

Noyau AMDA-NG - 2nde partie



MANUEL D'INSTALLATION

Référence : CDPP-MI-32500-463-CS

Version 1.5 du 04/02/2013

Tableau des signatures



Nom	Fonction	Date	Signature
Préparé par : N.Boursier		04/02/2013	
Vérifié par : S. Frayssines		04/02/2013	
Approuvé par : R.Patrier		04/02/2013	

Sommaire



1.	Introduction.....	6
2.	Documents applicables.....	6
3.	Documents de référence.....	6
4.	Configuration matérielle.....	6
5.	Installation des prérequis.....	8
5.1	Liste des prérequis.....	8
5.2	Installation des Outils de développement.....	9
5.3	Installation des prérequis du noyau AMDA-NG.....	10
5.4	Installation des prérequis de validation.....	13
6.	Installation du noyau AMDA-NG.....	13
6.1	Compilation en mode DEBUG.....	14
6.1.1	Configuration.....	14
6.1.2	Compilation.....	14
6.2	Compilation en mode RELEASE.....	14
6.2.1	Configuration.....	14
6.2.2	Compilation.....	14
6.3	Arborescence AMDA_Kernel.....	14

Table des illustrations



Liste des figures

Figure 1 : Les différentes plateformes..... 7

Liste des tableaux

Tableau 1 : Configuration matérielle 8
Tableau 2 : Liste des prérequis..... 9

Glossaire



Abréviation	Définition
AC-IS	Accord Cadre Informatique Spatiale
AMDA	Automated Multiple Dataset Analysis
AMDA-NG	Automated Multiple Dataset Analysis - New Generation
CCTP	Cahier des Clauses Techniques Particulières
CDPP	Centre de Données de la Physique des Plasmas
CNES	Centre National d'Études Spatiales
COTS	Commercial off-the-shelf
CSSI	Communication et Systèmes – Systèmes d'Information.
GCC	GNU Compiler Collection
GNU	GNU is Not Unix
IP	Internet Protocol
IRAP	Institut de Recherche en Astrophysique et Planétologie
XML	eXtended Markup Language
XSD	XML Schema Definition

1. INTRODUCTION

Ce document présente les procédures à dérouler pour réaliser l'installation du noyau AMDA-NG sur les plateformes de développement et de validation. Il précise les étapes d'installation comme suit :

- ✓ Le chapitre **4. Configuration matérielle** identifie les caractéristiques de la machine hôte du noyau AMDA-NG pour les plateformes de développement et de production.
- ✓ Le chapitre **5. Installation des prérequis** déroule les procédures d'installation des divers COTS nécessaires au développement et à la validation du logiciel.
- ✓ Enfin, le chapitre **6. Installation du noyau AMDA-NG** décrit la procédure de déploiement du logiciel.

2. DOCUMENTS APPLICABLES

DA01	Consultation AC-IS N°DAJ/AR/EO-2013.08449 "Développement du noyau AMDA-NG - Seconde Partie". CDPP-CO-32500-452-CNES, 06/05/2013, Ed. 01 Rev 00.
DA02	Spécification de besoins techniques pour la seconde prestation du nouveau noyau AMDA. CDPP-ST-32500-451-CES, 13/05/2013, Ed. 01 Rev 00
DA03	Cahier des Clauses Techniques Particulières AC-IS. DCT/PS-2011-003173.
DA04	Exigence de réponse aux clauses de sécurité des Systèmes d'information de l'Accord Cadre AC-IS. DCT/PS-2010-15734
DA05	Projet de cahier des prescriptions de Sécurité des Systèmes d'Information Accord Cadre Informatique Spatiale Clauses Générique. DCT/PS-2011-003191
DA06	Exigences Normatives associées aux prestations de développement et de maintenance dans le domaine de l'informatique spatiales. ACIS-ACIBS-SP-GEN-1-CNES

3. DOCUMENTS DE REFERENCE

DR01	Dossier d'architecture du noyau d'AMDA-NG. CDPP-AR-32500-382-SI, Ed. 02 Rev. 01, 29/11/2012.
DR02	Dossier de conception du noyau AMDA-NG. CDPP-CD-32500-436-SI, Ed. 01 Rev. 06, 11/02/2013.
DR03	Dossier de contrôle des interfaces du noyau AMDA-NG. CDPP-IF-32500-438-SI, Ed. 01 Rev. 04, 05/02/2013.
DR04	Étude sur les solutions alternatives à IDL. CDPP-NT-32500-383-SI, Ed. 01 Rev. 02, 10/01/2010.
DR05	Manuel d'installation de AMDA Kernel. CDPP-MI-32500-440-SI, Ed. 01 Rev 05, 11/02/2013.

