



**DOSSIER DE
CONTROLE DES
INTERFACES DU
NOYAU AMDA-NG**

Rédigé par : <i>Architecte : Gaëtan SCHNELLER</i> <i>Chef de projet AKKA : Freddy CASIMIR</i>	Diffusé à : CNES/ IRAP
Approuvé par : Chef de projet CNES : DUFOURG Nicolas	

LISTE DES MODIFICATIONS DU DOCUMENT

Vers.	Date	Paragraphe	Description de la modification
01.00	06/11/2012		Création du document
01.01	16/11/2012		Prise en compte FEPS ND6
01.02	29/11/2012		Prise en compte FEPS CNES
01.03	15/01/2013		Mise à jour Release 2 Sprint 3
01.04	05/02/2013		Mise à jour Release 3 Sprint 1

SOMMAIRE

1	INTRODUCTION	5
2	ARCHITECTURE	6
3	INTERFACES EXTERNES	7
3.1	Ligne de commande	7
3.1.1	EXE_AMDA_Kernel	7
3.1.2	amdaXMLRequestorTool	8
3.1.3	amdaParameterGenerator	8
3.2	Fichiers de propriété	8
3.2.1	Fichier de propriété du noyau AMDA.....	8
3.2.2	Fichier de propriété de log4cxx	9
3.3	Interface DD server.....	10
3.4	Journalisation	10
3.5	fichierS XML d'un paramètre	10
3.5.1	XSD de description du fichier XML d'un paramètre.....	10
3.5.2	XSD de description du fichier XML d'un paramètre, partie get type « amdaParam »	12
3.5.3	XSD de description du fichier XML d'un paramètre, partie get type DDBASE	12
3.6	Description XSD d'une requête	13
3.6.1	XSD Général d'une requête	13
3.6.2	XSD interval nœud d'un times.....	15
3.6.3	XSD download nœud d'un output.....	15
3.7	Description des plugins	16
3.8	Génération de fichiers temporaires.....	17
3.8.1	Nettoyage	17
3.9	Description des messages d'erreur	17
4	DOCUMENTS APPLICABLES ET DE REFERENCE (A/R)	19
5	GLOSSAIRE ET ABREVIATIONS.....	20
5.1	Glossaire	20

5.2 Abréviations..... 20

1 INTRODUCTION

L'objectif du document est de décrire les interfaces externes du noyau AMDA.

Ce document vient en complément du document d'architecture du noyau d'AMDA [A1] et du dossier de conception [A2].

2 ARCHITECTURE

Voir le dossier d'architecture du noyau d'AMDA [A1], en particulier le paragraphe 3.

3 INTERFACES EXTERNES

Le noyau AMDA se présente sous la forme d'un ensemble de bibliothèques et d'exécutables qui les utilisent.

Ces exécutables, EXE_AMDA_Kernel, amdaParameterGenerator et amdaXMLRequestorTool, sont utilisables en ligne de commande.

Ils ont en commun la lecture de deux fichiers de propriétés qui doivent être présents dans leur répertoire d'exécution et génèrent :

- facultativement un fichier de sortie (suivant la requête)
- au moins un fichier de journalisation.

Ils retournent :

- 0 en cas de fonctionnement nominal
- un code d'erreur compris entre -255 et +255 en cas de fonctionnement non nominal.

Ils diffèrent sur le moyen de lire une requête :

- EXE_AMDA_Kernel et amdaParameterGenerator lisent leurs requêtes à partir des arguments de la ligne de commande
- amdaXMLRequestorTool lit sa requête à partir d'un fichier XML donné en argument.

Les interfaces externes du noyau AMDA sont décrites ci-après.

