Starlink Project
Starlink User Note 265.0

A.G. Gibb, T. Jenness, F. Economou

10 Jun 2014

PICARD — a PIpeline for Combining and Analyzing Reduced Data
Version 1.0.0
User's Guide

Abstract

PICARD is a facility for combining and analyzing reduced data, normally the output from the ORAC-DR data reduction pipeline. This document describes an introduction to using PICARD for processing instrument-independent data.

Contents

1 Introduction

The ORAC-DR pipeline (SUN/230) is a suite of recipes and primitives for the automated processing of raw instrument data into scientifically-useable products. However, this is only the start point for the analysis and further operations on these data is inevitable. The **PI**peline for **C**ombining and **A**nalyzing **R**educed **D**ata (PICARD) is a modification to ORAC-DR which allows pipeline-processed data to be manipulated using generic, instrument-independent methods. Furthermore, it is inefficient to begin at the computationally-expensive raw data stage again for every minor adjustment to the analysis. Thus ORAC-DR and PICARD together represent two halves of the data reduction and analysis workflow.

This document will describe the basics of using PICARD and contains a summary of available processing recipes.

1.1 Document conventions

In an attempt to make this document clearer to read, different fonts are used for specific structures.

Starlink package names are shown in small caps (e.g. SMURF); individual task names are shown in sans-serif (e.g. makemap). PICARD recipe and primitive names are also shown in sans-serif and are always upper case (e.g. REMOVE_BACKGROUND).

Text relating to filenames (including suffices for data products), key presses or entries typed at the command line are also denoted by fixed-width type (e.g. `% smurf`), as are parameters for tasks which are displayed in upper case (e.g. `METHOD`).

References to Starlink documents, i.e., Starlink User Notes (SUN), Starlink General documents (SG) and Starlink Cookbooks (SC), are given in the text using the document type and the corresponding number (e.g. SUN/95). Non-Starlink documents are cited in the text and listed in the bibliography.

File name suffices represent the text between the final underscore character and the three-letter `.sdf` extension. For example, a file named `s4a20101020_00002_0001_cal.sdf` has the suffix `_cal`.

2 PICARD overview

PICARD is a tool for analyzing and combining a batch of astronomical data files that have previously had their instrumental signatures removed (for example by running ORAC-DR on the raw data). It is designed to be instrument-independent. PICARD uses the same infrastructure as ORAC-DR, where data are processed by recipes which contain a series of processing steps called primitives.

PICARD is designed to be easy to use. It needs no initialization, has few options and, by default, assumes that all input/output occurs in the current working directory.

2.1 Requirements for running PICARD

ORAC-DR (and thus PICARD) requires a recent Starlink installation. The latest release may be obtained from `http://starlink.eao.hawaii.edu/starlink`. Since ORAC-DR development is an ongoing process, it is recommended that the newest builds be used. These builds can be obtained from: `http://starlink.eao.hawaii.edu/starlink/rsyncStarlink` and may be kept up-to-date with rsync.

The Starlink Perl installation (Starperl) must be used to run the pipeline due to the module requirements. The Starlink environment should be initialized as usual before running PICARD.

2.2 Important environment variables

PICARD does not need to have specific environment variables defined (other than those initialized as part of Starlink). Data are read from and written to the current working directory by default. However, it is possible to define an alternative location for the output data via `ORAC_DATA_OUT` (which is used by ORAC-DR).

Two other specialized environment variables may be defined by users who wish to write their own processing routines: see Section 4 for more information.

2.3 Running PICARD

The only mandatory arguments are the name of the recipe and a list of the files to process. Running PICARD is as easy as typing

```
% picard <options> RECIPE *.sdf
```

where RECIPE is the name of the processing recipe to use and `*.sdf` is the list of files to process. In practice, everything after the recipe name is treated as an input file. The recipe will be applied to all input files, which must be in NDF format. Currently there is no automated conversion from FITS.

More generally:

```
% picard [options] RECIPE FILES
```

where `[options]` are command-line options of the form `-option` or `-option value`. Note that the options must be given before the recipe. The options are described in more detail below.

2.4 PICARD options

PICARD has a number of command-line options which may be used to control the processing and feedback.

General Options

-help

Lists help text summarizing the command usage.

-version

Prints out the pipeline version information.

-man

Displays the full manual page.

-debug

Enable debugging output, listing primitive entry and exit points, timing and calls to algorithm engines.

-verbose

Enable verbose output from algorithm engines.

-files

File name of a flat ASCII text file containing a list of files to be processed, one file per line. Files specified this way are added to the list of files given as command line arguments.

-rechelp

Show help text for the given recipe.

Windows and Output

-log sfhx

Similar to ORAC-DR, this option controls whether the pipeline output is logged to the terminal screen (`s`), log file (`f`), html log file (`h`) or to an X-window (`x`). The default is `fx`. To avoid opening an X-window, `sf` is recommended.

-nodisplay

Do not launch the display system. No data will be displayed, and GWM, GAIA, etc. windows will not be launched.

Recipe Selection

-recsuffix

Modify the recipe search algorithm such that a recipe variant can be selected if available. For example with '`-recsuffix QL`' a recipe named MYRECIPE_QL would be picked up in preference to MYRECIPE.

Multiple suffices can be supplied using a comma separator.

```
-recsuffix QL1,QL2
```

-recpars

Recipe behaviour can be controlled by specifying a recipe parameters file. This is a file in INI format with a block per recipe name.

```
[RECIPE_NAME]
param1=value1
param2=value2
```

See the documentation for individual recipes in Appendix B for supported recipe parameters.

3 Hints and tips

This section lists a handful of useful hints and tips for running PICARD.

