The NuSTAR Data Analysis Software Guide

M. Perri, S. Puccetti, N. Spagnuolo, R. Ficcadenti
(ASI Space Science Data Center)

&

A. Davis, K. Forster, B. Grefenstette, F. Harrison, K. Madsen
(California Institute of Technology)

Version 1.9.6
July 2020

Document History

Version	Date	Comment
1.0	August 18, 2012	First version
1.1	September 21, 2012	Revised and updated for NuSTARDAS v. 0.6.1
1.2	October 31, 2012	Revised and updated for NuSTARDAS v. 0.8.0
1.3	August 29, 2013	First public version, revised and updated for NuSTARDAS v. 1.2.0
1.4	November 21, 2013	Revised and updated for NuSTARDAS v. 1.3.0. Addition of analysis cases section 6.
1.5	December 13, 2013	Revised and updated for NuSTARDAS v. 1.3.1.
1.6	April 23, 2014	Revised and updated for NuSTARDAS v. 1.4.0.
1.7	May 30, 2014	Revised and updated for NuSTARDAS v. 1.4.1.
1.7.1	October 30, 2014	Minor spelling and grammar corrections.
1.8	July 20, 2015	Revised and updated for NuSTARDAS v. 1.5.1.
1.9	April 20, 2016	Revised and updated for NuSTARDAS v. 1.6.0.
1.9.1	January 24, 2017	Revised and updated for NuSTARDAS v. 1.7.1. This was a minor update with bug fixes to *nucalcsaa, nupipeline, nuscreen* and *nufilter* software modules. See release notes for HEASoft v6.20.
1.9.2	July 14, 2017	Revised and updated for NuSTARDAS v. 1.8.0. This was an update to the modules *nucalcsaa, nupipeline, nuexpomap* and *numkarf* software modules. See release notes for HEASoft v6.22.
1.9.3	September 28, 2017	Updated reference to NuSTARDAS and CALDB version in section 1.1
1.9.4	October 29, 2019	Update orbit file to be used for the barycenter correction
1.9.5	April 10, 2020	Revised and updated for NuSTARDAS v. 1.9.2
1.9.6	July 7, 2020	Revised and updated for NuSTARDAS v. 2.0.0

Table 1-1: Document History

Contents

..... 5

1 INTRODUCTION 9

1.1 Scope 9

1.2 Software Characteristics 9

1.3 Software Breakdown 9

1.4 Downloading and installing 11

1.5 Guide Organization 12

2 NuSTAR DATA FILES 13

2.1 NuSTAR File Naming Convention 13

2.2 Archive Directory Structure 13

3 DATA CALIBRATION (STAGE 1) 17

3.1 Standard Data Processing with *'nupipeline'* (Stage 1) 17

3.2 Metrology Data Processing (*'numetrology'*) 18

3.3 Attitude File Correction (*'nuattcorr'*) 19

3.4 Bad Pixels Flagging (*'nuflagbad'*) 20

3.5 Hot Pixels Searching and Flagging (*'nuhotpix'*) 21

3.6 Energy Corrections, Event Reconstruction and GRADE Assignment (*'nucalcpha'*) 22

3.7 Gain Correction (*'nucalcpi'*) 26

3.8 Events Flagging (*'nuflagevt'*) 27

3.9 Coordinates Transformation (*'nucoord'*) 28

3.9.1 Detector Coordinates Calculation (*'nucalcpos'*) 29

3.9.2 Sky Coordinates Calculation (*'coordinator'*) 30

3.9.3 Optical Axis, Aperture Stop Center and Detector Reference Pixel Sky Positions (*'nuskypos'*) 31

4 EVENTS SCREENING AND EXPOSURE MAPS GENERATION (STAGE 2) 33

4.1 Standard Data Processing with *'nupipeline'* (Stages 1 and 2) 33

4.2 Rationale 34

4.3 SAA passages calculation (*'nucalcsaa'*) 34

4.4 Filter File Generation (*'nufilter'*) 36

4.5 Good Time Intervals Generation and Data Screening (*'nuscreen'*) 37

4.6 Dead Time Correction (*'nulivetime'*) 38

4.7 Screening criteria 39

4.7.1 Screening Criteria associated with the attitude and orbital parameters 39

4.7.2 Screening Criteria associated with the mast position 39
4.7.3 Screening Criteria associated with the instrumental parameters 40
4.7.4 Standard Screening Expressions for attitude/orbital/instrumental parameters 40
4.7.5 Screening Criteria associated with the event proprieties 41
4.7.6 SCIENCE observational mode: summary. 42
4.8 Examples of how to screen the data with custom selections 43
4.8.1 Example of how to use *'nuscreen'* 43
4.8.2 Example of how to use nupipeline 43
4.9 Generation of Exposure Maps (*'nuexpomap'*) 44
4.9.1 Exposure maps generation in 'nupipeline' 46
4.10 Splitting of the Spacecraft Science 06 cleaned event files (*'nusplitsc'*) 46
4.11 Transformation of SKY coordinates in DET1 and DET2 coordinates (*'nuskytodet'*) 48
5 EXTRACTION OF PRODUCTS (STAGE 3) 49
5.1 Standard Data Processing with *'nuproducts'* (Stage 3) 50
5.2 Standard Data Processing with *'nupipeline'* (Stages 1, 2 and 3) 51
5.3 The Extraction of high-level data products (*'nuproducts'*) 51
5.4 The Extraction of high-level data products with 'nupipeline' 57
5.5 Apply backscale correction to energy spectra (*'nubackscale'*) 58
5.6 Apply livetime, PSF/EXPOSURE and vignetting corrections to light-curves (*'nulccorr'*) 59
5.7 Generation of ARF files (*'numkarf'*) 61
5.8 Generation of RMF files (*'numkrmf'*) 63
6 SPECIFIC ANALYSIS CASES 65
6.1 Standard Processing 65
6.2 Extraction of light-curves with a user defined energy range 66
6.3 Generation of high-level data products with a user defined temporal filter 67
6.4 Generation of cleaned event files with a user defined temporal filter 67
6.5 Metrology data out of range: recovery of lost exposure time 67
6.6 Barycenter Correction 68
6.6.1 Method 1: Level 3 files correction with *'nuproducts'* 68
6.6.2 Method 2: light-curve correction with *'barycorr'* 69
6.7 Scientific analysis of spacecraft 06 cleaned event files 69
7 CALIBRATION FILES 73
7.1 Introduction 73
7.2 List of calibration files used by NuSTARDAS 73

8 APPENDIX A: FITS FILE STRUCTURE 76
8.1 Level 1 File Format 76
8.2 Level 1a File Format 77
8.3 Level 2 File Format 78
8.4 GTI table FITS Format 80
8.5 Bad Pixel table FITS Format 80
8.6 Filter File Format 81
8.7 Optical Axis File Format 82
8.8 DET1 Reference Pixel File Format 83
8.9 Mast aspect solution File Format 83
8.10 Position Sensing Detector File Format 84

Index of Tables

Table 1-1: Document History 3
Table 7-1: Calibration Files 73
Table 8-1: Level 1 FITS File structure 76
Table 8-2: Level 1 FITS File Events Table Columns 76
Table 8-3: Level 1a FITS File structure 77
Table 8-4: Level 1a FITS File Events Table Columns 77
Table 8-5: Level 2 FITS File structure 78
Table 8-6: Level 2 FITS File Events Table Columns 79
Table 8-7: GTI Table columns 80
Table 8-8: Bad Pixels Table columns 80
Table 8-9: Filter FITS File structure 81
Table 8-10: Filter Table Columns 81
Table 8-11: Optical Axis FITS File structure 82
Table 8-12: Optical Axis Table Columns 82
Table 8-13: DET1 Reference Pixel FITS File structure 83
Table 8-14: DET1 Reference Pixel Table Columns 83
Table 8-15: Mast aspect solution FITS File structure 83
Table 8-16: Mast aspect solution Table Columns 83
Table 8-17: Position Sensing Detector FITS File structure 84
Table 8-18: Position Sensing Detector Table Columns 84

Index of Figures

Figure 1: NuSTARDAS data reduction flow diagram 10

Figure 2: Definition of the NuSTAR event grades 25

Figure 3: Relationship between the raw telemetry coordinates of the four sub-detectors (coordinates 'RAWX', 'RAWY') and the Focal Plane Bench Frame (coordinates 'DET1X', 'DET1Y'). 29

Figure 4: Relationship between the detector coordinates in the Focal Plane Bench frame (columns 'DET1X','DET1Y') and in the Optics Bench frame (columns 'DET2X', 'DET2Y'). 30

Figure 5: NuSTAR FPMA sky images of the Circinus Galaxy (obsID 60002039002) for observation modes 01 (left) and 06 (right). In the mode 01 image, derived from CHU4 data, the target appears as a point-like source, while the mode 06 image shows multiple centroids from the same target due to the poor aspect reconstruction and the switching between different CHU1/CHU2/CHU3 combinations. 70

Figure 6: NuSTAR FPMA sky images of the Circinus Galaxy (obsID 60002039002) for observation mode 06 as decomposed according to different CHU1/CHU2/CHU3 combinations using the '*nusplitsc'* software module. In this example the target appears as point-like source for the CHU3 and CHU1&3 combinations. 71

1 INTRODUCTION

1.1 Scope

This Guide describes the NuSTAR Data Analysis Software (NuSTARDAS), the software for the processing of scientific data taken with the NuSTAR observatory. In this guide the processing procedures for the calibration, the screening and the generation of high-level NuSTAR data products (energy spectra, light-curves, sky images) to be used for scientific analysis are discussed. The NuSTARDAS package has been jointly developed by the ASI Space Science Data Center (SSDC, Italy) and the California Institute of Technology (Caltech, USA).

This document does not provide a description of the NuSTAR observatory and of its scientific payload for which the reader is referred to the primary mission description document (Harrison et al., 2013, ApJ, 770, 103). This guide is a living document, following the evolution of the processing algorithms, the calibration information and the data analysis techniques. This version of the guide is written based on version **2.0.0** of the NuSTARDAS package and version **20200513** of the NuSTAR CALDB. It will be regularly updated following the future software changes.

> *Note that analysis of observations performed after an adjustment to the onboard laser metrology system on March 17th, 2020 will require NuSTARDAS version 1.9.2 (and later).*

1.2 Software Characteristics

The NuSTARDAS package produces cleaned, calibrated event list files and standard high-level scientific products starting from the FITS formatted telemetry data. The software is designed as a collection of software modules (tasks) each dedicated to a single function. The package includes also a main script, the NuSTAR pipeline (named *'nupipeline'*) allowing users to automatically run in sequence all the tasks for the data processing.

The NuSTAR tasks are written in FTOOLS style and the whole package is fully compatible with the HEASoft software, maintained and distributed by the NASA-HEASARC data center (`http://heasarc.gsfc.nasa.gov`). The NuSTARDAS package is officially integrated in the HEASoft and distributed to the users within its standard public releases. Questions about the NuSTARDAS software package can be addressed to the HEASARC help desk at `http://heasarc.gsfc.nasa.gov/cgi-bin/Feedback` where you should select “NuSTAR” as the mailing list.

Multi-mission FTOOLS from the HEASoft package are also used in several NuSTARDAS modules. The NuSTAR software parameter interface is implemented with the PIL (Parameter Interface Library) library and the I/O with the FITS data files makes use of the FITSIO library.

The NuSTAR software tasks retrieve the calibration files structured in HEASARC's calibration database (CALDB) system. The NuSTARDAS data processing software uses FITS data format following the NASA-OGIP standards as inputs and outputs and makes exclusive use of open source software (C and Perl languages).

1.3 Software Breakdown

The input to the NuSTARDAS package is the NuSTAR FITS formatted telemetry data produced at the NuSTAR Science Operations Center (NuSTAR SOC) in Caltech. The NuSTARDAS data processing is organized in three distinct stages for the calibration, the screening and the extraction of high-level scientific products as illustrated in Fig. 1 showing the NuSTARDAS high-level data reduction flow diagram.

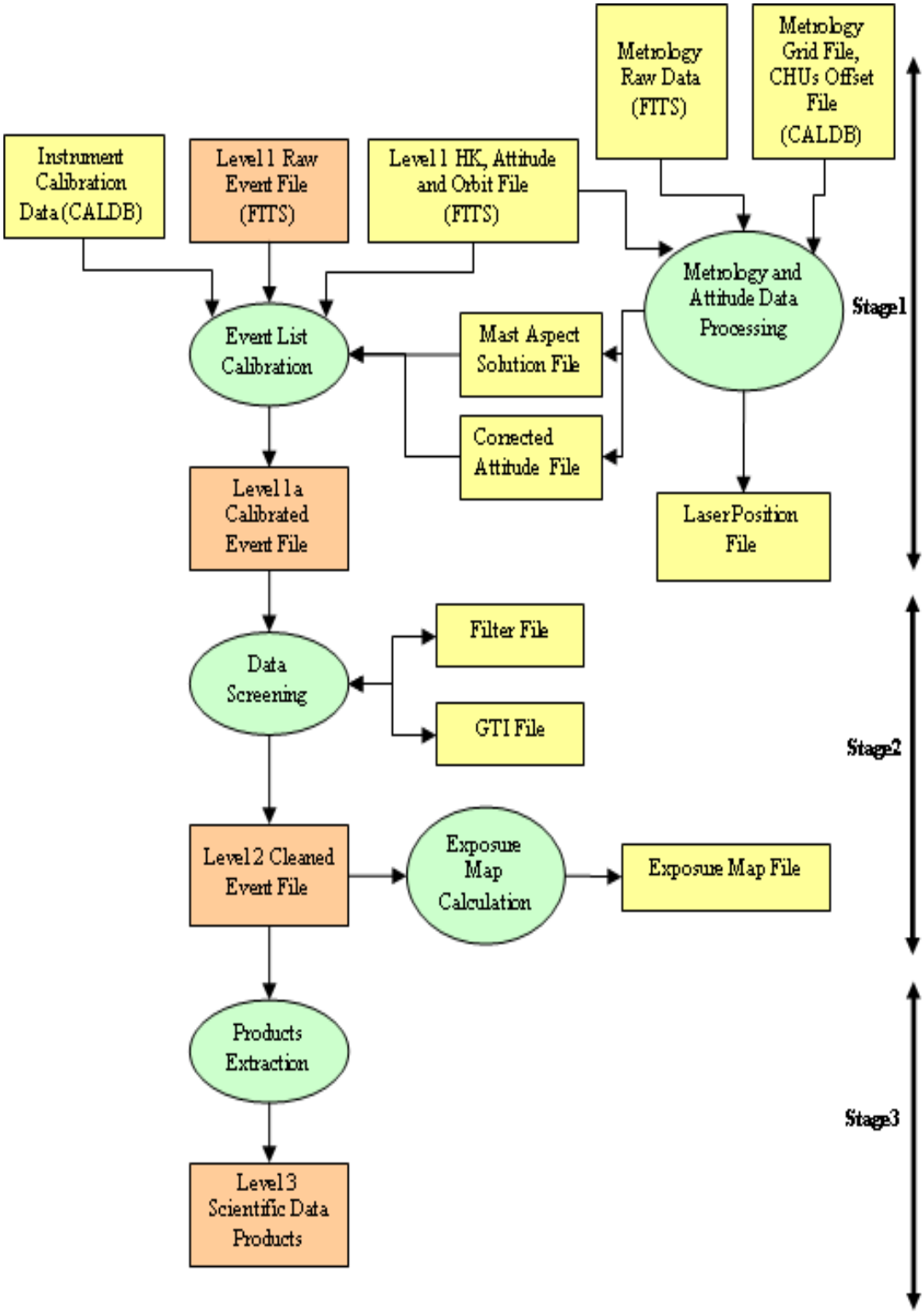

Figure 1: NuSTARDAS data reduction flow diagram

These three main processing stages are:

1. **Data Calibration**: FITS formatted telemetry processing to produce calibrated event files (*Stage 1*);
2. **Data Screening**: calibrated event files filtering by applying cleaning criteria on specified orbital/instrument parameters and event properties to produce cleaned event files (*Stage 2*);
3. **Products Extraction**: generation of high-level scientific products (light-curves, energy spectra, sky images, Ancillary Response Files, Redistribution Matrix Files) (*Stage 3*).

The processing steps listed above defines the following levels of science data:

- **Level 0**: Raw Telemetry packets;
- **Level 1**: Telemetry formatted in FITS format;
- **Level 1a**: Calibrated Events Files;
- **Level 2**: Cleaned and Calibrated Event Files;
- **Level 3**: High-level Scientific Products (e.g. light-curves, energy spectra, sky images).

The files generated with the NuSTARDAS package (e.g. Level 1a/2 event files, Level 3 energy spectra and light-curves) can be read into widely used high-energy astronomy multi-mission data analysis programs such as XSELECT, XSPEC, XRONOS and XIMAGE.

The NuSTARDAS package is run at the NuSTAR SOC in Caltech where it is used to generate the different levels of NuSTAR scientific archive. The package has also been designed to allow users to reproduce any stage of the data processing which could be necessary, for example, because of improved calibration information, updated software modules or because the user wants to use a non-standard data processing. To this end, the *'nupipeline'* script is the main interface for the users. Note that two specific keywords storing the NuSTARDAS software and CALDB versions used to process the data (named 'SOFTVER' and 'CALDBVER', respectively) are added/updated to the output event files by the '*nupipeline'* script.

Once installed, specific help files for all the NuSTARDAS modules can be viewed by using the command **'fhelp task_name'** (e.g. *'fhelp nupipeline'*).

1.4 Downloading and installing

The NuSTARDAS package is integrated in the NASA-HEASARC HEASoft software and can be downloaded at the following link:

http://heasarc.gsfc.nasa.gov/docs/software/lheasoft/

The installation instructions can be found at the same address. Note that the NuSTAR calibration data provided in the CALDB must be installed in order to processing the NuSTAR scientific data. The NuSTAR CALDB can be downloaded from:

http://heasarc.gsfc.nasa.gov/FTP/caldb/data/nustar/fpm/goodfiles_nustar_fpm.tar.gz

Detailed installation instructions are available at:

http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/install.html

Note that the NuSTAR CALDB can be accessed remotely over the internet without having to download the calibration data locally; for instructions see:

http://heasarc.gsfc.nasa.gov/docs/heasarc/caldb/caldb_remote_access.html

Note that remote access may be slower than accessing the CALDB from a local installation.

1.5 Guide Organization

This guide is organized as follows. Chapter 2 describes the NuSTAR data files to provide a basic knowledge of their content, the file naming convention and the archive structure. Chapter 3 gives a description of the processing steps and the software modules involved in the data calibration (Stage 1). Chapter 4 describes the standard screening criteria that are applied to the data and the specific software modules used to produce calibrated and cleaned event files and sky exposure maps (Stage 2). In chapter 5 the third stage of the data reduction, i.e. the generation of high-level data products (e.g. energy spectra, light-curves and sky images), is illustrated. Examples of standard data processing command lines and procedures for some specific analysis issues are given in Chapter 6. In Chapter 7 the calibration files used in the data processing are described. Finally, the appendix details the table formats for the NuSTAR science files.

2 NuSTAR DATA FILES

The NuSTARDAS data processing software uses FITS data format following the NASA-OGIP standards as inputs and outputs. In this chapter we provide a basic knowledge of the file naming convention adopted for NuSTAR, their content and how they are located in the NuSTAR archive structure.

2.1 NuSTAR File Naming Convention

The file name format for the NuSTAR science files uses the following convention:

nuObservationID [M][xx]_[ll].ss

where:

- 'nu' is a prefix to indicate the mission name (NuSTAR);
- 'ObservationID' is a 11 digits number to identify the observation (hereafter obsID);
- 'M' is a one character string that identifies the focal Plane Module (A or B);
- 'xx' is a code to identify the observing mode as defined below:
    - 01 (SCIENCE): normal observing scientific mode;
    - 02 (OCCULTATION): Earth in the field of view;
    - 03 (SLEW): data taken during a spacecraft slew;
    - 04 (SAA): South Atlantic Anomaly passages;
    - 05 (CALIBRATION): on-board calibration radioactive source in the field of view;
    - 06 (SCIENCE_SC): attitude reconstruction from the spacecraft bus star trackers;
- 'll' indicates for event files the processing level (*'uf'* for Level 1 and Level 1a files, *'cl'* for Level 2 files). For other data files it describes their content (e.g. *'met'* for raw metrology data, *'mast'* for mast aspect solution file, *'ex'* for exposure maps);
- 'ss' is the file extension and indicates the data type (e.g. *'evt'* for event files, *'img'* for sky images, *'hk'* for housekeeping files, *'lc'* for light-curves, *'pha'* for energy spectra).

The quantities in square brackets can be omitted, for example i) 'M' (A or B) is not used for data files from the metrology laser system and ii) the observing mode code 'xx' is not used for Level 1/1a event files since the data splitting is carried out during the Stage 2 (see Chapter 4).

2.2 Archive Directory Structure

The NuSTAR public data archive can be accessed at the HEASARC:

http://heasarc.gsfc.nasa.gov/FTP/nustar/data/obs/

and at the SSDC:

http://www.ssdc.asi.it/mmia/index.php?mission=numaster

For each observation, the NuSTAR science data files are stored in the archive in a single directory named after the observationID number.

Each directory has the following structure:

\observationID

\auxil *\hk* *\event_uf* *\event_cl*

Event files from both focal plane modules, housekeeping, metrology and other types of data (e.g. attitude and orbit files) are contained in different sub-directories according to their content and/or level of processing as detailed below. The obsID 10012001002 (3C 273 calibration observation in July 2012) is used as an example.

The data archive may also contain files generated during standard data processing of each observationID at the NuSTAR Science Operations Center at Caltech. These files are not used as input to any NuSTARDAS modules.

- `pipe.log` Log of the output from nupipeline processing
- `nu10012001002.cat.gz` Catalog of files in the archive

The standard pipeline processing log may be useful for comparison to the STDOUT during execution of NuSTARDAS modules.

The following lists contain all the input files used for the NuSTARDAS data processing.

1. The *'auxil'* directory stores the following files:

- `nu10012001002_att.fits` Corrected star tracker attitude file
- `nu10012001002_orb.fits` Orbit file
- `NUSTAR_TLE_ARCHIVE.txt.YYYYDDD` NuSTAR Two-Line Elements file

2. In the *'hk'* directory the following housekeeping FITS files are contained:

- `nu10012001002_ceb.hk` Central Electronics Box (CEB) housekeeping file
- `nu10012001002A_fpm.hk` FPMA instrument housekeeping file
- `nu10012001002B_fpm.hk` FPMB instrument housekeeping file
- `nu10012001002A_dspx.fits` FPMA disabled pixel map file
- `nu10012001002B_dspx.fits` FPMB disabled pixel map file
- `nu10012001002_met.fits` Raw Metrology Data file
- `nu10012001002_chu123.fits` Spacecraft bus star tracker housekeeping file
- `nu10012001002_chu4.fits` Optical bench star tracker housekeeping file
- `nu10012001002_eng.hk` Spacecraft bus attitude quaternion file
- `nu10012001002_obeb.hk` Optical Bench Electronics Box housekeeping file

3. The *'event_uf'* directory stores the Level 1 science event files:

- `nu10012001002A_uf.evt` L1 FPMA event file
- `nu10012001002B_uf.evt` L1 FPMB event file

The *'event_cl'* directory of the archive stores the output files generated by a full processing (Stages 1, 2 and 3). A full list of the output files generated by a standard run of the *'nupipeline'* script is given below. Note that in the public archive the processed attitude file and the instrument housekeeping files are located in the '*auxil'* and '*hk'* directories, respectively (see individual footnotes).

