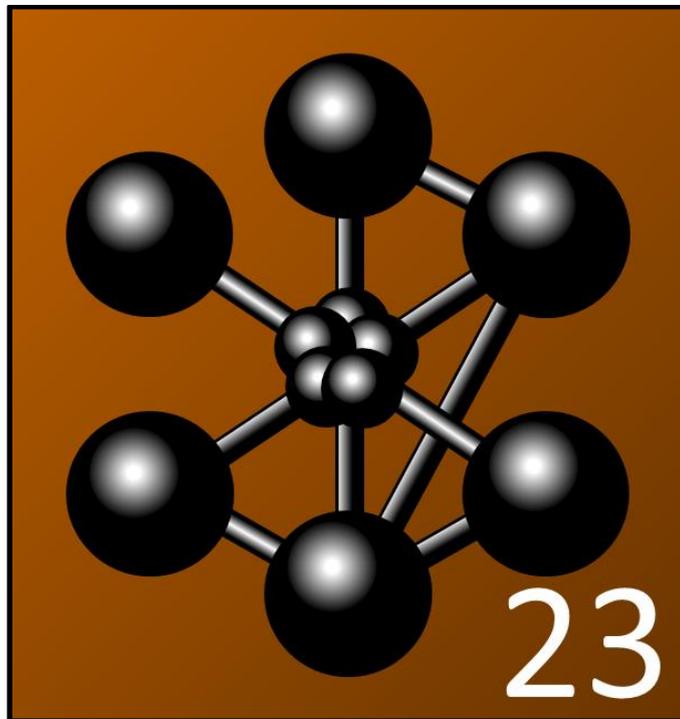


Cast3M 2023

Notes de fabrication de Cast3M 2023



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 des Énergies 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

ASSURANCE QUALITE CAST3M	1
1. PRESENTATION DE CAST3M 2023.....	4
2. PLATEFORMES DE PRODUCTION DE CAST3M	5
2.1 PC – GNU/LINUX (32 BITS).....	5
2.2 PC – GNU/LINUX (64 BITS).....	5
2.3 PC – WINDOWS (32 ET 64 BITS).....	5
3. ÉLABORATION DE LA VERSION 2023 DE CAST3M	6
3.1 OBJET	6
3.2 ÉTAPES DE L'ÉLABORATION DE LA VERSION	6
3.3 SCHEMA DE PRINCIPE DE LA PREPARATION DES VERSIONS ANNUELLES DE CAST3M.....	14

1. PRESENTATION DE CAST3M 2023

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 thermo(visco)plasticité, la mécanique de la rupture, le post-flambage, l'endommagement, la fatigue et la ruine des structures. 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'Arrhenius, modèles EBU ou corrélations, modèles de recombineur catalytique), et des modèles de condensation (condensation en paroi – corrélation 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 opérands 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 : « éducation et recherche » et « 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 Ésope est fourni avec l'exécutable de Cast3M dans le but de traduire les programmes Ésope vers des programmes en Fortran 77. Des scripts de compilation et d'édition des liens sont également fournis afin de pouvoir construire une version modifiée de Cast3M.
- 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 – GNU/LINUX (32 BITS)

Plateforme de compilation :

Modèle de système : CentOS Linux release 6.10 (Final) 2.6.32-754.35.1.el6.centos.plus.x86_64
Type de processeur : Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz
Mémoire Vive : 16 Go

Plateforme de test :

Modèle de système : Debian Linux Squeeze 6.0.10 for i386
Type de processeur : Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz
Mémoire Vive : 2 Go

2.2 PC – GNU/LINUX (64 BITS)

Plateforme de compilation :

Modèle de système : CentOS Linux release 6.10 (Final) 2.6.32-754.35.1.el6.centos.plus.x86_64
Type de processeur : Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz
Mémoire Vive : 16 Go

Plateforme de test :

Modèle de système : Debian Linux Squeeze 6.0.10 for x86_64
Type de processeur : Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz
Mémoire Vive : 4 Go

2.3 PC – WINDOWS (32 ET 64 BITS)

Plateforme de compilation et de test :

Modèle de système : Windows 10.0.19041.2673
Type de processeur : Intel(R) Core(TM) i7-8850H CPU @ 2.60GHz 2.59 GHz
Mémoire Vive : 32 Go

