



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

IIINSTRUMENT Pipeline User Manual

VLT-MAN-ESO-19500-XXXX

Issue 0.7

Date March 2014

Prepared: ESO IIINSTRUMENT Pipeline Team March 2014
Name Date Signature

Approved: P.Ballester, M. Peron
Name Date Signature

Released: P. Quinn
Name Date Signature

This page was intentionally left blank

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	3 of 40

Change record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
1.0	09/07/2004	All	First version
2.0	15/06/2005	All	Second version

This page was intentionally left blank

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	5 of 40

Contents

1	Introduction	7
1.1	Purpose	7
1.2	Acknowledgements	7
1.3	Scope	7
1.4	Reference and applicable documents	8
2	Overview	10
3	IIINSTRUMENT Instrument Description	11
3.1	Instrument overview	11
4	Quick start	12
4.1	IIINSTRUMENT pipeline recipes	12
4.2	An introduction to Gasgano and EsoRex	12
4.2.1	Using Gasgano	13
4.2.2	Using EsoRex	15
4.3	Example of data reduction using EsoRex	17
5	Known Problems	18
6	Instrument Data Description	19
6.1	General frames	19
6.2	Direct imaging frames	20
7	Static Calibration Data	22
7.1	Photometric table	22
7.2	Photometric catalog	22
8	Data Reduction	24
8.1	Data reduction overview	24
8.2	Required input data	25
8.3	Reduction cascade	26

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	6 of 40

9	Pipeline Recipes Interfaces	29
9.1	si_rec_detlin	29
9.1.1	Input	29
9.1.2	Output	30
9.1.3	Quality control	30
9.1.4	Parameters	30
9.2	si_rec_mdark	30
9.2.1	Input	30
9.2.2	Output	31
9.2.3	Quality control	31
9.2.4	Parameters	31
10	Algorithms	33
10.1	General Algorithms	33
10.1.1	Bad pixel cleaning	33
10.1.2	Cosmic rays removal	34
10.2	Recipes Algorithms	35
10.2.1	gi_rec_bias	35
10.2.2	gi_rec_dark	35
10.2.3	gi_rec_flat	35
A	Installation	36
A.1	Supported platforms	36
A.2	Building the IIINSTRUMENT pipeline	36
A.2.1	Requirements	36
A.2.2	Compiling and installing the IIINSTRUMENT pipeline	37
B	Abbreviations and acronyms	39
C	Troubleshooting Guide	40

ESO	IINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	7 of 40

1 Introduction

1.1 Purpose

The IINSTRUMENT pipeline is a subsystem of the *VLT Data Flow System* (DFS). It is used in two operational environments, for the *ESO Data Flow Operations* (DFO), and for the *Paranal Science Operations* (PSO), in the quick-look assessment of data, in the generation of master calibration data, in the reduction of scientific exposures, and in the data quality control. Additionally, the IINSTRUMENT pipeline recipes are made public to the user community, to allow a more personalised processing of the data from the instrument. The purpose of this document is to describe a typical IINSTRUMENT data reduction sequence with the IINSTRUMENT pipeline.

This manual is a complete description of the data reduction recipes implemented by the the IINSTRUMENT pipeline, reflecting the status of the IINSTRUMENT pipeline as of xx.xx.2006 (version 1.xx.xx).

1.2 Acknowledgements

The IINSTRUMENT pipeline is based on the SPIFFI Data Reduction Software developed by the Max-Planck-Institut für extraterrestrische Physik (MPE). We would like to thank the SPIFFI team for providing ESO with a complete and efficient data reduction software and for their help in documenting, testing, debugging the recipes and the pipeline during several commissioning and science verifications phases. We are particularly grateful to the MPE responsables for the data reduction: Jurgen Schreiber, Matthew Horrobin and Roberto Abuter for their contributions and support.

This release benefits also from the feedback provided by the IINSTRUMENT SV team and IINSTRUMENT instrument operations team. In particular we would like to thank Wolfram Howard (ESO, Data Flow Operations department) to have provided a lot of support to test the pipeline and several suggestions for improvements. Jutta Neumann, an ESO paid associate, very kindly tested the pipeline and provided valuable advice to improve from a scientific user prospective either the software and the documentation. Useful was the feedback provided by Joachim Hankel, an ESO paid associate, to improve the documentation.

1.3 Scope

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

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

Additional information on QFITS, the Common Pipeline Library (CPL) and ESOREX can be found respectively at [4], [5], [6]. The Gasgano tool is described in [14]. A description of the instrument is in [7]. The IINSTRUMENT instrument user manual is in [8] while results of Science Verifications (SV) are at [9].

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	8 of 40

1.4 Reference and applicable documents

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	9 of 40

[1] ESO/SDD/DFS, <http://www.eso.org/cpl/>. *CPL home page*. 36

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	10 of 40

2 Overview

In collaboration with instrument consortia, the Data Flow Systems Department (DFS) of the Data Management and Operation Division is implementing data reduction pipelines for the most commonly used VLT/VLTI instrument modes. These data reduction pipelines have the following three main purposes:

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

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

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

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

ESO offers two front-end applications for launching pipeline recipes, *Gasgano* [14] and *EsoRex*, both included in the pipeline distribution (see Appendix A, page 36). These applications can also be downloaded separately from <http://www.eso.org/gasgano> and <http://www.eso.org/cpl/esorex.html>. An illustrated introduction to Gasgano is provided in the "Quick Start" Section of this manual (see page 12).

The IIINSTRUMENT instrument and the different types of IIINSTRUMENT raw frames and auxiliary data are described in Sections 3, 6, and 7.

A brief introduction to the usage of the available reduction recipes using Gasgano or EsoRex is presented in Section 4. In section 5 we advise the user about known data reduction problems providing also possible solutions.

An overview of the data reduction, what are the input data, and the recipes involved in the calibration cascade is provided in section 8.

More details on what are inputs, products, quality control measured quantities, and controlling parameters of each recipe is given in section 9.

More detailed descriptions of the data reduction algorithms used by the individual pipeline recipes can be found in Section 10.

In Appendix A the installation of the IIINSTRUMENT pipeline recipes is described and in Appendix B a list of used abbreviations and acronyms is given.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	11 of 40

3 IIINSTRUMENT Instrument Description

IIINSTRUMENT has been developed by ESO and the Max-Planck-Institut für extraterrestrische Physik (MPE) in Garching.

The instrument has been made available to the community and started operations in Paranal on April 1st, 2005.

In this chapter a brief description of the IIINSTRUMENT instrument is given. A more complete documentation can be found in the IIINSTRUMENT User Manual, downloadable from <http://www.eso.org/instruments/sin>

3.1 Instrument overview

IIINSTRUMENT is a near-infrared (1.05 - 2.45 μm) integral field spectrograph (SPIFFI, developed by MPE) fed by an adaptive optics module (MACAO, developed by ESO, more details are given in [11]).

Figure 3.1: An inside view of SPIFFI: The cryostat cover and the reinforcing structure have been removed to provide a free view on the opto-mechanical components of SPIFFI. The light enters from the top, and passes the sky-spider. The pre-optics with a filter-wheel and interchangeable lenses provides three different image scales. The image slicer re-arranges the two-dimensional field into a pseudo-long slit, which is perpendicular to the base plate. Three diamond turned mirrors collimate the light onto the gratings. In total, four gratings are implemented on the grating drive. A multiple-lens system then focuses the spectra on a Rockwell HAWAII array. The diameter of the instrument is 1.3m.

