EUROPEAN SOUTHERN OBSERVATORY

Organisation Européenne pour des Recherches Astronomiques dans l'Hémisphère Austral

Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre

VERY LARGE TELESCOPE

FORS2 SPECTROPOLARIMETRY EDPS-GUI tutorial

1 rev. 20260622

Issue 1

Date June 22 2026

Prepared:	Danuta Dobrzycka	June 22 2026	.....
	Name	Date	Signature
Approved:			.....
	Name	Date	Signature
Released:			.....
	Name	Date	Signature

This page was intentionally left blank

Change record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
1	22/06/2026	all	First public release for 5.8.5

This page was intentionally left blank

Contents

1 Introduction

1.1 Scope

This document describes how to reduce FORS SPECTROPOLARIMETRY data with the `edps-gui` (Graphic User Interface), the dashboard of the ESO Data Processing System (`EDPS`), which is the recommended interface to reduce data from ESO telescopes. Details on the FORS SPECTROPOLARIMETRY data reduction stream and how to configure the reduction to meet specific scientific needs are also given.

For a more extensive documentation on the `edps-gui` itself, consult the dedicated manual here.

For a description of the FORS SPECTROPOLARIMETRY pipeline itself, consult the pipeline manual available at:
https://www.eso.org/sci/software/pipe_aem_table.html.

Note: this tutorial refers to:

- FORS SPECTROPOLARIMETRY instrument pipeline named `fors`, version 5.8.5.
- FORS SPECTROPOLARIMETRY workflow: `fors.fors_pmos_wkf`
- `EDPS` version 1.5.7.
- `edps-gui` version .91.

1.2 What is `EDPS`?

The ESO Data Processing System (`EDPS`) is a framework to run ESO's data processing pipelines and it is meant to eventually replace the previous ESOReflex environment. The general principles of `EDPS` have been described by Freudling, Zampieri, Coccato et al. [2024, A&A, 681, A93]. Please refer to that paper if you have used `EDPS` for research resulting in a scientific publication.

Each of ESO's data processing pipeline consists of a series of standalone programs called *recipes*. Each recipe is designed to process certain type(s) of input data. The processing of these input data typically requires a range of auxiliary files such as calibration files. `EDPS` is designed to select appropriate input data for the different recipes of a pipeline, and execute them in sequence. This is done by specifying for each pipeline the workflow for organizing data and executing the recipes. Then, the workflow can be used to process a set of data fully automatically.

1.3 Main concepts

`EDPS` is an environment designed to execute the recipes of an instrument pipeline according to a series of instructions. The main concepts in `EDPS` are:

- **Workflow** and **reduction cascades**. A workflow is a series of instructions designed to reduce data with an instrument pipeline in potentially multiple ways, by carrying on a sequence of tasks. Each workflow can

define multiple reduction cascades, depending on the scientific needs. For example, the same workflow can be used to process data following different strategies that trigger different reduction steps (e.g. in one strategy flux calibration can be omitted) or different end-points (e.g., combine different science exposures, or stop after the reduction of individual exposures without combining them). Each of these "strategies" defines a "reduction cascade".

- **Task**, **jobs**, and **recipes**. A task is an element in the workflow that performs a given step of the data reduction cascade. Tasks are often associated to a recipe of the underlying instrument pipeline. A jobs is a work unit in a processing environment, that runs a recipe on a set of input data with a set of recipe parameters. A single task can generate several jobs: for example, a "bias" task, can generate multiple jobs, each of the running the bias recipe on a different set of input files.
- **Dataset**. A dataset is a collection of files, that are needed to perform the data reduction as specified by the workflow. It consists, for example, of one or more science files plus the calibrations needed to process them. In `EDPS`, datasets have a hierarchical structure, which highlights the connections between the various files and tasks (e.g., task A is an input to task B).
- **Target** and **Target category**. The "target", or the "target task" is the end point of the reduction cascade. When specifying a target, `EDPS` will process all and only the files needed to execute it. For example, if my target is "science", and the science files need the bias files, `EDPS` will process only the biases that have been selected to process those science files; then it processes the science using the product of the bias reduction. However, if my target is bias, then `EDPS` will process all and only the bias files, regardless they are not used by any science. In this case, `EDPS` does not process the science, as it has already reached the end reduction point (e.g., process all biases). The "Target category" is a group of targets that have similar purposes. For example, the target category "science", includes all the tasks that deliver final scientific products, the target category "qc1calib" includes all and only the tasks that processes calibrations (e.g., bias, flat fields, standard stars).

1.4 Installation

1.4.1 Prerequisites

Prerequisites for a well functioning installation of `EDPS` and `EDPS`-gui are:

- Recent Firefox or Chrome browser, Python 3.11 or higher (but there are issues with Python 3.14).
- At least one ESO pipeline with EDPS workflow should be in your system. To install the desired ESO pipelines, follow the instructions in the ESO pipelines pages. NOTE: the `apptainer` installation method is currently not supported. After the installation, the esorex command must be in the path. To test whether the installation was successful, type

  ```
  esorex --recipes
  ```

  A list of available recipes should appear.
- Install `graphviz`, `fv`, and `ds9`, which have to be included in the system path (defining aliases not enough). On linux, `Graphviz` can be easily installed via:

```
sudo apt install graphviz (Debian, Ubuntu)
sudo dnf install graphviz (Fedora)
```

Check the `Graphviz` webpage for installation instructions for other OS.

`fv` and `ds9`, are optional. To install them, follow the instructions in corresponding webpages. You can test whether these three packages are installed and their paths are correctly set by typing on a terminal:

```
dot -V
fv  -version
ds9 -version
```

1.4.2 Installation steps

To install EDPS follow these steps:

- Create a new Python virtual environment and activate it:

  ```
  python3 -m venv edpsgui
  . edpsgui/bin/activate
  ```

  Make sure the python3 version is 3.11 or higher, but not 3.14.

- Install the required packages:

  ```
  pip install --extra-index-url \
      https://ftp.eso.org/pub/dfs/pipelines/repositories/stable/src \
      edps edpsgui edpsplot adari_core
  ```

To run the `edps-gui` type from a terminal (with the active environment):

```
edps-gui
```

Important note. The first time `edps-gui` is executed, you will be asked to specify the directory where the reduction products (fits files and quality plots) will be stored. The default location is `$HOME/EDPS_data`. During the first execution, a configuration file named `application.properties` will also be saved in the directory (newly created) `$HOME/.edps`.

2 Reducing demo data

Follow this procedure to quickly reduce FORS SPECTROPOLARIMETRY demo data. We assume that the `EDPS`, `edps-gui`, the FORS SPECTROPOLARIMETRY pipeline and its associated demo data are installed in your system. For general instructions on how to install `EDPS` and the pipeline, see Section 1.4 or please visit: https://www.eso.org/sci/software/pipe_aem_main.html.

2.1 Setting the workflow

Proceed as follows:

1. If not done already, activate the `EDPS` virtual environment, defined during installation (Sect. 1.4).
2. Start the `edps-gui` dashboard by typing:

   ```
   edps-gui
   ```

   The `edps-gui` dashboard will start in a browser window (Figure 1).

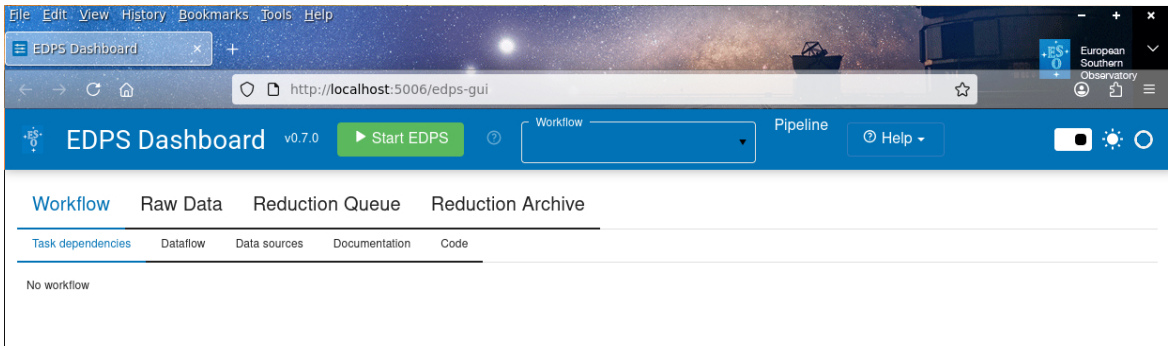

Figure 1: The empty `edps-gui` Dashboard; the underlying `EDPS` engine has not yet been started and no workflow has been loaded.

3. Optionally, before starting `EDPS`, one can specify new settings by pressing Help –> Settings (Figure 2).

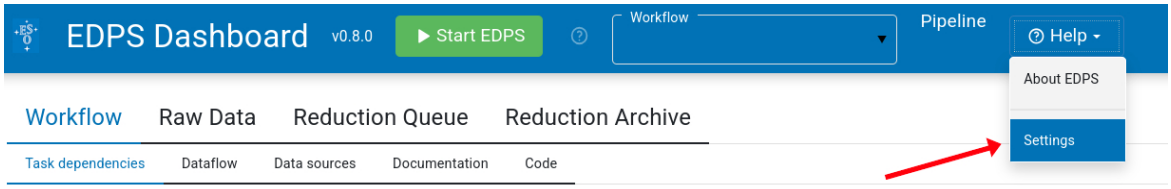

Figure 2: The “Help” –> “Settings” menu.

4. On the browser window with the dashboard, press the button ‘Start EDPS‘.
5. Choose the `fors.fors_pmos_wkf` workflow from the list in the ’Workflow’ field. The workflows offered in this selector depend on the installed pipelines. The graphic workflow representation will appear as in Figure 4.

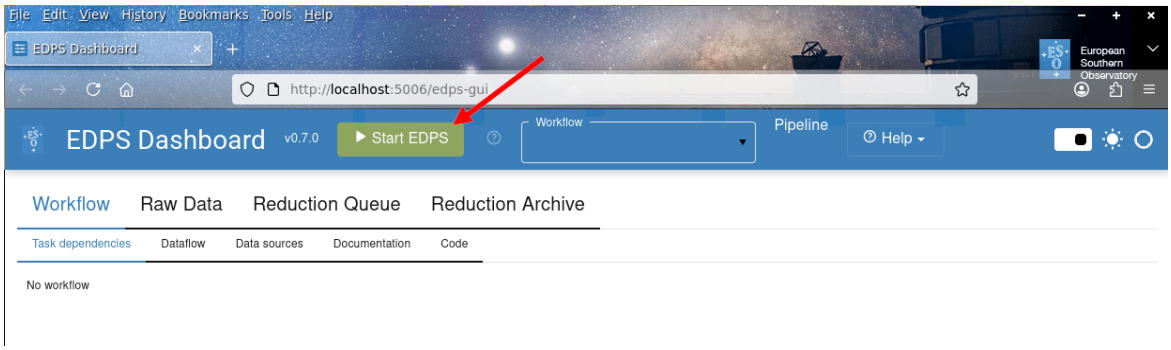

Figure 3: The “Start EDPS” button.