- Make sure the .sdf extension is included in the filename if passing in a single file.
- A single recipe parameter file can be used for multiple recipes:

  ```
  [RECIPE1]
  PARAM1 = VALUE1
  PARAM2 = VALUE2

  [RECIPE2]
  PARAM_A = VALUE_A
  PARAM_B = VALUE_B
  ```

- If the environment variable ORAC_DATA_OUT is defined, any files created by PICARD will be written in that location. Check there if new files are expected but do not appear in the current directory.
- A list of files can be given via the -files option, e.g.

  ```
  % picard -log s RECIPE_NAME -files filestoprocess.lis
  ```

  Remember to include the .sdf for each file in the list in the file filestoprocess.lis.
- All data should be from the same instrument. For SCUBA-2 users, this means data from a single wavelength.
- By default PICARD does not know how to display the files produced as part of processing, especially if relying on instrument-specific features. If display is required, make a copy of the file disp.dat located in $ORAC_CAL_ROOT/inst_name where inst_name is the (lowercase) name of the instrument from which the data originated (e.g. scuba2).
- Be as specific as possible when providing the list of input files to avoid the possibility of processing output files from PICARD in subsequent runs: running with *.sdf is generally a bad idea. Creating a text file containg the relevant filenames is best (see above).

4 Writing PICARD recipes and primitives

PICARD allows users to write their own primitives and recipes for processing reduced data, although a number of (mostly SCUBA-2) recipes exist. Interested users are advised to read SUN/230 and SUN/233 for further details. The user can specify the environment variables `ORAC_RECIPE_DIR` and `ORAC_PRIMITIVE_DIR` to point to the locations containing recipes and primitives.

While PICARD is designed to be instrument-independent, it is possible to access methods from supported instrument classes, provided that all the input data are from the same instrument. This is particularly useful for accessing instrument-specific values and methods provided by the calibration class, for example. It also provides access to instrument-specific recipes and primitives.

A Alphabetical list of PICARD recipes

CALC_SCUBA2_AVPSPEC Calculate average bolometer power spectra from SCUBA-2 data

CALC_SCUBA2_FCF Calculate FCFs from SCUBA-2 calibrators

CALC_SCUBA2_NEFD Calculate NEFDs from SCUBA-2 images

CALIBRATE_SCUBA2_DATA Calibrate SCUBA-2 data

COADD_JSA_TILES co-add JSA tiles together by tile number

CREATE_MOMENTS_MAP Creates a moments map from a spectral line cube

CREATE_PNG Create a PNG from the current Frame object.

CROP_SCUBA2_IMAGES Trim images to the defined map area

MOSAIC_JCMT_IMAGES Coadd images produced by JCMT instruments

PICARD_DEMONSTRATOR Simple recipe to test Picard infrastructure

SCUBA2_REMOVE_BACKGROUND Remove a background from images

SCUBA2_CHECK_CAL perform SCUBA-2 calibration checks on standard sources

SCUBA2_CHECK_RMS calculate RMS and NEFD via two methods to compare with ITC

SCUBA2_DISPLAY_PCA Calculate and display properties of PCA components

SCUBA2_JACKKNIFE calculate optimal map using jack-knife noise estimator

SCUBA2_JACKKNIFE_PSF create a scaled PSF from maps with fake sources added

SCUBA2_MAP_PSPEC Calculate the noise power spectrum of a SCUBA-2 map

SCUBA2_MATCHED_FILTER Apply a matched filter to input images

SCUBA2_PHOTOM perform aperture photometry on SCUBA-2 images

SCUBA2_REGISTER_IMAGES Register SCUBA-2 images to a common position

SCUBA2_SASSY analyze a single SASSy field

STACK_JCMT_FRAMES stack images produced by JCMT instruments

UNCALIBRATE_SCUBA2_DATA Undo the calibration for SCUBA-2 images

UNTRIM_JSA_TILES restore JSA tiles to full size

B Specifications of PICARD recipes

The following pages describe the current PICARD recipes in detail.

CALCULATE_EFFICIENCIES
calculate eta_mb and eta_ap from a reduced ACSIS observation

Description:

Currently this only works for ACSIS observations. It will look at pixel 1,1 of the input cube.

Available Parameters

The following recipe parameters can be set via the -recpars

option:

Display :

None.

Output Files :

- Creates an output log file called log.efficiency.

Pod Errors :

Hey! The above document had some coding errors, which are explained below: Around line 20: You forgot a ' =back' before ' =head1'

CALCULATE_STANDARD
Perform standard analysis on JCMT standards

Description:

Currently this only works for ACSIS observations.

Available Parameters

The following recipe parameters can be set via the -recpars

option:

CALCULATE_STANDARD_ALWAYS

Display :

None.

Output Files :

- Creates an output log file called log.standard.

Pod Errors :

Hey! The above document had some coding errors, which are explained below: Around line 20: You forgot a ' =back' before ' =head1'

CALC_SCUBA2_AVPSPEC
Calculate average bolometer power spectra from SCUBA-2 data

Description:

Calculate the average bolometer power spectra from raw SCUBA-2 data. Output files can be displayed with KAPPA linplot.

Notes:

- The input data must be raw SCUBA-2 data.
- Produces one output file per subarray with suffix _avpspec.

Available Parameters

The following parameter can be set via the -recpars option:

DISPLAY

Flag to control the display of power spectra. The recipe will attempt to display spectra by default.

Display :

The power spectrum for each file is displayed if desired, up to a maximum of four. Note that a suitable disp.dat must be present in the output directory, or the environment variable ORAC_DATA_CAL must point to the location of the SCUBA-2 version.

CALC_SCUBA2_FCF
Calculate FCFs from SCUBA-2 calibrators

Description:

Calculate the FCF from reduced SCUBA-2 images of calibration sources. Known calibrators will be recognized and the appropriate flux for the current wavelength will be used. Users may also derive FCFs from non-standard sources provided an input flux is given. See the notes and the description of the recipe parameters for further details.

Notes:

- The input data should be uncalibrated (units of pW). (Calibrated data files may be restored to an uncalibrated state by using the recipe UNCALIBRATE_SCUBA2_DATA.)
- The results of the calculation are printed to the screen and written to two log files, log.fcf which contains the FCFs along with their estimated uncertainties, and log.fit_fcf which contains the parameters derived from the fits to the source.
- Fluxes for non-standard sources may be specified by the FLUX_850 or FLUX_450 parameter.
- Multiple non-standard sources may be processed by the recipe by appending the source name to the relevant flux. The source name should be in upper case with spaces removed. For example, FLUX_850.DGTAU. If source names are not appended, the same flux will be used for all.
- Currently there is no way to specify peak and total fluxes separately for user-provided sources.
- Specifying fluxes for multiple sources will generate a warning about unsupported recipe parameters for all the sources not being processed with the current pass through the recipe. These warnings can safely be ignored.