4. CONFIGURATION MATERIELLE

Tout serveur hôte pour le noyau AMDA-NG doit avoir les caractéristiques suivantes :

- ✓ Système d'exploitation : CentOS 6.3 64bits
- ✓ Les packages 32bits sont installés
- ✓ Le port 8081 est ouvert

- ✓ Ce serveur a accès à DDServer (adresse IP déclarée après du pare-feu)
- ✓ L'adresse IP du serveur est déclarée dans le fichier /etc/hosts (la commande `hostname -i` doit retourner l'adresse IP du serveur)
- ✓ Il existe un utilisateur disposant des droits d'administration

Le schéma ci-dessous présente l'infrastructure générale mise en place pour les plateformes de développement et de production.

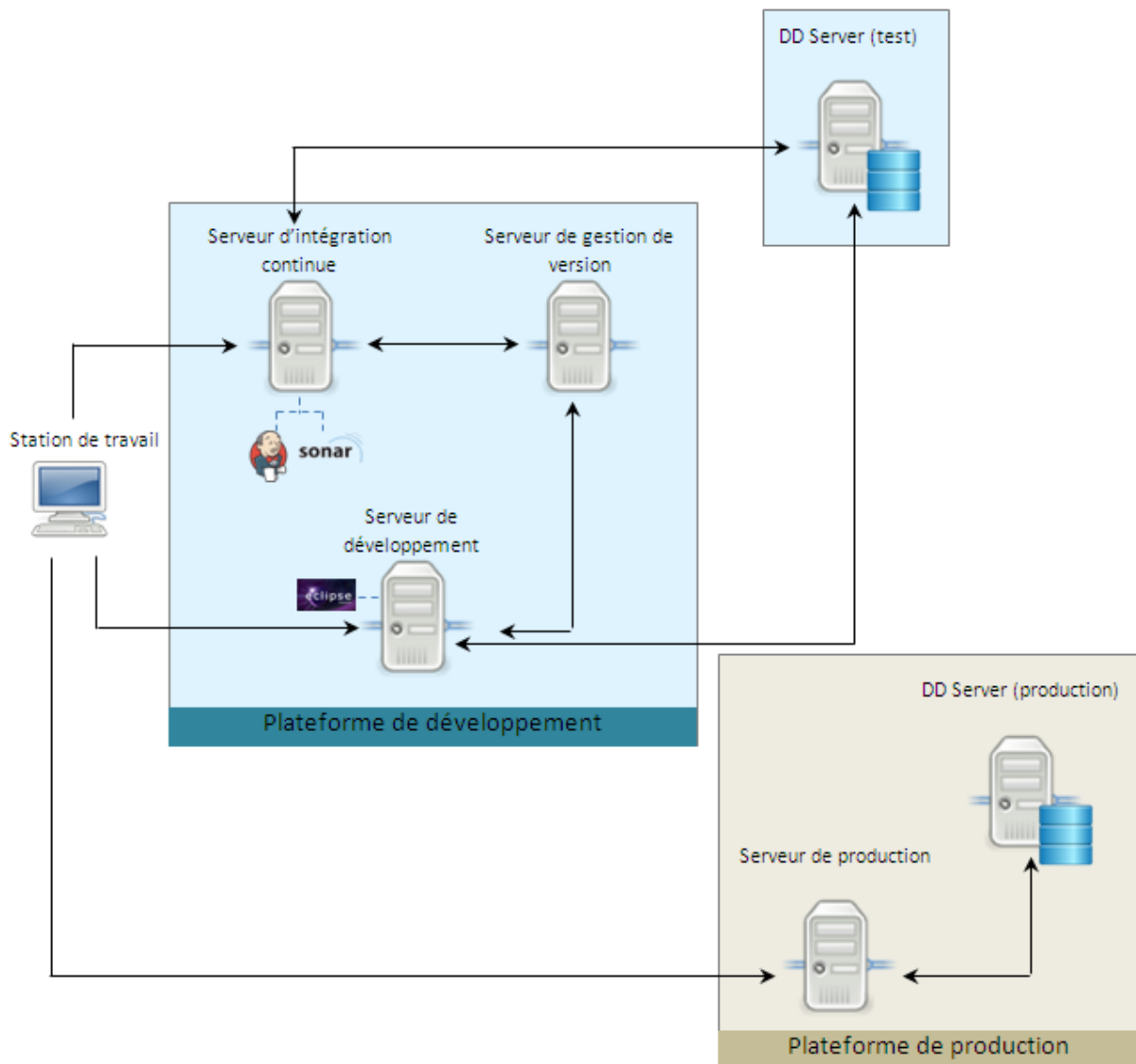


Figure 1 : Les différentes plateformes

Serveur	Prérequis	Description
Développement	Espace requis : <ul style="list-style-type: none"> ✓ 8Go d'espace disque pour le système ✓ 200Mo pour /boot ✓ 1Go par développeur (/home) ✓ 8Go pour l'installateur (/home) 	Contient les sources des COTS pour compilation et les binaires associés.
Intégration	Espace requis : <ul style="list-style-type: none"> ✓ 8Go d'espace disque pour le 	Contient les binaires des COTS récupérés de la plateforme de

	système ✓ 200Mo pour /boot ✓ 3Go pour l'installateur (/home)	développement. Récupère les sources du noyau AMDA-NG depuis le serveur de gestion de configuration.
Production	Espace requis : ✓ 8Go d'espace disque pour le système ✓ 200Mo pour /boot	Contient les binaires des COTS récupérés de la plateforme de développement ainsi que les binaires générés par la plateforme d'intégration continue.

