EUROPEAN SOUTHERN OBSERVATORY

Organisation Européene pour des Recherches Astronomiques dans l'Hémisphère Austral
Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre

ESO - European Southern Observatory
Karl-Schwarzschild Str. 2, D-85748 Garching bei München

Very Large Telescope
Paranal Science Operations
GIRAFFE data reduction cookbook

Doc. No. VLT-MAN-ESO-13700-4034

Issue 85.0, Date 29/08/2009

Prepared C. Melo, J. Smoker . . . . . .
Date Signature

Approved G. Marconi . . . . . .
Date Signature

Released C. Dumas . . . . . .
Date Signature

This page was intentionally left blank

Change Record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
79	26/08/2006		First version
85	29/08/2009	All	Update: Bias history effect, How to make slit geometry table. ESOrex section expanded, Sects on common problems in reducing data and on reducing data with old calibrations added. Example of sky lines added. IFU and Argus section improved.

Issues to be covered in a future version

Different extraction methods. Sky subtraction example. Argus data reduction.

This page was intentionally left blank

Contents

1 Introduction

1.1 Purpose

1.2 Reference documents

1.3 Abbreviations and acronyms

The following abbreviations and acronyms are used in this document:

SciOp	Science Operations
ESO	European Southern Observatory
Dec	Declination
eclipse	ESO C Library Image Processing Software Environment
ESO-MIDAS	ESO's Munich Image Data Analysis System
FITS	Flexible Image Transport System
IRAF	Image Reduction and Analysis Facility
PAF	PArameter File
RA	Right Ascension
UT	Unit Telecope
VLT	Very Large Telescope

1.4 Stylistic conventions

The following styles are used:

bold	in the text, for commands, etc., as they have to be typed.
italic	for parts that have to be substituted with real content.
box	for buttons to click on.
`teletype`	for examples and filenames with path in the text.

Bold and *italic* are also used to highlight words.

Table 1: A few multifiber spectrographs around the world.

Instrument	Telescope	Observatory	Number of objects	FOV	Resolution
Hectospec	6.7-m MMT	MMT	300	1 deg	1000–2500
6dF	1.2-m UK Schmidt	AAO	150	5.7 deg	
2dF	3.9-m AAT	AAO	400	2 deg	
Hydra	3.5-m WYIN	KPNO	90	1 deg	
FLAMES	8.2-UT2	VLT	135/8	25 arcmin	5600-46000/48000

2 A brief overview of data reduction of multi-fiber spectroscopy data

This section presents a brief description of the reduction of multi-fiber spectroscopic data. If you are a beginner who just got your first data set, this section is probably worth reading. Although the data collected with FLAMES/GIRAFFE is used as an example, the steps outlined here are the typical ones for data reduction of any multi-fiber spectrograph.

The experienced user might want to jump this section going directly to Sect. 3 where an example of the use of the ESO GIRAFFE pipeline is given. Like any other reduction package, the GIRAFFE pipeline has many adjustable parameters allowing to fine-tune the data-reduction. We refer to the pipeline manual to a full description of these parameters.

2.1 Multi-fiber spectroscopy

If you have already had a look at one of your raw science frames, the advantage of using a multi-fiber spectrograph is clear. In one single shot hundreds of objects can be observed. Fibers can be placed at almost any place within the telescope focal plane (within 25arcmin in the case of FLAMES) as shown in Fig. 1. This multiplex capability has of course a cost. Due to the limited size of the detectors only a small piece of the spectrum is recorded for each target. Also, the fibers most commonly used in astronomy have poor transmission in the blue region of the spectrum. An advantage in fibres over Multi Object Spectroscopy (MOS) is that there are no issues with slit centering which can impact on radial velocities with slits.

There are number of multi-fiber spectrographs around the world. The main characteristics of some of them are given in Table 1.

In the case of FLAMES, the fibers are arranged in a circular pattern around a plate of the size of the telescope focal plane. The fiber end "looking" at the sky has a magnetic button on it. The magnetic side of this button sticks to the plate whereas the other side is open to leave the light of your target to get into the button. In the case of FLAMES the light that enters into the button is deviated into the fibers by a tiny prism. The other end of these fibers are arranged along the long-slit of the spectrograph. Once the light of the fibers get inside the spectrograph, the desired spectral order is selected by order sorting filters. It is then reflected into a double pass collimator and goes to the grating. After an intermediate spectrum is formed, the light is finally re-imaged on the CCD. Although all multi-fiber instruments differ from each other in technical details, the basic idea is the same for all of them.

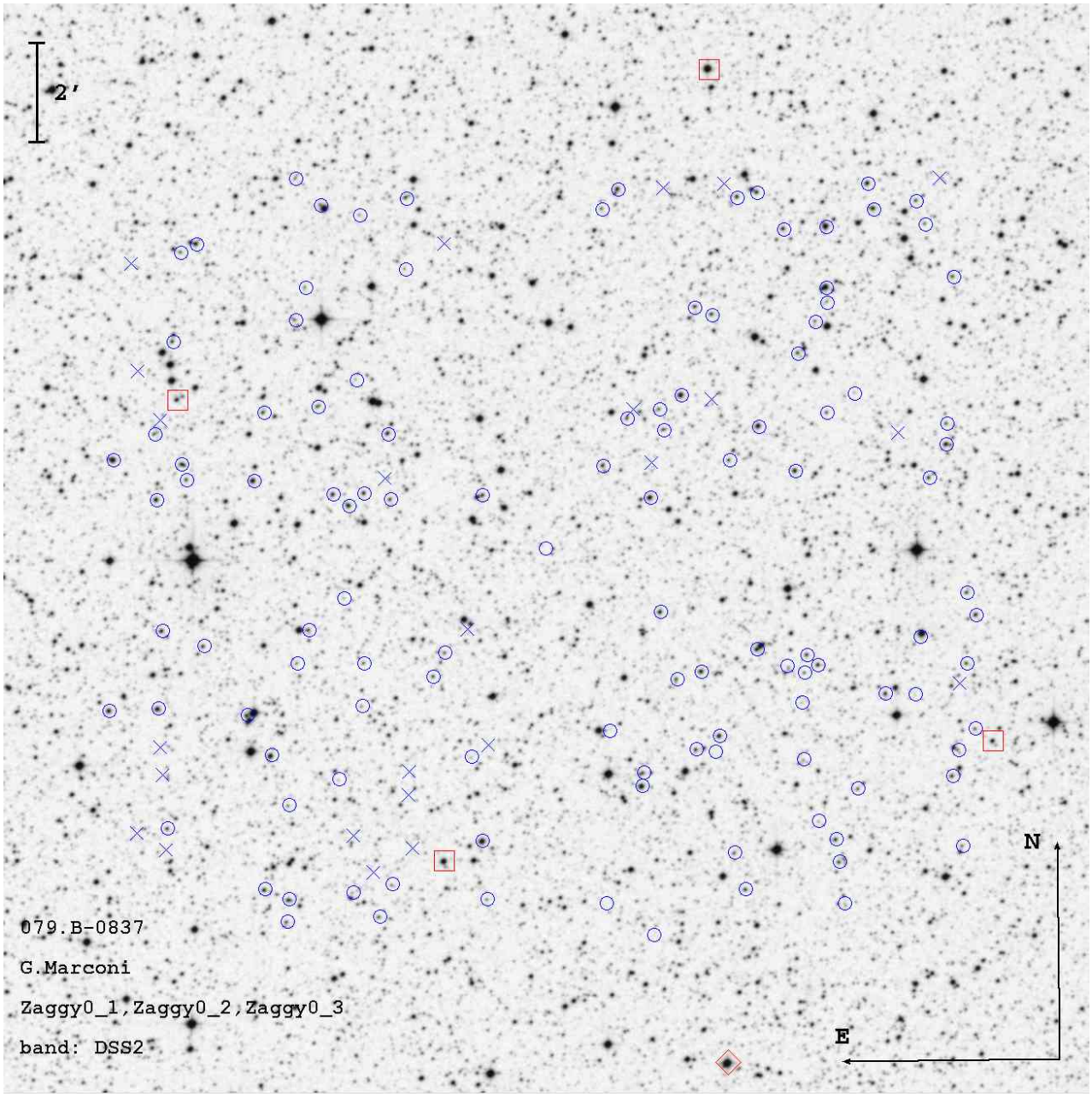

Figure 1: An example of the potential of multi-fiber spectrographs. In one shot up to 135 spectra are recorded by GIRAFFE and up 8 by UVES. The figure is a finding chart of a typical FLAMES observation. Circles indicate science targets. Sky positions are marked with crosses and the four FACBs used for centering the field are seen as squares.

The basic steps of the reduction of multi-fiber spectroscopy data are the following:

- correcting frames for detector cosmetic effects
- determining the location of your data on the detector, i.e., fiber tracing
- extraction of flat-field spectrum and determination fiber transmission
- scattered light correction
- standard star calibration in the case of IFU or ARGUS
- wavelength calibration
- extraction of science data
- sky subtraction

2.2 Correcting detector cosmetic effects

Data reduction of any nature starts by correcting the detector defects referred as cosmetics. These effects and the way to correct them have largely described in different cookbooks[1]. Here we briefly described the main defects.

- **Subtracting the Bias level.** A bias voltage is routinely applied to CCD detectors to ensure that, as near as possible they are operating in a linear manner. This current has the effect that a non-zero count is recorded in all pixels.
- **Subtracting the dark current.** Dark current arises from thermal energy within the silicon lattice comprising the CCD. Electrons are created over time that are independent of the light falling on the detector. These electrons are captured by the CCD's potential wells and counted as signal.
- **Bad pixel correction.** Any detector has a certain number of pixels that are bad, in the sense that these bad pixels record the information inaccurately. This happens because either they are brighter than the others (hot pixels) or because they have low or no sensitivity at all (dead pixels). Bad pixels (or bad columns) are fixed by interpolating the signal in the neighbor pixels (or columns).
- **Cosmic-ray hits.** When a high energy particle hits the CCD, it loses its energy by knocking the atoms constituting the chip itself. That liberates many electrons that cause a bright spot on the image. These high energy particle can either be genuine cosmic rays (exotic particle produced by exploding supernovae, black holes, etc.), or just the product of the decay of some radioactive atoms present in the lenses just above the CCD.
- **Correction of pixel-to-pixel variations.** Pixels in a CCD have all different sensitivities. This means that some of them will convert the light photons more efficiently into electrons than others. Thus an uniform light source like the bright sky or an illuminated screen will not appear uniform on the CCD. This effect is corrected by taking uniformly illuminated images (or flat-fields). Those images are used to construct a sensitivity map of the CCD.

[1] A good starting point is the cookbook *A User's Guide to CCD Reductions with IRAF*, by Philip Massey which can be found in the IRAF website http://iraf.noao.edu

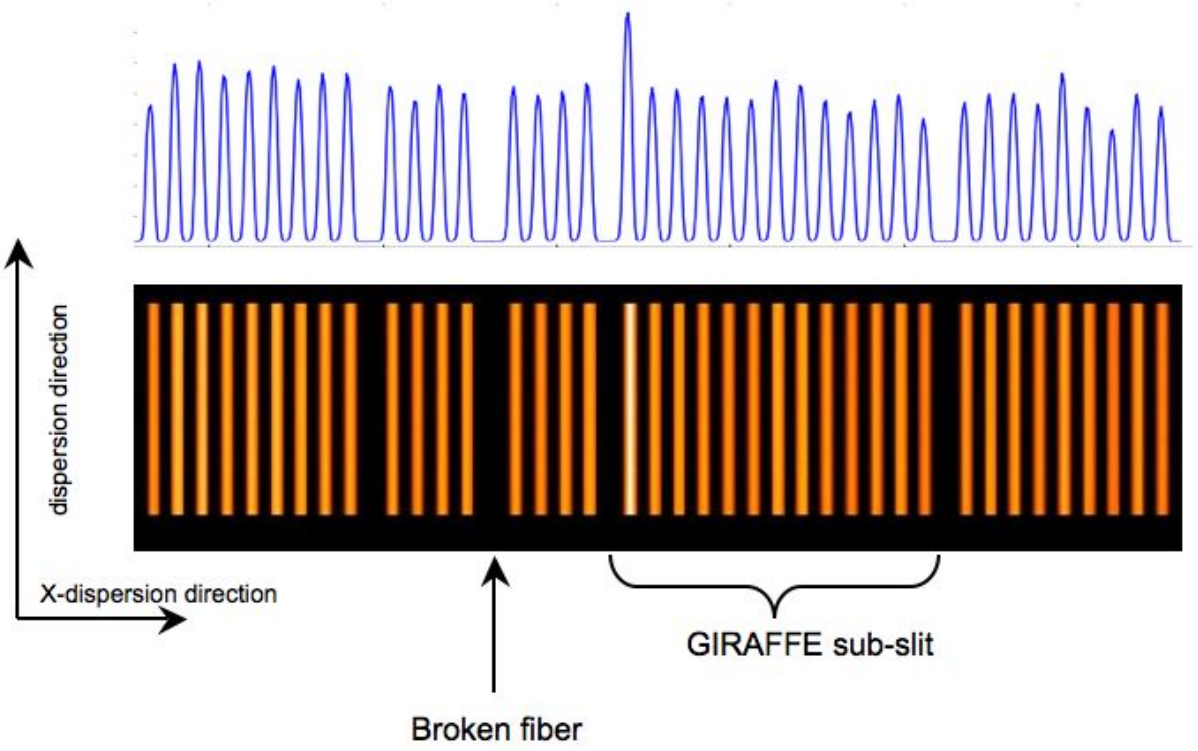

Figure 2: An extract of a raw image of a flat-filed frame is shown the bottom panel. The sub-slits (packets of fibers) defined in the previous figure are clearly seen in the image as well as a broken fiber. In the top panel, a cross section of the frame is shown where the nearly Gaussian profile of the fibers can be seen.

In the case of spectroscopic data, the first three steps are carried out in the same way as done in imaging data reduction whereas cosmic ray cleaning and flat-field corrections are not. The correction of these two effects will be discussed in the next sessions.

2.3 Fiber localization and tracing

As described above and in Fig. 2, the fibers are arranged side by side along the spectrograph slit. After being dispersed by the grating, the spectrum of each is recorded on the CCD also side by side. The direction along which the light is dispersed is called the dispersion direction. The direction perpendicular to the dispersion is called the cross-dispersion direction (or spatial direction in slit spectroscopy). These directions are also indicated in Fig. 2.

Thus the first task in the data reduction process (after cleaning the detector defects) is to know where the spectrum of each fiber actually is on your 2 dimensional CCD. This processes is called fiber localization.

First, a exposure with all fibers uniformly illuminated by a calibration lamp is taken. This same exposure will be used to flat-field the data later. Then a line is cut along the cross-

dispersion direction. In the top panel of Fig. 2 we see a series peaks more or less evenly spaced. Each of these peaks corresponds to a fiber. In many pipelines, the fiber profiles is approximated by a Gaussian function. The pipeline fit each of those peaks with a Gaussian function and stores for each fiber its center and width.

In both panels of Fig. 2 we can easily distinguish three packets of fibers with a larger gap in between. Each packet represents a GIRAFFE sub-slit. We might also find a gap within a given packet. This happens when a fiber is broken. In the bottom of Fig. 2 we show an extract of a raw image of a flat-field exposure. Three packets of fibers are seen. In the second one, there is a missing (broken) fiber.

In order to deal with broken fibers, the pipeline uses the fact that the size of the fibers is known and the instrument is stable to a point that the center of the fibers don't move by more than 1 pixel. So the pipeline knows where a given fiber should lie and if the localization algorithm cannot measure any signal there that fiber is declared as broken.

The second step once the initial position of the fibers is known is to determine the fiber profile along the dispersion direction. Using the initial position for a given fiber, the pipeline moves a couple of pixels along the dispersion direction and, again, it carries out a Gaussian fit at this new position. A new center and width are found. This is repeated until the edge of the CCD is reached. At the end the pipeline determine a sort of tube or tunnel where the science data will be recorded. An example of these tubes are shown at Fig. 3.

2.4 Extraction, flat-field spectra and fiber transmission

Once these "tubes" have been determined we can extract the signal on the CCD. The first thing to be extracted using the same flat-field frame is the flat-field spectrum.

From Fig. 2 we know already that the signal spreads over many pixels. In the case of GIRAFFE, the MEDUSA fiber profile is spread over 6 pixels. There are two ways of summing the information spread over the fiber profile. In the simplest case we add up all pixels inside the fiber profile. This is what is called standard or summed extraction.

The standard extraction ignores the fact that there pixels which contains more counts (better quality information) than others. They all contribute with equal weight to the final spectrum. Since the noise associated to each pixel is given by the squared-root of the number of counts on this pixel (Poisson noise), we can easily see that give the same weight to pixel with lower counts means that we are adding noise to our final spectrum.

The shape of the fiber profile can be used as a weight function, thus instead of a simple addition we weight its flux by its noise. In this way, better pixels will give a higher contribution to the final spectrum. This is called optimum extraction (e.g., Horne, 1986, PASP 98, 609). Note that at present optimal extraction is only present for Medusa and not for Argus or IFU.

The optimum extraction has an additional advantage with respect to the standard extraction. Since we know that the distribution of the intensity of the pixels should follow a smooth and continuous function, any pixel deviating a few per cent of this profile is likely to be cosmic-ray! The pixel hit by a cosmic ray can be replaced by the interpolation of its neighbors cleaning the final spectrum.

Extraction of the spectrum of the flat lamp has two main functions. The first is to correct the pixel-to-pixel variation in our science data. Second, the amount of light entering the fibers is supposed to be similar. Thus any difference of the intensity of the extract flat field spectrum is due to differences in the fiber transmission.

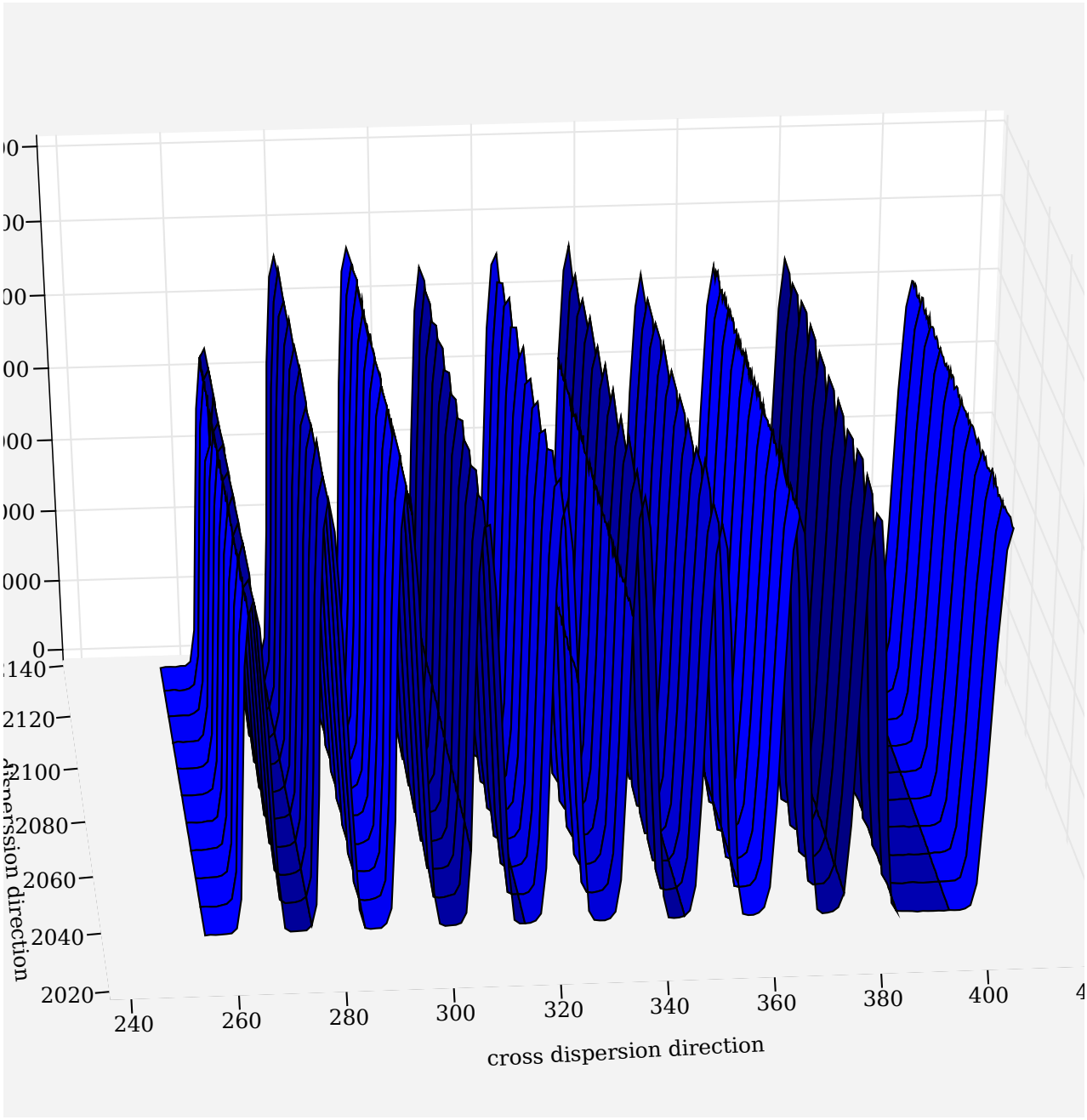

Figure 3: 3D representation of the first packet seen in Fig. 2 showing the Gaussian tubes.

Table 2: GIRAFFE fiber transmission. The values given all losses, focal ratio degradation, optics and coupling. For wavelengths redder than 600nm the transmission is constant. Variations of a few percent between different fibers are measured (see Pasquini et al. 2003).

Fiber type	370nm	400 nm	450 nm	600 nm
MEDUSA	0.47	0.52	0.55	0.61
ARGUS	0.52	0.58	0.62	0.70
IFU	0.49	0.55	0.58	0.66

In imaging or even in slit spectroscopy, one can carry out a two-dimensional flat-field correction. This means that (after some manipulation) your science frame can be divided by the flat-field image.

In the case of fiber spectroscopy we have seen that the intensity of the pixels drops quickly at the edge of the fiber profile. If the two frames are slightly miss-aligned, (i.e., the two profiles don't match exactly each other), the division will produce an sort of parabola instead of a flat-image.

Flat-field are corrections are done in one dimension, i.e., the extracted science data is divided by the flat-field spectrum. In this way we avoid introducing artifact due to the mismatch of the science and the flat-field.

