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

ESPRESSO Pipeline User Manual

ESO-331895

Issue 3.2.0

Date 2024-05-15

Prepared:	A. Modigliani, D. Sosnowska, C. Lovis	2024-05-15	
	Name	Date	Signature
Approved:	P.Ballester		
	Name	Date	Signature
Released:	M. Peron		
	Name	Date	Signature

Change record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
1.0	10/12/2018	all	First public release,pipeline version 1.1.0
1.0.2	25/03/2019	all	release update
1.2.3	24/05/2019	3	Minor documentation updates
1.3.2	31/07/2019	all	Updated, restructured, simplified, clarified
2.0.0	18/11/2020	all	Updated, restructured, simplified, clarified
2.2.1	28/05/2021	all	Updated
2.3.0	03/03/2021	-	Description van-Dokkum cosmic rejection
2.3.3	01/06/2021	-	`CRH_MAP` is optional input of science recipe Updated order tracing recipe to use the inter-order background measurement for orders tracing.
2.4.0	11/07/2022	-	CoP 2022 release. Fixed bugs on `espdr_wave_THAR_THAR` and `espdr_wave_LFC_LFC`.
		-	Improved extracted spectrum noise determination, fixing a bug in the master dark contribute.
3.0.0	15/05/2023	-	Support HARPS and NIRPS, FP cavity lengh chromatic drift correction Updated CCF masks, corrected dark current computation
3.1.0	28/06/2023	-	EDPS support
3.2.0	15/05/2024	-	Coordinated release update

This page was intentionally left blank

Contents

1 Introduction

1.1 Purpose

The ESPRESSO pipeline is a subsystem of the *VLT Data Flow System* (DFS). Its target user is both the user community and ESO *Data Products Department* (DPD) in the generation of master calibration data, in the reduction of scientific exposures, and in the data quality control. It should also serve as a quick look tool for *Paranal Science Operations* (PSO). Additionally, the ESPRESSO pipeline recipes are made public to the user community, to allow a more personalised processing of the data from the instrument. The ESPRESSO pipeline was developed with the declared objective of providing precise Radial Velocity (RV) data and accurate wavelength calibration. The purpose of this document is to describe a typical ESPRESSO data reduction sequence with the ESPRESSO pipeline.

This manual is a complete description of the data reduction recipes implemented by the the ESPRESSO pipeline, reflecting the status of the ESPRESSO pipeline as of May 15, 2024 (version 3.2.0).

1.2 Acknowledgments

The ESPRESSO pipeline has been designed, implemented and developed by the Geneva Observatory. We are particularly grateful to the responsibles for the data reduction: Christophe Lovis, Danuta Sownowska and Alex Segovia for their contributions and support.

Andrea Modigliani, from ESO, provided the Reflex workflow, DFS support and most of the documentation. We thank also Pedro Figueira, Andrea Mehner and Richard Anderson for providing useful feedback to improve the reflex workflow and documentation.

1.3 Scope

This document describes the ESPRESSO pipeline used at ESO-Garching and ESO-Paranal for the purpose of data assessment and data quality control.

Updated versions of the present document may be found on [1]. For general information about the current instrument pipelines status we remind the user of [2]. Quality control information are at [3].

Additional information on the Common Pipeline Library (CPL) and ESOREX can be found respectively at [4], [5]. A description of the instrument is in [6]. The ESPRESSO instrument user manual is in [7]. The ESPRESSO Reflex tutorial is in [8].

1.4 Reference documents

[1] ESPRESSO Pipeline Users' Manual VLT-MAN-ESO-19500-XXXX
www.eso.org/pipelines
[2] Current pipeline status
www.eso.org/observing/dfo/quality/pipeline-status.html
[3] ESO-Data Flow Operation home page http://www.eso.org/observing/dfo/quality/
[4] CPL home page www.eso.org/cpl
[5] ESOREX home page www.eso.org/cpl/esorex.html
[6] ESPRESSO home page
www.eso.org/sci/facilities/paranal/instruments/espresso
[7] VLT ESPRESSO User Manual VLT-MAN-ESO-14700-3517
www.eso.org/sci/facilities/paranal/instruments/espresso/doc
[8] Reflex ESPRESSO Tutorial www.eso.org/pipelines

2 Overview

In collaboration with instrument consortia, the Pipeline Systems Department (PSD) of the Software Development Division is implementing data reduction pipelines for the most commonly used VLT/VLTI instrument modes. These data reduction pipelines have three main purposes:

Data quality control: pipelines are used to produce the quantitative information necessary to monitor instrument performance.

Master calibration product creation: pipelines are used to produce master calibration products (*e.g.*, combined bias frames, super-flats, wavelength dispersion solutions).

Science product creation: using pipeline-generated master calibration products, science products are produced for the supported instrument modes (*e.g.*, combined ISAAC jitter stacks; bias-corrected, flat-fielded FORS images, wavelength-calibrated UVES spectra). The accuracy of the science products is limited by the quality of the available master calibration products and by the algorithmic implementation of the pipelines themselves. In particular, adopted automatic reduction strategies may not be suitable or optimal for all scientific goals.

Instrument pipelines consist of a set of data processing modules that can be called from the command line, from the automatic data management tools available on Paranal or from Gasgano.

ESO offers two front-end applications for launching pipeline recipes, *Gasgano* and *EsoRex* [5] both included in the pipeline distribution (see Appendix A, page 74). These applications can also be downloaded separately from `www.eso.org/gasgano` and `www.eso.org/cpl/esorex.html`. GASGANO is a Data File Organiser developed and maintained by ESO to help its user community to manage and organise in a systematic way the astronomical data observed and produced by all VLT compliant telescopes, i.e. by telescopes which are being operated through Observation Blocks. The tool also supports FITS files which are not generated by those telescopes with limited functionality. For further information, please, refer to the latest `Gasgano User's Manual`. In the special case of the ESPRESSO pipeline, which uses a lot of RAM, we decided not to provide Gasgano support.

Recent pipeline improvements are listed in Section 3. The ESPRESSO instrument is described in Section 4. In section 5 we list known data reduction problems for the ESPRESSO pipeline and possible solutions. The different types of ESPRESSO raw frames and auxilliary data are described in Sections 6, and 7. An overview of the data reduction, the input data, and the recipes involved in the calibration cascade is provided in section 8. More details on inputs, products, quality control measured quantities, and controlling parameters of each recipe is given in section 9. A brief introduction to the usage of the available reduction recipes using EsoRex is presented in Section 10. More detailed descriptions of the data reduction algorithms used by the individual pipeline recipes can be found in Section 11.

In Appendix A the installation of the ESPRESSO pipeline recipes is described.

3 Recent improvements

This release of the ESPRESSO pipeline (version 3.2.0) includes the following major improvements with respect to public release: 2.4.0

- Support for HARPS and NIRPS.
- FP cavity length calibration taking into account chromatic drifts with time (instead of a single achromatic offset).
- New CCF masks with weights taking into account the true Doppler information content of individual lines (i.e. also considering FWHM instead of contrast only).
- Proper calculation and propagation of dark current noise.
- Mitigation of bugs in the exposure meter data tables affecting the flux-weighted mid-exposure time calculation.
- Support for EDPS workflows.

4 ESPRESSO Instrument Description

ESPRESSO is the Echelle SPectrograph for Rocky Exoplanets and Stable Spectroscopic Observations installed at the incoherent combined Coudé facility of the VLT. It is an ultra-stable fibre-fed échelle high-resolution spectrograph which collects the light from either a single UT or the four UTs simultaneously via the so-called UT Coudé trains. The whole system is built to reach the ultimate radial-velocity precision of 10 cm/s over a timespan of 10 years in single UT mode.

In this chapter a brief description of the ESPRESSO instrument is given. A more complete documentation can be found in the ESPRESSO User Manual, downloadable from
www.eso.org/sci/facilities/paranal/instruments/espresso.

4.1 Instrument overview

ESPRESSO is the ESO/VLT high-resolution spectrograph designed to measure precise radial velocities on a long timespan with the main scientific aim of detecting and characterising Earth mass exoplanets in the habitable zone of solar-like stars.

ESPRESSO is a highly-stabilized fibre-fed échelle spectrograph that can be fed with light from either a single or the four Unit Telescopes (UTs) simultaneously. The instrument is installed at the incoherent combined Coudé focus (ICCF) of the VLT. The light from an astronomical source is redirected from the telescopes to the detectors through three components of the ICCF facility: the UT Coudé trains, the front end units, and the spectrograph itself. The Coudé Trains (CT) bring the light from each telescope to the Combined Coudé Lab (CCL) through 11 optical elements, including mirrors, lenses, and prisms. The four Front Ends (one for each UT) receive the light from the CTs and feed the spectrograph entrance fibres. The Fibre Link transports the light from the Front Ends to the vacuum vessel. The latter is thermally stabilized at the mK level and stabilized in pressure down to the μbar. The light is then going through the optical components of the spectrograph and split up into a red and a blue spectrum which are recorded on the corresponding science detectors.

The spectrograph is fed by two fibres, one for the target and the other one for simultaneous calibration (either the sky or a simultaneous reference: Fabry-Pérot, Laser Frequency Comb, or Thorium-Argon lamp). The light from the two fibres is recorded onto a blue (380-525nm) and a red (525-788nm) CCD mosaic. ESPRESSO can operate in three main instrument configurations: High Resolution 1-UT (HR), Ultra High-Resolution 1-UT (UHR) and Medium Resolution 4-UT (MR). The main characteristics of these modes are summarised in Table 4.1.0.

4.1.1 Instrument description

ESPRESSO is the first instrument of a VLT facility that can host instrumentation at the so-called Incoherent Combined Coudé Focus (ICCF). As such, in order to describe the instrument, one needs to understand the three components through which the light travels from the telescope to the detectors. These are: the UT Coudé Trains, the Front Ends, and the Spectrograph itself. Each of these components are described below. For a more detailed description, we point the interested reader to the ESPRESSO user manual.

Table 4.1.0: EXPRESSO specifications

	HR(1-UT)	UHR(1-UT)	MR(4-UT)
Wavelength range	380-788 nm	380-788 nm	380-788 nm
Resolving power (median)	140,000	190,000	70,000
Aperture on sky	$1''.0$	$0''.5$	$4x1''.0$
Total peak efficiency	11%	6%	11%
RV precision (requirement)	$<$ 10 cm/s	$<$ 5 m/s	$<$ 5 m/s
Limiting V-band magnitude	$\approx$ 17	$\approx$ 16	$\approx$20
Binning	1x1, 2x1	1x1	4x2, 8x4
Spectral sampling (average)	4.5 px	2.5 px	5.5 px (binned x2)
Spectral sampling per slice	9.0(4.5) px	5.0 px	5.5 px (binned x4)
Number of slices	2	2	1 (merged)
Detector read-out mode	FAST (1x1), SLOW (2x1)	FAST	SLOW
CCD pixel binning	1x1 and 2x1	1x1	4x2 and 8x4
RON (blue detector)	8e-/pix	8e-/pix	3e-/pix
RON (red detector)	5e-/pix	5e-/pix	2e-/pix
CONAD	1.1e-/ADU	1.1e-/ADU	1.1e-/ADU

4.1.2 The Coudé Trains

Distances between each UT and the Combined Coudé Laboratory (CCL) range between 48 m for UT2 and 69 m for UT1. A trade-off analysis between solutions based on mirrors, prisms, lenses, and fibres, points towards a full-optics solution. The chosen design with the position of the 11 optical elements is shown below in the second figure. The Coudé train picks up the light through a prism at the level of the Nasmyth-B platform and routes the beam through the UT mechanical structure down to the UT Coudé room, and farther to the CCL along the existing incoherent light ducts. The four trains relay a field of 17 arcsec around the acquired object to the CCL. The selected concept to convey the light of the telescope from the Nasmyth-B focus to the entrance of the tunnel in the Coudé room (CR) below each UT is based on a set of 6 prisms (with some power). The light is directed from the UT's Coudé room towards the CCL using 2 large lenses. The beams from the four UTs meet in the CCL, where mode selection and beam conditioning is performed by the fore-optics of the Front-End sub-system.

4.1.3 The Front Ends

The Front-End transports the beam received from the Coudé, once corrected for atmospheric dispersion by the ADC, to the common focal plane where the pickups for the spectrograph fiber feeds are located. While performing such a beam conditioning, the Front-End can apply pupil and field stabilizations. These are achieved via two independent control loops each composed of a technical camera and a tip-tilt stage. Another dedicated stage delivers a focusing function. In addition, the Front-End handles the injection of the calibration light, prepared in the Calibration Unit, into the fibers and then into the spectrograph. As wavelength calibration sources, ESPRESSO is equiped with a laser frequency comb (LFC), two ThAr lamps and a Fabry Perot Etalon system for simultaneous measurement of the radial velocity (RV) drift. A toggling mechanism handles the

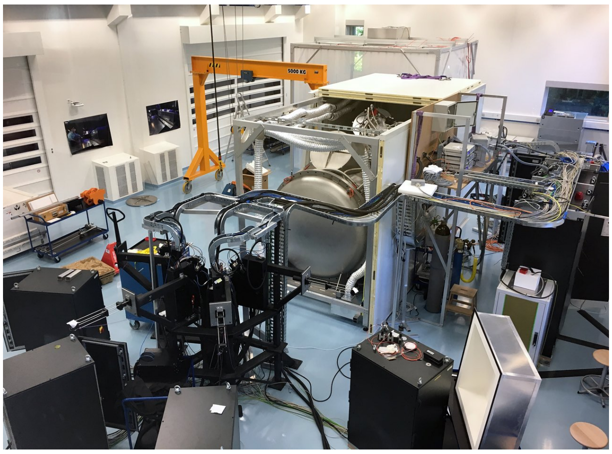

Figure 4.1.1: An view of ESPRESSO

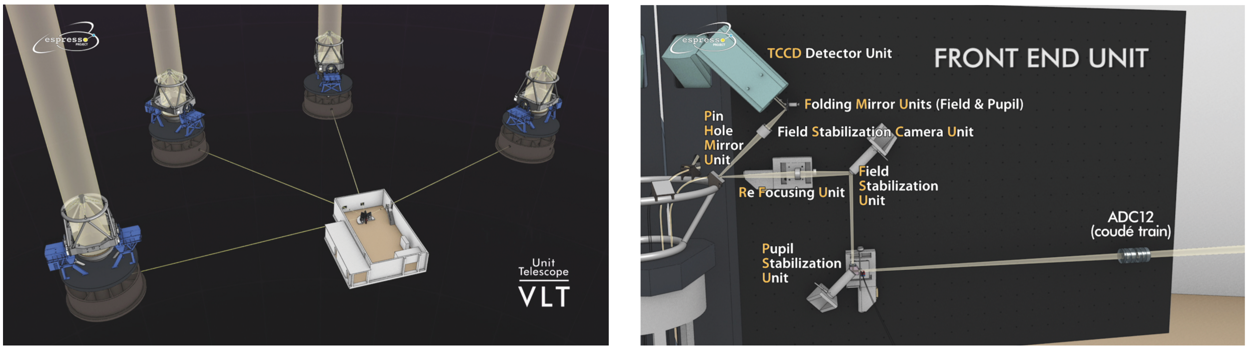

Figure 4.1.2: An view of Coudé Trains (left) and of the front end unit (right).

selection between the possible observing modes in a passive way.

Link sub-system relays the light from the Front-End to the spectrograph and forms the spectrograph pseudo-slit inside the vacuum vessel. The 1-UT mode uses two octagonal fibres each, one for the object and one for the sky or simultaneous reference. In the high-resolution (singleHR) mode, the fibre has a core of 140 μm, equivalent to 1 arcsec on the sky; in the ultra-high resolution (singleUHR) mode, the fibre core is 70 μm and the covered field of view is 0.5 arcsec. The fibre entrances are organized in pickup heads that are moved to the focal plane of the Front End when the specific bundle of the specific mode is selected. In the 4-UT mode (multiMR), four object fibres and four sky/reference fibres are fed simultaneously by the four telescopes. The four object fibres will finally feed a single square 280 μm object fibre, while the four sky/reference fibres will feed a single square 280 μm sky/reference fibre. In the 4-UT mode, the spectrograph will see a pseudo: the four individual fibres are bundled together and fed into a square fiber, that is then imaged by the instrument leading to a single square image on the detector. Another essential task performed by the Fibre-link sub-system is the light scrambling. The use of a double-scrambling optical system will ensure both scrambling of the near field and far field of the light beam. A high scrambling gain, which is crucial to obtain the required RV precision in the 1-UT mode is achieved by the use of octagonal fibres (Chazelas et al. 2011).

4.1.4 The Spectrograph

The spectrograph optics are mounted in a 3-dimensional optical bench specifically designed to keep the optical system within the thermo-mechanical tolerances required for high-precision radial velocity measurements. The bench is mounted in a vacuum vessel in which 10-3 mbar class vacuum is maintained during the entire duty cycle of the instrument. The temperature at the level of the optical system is required to be stable at the mK level in order to avoid both short-term drift and long-term mechanical instabilities. Such an ambitious requirement is obtained by locating the spectrograph in a multi-shell active thermal enclosure system. Each shell improves the temperature stability by a factor of 10, thus getting from typically Kelvin-level variations in the CCL down to 1 mK stability inside the vacuum vessel and on the optical bench.

At the entrance of the spectrograph, an anamorphic pupil slicing unit (APSU) shapes the beam in order to compress the beam in cross-dispersion direction but not in main-dispersion direction, where high resolving power needs to be achieved. In the latter direction, however, the pupil is sliced and superimposed on the echelle grating to minimize its size. The rectangular white-pupil is then re-imaged and compressed by the anamorphic VPH grism. Given the wide spectral range and the required efficiency, two large 90x90 mm CCD detectors are required to record the full spectrum. Therefore, a dichroic beam splitter separates the beam in a blue and a red channel which in turn allows to optimize each spectroscopic arm for image quality and optical efficiency. Each of the two blue and red cross-dispersers has the function of separating the dispersed spectrum in all its spectral orders. In addition, an anamorphism is re-introduced to make the pupil square and to compress the order width in cross-dispersion direction, such that the inter-order space is maximized.

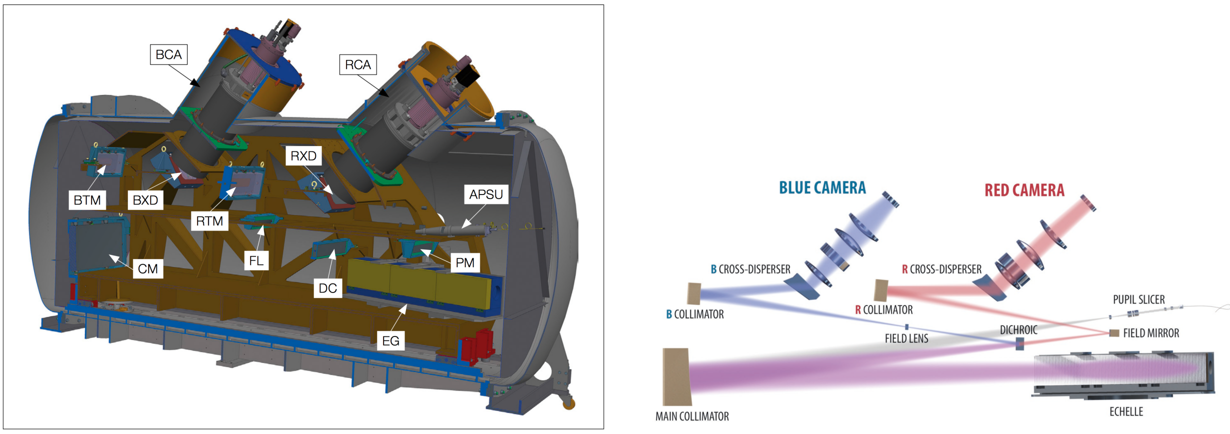

Figure 4.1.3: ESPRESSO: opto-mechanical path (left) and optical path (right).