Tableau 1 : Configuration matérielle

5. INSTALLATION DES PREREQUIS

Les pré-requis doivent être installés dans l'ordre défini par ce document. Pour réaliser ces installations, il faut à minima disposer du droit `sudo` ou être administrateur.

Certains éléments sont installés via le gestionnaire de paquets `yum` qui doit être configuré pour accéder aux dépôts CentOS. D'autres éléments sont installés à partir d'archives fournies en tant que COTS.

5.1 Liste des prérequis

Nom	Version	Origine	Dépendances	Justification
apr	1.4.6	Archive livrée		Requis par log4cxx
apr-util	1.5.1	Archive livrée		Requis par log4cxx
binutils	2.22	Archive non livrée Installé par le script bld.sh	cloog 0.17.0 gmp 5.0.5 mpc 1.0 mpfr 3.1.1 ppl 1.0	Requis par boost
boost	1.51.0	Archive non livrée Installé par le script bld.sh	Python 2.7 binutils 2.22 bzip2	Requis par AMDA_Kernel
bzip2-devel	1.0.5-7	Archive livrée		Requis par boost (iostreams)
cairo	1.8.8-3.1	Dépôt CentOS		Requis par le module Plot (device Plplot)
cairo-devel	1.8.8-3.1	Dépôt CentOS		Requis par le module Plot (device Plplot)
cloog	0.17.0	Archive non livrée Installé par le script bld.sh	gmp 5.0.5	Requis par GCC
CMake	2.8.12	Archive livrée		Plateforme de compilation
CSLIM/Fitnessse		Archive livrée		Complément FitNesse pour c++ (environnement de validation)
cURL	7.32.0	Archive livrée		Requis par AMDA_Kernel
GCC	4.7.2	Archive non livrée Installé par le script bld.sh	libiconv 1.14 cloog 0.17.0 gmp 5.0.5	Compilateur

