

# Noyau AMDA-NG - 2nde partie



## DOSSIER DE CONTROLE DES INTERFACES

Référence : CDPP-IF-32500-458-CS

Version 1.4 du 23/01/2014

## Tableau des signatures



Nom	Fonction	Date	Signature
<b>Préparé par :</b> N.Boursier		23/01/2014	
<b>Vérifié par :</b> S. Frayssines		23/01/2014	
<b>Approuvé par :</b> R.Patrier		23/01/2014	

## Sommaire



1.	Introduction.....	7
2.	Documents applicables.....	7
3.	Documents de référence.....	7
4.	Présentation du Système.....	7
4.1	Nomenclature des interfaces.....	8
5.	Liste des Interfaces.....	9
5.1	Interfaces externes.....	9
5.1.1	Fichier de configuration du noyau AMDA.....	9
5.1.2	Journalisation.....	9
5.1.3	Données externes.....	9
5.2	Interfaces internes.....	9
6.	Description des interfaces externes.....	10
6.1	Fichiers de configuration du noyau AMDA.....	10
6.1.1	IF_E_ADM_CONF : app.properties.....	10
6.1.1.1	Description détaillée.....	10
6.1.2	IF_E_ADM_AMDA_PROP : amda.properties.....	11
6.1.2.1	Description détaillée.....	11
6.1.3	IF_E_ADM_PLOT_CONF.....	11
6.1.3.1	Description détaillée.....	12
6.1.3.1.1	Section default.....	12
6.1.3.1.2	Section colormap.....	13
6.2	Journalisation.....	14
6.2.1	IF_E_AMD_CONF-LOG.....	14
6.2.1.1	Description détaillée.....	14
6.2.1.2	Exemple.....	14
6.2.2	IF_E_KERNEL_LOG.....	15
6.3	Données externes.....	15
6.3.1	IF_E_DDSEV_GET-PARAM.....	15
6.3.1.1	Description détaillée.....	16
6.3.2	IF_E_WEBINT_PARAM.....	16
6.3.2.1	Schémas XSD.....	16
6.3.2.1.1	all.xsd.....	16
6.3.2.1.2	parameter.xsd.....	17
6.3.2.1.3	getddbase.xsd.....	18

6.3.2.1.4	getamdaparam.xsd.....	18
6.3.2.2	Exemples.....	18
6.3.2.2.1	Description de paramètre AMDA.....	18
6.3.2.2.2	Description de paramètre de base.....	19
6.3.3	IF_E_WEBINT_REQUEST.....	19
6.3.3.1	Schémas XSD.....	19
6.3.3.1.1	all.xsd.....	19
6.3.3.1.2	request.xsd.....	20
6.3.3.1.3	interval.xsd.....	21
6.3.3.1.4	download.xsd.....	21
6.3.3.1.5	intervalTrue.xsd.....	22
6.3.3.1.6	dataMining.xsd.....	22
6.3.3.1.7	plot.xsd.....	23
6.3.3.1.8	timePlot.xsd.....	30
6.3.3.1.9	xyPlot.xsd.....	31
6.3.3.1.10	postProcessing.xsd.....	32
6.3.4	IF_E_DEV_PLUGIN-LIB.....	32
6.3.5	IF_E_DEV_PLUGIN-FONC.....	33
6.3.5.1	Description détaillée.....	33
7.	Description des interfaces internes.....	33
7.1.1	IF_I_KERNEL_CC-TMP.....	33
7.1.1.1	Description détaillée.....	34
7.1.2	IF_I_KERNEL_PLUGIN-LIB.....	34
7.1.2.1	Description détaillée.....	35

## Table des illustrations



## Liste des figures

Figure 1 : Interfaces externes du système \_\_\_\_\_ 8 |

## 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
CSSI	Communication et Systèmes – Systèmes d'Information.
IRAP	Institut de Recherche en Astrophysique et Planétologie
DV	<b>Plan de développement</b>
NFS	<b>Network File System</b>

# 1. INTRODUCTION

Ce document est initialisé à partir du document [DR03]. Il décrit les interfaces du noyau AMDA-NG. Il est structuré comme suit :

- ✓ Le chapitre **4. Présentation du Système** présente une vue synthétique et globale du système afin de mettre en évidence les interfaces.
- ✓ Le chapitre **5. Liste des Interfaces** liste l'ensemble des interfaces internes et externes du système.
- ✓ Le chapitre **6. Description des interfaces externes** décrit les interfaces externes du système.
- ✓ Enfin, le chapitre **7. Description des interfaces internes** décrit les interfaces internes du système.

## 2. DOCUMENTS APPLICABLES

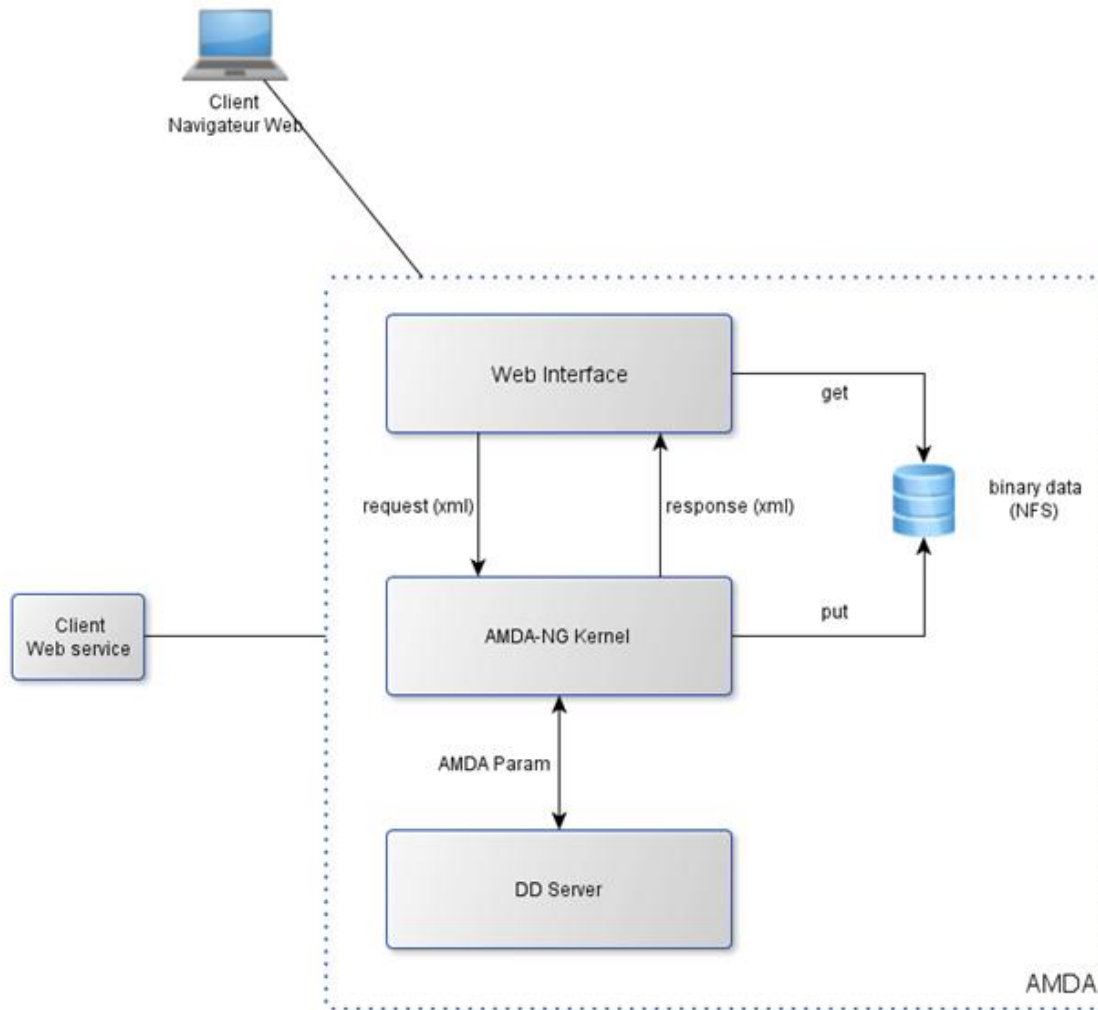
DA01	Consultation AC-IS N°DAJ/AR/EO-2013.08449 "Développement du noyau AMDA-NG - Seconde Partie". CDPP-CO-32500-452-CNES, 06/05/2013, Ed. 01 Rev 00.
DA02	Spécification de besoins techniques pour la seconde prestation du nouveau noyau AMDA. CDPP-ST-32500-451-CES, 13/05/2013, Ed. 01 Rev 00
DA03	Cahier 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. PRESENTATION DU SYSTEME

