



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

Reflex KMOS Tutorial

VLT-MAN-ESO-19500-....

Issue 5.1

Date 2019-07-30

Prepared: L. Coccato, M. Neeser, Y. Jung W. Freudling, E. Sani 2019-07-30
.....
Name Date Signature

Approved: -
.....
Name Date Signature

Released: -
.....
Name Date Signature

This page was intentionally left blank

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 3 of 81 |

Change record

| Issue/Rev. | Date | Section affected | Reason/Initiation/Documents/Remarks |
|------------|------------|-----------------------------|--|
| 0.1 | 25/06/2013 | All | Initial Release |
| 0.2 | 15/07/2013 | All | Description of Demo Data |
| 1.0 | 19/09/2013 | All | Revision of product data description and incl. of interactive mode |
| 1.1 | 29/10/2013 | All | Use Reflex 2.5 and Integration in ESO SVN |
| 1.2 | 19/03/2014 | §9 | Description of interactive actors (v. 1.3.0), including parameter tables |
| 1.2.1 | 28/04/2015 | §5 | Inclusion of new files in the demo dataset. |
| 1.2.5 | 29/06/2015 | All; incl. new §2 | Update of figures and text to sync with v. 1.3.12. |
| 1.3 | 29/06/2015 | All | Update of figures and text to sync with v. 1.3.13 and Reflex 2.8. |
| 1.4 | 07/10/2015 | All | Update of figures and text to sync with v. 1.3.14. |
| 1.5 | 01/01/2016 | 2, 4, 5, 7, 10 | In sync with version 1.3.16 |
| 1.6 | 22/01/2016 | All | In sync with version 1.3.17 |
| 1.7 | 22/01/2016 | | In sync with version 1.3.18 |
| 2.0 | 11/10/2016 | 4-6,8-10 | In sync with version 1.4.0 |
| 3.0 | 01/02/2017 | 2, 5, 8 | In sync with version 1.4.3 |
| 3.1 | 01/04/2018 | All | In sync with pipeline version 2.0.0 and reflex version 2.9. Inclusion of molecfit and calctrans. |
| 4.0 | 01/04/2019 | 2, 4-6, 8-12, 14 | Possibility to combine exposures from multiple OBs and reduce/combine acquisition exposures. New demo dataset. Added interactive features for molecfit/calctrans modules. |
| 5.0 | 01/05/2019 | 2, 4-6, 8-12, 14 2, 8-10 | Possibility to run molecfit/calctrans on science files Additional strategy for level correction (intra-slices algorithm). |
| 6.0 | 01/11/2019 | | Workflow can reduce datasets using either raw or master calibrations. |
| 7.0 | 1/05/2020 | all | In sync with Reflex 2.10 |

This page was intentionally left blank

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 5 of 81 |

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction to <code>Esoreflex</code> | 9 |
| 2 | Workflow Status | 10 |
| 3 | Software Installation | 12 |
| 3.1 | Installing Reflex workflows via <code>macports</code> | 12 |
| 3.2 | Installing Reflex workflows via <code>rpm/yum/dnf</code> | 12 |
| 3.3 | Installing Reflex workflows via <code>install_esoreflex</code> | 13 |
| 3.4 | Demo Data | 14 |
| 4 | Quick Start: Reducing The Demo Data | 16 |
| 4.1 | Different types of Datasets | 18 |
| 5 | About the main <code>esoreflex</code> canvas | 21 |
| 5.1 | Saving And Loading Workflows | 21 |
| 5.2 | Buttons | 21 |
| 5.3 | Workflow States | 21 |
| 6 | The KMOS Workflow | 22 |
| 6.1 | Workflow Canvas Parameters | 22 |
| 6.2 | KMOS-specific workflow parameters | 23 |
| 6.3 | Workflow Actors | 25 |
| 6.3.1 | Simple Actors | 25 |
| 6.3.2 | Lazy Mode | 26 |
| 7 | Reducing your own data | 27 |
| 7.1 | The <code>esoreflex</code> command | 27 |
| 7.2 | Launching the workflow | 27 |
| 7.3 | Workflow Steps | 29 |
| 7.3.1 | Data Organisation And Selection | 29 |
| 7.3.2 | <code>DataSetChooser</code> | 30 |
| 7.3.3 | The workflow data-reduction cascade | 31 |

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 6 of 81 |

| | | |
|-----------|---|-----------|
| 7.3.4 | The ProductExplorer | 33 |
| 7.4 | Description of the reduced KMOS data | 35 |
| 7.4.1 | Description of final products | 38 |
| 8 | Optimising Your Results Through Workflow Interaction | 40 |
| 8.1 | General characteristics of the interactive windows | 40 |
| 8.2 | Wavelength Calibration | 41 |
| 8.3 | Telluric Standard Star Calibration | 43 |
| 8.4 | Reduction of science frames: reconstruction of datacubes | 45 |
| 8.4.1 | Flux calibration | 45 |
| 8.4.2 | Level correction: removing overscan and intra-slice light | 46 |
| 8.4.3 | Optimizing the sky removal via workflow parameters | 48 |
| 8.4.4 | Object / sky association for sky subtraction | 50 |
| 8.4.5 | Example of object/sky association table. | 51 |
| 8.5 | Science reduction: datacube combination | 52 |
| 8.5.1 | Combining exposures from different Observing Blocks | 53 |
| 9 | Telluric correction | 56 |
| 9.1 | Notes on the use of Molecfit tools | 57 |
| 9.2 | Computation of atmospheric model: running molecfit on selected spectra | 58 |
| 9.2.1 | Tips for an efficient fit. | 59 |
| 9.3 | Computation of atmospheric transmission: running calctrans | 60 |
| 9.4 | Correction for atmospheric transmission | 62 |
| 9.5 | Extraction of 1D science spectra | 63 |
| 9.6 | Selection of reference spectra | 64 |
| 10 | Instrumental response | 67 |
| 11 | Tips | 69 |
| 11.1 | Specifying the number of cores to use via OMP_NUM_THREADS | 69 |
| 11.2 | Dealing with residual sky emission lines. | 69 |
| 11.3 | Process data when OBJECT and SKY are taken from two separate Observing Blocks | 71 |

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 7 of 81 |

| | | |
|-----------|--|-----------|
| 11.4 | Improving the resolution of spectral line | 72 |
| 11.5 | Optimize the alignment of exposures before combination | 72 |
| 11.6 | Global background subtraction (post-processing) | 72 |
| 12 | Frequently Asked Questions | 73 |
| 13 | Troubleshooting | 76 |

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 8 of 81 |

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 9 of 81 |

1 Introduction to `Esoreflex`

This document is a tutorial designed to enable the user to to reduce his/her data with the ESO pipeline run under an user-friendly environmet, called `Esoreflex`, concentrating on high-level issues such as data reduction quality and signal-to-noise (S/N) optimisation.

`Esoreflex` is the ESO Recipe Flexible Execution Workbench, an environment to run ESO VLT pipelines which employs a workflow engine to provide a real-time visual representation of a data reduction cascade, called a workflow, which can be easily understood by most astronomers. The basic philosophy and concepts of Reflex have been discussed by [Freudling et al. \(2013A&A...559A..96F\)](#). Please reference this article if you use Reflex in a scientific publication.

Reflex and the data reduction workflows have been developed by ESO and instrument consortia and they are fully supported. If you have any issue, please contact usd-help@eso.org for further support.

A workflow accepts science and calibration data, as downloaded from the archive using the `CalSelector` tool¹ (with associated raw calibrations) and organises them into `DataSets`, where each `DataSet` contains one science object observation (possibly consisting of several science files) and all associated raw and static calibrations required for a successful data reduction. The data organisation process is fully automatic, which is a major time-saving feature provided by the software. The `DataSets` selected by the user for reduction are fed to the workflow which executes the relevant pipeline recipes (or stages) in the correct order. Full control of the various recipe parameters is available within the workflow, and the workflow deals automatically with optional recipe inputs via built-in conditional branches. Additionally, the workflow stores the reduced final data products in a logically organised directory structure employing user-configurable file names.

This tutorial deals with the reduction of KMOS IFU observations only via the `KMOS Reflex` workflow. The user is referred to the KMOS user manual (Cirasuolo [4]²) and to the ESO instrument web pages³ for more information on the instrument itself as well as a summary of available documentation, recent news, and tools. A brief description of the KMOS data flow is given in Davies [2]⁴.

The quick start section (see Section 4) describes the minimum effort to get started, and it makes up only few pages of text in this tutorial. User support for this software is available by sending enquiries to usd-help@eso.org.

¹<http://www.eso.org/sci/archive/calselectorInfo.html>

²available at: <http://www.eso.org/sci/facilities/paranal/instruments/kmos/doc>

³<http://www.eso.org/sci/facilities/develop/instruments/kmos.html>

⁴available at: <http://www.mpe.mpg.de/389507/davies-7735-254x.pdf>

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 10 of 81 |

2 Workflow Status

The KMOS Reflex workflow, in its current version, has reached an advanced level and is capable, together with its underlying KMOS pipeline, of delivering high quality, science-ready data products. The Reflex workflow is built upon the KMOS pipeline delivered by the MPE and improved on by ESO, using experience gained in the first few years of KMOS operations and from the KMOS User community.

The first step of the KMOS Reflex workflow is to organize the data of this complex instrument into an associated, organized, and classified structure based on the calibration type, its proximity in time to the science data, its rotator angle, and its match with the filter and grism. The User will be warned if any calibration frames are missing.

The KMOS Reflex workflow will correct the frames for their dark level and structure, flat-field the data, compute a wavelength solution, apply an illumination correction, apply a standard star flux calibration and telluric correction, and create a cube reconstruction of the science data. A large number of data products are created and retained for the User to assess the quality of the pipeline processing.

Starting from version 2.0.10, the workflow can combine data of the same target from multiple Observing Blocks, and reduce acquisition frames (if available) and (optionally) combine them with science data.

Starting from version 2.7 the workflow can reduce datasets using either raw or master calibrations.

During the processing within the Reflex workflow, the User has the ability to modify a large number of pipeline parameters in order to optimize the data processing. This is most conveniently done within the interactive Actors associated with the wavelength calibration, the standard star, and the science data reduction. Moreover, the KMOS workflow allows the User to select several data reduction strategies.

- The illumination correction can be performed by using dedicated twilight flats or internal lamp flats. Despite the twilight flats have the same optical path as the scientific data, in many cases the use of internal lamp gives best results in terms of uniformity of the correction and signal-to-noise.
- The telluric correction is computed by the workflow from the observed telluric standard or directly on the science spectra. If the telluric standard is used, two different algorithms are available. The first (recipe *kmos_std_star*) compares the observed telluric standard star with a spectral model that accounts for the star spectral type. The second algorithm exploits the molecfit and calctrans software. First, it fits an atmospheric model to the observed observed telluric standard (recipe *kmos_molecfit_model*) accounting also for some instrumental effects (adjustment of wavelength calibration, and variable instrumental line spread function profiles). Secondly, it construct the telluric correction by using the atmospheric model and accounting for the airmass difference between the observations of the scientific target and the telluric standard star (recipe *kmos_molecfit_calctransa*). If the telluric correction is evaluated directly on the science data, the workflow uses molecfit and calctrans software directly on the science frames.
- The correction for instrument response can take into account the differences between different IFUs.

Although the KMOS pipeline and Reflex workflow have reached high levels of sophistication, the complexity and variety of KMOS data implies that there will always be room for improvement. In particular:

- The background sky subtraction is very sensitive to the quality of the input data and may not always work optimally.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 11 of 81 |

- For observations carried on extremely good seeing conditions (e.g., $\text{FWHM} \leq 0.''5$ arcseconds), point-like sources are undersampled in the chip, therefore the reconstruction algorithm can produce ripples (i.e. a wavy-pattern along wavelength) in the final reconstructed spectrum. This effect can be mitigated, although not fully corrected at the moment, by using the cubic-spline interpolation for both standard star and science observation (`-imethod=CS`), and by enlarging the aperture radius over which the standard star spectrum and the science spectrum are extracted (see Section 13).
- An optimal source and spectral extraction is not yet available within the KMOS pipeline or workflow. In this case it is up to the User to match a spectral extraction to the characteristics of their sources.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 12 of 81 |

3 Software Installation

`Esoreflex` and the workflows can be installed in different ways: via package repositories, via the `install_esoreflex` script or manually installing the software tar files.

The recommended way is to use the package repositories if your operating system is supported. The `macports` repositories support macOS 10.11 to 10.14, while the `rpm/yum` repositories support Fedora 28 to 31, CentOS 7, Scientific Linux 7. For any other operating system it is recommended to use the `install_esoreflex` script.

The installation from package repository requires administrative privileges (typically granted via `sudo`), as it installs files in system-wide directories under the control of the package manager. If you want a local installation, or you do not have `sudo` privileges, or if you want to manage different installations on different directories, then use the `install_esoreflex` script. Note that the script installation requires that your system fulfill several software prerequisites, which might also need `sudo` privileges.

Reflex 2.10 needs java JDK 11 to be installed.

Please note that in case of major or minor (affecting the first two digit numbers) Reflex upgrades, the user should erase the `$HOME/KeplerData`, `$HOME/.kepler` directories if present, to prevent possible aborts (i.e. a hard crash) of the `esoreflex` process.

3.1 Installing Reflex workflows via `macports`

This method is supported for the macOS operating system. It is assumed that `macports` (<http://www.macports.org>) is installed. Please read the full documentation at <http://www.eso.org/sci/software/pipelines/installation/macports.html>.

3.2 Installing Reflex workflows via `rpm/yum/dnf`

This method is supported for Fedora 26 to 29, CentOS 7, Scientific Linux 7 operating systems, and requires `sudo` rights. To install, please follow these steps

1. Configure the ESO repository (This step is only necessary if the ESO repository has not already been previously configured).

- If you are running Fedora 26 or newer, run the following commands:

```
sudo dnf install dnf-plugins-core
sudo dnf config-manager --add-repo=ftp://ftp.eso.org/pub/dfs/
pipelines/repositories/stable/fedora/esorepo.repo
```

- If you are running CentOS 7, run the following commands:

```
sudo yum install yum-utils ca-certificates yum-conf-repos
sudo yum install epel-release
sudo yum-config-manager --add-repo=ftp://ftp.eso.org/pub/dfs/
pipelines/repositories/stable/centos/esorepo.repo
```

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 13 of 81 |

- If you are running SL 7, run the following commands:

```
sudo yum install yum-utils ca-certificates yum-conf-repos
sudo yum install yum-conf-epel
sudo yum-config-manager --add-repo=ftp://ftp.eso.org/pub/dfs/
pipelines/repositories/stable/sl/esorepo.repo
```

2. Install the pipelines

- The list of available top level packages for different instruments is given by:

```
sudo dnf list esopipe-\*-all # (Fedora 26 or newer)
sudo yum list esopipe-\*-all # (CentOS 7, SL 7)
```

- To install an individual pipeline use the following (This example is for X-Shooter. Adjust the package name to the instrument you require.):

```
sudo dnf install esopipe-xshoo-all # (Fedora 26 or newer)
sudo yum install esopipe-xshoo-all # (CentOS 7, SL 7)
```

- To install all pipelines use:

```
sudo dnf install esopipe-\*-all # (Fedora 26 or newer)
sudo yum install esopipe-\*-all # (CentOS 7, SL 7)
```

For further information, please read the full documentation at
<http://www.eso.org/sci/software/pipelines/installation/rpm.html>.

3.3 Installing Reflex workflows via `install_esoreflex`

This method is recommended for operating systems other than what indicated above, or if the user has no sudo rights. Software dependencies are not fulfilled by the installation script, therefore the user has to install all the prerequisites before running the installation script.

The software pre-requisites for Reflex 2.9 may be found at:
http://www.eso.org/sci/software/pipelines/reflex_workflows

To install the Reflex 2.9 software and demo data, please follow these instructions:

1. From any directory, download the installation script:

```
wget ftp://ftp.eso.org/pub/dfs/reflex/install_esoreflex
```

2. Make the installation script executable:

```
chmod u+x install_esoreflex
```

3. Execute the installation script:

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 14 of 81 |

```
./install_esoreflex
```

and the script will ask you to specify three directories: the download directory `<download_dir>`, the software installation directory `<install_dir>`, and the directory to be used to store the demo data `<data_dir>`. If you do not specify these directories, then the installation script will create them in the current directory with default names.

4. Follow all the script instructions; you will be asked whether to use your Internet connection (recommended: yes), the pipelines and demo-datasets to install (note that the installation will remove all previously installed pipelines that are found in the same installation directory).
5. To start `Reflex`, issue the command:

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

It may also be desirable to set up an alias command for starting the `Reflex` software, using the shell command `alias`. Alternatively, the `PATH` variable can be updated to contain the `<install_dir>/bin` directory.

3.4 Demo Data

Together with the pipeline you will also receive two demo data sets, that allow you to run the `Reflex` KMOS workflow without any changes in parameters. This way you have data sets to experiment with before you start to work on your own data.

Note that you will need a minimum of ~ 14 GB of free disk space for the pipeline, calibration files, and `reflex` itself, and ~ 8 GB for the demo data. Also, you need at least ~ 12 Gb for processing the demo datasets once with default parameters.

The KMOS workflow organizes the input data and defines datasets. Typically, there are 3 different types, depending on their content.

- Type 1: Dataset from single Observing Block (OB). They contain `SCIENCE` that belong to the same OB (they all have the same `TPL.START` header keyword), and they are identified with the suffix `_tpl` in the name.
- Type 2: Dataset from different OBs but of the same target. They contain `SCIENCE` that belong to the same target (they all have the same `OBS.TARG.NAME` header keyword), and they are identified with the suffix `_combine_OBs` in the name. They contain at least two sets of `SCIENCE` frames from two different OBs.
- Type 3: Reconstructed cubes. They contain the reconstructed cubes of the same target (they all have the same `OBS.TARG.NAME` header keyword) obtained from previous reduction. They can contain data from different OBs.

In the demo dataset folder, the raw input sets of data defines the following datasets:

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 15 of 81 |

- Three SCIENCE OBs of the merger remnant galaxy NGC 6240 observed in mosaic mode (mapping 8). Each OB consists of 4 exposures of the same field of view (32.5"×16.3") and 1 sky exposure. This set of files generates four datasets: three type 1 datasets, one for each Observing Block (KMOS.2015-05-18:T05:38:12.110_tpl, KMOS.2015-05-18:T06:47:33.130_tpl, KMOS.2015-05-29:T03:18:11.720_tpl), and one type 2 dataset that is designed to combine the 3 Observing Blocks (KMOS.2015-05-18:T05:38:12.110_combine_OBs).
- One single SCIENCE observing block (OB) of the HII, star-formation region Gum 43 (RCW 65) executed in freedither mode mode. The OB consists of 9 exposures, each of them targeting 19 individual regions within Gum 43, 5 on exposures on offset-sky, and 1 acquisition exposure with its associated sky exposure. This set of files generates 2 datasets: KMOS.2013-06-30:T23:48:06.049_tpl (type 1) and KMOS.2013-06-30:T23:48:06.049_combine_OBs (type 2). Because these data belongs to the same observing block, the last dataset (combination of multiple OBs) is incomplete because it requires at least 2 OBs.
- Two sets of reconstructed cubes (PRO . CATG = SINGLE_CUBES). These are the products of two already-reduced Observing Blocks of the lens system MACS J0416, observed in nod-to-sky mode, that can be combined together. The OBs consists of 5 exposures each and target 13 individual objects. This set of files generates one type 3 dataset (MACS0416-K- [...]).

Note: for a generic set of data, the header keyword HIERARCH ESO TPL ID indicates the template used for observations, and it specifies the observing mode (e.g., mosaic, nod-to-sky). In non-mosaic modes there are other important header keywords that describe the observational set up. They are (with n ranging form 1 to 24):

- HIERARCH ESO OCS ARMn TYPE It specify if the n-th arm is pointing to an object ("O") or to a sky position ("S").
- HIERARCH ESO OCS ARMn NAME It specify the name of the object that is targeted by the n-th arm
- HIERARCH ESO OCS ARMn LOCK. It specify if the arm was in use or not

All demo files include the raw calibration frames darks, lamp flats, and arc lamps. Static calibrations (arc line lists, atmosphere model, OH spectral lines, and arc reference lines) are included in the pipeline distribution.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 16 of 81 |

4 Quick Start: Reducing The Demo Data

For the user who is keen on starting reductions without being distracted by detailed documentation, we describe the steps to be performed to reduce the science data provided in the KMOS demo data set supplied with the `esoreflex 2.9` release. By following these steps, the user should have enough information to perform a reduction of his/her own data without any further reading:

1. First, type:

```
esoreflex -l
```

If the `esoreflex` executable is not in your path, then you have to provide the command with the executable full path `<install_dir>/bin/esoreflex -l`. For convenience, we will drop the reference to `<install_dir>`. A list with the available `esoreflex` workflows will appear, showing the workflow names and their full path.

2. Open the `kmoss` by typing:

```
esoreflex kmoss&
```

Alternatively, you can type only the command `esoreflex` the empty canvas will appear (Figure 4.1) and you can select the workflow to open by clicking on `File -> Open File`. Note that the loaded workflow will appear in a new window. The `kmoss` workflow is shown in Figure 4.2.

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 .
4. Change directories set-up. Under “Setup Directories” in the workflow canvas there are seven parameters that specify important directories (green dots).

By default, the `ROOT_DATA_DIR`, which specifies the working directory within which the other directories are organised. is set to your `$HOME/reflex_data` directory. All the temporary and final products of the reduction will be organized under sub-directories of `ROOT_DATA_DIR`, therefore make sure this parameter points to a location where there is enough disk space. To change `ROOT_DATA_DIR`, double click on it and a pop-up window will appear allowing you to modify the directory string, which you may either edit directly, or use the button to select the directory from a file browser. When you have finished, click to save your changes.

Changing the value of `RAW_DATA_DIR` is the only necessary modification if you want to process data other than the demo data

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

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 17 of 81 |

in `RAW_DATA_DIR` or in `CALIB_DATA_DIR`, otherwise datasets may be incomplete and cannot be processed. However, if the same reference file was downloaded twice to different places this creates a problem as `esoreflex` 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 4.1) that lists the datasets along with the values of a selection of useful header keywords⁵. 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. A full description of the options offered by the `Data Set Chooser` will be presented in Section 7.3.2.
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 4.4 shows the *Product Explorer* window. A full description of the *Product Explorer* will be presented in Section 7.3.4.
10. After the workflow has finished, all the products from all the datasets can be found in a directory under `END_PRODUCTS_DIR` named after the workflow start timestamp. Further subdirectories will be found with the name of each dataset.

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 `KMOS` workflow that merit a look at the rest of this tutorial.

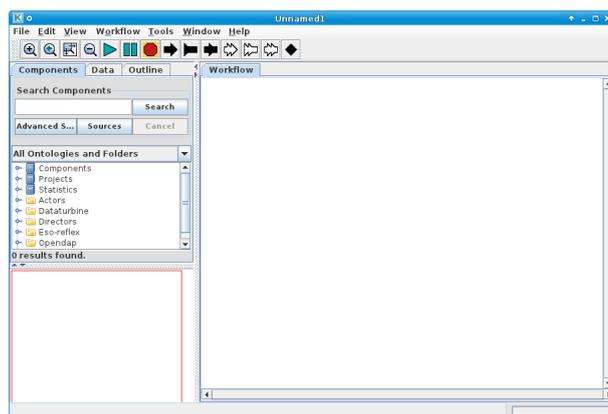


Figure 4.1: *The empty Reflex canvas.*

⁵The keywords listed can be changed by double clicking on the `DataOrganiser` Actor and editing the list of keywords in the second line of the pop-up window. Alternatively, instead of double-clicking, you can press the right mouse button on the `DataOrganiser` Actor and select `Configure Actor` to visualize the pop-up window.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 18 of 81 |

4.1 Different types of Datasets

As shown in Figure 4.1, there are 3 types of datasets.

- The exposures associated to a single observing block (OB). The dataset is identified with the suffix `_tpl` in its name, and all the `SCIENCE` exposures that belong to this dataset have the same `TPL.START`.
- The exposures associated to a specific object, even from different OBs. The dataset is identified with the suffix `_combine_OBs` in its name. All the `SCIENCE` exposures that belong to this dataset have the same `OBS.TARG.NAME`.
- The reconstructed datacubes `SCI_RECONSTRUCTED` or `SINGLE_CUBES` produced by previous reduction that are associated to a specific object, even from different OBs.

Note: during the workflow execution, a number of interactive windows associated to specific pipeline recipes will be displayed. In these windows, the user can modify the parameters of corresponding recipe and re-run it. They will be explained in Section 8; press the “Continue” button to proceed with the workflow execution.

