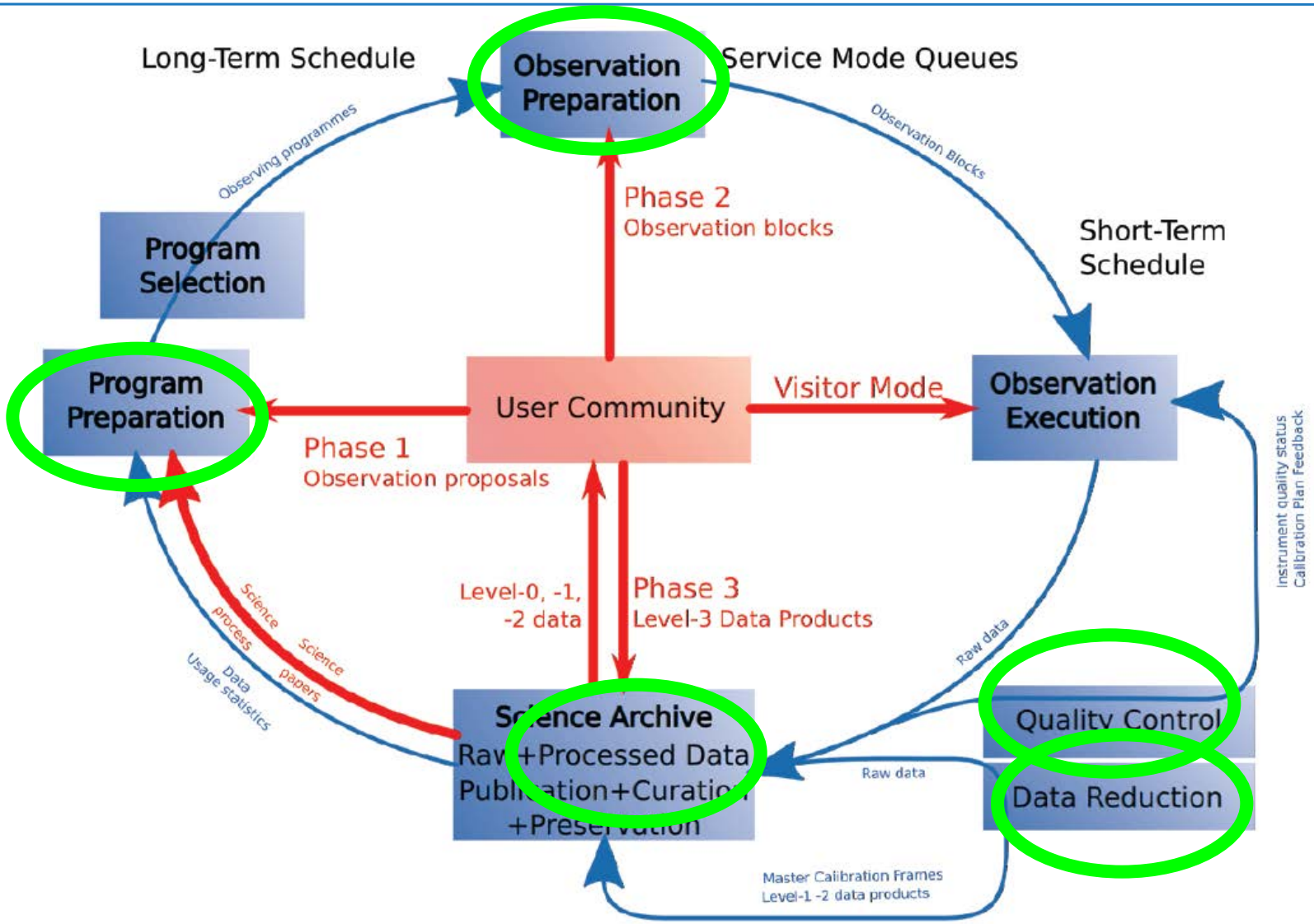
ESO's distributed Data Reduction Systems

Wolfram Freudling

European Southern Observatory

The end-to-end operation model



Usage of data reduction tools

Observatory

@telescope & ESO HQ

- Quality control at telescope and at ESO headquarter
- Processing of pre-imaging data
- Production and certification of master calibration frames
- Production of science data products for the archive facility.

