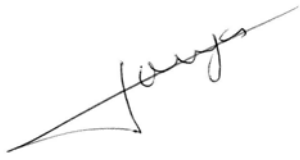
IASI

THALES INFORMATION SYSTEMS

IA-MI-2100-9554-THA
Issue : 02 Date : 23/08/2002
Revision : 11 Date : 06/04/2011
Ref. : - MT : X Distribution Code : E

MANUEL D'INSTALLATION
OPS IASI LEVEL1 SOFTWARE INSTALLATION MANUAL

Written by : BRUNEL Samuel	THALES SERVICES	Date : 11/04/2011	
Approved by : TOUJAS Chantal	THALES SERVICES	Date : 11/04/2011	
For application : BOTELLA Christine	THALES SERVICES	Date : 11/04/2011	

18, avenue Edouard Belin 31401 Toulouse CEDEX 9 Tél. 05 61 27 31 31

INDEX SHEET

CONFIDENTIALITY : NC	**KEYWORDS : software installation, IASI, algorithm, Image Processing**

TITLE : MANUEL D'INSTALLATION
OPS IASI LEVEL1 SOFTWARE INSTALLATION MANUAL

AUTHOR(S) : BRUNEL Samuel THALES SERVICES

SUMMARY : This document presents the installation environment and procedure of the OPS facility.

RELATED DOCUMENTS : Stand alone document.	**LOCALIZATION :**

VOLUME : 1	**TOTAL NUMBER OF PAGES : 36** **INCLUDING PRELIMINARY PAGES : 6** **NUMBER OF SUPPL. PAGES : 0**	**COMPOSITE DOCUMENT : N**	**LANGUAGE : EN**

CONFIGURATION MANAGEMENT : NG	**CM RESP. :**

REASONS FOR EVOLUTION : Mise à jour pour IA-FT-2765 (DM-2100-OPS)

CONTRACT : 01/8937

HOST SYSTEM :
Microsoft Word 11.0 (11.0.6568)
G:\Thales\prive\Projets\pleiades\MODELES CNES\MODELES GDOC\GDOC 4.0.5\ModeleGDOCIndus_en.dot
Version GDOC : v4.0.5
Base projet : G:\Thales\prive\Projets\iasi-ops\Modèles CNES\IASI-thales01

INTERNAL DISTRIBUTION

Name	Entity	Internal Postal Box	Observations
BLUMSTEIN Denis	DCT/PO/EV	2504	
CHALON Gilles	DCT/PO/EV	2504	
PONCE Ghislaine	DCT/PO/EV	2504	
SEGALEN Barbara	DCT/PO/EV	2504	
DUPLAA Michel	DTS/MID/VM/D	1502	
MARQUIER Henry	DCT/PO/EV	1321	
MORENO Richard	DCT/PS/TIS	1321	
BAILLY Isabelle	DCT/PS/TIS	1321	
RAYSSIGUIER Michel	DTS/OT/QTIS/VP	811	
RICHARD Pascal	DSI/EP/SL	3517	
GOMEZ Marie-Hélène	DCT/PS/TIS	1321	
MATHIEU Nathalie	EUROGICIEL	1415	
FJORTOFT Roger	DCT/SI/EI	1219	
VANNET Carole	EUROGICIEL		
LONJOU Vincent	DCT/ME/EI		

EXTERNAL DISTRIBUTION

Name	Entity	Observations
AYER Patrick	THALES SERVICES	
BOBIN Serge	THALES SERVICES	
BRANET Pascal	THALES SERVICES	
PASCAL Jean-Luc	THALES SERVICES	
CABANE Philippe	THALES SERVICES	
MASSART Benjamin	THALES SERVICES	
BRUNEL Samuel	THALES SERVICES	
TOUJAS Chantal	THALES SERVICES	
BOTELLA Christine	THALES SERVICES	
RANDRIA Prosper	THALES SERVICES	

CHANGES

Issue	Rev.	Date	Reference, Author(s), Reasons for evolution
02	11	06/04/2011	- BRUNEL Samuel THALES SERVICES Mise à jour pour IA-FT-2765 (DM-2100-OPS)
02	10	20/11/2008	- BRUNEL Samuel THALES SERVICES DM2458 : Migration vers AIX 6.1 et compilateur XLC v 8
02	09	13/09/2007	- PASCAL Jean-Luc THALES SERVICES BRANET Pascal THALES SERVICES MASSART Benjamin THALES SERVICES BRUNEL Samuel THALES SERVICES Action 369 - Indiquer les versions de lex et yacc à utiliser
02	08	16/03/2006	- MASSART Benjamin THALES IS Environment variables settings added.
02	07	18/03/2004	- PASCAL Jean-Luc THALES IS BRANET Pascal THALES IS modifying chapter 4.4.1 [LIBPATH initialisation] adding METOP_ENV variable for METOP CFI (CR232)
02	06	29/09/2003	- BRANET Pascal THALES IS modifying chapter 5 [IA-2100-FA-64-CN]
02	05	03/06/2003	- BRANET Pascal THALES IS specifications for installing the validation environment
02	04	19/05/2003	- BRANET Pascal THALES IS OPS-V1 delivery
02	03	19/02/2003	- BRANET Pascal THALES IS OPS-PKPV delivery
02	02	12/12/2002	- BRANET Pascal THALES IS Modifying chapter 4.2 [IA-2100-FA-10-CN] Adding chapter 4.4.2.2 [IA-2100-FA-7-CN]
02	01	29/11/2002	- BRANET Pascal THALES IS

			OPS-V0 delivery
02	00	23/08/2002	- BRANET Pascal THALES IS GDOC format
01	00	14/06/2002	- BRANET Pascal THALES IS Creation of the document