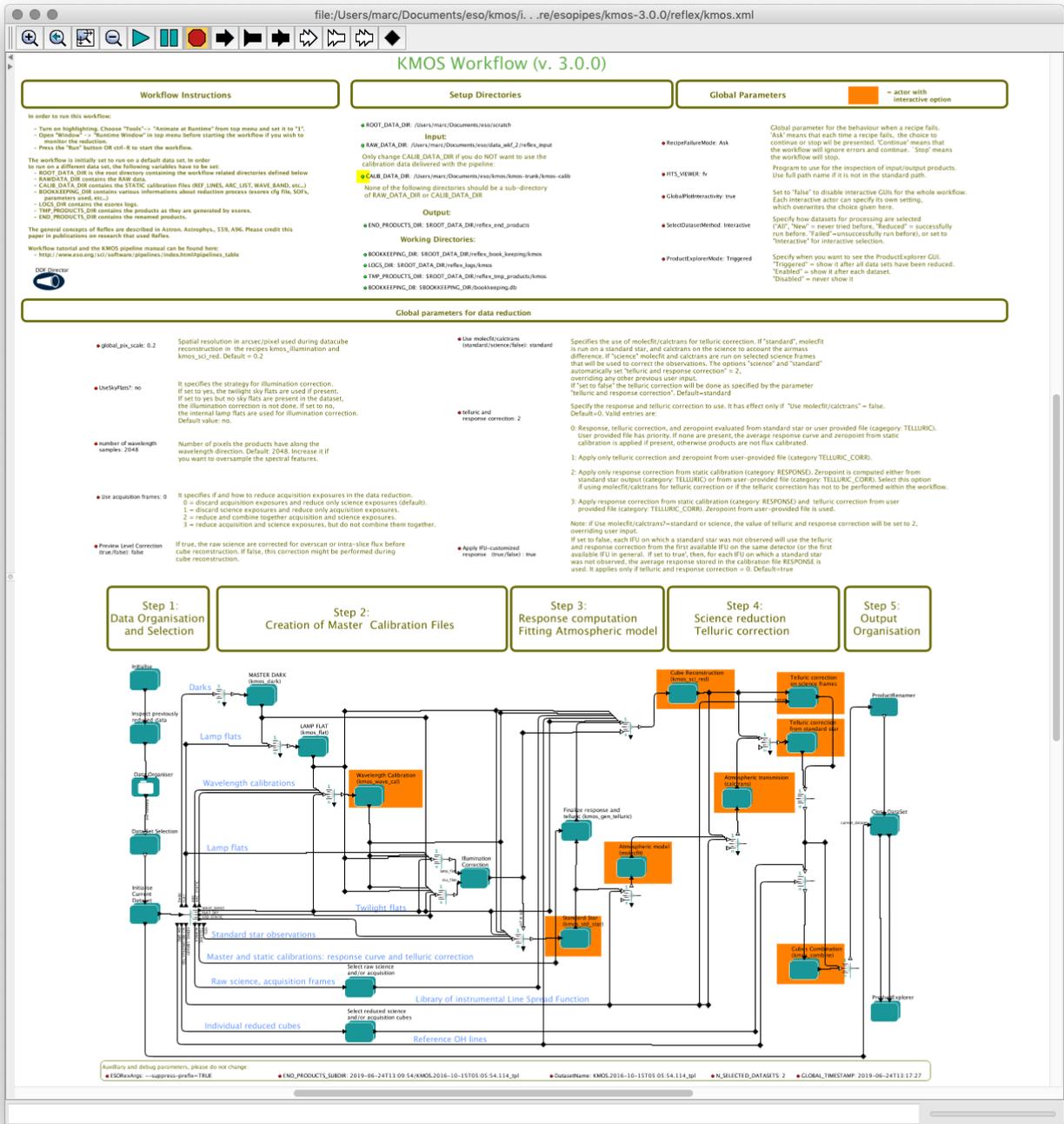


Figure 4.2: KMOS workflow general layout.

| | | | |
|------------|-----------------------------|---------------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 20 of 81 |

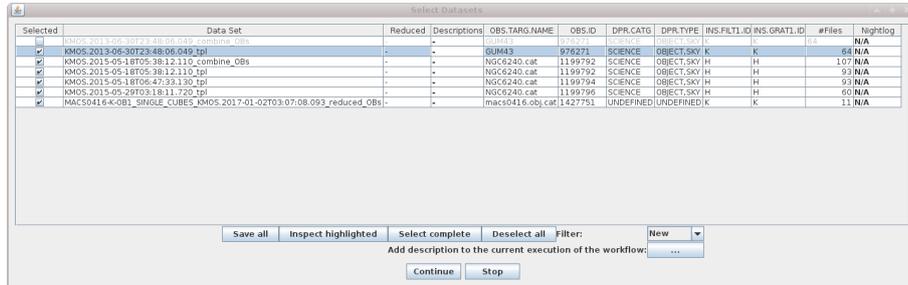


Figure 4.3: The “Select Datasets” pop-up window

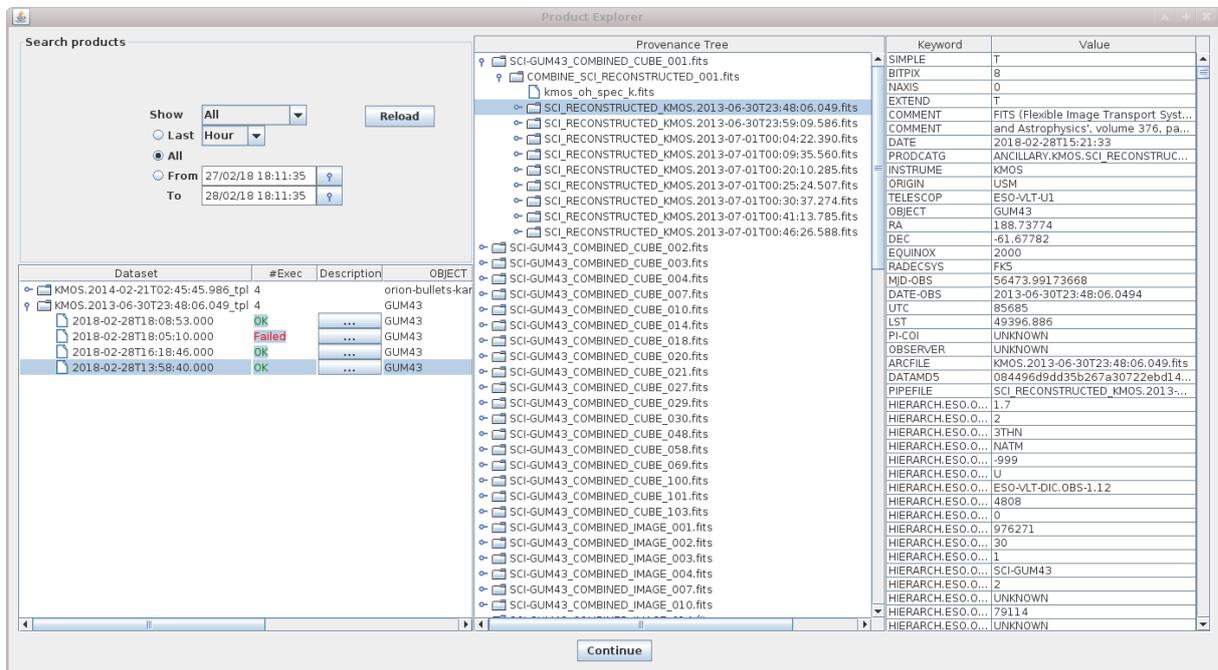


Figure 4.4: The KMOS product explorer.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 21 of 81 |

5 About the main `esoreflex` canvas

5.1 Saving And Loading Workflows

In the course of your data reductions, it is likely that you will customise the workflow for various data sets, even if this simply consists of editing the `ROOT_DATA_DIR` to a different value for each data set. Whenever you modify a workflow in any way, you have the option of saving the modified version to an XML file using `File -> Export As` (which will also open a new workflow canvas corresponding to the saved file). The saved workflow may be opened in subsequent `esoreflex` sessions using `File -> Open`. Saving the workflow in the default Kepler format (`.kar`) is only advised if you do not plan to use the workflow with another computer.

5.2 Buttons

At the top of the `esoreflex` canvas are a set of buttons which have the following functions:

-  - Zoom in.
-  - Reset the zoom to 100%.
-  - Zoom the workflow to fit the current window size (Recommended).
-  - Zoom out.
-  - Run (or resume) the workflow.
-  - Pause the workflow execution.
-  - Stop the workflow execution.

The remainder of the buttons (not shown here) are not relevant to the workflow execution.

5.3 Workflow States

A workflow may only be in one of three states: executing, paused, or stopped. These states are indicated by the yellow highlighting of the , , and  buttons, respectively. A workflow is executed by clicking the  button. Subsequently the workflow and any running pipeline recipe may be stopped immediately by clicking the  button, or the workflow may be paused by clicking the  button which will allow the current actor/recipe to finish execution before the workflow is actually paused. After pausing, the workflow may be resumed by clicking the  button again.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 22 of 81 |

6 The KMOS Workflow

The KMOS workflow canvas is organised into a number of areas. From top-left to top-right you will find general workflow instructions, directory parameters, and global parameters. In the middle row you will find five boxes describing the workflow general processing steps in order from left to right, and below this the workflow actors themselves are organised following the workflow general steps.

6.1 Workflow Canvas Parameters

The workflow canvas displays a number of parameters that may be set by the user. Under “Setup Directories” the user is only required to set the `RAW_DATA_DIR` to the working directory for the dataset(s) to be reduced, which, by default, is set to the directory containing the demo data. The `RAW_DATA_DIR` is recursively scanned by the `Data Organiser` actor for input raw data. The directory `CALIB_DATA_DIR`, which is by default within the pipeline installation directory, is also scanned by the `Data Organiser` actor to find any static calibrations that may be missing in your dataset(s). If required, the user may edit the directories `BOOKKEEPING_DIR`, `LOGS_DIR`, `TMP_PRODUCTS_DIR`, and `END_PRODUCTS_DIR`, which correspond to the directories where book-keeping files, logs, temporary products and end products are stored, respectively (see the Reflex User Manual for further details; [5]).

There is a mode of the `Data Organiser` that skips the built-in data organisation and uses instead the data organisation provided by the `CalSelector` tool. To use this mode, click on `Use CalSelector associations` in the `Data Organiser` properties and make sure that the input data directory contains the XML file downloaded with the `CalSelector` archive request (note that this does not work for all instrument workflows).

Under the “Global Parameters” area of the workflow canvas, the user may set the `FITS_VIEWER` parameter to the command used for running his/her favourite application for inspecting FITS files. Currently this is set by default to `fv`, but other applications, such as `ds9`, `skycat` and `gaia` for example, may be useful for inspecting image data. Note that it is recommended to specify the full path to the visualization application (an alias will not work).

By default the `EraseDirs` parameter is set to `false`, which means that no directories are cleaned before executing the workflow, and the recipe actors will work in `Lazy Mode` (see Section 6.3.2), reusing the previous pipeline recipe outputs if input files and parameters are the same as for the previous execution, which saves considerable processing time. Sometimes it is desirable to set the `EraseDirs` parameter to `true`, which forces the workflow to recursively delete the contents of the directories specified by `BOOKKEEPING_DIR`, `LOGS_DIR`, and `TMP_PRODUCTS_DIR`. This is useful for keeping disk space usage to a minimum and will force the workflow to fully re-reduce the data each time the workflow is run.

The parameter `RecipeFailureMode` controls the behaviour in case that a recipe fails. If set to `Continue`, the workflow will trigger the next recipes as usual, but without the output of the failing recipe, which in most of the cases will lead to further failures of other recipes without the user actually being aware of it. This mode might be useful for unattended processing of large number of datasets. If set to `Ask`, a pop-up window will ask whether the workflow should stop or continue. This is the default. Alternatively, the `Stop` mode will stop the workflow execution immediately.

The parameter `GlobalPlotInteractivity` controls whether the interactive windows will appear for those windows which are *enabled* by default. The possible values are `true`, `false`. Take into account that some

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 23 of 81 |

windows are disabled in the default configuration and therefore are not affected by this parameter.

The parameter `ProductExplorerMode` controls whether the `ProductExplorer` actor will show its window or not. The possible values are `Enabled`, `Triggered`, and `Disabled`. `Enabled` opens the `ProductExplorer` GUI at the end of the reduction of each individual dataset. `Triggered` (default and recommended) opens the `ProductExplorer` GUI when all the selected datasets have been reduced. `Disabled` does not display the `ProductExplorer` GUI.

6.2 KMOS-specific workflow parameters

The main KMOS workflow canvas contains several parameters that are specific to the KMOS data reduction and specify the reduction strategy. They are:

- **global_pix_scale.** It specifies the spatial pixel scale in “/pixel of the outputs. The default value (0.2) is consistent with the nominal spatial sampling of KMOS, and is suited for the majority of the observations. If combining many exposures of extended sources taken in good seeing conditions with an offset smaller than 0.2 arcseconds it might be worth decreasing this value (e.g., to 0.1) to improve the spatial sampling of the final combined cube.
- **UseSkyFlats?** It specifies the strategy for the illumination correction. If set to “yes” (no quotation marks), the actor delivers the illumination correction constructed using sky flats (`FLAT_SKY`), if present, otherwise it delivers nothing and the illumination correction is not performed. Any other value of the variable makes the actor to deliver the illumination correction produced using lamp flats (`FLAT_ON`). If the twilight flats are taken with the rotator angle close to the science observation, it might be advisable to set the parameter to yes; otherwise it might be advisable to use lamp flats. The user is encouraged to reduce the datasets with both strategies and evaluate the benefits. The advantage of using skyflats is that the light paths is the same as the science data, however the rotator angle might differ. The advantage of using lamp flats is that they are taken with multiple rotator angles, however the light path differs from the science data. One diagnostic tool is to compare the reconstructed cubes of the science data obtained with the two strategies without removing the sky (this can be done by setting `no-subtract` to `true` and `skytweak` to `false` in the `Cube Reconstruction` actor). The best strategy is the one that shows the more uniform reconstructed image. So far, there is indication that setting **UseSkyFlats?** = `false` (i.e. using `FLAT_ON`) gives the best results.
- **number of wavelength samples.** It specifies the number of pixels to use to sample the wavelength axis. Default 2048, that corresponds to inverse dispersions of $2.16 \text{ \AA pixel}^{-1}$ (H band), $4.64 \text{ \AA pixel}^{-1}$ (HK band), $1.51 \text{ \AA pixel}^{-1}$ (IZ band), $2.81 \text{ \AA pixel}^{-1}$ (K band), and $1.75 \text{ \AA pixel}^{-1}$ (YJ band). It corresponds to the recipe parameter `-b_samples`.
- **Use acquisition frames.** It specifies if and how acquisition exposures have to be reduced and combined. Valid values are:
 - 0 = discard acquisition exposures and reduce only science exposures (default).
 - 1 = discard science exposures and reduce only acquisition exposures.
 - 2 = reduce and combine together acquisition and science exposures.
 - 3 = reduce acquisition and science exposures, but do not combine them together.
 Note that, even if option 2 is selected, the combination will be done only for those IFUs that target the same source, as specified by the `ARMn.NAME` keyword.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 24 of 81 |

- **telluric and response correction** It specifies the strategy of the telluric and instrument response corrections, and the flux calibration to adopt. by selecting which input file has to be used by `Cube Reconstruction`. It can have one of the following values:

- 0 Response, telluric correction, and zeropoint are evaluated from standard star observations in the `Standard Star` actor. These calibrations are saved into the product of `kmos_std_star` recipe with category `TELLURIC`. Alternatively, the user can provide its own `TELLURIC` file into the raw data directory; the user provide file has priority over the one produced by the pipeline. If no `TELLURIC` file is present in the dataset, nor computed by the workflow, the average response curve and zeropoint from static calibration (category `RESPONSE`) are applied if present, otherwise products are not flux calibrated.
- 1 Apply only telluric correction and zeropoint from user-provided file with category `TELLURIC_CORR` in the raw data directory. This file is supposed to contain information only on the telluric correction and zeropoint. The format is identical to that of the `TELLURIC` products.
- 2 This is the default option. Apply only response correction from static calibration (category: `RESPONSE`). Zeropoint is computed either from standard star or from user-provided file (category: `TELLURIC`). If no user-provided file nor standard star observations are present, the mean zeropoint information stored in the header of the static `RESPONSE` calibration file is used. Select this option if setting **Use molecfi/calctrans = standard** or if the telluric correction has not to be performed within the workflow.
- 3 Combine response correction from static calibration (category: `RESPONSE`) and telluric correction from user provided file (category: `TELLURIC_CORR`) and apply it to the science data. The combination of the two corrections are saved into a fits file (category `TELLURIC`). The zeropoint from user provided file is used.

Note that if **Use molecfi/calctrans** = “standard”, then the value of **telluric and response correction** is automatically set to 2, overriding any previous user input.

For convenient, we describe here the content of the `TELLURIC`, `TELLURIC_CORR`, and `RESPONSE` files:

- `TELLURIC`. The file contains the combined information of instrument response and telluric correction, and the zeropoint. By default, it is created within the `Standard Star` actor (recipe `kmos_std_star`) by comparing the extracted 1D spectrum of the standard star with a model of that star that takes into account its magnitude and spectral type, and the average grism spectral resolution.
- `TELLURIC_CORR`. The file contains only the information of the telluric correction and zeropoint. It is generated by the recipe `kmos_molecfi_calctrans`. A `TELLURIC_CORR` file can be present in the input raw directory and can be used instead of those produced by the workflow.
- `RESPONSE`. The file contains only the median instrument response and the median zeropoint, computed a set of A0 standard stars observed during the instrument lifetime. The responses was extracted by removing the atmospheric telluric contribution (evaluated using `molecfi`) from the `TELLURIC` files produced by the pipeline. The median response has errors of $\sim 5\%$ (in the regions not affected by telluric absorptions) up to $\sim 15\%$ (in the regions most affected by telluric absorptions). The current pipeline installation provides these response files in the static calibration directory. Alternatively, the user can follow the prescription outlined in Section 10 to generate its own response calibration.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 25 of 81 |

- **Apply IFU-customized response? (yes/no).** It applies only if **telluric and response correction = 0**. If set to “no”, each IFU on which a standard star was not observed will use the response correction from the first available IFU on the same detector (or the first available IFU in general, if there is none in the same detector). If set to “yes”, then, for each IFU on which a standard star was not observed, the average response stored in the calibration file `RESPONSE` is used. The set-up **telluric and response correction = 0** and **Apply IFU-customized response=false** (that has effects only if **Use molecfit/calctrans = false**) it is recommended only if the telluric star is observed in all the IFUs.

- **Use molecfit/calctrans.** If set to “standard” or “science”, it uses the molecfit and calctrans algorithms to correct for telluric absorption features. If “standard” the atmospheric model is computed on observed standard stars and the correction for the science frames is computed by taking into account the difference in airmass between the standard stars and the first science exposure in the observing block. If this option is selected, the parameter **telluric and response correction** is set to 2 overriding previous used decision. If set to “standard”, but standard stars observations are not present in the dataset, the scientific data are corrected only for instrumental response, not for telluric absorptions. If set to “science” the molecfit/calctrans algorithms are executed on science data and not on the standard star.

The use of molecfit/calctrans gives in general better results than the “traditional” method, in particular for bright objects, but it is very slow. Moreover, if the continuum of the target is very dim, one might consider not to apply the telluric correction at all.

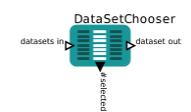
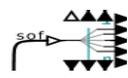
More information can be found in Section 9.

- **Preview Level correction** If true, the raw science are corrected for overscan or intra-slice flux before cube reconstruction. If false, this correction might be performed during cube reconstruction. Default (false). The default strategy, running overscan correction when doing cube reconstruction typically gives the best results. Preview Level correction = true can be used only if the user wants to explore intra-slice correction algorithm, which is can be used only if the object is very faint. More information are available on Section 8.4.2.

6.3 Workflow Actors

6.3.1 Simple Actors

Simple actors have workflow symbols that consist of a single (rather than multiple) green-blue rectangle. They may also have an icon within the rectangle to aid in their identification. The following actors are simple actors:

-  - The DataOrganiser actor.
-  - The DataSetChooser actor (inside a composite actor).
-  - The FitsRouter actor Redirects files according to their categories.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 26 of 81 |

-  - The `ProductRenamer` actor.
-  - The `ProductExplorer` actor (inside a composite actor).

Access to the parameters for a simple actor is achieved by right-clicking on the actor and selecting `Configure Actor`. This will open an “Edit parameters” window. Note that the `Product Renamer` actor is a jython script (Java implementation of the Python interpreter) meant to be customised by the user (by double-clicking on it).

6.3.2 Lazy Mode

By default, all `RecipeExecutor` actors in a pipeline workflow are “Lazy Mode” enabled. This means that when the workflow attempts to execute such an actor, the actor will check whether the relevant pipeline recipe has already been executed with the same input files and with the same recipe parameters. If this is the case, then the actor will not execute the pipeline recipe, and instead it will simply broadcast the previously generated products to the output port. The purpose of the Lazy Mode is therefore to minimise any reprocessing of data by avoiding data re-reduction where it is not necessary.

One should note that the actor’s Lazy Mode depends on the contents of the directory specified by the parameter `BOOKKEEPING_DIR` and the relevant FITS file checksums. Any modification to the directory contents and/or the file checksums will cause the corresponding actor to run the pipeline recipe again when executed, thereby re-reducing the input data.

The re-reduction of data at each execution may sometimes be desirable. To force a re-reduction of data for any single `RecipeExecutor` actor in the workflow, right-click the actor, select `Configure Actor`, and uncheck the Lazy mode parameter tick-box in the “Edit parameters” window that is displayed. For many workflows the `RecipeExecutor` actors are actually found inside the composite actors in the top level workflow. To access such embedded `RecipeExecutor` actors you will first need to open the sub-workflow by right-clicking on the composite actor and then selecting `Open Actor`.

To force the re-reduction of all data in a workflow (i.e. to disable Lazy mode for the whole workflow), you must uncheck the Lazy mode for every single `RecipeExecutor` actor in the entire workflow. It is also possible to change the name of the bookkeeping directory, instead of modifying any of the Lazy mode parameters. This will also force a re-reduction of the given dataset(s). A new reduction will start (with the lazy mode still enabled), but the results of previous reduction will not be reused. Alternatively, if there is no need to keep any of the previously reduced data, one can simply set the `EraseDirs` parameter under the “Global Parameters” area of the workflow canvas to `true`. This will then remove all previous results that are stored in the bookkeeping, temporary, and log directories before processing the input data, in effect, starting a new clean data reduction and re-processing every input dataset. *Note: The option `EraseDirs = true` does not work in esoreflex version 2.9.x and makes the workflow to crash.*

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 27 of 81 |

7 Reducing your own data

In this section we describe how to reduce your own data set.

First, we suggest the reader to familiarize with the workflow by reducing the demo dataset first (Section 4), but it is not a requirement.

7.1 The `esoreflex` command

We list here some options associated to the `esoreflex` command. We recommend to try them to familiarize with the system. In the following, we assume the `esoreflex` executable is in your path; if not you have to provide the full path `<install_dir>/bin/esoreflex`

To see the available options of the `esoreflex` command type:

```
esoreflex -h
```

The output is the following.

```
-h | -help           print this help message and exit.
-v | -version       show installed Reflex version and pipelines and exit.
-l | -list-workflows list available installed workflows and from
                    ~/KeplerData/workflows.
-n | -non-interactive enable non-interactive features.
-e | -explore       run only the Product Explorer in this workflow
-p <workflow> | -list-parameters <workflow>
                    lists the available parameters for the given
                    workflow.
-config <file>     allows to specify a custom esoreflex.rc configuration
                    file.
-create-config <file> if <file> is TRUE then a new configuration file is
                    created in ~/.esoreflex/esoreflex.rc. Alternatively
                    a configuration file name can be given to write to.
                    Any existing file is backed up to a file with a '.bak'
                    extension, or '.bakN' where N is an integer.
-debug            prints the environment and actual Reflex launch
                    command used.
```

7.2 Launching the workflow

We list here the recommended way to reduce your own datasets. Steps 1 and 2 are optional and one can start from step 3.

1. Type: `esoreflex -n <parameters> kmos` to launch the workflow non interactively and reduce all the datasets with default parameters.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 28 of 81 |

<parameters> allows you to specify the workflow parameters, such as the location of your raw data and the final destination of the products.

