srccnum+=1
    ECHO !NomF2!
    TITLE Compilation - %BIT% bits : !NomF2!
    ECHO Sources traitees : !srccnum!/!srccompt%
    CALL :COMPIL_UN_SOURCE !NomF2!
    IF ERRORLEVEL !ERROLOC! SET /A ERROLOC!=!ERRORLEVEL!+1
    SET /A ERROLEV!=!ERROLOC!-1
  ) ELSE (
    ECHO Fichier inexistant / Non existing file : !NomF2!
    ECHO.
  )
  IF "%DELETE_BALISE%"=="VRAI" DEL "%NOM_BALISE%"

:BILAN_COMPILATION
REM Bilan des erreurs de compilation s'il y en a eu
IF EXIST ZZZ_Erreur_Compilation%SUFFIX1%.log (
  IF %ERROLEV% EQU 0 SET /A ERROLEV=20
  ECHO %LIGNEUP%           Liste des echecs de compilation
  ECHO %LIGNEMIDDLE%
  TYPE ZZZ_Erreur_Compilation%SUFFIX1%.log
  ECHO %LINEDOWN%
) ELSE (
  ECHO %LIGNEUP%           Compilation des sources reussie
  ECHO %LINEDOWN%
)
EXIT %EXIT_CMD% %ERROLEV%

:COMPIL_UN_SOURCE
SET NomF=%~n1
SET NomF1=%~1
SET NOMF3="%NOMF2%"

REM Nettoyage des eventuels fichiers de compilation
IF EXIST "%NOMF%".o DEL "%NOMF%".o
IF EXIST "%NOMF%".log DEL "%NOMF%".log
SET /A ERROLEV=0

IF "%NomF1%"=="%NomF%.c" (
  REM Compilation du fichier .
  CALL %CC% -v -Wall -m%BIT% %COMP_OPTIM% -DWIN%BIT% -D REENTRANT -
  DFOR_WINDOWS -I"%CASTEM_REP%"\\include\\c -c "%NOMF%.c" > "%NOMF%.log
  2>&1
  REM -DFOR_WINDOWS : Pour la compilation de perm.c qui gèle la version
  industrielle de Cast3M

) ELSE IF "%NomF1%"=="%NomF%.eso" (
  REM Nettoyage des eventuels fichiers existants
  IF EXIST "%NOMF%".f DEL "%NOMF%".f
  IF EXIST "%NOMF%".s DEL "%NOMF%".s
  IF EXIST "%NOMF%".lst DEL "%NOMF%".lst
  IF EXIST "%NOMF%".err DEL "%NOMF%".err
  IF EXIST "%NOMF%".log DEL "%NOMF%".log
  SET ESOPC_OUT=%NOMF%.f
  SET ESOPC_LST=%NOMF%.lst
  SET ESOPC_ERR=%NOMF%.err
  CALL "%ESOPEEXEC%" < "%NOMF%".eso

  REM Verification du code de sortie du traducteur : si c'est superieur
  ou egale a 4 alors il y a une erreur de traduction
  IF ERRORLEVEL 16 (
    REM Niveau d'erreur SUPERIEUR ou EGAL a 16
    SET /A ERROLEV=16
    ECHO %NOMF3:~1,18% NON TRADUIT
    >> ZZZ_Erreur_Compilation%SUFFIX1%.log

) ELSE IF ERRORLEVEL 4 (
  REM Niveau d'erreur SUPERIEUR ou EGAL a 4
  SET /A ERROLEV=4
  ECHO %NOMF3:~1,18% CONTIENT UNE OU PLUSIEURS SUBROUTINES NON
  TRADUITES >> ZZZ_Erreur_Compilation%SUFFIX1%.log

) ELSE (
  REM Compilation du fortran
  CALL %FC% -v -Wall -m%BIT% %COMP_OPTIM% %F_OPTIONS% -c "%NOMF%.f
  -I"%CASTEM_REP%"\\include\\eso> "%NOMF%.log 2>&1
  IF EXIST "%NOMF%".s CALL %AS% "%NOMF%".s -o "%NOMF%".o

) ELSE IF "%NomF1%"=="%NomF%.f" (
  REM Compilation du fichier .f
  CALL %FC% -v -Wall -m%BIT% %COMP_OPTIM% %F_OPTIONS% -c "%NOMF%.f -
  I"%CASTEM_REP%"\\include\\eso> "%NOMF%.log 2>&1

) ELSE IF "%NomF1%"=="%NomF%.F" (
  REM Compilation du fichier .F
  CALL %FC% -v -Wall -m%BIT% %COMP_OPTIM% %F_OPTIONS% -c "%NOMF%.f -
  I.> "%NOMF%.log 2>&1

REM Cas des fichiers qui ne sont pas .eso ou .
) ELSE (
  ECHO Source non reconnue / Unknown source file : %NOMF2%
)

REM Verification si le binaire .o a ete genere
IF NOT EXIST "%NOMF%".o (
  ECHO %NOMF3:~1,18% NON COMPILE
  >> ZZZ_Erreur_Compilation%SUFFIX1%.log
  SET /A ERROLEV=20
) ELSE (
  REM Suppression des fichiers inutiles
  IF EXIST "%NOMF%.log (
    IF NOT DEFINED LOG_FILE DEL "%NOMF%.log
    IF NOT DEFINED FORTRAN_FILE DEL "%NOMF%.f
    IF EXIST "%NOMF%.lst DEL "%NOMF%.lst
    IF EXIST "%NOMF%.err DEL "%NOMF%.err
  )
)

:SORTIE_SCRIPT
EXIT /B !ERROLEV!

```

b. Script compilcast19

```
#!/bin/bash

#Recuperation des variables d'environnement
if [ "${VERSION}" == "" ] ; then VERSION=%(XXANNEEXX)
fi
if [ "${REVISION}" == "" ] ; then REVISION=%(XXREVISIONXX)
fi
if [ "${PLATEFORME}" == "" ] ; then PLATEFORME=%(XXPLATEFORMEXX)
fi
if [ "${CASTEM_REP}" == "" ] ; then CASTEM_REP=%{INSTALL_PATH}
fi

# Detection de l'architecture ou forcee si BIT est defini dans
l'environnement
if [ `uname -m` == "x86_64" ] || [ "$BIT" == "64" ] && [ !
"$BIT" == "32" ] ; then
    BIT="64"
    GNU_VER="8.2.0"
    GNU_PATH=${CASTEM_REP}/GCC/GCC-x86_64
else
    BIT="32"
    GNU_VER="6.3.0"
    GNU_PATH=${CASTEM_REP}/GCC/GCC-i686
fi

# Repertoires lies a MPI (includes)
# Variable d'environnement definissant la commande MPI :
MPI_RUNCMD
if [ "${MPI_LIBREP}" == "" ] ; then
    MPI_LIBREP=${CASTEM_REP}/mpi/mpis${BIT}/lib ; fi
if [ "${MPI_EXECREP}" == "" ] ; then
    MPI_EXECREP=${CASTEM_REP}/mpi/mpis${BIT}/bin ; fi
if [ "${MPI_INCREP}" == "" ] ; then
    MPI_INCREP=${CASTEM_REP}/mpi/mpis${BIT}/include ; fi

# Repertoire pour les librairies necessaires (pour l'execution
ou l'édition des liens)
if [ "${PLATEFORME}" == "Linux" ] ; then
    export
    LD_LIBRARY_PATH="${CASTEM_REP}/lib${BIT}:$(GNU_PATH)/lib:${GNU_P
ATH}/lib64:${MPI_LIBREP}:$LD_LIBRARY_PATH"
    # Parfois "ld" ne trouve pas crt1.o il faut ajouter des
    repertoires a LIBRARY_PATH
    export
    LIBRARY_PATH="${LIBRARY_PATH}:/usr/lib64:/usr/lib:/usr/lib/x86_6
4-linux-gnu:/usr/lib/i386-linux-gnu"

elif [ "${PLATEFORME}" == "MAC" ] ; then
    export
    DYLD_LIBRARY_PATH="${CASTEM_REP}/lib${BIT}:$(GNU_PATH)/lib:${GNU
_PATH}/lib64:${MPI_LIBREP}:$DYLD_LIBRARY_PATH"
fi

# Ajout du chemin vers mpiprun au PATH
if [ -x ${MPI_EXECREP}/mpiprun ] ; then
    export PATH=${MPI_EXECREP}:$PATH
fi

# Detection de la variable d'environnement GFORTRAN
if [ "${GFORTRAN}" == "" ] ; then
    GFORTRAN=${GNU_PATH}/bin/gfortran-${GNU_VER}
fi

if [ "${GFORTRAN}" == "" ] ; then
    GFORBIN="FAUX"
else
    GFORBIN="VRAI"
fi

# Detection de la variable d'environnement GCC
if [ "${GCC}" == "" ] ; then
    GCC=${GNU_PATH}/bin/gcc-${GNU_VER}
fi

if [ "${GCC}" == "" ] ; then
    GCCBIN="FAUX"
else
    GCCBIN="VRAI"
fi

CASTEM_INC=-I${CASTEM_REP}/include/c -I/usr/X11R6/include -
I/usr/local/include -I/opt/X11/include
CASTEM_DEFINE=-DLINUX
if [ "$BIT" == "64" ] ; then
    CASTEM_DEFINE=$(CASTEM_DEFINE)${BIT}
    if [ ${PLATEFORME} == "MAC" ] ; then
        CASTEM_DEFINE=$(CASTEM_DEFINE) -DOSX"
    fi
fi

# Traducteur ESOPE (possibilite de le surcharger par la variable
d'environnement "ESOPE_EXEC")
if [ "${ESOPE_EXEC}" == "" ] ; then
    if [ -x esop ${BIT} ${VERSION} ] ; then
        ESOPE_EXEC=./esop_${BIT}_${VERSION}"
    else
        ESOPE_EXEC="${CASTEM_REP}/bin/bin_esope_${PLATEFORME}_${BIT}_${V
ERSION}"
    fi
fi
```

```
# Definition de variables
AIDE="FAUX"          # Si VRAI permet d'afficher le manuel -Fr-
du script
HELP="FAUX"           # Si VRAI permet d'afficher le manuel -En-
du script
CONTROLE_MODE="FAUX" # Si VRAI utilise l'option CONTROLE de
Esope
COMP_OPTIM="-O5"      # Par defaut -O5 pour l'optimisation lors
de la compilation
FORTRAN_FILE="FAUX"   # Si VRAI le fichier fortran77 issu de la
traduction est conserve
LOG_FILE="FAUX"        # Si VRAI le fichier log issu de la
compilation est conserve
NIV_ERR="FAUX"         # Si VRAI permet d'afficher une sortie avec
erreur du script
BAD_ARG="FAUX"         # Si VRAI permet d'afficher une sortie avec
erreur d'argument
ESOPE_TRADE="FAUX"     # En cas de fabrication de la librairie
ESOPE
PARALLELLE_COMP="FAUX"
NBCPU="1"
ETOILE="*****"
*****"

# Suppression des fichiers issus d'une precedente compilation
if [ -f ZZZ_Non_Compile.log ] ; then \rm -f ZZZ_Non_Compile.log ;
fi
if [ -f ZZZ_Non_traduit.log ] ; then \rm -f ZZZ_Non_traduit.log ;
fi

# Teste le nombre d'arguments d'entree
if [ $# == "0" ] ; then
    # Cas du nombre d'arguments nul
    BAD_ARG="VRAI"

else
    # Fabrication des listes d'options et d'arguments
    LISTE_ESO_1=""
    LISTE_F_1=""
    LISTE_C_1=""
    LISTE_BAD_ARG=""
    for i in "$@" ; do
        # "--aide" ==> affichage de l'aide en Francais
        # "--help" ==> affichage de l'aide en Anglais
        # "-c" ==> option "CONTROLE" de Esope
        # "-d" ==> option de compilation -g activee
        # "-cd" ==> Option CONTROLE et mode debug
        # "-f" ==> conserve le fichier .f traduit depuis l'Esope
        en fortran77 (Automatique si -d)
        # "-log" ==> conserve la sortie du compilateur dans un
        fichier.log
        # "-Ph" ==> Compilation des sources en parallele sur n
        processus
        # "-S" ==> Code Assembleur de la source en sortie
        # "-z" ==> option de compilation -O0 activee
        # "--ESOPE" ==> Compilation du traducteur ESOPE [SEMT/LM2S
        seulement]

        if [ -f ${i} ] ; then
            if [ `dirname ${i}` == "." ] ; then
                REP=""
            else
                REP=`dirname ${i}`/
            fi
            fi

            case ${i} in
                '--aide') AIDE="VRAI";break;;
                '--help') HELP="VRAI";break;;
                '-c')
                    CONTROLE_MODE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                '-d')
                    COMP_OPTIM="-g"
                ;;
                ;FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                '-cd')
                    CONTROLE_MODE="VRAI";COMP_OPTIM="-g"
                ;;
                ;FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                '-z')
                    COMP_OPTIM="-O0";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                ;;
                '-f')
                    FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                    '-log')
                        LOG_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                ;;
                '-ESOPE')
                    ESOPE_TRADE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                    '-S')
                        COMP_OPTIM="-S -g -fverbose-
asm";FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
                    ;;
                    '-P*')
                        if [[ ! ${i} =~ ^-P([0-9]+)$ ]]; then
                            echo 'Il manque le nombre de processus apres -
P. Exemple : -P4'
                            LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}";
                        fi
                        NBCPU=`echo ${i} | cut -c 3-10`;
                    ;;
                    ;;
                ;;
                '-P')
                    if [[ ! ${i} =~ ^-P([0-9]+)$ ]]; then
                        echo 'Il manque le nombre de processus apres -
P. Exemple : -P4'
                        LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}";
                    fi
                    NBCPU=`echo ${i} | cut -c 3-10`;
                ;;
                ;;
            esac
        else
            COMP_OPTIM="${i}";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";;
            "${i}";basename ${i}.eso`_eso"
        LISTE_ESO_1="${LISTE_ESO_1} ${i}";;
        "${i}";basename ${i}.f.f") LISTE_F_1="${LISTE_F_1} ${i}";;
        ;;
    esac
fi
```

NOTE DE FABRICATION DE CAST3M 2019

```

    "${REP}`basename ${i}.c`.c")      LISTE_C_1="${LISTE_C_1}
${i}";;
*) BAD_ARG="VRAI"
    LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}";
esac
done
fi

if [ "${AIDE}" == "VRAI" ] || [ "${HELP}" == "VRAI" ]; then
# Affichage de l'aide si elle est demandee par l'utilisateur
if [ "${AIDE}" == "VRAI" ]; then
    # Affiche l'aide du script en Francais
    echo ''
    echo "${ETOILE}"
    echo "NOM"
    echo "    compilcast${VERSION} ==> Compilation de sources
Esope (.eso) et C"
    echo ''
    echo "VERSION"
    echo "    Version du Script : ${VERSION}.${REVISION}"
    echo ''
    echo "SYNTAXE"
    echo "    compilcast${VERSION} [OPTION]...
[LISTE_FICHIER]..."
    echo ''
    echo "DESCRIPTION"
    echo "    --aide ==> Affiche le manuel de cette commande
en Francais"
    echo "    --help ==> Affiche le manuel de cette commande
en Anglais"
    echo "    -c    ==> Option CONTROLE d'Esope"
    echo "    -d    ==> Option de compilation en mode debug"
    echo "    -cd   ==> Option CONTROLE et mode debug"
    echo "    -f    ==> Ne supprime pas le fichier fortran77
issu de la traduction"
    echo "        des sources en Esope"
    echo "    -log   ==> conserve la sortie du compilateur
dans un fichier .log"
    echo "    -Pn    ==> Compilation des sources en parallele
sur n processus"
    echo "    -S    ==> Code Assembleur de la source en
sortie"
    echo "    -z    ==> Option de compilation en optimisation
OO"
    echo "    -ESOPE ==> Compilation du traducteur ESOPE
[SEMT/LM2S seulement]"
    echo ''
    echo "EXEMPLES"
    echo "    compilcast${VERSION} fichier1.eso fichier2.eso
fichier3.c"
    echo "        ==> Compile les 3 sources normalement"
    echo ''
    echo "    compilcast${VERSION} -d fichier1.eso fichier2.eso
fichier3.c"
    echo "        ==> Compile les 3 sources en mode debug"
    echo ''
    echo "    compilcast${VERSION} -d -c fichier1.eso
fichier2.eso fichier3.c"
    echo "        ==> Compile les 3 sources en mode debug avec
Esope CONTROLE activee"
    echo ''
    echo "    compilcast${VERSION} *.eso *.c"
    echo "        ==> Compile les toutes les sources Esope et
C du repertoire"
    echo ''
    echo "AUTEUR"
    echo "    Script ecrit par Clement BERTHINIER"
    echo ''
    echo "VOIR AUSSI"
    echo "    Aide du Script 'castem${VERSION}' ==>
'castem${VERSION} --aide'"
    echo "    Aide du Script 'essaicast${VERSION}' ==>
'essaicast${VERSION} --aide'"
    echo "${ETOILE}"
    echo ''

else
# Affiche l'aide du script en Anglais
    echo ''
    echo "${ETOILE}"
    echo "NAME"
    echo "    compilcast${VERSION} ==> Compilation of Esope
(.eso) and C sources"
    echo ''
    echo "VERSION"
    echo "    Script Version : ${VERSION}.${REVISION}"
    echo ''
    echo "SYNTAX"
    echo "    compilcast${VERSION} [OPTION]... [FILE_LIST]..."
    echo ''
    echo "DESCRIPTION"
    echo "    --aide ==> Print the manual of this script in
French"
    echo "    --help ==> Print the manual of this script in
English"
    echo "    -c    ==> CONTROLE Option for Esope"
    echo "    -d    ==> Debug mode compilation option"
    echo "    -cd   ==> CONTROLE Option and debug mode"
    echo "    -f    ==> Keep the fortran77 source file
produced by the conversion"
    echo "        from Esope sources"
    echo "    -log   ==> Keep the output of the compiler in a
.log file"
    echo "    -Pn    ==> Parallel compilation of sources whith
n process"
    echo "    -S    ==> Assembler Code of the source file as
output"

```

