

# DR01/02 Topical Report: Workflow

1st April 2011

## 1 Introduction

The goal of project A of the Austrian in-kind contribution is to provide software modules for the reduction of imaging data from VLT instruments operating in the optical and near-infrared (NIR) wavelength regimes. The deliverables consist in recipes that are executed via `esorex`, either directly from the command line or through a graphical interface (`gasgano`, `Kepler/Reflex`). Each recipe performs one specific task, for example the subtraction of an estimate of the sky background from the science exposures. The development of the recipes is based on the Common Pipeline Library (CPL) and creates products that are compliant with data flow system (DFS) requirements.

The reduction of a set of science data requires a series of recipes to be executed in sequence; this sequence is referred to as a *workflow*. In this document we describe the conceptual workflows that guide the development of the recipes for this project. The workflows have been implemented in `kepler` (Sect. 2) in collaboration with Pascal Ballester and Armin Gabasch. At the time of writing, no executable `Kepler` workflow is used by the Vienna team as developers have found it more convenient to use shell scripts for testing individual recipes or groups of recipes in their bottom-up approach to the development of the modules.

The workflows described here aim at providing standard reduction procedures for the imaging instruments on the Very Large Telescope (VLT) with the most common observing strategies. It is not *a priori* possible to cover all observing situations that might arise. The modularity of the recipes gives the user the freedom to adapt the workflows to their particular needs.

## 2 ESO Reflex

The ESO Reflex system provides **actors** for `kepler`, an environment to execute scientific workflows. A components of a workflow in `kepler` are called “actors” which perform certain tasks. The execution of a workflow is controlled by a “director” according to certain execution models. In our case, the most common task of an actor is to perform a system call to `esorex` which in turn executes a particular recipe. Actors are available to perform additional tasks such as listing directories or displaying graphics. Composite actors contain “sub-workflows” consisting of a series of actors that together perform a specific task. Workflows are stored as `xml` files but are displayed graphically as flow diagrams.

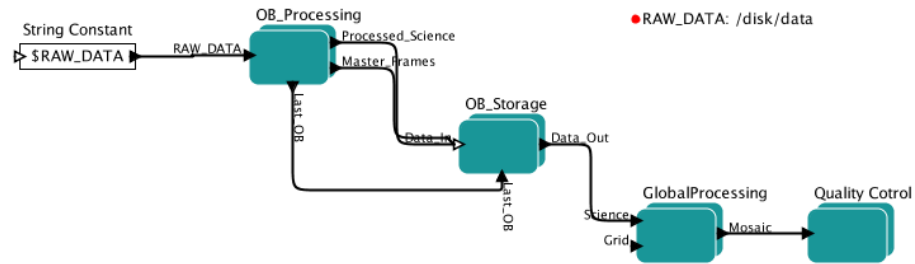


Figure 1: Basic structure of the workflow for imaging data.

Workflows can be modified in the graphical environment by moving components around or changing connections between components.

### 3 General workflow for imaging data

The top level of the conceptual workflow for imaging data is shown in Fig. 1. The reduction workflow has two main parts which are indicated by the composite actors `OB_Processing` and `GlobalProcessing`. The input data include all science and calibration data that are needed to produce a mosaic image of a given field, the output of the workflow. Typically, these data come from several or many observing blocks (OBs). The first steps in the data reduction operate on data from a single OB whereas the later parts leading up to the mosaicing of the exposures combines the data from all OBs of the field.

#### 3.1 Observing block processing

Reduction steps that require calibration images to have been collected close in time to the exposure of the science frames that are to be corrected are performed on data from a single OB; this concerns in particular preprocessing (bias, dark, flat field) and sky background correction in the NIR. The contents of the composite actor `OB_Processing` are shown in Fig. 2. The actor `OB_DataOrganiser` sorts the input data by the observing blocks during which they were obtained. The data from each OB are then fed to the composite actor `OB_Preprocessing` which contains tasks such as creation and application of master bias, dark and flat field frames.<sup>1</sup> The resulting master calibration frames are archived.

The preprocessed science images and possibly the corresponding noise images are transferred to the next composite actor, “Background”, for estimation and subtraction of the sky and other additive background contributions. Different methods have to be applied according to the type of data that are to be processed and consequently the workflow inside `Background` (Fig. 3) has several branches which are selected by the workflow parameter `Method`.