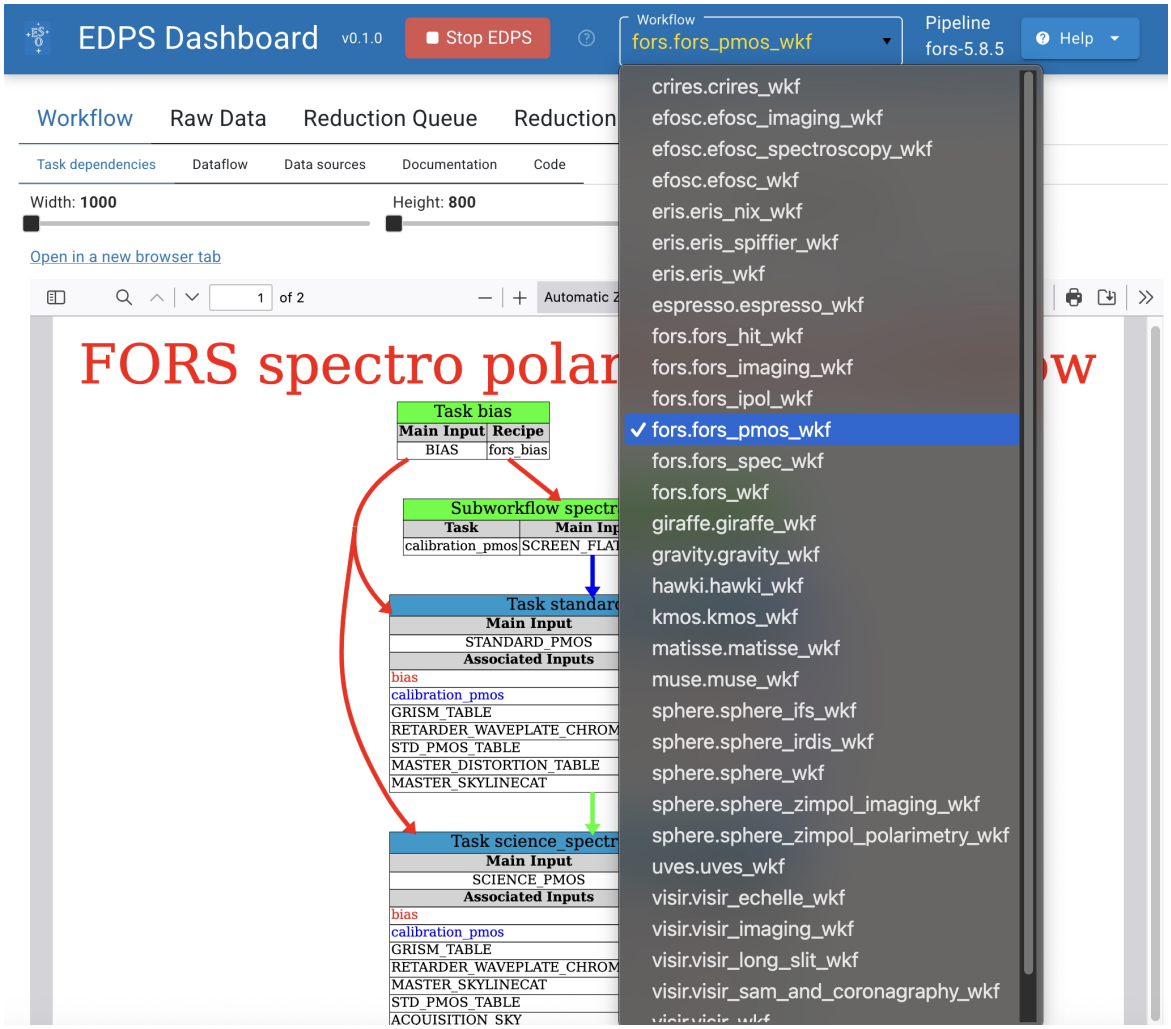

Figure 4: The `edps-gui` with the FORS SPECTROPOLARIMETRY workflow loaded.

2.2 Selecting the input data

1. Press ‘Raw Data‘ to enter the corresponding tab, as in Figure 5.

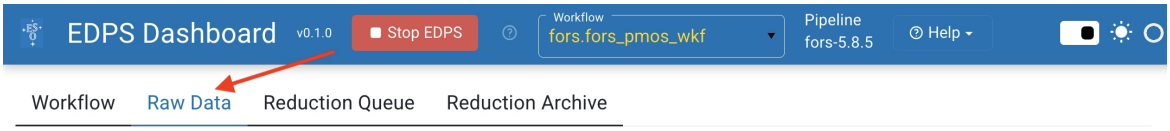

Figure 5: How to select RAW data Tab.

2. Press ‘Select Inputs‘. A selection window will appear that allows to select data that are stored on a local disk (Figure 6).

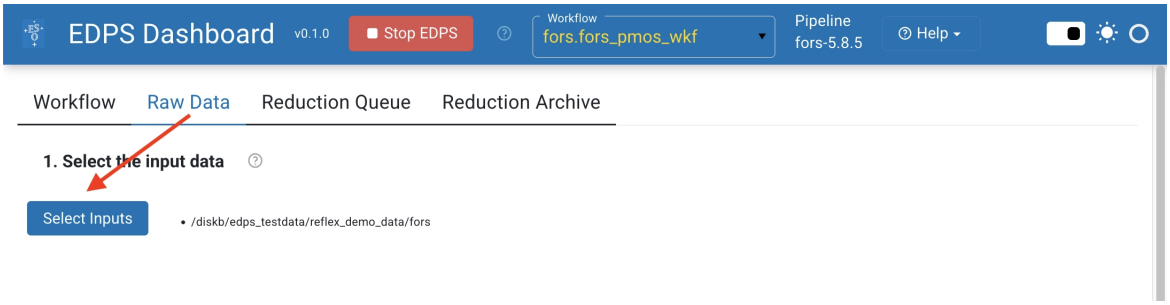

Figure 6: How to select input data.

3. (Optional). Select the reduction target, configure the workflow parameter and specify the association preferences. These steps are optional. For more information see Section **??**.

4. Press ‘Create Datasets‘. A list of datasets appears, one line for each set of science data (Figure 7).

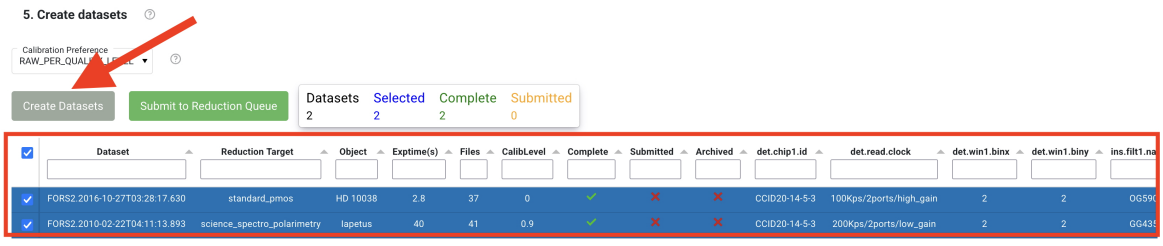

Figure 7: How to inspect the input data directory to create datasets.

5. Choose the datasets that should be processed (Figure 8)

   and send them to the data reduction queue by pressing ‘Submit to Reduction Queue‘. Note that this action does not start the reduction automatically.

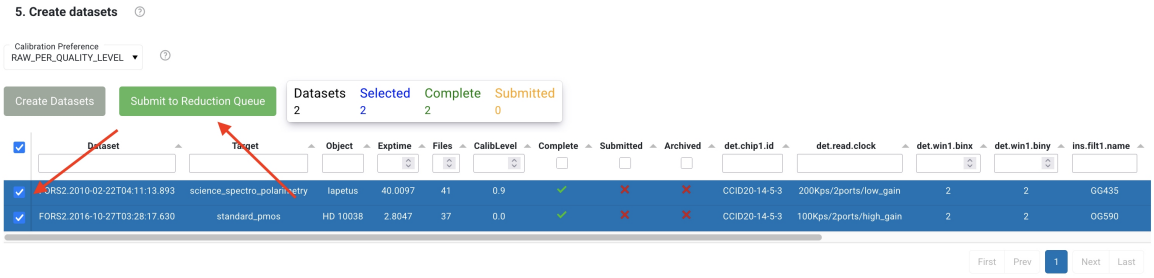

Figure 8: How to send the selected datasets to the Reduction Queue for processing.

2.3 Start the reduction

1. Press the ‘Reduction Queue‘ tab (Figure 9).

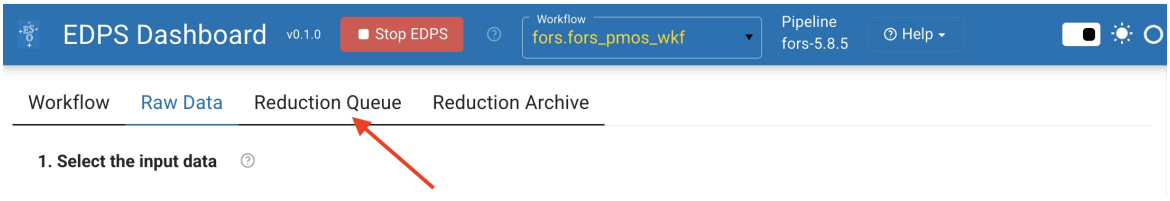

Figure 9: How to select Reduction Queue tab.

2. Select the datasets you’d like to reduce.

3. (Optional). Configure the workflow and recipe parameters by pressing the wheel button  to open the configuration editor. See Section 4.2 for more information on the configuration editor.

4. Press the ‘Reduce‘ button (Figure 10). The selected data will now be processed with the configured parameters.

Congratulations! You reduced your first data with the `EDPS` dashboard! All the reduced data are saved in the EDPS_data directory specified when executing `edps-gui` for the first time.

2.3.1 Quality plots

It is possible to inspect the information on each job, such as quality plots showing the products (for the most important jobs and products), the list of inputs and output files, the recipe parameters and logs. All the information on the job processing can be inspected from the ‘Reduction Queue‘ window. While quality plots are produced only for completed jobs, the other relevant information is available also for failed jobs.

The information associated for the main product can be inspected by pressing the magnifying glass symbol at the right side of each dataset. To inspect the information associated to other individual jobs (e.g., calibrations), proceed as follows:

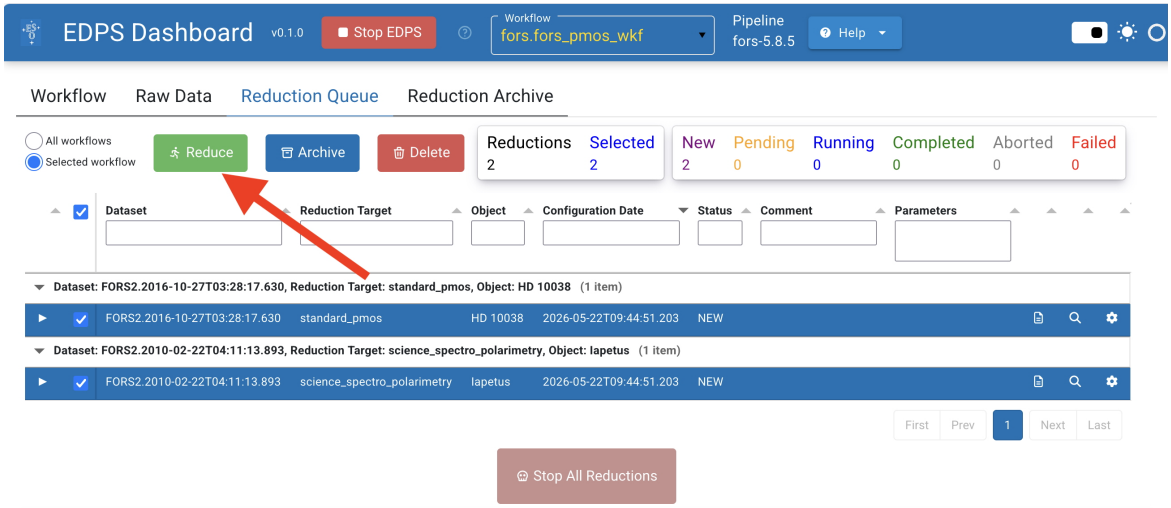

Figure 10: Reduce.

- Expand the desired dataset by pressing the black arrow on its left. The list of jobs will appear with the associated status (COMPLETED, RUNNING, PENDING, MISSING, ABORTED, FAILED)

- Press the magnifying glass symbol at the right side of the job you want to inspect.

