


<div data-bbox="140 107 252 163"> The EUMETSAT Network of Satellite Application Facilities </div> <div data-bbox="284 118 547 188">  <b>NWP SAF</b>  Numerical Weather Prediction </div>	<h1>AAPP v9 Top-level Design</h1>	Doc ID :NWPSAF- MO-DS-046 Version : 1.0 Date : 23/05/2022
---	-----------------------------------	---

## AAPP v9 Top-level Design

This documentation was developed within the context of the EUMETSAT Satellite Application Facility on Numerical Weather Prediction (NWP SAF), under the Cooperation Agreement dated 7 September 2021, between EUMETSAT and the Met Office, UK, by one or more partners within the NWP SAF. The partners in the NWP SAF are the Met Office, ECMWF, DWD and Météo France.

COPYRIGHT 2022, EUMETSAT, ALL RIGHTS RESERVED.

Change record			
Version	Date	Author / changed by	Remarks
1.0	23/05/2022	N C Atkinson	

## Table of Contents

<b>1.</b>	<b>INTRODUCTION .....</b>	<b>3</b>
<b>1.1</b>	<b>Reference documents.....</b>	<b>3</b>
<b>1.2</b>	<b>Design Drivers .....</b>	<b>3</b>
<b>2.</b>	<b>COMPONENTS OF AAPP .....</b>	<b>4</b>
<b>3.</b>	<b>AAPP EPS-SG PRE-PROCESSOR STRUCTURE .....</b>	<b>4</b>
<b>3.1</b>	<b>DIRECTORIES .....</b>	<b>4</b>
<b>3.2</b>	<b>DEPENDENCIES .....</b>	<b>5</b>
<b>3.3</b>	<b>BUILD SYSTEM.....</b>	<b>5</b>
<b>3.4</b>	<b>DELIVERY OF EXECUTABLES .....</b>	<b>5</b>
<b>3.5</b>	<b>DATA STORAGE.....</b>	<b>6</b>
<b>3.6</b>	<b>USER OPTIONS .....</b>	<b>6</b>
<b>4.</b>	<b>FUNCTIONALITY OVERVIEW .....</b>	<b>6</b>
<b>4.1</b>	<b>MWS standalone.....</b>	<b>6</b>
<b>4.2</b>	<b>IASI-NG and MWS .....</b>	<b>8</b>
<b>5.</b>	<b>RELATIONSHIPS TO REQUIREMENTS .....</b>	<b>12</b>

## 1. INTRODUCTION

This document defines the top level design for AAPP version 9, in accordance with the requirements of the NWP SAF.

### 1.1 Reference documents<sup>1</sup>

[RD-1]	NWPSAF-MF-UD-001, AAPP Documentation Scientific Description
[RD-2]	NWPSAF-MF-UD-002, AAPP Documentation Software Description
[RD-3]	NWPSAF-MF-UD-003, AAPP Documentation Data Formats
[RD-4]	NWPSAF-MO-UD-004, AAPP Overview
[RD-5]	NWPSAF-MO-UD-005, AAPP Installation Guide
[RD-6]	NWPSAF-MO-UD-036, AAPP User Guide
[RD-7]	NWPSAF-MF-UD-003, OPS-LRS User Manual
[RD-8]	NWPSAF-MO-UD-027, Annex to AAPP scientific documentation: Pre-processing of ATMS and CrIS
[RD-9]	NWPSAF-MO-DS-034, AAPP Version 8 Top Level Design
[RD-10]	NWPSAF-MO-DS-045, AAPP Version 9 Product Specification
[RD-11]	NWPSAF-MO-DS-044, Design proposals for EPS-SG pre-processing in AAPP and MWIPP

### 1.2 Design Drivers

The main addition in AAPP v9, compared with v8, is to support processing of data from the IASI-NG and MWS instruments on the Metop-SG-A satellite series, as explained in [RD-10].

Unlike Metop first generation, AAPP will not be required to carry out instrument calibration and navigation for direct broadcast applications. That function will be done in dedicated level 1 processors (EPSSGL1). Instead, AAPP will need to ingest

- The level 1 data files that are created by the level 1 processors (netCDF format)
- Global and regional level 1 data, distributed by EUMETSAT (BUFR and netCDF format).