Community

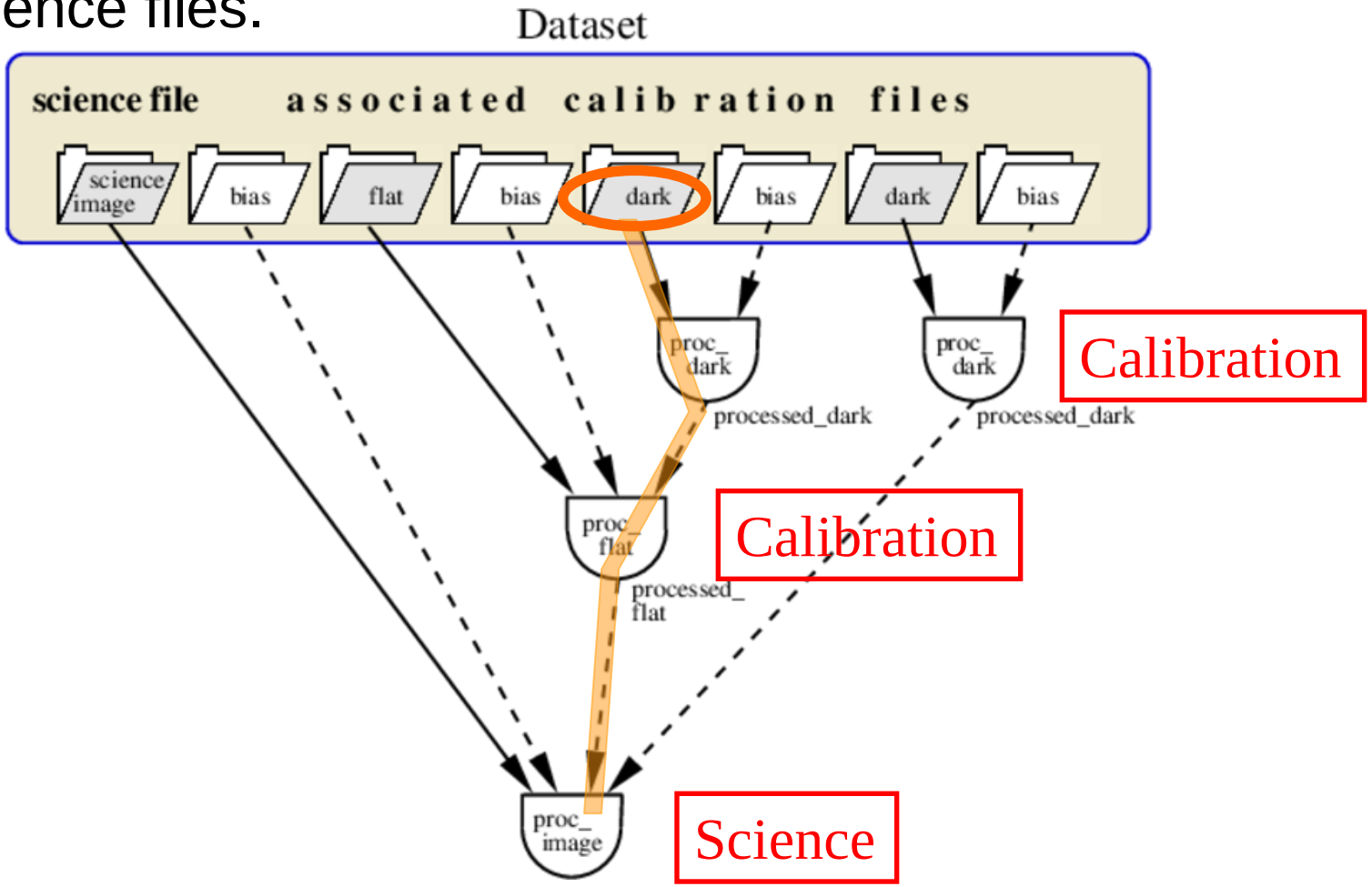
- Batch processing of data from observations or archive
- Interactive data reduction
- Optimizing science results

ESO Data Reduction Infrastructure

ESO Internal only	Public	communalities
Data organization		
DO at telescope, ABbuilder QC Garching	CalSelector service, Gasgano, Reflex DO	OCA language
Algorithms		
Quick look recipes	CPL pipeline recipes, Reflex Python plotting, (Molecfit and other tools)	CPL
Data Reduction workflows		
Paranal & QC cascades, Phoenix	Reflex	Esorex recipe execution

Data Organization

Organise files into data sets = all files needed to reduce one set of science files.