Stage1:

```
nu10012001002_psd.fits          Position Sensing Detector file
nu10012001002_psdcorr.fits      Corrected Position Sensing Detector file
nu10012001002_mast.fits         Mast aspect solution file
nu10012001002_att.fits[1]       Corrected star tracker Attitude file
nu10012001002_obeb.hk           Optical Bench Electronics Box housekeeping file
nu10012001002A_fpm.hk[2]        FPMA updated instrument housekeeping file
nu10012001002B_fpm.hk[2]        FPMB updated instrument housekeeping file
nu10012001002A_bp.fits          FPMA Bad Pixel file
nu10012001002B_bp.fits          FPMB Bad Pixel file
nu10012001002A_hp.fits          FPMA Hot Pixel file
nu10012001002B_hp.fits          FPMB Hot Pixel file
nu10012001002A_oa.fits          FPMA Optical Axis file
nu10012001002B_oa.fits          FPMB Optical Axis file
nu10012001002A_det1.fits        FPMA DET1 Reference Pixel file
nu10012001002B_det1.fits        FPMB DET1 Reference Pixel file
nu10012001002A_uf.evt           L1a FPMA calibrated event file
nu10012001002B_uf.evt           L1a FPMB calibrated event file
```

Stage 2:

```
nu10012001002A.mkf              FPMA Filter file
nu10012001002B.mkf              FPMB Filter file
nu10012001002A.attorb           FPMA Prefilter output file
nu10012001002B.attorb           FPMB Prefilter output file
nu10012001002A01_gti.fits       FPMA GTI file for obs. mode 01
nu10012001002B01_gti.fits       FPMB GTI file for obs. mode 01
nu10012001002A01_cl.evt         L2 FPMA event file for obs. mode 01
nu10012001002B01_cl.evt         L2 FPMB event file for obs. mode 01
nu10012001002A02_gti.fits       FPMA GTI file for obs. mode 02
nu10012001002B02_gti.fits       FPMB GTI file for obs. mode 02
nu10012001002A02_cl.evt         L2 FPMA event file for obs. mode 02
nu10012001002B02_cl.evt         L2 FPMB event file for obs. mode 02
nu10012001002A03_gti.fits       FPMA GTI file for obs. mode 03
nu10012001002B03_gti.fits       FPMB GTI file for obs. mode 03
nu10012001002A03_cl.evt         L2 FPMA event file for obs. mode 03
nu10012001002B03_cl.evt         L2 FPMB event file for obs. mode 03
nu10012001002A04_gti.fits       FPMA GTI file for obs. mode 04
nu10012001002B04_gti.fits       FPMB GTI file for obs. mode 04
nu10012001002A04_cl.evt         L2 FPMA event file for obs. mode 04
nu10012001002B04_cl.evt         L2 FPMB event file for obs. mode 04
nu10012001002A05_gti.fits       FPMA GTI file for obs. mode 05
nu10012001002B05_gti.fits       FPMB GTI file for obs. mode 05
nu10012001002A05_cl.evt         L2 FPMA event file for obs. mode 05
nu10012001002B05_cl.evt         L2 FPMB event file for obs. mode 05
nu10012001002A06_gti.fits       FPMA GTI file for obs. mode 06
nu10012001002B06_gti.fits       FPMB GTI file for obs. mode 06
nu10012001002A06_cl.evt         L2 FPMA event file for obs. mode 06
```

[1] This file is stored in the *'auxil'* archive directory.
[2] These files are stored in the *'hk'* archive directory.
[3] These files are not delivered to the *'event_cl'* archive directory.

```
nu10012001002B06_cl.evt        L2 FPMB event file for obs. mode 06
nu10012001002A01_ex.img[3]     FPMA exposure map for obs. mode 01
nu10012001002B01_ex.img[3]     FPMB exposure map for obs. mode 01
```

Stage 3:

```
nu10012001002A01_sk.img        L3 FPMA image for obs. mode 01
nu10012001002B01_sk.img        L3 FPMB image for obs. mode 01
nu10012001002A01_sr.pha[3]     L3 FPMA source energy spectrum
nu10012001002B01_sr.pha[3]     L3 FPMB source energy spectrum
nu10012001002A01_sr.lc[3]      L3 FPMA source light-curve
nu10012001002B01_sr.lc[3]      L3 FPMB source light-curve
nu10012001002A01_bk.pha[3]     L3 FPMA background energy spectrum
nu10012001002B01_bk.pha[3]     L3 FPMB background energy spectrum
nu10012001002A01_bk.lc[3]      L3 FPMA background light-curve
nu10012001002B01_bk.lc[3]      L3 FPMB background light-curve
nu10012001002A01_im.gif/ps     FPMA Sky image plot
nu10012001002A01_ph.gif/ps     FPMA source energy spectrum plot
nu10012001002A01_lc.gif/ps     FPMA source light-curve plot
nu10012001002B01_im.gif/ps     FPMB Sky image plot
nu10012001002B01_ph.gif/ps     FPMB source energy spectrum plot
nu10012001002B01_lc.gif/ps     FPMB source light-curve plot
nu10012001002A01_sr.arf[3]     L3 FPMA source ARF file
nu10012001002B01_sr.arf[3]     L3 FPMB source ARF file
nu10012001002A01_sr.rmf[3]     L3 FPMA source RMF file
nu10012001002B01_sr.rmf[3]     L3 FPMB source RMF file
nu10012001002A01_src.reg       L3 FPMA source extraction region file
nu10012001002B01_src.reg       L3 FPMB source extraction region file
```

The content of these files and a description of the processing modules used to generate them will be given in the following chapters.

[3] These files are not delivered to the *'event_cl'* archive directory.

3 DATA CALIBRATION (STAGE 1)

This chapter describes the Stage 1 of the NuSTAR data reduction during which the calibration steps are applied to the science event files. This stage also includes 1) the processing of the data from the laser metrology system, measuring the observatory mast alignment temporal changes, and 2) the correction of the spacecraft attitude file.

The inputs of the calibration stage are the Level 1 files storing the telemetry data in FITS format. Stage 1 produces in output the Level 1a calibrated event files and the processed metrology and attitude data files.

For the science event files the following data levels are defined:

- Level 1: Event files containing the original telemetry information reformatted into FITS.
- Level 1a: Event files with the same content of the Level 1 with additional calibrated information.

We note that between Level 1 and Level 1a there is no loss of information, i.e. all the telemetry columns are preserved. It is therefore possible to re-run the Stage 1 NuSTAR software modules on the Level 1a event files to apply new calibration updates (data re-processing).

In this Chapter the various NuSTAR software calibration modules are presented in the order used for the standard data processing along with a description of how they can be used. These calibration steps are coded in the first stage of the *'nupipeline'* script and cases of how to run the pipeline are illustrated in the next Section 3.1.

The list of the Stage 1 software modules is the following:

- *numetrology* - Metrology data processing
- *nuattcorr* - Correction of attitude file
- *nuflagbad* - Flagging events for bad pixels
- *nuhotpix* - Searching for hot and flickering pixels
- *nucalcpha* - Energy correction, PHAS and GRADE calculation
- *nucalcpi* - Gain correction and PI column calculation
- *nuflagevt* - Flagging of events according to various cuts criteria
- *nucoord* - Conversion of Raw coordinates into Detector and Sky coordinates
- *nucalcpos* - Conversion of Raw coordinates into Detector coordinates
- *nuskypos* - Calculation of SKY positions of the optical axis, aperture stop center and DET1 reference pixel

3.1 Standard Data Processing with *'nupipeline'* (Stage 1)

The calibration steps described in this chapter are coded in the *'nupipeline'*, the script designed to automatically run in sequence all the tasks of the NuSTAR data processing. The majority of the input parameters of the single software modules can also be set directly running the *'nupipeline'* and appropriate default values have been set for most of them.

The *'nupipeline'* script assumes that the data files have been downloaded from the archive and it searches in the obsID directory structure for the input files. The directory path of the obsID and the stem of the input files must be provided in input through the parameters *'indir'* and *'steminputs'*, respectively.

The *'exitstage'* input parameters of *'nupipeline'* allows the users to terminate the data reduction at different processing stages (described in Chapter 1). The default value is *'exitstage*=2', i.e. the data are calibrated *and* screened (see next chapter). Below we provide an example of how to run *'nupipeline'* to execute the complete Stage 1 for a standard NuSTAR data processing:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out exitstage=1
```

By default, the *'nupipeline'* processes data files for both Focal Plane Modules (FPMA and FPMB). By setting the input parameter *'instrument'* to *'FPMA'* or *'FPMB'* the users can choose to process only the event files of a single Focal Plane Module, as illustrated below for FPMB:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out exitstage=1 instrument=FPMB
```

3.2 Metrology Data Processing (*'numetrology'*)

The NuSTARDAS software module *'numetrology'* processes the data from the laser metrology system, the device on-board NuSTAR tracking temporal changes of the alignment of the mast connecting the detectors focal plane and the optics bench mirror system.

The task requires in input the following files (parameter names are given in parenthesis):

- Raw Metrology File (*metrologyfile*)
- Metrology grid File (*metgridfile*)
- Spacecraft Alignment File (*alignfile*)

The files generated in output by the task are:

- Position Sensing Detector File (*outpsdfile*)
- Corrected Position Sensing Detector File (*outpsdfilecor*)
- Mast Aspect Solution File (*mastaspectfile*)

The task first calculates the X and Y positions as a function of time of the laser spots on the two Position Sensing Detectors ('PSD0' and 'PSD1') located on the detectors focal plane bench. The X/Y coordinates are written in the output file specified through the input parameter *'outpsdfile'*.

Next, if the input parameter 'psdflag' is set to 'yes' (default value), the X/Y coordinates are corrected for the distortions introduced by the response of the two laser detectors. To this end, the information stored in the Metrology Grid file (read in input via the parameter *'metgridfile'*) is used and the results are written in the output file *'outpsdfilecor'*. During this step, a specific column, named 'METGRID_FLAG', is added to the *'outpsdfilecor'* file to flag time intervals during which the laser spots fall outside the X and/or Y ranges covered by the

Metrology Grid File. By setting 'psdcal=no', the X/Y coordinates are not linearized and their original values are copied into the outputfile 'outpsdfilecor'. The allowed values of the 'METGRID_FLAG' column are '0' (laser spots within the calibrated grid), '1' (laser spots out of the calibrated grid) and '2' (PSD X/Y coordinates not linearized).

In the third step, the lasers positions are used to calculate the Mast Aspect Solution File storing, as a function of time, the roto-translations tracking the spacecraft mast movements. The roto-translations terms are written in the columns 'T_FBOB' and 'Q_FBOB' of the output file *'mastaspectfile'*.

The task processes the raw metrology data (steps 1 and 2 described above) if the input parameter *'metflag'* is set to *'yes'* (default value). By setting *'metflag=no'* only the step 3 is executed by the task and the corrected Position Sensing Detector File must be provided in input by the user through the parameter *'inpsdfilecor'*.

Usage Example:

1. Process the raw metrology data file 'nu10012001002_met.fits' using the calibration information stored in the CALDB. The mast aspect solution file is written in the output file 'nu10012001002_mast.fits'. The X/Y coordinates of the PSD detectors, before and after the correction for the distortions, are stored in the output files 'nu10012001002_psd.fits' and 'nu10012001002_psdcorr.fits', respectively.

```
> numetrology metrologyfile=nu10012001002_met.fits
```

3.3 Attitude File Correction (*'nuattcorr'*)

The NuSTARDAS software module *'nuattcorr'* corrects the attitude file for the offset between the Spacecraft Bus (SC) and the Camera Head Unit #4 (CHU4) coordinate system. The task applies the correction only to the data derived from the spacecraft bus star trackers (CHU123), i.e. to the attitude file rows with the column 'SOURCE=2'. The offset is given by a quaternion read from the CALDB file *'chuoffsetfile'*.

The task requires in input the following files:

- Attitude File (*attfile*)
- CALDB CHUs Quaternion Offset File (*chuoffsetfile*)

The files generated in output by the task are:

- Corrected Attitude File (*outattfile*)

Usage Example:

1. Apply the offset quaternion to the input attitude file 'nu10012001002_att.fits'. The corrected attitude file is written in the output file 'nu10012001002_att_corr.fits'.

```
> nuattcorr attfile=nu10012001002_att.fits
  outattfile=nu10012001002_att_corr.fits
```

3.4 Bad Pixels Flagging (*'nuflagbad'*)

The NuSTARDAS software module *'nuflagbad'* flags events occurring in known bad pixels of the NuSTAR detectors.

The task requires in input the following files:

- Level 1/1a Event File (*infile*)
- On-board disabled pixel file (*dispixfile*)
- CALDB on-ground bad pixel file (*bpfile*)
- User bad pixel file (user*bpfile*)

The files generated in output by the task are:

- Calibrated Level 1a Event File (*outfile*)
- Bad Pixels File (*outbpfile*)

Three different input files that identify bad pixels can be used by the task for event flagging:

1. On-ground CALDB Bad Pixel Calibration File including the most up to date information about known stable bad pixels;
2. Disabled Pixel File storing the list of pixels disabled by the on-board processing for the current observation;
3. User supplied Bad Pixel File including the list of additional bad pixels provided by the user (with the same format as the on-ground CALDB bad pixel file).

The on-ground CALDB Bad Pixel File and the User Bad Pixel File contain for each bad pixel a column 'TIME' including the start epoch at which the pixel has to be considered bad. The Disabled Pixel File, including the list of pixels disabled by the on-board software for the current observation, stores for each disabled pixel the corresponding time interval in the columns 'TIME' and 'TIME_STOP'.

All events are flagged using the information obtained from the three input bad pixel files and the user can choose which bad pixel file to use. Events are marked using the *'nuflagbad'* module by adding/updating the 'STATUS' column of the input Level 1/1a event file. The format of this column is a 16-bit binary number and the value of the 'STATUS' column specifies the quality of the flagging (i.e. if the event falls on a stable bad pixel from the on-ground CALDB File or on an on-board disabled pixel).

The task also flags events if any of the 8 neighborhood pixels falls on a bad pixel or outside the detector boundaries (i.e. the event is located on a detector edge). The position of these neighbor pixels is stored in a new column 'BADPOS' added/updated by the task. The format of this column is an 8-bit binary number.

The list of flags in the 'STATUS' column used by the task is the following:

- b0000000000000001 Event falls in a bad pixel from on-ground CALDB Bad Pixel File
- b0000000000000010 Event falls in a bad pixel from on-board disabled pixel
- b0000000000000100 Event falls in a bad pixel from user bad pixel file
- b0000000000001000 Event has a neighbor bad from bad/disabled/user pixel list
- b0000000000010000 Event falls in a detector edge pixel

In addition to the event flagging, the task also stores the list of bad pixels in four distinct 'BADPIX' extensions, one for each of the four detectors, of the input event file. Optionally, if requested by the user through the parameter *'outbpfile'*, the bad pixels list is also written in a separate output file. The 'BADPIX' extensions

contain the positional and temporal information and a 16-bit binary number column, named 'BADFLAG', indicating the class of the bad pixel with the following meaning:

- b0000000000000001 Bad pixel from on-ground CALDB Bad Pixel File
- b0000000000000010 Disabled pixel from on-board software
- b0000000000000100 Bad pixel in the file provided by the user

Usage Examples:

1. Flag the events of the input event file 'nu10012001002A_uf.evt' considering the bad pixels listed in the on-ground CALDB Bad Pixel File and in the Disabled Pixel File named 'nu10012001002A_dspx.fits'. The list of bad pixels is stored in the 'BADPIX' extensions of the output event file ('nu10012001002A_out.evt') and in the output bad pixel fits file named 'nu10012001002A_bp.fits'.

```
> nuflagbad infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
  dispixfile=nu10012001002A_dspx.fits outbpfile=nu10012001002A_bp.fits
```

2. Flag the events of the input event file 'nu10012001002A_uf.evt' considering the bad pixels listed in the on-ground CALDB Bad Pixel File, in the Disabled Pixel File named 'nu10012001002A_dspx.fits' and in a user provided file named 'userbadpix.fits'. The list of bad pixels is stored in the 'BADPIX' extensions of the output event file ('nu10012001002A_out.evt').

```
> nuflagbad infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
  dispixfile=nu10012001002A_dspx.fits userbpfile=userbadpix.fits
  outbpfile=NONE
```

3.5 Hot Pixels Searching and Flagging (*'nuhotpix'*)

The NuSTARDAS software module *'nuhotpix'* searches and flags anomalous (hot and flickering) pixels. These are detector pixels that may show unstable states of noise signature and if this signal is much greater then the background level they appear as isolated pixel signals resembling X-ray events.

The task requires in input the following files:

- Level 1/1a Event File (*infile*)

The files generated in output by the task are:

- Calibrated Level 1a Event File (*outfile*)
- Hot Pixels File (*outhpfile*)

The hot/flickering pixels search is achieved by applying statistical tests to a set of counts images generated from the input event file. Each counts image is accumulated during a time interval with a duration defined by the input parameter *'binsize'.*

For each image, hot/flickering pixels are identified comparing the counts in each pixel to the mean background counts. First, for each pixel, the probability for its counts to be a Poisson fluctuation of the mean background of the detector is computed. The parameter *'impfac'* allows the user to rescale the background level. If the pixel probability is lower than a Poisson probability threshold (set by the user through the parameter *'logpos'*), the pixel is considered a hot/flickering pixel candidate. To confirm the hot/flickering pixel candidates, their counts are compared to the mean counts of the surrounding pixels in a square cell with size of the order of the Point

Spread Function (parameter *'cellsize'*) to discriminate a hot/flickering pixel from a pixel of a celestial X-ray source in the field of view.

As a second step, by setting the parameter *'cleanflick'* to "yes" (default), hot/flickering pixels are searched using a background estimated locally in its surrounding pixels contained in a cell with size defined by *'cellsize'*. If the local background is zero, the pixel is flagged as hot/flickering if the number of its counts is larger than a threshold value set by the parameter *'bthresh'*. In this second step, setting the parameter 'iterate' to 'yes' enables iterating the search for hot/flickering pixels.

As in the case of the *'nuflagbad'* task, the *'nuhotpix'* module updates the 'STATUS' column of the event file. The task also flags events if any of the 8 neighborhood pixels falls on a hot/flickering pixel. The position of these neighbor pixels is stored in a new column, named 'HOTPOS', added by the task. The format of this column is an 8-bit binary number.

The values of the 'STATUS' column used by 'nuhotpix' is the following:

- b0000000000100000 Event falls in a hot/flickering pixel
- b0000000001000000 Event has a neighbor hot/flickering pixel

In addition to the event flagging, the task also stores the list of hot/flickering pixels in four distinct 'BADPIX' extensions, one for each of the four detectors, of the input event file. Optionally, if requested by the user through the parameter *'outhpfile'*, the bad pixels list is also written in a separate output file. The 'BADPIX' extensions contain the positional and temporal information and a 16-bit binary number column, named 'BADFLAG', indicating the class of the bad pixel. The start and end time values of the time interval where the pixel is hot/flickering is also stored in the two columns 'TIME' and 'TIME_STOP'. The 16-bit binary number column 'BADFLAG' (see 'nuflagbad' help file) is updated by 'nuhotpix' as follows:

- b0000000000100000 Event falls in a hot/flickering pixel

Usage Examples:

1. Search and flag hot/flickering pixels in the NuSTAR event file named nu10012001002A_uf.evt. The list of hot/flickering pixels is stored in the 'BADPIX' extensions of the output event file and in the output hot pixel fits file named 'nu10012001002A_outhp.fits'.

```
> nuhotpix infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
```

3.6 Energy Corrections, Event Reconstruction and GRADE Assignment (*'nucalcpha'*)

The NuSTARDAS software module *'nucalcpha'* processes the NuSTAR event files to correct photon energies, reconstruct events and assign a grade to each event.

The task requires in input the following files:

- Level 1/1a Event File (*infile*)
- CALDB Capacitor Offset File (*offsetfile*)
- CALDB Grade File (*gradefile*)
- CALDB PHA Parameter File *(phaparfile*)

The files generated in output by the task are:

- Calibrated Level 1a Event File (*outfile*)

As a first step, various corrections to the energy of the events are applied. The *'nucalcpha'* task reads the 'PREPHAS' and 'POSTPHAS' columns of the input event files storing, for each event, the nine charge values

(of the 3x3 square pixel array) taken before and after the on-board trigger, respectively. The pre-trigger values are first subtracted from the post-trigger ones and then the "capacitor offset", the "time of rise" and the "common mode" corrections are applied to the energies.

The corrected energies are written in the 'PHAS' column of the output event file. Moreover, the task adds/updates the 'SWTRIG' column of the output event file. The algorithm used is summarized below.

For each event (3x3 pixels array) the task applies the following processing steps:

1. Computation of RAWPHAS (9 dimensional vector):

   RAWPHAS[9] = POSTPHAS[9] - PREPHAS[9]

2. Offset correction, i.e. computation of 'OFFPHAS' (9 dimensional vector):

   OFFPHAS[9] = RAWPHAS[9] - OFFSET[9]

where the 9 values of OFFSET[9] are read from the input CAP_OFFSET file and are dependent on:

- the coordinates (RAWX, RAWY) of the pixels (3X3 array)
- the capacitor number (S_CAP column) of the event the detector (DET_ID column)

3. Time of rise correction, i.e. computation of 'TRPHAS' (9 dimensional vector):

   TRPHAS[9] = OFFPHAS[9] * (1+(NUMRISE/DENRISE)*TIMERISE[9])

where NUMRISE, DENRISE are read from the input event file. The 9 values of TIMERISE[9] are read by default from the input PHAPAR file and are dependent on:

- the coordinates (RAWX, RAWY) of the pixels (3X3 array)
- the detector (DET_ID column)

If the parameter 'phaparfile' is set to 'NONE', the 9 values are provided in input to the task through the specific parameter 'timerise' (default=0.0016).

4. Common mode correction, i.e. computation of 'PHAS' (9 dimensional vector):

   4a) exclusion of the M pixels located out of the detector or falling on bad/hot pixels

   4b) selection of the N pixels ("E+") with TRPHAS >= 'EVTTHR[9]' (not including the M pixels defined in point 4a)

   4c) selection of the 9-N-M pixels ("E-") with TRPHAS < 'EVTTHR[9]'

   The 9 values of EVTTHR[9] are read by default from the input PHAPAR file and are dependent on:

   - the coordinates (RAWX, RAWY) of the pixels (3X3 array)
   - the detector (DET_ID column)

   If the parameter 'phaparfile' is set to 'NONE', the 9 values are provided in input to the task through the specific parameter 'evtthr' (default=55).

4d) calculation of the mean of the pixels with TRPHAS < 'EVTTHR[9]':

< E- >=(&sum E-)/(9-N-M)

4e) computation of PHAS (9 dim. vector) for the N pixels "E+":

PHAS[9] = TRPHAS[9] - < E->

4f) computation of PHAS (9 dim. vector) for the 9-N-M pixels "E-":

PHAS[9] = TRPHAS[9]

4g) setting to zero PHAS values for the M pixels out of detector:

PHAS[9] = 0

5. computation of 'SWTRIG' column (9 dimensional vector):

   SWTRIG[9] = 1 for pixels with TRPHAS >= 'EVTTHR[9]'

   SWTRIG[9] = 0 for pixels with TRPHAS < 'EVTTHR[9]'

In the second step, the task reconstructs the event charge splitting topology and assigns to them a grade following a charge pattern definition stored in the input CALDB file *'gradefile'* (see Figure 2). The values of the event grades are written in the 'GRADE' column of the output event file.