Fibers are not perfect devices. A certain amount of photons that enter in one end don't make it to the other end of the fibers. The amount of lost photons depends of their energy (or wavelength). Typical transmissions as a function of wavelength for the different fiber systems of FLAMES/GIRAFFE are given in Table 2. Values are taken form Pasquini et al. (2003, SPIE 4841, 1682)[2].

Now if you consider a set of fibers sharing the same characteristics (like the MEDUSA fibers in FLAMES, for instance), although they have a similar behavior, they are not exactly similar to each other. Some of them carry light better than others.

In a flat-field frame, the amount of light entering the fibers is assumed to be the same for all fibers. Thus comparing the intensity of the extracted flat-field spectra, we can derive what is called the fiber relative transmission.

This is important when one wants to do additive operations with the fibers and critical in operations like sky subtraction as described in Sec. 2.8 and in Wyse & Gilmore (1992, MNRAS 257, 1).

2.5 Scattered light correction

A better idea of what scattered light is given in Fig. 4. In this figure we show a zoom-in of the base of a packet of fibers. The solid line and dashed lines represent the fiber profiles before and after the bias subtraction. We see that even after the bias removal the signal doesn't go to zero. This remaining signal is the scattered light.

This is because part of the light is scattered inside the spectrograph. This scattered light has two components. A smooth one, covering the whole CCD which is proportional to the amount of light entering the spectrograph.

[2]This paper is available at http://www.eso.org/instruments/flames/doc/spie.ps

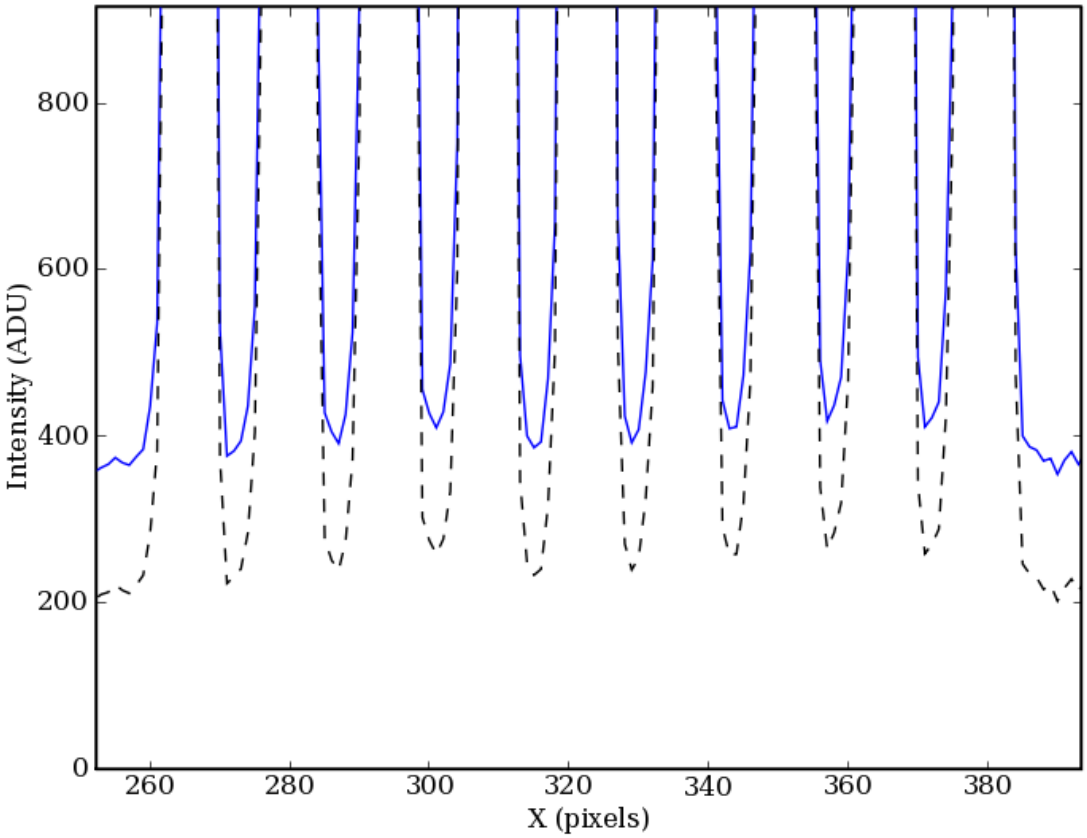

Figure 4: Cut across the fibers. Solid and dashed lines show the minimum level before and after bias subtraction. The remaining ADUs seen in the case of the bias subtracted frame are due to the dark current and scattered light.

A second component is a local one and it is caused by the presence of bright objects (or a simultaneous comparison lamp). In this case, it might happen that the charges of the CCD will jump to the neighbor pixels.

The smooth component is easy to subtract. A two dimensional fit is carried out on the whole CCD using the points of the detector in the gap between two adjacent fibers.

The local component might require much detailed look in the light in the inter-fiber regions to determine whether or not this is an issue. The local component of the scattered light behaves like an extra continuum (i.e., with no spectral features) whose spectral energy distribution follows the one of the object causing the scattered light.

A good correction of the scattered light is essential to achieve an accurate sky subtraction.

2.6 Wavelength calibration

If you look at your spectrum after extraction you might already recognize a few features on it (Hydrogen lines, Li in the case of young stars, etc.). But having that in pixel space is pretty much useless. This is what the wavelength calibration lamp does. Wavelength calibration is achieved using a Hallow-Cathode-Lamp.

An HCL usually consists of a glass tube containing a cathode made of the material of interest, an anode, and a buffer gas (usually a noble gas). A large voltage across the anode and cathode will cause the buffer gas to ionize, creating a plasma. These ions will then be accelerated into the cathode, sputtering off atoms from the cathode. These atoms will in turn be excited by collisions with other atoms/particles in the plasma. As these excited atoms decay to lower states, they will emit photons, which can then be detected and a spectrum can be determined.

The wavelengths of the emission line spectra of these lamps are known from laboratory tests. From our ThAr frame, we measure the (x, y) position on the CCD for the emission lines. From an atlas of emission lines[3] we can associate a pixel to a wavelength. By means of a polynomial fit we can compute the transformation function from pixel to wavelength space, $\lambda \rightarrow f(x, y)$.

2.7 Extraction of the science

The science data is extracted in the same fashion as described above for the flat-field. After extraction, the scattered light is removed, the science spectrum on each fiber is divided by its respective flat-field spectrum, correct for the fiber transmission variants and the keywords containing the information about the wavelength calibration are added to the fits header of the image.

Since the description of these keywords vary from package to package, in most of the cases, a process called rebin is carried out in which we resample our spectra in order to have a constant step in wavelength ($\Delta\lambda = cte$). The keywords used describing an evenly sampled spectrum obey the FITS standards and therefore is the same regardless the data reduction package you are using. Also, rebinned spectra can be easily read as a vector by your own programs written in FORTRAN, C, python, etc.

Your spectra are ready to be analyzed.

[3]NOAO provides Spectral Atlases for different lamps at http://www.noao.edu/kpno/specatlas/index.html

2.8 Sky subtraction

In the case you are dealing with very faint source whose signal is close to the read-out noise of the CCD, you might want to carry out sky subtraction. With some care, sky subtraction as good as 1–3% can achieved. This requires:

- proper bias and dark correction
- scattered light correction
- fiber-to-fiber transmission

Before submitting a proposal PIs should consider downloading previous GIRAFFE spectra from the ESO archive to see how badly their spectra will be contaminated by sky lines. Figures 5 and 6 show how strong sky lines can be in the red.

We note that the current version of the GIRAFFE pipeline does **not** perform sky subtraction. PIs should consider referring to the following articles (amongst others) on how to remove sky features in FLAMES data:

- Battaglia et al., (2008), MNRAS, 383, 183. Contains a detailed description of how sky lines can removed from FLAMES-GIRAFFE spectra.
- Koch et al. (2007), AJ, 134, 566. An estimate in the final accuracy of sky subtraction of $\sim$3 per cent is given for Leo spectra.
- Koch et al. (2006), AJ, 131, 895. An estimate in the final accuracy of sky subtraction of $\sim$2 per cent is given for Carina spectra.

Finally, we refer to Wyse & Gilmore (1992,MNRAS 257, 1) for a very good discussion in the problematic of achieving accurate sky subtraction and how to assess the quality of the scattered light correction and the final sky subtraction using the inter-fiber regions and the broken fibers.

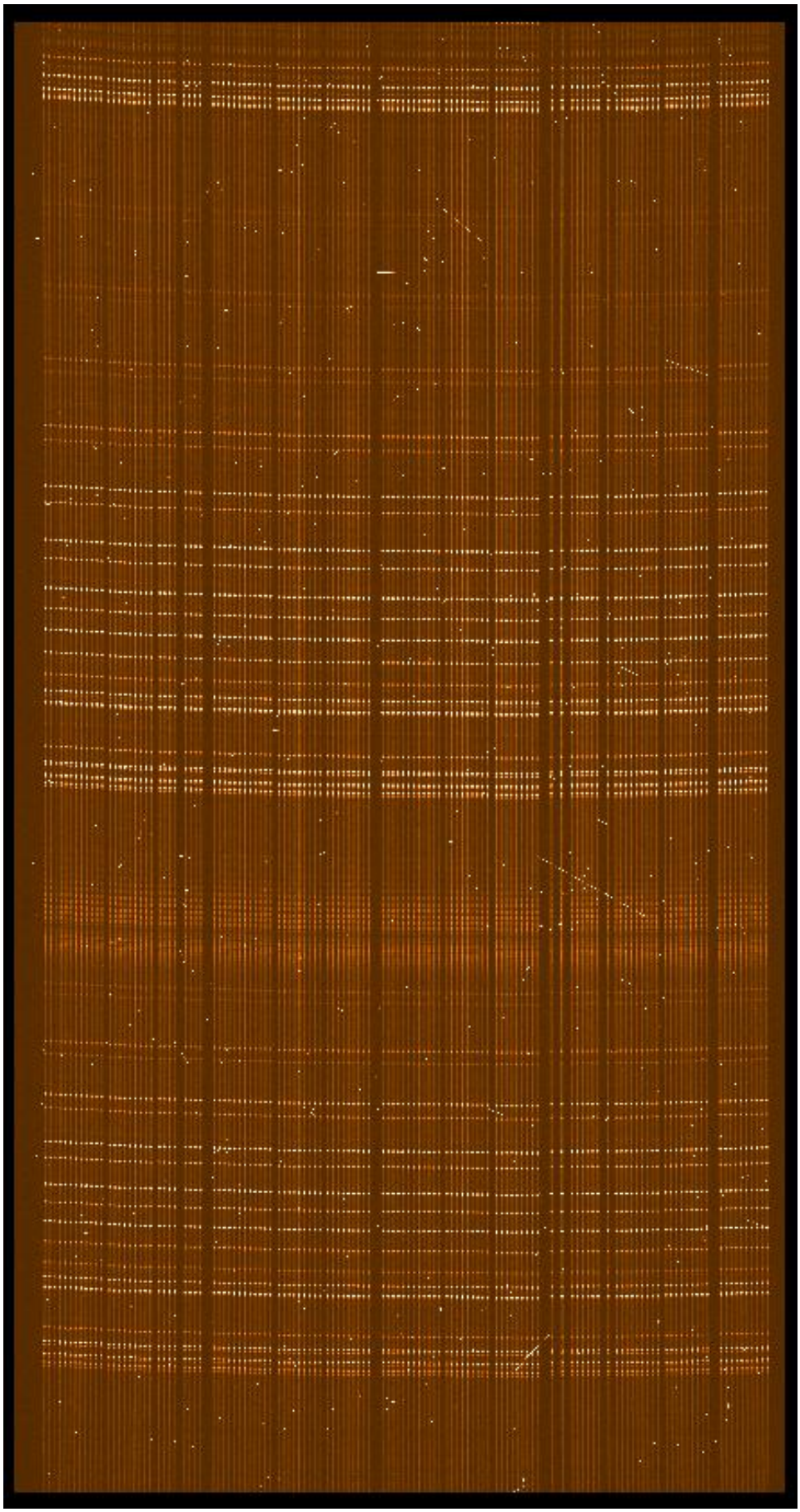

Figure 5: This image shows how especially in the red that there are many sky lines. Removing them can be critical to obtaining good science output. The exposure was taken using GIRAFFE at L881.7-nm for 2750-s.

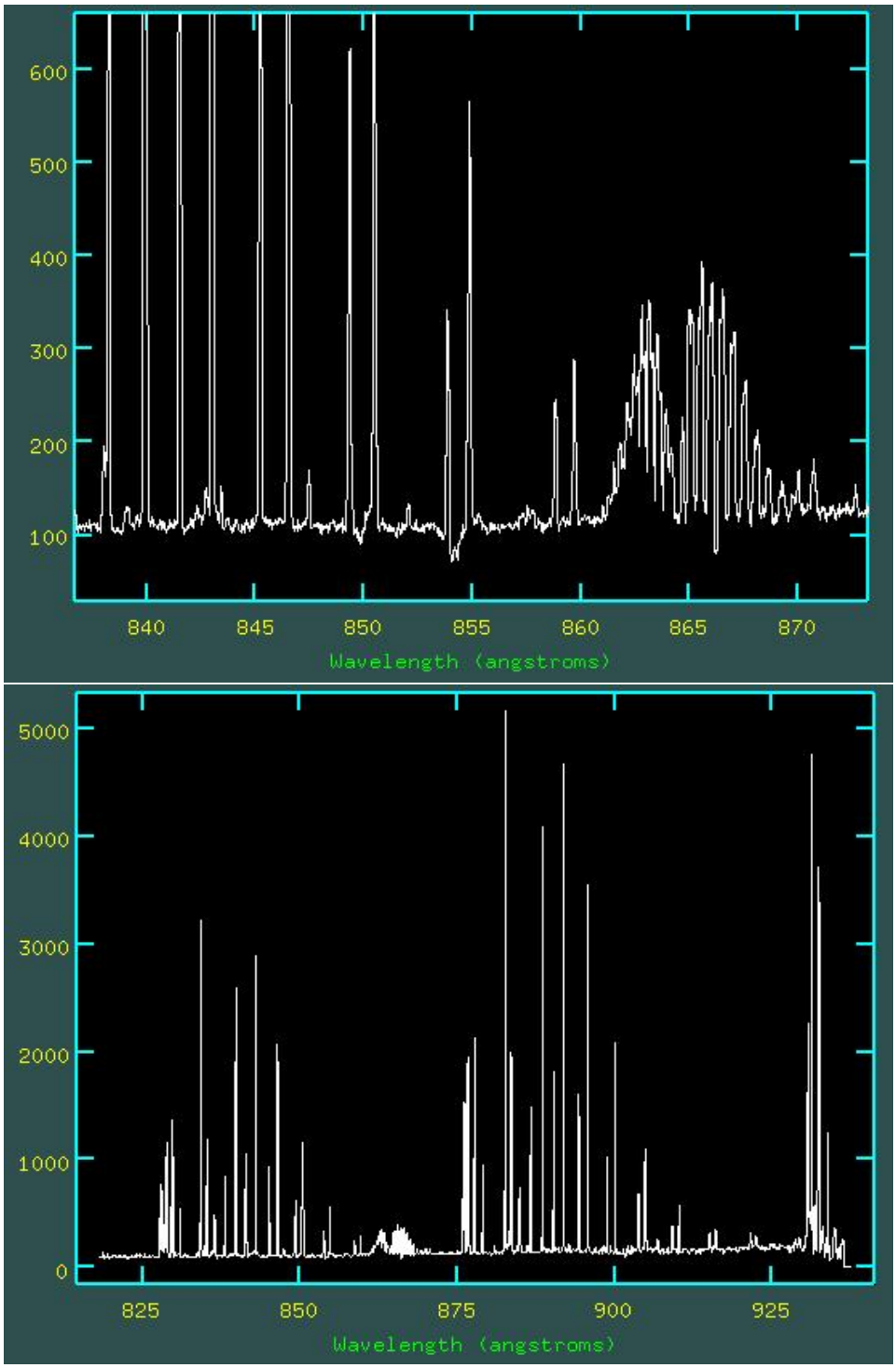

Figure 6: Extracted spectra of the image in Fig. 5 showing a number of bright sky emission lines.

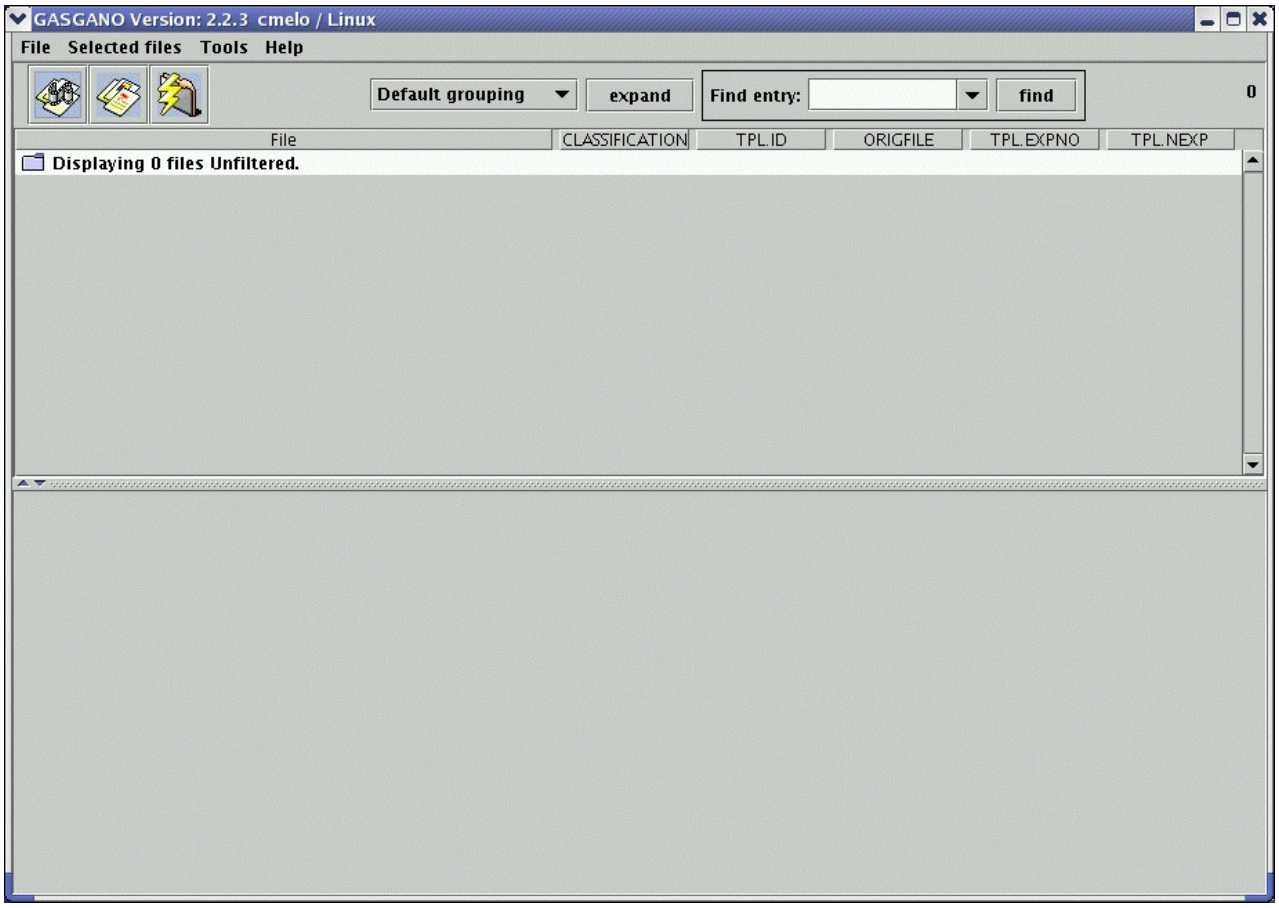

Figure 7: Entry screen of gasgano

3 Pipeline in action: Gasgano, the friendly way

3.1 Before you start

In order to follow this cookbook you need:

- to have the GIRAF-kit installed on your computer[4]
- to have downloaded the demo data from http://www.eso.org/instruments/flames/doc.

3.2 Starting gasgano

start gasgano by typing in command line shell:

```
184dhcp125:GIR-COOKBOOK 38> gasgano &
```

Add the directory containing your raw data, the place where the reduced data will be placed and the giraffe calibration database delivered with the GIRAFFE kit to the list of gasgano directories by clicking on `FILE` and then `ADD`[5]

[4] it can be downloaded at http://www.eso.org/sci/data-processing/software/pipelines/. It contains the GIRAFFE pipeline and its manual, the calibration database, gasgano, and esorex.

[5] Gasgano is a powerful file organizer with many different functionalities. For a detailed description, please refer to the GASGANO user manual.

3.3 Case 1: your calibration data is up-to-date

In this case only the recipe called `giscience` is needed. `giscience` does the final extraction of your science data using an existing calibration database.

As input for `giscience` you need your raw science along a number of calibration products (page 49, Sec. 9.4.1 of the GIRAFFE pipeline user manual). These files are created at the moment you reduce your calibration from scratch (see Sec. 3.4).

1. your science raw frame
2. `MASTER_BIAS`. Two dimensional master bias frame produced by the recipe `gimasterbias` (but see section 3.4.1 about the bias history effect).
3. `FF_LOCCENTROID`. Table created by `giflatfield` containing the center of the PSF profiles fitted for each wavelength bin along the dispersion direction. Gaussian fit is not the default but rather a particular case of the function $PSF(x) = A \times e^{(-|(x-x_{center}|/W)^{\gamma}} + background$. The GIRAFFE pipeline allows for different fitting functions and methods to derive the centroid.
4. `FF_LOCWIDTH`. Also created by `giflatfield` to store the FWHM the fitted PSF function fit (as above) along the dispersion direction
5. `FF_PSFCENTROID`
6. `FF_PSFWDITH`

   You can use either the FF_LOCCENTROID/WIDTH or the FF_PSFCENTROID/WIDTH files as input. The information they contain is the same, but they are determined by different methods. The FF_LOCCENTROID is computed as the location of the barycenter of the flux distribution within the extraction region, and the corresponding width is the half width of the extraction region. The FF_PSFCENTROID is computed from the fit of the fiber profile, and the (half)width from the FWHM. You can even mix the 2 kind of frames, i.e. use for instance the FF_LOCCENTROID together with the FF_PSFWIDTH.