3. ÉLABORATION DE LA VERSION 2023 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)
- GNU/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 figer l'état des développements de Cast3M sur le réseau du SEMT à la date du 31/12 de l'année N-1. Pour cela, une branche est créée sur le dépôt « castem » du réseau Tuleap du CEA (<https://codev-tuleap.intra.cea.fr>). Elle prend le nom de la forme BR_Cast3M_<année_de_la_version> et comprend des sources C, des sources ÉSOPE, des procédures, des notices, des cas-tests et un fichier d'erreurs 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 castem_products_<année_de_la_version>. Ce répertoire est issu du dépôt Git du même nom, possédant une branche avec le même nom que celle du dépôt castem (BR_Cast3M_<année_de_la_version>) qui constitue un projet dédié à la construction de Cas3M. La Figure 1 décrit l'arborescence de ce répertoire. Le contenu de chacun de ces répertoires est détaillé dans les paragraphes qui suivent.

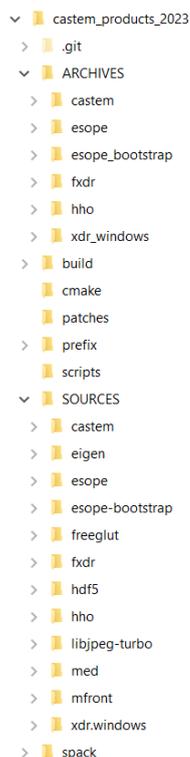


Figure 1. Arborescence des répertoires pour la préparation de Cast3M 2023

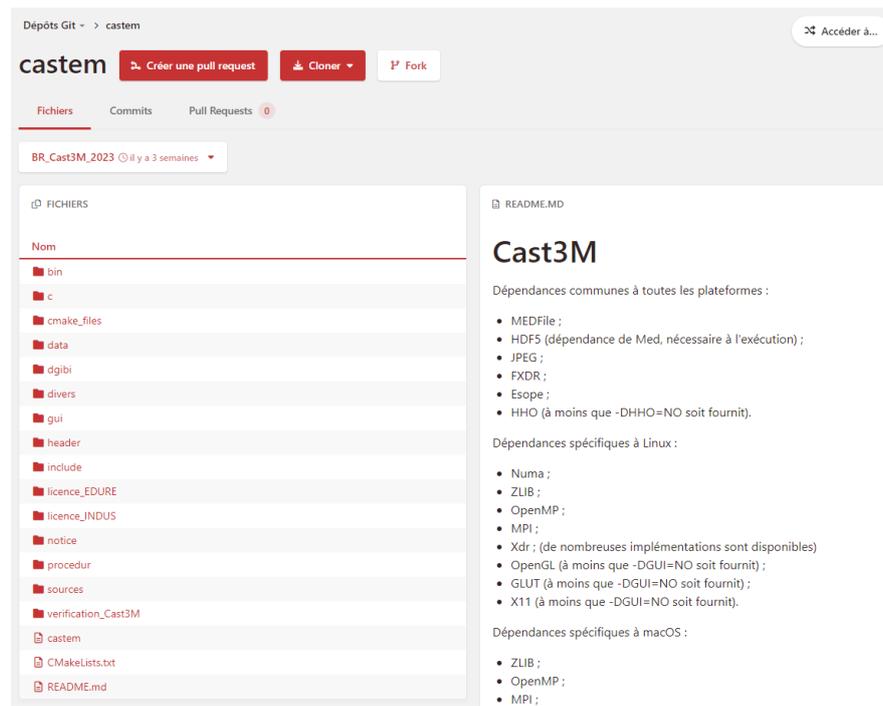


Figure 2. Branche BR_Cast3M_2023 du dépôt Git castem (Tuleap)

- Le répertoire `castem` de la branche `BR_Cast3M_2023` contient la saisie de la version de développement de Cast3M vierge de toutes corrections au 31/12 telle que présente sous l'atelier de développement présent sur le serveur de fichiers Titania. Cette branche ne contient aucun développement poussé au-delà de cette date. Le contenu de ce répertoire est constitué de fichiers et de dossiers à récupérer dans le répertoire `/home/castem-public/castem/` du serveur Titania. Le script `Mise_a_jour_des_miroirs_git.sh` propose une option permettant la saisie de la version. La liste des répertoires à récupérer est donnée sur la Figure 3.

