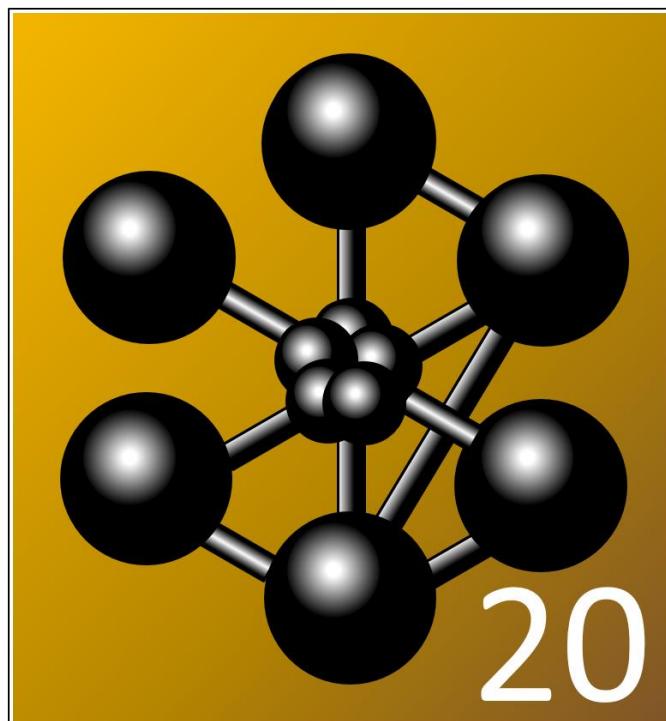


# Assurance Qualité CAST3M

Cast3M 2020

---

## Note de fabrication de Cast3M 2020



---

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 Sécurité Nucléaire (IRSN) dans le cadre des analyses de sûreté des installations nucléaires françaises.

---

## SOMMAIRE

<b>ASSURANCE QUALITE CAST3M .....</b>	<b>1</b>
<b>1. PRESENTATION DE CAST3M 2020.....</b>	<b>5</b>
<b>2. PLATEFORMES DE PRODUCTION DE CAST3M .....</b>	<b>6</b>
2.1 PC – LINUX (32 BITS).....	6
2.2 PC – LINUX (64 BITS).....	6
2.3 PC – WINDOWS (32 BITS) .....	6
2.4 PC – WINDOWS (64 BITS) .....	6
<b>3. ÉLABORATION DE LA VERSION 2020 DE CAST3M .....</b>	<b>7</b>
3.1 OBJET .....	7
3.2 ÉTAPES DE L'ELABORATION DE LA VERSION.....	7
3.2.1 Phase 1 : le 31/12 de l'année « N-1 ».....	7
3.2.2 Phase 2 : du 01/01 au 28/02 de l'année « N » .....	11
3.2.3 Phase 3 : du 01/03 au 31/03 de l'année « N » .....	12
3.2.3.1 Compilateurs GNU Linux 64-bits .....	12
3.2.3.2 Compilateurs GNU Linux 32-bits .....	12
3.2.3.3 Compilateurs GNU Windows 64-bits.....	13
3.2.3.4 Compilateurs GNU Windows 32-bits .....	14
3.2.3.5 Bibliothèques externes .....	15
3.2.3.6 Traducteur Esope vers FORTRAN77.....	15
3.2.3.7 Compilation de Cast3M.....	15
3.2.4 Remarques spécifiques à la plateforme Linux .....	16
3.3 SCHEMA DE PRINCIPE DE LA PREPARATION DES VERSIONS ANNUELLES DE CAST3M.....	17
<b>ANNEXE A.     SCRIPT REPERCUTER.SH .....</b>	<b>18</b>
<b>ANNEXE B.     SCRIPT RECUPERATION_ANOMALIES_AUTOMATIQUE.SH .....</b>	<b>20</b>
<b>ANNEXE C.     SCRIPT PACKAGE.PY.....</b>	<b>22</b>
<b>ANNEXE D.     SCRIPT INSTALL.SH.....</b>	<b>23</b>
<b>ANNEXE E.     SCRIPT CONSTRUCTION_CASTEM_AVEC_SPACK.SH.....</b>	<b>25</b>
<b>ANNEXE F.     SCRIPT ENVIRONNEMENT_CAST3M20.BAT ET ENVIRONNEMENT_CAST3M20.....</b>	<b>29</b>
<b>ANNEXE G.     SCRIPTS COMPILCICAST20.BAT ET COMPILCICAST20 .....</b>	<b>32</b>
<b>ANNEXE H.     SCRIPTS ESSAICAST20.BAT ET ESSAICAST20 .....</b>	<b>43</b>
<b>ANNEXE I.     SCRIPTS CAST_UTIL20.BAT ET CAST_UTIL20.....</b>	<b>50</b>
<b>ANNEXE J.     SCRIPTS CASTEM20.BAT ET CASTEM20 .....</b>	<b>53</b>
<b>ANNEXE K.     DOCUMENTATION CAST3M .....</b>	<b>65</b>
<b>ANNEXE L.     TRAÇABILITE .....</b>	<b>67</b>



## 1. PRESENTATION DE CAST3M 2020

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 thermoplasticité, la mécanique de la rupture, le post-flambage, la ruine des structures, la fatigue. Les structures étudiées sont 1D, 2D ou 3D et de nombreuses lois de comportement des matériaux sont implémentées.

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-Stokes en milieu chargé, équations d'énergie), des modèles de combustion (cinétique d'Arrhénius, modèles EBU ou corrélations, modèles de recombinateur catalytique), et des modèles de condensation (condensation en paroi (Chilton-Colburn), condensation en masse).

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.

Cast3M est un code muni d'un langage 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 :

Modèle de système	: Mandriva Linux release 2011.0 (Official) for i586
Type de processeur	: GenuineIntel Intel® Core™ i5-6500 3,20GHz
Mémoire Vive	: 3000 Mo

Plateforme de test :

Modèle de système	: Debian Linux Squeeze 6.0.10 for i386
Type de processeur	: GenuineIntel Intel® Pentium® CPU G640 2,80GHz
Mémoire Vive	: 1500 Mo

### 2.2 PC – LINUX (64 BITS)

Plateforme de compilation :

Modèle de système	: Mandriva Linux release 2011.0 (Official) for x86_64
Type de processeur	: GenuineIntel Intel® Core™ i5-6500 3,20GHz
Mémoire Vive	: 3000 Mo

Plateforme de test :

Modèle de système	: Debian Linux Squeeze 6.0.10 for x86_64
Type de processeur	: GenuineIntel Intel® Pentium® CPU G640 2,80GHz
Mémoire Vive	: 1500 Mo

### 2.3 PC – WINDOWS (32 BITS)

Plateforme de compilation et de test :

Modèle de système	: Windows 7 Professionnel, Service Pack 1
Type de processeur	: GenuineIntel Intel® Core™ i5-6500 3,20GHz
Mémoire Vive	: 8Go

### 2.4 PC – WINDOWS (64 BITS)

Plateforme de compilation et de test :

Modèle de système	: Windows 7 Professionnel, Service Pack 1
Type de processeur	: GenuineIntel Intel® Core™ i5-6500 3,20GHz
Mémoire Vive	: 8Go

### 3. ÉLABORATION DE LA VERSION 2020 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)

#### 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 : 2020). La Figure 1 décrit l'arborescence de ce répertoire. Le contenu de chacun de ces répertoires est détaillé dans les paragraphes qui suivent.

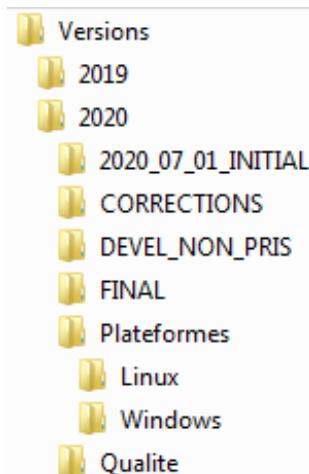


Figure 1. Arborescence des répertoires pour la préparation de Cast3M 2020

- Le répertoire 2020\_07\_01\_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.

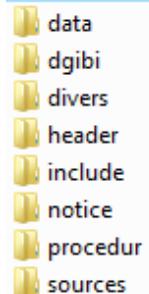


Figure 2. Liste des répertoires à récupérer dans /u2/castem/ sur le réseau du SEMT

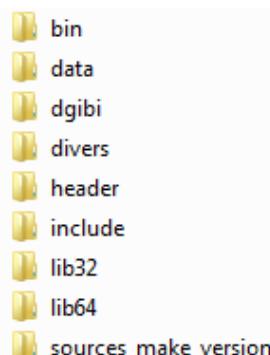
- dgibi : cas tests de la version de développement de Cast3M.
- divers : fichiers de données externes nécessaires pour l'exécution de la base des cas tests.
- header : fichiers d'entêtes.
- include : comprend l'ensemble des includes ESOPE nécessaires.
- notice : comprend l'ensemble des notices disponibles dans Cast3M.
- procedur : comprend l'ensemble des procédures disponibles 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 suivi du numéro d'anomalie.
  - 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 Linux et Windows.
  - Pour chacune des plateformes, un répertoire Dev est créé et contient les dossiers présentés sur la Figure 3. Le script Repercuter.sh génère automatiquement l'arborescence complète ; voir Annexe A
  - Un répertoire GCC contient l'ensemble des outils de développement GNU.
  - Concernant la plateforme Linux, les fonctions définies dans le script Repercuter.sh sont appellées depuis le script Install.sh (voir 0), qui permet la construction automatique de Cast3M sur cette plateforme, voir paragraphe 3.2.4.



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

- Le répertoire bin (dans le répertoire dev) 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 :

- ftnchek-32bits	
- ftnchek-64bits	
- cast_evo120	
- cast_UTIL20	
- castem20	
- compilcast20	
- essaicast20	
- environnement_Cast3M20	(Windows seulement)
- LOGO_ASCII_20.txt	(UNIX & Windows différents)
- liste_rlwrap20	(UNIX seulement)
- rlwrap_32	(UNIX seulement)
- rlwrap_64	(UNIX seulement)

Chaque script est renommé avec l'année en cours dans son 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).

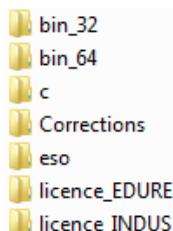


Figure 4. Arborescence du répertoire sources\_make\_version

- 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 les sources spécifiques permettant de différencier la version « education & recherche » de la version « industrielle » de Cast3M.

- Sources spécifiques pour la licence « education & 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. Lorsqu'une évolution concerne le fichier « GIBI.ERREUR » ou des fichiers de type « .eso », « .procedur », « .dgibi », « .notice » ou « .INC », ces fichiers sont récupérés automatiquement dans un dossier au format AA\_MM\_JJ\_numero\_evolution dans le répertoire CORRECTIONS, grâce au script recuperation\_modifications\_automatique.sh (voir Annexe B). Lorsque l'évolution concerne des fichiers de type « .c » ou « .h », ceux-ci sont récupérés depuis l'archive /u2/castem/castem.arc. Par la suite, les évolutions sont triées entre celles qui correspondent à des nouveaux développements et celles qui correspondent à des corrections d'anomalies. Seules les corrections d'anomalies sont intégrés à la nouvelle version de Cast3M, les nouveaux développements sont omis et seront intégrés à la version de Cast3M de l'année suivante.

La répercussion des corrections est effectuée à l'aide du script Repercuter.sh, voir Annexe A.

- Lorsque des sources (.c ou .eso) ou includes (.INC ou .h) sont présents, les exécutables et bibliothèques de la plateforme cible sont automatiquement supprimés (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 de la plateforme cible et ceux du répertoire FINAL sont automatiquement supprimés (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 et LINUX 32/64-bits).

Afin de réaliser cette tâche, il faut disposer d'une suite de compilateurs. Ensuite, il sera nécessaire de compiler quelques bibliothèques 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=/home/thibault/CEA/travail/CASTEM20/Plateformes/Linux64/castem-
20/GCC/GCC-x86_64/bin/gcc-9.2.0
COLLECT_LTO_WRAPPER=/home/thibault/CEA/travail/CASTEM20/Plateformes/Linux64/castem-
-20/GCC/GCC-x86_64/bin/..../libexec/gcc/x86_64-pc-linux-gnu/9.2.0/lto-wrapper
Target: x86_64-pc-linux-gnu
Configured with: /home/kk2000/spack-stage/spack-stage-gcc-9.2.0-
wb7zkukzpxlzjfxkgx4cyo5cfer2qhau/spack-src/configure --
prefix=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-sandybridge/gcc-
9.2.0/gcc-9.2.0 --enable-languages=c,c++,fortran --program-suffix=-9.2.0 --
disable-libquadmath --disable-libquadmath-support --with-
mpfr=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-sandybridge/gcc-
9.2.0/mpfr-3.1.6 --with-gmp=/home/kk2000/Documents/CASTEM20/linux-
mandrivalinux2011-sandybridge/gcc-9.2.0/gmp-6.1.2 --disable-multilib --with-
system-zlib --with-mpc=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-
sandybridge/gcc-9.2.0/mpc-1.1.0 --with-isl=/home/kk2000/Documents/CASTEM20/linux-
mandrivalinux2011-sandybridge/gcc-9.2.0/isl-0.20
Thread model: posix
gcc version 9.2.0 (GCC)
```

#### 3.2.3.2 Compilateurs GNU Linux 32-bits

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

```
Utilisation des specs internes.
COLLECT_GCC=../linux-mandrivalinux2011-i686/gcc-6.5.0/gcc-6.5.0/bin/gcc-6.5.0
COLLECT_LTO_WRAPPER=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-
i686/gcc-6.5.0/gcc-6.5.0/libexec/gcc/i686-pc-linux-gnu/6.5.0/lto-wrapper
Cible : i686-pc-linux-gnu
Configuré avec: /home/kk2000/spack-stage/spack-stage-gcc-6.5.0-
5vbjr2m2oxndz77jdlew4xe3lhjsluzd/spack-src/configure --
prefix=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-i686/gcc-6.5.0/gcc-
6.5.0 --enable-languages=c,c++,fortran --program-suffix=-6.5.0 --disable-
libquadmath --disable-libquadmath-support --with-
mpfr=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-i686/gcc-6.5.0/mpfr-
3.1.6 --with-gmp=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-i686/gcc-
6.5.0/gmp-6.1.2 --disable-multilib --with-system-zlib --with-
mpc=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-i686/gcc-6.5.0/mpc-
1.1.0 --with-isl=/home/kk2000/Documents/CASTEM20/linux-mandrivalinux2011-i686/gcc-
6.5.0/isl-0.18
Modèle de thread: posix
gcc version 6.5.0 (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=bin\gcc.exe

COLLECT_LTO_WRAPPER=d:/Users/tlindeck/Desktop/Fabrication_Cast3M_WIN/Dev/GCC/GCC-x86_64/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-fuzzy-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=' -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=bin\gcc.exe
COLLECT_LTO_WRAPPER=d:/Users/tlindeck/Desktop/Fabrication_Cast3M_WIN/Dev/GCC/GCC-i686/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-fully-dynamic-
string --enable-version-specific-runtime-libs --enable-libstdcxx-fs=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='-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' 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=' -pipe -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 Bibliothèques externes

Toutes les bibliothèques nécessaires doivent être compilées avec la même version de GCC que celle utilisée pour Cast3M. On utilise toujours la version statique des bibliothèques (fichier « .a ») lorsque cela est possible, renommée en ajoutant l'architecture (32 ou 64 bits) et placée dans le répertoire lib32/ ou lib64/ correspondant au répertoire Cast3M de chaque plateforme. La liste exhaustive des bibliothèques est :

- **XDR** – Uniquement pour Linux. On récupère les versions 32 bits et 64 bits à l'aide de la commande ‘urpmi libtirpc-devel’. Pour l'architechnture 64 bits, le chemin d'installation de la bibliothèque est « /usr/lib64/libtirpc.a ». Pour l'architechnture 64 bits, le chemin d'installation de la bibliothèque est « /usr/lib/libtirpc.a ».
- **FXDR** - La version 32 bits est la version disponible sur [http://meteora.ucsd.edu/~pierce/fxdr\\_home\\_page.html](http://meteora.ucsd.edu/~pierce/fxdr_home_page.html) et la version 64 bits est une version modifiée disponible sur le réseau local du SEMT
- **JPEG** – La version utilisée pour Cast3M 2020 est la version 9c disponible sur <http://www.ijg.org>
- **ZLIB** – La version utilisée pour Cast3M 2020 est la version 1.2.11 disponible sur <http://zlib.net>
- **HDF5** - La version utilisée pour Cast3M 2020 est la version 1.10.3 disponible sur <https://support.hdfgroup.org>
- **MED** - La version utilisée pour Cast3M 2020 est la version 4.0.0 disponible sur <http://files.salome-platform.org>
- **FreeGLUT** - La version utilisée pour Cast3M 2020 est la version 3.0.0 disponible sur <http://prdownloads.sourceforge.ne>
- **openMPI** - La version utilisée pour Cast3M 2020 est la version 4.0.0 disponible sur <https://www.open-mpi.org>
- **ESOPE\_GEMAT** : La version du traducteur et du gestionnaire de mémoire ESOPE est disponible sur le réseau local du SEMT : /u2/esope/esope2019

### 3.2.3.6 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      compilcast20 -ESOPE \*.eso \*.c
- Edition des liens avec la commande      essaicast20 -ESOPE

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

### 3.2.3.7 Compilation de Cast3M

Les sources (eso et C) sont compilées à l'aide du script compilcast20 -f \*.eso \*.c (UNIX et WINDOWS) (voir Annexe G).

- 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 .txt 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. 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 essaicast20 (voir Annexe H), selon l'architecture en cours de compilation.

- Lors de la toute première tentative de faire l'édition des liens, ni l'exécutable ni la bibliothèque 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\_20.txt 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 bibliothèque 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 et ce pour toutes les plates-formes et toutes les architectures supportées. Le script castem20 -test (voir Annexe J) 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.

### 3.2.4 Remarques spécifiques à la plateforme Linux

Sur cette plateforme, la construction de Cast3M, de ses dépendances et des compilateurs est automatisée. Cela est rendu possible par l'utilisation du gestionnaire de paquet Spack (<https://spack.io>). Le script Install.sh (voir 0) est responsable de la construction de Cast3M; ce script est appellé par Spack lorsque le « package castem » - dont le fichier principal package.py est disponible en Annexe C - est construit. Enfin, le script construction\_castem\_avec\_spack.sh (0) permet d'effectuer les phases 1, 2 et 3 présentées dans ce document, de façon automatique.