Available Parameters

The following recipe parameters can be set via the -recpars

option:

APERTURE_RADIUS

Radius of aperture (in arcsec) for masking out source (otherwise 30 arcsec).

AUTOPHOTOM

A flag to indicate whether to use KAPPA autophotom for aperture photometry. If not specified, autophotom will be used. If false, the aperture photometry is carried out using a less-optimized method. Leaving this parameter as the default is highly recommended.

AUTOPHOTOM_INNER
Scale factor (in terms of aperture radius) for inner radius of annulus used for background estimate (default is 1.25).

AUTOPHOTOM_OUTER
Scale factor for outer radius of annulus used for background estimate (default is 2).

FLUX_450
Source flux density at 450 um in Jy. Source-specific values may be given by appending the source name in upper case with spaces removed.

FLUX_850
Source flux density at 850 um in Jy. Source-specific values may be given by appending the source name in upper case with spaces removed.

KEEPFILES
Flag to indicate whether to delete intermediate files created by the recipe. Default is to keep all files.

LOGFILE
Flag to denote whether to write results to a log file at the end of processing. Default is 1 (write a log file).

Display :

No display is used by this recipe.

CALC_SCUBA2_NEFD
Calculate NEFDs from SCUBA-2 images

Description:

A simple PICARD recipe to calculate the noise equivalent flux density (NEFD) from reduced SCUBA-2 images. The NEFD image is written as an NDF component (called nefd) to the original file (or a calibrated version thereof) under the .more.smurf hierarchy.

Notes:

- The input data should be calibrated in mJy/beam, but will be calibrated if necessary.
- The output file from this recipe will have suffix _cal if the input data were uncalibrated. The NEFD image will be in this file (though see KEEPFILES below).
- The median and effective NEFD are printed to the screen.

Available Parameters

The following recipe parameter can be set via the -recpars option:

KEEPFILES

Flag to denote whether to write the NEFD image as an NDF component in the output file. Default is 1 (yes). If set to 0 the the median and effective NEFDs will be derived and printed to the screen only; the image will not be saved.

Display :

None.

CALIBRATE_SCUBA2_DATA
Calibrate SCUBA-2 data

Description:

Calibrate SCUBA-2 data with a given or default FCF. The units of the input data are checked and the appropriate default FCF chosen. The output files have a suffix of _cal.

This recipe may be used to convert between different (known) calibration types, e.g., from mJy/arcsec∗∗2 to mJy/beam and vice versa.

Notes:

- The input data must not have the same units as the desired calibration.
- All input data will have the same calibration applied.
- The FCF FITS header will be updated with the value used to calibrate the data. The units will be in the comment.

Available Parameters

The following recipe parameters can be set via the -recpars

option:

FCF

FCF to use to calculate data. The same value is used for all files. The standard SCUBA-2 FCF will be used if not given.

FCF_CALTYPE

FCF type for determining the output units. May be BEAM (default) or ARCSEC

Display :

None.

Output Files :

- Creates an output file for each calibrated input file with suffix _cal.

CALIBRATE_SIDEBAND_RATIO
Apply sideband ratio corrections

Description:

Apply a sideband ratio correction to an observation. This will check the current sideband, and by default will apply the standard JCMT sideband correction. It will first of all undo any previous sideband correction applied that wrote its value into the SDBDCORR FITS header. The output files have a suffix of _cal.

Currently this only works for RxA3m observations.

Notes:

- The SDBNCORR FITS header will be updated with the value used to calibrate the data.

Available Parameters

The following recipe parameters can be set via the -recpars

option:

SIDEBAND

The sideband to apply the correction for. If not given, the current system sideband will be used.

SIDEBAND_CORR_FACTOR

The correction factor to apply. The caller has to ensure that it is valid for the LO_frequency in the provided fields, as no checking of LO_frequency will be done when this option is used. If this recpar is not given, the correction factor will be calculated from the polynomials stored in the ACSIS-calibration system code.

Display :

None.

Output Files :

- Creates an output file for each calibrated input file with suffix _cal.

COADD_JSA_TILES
Co-add JSA tiles together by tile number

Description:

This recipe takes a set of JSA tiles and groups them by tile number. It then co-adds the tiles for each tile.

CREATE_MOMENTS_MAP
Creates a moments map from a spectral-line cube

Description:

This recipe is used to create a moments map (or multiple moments maps) from a cube. It smooths the cube in frequency and spatial extents, then finds clumps of emission. Everything in the cube not found in a clump is masked out, then the masked cube is collapsed to form the moments map.

Notes:

The Clumpfind algorithm selects the emission by walking down peaks to lower intensities.

Available Parameters

The following parameters can be set via the -recpars option:

BASELINE_ORDER

The polynomial order that will be used when estimating baselines. [1]

FREQUENCY_SMOOTH

The number of channels to smooth in the frequency axis when smoothing to determine baselines. This number should be small (∼10) for narrow-line observations and large (∼25) for broad-line observations. [25]

MOMENTS

The moment maps to create. These are any of the values allowed for the ESTIMATOR parameter to the COLLAPSE task, but in reality this should probably be ' integ' , ' iwc' , and/or ' itd' . Any number of moments can be given in a comma-separated string. [' integ']

MOMENTS_LOWER_VELOCITY

An optional lower velocity in km/s, below which no data will be used when creating the moments map. When it is undefined, the full velocity range is used. [undef]

MOMENTS_SNR

Whether or not to do clump detection on an signal-to-noise cube instead of the signal cube. Enabling this is useful for data taken in varying conditions. [0]

MOMENTS_UPPER_VELOCITY

An optional upper velocity in km/s, above which no data will be used when creating the moments map. When it is undefined, the full velocity range is used. [undef]

SPATIAL_SMOOTH

The number of pixels to smooth in both spatial axes when smoothing to determine baselines. [3]

Output Data :

For each moment chosen through the MOMENTS parameter there is NDF called pg<utdate>_<obsno>_<m where <utdate> is the UT date, <obsno> is the observation number (without any leading zeroes), and <moment> is the moment name. A spectrum from the highest signal in NDF pg<utdate>_<obsno>_sp.

