IASI L1 Interface Control Document

	Responsibility - Company	Date	Signature
Prepared by	PPF_IASI_L1 Facility Engineer - Alcatel	07/03/03	p.o. MARCHAND
Approved by (Technical) F. DIDELOT	Technical Manager - Alcatel	10/3/03	
Approved by (Quality) M QUERE	Quality Assurance Manager - Alcatel	10 MAR 2003	H. Quéré
Authorised by D JAN	Project Manager - Alcatel	10/03/03	

Configuration Item number:
if applicable

Summary: This document describe all external interfaces of the PPF along with their exact format

Document submitted to EUMETSAT GS Management for:

☒ Approval ☐ Review ☐ information

Signature

LEO/Rec. 7/3/05

ALCATEL
SPACE

IASI L1 Interface Control Document

	Responsibility - Company	Date	Signature
Prepared by	PPF_IASI_L1 Facility Engineer - Alcatel	07/03/03	p.o MARCHANDOR Marchandon
Approved by (Technical) F. DIDELOT	Technical Manager - Alcatel	10/3/03	
Approved by (Quality) M QUERE	Quality Assurance Manager - Alcatel	10 MAR 2003	H. Quéré
Authorised by D JAN	Project Manager - Alcatel	10/03/03	

Configuration Item number:
if applicable

Summary: This document describe all external interfaces of the PPF along with their exact format

Document submitted to EUMETSAT GS Management for:

☒ **Approval** ☐ **Review** ☐ *Information*

Signature

ALCATEL
SPACE

CHANGE RECORDS

ISSUE	DATE	§ CHANGE RECORD of EPS-ASPI-IR-0282	AUTHOR
1 0	30/09/99	First issue of the document	
1.1	15/05/01		
2 0	26/11/2001	Addition of command Step and suppression of the *polling for data* policy Rewriting of sections Workorder, stage & command TM The Workorder section includes rules aimed at guaranteing the continuity of the processing Modification of section Commands Clarification of sections Auxiliary data, Configured items, reports The static mode (several steps in a single run) is now restricted to IASI1 use Other PPFs are launched anew for each granule Addition of sections commands & TM sequencing, local info logging	
3 1	21/05/2002	The policy around the time interval specified in the workorder has been reformated in a separate section and reworded Error in Section 3 1 9 Reports: the report filename is send on a file delivery TM and not on the command completion A section about aux data including OBT-UTC, xml config and PPF generated aux data has been added In annex defining the MPHR filling responsibility, the fields constituting the OSV shall now be filled by PPF. Definition of a filling for dummy records Addition of an event trace diagram for the static case in section 3 1 Addition of section Data Registration including a detail of the interface with MCS	FF

3 2	30/09/02	RIDs from CGS CDR : - CGSCDR-PPF-IL1IS-CNES-RM-004 Removal of all reference to DDT - CGSCDR-PPF-IL1IS-CNES-RM-015 · "data registration doc" has been replaced as reference by "constraint & rules" and section "PPF registration" - CGSCDR-PPF-IL1IS-CNES-RM-016 - CGSCDR-PPF-IL1IS-CNES-RM-019 · overlap for AVHRR L1b has been added - CGSCDR-PPF-IL1IS-CNES-RM-023 . - CGSCDR-PPF-IL1IS-CNES-RM-020 - Update includes Default rule for the selection of aux data Confirmation that L0 includes the OSV in MPH In report MdrOk becomes MdrTotalCount Command completion has field status set to *aborted* in case of a break Addition of rule9 when a command is rejected, no completion is sent Conversion OBT-UTC no longer requested Use of UTC from GRH is requested 80 avhrr lines are necessary before and after the time interval to be processed - CGSCDR-PPF-IL1IS-CNES-RM-023 Refer to section 3 1 14 - CGSCDR-PPF-IL1IS-CNES-RM-024 CGSCDR-PPF-IL1IS-CNES-RM-009 1 Aux data files have been removed and mentionned as configuration files (section 3 3 2) Mention of predicted manœuvres and umbra/penumbra events Constraint on the location of the Workorder DTD	FF
3 3e	07/03/03	Comments from Workshop EPS-ASPI-MN-0869 #5 annex modified wrt GPFS 6 3 #6 BREAK for last or unique STEP clarified Comments of fax EUM EPS GSE FAX 02 371 on IASI L1 ISD responded in fax DSO/FF/OS/S/P/864/03 #4 StageTM, the filename is relative to the working root directory #5 format of the start time is consistent #6 . additional comments on SUSPEND command in section 6 Consistency of the examples wrt the dtd Workorder sample root tag <PPF_Work_Order> is added PPF report sample - root tag is corrected to be <PPF_report> - <MdrOk> becomes <MdrTotalCount>	

ISSUE	DATE	§ CHANGE RECORD of EPS-ASP-ID-1149	AUTHOR
1 0	07/03/03	Document based on EPS-ASPI-IR-0282	

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 PURPOSE OF THE DOCUMENT

This document describes all external interfaces of a PPF facility It gives the design of these interfaces.

1.2 SCOPE OF INTERFACES

The PPF interfaces only with the PGF and the MCS Local Agent (MLA)

Data flows input or output between PPFs and PGF have been identified through separate interfaces

PPF are given at startup command line parameters and auxiliary input data They receive the data to process and generate progress status and reports up to the end of corresponding production They finally make available higher level products on their disks. The PGF is interfaced with the PPF through a set of commands supported by stdin/stdout mechanism

1.3 APPLICABLE AND REFERENCE DOCUMENTS

1.3.1 Applicable document

Table 1.3-1 Applicable Documents

Document	Identifier	Internal Reference
AVHRR Product Format Specification	EPS MIS SPE 97231	EUM-AD-0037
Constraints and rules for PPS & section 3 2 of this doc	EPS-ASPI-SP-0261	AD9
EPS Generic product format specification	EPS/CGS/SPE/96167	EUM-AD-0036
EPS Product Conventions document	EUM EPS SYS TEN 990007	EUM-AD-0098
IASI L1 Product Format Specification	EPS MIS SPE 990003	EUM-AD-0118
ISI MCS local agent (MLA) Software User Manual	ISI-383-0031-05	

1.3.2 Reference Documents

Table 1.3-2 Reference Documents

Document	Identifier	Internal Reference
Glossary of terms and abbreviations list	EPS-ASPI-LD-0010	A-AD-0010

1.4 DOCUMENT OVERVIEW

This document contains 4 chapters which are structured as follows

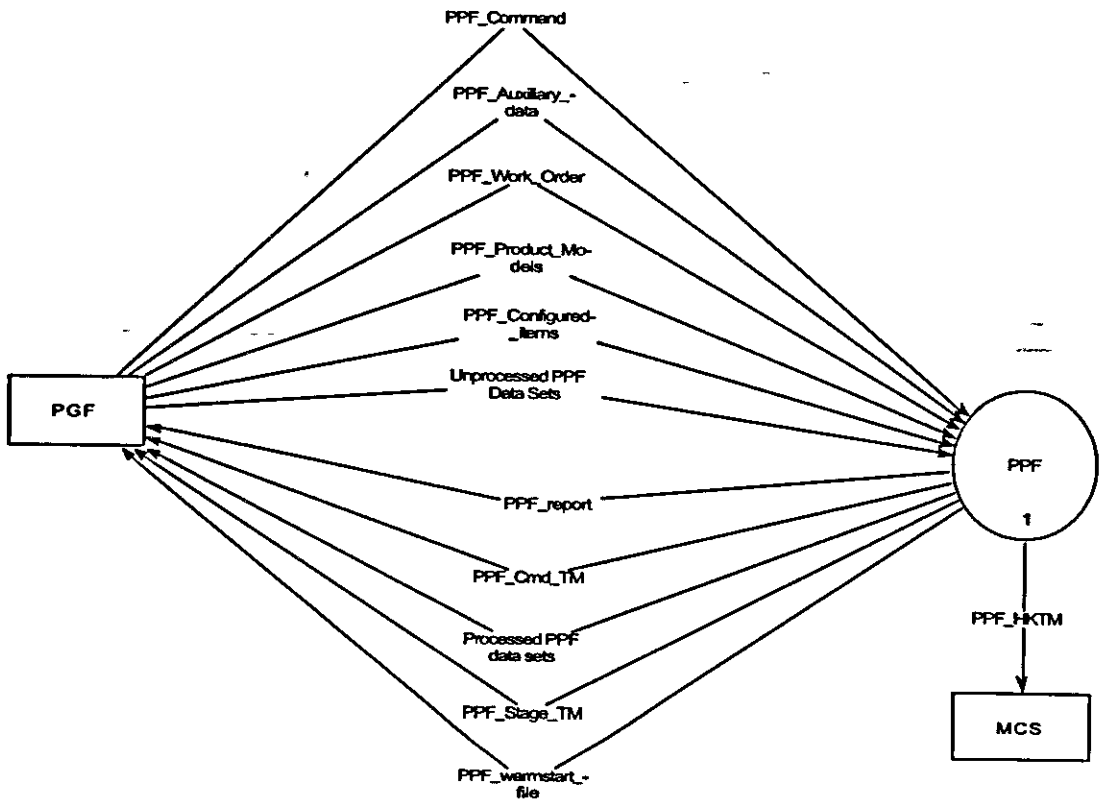
- Chapter 1 this chapter,
- Chapter 2 presents the PPF data flow context diagram in order to give an overview of its external interfaces,
- Chapter 3 defines flows and interfaces mechanisms between PPF and PGF. Last section of chapter 3 gives the main principles of the communication with MCS ,
- Chapter 4 defined the link types between the PPF_IASI_L1 and the EPS CGS ;
- Chapter 5 contains characteristics and data structure of each interface ,
- Chapter 6 contains the structure data of the interfaces
- Chapter 7 contains a glossary ;
- Chapter 8 contains an index

2. OVERVIEW OF INTERFACES

2.1 CONTEXT DIAGRAM

The reduce PGF Context diagram to PPF scope is shown on the following figure.

Figure 2.1-1 PPF generic context diagram



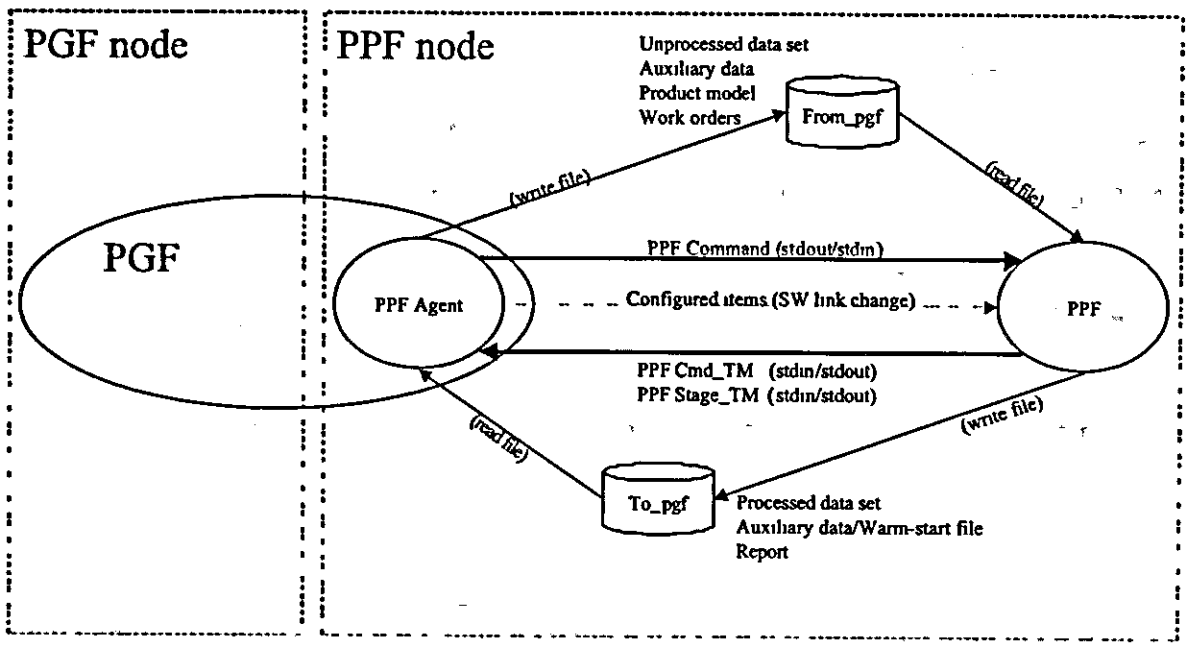
The interfaces are as follow

- All kinds of PPF are activated with unprocessed parent products and appropriate parameters The PGF is receiving intermediate and final status and reports from current PPF processing After a while, the PPF issues the higher level products required The PPF can also be required for aborting its task (STOP command),
- PPF Software and configuration updates can also be issued by PGF, on MCS request They are not handled by the PPF but through basic file replacements or updates operated through the OS while the PPF is not running

THIS PAGE IS INTETNIONALLY LEFT BLANK

3. INTERFACE FLOWS

The following diagram present the communication with PGF. It is entirely managed on the local host machine ("PPF agent" means agent of the PGF controlling a PPF) "Configured Item" is a switch of a directory link (a unix feature) that does not involve PPF software.