Le schéma ci-dessous rappelle l'architecture du système et met en évidence les interfaces externes qu'il met en œuvre.



**Figure 1 : Interfaces externes du système**

Un utilisateur saisit une demande via un client IHM web (EXT-JS). Cette demande est traitée par l'interface « Web Interface » qui la transmet sous la forme d'une requête XML au noyau AMDA (AMDA-NG Kernel). Ce dernier traite la requête en s'appuyant sur le serveur DD Server pour obtenir les données relatives aux valeurs des paramètres AMDA puis retourne une réponse, au format XML également, à l'interface « Web Interface ». Finalement, l'interface « Web Interface » émet la réponse à destination du poste client.

## 4.1 Nomenclature des interfaces

La nomenclature utilisée pour identifier les interfaces dans ce document respecte le format suivant :

**IF\_<type>\_<responsable>\_<nom>**

avec :

<type> : I (Interne) ou E (Externe)

<responsable> : identifiant du producteur de l'interface. Les valeurs possibles pour ce champ sont :

AMD : Administrateur du système

DEV : Développeur de plugins

WEBINT : Interface Web (à venir)

DDSERV : DDServer

KERNEL: Noyau AMDA-NG

<nom> : identifie de manière unique l'interface selon les fonctionnalités offertes.



## 5. LISTE DES INTERFACES

### 5.1 Interfaces externes

#### 5.1.1 Fichier de configuration du noyau AMDA

Identification	Description	Producteur	Consommateur
IF_E_ADM_CONF	Fichier de configuration du noyau AMDA-NG	ADM	KERNEL
IF_E_ADM_AMDA_PROP	Fichier de propriétés AMDA	ADM	KERNEL
IF_E_ADM_PLOT_CONF	Fichier de configuration des valeurs par défaut pour le module Plot	ADM	KERNEL

#### 5.1.2 Journalisation

Identification	Description	Producteur	Consommateur
IF_E_AMD_CONF-LOG	Fichier de configuration log4cxx	ADM	KERNEL
IF_E_KERNEL_LOG	Fichier de logs	KERNEL	-

#### 5.1.3 Données externes

Identification	Description	Producteur	Consommateur
IF_E_DDSEV_GET-PARAM	Interface de récupération des données	DDSEV	KERNEL
IF_E_WEBINT_PARAM	Fichier XML de description d'un paramètre	WEBINT	KERNEL
IF_E_WEBINT_REQUEST	Fichier XML de description d'une requête	WEBINT	KERNEL
IF_E_DEV_PLUGIN-LIB	Plugin proposant de nouvelles fonctionnalités	DEV	KERNEL
IF_E_DEV_PLUGIN-FONC	Plugin proposant une fonction mathématique	DEV	KERNEL

### 5.2 Interfaces internes

Identification	Description	Producteur	Consommateur
IF_I_KERNEL_CC-TMP	Fichiers de code source temporaires.	KERNEL	KERNEL
IF_I_KERNEL_PLUGIN-LIB	Plugin généré à la volée et responsable d'un calcul sur un paramètre donné	KERNEL	KERNEL

## 6. DESCRIPTION DES INTERFACES EXTERNES

### 6.1 Fichiers de configuration du noyau AMDA

#### 6.1.1 IF\_E\_ADM\_CONF : app.properties

Interface IF_E_ADM_CONF	
<b>Nomenclature</b> : app.properties	<b>Type</b> : fichier ASCII
Description	
<b>Rôle</b> : Localiser les emplacements des interfaces fichiers, des plugins et paramétrer la procédure de génération de code.	
Producteur	Consommateur
ADM : Administrateur	KERNEL : Noyau AMDA-NG
Espaces de stockage	
Répertoire d'exécution du noyau AMDA-NG	
Production	
<b>Fréquence de génération</b> : Lors de la configuration du noyau.	
<b>Portée</b> : KERNEL (Noyau AMDA)	<b>Volumétrie</b> : quelques Ko

##### 6.1.1.1 Description détaillée

Ce fichier contient une liste de propriétés sous la forme clé=valeur. Le caractère # en début de ligne indique que le reste de la ligne est un commentaire.

Il contient les propriétés de localisation suivantes :

- ✓ 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.parameter.xsd : chemin absolu ou relatif du fichier XSD de description des paramètres
- ✓ app.request.xsd : chemin absolu ou relatif du fichier XSD de description des requêtes
- ✓ app.plugin : répertoire de dépôt des plugins AMDA plugin directory
- ✓ app.plot.configfile : nom du fichier de configuration du module Plot

Ainsi que le paramétrage de la procédure de génération de code :

- ✓ app.process.src : le répertoire de destination de fichiers source 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

## 6.1.2 IF\_E\_ADM\_AMDA\_PROP : amda.properties

Interface IF_E_ADM_AMDA_PROP	
<b>Nomenclature</b> : amda.properties	<b>Type</b> : fichier ASCII
Description	
<b>Rôle</b> : Définit les valeurs de propriétés du noyau AMDA	
Producteur	Consommateur
ADM : Administrateur	KERNEL : Noyau AMDA-NG
Espaces de stockage	
Répertoire d'exécution du noyau AMDA-NG	
Production	
<b>Fréquence de génération</b> : Lors de la configuration du noyau.	
<b>Portée</b> : KERNEL (Noyau AMDA)	<b>Volumétrie</b> : quelques Ko

### 6.1.2.1 Description détaillée

Ce fichier contient une liste de propriétés sous la forme clé=valeur. Le caractère # en début de ligne indique que le reste de la ligne est un commentaire.

