Very Large Telescope

Paranal Science Operations

GIRAFFE data reduction cookbook

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

Issue 79.0, Date 26/08/2006

C. Melo

Prepared . . . .
                                    Date   Signature

G. Marconi

Approved . . . .
                                    Date   Signature

O. Hainaut

Released . . . .
                                    Date   Signature

This page was intentionally left blank

Change Record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
79	First version published		

This page was intentionally left blank

Contents

1 Introduction   **1**

    1.1   Purpose 1

    1.2   Reference documents 1

    1.3   Abbreviations and acronyms 1

    1.4   Stylistic conventions 1

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

    2.1   Multi-fiber spectroscopy 2

    2.2   Correcting detector cosmetic effects 4

    2.3   Fiber localization and tracing 5

    2.4   Extraction, flat-field spectra and fiber transmission 6

    2.5   Scattered light correction 8

    2.6   Wavelength calibration 10

    2.7   Extraction of the science 10

    2.8   Sky subtraction 10

3 Pipeline in action: Gasgano, the friendly way   **11**

    3.1   Before you start 11

    3.2   Starting gasgano 11

    3.3   Case 1: your calibration data is up-to-date 11

    3.4   Case 2: Making your own calibration database 21

       3.4.1   gimasterbias 22

       3.4.2   gimasterflat 22

       3.4.3   giwavecalibration 24

       3.4.4   giscience 24

4 Pipeline in action: scripting your data reduction with ESOrex   **27**

A Preparing your sof file: a first example   **30**

B Note for Mac users   **31**

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
Hectospec	6.7-m MMT	MMT	300
6dF	1.2-m UK Schmidt	AAO	150
2dF	3.9-m AAT	AAO	400
Hydra	3.5-m WYIN	KPNO	90
FLAMES	8.2-UT2	VLT	135/8

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

This section aims at making 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 Sec 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 parameter allowing to fine-tune the data-reduction. We refer to the user 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 spectrum.

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" into 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. All this is shown in Figure 1. 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-fibers differ from each other in technical details, the basic idea is the same for all of them.

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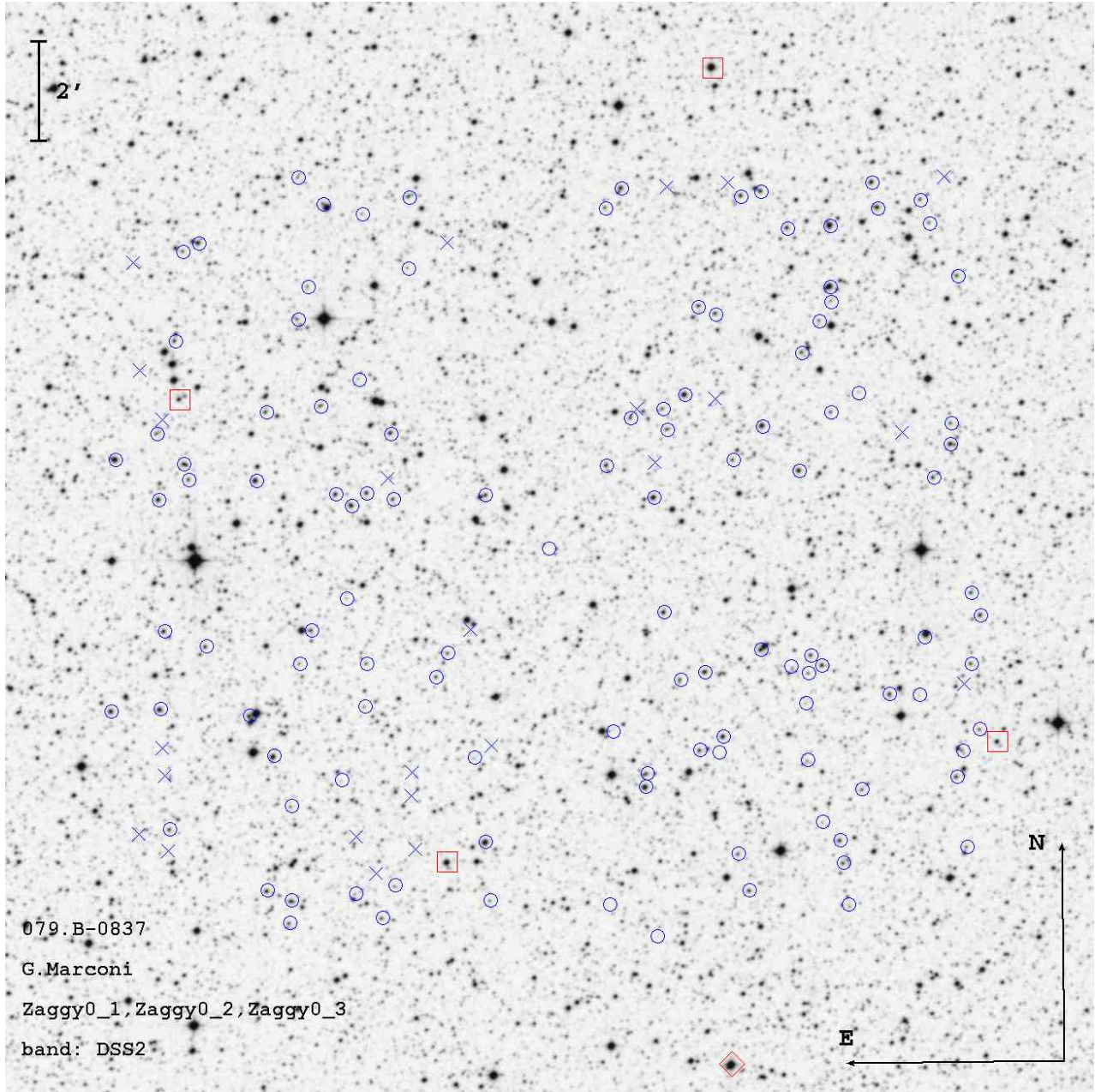
Figure 1: An example of the potentiality of multi-fiber spectrographs. In one shot up to 135 spectra are recorded by GIRAFFE and up 8 by UVES. The figure a finding chart of a typical FLAMES observation. Circles indicate science targets. Sky positions are marked with crosses and the four FACBs used for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

- 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

- 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.

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.

---

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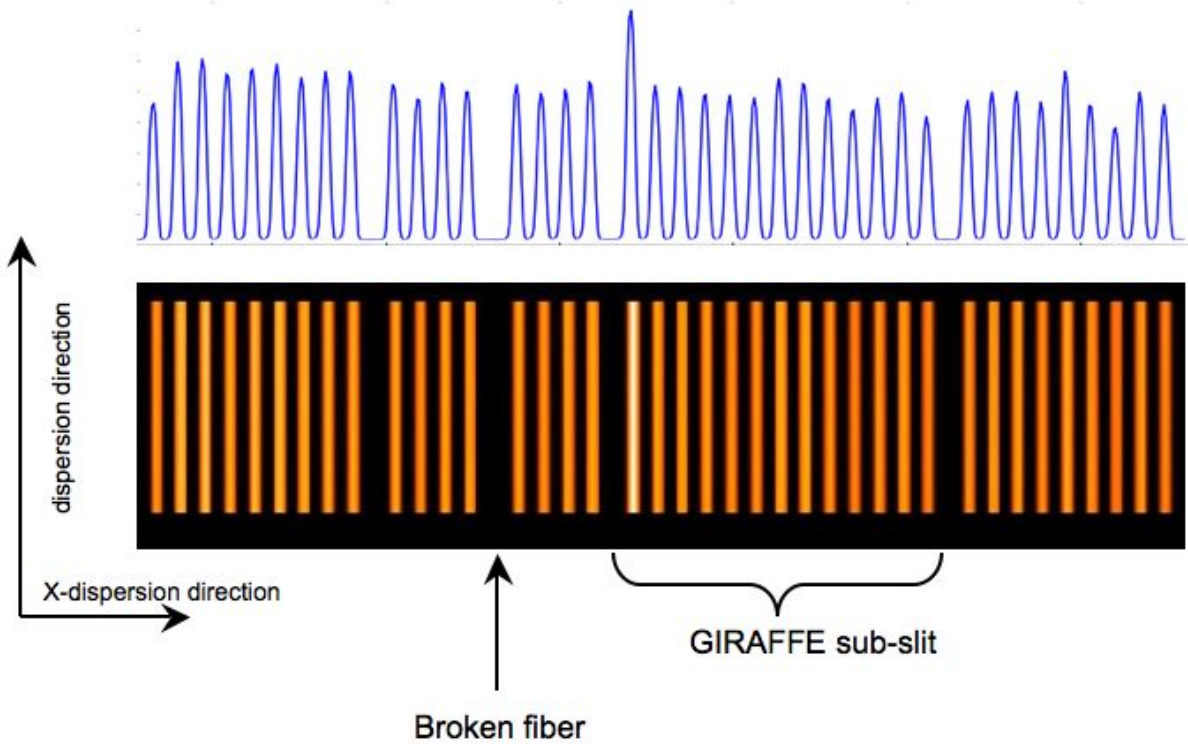
Figure 2: An extract of a raw image of a flat-filed frame is shown the bottom panel. The sub-slits (packets of fibers) defined in the previous figure a clear seen in the image. As well as a broken fiber. In the top panel, a cross section of the frame is shown where the nearly gaussian profile of the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