```

    echo "    -z    ==> Optimisation OO for the compilation"
    echo "    -ESOPE ==> Compilation ESOPE translator
[SEMT/LM2S only]"
    echo ''
    echo "EXAMPLES"
    echo "    compilcast${VERSION} fichier1.eso fichier2.eso
fichier3.c"
    echo "        ==> Compile the 3 sources normally"
    echo "    compilcast${VERSION} -d fichier1.eso fichier2.eso
fichier3.c"
    echo "        ==> Compile the 3 sources with debug mode"
    echo "    compilcast${VERSION} -d -c fichier1.eso
fichier2.eso fichier3.c"
    echo "        ==> Compile the 3 sources with debug mode
and Esope CONTROLE mode"
    echo ''
    echo "    compilcast${VERSION} *.eso *.c"
    echo "        ==> Compile all the Esope and C sources in
the directory"
    echo ''
    echo "AUTHOR"
    echo "    Script written by Clement BERTHINIER"
    echo ''
    echo "SEE ALSO"
    echo "    Manual for 'castem${VERSION}' script ==>
'castem${VERSION} --help'"
    echo "    Manual for 'essaicast${VERSION}' script ==>
'essaicast${VERSION} --help'"
    echo "${ETOILE}"
    fi

else
if [ "${BAD_ARG}" == "VRAI" ]; then
# Sortie d'erreur sur les arguments
    echo ''
    echo "${ETOILE}"

    if [ "${LISTE_BAD_ARG}" == "" ]; then
        echo "Aucun argument / No arguments"
        echo ''

    else
        echo "Liste des arguments invalides / List of unavailable
arguments"
        echo "${LISTE_BAD_ARG}"
        echo ''
        fi

        echo "    Pour en savoir d'avantage : ' compilcast${VERSION} -
-aide ''"
        echo "    To learn more : ' compilcast${VERSION} -
-help ''"
        echo "${ETOILE}"
        NIV_ERR="VRAI"
        fi

    if [ "${GFORBIN}" == "FAUX" ]; then
        echo ''
        echo "${ETOILE}"
        echo "    Vous avez besoin de gfortran (version 4.3.2 ou
superieure)"
        echo "    You need gfortran (version 4.3.2 or
above)"
        echo "        NIV_ERR="VRAI"
        fi

    if [ "${GCCBIN}" == "FAUX" ]; then
        echo ''
        echo "${ETOILE}"
        echo "    Vous avez besoin de gcc (version 4.3.2 ou
superieure)"
        echo "    You need gcc (version 4.3.2 or
above)"
        echo "        NIV_ERR="VRAI"
        fi

# Teste si les fichiers .eso contenus dans les listes existent
LISTE_ESO_2=""
for i in ${LISTE_ESO_1}; do
if [ ! -f ${i} ]; then
    echo "Fichier inexistant / Non existing file : '${i}'"
else
    LISTE_ESO_2="${LISTE_ESO_2} ${i}"
    LISTE_SRC="${LISTE_SRC} ${i}"
fi
done

# Teste si les fichiers .f contenus dans les listes existent
LISTE_F_2=""
for i in ${LISTE_F_1}; do
if [ ! -f ${i} ]; then
    echo "Fichier inexistant / Non existing file : '${i}'"
else
    LISTE_F_2="${LISTE_F_2} ${i}"
    LISTE_SRC="${LISTE_SRC} ${i}"
fi
done

# Teste si les fichiers .c contenus dans les listes existent
LISTE_C_2=""
for i in ${LISTE_C_1}; do
if [ ! -f ${i} ]; then
    echo "Fichier inexistant / Non existing file : '${i}'"

```

```

else
  LISTE_C_2=$(LISTE_C_2) ${i}"
  LISTE_SRC=$(LISTE_SRC) ${i}"
fi
done

if [ $NBCPU -gt 1 ]; then
  PARALLELE_COMP="VRAI"
elif [ $NBCPU -eq 1 ]; then
  PARALLELE_COMP="FAUX"
else
  echo 'Le nombre de processus apres -P est invalide'
fi

if [ "$PARALLELE_COMP" = "FAUX" ]; then

  # Options de Compilation du fortran
  if [ $(BIT) == "64" ]; then
    f_options="-v -Wall -m$(BIT) ${COMP_OPTIM} -fdefault-
integer-8 -fdefault-real-8 -fdefault-double-8 -fopt-info -
ftracer -pthread -mtune=generic -mfpmath=sse -fgcse-sm -fgcse-
las -freciprocal-math -ftree-loop-distribution -fno-trapping-
math -fno-aggressive-loop-optimizations -finit-real=nan -finit-
integer=-2147483647 -finit-logical=false -frename-registers -
frecursive -fopenmp
-c"
    if test $(PLATEFORME) = "MAC" ; then
      f_options="-v -Wall -m$(BIT) ${COMP_OPTIM} -fdefault-
integer-8 -fdefault-real-8 -fdefault-double-8 -fopt-info -
ftracer -pthread -mtune=generic -mfpmath=sse -fgcse-sm -fgcse-
las -freciprocal-math -ftree-loop-distribution -fno-trapping-
math -fno-aggressive-loop-optimizations -finit-real=nan -finit-
integer=-2147483647 -finit-logical=false -frename-registers -
frecursive -fopenmp
-c"
    fi
  else
    f_options="-v -Wall -m$(BIT) ${COMP_OPTIM} -fopt-info -
ftracer -pthread -mtune=generic -march=pentium4 -msse -
mfpmath=sse -fgcse-sm -fgcse-las -freciprocal-math -ftree-loop-
distribution -fno-trapping-math -fno-aggressive-loop-
optimizations -finit-real=nan -finit-integer=-2147483647 -finit-
logical=false -frename-registers -frecursive -fopenmp -fno-tree-
dse -fno-tree-fre -fno-tree-pre -fno-tree-loop-vectorize -fno-
tree-dominator-opts -fno-predictive-commoning -fno-caller-saves
-c"
  fi
fi

#####
# Traite la liste des ".eso"
#####

if [ "$(LISTE_ESO_2)" != "" ] ; then

  if [ "$GFORBIN" != "FAUX" ] ; then
    # Variables d'environnement pour Esope
    if [ "$(CONTROLE_MODE)" == "VRAI" ]; then
      export
    ESOPE_PARAM="NORME=TRADUCTEUR, FORT=UNIX$(BIT), ESOPE=10000000, CON-
TROL"
    else
      export
    ESOPE_PARAM="NORME=TRADUCTEUR, FORT=UNIX$(BIT), ESOPE=10000000"
    fi

    # Calcul du nombre de fichier ".eso" à traiter
    nb_eso=$( \ls -1 $(LISTE_ESO_2) | wc -l )

    # Calcul du nombre d'include ".INC" présents dans le
    répertoire courant
    nb_INC=$( \ls -1 *.INC 2> /dev/null | wc -l )

    # Menage d'une compilation précédente
    for i in $(LISTE_ESO_2) ; do
      i_base=`basename $i .eso`
      if [ -f ${i_base}.o ] ; then \rm -f ${i_base}.o ;
    fi
    if [ -f ${i_base}.f ] ; then \rm -f ${i_base}.f ;
    fi
    if [ -f ${i_base}.lst ] ; then \rm -f ${i_base}.lst ;
    fi
    if [ -f ${i_base}.log ] ; then \rm -f ${i_base}.log ;
    fi
  done

  # Compilations des ".eso"
  COMPT=0

  # Répertoire des includes pour le traducteur ESOPE
  (possibilité de le surcharger par la variable d'environnement
  "ESOPE_INC")
  if [ "$(ESOPE_INC)" == "" ] ;then
    if [ "$(ESOPE_TRAD)" == "VRAI" ] ;then
      export ESOPE_INC=$ESOPE_REP/include
    else
      export ESOPE_INC=$CASTEM_REP/include/eso
    fi
  fi

  for i in $(LISTE_ESO_2) ; do
    echo
    echo $i
    i_base=`basename $i .eso`


  export ESOPE_OUT=${i_base}.f
  export ESOPE_LST=${i_base}.lst

  # Conversion de l'Esope en fortran
  $ESOPE_EXEC < $i
  CODE_RETUR_ESOPE=$?

  # On compile certaines sources avec des options
  différentes :
  if [ ${i_base} == "ddot2" ] ; then
    f_options_associative=""
  else
    f_options_associative="-fassociative-math -fno-
signed-zeros"
  fi

  if [ ${CODE_RETUR_ESOPE} = '0' ] ; then
    if [ -f ${i_base}.lst ] ; then \rm -f ${i_base}.lst
  fi

  $GFORTRAN ${f_options} ${f_options_associative}
${i_base}.f -I$(MPI_INCREP) > ${i_base}.log 2>&1

  #Assembleur "-S" :
  if [ -f ${i_base}.s ] ; then
    as ${i_base}.s -o ${i_base}.o
  fi

  # Verification de la compilation de la source
  if [ ! -f ${i_base}.o ] ; then
    echo Erreur de compilation consultez ${i_base}.log
    grep -i -nB 4 'Error:' ${i_base}.log
    echo
    echo ${i_base}.f ' Non compile' >>
ZZZ_Non_Compile.log
  else
    # Nettoyage des fichiers
    if [ "${FORTRAN_FILE}" != "VRAI" ] ; then
      rm -f ${i_base}.f
    fi
    if [ "${LOG_FILE}" == "FAUX" ] ; then
      rm -f ${i_base}.log
    fi
  fi

  else
    echo Erreur de traduction de consultez ${i_base}.lst
    grep -i -nB 6 'ERROR :.' ${i_base}.lst
    echo
    echo ${i_base}.eso ' Non traduit en FORTRAN' >>
ZZZ_Non_traduit.log
  fi

  COMPT=$((COMPT+1))
  echo "Sources traitees : ${COMPT}/${nb_eso}"

done

fi
fi

#####
# Traite la liste des ".f"
#####

if [ "$(LISTE_F_2)" != "" ] ; then
  if [ "$GFORBIN" != "FAUX" ] ; then

    # Calcul du nombre de fichier ".f" à traiter
    nb_f=$( \ls -1 $(LISTE_F_2) | wc -l )

    # Menage d'une compilation précédente
    for i in $(LISTE_F_2) ; do
      i_base=`basename $i .f`
      if [ -f ${i_base}.o ] ; then \rm -f ${i_base}.o ;
    fi
    if [ -f ${i_base}.log ] ; then \rm -f ${i_base}.log ;
    fi
  done

  # Compilations des ".f"
  COMPT=0

  for i in $(LISTE_F_2) ; do
    echo
    echo $i
    i_base=`basename $i .f`


    #On compile certaines sources avec des options
    différentes :
    if [ ${i_base} == "ddot2" ] ; then
      f_options_associative=""
    else
      f_options_associative="-fassociative-math -fno-
signed-zeros"
    fi

    $GFORTRAN ${f_options} ${f_options_associative} ${i} -
I$(MPI_INCREP) > ${i_base}.log 2>&1

    #Assembleur "-S" :
    if [ -f ${i_base}.s ] ; then
      as ${i_base}.s -o ${i_base}.o
    fi
  done
fi

```



```

# Verification de la compilation de la source
if [ ! -f ${i_base}.o ]; then
    echo Erreur de compilation consultez ${i_base}.log
    grep -i -nB 4 'Error:' ${i_base}.log
    echo
    echo ${i_base}.f ' Non compile' >> ZZZ_Non_Compile.log
else
    # Nettoyage des fichiers
    if [ "${LOG_FILE}" == "FAUX" ]; then
        rm -f ${i_base}.log
    fi
    done
    COMPT=$((COMPT+1))
    echo "Sources traitees : ${COMPT}/${nb_f}"
done
fi

#####
# Traite la liste des ".c"
#####

if [ "${LISTE_C_2}" != "" ] ; then
    # Calcul du nombre de fichier ".c" à traiter
    nb_c=$(ls -1 ${LISTE_C_2} | wc -l)

    # Ménage d'une compilation précédente
    for i in ${LISTE_C_2} ; do
        i_base=`basename $i .c`
        if [ -f ${i_base}.o ]; then rm -f ${i_base}.o ;
    fi
    if [ -f ${i_base}.log ]; then rm -f ${i_base}.log ;
    fi
done

# Compilations des ".c"
COMPT=0
# Sur Yosemite, le compilateur GCC ne permet des
optimisation que de niveau -O3 au max
if test ${PLATEFORME} = "MAC" ; then
    COMP_OPTIM="-O3"
fi
for i in ${LISTE_C_2} ; do
    # Compilations des .c
    echo
    echo $i
    i_base=`basename $i .c`

    SIGCC -Wall -m${BIT} ${COMP_OPTIM} ${CASTEM_DEFINE} -
DDATE_ONLY -DFOR_LINUX ${CASTEM_INC} -c $i > ${i_base}.log 2>&1

    # Assembleur "-S" :
    if [ -f ${i_base}.s ]; then
        as ${i_base}.s -o ${i_base}.o
    fi

    # Vérification de la compilation de la source
    if [ ! -f ${i_base}.o ]; then
        echo Erreur de compilation consultez ${i_base}.log
        grep -i -nB 4 'Error:' ${i_base}.log
        echo
        echo ${i_base}.c ' Non compile' >> ZZZ_Non_Compile.log
    else
        if [ "${LOG_FILE}" == "FAUX" ]; then
            rm -f ${i_base}.log
        fi
        rm -f `basename $i .c`.log
    fi
    COMPT=$((COMPT+1))
    echo "Sources traitees : ${COMPT}/${nb_c}"
done
fi

else # "$PARALLELLE_COMP" = "VRAI"
j=1
LISTE_SRC_ARR=(${LISTE_SRC})
LISTE_SRC_ARR_SIZE=${#LISTE_SRC_ARR[*]}
for i in ${!LISTE_SRC_ARR[*]} ; do
    if [ ${i} -gt $(( ${LISTE_SRC_ARR_SIZE}*${j}/$NCPUs )) ] ;
then
        compilcast ${VERSION} ${OPTIONS_SCRIPT} ${LISTE_SRC_TEMP}
        LISTE_SRC_TEMP=""
        let j++
    fi
    LISTE_SRC_TEMP="${LISTE_SRC_TEMP} ${LISTE_SRC_ARR[$i]}"
done
compilcast ${VERSION} ${OPTIONS_SCRIPT} ${LISTE_SRC_TEMP} &
wait

#####
# Affichage en sortie de Script selon ce qu'il s'est passé
#####

```

```

#####
if [ -f ZZZ_Non_traduit.log ]; then
    echo ''
    echo "${ETOILE}"
    echo 'Erreurs de traduction : '
    echo ''
    cat ZZZ_Non_traduit.log
    echo ''
    echo ''
    echo "consultez les fichiers '.lst'"
    echo "${ETOILE}"
    NIV_ERR="VRAI"
fi

if [ -f ZZZ_Non_Compile.log ]; then
    echo ''
    echo "${ETOILE}"
    echo 'Erreurs de compilation : '
    echo ''
    cat ZZZ_Non_Compile.log
    echo ''
    echo ''
    echo "consultez les fichiers '.log'"
    echo "${ETOILE}"
    NIV_ERR="VRAI"
fi

if [ ${SAIDE} == "FAUX" ] && [ ${HELP} == "FAUX" ] ; then
    if [ ${NIV_ERR} == "FAUX" ] ; then
        echo ''
        echo "${ETOILE}"
        echo '*' Tache terminee avec succes / Successfully
completed job *
        echo "${ETOILE}"
        echo ''
    else
        echo ''
        echo "${ETOILE}"
        echo '*' Des erreurs sont survenues / Errors have
occurred
        echo "${ETOILE}"
        echo ''
    fi
fi

#####
# Vérification par ftnchek
#####

if [ -f ZZZ_ftnchek.log ]; then rm -f ZZZ_ftnchek.log; fi
if [ -x ${CASTEM REP}/bin/ftnchek-${BIT}bits ]; then
    # Calcul du nombre de fichier .f présents dans le
repertoire courant
    ls -1 *.* > /dev/null 2>&1
    CODE_RETOUR=$?

    if [ ${CODE_RETOUR} -eq 0 ] ; then
        ${CASTEM REP}/bin/ftnchek-${BIT}bits -
include=${MPI_INCREP} -nopretty -notruncation -crossref=calls -
errors=100 -f77character,continuation,cray-pointer,format-edit-
descr,implicit-none,intrinsic,long-name,mixed-expr,long-
line,quotemark,relops,semicolon,statement-order -brief -wrap=500
-calltree *.f > ZZZ_ftnchek_1.log

        # Recherche des variables "Utilisees non initialisées"
        NBR_VAR=`grep -i 'is used' ZZZ_ftnchek_1.log | wc -l`
        if [ "${NBR_VAR}" != "0" ]; then
            echo "${ETOILE}"
        >> ZZZ_ftnchek.log
            echo " ${NBR_VAR} VARIABLES NON INITIALISEES detectées"
        >> ZZZ_ftnchek.log
            echo "${ETOILE}"
        >> ZZZ_ftnchek.log
            echo ""
        >> ZZZ_ftnchek.log
            grep -i -nB 1 'is used' ZZZ_ftnchek_1.log
        >> ZZZ_ftnchek.log
            echo "${ETOILE}"
        >> ZZZ_ftnchek.log
            echo ""
        >> ZZZ_ftnchek.log
            fi

        # Recherche des SUBROUTINES "Not Called"
        NBR_SUB=`grep -i 'not called' ZZZ_ftnchek_1.log | grep -
iv 'entry' | wc -l`
        if [ "${NBR_SUB}" != "0" ]; then
            echo "${ETOILE}"
        >> ZZZ_ftnchek.log
            echo " ${NBR_SUB} SUBROUTINES JAMAIS APPELEES"
        >> ZZZ_ftnchek.log
            echo "${ETOILE}"
        >> ZZZ_ftnchek.log
            echo ""
        >> ZZZ_ftnchek.log
            grep -i 'not called' ZZZ_ftnchek_1.log | grep -iv
'entry' >> ZZZ_ftnchek.log
            echo "${ETOILE}"
        >> ZZZ_ftnchek.log
    fi
fi

```

```

        echo ""
>> ZZZ_ftnchek.log
    fi

    # Recherche des erreurs de syntaxe
    NBR_ERREURS=`grep "syntax error" ZZZ_ftnchek_1.log | fgrep -v -e "0" -e "near" | wc -l`
    if [ "$NBR_ERREURS" != "0" ]; then
        echo "${ETOILE}"
    fi
>> ZZZ_ftnchek.log
    echo "Bilan des ERREURS trouvées par ftnchek-"
$(BIT)bits" >> ZZZ_ftnchek.log
    echo "${ETOILE}"
>> ZZZ_ftnchek.log
    grep "syntax error" ZZZ_ftnchek_1.log | fgrep -v -e
"0" -e "near" >> ZZZ_ftnchek.log
    echo ""
>> ZZZ_ftnchek.log
    fi

    # Recherche des warnings d'incompatibilité de Type
    NBR_INCOMP=`grep -i "incompatible" ZZZ_ftnchek_1.log | wc -l`
    if [ "$NBR_INCOMP" != "0" ]; then
        echo "${ETOILE}"
    fi
>> ZZZ_ftnchek.log
    echo "Bilan des INCOMPATIBILITES trouvées par ftnchek-"
$(BIT)bits" >> ZZZ_ftnchek.log
    echo "${ETOILE}"
>> ZZZ_ftnchek.log
    grep -i -nA 1 "incompatible" ZZZ_ftnchek_1.log
>> ZZZ_ftnchek.log
    echo ""
>> ZZZ_ftnchek.log
    fi

    # Recherche des subroutines avec des Noms de plus de 6
    caracteres
    NBR_NOMS6=`grep -i "Names longer than 6 chars" ZZZ_ftnchek_1.log | wc -l`
    if [ "$NBR_NOMS6" != "0" ]; then
        echo "${ETOILE}"
    fi
>> ZZZ_ftnchek.log
    echo "Bilan des NOMS de PLUS de 6 CARACTERES"
>> ZZZ_ftnchek.log
    echo "${ETOILE}"
>> ZZZ_ftnchek.log
    grep -i "Names longer than 6 chars" ZZZ_ftnchek_1.log
>> ZZZ_ftnchek.log
    echo ""
>> ZZZ_ftnchek.log
    fi