Il contient les propriétés suivantes :

- ✓ `version` : version du noyau AMDA courant
- ✓ `createdby` : copyright affiché sur les outputs de type Plot

## 6.1.3 IF\_E\_ADM\_PLOT\_CONF

Interface IF_E_ADM_PLOT_CONF	
<b>Nomenclature</b> : configurable via IF_E_ADM_CONF : app.properties	<b>Type</b> : fichier XML
Description	
<b>Rôle</b> : Définit des valeurs par défaut exploitées par le module Plot	
Producteur	Consommateur
ADM : Administrateur	KERNEL : Noyau AMDA-NG
Espaces de stockage	
Répertoire d'exécution du noyau AMDA-NG	
Production	
<b>Fréquence de génération</b> : Lors de la configuration du noyau.	
<b>Portée</b> : KERNEL (Noyau AMDA)	<b>Volumétrie</b> : quelques Ko

## 6.1.3.1 Description détaillée

### 6.1.3.1.1 Section default

Cette section définit les valeurs par défaut des éléments de plot tels que la page, les panels, les axes et les différents types de plot. Les tableaux suivants détaillent le rôle de ces valeurs.

Page		
Attribut	Valeurs possibles	Rôle
dimension	ISO A4, US letter	Format de la page
orientation	landscape, portrait	Orientation de la page
mode	color, grayscale	Page couleur/échelle de gris
dpi	Entier positif	dot-per-inch, dimensionne les outputs aux formats png et svg
font.name	sans-serif, serif, monospace, script	Police par défaut de la page
font.size	Entier positif	Taille de police par défaut de la page
margin.x	Entier positif en mm	Marge à gauche et à droite de la page
margin.y	Entier positif en mm	Marge en haut et en bas de la page

Panel		
Attribut	Valeurs possibles	Rôle
resolution	Entier positif	Nombre de points max par série
backgroundColor	Une couleur définie par le triplet d'entiers [r,g,b]	Couleur de fond du panel
charSizeUnit	Entier positif	Marge autour de la zone de plot
title	Texte	Titre du panel, par défaut startdate - stopdate
title.position	top, bottom	Position du titre sur le panel
title.align	left, right, center	Alignement du titre dans le panel
timeAxis.id	Texte	Identifiant de l'axe du temps (cf. élément timePlot)
timeAxis.format	dd/mm/yy, ddd/yy	Format des dates dans les libellés de l'axe du temps
timeAxis.reverse	true, false	Axe inversé ou non
timeAxis.color	Une couleur définie par le triplet d'entiers [r,g,b]	Couleur de l'axe du temps
timeAxis.position	top, bottom	Position de l'axe du temps sur le plot
timeAxis.thickness	Entier positif	Epaisseur du trait de l'axe du temps
timeAxis.legend.text	Texte	Texte de la légende de l'axe du temps
X[y]Axis.scale	linear, logarithmic	Type d'échelle d'un axe numérique en abscisse[ordonnée]
X[y]Axis.reverse	true, false	Axe inversé ou non
X[y]Axis.color	Une couleur définie par le triplet d'entiers [r,g,b]	Couleur d'un axe numérique en abscisse[ordonnée]

X[y]Axis.position	top, bottom	Position d'un axe numérique en abscisse[ordonnée] sur le plot
X[y]Axis.thickness	Entier positif	Epaisseur du trait d'un axe numérique en abscisse[ordonnée]

TimePlot		
Attribut	Valeurs possibles	Rôle
xAxis	Texte	Identifiant de l'axe du temps (cf. attribut panel.timeAxis.id)
line.type	no, line	Type de ligne (no = pas de ligne)
line.style	plain, dot, long-spaced-dot, long-short-dot	Style de ligne
line.width	Entier positif	Epaisseur de la ligne
symbol.type	no, dot, plus, wildcard, circle, crux, square, triangle, crux-in-circle, dot-in-circle, square-2, diamond, star, christian-crux, david-star, full-square, full-circle, full-star	Forme du symbole
symbol.size	Entier positif, en mm	Taille du symbole
symbol.color	Une couleur définie par le triplet d'entiers [r,g,b]	Couleur du symbole

timePlot/xyPlot		
Attribut	Valeurs possibles	Rôle
xAxis	Texte	Identifiant de l'axe du temps (cf. attribut panel.timeAxis.id)
line.type	no, line	Type de ligne (no = pas de ligne)
line.style	plain, dot, long-spaced-dot, long-short-dot	Style de ligne
line.width	Entier positif	Epaisseur de la ligne
symbol.type	no, dot, plus, wildcard, circle, crux, square, triangle, crux-in-circle, dot-in-circle, square-2, diamond, star, christian-crux, david-star, full-square, full-circle, full-star	Forme du symbole
symbol.size	Entier positif, en mm	Taille du symbole
symbol.color	Une couleur définie par le triplet d'entiers [r,g,b]	Couleur du symbole

### 6.1.3.1.2 Section colormap

Cette section définit les différentes palettes de couleurs sélectionnables par l'utilisateur.

colormap		
Attribut	Valeurs possibles	Rôle
path	Texte	Chemin relatif vers le répertoire contenant les fichiers descriptifs

		des palettes de couleurs
grayscale[color].default	Entier positif	Index de la palette utilisée par défaut
grayscale[color].file.index	Entier positif	Index d'une palette, unique
grayscale[color].file	Texte	Nom du fichier décrivant la palette

## 6.2 Journalisation

### 6.2.1 IF\_E\_AMD\_CONF-LOG

Interface IF_E_AMD_CONF-LOG	
<b>Nomenclature</b> : configurable via IF_E_ADM_CONF : app.properties	<b>Type</b> : fichier ASCII
<b>Description</b>	
<b>Rôle</b> : Configurer le logger	
<b>Producteur</b>	<b>Consommateur</b>
AMD : Administrateur	KERNEL : Noyau AMDA-NG
<b>Espaces de stockage</b>	
Répertoire d'exécution du noyau AMDA-NG	
<b>Production</b>	
<b>Fréquence de génération</b> : Lors de la configuration du noyau AMDA-NG	
<b>Portée</b> : KERNEL (Noyau AMDA-NG)	<b>Volumétrie</b> : quelques Ko