   If you provide both kinds for the centroid or the width, the FF_PSF variant is preferred.
7. `DISPERSION SOLUTION`. Wavelength calibration solution found with `giwavecal`
8. `SLIT_GEOMETRY_SETUP`. Table containing the off-set table to be applied to each fiber in order to correct for the curvature along of the GIRAFFE slit.
9. `GRATING_DATA`. Static table containing information about the physical model of GIRAFFE

Files 2 - 7 are located in the calibration database delivered with the giraffe-kit (/home2/GIRAFFE-ESO/giraf-calib-1.0/cal).

Since the actual filenames of item 2-7 are rather long, in the table above we indicated the `PRO.CATG` keyword:

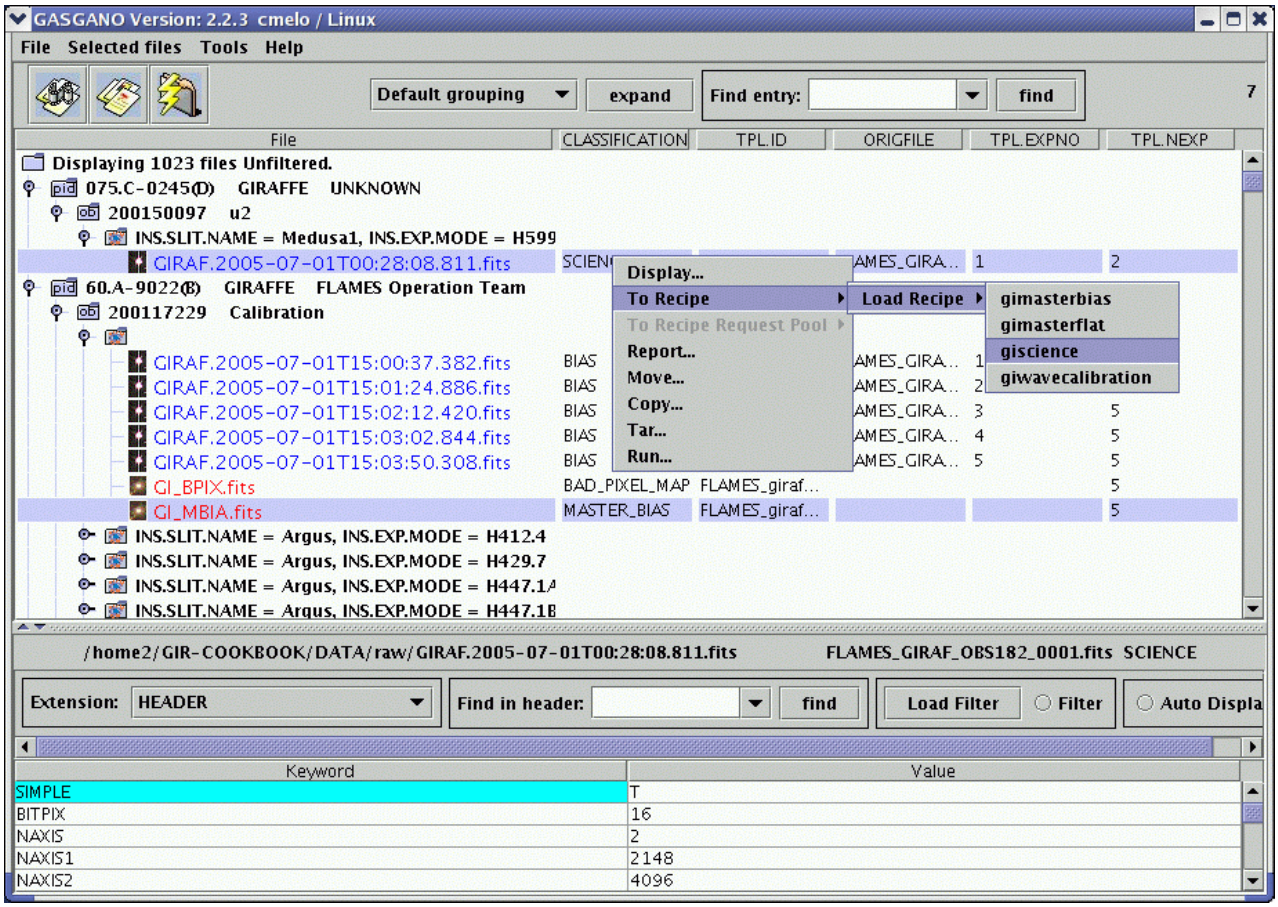

Figure 8: Passing a raw science frame to the recipe `giscience` using gasgano. Files are first selected by holding `CTRL` key and clicking on the calibration and science files. Then with the right button, they are sent the a given recipe. In the example below, the input files are sent to the recipe `giscience`.

FILE	PRO CATG
GI_PDIS_Medusa1_H599.3_o9.tfits	DISPERSION_SOLUTION
GI_PFEX_Medusa1_H599.3_o9.fits	FF_EXTSPECTRA
GI_PLOC_Medusa1_H599.3_o9.fits	FF_LOCCENTROID
GI_PLOW_Medusa1_H599.3_o9.fits	FF_LOCWIDTH
GI_MBIA.fits	MASTER_BIAS

Inside gasgano the keyword `PRO.CATG` appears in the column `CLASSIFICATION`. Now we select in addition to the input science raw frame, all corresponding calibrations. In order to select multiple files in gasgano hold the `CTRL` key and click on the calibration and science files. Once all files are selected, click the right button to open a pull-down menu from which you can choose to which recipe you want to send the input files you just selected (Figure 8).

As shown in Figure 9, a new window will open showing the input parameters for the recipe as well the input frames. Choose the directory where the reduction product should go and then click on `Execute`.

In the Log Messages sub-window you can follow what is going on. If one of the mandatory input files is missing the recipe will stop and the cause of the crash is indicated in the Log window. In the example above the input file `DISPERSION_SOLUTION` is missing (Figure 10). A

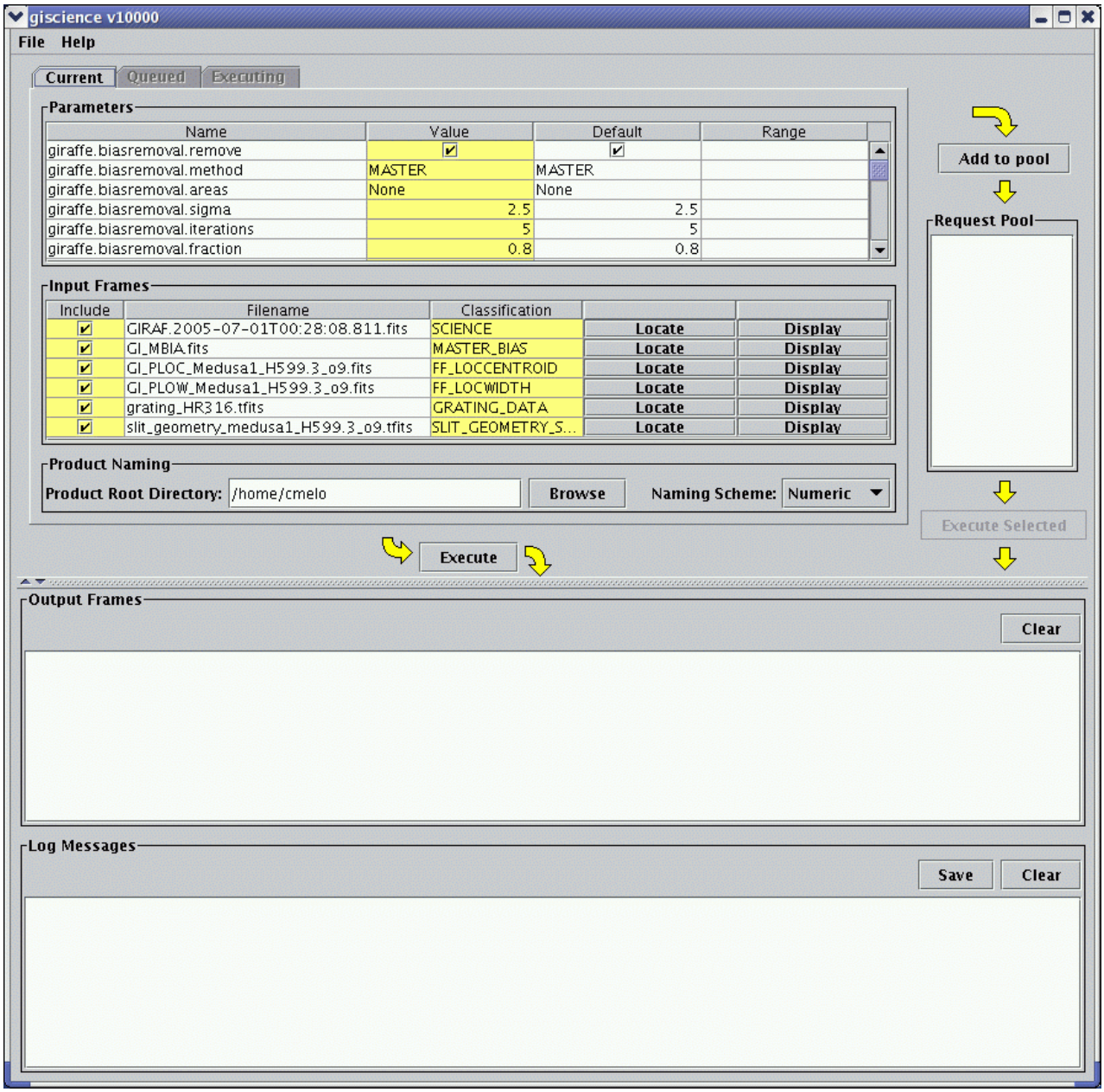

Figure 9: Pop-up panel controlling the parameters and the arguments of `girscience`.

log file is written in the directory chosen to have the reduced data.

```
184dhcp133:reduced 12> ls
giscience_2006-04-30_05:16:12.log
Sun Apr 30 05:16:31 CLT 2006.
  /home2/GIRAFFE-ESO/calib/giraf-calib-1.0b/cal/GI_PLOC_Medusa1_H599.3_o9.fits
group=CALIB level=INTERMEDIATE type=IMAGE tag="FF_LOCCENTROID"
  /home2/GIRAFFE-ESO/calib/giraf-calib-1.0b/cal/GI_PLOW_Medusa1_H599.3_o9.fits
group=CALIB level=INTERMEDIATE type=IMAGE tag="FF_LOCWIDTH"
  /home2/GIRAFFE-ESO/calib/giraf-calib-1.0b/cal/grating_HR316.tfits group=RAW
level=INTERMEDIATE type=IMAGE tag="GRATING_DATA"
  /home2/GIRAFFE-ESO/calib/giraf-calib-1.0b/cal/slit_geometry_medusa1_H599.3_o9.tfits
group=RAW level=INTERMEDIATE type=IMAGE tag="SLIT_GEOMETRY_SETUP"
05:16:12 [ INFO  ] No bad pixel map present in frame set.
05:16:12 [ INFO  ] No master bias present in frame set.
05:16:12 [ INFO  ] No scattered light model present in frame set.
ERROR: 05:16:13 [ ERROR ] Missing master bias frame! Selected bias removal
method requires a master bias frame!
Completion status: FAILURE
Execution error: Execution failed with code 1
```

Select the missing file in the gasgano main window and try again. If no problem occurs, the Log Message indicates "Completion status: SUCCESS" and the following files are placed in the reduced directory:

```
184dhcp133:reduced 16> ls -rtl
total 42896
-rw-rw-r--  1 cmelo  cmelo      2364 Apr 30 05:16 giscience_2006-04-30_05:16:12.log
-rw-rw-r--  1 cmelo  cmelo  33586560 Apr 30 05:18 science_reduced_0000.fits
-rw-rw-r--  1 cmelo  cmelo   1425600 Apr 30 05:18 science_extspectra_0000.fits
-rw-rw-r--  1 cmelo  cmelo   1425600 Apr 30 05:18 science_extpixels_0000.fits
-rw-rw-r--  1 cmelo  cmelo   1425600 Apr 30 05:18 science_exterrors_0000.fits
-rw-rw-r--  1 cmelo  cmelo   1425600 Apr 30 05:18 science_exttraces_0000.fits
-rw-rw-r--  1 cmelo  cmelo   2269440 Apr 30 05:18 science_rbnspectra_0000.fits
-rw-rw-r--  1 cmelo  cmelo   2269440 Apr 30 05:18 science_rbnerrors_0000.fits
-rw-rw-r--  1 cmelo  cmelo      5204 Apr 30 05:18 giscience_2006-04-30_05:18:15.log
```

The name convention is the following. The recipe name, followed by the type of the product and a counter which increments automatically in order to avoid overwriting the products already present in the directory. Note that the incrment *only* will work if you have `Naming Scheme: Numeric` as opposed to `Naming Scheme: Overwrite`.

Let us have a look in the reduced spectra. A description of the files produced by the girscience recipe is given at user manual of the GIRAFFE pipeline (Sec. 9.4.5, p. 58). You most likely are interested in looking at the file containing your rebinned reduced spectra which according to the pipeline name scheme is `science_rbnspectra_NNNN`.

This file contains two HDUs, the first one with the image itself and a second one with a binary table with the information of the configuration file used for fiber allocation.

Any information in the image header can be easily retrieved with the `dfits` and `fitsort`[6] commands, for instance:

[6] `dfits` and `fitsort` are part of the ECLIPSE reduction routines and come with scisoft.

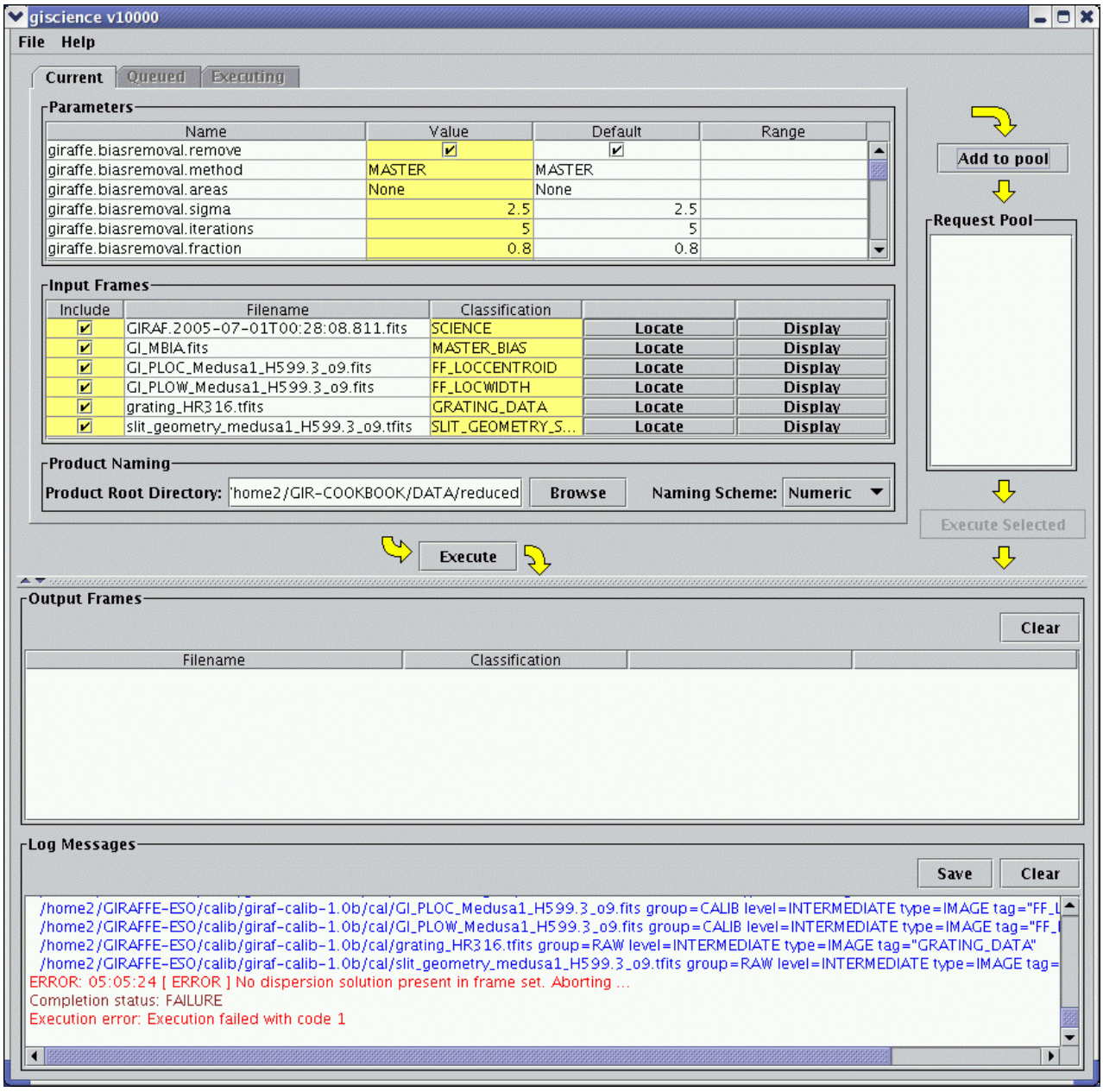

Figure 10: In the log sub-window of `giscience` we clearly see the reason for failure. In the example shown here, the file containing the dispersion solution is missing.

```
184dhcp133:reduced 31> dfits science_rbnspectra_0000.fits |\
fitsort OBS.TARG.NAME EXPTIME
FILE                            OBS.TARG.NAME           EXPTIME
science_rbnspectra_0000.fits    NGC6253_center_field    2699.9981
```

also the header to fits table can be accessed with dfits:

```
184dhcp133:reduced 32> dfits -x 1 science_rbnspectra_0000.fits | more
====> file science_rbnspectra_0000.fits (main) <====
===> xtension 1
XTENSION= 'BINTABLE' / FITS Binary Table Extension
BITPIX  =                    8 / 8-bits character format
NAXIS   =                    2 / Tables are 2-D char. array
NAXIS1  =                  103 / Bytes in row
NAXIS2  =                   84 / No. of rows in table
PCOUNT  =                    0 / Parameter count always 0
GCOUNT  =                    1 / Group count always 1
TFIELDS =                   14 / No. of col in table
TFORM1  = '1J      ' / Format of field
TTYPE1  = 'INDEX   ' / Field label
TUNIT1  = '        ' / Physical unit of field
TFORM2  = '1J      ' / Format of field
TTYPE2  = 'FPS     ' / Field label
.
.
.
```

The image itself is a 2D frame, with one of the axis being the dispersion direction and the other the object number. Therefore the size of the image can vary according to the number of allocated fibers.

In the example pyraf (Iraf module to python) is used but any other data manipulation package can be used (IRAF, IDL, Midas, fitsio inside C or Fortran programs, etc...).

For those using pyraf/iraf

load `onedspec` and then change the dispersion axis:

```
PyRAF 1.1 (2003Oct17) Copyright (c) 2002 AURA
Python 2.3.3 Copyright (c) 2001, 2002, 2003 Python Software Foundation.
Python/CL command line wrapper
  .help describes executive commands
--> onedspec
onedspec/:
 aidpars@       dopcor          reidentify      sensfunc        specplot
 autoidentify   fitprofs        rspectext       setairmass      specshift
 bplot          identify        sapertures      setjd           splot
 calibrate      lcalib          sarith          sfit            standard
 continuum      mkspec          sbands          sflip           telluric
 deredden       names           scombine        sinterp         wspectext
 dispcor        ndprep          scoords         skytweak
 disptrans      refspectra      scopy           slist
--> iraf.onedspec.dispaxis=2
```

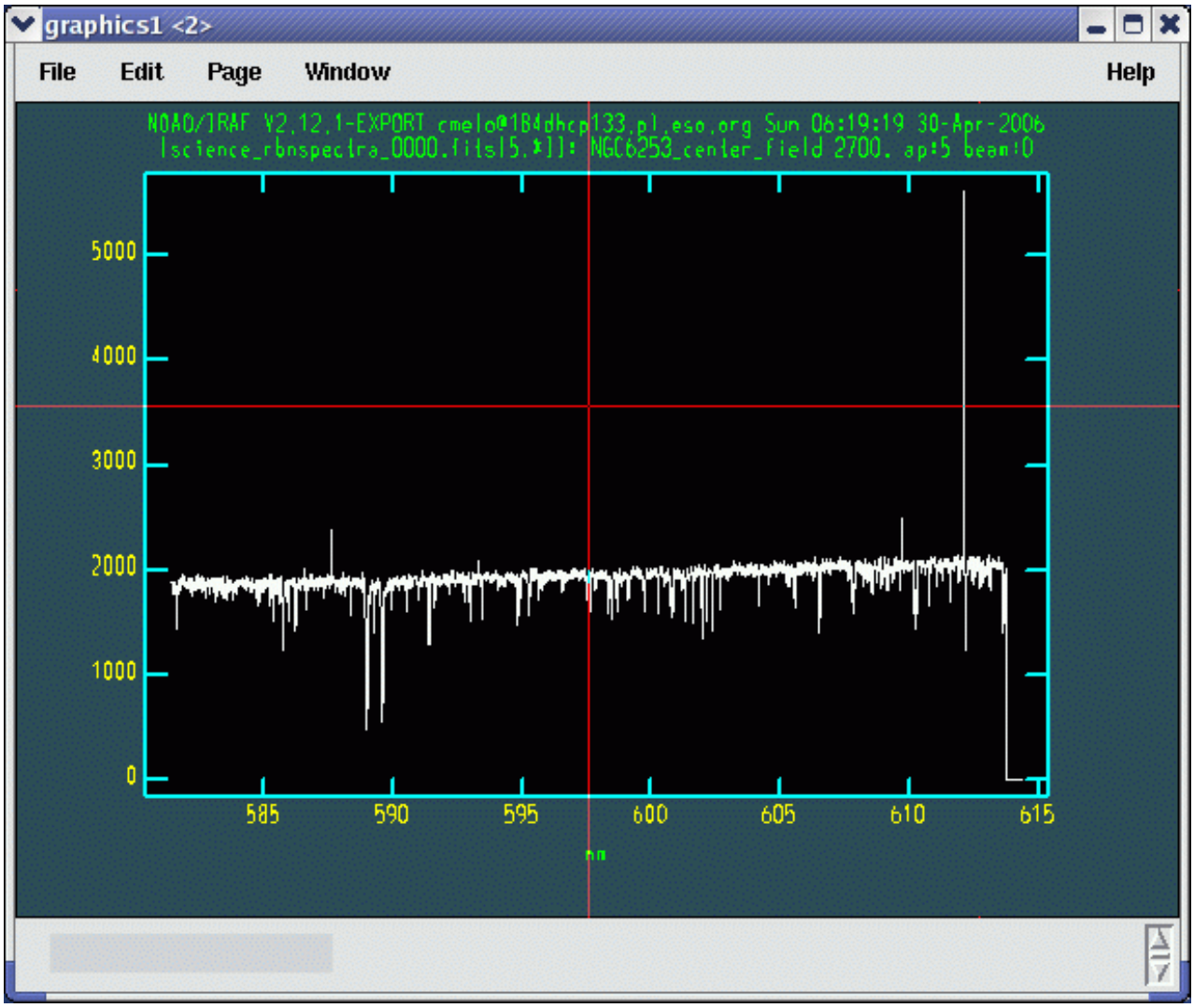

Figure 11: Stellar spectrum of a member of NGC6253 in aperture 5.

then plot:

```
--> splot science_rbnspectra_0000.fits
```

In the first fiber we see the ThAr spectra of the simultaneous calibration fiber of GIRAFFE. Moving to the other apertures in the image we recognize a stellar spectrum in aperture 5 as shown in Figure 11. But to which target am I looking at? The answer is found looking into the binary table. For this we use the command "dtfits" (not dfits as above!). So the column INDEX corresponds to the aperture seen in the image. Thus, in the example above, we were looking at the star ngc6253_mem4636. The column RP gives the GIRAFFE fibers allocated to the object. Therefore ngc6253_mem4636 was allocated to fiber# 18.

```
184dhcp133:reduced 38> dtfits science_rbnspectra_0000.fits | more
#
# file           science_rbnspectra_0000.fits
# extensions     1
# ----------------------------------------------
# XTENSION       1
# Number of columns 14
#
INDEX|FPS|SSN|PSSN| RP|  Retractor|FPD|              OBJECT|      R|   THETA|   ORIENT|     RA|     DEC|MAGNITUDE
    1|  1|  1|   1| -1|Calibration|  1|CALSIM              |      0|       0|        0|      0|       0|         0
    2|  2|  1|   2| 24|P1-MC1-12  |  2|ngc6253_candout_7708| 174590|0.561108|   3.6805| 254.74|-52.5992|    16.288
    3|  3|  1|   3| 22|P1-MC1-11  |  3|ngc6253_candout_7797| 153442|0.247286|  3.76381|254.709|-52.6178|    16.436
    4|  4|  1|   4| 20|P1-MC1-10  |  4|ngc6253_candout_7395| 108711|0.345902|  3.62651|254.734|-52.6339|    15.537
    5|  5|  1|   5| 18|P1-MC1-9   |  5|ngc6253_mem4636     |39245.9| 1.00779|  3.49414|254.778| -52.663|    14.846
    6|  6|  1|   6| 16|P1-MC1-8   |  6|ngc6253_candout_7370| 203416|0.304428|  3.54036|254.696|-52.5943|    15.478
    7|  7|  1|   7| 14|P1-MC1-7   |  7|ngc6253_cand_4493   |56346.2| 6.11965|  3.51313|254.735|-52.6657|    15.659
.
.
```

Note that the INDEX is the Aperture number that is shown when you splot the spectrum in IRAF. This will not always be the same as the FPS number due to missing fibres.