Pre-processing should include:

- Spatial filtering for MWS
- Microwave scattering tests for MWS, including a 229 GHz scattering test
- Support for Principal Components for IASI-NG; generation of reconstructed radiances; spatial thinning; ability to change the spectral resolution (i.e. convert IASI-NG to the resolution of IASI)
- Generate BUFR outputs that are suitable for NWP use.

There is also a requirement for AAPP v9 to support the functionality of AAPP v8 for existing satellites (Metop, NOAA, FY-3), as detailed in [RD-10].

A preliminary assessment of the design for the EPS-SG processor in AAPP was provided in [RD-11], and user feedback was invited. The feedback was incorporated into the requirements specification [RD-10], but basically affirmed that the design of [RD-11] was feasible.

<sup>1</sup> See <https://nwpsaf.eu/site/software/aapp/documentation/>

## 2. COMPONENTS OF AAPP

AAPP v8 comprises two distinct components:

1. The “core” AAPP, comprising
  - Level 1 processors for NOAA-POES and Metop (AMSU, HIRS, AVHRR)
  - Pre-processing modules (re-mapping, etc.)
  - Format conversion tools (including BUFR encoding)
2. The IASI level 1 processor OPS-LRS

The above two components are delivered as separate packages, and can be built separately.

For AAPP v9, a third component will be added, to handle all functionality related to EPS-SG. In the current document, we refer to this component as the *AAPP EPS-SG pre-processor*.

The AAPP EPS-SG pre-processor will have the following characteristics:

- Delivered separately from core AAPP and OPS-LRS, but still formally part of AAPP
- Coded in Fortran90, with interfaces to the netCDF Fortran90 API and also to ecCodes.
- Will use a *make* system based on that already implemented in the NWP SAF deliverables MWIPP<sup>2</sup> and IRSP<sup>3</sup>.

This document details the design of the AAPP EPS-SG pre-processor. For information on the design of core AAPP and OPS-LRS, please see [RD-9] and [RD-7] respectively. ***The designs of these components are unchanged.***

## 3. AAPP EPS-SG PRE-PROCESSOR STRUCTURE

### 3.1 DIRECTORIES

The AAPP EPS-SG pre-processor directory structure is shown in Figure 1

<u>Directory</u>	<u>Purpose</u>
bin	Compiled executables
build	.a and .mod files
control	namelist files
definitions	
bufr	Local BUFR tables for ecCodes
src	
bin	Main programs
libcommon	Utilities
libiasi-ng	Subroutines for IASI-NG
libmapping	Subroutines for re-mapping
libmws	Subroutines for MWS

**Figure 1: Directory structure for AAPP EPS-SG pre-processor**

<sup>2</sup> <https://nwp-saf.eumetsat.int/site/software/mwipp/>

<sup>3</sup> <https://nwp-saf.eumetsat.int/site/software/irspp/>

## 3.2 DEPENDENCIES

The external libraries linked directly by the AAPP EPS-SG pre-processor are shown in Table 1.

**Table 1: External libraries used in the AAPP EPS-SG pre-processor**

Library	Typical version	URL
ecCodes	2.24.2	<a href="https://www.ecmwf.int/en/computing/software">https://www.ecmwf.int/en/computing/software</a>
netcdf-f	4.5.3	<a href="https://www.unidata.ucar.edu/downloads/netcdf/">https://www.unidata.ucar.edu/downloads/netcdf/</a>

These libraries in turn have dependencies shown in Table 2.

**Table 2: Libraries that may be needed when building the external libraries of Table 1**

Library	Typical version	URL
cmake	3.21.2 (needs 3.6 or higher)	<a href="https://cmake.org/download/">https://cmake.org/download/</a> (needed for building ecCodes)
hdf5	1.10.7	<a href="https://support.hdfgroup.org/ftp/HDF5/releases/">https://support.hdfgroup.org/ftp/HDF5/releases/</a>
netcdf-c	4.8.1	<a href="https://www.unidata.ucar.edu/downloads/netcdf/">https://www.unidata.ucar.edu/downloads/netcdf/</a>
zlib	1.2.11	<a href="https://zlib.net/">https://zlib.net/</a> (zlib is normally part of the Linux distribution)

A script will be provided in the AAPP EPS-SG pre-processor to build these libraries, for users who do not already have them installed.