TABLE OF CONTENTS

GLOSSARY AND LIST OF TBC AND TBD ITEMS

API	Applicative Program Interface
CFI	Customer Furnished Item
CGS	Core Ground Segment : segment-sol développé par ALCATEL sous contrat d'EUMETSAT, et dans lequel l'OPS ira s'insérer
GDD	Gestionnaire de Données et de Diffusion : sous-système du CNES
IASI	Infrared Atmospheric Sounding Interferometer : interferomètre de sondage atmosphérique dans l'infrarouge.
JDBS	JdB server
MCS	Monitoring and Control Segment
MLA	MCS Local Agent
MP	Main Process
MSGS	Message Server
OPS	Logiciel Opérationnel (Operational Software) : correspond au IASI level 1 PPS dans les glossaires d'EUMETSAT. PPS=Product Processing Software
OS	Operating System
PGF	Product Generation Facility
SIF	Simulteurs d'Interfaces du F-PAC ENVISAT (sous système du CNES)
TES	Time Event Server
WOM	Work Order Manager

List of TBC items :

List of TBD items :

1.OVERVIEW

1.1.APPLICABLE AND REFERENCE DOCUMENTS

The contractual applicable and reference documents are listed in the « Liste Unique » document IA-LD-2100-9550-THA.

1.2.OBJECTIVES

The installation procedure is divided of two parts:

- build the executable software from the source code,
- install the software on the target computer.

The current document is the Software Installation Manual [DA114] which depicts the procedures to install the OPS software on the target computer.

The intended readers are the CGS operators who are in charge to install the OPS software. This operator has to be familiar with the Unix operating system (AIX/Linux) to be able to proceed to the installation without block.

1.3.USING DOCUMENT

Section 1 presents an overview of the document.

Section 2 presents an OPS technical overview.

Section 3 describes the OPS installation procedures.

2.TECHNICAL OVERVIEW

2.1.OPS FUNCTIONS

The main OPS function is to generate IASI L1 products from IASI L0 data under the PGF control.

The Production processus is split into 3 steps:

- processing initialisation : configuration parameters loading,
- data computing,
- products generation.

The functions outcome of monitoring and control requirements are :

- OPS sub system monitoring and control,
- work order retrieving and processing,
- log and trace event management,
- anomalies management.

2.2.ARCHITECTURE

The OPS software is compound of the following 5 permanent UNIX processus:

- the MP processus (Main Process) is in charge of monitoring and control the OPS subsystem : start/stop the facility, collect and execute the PGF commands, collect and generate the HKTM status.
- the TES processus (Time Event Server) is in charge to inform a subscriber process when a scheduled event timer occurs (periodic timer or punctual date).
- the MSGS processus (Message Server) is in charge to collect and dispatch the inter-processus messages. This processus provides a centralized and generic mechanism to exchange applicative messages between software processus.
- the JDBS processus (JDB Server) is in charge to collect log events and log traces messages generated by the OPS processus and send them to the MCS.
- the WOM processus (Work Order Manager) is in charge to manage the data processing commands (STEP/SUSPEND/RESUME/BREAK) provided by the PGF and to control the data processing executed by the SD processus.
- the SD processus (Data Server) is in charge to process IASI L0 input data in order to generate IASI L1 data according to the processing description provided by the WOM.

The following figure shows the OPS software architecture.

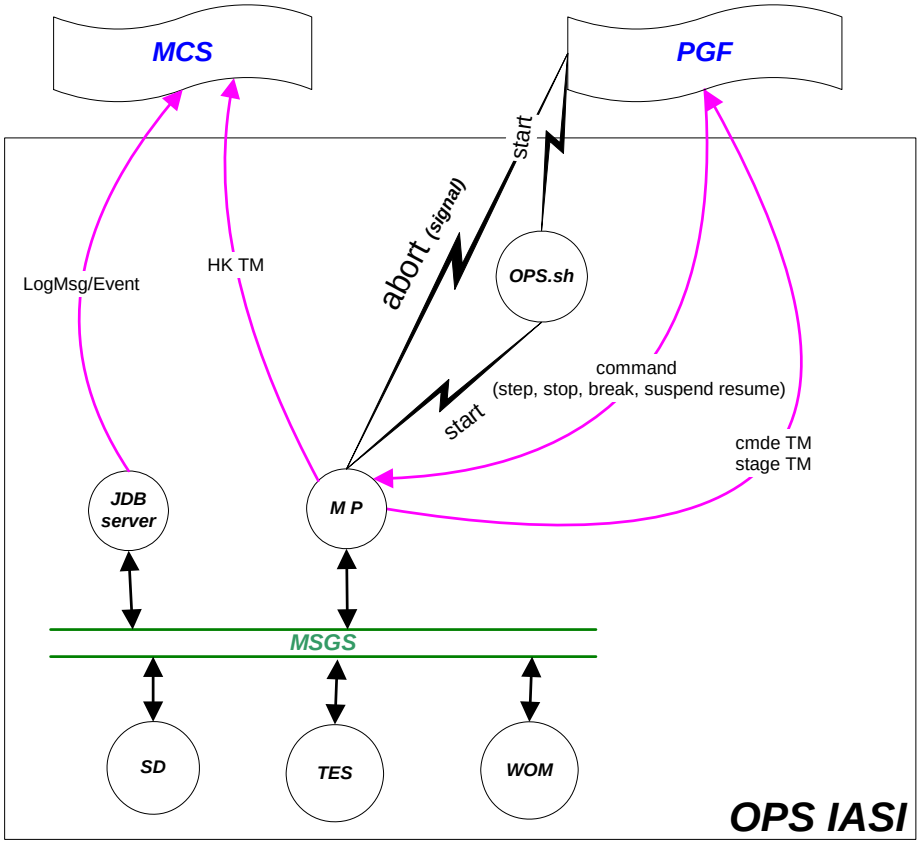


Figure 1 : OPS software architecture

3.GENERATION

The OPS software generation is described in the Software Transfer Document [DA120].