3.4 Case 2: Making your own calibration database

If you are only interested in a quick look of your data you can probably use the database delivered with the GIRAFFE KIT. However, for any other scientific application you must use your most recent calibrations. The position of the spectra on the CCD is a function of the ambient conditions (temperature and pressure). It also depends on the reproducibility of the grating which moves according to the set-up chosen. Thus the use of fresh calibrations ensures that the pipeline will extract your data on the right place.

In addition, a better wavelength calibration is achieved since little shifts (below 1 pixel level) are expected to take place within the time gap between your science frame and your calibration (probably a few hours and no more than 1 day). Also, your slit geometry determination is updated.

Even in the case we want to rebuild your calibration database, a few static files are still needed. Thus the best way to organize these static files is to create a directory called `static` to place these files. In the example the data is organized as follow:

```
limari:DATA 78> ls -1
raw
reduced
static
```

- **The raw data**. For each raw science frame, a set of 5 biases, 3 flat-fields and 1 arc frame are produced as part of the FLAMES/GIRAFFE calibration plan. In the case of ARGUS and IFU, a flux standard is also provided. For ARGUS flat-fields, a nasmyth screen is used instead the robot flat. These screen flats provide by far a more uniform illumination and a better correction of the fiber-to-fiber variations.

```
limari:raw 67> dfits *.fits | fitsort dpr.type
FILE                                    DPR.TYPE
GIRAF.2005-07-01T00:28:08.811.fits      OBJECT,SimCal
GIRAF.2005-07-01T14:16:54.585.fits      LAMP,FLAT
GIRAF.2005-07-01T14:18:34.303.fits      LAMP,FLAT
GIRAF.2005-07-01T14:20:12.871.fits      LAMP,FLAT
GIRAF.2005-07-01T14:22:34.861.fits      LAMP,WAVE
GIRAF.2005-07-01T15:00:37.382.fits      BIAS
GIRAF.2005-07-01T15:01:24.886.fits      BIAS
GIRAF.2005-07-01T15:02:12.420.fits      BIAS
GIRAF.2005-07-01T15:03:02.844.fits      BIAS
GIRAF.2005-07-01T15:03:50.308.fits      BIAS
```

- **The static data**. The static data are fits table containing information about the physical model of GIRAFFE gratings, a catalogue of ThAr lines and the slit geometry table. Whereas the two first tables are really static, the slit geometry does change although in a very long time-scale (months). Looking at the rebinned arc spectrum produced by the wavelength calibration recipe will give you an idea if you have to remake it. Sections 3.4.4 and 8 describe how to make this slit geometry table in gasgano and ESOrex.

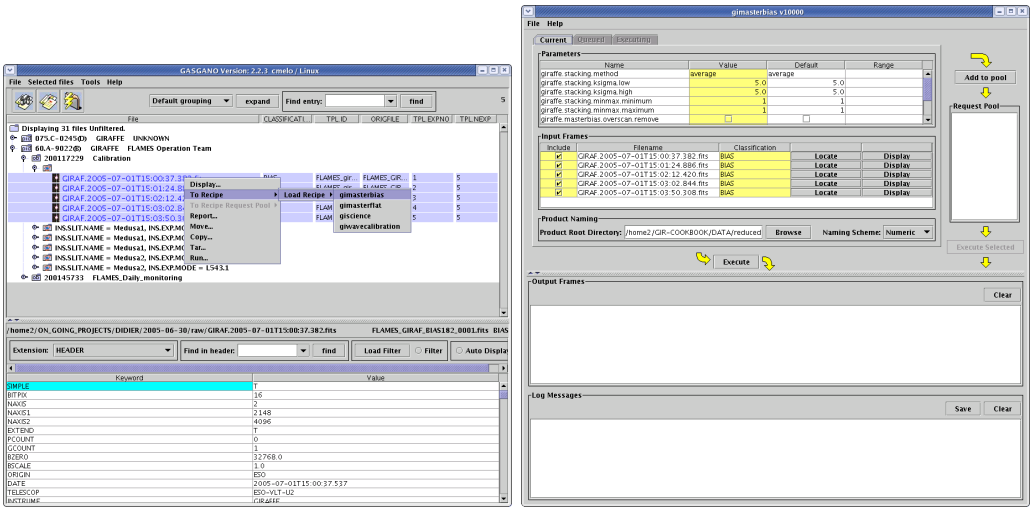

Figure 12: Master BIAS reduction.

```
limari:static 71> dfits *.*fits | fitsort pro.catg
FILE                              PRO.CATG
grating_HR316.tfits               GRATING_DATA
grating_LR600.tfits               GRATING_DATA
line_catalog_ThAr.tfits           LINE_CATALOG
slit_geometry_medusa1.tfits       SLIT_GEOMETRY_MASTER
```

Finally, before we start, keep in mind that ESO pipelines are in general QC oriented pipelines. This means that the quality of your data reduction can be assessed by looking at the QC KW added to the image header by the pipeline. A list of these is available in the GIRAFFE pipeline manual.

3.4.1 gimasterbias

Until July 7th 2008 there was a 'history effect' for biases that meant that the first bias in the sequence of 5 had lower flux than the remaining four. The effect was upto 2 ADU for the new CCD Carreras and perhaps 0.3 ADU for the old CCD Bruce. See:

```
www.eso.org/sci/facilities/paranal/instruments/flames/inst/features
                                        /FLAMES_GIRAFFE_Bias_History_Effect.html
```

This history effect when present can cause the master bias to be too high compared with the science data and hence cause over-subtraction of the bias level. To avoid this one can use the 'PROFILE' option of bias removal in the science reduction and not the master bias. After July 7th 2008 the levels in all biases should be the same.

The above noted, one should still run the `gimasterbias` command to produce the bad-pixel map. If you decide to use the master bias for the science reduction then be sure to compare the bias-subtracted science frame.

To produce the master bias and bad pixel map, select all BIAS FRAMES and pass them to the recipe gimasterbias as shown in the left panel of Fig 12. A new window appears (right panel of Fig 12) where all parameters related to the recipe `gimasterbias` can be controlled.

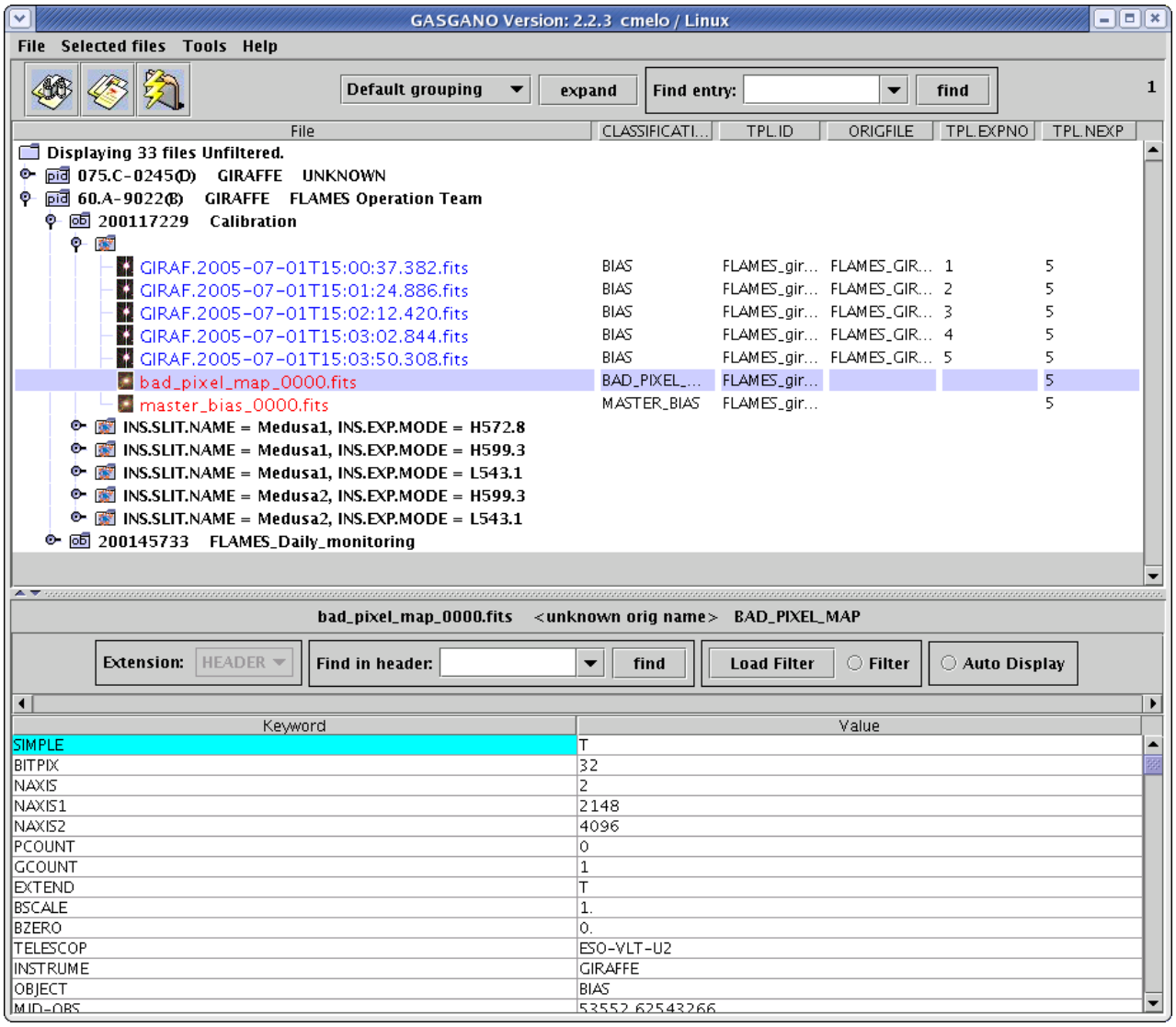

Figure 13: Gasgano automatically updates the list of files. The reduced files created by `gimasterbias` are seen in the gasgano file list.

For a full description of the parameters of each recipe please refer to the pipeline manual. In this new window, change the directory where the pipeline products are going to be placed and and add it to the gasgano list. A similar window exist for all recipes. There you have full control of the recipe parameters. You can also change the input list and the output. The log sub-window at the bottom of the main window allows you to follow what is going on. A copy of the log messages is dumped on the disk.

When you are happy with the parameters hit `Execute`. The products (`master_bias_0000.fits` and `bad_pixel_map_000.fits`) now appear automatically in gasgano as shown in Fig 13.

Note that the default stacking method is 'average' which generally gives a lower noise on the combined master bias than 'median', at the expense of possible problems with bad pixels or cosmic rays.

QC parameters produced by `gimasterbias` are logged in `qc_000N.paf`, an example is shown below:

```
[jsmoker@184dhcp22 Smartt_Gasgano]$ more qc0000.paf
```

```
PAF.HDR.START;
PAF.TYPE            QC1 parameters;          # Type of parameter file
PAF.ID              ;
PAF.NAME            qc0000.paf;              # Name of PAF
PAF.DESC            ;                        # Short description of PAF
PAF.CRTE.NAME       jsmoker;                 # Name of creator
PAF.CRTE.DAYTIM     2009-03-14T20:38:11.000; # Civil time for creation
PAF.LCHG.NAME       ;                        # Author of par. file
PAF.LCHG.DAYTIM     ;                        # Timestamp for last change
PAF.CHCK.NAME       ;                        # Name of appl. checking
PAF.CHCK.DAYTIM     ;                        # Time for checking
PAF.CHCK.CHECKSUM   ;                        # Checksum for the PAF
PAF.HDR.END;
#------------------------------------------------------------------------------
QC.DID              "ESO-VLT-DIC.GIRAFFE_QC-0.2"; # QC1 dictionary
ARCFILE             "GIRAF.2003-04-15T00:47:11.122.fits"; # Archive File Name
TPL.ID              "FLAMES_giraf_cal_bias"; # Template signature ID
PRO.CATG            "MASTER_BIAS";           # Pipeline product category
PRO.DATAAVG         158.56187685038;         # Mean of pixel values
PRO.DATARMS         29.4017477751589;        # Standard deviation of pixel values
PRO.DATAMED         158.2;                   # Median of pixel values
PRO.DATANCOM        5;
QC.BIAS.MASTER.MEDIAN 158.199996948242;      # Median master bias level (ADU)
QC.BIAS.MASTER.MEAN 158.26553557018;         # Mean master bias level (ADU)
QC.BIAS.MASTER.RMS  2.06290380948411;        # RMS of master bias level (ADU)
QC.OUT1.RON.RAW     1.84526219570382;        # Readout noise (raw)
QC.OUT1.RON.MASTER  0.834264929284362;       # Readout noise (master)
QC.OUT1.STRUCT.X    0.0620529817523003;      # Structure along the x axis
QC.OUT1.STRUCT.Y    0.517048258124353;       # Structure along the y axis
```

3.4.2 gimasterflat

In order to reduce the FF we need two static tables. In the case of the recipe `gimasterflat`, the slit geometry (`CLASSIFICATION = SLIT_GEOMETRY_MASTER`, make sure to chose the one corresponding to the plate used for the science data you want to reduce) and the grating data (`GRATING_DATA`, here also you have to chose the right one, in our example the data have been taken with the LR grating.

It's likely that you have to adjust the number of fibers to be found. By default the recipe tries to find 136, but in practice we fit 135 on the chip and in addition there are always broken fibers. Note that it is best to do this by explicitly defining the fibres using `giraffe.fibers.spectra` (see below).

As a first pass we run the recipe with the default number of fibres and by removing the bias using the overscan region (`giraffe.biasremoval.method='PROFILE'`).

In Fig.14 we see that the recipe failed because *only* 134 were found instead of 136 fibres.

The best way to get things working is to display the flatfield and see what fibre or fibres are missing. Fig. 15 shows an example in which Fibre 27 (counting left to right for Medusa, about a quarter of the way from the left) has no flux. Fixing `giraffe.fibers.spectra` in

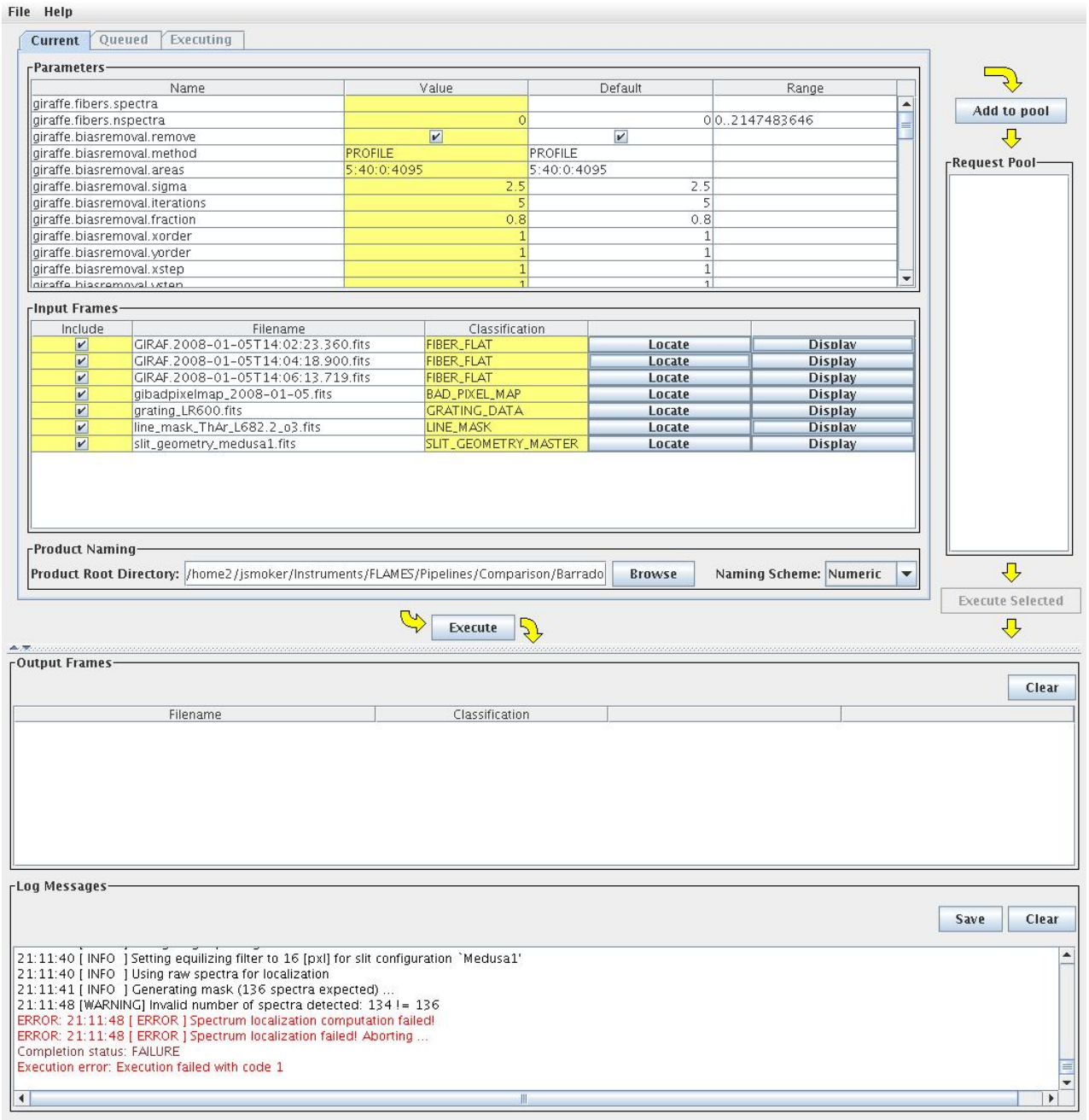

Figure 14: Recipe `gimasterflat` in action. The recipe crashed due to the fact that the specified number of fibers was not found by the pipeline (see Log Message window in the bottom of the panel).

the Parameter sub-window `gimasterflat to '1-26,28-135'` ignores fibre 27 therefore runs fine (see Fig. 16) and produces a number of tables which are necessary to the extraction of the ThAr and the science spectra. The FF spectra for each fiber is already extracted although still in the pixel space (Fig. 17).

`giraffe.fibers.nspectra` should be used with care since it selects the `nfibers` from the left to right. Consider the following example. In the middle of the night a fiber just broken and had to be disabled. In this case, even though we can tell the pipeline to look for `nfibers-1` (accounting for the broken fiber), the localization process will fail because it cannot find the flat-field signal where it was supposed to be. When this happens one has to explicitly tell the pipeline which fibers are **enabled**. For this we use the parameters `giraffe.fibers.spectra` as described above which gives the pipeline the list of enabled fibers. For more details, please refer also to Sec. 9.2.3 of the pipeline user manual.

After creating your master flat you may want to make sure that the fibre localisation has worked. In order to do this it is necessary to look at the product `ff_loccentroid.fits` as shown in Fig. 18 and compare with the master flatfield. In the product ff_loccentroid.fits the 'x' coordinate responds to the fibre number, the 'y' coordinate to the 'y' coordinate on the detector, and the 'value' to the 'x' coordinate of the fitted localisation of the fibre. If we move the mouse to position (10,2000) in ff_loccentroid.fits, say, we see the 'value' of '160.62'. Now if we display `master_fiber_flat.fits` and move the mouse to the position of fibre 10 (counting left to right for Medusa), then we see that the centre of this fibre is x=161.67, an offset of about 1 pixel. Note that this is just due to the difference in counting convention in the two products which always show this offset. In this case this means that the localisation centroid ff_loccentroid for the master_fiber_flat are identical to within 0.05 pixels. **Shifts of more than a pixel should be investigated as they may indicate problems with the data reduction.**

3.4.3 giwavecalibration

The method used by the GIRAFFE pipeline is based on a simple optical model of the spectrograph. Given the position of the fibers in the focal plane (which is what is usually referred to as slit geometry), and the wavelength (of an arc-lamp line) the model predicts the position of this line on the CCD.