The spectrograph operates with 4 gratings (J, H, K, H+K) providing a spectral resolution around 2000, 3000, 4000 in J, H, K, respectively, and 1500 in H+K - each wavelength band fitting fully on the 2048 pixels of the Hawaii 2RG (2kx2k) detector in the dispersion direction. The IIINSTRUMENT field of view on the sky is sliced into 32 slices.

ESO	IINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	12 of 40

4 Quick start

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

4.1 IINSTRUMENT pipeline recipes

The current IINSTRUMENT pipeline is based on a set of 8 stand-alone recipes involved in the data reduction cascade:

si_rec_detlin to evaluate the detector non linearity and generate a corresponding non linear pixel map.

si_rec_mdark to create a master dark and a hot-pixel map.

si_rec_mflat to create a master flat and a map of pixels which have intensities greater than a given threshold.

si_rec_distortion to compute the optical distortions and slitlets distances.

si_rec_wavecal for wavelength calibration.

si_rec_psf for PSF standard data reduction.

si_rec_stdstar for STD standard data reduction.

si_rec_objnod for science standard data reduction.

4.2 An introduction to Gasgano and EsoRex

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

The process of data classification and association is known as data organisation. The *DO Category* is the label assigned to a data type as a result of data classification.

An instrument pipeline consists of a set of data processing modules that can be called from different host applications, either from the command line with *EsoRex*, from the automatic data management tools available at Paranal, or from the graphical *Gasgano* tool.

Gasgano is a data management tool that simplifies the data organisation process, offering automatic data classification and making the data association easier (*even if automatic association of frames is not yet provided*). *Gasgano* determines the classification of a file by applying an instrument specific rule, while users must provide this information to the recipes when they are executed manually using *EsoRex* from the command line. In addition, *Gasgano* allows the user to execute directly the pipeline recipes on a set of selected files.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	13 of 40

4.2.1 Using Gasgano

To get familiar with the IIINSTRUMENT pipeline recipes and their usage, it is advisable to begin with *Gasgano*, because it provides a complete graphic interface for data browsing, classification and association, and offers several other utilities such as easy access to recipes documentation and preferred data display tools.

Gasgano can be started from the system prompt in the following way:

```
gasgano &
```

The *Gasgano* main window will appear. On Figure 4.1 (next page), a view on a set of IIINSTRUMENT IFU data is shown as an example. *Gasgano* can be pointed to the directories where the data to be handled are located using the navigation panels accessible via the *Add/Remove Files* entry of the *File* menu (shown on the upper left of the figure).

The data are hierarchically organised as preferred by the user. After each file name are shown the classification, the instrument setup id (which indicates the band), the instrument pre-optic (which indicates the camera setting), the template exposure number and the number of exposures in the template, and the value of the DPR.TYPE.

More information about a single frame can be obtained by clicking on its name: the corresponding FITS file header will be displayed on the bottom panel, where specific keywords can be opportunely filtered and searched. Images and tables may be easily displayed using the viewers specified in the appropriate *Preferences* fields.

Frames can be selected from the main window for being processed by the appropriate recipe: on Figure 4.2, the standard star frame and a sky frame, already produced master bad pixel map and master flat field frames, together with distortion and slitlet distance tables, and the necessary static calibration tables, are all selected and sent to the *si_rec_stdstar* recipe. This will open a *Gasgano* recipe execution window (see Figure 4.3), having all the specified files listed in its *Input Frames* panel.

Help about the recipe may be obtained from the *Help* menu. Before launching the recipe, its configuration may be opportunely modified on the *Parameters* panel (on top). The window contents might be saved for later use by selecting the *Save Current Settings* entry from the *File* menu, as shown in figure.

At this point the recipe can be launched by pressing the *Execute* button. Messages from the running recipe will appear on the *Log Messages* panel at bottom, and in case of successful completion the products will be listed on the *Output Frames* panel, where they can be easily viewed and located back on the *Gasgano* main window.

Please refer to the *Gasgano User's Manual* [7] for a more complete description of the *Gasgano* interface.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	14 of 40

Figure 4.1: *The Gasgano main window.*

Figure 4.2: *Selecting files to be processed by a IIINSTRUMENT pipeline recipe.*

Figure 4.3: *The Gasgano recipe execution window.*

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	15 of 40

4.2.2 Using EsoRex

EsoRex is a command line utility for running pipeline recipes. It may be embedded by users into data reduction scripts for the automation of processing tasks. On the other side, *EsoRex* doesn't offer all the facilities available with *Gasgano*, and the user must classify and associate the data using the information contained in the FITS header keywords (see Section 6.1, page 19). The user should also take care of defining the input set-of-frames and the appropriate configuration parameters for each recipe run:

The set-of-frames: Each pipeline recipe is run on a set of input FITS data files. When using *EsoRex* the filenames must be listed together with their DO category in an ASCII file, the *set-of-frames* (SOF), that is required when launching a recipe.¹

Here is an example of SOF, valid for the *si_rec_wavecal* recipe²:

```

/file_path/SINFO.2004-08-14T10:20:56.497.fits  WAVE_LAMP
/file_path/SINFO.2004-08-14T10:22:44.285.fits  WAVE_LAMP
/file_path/xenon.tfits                        REF_LINE_ARC
/file_path/MASTER_BP_MAP_H_250.fits          MASTER_BP_MAP
/file_path/MASTER_LAMP_FLAT_H_250.fits       MASTER_FLAT_LAMP
/file_path/DISTORTION_H.fits                  DISTORTION
/file_path/drs_setup_wave.tfits              DRS_SETUP_WAVE
/file_path/SLIT_POS_H_250.tfits              SLIT_POS

```

It contains for each input frame the full path file name and its DO category. The pipeline recipe will access the listed files when required by the reduction algorithm.

Note that the IIINSTRUMENT pipeline recipes do not verify in any way the correctness of the *DO Category* specified by the user in the SOF. The reason of this lack of control is that the IIINSTRUMENT recipes are just the DRS component of the complete pipeline running on Paranal, where the task of data classification and association is carried out by separate applications. Moreover, using *Gasgano* as an interface to the pipeline recipes will always ensure a correct classification of all the data frames, assigning the appropriate DO category to each one of them (see Section 4.2.1, page 13).

A recipe handling an incorrect SOF may stop or display unclear error messages at best. In the worst cases, the recipe would apparently run without any problem, producing results that may look reasonable, but are actually flawed.

EsoRex syntax: The basic syntax to use ESOREX is the following:

esorex [esorex_options] recipe_name [recipe_options] set_of_frames

To get more information on how to customise ESOREX (see also [7]) run the command:

esorex - -help

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

esorex - -create-config

¹The set-of-frames corresponds to the *Input Frames* panel of the *Gasgano* recipe execution window (see Figure 4.3, page 14).