## 3.3 BUILD SYSTEM


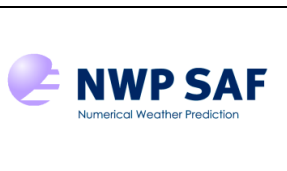
A *configure\_aapp\_epssg.sh* script will be provided, which searches for dependency libraries, creates a top-level Makefile.ARCH, and also creates a shell script *aapp\_epssg\_env.sh* to be sourced by the user at run time.

The user can then run *make* to compile the software. Makefiles are provided as part of the distribution, one in each directory under *src* (see Figure 1). The Makefiles have a simple structure so can easily be modified if needed.

## 3.4 DELIVERY OF EXECUTABLES

As specified in [RD-10], users will also be given the option to receive pre-built executables, with dynamic libraries (following the approach of IRSP). The current baseline is that these will be built on a RedHat 7 or 8 system. The directory structure will be similar to that shown in Figure 1, but *src* will be omitted and *bin* will be fully populated. A *lib* directory will be added, to contain dynamic libraries.

Consideration may also be given to providing Docker or Singularity images, if a requirement is identified. This does not affect the fundamental design.

 	<h1>AAPP v9 Top-level Design</h1>	Doc ID :NWPSAF- MO-DS-046 Version : 1.0 Date : 23/05/2022
---	-----------------------------------	---

## 3.5 DATA STORAGE

The Core AAPP was developed in the 1990s when computer memory was at a premium. Thus the core AAPP uses intermediate-level binary files (e.g. \*.l1b, \*.l1c) to store data even in cases when the output format is BUFR. For the AAPP EPS-SG pre-processor, intermediate files are not used: data are read from the input files (e.g. netCDF) into memory, are processed and then written to the output (e.g. BUFR). Fortran modules (e.g. mws\_data\_mod.f90) define the data variables, and memory is allocated at run-time.

## 3.6 USER OPTIONS

In the core AAPP, user options are provided by the following means:

- Shell script arguments (all executables have a corresponding shell script)
- Environment variables
- Plain text data files

In the AAPP EPS-SG pre-processor, the following scheme is used:

- The top-level executable has a small number of command-line options, typically to specify an input file, an output file and a top-level *namelist*. Dedicated shell scripts are not needed.
- All other options are provided via Fortran namelist files. Each executable may have several namelists for different functions (e.g. one to define the averaging, one to define the BUFR encoding). Example namelists will be provided, which the user is free to customise.

Fortran90 namelists provide a convenient way for the user to initialise specified variables that are used in the program.

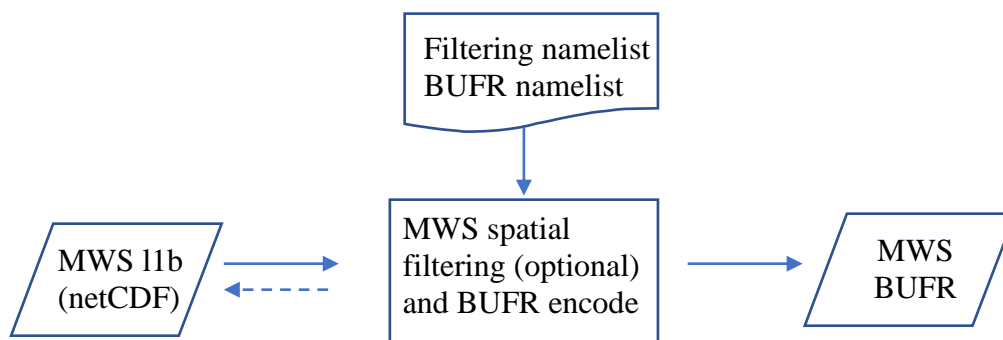
## 4. FUNCTIONALITY OVERVIEW

### 4.1 MWS standalone

For MWS, the following requirements are foreseen:

1. Ingest MWS level 1b data in netCDF or BUFR format
2. Perform spatial filtering and thinning in the same way as is implemented for ATMS, in order to reduce noise and to match the footprint sizes. The user can select either FFT methods or  $n \times n$  averaging. See [RD-8].
3. Write out data in BUFR, or append the filtered brightness temperatures to the input netCDF file.
4. Microwave tests as currently implemented in AAPP, including surface classification, precipitation, scattering. The surface-sensitive channels are the same as AMSU/ATMS (apart from polarisation), but we have the additional 229 GHz channel available for ice cloud detection. The BUFR sequence mentioned above does not include these fields, so a second BUFR sequence may be needed (equivalent to the current ATMS1D).