#### 6.2.1.1 Description détaillée

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\\_hel%2Fcontent\\_engine\\_api%2Fguide%2Flogging\\_procedures.htm](http://pic.dhe.ibm.com/infocenter/p8docs/v4r5m1/index.jsp?topic=%2Fcom.ibm.p8.doc%2Fdeveloper_hel%2Fcontent_engine_api%2Fguide%2Flogging_procedures.htm)

#### 6.2.1.2 Exemple

```
log4j.rootLogger=info, 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=%X{PID} %r %5p [%t] (%r
ms) (%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 (%r ms) %c - %m%n
```

## 6.2.2 IF\_E\_KERNEL\_LOG

Interface IF_E_KERNEL_LOG	
<b>Nomenclature</b> : configurable via IF_E_ADM_CONF : app.properties [exemple.log]	<b>Type</b> : fichier ASCII
Description	
<b>Rôle</b> : Journaliser les messages émis par le noyau AMDA-NG	
Producteur	Consommateur
<i>KERNEL : Noyau AMDA-NG</i>	-
Espaces de stockage	
Configurable via IF_E_ADM_CONF : app.properties [Répertoire d'exécution du noyau AMDA-NG]	
Production	
<b>Fréquence de génération</b> : Fichier journalier	
<b>Portée</b> : <i>KERNEL (Noyau AMDA-NG)</i>	<b>Volumétrie</b> : variable

## 6.3 Données externes

### 6.3.1 IF\_E\_DDSEV\_GET-PARAM

Interface IF_E_DDSEV_GET-PARAM	
<b>Nomenclature</b> : sans objet	<b>Type</b> : TCP/IP
Description	
<b>Rôle</b> : Récupérer des données sur DD server	
Producteur	Consommateur
<i>DDSEV : DD Server</i>	<i>KERNEL : Noyau AMDA-NG</i>
Espaces de stockage	
Aucun	
Production	
<b>Fréquence de génération</b> : Flux sur demande	
<b>Portée</b> : <i>KERNEL (Noyau AMDA-NG)</i>	<b>Volumétrie</b> : variable

### 6.3.1.1 Description détaillée

La récupération des données sur DD server utilise la 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 précisée à l'adresse suivante : <http://manunja.cesr.fr/~budnik/AMDANEW/DDHTML/DOC/>

### 6.3.2 IF\_E\_WEBINT\_PARAM

Interface IF_E_WEBINT_PARAM	
<b>Nomenclature</b> : variable, <nom-param>.xml	<b>Type</b> : XML
<b>Description</b>	
<b>Rôle</b> : Décrire un paramètre	
<b>Producteur</b>	<b>Consommateur</b>
<i>WEBINT : Interface web</i>	<i>KERNEL : Noyau AMDA-NG</i>
<b>Espaces de stockage</b>	
Répertoire référencé par la propriété <code>app.param.path</code> de l'interface <b>IF_E_ADM_CONF</b> : <b>app.properties</b>	
<b>Production</b>	
<b>Fréquence de génération</b> : Sur demande utilisateur	
<b>Portée</b> : <i>KERNEL (Noyau AMDA-NG)</i>	<b>Volumétrie</b> : Quelques Ko

#### 6.3.2.1 Schémas XSD

Les fichiers de description des paramètres respectent leur définition identifiée dans les schémas XSD présentés ci-après.

L'emplacement de ces schémas est configurable via la propriété `app.parameter.xsd` de l'interface **IF\_E\_ADM\_CONF** : **app.properties**.

##### 6.3.2.1.1 all.xsd

Ce schéma est le point d'entrée pour la validation des fichiers XML de description des paramètres.

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



```

<xs:include schemaLocation="getddbbase.xsd" />

<xs:include schemaLocation="getamdaparam.xsd" />

</xs:schema>

```

### 6.3.2.1.2 parameter.xsd

Cette description ne contient que le squelette d'un paramètre. Les éléments de configuration (exemple : « get ») sont chargés et validés à partir d'autres schémas XSD dédiés lors du traitement de l'élément. Actuellement, deux types de paramètres (substitution de Getter) sont gérés :

- ✓ les paramètres AMDA : **getamdaparam.xsd**
- ✓ les paramètres de base : **getddbbase.xsd**

Leurs descriptions respectives sont présentées par la suite.

```

<?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="gap_threshold" 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: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>

```

### 6.3.2.1.3 [getddbbase.xsd](#)

Ce schema décrit un paramètre de base: l'élément param:get contient un element vi.

```

<?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>

```

### 6.3.2.1.4 [getamdaparam.xsd](#)

Ce schema décrit un paramètre AMDA : l'élément param:get contient un element amdaParam.

```

<?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>

```

## 6.3.2.2 Exemples

### 6.3.2.2.1 [Description de paramètre AMDA](#)

```

<?xml version="1.0"?>

```

```

<param xml:id='imf_cond'>
  <get>
    <amdaParam name='imf' />
  </get>
  <process>greater_than($imf(0),0)</process>
  <output/>
</param>

```

### 6.3.2.2.2 Description de paramètre de base

```

<?xml version="1.0" encoding="UTF-8"?>
<param xml:id="dst">
  <fill_value>-1.0e+31</fill_value>
  <get>
    <vi name="ground:based:dst">
      <baseParam name="DST"></baseParam>
    </vi>
  </get>
  <process/>
  <output/>
</param>

```

## 6.3.3 IF\_E\_WEBINT\_REQUEST

Interface IF_E_WEBINT_REQUEST	
<b>Nomenclature</b> : variable	<b>Type</b> : XML
<b>Description</b>	
Rôle : Décrire une requête sur un ou plusieurs paramètres	
<b>Producteur</b>	<b>Consommateur</b>
<i>WEBINT : Interface Web</i>	<i>KERNEL : Noyau AMDA-NG</i>
<b>Espaces de stockage</b>	
Variable, le chemin de la requête est passé en paramètre de l'exécutable du noyau AMDA-NG.	
<b>Production</b>	
Fréquence de génération : Sur demande utilisateur	
<b>Portée</b> : KERNEL (Noyau AMDA-NG)	<b>Volumétrie</b> : quelques Ko

### 6.3.3.1 Schémas XSD

Les fichiers de requête respectent leur définition identifiée dans les schémas XSD présentés ci-après. L'emplacement de ces schémas est configurable via la propriété `app.request.xsd` de l'interface **IF\_E\_ADM\_CONF** : `app.properties`.