²We list the file SLIT_POS_H_250.fits as an input file, as, for robustness, we suggest the user to set the parameter **slit-pos_bootstrap_switch** to FALSE. A different setting would allow to reduce the data without including the SLIT_POS table

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	16 of 40

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

esorex --recipes

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

esorex --params recipe_name

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

esorex --help recipe_name

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

esorex --man-page recipe_name

Recipe configuration: Each pipeline recipe may be assigned an *EsoRex* configuration file, containing the default values of the parameters related to that recipe.³ The configuration files are normally generated in the directory `$HOME/.esorex`, and have the same name as the recipe to which they are related, with the filename extension `.rc`. For instance, the recipe `si_rec_wavecal` has its *EsoRex* generated configuration file named `si_rec_wavecal.rc`, and is generated with the command:

esorex --create-config si_rec_wavecal

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

```
# --stack-warpfix_kernel
# Warpfix kernel: (tanh | sinc | sinc2 | lanczos | hamming | hann)
sinfoni.stacked.warpfix_kernel=tanh
```

In this example, the parameter `sinfoni.stacked.warpfix_kernel` is set to the value `tanh`. In the configuration file generated by *EsoRex*, one or more comment lines are added containing information about the possible values of the parameter, and an alias that could be used as a command line option.

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

The command

esorex --create-config recipe_name

generates a default configuration file **recipe_name.rc** in the directory **\$HOME/.esorex**⁴.

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

esorex --recipe-config=my_alternative_recipe_config

Recipe parameters are provided in section 9 and their role is described in Section 10.

More than one configuration file may be maintained for the same recipe but, in order to be used, a configuration file not located under `$HOME/.esorex`, or having a name different from the recipe name, should be explicitly specified when launching a recipe.

³The *EsoRex* recipe configuration file corresponds to the *Parameters* panel of the *Gasgano* recipe execution window (see Figure 4.3, page 14).

⁴If a number of recipe parameters are specified on the command line, the given values will be used in the created configuration file.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	17 of 40

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

esorex si_rec_wavecal si_rec_wavecal.sof

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

esorex si_rec_wavecal --wcal-pixel_tol=3.0 si_rec_wavecal.sof

For more information on *EsoRex*, see <http://www.eso.org/cpl/esorex.html>.

4.3 Example of data reduction using EsoRex

A simple, typical data reduction procedure is described here.⁵

We suggest the user to organize his data per type, observed band and camera setting. Dark frames may be grouped per detector DIT, frames to compute distortion and frames to compute detector non linearities may be organized per observed band. The detector DIT is given by the value of the FITS keyword DET DIT⁶. The observed band is indicated by the value of the FITS keyword INS SETUP ID. The camera setting is indicated by the value of INS OPT11 NAME. In the examples below we suppose the user has data acquired in band K and with the 100 mas pre-optic setting, and DIT=600. In the following examples */path_raw/* indicates the full path to the source tree directory containing raw data.

Dark Frames: those frames are characterized by DPR.TYPE='DARK',

```
/path_raw/DARK/600/SINFO.2004-08-23T09:36:12.316.fits DARK
/path_raw/DARK/600/SINFO.2004-08-23T09:51:59.824.fits DARK
/path_raw/DARK/600/SINFO.2004-08-23T10:07:40.760.fits DARK
```

Detector linearity flat field frames: those frames are characterized by DPR.TYPE='LINEARITY,LAMP'

```
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:09:50.882.fits LINEARITY_LAMP
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:10:07.455.fits LINEARITY_LAMP
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:10:23.047.fits LINEARITY_LAMP
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:10:38.240.fits LINEARITY_LAMP
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:10:56.403.fits LINEARITY_LAMP
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:11:31.128.fits LINEARITY_LAMP
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:11:59.183.fits LINEARITY_LAMP
/path_raw/LINEARITY/K/SINFO.2005-02-26T20:12:27.247.fits LINEARITY_LAMP
```

⁵The procedure using *Gasgano* is conceptually identical.

⁶We omit here the prefix HIERARCH ESO

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	18 of 40

5 Known Problems

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	19 of 40

6 Instrument Data Description

IIINSTRUMENT data can be separated into *raw* frames and *product* frames. Raw frames are the unprocessed output of the IIINSTRUMENT instrument observations, while product frames are either the result of the II-INSTRUMENT pipeline processing (as reduced frames, master calibration frames, etc.), or are outsourced (as standard stars catalogs, lists of grism characteristics, etc.).

Any raw or product frame can be classified on the basis of a set of keywords read from its header. Data classification is typically carried out by the DO or by *Gasgano* [7], that apply the same set of classification rules. The association of a raw frame with calibration data (*e.g.*, of a science frame with a master bias frame) can be obtained by matching the values of a different set of header keywords.

Each kind of *raw* frame is typically associated to a single IIINSTRUMENT pipeline recipe, *i.e.*, the recipe assigned to the reduction of that specific frame type. In the pipeline environment this recipe would be launched automatically. In some cases two recipes are assigned, one meant for the reduction of a single frame of that type, and the other for the reduction of a *stack* of frames of the same type, as happens in the case of jittered science observations.

A *product* frame may be input to more than one IIINSTRUMENT pipeline recipe, but it may be created by just one pipeline recipe (with the same exceptions mentioned above). In the automatic pipeline environment a product data frame alone wouldn't trigger the launch of any recipe.

In the following all raw and product IIINSTRUMENT data frames are listed, together with the keywords used for their classification and correct association. The indicated *DO category* is a label assigned to any data type after it has been classified, which is then used to identify the frames listed in the *Set of Frames* (see Section 4.2.2, page 15).

Raw frames can be distinguished in *general* frames, *direct imaging* frames, *MOS* frames and *IFU* frames. Their intended use is implicitly defined by the assigned recipe.

6.1 General frames

These are data that are in principle independent of the instrument mode (direct imaging, MOS, or IFU), as is the case for bias and dark exposures. The keyword `ESO INS MODE` is set accordingly to 'IMG' for direct imaging frames, and to 'MOS' for any calibration associated to spectroscopy (either MOS or IFU), to indicate the intended use for the data.

- **Bias:**

DO category: BIAS

Processed by: `vmbias`

Classification keywords:

DPR CATG = CALIB

DPR TYPE = BIAS

DPR TECH = IMAGE

Association keywords:

INS MODE

OCS CON QUAD

DET CHIP1 ID

DET WIN1 NY

DET WIN1 BINX

DET WIN1 BINY

Note:

Instrument mode

Quadrant used

Chip identification

No of pixels in y

Binning along X

Binning along Y

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	20 of 40

DET READ MODE	Readout method
DET READ SPEED	Readout speed
DET READ CLOCK	Readout clock pattern

- **Dark current:**

DO category: DARK
Processed by: vmdark

Classification keywords:

DPR CATG = CALIB
DPR TYPE = DARK
DPR TECH = IMAGE

Association keywords:

INS MODE
OCS CON QUAD
DET CHIP1 ID
DET WIN1 NY
DET WIN1 BINX
DET WIN1 BINY
DET READ MODE
DET READ SPEED
DET READ CLOCK

Note:

Instrument mode
Quadrant used
Chip identification
No of pixels in y
Binning along X
Binning along Y
Readout method
Readout speed
Readout clock pattern

- **Screen flat field for gain determination and bad pixels detection:**

DO category: DETECTOR_PROPERTIES
Processed by: vmdet

Classification keywords:

DPR CATG = CALIB
DPR TYPE = FLAT, LAMP
DPR TECH = IMAGE or MOS
TPL ID = IIINSTRUMENT_img_tec_DetLin
or IIINSTRUMENT_mos_tec_DetLin

Association keywords:

INS MODE
OCS CON QUAD
DET CHIP1 ID
DET WIN1 NY
DET WIN1 BINX
DET WIN1 BINY
DET READ MODE
DET READ SPEED
DET READ CLOCK

Note:

Instrument mode
Quadrant used
Chip identification
No of pixels in y
Binning along X
Binning along Y
Readout method
Readout speed
Readout clock pattern

6.2 Direct imaging frames

The direct imaging mode is used to record signal without using any grism.