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 dispersion direction. The direction perpendicular to the dispersion is called cross-dispersion direction (or spatial direction in slit spectroscopy). These directions are also indicate in Fig. 2.

Thus the first task in the data reduction process (after cleaning the detector defects) is to know where the spectra of each fiber actually are 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 latter. 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 guassian 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-fiel 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 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).

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.

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 divide by the flat-field image.

In the case of fiber spectroscopy we have seen that the intensity of the pixels drops quickly on 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
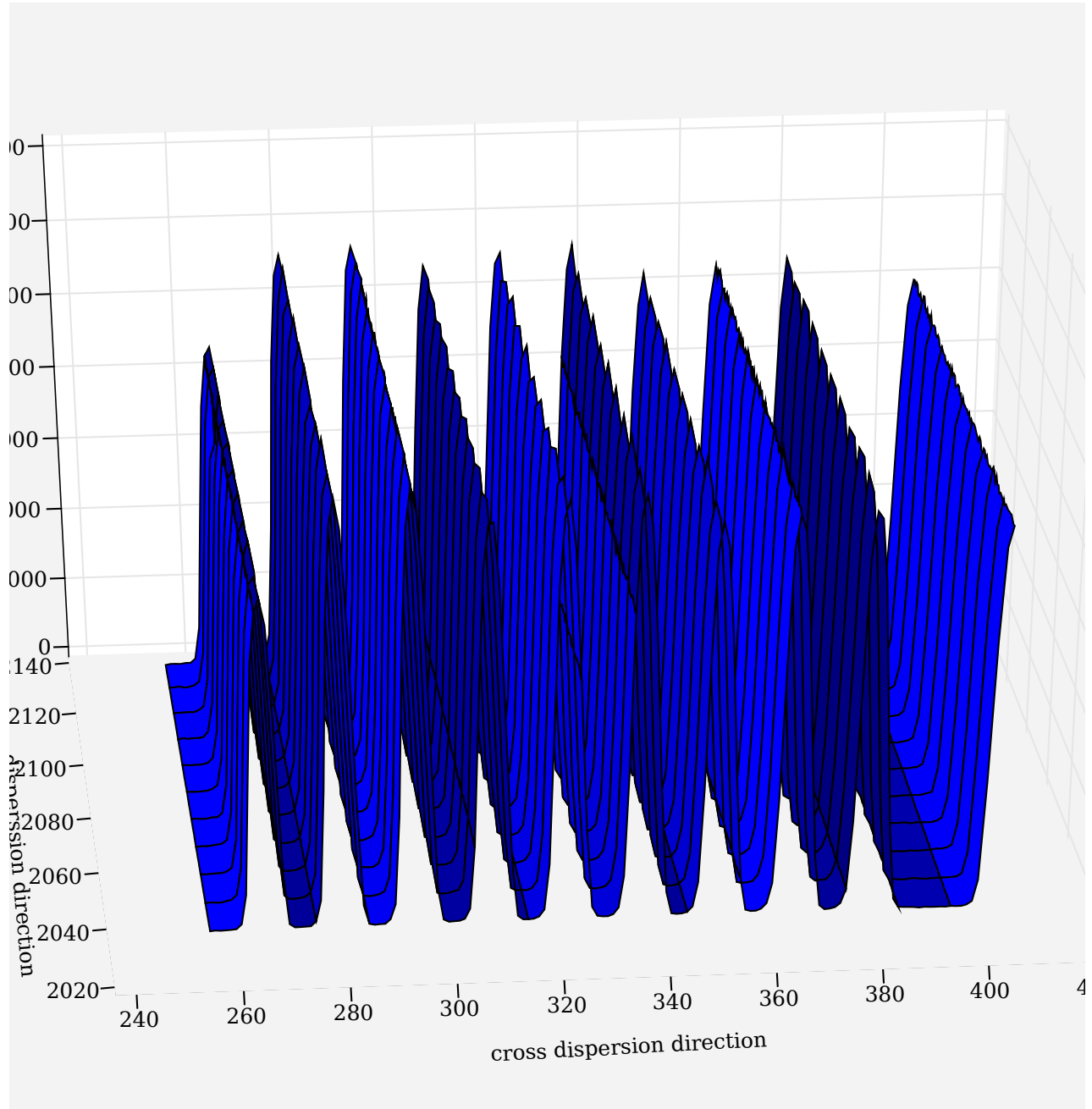
Figure 3: 3D representation of the first packet seen in Fig. 2 showing the gaussian tubes.

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

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.

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 be subtracted. 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

---

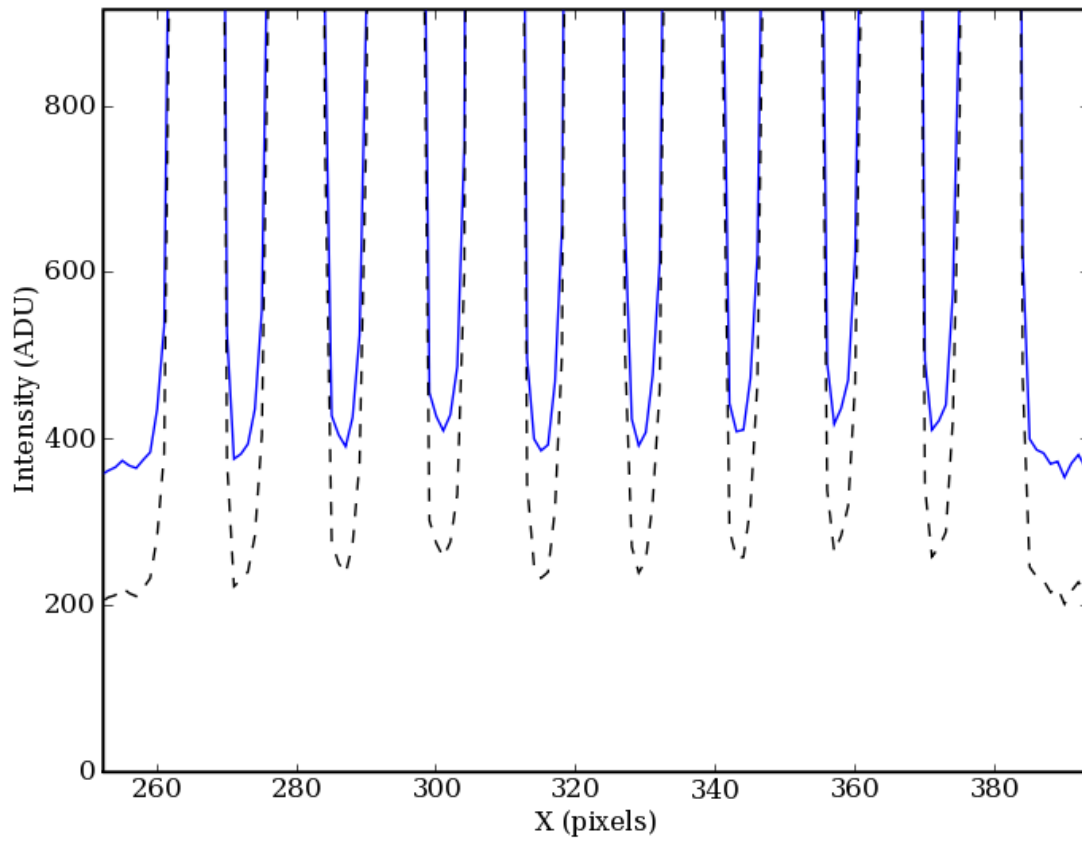[2]This paper is available at http://www.eso.org/instruments/flames/doc/spie.ps