The methods and their implementation in terms of CPL recipes developed by us is described in much detail in the report *DR01: Sky subtraction*. For completeness, we list here the recipes corresponding to the actors in the different workflows (Figs. 4 and 5).

<sup>1</sup>Preprocessing is not part of the Austrian in-kind project but is expected to be done using the instrument-specific pipelines.

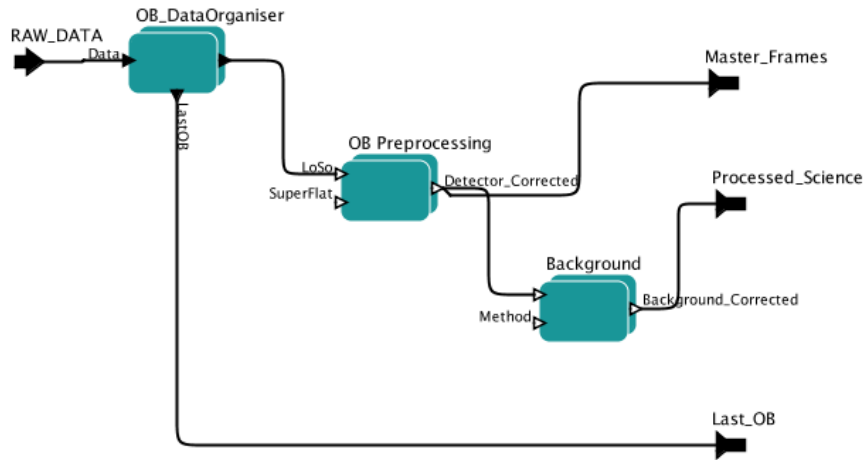


Figure 2: Sub-workflow performing reduction steps applied on an observing block basis. The series of actors shown here are grouped as the composite actor `OB_Processing` in Fig. 1. The sub-workflow loops over all OBs until the last OB is reached.

The sub-workflow `Optical` shown in Fig. 4 is designed to work on data which have a background that varies smoothly on scales that are large compared to the size of the largest objects in the field. This type of background is typical for observations in the optical wavebands. The relevant recipes<sup>2</sup> are

- `esosoft_create_mask`: actor `Detect_obj_1`
- `esosoft_compute_smooth_bkg`: actor `Compute_bkg_1`
- `esosoft_subtract_bkg`: actor `Subtract_bkg_1`

A iterative loop over these steps can easily be implemented, such that new masks are created from either the individual background-subtracted images from the previous iteration or from an astrometrically correct stack of these, allowing for masking of faint objects which might bias the background estimation.

The sub-workflow `NIR_Jitter` shown in Fig. 5 applies to the strongly, both spatially and temporally variable background seen in near-infrared observations, e.g. with HAWK-I and ISAAC. The method in this sub-workflow estimates the background from the series of dithered/jittered science exposures by a two-pass method. The correspondence of recipes to actors is as follows:

- `esosoft_compute_bkg`: actors `Compute_bkg_1` and `Compute_bkg_2`
- `esosoft_subtract_bkg`: actors `Subtract_bkg_1` and `Subtract_bkg_2`
- `esosoft_shift_and_add`: actors `Combine_1` and `Combine_QC`
- `esosoft_create_mask`: actor `Detect_obj_1`

<sup>2</sup>The prefix “esosoft” is likely to change in the future.

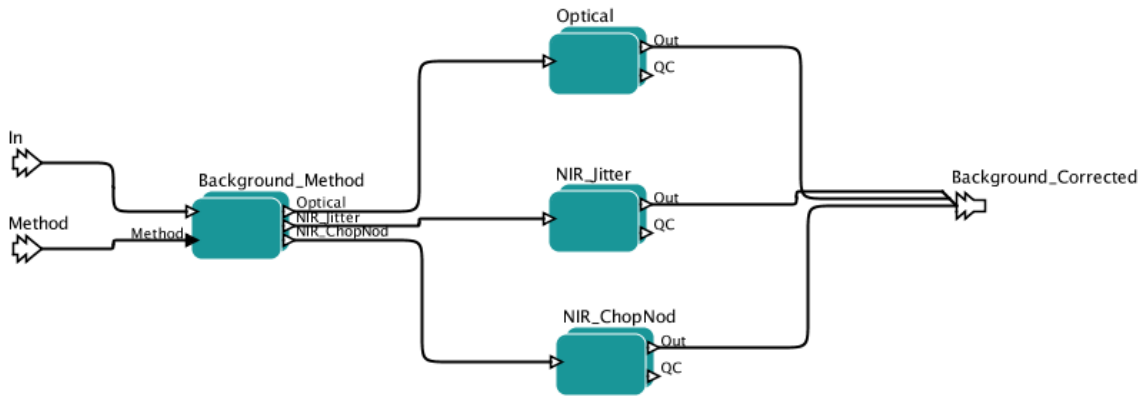


Figure 3: Contents of composite actor Background (Fig. 2). The workflow parameter Method selects which background estimation method to use.