4.2 Detector layout

ESPRESSO data include two images, one for each camera, that are stored in two extensions of a FITS files. The detector geometry is described in the following image.

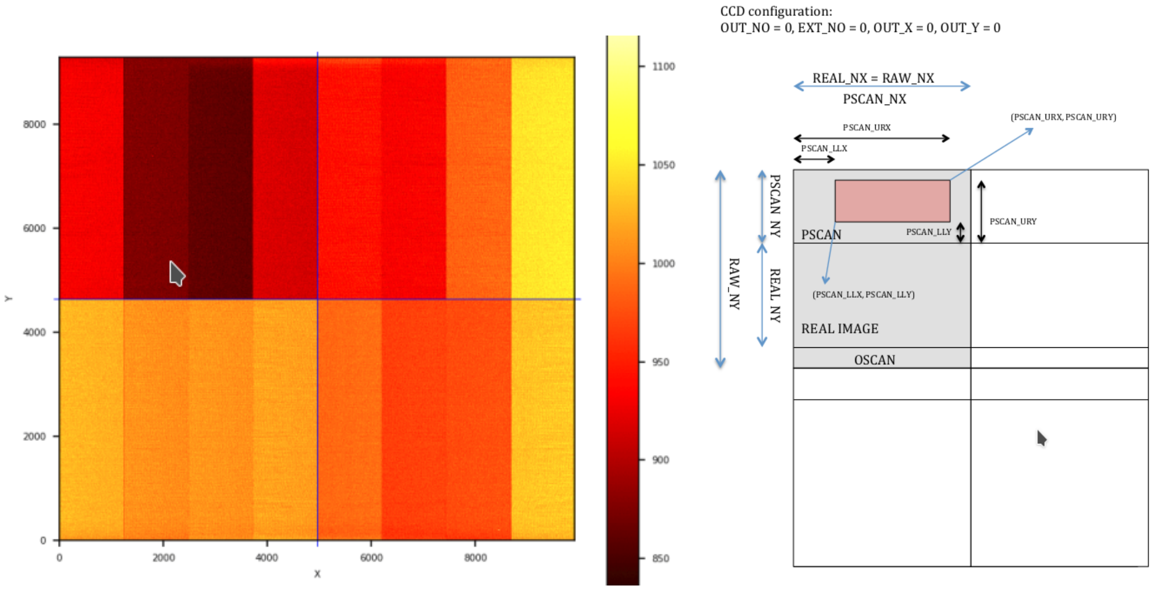

Figure 4.2.1: ESPRESSO master bias (left) and detailed geometry of the read-out regions (right). As shown in the image each detector has 16 read-out regions each with a different bias level.

5 Known problems

Several pipeline problems have been identified:

- **Slow execution time.**The data reduction chain in SINGLE mode may take around 45min on a regular computer (RAM=16 GB).

  The code parallelisation with OpenMP does enable faster reduction on multi-core platforms. The speed improves significantly if data and reduction is done on very fast I/O disks, like SSDs, and processors with high clock speed.

- **Failure of THAR-FP or FP-THAR reduction** The reduction of THAR-FP or FP-THAR fails sometime, when the `espdr_wave_FP` recipe does not detect FP peaks on the whole order. This can be due to a strong cosmic, badly placed. The consequence is the impossibility to construct the wavelength solution. In this case it is possible to take another FP-FP raw frame for the reduction.

  Other source of possible failures may lie in the use of inputs that have not passed the pipeline checks ("QC <recipe> CHECK"). A check is successful if the value of the keyword is 1.

- **Cosmic rays in raw calibrations.** There have been reported few cases in which the science pipeline crashes with no apparent reason, providing the following error message:

```
10:00:07 [WARNING] espdr_correct_flux: [tid=000] ESC[31mFlux correction not performed:
                   no flux template available for spectral type F5ESC[0m
10:00:07 [ INFO ] : [tid=000] Computing CCF for fibre A sky_sub for size_y = 170 ...
10:00:47 [ ERROR ] espdr_sci_red: [tid=000] espdr_compute_CCF failed for sky_sub:
                   Access beyond boundaries
10:00:47 [ ERROR ] cpl_errorstate_dump: [tid=000] Lost 11192 CPL error(s)
10:00:47 [ ERROR ] cpl_errorstate_dump_one_level: [tid=000] [11193/11212]
                   'Access beyond boundaries' (11) at cpl_image_get:cpl_image_io.c:747
```

  To our best knowledge, the problem is associated to the presence of cosmic rays on some raw calibrations (category: `WAVE,FP,FP`), which led to a product that caused the failure in the science recipe. Future versions of the pipeline will be able to identify such issues and flag them appropriately.

  To overcome the issue, two solutions are currently available:

  - Remove the cosmic ray from the faulty calibrations with external tools before starting the data reduction.
  - Replace the calibrations with those from a previous or following day. It might be worth replacing all the calibrations (all WAVE, ORDERDEF, and FLAT types), even if not all are affected by cosmic rays, to avoid fibre misalignment between calibrations from different days.

- **Interference pattern introduced by the Coude Train** ESPRESSO spectra are affected by two different interference patterns induced by coude train optics. Aproximately sinusoidal "wiggles" become apparent when spectra taken in different telescope positions are divided by each other. The first set of wiggles has a period of 30 Åat 600 nm and an amplitude of $\sim$ 1%). In contrast, the second set of wiggles has a shorter period of 1 Åat 600 nm and an amplitude of $\sim$ 0.1 %. For further information the user is referred to Allart et al. 2020. The consortium and ESO are working together to characterize this effect.

- **wave_FP fails** Occasionally WAVE,FP,FP frames may have cosmics that may induce a recipe failure. For example we encounter this problem and the reported error was like the following:

```
[ INFO  ] espdr_wave_FP: Checking FP peaks for fibre A
[WARNING] espdr_wave_FP: Delta x bigger for order 92 pxl 297.402747, delta =
[WARNING] espdr_wave_FP: Peak_pxls: 291.580109, 297.402747, 309.016369
[ ERROR ] espdr_wave_FP: Missing peaks in the FP image, aborting.
[ ERROR ] espdr_wave_FP: Dumping all 1 error(s):
[ ERROR ] espdr_wave_FP:   [1/1] 'Input data do not match' (13) at espdr_wave
[ ERROR ] esorex: Execution of recipe 'espdr_wave_FP' failed, status = 13
[ INFO  ] esorex: Calculating product checksums
[WARNING] esorex: Writing of output sof omitted due to previous errors
```

Already the warning indicates something suspicious happen. Often the source of the problem is one or more cosmic on the frame. Often this problem can be solved during extraction simply by reducing the parameter **fp_extraction_ksigma** from the default 5 to a smaller value.

- **espdr_orderdef detection of unespected number of orders** The ESPRESSO pipeline cross checks that the number of detected orders is as expected. Occasionally may occur that the number of detected orders changes, for example due to a cosmic that may bridge the two fibers traces corresponding to a given order, this leading to the detection of one less slice and later to a failure in the reduction chain. This may occur more frequently on binned data, in particular the setting 4x2 and 8x4 where the inter-slice space is very small. In such cases we recommend the user to select data from the previous or next day, that possibly do not have such a problem.

- **dependency of results from operative systems**

  In some cases, it has been noticed a small variation on the radial velocity depending on the operative system of few cm/sec. This has to do with the intrinsic precision of the compilers. It is therefore recommended to use the same hardware when reducing the data.

For updated information we recommend the user to also read the Data Reduction F.A.Q. page: *http://www.eso.org/sci/data-processing/faq.html*.

6 Instrument Data Description

The ESPRESSO instrument produces raw data in five different configurations or modes. Those are defined by the main instrument mode and detector binning. The corresponding FITS keywords are:

- HIERARCH ESO INS MODE: instrument mode, defining the fiber head used and the spectral resolution and sampling.
- HIERARCH ESO DET BINX: detector binning in cross-dispersion direction (also called spatial direction)
- HIERARCH ESO DET BINY: detector binning in main dispersion direction

The five available configurations are:

- SINGLEHR mode and 1x1 binning (HR11)
- SINGLEHR mode and 2x1 binning (HR21)
- SINGLEUHR mode and 1x1 binning (UHR)
- MULTIMR mode and 4x2 binning (MR42)
- MULTIMR mode and 8x4 binning (MR84)

The ESPRESSO pipeline considers these five configurations as independent instruments for the purpose of data reduction. This means that, to be processed by the pipeline a science frame in a given configuration requires a complete calibration dataset in the same configuration. As part of the ESPRESSO operational scheme at the observatory, the acquisition of the required calibration frames is automatically triggered based on the science data obtained during the night and the need for instrument monitoring.

To reduce a science frame, the following raw calibration frames are needed:

type	# frames	comments
BIAS	10	to determine RON, bias level, residuals
DARK	5	long exposure time (>20min), to determine hot pixels
LED	2x5	2 different exposure times, to determine non linear pixels and convertion factor
ORDERDEF	2	each frame illuminates only one fiber, to trace orders
FLAT	2x10	2 sets, illuminating fib A and B,
		to determine short and long scale responsiveness variations
FP-FP	1	both fibres illuminated by Fabry Perot etalon, for wavelength calibration
Thorium-FP	1	fib A illuminated by ThAr, fibre B by FP, for wavelength calibration
FP-Thorium	1	fib B illuminated by ThAr, fibre A by FP, for wavelength calibration
SKY,SKY	1	both fibres observing zenith sky, to determine relative fibre efficiency
STD,SKY	1	fib A observes a spectrophotometric standard, fib B observes the sky
		to flux calibrate the observed science spectrum

Optionally, the following calibration frames are needed for wavelength calibration with the laser frequency comb (LFC):

type	# frames	comments
LFC-FP	1	for wavelength calibration, optional
FP-LFC	1	for wavelength calibration, optional

Optionally, the following calibration frame is needed to correct for cross-fiber contamination by the FP in simultaneous-reference mode:

type	# frames	comments
DARK-FP	1	frame to determine contamination of fib B on A

It is also necessary to have a set of static calibration data (see next Section).

The following sections provides a brief description of each raw data type involved in the data reduction chain.

6.1 BIAS frames

Bias frames are zero exposure frames taken to measure the readout noise, the mean bias level, and fixed-pattern structure in the bias level for each read-out port (refer to Figure 4.2.1 for an example of bias detector image).

6.2 DARK frames

Dark frames are taken to measure the detector dark current and identify hot pixels. A set of five input frames is acquired, each with an exposure time of 3600s.

6.3 Detector Flat-field (LED) frames

Detector flat-field frames are images taken while illuminating the detector with a uniform LED light. At least two sets, each composed by at least five frames, with different exposure times are required.

6.4 ORDERDEF frames

Two frames are acquired: ORDERDEF_A and ORDERDEF_B and used for order/slice definition, identification and tracing for fibre A and B. A continuum light source is used to illuminate fibre A (or B), while fibre B (or A) is dark (separate frames for different fibres are necessary for the automatic identification of orders or slices).

6.5 FLAT frames

Two set of frames are acquired: FLAT_A and FLAT_B and used to derive the order profile in cross-dispersion direction, the spectral flat-field, and the blaze function of the spectrograph for fibres A and B. A continuum light source is used to illuminate fibre A (or fibre B), while fibre B (or fibre A) is dark. Several exposures (at least 10) are required to reach sufficiently high SNR.

6.6 Wavelength calibration frames

Three sets of frames are acquired to perform the wavelength calibration. These are acquired with different calibration sources and combined together to cross-calibrate the wavelength references of the two fibres.

- `FP_FP`: This frame is obtained illuminating both fibres with a Fabry Perot.
- `THAR_FP`: This frame is obtained illuminating fibre A with a ThAr lamp and fibre B with a Fabry Perot.
- `FP_THAR`: This frame is obtained illuminating fibre B with a ThAr lamp and fibre A with a Fabry Perot.
- `LFC_FP` (optional): This frame is obtained illuminating fibre A with a laser comb and fibre A with a Fabry Perot.
- `FP_LFC` (optional): This frame is obtained illuminating fibre B with a laser comb and fibre A with a Fabry Perot.

6.7 Contamination by simultaneous reference frames

- `CONTAM_FP`: Measurement of contamination light induced on fibre A by FP simultaneous reference on fibre B. Fibre A is dark, while FP light is injected into fibre B as in a science exposure with simultaneous reference (i.e. same flux level).

6.8 Fiber-to-fiber Relative Efficiency frames

`EFF_AB`: Relative efficiency of fibre B vs. fibre A as a function of wavelength. During twilight, the telescope is pointed at zenith and the skylight is injected into both fibres.

6.9 Spectrophotometric Calibration (FLUX) frames

`FLUX_CALIB`: A spectrophotometric standard star is observed, as in a science exposure, with sky.

6.10 SCIENCE frames

The following frames are acquired:

- `SCIENCE_FP`: Science exposure (target on fibre A) with Fabry-Perot simultaneous reference on fibre B.
- `SCIENCE_SKY`: Science exposure (target on fibre A) with simultaneous sky on fibre B.

7 Static Calibration Data

In the following section static calibration (called also ancillary) data required for ESPRESSO data reduction are listed. Static calibration data correspond to Images or Tables describing configuration setups, reference data, or characteristics of the instrument that are considered to be fixed. This sets them apart from the calibration previously described. For each of them we indicate the corresponding value of the HIERARCH ESO PRO CATG, in short its PRO.CATG, which has to be used to identify the frames listed in the *Set of Frames* (see Section 10.3.1, page 60).

7.1 CCD Geometry Table

These are the static CCD configuration tables describing the CCD geometry. The table contains the number and sizes of the detectors, outputs and prescan and overscan regions. There is one table for each of the supported detector binnings. Its PRO.CATG is `CCD_GEOM`.

7.2 Instrument Configuration Table

These are the static instrument configuration tables providing the pipeline recipes with all necessary input parameters that are intimately linked to the instrument configuration being used. There will be one such table per instrument mode (i.e. five for ESPRESSO) and period of validity. Its PRO.CATG is `INST_CONFIG`.

7.3 Led Flat Field gain windows Table

These static tables indicate for each detector and detector read-out region the windows where the conversion factor is computed. There is one table for each of the supported detector binnings. Their PRO.CATG is `LED_FF_GAIN_WINDOWS`.

7.4 Wavelength Matrix Images

These are the wavelength calibration arrays (one per fibre) in S2D format, with the wavelength of each extracted pixel stored as data value. These are used as static input frames in the flat and wavelength (THAR/FP or FP/THAR) data reduction. Their PRO.CATG is `STATIC_WAVE_MATRIX_A/B`[1].

[1] As static WAVE and DLL matrix frames, have a large size, to reduce the size of the pipeline package, we decided to include them in the demo data, available `http://www.eso.org/sci/software/pipelines/`, and to deliver them to the user together with the raw data when retrieved with the calSelector tool. The FITS filename of the frame delivered with the demo data contains the instrument mode, the binning, the corresponding PRO.CATG and fibre information and a date that correspond to the date since the frame is valid. Raw data should be associated with the closest in time static frame, which is the default behavior in case of Reflex based data reduction.

7.5 DLL Matrix Images

These are the pixel widths in wavelength calibration arrays (one per fibre) in S2D format, with the width in wavelength of each extracted pixel stored as data value. These are used as static input frames in the wavelength (THAR/FP or FP/THAR) and science data reduction. Their PRO.CATG is `STATIC_DLL_MATRIX_A/B`[2].

7.6 THAR line tables

These are the static ThAr line tables (one per fibre) containing the wavelengths, approximate positions, flux and values of D - the FP mirrors distance, for the emission lines found in the ThAr extracted spectra (S2D). The column called "grouping" is used to save the mode of the corresponding FP peak. The table is not expected to change over the lifetime of the instrument owing to the high long-term stability of ESPRESSO. Their PRO.CATG is `REF_LINE_TABLE_A/B`.

7.7 Flux Standard Star tables

The Flux Standard Star Table lists all the stars (part of the calibration plan) for which we have a precise physical flux calibration [ergs/s/cm^2/A] on an even wavelength scale. The table contains absolute spectral energy distributions of a sample of spectrophotometric standard stars. It is a required input of the recipe `espdr_cal_flux` during response and efficiency estimation, where the observed photometric standard star spectrum, after proper rescaling by exposure time, gain and atmospheric extinction, is divided by the spectrum of the corresponding standard star from this catalog. Its PRO.CATG is `STD_TABLE`.

7.8 Extinction Table

This is the table containing the atmospheric extinction curve for Paranal, as provided in Patat et al. (2011), A&A 527, 91 (their Appendix B). This is used in the `espdr_cal_flux` to determine the efficiency and instrument response and in the `espdr_sci_red`, to flux calibrate the observed science object spectrum. Its PRO.CATG is `EXT_TABLE`.

7.9 Flux Template Table

This is the table containing observed spectral energy distributions of a sample of reference stars with different spectral types spanning late-F to early-M. Suitable stars are solar-metallicity dwarf stars of spectral types F to M, observed at high SNR and at low airmass. To build the flux template, the S2D flux of the star is summed in each spectral order and normalised to one at an arbitrary wavelength (e.g. 550 nm). The flux template is simply the integrated normalised flux in each spectral order. Flux Template table corresponds to the flux distribution as a function of wavelength for stars of different spectral types. This primordial flux is used to correct the effect of chromatic extinction of the atmosphere, composed of several factors, like Rayleight scattering and ozone + aerosols absorption. Its PRO.CATG is `FLUX_TEMPLATE`.

[2] DLL stands for Delta LL, a short notation corresponding to Delta Wavelength.

7.10 CCF Template Tables

The template tables, or CCF masks, are used by the cross-correlation process. They consist of a list of rest (RV=0) wavelengths for the spectral lines of interest, along with their relative depth (contrast) with respect to the continuum. Depths are used as weights in the CCF process since Doppler precision is proportional to line depth. Their PRO.CATG are `MASK_TABLE`.

These CCF masks are created from actual ESPRESSO spectra of stars covering a range of sub-spectral types. Spectral lines are identified through an automatic procedure and included in the mask if they meet a number of criteria such as large enough depth and limited blending. The current pipeline version includes CCF masks for all M dwarfs down to M5 (individual fits files). The masks cover the full wavelength range of ESPRESSO. For QSO observations the CCF computation on the object spectrum is tuned off.

8 Data Reduction

We give below an overview of the global reduction cascade, starting from basic calibrations down to science reduction. The ESPRESSO association map is shown in Figure 8.0.0.

- Detector bias level and readout noise are measured on stacked BIAS frames (master bias). The stacking is used to remove the cosmics. In ESPRESSO the mean bias level and readout noise in any raw frame are best obtained from the overscan regions. The master bias is thus not used as such in the reduction cascade. A master bias *residual* frame is generated by the `espdr_mbias` recipe by subtracting the overscan from the master bias. It is used instead to subtract the fixed residual bias level structure across detector outputs. The residual bias level pattern is not negligible for low-SNR science exposures.