Usage Examples:

1. Calculate PHAS and GRADE values for the event file 'nu10012001002A_uf.evt'. The values of the event software threshold, the time of rise correction coefficients, the capacitor offset values and the GRADE classification are read from the specific CALDB files. The results are written in the output event file 'nu10012001002A_out.evt'.

```
> nucalcpha infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
```

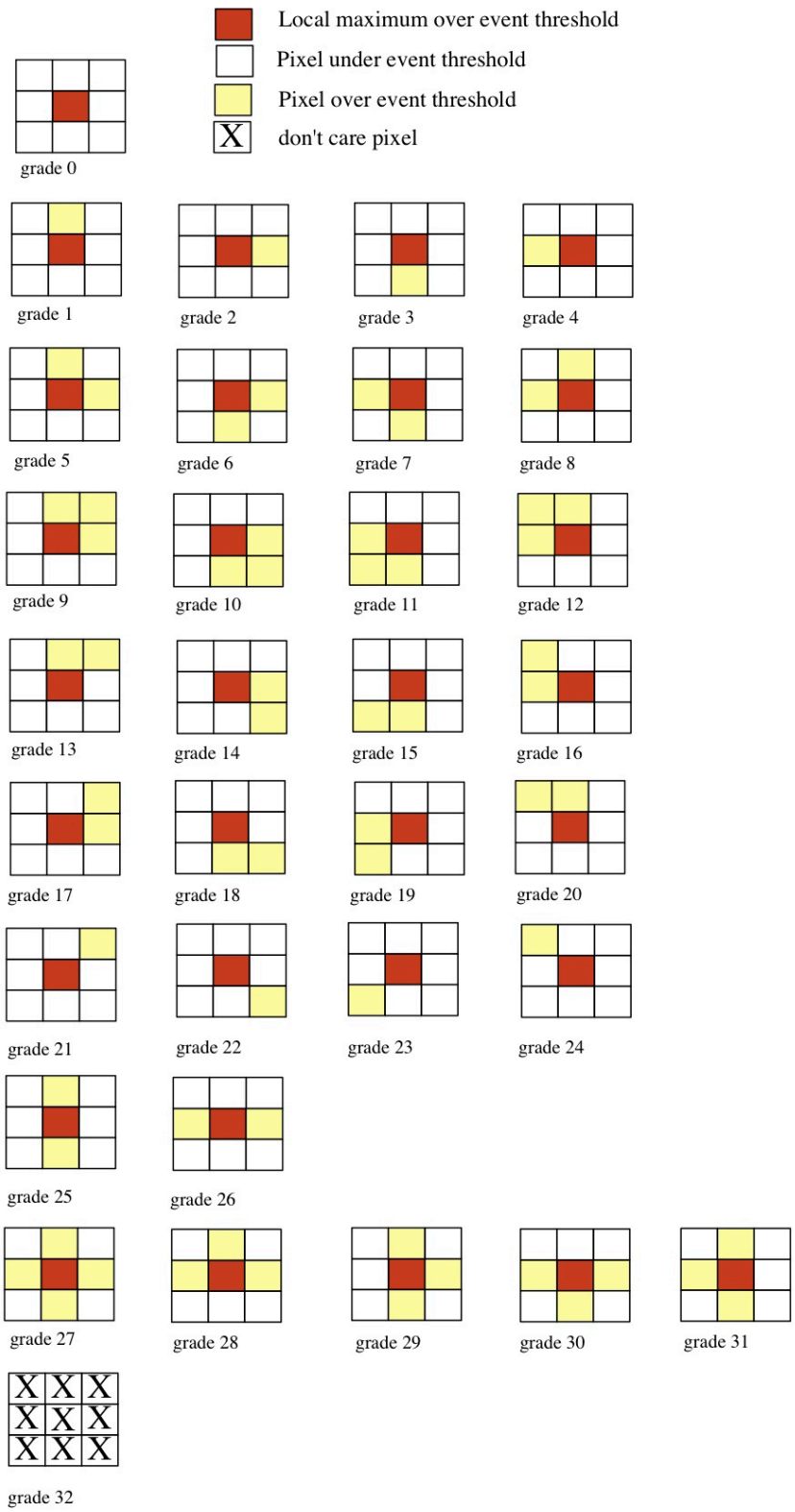

Figure 2: Definition of the NuSTAR event grades

3.7 Gain Correction (*'nucalcpi'*)

The NuSTARDAS software module *'nucalcpi'* processes the NuSTAR event files to convert the charge associated to each event from electronic units ("Pulse Height Amplitude", PHAS) into physical energy units ("Pulse Invariant", PI) by applying gain corrections.

The task requires in input the following files:

- Level 1/1a Event File (*infile*)
- Instrument Housekeeping File (*hkfile*)
- CALDB Gain File (*gainfile*)
- CALDB Charge Loss Correction File *(clcfile*)
- CALDB Charge Loss Correction Filter File *(clcfilterfile*)

The files generated in output by the task are:

- Calibrated Level 1a Event File (*outfile*)

The *'nucalcpi'* task applies a conversion between signal charge and photon energy, which is pixel, grade and temperature dependent. In addition, a temporal evolution of the gain is also included. Specifically, the task reads, for each event, the 'PHAS' column of the input event file and the temperature associated to each event from the input Housekeeping file. Moreover, its position on the detector ('RAWX', 'RAWY' columns) and its grade ('GRADE' column) are read.

Next, by using the gain values stored in the CALDB gain file (parameter *'gainfile'*), the task calculates the energy in physical units and stores the values in a new column (named 'PI') of the output event file. Also, specific charge-loss corrections for multiple pixels events are applied using information from the two input CALDB files *'clcfile'* and *'clcfilterfile'*.

The task also computes the PI values for the 3x3 array pixels below the software threshold and stores the result in the 'SURRPI 'column.

The gain coefficient values have been evaluated from ground calibration data at three fixed temperatures of the detector and may be periodically updated using the results of the flight calibration data analysis. The charge loss correction is applied to events with grades 1-8 using the information stored in the input CALDB Charge Loss Correction and Charge Loss Correction Filter files.

The PI calculation is the result of an interpolation on temperature and on time. First, for each event the two rows valid for the epochs closest to the time of the event are chosen. Then for each row, given the temperature associate to the event, the task performs a temperature interpolation of the gain coefficients between the two nearest temperatures, thus obtaining two sets of coefficients for two contiguous times. Finally, a second interpolation with respect to time between these two sets of coefficients is performed.

The temperature of each event is read from the input Housekeeping file (input parameter *'hkfile'*). Optionally, by setting *'hkfile'* to 'NONE', the temperature associated to all the events is read from the input parameter *'temperature'*.

The unit of the 'PI' and 'SURRPI' columns is set to 40 eV per channel.

Usage Examples:

1. Calculate PI values for the input event file 'nu10012001002A_uf.evt'. The detector temperature associated to each event is read from the input file 'nu10012001002A.hk'. The results are written in the output event file 'nu10012001002A_out.evt'.

```
> nucalcpi infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
  hkfile=nu10012001002A_fpm.hk
```

3.8 Events Flagging (*'nuflagevt'*)

The NuSTARDAS software module *'nuflagevt'* flags the events of NuSTAR event files according to various conditions.

The task requires in input the following files:

- Level 1/1a Event File (*infile*)
- CALDB Depth Cut File (*depthcutfile*)
- CALDB Event Cut File (*eventcutfile*)

The files generated in output by the task are:

- Calibrated Level 1a Event File (*outfile*)

The *'nuflagevt'* task first flags the events on the basis of their depth of interaction in the detectors, accounting for the total absorption probability in CZT along the direction of the original photon. The aim is to flag the internal detector background events thus improving the S/N.

The depth cut profiles are read from the input CALDB file *'depthcutfile'* which stores, for each pixel of the detectors, three energy depth cut channels 'E1', 'E2' and 'E3' as a function of the event energy. For each event, *'nuflagevt'* selects the CALDB 'E1', 'E2' and 'E3' values corresponding to its position on the detector and energy ('RAWX', 'RAWY' and 'PI' columns on the input event file) and flags it according to the following criteria:

- `SURRPI ≤ E1  (if E2,E3 > 0)`
- `E2 < SURRPI ≤ E1 (if E2,E3 < 0)`
- `SURRPI ≤ E3  (if E2,E3 < 0)`

where 'SURRPI' is a column read from the input event file and stores the sum of the energies of the un-triggered surrounding pixels, i.e. the pixels in the 3x3 array which are below the software threshold.

Next, the task flags events according to the values of the 'PRIOR', 'RESET', 'PREPHAS[5]', and 'PI' columns of the input event file. Specifically, "baseline", "prior/reset", "prior", "reset" and "energy" cuts are applied using specific conditions on the values of these columns as detailed below:

1. "Baseline cut":
   - `(BASELINE < BASELINE1 and PI < PI_BASELINE) or (BASELINE > BASELINE2)`
2. "Prior/reset cut" (FPMA only):
   - `(PRIOR2 < PRIOR < PRIOR3 and PI < PI_PRIOR) or`
   - `(RESET2 < RESET < RESET3 and PI1_RESET < PI < PI2_RESET)`
3. "Prior cut":
   - `PRIOR < PRIOR1`
4. "Reset cut":
   - `RESET < RESET1`
5. "Energy cut":
   - `PI < 0 or PI > 4095`

where the threshold values *BASELINE1, PI_BASELINE, BASELINE2, PRIOR1, PRIOR2, PRIOR3, PI_PRIOR, RESET1, RESET2, RESET3, PI1_RESET, PI2_RESET* are read from the input CALDB file 'evtcutfile'.

As in the case of the *'nuflagbad'* module, the task updates the 'STATUS' column of the event file. The values of the 'STATUS' column used by *'nuflagevt'* are the following:

- b0000000010000000 Event fails depth cut
- b0000000100000000 Event fails baseline cut
- b0000001000000000 Event fails prior/reset cut
- b0000010000000000 Event fails prior cut
- b0000100000000000 Event fails reset cut
- b0001000000000000 Event with PI out of range

Usage Examples:

1. Flag events in the NuSTAR event file named nu10012001002A_uf.evt. The 'STATUS' column of the output event file 'nu10012001002A_out.evt' is updated accordingly.

```
> nuflagevt infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
```

3.9 Coordinates Transformation (*'nucoord'*)

The NuSTARDAS software module *'nucoord'* converts the coordinates of each event from the telemetry raw sub-detector (hybrid) values into celestial sky coordinates using information from the Mast Aspect Solution File and the Spacecraft Attitude File.

The following four coordinates systems are defined (event files columns names are indicated in parenthesis):

- (RAWX, RAWY): telemetry sub-detector (hybrid) coordinates (pixel size = 12.3'')
- (DET1X, DET1Y): Focal Plane Bench Frame (FB) detector coordinates (pixel size = 2.46'')
- (DET2X, DET2Y): Optics Bench Frame (OB) detector coordinates (pixel size = 2.46'')
- (X, Y): Celestial Sky Coordinates (pixel size = 2.46'')

The *'nucoord'* task also calculates, as a function of time, the sky positions of the telescope optical axis, of the aperture stop center and of a reference pixel of the detector.

The task requires in input the following files:

- Level 1/1a Event File (*infile*)
- Mast Aspect Solution File (*mastaspectfile*)
- Spacecraft Attitude File (*attfile*)
- CALDB Pixel Location File (*pixposfile*)
- CALDB Spacecraft Alignment File (*alignfile*)
- CALDB Telescope Definition File (*teldef*)

The files generated in output by the task are:

- Calibrated Level 1a Event File (*outfile*)
- Optical Axis File (*optaxisfile*)
- DET1 Reference Pixel File (*det1reffile*)

The *'nucoord'* module is a script that runs in sequence the software modules *'nucalcpos'*, *'coordinator'* and *'nuskypos'*. These modules are discussed in the following sub-sections.

3.9.1 Detector Coordinates Calculation (*'nucalcpos'*)

The first module run by '*nucoord*' is the *'nucalcpos'* task to calculate the detector coordinates in the detectors Focal Plane Bench Frame ('DET1') and in the Optics Bench Frame ('DET2') starting from the telemetry sub-detector coordinates ('RAW' system).

The task requires in input the following files:

- Level 1/1a Event File (*infile*)
- Mast Aspect Solution File (*mastaspectfile*)
- CALDB Pixel Location File (*pixposfile*)
- CALDB Spacecraft Alignment File (*alignfile*)

The files generated in output by the task are:

- Calibrated Level 1a Event File (*outfile*)

The *'nucalcpos'* task reads the 'RAWX' and 'RAWY' columns of the input event file storing the original telemetry coordinates of the events for each of the four focal plane module sub-detectors (*'DET0'*, *'DET1'*, *'DET2'* and *'DET3'*). For each sub-detector, the range of the 'RAWX' and 'RAWY' columns is 0-31.

As a first step, the task transforms the RAW coordinates of the four sub-detectors in a new common coordinate system referred to the detectors Focal Plane Bench Frame (FB). This first step includes a change of the pixel size (by a factor of 1/5) and a correction of the detector pixels positions using the information from the input CALDB pixel location file (parameter *'pixposfile'*).

The new coordinates values are saved in two new columns ('DET1X' and 'DET1Y') of the output event file. The DET1 pixel size is 12.3/5=2.46 arc seconds and the column range is 1-360. The coordinates transformation between the 'RAW' and the 'DET1' system is illustrated in Figure 3.

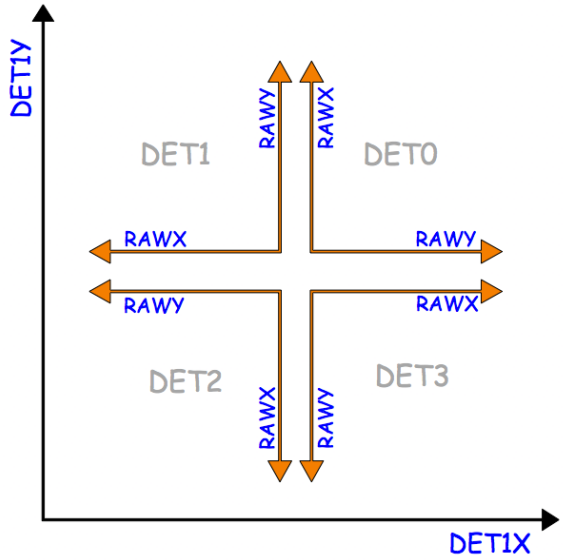

Figure 3: Relationship between the raw telemetry coordinates of the four sub-detectors (coordinates 'RAWX', 'RAWY') and the Focal Plane Bench Frame (coordinates 'DET1X', 'DET1Y').

Next, the task uses the time dependent roto-translations stored in the Mast Aspect Solution file to calculate the detector coordinates in the Optics Bench Frame (OB). These coordinates are written in two new columns (‘DET2X’ and ‘DET2Y’) of the output event file.

This transformation enables the correction of coordinates for the movement of the deployed mast connecting the detectors Focal Plane Bench to the Optics Bench. The coordinates transformation between the 'DET1' and the 'DET2' systems is illustrated in Figure 4.

The DET2 pixel size is the same as the DET1 system size (2.46 arc seconds) while the column range is 1-600. Additional details on the algorithm can be found in the help file of the task.

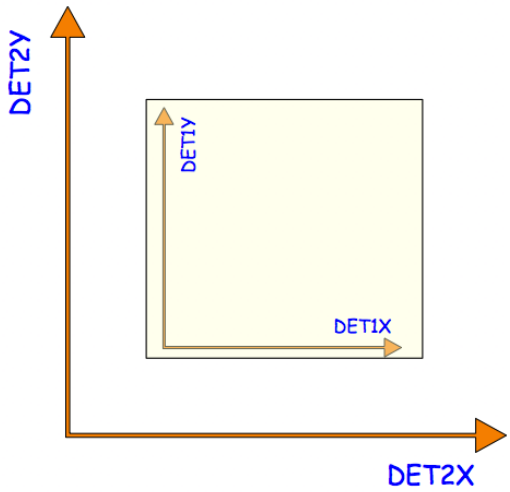

Figure 4: Relationship between the detector coordinates in the Focal Plane Bench frame (columns 'DET1X','DET1Y') and in the Optics Bench frame (columns 'DET2X', 'DET2Y').

3.9.2 Sky Coordinates Calculation (*'coordinator'*)

The second module run by '*nucoord*' is the *'coordinator'* FTOOL. This is a multi-mission tool distributed by the HEASARC as part of the HEASoft package. For NuSTAR it is used to convert the 'DET2' coordinates in 'SKY' celestial coordinates taking into account the spacecraft attitude information.

The task requires in input the following files:

- Level 1/1a Event File (*eventfile*)
- Spacecraft Attitude File (*attfile*)
- CALDB Telescope Definition File (*teldef*)

The SKY celestial coordinates are stored in two new columns (‘X’ and ‘Y’) of the event file. The 'X' and 'Y' column range is 1-1000.

3.9.3 Optical Axis, Aperture Stop Center and Detector Reference Pixel Sky Positions (*'nuskypos'*)

The third module run by 'nucoord' is the NuSTARDAS task *'nuskypos'* that calculates, as a function of time, the sky positions of the telescope optical axis, of the aperture stop center and of a reference pixel of the detector in the DET1 coordinate system.

The task requires in input the following files:

- Mast Aspect Solution File (*mastaspectfile*)
- Spacecraft Attitude File (*attfile*)
- CALDB Spacecraft Alignment File (*alignfile*)
- CALDB Telescope Definition File (*teldef*)

The files generated in output by the task are:

- Optical Axis File (*optaxisfile*)
- DET1 Reference Pixel File (*det1reffile*)

As a first step, the *'nuskypos'* task reads the position of the optical axis in the DET2 frame from the input CALDB spacecraft align file (parameter *'alignfile'*) and transforms it in the SKY frame. This allows the motion of the optical axis to be tracked in the SKY coordinate system and it is used to calculate the off-axis angle of a given celestial source by the *'numkarf'* task during the Stage 3. The SKY position of the optical axis as a function of time is stored in the output file *'optaxisfile'*.

The 'nuskypos' module also calculates the DET2 and SKY coordinates of the center of the aperture stop. The location of the aperture stop center in the DET1 frame is stored in the CALDB *'alignfile'*. The task first reads this position and converts the Focal Plane Bench Frame (DET1) coordinates into the Optics Bench Frame (DET2) detector coordinates using the mast aspect solution file (parameter *'mastaspectfile'*). Afterward, the DET2 coordinates are transformed in SKY coordinates using the input spacecraft attitude and teldef files. The DET2 and SKY positions of the aperture stop center are stored in the output file *'optaxisfile'*.

Finally, *'nuskypos'* calculates, as a function of time, the SKY coordinates of a reference pixel in the DET1 frame using information from the mast aspect solution file and the spacecraft attitude file. This information is used by the *'nuexpomap'* task to generate the exposure maps. The SKY position of the detector reference point as a function of time is stored in the output file *'det1reffile'*. Additional details about the algorithm can be found in the help file of the task.

Usage Examples:

1. Convert the RAW coordinates of the input event file 'nu10012001002A_uf.evt' to DET1 and DET2 coordinates using the pixel location file stored in the CALDB. The DET1X/DET1Y and DET2X/DET2Y coordinates are written in the output file 'nu10012001002A_out.evt'.

```
> nucalcpos infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
  mastaspectfile=nu10012001002_mast.fits
```

2. Calculate the SKY positions of the optical axis file and of the (350,350) DET1 reference pixel and store them in the output files 'nu10012001002A_oa.fits', 'nu10012001002A_det1.fits', respectively.

```
> nuskypos mastaspectfile=nu10012001002_mast.fits instrument=FPMA
  attfile=nu10012001002_att.fits pntra=258.3014 pntdec=-23.7678
  alignfile=nuCalign20100101v007.fits teldef=nuA20100101v002.teldef
  optaxisfile=nu10012001002A_oa.fits det1reffile=nu10012001002A_det1.fits
```

3. Convert the RAW coordinates of the input event file 'nu10012001002A_uf.evt' to DET1, DET2 and SKY coordinates using the pixel location file stored in the CALDB. The DET1X/DET1Y, DET2X/DET2Y and X/Y coordinates are written in the output file 'nu10012001002A_out.evt'. The optical axis file and the DET1 reference file are also generated.

```
> nucoord infile=nu10012001002A_uf.evt outfile=nu10012001002A_out.evt
  mastaspectfile=nu10012001002_mast.fits attfile=nu10012001002_att.fits
  pntra=258.3014 pntdec=-23.7678 optaxisfile=nu10012001002A_oa.fits
  det1reffile=nu10012001002A_det1.fits
```

4 EVENTS SCREENING AND EXPOSURE MAPS GENERATION (STAGE 2)

This chapter describes the Stage 2 of the NuSTAR data reduction devoted to the screening of event files by applying cleaning criteria on specified attitude/orbital/instrument parameters and event properties. Stage 2 also includes the generation of sky exposure maps.

The inputs of the screening stage are the Level 1a calibrated event files. Stage 2 produces in output the Level 2 calibrated and cleaned event files to be used for scientific analysis. In the following the Stage 2 NuSTAR software modules are presented in the same order used for the standard data processing along with software usage examples. These processing steps are coded in the second stage of the *'nupipeline'* script and cases of how to run the pipeline are illustrated in the next section.

The list of the Stage 2 software modules is the following:

- *nucalcsaa* – Calculation of the SAA passages
- *nufilter* - Filter file generation.
- *nuscreen* - Good Time Intervals generation and event file screening.
- *nulivetime* - Event files dead time correction.
- *nuexpomap* - Sky exposure maps generation.
- *nuskytodet* – Calculation of the DET1 and DET2 coordinates from SKY coordinates.
- *nusplitsc* – Split NuSTAR Spacecraft Science 06 cleaned event files.

4.1 Standard Data Processing with *'nupipeline'* (Stages 1 and 2)

The data reduction steps described in this chapter are coded in the *'nupipeline'* script. The majority of the input parameters of the single software modules can also be set directly running the *'nupipeline'* and appropriate default values have been set for most of them.

By setting the input parameter *'exitstage=2'* (default value), the *'nupipeline'* script allows calibration (stage 1) and screening (stage 2) of the data using a single command line. Both data from the FPMA and the FPMB modules are processed by default. The standard command to be used is:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out
```

Note that in this command line it is not necessary to specify *'exitstage=2'* since *'2'* is the default value of this parameter. All the 'nupipeline' output files are generated in a directory named *"pipeline_out".*

By providing in input a GTI file through the input parameter '*usrgtifile'* the '*nupipeline'* applies the corresponding temporal filter, including the livetime correction, to the output calibrated and cleaned event files:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out usrgtifile=myGTI.fits
```

We stress that if a temporal filter is applied using the XSELECT package to the calibrated and cleaned event file, the output filtered event file is not corrected for the rate-dependent instrument livetime. **In this case it is necessary to update the EXPOSURE and LIVETIME keywords of the output event file by running the '*nulivetime'* software module.**

To generate also the sky exposure maps the input parameter *'createexpomap'* must be set to *'yes'* (default value is *'no'*) as illustrated in the following command line:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out createexpomap=yes
```

To split of the 06 cleaned event files (see Section 4.10) the input parameter *'runsplitsc'* must be set to *'yes'* (default value is *'no'*) as illustrated in the following command line:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out runsplitsc=yes
```

4.2 Rationale

The calibrated Level 1a event files can include "bad" events and/or *"bad"* time intervals that need to be excluded before the files can be used for the scientific data analysis. This chapter lists the criteria and the methods that are used to screen the data.

The NuSTAR Calibration Team has defined the standard screening criteria used to produce the cleaned Level 2 event files (see Section 4.7). Starting from the Level 2 files, high-level scientific products (sky images, light-curves and energy spectra) can be extracted. We will see that it is also possible to apply a different data screening which may be more or less conservative than the standard one.