For example, type (in a single command line):

```
esoreflex -n
  -RAW_DATA_DIR /home/user/my_raw_data
  -ROOT_DATA_DIR /home/user/my_reduction
  -END_PRODUCTS_DIR $ROOT_DATA_DIR/reflex_end_products
kmos
```

to reduce the complete datasets that are present in the directory `/home/user/my_raw_data` and that were not reduced before. Final products will be saved in `/home/user/my_reduction/reflex_end_products`, while book keeping, temporary products, and logs will be saved in sub-directories of `/home/user/my_reduction/`. If the reduction of a dataset fails, the reduction continues to the next dataset. It can take some time, depending on the number of datasets present in the input directory. For a full list of workflow parameters type `esoreflex -p kmos`. Note that this command lists only the parameters, but does not launch the workflow.

Once the reduction is completed, one can proceed with optimizing the results with the next steps.

2. Type:

```
esoreflex -e kmos
```

to launch the Product Explorer. The Product Explorer allows you to inspect the data products already reduced by the `kmos esoreflex` workflow. Only products associated with the workflow default book-keeping database are shown. To visualize products associated to given bookkeeping database, pass the full path via the `BOOKKEEPING_DB` parameter:

```
esoreflex -e BOOKKEEPING_DB <database_path> kmos
```

to point the product explorer to a given `<database_path>`, e.g., `/home/username/reflex/reflex_bookkeeping/test.db`

The Product Explorer allows you to inspect the products while the reduction is running. Press the button to update the content of the Product Explorer. This step can be launched in parallel to step 1.

A full description of the Product Explorer will be given in Section [7.3.4](#)

3. Type:

```
esoreflex kmos &
```

to launch the `kmos esoreflex` workflow. The `kmos` workflow window will appear (Fig. 4.2). Please configure the set-up directories `ROOT_DATA_DIR`, `RAW_DATA_DIR`, and other workflow parameters as needed. Just double-click on them, edit the content, and press . Remember to specify the same `<database_path>` as for the Product Explorer, if it has been opened at step #2, to synchronize the two processes.

4. (Recommended, but not mandatory) On the main `esoreflex` menu set `Tools -> Animate at Runtime` to 1 in order to highlight in red active actors during execution.

5. Press the button to start the workflow. First, the workflow will highlight and execute the `Initialise` actor, which among other things will clear any previous reductions if required by the user (see Section 6.1).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 29 of 81 |

Secondly, if set, the workflow will open the Product Explorer, allowing the user to inspect previously reduced datasets (see Section 7.3.4 for how to configure this option).

7.3 Workflow Steps

7.3.1 Data Organisation And Selection

The `DataOrganiser` (DO) is the first crucial component of a Reflex workflow. The DO takes as input `RAW_DATA_DIR` and `CALIB_DATA_DIR` and it detects, classifies, and organises the files in these directories and any subdirectories. The output of the DO is a list of “DataSets”. A DataSet is a special Set of Files (SoF). A DataSet contains one or several science (or calibration) files that should be processed together, and all files needed to process these data. This includes any calibration files, and in turn files that are needed to process these calibrations. Note that different DataSets might overlap, i.e. some files might be included in more than one DataSet (e.g., common calibration files).

A DataSet lists three different pieces of information for each of its files, namely 1) the file name (including the path), 2) the file category, and 3) a string that is called the “purpose” of the file. The DO uses the OCA⁶ rules to find the files to include in a DataSet, as well as their categories and purposes. The file category identifies different types of files, and it is derived by information in the header of the file itself. A category could for example be `RAW_CALIBRATION_1`, `RAW_CALIBRATION_2` or `RAW_SCIENCE`, depending on the instrument. The purpose of a file identifies the reason why a file is included in a DataSet. The syntax is `action_1/action_2/action_3/ ... /action_n`, where each `action_i` describes an intended processing step for this file (for example, creation of a `MASTER_CALIBRATION_1` or a `MASTER_CALIBRATION_2`). The actions are defined in the OCA rules and contain the recipe together with all file categories required to execute it (and predicted products in case of calibration data). For example, a workflow might include two actions `action_1` and `action_2`. The former creates `MASTER_CALIBRATION_1` from `RAW_CALIBRATION_1`, and the later creates a `MASTER_CALIBRATION_2` from `RAW_CALIBRATION_2`. The `action_2` action needs `RAW_CALIBRATION_2` frames and the `MASTER_CALIBRATION_1` as input. In this case, these `RAW_CALIBRATION_1` files will have the purpose `action_1/action_2`. The same DataSet might also include `RAW_CALIBRATION_1` with a different purpose; irrespective of their purpose the file category for all these biases will be `RAW_CALIBRATION_1`.

The Datasets created via the `DataOrganiser` will be displayed in the `DataSet Chooser`. Here the users have the possibility to inspect the various datasets and decide which one to reduce. By default, DataSets that have not been reduced before are highlighted for reduction. Click either `Continue` in order to continue with the workflow reduction, or `Stop` in order to stop the workflow. A full description of the `DataSet Chooser` is presented in Section 7.3.2.

Once the `Continue` is pressed, the workflow starts to reduce the first selected DataSet. Files are broadcasted according to their purpose to the relevant actors for processing.

The categories and purposes of raw files are set by the DO, whereas the categories and purpose of products generated by recipes are set by the `RecipeExecuter`. The file categories are used by the `FitsRouter`

⁶OCA stands for OrganisationClassificationAssociation and refers to rules, which allow to classify the raw data according to the contents of the header keywords, organise them in appropriate groups for processing, and associate the required calibration data for processing. They can be found in the directory `<install_dir>/share/esopipes/<pipeline-version>/reflex/`, carrying the extension `.oca`

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 30 of 81 |

to send files to particular processing steps or branches of the workflow (see below). The purpose is used by the `SofSplitter` and `SofAccumulator` to generate input SoFs for the `RecipeExecutor`. The `SofSplitter` and `SofAccumulator` accept several SoFs as simultaneous input. The `SofAccumulator` creates a single output SoF from the inputs, whereas the `SofSplitter` creates a separate output SoF for each purpose.

7.3.2 DataSetChooser

The `DataSetChooser` displays the `DataSets` available in the “Select Data Sets” window, activating vertical and horizontal scroll bars if necessary (Fig. 4.1).

Some properties of the `DataSets` are displayed: the name, the number of files, a flag indicating if it has been successfully reduced (a green OK), if the reduction attempts have failed or were aborted (a red FAILED), or if it is a new dataset (a black "-"). The column "Descriptions" lists user-provided descriptions (see below), other columns indicate the instrument set-up and a link to the night log.

Sometimes you will want to reduce a subset of these `DataSets` rather than all `DataSets`, and for this you may individually select (or de-select) `DataSets` for processing using the tick boxes in the first column, and the buttons `Deselect All` and `Select Complete` at the bottom, or configure the “Filter” field at the bottom left. Available filter options are: "New" (datasets not previously reduced will be selected), "Reduced" (datasets previously reduced will be selected), "All" (all datasets will be selected), and "Failed" (dataset with a failed or aborted reduction will be selected).

You may also highlight a single `DataSet` in blue by clicking on the relevant line. If you subsequently click on `Inspect Highlighted`, then a “Select Frames” window will appear that lists the set of files that make up the highlighted `DataSet` including the full filename⁷, the file category (derived from the FITS header), and a selection tick box in the right column. The tick boxes allow you to edit the set of files in the `DataSet` which is useful if it is known that a certain calibration frame is of poor quality (e.g: a poor raw flat-field frame). The list of files in the `DataSet` may also be saved to disk as an ASCII file by clicking on `Save As` and using the file browser that appears.

By clicking on the line corresponding to a particular file in the “Select Frames” window, the file will be highlighted in blue, and the file FITS header will be displayed in the text box on the right, allowing a quick inspection of useful header keywords. If you then click on `Inspect`, the workflow will open the file in the selected FITS viewer application defined by the workflow parameter `FITS_VIEWER`.

To exit from the “Select Frames” window, click `Continue`.

To add a description of the reduction, press the button `...` associated with the field "Add description to the current execution of the workflow" at the bottom right of the Select Dataset Window; a pop up window will appear. Enter the desired description (e.g. "My first reduction attempt") and then press `OK`. In this way, all the datasets reduced in this execution, will be flagged with the input description. Description flags can be visualized in the `SelectFrames` window and in the `ProductExplorer`, and they can be used to identify different reduction strategies.

To exit from the “Select DataSets” window, click either `Continue` in order to continue with the workflow reduction, or `Stop` in order to stop the workflow.

⁷keep the mouse pointer on the file name to visualize the full path name.

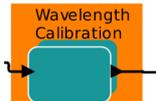
| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 31 of 81 |

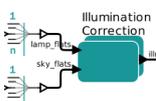
7.3.3 The workflow data-reduction cascade

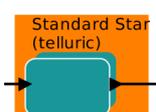
The present KMOS workflow is designed to process the datasets according to a specific data reduction cascade. This cascade triggers a series of pipeline recipes, which are associated to the following composite actors:

- 

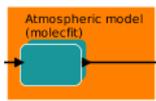
MASTER_DARK: it executes the recipe *kmos_dark*. It processes the dark frames and creates a master dark frame and bad pixel mask.
- 

LAMP_FLAT: it executes the recipe *kmos_flat*. It processes the lamp flats. It creates a master flat frame and calibration frames needed for spatial calibration for all three detectors. It also updates the bad pixel mask.
- 

Wavelength Calibration: it executes the recipe *kmos_wave_cal*. It processes the arc frames and creates the wavelength calibration frame needed for all three detectors.
- 

Illumination: it executes the recipe *kmos_illumination*. It computes the frame to correct for spatial non-uniformity from twilight sky spectra if present (category: FLAT_SKY) and internal flat fields (category: FLAT_ON). In the first case, only one rotator angle is processed as the Twilight flats are executed with one angle only; in the second case, all the 6 angles foreseen by the calibration plan are considered. The user decides which illumination correction to produce by setting the workflow parameter **UseSkyFlats?** (Section 6.2).
- 

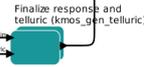
Standard Star: it executes the recipe *kmos_std_star*. It processes standard star observations and generates a response plus telluric correction curve. The category of this output file is TELLURIC and it is stored in the PRO.CATG header keyword. It is possible to provide a file with category TELLURIC_CORR along with the raw data. If this external file is present in the dataset, it will be used to process the science data instead of the one produced from the raw standard stars. The user-provided file must have the same structure than the telluric.fits product of *kmos_std_star*.

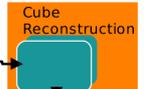
It is an optional recipe, and it is triggered only if standard/telluric stars observations are available in the dataset.
- 

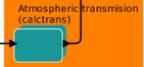
Atmospheric fit: it executes the recipe *kmos_molecfit_model*. It fits atmospheric models to the 1D spectrum of the standard star for each IFU containing data (category: STAR_SPEC, produced

| | | | |
|-----|----------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 32 of 81 |

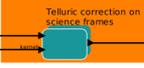
by the Standard Star actor). The fit is performed if the workflow parameter **Use molecfit/calctrans** is set to “standard” and if standard star observations are available in the dataset. The algorithm that is used by the recipe is `molecfit`; we refer the user to the Molecfit user manual for further information. The products of this recipe are `ATMOS_PARM` (a table with atmospheric parameters such as pressure, temperature, and humidity), `BEST_FIT_PARM` (a table containing the best fit atmospheric and instrumental parameters) `BEST_FIT_MODEL` (the best fit model to the data).

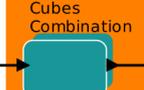
- 
`kmos_gen_telluric`. It finalizes the creation of the instrument response function and, eventually, the telluric correction. Its behavior is controlled by the combination of the global parameters **telluric** and **response correction**, **Use molecfit/calctrans**, and **Apply IFU-customized response? (yes/no)** (Section 6.2).

- 
`Cube Reconstruction`: it executes the recipe `kmos_sci_red`. It processes the raw science frames and creates a reduced and sky-subtracted datacube per IFU. The products are or are not flux calibrated, corrected for instrument response and/or telluric absorption depending on the content of the dataset and the value of the **telluric** and **response correction** workflow parameter (see Section 6.2).

- 
`Atmospheric transmission`: it executes the recipe `kmos_molecfit_calctrans`. The algorithm that is used by the recipe is `calctrans`; we refer the user to the Molecfit user manual for further information. The actor is triggered if the workflow parameter **Use molecfit/calctrans** is set to “standard” and if standard star observations are available in the dataset. It uses the atmospheric model computed by `kmos_molecfit_model` and one reduced scientific observation to compute the atmospheric transmission over all the wavelength range of the scientific data. It accounts for the difference in airmass between the scientific observations and the observations of the standard star. If a kernel library is provided, it accounts for the difference in spectral resolution between the IFU that was used to observe the standard star, and the IFU for which the telluric correction has to be computed for.

- 
`Telluric correction from standard star`: it executes the recipe `kmos_molecfit_correct`. It applies the telluric correction computed in `Atmospheric transmission` to the scientific datacubes produced by `kmos_sci_red`. The recipe is triggered if the workflow parameter **Use molecfit/calctrans** is set to “standard” and if standard star observations are available in the dataset. Note that, if **Use molecfit/calctrans** is set to “standard”, the telluric correction computed by `kmos_std_star` is not used by `kmos_sci_red` and the scientific datacubes are corrected for telluric features by the actor `Telluric correction`.

- 
`Telluric correction on science frames`. This actor is executed only if **Use molecfit/calctrans = science**. It will run `molecfit/calctrans` algorithms and corresponding interactive actors on science data. See Section 9 for further information.

- 
`Cubes combination`: it executes the recipe `kmos_combine`. It combines datacubes be-

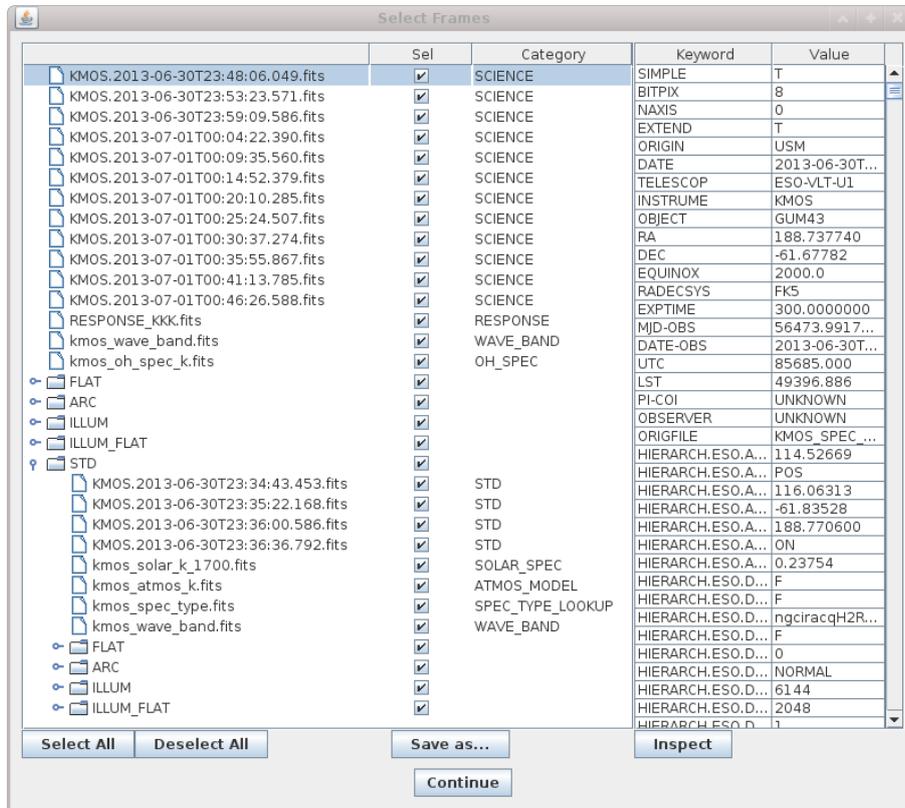


Figure 7.1: The “Selected Frame” pop-up window, obtained after pressing Inspect highlighted in the Select Datasets window (Fig. 4.1).

longing to the same target (or the same IFU) into a final datacube (one datacube per object, or IFU).

We refer the user to the KMOS pipeline manual for a complete description of the recipes and their parameters.

Some of the above actors trigger an interactive window, allowing the used to inspected the products and, eventually, to re-run the associated pipeline recipe with modified parameters. They are identified by the orange box and will be discussed in details in Section 8.

As noted in the Quick Start Section 4, the workflow will then proceed through its remaining processing steps and write out all pipeline products to the end products directory (specified by the parameter END_PRODUCTS_DIR under “Setup Directories” in the workflow canvas). The science data products from the tutorial data set are summarized in section 7.4. The intermediate pipeline calibration products can be found in subdirectories of the TMP_PRODUCT_DIR.

7.3.4 The ProductExplorer

The ProductExplorer is an interactive component in the esoreflex workflow whose main purpose is to list the final products with the associated reduction tree for each dataset and for each reduction attempt (see Fig. 4.4).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 34 of 81 |

Configuring the ProductExplorer

You can configure the ProductExplorer GUI to appear after or before the data reduction. In the latter case you can inspect products as reduction goes on.

1. To display the ProductExplorer GUI at the end of the data reduction:

- Click on the global parameter “ProductExplorerMode” before starting the data reduction. A configuration window will appear allowing you to set the execution mode of the Product Explorer. Valid options are:
 - "Triggered" (default). This option opens the ProductExplorer GUI when all the selected datasets have been reduced.
 - "Enabled". This option opens the ProductExplorer GUI at the end of the reduction of each individual dataset.
 - “Disable”. This option does not display the ProductExplorer GUI.
- Press the  button to start the workflow.

2. To display the ProductExplorer GUI “before” starting the data reduction:

- double click on the composite Actor "Inspect previously reduced data". A configuration window will appear. Set to "Yes" the field "Inspect previously reduced data (Yes/No)". Modify the field "Continue reduction after having inspected the previously reduced data? (Continue/Stop/Ask)". "Continue" will continue the workflow and trigger the DataOrganizer. "Stop" will stop the workflow; "Ask" will prompt another window deferring the decision whether continuing or not the reduction after having closed the Product Explorer.
- Press the  button to start the workflow. Now the ProductExplorer GUI will appear before starting the data organization and reduction.

Exploring the data reduction products

The left window of the ProductExplorer GUI shows the executions for all the datasets (see Fig. 4.4). Once you click on a dataset, you get the list of reduction attempts. Green and red flags identify successful or unsuccessful reductions. Each reduction is linked to the “Description” tag assigned in the “Select Dataset” window.

1. To identify the desired reduction run via the “Description” tag, proceed as follows:

- Click on the symbol at the left of the dataset name. The full list of reduction attempts for that dataset will be listed. The column Exec indicates if the reduction was successful (green flag: "OK") or not (red flag: "Failed").
- Click on the entries in the field "Description" to visualize the description you have entered associated to that dataset on the Select Dataset window when reducing the data.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 35 of 81 |

- Identify the desired reduction run. All the products are listed in the central window, and they are organized following the data reduction cascade.

You can narrow down the range of datasets to search by configuring the field "Show" at the top-left side of the ProductExplorer (options are: "All", "Successful", "Unsuccessful"), and specifying the time range (Last, all, From-to).

2. To inspect the desired file, proceed as follows:

- Navigate through the data reduction cascade in the ProductExplorer by clicking on the files.
- Select the file to be inspected and click with the mouse right-hand button. The available options are:
 - Options available always:
 - * Copy full path. It copies the full name of the file onto the clipboard. Shift+Ctrl+v to past it into a terminal.
 - * Inspect Generic. It opens the file with the fits viewer selected in the main workflow canvas.
 - * Inspect with. It opens the file with an executable that can be specified (you have to provide the full path to the executable).
 - Options available for files in the TMP_PRODUCTS_DIR directory only:
 - * command line. Copy of the environment configuration and recipe call used to generate that file.
 - * Xterm. It opens an Xterm at the directory containing the file.
 - Options available for products associated to interactive windows only:
 - * Display pipeline results. It opens the interactive windows associated to the recipe call that generated the file. Note that this is for visualization purposes only; the recipe parameters cannot be changed and the recipe cannot be re-run from this window.

7.4 Description of the reduced KMOS data

A number of intermediate pipeline products from the tutorial data set can be found in subdirectories of the TMP_PRODUCT_DIR. We refer to the kmos pipeline manual for a description of the products of each recipe.

The final products of the reduction pipeline can be found in the directory END_PRODUCT_DIR that is defined in the “Setup Directories” section at the top of the workflow.

The science data products from the tutorial data sets are summarized below, whereas their description will be done in Section 7.4.1:

- Science products for dataset: KMOS.2013-06-30T23:48:06.049_tpl
 - SCI-GUM43_COMBINED_CUBE_XXX.fits (**PRO.CATG: COMBINED_CUBE**). It is the re-constructed science cube for object name 'XXX'. In this dataset, XXX = 001, 002,003, 004, 007, 010 ,014, 018, 020, 021, 027, 029, 030, 048, 058, 069, 100, 101, 103.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 36 of 81 |

- SCI-GUM43_IDP_COMBINED_CUBE_XXX.fits (**PRO.CATG: IDP_COMBINED_CUBE**). Reconstructed science cube for object XXX , with 'XXX' as above, with format compatible with ESO Phase3 archive standards.
 - SCI-GUM43_COMBINED_IMAGE_XXX.fits (**PRO.CATG: COMBINED_IMAGE**). It is the image of the field of view for object name 'XXX', with 'XXX' as above, obtained by integrating the corresponding datacube along the wavelength direction.
 - SCI-GUM43_EXP_MASK_XXX.fits (**PRO.CATG: EXP_MASK**). It indicates the number of exposures that contributed to each spaxel in the field of view, for object name 'XXX', with 'XXX' as above.
 - SCI-GUM43_SINGLE_CUBES_KMOS.<time-stamp>.fits (**PRO.CATG: SINGLE_CUBES**). They are the 9 reconstructed cubes obtained from each individual exposure. Each exposure is identified by its <time-stamp>, that matches the one of the raw science frame.
- Science products for dataset KMOS.2015-05-18T05:38:12.110_tpl:
 - NGC6240-MOSAIC8-H_1_COMBINED_CUBE_ mapping.fits (**PRO.CATG: COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the observing block.
 - NGC6240-MOSAIC8-H_1_IDP_COMBINED_CUBE_ mapping.fits (**PRO.CATG: IDP_COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the observing block, with format compatible with ESO Phase3 archive standards.
 - NGC6240-MOSAIC8-H_1_COMBINED_IMAGE_ mapping.fits (**PRO.CATG: COMBINED_IMAGE**). It is the image of the field of view, obtained by integrating the corresponding datacube along the wavelength direction.
 - NGC6240-MOSAIC8-H_1_EXP_MASK_ mapping.fits (**PRO.CATG: EXP_MASK**). It indicates the number of exposures that contributed to each spaxel in the field of view.
 - NGC6240-MOSAIC8-H_1_SINGLE_ CUBES_KMOS.<time-stamp>.fits (**PRO.CATG: SINGLE_CUBES**). They are the 3 reconstructed cubes obtained from each individual exposure in that OB. Each exposure is identified by its <time-stamp>, that matches the one of the raw science frame.
 - Science products for dataset KMOS.2015-05-18T06:47:33.130_tpl:
 - NGC6240-MOSAIC8-H_2_COMBINED_CUBE_ mapping.fits (**PRO.CATG: COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the observing block.
 - NGC6240-MOSAIC8-H_2_IDP_COMBINED_CUBE_ mapping.fits (**PRO.CATG: IDP_COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the observing block, with format compatible with ESO Phase3 archive standards.
 - NGC6240-MOSAIC8-H_2_COMBINED_IMAGE_ mapping.fits (**PRO.CATG: COMBINED_IMAGE**). It is the image of the field of view, obtained by integrating the corresponding datacube along the wavelength direction.
 - NGC6240-MOSAIC8-H_2_EXP_MASK_ mapping.fits (**PRO.CATG: EXP_MASK**). It indicates the number of exposures that contributed to each spaxel in the field of view.
 - NGC6240-MOSAIC8-H_2_SINGLE_CUBES_KMOS.<time-stamp>.fits (**PRO.CATG: SINGLE_CUBES**). They are the 3 reconstructed cubes obtained from each individual exposure in that

| | | | |
|------------|-----------------------------|---------------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 37 of 81 |

OB. Each exposure is identified by its <time-stamp>, that matches the one of the raw science frame.