- DARK frames are used to measure the average detector dark current and identify the hot pixels. The master dark frame is generated via stacking and sigma-clipping to remove the cosmics. The mean dark current level per output and the hot pixels are computed on the master dark frame and stored in the hot pixel mask. The hot pixels are obtained through the sigma-clipping applied on the master image of each detector output.

- Detector flat-field frames are acquired to characterise unresponsive ("dead") pixels or pixels with a very different gain (so called: bad pixels). For each exposure time, a set of minimum five LED_FF frames is needed to remove the cosmics via the sigma clipping on the pixel-by-pixel basis. The bad pixel mask is obtained from the analysis of master LED_FF frames taken with different exposure times (linearity check). The conversion factor is measured for each detector output from the relation between flux level and standard deviation of pixel values. It is stored in the bad pixel mask.

- ORDERDEF frames (one per fibre) are used to identify and trace spectral orders or slices on the detector. Each order or slice is fitted in the cross-dospersion direction, with a gaussian every defined number of pixels. The gaussians peaks are fitted with a low degree polynomial, of which coefficients are saved in the recipe products.

- For each fibre, the order profile in cross-dispersion direction is found using high-SNR, co-added FLAT frames. Then, the orders are extracted from the FLAT frames using this profile, and the spectral flat-field is generated. The (extracted) blaze function is obtained through smoothing of the FLAT spectra and correction for the spectral energy distribution of the calibration lamp. The order extraction assumes that 1) main-dispersion direction runs approximately parallel to CCD rows/columns, and 2) slit image tilt is close to zero with respect to CCD columns/rows. In this case, order extraction becomes extremely simple and does not require wavelength calibration frames to track different positions along the slit. The ESPRESSO optical design makes this strategy possible (line tilt very close to zero), and even recommended owing to its simplicity. This method has been successfully applied to HARPS and other radial-velocity spectrographs.

- CONTAM frames are used to measure cross-fibre contamination on fibre A from the simultaneous reference on fibre B (ThAr lamp, laser comb or Fabry-Perot). Contamination frames are used during extraction in the science reduction.

- The relative efficiency of channels A and B as a function of wavelength is measured using EFF_AB frames, which are obtained through blue sky observations. The obtained relative efficiency is used to scale the subtraction on science exposures with simultaneous sky. Even though these are called fibre-to-fibre relative efficiency they, de facto, calibrate the whole Coude train as seen through fibre A and B. As such, they are UT dependent.

- The wavelength calibration for both fibres is determined using WAVE frames. It is done in two steps: first with the Fabry-Perot on both fibres and then with one of the fibres illuminated by ThAr lamp. In addition the wavelength solution can be computed using frames with Laser Frequency Comb. Since the LFC does not cover the whole spectral range, the THAR-FP solution is needed to complete the calibration products. On fibre A (science fibre), the wavelength solution is particularly important since it is used to calibrate the science spectrum and therefore sets the radial velocity zero point. On fibre B, the simultaneously acquired spectrum mainly serves as reference to calculate the drift on science frames that use FP as simultaneous reference.
- FLUX_CALIB frames are used to compute the absolute efficiency of the instrument as a function of wavelength, using spectrophotometric standard stars. The efficiency is computed from the comparison between the observed spectrum and reference flux table.The efficiency curve is used in the science reduction to calibrate the science spectrum in flux. The precision of the flux calibration is generally low because of highly variable fibre losses due to seeing.
- Finally, SCIENCE frames are of two different sub-types depending on the source of light on the fibre B: sky or Fabry-Perot. Science reduction makes use of all calibration products listed above and generates extracted S2D spectra and merged, rebinned S1D spectra, together with S2D and S1D error and quality maps. Finally, the cross-correlation function (CCF) of the S2D spectrum is computed and the radial velocity is obtained from a Gaussian fit to the CCF.

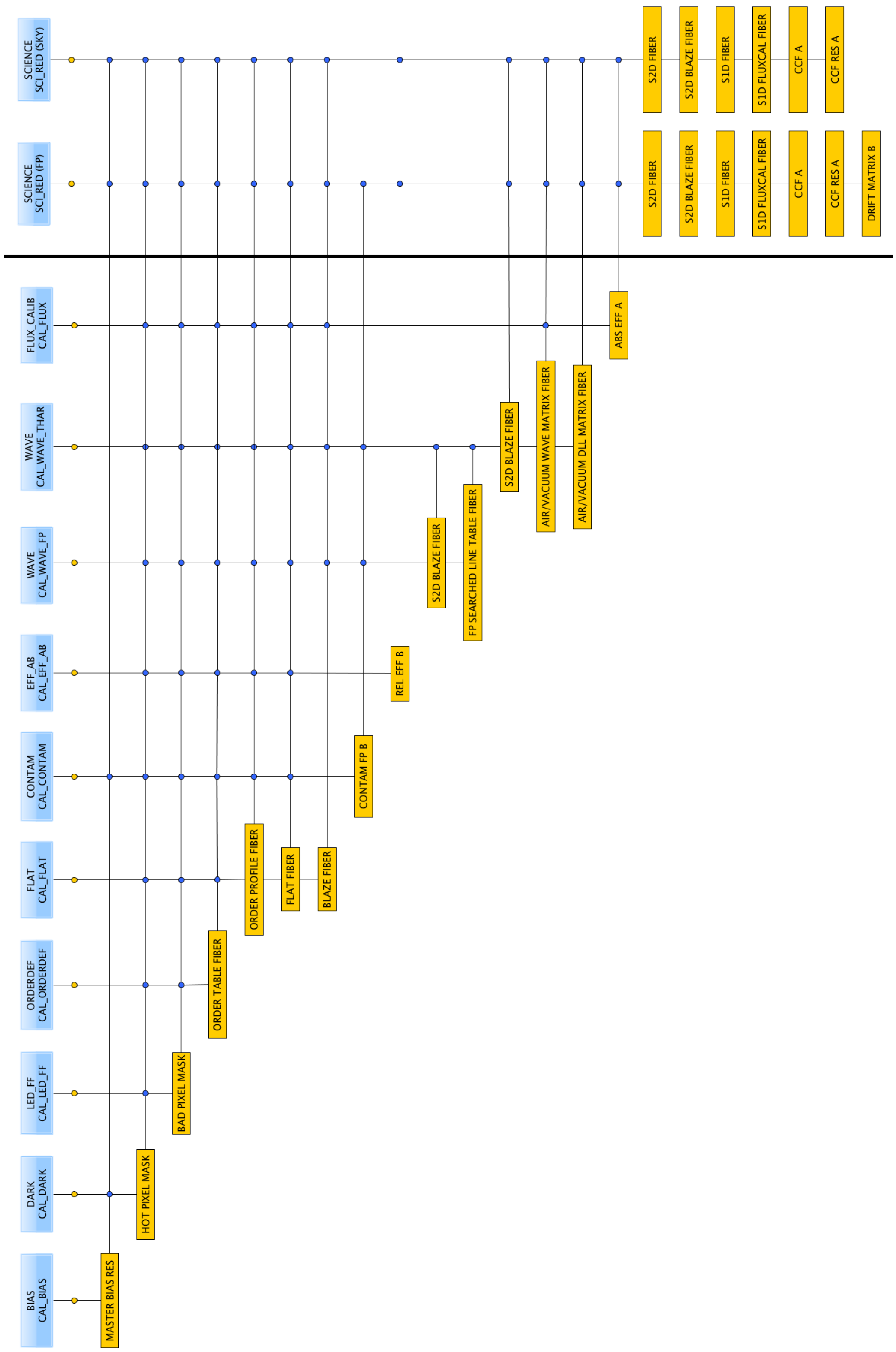

Figure 8.0.0: The cascade of the ESPRESSO pipeline recipes. For the calibration recipes, only the products used later in the reduction chain are listed. The science recipes have all the products listed.

9 Pipeline Recipes Interfaces

In this section we provide for each recipe [3] examples of the required input data (and their tags). In the following we assume that /path_file_raw/filename_raw.fits and /path_file_cdb/filename_cdb.fits are existing FITS files.

We also provide a list of the pipeline products for each recipe, indicating their default recipe name, the value of the FITS keyword HIERARCH ESO PRO CATG (in short PRO.CATG) and a short description.

The relevant keywords used to classify each frame are the following:

Association keyword	Information
HIERARCH ESO DPR TYPE	data type
HIERARCH ESO INS MODE	mode
HIERARCH ESO DET BINX	detector bin X
HIERARCH ESO DET BINY	detector bin Y
HIERARCH ESO PRO CATG	product category

For each recipe we list in a table the input parameters (as they appear in the recipe configuration file), the corresponding aliases (the corresponding names to be eventually set on command line) and their default values. Also quality control parameters are listed. Those are stored in relevant pipeline products. More information on instrument quality control can be found on `www.eso.org/qc`. ESPRESSO spectral format data of the BLUE and RED camera are stored in two image extensions.

The user may obtain brief description of the main input recipe parameters by typing

esorex -help recipe,

for example,

esorex -help `espdr_mbias`

A possible esorex configuration parameter value is –suppress-prefix=FALSE, in which case all products will be renamed with a prefix settable by the parameter –output-prefix, defaulted to `out_`, and an increasing number, like (`out_0000.fits`, `out_0001.fits`, `out_000N.fits`) For this reason the table briefly describing the products contains also a first column indicating the product ID, which is the value of the product number (with minimum significant digits).

The pipeline performs several quality assesment on the data. The result is stored in a QC keyword that has value "QC <specifier> CHECK", where the <specifier> indicates the kind of check done. A check is successful if the value of the keyword is 1. Some recipe may do several tests. It is also created a keyword "QC <recipe> CHECK", where <recipe> indicates the recipe executed, that indicates the overall product of all checks.

The PRO.CATG chosen for the extracted spectra falls into two groups: 'S2D_*' spectra, that can be of type A or B to represent the object or the calibration fibersi, respectively. These files are 2D images where the rows (Y axis) contain the extracted spectrum of a given order or slice. We remind the reader that in order to keep compact the spectrograph the input light beam from the input slices is split in two slices. These are dispersed by the grating and imaged on the detectors (one for each instrument arm) on different orders. As the input beam

[3] We do not describe here two recipes: the recipe espdr_single_bias, used to reduce HARPS data, and the recipe espdr_wave_TH_drift, used for the wavelength calibration when are not available enough good quality FP, FP frames.

is sliced in two component the order shows two components, adjacent one to another. These order slices are distinguishible in UHR and HR modes and overlap in the MR mode. There are 45 orders in the Blue and 40 in the red. In UHR and HR modes the pipeline extracts then two slices for each order, and in MR only one.

9.1 espdr_mbias

9.1.1 Input

type	TAG	n	setting
raw	BIAS	5...n	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match

9.1.2 Output

ID	PRO.CATG	type	Note
0	MASTER_BIAS	qc	Master bias
1	MASTER_BIAS_RES	cdb	Master bias residuals

9.1.3 Quality control

This recipe computes the following QC parameters (on the image extension n, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj BIAS RON	read out noise [ADU].
QC EXTn ROXi ROYj BIAS RON EL	read out noise [e-]
QC EXTn ROXi ROYj OVSC RON	read out noise on overscan region [ADU]
QC EXTn ROXi ROYj OVSC RON EL	read out noise on overscan region [e-]
QC BIAS RON CHECK	quality check on RON computation
QC EXTn ROXi ROYj BIAS MEAN	mean bias [ADU]
QC EXTn ROXi ROYj BIAS MEAN EL	mean value [e-]
QC EXTn ROXi ROYj BIAS STRUCTX	X structure
QC EXTn ROXi ROYj BIAS STRUCTY	Y structure
QC EXTn ROXi ROYj RES MEAN	mean value of residuals [ADU]
QC EXTn ROXi ROYj RES MEAN EL	mean value of residuals [e-]
QC EXTn ROXi ROYj RES STDEV	rms of residuals [ADU]
QC EXTn ROXi ROYj RES STDEV EL	rms of residuals [e-]
QC EXTn ROXi ROYj RES TEST	quality check on residual computation on read-out region n
QC RES TEST	overall quality check on residual computation
QC BIAS OUTLIERS	number out utliers on bias
QC OVSC OUTLIERS	number of outliers on region
QC BIAS MEAN CHECK	quality check on bias mean computation

QC BIAS CHECK	overall quality check assessment
QC BIAS METHOD	data reduction method used

9.1.4 Parameters

parameter name	description
bias_sig_clip_method	method for sigma clipping in MBIAS, can be: mean or median. [median]
bias_ksigma	ksigma for sigma clipping in MBIAS, must be between: 1.50 and 1000.00. [14.0]
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]

9.2 espdr_mdark

9.2.1 Input

type	TAG	n	setting
raw	DARK	5...n	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
cdb	MASTER_BIAS_RES	1	match

9.2.2 Output

ID	PRO.CATG	type	Note
0	HOT_PIXEL_MASK	cdb	Hot pixels mask
1	MASTER_DARK	qc	Master dark

9.2.3 Quality control

This recipe computes the following QC parameters (on the image extension n, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj DARK MEAN	mean dark [ADU/px/hour]
QC EXTn ROXi ROYj HOTPIX NB	number of hot pixels
QC EXTn COSMIC RATE PIX	number of cosmic hits per pixel per hour
QC EXTn COSMIC RATE CM2	number of cosmic hits per cm2 per hour
QC DARK COSMIC CHECK	check on cosmic rate computation results
QC DARK MEAN CHECK	check on mean dark results
QC DARK HOTPIX CHECK	check on hotpixel detection

QC DARK CHECK	overall check on dark

9.2.4 Parameters

parameter name	description
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
dark_sig_clip_method	method for sigma clipping in master DARK, can be: mean or median. [median]
dark_ksigma	ksigma for sigma clipping in MDARK, must be between: 1.50 and 1000.00. [10.0]
hot_pix_sig_clip_method	method for sigma clipping in HOT PIXELS, can be mean or median. [mean]
hot_pix_ksigma	ksigma for sigma clipping in HOT PIXELS, must be between: 1.50 and 1000.00. [10.0]
hot_pix_max_iter	maximal number of iterations in HOT PIXELS, must be between: 1 and 5000000. [10000]

9.3 espdr_led_ff

9.3.1 Input

type	TAG	n	setting
raw	LED_FF	10...n[4]	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
ref	LED_FF_GAIN_WINDOWS	1	match
cdb	HOT_PIXEL_MASK	1	match

9.3.2 Output

ID	PRO.CATG	type	Note
0	BAD_PIXEL_MASK	cdb	Bad pixels mask

9.3.3 Quality control

This recipe computes the following QC parameters (on the image extension n, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux on raw image [ADU]
QC EXTn ROXi ROYj CONAD	conversion factor [e-/ADU]
QC EXTn ROXi ROYj BADPIX NB	bad pixel number
QC EXTn ROXi ROYj MASTER MAX FLUX	max flux on master [ADU]

[4] At least five input raw frames for each exposure time, at least two exposure times.

QC EXTn ROXi ROYj MASTER MIN FLUX	min flux on master [ADU]
QC EXTn ROXi ROYj MASTER MEAN FLUX	mean flux on master [ADU]
QC MASTER MIN FLUX	overall min flux on master [ADU]
QC MASTER MAX FLUX	overall max flux on master [ADU]
QC SATURATION CHECK	check on frame saturation
QC LEDFF MIN FLUX CHECK	check on minimum flux
QC LEDFF BADPIX CHECK	check on bad pixels
QC LEDFF CONAD CHECK	check on conad
QC LEDFF CHECK	overall check on results

9.3.4 Parameters

parameter name	description
`ovsc_sig_clip_method`	method for sigma clipping in OVSC, can be mean or median. [mean]
`ovsc_ksigma`	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
`ovsc_max_iter`	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
`led_ff_sig_clip_method`	method for sigma clipping in master LED_FF, can be: mean or median. [median]
`led_ff_ksigma`	ksigma for sigma clipping in master LED_FF, must be between: 1.50 and 1000.00. [10.0]
`bad_pix_sig_clip_method`	method for sigma clipping in BAD PIXELS, can be mean or median. [mean]
`bad_pix_ksigma`	ksigma for sigma clipping in BAD PIXELS, must be between: 1.50 and 1000.00. [10.0]
`bad_pix_max_iter`	maximal number of iterations in BAD PIXELS, must be between: 1 and 5000000. [10000]

9.4 espdr_orderdef

9.4.1 Input

type	TAG	n	setting
raw	`ORDERDEF_A`	1	any
raw	`ORDERDEF_B`	1	any
ref	`CCD_GEOM`	+ 1	match
ref	`INST_CONFIG`	1	match
cdb	`HOT_PIXEL_MASK`	1	match
cdb	`BAD_PIXEL_MASK`	1	match

9.4.2 Output

ID	PRO.CATG	type	Note
0	`ORDER_TABLE_A`	cdb	Order table tracing fibre A
1	`ORDER_TABLE_B`	cdb	Order table tracing fibre B

9.4.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description

QC EXTn ROXi ROYj MAX FLUX	max flux on raw image [ADU]
QC ORDERm EXTn POS	order position [pix]
QC ORDERm EXTn RES MIN	min residuals [pix]
QC ORDERm EXTn RES MAX	max residuals [pix]
QC ORDERm EXTn RES STDEV	rms residuals [pix]
QC ORDERm EXTn NB	physical order number
QC ORDER NB	number of orders in the image
QC EXTn ROXi ROYj MAX FLUX	max flux, raw image [ADU]
QC SATURATION CHECK	check on saturation
QC ORDERDEF ORDER CHECK	check on number of detected orders
QC ORDERDEF STDEV CHECK	check on rms residuals
QC ORDERDEF MIN CHECK	check on min residuals
QC ORDERDEF MAX CHECK	check on max residuals
QC ORDERDEF CHECK	overall check

9.4.4 Parameters

parameter name	description
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]

9.5 espdr_mflat

9.5.1 Input

type	TAG	n	bin
raw	FLAT_A	5...n	any
raw	FLAT_B	5...n	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
ref	STATIC_WAVE_MATRIX_A	1	match
ref	STATIC_WAVE_MATRIX_B	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match

9.5.2 Output

ID	PRO.CATG	type	Note
0	FF_BACKGROUND_MAP_A	cdb	Background map for fibre A
1	FS2D_A	cdb	Order by order exracted spectrum for fibre A
2	BLAZE_A	cdb	Blaze for fibre A
3	FSPECTRUM_A	cdb	Order by order exracted spectrum for fibre A
4	ORDER_PROFILE_A	cdb	Order profile for fibre A
5	FF_BACKGROUND_MAP_B	cdb	Background map for fibre B
6	FS2D_B	cdb	Order by order exracted spectrum for fibre B
7	BLAZE_B	cdb	Blaze for fibre B

8	FSPECTRUM_B	cdb	Order by order exracted spectrum for fibre B
9	ORDER_PROFILE_B	cdb	Order profile for fibre B

9.5.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux on raw image [ADU]
QC ORDERm FLAT RMS	FLAT RMS in the order
QC ORDERm SNR	SNR in the order
QC ORDERm COSMIC NB	cosmics number in the order
QC EXTn ROXi ROYj MAX FLUX	max flux, raw image [ADU]
QC EXTn BKGR MEAN	bkgr mean in the extension [e-]
QC EXTn BKGR MIN	bkgr min in the extension [e-]
QC EXTn BKGR MAX	bkgr max in the extension [e-]
QC SATURATION CHECK	quality check on saturation
QC FLAT RMS CHECK	quality check on rms
QC FLAT SNR CHECK	quality check on SNR
QC FLAT BKGR CHECK	background quality check
QC FLAT CHECK	overall quality check