CREATE_PNG
Create a PNG from the current Frame object

Description:

This recipe creates a 256x256 pixel PNG file for each file in the current Frame object. It will only work properly on 1-D or 2-D images, throwing a warning if the input file is neither 1-D nor 2-D.

Notes:

- Creates output files with same name as input, but with extension .png.

CROP_SCUBA2_IMAGES
Trim images to the defined map area

Description:

Trim images from SCUBA-2 to a given size. The image may be trimmed using given spatial dimensions or a statistical estimator to threshold the image. In the former case, the output map is rectangular or circular; in the latter the shape is determined by the values in the image itself.

By default the method employs the map parameters in the FITS header. The map width and height may be overridden with recipe parameters. Note that if a map radius is given (when a circular output image is desired) the height and width are ignored.

The JCMT::MapArea Perl module is used to define a (rectangular) AST Region using the map parameters in the FITS header.

For the statistical estimator, the default is to trim the map excluding data where the exposure time is less than half of the median value.

Notes:

- Creates output file with suffix _crop, one for each input file.
- The statistical method works best on DAISYs and will likely not yield a smooth outline for PONGs.

Available Parameters

The following parameters can be set via the -recpars option:

CROP_METHOD
Method to use for cropping image. May be statistical, rectangle (default) or circle, each of which may be abbreviated to the first four characters.

MAP_HEIGHT
Height of output image in arcsec.

MAP_RADIUS
Radius of output image in arcsec. Overrides existence of MAP_HEIGHT and MAP_WIDTH.

MAP_WIDTH
Width of output image in arcsec.

STATS_COMP
NDF component to use to define the threshold. May be texp to use the exposure time image, or var or err to use the variance or error components.

STATS_ESTIMATOR

Statistical estimator for thresholding the image. May be MEAN, MEDIAN, MODE, MAX, MIN.

STATS_THRESH

Multiplier for STATS_ESTIMATOR above. Default is 1 if not specified (or 0.5 for texp).

Display :

None.

ESTIMATE_IMAGE_ALIGNMENT
estimates image alignment

Description:

This recipe estimates the alignment between the reference and input files.

JSA_CATALOGUE
Create a JSA style catalogue

Description:

This recipe takes a set of maps and produces JSA style 'extent' and 'peak' catalogues for each of them. It uses stilts to manipulate catalogues produced by CUPID FINDCLUMPS.

MAKE_HIPS
Create Hierarchical Progressive Survey

Description:

This recipe combines all input images into a HiPS structure.

Recipe Parameters

HIPS_DIR

The directory into which to write the HiPS strucutre.

In addition the following recipe parameters provide information to

be included in the HiPS properties file:

- HIPS_CREATOR

- HIPS_TITLE

- HIPS_INITIAL_RA (degrees)

- HIPS_INITIAL_DEC (degrees)

- HIPS_INITIAL_FOV (degrees)

MOSAIC_JCMT_IMAGES
Coadd images produced by JCMT instruments

Description:

A simple PICARD recipe combine SCUBA-2 or ACSIS images taking into account the EXP_TIME NDF component. Default behaviour is to combine all images into a single coadd/mosaic. However this can be overridden using the MOSAIC_EACH recipe parameter which will combine images based on the OBJECT in the FITS header.

Creates one or more output files with suffix _mos, using the name of the last file in the list as its base.

Notes:

- When combining all given images (default behaviour), the user should check that their positions do not differ by a large amount. The concept of large may depend on the input images (especially for 450 um SCUBA-2 data) and how much they fill in the region between the extreme positions.
- Creating an image spanning 10-20 degrees will probably result in images many GB in size unless the pixel scale is enlarged.
- For SCUBA-2 images the WEIGHTS and NEFD NDF components are also dealt with (if present).
- If the input data are uncalibrated, it is recommended that they be calibrated before mosaicking (calibrating the coadd/mosaic may not be reliable).

Available Parameters

The following parameters can be set via the -recpars option:

MOSAIC_EACH

Flag to indicate whether the data should be mosaicked by individual object or combined into a single output file. Default is to create a single output file.

MOSAIC_TASK

The mosaicking task to use either wcsmosaic (default) or makemos.

MAKEMOS_METHOD

The image combination method for makemos.

MAKEMOS_SIGMAS

The sigma-clipping threshold if MAKEMOS_METHOD is SIGMAS. Default is 4.

MASK_LOWVAR

Flag to indicate that files should have pixels with anomalously-low variances removed. Default is 0 (do not mask out low-variance pixels).

WCSMOSAIC_METHOD
Rebinning method for wcsmosaic and/or wcsalign. Default is nearest.

WCSMOSAIC_PARAMS
Additional parameters required for certain choices of WCSMOSAIC_METHOD.

Display :

None.

PICARD_DEMONSTRATOR
Simple recipe to test Picard infrastructure

Description:

Write out the name of each file.

SCUBA2_CHECK_CAL
Perform SCUBA-2 calibration checks on standard sources

Description:

Calculate fluxes, FCFs and beam size from a given uncalibrated map of a point source. The results are written to a log file called log.checkcal if desired.

Procedure:

- The images are cropped to the given size (as specified in the FITS headers or via the MAP_HEIGHT and MAP_WIDTH recipe parameters, whihc must be at least twice the diameter of the aperture).
- A background may be fitted and removed. (Optional - only if the REMOVE_BACKGROUND recipe parameter is true.)
- The beam size is determined using KAPPA beamfit.
- FCFs are calculated from the cropped (background-subtracted) image.
- The source flux and its uncertainty are derived from aperture photometry on these images. The background is estimate from an annulus with inner and outer radii of 1.25 and 2.0 times the aperture radius.
- The map is calibrated using either the standard FCF or the one derived above (if the USEFCF recipe parameter is true).
- The noise is calculated from the calibrated map.
- The matched filter is applied to the calibrated map.
- Results are written to a log file, log.checkcal.

By default this recipe only works on known calibration sources. However, the user may specify the source flux at 850 and/or 450 um by using recipe parameters called FLUX_850 and FLUX_450 respectively. The fluxes for different sources may be specified by appending the target name (in upper case with spaces removed), e.g. FLUX_850.HLTAU. (See also CALC_SCUBA2_FCF.)