4.INSTALLATION

4.1.HARDWARE AND SOFTWARE CONFIGURATION REQUISITE

4.1.1.AIX

The OPS-IASI nominal or standalone software versions are designed to be installed onto multi-processor IBM power PC workstation or node using AIX 6.1 operating System.

In order to achieve the installation stage, the minimum available space must be 140 Mo.

In order to reach the performance requirements, the nominal hardware configuration must provide as minimum the following characteristics:

CPU type	Power4
Mhz	1GHz (2x500MHz)
L1 Instruction Cache	128KB
L1 data cache	64KB
L2	~1,5 MB
L3	32MB
CPU Nb per node	4
Memory	2Gb
System Disks	2x36Gb
Internal Switch Throughput	2x150Mb/s

The minimum software configuration available onto the installation computer is :

- the operating system AIX 6.1. No specific parameter setting of the operating system AIX is required.
- the IDL library version 5.5 for AIX 5.1 OS (No change carried out on AIX 6.1).

The PGF is in charge :

- to declare the OPS users,
- to configure the hardware elements such as hard disk (filesystems, space),
- to install the CFI such as ESSL,

- to create the working directory tree, where the OPS installation is executed.

Important :

You must use the versions of lex and yacc provided with IBM AIX operating system.

4.1.2.Linux

The OPS-IASI nominal or standalone software versions are designed to be installed onto multi-processor Intel ® Core ™ i7 using Red Hat Enterprise Linux Server operating system.

In order to achieve the installation stage, the minimum available space must be 140 Mo.

In order to reach the performance requirements, the nominal hardware configuration must provide as minimum the following characteristics:

CPU type	i7- 870
Mhz	2.93GHz
Smart Cache	8 Mb
CPU Nb per node	4 (8 Threads)
Memory	4Gb
System Disks	> 10 Gb

The minimum software configuration available onto the installation computer is:

- the operating system Red Hat Enterprise Linux Server 5.5 . No specific parameter setting of the operating system is required.

The PGF is in charge:

- to declare the OPS users,
- to configure the hardware elements such as hard disk (filesystems, space),
- to install the CFI such as FFTW3,
- to create the working directory tree, where the OPS installation is executed.

Important :

You must use the versions of lex and yacc provided with Linux operating system.

4.2.INSTALL OSP RUN-TIME SOFTWARE (BINARY INSTALLATION KIT)

To install the OSP run-time from the tape:

Connect as **iasi_1mg** user

change in a temporary directory : **cd /tmp** (for example)

tar xvf *tape_device* to extract the installation kit

The following 3 files are extracted from the installation tar file :

- **OPS_Install_<version>.tar.Z,**
- **install_OPS.sh**,
- **CksumOPS_<version>**.

The **OPS_Install_<version>.tar.Z** file provides executable and configuration parameters for OPS software execution.

The **CksumOPS_<version>** file provides the **OPS_Install_<version>.tar.Z** file checksum. This checksum is used at the beginning of the OPS software installation procedure in order to verify the delivery integrity.

The **install_OPS.sh** file is the OPS software installation script.

To install a new version of the OPS software, use the **install_OPS.sh** script :

install_OPS.sh

The installation procedure is divided in 4 steps:

- cheksum verification :
  (waiting for message : Testing cksum OK).
- root installation directory identification :
  (message : Enter absolute directory, where you want to install OPS binary).
  (enter :<working_root_dir>/OPS_versions).[*the directory is available*]
- installation directory path confirmation :

(message: Do you want to install OPS binary files to <DIRECTORY>, return to install or Ctrl+C to abort).

- OPS software extraction.

The installation data are put in the root installation directory named **OPS_<version>.**

On successful generation, following message is logged :

```
Install success
```

Delete the unnecessary files :

change in the directory where the extracted files are stored

rm OPS_Install_<version>.tar.Z install_OPS.sh CksumOPS_<version>

If needed, create the sub-directory named “data” under :<working_root_dir>/tmp directory :

cd <working_root_dir>/tmp

mkdir data

4.3.RUN TIME DIRECTORY TREE

The result of the installation process shall be the following directory tree :