The line is searched around this initial position, and a PSF profile (not a Gaussian) is fitted to the detected peak to get the centroid position.

Having determined the line positions for every line for every fiber, the optical model is fitted to this data, using the slit offset and slit rotation angle in the focal plane as free parameters.

The model is accurate to about one pixel, and degrades towards the CCD edges. To compensate for that the residuals of the measured line positions with respect to the predicted positions is modeled by a 2D Chebyshev polynomial, which is used as a corrective term when re-binning the spectra[7].

The fitted optical model is described by FITS keywords in the header of the DISPERSION_SOLUTION product of the pipeline, while the coefficients of the polynomial are stored in the FITS table.

Another correction term is added during re-binning by correcting for residual wavelength shifts computed from the simultaneous calibration fibers. This is done in the science frame when

[7]For more details about the wavelength calibration process, we refer to Royer et al. (2002) which is available at http://www.eso.org/instruments/flames/doc/spie_royer.ps.gz

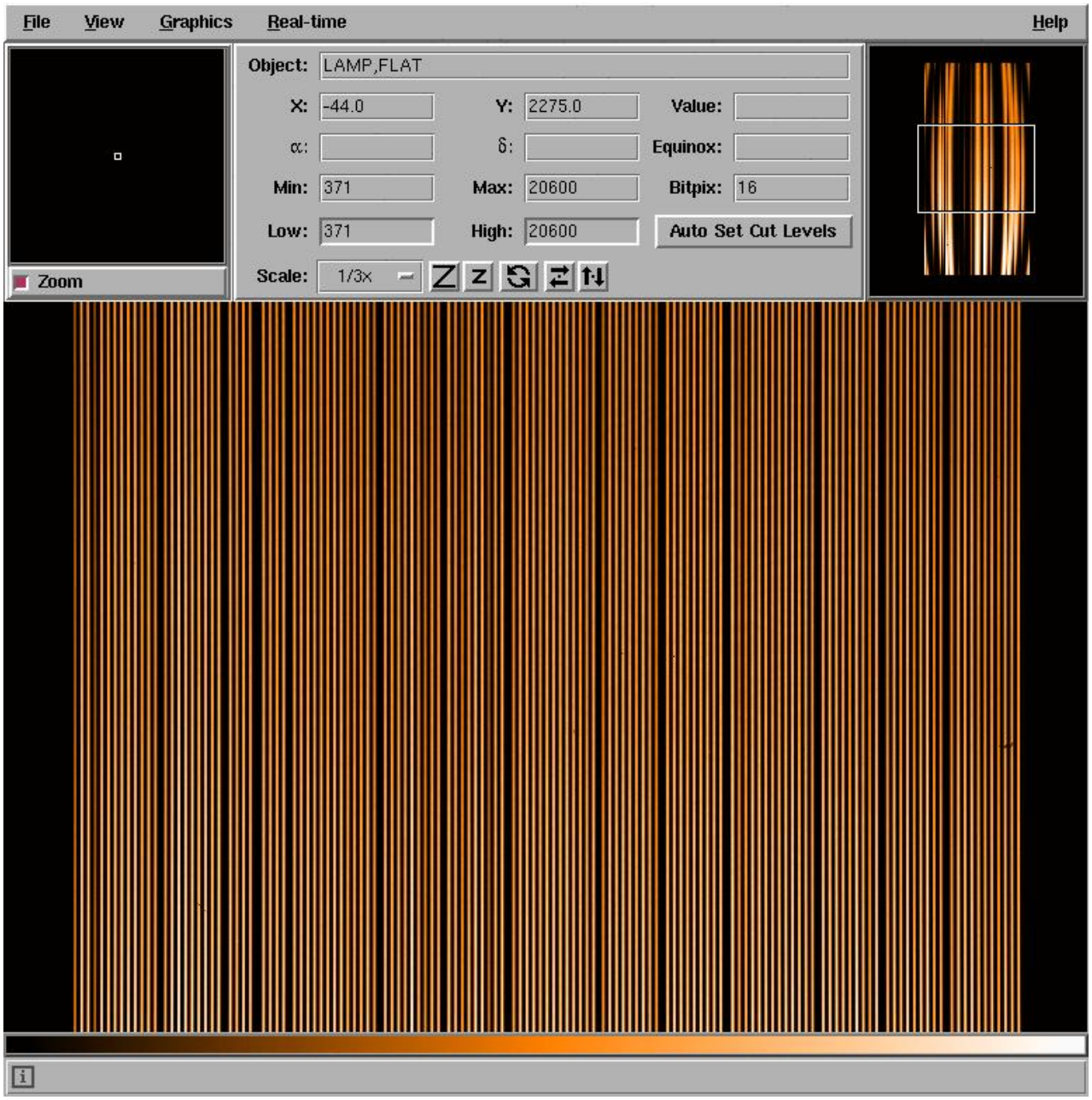

Figure 15: A Medusa1 flatfield image in which fibre 27 (about a quarter from the left) is missing. Note that for ARGUS the counting is done from right to left and not from left to right as in Medusa or IFU.

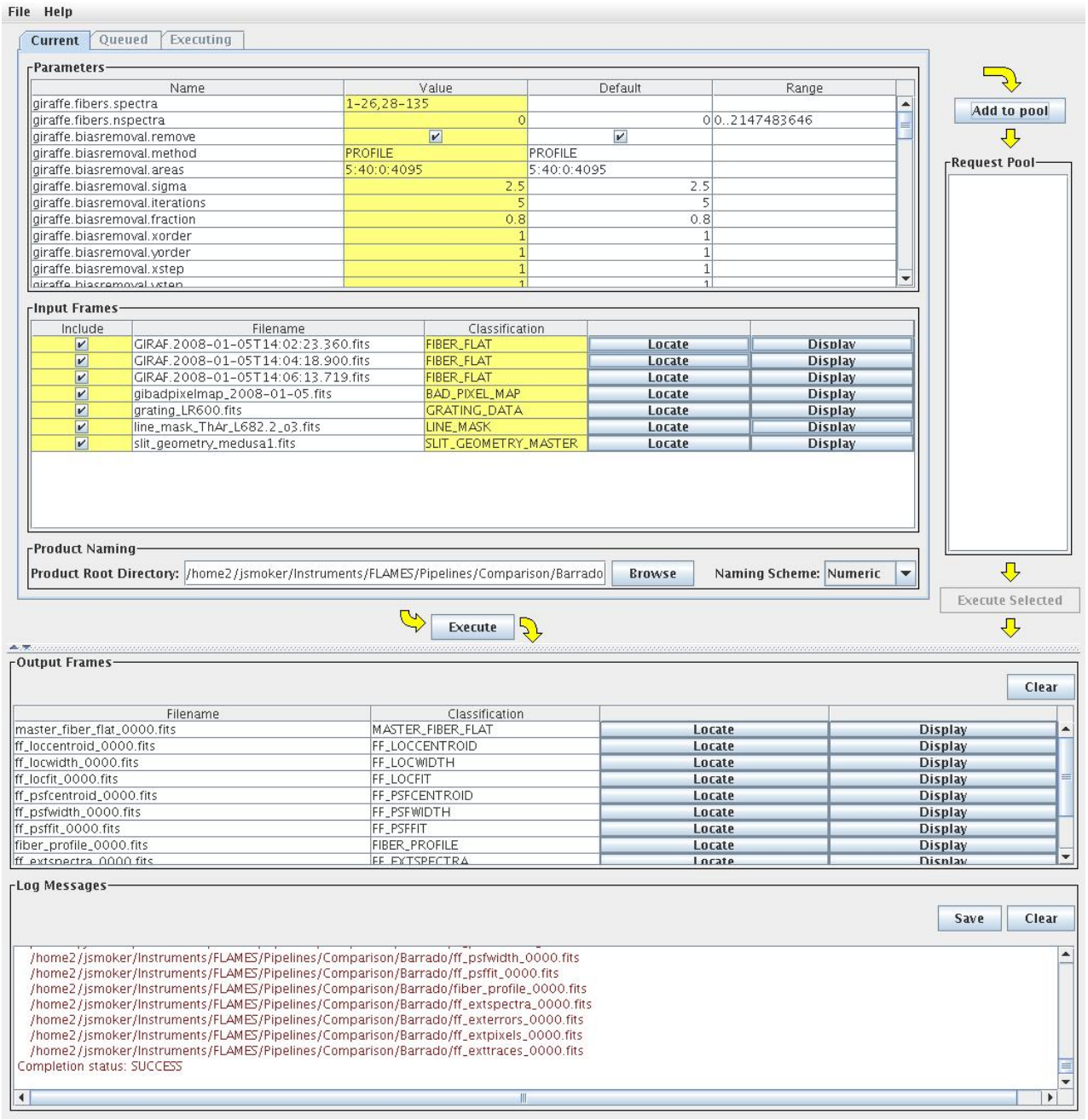

Figure 16: Recipe `gimasterflat` in action. The recipe runs fine as we have told it which fibres in the flat have signal by changing the parameter giraffe.fibers.spectra to '1-26,28-135'.

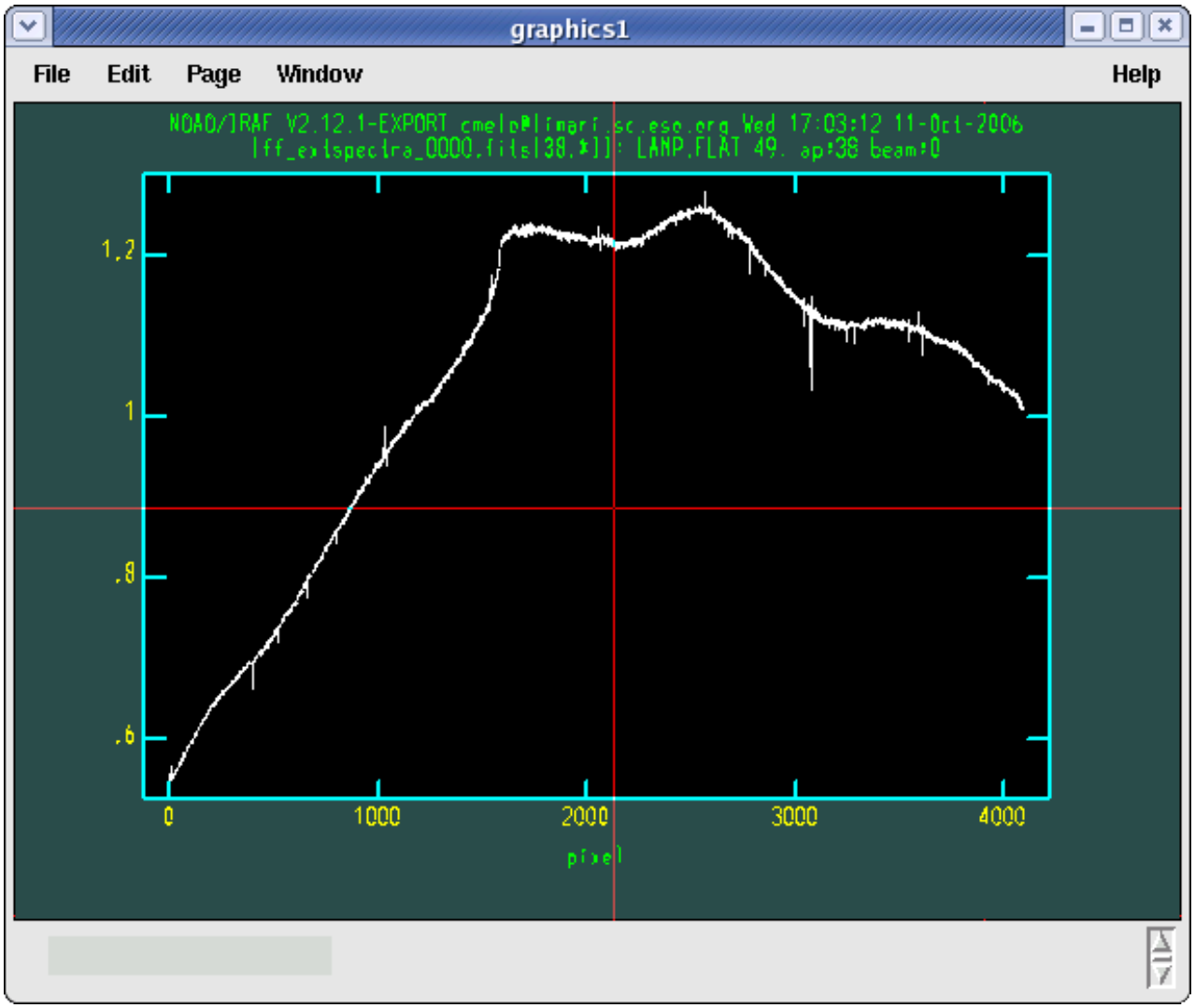

Figure 17: Extracted flat-field spectrum for one the fibers. The spectrum was produced by `gimasterflat` and will be wavelength calibrated by the next step carried out by `giwavecalibration`.

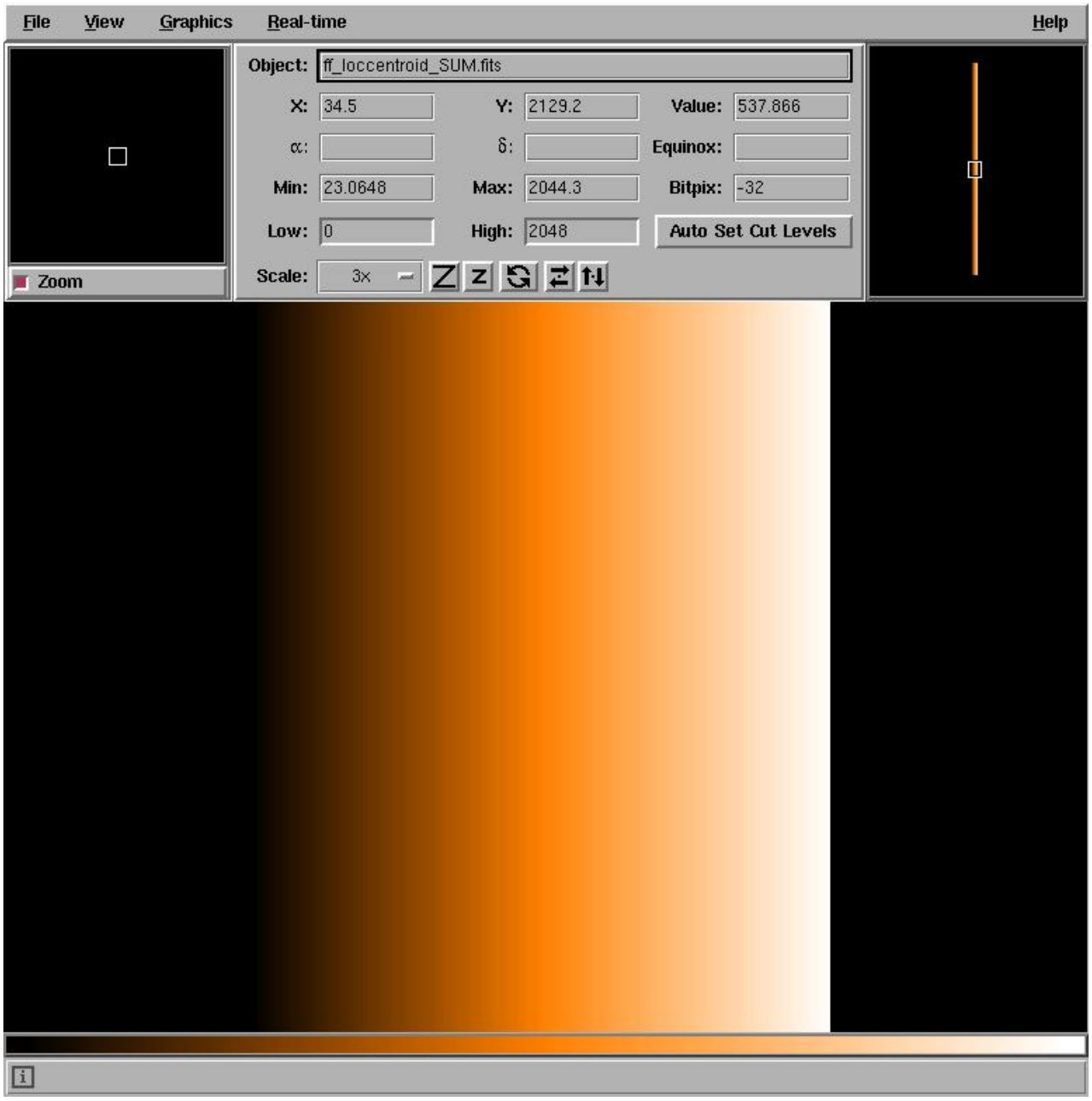

Figure 18: Localisation centroid frame ff_loccentroid.fits produced by gimasterflat. The 'x' coordinate is fibre number, the 'y' coordinate the 'y' position on the CCD and the 'value' at each point the 'x' coordinate of the localisation centroid. For a perfect fit there is an offset of 1 pixel between the 'value' in ff_loccentroid.fits and the 'x' centre within each fibre in master_fiber_flat.fits (i.e. 'value' for ff_loccentroid.fits for fibre x=N is 'x' value for master_fiber_flatfits - 1)

the simultaneous calibration fibre is used. The amount corrected is given in the table and the user can undo the correction.

After you have run giwavecalibration you should check the products to see that they look sensible. In particular, the file `arc_rbnspectra.fits` file should be displayed. The results of our first run are shown in (Fig. 19).

Note that in this figure there are big jumps in the 'y' direction between the fibres. If you see shifts like these in your data, then a good idea would be to **remake the slit geometry table**.

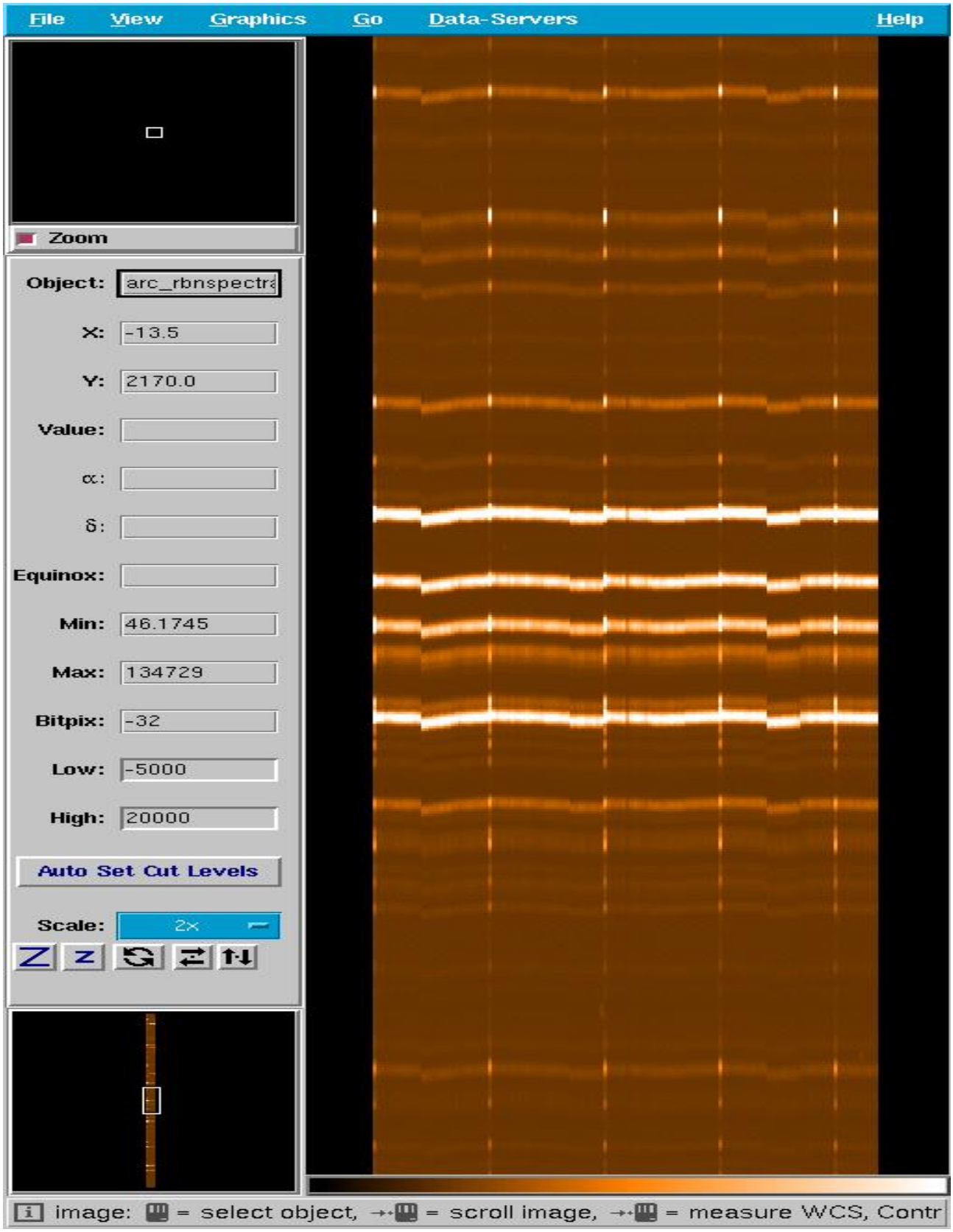

Figure 19: Original rebinned arc-line spectrum arc_rbnspectra.fits created by giwavecalibration. Jumps in the 'y' position signify that this is not a good wavelength solution and should be improved by creating a new slit geometry table.

3.4.4 giwavecalibration - remaking the slit geometry table

If you see jumps in your `arc_rbnspectra.fits` file, it is probably wise to remake the slit geometry table by re-running `giwavecalibration`. To do this you have to enable the `giraffe.wcal.slitgeometry` flag within `giwavecalibration` and re-run it. Typical inputs using gasgano are shown in Fig. 20).

On the first pass of giwavecalibration you may choose to run 8 iterations with a box of 20 pixels ("20,20,20,20,20,20,20,20") and in the 2nd go you use "10,10,10,10,10" (5 times).