    # Séparation
    echo "${ETOILE}"
>> ZZZ_ftnchek.log
    echo "                    Rapport Détailé"
>> ZZZ_ftnchek.log
    echo "${ETOILE}"
>> ZZZ_ftnchek.log

    # Fusion des fichiers
    cat ZZZ_ftnchek_1.log
>> ZZZ_ftnchek.log
    rm -f ZZZ_ftnchek_1.log
    fi
    fi
    fi
    fi

```

Annexe D. Scripts essaicast19.bat et essaicast19

a. Script essaicast19.bat

```

@ECHO OFF

SETLOCAL ENABLEDELAYEDEXPANSION
SET SCRIPT_REP=%~dp0
REM Chargement de l'environnement (Laisser %CASTEM_VERSION%, il est remplacé automatiquement par InstallJammer)
CALL %SCRIPT_REP%~-1%\environnement_Cast3M%XXANNEEXX%

REM Options de noms à rechercher pour l'édition des liens
IF %BITS% EQU 64 (
    SET U_OPTIONS=-u __BLOCK_DATA__ -u __main
    SET LD_FLAGS=
) ELSE (
    SET U_OPTIONS=-u __BLOCK DATA__ -u __main
    SET LD_FLAGS=-Wl,--large-address-aware)

REM Initialisations :
SET AIDE=
SET HELP=
SET VERBOS=-
SET LOG FILE=
SET /A ERROLEV=0
SET INDICE_CONTROLE=
SET INDICE_DEBUG=

REM Lecture des Arguments d'entrée
:DEBUT_LECTURE
SET ArgNAME1=%~1
REM Affichage de l'aide en Français
IF "%ArgNAME1%"=="-aide" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)
IF "%ArgNAME1%"=="/?" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)

REM Affichage de l'aide en Anglais
IF "%ArgNAME1%"=="-help" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)
IF "%ArgNAME1%"=="-h" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)

REM Edition des liens avec la librairie _c de Cast3M
IF "%ArgNAME1%"=="-c" (
    SET INDICE_CONTROLE=c
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Edition des liens avec la librairie _d de Cast3M
IF "%ArgNAME1%"=="-d" (
    SET INDICE_DEBUG=d
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Edition des liens avec la librairie _cd de Cast3M
IF "%ArgNAME1%"=="-cd" (
    SET INDICE_CONTROLE=c
    SET INDICE_DEBUG=d
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Fabrication de la librairie et Edition des liens du traducteur ESOPE
IF "%ArgNAME1%"=="-ESOPE" (
    SET ESOPE=VRAI
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Appel au linker en mode verbose
IF "%ArgNAME1%"=="-v" (
    SET VERBOS=-v
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Conservation du .log à la fin de la compilation
IF "%ArgNAME1%"=="-log" (
    SET LOG_FILE=VRAI
    SHIFT
    GOTO DEBUT_LECTURE
)

:LABEL_AIDE
IF DEFINED AIDE (
    REM Affiche l'aide en Français
)

```

```

ECHO .
ECHO %LIGNEUP%
ECHO NAME
ECHO essaicast%CASTEM_VERSION%%SPY% : Procéder à
l'édition des liens
ECHO Site web : http://www-cast3m.cea.fr/
ECHO %LIGNEMIDDLE%
ECHO VERSION
ECHO Version du script :
%CASTEM_VERSION%%SPY%.%CASTEM_REVISION%
ECHO %LIGNEMIDDLE%
ECHO SYNTAXE
ECHO essaicast%CASTEM_VERSION%%SPY% [OPTION]
ECHO %LIGNEMIDDLE%
ECHO DESCRIPTION
ECHO --aide : Affiche le manuel de cette commande en
français
ECHO --help : Affiche le manuel de cette commande en
anglais
ECHO -v : invoque le linker en mode verbose
ECHO -log : conserve le .log à l'issue de l'édition
des liens
ECHO -c : Utilisation de la librairie "_c"
[SEMT/LM2S seulement]
ECHO -d : Utilisation de la librairie "_d"
[SEMT/LM2S seulement]
ECHO -cd : Utilisation de la librairie "_cd"
[SEMT/LM2S seulement]
ECHO -ESOPE : Librairie et Traducteur ESOPE
[SEMT/LM2S seulement]
ECHO %LIGNEMIDDLE%
ECHO EXEMPLES
ECHO essaicast%CASTEM_VERSION%%SPY%
ECHO Réalise l'édition des liens dans le répertoire
courant et
ECHO produit un exécutable binaire
cast_%BITS%_%CASTEM_VERSION%.exe%SPY%
ECHO En cas d'échec un fichier
link cast_%BITS%_%CASTEM_VERSION%.log%SPY% est généré
ECHO %LIGNEMIDDLE%
ECHO AUTEUR
ECHO Script écrit par Clement BERTHINIER
ECHO %LIGNEMIDDLE%
ECHO VOIR AUSSI
ECHO Aide du Script 'castem%CASTEM_VERSION%`%SPY% :
'castem%CASTEM_VERSION% --aide`%SPY%
ECHO Aide du Script 'compilcast%CASTEM_VERSION%`%SPY% :
'compilcast%CASTEM_VERSION% --aide`%SPY%
ECHO %LIGNEDOWN%
EXIT /B 0
)
:LABEL_HELP
IF DEFINED HELP (
    REM Affiche l'aide en Français
    ECHO .
    ECHO %LIGNEUP%
    ECHO NAME
    ECHO essaicast%CASTEM_VERSION%%SPY% : Proceed to the
link
    ECHO Site web : http://www-cast3m.cea.fr/
    ECHO %LIGNEMIDDLE%
    ECHO VERSION
    ECHO Script version :
%CASTEM_VERSION%%SPY%.%CASTEM_REVISION%
    ECHO %LIGNEMIDDLE%
    ECHO SYNTAX
    ECHO essaicast%CASTEM_VERSION%%SPY%
    ECHO %LIGNEMIDDLE%
    ECHO DESCRIPTION
    ECHO --aide : Print the manual of this script in French
    ECHO --help : Print the manual of this script in
English
    ECHO -v : invoque the linker in verbose mode
    ECHO -log : keep the .log file at the end of the link
process
    ECHO -c : Use the library "_c" for the link
[SEMT/LM2S only]
    ECHO -d : Use the library "_d" for the link
[SEMT/LM2S only]
    ECHO -cd : Use the library "_cd" for the link
[SEMT/LM2S only]
    ECHO -ESOPE : Librairie and ESOPE Translator
[SEMT/LM2S only]
    ECHO %LIGNEMIDDLE%
    ECHO EXAMPLES
    ECHO essaicast%CASTEM_VERSION%%SPY%
)

```

```

ECHO Proceed to the link in the current directory
and generate a
ECHO binary executable
cast %BIT%_CASTEM_VERSION%.exe%SPY%
ECHO
ECHO If the process fail the file
link_cast_%BIT%_CASTEM_VERSION%.log%SPY% is created
ECHO
ECHO %LIGNEMIDDLE%
ECHO AUTEUR
ECHO Script written by Clement BERTHINIER
ECHO
ECHO %LIGNEMIDDLE%
ECHO SEE ALSO
ECHO Manual for 'castem%CASTEM_VERSION%'%SPY% :
'castem%CASTEM_VERSION% --help'%SPY%
ECHO Manual for 'compilcast%CASTEM_VERSION%'%SPY% :
'compilcast%CASTEM_VERSION% --help'%SPY%
ECHO %LINEDOWN%
EXIT /B 0
)

REM Fin de la lecture des arguments
:FIN_LECTURE

SET SUFFIX1=
IF NOT "%INDICE_CONTROLE%"%INDICE_DEBUG%"==" (
    SET SUFFIX1=%INDICE_CONTROLE% %INDICE_DEBUG%
)

IF NOT DEFINED ESOPE GOTO :CAST3M_SOURCES
REM
***** Cas de
la Librairie ESOPE et du Traducteur
*
REM
***** Cas de
la Librairie ESOPE et du Traducteur
*
REM
***** Creation de la librairie
ECHO.
ECHO %LIGNEUP%
ECHO Mise a jour de la librairie %LIBESOPE%
ECHO %LINEDOWN%
CALL %AR% -ruvs %ESOPE_REP%\lib\%LIBESOPE% *.o >nul 2>&1
CALL %AR% -d %ESOPE_REP%\lib\%LIBESOPE% esop10.o >nul 2>&1

REM Appel a gfortran pour faire 1 edition des liens
CALL %FC% %VERBOSE% -Wall -static -static-libgfortran -static-
libgcc -m%BIT% -o %esopX_Local%. *.* %ESOPE_REP%\lib\%LIBESOPE%
-lmingwthrd > link_%esopX_Local%.log 2>&1

REM Test d existence de %esopX_Local%.exe
SET /A ERROLEV=0
IF EXIST "%esopX_Local%.exe" (
    ECHO.
    ECHO %LIGNEUP%
    ECHO Edition des liens effectuee
    ECHO %LINEDOWN%
    IF EXIST "link_%esopX_Local%.log" DEL /Q
    link_%esopX_Local%.log"
)

REM Test d existence de esopX.exe
IF NOT EXIST "%ESOPE_REP%\bin\%esopX%.exe" (
    ECHO.
    ECHO %LIGNEUP%
    ECHO Generation de l'executable %esopX%.exe%SPY%
    ECHO %LINEDOWN%
) ELSE (
    ECHO.
    ECHO %LIGNEUP%
    ECHO Mise a jour de l'executable %esopX%.exe
    ECHO %LINEDOWN%
    MOVE "%ESOPE_REP%\bin\%esopX%.exe"
    "%ESOPE_REP%\bin\%esopX%.old">nul

    COPY %esopX_Local%.exe "%ESOPE_REP%\bin\%esopX%.exe">nul
) ELSE (
    SET /A ERROLEV=16
    ECHO.
    ECHO %LIGNEUP%
    ECHO ECHEC durant l'Edition des liens :

    IF EXIST "link_%esopX_Local%.log" (
        ECHO Consultez le fichier link_%esopX_Local%.log%SPY%
        ECHO %LINEDOWN%
    )
)

EXIT /B !ERROLEV!
:CAST3M_SOURCES

```

```

REM ****
REM *
Cas de Cast3M
*
REM ****
REM Repertoire ou trouver la licence
SET DIRLIC=%CASTEM_REP%\licence

REM Le nom du binaire depend du type de licence
IF EXIST "%DIRLIC%" (
    SET castX=bin_Cast3M_Win_INDUS_%BIT%_CASTEM_VERSION%
    SET LIBCASTEM=libcastem_INDUS_%BIT%.a
) ELSE (
    SET castX=bin_Cast3M_Win_DEVEL_%BIT%_CASTEM_VERSION%
    SET LIBCASTEM=libcastem_DEVEL_%BIT%_SUFFIX1%.a
)

REM Nettoyage des fichiers preliminaires
SET castX_Local=cast_%BIT%_CASTEM_VERSION%
IF EXIST %castX_Local%.exe DEL /Q %castX_Local%.exe

REM Test d existence de %LIBCASTEM%
IF NOT EXIST "%CASTEM_REP%\lib%BIT%\%LIBCASTEM%" (
    ECHO.
    ECHO %LIGNEUP%
    ECHO Generation de la librairie %LIBCASTEM%
    ECHO %LINEDOWN%
    CALL %AR% -ruvs %LIBCASTEM% *.o >nul 2>&1
    MOVE %LIBCASTEM% "%CASTEM_REP%\lib%BIT%">nul
    SET O_BINARY=%CASTEM_REP%\lib%BIT%\%LIBCASTEM%
) ELSE (
    REM Calcul du nombre de .o
    SET /A ocompt=0
    FOR %i IN (*.o) DO SET /A ocompt+=1

    IF !ocompt! EQU 0 (
        SET O_BINARY=%LIBCASTEM%
    ) ELSE IF !ocompt! GTR 3200 (
        CALL XCOPY /Y "%CASTEM_REP%\lib%BIT%\%LIBCASTEM%" >nul
        CALL %AR% -ruvs %LIBCASTEM% *.o >nul
    2>&1
        SET O_BINARY=%LIBCASTEM%
    ) ELSE (
        SET O_BINARY=*.o "%CASTEM_REP%\lib%BIT%\%LIBCASTEM%")
)

ECHO.
ECHO Utilisation de %O_BINARY%
REM Appel a gfortran pour faire 1 edition des liens
CALL %MINGWPATH%\gfortran %VERBOSE% -g -O %U_OPTIONS% -Wall -
-m%BIT% -o %castX_Local%. %O_BINARY% -static -static -static-libgfortran
-static-libgcc -pthread -lpthread -mthreads -mwin32 -mconsole -
lmingwthrd -L"%CASTEM_REP%\lib%BIT%" -lesope_%BIT% -
lcastem_dll_%BIT% -ljpeg -lfreetglut.dll %BIT% -lopengl32 -
lfdr %BIT% -lmedfwrap -lmedc -lhdf5 -lstdc++ -lgomp %LD_FLAGS%
> link_%castX_Local%.log 2>&1

REM Test d existence de %castX_Local%.exe
SET /A ERROLEV=0
IF EXIST "%castX_Local%.exe" (
    ECHO.
    ECHO %LIGNEUP%
    ECHO Edition des liens effectuee
    ECHO %LINEDOWN%
    IF NOT DEFINED LOG_FILE (
        IF EXIST "link_%castX_Local%.log" DEL /Q
    link_%castX_Local%.log"
    )

    REM Test d existence de castX.exe
    IF NOT EXIST "%CASTEM_REP%\bin\%castX%.exe" (
        ECHO.
        ECHO %LIGNEUP%
        ECHO Generation de l'executable %castX%.exe%SPY%
        ECHO %LINEDOWN%
        REM Le "ECHO F" vient pour dire que c'est un fichier
        qui est copie et non un repertoire...
        ECHO F | XCOPY "%castX_Local%.exe" "%CASTEM_REP%\bin\%castX%.exe">nul
    )

    ) ELSE (
        SET /A ERROLEV=16
        ECHO.
        ECHO %LIGNEUP%
        ECHO ECHEC durant l'Edition des liens :

        IF EXIST "link_%castX_Local%.log" (
            ECHO Consultez le fichier link_%castX_Local%.log%SPY%
            TYPE link_%castX_Local%.log
            ECHO %LINEDOWN%
        )
    )
)

IF EXIST %LIBCASTEM% DEL /Q %LIBCASTEM%
EXIT /B !ERROLEV!

```

b. Script essaicast19

```
#!/bin/bash
```

```
#Recuperation des variables d'environnement
```

NOTE DE FABRICATION DE CAST3M 2019

```

if [ "${VERSION}" == "" ] ; then VERSION=%{XXANNEEXX}
; fi
if [ "${REVISION}" == "" ] ; then REVISION=%{XXREVISIONXX} ;
; fi
if [ "${PLATEFORME}" == "" ] ; then PLATEFORME=%{XXPLATEFORMEXX}
; fi
if [ "${CASTEM_REPO}" == "" ] ; then CASTEM_REPO=%{INSTALL_PATH} ;
fi

# Detection de l'architecture ou forcee si BIT est defini dans
l'environnement
if [ `uname -m` == "x86_64" ] || [ "$BIT" == "64" ] && [ !
"$BIT" == "32" ] ; then
    BIT="64"
    GNU_VER="8.2.0"
    GNU_PATH=$(CASTEM_REPO)/GCC/GCC-x86_64
else
    BIT="32"
    GNU_VER="6.3.0"
    GNU_PATH=$(CASTEM_REPO)/GCC/GCC-i686
fi

# Repertoires lies a MPI (includes)
# Variable d'environnement definissant la commande MPI :
MPI_RUNCMD
if [ "${MPI_LIBREP}" == "" ] ; then
MPI_LIBREP=$(CASTEM_REPO)/mpi/mpis$BIT/lib ; fi
if [ "${MPI_EXCREP}" == "" ] ; then
MPI_EXCREP=$(CASTEM_REPO)/mpi/mpis$BIT/bin ; fi
if [ "${MPI_INCREP}" == "" ] ; then
MPI_INCREP=$(CASTEM_REPO)/mpi/mpis$BIT/include ; fi

# Repertoire pour les librairies necessaires (pour l'execution
ou l'édition des liens)
if [ "${PLATEFORME}" == "Linux" ] ; then
    export
LD_LIBRARY_PATH=$(CASTEM_REPO)/lib$BIT:$GNU_PATH/lib:$GNU_P
ATH)/lib64:$MPI_LIBREP:$LD_LIBRARY_PATH"
    # Parfois "ld" ne trouve pas crt1.o il faut ajouter des
reertoires a LIBRARY_PATH
    export
LIBRARY_PATH=$(LIBRARY_PATH):/usr/lib64:/usr/lib/x86_64-linux-
gnu:/usr/lib:/usr/lib/i386-linux-gnu"
elif [ "${PLATEFORME}" == "MAC" ] ; then
    export
DYLD_LIBRARY_PATH=$(CASTEM_REPO)/lib$BIT:$GNU_PATH/lib:$GNU_
PATH)/lib64:$MPI_LIBREP:$DYLD_LIBRARY_PATH"
fi

# Ajout du chemin vers mpirun au PATH
if [ -x $MPI_EXCREP/mpirun ] ; then
    export PATH=$MPI_EXCREP:$PATH
fi

# Detection de la variable d'environnement GFORTTRAN
if [ "$GFORTTRAN" == "" ] ; then
    GFORTTRAN=$(GNU_PATH)/bin/gfortran-$GNU_VER
fi

if [ "$GFORTTRAN" == "" ] ; then
    GFORBIN="FAUX"
else
    GFORBIN="VRAI"
fi

# Detection de la variable d'environnement GCC
if [ "$GCC" == "" ] ; then
    GCC=$(GNU_PATH)/bin/gcc-$GNU_VER
fi

if [ "$GCC" == "" ] ; then
    GCCBIN="FAUX"
else
    GCCBIN="VRAI"
fi

# Fonction de recherche de bibliothèques
searchPath="/usr/lib64:/usr/local/lib64:/usr/lib/x86_64-linux-
gnu:/usr/X11/lib64:/usr/X11R6/lib64:/opt/X11/lib:/usr/lib:/usr/l
ocal/lib:/usr/lib/i386-linux-
gnu:/usr/X11/lib:/usr/X11R6/lib:${CASTEM_REPO}/lib$BIT"
function findLib {
IFS="
while [ "$foundPath" == "" ] ; do
    file=$1
    for fold in $searchPath
    do
        echo "Search $fold/$file" 1>&2
        if [ -f "$fold/$file" ]
        then
            echo $fold/$file
            return 0
        fi
    done
    shift
    if [ $# == "0" ] ; then
        return 1
    fi
done
}

# Definition de variables
AIDE="FAUX"           # Si VRAI permet d'afficher le manuel -Fr-
du script

```

```

HELP="FAUX"          # Si VRAI permet d'afficher le manuel -En-
du script
BAD_ARG="FAUX"       # Si VRAI permet d'afficher une sortie avec
erreur d'argument
ESCOPE="FAUX"        # Fabrication de la librairie et édition
des liens du traducteur Escope
ETOILE="*****"
*****"