### 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.

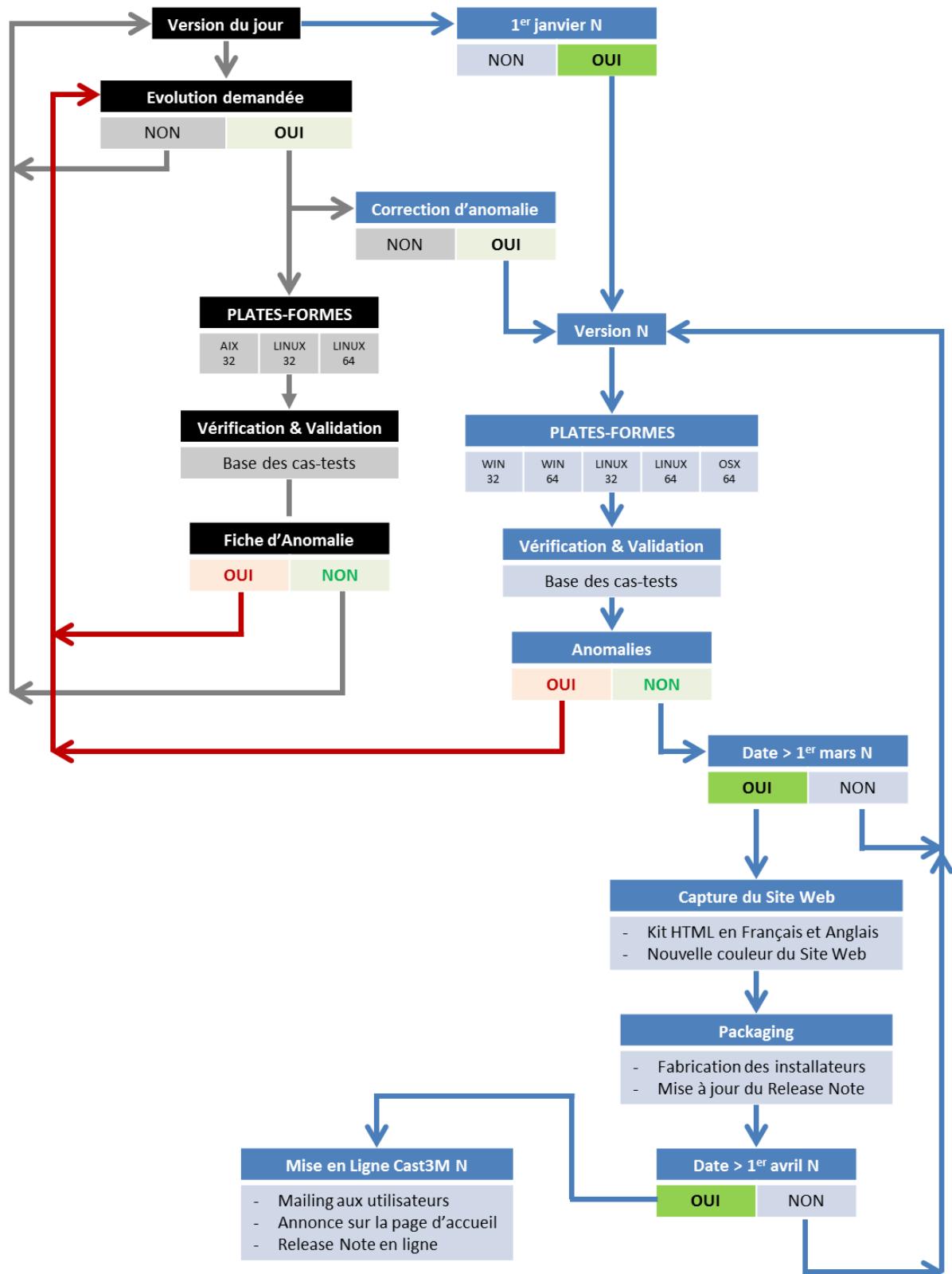


Figure 5. Organigramme de la préparation des versions annuelles de Cast3M

## Annexe A. Script Repercuter.sh

```
#!/bin/bash

#####
# Fonctionnalites du script Repercuter.sh:
#
# Ce Script prend en argument une liste de Répertoire (avec
# séparateur ':') #
# contenant des évolutions de Cast3M (.dgibi, .procedur,
.notice, .eso, #
# .INC, GIBI.ERREUR).#
#
#           * Les fichiers spécifiques à la plateforme sont
installés dans #
# le répertoire définit avec l'option obligatoire '--prefix='
#
#           * Les fichiers communs à toutes les plateformes
sont installés dans #
# le répertoire définit avec l'option '--repertoire_final=' si fournie. #
#####
#Ce script peut être utilisé de deux façons différentes :
# a) Le script est appelé depuis un shell interactif en
fournissant les arguments de contrôle :
#           ex: ./Repercuter.sh --prefix=/home/invite/test
--repertoire_a_repercuter_initial=/home/invite/rep_initial
#           La liste des arguments valides est la suivante
:
#           --prefix=...
#           --repertoire_final=...
#           --repertoire_a_repercuter_initial=...
#           --repertoire_a_repercuter_correction=...
#           --repertoire_a_repercuter_divers=...
# b) La fonction 'repercute' définie ici est utilisée depuis
un script :
#           Cela nécessite d'avoir "source" Repercuter.sh.
#           À la place de fournir les arguments de contrôle,
certaines variables doivent alors être définies :
#           PREFIX
#           REP_FINAL           à la
place de '--repertoire_final='...
#           REP_REPERCUTE_INITIAL   à la
place de '--repertoire_a_repercuter_initial='...
#           REP_REPERCUTE_CORRECTION à la
place de '--repertoire_a_repercuter_correction='...
#           REP_REPERCUTE_DIVERS   à la
place de '--repertoire_a_repercuter_divers='...

#1) définition des fonctions :
function erreur {
    if [ $# -eq 0 ]; then
        echo "usage : ./prog ${args_list[@]} >&2
    else
        echo "erreur : $1" >&2
    fi
    [ "${_KEEP_GOING}" == "FAUX" ] && exit 1
}

function creation_repertoire {
    for d in $@; do
        [ ! -d "${d}" ] && mkdir -p "${d}"
    done
}

function install {
    rep_installation=$1
    fichiers_a_installer=$2
    [ -z "${fichiers_a_installer}" ] && return
    creation_repertoire "${rep_installation}"
    for f in ${fichiers_a_installer}; do
        cp -rf ${f} ${rep_installation}/
    done
}

function repercute {

    function repertoires_final_doit_être_utilisé {
        [ -n "$(_REP_FINAL)" ] && [ -d "$(_REP_FINAL)" ] && echo
"VRAI" || echo "FAUX"
    }
}
```

```
function castem_bin_doit_être_utilisé {
    [ -n "${CASTEM_BIN_PREFIX}" ] && [ -d
"${CASTEM_BIN_PREFIX}" ] && echo "VRAI" || echo "FAUX"
}

if [ "$1" == "initial" ]; then
    rep="$(_REP_REPERCUTE_INITIAL)"
    repertoires_Dev=( dgibi procedur notice sources_make_version/eso sources_make_version/c include/eso include/c data/ )
elif [ "$1" == "correction" ]; then
    rep="$(_REP_REPERCUTE_CORRECTION)"
    repertoires_Dev=( dgibi procedur notice sources_make_version/Corrections sources_make_version/c include/eso include/c data/ )
else
    erreur "mauvaise utilisation de la fonction 'repercute'"
fi

types=( dgibi procedur notice eso c INC à la place de '--prefix=' h ERREUR )

declare -a liste_fichiers_trouves

repertoires_Final=( dgibi procedur notice sources sources include/eso include/c data/ )

#Traitement des arguments
declare -a rep_arr
for d in `echo ${rep} | tr ':' ' '`; do
    [ "$d" == "$(_REP_FINAL)" ] && erreur "Le chemin $d n'est pas
absolu."
    rep_arr+=(`find ${d} -type f`)
done

#Boucle sur les types de fichiers
for ((i=0; i < ${#types[@]}; i++)); do

    #Remplissage de liste_fichiers_trouves
    for f in ${rep_arr[@]}; do
        if [ `basename ${f} 2>/dev/null` == `basename ${f}
.${types[$i]} 2>/dev/null.${types[$i]}` ]; then
            liste_fichiers_trouves[$i]="${liste_fichiers_trouves[$i]} ${f}"
        fi
    done

    #Installation des fichiers :
    echo "fichiers ${types[$i]} à répercuter :"
    ls -1 ${liste_fichiers_trouves[$i]}
}
```

# NOTE DE FABRICATION DE CAST3M 2020

```

#Cas particulier des dgibi :
if [ "${types[$i]}" == "dgibi" ]; then
    for f in ${liste_fichiers_trouves[$i]}; do
        sed -i s\?'/u2/castem/divers'\?..../divers'\?g ${f}
        sed -i s\?'u/castem/divers'\?..../divers'\?g ${f}
    done
fi

#Installation dans le bon sous-repertoire de PREFIX/
install "${PREFIX}/$({repertoires}_Dev[$i])"
"${liste_fichiers_trouves[$i]}"

#Installation dans le bon sous-repertoire de FINAL/
if [ `repertoire_final_doit_être_utilisé` == "VRAI" ];
then
    #les fichiers .c ne sont pas fournis :
    [ "${types[$i]}" != "c" ] && install
"${REP_FINAL}/$({repertoires}_Final[$i]}"
"${liste_fichiers_trouves[$i]}"
fi

#Suppression des fichiers qui doivent être regénérés (garde-fou) :
if [ -n "${liste_fichiers_trouves[$i]}" ]; then

    #Si fichiers 'notice' présents, alors suppression de
${PREFIX}/data/CAST3M.MASTER :
if [ "${types[$i]}" == "notice" ]; then
    [ -f ${PREFIX}/data/CAST3M.MASTER ] && rm -f
${PREFIX}/data/CAST3M.MASTER
fi

    #Si fichiers 'procedur' présents, alors suppression de
${PREFIX}/data/CAST3M.PROC :
if [ "${types[$i]}" == "procedur" ]; then
    [ -f ${PREFIX}/data/CAST3M.PROC ] && rm -f
${PREFIX}/data/CAST3M.PROC
fi

    #Si fichiers sources présents, alors suppression de
${PREFIX}/bin/bin_Cast3M_* et de
${PREFIX}/lib{32,64}/libcastem_{EDURE,INDUS}_{32,64}.a :
if [ [ "${types[$i]}" == "eso" || "${types[$i]}" == "INC" ||
" ${types[$i]}" == "c" || "${types[$i]}" == "h" ]; then
    rm -f ${PREFIX}/bin/bin_Cast3M_*
    [ -f ${PREFIX}/lib32/libcastem_EDURE_32.a ] && rm -f
${PREFIX}/lib32/libcastem_EDURE_32.a
    [ -f ${PREFIX}/lib32/libcastem_INDUS_32.a ] && rm -f
${PREFIX}/lib32/libcastem_INDUS_32.a
    [ -f ${PREFIX}/lib64/libcastem_EDURE_64.a ] && rm -f
${PREFIX}/lib64/libcastem_EDURE_64.a
    [ -f ${PREFIX}/lib64/libcastem_INDUS_64.a ] && rm -f
${PREFIX}/lib64/libcastem_INDUS_64.a
fi

    fi

done

#Installation des fichiers 'divers' :
[ `repertoire_final_doit_être_utilisé` == "VRAI" ] &&
install "${REP_FINAL}/divers" ${REP_REPERCUTE_DIVERS}/*
install "${PREFIX}/divers" ${REP_REPERCUTE_DIVERS}/*
for f in ${rep_arrr[@]}; do
    divers="VRAI"
    for ((i=0; i < ${#types[@]}; i++)); do
        if [ `basename ${f} 2>/dev/null` == `basename ${f}` .
${types[$i]} 2>/dev/null` ]; then
            divers="FAUX"
        fi
    done
    if [ "$divers" == "VRAI" ]; then
        liste_fichiers_divers="${liste_fichiers_divers} ${f}"
    fi
done
[ `repertoire_final_doit_être_utilisé` == "VRAI" ] &&
install "${REP_FINAL}/divers" "${liste_fichiers_divers}"
install "${PREFIX}/divers" "${liste_fichiers_divers}"

#Cas particulier du cas-test 'acqulata.dgibi' :
if [ -f "${PREFIX}/dgibi/acqulata.dgibi" ]; then
    [ `repertoire_final_doit_être_utilisé` == "VRAI" ] && cp
-f "${PREFIX}/dgibi/acqulata.dgibi"
"${REP_FINAL}/divers/acqulata.dgibi"
cp -f "${PREFIX}/dgibi/acqulata.dgibi"
"${PREFIX}/divers/acqulata.dgibi"

fi

# Installation des sources espace dans PREFIX :
install "${PREFIX}/sources"
${PREFIX}/sources_make_version/Corrections/*
install "${PREFIX}/sources"
${PREFIX}/sources_make_version/eso/*
if [ -f ${PREFIX}/sources/*.c ]; then
    rm -f ${PREFIX}/sources/*.c
fi

#Copie de "castem-bin" dans PREFIX/bin :
[ `castem_bin_doit_être_utilisé` == "VRAI" ] && install
"${PREFIX}/bin" ${CASTEM_BIN_PREFIX}/bin/*

#Archivage de tout ce qui est dans le répertoire FINAL, pour
l'installateur IzPack :
if [ `repertoire_final_doit_être_utilisé` == "VRAI" ];
then
    for dir in ${REP_FINAL}/*; do
        zip -r ${dir}.zip ${dir}
    done
fi

#somme-nous source ?
SOURCE="FAUX"
if [ $( cd ${dir} && pwd ) / $( basename ${BASH_SOURCE[0]} ) != $( cd ${dir} && pwd ) / $( basename $0 ) ]; then
    SOURCE="VRAI"
fi

#2) travail :
if [ "$SOURCE" == "FAUX" ]; then
    KEEP_GOING="FAUX"

args_list=(
    '--prefix='
    '--repertoire_final='
    '--repertoire_a_repercuter_initial='
    '--repertoire_a_repercuter_correction='
    '--repertoire_a_repercuter_divers='
)
args_valeur=(
    'NULL'
    ''
    ''
    ''
)

#Lectures des arguments attendus :
for arg in ${@}; do
    let i=0
    for valeurs in ${args_list[@]}; do
        arg_="${arg/$valeurs/}"
        [ "${arg_}" != "${arg}" ] && args_valeur[$i]="${arg_}"
        let i++
    done
done

#Verification que les arguments obligatoires ont bien été lus
for valeurs in ${args_valeur[@]}; do
    [ "${valeurs}" == "NULL" ] && erreur
done

let i=0
#Variables globales du script, elles sont utilisées dans les
différentes fonctions.
PREFIX="${args_valeur[$((i++))]}"
REP_FINAL="${args_valeur[$((i++))]}"
REP_REPERCUTE_INITIAL="${args_valeur[$((i++))]}"
REP_REPERCUTE_CORRECTION="${args_valeur[$((i++))]}"
REP_REPERCUTE_DIVERS="${args_valeur[$((i++))]}"

if [ -n "${REP_REPERCUTE_INITIAL}" ]; then
    repercuter "initial"
fi
if [ -n "${REP_REPERCUTE_CORRECTION}" ]; then
    repercuter "correction"
fi
fi

```

## Annexe B. Script recuperation\_anomalies\_automatique.sh

```

#!/bin/bash

#Chaque jour, ce script detecte les nouvelles evolutions
#et recupere les fichiers concernees dans un repertoire
#dont le nom respecte le motif "AA_MM_JJ_E" :
# AA : l'annee
# MM : le mois
# JJ : le jour
# E : le numero de l'evolution

recupere_fichiers_depuis_serveur_semt(){
#menage :
[ -d ${data_rep} ] && rm -rf ${data_rep}
mkdir ${data_rep}

#recuperation des fichiers sur le serveur
test_erreur(){
if [ $? -ne 0 ]; then
echo "date` :: Probleme lors de la recuperation de
`${serveur}: ${castem_rep}/$1'" >> ${erreur_log}
cat ${erreur_log}
exit 1
fi
}

#fichiers suivis par hist.hist
sshpass -p ${mdp} scp
${user}@${serveur}: ${castem_rep}/hist.hist ${data_rep}/
test_erreur hist.hist
sshpass -p ${mdp} scp
${user}@${serveur}: ${castem_rep}/castem.arc ${data_rep}/
test_erreur castem.arc
sshpass -p ${mdp} scp -r
${user}@${serveur}: ${castem_rep}/procedur ${data_rep}/
test_erreur procedur
sshpass -p ${mdp} scp -r
${user}@${serveur}: ${castem_rep}/dgibi ${data_rep}/
test_erreur dgibi
sshpass -p ${mdp} scp -r
${user}@${serveur}: ${castem_rep}/notice ${data_rep}/
test_erreur notice

#fichiers non fichiers suivis par hist.hist : la date est
importante
sshpass -p ${mdp} scp -p
${user}@${serveur}: ${castem_rep}/GIBI.ERREUR ${data_rep}/
test_erreur GIBI.ERREUR
sshpass -p ${mdp} scp -r -p
${user}@${serveur}: ${castem_rep}/include ${data_rep}/
test_erreur include
}

recupere_fichiers_trouves()
{
hier_underscore=${1}
liste_fichier=${2}
liste_numero_anomalie=${3}
fichiers_esope=${4}

test_erreur()
if [ $? -ne 0 ]; then
echo "date` :: Probleme dans 'recupere_fichiers_trouves'"
>> ${erreur_log}
cat ${erreur_log}
exit 1
fi
}

if [ -n "${liste_fichier}" ]; then
#Passage en vecteur :
liste_fichier=(${liste_fichier})
liste_numero_anomalie=(${liste_numero_anomalie})
let size=${#liste_fichier[@]}
for (( i=0; i<size; i++ ))
do
fichier=${liste_fichier[$i]}
ano=${liste_numero_anomalie[$i]}
nom_rep="${hier_underscore}_${ano}"
[ ! -d ${nom_rep} ] && mkdir "${nom_rep}"
cd "${nom_rep}"
#echo "$ano - $fichier"
if [ "${fichiers_esope}" = "VRAI" ]; then

```

```

        arc -eon ${data_rep}/castem.arc ${fichier}
        test_erreur
    else
        cp ${fichier} .
    fi
    cd ..
done
fi
}

recherche_et_copie()
{
type=${1}
rep=${2}
extension=${3}
liste_fichier_ok=`mktemp`
grep ${type} ${liste_fichier} > ${liste_fichier_ok}
liste=
liste_numero_anomalie=
while read line; do
word=`echo "$line" | awk '{print tolower($2)}'` 
[ ${#word} -ge 8 ] && word=${word}* #le nom du fichier est
peut etre tronque
word=${word}.${extension}
ano=`echo "$line" | awk '{print $NF}'` 
#la casse des caracteres constituant le nom du fichier n'est
pas connue. Il faut donc faire appel a la commande find avec
l'option '-iname'
find_name=`mktemp`
find ${rep} -iname "${word}" > ${find_name}
while read fichier; do
liste+=${liste} ${fichier}
liste_numero_anomalie+=${liste_numero_anomalie} ${ano}
done <${find_name}
done <${liste_fichier_ok}
recupere_fichiers_trouves "${hier_underscore}" "${liste}"
"${liste_numero_anomalie}"
})

recupere_evolutions()
{
#0) Si /u2 n'est pas monte sur la machine, alors les donnees
necessaire sont recuperées via scp.
if [ ! -d /u2/castem ]; then
data_rep=`pwd`/.data
recupere_fichiers_depuis_serveur_semt
else
data_rep=/u2/castem
fi

#1) recuperation des lignes dans hist.hist
hier=`date --date="yesterday" +"%y/%m/%d"`
#la date comme elle est indiquee dans hist.hist :
hier_underscore=`date --date="yesterday" +"%y_%m_%d"`
liste_fichier=`mktemp`
grep "${hier}" ${data_rep}/hist.hist > ${liste_fichier}

#2) recuperation des fichiers

#2a) les fichiers eso :
liste_fichier_eso=`mktemp`
grep SOURCE ${liste_fichier} > ${liste_fichier_eso}
liste_eso=
liste_numero_anomalie=
while read line; do
eso=`echo "$line" | awk '{print tolower($2)}'.eso
liste_eso+=${liste_eso} ${eso}
ano=`echo "$line" | awk '{print $NF}'` 
liste_numero_anomalie+=${liste_numero_anomalie} ${ano}
done <${liste_fichier_eso}
recupere_fichiers_trouves "${hier_underscore}" "${liste_eso}"
"${liste_numero_anomalie}" "VRAI"

#2b) les fichiers procedur :
recherche_et_copie "PROCEDUR" "${data_rep}/procedur/"
"procedur"

#2c) les fichiers dgibi :
recherche_et_copie "DGIBI" "${data_rep}/dgibi/" "dgibi"

#2d) les fichiers notice :
recherche_et_copie "NOTICE" "${data_rep}/notice/" "notice"

