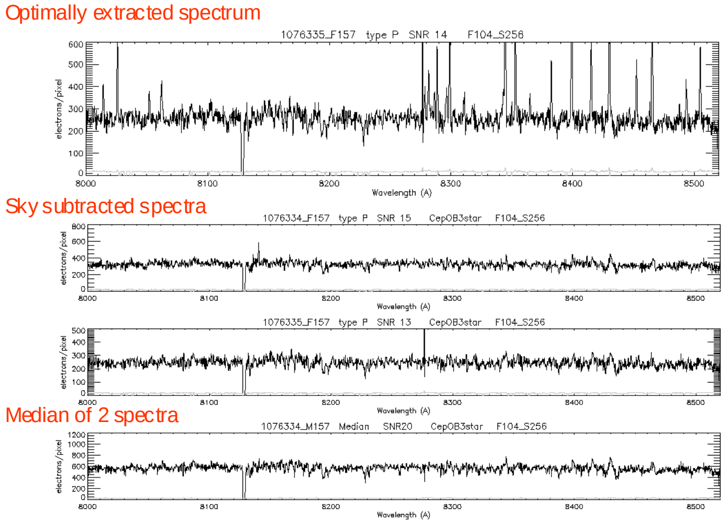
AF2/WYFFOS Data Reduction Pipeline - User manual
(version 1.0, issued 01-10-11)

1. Introduction

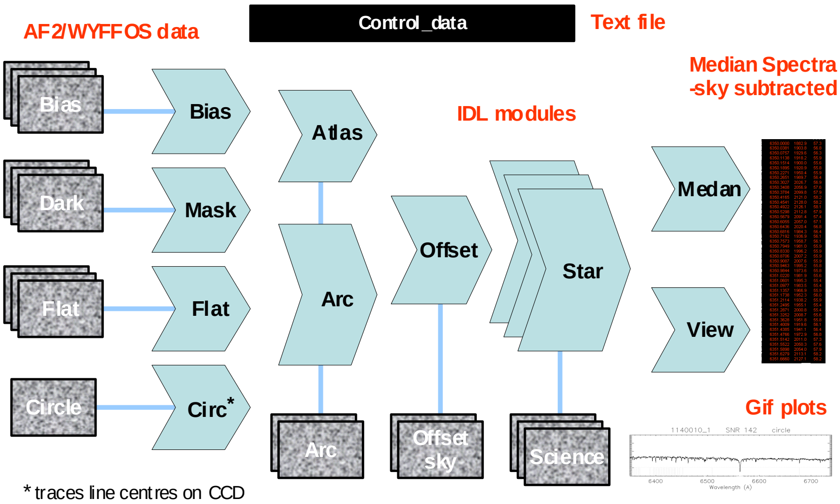
The AF2/WYFFOS data reduction pipeline is a modular software package that optimally extracts and calibrates spectra from the raw CCD science and calibration images recorded using the AF2/WYFFOS spectroscope to provide one dimensional, sky subtracted, wavelength calibrated, median spectra for individual targets referenced by fiber number (see Fig.1)

Fig.1: Typical output spectra



The pipeline is written in IDL and comprises a series of modules that correspond to the principle steps of the data extraction process (see Fig.2).

Fig. 2: Pipeline process



Parameters controlling the operation of the pipeline are listed in an ASCII text file named control_data.txt. This contains the names of the target and calibration data files, user specified parameters controlling the form and range of the output spectra and general parameters relating to the AF2/WYFFOS set up. Other parameters used in the extraction process are read directly from the AF2/WYFFOS data file headers.

The purpose of this note is to guide the user through the process of running the pipeline to reduce a set of one or more observations of a group of targets recorded over a relatively short time period such that all science frames may be extracted using a single set of calibration data files. This set of science and calibration data files is referred to as a reduction block (RB).

Fig.3: Main steps of data reduction

Step 1	Set up directories and load raw data Create/edit control files.
Step 2	Run calibration modules Bias, Mask, Flat, Circ.
Step 3	Run Atlas module to set arcline data (if required). Run Arc to extract arc lamp spectra and perform wavelength calibration.
Step 4 (if required)	Run Offset module to extract offset sky spectra, transform spectra onto a common wavelength base, and find median of offset sky observations.
Step 5	Run Star module to extract target spectra. Transform spectra onto a common wavelength base, and subtract median sky spectra (if required).
Step 6	Run Medan to create median of all observations in an RB. Run View to review output spectra in different formats.

The main steps required to set up and run the pipeline, listed in Fig. 3 are described in this note. A map of the pipeline software is given in Appendix 1.

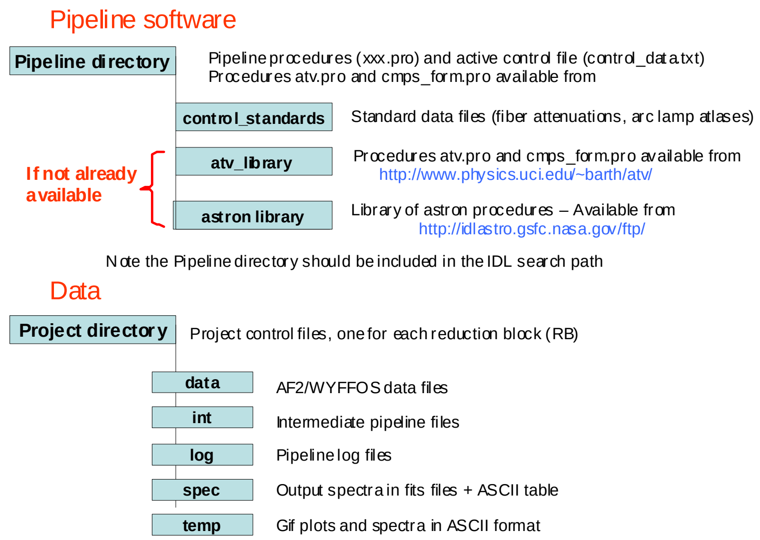
2. Installation

The directory structure required to run the pipeline is shown in Fig.4. The pipeline directory contains the IDL pipeline procedures and the active pipeline control file (named control_data.txt).

The subdirectory control_standards contains standard data files that are accessed by the pipeline software. One file is a default list of the relative fiber attenuations (see section 6). Others are atlas files for various arc lamp configurations (see section 9). Additional data files may be added to this subdirectory by the user.

Subdirectories containing the “astron” library of IDL subroutines and the interactive display tool ATV should be added if these are not already available on the user’s IDL installation.

Fig.4: Directory structure



A separate project directory is required for each distinct project. The top level directory contains the control files associated with a set of observations (one for each RB) plus 5 subdirectories contain input and output data. The name of the project directory is unique and is held in the control file. Default names of the subdirectories are shown in Fig.4. These can be changed in the control file.

The subdirectory data/ contains all the science and calibration files associated with the RBs within a project. These include copies of bias or dark files, recorded prior to the observing run, which are referenced by the RB. Data files follow the standard format and naming convention of AF2/WYFFOS data files.

3. Control files

Parameters controlling the operation of the pipeline are contained in the control file. This is an ASCII text file unique to a particular RB. The control file comprises a list of parameter values followed by a semi colon and the variable name. Appendix 2 shows a typical control file together with a description of the function of each parameter in the pipeline software.

The structure of the ASCII control file is shown in Fig. 5. There are three types of data

- Data that is expected to vary between RBs. This comprises key parameters, options and the data file numbers.
- Data that varies according to the AF2/WYFFOS set up. These are principally parameters relating to wavelength calibration.
- Quasi-standard parameters that do not generally vary from the default values given in the reference control file provided by the ING.

Fig.5: Control parameters

Key parameters (9)	Unique to an RB
Options (5)	Unique to an RB
Settings (16)	Quasi-standard parameters
Subdirectories (5)	Quasi-standard parameters
Limits (7)	Quasi-standard parameters
CCD data (21)	Quasi-standard parameters
Arc data (12)	Unique to AF2/WYFFOS set up
Data files	Unique to an RB
Arcline data	Unique to AF2/WYFFOS set up

Key data

OBNAME	Name of RB (unique within project directory)
PDIR	Name of project directory
FIBERID	# for single fiber or 999 to extract spectra for all fibers
DFIBER1	Fiber No. of 1st fiber lost in gap between two halves of CCD
DFIBER2	Fiber No. of 2nd fiber lost in gap between two halves of CCD
SKYOSET	0:use in frame sky subtraction, 1:use offset sky subtraction
SKYTYPE	0:no sky subtraction, 1:linear, 2:scaled, 3:scaled and clipped
WLMIN	Lower cut off for common wavelength base
WLMAX	Upper cut off for common wavelength base

Options

FIBRATTN	0:no attenuation, 1:calculated, 2:from std file, 3:rom local file
PROFILE	0:use flat file profile for extracation, 1:use top hat profile
FLATTYPE	0:normalise output to flat spectrum, 1:no normalisation made
SCATSUBT	0:no subtraction of scattered light, 1:subtract scattered light
WLTYPE	0:use log scale for standard wavelength, 1:use linear scale

A control file should mirror the reference control file containing a complete list of control parameters in the correct order with no blank lines. Comments following a semi-colon are discarded when the control file is read.

Data files are listed by type in nominal order of use, each block containing the number of files of that type followed by a list of their file numbers, typically a 7 digit number (see Appendix 2). Note the file numbers need not be contiguous although in practice the bias and flat files often are. A full description of the arc and arcline data is given in section 9.

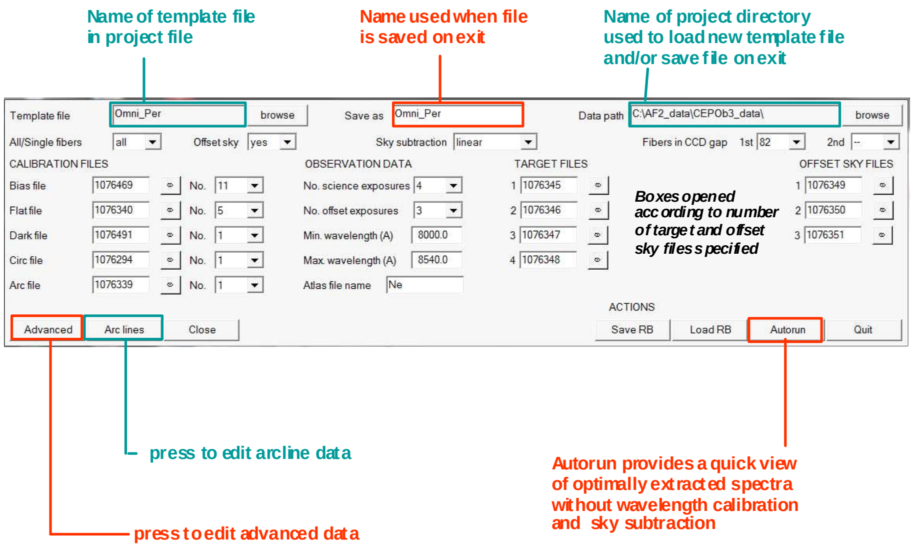
4. GUI for control data entry

Control files can be created and edited using a graphical user interface (pipeline procedure GUI). This procedure reads the active control file (i.e. the one stored in the pipeline directory), and displays selected items in the form of a basic GUI allowing the user to edit key parameters, select common options and set data file numbers in order to define a new RB.

The minimum information that must be entered by the user is the control file name and directory together with details of the bias, flat and target files. This is sufficient to run the pipeline in auto mode (without wavelength calibration and sky subtraction). Normally the pipeline extracts all fibers allocated to targets in the AF2 setup file. A single target can be selected for extraction by specifying the fiber number.

To use the pipeline proper the user must give details of the arc file(s) and the type of sky subtraction required. If offset sky is specified then the number and names of the offset sky frames must be added. The user may also specify a common wavelength range for extracted spectra. If these limits are not known then they can be generated by running the Altas module in which case the name of an atlas file is required.

Fig 6. GUI used to edit control files



Details of dark files must be specified in order to run the Mask module. It is also recommended to specify a lamp or sky flat file for the Circ module with uniformly illuminated fibers in circle set up. This is used to relate spectral lines on the CCD to fiber numbers and to identify fibers lost in the gap between the two halves of the CCD. The numbers of these lost fibers can be entered in the GUI or, if unknown, generated using the Circ module.

The lower left hand buttons are used to expand the GUI to display more detailed lists of control parameters in the form of editable tables. (see Fig. 7 & 8). These tables can be edited by entering a new value followed by <CR>. The “advanced” button displays the first 63 control parameter providing full control of the pipeline options and settings.