ESO Data Reduction Infrastructure

ESO Internal only	Public	communalities
Data organization		
DO at telescope, ABbuilder QC Garching	CalSelector service, Gasgano, Reflex DO	OCA language
Algorithms		
Quick look recipes	CPL pipeline recipes, Reflex Python plotting, (Molecfit and other tools)	CPL
Data Reduction workflows		
Paranal & QC cascades, Phoenix	Reflex	Esorex recipe execution

Data Organisation

- Pipeline (ESO Observatory, Data Organiser)
  - On-the-fly data processing (event driven)
  - Template-based processing
  - Static calibration database (only certified products are used)
- Quality Control (ESO Headquarters, ABbuilder)
  - Batch processing of complete data sets (all science and calibration data produced by one ESO instrument in one night)
  - Best available calibrations are used => data must be organized according to the **Calibration Cascade**
  - Science Archive (CalSelector)
  - Web-based association of calibration data in archive (raw and reduced) to science data
- Data Reduction by Community (Reflex, Gasgano)
  - Data organisation of files on disk for different processing cascades.

OCA Rules

- Data organisation defined as text files “OCA rules”
- Three types of rules:
  - Classification („This is a Raw Dark“)

```
if DPR.CATG=="CALIB" and DPR.TYPE==”DARK"  then
{
RAW.TYPE = "DARK";
}
```

  - Organization („These Raw Darks are processed together”)

```
select execute(MAKEDARK) from inputFiles
 where RAW.TYPE==”DARK“ group by TPL.START
```

  - Association (“select Biases based on properties of Raw Darks“ or “process these Raw Darks together using selected Biases“

```
action MAKEDARK
{
select file as MASTER_BIAS from calibFiles where
PRO.CATG=="MASTER_BIAS" and inputFile.DET.WIN1.BINX==DET.WIN1.BINX;
}
```

Algorithms

- Consortia are required to deliver pipeline recipes coded in C using the “ESO Common Pipeline Library” and “High

Instrument	Release Notes	Package	User Manual	Cookbook	Additional Documents	Additional Datasets	EsoReflex	Status
AMBER	2015-03-06	4.3.3	4.3.2					Operational on hold
CRIRES	2015-08-04	2.3.3	1.13	Cookbook				Operational on hold
EFOSC	2015-07-10	2.2.4	1.0			Demo Data		End of maintenance
FORS	2015-09-18	5.1.4	5.2			Demo Data (29 MB)	Tutorial: 1.0 (FORS-IMG) Tutorial: 1.0 (FORS-PMOS) Tutorial: 1.9 (FORS-SPEC) Demo Data: 0.8	Active
GIRAFFE	2015-11-02	2.14.2	2.14.2	Cookbook		Standard Calibration Files page		Operational on hold
HAWKI	2015-04-20	1.8.18	1.11			Demonstration Package (2,5 GB)		Operational on hold
ISAAC	2015-04-17	6.1.5	1.4			Static Calibration Files (50 MB)		End of maintenance
KMOS	2016-01-22	1.3.17	2.17				Tutorial: 1.6 Demo Data: 1.2	Active
MIDI	2015-04-15	2.8.4	2.8.3					End of maintenance
MUSE	2015-10-06	1.2.1	1.2.1			MUSE IFU 6 trace tables Leagacy MUSE static calibrations	Tutorial: 7.0 Demo Data: 1.3	Active
NACO	2015-06-01	4.4.1	1.1					Operational on hold
SINFONI	2015-10-26	2.7.0	19.5		ADA IV 2006 paper	Calibration Database Example (255 MB) Demonstration Package (1.2 GB)	Tutorial: 1.5 Demo Data: 0.2	Operational on hold
SOFI	2015-04-17	1.5.6	1.2					End of maintenance
SPHERE	2015-03-10	0.15.0						Active
UVES	2015-09-14	5.5.7	22.11 (UVES) 18.5 (UVES-FIBRE)			Demonstration Package (2.0 GB)	Tutorial: 6.6 (UVES) Tutorial: 1.5 (UVES-FIBRE) Demo Data: 4.4	Operational on hold
VIMOS	2015-10-05	3.0.6	7.0			Demonstration Package (1.7 GB)	Tutorial: 2.3 (VIMOS-IFU) Tutorial: 2.0 (VIMOS-MOS) Demo Data: 0.4	Active
VISIR	2016-02-25	4.1.7	1.5				Demo Data: 0.1	Operational on hold
XSHOOTER	2015-09-14	2.6.8	12.7			Additional NIR telluric model calatog (190 MB)	Tutorial: 2.6 Demo Data: 1.2	Operational on hold