3.1 LIGNE DE COMMANDE

3.1.1 EXE_AMDA_Kernel

EXE_AMDA_Kernel [--help|-h](--version|-v) | (--param|-p) paramName (--startTime|-s) startTime (--intervalTime|-d) intervalTime (--format|-f) format [--work-path|-w) path]

- --help|-h: this message

--version|-v: print version number

- --param|-p paramName: parameter name must reference a paramName.xml file
- --startTime|-s startTime: start time in YYYYDDHMMSSMLS
- --intervalTime|-d intervalTime: interval of time in YYYYDDHMMSSMLS
- --format|-f iso/dd/double: format type of time in output file
- --sampling|-r float: sampling in se
- --sampling-mode|-m classic: sampling mode only classic is implemented
- --work-path|-w path: work path where file output is generated "." by default

3.1.2 amdaXMLRequestorTool

amdaXMLRequestorTool [--help|-h](--version|-v) | [--work-path|-w] path] requesteFile.xml

- --help|-h: this message
- --version|-v: print version number
- --work-path|-w path: work path where file output is generated "." by default
- requesteFile.xml : see §3.6

3.1.3 amdaParameterGenerator

- amdaParameterGenerator (--help|-h)((--version|-v)((--param|-p) paramName)+)
 - Allowed options:
 - -h [--help] Produce help message
 - -v [--version] Program version
 - -p [--parameter] arg Parameter(s) name

3.2 FICHIERS DE PROPRIETE

3.2.1 Fichier de propriété du noyau AMDA

Le fichier principal de propriété doit être dans le répertoire d'exécution et avoir comme nom « app.properties ». C'est un fichier ASCII de type clef=valeur suivi d'un retour chariot.

Le caractère # en première colonne du fichier indique que le reste de la ligne est un commentaire.

Il contient :

- app.log4cxx.configfile : nom du fichier de configuration du module de journalisation
- app.param.path : chemin absolu ou relatif du répertoire contenant les fichiers xml décrivant les paramètres
- app.plugin : AMDA plugin directory
- le paramétrage de la procédure de génération de code :
 - app.process.src : le répertoire de destination de sources générés
 - app.process.lib : répertoire de destination des bibliothèques obtenues
 - app.process.CXX_COMPILER : le compilateur utilisé
 - app.process.CMAKE_CXX_FLAGS : option de compilation de la bibliothèque
 - app.process.INCLUDE : les includes "-I" de première nécessité
 - app.process.LIB : les bibliothèques de première nécessité

Exemple :

```
#Configuration file of Log4Cxx logger component
app.log4cxx.configfile=log4cxx.config
```



```
app.param.path=../config/DataBaseParameters
app.parameter.xsd=../config/DataBaseParameters/xsd/parameter/all.xsd
app.request.xsd=../config/DataBaseParameters/xsd/request/all.xsd
# Directory of AMDA plugin
app.plugin=../../build/Debug/plugin/
# Process parameters
app.process.src=src
app.process.lib=lib
app.process.CXX_COMPILER=g++
app.process.CMAKE_CXX_FLAGS=-g -std=c++0x -fPIC -Wall -ggdb -DLINUX -Dlinux -
D_REENTRANT -malign-double -pthread
app.process.INCLUDE=-I../src/InternLib -I../src/Common -I../src/Parameters -
I../src/DD_Client_Lib -I../src/helpers -I/opt/local/include -
I/opt/local/include/log4cxx
app.process.LIB=-L../build/Debug/lib/ -lParameters
```

3.2.2 Fichier de propriété de log4cxx

Ce fichier de configuration respecte la syntaxe des fichiers de propriétés de log4j. Une documentation est disponible sur la page suivante:

http://pic.dhe.ibm.com/infocenter/p8docs/v4r5m1/index.jsp?topic=%2Fcom.ibm.p8.doc%2Fdeveloper_help%2Fcontent_engine_api%2Fguide%2Flogging_procedures.htm

Exemple :

```
log4j.rootLogger=debug, stdout, F

log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout

# Pattern to output the caller's file name and line number.
log4j.appender.stdout.layout.ConversionPattern=%5p [%t] (%F:%L) - %m%n

log4j.appender.F=org.apache.log4j.FileAppender
log4j.appender.F.File=example.log
log4j.appender.F.Append=false
```

```
log4j.appender.F.layout=org.apache.log4j.PatternLayout
log4j.appender.F.layout.ConversionPattern=%p %t %c - %m%n
```

3.3 INTERFACE DD SERVER

Le principe de récupération des données sur DD server reste inchangé par rapport à la version précédente d'AMDA. Il utilise une librairie (DD_client) qui communique via des sockets TCP/IP.