9.5.4 Parameters

parameter name	description
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
flat_sig_clip_method	method for sigma clipping in master FLAT, can be: mean or median. [median]
flat_ksigma	ksigma for sigma clipping in MFLAT, must be between: 1.50 and 1000.00. [10.0]
background_sw	Background measurement activation (on/off). [on]
bkgr_grid_size_x	Grid size in x used to calculate the background, between: 16 and 10000. [577]
bkgr_grid_size_y	Grid size in y used to calculate the background, between: 16 and 10000. [256]
flat_extraction_method	Method used to extract orders. [horne]
flat_extraction_ksigma	ksigma for extraction, must be between: -1.00 and 20.00. [-1.0]

9.6 espdr_wave_FP

9.6.1 Input

type	TAG	n	setting
raw	FP_FP	1	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match

cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match
cdb	BLAZE_A	1	match
cdb	BLAZE_B	1	match

9.6.2 Output

ID	PRO.CATG	type	Note
0	S2D_FP_FP_A	cdb	S2D Extracted spectrum for fibre A
1	S2D_FP_FP_B	cdb	S2D Extracted spectrum for fibre B
2	S2D_BLAZE_FP_FP_A	cdb	S2D Extracted blaze for fibre A
3	S2D_BLAZE_FP_FP_B	cdb	S2D Extracted blaze for fibre B
4	FP_SEARCHED_LINE_TABLE_FP_FP_A	cdb	Detected FP peaks on fibre A
5	FP_SEARCHED_LINE_TABLE_FP_FP_B	cdb	Detected FP peaks on fibre B

9.6.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux on raw image [ADU]
QC ORDERm COSMIC NB	number of cosmics detected in order m
QC SATURATION CHECK	quality check on saturation
QC COSMIC NB	number of cosmics
QC WAVE CHECK	overall quality check

9.6.4 Parameters

parameter name	description
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
fp_extraction_method	Method used to extract orders. [horne]
fp_extraction_ksigma	ksigma for extraction, must be between: -2.00 and 20.00. [5.0]

9.7 espdr_wave_THAR

9.7.1 Input

type	TAG	n	setting
raw	THAR_FP	1	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
ref	STATIC_DLL_MATRIX_A	1	match
ref	STATIC_WAVE_MATRIX_A	1	match
ref	REF_LINE_TABLE_A	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match

cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match
cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match
cdb	BLAZE_A	1	match
cdb	BLAZE_B	1	match
cdb	S2D_BLAZE_FP_FP_B	1	match
cdb	FP_SEARCHED_LINE_TABLE_FP_FP_A	1	match

And for the other fibre a similar input where THAR_FP is replaced by FP_THAR, STATIC_WAVE/DLL_MATRIX_A are replaced by STATIC_WAVE/DLL_MATRIX_B, FP_SEARCHED_LINE_TABLE_FP_FP_A is replaced by FP_SEARCHED_LINE_TABLE_FP_FP_B and
REF_LINE_TABLE_A is replace by REF_LINE_TABLE_B

9.7.2 Output

ID	PRO.CATG	type	Note
0	S2D_THAR_FP_A	cdb	S2D Extracted spectrum for fibre A
1	S2D_THAR_FP_B	cdb	S2D Extracted spectrum for fibre B
2	S2D_BLAZE_THAR_FP_A	cdb	S2D Extracted blaze for fibre A
3	S2D_BLAZE_THAR_FP_B	cdb	S2D Extracted blaze for fibre B
4	WAVE_MATRIX_THAR_FP_A	cdb	Wave map (vacuum) for fibre A
5	AIR_WAVE_MATRIX_THAR_FP_A	cdb	Wave map (air) for fibre A
6	DLL_MATRIX_THAR_FP_A	cdb	DLL map (vacuum) fibre A
7	AIR_DLL_MATRIX_THAR_FP_A	cdb	DLL map (air) fibre A
8	WAVE_TABLE_THAR_FP_A	cdb	Table of coefficients of the wave sol fit for each order
9	FP_FITTED_LINE_TABLE_THAR_FP_A	cdb	Table of fitted FP lines used in the wave. sol. for fibre A
10	THAR_LINE_TABLE_THAR_FP_A	cdb	ThAr line table for fibre A
11	LINE_TABLE_RAW_THAR_FP_A	cdb	Table of ThAr lines fitted on the raw frame for fibre A

DLL files contain pixel widths in wavelength units. For the FP_THAR reduction the pipeline products have similar PRO.CATGs except that from Id 4 on they refer to fibre B instead of A.

9.7.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC DRIFT DETn SLOPE_O	Drift slope across the orders per detector
QC DRIFT DETn SLOPE_O_ERR	Drift slope error
QC DRIFT DETn SLOPE_X	Drift slope along the orders per detector
QC DRIFT DETn SLOPE_X_ERR	Drift slope error
QC DRIFT DETn CHISQ	CHI2 of the drift fit
QC DRIFT DETn REJECTED	nb of rejected pixels
QC DRIFT DETn MEAN	mean drift per detector
QC DRIFT DETn MEAN_ERR	mean drift error
QC DRIFT DETn FLUX_RATIO	flux ratio per detector
QC DRIFT DETn FLUX_RATIO_ERR	flux ratio error
QC DRIFT DETn FIRST_ORDER	first order on the detector

QC DRIFT DETn LAST_ORDER	last order on the detector
QC DRIFT CHI2 CHECK	quality check on the drift
QC ORDERm FP LINES NB	number of valid FP lines in order m
QC ORDERm CHI2	reduced CHI2 of ll solution for order m
QC ORDERm RMS	dispersion of residuals around mean
QC THAR RESOL MEDIAN	ThAr lines resolution median
QC THAR SATUR LINES NB	saturated ThAr lines number
QC THAR LINES NB	Valid ThAr lines number
QC THAR FLUX RATIO MEDIAN	Thar lines flux ratio median
QC THAR RV DIFF MEDIAN	[m/s] Thar lines RV difference median
QC FP D OFFSET	[Angstrom] D(lambda) offset
QC FP D OFFSET ERROR	[Angstrom] D(lambda) offset error
QC FP D OFFSET RMS	[Angstrom] D(lambda) offset RMS
QC FP LINES NB	Valid FP lines number
QC LINES ORDER MIN CHECK	quality check on min order residuals
QC LINES TOT CHECK	quality check on total lines number
QC FP CHI2 CHECK	quality check on chi2
QC FP RMS CHECK	quality check on rms THAR
QC THAR RESOL CHECK	quality check on resolution
QC THAR FLUX RATIO CHECK	QC on TH lines flux ratio
QC FP SOL QUAL CHECK	quality check on wave sol quality
QC WAVE CHECK	overall quality check

9.7.4 Parameters

parameter name	description
`ovsc_sig_clip_method`	method for sigma clipping in OVSC, can be mean or median. [mean]
`ovsc_ksigma`	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
`ovsc_max_iter`	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
`thar_extraction_method`	Method used to extract orders. [horne]
`thar_extraction_ksigma`	ksigma for extraction, must be between: -2.00 and 20.00. [-1.0]
`d_fit_poly_deg`	d fit poly deg in wave_THAR, must be between: 1 and 50. [25]
`d_fit_small_res_limit`	d fit small res_limit in wave_THAR, must be between: 99 and 99. [10.0]
`wave_sol_poly_deg`	wavelength solution poly deg in wave_THAR, must be between: 1 and 1. [10]

9.8 espdr_wave_LFC

9.8.1 Input

type	TAG	n	setting
raw	`FP_LFC`	1	any
ref	`CCD_GEOM`	1	match
ref	`INST_CONFIG`	1	match
cdb	`HOT_PIXEL_MASK`	1	match
cdb	`BAD_PIXEL_MASK`	1	match
cdb	`ORDER_TABLE_A`	1	match
cdb	`ORDER_TABLE_B`	1	match
cdb	`ORDER_PROFILE_A`	1	match
cdb	`ORDER_PROFILE_B`	1	match
cdb	`FSPECTRUM_A`	1	match
cdb	`FSPECTRUM_B`	1	match
cdb	`BLAZE_A`	1	match

cdb	BLAZE_B	1	match
cdb	DLL_MATRIX_THAR_FP_A	1	match
cdb	WAVE_MATRIX_THAR_FP_A	1	match

And for the other fibre a similar input where FP_LFC is replaced by LFC_FP and the WAVE and DLL MATRIXes for FP_THAR are used instead the THAR_FP ones.

9.8.2 Output

ID	PRO.CATG	type	Note
0	S2D_LFC_FP_A	cdb	S2D Extracted spectrum for fibre A
1	S2D_LFC_FP_B	cdb	S2D Extracted spectrum for fibre B
2	S2D_BLAZE_LFC_FP_A	cdb	S2D Extracted blaze for fibre A
3	S2D_BLAZE_LFC_FP_B	cdb	S2D Extracted blaze for fibre B
4	WAVE_MATRIX_LFC_FP_A	cdb	Wave map (vacuum) for fibre A
5	AIR_WAVE_MATRIX_LFC_FP_A	cdb	Wave map (air) for fibre A
6	DLL_MATRIX_LFC_FP_A	cdb	DLL map (vacuum) fibre A
7	AIR_DLL_MATRIX_LFC_FP_A	cdb	DLL map (air) fibre A
8	WAVE_TABLE_LFC_FP_A	cdb	Table of coefficients of the wave sol fit for each order
9	LFC_FITTED_LINE_TABLE_LFC_FP_A	cdb	Table of fitted LFC lines used in the wave. sol. for fibre A

For the FP_LFC reduction the pipeline products have similar PRO.CATGs except that from Id 4 on they refer to fibre B instead of A.

9.8.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux in the raw image [ADU]
QC ORDERm COSMIC NB	number of detected cosmics in order m
QC COSMIC NB	total number of detected cosmics
QC ORDERm LFC LINES NB	number of valid lines in order m
QC ORDERm CHI2	reduced CHI2 of ll solution for order m
QC ORDERm RMS	dispersion of residuals around mean
QC ORDERm LFC FIT WAVE	flag indicating if the wave sol was fitted with LFC for order m
QC LFC LINES NB	total number of valid LFC lines
QC LFC LINES ORDER MIN CHECK	quality check on min number of LFC lines per order
QC LFC LINES TOT CHECK	quality check on total detected lines number
QC LFC CHI2 CHECK	quality check on CHI2 of wave sol
QC LFC RMS CHECK	quality check on RMS of wave sol
QC SATURATION CHECK	Check on saturation
QC LFC WAVE CHECK	overall quality check

9.8.4 Parameters

parameter name	description
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]

lfc_extraction_method	Method used to extract orders. [horne]
lfc_extraction_ksigma	ksigma for extraction, must be between: -1.00 and 20.00. [10.0]
wave_sol_poly_deg	wavelength solution polynomial degree in wave_THAR, must be between: 3 and 10. [7]

9.9 espdr_cal_contam

9.9.1 Input

type	TAG	n	setting
raw	RAW_CONTAM_FP	1	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
cdb	MASTER_BIAS_RES	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match
cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match

9.9.2 Output

ID	PRO.CATG	type	Note
0	CONTAM_S2D_A	qc	S2D Extracted spectrum for fibre A
1	CONTAM_S2D_B	qc	S2D Extracted spectrum for fibre B
2	CONTAM_FP	cdb	Contamination of fibre B on fibre A

9.9.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux, raw image
QC SATURATION CHECK	quality check on saturation
QC ORDERm MAX FLUX	S2D flux max, fibre A
QC CONTAM FLUX CHECK	quality check on flux
QC CONTAM CHECK	overall quality check

9.9.4 Parameters

parameter name	description
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
background_sw	if on (default value) then background is subtracted, 1=on 0=off. [0]
contam_bkgr_grid_size_x	Grid size in x used to calculate the background, between: 128 and 512. [577]

contam_bkgr_grid_size_y	Grid size in y used to calculate the background, between: 128 and 512. [256]
contam_extraction_method	Method used to extract orders. [horne]
contam_extraction_ksigma	ksigma used to extract orders. [3.5]

9.10 espdr_cal_eff_ab

9.10.1 Input

type	TAG	n	setting
raw	EFF_AB	1	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match
cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match

9.10.2 Output

ID	PRO.CATG	type	Note
0	S2D_BLAZE_EFF_A	qc	S2D Extracted blaze spectrum for fibre A
1	S2D_BLAZE_EFF_B	qc	S2D Extracted blaze spectrum for fibre B
2	REL_EFF_B	cdb	spectrum describing the trasmission ratio fibre B/fibre A in pixel units.

9.10.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux, raw image [ADU]
QC SATURATION CHECK	check on saturation
QC EXTn BKGR MEAN	bkgr mean in the extension [e-]
QC EXTn BKGR MIN	bkgr min in the extension [e-]
QC EXTn BKGR MAX	bkgr max in the extension [e-]
QC EFFAB BKGR CHECK	check on background level
QC ORDERm SNR	SNR in the order m
QC ORDERm COSMIC NB	cosmics number in the order
QC EFFAB SNR CHECK	check on SNR
QC REL EFF MIN	min relative efficiency
QC REL EFF MAX	max relative efficiency
QC REL EFF MIN CHECK	check on computation min efficiency
QC REL EFF MAX CHECK	check on computation max efficiency
QC EFFAB CHECK	overall check on effab products

9.10.4 Parameters

parameter name	description
ovsc_sig_clip_method	method for sigma clipping in OVSC, can be mean or median. [mean]
ovsc_ksigma	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
ovsc_max_iter	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
eff_ab_bkgr_grid_size_x	Grid size in x used to calculate the background, between: 128 and 512. [577]
eff_ab_bkgr_grid_size_y	Grid size in y used to calculate the background, between: 128 and 512. [256]
eff_ab_extraction_method	Method used to extract orders. [horne]
eff_ab_extraction_ksigma	ksigma used to extract orders. [5.0]
eff_ab_poly_deg	Efficiency computation fit polynomial degree. [3]

9.11 espdr_cal_flux

9.11.1 Input

type	TAG	n	setting
raw	FLUX	1	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
ref	EXT_TABLE	1	match
ref	STD_TABLE	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match
cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match
cdb	BLAZE_A	1	match
cdb	BLAZE_B	1	match
cdb	WAVE_MATRIX_THAR_FP_A	1	match
cdb	WAVE_MATRIX_FP_THAR_B	1	match

9.11.2 Output

ID	PRO.CATG	type	Note
0	S2D_STD_A	qc	S2D Extracted spectrum for fibre A
1	S1D_STD_A	qc	S1D Extracted spectrum for fibre A
2	S1D_ENERGY_STD_A	cdb	S1D energy for fibre A
3	S2D_BLAZE_STD_A	cdb	S2D blaze spectrum for fibre A
4	AVG_FLUX_STD_A	cdb	average flux for fibre A
5	ABS_EFF_RAW_A	cdb	absolute raw efficiency for fibre A
6	ABS_EFF_A	cdb	absolute efficiency for fibre A

9.11.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux, raw image [ADU]
QC SATURATION CHECK	saturation [ADU] QC
QC ORDERm COSMIC NB	cosmics number in the order m
QC ORDERm SNR	SNR in the order m
QC CALFLUX SNR CHECK	check on SNR computed value
QC EXTn BKGR MEAN	bkgr mean in the extension [e-]
QC EXTn BKGR MIN	bkgr min in the extension [e-]
QC EXTn BKGR MAX	bkgr max in the extension [e-]
QC MIN RAW ABS EFF	min raw absolute efficiency
QC MAX RAW ABS EFF	max raw absolute efficiency
QC MEAN RAW ABS EFF	mean raw absolute efficiency
QC MIN EXTCORR ABS EFF	minimum extinction-corrected abs eff
QC MAX EXTCORR ABS EFF	maximum extinction-corrected abs eff
QC MEAN EXTCORR ABS EFF	mean extinction-corrected abs eff
QC MIN ABS EFF	minimum absolute efficiency
QC MAX ABS EFF	maximum absolute efficiency
QC MEAN ABS EFF	mean absolute efficiency
QC MIN ABS EFF CHECK	minimum absolute efficiency check
QC MAX ABS EFF CHECK	maximum absolute efficiency check
QC BERV[5]	barycentric correction [km s^{-1}]
QC BJD	barycentric Julian date (TDB) [JD]
QC BERVMAX	barycentric max [km s^{-1}]
QC CALFLUX SEEING KW CHECK	check on seeing
QC CALFLUX CHECK	overall check on recipe

9.11.4 Parameters

parameter name	description
`ovsc_sig_clip_method`	method for sigma clipping in OVSC, can be mean or median. [mean]
`ovsc_ksigma`	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
`ovsc_max_iter`	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
`background_sw`	if 1 (default value) then background is subtracted, 1=on 0=off. [0]
`bkgr_grid_size_x`	Grid size in x used to calculate the background, between: 128 and 512. [577]
`bkgr_grid_size_y`	Grid size in y used to calculate the background, between: 128 and 512. [256]
`flux_extraction_method`	Method used to extract orders. [horne]
`flux_extraction_ksigma`	ksigma used to extract orders. [3.5]
`flux_poly_deg`	Efficiency computation fit polynomial degree. [6]

9.12 espdr_sci_red

9.12.1 Input

type	TAG	n	setting
raw	`OBJ_FP`	1	any
ref	`CCD_GEOM`	1	match
ref	`INST_CONFIG`	1	match

[5] Barycentric Earth Radial Velocity (BERV)

ref	MASK_TABLE	11	match
ref	EXT_TABLE	1	match
ref	MASK_LUT	1	match
ref	FLUX_TEMPLATE	1	match
cdb	MASTER_BIAS_RES	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match
cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match
cdb	BLAZE_A	1	match
cdb	BLAZE_B	1	match
cdb	WAVE_MATRIX_THAR_FP_A	1	match
cdb	WAVE_MATRIX_FP_THAR_B	1	match
cdb	DLL_MATRIX_THAR_FP_A	1	match
cdb	DLL_MATRIX_FP_THAR_B	1	match
cdb	S2D_BLAZE_THAR_FP_B	1	match
cdb	WAVE_MATRIX_LFC_FP_A	?[6]	match
cdb	WAVE_MATRIX_FP_LFC_B	?	match
cdb	DLL_MATRIX_LFC_FP_A	?	match
cdb	DLL_MATRIX_FP_LFC_B	?	match
cdb	CONTAM_FP	?	match
cdb	REL_EFF_B	1	match
cdb	ABS_EFF_A	?	match
cdb	CRH_MAP	?	match

For the observations with sky on fibre B - OBJ_SKY, the inputs for drift computation (S2D_BLAZE_THAR_FP_B) and the contamination (CONTAM_FP) are not needed. Note that there can be cross-talk from the object fibre to the sky fibre of up to 1%. This contamination is not corrected for and is thus over-subtracted in the SKYSUB frames.

The LFC products are needed if the user wishes to use the LFC wavelength calibration products. In this case s/he has to set the parameter wave_cal_source to LFC. Even if the WAVE/DLL_MATRIX_THAR_FP/FP_THAR are provided, they will not be used in this case.