By default a log file is written containing a variety of information about the data and the values calculated.

Notes:

- The input data must be uncalibrated in order to calculate an FCF from calibrator observations. (The PICARD recipe UNCALIBRATE_SCUBA2_DATA can be used to undo the default calibration.)
- The default behaviour is to leave every file created during the recipe on disk. This may not be desirable - see the KEEPFILES recipe parameter below to reduce the number of output files.

- Re-processing data already processed by this recipe is not recommended.
- If the recipe parameter FITSURFACE_KEEPSURFACE is true, then a file will be created (for each input file) with suffix _surface.
- Documentation for other recipes may list other recipe parameters that appear to be applicable to some of the steps in this recipe, but are not shown due to the possibility of adverse interactions.
- The fits for beam size and FCF calculation are independent and are not guaranteed to use the same parameters. The FIT_GAUSSIAN recipe parameter may be used to enforce a gaussian or non-gaussian fit. By default, a two-component gaussian will be fitted for known calibrators with a signal-to-noise ratio of at least 100. If the signal-to-noise is less than 100, the beam fit will fall back to a single gaussian while the FCF fit will be a single component with an unconstrained radial falloff parameter.

Available Parameters

The following parameters can be set via the -recpars option:

APERTURE_RADIUS
Radius of aperture in arcsec for calculating total flux. The default is 30 arcsec.

BACKGROUND_FITMETHOD
Method to use for removing background. May be fitsurface, findback, plane or dc. Default is fitsurface.

FINDBACK_BOX
Size of the box (in pixels) used by findback. Default is 11.

FIT_GAUSSIAN
Flag to indicate whether or not to force a Gaussian fit to the source when estimating the beam parameters. Default is 1 (fit Gaussian).

FIT_FIXAMP
A flag to denote that the amplitude of the fit to the source should be fixed as the peak value in the map. Default is 0 (amplitude is a free parameter).

FIT_FIXBACK
Specifies the background level to be used in the fit to the source. May be ! to allow the background to float. If not given, the default is either a fixed level of 0 for known calibrators, or the background is left as a free parameter.

FITSURFACE_FITPAR
Up to two values which define either the order of the polynomial (for polynomial) or the number of knots (for spline) in the X and Y directions respectively. A single number means the same value is used for both axes. Default is 2 for polynomial, 4 for spline.

FITSURFACE_FITTYPE
Type of fit to use with fitsurface. May be polynomial or spline. Default is polynomial.

FITSURFACE_KEEPSURFACE
A flag to denote whether or not to keep the fitted surface on disk. Useful for debugging purposes. Default is 0 (do not keep on disk).

FLUX_450
Source flux density at 450 um in Jy. Source-specific values may be given by dot-appending the source name in upper case with spaces removed. For example, FLUX_450.DGTAU.

FLUX_850
Source flux density at 850 um in Jy. Source-specific values may be given by dot-appending the source name in upper case with spaces removed (see above).

KEEPFILES
A flag to indicate whether or not to keep all files produced by the recipe. May be 0 to keep no files, or +1 to keep only files with suffix _crop, _back and _mf. Default is -1 (keep all files).

LOGFILE
Flag to denote whether to write results to a log file at the end of processing. Default is 1 (write log file).

MAP_HEIGHT
Height of map in arcsec after cropping. Must be at least twice the aperture diameter. Default is that in the FITS header.

MAP_RADIUS
Radius in arcsec of the circular region to define the map. Must be at least twice the aperture radius. Overrides the use of MAP_HEIGHT and MAP_WIDTH.

MAP_WIDTH
Width of map in arcsec after cropping. Must be at least twice the aperture diameter. Default is that in the FITS header.

MASK_SOURCE
Flag to denote whether to mask the source before removing the background. Default is 0 (do not mask the source).

NOISE_METHOD
Method used to calculate the noise in the calibrated image. May be VARIANCE to use the variance, MASK to mask out the source and calculate the image-plane standard deviation, or MINIMUM to determine the lowest standard deviation in a series of apertures placed on the image. Default is VARIANCE, and minimum match is supported.

PSF_MATCHFILTER
Name of a file to use as the PSF when applying the matched filter.

REMOVE_BACKGROUND
A flag to indicate whether or not a background should be estimated and removed from the image. Default is 0 (do not remove a background).

USEFCF
Flag to denote whether to calibrate the data using the FCFs derived in this recipe (1) or use standard FCFs (0). Standard FCFs will be used if not specified.

USEFCF_CALTYPE

Calibration type to use if USEFCF is 1. May be ARCSEC, BEAM or BEAMMATCH. Default is BEAM.

Display :

None.

Logfile Format :

The log file contains the following entries: 1. Input filename 2. UT date (YYYY-MM-DDTHH:MM:SS.S) 3. Object name 4. Observation number 5. Wavelength (um) 6. Airmass 7. 225-GHz tau 8. tau at observing wavelength 9. Aperture radius (arcsec) 10. USEFCF flag 11. Flux within an aperture (pW) 12. Uncertainty in derived flux above (pW) 13. Noise derived from variance (mJy/beam) 14. Flux conversion factor derived using the integrated flux within an aperture (Jy/sq arcsec/pW) 15. Uncertainty in above FCF 16. Flux conversion factor derived from peak of Gaussian fit to source (Jy/beam/pW) 17. Uncertainty in above FCF 18. Flux conversion factor derived from peak of Gaussian fit to source after processing with a matched filter (Jy/beam/pW) 19. Uncertainty in above FCF 20. FWHM of main beam (arcsec) 21. Contribution of error beam (per cent) 22. Flag to indicate whether the beam fit was a Gaussian

SCUBA2_CHECK_RMS
Calculate RMS and NEFD via two methods to compare with ITC

Description:

Calculate the RMS and NEFD from an input image to compare with the integration time calculator (ITC). The corresponding quick-look log file are read, if they exist, to obtain NEPs from which RMS and NEFD values may also be derived.

The average NEP and its standard deviation are determined for the observation corresponding to the current file. The FCF is used to convert that to an NEFD and thus an RMS using the length of the observation. The RMS for the map is derived from its error component, and the NEFD computed from this and the exposure time image. Finally the ITC is used to determine the expected noise and NEFD for the integration (elapsed) time.