The last recipe in this list creates mask for objects detected in an astrometrically correct stack of the preliminary background-subtracted images. Implementing a loop over these steps (i.e. expanding the two-pass method to an  $n$ -pass method) should present no difficulties. In the final step of this sub-workflow, a new astrometrically correct stack of the corrected exposures is created and stored for quality control purposes along with the background-subtracted exposures.

The third method indicated by a composite actor in Fig. 3 refers to the case where in-between a series of science exposures over the course of an OB the telescope is pointed (“nodded”<sup>3</sup>) at a nearby empty patch of sky. During the reduction, exposures taken at the sky position are combined to create a correction image that is subtracted from the exposures taken at the object position. In the report *DR01 - Sky subtraction* we describe an application of this method. However, a Reflex workflow has not yet been implemented because only one data set using this strategy is available and it is not clear that the observing pattern used there is typical for the application of this method with ESO instruments.

### 3.2 Global processing and mosaicing

At this stage, the exposures are essentially science-ready.<sup>4</sup> It is desired, however, to co-add all the exposures from a given target field into one “mosaic” image. To this end, the individual exposures have to be astrometrically calibrated beyond the standard “linear” WCS that is written to the fits headers at the time of observation and then in general rebinned to the pixel grid of the output mosaic image. Indeed, in order to co-add two exposures taken at neighbouring target positions such that there is only a small region of overlap requires determining the distortions to sufficient accuracy that the two images

<sup>3</sup>We refer to “nodding” as a telescope movement which explicitly goes to a different target position, as opposed to “dithering” which uses small telescope offsets but pointing at essentially the same position.

<sup>4</sup>Except for photometric calibration. A calibration based on observations of standard stars is, however, not part of the work statement for the Austrian in-kind project. One might try to identify reference objects in the exposures or (more usefully) in the final mosaic but while this approach is adequate for NIR data (using the 2MASS catalogue as reference) it is not for optical data (due to differing filter curves).

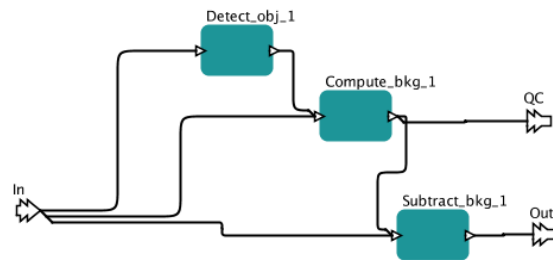


Figure 4: Estimation and subtraction of smooth background components. These are typical for imaging in the optical wavebands. The sub-workflow shown here corresponds to the composite actor `Optical` in Fig. 3.