The “arcline” button displays a further 12 parameters that set parameters for wavelength calibration and the display of the arc lamp atlas together with a table of the pixel positions and wavelengths for individual calibration lines. Arcline data for standard RBs can be copied from an existing control file. Where the AF2/WYFFOS set up has been changed requiring a new set of arcline data this can be generated using the separate interactive procedure, Atlas, described in Section 9. The lower right hand buttons initiate the following actions:

Save RB saves the current data set to the specified file name and project directory.
Load RB saves the current data set and loads a copy as the new active control file.
Autorun saves the current data set, loads it, and runs the pipeline in auto mode.

Fig 7. Expanded GUI showing advanced settings

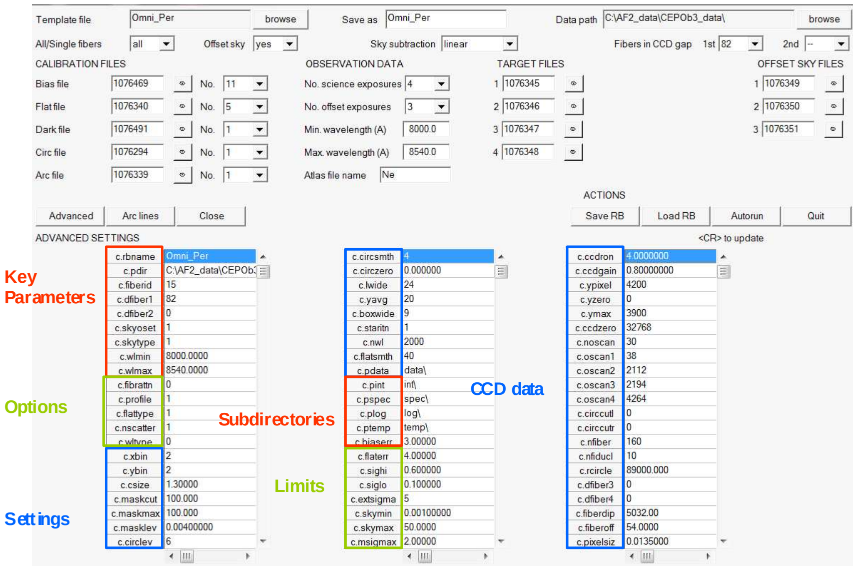
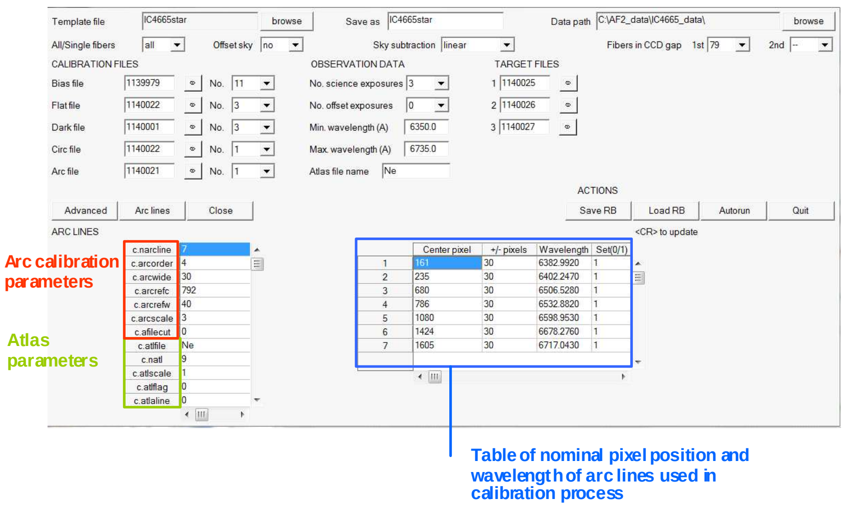


Fig 8. Expanded GUI showing arc line data



The graphical user interface also provides the facility to view the header data and raw CCD images of the AF2/WYFOS data files. Pressing the eye symbol alongside the data file name causes the selected file to be displayed in ATV (see Fig. 9). ATV is an IDL program for display and analysis of astronomical images written and maintained by Aaron Bath. It is similar in appearance and has many of the functions of the SAO image utility ds9.

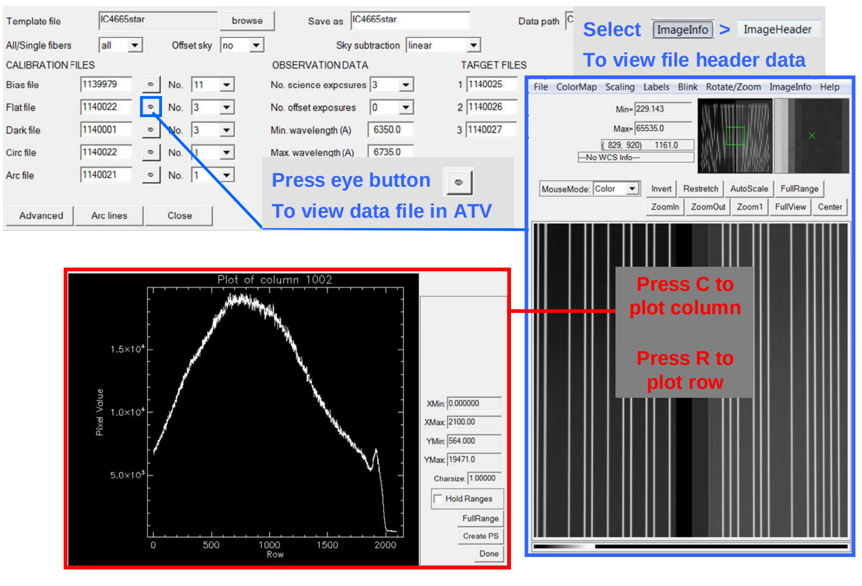
The example in Figure 9 shows the ATV display of a Flat file. The image (in units of ADU per pixel) can be manipulated interactively in the ATV window to examine particular pixels. Pressing C plots the column containing the selected pixel and pressing R plots the corresponding row. Selecting image info followed by image header opens a new window displaying the fits file header. The full set of commands is described in ATV's help utility.

Further information and the program source code are available from Aaron Bath's excellent webpage [www.physics.uci.edu/~barth/**atv**/](www.physics.uci.edu/~barth/atv/). The utility was first presented in reference Barth, A. J. 2001, in ASP Conf. Ser., Vol. 238, Astronomical Data Analysis Software and Systems X, eds. F. R. Harnden, Jr., F. A. Primini, & H. E. Payne (San Francisco: ASP), 385.

5. Pipeline software

Fig. 2 shows the top level pipeline procedures referred to here as modules. Modules can be run individually from the IDL command line, in the sequence shown in figure 2, or run in batch mode (see Section 11). The pipeline can also be run in auto mode directly from the base GUI. This mode is used to take a quick look at the target spectra. It runs a reduced version of the pipeline, without wavelength calibration or sky subtraction, requiring only the minimum number of control parameters to be specified by the user (see Fig. 6).

Fig 9. Using ATV to view data input files



Modules have no input parameters (although some have options). Instead, the first action of these top level procedures is to read the active control file. This determines the subsequent operation of the module. Generally speaking the function of a module is to read a set of data files from disk, carry out some calculations and write the results as a new data file to disk. Modules also produce reports and error messages to a log file. The exception is the Atlas module which is used to modify arcline data held in the control file.

Figure 10 shows the different types of data file, their naming convention and their subdirectory. The number 1234567 refers to the number of the lead AF2/WYFFOS CCD image used to generate the data file. Where a file represents the average of several input files the output file takes the number of the leading (first) input file.

All fits files generated by the pipeline have a common header format comprising
1) parameters read from the AF2/WYFFOS file header for the CCD image
2) parameters read from the AF2/WYFFOS file header for each fiber
3) a copy of the RB control file

Details of the pipeline fits file header are given in Appendix 3.

When a control file is loaded, a new log file is created in the log/ directory with the name xxx_log.txt where xxx is the name of the RB (the first item in the control file). The log file is unique to an RB and records all reports produced by the pipeline until a new control file of the same name is loaded at which point it is overwritten.

Fig. 10 Data files

FILE NAME	HEADER	CONTENT	DIRECTORY
RAW DATA			
r1234567.fit	AF2/WYFFOS	Raw Giraffe CCD image	data/
INTERMEDIATE IMAGES			
b1234567.fit	Pipeline	Median bias image	int/
m1234567.fit	Pipeline	Mask image	int/
f1234567.fit	Pipeline	Median flat image	int/
INTERMEDIATE ARRAYS			
c1234567.fit	Pipeline	Array of co-ordinates of line centres	int/
a1234567.fit	Pipeline	Array of arc spectra versus local wavelength	int/
s1234567.fit	Pipeline	Array of science spectra versus local wavelength	int/
t1234567.fit	Pipeline	Array of offset sky spectra versus local wavelength	int/
OUTPUT SPECTRA			
p1234567.fit	Pipeline	Extracted spectra on common wavelength base	spec/
q1234567.fit	Pipeline	Median spectra on common wavelength base	spec/
o1234567.fit	Pipeline	Offset sky spectra on common wavelength base	spec/
u1234567.fit	Pipeline	Median spectra vs pixel number (auto mode)	spec/
GRAPHICAL OUTPUT			
1234567_F123.gif	None	Plot of spectra versus wl for a single science spectra	temp/
1234567_M123.gif	None	Plot of spectra versus wl for a median science spectra	temp/
1234567_M123.ps	None	Encapsulated Postscript plot of spectra versus wl	temp/
ASCII FILE OUTPUT			
p1234567.txt	None	ASCII table of fiber data for extracted science file	spec/
r1234567.txt	None	Copy of AF2/WYFFOS file header for science file	temp/
s1234567.txt	None	Copy of Pipeline file header for science file	temp/
1234567_M123.txt	None	File of wl, spectra & err for a median science spectra	temp/

Modules produce reports and error messages which are stored in the log file specific to the RB. There are three types of error message:

- WARNING – no action required (typically warning of high data noise).
- ERROR – requires correction but the pipeline process can continue.
- FATAL ERROR – requires the pipeline process to be stopped.

When a fatal error occurs a stop statement is executed in the active procedure.

Modules also produce volatile reports to the log directory. These are text files or gif images that are overwritten the next time the module is run for any RB in the same project directory. These files are useful to help diagnose quality problems.

6. Calibration modules

There are four calibration modules in the pipeline process which are run prior to extracting spectra (see Fig, 2). These are Bias, Mask, Flat and Circ.
In principle the Bias and Mask files need only be run if the relevant control parameters and/or data file numbers have changed between RBs. However, their run times are short, so it is good practice to rerun all modules for each RB.

Table 1 gives functional details of the Bias module. The module displays an image of the bias file which is also saved to the log directory (see Fig 11).

The bias file is used to debias all raw data images read by the pipeline (i.e. dark, flat, arc and science images). When images are debiased the average signal level in the overscan regions is used to correct for any change in dc bias level over time. The two halves of the CCD image are corrected separately.

Table 1 Bias Module

Procedure	**Bias**
Function	Check the input file type is correct Create a median bias file
Input	Raw data bias files
Output	Biasfile (b1234567.fit) containing the median bias image
Reports to log	'RUNNING PROCEDURE: Bias' New Bias file created:b1234567 Saving copy of Bias.gif
Report files	Copy of median bias image (bias.gif)
Error	FATAL ERROR: readout binning does not match control data FATAL ERROR: incorrect file type, file type is xxx WARNING: high noise in bias images ### adu

Table 2 gives details of the Mask module. The mask image is set to 0 when the dark current in a pixel exceeds a specified level. In normal mode the cut-off level is given by the control parameter c.maskcut. In auto mode the cut of level is calculated to mask a specified fraction of pixels given by the control parameter c.masklev. The module displays a plot of the fraction of masked pixels versus the cut-off level and an image of the mask file (see Fig .11).