# Teste le nombre d'arguments d'entrée
if [ $# == "0" ] ; then
    # Cas du nombre d'arguments nul
    BAD_ARG="VRAI"

else
    # Fabrication des listes d'options et d'arguments
    LISTE_BAD_ARG=""
    for i in "$@" ; do
        # "--aide" ==> affichage de l'aide en Français
        # "--help" ==> affichage de l'aide en Anglais
        case $i in
            "--aide") AIDE="VRAI";break;;
            "--help") HELP="VRAI";break;;
            "-ESCOPE") ESCOPE="VRAI";;
            *) BAD_ARG="VRAI"
                LISTE_BAD_ARG="$LISTE_BAD_ARG $i";;
        esac
    done
fi

if [ "$AIDE" == "VRAI" ] || [ "$HELP" == "VRAI" ] ; then
    # Affichage de l'aide si elle est demandée par l'utilisateur
    if [ "$AIDE" == "VRAI" ] ; then
        # Affiche l'aide du script en Français
        echo ''
        echo "$ETOILE"
        echo "NOM"
        echo "    essaicast${VERSION} ==> Edition des liens"
        echo ''
        echo "VERSION"
        echo "    Version du Script : ${VERSION}.${REVISION}"
        echo ''
        echo "SYNTAXE"
        echo "    essaicast${VERSION}"
        echo ''
        echo "DESCRIPTION"
        echo "    --aide ==> Affiche le manuel de cette commande
en Français"
        echo "    --help ==> Affiche le manuel de cette commande
en Anglais"
        echo ''
        echo "EXEMPLES"
        echo "    essaicast${VERSION} "
        echo "    ==> Réalise l'édition des liens dans le
repertoire courant"
        echo "        En cas de succès: "
        echo "        la commande affiche: Edition des
liens réussis"
        echo "        ET"
        echo "        produit un executable binaire
cast_${BIT}_${VERSION}"
        echo "        dans le repertoire courant."
        echo ""
        echo "        En cas d'échec: "
        echo "        Ou bien, l' executable n'a pas été
généré => la commande affiche: "
        echo "        ECHEC durant l'édition des liens
"
        echo "        ET "
        echo "        produit un fichier d'erreur
link_cast_${BIT}_${VERSION}.log"
        echo ""
        echo "        ou alors, certaines librairies
dynamiques sont manquantes, "
        echo "        => la commande liste les liens. "
        echo ''
        echo "AUTEUR"
        echo "    Script écrit par Clément BERTHINIER"
        echo ''
        echo "VOIR AUSSI"
        echo "    Aide du Script 'castem${VERSION}' ==>
castem${VERSION} --aide"
        echo "    Aide du Script 'compilcast${VERSION}' ==>
complicast${VERSION} --aide"
        echo "$ETOILE"
        echo ''
    else
        # Affiche l'aide du script en Anglais
        echo ''
        echo "$ETOILE"
        echo "NAME"
        echo "    essaicast${VERSION} ==> Réalise link edition in
current directory"
        echo ''
        echo "VERSION"
        echo "    Script Version : ${VERSION}.${REVISION}"
        echo ''
        echo "SYNTAX"
        echo "    essaicast${VERSION}"
        echo ''
        echo "DESCRIPTION"
        echo "    --aide ==> Print the manual of this script in
French"
        echo "    --help ==> Print the manual of this script in
English"
        echo ""
    fi
fi

```

```

echo "EXAMPLES"
echo "    essaicast${VERSION} "
echo "        => Realise link edition in the current
directory"
echo "            If link edition has been successful "
echo "                the terminal prints: Edition des
liens reussis"
echo "                    AND"
echo "                        an executable binary
cast ${BIT}_${VERSION} is created"
echo "                            in the current directory"
echo ""
echo "                    If a problem occurs... "
echo "                    If the executable binary has not
been generated, the terminal"
echo "                        the terminal prints: ECHEC durant
l'Edition des liens "
echo "                            AND "
echo "                                an error file
link cast ${BIT}_${VERSION}.log is created"
echo ""
echo "                    If some dynamic libraries are
missing, the terminal lists them. "
echo "SEE ALSO"
echo "    Help items for 'castem${VERSION}'      ==>
'castem${VERSION} --help'"
echo "    Help items for 'compilcast${VERSION}' ==>
'compilcast${VERSION} --help'"
echo "${ETOILE}"
echo "
fi

else

if [ "${ESOPE}" == "VRAI" ]; then
#####
# Cas de la librairie Esope et du traducteur #
#####
LIBESOPE=libesope_${BIT}.a
ESOPE_EXEC=bin/esope_${PLATEFORME}_${BIT}_${VERSION}
ESOPE_EXEC_LOCAL=esop_${BIT}_${VERSION}
if [ -f ${ESOPE_EXEC_LOCAL} ]; then \rm -f ${ESOPE_EXEC_LOCAL}
; fi

echo
echo
#####
# Mise a jour de la librairie ${LIBESOPE}
#
echo
#####
ar -ruvs ${ESOPE REP}/lib/${LIBESOPE} *.o > /dev/null
2>&1
ar -d ${ESOPE REP}/lib/${LIBESOPE} esop10.o > /dev/null
2>&1

if [ ${PLATEFORME} == "Linux" ] ; then
OPT_DEP="-umain"
LIB_DEP="-irt"
else
OPT_DEP=""
LIB_DEP=""
fi

${FORTRAN} -v -Wall -static-libgfortran -static-libgcc -
static-libstdc++ -m${BIT} -o ${ESOPE_EXEC_LOCAL} ${OPT_DEP} *.o
${ESOPE REP}/lib/${LIBESOPE} -lpthread ${LIB_DEP} >
link_${ESOPE_EXEC_LOCAL}.log 2>&1

# Test d'existance de ${ESOPE_EXEC_LOCAL}
if [ -x ${ESOPE_EXEC_LOCAL} ]; then
\rm -f link_${ESOPE_EXEC_LOCAL}.log

# Test d'existance de ${ESOPE_EXEC}
if [ ! -x ${ESOPE REP}/bin/${ESOPE_EXEC} ]; then
echo
echo
#####
# Generation de l'executable ${ESOPE_EXEC}
#
echo "        Generation de l'executable ${ESOPE_EXEC}"
#
echo
#####
# Cas ou le binaire Standard est bien present dans le
repertoire /bin de Cast3M
echo
echo
#####
# Mise a jour de l'executable ${ESOPE_EXEC}
#
echo "        Mise a jour de l'executable ${ESOPE_EXEC}"
#
echo
#####
# mv ${ESOPE REP}/bin/${ESOPE_EXEC}
${ESOPE REP}/bin/${ESOPE_EXEC}.old
fi
cp ${ESOPE_EXEC_LOCAL} ${ESOPE REP}/bin/${ESOPE_EXEC}

else
# Cas ou l'executable n'a pas ete genere
echo
echo
#####
# ECHEC durant l'Edition des liens
#
echo "        ECHEC durant l'Edition des liens"
; ;;

```

```

if [ -f link_${ESOPE_EXEC_LOCAL}.log ]; then
echo "*   Consultez le fichier"
link ${ESOPE_EXEC_LOCAL}.log
*
fi
echo
*****
fi
exit 0
fi

#####
# Cas de Cast3M #
#####

# Choix du Type de licence
if [ -d ${CASTEM REP}/licence ] || [ "$LICENCETYPE" == "INDUS" ]
&& [ ! "$LICENCETYPE" == "EDURE" ]; then
# Cas de la licence INDUSTRIELLE
LICENCETYPE="INDUS"
else
# Cas de la licence EDUCATION - RECHERCHE
LICENCETYPE="EDURE"
fi

# Definition de variables
castX="bin_Cast3M_${PLATEFORME}_${LICENCETYPE}_${BIT}_${VERSION}"
"
castX_Local="cast_${BIT}_${VERSION}"
GLUTLIB="FAUX"
GLULIB="FAUX"
GLLIB="FAUX"
X11LIB="FAUX"
XxfLIB="FAUX"

# Nettoyage d'une precedente operation d'édition des liens
if [ -f link ${castX_Local}.log ]; then \rm -f
link ${castX_Local}.log ; fi
if [ -x ${castX_Local} ] ; then \rm -f ${castX_Local}
; fi

#####
# Recherche de la librairie libglut
#####
if [ ${PLATEFORME} == "Linux" ] ; then
GLUTLIB="-lglut_${BIT}"
else
# Cas de la Plateforme MAC OSX 64-bits
GLULoc=$(findLib libglut.dylib libglut.3.dylib)
if [ $? == 0 ]; then
GLUTLIB=${GLULoc}
fi
fi

#####
# Recherche de la librairie libGLU.so dans les repertoires
#####
if [ ${PLATEFORME} == "Linux" ] ; then
GLULoc=$(findLib libGLU.so libGLU.so.1)
if [ $? == 0 ]; then
GLULIB=${GLULoc}
fi
else
# Cas de la Plateforme MAC OSX 64-bits
GLULoc=$(findLib libGL.dylib libGL.so.1 libGL.1.dylib)
if [ $? == 0 ]; then
GLULIB=${GLULoc}
fi
fi

#####
# Recherche de la librairie libGL.so dans les repertoires
#####
if [ ${PLATEFORME} == "Linux" ] ; then
GLLoc=$(findLib libGL.so libGL.so.1)
if [ $? == 0 ]; then
GLLIB=${GLLoc}
fi
else
# Cas de la Plateforme MAC OSX 64-bits
GLLoc=$(findLib libGL.dylib libGL.1.dylib)
if [ $? == 0 ]; then
GLLIB=${GLLoc}
fi
fi

#####
# Recherche de la librairie libX11 dans les repertoires
#####
if [ ${PLATEFORME} == "Linux" ] ; then
X11Loc=$(findLib libX11.so libX11.so.6)
if [ $? == 0 ]; then
X11LIB=${X11Loc}
fi
else
# Cas de la Plateforme MAC OSX 64-bits
; ;;

```



```

X11Loc=$(findLib libX11.dylib libX11.6.dylib)
if [ $? == 0 ] ; then
  X11LIB=$X11Loc
fi

#####
# Recherche de la librairie libXxf86vm dans les repertoires
#####
if [ $PLATEFORME == "Linux" ] ; then
  XxfLoc=$(findLib libXxf86vm.so libXxf86vm.so.1)
  if [ $? == 0 ] ; then
    XxfLIB=$XxfLoc
  fi
else
  # Cas de la Plateforme MAC OSX 64-bits
  XxfLoc=$(findLib libXxf86vm.dylib libXxf86vm.1.dylib)
  if [ $? == 0 ] ; then
    XxfLIB=$XxfLoc
  fi
fi

#####
# Teste la presence de fichier .o dans le repertoire courant
\ls -l *.o > /dev/null 2>&1
CODE_RETOUR=$?

if [ $CODE_RETOUR -eq 0 ] ; then
  # Compte le nombre de binaire .o dans le repertoire courant
  # ajouter a Cast3M (en ommettant main.o et bdata.o)
  nb_o=$( ls -1 *.o | wc -l )
  if [ $nb_o -gt 0 ] ; then
    # Presence du *.o
    O_BINARY='*.*'
  else
    # Absence du *.o
    O_BINARY=''
  fi
else
  # Absence du *.o
  O_BINARY=''
fi

# Test d'existence de libcastem_${LICENCETYPE}_${BIT}.a
if [ ! -f ${CASTEM REP}/lib${BIT}/libcastem_${LICENCETYPE}_${BIT}.a ] ;
then
  echo
  echo
*****"
echo "*           Generation de la librairie
libcastem_${LICENCETYPE}_${BIT}.a          *"
echo
echo
*****"
echo
ar -ruvs
${CASTEM REP}/lib${BIT}/libcastem_${LICENCETYPE}_${BIT}.a ./*.o
>/dev/null 2>&1

# Absence du *.o
O_BINARY=' '
fi

if [ $PLATEFORME == "Linux" ] ; then
  OPT_DEP="umain"
else
  OPT_DEP=""
fi

# Définition des Librairies a utiliser pour faire l'édition
des liens
if [ $PLATEFORME == "MAC" ] ; then
  LIBRARY REP="-L${MPI_LIBREP} -L${CASTEM REP}/lib${BIT}"
  STATIC_LIBRARY="-lcastem_${LICENCETYPE}_${BIT} -lesope ${BIT} -lfxdr ${BIT} -ljpeg ${BIT} -lmed ${BIT} -lhdf5 ${BIT} -lz ${BIT}"
  SHARED_LIBRARY="${GLUTLIB} ${GLULIB} ${GLLIB} ${X11LIB} ${XxfLIB} -lmpi_mpifh -lmpi -lpthread -ldl -lstdc++"
  OPT_DEP=""
else
  # if [ ${BIT} == "64" ] ; then
  #   # Architectures 64-bits
  #   LIBRARY REP="-L${MPI_LIBREP} -L${CASTEM REP}/lib${BIT} -L/usr/lib/x86_64-linux-gnu -L/usr/X11R6/lib64 -L/usr/X11/lib -L/usr/local/lib64 -L/usr/lib64"
  #   # else
  #   #   # Architectures 32-bits
  #   #   LIBRARY REP="-L${MPI_LIBREP} -L${CASTEM REP}/lib${BIT} -L/usr/X11R6/lib -L/usr/local/lib -L/usr/lib"
  #   # fi
  LIBRARY REP="-L${MPI_LIBREP} -L${CASTEM REP}/lib${BIT}"
  STATIC_LIBRARY="-lcastem_${LICENCETYPE}_${BIT} -lesope ${BIT} -lfxdr ${BIT} -ljpeg ${BIT} -lmed ${BIT} -lhdf5 ${BIT} -lz ${BIT} ${GLUTLIB} -luma ${BIT}"
  SHARED_LIBRARY="${GLULIB} ${GLLIB} ${X11LIB} ${XxfLIB} -lmpi_mpifh -lmpi -lpthread -lrt -ldl -lstdc++"
  OPT_DEP="umain -u_BLOCK DATA -fopenmp"
fi

#~ #Pour les distributions basees sur Redhat, il faut le
paquet "libtirpc-devel" d'installer sur la machine :

```

```

#~ if [ -f /etc/redhat-release ] ; then
#~   test1=$(grep Fedora /etc/redhat-release) #VOIR avec
Gauthier pour le cas de ARCH LINUX
#~ #Sur CentOS,
pas de PB d'apres Gauthier !
#~ if [ $? == 0 ] ; then
#~   #~ SHARED_LIBRARY="SSHARED_LIBRARY -ltirpc"
#~   #~ echo "Attention: Assurez-vous que le paquet libtirpc-
devel est bien installé sur votre machine"
#~   #~ fi
#~ fi

#Avec librpc_32.a (obtenu dans "/usr/lib/libtirpc.a" lors de
"urpmi libtirpc-devel") :
STATIC_LIBRARY="-lcastem_${LICENCETYPE}_${BIT} -lesope ${BIT} -lfxdr ${BIT} -ljpeg ${BIT} -lmed ${BIT} -lhdf5 ${BIT} -lz ${BIT} ${GLUTLIB} -luma ${BIT} -lrpc ${BIT}"

fi

# Edition des liens pour generer le fichier "${castX Local}"
${GFORTRAN} -v -m${BIT} -static-libgfortran -o ${castX Local}
${OPT_DEP} ${O_BINARY} ${LIBRARY REP} ${STATIC_LIBRARY}
${SHARED_LIBRARY} > link_${castX Local}.log 2>&1

# Test d'existence de ${castX Local}
if [ -x ${castX Local} ] ; then
  \rm -f link_${castX Local}.log

# Test d'existence de
bin_Cast3M ${PLATEFORME} ${LICENCETYPE} ${BIT} ${VERSION}
if [ ! -x
${CASTEM REP}/bin/bin_Cast3M_${PLATEFORME}_${LICENCETYPE}_${BIT} ${VERSION} ] ; then
  echo
  echo
*****"
echo "*           Generation de l'executable
bin_Cast3M ${PLATEFORME} ${LICENCETYPE} ${BIT} ${VERSION}  *"
echo
mv ${castX Local}
${CASTEM REP}/bin/bin_Cast3M_${PLATEFORME}_${LICENCETYPE}_${BIT} ${VERSION}

else
# Cas ou le binaire Standard est bien present dans le
repertoire /bin de Cast3M
  echo
  echo
*****"
echo "*           Edition des liens reussie
*****"
else
# Cas ou l'executable n'a pas ete genere
  echo
  echo
*****"
echo "*           ECHEC durant l'Edition des liens
*****"
if [ -f link_${castX Local}.log ] ; then
  echo "*           Consultez le fichier link_${castX Local}.log
*****"
fi
echo
echo
fi

else
# Cas ou des librairies sont manquantes pour effectuer l'édition
des liens
  echo
  echo
*****"
echo "*           Certaines librairies sont manquantes
*****"
echo "*           Some      librairies are      missing
*****"
echo
*****"
if [ "$GLUTLIB" == "FAUX" ] ; then
  echo
  echo "           Installez le paquet contenant la librairie glut
${BIT}-bit"
  fi

if [ "$GLULIB" == "FAUX" ] ; then
  echo
  echo "           Installez le paquet contenant la librairie GLU
${BIT}-bit"
  fi

if [ "$GLLIB" == "FAUX" ] ; then
  echo
  echo "           Installez le paquet contenant la librairie GL
${BIT}-bit"
  fi
fi

```

```
if [ "$X11LIB" == "FAUX" ]; then
  echo
  echo "      Installez le paquet contenant la librairie X11
${BIT}-bit"
fi

if [ "$XxfLIB" == "FAUX" ]; then
  echo
  echo "      Installez le paquet contenant la librairie
libXxf86vm ${BIT}-bit"
fi

if [ "$GFORBIN" == "FAUX" ]; then
  echo
  echo "      Vous avez besoin de gfortran (version 4.3.2 ou
superieure)"
  echo "      You need gfortran           (version 4.3.2 or
above)      "
fi

echo
*****"
echo
fi

# Nettoyage du repertoire temp
if [ -d temp ] ; then \rm -rf temp ; fi
fi
```

Annexe E. Scripts cast_UTIL19.bat et cast_UTIL19

a. Script cast_UTIL19.bat

```

@ECHO OFF
SETLOCAL ENABLEDELAYEDEXPANSION

SET SCRIPT_REP=%~dp0

REM Chargement de l'environnement (Laisser %CASTEM_VERSION%, il est remplace automatiquement par InstallJammer)
IF NOT EXIST %SCRIPT_REP:~0,-1%\environnement_Cast3M%XXANNEEXX%.bat (
    ECHO %LIGNEUP%
    ECHO u Le script environnement_Cast3M%CASTEM_VERSION%.bat n'existe pas...
    ECHO u Contactez le Support Cast3M
    ECHO %LIGNEDOWN%
    ECHO.
    EXIT /B 0
) ELSE (
    CALL %SCRIPT_REP:~0,-1%\environnement_Cast3M%XXANNEEXX%
)

REM Espaces remplacant l'annee dans les ECHO
SET SPY=""
IF "%XXANNEEXX%"==""
(
    SET SPY=" "
)
REM : Retrait des doubles cotes
SET SPY=%SPY:~1,-1%

SET LIGNEUP=oooooooooooooooooooooooooooooooooooooooooooooooooooo
SET LIGNEDOWN=oooooooooooooooooooooooooooooooooooooooooooo
SET LIGNEMIDDLE=oooooooooooooooooooooooooooooooooooooooooooo