For a more flexible data reduction this release allows the user to provide the CRH_MAP. The user has to make sure this frame contains two images of the pixels locations affected by cosmics in two extensions. Each image must have the same X and Y sizes and the same ARCFILE of the CCD_CORR_SCIENCE frame.

9.12.2 Output for OBJ_FP

ID	PRO.CATG	type	Note
0	CCF_RESIDUALS_A	qc	residuals from CCF computation
1	DRIFT_MATRIX_B	qc	drift matrix relative to fibre B
2	S2D_BLAZE_A	cdb	S2D spectrum blaze fibre A
3	S2D_A	cdb	S2D spectrum for fibre A
4	S2D_BLAZE_B	cdb	S2D spectrum blaze fibre B
5	S2D_B	cdb	S2D spectrum for fibre B
6	S1D_A	cdb	S1D spectrum for fibre A
7	S1D_FINAL_A	cdb	Table containing all above S1D products for fibre A. The format is compatible with ESO archive standard.

[6] ? indicates an optional input

8	S1D_FLUXCAL_A	cdb	S1D flux calibrated spectrum for fibre A
9	S1D_B	cdb	S1D spectrum for fibre B
10	S1D_FINAL_B	cdb	Table containing all above S1D products for fibre B.
11	CCF_A	cdb	CCF for fibre A
12	CCD_CORR_SCIENCE	-	frame on which cosmics are dectected. [7]
13	CRH_MAP	-	(van-dokkum) cosmic detection map[8]
14	FCRH_MAP	-	post filtered cosmic detection map[9]

If the user does not provide ABS_EFF_A optional input frame the "FLUXCAL" products are not created.

The 'S2D_*' data products contain the extracted spectrum order by order or slice (depending on the mode), the associated error and quality pixel map in the extensions SCIDATA, ERRDATA, QUALDATA respectively. The 'S2D_*' products also contain the wavelength solutions and the detector pixels sizes computed in vacuum and air in the extensions WAVEDATA_VAC_BARY, WAVEDATA_AIR_BARY,DLLDATA_VA DLLDATA_AIR_BARY, respectively.

The 'S1D_*' product tables provides the wavelength solution (in vacuum and air), the merged spectrum resampled to an equidistant grid, its error and pixel quality flag.

The S1D_FINAL_B product table contains the same information as the S1D products, in Phase3 compliant data format. It also includes the computed Signal to Noise ratio (SNR).

The fibre S1D_FINAL_A product table includes the flux calibrated (and, when available, sky subtracted) object spectrum, the SNR, and information on the flux in electrons (and eventually the one of the sky spectrum) together with the associated error and pixel quality.

The CCF_A data contains the computed CCF for each order and the associated error and pixel quality for each order/slice (depending on the mode) and an additional first raw for the weighted sum of the CCF.

9.12.3 Output for OBJ_SKY

ID	PRO.CATG	type	Note
0	CCF_RESIDUALS_A	qc	residuals from CCF computation
1	S2D_SKYSUB_A	cdb	S2D spectrum sky-subtracted for fibre A
2	S2D_BLAZE_A	cdb	S2D spectrum blaze fibre A
3	S2D_A	cdb	S2D spectrum for fibre A
4	S2D_BLAZE_B	cdb	S2D spectrum blaze fibre B
5	S2D_B	cdb	S2D spectrum for fibre B
6	S1D_SKYSUB_A	cdb	S1D spectrum sky-subtracted for fibre A
7	S1D_FINAL_A	cdb	Table containing all above S1D products for fibre A. The format is compatible with ESO archive standard.
8	S1D_SKYSUB_FLUXCAL_A	cdb	S1D spectrum sky-subtracted flux calibrated for fibre A
9	S1D_A	cdb	S1D spectrum for fibre A
10	S1D_FLUXCAL_A	cdb	S1D flux calibrated spectrum for fibre A
11	S1D_B	cdb	S1D spectrum for fibre B
12	S1D_FINAL_B	cdb	Table containing all above S1D products for fibre B.
13	CCF_A	cdb	CCF for fibre A
14	CCF_B	cdb	CCF for fibre B

If the user does not provide ABS_EFF_A optional input frame the "FLUXCAL" products are not created.

[7] optional product to inspect cosmic detection obtained if extra_products_sw is set to True
[8] optional product to inspect cosmic detection obtained if extra_products_sw is set to True
[9] optional product to inspect cosmic detection obtained if extra_products_sw is set to True and –lacosmic.post-filter-x/y are $>= 1$

9.12.4 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj BIAS RON	RON[ADU]
QC EXTn ROXi ROYj MAX FLUX	max flux, raw image [ADU]
QC SATURATION CHECK	saturation [ADU] QC
QC ORDERm SNR	SNR in the order m
QC ORDERm COSMIC NB	cosmics number in the order m
QC ORDERm COSMIC NB2	Additional cosmics found in each order
QC ORDERm FLUX CORR	flux correction for the order m
QC FLUX CORR MIN	min of flux correction
QC FLUX CORR MAX	max of flux correction
QC SCIRED FLUX CORR CHECK	flux correction QC
QC DRIFT DETn SLOPE_O	Drift slope across the orders per detector
QC DRIFT DETn SLOPE_O_ERR	Drift slope error
QC DRIFT DETn SLOPE_X	Drift slope along the orders per detector
QC DRIFT DETn SLOPE_X_ERR	Drift slope error
QC DRIFT DETn CHISQ	CHI2 of the drift fit
QC DRIFT DETn REJECTED	nb of rejected pixels
QC DRIFT DETn MEAN	mean drift per detector
QC DRIFT DETn MEAN_ERR	mean drift error
QC DRIFT DETn FLUX_RATIO	flux ratio per detector
QC DRIFT DETn FLUX_RATIO_ERR	flux ratio error
QC DRIFT DETn FIRST_ORDER	first order on the detector
QC DRIFT DETn LAST_ORDER	last order on the detector
QC DRIFT CHI2 CHECK	quality check on the drift
QC BERV[10]	barycentric correction [km s^{-1}]
QC BJD	barycentric Julian date (TDB) [JD]
QC BERVMAX	barycentric max [km s^{-1}]
QC EXTn BKGR MEAN	bkgr mean in the extension [e-]
QC EXTn BKGR MIN	bkgr min in the extension [e-]
QC EXTn BKGR MAX	bkgr max in the extension [e-]
QC CCF RV	radial velocity [km s^{-1}]
QC CCF RV ERROR	error on Radial velocity [km s^{-1}]
QC CCF FWHM	CCF FWHM [km s^{-1}]
QC CCF FWHM ERROR	CCF FWHM error [km s^{-1}]
QC CCF CONTRAST	CCF contrast
QC CCF CONTRAST ERROR	CCF contrast error
QC CCF CONTINUUM	CCF continuum level [e-]
QC CCF MASK	CCF mask used
QC CCF FLUX ASYMMETRY	CCF asymmetry [km s^{-1}]
QC CCF FLUX ASYMMETRY ERROR	CCF asymmetry error [km s^{-1}]
QC CCF BIS SPAN	CCF bisector span (km s^{-1})
QC CCF BIS SPAN ERROR	CCF bisector span error [km s^{-1}]
QC SCIRED CHECK	overall science check

Please note that previous list refers to the case of `SCIENCE_FP`. In case of `SCIENCE_SKY`, there is no drift computation, nor corresponding check. Moreover in that case `S2D_BLAZE_THAR_FP` is not a necessary input.

9.12.5 Parameters

[10] Barycentric Earth Radial Velocity (BERV)

parameter name	description
`ovsc_sig_clip_method`	method for sigma clipping in OVSC, can be mean or median. [mean]
`ovsc_ksigma`	ksigma for sigma clipping in OVSC, must be between: 1.50 and 1000.00. [4.0]
`ovsc_max_iter`	maximal number of iterations in OVSC, must be between: 1 and 5000000. [10]
`wave_cal_source`	Wavelength calibration source to be used on science fiber ('THAR' or 'LFC'). [THAR]
`ksigma_cosmic`	ksigma for removing cosmics on fiber A or SKY, -1.0 - no cosmics removal. [3.5]
`rv_center`	Approximate RV in km s^{-1}. In case of default (-9999) the pipeline will use the value of the FITS keyword OCS OBJ RV. [-9999.0]
`rv_range`	Half Range for the RV table [km s^{-1}]. [20.0]
`rv_step`	Range's step for the RV table [km s^{-1}] . [0.5]
`mask_table_id`	Mask table to be used for CCF computation, defined by it's spectral type Id, e.g. 'F9', 'G2', 'G8', 'G9', 'K2', 'K6', 'M0', 'M2', 'M3', 'M4', or 'M5'. In case of default ('XX') the pipeline will read the value of the FITS keyword OCS OBJ SP TYPE and uses a correspondence table to assign the closest available mask to the spectral type given by the FITS keyword or the user. [XX] If none specified, the G2 is used. For QSO observations the CCF computation on the object spectrum is tuned off.
`extraction_method`	Method used to extract orders: 'horne' or 'simple' [horne]
`background_sw`	Background measurement activation (on/off). [on]
`flux_correction_type`	Flux correction: NONE, AUTO or <spectral_type>. Spectral types are the one listed for the parameter `mask_table_id`. In case of AUTO the behaviour is as for `mask_table_id` set to 'XX'. For QSO observations the correction is performed. [AUTO]
`drift_correction_sw`	Drift correction activation (on/off). [on]
`bias_res_removal_sw`	Flag indicating to remove or not MB residuals. [on]
`sci_bkgr_grid_size_x`	Grid size in x used to calculate the background, between: 16 and 10000. [577]
`sci_bkgr_grid_size_y`	Grid size in y used to calculate the background, between: 16 and 10000. [256]
`drift_method_fp`	Method adopted to compute drift. [flux_global_drift_global_sequential_fit]
`drift_space`	Space to compute drift (pixel/velocity). [pixel]
`drift_ksigma`	ksigma for computing drift, -1.0 - no ksigma clip. [50.0]
`sky_sub_method`	Method used to subtract the sky (pixel-by-pixel/smoothed). [pixel-by-pixel]
`sky_sub_sliding_box_size`	Sliding box size in smoothed sky subtraction. [50]
`slit_loss`	Slit loss correction in flux calibration. [-1.0]. A null or negative value calculates the correction from the seeing (FWHM.IA header keyword). A positive value indicates the correction to apply (the flux calibrated spectrum will be divided by the value specified by the parameter) instead of the one computed by the seeing. For OBJ,SKY observations, the correction is applied only to sky-subtracted products.
`cosmic_detection_sw`	LA Cosmic detection activation switch. [0]
`lacosmic.post-filter-x`	X Size of the post filtering kernel. [0]
`lacosmic.post-filter-y`	Y Size of the post filtering kernel. [0]
`lacosmic.post-filter-mode`	Post filtering mode. [dilation]
`lacosmic.sigma_lim`	LA Cosmic Poisson fluctuation threshold. [4.0]
`lacosmic.f_lim`	LA Cosmic minimum contrast. [4.0]
`lacosmic.max_iter`	LA Cosmic max number of iterations. [5]
`extra_products_sw`	Set to TRUE to create extra products to inspect LA Cosmic results. [FALSE]

9.13 espdr_compu_drift

The compu_drift recipe is not part of the reduction cascade. It can be launched by hand, e.g. with esorex.

9.13.1 Input

type	TAG	n	setting

cdb	S2D_BLAZE*	2..n	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
cdb	WAVE_MATRIX*	1	match
cdb	DLL_MATRIX*	1	match

The drift is computed on the calibration source (THAR, FP or LFC) S2D de-flatted, non de-blazed spectra. For example, to compute the drift with fibre B on FP, input files can be of type OBJ_FP, THAR_FP, FP_FP. The WAVE and DLL MATRIXes have to match the fibre for which the drift is computed, in the case of fibre B on FP it should be *MATRIX_FP_THAR_B. The first frame in the S2D list is the reference frame for the drift calculation.

9.13.2 Output

ID	PRO.CATG	type	Note
0	DRIFT_<calib_source_A_and_B>_0	qc	Drift computed for the slice 0
1	DRIFT_<calib_source_A_and_B>_1	qc	Drift computed for the slice 1

9.13.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC DRIFT REF_ARC_FILE	Reference for drift computation
QC DRIFT METHOD	Drift method used
QC DRIFT SPACE	Drift space used
QC DRIFT DETn SLOPE_O	Drift slope across the orders per detector
QC DRIFT DETn SLOPE_O_ERR	Drift slope error
QC DRIFT DETn SLOPE_X	Drift slope along the orders per detector
QC DRIFT DETn SLOPE_X_ERR	Drift slope error
QC DRIFT DETn CHISQ	CHI2 of the drift fit
QC DRIFT DETn REJECTED	nb of rejected pixels
QC DRIFT DETn MEAN	mean drift per detector
QC DRIFT DETn MEAN_ERR	mean drift error
QC DRIFT DETn FLUX_RATIO	flux ratio per detector
QC DRIFT DETn FLUX_RATIO_ERR	flux ratio error
QC DRIFT DETn FIRST_ORDER	first order on the detector
QC DRIFT DETn LAST_ORDER	last order on the detector

9.13.4 Parameters

parameter name	description
method_thar	method for drift computation for ThAr spectrum [flux_global_drift_global_sequential_fit]
method_fp	method for drift computation for FP spectrum [flux_global_drift_global_sequential_fit]
method_lfc	method for drift computation for LFC spectrum [flux_global_drift_global_sequential_fit]
space	Type of drift computation: ni the space of pixels or velocities [pixel]
ksig	ksigma for rejecting outliers o=in the drift fit [50.0]
max_flux_threshold	Maximum allowed flux [700000.0]
min_flux_threshold	Minimum allowed flux [1000.0]

9.13.5 Example

Here is the SOF example of the compu_drift recipe. The S2Ds are extracted S2D spectra of a calibration source FP on fibre A. The first frame in the S2D list is the reference frame for the drift calculation. The WAVE_MATRIX and the DLL_MATRIX have to be the wavelength solutions for the fibre for which the drift is computed, fibre A in this example.

```
/data/cal/ESPRESSO_1x1_CCD_geom_config.fits CCD_GEOM
/data/cal/ESPRESSO_SINGLEHR_1x1_master_inst_config_2018-10-14.fits MASTER_INST_CONFIG

/data/reduced/2019-06-24/r.ESPRE.2019-06-25T12:26:36.874_WAVE_MATRIX_THAR_FP_A.fits WAVE_MATRIX
/data/reduced/2019-06-24/r.ESPRE.2019-06-25T12:26:36.874_DLL_MATRIX_THAR_FP_A.fits DLL_MATRIX

/data/reduced/2019-06-24/r.ESPRE.2019-06-25T12:20:17.239_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-24/r.ESPRE.2019-06-25T12:36:03.728_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-24/r.ESPRE.2019-06-25T14:53:17.094_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-26/r.ESPRE.2019-06-26T17:02:01.813_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-27/r.ESPRE.2019-06-27T17:33:03.243_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-27/r.ESPRE.2019-06-27T18:09:53.007_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-27/r.ESPRE.2019-06-28T11:09:24.917_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-28/r.ESPRE.2019-06-28T19:12:16.051_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-28/r.ESPRE.2019-06-28T19:28:04.485_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
/data/reduced/2019-06-28/r.ESPRE.2019-06-29T11:03:26.043_S2D_BLAZE_FP_FP_A.fits S2D_BLAZE_FP_FP_A
```

And here is an example of the esorex command to launch the recipe:

```
esorex -output-dir=/data/reduced/DRIFT espdr_compu_drift drift_input_HR11_FP_FP_A.sof
```

9.14 espdr_wave_THAR_THAR

The espdr_wave_THAR_THAR recipe is not part of the reduction cascade. It can be launched by hand, e.g. with esorex.

9.14.1 Input

type	TAG	n	setting
raw	THAR_THAR	1	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match
cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match
cdb	BLAZE_A	1	match
cdb	BLAZE_B	1	match

9.14.2 Output

ID	PRO.CATG	type	Note
0	S2D_THAR_THAR_A	cdb	S2D Extracted spectrum for fibre A
1	S2D_THAR_THAR_B	cdb	S2D Extracted spectrum for fibre B
2	S2D_BLAZE_THAR_THAR_A	cdb	S2D Extracted blaze for fibre A
3	S2D_BLAZE_THAR_THAR_B	cdb	S2D Extracted blaze for fibre B

9.14.3 Quality control

NONE

9.14.4 Parameters

NONE

9.15 espdr_wave_LFC_LFC

The espdr_wave_THAR_THAR recipe is not part of the reduction cascade. It can be launched by hand, e.g. with esorex.

9.15.1 Input

type	TAG	n	setting
raw	LFC_LFC	1	any
ref	CCD_GEOM	1	match
ref	INST_CONFIG	1	match
cdb	HOT_PIXEL_MASK	1	match
cdb	BAD_PIXEL_MASK	1	match
cdb	ORDER_TABLE_A	1	match
cdb	ORDER_TABLE_B	1	match
cdb	ORDER_PROFILE_A	1	match
cdb	ORDER_PROFILE_B	1	match
cdb	FSPECTRUM_A	1	match
cdb	FSPECTRUM_B	1	match
cdb	BLAZE_A	1	match
cdb	BLAZE_B	1	match

9.15.2 Output

ID	PRO.CATG	type	Note
0	S2D_LFC_LFC_A	cdb	S2D Extracted spectrum for fibre A
1	S2D_LFC_LFC_B	cdb	S2D Extracted spectrum for fibre B
2	S2D_BLAZE_LFC_LFC_A	cdb	S2D Extracted blaze for fibre A
3	S2D_BLAZE_LFC_LFC_B	cdb	S2D Extracted blaze for fibre B

9.15.3 Quality control

This recipe computes the following QC parameters (on the image extension n, order m, and read-out region i,j):

QC key name	description
QC EXTn ROXi ROYj MAX FLUX	max flux on raw image [ADU]
QC ORDERm COSMIC NB	number of cosmics detected in order m
QC SATURATION CHECK	quality check on saturation
QC COSMIC NB	number of cosmics
QC LFC WAVE CHECK	overall quality check

9.15.4 Parameters

NONE

10 Quick start

This section describes the most immediate usage of the ESPRESSO pipeline recipes.

10.1 ESPRESSO pipeline recipes

The current ESPRESSO pipeline is based on a set of twelve stand-alone recipes involved in the data reduction cascade:

`espdr_mbias` Creates the master bias and master bias residuals frames;

`espdr_mdark` Creates the master dark & hot pixel mask;

`espdr_led_ff` Computes the mean gain per detector and detect the bad pixels;

`espdr_orderdef` Characterises the orders as imaged on the CCD;

`espdr_mflat` Creates the master flat, blaze and order profiles;

`espdr_wave_FP` Wavelength calibration with FP;

`espdr_wave_THAR` Wavelength calibration with THAR and FP;

`espdr_wave_LFC` Wavelength calibration with LFC and FP;

`espdr_cal_contam` Generates a contamination frame and checks contamination level on science fibre;

`espdr_cal_eff_ab` Computes the relative efficiency between sky and science fibres vs. wavelength;

`espdr_cal_flux` Measures the absolute efficiency curve;

`espdr_sci_red` Performs science reduction;

Other three stand-alone recipes are also provided:

`espdr_compu_drift` Measures instrumental drift on wavelength calibration spectra

`espdr_wave_THAR_THAR` S2D extraction of THAR,THAR frames

`espdr_wave_LFC_LFC` S2D extraction of LFC,LFC frames

10.2 An introduction to Reflex and EsoRex

Before being able to call pipeline recipes to process a set of data, the data must be correctly classified, and associated with the appropriate calibrations. The *Data Classification* consists of tasks such as: "What kind of data am I?", *e.g.*, BIAS, "to which group do I belong?", *e.g.*, to a particular Observation Block or observing template. *Data Association* is the process of selecting appropriate calibration data for the reduction of a set of raw science frames. Typically, a set of frames can be associated if they share a number of properties, such as instrument and detector configuration. Since all the required information is stored in the FITS headers, data association is based on a set of header keywords (called "association keywords") and the process is specific to each type of calibration. The process of data classification and association is known as *data organisation*.