4.3 SAA passages calculation (*'nucalcsaa'*)

The NuSTARDAS software module '*nucalcsaa'* calculates South Atlantic Anomaly (SAA) passages of the NuSTAR observatory.

The task requires in input the following files:

- Housekeeping file (*hkfile*)
- Level 1a event file (*evtfile*)
- Orbit file (*orbitfile*)
- SAA parameter file (*saaparfile*)

The file generated in output by the task is:

- Housekeeping file (*outfile)*

The '*nucalcsaa'* module identifies the SAA passages with three alternative algorithms (named '1', '2' and '3') that can be set by the user through the input parameter *'saacalc'*. For all types of calculations two different SAA

calculation modes are implemented ('STRICT' and 'OPTIMIZED') and the user can choose the mode through the input parameter *'saamode'*.

In the 'STRICT' mode the SAA passages are identified monitoring the level of the high-gain (low-gain) shield single rates stored in the 'SHLDHI' ('SHLDLO') column of the input housekeeping file for the '1' ('2' and '3') SAA calculation algorithm.

In the 'OPTIMIZED' mode the SAA passages found in the 'STRICT' mode are refined by requiring the presence of an increase of the CZT detector event count rates simultaneous to the observed shield single rates increase. The results for both SAA removal modes are written in a new column named 'SW_SAA' of the output housekeeping file.

In addition, by setting the input parameter '*tentacle'* to *'yes'*, the task allows identification and flagging of time intervals in which the CZT detector events count rates show an increase. This search is done only when the spacecraft is entering into the SAA region and a simultaneous shield singles rates increase is not required. The results are written in a new column named 'SW_TENTACLE' of the output housekeeping file. As for the 'STRICT' and 'OPTIMIZED' mode, also the 'tentacle' algorithm is different in the three *'nucalcsaa'* calculations ('1', '2' or '3').

The various numerical values used by the algorithm are stored in a specific CALDB file provided in input through the parameter '*saaparfile*'.

By default (*'saacalc=3'* and *'eliminatesource=yes'*), the CZT detector count rates are evaluated after the exclusion of the contribution of the brightest sources in the field of view. Note that for the case *'saacalc=1'* the algorithm has been tested only for steady sources with fluxes not dominating over the detector background and no elimination of the target source is done.

The *'tentacle'* option should be used only when the background count rate is not stable and shows time intervals of increased intensity. The time intervals identified by the task should be carefully checked by the user.

Usage example:

1. Calculate the 'SW_SAA' column using the 'OPTIMIZED' SAA calculation mode for *'saacalc=3'* (default). The new column is stored in the output file 'nu10012001002A_fpm_out.hk':

```
> nucalcsaa hkfile=nu10012001002A_fpm.hk orbitfile=nu10012001002_orb.fits
  evtfile=nu10012001002A_uf.evt outfile=nu10012001002A_fpm_out.hk
  saamode=OPTIMIZED tentacle=no
```

4.4 Filter File Generation (*'nufilter'*)

The NuSTARDAS software module *'nufilter'* creates a filter file, which is used for the data screening by the *'nuscreen'* module (see next section 4.5). The filter file contains the attitude and orbit derived quantities, the instrument housekeeping parameter values, and the quality of the reconstructed mast position as a function of time.

The task requires in input the following files:

- Housekeeping file (*hkfile*)
- CEB Housekeeping (*cebhkfile*)
- Attitude file (*attfile*)
- Position Sensing Detector file (*psdcorfile*)
- Orbit file (*orbitfile*)
- Makefilter configuration file (*mkfconfigfile*)
- NORAD Two Line Elements (TLEs) file (*tlefile*)
- Telescope definition file (*teldef)*
- 'Prefilter' configuration file (*preconfigfile*)
- Leap seconds file (*leapfile*)
- AtFunction rigidity file (*rigfile*)

The file generated in output by the task is:

- Filter file (*mkffile*)
- Attitude-orbit file (*attorbfile*)

In the first step, *'nufilter'* derives attitude and orbit related quantities by running the multi-mission '*prefilter'* FTOOL. The satellite attitude information included in the attitude file is interpolated and the NORAD Two Line Elements (TLEs) are propagated to determine satellite ephemeris quantities. This information is used to calculate quantities such as Elevation Angle, Bright Earth Angle, Sun Angle, Cut-off rigidity, etc. The output file of '*prefilter*' (parameter '*attorbfile*') contains the set of columns defined in the CALDB file '*preconfigfile*'.

In the second main step, the *'nufilter'* creates the filter file by running the multi-mission FTOOL *'makefilter'*. The housekeeping information is read from the input orbit file (parameter *'orbitfile'*), the housekeeping files (parameters *'hkfile'* and *'cebhkfile'*) and the spacecraft attitude file (parameter *'attfile'*). The quality of the reconstructed mast position is read from the *'*METGRID_FLAG*'* column of the position sensing detector file (parameter *'psdcorfile').* The *'makefilter'* configuration file, provided in input through the parameter *'mkfconfigfile'*, is a FITS file (by default read from the CALDB) which contains for each parameter the following information: the parameter name, the name of the FITS file, the name of the extension containing the parameter, the interpolation method, the calibration method, the output parameter name and comments for the corresponding keyword in the output filter file.

The interpolation method is used when the value of a specific parameter is not present at given times; the calibration method is applied when some simple numerical manipulations on the input HK parameters is needed. Currently, the default of the interpolation method is set to copy the last known value of that parameter. The calibration method has not been implemented yet. See also the *'makefilter'* help file for further details.

Usage example:

1. Create the output 'prefilter' file named nu10012001002A.attorb and a filter file named nu10012001002A.mkf in the output directory *'pipeline_out'*:

```
> nufilter hkfile=nu10012001002A_fpm.hk attfile=nu10012001002_att.fits
  orbitfile=nu10012001002_orb.fits cebhkfile=nu10012001002A_ceb.hk
  psdcorfile=nu10012001002A_psdcorr.fits outdir=./pipeline_out pntra=258.3014
  pntdec=-23.7678 tlefile=NUSTAR_TLE_ARCHIVE.txt.2013042
```

4.5 Good Time Intervals Generation and Data Screening (*'nuscreen'*)

The NuSTARDAS software module *'nuscreen'* filters calibrated event files by applying cleaning criteria on specified orbital, attitude and instrument parameters. To this end, specific Good Time Intervals files (GTIs) based on attitude/orbit and instrument housekeeping parameters are generated from the filter file. A data screening based on event properties (e.g. the charge pattern) is also applied to produce the output Level 1 cleaned science event files.

The task requires in input the following files:

- Level 1 event file (*infile*)
- Filter file (*mkffile*)
- Housekeeping Range file (*hkrangefile*)
- Events Range file (*evtrangefile*)
- Housekeeping file (*hkfile*)
- User GTI file (*usrgtifile*)

The files generated in output by the task are:

- Level 2 science event file (*outfile*)
- Good Time Interval file (*gtifile*)

The *'nuscreen'* task i) generates a GTI file based on attitude and/or instrument housekeeping parameters and quality of the reconstructed mast position, ii) screens the data using these GTIs, iii) screens events using a GRADE filter and/or a selection on the STATUS column, iv) applies the dead time correction to the temporal keywords of the screened event files.

The GTIs are calculated considering two different sets of parameters, one related to the satellite attitude and orbit (*'createattgti=yes'*) and the other related to instrument housekeeping (*'createinstrgti=yes'*). The GTI file is created by running the multi-mission FTOOL *'maketime'* and contains the time intervals where events are considered good for science data analysis.

Setting the input parameter *'gtiexpr'* to 'DEFAULT' and *'hkrangefile'* to *'CALDB'*, the attitude and instrument housekeeping parameters screening expressions are built using the standard criteria contained in the Housekeeping Calibration files, which are dependent on the observing mode selection provided in input through the parameter '*obsmode*'. The supported observing modes are SCIENCE, OCCULTATION, SLEW, SAA, CALIBRATION and SCIENCE_SC (see Section 2.1). It is possible to supply attitude and/or instrument non-standard criteria through the parameter *'gtiexpr'* providing a specific Boolean expression.

The calculated GTIs are used to screen the events by setting the parameter *'gtiscreen'* to *'yes'* (default value). An additional user created GTI file can also be provided in input by using the *'usrgtifile'* parameter.

'nuscreen' filters out the events previously flagged as "bad" in the 'STATUS' column of the event file (i.e. bad/hot pixels). Moreover, the task applies a grade selection by setting the *'evtscreen'* parameter to *'yes'*. The standard screening criteria for GRADE and STATUS are defined in the Event Range Calibration file and to use these the parameter *'evtrangefile'* is set to CALDB and *'gradeexpr'* and *'statusexpr'* parameters are set to *'DEFAULT'*. Non-standard screening criteria can be specified using the parameters *'gradeexpr'* and *'statusexpr'*. The first is for the selection on the GRADE column and values can be input as a range or a single number (e.g. *'gradeexpr=0-4'* to select GRADE range between 0 and 4; *'gradeexpr=0'* to select only GRADE equal to 0). These inputs are used with the 'filter grade' command in *'xselect'*. The second is for the selection on the STATUS column and the value should be input as a Boolean expression, e.g. *expr="STATUS==b0"* to select only good events (see Sections 3.4, 3.5 and 3.8 for the definition of the values in the STATUS column).

For the SCIENCE observing mode ('obsmode=SCIENCE'), in the case of the parameter 'gtiexpr=DEFAULT', the task also calculates exposure time (not livetime corrected) during which the metrology laser spots are out of the calibrated PSD grid. The value of the exposure time is printed in the processing log of the task as follows: *"Info: Metrology laser spots out of calibrated PSD grid for ONTIME=X seconds".* During these time intervals, in the case of a standard metrology data processing (*'nupipeline'* input parameter 'psdcal=yes'), the events are filtered out by *'nuscreen'*. A procedure to recover this lost exposure time is given in Chapter 6.

The parameters *'gtiexpr'* and *'statusexpr'* accept the expression directly from the command line or written into a text file and input by preceding the filename with '@' (e.g. statusexpr=@file.txt). The expression in the file can be arbitrarily complex and can extend over multiple lines of the file. Lines that begin with 2 slash characters (//) are ignored and can be used to add comments.

If all the screening parameters are set to 'yes', the output events file contains only good events and the GTI extension is updated.

'nuscreen' also applies the dead time correction to the temporal keywords of the output screened event files by running the '*nulivetime'* task (see Section 4.6).

Usage Example:

1. Generate the cleaned event file nu10012001002A01_cl.evt (DEFAULT name) and the GTI FITS file nu10012001002A01_gti.fits (DEFAULT name) using the screening expression to create the GTI written in the Housekeeping Range Calibration file and the events selection criteria defined in the Events Range Calibration file:

```
> nuscreen obsmode=SCIENCE infile=nu10012001002A_uf.evt gtiscreen=yes
  evtscreen=yes gtiexpr=DEFAULT gradeexpr=DEFAULT statusexpr=DEFAULT
  createattgti=yes createinstrgti=yes outdir=./pipeline_out
  hkfile=nu10012001002A_fpm.hk mkffile=nu10012001002A.mkf outfile=DEFAULT
```

4.6 Dead Time Correction (*'nulivetime'*)

The NuSTARDAS software module '*nulivetime'* calculates the dead time corrections necessary to update the temporal keywords of the screened NuSTAR event files. It is run automatically by the *'nuscreen'* task (see previous section).

The task requires in input the following files:

- Level 2 event file (*infile*)
- Housekeeping file (*hkfile*)

The file generated in output by the task is:

- Level 2 event file with updated temporal keywords (*outfile*)

'nulivetime' applies the dead time corrections and updates the temporal keywords of the screened NuSTAR event files. The dead time correction is calculated using information stored in the input file *'hkfile'* and written in the keyword 'DEADC' of the output event file. *'nulivetime'* also updates the 'EXPOSURE' and 'LIVETIME' keywords of the output event file.

Usage example:

1. Apply dead time corrections to the input cleaned event file 'nu10012001002A01_cl.evt'.

```
> nulivetime infile=nu10012001002A01_cl.evt hkfile=nu10012001002A_fpm.hk
  outfile=NONE
```

4.7 Screening criteria

In this section we describe the screening criteria associated with i) attitude, orbital and/or instrumental parameters, ii) quality of the reconstructed mast position and iii) specific events properties which are applied by the *'nuscreen'* module to produce Level 2 cleaned event files.

The values of the attitude, orbital and instrumental parameters and the parameter related to the quality of the reconstructed mast position (see next sections), as a function of time are included in the Filter File (*.mkf) and are used to generate Good Time Intervals (GTIs) for which their values are within specified ranges.

4.7.1 Screening Criteria associated with the attitude and orbital parameters

One set of screening criteria is obtained by considering parameters related to the satellite position and the telescopes boresight direction. These parameters are part of the standard screening in *'nuscreen'* (and in the *'nupipeline'* script). Non standard screening expressions can be set through the parameter *'gtiexpr'* (see examples in Section 4.8). The Filter File columns that can be used for the GTIs generation are:

- **SAA_A**, **SAA_B**: South Atlantic Anomaly (for focal plane module A and B, respectively) flags with value '0' (observatory outside the SAA) or '1' (observatory in the SAA);
- **SW_SAA, SW_TENTACLE:** refined South Atlantic Anomaly flags as evaluated by the *'nucalcsaa'* module (see Section 4.3), with value '0' (observatory outside the SAA) or '1' (observatory in the SAA);
- **ELV:** elevation angle between the telescopes boresight and the Earth limb (degrees);
- **SLEW**: flag with value '0' (observatory not slewing) or '1' (observatory performing a slew between two targets);
- **SOURCE**: indicates the origin for the reconstruction of the attitude file ('1' = attitude from the star tracker on the Optics Bench "CHU4", used for scientific analysis; '2' = attitude from spacecraft star trackers "CHU123");
- **RESIDUAL:** quality of the attitude reconstruction from the "CHU4" star tracker on the Optics Bench.

4.7.2 Screening Criteria associated with the mast position

For the NuSTAR observatory, a good reconstructed mast position is crucial to evaluate accurate sky coordinates for each event. The Position Sensing Detectors (PSDs) are two optical detectors located in the focal plane to monitor and evaluate the mast position as a function of the time. The PSDs are calibrated only in a specific grid, which is stored in the CALDB metrology grid file. Occasionally, due the mast movements, the laser spots can be outside this calibrated grid and in this case the sky coordinates are not calculated. As a consequence, the corresponding time intervals must be excluded from the GTIs. A specific column computed by the *'numetrology'* task, named 'METGRID_FLAG', of the Filter File provides information on the metrology grid data. This column is used for the GTIs generation and its values are defined as follows:

- **METGRID_FLAG**: the value of this parameter is '0' (laser spots within the calibrated grid), '1' (laser spots out of the calibrated grid) or '2' (PSD X/Y coordinates not linearized)

4.7.3 Screening Criteria associated with the instrumental parameters

Screening on specific instrument parameters is done to ensure that data are always included within certain specific ranges. The *'nuscreen'* software module can handle screening criteria to define GTIs according to specific instrumental housekeeping parameter range values. The instrumental parameters and the optimized range of the Filter File instrumental columns used for the GTIs generation are:

- **CAL0STAT**: flag with value '0' (on-board calibration sources not in FPMA FOV) or '1' (on-board calibration sources in FPMA FOV);
- **CAL1STAT**: flag with value '0' (on-board calibration sources not in FPMB FOV) or '1' (on-board calibration sources in FPMB FOV);
- **LIVETIME:** exposure corrected for the dead time.

4.7.4 Standard Screening Expressions for attitude/orbital/instrumental parameters

The unfiltered NuSTAR Level 1a science event files can be divided in up to six different Level 2 cleaned event files, according to the standard screening on the values of attitude/orbital/instrumental parameters.

Each cleaned event file stores the data taken in one of the supported observational modes (see also Section 2.1) as detailed below:

1. **SCIENCE (01)**: data from the science target with good attitude reconstruction. The following screening expressions are used:
   - `ELV>3 && SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=0 && SOURCE=1 && RESIDUAL<7 && CAL0STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMA)`
   - `ELV>3 && SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=0 && SOURCE=1 && RESIDUAL<7 && CAL1STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMB)`

2. **OCCULTATION (02)**: data during the Earth occultation. The following screening expressions are used:
   - `ELV<=3 && SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=0 && CAL0STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMA)`
   - `ELV<=3 && SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=0 && CAL1STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2)(for FPMB)`

3. **SLEW (03)**: data taken during a spacecraft slew. The following screening expressions are used:
   - `SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=1 && CAL0STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMA)`
   - `SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=1 && CAL1STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMB)`

4. **SAA (04)**: data taken during a SAA passage. The following screening expression is used:
   - `(SAA_A=1||SAA_B=1||SW_SAA=1||SW_TENTACLE=1) && (LIVETIME=0 && (METGRID_FLAG=0||METGRID_FLAG=2)) (for FPMA and FPMB)`

5. **CALIBRATION (05)**: data when the on-board calibration source is in the Field Of View. The following screening expressions are used:
   - `SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && CAL0STAT=1 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMA)`
   - `SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && CAL1STAT=1 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMB)`

6. **SCIENCE_SC (06)**: data from the science target with poor attitude reconstruction. The following screening expressions are used:
   - `ELV>3 && SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=0 && SOURCE=2 && CAL0STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMA)`
   - `ELV>3 && SAA_A=0 && SAA_B=0 && SW_SAA=0 && SW_TENTACLE=0 && SLEW=0 && SOURCE=2 && CAL1STAT=0 && LIVETIME>0 && (METGRID_FLAG=0||METGRID_FLAG=2) (for FPMB)`

4.7.5 Screening Criteria associated with the event proprieties

During data screening additional selections, associated with characteristics of the events themselves, are applied.

The current NuSTAR screening criteria are:

- Removal of events on bad and hot/flickering pixels (*'nuflagbad'* and *'nuhotpix'*, see Sections 3.4 and 3.5)
- Removal of events failing the depth of interaction cut (see *'nuflagevt',* see Section 3.8)
- Removal of events failing the *'*baseline*'*, *'*prior*'* and *'*reset*'* cuts (see *'nuflagevt'* see Section 3.8)
- Removal of events with PI values out of range (*'nuflagevt',* see Section 3.8)
- Grade selection (*'nucalcpha',* see Section 3.6)

These criteria are applied to the data using the information stored in the 'STATUS' column of the event file storing the type of flag according to the following definitions:

- `b0000000000000000 Good event`
- `b0000000000000001 Event falls in a bad pixel from on-ground CALDB Bad Pixel File`
- `b0000000000000010 Event falls in a bad pixel from on-board disabled pixel`
- `b0000000000000100 Event falls in a bad pixel from user bad pixel file`
- `b0000000000001000 Event has a neighbor bad from bad/disabled/user pixel list`
- `b0000000000010000 Event falls in a detector edge pixel`
- `b0000000000100000 Event falls in a hot/flickering pixel`

- `b0000000001000000` Event has a neighbor hot/flickering pixel
- `b0000000010000000` Event fails depth cut
- `b0000000100000000` Event fails baseline cut
- `b0000001000000000` Event fails prior/reset cut
- `b0000010000000000` Event fails prior cut
- `b0000100000000000` Event fails reset cut
- `b0001000000000000` Event with PI out of range

The standard expression used to filter out for bad events is:

```
STATUS = b000000000x0xx000
```

This expression excludes from the event file all the flagged events, except i) events having a neighbor bad from bad/disabled/user pixel list, ii) events falling in a detector edge pixel and iii) events having a neighbor hot/flickering pixel.

In addition to the selection on the 'STATUS' column, the events are filtered according to the value of the 'GRADE' and ‘SHIELD’ columns. A single X-ray event can in fact be spread in more than one detector pixel (up to 4 contiguous pixels). The grade is a classification of the shape of X-ray events on the detector. The single pixel events (i.e. grade 0) have better spectral resolution than others. Single or double pixel events are more likely to be X-ray events than three-four pixel events. There are 32 grades defined for NuSTAR (as described in Chapter 3) and their values are stored in the 'GRADE' column of the event file.

The standard grade and shield selection defined by the NuSTAR Calibration Team and stored in the CALDB is:

```
(GRADE ≥ 0 && GRADE ≤ 26) && (SHIELD=0).
```

4.7.6 SCIENCE observational mode: summary.

The observing mode "**SCIENCE"** (01) is the mode used for the extraction of high-level scientific products. The standard screening criteria used to generate cleaned Level 2 "01" event files in the archive for this observing mode is summarized below:

1. Removal of the following time intervals:
    - During a slew of the satellite;
    - When the Earth is in the Field Of View;
    - When the satellite is in the South Atlantic Anomaly;
    - When the angle between telescope boresight and Earth limb is equal or less than 3 deg;
    - When the on-board calibration source is in the Field Of View;
    - When the attitude reconstruction is not derived from the star tracker mounted on the Optics Bench;
    - When the quality of the attitude reconstruction from the star tracker on the Optics Bench is low (RESIDUAL > 7);
    - When either of the laser spots tracking the mast motion are outside the PSD calibrated grid;
    - When the LIVETIME is zero.
2. Removal of bad pixels (dead, hot and flickering), events failing the depth/baseline/prior/reset cuts, events with PI channels out of range.
3. The grade of the event in the range 0-26.

4.8 Examples of how to screen the data with custom selections

The following two equivalent methods allow performing customized screening:

1. By running the *'nuscreen'* module (see Section 4.5) specifying screening criteria via the input parameters *'gtiexpr '*, *'gradeexpr'*, and *'statusexpr'*.

2. By running the Stage 2 of the *'nupipeline'* script. Within *'nupipeline',* the screening criteria are specified by the values of the input parameters *'gradeexpr'*, *'statusexpr'* and *'gtiexpr01'*, *'gtiexpr02'*, *'gtiexpr03'*, *'gtiexpr04'*, *'gtiexpr05'*, 'gtiexpr06'. Note that by default (parameter *'obsmode=ALL'*) the *'nupipeline'* runs the *'nuscreen'* module six times.

4.8.1 Example of how to use *'nuscreen'*

For detailed spectral analysis it could be necessary to optimize the energy resolution. To this aim, users can decide to select only events with grades equal to '0' (single pixel events) starting from the Level 2 event file. The command line to be used is:

```
> nuscreen infile=nu10012001002A01_cl.evt gtiscreen=no evtscreen=yes
  gtiexpr=NONE gradeexpr=0 statusexpr=NONE outdir=./ hkfile=nu10012001002A_fpm.hk
  outfile=nu10012001002A01_cl_grade0.evt
```

In this example the GTIs are not applied, by setting the *'gtiscreen'* parameter to *'no'*, since the input event file was already cleaned. The grade selection is done setting the parameter *'evtscreen'* to yes. The parameter *'gradeexpr'* specifies the values to filter in the column 'GRADE'.

4.8.2 Example of how to use nupipeline

The above example above can also be handled by *'nupipeline'*. In this case both Stage 1 and Stage 2 are executed and the command to be used is:

```
> nupipeline indir=./10012001002 outdir=./pipeline_out steminputs=nu10012001002
  obsmode=SCIENCE instrument=FPMA evtscreen=yes gradeexpr=0
```

4.9 Generation of Exposure Maps (*'nuexpomap'*)

The NuSTARDAS software module *'nuexpomap'* generates Sky Exposure Maps storing, for a given observation, the net exposure time for each sky pixel. The exposure map accounts for detector bad and hot pixels, detector gaps, attitude variations, mast movements and, optionally, telescope vignetting.