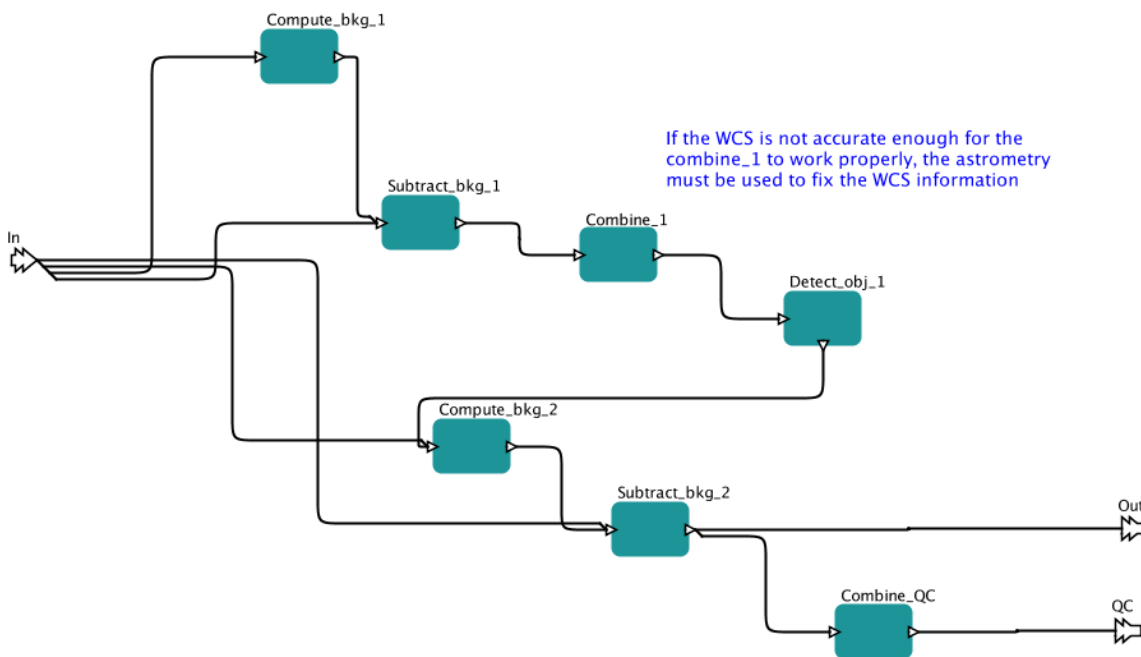


Figure 5: Estimation and subtraction of spatially and temporally variable background components from a series of dithered/jittered science exposures. This type of background is typical for near-infrared observations but the method can also be applied to fringes in red optical wavebands. The sub-workflow shown here corresponds to the composite actor `NIR_Jitter` in Fig. 3.

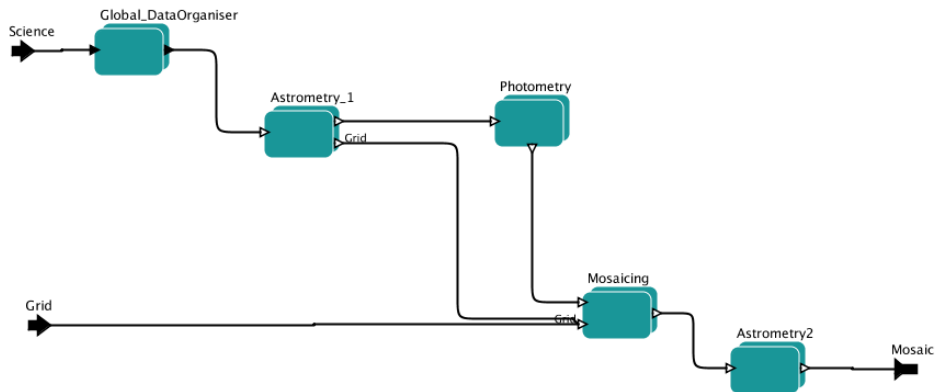


Figure 6: Sub-workflow `GlobalProcessing` from Fig. 2 taken from the conceptual workflow `Mosaicing.xml` created in March 2010. This is for illustrative purposes only, as the real workflow has changed since then.

of the same object in the overlap region are mapped to the same positions in the output grid so that the PSF of the sum of the two images is not degraded compared to the PSF in the input images.

A simple approach would calibrate each exposure separately on the basis of identification of objects in the exposure with a reference catalogue. The achievable accuracy is limited by the number density of objects in the reference catalogue and their positional accuracy which is often worse than what is reached in the deep pointed observations made with modern telescopes and instruments. For a precise calibration for the purposes of mosaicing the information contained in the overlap regions between exposures should be taken into account. The calibration procedure therefore assembles all exposures of the field taken during all OBs dedicated to that field, hence the astrometric calibration is part of the `GlobalProcessing` sub-workflow in Fig. 2. Also to this sub-workflow belong the (relative) photometric matching (by applying scaling factors to correct for differences in air mass and atmospheric transparency) and seeing matching (by convolution with a kernel to, alas, degrade the PSF of all exposures to the one with the largest seeing).

