European Organisation for Astronomical Research in the Southern Hemisphere

Programme: GEN

Project/WP: Science Operation Software Department

IIINSTRUMENT Pipeline User Manual

Document Number: ESO-XXXXXX

Document Version: 0.1

Document Type: Manual (MAN)

Released on: 2016-05-02

Document Classification: Public

Prepared by: IIINSTRUMENT Pipeline Team

Validated by:

Approved by:

Name

Authors

Name	Affiliation
N.N.	Far out in the uncharted backwaters of the unfashionable end of the western spiral arm of the Galaxy

Change Record from previous Version

Affected Section(s)	Changes/Reason/Remarks
All	First draft

Contents

1 Introduction

This template contains the basic layout of the User Manual and comments/instructions in "red" on how the different sections should be filled. They also indicate which parts are provided by ESO. These comments should be removed from the final document. If there are questions regarding the template, the instructions or the intended content of certain sections, please get in touch with your ESO contact person.

1.1 Scope

1.2 Acknowledgements

1.3 Stylistic conventions

Throughout this document the following stylistic conventions are used:

bold	in text sections for commands and other user input which has to be typed as shown
italics	in the text and example sections for parts of the user input which have to be replaced with real contents
`teletype`	in the text for FITS keywords, program names, file paths, and terminal output, and as the general style for examples, commands, code, etc

In example sections expected user input is indicated by a leading shell prompt.

In the text **bold** and *italics* may also be used to highlight words.

1.4 Notational Conventions

Hierarchical FITS keyword names, appearing in the document, are given using the dot–notation to improve readability. This means, that the prefix "HIERARCH ESO" is left out, and the spaces separating the keyword name constituents in the actual FITS header are replaced by a single dot.

2 Related Documents

2.1 Applicable Documents

[AD01] ESO-XXXXXX 1.0 TBD

2.2 Reference Documents

[RD01] ESO-XXXXXX IIINSTRUMENT User Manual

3 Definitions, Acronyms and Abbreviations

CalibDB	Calibration Database
CPL	Common Pipeline Library
CCD	Charge Coupled Device
DFS	Data Flow System
DRS	Data Reduction System
ESO	European Southern Observatory
EsoRex	ESO Recipe Execution Tool
FITS	Flexible Image Transport System
FOV	Field Of View
GUI	Graphical User Interface
LSF	Line Spread Function
IIINSTRUMENT	TODO
OB	Observation Block
pixel	picture element (of a raster image)
PSF	Point Spread Function
QC	Quality Control
SDP	Science Data Product
SOF	Set Of Frames
TBD	To be defined
TBC	To be confirmed
VLT	Very Large Telescope
WCS	World Coordinate System