#### 6.3.3.1.1 all.xsd

Ce schéma est le point d'entrée pour la validation des fichiers XML de requête.

```

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

```

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="interval.xsd" />
  <xs:include schemaLocation="download.xsd" />
  <xs:include schemaLocation="intervalTrue.xsd" />
  <xs:include schemaLocation="dataMining.xsd" />
  <xs:include schemaLocation="plot.xsd" />
  <xs:include schemaLocation="asciiPlot.xsd" /> <!-- just for test -->
  <xs:include schemaLocation="xyPlot.xsd" />
  <xs:include schemaLocation="timePlot.xsd" />
  <xs:include schemaLocation="tickPlot.xsd" />
  <xs:include schemaLocation="postProcessing.xsd" />
  <xs:include schemaLocation="request.xsd" />
</xs:schema>

```

### 6.3.3.1.2 request.xsd

Cette description ne contient que le squelette d'une requête. Les éléments de configuration tels que `TimeElement` et `OutputElement` sont décrits dans des XSD dédiés à leurs interprétations réelles.

Il existe actuellement les implémentations suivantes :

- ✓ En substitution de l'élément `TimeElement`:
  - `interval`, décrit dans **interval.xsd**
- ✓ En substitution de l'élément `OutputElement` :
  - `download`, décrit dans **download.xsd**
  - `intervalTrue`, décrit dans **intervalTrue.xsd**
  - `dataMining`, décrit dans **dataMining.xsd**

```

<?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:schema>

```

### 6.3.3.1.3 [interval.xsd](#)

Ce schéma décrit les implémentations d'un interval de temps :

- ✓ par valeurs bornées (date de départ et délai),
- ✓ par référence à un fichier TimeTable

```

<?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:element name="timetable" substitutionGroup="TimeElement" >
    <xs:complexType>
      <xs:attribute name="id" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

### 6.3.3.1.4 [download.xsd](#)

Ce schéma décrit un output de type download.

```

<?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="PostProcessingElement" abstract="true" />

  <xs:element name="download" substitutionGroup="OutputElement" >
    <xs:complexType>
      <xs:sequence>
        <xs:element name="timeFormat" type="xs:string" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="fileFormat" type="xs:string" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="fileName" type="xs:string" minOccurs="0"
          maxOccurs="1"/>
        <xs:element name="param" minOccurs="1" maxOccurs="unbounded">
          <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:element name="timeResolution" minOccurs="0" maxOccurs="1"/>
<xs:element name="outputStructure" minOccurs="0" maxOccurs="1">
    <xs:simpleType >
        <xs:restriction base="xs:string">
            <xs:enumeration value="one-file" />
            <xs:enumeration value="one-file-per-interval" />
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element ref="PostProcessingElement" minOccurs="0" maxOccurs="1" />
</xs:sequence>
<xs:attribute name="precision" type="xs:string" />
</xs:complexType>
</xs:element>
</xs:schema>

```

### 6.3.3.1.5 [intervalTrue.xsd](#)

Ce schéma décrit un output conditionné à la validité d'une expression.

```

<?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="intervalTrue" substitutionGroup="OutputElement" >
        <xs:complexType>
            <xs:sequence>
                <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:complexType>
    </xs:element>
</xs:schema>

```

### 6.3.3.1.6 [dataMining.xsd](#)

Ce schéma décrit un output conditionné à la validité d'une expression.

```

<?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="dataMining" substitutionGroup="OutputElement" >

```



```

<xs:complexType>
  <xs:sequence>
    <xs:element name="timeFormat" minOccurs="1" maxOccurs="1"
      type="TimeFormatType"/>
    <xs:element name="fileFormat" minOccurs="1" maxOccurs="1"
      type="FileFormatType"/>
    <xs:element name="outputStructure" minOccurs="0" maxOccurs="1"
      type="OutputStructureType"/>
    <xs:element name="fileName" minOccurs="0" maxOccurs="1"/>
    <xs:element name="param" minOccurs="0" maxOccurs="1">
      <xs:complexType>
        <xs:attribute name="id" type="xs:string"
          use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element ref="PostProcessingElement" minOccurs="0"
      maxOccurs="1"/>
  </xs:sequence>
</xs:complexType>
</xs:element>

<xs:simpleType name="TimeFormatType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ISO" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="FileFormatType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ASCII" />
    <xs:enumeration value="XML" />
    <xs:enumeration value="VOT" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="OutputStructureType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="one-file" />
    <xs:enumeration value="one-file-per-interval" />
  </xs:restriction>
</xs:simpleType>

</xs:schema>

```

### 6.3.3.1.7 [plot.xsd](#)

Ce schéma décrit un output de type tracé.

```

<?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="PlotElement" abstract="true"/>

  <xs:complexType name="FontType">
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="size" type="xs:integer"/>
  </xs:complexType>

  <xs:attributeGroup name="ColorGroup">
    <xs:attribute name="color" type="xs:string" />
    <xs:attribute name="colorMapIndex" type="xs:integer" />
  </xs:attributeGroup>

  <xs:complexType name="TitleType">
    <xs:simpleContent>

```

```

<xs:extension base="xs:string">
  <xs:attributeGroup ref="titleGroup"/>
</xs:extension>
</xs:simpleContent>
</xs:complexType>

<xs:attributeGroup name="titleGroup">
  <xs:attribute name="position">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="top" />
        <xs:enumeration value="bottom" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="align">
    <xs:simpleType >
      <xs:restriction base="xs:string">
        <xs:enumeration value="center" />
        <xs:enumeration value="left" />
        <xs:enumeration value="right" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attributeGroup ref="labelGroup"/>
</xs:attributeGroup>

<xs:attributeGroup name="axisGroup">
  <xs:attribute name="origin">
    <xs:simpleType>
      <xs:restriction base="xs:float">
        <xs:enumeration value="0.0"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="position">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="top"/>
        <xs:enumeration value="bottom"/>
        <xs:enumeration value="left"/>
        <xs:enumeration value="right"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="thickness" type="xs:integer"/>
  <xs:attributeGroup ref="ColorGroup"/>
  <xs:attribute name="reverse" type="xs:boolean"/>
  <xs:attribute name="visible" type="xs:boolean"/>
</xs:attributeGroup>

<xs:attributeGroup name="labelGroup">
  <xs:attribute name="fontName" type="xs:string"/>
  <xs:attribute name="fontSize" type="xs:integer"/>
  <xs:attribute name="style" type="xs:string"/>
  <xs:attributeGroup ref="ColorGroup"/>
</xs:attributeGroup>

<xs:attributeGroup name="legendGroup">
  <xs:attribute name="text" type="xs:string"/>
  <xs:attributeGroup ref="labelGroup"/>
</xs:attributeGroup>

<!-- Define configuration for parameter -->
<xs:complexType name="ParametersType">
  <xs:annotation>
    <xs:documentation>Parameter identifier</xs:documentation>
  </xs:annotation>
  <xs:sequence>