- **Exposure of calibration mask:**

DO category: MASK_TO_CCD
Processed by: vmmasktocc

Classification keywords:

DPR CATG = CALIB
DPR TYPE = OTHER, LAMP
DPR TECH = IMAGE

Association keywords:

INS MODE
OCS CON QUAD
INS ADF ID

Note:

Instrument mode
Quadrant used
ADF file ID

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	21 of 40

TPL ID = IIINSTRUMENT_img_tec_MaskTSC FILTER[1-4] ID

DET CHIP1 ID
DET WIN1 NY
DET WIN1 BINX
DET WIN1 BINY
DET READ MODE
DET READ SPEED
DET READ CLOCK

Filter ID on each beam
Chip identification
No of pixels in y
Binning along X
Binning along Y
Readout method
Readout speed
Readout clock pattern

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	22 of 40

7 Static Calibration Data

In the following all raw and product IIINSTRUMENT data frames are listed, together with the keywords used for their classification and correct association. The indicated *DO category* is a label assigned to any data type after it has been classified, which is then used to identify the frames listed in the *Set of Frames* (see Section 4.2.2, page 15).

7.1 Photometric table

The optional determination of the frame magnitude zeropoint from the table of detected standard stars (see ahead) would require to specify in the input SOF a photometric table. The photometric table simply holds the necessary parameters for the magnitude zeropoint computation, as listed in Table 7.1. The standard photometric tables in the calibration directories are named `ipc_f.q.tfits` (where *f* is the filter name and *q* the quadrant number).

Keyword	Example	Explanation
PRO MAG ZERO	28.15	Expected magnitude zeropoint
PRO EXTINGT	0.25	Atmospheric extinction coefficient
PRO COLTERM	0.01	Correction for star colour
PRO COLOUR	'B-V'	Colour system used
PRO MAGZERO RMS	0.05	Error on expected zeropoint
PRO EXTINGT RMS	0.00	Error on extinction coefficient
PRO COLTERM RMS	0.00	Error on colour term

Table 7.1: *Photometric table entries.*

7.2 Photometric catalog

The photometric catalog currently used can be found in the directory `$PIPE_HOME/vimos/ima/cal`, in the file `phstd_stetson.tfits` (see Table 7.2). This table includes the photometric stars from the Stetson's fields (see <http://cadwww.dao.nrc.ca/standards>); Landolt's stars (Landolt 1992, AJ 104, 340) that can be found in the Stetson's fields are also included, to permit the determination of zeropoints also in the U band.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	23 of 40

Column name	Explanation
ID	Star identification string
RA	RA of star
DEC	Dec of star
MAG_U	U magnitude of star
MAG_B	B magnitude of star
MAG_V	V magnitude of star
MAG_R	R magnitude of star
MAG_I	I magnitude of star

Table 7.2: *Photometric catalog entries.*

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	24 of 40

8 Data Reduction

In this section, after an overview of the main problems the data reduction needs to solve, we list the required data and the recipes which allow to solve them, giving the data reduction sequence necessary to reduce calibration and science data.

8.1 Data reduction overview

The principle of integral field spectroscopy is described by figure 8.1. A two-dimensional image of the sky is separated by an image slicer into several components. Those are then aligned on a slit and dispersed to separate its spectral information and then imaged on a detector. The main IIINSTRUMENT data reduction problems to solve are the following.

- Correct for the detector signature: bad pixels, detector contribution to the measured signal, flat fielding (correct pixel to pixel gain variations and relative slitlet throughput differences), correct geometric distortions.
- Perform the wavelength calibration.
- Reconstruct the image FOV from the 32 image slices in a format which contains both the spatial and the spectral information.
- Devise proper calibrations and observations to be able to properly correct the emission from the sky, from the instrument and from the telescope which are very strong in the NIR. This requires to take sky frames together with the object frames in the night observations, daily calibrations with the flat lamp switched on and off, and possibly dark frames.

Figure 8.1: Integral Field Spectroscopy data reduction principle

In the following description we also indicate in parenthesis for each frame the corresponding PRO.CATG. To locate the detector bad pixels one uses a bad pixel map. A master bad pixel map resulting from the combination of a set of (different) bad pixel maps is generated by the master flat recipe. First of all, as the detector is known to have construction defects, these will be indicated by a reference bad pixel map (REF_BP_MAP). Hot pixels will be determined on dark frames (BP_MAP_HP). Non-linear response pixels are instead indicated by a bad pixel map (BP_MAP_NL) obtained by evaluating the pixel response of a set of flat exposures of increasing intensity. Other bad pixels (BP_MAP_NO) are determined on a set of flat fields (on the master flat field). A master flat field (MASTER_FLAT_LAMP) generated from a set of raw flat fields, is used to correct the different detector pixel sensitivities. It is known that the image sliced and projected on the detector is affected by distortions. The si_rec_distortion recipe computes the distortions (DISTORTION) and the slitlet distances (SLITLETS_DISTANCE). It uses a set of raw frames where only the first column of each slitlet is illuminated through fibres. In addition flats and arcs are taken within the north south template as required to reduce the data. A set of “on” and “off” arc lamp frames, a reference line table, a master flat field, the optical distortions map

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	25 of 40

and a good guess of the slitlet positions are input of the wavecal recipe. This recipe determines a wavelength map and obtains a better computation of the left edge position of each slitlet.

Science (or PSF or telluric standard) frames are corrected for sky background, flat field, distortions, and calibrated in wavelength. The sorted slitlets are then stacked in a cube, taking into account the relative distances of the position of the edge of each slitlet to the one of the first slitlet, with a final image realignment to get sub pixel accuracy. The final product is thus a 3D data cube where the full spatial information is stored along the X and Y directions, and the wavelength information is stored along the Z direction. Each plane of the cube is a monochromatic reconstruction of the IIINSTRUMENT FOV.

The north south test template traces each of the 32 slitlets by only one fibre exposure; therefore non linearities of the image scale within the 64 pixel of a single slitlets are currently not corrected and could cause minor slice to slit ripples in the reconstructed cube.

8.2 Required input data

To be able to reduce science data one needs to use raw, product data and pipeline recipes in a given sequence which provides all the input necessary to each pipeline recipe. We call this sequence a data reduction cascade. The IIINSTRUMENT data reduction cascade involves the following input data:

- Reference files:
 - A reference bad pixel map, indicating known detector defects.
 - A list of arc lamp emission lines containing vacuum wavelengths and predicted intensities for wavelength calibration.
 - The DRS_SETUP_WAVE table as input of si_rec_distortion and si_rec_wavecal to specify parameter peculiar of the wavelength calibration algorithm.
 - A constant position assumed for the first column.
- Raw frames:
 - Linearity flat frames, to determine a map of non linear bad pixels.
 - Darks, to determine master darks.
 - Flat fields, to determine master flats.
 - Fibre frames, to trace the first column of each slitlet and, using also on/off lamp flats and arc lamp frames, to compute the optical distortions and slitlet distances.
 - Arc lamps, to perform the wavelength calibration.
 - Sky frames, to evaluate and subtract the strong and time-variable NIR sky emission.
 - Telluric STD star frames, to correct telluric absorption features.
 - PSF standards, to evaluate the strehl.
 - Science frames, to finally do science.
- Calibration data products.
 - Bad pixel maps, to correct for the detector defects.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	26 of 40

- Distortion coefficients, to correct for the optical distortions.
- Master flats, to correct for different detector pixel efficiencies.
- Master darks, to correct for the instrument bias if no off or sky frames are available.
- Slitlet distances, to be able to properly reconstruct a cube.
- Slitlets' left edge positions, to be able to properly reconstruct a cube.
- Wavelength maps, to obtain a cube calibrated in wavelength.