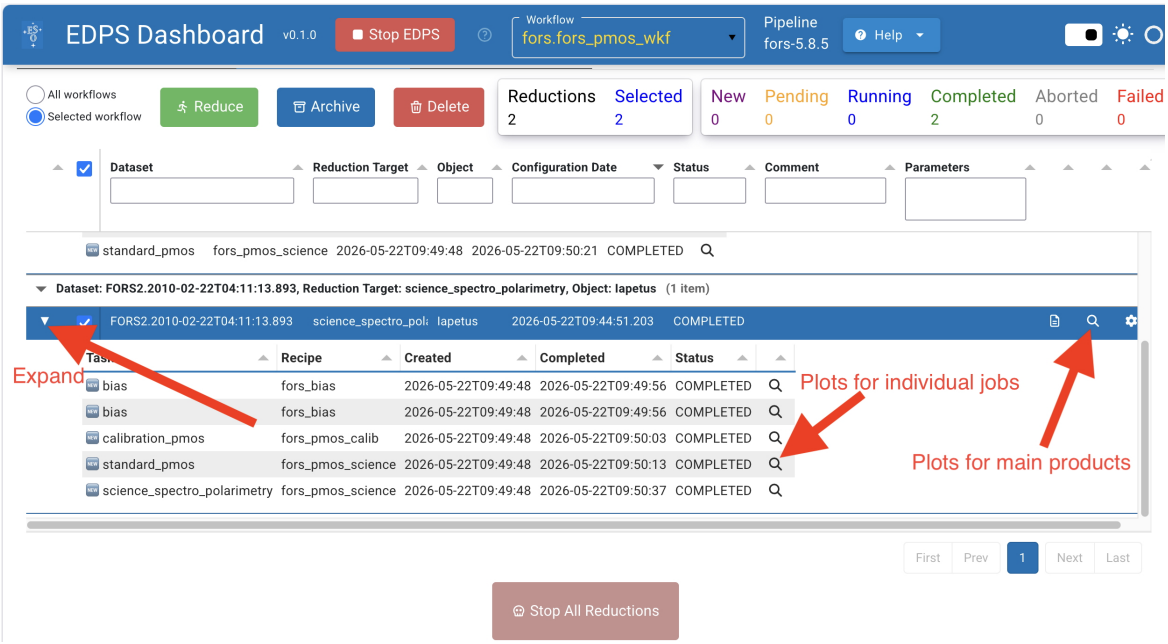

Figure 11: How to look for job information from the Reduction Queue tab.

2.4 Exporting the final products

Completed reductions can be 'Archived' (i.e. declared 'completed' because no more work is needed) and removed from the Reduction Queue. Additionally, even if all products for all tasks are saved in the EDPS_data directory, the most important products can be 'exported' to a desired location.

To do so, proceed as follows:

1. In the 'Reduction Queue' tab, select the dataset and the dataset for which you want to export the final products, and press the 'Archive' button.

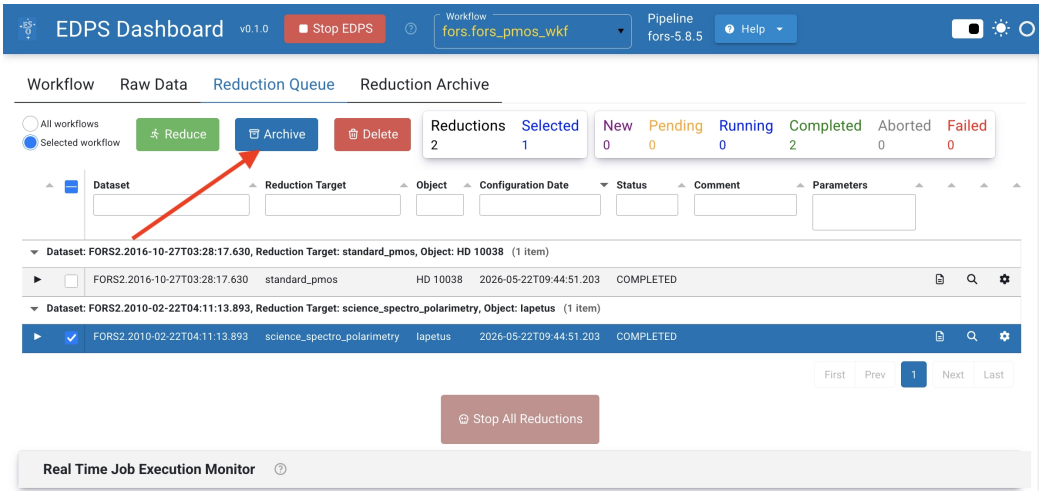

Figure 12: How to archive a completed reduction from the Reduction Queue tab.

2. Go in the Reduction Archive tab and click on the 'Export' button. A new tab window appears where you can indicate the directory you want to copy your final products; finally press "Export" to copy the data.

Exported products are organized by 'DATASET' (named as the first scientific exposure of the dataset), and 'TIMESTAMP' (time of start of reduction)

The final products saved in the specified directory are:

- Multi-extension science fits file, where the individual spectra are provided as rows. There are as many extensions as input scientific exposures. Their name format is SPECTRUM_PMOS_ followed by the exposure identifier (header keyword 'arcfile').
- Fits files with or without extensions and fits tables. Their name format is e.g. STD_ followed by the product identifier (header keyword 'pro.catg')

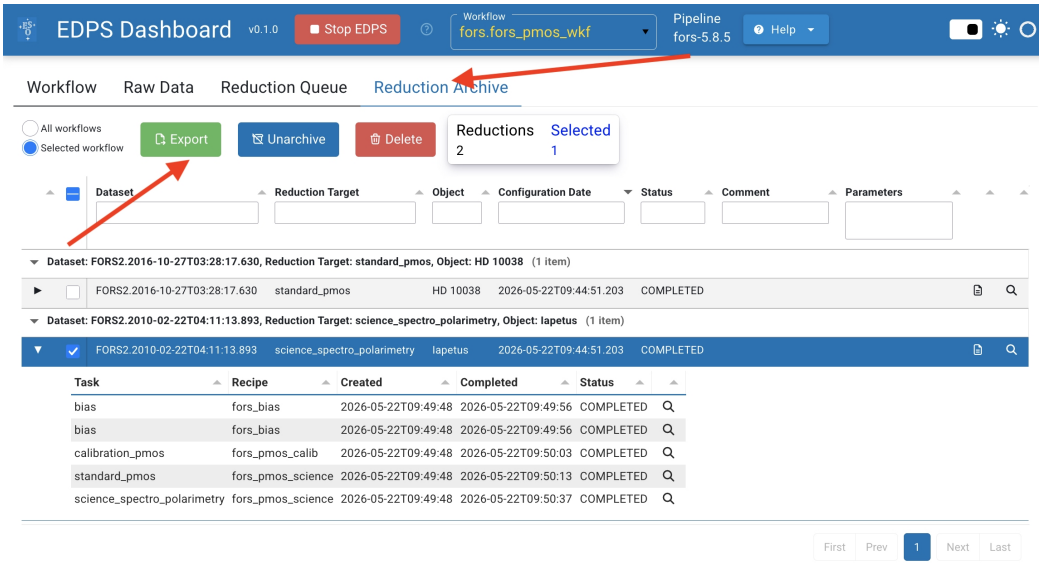

Figure 13: The reduction archive tab. This table contains all the different configurations of datasets that are declared "finished" and removed from the Reduction Queue. From this page, the user can export the most important files into a desired local directory.

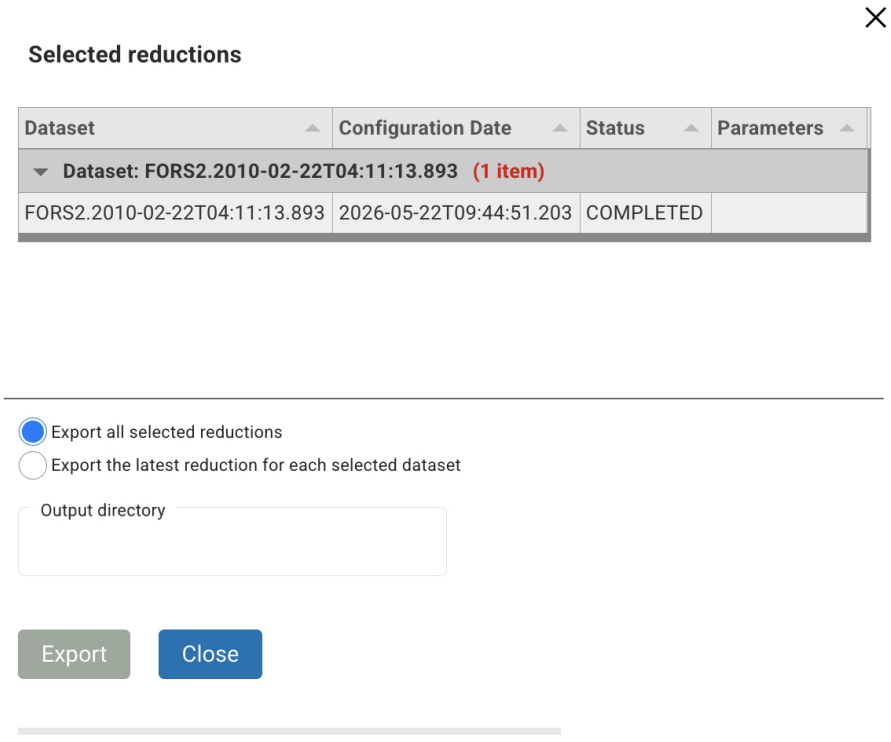

Figure 14: The EXPORT dialogue window, where the user can decide which reduced configuration to save and where.

3 The FORS SPECTROPOLARIMETRY data reduction flow.

The spectropolarimetric mode of FORS (PMOS) is enabled by a retarder waveplate (half-wave or quarter-wave retardance between the fast and slow axes) followed by a Wollaston prism (the "analyzer") which splits the light beam into two orthogonal polarizations. In order for the two beams not to overlap on the detector, a strip mask is formed by every second MOS slit jaw carrier arm across the field of view of the instrument as shown in Figure 15.

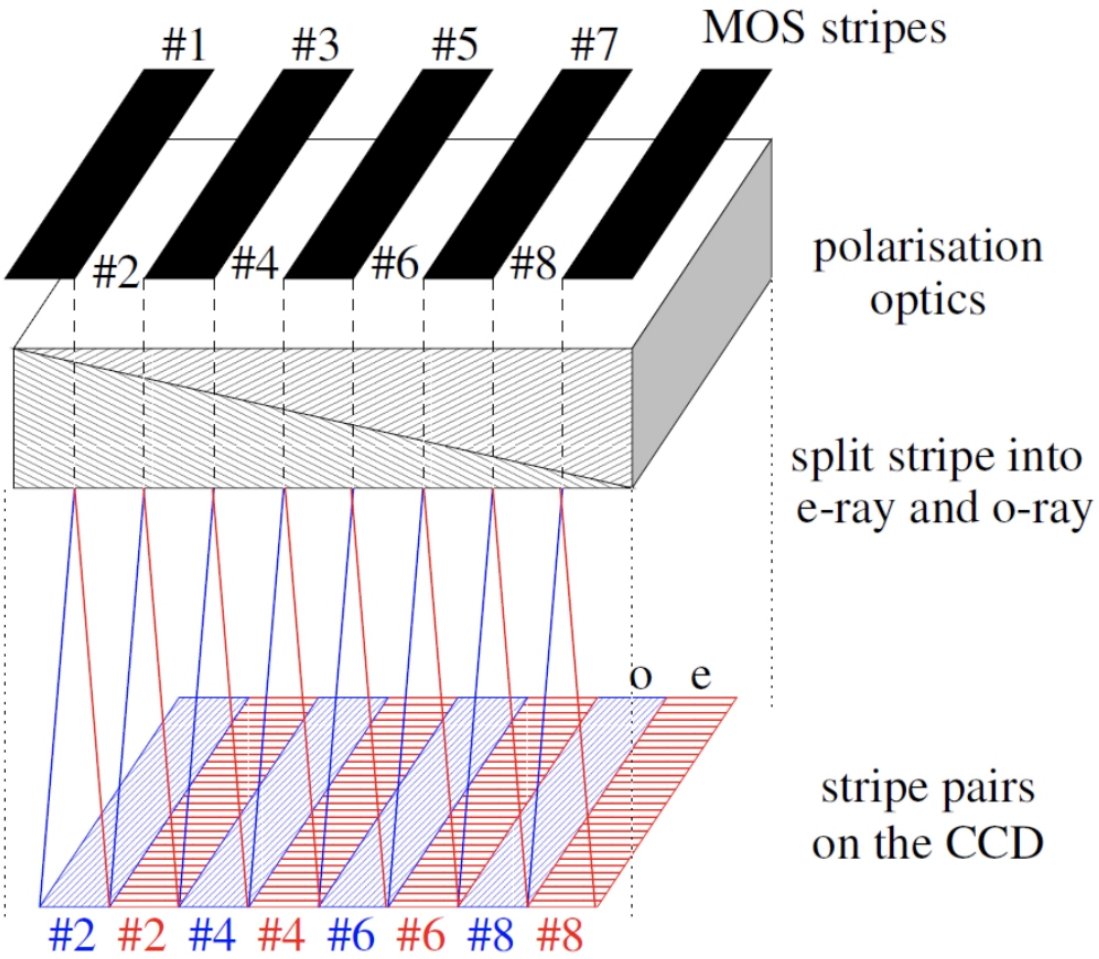

Figure 15: Strip mask formed in the focal plane to allow recording of the ordinary o-ray and extraordinary e-ray beams on the detector. In the even numbered strips, MOS slits can be moved to a particular target position. For a single point source one strip is sufficient.