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

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, $\lambda \to f(x, y)$.

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.

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% can achieved. This requires:

- proper bias and dark correction

---

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

- scattered light correction

- fiber-to-fiber transmission

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.

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.

- to have the demo data organized in your disk under the following structure (is it really important?)

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]

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 `giscince` 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 `gibias`

---

[4]it can be downloaded at http://www.eso.org/projects/dfs/dfs-shared/web/vlt/vlt-instrument-pipelines.html. 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.
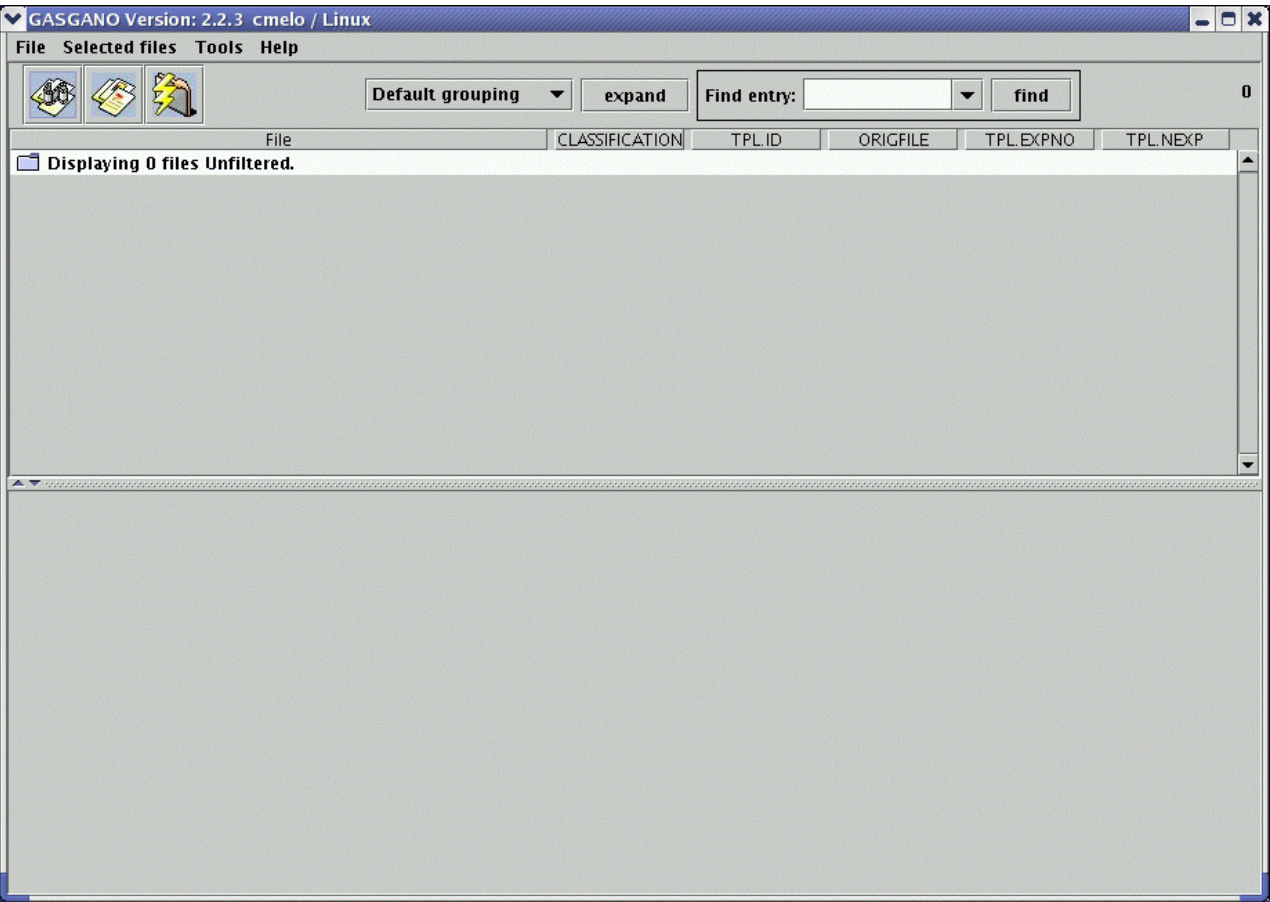
Figure 5: Entry screen of gasgano

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. `DISPERSION SOLUTION`. Wavelength calibration solution found with `giwavecal`

6. `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.

7. `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:

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 `CLASSFICATION`. 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 6).

As shown in Figure 7, 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 the 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 8). A 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