3.1 PGF TO PPF INTERFACE

PGF is the software managing the overall computations All the PPFs are activated with the same set of interfaces

A PPF receives infos from

- Command line arguments ,
- Commands (e g STOP) ,
- Files (Work-order, L0 or L1, Auxiliary data)

All the necessary files are sent to the PPF file system before each command Step is issued

3.1.1 PPF Configured items

The configuration of a PPF includes *executables, startup script, config file* (a few operational parameters)

At least 2 versions of configuration are maintained in any PPF node, at the same time without interfering
PGF manages the versions as distinct directory trees, such a tree is static and *does not include* the working root directory for file exchanges
PGF switches from a version to another by mean of Unix soft links

A PPF can assume its config items are unchanged during a whole run

Besides, the startup script file allows to set some *configuration* information, in particular

- the full path name of the config file (if any) ;
- The number of processors allocated to a run (for IASI1 only)

They are set inside the script either as parameters of the command line starting the exe file, or through an environment variable (set in the script)
The PPF shall not assume any specific environment variable to be set prior to the startup script execution

The PPF shall not make any assumption about the static configuration trees (the only 2 info needed path to the exe and path to the config file are set in the startup script)

3.1.2 Start (i e. launch)

The PPF software is started through a Unix command line calling a dedicated shell script
This script accepts the following command line arguments

- A command_Id an integer to be used as a reference for later communications – a variable number of characters in [0-9] ,
- the working root directory – starting by a '/ ' and refering to an absolute path,
- a mode (Nominal versus Investigation) – possibly triggering 2 distinct executables – A valid value is `N` or `I` (uppercase character)
  The investigation mode is intended to be run in the operationnal context, the penalty on execution time shall then be less than a factor 2 or 3

Each *working root directory* contains the following subdirectories:

to_pgf	*where the PPF shall write L1 or L2, possible aux files, reports PPF ensures the cleanup of this directory at the start*
tmp	*an area under PPF responsibility – including responsibility for its cleanup*
xxx_wo	*used for the workorders – no cleanup of workorder, a rolling set of 999 workorders is kept for investigation purposes (especially if the PPF crashes) The PPF shall not assume a predefined name for this directory as the path from the working root directory is specified along with the name as a parameter of each STEP command*
yyy_aux	*The PPF may find there certain aux data, in such a case, yyy_aux is mentionned in the workorder - the name and usage of such a directory is the CGS integration team responsibility*
zzz_L0	*idem for level 0, level 1*

The directories xxx_wo, yyy_aux, zzz_L0 are under control of the CGS integration team and set in a configuration file of PGF Other directories may be added, in particular for the product model, by decision of the CGS integration team Their names are totally free and no assumption must be made on the inclusion of any substring 'wo', 'aux',... in these names

This configurability allows, for each aux data type and even for L0 or L1, to choose between a local copy on the PPF local disk, or a PPF remote access to a global repository, without any impact on the PPF side

The cleanup policy of the working root directory follows the principle "The facility that creates or copy a file is in charge of its removal". Consequently, the PPF is expected to perform a clean-up on its start rather than on its termination This solution allows a post problem analysis and is needed anyway in the case of a run following a kill. In practice, PPF is responsible for cleaning subdirectories to_pgf and tmp

3.1.3 Commands

The following commands can be sent by the PGF.

Step *+ workorder_filename* a request to process the job specified in the workorder file specified as parameter of the command The input files needed for the job are assumed to be available on reception of the Step command
On termination of the job, PPF shutdown – *except IASI1 which shall wait for a new Step*
The filename as specified on the command line can be either absolute or relative to the working root directory It can then be processed in the same way than the unprocessed data and aux data filenames (see 3 1 4 Workorder, alinea 6 & 7)

Break: aborts the current processing step but does not terminate the run, processing of commands resume with the next available command (if no command is available the PPF waits for one) – *this command is dedicated to IASI1, other PPFs shall not reject a Break but process it as a Stop*

Stop it is a soft shutdown The PPF shall respond with a cleanup task and the delivery of a report The PPF shall terminate its own processes and threads properly No further command is processed after the reception of a stop

Suspend: pauses the processing but does not free the resources Once suspended, all commands but Stop and Resume are discarded

Resume: restart the processing after a supend It has no effect if the PPF was not suspended

Abort: abort a current processing The Main Process (created at launch) and its childs will be forced to terminate by a Unix signal This function requires no specific code in the PPF

Commands are received on the Unix pipe *StdIn* according to the following pattern

PPF_Command { *Command_type* Step, Break, Stop, Suspend, Resume
Command_Id Integer *sort of a counter used as reference for later acks*
workorder_filename String *along with the 'STEP' command only* }

Under a full ASCII format, all fields of variable size separated by white spaces and ending by a newline, e g .

```
STEP 137 workorder/orbit178_wo_001
SUSPEND 1815
```

3.1.4 Workorder

The workorder accurately defines the processing to perform - two different runs using the same work-order and associated data files, shall give the same results The major fields in the work order file are

1. Processing type
   - a string of variable size specific to each PPF (case non-sensitive)
   It specifies the requested outputs, e g L1a, L1b,
   The possible values for *processing type* are defined in each PPF user manual (SUM)
2. Start time T_0 , *UTC time, format* YYYYMMDDHHMMSSZ or YYYYMMDDHHMMSS TTTZ , .TTT *is an optional decimal fraction of seconds Z stands for character 'Z' and acts as a terminator.*
3. Stop time T_1 , *idem*

$]T_0, T_1]$ is the time span of the data to process, it roughly represents the span of the output granule to be generated
A run on $]T_0, T_1]$ followed by a run on $]T_1, T_2]$ lead to contiguous, non-overlapping output products

Moreover, PPF must ensure that the stoptime of the output products (if generated) is inside $]T_0, T_1]$. In particular, in the special case of missing inputs at the end, the dummy record added to flag the lack of data shall have its endtime set to T_1

The exact interpretation of $]T_0, T_1]$ is specific to each PPF and detailed in next section

4. Time Interval Flag 'First', 'Middle', 'Last', 'Full' (case non-sensitive).
   'First', 'Middle', 'Last' identify the position of the current processing in the dump, 'Full' is used in case of a standalone run
   Information is useless for Ascat, Gome, Atovs, Gras, IasiL2
   For IASI L1, 'Last' or 'Full' triggers the shutdown of the PPF on completion of this workorder

5. Time Interval Counter: integer, *a variable number of characters*
   The rank of the current processing in the dump 0 when Time Interval Type is 'First', undefined when Time Interval Type is 'Full'.

6. Unprocessed data (the names of L0 or L1) – a list of *filenames*

   The absolute path to a file can be built as
   - if *filename* begins with a '/' then it is *filename*
   - else build *working_root_dir + filename*

   filename is assumed to be exact (not a template) and may include several '/' – if the file does not exist, it is a fatal error

7. The aux data filenames – a list of *filenames (exact or template)*

   The absolute path to a file can then be built as
   - if *filename* begins with a '/' then it is *filename*
   - else build *working_root_dir + filename*

   filename is assumed to be exact or to be a template if '*' & '?' are found in the filename string The case of missing file is dealt with as follows:
   - An exact filename (the string does not include any '*' or '?') specifies a mandatory file, that is if the file does not exist, it is a fatal error However if the file exists, allows processing but shall be used out of its validity range, only a warning is logged
   - When a pattern is specified (one or several '* or '?' in the filename string), if no file matches the pattern or none is acceptable then log an info message and trigger a degraded processing (the degraded processing can be to terminate)

   The PPF does not wait for a potentially arriving aux data PGF is in charge of waiting A time-out, if no arrival occurs, allows the PGF to start a PPF without an aux data (according to a PGF configuration)

8 The names of the product models – a list of *filenames*

The absolute path to the file can then be built as

working_root_dir + filename - filename should not start with a '/'

filename is assumed to be the exact name of an existing file

The name of the product finally delivered by the PPF will however differ. at least the start & stop times shall be updated

A disctinct product model for each requested *product* shall be delivered No product model is delivered for aux files or context files

File name convention for work order The PPF is not expected to make any check on the filename specified along with the Step command
The convention is given in [AD9]

The workorder internal format is XML (DTD PPF_Work_Order dtd)

An example of workorder is given below

```
<PPF_Work_Order>
    <ProcessingType>Iasi_L0_1c</ProcessingType>
    <StartTime>20011022185535Z</StartTime>
    <StopTime>20011022185835.018567</StopTime>
    <TimeIntervalFlag>MIDDLE</TimeIntervalFlag>
    <TimeIntervalCounter>31</TimeIntervalCounter>
    <UnProcData>Un_proc/IASI_xxx_00_M01</UnProcData>
    <AuxData>Un_proc/AVHR_xxx_1B_M01</AuxData>
    <AuxData>/EPS/GlobalAuxData/xxxx_CFG_00_M01_20011008143513Z_xxxxxxxxxxxxxxxxZ_
        CGS1_0001_0000_0001_0000_20011009120700Z_R_F_xxxxxxxxxxxxxxxxZ</AuxData>
    <AuxData>/aux_data/iasi/xxxx_EOP_00_M01_20010906143000Z_*Z_CGS1_0001_0000_0001_
        0000_20010900Z_R_F_*Z</AuxData>
    <ProdModel>product_model/IASI_xxx_1A_M01</ProdModel>
</PPF_Work_Order>
```

Location of DTD file If the PPF needs the DTD file available at run time, It shall be able to use a file located at a place different from what may be specified in the workorders themselves (the second line of an XML file usually contains the full path to its DTD)
The DTD shall actually be statically stored, as any executable or config file, in the installation directory tree

3.1.5 Time Interval

The time interval specified in the workorder is only indicative as the generated product shall respect the natural boundaries of the products This section details the rules controlling the determination of these boundaries

A prerequisite is first stated, then the IASI1 PPF commitment follows

Require PPF needs as input both Iasi L0 and Avhrr L1b

L0 in input must encompass $]T_0 - u, T_1]$, u = 1 IASI line = 8s
the additional line allow to be sure to cover any line ending in the interval $]T_0, T_1]$
note each line is scattered on several source packets

Avhrr L1b in input is compounded of 2 PDUs covering at least $]T_0 - w, T_1 + w]$,
where w = 80 AVHRR lines = 14s , allows to be sure that 80 Avhrr lines are available ahead (resp after) the current IASI FOVS
Note : 80 AVHRR lines are necessary to perform coregistration of IASI imagettes with avhrr

Ensure The output L1b includes all the *lines* ending in $]T_0, T_1]$
The maximum shift of L1b relatively to $]T_0, T_1]$ is line_duration = 8s

Additionally, the times to be used for the above determination (determining if a given record shall be processed) must imperatively be the UTC times from the record headers (MDR.GRH RECORD_START_TIME) – Only these times are guaranteed to be consistent with the time window specified in the work order. Note that RECORD_START_TIME and RECORD_STOP_TIME of any given L0 MDR are equal

3.1.6 Product Model

The product model is used to convey information requested in a MPH but unknown to PPFs. One product model per type of product to be generated is delivered by PGF.

The list of fields filled by PGF (that are not to be modified by the PPF) is provided in Annex A

The info provided in the product model is not intended to be used by the PPF. In particular, the Orbit State Vector which is available in the product model , is also available in the MPH of the L0 or L1, the second source shall be used for the PPF computations

In some special cases (GomeSPA, AscatCalibrationCampaign) the job specified in the workorder does not trigger the generation of any product but of an aux data file In these cases, no product model is provided

3.1.7 Auxiliary Data

Aux data are made available to the PPF in its file system, in directories specified in the workorder (either absolute path or relative to the working root directory)

Each aux data have a *type* carved in its filename e g characters 0 to 11
and a *start_of_validity* date : e g characters 12 to 26
The file name convention is close to the product file name Eumetsat convention, the exact definition of the *type* and *validity time* slots is in [AD9]

The format of an aux data file is deduced from its type No standard MPH is expected in an aux file.