The MOS slits can be positioned in the free strips along the dispersion axis to desired target locations. Frames are recorded at different retarder plate angles in order to allow the computation of the Stokes parameters for the spectra (Bagnulo S., et al., 2009, PASP, 121, 993 7).

The overall data flow of the FORS SPECTROPOLARIMETRY pipeline is displayed in Figure 16.

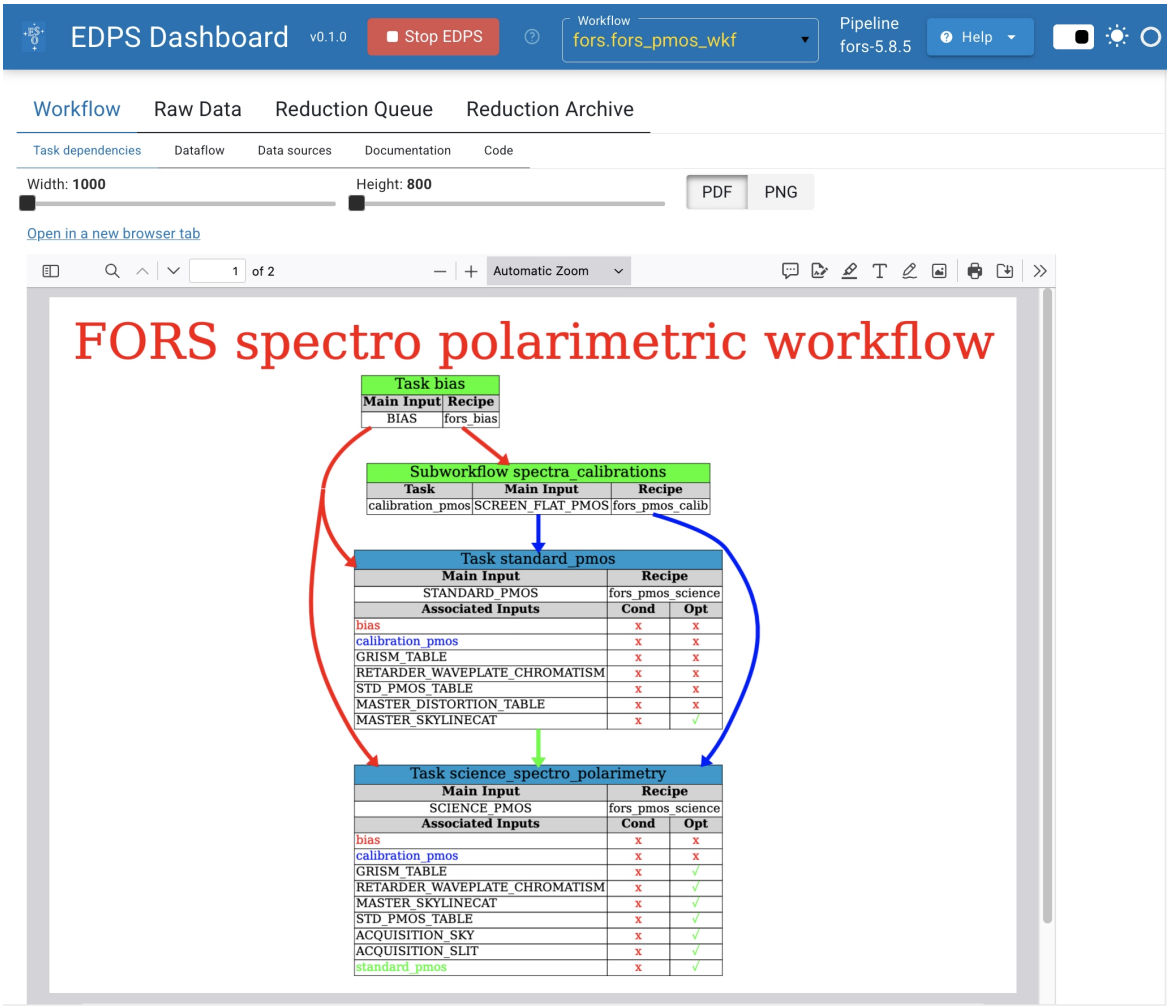

Figure 16: The data reduction cascade of the FORS SPECTROPOLARIMETRY workflow.

The reduction cascade is organized in tasks, which represent well-defined steps in the process. Tasks can be grouped inside sub-workflows. Each task runs a recipe; the detailed description of the algorithms, input, outputs and recipe parameters used in each recipe are available in the pipeline manual. Here, we present only the description of most important features.

The `fors.fors_pmos_wkf` EDPS workflow is designed to execute the tasks that deliver the final reduced data cube for each dataset. It can be either the product of a single exposure, or the combination of multiple exposures. Only calibrations needed by the selected the scientific exposures are processed.

It is possible to set `EDPS` to perform the data reduction until a certain step of the reduction chain (e.g. to reduce only standar stars, or only flat fields). This is done by specifying the desired tasks in the field **Select reduction target** of the **Raw Data** tab.

The reduction steps of the `fors.fors_pmos_wkf` workflow are listed below. Before starting the reduction, the parameters of the recipes associated to each task can be configured by pressing the button  close to each

dataset configuration. See for more info on the configuration editor 4.2

3.1 Creation of the Master Bias

Task: **bias** Recipe: **fors_bias**

In this step the combined MASTER_BIAS is produced.

The raw bias frames are taken with an exposure time of 0 seconds and a closed shutter. They thus record only the signal that is added during the read-out of the CCD to avoid negative numbers. A sequence of 5 or 20 raw bias frames is taken the day following the observations as part of the FORS calibration plan (the calibration plan was changed in November 2014; since then 20 bias frames have been taken per setting instead of 5).

Figure 17 shows an example of the quality plot for the MASTER_BIAS. The image should be uniform, without visible flux gradient. You can also display another extension, IMAGE.ERR for the estimated error per pixel information. See Section 4.3 for more information on how to improve the product if needed.

3.2 Creating the master Flat Field, determining coefficients for wavelength calibration and correction of spatial distortion

Task: **calibration_pmos** Recipe: **fors_pmos_calib**

In this one step, the input screen flat raw calibration frames: SCREEN_FLAT_PMOS together with the PMOS arc lamp raw frame: LAMP_PMOS are used to create a normalized master flat-field, a 2D dispersion solution, correction for spatial distortion, and to determine the slit limits.

The purpose of the flat-field is to remove the pixel-to-pixel sensitivity variations across the detector. As these variations act as a noise source, the precision of the flat-field correction will have direct consequences on the polarimetric precision that can be achieved, and on the signal-to-noise ratio of the reduced observations. For the spectroscopic modes one will use internal screen flats in most cases. These flats are taken during daytime with the telescope pointing to zenith and the instrument in calibration position. The flats are taken at zero retarder plate angle.

For the wavelength calibration the He, HgCd, and Ar lamps (at the lowest spectral resolution with grism 150I) and in addition the Ne lamp (at higher resolution) are used. Wavelength calibration exposures are done during the day only and at all retarder plate angles used for the science observations.

Products:

- CURV_COEFF_PMOS - table with the coefficients of the spatial curvature fitting polynomials
- CURV_TRACES_PMOS - table with the y CCD positions of the detected spectral edges at different x CCD positions.
- DELTA_IMAGE_PMOS - deviation from the linear term of the fitting wavelength calibration polynomials. This is a multi-extension FITS file, with one image extension for each input arc lamp exposure.

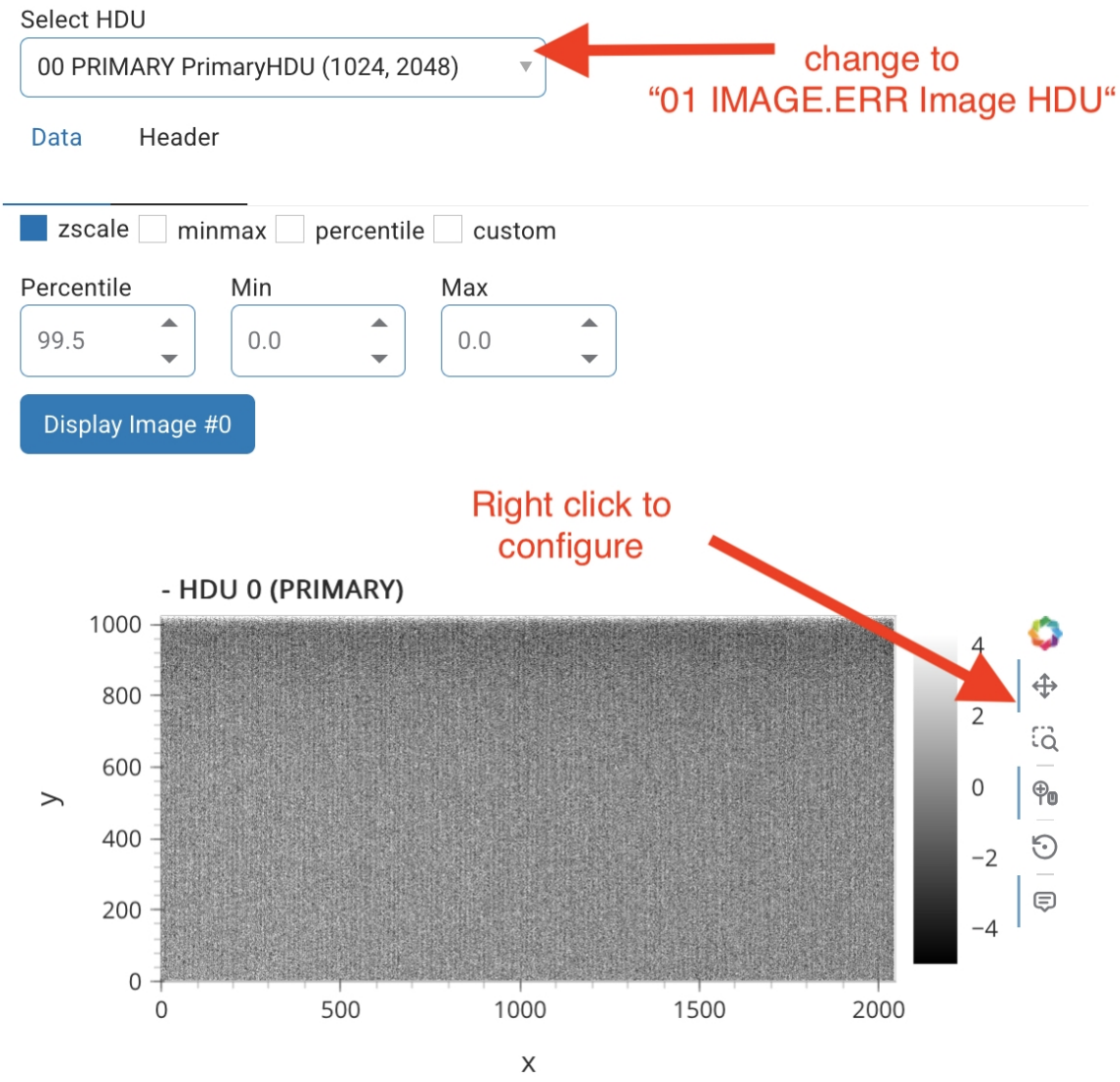

Figure 17: The quality plot of the MASTER_BIAS.

- DISP_COEFF_PMOS - tables with the wavelength calibration polynomial coefficients. This is a multi-extension FITS file, with one table extension for each input arc lamp exposure. Each table contains as many rows as in the REDUCED_LAMP_PMOS images, ordered in the same way.

- DISP_RESIDUALS_PMOS - residuals of each wavelength calibration fit (in pixels). These images are only created and inserted in different extensions of a FITS files if the –check configuration parameter is set.

- DISP_RESIDUALS_TABLE_PMOS - tables containing different kinds of residuals of a sample of wavelength calibration fits.