REM ##### Procedures --> UTILPROC
REM #####
SET /A nb_proc=0
FOR %%i IN (*.procedur) DO SET /A nb_proc+=1

IF EXIST UTILPROC DEL /F UTILPROC

IF NOT %nb_proc% == 0 (
    ECHO.
    ECHO Nombre de procedures : %nb_proc%
    COPY *.procedur ZZZ_procedur.procedur>nul

    REM Lancement de Cast3M
    ECHO 'UTIL' 'PROC' 'ZZZ_procedur.procedur'; 'FIN'; | castem%CASTEM_VERSION%>nul
    DEL /F ZZZ_procedur.procedur
    IF EXIST fort.3 DEL /F fort.3
    IF EXIST fort.98 DEL /F fort.98

    ECHO %LIGNEUP%
    ECHO u Un fichier UTILPROC a ete cree
    ECHO %LIGNEDOWN%
)

REM ##### Notices --> UTILNOTI
REM #####
SET /A nb_noti=0
FOR %%i IN (*.notice) DO SET /A nb_noti+=1

IF EXIST UTILNOTI DEL /F UTILNOTI

IF NOT %nb_noti% == 0 (
    ECHO.
    ECHO Nombre de notice : %nb_noti%
    COPY *.notice ZZZ_noti.notice>nul

    REM Lancement de Cast3M
    ECHO 'UTIL' 'NOTI' 'ZZZ_noti.notice'; 'FIN'; | castem%CASTEM_VERSION%>nul
    DEL /F ZZZ_noti.notice
    IF EXIST fort.3 DEL /F fort.3
    IF EXIST fort.98 DEL /F fort.98

    ECHO %LIGNEUP%
    ECHO u Un fichier UTILNOTI a ete cree
    ECHO %LIGNEDOWN%
)

EXIT /B 0

```

b. Script cast_UTIL19

```
#!/bin/bash

#Recuperation des variables d'environnement
if [ "${VERSION}" == "" ] ; then VERSION=%(XXANNEEXX)
; fi
if [ "${REVISION}" == "" ] ; then REVISION=%(XXREVISIONXX)
; fi
if [ "${PLATEFORME}" == "" ] ; then PLATEFORME=%(XXPLATEFORMEXX)
; fi
if [ "${CASTEM_REP}" == "" ] ; then CASTEM_REP=%{INSTALL_PATH}
; fi
castScript="castem${VERSION}"

# Nettoyage preliminaire
if [ -f UTILNOTI ] ; then rm -f UTILNOTI; fi
if [ -f UTILPROC ] ; then rm -f UTILPROC; fi
if [ -f ZZZ_Procedures_ERREURS.log ] ; then rm -f
ZZZ_Procedures_ERREURS.log ; fi
if [ -f ZZZ_Notices_ERREURS.log ] ; then rm -f
ZZZ_Notices_ERREURS.log ; fi
ETOILE=""

#####
## PROCEDURES --> UTILPROC
#####
nb.proc=`( ls -1 | fgrep '.procedur' | wc -l )`
if [ $nb.proc -gt 0 ] ; then

    echo "${ETOILE}"
    echo "|"
    echo "LISTE DES PROCEDURES UTILISATEUR"
    |
    echo "${ETOILE}"

    for i in *.procedur ; do
        # Affichage du nom de la procedure
        echo "| ${i}"
        #
        # Concatenation des procedures en une seule
        cat ${i} > ZZZ_proc.procedur
    done

    # Ajout des $$$ à la toute fin du fichier
    # ATTENTION : Ceci sera retire si toutes les procedures se
    terminent par $$$
    echo '$$$' >> ZZZ_proc.procedur

    # Generation du fichier .dgibi
    echo "util proc 'ZZZ_proc.procedur';" > ZZZ_proc.dgibi
    echo "fin;" >> ZZZ_proc.dgibi

    # Lancement de Cast3M
    $castScript ZZZ_proc.dgibi > ZZZ_proc.res 2>1
    rm -f ZZZ_proc.res ZZZ_proc.procedur ZZZ_proc.dgibi
    ZZZ_proc.trace

    # Generation de CAST3M.PROC si ce dernier n'existe pas
    if [ ! -f ${CASTEM_REP}/data/CAST3M.PROC ] ; then
        mv UTILPROC ${CASTEM_REP}/data/CAST3M.PROC
    fi
fi

#####
## NOTICES --> UTILNOTI
#####
nb.noti=`( ls -1 | fgrep '.notice' | wc -l )`
if [ $nb.noti -gt 0 ] ; then

    # La notices '*.notice' est supprimee : Elle est en double
    # avec etoile.notice
    # if [ -f './*.notice' ] ; then rm -f './*.notice' ; fi

    # Les notices avec des noms speciaux sont renommees elles font
    # planter les scripts
    if [ -f './SPAL.notice' ] ; then mv ./SPAL.notice'
    spal.notice
    ; fi
    if [ -f './<eg.notice' ] ; then mv ./<eg.notice'
    aaa_inf_egal.notice
    ; fi
    if [ -f './>eg.notice' ] ; then mv ./>eg.notice'
    aaa_sup_egal.notice
    ; fi
    if [ -f './<.notice' ] ; then mv ./<.notice'
    aaa_inf_strict.notice
    ; fi
    if [ -f './>.notice' ] ; then mv ./>.notice'
    aaa_sup_strict.notice
    ; fi
    if [ -f './+.notice' ] ; then mv ./+.notice'
    aaa_addition.notice
    ; fi
    if [ -f './-.notice' ] ; then mv ./-.notice'
    aaa_soustraction.notice
    ; fi
    if [ -f './etoil.notice' ] ; then mv ./etoil.notice'
    aaa_produit.notice
    ; fi
    if [ -f './*.notice' ] ; then mv ./*.notice'
    aaa_produit2.notice
    ; fi

```

```

        if [ -f './aaal.notice' ] ; then mv ./aaal.notice'
        aaa_division.notice
        ; fi
        if [ -f './**.notice' ] ; then mv ./**.notice'
        aaa_puissance.notice
        ; fi

        echo "${ETOILE}"
        echo "|"
        echo "LISTE DES NOTICES UTILISATEUR"
        |
        echo "${ETOILE}"

        for i in *.notice ; do
            # Affichage du nom de la notice
            echo "| ${i}"

            # Concatenation des notices en une seule
            cat ${i} >> ZZZ_noti.notice

            # Ajout des $$$ à la toute fin du fichier
            # ATTENTION : Ceci sera retire si toutes les notices se
            terminent par $$$
            echo '$$$' >> ZZZ_noti.notice

            # Generation du fichier .dgibi
            echo "util noti 'ZZZ_noti.notice';" > ZZZ_notice.dgibi
            echo "fin;" >> ZZZ_notice.dgibi

            # Lancement de Cast3M
            $castScript ZZZ_ntcice.dgibi > ZZZ_noti.res 2>1
            rm -f ZZZ_noti.res ZZZ_noti.notice ZZZ_notice.dgibi
            ZZZ_notice.trace

            # Generation de CAST3M.MASTER si ce dernier n'existe pas
            if [ ! -f ${CASTEM_REP}/data/CAST3M.MASTER ] ; then
                mv UTILNOTI ${CASTEM_REP}/data/CAST3M.MASTER
            fi
        fi

        #####
        ## Gestion des ERREURS rencontres
        #####
        # Affichage des Problemes rencontres dans les procedures
        if [ -f ZZZ_Procedures_ERREURS.log ] ; then
            echo '|'
            cat ZZZ_Procedures_ERREURS.log
            echo '|'
        fi

        # Affichage des Problemes rencontres dans les notices
        if [ -f ZZZ_Notices_ERREURS.log ] ; then
            echo '|'
            cat ZZZ_Notices_ERREURS.log
            echo '|'
        fi

        # Menage dans les fichiers generes
        rm -f fort.*
```

Annexe F. Scripts castem19.bat et castem19

a. Script castem19.bat

```

@ECHO OFF
SETLOCAL ENABLEDELAYEDEXPANSION

REM Enregistrement de l'heure initiale
SET STARTDATE0=%DATE%
SET STARTTIME0=%TIME%

SET SCRIPT_REP=%~dp0

REM Chargement de l'environnement (Laisser %CASTEM_VERSION%, il
REM est remplacé automatiquement par InstallJammer)
CALL %SCRIPT_REP%~-1%\environnement_Cast3M%XXANNEEXX%

REM Définition des chemins pour les fichiers ERREUR, MASTER et
PROC
SET CASTEM_ERREUR=%CASTEM_REP%\data\GIBI.ERREUR
SET CASTEM_NOTICE=%CASTEM_REP%\data\CAST3M.MASTER
SET CASTEM_PROC=%CASTEM_REP%\data\CAST3M.PROC
SET DIRLIC=%CASTEM_REP%\licence

REM Le nom du binaire dépend du type de licence
IF EXIST "%DIRLIC%" (
    SET castX=bin_Cast3M_Win_INDUS_%BIT%_%CASTEM_VERSION%.exe
    SET CASTEM_LICENCE=%castX% Licence
INDUSTRIELLE
) ELSE (
    SET castX=bin_Cast3M_Win_DEVEL_%BIT%_%CASTEM_VERSION%.exe
    SET CASTEM_LICENCE=%castX% Licence EDUCATION -
RECHERCHE
)

SET castX_Local=cast_%BIT%_%CASTEM_VERSION%.exe

IF EXIST "%TMP%" (
    SET ESOPE_TEMP=%TMP%
) ELSE (
    SET ESOPE_TEMP=C:/tmp)
SET MIF_PATH=%CASTEM_REP%\header

REM Initialisations :
SET OPTIONS=
SET PAUSEFIN=
SET /A ERROLEV=0
SET DEBUG=
SET AIDE=
SET HELP=
SET SWAP=VRAI
SET Val=
SET Vallibre=
SET NomF=
SET NomF2=
SET ESOPE_PARAM=
SET Error_Liste=
SET MEMDEF=FAUX
SET MEMLIB=FAUX
SET ZERMEM=FAUX
SET TEST_BASE=FAUX
SET LISTE_JDD=

REM Les espaces après %USERNAME% et %CD% sont importants
SET UTILISATEUR=%USERNAME%
SET REPERTOIRE_COURANT=%CD%

REM Lecture des Arguments d'entrée
:DEBUT_LECTURE
SET ArgNAME1=%~1
SET ArgNAME2=%2

REM Affichage de l'aide en Français
IF "%ArgNAME1%"=="-aide" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)
IF "%ArgNAME1%"=="-aide" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)
IF "%ArgNAME1%"=="aide" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)
IF "%ArgNAME1%"=="/?" (
    SET AIDE=VRAI
    GOTO LABEL_AIDE
)

REM Affichage de l'aide en Anglais
IF "%ArgNAME1%"=="--help" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)
IF "%ArgNAME1%"=="-help" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)
IF "%ArgNAME1%"=="help" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)

SET HELP=VRAI
GOTO LABEL_HELP
)

IF "%ArgNAME1%"=="-h" (
    SET HELP=VRAI
    GOTO LABEL_HELP
)

REM Lancement avec l'icône : pause à la fin du script pour ne
pas perdre le contenu de la fenêtre
IF "%ArgNAME1%"=="--pause" (
    SET PAUSEFIN=VRAI
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Lancement avec l'icône : pause à la fin du script pour ne
pas perdre le contenu de la fenêtre
IF "%ArgNAME1%"=="-test" (
    SET TEST_BASE=VRAI
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Recuperer dans le répertoire de Cast3M des sources,
includes, cas-tests, procédures, notices
IF "%ArgNAME1%"=="-r" (
    SHIFT
    GOTO RECUPERATION_FICHIER
)

REM Activation du mode DEBUG
IF "%ArgNAME1%"=="-d" (
    IF EXIST "%MINGWPATH%\gdb.exe" (
        SET DEBUG=%MINGWPATH%\gdb.exe
        SET OPTIONS=%OPTIONS% %ArgNAME1%
        SHIFT
        GOTO DEBUT_LECTURE
    ) ELSE (
        ECHO.
        ECHO Vous devez installer la version DEVELOPPEUR de Cast3M
pour utiliser
        ECHO cette option
        GOTO mess_fin)
        SHIFT
        GOTO DEBUT_LECTURE
    )

REM Désactivation du débordement Mémoire
IF "%ArgNAME1%"=="-NOSWAP" (
    SET SWAP=FAUX
    SET OPTIONS=%OPTIONS% %ArgNAME1%
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Réglage de la mémoire réservée par Cast3M
IF "%ArgNAME1%"=="-MEM" (
    IF "%ArgNAME2%"==""
        ECHO.
        ECHO Il manque la quantité de mémoire à réservé après
l'option -MEM
        EXIT /B 1
    ) ELSE (
        SET Val=%ArgNAME2%
        SET MEMDEF=VRAI
        SET OPTIONS=%OPTIONS% %ArgNAME1% %ArgNAME2%
        SHIFT
        SHIFT
        GOTO DEBUT_LECTURE
    )

REM Réglage de la mémoire laissée libre au système
IF "%ArgNAME1%"=="-LIBRE" (
    IF "%ArgNAME2%"==""
        ECHO.
        ECHO Il manque la quantité de mémoire à laisser libre après
l'option -LIBRE
        EXIT /B 1
    ) ELSE (
        SET Vallibre=%ArgNAME2%
        SET MEMLIB=VRAI
        SET OPTIONS=%OPTIONS% %ArgNAME1% %ArgNAME2%
        SHIFT
        SHIFT
        GOTO DEBUT_LECTURE
    )

REM Réglage du nombre de CPU réservés par Cast3M (Par défaut la
totalité des CPU disponibles sont pris)
IF "%ArgNAME1%"=="-NCPU" (
    IF "%ArgNAME2%"==""
        ECHO.
        ECHO Il manque le nombre de CPU souhaités après l'option -
NCPU
        EXIT /B 2
    ) ELSE (

```

```

SET /A CASTEM_NCPU=%ArgNAME2% 2>nul
IF NOT "!CASTEM_NCPU!"=="%ArgNAME2%" (
    ECHO.
    ECHO Le nombre de CPU doit etre un ENTIER
    EXIT /B 3
) IF !CASTEM_NCPU! LSS 1 (
    ECHO.
    ECHO Le nombre de CPU doit etre un ENTIER superieur ou
egal a 1
    EXIT /B 4
)
SET OPTIONS=%OPTIONS% %ArgNAME1% %ArgNAME2%
SHIFT
SHIFT
GOTO DEBUT_LECTURE
)

REM remise a zero physique de la MEMOIRE
IF "%ArgNAME1%"=="-ZERMEM" (
    SET ZERMEM=VRAI
    SET OPTIONS=%OPTIONS% %ArgNAME1%
    SHIFT
    GOTO DEBUT_LECTURE
)
REM Faire UTILPROC et UTILNOTI
IF "%ArgNAME1%"=="-u" (
    CALL "%CASTEM_REP%\bin\cast_UTIL%CASTEM_VERSION%"
    SHIFT
    GOTO DEBUT_LECTURE
)
REM Si aucun cas precedent n'a ete rencontre c'est un jeux de
donnees
SET LISTE_JDD=%LISTE_JDD% %ArgNAME1%

REM Nom complet donne en argument
SET CASTEM_PROJET=%ArgNAME1%
REM Nom sans extension donne en argument
SET NomF2=%~n1
REM Repertoire du jeu de donnees (Drive Letter)\Chemin sans Drive
Letter)
SET NomF3=%~d1%~p1

REM Quitter la boucle des arguments lorsqu'on en lit plus
IF "%ArgNAME2%" == "" GOTO FIN_LECTURE
SHIFT
GOTO DEBUT_LECTURE

REM Fin de la lecture des arguments
:FIN_LECTURE

REM ****
REM ****
REM ***** AIDE
IF DEFINED AIDE (
REM Affiche l'aide en Francais
    ECHO.
    ECHO %LIGNEUP%
    ECHO NOM
    ECHO castem%CASTEM_VERSION%%SPY% : Logiciel de calcul par
Element Finis
    ECHO Site web : http://www-cast3m.cea.fr/
    ECHO
    ECHO %LIGNEMIDDLE%
    ECHO VERSION
    ECHO Version du Script :
%CASTEM_VERSION%%SPY%.%CASTEM_REVISION%
    ECHO
    ECHO %LIGNEMIDDLE%
    ECHO SYNTAXE
    ECHO castem%CASTEM_VERSION%%SPY% [OPTION]...
[LISTE_FICHIERS]...
    ECHO
    ECHO %LIGNEMIDDLE%
    ECHO DESCRIPTION
    ECHO --aide : Affiche le manuel de cette commande en
Francais
    ECHO --help : Affiche le manuel de cette commande en
Anglais
    ECHO -test : execute la base des cas-tests de Cast3M
    ECHO -r : Recupere des fichier depuis le
repertoire de Cast3M :
    ECHO .dgibi, .procedur, .notice, .eso, .c,
.INC, .h
    ECHO -u : Construit UTILPROC et UTILNOTI
    ECHO -d : Lance Cast3M avec gdb [Version
developpeur]
    ECHO -MEM Val1: Memoire reservee par Cast3M [MOTS, Mo
ou Go]
    ECHO -Exemple en MOTS : Val1=360027352
    ECHO -Exemple en Mo : Val1=1500Mo
    ECHO -Exemple en Go : Val1=2Mo
    ECHO -LIBRE Val2: Memoire laissee libre pour le systeme
    ECHO Reglage prioritaire sur -MEM Val1
    ECHO -Exemple en MOTS : Val2=360027352
    ECHO -Exemple en Mo : Val2=1500Mo
    ECHO -Exemple en Go : Val2=2Mo
    ECHO -ZERMEM : Remise a zero physique de la memoire
    ECHO -NOSWAP : Interdiction d'utiliser le fichier de
debordement
    ECHO -NCPU Val : Nombre maximum de CPU utilises par
Cast3M
    ECHO
    ECHO %LIGNEMIDDLE%
    ECHO VARIABLES D'ENVIRONNEMENT UTILES
)

```

```

ECHO      CASTEM_VERSION      : Annee de la version de Cast3M
ECHO      CASTEM_REVISION     : Numero de la revision pour cette
annee
ECHO      CASTEM REP         : Repertoire dans lequel est
installe Cast3M
ECHO      CASTEM_PLATEFORME   : Plateforme sur laquelle est
installe Cast3M
ECHO      CASTEM_LICENCE      : Licence de Cast3M installe
ECHO
ECHO %LIGNEMIDDLE%
ECHO EXAMPLES
ECHO      castem%CASTEM_VERSION%%SPY%
ECHO      Lance Cast3M sans jeu de donnee : Mode
Interactive
ECHO
ECHO      castem%CASTEM_VERSION%%SPY% fichier
ECHO      Lance le jeu de donnee 'fichier'
ECHO
ECHO      castem%CASTEM_VERSION%%SPY% -d fichier
ECHO      Lance le jeu de donnee 'fichier' dans
l'environnement
ECHO      gdb
ECHO
ECHO      castem%CASTEM_VERSION%%SPY% -u fichier
ECHO      Construit UTILPROC et UTILNOTI avec les
fichiers .procedur
ECHO      et .notice du repertoire courant et lance le
jeu de donnee
ECHO      'fichier'
ECHO
ECHO %LIGNEMIDDLE%
ECHO AUTEUR
ECHO      Script ecrit par Clement BERTHINIER
ECHO
ECHO %LIGNEMIDDLE%
ECHO VOIR AUSSI
ECHO      Aide de 'compilcast%CASTEM_VERSION%%SPY%':
'compilcast%CASTEM_VERSION% --aide' %SPY% [Version developpeur]
ECHO      Aide de 'essaicast%CASTEM_VERSION%%SPY%':
'essaicast%CASTEM_VERSION% --aide' %SPY% [Version developpeur]
ECHO
ECHO %LIGNEDOWN%
EXIT /B !ERROLEV!
)