This workflow for steps 1-3 is illustrated in Figure 2.



**Figure 2: Workflow for the MWS processor, when the input is netCDF (MWS\_main.exe)**

AAPP currently implements a number of scattering tests for AMSU, MHS and ATMS:

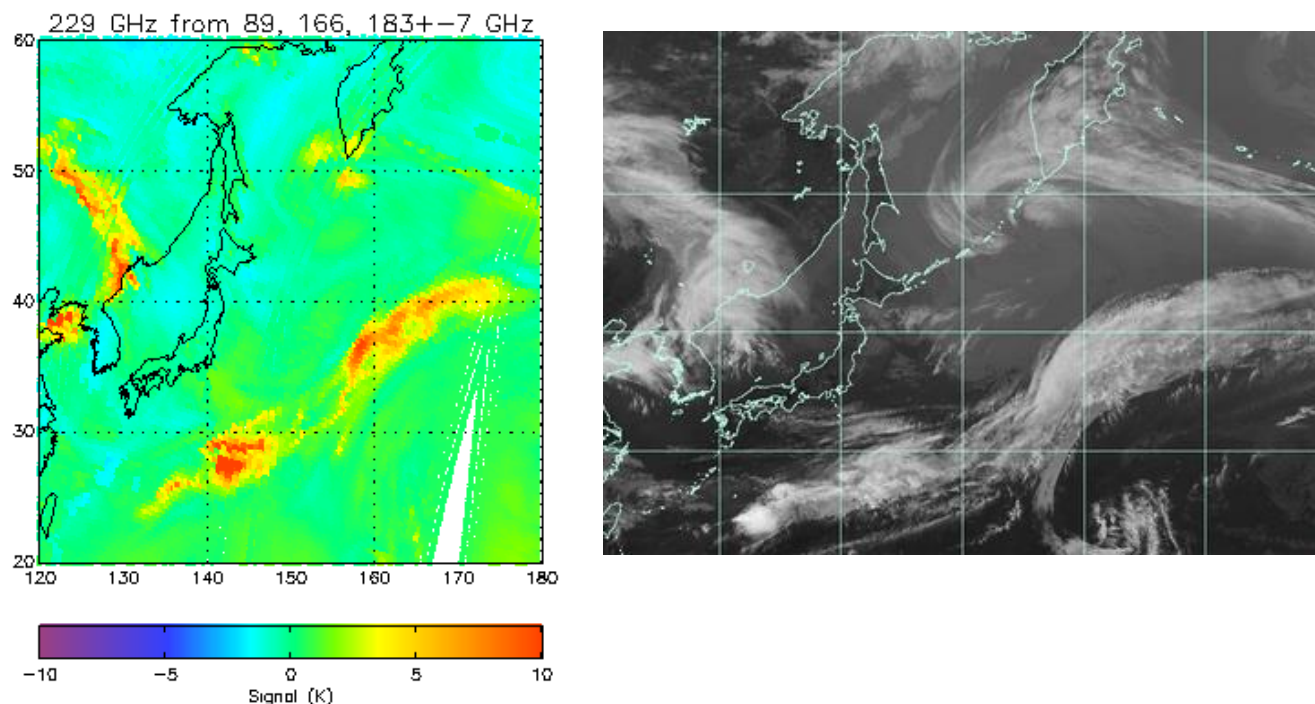
1. Scattering test (ocean): predict 89 GHz from linear combination of 23.8, 31.4 and 50.3 GHz. A scattering index (difference between predicted and observed BT) is written to the level 1d file. The linear coefficients are scan-angle dependent (cubic polynomial in  $1-\sec(z)$ , where  $z$  is the zenith angle of the satellite as viewed from earth).
2. Cirrus test (ocean): predict  $183 \pm 7$  GHz from linear combination of 23.8, 89 and 150 (AMSU-B) or 157 (MHS). A cirrus index is written to the level 1d file. The linear coefficients are scan-angle dependent (cubic polynomial in  $1-\sec(z)$ ).
3. Surface test: identifies 8 surface types by minimising, for each spot, a cost function that uses 23.8, 31.4 and 50.3 GHz. The cost function is  $(\mathbf{T}-\mathbf{T}_m)^T \mathbf{C} (\mathbf{T}-\mathbf{T}_m)$  where  $\mathbf{T}$  are the measured brightness temperatures (column vector) for the three channels and  $\mathbf{T}_m$  are the mean BTs (1 value per channel for each surface type, at 5 different zenith angles).  $\mathbf{C}$  is a covariance matrix (one matrix for each surface type, at 5 different zenith angles).
4. NWC SAF (Bennartz) test, using the channels at 89GHz, 150 GHz and (over land only) 23 GHz.
5. Grody light rainfall (stated as being “redundant” in the AAPP documentation). Flag as rain if  $38.0 + 0.880 T_2 > T_1$  (where  $T_2$  is brightness temperature of channel 2 and  $T_1$  is channel 1)
6. Crosby, Ferraro, Wu (stated as being “redundant” in the AAPP documentation). Probability of precipitation =  $1/(1+\exp(-f))$  where  $f = 10.5 + 0.184 T_1 - 0.221 T_2$