After `giwavecalibration` has been run again, you should look anew at the `arc_rbnspectra.fits` file to see if there has been any improvement. An example is shown in (Fig. 21). The jumps in 'y' have now disappeared so the wavelength solution appears to be good. This can be checked by looking at the radial velocity error RVERR in the slit geometry table as shown in (Fig. 22). In the example shown the errors are typically 0.003 $\mathrm{km\,s^{-1}}$ (or 3 $\mathrm{m\,s^{-1}}$). We note that the radial velocity accuracy of your science targets will be much worse than this, the best achieved velocity accuracy to date being around 30 $\mathrm{m\,s^{-1}}$ over a period of 5 days (Loeillet et al. A&A, 479, 865, 2008).

3.4.5 giscience

The recipe `giscience` can now be executed using the files produced by `gimasterbias`, `gimasterflat` and `giwavecalibration`. In the present version of the pipeline, the extraction performed by `giscience` adds up the signal inside the PSF fitted by `gimasterflat`. Optimum (weighted) extraction is now available that also does the background (not sky) subtraction too.

`giscience` also flat-field the data and corrects for the fiber-to-fiber transmission difference using the information produced by `gimasterflat`. Flat-field and transmission corrections can be controlled by the input parameters defined by the user. Please, refer to the user manual for more details. `giscience` produces also an error spectrum which is the standard deviation of the re-sampled fluxes for each wavelength bin.

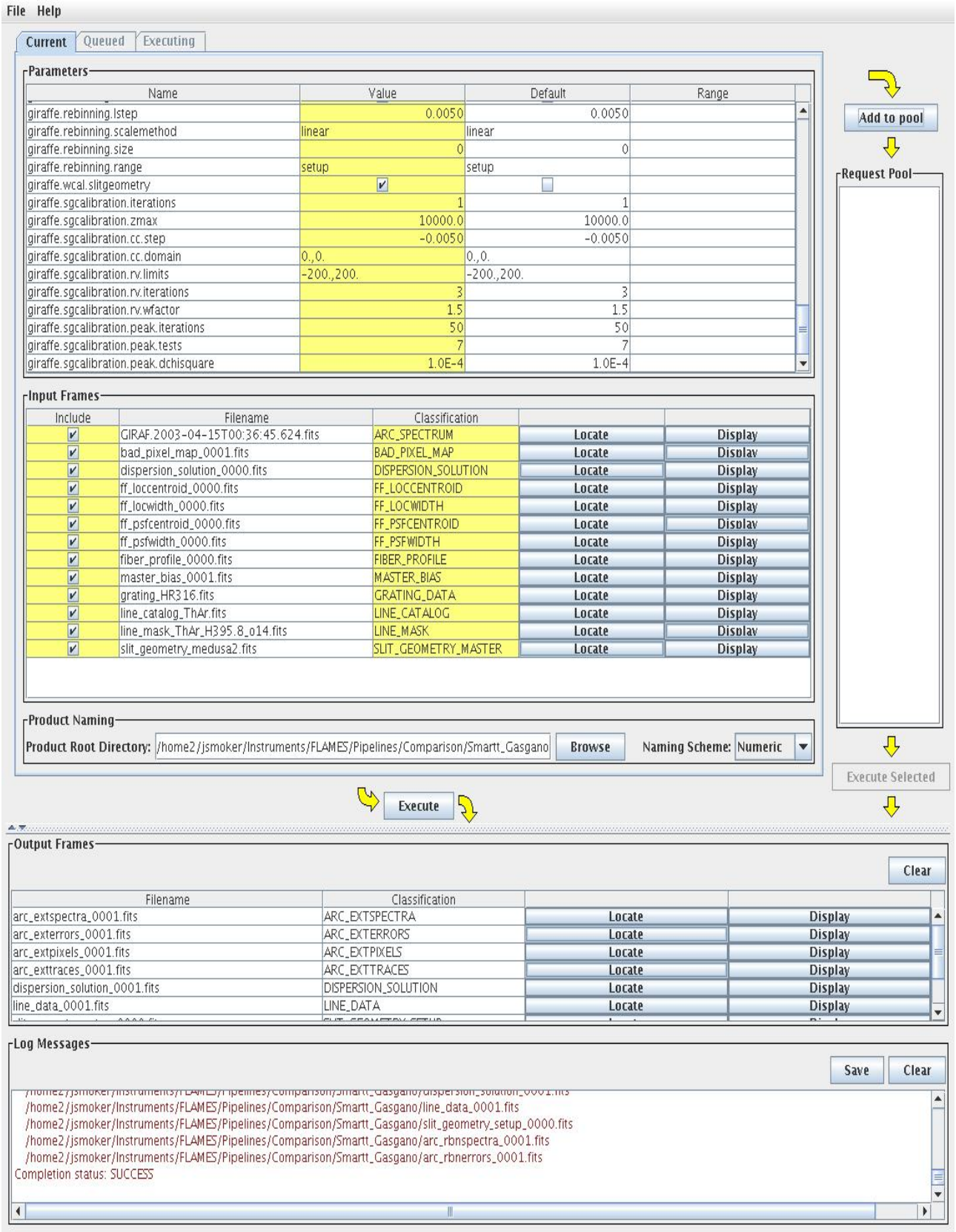

Figure 20: Inputs to `giwavecalibration` using gasgano in order to create a new slit geometry table.

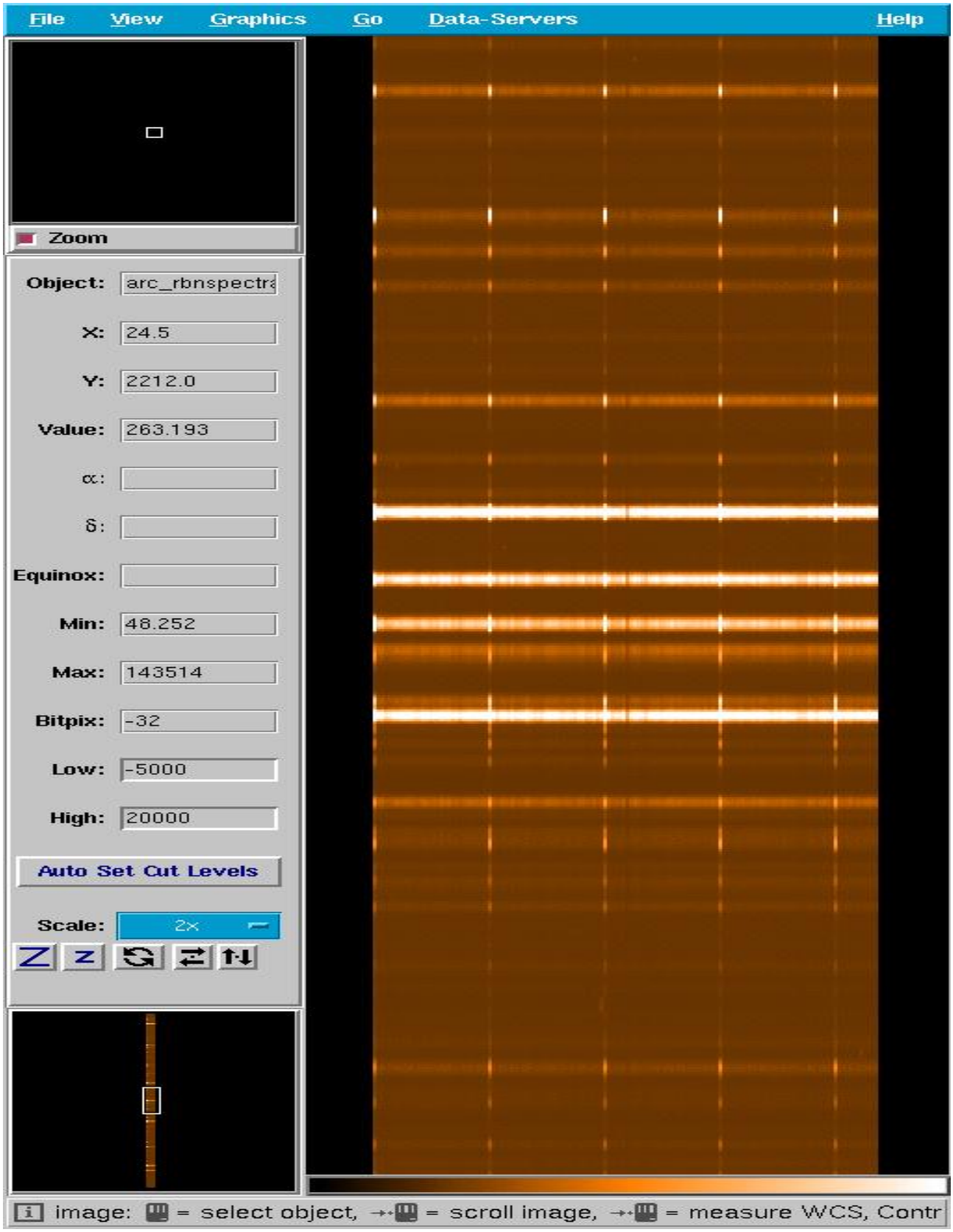

Figure 21: Rebinned arc-line spectrum arc_rbnspectra.fits created by `giwavecalibration` after making a new slit geometry table. The jumps in 'y' are much less pronounced than in Fig 19.

File Edit Options Data-Servers

Search Results (136)

INDEX	FPS	SSN	XF	YF	ZF	ZDEFOCUS	RV	RVERR	RESOLUTION
1	1	1	93.1782	37.3398	0.6937	-0.1782	0.231945	0.00323477	14984.8
2	2	1	93.184	36.8198	0.7001	-0.1477	0.171355	0.00342404	16472.6
3	3	1	93.1825	36.2998	0.7065	-0.1174	0.182567	0.00340234	16274.1
4	4	1	93.1839	35.7799	0.7129	-0.0876	0.147593	0.00358495	16634.6
5	5	1	93.1835	35.2599	0.7194	-0.058	0.176119	0.00354971	16500.9
6	6	1	93.1902	34.74	0.7258	-0.0289	0.16136	0.00347851	16357.1
7	7	1	93.1962	34.22	0.7322	0	0.154024	0.00335576	16882
8	8	1	93.1976	33.7	0.7386	0.0285	0.159423	0.00347449	16200.7
9	9	1	93.1985	33.1801	0.7451	0.0567	0.137595	0.00343178	16138.3
10	10	1	93.1995	32.6601	0.7515	0.0845	0.128764	0.00343981	16053.6
11	11	1	93.1984	32.1402	0.7579	0.112	0.143417	0.00348608	15964.5
12	12	1	93.1965	31.6202	0.7643	0.1392	0.159096	0.00355957	16134.7
13	13	1	93.2076	31.1002	0.7707	0.166	0.0835813	0.00356177	16183.5
14	14	2	93.3142	30.2499	0.475	-0.0971	-0.254631	0.00339826	16405.8
15	15	2	93.3028	29.7299	0.4803	-0.0723	-0.25714	0.0034156	16460
16	16	2	93.2904	29.2099	0.4855	-0.0479	-0.194354	0.0035145	16337.2
17	17	2	93.2805	28.69	0.4908	-0.0238	-0.158393	0.00347458	16283.6
18	18	2	93.2729	28.17	0.4961	0	-0.14358	0.00347986	16352.7
19	19	2	93.2623	27.65	0.5014	0.0234	-0.0466289	0.00338651	16075
20	20	2	93.254	27.1301	0.5067	0.0465	-0.0954385	0.00346265	16109.5
21	21	2	93.2455	26.6101	0.512	0.0693	-0.016709	0.00342234	16061
22	22	2	93.2309	26.0901	0.5173	0.0917	0.037551	0.00357151	16040.4
23	23	3	93.2135	25.2399	0.3179	-0.0803	0.0624736	0.00347723	16380.9
24	24	3	93.2039	24.7199	0.3223	-0.0597	0.153443	0.00352105	16818.6
25	25	3	93.2041	24.2	0.3266	-0.0395	0.107512	0.00348652	16381.4
26	26	3	93.1995	23.68	0.331	-0.0196	0.163205	0.00342611	16267.4
27	27	3	93.1971	23.16	0.3353	0	0.138406	0.00347739	15987
28	28	3	93.1884	22.64	0.3397	0.0192	0.185514	0.00351302	16054.6
29	29	3	93.1862	22.12	0.344	0.0381	0.195105	0.00347557	16198.4
30	30	3	93.1859	21.6001	0.3484	0.0567	0.186927	0.00346506	16050.1
31	31	3	93.1743	21.0801	0.3527	0.0749	0.223937	0.00346037	16116.3

Search Plot Filter More Info Preview Stop Close

Figure 22: New slit geometry table created by `giwavecalibration`. The errors in Radial Velocity are in $\mathrm{km\,s^{-1}}$. Note that errors in the wavelength calibration for the science will be larger than this due to S/N constraints or because of instrumental shifts between science and calibration images.

4 IFU and Argus image reconstruction

In the case of 3D spectroscopic observations with IFU or Argus, a data cube containing the spatial information for each wavelength bin is generated. An error cube is also generated as shown below.

```
-rw-r--r--    1 cmelo    astro        6711 May  8 08:33 giscience_2007-05-05_17:09:52.log
-rw-r--r--    1 cmelo    astro     5166720 May  8 08:34 science_exterrors_0000.fits
-rw-r--r--    1 cmelo    astro     5166720 May  8 08:34 science_extpixels_0000.fits
-rw-r--r--    1 cmelo    astro     5166720 May  8 08:34 science_exttraces_0000.fits
-rw-r--r--    1 cmelo    astro     9524160 May  8 08:34 science_rbnspectra_0000.fits
-rw-r--r--    1 cmelo    astro       34560 May  8 08:34 science_rcspectra_0000.fits
-rw-r--r--    1 cmelo    astro       34560 May  8 08:34 science_rcerrors_0000.fits
-rw-r--r--    1 cmelo    astro     9524160 May  8 08:34 science_rbnerrors_0000.fits
-rw-r--r--    1 cmelo    astro    33583680 May  8 08:34 science_reduced_0000.fits
-rw-r--r--    1 cmelo    astro     5166720 May  8 08:34 science_extspectra_0000.fits
-rw-r--r--    1 cmelo    astro     8939520 May  8 08:34 science_cube_spectra_0000.fits
-rw-r--r--    1 cmelo    astro     8939520 May  8 08:34 science_cube_errors_0000.fits
```

At the moment, there is no dedicated tool for GIRAFFE data cubes. A nice one which was developed for SINFONI is `QfitsView` written by Thomas Ott and which comes with `scisoft` released by ESO. You can also install it yourself by downloading it from:

http://www.mpe.mpg.de/ ott/QFitsView/

It has many nice functionalities to analyze and visualise your data. The new version of QFitsView will read Argus cubes straight out of the box without any need for changes to the GIRAFFE headers. You need to look for a "HIERARCH ESO PRO CATG" of "SCIENCE_CUBE_SPECTRA" which will be something like: r.GIRAF.2009-02-14T02:39:37.821_0009.fits Note that at present that cubes are only produced for ARGUS but a future version will also make them for the IFU fibres.

An alternative is `gaia` which is available at:

http://star-www.dur.ac.uk/ pdraper/gaia/gaia.html

Now `science_cube_spectra_0000`fits can be read by QFitsView as shown in Fig. 23.

The IFUs on the two plates have somewhat different responses. Fig 24 shows a sky spectrum taken on plate 1 and and on plate 2.

Fig. 25 shows an example of a raw GIRAFFE image in which 15 stars were observed using IFU mode with Fig. 26 displaying the reconstructed image of the same field that is produced by the pipeline.

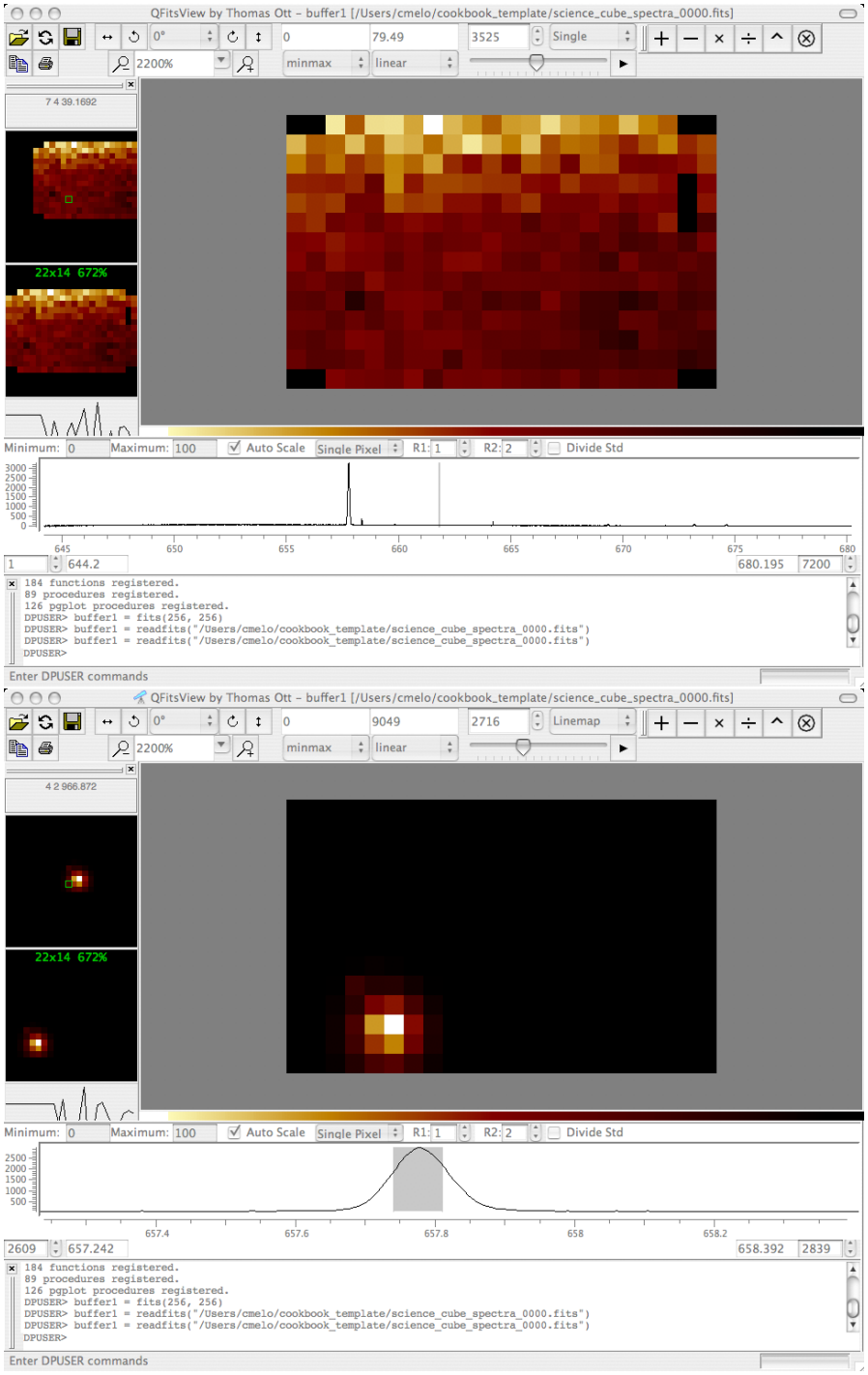

Figure 23: Argus cube of an emission line object produced by the pipeline and visualized by QFitsView. The upper panel show the flat image, i.e, the whole wavelength range is considered. Nothing is really seen with respect to the background.In the lower panel, one the core of the emission is display. The object pops-up with respect to the background.

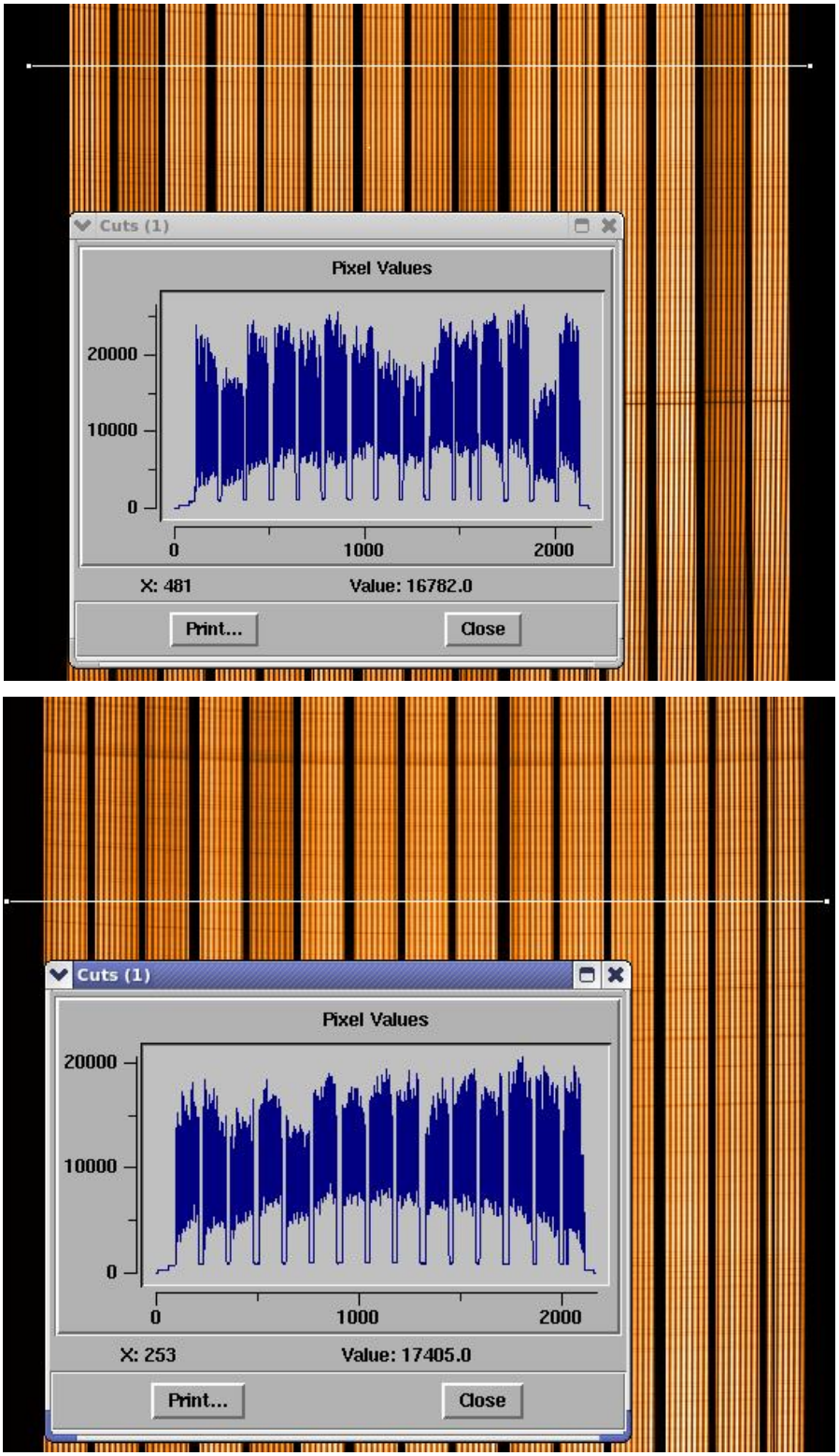

Figure 24: Raw GIRAFFE IFU images of the solar spectrum on plate 1 (top) and plate 2 (bottom). Variations in the IFU responses are clear.