Calibration data products can be generated from raw data using the pipeline recipes. Alternatively the user may use calibration products obtained from the ESO archive or from the ESO Data Flow Operation department. Master bad pixel maps, the bad pixel maps coming from the standard flats, the master flats, the slitlets position table, the wavelength map depend from the observed band and instrument's pre-optic. Bad pixel maps coming from the non linearity test, distortion tables, slitlet distances, reference line tables depend only on the observed band. The first column table, the reference bad pixel map, and master darks depend neither from the band nor the pre-optic. Science data requiring master dark need to have matching values of the FITS keyword HIERARCH ESO DET DIT.

8.3 Reduction cascade

The IIINSTRUMENT data reduction follows the following sequence. A short description of the available recipes is given in section 4.1. In parenthesis we provide the value of the DO category corresponding to each frame.

- Run **si_rec_detlin** on a set of flats with increasing intensity (LINEARITY_LAMP) to determine the non linearity pixels bad pixel map (BP_MAP_NL).
- Run **si_rec_mdark** on a set of raw darks (DARK) to determine the master dark (MASTER_DARK) and the hot pixels bad pixel map (BP_MAP_HP). This map depends on DIT.
- Run **si_rec_mflat** on a set of standard flat fields (FLAT_LAMP), the BP_MAP_NL and the REF_BP_MAP to determine the master bad pixel map (MASTER_BP_MAP) and the master lamp flat (MASTER_FLAT_LAMP).
- Run **si_rec_distortion** on a set of fibre flats (FIBRE_NS), on/off arc lamps (WAVE_NS) and on/off lamp flats (FLAT_NS), using a reference line table (REF_LINE_ARC) to determine the optical distortions (DISTORTION) and the slitlet distances (SLITLET_DISTANCES). To set a few data reduction parameters which depends from the observed band and used instrument pre-optics the user has also to provide in input a DRS_SETUP_WAVE table frame.
- Run **si_rec_wavecal** on a set of arc lamp frames (FLAT_WAVE), a MASTER_BP_MAP, a MASTER_FLAT_LAMP, a DISTORTION, and a REF_LINE_ARC to determine the wavelength map (WAVE_MAP) and the slitlet edge position table (SLIT_POS). To set a few data reduction parameters which depends from the observed band and used instrument pre-optics the user has also to provide in input a DRS_SETUP_WAVE table. If the parameter **wcal-slitpos_bootstrap** has value set to FALSE, as we suggest for robustness, the user need to provide in input also an appropriate SLIT_POS table, for example the one we provide as part of ddata reduction kit.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	27 of 40

- Run **si_rec_psf** on PSF standards and a MASTER_BP_MAP, a MASTER_FLAT_LAMP, a DISTORTION, a SLITLET_DISTANCES, a SLIT_POS and a WAVE_MAP, to reduce the PSF standard and get information on the instrument's strehl.
- Run **si_rec_stdstar** on a reference telluric standard (STD) and a MASTER_BP_MAP, a MASTER_FLAT_LAMP, a DISTORTION, a SLITLET_DISTANCES, and a SLIT_POS and WAVE_MAP, to reduce the telluric standards and get information on the instrument's response.
- Run **si_rec_objnod** on your scientific data (OBJECT_NODDING) and a MASTER_BP_MAP, a MASTER_FLAT_LAMP, a DISTORTION, a SLITLET_DISTANCES, a SLIT_POS and a WAVE_MAP to reduce science data.

The main data products involved in the data reduction cascade are indicated in the IIINSTRUMENT association map shown in Figure [8.2](#). It summarise dependencies between raw data, calibration products and recipes involved in the correction of the instrument signature and reduction of science data. Examples of set of frames input for each recipe are provided in section [9](#).

ESO	IINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	28 of 40

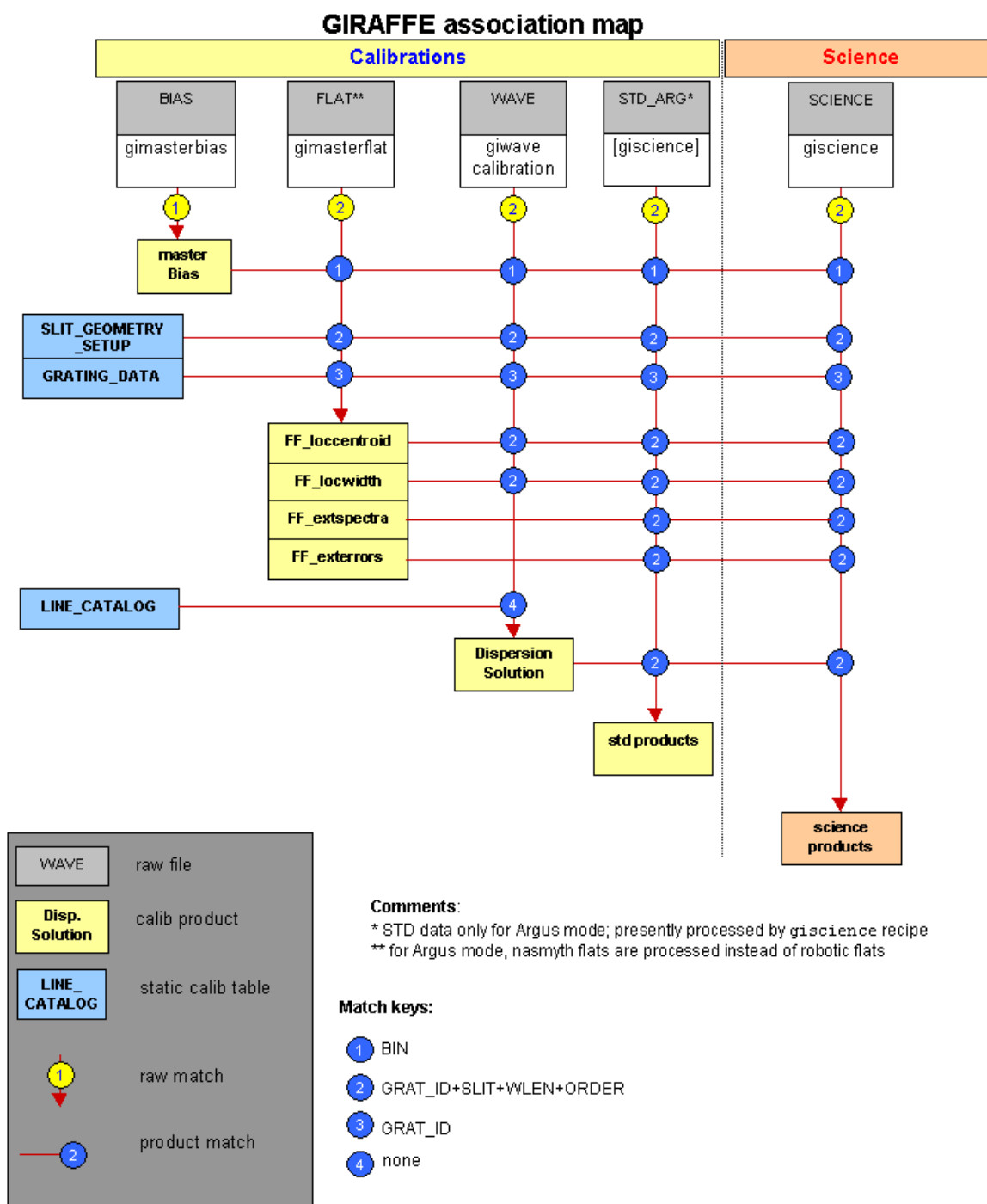


Figure 8.2: IINSTRUMENT Association Map.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	29 of 40