The results are written to a log file, log.checkrms.

Notes:

- The input map is trimmed to a circle 90 arcsec in radius unless otherwise specified by the recipe parameter below.
- The input files must correspond to single observations, not coadds, because the elapsed time cannot be calculated for coadds. The recipe will print an error if it detects that the input data have been coadded/mosaicked.
- The input data are calibrated in mJy/beam if necessary.

Available Parameters

The following parameters can be set via the -recpars option:

KEEPFILES

Flag to denote whether to delete intermediate files. Default is to keep all intermediates. If set to 1, then only the cropped files will be kept (with suffix _crop). If set to 0 then all intermediate files will be deleted.

MAP_RADIUS

Radius of map in arcsec. Default is 90.

STATS_ESTIMATOR

Estimator for NEFD and RMS values derived from map. May be mean or median (default).

Display :

None.

SCUBA2_DISPLAY_PCA
Calculate and display properties of PCA components

Description:

Apply PCA processing to raw SCUBA-2 data to determine dominant modes.

The input data should contain a fast-ramp flatfield (taken prior to the target data).

Notes:

- Input files must contain raw SCUBA-2 data.
- Input data should be from a single subarray only, and for a single observation. However, no checks are made that this is actually the case.
- If results are to be calculated by the recipe, then all the input data are used. This could lead to long run times with a large number of files as the data are pre-processed with SMURF sc2clean.

Available Parameters

The following parameters can be set via the -recpars option:

LOGFILE

Flag to denote whether to write results to a log file at the end of processing. Default is 1 (write a log file).

PCA_COMP

PCA components to analyze and display. Default is 0 to 5. The components may be specified either as a comma-separated list (e.g. 0,1,2,3 etc - they need not be contiguous or in order), or as a Perl array slice (e.g. 0..3). The number of components must be no more than 8. If more than 8 are given, only the first 8 are used.

PCA_KEEPFILES

Flag to indicate which files should be kept on disk. Default is 1 which keeps the PCA amplitude, component and power spectrum files on disk. A value of 0 deletes all files, while a value of

- 1 indicates that all files should be kept on disk.

PCA_REUSE

Flag to indicate that existing data should be used if present. Default is 1 (reuse).

Display :

The results for each chosen PCA component are displayed in up to two KAPVIEW windows. The left-hand column displays the amplitude scaled between +/-2 sigma, the next column displays the component as a function of time and the third column shows the power spectrum of each component. Each KAPVIEW window can display results for up to 4 PCA components.

SCUBA2_JACKKNIFE
Calculate optimal map using jack-knife noise estimator

Description:

Use a jack-knife method to remove residual low-spatial frequency noise and create an optimal match-filtered output map. The recipe proceeds as follows:

- The input images are coadded to produce a total signal map.
- The observations are divided into two groups (with alternate files going into each group) which are coadded separately. These coadds are subtracted from one another to create the jack-knife map.
- The azimuthally-average angular power spectrum of the jack-knife map (which should consist purely of noise) is calculated and used to remove residual low-spatial frequency noise from the signal map and the given (map-filtered) psf. This is the so-called whitening step (because it produces a map which has a noise power spectrum that is white).
- The whitened signal map is processed with a matched filter using the whitened psf image as the psf.
- The jack-knife map is also whitened and processed with the matched filter. This map should consist purely of noise.
- Signal-to-noise ratio maps are created for the filtered versions of the signal map and the jack-knife map.

The outcome (the match-filtered whitened signal map with suffix _mf) should be the optimal map with white noise properties. This is the map to be used for science goals.

Notes:

- Ideally there should be an even number of observations, but this is not important if the number of input files is large.
- A fuller description of the procedure may be found in the documentation for the SCUBA-2 recipe REDUCE_SCAN_FAINT_POINT_SOURCES_JACKKNIFE.
- It is recommended that the PICARD recipe SCUBA2_JACKKNIFE_PSF be run to create a suitable PSF to use for this recipe.

Available Parameters

The following parameters can be set via the -recpars option:

JACKKNIFE_METHOD
Method for creating jack-knife map. May be alternate to use every other file to create the two halves, or half to use the first N/2 files (by date) for one half of the jack-knife and the remainder for the other. Default is alternate.

PSF_BOX
Size of square region (in pixels) use to define effective PSF.

PSF_MATCHFILTER
Name of a file to use as the map-filtered PSF.

STATS_COMP
Name of component to use when determining the threshold level. Default is texp (the EXP_TIME component).

STATS_ESTIMATOR
Statistical estimator to use to determine threshold level. May be max, mean, median, or min. Default is median.

STATS_THRESH
Threshold multiplier - the threshold will be this value multiplied by the estimator. Default is 0.5 if using the exposure time, 1 otherwise.

WHITEN_BOX
Size of the region used to calculate the angular power spectrum for removing residual low-frequency noise in the data. Default is a square region bounded by the noise being less than twice the minimum value.

WHITEN_ESTIMATOR
Statistical estimator to determine the threshold level to define the size of the whitening region. May be MIN, MEAN or MEDIAN. Default is MIN (see WHITEN_BOX).

WHITEN_THRESH
The threshold multiplier at which to define the size of the whitening region. Default is 2 (see WHITEN_BOX).

Display :

None.

SCUBA2_JACKKNIFE_PSF
Create a scaled PSF from maps with fake sources added

Description:

This recipe combines a series of maps that have had artificial gaussians added in and scales the coadd using information about the original scaling to enable the FCF to be corrected for filtering in the map-maker.

The input files should be those created by a call to the SCUBA-2 pipeline recipe REDUCE_SCAN_FAINT_POINT_SOURCES_JACKKNIFE (or one of its aliases) that have had the artificial gaussian added at the map-making stage. These will have a suffix of _mappsf.

The PSF is scaled using a factor read from the FITS headers of the input files if available, or from a recipe parameter (see below).

This PSF may then be given to SCUBA2_JACKKNIFE via the recipe parameter PSF_MATCHFILTER.