```

```

#2e) Recuperation de GIBI.ERREUR seulement s'il a ete modifie
durant les 24 dernieres heures :


```

## Annexe C. Script package.py

```

from spack import *

class Castem(Package):
    url='file:///.../castem20.zip'

    depends_on('repo.rpc-precompiled')
    depends_on('repo.hdf5')
    depends_on('repo.med')
    depends_on('libjpeg')
    depends_on('repo.freeglut')
    depends_on('zlib')
    depends_on('repo.openmpi')
    depends_on('repo.fxdr')
    depends_on('repo.esope')
    depends_on('repo.castem-bin')
    depends_on('repo.tfel')
    depends_on('numactl')
    depends_on('binutils', when='+binutils')

version('20', sha256='08931550d7588889d3dc4b9cc4d1c37a86facaf3036596201d4
ccde536544d5b')

variant('castem_release',
       default=False,
       description='Build release version of Cast3M')
variant('repertoire_a_repercuter_initial',
       default=' ',
       description='repertoire contenant les sources de Cast3M à
utiliser')
variant('repertoire_a_repercuter_correction',
       default=' ',
       description='repertoire contenant les sources de Cast3M à
utiliser')
variant('repertoire_a_repercuter_divers',
       default=' ',
       description='repertoire contenant les divers de Cast3M')
variant('repertoire_final',
       default=' ',
       description='repertoire FINAL (fabrication de la version
annuelle)')
variant('binutils',
       default=False,
       description='utilisation du binutils specifie')

def setup_environment(self, spack_env, run_env):
    spec = self.spec
    if spec.satisfies('+binutils'):
        #On souhaite utiliser en priorité ce qui se trouve dans
        'binutils' :
            spack_env.prepend_path('PATH', spec['binutils'].prefix.bin)
            spack_env.prepend_path('LD_LIBRARY_PATH',
spec['binutils'].prefix.libs)
            spack_env.prepend_path('LIBRARY_PATH',
spec['binutils'].prefix.libs)

def install(self, spec, prefix):
    options = [
        '--prefix={0}'.format(prefix),
        '--with-esope-version={0}'.format(self.version),
        '--with-castem-bin-version={0}'.format(self.version),
        '--with-hdf5={0}'.format(spec['hdf5'].prefix),
        '--with-med={0}'.format(spec['med'].prefix),
        '--with-libjpeg={0}'.format(spec['libjpeg'].prefix),
        '--with-freeglut={0}'.format(spec['freeglut'].prefix),
        '--with-zlib={0}'.format(spec['zlib'].prefix),
        '--with-openmpi={0}'.format(spec['openmpi'].prefix),
        '--with-fxdr={0}'.format(spec['fxdr'].prefix),
        '--with-esope={0}'.format(spec['esope'].prefix),
        '--with-castem-bin={0}'.format(spec['castem-bin'].prefix),
        '--with-mfront={0}'.format(spec['tfel'].prefix),
        '--with-numactl={0}'.format(spec['numactl'].prefix),
        '--with-rpc={0}'.format(spec['rpc-precompiled'].prefix),
    ]
    if '+castem_release' in spec:
        options.append('--castem_release=VRAI')

    repertoire_a_repercuter_initial =
self.spec.variants['repertoire_a_repercuter_initial'].value
    if repertoire_a_repercuter_initial:
        options.append('--
repertoire_a_repercuter_initial={0}'.format(repertoire_a_repercuter_init
ial))

    repertoire_a_repercuter_correction =
self.spec.variants['repertoire_a_repercuter_correction'].value
    if repertoire_a_repercuter_correction:
        options.append('--
repertoire_a_repercuter_correction={0}'.format(repertoire_a_repercuter_c
orrection))

    repertoire_a_repercuter_divers =
self.spec.variants['repertoire_a_repercuter_divers'].value
    if repertoire_a_repercuter_divers:
        options.append('--
repertoire_a_repercuter_divers={0}'.format(repertoire_a_repercuter_diver
s))

    repertoire_final =
self.spec.variants['repertoire_final'].value
    if repertoire_final:
        options.append('--
repertoire_final={0}'.format(repertoire_final))

    install_sh = Executable('./Install.sh')
    install_sh(*options)

```



## Annexe D. Script Install.sh

```
#!/bin/bash

#####
# Fonctionnalites du script Install.sh:
#
# Ce Script est normalement utilise par spack (voir package.py de
Castem). #
# Il permet, lorsque les differentes dependances de Cast3M
#
# ont ete construites, de construire ce dernier.
#
#####

#Les arguments obligatoires sont :
# --prefix= Chemin d'installation de Cast3M
# --with-hdf5= Chemin d'installation de HDF5
# --with-med= Chemin d'installation de MED
# --with-libjpeg= Chemin d'installation de LIBJPEG
# --with-freeglut= Chemin d'installation de FREEGLUT
# --with-zlib= Chemin d'installation de ZLIB
# --with-openmpi= Chemin d'installation de OPENMPI
# --with-fxdr= Chemin d'installation de FXDR
# --with-esope= Chemin d'installation de ESOP
# --with-castem-bin= Chemin d'installation de Castem-bin
# --with-numactl= Chemin d'installation de NUMACTL
# --with-rpc= Chemin d'installation de RPC-
PRECOMPILED

#Les arguments optionnels sont :
# --castem_release= VRAI ou FAUX
# --repertoire_final= Repertoire 'FINAL' pour
installation fichiers communs
#
# (voir Repercuter.sh)
# --with-esope-version= Version d'Esope a utiliser
# --with-castem-bin-version= Version de Castem-bin a
utiliser
# --repertoire_a_repercuter_initial= voir Repercuter.sh
# --repertoire_a_repercuter_correction= voir Repercuter.sh
# --repertoire_a_repercuter_divers= voir Repercuter.sh

this_rep=$( cd "$( dirname "${BASH_SOURCE[0]}" )" && pwd )

KEEP_GOING="FAUX"

#Utilisation de la fonction 'repercute' du script Repercuter.sh
source ${this_rep}/Repercuter.sh

function ajout_a_cpath {
echo
${NUMACTL_PREFIX}/include:${LIBJPEG_PREFIX}/include:${ESOPE_PREFIX}/incl
ude/c:/usr/include/x86_64-linux-gnu
}

function installation_esope_et_castem_bin {
#installation de 'castem-bin' :
install "${PREFIX}/bin" "${CASTEM_BIN_PREFIX}/bin/*"
#installation de 'esope' :
install "${PREFIX}/bin/" "${ESOPE_PREFIX}/bin/*"
install "${PREFIX}/lib64/" "${ESOPE_PREFIX}/lib/*"
install "${PREFIX}/lib32/" "${ESOPE_PREFIX}/lib/*"
}

function copie_fichiers_initiaux {
#verification de Cast3M :
install "${REP_FINAL}/*" "${this_rep}/verification_Cast3M"
install "${PREFIX}/*" "${this_rep}/verification_Cast3M"

#fichier lies aux licences :
install "${PREFIX}/sources_make_version/licence_EDURE"
"${this_rep}/licence_EDURE/*"
install "${PREFIX}/sources_make_version/licence_INDUS"
"${this_rep}/licence_INDUS/*"

#header :
install "${REP_FINAL}/*" "${this_rep}/header"
install "${PREFIX}/*" "${this_rep}/header"
}

function installation_libraries {

```

```
function trouve_archive {
prefix="$1"
lib_name="$2"
TROUVE="FAUX"

cd `mktemp -d`
for archive in ${prefix}/lib*/${lib_name}.a; do
    cp ${archive} .
    _archive=`basename ${archive}`
    obj=`ar t ${_archive} | head -1` 
    ar x ${_archive} ${obj}
    LC_ALL=C file ${obj} | grep "x86-64" 2>&1 > /dev/null
    if [ $? -eq 0 ]; then
        [ "${BIT}" == "64" ] && {
            TROUVE="VRAI"
            break;
        }
    else
        [ "${BIT}" == "32" ] && {
            TROUVE="VRAI"
            break;
        }
    fi
    rm -f ${_archive} ${obj}
done

if [ "$TROUVE" == "VRAI" ]; then
    echo "${archive}"
else
    erreur "Aucune archive trouvées"
fi
}

rep="${PREFIX}/lib${BIT}"
mkdir -p ${rep}
cp `trouve_archive "${HDF5_PREFIX}" "libhdf5"` ${rep}/libhdf5_${BIT}.a
cp `trouve_archive "${MED_PREFIX}" "libmed"` ${rep}/libmed_${BIT}.a
cp `trouve_archive "${LIBJPEG_PREFIX}" "libjpeg"`
${rep}/libjpeg_${BIT}.a
cp `trouve_archive "${FREEGLUT_PREFIX}" "libglut"`
${rep}/libglut_${BIT}.a
cp `trouve_archive "${ZLIB_PREFIX}" "libz"` ${rep}/libz_${BIT}.a
cp `trouve_archive "${FXDR_PREFIX}" "libfxdr"` ${rep}/libfxdr_${BIT}.a
cp `trouve_archive "${ESOPE_PREFIX}" "libesope"` ${rep}/
cp `trouve_archive "${NUMACTL_PREFIX}" "libnuma"`
${rep}/libnuma_${BIT}.a #On utilise celle du systeme plutot ?
cp `trouve_archive "${RPC_PREFIX}" "librpc"` ${rep}/librpc_${BIT}.a

#installation des librairies pre-compilées :
cp ${this_rep}/lib${BIT}/* ${rep}/
}

function installation_gcc_mpi_mfront {
#Installation de mpi :
install "${PREFIX}/mpi/mpis${BIT}" "${MPI_PREFIX}/*"
#Installation de Mfront :
if [ "${BIT}" == "64" ]; then
    install "${PREFIX}/MFRONT/mfront-x86_64" "${MFRONT_PREFIX}/*"
    else
    install "${PREFIX}/MFRONT/mfront-i686" "${MFRONT_PREFIX}/*"
fi
}

function construction {
#Environnement pour la construction des .c de Cast3M :
export CPATH='ajout_a_cpath':${CPATH}

#Creation des repertoires necessaires a la construction :
creation_repertoire ${PREFIX}/sources_make_version/bin_32 \
${PREFIX}/sources_make_version/bin_64 \
${PREFIX}/sources_make_version/c \
${PREFIX}/sources_make_version/Corrections \
${PREFIX}/sources_make_version/eso \
${PREFIX}/sources_make_version/licence_EDURE \
${PREFIX}/sources_make_version/licence_INDUS

echo "SRC"
rm -f ${PREFIX}/bin/bin_Cast3M_*
```





## Annexe E. Script construction\_castem\_avec\_spack.sh

```
#!/bin/bash

CASTEM_RELEASE="VRAI"
GCC_CIBLE="9.2.0"
GCC_INI="8.2.0"

this_rep=$( cd "$( dirname "${BASH_SOURCE[0]}" )" && pwd )"
REP_TRAVAIL="${HOME}/Documents/CASTEM20"

#Sous Mandriva, il est nécessaire de construire Python depuis les
sources :
#Les images disques des machines Mandriva 32 et 64 bits utilisées sont
disponibles sur :
#
ftp://ftp.nluug.nl/pub/os/Linux/distr/Mandrivalinux/official/iso/2011/
if [ -f /etc/release ] && [[ `cat /etc/release` =~ ^Mandriva ]]; then

#Utilisation du GCC système :
GCC_INI="4.6.1"

#Note1 :
#Le package 'mesa-glu' a besoin de la libGL. Celle-ci doit
s'appeler exactement 'libGL.so' sans suffixe
#Il est parfaitement nécessaire d'avoir défini les variables
d'environnements suivantes :
# GL_LIBS="-L/usr/lib64/mesa -lGL"
# GL_CFLAGS="$(GL_LIBS)"
# LIBRARY_PATH="$(gl_dir):$LIBRARY_PATH"

#Note2 :
#Seuls les paquets suivants ont été installés dans les machines
virtuelles :
# tcl-devel libmesagl1-devel libmesagl1-devel libxi-devel gcc-c++

#Note3 :
#Pour une construction portable, il est nécessaire de patcher le
fichier "${SPACK_DIR}/lib/spack/spack/build_environment.py" :
# - env.set('SPACK_TARGET_ARGS', isa_arg)
# + env.set('SPACK_TARGET_ARGS', '-march=x86-64 -mtune=generic')
=> Linux64
# + env.set('SPACK_TARGET_ARGS', '-march=i686 -mtune=generic')
=> Linux32
#plus d'informations sur https://gcc.gnu.org/onlinedocs/gcc/x86-
Options.html

#SPACK - installation et environnement :
CHROOT="${REP_TRAVAIL}/spack"
if [ ! -d ${CHROOT}/Python ]; then
    miroir="192.168.1.23:81"
    ${this_rep}/construction_logiciel_automatique.sh install
    http://$miroir/zlib-1.2.11.tar.gz \
        http://$miroir/libffi-3.3.tar.gz \
        http://$miroir/openssl-1.1.1f.tar.gz \
            --root=${CHROOT}
    ${this_rep}/construction_logiciel_automatique.sh install m4 --
    root=${CHROOT}
    ${this_rep}/construction_logiciel_automatique.sh install autoconf
    env m4 --root=${CHROOT}
    ${this_rep}/construction_logiciel_automatique.sh install
    http://$miroir/modules.zip \
        env      autoconf \
        --construire-dans-
src-dir \
        --root=${CHROOT}
    ${this_rep}/construction_logiciel_automatique.sh install
    http://$miroir/Python-3.8.0.tar.xz \
        env      openssl \
        libffi \
        zlib \
        --root=${CHROOT}
    ${this_rep}/construction_logiciel_automatique.sh install
    http://$miroir/curl-7.69.1.tar.gz \
        env      openssl \
        libffi \
        zlib \
        --root=${CHROOT}
    (cd ${CHROOT}
    wget http://$miroir/spack-0.13.3.tar.gz
    tar -xvf spack-0.13.3.tar.gz)
fi

source ${this_rep}/construction_logiciel_automatique.sh env openssl
libffi zlib curl --root=${CHROOT}

SPACK_DIR="${CHROOT}/spack-0.13.3"
MODULE_DIR="${CHROOT}/modules/prefix"
PYTHON="${CHROOT}/Python/prefix/bin/python3.8"
SPACK="${PYTHON} ${SPACK_DIR}/bin/spack"
PIP="${CHROOT}/Python/prefix/bin/pip3.8"

else

SPACK_DIR="/home/thibault/CEA/travail/CASTEM20/spack/spack-0.13.3"
MODULE_DIR="/home/thibault/CEA/travail/CASTEM20/modules/prefix"
PYTHON="python3"
SPACK="${PYTHON} ${SPACK_DIR}/bin/spack"
PIP="pip3"

fi

SPACK_CONF_DIR="${HOME}/.spack"
mkdir -p ${SPACK_CONF_DIR}/linux
${SPACK} repo add ${this_rep}/repo

#Spécifie machine :
if test `uname -s` = "Linux" ; then
    PLATEFORME="Linux"
else
    PLATEFORME="MAC"
fi
if [ `uname -m` == "x86_64" ] ; then
    BIT="64"
    GCC_CIBLE="9.2.0"
else
    BIT="32"
    GCC_CIBLE="6.5.0"
fi

if [ "${PLATEFORME}" == "Linux" ] ; then
    COMPILERS_CONFIG_FILE="${SPACK_CONF_DIR}/linux/compilers.yaml"
fi

THIS_REPO=$( cd "$( dirname "${BASH_SOURCE[0]}" )" && pwd )

cd ${THIS_REPO}

function erreur () {
    [ -n "$1" ] && THIS="$1"
    echo "Une erreur est survenue dans ${THIS}."
    exit 1
}

function get_os_name_from_spack {

${PYTHON} <<EOF
from __future__ import print_function

import os
import sys

# Allow spack libs to be imported in our scripts
spack_lib_path = "${SPACK_DIR}/lib/spack/"
sys.path.insert(0, spack_lib_path)

# Add external libs
spack_external_libs = os.path.join(spack_lib_path, 'external')
sys.path.insert(0, spack_external_libs)

if sys.version_info[:2] == (2, 6):
    sys.path.insert(0, os.path.join(spack_external_libs, 'py26'))

if 'ruamel.yaml' in sys.modules:
    del sys.modules['ruamel.yaml']

if 'ruamel' in sys.modules:
    del sys.modules['ruamel']

import spack.operating_systems.linux_distro
if sys.platform == 'darwin':
    print(spack.operating_systems.mac_os.MacOs())
elif sys.platform == 'linux':
    print(spack.operating_systems.linux_distro.LinuxDistro())
else:
    print('OS_NON_DEFINI')
EOF
}
```





```

spack load mpfr
spack load gmp

else
erreur
fi

cat <<EOF > ${COMPILERS_CONFIG_FILE}
compilers:
- compiler:
  spec: gcc@${compilateur_ver}
  paths:
    cc: ${prefix}/bin/gcc${suffix}
    cxx: ${prefix}/bin/g++${suffix}
    f77: ${prefix}/bin/gfortran${suffix}
    fc: ${prefix}/bin/gfortran${suffix}
  flags: {}
  operating_system: `get_os_name_from_spack`
  #target: `uname -m`
  modules: []
  environment: {}
  extra_rpaths: []
EOF

}

. ${SPACK_DIR}/share/spack/setup-env.sh
source ${MODULE_DIR}/init/bash

[ "${CASTEM_RELEASE}" == "VRAI" ] && spack_arg1="+castem_release" ||
spack_arg1=""

config_packages

function spack_load {
for arg in $@; do
  spack load ${arg} > /dev/null 2>&1 || return 1
done
}

function exit_subshell_si_travail_deja_effectue {
spack location --install-dir "$@" > /dev/null 2>&1
[ $? -eq 0 ] && exit 0
}

#fonction dans un subshell (( ... )) pour ne pas polluer l'environnement
function construction_gcc (){

  #A t-on vraiment besoin de travailler ?
  exit_subshell_si_travail_deja_effectue repo.gcc@${GCC_CIBLE}
%gcc@${GCC_INI}

  env_spack
  PATH=${HOME}/CASTEM2019/GCC/GCC-x86_64/bin:${PATH} config_gcc
"${GCC_INI}"
  nettoyage_correction

  spack_load isl@gcc@${GCC_INI} mpc@gcc@${GCC_INI} mpfr@gcc@${GCC_INI}
gmp@gcc@${GCC_INI}
  ${SPACK} install repo.gcc@${GCC_CIBLE} %gcc@${GCC_INI} ${spack_arg1}
}

#fonction dans un subshell (( ... )) pour ne pas polluer l'environnement
function construction_binutils (){

  #A t-on vraiment besoin de travailler ?
  exit_subshell_si_travail_deja_effectue binutils %gcc@${GCC_CIBLE}

  env_spack
  config_gcc "${GCC_CIBLE}" "${GCC_INI}"
  nettoyage_correction

  ${SPACK} install binutils %gcc@${GCC_CIBLE}
}

#fonction dans un subshell (( ... )) pour ne pas polluer l'environnement
function construction_gcc_bs (){

  #A t-on vraiment besoin de travailler ?
  exit_subshell_si_travail_deja_effectue repo.gcc@${GCC_CIBLE}
%gcc@${GCC_CIBLE}

  env_spack
  config_gcc "${GCC_CIBLE}" "${GCC_INI}"
  nettoyage_correction
  spack load binutils %gcc@${GCC_CIBLE}
}

${GCC_CIBLE} install repo.gcc@${GCC_CIBLE} %gcc@${GCC_CIBLE} ${spack_arg1}
}

#fonction dans un subshell (( ... )) pour ne pas polluer l'environnement
function construction_castem (){

  #A t-on vraiment besoin de travailler ?
  exit_subshell_si_travail_deja_effectue repo.castem@20
env_spack
config_gcc "${GCC_CIBLE}" "${GCC_CIBLE}"
nettoyage_correction

${SPACK} install repo.castem@20 \
${spack_arg1} \

#fonction dans un subshell (( ... )) pour ne pas polluer l'environnement
function construction_castem (){

  #A t-on vraiment besoin de travailler ?
  exit_subshell_si_travail_deja_effectue repo.castem@20

  #A t-on vraiment besoin de travailler ?
  exit_subshell_si_travail_deja_effectue repo.castem@20

  env_spack
  config_gcc "${GCC_CIBLE}" "${GCC_CIBLE}"
  nettoyage_correction

  ${SPACK} install repo.castem@20 \
  ${spack_arg1} \

#repertoire_a_repercuter_initial=/home/kk2000/Documents/CASTEM20/2020_07_01_INITIAL \
#repertoire_a_repercuter_correction="${CORRECTION REP}" \
#repertoire_a_repercuter_divers=/home/kk2000/Documents/CASTEM20/2020_07_01_INITIAL/divers \
#repertoire_final=/home/kk2000/Documents/CASTEM20/FINAL \
#repo.binutils@gcc@${GCC_CIBLE} \
#repo.med@4.0.0 \
#repo.openmpi@4.0.0 \
#repo.fxdr@20 \
#repo.esope@20 \
#repo.esope-precompiled@19 \
#repo.castem-bin@20 \
#tfel@3.3.0 +portable-build +castem -abaqus -aster -europlexus -cyrano -ansys -fortran -python -lsdyna -python_bindings
}

function installation_finale_castem_release {
#installation GCC Cast3M (release) :
CASTEM REP="${SPACK} location --install-dir castem
%gcc@${GCC_CIBLE} "
GNU_PATH="${CASTEM REP}/GCC"
if [ "$BIT" == "64" ]; then
mkdir "${GNU_PATH}"
GNU_PATH="${GNU_PATH}/GCC-x86_64"
else
mkdir "${GNU_PATH}"
GNU_PATH="${GNU_PATH}/GCC-i686"
fi
cp -r "${SPACK} location --install-dir gcc@${GCC_CIBLE}
%gcc@${GCC_CIBLE}" "${GNU_PATH}"
binutils_prefix=${SPACK} location --install-dir binutils
%gcc@${GCC_CIBLE} "
cp ${binutils_prefix}/bin/* ${GNU_PATH}/bin/
cp ${binutils_prefix}/lib/* ${GNU_PATH}/lib/
for lib in isl mpc mpfr gmp; do
  cp ${SPACK} location --install-dir ${lib}
%gcc@${GCC_CIBLE} "/lib/lib${lib}.so* ${GNU_PATH}/lib/
done

#il faut aussi la 'libintl' de gettext :
cp ${SPACK} location --install-dir gettext
%gcc@${GCC_CIBLE} "/lib/libintl.so* ${GNU_PATH}/lib/
#il faut aussi la 'libiconv' de libiconv :
cp ${SPACK} location --install-dir libiconv
%gcc@${GCC_CIBLE} "/lib/libiconv.so* ${GNU_PATH}/lib/

#On doit creer certains ZIP qui sont utilises par l'installateur
IzPack :
rep_a_zipper=(
  ${CASTEM REP}/mpi/mpi32
  ${CASTEM REP}/mpi/mpi64
  ${CASTEM REP}/GCC/GCC-x86_64
  ${CASTEM REP}/GCC/GCC-i686
  ${CASTEM REP}/MFRONT/mfront-x86_64
  ${CASTEM REP}/MFRONT/mfront-i686
)
for dir in ${rep_a_zipper[@]}; do
[ ! -d ${dir} ] && mkdir ${dir}
cd ${dirname ${dir}}
zip_dir=basename ${dir}
zip -r ${zip_dir}.zip ${zip_dir}
done

#repertoire 'doc' :
mkdir ${CASTEM REP}/doc
}