An instrument pipeline consists of a set of data processing modules that can be called from different host applications, namely:

- *Reflex* is a graphical tool that helps the user to execute data reduction workflows that contain several recipes. *Reflex* takes care of grouping the different data sets, associating the calibration frames and managing the interdependencies between recipes in the calibration cascade. **Reflex is the recommended software tool for reducing your data.**
- *EsoRex* is a command line tool used to run each pipeline recipe. *EsoRex* commands can be easily scripted.
- The Paranal observatory implements automatic data management tools that trigger the execution of pipeline recipes. This aspect is not covered in this manual.

10.2.1 Using *Reflex*

Reflex is the recommended tool to reduce complete data sets that include all the calibration frames. It is an advanced, yet easy to use, tool geared towards maximum scientific return. It is based on the workflow engine *Kepler*.

Please refer to [8] for the installation procedure which also contains a detailed description of the *Reflex* application. The following provides a very brief summary of how to use Reflex.

Once installed, *Reflex* can be executed with the command:

```
user@host# esoreflex &
```

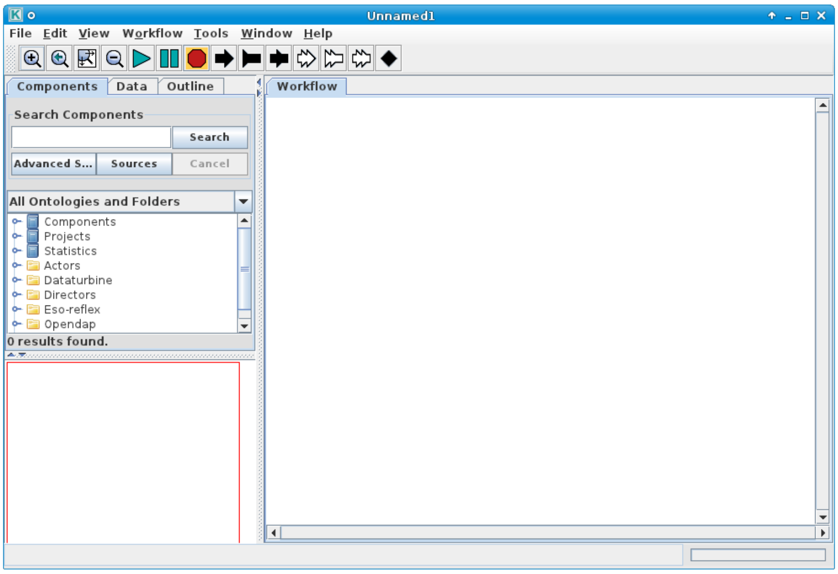

Figure 10.2.1: *Fresh Reflex canvas.*

The main concepts in *Reflex* are workflows and actors. Workflows are canvasses which show the interdependence of the pipeline recipes, allowing the user to easily obtain an overview of the reduction steps. Workflows have the advantage of requiring a small learning curve in order to get the pipeline running.

Actors are entities that perform operations. *Reflex* combines actors that correspond to pipeline recipes that perform data reduction steps, and actors useful for managing data files, such as the DataOrganizer and the FitsRouter. Each actor can be configured by right-clicking on it and selecting *Configure Actor* as shown in Figure 10.2.2. In the case of the recipe actors, the recipe parameters are part of the actor and make up the second group of parameters.

In addition to those elements, the workflow contains variables that contain the most important settings, such as the directories where data is located and will be saved.

10.3 Quick Start: Reducing The Demo Data

We describe the steps to reduce the science data provided in the ESPRESSO demo data set supplied with the `Reflex 2.11` release. By following these steps, the user should have enough information to perform a reduction of his/her own data:

1. Start the `Reflex` application:

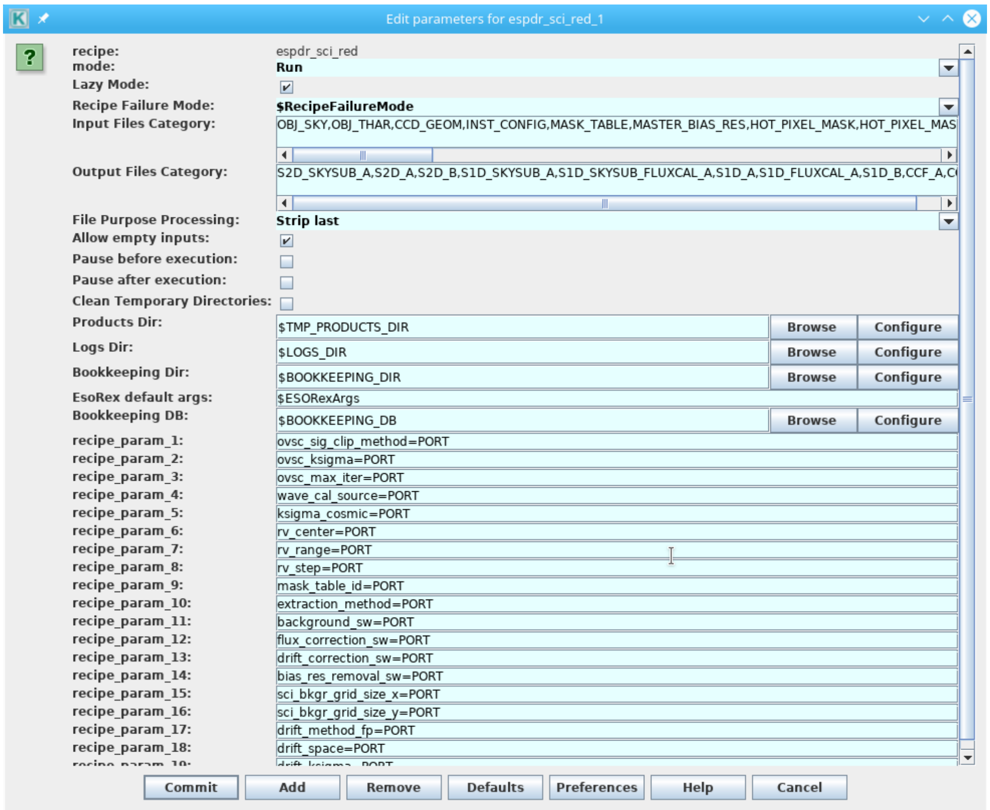

Figure 10.2.2: *Parameters of a recipe actor.*

```
esoreflex &
```

If `install_esoreflex` was used or manual installation was performed then the start command is:

```
<install_dir>/bin/esoreflex &
```

The empty `Reflex` canvas as shown in Figure 10.3.1 will appear.

2. Now open the ESPRESSO workflow by clicking on `File -> Open File`, selecting first `espdr-3.2.0` and then the file `espdr.xml` in the file browser. You will be presented with the workflow canvas shown in Figure 10.3.2. Note that the workflow will appear as a canvas in a new window.

3. To aid in the visual tracking of the reduction cascade, it is advisable to use component (or actor) highlighting. Click on `Tools -> Animate at Runtime`, enter the number of milliseconds representing the animation interval (100 ms is recommended), and click `OK`.

4. Under "Setup Directories" in the workflow canvas there are seven parameters that specify important directories (green dots). Changing the value of `ROOT_DATA_DIR` and/or `RAW_DATA_DIR` is the only necessary modification if you want to process data other than the demo data[11], since the value of this parameter specifies the working directory within which the other directories are organised. Double-click on the parameter `ROOT_DATA_DIR` and a pop-up window will appear allowing you to modify the directory string, which you may either edit directly, or use the `Browse` button to select the directory from a file browser. When you have finished, click `OK` to save your changes.

[11] If you used the install script `install_esoreflex`, then the value of the parameter `ROOT_DATA_DIR` will already be set correctly to the directory where the demo data was downloaded.

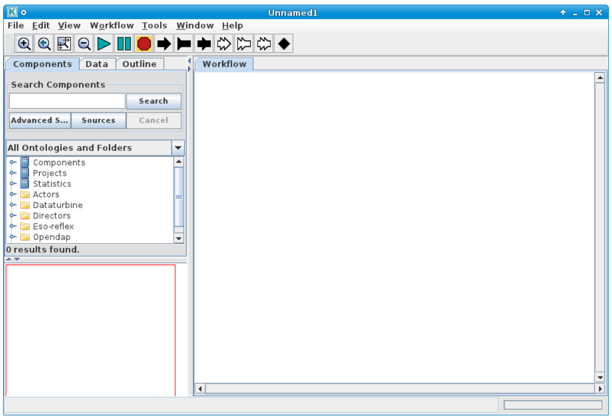

Figure 10.3.1: *The empty* `Reflex` *canvas.*

5. Click the  button to start the workflow
6. The workflow will highlight the `Data Organiser` actor which recursively scans the raw data directory (specified by the parameter `RAW_DATA_DIR` under "Setup Directories" in the workflow canvas) and constructs the DataSets. Note that the raw and static calibration data must be present either in `RAW_DATA_DIR` or in `CALIB_DATA_DIR`, otherwise DataSets may be incomplete and cannot be processed. Please also note that if the same reference file was downloaded twice to different places this creates a problem as `Reflex` cannot decide which one to use.
7. The `Data Set Chooser` actor will be highlighted next and will display a "Select Datasets" window (see Figure 10.3.3) that lists the DataSets along with the values of a selection of useful header keywords[12]. The first column consists of a set of tick boxes which allow the user to select the DataSets to be processed. By default all complete DataSets which have not yet been reduced will be selected.
8. Click the `Continue` button and watch the progress of the workflow by following the red highlighting of the actors. A window will show which DataSet is currently being processed.
9. Once the reduction of all DataSets has finished, a pop-up window called *Product Explorer* will appear, showing the datasets which have been reduced together with the list of final products. This actor allows the user to inspect the final data products, as well as to search and inspect the input data used to create any of the products of the workflow. Figure 10.3.4 shows the Product Explorer window.
10. After the workflow has finished, all the products from all the DataSets can be found in a directory under `END_PRODUCTS_DIR` with the named with the workflow start timestamp. Further subdirectories will be found with the name of each DataSet.
11. Here we have described what a user should do to reduce data different from the ones of the demo data. If a user runs the workflow using the demo data, there is no need to change "Setup Directories" (step 4).

Well done! You have successfully completed the quick start section and you should be able to use this knowledge to reduce your own data. However, there are many interesting features of `Reflex` and the ESPRESSO workflow that merit a look at the rest of this tutorial.

[12]The keywords listed can be changed by right-clicking on the `DataOrganiser` Actor, selecting `Configure Actor`, and then changing the list of keywords in the second line of the pop-up window.

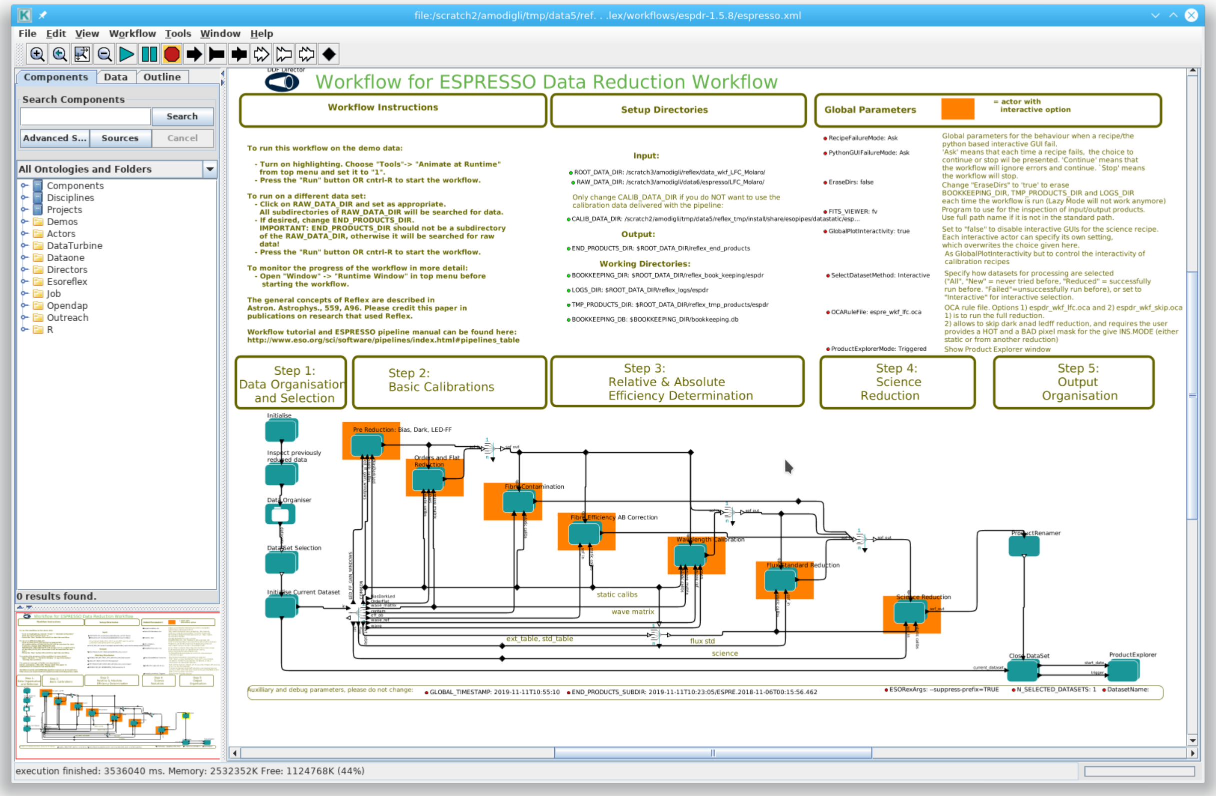

Figure 10.3.2: *ESPRESSO workflow general layout.*

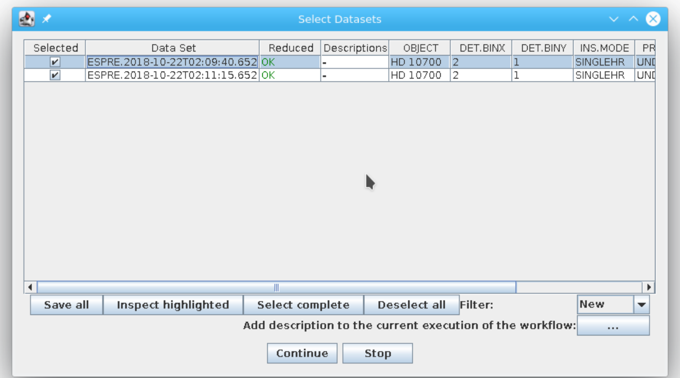

Figure 10.3.3: *The "Select Datasets" pop-up window.*

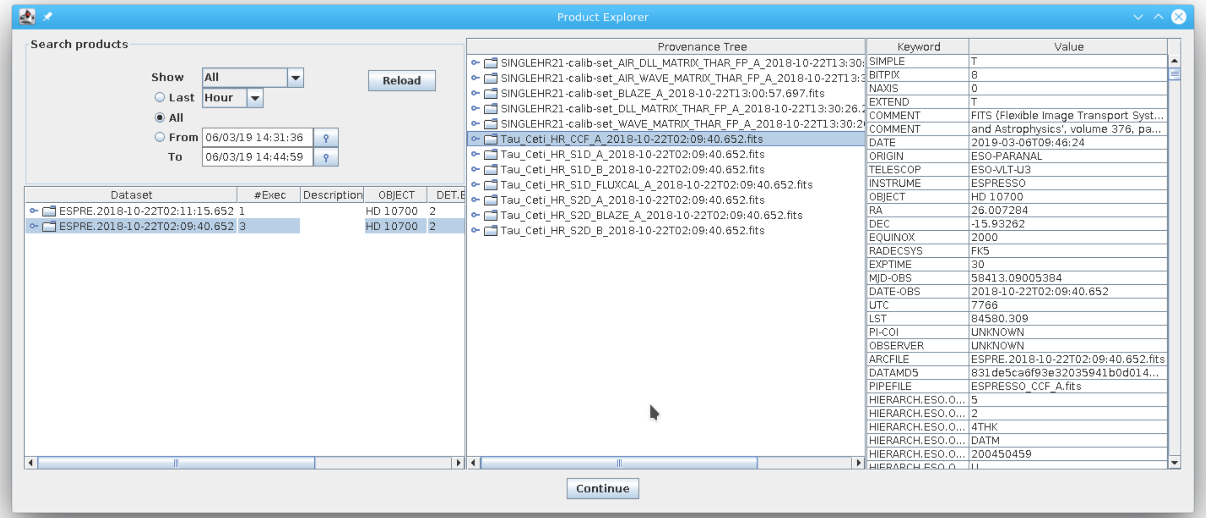

Figure 10.3.4: *The Provenance Explorer shows all datasets reduced in previous executions together with the full reduction chain for all the pipeline products.*

10.3.1 Using EsoRex

EsoRex is a command line utility for running pipeline recipes. Users may embed EsoRex into data reduction scripts. However, with *EsoRex* users must manually classify and associate the data using the information contained in the FITS header keywords (see Section 6, page 22). The user has to define the input set-of-frames (SOF) and potentially set configuration parameters.

The set-of-frames: Each pipeline recipe is executed for a set of input FITS data files. When using *EsoRex* the filenames must be listed together with their Data Organiser (DO) category [13] in an ASCII file, the *set-of-frames* (SOF), that is required when launching a recipe. A Set Of Frames (SOF) is a an ASCIII text file that lists each input frame (full path), specifying its DO category. Here is an example of SOF, valid for the *espdr_orderdef* recipe

```
/file_path/ESPRE.2018-09-02T10:24:23.827.fits ORDERDEF_A
/file_path/ESPRE.2018-09-02T10:25:30.191.fits ORDERDEF_B

/file_path/ESPRESSO_SINGLEHR_1x1_inst_config_2018-10-14.fits INST_CONFIG
/file_path/ESPRESSO_1x1_CCD_geom_config.fits CCD_GEOM

/file_path/ESPRESSO_hot_pixels.fits HOT_PIXEL_MASK
/file_path/ESPRESSO_bad_pixels.fits BAD_PIXEL_MASK
```

Note that the ESPRESSO pipeline recipes do not verify in any way the correctness of the classification tags specified by the user in the SOF. In the above example, the recipe *espdr_orderdef* will treat the frame `/file_path/ESPRE.2018-09-02T10:24:23.827.fits` as a ORDERDEF_A, the frame `/file_path/ESPRESSO_master_bias_res.fits` as a MASTER_BIAS_RES, etc., even when they do not contain this type of data. The recipe will also assume that all frames are associated correctly, *i.e.*, that they all come from the same instrument mode, and detector bin setting, and that the appropriate calibration files have been specified.