Notes:

- Care should be taken to use the (_mappsf) files that correspond to the signal maps.
- The output PSF has suffix _effpsf.
- To ensure consistency, the PSF scaling factor in the FITS header takes precedence over a value derived from any given recipe parameters.

Available Parameters

The following recipe parameters can be set via the -recpars

option:

FAKEMAP_CONSTSNR

A flag to indicate whether the scale factor below should be scaled by the square-root of the number of files. Should be left unset unless it was also specified when the ORAC-DR recipe was originally run.

FAKEMAP_SCALE

Amplitude of the fake source (in Jy) added to the timeseries to assess the map-making response to a point source.

Display :

None.

SCUBA2_MAPSTATS
report image statistics and calibrator analysis

Description:

Perform basic image analysis for input maps, plus determine beam size and FCFs from calibrators.

The results are written to the log files log.mapstats and log.calstats (known calibrators only).

Notes:

- The image statistics are calculated over the rectangular area defined by the requested map size in the FITS headers MAP_HGHT, MAP_WDTH and MAP_PA. For SCUBA2 maps this will normally be a square region in the centre of the map.
- The input files must correspond to single observations, not coadds, because the elapsed time cannot be calculated for coadds. The recipe will print an error if it detects that the input data have been coadded/mosaicked.
- The results are quoted in the same units as the input file. Note, however, the log file will contain the units corresponding to the first entry.
- See also SCUBA2_CHECK_CAL and SCUBA2_CHECK_RMS.

Available Parameters

The following parameters can be set via the -recpars option:

STATS_ESTIMATOR

Estimator for exposure time, NEFD and RMS values derived from map. May be mean or median (default).

Display :

None.

SCUBA2_MAP_PSPEC
Calculate the noise power spectrum of a SCUBA-2 map

Description:

Calculate the azimuthally-averaged spatial power spectrum of a SCUBA-2 map. The map is trimmed to a given size (rectangular or circular) before having the source emission masked out. The power spectrum is calculated and smoothed with a 5-pixel boxcar. The peak and half-power points are determined and reported to the user in terms of angular scale on the sky. Finally the mean power on various scales is calculated and reported from the beamsize up to the length scale defined by the filtering when the map was made.

Notes:

Creates a 1-dimensional output file for each input file with suffix _pspec.

Available Parameters

The following parameters can be set via the -recpars option:

APERTURE_RADIUS
Radius of aperture used to mask out a source at the map centre (if MASK_METHOD = aperture). Default is 30 arcsec.

CROP_METHOD
Method to use for trimming image. May be rectangle or circle.

MAP_HEIGHT
Height of output map in arcsec.

MAP_RADIUS
Radius of output map in arcsec (if CROP_METHOD=circle).

MAP_WIDTH
Width of output map in arcsec.

MASK_METHOD
Method for masking out source emission. May be aperture or snr (default).

SNRCUT
Signal-to-noise ratio cut to apply to mask out source emission (if MASK_METHOD=snr). Default is 3 if not specified.

VERBOSE
Flag to indicate more information should be written to the screen. Default is 0 (do not write extra).

Display :

None.

SCUBA2_MATCHED_FILTER
Apply a matched filter to input images

Description:

Apply a matched filter to SCUBA-2 images with the aim of enhancing point sources. The given images are convolved with a PSF, which the user can supply or is created by the recipe. Before the convolution, the maps and the PSF are smoothed with a gaussian, and these smoothed versions are subtracted from the unsmoothed versions.

The default PSF is a two-component gaussian model of the JCMT beam with relative amplitudes and FWHM as given in the SCUBA-2 calibration paper (Mairs et al. 2021, AJ, 162, 191).

Notes:

- Cropping the images before applying this filter to remove large-scale junk around the edge can improve results.
- Input data should all be able to use the same PSF image (if specified).
- Creates an output file for each input file with suffix _mf.
- If no PSF supplied, creates a PSF file for each input file with suffix _psf unless the KEEPFILES recipe parameter is false.

Available Parameters

The following parameters can be set via the -recpars option:

KEEPFILES

A flag to indicate that the PSF created by this recipe should remain on disk after processing. If not specified, the PSF will be deleted if one is created. This parameter is ignored if a PSF file is given (see PSF_MATCHFILTER).

PSF_MATCHFILTER

Name of an NDF file containing a suitable PSF. Must exist in the current working directory. If not specified, the recipe will calculate one itself for each input file.

PSF_NORM

Normalization scheme used for the PSF created by this recipe if one is not specified using the above parameter. Maybe be PEAK or SUM to indicate whether the Gaussian PSF should have a peak of unity or a sum of unity. If not specified, the recipe assumes PEAK.

SMOOTH_DATA

Flag to denote whether or not the image and PSF should be smoothed and have the smoothed version subtracted from the original. If not specified, the recipe assumes a value of 1 (smooth and subtract).

SMOOTH_FWHM

FWHM of Gaussian used to smooth data and PSF images before convolving with the PSF. If not specified the recipe assumes 30 arcsec.

Display :

None.

SCUBA2_PHOTOM
Perform aperture photometry on SCUBA-2 images

Description:

Perform aperture photometry on SCUBA-2 images using the chosen method. There are three methods:

Use the Starlink AUTOPHOTOM package to perform aperture photometry using the given aperture and annulus dimensions (default). The default aperture radius is 30 arcsec.Calculate the sum within a given aperture, correcting for any DC offset by analyzing the image outside the aperture.As above, but estimate the background offset using an annulus (see the parameters ANNULUS_INNER and ANNULUS_OUTER below).The results are written to a log file called log.flux.

Notes:

- It is assumed that the images can be used as is with no further requirement for cropping or background removal.
- Input data should be calibrated - the recipe does not apply any calibration.

Available Parameters

The following parameters can be set via the -recpars option:

ANNULUS
Flag to denote whether to use an annulus for background estimation.

ANNULUS_INNER
Inner radius for annulus as a multiplier of the aperture radius.

ANNULUS_OUTER
Outer radius for annulus as a multiplier of the aperture radius.

APERTURE_RADIUS
Radius of aperture in arcsec for calculating total flux.

AUTOPHOTOM
Flag to denote whether to use the autophotom package for photometry.

REGISTER_DEC
Declination of position of aperture (DD:MM:SS format).