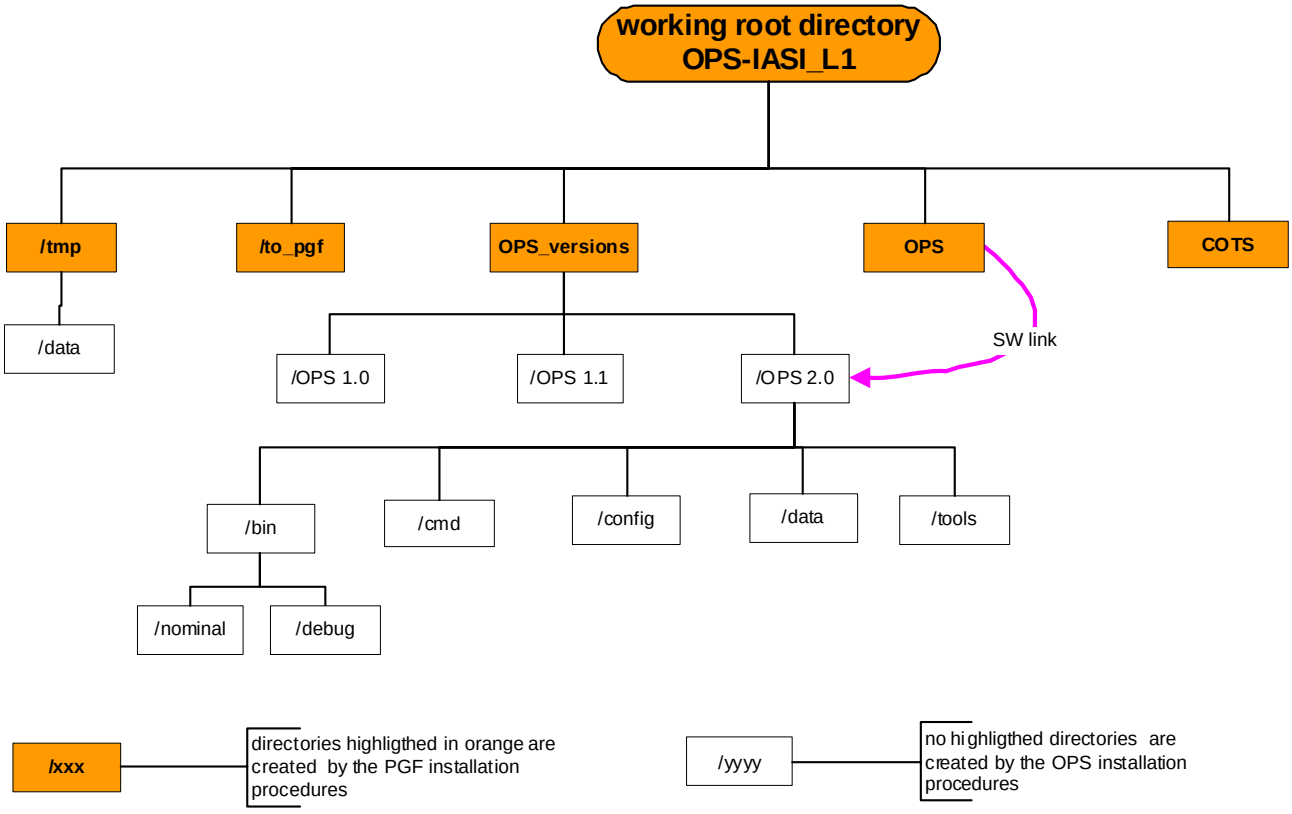


Figure 2 : OPS run time directory tree

Directories	Comments
/tmp	This directory is created by the PGF. It is used by the OPS to store temporary data files. This directory is purged by the OPS at the OPS start up.
/to_pgf	This directory is created by the PGF. The OPS puts the generated products and data in this directory. The PGF is in charge to collect them and purge the directory.
/OPS	This directory is created by the PGF. The PGF links this directory with the installation directory of the current OPS version.
/COTS	This directory is created by the PGF. It is managed by the PGF is used to install COTS.
/OPS_versions	This directory is created by the PGF. This directory is the root directory where all available OPS versions are installed.
/OPS_version/OPS_Vx-x	This directory is created by the OPS installation script. It is the root installation directory of the version <version> of the OPS.
/OPS_version/OPS_Vx-x/bin	This directory is created by the OPS installation script. The OPS software executable are installed in this directory.
/OPS_version/OPS_Vx-x/cmd	This directory is created by the OPS installation script. The OPS start up shell scripts are installed in this directory.
/OPS_version/OPS_Vx-x/config	This directory is created by the OPS installation script. The OPS configuration files (see §4.4) are installed in this directory.
/OPS_version/OPS_Vx-x/tools	This directory is created by the OPS installation script. The software tools used during the acceptance tests are stored in this directory.

Directories	Comments
/OPS_version/OPS_Vx-x/data	This directory is created by the OPS installation script. The input data files and reference data output files for the selftests are stored in this directory.

4.4.CONFIGURATION PROCEDURES

OPS configuration parameters are stored in ASCII files available in /OPS_version/OPS_Vx-x/config directory.

The OPS configuration parameters are available:

- either in environment variables files : each OPS software process is configured thanks to its own set of environment variables stored in a file named <**processus name**>**.env** , plus a global set of environment variables stored in a file named **OPS.env**.
- or in data values configuration file : a dedicated file is defined for each type of data.

4.4.1.The environment variables sets

The following parameters are defined in OPS.sh and OPS.env files :

- installation directories,
- inter-processus communication parameters,
- start up parameters,
- MLA configuration interface parameters (`LA_API_MODE,…`).

The parameters the most frequently updated are the following ones :

environment variable name	parameter
OPS.sh shell script	
REP_ROOT	Path to the /OPS (see §4.3) directory
REP_CONF_DIR	Path to the directory where are stored the configurations files
LA_API_MODE	values are defined in [DR16]
LA_API_CMD	values are defined in [DR16]
LA_API_TLM	values are defined in [DR16]
LA_API_LOG	values are defined in [DR16]
OPS.env file	
CHN_FACILITY	Facility Identification (see [DA17])
CHN_OPS_INSTAL_TYPE	OPS Installation Type (allowed values are : “CGS” or “STANDALONE”)

The main environment variables stored in the <process>.env files (MP.env, TES.env, WOM.env, SERVER.env, SD.env) are :

environment variable name	parameter
MP.env file	
NUM_HKTM_PERIODICITY	Periodicity to deliver the HKTM status to the MCS.
NUM_CMD_PERIODICITY	Periodicity to retrieve the PGF commands
SPACECRAFT_ID	Spacecraft identification : M01, M02, M03 , N16 or N17 for respectively METOP 01, METOP 02, METOP 03, NOAA 16 and NOAA 17 satellites. The default value is M01 if the environment variable is not defined.
SD.env file	
METOP_ENV	Parameters used by METOP CFI (geolocalisation) (allowed values are : “TEST” or “NORMAL”)
NUM_STATUS_RES_SD	Periodicity to deliver the HKTM status to the MP
DATE_MARGE_LIGNE_AVHRR	precision of the AVHRR integration period in milliseconds. The default value is 1 if the environment variable is not defined.
CONTEXT_SOURCE	author of the context : IAST, CGS1. The default value is CGS1 if the environment variable is not defined.
TES.env file	
NUM_STATUS_RES_SD	Timer accuracy
WOM.env file	
NUM_DUREE_GRANULE	The granule mean time duration