- Scientific oversight by Garching instrument project scientists during development, Science Data Products Group and Paranal instrument scientist during operation

High level Data Reduction Library HDRL

A. Gabasch

- Overscan computation and subtraction
- Master frame combination(Bias/Dark/Flat)
- Cosmic ray detection on a single image
- Bad pixel determination on
  - Single images.
  - Stacks of identical images, e.g. bias/dark frames.
  - Sequence of images, e.g. domeflats with different exposure time.
- Computation of the Strehl ratio
- Fringe detection and removal
- Source detection/extraction

More to come:

- Optimal Extraction
- Wavelength calibration
- Detection of pick-up noise
- …

Data Reduction Workflows

- Even when algorithms (recipes) are given, data reduction involves many decisions:
  - Which data do I process first?
  - Which files are processed by which recipes?
  - Which files need to be re-processed?
  - How do I organize the output?
- Workflow differs for different use cases. Example:
  - QC0 use static calibration
  - QC1 processes all calibrations independent of later use
  - Desktop reduction processes all data for given datasets
  - Science data processing for archive uses pre-computed calibration files

Scientific Workflow Systems

- System to define and execute series of data manipulation steps
- Flexible intuitive workflow essential for desktop data reduction

- Reflex uses Kepler https://kepler-project.org
- Kepler provides the graphical user interface
- Reflex is a collection of Kepler components (“actors”) that allow to execute ESO recipes, data display GUIs, and Python scripts
- Available for Linux and OS X

DDF Director

HAWK-I Workflow (v. 1.0)

Workflow Instructions

To run this workflow on the demo data:

- **Turn on highlighting. Choose "Tools"-> "Animate at Runtime" from top menu and set it to "1".**
- **Press the "Run" button OR cntrl-R to start the workflow.**

To run on a different data set:

- **Click on ROOT_DATA_DIR and set as appropriate. All subdirectories of RAWDATA_DIR will be searched for data.**
- **If desired, change END_PRODUCTS_DIR.**
- **Press the "Run" button OR cntrl-R to start the workflow.**

To monitor the progress of the workflow in more detail:

- **Open "Window" -> "Runtime Window" in top menu before starting the workflow.**

Setup Directories

Input:

- ROOT_DATA_DIR: /sdp_test_data/mneeser/UK/assess_hawki_1.1/A1689
- RAWDATA_DIR: $ROOT_DATA_DIR/RAW
- CALIB_DATA_DIR: /sdp_test_data/mneeser/Reflex_workflows/Reflex_hawki_vimo...
- TWOMASS_CATALOGUE_DIR: /sdp_test_data/CASU_2MASS
- PPMXL_CATALOGUE_DIR: /sdp_test_data/CASU_PPMXL
- LOCAL_CATALOGUE_DIR:

Working Directories:

- BOOKKEEPING_DIR: $ROOT_DATA_DIR/reflex_book_keeping/hawki
- LOGS_DIR: $ROOT_DATA_DIR/reflex_logs/hawki
- TMP_PRODUCTS_DIR: $ROOT_DATA_DIR/reflex_tmp_products/hawki

Output:

- END_PRODUCTS_DIR: $ROOT_DATA_DIR/reflex_end_products

Global Parameters

= actor with interactive option

- RecipeFailureMode: Ask — Global parameter for the behaviour when a recipe fails. 'Ask': choice to continue or stop will be presented 'Continue': workflow will ignore errors and continue 'Stop': workflow will stop.
- EraseDirs: false — 'true': BOOKKEEPING_DIR, TMP_PRODUCTS_DIR and LOGS_DIR directories erased at start of workflow; Lazy mode will not work
- FITS_VIEWER: fv — name of local fits viewer/editor application to inspect files
- GlobalPlotInteractivity: true — Turn on/off interactive GUIs for all subworkflows NB: Enable_Interactivity inside subworkflows have precedence
- ProvenanceExplorerEnabled: true
- SelectDatasetMethod: interactive
- Compute readgain table: false — true: computed the readgain table from raw frames in the dataset. false: uses the readgain table from static calibrations. Default: false.