Quatre principales étapes sont nécessaires pour récupérer des données:

- OPENINSREQ : ouvre un dataset et retourne un identifiant unique
- CLOSEINSREQ : ferme un dataset (identifiant en paramètre)
- TIMESETREQ: positionne un pointeur sur le Start Time (Start Time et identifiant en paramètre)
- DATAGETREQ: à partir du Start Time positionné, déplace le pointeur durant la requête par intervalle de temps (indiqué en paramètre). Le retour est un flux dans une socket. En en-tête sont indiqués le nombre d'enregistrements et la taille des paramètres, suivent les données.

La description détaillée est indiquée au : <http://manunja.cesr.fr/~budnik/AMDANEW/DDHTML/DOC/>

3.4 JOURNALISATION

Un fichier de journalisation (log), exemple.log, est généré dans le répertoire d'exécution. Le format, l'emplacement, le niveau et le type de journalisation du fichier peuvent être modifiés via le fichier de propriété log4cxx.

3.5 FICHIERS XML D'UN PARAMETRE

Le noyau AMDA a comme paramètre, le nom du paramètre qu'il doit traiter.

Cette chaîne de caractère, « paramName », référence un fichier xml de nom « paramName.xml » qui doit se trouver dans le répertoire référencé par « app.param.path » dans le fichier de propriété du noyau AMDA.

Les fichiers décrivant les paramètres sont donc au format xml et respectent les schémas XSD décrits dans les paragraphes ci-dessous.

Ces fichiers XSD sont placés sous : « .\DataBaseParameters\xsd » (relativement au répertoire d'exécution).

3.5.1 XSD de description du fichier XML d'un paramètre

Cette description ne contient que le squelette d'un paramètre. Les éléments de configuration (exemple : « get ») sont chargés et validés à partir d'un second XSD lors du traitement de l'élément.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```

<xs:complexType name="GetterType">
  <xs:attribute name="name" type="xs:string" />
</xs:complexType>

<xs:element name="Getter" type="GetterType" abstract="true" />

<xs:element name="param">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="time_resolution" type="xs:integer"
        minOccurs="0" maxOccurs="1" />
      <xs:element name="fill_value" type="xs:double" minOccurs="0"
/>
      <xs:element name="tensor_order" type="xs:integer"
        minOccurs="0" />
      <xs:element name="get">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="Getter" minOccurs="1"
maxOccurs="unbounded" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="clbProcess" type="xs:string" minOccurs="0"
maxOccurs="unbounded" />
      <xs:element name="process" type="xs:string" />
      <xs:element name="output">
        <xs:complexType>
          <xs:sequence>
            <xs:any minOccurs="0" maxOccurs="unbounded"
              processContents="skip" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```

        </xs:element>

    </xs:sequence>

    <!--xs:attribute name="xml:id" type="xs:ID" use="required"/ -->
    <xs:attribute name="id" type="xs:ID" use="required" />
    <!--xs:attribute ref="xml:id" use="required" /-->

</xs:complexType>
</xs:element>
</xs:schema>

```

Les éléments `get`, `process` et `output` sont génériques et leurs différentes descriptions sont données ci-après.

3.5.2 XSD de description du fichier XML d'un paramètre, partie get type « amdaParam »

Quand l'élément « *amdaParam* » est rencontré sous l'élément « `get` », c'est le XSD suivant qui est utilisé pour la validation du nœud.

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:include schemaLocation="parameter.xsd" />
    <xs:element name="amdaParam" substitutionGroup="Getter" type="GetterType"/>
</xs:schema>

```

3.5.3 XSD de description du fichier XML d'un paramètre, partie get type DDBASE

Quand l'élément « *baseParam* » est rencontré sous l'élément « `get` », c'est le XSD suivant qui est utilisé pour la validation du nœud.