4.4.1.1.OPS.env

```
#!/bin/ksh
# $Header: /home/ops_gc/cvs-repository/OPS/MP/src/OPS.env,v 1.9 2003/02/13 10:41:34
iasi Exp $
# ------------------------------------------------------------------*/
# Composant : OPS
# CLASSE : Environnement
#
# Commentaire :
#    Parametrage globale pour le systeme OPS
#
# ------------------------------------------------------------------*/
# HISTORIQUE
# VERSION : 21/03/2000 : A.MONTMORY
# Creation du fichier
#
# Origine     Date        Auteur   Commentaire
# ------------------------------------------
# FA-ID : <N°> : ../../.... : ... : ...
#
# SIF-00-001 12/07/00 A.MONTMORY
# Mise en coherence configuration. Rajout de la definition des adresses
# d'echange avec le DFE,DFR,INVE,INVR.
#
# FIN-HISTORIQUE
# ------------------------------------------------------------------*/

# Adresse de connexion pour trouver le MSGS server
#-------------------------------------------------

export ADR_MSG_SERVER_HOST=`hostname -s`
export ADR_MSG_SERVER_PORT="4000"
BIN_MSG_SERVER="MSGS__Serveur"
nbRunningMSGS=`ps -ef | grep $BIN_MSG_SERVER |grep -v grep | wc -l`
let ADR_MSG_SERVER_PORT=ADR_MSG_SERVER_PORT+nbRunningMSGS


# Liste des adresses abonnement
#------------------------------
export LISTE_ADR_LISTE_TES="SERVEUR_TES"
export LISTE_ADR_LISTE_MP="MP_INPUT"
export LISTE_ADR_LISTE_JDB="SERVEUR_JDB"
export LISTE_ADR_LISTE_WOM="SERVEUR_WOM"
export LISTE_ADR_LISTE_SD="SERVEUR_SD"

# Liste des adresses pour adresser individuellement chaque processus
# les adresses doivent se retrouver dans les listes ci-dessus
#-----------------------------------------------------------------
export ADR_WOM_INPUT="SERVEUR_WOM"
export ADR_MP_INPUT="MP_INPUT"
export ADR_TES_INPUT="SERVEUR_TES"
export ADR_JDB_INPUT="SERVEUR_JDB"
export ADR_SD_INPUT="SERVEUR_SD"
```

```
# Liste des adresses pour terminer les processus
#-----------------------------------------------
export ADR_SD_FIN="SERVEUR_SD"
export ADR_WOM_FIN="SERVEUR_WOM"
export ADR_TES_FIN="SERVEUR_TES"
export ADR_SERVER_FIN="SERVEUR"
export ADR_JDBS_FIN="SERVEUR_JDB"
export ADR_MP_FIN="MP_INPUT"

# Definition des repertoires du sous-systeme REP
#-----------------------------------------------

# Command (shell script start/stop)
export REP_CMD_DIR=$REP_ROOT/cmd

# Executables
export REP_BIN_DIR=$REP_ROOT/bin
export REP_BIN_DEBUG_DIR=$REP_BIN_DIR/debug
export REP_BIN_NOMINAL_DIR=$REP_BIN_DIR/nominal

# Interfaces
export REP_PGF_IN_DIR=${REP_WORKING_ROOT}/input
export REP_PGF_OUT_DIR=${REP_WORKING_ROOT}/to_pgf

# Data
export REP_DATA_DIR=${REP_ROOT}/data

# Temporary directory for processing
export REP_TMP_DIR=${REP_WORKING_ROOT}/tmp
export REP_DATA_TMP_DIR=${REP_TMP_DIR}/data

# Tools
export REP_TOOLS_DIR=$REP_ROOT/tools

# Debug Data
export REP_DEBUG_DIR=${REP_DATA_TMP_DIR}

# Facility Identification
#------------------------
export CHN_FACILITY=I1

# Installation Type (CGS, STANDALONE)
#------------------------------------
export CHN_OPS_INSTAL_TYPE="STANDALONE"
export CHN_STANDALONE_LOG_FILE="${REP_TMP_DIR}/LOG_"
export CHN_STANDALONE_TRACE_FILE="${REP_TMP_DIR}/TRACE_"
export CHN_STANDALONE_CMD_TM_FILE="${REP_TMP_DIR}/CMD_"
export CHN_STANDALONE_HKTM_FILE="${REP_TMP_DIR}/HKTM_"

# MLA Installation
#-----------------
# MLA STUB
export LA_API_MODE=API_STUB
export LA_API_CMD=${REP_TMP_DIR}/MLA_CMD.txt
export LA_API_TLM=${REP_TMP_DIR}/MLA_TLM.txt
export LA_API_LOG=${REP_TMP_DIR}/MLA_LOG.txt
export LA_API_MLAHOST=
```

```
# XERCES Installation
#-----------------
if [ -$LIBPATH = "-" ]
then
export LIBPATH=$REP_BIN_DEBUG_DIR:$LIBPATH:/usr/lib:/lib
else
export LIBPATH=$REP_BIN_DEBUG_DIR:/usr/lib:/lib
fi
```