Selection Several aux files of the same type may be available and the PPF selects the most appropriate one according to its *start_of_validity* date
If needed by the algorithm, other fields in the filename could be used to help the selection Such a field would use characters 62 . 71 (the last field of an aux file name)
PPF can also use VALIDITY_END and CREATION_DATE
Default rule is

- select first the auxiliary data covering the relevant sensing coverage with respect to the validity fields of the file name,
- if several files are available, select the auxiliary data file with the most recent creation_date

In Any aux file delivered to the PPF is read-only even if not protected through the file system, it shall not be modified by the PPF

Out An aux file can be produced by the PPF. Its file name must comply to the pattern defined in [AD9] Some fields in the name will be copied from L0 filename
It is written in sub-directory *to_pgf* of the working root directory

According to these rules, a *dynamic* aux file is a file regularly re-issued by the PPF These files are discussed in the last section of chapter 3

Context file refers to data saved and retrieved from one run to another (e g a set of parameters regularly re-estimated to best estimate the status of the instrument at a given time)
Contexts are implemented as dynamic auxiliary files re-issued on completion of each Step e g xxx_calib_utc1 is taken as input of a processing while xxx_calib_utc2 is delivered as output
All successive issues of an aux file are kept (for a limited period) by the PGF
During a reprocessing, the successive issues of each file type are made available to the PPF which shall select the most appropriate one

3.1.8 Unprocessed Data files (main input)

The Unprocessed Data files (refered as slice for L0, as PDU for L1) will contain MPH, SPH and measurement records For a given run, it may encompass a variable time interval and may be compounded of several files

They are located in the file system of the PPF as detailled in the workorder section of this document

L0 and higher level includes the OSV (orbit state vector) in the MPH This information shall be used for geolocation
In the same way, each MDR of a L0 includes the UTC time of the measurement This time (MDR GRH RECORD_START_TIME) shall be used for the geolocation of the measurement unless specified otherwise by the algorithm specification

3.1.9 Reports

A report is created on completion of each Step command, processing being either successful, in error or as a result of a Stop or a Break command

The availability of a report is mentionned by a specific file delivery TM This TM specifies the name of the report as parameter.

The report includes·

1. copy of the original work-order.
2. Filename of each product generated during the step If a failure occurred which prevent the product to be generated, only the product identification is used for the filename. The product id is made of 15 characters
   (<INST>_<PROD_TYPE>_<PROC_LEVEL>_<SPACECRAFT>)
3. List of all files actually used as input, as they may be different from the one specified in the workorder (or as the workorder may contain templates)
4. Processing PCD.
5. Elapsed time and user time for each processing stage performed.
6. Concise LOG messages to indicate any cause of failure

When the PPF processing encounters fatal errors, the following details are included in the report.

1. failure to find, read, or write a file.
2. invalid file format or contents (workorder, aux data, L0, ...)

The file name convention for the reports is *work_order_name*_**001.rpt**

The format of the report is defined by a specific DTD An example (not realistic nor fully consistent) is given below to introduce the contents and syntax

```
<?xml version=1.0 encoding="UTF-8"?>  -- however, only chars in 0  127 are expected
<!DOCTYPE Report SYSTEM "/some/path/to/report.dtd">

<PPF_report>

    <WorkOrder> -- copy of the workorder content except for the preliminary lines
        <ProcessingType> … </ProcessingType>
        <ProdModel> … </ProdModel>
    </WorkOrder>

    <StageInfo> -- the PPF stages are defined in a specific way by each PPF
        <StageName>L1a</StageName>  -- name on PPF will
```



```
        <!-- - - - - - - - - - - - the files actually used for processing - - - - - - - - - - - - -->
        <UsedFileName>/absolute/path/ASCA_AUX_1A_M01_200210064511_…</UsedFileName>
        <UsedFileName>/absolute/path/COMM_AUX_xx_M01_200110064511_…</UsedFileName>

        <GeneratedFile> -- a separate section for each file generated in the stage
            <FileName>ASCA_xxx_1A_M01_2001100645112_…</FileName>
              -- use " temporary " if L1a is not output (when a direct processing from L0 to L1b is allowed)

            <MdrTotalCount>4213</MdrTotalCount>
            <MdrDegraded>57</MdrDegraded>   -- identical to field in MPHR

            <DataGap>   -- summary of Gaps due to a given reason
                <GapReason>(01) dummy input data</GapReason>
                <GapCount>2</GapCount>   -- Number of gaps due to the reason above
            </DataGap>

        </GeneratedFile>

        <GeneratedFile> -- the context file
            <FileName>ASCA_CTX_xx_M01_2001100645112_…</FileName>
        </GeneratedFile>

        <GeneratedFile> -- as example of an aux file generated on a data event (not requested in wo)
            <FileName>ASCA_GAP_xx_M01_2001100645112_…</FileName>
        </GeneratedFile>

        <MHSTelemetry>   -- this section exists only for MHS & HIRS
            <![CDATA[   data extracted from source packets   ]]>
        </MHSTelemetry>

        <ProcessingInfo>
            <ProcessingStart>20020401123456.432Z</ProcessingStart>
            <ProcessingStop>20020401123741 456Z</ProcessingStop>
            <ElapsedTime>123 450</ElapsedTime>
            <UserTime>100 789</UserTime>
        </ProcessingInfo>

        <LogMessage>ok</LogMessage>

    </StageInfo>


    <StageInfo>
    <StageName>L1b</StageName>

     <UsedFileName>/absolute/path/ASCA_xxx_1A_M01_200110064511_</UsedFileName>
        <UsedFileName>/absolute/path/COMM_AUX_xx_M01_200110064511_</UsedFileName>

        <GeneratedFile>
            <FileName>ASCA_SZO_1B_M01_20020401121547_…</FileName>
            <MdrTotalCount>4213</MdrTotalCount>
            <MdrDegraded>57</MdrDegraded>
        </GeneratedFile>
```



```
        <GeneratedFile>
            <FileName>ASCA_xxx_1B_M01_2002100645107_.</FileName>
            <MdrTotalCount>4213</MdrTotalCount>
            <MdrDegraded>0</MdrDegraded>

            <DataGap>  -- summary of Gaps due to a given reason
                <GapReason>(01) dummy input data</GapReason>
                <GapCount>2</GapCount>  -- Number of gaps due to the reason above
            </DataGap>

            <DataGap>
                <GapReason>(05) processing impossible</GapReason>
                <GapCount>1</GapCount>
            </DataGap>

        </GeneratedFile>

        <ProcessingInfo>
            <ProcessingStart>20020401123456 432Z</ProcessingStart>
            <ProcessingStop>20020401123741.456Z</ProcessingStop>
            <ElapsedTime>123.450</ElapsedTime>
            <UserTime>100.789</UserTime>
        </ProcessingInfo>

        <LogMessage>Warning  Not enough L1b coverage</LogMessage>

    </StageInfo>

</PPF_report>
```

Note1 possible reasons for data gaps shall be numbered, possible values identified up to now are

"(01) dummy input data" *a dummy record is present in one of the mandatory inputs*

"(02) gap due to transponder echoes" *for Ascat L1b transponder echoes encountered in L1a*

"(03) gap due to GCM packets" *for Ascat L1b, when GCM are encountered in L1a*

"(04) gap due to ?" *for Gome L1b calibration measurements in L1a ?*

"(05) processing impossible" *the values of the input does not allow to run through any processing*

Then at most 5 different sections DataGap can appear in a given GeneratedFile section The reasons not represented by any data gap in the file are not mentionned at all (a count of zero shall then never occur and the nominal case is no DataGap section at all)

Other numbers can be defined to represent special cases not already identified

3.1.10 Processed data (main output)

L1 or L2 generated by the PPF are delivered in files (sometimes refered to as PDU) complying to the format of a product with MPH-SPH-IPRs-data

They shall be written in sub-directory *to_pgf* of the working root directory
Their name must comply to the product model name specified in the workorder with their time fields updated to match the actual content of the file

Start time and stop time fields are specific slots in the file name They are defined in the product naming convention document The file name of a newly generated L1 or L2 is sent as a parameter of the TM reporting the delivery.

The PPF is responsible for the cleaning of the to_pgf directory (for processed data and reports), *on start* of the PPF

3.1.11 Dummy records in products (as input or output)

In case part of a product cannot be generated, either because necessary inputs are missing, either because the values themselves are unsuited for processing, a corresponding dummy record shall be output to materialise the gap in the output

The implementation of the dummy record is specified in [EUM-AD-0036]

3.1.12 Stage & Command TM (from PPF to PGF)

TM stands for telemetry and, in this section, refers to formated messages sent to PGF. They are restricted to events related to the system, that is, not intended to report computation progress (statuses sent to MCS cover this need).

TM are sent on the Unix pipe *StdOut*

The requested TMs with their associated parameters are described below

Command Acknowledgement is sent on reception of a command from the PGF Also sent after *Launch* with value Start.

Cmd_TM { *Command_Group* ACK *to differentiate this TM from a stage TM*
Command_type Start, *sent on PPF launch as the first exchange of the run*
Step, Break, Stop,
Suspend, Resume
Command_Id Integer *a reference to the original command*
Command_Status Received (coded '0') or Rejected ('2') }

Under a full ASCII format, all fields of variable size separated by white spaces and ending by a newline, e g

```
ACK STEP 137 0
ACK SUSPEND 1815 0
```

Command completion is sent on completion of a command It uses the same format than the previous one with different parameter values

Cmd_TM { *Command_Group* ACK *to differentiate this TM from a stage TM*
Command_type Start, *sent just before shut down of the PPF*
Step, Break, Stop,

Suspend, Resume *from the original command*

Command_Id Integer *a reference to the original command*

Command_Status Completed (coded '1'), *on normal processing condition*
Aborted ('3') *on fatal error (including interruption by a Stop or a Break)*

}

For Suspend, Resume, the completion occurs in a short delay (no other TM will interleave,
For Break and Stop, the completion occurs when processing has stopped
For Step, the completion corresponds to the end of the processing step
For Start, the completion occurs on shutdown of the run

Under a full ASCII format, all fields of variable size separated by white spaces and ending by a newline, e g. :

```
ACK STEP 137 1
ACK SUSPEND 1815 1
```

File delivery TM are asynchroneously sent for each file delivered

Stage_TM { *Command_Group* STAGE *to differentiate this TM from a command-TM*
stage_Id String *to be set to L1a, L1b, or any other meaningful text*
Command_Id Integer *a reference to the original Step command*
category Product ('P'), Report ('R'), Aux file ('W'), No file ('N')
filename String *the name of the delivered file relatively to the working root directory, possibly blank if category is 'N'* }

Under a full ASCII format, all fields of variable size separated by white spaces and ending by a newline, e g :

```
STAGE L1a 137 P unproc/L1/IASI_L1_011010196_0110100207
ACK SUSPEND 1815 0
```

3.1.13 commands & TMs sequencing and nesting

The sequencing of commands, acknowledgements and stage TM must comply to rules described below.