9 Pipeline Recipes Interfaces

In this section we provide for each recipe examples of the required input data and their tags. In the following we assume that `/path_file_raw/filename_raw.fits` and `/path_file_cdb/filename_cdb.tfits` are existing FITS files (e.g. `/data1/sinfony/com2/SINFO.2004-08-16T02:54:04.353.fits` and `/cal/sinfo/ifu/cal/DISTORTION_K.tfits`).

We also provide a list of the pipeline products for each recipe, indicating their default recipe name, the value of the FITS keyword `HIERARCH ESO PRO CATG` (in short `PRO.CATG`) and a short description. The relevant keywords are `PRO.CATG`, used to classify each frame, and to associate to each raw frame the proper calibration frame:

Association keyword	Information
<code>HIERARCH ESO INS SETUP ID</code>	band
<code>HIERARCH ESO INS OPTI1 NAME</code>	Pixel scale
<code>HIERARCH ESO DET DIT</code>	Integration time

For each recipe we also list in a table the input parameters (as they appear in the recipe configuration file), the corresponding aliases (the corresponding names to be eventually set on command line) and their default values. Also quality control parameters are listed. Those are stored in relevant pipeline products. More information on instrument quality control can be found on <http://www.eso.org/qc>

In addition to the products mentioned below, all recipes produce a PAF (VLT parameter file) which is an intermediate pipeline data file containing quality control parameter values.

We distinguish between recipes involved in the data reduction cascade (having prefix `si_rec`) and user utilities (with prefix `si_utl`).

9.1 si_rec_detlin

The recipe `si_rec_detlin` computes the detector responsivity as a function of the pixel intensity and determines when it becomes non linear.

9.1.1 Input

```

/path_file_raw/SINFO.2004-08-16T02:54:04.353.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:53:37.089.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:52:23.028.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:51:59.774.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:50:38.991.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:50:11.797.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:49:04.887.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:48:36.792.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:47:28.191.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:47:07.438.fits LINEARITY_LAMP
/path_file_raw/SINFO.2004-08-16T02:45:59.488.fits LINEARITY_LAMP

```

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	30 of 40

/path_file_raw/SINFO.2004-08-16T02:45:42.255.fits LINEARITY_LAMP

9.1.2 Output

default recipe file name	PRO.CATG	short description
lin_det_info.tfits	LIN_DET_INFO	Table with coefficients of non linear fit to median of each flat image image
gain_info.tfits	GAIN_INFO	Table with detector's gain values
out_bp_lin_coeffsCube.fits	BP_COEFF	image with coefficients of non linear fit to pixel's intensity used to evaluate non linearity
out_bp_lin.fits	BP_MAP_NL	Non linear bad pixel map

9.1.3 Quality control

The pipeline computes the number of non linear pixels per grating, the detector gain.

Non linear bad pixels The pipeline computes the number of non linear bad pixels. Those are given by QC.BP-MAP.NBADPIX and are obtained with the method QC.BP-MAP.METHOD.

Detector gain The detector gain is given by the value of QC.GAIN.

9.1.4 Parameters

parameter	alias	default
sinfoni.bp_lin.order	bp_lin-order	2
sinfoni.bp_lin.thresh_sigma_factor	bp_lin-thresh_sigma_fct	10.0
sinfoni.bp_lin.nlin_threshold	bp_lin-nlin_threshold	0.5
sinfoni.bp_lin.low_rejection	bp_lin-lo_rej	10.0
sinfoni.bp_lin.high_rejection	bp_lin-hi_rej	10.0

9.2 si_rec_mdark

The recipe si_rec_mdark generates a master dark from a set of raw darks by stacking frames with rejection of outliers. It also generates a bad pixel map flagging the hot-current pixels.

9.2.1 Input

/path_file_raw/SINFO.2004-08-16T01:24:53.070.fits DARK
/path_file_raw/SINFO.2004-08-16T01:09:22.905.fits DARK
/path_file_raw/SINFO.2004-08-16T00:53:51.890.fits DARK

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	31 of 40

/path_file_raw/SINFO.2004-08-16T00:38:14.994.fits DARK

9.2.2 Output

default recipe file name	PRO.CATG	short description
out_bp_noise.fits	BP_MAP_HP	bad pixel map, method="Noise"
out_dark.fits	MASTER_DARK	master dark

9.2.3 Quality control

Dark frames are processed to monitor the RON, Read Out Noise per DIT, the FPN, Fixed Patter Noise per DIT, the detector counts per DIT, the number of hot pixels per DIT.

RON The RON is computed on the whole detector chip and given as value of the QC.RON parameter. For quality control those values are monitored as a function of time and DIT. Two consecutive frames are subtracted from each other and the median standard deviation of a limited number of samples is taken and normalised to DET.NDIT=1. The RON is computed in two regions and is given by the values of QC.RON1 and QC.RON2.

Dark median counts The median and standard deviation of the counts in the master dark frame are monitored by DFO. Its value and standard deviation are given by the values of QC.DARKMED.AVE and QC.DARKMED.STDEV

Fixed Pattern Noise A histogram of the master dark is produced, and a fit is applied; the standard deviation (sigma) of the Gaussian is the FPN. This value, logged by parameter QC.DARKFPN, is monitored for different DITs. The FPN should scale linearly with the number of counts. For this reason the ratio FPN/counts is monitored for different DITs.

Number of hot pixels The number of pixels having an intensity greater than a threshold is monitored in the parameter QC.BP-MAP.NBADPIX

9.2.4 Parameters

parameter	alias	default
sinfoni.bp_noise.thresh_sigma_factor	bp_noise-thresh_sigma_fct	10.0
sinfoni.bp_noise.low_rejection	bp_noise-lo_rej	10.0
sinfoni.bp_noise.high_rejection	bp_noise-hi_rej	10.0
sinfoni.dark.low_rejection	dark-lo_rej	0.1
sinfoni.dark.high_rejection	dark-hi_rej	0.1
sinfoni.dark.qc_ron_xmin	dark-qc_ron_xmin	1
sinfoni.dark.qc_ron_xmax	dark-qc_ron_xmax	2048

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	32 of 40

sinfoni.dark.qc_ron_ymin	dark-qc_ron_ymin	1
sinfoni.dark.qc_ron_ymax	dark-qc_ron_ymax	2048
sinfoni.dark.qc_ron_hsize	dark-qc_ron_hsize	4
sinfoni.dark.qc_ron_nsamp	dark-qc_ron_nsamp	100
sinfoni.dark.qc_fpn_xmin	dark-qc_fpn_xmin	1
sinfoni.dark.qc_fpn_xmax	dark-qc_fpn_xmax	2047
sinfoni.dark.qc_fpn_ymin	dark-qc_fpn_ymin	1
sinfoni.dark.qc_fpn_ymax	dark-qc_fpn_ymax	2047
sinfoni.dark.qc_fpn_hsize	dark-qc_ron_hsize	2
sinfoni.dark.qc_fpn_nsamp	dark-qc_ron_nsamp	1000

ESO	IINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	33 of 40

10 Algorithms

In this section the data reduction procedures applied by the 23 pipeline recipes currently in use (see Section 4.1) are described in some detail. Common algorithms, as cosmic rays removal or bad pixel cleaning, are described separately.