:LABEL_HELP
IF DEFINED HELP (
REM Affiche l'aide en Anglais
ECHO.
ECHO %LIGNEUP%
ECHO NOM
ECHO castem%CASTEM_VERSION%%SPY% : Finite Element solver
Software
ECHO Site web : http://www-cast3m.cea.fr/
ECHO
ECHO %LIGNEMIDDLE%
ECHO VERSION
ECHO Script Version :
%CASTEM_VERSION%%SPY%.%CASTEM_REVISION%
ECHO
ECHO %LIGNEMIDDLE%
ECHO SYNTAX
ECHO castem%CASTEM_VERSION%%SPY% [OPTION]...
[FILES_LIST]...
ECHO
ECHO %LIGNEMIDDLE%
ECHO DESCRIPTION
ECHO --aide : Print the manual of this script in
French
ECHO --help : Print the manual of this script in
English
ECHO -test : run the Cast3M testing files
ECHO -r : Retrieves files from the Cast3M
directory :
ECHO .dgibi, .procedur, .notice, .eso, .c,
.INC, .h
ECHO -u : Build UTILPROC and UTILNOTI
ECHO -d : Execute Cast3M with gdb [Developpeur
Version]
ECHO -MEM Val1: Memory allocated by Cast3M [MOTS, Mo or
Go]
ECHO -Example in MOTS : Val1=360027352
ECHO -Example in Mo : Val1=1500Mo
ECHO -Example in Go : Val1=2Mo
ECHO -LIBRE Val2: Free memory for the system
ECHO Overwrite -MEM Val1
ECHO -Exemple in MOTS : Val2=360027352
ECHO -Exemple in Mo : Val2=1500Mo
ECHO -Exemple in Go : Val2=2Mo
ECHO -ZERMEM : The memory is physically defined to low
state
ECHO -NOSWAP : The SWAP is not allowed
ECHO -NCPU Val : Maximum number of CPU used by Cast3M
ECHO
ECHO %LIGNEMIDDLE%
ECHO USEFUL ENVIRONNEMENT VARIABLES
ECHO      CASTEM_VERSION      : Year of the version of Cast3M
ECHO      CASTEM_REVISION     : Update number for this year
ECHO      CASTEM REP         : Cast3M Install directory
ECHO      CASTEM_PLATEFORME   : Platform on which is installed
Cast3M
ECHO      CASTEM_LICENCE      : Cast3M type of License
ECHO
ECHO %LIGNEMIDDLE%
ECHO EXAMPLES
ECHO      castem%CASTEM_VERSION%%SPY%
ECHO      Execute Cast3M without any input file :
Interactive mode

```



```

ECHO      castem%$CASTEM_VERSION%$SPY% file
ECHO      Execute Cast3M with the input file 'file'
ECHO      castem%$CASTEM_VERSION%$SPY% -d file
ECHO      Execute Cast3M with the input file 'file'
ECHO      in the gdb environment
ECHO      castem%$CASTEM_VERSION%$SPY% -u file
ECHO      Build UTILPROC and UTILPROC with the files
.procedur and
ECHO      .notice of the current directory and execute
the input file
ECHO      'file'
ECHO
ECHO %$LIGNEMIDDLE%
ECHO AUTHOR
ECHO   Script written by Clement BERTHINIER
ECHO
ECHO %$LIGNEMIDDLE%
ECHO SEE ALSO
ECHO   Manual for
'compilcast%$CASTEM_VERSION%$SPY%:' compilcast%$CASTEM_VERSION% --help%$SPY%[Developper version]
ECHO   Manual for 'essaicast%$CASTEM_VERSION%$SPY%
:essaicast%$CASTEM_VERSION% --help%$SPY%[Developper version]
ECHO
ECHO %$LIGNEDOWN%
EXIT /B !ERROLEV!
)

REM ****
***** REMARQUE : Les commandes suivantes sont pour les utilisateurs qui
***** veulent compiler le code source de Cast3M
***** et non pas utiliser la distribution pré-assemblée
***** qui est fournie par défaut.

IF %$SWAP%==VRAI (
  REM NTRK : Nombre de blocs de debordement
  REM LTRK : Taille des blocs du fichier de debordement
  SET ESOPE_PARAM=NTRK=300000,LTRK=1
)

IF %$MEMDEF%==VRAI (
  REM ESOPE : (Optionnel) Memoire reservee au lancement de
  Cast3M (On peut mettre une unite : Go, Mo)
  IF "!ESOPE_PARAM!"==""
    SET ESOPE_PARAM=ESOPE=!Val!
  ) ELSE (
    SET ESOPE_PARAM=!ESOPE_PARAM!,ESOPE=!Val!
  )
)

IF %$MEMLIB%==VRAI (
  REM LIBRE : (Optionnel) Memoire laissee libre pour le systeme
  (On peut mettre une unite : Go, Mo)
  IF "!ESOPE_PARAM!"==""
    SET ESOPE_PARAM=LIBRE=!Vallibre!
  ) ELSE (
    SET ESOPE_PARAM=!ESOPE_PARAM!,LIBRE=!Vallibre!
  )
)

IF %$ZERMEM%==VRAI (
  REM ZERMEM : OUI (remet la memoire physiquement a 0), NON
  (remise a zero virtuelle ==> Par defaut)
  IF "!ESOPE_PARAM!"==""
    SET ESOPE_PARAM=ZERMEM=OUI
  ) ELSE (
    SET ESOPE_PARAM=!ESOPE_PARAM!,ZERMEM=OUI)
)

IF %$TEST_BASE%==VRAI (
  REM Lancement de la base des Cas-Tests
  IF "%CD%"=="%$CASTEM_REP%" (
    ECHO Impossible de lancer la base des cas-tests dans le
    repertoire d'installation
    SET /A ERROLEV=20
    EXIT /B !ERROLEV!
  )

  REM Preparation des repertoires de travail
  IF EXIST dgibi (
    DEL /S /Q dgibi\*>nul
  )
  IF EXIST divers (
    DEL /S /Q divers\*>nul
  )

  CALL XCOPY /S /I "%$CASTEM_REP%\dgibi" dgibi >nul
  CALL XCOPY /S /I "%$CASTEM_REP%\divers" divers>nul

  IF EXIST %$castX_Local% CALL XCOPY /Y %$castX_Local% dgibi>nul
  IF EXIST UTILPROC CALL XCOPY /Y UTILPROC dgibi>nul
  IF EXIST UTILNOTI CALL XCOPY /Y UTILNOTI dgibi>nul
  CD dgibi

  SET LISTE_JDD=*.dgibi
  SET OPTIONS=-MEM 1500Mo -NOSWAP -ZERMEM
  SET /A dgibicompt=0
  FOR %%i IN (%$LISTE_JDD%) DO SET /A dgibicompt+=1
  IF !dgibicompt! GTR 1 GOTO LISTE_DGIBI
)

REM ****
***** REMARQUE : Les commandes suivantes sont pour les utilisateurs qui
***** veulent compiler le code source de Cast3M
***** et non pas utiliser la distribution pré-assemblée
***** qui est fournie par défaut.

REM interpretation des caracteres d'échappement le cas échéant
SET /A dgibicompt=0

FOR %%i IN (%$LISTE_JDD%) DO SET /A dgibicompt+=1

REM Si on a plusieurs fichiers dans la ligne de commande, ils
sont exécutés automatiquement comme la base des cas-tests
IF %$dgibicompt% GTR 1 GOTO LISTE_DGIBI

SET NomF3text=%$NomF3%

SET NomF2text=%$CASTEM_PROJECT%

ECHO.
ECHO %$LIGNEUP%
ECHO %$CASTEM_LICENCE%
ECHO %$LIGNEMIDDLE%

IF EXIST %$castX_Local% (
  REM Execution de l'executable cast local
  ECHO EXECUTION de %$castX_Local%$SPY% LOCAL
  SET CASTEXEC=%$castX_Local%
) ELSE (
  REM Execution de l'executable Cast3M d'origine
  ECHO EXECUTION de %$castX%$SPY% ORIGINAL
  ECHO Realisation
%$CASTEM_VERSION%$SPY%.0.%$CASTEM_REVISION%
  SET CASTEXEC=%$CASTEM_REP%\bin\%$castX%")
  REM retrait des doubles cotes
  SET CASTEXEC=%$CASTEXEC:~1,-1%

ECHO %$LIGNEMIDDLE%
ECHO UTILISATEUR : %$UTILISATEUR:~0,54%
ECHO REPERTOIRE EXEC: %$REPERTOIRE_COURANT:~0,54%

REM pour MFRONT
IF NOT "%$NomF3%"==""
  REM Ajoute le repertoire du Jeu de Donnees\src au PATH
  SET PATH=%$NomF3%\src;!PATH!
) ELSE (
  REM Ajoute le repertoire courant\src au PATH
  SET PATH=%$CD%\src;!PATH!)

IF NOT "%$CASTEM_PROJECT%"==""
  REM Cas ou un nom de fichier est donne
  TITLE Cast3M 20%$CASTEM_VERSION% - %$BIT%bits : %$CASTEM_PROJECT%
  ECHO REPERTOIRE JEU : %$NomF3text:~0,54%
  ECHO NOM FICHIER : %$NomF2text:~0,54%
  IF EXIST %$CASTEM_PROJECT% (
    SET /A size=0
    CALL :SIZE_FILE %$CASTEM_PROJECT%
    REM Les espaces apres octets sont importants
    SET size!=size! octets
    ECHO TAILLE FICHIER : !size:~0,54!
  )
) ELSE (
  REM Cas ou aucun nom de fichier n'est donne
  TITLE Cast3M 20%$CASTEM_VERSION% - %$BIT%bits)

ECHO DATE DEBUT : %$STARTDATE0%
ECHO HEURE DEBUT : %$STARTTIME0%

ECHO %$LIGNEMIDDLE%
TYPE "%$CASTEM_REP%\bin\LOGO_ASCII_%$CASTEM_VERSION%.txt"
ECHO %$LIGNEDOWN%

CALL %$DEBUG% "%$CASTEXEC%"
SET /A ERROLEV=%$ERRORLEVEL%

REM Suppression des fichiers UTILPROC, UTILNOTI, .dgibi, .ps
s'ils sont de taille nulle
SET /A size=-1
IF EXIST UTILPROC (
  CALL :SIZE_FILE UTILPROC
)
IF $size==0 DEL UTILPROC>nul 2>&1

SET /A size=-2
IF EXIST UTILNOTI (
  CALL :SIZE_FILE UTILNOTI
)
IF $size==0 DEL UTILNOTI>nul 2>&1

SET /A size=-3
IF EXIST "%$NomF2%.ps" (
  CALL :SIZE_FILE %$NomF2%.ps
)
IF $size==0 DEL "%$NomF2%.ps">nul 2>&1

SET /A size=-4
IF EXIST "%$CASTEM_PROJECT%" (
  CALL :SIZE_FILE %$CASTEM_PROJECT%
)
IF $size==0 DEL "%$CASTEM_PROJECT%">nul 2>&1

SET /A size=-5
IF EXIST fort.3 (
  CALL :SIZE_FILE fort.3
)
IF $size==0 DEL fort.3>nul 2>&1

SET /A size=-6
IF EXIST fort.24 (
  CALL :SIZE_FILE fort.24
)
IF $size==0 DEL fort.24>nul 2>&1

REM pause effectuée si l'option --pause a été renseignée
IF DEFINED PAUSEFIN (pause)
GOTO MESS_FIN

```

```

REM
*****
REM Lancement d'une liste de jeux de donnees
:LISTE_DGIBI
ECHO.

REM Lancement de la liste des Cas-Tests
SET /A dgibinum=0
SET /A dgibieerr=0
FOR %%i IN (%LISTE_JDD%) DO (
    SET NomF=%%i
    SET NomF2=%NomF:-0,-6!
    ECHO !NomF!
    ECHO 'FIN'; | castem%CASTEM_VERSION% !OPTIONS! !NomF! >
    !NomF2!.err 2>&1

    REM Verifie la presence de la chaine de caractere "ARRET DU
    PROGRAMME CAST3M NIVEAU D'ERREUR: 0" : Si elle est absente, le
    cas-tests a echoue
    SET /A ERROR_CASTEM=0
    FOR /f "delims=%" %%i in ('find /C "ARRET DU PROGRAMME CAST3M
    NIVEAU D'ERREUR: 0" !NomF2!.err ^| find /C /I "!NomF2!.err:
    0"') DO SET /A ERROR_CASTEM=%i

    IF !ERROR_CASTEM==0 (
        SET /A dgibinum+=1
        MOVE /Y !NomF2!.err !NomF2!.res>nul 2>&1
    ) ELSE (
        SET /A dgibieerr+=1
        REM ATTENTION : les espaces dans NomF3 sont importants pour
        la presentation
        SET NomF3=!NomF!
        ECHO !NomF3:~0,7!>>ZZZ_ERROR.log
        IF !dgibinum! GTR 0 ECHO Cas-tests reussis :
        !dgibinum! /!dgibicompt!
        IF !dgibieerr! GTR 0 ECHO Cas-tests echoues :
        !dgibieerr! /!dgibicompt!
        ECHO.
    )
)

REM Affichage final apres l'execution
ECHO %LIGNEUP%
IF !dgibieerr==0 (
    ECHO LES CAS-TESTS ONT ETE EXECUTES AVEC SUCCES
) ELSE (
    SET /A ERROLEV=24
    SET NBR_ERROR=!dgibieerr!
    ECHO LES !NBR_ERROR:~0,4! CAS-TESTS SUIVANTS
    ONT ECHOUES
    ECHO %LIGNEMIDDLE%
    TYPE ZZZ_ERROR.log
    ECHO %LIGNEMIDDLE%
    ECHO Consultez les fichiers .err correspondants
)

ECHO %LIGNEDOWN%
:MESS_FIN
REM Affichage final : Heures, durees
SET ENDDATE0=%DATE%
SET ENDTIME0=%TIME%

REM convert STARTTIME0 and ENDTIME0 to centiseconds
IF "%STARTTIME0:~0,1%"=="0" SET /A
STARTTIME=%STARTTIME0:~1,1%*360000
IF NOT "%STARTTIME0:~0,1%"=="0" SET /A
STARTTIME=%STARTTIME0:~0,2%*360000
IF "%STARTTIME0:~3,1%"=="0" SET /A STARTTIME=%STARTTIME% +
%STARTTIME0:~-4,1%*6000
IF NOT "%STARTTIME0:~3,1%"=="0" SET /A STARTTIME=%STARTTIME% +
%STARTTIME0:~3,2%*6000
IF "%STARTTIME0:~6,1%"=="0" SET /A STARTTIME=%STARTTIME% +
%STARTTIME0:~-7,1%*100
IF NOT "%STARTTIME0:~6,1%"=="0" SET /A STARTTIME=%STARTTIME% +
%STARTTIME0:~-6,2%*100
IF "%STARTTIME0:~9,1%"=="0" SET /A STARTTIME=%STARTTIME% +
%STARTTIME0:~10,1%
IF NOT "%STARTTIME0:~9,1%"=="0" SET /A STARTTIME=%STARTTIME% +
%STARTTIME0:~-9,2%

IF "%ENDTIME0:~0,1%"=="0" SET /A
ENDTIME=%ENDTIME0:~1,1%*360000
IF NOT "%ENDTIME0:~0,1%"=="0" SET /A
ENDTIME=%ENDTIME0:~0,2%*360000
IF "%ENDTIME0:~3,1%"=="0" SET /A ENDTIME=%ENDTIME% +
%ENDTIME0:~4,1%*6000
IF NOT "%ENDTIME0:~3,1%"=="0" SET /A ENDTIME=%ENDTIME% +
%ENDTIME0:~3,2%*6000
IF "%ENDTIME0:~6,1%"=="0" SET /A ENDTIME=%ENDTIME% +
%ENDTIME0:~-7,1%*100
IF NOT "%ENDTIME0:~6,1%"=="0" SET /A ENDTIME=%ENDTIME% +
%ENDTIME0:~6,2%*100
IF "%ENDTIME0:~9,1%"=="0" SET /A ENDTIME=%ENDTIME% +
%ENDTIME0:~10,1%
IF NOT "%ENDTIME0:~9,1%"=="0" SET /A ENDTIME=%ENDTIME% +
%ENDTIME0:~-9,2%

REM calculating the duration is easy
SET /A DURATION=%ENDTIME%-%STARTTIME%

REM we might have measured the time inbetween days
IF %ENDTIME% LSS %STARTTIME% SET /A DURATION=%STARTTIME% -
%ENDTIME%

```

```

REM now break the centiseconds down to hours, minutes, seconds
and the REMaining centiseconds
SET /A DURATIONH=%DURATION% / 360000
SET /A DURATIONM=(%DURATION% - %DURATIONH%*360000) / 6000
SET /A DURATIONS=(%DURATION% - %DURATIONH%*360000 -
%DURATIONM%*6000) / 100
SET /A DURATIONCS=(%DURATION% - %DURATIONH%*360000 -
%DURATIONM%*6000 - %DURATIONS%*100)

REM some formatting
IF %DURATIONH% LSS 10 SET DURATIONH=0%DURATIONH%
IF %DURATIONM% LSS 10 SET DURATIONM=0%DURATIONM%
IF %DURATIONS% LSS 10 SET DURATIONS=0%DURATIONS%
IF %DURATIONCS% LSS 10 SET DURATIONCS=0%DURATIONCS%

REM les espaces a la fin de DURATION_TOT sont importants
SET
DURATION_TOT=%DURATIONH%:%DURATIONM%:%DURATIONS%,%DURATIONCS%

ECHO.
ECHO %LIGNEUP%
ECHO                                     INFORMATIONS FINALES
ECHO %LIGNEMIDDLE%
ECHO DATE DEBUT : %STARTDATE0% HEURE DEBUT :
%STARTTIME0%
ECHO DATE FIN : %ENDDATE0% HEURE FIN :
%ENDTIME0%
ECHO                                     DUREE :
%DURATION_TOT:~0,20%
ECHO
ECHO Support Cast3M : http://www-
cast3m.cea.fr/index.php?page=mailsupport
ECHO Site Web Cast3M : http://www-cast3m.cea.fr/index.php
ECHO %LIGNEDOWN%

EXIT /B !ERROLEV!