1. Each received command triggers 2 TMs an acknowledge TM and a completion TM

2 The acknowledgement is the first response given by the PPF to a command

3 Commands must be seen as opening brackets while the corresponding command Completion are the closing brackets According to this convention, brackets must be nested without interleaving closing happen in the reverse order than opening

4 File deliveries (Stage TM) shall occur inside a step after the Step-ack TM and before the Step-completion TM

5 On reception of a command Suspend, Resume, No command is read and no TM sent before ack TM and completion TM are sent

6 On reception of a command Break or Stop, No further command is read File deliveries may occur before Break or Stop completion is sent Step completion must occur after Break or Stop completion

7 acknowledgement of command Start (Start is actually the launch) is the first TM sent by a PPF It is actually the first exchange between PPF and PGF PGF waits for this ack to send its first command

8 Completion of command Start (Start is actually the launch) is the last TM sent by the PPF It follows a step completion TM

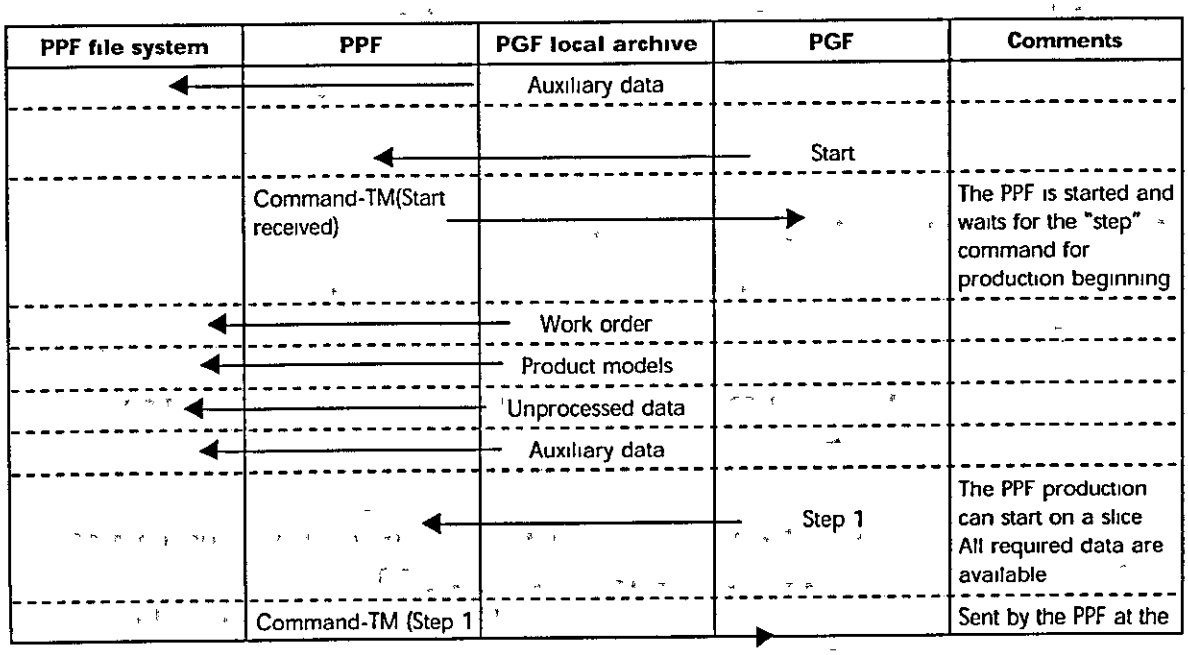
9 When a command is acknowledged as rejected by the PPF, no command completion is sent

An example of commands and TM sequencing is given in section Event Trace Diagram

3.1.14 IASI1/PGF Event trace diagram

In the followings figures the STEP command are numbered to make the figures reading easier It is then warned that the attached number is not the Command_Id of the command syntax

3 1 14 1 Nominal case



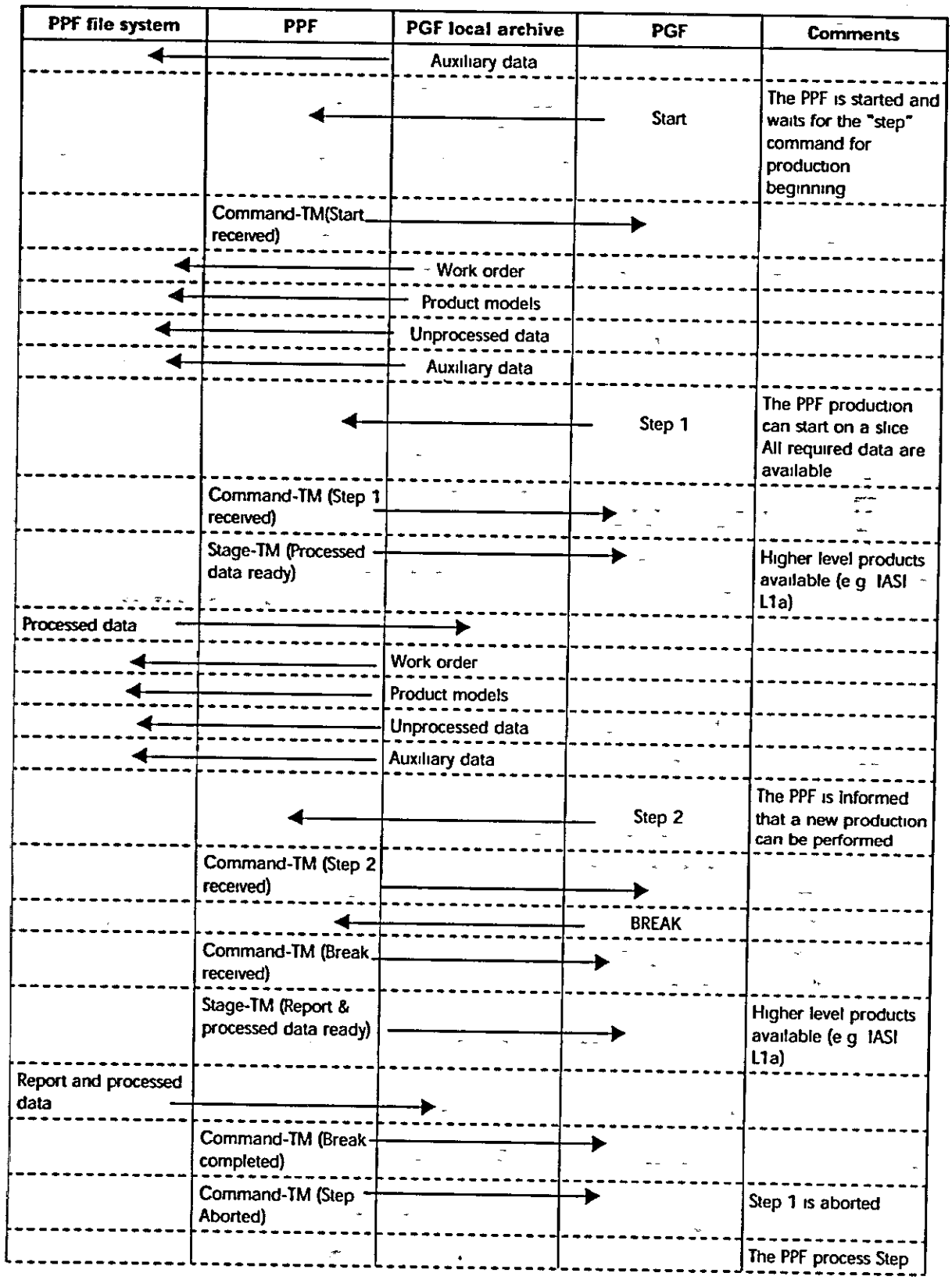
PPF file system	PPF	PGF local archive	PGF	Comments
	received)			completion of its initialisation phase
	Stage-TM (Processed data ready)	→	→	Higher level products available (e g IASI L1a)
Processed data →		→		
	←	Work order		
	←	Product models		
	←	Unprocessed data		
	←	Auxiliary data		
	←		Step 2	The PPF is informed that a new production can be performed Note that Step command can be sent by PGF before or after the reception of the « completed » Command-TM of the previous step
	Command-TM (Step 2 received)	→	→	
	Stage-TM (Report & processed data ready)	→	→	Higher level products available (e g IASI L1c)
Report and processed data →		→		
	Command-TM · (Step 1 completed)	→	→	To be sent before beginning the processing of the second step
	Stage TM	→	→	The PPF start production on the next slice (Step 2)
	Command-TM ("Start" completed)	→	→	The start command is completed

3 1.14 2 SUSPEND/RESUME case

PPF file system	PPF	PGF local archive	PGF	Comments
	←	Auxiliary data		
	←		Start	The PPF is started and waits for the "step" command for production beginning

PPF file system	PPF	PGF local archive	PGF	Comments
	Command-TM(Start received)	→		
		← Work order		
		← Product models		
		← Unprocessed data		
		← Auxiliary data		
			← Step 1	The PPF production can start on a slice All required data are available
	Command-TM (Step 1 received)	→		
	Stage-TM (Processed data ready)	→		Higher level products available (e g IASI L1a)
Processed data →				
		← Work order		
		← Product models		
		← Unprocessed data		
		← Auxiliary data		
			← Step 2	The PPF is informed that a new production can be performed
	Command-TM (Step 2 received)	→		
			← Suspend	
	Command-TM (Suspend received)	→		
	Command-TM (Suspend completed)	→		
			← Resume	
	Command-TM (Resume received)	→		
	Command-TM (Resume completed)	→		
	Stage-TM (Report & processed data ready)	→		Higher level products available (e g IASI L1c)
Report and processed data →				
	Command-TM (Step 1 completed)	→		
	Command-TM ("Start" completed)	→		The start command is completed

3.1.14 3 BREAK case



PPF file system	PPF	PGF local archive	PGF	Comments
				2
	Command-TM (Step Completed) for Step 2	→		
	Command-TM ("Start" completed)	→		The start command is completed

3 1.14 4 STOP case

PPF file system	PPF	PGF local archive	PGF	Comments
		← Auxiliary data		
			← Start	The PPF is started and waits for the "step" command for production beginning
	Command-TM(Start received)	→		
		← Work order		
		← Product models		
		← Unprocessed data		
		← Auxiliary data		
			← Step 1	The PPF production can start on a slice All required data are available
	Command-TM (Step 1 received)	→		
	Stage-TM (Processed data ready)	→		Higher level products available (e g IASI L1a)
Processed data	→			
		← Work order		
		← Product models		
		← Unprocessed data		
		← Auxiliary data		
			← Step 2	The PPF is informed that a new production can be performed
	Command-TM (Step 2 received)	→		
			← STOP	
	Command-TM (Stop received)	→		
	Stage-TM (Report & processed data ready)	→		Higher level products available (e g IASI

PPF file system	PPF	PGF local archive	PGF	Comments
				L1a)
Report and processed data		→		
	Command-TM (Stop completed)	→	→	
	Command-TM (Step Aborted)	→	→	Step 1 is aborted
	Command-TM (Step Aborted)	→	→	Step 2 is aborted
	Command-TM ("Start" aborted)	→	→	The start command is aborted

3.1.14 5 Non nominal case : the command cannot be processed

Error case	Acknowledgement
The command_type is neither START, STEP, STOP, SUSPEND, RESUME, or BREAK	ACK Command_type Command_id 2
Start command : The working root directory is not found The running mode is neither « N » nor « I » A mandatory configuration file is not found	 ACK START Command_id 2 ACK START Command_id 2 ACK START Command_id 2
Step /Break/Suspend/Resume/Stop command	
The command identifier is not the one of the START command	ACK Command_type Command_id 2
Step command The step command is received while a SUSPEND is currently active	 ACK STEP Command_id 2
The work order is not found A mandatory file is not found (unprocessed data, auxiliary data)	ACK STEP Command_id 2 ACK STEP Command_id 2
The initialisation phase of the PPF has to be aborted	ACK STEP Command_id 2 (followed by ACK START Command_id 3)
Break command The break command is received while a SUSPEND is currently active	 ACK STEP Command_id 2
Break/Suspend command The command is received while there is no STEP command currently running	 ACK Command_type Command_id 2
Resume command The command is received while there is no SUSPEND command previously received	 ACK RESUME Command_id 2

When a START is rejected, the PPF is then not started, and will then not accept any other command

When a STEP command is rejected :

- For a PPF in dynamic mode, the PPF stops then and send "ACK START 3" ,
- For a PPF in static mode, the PPF will wait for the next STEP

3.2 PPF TO MCS INTERFACE

Relationship between the PPF sofware and the MCS is encapsulated in an API (MCS Local Agent API) The general requirements of those mechanisms are given in [AD8], section 4 Among them, it is worth recalling that the MCS does not send any command nor information to the PPF (communication is one-way)

3.2.1 PPF Registration

At the beginning of each run, a PPF initiates the communication with MCS and identify itself through 2 letters configurable (as a field in the config file of the PPF or, if no config file exists, directly in the startup script)
Several instances of the same PPF running concurrently use the same identifier Their hk-tm are differentiated on other parameters (mainly the start time of the product currently under generation)

Identification is done through a call to routine *mla_initialisation* of MLA library (1 call per process using the library)

`mla_Initialisation (id, 0),` with *id* = `AS` for Ascat
`A1` for Atovs1
`A2` for Atovs2
`GO` for Gome
`GR` for Gras
`I1` for Iasi1
`I2` for Iasi2
(nevertheless configurable)

the 2nd parameter must be kept to 0, value 1 is reserved for processes receiving commands from MCS

Following this initialisation, 3 levels of information shall be sent to MCS *HKTM (or statuses)* to say processing continues, *Event* to report asynchronous events, *Trace* to log information related to the investigation mode They are detailed below