```

```
construction_gcc
if [ $? -ne 0 ]; then
  construction_gcc #2eme chance
  [ $? -ne 0 ] && erreur "construction_gcc_1"
fi

construction_binutils
[ $? -ne 0 ] && erreur "construction_binutils"

construction_gcc_bs
[ $? -ne 0 ] && erreur "construction_gcc_bs"
construction_castem
[ $? -ne 0 ] && erreur "construction_castem"
[ "${CASTEM_RELEASE}" == "VRAI" ] && installation_finale_castem_release
```



**Annexe F. Script environnement\_Cast3M20.bat et environnement\_Cast3M20**

## **1. Script environnement\_Cast3M20.bat**

```

REM Versions de HDF5 et MED
SET HDF5_VERSION=1.10.3
SET MED_VERSION=4.0.0

REM definition du titre de la fenetre de commande
TITLE Environnement de Developpement MinGW - %BIT%bits
EXIT /B 0

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

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

```

## 2. Script environnement\_Cast3M2020

```

#!/bin/bash
# Definition des variables d'environnement pour Cast3M

function get_gnu_suffix {
    f=$(${GNU_PATH}/bin/gfortran*)
    [ -n "${f[0]}" ] && {
        f=`basename "${f[0]}" 2> /dev/null`
        [ $? -eq 0 ] && {
            echo "${f:8}"
        }
    }
}

# Definition des variables d'environnement
if [ "%{GENERE_PAR_IZPACK}" == "VRAI" ]; then
    #On tente d'utiliser les variables definies par IzPack :
    [ -z "${CASTEM_VERSION}" ] && CASTEM_VERSION=%{XXANNEEXX}
    [ -z "${CASTEM_REVISION}" ] && CASTEM_REVISION=%{XXREVISIONXX}
    [ -z "${CASTEM_PLATEFORME}" ] && CASTEM_PLATEFORME=%{XXPLATEFORMEXX}
    [ -z "${CASTEM_REPO}" ] && CASTEM_REPO=%{INSTALL_PATH}
else
    #On tente de definir les variables nous meme :
    [ -z "${CASTEM_VERSION}" ] && CASTEM_VERSION=`f=$(basename ${BASH_SOURCE[0]})` && [[ ${f} =~ ([0-9]+)$ ]] && echo ${f: -2}`
    [ -z "${CASTEM_PLATEFORME}" ] && CASTEM_PLATEFORME=`[ $uname -s ] = "Linux" ] && echo Linux || echo MAC`
    [ -z "${CASTEM_REPO}" ] && CASTEM_REPO=$( cd "$( dirname "${BASH_SOURCE[0]}")/.." && pwd )"
fi

# Choix du Type de licence
if [ -d ${CASTEM_REPO}/licence ] || [ "$LICENCETYPE" == "INDUS" ] && [ ! "$LICENCETYPE" == "EDURE" ]; then
# Cas de la licence INDUSTRIELLE
    LICENCETYPE="INDUS"
else
# Cas de la licence EDUCATION - RECHERCHE
    LICENCETYPE="EDURE"
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"
    [ -z "${GNU_PATH}" ] && GNU_PATH=${CASTEM_REPO}/GCC/GCC-x86_64
    [ -z "${TFELHOME}" ] && export TFELHOME=${CASTEM_REPO}/MFRONT/mfront-x86_64
else
    BIT="32"
    [ -z "${GNU_PATH}" ] && GNU_PATH=${CASTEM_REPO}/GCC/GCC-i686
    [ -z "${TFELHOME}" ] && export TFELHOME=${CASTEM_REPO}/MFRONT/mfront-i686
fi
GNU_SUFFIX=`get_gnu_suffix`

# Repertoires lies a MPI (includes)
# Variable d'environnement definitissant la commande MPI : MPI_RUNCMD
[ -z "${OPAL_PREFIX}" ] && OPAL_PREFIX=${CASTEM_REPO}/mpi/mpi$BIT && export OPAL_PREFIX
[ -z "${MPI_LIBREP}" ] && MPI_LIBREP=${OPAL_PREFIX}/lib
[ -z "${MPI_EXCREP}" ] && MPI_EXCREP=${OPAL_PREFIX}/bin
[ -z "${MPI_INCREP}" ] && MPI_INCREP=${OPAL_PREFIX}/include
[ -x ${MPI_EXCREP}/mpirun ] && PATH=${MPI_EXCREP}:$PATH && export PATH

# Repertoire pour les librairies necessaires (pour l'execution ou l'édition des liens)
if [ "%{CASTEM_PLATEFORME}" == "Linux" ]; then
    export LD_LIBRARY_PATH=${CASTEM_REPO}/lib$BIT:${GNU_PATH}/lib:${GNU_PATH}/lib64:${MPI_LIBREP}:$(TFELHOME)/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_64-linux-gnu:/usr/lib/i386-linux-gnu"
elif [ "%{CASTEM_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 ${MPI_EXECREP}/mpirun ]; then
    export PATH=${MPI_EXECREP}:$PATH
fi

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

if [ -z "$GFORTRAN" ] ; then
    GFORBIN="FAUX"
else
    GFORBIN="VRAI"
fi

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

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

#Variables necessaires a MFRONT (compilcast) :
if [ -z "$CC" ] ; then
    export CC=${GNU_PATH}/bin/gcc${GNU_SUFFIX}
fi
if [ -z "$CXX" ] ; then
    export CXX=${GNU_PATH}/bin/g++${GNU_SUFFIX}
fi

# Ajout au PATH du repertoire contenant les binaires
export PATH=${CASTEM_REP}/bin:${GNU_PATH}/bin:${TFELHOME}/bin:$PATH

```

## Annexe G. Scripts compilcast20.bat et compilcast20

### 1. Script compilcast20.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_OPTIM_OPTIONS=-ftracer -fgcse-sm -fgcse-las -fopt-info -ftree-
loop-distribution -frename-registers -fassociative-math -freciprocal-
math -fno-signaling-nans -fno-signed-zeros -fno-trapping-math
SET F_MANDATORY_OPTIONS=-pthread -mtthreads -mfpmath=sse -mtune=generic -
ffpe-summary=invalid,zero,overflow -fopenmp -frecursive -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=
    REM SET F_INIT=-finit-logical=false -finit-character=32 -finit-
real=snan -finit-integer=2147483647
    SET F_INIT=-finit-logical=false -finit-character=32 -finit-real=snan -
finit-integer=-2147483648
) 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
REM SET F_INIT=-finit-logical=false -finit-character=32 -finit-real=snan
-finit-integer=-9223372036854775808
REM SET F_INIT=-finit-logical=false -finit-character=32 -finit-real=snan
-finit-integer=9223372036854775807
    SET F_INIT=-finit-logical=false -finit-character=32 -finit-real=snan -
finit-integer=-2147483648)

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

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

SET ESOPE_INC=%CASTEM_REPO%\include\eso
SET ESOPE_PARAM=NORME=TRADUCTEUR,FORT=WIN%BIT%,ESOPE=1Go

REM Initialisations :
SET FORTRAN_FILE=
SET LOG_FILE=
SET LISTE_SRC=
SET PAUSEFIN=
SET AIDE=
SET HELP=
SET PARALLELLE_COMP=FAUX
SET DELETE_BALISE=FAUX
SET NOM_BALISE=
SET SUFFIXI=
SET FILE_LIST=
SET MFRONT_DEBUG=
SET EXIT_CMD=/B
SET /A ERROLEV=0
SET /A NBCPU=1
SET lacle=
SET lasource=

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!=ESOPPARAM!,CONTROL
            SET OPTIONS_SCRIPT!=OPTIONS_SCRIPT! %ArgNAME1%
            SET SUFFIXI=_c
            SHIFT
            GOTO DEBUT_LECTURE
        )

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

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

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



```

GOTO DEBUT_LECTURE
)

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

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

REM compilation avec la couverture de code GCOV
IF "%ArgNAME1%"=="--gcov" (
    SET OPTIONS_SCRIPT!=OPTIONS_SCRIPT! %ArgNAME1%
    SET F_OPTIONS!=F_OPTIONS! -fprofile-arcs -ftest-coverage
    SET FORTRAN_FILE=VRAI
    SHIFT
    GOTO DEBUT_LECTURE
)

REM compilation du traducteur ESOPE
IF "%ArgNAME1%"=="-ESOPE" (
    SET ESOPE_INC=%ESOPE_REPO%\include
    SET OPTIONS_SCRIPT!=OPTIONS_SCRIPT! %ArgNAME1%
    IF EXIST %esopX_Local% (
        SET ESOPEEXEC="%esopX_Local%"
    ) ELSE (
        SET
        ESOPEEXEC="%CASTEM_REPO%\bin\bin_esope_Win_%BIT%_%CASTEM_VERSION%.exe"
    )
    REM retrait des doubles cotes
    SET ESOPEEXEC!=ESOPEEXEC:-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%
    SET /A ERROLEV=1
    GOTO SORTIE_SCRIPT
) ELSE (
    SET /A NBCPU=%ArgNAME1:~2,2%
    IF "!NBCPU!"=="0" (
        ECHO .
        ECHO %LIGNEUP%
        ECHO ° le nombre de processus est incorrect : %ArgNAME1:~0,4%
    .
    ECHO %LINEDOWN%
    SET /A ERROLEV=1
    GOTO SORTIE_SCRIPT
)
)
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%"=="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
pour Cast3M
    ECHO ° Extension traitees : .c .eso .f .F .mfront
    .
    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 ° 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 ° --gcov : Compilation pour analyser la couverture de code
    .
    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 activee 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
.txt
    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 ° 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 écrit par Clement BERTHINIER
.
ECHO °
.
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 %LINEDOWN%
SET /A ERROLEV=1
GOTO SORTIE_SCRIPT
)
:LABEL_HELP
IF DEFINED HELP (
REM Affiche l'aide en Français
ECHO .
ECHO %LIGNEUP%
ECHO °NAME
.
ECHO °      compilcast%CASTEM_VERSION%`%SPY% : Compilation of sources for
Cast3M
ECHO °      Compatible extensions: .c .eso .f .F .mfront
.
ECHO °      Web site : http://www-cast3m.cea.fr/
.
ECHO °
.
ECHO %LIGNEMIDDLE%
ECHO °VERSION
.
ECHO °      Script version : %CASTEM_VERSION%`%SPY%.%CASTEM_REVISION%
.
ECHO °
.
ECHO %LIGNEMIDDLE%
ECHO °SYNTAX
.
ECHO °      compilcast%CASTEM_VERSION%`%SPY% [OPTION]... [FILE_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 °      --gcov : Compilation for code coverage analysis
.
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 .txt 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 °
.
ECHO %LIGNEMIDDLE%
ECHO °EXAMPLES
.
ECHO °      compilcast%CASTEM_VERSION%`%SPY% file1.eso file2.eso *.c
.
ECHO °      Compile sources normally
.
ECHO °
.
ECHO °      compilcast%CASTEM_VERSION%`%SPY% -d f*.eso file2.c
.
ECHO °      Compile sources with debug mode
.
ECHO °
.
ECHO °      compilcast%CASTEM_VERSION%`%SPY% -d -c *.eso *.c
.
ECHO °      Compile sources with debug mode and CONTROL for Esope
.
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 'essaicast%CASTEM_VERSION%`%SPY% :'
'essaicast%CASTEM_VERSION% --help`%SPY%
ECHO °
.
ECHO %LINEDOWN%
SET /A ERROLEV=1
GOTO SORTIE_SCRIPT
)

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 ménage avant de commencer (Lance en parallèle on ne
supprime pas)
IF DEFINED EXIT_CMD (
    IF EXIST "ZZZ_Erreur_Compilation%SUFFIX1%.txt" DEL
"ZZZ_Erreur_Compilation%SUFFIX1%.txt">>nul
ECHO.)

REM Calcul du nombre de sources
SET /A srccompt=0

REM Recopie du contenu de !FILE_LIST! dans OPT1 parce que le "FOR" ne
supporte pas les "!" pour la première option
SET OPT1=!FILE_LIST!
FOR %OPT1% %%i IN (!LISTE_SRC!) DO (
    SET /A srccompt+=1
)

SET /A src_restantes=%srccompt%

REM Préparation pour la traduction / compilation en parallèle
REM En raison d'une limitation de la taille des variables
d'environnement en BATCH, on ne peut pas dépasser 750 sources par
process en parallèle

SET /A MAX_SRC_LIST=25
SET /A JOB_LOC=0
SET /A srccnum=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 vérification du travail en parallèle
IF !NBCPU! GTR 1 (
    SET PARALLEL_COMP=VRAI
) ELSE IF !NBCPU! EQU 1 (

```



```

SET PARALLEL_COMP=FAUX
) ELSE (
SET PARALLEL_COMP=FAUX
ECHO.
ECHO %LIGNEUP%
ECHO ° Le nombre de processus apres -P est invalide
.

ECHO %LINEDOWN%
SET /A ERROLEV=1
GOTO SORTIE_SCRIPT)

IF "%PARALLEL_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=1
) ELSE (
  SET /A proccour+=1
)

REM Compilation des listes en parallel
SET /A NBCPU!=procmaxi!
FOR /L %%i IN (1, 1, !procmaxi!) DO (
ECHO %%i>compilcast_PARA_BALISE%%i.txt
START "COMPILEATION" /SEPARATE /REALTIME 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 NOT "!nbwait!"=="0" (
  TIMEOUT 2 /NOBREAK >nul
  GOTO DEBUT_ATTENTE
)
GOTO BILAN_COMPILED

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 srcrenum+=1
    ECHO !NomF2!
    TITLE Compilation - %BIT%bits : !NomF2!
    ECHO Sources traitees : !srcrenum!/%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" (
  IF EXIST "%NOM_BALISE%" DEL "%NOM_BALISE%"
  IF EXIST "!LISTE_SRC!" DEL "!LISTE_SRC!"
)

:BILAN_COMPILED
REM Bilan des erreurs de compilation s'il y en a eu
IF EXIST ZZZ_Erreur_Compilation%SUFFIX1%.txt (
  IF %ERROLEV% EQU 0 SET /A ERROLEV=20
  ECHO %LIGNEUP%
  ECHO °           Liste des echecs de compilation
.

  ECHO %LIGNEMIDDLE%
  TYPE ZZZ_Erreur_Compilation%SUFFIX1%.txt
  ECHO %LINEDOWN%
) ELSE (
  ECHO %LIGNEUP%
    ECHO °           Compilation des sources reussie
.
  ECHO %LINEDOWN%)

EXIT %EXIT_CMD% SORTIE_SCRIPT