```





```

<xs:element name="param" minOccurs="1" maxOccurs="unbounded">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="default"
        type="ParameterDefaultPropertiesType" minOccurs="0"
        maxOccurs="1">
      </xs:element>
      <xs:element name="serie" type="ParameterYPropertiesType"
maxOccurs="unbounded" minOccurs="0">
        <xs:annotation>
          <xs:documentation>
            define drawing properties for one
            serie associated to that parameter.
            At least one serie should be
            attached to a parameter, but more
            can be associated when related
            parameter is not a simple type (a
            vector for instance).
          </xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

<!-- Define base axis -->
<xs:complexType name="AxisElement">
  <xs:sequence>
    <xs:element name="range" minOccurs="0" maxOccurs="1">
      <xs:complexType>
        <xs:attribute name="min" type="xs:float"/>
        <xs:attribute name="max" type="xs:float"/>
        <xs:attribute name="extend" type="xs:boolean"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="tick" minOccurs="0" maxOccurs="1">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="majorNumber" type="xs:integer"/>
            <xs:attribute name="minorNumber" type="xs:integer"/>
            <xs:attribute name="majorSpace" type="xs:float"/>
            <xs:attribute name="minorSpace" type="xs:float"/>
            <xs:attribute name="position">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:enumeration value="inwards"/>
                  <xs:enumeration value="outwards"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:attribute>
            <xs:attribute name="lengthFactor" type="xs:float"/>
            <xs:attribute name="minorGrid" type="xs:boolean"/>
            <xs:attribute name="majorGrid" type="xs:boolean"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:element name="legend" minOccurs="0" maxOccurs="1">
      <xs:complexType>
        <xs:attributeGroup ref="legendGroup"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attributeGroup ref="axisGroup"/>
</xs:complexType>

```

```

<!-- Define axis that can only be used in X axis -->

<!-- Time axis base -->
<xs:complexType name="TimeAxis">
  <xs:complexContent>
    <xs:extension base="AxisElement">
      <xs:attribute name="format" type="xs:string"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- Epoch axis base -->
<xs:complexType name="EpochAxis">
  <xs:complexContent>
    <xs:extension base="AxisElement">
      <xs:attribute name="format" type="xs:string"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- Digital axis base -->
<xs:complexType name="DigitalAxis">
  <xs:complexContent>
    <xs:extension base="AxisElement">
      <xs:attribute name="id" use="required"/>
      <xs:attribute name="scale">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="linear"/>
            <xs:enumeration value="logarithmic" />
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:element name="plot" substitutionGroup="OutputElement" >
  <xs:complexType>
    <xs:sequence>
      <xs:element name="page" minOccurs="1" maxOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="title" type="TitleType" minOccurs="0" maxOccurs="1"/>
            <xs:element name="margin" minOccurs="0" maxOccurs="1">
              <xs:complexType>
                <xs:attribute name="x" type="xs:decimal"/>
                <xs:attribute name="y" type="xs:decimal"/>
              </xs:complexType>
            </xs:element>
            <xs:element name="font" type="FontType" minOccurs="0" maxOccurs="1"/>
            <xs:element name="panelDefaults" minOccurs="0" maxOccurs="1">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="font" type="FontType" minOccurs="0"
maxOccurs="1"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
            <xs:attribute name="resolution" type="xs:integer"/> <!-- default
series max resolution -->
            <xs:attribute name="backgroundColor" type="xs:string"/>
            <xs:attribute name="colorMapIndex" type="xs:integer"/>
            <xs:attribute name="charSizeUnits" type="xs:integer"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="panel" minOccurs="1" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>

```



```

        <xs:element name="bounds" minOccurs="0" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="x" type="xs:decimal"/>
            <xs:attribute name="y" type="xs:decimal"/>
            <xs:attribute name="width" type="xs:decimal"/>
            <xs:attribute name="height" type="xs:decimal"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="font" type="FontType" minOccurs="0"
maxOccurs="1"/>
        <xs:element name="title" type="TitleType" minOccurs="0"
maxOccurs="1"/>
        <xs:element ref="PlotElement" minOccurs="1" maxOccurs="1"/>
        <xs:element name="tickPlot" minOccurs="0" maxOccurs="1">
          <xs:complexType >
            <xs:annotation>
              <xs:documentation>
                This enable to add decoration for time plot.
              </xs:documentation>
            </xs:annotation>
            <xs:sequence>
              <xs:element name="params" type="TickPlotParams" minOccurs="0"
maxOccurs="1"/>
              </xs:sequence>
              <xs:attribute name="format" type="xs:string" use="optional">
                <xs:annotation>
                  <xs:documentation>tickmarks format. Uses printf format
specification. (resulting string will be limited to 40 chars, due to pplot limitation)
                  </xs:documentation>
                </xs:annotation>
              </xs:attribute>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
        <xs:attribute name="format" use="required">
          <xs:simpleType >
            <xs:restriction base="xs:string">
              <xs:enumeration value="png" />
              <xs:enumeration value="pdf" />
              <xs:enumeration value="ps" />
              <xs:enumeration value="svg" />
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="dimension" >
          <xs:simpleType >
            <xs:restriction base="xs:string">
              <xs:enumeration value="ISO A4" />
              <xs:enumeration value="US letter" />
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="orientation">
          <xs:simpleType >
            <xs:restriction base="xs:string">
              <xs:enumeration value="landscape" />
              <xs:enumeration value="portrait" />
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="mode">
          <xs:simpleType >
            <xs:restriction base="xs:string">
              <xs:enumeration value="color" />

```

```

        <xs:enumeration value="grayscale" />
    </xs:restriction>
</xs:simpleType>
</xs:attribute>
</xs:complexType>
</xs:element>
<xs:element ref="PostProcessingElement" minOccurs="0" maxOccurs="1"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:complexType name="LineType">
    <xs:annotation>
        <xs:documentation>Line specifications. Define properties for a line attached to a
parameter.
Properties set there will override default value or general properties that could have
been set in tag's parent</xs:documentation>
    </xs:annotation>
    <xs:attribute name="type">
        <xs:annotation>
            <xs:documentation>Line type. Can be one of :
no : no line is drawn
line: a line is drawn between each points
histo: an histogram is drawn for each point

Note : This attribute may be unnecessary since kind of line may be deduce from Plot Type
[TBC]
        </xs:documentation>
    </xs:annotation>
</xs:simpleType>
    <xs:restriction base="xs:string">
        <xs:enumeration value="no"></xs:enumeration>
        <xs:enumeration value="line"></xs:enumeration>
        <xs:enumeration value="histo"></xs:enumeration>
    </xs:restriction>
</xs:simpleType>
</xs:attribute>
    <xs:attribute name="style">
        <xs:annotation>
            <xs:documentation>Line Style. Can be one of the following values :
- plain : plain line style
- dot : line with short dashes and gaps
- long-spaced-dot : line with long dashes and long gaps.
- long-short-dot : line with long dashes and short gaps.</xs:documentation>
        </xs:annotation>
    </xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:enumeration value="plain"></xs:enumeration>
            <xs:enumeration value="dot"></xs:enumeration>
            <xs:enumeration value="long-spaced-dot"></xs:enumeration>
            <xs:enumeration value="long-short-dot"></xs:enumeration>
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
    <xs:attribute name="width" type="xs:int" default="1">
        <xs:annotation>
            <xs:documentation>Line width in mm. Default is 1mm</xs:documentation>
        </xs:annotation></xs:attribute>
    <xs:attributeGroup ref="ColorGroup"/>
</xs:complexType>

<xs:complexType name="SymbolType">
    <xs:annotation>
        <xs:documentation>Symbol displayed for each points. By default, no symbol is drawn
if a line is defined. A 'point' symbol is used othrewise</xs:documentation>
    </xs:annotation>
    <xs:attribute name="type">
        <xs:annotation>
            <xs:documentation>Symbol type.</xs:documentation>