Fig.11 Calibration images

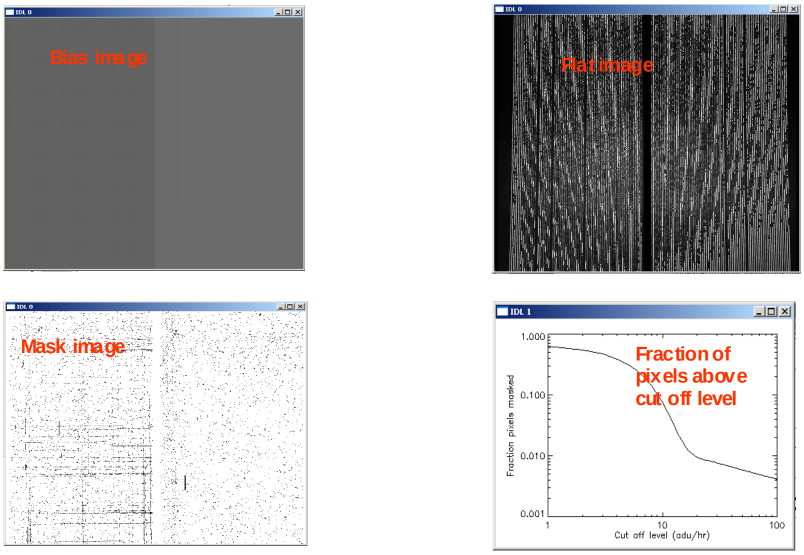


Table 2 Mask module

Procedure	**Mask**
Function	Check the input file type is correct In case of multiple dark files determine the minimum dark current per pixel Create mask of pixels that show a dark current above the specified cut-off level
Options	Mask,/AUTO - Mask a fraction of pixels rather than a fixed dark current per pixel
Input	Raw data dark files
Output	Maskfile (m1234567.fit) containing mask image
Reports to log	RUNNING PROCEDURE: Mask Masking #.##% of pixels with > ##.## adu/hour New mask image created:m1234567 using # files Saving copy of Mask.gif
Report files	Copy of mask image (mask.gif) Table of fraction of masked pixels versus cut-off level (mask.txt)
Error	WARNING: no dark file specified FATAL ERROR: incorrect file type, file type is xxx FATAL ERROR: readout binning does not match control data ERROR: mask cut greater than specified maximum level

Table 3 gives details of the Flat module. This module calculates the median of a set of debiased flat images. Individual flats are scaled according to their mean value before the median value is taken to compensate for any global change in signal level between flats. Warnings are given if the (smoothed) peak count per pixel of the Flat image is outside limits specified in the control file (parameters c.sighi and c.siglo).

Table 4 gives functional details of the Circ module. This module traces the x-pixel position of the center of the spectral line produced by each active fiber as a function of y-pixel. The input files are a set of uniformly illuminated images (observations of the twilight sky or flats) recorded with the fibers in a circle configuration. A series of smoothed cuts through this image are used to locate the line centres (see Fig. 11). The degree of smoothing and the cut level are set by control parameters c.circsmth and c.circzero.

The module identifies the fiber number associated with the each line on the CCD image by counting the number of peaks from each side of the CCD to the center taking account of dead fibers identified in the AF2/WYFFOS header. In this way it identifies those fibers when the signal is lost in the overscan region between the two halves of the CCD (see Fig 12).

Table 3 Flat Module

Procedure	**Flat**
Function	Checks the input file type, binning and fiber allocation are correct Create a median flat image
Input	Raw data flat files
Output	Flatfile (f1234567.fit) containing median flat image
Reports to log	RUNNING PROCEDURE:Flat New Flat file created:f1234567 median of # file Saving copy of Flat.gif
Report files	Copy of median flat image (flat.gif)
Error	FATAL ERROR: incorrect file type, file type is xxx FATAL ERROR: readout binning does not match control data ERROR: central wavelength not compatible with target file r1234567 FATAL ERROR: allocation of active fibers incompatible with target file r1234567 WARNING: high noise in flat files, ### adu ERROR: Saturation of flat image (>99% full scale) WARNING: High flat image, ## of full scale WARNING: Low flat image, ## of full scale

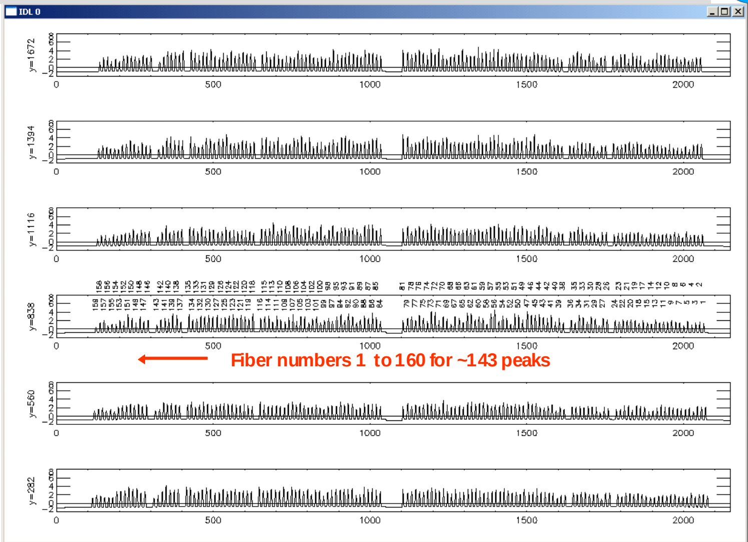
In standard mode the module halts the pipeline if the fibers identified as being in the gap do not match those specified in the header (c.dfiber1 and c.dfiber2). When the module is run in auto mode the values of c.dfiber1 and c.dfiber2 in the control file are overwritten with the newly identified values. In this case the annotated plot should be checked by eye.

Table 4 Circ module

Procedure	**Circ**
Function	Check input file type and binning are correct Trace centreline of active fibers across raw data image Creates an offset image of the Circ file Sets values of relative fiber gains in file header
Options	Circ,/AUTO identifies fibers located in gap and overwrites values in control file
Input	Raw data file for a uniformly illuminated image with fibers in circle set up
Output	Circfile containing offset image and centerline data (c1234567.fit)
Report to log	RUNNING PROCEDURE:Circ ### lines identified with # fiber(s) in central gap, number(s) ## and ## Fiber gains set to 1 / calculated from mean line counts / read from file xxx New mask image created:m1234567.fit using # dark files Saving copy of Circ.gif
Report files	Plot of image profiles showing the fiber number associated with each line (Circ.gif) Table of pixel positions of peaks in image profile for each active fiber
Error	WARNING :no circ file specified - using first flat file FATAL ERROR: incorrect file type, file type is xxx FATAL ERROR: readout binning does not match control data FATAL ERROR: incorrect width of CCD line profile, c.lwide/c.xbin must be even ERROR: saturation of circ image (>99% full scale) WARNING: high circ image, ## of full scale WARNING: low circ image, ## of full scale FATAL ERROR: unable to identify peaks correctly in image profile FATAL ERROR: variable number of peaks found in image profile FATAL ERROR: error in header data, incorrect fibers identified with gap FATAL ERROR: file fiber_attn.txt not found FATAL ERROR: list of fiber gains does not hold values for all valid fibers FATAL ERROR: cannot select single fiber if Circfile is also a target file

If there is no file available with the fibers in the circle configuration then the Circ module uses the first flat file in order to identify line centres (for illuminated fibers) in auto mode. This process is less reliable since it is more difficult to check the allocation of lines to fiber numbers by eye. If the line associated with one fiber is half in/half out of the central gap such that it cannot be correctly extracted then x-pixels adjacent to the central gap can be masked by setting non-zero values of parameters c.circcutl and c.circcutr.

Fig. 12 Circ module plot

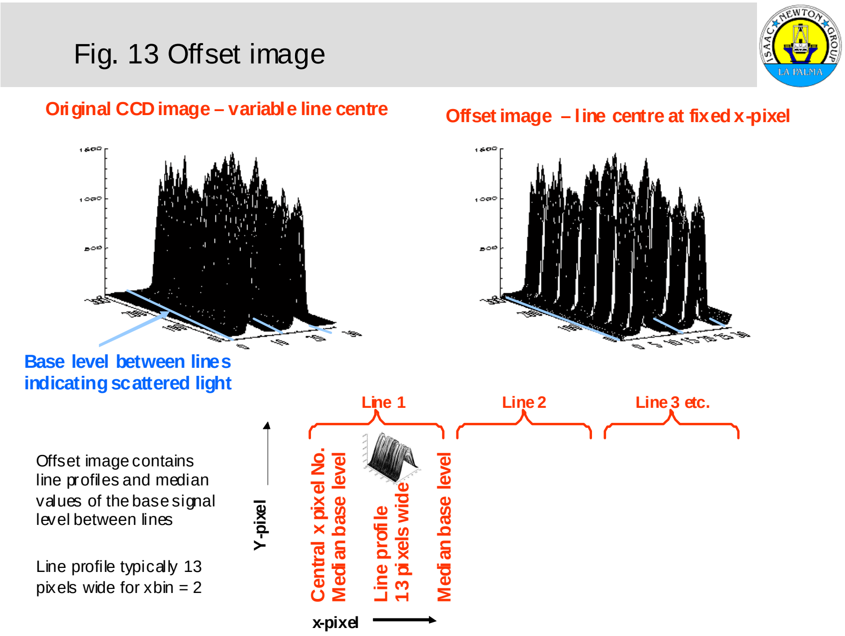


The line centres established from the profiles shown in Fig. 12 are interpolated to give x-pixel position of each active fiber as a function of y-pixel position. These values are used to generate an offset image where the line centre of individual fibers is positioned at a fixed x-location, to the nearest half pixel (see Fig. 13). The median value of the signal level midway between lines is evaluated and stored alongside the offset image (see Fig. 13). This data is later used to compensate for the effect of scattered light in the CCD. The offset array contains:

1. The original x position of the line centre on the CCD image (for each y value).
2. The raw signal of the line spectra extending ±6 pixels (for x2 binning) from the centre line in the x direction.
3. The median signal between lines estimated from a rolling median in the y direction of the pixel values halfway between lines.

The final function of the Circ module is to set the values of the relative attenuation of the fibers which are part copied the file header data, f (see Appendix 3). The source of the attenuation values depends on the control parameter c.fibrattn;

0) Fiber attenuations are set uniformly to 1.
1) Attenuations are calculated from the mean line counts in the Circfile image.
2) Attenuations are read from file fiber_attn.txt in directory control_standards/.
3) Attenuations are read from file fiber_attn.txt in the project directory log/.



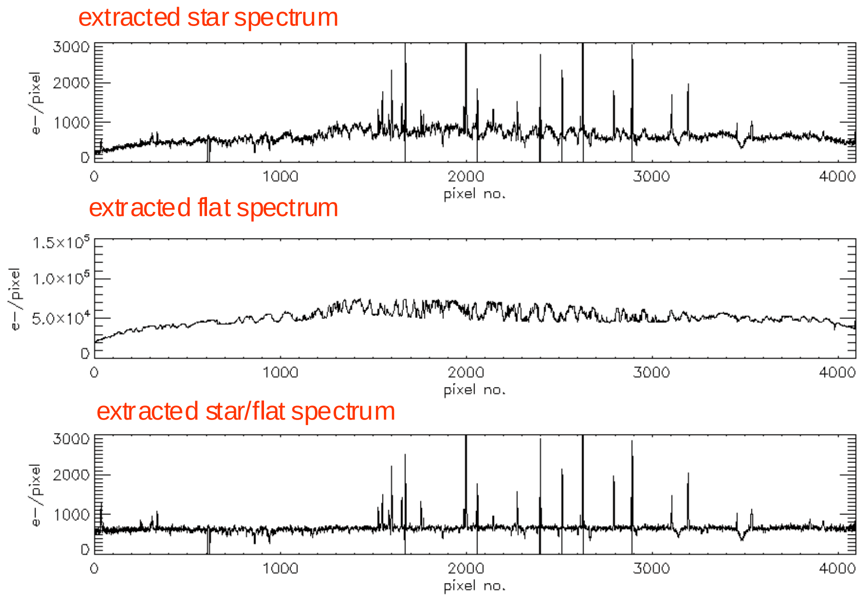
7. Optimal extraction

The process used for optimal extraction follows the algorithm given by Horne (1986).This applies a non-uniform weight to pixels in the extraction sum to minimize statistical noise whilst preserving photometric accuracy (reference Horne K., 1986, PASP, 98, 609). Horne showed that the ideal weighting function is the normalised spatial profile of the image (in the x direction) at a given wavelength. For the pipeline the profile is estimated by taking a boxcar average of the flat image in the wavelength direction.

Using this smoothed flat profile the total electron count at each pixel in the wavelength direction still depends on the weighted mean pixel response across the line. To compensate for variations in pixel response, spectra are extracted both for the measured image (which can be a science, flat or arc frame) and for the flat image, their ratio being taken to give the final science spectrum. This ratio is scaled to the mean of the flat spectrum, so that the final science spectrum gives a measure of the total number of electrons captured at each pixel position, albeit weighted according to the modulation of the spectral intensity of the flat lamp (see Fig. 14).

The algorithm used to optimally extract spectra from individual images is shown in Fig. 15. The principle input files are an offset array of the target spectra, $S_{x\lambda}$ and flat image, $F_{x\lambda}$, both scaled according to the instrument gain in electrons per adu, together with a mask image, $M_{x\lambda}$, where x is the pixel number normal to the wavelength direction and λ is the pixel number in the wavelength direction. The variance of the measured spectral image is calculated as $V_{x\lambda} = (S_{x\lambda} + RON)$.