```

---

```

REM
_____
:COMPIL_UN_SOURCE
SET NomF=%~n1
SET NomF1=%~1
SET NOMF3="%NOMF2%" "
SET /A ERROLEV=0

REM Nettoyage des eventuels fichiers de compilation
IF EXIST "%NomF%.o" DEL "%NomF%.o"
IF EXIST "%NomF%.txt" DEL "%NomF%.txt"

IF "%NomF1%"=="%NomF%.c" (
  REM Compilation du fichier .
  CALL %CC% -v -Wall -m%BIT% %COMP_OPTIM% -DWIN32 -DWIN%BIT% -
D_REENTRANT -DFOR_WINDOWS -DONCRPCDLL -I"%CASTEM_REPO%\include\c" -I. -c
"%NomF%.c > "%NomF%.txt 2>&1
  REM -DFOR_WINDOWS : Pour la compilation de perm.c qui gère la version
industrielle de Cast3M
  REM -DONCRPCDLL : Pour la compilation des .c de XDR
(/u2/castem/win64/)

) ELSE IF "%NomF1%"=="%NomF%.eso" (
  REM Verification de la CLE d'évolution
  IF EXIST "%CASTEM_REPO%\fic.cle" (
    REM FOR /f "delims=" %%n in ('CALL head -1 "%NomF%.eso"
^| CALL grep -w SOURCE ^| CALL cut -c 58-62 2>nul') do SET lasource=%%n
    REM FOR /f "delims=" %%n in ('CALL grep -wi "%NomF%" ^
"%CASTEM_REPO%\fic.cle" ^| CALL grep -w SOURCE ^| CALL cut -c 58-62
2>nul') do SET lacle=%%n

    REM dans le cas des nouvelles sources ou de defdat.eso il n'y a pas
les entetes, il faut donc prendre des precautions
    IF "%!lacle!:]==:" (
      SET lacle=0
      SET lasource=0
    )
    IF "%!lasource!:]==:" (
      SET lasource=0
    )

    REM Conversion des chaines en nombres !
    SET /A lacle!=lacle!
    SET /A lasource!=lasource!
    IF !lasource! LSS !lacle! (
      ECHO ° %NOMF3:~1,18% CLE INCORRECTE : SOURCE ANTIDATEE
    ) >> ZZZ_Erreur_Compilation%SUFFIX1%.txt
    SET /A ERROLEV=99
    GOTO SORTIE_SCRIPT
  )
)

REM Nettoyage des eventuels fichiers existants
IF EXIST "%NomF%.lst" DEL "%NomF%.lst"
IF EXIST "%NomF%.f"   DEL "%NomF%.f"
IF EXIST "%NomF%.s"   DEL "%NomF%.s"

SET ESOPC_OUT=%NomF%.f
SET ESOPC_LST=%NomF%.lst
CALL "%ESOPCEXEC%" < "%NomF%.eso"
SET /A ERROLEV!=ERRORLEVEL!
ECHO ERRORLEVEL !ERROLEV!>>"%NomF%.lst"

REM Verification du niveau d'erreur
IF "%!ERROLEV!:]==0" (
  REM Compilation du fortran
  IF EXIST "%NomF%.lst" DEL "%NomF%.lst"
:COMPIL_F AGAIN
  CALL %FC% -v -Wall -m%BIT% %COMP_OPTIM% %F_OPTIONS% -c "%NomF%.f" -
-I"%CASTEM_REPO%\include\esof" "%NomF%.txt" 2>&1
  SET /A ERROLEV!=ERRORLEVEL!
  ECHO ERRORLEVEL !ERROLEV!>>"%NomF%.txt"
REM En compilation parallele, parfois on a des "f951.exe: out of
memory allocating XXXXX bytes", je refais la tentative de compilation
  IF !ERROLEV! EQU 1 GOTO COMPIL_F AGAIN
  IF EXIST "%NomF%.s" CALL %AS% "%NomF%.s" -o "%NomF%.o"
    IF EXIST "%NomF%.o" IF NOT DEFINED FORTRAN_FILE DEL "%NomF%.f"
  ) ELSE (
    ECHO ° %NOMF3:~1,18% NON TRADUIT
  ) >> ZZZ_Erreur_Compilation%SUFFIX1%.txt
  GOTO SORTIE_SCRIPT
) 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%.txt" 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%.txt" 2>&1

) ELSE IF "%NomF1%"=="%NomF%.mfront" (
    REM Compilation du fichier .mfront
    CALL %TFELHOME%\bin\mfront --interface=castem --obuild
%MFRONT_DEBUG% "%NomF%.mfront"
    GOTO SORTIE_SCRIPT

REM Cas des fichiers qui ne sont pas .eso ou .c
    ) ELSE (
        REM Verification si le binaire .o a ete genere
        IF NOT EXIST "%NomF%.o" (
            ECHO ° %NOMF3:~1,18% NON COMPILE
            ° >> ZZZ_Erreur_Compilation%SUFFIX1%.txt
            SET /A ERROLEV=20
        ) ELSE (
            REM Suppression des fichiers inutiles
            IF EXIST "%NomF%.txt" (
                IF NOT DEFINED LOG_FILE      DEL "%NomF%.txt"
            ) )
        :SORTIE_SCRIPT
        EXIT /B !ERROLEV!
    )
)

```

## 2. Script compilcast20

```

#!/bin/bash

#Recuperation des variables d'environnement
[ -z "${CASTEM_VERSION}" ] && CASTEM_VERSION=`f=${basename
${BASH_SOURCE[0]}} && [[ "${f}" =~ ([0-9]+)$ ]] && echo ${f: -2}`` 
[ -z "${CASTEM_REPO}" ] && CASTEM_REPO="$( cd "$("dirname
"${BASH_SOURCE[0]}")/.." && pwd )"
source ${CASTEM_REPO}/bin/environnement_Cast3M${CASTEM_VERSION}

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

# Traducteur ESOPE (possibilite de le surcharger par la variable
d'environnement "ESOPE_EXEC")
if [ -z "${ESOPE_EXEC}" ]; then
  if [ -x esop_${BIT}_${CASTEM_VERSION} ]; then
    ESOPE_EXEC="./esop_${BIT}_${CASTEM_VERSION}"
  else
    ESOPE_EXEC="${CASTEM_REPO}/bin/bin_esope_${CASTEM_PLATEFORME}_${B
IT}_${CASTEM_VERSION}"
  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
BAD_ARG="FAUX"          # Si VRAI permet d'afficher une sortie avec
erreur d'argument
ESOPE_TRAD="FAUX"       # En cas de fabrication de la librairie
ESOPE
NBCPU="1"
MFRONT_DEBUG=
COMP_GENE="-c"          # On transforme la source en '.o' par
defaut
F_OPTIONS=""             # Options de compilation FORTRAN
let RETURN=0              # Valeur retournee par le script
ETOILE="*****"
*****"

# Suppression des fichiers issus d'une precedente compilation
if [ "${PARALLEL_COMP}" != "CHILD" ]; then
  [ -f ZZZ_Non_Compile.txt ] && rm -f ZZZ_Non_Compile.txt
  [ -f ZZZ_Non_traduit.txt ] && rm -f ZZZ_Non_traduit.txt
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_MFRONT_1=""
  LISTE_BAD_ARG=""
  for i in "$@"; do
    # "--aide" ==> affichage de l'aide en Francais
    # "--help" ==> affichage de l'aide en Anglais

```

```

    # "--gcov" ==> Compilation pour analyser la couverture de
code
    # "-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 .txt
    # "-Pn"     ==> Compilation des sources en parallele sur n
processus
    # "-S"      ==> Code Assembleur de la source en sortie
    # "-ESOPE"   ==> Compilation du traducteur ESOPE [SEMT/LM2S
seulement]

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

      case ${i} in
        '--aide') AIDE="VRAI";break;;
        '--help') HELP="VRAI";break;;
        '--gcov') F_OPTIONS="-fprofile-arcs -ftest-coverage";
        FORTRAN_FILE="VRAI"; OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";
        '-c')
        CONTROLE_MODE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";
        '-d') COMP_OPTIM="-g"
        FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT}
${i}";MFRONT_DEBUG="--debug";
        '-cd') CONTROLE_MODE="VRAI";COMP_OPTIM="-g"
        FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT}
${i}";MFRONT_DEBUG="--debug";
        '-f')
        FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";
        '-log') LOG_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT}
${i}";
        '-ESOPE')
        ESOPE_TRAD="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";
        '-S') COMP_OPTIM="--ggdb -fverbose-asm";COMP_GENE="-
S";FORTRAN_FILE="VRAI";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";
        '-P') if [ ! "$i" = ~^-[0-9]+$ ]; then
          echo 'Il manque le nombre de processus apres -
P. Exemple : -P4'
          LISTE_BAD_ARG="${LISTE_BAD_ARG} ${i}";
        fi
        NBCPU=`echo ${i} | cut -c 3-10`;
        '-O')
        COMP_OPTIM="${i}";OPTIONS_SCRIPT="${OPTIONS_SCRIPT} ${i}";
        "${REP}`basename ${i} .eso`.eso"
        LISTE_ESO_1="${LISTE_ESO_1} ${i}";
        "${REP}`basename ${i} .f`.f"      LISTE_F_1="${LISTE_F_1}
${i}";
        "${REP}`basename ${i} .c`.c"      LISTE_C_1="${LISTE_C_1}
${i}";
        "${REP}`basename ${i} .mfront` .mfront"
        LISTE_MFRONT_1="${LISTE_MFRONT_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${CASTEM_VERSION} : Compilation des
sources pour Cast3M"
        echo "  Extension traitees : .c .eso .f .F .mfront"
        echo "  Site web : http://www-cast3m.cea.fr/"
        echo ''
        echo "VERSION"
        echo "  Version du Script :
${CASTEM_VERSION}.${CASTEM_REVISION}"

```

```

echo ''
echo "SYNTAXE"
echo "    compilcast${CASTEM_VERSION} [OPTION]...
[LISTE_FICHIER]..."
echo ''
echo "DESCRIPTION"
echo "    --aide : Affiche le manuel de cette commande en
Français"
echo "    --help : Affiche le manuel de cette commande en
Anglais"
echo "    --gcov : Compilation pour analyser la couverture
de code"
echo "    -c : Option CONTROLE d'Esope"
echo "    -d : Option de compilation en mode debug"
echo "    -cd : Option CONTROLE et mode debug"
echo "    -On : Optimisations activees
n=(s,g,fast,0,1,2,3)"
echo "    -f : Ne supprime pas le fichier fortran77
issu de la traduction"
echo "        des sources en Esope"
echo "    -log : conserve la sortie du compilateur dans
un fichier .txt"
echo "    -Pn : Compilation des sources en parallele
sur n processus"
echo "    -S : Code Assembleur de la source en sortie"
# echo "    -ESOPE : Compilation du traducteur ESOPE
[SEMT/LM2S seulement]"
echo ''
echo "EXEMPLES"
echo "    compilcast${CASTEM_VERSION} fichier1.eso
fichier2.eso fichier3.c"
echo "        Compile les 3 sources normalement"
echo ''
echo "    compilcast${CASTEM_VERSION} -d fichier1.eso
fichier2.eso fichier3.c"
echo "        Compile les 3 sources en mode debug"
echo ''
echo "    compilcast${CASTEM_VERSION} -d -c fichier1.eso
fichier2.eso fichier3.c"
echo "        Compile les 3 sources en mode debug avec
Esope CONTROLE activee"
echo ''
echo "    compilcast${CASTEM_VERSION} *.eso *.c"
echo "        Compile les toutes les sources Esope et C du
repertoire"
echo ''
echo "AUTEUR"
echo "    Script ecrit par Clement BERTHINIER"
echo ''
echo "VOIR AUSSI"
echo "    Aide du Script 'castem${CASTEM_VERSION}' ==>
'castem${CASTEM_VERSION} --aide'"
echo "    Aide du Script 'essaicast${CASTEM_VERSION}' ==>
'essaicast${CASTEM_VERSION} --aide'"
echo "${ETOILE}"
echo ''

else
    # Affiche l'aide du script en Anglais
echo ''
echo "${ETOILE}"
echo "NAME"
echo "    compilcast${CASTEM_VERSION} ==> Compilation of
Esope (.eso) and C sources"
echo "    Compatible extensions: .c .eso .f .F .mfront"
echo "    Web site : http://www-cast3m.cea.fr/"
echo ''
echo "VERSION"
echo "    Script Version :
${CASTEM_VERSION}.${CASTEM_REVISION}"
echo ''
echo "SYNTAX"
echo "    compilcast${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 "    --gcov : Compilation for code coverage analysis"
echo "    -c : CONTROLE Option for Esope"
echo "    -d : Debug mode compilation option"
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 "                from Esope sources"
echo "    -log    : Keep the output of the compiler in a
.txt file"
echo "    -Pn     : Parallel compilation of sources whith n
process"
echo "    -S      : Assembler Code of the source file as
output"
# echo "    -ESOPE  : Compilation ESOPE translator
[SEMT/LM2S only]"
echo ''
echo "EXAMPLES"
echo "    compilcast${CASTEM_VERSION} fichier1.eso
fichier2.eso fichier3.c"
echo "        Compile the 3 sources normally"
echo ''
echo "    compilcast${CASTEM_VERSION} -d fichier1.eso
fichier2.eso fichier3.c"
echo "        Compile the 3 sources with debug mode"
echo ''
echo "    compilcast${CASTEM_VERSION} -d -c fichier1.eso
fichier2.eso fichier3.c"
echo "        Compile the 3 sources with debug mode and
Esope CONTROLE mode"
echo ''
echo "    compilcast${CASTEM_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${CASTEM_VERSION}' script ==>
'castem${CASTEM_VERSION} --help'"
echo "    Manual for 'essaicast${CASTEM_VERSION}' script ==>
'essaicast${CASTEM_VERSION} --help'"
echo "${ETOILE}"
echo ''
fi

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

    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${CASTEM_VERSION} --aide ''"
    echo " To learn more      : "
    compilcast${CASTEM_VERSION} --help ''
    echo "${ETOILE}"
    let RETURN=1
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)      "
    let RETURN=1
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)      "
    let RETURN=1
fi

```

# NOTE DE FABRICATION DE CAST3M 2020

```

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

# Teste si les fichiers .mfront contenus dans les listes existent
LISTE_MFRONT_2=""
for i in ${LISTE_MFRONT_1}; do
  if [ ! -f ${i} ]; then
    echo "Fichier inexistant / Non existing file : '${i}'"
  else
    LISTE_MFRONT_2="${LISTE_MFRONT_2} ${i}"
    LISTE_SRC="${LISTE_SRC} ${i}"
  fi
done

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

if [ "$PARALLEL_COMP" = "FAUX" ] || [ "$PARALLEL_COMP" = "CHILD" ]; 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 -reciprocal-math -ftree-loop-distribution -fno-trapping-
math -fno-aggressive-loop-optimizations -finit-real=nan -finit-
integer=-2147483647 -finit-logical=false -frename-registers -
frecursive -fopenmp ${F_OPTIONS}"
    INCLUDE_MPI="-I${MPI_INCREP}"
    if test ${CASTEM_PLATEFORM} = "MAC" ; then
      F_OPTIONS="-v -Wall -m${BIT} ${COMP_OPTIM} -fdefault-
integer-8 -fdefault-real-8 -fdefault-double-8 -fopt-info -
ftracer -pthread -mtune=generic -mfpmath=sse -fgcse-sm -fgcse-
las -reciprocal-math -ftree-loop-distribution -fno-trapping-
math -fno-aggressive-loop-optimizations -finit-real=nan -finit-
integer=-2147483647 -finit-logical=false -frename-registers -
frecursive -fopenmp ${F_OPTIONS}"
      INCLUDE_MPI=
    fi
  else
    F_OPTIONS="-v -Wall -m${BIT} ${COMP_OPTIM} -fopt-info -
ftracer -pthread -mtune=generic -march=pentium4 -msse -
mfpmath=sse -fgcse-sm -fgcse-las -reciprocal-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
${F_OPTIONS}"
    INCLUDE_MPI="-I${MPI_INCREP}"
  fi

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

if [ "${LISTE_ESO_2}" != "" ] ; then
  if [ "$GFORBIN" != "FAUX" ] ; then
    # Variables d'environnement pour Esope
    if [ "${CONTROLE_MODE}" == "VRAI" ]; then
      export
    else
      export
    fi
    ESOPE_PARAM="NORME=TRADUCTEUR,FORT=UNIX${BIT},ESOPE=10000000,CON-
TROL"
  else
    # 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
    repertoire courant
    nb_INC=`( \ls -1 *.INC 2>/dev/null | wc -l )`

    # Menage d'une compilation precedente
    for i in ${LISTE_ESO_2} ; do
      i_base=`basename ${i}.eso`
      [ -f ${i_base}.o ] && rm -f ${i_base}.o
      [ -f ${i_base}.f ] && rm -f ${i_base}.f
      [ -f ${i_base}.lst ] && rm -f ${i_base}.lst
      [ -f ${i_base}.txt ] && rm -f ${i_base}.txt
    done

    # Compilations des ".eso"
    COMPT=0

    # Repertoire des includes pour le traducteur ESOPE
    # (possibilite 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
      ${FFORTRAN_EXEC} < ${i}
      CODE_RETOUR_ESOPE=$?

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

      if [ ${CODE_RETOUR_ESOPE} = '0' ]; then
        [ -f ${i_base}.lst ] && rm -f ${i_base}.lst
        ${FFORTRAN} ${F_OPTIONS} ${COMP_GENE} ${F_OPTIONS_associative} ${i_base}.f ${INCLUDE_MPI} >
${i_base}.txt 2>&1
        #Assembleur "-S" :
        if [ -f ${i_base}.s ]; then
          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}.txt
    grep -i -nB 4 'Error:' ${i_base}.txt
    echo
    echo ${i_base}.f ' Non compile' >>
ZZZ_Non_Compiler.txt
else
    # Nettoyage des fichiers
    if [ "${FORTRAN_FILE}" != "VRAI" ] ; then
        rm -f ${i_base}.f
    fi
    if [ "${LOG_FILE}" == "FAUX" ] ; then
        rm -f ${i_base}.txt
    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.txt
fi

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

done
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 precedente
        for i in ${LISTE_F_2} ; do
            i_base=`basename $i .f`
            [ -f ${i_base}.o ] && rm -f ${i_base}.o
            [ -f ${i_base}.txt ] && rm -f ${i_base}.txt
        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
            #differentes :
            if [ ${i_base} == "ddot2" ] ; then
                F_OPTIONS_associative=""
            else
                F_OPTIONS_associative="-fassociative-math -fno-
signed-zeros"
            fi

            ${GFORTRAN} ${F_OPTIONS} ${COMP_GENE}
            ${F_OPTIONS_associative} $i ${INCLUDE_MPI} > ${i_base}.txt 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}.txt
                grep -i -nB 4 'Error:' ${i_base}.txt
                echo
            fi
        done
    fi
fi