			mpc 1.0 mpfr 3.1.1 ppl 1.0	
glibc	2.12.1-1.107.el6	Archive livrée		Bibliothèque standard C
glibc-devel	2.12.1-1.107.el6	Archive livrée		Bibliothèque standard C
gmp	5.0.5	Archive non livrée Installé par le script bld.sh		Requis par GCC
libarchive	3.1.2	Archive livrée		Requis par AMDA_Kernel
libiconv	1.14	Archive non livrée Installé par le script bld.sh		Requis par GCC
libpng-devel	1.2.49-1 (minimum)	Dépôt CentOS		Requis pour les tests Fitness (environnement de validation)
libXML2	2.7.6-x	Dépôt CentOS		Requis par AMDA_Kernel
Log4cxx	0.10.0	Archive livrée	apr 1.4.6 apr-util 1.5.1	Gestionnaire de traces et logs
mpc	1.0	Archive non livrée Installé par le script bld.sh	gmp 5.0.5 mpfr 3.1.1	Requis par GCC
mpfr	3.1.1	Archive non livrée	gmp 5.0.5	Requis par GCC
OpenJDK	1.6.0	Dépôt CentOS		Requis par FitNesse (environnement de validation)
pango	1.28.1-3	Dépôt CentOS		Requis par le module Plot (device Plplot)
pango-devel	1.28.1-3	Dépôt CentOS		Requis par le module Plot (device Plplot)
PLplot	5.9.10	Archive livrée	cairo-devel pango-devel	Requis pas AMDA_Kernel
ppl	1.0	Archive non livrée Installé par le script bld.sh	gmp 5.0.5	Requis par GCC

Tableau 2 : Liste des prérequis

5.2 Installation des Outils de développement

```
yum groupinstall 'Development Tools'
```

Cette commande installe les packages suivants :

- ✓ Package flex-2.5.35-8.el6.x86_64
- ✓ Package gcc-4.4.7-3.el6.x86_64
- ✓ Package redhat-rpm-config-9.0.3-42.el6.noarch
- ✓ Package rpm-build-4.8.0-32.el6.x86_64
- ✓ Package 1:make-3.81-20.el6.x86_64
- ✓ Package patch-2.6-6.el6.x86_64
- ✓ Package 1:pkgconfig-0.23-9.1.el6.x86_64
- ✓ Package gettext-0.17-16.el6.x86_64
- ✓ Package automake-1.11.1-4.el6.noarch

- ✓ Package bison-2.4.1-5.el6.x86_64
- ✓ Package libtool-2.2.6-15.5.el6.x86_64
- ✓ Package autoconf-2.63-5.1.el6.noarch
- ✓ Package gcc-c++-4.4.7-3.el6.x86_64
- ✓ Package binutils-2.20.51.0.2-5.36.el6.x86_64
- ✓ Package patchutils-0.3.1-3.1.el6.x86_64
- ✓ Package byacc-1.9.20070509-7.el6.x86_64
- ✓ Package indent-2.2.10-7.el6.x86_64
- ✓ Package systemtap-1.8-7.el6.x86_64
- ✓ Package diffstat-1.51-2.el6.x86_64
- ✓ Package elfutils-0.152-1.el6.x86_64
- ✓ Package cvs-1.11.23-15.el6.x86_64
- ✓ Package rcs-5.7-37.el6.x86_64
- ✓ Package subversion-1.6.11-9.el6_4.x86_64
- ✓ Package gcc-gfortran-4.4.7-3.el6.x86_64
- ✓ Package 1:doxygen-1.6.1-6.el6.x86_64
- ✓ Package intltool-0.41.0-1.1.el6.noarch
- ✓ Package git-1.7.1-3.el6_4.1.x86_64
- ✓ Package ctags-5.8-2.el6.x86_64
- ✓ Package cscope-15.6-6.el6.x86_64
- ✓ Package swig-1.3.40-6.el6.x86_64

5.3 Installation des prérequis du noyau AMDA-NG

- ✓ **OpenJDK 1.6.0 (ou supérieur)**

```
yum install java-1.6.0-openjdk
```

- ✓ **libxml2 (2.7.6-x.el6)**

```
yum install libxml2
yum install libxml2-devel
```

Dans la procédure d'installation qui suit, le répertoire contenant ces COTS est désigné par la variable \$COTS.