3.2.2 LogEvent

As a general rule, logEvent is intended to report the same level of info than the report Three levels of severity are possible, they shall be selected in the following cases

logEvent(Alarm) if an intervention of the operator is expected, e g : *missing necessary aux data, system error, crash of the processing* This exclude any numerical instability due poor values of the input data (poor values in the input files is not an error but a special case of processing)

logEvent(Warning) reports an abnormal situation, e g :

example 1 *Not enough input data to satisfy the workorder request*
example 2: *an aux file validity starttime is posterior to the data to be processed.*

logEvent(Info) a few per granule *start+PPFversion, change of processing stage as L1a or L1b, completion of the current workorder.*
example. generation of an occasional aux data file *(ASCAT transponder echoes, GOME In-flight calibration, ..)*

They are sent through a call to routine logEvent of MLA library with parameter *Event* built as follow (in each column, the possible values are displayed)·

Event.eventName

Facility 2 CHARS	Location 2 CHARS	Severity 1 CHAR	Type 1 CHAR	Service 3 CHARS	message ID 3 CHARS
AS for Ascat **A1** for Atovs1 **A2** for Atovs2 **GO** for Gome **GR** for Gras **I1** for Iasi1 **I2** for Iasi2	**GS** *fixed*	**A** for Alarm **W** for Warning **I** for Info *According to description above*	**P** Product **A** Aux Dat **S** Soft	**M01** for Metop1 **M02** for Metop2 **M03** for Metop3 **N16** for Noaa16 **N17** for Noaa17 *For metop, 2nd char is digit zero (not letter 'O') the info is found in input product (L0 or L1b), record MPHR, field SPACECRAFT*	A number (for instance **010**, 020,030,040,) *Identify the event type one unique number for each textual description of field event eventText*

Event.eventText

Textual Information 68 CHARS (constant EVT_STR_LEN in MLA)
text describing the event

The list of events possibly sent by the PPF shall be documented in its Software User Manual
The different fields in eventName must be thought as keys for filtering these messages on a display

3.2.3 LogTrace

logTrace accepts whatever the programmer needs to log for *remote* debug
It is intended to be the output for the mode Investigation described in the PPF FRS
The volume of data logged shall not slow down the processing by more than a factor 2 (for instance, it shall be avoided to log 1 message per pixel)
This trace must be thought as a "remote debug" first help, in case an anomaly is detected on site It is not intended to replace other debug means during development

In mode Nominal, it is reduced to a minimum

Traces are sent through a call to routine logTrace of library MLA with parameter *Trace* built as follow (in each column, the possible values are displayed)

Trace.traceName

Facility 2 CHARS	Location 2 CHARS	Severity 1 CHAR	Type 1 CHAR	Service 3 CHARS	message ID 3 CHARS
Idem logEvent	*Idem logEvent*	`I` for Info	`S` for Soft	*Idem logEvent*	*Idem logEvent*

Trace.traceText

Textual Information 68 CHARS (constant TRA_STR_LEN in MLA)
text

Trace text shall include, at relevant points, the "source file name" and "line number" This info is not requested for each trace message but rather for a set of related outputs

3.2.4 PPF CGS HK-TM (status)

Such messages are sent to MCS on a regular basis and on some events (start, stop) They convey some variables intended to report of the instantaneous status of the processing (actually an average on the processing performed since previous status)

They follow the rules described in section 2 1 3 and are sent through a call to routine mla_SendFunctionStatus of the MLA library

A separate call shall be made for each value (int, double, string) to be sent However, sendings are made per batch (1 call to send a sensing time followed by 1 call to send a quality summary, followed by the start time) The different calls compounding a single batch shall have the same *time stamping* (unix time of the local host just before the sending)

The format of each single call is described below

timeParam

Shall contain the *time stamping* of the batch of sendings
It is determined through a system call yielding the unix time of the local host, just some instructions prior to the sendings

telem.paramName

Facility 2 CHARS	Category 1 CHAR	Location 2 CHARS	Type 1 CHAR	Service 3 CHARS	message ID 3 CHARS
`AS, A1` – *(Idem logEvent)*	**O** operᵃᵗ **I** interface **R** resource **P** performᵃⁿᶜᵉ **S** status	`GS` *(Idem logEvent)*	`P` Product `A` Aux Dat `S` Soft `R` Resourᶜᵉ	`M01` – *(Idem logEvent)*	A number allowing to distinguish 2 TMs whose 5 previous fields are identical Moreover, forTMs of category S, the last 2 chars of field ID are used to represent the type of file under generation

telem.validity

Nominally **set to 1.** Value 0 can be used in case no meanigful data are available (a value 0 indicates to MCS that the next field (*valueString*) is not filled).

telem.valueString

To be encoded according to the MLA user manual, depending on the type of data sent

3.2.4.1 Predefined status

The following HK-TMs constitute the pattern of a batch of status Such a batch of HK-TM is sent repeatedly at a spacing of 15s to 60s and after completion of a step

.paramName						valueString	
Facility	Categ	Location	Type	Service	ID	~36 CHARs	
AS	**P**	GS	**S**	M01	**010**	***PPF version*** **String**	Constant during the run
AS	**O**	GS	**S**	M01	**010**	***Readiness*** **Integer 1, 2, 3, 4**	1 for Idle (e g PPF waiting for a command) 2 for Working (the nominal value) 3 for recoverable error (use is optionnal) 4 for Unrecoverable error
AS	**S**	GS	**P** *or* **A**	M01	**1xx**	***Start_time*** **YYYYMMddHHmmssZ**	Sensing start time of the products being generated This value is usually slightly anterior to field start_time of the workorder
AS	**S**	GS	**P** *or* **A**	M01	**2xx**	***Current sensing time*** **YYYYMMddHHmmssZ**	Sensing time of the latest mearurement processed
AS	**S**	GS	**P** *or* **A**	M01	**3xx**	***Instantaneous quality*** **integer in 0 100**	Computed as the percentage of properly generated records among the records processed since last status sending Value 100 correspond to a nominal processing Value 0 correspond to the extreme case where no valid measurement has been computed since previous status e g in case of a data gap
AS	**S**	GS	**P** *or* **A**	M01	**4xx**	***Start orbit number*** **integer**	Found in MPHR of L0, L1a or L1b taken as input 0 in case no L0, L1a nor L1b are taken as input (in some special calibration stages where the input are accumulated aux data)

Facility	Categ	Location	Type	Service	ID	36 CHARs	
AS	S	GS	P or A	M01	5xx	OK or FAILURE	FAILURE is reported as soon as the requested workorder cannot be fulfilled e g. when a STOP command has interrupted the processing, when necessary aux data are missing or when not enough L0 is supplied OK is the nominal value OK is still used in case of numerical problems e g when measurements are out of range
AS	R	GS	R	M01	010	*Instantaneous cpu usage* integer in 0 ..100	Can be computed as user_time or elapsed_time or from other system calls It should be almost 1 when the host machine is dedicated to a single PPF

Values **P** or **A** in field *type* and **xx** in field *ID* are set the same value for all 5 HKTM of the batch They represent the product under generation with the coding given in the following table

Name	Description	Level	type	xx
IASI_xxx_1A_	IASI Level 1a product	1a	P	49
IASI_xxx_1B_	IASI Level 1b product	1b	P	59
IASI_xxx_1C_	IASI Level 1c product	1c	P	69
IASI_ENG_01_	IASI Engineering data product	1	P	89
IASI_VER_01_	IASI verification data product	1	P	99

3.2.5 PPF Roles to function allocation

Management of rights to access a PPF is obtained through standard Unix rights on files with no ensuing impact on the PPF delivery

The following table gathers the different predefined UID within the PPF environment, on the PPF host

facility	install	Run-time	MMI (user/role)	File interface
PPF IASI1	iasi_1mg	iasi_1	N/A	mla, pgf

<MCCuser> : personal login of one user

<external entities> according to external ISD

3.3 AUX DATA FILES

3.3.1 XML aux data file

Any ASCII aux file shall be in XML, its format being described as an XML DTDs in a separate file
Moreover, as for the workorder DTD, the PPF may require the DTD file at run time In this case, the PPF shall use a DTD file statically stored, as other config files, in the installation directory tree Actually, the file refered in the second line of an XML file, <!DOCTYPE > tag, may not be available

Each XML file shall include header fields for version and history as exemplified below

```
<FormatVersion>1 7</FormatVersion>   refers to the definition of the format as described in PPF SUM
<ContentVersion>1 7 2draft</ContentVersion>   refers to the values of each field
<History>   a summary of past modifications of content
20 Jan 2003, Threshold for PRT counts have been increased
</History>
```

Then the body of the file is organised in sections, each section corresponding to an entry of the auxiliary data inventory document, e g :

```
<AVHRR_L1_PGS_COF_CAL_AVHRR>   a given entry of the aux data inventory
     <PRT_weighting_factors>0 351 0 3113 0 178 0 1597</PRT_weighting_factors>
     ...
</AVHRR_L1_PGS_COF_CAL_AVHRR>
<AVHRR_L1_PGS_COF_ISRF>   another entry of the aux data inventory
     ...
</AVHRR_L1_PGS_COF_ISRF>
...
```

These section tags allows to minimize the number of configuration files (roughly 1 per instrument)

The exact format of such a file is given in the corresponding PPF user manual (SUM)

IASI Auxiliary files list

The auxiliary files identified for IASI L1 PPF are

File name	description	type	In/Out	scope
AVHR_xxx_1B_	AVHRR level 1b product compliant to AVHRR PFS document	product	in	disseminated
xxxx_OBT_xx_	OBT-UTC correlation factors (if the need is	ASCII/X	in	Internal CGS

File name	description	type	In/Out	scope
	confirmed by CNES)	ML		FDF/PGF /PPF
xxxx_OSV_xx_	Metop orbit predictions a sequence of OSV at consecutive times includes predicted manoeuvres	ASCII/X ML	in	Internal CGS FDF/PGF /PPF
xxxx_SVM_xx_	Predicted geometric events Includes the events start and end of umbra/penumbra	ASCII/X ML	in	Internal CGS FDF/PGF /PPF
IASI_CTX_xx	IASI Context file Historical account of interferometric axis Recursive data for filters and degraded modes	binary	in/out	internal PPF/PGF

Note:

IASI L1 PPF also uses parameters from several configuration files
(a configuration file differs from an aux file by not being managed by PGF but copied, on software installation, along with executable file)

The following 3 configuration files are defined

Config File name	description	type	In/Out	scope
IASI_GRD_xx	IASI L1 ground configuration file	binary	in	IASI Tech
IASI_BRD_xx	IASI L1 board/ground configuration file	binary	in	IASI Tech
IASI_SDB_xx	IASI spectral data base Calibration functions for spectra Apodisation functions	binary	in	IASI Tech

xxxx_OSV_xx:

for illustration, a sample of OSV auxdata file is given below

```
<?xml version="1.0"?>
<!DOCTYPE FD_Metop_NOAA_Orbit_Prediction SYSTEM
"FD_Metop_NOAA_Orbit_Prediction.dtd">

<FD_Metop_NOAA_Orbit_Prediction>
 <ELEMENT ID="M01">
  <ORBIT_ELEMENT_TYPE>MEAN</ORBIT_ELEMENT_TYPE>
  <ORBIT_REFERENCE_FRAME>4</ORBIT_REFERENCE_FRAME>
  <INTERPOLATION_COEFFICIENT>
    <VALUE_COEFFICIENTS>1564</VALUE_COEFFICIENTS>
    <VALUE_COEFFICIENTS>34764</VALUE_COEFFICIENTS>
    <VALUE_COEFFICIENTS>76432.3</VALUE_COEFFICIENTS>
  </INTERPOLATION_COEFFICIENT>
  <ORBIT_START_ID="1001">
      <FLAG>ANX</FLAG>
      <TIME>01011:0057898000:00000</TIME>
      <SM_AXIS>6378.137</SM_AXIS>
      <ECCENTRICITY>0.003353</ECCENTRICITY>
      <INCLINATION>1357.900000</INCLINATION>
      <PERIGEE>12357.110000</PERIGEE>
```



```
      <RT_ASCENSION>1317.190000</RT_ASCENSION>
      <MEAN_ANOMALY>2329.310000</MEAN_ANOMALY>
      <X>6906.812212</X>
      <Y>2056.232193</Y>
      <Z>-0.000000</Z>
      <X_VELOCITY>0.462905</X_VELOCITY>
      <Y_VELOCITY>-1.585657</Y_VELOCITY>
      <Z_VELOCITY>7.353034</Z_VELOCITY>
    </ORBIT_START>
  </ELEMENT>
</FD_Metop_NOAA_Orbit_Prediction>
```

Note that the associated .dtd file mentionned in <!DOCTYPE> tag (line 2) will not be delivered The DTD stored in config shall be used by the PPF.