```

```

</xs:annotation>
<xs:simpleType>
  <xs:restriction base="xs:string">
    <xs:enumeration value="no"></xs:enumeration>
    <xs:enumeration value="dot"></xs:enumeration>
    <xs:enumeration value="plus"></xs:enumeration>
    <xs:enumeration value="wildcard"></xs:enumeration>
    <xs:enumeration value="circle"></xs:enumeration>
    <xs:enumeration value="crux"></xs:enumeration>
    <xs:enumeration value="square"></xs:enumeration>
    <xs:enumeration value="triangle"></xs:enumeration>
    <xs:enumeration value="crux-in-circle"></xs:enumeration>
    <xs:enumeration value="dot-in-circle"></xs:enumeration>
    <xs:enumeration value="square-2"></xs:enumeration>
    <xs:enumeration value="diamond"></xs:enumeration>
    <xs:enumeration value="star"></xs:enumeration>
    <xs:enumeration value="christian-crux"></xs:enumeration>
    <xs:enumeration value="david-star"></xs:enumeration>
    <xs:enumeration value="full-square"></xs:enumeration>
    <xs:enumeration value="full-circle"></xs:enumeration>
    <xs:enumeration value="full-star"></xs:enumeration>
  </xs:restriction>
</xs:simpleType>
</xs:attribute>
<xs:attribute name="size" type="xs:int" default="4">
  <xs:annotation>
    <xs:documentation>
      symbol size in mm. Default is 4mm
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
<xs:attributeGroup ref="ColorGroup"/>
</xs:complexType>

<xs:complexType name="ParameterXPropertiesType">
  <xs:attribute name="index" type="xs:int">
    <xs:annotation>
      <xs:documentation>
        Index of serie to put on x axis (in case Parameter is not a
        scalar type).
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="xAxis" type="xs:string" use="required"></xs:attribute>
</xs:complexType>

<xs:complexType name="ParameterYPropertiesType">
  <xs:sequence>
    <xs:element name="line" type="LineType" maxOccurs="1"
      minOccurs="0">
    </xs:element>
    <xs:element name="symbol" type="SymbolType" maxOccurs="1"
      minOccurs="0">
    </xs:element>
  </xs:sequence>
  <xs:attribute name="index" type="xs:int">
    <xs:annotation>
      <xs:documentation>
        Index of serie to plot (in case Parameter is not a
        scalar type). Default to 1 if Parameter is a simple
        type.
      </xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="yAxis" type="xs:string"></xs:attribute>
  <xs:attribute name="xAxis" type="xs:string"></xs:attribute>
  <xs:attribute name="resolution" type="xs:integer"/>
</xs:complexType>

```

```

<xs:complexType name="ParameterDefaultPropertiesType">
  <xs:annotation>
    <xs:documentation>
      Default values for all series. Can be overridden in
      each series element.
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="line" type="LineType" minOccurs="0" maxOccurs="1">
      <xs:annotation>
        <xs:documentation>Default line properties</xs:documentation>
      </xs:annotation></xs:element>
    <xs:element name="symbol" type="SymbolType" minOccurs="0" maxOccurs="1">
      <xs:annotation>
        <xs:documentation>Default symbol properties</xs:documentation>
      </xs:annotation></xs:element>
  </xs:sequence>
  <xs:attribute name="yAxis" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation>Default yAxis value</xs:documentation>
    </xs:annotation></xs:attribute>
  <xs:attribute name="xAxis" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation>Default xAxis value</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attributeGroup ref="ColorGroup"/>
</xs:complexType>

<xs:complexType name="TickmarksType">
  <xs:attribute name="format" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation>tickmarks format. Uses printf format specification.
      (resulting string will be limited to 40 chars, due to pplot
      limitation)</xs:documentation>
    </xs:annotation></xs:attribute>
  </xs:complexType>

<xs:complexType name="TickmarksType">
  <xs:attribute name="format" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation>tickmarks format. Uses printf format specification.
      (resulting string will be limited to 40 chars, due to pplot
      limitation)</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
</xs:schema>

```

### 6.3.3.1.8 [timePlot.xsd](#)

Ce schéma décrit un output de type tracé  $f(t)$  avec ou sans tickmark, il s'appuie sur le schéma `plot.xsd`.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="plot.xsd" />
  <xs:element name="timePlot" substitutionGroup="PlotElement">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="params" type="ParametersType" minOccurs="0" maxOccurs="1"/>
        <xs:element name="axes" minOccurs="1" maxOccurs="1">
          <xs:complexType>
            <xs:sequence>

```

```

<xs:element name="xAxis">
  <xs:complexType>
    <xs:sequence minOccurs="1" maxOccurs="1">
      <xs:element name="timeAxis" type="TimeAxis"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="yAxis">
  <xs:complexType>
    <xs:sequence minOccurs="1" maxOccurs="unbounded">
      <xs:element name="digitalAxis" type="DigitalAxis"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

### 6.3.3.1.9 xyPlot.xsd

Ce schéma décrit un output de type tracé  $f(x)$ , il s'appuie sur le schéma `plot.xsd`.

```

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

  <xs:element name="xyPlot" substitutionGroup="PlotElement">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="params" minOccurs="0" maxOccurs="1">
          <xs:complexType>
            <xs:annotation>
              <xs:documentation>Parameter serie definition</xs:documentation>
            </xs:annotation>
            <xs:sequence>
              <xs:element name="param" minOccurs="1" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="default"
                      type="ParameterDefaultPropertiesType" minOccurs="0"
                      maxOccurs="1">
                    </xs:element>
                    <xs:element name="xserie"
                      type="ParameterXPropertiesType"
                      maxOccurs="unbounded" minOccurs="0">
                      <xs:annotation>
                        <xs:documentation>
                          define xaxis values
                        </xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element name="yserie"
                      type="ParameterYPropertiesType"
                      maxOccurs="unbounded" minOccurs="0">
                      <xs:annotation>
                        <xs:documentation>
                          define drawing properties for one
                          serie associated to that parameter.
                          At least one serie should be
                          attached to a parameter, but more
                          can be associated when related
                          parameter is not a simple type (a
                          vector for instance).
                        </xs:documentation>
                      </xs:annotation>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

        </xs:annotation>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string"/>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="axes" minOccurs="1" maxOccurs="1">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="xAxis">
        <xs:complexType>
          <xs:sequence minOccurs="1" maxOccurs="1">
            <xs:choice>
              <xs:element name="digitalAxis" type="DigitalAxis"/>
              <xs:element name="epochAxis" type="EpochAxis"/>
            </xs:choice>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="yAxis">
        <xs:complexType>
          <xs:sequence minOccurs="1" maxOccurs="unbounded">
            <xs:element name="digitalAxis" type="DigitalAxis"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  </xs:element>
</xs:sequence>
<xs:attribute name="isotropic" type="xs:boolean"/>
</xs:complexType>
</xs:element>
</xs:element>
</xs:schema>