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:include schemaLocation="parameter.xsd" />

    <xs:complexType name="clb-type">
        <xs:attribute name="name" type="xs:string" />
    </xs:complexType>

    <xs:complexType name="baseParamType">
        <xs:sequence>

```

```

        <xs:element name="tensor_order" type="xs:integer"
            minOccurs="0" maxOccurs="1" />
        <xs:element name="clb" type="clb-type" minOccurs="0"
            maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required" />
</xs:complexType>

<xs:complexType name="vi-type">
    <xs:complexContent>
        <xs:extension base="GetterType">
            <xs:sequence>
                <xs:element name="baseParam" type="baseParamType"
                    minOccurs="1" maxOccurs="unbounded" />
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:element name="vi" substitutionGroup="Getter" type="vi-type" />
</xs:schema>

```

3.6 DESCRIPTION XSD D'UNE REQUETE

3.6.1 XSD Général d'une requête

Cette description ne contient que le squelette d'une requête. Les éléments de configuration tels que « TimeElement » et « OutputElement » sont décrits dans des XSD correspondants à des interprétations réelles telles que : « interval » ou « download ».

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

    <xs:element name="TimeElement" abstract="true" />

```

```

<xs:element name="OutputElement" abstract="true" />

<xs:element name="request">
  <xs:complexType>
    <xs:sequence>

      <xs:element name="params">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="param" minOccurs="1"
maxOccurs="unbounded">
              <xs:complexType>
                <xs:attribute name="id"
type="xs:string" />
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>

      <xs:element name="times">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="TimeElement" minOccurs="1"
maxOccurs="unbounded" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>

      <xs:element name="outputs">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="OutputElement"
minOccurs="1" maxOccurs="unbounded" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        </xs:sequence>
    </xs:complexType>
</xs:element>

</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

3.6.2 XSD interval nœud d'un times

Quand l'élément « *interval* » est rencontré sous l'élément « times » et implémente l'élément abstrait « TimeElement », c'est le XSD suivant qui est utilisé pour la validation du nœud.

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:include schemaLocation="request.xsd" />

    <xs:element name="interval" substitutionGroup="TimeElement" >
        <xs:complexType>
            <xs:sequence>
                <xs:element name="startTime" type="xs:string" minOccurs="1"
maxOccurs="1"/>
                <xs:element name="timeInterval" type="xs:string"
minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

3.6.3 XSD download nœud d'un output

Quand l'élément « *download* » est rencontré sous l'élément « outputs », c'est le XSD suivant qui est utilisé pour la validation du nœud.

```

<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="request.xsd" />

  <xs:element name="download" substitutionGroup="OutputElement" >
    <xs:complexType>
      <xs:sequence>
        <xs:element name="timeFormat" type="xs:string" minOccurs="0"
maxOccurs="1"/>
        <xs:element name="param" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="index" type="xs:integer"
minOccurs="0" maxOccurs="unbounded"/>
              <xs:element name="calibration_info"
type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="id" type="xs:string"
use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="precision" type="xs:string" />
    </xs:complexType>
  </xs:element>
</xs:schema>

```

3.7 DESCRIPTION DES PLUGINS

AMDA-Kernel suit une conception modulaire sous forme de Plugin. L'application accepte l'ajout de nouvelles fonctionnalités sous forme de bibliothèques « *.so » et de fichier d'interface .hh. Ces bibliothèques sont placées sous le répertoire indiqué par le fichier propriété de l'application « app.plugin ». Ces Plugins sont de deux types et doivent respecter des critères stricts : voir [A2: Composant Plugin].

3.8 GENERATION DE FICHIERS TEMPORAIRES

AMDA-Kernel, dans le but de faire des calculs dynamiques, crée des fichiers « .cc » dans le but de les compiler en librairie dynamique « .so » qui sont chargés dynamiquement.

Les fichiers « .cc » sont temporaires et générés dans un sous-répertoire du répertoire indiqué par le fichier propriété de l'application « app.process.src ». Ce sous répertoire (nomé NomParamXXXXXX avec XXXXX prenant une valeur unique) sera supprimé à la fin de la génération des « .so ».

Les bibliothèques dynamiques « .so » sont générées sous le répertoire indiqué par le fichier propriété de l'application « app.process.lib ».

Elles sont associées à un paramètre (fichier nomParam.xml) et ont comme nom : « nomParam.so ».

Elles sont conservées et peuvent être réutilisées à chaque exécution.

3.8.1 Nettoyage

Dans le cas de la suppression d'un fichier paramètre xml, le fichier .so correspondant doit être supprimé.

Dans le cas d'une modification de la version des exécutables AMDA, toutes les bibliothèques « .so » doivent être supprimées.

3.9 DESCRIPTION DES MESSAGES D'ERREUR

L'exécutable AMDA retourne 0 s'il n'y a pas d'erreur, comme le recommande UNIX. En cas d'erreur il retourne le code décrit ci-dessous :