- ✓ **glibc et glibc-devel**

Vérifier la version de glibc :

```
$ rpm -qa glibc
glibc-2.12-1.107.el6.i686
glibc-2.12-1.107.el6.x86_64
```

Si la version n'est pas celle attendue, saisir la commande suivante :

```
yum install glibc
```

Finalement, installer les packages devel relatifs.

```
yum install glibc-devel
```

- ✓ **bzip2-devel**

```
cd $COTS/yum
rpm -ivh bzip2-devel-1.0.5-7.el6_0.x86_64.rpm
```

✓ **log4cxx**

Les COTS relatifs à log4cxx sont installés dans **/opt/local/**.

```
cd $COTS/log4cxx
tar xzvf apr-1.4.6.tar.gz
tar xzvf apr-util-1.5.1.tar.gz
tar xzvf apache-log4cxx-0.10.0.tgz
cd apr-1.4.6
./configure --prefix=/opt/local
make
sudo make install
cd ../apr-util-1.5.1
./configure --prefix=/opt/local --with-apr=$COTS/log4cxx/apr-1.4.6
make
sudo make install
cd ../apache-log4cxx-0.10.0
./configure --prefix=/opt/local --with-apr=$COTS/log4cxx/apr-1.4.6 --with-apr-util=$COTS/log4cxx/apr-util-1.5.1
export LD_LIBRARY_PATH=/opt/local/lib:$ LD_LIBRARY_PATH
make
sudo make install
```

✓ **cURL**

```
cd $COTS/cURL
tar xzvf curl-7.32.0.tar.gz
cd curl-7.32.0
./configure --prefix=/opt/local
make
sudo make install
```

Attention : cURL est utilisé pour accéder aux timetables distantes via http. Le cas échéant, la configuration d'un éventuel proxy peut-être réalisée en configurant la variable `http_proxy` (la casse est importante) de la manière suivante :

```
export http_proxy=http://[user:password@]<proxy-address>:<proxy-port>
```

Exemple : `export http_proxy=http://amda:pass@proxy.irap.fr:3152`

Cette configuration doit être effectuée de manière systématique en ajoutant cette ligne dans le fichier `.bashrc` par exemple.

✓ **Cmake**

Attention : la compilation de cmake requiert la version de gcc fournie avec CentOs 6.3 : gcc (GCC) 4.4.7 20120313 (Red Hat 4.4.7-3). Si une version plus récente est installée (vérifier la version courante à l'aide la commande `gcc --version`), modifier la variable PATH temporairement de la façon suivante :

1. Sauvegarder la variable PATH : `PATH_SAVE=$PATH`
2. Modifier la variable PATH `PATH=/usr/bin:$PATH`

puis installer cmake en suivant la procédure ci-dessous et rétablir la variable PATH `PATH=$PATH_SAVE`.

```
cd $COTS/cmake
tar xzvf cmake-2.8.12.tar.gz
cd cmake-2.8.12
./configure
make
sudo make install
```

✓ cairo[-devel] et pango[-devel]

```
yum install cairo
yum install cairo-devel
yum install pango
yum install pango-devel
```

Ces bibliothèques sont utilisées pour générer les sorties PNG, PS etc. du module Plot. Elles s'appuient sur des librairies système pour, notamment, afficher les différentes polices d'écriture. Vérifier également la présence des packages de police en saisissant la commande suivante :

```
rpm -qa *font*
```

Cette liste doit contenir a minima les packages suivants (a priori installés par défaut), le cas échéant, les packages manquants devront être installés :

```
xorg-x11-font-utils-7.2-11.el6.x86_64
fontconfig-2.8.0-3.el6.x86_64
liberation-sans-fonts-1.05.1.20090721-5.el6.noarch
dejavu-fonts-common-2.30-2.el6.noarch
ghostscript-fonts-5.50-23.1.el6.noarch
libXfont-1.4.1-2.el6_1.x86_64
fontconfig-devel-2.8.0-3.el6.x86_64
dejavu-sans-fonts-2.30-2.el6.noarch
libfontenc-1.0.5-2.el6.x86_64
fontconfig-2.8.0-3.el6.i686
urw-fonts-2.4-10.el6.noarch
fontpackages-filesystem-1.41-1.1.el6.noarch
liberation-fonts-common-1.05.1.20090721-5.el6.noarch
```