10.1 General Algorithms

10.1.1 Bad pixel cleaning

Bad pixel cleaning consists of replacing any bad pixel value with an estimate based on a set of surrounding *good* pixel values. This operation is generally applied to science product frames, having little or no sense when applied to master calibration products. Nevertheless all the IINSTRUMENT pipeline recipes allow bad pixel cleaning on any product frame, for debug reasons or for any other purpose that may be appropriate.

The routine currently used by the IINSTRUMENT pipeline recipes performs a bad pixel correction based on the content of a given bad pixel table (CCD_TABLE). If the number of bad pixels is more than 15% of the total number of CCD pixels, the correction is not applied.

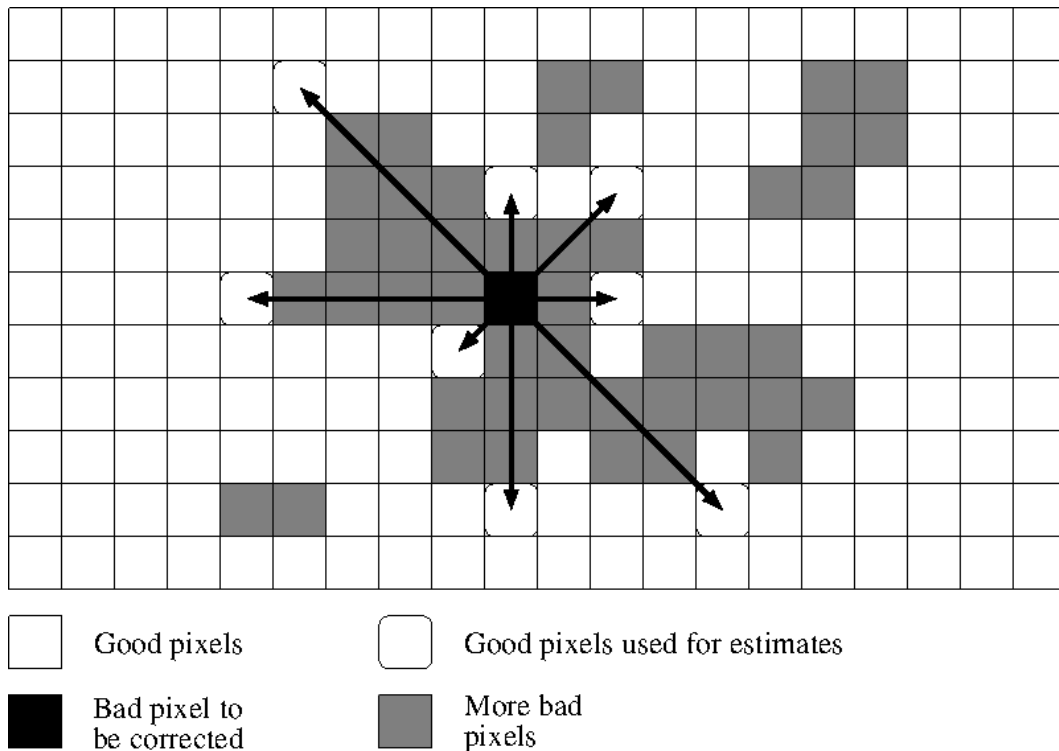


Figure 10.1: *Good pixels to be used in the estimate of a given bad pixel are searched along the indicated directions..*

Any bad pixel is given a new value, computed as follow: the closest good pixels along the vertical, the horizontal,

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	34 of 40

and the two diagonal directions are found (see Figure 10.1). This search is done within a distance of 100 pixels. If no good pixel is found within this range, then the bad pixel is not corrected. All the good pixels found within range will be used to compute the bad pixel value.

For each of the four fundamental directions, an estimate of the considered bad pixel can generally be obtained. If two good pixel values are available for a given direction, the estimate is their linear interpolation at the bad pixel position. If just one good pixel value is available for a given direction, then the value itself will be the estimate of the bad pixel value. No estimate can be obtained from directions where no good pixel was found.

If the available number of estimates is greater than 1, the bad pixel value is taken as the median of the estimates (defining the median of an even number of values as the mean of the two central values), otherwise it is simply set to the single estimate available.

10.1.2 Cosmic rays removal

The core of a cosmic rays removal procedure is to determine what is and what is *not* a cosmic ray. The algorithm used for this purpose by the IIINSTRUMENT pipeline recipes is the same applied by the MIDAS command `FILTER/COSMIC`, with some extensions.

Initially all pixels having an abnormal excess with respect to the local noise level are flagged as possibly belonging to a cosmic ray event (that typically would involve a group of contiguous pixels). A candidate is selected at any pixel (x, y) having a value $F(x, y)$ exceeding a given threshold. This threshold, expressed in units of noise sigma, is specified by the recipe parameter *CosmicsThreshold*. A value 4.0 gives typically good results. The theoretical noise $N(x, y)$ of the image at any given pixel position (x, y) is estimated in ADU as

$$N(x, y) = \sqrt{r^2 + \frac{M(x, y)}{g}}$$

where $M(x, y)$ is the median value of the 8 pixels surrounding the (x, y) position and r is the read-out-noise, both in ADU, and g is the gain factor in e^-/ADU . Then a pixel (x, y) is taken as a cosmic ray candidate if

$$F(x, y) > k \cdot N(x, y)$$

with k the number of noise sigmas used in thresholding.

After this step is completed, all the groups of contiguous cosmic rays candidates are identified. For each group, the position of its maximum pixel value is determined, and the mean \overline{F}_8 of its 8 surrounding pixels is computed. A given group will be taken as a cosmic ray event if it fulfils the condition

$$F_{max} - S > R \cdot (\overline{F}_8 - S)$$

where F_{max} is the maximum pixel value within the considered group, S the fundamental background level (corresponding to the sky level in imaging science exposures), and R is a shape parameter for discriminating between objects and cosmic rays. The ratio R is specified by the recipe parameter *CosmicsRatio*. A value of 2.0 gives typically good results.

Once all the pixels affected by cosmic ray events has been located and listed in a cosmic ray events table, their values are interpolated using the procedure described in Section 10.1.1. If a bad pixel table is also given to a recipe, then the bad pixels are avoided in the interpolation procedure.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	35 of 40

10.2 Recipes Algorithms

10.2.1 gi_rec_bias

Removing the bias from any raw frame is a relatively simple process, but not simple enough to avoid a description on its own.

A master bias frame (`MASTER_BIAS`) is used to remove the bias level (and, if present, the fixed-pattern-noise related to the bias) from a raw frame. Typically a master calibration is produced with its overscan regions trimmed, and if this is the case with the master bias used then its missing overscan regions are extrapolated by repeating the signal contained in its border regions with equal size.