REGISTER_RA
Right ascension of position of aperture (HH:MM:SS format).

STATS_ESTIMATOR

Background estimator for aperture photometry. Default is median.

Display :

None.

SCUBA2_REGISTER_IMAGES
Register SCUBA-2 images to a common position

Description:

Register SCUBA-2 images to a common sky position. The position may be specified, however, the WCS SkyRef attribute is used if the source is a calibrator; (0,0) is used for images in offset coordinate systems. If no reference position has been established, the recipe finds the brightest peak in the image and attempts to use that. All subsequent images will then be registered to the first.

A fit is performed at the peak position in each image, using the reference as an initial guess, and the tangent-plane X,Y offsets between the peak and reference positions are calculated. The offsets are applied to the WCS. Note that only a linear shift is performed; this recipe is not a full astrometry matching routine and does not deal with rotations between images.

The reference position is assumed to be RA/Dec coordinates if given in sexagesimal format, otherwise Galactic (in degrees).

Notes:

- Creates an output file for each input file with suffix _reg.
- A reference position should always be given for non-calibrators.
- The reference position should be that of a known source in each image, and that source must be present in all images.
- Supported coordinate systems are GAPPT, J2000 (including FK5, ICRS) and Galactic.
- The coordinate system of the reference position does not have to match that of the images to be registered.

Available Parameters

The following parameters can be set via the -recpars option:

REGISTER_IMAGES

Flag to indicate that the given images should all be shifted to a common position. No action will be taken if this flag is false (0).

REGISTER_X

X coordinate of reference position. May be Right Ascension (in HH:MM:SS.S format) or Galactic longitude (in decimal degrees).

REGISTER_Y

Y coordinate of reference position. May be Declination (in DD:MM:SS.S format) or Galactic latitude (in decimal degrees).

Display :

None.

SCUBA2_REMOVE_BACKGROUND
Remove a background from SCUBA-2 images

Description:

Fit and remove a background from one or more SCUBA-2 images. This recipe will work best on simple images, such as those containing a single, compact source near the map centre.

Notes:

- Creates an output file for each input file with a suffix <_back>.
- The background estimate is likely to be poor for images that contain extended sources.

Available Parameters

The following parameters can be set via the -recpars option:

MASK_SOURCE

Flag to denote whether to mask the source before removing the background. Default is 0 (do not mask the source).

APERTURE_RADIUS

Radius of aperture (in arcsec) used to mask out source. Default is about twice the beamsize.

BACKGROUND_FITMETHOD

Method to use for removing background. May be fitsurface, findback, plane or dc. Default is fitsurface.

FITSURFACE_FITTYPE

Type of fit to use with fitsurface. May be polynomial or spline. Default is polynomial.

FITSURFACE_FITPAR

Up to two values which define either the order of the polynomial (for polynomial) or the number of knots (for spline) in the X and Y directions respectively. A single number means the same value is used for both axes. Default is 2 for polynomial, 4 for spline.

FITSURFACE_KEEPSURFACE

A flag to denote whether or not to keep the fitted surface on disk. Useful for debugging purposes. Default is 0 (do not keep on disk).

FINDBACK_BOX

Size of the box (in pixels) used by findback. Default is 11.

Default values are those used if the parameter is not specified.

Display :

None.

SCUBA2_SASSY
Analyze a single SASSy field

Description:

Analyze individual maps of SASSy fields, combine them into a single coadd and apply a matched filter before running a source-detection algorithm. Detected sources are written to a CUPID catalogue file with suffix _cat. Statistics are written to a log file called log.sassy.

The statistics are calculated within the area defined by the MAP_HGHT and MAP_WDTH FITS headers, or by equivalent recipe parameters (below).

See the documentation for the SCUBA2_MATCHED_FILTER recipe for matched-filter-specific parameters which may also be specified.

Notes:

The image is trimmed to a circle of radius 4500 arcsec before applying the matched filter.

Available Parameters

The following parameters can be set via the -recpars option:

CROP_METHOD

Method to use for cropping image. May be statistical, rectangle (default) or circle, each of which may be abbreviated to the first four characters.

LOGFILE

A flag to indicate whether or not a log file (called log.sassy) should be written to disk. Default is 1 (yes).

MAP_HEIGHT

Map height in arcsec. Default is to use the value in the FITS header.

MAP_RADIUS

Radius of output image in arcsec. Overrides existence of MAP_HEIGHT and MAP_WIDTH.

MAP_WIDTH

Map width in arcsec. Default is to use the value in the FITS header.

Display :

None.

STACK_JCMT_FRAMES
Stack images produced by JCMT instruments into a 3-d cube

Description:

Stack SCUBA-2 or ACSIS images into a 3-d cube with time as the third axis.

By default the recipe will write out a separate output file for each UT date in the list of input files. SCUBA-2 data will also be sorted by the shutter setting. The user may give a list of additional FITS headers for collating the input files.

The user may also provide the name of an NDF extension which will be stacked instead of the top-level data component (e.g. NEP).

Notes:

- Creates output files based on the name of the first file in the stack with suffix _stack, unless there is only 1 file to stack.
- The given FITS header keywords must exist in every file, and are not validated before accessing.

Available Parameters

The following parameters can be set via the -recpars option:

NDF_EXTEN

The name of an NDF extension to stack, rather than the top-level data structure. It must be located under the .more.smurf hierarchy, and no check is made that it exists before attempting to access it.

STACK_KEYS

A list of FITS header keywords to be used to sort the files before stacking. Only files with matching FITS header values will be used in the stack.

Display :

None.

UNCALIBRATE_SCUBA2_DATA
Undo the calibration for SCUBA-2 images

Description:

Undo the applied SCUBA-2 calibration. The units of the input data are checked and either the FCF in the FITS header (if present) or the appropriate default FCF is chosen. The output files have a suffix of _uncal, the units are set to pW, and the FITS header item FCF is removed if present.

Available Parameters

None.

Display :

None.

Output Files :

Creates an output file for each calibrated input file with suffix _uncal.

UNTRIM_JSA_TILES
Restore JSA tiles to full size

Description:

This recipe takes JSA-tiled data files and untrims them so that they each cover the whole area of the corresponding JSA tiles.