.
.
.
```
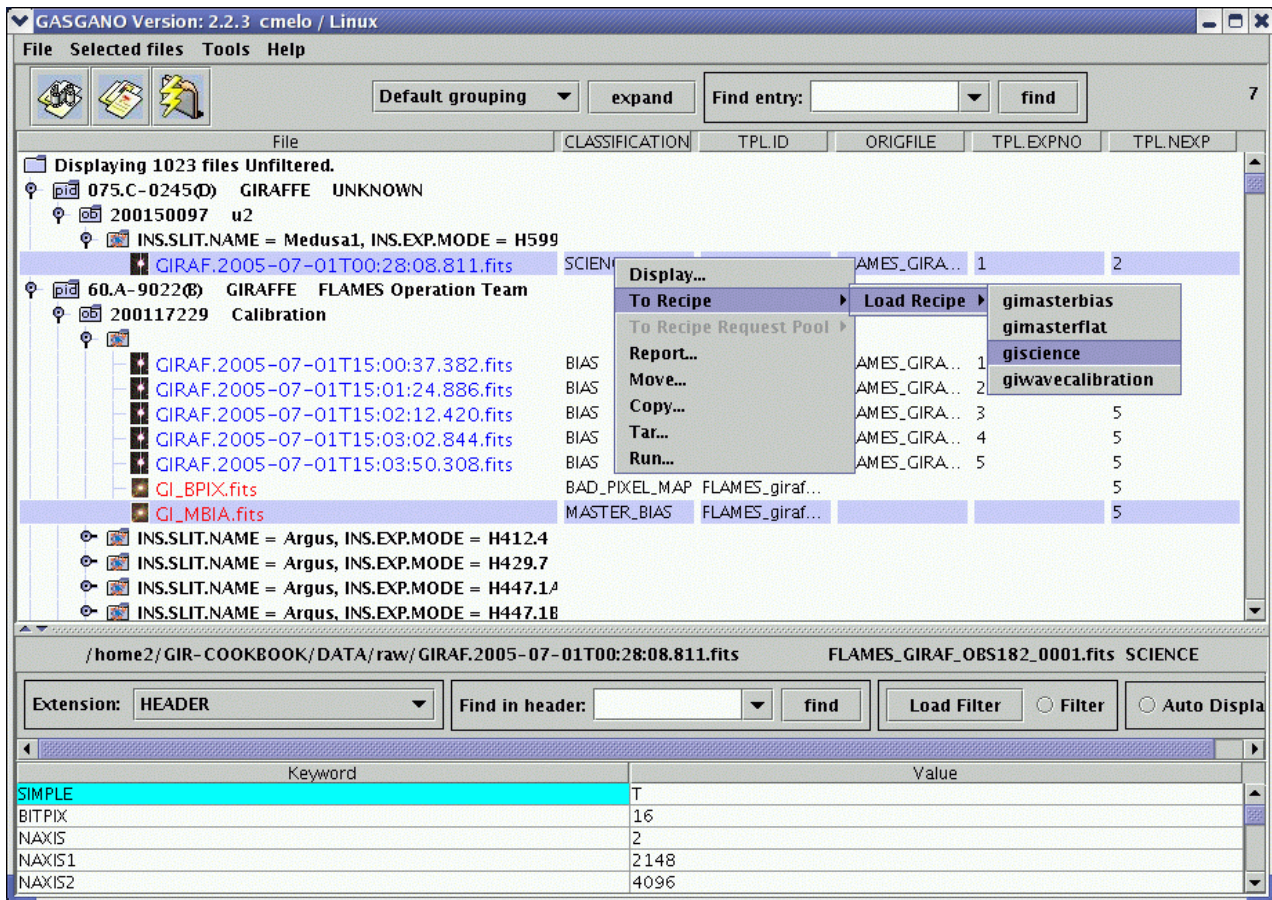
Figure 6: Passing a raw science frame to the recipe `girscience` using gasgano. Files are first selected by holding `CTRL` key and clicking on the calibration and science files. Then with the right button, they are sent the a given recipe. In the example below, the input files are sent to the recipe `giscience`.
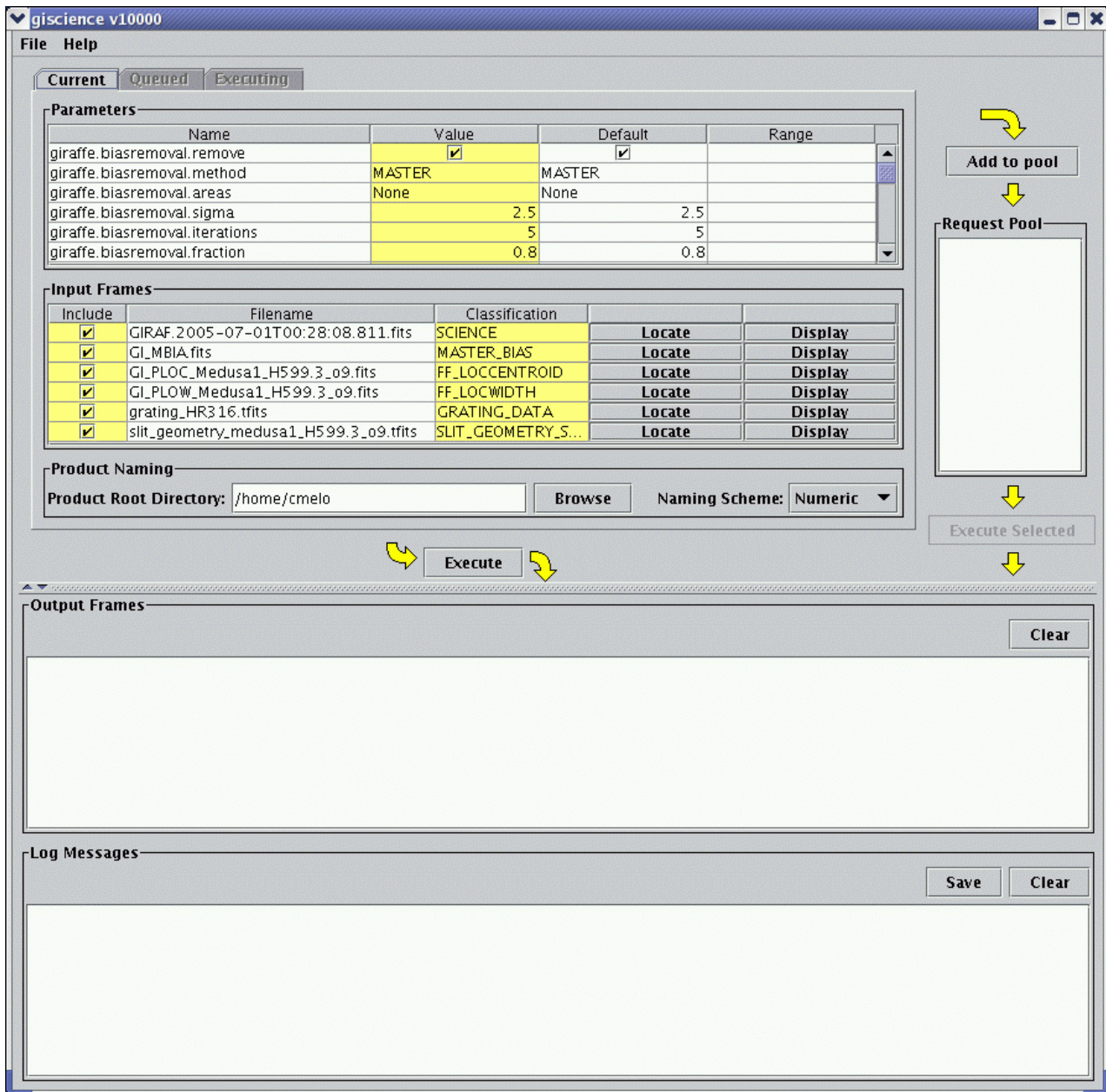
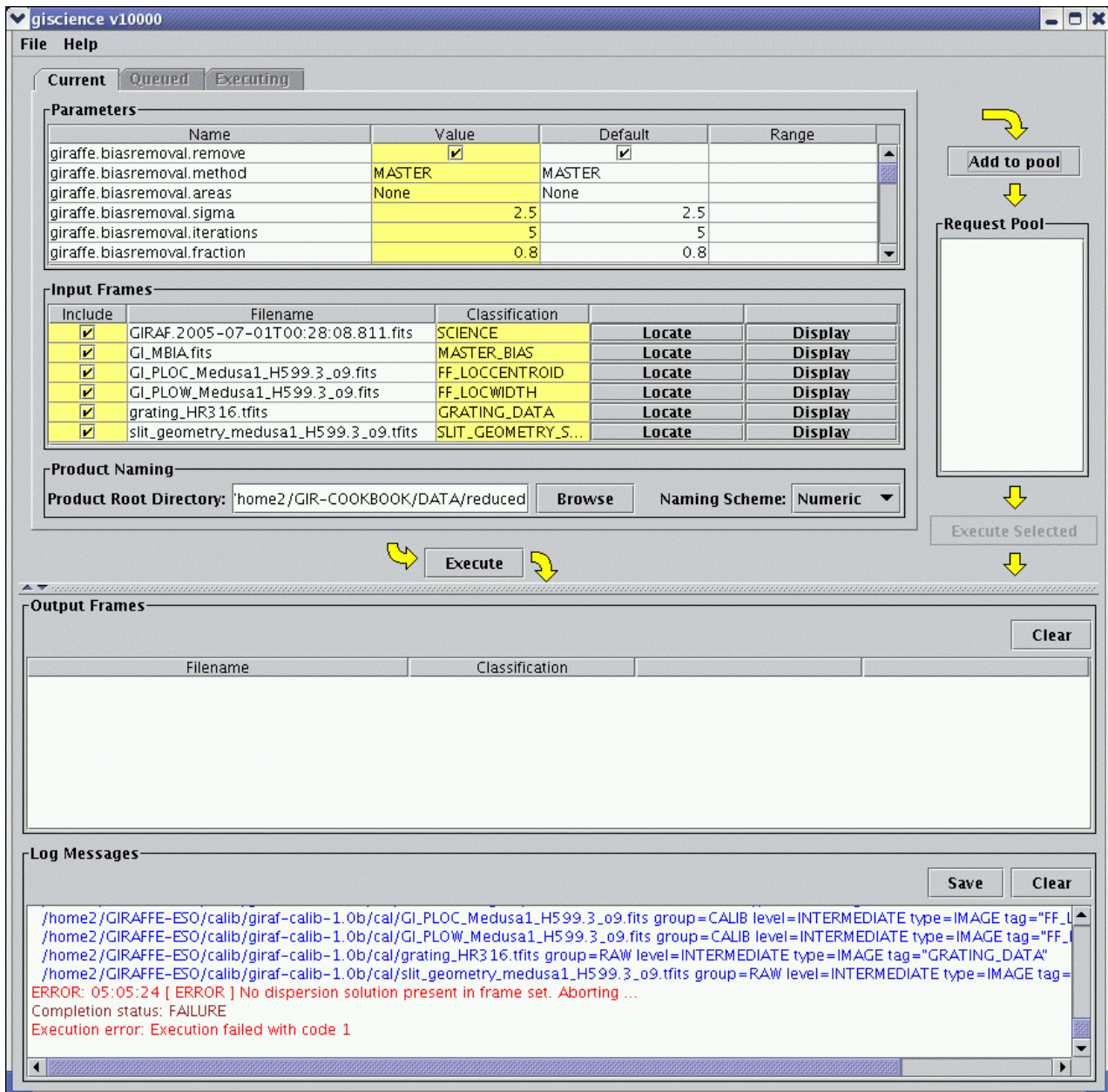Figure 7: Pop-up panel controlling the parameters and the arguments of `girscience`.

Figure 8: 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.