4.4.1.2.JDBS.env

```
#!/bin/ksh

# $Header $
# ----------------------------------------------------------------------*/
# Composant : JDBS
# CLASSE : Environnement
#
# Commentaire :
#    Parametrage du serveur de Messages JDB
#
# ----------------------------------------------------------------------*/
# HISTORIQUE
# VERSION : 21/03/2000 : A.MONTMORY
# Creation du fichier
#
# Origine     Date        Auteur   Commentaire
# ------------------------------------------
# FA-ID : <N°> : ../../.... : ... : ...
#
# FIN-HISTORIQUE
# ----------------------------------------------------------------------*/


# Variable d'environnement pour JDBS
#-----------------------------------

# Repertoire de stockage du fichier journal de bord temporaire
export REP_JDB_TEMPORAIRE=${REP_TMP_DIR}

# Nom du fichier journal de bord temporaire
export FIC_JDB="fichier.jdb"

# Repertoire de stockage du fichier journal de bord
export REP_REPERTOIRE_JDB=${REP_JDB_TEMPORAIRE}

# Localisation et nom du fichier verrou pour acces exclusif
# au fichier journal de bord temporaire
export FIC_VERROU=${REP_JDB_TEMPORAIRE}/verrou.lck
```

4.4.1.3.MP.env

```
#!/bin/ksh
# $Header: /home/ops_gc/cvs-repository/OPS/MP/src/MP.env,v 1.11 2004/09/22 09:40:27
bmassart Exp $
# ----------------------------------------------------------------------*/
# Composant : MP
# CLASSE : Environnement
#
# Commentaire :
#    Parametrage du Main Process
#
# ----------------------------------------------------------------------*/
# HISTORIQUE
#
# VERSION : 21/03/2000 : A.MONTMORY
# Creation du fichier
#
# SIF-00-001 12/07/2000 A.MONTMORY
# Mise en coherence configuration. Rajout de la definition des répertoires
# d'echange avec le GDD.
#
# SIF-00-057 06/09/2000 A.MONTMORY
# Changement du rapport d'envoi des Rapport de SI maintenant sous
# /to_cmc/rapport_ss_instruction/<mpid>
#
# ----------------------------------------------------------------------*/
# VERSION : 28/08/2002 : PBR
# Adapation a l'OPS-IASI
#
# OPS-I00-008 19/11/2002 PBR
# Une seule instance du WOM et du SD sont déclarées dans
# LISTE_CHN_PROC_APPLICATIFS.
#
# FA-ID :IA-2100-FA-171-CN : 09/03/2006 : B.MASSART
# Spacecraft identification added
#
# CR-ID :IA-2100-CR-292-CN : 09/03/2006 : B.MASSART
# FA-ID :IA-2100-FA-171-CN : 09/03/2006 : B.MASSART
#   Adding SPACECRAFT_ID dependance to Status file name.
#
# FIN_HISTORIQUE
# ----------------------------------------------------------------------*/

# MP server environment variables
#--------------------------------

# Service processes list to be activated by the MP
export LISTE_CHN_PROC_SERVICE="SERVER,TES,JDBS"

# Application process list to be activated by the MP
export LISTE_CHN_PROC_APPLICATIFS="WOM,SD"

# Resources status sending period (in seconds)
export NUM_HKTM_PERIODICITY=60
```

```
# Command listening period (in seconds)
export NUM_CMD_PERIODICITY=1

# CR292 CVS : Spacecraft identification
# M01, M02 or M03 for METOP satellite ;  N16 or N17 for NOAA satellite
export SPACECRAFT_ID=M01

# Default status file name FA171 CVS
export FIC_MP_FICHIER_STATUS=${REP_CONF_DIR}/Status_${SPACECRAFT_ID}.cfg

# Message catalog
export CAT_MESS=MP
export NLSPATH=${NLSPATH}:${REP_CONF_DIR}/%N.cat
```

4.4.1.4.SERVER.env

```
#!/bin/ksh
# $Header:$
# ------------------------------------------------------------------*/
# Composant : MSGS
# CLASSE : Environnement
#
# Commentaire :
#    Parametrage du Serveur de messages
#
# ------------------------------------------------------------------*/
# HISTORIQUE
# VERSION : 21/03/00 : A.MONTMORY
# Creation du fichier
#
# Origine     Date        Auteur   Commentaire
# ------------------------------------------
# FA-ID : <N°> : ../../.... : ... : ...
#
# FIN-HISTORIQUE
# ------------------------------------------------------------------*/

# Variable d'environnement pour MSGS
#---------------------------------
```

4.4.1.5.SD.env

```
#!/bin/ksh
# $Header: /home/ops_gc/cvs-repository/OPS/SD/FRW/src/SD.env,v 1.8 2004/03/18
09:21:43 jlp Exp $
# ------------------------------------------------------------------------*/
# Composant : SD
# CLASSE : Environnement
#
# Commentaire :
#    Parametrage du Serveur de Donnees
#
# ------------------------------------------------------------------------*/
# HISTORIQUE
#
# VERSION : 15/11/2002 : P. CABANE
# Creation du fichier
#
# Origine     Date        Auteur   Commentaire
# ----------------------------------------
# DM-ID :IA-2100-CR-286-CNE : 21/02/2006 : B.MASSART
# Ajout de la variable d'environnement DATE_MARGE_LIGNE_AVHRR
#
# DM-ID :IA-CR-2100-293-CNE : 08/03/2006 : B.GUENDE
# Ajout de la variable d'environnement CONTEXT_SOURCE
#
# FIN-HISTORIQUE
# ------------------------------------------------------------------------*/


# SD environment variables
#---------------------------------

# Messages list
export CAT_MESS=SD

# Messages reading timeout
export NUM_TIMEOUT_SD=0.1

# Ressources status sending period (in seconds)
export NUM_STATUS_RES_SD=15

# METOP configuration
export METOP_ENV=TEST

# CR286 CVS :AVHRR dating precision (in milliseconds)
export DATE_MARGE_LIGNE_AVHRR=3

# CR293 CVS : Source of generated context
export CONTEXT_SOURCE=CGS1
```