Fig. 14 Intermediate spectra



The first step of the process in Fig. 15 is the calculation of the spatial profile, $P_{x\lambda}$ based on the smooth flat image. In the case of low resolution spectra the flat image may be zero (or less due to noise in the debiasing process) at the blue end of the spectra. In this case the profile is set to zero and the affected pixels are masked (i.e. set to zero) in the output spectra.

Fig. 15 Optimal extraction

FLOW CHART

(1) Evaluate spatial profile

(i) $P_{x\lambda} = F_{x\lambda} / \sum_{X\,min}^{X\,max} F_{x\lambda}$ and $P_{x\lambda} > 0$

smoothed over N pixels in wavelength

(2) Optimal extraction

(i) $s_\lambda = \sum_{X\,min}^{X\,max} M_{x\lambda} P_{x\lambda} S_{x\lambda} / V_{x\lambda} \Big/ \sum_{X\,min}^{X\,max} M_{x\lambda} P_{x\lambda} P_{x\lambda} / V_{x\lambda}$

(ii) $v_\lambda = \sum_{X\,min}^{X\,max} M_{x\lambda} P_{x\lambda} \Big/ \sum_{X\,min}^{X\,max} M_{x\lambda} P_{x\lambda} P_{x\lambda} / V_{x\lambda}$

(iii) $m_\lambda =$ **0 where** $\sum_{X\,min}^{X\,max} M_{x\lambda} P_{x\lambda} > ½$ else $m_\lambda = 1$

(3) mask pixels showing excess (>5σ) variation

(i) $D_{x\lambda} = S_{x\lambda} - s_\lambda P_{x\lambda}$ from x_{min} to x_{max}

(ii) σ_λ = standard deviation ($D_{x\lambda}$)

(iii) $M_{x\lambda} = 0$ where $Dx\lambda > 5\ \sigma_\lambda$

NOTATION

$S_{x\lambda}$ = Measured spectrum image
$V_{x\lambda}$ = Variance in spectrum image
$F_{x\lambda}$ = Associated flat spectrum image
$M_{x\lambda}$ = Mask image
$P_{x\lambda}$ = Spatial profile for optimal extraction
s_λ = raw extracted spectra
v_λ = raw variance
m_λ = mask vector
$D_{x\lambda}$ = deviation from predicted profile

Process for optimal extraction taken from Horne (1985)

The next step evaluates the extracted spectrum, s_λ, as a function of pixel number. This is the total of the electron counts per pixel measured across the image of the spectrum at each wavelength position weighted according to the spatial distribution function P and the variance V.

While the algorithm corrects for the effects of masked pixels in the CCD image, there is a limit to how much of the image is masked at a given y pixel position before the extracted spectrum at that point is no longer valid. The limit applied here is that if more than 50 per cent of the CCD image (weighted according to the spatial profile) is masked then the corresponding pixel is masked from the extracted spectrum.

In the third step the mask array, $M_{x\lambda}$, is updated to take account of excessive noise at individual pixels of the line spectrum on the CCD image introduced typically by cosmic ray hits To do this, the difference between the measured signal and the signal predicted from the product of the extracted spectrum and the spatial profile is calculated. If the difference for a given pixel exceeds (typically) 5σ then the mask array is set to zero for that pixel. The extraction process is then repeated.

The profile used in the extraction process is set by the option c.profile:

0) A top hat profile is used producing non-optimal extraction
1) The spatial profile is found from the flat image for optimal extraction.

The effect of scattered light is reduced by subtracting the local background signal level which is estimated by linearly interpolating the median signal recorded between adjacent lines (see Fig. 13). This background level is subtracted from the profiles of both flat and target spectra prior to optimal extraction. The option to reduce the effect of scattered light is set by parameter c.nscatter:

0) No compensation is made for scattered light
1) The background level due to scattered light is subtracted before extraction.

<u>8. Wavelength calibration</u>

Table 4 gives functional details of the Arc module. This module optimally extracts the arc spectra as a function of y-pixel position then uses these intermediate spectra to determine the wavelength calibration.

When two arc files are specified in order to reference lines from two arc lamps simultaneously the resulting arc spectrum is a combination of the intermediate spectrum from arcfile 1 for the first N pixels and the corresponding spectrum from arcfile 2 for the remaining pixels (where the value of N is set by control parameter c.alinecut).

The Arc module uses a predefined table of arcline data (part of the control data) to identify the approximate y-pixel location and exact wavelengths for a set of well-separated unsaturated lines in the arc spectra (see control data in Fig. 8). The precise position of the peaks (in pixels) is then found by fitting Gaussian profiles over a short section of the spectrum (typically ±15 pixels) encompassing each peak. The results are used to determine the coefficients of a cubic polynomial of wavelength against pixel number for each active fiber. These coefficients are then used to provide a wavelength scale for the each fiber which is valid for the current RB.

IMPORTANT NOTE : the table of arcline data shows the y-pixel position (and range) of selected lines in the arc spectra in terms of reduced pixels i.e the y-pixel value of the binned CCD image. So if y-binning is changed between observations then y-pixel values in the arcline table must be scaled accordingly.

Table 5 Arc module

Procedure	**Arc**
Function	Check input file type, binning and fiber allocations are correct Create intermediate file of optimally extracted spectra Calibrate wavelength against pixel position for active fibers
Options	Arc,/NOEXTRACT - skips optimal extraction stage Arc,/ATLAS - runs procedure Atlas to create/edit arcline data
Input	Raw data image(s) of the arc lamp
Output	Arcfile containing extracted arc spectra & wavelength per pixel (a1234567.fit)
Report to log	RUNNING PROCEDURE:Arc For file number 1234567 Extracting file r1234567 (repeat for each arc file) FLAT Iteration 0 masked points =### ARC Iteration 0 masked points = ### For file number 1234567 New Arc file created:a1234567 Lamp Ne max delta for fit 0.### pixels (see file arc.txt) Common wavelength range ####.# to #####.# A Arcline wavelength mean_delta_fit rms_delta_fit 1 ####.### #.### #.### (repeat for each arcline)
Report files	Image of five plots showing stages of the calibration process (Arc.gif) Table of wavelength limits and max deviation of calibration lines per fiber (Arc.txt)
Error	FATAL ERROR: incorrect file type, file type is xxx FATAL ERROR: readout binning does not match control data ERROR: central wavelength not compatible with target file r1234567 FATAL ERROR: allocation of active fibers incompatible with target file r1234567 ERROR: saturation of extracted image (>99% full scale) WARNING: high extracted image, ## of full scale WARNING: Flat signal zero over part of spectra - a total of ## bad pixels masked FATAL ERROR: file of arc spectra not found, invalid option \noextract ERROR: specified minimum wavelength greater than actual maximum ERROR: specified maximum wavelength greater than actual maximum

The Arc module produces graphics at stages of the calibration process which serve as a visual check that the process is working correctly (see Fig. 16).

Plot (a) shows an overlay of the arc spectra after applying an approximate geometric offset per fiber on the y-axis. Also shown are two markers identifying the arcline subsequently used to derive more precise values of the y-axis offsets. The mean value and spacing of these markers are set by control parameters c.arcrefc and c.arcrefw.

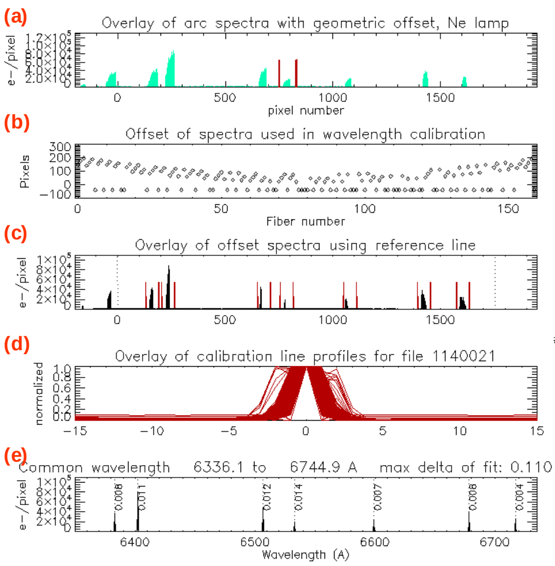
Plot (b) shows the y-axis offsets as a function of fiber number across the CCD calculated from the y-pixel position of the strongest arcline between the markers in Plot (a). Inactive fibers are indicated by negative values.

Plot (c) shows an overlay of the arc spectra using the calculated values of y-offset. Also shown are markers identifying the arclines that will be used for the wavelength calibration. The mean value and spacing of these markers is set by the control parameters in the arcline table (see Fig. 8).

Plot (d) is a quality control plot. It shows an overlay of the local sections of the (normalise) peaks in arc spectra that are fitted with a Gaussian profile. These should all show a maximum centered at zero.

Plot (e) shows an overlay of the calibrated spectra. Spectral lines used in the calibration procedure are indicated by a dotted line together with the rms value of the deviation between the measured peak and the polynomial fit for each fiber. Also shown are the limits of the common wavelength range for the active fibers.

Fig 16 Reports from Arc module



Report to log file

#Arcline	wavelength	mean_delta_fit	rms_delta_fit
1	6382.9920	-0.000	0.009
2	6402.2470	0.000	0.012
3	6506.5280	0.002	0.010
4	6532.8820	0.000	0.011
5	6598.9530	-0.004	0.006
6	6678.2760	0.004	0.007
7	6717.0430	-0.002	0.003

Report to Arc.txt

#Fiber	min_wavelength	max_wavelength	delta_fit_lines
0	6307.075	6755.663	0.005
1	6301.650	6750.317	0.005
2	6296.366	6745.083	0.006
3	6309.328	6758.063	0.004
4	6303.998	6752.745	0.003
5	6298.522	6747.351	0.003
6	6311.479	6760.305	0.000
7	6306.140	6755.016	0.100

Overall statistics showing the quality of the fit of the polynomial functions to lines in the arc spectra are saved to the log file while more detailed information on the quality of fit for individual fibers is given in file Arc.txt (see Fig. 16). Note the arc module calls the IDL procedure GAUSSFIT which gives the message "Program caused arithmetic error: Floating underflow". This message is OK in this context.

9. Arcline data

The Altas module is used to create and edit the set of arcline data for wavelength calibration. Functional details of the Atlas module are shown in Table 6. It can be run independently or called from the Arc module using the option Arc,/atlas.

Table 6 Atlas module

Procedure	**Atlas**
Function	A GUI that is used to create and edit the arcline data for wavelength calibration Select the line of the arc spectra used to calculate y-offsets Select the Atlas file name and common wavelength range for the arc spectra Select the wavelength and pixel position of lines used for wavelength calibration Calibrate wavelength against pixel position for active fibers
Input	Intermediate arc spectra (a1234567.fit) Contol_data.txt - file containing current control data
Output	Contol_data.txt - modified control file with changes made to arcline data
Report to log	RUNNING PROCEDURE:Atlas Overwriting arcline data for control file ### (if selected) New arcline calibration data saved to control file (if selected) Quit arcline calibration without changing control file (if selected)
Error	FATAL ERROR: Atlas file: xxx not found ERROR: Atlas shows no lines in specified wavelength range'

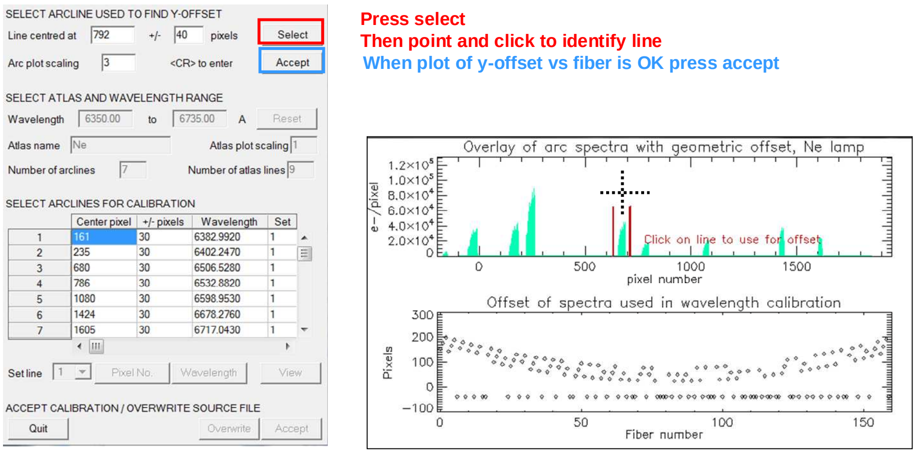
The module displays a GUI that takes the user through the steps required to create a set of arcline data, These are:

1) Select the arcline used to find the y-offsets
2) Select an arc lamp atlas file and the wavelength range
3) Select arclines used for wavelength calibration
4) Accept the calibration and the update control file(s)

Fig. 17 shows the GUI and the graphic display used to set markers that identify an isolated arcline near the midpoint of the spectra that will be used to determine the y-offsets of individual fibers. The center pixel and spacing of the markers can be entered directly into the GUI. Alternatively the center pixel can be picked by pointing and clicking on an arcline in the graphic display.