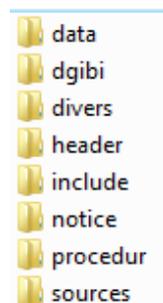


Figure 3. Liste des répertoires à récupérer dans `castem/` du serveur Titania

- `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'entête (pour la sortie de fichiers `.mif` au format Adobe FrameMaker)
- `include` : comprend l'ensemble des Includes ÉSOPE 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 Titania à l'adresse suivante :
`/home/castem-public/castem/GIBI.ERREUR`
- `sources` : répertoire supplémentaire créé pour y placer l'ensemble des sources `.c`, `.eso`, `.h`. Ces fichiers sont extraits à l'aide de la commande
`arc -eon /home/castem-public/castem/castem.arc `*. *``

- Le répertoire `castem_products_2023/SOURCES/castem` (voir Figure 1 et 4) contient l'ensemble des répertoires communs à toutes les plates-formes et mis à jour de toutes les corrections d'anomalies.

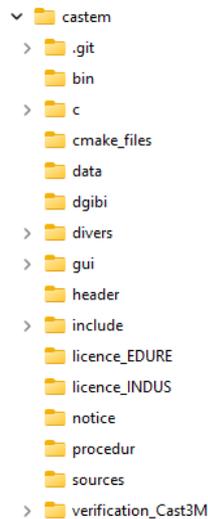


Figure 4. Arborescence des répertoires du dépôt `castem`

- Le répertoire `bin` contient tous les scripts et exécutables. La liste des fichiers présents dans le répertoire `bin` avant la construction de Cast3M est la suivante :



Figure 5. Arborescence du répertoire « `bin` » du dépôt `castem`

Ces fichiers et exécutables permettent la construction de Cast3M pour l'ensemble des plateformes (Windows et Linux).

Chaque script est renommé avec l'année en cours dans son nom pour assurer la cohabitation de différentes versions annuelles.



- La construction de Cast3M s'effectue dans un répertoire `build`, créé à la racine du projet `castem_products` (Figure 1).
 - Sous GNU/Linux, la construction de Cast3M se fait via des commandes CMake (voir 3.2.3)
 - Sous Windows, l'équivalent de ces commandes est envoyé sous forme de commandes MinGW.
 - Sous l'ensemble des plateformes, le dossier `build` est à créer manuellement.
 - Le répertoire `c` contient les sources C de Cast3M
 - Le répertoire `SOURCES` contient les sources Esope de Cast3M
 - Le répertoire `castem_products_2023\build\castem-prefix\src\castem-build` : contient toutes les sources compilées ainsi que les sources Fortran 77 (.f) issues de la traduction des sources Esope. Ce répertoire contient à la fois les sources compilées pour les architectures 32-bits et 64-bits.
 - Les répertoires `licence_EDURE` et `licence_INDUS` contiennent les sources spécifiques permettant de différencier la version sous licence « éducation & recherche » de la version sous licence « industrielle » de Cast3M.
 - *Sources spécifiques pour la licence « éducation & recherche »*
 - `defdat.eso`
 - `fin.eso`
 - `journa.eso`
 - `pilot.eso`
 - `verdat.c`
 - *Sources spécifiques pour la licence « industrielle »*
 - `pilot.eso`
 - `journa.eso`
 - `fin.eso`
 - `perm.c`

3.2.2 Phase 2 : du 01/01 au 31/03 de l'année « N »

Durant cette phase, les corrections d'anomalies qui ont lieu dans la version du jour de Cast3M sont intégrées à la branche `BR_Cast3M_2023`.

L'ensemble des évolutions de Cast3M est répertorié dans le fichier `/home/castem-public/castem/hist.hist` sur le réseau Titania et est consultable en ligne sur le site Cast3M (<http://www-cast3m.cea.fr/index.php?page=anomalies>).

Les fichiers impactés par une évolution sont automatiquement récupérés le lendemain dans la branche `master` (version du jour) via le script `Mise_a_jour_des_miroirs_git` qui récupère les évolutions en créant un `commit`.

Une fois cette phase terminée, les évolutions sont triées et le paquet des sources de la version de l'année « N » de Cast3M est mis à jour via un nouveau `commit` effectué dans la branche `BR_Cast3M_2023`. Les nouveaux développements sont omis et seront intégrés à la version de Cast3M de l'année suivante.