```
  /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.

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:

```
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
```

---

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

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
```

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
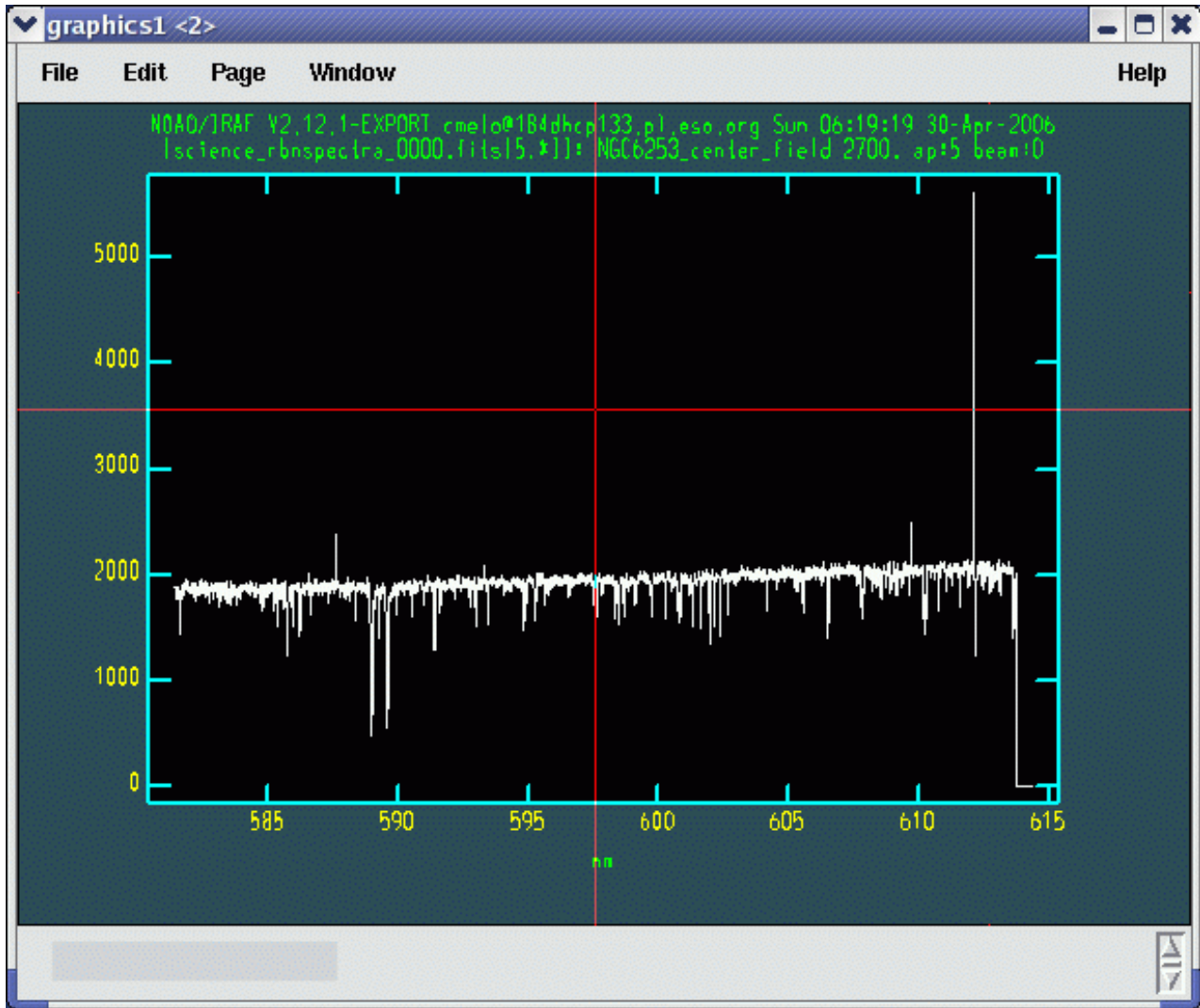
Figure 9: Stellar spectrum of a member of NGC6253 in aperture 5.

shown in Figure 9. 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| 1744590|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
```

. . . .