[13] The indicated *Data Organizer category* is a label assigned to any data type after it has been classified, which is then used to identify the frames listed in the *set-of-frames*

EsoRex syntax: **esorex [esorex_options] recipe_name [recipe_options] set_of_frames**

To get more information on how to customise ESOREX (see also [5]) run the command:

esorex - -help

To generate a configuration file esorex.rc in the directory $HOME/.esorex run the command:

esorex - -create-config

A list of all available recipes, each with a one-line description, can be obtained using the command:

esorex - -recipes

All recipe parameters (aliases) and their default values can be displayed by the command

esorex - -params recipe_name

To get a brief description of each parameter meaning execute the command:

esorex - -help recipe_name

To get more details about the given recipe give the command at the shell prompt:

esorex - -man-page recipe_name

Recipe configuration file: Each pipeline recipe can be controlled by specifying its configuration parameters. One way is to define them on command line. Another way is using the ESOReflex interface. Alternatively parameters can be defined through a recipe configuration file. This is an ASCII file listing parameters, their values and and their description. Recipe configuration files are normally (by default) generated in the directory `$HOME/.esorex`, and have the same name as the recipe to which they are related, with the filename extension `.rc`. For instance, the recipe *espdr_sci_red* has its *EsoRex* generated configuration file named `espdr_sci_red.rc`, and is generated with the command:

esorex - -create-config espdr_sci_red

The definition of one parameter of a recipe may look like this:

```
# --rv_center
# Approximate RV.
espdr.espdr_sci_red.rv_center=0.0
```

In this example, the parameter `espdr.espdr_sci_red.rv_center` is set to the value `0.0`. In the configuration file generated by *EsoRex*, one or more comment lines describing the parameter parameter purpose are added, and an alias that could be used as a command line option.

The recipes provided by the ESPRESSO pipeline are designed to implement a cascade of macro data reduction steps, each controlled by its own parameters. For this reason and to prevent parameter name clashes we specify as parameter prefix not only the instrument name but also the name of the recipe they refer to. Shorter parameter aliases are made available for use on the command line.

The command

esorex - -create-config recipe_name

generates a default configuration file **recipe_name.rc** in the directory **$HOME/.esorex**[14].

A recipe configuration file different from the default one can be specified on the command line:

esorex - -recipe-config=my_alternative_recipe_config

The reader can find a complete description of all recipe parameters in section 9. Their purpose is described in Section 11.

For a configuration file to be used as configuration file, the file should be explicitly defined with the syntax above.

Recipe execution: A recipe can be run by specifying its name to *EsoRex*, together with the name of a set-of-frames. For instance, the following command line would be used to run the recipe *espdr_sci_red* for processing the files specified in the set-of-frames `espdr_sci_red.sof`:

esorex espdr_sci_red espdr_sci_red.sof

The recipe parameters can be modified either by editing directly the used configuration file, or by specifying new parameter values on the command line using the command line options defined for this purpose. Such command line options should be inserted after the recipe name and before the SOF name, and they will supersede the system defaults and/or the configuration file settings. For instance, to set the *espdr_sci_red* recipe *rv_center* parameter to `3.0`, the following should be typed:

esorex espdr_sci_red - -rv_center=3.0 espdr_sci_red.sof

For more information on *EsoRex*, see `www.eso.org/cpl/esorex.html`.

[14] If a number of recipe parameters are specified on the command line, the given values will be used in the created configuration file.

11 Algorithms and recipe details

In this section we describe the main algorithms implemented in the ESPRESSO pipeline recipes. Relevant data reduction parameters are typed in **bold** face. The full name of the parameter **bias_ksigma** of the recipe `espdr_mbias` is actually `espdr.espdr_mbias.bias_ksigma`. For convenience we omit the common prefix `espdr.espdr_mbias.` for parameter description, the common prefix `espdr_` for the recipe description as well as the step prefix name for the algorithm description.

11.1 Algorithms

11.1.1 Master frame combination using a kappa-sigma-clipped mean/median

A kappa-sigma-clipped mean or median may be used on a per-pixel basis to combine a set of frames into a master frame, thereby excluding outlier pixel values, i.e. the cosmics. In this technique, the median gives more accurate results. The user may set the method (**sig_clip_method**): mean or median, and the kappa value (**ksigma**) controlling thresholds used to clip outliers. This algorithm is used in all recipes that do frame stacking: mbias, mdark, led-ff and mflat.

11.1.2 Order definition

Orders and slices are identified on ORDERDEF frames through a clumping algorithm that groups together neighbouring pixels whose values exceed a given flux threshold. The algorithm acts recursively on all pixels meeting the flux criterion, within a given distance, until all pixels have been classified either as belonging to a clump or as background.

Clumps are then screened against several criteria to establish whether they represent valid orders/slices: minimum length in main dispersion direction and minimum width in cross-dispersion direction. The clumps that do not meet the criteria are discarded.

After that, orders and slices are assigned physical interference orders. This numbering is established by using input parameters specifying the number of slices per order, the physical number of the first detected order, the number of valid orders to be found, and the typical inter-slice distance on the detector.

Finally, order shapes and positions in cross-dispersion direction are fitted at several locations along the main dispersion using an appropriate analytical model (Gaussian or sum of Gaussians). Note that the Gaussian or multi-Gaussian fit is only needed to approximately measure order center. Then, a low-order polynomial is fitted to the Y position of each order as a function of X, where X is the main dispersion direction and Y is the cross-dispersion direction. Only the center of the order in Y is relevant at this stage, not their shape in the Y direction. The only purpose of these polynomials is to provide order center vs. X position to 0.5 pixel precision in order to define the window of extraction that will be used by the subsequent spectrum extraction procedure.

11.1.3 Background subtraction

The strategy to measure the diffuse background light on the detector is the following:

- Divide the detector into a grid of small regions (100x100 pixels in size)
- In each region, excluding the pixels belonging to spectral orders, build an histogram of (background) pixel values.
- Estimate the mode of the pixel value distribution by fitting a second-order polynomial to the histogram bins closest to the peak of the distribution. The mode of the distribution is taken as the best approximation of the local background level.
- Interpolate the background levels measured in each region over the whole detector using cubic splines. These cubic splines have low order or few nodes, i.e., are smoothly varying over the detector.

A background map is then generated and subtracted from the raw frame.

11.1.4 Spectrum Extraction

As of today, all major fiber-fed spectrographs designed for precise radial velocity work produce relatively symmetric, round slit images on the detector (e.g. ELODIE, CORALIE, HARPS, SOPHIE). This results from the use of circular fibres and simple light injection with no image or pupil slicing at spectrograph entrance. Moreover, the curvature of echelle orders on the detector is generally small. The dispersion direction is constant and along the lines, or, in other words, the slit tilt is zero accross the wavelength range. In this case, spectrum extraction can be made by simply summing the flux in cross-dispersion direction, parallel to the CCD rows/columns. The resulting S2D spectrum is sampled on the grid of extracted pixels. Note that this extraction process has the advantage of avoiding any resampling procedure.

In the case of ESPRESSO, the situation is more complex since pupil slicing and anamorphic magnification are used to achieve high spectral resolution while using a grating with the same size as that of HARPS. The slit image on the detector is made of two ellipsoidal slices aligned in cross-dispersion direction. Depending on the exact geometry of the slit images, image quality and desired spectral resolution, it may or may not be possible to extract the spectrum by summing along the CCD rows/columns in cross-dispersion direction. ESPRESSO was designed such that the Anamorphic Pupil Slicer Unit (APSU) can be optimised to obtain a slit image tilt close to zero and ensure a precise alignement of the two slices along CCD rows/columns. The residual tilts are only a few deg and slice misalignments never exceed 2-3 microns anywhere on the detector, i.e. much less than the size of a pixel (10 microns). Given the minimum sampling of 2 pixels per FWHM in the ultra-high resolution mode, this means that a simple extraction procedure parallel to CCD rows/columns will decrease the spectral resolution by only a few percent in the worst case, i.e. at CCD edges. We consider this as fully acceptable given the simplicity and advantages of the extraction along CCD rows/columns (e.g. no resampling).

For the modes with the two slices of an order well separated on the CCD (SINGLEHR11, SINGLEHR21 and SINGLEUHR), each slice is extracted separately and treated as if it was an independent interference order. This results in each physical order being represented twice in the S2D extracted spectrum.

Two spectrum extraction functions are provided in the ESPRESSO DRL: simple summation, and optimal extraction following the well-known Horne method (Horne 1986, PASP 98, 609). We investigated refinements introduced by e.g. Marsh (1989) and Donati et al. (1997), but they are generally not relevant in the case of ESPRESSO where the slit tilt on the CCD is negligible. Nevertheless, the Horne optimal extraction algorithm has been modified significantly and optimised for ESPRESSO in the following ways:

- Order profiles in cross-dispersion direction are obtained directly from spectral flat-field exposures at high SNR, which serve as empirical models. Such exposures are part of the standard calibration sets and must be obtained within 24 hours of science observations. We make use here of the high stability of ESPRESSO which guarantees that the position and shape of spectral orders remain essentially constant between calibrations and science exposures.
- If the number of hot and bad pixels does not exceed a significant fraction (0.25) of the pixels in the extraction window, hot and bad pixels are corrected by interpolating the order profile across them before extracting the spectra. In this way the knowledge and smoothness of the order profile can be used to optimally correct for these CCD defects.
- For exposures taken with simultaneous reference on fiber B (ThAr/LFC/FP), a *contamination frame* is used as additional empirical model in the optimal extraction. This means that the spectrum is modelled as the sum of a scaled order profile and a scaled contamination profile. The obtained science flux is therefore corrected from direct ThAr/comb/FP contamination. As for the *usual* optimal extraction, the model is linear in both model parameters (science and contamination fluxes), and the solution is thus obtained by analytically solving the associated linear least-squares problem.

11.1.5 Flat-Fielding

The extraction procedure described above delivers a spectrum that is still affected by variable pixel-to-pixel sensitivity along the main dispersion direction. To correct for that, the spectrum is divided by the normalised extracted flat-field, obtained from co-added FLAT frames and extracted in exactly the same way as the science spectrum.

We note that this flat-fielding procedure will remove not only pixel-to-pixel sensitivity variations, but also fringing effects, at least when the science spectrum is broadly similar to the one of the flat-field lamp. This is the case for all stellar spectra at moderate to high SNR. In the case of a very low SNR spectrum dominated by sky emission lines, the correction of fringing may in principle be less accurate. However, fringing by sky lines is expected to be a very small effect at the high spectral resolution of ESPRESSO. The calibration images of ESPRESSO reveal no fringing at all.

11.1.6 Wavelength Calibration

There are two methods used for the wavelength calibration, according to the calibration source used: ThAr lamp together with Fabry-Perot, or Laser Frequency Comb.

ThAr and Fabry-Perot Lamps

First the FP peaks are identified on the frame with both fibres illuminated with the Fabry-Perot lamp. Then the tables of these peaks are passed to the reduction of the frames in which one of the fibres is illuminated with the ThAr lamp, while the other fibre has the Fabry-Perot light for the drift calculation. Combining the detected FP peaks and the carefully chosen ThAr lines with precise wavelengths assigned, allows to compute the precise wavelength solution per extracted order. The wavelength solution fit is done on the detected Fabry-Perot peaks, which are anchored to the precise wavelength through the known ThAr lines.

ThAr spectra are characterised by a high density of emission lines with many blends and very different intensities. Identifying and properly fitting all these lines is a challenging task, especially when aiming at a repeatability of better than 1 m/s in the wavelength calibration. Variable line intensities make it necessary to use a static ThAr table to initiate the wavelength calibration process. This table indicates the approximate position of lines. Each ESPRESSO mode has its own ThAr lines table.

ThAr lamps evolve with time, and the ageing process has several consequences on their spectra. Both line intensities and wavelengths are affected. To mitigate this, a line-by-line comparison with the static ThAr table is performed to identify and reject individual lines potentially affected by fitting instabilities.

Laser Frequency Comb

Giving the previously computed wavelength solution, the Laser Frequency Comb allows to achieve a more stable and precise solution. No static table is needed to initiate the wavelength calibration process, since emission LFC lines can be searched and identified automatically thanks to the absolute accuracy of the comb pattern. Then, the wavelength of each peak is determined. To do this, the DRL first computes a list of theoretical LFC wavelengths based on the known repetition rate and anchor frequency of the LFC. Then, these wavelengths are assigned to individual peaks by comparison with a previously-obtained ThAr wavelength calibration, which provides a suitable first guess for the wavelength of the LFC lines. Since the LFC does not cover the whole spectral range, the orders with insufficient number of detected LFC lines (less than the limit fixed in the instrument configuration) have to take the solution from the ThAr method. Also, the orders, for which the LFC solution has the RMS or the CHI2 higher than limits fixed in the instrument configuration, have the ThAr wavelength solution assigned. The FITS header keywords indicate for each order where the wavelength solution is coming from. The LC solution is provided for the wavelengths from 5020 A to 7344 A, i.e. orders 81 to 158 for SINGLE modes or 41 to 78 for MULTI modes.

11.1.7 Instrumental Drift Measurement

The simultaneous reference fibre of ESPRESSO is used either for sky subtraction or instrumental drift correction. Instrumental drift is measured using wavelength calibration sources, either the ThAr lamp, the LFC or a stabilised Fabry-Perot etalon (FP). Drift is measured by comparing two similar S2D spectra on the simultaneous reference fibre: one that is acquired at the time of the wavelength calibration (with both fibre A and fibre B illuminated by a calibrator), and one that is acquired simultaneously with the science observation. The algorithm used by the pipeline to compute the drift is based on the method described in Connes (1985), ApSS 110, 211, and Bouchy et al. (2001), A&A 374, 733. The drift is simply obtained from a first-order Taylor expansion of the spectrum at each individual pixel, i.e. the measured flux difference between the two spectra at pixel X is converted into a position difference using the measured spectrum derivative at pixel X. This obviously assumes that the spectrum derivative is constant over a length X that is the shift between the two spectra. Therefore, this method only works for drifts that are small relative to the FWHM of the spectral lines, i.e. typically smaller than one pixel. The pixel size in ESPRESSO is equal to 500 m/s, while the expected instrumental drifts are smaller than 1 m/s over 24 hours. The method described here is thus applicable for corrections in between daily wavelength calibrations. It has the significant advantage that no modelling of the spectrum is needed to measure the drift, since each individual pixel independently contributes to the total drift. Note that pixels are weighted according to their Doppler information content, which is proportional to the square of the spectrum derivative.

This algorithm has been successfully used on HARPS to measure drifts of ThAr spectra. For ESPRESSO a few improvements were made to the method to adapt it to LFC/FP spectra and make it more robust in general. The following aspects are addressed:

- Flux normalisation: the method only works if the two spectra are normalised to the same flux level. Thus a global flux normalisation on the S2D spectrum is performed.

- Local vs. global drift measurement: instrumental drifts may vary across the spectral format, i.e. a single, global value for the drift may be too coarse an approximation. It is possible to specify to the algorithm whether the drift computation and correction should be global, chip-by-chip or order-by-order.

With the algorithm described here, the instrumental drifts can be reliably measured to a precision of 1-2 cm/s globally (if not limited by photon noise).

11.1.8 Sky Subtraction

Sky subtraction in ESPRESSO can be made by observing in simultaneous sky mode and subtracting the sky spectrum obtained on fibre B from the science spectrum on fibre A. In practice, this process involves several steps:

- Measure the relative throughput of fibre B with respect to fibre A as a function of wavelength (in S2D format) by acquiring EFF_AB exposures, which are obtained by pointing the telescope at daylight sky. Note that the sky fills the entire fibre aperture uniformly, so that pointing/guiding errors or telescope pointing jitter have no effect on the amount of sky flux entering the spectrograph.
- Scale the S2D simultaneous sky spectrum using the previously-obtained relative efficiency curve.
- Rebin the scaled S2D sky spectrum to the same wavelength scale as fibre A. The resampling is done by spline interpolation of the cumulative flux vs. extracted pixel, ensuring flux conservation. Note that the wavelength scales of fibres A and B are extremely similar, so that the resampling will essentially consist of a small shift of the spectrum that is slowly varying along spectral orders.
- Subtract the scaled and rebinned S2D simultaneous sky spectrum from the S2D science spectrum.

For ESPRESSO two sky subtraction methods are available: *pixel-by-pixel* and *smoothed*. The first method subtracts the sky spectrum on a pixel-by-pixel basis. The second one performs first a sliding average of the sky spectrum and then subtracts if from the science spectrum. For very low S/N regime, readout limited or close to it, it is recommended to use the option smooth, thus reducing the readout noise contribution from the step of sky subtraction.

The precision of sky subtraction with ESPRESSO is essentially photon- and readout-noise limited, provided the sky fibre *sees* the same sky as the science fibre.

11.1.9 Barycentric Correction

In the context of high-resolution astronomical spectroscopy and high-precision radial velocity measurements, it is necessary to compute to high accuracy the projection of the velocity vector of an Earth-bound observer along the line of sight to an astronomical target, at the time of the observation, as measured within the ICRS reference frame (centred on and at rest with respect to the Solar System barycenter). The computation should take into account both the Earth's orbital motion around the Sun and the observer's motion due to Earth rotation. For ESPRESSO a new code for this computation was developed, based on recent Solar System ephemerides produced at Institut de Mécanique Céleste et de Calcul des Ephémérides (IMCCE) in Paris.

The algorithm:

- Computes the observer's projected velocity in the direction of the target in the ICRS reference frame at the time of the observation with the accuracy of at least 1 cm/s.
- Computes the barycentric time of light arrival, i.e. the time at which the light from the target reaches the Solar System barycenter given its detection by the Earth-bound observer at the time of the observation with the accuracy of at least 1s.
- Computes an upper bound to the maximum value of the barycentric correction over one year for any given target.

Input parameters:

- Right ascension of target in decimal hours, ICRS system, epoch 2000.0

- Declination of target in decimal degrees, ICRS system, epoch 2000.0
- Proper motion of target in right ascension (mu_alpha cos delta) in arcsec/year, ICRS system
- Proper motion of target in declination in arcsec/year, ICRS system
- Date of observation in UTC (year, month, day, decimal hour)
- Longitude of observer in decimal degrees
- Latitude of observer in decimal degrees
- Altitude of observer above sea level in km

Output parameters:

- Velocity component of observer projected onto target line of sight in km/s, ICRS system (abridged as BERV for Barycentric Earth Radial Velocity)
- Total velocity of observer in km/s, ICRS system
- Barycentric time of light arrival as TDB Julian date
- Yearly maximum value of BERV (upper bound)

Step-by-step description:

- Compute Julian date from given date and time of observation
- Compute ICRS target coordinates at the time of observation (i.e. take proper motion into account)
- Compute observer's velocity vector with respect to geocenter at the time of observation, ICRS system (correct for precession and nutation)
- Compute Earth's orbital velocity vector at the time of observation, ICRS system
- Combine rotational and orbital motion into total velocity vector
- Project velocity vector onto target line of sight to obtain BERV
- Compute barycentric time of light arrival
- Compute yearly upper bound to BERV

11.1.10 Order Rebinning and Merging

The rebinning and merging process to create a S1D spectrum from the S2D spectrum is as follows:

- Define a uniform wavelength grid with a constant wavelength step that is close to the average pixel size in wavelength units.
- For each spectral order, build the cumulative flux distribution vs. extracted pixel.
- Interpolate the cumulative flux function onto the uniform wavelength grid using cubic splines.
- Build the resampled spectrum by differentiating the interpolated cumulative flux.
- Merge resampled spectral orders, computing the weighted average of rebinned pixels where spectral orders overlap.