Fig. 18 shows the next stage of the process. Here the upper plot of the graphic display shows an overlay of the offset arc spectra together with markers identifying the lines to be used for wavelength calibration. The lower plot shows a section of the arc lamp atlas covering the expected wavelength range of the arc spectra.

Fig. 17 Atlas module -select arcline for y-offset



The offset arc spectra and atlas plots are used together to identify the pixel position and wavelength of arclines for wavelength calibration. The table of arcline data shown in the GUI can be edited manually. Alternately the (nominal) pixel positions of lines can be set by pointing and clicking on the plot of arc spectra and the wavelength set by clicking on the matching line in the atlas plot (see Fig. 18).

Fig. 18 Atlas module –select atlas and wavelengths

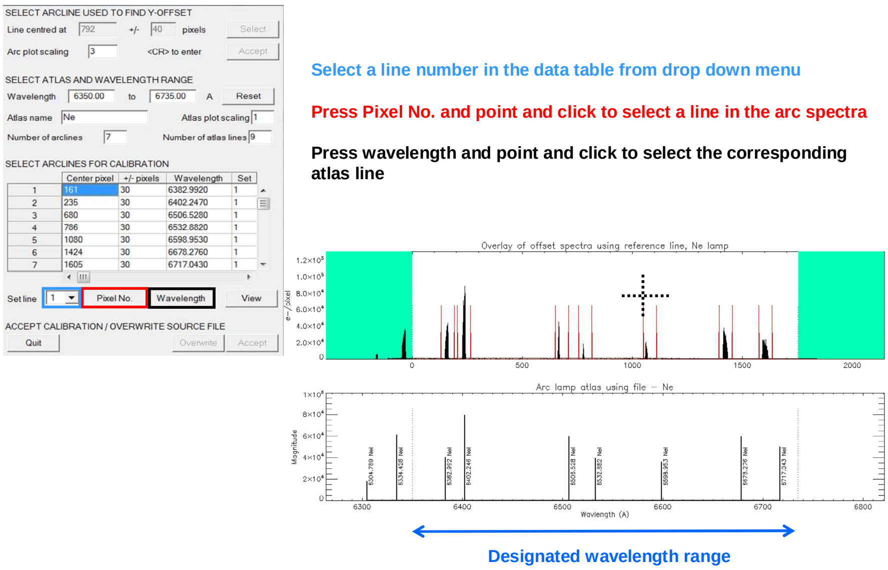
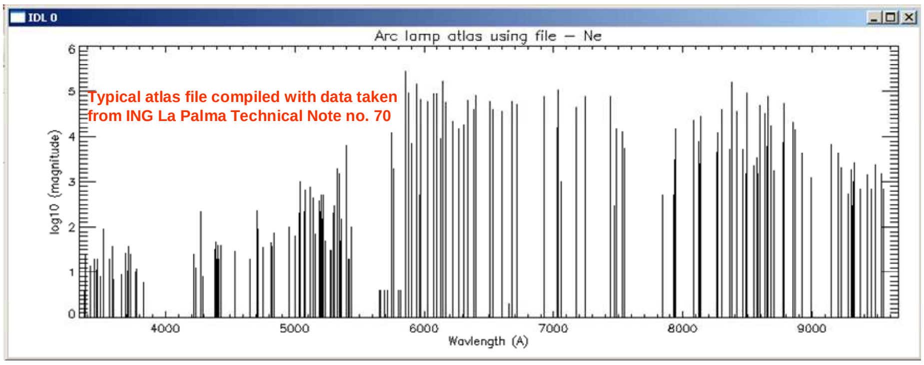


Fig.19 Atlas file



Format of atlas file Ne_He

Wavelength (A)	ID	Magnitude
6929.467	NeI	(77000)
7024.050	NeI	(16000)
7032.413	NeI	(110000)
7059.107	NeI	(1000)
7065.188	HeI	(-1)
7173.939	NeI	(45000)
7245.167	NeI	(75000)
7281.349	HeI	(-1)
7438.899	NeI	(77000)

Notes on format

- If magnitudes are unknown then equal values are assigned (typically (1)) to all lines.
- Where file covers a combination of two lamps then magnitudes of the second lamp are stored as negative values.

The reference Atlas file is selected by name in the GUI (parameter c.atlfile). Atlas files show the approximate magnitude and exact wavelengths of calibration lines for various arc lamps and lamp combinations (see Fig. 19). They are stored in subdirectory control_standards (see Figure 6). Additional atlas files can be added by the user.

The wavelength range of the atlas plot is set to match the specified wavelength range of the arc spectra (see Fig. 18). If the wavelength limits are not known then limits can be estimated from the arc file header data by pressing the reset button. These estimated limits are usually good enough to allow the user to relate the principle lines in the arc spectra to corresponding lines in the atlas file.

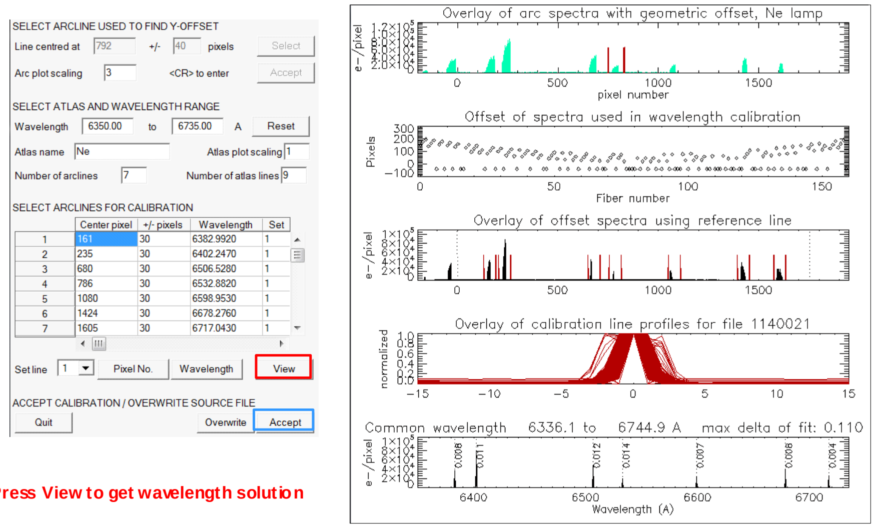
The number of atlas lines displayed is set in the GUI and selected according to their size (biggest first). The scale of the plot can also be adjusted to help identify smaller lines. Where atlas file is a composite of two different arc lamps the lines of the corresponding atlas file are displayed in two colours, one for each lamp. Typically for the He_Ne arclamp atlas the helium lines corresponding to arc lamp1 are displayed in black and the neon lines corresponding to arclamp 2 are displayed in red.

When images of two arc lamp arc spectra are used the displayed arc lamp spectra are a combination of the extracted spectra from arcfile 1 for the first N pixels and the extracted spectra from arcfile 2 for the remaining pixels (where the value of N is set by control parameter c.alinecut). Calibration lines on the left hand side of the display of offset spectra (up to pixel c.alinecut) should be matched to the atlas lines of lamp 1 and those on the right hand side matched to the altas lines of lamp 2.

Having established a preliminary list of arc lines the user can review the resulting wavelength calibration. Viewing the wavelength calibration displays the five plots of the Arc module and shows the resulting limits of the common wavelength range (see Fig. 20). Only lines marked as set in the arcline table are used for this calculation. The number of lines set should be higher than the order of the wavelength fit (c.arcorder).

The user can either accept this solution or return to edit the arcline data until a satisfactory solution is obtained. When the solution is accepted the new set of arcline data is copied to the current control file. This data can also be written to the source control file stored in the project directory. At the same time parameters specify the common wavelength range (c.wlmin and c.wlmax) are updated if either of their values are outside the limits computed for the wavelength solution.

Fig. 20 Atlas module – view wavelength solution



Press View to get wavelength solution

If line fits are ok PRESS Accept
OR go back and change the arcline data

.

10. Science spectra

Table 7 shows functional details of the module Star. The Star module first extracts the science spectra, both designated targets (type P) and sky fibers (type S), copies the wavelength calibration from the associated intermediate arc file and saves the intermediate science spectra to file (s1234567.fit). The module then processes the intermediate spectra to produce sky subtracted output spectra on a common wavelength base (see Fig. 21).

Table 7 Star Module

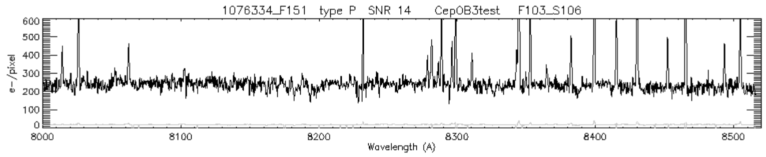
Procedure	**Star**
	Optimal extraction of spectra for a set of star files Interpolates spectra on to a common wavelength scale Sky subtraction using either median value of sky fibers or offset sky spectra
Options	Star,/NOEXTRACT – skips optimal extraction stage Star,/PLOT - saves plots of individual spectra for each fiber Star,/AUTO - Optimal extraction vs pixel number (i.e. no wavelength calibration)
Input	Raw data images of designated targets
Output	Starfiles (s1234567.fit) containing intermediate target spectra Specfiles (p1234567.fit) sky subtracted spectra over common wavelength range ASCII tables (p1234567.txt) fiber no, line, RA, Dec, Mag, Mean, SNR, sky lev,name
Report to log	RUNNING PROCEDURE:Star For file number(s), 1234567, 1234567, 1234567 Extracting file r1234567 (Repeated for each target) FLAT Iteration 0 masked points = ### (etc.) TARGET Iteration 0 masked points = ### (etc.) Add wavelength calibration from file: a1234567 Star files created: s1234567 Evaluating spectra on common wavelength base Average sky fiber signal ##.# +/- ##.# e-/pixel Reading offset sky from o1234567 (in case of offset sky) Spectra saved to file p1234567
Error	FATAL ERROR: readout binning does not match control data ERROR: central wavelength not compatible with target file r1234567 FATAL ERROR: allocation of active fibers incompatible with target file r1234567 FATAL ERROR: control file shows incompatible sky type Cannot have skytype 0, and skyoset 0 for since fiber extraction ERROR: Saturation of extracted image (>99% full scale) WARNING: Flat signal zero over part of spectra a total of ## bad pixels masked WARNING: high extracted image, ## of full scale FATAL ERROR: wavelength calibration data not present FATAL ERROR: offset sky file o1234567 not found (if offset sky specified) WARNING: sky signal is low for scaled sky subtraction WARNING: Parameter c.skymax is low, less than 3 sigma of median sky signal WARNING: poor convergence of sky subtraction for fiber # scale parameter set to 1

Intermediate spectra are interpolated over masked sections and scaled by the fiber attenuations. These spectra are then interpolated on to a common wavelength base. Limits of the common wavelength range are shown in the base GUI and are set by control parameters c.wlmin and c.wlmax. They should be within the limits identified by the Arc procedure (see Fig. 16). The number of wavelength steps is specified by parameter c.nwl and their distribution is set by parameter c.wltype where:

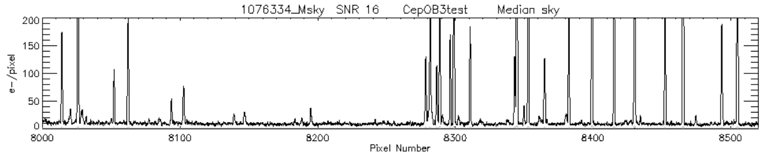
0) Gives a log wavelength scale
1) Gives a linear wavelength scale.