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). It is not clear when we should determine a new slit geometry. For the moment we recommend to redo it for observations separated by more than a 2 weeks.
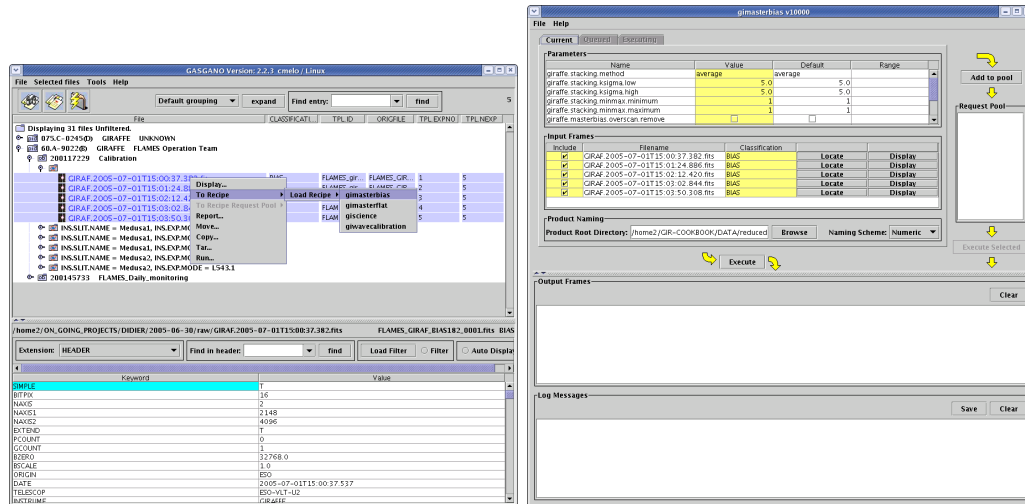
Figure 10: Master BIAS reduction.

```
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.

3.4.1 gimasterbias

Select all BIAS FRAMES and pass them to the recipe gimasterbias as shown in the left panel of Fig 10. A new window appears (right panel of Fig 10) where all parameters related to the recipe `gimasterbias` can be controlled. 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 gasgago as shown in Fig 11.

- QC parameters returned by the recipes (important)

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
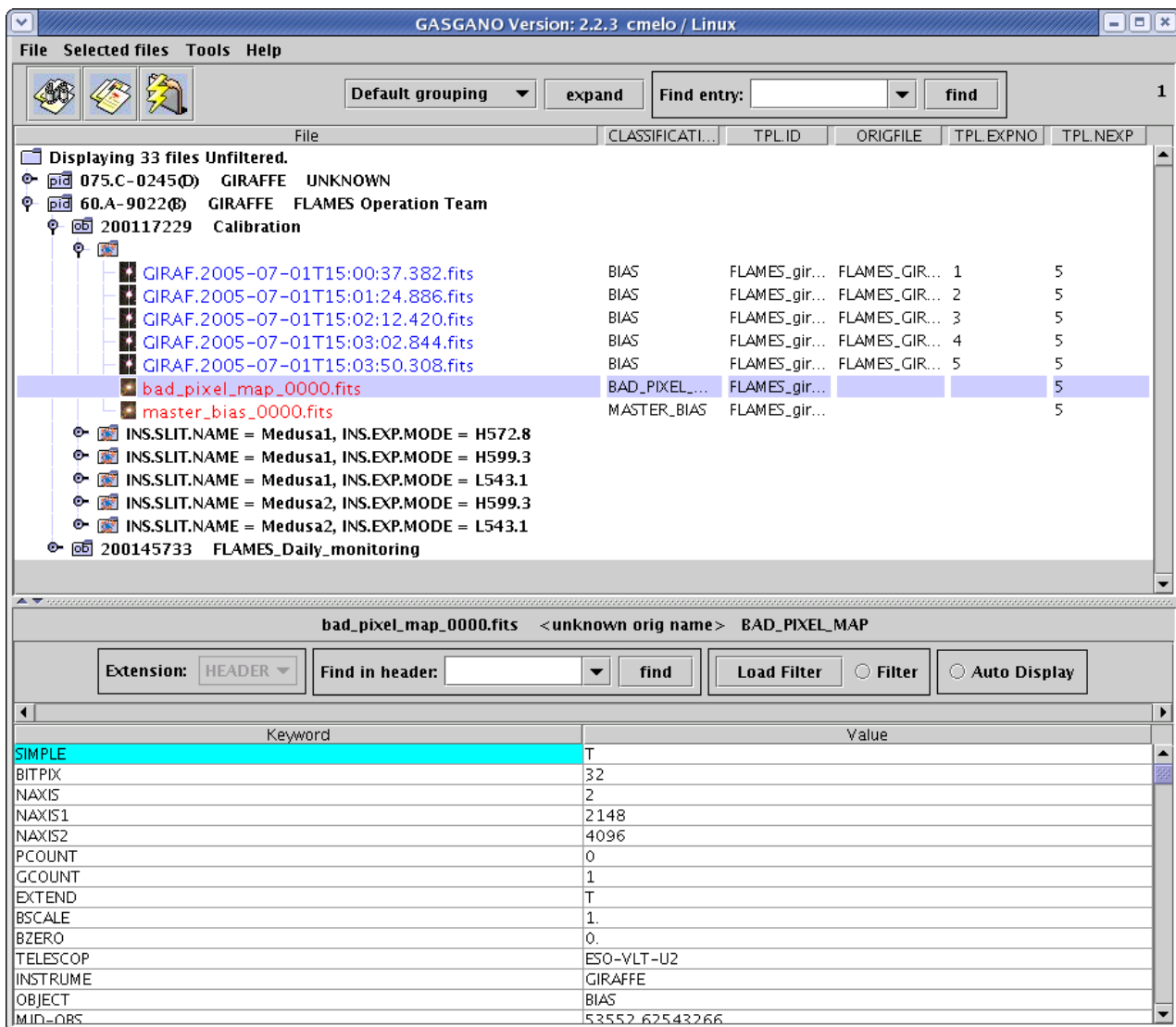
Figure 11: Gasgano automatically updates the list of files. The reduced files created by `gimasterbias` are seen in the gasgano file list.

(`GRATING_DATA`, here also you have to chose the right one, in our example the data have been taken with the HR 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.

In Fig.12 we see that the recipe failed because *only* 133 were found instead of 136. Fixing `giraffe.fibers.nspectra` in the Parameter sub-window `gimasterflat` runs fine 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. 13).

`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` which gives the pipeline the list of enabled fibers. For more details, please refer to Sec. 9.2.3 of the pipeline user manual.

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[8].

3.4.4   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

---

[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

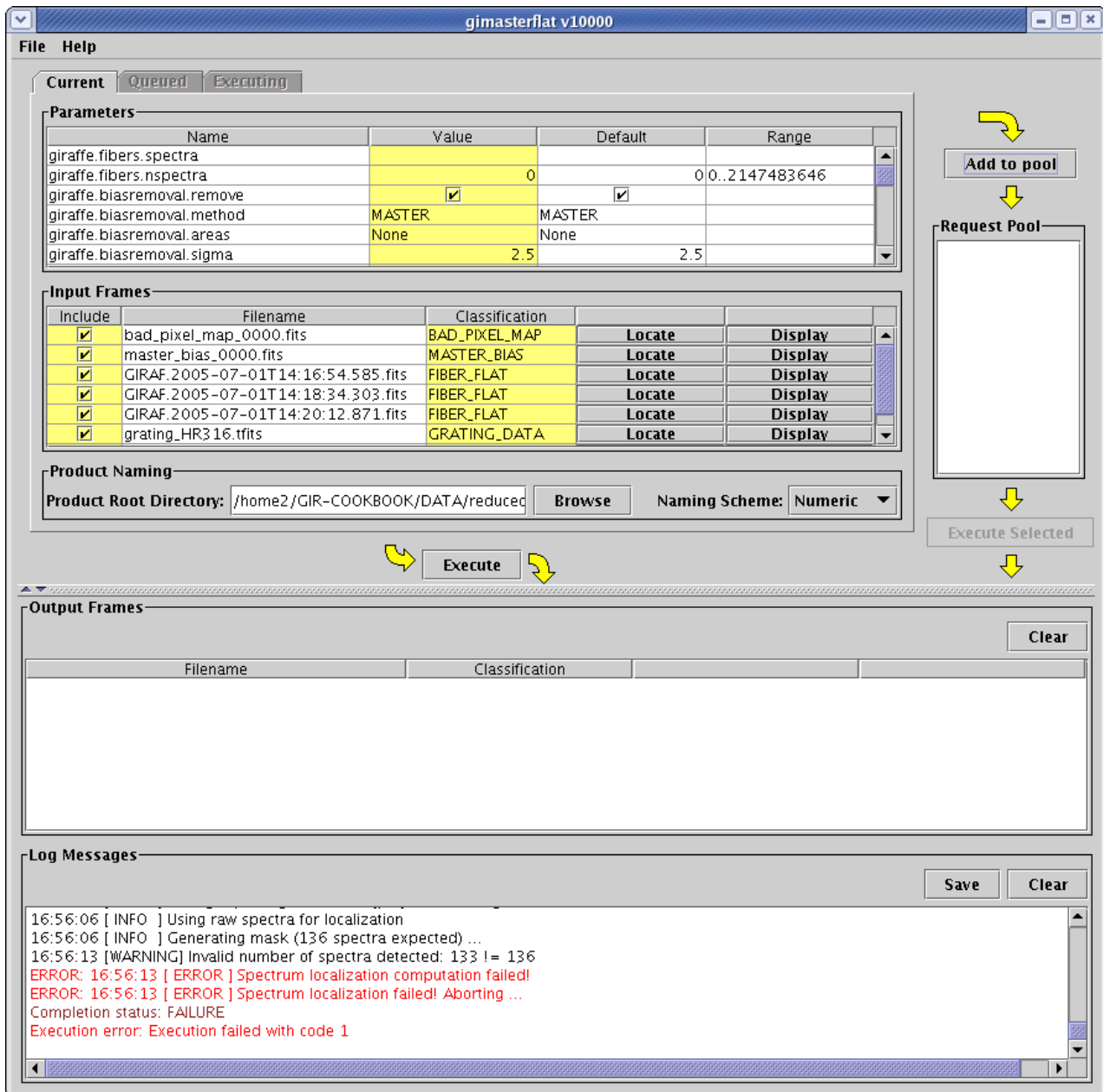[8]This is not yet implemented in the current version of the pipeline

Figure 12: 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).
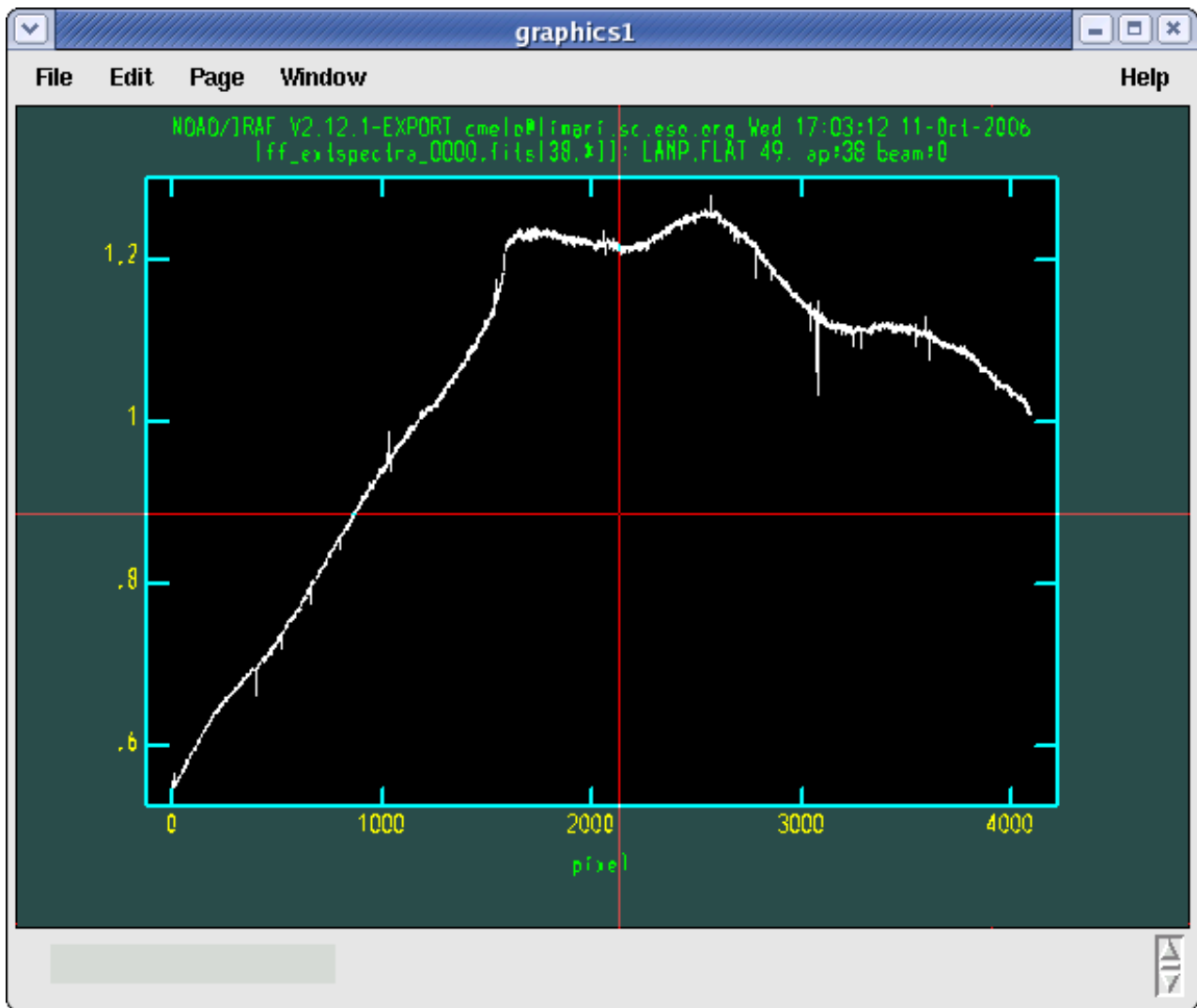
Figure 13: Extract flat-field spectrum for one the fibers. Spectrum was produced by `gimasterflat` and will be wavelength calibrated by the next step carried out by `giwavecalibration`.

`giscience` adds up the signal inside the PSF fitted by `gimasterflat`. Optimum (weighted) extraction will be implemented in the next release.

`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 resampled fluxes for each wavelength bin.

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. It is easy to install and has many nice functionalities to analyze your data. There is a caveat though. The world coordinate system of the GIRAFFE cubes follow the Euro3D recommendations, whereas QfitsView doesn't. Therefore in order to make QfitsView to be able to read the wavelength calibration of the GIRAFFE pipeline you have to add a keyword called `CDELT3` its value should be the same as the keyword `CD3_3` already present in the fits header of the cube. If you use `IRAF` or `PyRAF` you can do:

```
--> hedit science_cube_spectra_0000.fits CDELT3 '(CD3_3)' add+ update+
add science_cube_spectra_0000.fits,CDELT3 = 0.005
update science_cube_spectra_0000.fits ? (yes):
science_cube_spectra_0000.fits updated
```

Now `science_cube_spectra_0000`fits. can read by QFitsView as shown in Fig. 14.

4   Pipeline in action: scripting your data reduction with ESOrex

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 important, it gives you the opportunity to play with the different parameters and check how they impact in your data product in real time.
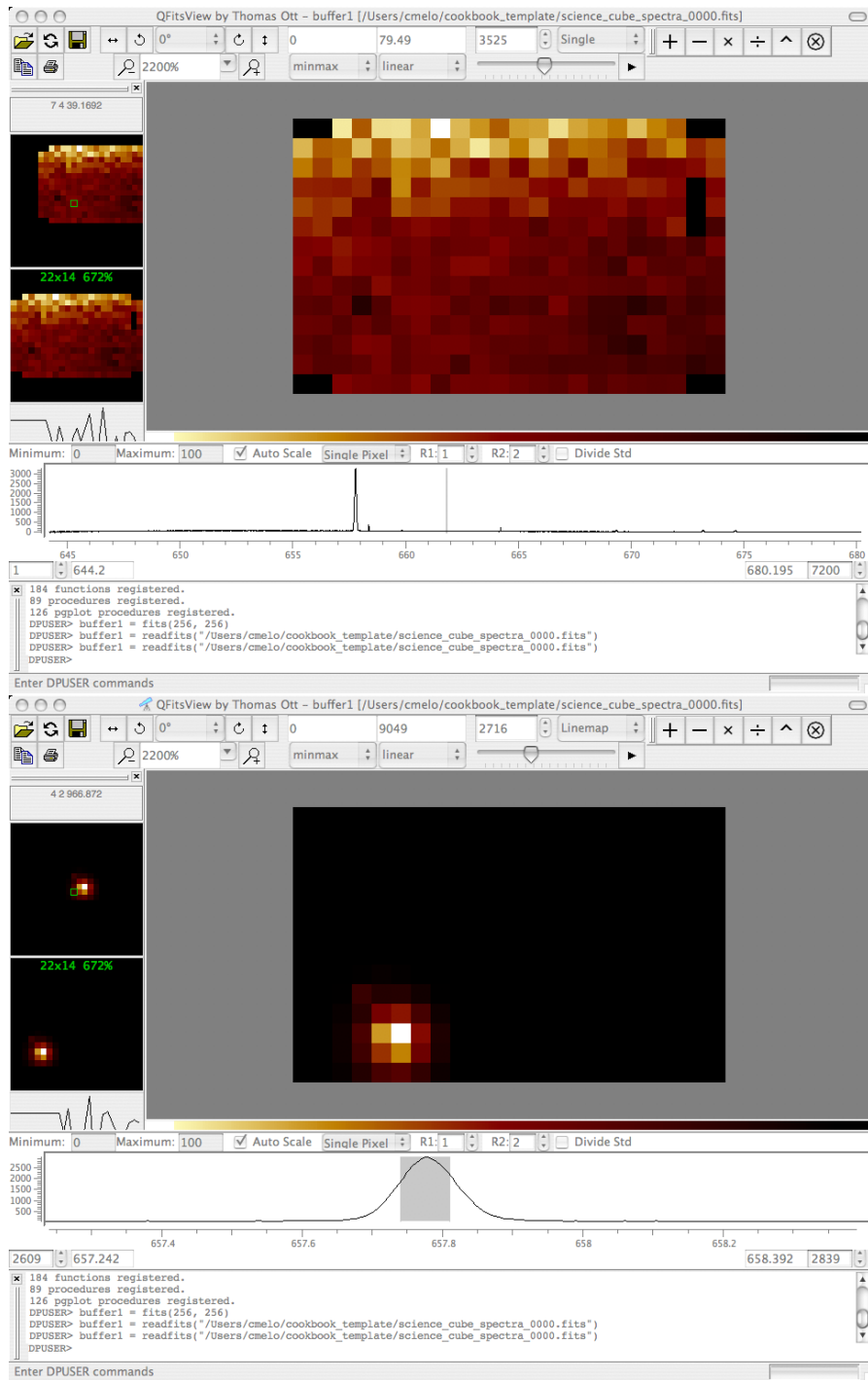
Figure 14: Argus cube of an emission line object produced by the pipeline and visualized by QFitsView. The upper panel show the flat image, i.e, the whole wavelength range is considered. Nothing is really seen with respect to the background. In the lower panel, one the core of the emission is display. The object pops-up with respect to the background.

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[9]. 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 `../raw/GIRAF.2005-07-01T00:28:08.811.fits`. All other files or tables were produced by the reduction of the calibration frames or are static tables.