The task requires in input the following files:

- Level 2 event file (*infile*)
- Attitude file (*attfile*)
- Mast Aspect Solution file (*mastaspectfile*)
- DET1 Reference Point file (*det1reffile*)
- Pixel Location File (*pixposfile*)
- Alignment File (*alignfile*)
- Telescope Definition File (*teldef*)
- Instrument Probability Map File (*instrprobmapfile*)
- Vignetting File (*vignfile*)

The file generated in output by the task is:

- Exposure Map file (*expomapfile*)

The NUSTARDAS *'nuexpomap*' task first generates a set of instrument maps in Focal Plane Bench DET1 frame where bad/hot pixels and detector gaps are marked using information from the 'BADPIX' extension of the input event file (parameter *'infile'*) and from the input CALDB Instrument Probability Map File (parameter *'instrprobmapfile'*), respectively.

The set of DET1 instrument maps is constructed as follows. First, the task reads the input file *'det1reffile'* storing the X/Y SKY position as a function of time of a reference pixel in the DET1 coordinate system. Second, the X/Y offsets relative to their minimum value in the *'det1reffile'* are calculated. Third, the X/Y offsets are binned with a bin size in pixels specified by the input parameter *'pixbin'*. Each bin defines a time interval during which the spatial fluctuations due to attitude variations and mast movements are below the *'pixbin'* value and are considered constant. This information is then used by *'nuexpomap'* to build the aspect histogram defining, for each spatial offset bin, the time interval and duration of each single DET1 instrument map. By setting the parameters *'offsetfile'* and *'aspecthistofile'* to something other then *'NONE'* the task saves in the current directory the spatial offset and aspect histogram files, respectively.

Each DET1 instrument map is then transformed to the Optics Bench DET2 frame using the mast aspect solution (parameter *'mastaspectfile'*) and the CALDB align file (parameter *'alignfile'*). In this step, *'nuexpomap'* makes use of the multi-mission FTOOLs *'combinexform'* and *'imagetrans'*.

Optionally, if the parameter *'vignflag'* is set to *'yes'*, the task applies the vignetting correction to the DET2 instrument maps using the vignetting CALDB file (parameter *'vignfile'*). The vignetting correction is applied at the energy value specified by the parameter *'energy'*.

Next, the DET2 instrument maps are transformed to the SKY frame using the attitude information provided in input through the parameter *'attfile'*. To this end, the multi-mission FTOOLs *'getxform', 'imagetrans'* and *'getwcs'* are used by *'nuexpomap'*. The total exposure map, storing in each pixel the net exposure time of the observation, is then calculated by summing the single exposure maps in sky coordinates. Dead time corrections are included in the total exposure map.

By setting the parameters *'det1instrfile'*,*'det2instrfile'* and *'skyinstrfile'* to something other then *'NONE'* the task saves in the current directory the single DET1, DET2 and SKY instrument maps as FITS IMAGE extensions in the corresponding output files.

Optionally, the task allows the generation of smaller exposure maps covering a user defined region of the SKY grid, as defined by the input parameters '*skyx*', '*skyy*' and '*skysize*'. In this case *'nuexpomap'* uses the task '*nuskytodet*' to transform *'skyx'* and *'skyy'* in the DET1 and DET2 coordinates, needed to generate the exposure maps.

The *'nuexpomap'* task can also be executed starting from an input DET2 instrument map produced in a previous run, allowing to skip the first processing steps. In this case the DET2 instrument map should be full size (1000x1000 pixels) and not corrected for vignetting. The user should provide the input DET2 map through the parameter 'indet2instrfile'.

Usage Examples:

1. Generate the exposure map nu10012001002A01_ex10.img including the vignetting correction at energy 10 keV (default), starting from the Level 2 event file nu10012001002A01_cl.evt. The command line to be used is:

```
> nuexpomap infile=nu10012001002A01_cl.evt attfile=nu10012001002_att.fits
  mastaspectfile=nu10012001002_mast.fits det1reffile=nu10012001002A_det1.fits
  expomapfile=nu10012001002_ex10.img
```

2. Same as above, but providing in input a DET2 instrument map (not corrected for vignetting):

```
> nuexpomap infile=nu10012001002A01_cl.evt attfile=nu10012001002_att.fits
  indet2instrfile=nu40060001001A01_det2.fits expomapfile=nu10012001002_ex10.img
```

3. The vignetting correction can be applied to the exposure maps at an energy different from the default value, by setting the value of the parameter *'energy'*. In the example below the energy used for the vignetting correction is 30 keV.

```
> nuexpomap infile=nu10012001002A01_cl.evt attfile=nu10012001002_att.fits
  mastaspectfile=nu10012001002_mast.fits det1reffile=nu10012001002A_det1.fits
  expomapfile=nu10012001002_ex30.img energy=30
```

4. The exposure map can be also generated without applying the vignetting correction, starting from the Level 2 event file nu10012001002A01_cl.evt. In this case the command line to be used is:

```
> nuexpomap infile=nu10012001002A01_cl.evt attfile=nu10012001002_att.fits
  mastaspectfile=nu10012001002_mast.fits det1reffile=nu10012001002A_det1.fits
  expomapfile=nu10012001002_ex_no_vig.img vignflag=no
```

4.9.1 Exposure maps generation in 'nupipeline'

The exposure map for the '01' observation mode (i.e. "SCIENCE") can be directly generated when running the *'nupipeline'* script by setting the parameter *'*createexpomap*'* to *'yes'* (default value is *'no'*), as illustrated in the following examples.

Usage examples:

1. The results of *'nuexpomap'* in Example 1 of section 4.9, can be reproduced by running '*nupipeline'* as follows:

```
> nupipeline indir=./10012001002 outdir=./pipeline_out steminputs=nu10012001002
  obsmode=SCIENCE instrument=FPMA createexpomap=yes
```

2. The results of *'nuexpomap'* in Example 3 of section 4.9, can be reproduced by running '*nupipeline'* as follows:

```
> nupipeline indir=./10012001002 outdir=./pipeline_out steminputs=nu10012001002
  obsmode=SCIENCE instrument=FPMA createexpomap=yes energy=30
```

3. The results of *'nuexpomap'* in Example 4 of Section 4.9, can be reproduced by running '*nupipeline'* as follows:

```
> nupipeline indir=./10012001002 outdir=./pipeline_out steminputs=nu10012001002
  obsmode=SCIENCE instrument=FPMA createexpomap=yes expovignflag=no
```

4.10 Splitting of the Spacecraft Science 06 cleaned event files (*'nusplitsc'*)

The NuSTARDAS software module *'nusplitsc'* decomposes the NuSTAR SCIENCE_SC (06) event files into a maximum of seven distinct event files corresponding to distinct combinations of Camera Head Units (CHU) data as derived from the spacecraft bus star trackers (CHU #1, #2 and #3).

The task requires in input the following files:

- Level 2 SCIENCE_SC 06 cleaned event file (*infile*)
- CHU123 Housekeeping file (*chu123hkfile*)
- Instrument Housekeeping file (*hkfile*)

The files generated in output by the task are:

- 06 cleaned event files (up to seven) decomposed according to CHU123 combinations

NuSTAR SCIENCE_SC (06) observing mode events refer to scientific data collected during time intervals in which an aspect solution from the on-board star tracker located on the X-ray optics bench (Camera Head Unit #4, CHU4), the primary method for determining the absolute pointing, is not available. During these time intervals, the aspect reconstruction used to calculate sky coordinates is derived using data from the three star trackers located on the spacecraft bus (Camera Head Units #1, #2, #3, CHU1/CHU2/CHU3). In these cases the accuracy of the sky coordinates degrades to about 2 arcminutes, mainly due to thermal flexing of the spacecraft bus star cameras. Moreover, the three star tracker camera unit data can be integrated in seven different combinations (CHU1 only, CHU2 only, CHU3 only, CHU1&2, CHU1&3, CHU2&3, and CHU1&2&3) that are characterized by a specific aspect reconstruction. Due to the switching between different CHU

combinations, the sky images can then show multiple centroids of the same source precluding an accurate scientific analysis of these data.

The *'nusplitsc'* software module first generates a set of seven GTI files based on CHU1/CHU2/CHU3 housekeeping parameters stored in the input file *'chu123hkfile'*. The GTI files are created by running the multi-mission task *'maketime'*. The GTI selection expressions, corresponding to the seven CHU combinations, are read from the input file *'chu123rangefile'* (CALDB by default). In the second step, the task splits the 06 input cleaned event file applying the calculated GTIs through the multi-mission package *'xselect'*. Finally, *'nusplitsc'* applies the dead time correction to the temporal keywords of the output event files by running the *'nulivetime'* task (see Section 4.6 for additional details).

The event files generated by '*nusplitsc*' need to be carefully checked before the scientific analysis. The user should first set the parameter '*splitmode*' to '*NORMAL*' (default value) and inspect the output images. Some blurring and deformation of the PSF is still to be expected, but if multiple centroids are observed, then the user should rerun '*nusplitsc*' with '*splitmode=STRICT*' to remove some CHU1/CHU2/CHU3 flickering, which can be responsible for the multiple centroids. If there are still multiple centroids present after this run, setting the additional option '*timecut=yes*' (='*no*' default) may remove the remaining multiple centroids. This option can be used together with either '*NORMAL*' or '*STRICT*'. Note that in general '*STRICT*' will remove more data than '*NORMAL*'. The output of '*nusplitsc*' informs the user of the exposure times of the output event files. We stress that if multiple centroids are present in a given CHU1/CHU2/CHU3 combination, the corresponding event file should not be used for the generation of high-level data products (i.e. energy spectra and light-curves).

By default this task applies the stem of the input files to the output files (input parameter *'stemout=DEFAULT'*). The naming convention of the generated event files in the case *'splitmode=NORMAL'* is the following:

```
nu10012001002A06_chu1_N_cl.evt      L2 FPMA 06 event file for CHU1 only
nu10012001002A06_chu2_N_cl.evt      L2 FPMA 06 event file for CHU2 only
nu10012001002A06_chu3_N_cl.evt      L2 FPMA 06 event file for CHU3 only
nu10012001002A06_chu12_N_cl.evt     L2 FPMA 06 event file for CHU1 and CHU2
nu10012001002A06_chu13_N_cl.evt     L2 FPMA 06 event file for CHU1 and CHU3
nu10012001002A06_chu23_N_cl.evt     L2 FPMA 06 event file for CHU2 and CHU3
nu10012001002A06_chu123_N_cl.evt    L2 FPMA 06 event file for CHU1 and CHU2 and CHU3
```

In the case *'splitmode=STRICT'* in the output file names the "`_N_`" string is replaced by "`_S_`".

To extract the high-level data products from the output event files the *'nuproducts'* software module with standard command lines can be used. Note that each CHU1/CHU2/CHU3 combination has a specific aspect reconstruction with a typical scatter of about 2 arcminutes. For this reason, a set of specific spatial source and backgroud extraction regions should be used to extract the scientific products (see Section 6.7 for a specific example).

Usage Examples:

1. Split the input cleaned event file 'nu10012001002A06_cl.evt' according to individual CHU1/CHU2/CHU3 combinations using the *NORMAL* splitting mode. The command line to be used is:

```
> nusplitsc infile=nu10012001002A06_cl.evt chu123hkfile=nu10012001002_chu123.fits
  hkfile=nu10012001002A_fpm.hk
```

2. Split the input cleaned event file 'nu10012001002A06_cl.evt' according to individual CHU1/CHU2/CHU3 combinations using the *STRICT* splitting mode. The command line to be used is:

```
> nusplitsc infile=nu10012001002A06_cl.evt chu123hkfile=nu10012001002_chu123.fits
  hkfile=nu10012001002A_fpm.hk splitmode=STRICT
```

4.11 Transformation of SKY coordinates in DET1 and DET2 coordinates (*'nuskytodet'*)

The NUSTARDAS '*nuskytodet*' calculates as a function of time the Focal Plane Bench Frame coordinates (DET1) and the Optics Bench Frame coordinates (DET2) of an input position in SKY coordinates.

The task requires in input the following files:

- Attitude file (*attfile*)
- Mast Aspect Solution file (*mastapectfile*)
- Alignment file (*alignfile*)
- Telescope definition file (*teldef*)

The file generated in output by the task is:

- SKY Reference Pixel File in DET1 and DET2 system (skydetfile)

'*nuskytodet*' reads the input SKY coordinates from the parameters '*skyxref*' and '*skyyref*'. The transformations from SKY to DET1 and DET2 frames are calculated using information from the attitude file (*'attfile'*), the mast aspect solution file ('*mastaspectfile*') and the two CALDB files '*teldef*' and '*align*'. The DET1 and DET2 values as a function of time are stored in the output file '*skydetfile*'.

Usage Examples:

1. Calculate the DET1 and DET2 coordinates of the SKY pixel (X,Y)=(500,500) and store them in the output file 'nu10012001002A_skydet.fits'.

```
> nuskytodet instrument=FPMA mastaspectfile=nu10012001002_mast.fits
  attfile=nu10012001002_att.fits pntra=258.3014 pntdec=-23.7678
  skydetfile=nu10012001002A_skydet.fits
```

5 EXTRACTION OF PRODUCTS (STAGE 3)

This chapter describes Stage 3 of the NuSTAR data reduction, which is dedicated to the generation of high-level scientific products (light-curves, energy spectra, sky images, ARF and RMF files).

The inputs to Stage 3 are the Level 2 cleaned and calibrated event files and other files generated during the previous processing stages. Stage 3 produces in output the Level 3 data products to be used for scientific analysis.

In the following sections the Stage 3 NuSTAR software modules are presented in the same order used for the standard data processing along with software usage examples. These processing steps are coded in a dedicated module named *'nuproducts'* which constitutes the third stage of the *'nupipeline'* script.

Examples of how to execute Stage 3 automatically for standard data processing are illustrated in the next two sections.

The list of the Stage 3 software modules is the following:

- *nuproducts* - Extract high-level data products from a Level 2 event file
- *nubackscale* - Apply backscale correction to energy spectra (called by *'nuproducts'*)
- *nulccorr* - Apply livetime, PSF/EXPOSURE and vignetting corrections to light-curves (called by *'nuproducts'*)
- *numkarf* - Generates an ARF file for an input PHA file (called by *'nuproducts'*)
- *numkrmf* - Generates an RMF file for an input spatial region file (called by *'nuproducts'*)

The following Level 3 data products are generated during this stage:

- Source energy spectrum
- Background energy spectrum
- Redistribution Matrix File (RMF)
- Ancillary Response File (ARF)
- Source light-curve
- Background light-curve
- Field image in sky coordinates

The level 3 NuSTAR files are FTOOLS compliant and users can perform spectral analysis by using the *XSPEC* package and temporal analysis with the *LCURVE* or *XRONOS* software packages (see `http://heasarc.gsfc.nasa.gov/docs/xanadu/xanadu.html`).

Note that the '*nuproducts*' module makes use of the *XSELECT* multi-mission software package to extract energy spectra and light-curves. The module also applies to them NuSTAR specific corrections (e.g. livetime, backscale and PSF corrections). **It is therefore strongly suggested that energy spectra and light-curves are extracted using the *'nuproducts'* module and not separately using manually the *XSELECT* package.**

5.1 Standard Data Processing with *'nuproducts'* (Stage 3)

To apply the data reduction steps for the extraction of high-level scientific products the *'nupipeline'* task makes use of a dedicated module named *'nuproducts'* (see Section 5.3). The *'nuproducts'* task automatically retrieves all the necessary input files from a directory specified by the input parameter *'indir'* which must be set to the output directory of a previous run of *'nupipeline'* (Stages 1 and 2). An example of a standard command line for a **point-like** source is:

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
```

In this case *'nuproducts'* is run from the same directory used to execute the *'nupipeline'* run (*pipeline_out* in the various command line examples) and generates in the *'products'* directory the high-level data products **for the 01 cleaned event file**, e.g. the source energy spectrum, background energy spectrum, RMF file, ARF file, corrected source light-curve, corrected background light-curve, and field image in sky coordinates (FPMB module in the example). The source and background extraction regions are read from the input ASCII files *'source.reg*' and *'background.reg*', respectively. These files should have a format compatible with *XSELECT* and *DS9*.

In the case of **extended** sources the input parameter '*extended'* should be set to '*yes'* as in the following example:

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  extended=yes
```

Optionally, by setting the input parameter *'rungrppha=yes'*, the *'nuproducts'* module generates in output also a grouped source energy spectrum (parameter *'grpphafile'*). This is achieved through an internal run of the FTOOL *'grppha'*. The energy spectrum is grouped such that a minimum of *'grpmincounts'* counts is contained in each bin and the energy channels lower than *'grppibadlow'* and higher than *'grppibadhigh'* are flagged as bad. In the following the command line to be used to generate a source grouped energy spectrum is shown:

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  rungrppha=yes grpmincounts=30 grppibadlow=35 grppibadhigh=1909
```

The above example generates a binned energy spectrum (named '*nu10012001002B01_sr_grp.pha*') with a minimum of 30 counts in each new energy channel and with PI channels '0-34' and '1910-4095' flagged as bad. Note that the '35-1909' PI range corresponds to the 3-78 keV calibrated energy range

To extract the Level 3 scientific products with the *'nuproducts'* module directly from the *'event_cl'* archive directory it is also necessary to set the *'attfile'* and *'hkfile'* input parameters as follows:

```
> nuproducts indir=./event_cl instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  attfile=/archive/10012001002/auxil/nu10012001002_att.fits
  hkfile=/archive/10012001002/hk/nu10012001002B_fpm.hk
```

To extract Level 3 scientific products with the *'nuproducts'* module from a decomposed 06 cleaned event file (see Sections 4.10 and 6.7) it is necessary to set the *'infile'* input parameters as follows:

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  infile=nu10012001002B06_chu1_N_cl.evt srcregionfile=source06_chu1.reg
  bkgregionfile=background06_chu1.reg outdir=./products06
```

5.2 Standard Data Processing with *'nupipeline'* (Stages 1, 2 and 3)

The data reduction steps for the extraction of high-level scientific products are also coded in the third stage of the *'nupipeline'* script which directly runs the *'nuproducts'* module. This can be done by setting the input parameter *'exitstage=3'*. In this case the *'nupipeline'* script, in a single command line, applies to the data the calibration steps (Stage 1), the screening steps (Stage 2) and extracts the high-level data products (Stage 3). By default, data from both modules (FPMA and FPMB) are processed.

The standard command line to be used is:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out exitstage=3 srcregionfile=source.reg
  bkgregionfile=background.reg
```

5.3 The Extraction of high-level data products (*'nuproducts'*)

The NUSTARDAS module '*nuproducts*' is a perl module which utilizes the *XSELECT* multi-mission software package to extract source and background light-curves and energy spectra from the input calibrated and screened event files (Level 2). By default, *'nuproducts'* calls the '*numkrmf*' and '*numkarf*' tasks to generate the RMF and/or ARF files appropriate for the spectral analysis.

The '*nuproducts*' module also updates the values of the output source spectrum keywords 'BACKFILE', 'RESPFILE' and 'ANCRFILE' with the names of the background spectrum file, the RMF file and the ARF file, respectively. For spectral analysis (e.g. by using *XSPEC*), users should decide on grouping or not the source spectrum using an external tool (e.g. by using the FTOOL '*grppha'*). By default, *'nuproducts'* calls the *'nulccorr'* task, to apply various corrections to the output light-curves, and the *'nubackscale'* task to apply backscale corrections to the output energy spectra.

By default, the high-level data products are extracted from the 01 cleaned event file (input parameter *'infile=DEFAULT'*). The input parameters needed to run *'nuproducts'* are the Right Ascension and Declination of the source (i.e. parameters '*srcra'* and '*srcdec'*), the Right Ascension and Declination of the center of the background region (i.e. parameters '*bkgra'* and '*bkgdec'*), or alternatively, the source and background extraction region files. Also, the data input directory, the stem for FITS input files, the chosen NuSTAR detector (FPMA or FPMB) and the output directory for the generation of the products, must be specified.

The task requires in input the following files:

- Level 2 event file (*infile*)
- Source region file (*srcregionfile*)
- Background region file (*bkgregionfile*)
- Attitude file (*attfile*)
- Mast Aspect Solution file (*mastaspectfile*)
- OBEB Housekeeping FITS file (*obebhkfile*) (from *NuSTARDAS* 2.0.0)
- DET1 Reference Point file (*det1reffile*)
- Optical Axis file (*optaxisfile*)
- User GTI file (*usrgtifile*)
- Housekeeping file (*hkfile*)
- Pixel location file (*pixposfile*)
- Alignment file (*alignfile*)
- Telescope definition file (*teldef*)
- Instrument Probability Map File (*instrprobmapfile*)
- 2D-PSF file for Light curve correction (*psffile*)
- Vignetting file (*vignfile*)
- On-axis ARF file (*inarffile*)
- 2D-PSF Grouping File for ARF generation (*grppsffile*)
- Aperture Stop Correction file (*apstopcorrfile*)
- Ghost Rays Correction file (*grcorrfile*)
- DETABS Correction file (*detabscorrfile*)
- RMF Grouping File for RMF file generation (*grprmffile*)

The files generated in output are the following:

- Source spectrum (*phafile*)
- Source corrected light-curve (*lcfile*)
- Source image (*imagefile*)
- Background spectrum (*bkgphafile*)
- Background corrected light-curve (*bkglcfile*)
- Response Matrix file (*outrmffile)*
- Ancillary Response file (*outarffile*)

The source spectrum and light-curve are extracted by filtering the data on a spatial region in SKY coordinates. If the parameter *'srcregionfile'* is set to '*DEFAULT'* it is possible, through the parameters *'srcra'*, *'srcdec'* and *'srcradius'*, to specify the Right Ascension and Declination of the center and the radius of a circular extraction region. An extraction region file with an arbitrary shape can be input through the parameter *'srcregionfile'*. This file should have a format compatible with *XSELECT* and *DS9*. Note that, for point-like sources, while arbitrary region geometries can be used for products extraction, specific region geometries are required for the generation of corrected ARFs and the PSF correction of light-curves. For a list of the supported geometries of the extraction regions see the sections dedicated to the modules *'numkarf'* and *'nulccorr'* later in this Chapter.

A full field of view image in sky coordinates is produced if the parameter *'imagefile'* is different from *'NONE'*. **Note that energy spectra and light-curves are not background subtracted at this point.**

By setting the parameter *'bkgextract'* to *'yes'* (default value) it is possible to extract the spectrum and light-curve of the background from a region file specified by the parameter *'bkgregionfile'*. Alternatively (*'bkgregionfile'='DEFAULT'*), it is possible to define through the parameters *'bkgra'*, *'bkgdec'*, *'bkgradius1'* and *'bkgradius2'* the Right Ascension, Declination of the center and the inner and outer radii of an annular extraction region.

If the input parameter '*runbackscale*' is set to 'yes' (default), the 'BACKSCAL' keywords of the source and background energy spectra are updated by running the '*nubackscale'* task (see Section 5.5).

By default (parameter '*correctlc=yes'*), the task applies the livetime, PSF/EXPOSURE and vignetting corrections to the output source light-curves by running the '*nulccorr'* task (see Section 5.6). In the case of background light-curve only livetime and EXPOSURE corrections are applied. Note that in the case of point-like sources the EXPOSURE correction is always applied to the background light-curve. In the case of extended sources the background light-curve EXPOSURE correction is managed by the value of the input parameter *'lcexpoflag'*.

For the extraction of light-curves it is possible set the binsize (in seconds) via the parameter *'binsize'*. For the extraction of light-curves and sky images the minimum and maximum values of the PI channels can be changed via the parameters *'pilow'* and *'pihigh'*. Moreover, it is possible to apply a user defined temporal filter to the event file providing in input a FITS GTI file (parameter *'usrgtifile'*).

By setting the parameter *'plotdevice'* to GIF or PS, it is possible to choose the device for plotting the generated image, light curve and spectrum.

Optionally, by setting the input parameters *'rummkarf'* and *'runmkrmf'* to *'yes'* (default value), the task also generates the ARF and RMF files appropriate for the spectral analysis of the source energy spectrum. This is achieved by running the tasks '*numkarf'* and 'numkrmf' (see Sections 5.7 and 5.8, respectively).

The task can also optionally apply the barycenter correction to the output files by setting the input parameter *'barycorr'* to *'yes'* (default is *'no'*). The barycenter correction is calculated through the multimission FTOOL *'barycorr'*. The correction is applied to the following input files: *'infile'*, *'mastaspectfile'*, *'attfile'*, *'det1reffile'*, *'optaxisfile', 'hkfile'* and *'obebhkfile'*. Note that when *'nuproducts'* is run providing in input a user GTI file (parameter *'usrgtifile'*) the barycenter correction is applied to it if the input parameter '*usrgtibarycorr*' is set to *'yes'* (default value).

To apply the barycenter correction it is necessary to set the input parameter *'orbitfile'* to the "attitude-orbit file" generated by the '*nufilter'* task (i.e. *nu*A/B.attorb* file located in the '*nupipeline'* output directory) and specify the position of the source (parameters *'srcra_barycorr'* and *'srcdec_barycorr'*). The barycenter corrected cleaned event file can be saved in the output directory by setting the parameter *'write_baryevtfile=yes'*.

Optionally, by setting the input parameter *'rungrppha'* to *'yes'*, the output source energy spectrum is grouped through an internal run of the FTOOL *'grppha'*. The energy spectrum is grouped such that a minimum of *'grpmincounts'* counts is contained in each bin. Also, the energy channels lower than *'grppibadlow'* and higher than *'grppibadhigh'* are flagged as bad.

Usage examples

Point-like source case:

5.3A. Default Circular and annular extraction regions for source and background, respectively:

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcra=187.2779154 srcdec=2.0523883 bkgra=187.2779154
  bkgdec=2.0523883
```

In this example *nuproducts* generates in the output directory the following data products:

- source energy spectrum (pha file), extracted in circular region centered on *srcra* and *srcdec* with radius of 20 software pixels (parameter *srcradius*). The 'BACKSCAL' keyword in the source spectrum is updated for the exposure variations within the extraction region.
- background energy spectrum (pha file), extracted in annular region centered on *bkgcra* and *bkgdec* with inner and outer radius of 50 (parameter *bkgradius1*) and 80 (parameter *bkgradius2*) software pixels, respectively. The 'BACKSCAL' keyword in the background spectrum is updated for the exposure variations within the extraction region.
- RMF file, appropriate for the source energy spectrum.
- ARF file, appropriate for the point-like source spectrum.
- Source light-curve (not background subtracted) corrected for the livetime, the PSF and vignetting corrections, applied at energy *'lcenergy'* =10 keV.
- Background light-curve (corrected for the livetime and EXPOSURE corrections).
- Field image in sky coordinates.