- Science products for dataset `KMOS.2015-05-29T03:18:11.720_tpl`:
 - `NGC6240-MOSAIC8-H_3_COMBINED_CUBE_mapping.fits` (**PRO.CATG: COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the observing block.
 - `NGC6240-MOSAIC8-H_3_IDP_COMBINED_CUBE_mapping.fits` (**PRO.CATG: IDP_COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the observing block, with format compatible with ESO Phase3 archive standards.
 - `NGC6240-MOSAIC8-H_3_COMBINED_IMAGE_mapping.fits` (**PRO.CATG: COMBINED_IMAGE**). It is the image of the field of view, obtained by integrating the corresponding datacube along the wavelength direction.
 - `NGC6240-MOSAIC8-H_3_EXP_MASK_mapping.fits` (**PRO.CATG: EXP_MASK**). It indicates the number of exposures that contributed to each spaxel in the field of view.
 - `NGC6240-MOSAIC8-H_3_SINGLE_CUBES_KMOS.<time-stamp>.fits` (**PRO.CATG: SINGLE_CUBES**). They are the 3 reconstructed cubes obtained from each individual exposure in that OB. Each exposure is identified by its <time-stamp>, that matches the one of the raw science frame.
- Science products for dataset `KMOS.2015-05-18T05:38:12.110_combine_OBs`:
 - `NGC6240-MOSAIC8-H_2_COMBINED_CUBE_mapping.fits` (**PRO.CATG: COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the 3 observing blocks.
 - `NGC6240-MOSAIC8-H_2_IDP_COMBINED_CUBE_mapping.fits` (**PRO.CATG: IDP_COMBINED_CUBE**). Combined cube (mosaic) obtained from all the exposures in the 3 observing blocks, with format compatible with ESO Phase3 archive standards.
 - `NGC6240-MOSAIC8-H_2_COMBINED_IMAGE_mapping.fits` (**PRO.CATG: COMBINED_IMAGE**). It is the image of the field of view, obtained by integrating the corresponding datacube along the wavelength direction.
 - `NGC6240-MOSAIC8-H_2_EXP_MASK_mapping.fits` (**PRO.CATG: EXP_MASK**). It indicates the number of exposures that contributed to each spaxel in the field of view.
 - `NGC6240-MOSAIC8-H_N_SINGLE_CUBES_KMOS.<time-stamp>.fits` (**PRO.CATG: SINGLE_CUBES**). They are the 3 reconstructed cubes obtained from each individual exposure in the N=1,2,3 OBs. Each exposure is identified by its <time-stamp>, that matches the one of the raw science frame.
- Science products for dataset `MACS0416-K-OB1_SINGLE_CUBES_KMOS.2017-01-02T02:58:12.544_reduced_OBs`:
 - `MACS0416-K-OB1_COMBINED_CUBE_MACSJ0416_XXX.fits` (**PRO.CATG: COMBINED_CUBE**). It is the reconstructed science cube for object name 'XXX'. In this dataset, XXX = star, star_sky, ARC1a, ARC1a_sky, ARC1b, ARC1b_sky, ARC2, ARC3, ARC4, ARC4_sky, ARC5, ID01_sky, ID02, ID02_sky, ID03, ID04, ID04_sky, ID05, ID07, ID07_sky, ID08_sky, LAE1a, LAE1b_sky.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 38 of 81 |

- MACS0416-K-OB1_IDP_COMBINED_CUBE_MACSJ0416_XXX.fits (**PRO.CATG: IDP_COMBINED_CUBE**). Reconstructed science cube for object XXX, with 'XXX' as above, with format compatible with ESO Phase3 archive standards.
- MACS0416-K-OB1_COMBINED_IMAGE_MACSJ0416_XXX.fits (**PRO.CATG: COMBINED_IMAGE**). It is the image of the field of view for object name 'XXX', with 'XXX' as above, obtained by integrating the corresponding datacube along the wavelength direction.
- MACS0416-K-OB1_EXP_MASK_MACSJ0416_XXX.fits (**PRO.CATG: EXP_MASK**). It indicates the number of exposures that contributed to each spaxel in the field of view, for object name 'XXX', with 'XXX' as above.

7.4.1 Description of final products

In this section we provide a short description of the workflow products that are stored in the `reflex_end_products` directory. They are produced by the pipeline recipes `kmoss_sci_red` and `kmoss_combine`. For further information, please consult the KMOS pipeline manual. In the list, the products are identified by their PRO.CATG keyword.

- **COMBINED_CUBE**. Product of the recipe `kmoss_combine`, which is triggered by the Cubes Combination actor. Its a 3 extensions fits file. The first extension contains the primary header. The second extension contains the fully reduced datacube, obtained by combining the reconstructed cubes of the IFUs in each exposure that belong to the same object. The datacube is a three-dimensional array (x, y, λ) , where the first two dimensions represent the spatial coordinates on the sky (RA and DEC, respectively). The third dimension is wavelength. Therefore for a given (x, y) , the datacube shows the spectrum obtained at those RA and DEC coordinates on the sky.

Units are $\text{ergs cm}^{-2} \text{ \AA}^{-1} \text{ s}^{-1}$ for flux calibrated data (as in the nod-to-sky demo dataset) or ADU s^{-1} for non flux calibrated data.

The third extension contains the error cube associated to the first data extension. The error vector in each spaxel is computed by the standard deviation of the input spectra (if at least 3). Each dataset produces as many **COMBINED_CUBE** as targeted objects. In the case of mosaic exposure, only one **COMBINED_CUBE** is produced.

COMBINED_CUBE are produced by combining the **SCI_RECONSTRUCTED** (products of the recipe `kmoss_sci_red`, see below) or the **SINGLE_CUBES** (products of the recipe `kmoss_molecfits_correct`). The combination is automatically done either i) by combining the extensions of the individual **SINGLE_CUBES** files that belong to the same target (for nod-to-sky exposures); or ii) by combining all the **SINGLE_CUBES** for mosaic exposures.

- **IDP_COMBINED_CUBE**. Same as before **COMBINED_CUBE**, but with header keywords that compliant with ESO Phase3 archive service. In addition, the error extension is computed spaxel by spaxel by a running standard deviation of the data spectrum.
- **COMBINED_IMAGE**. Product of the recipe `kmoss_combine`, which is triggered by the Cubes Combination actor. Reconstructed image associated to the **COMBINED_CUBE** product. Each dataset produces one **COMBINED_IMAGE** for each **COMBINED_CUBE**.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 39 of 81 |

Units are $\text{ergs cm}^{-2} \text{s}^{-1}$ for flux calibrated data (as in the nod-to-sky demo dataset) or ADU s^{-1} for non flux calibrated data.

- **EXP_MASK.** Product of the recipe *kmos_combine*, which is triggered by the Cubes Combination actor. Exposure map associated to a co-added datacube. The value at each pixel tells with how many exposures contributed to that position on the sky. Each dataset produces one exposure map for each combined cube.
- **SCI_RECONSTRUCTED.** Product of the recipe *kmos_sci_red*, which is triggered by the Cube Reconstruction actor. These files contain, in each extension, a reconstructed datacube associated to a given IFU (which is associated to a target or to a specific location of a mosaic). As in the previous case, each datacube has 3 dimensions (two spatial and one wavelength). The combination of all the reconstructed cubes associated to the same target produces the **COMBINED_CUBE**. Each dataset produces as many **SCI_RECONSTRUCTED** as exposures in that dataset, that are not dedicated only to sky. Units are $\text{ergs cm}^{-2} \text{\AA}^{-1} \text{s}^{-1}$ for flux calibrated data (as in the nod-to-sky demo dataset) or ADU s^{-1} for non flux calibrated data.
- **SINGLE_CUBES.** Product of the recipe *kmos_molecfi_t_correct*. They have the same format and meaning of **SCI_RECONSTRUCTED**, but they are divided by the telluric correction computed by *kmos_molecfi_calctrans*.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 40 of 81 |

8 Optimising Your Results Through Workflow Interaction

In this Section, we use the information from Section 4 along with one KMOS demo data supplied with the installation to illustrate how to optimise the scientific products in terms of quality and signal-to-noise ratio.

The examples shown in this Section refer to the first demo-dataset (object name: Gum 43); information related to the second dataset will be added whenever relevant.

This is work in progress and the contents of this section will grow as we gain experience and add functionality. Optimising the results is achieved by interaction with the workflow actors via interactive windows displayed at key data reduction points in the data flow, which enable iteration of certain recipes in order to obtain better results.

We recommend that the user has already carried out the reductions for all demo DataSets as described in Section 4, although this is not a pre-requisite to following this section. By doing this, the user will be taking advantage of the workflow Lazy Mode, with minimal waiting time between various pipeline recipe executions.

To start the process, execute the first 5 steps indicated in the Quick Start Section 4.

As discussed previously, the first operation the KMOS workflow does is to group the raw data (science, raw calibrations, and static calibrations) into datasets to be reduced. The datasets will be then processed in series.

The second operation is to direct all the files of a dataset to the correct pipeline recipe, so that the data reduction could start and the various recipes can be triggered in the correct order with the correct set of input files. The data reduction cascade operated by the KMOS workflow triggers the recipes according to the data reduction cascade outlined in Section 7.3.3.

Some of the recipes executed by the workflow trigger an interactive window, allowing the user to inspect the products and, eventually, to re-run the recipe with modified parameters. Section 8.1 describes the common properties of the interactive windows. Sections (8.2 – 8.5) deal with the individual interactive windows.

8.1 General characteristics of the interactive windows

For each of the interactive windows described above, the panel of buttons at the very top-left of the window may be used to manipulate the displayed plots. The buttons have the following functions:

-  - Reset all the plot ranges to their original values.
-  - Undo the last modification of the plot ranges (if possible).
-  - Redo the next modification of the plot ranges (if possible).
-  - Selecting this button allows the user to use the mouse to shift the plot ranges by left-clicking on the target plot canvas and then dragging the mouse around while keeping the left mouse button held down, and releasing when ready.
-  - Selecting this button allows the user to zoom in on each plot by left-clicking on the target plot canvas to mark the top-left corner of a rectangle and then dragging the mouse to the bottom-right corner of the

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 41 of 81 |

rectangle and releasing. The plot ranges will then be modified to match the rectangle that was defined. The following constraints can be made by holding simultaneously a key while pressing the left mouse button:

Constrain pan/zoom to x axis hold X
 Constrain pan/zoom to y axis hold Y
 Preserve aspect ratio hold Ctrl

-  - Clicking this button opens a “Configure subplots” window that allows the user to adjust the spacing and positioning of the individual plots.
-  - Clicking this button opens a “Save to file” window which allows the user to save a screenshot of the current interactive window.
-  - Clicking this button allows the user to change the display levels of the 2-dimensional merged image by clicking on pixels within the 2-dimensional image (similar to ds9).

Use these buttons to inspect the plots in the interactive window in more detail.

The right-hand side of the interactive window allows to change the recipe parameters and re-execute the it. One can re-use values for for subsequent recipe executions by clicking the button ‘Use the parameters as initial values in subsequent executions of this recipe’, and then press “Continue WKF”. Note that the changes will be lost once reflex is closed. To save them in reflex, please save the workflow by clicking on “File” and “Export as xml” in the main reflex canvas. .

The bottom right list indicates all the files that were used as input to the interactive window, identified by their category. If the user wants to inspect a particular file or assess the recipe products with another viewer or script, (s)he can copy the full file path from the list.

8.2 Wavelength Calibration

In the KMOS workflow, the interactive actor `Wavelength Calibration` is identifiable by an orange rectangle encompassing the actor name. The interactive mode is enabled by default. Should you wish to change that use `Open Actor` to get access to the the components of the interactive actors, then double-click on the composite actors, setting the “EnableInteractivity” parameter to `false`, and clicking Commit to save the changes to the workflow.

Figure 8.1 shows the interactive window that will pop-up at the end of the execution of the `kmoss_wave_cal` pipeline routine. The image panels at the top of the window show the reconstructed arc frames for each of the three detectors at one of the six rotator position angles. To view the reconstructed arc frames at another rotator angle (either 0, 60, 120, 180, 240, or 300 degrees), simply click on the ovals in the *Angle selection* box. Exposures at different angles are taken to account for the instrument flexures. When applying the wavelength correction to a dataset, the KMOS pipeline automatically selects the calibration with the closest angle. The output file with the reconstructed arc frames has 18 extensions (3 extensions per detector, times 6 angles). For each of the 18 reconstructed arc frames, the relative average offset of the reconstructed arc lines are plotted in units of pixels for each of the Argon and Neon arc lines (bottom two plots). One-half of the length of the error bars associated with each data point gives the average FWHM (in pixels) for each arc line.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 42 of 81 |

On the right-hand edge of the interactive window the user may modify a number of parameters to improve the wavelength calibration. If parameters have been changed, then clicking on the `Re-run Recipe` will re-execute the `kmos_wave_cal` pipeline routine.

Other recipe parameters that are not listed in the interactive window can be modified by opening the `Wavelength Calibration` actor, double clicking on the `RecipeExecuter` associated to the `kmos_wave_cal` recipe (the green box named `kmos_wave_cal_1`), and editing the desired parameter in the apposite field. It is advisable not to change the values that have the `PORT` value. Indeed, these parameters get their value from the interactive window; hardcoding a value will prevent them of being edited from the interactive window anymore.

It could happen that the Argon lines have higher absolute residuals than Neon lines, as shown in figure 8.1 for the first demo dataset. This might be due to the fact that Ne lines are more numerous and on average more intense than Argon lines, therefore they drive the wavelength calibration. The difference is less than a pixel and negligible if compared to uncertainties. However, the effect can be mitigated by increasing the `-order` parameter to 7 and press the `Re-run Recipe` button.

Note: `-order=0` uses the grism hard-coded value, which is 6 for the H, K, and YJ grisms, 4 for IZ grism, and 5 for HK grism.

| | | | |
|------------|----------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 43 of 81 |

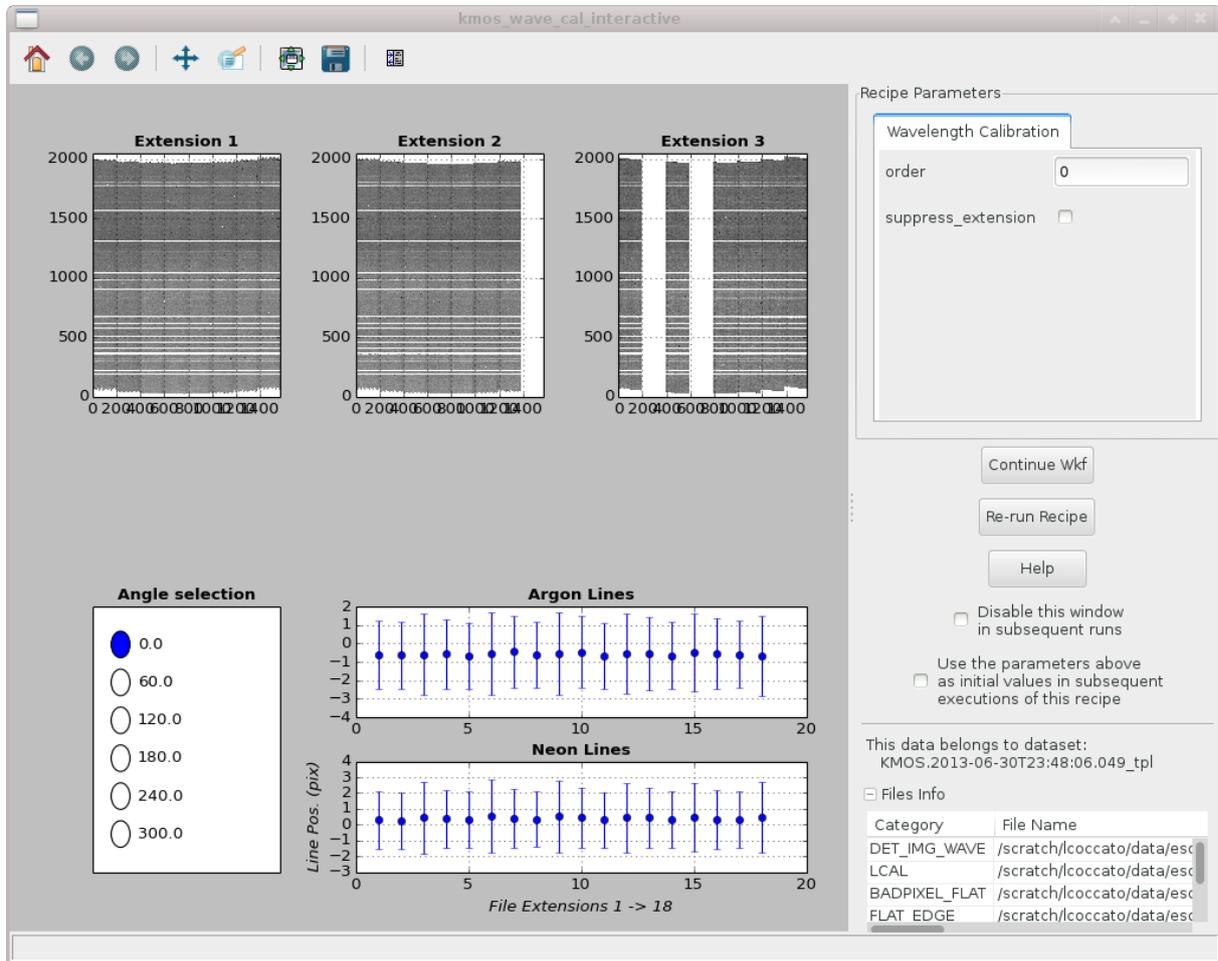


Figure 8.1: The interactive window for the Wavelength Calibration actor as displayed for the KMOS tutorial data set..

8.3 Telluric Standard Star Calibration

In the KMOS workflow, the interactive actor `Standard Star` is identifiable by an orange rectangle encompassing the actor name. The interactive mode is enabled by default. Should you wish to change that use `Open Actor` to get access to the the components of the interactive actors, then double-click on the composite actors, setting the “EnableInteractivity” parameter to `false`, and clicking `Commit` to save the changes to the workflow.

Figure 8.2 shows the interactive window that will pop-up at the end of the execution of the `kmos_std_star` pipeline routine. Note that for the Orion nebula dataset, there are no standard stars, therefore the recipe and the interactive window are not triggered. The image panel at the top-right of the window shows the median collapsed data cube (pipeline product: `STD_IMAGE`) for a single IFU. The total number of IFU’s dedicated to observing a telluric standard star are listed to the left of this image and clicking on the circle will display the standard star from another IFU. Typically, 3 standard stars are observed (one per detector). If less than one star

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 44 of 81 |

per detector is observed, the final products will not be flux calibrated nor telluric corrected.

At the bottom of the interactive window the extracted standard star spectrum (for the selected IFU) is displayed (pipeline product: `STAR_SPEC`) in flux units of ADU and wavelength units of microns.

On the right-hand edge of the interactive window the user may modify a number of parameters to improve the telluric standard star calibration. If parameters have been changed, then clicking on the `Re-run Recipe` will re-execute the `kmos_std_star` pipeline routine.

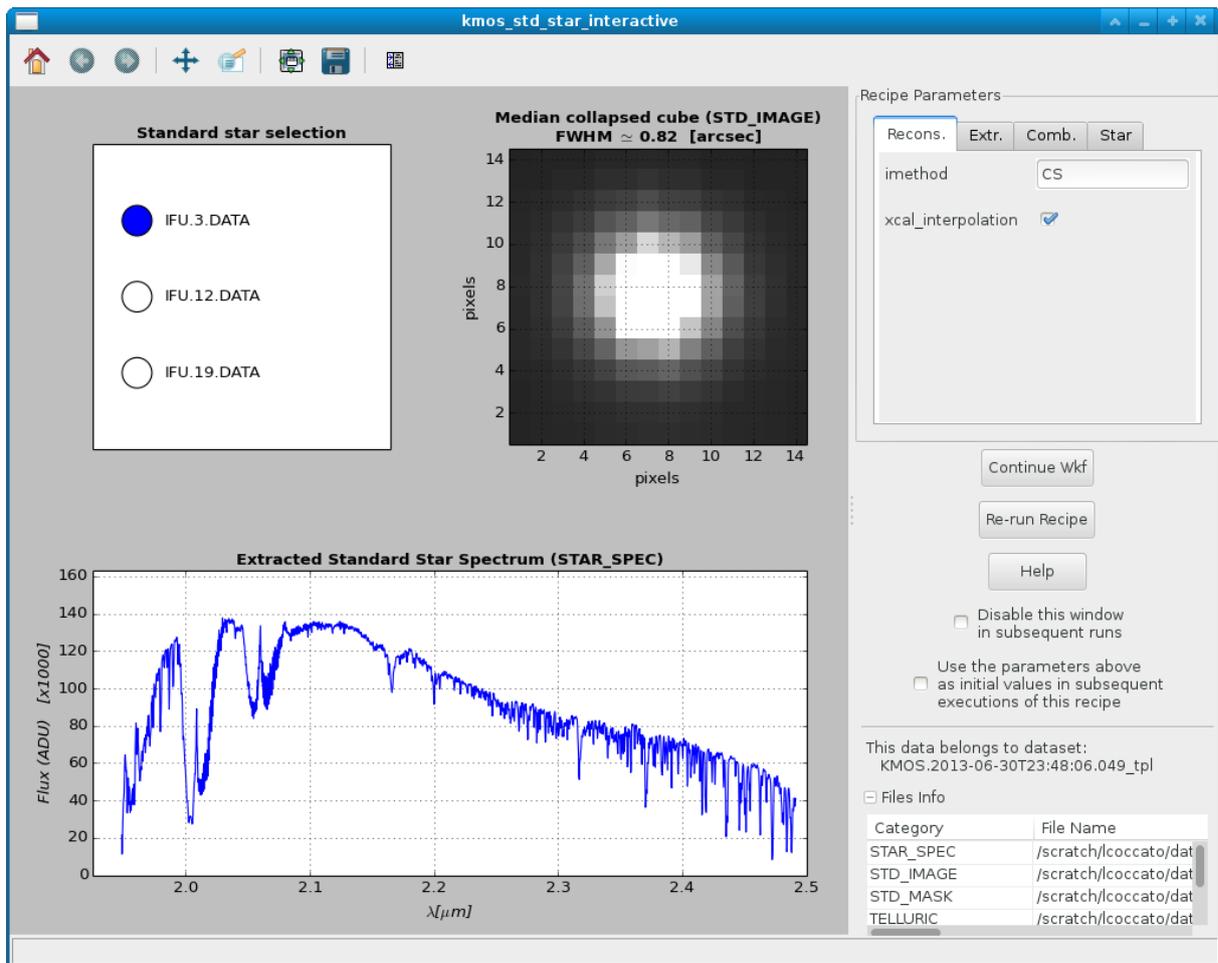


Figure 8.2: The interactive window for the Telluric Standard Star Calibration actor as displayed for the KMOS tutorial data set.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 45 of 81 |

8.4 Reduction of science frames: reconstruction of datacubes

In the KMOS workflow, the interactive actor `Cube Reconstruction` is identifiable by an orange rectangle encompassing the actor name. The interactive mode is enabled by default. Should you wish to change that use `Open Actor` to get access to the the components of the interactive actors, then double-click on the composite actors, setting the “EnableInteractivity” parameter to `false`, and clicking `Commit` to save the changes to the workflow.

Figure 8.3 shows the interactive window that will pop-up at the end of the execution of the `kmos_sci_red` pipeline routine.

The selection panel at the top of the window shows the list of individual reconstructed science images (pipeline products: `SCI_RECONSTRUCTED`) in the processed data set. Each file can be selected by clicking on the oval with the left mouse button.

The central selection panel summarizes the status of each individual IFU for the selected frame. IFU’s for which an extracted spectrum can be viewed are shown as `Active` and are plotted in green. The other IFU statuses, in `Collision`, `NotInPAF`, `Empty`, `Locked` and `NotInPAF&Locked` are shown in cyan, blue, black, magenta, and red, respectively. These IFU’s cannot be selected. To select an `Active` IFU the user should use the centre mouse button. The selected IFU is then shown with a bar beneath its numbered box.

If you drag the mouse over an IFU you will be prompted with the name of the target observed with that IFU. *Warning:* it might be necessary to drag the mouse over the Spectrum plot (see below) to refresh the information.

At the bottom of the interactive window the extracted science frame spectrum (for the selected IFU) is displayed with wavelength units of microns. The flux is shown as a blue line, while the location of sky emission lines (*OH*) are marked in red. Currently, the extracted spectrum is derived by averaging the entire cube along the spatial dimensions and then plotted as a function of wavelength. To the right of the extracted spectrum is a median collapsed image of the reconstructed science data cube.

On the right-hand edge of the interactive window the user may modify a number of parameters to improve the science frame reduction. Other recipe parameters that are not present in the interactive window can be modified by opening the `Cube reconstruction` actor, double clicking on the `RecipeExecuter` associated to the `kmos_sci_red` recipe (the green box named `kmos_sci_red_1`), and editing the desired parameter in the apposite field. It is advisable not to change the values that have the `PORT` value. Indeed, these parameters get their value from the interactive window; hardcoding a value will prevent them of being edited from the interactive window anymore.