The tests were originally formulated to work with AMSU-A and AMSU-B. They have later been used for AMSU-A and MHS, and for ATMS, with unchanged coefficients, i.e. not optimal. For MWS, the polarisation of channel 1 (23.8 GHz) is QH, compared with QV for all heritage instruments. This means that the coefficients will need to be modified.

The generation of coefficients for microwave scattering indexes is beyond the scope of a top level design document. It will be addressed in the scientific documentation. However, in terms of the design, it is proposed to implement the above tests 1, 3 and 4, with coefficients held in a namelist. This means that they can be easily updated as better simulations become available.



In place of the cirrus test, it is proposed to implement a new test that uses the same regression method as before but in which 229 GHz is predicted from channels at 89 GHz, 166 GHz and  $183 \pm 7$  GHz. This is much more sensitive to cirrus than the old cirrus index, and works over both land and sea – see Figure 3



**Figure 3: Left 229 GHz cirrus signal (difference between predicted 229 GHz and measured), generated from RTTOV-SCAT simulation, for Met Office Unified Model fields. Right: the corresponding Himawari-8 IR 10.45 micron image at 00:00 on 1st June 2021.**

## 4.2 IASI-NG and MWS

For IASI-NG, the following requirements are foreseen

1. For normal NRT use, most users would expect to ingest IASI-NG level 1d files (netCDF or BUFR), which include PC scores but no radiances. However, some centres may wish to use level 1c instead of, or in addition to, the level 1d.
2. Generate reconstructed radiances for a channel selection, or alternatively raw radiances for a channel selection if level 1c has been read
3. By converting the eigenvectors (or spectra), there should be an option to generate reconstructed radiances equivalent to those from IASI first generation.
4. Map MWS to IASI-NG (with spatial filtering for MWS as discussed in 4.1)
5. Spatial thinning for IASI-NG. Similar to that implemented for IASI
6. Write out data in BUFR, or append the filtered brightness temperatures to the input netCDF file.



7. Direct broadcast users may need to convert from level 1c to level 1d, if this functionality is not provided in the level 1 package. The inputs and outputs will be netCDF. (BUFR conversion can be done later).

As mentioned earlier, one difference between IASI-NG and IASI is that IASI has a 2x2 detector array whereas IASI-NG has a 4x4 array (see Figure 4). This means that the IASI thinning strategy of selecting 1 field of view per field of regard (i.e. 1 in 4, giving a spatial sampling of approximately 50km) is not applicable to IASI-NG. If the user selects thinning in the BUFR product, the following options are proposed, controlled via a namelist:

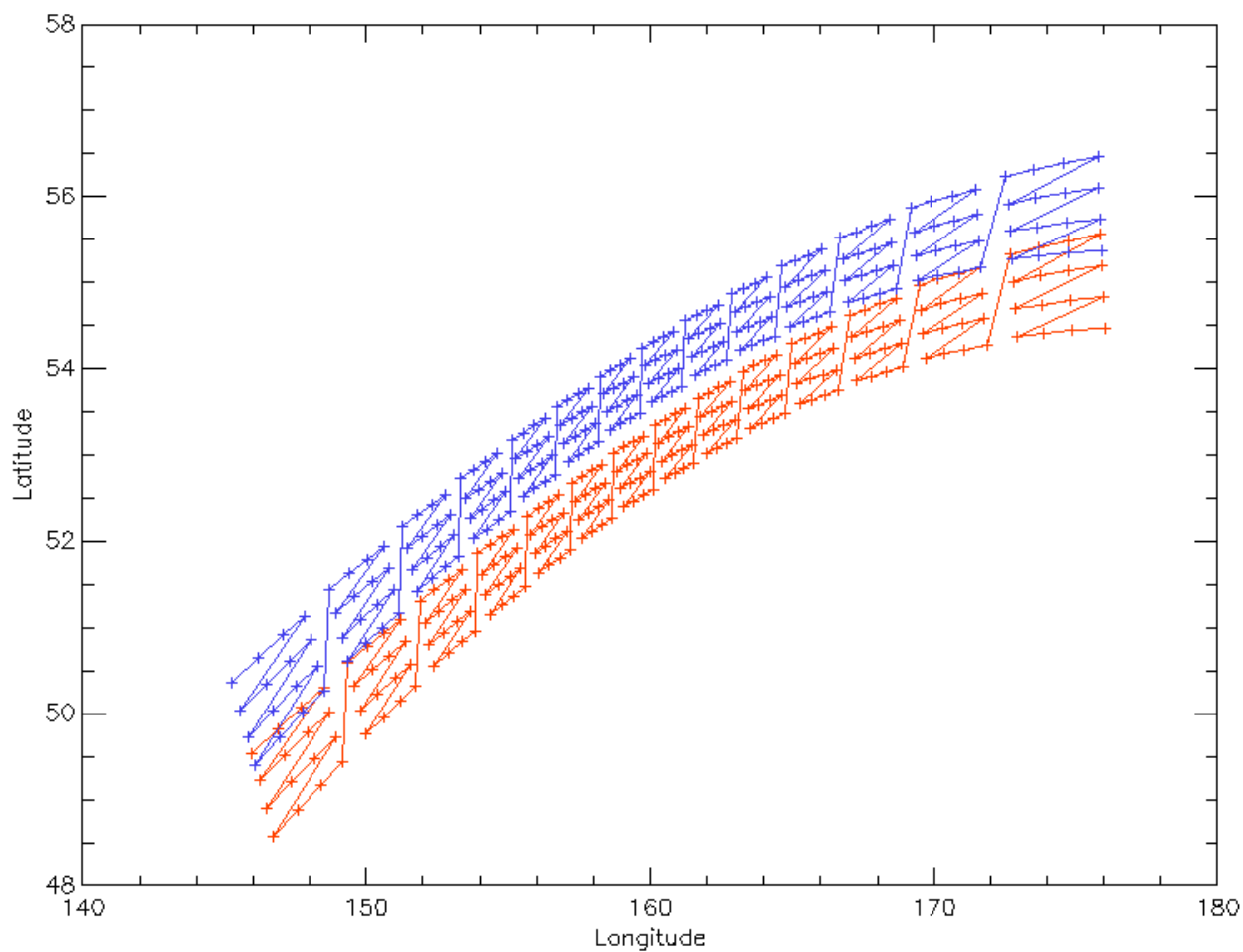
1. The FOVs most likely to be clear, based on METImage cloud mask (if present – see comment below). Choose 1 FOV per field of regard (~100km) or 4 FOVs (~50km)
2. A fixed set of FOVs in each field of regard. The user may choose a different set for different scan lines, e.g. a 4-scan repeating pattern.
3. Warmest IASI-NG FOV for a specified channel in band 1 (e.g. close to 10.8 microns).
4. Method 2 in polar regions (defined by a latitude threshold) and method 3 elsewhere.

The first option is new; the others mirror the options available in AAPP for CrIS.

Note that provision of cloud information from METImage is planned as an option in the IASI-NG level 1 processor (EPSSGL1PP). For direct broadcast users, this will make use of the PPS<sup>4</sup> cloud mask. Therefore AAPP itself will *not* ingest any METImage data.