- MAPPED_NORM_FLAT_PMOS - rectified and wavelength calibrated normalised screen flat field image
- MAPPED_SCREEN_FLAT_PMOS - rectified and wavelength calibrated master screen flat field image
- MASTER_NORM_FLAT_PMOS - normalised flat field image, derived dividing the master screen flat by its smoothed version. Comparing this image with the MASTER_SCREEN_FLAT_PMOS may give an immediate feeling of the goodness of the computed curvature model used for the extraction of the normalised spectra.
- MASTER_SCREEN_FLAT_PMOS - combined flat field image. It is the sum of all the input screen flat fields.
- REDUCED_LAMP_PMOS - rectified and wavelength calibrated arc lamp images, inserted in different extensions of an output FITS file.
- SLIT_LOCATION_PMOS - slit positions, both on the CCD and on the rectified images of the arc lamp exposures (REDUCED_LAMP_PMOS).
- SLIT_MAP_PMOS - map of central wavelength on the CCD. This image is only created if the –check configuration parameter is set
- SPATIAL_MAP_PMOS - map of spatial positions on the CCD.
- SPECTRA_DETECTION_PMOS - result of the preliminary wavelength calibration applied to the arc lamp exposure. This image is only created if the –check configuration parameter is set.
- SPECTRAL_RESOLUTION_PMOS - Mean spectral resolution for each reference arc lamp line. This is a multi-extension FITS file, with one extension for each input arc lamp frame.
- WAVELENGTH_MAP_PMOS - map of wavelengths on the CCD. This is a multi-extension FITS file, with one extension for each input arc lamp frame. Each image has the same size of the CCD, where each pixel has the value of the wavelength at its center, if available.

The important thing in this step is to check the quality of the wavelength calibration, detection of the slitlets, the flat field combination and normalization. Figure 18 shows how to examine various products of the fors_pmos_calib recipe.

The wavelength calibration can be checked looking at the SPECTRA_DETECTION_PMOS product (see Figure 19). In this product the arc lamp lines should run straight from top to bottom without any empty rows between them. Particular attention should be given to lines at the blue and red ends of each spectrum, where the polynomial fit is more sensitive to small variations of the signal. Some arc lines may show gaps due to the placement of the slits, but empty rows without any lines point towards problems with the detection of the arc lamp lines. See Section 4.4 for further information on how to improve the wavelength calibration if necessary.

The quality of the flat field and detection of the slitlets can be checked inspecting e.g. the SPATIAL_MAP_PMOS or the MASTER_NORM_FLAT_PMOS products (see Figure 20). The spatial map has as pixel values the distance of a pixel from the bottom of the respective slitlet. The regions of the slitlets should not be strongly curved nor should regions of different slitlets overlap with each other. Because all MOS slitlets have similar length the spatial map should show a similar gradient for all. Note, that it is not the case for the third slitlet from the top. See Section 4.4 for tips on how to fix such a problem. All the slitlets should be detected and there should be no

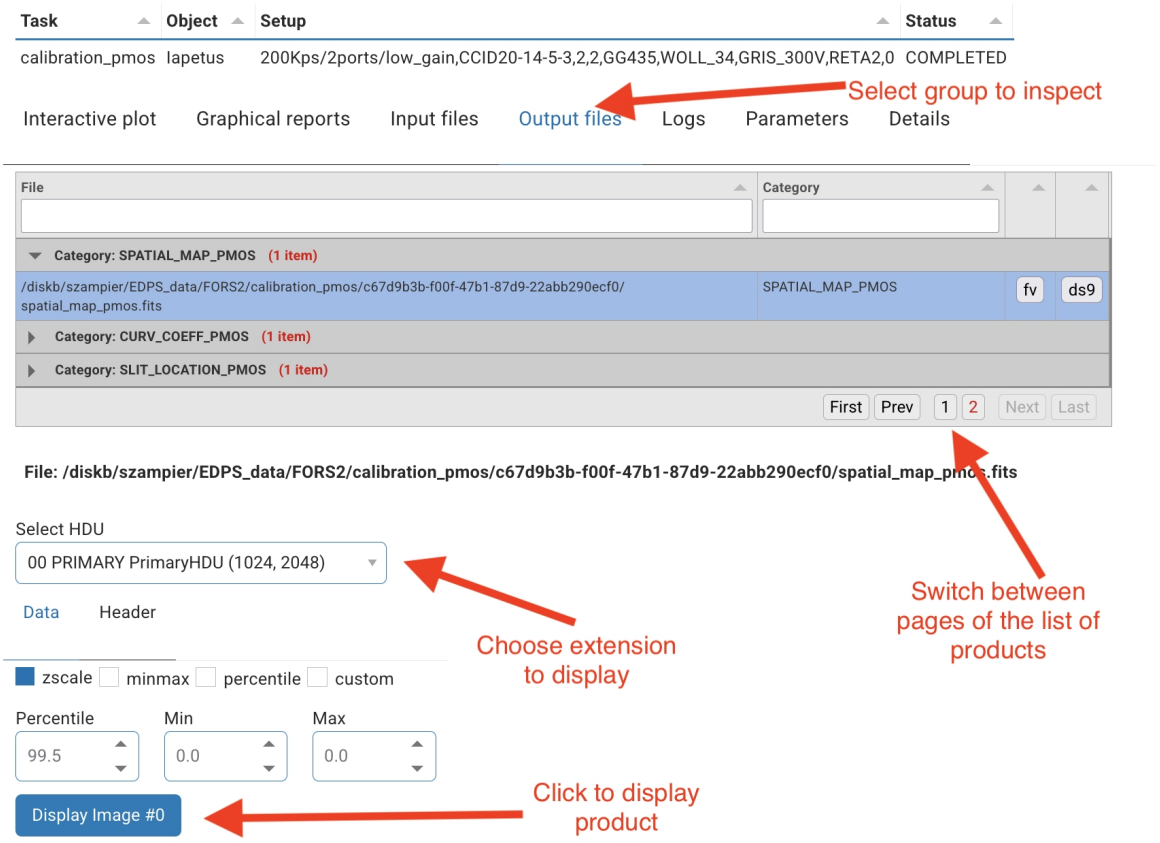

Figure 18: Examining products of the recipe fors_pmos_calib.

spurious detections (e.g. one slitlet detected as several) nor should several slitlets be detected only as one (as is the case in Figure 20, see Section 4.4 on how to fix such issue), their edges should be parallel and the areas covered by them should be identical.

3.3 Processing Standard Stars and Scientific Observations

Task: **science_spectro_polarimetry** Recipe: **for_pmos_science**

In this task the science or the standard star files are processed, applying the calibrations processed in the previous steps. The science files are typically taken at these retarder angles, depending on the desired level of reduction of systematic error:

- Circular polarimetry:
  - 2 angles: -45.0, 45.0
  - 4 angles: -45.0, 45.0, 135.0, 225.0

- Linear polarimetry:

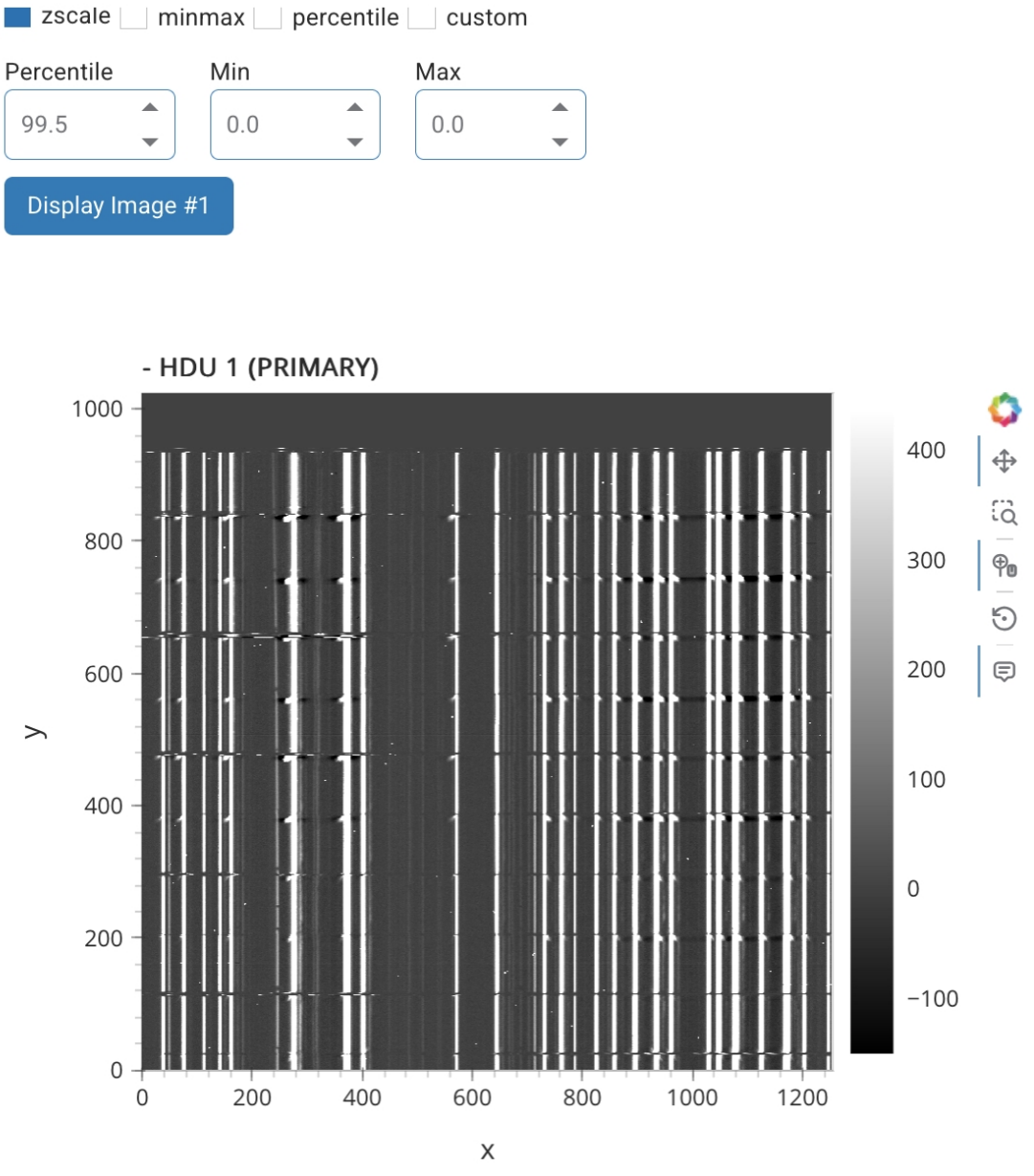

Figure 19: Example of the SPECTRA_DETECTION_PMOS product of the recipe fors_pmos_calib.

- 4 angles: 0.0, 22.5, 45.0, 67.5
- 8 angles: 0.0, 22.5, 45.0, 67.5, 90.0, 112.5, 135.0, 157.5
- 16 angles:0.0, 22.5, 45.0, 67.5, 90.0, 112.5, 135.0, 157.5, 180.0, 202.5, 225.0, 247.5, 270.0, 292.5, 315.0, 337.5

Products:

1-dimensional extracted spectra (*_REDUCED_*, created only if spectra are identified and can be extracted).