3.2.3 Phase 3 : du 01/04 au 31/05 de l'année « N »

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

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

3.2.3.1 Construction de Cast3M 2023 pour GNU/Linux

La génération des Makefiles pour la construction de Cast3M 2023 pour Linux est réalisée via la commande CMake suivante :

```
cmake /home/kk2000/castem_products_2023/  
-DCMAKE_INSTALL_PREFIX=/home/kk2000/castem_products_2023/prefix  
-DMPI_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-  
12.2.0/openmpi-3.1.6  
-DGCC_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-12.2.0/gcc-  
12.2.0/  
-DISL_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-12.2.0/isl-  
0.21  
-DMPC_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-12.2.0/mpc-  
1.1.0  
-DMPFR_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-12.2.0/mpfr-  
4.0.2  
-DGMP_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-12.2.0/gmp-  
6.1.2  
-DZLIB_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-12.2.0/zlib-  
1.2.11  
-DHWLOC_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-  
12.2.0/hwloc-1.11.11  
-DZSTD_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-12.2.0/zstd-  
1.4.5/  
-DBINUTILS_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-  
12.2.0/binutils-2.35.1/  
-DGETTEXT_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-  
12.2.0/gettext-0.21/  
-DLIBICONV_ROOT=/home/kk2000/spack-0.16.0/opt/spack/linux-centos6-sandybridge/gcc-  
12.2.0/libiconv-1.16/
```

Puis la construction est lancée via la commande :

```
make
```

3.2.3.2 Construction de Cast3M 2023 pour GNU Windows

Pour la version 64 bits, on utilise le compilateur **MinGW-GCC-12.2.0** téléchargeable à l'adresse https://github.com/brechtsanders/winlibs_mingw/releases/download/10.2.0-11.0.0-8.0.0-r8/winlibs-x86_64-posix-seh-gcc-12.2.0-mingw-w64-8.0.0-r8.zip.

Pour la version 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/>

La génération des Makefiles pour la construction de Cast3M 2023 pour Windows est réalisée via la commande CMake suivante :

```
cmake -G "MinGW Makefiles" C:\castem_products_2023 -  
DCMAKE_INSTALL_PREFIX=C:\castem_products_2023\prefix -DGCC_ROOT=C:\GCC_12 -  
DLICENCETYPE="EDUCRECH"
```

Puis la construction est lancée via la commande :

```
mingw32-make
```



3.2.3.3 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, placée dans le répertoire lib.

La liste exhaustive des bibliothèques est :

- **XDR** – (External Data Representation)
Pour Linux, différentes implémentations existent. On utilise l'implémentation fournie par la bibliothèque libtirpc.

Pour Windows, on utilise l'implémentation fournie par la bibliothèque « xdr_windows » (dépôt dans Tuleap). Il s'agit d'un *fork* de <https://sourceforge.net/projects/oncrpc-windows/>, adapté aux besoins de Cast3M. L'installation de cette bibliothèque fournit les fichiers :
 - `${CMAKE_INSTALL_PREFIX}/lib/libxdr.a` : bibliothèque XDR qui doit être utilisée lors de l'édition des liens de Cast3M ;
 - `${CMAKE_INSTALL_PREFIX}/include/*` : fichiers de développements pour la bibliothèque XDR ainsi que des fichiers de la libC qui sont nécessaires.
- **FXDR** – Binding Fortran pour XDR. Il s'agit d'un fork de http://meteora.ucsd.edu/~pierce/fxdr_home_page.html, adapté aux besoins de Cast3M. L'installation de cette bibliothèque fournit les fichiers :
 - `${CMAKE_INSTALL_PREFIX}/lib/libfxdr.a` : bibliothèque FXDR qui doit être utilisée lors de l'édition des liens de Cast3M ;
 - `${CMAKE_INSTALL_PREFIX}/include/fxdr.inc` : non utilisé
 - `${CMAKE_INSTALL_PREFIX}/local/fxdr.3f` : non utilisé
- Les bibliothèques pour l'**interface graphique** (opérateurs TRAC et DESS) de Cast3M. Cast3M dispose de deux interfaces graphiques, dont l'implémentation dépend de la plateforme :
 - L'interface "X" (activable en faisant « OPTI TRAC X ; », activée par défaut) :
 - Linux : utilisation directe de la bibliothèque **X11** ;
 - Windows : utilisation de l'API Win32 ;
 - L'interface "OPEN" (activable en faisant « OPTI TRAC OPEN ; ») :
 - Linux et Windows : Utilisation des bibliothèques **OpenGL** et **FreeGLUT** (la version utilisée est la 3.2.1 disponible sur <http://prdownloads.sourceforge.net>)
- Les bibliothèques pour le **parallélisme** :
 - Linux : **OpenMPI** utilisé par l'opérateur COLL (la version utilisée est la 4.1.0 disponible sur <https://www.open-mpi.org>)
 - Linux et Windows : **libpthread** (les fonctions pour le multi-threading sont définies dans threadid.c).
- La bibliothèque **CMake** fournie par « castem_products » (uniquement pour la construction).
- **JPEG** – La version utilisée pour Cast3M 2023 est la version 9d disponible sur <http://www.ijg.org>.
- **ZLIB** – La version utilisée pour Cast3M 2023 est la version 1.2.11 disponible sur <http://zlib.net>.
- **HDF5** - La version utilisée pour Cast3M 2023 est la version 1.10.3 disponible sur <https://support.hdfgroup.org>
- **MED** - La version utilisée pour Cast3M 2023 est la version 4.1.0 disponible sur <http://files.salome-platform.org>
- **esope** : - Bibliothèque et exécutable Esope pour Cast3M. L'installation d'esope fournit les fichiers :
 - `${CMAKE_INSTALL_PREFIX}/bin/esope` : traducteur Esope vers Fortran ;
 - `${CMAKE_INSTALL_PREFIX}/lib/libesope.a` : définition des symboles utilisés lors de la traduction + gestion Mémoire Esope (GEMAT) ;
 - `${CMAKE_INSTALL_PREFIX}/include/esope.h` : nécessaire pour compiler la source threadid.c de Cast3M. C'est dans threadid.c que sont définies les fonctions pour l'utilisation des threads dans Cast3M.
- **esope_bootstrap** : - Bibliothèques et exécutable Esope déjà construits, pour Linux, Windows et MacOS. Ce dépôt permet d'obtenir une version prête à l'emploi d'Esope, nécessaire pour pouvoir