REM
*****
REM * Definition des fonctions utiles dans ce script
REM
*****
:RECUPERATION_FICHIER
SET ArgNAME1=%~1
SET Ext1=%~x1

IF "%Ext1%"=="dgibi" (
    IF NOT EXIST "%CASTEM_REP%\dgibi\%ArgNAME1%" (
        SET Error_Liste=%Error_Liste% %ArgNAME1%
    ) ELSE (
        COPY "%CASTEM_REP%\dgibi\%ArgNAME1%" . >nul)
    SHIFT
    GOTO RECUPERATION_FICHIER
)

IF "%Ext1%"=="procedur" (
    IF NOT EXIST "%CASTEM_REP%\procedur\%ArgNAME1%" (
        SET Error_Liste=%Error_Liste% %ArgNAME1%
    ) ELSE (
        COPY "%CASTEM_REP%\procedur\%ArgNAME1%" . >nul)
    SHIFT
    GOTO RECUPERATION_FICHIER
)

IF "%Ext1%"=="notice" (
    IF NOT EXIST "%CASTEM_REP%\notice\%ArgNAME1%" (
        SET Error_Liste=%Error_Liste% %ArgNAME1%
    ) ELSE (
        COPY "%CASTEM_REP%\notice\%ArgNAME1%" . >nul)
    SHIFT
    GOTO RECUPERATION_FICHIER
)

IF "%Ext1%"=="eso" (
    IF NOT EXIST "%CASTEM_REP%\sources\%ArgNAME1%" (
        SET Error_Liste=%Error_Liste% %ArgNAME1%
    ) ELSE (
        COPY "%CASTEM_REP%\sources\%ArgNAME1%" . >nul)
    SHIFT
    GOTO RECUPERATION_FICHIER
)

IF "%Ext1%"=="c" (
    IF NOT EXIST "%CASTEM_REP%\sources\%ArgNAME1%" (
        SET Error_Liste=%Error_Liste% %ArgNAME1%
    ) ELSE (
        COPY "%CASTEM_REP%\sources\%ArgNAME1%" . >nul)
    SHIFT
    GOTO RECUPERATION_FICHIER
)

IF "%Ext1%"=="h" (
    IF NOT EXIST "%CASTEM_REP%\include\c\%ArgNAME1%" (
        SET Error_Liste=%Error_Liste% %ArgNAME1%
    ) ELSE (
        COPY "%CASTEM_REP%\include\c\%ArgNAME1%" . >nul)
    SHIFT
    GOTO RECUPERATION_FICHIER
)

IF "%Ext1%"=="INC" (
    IF NOT EXIST "%CASTEM_REP%\include\eso\%ArgNAME1%" (
        SET Error_Liste=%Error_Liste% %ArgNAME1%
    ) ELSE (
        COPY "%CASTEM_REP%\include\eso\%ArgNAME1%" . >nul)
    SHIFT
    GOTO RECUPERATION_FICHIER
)

```



```
COPY "%CASTEM_REPO%\include\eso\%ArgNAME1%" . >nul)
SHIFT
GOTO RECUPERATION_FICHIER
)

IF NOT "%ArgNAME1%"==""
SET Error_Liste=%Error_Liste% %ArgNAME1%
SHIFT
GOTO RECUPERATION_FICHIER
)

IF NOT "%Error_Liste%"==""
ECHO Un ou des fichiers n'ont pas été extraits :
ECHO %Error_Liste%
)
EXIT /B 0

REM
*****
*:SIZE_FILE
REM Calcul la taille d'un fichier en octets passé en argument
SET size=%~z1
EXIT /B 0
```

b. Script castem19

```

#!/bin/bash

#Recuperation des variables d'environnement
if [ "${VERSION}" == "" ] ; then VERSION=%(XXANNEEXX)
; fi
if [ "${REVISION}" == "" ] ; then REVISION=%(XXREVISIONXX)
; fi
if [ "${PLATEFORME}" == "" ] ; then PLATEFORME=%(XXPLATEFORMEXX)
; fi
if [ "${CASTEM REP}" == "" ] ; then CASTEM REP=%{INSTALL_PATH}
; fi

#Pour IzPack "castem19 -test"
which castem19 >/dev/null
if [ $? == 1 ] ; then
    export PATH=$CASTEM REP/bin:$PATH
fi

# Detection de l'architecture ou forcee si BIT est defini dans
l'environnement
if [ `uname -m` == "x86_64" ] || [ "$BIT" == "64" ] && [ !
"$BIT" == "32" ] ; then
    BIT="64"
    GNU_VER="8.2.0"
    GNU_PATH=$CASTEM REP/GCC/GCC-x86_64
    export TFEHOME=$CASTEM REP/MFRONT/mfront-x86_64
else
    BIT="32"
    GNU_VER="6.3.0"
    GNU_PATH=$CASTEM REP/GCC/GCC-i686
    export TFEHOME=$CASTEM REP/MFRONT/mfront-i686
fi

# Repertoires lies a MPI (includes)
# Variable d'environnement definissant la commande MPI :
MPI_RUNCMD
if [ "${MPI_LIBREP}" == "" ] ; then
    MPI_LIBREP=$CASTEM REP/mpi/mpis[$BIT]/lib ; fi
if [ "${MPI_EXECREP}" == "" ] ; then
    MPI_EXECREP=$CASTEM REP/mpi/mpis[$BIT]/bin ; fi
if [ "${MPI_INCREP}" == "" ] ; then
    MPI_INCREP=$CASTEM REP/mpi/mpis[$BIT]/include ; fi

# Repertoire pour les librairies necessaires (pour l'execution
ou l'édition des liens)
if [ "${PLATEFORME}" == "Linux" ] ; then
    export LD_LIBRARY_PATH=$CASTEM REP/lib$BIT:$GNU_PATH/lib:$GNU_P
ATH/lib64:$MPI_LIBREP:$TFEHOME/lib/:src:LD_LIBRARY_PATH"
    # Parfois "ld" ne trouve pas crt1.o il faut ajouter des
    repertoires a LIBRARY_PATH
    export LIBRARY_PATH="$LIBRARY_PATH:/usr/lib64:/usr/lib:/usr/lib/x86_6
4-linux-gnu:/usr/lib/i386-linux-gnu"
elif [ "${PLATEFORME}" == "MAC" ] ; then
    export DYLD_LIBRARY_PATH=$CASTEM REP/lib$BIT:$GNU_PATH/lib:$GNU_P
ATH/lib64:$MPI_LIBREP:$DYLD_LIBRARY_PATH"
fi

# Ajout du chemin vers mpiprun au PATH
if [ -x $MPI_EXECREP/mpiprun ] ; then
    export PATH=$MPI_EXECREP:$PATH
fi

# Choix du Type de licence
if [ -d $CASTEM REP/licence ] || [ "$LICENCETYPE" == "INDUS" ]
&& [ ! "$SLICENCETYPE" == "EDURE" ] ; then
    # Cas de la licence INDUSTRIELLE
    LICENCETYPE="INDUS"
else
    # Cas de la licence EDUCATION - RECHERCHE
    LICENCETYPE="EDURE"
fi

# Definition de variables
castX="bin_Cast3M_${PLATEFORME}_${LICENCETYPE}_${BIT}_${VERSION}"
castX_Local="cast_${BIT} ${VERSION}"
AIDE="FAUX"          # Si VRAI permet d'afficher le manuel -Fr-
du script
HELP="FAUX"          # Si VRAI permet d'afficher le manuel -En-
du script
BAD_ARG="FAUX"        # Si VRAI permet d'afficher une sortie avec
erreur d'argument
LISTE_BAD_ARG=""      # Contient la liste des arguments qui ne
sont pas corrects
NO_ARG="FAUX"          # Si VRAI signifie que le script est lance
sans arguments
NEW_FICHIER="FAUX"      # Si VRAI le fichier d'entree est cree et
contiendra les commandes tapées en interactif
C_UTIL="FAUX"          # Si VRAI permet de faire UTILPROC et
UTILNOTI
DEBUG_MODE="FAUX"        # Si VRAI lance Cast3M en mode debug
VALGRIND_MODE="FAUX"      # Si VRAI lance Cast3M dans l'environnement
Valgrind
GET_FICHIER="FAUX"      # Si VRAI signifie que l'on souhaite
recuperer une liste de dgibi, sources, procedures ou une notices
MEMDEF="FAUX"          # Si VRAI signifie que l'on bride la
memoire disponible
VALMEM=0

```

```

CPUDEF="FAUX"          # Si VRAI signifie que l'on bride le nombre
de CPUs
VALCPU=0
TEST="FAUX"          # Si VRAI permet de lancer l'ensemble de la
base de donnees
ETOILE=""

# Definition de quelques limites
ulimit -s 2048
ulimit -c 0

export DIRLIC="$CASTEM REP/licence"
export ESOPE_PARAM="NTRK=1,LTRK=1" # NTRK=: Nombre de blocs de
debordement
                                         # LTRK=: Taille des blocs du
fichier de debordement
                                         # ESOPE=:(Optionnel) Memoire
Virtuelle reservee au lancement de Cast3M en MOTS (1 MOT = 4
octets en 32-bits et 8 octets en 64-bits)
export ESOPE_TEMP="/tmp"
export MIF_PATH="$CASTEM REP/header"

export CASTEM_ERREUR="$CASTEM REP/data/GIBI.ERREUR"
export CASTEM_NOTICE="$CASTEM REP/data/CAST3M.MASTER"
export CASTEM_PROC="$CASTEM REP/data/CAST3M.PROC"

# Teste le nombre d'arguments d'entree
COMPT_DGI_1=0
if [ $# == "0" ] ; then
    # Cas du nombre d'arguments nul
    NO_ARG="VRAI"
else
    # Boucle sur les arguments
    LISTE_DGI_1=""
    LISTE_PRO_1=""
    LISTE_NOT_1=""
    LISTE_ESO_1=""
    LISTE_NUM="^[-0-9]+\$"
    SORT_WHI="FAUX"

    while [ "$SORT_WHI" == "FAUX" ] ; do
        arg_num1=$1
        arg_num2=$2
        SORT_WHI="TRUE"
        if [ "$arg_num1" == "--aide" ] ; then
            AIDE="VRAI";
        elif [ "$arg_num1" == "--help" ] ; then
            HELP="VRAI"
        elif [ "$arg_num1" == "-u" ] ; then
            C_UTIL="VRAI"
            SORT_WHI="FAUX"
            shift
        elif [ "$arg_num1" == "-d" ] ; then
            DEBUG_MODE="VRAI"
            SORT_WHI="FAUX"
            shift
        elif [ "$arg_num1" == "-V" ] ; then
            VALGRIND_MODE="VRAI"
            SORT_WHI="FAUX"
            shift
        elif [ "$arg_num1" == "-r" ] ; then
            GET_FICHIER="VRAI"
            SORT_WHI="FAUX"
            shift
        elif [ "$arg_num1" == "-test" ] ; then
            TEST="VRAI"
            SORT_WHI="FAUX"
            shift
        elif [ "$arg_num1" == "-MEM" ] ; then
            MEMDEF="VRAI"
            VALMEM=$arg_num2
            SORT_WHI="FAUX"
            shift
            shift
        elif [ "$arg_num1" == "-NCPU" ] ; then
            CPUDEF="VRAI"
            if [ [ $arg_num2 == ${LISTE_NUM} ] ] && [ $arg_num2 -gt
0 ] ; then
                VALCPU=$arg_num2
                SORT_WHI="FAUX"
                shift
                shift
            else
                echo 'Le nombre de CPU doit etre un entier superieur ou
egal a 1'
                exit 2
            fi
        fi
    done
fi

```



```

else
if [ "${arg_num1}" == "-* " ]; then
BAD_ARG="VRAI"
LISTE_BAD_ARG="${LISTE_BAD_ARG} ${arg_num1}"
SORT_WHI="FAUX"
shift

else
if [[ ${arg_num1} != "" ]] && [ "${arg_num1}" ==
`dirname ${arg_num1}`/basename ${arg_num1}.dgibi`.dgibi" ];
then
LISTE_DGI_1="${LISTE_DGI_1} ${arg_num1}"
COMPT_DGI_1=$((COMPT_DGI_1+1))
SORT_WHI="FAUX"
shift

elif [[ ${arg_num1} != "" ]] && [ "${arg_num1}" ==
`basename ${arg_num1}.dgibi`.dgibi" ]; then
LISTE_DGI_1="${LISTE_DGI_1} ${arg_num1}"
COMPT_DGI_1=$((COMPT_DGI_1+1))
SORT_WHI="FAUX"
shift

elif [[ ${arg_num1} != "" ]] && [ "${arg_num1}" ==
`basename ${arg_num1}.procedur`.procedur" ]; then
LISTE_PRO_1="${LISTE_PRO_1} ${arg_num1}"
SORT_WHI="FAUX"
shift

elif [[ ${arg_num1} != "" ]] && [ "${arg_num1}" ==
`basename ${arg_num1}.notice`.notice" ]; then
LISTE_NOT_1="${LISTE_NOT_1} ${arg_num1}"
SORT_WHI="FAUX"
shift

elif [[ ${arg_num1} != "" ]] && [ "${arg_num1}" ==
`basename ${arg_num1}.eso`.eso" ]; then
LISTE_ESO_1="${LISTE_ESO_1} ${arg_num1}"
SORT_WHI="FAUX"
shift

elif [[ ${arg_num1} != "" ]] && [ "${arg_num1}" ==
`basename ${arg_num1}.INC`.INC" ]; then
LISTE_INC_1="${LISTE_INC_1} ${arg_num1}"
SORT_WHI="FAUX"
shift

else
if [[ ${arg_num1} != "" ]]; then
BAD_ARG="VRAI"
LISTE_BAD_ARG="${LISTE_BAD_ARG} ${arg_num1}"
SORT_WHI="FAUX"
shift
fi
fi
fi
done
fi

# Limite de la mémoire
if [ "${MEMDEF}" == "VRAI" ]; then
export ESOPE_PARAM="${ESOPE_PARAM},ESOPE=$({VALMEM}"
fi

# Limite de l'usage de CPU
if [ "${CPUDEF}" == "VRAI" ] && [ "${VALCPU}" -gt "0" ]; then
export CASTEM_NCPU=${VALCPU}
fi

# Les différentes possibilités sont passées en revue
if [ "${BAD_ARG}" == "FAUX" ]; then
# Cas où les arguments sont bien des options existantes et/ou
des fichiers avec la bonne extension
if [ "${C UTIL}" == "VRAI" ]; then
# Cas où l'on souhaite construire UTILPROC & UTILNOTI
cast_UTIL${VERSION}
if [ -f "ZZZ_noti.trace" ]; then rm -f ZZZ_noti.trace
; fi
if [ -f "ZZZ_procedur.trace" ]; then rm -f
ZZZ_procedur.trace ; fi
fi

if [ "${TEST}" == "VRAI" ]; then
# Lancement de la base des Cas-Tests si on passe ici
echo ''
if [ "${CASTEM_REP}" == "`pwd`" ]; then
echo " Impossible de lancer la base des cas-tests dans le
repertoire d'installation"
echo ''
else
# Préparation des répertoires de travail
if [ -d dgibi ]; then rm -rf dgibi ; fi
if [ -d divers ]; then rm -rf divers ; fi
mkdir dgibi divers
cp -r ${CASTEM_REP}/dgibi .
cp -r ${CASTEM_REP}/divers .

if [ -f ${castX_local} ]; then cp ${castX_local} dgibi;fi
if [ -f UTILNOTI ]; then cp UTILNOTI dgibi ;fi
if [ -f UTILPROC ]; then cp UTILPROC dgibi ;fi
cd dgibi

```

```

# Teste la présence de fichier .dgibi dans le répertoire
courant
\ls -l *.dgibi > /dev/null 2>&1
CODE_RETOUR=$?

if [ ${CODE_RETOUR} -eq 0 ]; then
echo "|"
# Nombre de fichier .dgibi
dgibicompt=`(ls -l *.dgibi | wc -l)`

dgibinum=0
dgibierr=0
for i in * .dgibi ; do
# Lancement des cas tests
NOM_RES=basename $i .dgibi.res
NOM_ERR=basename $i .dgibi.err
echo "| $i"
echo "FIN;" | castem$VERSION $i > ${NOM_RES} 2>&1

# Vérifie la présence de la chaîne de caractère "ARRET
DU PROGRAMME CAST3M NIVEAU D'ERREUR: 0" : Si elle est absente,
les cas-tests échouent
ERROR_CASTEM=`fgrep -c "ARRET DU PROGRAMME CAST3M
NIVEAU D'ERREUR: 0" ${NOM_RES}`
if [ ${ERROR_CASTEM} = "1" ]; then
dgibinum=$((dgibinum + 1))
else
dgibierr=$((dgibierr + 1))
mv ${NOM_RES} ${NOM_ERR}
echo "$i" >> ZZZ_ERROR.log
fi

echo "| Cas-tests réussis :"
${dgibinum}"/"${dgibicompt}
if [ ${dgibierr} -gt 0 ]; then echo "| Cas-tests
échoués : ${dgibierr}"/"${dgibicompt};fi
echo "|"
done

# Affichage final après l'exécution
echo "${ETOILE}"
if [ ${dgibierr} -eq 0 ]; then
echo "| LES CAS-TESTS ONT ETE EXECUTÉS
AVEC SUCCÈS"
else
echo "| LES ${dgibierr} CAS-TESTS
SUIVANTS ONT ÉCHOUÉS"
echo "${ETOILE}"
cat ZZZ_ERROR.log
echo "${ETOILE}"
echo "| Consulter les fichiers .err correspondants"
fi
echo "${ETOILE}"
fi
cd ..

elif [ "${AIDE}" == "VRAI" ]; then
# Affiche l'aide du script en Français
echo ''
echo "${ETOILE}"
echo "NOM"
echo " castem${VERSION} : Logiciel de calcul par Éléments
Finis"
echo " Site web : http://www-cast3m.cea.fr/"
echo ''
echo "VERSION"
echo " Version du Script : ${VERSION}.${REVISION}"
echo ''
echo "SYNTAXE"
echo " castem${VERSION} [OPTION]... [LISTE_FICHIERS]..."
echo ''
echo "DESCRIPTION"
echo " --aide : Affiche le manuel de cette commande en
Français"
echo " --help : Affiche le manuel de cette commande en
Anglais"
echo " -r : Copie une liste de dgibi,
procedure, notice, source"
echo " et inclut fournis avec Cast3M"
echo " -test : Exécute la base des cas-tests de
Cast3M"
echo " -u : Construit UTILPROC et UTILNOTI"
echo " -d : Lance Cast3M avec gdb (si gdb
est installé)"
echo " non compatible avec '-V'"
echo " -V : Lance Cast3M avec valgrind (si
valgrind est installé)"
echo " non compatible avec '-d'"
echo " -MEM Val: Mémoire réservée par Cast3M en
MegaOctets"
echo " -NCPU Val: Nombre maximum de CPU utilisées par
Cast3M"
echo ''
echo "EXEMPLES"
echo " castem${VERSION} -r 'p*.procedur' 'm?in.eso'
'*1.notice' 'elas*.dgibi"'
echo " Recupère la liste de fichiers : les
expressions régulières"
echo " sont interprétées"
echo ''
echo " castem${VERSION}"
echo " Lance Cast3M sans jeu de données : Mode
Interactif"
echo ''
echo " castem${VERSION} fichier.dgibi"

```