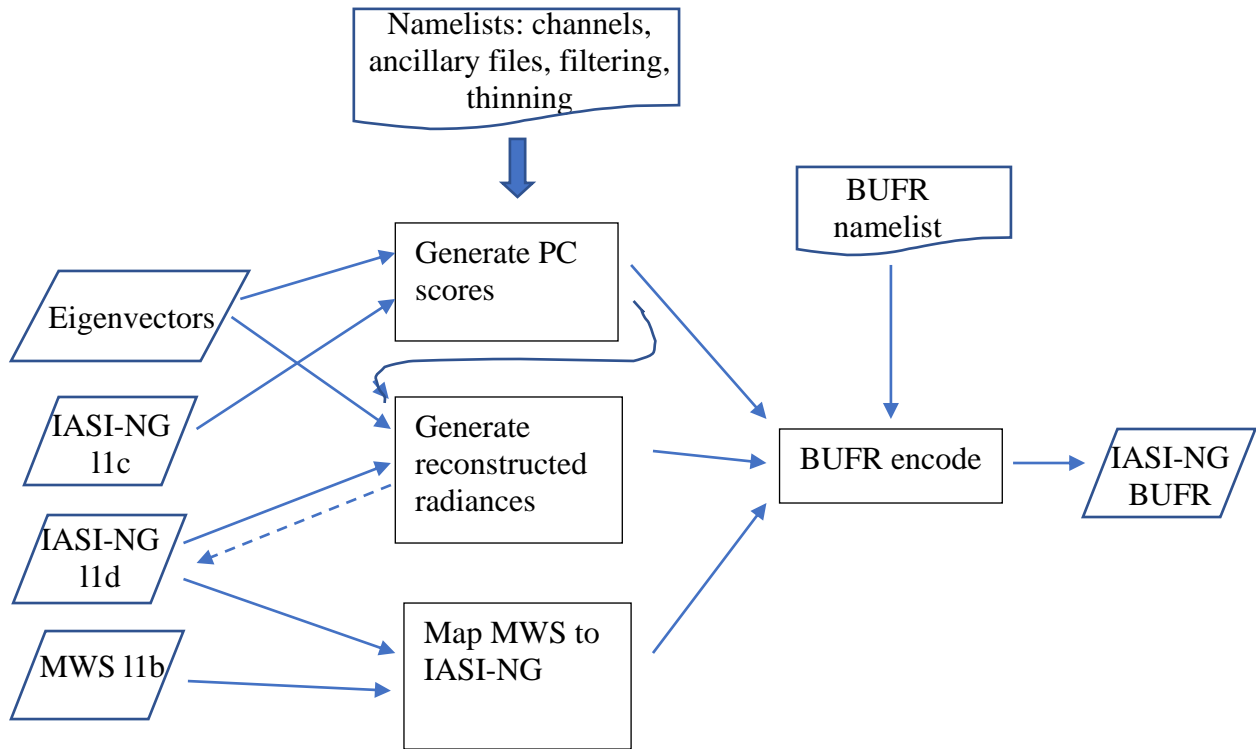
---

<sup>4</sup> <https://www.nwcsaf.org/>

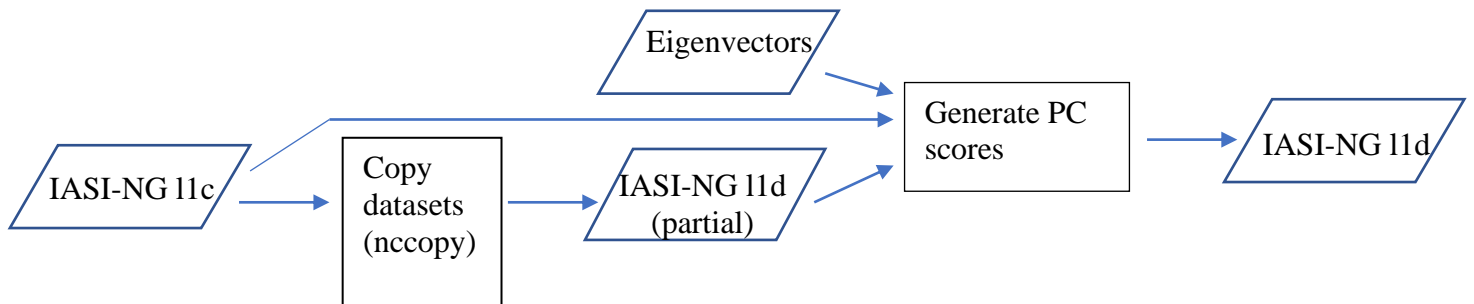


**Figure 4: IASI-NG sample locations for 2 successive scan lines (red then blue). There are 14 fields of regard per scan (224 spectra)**

Workflows are illustrated in Figure 5 and Figure 6.