✓ PLplot

```
cd $COTS/PLplot
tar xzvf plplot-5.9.10.tar.gz
cd plplot-5.9.10
mkdir build_dir
cd build_dir
cmake -DCMAKE_INSTALL_PREFIX=/opt/local ..
make
sudo make install
```

✓ GCC et boost

Les prérequis suivants sont installés dans **/opt/tool/gcc/4.7.2/**, ce répertoire est créé au cours de la procédure d'installation. Le script `bld.sh` télécharge (`wget`) puis installe chacun des prérequis. Les éventuels paramètres de proxy pour `wget` sont à configurer dans un fichier `.wgetrc` dans le home de l'utilisateur d'installation.

Créer le répertoire d'installation :

```
sudo mkdir -p /opt/tools/gcc/4.7.2
sudo chmod -R 0777 /opt/tools/gcc/4.7.2
cd /opt/tools/gcc/4.7.2
```

Copier le script d'installation dans ce répertoire :

```
cp $COTS/bld.sh .
```

Puis l'exécuter :

```
chmod a+x bld.sh
./bld.sh | tee -a bld.log
```

Une fois le script exécuté avec succès et les bibliothèques installées, il est possible de supprimer les répertoires /opt/tools/gcc/4.7.2/bld, /opt/tools/gcc/4.7.2/archive et /opt/tools/gcc/4.7.2/src pour récupérer de l'espace disque.

✓ libarchive

```
cd $COTS/libarchive
tar xzvf libarchive-3.1.2.tar.gz
cd libarchive-3.1.2
./configure --prefix=/opt/local
make
sudo make install
```

5.4 Installation des prérequis de validation

✓ Fitness/CSLIM

```
cd $COTS/fitness
tar xzvf fitness-cpp.tar.gz
cd fitness-cpp
./makeAll.sh
```

Remarque : Ignorer le core dump généré par l'exécutable de test.

```
sudo ./install.sh
./cleanAll.sh
```

✓ libpng-devel

```
yum install libpng-devel
```

6. INSTALLATION DU NOYAU AMDA-NG

Le noyau AMDA est installé dans un répertoire \$INSTALL/AMDA_Kernel_Vx.x.x avec x.x.x la version du noyau. Si ce répertoire contient déjà une installation de la même version du noyau AMDA, la renommer.

Les variables d'environnement suivantes doivent être définies :

```
GXX_HOME="/opt/tools/gcc/4.7.2/rtf"
export PATH="$GXX_HOME/bin:/opt/local:$PATH"
export LD_LIBRARY_PATH="$GXX_HOME/lib64:$GXX_HOME/lib:$LD_LIBRARY_PATH"
```

Décompresser l'archive <date>-AMDA_Kernel_V1.1.0.tgz dans le répertoire \$INSTALL :

```
tar xzvf <date>-AMDA Kernel V1.1.0.tgz
```

6.1 Compilation en mode DEBUG

Les binaires sont générés dans `$INSTALL/AMDA_Kernel/build/Debug/`.

6.1.1 Configuration

```
cd $INSTALL/AMDA_Kernel_Vx.x.x
export BUILD_TYPE=Debug
cmake -E make_directory build
cmake -E chdir build cmake -DCMAKE_C_COMPILER=/opt/tools/gcc/4.7.2/rtf/bin/gcc
-DCMAKE_CXX_COMPILER=/opt/tools/gcc/4.7.2/rtf/bin/g++ -DCMAKE_BUILD_TYPE=Debug
..
```

Note : le script `deploy-debug.sh` à la racine du produit exécute ces mêmes commandes, il peut être exécuté depuis le répertoire racine du produit. Cela étant, il n'assure pas de vérification entre chaque étape : il est préférable, lors de la réception d'une nouvelle version du produit, d'exécuter les commandes manuellement. Ce script est essentiellement destiné à simplifier le déploiement d'une version connue.