4. LINK TYPES

A link type is associated to each interface, it specifies

- The data *source* for the interface
- The data *sink* for the interface

Each link type is "instantiated" with regards to the Ground Segment (G/S) where the facility is installed The identified Ground Segments are

- G/S 1 (Ground Segment 1 in MCC)
- G/S 2 (Ground Segment 2 in MCC)

This section presents all "link-type" used for each interface

Link type : PGF/PPF_(In_GS)

PGF	[GS_1]	=>	PPF	[GS_1]
PGF	[GS_2]	=>	PPF	[GS_2]

4.1 LINK TYPE : PPF/MCS_(IN_GS)

PPF	[GS_1]	=>	MCS	[GS_1]
PPF	[GS_2]	=>	MCS	[GS_2]

4.2 LINK TYPE : PPF/PGF_(IN_GS)

PPF	[GS_1]	=>	PGF	[GS_1]
PPF	[GS_2]	=>	PGF	[GS_2]

THIS PAGE IS INTENTIONALLY LEFT BLANK

5. INTERFACE DEFINITIONS

Specific definitions

[1]The purpose of the "Interface definition" paragraph is to provide a functional description of the flows related to the document with the following fields :

- **Name :** interface name which is a unique identifier
- **Interface number :** unique identifier
- **Summary :** brief summary of the interface
- **History :** creation/changes/deletion of the interface
- **Link type :** identification of the functional link (source – destination), relevant for CGS reader only
- **Data structure :** name of the structure attached to the interface and described in the next section
- **Interface Type :** characterisation of the flow type (file, byte stream) which has to be compliant with the Support service definition
- **Encoding :** specifies the type of encoding related to a file (ASCII, binary, mixed, XML)
- **Support description :** description of the physical support category of the interface, the support type can be defined amongst (Internet, Eumetsat LAN, Dissemination Network, media...)
- **Satellite :** identification of a dedicated spacecraft when relevant
- **Use/Description :** use and description of the interface
- **Support service :** specifies the mechanism which will be used to exchange data for this interface, the mechanism can be a well known protocol (e g TCP/IP) or a bespoke protocol defined for the CGS (e g Generic File Transfer)
- **Document reference :** identification of the document(s) leading to this interface definition

[1] DDT_v6 13 0.28-3 3e-07/03/03

Interface Name: PPF_Auxiliary_Data

Interface number. IF0346

Summary PPF Auxiliary Data

History
12/11/02 15 13 no more ComFunction definition OR DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
08/02/02 10 13 MeanFrequency=as needed
08/02/02 10 13 Description=modified
30/01/02 10 37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
15/01/02 10 06 number
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
20/12/01 10 05 Support_service=PUSH mechanism
20/12/01 10 04 Satellite=NOAA and METOP
20/12/01 10 03 ComFunction=Aux Data Reception CF
20/12/01 10 02 Support_service=Configurable
20/12/01 10 01 SupportDescription=PGF LAN
30/09/99 10 00 Creation

Link Type: PGF/PPF_(In_GS)

Data Structure eps_aux_data

Interface Type File

Encoding mixed

Support description: PGF LAN

Mean frequency: as needed

Satellite: NOAA and METOP

Use_Description

Auxiliary dynamic and static Data provided by the PGF to the PPF for the product processing task
Auxiliary data files are transferred periodically by PGF to PPF file system

Support service: PUSH mechanism

Document Reference:

5.1 INTERFACE NAME: PPF_CMD_TM

Interface number: IF0347

Summary: PPF TM in response to command (status)

History:
12/11/02 15 13 no more ComFunction definition OR-DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10:37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
20/12/01 10:05 Type=Software
20/12/01 10:04 Satellite=N/A
20/12/01 10:03 Satellite=NOAA and METOP
20/12/01 10:02 SupportDescription=PGF LAN
20/12/01 10:01 SupportDescription=N/A
30/09/99 10 00 Creation

Link Type: PPF/PGF_(In_GS)

Data Structure ppf command status

Interface Type: Software

Encoding: ASCII

Support description: PGF LAN

Mean frequency: as needed

Use_Description:

TM used to indicate the acknowledgement of a PGF command to a PPF

Support service: Stdin Stdout (Mechanism)

Document Reference:

5.2 INTERFACE NAME: PPF_COMMAND

Interface number: IF0348

Summary PPF Command

History
12/11/02 15 13 no more ComFunction definition OR-DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10 37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
20/12/01 10 03 Satellite=N/A
20/12/01 10 02 Satellite=NOAA and METOP
20/12/01 10 01 SupportDescription=PGF LAN
30/09/99 10 00 Creation

Link Type: PGF/PPF_(In_GS)

Data Structure ppf command

Interface Type: Software

Encoding: ASCII

Support description PGF LAN

Mean frequency. per granule duration

Use_Description:

The PPF receives from PGF commands which are either START /ABORT, which are signals handled by the PPF, or STOP, SUSPEND, RESUME which are messages handled by the PPF The START specifies the mode for the PPF execution either NOMINAL or INVESTIGATION

Support service Stdin Stdout (Mechanism)

Document Reference:

5.3 INTERFACE NAME: PPF_CONFIGURED_ITEMS

Interface number: IF0349

Summary: PPF Configured items

History:
12/11/02 15 13 no more ComFunction definition OR-DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10 37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
20/12/01 10 04 Satellite=N/A
20/12/01 10 03 SupportDescription=PGF LAN
20/12/01 10 02 ComFunction=Monitoring and Control CF
20/12/01 10 01 SupportDescription=N/A
30/09/99 10 00 Creation

Link Type: PGF/PPF_(In_GS)

Data Structure: operational_type

Interface Type: Software

Encoding: mixed

Support description: PGF LAN

Use_Description:

This interface is used to supply PPF with new executable S/W algorithm and static tables PGF receives an instruction from MCS to install a new version of PPF First, PGF installs a new release only when PPF is not running Installation is transparent for PPF and is made on the PPF local disk
Then PGF is responsible for switching from the old version to the new one Finally PGF is responsible for deleting old SW version on the PPF local disk

Support service: Installation

Document Reference:

5.4 INTERFACE NAME: PPF_HKTM

Interface number: IF0350

Summary PPF HouseKeeping TeleMetry

History
12/11/02 15 13 no more ComFunction definition OR-DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
08/02/02 10 15 MeanFrequency=CGS TM update rate (1mn)
08/02/02 10 14 MeanFrequency=CGS TM update rate (1')
24/01/02 10 02 SupportDescription
24/01/02 10 01 Support_service=Facility M&C
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
30/09/99 10 00 Creation

Link Type PPF/MCS_(In_GS)

Data Structure t_facility_hktm

Interface Type Message

Encoding: ASCII

Support description

Mean frequency CGS TM update rate (1mn)

Satellite METOP

Use_Description
Current function status

Support service Facility M&C

Document Reference

5.5 INTERFACE NAME: PPF_PRODUCT_MODELS

Interface number: IF0351

Summary: PPF Product Models

History:
12/11/02 15 13 no more ComFunction definition OR DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10:37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
20/12/01 10:05 ComFunction=Monitoring and Control CF
20/12/01 10:04 Support_service=PUSH mechanism
20/12/01 10:03 SupportDescription=PGF LAN
20/12/01 10:02 Satellite=NOAA and METOP
20/12/01 10:01 ComFunction=N/A
30/09/99 10:00 Creation

Link Type: PGF/PPF_(In_GS)

Data Structure eps_product

Interface Type: File

Encoding: mixed

Support description: PGF LAN

Mean frequency: per granule duration

Satellite: NOAA and METOP

Use_Description:

Product models are provided by PGF to PPF There is a model for each type of product to be generated It contains the MPH partially pre-filled by PGF with well-known information So PPF has to take the structure and fill the empty fields with relevant information

Support service: PUSH mechanism

Document Reference:

5.6 INTERFACE NAME: PPF_REPORT

Interface number: IF0352

Summary. PPF report

History
12/11/02 15 13 no more ComFunction definition OR DDT-0102
14/05/02 20 00 -CDR-PPF_GRAS- MileStone for facility(PPF) MileStone
13/05/02 18 45 -CDR-PPF_IASI_L2- MileStone for facility(PPF) MileStone
30/01/02 10 37 -CDR-PGF- MileStone for facility(PGF) MileStone
04/01/02 15 25 -CDR-PPF_ASCAT- MileStone for facility(PPF) MileStone
04/01/02 15 25 -CDR-PPF_GOME- MileStone for facility(PPF) MileStone
04/01/02 15 25 -CDR-PPF_ATOVS- MileStone for facility(PPF) MileStone
20/12/01 10 03 Satellite=NOAA and METOP
20/12/01 10 02 Support_service=GET mechanism
20/12/01 10:01 SupportDescription=PGF LAN
30/09/99 10 00 Creation

Link Type PPF/PGF_(In_GS)

Data Structure t_ppf_report

Interface Type File

Encoding XML

Support description PGF LAN

Mean frequency per granule duration

Satellite. NOAA and METOP

Use_Description

Command execution report generated at the end of processing, and which details it

Support service GET mechanism

Document Reference

5.7 INTERFACE NAME: PPF_STAGE_TM

Interface number: IF0353

Summary: PPF stage TM (status)

History:
12/11/02 15 13 no more ComFunction definition OR-DDT-0102
14/05/02 20:00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10 37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
20/12/01 10:02 Satellite=NOAA and METOP
20/12/01 10:01 SupportDescription=PGF LAN
30/09/99 10:00 Creation

Link Type: PPF/PGF_(In_GS)

Data Structure ppf stage TM

Interface Type: Software

Encoding: ASCII

Support description: PGF LAN

Mean frequency: as needed

Satellite: NOAA and METOP

Use_Description:

The stage status is sent by PPF to PGF asynchronously whenever there is a change of the processing state, and when an algorithm processing stage is achieved

Support service: Stdin Stdout (Mechanism)

Document Reference:

5.8 INTERFACE NAME: PPF_WARMSTART_FILE

Interface number: IF0354

Summary PPF warmstart file

History
12/11/02 15 13 no more ComFunction definition OR-DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10 37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
20/12/01 10 07 Encoding=mixed
20/12/01 10 06 ComFunction=Aux Data Reception CF
20/12/01 10 05 Satellite=NOAA and METOP
20/12/01 10 04 ComFunction=Monitoring and Control CF
20/12/01 10 03 Support_service=GET mechanism
20/12/01 10 02 SupportDescription=PGF LAN
20/12/01 10 01 ComFunction=Aux Data Reception CF
08/03/01 10 00 Creation

Link Type PPF/PGF_(In_GS)

Interface Type: File

Encoding mixed

Support description PGF LAN

Mean frequency per granule duration

Satellite: NOAA and METOP

Use_Description

This file contains a set of parameters regularly re-estimated to best estimate the status of the instrument at a given moment in time These parameters are stored by the PPF and sent to PGF at the end of granule production, so they can be loaded in case of NRT processing backlog processing and reprocessing

Support service. GET mechanism

Document Reference

5.9 INTERFACE NAME: PPF_WORK_ORDER

Interface number: IF0355

Summary: PPF Work Order file

History:
12/11/02 15 13 no more ComFunction definition OR DDT-0102
14/05/02 20 00 –CDR–PPF_GRAS– MileStone for facility(PPF) MileStone
13/05/02 18 45 –CDR–PPF_IASI_L2– MileStone for facility(PPF) MileStone
30/01/02 10 37 –CDR–PGF– MileStone for facility(PGF) MileStone
04/01/02 15 25 –CDR–PPF_ASCAT– MileStone for facility(PPF) MileStone
04/01/02 15 25 –CDR–PPF_GOME– MileStone for facility(PPF) MileStone
04/01/02 15 25 –CDR–PPF_ATOVS– MileStone for facility(PPF) MileStone
20/12/01 10 07 Satellite=NOAA and METOP
20/12/01 10 06 Support_service=PUSH mechanism
20/12/01 10 05 SupportDescription=PGF LAN
20/12/01 10 04 ComFunction=Monitoring and Control CF
20/12/01 10 03 Support_service=N/A
20/12/01 10 02 SupportDescription=N/A
08/08/01 10 01 change use description
30/09/99 10 00 Creation

Link Type: PGF/PPF_(In_GS)

Data Structure t_work_order

Interface Type: File

Encoding: XML

Support description: PGF LAN

Mean frequency: per granule duration

Satellite: NOAA and METOP

Use_Description:

The Work Order file is used to transfer information needed for processing It contains all information needed by PPF.