```

```

        echo ${i_base}.f ' Non compile' >>
ZZZ_Non_Compiler.txt
else
    # Nettoyage des fichiers
    if [ "${LOG_FILE}" == "FAUX" ] ; then
        rm -f ${i_base}.txt
    fi
fi
#####
COMPT=$((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 )`

    # Menage d'une compilation precedente
    for i in ${LISTE_C_2} ; do
        i_base=`basename $i .c`
        [ -f ${i_base}.o ] && rm -f ${i_base}.o
        [ -f ${i_base}.txt ] && rm -f ${i_base}.txt
    done

    # Compilations des ".c"
    COMPT=0
    # Sur Yosemite, le compilateur GCC ne permet des
    optimisation que de niveau -O3 au max
    if test ${CASTEM_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`

        ${GCC} -Wall -m${BIT} ${COMP_OPTIM} ${CASTEM_DEFINE} -
        DDATE_ONLY -DFOR_LINUX ${CASTEM_INC} ${COMP_GENE} $i >
        ${i_base}.txt 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}.txt
            grep -i -nB 4 'Error:' ${i_base}.txt
            echo
            echo ${i_base}.c ' Non compile' >> ZZZ_Non_Compiler.txt
        else
            if [ "${LOG_FILE}" == "FAUX" ] ; then
                rm -f ${i_base}.txt
            fi
            rm -f `basename $i .c`.txt
        fi

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

    done
fi
#####
# Traite la liste des ".mfront"
#####
if [ "${LISTE_MFRONT_2}" != "" ] ; then
    # Calcul du nombre de fichier ".mfront" à traiter
    nb_mfront=`( \ls -1 ${LISTE_MFRONT_2} | wc -l )`
```



```

# Compilations des ".mfront"
COMPT=0
for i in ${LISTE_MFRONT_2} ; do
    # Compilations des .mfront
    echo
    echo $i

    ${TFELHOME}/bin/mfront --interface=castem --obuild
${MFRONT_DEBUG} $i

    CODE_RETOUR_MFRONT=$?
    if [ ${CODE_RETOUR_MFRONT} -eq 1 ]; then
        echo Erreur de compilation
        echo
        echo $i ' Non compile' >> ZZZ_Non_Compiled.txt
    fi

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

done
fi

else # "$PARALLELLE_COMP" = "VRAI"

j=1
export PARALLELLE_COMP="CHILD"
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)/${NBCPU})) ] ;
then
        compilcast${CASTEM_VERSION} ${OPTIONS_SCRIPT}
${LISTE_SRC_TEMP} &
        LISTE_SRC_TEMP=""
        let j++
    fi
    LISTE_SRC_TEMP="$LISTE_SRC_TEMP ${LISTE_SRC_ARR[$i]}"
done
compilcast${CASTEM_VERSION} ${OPTIONS_SCRIPT}
${LISTE_SRC_TEMP} &
wait
export PARALLELLE_COMP=
PARALLELLE_COMP="VRAI"

fi

if [ "$PARALLELLE_COMP" = "FAUX" ] || [ "$PARALLELLE_COMP" =
"VRAI" ]; then

#~ echo "ici 2 !, PWD=`pwd`"
#~ echo "ici 2 !, LS=`ls`"

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

if [ -f ZZZ_Non_traduit.txt ];then
    echo ''
    echo "${ETOILE}"
    echo 'Erreurs de traduction : '
    echo ''
    cat ZZZ_Non_traduit.txt
    echo ''
    echo ''
    echo "consultez les fichiers '.lst'"
    echo "${ETOILE}"
    let RETURN=1
fi

if [ -f ZZZ_Non_Compiled.txt ]; then
    echo ''
    echo "${ETOILE}"
    echo 'Erreurs de compilation : '
    echo ''
    cat ZZZ_Non_Compiled.txt
    echo ''
    echo ''
    echo "consultez les fichiers '.txt'"
    echo "${ETOILE}"
    let RETURN=1
fi

```

---

```

    fi

    if [ ${SAIDE} == "FAUX" ] && [ ${HELP} == "FAUX" ] ; then
        if [ ${RETURN} -eq 0 ] ; then
            echo ''
            echo "${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

#####
# Verification par ftnchek
#####

if [ -f ZZZ_ftnchek.txt ]; then
    rm -f ZZZ_ftnchek.txt; fi

if [ -x ${CASTEM_REP}/bin/ftnchek-${BIT}bits ]; then
    # Calcul du nombre de fichier .f presents dans le
repertoire courant
    \ls -l *.f > /dev/null 2>&1
    CODE_RETOUR=$?

    if [ ${CODE_RETOUR} -eq 0 ] ; then

        if [ "${CASTEM_PLATEFORME}" == "Linux" ]; then
            INCLUDE_MPI="-include=${MPI_INCREP}"
        elif [ "${CASTEM_PLATEFORME}" == "MAC" ]; then
            INCLUDE_MPI=
        fi
        ${CASTEM_REP}/bin/ftnchek-${BIT}bits ${INCLUDE_MPI} -
nopretty -notruncation -crossref=calls -errors=100 -
f77=character,continuation,cray-pointer,format-edit-
descr,implicit-none,intrinsic,long-name,mixed-expr,long-
line,quotemark,rellops,semicolon,statement-order -brief -wrap=500
-calltree *.f > ZZZ_ftnchek_1.txt

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

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

```

```

        echo ""
>> ZZZ_ftnchek.txt
    fi

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

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

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

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

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

exit ${RETURN}

```

## Annexe H. Scripts essaicast20.bat et essaicast20

### 1. Script essaicast20.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 noms a 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 VERBOSE=
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 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 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 VERBOSE=-v
    SHIFT
    GOTO DEBUT_LECTURE
)

REM Conservation du .txt a 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 Francais
    ECHO.
    ECHO %LIGNEUP%
    ECHO °NAME
    .
    ECHO °      essaicast%CASTEM_VERSION%%SPY% : Procede a
    l'édition des liens
    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 °      essaicast%CASTEM_VERSION%%SPY% [OPTION]
    .
    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 °      -v     : invoque le linker en mode verbose
    .
    ECHO °      -log   : conserve le .txt a 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 °
    .
    ECHO %LIGNEMIDDLE%
    ECHO °EXAMPLES
    .
    ECHO °      essaicast%CASTEM_VERSION%%SPY%
    .
    ECHO °      Realise l'édition des liens dans le
    repertoire courant et
    ECHO °      produit un executable binaire
    cast_%BIT%_%CASTEM_VERSION%.exe%SPY%
    ECHO °
    .
    ECHO °      En cas d'échec un fichier
    link_cast_%BIT%_%CASTEM_VERSION%.txt%SPY% est généré
)

```

```

ECHO °
.
ECHO %LIGNEMIDDLE%
ECHO °AUTEUR
.
ECHO °      Script écrit par Clement BERTHINIER
.
ECHO °
.
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 Francais
ECHO .
ECHO %LIGNEUP%
ECHO °NAME
.
ECHO °      essaicast%CASTEM_VERSION%%SPY% : Proceed to the
link           .
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 °      essaicast%CASTEM_VERSION%%SPY%
.
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 °      -v      : invoque the linker in verbose mode
.
ECHO °      -log     : keep the .txt 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   : Librairy and ESOPE Translator
[SEMT/LM2S only] .
ECHO °
.
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%.txt%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 %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%
)

IF NOT DEFINED ESOPE GOTO :CAST3M_SOURCES
REM
*****
REM *                                         Cas de
la Librairie ESOPE et du Traducteur
*
REM
*****
SET esopX=bin_esope_Win_%BIT%_%CASTEM_VERSION%
SET LIBESOPE=libesope_%BIT%$SUFFIX1%.a
SET esopX_Local=esop_%BIT%_%CASTEM_VERSION%
IF EXIST "%esopX_Local%.exe" DEL /Q "%esopX_Local%.exe"

REM Creation de la librairie
ECHO .
ECHO %LIGNEUP%
ECHO °      Mise a jour de la librairie %LIBESOPE%
.
ECHO %LIGNEDOWN%
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 l edition des liens
CALL %FC% %VERBOSE% -Wall -static -static-libgfortran -static-
libgcc -m%BIT% -o %esopX_Local%.o %ESOPE_REP%\lib\%LIBESOPE%
-lmingwthrd > link_%esopX_Local%.txt 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 %LIGNEDOWN%
IF EXIST "link_%esopX_Local%.txt" DEL /Q
"link_%esopX_Local%.txt"

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 %LIGNEDOWN%
) ELSE (
ECHO .
ECHO %LIGNEUP%
ECHO °      Mise a jour de l'executable %esopX%.exe
.
ECHO %LIGNEDOWN%
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 .

```

# NOTE DE FABRICATION DE CAST3M 2020

```

ECHO %LIGNEUP%
ECHO ° ECHEC durant l'Edition des liens :
°

IF EXIST "link_%esopX_Local%.txt" (
    ECHO ° Consultez le fichier link_%esopX_Local%.txt
%SPY%
    ECHO %LIGNEDOWN%
) )

EXIT /B !ERROLEV

:CAST3M_SOURCES
REM ****
REM *
Cas de Cast3M
*
REM ****
REM Repertoire ou trouver la licence
SET DIRLIC=%CASTEM_REPO%\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_REPO%\lib%BIT%\%LIBCASTEM%" (
    ECHO .
    ECHO %LIGNEUP%
    ECHO ° Generation de la librairie %LIBCASTEM%
    ECHO %LIGNEDOWN%
    CALL %AR% -ruvs %LIBCASTEM% *.o >nul 2>&1
    MOVE %LIBCASTEM% "%CASTEM_REPO%\lib%BIT%">nul
    SET O_BINARY=%CASTEM_REPO%\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=%CASTEM_REPO%\lib%BIT%\%LIBCASTEM%
    ) ELSE IF !ocompt! GTR 3200 (
        CALL XCOPY /Y "%CASTEM_REPO%\lib%BIT%\%LIBCASTEM%" >nul
        CALL %AR% -ruvs %LIBCASTEM% *.o >nul
    2>&1
        SET O_BINARY=%LIBCASTEM%
    )
)

```

```

) ELSE (
    SET O_BINARY=*.o "%CASTEM_REPO%\lib%BIT%\%LIBCASTEM%"))

ECHO .
ECHO Utilisation de %O_BINARY%
REM Appel a gfortran pour faire l edition des liens
CALL %FC% %VERBOSE% -g -O %U_OPTIONS% -Wall -m%BIT% -o
%castX_Local% %O_BINARY% -static -static-libgfortran -static-
libgcc -pthread -lgcov -lpthread -mthreads -mwin32 -mconsole -
lmingwthrd -L"%CASTEM_REPO%\lib%BIT%" -lesope_%BIT% -
lcastem_dll_%BIT% -ljpeg -lfreeglut.dll %BIT% -lopengl32 -
lfxdr_%BIT% -lmedfwrap_%MED_VERSION% -lmedC_%MED_VERSION% -
lhdf5_%HDF5_VERSION% -lstdc++ -lgomp %LD_FLAGS% >
link_%castX_Local%.txt 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 %LIGNEDOWN%
    IF NOT DEFINED LOG_FILE (
        IF EXIST "link_%castX_Local%.txt" DEL /Q
        "link_%castX_Local%.txt"
    )
)

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

) ELSE (
    SET /A ERROLEV=16
    ECHO .
    ECHO %LIGNEUP%
    ECHO ° ECHEC durant l'Edition des liens :
    ECHO %LIGNEDOWN%
    IF EXIST "link_%castX_Local%.txt" (
        ECHO ° Consultez le fichier link_%castX_Local%.txt%SPY%
    )
    IF EXIST %LIBCASTEM% DEL /Q %LIBCASTEM%
)

EXIT /B !ERROLEV!

```

## 2. Script essaicast20

```

#!/bin/bash

#Recuperation des variables d'environnement
[ -z "${CASTEM_VERSION}" ] && CASTEM_VERSION=`f=${(basename
${BASH_SOURCE[0]})} && [[ ${#f} =~ ([0-9]+) ]] && echo ${f: -2}`;
[ -z "${CASTEM_REP}" ] && CASTEM_REP="${( cd "${( dirname
${BASH_SOURCE[0]} )}/... && pwd )}"
source ${CASTEM_REP}/bin/environnement_Cast3M${CASTEM_VERSION}

# Fonction de recherche de bibliotheques
function findLib {

    function find_in_common_path {
        file="$1"
        searchPath="/usr/lib/x86_64-linux-gnu:/usr/lib/x86_64-linux-
gnu/mesa:/usr/lib64:/usr/local/lib64:/usr/X11/lib64:/usr/X11R6/1
ib64:/opt/X11/lib:/usr/lib:/usr/local/lib:/usr/lib/i386-linux-
gnu:/usr/X11/lib:/usr/X11R6/lib:${CASTEM_REP}/lib${BIT}"
        IFS=":"
        for fold in $searchPath; do
            if [ -f "$fold/$file" ]; then
                echo "$fold/$file"
                unset IFS
                return 0
            fi
        done
        unset IFS
        return 1
    }

    for arg in "$@"; do
        find_in_common_path ${arg} && return 0
    done

    if [ "${VERIFICATION_SCRIPTS_CASTEM}" == "VRAI" ]; then
        f=`mktemp`
        echo "$@" | tr ' ' '\n' > ${f}
        corrige_libs_manquante ${f}
        if [ $? -eq 0 ]; then
            echo ${f}/lib/-1 | cut -d'.' -f 1
            return 0
        fi
        echo $"printf \"$mess_err\" | uniq"
    fi

    return 1
}

# 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
VERBOSE=
LOG_FILE="FAUX"
BAD_ARG="FAUX"         # Si VRAI permet d'afficher une sortie avec
erreur d'argument
ESCOPE="FAUX"          # Fabrication de la librairie et edition
des liens du traducteur Escope
let RETURN=0           # Valeur retournee par le script
ETOILE="*****"
*****"

# 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_BAD_ARG=""
    for i in "$@" ; do
        # "--aide" ==> affichage de l'aide en Francais
        # "--help" ==> affichage de l'aide en Anglais
        case ${i} in
            '--aide') AIDE="VRAI";break;;
            '--help') HELP="VRAI";break;;
            '-v') VERBOSE=-v;;
            '-log') LOG_FILE="VRAI";;
            '-ESCOPE') ESCOPE="VRAI";;
            *) BAD_ARG="VRAI"
        esac
    done
fi

```

```

        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 "NAME"
        echo "    essaicast${CASTEM_VERSION} ==> Edition des liens"
        echo ''
        echo "VERSION"
        echo "    Version du Script :"
        ${CASTEM_VERSION}.${CASTEM_REVISION}"
        echo ''
        echo "SYNTAXE"
        echo "    essaicast${CASTEM_VERSION}"
        echo ''
        echo "DESCRIPTION"
        echo "    --aide : Affiche le manuel de cette commande en
francais"
        echo "    --help : Affiche le manuel de cette commande en
anglais"
        echo "    -v : invoque le linker en mode verbose"
        echo "    -log : conserve le .txt a l'issue de l'édition
des liens"
        # echo "    -ESCOPE : Librairie et Traducteur ESOPE
        [SEMT/LM2S seulement]"
        echo ''
        echo "EXAMPLES"
        echo "    essaicast${CASTEM_VERSION} "
        echo "        Realise l'édition des liens dans le
repertoire courant et"
        echo "        produit un executable binaire"
        cast_${BIT}_${CASTEM_VERSION}"
        echo ''
        echo "        En cas d'échec un fichier
link_cast_${BIT}_${CASTEM_VERSION}.txt est généré"
        echo ''
        echo "AUTEUR"
        echo "    Script écrit par Clément BERTHINIER"
        echo ''
        echo "VOIR AUSSI"
        echo "    Aide du Script 'castem${CASTEM_VERSION}' ==>
castem${CASTEM_VERSION} --aide"
        echo "    Aide du Script 'compilcast${CASTEM_VERSION}' ==>
compilcast${CASTEM_VERSION} --aide"
        echo "${ETOILE}"
        echo ''
    else
        # Affiche l'aide du script en Anglais
        echo ''
        echo "${ETOILE}"
        echo "NAME"
        echo "    essaicast${CASTEM_VERSION} ==> Realise link
edition in current directory"
        echo ''
        echo "VERSION"
        echo "    Script Version :"
        ${CASTEM_VERSION}.${CASTEM_REVISION}"
        echo ''
        echo "SYNTAX"
        echo "    essaicast${CASTEM_VERSION}"
        echo ''
        echo "DESCRIPTION"
        echo "    --aide ==> Print the manual of this script in
French"
        echo "    --help ==> Print the manual of this script in
English"
        echo "    -v ==> invoke the linker in verbose mode"
        echo "    -log ==> keep the .txt file at the end of the
link process"
        # echo "    -ESCOPE : Librairie et ESOPE Translator
[SEMT/LM2S only]"
        echo ''
        echo "EXAMPLES"
        echo "    essaicast${CASTEM_VERSION} "
        echo "        Proceed to the link in the current directory
and generate a"
    fi
fi

```

```

        echo "          binary executable
cast_${BIT}_${CASTEM_VERSION}"
echo ''
echo "          If the process fail the file
link_cast_${BIT}_${CASTEM_VERSION}.txt is created"
echo ''
echo "AUTEUR"
echo "      Script written by Clement BERTHINIER"
echo ''
echo "SEE ALSO"
echo "      Manual for 'castem${CASTEM_VERSION}'      :
'castem${CASTEM_VERSION} --help'"
echo "      Manual for 'compilcast${CASTEM_VERSION}'      :
'compilcast${CASTEM_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_${CASTEM_PLATEFORME}_${BIT}_${CASTEM_VERSION}
ESOPE_EXEC_LOCAL=esop_${BIT}_${CASTEM_VERSION}
if [ -f ${ESOPE_EXEC_LOCAL} ]; then \rm -f ${ESOPE_EXEC_LOCAL}
; fi

echo
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 [ ${CASTEM_PLATEFORME} == "Linux" ] ; then
    OPT_DEP="-umain"
    LIB_DEP="-lrlt"
else
    OPT_DEP=""
    LIB_DEP=""
fi

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

# Test d'existence de ${ESOPE_EXEC_LOCAL}
if [ -x ${ESOPE_EXEC_LOCAL} ]; then
    if [ "${LOG_FILE}" == "FAUX" ]; then
        rm -f link_${ESOPE_EXEC_LOCAL}.txt
    fi

    # Test d'existence de ${ESOPE_EXEC}
    if [ ! -x ${ESOPE REP}/bin/${ESOPE_EXEC} ]; then
        echo
        echo
*****
        echo "*      Generation de l'executable ${ESOPE_EXEC}"
*"
        echo
*****
        echo ''
    else
        # Cas ou le binaire Standard est bien présent dans le
        # repertoire /bin de Cast3M
        echo
        echo
*****
        echo "*      Mise a jour de l'executable ${ESOPE_EXEC}"
*"
        echo
*****
        echo ''
    fi
fi

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 été généré
echo
echo
*****
echo "*      ECHEC durant l'Edition des liens
*"

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

# Définition de variables
castX="bin_Cast3M_${CASTEM_PLATEFORME}_${LICENCETYPE}_${BIT}_${CASTEM_VERSION}"
castX_Local="cast_${BIT}_${CASTEM_VERSION}"
GLUTLIB="FAUX"
GLULIB="FAUX"
GLLIB="FAUX"
XLIB="FAUX"
XxfLIB="FAUX"

# Nettoyage d'une précédente opération d'édition des liens
if [ -f link_${castX_Local}.txt ]; then \rm -f
link_${castX_Local}.txt ; fi
if [ -x ${castX_Local} ] ; then \rm -f ${castX_Local}
; fi