Catalogue Selection

Select catalogs (on local machine and/or for CDS search). Input values must be within the allowed ranges; at least one photometric and one astrometric valid catalogues must be provided (either from local machine or for CDS search). If these criteria are not fulfilled, the workflow will stop. No check is done beforehand.

- ASTROM_CATALOGUE: 1
- PHOTOM_CATALOGUE: 0

Select catalogues on local machine 0: 2MASS; 1: PPMXL; 2: LOCAL

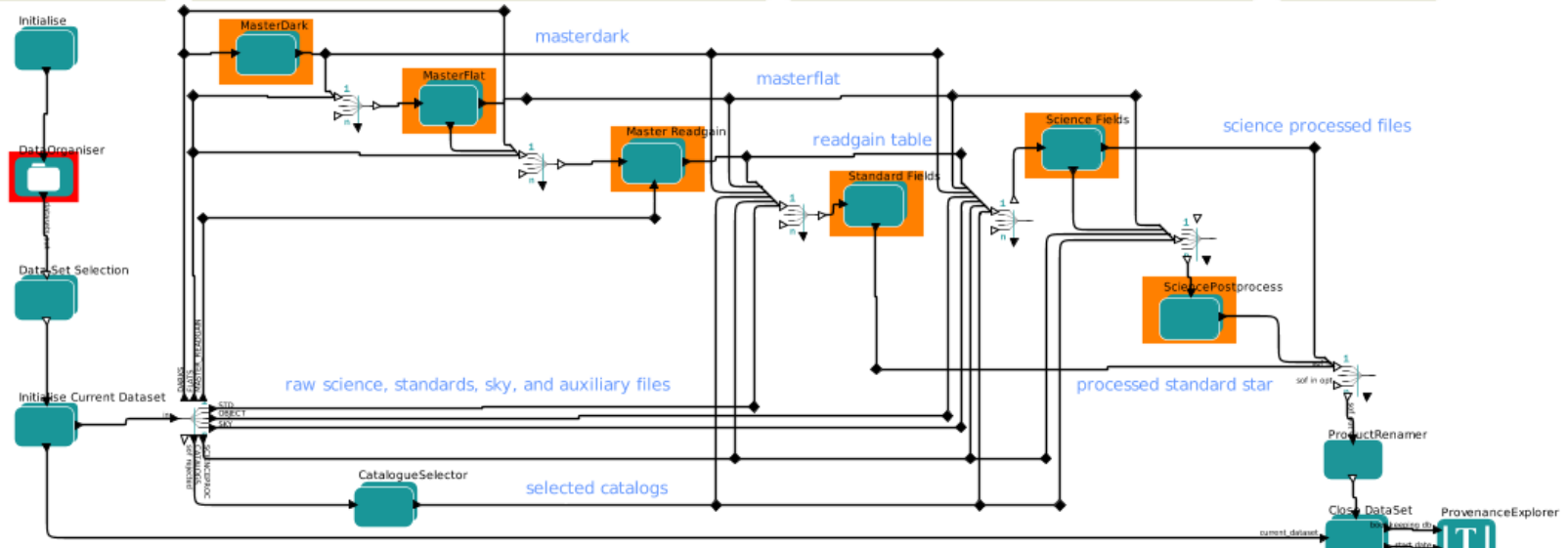
- CDS ASTROMETRIC CATALOGUE: wise — CDS astrometric catalogue, <none | 2mass | usnob | ppmxl | wise>
- CDS PHOTOMETRIC CATALOGUE: 2mass — CDS photometric catalogue, <none | 2mass | ppmxl | wise>