Fig.21 Spectra on standard wavelength base

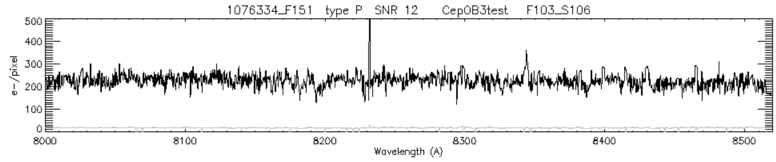
(a) Star spectrum – no sky subtraction



(b) Median sky spectrum



(a) Star spectrum – sky subtracted (scaled and masked)



There are two options for the source of the sky spectrum. The most efficient method in terms of observation time is to calculate a median sky spectrum from a number of sky allocated fibers. The alternative, which can be more accurate under certain circumstances, is to record a series of offset sky observations and use their median value to determine the sky spectra of individual fibers. The source of the median sky spectrum is shown on the base GUI (see Fig. 6) and is set by the control parameter c.skyoset:

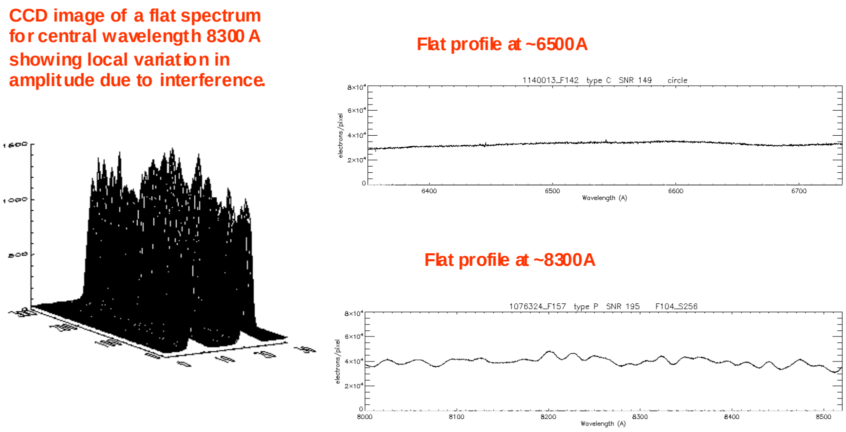
0) A median sky spectrum is found from the sky fibers in the science file.
1) Sky spectra of individual fibers are found from offset sky observations.

The median sky spectrum is calculated within the Star module. Where offset sky observations are being used, an extra module, Offset, is run to extract the appropriate sky spectra.

There are four options for the method of sky subtraction. These include an option to scale the median sky spectrum prior to subtraction in order to minimize the cross correlation function between the sky and the sky subtracted spectrum. There is also an option to mask major sky lines, where the sky subtracted target spectrum is masked and then interpolated over the width of sky lines which show a pixel count exceeding a reference value (c.skymax) in the median sky spectra. The method of sky subtraction is selected on the base GUI (Fig. 6) and set by the control parameter c.skytype;

0) No sky subtraction is made.
1) The (median or offset) sky spectrum is subtracted from the target spectrum.
2) The median sky spectrum is scaled prior to subtraction.
3) The median sky is scaled and the output spectrum is masked over sky lines.

Fig. 22 Interference on far red spectra



At far red wavelengths the CCD exhibits interference producing local variations in the signal amplitude (see Fig. 22). Optimally extracted spectra are normally scaled to the flat spectrum to compensate for this effect.

The option is available in the Star module to recover the true spectral profile by multiplying the output spectrum by the smoothed flat lamp profile of individual fibers. Smoothing averages out the local variations in pixel response while preserving the spectral profile of the flat lamp. The degree of flat lamp smoothing is set by parameter c.flatsmth. The option to multiply by the flat lamp profile is set by the control parameter c.flattype:

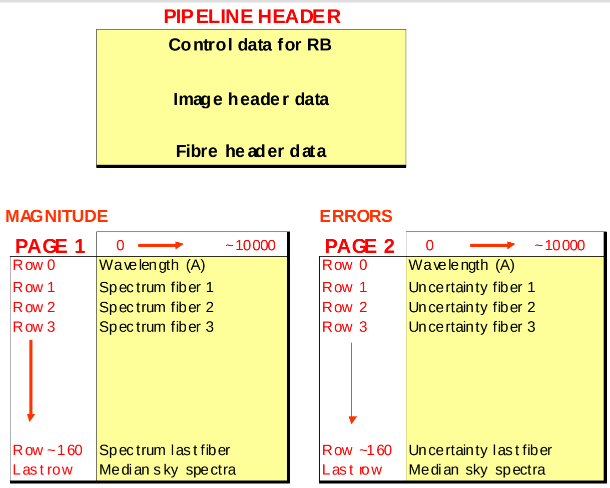
0) No action is taken.
1) Output spectra are multiplied by the flat lamp profile.

Output spectra are saved as a two page fits file (p1234567.fit) to directory spec/. The format of this file is shown in Fig. 23. A text file is also generated and saved to the same directory (p1234567.txt). This shows details of the targets for each active fiber (see Fig. 24). Plots of the output spectra are displayed on screen. These plots can be saved to the directory temp/ by selecting the option Star,/plot.

The option Star,/noextract causes the module to skip the optimal extraction phase, reading the intermediate spectra from previously generated files. This option saves processing time if the only changes required are those affecting the latter stages of the pipeline process.

The option Star,/auto is used to provide a quick view of the raw target spectra. In auto mode target spectra are optimally extracted and saved to the intermediate spectra file (s1234567.fit) without the addition of wavelength data.

Fig. 23 Format of pipeline output file



p1234567.fit - results for individual observation
q1234567.fit - median of a set of observations

Table 8 shows functional details of the module Medan. This module reads the output spectra for each science observation in an RB and evaluates the median spectra for each fiber (see Fig. 25). Spectra are normalised to their mean value before the median is calculated to compensate for variations in seeing between frames. The output spectra are the product of this median spectra times the number of observations. The rms deviation is the median rms deviation of individual spectra multiplied by the square root of the number of observations.

Fig. 24 Star module text file

p1234567.txt Summary table in ASCII format

FILE	LINE	TYPE	MEAN	SNR	RA	DEC	MAGN	SKY_SUB	NAME
FSKY	0	S	7.6	3.	17 45 34.768	+05 29 47.14	999.00	0.000	Median Sky
F001	142	P	62.5	5.	17 47 09.970	+05 29 06.42	16.93	1.000	JCO7_512
F002	141	P	822.2	26.	17 46 59.316	+05 30 10.30	14.07	1.000	JCO7_077
F003	140	P	158.4	9.	17 47 30.035	+05 31 36.49	14.20	1.000	JCO7_027
F004	139	P	3236.4	44.	17 46 25.717	+05 31 09.16	12.34	1.000	JCO8_038
F005	138	S	9.0	1.	17 46 53.013	+05 33 17.02	99.99	0.000	s0530_2470
F006	137	P	5363.4	63.	17 47 15.990	+05 35 09.42	11.45	1.000	JCO4_295
F008	135	P	1579.4	36.	17 46 13.028	+05 32 46.60	13.69	1.000	JCO5_347

etc.

1234567_L###.gif Plots of spectra (optional)

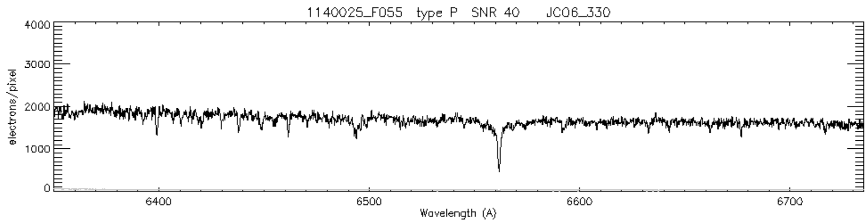
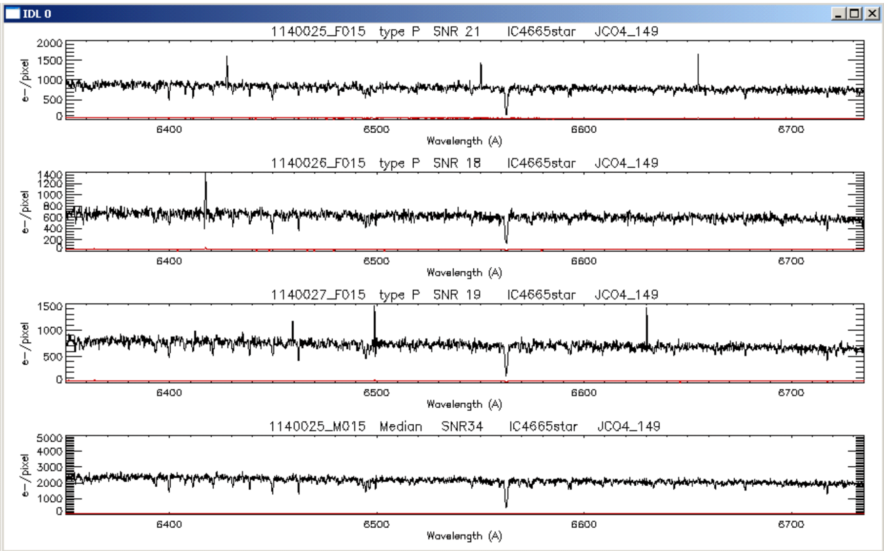


Table 8 Medan module

Procedure	**Medan**
Function	Evaluate median values of extracted spectra, plots and saves results
Option	Median,/AUTO - plots data against pixel value (i.e. no wavelength calibration)
Input	Specfiles (p1234567.fit) containing spectra for each frame Note Starfile (s1234567.fit) used in case of /AUTO mode
Output	Median file (q1234567.fit) containing spectra over common wavelength range Median file (u1234567.fit) containing spectra versus pixel no. (for auto mode) Plots showing individual spectra and resulting median spectra for each active fiber
Report to log	RUNNING PROCEDURE:Medan Medan of files: , p1234567, p1234567, p1234567 total exposure time #### sec Median spectra saved to file q1234567 Saving plots to directory c:/####/temp/

Fig. 25 Medan module plots



The output spectra are saved to a two page fits file (q1234567.fit, see Fig. 23). Combined plots of the reduced spectra for individual observations together with the median spectra are saved to subdirectory temp/ under file name 1234567_M###.gif where ### is the fiber number.

In case of a single observation the science spectrum per fiber is copied to the median spectra file. When only two observations are available then the average is taken as the mean of the two base spectra unless the difference between the two spectra is greater than a fixed factor (c.msigmax) times the uncertainty in the mean, in which case the value of the average spectrum is taken as the lower of the two base spectra.

In /auto mode spectra for each observation are read from the intermediate spectra files (s1234567.fit). In this case the median spectra are plotted as a function of pixel number. Median output spectra versus pixel number are saved to file u1234567.fit in subdirectory spec/ and plots are saved to subdirectory temp/.

Table 9 Offset Module

Procedure	**Offset**
Function	Optimal extraction of set of offset sky observations Interpolates spectra on to a common wavelength scale Evaluate median values of offset sky spectra from repeated frames
Options	Offset,/NOEXTRACT – skips optimal extraction stage
Input	Raw data images of offset sky
Output	Flie containing intermediate offset sky spectra per observation (t1234567.fit) File of median offset sky spectra on a common wavelength range (o1234567.fit)
Report to log	RUNNING PROCEDURE:Offset For file number(s), 1234567, 1234567, 1234567 Extracting file r1234567 (repeated for each target) FLAT Iteration 0 masked points = ### (repeat for each iteration) TARGET Iteration 0 masked points = ### Add wavelength calibration from file: a1234567 Offset file created: t1234567 Evaluating median offset sky spectra for target fibers Median offset sky spectra saved to file o1234567 Saving plots to directory C:/CepOB3_data/temp/
Error	FATAL ERROR: no target fibers found for sky offset FATAL ERROR: readout binning does not match control data ERROR: central wavelength not compatible with target file r1234567 FATAL ERROR: allocation of active fibers incompatible with target file r1234567 ERROR: saturation of extracted image (>99% full scale) WARNING: Flat signal zero over part of spectra a total of ## bad pixels masked WARNING: high extracted image, ## of full scale

The offset module is used to determine the median sky spectrum per fiber from a set of offset sky observations. The processes followed are similar to those described for the Star and Medan modules but without sky subtraction. Functional details of the module Offset are shown in Table 9.

The offset sky spectra of target fibers (type P) are optimally extracted and the wavelength calibration is copied from the intermediate arc file. The intermediate spectra are then interpolated over masked sections, scaled according to the relative fiber attenuations and interpolated on to a common wavelength scale. The median value of the individual frames is then taken to determine the median offset sky per fiber.

The output spectra is the product of this median spectra multiplied by the number of observations. The exposure time in the median offset sky header is the total exposure time for all offset sky observations.

11. Pipeline Utilities

Pipeline utilities are provided to load control files, run the pipeline modules in sequence and view the resulting spectra in various formats. The utility LoadRB allows the user to select a new control file and save it to the working directory so that it becomes the active control file. Details of the utility LoadRB are shown in Table 10.