Support service: PUSH mechanism

Document Reference:

5.10 INTERFACE NAME: PROCESSED PPF DATA SETS

Interface number IF0356

Summary Processed PPF data sets

History
12/11/02 15 13 no more ComFunction definition OR DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10 37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
20/12/01 10 05 Support_service=GET mechanism
20/12/01 10 04 Satellite=NOAA and METOP
20/12/01 10 03 ComFunction=Data Acquisition CF
20/12/01 10 02 Support_service=PUSH mechanism
20/12/01 10 01 SupportDescription=PGF LAN
30/09/99 10 00 Creation

Link Type PPF/PGF_(In_GS)

Data Structure eps_product

Interface Type Product File (PDUs or full product)

Encoding mixed

Support description: PGF LAN

Satellite NOAA and METOP

Use_Description.

Products generated by the PPF

Support service. GET mechanism

Document Reference

5.11 INTERFACE NAME: UNPROCESSED PPF DATA SETS

Interface number: IF0357

Summary: Unprocessed PPF data sets

History:
12/11/02 15 13 no more ComFunction definition OR-DDT-0102
14/05/02 20 00 ~CDR~PPF_GRAS~ MileStone for facility(PPF) MileStone
13/05/02 18 45 ~CDR~PPF_IASI_L2~ MileStone for facility(PPF) MileStone
30/01/02 10 37 ~CDR~PGF~ MileStone for facility(PGF) MileStone
04/01/02 15 25 ~CDR~PPF_ATOVS~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_ASCAT~ MileStone for facility(PPF) MileStone
04/01/02 15 25 ~CDR~PPF_GOME~ MileStone for facility(PPF) MileStone
20/12/01 10 08 Satellite=NOAA and METOP
20/12/01 10 07 Type=Product File (PDUs or full product)
20/12/01 10 06 ComFunction=Data Acquisition CF
20/12/01 10 05 Support_service=PUSH mechanism
20/12/01 10 04 Encoding=mixed
20/12/01 10 03 Type=File
20/12/01 10 02 SupportDescription=PGF LAN
08/08/01 10 01 change support service
30/09/99 10 00 Creation

Link Type: PGF/PPF_(In_GS)

Data Structure: eps_product

Interface Type: Product File (PDUs or full product)

Encoding: mixed

Support description: PGF LAN

Mean frequency: per granule duration

Satellite: NOAA and METOP

Use_Description:

Science Data (from Level 0 to Level 1) granules provided by the PGF to the PPF for the product processing task

Support service: PUSH mechanism

Document Reference:

THIS PAGE IS INTENTIONALLY LEFT BLANK

6. TYPE DEFINITIONS

PPF_IASI_L1

Type : eps_aux_data

Structure level: 0

Structure layout of eps_aux_data

Refer to "EPS Auxiliary data Inventory" AD-65 for the format definition of the auxiliary data

6.1 TYPE : EPS_PRODUCT

6.1.1 Structure level: 0

Structure layout of eps_product

Refer to PFS document

6.2 TYPE : OPERATIONAL_TYPE

6.2.1 Structure level: 0

Structure layout of operational_type

Related to interfaces involving an operator (not automatically handled by a facility)

No structure definition needed in this document.

6.3 TYPE : PPF STAGE TM

6.3.1 Structure level: 0

Structure layout of ppf stage TM

Argument	Comments	Type	Rank
Command group	'STAGE for stage TM status	stdin/stdout argument line	1st
Proccessing Stage Identifier	this is an identification of the processing stage of the PPF, i e Initialisation, preprocessing etc	stdin/stdout argument line	2nd
Command identifier	1 to 5 characters)] which is linked to the command identification received through MCS commands	stdin/stdout argument line	3rd
Type of file generated	N for no file generated R for report W for warm-start file or P for processed data	stdin/stdout argument line	4nd
file name generated	filename of the generated file relatively to the working root directory	stdin/stdout argument line	5nd

6.3.2 Structure level: 1

Structure layout of stdin/stdout argument line

Argument in a line via stdin or stdout mechanism

6.4 TYPE : PPF COMMAND

6.4.1 Structure level: 0

Structure layout of ppf command

Name	Comments	Type	
Start command	it starts the PPF Main Process or script file	PGF/PPF Command Start	-
Step command	it starts the processing of the incoming slice or PDU When PPF is running it can receive one or several step command (i e dynamic PPF receive only one step command while static PPF receive several step commands all along the dump for each slice to produce)	PGF/PPF Command Step	-
Suspend command	pauses the current processing of PPF The suspend command may be received at any time after the PPF has been started The SUSPEND has to be performed in less than a few seconds from the reception of the command When the command is received the current processing stage is completed but subsequent stages are not started When in the suspend state the PPF programs remains active and maintains ownership of the node resources	PGF/PPF Command Suspend	-
Resume command	resumes current processing After a Suspend command when the Resume command is received PPF continues its processing by starting the next processing stage	PGF/PPF Command Resume	-
Break command	BREAK is processed as a STOP for all PPFs except IASI L1 For IASI L1 - if the BREAK is received for the last or the unique STEP of the dump it is handled as a STOP - else it leads to abort the current processing the processing will resume with the next command	PGF/PPF Command Break	-
Stop command	it is a soft shutdown The Main Process (MP) shall respond with a cleanup task PPF shall terminate its own processes and threads properly before	PGF/PPF Command Stop	-
Abort command	aborts a current processing The Main Process (MP) and its children will be forced to terminate by a S/W signal sent by PGF	PGF/PPF Command Abort	-

6.4.2 Structure level: 1

Structure layout of PGF/PPF Command Abort

UNIX SIGNAL (KILL)

Structure layout of PGF/PPF Command Break

Argument	Comments	Type	Rank
Command type	string of the command 'BREAK'	stdin/stdout argument line	1st
Command identifier	int (1 to 5 characters) which is linked to the command identification received through MCS commands	stdin/stdout argument line	2nd

Structure layout of PGF/PPF Command Resume

Argument	Comments	Type	Rank
Command type	string of the command 'RESUME'	stdin/stdout argument line	1st
Command identifier	int (1 to 5 characters)] which is linked to the command identification received through MCS commands	stdin/stdout argument line	2nd

Structure layout of PGF/PPF Command Start

Argument	Comments	Type	Rank
startup script file name	name of script or process to start	command line	0
command identifier	(1 to 5 characters)] which is linked to the command identification received through MCS commands,	command line	1st
working root directory	last character is /,	command line	2nd
running mode	N for Operational mode (nominal) PPF is activated with no traces I for Trace mode (investigation) similar to the operational mode, except PPF logs more information to the system log files This mode is intended for debugging purpose	command line	3rd

Structure layout of PGF/PPF Command Step

Argument	Comments	Type	Rank
Command type	string of the command 'STEP	stdin/stdout argument line	1st
Command identifier	int (1 to 5 characters)], which is linked to the command identification received through MCS commands,	stdin/stdout argument line	2nd
Work Order file name	relative or absolute path	stdin/stdout argument line	3rd

Structure layout of PGF/PPF Command Stop

Argument	Comments	Type	Rank
Command type	string of the command 'STOP'	stdin/stdout argument line	1st
Command identifier	int (1 to 5 characters) which is linked to the command identification received through MCS commands,	stdin/stdout argument line	2nd

Structure layout of PGF/PPF Command Suspend

Argument	Comments	Type	Rank
Command type	string of the command 'SUSPEND'	stdin/stdout argument line	1st
Command identifier	Int (1 to 5 characters)] which is linked to the command identification received through MCS commands.	stdin/stdout argument line	2nd

6.4.3 Structure level: 2

Structure layout of command line

command line : name of script or process to start

Structure layout of stdin/stdout argument line

Argument in a line via stdin or stdout mechanism

6.5 TYPE : PPF COMMAND STATUS

6.5.1 Structure level: 0

Structure layout of ppf command status

Name	Comments	Type	
command group	'ACK for command acknowledgement	stdin/stdout argument line	1st
command type	string of the command START' , 'STOP , 'STEP' , 'BREAK' , 'RESUME' or 'SUSPEND	stdin/stdout argument line	2nd
command identifier	Int (1 to 5 characters) which is linked to the command identification originally received by PGF through MCS commands	stdin/stdout argument line	3nd
command status	(0 for Received 1 for Completed 2 for Rejected or 3 for Aborted)	stdin/stdout argument line	4nd

6.5.2 Structure level: 1

Structure layout of stdin/stdout argument line

Argument in a line via stdin or stdout mechanism

6.6 TYPE : T_FACILITY_HKTM

6.6.1 Structure level: 0

Structure layout of t_facility_hktm **record**

Name	Description	Type	Dim
hktm_param_name	parameter name	t_hktm_param_name	1
hktm_status_level	status level	t_enum_status_level	1
hktm_string_value	string value	t_str_line	1

6.6.2 Structure level: 1

Structure layout of t_enum_status_level **enum**

HKTM status level unknown, full

Name	Description		Value
STATUS_LEVEL_UNKNOWN		=	0
STATUS_LEVEL_FULL		=	1

Structure layout of t_hktm_param_name **terminal**

PAR_NAM_LEN defined in local agent header file LA_api h

Format: %s
Type C or Dim: [PAR_NAM_LEN]

Structure layout of t_str_line **terminal**

Format: %s
Type C or Dim: [80]

6.7 TYPE : T_PPF_REPORT

6.7.1 Structure level: 0

Structure layout of t_ppf_report **record**

begin test

Name	Description	Type	Dim
WorkOrder		t_work_order	1
StageInfo		t_stage_info	*

6.7.2 Structure level: 1

Structure layout of t_stage_info **record**

Name	Description	Type	Dim
StageName		t_str_short_info	1
UsedFileName		t_xml_pcdata	*
GeneratedFile		t_generated_file	*
InstrumentTelemetry		t_xml_pcdata	?
ProcessingInfo		t_processing_info	1
LogMessage		t_str_text	1

Structure layout of t_work_order **record**

The work_order file is used to transfer additionnal parameters needed for PPF processing

This interface is a file

The work order file shall be created in ASCII (text file , not binary file)

See § 3 4 2 in doc EPS-ASPI-IR-0069

Name	Description	Type	Dim
ProcessingType	This field can be generated directly from PGF configuration file L00_1A Define processing from Level 0 to Level 1a L00_1B Define processing from Level 0 to Level 1b L1A_1B Define processing from Level 1a to Level 1b CAL Define a calibration processing, or Others	t_xml_pcdata	1
SensingStart	Coordinated Universal Time(UTC) appearing in "clear text", presented as a string of 15 significant characters with the following format YYYYMMDDHHMMSS TTTZ where TTT is optional **Format** YYYYMMDDHHMMSS[TTT]Z	t_xml_pcdata	1
SensingStop	Coordinated Universal Time(UTC) appearing in "clear text", presented as a string of 15 significant characters with the following format YYYYMMDDHHMMSS TTTZ where TTT is optional **Format** YYYYMMDDHHMMSS[TTT]Z	t_xml_pcdata	1
TimeIntervalFlag	String (from 4 to 6 characters)] FIRST This is the first time interval of the dump MIDDLE This is a normal time interval of the dump LAST This is the last time interval of the dump FULL This is a complete product, no slicing (i e L1 PDU for reprocessing))	t_xml_pcdata	1
TimeIntervalCounter	Counter [Integer (1 to 5 characters)] reset to zero at the beginning of each dump Several counters are incremented separately for each instrument type	t_xml_pcdata	1
UnProcData	The input unprocessed datafiles lis set including the relative or absolute path and the filenames of data (this can be the L0 slice filename or L1 PDU filename) Several unprocessed data can be declared into this section [List of String]	t_xml_pcdata	10

AuxData	Auxiliary data list including the relative or absolute path and the auxiliary data filenames and a flag requesting the PPF Relative path can be used when data are stored into from_pgf directory while Absolute path can be used when data are stored into global directory Logical filename is used for NRT, while complete filename is used for Reprocessing [List of String]	t_xml_pcdata	20
ProdModel	Output Product model filename, which corresponds to the output products to generate There is one product model per product type to be generated by the PPF [List of String]	t_xml_pcdata	10

6.7.3 Structure level: 2

Structure layout of t_generated_file — **record**

Name	Description	Type	Dim
FileName		t_xml_pcdata	1
MdrTotalCount	Number of MDRs in the current file (identical to field TOTAL_MDR of the MPHR) **Format** %lu **Unit** count	t_int_unsigned_four_bytes	?
MdrDegraded	**Format** %lu **Unit** count	t_int_unsigned_four_bytes	?
DataGap		t_data_gap	