Step 1: Data Organisation and Selection

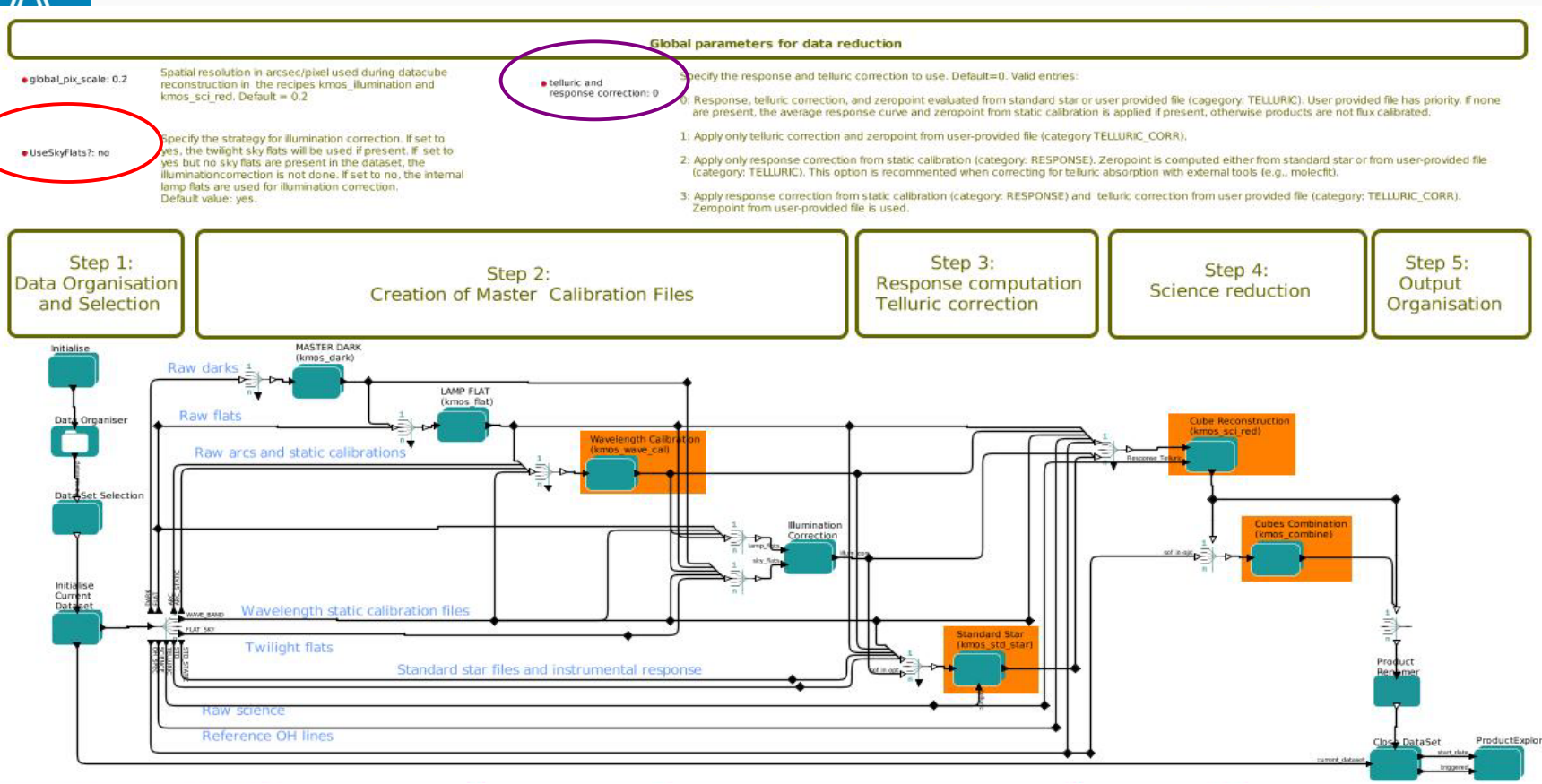
Step 2: Creation of Master Calibration Files