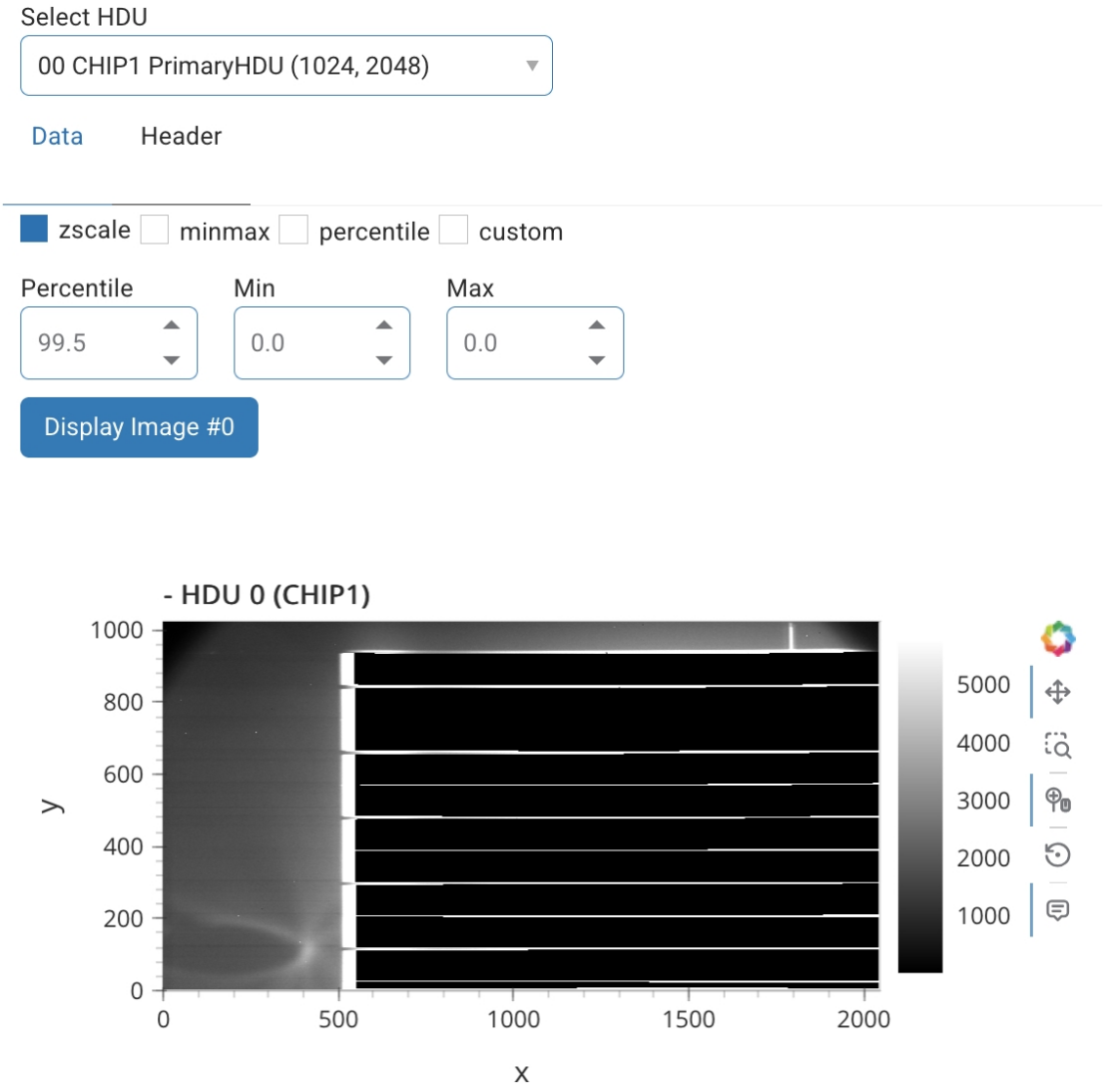

Figure 20: Example of the MASTER_NORM_FLAT_PMOS product of the recipe fors_pmos_calib.

The individual spectra are provided as rows in a FITS file with as many extensions as input scientific exposures. The correspondence between these rows and the 2-dimensional frames and/or slit identifications can be obtained from OBJECT_TABLE_SCI/STD_PMOS. All extracted spectra have the same format.

- REDUCED_SCI/STD_PMOS - spectra
- REDUCED_ERROR_SCI/STD_PMOS - 1 sigma error of spectra
- REDUCED_SKY_SCI/STD_PMOS - fitted sky spectra

2-dimensional polarization spectra (same format as 1-dimensional extracted spectra). X may be any of

ANGLE, I, L, Q, U, V, where V (circular polarisation), Q, U (linear polarisation), L (total linear polarisation, i.e. the geometrical sum of the Q and U components), ANGLE (direction of the linear polarisation vector in degrees, and finally I (flux in ADU/s). Note that V, Q, U and L are all normalized by I. REDUCED_ERROR_SCI/STD_Q_PMOS, _U_PMOS, _L_PMOS, and _ANGLE_PMOS are produced when the observation was performed using the half-wave retarder plate, while REDUCED_ERROR_SCI/STD_V_PMOS is produced when the observation was performed using the 1/4-wave retarder plate. The product REDUCED_ERROR_SCI/ is always produced. Please note that the polarization parameters U, Q and ANGLE are given with respect to the positive y-axis of the raw data.

- REDUCED_X_SCI/STD_PMOS - extracted polarisation signals from the object sources.
- REDUCED_ERROR_X_SCI/STD_PMOS - 1 sigma errors of the extracted polarisation signals
- OBJECT_TABLE_POL_SCI/STD_PMOS - table with position information for detected spectra, with the information from the extensions included within one table.

2-dimensional wavelength calibrated and distortion corrected frames (*_MAPPED_*, with as many extensions as input scientific exposures)

- MAPPED_ALL_SCI/STD_PMOS - frame without sky subtraction
- MAPPED_SCI/STD_PMOS - frame, sky-subtracted
- MAPPED_SKY_SCI/STD_PMOS - frame with fitted sky background

If sky alignment is requested (skyalign > 0) the following products are provided in addition to the ones listed above:

- DISP_COEFF_SCI/STD_PMOS - adjustment of the input DISP_COEFF_PMOS table
- SKY_SHIFTS_SLIT_SCI/STD_PMOS - table with sky line shifts
- WAVELENGTH_MAP_SCI/STD_PMOS - wavelength map adjusted for sky line shifts

In this step, the quality of the sky subtraction and the spectrum extraction should be checked.

The FORS PMOS pipeline records the 1-dimensional spectra as rows in an image. For the spectroscopic products these images have as many extensions as observed retarder plate angles (=exposures), and each exposure will create two extracted 1-dimensional spectra per object. The pipeline extracts any signal it finds, so there may be more spectra than intended targets. The extracted polarimetric parameter spectra are stored in single-extension FITS files, with one row per extracted polarimetric spectrum.

For a typical case of four exposures/retarder angles with only 1 bright target and no other objects the spectroscopic products will have four extensions with two rows each for the two spectra created by the Wollaston prism. The polarimetric products will have one extension with 1 row.

Figure 21 shows an example of the mapped sky-subtracted, wavelength-calibrated 2-dimensional spectrum frame (MAPPED_SCI_PMOS) (first exposure only) displayed with fv. Every object creates two spectra due

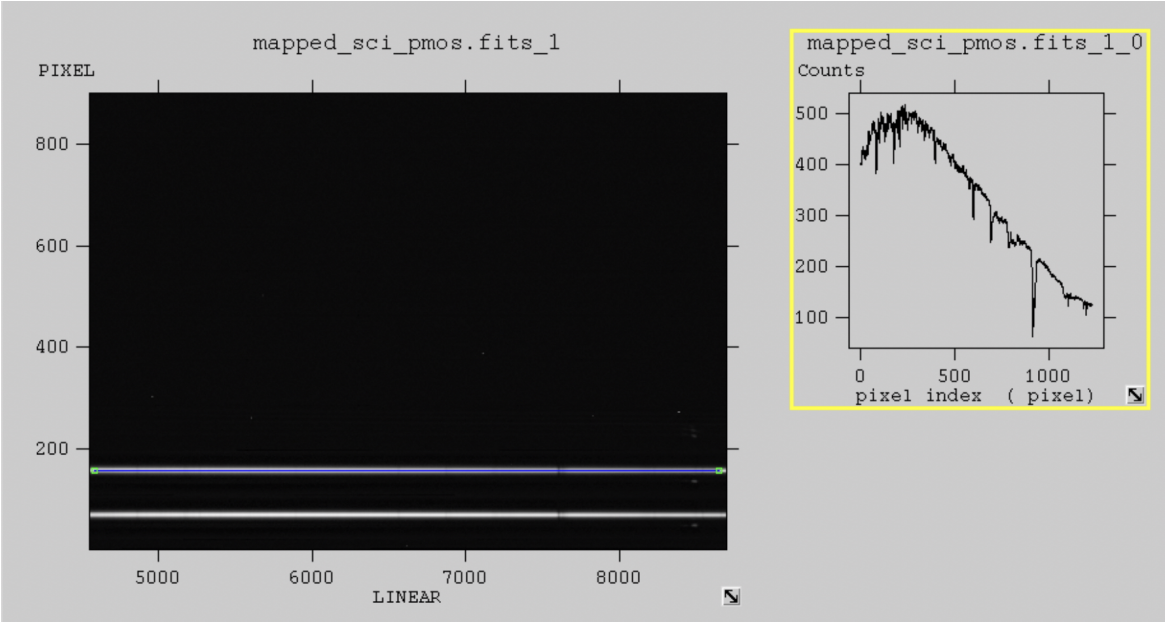

Figure 21: Example of the mapped sky-subtracted 2-dimensional science product (MAPPED_SCI_PMOS) (first exposure only) displayed in fv. The spectrum is in ADU/sec, i.e. not flux-calibrated.

to the splitting of the light by the Wollaston prism. The extracted science spectrum should not show strong residuals of sky lines. The spectra should have a generally smooth look, and will only appear to be noisier in those regions where bright sky lines were subtracted.

The number of extensions corresponds to the number of retarder angles recorded. Please note that there is no one-to-one relation between "extracted spectrum" and "object". For each object there will be two extracted spectra (one for each light beam). For instance, if a spectro-polarimetric observation consisted of four exposures at four different angles of the retarder plate, each object would have 4 angles x 2 beams = 8 spectra.

The best way to ensure that the sky was subtracted optimally, at least at the positions of the objects to extract, is to check that the residual noise is compatible with the statistical error associated to the extracted object spectra. The extracted spectra are contained in the REDUCED_SCI_PMOS image (one extracted spectrum for each row). Their error spectra (at a 1-sigma level) are contained in the REDUCED_ERROR_SCI_PMOS image. The regions of the extracted spectra corresponding to a (bright) sky line will include a few noisier points, which deviation from the spectral continuum should (almost) never pass the 3-sigma deviation.

In some cases the object can have rather low signal, because it was observed by chance in the same slit as the intended target.

The sky spectra, extracted from the modeled sky frames in exactly the same way as the object spectra from the sky-subtracted frames, can be found in REDUCED_SKY_SCI_PMOS.

Note that if the barycentric correction is applied to the products, the wavelengths are re-computed to match the desired reference system. No interpolation is done on the spectrum itself, only wavelengths are changed. This means that spectra of the same target might be defined at different wavelengths, depending on the velocity correction. This has to be taken into account when combining spectra for further analysis.

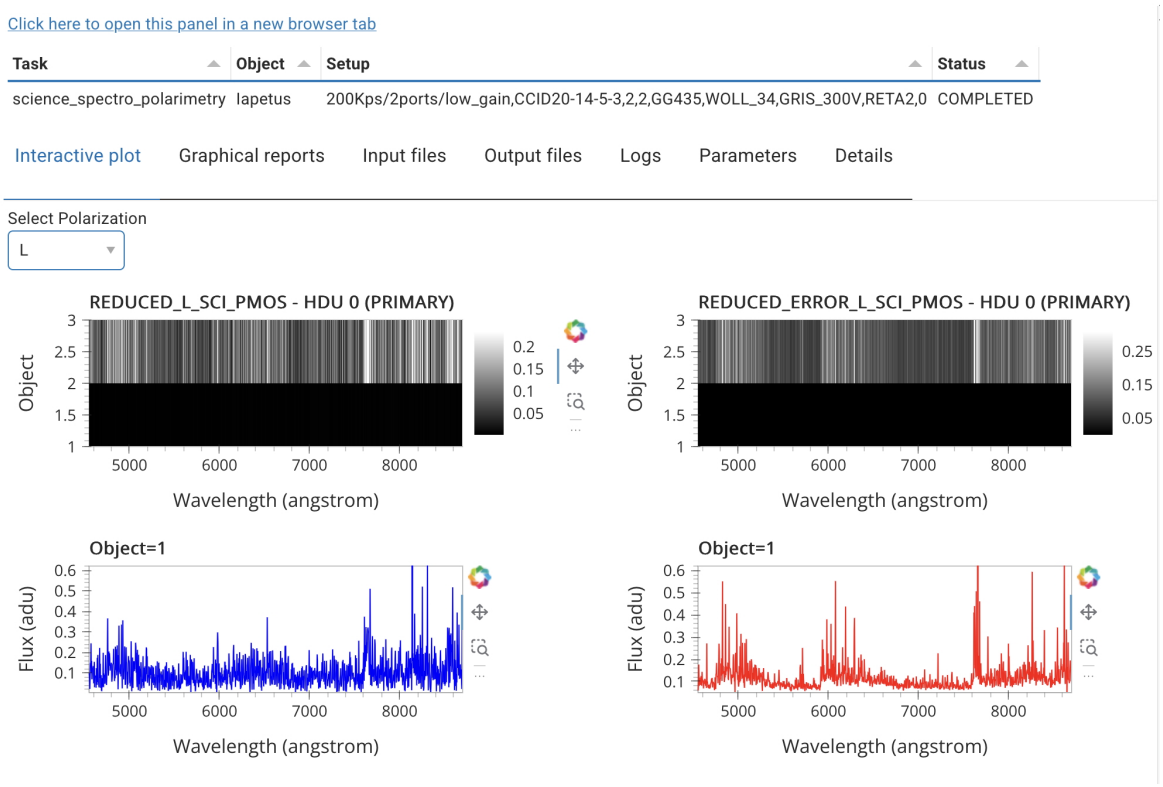

Figure 22: Example of the Interactive plot for the Task: **science_spectro_polarimetry**. It shows the Stokes L (fractional degree of polarization) parameter extracted for the science object (REDUCED_L_SCI_PMOS). The Stokes parameter Q, U, V, and ANGLE can be plotted in the same way via "Select Polarization". Further examination of the "Input files" and the "Output files" can be done by clicking on the corresponding links.