Structure layout of t_processing_info — **record**

Name	Description	Type	Dim
ProcessingStart	String of 15 significant characters **Format** YYYYMMDDHHMMSS TTTZ	t_xml_pcdata	1
ProcessingStop	String of 15 significant characters **Format** YYYYMMDDHHMMSS TTTZ	t_xml_pcdata	1
ElapsedTime	**Format** %5 3f **Unit** seconds	t_float_duration	1
UserTime	**Format** %5 3f **Unit** seconds	t_float_duration	1

Structure layout of t_str_short_info — **terminal**

Format %s
Type C or Dim [40]

6.7.4 Structure level: 3

Structure layout of t_data_gap **record**

Name	Description	Type	Dim
GapReason		t_str_text	1
GapCount		t_int_unsigned_four_bytes	1

Structure layout of t_float_duration **typedef**

Unit: seconds
Format: % 3f

Name	Description	Type	Dim
t_float_duration		t_float_number	1

Structure layout of t_xml_pcdata **terminal**

```
<!ELEMENT t_xml_pcdata (#PCDATA)>

<!ATTLIST t_xml_pcdata

  format CDATA #FIXED '%s'>
```

Format: %s
Type C or Dim: char *

6.7.5 Structure level: 4

Structure layout of t_float_number **terminal**

The identifier "floatreal" has to be defined depending of the machine implementation

Type C or Dim floatreal

Structure layout of t_int_unsigned_four_bytes **terminal**

The identifier "doubleword" has to be defined depending of the machine implementation

Format: %lu
Type C or Dim doubleword

Structure layout of t_str_text **terminal**

Format. %s
Type C or Dim [255]

6.8 TYPE : T_WORK_ORDER

6.8.1 Structure level: 0

Structure layout of t_work_order **record**

The work_order file is used to transfer additionnal parameters needed for PPF processing

This interface is a file

The work order file shall be created in ASCII (text file , not binary file)

See § 3 4 2 in doc EPS-ASPI-IR-0069

Name	Description	Type	Dim
ProcessingType	This field can be generated directly from PGF configuration file L00_1A Define processing from Level 0 to Level 1a, L00_1B Define processing from Level 0 to Level 1b, L1A_1B Define processing from Level 1a to Level 1b, CAL Define a calibration processing or Others	t_xml_pcdata	1
SensingStart	Coordinated Universal Time(UTC) appearing in "clear text", presented as a string of 15 significant characters with the following format YYYYMMDDHHMMSS TTTZ where TTT is optional **Format** YYYYMMDDHHMMSS[TTT]Z	t_xml_pcdata	1
SensingStop	Coordinated Universal Time(UTC) appearing in "clear text", presented as a string of 15 significant characters with the following format YYYYMMDDHHMMSS TTTZ where TTT is optional **Format** YYYYMMDDHHMMSS[TTT]Z	t_xml_pcdata	1
TimeIntervalFlag	String (from 4 to 6 characters)] FIRST This is the first time interval of the dump MIDDLE This is a normal time interval of the dump LAST This is the last time interval of the dump FULL This is a complete product, no slicing (i e L1 PDU for reprocessing))	t_xml_pcdata	1
TimeIntervalCounter	Counter [Integer (1 to 5 characters)] reset to zero at the beginning of each dump Several counters are incremented separately for each instrument type	t_xml_pcdata	1
UnProcData	The input unprocessed datafiles list set including the relative or absolute path and the filenames of data (this can be the L0 slice filename or L1 PDU filename) Several unprocessed data can be declared into this section [List of String]	t_xml_pcdata	10
AuxData	Auxiliary data list including the relative or absolute path and the auxiliary data filenames and a flag requesting the PPF Relative path can be used when data are stored into from_pgf directory while Absolute path can be used when data are stored into global directory Logical filename is used for NRT, while complete filename is used for Reprocessing [List of String]	t_xml_pcdata	20
ProdModel	Output Product model filename which corresponds to the output products to generate There is one product	t_xml_pcdata	10

model per product type to be generated by the PPF [List of String]

6.8.2 Structure level: 1

Structure layout of t_xml_pcdata **terminal**

```
<!ELEMENT t_xml_pcdata (#PCDATA)>

<!ATTLIST t_xml_pcdata

  format CDATA #FIXED '%s'>
```

Format %s
Type C or Dim char *

7. GLOSSARY

Please refer to the document "Glossary of terms and abbreviations [EPS-ASPI-LD-0010]

THIS PAGE IS INTENTIONALLY LEFT BLANK

8. INDEX

THIS PAGE IS INTENTIONALLY LEFT BLANK

APPENDIX A: Product model (records to be filled by PPF)

The first column indicates if the corresponding field shall be filled by PPF or PGF.

Fill by	Field	Description
PGF	RECORD_CLASS	Class of Record
PGF	INSTRUMENT_GROUP	Defining group for record subclasses
PGF	RECORD_SUBCLASS	Subclass of this record class
PGF	RECORD_SUBCLASS_VERSION	Version of this particular format of the record case
PGF	RECORD_SIZE	Total size of the record case (including this header)
PPF	RECORD_START_TIME	Start Time for this record - context will depend on record class
PPF	RECORD_STOP_TIME	Stop Time for this record - context will depend on record class
PPF	PRODUCT_NAME	Complete name of the product
PPF	PARENT_PRODUCT_NAME_1	Name of the parent product from which this product has been produced For Level 0 products, this field is filled with lower case x's
PPF	PARENT_PRODUCT_NAME_2	Name of the parent product from which this product has been produced For Level 0 products or products for which this is not appropriate, this field is filled with lower case x's
PPF	PARENT_PRODUCT_NAME_3	Name of the parent product from which this product has been produced For Level 0 products or products for which this is not appropriate, this field is filled with lower case x's
PPF	PARENT_PRODUCT_NAME_4	Name of the parent product from which this product has been produced For Level 0 products or products for which this is not appropriate this field is filled with lower case x's
PGF	INSTRUMENT_ID	Instrument identification
PGF	INSTRUMENT_MODEL	Instrument Model identification
PPF	PRODUCT_TYPE	Product Type
PPF	PROCESSING_LEVEL	Processing Level Identification
PGF	SPACECRAFT_ID	Spacecraft identification
PPF	SENSING_START	UTC Time of start of sensing data in this object (PDU, ROI or Full Product)
PPF	SENSING_END	UTC Time of end of sensing data in this object (PDU, ROI or Full Product)
PGF	SENSING_START_THEORETICAL	Theoretical UTC Time of start of sensing data in the dump from which this object is derived This data is the predicted start time at the MPF level

Fill by	Field	Description
PGF	SENSING_END_THEORETICAL	Theoretical UTC Time of end of sensing data in the dump from which this object is derived This data is the predicted end time at the MPF level
PGF	PROCESSING_CENTRE	Processing Centre Identification
PPF	PROCESSOR_MAJOR_VERSION	Processing chain major version number
PPF	PROCESSOR_MINOR_VERSION	Processing chain minor version number
PPF	FORMAT_MAJOR_VERSION	Dataset Format Major Version number
PPF	FORMAT_MINOR_VERSION	Dataset Format Minor Version number
PPF	PROCESSING_TIME_START	UTC time of the processing at start of processing for the product
PPF	PROCESSING_TIME_END	UTC time of the processing at end of processing for the product
PGF	PROCESSING_MODE	Identification of the mode of processing
PGF	DISPOSITION_MODE	Identification of the diposition mode
PGF	RECEIVING_GROUND_STATION	Acquisition Station Identification
PGF	RECEIVE_TIME_START	UTC time of the reception at CDA for first Data Item
PGF	RECEIVE_TIME_END	UTC time of the reception at CDA for last Data Item
PGF	ORBIT_START	Start Orbit Number, counted incrementally since launch
PGF	ORBIT_END	Stop Orbit Number
PPF	ACTUAL_PRODUCT_SIZE	Size of the complete product
	ASCENDING NODE ORBIT PARAMETERS	
PPF	STATE_VECTOR_TIME	Epoch time (in UTC) of the orbital elements and the orbit state vector this corresponds to the time of crossing the ascending node for ORBIT_START
PPF	SEMI_MAJOR_AXIS	Semi major axis of orbit at time of the ascending node crossing
PPF	ECCENTRICITY	Orbit eccentricity at time of the ascending node crossing
PPF	INCLINATION	Orbit inclination at time of the ascending node crossing
PPF	PERIGEE_ARGUMENT	Argument of perigee at time of the ascending node crossing
PPF	RIGHT_ASCENSION	Right ascension at time of the ascending node crossing
PPF	MEAN_ANOMALY	Mean anomaly at time of the ascending node crossing
PPF	X_POSITION	X position of the orbit state vector in the orbit frame at ascending node
PPF	Y_POSITION	Y position of the orbit state vector in the orbit frame at ascending node
PPF	Z_POSITION	Z position of the orbit state vector in the orbit frame at

Fill by	Field	Description
		ascending node
PPF	X_VELOCITY	X velocity of the orbit state vector in the orbit frame at ascending node
PPF	Y_VELOCITY	Y velocity of the orbit state vector in the orbit frame at ascending node
PPF	Z_VELOCITY	Z velocity of the orbit state vector in the orbit frame at ascending node
PGF	EARTH_SUN_DISTANCE_RATIO	Earth-Sun distance ratio - ratio of current Eart-Sun distance to Mean Earth-Sun distance
PGF	LOCATION_TOLERANCE_RADIAL	Nadir Earth location tolerance radial
PGF	LOCATION_TOLERANCE_CROSSTRACK	Nadir Earth location tolerance cross-track
PGF	LOCATION_TOLERANCE_ALONGTRACK	Nadir Earth location tolerance along-track
PGF	YAW_ERROR	Constant Yaw attitude error
PGF	ROLL_ERROR	Constant Roll attitude error
PGF	PITCH_ERROR	Constant Pitch attitude error
	LOCATION SUMMARY	
PGF	SUBSAT_LATITUDE_START	Latitude of sub-satellite point at start of the data set
PGF	SUBSAT_LONGITUDE_START	Longitude of sub-satellite point at start of the data set
PGF	SUBSAT_LATITUDE_END	Latitude of sub-satellite point at end of the data set
PGF	SUBSAT_LONGITUDE_END	Longitude of sub-satellite point at end of the data set
	Leap Second Information	
PGF	LEAP_SECOND	Occurence of Leap second within the product Field is set to -1, 0 or +1 dependent upon occurrence of leap second and direction
PGF	LEAP_SECOND_UTC	UTC time of occurrence of the Leap Second (If no leap second in the product, value is null)
	Record counts	
PPF	TOTAL_RECORDS	Total count of all records in the product
PPF	TOTAL_MPHR	Total count of all MPHRs in product (should always be 1!)
PPF	TOTAL_SPHR	Total count of all SPHRs in product (should be 0 or 1 only)
PPF	TOTAL_IPR	Total count of all IPRs in the product
PPF	TOTAL_GEADR	Total count of all GEADRs in the product
PPF	TOTAL_GIADR	Total count of all GIADRs in the product
PPF	TOTAL_VEADR	Total count of all VEADRs in the product
PPF	TOTAL_VIADR	Total count of all VIADRs in the product

Fill by	Field	Description
PPF	TOTAL_MDR	Total count of all MDRs in the product
	Record Based Generic Quality Flags	
PPF	COUNT_DEGRADED_INST_MDR	Count of MDRs with degradation due to instrument problems
PPF	COUNT_DEGRADED_PROC_MDR	Count of MDRs with degradation due to processing problems
PPF	COUNT_DEGRADED_INST_MDR_BLOCKS	Count of the number of blocks of MDRs degraded due to degraded instrument
PPF	COUNT_DEGRADED_PROC_MDR_BLOCKS	Count of the number of blocks of MDRs degraded due to degraded processing
	Time Based Generic Quality Flags	
PPF	DURATION_OF_PRODUCT	The duration of the product in milliseconds
PPF	MILLISECONDS_OF_DATA_PRESENT	The total amount of data present in the product
PPF	MILLISECONDS_OF_DATA_MISSING	The total amount of data missing from the prodcut
	Regional Product Information	
PPF	SUBSETTED_PRODUCT	Set when product has been subsetted (e g geographically subsetted using a region of interest filter) Implies the presence of one or more UMARF GIADRs in GAD section for product retrieved from UMARF

END OF DOCUMENT