4.4.1.6.TES.env

```
#!/bin/ksh
# $Header$
# ------------------------------------------------------------------*/
# Composant : TES
# CLASSE : Environnement
#
# Commentaire :
#    Parametrage du Serveur de temps pour le sous systeme simule ARF
#
# ------------------------------------------------------------------*/
# HISTORIQUE
# VERSION : 21/03/2000 : A.MONTMORY
# Creation du fichier
#
# Origine     Date        Auteur   Commentaire
# ------------------------------------------
# FA-ID : <N°> : ../../.... : ... : ...
#
# FIN-HISTORIQUE
# ------------------------------------------------------------------*/

# Variable d'environnement pour TES
#---------------------------------

# Periode de traitement des evenements timer (en secondes)
export NUM_TIMEOUT_TES=0.1
```

4.4.1.7.WOM.env

```
#!/bin/ksh
# $Header $
# ----------------------------------------------------------------------*/
# Composant : WOM
# CLASSE : Environnement
#
# Commentaire :
#    Paramétrage du Serveur de Donnees
#
# ----------------------------------------------------------------------*/
# HISTORIQUE
#
# VERSION : 15/11/2002 : P. CABANE
# Creation du fichier
#
# Origine     Date        Auteur   Commentaire
# ------------------------------------------
# FA-ID : <N°> : ../../.... : ... : ...
#
# FIN-HISTORIQUE
# ----------------------------------------------------------------------*/


# Variable d'environnement pour WOM
#----------------------------------

# Catalogue de messages
export CAT_MESS=WOM

# Timeout d'attente pour la lecture des messages
export NUM_TIMEOUT_WOM=0.1

# Durée d'un granule (22 lignes * 8 secondes)
export NUM_DUREE_GRANULE=176
```

4.4.2.Data values configuration files

4.4.2.1.HKTM status

The initial values of the HKTM parameters are stored in the Status.cfg file. The following parameters are stored for each HKTM status :

PARAM_NAME	Name of the HKTM status
VALIDITE	0 : invalid (this HKTM status is never provided) 1 : valid (this HKTM status is provided when available).
DISPONIBILITE	0 : not available at OPS start up 1 : available at OPS start up (the VALEUR_CHAINE above is provided).
VALEUR_CHAINE	Initial value of the current HKTM status.

4.4.2.2.Processing Description

The chronology of IASI_L0 data processing and the dispatching onto the hardware configuration (number of used threads) are described in the following configuration files :

- OPS_SD.cfg,
- ModelesTraitements.cfg,
- ModelesTaches.cfg,
- ModelesActions.cfg.

These files are available under the OPS configuration directory.

4.4.2.2.1.OPS_SD.cfg Configuration File

The general parameters are stored in the “OPS_SD.cfg” configuration file :

- in the [MultiThreading] section, the “NombreDeThreads“ parameter defines the maximum number of thread allocated to a run. This number is equal to the number of computing thread (equal to the number of processors allocated to a run) plus one thread for the monitoring and control function.
- the [EtatServeur] section provides 2 parameters. The “Timeout” parameter (in milli seconds) is the maximum waiting delay for a thread availability. The “PeriodeDeScrutation” parameter (in milli-seconds) defines the actions triggering

periodicity.

An exemple of the OPS_SD.cfg configuration file is given below.

```
 [MultiThreading]
NombreDeThreads=5

[EtatServeur]
Timeout=100000
PeriodeDeScrutation=1000000
```

4.4.2.2.2.Others Configuration Files

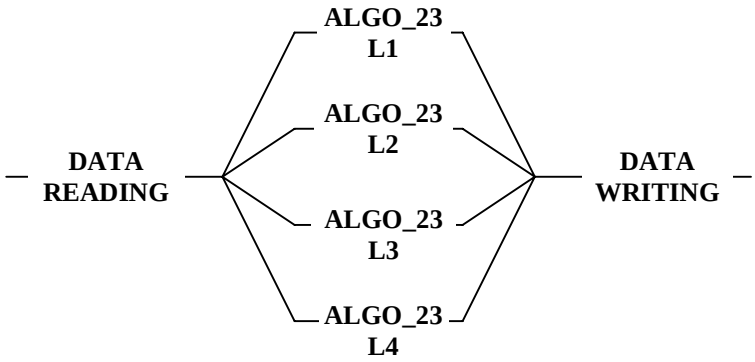
The others files describe the chronology of algorithms processing.

The “ModelesTraitements.cfg” configuration file describes for each OPS processing type (L00_1C, L1A_1C, L1B_1C) the chronology of tasks execution. The tasks are described in the “ModelesTaches.cfg” configuration file. Each task is compound of actions which are executed in parallel or in sequence. In the “ModelesActions.cfg” configuration file each action is joinned to a sequence of low level IASI L1 processing algorithms. In our application, a task is mapped to a single action.

ModelesTraitements.cfg

This file describes a tasks execution diagram in a multithread architecture. This description models either the execution of tasks in sequence, or the execution of tasks in parallel or the synchronization of tasks.

The following figure shows an example of tasks execution diagram. A data reading task is executed before the Algo_23 task which is executed for each line on a thread. When all the line are processed with the Algo_23 task, the data produced are writing.



The syntax of this configuration file is the following one :

```
[TREATMENT NAME]
NOMBRE TACHES=xxx
```

```
TACHE1=xxx
TYPE TACHE1={ RDV | LIGNE | AUCUN }
…
TACHEn=xxx
TYPE TACHEn={ RDV | LIGNE | AUCUN }
```

The <TYPE TACHE> parameter defines the mechanism of synchronization between the tasks :

- a RDV task (RenDez-Vous) is a task which is executed only if all the previous task are finished,
- a LIGNE task defines a task which is instanciated for each IASI L0 line. Each instanciated task is executed a dedicated thread. So, the IASI L0 lines are processed in parallel on each available thread.
- a AUCUN task is a task which is executed in parallel of another LIGNE or AUCUN task.