```
AMDA_EXIT_OK = 0,  
AMDA_PARSE_COMMAND_LINE_KO = 1, ///  
AMDA_TYPE_DATA_UNKNOWN = 2, ///  
AMDA_FILEPROPERTIES_ERR = 3, ///  
AMDA_INFORMATION_PARAM_ERR = 4, ///  
AMDA_INFORMATION_REQUEST_ERR = 5, ///  
AMDA_PARAM_OUTPUT_ERR = 6, ///  
AMDA_PROCESS_ERR = 7, ///  
AMDA_PARAM_NOT_FOUND_ERR = 8, ///  
AMDA_PARAM_SOME_ERR = 9, ///  
AMDA_OPER_NOT_ALLOWED = 10, ///  
AMDA_ERROR_UNKNOWN = 100,
```

```

/*----- Errors of DD_serveur -----*/
AMDA_ERROR_NOCONNECTION =-1, // No connection with DD_server
AMDA_ERROR_TOOMANYREQ   =-2, // too many request
AMDA_ERROR_REQSENDERR   =-3,
AMDA_ERROR_REPLYRECERR  =-4, // receive an server error
AMDA_ERROR_NOID         = -5, //Id of request not valid
AMDA_ERROR_NOEFFFILE    =-6, //Data base not consistent
AMDA_ERROR_NOTIMESFILE  =-7, //Data base not consistent
AMDA_ERROR_NOINFOFILE   =-8, //Data base not consistent
AMDA_ERROR_NOVAR        = -9, // Variable unknown
AMDA_ERROR_OUTOFTIME    =-10, //Out of data time
AMDA_ERROR_CACHERR      =-11, /** The error ocured when program manipulated the
cach */
AMDA_ERROR_CACHTOOREC   =-33 ,/** Off files in the CASH are created less then 5 sec
ago */
AMDA_ERROR_WAITEXTCALL  =-34, /** Server Waits external call finish */
AMDA_ERROR_TRYAGAIN     = -35, /** Times file is locked. This request should be
repeated again */
AMDA_ERROR_NODATAATTIME =-36, /** Requested time is inside NODATA time interval */
AMDA_ERROR_GAPISSMALL   =-37, /** Gap is too small to call external archive */
AMDA_ERROR_CTIMEISSHIFTED =-38 ,/** The request was in NoData interval, try again
with time at the end */
AMDA_ERROR_DATAFILEERR  = -12, /** The error while manipulate with data file */
AMDA_ERROR_ENVIRERROR   =-13 , /** Error in Environments */
AMDA_ERROR_NOPOINTER    =-14 , /** The pointer points nowhere, SetTime were failed */
AMDA_ERROR_INTSERERR    =-15 , /** Server internal error */
AMDA_ERROR_TIMEINEMPTY  =-16, /** No data for specified time interval */
AMDA_ERROR_NOEXEC       = -17 , /** No executable */
/*----- Errors of DD_serveur(Access) -----*/
AMDA_ERROR_OPERMISSION  = -100, /** Permission denied to connect with DD_server*/
AMDA_ERROR_NOHOST       = -101, /** Permission denied to connect with DD_server*/
AMDA_ERROR_NNOUSERSFILE = -102

```

4 DOCUMENTS APPLICABLES ET DE REFERENCE (A/R)

A/R	Référence	Titre
A1	CDPP-AR-32500-382-SI	Dossier d'architecture du noyau d'AMDA-NG
A2	CDPP-CD-32500-436-SI	Dossier de conception du noyau d'AMDA-NG

5 GLOSSAIRE ET ABREVIATIONS

5.1 GLOSSAIRE

Terme	Définition
DD server	serveur permettant l'accès à la base de données des paramètres basiques AMDA

5.2 ABREVIATIONS

Abréviation	Nom détaillé