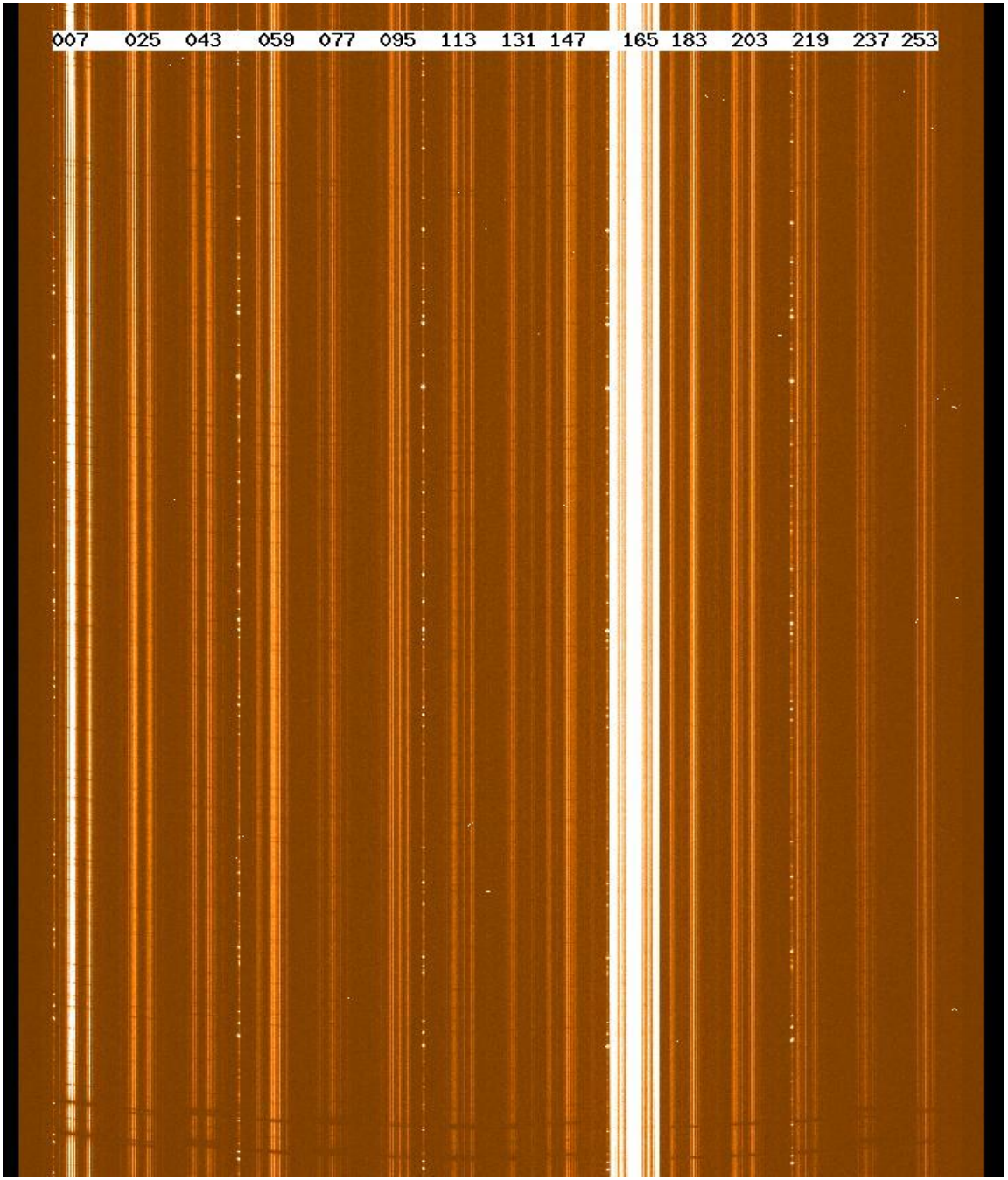

Figure 25: Raw GIRAFFE image of a field in which 15 IFU units were deployed. The reconstructed image is shown in Fig. 26.

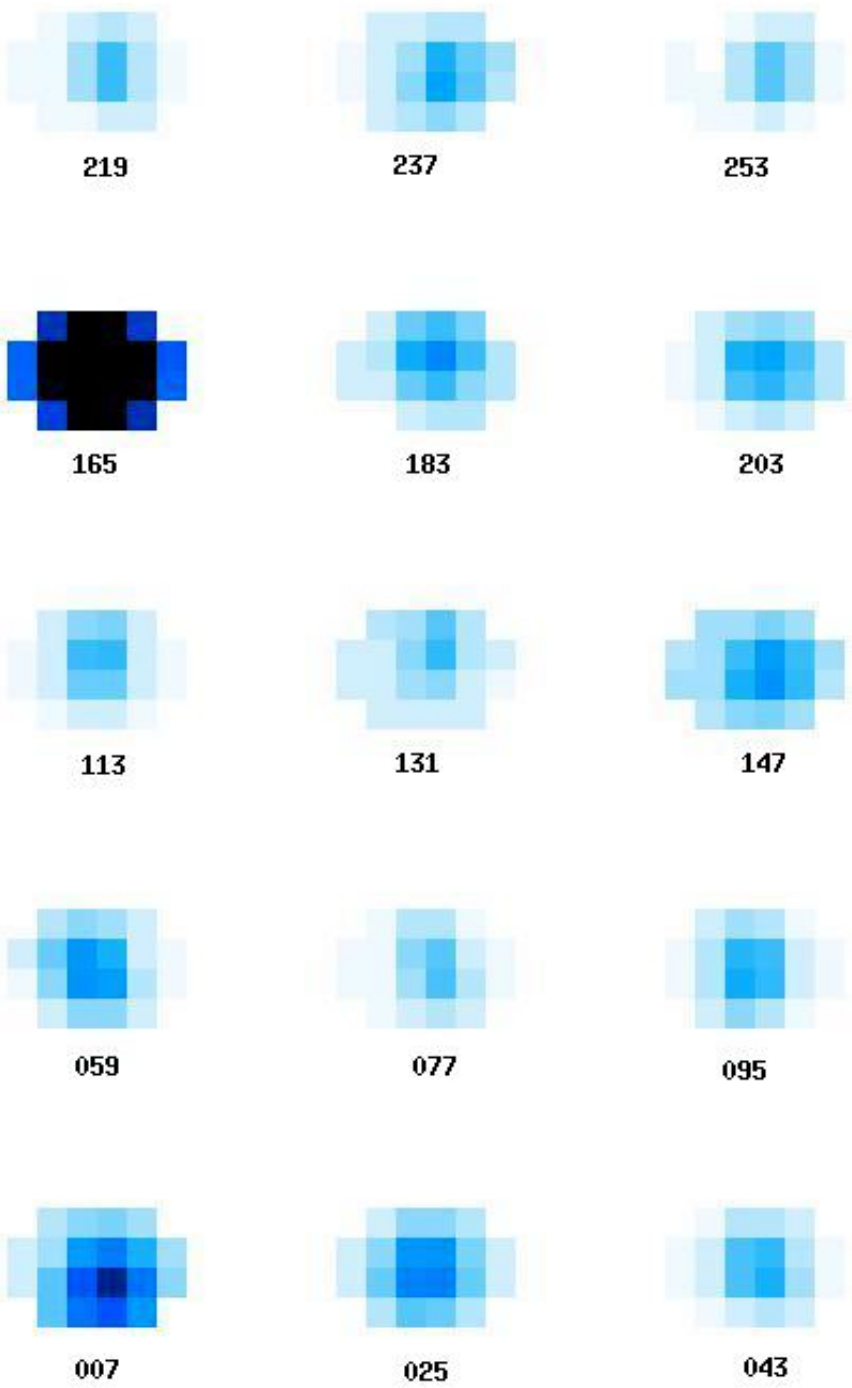

Figure 26: Reconstructed image of 15 IFU units produced by the pipeline. The raw image is shown in Fig. 25.

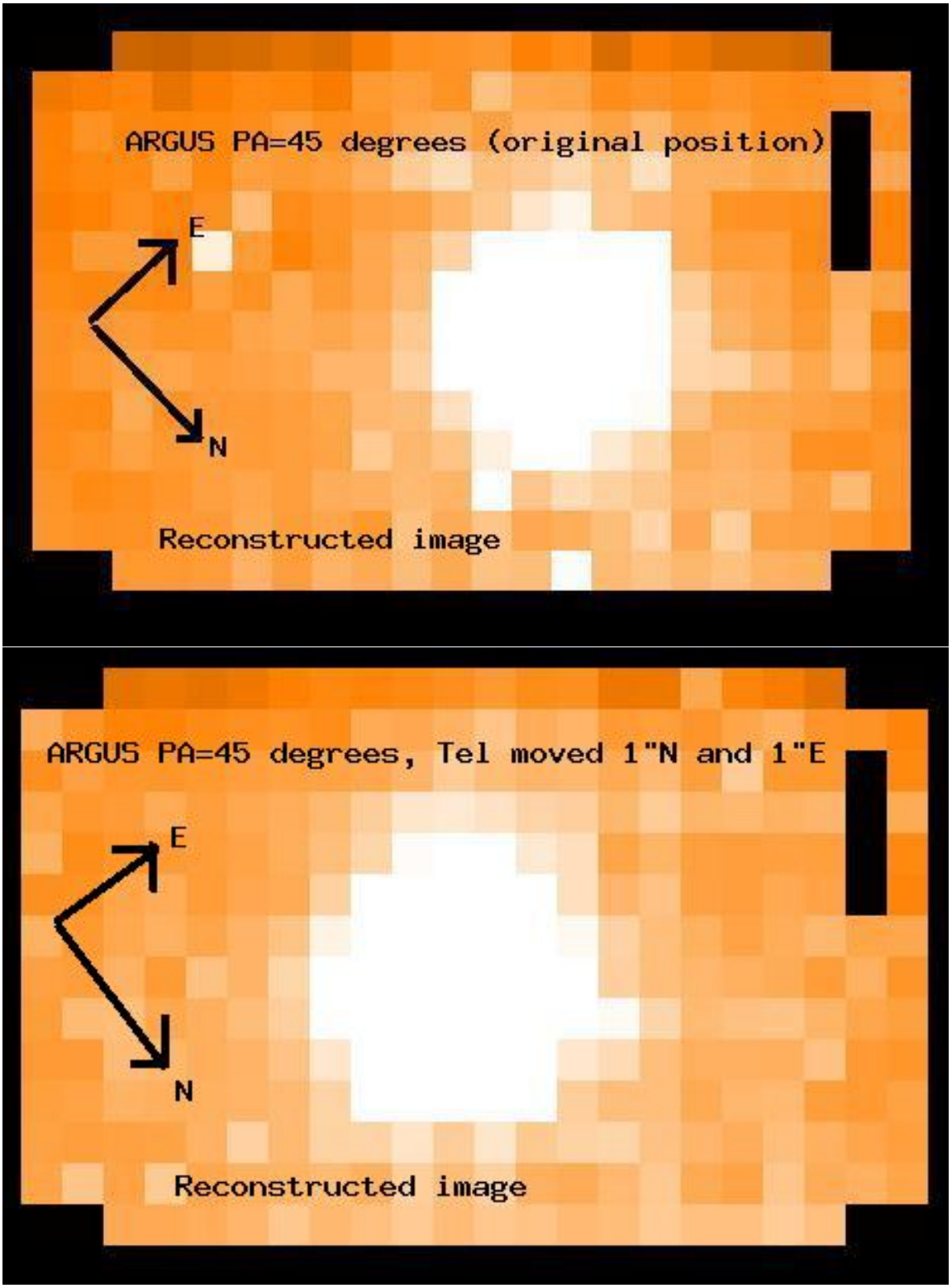

Figure 27: Argus reconstructed image with Position angle in the acquisition set to +45 degrees. Top panel: Original pointing. Bottom panel: Telescope moved by 1.0 arcseconds North and 1.0 arcseconds East i.e. the object moves 1.0 arcseconds South and 1.0 arcseconds West on ARGUS.

5 Pipeline in action: scripting your data reduction with ESOrex and .sof files

Using gasgano as pipeline GUI is a powerful way to get a feeling of how the GIRAFFE pipeline works. It allows you to get quickly familiarized with the input files and tables (optional and mandatory). Most importantly, it gives you the opportunity to play with the different parameters and check how they impact in your data product in real time.

However, once you have found your ideal set of parameters for each recipe you might want to automatize your data reduction without have to highlight different files and tables.

At this point you need to use EsoRex. EsoRex is a powerful parser which allows you to call a given recipe with a *set of files* as input parameters. Moreover you can pass values to the different parameters of each recipe via command line options or via a configuration file[8]. Below we give a simple example of how to use EsoRex.

In order to use EsoRex you have to prepare your input `.sof` files (set of files) which contains, as expected, a list of files to be used by a given recipe. In the example below, our raw science frame is `GIRAF.2003-04-14T05:49:22.740.fits` and /path is equivalent to /home/jsmoker/GIRAFFE-DRS/giraf-kit-2.6.0/calib/ (note that you should use a more up-to-date version of the pipline, like 2.8.1 at the time of writing).

For the input file to create the **master bias and bad pixel map**:

```
smoker@ESO$ cat 000_BIAS_2003-04-15.txt
GIRAF.2003-04-15T00:47:11.122.fits      BIAS
GIRAF.2003-04-15T00:47:58.821.fits      BIAS
GIRAF.2003-04-15T00:48:46.582.fits      BIAS
GIRAF.2003-04-15T00:49:37.202.fits      BIAS
GIRAF.2003-04-15T00:50:24.933.fits      BIAS
```

For the input file to create the **master flatfield and localisation centroids**:

```
smoker@ESO$ cat 000_FIBRE_FLAT_Medusa2_2003-04-15.sof
GIRAF.2003-04-15T00:11:23.320.fits       FIBER_FLAT
GIRAF.2003-04-15T00:17:29.884.fits       FIBER_FLAT
GIRAF.2003-04-15T00:23:36.497.fits       FIBER_FLAT
/path/giraf-2.6.0/cal/grating_HR316.fits              GRATING_DATA
/path/giraf-2.6.0/cal/line_mask_ThAr_H395.8_o14.fits  LINE_MASK
/path/giraf-2.6.0/cal/slit_geometry_medusa2.fits      SLIT_GEOMETRY_MASTER
gimasterbias_2003-04-15.fits    MASTER_BIAS
gibadpixelmap_2003-04-15.fits   BAD_PIXEL_MAP
```

[8]Please consult the EsoRex manual

For the input file to create the **dispersion solution** and **new slit geometry table** using the default slit geometry file (first pass of recipe giwavecalibration within esorex):

```
smoker@ESO$ cat 000_WAVE_Medusa2_OPT_2003-04-14.sof
GIRAF.2003-04-15T00:36:45.624.fits      ARC_SPECTRUM
gimasterbias_2003-04-15.fits            MASTER_BIAS
gibadpixelmap_2003-04-15.fits           BAD_PIXEL_MAP
ff_loccentroid_OPT.fits                 FF_LOCCENTROID
ff_psfcentroid_OPT.fits                 FF_PSFCENTROID
ff_locwidth_OPT.fits                    FF_LOCWIDTH
ff_psfwidth_OPT.fits                    PSF_WIDTH
fiber_profile_OPT.fits                  FIBER_PROFILE
/path/giraf-2.6.0/cal/grating_HR316.fits              GRATING_DATA
/path/giraf-2.6.0/cal/line_mask_ThAr_H395.8_o14.fits  LINE_MASK
/path/giraf-2.6.0/cal/slit_geometry_medusa2.fits      SLIT_GEOMETRY_MASTER
/path/giraf-2.6.0/cal/line_catalog_ThAr.fits          LINE_CATALOG
```

-here although we have specified FF_LOCCENTROID/LOCWIDTH **and** FF_PSFCENTROID/PSF_WIDTH by default the FF_PSF is preferred. For the input file to create the dispersion solution and a **new slit geometry file** (second pass of recipe giwavecalibration within esorex):

```
smoker@ESO$ cat 000_WAVE_Medusa2_OPT_NewSG_2003-04-14.sof
GIRAF.2003-04-15T00:36:45.624.fits      ARC_SPECTRUM
gimasterbias_2003-04-15.fits            MASTER_BIAS
gibadpixelmap_2003-04-15.fits           BAD_PIXEL_MAP
ff_loccentroid_OPT.fits                 FF_LOCCENTROID
ff_psfcentroid_OPT.fits                 FF_PSFCENTROID
ff_locwidth_OPT.fits                    FF_LOCWIDTH
ff_psfwidth_OPT.fits                    PSF_WIDTH
fiber_profile_OPT.fits                  FIBER_PROFILE
slit_geometry_setup_OPT_NewSG.fits      SLIT_GEOMETRY_MASTER
/path/giraf-2.6.0/cal/grating_HR316.fits              GRATING_DATA
/path/giraf-2.6.0/cal/line_mask_ThAr_H395.8_o14.fits  LINE_MASK
/path/giraf-2.6.0/cal/line_catalog_ThAr.fits          LINE_CATALOG
```

...and finally the .sof file for the **science reduction**:

```
smoker@ESO$ cat 000_SCIENCE_Medusa2_Flat_OPT_2003-04-14.sof
GIRAF.2003-04-14T05:49:22.740.fits      SCIENCE
gimasterbias_2003-04-15.fits    MASTER_BIAS
gibadpixelmap_2003-04-15.fits   BAD_PIXEL_MAP
ff_loccentroid_OPT.fits FF_LOCCENTROID
ff_psfcentroid_OPT.fits FF_PSFCENTROID
ff_locwidth_OPT.fits    FF_LOCWIDTH
ff_psfwidth_OPT.fits    PSF_WIDTH
fiber_profile_OPT.fits  FIBER_PROFILE
ff_extspectra_OPT.fits  FF_EXTSPECTRA
ff_exterrors_OPT.fits   FF_EXTERRORS
master_fiber_flat_OPT.fits      MASTER_FIBER_FLAT
dispersion_solution_OPT.fits    DISPERSION_SOLUTION
slit_geometry_setup_OPT.fits    SLIT_GEOMETRY_MASTER
/path/giraf-2.6.0/cal/grating_HR316.fits              GRATING_DATA
/path/giraf-2.6.0/cal/line_mask_ThAr_H395.8_o14.fits  LINE_MASK
```

Once you get your set of files ready, you simply call EsoRex as shown below, where in this case '/path' is equivalent to '/home/jsmoker/GIRAFFE-DRS/giraf-2.6/recipes/.libs'
First make the **master bias and bad pixel map**:

```
jsmoker@ESO$ esorex --recipe-dir="/path/" gimasterbias 000_BIAS_2003-04-15.txt
jsmoker@ESO$ mv esorex.log 000_BIAS_2003-04-15_esorex.log
```

Now the **master flat and the fibre localisation**:

```
jsmoker@ESO$ esorex --recipe-dir="/path" gimasterflat --fiber-splist="1-26,28-136"
 -extr-method="OPTIMAL" 000_FIBRE_FLAT_Medusa2_2003-04-15.sof  (all on one line)
jsmoker@ESO$ mv esorex.log 000_FIBRE_FLAT_Medusa2_OPT_2003-04-15.log
```

Now the **first pass to determine the wavelength solution** and to make a new **slit geometry table:**

```
jsmoker@ESO$ esorex --recipe-dir="/path" giwavecalibration --wcal-slit='true'
                    --wcal-lfoffset=20 --wcal-lswidth=20,20,20,20,20
                    000_WAVE_Medusa2_OPT_2003-04-14.sof  (all on one line)
jsmoker@ESO$ mv esorex.log 000_WAVE_Medusa1_SUM_NewSG_2004-06-14.log
jsmoker@ESO$ mv slit_geometry_setup.fits slit_geometry_setup_SUM_NewSG.fits
```

Now the **second pass** to determine an **improved wavelength solution** using the created slit geomtry table:

```
jsmoker@ESO$ esorex --recipe-dir="/path" giwavecalibration --wcal-slit='true'
                    --wcal-lfoffset=10 --wcal-lswidth=10,10,10,10,10
                    000_WAVE_Medusa2_SUM_NewSG_2003-04-14.sof  (all on one line)
jsmoker@ESO$ mv esorex.log 000_WAVE_Medusa1_SUM_NewSG_2004-06-14.log
```

Now do the **science reduction**:

```
jsmoker@eso esorex --recipe-dir="/path" giscience --extr-method="OPTIMAL" --flat-apply="true"
               --transmission-apply="false" --remove-bias="true" --bsremove-method="PROFILE"
                              000_SCIENCE_Medusa2_Flat_OPT_2003-04-14.sof (all on one line)
jsmoker@eso mv esorex.log 000_SCIENCE_Medusa2_Flat_OPT_2003-04-14.log
```

...which will produced the final fits files of the rebinned spectra ready for further analysis.

6 Automating data reduction using EsOrEx

Here a very basic example how you can automatize your data reduction using EsoRex.
We start with a generic sof. The idea is to replace automatically the word `_FILE_` by the real name of the raw science frame we want to reduce. Let us call this generic sof file `sample.sof`.

```
cmelo@ESO> cat sample.sof

_FILE_ SCIENCE
bad_pixel_map_0000.fits        BAD_PIXEL_MAP
master_bias_0000.fits          MASTER_BIAS
dispersion_solution_0000.tfits DISPERSION_SOLUTION
ff_extspectra_0000.fits        FF_EXTSPECTRA
ff_loccentroid_0000.fits       FF_LOCCENTROID
ff_locwidth_0000.fits          FF_LOCWIDTH
../static/grating_HR316.tfits         GRATING_DATA
../static/slit_geometry_medusa1.tfits SLIT_GEOMETRY_MASTER
```

Now consider the within the same night you observed 3 different points with the same set-up producing the raw frames, `f1.fits, f2.fits, f3.fits`. The script shown below uses the Unix command `sed` to replace the word `_FILE_` in the generic sof `sample.sof` by the real name of the file we want to reduce. The result is put into a a sof file with the same name of the raw frame. In the line below this newly created sof is passed to Esorex.

```
foreach f (f1.fits f2.fits f3.fits)
cat sample.sof | sed "s/_FILE_/$f/" > $f:r".sof"
esorex giscience $f:r".sof"
end
```

7 Reducing data using calibrations taken far away in time

Note that it is **not recommended** to reduce data taken using calibrations taken a long time away from the science data. In particular, wavelength calibration accuracy should not be trusted due to shifts in the instrument. That said, if there are no nearby calibrations taken then you may consider it worthwhile to search the archive for matching calibrations taken further away in time.