8.4.1 Flux calibration

The flux calibration of the scientific data is performed by the `Cube Reconstruction` actor (`kmos_sci_red` recipe) by applying a conversion factor using the photometric zeropoint. The zeropoint is stored in the header of the `TELLURIC`, `TELLURIC_CORR` or `RESPONSE` files (keyword `HIEARCH ESO QC ZPOINT`). The workflow parameter **telluric and response correction** determines which file to provide to `kmos_sci_red`, and therefore which zeropoint to use. More information on the use of **telluric and response correction** can be found in Section 6.2.

The flux units of the products is $erg\ s^{-1}\ cm^{-2}\ \text{\AA}^{-1}$.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 46 of 81 |

Table 8.1: Zero Magnitude Flux Densities (from [3]).

| KMOS band | 2MASS band | Band pass for calibration (μm) | Zero magnitude flux density (F_0) | |
|-----------|------------|---------------------------------------|---|--|
| | | | $W/m^2/\mu m$ | $ph/s/m^2/\mu m$ |
| K | K | 2.028 – 2.290 | 4.283×10^{-10} | 4.65×10^9 |
| HK | H & K | 1.5365 – 1.7875 & 2.028 – 2.290 | 1.133×10^{-9} & 4.283×10^{-10} | 9.47×10^9 4.65×10^9 |
| H | H | 1.5365 – 1.7875 | 1.133×10^{-9} | 9.47×10^9 |
| YJ | J | 1.154 – 1.316 | 3.129×10^{-9} | 1.944×10^9 |
| IZ | — | 0.985 – 1.000 | 7.63×10^{-9} | 3.81×10^{10} |

Note: the flux calibration is performed only if there is a zeropoint for each detector stored in the input file (TELLURIC, TELLURIC_CORR or RESPONSE). For example, when using the TELLURIC computed via *kmos_standard*, this condition is ensured if the input data set contains at least three standard star observation (i.e. one per detector).

In the case the flux calibration cannot be performed, the products will be in units of ADU. In this latter case, the user can apply the flux conversion by himself. If the zeropoint of one detector is missing, the average value stored in the RESPONSE static calibration file can be used. The header keyword HIERARCH ESO QC MAD_THR contains the median absolute deviation of the zeropoints computed so far in the ESO archive, and can be used as an estimate for the zeropoint error.

The zeropoint (QC.ZPOINT) is defined so that:

$$\text{magnitude} = \text{QC.ZPOINT} - 2.5 \log_{10} (\text{counts/second})$$

The magnitude value of the standard star is read from the standard frame header (provided by the user during the preparation of the observing blocks), or entered in the interactive window of the standard star reduction process. To convert the SCI_RECONSTRUCTED data cubes from counts/second into physical units, the following conversion can be applied:

$$\text{Flux density} = \text{counts/second} \times F_0 \times 10^{-0.4 \times \text{QC.ZPOINT}}$$

where, F_0 is the zero magnitude flux density in whatever units have been used. The values of F_0 are listed in table 8.1 for a units of $W/m^2/\mu m$ and $ph/s/m^2/\mu m$. The conversion factor between $W/m^2/\mu m$ and $erg/cm^2/\text{\AA}$ is: $[erg/cm^2/\text{\AA}] = [W/m^2/\mu m]/10$

If you want to derive a line flux, simply integrate the counts over the line, convert the result to a flux density, and then multiply by the spectral size of a pixel (as given by the CDELTA3 keyword in the cubes, or the CDELTA1 keyword in the extracted spectra).

8.4.2 Level correction: removing overscan and intra-slice light

Science data can contain extra counts due to the electronics or other sources. These counts differ from one reading channel to another and can vary intensity across the detector. The kmos pipeline has two main strategies for removing this extra level (“level correction”).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 47 of 81 |

The screenshot shows the 'kmos_sci_red_interactive' window. It features a toolbar at the top and a main workspace divided into several sections:

- Files Selection (Left Mouse Button):** A list of 10 FITS files with corresponding visual representations of the IFU fibers. The files are:
 - SCI_RECONSTRUCTED_KMOS.2013-06-30T23:48:06.049.fits
 - SCI_RECONSTRUCTED_KMOS.2013-06-30T23:59:09.586.fits
 - SCI_RECONSTRUCTED_KMOS.2013-07-01T00:04:22.390.fits
 - SCI_RECONSTRUCTED_KMOS.2013-07-01T00:09:35.560.fits
 - SCI_RECONSTRUCTED_KMOS.2013-07-01T00:20:10.285.fits
 - SCI_RECONSTRUCTED_KMOS.2013-07-01T00:25:24.507.fits
 - SCI_RECONSTRUCTED_KMOS.2013-07-01T00:30:37.274.fits
 - SCI_RECONSTRUCTED_KMOS.2013-07-01T00:41:13.785.fits
 - SCI_RECONSTRUCTED_KMOS.2013-07-01T00:46:26.588.fits
- IFU Selection (Mouse middle button):** A row of 24 colored buttons representing different IFUs. The colors correspond to their status: Active (green), Locked (magenta), NotInPAF (blue), NotInPAF & Locked (red), and Collision (cyan). Buttons 1-12 are green, 13-16 are magenta, 17 is blue, 18 is red, and 19-24 are cyan.
- Spectrum:** A plot of flux density in $\text{erg sec}^{-1} \text{cm}^{-2} \text{Å}^{-1}$ versus wavelength λ in μm . The x-axis ranges from 2.0 to 2.5 μm . The y-axis ranges from 0.0 to 0.8. A blue line represents the observed spectrum, and a red line represents the OH emission lines.
- Collapsed image:** A 2D image showing the collapsed data for the selected IFUs. The axes are labeled 'pixels' and range from 2 to 14.
- Recipe Parameters:** A panel on the right with tabs for 'Recons.', 'Extr.', and 'Sky Sub.'. The 'Recons.' tab is active, showing parameters: 'imethod' set to 'CS', 'oscan' checked, and 'xcal_interpolation' checked. Below the parameters are buttons for 'Continue Wkf', 'Re-run Recipe', and 'Help'. There are also checkboxes for 'Disable this window in subsequent runs' and 'Use the parameters above as initial values in subsequent executions of this recipe'.
- Dataset and Files Info:** At the bottom right, it states 'This data belongs to dataset: KMOS.2013-06-30T23:48:06.049_tpl' and provides a table of file information.

| Category | File Name |
|-------------------|---------------|
| SCI_RECONSTRUCTED | /scratch/lcoc |

Figure 8.3: The interactive window for the Cube Reconstruction actor as displayed for the KMOS tutorial data set.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 48 of 81 |

The first strategy (overscan) evaluates the extra counts on pre- and over- scan regions; it evaluates the correction for each of the 32 read-out channels independently and for even and odd pixels.

The second strategy (intra slices), uses the non illuminated portion of the detector located in between the various slices and IFUs to evaluate the extra light and remove it. This strategy is to be used only for very faint sources, otherwise the cross-talk effect contaminates the intra-slices region and the pipeline over-correct the data.

These strategies can be selected by setting the `-lcmethod` parameter in the recipe *kmos_sci_red* (actor Cube Reconstruction), see the interactive window 8.3.

The values for the `-lcmethod` are:

- OSCAN It applies the overscan method. This is the default value.
- SLICES_MEAN. It applies the intra slices method. The input file is divided into a grid of 32×16 windows, each of them 64×128 pixels wide. The mean of the non illuminated portions (i.e., intra-slices pixels) of the detector within each window is subtracted from the data. Bad pixels are masked. Intra slice pixels and bad pixels are identified with the calibrations `BADPIXEL_DARK` and `LCORR` produced by the workflow.
- SLICES_MEDIAN. Same as `SLICES_MEAN` but the median is computed instead of the average.
- NONE. No correction is done.

The default strategy is to apply this correction directly with the science recipe *kmos_sci_red*, within the Cube Reconstruction actor. Alternatively, the user has the possibility to run a dedicated interactive recipe (*kmos_level_correct*) and apply the correction before the cube reconstruction. To do this, set the global workflow parameter **Preview Level Correction = true**.

The interactive window 8.4 will pop-up. The plot area allows to select the corrected frame and the detector. The recipe parameter area allows to select the desired strategy. If the correction is allowed here, then the workflow automatically sets the `-lcmethod` in *kmos_sci_red* to `NONE`, to avoid to apply the correction twice. This is the case even if the user sets `-lcmethod=NONE` in *kmos_level_correct*.

If the global workflow parameter **Preview Level Correction = false**, the correction will be done by *kmos_sci_red* with `-lcmethod = OSCAN`.

In general, the default strategy `-lcmethod=OSCAN` within the Cube reconstruction (**Preview Level Correction = false**) is the recommended strategy. Intra-slice correction can improve a bit the overall background subtraction (but not the emission sky lines) and it is recommended only for very faint sources. Indeed, bright sources genera cross-talk between slices, therefore the intra-slice area is contaminated by this source of light in the target exposures (which will be over-corrected) and not in the sky exposures. This will result in a over subtraction of the sky continuum for the slices that are more affected by the cross-talk.

8.4.3 Optimizing the sky removal via workflow parameters

The *kmos_sci_red* recipe triggered by the Cube Reconstruction actor automatically assigns a sky IFU to an object IFU. The adopted criteria is to use the same IFU as the object, which is closest in time. To override the automatic assignment, please refer to Section 8.4.4.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 49 of 81 |

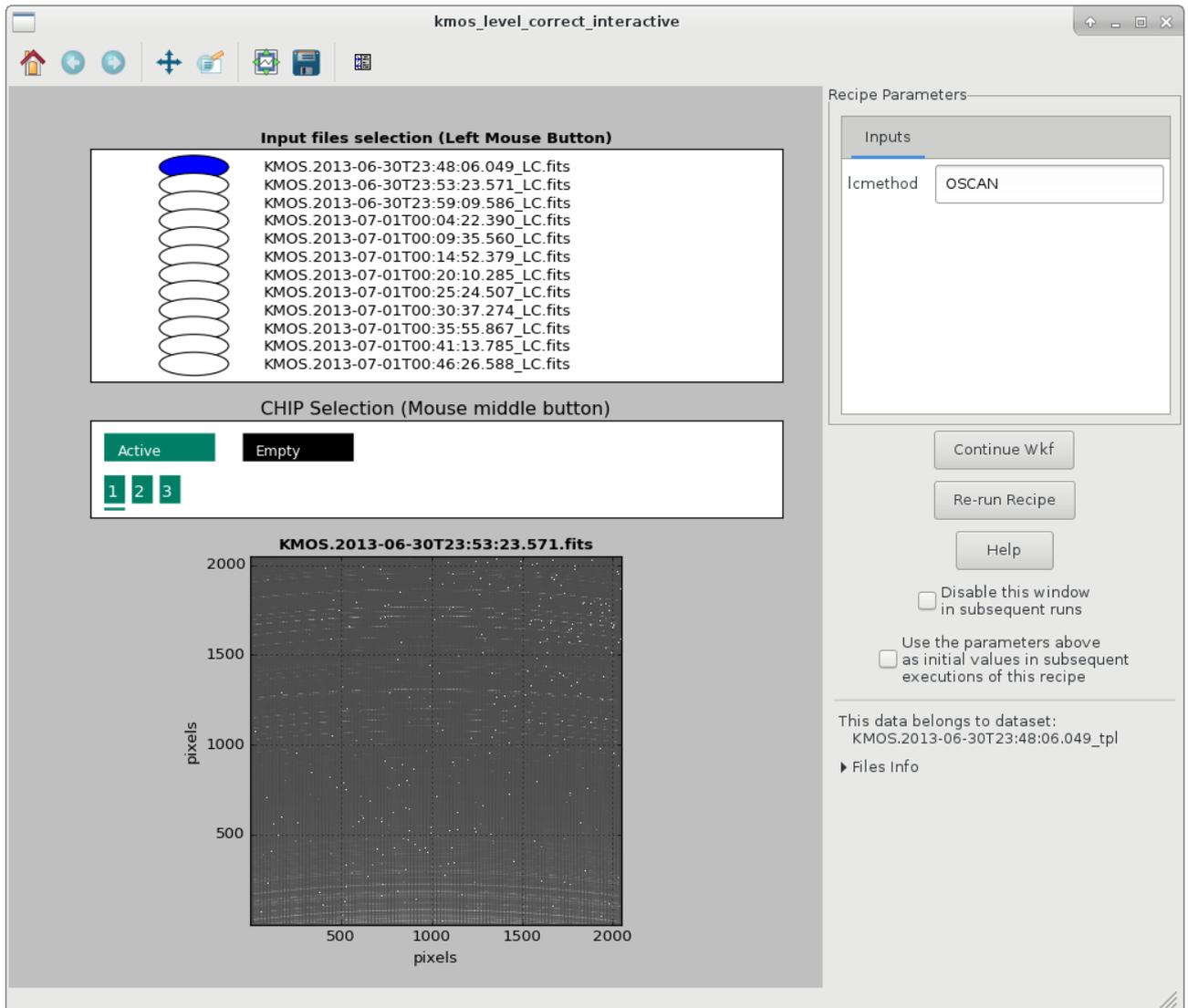


Figure 8.4: The interactive window for the *kmos_level_correct* recipe actor as displayed for the KMOS tutorial data set.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 50 of 81 |

Once the Object / Sky pair is identified, the recipe proceed to remove the sky from the corresponding object IFU. To suppress sky subtraction, set `-no_subtract=true` and `-sky_tweak=false`. This setup will create also reconstructed sky cubes.

There are several steps involved in the sky subtraction. First, the object cube is reconstructed and aligned to a reference spectrum with OH emission lines. This includes a 2nd polynomial order fit to compensate non optimal wavelength calibration. The alignment is triggered if the dataset contains a `OH_LINES` reference file, which is the default case and the recommended strategy. To remove this step, deselect the `OH_LINES` reference file from the dataset.

Then the user has the option to re-scale the intensities of the sky emission lines measured on the sky cube to match those observed in the science cube. This option exploits the sky tweaking algorithm described in Davies et al. 2007, MNRAS, 375, 1099, that defines groups of lines with the same scaling factor. This option is triggered setting `-sky_tweak=true` (default). In some (rare) cases, switching off the `-skytweak` method gives better results than the default configuration. Note that you cannot trigger this option and set `-no_subtract=true`.

The sky lines scaling factors computed via the sky tweaking algorithm can be not correct if there is still a residual mismatch in wavelength between the object spectrum and the science spectrum. This scenario can be identified by the presence of P-Cygni profiles in the residual sky lines in the reconstructed datacubes. In some cases, it is possible to correct for this wavelength mismatch by applying an additional re-alignment of the skylines measured in the sky and object cubes. This can be done by setting the parameter `-stretch=true`.

The algorithm detects bright emission lines from 1D sky spectra extracted from the object and the sky datacubes, identifies the same lines in both spectra, computes the difference in position, and computes a polynomial correction to minimize these differences. This correcting polynomial is then applied to the sky cube only in order to align it better to the object cube.

Note that the alignment of the sky cube to the `OH_LINES` reference spectrum is still done before applying the stretching algorithm, unless `-skip_sky_oh_align` is set to `true`. Skipping the sky alignment reduces the number of interpolations to reconstruct the sky datacube. However, in case of large offsets, it might still be useful to have this initial alignment before applying the stretching algorithm. Note that `-skip_sky_oh_align=true` has an effect only if the stretching algorithm is activated.

The outcome of the stretching algorithm needs to be evaluated with care. It is suggested to use high order polynomials (e.g. between 4 and 14) by setting the `-stretch_degree` parameter accordingly (the default degree is 8). The use of high order polynomials help in correcting for small wavelength mismatches in most of the spectral range, however it can introduce spurious effects at the edges of the wavelength range.

Figure 8.5 compares the results of the stretching algorithm on the first demo dataset (first exposure, first IFU) with default configuration. The amplitude of P-Cygni profiles observed in residual skylines are diminished when using the stretching algorithm.

8.4.4 Object / sky association for sky subtraction

The `kmoss_sci_red` recipe parameter `-obj-sky-table` allows the user to specify the path to an ascii file that associates every object exposure with its corresponding sky, overriding the automatic association done by the recipe itself.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 51 of 81 |

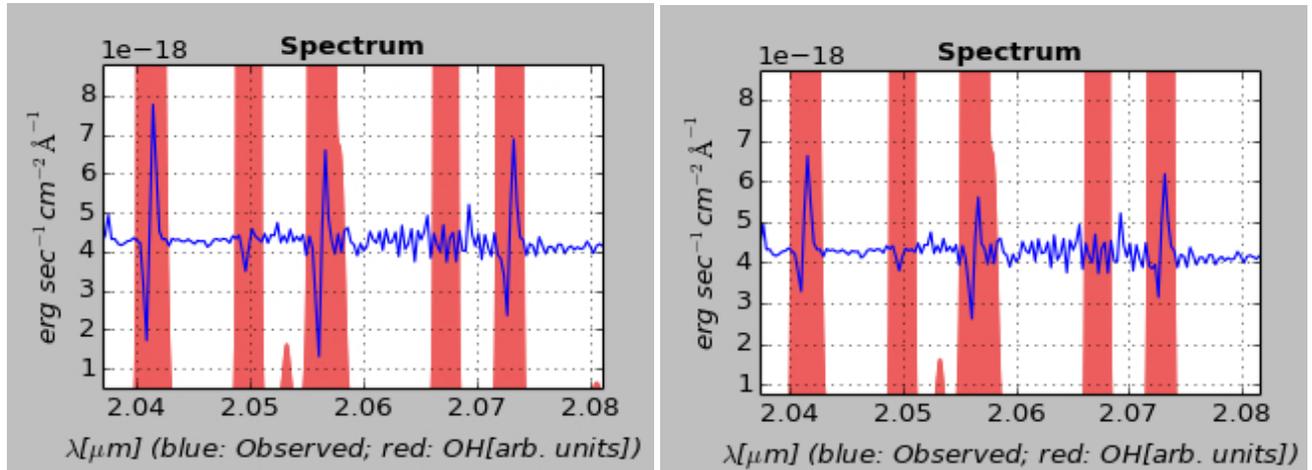


Figure 8.5: Residual sky lines with no stretching algorithm (left) and with stretching algorithm (right). The figures are zooms obtained from the interactive window of the Cube Reconstruction actor. The comparison refers to a small portion of the wavelength range, for the IFU 1 of the first exposure in the first demo dataset. The amplitude of P-Cygni profiles observed in residual skylines are diminished when using the stretching algorithm. The red stripes identify OH reference sky lines.

In the case the user wants to override the automatic association and feed the pipeline with his/her own, the suggested procedure is:

1. Run the workflow, selecting only one dataset and wait till the `kmos_sci_red` interactive window appears.
2. Locate the file `obj_sky_table.txt` produced by the `kmos_sci_red` recipe. It is located in the directory: `BOOK_KEEPING_DIR/kmos_sci_red/latest`.
3. Copy it into a safe place, and rename in a way that makes easy to associated it with the current dataset.
4. Change it according to the needs (an example is provided below). Each time the file is edited, it should be saved with a new name, otherwise the reflex lazy mode will ignore the changes and will re-use the previous results.
5. Enter the edited file name with its full path to the `-obj-sky-table` field in the interactive window.
6. Press `Re-run Recipe` to apply the changes. Each time a new file name is entered, Reflex will re-run the recipe. If a file has already been used (and if the other recipe parameters are unchanged), then the lazy mode will be activate and re-use the previous results without re-running the recipe. It is recommended not to save this parameters as new default, otherwise the other datasets will also use this object-sky association file.

8.4.5 Example of object/sky association table.

The ascii file that associates every object exposure with its corresponding sky looks like the following (the caption lines are not shown here, for sake of clarity).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 52 of 81 |

```
# [caption lines skipped]
Object/sky associations of frames tagged as: SCIENCE

index: filename:
# 0: reflex_input/kmos-demo-reflex-1.1/raw/KMOS.2013-06-30T23:48:06.049.fits
# 1: reflex_input/kmos-demo-reflex-1.1/raw/KMOS.2013-06-30T23:53:23.571.fits
# 2: reflex_input/kmos-demo-reflex-1.1/raw/KMOS.2013-06-30T23:59:09.586.fits
-----
IFU          1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18  19  20 21 22 23 24
-----
frame # 0:  reflex_input/kmos-demo-reflex-1.1/raw/KMOS.2013-06-30T23:48:06.049.fits
           type:  0  0  0  0  0  0  0  0  0  0  0  0  .  .  0  .  0  0  0  0  0  0  0  0
           sky in #: 1  1  1  1  1  1  1  1  1  1  1  1  .  .  1  .  1  1  1  1  1  1  1  1
frame # 2:  reflex_input/kmos-demo-reflex-1.1/raw/KMOS.2013-06-30T23:59:09.586.fits
           type:  0  0  0  0  0  0  0  0  0  0  0  0  .  .  0  .  0  0  0  0  0  0  0  0
           sky in #: 1  1  1  1  1  1  1  1  1  1  1  1  .  .  1  .  1  1  1/20 1  1  1  1  1
-----
```

The IFUs pointing to Object and Sky are identified with the letters “O” and “S”, respectively. Inactive arms are identified with a dot “.”. For each frame and IFU containing an object, the corresponding frame and IFU containing the sky is displayed in the line below. If the sky location is described with a single number, e.g. N (N is an integer), then the sky to be used is on the exposure with index N , on the same IFU as the object.

If the sky location is described in the format N/M (N and M integers, with $1 < M < 24$), then the sky to be used is located in the exposure with index N and IFU M .

In the example shown above, in the object frame `KMOS.2013-06-30T23:48:06.049.fits` (that has index 0) each n -th IFU has the corresponding sky cube in the same n -th IFU of the frame with index 1 (`KMOS.2013-06-30T23:53:23.571.fits`)

The same is true for the object frame `KMOS.2013-06-30T23:59:09.586.fits` (that has index 2), except for IFU 19. The corresponding sky has to be found in the frame with index 1, at ifu 20.

8.5 Science reduction: datacube combination

The next step in the data reduction is the combination of the reconstructed cubes that belong to the same target, or to reconstruct the full mosaic in the case of mosaic observations. This is automatically done in the `Cubes Combination` actor.

This actor is identifiable by an orange rectangle encompassing the actor name. The interactive mode is enabled by default. Should you wish to change that use `Open Actor` to get access to the the components of the interactive actors, then double-click on the composite actors, setting the “EnableInteractivity” parameter to `false`, and clicking `Commit` to save the changes to the workflow.

Figure 8.6 shows the interactive window that will pop-up at the end of the execution of the `kmos_combine` pipeline recipe (for the first demo-dataset).

The file list on the top-left panel (“Files selection”) shows the names of the combined datacubes produced by the recipe. The user can select the desired output file name to inspect by double-clicking with the left mouse button (it might require few seconds for the other windows to refresh).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 53 of 81 |

The right window (“File Spectra”) displays the spectra of the reconstructed cubes that contributed to the creation of the selected combined datacube. This window might not be available for observations taken in mosaic mode. These spectra are obtained by integrating the datacube along the 2 spatial direction. These are the spectra of the various extensions of the `SCI_RECONSTRUCTED` datacubes, which are combined into the selected datacube. On the right part of the plot, a code `n:m` for each spectrum is given, specifying the raw file number (`n`) and the IFU number (`m`). If one of the input spectra to be combined displays significant deviations from the others, it might be worth considering to remove it from the combination. To remove some datacubes from the combination, insert the corresponding `n:m` codes in the `skipped_frames` recipe parameter. Spectra of excluded frames will be grayed out in the File Spectra window.

The bottom left plot (“Spectrum”) shows in blue the spectrum of the selected combined datacube (obtained by integrating the datacube along the 2 spatial direction). It also shows in red the location of the sky emission lines (OH). The bottom right plot (“Collapsed image”) shows the reconstructed field of view of the selected datacube, obtained by integrating the datacube along the wavelength direction. For mosaic observations, this will show the full mosaic field of view.

On the right-hand edge of the interactive window the user may modify a number of parameters to improve the science frame reduction. If parameters have been changed, then clicking on the Re-run Recipe will re-execute the `kmoss_combine` pipeline routine.

At the bottom of the interactive window, the user can select 3 visualization modes. The first (default) shows the window as in Figure 8.6. The second shows the list of output spectra (the “Files Selection” plot), the spectrum (the “Spectrum” plot) and the reconstructed field of view (the “Collapsed Image” plot), of the selected output datacube. The last option shows only the input spectra (the “File Spectra” plot) that contributed to the creation of the selected datacube.

8.5.1 Combining exposures from different Observing Blocks

By design, the KMOS workflow can classify, reduce, and combine the exposures associated to a single observing block (OB), the exposures associated to a specific object, even from different OBs, and the reconstructed datacubes `SCI_RECONSTRUCTED` or `SINGLE_CUBES` produced by previous reduction that are associated to a specific object, even from different OBs (see Section 4.1).