The master bias is subtracted from the raw data frame, whose overscan regions are then trimmed away. Optionally (when the *BiasMethod* recipe parameter is set to “*Zmaster*”) the residual signal in the overscan regions is averaged along the *X* CCD coordinate, and the obtained mean *Y* values are modeled with a second order polynomial fitting. This model is then subtracted from the rest of the image.

10.2.2 gi_rec_dark

Subtracting the dark current component from any raw frame consists of multiplying an input master dark frame by the exposure time (in seconds) of the frame to be corrected, and then subtract such rescaled dark frame from it. The dark level is quite low for IIINSTRUMENT CCDs (about $5 \text{ e}^- \cdot \text{h}^{-1} \cdot \text{pixel}^{-1}$), so this operation would be in most cases superfluous.

10.2.3 gi_rec_flat

The flat field correction merely consists of dividing the frame to be corrected by a given master sky flat field frame produced by the recipe *vmimflatsky* for direct imaging observation (see Section 10.2.3, page 35), or produced by the recipe *vmspflat* for MOS observations (see Section 10.2.3, page 35).

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	36 of 40

A Installation

This chapter gives generic instructions on how to obtain, build and install the IIINSTRUMENT pipeline. Even if this chapter is kept as up-to-date as much as possible, it may not be fully applicable to a particular release. This might especially happen for patch releases. One is therefore advised to read the installation instructions delivered with the IIINSTRUMENT pipeline distribution kit. These release-specific instructions can be found in the file `README` located in the top-level directory of the unpacked IIINSTRUMENT pipeline source tree. The supported platforms are listed in Section A.1. It is recommended reading through Section A.2.2 before starting the installation.

A bundled version of the IIINSTRUMENT pipeline with all the required tools and an installer script is available from <http://www.eso.org/pipelines/>, for users who are not familiar with the installation of software packages.

A.1 Supported platforms

The utilisation of the GNU build tools should allow to build and install the IIINSTRUMENT pipeline on a variety of UNIX platforms, but it has only been verified on the VLT target platforms:

- Linux (glibc 2.1 or later),
- Sun Solaris 2.8 or later,

using the GNU C compiler (version 3.2 or newer).

A.2 Building the IIINSTRUMENT pipeline

This section shows how to obtain, build and install the IIINSTRUMENT pipeline from the official source distribution.

A.2.1 Requirements

To compile and install the IIINSTRUMENT pipeline one needs:

- the GNU C compiler (version 3.2 or later),
- the GNU `gzip` data compression program,
- a version of the `tar` file-archiving program, and,
- the GNU `make` utility.

An installation of the Common Pipeline library (CPL) must also be available on the system. Currently the CPL version 2.1.1 or newer is required. The CPL distribution can be obtained from [1].

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	37 of 40

Please note that CPL itself depends on an existing qfits installation. The qfits sources are available from the CPL download page or directly from the qfits homepage at <http://www.eso.org/projects/aot/qfits>. In conjunction with CPL 2.1.1 qfits 5.3.1 must be used.

In order to run the IIINSTRUMENT pipeline recipes a front-end application is also required. Currently there are two such applications available, a command-line tool called *EsoRex* and the Java based data file organizer, *Gasgano*, which provides an intuitive graphical user interface (see Section 4.2, page 12). At least one of them must be installed. The *EsoRex* and *Gasgano* packages are available at <http://www.eso.org/cpl/esorex.html> and <http://www.eso.org/gasgano> respectively.

For installation instructions of any of the additional packages mentioned before please refer to the documentation of these packages.

A.2.2 Compiling and installing the IIINSTRUMENT pipeline

The IIINSTRUMENT pipeline distribution kit 1.0 contains:

iiinstrument-manual-1.0.pdf	The IIINSTRUMENT pipeline manual
install_pipeline	Install script
qfits-5.3.1.tar.gz	QFITS 5.3.1
cpl-2.1.1.tar.gz	CPL 2.1.1
esorex-3.5.1.tar.gz	esorex 3.5.1
gasgano-2.2.3-Linux.tar.gz	GASGANO 2.2.3 for Linux
gasgano-2.2.3-SunOS.tar.gz	GASGANO 2.2.3 for SunOS
eclipse-sinfo.tar.gz	eclipse library modified to support IIINSTRUMENT data reduction
iiinstrument-1.2.0.tar.gz	IIINSTRUMENT 1.2.0
iiinstrument-calib-1.2.0.tar.gz	IIINSTRUMENT calibration files 1.2.0

Here is a description of the installation procedure:

1. Change directory to where you want to retrieve the IIINSTRUMENT pipeline recipes 1.2.0 package. It can be any directory of your choice but not:

```
$HOME/gasgano
$HOME/.esorex
```

2. Download from the ESO ftp server, <http://www.eso.org/pipelines/>, the latest release of the IIINSTRUMENT pipeline distribution.
3. Verify the checksum value of the tar file with the cksum command.
4. Unpack using the following command:

```
tar -xvf iiinstrument-kit-1.2.tar
```

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	38 of 40

Note that the size of the installed software (including *Gasgano*) together with the static calibration data is about 27Mb.

5. Install: after moving to the top installation directory,

```
cd sinfo-kit-1.2
```

it is possible to perform a simple installation using the available installer script (*recommended*):

```
./install_pipeline
```

(beware: the execution may take a few minutes on Linux and several minutes on SunOS).

Note that this release still needs to link to the eclipse library. At the end of the installation the user in addition to follow what reported by the installation script, needs to source an file (\$HOME/..eclipse_bash.rc or \$HOME/..eclipse_bash.rc, depending from the user shell) to set a few environment variables used by a few low level eclipse library based modules.

By default the script will install the IIINSTRUMENT recipes, *Gasgano*, *EsoRex*, all the necessary libraries, and the static calibration tables, into a directory tree rooted at \$HOME. A different path may be specified as soon as the script is run.

The only exception to all this is the *Gasgano* tool, that will always be installed under the directory \$HOME/gasgano. Note that the installer will move an existing \$HOME/gasgano directory to \$HOME/gasgano.old before the new *Gasgano* version is installed.

Important: the installation script would ensure that any existing *Gasgano* and *EsoRex* setup would be inherited into the newly installed configuration files (avoiding in this way any conflict with other installed instrument pipelines).

Alternatively, it is possible to perform a manual installation (*experienced users only*): the README file located in the top installation directory contains more detailed information about a step-by-step installation.

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	39 of 40

B Abbreviations and acronyms

ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
CalibDB	Calibration Database
CPL	Common Pipeline Library
DFO	Data Flow Operations department
DFS	Data Flow System department
DMD	Data Management and Operations Division
DRS	Data Reduction System
ESO	European Southern Observatory
ESOREX	ESO-Recipe Execution tool
FITS	Flexible Image Transport System
FOV	Field Of View
FPN	Fixed Patter Noise
GUI	Graphical User Interface
OB	Observation Block
PSO	Paranal Science Operations
QC	Quality Control
RON	Read Out Noise
SOF	Set Of Frames
UT	Unit Telescope
VLT	Very Large Telescope

ESO	IIINSTRUMENT Pipeline User Manual	Doc:	VLT-MAN-ESO-19500-XXXX
		Issue:	Issue 0.7
		Date:	Date March 2014
		Page:	40 of 40

C Troubleshooting Guide