Table 10 LoadRB utility

Procedure	**Loadob**
Function	Selects an RB control file from a network directory
Option	Loads a copy as the active control file Resets the log file of the selected RB
Input	Control file in project directory
Output	Active control file in working directory
Report to log	Loading control_file: C:/xxx/xxx.txt

Table 11 shows details of the Pipeline utility. This utility runs the pipeline modules in sequence to produce a set of optimally extracted spectra. The following options can be selected:

1) **Pipeline,/auto** is used to take a quick look at a set of science data in the form of optimally extract spectra with no wavelength calibration or sky subtraction. Modules are run in Auto mode allowing key control parameters to be generated from AF2/WYFFOS header data rather than entered by the user.

2) **Pipeline,/full** is used to process the first RB to be analyzed with a new AF2/WYFFOS setup. It runs all of the calibration modules, extracting the intermediate arc spectra and then displays the Atlas GUI so that the user can generate a new set of arcline data for wavelength calibration. The pipeline then produces sky subtracted spectra on a common wavelength base.

3) **Pipeline** is used to process repeat RBs in a project with the same AF2/WYFFOS set up. It runs a reduced set of calibration files (only Flat and Circ) and uses the existing set of arcline data for wavelength calibration allowing the extraction process to run without user intervention.

The utility ATV can be run as a stand alone procedure to load and view all types of pipeline fits files (see Fig. 9) and to export encapsulated PostScript plots of the image arrays and/or plots of sections through the image, The associated utility cmps_form.pro (which must be present for ATV to run) is used to set the output file name and PostScript parameters. Note the CCD image in the AF2 data files is held in Fits extension 1.

Table 11 Pipeline utility

Procedure	**Pipeline**
Function	Runs the pipeline modules to extract spectra using project calibration data
	Modules run Flat > Circ > Arc > Offset > Star > Medan
Options	Pipeline,/full runs all modules generating bias and mask files and arcline data
	Bias > Mask > Flat > Circ > Arc (with Atlas) > Offset > Star > Medan
	Pipeline,/auto runs modules in auto mode
	Use existing Bias and Mask files, no sky subtraction or wavelength calibration
	Pipeline,/auto,/full runs all modules in auto mode
	Generates Bias and Mask files, no sky subtraction or wavelength calibration
Input	Raw data files + intermediate Bias and Mask if available
Output	Standard output for each module run
Report to log	RUNNING PROCEDURE:Pipeline
	Using control file: xxx
	Setting auto mode options: profile:1, fibbrattn:1, flattype:0, nscatter:1 (if applicable)
	Then one of the following:
	Running all modules in Auto mode
	Using existing bias and mask files, running remaining modules in Auto mode
	Running all modules
	Using existing bias and mask files
	Followed by standard reports for each pipeline model
Error	ERROR: target file header different no. of fibers (#) and/or fiducials (#)
	Overwriting data in control file to match target data
	ERROR: target file header shows different readout binning (X:#, Y:#)
	Overwriting data in control file to match target data
	FATAL ERROR: control file shows incompatible sky type
	Use skytype:0 or set offset sky for single fiber extraction
	ERROR: intermediate Bias file not found - running Bias
	FATAL ERROR: designated Bias file shows incompatible binning
	ERROR: intermediate Mask file not found - running Mask
	FATAL ERROR: designated Mask file shows incompatible binning

The utility View is used to view extracted spectra. It reads output spectra files for the current RB and saves output to the subdirectory temp/ of the project directory. Table 12 gives details of the View utility. The following options are available;

1) **View[,mag,err]** saves text copies of the file headers for input and output data files and returns data arrays mag and err containing the magnitude and error of the output spectra in the format shown in Fig. 23. The AF2/WYFFOS data file header of the first target file is saved to file r1234567.txt and the pipeline header (read from the intermediate output spectra) is saved as s1234567.txt.

2) **View,/plot** reads the median output spectra (q1234567.fit) and plots the spectra of active fibers. These are saved as gif images in directory temp/ as files named 1234567_M###.gif (where ### is the fiber number).

3) **View,/ps** reads the median output spectra (q1234567.fit) and generates PostScript plots of the medain spectra of active fibers (15cm wide by 6cm high). These are saved to directory temp/ as files 1234567_M###.ps

4) **View,/text** reads the median output spectra (q1234567.fit) and saves the spectra of active fibers as ASCII files 1234567_M###.txt of wavelength, magnitude and error. The format of these files is shown in Fig. 26.

5) **View,/attenuation** uses the mean values of spectra measured on a uniformly illuminated target to determine the fiber attenuations relative to the mean. Results are shown graphically (see Fig. 26) and are saved to the log/ directory both as a gif plot (fiber_attn.gif) and as a text file (fiber_attn.txt). Fibers must be configured in a circle set up and the fiber gains set to unity (c.fibattn = 0) in order to use this option.

Table 12 View utility

Procedure	**View**
Function	View and save file headers and output spectra in different formats Saves text copies of the file headers for input and output data files
Options	View,xxx returns a two page data array containing the output spectra View,/plot - plots the output spectra for active fibers and saves a copy to file View,/text - saves the output spectra for active fibers to individual ASCII files View,/gain - Saves a table and plot of relative fiber attenuations to log/ directory
Input	Target data file and files of intermediate and output spectra
Output	Plots and/or text files for individual spectra and/or a table of fiber attenuations
Report to log	RUNNING PROCEDURE:View Saving copy of AF2/WYFFOS header for file: r1234567 Saving copy of pipeline header for file: s1234567 Then one of the following: Saving GIF plot(s) of xxx spectra to c:/xxx/xxx Saving PS plot(s) of xxx spectra to c:/xxx/xxx Saving ASCII copies of xxx spectra to c:/xxx/xx Evaluating fiber attenuation using source file sxxxxx.txt Standard deviation in attenuations xx.x%
Error	FATAL ERROR: file not found:xxx FATAL ERROR: invalid fiber set up, targets must be in circle setup for attenuations ERROR: fiber gains invalid, fiber attenuation should be unity in source file

Fig. 26 Outputs from utility View

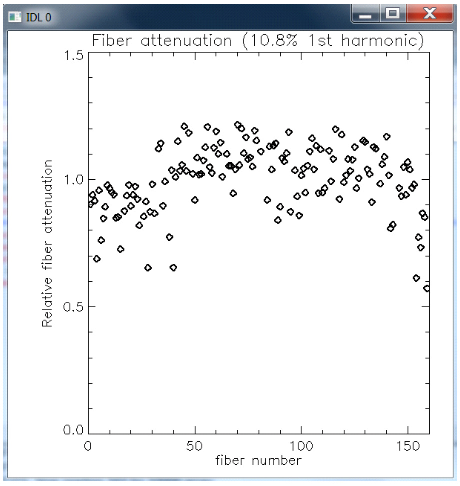
Output spectra in ASCII format
in file 1234567_M###.txt

6350.000	1177.1	38.5
6350.038	1157.0	38.2
6350.075	1140.3	37.9
6350.112	1126.1	37.7
6350.149	1110.5	37.6
6350.187	1102.5	37.7
6350.224	1105.6	37.8
6350.262	1114.3	38.1
etc.		

Table of fiber attenuations
in log file fiber_attn.txt

#Fiber	attn
1	0.9192
2	0.9451
3	0.9054
4	0.6493
5	0.9461
6	0.7623
etc.	

Plot of fiber attenuations



A graphical user interface, GUO.pro, is provided to view contents of the current control file data and the resulting output spectra (see Fig. 27). It is similar in appearance to the data entry interface shown Fig. 6 but in this case control parameters can only be viewed, not changed. Output spectra from individual and median observations can be viewed using the ATV utility. Action buttons at the bottom right of the display are provided to output the median spectra in different formats - ASCII files, GIF plots or PostScript plots.

Fig 27. Viewing output files with utility GUO

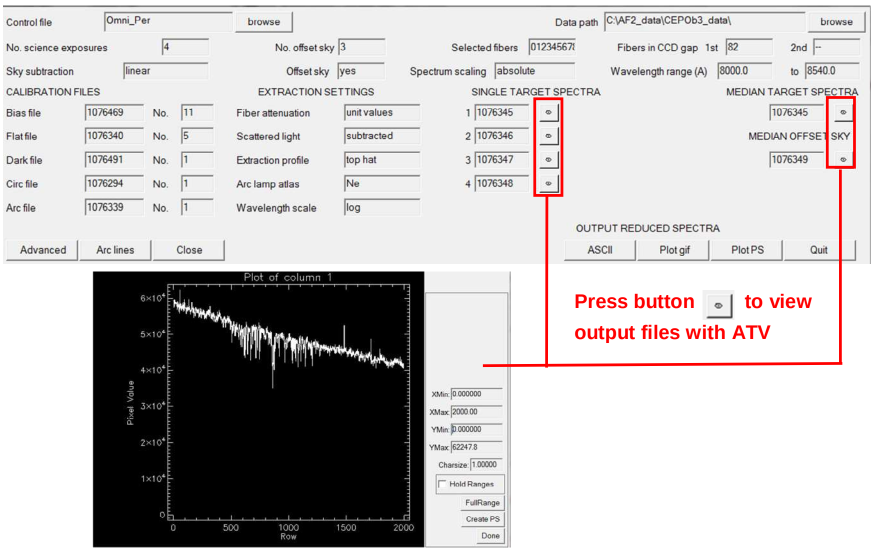


Fig. 28 lists the procedures used to read and write pipeline data files. These can be accessed by users who wish to write additional IDL modules to carry out futher processing of extracted spectra.

Read_data reads a file in the standard AF2/WYFFOS format and ,in effect, converts it to the format used by the pipeline procedures returning the standard pipeline header (structures c,h and f) and a copy of the CCD image. Read_file reads a file in the standard pipeline format returning the standard pipeline header (structures c,h and f) and a copy of the file data array.

Fig. 28 Data file I/O

read_control,c [,file]	Reads ASCII file containing control data Returns control data structure, c Reads current control file unless **file** is specified (e.g. read_control,c,file="c:/p_directory/rbname.txt")
write_control,c [,file]	Writes ASCII file containing control data Input is control data structure, c Saves to current control file unless **file** is specified
read_data,1234567,c,h,f,image [,path]	Reads AF2/WYFFOS data file r1234567.fit Returns pipeline header (c,h,f) and binary image Reads from subdirectory data/ unless **path** is specified
write_file,file,c,h,f,array[,page]	Writes fits file of pipeline header and data array Input pipeline header (c,h,f) & data array (for current page) **page** is specified for a multi page file, default is single page
read_file,file,c,h,f,array[,page]	Reads fits file of pipeline header and data array Returns pipeline header (c,h,f) & data array (for current page) **page** is specified for a multi page file, default is single page
printl,c.log,'text string'	Appends specified text string to log file named c.log

Pipeline header structure **c** - control data for the reduction block
+ structure **h** - parameters from the AF2/WYFFOS file header for the CCD image
+ structure **f** - parameters from the AF2/WYFFOS file header for individual fibers

Appendix 1: Pipeline call structure

The pipeline comprised ten top level procedures which call a total of ten first level subroutines which in turn call three second level subroutines in addition to the I/O routines shown in in Fig. 28. The call structure of the pipeline is shown below.

Bias

Create median bias file (see Table 1)
Calls: File I/O procedures (see Fig. 28)

Mask

Create mask file (see Table 2)
Calls: File I/O procedures (see Fig. 28)
file_debias

Flat

Create median flat file (see Table 3)
Calls: File I/O procedures (see Fig. 28)
file_debias

Circ

Trace line centre on CCD (see Table 4)
Calls: File I/O procedures (see Fig. 28)
file_debias
circ_make
file_offset

Arc

Extract arc spectra (see Table 5)
Calls: File I/O procedures (see Fig. 28)
extr_file
Atlas (optional call)
arc_offset
arc_cal

Offset

Extract offset sky spectra (see Table 9)
Calls: File I/O procedures (see Fig. 28)
extr_file
star_make
medan_make

Star

Extract target spectra (see Table 7)
Calls: File I/O procedures (see Fig. 28)
extr_file
star_make

Medan

Median of extracted spectra (see Table 8)
Calls: File I/O procedures (see Fig. 28)
medan_make

view

View output spectra (see Table 12)
Calls: File I/O procedures (see Fig. 28)
view_plot
view_text
view_attn

loadrb

Load new active control file (see Table 10)
Calls: File I/O procedures (see Fig. 28)

pipeline

Run set of pipeline modules(see Table 11)
Calls: File I/O procedures (see Fig. 28)
Pipeline modules (see above)

file_debias

Returns debiased CCD data image
Calls: File I/O procedures (see Fig. 28)

circ_make

Returns details of line centres on CCD
Calls: File I/O procedures (see Fig. 28)

file_offset

Returns image with spectra at x-axis value
Calls: None

extr_file

Returns intermediate extracted spectra
Calls: File I/O procedures (see Fig. 28)
file_debias
file_offset
extr_make

arc_offset

Returns y-axis offsets of arc spectra
Calls: None

arc_cal

Returns wavelength calibration for spectra
Calls: None

star_make

Returns output spectra for single observation
Calls: File I/O procedures (see Fig. 28)
sky_sub
flat_scale

medan_make

Returns median of a set of output spectra
Calls: None

view_plot

Save plots of median output spectra to file
Calls: None

view_text

Saves median output spectra in ASCII format
Calls: None

view_attn

Saves fiber attenuations to file
Calls: None

extr_make

Returns an optimally extected spectrum
Calls: None

sky_sub

Returns a sky subtracted spectrum
Calls: File I/O procedures (see Fig. 28)

flat_scale

Returns a smooth flat lamp profile
Calls: None

In addition to the pipeline procedures shown above, three graphical user interfaces are provided to create and edit the control file (the ASCII data file containing the list of parameters that control pipeline operation) and to view input and output data files. The call structure of these procedures is shown below.