After downloading the relevant calibrations and science frames, the first thing you should do is to check the "INS.TEMP53" value in the headers and make sure that they are within 1.5 C (see Sect. 8.3). In any case the cross-dispersion value of your calibration data and science targets should match to within less than ∼0.8 of a pixel or else there may be problems with the data reduction.

Fig. 28 shows GIRAFFE observations of a part of an order for a single star in BH 144 taken at different dates. Fig 29 shows a single dates observations (2004-07-03) but reduced using calibrations taken on 2004-07-03, 2004-12-24, 2005-01-06 and 2005-02-18. The region of interest was normalised by fitting a 0th order polynomial and the wavelengths forced to match. The conclusion for these Medusa observations is that the flatfielding at least worked well with data taken far away with time but with similar values of cross-dispersion. Whether this generally holds true is not clear and we again remind the reader that the wavelength calibration would be very inaccurate.

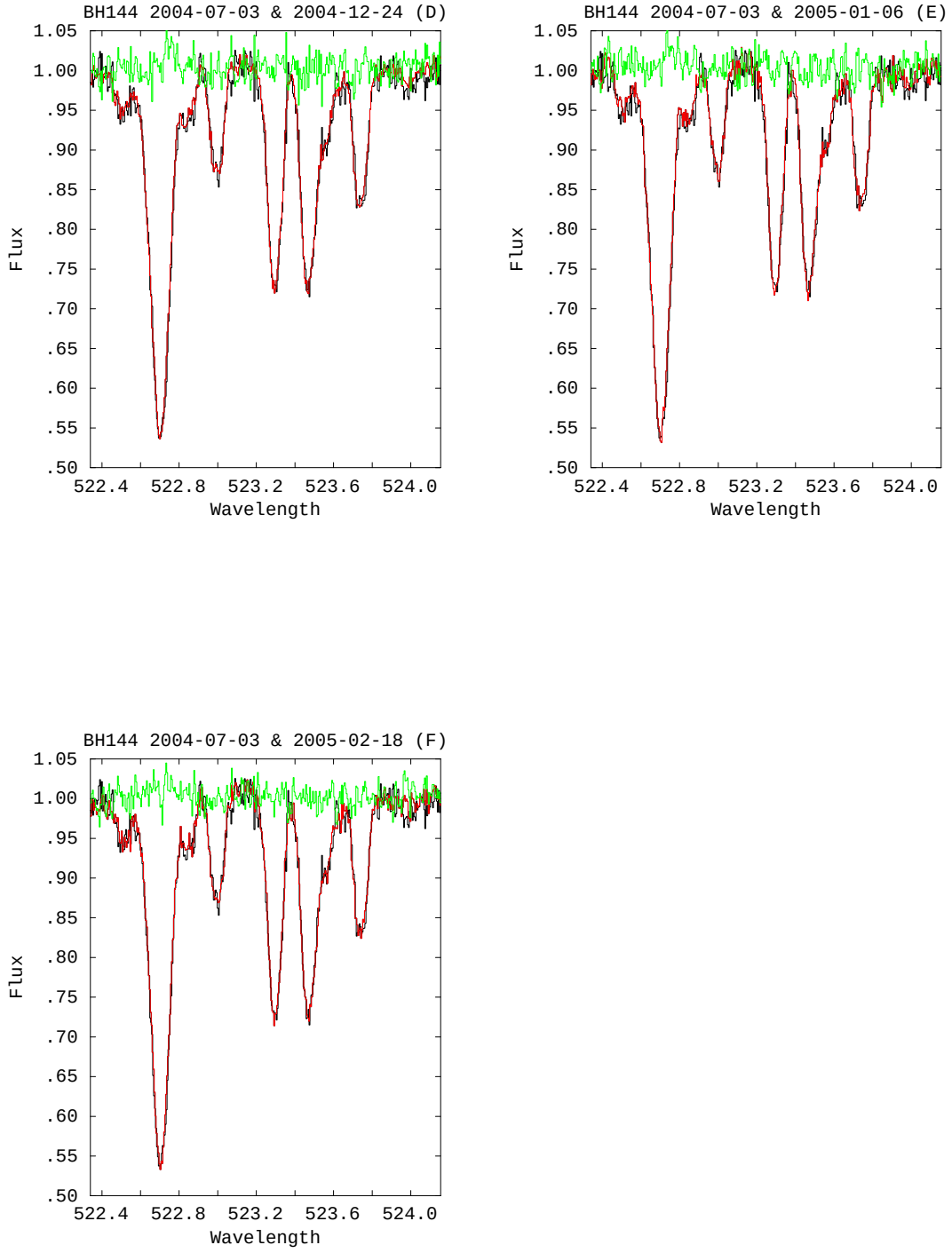

Figure 28: Reduced spectra of a single star in the cluster BH144 taken on 2004-07-03 compared with three other dates. Black line:2004-07-03. Red lines: (D) 2004-12-24 (E) 2005-01-06 and (F) 2005-02-18. Green lines show the ratio of the normalised spectra compared with 2004-07-03.

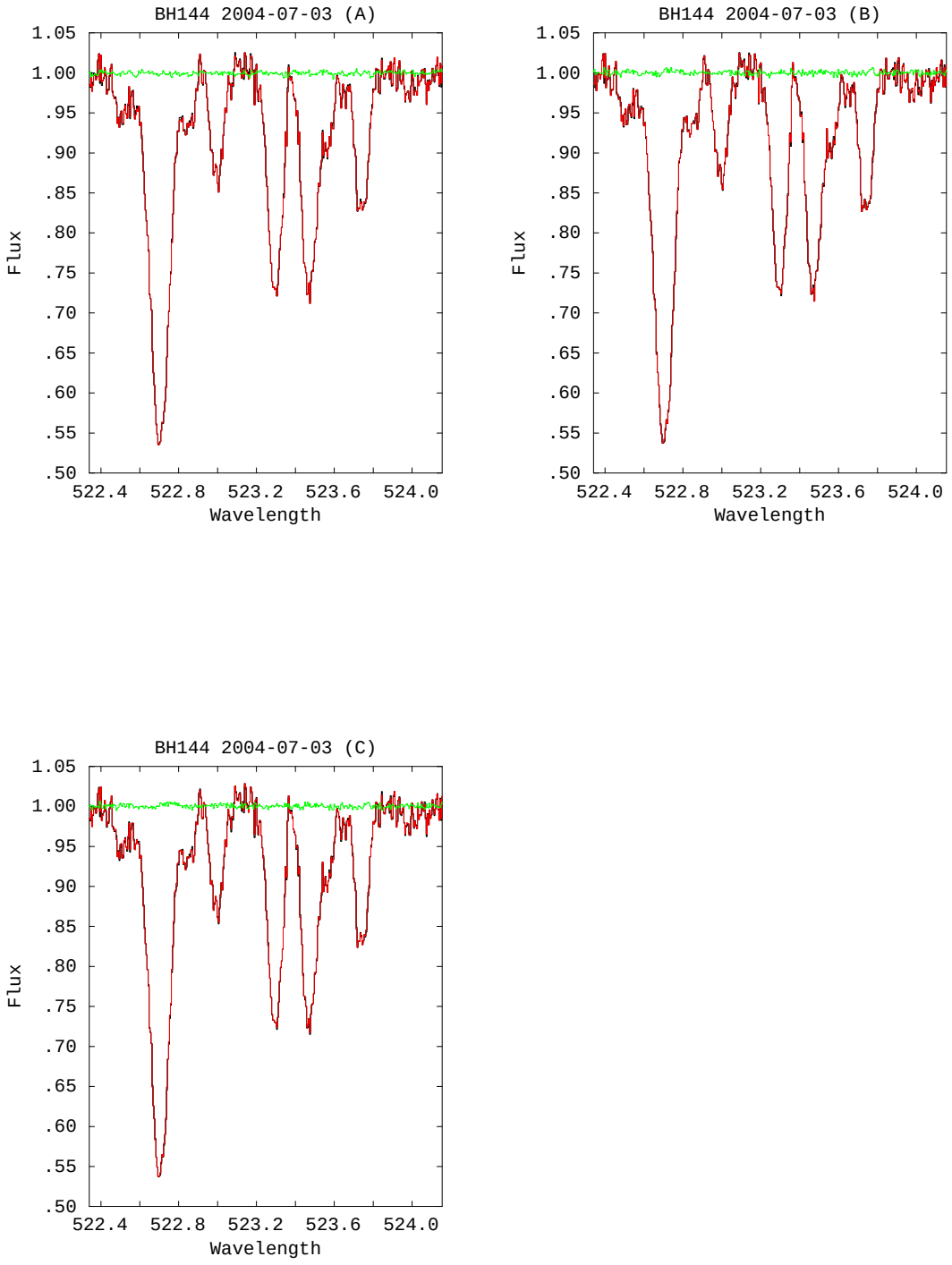

Figure 29: Reduced spectra of a single star in the cluster BH144 taken on 2004-07-03 and reduced using calibrations taken on different dates. (A) Data calibrated using 2004-07-03 and 2004-12-24 calibrations. (B) Data calibrated using 2004-07-03 and 2005-01-06 calibrations (C) Data calibrated using 2004-07-03 and 2005-02-18 calibrations. Green lines show the ratio of the reduced spectra calibrated using the different dates.

8 Common problems during data reduction

8.1 Incorrect input files

Many times problems with the GIRAFFE pipeline are caused by incorrect input files. Unfortunately the error reporting is not always intuitive and will be improved in future versions. Hence you should take a **lot** of care in ensuring that the input files are correct.

An example is given below of an esorex command to create a fibre flatfield.

```
[jsmoker@184dhcp22 ESO]$ cat Fibre_Flat_Medusa1_Bad_Sof.sof
GIRAF.2008-01-05T14:02:23.360.fits       FIBER_FLAT
GIRAF.2008-01-05T14:04:18.900.fits       FIBER_FLAT
GIRAF.2008-01-05T14:06:13.719.fits       FIBER_FLAT
/path/giraf-2.6.0/cal/grating_HR316.fits              GRATING_DATA
/path/giraf-2.6.0/cal/line_mask_ThAr_L682.2_o3.fits   LINE_MASK
/path/giraf-2.6.0/cal/slit_geometry_medusa1.fits      SLIT_GEOMETRY_MASTER
gimasterbias_2008-01-05.fits    MASTER_BIAS
gibadpixelmap_2008-01-05.fits   BAD_PIXEL_MAP
```

Let's try and run it (where '/path' is where your recipe directory lies)...

```
[jsmoker@184dhcp22 ESO]$ esorex --recipe-dir="/path" gimasterflat --fiber-splist="1-26,28-135"
                         -extr-method="SUM" Fibre_Flat_Medusa1_Bad_Sof.sof (all on one line)

<snip>
[ INFO  ] gimasterflat: Average # of in-borders pixels per spectra: 3.447e+04
[ INFO  ] gimasterflat: Average lost pixels per spectra: 0.09938
[ INFO  ] gimasterflat: Average lost pixels at upper border: 1.039
[ INFO  ] gimasterflat: Average lost pixels at lower border: -0.9396
[ INFO  ] gimasterflat: Average spectrum width: 10.3 +/- 2.286, (min, max) = (0, 10.91)
[ INFO  ] gimasterflat: Writing fiber localization ...
[WARNING] gimasterflat: Missing dark value property (ESO PRO DARK VALUE) will be set to 0.00!
[ INFO  ] gimasterflat: Fitting fiber profiles ...
[ INFO  ] gimasterflat: Fitting PSF profile parameters ...
[ INFO  ] gimasterflat: Writing fiber traces ...
[ INFO  ] gimasterflat: Extracting spectra ...
[ INFO  ] gimasterflat: Computing relative fiber transmission ...
[ ERROR ] gimasterflat: Relative transmission computation failed! Aborting ...
[ ERROR ] esorex: Execution of recipe 'gimasterflat' failed, status = 1
```

What is the problem? Well in this case it was the fact that we used 'grating_HR316.fits' for the GRATING_DATA and not 'grating_LR600.fits'. **If we run with grating_LR600.fits then the reduction runs fine**. Similar problems would be encountered if you use an incorrect LINE_MASK. Often such errors are caused by editing old versions of '.sof' files used for other setups and forgetting to change all the input files.

8.2 Old versions of EsoReX

Sometimes you may have different versions of EsOrEx on your system, from previous version of the GIRAFFE or UVES pipelines. If so, make sure that you are using the correct one!

For example:

```
[jsmoker@184dhcp22 ESO]$ which esorex
/home/jsmoker/UVES-DRS-CPL/bin/esorex  (an old version!!!)
```

....fails as the EsOrEx version was out of date. However, when we use the correct version the spectra are reduced without problems:

```
[jsmoker@184dhcp22 ESO]$ which esorex
/home/jsmoker/GIRAFFE-DRS/giraf-kit-2.6.0/bin/esorex
```

8.3 Mis-allignment between calibration data and science frames

Because GIRAFFE is not a thermally-controlled instrument, there can be shifts in the 'cross dispersion' position of the fibres when comparing science and calibration frames. ESO tries to minimise these shifts by taking calibrations as close as possible in time to the science frames and by re-scheduling observations if the shift is so big that data cannot be reduced by the nearest calibrations. The shift as a function in temperature for both gratings can be found on the following webpage:

`www.eso.org/observing/dfo/quality/GIRAFFE/reports/HEALTH//trend_report_STABILITY_HC.html`

Examples of the shifts in the HR and LR gratings in the x-axis with temperature are shown in Fig. 30

If your reduction is not working, it may be a good idea to look at the temperature at which the science and calibrations were taken by looking at the keyword "INS.TEMP53", viz:

```
[jsmoker@184dhcp22 ESO]$ dfits GIR*fits | grep -i temp | grep 53
HIERARCH ESO INS TEMP53 NAME = 'Surface table temperature top' / Temperature sen
HIERARCH ESO INS TEMP53 VAL  =    13.500000 / Temperature Sensor numeric value.
HIERARCH ESO INS TEMP53 NAME = 'Surface table temperature top' / Temperature sen
HIERARCH ESO INS TEMP53 VAL  =    13.830000 / Temperature Sensor numeric value.
HIERARCH ESO INS TEMP53 NAME = 'Surface table temperature top' / Temperature sen
HIERARCH ESO INS TEMP53 VAL  =    13.830000 / Temperature Sensor numeric value.
HIERARCH ESO INS TEMP53 NAME = 'Surface table temperature top' / Temperature sen
HIERARCH ESO INS TEMP53 VAL  =    13.840000 / Temperature Sensor numeric value.
HIERARCH ESO INS TEMP53 NAME = 'Surface table temperature top' / Temperature sen
HIERARCH ESO INS TEMP53 VAL  =    13.840000 / Temperature Sensor numeric value.
```

In the case above, all data were taken within a temperature of 0.34 Celcius, so shifts between science and calibrations should be small. You can double-check this by displaying an image and measuring the (x,y) position of the centre of a science fibre and flatfield. If the 'x' position is shifted by more than 0.8 pixels then it is possible that the pipeline will not be able to flatfield the data.

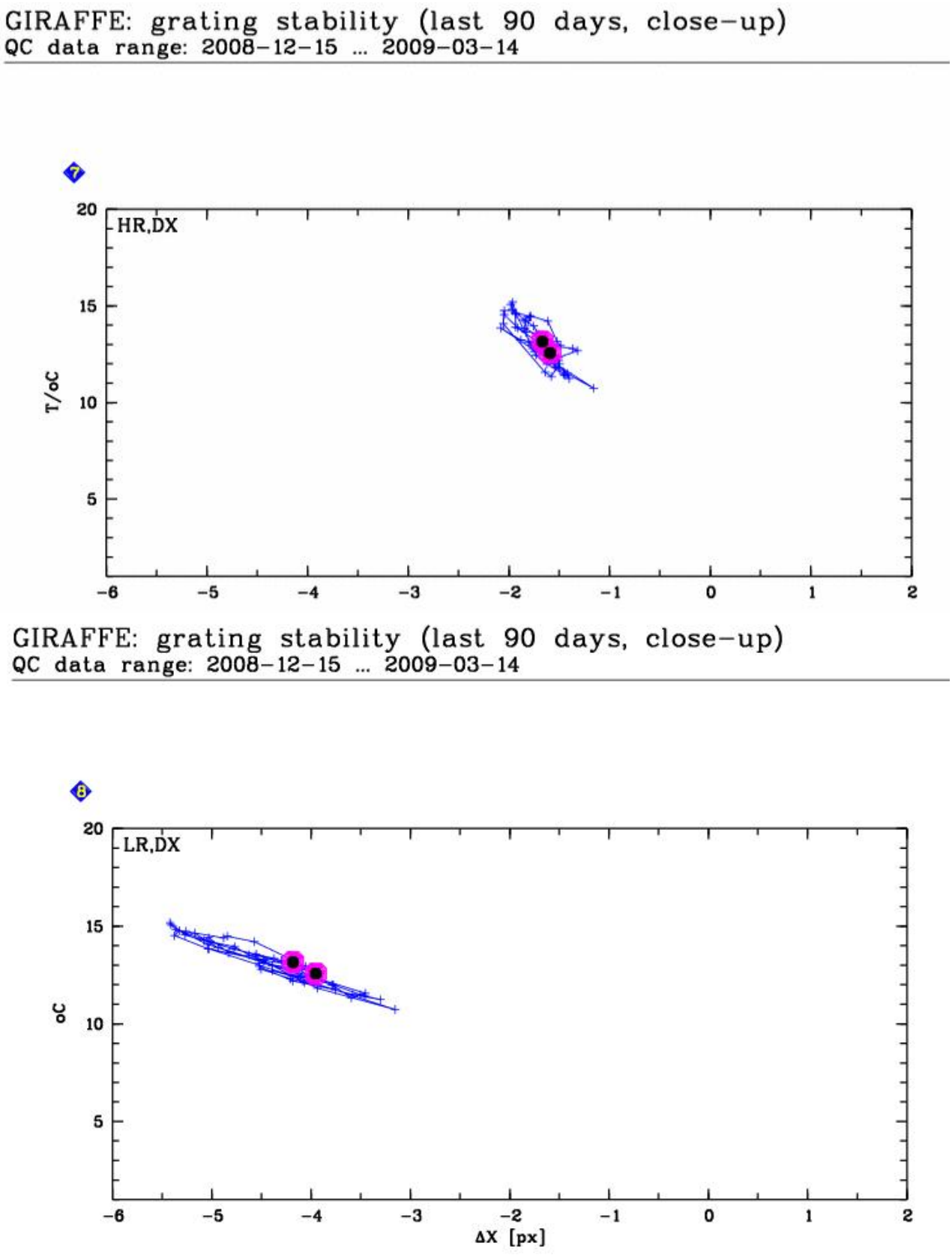

Figure 30: Shift of GIRAFFE X-axis with temperature in late 2008/early 2009. Calibrations are taken the day after the science and the temperatures are inspected so that they are within 1.5 C.

8.4 "Cannot create local file" error

The following error sometimes occurs in within `giscience` when one of the input files `ff_extspectra.fits` contains zeros:

```
"ERROR: 11:03:13 [ ERROR ] Cannot create local file! Aborting ...
Completion status: FAILURE
Execution error: Execution failed with code 1 "
```

This error is related to the new GIRAFFE CCD (sensitivity to cosmic rays). It was normally not a problem with the old CCD so there was no check necessary, but with the new CCD you can be unfortunate that all pixels belonging to a wavelength bin of a fiber are considered as bad pixels, i.e. the pixel value in the extracted flat field is set to 0.

A workaround for old versions of the pipeline is to add a small constant (0.000001) to `ff_extspectra.fits` and now giscience should successfully complete the reduction. Newer versions of the pipelines do not produce this error.

8.5 "Spectrum localization computation failed!" within gimasterflat

This error can be caused by a number of reasons. One common one is that the number of fibres traced is not correct.

In the following case the default of 137 fibres was used for a H395.8 setting;

```
cat gimasterflat_2009-04-15_14:36:37.log
...snip...
14:36:44 [ INFO  ] Generating mask (137 spectra expected) ...
14:36:50 [ INFO  ] 137 spectra detected in 1 wavelength bins
14:36:50 [ INFO  ] Computing spectrum positions and widths in pixel range [2053,2053]
14:36:50 [ INFO  ] Not enough data points 1 for 4 order fit
ERROR: 14:36:50 [ ERROR ] Spectrum localization computation failed!
ERROR: 14:36:50 [ ERROR ] Spectrum localization failed! Aborting ...
Completion status: FAILURE
Execution error: Execution failed with code 1
```

However, if `giraffe.fibers.spectra` was set within `gimasterflat` in `gasgano`, the recipe works fine:

```
cat gimasterflat_2009-04-15_14:41:06.log
...snip...
14:41:12 [ INFO  ] Fiber setup taken from flat field frame
                   '/home2/jsmoker/Instruments/FLAMES/Pipelines/Comparison
                       /Smartt_Gasgano/GIRAF.2003-04-15T00:11:23.320.fits'.
14:41:12 [ INFO  ] Setting number of fibers (ESO PRO SLIT NFIBRES) to 136
14:41:12 [ INFO  ] Bias sigma value: 2.39 e-
...snip...
```

```
/home2/jsmoker/Instruments/FLAMES/Pipelines/Comparison/Smartt_Gasgano
```

```
                                                              /ff_exterrors.fits
   /home2/jsmoker/Instruments/FLAMES/Pipelines/Comparison/Smartt_Gasgano
                                                              /ff_extpixels.fits
   /home2/jsmoker/Instruments/FLAMES/Pipelines/Comparison/Smartt_Gasgano
                                                              /ff_exttraces.fits
Completion status: SUCCESS
```

....and the spectra are reduced.

A Note for Mac users

Although the new generation of ESO pipelines based on CPL (Common Pipeline Libraries) has no official Mac OS support, some of the CPL pipelines have been reported to compile without problems on Mac OS machines (e.g., SINFONI and UVES). Care must be taken however, because even if the pipeline appears to work there may be subtle effects that cause incorrect results.

___oOo___