This technique is chosen because it conserves the integrated flux within any two wavelengths of the original pixel grid. Moreover, cubic splines are well suited to ESPRESSO spectra because of the well-sampled PSF of the instrument (4 pixels FWHM in singleHR mode), which ensures that the spectrum derivatives can be numerically estimated in a reliable way.

11.1.11 Radial Velocity Computation

The ESPRESSO DRL implements a cross-correlation module that computes the cross-correlation function (CCF) of a S2D spectrum with respect to a binary template (mask) of a given spectral type. The radial velocity (RV) is then obtained from a Gaussian fit to the CCF. This technique has been successfully used on the ELODIE, CORALIE, HARPS, SOPHIE and HARPS-N spectrographs (see Baranne et al. 1996, A&AS 119, 373, and Pepe et al. 2002, A&A 388, 632). One of its main advantages is that CCFs can be computed in an automatic way using a line mask. Line masks are simply lists of central wavelengths and depths of spectral lines, and can be created for various spectral types. We note here that the CCFs of slowly-rotating stars are extremely well approximated by Gaussian profiles with a flat continuum. However, the particular fitting function does not matter much; the crucial aspect is to fit the CCFs of a given star with always the same function to avoid systematic effects on the derived radial velocities. As such, it is fundamental to reduce all the spectra with the same mask and CCF parameters, as comparison between reductions with different parameters will be affected by systematics.

The main steps of the algorithm are:

- Compare the global flux distribution in the S2D spectrum to a static flux template that approximately corresponds to the spectral type of the star. The S2D flux is scaled accordingly to match the flux distribution of the template. In this way, spectra of any given star are always brought to the same flux distribution, which ensures that variable atmospheric conditions will not induce systematic effects in the CCF computation.
- Shift the wavelength scale of the S2D spectrum to the Solar System barycenter using the barycentric correction.
- Define a uniform radial velocity grid that is approximately centered on the radial velocity of the star.
- For a given RV value in the grid, shift the line mask by the corresponding Doppler shift, project the line mask onto the S2D spectrum using a specified line width (about one pixel), and sum the S2D flux that goes through the so-defined mask *holes*. The flux from partial pixels is computed via simple linear interpolation. The sum is actually a weighted sum, using line depths as weights to optimally extract the Doppler information. During this process, the S2D spectrum is locally blaze-corrected to remove any continuum slope around spectral lines. This produces one point of the CCF.
- Loop over all RV values in the grid.
- Fit a Gaussian profile to the CCF to derive RV, FWHM and contrast.

Note that, by construction, CCFs are simply co-added spectral lines in velocity space, weighted by their depth and continuum flux and corrected of the transmission of the grating (blaze function). As such they can be considered as a *master* spectral line for the star.

11.1.12 Removal of cosmic rays

.

Cosmic rays are identified via a ksigma rejection when extracting the one-dimensional spectrum of one order. In each recipe that does it, the clipping is regulated by a dedicated parameter. In the science recipe, `espr_sci_red` the clipping is regulated by the parameter **ksigma_cosmic**. A value of -1 turns off the ksigma clipping.

The science recipe has also an additional algorithm to detect cosmic rays, that exploits LACosmic (van Dokkum 2001, PASP, 113, 1420). The algorithm is applied to the so-called detector cleaned frames, i.e. the raw science where the overscan regions have been trimmed and their contribution subtracted. Cosmics identified this way are masked during the ectraction of the one-dimensional spectra.

Both algorithms can be run simultaneously.

The LA Cosmic algorithm is regulated by the following paramters: **cosmic_detection_sw**.

- **cosmic_detection_sw**. It turns LA Cosmic on (1) or off (0). By default, its value is taken by the input instrument configuration table. The algorithm is turned on for SKY-mode and it is turned off for FP mode observations.
- **lacosmic.post-filter-mode** : dilage/dilute. It specifies whether cosmic ray mask has to be dilated (i.e. expanded) or diluted (i.e. shrinked). It takes effect only if the post-filter-x/y values are positive.
- **lacosmic.post-filter-x** : X Size of the post filtering kernel. The mask is dilated or diluted by this amount along x direction. Defaut: 0.

- **lacosmic.post-filter-y** : Y Size of the post filtering kernel. The mask is dilated or diluted by this amount along y direction. Default: 0.
- **lacosmic.sigma_lim** : LA Cosmic Poisson fluctuation threshold. It is the minimum value of the fluctuation image, obtained by dividing the Laplacian image (a second order derivative of the original image along x and y) and the noise model. High values of **sigma_lim** find the more intense cosmics and have less risk to detect false positives. Low values are more efficient in finding also faint cosmics, but have higher risk to detect false positives.

  The default values are taken by the input instrument configuration table: 5 (SHR-SKY, 2x1), 7 (SHR-FP, 2x1), 7 (SHR 4x2), 8 (SUHR 1x1), 10 (MHR 4x2), 8 (MHR 8x6).
- **lacosmic.f_lim** : LA Cosmic minimum contrast between the Laplacian image (see above) and the fine-structure image (created from the original image by a combination of median filters). High values of **f_lim** find the more intense cosmics and have less risk to detect false positives. Low values are more efficient in finding also faint cosmics, but have higher risk to detect false positives.

  The default values are taken by the input instrument configuration table: 5 (SHR-SKY, 2x1), 5 (SHR-FP, 2x1), 5 (SHR 4x2), 8 (SUHR 1x1), 4 (MHR 4x2), 5 (MHR 8x6).
- **lacosmic.max_iter** : LA Cosmic max number of iterations. [5]
- **extra_products_sw** : Set to TRUE to create extra products to inspect LA Cosmic results. [FALSE]

11.1.13 Error and Bad Pixel Propagation

S2D Spectra

All significant error sources are propagated through the pipeline using the standard error propagation formulae. We list below the noise sources that are taken into account in the science reduction cascade producing the S2D spectra:

- Detector readout noise
- Detector dark current noise
- Total photon noise on science fiber (science target + diffuse background)
- Flat-fielding noise
- Sky noise if applicable (simultaneous sky mode)

Error propagation for S2D spectra is relatively straightforward since there is no resampling step in the process. Extracted pixels remain independent of each other.

If not too many, bad pixels and hot pixels will be ignored during the spectrum extraction process. The empirical, high-SNR extraction profile will be interpolated across those pixels, so that the extracted science spectrum is effectively corrected. The presence of bad/hot pixels will be reported in the quality maps associated to science products; they are therefore traceable throughout the reduction chain.

S1D Spectra

The resampling process that is necessary to generate S1D spectra from S2D spectra inevitably introduces correlations between adjacent rebinned pixels in the S1D spectra. To calculate the errors on the S1D spectra one propagates errors in the usual way to the S1D spectrum, while including in the error computation an additional array containing the correlation factor of each rebinned pixel. This factor is the quadratic sum of the relative contributions of the original pixels to the rebinned pixel, considering that a maximum of two original pixels contribute to the rebinned pixel. In this way we believe all the information on noise properties is conveyed into the S1D spectrum. We note however that the existence of correlations makes the use of the S1D spectrum non-trivial for science purposes if a rigorous treatment of the noise is required. It is up to the user to understand how to deal with this issue, e.g. when performing least-squares fitting of models to the S1D spectrum.

In general, the use of the S2D spectrum is recommended whenever possible to avoid correlations between adjacent data points, which are unavoidable as soon as some resampling is performed.

Cross-Correlation Functions

Uncertainties on the CCF data points are obtained by propagation of S2D error maps through the cross-correlation stage, which is a simple additive process (fluxes from many spectral lines are co-added). Finally, an estimate of the radial velocity uncertainty is obtained by converting CCF flux errors into RV errors using the measured CCF derivative, as described in Boisse et al. (2010), A&A 523, 88 (appendix).

11.2 Recipes

An overview of the data reduction cascade is given in Section 8.

11.2.1 mbias

The recipe generates a master bias frame from a set of minimum five raw bias frames. All the input raw frames are stacked and a sigma-clipping algorithm is applied in order to remove the cosmics. Then, the overscan is removed from the master bias image and the rest is save as a master bias residuals. These residuals are applied in several recipes in the reduction chain. The BIAS RON and mean and the overscan RON are computed on the master bias frame and saved in the FITS header as QC KWs [15]. The mean and stdev are computed on the master bias residuals frame and saved in the FITS header as QC KWs.

The BIAS subtraction on the raw images is done by subtracting the contribution of the the overcan/prescan regions of the raw images themseves, not by subtracting the master bias frame previously computed. This allows the subtraction of the BIAS level as a function of the detector line. The residuals after the subtraction correspond to the variations within each line (i.e., as a function of column). These are subtracted from the spectra under the assumption that they are stable over the timescale of one day.

11.2.2 mdark

The recipe creates a master dark frame from a set of minimum five dark frames corrected for the bias. All the input raw frames are stacked and a sigma-clipping algorithm is applied to remove the cosmics. The cosmics rates by extension (blue and red) are saved in the FITS header as QC KWs. The hot pixels are detected via the sigma-clipping algorithm and saved in the hot_pixels frame. Their number per detector output is saved in the FITS header as QC KW. The mean dark current is computed for each detector output and saved in the FITS header as QC KW.

11.2.3 led_ff

The recipe detects the bad pixels and computes the conversion factor for each detector output. For this at least two sets with different exposure times of minimum five frames each are needed. The frames are the images of the detector fully illuminated by a LED. The recipe first removes the bias and the dark current and then stacks the frames with the same exposure time. The cosmics are removed via the sigma-clipping algorithm. The bad pixels are detected via checking of the linearity of their behaviour with respect to the exposure time. Their number is saved in the FITS header as QC KW. The conversion factor is computed within a window defined for each detector output and saved in the FITS header as QC KW. The raw frames are checked against the saturation and the corresponding QC KW is set. The same is done for the minimal flux. The computed conversion factor is checked against the theoretical value taken from the FITS header of the raw frame.

11.2.4 orderdef

The recipe detects the orders position in the image. The input frames are two flat images: one for fibre A and one for fibre B. Each input raw frame is checked for the saturation and cleaned from the CCD signature (bias, dark current, gain). A FITS keyword QC CHECK is added and set to 1 in case of correct and 0 for bad quality results. Than the orders detection is performed:

- For each pixel, decide if it belongs to an order or to the background.

[15] The RON is computed on each detector read out region as the product $stddev \times \sqrt{frames\_nb}$ where $frames\_nb$ is the number of input frames, and $stddev$ is the standard deviation of the master bias overscan corrected values in the given region, determined using the Median Absolute Deviation and after (kappa sigma) removal of the outliers.

- Run the clumps detection algorithm, which marks the pixels belonging to the same clump.
- Detect which of the clumps are valid orders.
- Attribute interference / physical order numbers.
- Fit the valid orders position.
- Mark the orders position on the ORDER_TABLE products.
- Save the coefficients of the fit in the 2nd extension of the ORDER_TABLE products.

11.2.5 mflat

The recipe generates the extracted flat-field, blaze function and the order profile. Recipe steps:

- Check the raw frame for saturation.
- Remove the detector signatures (correct for overscan level).
- Measure and subtract inter-order background.
- Perform optimal extraction of orders for both fibres.
- Compute the blaze function.
- Correct the blaze function for spectral energy distribution of the source and instrumental efficiency by fitting the flux distribution at blaze peak.
- Save the QC KWs in the FITS header of the products: max flux per detector output, background min, max and mean per extension (blue and red), number of removed cosmics per order, SNR per order, RMS of the flat per order, QC CHECKS
- Save the ORDER_PROFILE, the FLAT and the BLAZE for each fibre

11.2.6 cal_contam

The recipe generates a contamination frame and checks the contamination level on science fibre. Recipe steps:

- Check the raw frame for saturation.
- Remove the detector signatures (correct for overscan level).
- Measure and subtract inter-order background.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field.
- Measure the maximum contamination level in extracted spectrum and compare it with the specified threshold.
- Save the contamination frame with QC KWs.

11.2.7 cal_eff_ab

The recipe computes the relative efficiency between sky and science fibres vs. wavelength. Recipe steps:

- Check the raw frame for saturation.
- Remove the detector signatures.
- Measure and subtract inter-order background.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field.
- Divide reference spectrum by science spectrum and fit a low-order polynomial across each order.
- Save the relative efficiency frame with QC KWs.

11.2.8 wave_FP

Recipe detects the Fabry-Perot peaks. Recipe steps:

- Check the raw frame for saturation.
- Remove the detector signatures.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field and the blaze.
- Search for the FP lines.
- Fit the FP lines.
- Check if there are no lines missing.
- Save the FP lines table and S2D, S2D_BLAZE products with QC KWs.

11.2.9 wave_THAR

Recipe performs the wavelength calibration using the ThAr lamp and Fabry-Perot spectra. Recipe steps:

- Remove the detector signatures.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field and the blaze.
- Fit the ThAr lines provided in a static table.
- Measure drift and apply it to the FP lines table (product of the wave_FP recipe).
- Check if there are no FP lines missing.
- Compute the wavelength for all the FP lines, using ThAr lines.
- Fit D lambda.
- Fit the wavelength solution.
- Fit the ThAr raw lines.
- Save the products with QC KWs.

11.2.10 wave_LFC

Recipe performs the wavelength calibration using the Laser Frequency Comb spectrum. Recipe steps:

- Remove the detector signatures.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field and the blaze.
- Search for the LFC lines.
- Fit the LFC lines.
- Assign wavelengths to the LFC lines.
- Fit the wavelength solution.
- Complete the wavelength solution with the ThAr one for orders with no LFC flux.
- Save the products with QC KWs.

11.2.11 cal_flux

The recipe measures the absolute efficiency curve. Recipe steps:

- Check the raw frame for saturation.
- Remove the detector signatures.
- Measure and subtract inter-order background.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field.
- Assign wavelength solution to the spectrum.
- Rebin and merge the orders.
- Convert S1D spectrum from detected photoelectrons to physical units (erg/s/cm2/A).
- Measure and retrieve the flux of the spectrophotometric standard star, and compute absolute efficiency at the reference wavelengths.
- Interpolate efficiency measurements onto the S1D wavelength scale using cubic splines.
- Save the absolute efficiency frame with QC KWs.

11.2.12 sci_red

The recipe performs the science reduction. Recipe steps:

- Check the raw frame for saturation.
- Remove the detector signatures.
- Identify cosmic rays.
- Measure and subtract inter-order background.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field.
- Assign wavelength solution to the spectrum.
- In simultaneous sky mode:
  - Scale and rebin sky spectrum.
  - Subtract sky.
- In simultaneous reference mode:
  - Measure instrumental drift.
  - Correct instrumental drift.
- Compute barycentric correction using FITS header information.
- Correct wavelength solution from BERV, shifting it to the barycenter of the SS.
- Rebin and merge orders.
- Convert S1D flux into physical units.
- Flux calibrate S1D spectrum (with and without sky subtraction) using absolute efficiency curve.
- Measure fibre centring on integrated guiding image and perform quality control based on specified tolerances.
- Measure flux-weighted mid-exposure time on exposure-meter data and perform quality control.
- Filter residual cosmic hits from S2D spectrum.
- Correct flux distribution in S2D spectrum.
- Compute the radial velocity.
- Compute the CCF bisector.
- Save the S2D, S1D and CCF products with QC KWs.

11.2.13 compu_drift

Recipe computes the drift for a sequence of S2D calibration source spectra. The first S2D in the list is taken as reference. Recipe steps:

- Load S2D images with errors and quality flags
- Extract slices from S2Ds
- Set the first and last orders per detector
- Compute the drift per slice per detector
- Save the QC KWs per detector and per slice
- Save the drift results with QC KWs.

11.2.14 wave_THAR_THAR

Recipe extracts the S2D spectrum of THAR-THAR frame. Recipe steps:

- Remove the detector signatures.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field and the blaze.
- Save the S2D and S2D_BLAZE products with QC KWs.

11.2.15 wave_LFC_LFC

Recipe extracts the S2D spectrum of LFC-LFC frame. Recipe steps:

- Check the raw frame for saturation.
- Remove the detector signatures.
- Perform optimal extraction of orders for both fibres.
- Correct the flat-field and the blaze.
- Save the S2D and S2D_BLAZE products with QC KWs.

A Installation

A.1 System Requirements

The processing of ESPRESSO data is very demanding in terms of computing resources. In particular it requires a machine with sufficient memory installed. Less critical but still important is the number of available CPU core sand the amount of available disk space. Because of the memory constraints, the ESPRESSO pipeline is only supported on 64-bit platforms. The recommended platform is a powerful workstation with a recent 64-bit Linux system. The minimum system configuration is:

- 32 GB of memory
- 4 CPU cores (physical cores)
- 1 TB of free disk space
- GCC 8.3.1 (or newer)

We recommend the user to have more resources:

- 64 GB of memory
- 8 or more CPU cores (physical cores)
- 12 TB of free disk space

The peak memory consumption is with the flat recipe, of 12 GB for each data set, in case of 20 input flats. Using Reflex as data processing tool, data sets will be processed sequentially, thus a single user will not require a large amount of memory. More useful can instead be to have a large disk space if a user processes without using interactivity a lot of data. If a user process the data using esorex, for example with a customised system of scripts, it is possible that several esorex based recipes execution in parallel, and consequently memory requirements are higher.

A.2 Installing the Pipeline KIT

The ESPRESSO pipeline with all the required tools is available from `www.eso.org/pipelines`, and depending on user's platform we recommend respectively to install via RPM (Linux) or MacPorts (macOS) and if the user's platform is not one of the supported Linux or mac OS, but is a recent Linux or Mac OS X, an installer script is available.

However, the recommended target platform for using the ESPRESSO DRS is a 64-bit Linux system, since Mac OS X imposes certain restrictions when it comes to running the pipeline. To install the pipeline unpack the kit in a temporary location, go to the top level directory of the unpacked distribution package and execute the installer script as shown in the following example.

Note: The installation script uses the compiler which is found first in the path! If more than one compiler are installed on the system one should make sure that an appropriate 64-bit compiler will be found first when the installation script is executed!

```
tar -zxf espdr-kit-X.Y.Z.tar.gz
cd espdr-kit-X.Y.Z.tar.gz
./install\_pipeline
```

Then follow the instructions on the screen. Once the script finishes successfully and the path variables have been set, the installation of the ESPRESSO pipeline is complete.

ESPRESSO provides also a Data Analysis Software. This can be installed similarly.

```
tar -zxf espda-kit-X.Y.Z.tar.gz
cd espda-kit-X.Y.Z.tar.gz
./install\_pipeline
```

A.3 Installing the Software including ESOReflex

Installation via RPM or MacPorts is recommended. In case the user platform is not one for which RPM and MacPorts are provided the user may use the `install_esoreflex script.`

ESOReflex can be installed as:

```
./install_esoreflex
```

Then follow the instructions on the screen (selecting ESPRESSO and/or ESPRESSO-DAS) Once the script finishes successfully and the path variables have been set, the installation of the ESPRESSO ESOReflex workflow and the pipeline are complete.