If the user want to combine a number of exposures not grouped by the Data Organizer, (s)he has to run the `kmoss_combine` pipeline recipe on the workflows products manually. The suggested procedure is the following:

- Reduce all the OBs to be combined with the reflex workflow. The products to be combined (i.e., the `SINGLE_CUBES` or `SCI_RECONSTRUCTED` frames) will be saved into the directory specified in end products direcotry (see Section 6).
- Create the set of frames file (SOF) including only the products to be combined. It is an ascii file that specifies all the inputs of the pipeline recipe to be run and their categories. In the following example, the `data.sof` file contains the list of all the observations taken in 3 different OBs:

```
reflex_end_products\dataset1\SINGLE_CUBES_1.fits SINGLE_CUBES
reflex_end_products\dataset1\SINGLE_CUBES_2.fits SINGLE_CUBES
reflex_end_products\dataset1\SINGLE_CUBES_3.fits SINGLE_CUBES
```

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 54 of 81 |

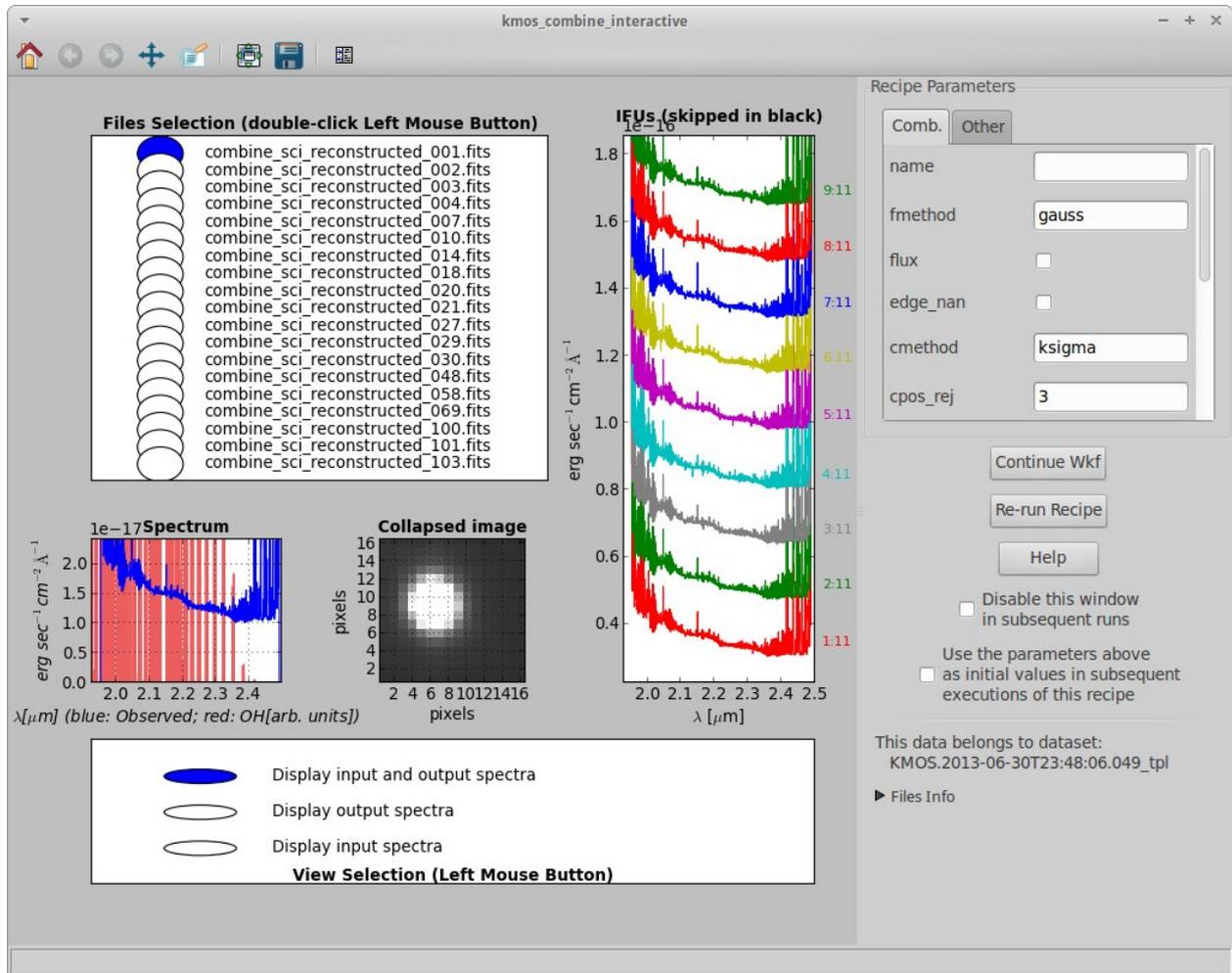


Figure 8.6: The interactive window for the Cube Combination actor as displayed for the KMOS tutorial data set.

```

reflex_end_products\dataset2\SINGLE_CUBES_1.fits SINGLE_CUBES
reflex_end_products\dataset2\SINGLE_CUBES_2.fits SINGLE_CUBES
reflex_end_products\dataset3\SINGLE_CUBES_1.fits SINGLE_CUBES
reflex_end_products\dataset3\SINGLE_CUBES_2.fits SINGLE_CUBES
reflex_end_products\dataset3\SINGLE_CUBES_3.fits SINGLE_CUBES
reflex_end_products\dataset3\SINGLE_CUBES_4.fits SINGLE_CUBES

```

Note that the data.sof must contain the full path to the file, or the relative path from the command line execution.

- Execute the pipeline recipe *kmos_combine* with the following command line.

```
esorex kmos_combine data.sof
```

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 55 of 81 |

For a full list of the recipe parameters and inputs/outputs, please consult the KMOS pipeline Manual. It is recommended to combine data of the same exposure time.

| | | | |
|------------|----------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 56 of 81 |

9 Telluric correction

The current KMOS workflow allows the user to select 5 strategies to correct the observations for atmospheric transmission.

1. **Empiric method.** Evaluation of telluric correction by comparing the extracted spectrum of the standard star with its model. This option is faster, but it might not give the best results. Select this option only if other options fail, if the target has low signal to noise ratio (e.g. < 10 per pixel, evaluated on wavelength region where the contamination of telluric features is minimal) or if telluric correction is not fundamental for your science.

This option can be selected by setting **Use molecfit/calctrans = false** and **telluric and response correction = 0** in the main workflow canvas.

If standard star are not observed with the same IFUs as the science, it is advisable to compensate for the different instrument responses of various IFUs by setting **Apply IFU-customized response=true** in the main workflow canvas.

2. **Molecfit on standard.** This strategy uses the 1D spectrum of the standard star to determine the column density of the molecules in the atmosphere that cause the telluric absorption; individual IFUs are fitted independently. The fit is done in the actor “Atmospheric model” (recipe: *kmos_molecfit_model*). . This actor is described in Section 9.2. Then, the telluric correction is computed in the actor “Atmospheric Transmission” (recipe: *kmos_molecfit_calctrans*) and accounts for the difference in airmass between the observations of the standard star and the science⁸, and the difference in the spectral resolutions (as function of wavelength) between the IFUs associated to the standard star and the science. This actor is described in Section 9.3.

The correction itself is applied in the actor “Telluric correction from standard star” (recipe: *kmos_molecfit_correct*), which is described in Section 9.4.

This strategy gives in general relatively good results and it is the default option, despite it is time consuming.

To select this option set **Use molecfit/calctrans = standard** and **telluric and response correction = 2** in the main workflow canvas.

3. **Molecfit on science** This strategy runs the same recipes *kmos_molecfit_model*, *kmos_molecfit_calctrans*, and *kmos_molecfit_correct* directly on the science reconstructed cubes, which are processed separately.

For each reconstructed science cube, the user selects a bright reference science exposure and some IFUs to use as telluric standard for the determination of the atmospheric model. 1D spectra are extracted from the reference cubes and fitted with *kmos_molecfit_model*.

The extraction step is described in Section 9.5, whereas the selection and association science-reference steps are described in Section 9.6

All these steps are executed within the actor “Telluric correction on science frame”.

This strategy gives the best results, as it minimizes the changes between atmospheric conditions and instrument flexures between the reference and the science spectra to correct. It requires however the

⁸The airmass of the first science exposure is used as reference for all the observations in that OB.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 57 of 81 |

presence of bright sources among the scientific exposures to use as reference, an interactive association of the science and reference spectra, and a fine-tuning of the wavelength regions to fit to avoid spectral features that are proper of the reference spectrum and not of the atmosphere.

To select this option set **Use molecfit/calctrans = science** and **telluric and response correction = 2** in the main workflow canvas.

4. **User-provided telluric correction files.** Alternatively, the user can provide its own telluric correction and put it together with the input raw data. Each OB will use the telluric correction closer in time, within 1 day time-frame. The format of the user-provided file must be as the output of *kmos_molecfit_calctrans*.

To select this option set **Use molecfit/calctrans = false** and **telluric and response correction = 3**. If the user-provided file contains also the instrumental response together with the telluric correction, then set **Use molecfit/calctrans = false** and **telluric and response correction = 1**.

5. **No telluric correction.** To select this option set **Use molecfit/calctrans = false** and **telluric and response correction = 2**. Select this option if the telluric correction is either not needed or to be performed afterwards via external tools.

For detailed information on Molecfit and calctrans, we refer to the Molecfit user manual available at:

<http://www.eso.org/sci/software/pipelines/skytools/molecfit>

9.1 Notes on the use of Molecfit tools

Telluric correction is advisable for scientific targets that have detectable continuum; for emission lines only targets it is advisable to skip the telluric correction.

The use of molecfit/calctrans in general provides the best results, as it construct a model of the atmospheric transmission, and does not transfer to the science all the issues that can be included in the extraction of the 1D standard star spectrum, such as low S/N, cosmic-rays, and recipe failure in the absorption line fitting. However, it is very slow, and the user might want to optimize the atmospheric model parameters to improve the execution speed and the fit performances (see Section 9.2).

A part from avoiding issues of the standard star spectrum extraction, the benefits of using of molecfit/calctrans are visible for high S/N targets ($S/N \geq 20$). If the target has low S/N but telluric correction is still wanted, then one can first give a try with **Use molecfit/calctrans = false** and **telluric and response correction = 0**.

Another advantage of using molecfit/calctrans is that it accounts for the dependency of the instrumental spectral resolution from wavelength, IFU and grating, and rotator angle. This is advisable for example, if a bright target was not observed with the same IFU and rotator angle as the telluric standard star.

On the other hand, one might prefer to use the output of *kmos_std_star* (**Use molecfit/calctrans = false**) for telluric correction in the case that the standard star is observed with the same IFUs as the target. In this way, the determination of the response curve is based on the data from that night, and does not rely on average static calibration as in the molecfit/calctrans case (see also Section 10).

The strategy of running molecfit directly on science has the advantage that the atmospheric model is obtained under the same atmospheric conditions (water vapors, column density of molecules) as the science spectra. This

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 58 of 81 |

strategy is recommended if there are science data with high signal to noise and few intrinsic features. These data can be used to determine the correction for all other science frames in the dataset.

The above steps are re-iterated for each reconstructed cube and each observing block.

9.2 Computation of atmospheric model: running molecfit on selected spectra

The KMOS workflow has two actors dedicated to the computation of the atmospheric model and the column density of molecules in the atmosphere. One is named `Atmospheric fit`, visible in the main canvas and identifiable by an orange rectangle encompassing the actor name. This is executed if the parameter `Use molecfit/calctrans` is set to “standard”. The second is embedded in the actor `Telluric correction on science frames` and it is executed if the parameter `Use molecfit/calctrans` is set to “science”.

The interactive features are common for both cases and they are described here. Figure 9.1 shows the interactive window that will pop-up at the end of the execution of the actor.

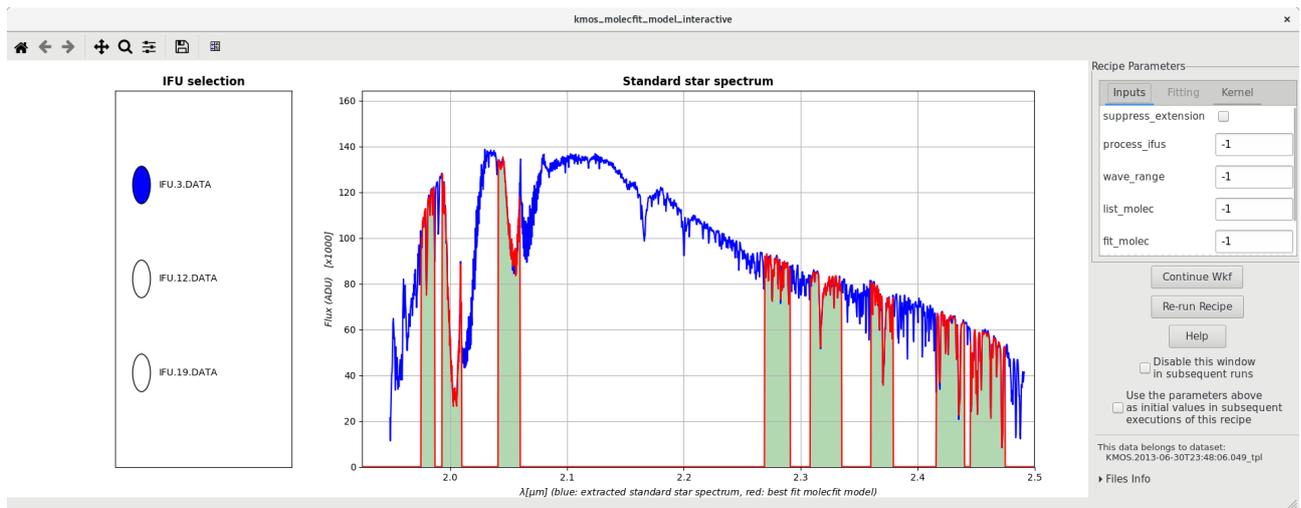


Figure 9.1: The interactive window for the `Atmospheric fit` (recipe: `kmos_molecfit_model`). This example refers to the *GUMM-43* demo dataset..

The left part of the window allows to select the processed IFU to be inspected. The plot area shows the input extracted 1D spectrum in blue, and the best fit model obtained by molecfit in red. The wavelength regions used by the fit are highlighted in green, they are editable via direct input on the recipe parameter area (see also 9.2.1. Tool tip information displaying the best fit parameters and atmospheric model are prompted by moving the mouse cursor on the top of the plots.

Table 9.1 lists the full list of parameters that are configurable via the interactive window. In the following section, we will discuss a number of them.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 59 of 81 |

9.2.1 Tips for an efficient fit.

The default parameters should provide a good model of the input spectrum in the majority of the cases. Here we list some tips to improve the fit or the recipe efficiency if the user needs it. For testing purposes, one can limit the number of IFUs to fit (recipe parameter `-process_ifu`) and fit them all once the best configuration has been found.

- **Wavelength ranges.** The fit is performed on a sub-set of wavelength ranges and not to the entire spectrum; this is found to be more efficient and less time-consuming. The advice is to select few wavelength ranges that include the expected molecules that are observable in the spectrum (see Figure 9.2 for an atlas). At the moment, the ranges to be fit can be inserted by direct input to the recipe parameter `wave_range`. Future releases will allow to specify the wavelength ranges by clicking with the mouse in the plot window.

For example, the entry `'0.815,0.830,0.972,0.986'` will perform the fit in the two ranges $0.815 < \lambda[\mu m] < 0.830$ and $0.972 < \lambda[\mu m] < 0.986$. Please, include the quotation marks “ ’ ”, do not include spaces. The default value “-1” uses the hard-coded ranges (values in in μm):

- IZ: `'0.815,0.830,0.894,0.899,0.914,0.919,0.929,0.940,0.972,0.986'`
- YJ: `'1.106,1.116,1.075,1.083,1.131,1.137,1.139,1.149,1.155,1.166,1.177,1.189,1.201,1.209,1.263,1.276,1.294,1.303,1.312,1.336'`
- H: `'1.482,1.491,1.500,1.512,1.559,1.566,1.598,1.605,1.575,1.583,1.622,1.629,1.646,1.671,1.699,1.711,1.721,1.727,1.746,1.758,1.764,1.767,1.773,1.780,1.789,1.794'`
- K: `'1.975,1.987,1.993,2.010,2.041,2.060,2.269,2.291,2.308,2.335,2.360,2.379,2.416,2.440,2.445,2.475'`
- HK: `'1.575,1.584,1.594,1.606,1.646,1.671,1.756,1.771,1.781,1.811,1.945,1.969,1.975,1.987,1.993,2.030,2.043,2.089,2.242,2.294,2.308,2.335,2.360,2.379'`

It is good advice to limit the number of wavelength regions to 4 or 5. The default ranges are defined to provide a good fit for all the possible cases at the cost of high execution time, but in general, they are redundant. Regions have to be selected to include strong telluric features, avoiding at the same time regions where transmission is close to 0. See Figure 9.2 as reference. All molecules expected to contribute in a specific instrumental wavelength range should be considered when defining the fitting regions.

It is also advisable to define narrow wavelength regions, so that the continuum can be approximated by a 1st or 2nd order polynomial.

- **Molecules.** In the fit, only the molecules that appears in the wavelength region of the grism should be fitted. The recipe parameters `-relcol`, `-fit_molec`, and `-list_molec` should then be defined accordingly. The hard-coded values depends on the instrument configuration and should be valid. Change the default only if you have evidence that a given molecule is not affecting the transmission.

Hard-coded defaults of `-list_molec` are listed in Table 9.1. For a full list of supported molecules, please consult the molecfits user manual.

- **Instrument setup (continuum and wavelength calibration).** It is advisable to fit both the continuum shape and the wavelength solution. Each wavelength region used in the fit has the continuum fitted independently; it is recommended to use a small polynomial degree such as 1 or 2 (unless the extension of

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 60 of 81 |

the wavelength regions justifies the use of larger values). A global wavelength solution is used for all the wavelength regions. The wavelength solution is found by comparing the position of the telluric features in the observations with those of the internal molecfit atlas. Note that the output spectrum will not be adapted to the new wavelength scheme, in order not to have extrapolation problems at the edges of the wavelength region. It is advisable to use a small polynomial degree (1 to 4).

- **Instrumental spectral resolution (kernel).** The fit to the observations accounts for the instrumental spectral resolution in several ways. The default is to use static calibration that models the wavelength dependent (Gaussian) FWHM for each IFU of each grating at several rotator angles (see the `kernel_X.fits` files distributed with the pipeline). Alternatively, the user can let the recipe to fit the instrumental spectral resolution. The fit includes Gaussian, Lorentian, and Voigt profiles. The spectral resolution can be constant in terms of FWHM (angstrom) or in terms of resolving power (fixed R). The latter is rarely used, but it can be activated by setting `-varkern=true`.

In the majority of the cases, the best results are obtained by using the provided static calibration.

- **Fit precision.** The fit precision is regulated by `-ftol` and `-xtol`. Default values are generally good. If fitting precision is increased, the improvement of the fit is not noticeable but the computation time increases a lot. One might set `-xtol` to 0.01 (instead of 0.001) during the testing, and set back the default resolution once the optimal configuration has been found.

Future releases will allow to specify the wavelength ranges by clicking with the mouse in the plot window.

9.3 Computation of atmospheric transmission: running calctrans

Once the parameters of the atmosphere are computed (either from a telluric standard or from a science frame), the workflow combines them with the reconstructed cube to determine the full telluric correction. The process accounts for the airmass difference between reference spectrum that was used to model the atmosphere (either telluric standard or science) and the science observations to be corrected. The recipe also account for the different instrumental resolution between a) the IFU X, in which the reference spectrum was observed, and b) IFU Y, in which the scientific target was observed. The IFU X and IFU Y match is done automatically, or can be specified by the user.

This is done by the two interactive actors `Atmospheric transmission`, that execute the pipeline recipe `kmoss_molecfit_calctrans`. One is visible in the main canvas, and it is executed if **Use molecfit/calctrans** is set to “standard”. The second is embedded in the actor **Telluric correction on science frames** and it is executed if the parameter **Use molecfit/calctrans** is set to “science”. Their interactive features are identical and they are described here and shown in Figure 9.3.

The left part of the window allows to select the processed IFU to be inspected. Only the first input science spectrum (`SCI_RECONSTRUCTED`) is used by the recipe. The plot area shows: the extracted 1D science spectrum as produced by `Cube reconstruction` and integrated over the spatial directions (blue, upper panel) the corrected spectrum (red, upper panel), and the applied telluric correction (bottom panel).

The user can select which `IFU_Y` solution to use for each `IFU_X` in the science by specifying the corresponding recipe parameter `IFU_X` (with X ranging from 1 to 24). For example, setting the recipe parameter `IFU_1=3` will use the solution determined by the molecfit model obtained on IFU 3 (Y=3) to calibrate the scientific data

| | | | |
|------------|-----------------------------|---------------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 61 of 81 |

Table 9.1: Parameters that a user can manipulate within the `Atmospheric fit` interactive window

| Parameter | Default value | Explanation |
|----------------------------------|---------------|---|
| <code>-process_ifus</code> | -1 | A list of IFUs to process. If set to -1, all the IFUs that have data will be process. |
| <code>-suppress_extension</code> | false | Suppress arbitrary file name extension.(TRUE (apply) or FALSE (don't apply)). |
| <code>-wave_range</code> | -1 | A list of numbers defining the wavelength ranges to fit in the grating. If set to -1, grating dependent values are used (see text for reference). |
| <code>-list_molec</code> | -1 | A list of molecules to fit. If set to -1, grating dependent default values are used: IZ: 'H2O'; YJ: 'H2O,CO2,CH4,O2'; H: 'H2O,CO2,CO,CH4'; K='H2O,CO2,CH4', HK='H2O,CO2,CH4'. |
| <code>-fit_molec</code> | -1 | Flags to fit the column density of the corresponding list_molec. If set to -1, grating dependent values are used (see manual for reference). |
| <code>-relcol</code> | -1 | Column density relative to atmospheric profile of the corresponding list_molec in grating. If set to -1 grating dependent default values are used (see manual for reference). |
| <code>-ftol</code> | 0.01 | Relative chi-square convergence criterion. |
| <code>-xtol</code> | 0.001 | Relative parameter convergence criterion. |
| <code>-fit_cont</code> | true | Flag to enable/disable the polynomial fit of the continuum. |
| <code>-cont_n</code> | 1 | Degree of polynomial continuum fit. |
| <code>-fit_wlc</code> | true | Flag to enable/disable the refinement of the wavelength solution. |
| <code>-wlc_n</code> | 2 | Polynomial degree of the refined wavelength solution. |
| <code>-wlc_const</code> | 0 | Initial constant term for wavelength adjustment (shift relative to half wavelength range). |
| <code>-use_input_kernel</code> | true | It specifies to use or not the input kernel, if provided. |
| <code>-fit_res_box</code> | false | Fit resolution by Boxcar LSF (if input kernel is not provided or discarded). |
| <code>-relres_box</code> | 0 | Initial value for FWHM of Boxcar rel. to slit width if input kernel is not provided or discarded (Range between 0 and 2) |
| <code>-fit_res_gauss</code> | true | Fit resolution by Gaussian (if input kernel is not provided or discarded). |
| <code>-res_gauss</code> | -1.0 | Initial value for FWHM of the Gaussian in pixels if input kernel is not provided or discarded. The default value -1 uses the following hard-coded values: IZ=1.84, YJ=1.82, H=1.76, K=1.73, HK=2.06. |
| <code>-fit_res_lorentz</code> | false | Fit resolution by Lorentz (if input kernel is not provided or discarded). |
| <code>-res_lorentz</code> | 0.5 | Initial value for FWHM of the Lorentz in pixels (if input kernel is not provided or discarded). |
| <code>-kernmode</code> | false | Voigt profile approx. or independent Gauss and Lorentz (if input kernel is not provided or discarded). |
| <code>-kernfac</code> | 5.0 | Size of Gaussian/Lorentz/Voigt kernel in FWHM (if input kernel is not provided or discarded). |
| <code>-varkern</code> | false | Does the kernel size increase linearly with wavelength if input kernel is not provided or discarded? |

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 62 of 81 |

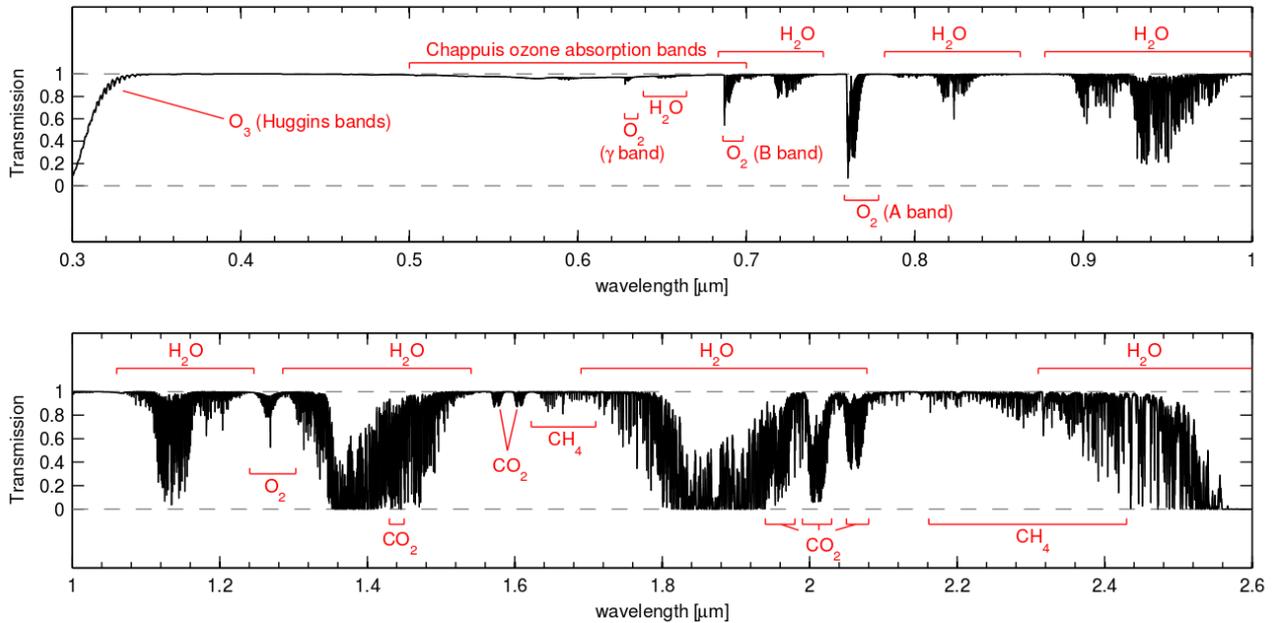


Figure 9.2: Synthetic absorption spectrum of the sky between 0.3 and 2.6 microns calculated with LBLRTM (resolution $R=10000$) using the annual mean profile for Cerro Paranal (Noll et al. 2012), adapted from Smette et al. 2015, A&A, 576, 77. The eight main molecules O_2 , O_3 , H_2O , CO , CO_2 , CH_4 , OCS , and N_2O contribute more than 5% to the absorption in some wavelength regimes. The red regions mark the ranges where they mainly affect the transmission, minor contributions of these molecules are not shown..

on IFU 1 ($X=1$). The recipe accounts for differences in airmass between standard star and science and, if kernel is provided and used, it also account for differences in instrumental dispersion between IFUs X and Y. A value of -1 triggers the following automatic matching:

1. If IFU_Y in ATMOS_PARM and BEST_FIT_PARM contains data, then IFU_X=IFU_Y.
2. Otherwise the first IFU_first_chip_Y with valid data among those that belong to the same detector as IFU_Y will be used.
3. If there are no available IFU for that detector, the first IFU_first among the input data that contains data will be used.