4 Overview

Document overview and purpose goes here

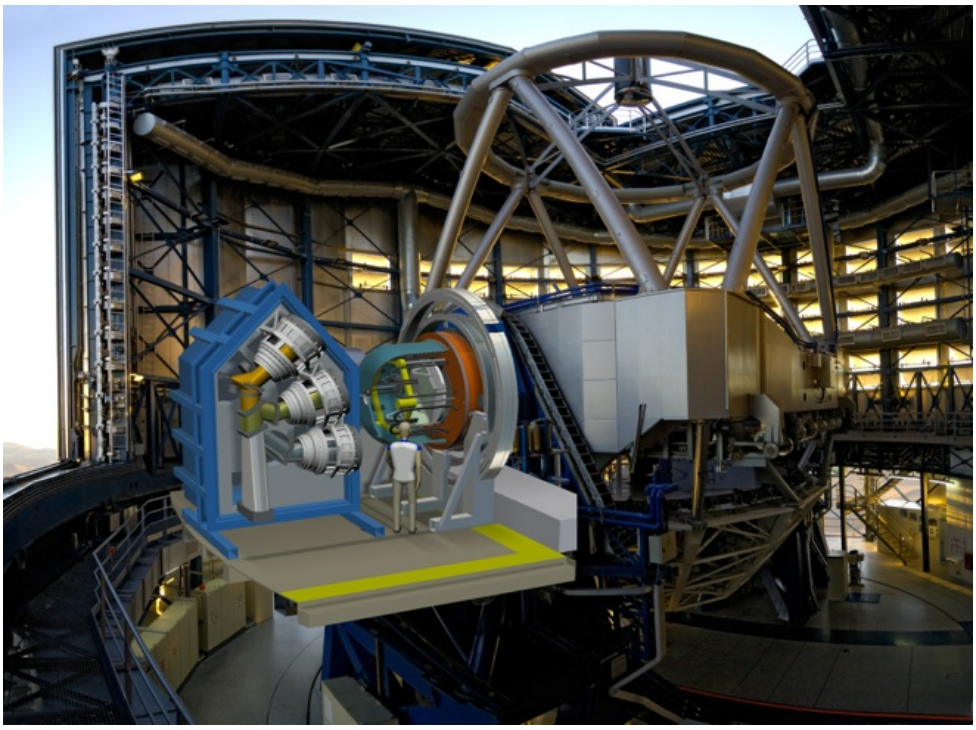

Figure 5.1: Graphical representations of IIINSTRUMENT.

5 The IIINSTRUMENT Instrument

Instrument overview, description of instrument modes, etc.

6 Quick Start

Cookbook-like description of pipeline recipes and their usage

6.1 The IIINSTRUMENT Pipeline Recipes

Overview and/or summary of available pipeline recipes, i.e. list of available pipelines and a brief (one-liner) description of its purpose.

6.2 Running the IIINSTRUMENT Pipeline Recipes

6.2.1 Getting Started with Gasgano

Basic description of Gasgano and its usage with the IIINSTRUMENT and its recipes. Maybe omitted if Gasgano, for technical reasons cannot be supported (MUSE memory requirements for instance). If present, major parts will be provided by ESO.

6.2.2 Getting Started with EsoRex

Basic description of EsoRex. Contents will partially be provided by ESO, example follows.

EsoRex is a command-line tool which can be used to execute the recipes of all standard VLT/VLTI instrument pipelines. With *EsoRex* in your path, the general structure of an *EsoRex* command line is

```
1> esorex [esorex options] [recipe [recipe options] [sof [sof]...]]
```

where options appearing before the recipe name are options for *EsoRex* itself, and options given after the recipe name are options which affect the recipe.

All available *EsoRex* options can be listed with the command

```
1> esorex --help
```

and the full list of available parameters of a specific recipe can be obtained with the command

```
1> esorex --help <recipe name>
```

The output of this command shows as parameter values the current setting, i.e. all modifications from a configuration file or the command line are already applied.

The listing of all recipes known to *EsoRex* can be obtained with the command

```
1> esorex --recipes
```

The last arguments of an *EsoRex* command are the so-called *set-of-frames*. A *set-of-frames* is a simple text file which contains a list of input data files for the recipe. Each input file is followed by an unique identifier (frame classification or frame tag), indicating the contents of this file. The input files have to be given as an absolute path, however *EsoRex* allows the use of environment variables so that a common directory prefix can be abreviated. Individual lines may be commented out by putting the hash character (#) in the first column. An example of a *set-of-frames* is shown in the following:

```
1> cat bias.sof
/data/iiinstrument/raw/IIINSTRUMENT.2019-03-29T09:48:53.153.fits BIAS
$RAW_DATA/IIINSTRUMENT.2019-03-29T09:50:36.645.fits BIAS
$RAW_DATA/IIINSTRUMENT.2019-03-29T09:52:16.513.fits BIAS
$RAW_DATA/IIINSTRUMENT.2019-03-29T09:53:47.996.fits BIAS
#$RAW_DATA/IIINSTRUMENT.2019-03-29T09:55:04.515.fits BIAS
```

These *set-of-frames* files will have to be created by the user using a text editor, for instance. Which classification has to be used with which recipe will be shown in section 6.3

Finally, if more than one *set-of-frames* is given on the command-line *EsoRex* concatenates them into a single *set-of-frames*.