```
limari:reduced 62> cat giscience.sof
../raw/GIRAF.2005-07-01T00:28:08.811.fits 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
```

Once you get your set of files ready, you simply call EsoRex as shown below:

```
limari:reduced 63> esorex giscience giscience.sof
```

The log with the processing in flushed to the screen and also dumped into a log file.

```
17:21:19 [ INFO  ] giraffe_rebin_spectra  Wavelength range    : Setup
17:21:35 [ INFO  ] move_products  Created product
                   /home2/GIR-COOKBOOK/DATA/reduced/out_0000.fits
17:21:35 [ INFO  ] move_products  Created product
                   /home2/GIR-COOKBOOK/DATA/reduced/out_0001.fits
17:21:35 [ INFO  ] move_products  Created product
                   /home2/GIR-COOKBOOK/DATA/reduced/out_0002.fits
17:21:35 [ INFO  ] move_products  Created product
                   /home2/GIR-COOKBOOK/DATA/reduced/out_0003.fits
17:21:35 [ INFO  ] move_products  Created product
                   /home2/GIR-COOKBOOK/DATA/reduced/out_0004.fits
17:21:35 [ INFO  ] move_products  Created product
                   /home2/GIR-COOKBOOK/DATA/reduced/out_0005.fits
17:21:35 [ INFO  ] move_products  Created product
                   /home2/GIR-COOKBOOK/DATA/reduced/out_0006.fits
17:21:35 [ INFO  ] move_products  7 products created
```
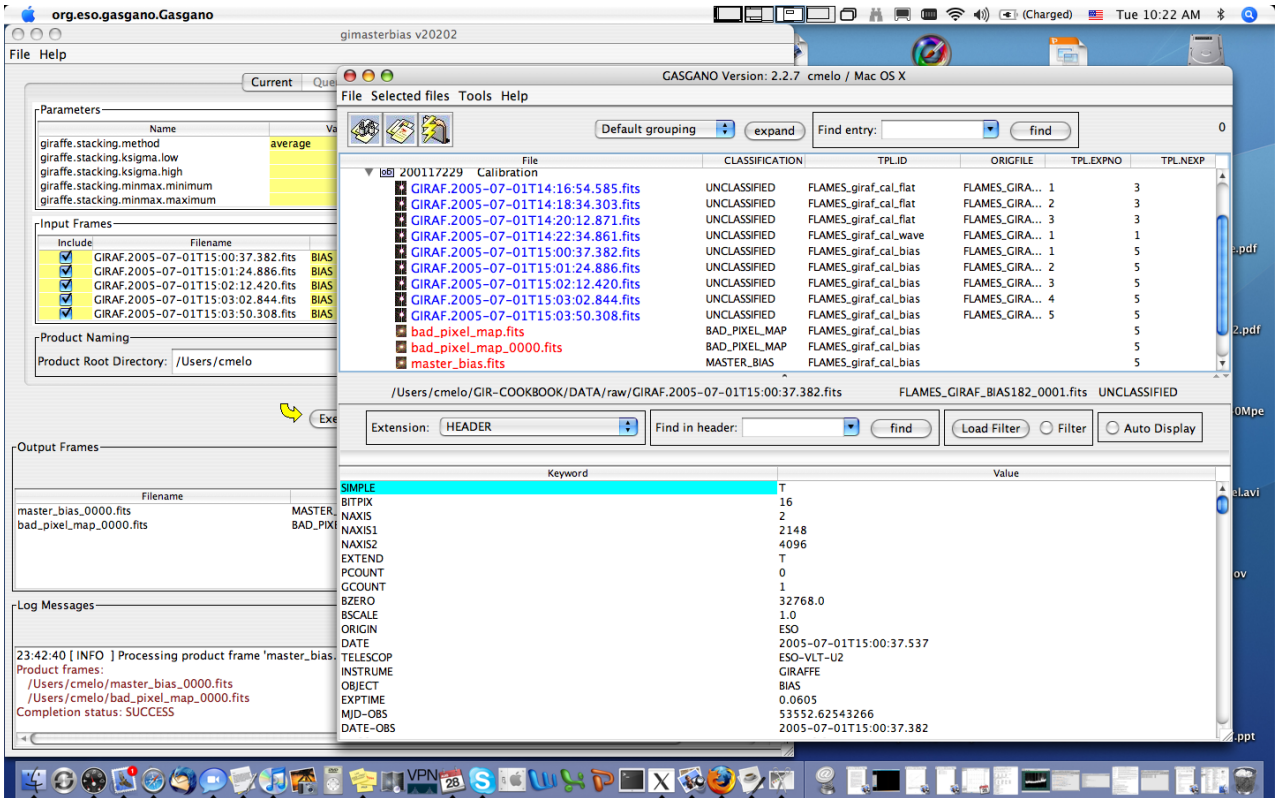
---

[9]Please consult the EsoRex manual

Figure 15: Aperçu of the next release of the GIRAFFE pipeline running on a Mac OS machine.

A   Preparing your sof file: a first example

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`.

```
limari:reduced 12> 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
```

B   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 were reported to compile without problems on Mac OS machines (e.g., SINFONI and UVES). The present version of the GIRAF pipeline does not work on Mac OS. A new release will be available soon (already running in Paranal). This version compiles smoothly and can be used via esorex and/or gasgano (Figure 15.

$$- - - \mathbf{oOo} - - -$$