6.1.2 Compilation

```
cd $INSTALL/AMDA_Kernel_Vx.x.x
cmake -build build
make -C build install VERBOSE=1
```

6.2 Compilation en mode RELEASE

Les binaires sont générés dans `$INSTALL/AMDA_Kernel/build/Release/`.

6.2.1 Configuration

```
cd $INSTALL/AMDA_Kernel_Vx.x.x
cmake -E make_directory build
cmake -E chdir build cmake -DCMAKE_C_COMPILER=/opt/tools/gcc/4.7.2/rtf/bin/gcc
-DCMAKE_CXX_COMPILER=/opt/tools/gcc/4.7.2/rtf/bin/g++ -
DCMAKE_BUILD_TYPE=Release ..
```

Note : le script `deploy-release.sh` à la racine du produit exécute ces mêmes commandes, il peut être exécuté depuis le répertoire racine du produit. Cf. note relative au script `deploy-debug.sh`.

6.2.2 Compilation

```
cd $INSTALL/AMDA_Kernel_Vx.x.x
cmake -build build
make -C build install VERBOSE=1
```

6.3 Arborescence AMDA_Kernel

Une fois installé, l'arborescence de `AMDA_Kernel` est la suivante :

`AMDA_Kernel_Vx.x.x`

- **app-debug**
 - ⇒ Contient les fichiers de configuration pour DDLogin et `AMDA_Kernel` en version Debug

- **src**
 - ⇒ Vide lors de l'installation, destiné à accueillir les fichiers sources générés à la volée par ADMA_Kernel
- **lib**
 - ⇒ Vide lors de l'installation, destiné à accueillir les bibliothèques compilées et générées à la volée par ADMA_Kernel
- **app-release**
 - ⇒ Contient les fichiers de configuration pour DDLogin et AMDA_Kernel en version Release
 - **src**
 - ⇒ Vide lors de l'installation, destiné à accueillir les fichiers sources générés à la volée par ADMA_Kernel
 - **lib**
 - ⇒ Vide lors de l'installation, destiné à accueillir les bibliothèques compilées et générées à la volée par ADMA_Kernel
- **build**
 - **Debug**
 - ⇒ Contient les binaires générés par la compilation en mode Debug
 - **Release**
 - ⇒ Contient les binaires générés par la compilation en mode Release
- **cmake**
 - ⇒ Contient des éléments de génération pour AMDA_Kernel
- **config**
 - ⇒ Contient les schémas XSD de description des fichiers XML de requête et de description de paramètres
- **script**
 - ⇒ Contient les scripts de test de DDLogin
- **src**
 - ⇒ Contient les sources de l'application
- **src_C**
 - ⇒ Contient les sources C de (not. DDLogin)
- **test**
 - ⇒ Contient la base de tests FitNesse ainsi que le code source associé

Versions successives



Version	Date	Émetteur	Vérificateur	Approbateur	Motif
1.5	04/01/2014	N.Boursier	S.Frayssines	R.Patrier	Précision sur les installations des librairies cairo et pango. Mention des scripts d'aide au déploiement. Liste des polices d'écriture attendues sur le système pour le module Plot.
1.4	23/01/2014	N.Boursier	S.Frayssines	R.Patrier	Ajout des installations des librairies cairo et pango.
1.3	09/01/2014	N.Boursier	S.Frayssines	R.Patrier	Corrections mineures de la procédure d'installation.
1.2	08/11/2013	N.Boursier	S.Frayssines	R.Patrier	Modification de la version de cmake et de la procédure d'installation associée. Modification de la procédure d'installation de AMDA_Kernel pour prendre en compte la nouvelle version de cmake. Montée de version de Plplot – sans modification de sa procédure d'installation.
1.1	08/10/2013	N.Boursier	S.Frayssines	R.Patrier	Description des installations des librairies cURL, PLplot et libarchive.
1.0	16/09/2013	N. Boursier	S. Frayssines	R.Patrier	Création du document

Diffusion



*Ce document est mis à disposition sous forme informatique sur serveur.
Il n'est donc pas formellement diffusé sous forme papier.*