Thanks to this formalism, the execution diagram depicted above is modelled by the following 3 tasks :

- one RDV task DATA_READING,
- one LIGNE task ALGO_23,
- one RDV task DATA_WRITING,

which are stored in the following way in the configuration file :

```
[0-1A]
NOMBRE TACHES=3
TACHE1=DATA_READING
TYPE TACHE1=RDV
TACHE2=ALGO_23
TYPE TACHE2=LIGNE
TACHE3=DATA_WRITING
TYPE TACHE3=RDV
```

An example of the ModelesTraitements.cfg configuration file is given below.

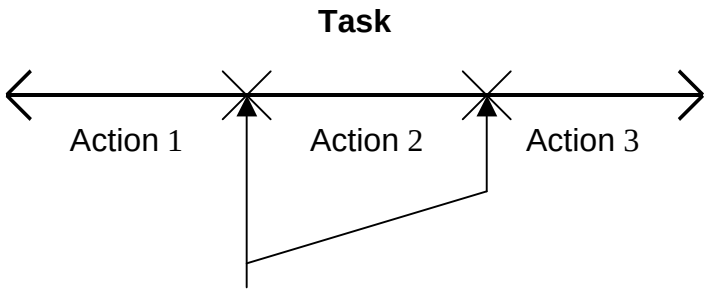
```
[L1a]
NOMBRE TACHES=4
TACHE1=ChaineImageDeb
TYPE TACHE1=RDV
TACHE2=ChaineImageISRFEM
TYPE TACHE2=LIGNE
TACHE3=FiltrageAxeInterf
TYPE TACHE3=RDV
TACHE4=ChaineProd0_1C
TYPE TACHE4=LIGNE

[L1b]
NOMBRE TACHES=2
TACHE1=FiltrageAxeInterf
TYPE TACHE1=RDV
TACHE2=ChaineProd1A_1C
TYPE TACHE2=LIGNE
```

```
[L1c]
NOMBRE TACHES=2
TACHE1=FiltrageAxeInterf
TYPE TACHE1=RDV
TACHE2=ChaineProd1B_1C
TYPE TACHE2=LIGNE
```

ModelesTaches.cfg

This file describes the division of a task in a sequence of actions. This mechanism allows to set check points between the actions in order to handle if needed events from another thread. In OPS context, a task is mapped to a single action.



Break points checking if end of the task has been requested

The syntax of this configuration file is :

```
[TASK NAME]
SIMULATION=-1
NOMBRE ACTIONS=xxx
ACTION1=xxx
…
ACTIONn=xxx
```

If the <SIMULATION> parameter is defined (expressed in seconds and different from –1 value), the task is not executed, but the task execution duration is simulated. The following values are available :

- -1 : the real task is executed,
- >0 : task is simulated (sleeping of "SIMULATION" seconds),
- other values : sleeping of 1 second.

An example of the ModelesTaches.cfg configuration file is given below.

```
[ChaineImageDeb]
SIMULATION=-1
NOMBRE ACTIONS=1
ACTION1=ChaineImageDeb

[ChaineImageISRFEM]
SIMULATION=-1
NOMBRE ACTIONS=1
ACTION1=ChaineImageISRFEM

[FiltrageAxeInterf]
SIMULATION=-1
NOMBRE ACTIONS=1
ACTION1=FiltrageAxeInterf

[ChaineProd0_1C]
SIMULATION=-1
NOMBRE ACTIONS=1
ACTION1=ChaineProd0_1C

[ChaineProd1A_1C]
SIMULATION=-1
NOMBRE ACTIONS=1
ACTION1=ChaineProd1A_1C

[ChaineProd1B_1C]
SIMULATION=-1
NOMBRE ACTIONS=1
ACTION1=ChaineProd1B_1C
```

ModelesActions.cfg

This configuration file provides a mechanism to specify if the execution of an action is simulated or not.

The <SIMULATION> parameter has the same meaning as <SIMULATION> parameter in ModelesTaches.cfg.

An example of the ModelesActions.cfg configuration file is given below.

```
[ChaineImageDeb]
SIMULATION=-1

[ChaineImageISRFEM]
SIMULATION=-1

[FiltrageAxeInterf]
SIMULATION=-1

[ChaineProd0_1C]
SIMULATION=-1
```

```
[ChaineProd1A_1C]
SIMULATION=-1

[ChaineProd1B_1C]
SIMULATION=-1
```

5.DATA AND VALIDATION TOOLS INSTALLATION

Data and validation tools are installed in the /OPS_version/OPS_Vx-x/tools and /OPS_version/OPS_Vx-x/data directories as described in the chapter 4.3.

The validation tools are delivered in the installation kit and are installed on the target computer by the installation procedures.

The data files are delivered onto two dedicated data cartridge (DDS-3 format):

- The first contain the validation environment

To extract the data set, execute the following commands :

Connect as **iasi_1mg** user

cd <*WORKING_ROOT_DIR*>/OPS_versions/OPS_<version>/data

tar xvf *tape_device* to extract the data

- The second one contain the reference data (L0/L1)

To extract the data set, execute the following commands

Connect as **iasi_1mg** user

mkdir <*WORKING_ROOT_DIR*>/../../../tmp (not necessary if validation test INS_OPS_N_02 is played)

cd <*WORKING_ROOT_DIR*>/../../../tmp

tar xvf *tape_device* to extract the data