5.3B. It is also possible to use source and/or background extraction region geometries different from circular and annular ones, respectively. In this case the region files with an arbitrary shape can be input through the parameters '*srcregionfile'* and/or *'bkgregionfile',* respectively. However, note that the task *'numkarf'*

supports only circular, elliptical and annular extraction regions centered on the source. The 'CIRCLE-CIRCLE' region is not supported, so we suggest using the 'ANNULUS' region. Moreover *'numkarf'* only applies the ghost-ray correction (parameter *'grflag'*) for point-like sources (parameter *'extended=no'*) and for circular extraction regions for the source:

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  lcfile=NONE bkglcfile=NONE imagefile=NONE
```

In this example *nuproducts* generates in the output directory the following data products:

- source energy spectrum (pha file), extracted in the region *source.reg*. The 'BACKSCAL' keyword of the source spectrum is updated for the exposure variations within the extraction region.
- background energy spectrum (pha file), extracted in the region *background.reg*. The 'BACKSCAL' keyword of the background spectrum is updated for the exposure variations within the extraction region.
- RMF file (appropriate for the source spectrum),
- ARF file (for a point-like source spectrum).

5.3C. It is also possible to apply a time filter to the event file and to the high-level data products by providing an input FITS GTI (Good time interval) file (parameter *'usrgtifile'*):

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  usrgtifile=myGTIs.fits
```

5.3D. To analyze light-curves in a specific energy range and temporal bin size, it is necessary to set the values of the parameters '*pilow'* and '*pihigh',* which define the minimum and maximum value of the PI to be included in the extracted light curve, and the parameter *binsize:*

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  phafile=NONE outdir=./products srcregionfile=source.reg bkgextract=no
  bkglcfile=NONE imagefile=NONE runmkarf=no runmkrmf=no pilow=100 pihigh=1000
  binsize=500 lcnergy=5
```

In this example *nuproducts* generates in the directory products only the source light-curve extracted using the spatial region *source.reg*, the energy range 100-1000 (in PI channels) and with a temporal bin size of 500 s. The source light-curve, not background subtracted, is corrected for the livetime and the PSF and vignetting corrections are applied at energy 5 keV.

5.3E. Barycenter correction (with default circular and annular extraction regions for source and background, respectively):

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcra=187.2779154 srcdec=2.0523883 bkgra=187.2779154
  bkgdec=2.0523883 barycorr=yes srcra_barycorr=187.2779154
  srcdec_barycorr=2.0523883 orbitfile=./pipeline_out/nu10012001002B.attorb
```

In this example *nuproducts* generates in the output directory the following barycenter corrected data products:

- source energy spectrum (pha file), extracted in circular region centered on *srcra* and *srcdec* with radius of 20 software pixels (parameter *srcradius*). The 'BACKSCAL' keyword in the source spectrum is updated for the exposure variations within the extraction region.
- background energy spectrum (pha file), extracted in annular region centered on *bkgcra* and *bkgdec* with inner and outer radius of 50 (parameter *bkgradius1*) and 80 (parameter *bkgradius2*) software pixels, respectively. The 'BACKSCAL' keyword in the background spectrum is updated for the exposure variations within the extraction region.
- RMF file, appropriate for the source energy spectrum.
- ARF file, appropriate for the point-like source spectrum.
- Source light-curve (not background subtracted) corrected for the livetime, the PSF and vignetting corrections, applied at energy *'lcenergy'* =10 keV .
- Background light-curve (corrected for the livetime and EXPOSURE corrections).
- Field image in sky coordinates.

5.3F. Same as Example 5.3B, in addition a grouped source energy spectrum is generated (named 'nu10012001002B01_sr_grp.pha'):

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  lcfile=NONE bkglcfile=NONE imagefile=NONE rungrppha=yes grpmincounts=30
  grppibadlow=35 grppibadhigh=1909
```

Extended sources case.

In the case of an extended source the Point Spread Function and Ghost-Rays corrections are not applied to the output ARF file. The correction terms include the vignetting, the aperture stop, the detector absorption and the EXPOSURE corrections. To generate the ARF appropriate for an extended source, the value of the parameters *'extended'* must be set to *'yes'* (default value is *'no'*).

5.3G.

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  lcfile=NONE bkglcfile=NONE imagefile=NONE extended=yes
```

In this example *nuproducts* generates in the output directory the following files:

- source energy spectrum (pha file), extracted in the region *source.reg*. The 'BACKSCAL' keyword of the source spectrum is updated for the exposure variations within the extraction region.
- background energy spectrum (pha file), extracted in the region *background.reg*. The 'BACKSCAL' keyword of the background spectrum is updated for the exposure variations within the extraction region.
- RMF file (appropriate for the source spectrum).
- ARF file (for the extended source spectrum).

5.4 The Extraction of high-level data products with 'nupipeline'

The *'nuproducts'* module for the extraction of high-level products is coded in the third stage of the *'nupipeline'* script. Therefore by using *'nupipeline'* setting the parameter *'exitstage'* to *'3'*, the script calls *'nuproducts'* and extracts the high-level data products **for the 01 cleaned event files**. In the examples given below the same numbering of the previous section is used. Note that it is not possible to generate source grouped energy spectra with the *'nupipeline'* module.

Usage examples:

5.4A.

```
> nupipeline indir=./10012001002 outdir=./pipeline_out
steminputs=nu10012001002 srcra=187.2779154 srcdec=2.0523883
obsmode=SCIENCE instrument=FPMB exitstage=3 bkgra=187.2779154
bkgdec=2.0523883
```

5.4B.

```
> nupipeline indir=./10012001002 outdir=./pipeline_out
steminputs=nu10012001002 obsmode=SCIENCE instrument=FPMB exitstage=3
srcregionfile=source.reg bkgregionfile=background.reg fpmb_lcfile=NONE
fpmb_bkglcfile=NONE fpmb_imagefile=NONE
```

5.4C.

```
> nupipeline indir=./10012001002 outdir=./pipeline_out
steminputs=nu10012001002 obsmode=SCIENCE instrument=FPMB exitstage=3
srcregionfile=source.reg bkgregionfile=background.reg
productsusrgtifile=myGTIs.fits
```

5.4D.

```
> nupipeline indir=./10012001002 outdir=./pipeline_out
steminputs=nu10012001002 obsmode=SCIENCE instrument=FPMB exitstage=3
srcregionfile=source.reg bkgextract=no fpmb_phafile=NONE
fpmb_bkglcfile=NONE fpmb_imagefile=NONE runmkarf=no runmkrmf=no pilow=100
pihigh=1000 binsize=500
```

5.4E.

```
> nupipeline indir=./10012001002 outdir=./pipeline_out
steminputs=nu10012001002 srcra=187.2779154 srcdec=2.0523883
obsmode=SCIENCE instrument=FPMB exitstage=3 bkgra=187.2779154
bkgdec=2.0523883 barycorr=yes srcra_barycorr=187.2779154
srcdec_barycorr=2.0523883
```

5.4G. (EXTENDED SOURCES):

```
> nupipeline indir=./10012001002 outdir=./pipeline_out
steminputs=nu10012001002 obsmode=SCIENCE instrument=FPMB exitstage=3
srcregionfile=source.reg bkgregionfile=background.reg fpmb_lcfile=NONE
fpmb_bkglcfile=NONE fpmb_imagefile=NONE extended=yes
```

5.5 Apply backscale correction to energy spectra (*'nubackscale'*)

The NuSTARDAS software module *'nubackscale'* generates a corrected source and/or background spectrum accounting for the exposure variations within the extraction regions. The corrections are calculated using the sky exposure map generated internally by the '*nuexpomap*' task. A sky exposure map, not corrected for vignetting, can also be provided in input to the task by setting the input parameter *'inexpomapfile'*, allowing to reduce the execution time. The task generates output FITS PHA files ('*srcoutfile*' and/or '*bkgoutfile*') with updated values of the 'BACKSCAL' keyword.

The task requires in input the following files:

- source spectrum PHA file (*srcphafile*)
- background spectrum PHA file (*bkgphafile*)
- Level 2 event file (*evtfile)*
- Attitude file (*attfile*)
- Mast Aspect Solution file (*mastaspectfile*)
- DET1 Reference Point file (*det1reffile*)
- Pixel location file (*pixposfile*)
- Alignment file (*alignfile*)
- Telescope definition file (*teldef*)
- Instrument Probability Map File (*instrprobmapfile*)

The files generated in output by the task are:

- source spectrum PHA file (*srcoutfile*)
- background spectrum PHA file (*bkgoutfile*)

This module is called by the *'nuproducts'* module.

Usage example:

1. In this example *'nubackscale'* will update, with the backscale correction, the BACKSCAL keyword of the input source and background spectra: '*nu10012001002A01_sr.pha'* and '*nu10012001002A01_bkg.pha'*, respectively:

```
> nubackscale evtfile=nu10012001002A01_cl.evt
  srcphafile=nu10012001002A01_sr.pha attfile=nu10012001002_att.fits
  det1reffile=nu10012001002A_det1.fits  mastaspectfile=nu10012001002_mast.fits
  bkgphafile=nu10012001002A01_bkg.pha srcoutfile=NONE bkgoutfile=NONE
```

2. Using this command line, *'nubackscale'* applies the backscale correction to the input source and background spectra. The corrected spectra are written in the output files *'nu10012001002A01_sr_corr.pha*' and *'nu10012001002A01_bk_corr.pha'*:

```
> nubackscale evtfile=nu10012001002A01_cl.evt
  srcphafile=nu10012001002A01_sr.pha attfile=nu10012001002_att.fits
  det1reffile=nu10012001002A_det1.fits  mastaspectfile=nu10012001002_mast.fits
  bkgphafile=nu10012001002A01_bkg.pha srcoutfile=nu10012001002A01_sr_corr.pha
  bkgoutfile=nu10012001002A01_bkg_corr.pha
```

3. Same as Example 1, but with a sky exposure map (not corrected for vignetting) provided in input:

```
> nubackscale evtfile=nu10012001002A01_cl.evt
  srcphafile=nu10012001002A01_sr.pha attfile=nu10012001002_att.fits
  det1reffile=nu10012001002A_det1.fits  mastaspectfile=nu10012001002_mast.fits
  bkgphafile=nu10012001002A01_bkg.pha srcoutfile=NONE bkgoutfile=NONE
  inexpomapfile=nu10012001002A01_ex.img
```

5.6 Apply livetime, PSF/EXPOSURE and vignetting corrections to light-curves (*'nulccorr'*)

The NuSTARDAS software module *'nulccorr'* generates a corrected light-curve accounting for the livetime, the PSF/EXPOSURE and vignetting corrections. The livetime correction is always applied and it is calculated using the values of the 'LIVETIME' column in the input file '*hkfile'*.

The task requires in input the following files:

- Light curve file (*lcfile*)
- Spectrum pha file (*phafile*)
- Optical Axis File (*optaxisfile*)
- Level 2 event file (*infile*)
- Attitude file (*attfile*)
- Mast Aspect Solution file (*mastapectfile*)
- DET1 Reference Point file (*det1reffile*)
- Instrument Housekeeping File (*hkfile*)
- Pixel location file (*pixposfile*)
- Alignment file (*alignfile*)
- Telescope definition file (*teldef*)
- Instrument Probability Map File (*instrprobmapfile*)
- 2D-PSF file (*psffile*)
- Vignetting file (*vignfile*)

The file generated in output by the task is:

- Corrected light curve file (*outlcfile*)

For point-like sources (parameter *'extended=no'*), the task applies to the output light-curve the PSF and vignetting corrections by setting the input parameters '*psfflag*' and '*vignflag*' to 'yes' (default values), respectively. The PSF term includes the corrections for the fraction of photons falling outside the light-curve extraction region and for bad pixels and detector gaps located within the extraction region. The vignetting and PSF corrections are applied using the energy value specified by the parameter '*energy*'.

For the point-like case, circular, elliptical and annular extraction regions centered on the source are supported by the task if the PSF correction is applied. No restrictions on the shape of the extraction region are present if the input parameter *'psfflag'* is set to *'no'*.

For extended sources (parameter *'extended=yes'*), the task applies to the output light-curve the EXPOSURE and vignetting corrections when setting the input parameters '*expoflag*' and '*vignflag*' to 'yes' (default values), respectively. The EXPOSURE correction term accounts for bad pixels and detector gaps located within the extraction region. Also in this case, the vignetting correction is applied using the energy value specified by the parameter '*energy*'. For the extended case there are no restrictions on the shape of the extraction region.

By default the PSF/EXPOSURE and vignetting corrections are calculated using a sky instrument map (see *'nuexpomap'*) generated internally by the task. To reduce the task execution time, it is possible to provide in input to *'nulccorr'* a sky instrument map file and the associated aspect histogram file through the parameters *'inskyinstrfile'* and *'inaspecthistofile'*, respectively. Note that in the latter case to produce vignetting corrected output light-curves it is necessary to provide in input a sky instrument map corrected for vignetting.

The task generates an output FITS light curve ('*outlcfile*') with the 'RATE' and 'ERROR' columns corrected for the different factors. The original values of the 'RATE' and 'ERROR' columns are saved and stored in the two new columns 'RATE_ORIG' and 'ERROR_ORIG' of the output light-curve. The task also sets the value of the 'DEADAPP' keyword to 'T' to reflect the fact that the output light-curve is livetime corrected.

The task adds or updates to the output light-curve the keyword 'NULCCO' with value 'T' to specify that the light-curve has been corrected.

This module is called by the *'nuproducts'* module.

Usage examples:

1. In this example *'nulccorr'* applies the livetime, PSF and vignetting corrections at energy 10 keV to the input light curve:

```
> nulccorr evtfile=nu10012001002A01_cl.evt outlcfile=NONE
  srcphafile=nu10012001002A01_sr.pha attfile=nu10012001002_att.fits
  det1reffile=nu10012001002A_det1.fits mastaspectfile=nu10012001002_mast.fits
```

2. The energy of PSF and vignetting corrections can be specified by the parameter *'energy'*. In the example below *'nulccorr'* applies the correction at an energy of 5 keV:

```
> nulccorr evtfile=nu10012001002A01_cl.evt outlcfile=NONE energy=5
  srcphafile=nu10012001002A01_sr.pha attfile=nu10012001002_att.fits
  det1reffile=nu10012001002A_det1.fits mastaspectfile=nu10012001002_mast.fits
```

3. **Extended source**: in this case the light-curve is corrected for the EXPOSURE variations (i.e. detectors edges/gaps, bad/hot pixel), but no PSF correction is applied. The parameter *'extended'* has to be set to *'yes'* for the extended case, as in the example below:

```
> nulccorr evtfile=nu10012001002A01_cl.evt outlcfile=NONE
  srcphafile=nu10012001002A01_sr.pha attfile=nu10012001002_att.fits
  det1reffile=nu10012001002A_det1.fits mastaspectfile=nu10012001002_mast.fits
  extended=yes
```

4. Background light curve: in this case the light curve should be corrected only for the livetime and EXPOSURE variations. The command line to be used for a background light-curve is:

```
> nulccorr evtfile=nu10012001002A01_cl.evt outlcfile=NONE
  srcphafile=nu10012001002A01_sr.pha attfile=nu10012001002_att.fits
  det1reffile=nu10012001002A_det1.fits mastaspectfile=nu10012001002_mast.fits
  extended=yes vignflag=no
```

5.7 Generation of ARF files (*'numkarf'*)

The NuSTARDAS software module *'numkarf'* generates an OGIP-style Ancillary Response File (ARF) for an input PHA file, which has been extracted by *'nuproducts'* or *'nupipeline'* or *XSELECT*.

The task requires in input the following files:

- Source spectrum PHA file (*phafile*)
- Source region file (*srcregionfile*)
- Optical Axis File (*optaxisfile*)
- Level 2 event file (*infile*)
- Attitude file (*attfile*)
- Mast Aspect Solution file (*mastaspectfile*)
- OBEB Housekeeping FITS File (*obebhkfile*) (from *NuSTARDAS* 2.0.0)
- DET1 Reference Point file (*det1reffile*)
- On-axis ARF File (*inarffile*)
- Pixel location file (*pixposfile*)
- Alignment file (*alignfile*)
- Telescope definition file *(teldef*)
- Instrument Probability Map File (*instrprobmapfile*)
- 2D-PSF Grouping File (*grppsffile*)
- Vignetting file (*vignfile*)
- Aperture Stop Correction file (*apstopcorrfile*)
- Ghost Rays Correction file *(grcorrfile*)
- DETABS Correction file (*detabscorrfile*)

The task generates in output the following files:

- ARF file (*outfile*)
- Off-Axis file (*offaxisfile*)
- Off-Axis Histogram file (*offaxishisto*)
- Aperture Stop Histogram file (*apstophisto*)
- Ghost Rays Histogram file (*grhisto*)

'numkarf' generates an OGIP-style Ancillary Response Function (ARF) file which is suitable for input into the spectral fitting program *XSPEC*. The ARF file contains the effective area of the telescope as a function of energy needed to perform spectral analysis. This is calculated by correcting the input CALDB ARF file, storing the effective area of a source located on-axis without Point Spread Function (PSF) losses.

The input CALDB ARF file is modified taking into account the vignetting, the aperture stop, the ghost rays, the detector absorption and the PSF corrections. These terms, which are functions of the energy, are dependent on the source off-axis angle and azimuth angle. The PSF and ghost rays corrections are also dependent on the radius of the PHA file extraction region. The PSF correction term also includes the corrections for the exposure map such as detector gaps and mast/attitude variations.

Starting from version 0.3.7 (*NuSTARDAS* 2.0.0), the '*numkarf*' module implements, for FPMA only, a CALDB ARF file query which is temperature dependent (CALDB version 20200513 or higher is required). The module reads the temperature from the 'OPT0_5TEMP' column of the input OBEB Housekeeping FITS File (parameter '*obebhkfile'*) and then calculates its average value.

The input files for on-axis effective area, PSF, aperture stop, ghost rays, detector absorption and vignetting are, by default, read from the Calibration Database (CALDB). For the PSF files, the CALDB stores a set of files which are energy dependent. The association between energy and CALDB PSF files is specified in the CALDB PSF Grouping File provided in input through the parameter '*grppsffile*'.

The time dependent off-axis angle, azimuth value and distance between the optical axis and the aperture stop center are calculated using the position of the telescope optical axis and of the aperture stop center stored in the input file '*optaxisfile*' (see '*nuskypos*' and '*nucoord*' help files), and written in the output file '*offaxisfile'*.

The distributions of off-axis and azimuth angles for the vignetting correction are stored in the output file '*offaxishisto'*. The distributions of off-axis, azimuth angles and distance between the optical axis and the aperture stop center for the aperture stop and ghost rays corrections are stored in the output files '*apstophisto*' and '*grhisto*' files, respectively.

The vignetting, aperture stop, ghost rays and PSF corrections are calculated, taking into account the source off-axis and azimuth angle distributions, by a weighted mean of the input CALDB values.

If parameters '*psfflag*', '*apstopflag*', '*grflag*', '*detabs*' and '*vignflag*' are set to '*yes'*, the PSF, the aperture stop, the ghost rays, the detector absorption and the vignetting corrections are applied to the input ARF file, respectively.

The source position and the radius of the extraction region are read from the input spectrum (parameter '*phafile*') in the 'REG00101' extension. Only for the detector absorption correction, the source extraction region is read from the input parameter '*srcregionfile*'. We note that the source extraction region specified though the parameter '*srcregionfile'* must be the same used to extract the input energy spectrum. Circular, elliptical and annular extraction regions centered on the source are supported by the task. Note that the ghost rays correction is supported only for point-like sources (parameter '*extended*=no') and for circular source extraction regions.

The task can also generate the ARF file for extended sources, such as clusters of galaxies or supernova remnants, by setting the parameter '*extended*' to *'yes'*. To follow the spatial variations of the detector properties for the extended source case, the extraction region file, read from the first extension of the input spectrum file, is divided in boxes of size chosen by setting the parameter '*boxsize*'. For each sub-region, an ARF is produced with the vignetting correction, the aperture stop correction and the detector absorption correction, but without including the point-like PSF correction. This set of ARF files are then combined using the '*addarf*' FTOOL to produce the output ARF. Note that for the extended case there no restrictions on the shape of the extraction region.

For extended sources, the task execution time can be reduced by providing in input a sky exposure map file not corrected for vignetting (parameter *'inexpomapfile'*).

This module is called by the *'nuproducts'* task.