Step 3: Standard Star and Science Images Reduction

Step 4: Output Organisation



KMOS WORKFLOW

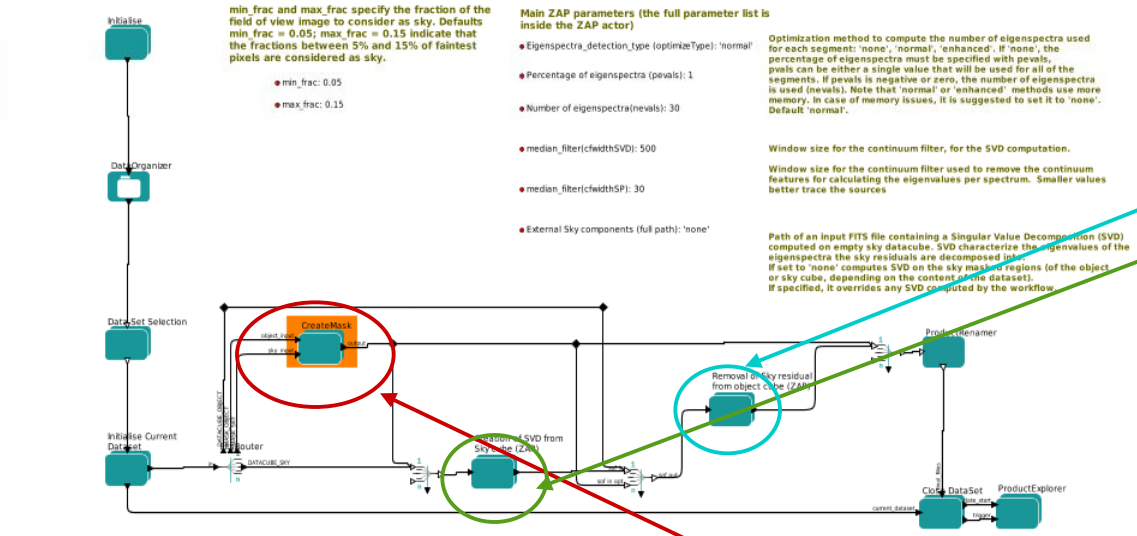


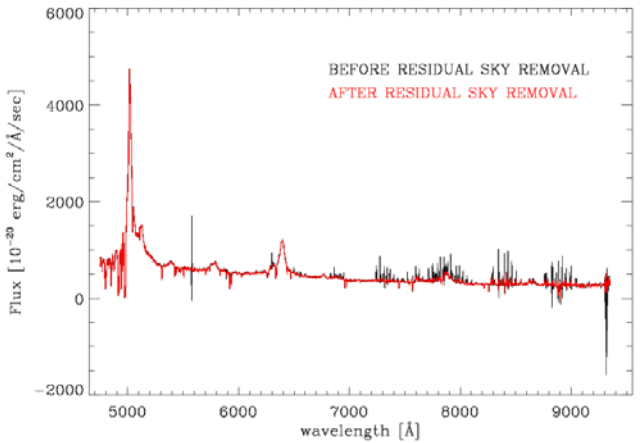
Global parameters that allows to define:

- The illumination correction strategy: twilight flats or lamp flats.
- The Telluric correction and response curve correction strategy:
  - Response and telluric correction from pipeline (using observed standard star)
  - Merge the static Response calibration with an user-provided telluric correction (obtained, e.g., using molecfit).
  - Response correction only (from static calibration).
  - Telluric correction only (from user-provided calibration)

MUSE-ZAP workflow

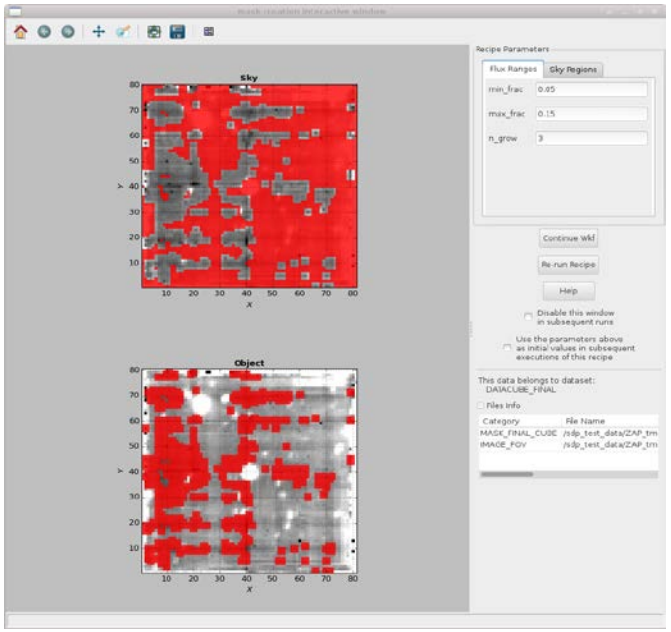
Step 1: Data Organisation and Selection	Step 2: Creation of Sky Mask	Step 3: Removing residual sky lines from datacube	Step 4: Output Organisation





Sky residuals are decomposed into
principal components either using:

- Sky regions around the target;
- Dedicated sky observations (if present in the dataset).



Interactive window to design the sky-mask.

- It uses the Python code by Soto et al. ( )
- It is independent of the muse pipeline.

Summary

- ESO supports data reductions for 3 different use cases:
  - Operations, quality control
  - Archive
  - Community
- Different use cases use different strategies and tools for
  - Data organization
  - Reduction workflow
- ESO's developped unique infrastructure to share data reduction components