The task: **science_spectro_polarimetry** Interactive plot (Figure 22) shows the pipeline determined Stokes L (fractional degree of polarization) parameter of the science object (REDUCED_L_SCI_PMOS). The Stokes Q, U, V, and ANGLE parameter can be displayed as well via the "Select Polarization" menu. In comparison, the REDUCED_L_SCI_PMOS values (top left) should be higher than the corresponding REDUCED_ERROR_L_SCI_PMOS values (top right) for the intended science target.

From here also the "Input files", "Output files", "Logs" etc. can be examined by clicking on the corresponding links.

3.4 Static calibration data

In addition to calibration frames taken regularly, the FORS pipeline also uses static calibration tables related to spectropolarimetry.

1. Arc lamp line wavelengths (MASTER_LINECAT). This table contains the reference wavelengths (in Å) for the arc lamp used.

2. Grism table (GRISM_TABLE). This table contains grism-specific parameters, like the dispersion in Å/pixel, the start and end wavelength, the polynomial degree to be used for the wavelength calibration, etc.

3. Distortion table (MASTER_DISTORTION_TABLE). This table is necessary for the identification of the ordinary and extraordinary spectral beams.

4. Waveplate chromatism (RETARDER_WAVEPLATE_CHROMATISM). This table provides information on the color dependency of the direction of the linear polarization vector.

5. Polarimetric standard stars catalogue (STD_PMOS_TABLE). A table named fors2_pol_sta.fits, listing the measured linear polarisation from a number of standard stars, is available in the calibration directory.

4 Overview of all the data reduction configuration options

4.1 Selection of most appropriate calibrations

By default, EDPS associates raw calibrations to the reduction process. It is also possible to use pre-processed calibrations (a.k.a. master calibrations) if available, in order to speed up the reduction. The preference can be specified in the Raw Data tab, before creating the datasets.

Possible values of the Calibration Preferences are:

- **raw_per_quality_level**: At equal quality of reduction, association of raw calibrations is preferred. This is the default.
- **master_per_quality_level**: At equal quality of reduction, association of master calibrations is preferred.
- **raw**: Association of raw calibration is preferred, despite the quality of results.
- **master**: Association of master calibration is preferred, despite the quality of results.

When master calibrations are used, the reduction step needed to process raw calibrations are not executed. The reduction then moves directly to the process of scientific exposures.

For example, if reduction speed for a quick check is preferred over a high quality reduction, one can select "master". In this case, old master calibrations are associated even if there are raw calibrations closer in time (and therefore more likely to ensure better quality products).

The quality level that the selected calibrations deliver is indicated close to each dataset in the ‘Raw input‘ tab, under the colum ‘CalibLevel‘. CalibLevel=0 indicates that calibrations that follow the rules of the instrument calibration plans have been selected. The higher the number, the poorer the quality of the products.

4.2 Configuration of parameters: the configuration editor

The data reduction of each dataset can be configured according to the scientific needs using an appropriate configuration editor.

The EDPS workflows contain two types of parameters and they both have default values that can be modified to improve the data reduction.

- **Workflow parameters** (for some workflows only) are global and they are applied to the entire workflow. They are accessible both in the ‘Raw Data‘ tab, prior to the creation of a dataset, and in the ‘Reduction Configuration‘ editor, in the ‘Reduction queue‘ tab. Note: some workflow parameters were already configured before creating the dataset and sending it to the reduction queue. Here, they can be changed again. Please, note that the parameters have an effect only on the files that are already in the dataset. If one specifies a parameter that should include extra files in the dataset (e.g., the inclusion of more calibrations), files are not added and the reduction might fail. If you need to change a parameter that modifies the dataset content, please go back to the Raw data tab and create a new dataset.

- **Recipe parameters** are specific to the individual recipes and can be configured per task. They are accessible in the ‘Reduction Configuration‘ editor, in the ‘Reduction queue‘ tab.

This editor allows to configure the data reduction for a given dataset by specifying workflow and recipe parameters.

Note: some workflow parameters were already configured before creating the dataset and sending it to the reduction queue. Here, they can be changed again. Please, note that the parameters have an effect only on the files that are already in the dataset. If one specifies a parameter that should include extra files in the dataset (e.g., the inclusion of more calibrations), files are not added and the reduction might fail. If you need to change a parameter that modifies the dataset content, please go back to the Raw data tab and create a new dataset.

To open the editor, click on the wheel button  next to the dataset you desire to configure the reduction for. A window with the configuration editor appears as shown Figure 23.

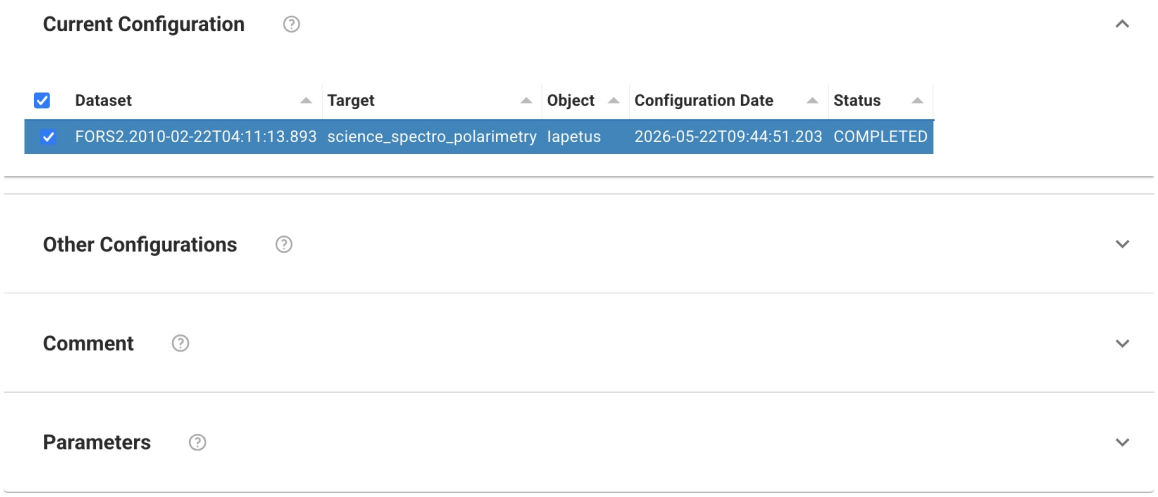

Figure 23: The Reduction Configuration editor. It contains 4 sections, that indicate the current configuration, list of other configurations to set, comments to insert, and the parameters to modify.

The editor is divided into 4 parts, which can be accessed pressing the corresponding expansion arrow.

- **Current configuration.** It indicates the name of the selected configuration for a given dataset (Figure 24).
- **Other configurations.** It allows to specify other configurations, to which the changes shall be copied to (Figure 25).
- **Comment.** It allows to specify a comment to describe the configuration. It is possible to append or replace a comment (Figure 26). Comments can be changed on all configurations. It is possible to save the comment for the current configuration only, or for all the selected configurations.
- **Parameters.** A window as in Figure 27 appears.

  The window allows to:

- Select the parameter set. A pre-determined list of workflow parameters and recipe parameters for a given use case. For the majority of the cases, the "science" parameter set can be used.
- Edit the workflow parameters. These are parameters that regulates the reduction strategy, e.g. whether to use a given calibration or not, or to trigger a certain reduction step. Note that if the changes imply that some files not in the dataset are needed, the reduction might fail. In case, go back to the raw data tab, edit the workflow parameters there, and recreate the datasets.
- Edit the recipe parameters. These are parameters associated to the recipe of a given task. Note: the same recipe parameters can be configured differently for the tasks that run the same recipe. Default parameters are shown (albeit some parameters can be dynamic, e.g. 'EDPS' changes their value depending on the type of input data).

Change the values according to the needs and then select whether to save it to the current or the selected configurations. Note, complete configurations cannot be modified, new configurations will be automatically created instead.

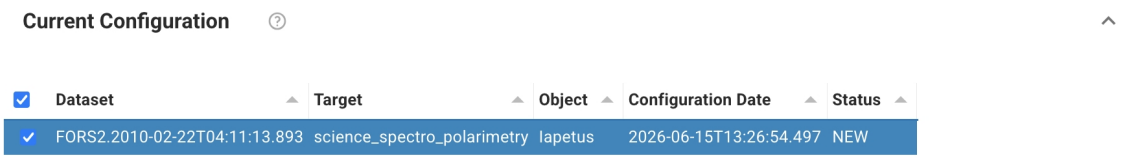

Figure 24: The first part of the Reduction Configuration Editor, that indicates the selected configuration.

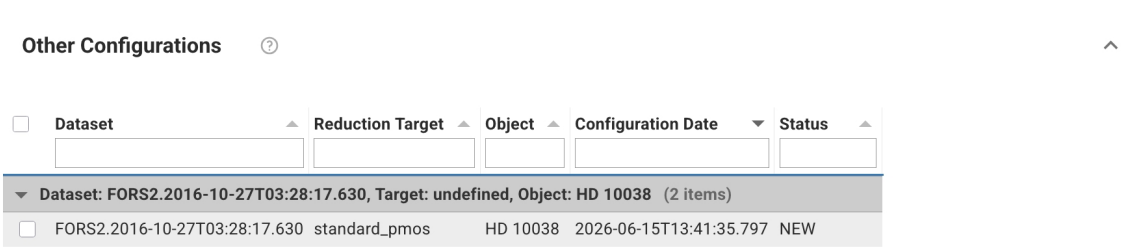

Figure 25: The second part of the Reduction Configuration Editor, that indicates other configurations for which we'd like to apply the changes.

4.2.1 Apply the same parameters to multiple datasets and configurations

A common user case is to apply the same set of optimized parameters to other datasets or other configurations. This is possible with the **Other configurations** tab of the configuration editor. To do so, proceed as follows:

- From the Reduction Queue tab, select the dataset/configuration that has the parameters you want to adopt for other reductions.
- Open the configuration editor by clicking on the weel-button on the right-hand side, at the end of the line of the selected configuration.

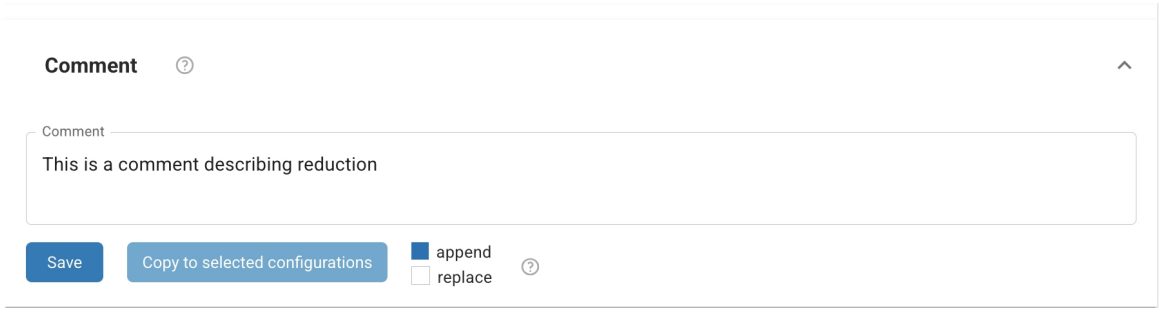

Figure 26: The third part of the Reduction Configuration Editor, that allows to specify a comment to the selected configurations.