9.4 Correction for atmospheric transmission

Once the full telluric corrections are computed by the calctrans step, the workflow applies them to the science spectra. This step is done by the two actors Telluric correction.One is visible in the main canvas, and it is executed if Use molecfi/calctrans is set to “standard”. The second is embedded in the actor Telluric correction on science frames and it is executed if the parameter Use molecfi/calctrans is set to “science”. Their interactive features are identical and they are described here and shown in Figure 9.4

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 63 of 81 |

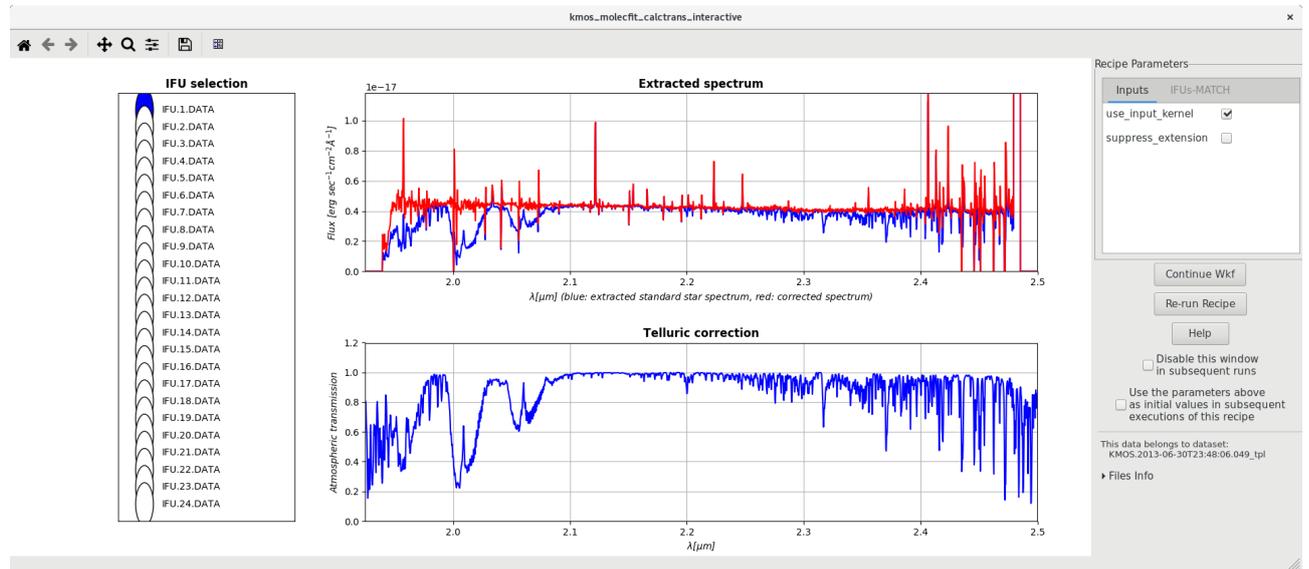


Figure 9.3: The interactive window for the Atmospheric transmission (recipe: *kmos_molecfit_calctrans*). This example refers to the GUMM-43 demo dataset..

The upper part of the window allows to select the input file, the central part the IFU. The bottom part of the plot area compares the extracted 1D science spectrum as produced by `Cube reconstruction` and integrated over the spatial directions (blue, upper panel) the corrected spectrum (red, upper panel). Wavelengths in which the atmospheric transmission is below the threshold indicated by the recipe parameter `min_threshold` are not corrected (default: 0.01).

9.5 Extraction of 1D science spectra

This stage is executed only if `Use molecfit/calctrans = "science"`, and it is embedded within the actor `Telluric correction` on science frames.

This interactive window allows the user to extract 1D spectra from the science exposures. These spectra are used to fit the atmospheric model that will be used to determine the telluric correction. It is important to do a good extraction only for those spectra and those IFUs that will be chosen as reference and fit by `molecfit`. The extraction on other datacubes is irrelevant and the 1D spectra are used only for visualization purposes. This interactive actor is executed only if `Use molecfit/calctrans = "science"`, and it is embedded within the actor `Telluric correction` on science frames.

Figure 9.5 shows the interactive window. The left part of window shows the extracted spectrum of the selected IFUs; IFUs can be selected from the radio buttons. The recipe part allows to change the recipe parameters and re-execute the workflow. The window is executed for each reconstructed cube.

It is advisable to set the recipe parameter `center` to `-1,-1`, so that the center is automatically set to the peak of the image in each IFU.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 64 of 81 |

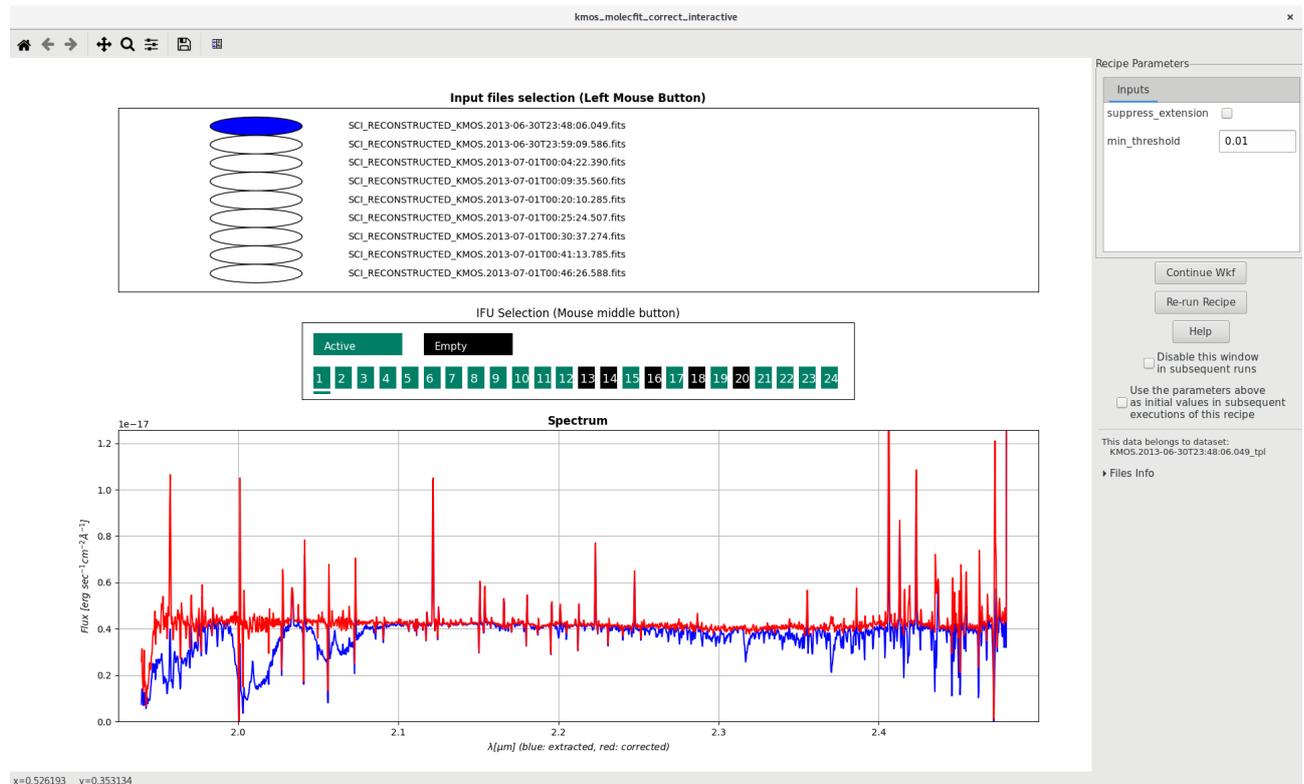


Figure 9.4: The interactive window for the Telluric correction (recipe: `kmos_molecfit_correct`). This example refers to the GUMM-43 demo dataset..

9.6 Selection of reference spectra

This stage is executed only if `Use molecfit/calctrans = "science"`, and it is embedded within the actor `Telluric correction` on science frames. The actor allows the user to select, for each scientific exposure, the reference exposure and the IFUs to use for the evaluation of the atmospheric model. This actor is interactive by default, and, contrarily to the other actors, its interactivity is not regulated by the `GlobalPlotInteractivity` parameter. The rationale is that if the user wants to run `molecfit` on science frames, (s)he must be allowed to specify most suitable reference science frame to run `molecfit`.

The plot area has 2 plots (see Figure 9.6). The plot on the left shows the extracted spectra of the 24 IFUs in the science exposure to be corrected. The plot on the right shows the extracted spectra of the 24 IFUs in the reference exposure, that will be treated as standard star and used to compute the atmospheric model. The selected IFUs are highlighted in red and they can be specified on the right side of the window by the parameter `PROCESS_IFUS`. The reference frame can be selected from the radio button on the top of the window (Files selection).

Selection of IFUs and reference frame. First, inspect all desired combinations and make the choice. Then, select the desired IFUs: specify the IFUs number separated by a coma in the `PROCESS_IFUS` field and then press `Re-run Recipe`. A value of `-1` will process all valid IFUs. Finally, select the desired file with the radio

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 65 of 81 |

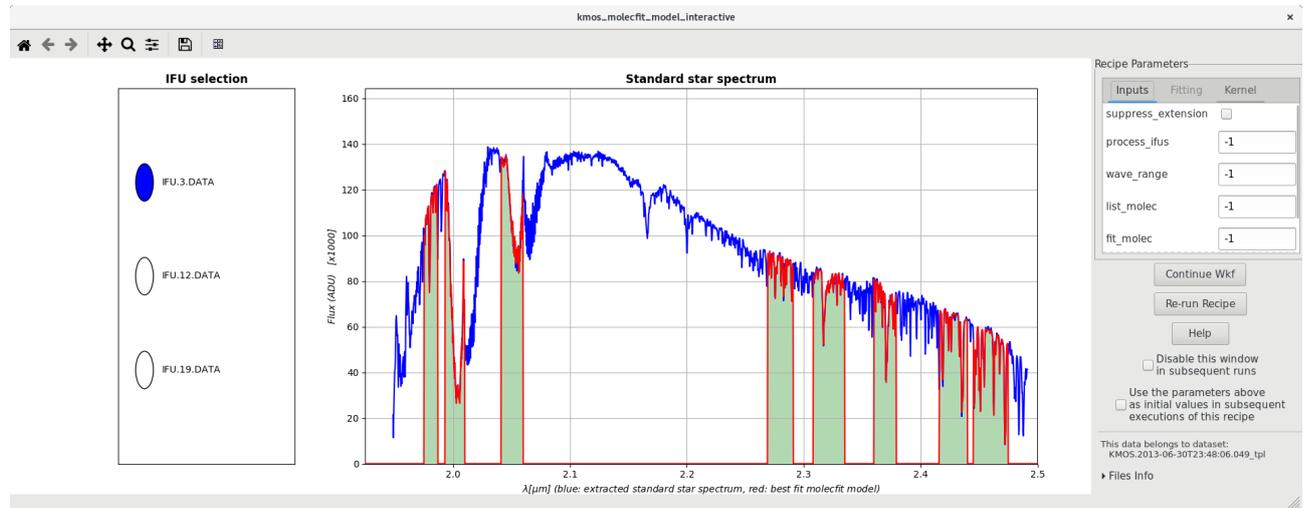


Figure 9.5: One of the the interactive windows for the Telluric correction on science frames actor (recipe: `kmos_extract_spec`). This example refers to the first reconstruction cubes of the GUMM-43 dataset..

buttons and press Continue wkf. The selected reference frame will be send to the next steps (molecfit and caltrans, Sections 9.2 and 9.3) to generate the telluric correction for the science frame to be corrected. The correction will be then done as in Section 9.4.

The loop (select reference, molecfit, caltrans, correction) is then repeated for the next scientific frame (9 times in the case of the GUMM-43 dataset).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 66 of 81 |

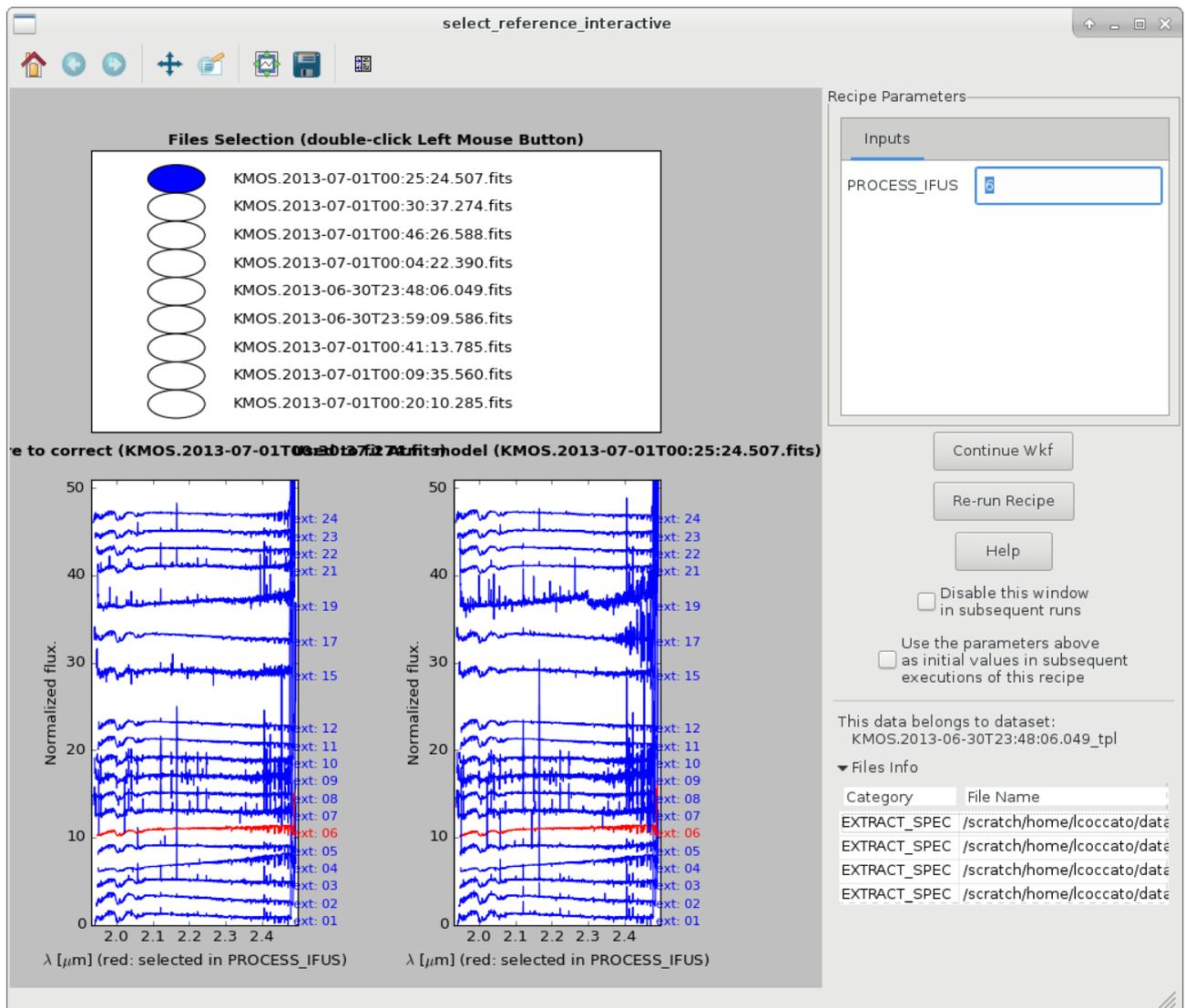


Figure 9.6: One of the the interactive windows for the Telluric correction on science frames actor (recipe: `kmos_extract_spec`). This example refers to the first reconstruction cubes of the GUMM-43 dataset.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 67 of 81 |

10 Instrumental response

The current KMOS pipeline distribution has a static calibration file that contains the average instrument responses and zeropoint for all the IFUs (category: `RESPONSE`). The calibration is accurate from 5% (in the regions not affected by telluric absorptions) up to 15% (in the regions most affected by telluric absorptions).

The correction stored in `RESPONSE` is applied to the data if the workflow parameter **telluric and response correction** is set either to 2 or 4. If this workflow parameter is set to 0 and **Use molecfit/caltrans** = “false”, the workflow uses the telluric star observations to generate both response and telluric correction. The IFUs that were used to observe the standard star will be used to generate the correction for all the other IFUs. However, the instrument response is different for each IFU. The user can correct this by setting **Apply IFU-customized response**=true. This parameter modifies the response of an IFU by using the average response stored in the static calibration `RESPONSE`.

Setting **telluric and response correction = 0** and **Apply IFU-customized response**=false is recommended only if the telluric star is observed in all the IFUs and if using **Use molecfit/caltrans** = “false”.

In the following, we describe how to compute a custom response curve, specific for a given set of observations. Because it is very difficult to disentangle the contribution of instrument response and telluric atmospheric absorption, the procedure will not return a much better precision than what provided by the calibration files. However, in rare cases, it might be needed to create a custom response if the continuum shape of the target suggests it.

The product of the recipe `kmos_std_star` `telluric.fits` contains both the information on the telluric absorption and the wavelength dependence of the response curve. On the other hand, the telluric correction produced by `molecfit` contains only the information on the telluric absorption.

It is therefore possible to extract the instrument response contribution by dividing the `telluric.fits` file by the `molecfit` transmission. The following example refers to the case of one standard star spectrum. It has to be repeated for all the spectra included in `telluric.fits` file.

- Get the data in one valid extension of the `telluric.fits` produced by `kmos_std_star` (e.g., extension number 5, that refers to detector number 1), and the corresponding telluric correction obtained by `molecfit` for the same star and extension. Let’s refer to these quantities by `t_kmos` and `t_molecfit`
- If needed, resample `t_molecfit` to the same wavelength vector as `t_kmos`.
- Compute the ratio `t_kmos / t_molecfit`. An example of such a ratio is shown in Figure 10.1 . The large-scale trend of this curve represents the instrument response curve, whereas the high frequency structures, i.e. the spikes in the curve, are due to the differences between the telluric features computed by the `kmos` pipeline and `molecfit`.
- Isolate the general trend of the ratio computed above, i.e. by applying a median filter or kernel smoothing. It might be necessary to extrapolate the results for the edges, where the noise is higher. The general trend represents the instrument response, and it is highlighted by the red curve in Figure 10.1 . In the particular example of Fig. 10.1 , the smoothing has been done by a boxcar average with 200 pixels width, as implemented in the `SMOOTH` IDL function. The response below 1.46 microns has been obtained as extrapolation of 1 degree polynomial fitted in the 1.46 and 1.49 microns range. The response above 1.83 has been set to constant.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 68 of 81 |

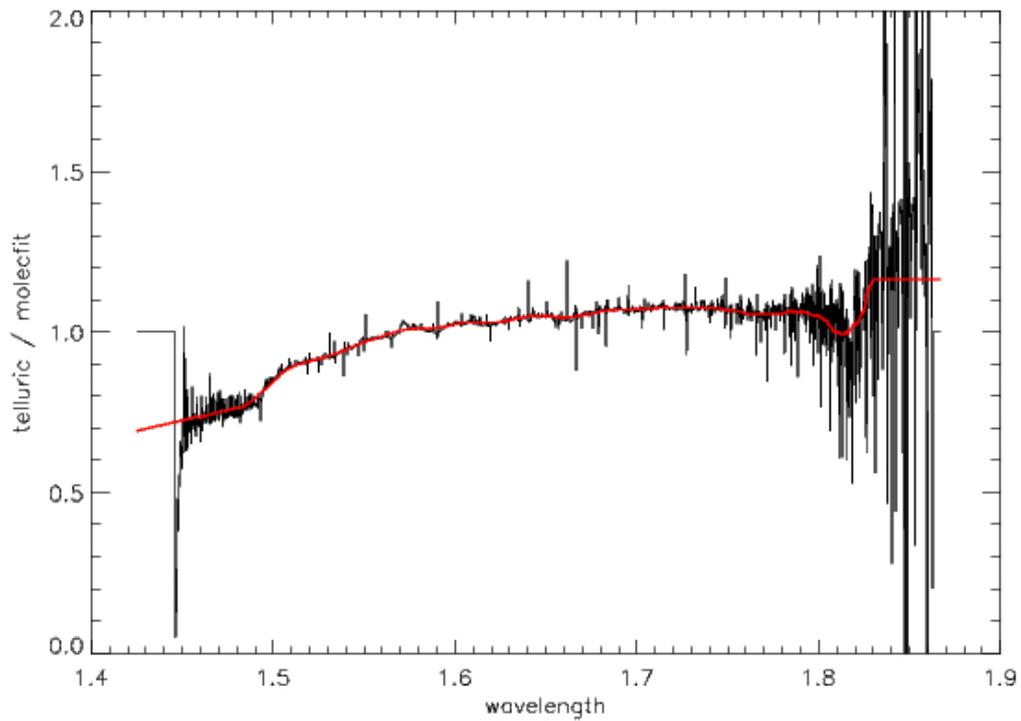


Figure 10.1: Example of the ratio between an extension of the telluric.fits file produced by the KMOS pipeline and the telluric correction produced by molecfit. The red curve, obtained by smoothing this ratio with a 200 pixels wide box car average (smooth function in IDL), represent the instrument response for that IFU.

- Repeat the above steps for all the valid extensions of telluric.fits and collect all the response curves.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 69 of 81 |

11 Tips

In this Section we present some suggestions that can help the user to improve some aspects of the data reduction. This section is in constant change, and it will be updated while we gain experience in the data reduction and add functionality to the pipeline.

The following suggestions work in some cases, but there is no guarantee that they are valid for all the datasets.