```

### 6.3.3.1.10 [postProcessing.xsd](#)

Ce schéma décrit les différents types de post-traitements fournis par le noyau AMDA.

```

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

  <xs:element name="postProcess" substitutionGroup="PostProcessingElement" >
    <xs:complexType>
      <xs:sequence>
        <xs:element name="zip" minOccurs="0" maxOccurs="1"/>
        <xs:element name="tar" minOccurs="0" maxOccurs="1"/>
        <xs:element name="gzip" minOccurs="0" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

</xs:schema>

```

## 6.3.4 IF\_E\_DEV\_PLUGIN-LIB

Interface IF_E_DEV_PLUGIN-LIB	
Nomenclature : variable (*.so)	Type : librairie



		dynamique
<b>Description</b>		
<b>Rôle</b> : Ajouter dynamiquement une nouvelle fonctionnalité/une nouvelle fonction mathématique		
<b>Producteur</b>		<b>Consommateur</b>
<i>DEV : développeur</i>		<i>KERNEL : Noyau AMDA-NG</i>
<b>Espaces de stockage</b>		
Répertoire référencé par la propriété <code>app.plugin</code> de l'interface <b>IF_E_ADM_CONF</b> : <b>app.properties</b>		
<b>Production</b>		
<b>Fréquence de génération</b> : sans objet		
<b>Portée</b> : <i>KERNEL (Noyau AMDA-NG)</i>		<b>Volumétrie</b> : Variable

### 6.3.5 IF\_E\_DEV\_PLUGIN-FONC

<b>Interface IF_E_DEV_PLUGIN-FONC</b>	
<b>Nomenclature</b> : variable (*.h)	<b>Type</b> : Entête C++
<b>Description</b>	
<b>Rôle</b> : Ajouter dynamiquement une nouvelle fonction mathématique	
<b>Producteur</b>	
<b>Consommateur</b>	
<i>DEV : développeur</i>	<i>KERNEL : Noyau AMDA-NG</i>
<b>Espaces de stockage</b>	
Sous répertoire du répertoire référencé par la propriété <code>app.plugin</code> de l'interface <b>IF_E_ADM_CONF</b> : <b>app.properties</b>	
<b>Production</b>	
<b>Fréquence de génération</b> : sans objet	
<b>Portée</b> : <i>KERNEL (Noyau AMDA-NG)</i>	<b>Volumétrie</b> : Variable

#### 6.3.5.1 Description détaillée

Un plugin d'ajout de fonctions mathématiques est déposé dans un sous-répertoire dédié `<nom_fonction>` et se décline en un fichier `<nom_fonction>.h` + éventuellement une librairie dynamique relative `<nom_fonction>.so`.

## 7. DESCRIPTION DES INTERFACES INTERNES

### 7.1.1 IF\_I\_KERNEL\_CC-TMP

Interface IF\_I\_KERNEL\_CC-TMP

<b>Nomenclature</b> : variable (*.cc)		<b>Type</b> : Fichier source C++
<b>Description</b>		
<b>Rôle</b> : Formaliser une fonction de calcul dont le code est une chaîne c++ générée à partir d'une expression écrite dans une balise <code>process</code> d'un fichier XML de description de paramètre		
<b>Producteur</b>		<b>Consommateur</b>
<i>KERNEL : Noyau AMDA-NG</i>		<i>KERNEL : Noyau AMDA-NG</i>
<b>Espaces de stockage</b>		
Sous-répertoire unique du répertoire référencé par la propriété <code>app.process.src</code> de l'interface <b>IF_E_ADM_CONF</b> : <code>app.properties</code>		
<b>Production</b>		
<b>Fréquence de génération</b> : Sur requête utilisateur		
<b>Portée</b> : <i>KERNEL (Noyau AMDA-NG)</i>		<b>Volumétrie</b> : variable

### 7.1.1.1 Description détaillée

Les fichiers source « .cc » sont générés dans un sous-répertoire nommé `<NomParam><unique-id>`. Celui-ci est supprimé automatiquement après la génération de la librairie associée, cf. interface **IF\_I\_KERNEL\_PLUGIN-LIB**.

### 7.1.2 IF\_I\_KERNEL\_PLUGIN-LIB

<b>Interface IF_I_KERNEL_PLUGIN-LIB</b>		
<b>Nomenclature</b> : variable (*.so)		<b>Type</b> : Librairie dynamique
<b>Description</b>		
<b>Rôle</b> : Réaliser un calcul mathématique pour un paramètre donné		
<b>Producteur</b>		<b>Consommateur</b>
<i>KERNEL : Noyau AMDA-NG</i>		<i>KERNEL : Noyau AMDA-NG</i>
<b>Espaces de stockage</b>		
Répertoire référencé par la propriété <code>app.process.lib</code> de l'interface <b>IF_E_ADM_CONF</b> : <code>app.properties</code>		
<b>Production</b>		
<b>Fréquence de génération</b> : Sur requête utilisateur, s'il n'existe pas déjà ou si la définition du paramètre a changé depuis la dernière génération		
<b>Portée</b> : <i>KERNEL (Noyau AMDA-NG)</i>		<b>Volumétrie</b> : variable

### 7.1.2.1 Description détaillée

Une telle librairie est associée à un paramètre donné et a pour nom `<nomParam>.so` ; elle est conservée et peut être réutilisée à chaque exécution.

Dans le cas de la suppression du fichier paramètre XML, la librairie associée « .so » doit être supprimée.

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

## Versions successives



Version	Date	Émetteur	Vérificateur	Approbateur	Motif
1.4	23/01/2014	N.Boursier	S.Frayssines	R.Patrier	Modification des schémas XSD des requêtes pour TickPlot.
1.3	10/01/2014	N.Boursier	S.Frayssines	R.Patrier	Mise à jour des schémas XSD des requêtes pour DataMining et Download. Suppression de l'interface NFS. Ajout des schémas XSD de la requête de Plot.
1.2	08/10/2013	N.Boursier	S.Frayssines	R.Patrier	Mise à jour des schémas XSD des requêtes pour DataMining et PostProcessing
1.1	16/09/2013	N.Boursier	S.Frayssines	R.Patrier	Prise en compte des FEPS : 1 et 7
1.0	26/07/2013	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.*