**Figure 5: Workflow for the IASI-NG processors (NG\_main.exe).** Not all functions are mandatory, e.g. generation of PC scores is only needed if level 1d is not supplied.



**Figure 6 Workflow for IASI-NG conversion from level 1c to 1d (NG\_1c\_to\_1d.sh)**

For mapping of MWS to IASI-NG, it is planned to use the same method as is implemented for ATMS and CrIS, documented in [RD-8]. The existing code *map\_atms\_to\_cris.F* will be converted to Fortran90 and made generic so that it can be used in the future.

Considering the implementation of an option to degrade IASI-NG data to IASI-like spectral resolution (i.e.  $0.25 \text{ cm}^{-1}$  sampling), this is most easily accomplished when working with PC scores, because it is only necessary to modify the eigenvectors. However, the same method can be used with IASI-NG full-spectra. In brief, for each spectrum (or eigenvector) a forward Fast Fourier

 	<h1>AAPP v9 Top-level Design</h1>	Doc ID :NWPSAF- MO-DS-046 Version : 1.0 Date : 23/05/2022
---	-----------------------------------	---

Transform (FFT) is applied, the interferogram is truncated and an inverse FFT is applied. An appropriate channel selection can then be used.

## 5. RELATIONSHIPS TO REQUIREMENTS

In this section we demonstrate how the design of the AAPP EPS-SG pre-processor satisfies the requirements of [RD-10]. Text in the “Requirement” column is taken from [RD-10].

Requirement	How it will be met
Data ingest (non BUFR) <ul style="list-style-type: none"> <li>Ingest level 1 MWS and IASI-NG data in netCDF format. This includes output from the EPS-SG direct broadcast software (EPSSGL1)</li> </ul>	Via the netCDF Fortran API. Note that the “native” EPS-SG level 1 formats are common to global data distribution and to direct broadcast processing.
BUFR input and output <ul style="list-style-type: none"> <li>Ingest level 1 MWS and IASI-NG data in BUFR format</li> <li>Convert level 1 MWS and IASI-NG data from netCDF to BUFR that is compatible with EUMETSAT products</li> </ul>	Input and output of BUFR data will be via ecCodes Fortran API.  At the time of writing, EUMETSAT have not yet finalised the BUFR sequences to be used for MWS and IASI-NG, though prototypes are available. The software is capable of writing standard sequences or bespoke sequences, and local descriptors can be defined if necessary.
Sounder pre-processing: <ul style="list-style-type: none"> <li>Generate microwave scattering indexes for MWS</li> <li>Pre-processing of MWS and IASI-NG: spatial filtering of MWS; map MWS to IASI-NG; spectral and spatial thinning of IASI-NG.</li> <li>Option to degrade IASI-NG data to IASI-like spectral resolution (i.e. 0.25 cm<sup>-1</sup> sampling)</li> </ul>	These steps are discussed in detail in section 4.  For microwave scattering indexes, the approach will follow that of the Core AAPP, but with coefficients provided via namelist rather than ad-hoc data files.  Mapping will follow the approach taken for mapping ATMS to CrIS, using actual geolocation rather than time stamps and nominal scan geometry.  IASI-NG spectral response function manipulation will be via FFT.
AAPP outputs: Different user applications require different output formats. Supported formats include: <ul style="list-style-type: none"> <li>netCDF for IASI-NG and MWS. Outputs may be appended to the input files where applicable.</li> <li>BUFR, for MWS alone, IASI-NG alone and for MWS mapped to IASI-NG</li> </ul>	For BUFR output, see comments earlier in the table.  NetCDF output will generally follow the approach adopted in MWIPP and IRSP, with derived fields being appended to the input file. If a requirement emerges for more specialised products, that can be accommodated within the design.