```

echo " Lance le jeu de donnee 'fichier.dgibi' s'il existe"
echo " Si 'fichier.dgibi' n'existe pas il sera cree"
echo ''
echo " castem${VERSION} -u fichier.dgibi"
echo " Construit UTILPROC & UTILNOTI et execute le jeu de donnees 'fichier.dgibi'"
echo ''
echo " castem${VERSION} [-d | -V ] fichier.dgibi"
echo " Lance le jeu de donnee 'fichier.dgibi' dans l'environnement "
echo " gdb ou valgrind"
echo ''
echo " castem${VERSION} -u [-d | -V ] fichier.dgibi"
echo " Construit UTILPROC & UTILNOTI et lance le jeu de donnees"
echo " 'fichier.dgibi' dans l'environnement gdb ou valgrind"
echo ''
echo " castem${VERSION} -test"
echo " Lance la base des cas tests de Cast3M"
echo ''
echo " AUTEUR"
echo " Script ecrit par Clement BERTHINIER"
echo ''
echo " VOIR AUSSI"
echo " Aide du Script 'compilcast${VERSION}' : 'complicast${VERSION} --aide'"
echo " Aide du Script 'essaicast${VERSION}' : 'essaicast${VERSION} --aide'"
echo " (Version developpeur de Cast3M seulement)"
echo "${ETOILE}"
echo ''

elif [ "${HELP}" == "VRAI" ]; then
# Affiche l'aide du script en Anglais
echo ''
echo "${ETOILE}"
echo "NAME"
echo " castem${VERSION}: Finite Element solver Software"
echo " Web site: http://www-cast3m.cea.fr/"
echo ''
echo "VERSION"
echo " Script Version : ${VERSION}.${REVISION}"
echo ''
echo "SYNTAX"
echo " castem${VERSION} [OPTION]... [FILE_LIST]..."
echo ''
echo "DESCRIPTION"
echo " --aide : Print the manual of this script in French"
echo " --help : Print the manual of this script in English"
echo " -r : Copy a list of dgibi, procedure, notice, source"
echo " and include distributed with Cast3M"
echo " -test : run the Cast3M testing files"
echo " -u : Build UTILPROC and UTILNOTI :
compatible with '-d' et '-V'"
echo " -d : Execute Cast3M with gdb (if gdb is installed)
not compatible with '-V'"
echo " -V : Execute Cast3M with valgrind (si valgrind is installed)
not compatible with '-d'"
echo " -MEM val: Allocated memory by Cast3M in MegaOctets"
echo " -NCPU Val: Maximum number of CPU used by Cast3M"
echo ''
echo "EXAMPLES"
echo " castem${VERSION} -r 'p*.procedur' 'm?in.eso'
'*1.notice' 'elas*.dgibi'
echo " : Copy in your directory the file list :
regular expressions "
echo " are interpreted"
echo ''
echo " castem${VERSION}"
echo " : Execute Cast3M without any input file :
Interactive mode"
echo ''
echo " castem${VERSION} fichier.dgibi"
echo " : Execute Cast3M with the input file 'fichier.dgibi' if it exists"
echo " If 'fichier.dgibi' doesn't exist it will be created"
echo ''
echo " castem${VERSION} -u fichier.dgibi"
echo " : Build UTILPROC & UTILNOTI and execute the input file 'fichier.dgibi'"
echo ''
echo " castem${VERSION} [-d | -V ] fichier.dgibi"
echo " : Execute Cast3M with the input file 'fichier.dgibi' in the gdb or valgrind environment"
echo ''
echo " castem${VERSION} -u [-d | -V ] fichier.dgibi"
echo " : Build UTILPROC & UTILNOTI and execute the input file "
echo " 'fichier.dgibi' with gdb or valgrind"
echo ''
echo " castem${VERSION} -test"
echo " runs all test cases"
echo ''
echo "AUTHOR"
echo " Script written by Clement BERTHINIER"
echo ''

```

```

echo "SEE ALSO"
echo " Manual for 'compilcast${VERSION}' :
'complicast${VERSION} --help'"
echo " Manual for 'essaicast${VERSION}' :
'essaicast${VERSION} --help'"
echo " (Developper version of Cast3M only)"
echo "${ETOILE}"
echo ''

elif [ "${GET_FICHIER}" == "VRAI" ]; then
# Cas ou on demande a recuperer des fichiers dans les repertoire d'origine
# Teste si les fichiers de la liste LISTE_DGI_1 existent
LISTE_DGI_2=""
for i in ${LISTE_DGI_1}; do
for j in ${CASTEM_REPO}/dgibi/${i}; do
# permet de developper des listes contenant des caracteres '*' et '?'
if [ ! -f ${j} ]; then
BAD_ARG="VRAI"
LISTE_BAD_ARG="${i}"
else
LISTE_DGI_2="${LISTE_DGI_2} ${j}"
fi
done
done
if [ "${LISTE_DGI_2}" != "" ]; then
cp ${LISTE_DGI_2}.
fi

# Teste si les fichiers de la liste LISTE_PRO_1 existent
LISTE_PRO_2=""
for i in ${LISTE_PRO_1}; do
for j in ${CASTEM_REPO}/procedur/${i}; do
# permet de developper des listes contenant des caracteres '*' et '?'
if [ ! -f ${j} ]; then
BAD_ARG="VRAI"
LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}"
else
LISTE_PRO_2="${LISTE_PRO_2} ${j}"
fi
done
done
if [ "${LISTE_PRO_2}" != "" ]; then
cp ${LISTE_PRO_2}.
fi

# Teste si les fichiers de la liste LISTE_NOT_1 existent
LISTE_NOT_2=""
for i in ${LISTE_NOT_1}; do
for j in ${CASTEM_REPO}/notice/${i}; do
# permet de developper des listes contenant des caracteres '*' et '?'
if [ ! -f ${j} ]; then
BAD_ARG="VRAI"
LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}"
else
LISTE_NOT_2="${LISTE_NOT_2} ${j}"
fi
done
done
if [ "${LISTE_NOT_2}" != "" ]; then
cp ${LISTE_NOT_2}.
fi

# Teste si les fichiers de la liste LISTE_ESO_1 existent
LISTE_ESO_2=""
for i in ${LISTE_ESO_1}; do
for j in ${CASTEM_REPO}/sources/${i}; do
# permet de developper des listes contenant des caracteres '*' et '?'
if [ ! -f ${j} ]; then
BAD_ARG="VRAI"
LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}"
else
LISTE_ESO_2="${LISTE_ESO_2} ${j}"
fi
done
done
if [ "${LISTE_ESO_2}" != "" ]; then
cp ${LISTE_ESO_2}.
fi

# Teste si les fichiers de la liste LISTE_INC_1 existent
LISTE_INC_2=""
for i in ${LISTE_INC_1}; do
for j in ${CASTEM_REPO}/include/eso/${i}; do
# permet de developper des listes contenant des caracteres '*' et '?'
if [ ! -f ${j} ]; then
BAD_ARG="VRAI"
LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}"
else
LISTE_INC_2="${LISTE_INC_2} ${j}"
fi
done
done
if [ "${LISTE_INC_2}" != "" ]; then
cp ${LISTE_INC_2}.
fi

else
# Cas ou on demande a lancer Cast3M
if [ ${COMPT_DGI_1} -gt 1 ] || [ "${LISTE_PRO_1}" != "" ] || [
"${LISTE_NOT_1}" != "" ] || [ "${LISTE_ESO_1}" != "" ];then

```

NOTE DE FABRICATION DE CAST3M 2019

```

# Cas ou plus d'un fichier dgibi est donne en argument
avec eventuellement des fichiers d'un autre type
BAD_ARG="VRAI"
LISTE_BAD_ARG="$@"

elif [ ${COMPT_DGI_1} == 0 ]; then
    # Cas ou il n'y a pas de fichier donne en argument

    # Suppression des fichiers issus d'une precedente
utilisation de castem${(VERSION)} sans arguments
if [ -f "Cast3M_defaut_20${(VERSION)}.dgibi" ] ; then rm -f
"Cast3M_defaut_20${(VERSION)}.dgibi" ; fi
if [ -f "Cast3M_defaut_20${(VERSION)}.ps" ] ; then rm -f
"Cast3M_defaut_20${(VERSION)}.ps" ; fi
if [ -f "Cast3M_defaut_20${(VERSION)}.trace" ] ; then rm -f
"Cast3M_defaut_20${(VERSION)}.trace" ; fi

export CASTEM_PROJET="Cast3M_defaut_20${(VERSION)}"
LISTE_DGI_1=${CASTEM_PROJET}.dgibi
touch ${LISTE_DGI_1}
# Affiche la Notice de INFO pour la version Industrielle
et ainsi le nombre de jours restants
if [ -d ${DIRLIC} ] ; then echo 'INFO INFO;' >>
${LISTE_DGI_1} ; fi

echo "${ETOILE}"
echo "|"                                INFORMATIONS SUR LE CALCUL
|
echo "${ETOILE}"
echo "| UTILISATEUR : $USER"
echo "| REPERTOIRE : `pwd`/"
echo "| DATE       : `date`"

elif [ -f ${LISTE_DGI_1} ] ; then
    # Cas ou le fichier donne en entree existe
    export CASTEM_PROJET="dirname ${LISTE_DGI_1}`/basename
${LISTE_DGI_1} .dgibi"

    if [ "$PLATEFORME" != "MAC" ];then
        TAILLE=`stat -c %s ${LISTE_DGI_1}` 2>/dev/null
    fi

    echo "${ETOILE}"
    echo "|"                                INFORMATIONS SUR LE CALCUL
|
    echo "${ETOILE}"
    echo "| UTILISATEUR : $USER"
    echo "| REPERTOIRE : `pwd`/`dirname ${LISTE_DGI_1}`"
    echo "| NOM FICHIER : basename ${LISTE_DGI_1}"
    if [ "$TAILLE" != "" ]; then
        echo "| TAILLE      : $TAILLE octets"
    fi
    echo "| DATE       : `date`"

else
    # Cas ou le fichier en entree n'existe pas
    NEW_FICHIER="VRAI"
    export CASTEM_PROJET="dirname ${LISTE_DGI_1}`/basename
${LISTE_DGI_1} .dgibi"
    touch ${CASTEM_PROJET}.dgibi
    # Affiche la Notice de INFO pour la version Industrielle
et ainsi le nombre de jours restants
    if [ -d ${DIRLIC} ] ; then echo 'INFO INFO;' >>
${CASTEM_PROJET}.dgibi ; fi

    echo "${ETOILE}"
    echo "|"                                INFORMATIONS SUR LE CALCUL
|
    echo "${ETOILE}"
    echo "| UTILISATEUR : $USER"
    echo "| REPERTOIRE : `pwd`/`dirname ${LISTE_DGI_1}`"
    echo "| NOM FICHIER : basename ${LISTE_DGI_1}"
    echo "| TAILLE      : `stat -c %s ${LISTE_DGI_1}`"
    octets"
    echo "| DATE       : `date`"
fi

if [ -x ${castX_local} ] ; then
    # Message lancement d'un cast local
    echo "${ETOILE}"
    echo "|"                                EXECUTION de ${castX_Local}
LOCAL
    echo "${ETOILE}"
    cat ${CASTEM REP}/bin/LOGO_ASCII_${VERSION}.txt
    echo "${ETOILE}"

else
    # Message lancement de Cast3M d'origine
    echo "${ETOILE}"
    if [ ${PLATEFORME} == "MAC" ] ; then
        echo "|"                                EXECUTION de ${castX} ORIGINAL
    else
        echo "|"                                EXECUTION de ${castX} ORIGINAL
    fi
    echo "|"                                Realisation
${VERSION}.0.${REVISION}
    echo "${ETOILE}"
    cat ${CASTEM REP}/bin/LOGO_ASCII_${VERSION}.txt
    echo "${ETOILE}"
fi

# Definition de l'executable a lancer
if [ -x ${castX_local} ] ; then
    EXEC_CAST=./${castX_local}
else

```

```

    EXEC_CAST=${CASTEM REP}/bin/${castX}

    if [ "${DEBUG_MODE}" == "VRAI" ];then
        # Lancement avec gdb
        OPTION_CAST=gdb
    elif [ "${VALGRIND_MODE}" == "VRAI" ];then
        # Lancement avec valgrind
        OPTION_CAST=valgrind
    fi

    if [ "$BIT" == "32" ] && [ ! "${MPI_RUNCMD}" == "" ] ; then
        # Sortie d'erreur : MPI non fonctionnel actuellement en
32-bits
        echo ''
        echo "${ETOILE}"
        echo " MPI : non fonctionnel en 32-bits / MPI : not
available on 32-bits systems"
        echo "${ETOILE}"
        echo ''
    else
        time ${CASTEM REP}/bin/rlwrap ${BIT} -f
${CASTEM REP}/bin/liste rlwrap ${VERSION} ${OPTION_CAST}
${MPI_RUNCMD} ${EXEC_CAST}
    fi

    echo ""
    echo "${ETOILE}"
    echo ""

    if [ ${COMPT_DGI_1} == 1 ]; then
        # Cas ou 1 fichier est donne en argument
        if [ -f fort.25 ] ; then mv fort.25 "${CASTEM_PROJET}.lgi"
    fi
    if [ -f fort.97 ] ; then mv fort.97 "${CASTEM_PROJET}.mif"
    fi
    if [ "${NEW_FICHIER}" == "VRAI" ];then
        if [ -f "${CASTEM_PROJET}.trace" ] ; then mv
"${CASTEM_PROJET}.trace" "${CASTEM_PROJET}.dgibi"; fi
    fi

    else
        # Cas ou aucun argument n'est donne
        if [ -f fort.25 ] ; then mv fort.25
"default_castem_${VERSION}.lgi"; fi
        if [ -f fort.97 ] ; then mv fort.97
"default_castem_${VERSION}.mif"; fi
        if [ -f "${CASTEM_PROJET}.trace" ] ; then mv
"${CASTEM_PROJET}.trace" "${CASTEM_PROJET}.dgibi" ; fi
    fi

    # Un peu de menage
    # suppression des fichiers UTILPROC et/ou UTILNOTI s'ils
sont de taille nulle
    find . -maxdepth 1 -name "UTIL*" -size 0 -exec rm -f {} ;;

    # suppression du fichier .ps s'il est de taille nulle
    find `dirname ${LISTE_DGI_1}` -maxdepth 1 -name "`basename
${LISTE_DGI_1} .dgibi.ps`" -size 0 -exec rm -f {} ;;
    fi
fi

#####
# Affichage en sortie de Script selon ce qu'il s'est passe
#####
if [ "${BAD_ARG}" == "VRAI" ] && [ "${LISTE_BAD_ARG}" != "" ] ;
then
    # Sortie d'erreur sur les arguments
    echo ''
    echo "${ETOILE}"
    echo " Arguments invalides / Unavailable arguments "
    echo " ${LISTE_BAD_ARG}"
    echo "${ETOILE}"
    echo ''
fi

```

Annexe G. Documentation Cast3M

a. Liens sur le site Cast3M

- <http://www-cast3m.cea.fr/index.php?xml=maj2011>
- <http://www-cast3m.cea.fr/index.php?xml=complements>
- <http://www-cast3m.cea.fr/index.php?xml=supportcours>

b. Documentation principale

Utiliser Cast3M

- Présentation et utilisation de castem2000 (Auteur E. Le Fichoux)
- Maillage (Auteur F. Di Paola)
- La procédure PASAPAS (Auteur T. Charras, F. Di Paola)
- Liste des modèles en mécanique non linéaire (Auteur F. Di Paola)
- Gibiane - Castem 2000 (Auteur T. Charras)
- Classification thématique des objets, opérateurs et procédures de Cast3M
- Post-traitement (Auteur F. Di Paola)

Exemples Cast3M :

- Annotated Testing Files (Auteur E. Le Fichoux)
- Exemples d'utilisation de la procédure PASAPAS (Auteur F. Di Paola)

Développer dans Cast3M :

- Développer dans Cast3M (Auteur T. Charras, J. Kichenin)

Assurance Qualité Cast3M

- Classification des cas tests de Cast3M 2019
- Note de fabrication de Cast3M 2019
- Notes de version de Cast3M 2019

c. Compléments :

- Le procedure di castem 2000 per l'analisi meccanica di strutture in materiale composito laminato (Auteur A. Miliozzi)
- Modélisation des structures de génie civil sous chargement sismique à l'aide de Castem 2000 (Auteur D. Combescure)
- Présentation des joints dilatants (Auteur P. Pegon)
- Dynamique du solide : modification du schéma de Newmark aux cas non linéaires (Auteur P. Verpeaux, T. Charras)
- Optimisation dans Cast3M (Auteur T. Charras, J. Kichenin)
- Un manuel d'utilisation de Cast3M (Auteur P. Pasquet)
- Initiation à la simulation numérique en mécanique des fluides à l'aide de Castem2000, Recueil d'exemples commentés (Auteur F. Dabbene, H. Paillère)
- Initiation à la simulation numérique en mécanique des fluides : Eléments d'analyse numérique (Auteur F. Dabbene, H. Paillère)
- Tutorial Cast3M pour la mécanique des fluides (Auteur F. Dabbene)

d. Supports de cours :

- Méthodes numériques avancées en Mécanique non linéaire (Auteur P. Verpeaux)
- Algorithmes et méthodes (Auteur P. Verpeaux)
- Frottement (Auteur P. Verpeaux)
- Non linéarités liées à la thermique (Auteur P. Verpeaux)
- Non convergence (Auteur P. Verpeaux)
- Eléments de dynamique des structures. Illustrations à l'aide de Cast3M (Auteur D. Combescure)
- Introduction à la méthode des éléments finis en mécanique des fluides incompressibles (Auteur S. Gounand).

Annexe H. Traçabilité

 <small>DE LA RECHERCHE À L'INDUSTRIE</small>		Note Technique DEN	Page 2/56
		Réf. : SEMT/LM2S/NT/2019-64669	
		Date : 09/04/2019	Indice : A

Note de fabrication de Cast3M 2019

NIVEAU DE CONFIDENTIALITÉ				
DO	DR	CCEA	CD	SD
X				

PARTENAIRES/CLIENTS	ACCORD	TYPE D'ACTION

RÉFÉRENCES INTERNES CEA			
DIRECTION D'OBJECTIFS	DOMAINE	PROJET	EOTP
DISN	SIMU	MECAN	A-MECAN-01-01
JALON	INTITULÉ DU JALON	DÉLAI CONTRACTUEL DE CONFIDENTIALITÉ	CAHIERS DE LABORATOIRE
JALON COB (PROJET MECAN)	Sortie de Cast3M 2019 : documents qualité associés		

SUIVI DES VERSIONS				
INDICE	DATE	NATURE DE L'ÉVOLUTION	PAGES ET CHAPITRES MODIFIÉS	
A	09/04/2019	Document initial	Toutes	

	NOM	FONCTION	VISAS	DATES
RÉDACTEUR	C. BERTHINIER	Ingénieur Chercheur		09/04/2019
VÉRIFICATEUR(S)	G. FOLZAN	Ingénieur Chercheur		19/04/19
AUTRE(S) VISA(S)				
APPROBATEUR	J. C. LE PALLEC	Chef de Laboratoire		19/04/2019
ÉMETTEUR	V. VANDENBERGHE	Chef de Service		25/04/19

Document propriété du CEA – Reproduction et diffusion externes au CEA soumises à l'autorisation de l'émetteur