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

#####
# Recherche de la librairie libGLU.so dans les répertoires
#####
if [ ${CASTEM_PLATEFORME} == "Linux" ] ; then
    ret=$(findLib libGLU.so libGLU.so.1)
    if [ $? == 0 ]; then
        GLULIB=$ret
    else
        GLULIB_ERR=$ret
    fi
fi

```

```

# Cas de la Plateforme MAC OSX 64-bits
ret=$(findLib libGLU.dylib libGLU.so.1 libGLU.1.dylib)
if [ $? == 0 ]; then
    GLULIB=${ret}
else
    GLULIB_ERR=${ret}
fi
fi

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

#####
# Recherche de la librairie libX11 dans les repertoires
#####
if [ $CASTEM_PLATEFORME == "Linux" ] ; then
    ret=$(findLib libX11.so libX11.so.6)
    if [ $? == 0 ]; then
        X11LIB=${ret}
    else
        X11LIB_ERR=${ret}
    fi
else
    # Cas de la Plateforme MAC OSX 64-bits
    ret=$(findLib libX11.dylib libX11.6.dylib)
    if [ $? == 0 ]; then
        X11LIB=${ret}
    else
        X11LIB_ERR=${ret}
    fi
fi

#####
# Recherche de la librairie libXxf86vm dans les repertoires
#####
if [ $CASTEM_PLATEFORME == "Linux" ] ; then
    ret=$(findLib libXxf86vm.so libXxf86vm.so.1)
    if [ $? == 0 ]; then
        XxfLIB=${ret}
    else
        XxfLIB_ERR=${ret}
    fi
else
    # Cas de la Plateforme MAC OSX 64-bits
    ret=$(findLib libXxf86vm.dylib libXxf86vm.1.dylib)
    if [ $? == 0 ]; then
        XxfLIB=${ret}
    else
        XxfLIB_ERR=${ret}
    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
    \E0 ajouter \E0 Cast3M (en omettant main.o et bdata.o)
    nb_o=$( ls -l *.o | wc -l )
    if [ ${nb_o} -gt 0 ] ; then
        # Presence du *.o
        O_BINARY='*' 
    else
        # Absence du *.o
        O_BINARY=' '
    fi
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
    ar -ruvs ${CASTEM_REP}/lib${BIT}/libcastem_${LICENCETYPE}_${BIT}.a ./*.o
> /dev/null 2>&1

    # Absence du *.o
    O_BINARY=' '
fi

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

# Definition des Librairies a utiliser pour faire l'édition
des liens
if [ ${CASTEM_PLATEFORME} == "MAC" ] ; then
    LIBRARY_REP="-L${CASTEM_REP}/lib${BIT}"
    STATIC_LIBRARY="-lcastem_${LICENCETYPE}_${BIT} -lesope_${BIT} -lfxdr_${BIT} -ljpeg_${BIT} -lmed_${BIT} -lhdf5_${BIT} -lz_${BIT}" 
    SHARED_LIBRARY="${GLULIB} ${GLULIB} ${GLLIB} ${X11LIB} ${XxfLIB} -lpthread -ldl -lstdc++"
    OPT_DEP=""
else
    LIBRARY_REP="-L${MPI_LIBREP} -L${CASTEM_REP}/lib${BIT}"
    STATIC_LIBRARY="-lcastem_${LICENCETYPE}_${BIT} -lesope_${BIT} -lfxdr_${BIT} -ljpeg_${BIT} -lmed_${BIT} -lhdf5_${BIT} -lz_${BIT} ${GLULIB} -l numa_${BIT} -lrpc_${BIT}" 
    SHARED_LIBRARY="${GLULIB} ${GLLIB} ${X11LIB} ${XxfLIB} -lmpi_mpifh -lmpi -lpthread -lrt -ldl -lstdc++"
    OPT_DEP="-umain -u_BLOCK_DATA__ -openmp"
fi

# Edition des liens pour generer le fichier "${castX_Local}"
${GFORTRAN} ${VERBOSE} -m${BIT} -static-libgfortran -o ${castX_Local} ${OPT_DEP} ${O_BINARY} ${LIBRARY_REP} ${STATIC_LIBRARY} ${SHARED_LIBRARY} > link_${castX_Local}.txt
2>&1

# Test d'existence de ${castX_Local}
if [ -x ${castX_Local} ] ; then
    if [ "${LOG_FILE}" == "FAUX" ] ; then
        rm -f link_${castX_Local}.txt
    fi
    # Test d'existence de
bin_Cast3M_${CASTEM_PLATEFORME}_${LICENCETYPE}_${BIT}_${CASTEM_ERSION}
    if [ ! -x ${CASTEM_REP}/bin/bin_Cast3M_${CASTEM_PLATEFORME}_${LICENCETYPE}_${BIT}_${CASTEM_VERSION} ] ; then
        echo
        echo
        ****
        echo "* Generation de l'executable
bin_Cast3M_${CASTEM_PLATEFORME}_${LICENCETYPE}_${BIT}_${CASTEM_VERSION} *"
        echo
        echo
        ****
        echo
    fi
fi

```

```

mv ${castX_Local}
${CASTEM_REF}/bin/bin_Cast3M_${CASTEM_PLATEFORME}_${LICENCETYPE}
_${BIT}_${CASTEM_VERSION}

else
# Cas ou le binaire Standard est bien present dans le
repertoire /bin de Cast3M
echo
echo
***** Edition des liens reussie *****
echo "*"
echo
*****
echo
fi

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

if [ -f link_${castX_Local}.txt ]; then
echo "*" Consultez le fichier link_${castX_Local}.txt
fi
echo
echo
*****
echo "* Certaines librairies sont manquantes"
echo "* Some librairies are missing"
echo
echo
if [ "$GLUTLIB" == "FAUX" ]; then
printf "$GLUTLIB_ERR"
fi

if [ "$GLULIB" == "FAUX" ]; then
printf "$GLULIB_ERR"
fi

if [ "$GLLIB" == "FAUX" ]; then
printf "$GLLIB_ERR"
fi

if [ "$X11LIB" == "FAUX" ]; then
printf "$X11LIB_ERR"
fi

if [ "$XxfLIB" == "FAUX" ]; then
printf "$XxfLIB_ERR"
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
*****
echo
let RETURN=1
fi

# Nettoyage du repertoire temp
[ -d temp ] && rm -rf temp

```

## Annexe I. Scripts cast\_UTIL20.bat et cast\_UTIL20

### 1. Script cast\_UTIL20.bat

```

@ECHO OFF
SETLOCAL ENABLEDELAYEXPANSION

SET SCRIPTREP=%~dp0

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

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

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 °           Un fichier UTILPROC a été créé °
    ECHO %LINEDOWN%)

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 °           Un fichier UTILNOTI a été créé °
    ECHO %LINEDOWN%)

EXIT /B 0

```

## 2. Script cast\_UTIL20

```

#!/bin/bash

#Recuperation des variables d'environnement
[ -z "${CASTEM_VERSION}" ] && CASTEM_VERSION=`f=${(basename
${BASH_SOURCE[0]})} && [[ "$f" =~ ([0-9]+)$ ]] && echo ${f: -2}`
[ -z "${CASTEM_REPO}" ] && CASTEM_REPO="`cd ${dirname
"${BASH_SOURCE[0]}"}/.. && pwd`"
source ${CASTEM_REPO}/bin/environnement_Cast3M${CASTEM_VERSION}

castScript="castem${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.txt ]; then rm -f
ZZZ_Procedures_ERREURS.txt; fi
if [ -f ZZZ_Notes_ERREURS.txt ]; then rm -f
ZZZ_Notes_ERREURS.txt; fi

ETOILE="_____"

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

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

    [ -f ZZZ_proc.procedur ] && rm -f ZZZ_proc.procedur
    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

    # Lancement de Cast3M.
    #Modification T.L. :
    #           Avant on ecrivait les instructions directement dans
    'ZZZ_proc.dgibi'.
    #           Cela posait probleme lors de la fabrication de
    Cast3M :
    #           Initialement, les fichiers 'CAST3M.*' n'existent pas,
    Cast3M demarre donc en erreur et les instructions du fichier
    DGIBI ne sont pas executees.
    #           On utilise "here document" de Bash pour faire
    comme si on ecrivait les instructions en interactif, une fois
    l'erreur passee.
    $castScript <<EOF > ZZZ_proc.res 2>&1
    util proc 'ZZZ_proc.procedur';
    fin;
    EOF
    rm -f ZZZ_proc.res ZZZ_proc.procedur ZZZ_procedur.trace

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

```

```

#####
# La notices '*.notice' est supprimee : Elle est en double
# avec etoil.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_substraction.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 './aaa1.notice' ]; then mv ./aaa1.notice'
    aaa_division.notice ; fi
    if [ -f './**.notice' ]; then mv ./**.notice'
    aaa_puissance.notice ; fi

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

    [ -f ZZZ_noti.notice ] && rm -f ZZZ_noti.notice
    for i in *.notice; do
        # Affichage du nom de la notice
        echo "| ${i}"

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

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

    # Lancement de Cast3M (de la meme facon que pour les
    procedures)
$castScript <<EOF > ZZZ_noti.res 2>&1
util noti 'ZZZ_noti.notice';
fin;
EOF
    rm -f ZZZ_noti.res ZZZ_noti.notice ZZZ_notice.trace

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

```

```

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

```

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

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

## Annexe J. Scripts castem20.bat et castem20

### 1. Script castem20.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 est remplacé automatiquement par InstallJammer)
CALL %SCRIPT_REP%~0,-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=                               Licence
INDUSTRIELLE
) ELSE (
    SET castX=bin_Cast3M_Win_DEVEL_%BIT%_%CASTEM_VERSION%.exe
    SET CASTEM_LICENCE=                               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
)
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_FICHIERES
)

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 quantite de memoire a reserver apres
l'option -MEM
EXIT /B 1
) ELSE (
SET Val=%ArgNAME2%
SET MEMDEF=VRAI
SET OPTIONS=%OPTIONS% %ArgNAME1% %ArgNAME2%
SHIFT
SHIFT
GOTO DEBUT_LECTURE
)

REM Reglage de la memoire laissee libre au systeme
IF "%ArgNAME1%"=="-LIBRE" (
IF "%ArgNAME2%"==""
ECHO.
ECHO Il manque la quantite de memoire a laisser libre apres
l'option -LIBRE
EXIT /B 1
) ELSE (
SET Vallibre=%ArgNAME2%
SET MEMLIB=VRAI
SET OPTIONS=%OPTIONS% %ArgNAME1% %ArgNAME2%
SHIFT
SHIFT
GOTO DEBUT_LECTURE
)

REM Reglage du nombre de CPU reserves par Cast3M (Par defaut la
totalite des CPU disponibles sont pris)
IF "%ArgNAME1%"=="-NCPU" (
IF "%ArgNAME2%"==""
ECHO.
ECHO Il manque le nombre de CPU souhaitez apres 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
*****
:LABEL_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,
°
ECHO ° .INC, .h, GIBI.ERREUR
°
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=2Go
°
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=2Go
°
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_PROJET : Nom du jeu de donnees en
argument °

```

# NOTE DE FABRICATION DE CAST3M 2020

```

ECHO °      CASTEM_VERSION      : Annee de la version de Cast3M
°
ECHO °      CASTEM_REVISION     : Numero de la revision pour
cette annee
°
ECHO °      CASTEM_REPO        : Repertoire dans lequel est
installle Cast3M
°
ECHO °      CASTEM_PLATEFORME   : Plateforme sur laquelle est
installle Cast3M
°
ECHO °      CASTEM_LICENCE      : Licence de Cast3M installe
°
ECHO °
°
ECHO %LIGNEMIDDLE%
ECHO °EXEMPLES
°
ECHO °      castem%CASTEM_VERSION%%SPY%
°
ECHO °      Lance Cast3M sans jeu de donnee : Mode
Interactif
°
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,
°
ECHO °                  .INC, .h, GIBI.ERREUR
°
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=2Go
°
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=2Go
°
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_PROJET    : Name of the current test file
°
ECHO °      CASTEM_VERSION   : Year of the version of Cast3M
°
ECHO °      CASTEM_REVISION  : Update number for this year
°
ECHO °      CASTEM_REPO      : Cast3M Install directory
°
ECHO °      CASTEM_PLATEFORME : Plateform 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 °
°
ECHO °      castem%CASTEM_VERSION%%SPY% file
°
ECHO °      Execute Cast3M with the input file 'file'
°
ECHO °
°
ECHO °      castem%CASTEM_VERSION%%SPY% -d file
°

```

```

ECHO ° Execute Cast3M with the input file 'file'
°
ECHO ° in the gdb environment
°
ECHO °
°
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 %LINEDOWN%
EXIT /B !ERROLEV!
)
REM
*****
*****
```

```

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_REPO%" (
        ECHO .
        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_REPO%\dgibi" dgibi >nul
    CALL XCOPY /S /I "%CASTEM_REPO%\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! GEQ 1 GOTO LISTE_DGIBI
)

REM
*****
***** REM interpretation des caracteres d'echappement le cas echeant
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 executes automatiquement comme la base des cas-tests
IF %dgibicompt% GTR 1 GOTO LISTE_DGIBI

SET NomF3text=%NomF3%
°

SET NomF2text=%CASTEM_PROJET%
°

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%.%CASTEM_REVISION%
°

    SET CASTEXEC="%CASTEM_REPO%\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_PROJET%"==""
    REM Cas ou un nom de fichier est donne
    TITLE Cast3M 20%CASTEM_VERSION% - %BIT%bits : %CASTEM_PROJET%
    ECHO ° REPERTOIRE JEU : %NomF3text:~0,54%°
    ECHO ° NOM FICHIER : %NomF2text:~0,54%°
    IF EXIST "%CASTEM_PROJECT%" (
        SET /A size=7
        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

```

# NOTE DE FABRICATION DE CAST3M 2020



```

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 %LIGNEDEDOWN%

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_PROJET%" CALL :SIZE_FILE %CASTEM_PROJET%
IF %size%==0 DEL "%CASTEM_PROJET%">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 effectuee si l'option --pause a ete renseignee
IF DEFINED PAUSEFIN (pause)
GOTO MESS_FIN

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 dgibierr=0
FOR %%i IN (%LISTE_JDD%) DO (
    SET NomF=%i
    SET NomF2=!NomF:-6!
    ECHO !NomF!
    ECHO 'FIN'; | castem%CASTEM_VERSION% !OPTIONS! !NomF! >
!NomF2!.err 2>&1

    REM Verifie la presence de la chaîne 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 dgibierr+=1
        REM ATTENTION : les espaces dans NomF3 sont importants pour
la presentation
        SET NomF3=!NomF!
    )

    ECHO ° !NomF3:-0,71!">ZZZ_ERROR.txt)
    IF !dgibinum! GTR 0 ECHO Cas-tests reussis :
!dgibinum!/!dgibicompt!
    IF !dgibierr! GTR 0 ECHO Cas-tests echoues :
!dgibierr!/!dgibicompt!
    ECHO.

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

ECHO %LIGNEDEDOWN%

:MESS_FIN
REM Affichage final : Durees, etc.
SET ENDDATE0=%DATE%
SET ENDTIME0=%TIME%

REM convert STARTTIME0 and ENDTIME0 to centiseconds
REM Instant initial
SET STARTANNEE0=%STARTDATE0:~6,4%
SET STARTMOIS0=%STARTDATE0:~3,2%
SET STARTJOUR0=%STARTDATE0:~0,2%
REM : Protection car les nombre commençant par "0" font
planter... : Nombre non valide. Les constantes numériques
doivent être soit décimales (17), soit hexadécimales (0x11),
soit octales (021).
IF %STARTJOUR0:~0,1% EQU 0 SET /A STARTJOUR0=%STARTDATE0:~1,1%
IF %STARTMOIS0% EQU 01 SET /A STARTNUJ= %STARTJOUR0%
IF %STARTMOIS0% EQU 02 SET /A STARTNUJ=31 + %STARTJOUR0%
IF %STARTMOIS0% EQU 03 SET /A STARTNUJ=59 + %STARTJOUR0%
IF %STARTMOIS0% EQU 04 SET /A STARTNUJ=90 + %STARTJOUR0%
IF %STARTMOIS0% EQU 05 SET /A STARTNUJ=120 + %STARTJOUR0%
IF %STARTMOIS0% EQU 06 SET /A STARTNUJ=151 + %STARTJOUR0%
IF %STARTMOIS0% EQU 07 SET /A STARTNUJ=181 + %STARTJOUR0%
IF %STARTMOIS0% EQU 08 SET /A STARTNUJ=212 + %STARTJOUR0%
IF %STARTMOIS0% EQU 09 SET /A STARTNUJ=243 + %STARTJOUR0%
IF %STARTMOIS0% EQU 10 SET /A STARTNUJ=273 + %STARTJOUR0%
IF %STARTMOIS0% EQU 11 SET /A STARTNUJ=304 + %STARTJOUR0%
IF %STARTMOIS0% EQU 12 SET /A STARTNUJ=334 + %STARTJOUR0%

IF "%STARTTIME0:~0,1%"=="0" SET /A STARTTIME=%STARTTIME0:~1,1%
IF %STARTTIME0:~1,1%*36000
IF NOT "%STARTTIME0:~0,1%"=="0" SET /A STARTTIME=%STARTTIME0:~0,2%*36000
IF "%STARTTIME0:~3,1%"=="0" SET /A STARTTIME=%STARTTIME0:~4,1%*6000
IF NOT "%STARTTIME0:~3,1%"=="0" SET /A STARTTIME=%STARTTIME0:~3,2%*6000
IF "%STARTTIME0:~6,1%"=="0" SET /A STARTTIME=%STARTTIME0:~7,1%*100
IF NOT "%STARTTIME0:~6,1%"=="0" SET /A STARTTIME=%STARTTIME0:~6,2%*100
IF "%STARTTIME0:~9,1%"=="0" SET /A STARTTIME=%STARTTIME0:~10,1%
IF NOT "%STARTTIME0:~9,1%"=="0" SET /A STARTTIME=%STARTTIME0:~9,2%

REM Instant final
SET ENDANNEE0=%ENDDATE0:~6,4%
SET ENDMOIS0=%ENDDATE0:~3,2%
SET ENDJOUR0=%ENDDATE0:~0,2%
REM : Protection car les nombre commençant par "0" font
planter... : Nombre non valide. Les constantes numériques
doivent être soit décimales (17), soit hexadécimales (0x11),
soit octales (021).
IF %ENDJOUR0:~0,1% EQU 0 SET /A ENDJOUR0=%ENDDATE0:~1,1%
IF %ENDMOIS0% EQU 01 SET /A ENDNUJ= %ENDJOUR0% +
(%ENDANNEE0%-%STARTANNEE0%)*365
IF %ENDMOIS0% EQU 02 SET /A ENDNUJ=31 + %ENDJOUR0%
IF %ENDMOIS0% EQU 03 SET /A ENDNUJ=59 + %ENDJOUR0%
IF %ENDMOIS0% EQU 04 SET /A ENDNUJ=90 + %ENDJOUR0%
IF %ENDMOIS0% EQU 05 SET /A ENDNUJ=120 + %ENDJOUR0%
IF %ENDMOIS0% EQU 06 SET /A ENDNUJ=151 + %ENDJOUR0%
IF %ENDMOIS0% EQU 07 SET /A ENDNUJ=181 + %ENDJOUR0%
IF %ENDMOIS0% EQU 08 SET /A ENDNUJ=212 + %ENDJOUR0%
IF %ENDMOIS0% EQU 09 SET /A ENDNUJ=243 + %ENDJOUR0%
IF %ENDMOIS0% EQU 10 SET /A ENDNUJ=273 + %ENDJOUR0%
IF %ENDMOIS0% EQU 11 SET /A ENDNUJ=304 + %ENDJOUR0%