The sub-workflow for `GlobalProcessing` as conceived in March 2010 is shown in Fig. 6. Since then, it was decided to base the astrometric calibration on the tools from the `Terapix` software suite, and consequently the currently available recipes are wrappers for `SExtractor` (to create object catalogues from the exposures), `Scamp` (which performs the astrometric calibration) and `SWarp` (which resamples the input images to the output grid and co-adds them to the final mosaic). As a result, the workflow has changed in detail compared to the one shown in Fig. 6, and is described in the report *DR01 - Astrometry*. Here, we briefly describe how the current workflow compares to the concept from March 2010.

The place of `Astrometry_1` is taken by the wrapper recipes for `SExtractor` and `Scamp`, whereas `swarp` performs `Mosaicing`. The composite actor `Astrometry_1` contains the sub-workflow shown in Fig. 7. The idea was to branch according to whether a useable reference catalogue was available or not and whether there was sufficient overlap between exposures. It turns out that `Scamp` can handle all of these situations, including the one without a reference catalogue, hence branching is done by specifying input parameters for `scamp` rather than calling a separate sub-workflow. The case where neither reference catalogue nor overlap information are available is of course hopeless. Here, the only

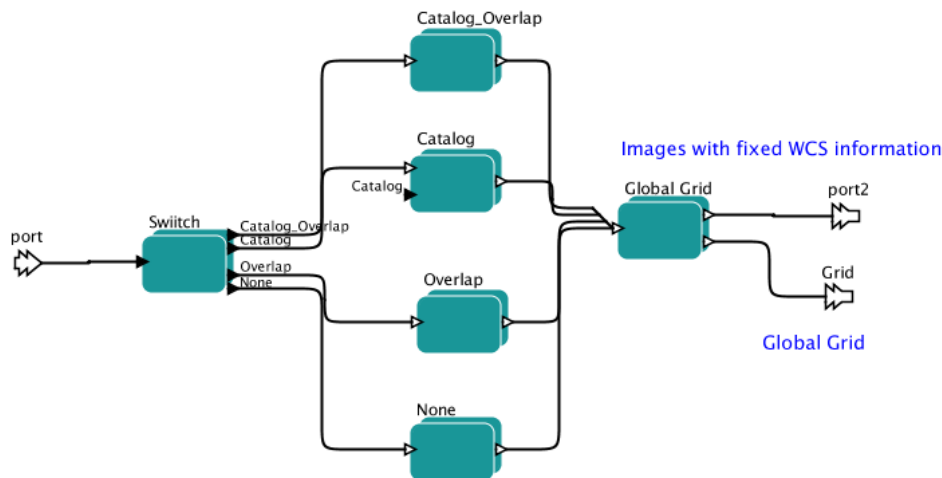


Figure 7: Sub-workflow `Astrometry_1` from Fig. 6 detailing the branching based on availability of an external astrometric catalogue.

remaining solution is to use the WCS created at observing time, i.e. to skip the astrometric calibration entirely.

The parameters of the “global grid” (center coordinates, pixel scale, size of the grid) are determined either automatically by `SWarp` based on the WCS information of the exposures or specified explicitly by the user. No separate recipe is required.

A recipe for photometric and seeing matching remains to be written. It will use the catalogues created by `SExtractor` to create matched object lists and determine the relative photometric scaling factors based on magnitude measurements and the required sizes of convolution kernels from measurements of the FWHM or half-light radii of stars, and apply these to the input images. The recipe has its place between `Scamp` and `SWarp`.

As discussed in the astrometry report, it is envisaged to take the final co-addition of the resampled images out of `SWarp`’s hands and to provide a recipe to do this using more sophisticated co-addition methods with outlier rejection.

The conceptual workflow for the astrometry contains a composite actor `Astrometry_2`. The idea here was to compare the output mosaic to an astrometric reference catalogue in order to detect any remaining shift (or to transfer the mosaic to a different astrometric system – not all astrometric reference catalogues are the same!). However, `Scamp` always uses a reference catalogue (and catalogues such as the USNO-B1.0 catalogue are always available) and hence produces absolute astrometric calibration as long as reference objects can be found in (some of) the exposures. A recipe for recalibrating the mosaic may be convenient for some purposes but does not appear to be urgently required.

## 4 Conclusions

As mentioned in this report, the Vienna team takes a bottom-up approach to the development of the software modules. Recipes are developed one by one and then tested individually or in small groups

of related recipes. Consequently, only limited attention has so far been given to designing an overall workflow or to using `Reflex` for running series of recipes. The recipe design strives for modularity so that the recipes can be combined in a very flexibly manner and workflows for different needs can be assembled without problem.