11.1 Specifying the number of cores to use via `OMP_NUM_THREADS`

The recipes associated to the telluric of telluric features that exploits molecfit algorithm (`kmos_molecfit_model`, `kmos_molecfit_calctrans`) can be executed in parallel on multiple processors by setting the environmental variable `OMP_NUM_THREADS` accordingly. To set the variable in `esoreflex` follow these steps:

1. Create a new configuration file. Type:

```
esoreflex -create-config myconfig.rc
```

A new configuration file will be created. Note that you might need to provide the full path of the `esoreflex` command.

2. Edit the configuration file and set the variable inside it `esoreflex.inherit-environment` from `FALSE` (default) to `TRUE`.
3. Launch the `kmos` workflow via the command:

```
env -i DISPLAY=:0 OMP_NUM_THREADS=24 <full_path>/esoreflex
    -config <full_path>/parallel.rc <full_path>/kmos.xml
```

This will launch reflex on a clean environment on which you have set the `OMP_NUM_THREADS` variable. Depending on your system, you might need to change the display number or add `xhost +` right after `env -i`. Also, you might need to change the number of cores to use (24 in the example above). The display number can be inferred by typing `echo $DISPLAY` before starting the procedure.

If you do not know the full path to the `kmos.xml` workflow, just type `esoreflex -l` to list the available workflows and their location before starting the above procedure.

11.2 Dealing with residual sky emission lines.

One of the major challenges with the KMOS data reduction is sky subtraction. In particular, residual of sky emission lines can be difficult to remove. Section 8.4.3 describes how to optimize the sky subtraction using the workflow interactive window. Here we list some strategies that can be applied to the reduced cubes produced by the reflex workflow.

- In Mosaic mode, if there are IFUs that contain negligible contamination from science sources, the following post-processing strategy can reduce the amplitude of residual sky lines.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 70 of 81 |

1. Run the workflow with your favourite recipe parameters, and set `-background` to “true” in the Cube Reconstruction actor. In this way, we remove at best the underlying continuum. Let’s refer to the combined final datacube (`PRO.CATG = COMBINE_SCI_RECONSTRUCTED`) obtained with this technique as “optimized residual”.
2. Run the workflow with your desired parameters, and set `-background` set to false in the Cube Reconstruction actor. Let’s refer combined final datacube obtained with this technique as “sky removed”.
3. In the “optimized residual” datacube, locate a portion in the field of view corresponding to one IFU where the science target contamination is minimal. Construct a single spectrum from these regions by averaging the spectra in the selected region. This represents the “spectrum of residuals”.
4. Subtract the “spectrum of residuals” from the “sky removed” datacube.

It is worth noting that with this technique, we do not subtract directly the sky lines evaluated on a single IFU. Indeed, sky lines can vary in different IFUs, and they are subtracted in step 2. We subtract “the residuals”, and the results are as good as long as the residual have a common structure among the IFUs (which is sometimes the case).

- For isolated and small sources (much smaller than the IFU size) it might be worth trying to evaluate the sky (or the sky residuals) to subtract on the spaxels around the scientific source. To skip the sky subtraction in the KMOS workflow, set `no_subtract = true` in the Cube Reconstruction actor
- For science sources that have spectral features more broad than than the typical size of sky residuals (which are few pixels), one can try a spectral median filtering for each spectrum in the reconstructed datacube. The value at wavelength of pixel coordinate z is compared to the background level b estimated as the median of the spectrum in the wavelength regions $z-2\cdot dz < z < z-dz$ and $z+dz < z < z+2\cdot dz$. If the value at z differs from the median more than $k \cdot \sigma$, where σ is the standard deviation evaluated in wavelength region specified above, then it is replaced by b . The following values are good for targets that have spectral features with FWHM larger than ~ 15 pixels: $dz = 3$, $k = 3$. It is advisable to check the object spectral features before and after the filtering to evaluate whether they have been deteriorated by the filtering procedure.
- If the datasets contains IFUs that, during the nodding-on-sky, were always assigned to sky positions, one can compute a datacube of sky residuals. A mean spectrum of residuals can be generated and by collapsing the datacube along the spatial directions, or by using the recipe `kmos_extract_spec`. This mean sky-residual spectrum can be scaled and subtracted from the scientific sky-subtracted datacubes.
- If skylines residuals show P Cygni profiles, consider to use the “stretching algorithm” (parameters `stretch` and `stretch_degree` in the Cube Reconstruction step (recipe `kmos_sci_red`)).
- If the object is very faint (basically no continuum), consider to evaluate the background from non-illuminated portions of the detector by setting the parameter Preview Level correction to True in the main canvas.
- When co-adding multiple exposures, double check if the alignment is done right. In the case of bright sources where you clearly see the object in the reconstructed frame, set `method=center` in the combination step (recipe `kmos_combine`).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 71 of 81 |

11.3 Process data when OBJECT and SKY are taken from two separate Observing Blocks

It is always recommended to observe object and sky within the same observing block. If this is done, the kmos workflow will include object and sky within the same dataset and process them together. If sky and object are from different OBs, they will be listed into two different datasets. In order to process your data you have to process these two datasets separately, without applying the sky subtraction. You can subtract the sky later, with the `kmo_arithmetic` or `kmos_sky_tweak` recipes via the `esorex` command line interface. To do so, execute the following steps:

1. Double-click on the Cube Reconstruction actor (`kmos_sci_red` recipe) and specify the following parameters:

```
init_no_subtract = TRUE
init_sky_tweak = FALSE
```

2. Run the workflow on the OBJECT dataset. Save the resulting `SCI_RECONSTRUCTED` files in a separated directory (e.g., `object_cubes`)
3. Run the workflow on the SKY dataset. Save the resulting `SCI_RECONSTRUCTED` files in a separated directory (e.g., `sky_cubes`).
4. If you want to remove the sky without applying the sky tweaking algorithm, execute the `kmos_arithmetic` recipe:

```
esorex kmo_arithmetic --operator='-' SCI_RECONSTRUCTED_object.fits
      SCI_RECONSTRUCTED_sky.fits
```

where `SCI_RECONSTRUCTED_object.fits` `SCI_RECONSTRUCTED_sky.fits` are the reduced and reconstructed datacubes of the OBJECT and SKY observation, respectively.

5. If you want to remove the sky and applying the sky tweaking algorithm (suggested) do as follows. Create a `SetOfFrames` file (e.g., `data.sof`) with an editor; the `sof` file must contain 1 object cube (not sky subtracted) and 1 sky cube (not sky subtracted). An example of the content of such `data.sof` file is:

```
object_cubes/SCI_RECONSTRUCTED_object.fits OBJECT_CUBE
sky_cubes/SCI_RECONSTRUCTED_sky.fits SKY_CUBE
```

where `SCI_RECONSTRUCTED_object.fits` `SCI_RECONSTRUCTED_sky.fits` are the reduced and reconstructed datacubes of the OBJECT and SKY observation, respectively.

Now, remove the sky from the object cube by executing:

```
esorex kmos_sky_tweak data.sof
```

The above instructions work if the object is at the same IFU number as the associated sky. If not, one has to extract the n -th IFU from the object and the corresponding m -th IFU from the sky and process them individually. The pipeline recipe `kmo_fits_strip` can be used to extract a given IFU from the `SCI_RECONSTRUCTED` cubes.

| | | | |
|------------|----------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 72 of 81 |

11.4 Improving the resolution of spectral line

Sometimes it is possible to improve the spectral resolution on the final combined datacube, and therefore to better sample the profile of the lines of the science object, by increasing the sampling in wavelength.

One can try to increase the **number of wavelength samples** parameter in the main canvas from 2048 (default) to 3075 (i.e., increase it by a factor 1.5). This will improve the sampling in wavelength and can result in a better spectral resolution in the final combined datacube (`COMBINED_CUBE`).

11.5 Optimize the alignment of exposures before combination

The combination procedure uses the information on the headers to combine the single datacubes that target the same object. This can led to spatial offsets (3 pixels at most), if the exposures are taken at different epochs (e.g. when combining multiple OBs).

If all targets are bright, we recoment to use the centroid algorithm (`-method=center` and `-fmethod = gauss` or `moffatt`).

If there are few bright targets (let's say 1 per detector) one strategy could be to combine first the bright targets. The recipe will save in the bookkeeping directory of `kmos_combine` the adopted offsets. Then, these offsets, optimized on bright targets, can be used for fain targets too; one has to specify the offset file to `kmos_combine` via the recipe parameter `-filename` and select `-method=user`.

11.6 Global background subtraction (post-processing)

The pipeline background subtraction can led to global over or undersubtraction of the background level, and this can hava spatial dependency. At the moment, the only way to coope with that is via post-processing. One can fit the reconstructed image using algorithms like `galfit` (Peng et al. 2006) or `imfit` (Erwin et al. 2015). The background can be modelled as a 1-st or 2nd order surface (see Figure 11.1. The modeled background can be then subtracted from the cube.

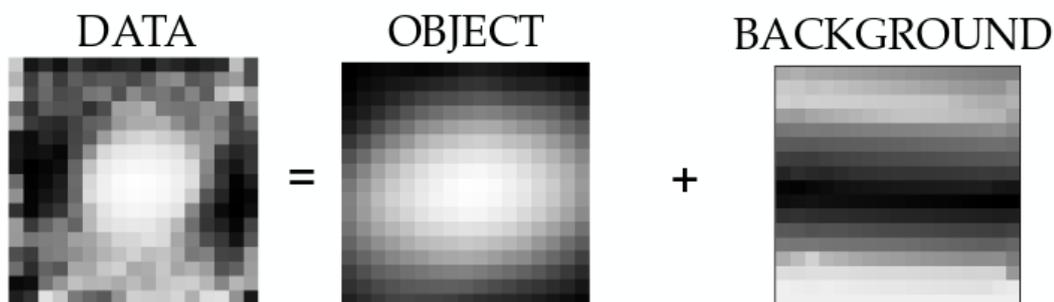


Figure 11.1: An example on how to remove residual background with post-processing (courtesy of D. Wilman). The data (left panel) are modeled by the contribution of the object (central panel, a galaxy with a Sersic profile in this example) and the residual background (right panel, a 2nd order polynomial surface)..

| | | | |
|-----|----------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 73 of 81 |

12 Frequently Asked Questions

- **The error window fills the whole screen - how can I get to the `Continue`/`Stop` buttons?**

Press the `Alt` key together with your left mouse button to move the window upwards and to the left. At the bottom the `Continue`/`Stop` buttons will be visible. This bug is known but could not yet be fixed.

- **I tried to Open (or Configure) an Actor while the workflow is running and now it does not react any more. What should I do?**

This is a limitation of the underlying Kepler engine. The only way out is to kill the workflow externally. If you want to change anything while a workflow is running you first need to pause it.

- **After a successful reduction of a data set, I changed this data set in some way (e.g. modified or removed some files, or changed the rules of the Data Organizer). When I restart Reflex, the Data Set Chooser correctly displays my new data set, but marks it as “reduced ok”, even though it was never reduced before. What does this mean?**

The labels in the column “Reduced” of the Data Set Chooser mark each dataset with “OK”, “Failed” or “-”. These labels indicate whether a data set has previously successfully been reduced at least once, all previous reductions failed, or a reduction has never been tried respectively. Data sets are identified by their name, which is derived from the first science file within the data set. As long as the data set name is preserved (i.e. the first science file in a data set has not changed), the Data Organizer will consider it to be the same data set. The Data Organizer recognizes any previous reductions of data sets it considers to be the same as the current one, and labels the current data set with “OK” if any of them was successful, even if the previously reduced data set differs from the current one.

Note that the Product Explorer will list all the previous reductions of a particular data set only at the end of the reduction. This list might include successful and/or unsuccessful reduction runs with different parameters, or in your case with different input files. The important fact is that these are all reductions of data sets with the same first raw science file. By browsing through all reductions of a particular raw science file, the users can choose the one they want to use.

- **Where are my intermediate pipeline products?** Intermediate pipeline products are stored in the directory `<TMP_PRODUCTS_DIR>` (defined on the workflow canvas, under Setup Directories) and organised further in directories by pipeline recipe.
- **Can I use different sets of bias frames to calibrate my flat frames and science data?** Yes. In fact this is what is currently implemented in the workflow(s). Each file in a DataSet has a purpose attached to it ([5]). It is this purpose that is used by the workflow to send the correct set of bias frames to the recipes for flat frame combination and science frame reduction, which may or may not be the same set of bias frames in each case.
- **Can I run Reflex from the command line?** Yes, use the command:

```
esoreflex -n <workflow_path>/<workflow>.xml
```

The `-n` option will set all the different options for Kepler and the workflows to avoid opening any GUI elements (including pipeline interactive windows).

It is possible to specify workflow variables (those that appear in the workflow canvas) in the command line. For instance, the raw data directory can be set with this command:

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 74 of 81 |

```
esoreflex -n -RAW_DATA_DIR <raw_data_path> \
    <workflow_path>/<workflow>.xml
```

You can see all the command line options with the command `esoreflex -h`.

Note that this mode is not fully supported, and the user should be aware that the path to the workflow must be absolute and even if no GUI elements are shown, it still requires a connection to the window manager.

- How can I add new actors to an existing workflow?** You can drag and drop the actors in the menu on the left of the Reflex canvas. Under `Eso-reflex -> Workflow` you may find all the actors relevant for pipeline workflows, with the exception of the recipe executer. This actor must be manually instantiated using `Tools -> Instantiate Component`. Fill in the “Class name” field with `org.eso.RecipeExecuter` and in the pop-up window choose the required recipe from the pull-down menu. To connect the ports of the actor, click on the source port, holding down the left mouse button, and release the mouse button over the destination port. Please consult the Reflex User Manual ([5]) for more information.
- How can I broadcast a result to different subsequent actors?** If the output port is a multi-port (filled in white), then you may have several relations from the port. However, if the port is a single port (filled in black), then you may use the black diamond from the toolbar. Make a relation from the output port to the diamond. Then make relations from the input ports to the diamond. Please note that you cannot click to start a relation from the diamond itself. Please consult the Reflex User Manual ([5]) for more information.
- How can I manually run the recipes executed by Reflex?** If a user wants to re-run a recipe on the command line he/she has to go to the appropriate `reflex_book_keeping` directory, which is generally `reflex_book_keeping/<workflow>/<recipe_name>_<number>`. There, subdirectories exist with the time stamp of the recipe execution (e.g. `2013-01-25T12:33:53.926/`). If the user wants to re-execute the most recent processing he/she should go to the `latest` directory and then execute the script `cmdline.sh`. Alternatively, to use a customized `esorex` command the user can execute

```
ESOREX_CONFIG="INSTALL_DIR/etc/esorex.rc"
PATH_TO/esorex --recipe-config=<recipe>.rc <recipe> data.sof
```

where `INSTALL_DIR` is the directory where Reflex and the pipelines were installed.

If a user wants to re-execute on the command line a recipe that used a specific raw frame, the way to find the proper `data.sof` in the bookkeeping directory is via `grep <raw_file> */data.sof`. Afterwards the procedure is the same as before.

If a recipe is re-executed with the command explained above, the products will appear in the directory from which the recipe is called, and not in the `reflex_tmp_products` or `reflex_end_products` directory, and they will not be renamed. This does not happen if you use the `cmdline.sh` script.

- If I enter “-” into an empty integer parameter of an interactive window it is automatically completed to “-1”. Why?**

The parameters are validated for correctness according to their type (e.g. string, integer, float). In the case of an integer or float parameter “-” alone is considered an invalid input and is therefore automatically completed to “-1”. This is part of the validation of input done by the WxPython library.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 75 of 81 |

- **Can I reuse the bookkeeping directory created by previous versions of the pipeline?**

In general no. In principle, it could be reused if no major changes were made to the pipeline. However there are situations in which a previously created bookkeeping directory will cause problems due to pipeline versions incompatibility. This is especially true if the parameters of the pipeline recipes have changed. In that case, please remove the bookkeeping directory completely.

- **How to insert negative values into a textbox?**

Due to a bug in wxPython, the GUI might appear to freeze when attempting to enter a negative number in a parameter's value textbox. This can be worked around by navigating away to a different control in the GUI with a mouse click, and then navigating back to the original textbox. Once focus is back on the original textbox the contents should be selected and it should be possible to replace it with a valid value, by typing it in and pressing the enter key.

- **I've updated my Reflex installation and when I run esoreflex the process aborts. How can I fix this problem?**

As indicated in Section 3, in case of major or minor (affecting the first two digit numbers) Reflex upgrades, the user should erase the `$HOME/KeplerData`, `$HOME/.kepler` directories if present, to prevent possible aborts (i.e. a hard crash) of the esoreflex process.

- **How can include my analysis scripts and algorithms into the workflow?**

EsoReflex is capable of executing any user-provided script, if properly interfaced. The most convenient way to do it is through the Python actor. Please consult the tutorial on how to insert Python scripts into a workflow available here: www.eso.org/sci/data-processing/Python_and_esoreflex.pdf

| | | | |
|-----|----------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 76 of 81 |

13 Troubleshooting

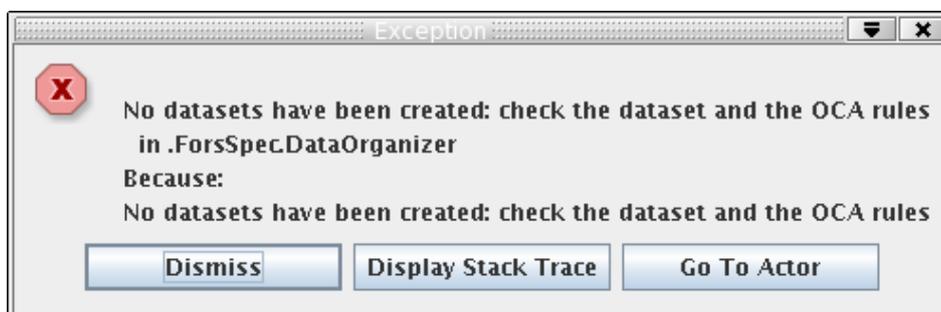


Figure 13.1: *TheDataOrganizer* interactive window reports an error “:No DataSets have been created, check the data set and the OCA rules.”.

1. I downloaded the data from the ESO archive, put them into a new directory, tried to run `Reflex` on them, but

(a) it crashes

This may happen if one of the files was downloaded only partially (check for a file with the extension `fits.Z.part`. You will have to download that file again in order to have an uncorrupted file (and remove the partial one).

(b) it fails with error message “:No DataSets have been created, check the data set and the OCA rules.”(see Figure 13.1.)

This error may be due to the fact that the data provided by the ESO archive are compressed (`<filename>.fits.Z`). Please remember to uncompress the data before executing `Reflex`.

(c) all DataSets are greyed out in the DataSets interactive window.

The ESO archive used with `CalSelector` does not always supply all static calibration files. As a consequence some/all DataSets are greyed out because they were missing such required data.

Missing static calibration should be found by `reflex` in

```
<install_directory>/calib/<pipeline_version>/cal.
```

2. The “Select DataSets” window displays my DataSets, but some/all of them are greyed out. What is going on?

If a DataSet in the “Select DataSets” window is greyed out, then it means that the DataSet that was constructed is missing some key calibration(s) (i.e. the DataSet is incomplete). To find out what calibration(s) are missing from a greyed out DataSet, click on the DataSet in question to highlight it in blue, and then click on the button `Inspect Highlighted`. The “Select Frames” window that appears will report the category of the calibration products that are missing (e.g. `MASTER_BIAS`). From this the user has then to determine the missing raw data (in this case bias frames). If static calibrations are missing the mechanism unfortunately does not work, but should be found by `reflex` in

```
<install_directory>/calib/<pipeline_version>/cal
```

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 77 of 81 |

3. The plots in the interactive windows does not allow me to properly inspect the products; how can I change or measure what it is plotted?

The plots in the interactive window are meant to provide a general visualization of the product. However, it is possible to inspect each file (input or output) with other visualization tools, or process them via custom scripts to evaluate the quality.

At the bottom right corner of each interactive window, the list of inputs/outputs files is given. Select with the mouse the file you would like to inspect and press Ctrl-C to copy its full path name.

It is also possible to change the general visualization tool in the Global Parameter section of the workflow, by editing the `FITS_VIEWER` variable.

4. The Spectrum extracted in a single spaxel in the reconstructed cubes show fluctuations in the level of continuum, a wavy pattern with wavelength. What is happening?

Exposures of bright and point-like sources taken in good seeing conditions (e.g., $leq 0.5$ arcsec) can display wavy-pattern in the shape of the continuum. This is due to the fact that the spatial PSF of the source is not properly sampled by the detector, and the reconstruction algorithm is not able to reconstruct the shape of the PSF. Using the Cubic Spline method in the reconstruction of standard star and scientific spectra help in mitigating the problem. If the patterns are still present, it is recommended to extract the source spectrum over a circular aperture (e.g., using the recipe `kmoss_extract_spec`, or visualization software like QFitsView); a radius of 2-3 pixels usually levels the fluctuations out and fixes the problem. Alternatively, one can smooth with a Gaussian Kernel the reconstructed cube or directly the raw data along the spatial direction. A Gaussian kernel with a sigma of 1 pixel or a top-hat kernel of 3 pixels are enough. The disadvantage of the kernel smoothing, is that the spatial resolution is deteriorated in both direction (if convolving the cube) or along one direction (if convolving the raw data). The recommended procedure is the extraction over a circular aperture. Among the alternative approaches, the recommended is to convolve the reconstructed cubes to avoid elongated PSF in the final results.

Extended sources are as less affected by this problem because their size is larger than the seeing.

5. The illumination procedure fails to process twilight sky flats.

The illumination procedure expects the twilight flats to be taken at the same rotator angle, as foreseen by the standard calibration plan. In some rare cases, the twilight flats are taken at different rotator angles, and the recipe fails to process them. To avoid this issue, it is advisable to use the internal flat to generate the illumination correction (done by setting the variable `UseSkyFlats?` in the main workflow canvas to “no”). If the user wants to use the twilight flats anyway, the best strategy is to change the angle in the header (header keyword: `HIERARCH ESO OCS ROT NAANGLE`) of the twilight raw flats and set them to a common value.

6. `kmoss_combine`: Cube Orientation mismatch error when processing multi-OB datasets

The KMOS recipe `kmoss_combine` triggered by the Cubes combination actor requires that the cubes to combine have the same orientation on the sky, i.e., the data are obtained with the same rotator offset. The value of the rotator is indicated in the header keyword `OCS .ROT .OFFANGLE`. In some rare cases, some datasets contain data taken with different angles, and therefore the Cubes Combination actor returns a window with the error `kmoss_combine: Cube Orientation mismatch`.

To avoid this, proceed as follows.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 78 of 81 |

- Reduce the individual OBs that contain the exposures to combine. Data will be saved the directory specified by the workflow, with the latest time stamp.
- Run the workflow again, specifying as `RAW_DATA_DIR` the directory that contains the results obtained at the step above. The workflow organizes the data so that cubes with the same orientation on the sky will be combined.

To combine data with different rotator angle, there are 2 strategies.

- (a) If the sources are point like, it is recommended to extract the spectra from each individual datacube, and then combine the spectra.
- (b) If the user wants to combine datacubes with different orientation, then (s)he has to de-rotate them with the recipe `kmos_rotate`. The value to derotate is `- OCS.ROT.OFFANGLE`. After the reconstructed cubes have been de-rotated, it is possible to combine them via the `kmos_combine` recipe (command line call via `esorex`), or via the KMOS workflow (if the de-rotated cubes are stored in the `RAW_DATA_DIR` directory).

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 79 of 81 |

Acknowledgements

The `Reflex` team in alphabetical order consists of Pascal Ballester, Daniel Bramich, Vincenzo Forchì, Wolfram Freudling, César Enrique García, Maurice Klein Gebbinck, Andrea Modigliani, Sabine Moehler, Mark Neeser & Martino Romaniello.

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 80 of 81 |

- [1] Forchì V., 2012, Reflex User Manual, VLT-MAN-ESO-19000-5037, Issue 0.7
- [2] Davies R., Berbel, A., Wiezorrek E., Ott T., Schreiber N. 2012, KMOS Data Flow: Reconstructing Data Cubes in One Step, Max-Planck Institut für extraterrestrische Physik. [9](#)
- [3] Davies R., Berbel, A., Wiezorrek E. 2014, SPARK INstructional Guide ofr KMOS data, VLT-MAN-KMO-146611-009, issue 1.5. [46](#)
- [4] Cirasuolo M., Sharples R., 2013, KMOS User Manual, VLT-MAN-KMO-146603-001, Issue 5.0 [9](#)

| | | | |
|------------|-----------------------------|--------|------------------------|
| ESO | Reflex KMOS Tutorial | Doc: | VLT-MAN-ESO-19500-.... |
| | | Issue: | Issue 5.1 |
| | | Date: | Date 2019-07-30 |
| | | Page: | 81 of 81 |

[5] Forchì V. *Reflex User's Manual*. ESO/SDD/DFS, <http://www.eso.org/reflex/>, 2012. VLT-MAN-ESO-19000-5037. 22, 73, 74