```

```

IF %ENDMOIS0% EQU 12 SET /A ENDNUJ=334 + %ENDJOUR0%

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 Calcul de la duree
SET /A DURATIONJ= (%ENDNUJ%-%STARTNUJ%) + ((%ENDANNEE0%-
%STARTANNEE0%)*365)
IF %ENDTIME% LSS %STARTTIME% SET /A DURATION=(%ENDTIME% -
!STARTTIME!) + 8640000
IF %ENDTIME% GEQ %STARTTIME% SET /A DURATION= %ENDTIME% -
%STARTTIME%

REM Conversion en Heures : Minutes : Secondes : Centisecondes
SET /A DURATIONH= %DURATION% / 36000
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 Ajout du 0 pour que le format soit identique
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
IF %DURATIONJ% EQU 0 SET
DURATION_TOT=%DURATIONH%:%DURATIONM%:%DURATIONS%,%DURATIONCS%
.

IF %DURATIONJ% EQU 1 SET DURATION_TOT=%DURATIONJ% Jour
%DURATIONH%:%DURATIONM%:%DURATIONS%,%DURATIONCS% .
IF %DURATIONJ% GTR 1 SET DURATION_TOT=%DURATIONJ% Jours
%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 %LINEDOWN%

EXIT /B !ERROLEV!

REM ****
REM * Definition des fonctions utiles dans ce scrit
REM ****
:RECUPERATION_FICHIERS
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_FICHIERS
)

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_FICHIERS
)

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_FICHIERS
)

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_FICHIERS
)

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_FICHIERS
)

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_FICHIERS
)

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_FICHIERS
)

IF "%ArgNAME1%"=="GIBI.ERREUR" (
IF NOT EXIST "%CASTEM_REP%\data\%ArgNAME1%" (
SET Error_Liste=%Error_Liste% %ArgNAME1%
) ELSE (
COPY "%CASTEM_REP%\data\%ArgNAME1%" . >nul)
SHIFT
GOTO RECUPERATION_FICHIERS
)

IF NOT "%ArgNAME1%"==" " (
SET Error_Liste=%Error_Liste% %ArgNAME1%
SHIFT
GOTO RECUPERATION_FICHIERS
)

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

REM ****
:SIZE_FILE
REM Calcul la taille d'un fichier en octets passe en argument
SET size=%~z1

```

```
REM Il arrive que la commande precedente echoue  
IF "%size%" == "" SET size=0  
EXIT /B 0
```

## 2. Script castem20

```

#!/bin/bash

#Recuperation des variables d'environnement
[ -z "${CASTEM_VERSION}" ] && CASTEM_VERSION=`f=${basename
${BASH_SOURCE[0]}} && [[ ${f} =~ ([0-9]+)$ ]] && echo ${f}: -
2`#
[ -z "${CASTEM_REP}" ] && CASTEM_REP="$( cd "$( dirname
"${BASH_SOURCE[0]}")/... && pwd )"
source ${CASTEM_REP}/bin/environnement_Cast3M${CASTEM_VERSION}

#fonction utilise pour 'castemXX -r fichiers'
function find_file {
    liste_rep=(dgibi
        "data" "data"
        "divers" "divers" "divers"
        "include/c"
        "include/eso"
        "notice"
        "procedur"
        "sources")
    liste_ext=(dgibi
        "MASTER" "PROC" "ERREUR"
        "med" "fem" "txt" "stl"
        "h"
        "INC"
        "notice"
        "procedur"
        "eso")

    motif="$1"
    repertoires_fichiers=""

    #tentative pour determiner dans quel repertoire il faudra
    rechercher le fichier :
    extension="${motif##*.*}"
    if [ -n "${extension}" ] && [ "${extension}" != '*' ]; then
        #transformation des "*" en ".*" pour regexp :
        extension=${extension//\*/.*}

        for (( i=0; i<${#liste_rep[@]}; i++ )); do
            if [[ ${liste_ext[$i]} =~ ^${extension}$ ]]; then
                repertoires_fichiers=${CASTEM_REP}/${liste_rep[$i]}"
                break
            fi
        done
    fi

    #Le repertoire n'a pas pu etre determine, on recherche dans
    tous les repertoires :
    [ -z "${repertoires_fichiers}" ] &&
    repertoires_fichiers=$(printf "%s\n"
    "${liste_rep[@]} | sort -u`

    echo "find ${repertoires_fichiers} -type f -iname
"${motif}""
}

# Definition de variables
castX="bin_Cast3M_${CASTEM_PLATEFORME}_${LICENCETYPE}_${BIT}_${CASTEM_VERSION}"
castX_Locale="cast_${BIT}_${CASTEM_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
contendra les commandes tapees 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
MEMLIB="FAUX"
VALLIBRE=0
ZERMEM="FAUX"
SWAP="VRAI"
CPUDEF="FAUX"         # Si VRAI signifie que l'on bride le
nombre de CPU
VALCPU=0
TEST="FAUX"           # Si VRAI permet de lancer l'ensemble de
la base de donnees
let RETURN=0           # Valeur retournee par le script
ETOILE="_____"
declare -a fichiers_trouves

# 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"

        # tentative lecture d'une option valide
        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"
    done
fi
}

```

```

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 [ "${arg_num1}" == "-LIBRE" ]; then
    MEMLIB="VRAI"
    VALLIBRE=${arg_num2}
    SORT_WHI="FAUX"
    shift
    shift

elif [ "${arg_num1}" == "-ZERMEM" ]; then
    ZERMEM="VRAI"
    SORT_WHI="FAUX"
    shift

elif [ "${arg_num1}" == "-NOSWAP" ]; then
    SWAP="FAUX"
    SORT_WHI="FAUX"
    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

    # liste des fichiers a recuperer :
    elif [ "$GET_FICHIER" == "VRAI" ] && [ "${arg_num1}" != "" ]
    ; then
        fichiers_trouves=`find_file ${arg_num1}`
        ${fichiers_trouves}
        if [ -z "$fichiers_trouves" ]; then
            BAD_ARG="VRAI"
            LISTE_BAD_ARG="${LISTE_BAD_ARG} ${arg_num1}"
        fi
        SORT_WHI="FAUX"
        shift

        #fichier DGIBI a executer
        elif [ "$GET_FICHIER" == "FAUX" ] && [ -f "${arg_num1}" ];
        then
            LISTE_DGI_1="${LISTE_DGI_1} ${arg_num1}" #LISTE_DGI_1 =>
            c'est historique ! en realite, seul un fichier DGIBI peut etre
            fournir, a mettre a jour !
            COMPT_DGI_1=$((COMPT_DGI_1+1))
            SORT_WHI="FAUX"
            shift

        #options invalides
        elif [[ ${arg_num1} != "" ]]; then
            BAD_ARG="VRAI"
            LISTE_BAD_ARG="${LISTE_BAD_ARG} ${arg_num1}"
            SORT_WHI="FAUX"
            shift
        fi
        done

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

    if [ "${MEMLIB}" == "VRAI" ]; then
        export ESOPE_PARAM="${ESOPE_PARAM},LIBRE=${VALLIBRE}"
    fi

    if [ "${ZERMEM}" == "VRAI" ]; then
        export ESOPE_PARAM="${ESOPE_PARAM},ZERMEM=OUI"
    fi

    if [ "${SWAP}" == "FAUX" ]; then
        export ESOPE_PARAM="${ESOPE_PARAM},NTRK=300000,LTRK=1"
    fi

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

    # Les differentes possibilites sont passees en revue
    if [ "${BAD_ARG}" == "FAUX" ]; then
        # Cas ou les arguments sont bien des options existantes et/ou
        # des fichiers avec la bonne extension
        if [ "${C_UTIL}" == "VRAI" ]; then
            # Cas ou l'on souhaite construire UTILPROC & UTILNOTI
            cast_UTIL${CASTEM_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
                # Preparation des repertoires 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 presence de fichier .dgibi dans le repertoire
                courant
                \ls -l *.dgibi > /dev/null 2>&1
                CODE_RETUR=$?

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

                    dgibinum=0
                    dgibieerr=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;" | ${CASTEM_REP}/bin/castem$CASTEM_VERSION
                        $i > ${NOM_RES} 2>&1

                        # 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
                        ERROR_CASTEM=`grep -c "ARRET DU PROGRAMME CAST3M
                        NIVEAU D'ERREUR: 0" ${NOM_RES}`
                        if [ ${ERROR_CASTEM} = "1" ] ; then
                            dgibinum=`expr ${dgibinum} + 1`
                        else
                            dgibieerr=`expr ${dgibieerr} + 1`
                            mv ${NOM_RES} ${NOM_ERR}
                            echo "$i" >> ZZZ_ERROR.txt
                        fi
                fi
            fi
        fi
    fi
fi

```

```

        echo "| Cas-tests reussis :"
${dgibinum}"/"${dgibicompt}
        if [ ${dgibierr} -gt 0 ] ; then echo "| Cas-tests
echoues :" ${dgibierr}"/"${dgibicompt} ;fi
        echo "|"
done
let RETURN=0

# Affichage final apres l'execution
echo "${ETOILE}"
if [ ${dgibierr} -eq 0 ] ; then
        echo "| LES CAS-TESTS ONT ETE EXECUTES
AVEC SUCCES"
else
        echo "| LES ${dgibierr} CAS-TESTS
SUIVANTS ONT ECHOUES"
        echo "${ETOILE}"
        cat ZZZ_ERROR.txt
        echo "${ETOILE}"
        echo "| Consultez les fichiers .err correspondants"
fi
echo "${ETOILE}"
fi
cd ..
fi

elif [ "${AIDE}" == "VRAI" ] ; then
# Affiche l'aide du script en Francais
echo ''
echo "${ETOILE}"
echo "NOM"
echo "    castem${CASTEM_VERSION} : Logiciel de calcul par
Element Finis"
echo "    Site web : http://www-cast3m.cea.fr/"
echo ''
echo "VERSION"
echo "    Version du Script :
${CASTEM_VERSION}.${CASTEM_REVISION}"
echo ''
echo "SYNTAXE"
echo "    castem${CASTEM_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 "    -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 :
compatible avec '-d' et '-V'"
echo "    -d           : Lance Cast3M avec gdb [Version
developpeur]"
echo "        non compatible avec '-V'"
echo "    -V           : Lance Cast3M avec valgrind (si
valgrind est installe)"
echo "        non compatible avec '-d'"
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=2Go"
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=2Go"
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 "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_REPO : Repertoire dans lequel est
installe Cast3M"
        echo "    CASTEM_PLATEFORME : Plateforme sur laquelle est
installe Cast3M"
        echo "    CASTEM_LICENCE : Licence de Cast3M installe"
echo ''
echo "EXEMPLES"
echo "    castem${CASTEM_VERSION}"
echo "        Lance Cast3M sans jeu de donnee : Mode
Interactif"
echo ''
echo "    castem${CASTEM_VERSION} fichier"
echo "        Lance le jeu de donnee 'fichier'"
echo ''
echo "    castem${CASTEM_VERSION} [-d | -V] fichier"
echo "        Lance le jeu de donnee 'fichier' dans
l'environnement"
echo "        gdb"
echo ''
echo "    castem${CASTEM_VERSION} -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 "    castem${CASTEM_VERSION} -r 'p*.procedur'
'main.eso' '*l.notice'"
echo "        'elas*.dgibi'"
echo "        Recupere la liste de fichiers : les
expressions regulieres"
echo "        sont interpretees"
echo ''
echo "    castem${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${CASTEM_VERSION}' :
'compilcast${CASTEM_VERSION} --aide'"
echo "    Aide du Script 'essaicast${CASTEM_VERSION}' :
'essaicast${CASTEM_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${CASTEM_VERSION}: Finite Element solver
Software"
echo "    Web site: http://www-cast3m.cea.fr/"
echo ''
echo "VERSION"
echo "    Script Version :
${CASTEM_VERSION}.${CASTEM_REVISION}"
echo ''
echo "SYNTAX"
echo "    castem${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 "    -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 :
compatible with '-d' and '-V'"
echo "    -d           : Execute Cast3M with gdb (if
gdb is installed)"
echo "        not compatible with '-V'"
echo "    -V           : Execute Cast3M with valgrind (if
valgrind is installed)"
echo "        not compatible with '-d'"
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=2Go"
echo "           -LIBRE Val2: Free memory for the system"
echo "           Overwrite -MEM Val1"
echo "           -Example in MOTS : Val2=360027352"
echo "           -Example in Mo   : Val2=1500Mo"
echo "           -Example in Go  : Val2=2Go"
echo "           -ZEROMEM : 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 "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 : Plateform on which is
installed Cast3M"
echo "      CASTEM_LICENCE : Cast3M type of License"
echo ""
echo "EXAMPLES"
echo "      castem${CASTEM_VERSION}"
echo "          Execute Cast3M without any input file : "
Interactive mode"
echo ""
echo "      castem${CASTEM_VERSION} file.dgibi"
echo "          Execute Cast3M with the input file
'file.dgibi' if it exists"
echo "          If 'file.dgibi' doesn't exist it will be
created"
echo ""
echo "      castem${CASTEM_VERSION} [-d | -V ] file.dgibi"
echo "          Execute Cast3M with the input file
'file.dgibi'"
echo "          in the gdb or valgrind environment"
echo ""
echo "      castem${CASTEM_VERSION} -u file.dgibi"
echo "          Build UTILPROC & UTILNOTI and execute the
input file"
echo "          'file.dgibi'"
echo ""
echo "      castem${CASTEM_VERSION} -r 'p*.procedur'
'main.eso' '*1.notice'"
echo "      'elas*.dgibi'"
echo "          Copy in your directory the file list :
regular expressions"
echo "          are interpreted"
echo ""
echo "      castem${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${CASTEM_VERSION}' :
'compilcast${CASTEM_VERSION} --help'"
echo "      Manual for 'essaicast${CASTEM_VERSION}' :
'essaicast${CASTEM_VERSION} --help'"
echo "          Developper version of Cast3M only"
echo "${ETOILE}"
echo ""

elif [ "${GET_FICHIER}" == "VRAI" ]; then

    #echo "fichier a copier :"
    let RETURN=1
    for f in ${fichiers_trouves[@]}; do
        #echo $f
        cp $f .
        let RETURN=0
    done

else # Cas ou on demande a lancer Cast3M

    if [ ${COMPT_DGI_1} -gt 1 ] || [ "${LISTE_PRO_1}" != "" ] ||
[ "${LISTE_NOT_1}" != "" ] || [ "${LISTE_ESO_1}" != "" ]; then
        # 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${CASTEM_VERSION} sans arguments
        if [ -f "Cast3M_defaut_20${CASTEM_VERSION}.dgibi" ];
then rm -f "Cast3M_defaut_20${CASTEM_VERSION}.dgibi"; fi
        if [ -f "Cast3M_defaut_20${CASTEM_VERSION}.ps" ];
then rm -f "Cast3M_defaut_20${CASTEM_VERSION}.ps"; fi
        if [ -f "Cast3M_defaut_20${CASTEM_VERSION}.trace" ];
then rm -f "Cast3M_defaut_20${CASTEM_VERSION}.trace"; fi

        export CASTEM_PROJET="Cast3M_defaut_20${CASTEM_VERSION}"
LISTE_DGI_1=${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 [ "$CASTEM_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 ${castX_Local}"
    LOCAL
    else
        # Message lancement de Cast3M d'origine
        echo "${ETOILE}"
        cat ${CASTEM_REP}/bin/LOGO_ASCII_${CASTEM_VERSION}.txt
        echo "${ETOILE}"

    else
        # Message lancement de Cast3M d'origine
        echo "${ETOILE}"
        if [ ${CASTEM_PLATEFORME} == "MAC" ]; then
            echo "|          EXECUTION de ${castX} ORIGINAL
|"
        else

```

```

        echo "| EXECUTION de ${castX} ORIGINAL
|"
    fi
    echo "| Realisation
${CASTEM_VERSION}.0.${CASTEM_REVISION}
|"
    echo "${ETOILE}"
    cat ${CASTEM REP}/bin/LOGO_ASCII_${CASTEM_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}
fi

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 ${CASTEM REP}/bin/rldwrap_${BIT} -f
${CASTEM REP}/bin/liste_rldwrap${CASTEM_VERSION} ${OPTION_CAST}
${MPI_RUNCMD} ${EXEC_CAST}
    let RETURN=$?
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_${CASTEM_VERSION}.lgi"; fi
            if [ -f fort.97 ]; then mv fort.97
"default_castem_${CASTEM_VERSION}.mif"; fi
            if [ -f "${CASTEM_PROJET}.trace" ]; then mv
"${CASTEM_PROJET}.trace" "${CASTEM_PROJET}.dgibi" ; fi
            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
#####
# Sortie d'erreur sur les arguments
echo ''
echo "${ETOILE}"
echo " Arguments invalides / Unavailable arguments "
echo " ${LISTE_BAD_ARG}"
echo "${ETOILE}"
echo ''
let RETURN=1
fi

#Valeur de retour - pour test fonctionnel des scripts :
exit ${RETURN}

```



## Annexe K. Documentation Cast3M

### 1. 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>

### 2. 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 2020
- Note de fabrication de Cast3M 2020
- Notes de version de Cast3M 2020

### 3. 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)

### 4. 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 L. Traçabilité

 DE LA RECHERCHE À L'INDUSTRIE		Note Technique DES	Page 2/72
		Réf. : SEMT/LM2S/NT/2020-66505	
		Date : 14/05/2020	Indice : A
Note de fabrication de Cast3M 2020			

NIVEAU DE CONFIDENTIALITE				
DO	DR	CCEA	CD	SD
X				

PARTENAIRES/CLIENTS	ACCORD	TYPE D'ACTION

REFERENCES INTERNES CEA			
DIRECTION D'OBJECTIFS	DOMAINE	PROJET	EOTP
DISN	SIMU	MECAN	A-MECAN-01-01
JALON	INTITULE DU JALON	DELAI CONTRACTUEL DE CONFIDENTIALITE	CAHIERS DE LABORATOIRE
SO			

SUIVI DES VERSIONS			
INDICE	DATE	NATURE DE L'EVOLUTION	PAGES ET CHAPITRES MODIFIES
A	14/05/2020	Document initial	Toutes

	NOM	FONCTION	VISAS	DATES
REDACTEUR	C. BERTHINIER	Ingénieur Chercheur		14/05/2020
VERIFICATEUR(S)	F. DI PAOLA	Ingénieur Chercheur		04/06/2020
AUTRE(S) VISA(S)				
APPROBATEUR	J.C. LE PALLEC	Chef de Laboratoire	 Jean-Claude LE PALLEC L'ATM MECANIQUE DE MULHOUSE	05/06/2020
EMETTEUR	V. VANDENBERGHE	Chef de Service	 Véronique VANDENBERGHE Chef de Service Mécanique de Mulhouse  Signature numérique de VANDENBERGHE Valérie VV173325B Date : 2020.06.09 10:27:31 +02'00'	08/06/2020

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