Usage examples:

1. In this example *numkarf* generates the ARF suitable for the point-like spectrum *nu10012001002A01_sr.pha* extracted by using the circular region *source.reg*. The ARF is corrected for the PSF, the aperture stop, the ghost rays, the detector absorption and the vignetting factors. If the source extraction region *source.reg* is not circular, the ghost rays correction is not applied to the output ARF file:

```
> numkarf infile=nu10012001002A01_cl.evt phafile=nu10012001002A01_sr.pha
  outfile=nu10012001002A01_sr.arf optaxisfile=nu10012001002A_oa.fits
  det1reffile=nu10012001002A_det1.fits srcregionfile=source.reg
  mastaspectfile=nu10012001002_mast.fits attfile=nu10012001002_att.fits
  obebhkfile=nu10012001002A_obeb.hk offaxisfile=NONE offaxishisto=NONE
  grhisto=NONE apstophisto=NONE
```

2. **Extended source**: in this case the ARF cannot be corrected for the PSF and ghost-rays, and only the aperture stop, the detector absorption and the vignetting corrections are applied. To generate the ARF appropriate for an extended source, the value of the parameter *'extended'* has to be set to 'yes'.

   Assuming that *nu10012001002A01_sr.pha* is the spectrum of the extended source extracted by using the region file *source.reg*, the command line to be used is:

```
> numkarf infile=nu10012001002A01_cl.evt srcregionfile=source.reg
  phafile=nu10012001002A01_sr.pha outfile=nu10012001002A01_sr.arf
  optaxisfile=nu10012001002A_oa.fits det1reffile=nu10012001002A_det1.fits
  offaxisfile=NONE offaxishisto=NONE grhisto=NONE apstophisto=NONE
  mastaspectfile=nu10012001002_mast.fits obebhkfile=nu10012001002A_obeb.hk
  attfile=nu10012001002_att.fits extended=yes
```

5.8 Generation of RMF files (*'numkrmf'*)

The NuSTARDAS software module *'numkrmf'* generates an OGIP-style Response Matrix File (RMF) which is suitable for input into the spectral fitting program XSPEC.

The task requires in input the following files:

- Level 2 event file (*infile*)
- Source region file (*srcregionfile*)
- CALDB RMF Grouping File (*grprmffile*)
- User GTI file (*usrgtifile*)

The file generated in output by the task is:

- RMF file (*outfile*)

For NuSTAR, the CALDB stores two sets of RMF files, to be used for event files filtered or not filtered for the depth cut. Each set stores the RMF file appropriate for each detector pixel or group of detector pixels. The association between pixels and CALDB RMF files is specified in the CALDB RMF Grouping File provided in input through the parameter '*grprmffile'*.

'*numkrmf*' first reads the keyword 'DEPTHCUT' of the input event file (see Section 3.9) to select the appropriate '*grprmffile'* CALDB file. Next, for each CALDB RMF file, it computes the number of events in the input event file (*'infile'*) within the input spatial region file (parameter '*srcregionfile*').

The response of the detector is finally generated by '*numkrmf*' calculating a weighted mean of the CALDB RMF files. The mean is computed using as weights the corresponding number of events and making use of the multi-mission FTOOL '*addrmf*'. Optionally, by setting the input parameter '*cmprmf*' to 'yes', the output RMF file is compressed using the multi-mission FTOOL '*cmprmf*'.

This module is called by the *'nuproducts'* task.

Usage example:

1. Generates the RMF file appropriate for the input spatial region *'source.reg'*:

```
> numkrmf infile=nu10012001002A01_cl.evt srcregionfile=source.reg
  outfile=nu10012001002A01_sr.rmf
```

6 SPECIFIC ANALYSIS CASES

In this Chapter we discuss some important issues dealing with the analysis of NuSTAR data. In fact, particular operative conditions during an observation may require specific procedures. Before providing the recipes to be followed by the user in such cases, a short description of a standard data processing is given.

6.1 Standard Processing

The first step in the data reduction of NuSTAR data is to calibrate and screen the Level 1 event files. This can be achieved by running the *'nupipeline'* script as illustrated in the following command line:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out obsmode=SCIENCE
```

Note that only the 'SCIENCE' observing mode data, storing the data with good attitude reconstruction, are considered in this example. Both data from the FPMA and the FPMB modules are processed. All the '*nupipeline*' output files are generated in the directory named *"pipeline_out".* In particular the output directory contains the calibrated and cleaned Level 2 event files *nu10012001002A01_cl.evt* and *nu10012001002B01_cl.evt*. Optionally, by setting the input parameter *'createexpomap=yes',* the sky exposure maps are also generated.

The second step in the data reduction is the extraction of high-level scientific products. To this end, the *'nuproducts'* module automatically retrieves all the necessary input files from the output directory of *'nupipeline'* (see above). An example of a standard command line for a **point-like** source for module FPMA is the following:

```
> nuproducts indir=./pipeline_out instrument=FPMA steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
```

This command generates in the *'products'* directory the high-level data products **for the 01 cleaned event file**, e.g. the source and background energy spectra, the RMF and ARF files for spectral analysis, the corrected source and background light-curves, and a field image in sky coordinates. In the case of extended sources, the option *'extended=yes'* must be added to the *'nuproducts'* command line. The *'nuproducts'* module also updates the values of the keywords 'BACKFILE', 'RESPFILE' and 'ANCRFILE' of the source energy spectra with the names of the background spectrum file, the RMF file and the ARF file, respectively.

The source and background extraction regions are read from the input ASCII files *'source.reg*' and *'background.reg*', respectively. These files should have a format compatible with *XSELECT* and *DS9*. Note that for point-like sources only **circular**, **elliptical**, and **annular** source extraction regions are supported by *'nuproducts'*. Note also that the ARF ghost rays correction, applied only for point-like sources, requires the use

of **circular** source extraction regions. For the extended source case there are no restrictions on the shape of the extraction region.

The '*nuproducts*' module makes use of the *XSELECT* multi-mission software package to extract source and background light-curves and energy spectra. The module automatically calls the *'nulccorr'* task, to apply livetime and PSF corrections to the output light-curves, and the *'nubackscale'* task to apply backscale corrections to the output energy spectra. **For these reasons, it is strongly suggested that energy spectra and light-curves are extracted from calibrated and cleaned event files using the *'nuproducts'* module and not manually with XSELECT**.

In the case of **extended** sources the input parameter '*extended*' should be set to '*yes*' as in the following example:

```
> nuproducts indir=./pipeline_out instrument=FPMB steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  extended=yes
```

To extract the Level 3 scientific products with the *'nuproducts'* module directly from the *'event_cl'* archive directory it is also necessary to set the *'attfile'* and *'hkfile'* input parameters as follows:

```
> nuproducts indir=./event_cl instrument=FPMA steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  attfile=/archive/10012001002/auxil/nu10012001002_att.fits
  hkfile=/archive/10012001002/hk/nu10012001002A_fpm.hk
```

Note that from *NuSTARDAS* version 2.0.0 (and later) a new input file (OBEB Housekeeping '*obeb.hk*') is required by the '*nuproducts*' module. If not present in the '*event_cl*' directory (which may be the case if '*nupipeline*' has not been run), the user should provide the path to the 'obeb.hk' file through the command line by adding to the above example `obebhkfile=/archive/10012001002/hk/nu10012001002A_obeb.hk` (this should also be done for processing FPMB).

6.2 Extraction of light-curves with a user defined energy range

The *'nuproducts'* module by default generates source and background light-curves in the 35-1909 PI energy channels, i.e. 3-78 keV. It is possible to extract the light-curves in user defined energy range through the input parameters '*pilow*' and '*pihigh*'. In the following example the light-curves are generated in the energy range 10-60 keV:

```
> nuproducts indir=./pipeline_out instrument=FPMA steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  pilow=210 pihigh=1459
```

6.3 Generation of high-level data products with a user defined temporal filter

The *'nuproducts'* module allows the user to apply a time filter to all the high-level data products generated in output, i.e. the source and background energy spectra and light-curves, the RMF and ARF files. This is achieved by providing in input a Good Time Interval (GTI) file in FITS format through the parameter *'usrgtifile'*. The command line to be used is the following:

```
> nuproducts indir=./pipeline_out instrument=FPMA steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  usrgtifile=mygti.fits
```

Note that in the case of a user GTI file it is not possible to generate the correct data products if these are extracted with the XSELECT package individually. **To produce Level 3 data products with a user defined temporal filter it is mandatory to use the *'nuproducts'* module.**

6.4 Generation of cleaned event files with a user defined temporal filter

The *'nupipeline'* module allows the user to apply a time filter to the output Level 2 cleaned event files by providing in input a GTI file through the input parameter '*usrgtifile'* as follows:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out usrgtifile=myGTI.fits
```

We note that the output cleaned event files are corrected for the rate-dependent instrument livetime through an internal run of the '*nulivetime'* software module.

We stress that if instead the GTI filter is applied by the user to the cleaned event files running externally the XSELECT package, the output filtered event file is not corrected for livetime. **In this case the user must update the EXPOSURE and LIVETIME keywords of the event file by running manually the '*nulivetime'* software module on the extracted event file.**

6.5 Metrology data out of range: recovery of lost exposure time

As described in Chapter 3, the first data reduction step for NuSTAR is the processing of the raw metrology data by the *'numetrology'* module. The laser metrology system on-board NuSTAR tracks the temporal changes of the alignment of the mast connecting the Focal Plane Bench (FB) and the Optics Bench (OB) mirror system. This system makes use of two lasers, located on the OB, and two Position Sensing Detectors ('PSDO' and 'PSD1') on the FB registering the positions of the laser spots (for more details see Harrison et al. 2013, ApJ, 770, 103).

The processing of the raw metrology data first calculates the raw X and Y positions of the laser spots on the detectors 'PSD0' and 'PSD1' as a function of time. Next, the X/Y coordinates are corrected for the distortions

introduced by the response of the two laser detectors. This correction is applied only if the laser spots are inside a defined calibrated range for X and Y values. In sporadic cases the laser spots fall outside this calibrated range and the corresponding time intervals are filtered out during the data screening.

For the 'SCIENCE' observing mode, the value of the exposure time (not livetime corrected) lost because of the laser spots outside the X/Y calibrated range is reported in a specific keyword of the cleaned Level 2 event file named 'NUPSDOUT'. The value of the lost exposure time is also reported in the STDOUT of '*nupipeline'* module as in the example below:

```
nuscreen_0.1.9: Info: Metrology laser spots out of calibrated PSD grid
for ONTIME=391.8 seconds

FPMA: .... Stage II: ** WARNING Metrology laser spots out of
calibrated PSD grid for ONTIME=391.8 seconds for Observing Mode '01'
```

To recover the lost exposure time, at the expense of some positional uncertainty, a specific *'nupipeline'* input parameter named *'psdcal'* allows to use of the raw X/Y coordinates of the laser spots without applying the distortion correction. This procedure is safe for the analysis of bright sources, however users must check the sky images to verify the accuracy of the reconstruction of celestial positions.

To use raw PSD X/Y positions, and recover the exposure time, the *'nupipeline'* input parameter *'psdcal'* must be set to *'no'* (default value is *'yes'*) by the user as in the example below:

```
> nupipeline indir=/archive/10012001002/ steminputs=nu10012001002
  outdir=./pipeline_out obsmode=SCIENCE psdcal=no
```

6.6 Barycenter Correction

This section describes how to apply the barycenter correction to the NuSTAR data making use of the multi-mission FTOOL *'barycorr'*, distributed as part of HEASoft software package by the NASA-HEASARC.

The barycenter correction can be applied to NuSTAR data using the following two methods:

1. run of the *'nuproducts'* module with the input parameter *'barycorr=yes'*;
2. run of *'barycorr'* to the light-curves generated by a standard run of *'nuproducts'.*

Note that in both cases the *'nuproducts'/'barycorr'* modules also apply NuSTAR specific clock offset corrections using information stored in an input file specified through the input parameter *'clockfile'* (set to *'CALDB'* by default).

6.6.1 Method 1: Level 3 files correction with *'nuproducts'*

The barycenter correction is applied to the output files by setting the *'nuproducts'* input parameter *'barycorr=yes'*. The correction is applied internally running the *'barycorr'* module on the following input files: *'infile'*, *'mastaspectfile'*, *'attfile'*, *'det1reffile'*, *'optaxisfile', 'hkfile'* and *'obebhkfile'*. Note that when *'nuproducts'* is run providing in input a user GTI file (parameter *'usrgtifile'*) the barycenter correction is applied to it if the input parameter '*usrgtibarycorr*' is set to *'yes'* (default value).

To apply the barycenter correction it is necessary to set the input parameter *'orbitfile'* to the "attitude-orbit file" generated by the '*nufilter'* task (i.e. *nu*A/B.attorb* file located in the '*nupipeline'* output directory) and specify the position of the source (parameters *'srcra_barycorr'* and *'srcdec_barycorr'*). The barycenter corrected cleaned

event file can be saved in the output directory by setting the parameter *'write_baryevtfile=yes'*. The example below gives the command line to be used:

```
> nuproducts indir=./pipeline_out instrument=FPMA steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
  barycorr=yes srcra_barycorr=187.2779154 srcdec_barycorr=2.0523883
  orbitfile=./pipeline_out/nu10012001002A.attorb write_baryevtfile=yes
```

6.6.2 Method 2: light-curve correction with *'barycorr'*

To apply the barycenter correction to the light-curve generated by a standard run of the *'nuproducts'* module the following command lines must be executed.

i. Standard *'nuproducts'* run, where *'pipeline_out'* is the *'nupipeline'* output directory:

```
> nuproducts indir=./pipeline_out instrument=FPMA steminputs=nu10012001002
  outdir=./products srcregionfile=source.reg bkgregionfile=background.reg
```

ii. Barycenter correction of the output light-curve run from the *'nuproducts'* output directory (named *'products'* in the above command line):

```
> barycorr infile=nu10012001002A01_sr.lc outfile=nu10012001002A01_sr_corr.lc
  orbitfiles=../pipeline_out/nu10012001002A.attorb ra=187.2779154 dec=2.0523883
```

6.7 Scientific analysis of spacecraft mode 06 cleaned event files

The NuSTAR spacecraft science (observation mode 06) cleaned event files refer to scientific target data collected during time intervals in which an aspect solution is not available from the on-board star tracker located on the X-ray optics bench (Camera Head Unit #4, or CHU4) because it is either blinded by a bright target or by the Earth. The CHU4 is the primary method for determining the telescopes absolute pointing and it is used for the sky coordinates reconstruction of the NuSTAR cleaned event files (i.e. observation mode 01). The CHU4 aspect reconstruction accuracy is $\pm$ 8 arcseconds.

When CHU4 is not available, the aspect reconstruction used to calculate sky coordinates is derived using data from the three star trackers located on the spacecraft bus (CHUs #1, #2, #3, CHU1/CHU2/CHU3). In these cases the accuracy of the sky coordinates degrades to about **± 2 arcminutes**, mainly due to thermal flexing of the spacecraft bus star cameras. In particular, the three star tracker camera unit data can be integrated in seven different combinations (CHU1 only, CHU2 only, CHU3 only, CHU1&2, CHU1&3, CHU2&3, and CHU1&2&3) that are characterized by a different aspect reconstruction. As a result of the switching between

different CHU1/CHU2/CHU3 combinations during an observation, the sky images derived from mode 06 event files can show **multiple centroids of the same source** precluding an accurate scientific analysis of these data. An example is given in Figure 5 for a NuSTAR observation of the Circinus Galaxy (obsID 60002039002).

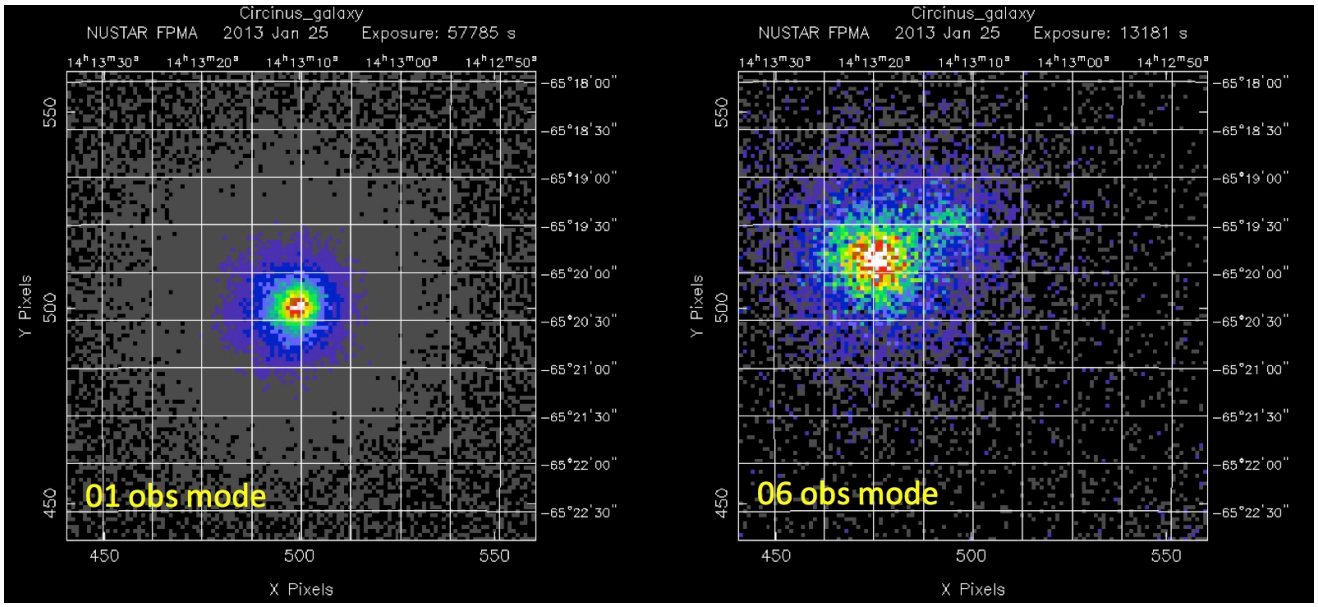

Figure 5: NuSTAR FPMA sky images of the Circinus Galaxy (obsID 60002039002) for observation modes 01 (left) and 06 (right). In the mode 01 image, derived from CHU4 data, the target appears as a point-like source, while the mode 06 image shows multiple centroids from the same target due to the poor aspect reconstruction and the switching between different CHU1/CHU2/CHU3 combinations.

The inaccuracies incurred from the spacecraft bus attitude solution are unique for each pointing and cannot be modeled. In particular, the severity of the offsets between different CHU1/CHU2/CHU3 combinations is dependent on the Solar aspect angle (Saa) and some unknown variables that make them unpredictable.

The presence of multiple centroids in the sky images derived from mode 06 event files prevents the possibility choosing a single correct spatial region. The use of a large extraction region is not recommended since for point-like sources the high-level processing software modules (*numkarf*, *nulccorr*) assume a single source at the center of the spatial region to apply the PSF corrections for effective areas and count rates.

To allow the scientific analysis of the observation mode 06 cleaned event files a specific NuSTARDAS software module named *'nusplitsc'* is available (see Sect. 4.10). This module decomposes the mode 06 event files into a maximum of seven distinct event files corresponding to the seven combinations of CHU1, CHU2 and CHU3.

The example below gives a '*nusplitsc'* command line for the NuSTAR observation of the Circinus Galaxy shown in Figure 5:

```
> nusplitsc infile=nu60002039002A06_cl.evt hkfile=nu60002039002A_fpm.hk
  chu123hkfile=nu60002039002A chu123.fits
```

The event files generated by this '*nusplitsc'* run are:

```
nu60002039002A06_chu3_N_cl.evt       FPMA 06 event file for CHU3 only
nu60002039002A06_chu13_N_cl.evt      FPMA 06 event file for CHU1 and CHU3
```

Note that the event files for the other CHU1/CHU2/CHU3 combinations are not generated because, for this observation, these configurations were not used.

The sky images corresponding to the CHU3 and CHU1&3 event files generated by *'nusplitsc'* are shown in Figure 6. The images show that in both cases the target source is characterized by a well defined centroid with a PSF consistent with that of a point-like source.

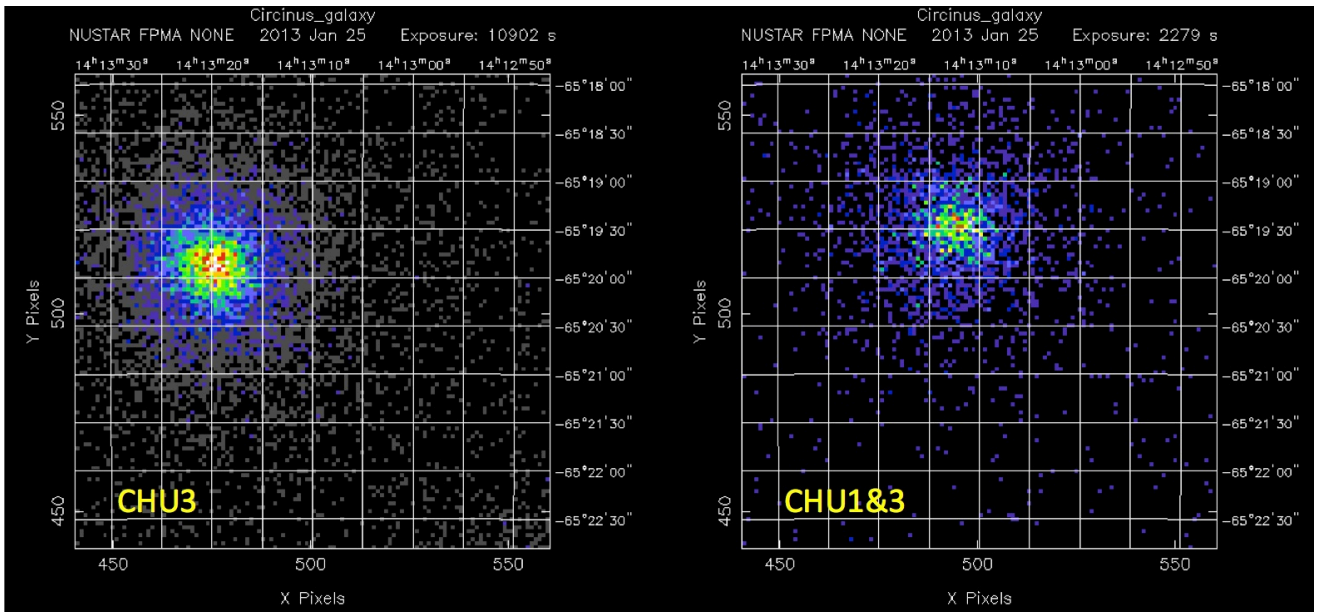

Figure 6: NuSTAR FPMA sky images of the Circinus Galaxy (obsID 60002039002) for observation mode 06 as decomposed according to different CHU1/CHU2/CHU3 combinations using the '*nusplitsc'* software module. In this example the target appears as point-like source for the CHU3 and CHU1&3 combinations.

The CHU3 and CHU1&3 event files generated by *'nusplitsc'* can then be used to extract high-level data products (e.g. energy spectra, light-curves, exposure maps). To this end, the *'nuproducts'* software module with standard command lines can be used. We stress that each CHU1/CHU2/CHU3 combination has a specific aspect reconstruction with a typical scatter of about 2 arcminutes and the optimal extraction region for one CHU combination cannot be used for another combination.

Below the two '*nuproducts'* command lines to be used in the Circinus Galaxy observation are shown:

```
> nuproducts infile=./pipeline_out/nu60002039002A06_chu3_N_cl.evt
  indir=./pipeline_out instrument=FPMA steminputs=nu60002039002
  outdir=./products06 srcregionfile=source_chu3.reg
  bkgregionfile=background_chu3.reg
```

```
> nuproducts infile=./pipeline_out/nu60002039002A06_chu13_N_cl.evt
  indir=./pipeline_out instrument=FPMA steminputs=nu60002039002
  outdir=./products06 srcregionfile=source_chu13.reg
  bkgregionfile=background_chu13.reg
```

where the source extraction region files are centered on the source as it appears in the corresponding decomposed mode 06 sky image.