construire une autre version d'Esope. L'installation d'esoep_bootstrap avec CMake fournit les mêmes fichiers que l'installation d'esoep.

- **hho** : - Bibliothèque pour l'implémentation de la méthode HHO (Hybrid High-Order) dans Cast3M. L'installation de la bibliothèque hho fournit les fichiers :
 - `${CMAKE_INSTALL_PREFIX}/lib/*` : bibliothèques hho qui doivent être utilisées lors de l'édition des liens de Cast3M ;
 - `${CMAKE_INSTALL_PREFIX}/include/*.mod` : modules Fortran. Le module `castem_hho` est utilisé dans la source `hhoc3m.F90` de Cast3M.

3.2.3.4 Traducteur Ésope vers FORTRAN77

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

- Compilation des sources avec la commande `compilcast23 -ESOPE *.eso *.c`
- Edition des liens avec la commande `essaicast23 -ESOPE`

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

3.2.3.5 Compilation de Cast3M

Les sources (`eso` et `C`) sont compilées à l'aide du script `compilcast23 -f *.eso *.c` (GNU/Linux et Windows).

- 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 dans le répertoire `bin`, où sera effectuée l'édition des liens à l'aide du script `essaicast23`, 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_23.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 `castem23 -test` 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 Titania). Celle-ci comprendra le nom du cas-test, la plate-forme ainsi que l'architecture en question.

3.3 SCHEMA DE PRINCIPE DE LA PREPARATION DES VERSIONS ANNUELLES DE CAST3M

La Figure 6 schématise les points précédents et met en évidence la manière dont est gérée l'élaboration d'une version annuelle de Cast3M.