6.3 Data Organization

Outline of how to organize IIINSTRUMENT data, i.e. how to get to a correct SOF file, which classification tags are accepted by the recipes and how they are defined in terms of header keywords.

7 Known Issues

*Know issues, problems and limitations related to the **current release** of the pipeline (if possible with a work-around) go here.*

8 IIINSTRUMENT Data Description

Description of the different raw frames processed by the pipeline, including data classification and association keywords (cf. example)

8.1 Bias

Frame tag: `BIAS`
Processed by: `iiinstrument_mbias`

Classification Keywords	Association Keywords	Remarks
`DPR.CATG = CALIB`	`DET.CHIP.ID`	Detector chip identification
`DPR.TYPE = BIAS`	`SEQ.ARM`	Spectrograph identification
	`DET.READ.CURID`	Detector read-out mode

An example is shown in Table 8.1.

9 Static Calibration Data

Description of the different static data files needed (line catalog for instance), including data classification and association keywords (cf. example).

9.1 Line Catalog

Frame Tag: `LINE_CATALOG`
Classification keywords: `PRO.CATG = LINE_CATALOG`

Column Name	Column Type	Description
`Lambda`	float	Line wavelength nm
`Flux`	float	Relative line flux
`Ion`	string	Ion from which the line originates
`Quality`	int	Quality flag (0: undetected line, 1: etc.)
`Comment`	string	Optional column for further remarks

An example is shown in Table **??**.

Figure 10.1: The IIINSTRUMENT "Assocoation Map" showing the required and optional input for each recipe.

10 Data Reduction

Description and of the data reduction process from raw files to the final product, including walk-through of a standard data reduction session.

10.1 The IIINSTRUMENT Date Reduction Pipeline

The IIINSTRUMENT association map is shown in Figure 10.1.

11 Recipe Reference

Recipe reference with detailed description of input frames, parameters products and provided QC information (see example below).

11.1 iiinstrument_mbias

11.1.1 Description

11.1.2 Input Frames

Frame Tag	Type	Count	Description
BIAS	raw	3 (min)	Raw bias
BAD_PIXEL_TABLE	calib	1 (opt)	Table of a priori known bad pixel locations

11.1.3 Product Frames

Default File Name	Frame Tag	Description
master_bias.fits	MASTER_BIAS	Master bias frame
bad_pixel_mask.fits	BAD_PIXEL_MASK	Bad pixel mask

11.1.4 Quality Control Parameters

`QC.BIAS.MASTER.AVG`	Average pixel value of the master bias frame
`QC.BIAS.MASTER.MED`	Median pixel value of the master bias frame
`QC.BIAS.MASTER.RMS`	RMS of the master bias frame

11.1.5 Recipe Parameters

Parameter	Type	Values	Description
stack-method	string	average, median, **ksigma**	Stacking method
clip-low	double	**3.**	Low sigma threshold for *ksigma* pixel rejection method

continued on next page

continued from previous page

Parameter	Type	Values	Description
clip-high	double	**3.**	High sigma threshold for *ksigma* pixel rejection method

12 Algorithms

The first part contains the detailed discussion and the mathematical description of the used algorithms. The second part contains the recipe algorithm, i.e. a high-level description of the recipe flow-chart and how it changes for different parameter settings.

12.1 General Algorithms

12.2 Recipes Algorithms

A Installation

Installation instructions for the different package types offered by ESO and build instructions for experienced users. This part will be filled by ESO to a large extent.

A.1 System Requirements

A.2 Installing the IIINSTRUMENT Pipeline

A.3 Building the IIINSTRUMENT Pipeline

A.3.1 Build Requirements

B Troubleshooting

Hints on diagnosing problems and possible work-arounds or debug settings.