The energy spectra from different CHU combinations, and the corresponding response files, can be added using multimission FTOOLS. We recommend the use of '*addspec'* or '*addascaspec*' which will combine the source and background PHA files as well as the RMFs and ARFs.

Note that for some observations the decomposed mode 06 event files may still show multiple/elongated centroids. In these cases the user can try to use the more conservative *'STRICT'* splitting algorithm (*'splitmode=STRICT'*) instead of the default one (*'splitmode=NORMAL'*).

7 CALIBRATION FILES

7.1 Introduction

The NuSTARDAS software interfaces with the calibration information via the Calibration Database (CALDB) which consists of a collection of files, each dedicated to a specific aspect of calibration, organized in a specific directory structure. Software retrieves the CALDB files by querying an index file, which contains records of the calibration files and their validity.

The CALDB files for NuSTAR can be divided in three categories:

- Files used by the software to calibrate events, for example, to calculate the PI values, or to record standard screening criteria.
- Files used in the analysis of extracted products, for example, the response matrices.
- Files not used directly by the software that contain instrument characteristics and are included in the calibration database for archival purposes.

The following sections list the NuSTAR CALDB files that are directly used by the NuSTAR software tasks and in the analysis of extracted products.

The calibration files are named according to the following convention:

nu[instrument][datatype][date]v[ver].[ext]

where *[instrument]* indicates the focal plane module, *[datatype]* provides an identifier for the calibration data, *[date]* indicates the first date of validity of the file and *[ver]* is the version number for that file. E.g. *nuCmetgrid20100101v001.fits*.

7.2 List of calibration files used by NuSTARDAS

The table in this section provides a quick reference for the calibration files required by the NuSTAR software tasks. The table contains five columns. The first lists the name of the software task; the second gives a brief description of the calibration files used by the task; the third indicates the Focal Plane Module ('A' for FPMA, 'B' for FPMB and 'C' for both) for which that calibration file is applicable; the fourth contains the *[datatype]* in the filename of the calibration file and/or the file extension (*[ext]*) if different from '.fits'; the last column lists the CALDB subdirectory containing the file.

Software Task	Calibration File Description	[FPM]	[datatype] /[ext]	directory
numetrology	Metrology Grid	C	metgrid	/fpm/bcf/metrology/
	Alignment	C	align	/fpm/bcf/align/
nuattcorr	CHUs Quaternion Offset	C	chuoffset	/fpm/bcf/instrument/
nuflagbad	Ground Bad Pixel	A,B	badpix	/fpm/bcf/badpix/
nuhotpix	-	-	-	-
nucalcpha	Capacitors offset	A,B	offset	/fpm/bcf/cap_offset/
	Event Grade	C	grade	/fpm/bcf/grade/
	Energy Calibration Coefficients	A,B	phapar	/fpm/bcf/instrument/
nucalcpi	Gain coefficients	A,B	gain	/fpm/bcf/gain/
	Charge Loss Coefficients	A,B	clc	/fpm/bcf/clc/
	Fluorescence photons energy boundaries	C	clcfilter	/fpm/bcf/clc/
nuflagevt	Depth Cut Coefficients	A,B	depthcut	/fpm/bcf/instrument/
	Event Cut Coefficients	A,B	evtcut	/fpm/bcf/instrument/
nucoord	Pixel Position Coefficients	A,B	pixpos	/fpm/bcf/pixpos/

	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
nucalcpos	Pixel Position Coefficients	A,B	pixpos	/fpm/bcf/pixpos/
	Alignment	C	align	/fpm/bcf/align/
nuskypos	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
nucalcsaa	SAA Parameters	C	saapar	/fpm/bcf/instrument/
nufilter	makefilter parameters	C	mkconf	/fpm/bcf/instrument/
	prefilter parameters	C	preconf	/fpm/bcf/instrument/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
nuscreen	Housekeeping range values Modes n = 1 to 6	A,B	hkrange0n	/fpm/bcf/instrument/
	GRADE and STATUS range values	A,B	evtrange	/fpm/bcf/instrument/
nulivetime	-	-	-	-
nusplitsc	CHU1/CHU2/CHU3 range values	C	chu123range	/fpm/bcf/instrument/
nuexpomap	Pixel Position Coefficients	A,B	pixpos	/fpm/bcf/pixpos/
	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
	DET1 Instrument Probability Map	A,B	instrprobmap	/fpm/bcf/instrmap/
	Vignetting	A,B	vign	/fpm/bcf/vign/
nuskytodet	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
nuproducts	Pixel Position Coefficients	A,B	pixpos	/fpm/bcf/pixpos/
	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
	DET1 Instrument Probability Map	A,B	instrprobmap	/fpm/bcf/instrmap/
	Ancillary Response	A,B	arf	/fpm/bcf/arf/
	Raytrace 2D PSF	A,B	2dpsfen	/fpm/bcf/psf/
	2D PSF files grouping	A,B	grppsf	/fpm/bcf/psf/
	Vignetting	A,B	vign	/fpm/bcf/vign/
	Aperture Stop ARF Correction	A,B	apstop	/fpm/bcf/aperture_stop/
	Ghost Rays ARF Correction	A,B	gr	/fpm/bcf/ghost_rays/
	Detector Absorption Coefficients	A,B	detabs	/fpm/bcf/detabs/
	RMF files grouping	A,B	grprmf	/fpm/bcf/rmf/
nulccorr	Pixel Position Coefficients	A,B	pixpos	/fpm/bcf/pixpos/
	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
	DET1 Instrument Probability Map	A,B	instrprobmap	/fpm/bcf/instrmap/
	Raytrace 2D PSF	A,B	2dpsfen	/fpm/bcf/psf/
	Vignetting	A,B	vign	/fpm/bcf/vign/
nubackscale	Pixel Position Coefficients	A,B	pixpos	/fpm/bcf/pixpos/
	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
	DET1 Instrument Probability Map	A,B	instrprobmap	/fpm/bcf/instrmap/
numkarf	Pixel Position Coefficients	A,B	pixpos	/fpm/bcf/pixpos/
	Alignment	C	align	/fpm/bcf/align/
	Telescope definition	A,B	teldef	/fpm/bcf/teldef/
	DET1 Instrument Probability Map	A,B	instrprobmap	/fpm/bcf/instrmap/
	Ancillary Response	A,B	arf	/fpm/bcf/arf/
	Raytrace 2D PSF	A,B	2dpsfen	/fpm/bcf/psf/

	2D PSF files grouping	A,B	grppsf	/fpm/bcf/psf/
	Vignetting	A,B	vign	/fpm/bcf/vign/
	Aperture Stop ARF Correction	A,B	apstop	/fpm/bcf/aperture_stop/
	Ghost Rays ARF Correction	A,B	gr	/fpm/bcf/ghost_rays/
	Detector Absorption Coefficients	A,B	detabs	/fpm/bcf/detabs/
numkrmf	RMF files grouping	A,B	grprmf	/fpm/bcf/rmf/

Table 7-1: Calibration Files

8 APPENDIX A: FITS FILE STRUCTURE

8.1 Level 1 File Format

The file structure of the telemetry formatted in FITS format is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	EVENTS	85(20)	Events Extension
2	BINTABLE	GTI	16(2)	GTI Extension

Table 8-1: Level 1 FITS File structure

The columns in the EVENTS extension are:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Event Time (seconds since Jan 1st 2010 00:00:00 UTC)
PRIOR	1D	s	-	-	-	Elapsed Livetime Since Prior Event - seconds.
NUMRISE	1J	-	-	-	-	Numerator of Time of Rise Estimator (17 bits, signed)
DENRISE	1J	-	-	-	-	Denominator of Time of Rise Estimator (14 bits, signed)
SUBFRM	1J	-	-	-	-	Time rel. to 1pps frame-sync: units: 16 1474560
PREPHAS	9I	-	-	-	-	Pre-trigger pulse-height values for central pixel plus 8 surrounding
POSTPHAS	9I	-	-	-	-	Post-trigger pulse-height values for central pixel plus 8 surrounding
RESET	1D	s	-	-	-	Event Time Relative to Most Recent Charge Pump Reset - seconds
TRIGGERS	9X	-	-	-	-	Hardware Triggers - same order as PRE and POST
DET_ID	1B	-	-	-	-	Detector ID (0,1,2,3)
S_CAP	1B	-	-	-	-	Starting Cap # (0..15)
HWTRIG	1B	-	-	-	-	Number of hardware triggers
RAWX	1B	-	-	-	-	X-position of central pixel in raw coordinates (column#: 0..31)
RAWY	1B	-	-	-	-	Y-position of central pixel in raw coordinates (row#: 0..31)
MODE	1B	-	-	-	-	1=CP mode, 0=Normal mode
STIM	1B	-	-	-	-	Stim event flag
UP	1B	-	-	-	-	uP (offset) event flag
SHIELD	1B	-	-	-	-	Shield hit flag
SHLD_T	1B	-	-	-	-	Shield timing (4 bits)
SHLD_HI	1B	-	-	-	-	High shield threshold flag

Table 8-2: Level 1 FITS File Events Table Columns

8.2 Level 1a File Format

The calibrated events FITS File structure is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	EVENTS	286(39)	Events Extension
2	BINTABLE	GTI	16(2)	GTI Extension
3	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET0
4	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET1
5	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET2
6	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET3

Table 8-3: Level 1a FITS File structure

The columns in the EVENTS extension are (the columns in the brackets may not be present, depending on the processing options):

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Event Time (seconds since Jan 1st 2010 00:00:00 UTC)
PRIOR	1D	s	-	-	-	Elapsed Livetime Since Prior Event - seconds.
NUMRISE	1J	-	-	-	-	Numerator of Time of Rise Estimator (17 bits, signed)
DENRISE	1J	-	-	-	-	Denominator of Time of Rise Estimator (14 bits, signed)
SUBFRM	1J	-	-	-	-	Time rel. to 1pps frame-sync: units: 16 1474560
PREPHAS	9I	-	-	-	-	Pre-trigger pulse-height values for central pixel plus 8 surrounding
POSTPHAS	9I	-	-	-	-	Post-trigger pulse-height values for central pixel plus 8 surrounding
RESET	1D	s	-	-	-	Event Time Relative to Most Recent Charge Pump Reset - seconds
TRIGGERS	9X	-	-	-	-	Hardware Triggers - same order as PRE and POST
DET_ID	1B	-	-	-	-	Detector ID (0,1,2,3)
S_CAP	1B	-	-	-	-	Starting Cap # (0..15)
HWTRIG	1B	-	-	-	-	Number of hardware triggers
RAWX	1B	-	-	-	-	X-position of central pixel in raw coordinates (column#: 0..31)
RAWY	1B	-	-	-	-	Y-position of central pixel in raw coordinates (row#: 0..31)
MODE	1B	-	-	-	-	1=CP mode, 0=Normal mode
STIM	1B	-	-	-	-	Stim event flag

UP	1B	-	-	-	-	uP (offset) event flag
SHIELD	1B	-	-	-	-	Shield hit flag
SHLD_T	1B	-	-	-	-	Shield timing (4 bits)
SHLD_HI	1B	-	-	-	-	High shield threshold flag
STATUS	16X	-	-	-	-	Event Quality Flag
BADPOS	8X	-	-	-	-	Position of neighbor bad pixels
HOTPOS	8X	-	-	-	-	Position of neighbor hot/flickering pixels
(RAWPHAS)	9I	-	-	-	-	Pixel Pulse-height raw values
(OFFPHAS)	9E	-	-	-	-	Pixel Pulse-height after offset correction
(TRPHAS)	9E	-	-	-	-	Pixel Pulse-height after time of rise correction
(PHAS)	9E	-	-	-	-	Pulse-height after common mode correction
GRADE	I	-	0	32	-	Event Grade
SWTRIG	9B	-	-	-	-	Event software threshold Flag
PIS_GAIN	9E	chan	-	-	-	Pixel Pulse Invariant before charge loss correction
SURRPI	1J	-	-	-	-	um of pulse invariants for pixels below threshold
PI_CLC	1E	-	-	-	-	Pixel Pulse Invariant after charge loss correction
PI	J	-	0	4095	-	Pulse Invariant
DET1X	1I	pixel	1	360	-	Event X position Focal Plane Bench Frame
DET1Y	1I	pixel	1	360	-	Event Y position Focal Plane Bench Frame
DET2X	1I	pixel	1	600	-	Event X position Optics Bench Frame
DET2Y	1I	pixel	1	600	-	Event Y position Optics Bench Frame
X	1I	-	1	1000	-	Event X position SKY
Y	1I	-	1	1000	-	Event Y position SKY

Table 8-4: Level 1a FITS File Events Table Columns

8.3 Level 2 File Format

The cleaned and calibrated events FITS File structure is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	EVENTS	294(41)	Events Extension
2	BINTABLE	GTI	16(2)	GTI Extension
3	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET0
4	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET1
5	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET2
6	BINTABLE	BADPIX	20(5)	Bad pixels Extension for DET3

Table 8-5: Level 2 FITS File structure

The columns in the EVENTS extension are (the columns in the brackets may not be present, depending on the processing options):

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Event Time (seconds since Jan 2010 00:00:00 UTC)
(PRIOR)	1D	s	-	-	-	Elapsed Livetime Since Prior Event - seconds.
(NUMRISE)	1J	-	-	-	-	Numerator of Time of Rise Estimator (17 bits, signed)
(DENRISE)	1J	-	-	-	-	Denominator of Time of Rise Estimator (14 bits, signed)
(SUBFRM)	1J	-	-	-	-	Time rel. to 1pps frame-sync: units: 16 1474560
(PREPHAS)	9I	-	-	-	-	Pre-trigger pulse-height values for central pixel plus 8 surrounding
(POSTPHAS)	9I	-	-	-	-	Post-trigger pulse-height values for central pixel plus 8 surrounding
(RESET)	1D	s	-	-	-	Event Time Relative to Most Recent Charge Pump Reset - seconds
(TRIGGERS)	9X	-	-	-	-	Hardware Triggers - same order as PRE and POST
DET_ID	1B	-	-	-	-	Detector ID (0,1,2,3)
(S_CAP)	1B	-	-	-	-	Starting Cap # (0..15)
(HWTRIG)	1B	-	-	-	-	Number of hardware triggers
RAWX	1B	-	-	-	-	X-position of central pixel in raw coordinates (column#: 0..31)
RAWY	1B	-	-	-	-	Y-position of central pixel in raw coordinates (row#: 0..31)
(MODE)	1B	-	-	-	-	1=CP mode, 0=Normal mode
(STIM)	1B	-	-	-	-	Stim event flag
(UP)	1B	-	-	-	-	uP (offset) event flag
(SHIELD)	1B	-	-	-	-	Shield hit flag
(SHLD_T)	1B	-	-	-	-	Shield timing (4 bits)
(SHLD_HI)	1B	-	-	-	-	High shield threshold flag
(STATUS)	16X	-	-	-	-	Event Quality Flag
(BADPOS)	8X	-	-	-	-	Position of neighbor bad pixels
(HOTPOS)	8X	-	-	-	-	Position of neighbor hot/flickering pixels
(RAWPHAS)	9I	-	-	-	-	Pixel Pulse-height raw values
(OFFPHAS)	9E	-	-	-	-	Pixel Pulse-height after offset correction
(TRPHAS)	9E	-	-	-	-	Pixel Pulse-height after time of rise correction
(PHAS)	9E	-	-	-	-	Pulse-height after common mode correction
GRADE	I	-	0	32	-	Event Grade

(SWTRIG)	9B	-	-	-	-	Event software threshold Flag
(PIS_GAIN)	9E	chan	-	-	-	Pixel Pulse Invariant before charge loss correction
SURRPI	1J	-	-	-	-	um of pulse invariants for pixels below threshold
(PI_CLC)	1E	-	-	-	-	Pixel Pulse Invariant after charge loss correction
PI	J	-	0	4095	-	Pulse Invariant
DET1X	1I	pixel	1	360	-	Event X position Focal Plane Bench Frame
DET1Y	1I	pixel	1	360	-	Event Y position Focal Plane Bench Frame
DET2X	1I	pixel	1	600	-	Event X position Optics Bench Frame
DET2Y	1I	pixel	1	600	-	Event Y position Optics Bench Frame
X	1I	-	1	1000	-	Event X position SKY
Y	1I	-	1	1000	-	Event Y position SKY

Table 8-6: Level 2 FITS File Events Table Columns

8.4 GTI table FITS Format

All event files have a GTI extension. The format is:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
START	1D	s	-	-	-	GTI Start Time
STOP	1D	s	-	-	-	GTI Stop Time

Table 8-7: GTI Table columns

8.5 Bad Pixel table FITS Format

All calibrated event files have a Bad pixels extension for each Detector ID. The format is:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
RAWX	1B	pixel	0	31	-	X-position in raw detector coordinates
RAWY	1B	pixel	0	31	-	Y-position in raw detector coordinates
TIME	1D	s	-	-	-	Start Time of Bad Pixel Interval
TIME_STOP	1D	s	-	-	-	Start Time of Bad Pixel Interval
BADFLAG	16X	-	-	-	-	Bad Pixel flag

Table 8-8: Bad Pixels Table columns

8.6 Filter File Format

The filter file structure is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	FILTER	115(37)	Filter Extension

Table 8-9: Filter FITS File structure

The columns in the FILTER extension are:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Time at which a value changed
OCCULTED	1B	-	-	-	-	Occulted Flag
SAA	1B	-	-	-	-	In SAA Flag (using flags/data from FPMA and FPMB)
SAA_A	1B	-	-	-	-	Time in SAA (using FPMA HK data only)
SAA_B	1B	-	-	-	-	Time in SAA (using FPMB HK data only)
GEOCOR	1D	GV	-	-	-	Vertical cutoff rigidity
DAY	1B	-	-	-	-	In Daylight Flag
SLEW	1B	-	-	-	-	Slew Flag
ORBIT	1J	-	-	-	-	Orbit number
SOURCE	1B	-	-	-	-	1 = CHU4, 2 = S/C bus, 3 = hybrid twixt 1&2
RESIDUAL	1B	-	-	-	-	CHU 4 Residual
LIVETIME	1D	s	-	-	-	Electronic livetime for current second
SHLDLO	1J	ct/s	-	-	-	Low-gain shield singles rate
SHLDHI	1J	ct/s	-	-	-	High-gain shield singles rate
SW_SAA	1B	-	-	-	-	ground software SAA flag
SW_TENTACLE	1B	-	-	-	-	ground software SAA flag
CAL0STAT	B	-	-	-	-	FPMA Calibration src flag: 0=OUT, 1=IN
CAL1STAT	B	-	-	-	-	FPMB Calibration src flag: 0=OUT, 1=IN
X0_INT	1E	mm	-	-	-	Sum of X laser intensities values for PSD0
Y0_INT	1E	mm	-	-	-	Sum of Y laser intensities values for PSD0
X1_INT	1E	mm	-	-	-	Sum of X laser intensities values for PSD1
Y1_INT	1E	mm	-	-	-	Sum of Y laser intensities values for PSD1
METGRID_FLAG	1B	-	-	-	-	MET Grid flag (=0 in grid, =1

						out of grid, =2 not corrected)
ANG_DIST	1E	deg	-	-	-	angular distance of pointing from nominal
RA	1E	deg	-	-	-	pointing axis right ascension
DEC	1E	deg	-	-	-	pointing axis declination
ROLL	1E	deg	-	-	-	pointing axis roll
ELV	1E	deg	-	-	-	angle between pointing and earth limb
BR_EARTH	1E	deg	-	-	-	angle between pointing and bright earth
COR_SAX	1E	GeV/c	-	-	-	magnetic cut off rigidity
FOV_FLAG	1I	-	-	-	-	0=sky; 1=dark earth; 2=bright earth
MCILWAIN_L	1E	-	-	-	-	McIlwain L parameter (SAX)
MOON_ANGLE	1E	deg	-	-	-	angle between pointing and moon vector
RAM_ANGLE	1E	deg	-	-	-	angle between pointing and velocity vector
SAA_PREFILTER	1I	-	-	-	-	1=in SAA; 0=not
SUNSHINE	1I	-	-	-	-	1=in sunshine; 0=not
SUN_ANGLE	1E	deg	-	-	-	angle between pointing and sun vector

Table 8-10: Filter Table Columns

8.7 Optical Axis File Format

The file structure is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	OPTICAL_AXIS	56(7)	Optical Axis Extension

Table 8-11: Optical Axis FITS File structure

The columns in the OPTICAL_AXIS extension are:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Event Time
X_OA	1D	pixel	-	-	-	Optical Axis X position (SKY Frame)
Y_OA	1D	pixel	-	-	-	Optical Axis Y position (SKY Frame)
DET2X_APSTOP	1D	pixel	-	-	-	Aperture Stop Center DET2X position (OB Frame)
DET2Y_APSTOP	1D	pixel	-	-	-	Aperture Stop Center DET2Y position (OB Frame)
X_APSTOP	1D	pixel	-	-	-	Aperture Stop Center X position (SKY Frame)
Y_APSTOP	1D	pixel	-	-	-	Aperture Stop Center Y position (SKY Frame)

Table 8-12: Optical Axis Table Columns

8.8 DET1 Reference Pixel File Format

The file structure is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	DET1_REFPOINT	24(3)	DET1 Reference Pixel Extension

Table 8-13: DET1 Reference Pixel FITS File structure

The columns in the DET1_REFPOINT extension are:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Event Time
X_DET1	1D	pixel	-	-	-	Detector Reference Point X position (SKY Frame)
Y_DET1	1D	pixel	-	-	-	Detector Reference Point Y position (SKY Frame)

Table 8-14: DET1 Reference Pixel Table Columns

8.9 Mast aspect solution File Format

The file structure is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	MAST_ASPECT	64(3)	Mast aspect solution Extension

Table 8-15: Mast aspect solution FITS File structure

The columns in the MAST_ASPECT extension are:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Time of the mast aspect solution
T_FBOB	3D	mm	-	-	-	Translation from Focal Plane to Optical Plane
Q_FBOB	4D	-	-	-	-	Quaternion from Focal Plane to Optical Plane

Table 8-16: Mast aspect solution Table Columns

8.10 Position Sensing Detector File Format

The file structure is:

HDU	Type	EXTNAME	Dim(col)	Description
0	PRIMARY	-	0	Primary Header
1	BINTABLE	PSDPOS	40(9)	Position Sensing Detector Extension

Table 8-17: Position Sensing Detector FITS File structure

The columns in the PSDPOS extension are:

TTYPE	TFORM	TUNIT	TLMIN	TLMAX	TZERO	Description
TIME	1D	s	-	-	-	Time of laser position measurement
X_PSD0	1E	mm	-	-	-	X-position of laser spot on PSD0
Y_PSD0	1E	mm	-	-	-	Y-position of laser spot on PSD0
X_PSD1	1E	mm	-	-	-	X-position of laser spot on PSD1
Y_PSD1	1E	mm	-	-	-	Y-position of laser spot on PSD1
X0_INT	1E	mm	-	-	-	Sum of X laser intensities values for PSD0
Y0_INT	1E	mm	-	-	-	Sum of Y laser intensities values for PSD0
X1_INT	1E	mm	-	-	-	Sum of X laser intensities values for PSD1
Y1_INT	1E	mm	-	-	-	Sum of Y laser intensities values for PSD1
METGRID_FLAG	1B	-	-	-	-	MET Grid flag (=0 in grid, =1 out of grid, =2 not corrected)

Table 8-18: Position Sensing Detector Table Columns