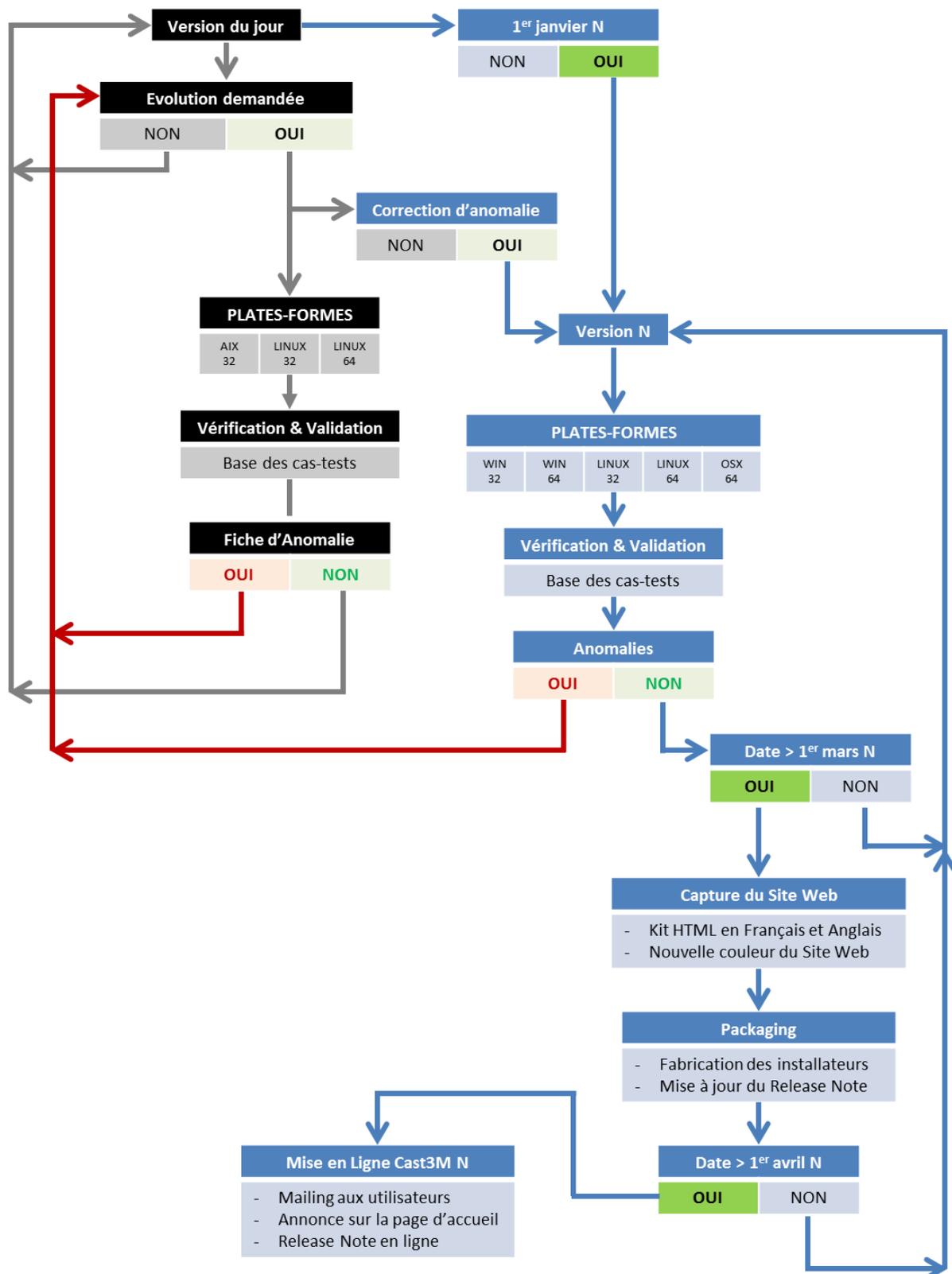


Figure 6. Organigramme de la préparation des versions annuelles de Cast3M



Annexe A. 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. Dépôts Tuleap

<https://codev-tuleap.intra.cea.fr/plugins/git/castem/>

Liste des dépôts utiles pour la construction et la distribution de Cast3M 2023 :

- castem
- esope
- esope_bootstrap
- castem_products
- fxdr
- hho
- xdr_windows
- packager

3. 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 2023
- Note de fabrication de Cast3M 2023
- Note de version de Cast3M 2023
- Guide de validation de Cast3M

4. 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)

5. 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 B. Traçabilité

		Note Technique DES	Page 2/23
		Réf. : SEMT/LM2S/NT/2023-72032	
		Date : 19/07/2023	Indice : A
Notes de fabrication de Cast3M 2023			

NIVEAU DE CONFIDENTIALITE				
DO	DR	CCEA	CD	SD
X				

PARTENAIRES/CLIENTS	ACCORD	TYPE D'ACTION

REFERENCES INTERNES CEA			
DIRECTION D'OBJECTIFS	DOMAINE	PROJET	EOTP
DPE	SIMU	MECAN	A-MECAN-01-04
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	19/07/2023	Document initial	Toutes

	NOM	FONCTION	VISAS	DATES
REDACTEUR	François DI PAOLA	Ingénieur chercheur	DI PAOLA Francois 2023.07.20 08:27:39 +02'00'	
VERIFICATEUR(S)	Gauthier FOLZAN	Ingénieur chercheur		FOLZAN Gauthier 2023.07.19 16:37:29 +02'00'
AUTRE(S) VISA(S)				
APPROBATEUR	Jean-Charles LE-PALLEC	Chef de laboratoire		Signature numérique de Jean-Charles LE-PALLEC Date : 2023.07.24 09:39:12 +02'00'
ÉMETTEUR	Sylvie NAURY	Chef de service	NAURY Sylvie	Signature numérique de NAURY Sylvie Date : 2023.07.24 10:19:11 +02'00'