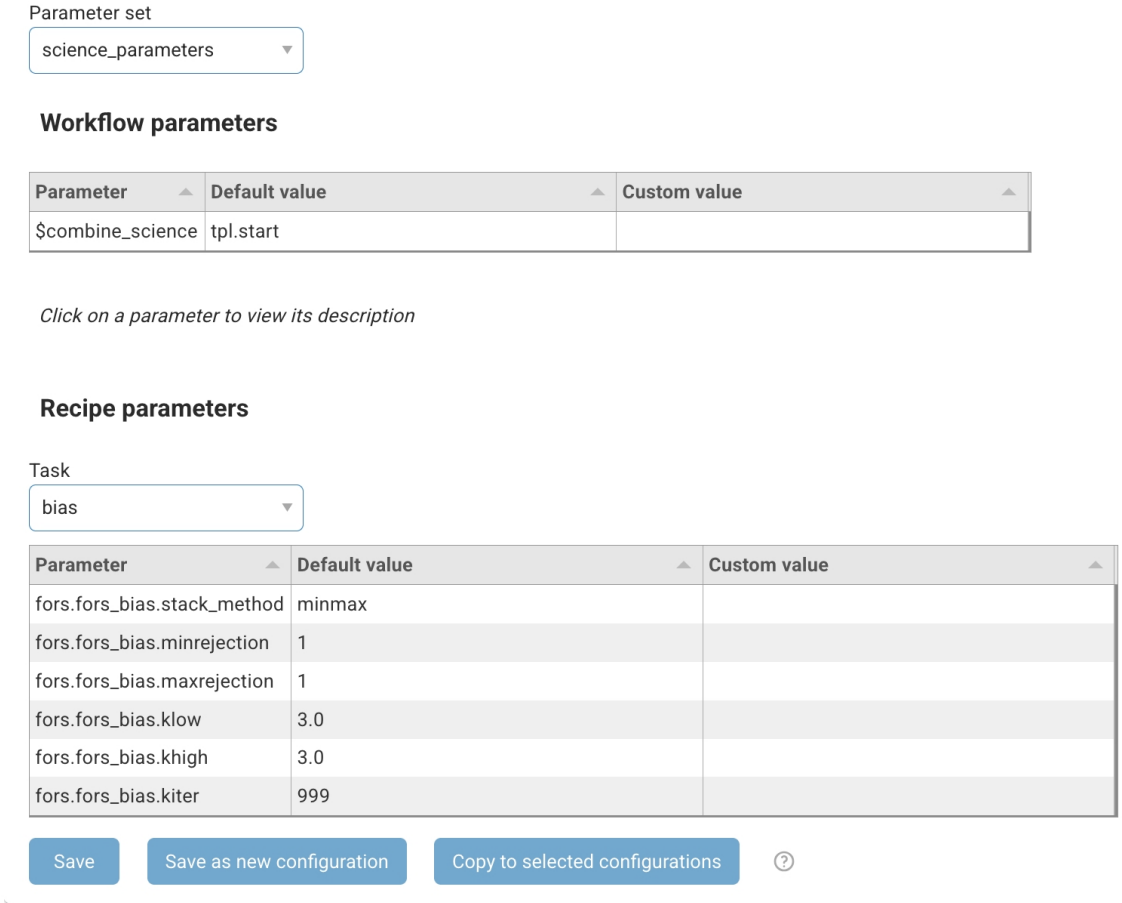

Figure 27: The fourth part of the Reduction Configuration Editor, that allows to specify the parameters sets and the recipe parameter per task. These settings can be applied to the "Selected Configuration" (Fig. 24) or to the "Other Configurations (Fig. 25).

- If necessary, edit the parameters.
- On the **Other configurations** tabs, select the datasets or configurations you want to reduce with the same parameters of the "Current configuration". Only datasets/configurations present in the Reduction Queue can be selected.
- Go on the Parameters tab of the configuration editor and click on the button **Copy to selected configurations**. Selected configurations that were labelled with "NEW" will be updated, selected configurations that were labelled with "COMPLETED/FAILED/ABORTED" will not be updated, but "NEW" configurations will be created instead.
- If you have edited paramters on the 3rd step, please note that the original "Current configuration" will not be updated by pressing the button **Copy to selected configurations**; only those selected in the "Other configurations" tab will be updated. If you want the edited parameters to be copied also to the "Current configuration", then press the "SAVE" button (available only if the Current configuration is labelled as NEW) or "Create new configuration" button (in this case the Current configuration is not updated, but a "NEW" configuration is created instead).

The newly created or updated configurations are now ready for reduction.

4.3 Producing optimal Master Bias

Figure 17 shows an example of the quality plot for the MASTER_BIAS Output file created with the recipe parameters as in Figure 27. The image should be smooth, without visible flux gradient. If it is not, check the input raw files. Exclude the visible "bad" ones if there are any.

4.4 Configuring fors_pmos_calib and improving results

The creation of the FORS PMOS master flat, wavelength solution and other calibration products (task: **calibration_pmos**, recipe: **fors_pmos_calib**) can be configured with the parameters as in Figure 28.

The recipe produces many products in different formats, with or without extensions. They can be examined as explained in Figure 18.

The ways to improve the results depend on the instrumental setup of the data. Figure 19 shows the wavelength-calibrated arc lamp frame for the grism 300V setup (product SPECTRA_DETECTION_PMOS). In this plot the arc lamp lines should run straight from top to bottom without any empty rows between them. Particular attention should be given to lines at the blue and red ends of each spectrum, where the polynomial fit is more sensitive to small variations of the signal. Some arc lines may show gaps due to the placement of the slits, but empty rows without any lines point towards problems with the detection of the arc lamp lines. Varying the parameter **peakdetection** (increasing to e.g. 450 to detect only the strongest arc lines, or decreasing to e.g. 10 to see all, even the weakest lines) usually solves this problem.

The Figure 20 shows the example of the grism 300V MASTER_NORM_FLAT_PMOS product where the slits were not properly traced (at y around 700). Increasing the tolerance for outliers **wreject** to 2 (for some setups also try 3) recovers all slitlets, but may leave some instabilities in the wavelength calibration. This can be further fixed by increasing the search window for lines **wradius** to 12 (for some setups also try 4, which is a default).

Task

calibration_pmos

Parameter	Default value	Custom value
fors.fors_pmos_calib.dispersion	0.0	
fors.fors_pmos_calib.peakdetection	0.0	
fors.fors_pmos_calib.wdegree	0	
fors.fors_pmos_calib.wradius	4	
fors.fors_pmos_calib.wreject	1	
fors.fors_pmos_calib.wcolumn	WLEN	
fors.fors_pmos_calib.cdegree	0	
fors.fors_pmos_calib.cmode	1	
fors.fors_pmos_calib.startwavelength	0.0	
fors.fors_pmos_calib.endwavelength	0.0	
fors.fors_pmos_calib.stack_method	average	
fors.fors_pmos_calib.minrejection	1	
fors.fors_pmos_calib.maxrejection	1	
fors.fors_pmos_calib.klow	3.0	
fors.fors_pmos_calib.khigh	3.0	
fors.fors_pmos_calib.kiter	999	
fors.fors_pmos_calib.ddegree	-1	
fors.fors_pmos_calib.dradius	10	
fors.fors_pmos_calib.qc	True	
fors.fors_pmos_calib.check	TRUE	
fors.fors_pmos_calib.nonlinear_level	60000.0	

Figure 28: The fourth part of the Reduction Configuration Editor, that allows to specify the parameters sets and the recipe parameters for task **calibration_pmos**. These settings can be applied to the "Selected Configuration" (Fig. 24) or to the "Other Configurations (Fig. 25).

5 List of workflow tasks

This is the list of all the tasks and associated recipes in the FORS SPECTROPOLARIMETRY workflow. Only some of them are needed for scientific reduction, they are indicated by the flag "yes" (triggered by default) or "optional" (triggered only if requested by a workflow parameter). Other tasks are not used for scientific reduction (they are indicated by the flag "no"), they are mainly used for instrument monitoring and they can be executed only by specifying them as target. Note that, when a task is specified as target, all the tasks that generate the calibrations needed for it are automatically executed.

TASK	RECIPE	Used in science reduction	Notes
bias	fors_bias	yes	Computes the master bias frame.
calibration_pmos	fors_pmos_calib	yes	Creates all necessary calibration products to process spectropolarimetric data.
science_spectro_polarimetry	fors_pmos_science	yes	Reduces spectropolarimetric science data.
standard_pmos	fors_pmos_science	yes	Reduces spectropolarimetric standard star data.

Table 1: FORS SPECTROPOLARIMETRY pipeline tasks overview

6 Frequently Asked Questions

- **Q1) Where can I find the final reduced data?**

  Answer: all the products of all the datasets and the reductions are saved into the EDPS_data directory, specified when executing the `edps-gui` for the first time. One can decide to export only the final products for selected datasets and only for the desired reduction attempts into another location for further analysis. See Section 2.4 for further instructions.

- **Q2) How do I stop the application?**

  Answer: Proceed as follows:

  1. Press "Stop EDPS" in the Dashboard.
  2. Type Ctrl-C in the terminal where the application is running. If the application doesn't terminate, type Ctrl-C again.
  3. Alternatively, kill the 'panel serve' process on your system, for example:

     ```
     ps -e | grep panel # get the process ID of the gui (<pid>).
     kill -9 <pid>
     ```

- **Q3) I have closed the browser window where the application is running. How can I reopen the application?**

  Answer: Point your browser to: `http://localhost:5006/edps-gui`

- **Q4) Where can I find some data that I can use to test the application?**

  Answer: Install the 'datademo' package provided with the pipeline installation or download the "Demo Data" package from https://www.eso.org/sci/software/pipe_aem_table.html.

  Please note that the demo data can be large (tens of Gigabytes).

  A convenient script to download demo data for any pipeline is also available and can be used from the command line:

  ```
  curl -O https://eso.org/sci/software/apptainer/eso_download_demodata.sh
  bash ./eso_download_demodata.sh
  ```

- **Q5) How can I start the edps-gui if the following message appears?**

  ```
  Cannot start Bokeh server, port 5006 is already in use
  ```

  Answer: The panel server was not closed properly. Kill it by typing:

  ```
  ps -e | grep panel # get the process ID of the gui (<pid>).
  kill -9 <pid>
  ```

- **Q6) How do I get additional support on EDPS or data reduction in general?**

  Answer: For suggestions, questions, or feedback in general, please open a ticket with the EDPS Support team. This https://support.eso.org/new-ticket?ticket%5Bticket_field_13%5D%5Bdata%5D=227 should take you directly to a webpage for creating and `EDPS` feedback ticket, but in case you want to navigate there 'manually', go to https://support.eso.org, login, click on "Submit Helpdesk Ticket", and specify the Help topic: "Post Observations", "ESO Data Processing System [EDPS]".

- **Q7) I have a lot of disk space, but when I install `EDPS` with `pip` or an ESO pipeline with `Homebrew` I get the error message: `Cannot mkdir: No space left on device`. How do I fix it?**

  Answer: This depends on how much disk space is allocated to the `/home`, `/var`, and `/tmp` directories. The final solution would be to resize the space allocated to the in the organization of the filesystem. However, we list here few tricks that might do the job.

  – Clearing the pip .cache to make space for new packages. Type the command:

    ```
    pip cache purge
    ```

    before installing `EDPS`.

  – Redirect the cache, `Homebrew` temporary build directories into a partition with enough space. Set some of the following environmental variables in your `.bashrc` file:

    ```
    export HOMEBREW_CACHE=<path_to_new_cache_directory>
    export XDG_CACHE_HOME=<path_to_new_cache_directory>
    export HOMEBREW_TEMP=<path_to_new_temporary_directory>
    export TMPDIR=<path_to_new_temporary_directory>
    ```

    The first moves only the location of `Homebrew` cache, the second the cache of most applications (instead of the default `/home/username/.cache`), the third moves the directory where `Homebrew` builds, extracts, and saves temporary files (instead of the defaults `/tmp` and `/var/tmp`). The last changes the global system temporary directory and affects most of the linux commands.

  – As extreme measure, one can move the `/home/linuxbrew/.linuxbrew` directory somewhere else, and create a symbolic link in `/home/linuxbrew`. For example:

    ```
    cd /home/linuxbrew
    mv -f .linuxbrew <path_to_new_directory>
    ln -s <path_to_new_directory> .linuxbrew
    ```

    Important note: this operation might break some internal links. Recipes requiring external packages such as `telluriccorr` might not work (impacts on KMOS, XSHOOTER, FORS2, and MOLECFIT pipelines).