GUI

Ceate/edit control file (see Fig. 6)
Calls: read_control (see Fig. 28) write_control (see Fig. 28) ATV Pipeline,/auto

pipeline,/auto

Runs pipeline in auto mode (see Table 11)
Calls: File I/O procedures (see Fig. 28) Bias Mask Flat Circ Star Medan

ATV

View and plot pipeline fits files
Calls: cmps_form

CPMS_FORM

Returns plot file name &PostScipt setting
Calls: none

Atlas

Select arc calibration lines (see Table 6)
Calls: File I/O procedures (see Fig. 28) arc_offset arc_atlas arc_cal arc_gui

arc_offset

Returns y-axis offsets of arc spectra
Calls: None

arc_altas

Adds atlas line data to control structure
Calls: read_control (see Fig. 28)

arc_cal

wavelength calibration for spectra
Calls: None

arc_gui

Displays atlas graphical user interface
Calls: write_control (see Fig. 28)

GUO

Display/save output spectra (see Fig. 6)
Calls: File I/O procedures (see Fig. 28) ATV view,/text view,/plot view,/ps

view

Save spectra in different formats (see Table 12)
Calls: File I/O procedures (see Fig. 28) view_text view_plot,/plot view_plot,/ps

Appendix 2: Control data

IDL procedure read_control,c reads data from the active control file into structure c. Table A1 shows the parameters in structure c, typical values and a descriptor. Note structure c contains additional working variables used to transfer data between modules and GUIs (see programmers notes).

Table A1 Control data

Type	Parameter	Value	Descriptor
Key	c.rbname	Omni_Per	Unique name of Reduction block RB
parameters	c.pdir	C:/CepOB3	Project directory path and name
	c.fiberid	15	999:extract spectra of all fibers
			###:extract spectra for single fiber no. ###
	c.dfiber1	82	Fiber number of first fiber lost in gap between two halves of CCD
	c.dfiber2	0	Fiber number of second lost fiber (0 if no second fiber lost)
	c.skyoset	1	0:A median sky spectrum is found from the sky fibers in the science file.
			1:Sky spectra of individual fibers are found from offset sky observations.
	c.skytype	1	0:No sky subtraction is made
			1:The sky spectrum is subtracted from the target spectrum.
			2:The median sky spectrum is scaled prior to subtraction.
			3:The median sky spectrum is scaled and to output masked over sky line
	c.wlmin	8000	Lower cut off for common wavelength range
	c.wlmax	8540	Upper cut off for common wavelength range
Options	c.fibrattn	0	0:Fiber attenuations are set uniformly to 1
			1:Attenuations calculated from the mean line counts in the offset image.
			2:Attenuations are read file fiber_attn.txt in directory control_standards/.
			3:Attenuations are read file fiber_attn.txt in directory project log directory.
	c.profile	1	0:A top hat profile is used producing non optimal extraction
			1:The spatial profile is found from the flat image for optimal extraction.
	c.flattype	0	0:Output spectra are as defined by extraction process (target/flat)
			1:Output spectra are multiplied by the flat lamp profile
	c.nscatter	1	0:No compensation is made for scattered light
			1:Background level due to scattered light is subtracted before extraction.
	c.wltype	0	0: Output spectra on a log wavelength scale
			1: Output spectra on a linear wavelength scale
Settings	c.xbin	2	Size x bin of CCD image in raw data file
	c.ybin	2	Size y bin of CCD image in raw data file
	c.csize	1.0	Plot character sizing – scale relative to standard size
	c.maskcut	100	Dark current cut-off level for a mask calculation
	c.maskmax	100	Upper limit cut-off values for mask calculation
	c.masklev	0.004	Fraction of pixels masked when autoranging mask cut-off level
	c.circlev	6	Number of profiles of circ file image used for line tracing
	c.circsmth	2	Smoothing applied to circ file profiles for line tracing
	c.circzero	0.025	Offset of zero line (as percent of the peak) for line tracing
	c.lwide	24	Parameter setting width of CCD trace across line profile
	c.ysmooth	10	Smoothing parameter used to remove pixel to pixel noise
	c.boxwide	9	Half-width of line in x pixels for top hat extraction
	c.staritn	1	Number of iterations of mask data for optimal extraction
	c.nwl	10000	Number of steps in common wavelegth range
	c.flatsmth	20	Number of pixel smoothing flat spectrum to produce smooth flat

Table A1 continued

Type	Parameter	Value	Descriptor
Subdirectories	c.pdata	data/	Subdirectory containing raw data files
	c.pint	int/	Subdirectory containing intermediate pipeline data files
	c.pspec	spec/	Subdirectory containing output spectra
	c.plog	log/	Subdirectory containing log files
	c.ptemp	temp/	Subdirectory containing temporary output files
Limits	c.biaserr	3	Warning level for high change in bias e.g. 6 x RON
	c.flaterr	4	Warning level for high change in flat signal e.g. 4 x Poison noise
	c.sighi	0.6	Warning level for high signal as fraction of full scale
	c.siglo	0.1	Warning level for low signal as fraction of full scale
	c.extsigma	5	Maximum error (in sigma) for unmasked pixel in optimal extraction
	c.skymin	0.001	Warning level for low sky signal in scaled sky subtraction
	c.skymax	50	Level of sky line that is clipped for masked sky subtraction
	c.msigmax	2	Max error criteria used for median of two signals
CCD data	c.ccdron	4	CCD read out noise (e-)
	c.ccdgain	0.8	CCD gain (count/e-)
	c.ypixel	4200	Number of CCD pixels in y direction
	c.ymax	3900	Maximum y-pixel on CCD image for acceptable signal
	c.ccdzero	32768	Zero level for binary CCD image
	c.noscan	30	Width of overscan region used in bias correction
	c.oscan1	38	x pixel positions of start of overscan region on LHS gap
	c.oscan2	2112	x pixel positions of start of overscan region on LHS image
	c.oscan3	2194	x pixel positions of start of overscan region on RHS gap
	c.oscan4	4264	x pixel positions of start of overscan region on RHS image
	c.circcutl	0	Addition number of pixels masked to LHS of gap for line tracing
	c.circcutr	0	Addition number of pixels masked to RHS of gap for line tracing
	c.nfiber	160	Number of fibers in AF2/WYFFOS set up
	c.nfiducl	10	Number of fiducials in AF2/WYFFOS set up
	c.rcircle	89000	Minimum radial position of fibers on plate+D36 in circle set up
	c.dfiber3	0	Spare dead fiber pointers
	c.dfiber4	0	Spare dead fiber pointers
	c.fiberdip	5032	Scaling parameter for overall geometric y-offset of fibers
	c.fiberoff	54	Scaling parameter for local geometric y-offset of fibers
	c.pixelsiz	0.0135	Size of CCD pixel
Arc calibation parameters	c.arcorder	4	Order of polynomial describing wavelength per pixel
	c.arcwide	30	Width of arc spectra used to fit Gaussian profile to peak
	c.arcrefc	1030	Pixel position of arcline used to determine y-offset of fibers
	c.arcrefw	50	Pixel range of arcline used to determine y-offset of fibers
	c.arcscale	10	Multiplier used to scale plots of arc spectra
Atlas parameters	c.atlfile	Ne	Name of atlas file referenced by Atlas procedure
	c.natl	9	Number of atlas lines displayed in atlas plot
	c.atlscale	1	Multiplier used to scale plots of atlas line magnitudes
	c.atlflag	0	Working variable: showing status of Atlas procedure
	c.atlaline	0	Working variable: showing line being edited in Atlas procedure

Table A1 continued

Type	Parameter	Value	Descriptor
Bias files	c.nbias	7	Number of bias files (1-21)
	c.dbias(0)	1076469	Bias file numbers
	c.dbias(1)	1076470	""
	c.dbias(2)	1076471	""
	c.dbias(3)	1076472	""
	c.dbias(4)	1076473	""
	c.dbias(5)	1076474	""
	c.dbias(6)	1076475	""
	c.dbias(7)	1076476	""
Flat files	c.nflat	3	Number of flat files (1-11)
	c.dflat(0)	1076340	Flat file numbers
	c.dflat(1)	1076341	""
	c.dflat(2)	1076342	""
Dark files	c.nmask	1	Number of dark files (1-5)
	c.dmask(0)	1076491	Dark file numbers
Circ files	c.ncirc	1	Number of circ files (0 or 1)
	c.dcirc(0)	1076294	Circ file number
Arc files	c.narc	1	Number of arc files (1 or 2)
	c.darc (0)	1076339	Arc file number
Target file	c.nstar	3	Number of target files (1 to 11)
	c.dstar(0)	1076345	Star file numbers
	c.dstar(1)	1076346	""
	c.dstar(2)	1076347	""
Offset sky files	c.noset	2	Number of oset files (0 to 11)
	c.doset(0)	1076349	Offset sky file numbers
	c.doset(1)	1076350	""
Arcline data	c.narcline	5	No. of lines in arcline table
	c.calwl(0)	8082.458	Central wavelength of arcline
	c.pixc(0)	426	Center pixel of arcline in arc lamp spectra
	c.pixw(0)	25	Pixel range for arcline in arc lamp spectra
	c.flag(0)	1	0:arcline not used / 1:arcline used for wavelength calibration
	c.calwl(1)	8136.406	""
	c.pixc(1)	575	""
	c.pixw(1)	25	""
	c.flag(1)	1	""
	c.calwl(2)	8377.607	""
	c.pixc(2)	1289	""
	c.pixw(2)	25	""
	c.flag(2)	1	""
	c.calwl(3)	8418.427	""
	c.pixc(3)	1414	""
	c.pixw(3)	25	""
	c.flag(3)	1	""
	c.calwl(4)	8495.36	""
	c.pixc(4)	1660	""
	c.pixw(4)	25	""
	c.flag(4)	1	""

Appendix 3: Pipeline header

Fits files generated by the pipeline all share a common header structure. This comprises three elements:

1. Observational data read for the AF2/WYFFOS source file header
2. Fiber data read from the AF2/WYFFOS source file header
3. A copy of the control_file data.

Table A2 shows the components of observational data. These are loaded into IDL structure h when a pipeline file is read.

Table A2 Observational data

Structure h	Fit header value	Descriptor
h.run	RUN = 1140025 /	Run number
h.obstype	OBSTYPE = 'TARGET ' /	Type of observation, e.g. TARGET
h.wyffos	WYFFOS = 'WYFFOS ' /	Name of instrument configuration
h.cenwave	CENWAVE = 6600 /	Approx. central wavelength (Angstroms)
h.disperse=	DISPERSE= 17 /	Nominal dispersion (Angstroms/mm)
h.linesmm	LINESMM = 632 /	No. of grating rulings per mm
h.lamp	LAMP = 'OFF ' /	WYFFOS Comparison Lamps
h.wyfconf	WYFCONF = 'REFLECTION ' /	WYFFOS Configuration (Reflection or Trans.)
h.wyfgrat	WYFGRAT = 'ECHELLE ' /	Name of grating
h.wyforder=	WYFORDER= ' 4' /	Wavelength order (Echelle grating)
h.confra	CONFRA = '17 45 34.5 ' /	Field RA as defined in CONFIGURE
h.confdec	CONFDEC = '+05 29 53.85 ' /	Field Dec as defined in CONFIGURE
h.field	FIELD = 'IC4665_B1 ' /	Field name as defined in CONFIGURE
h.ndead	NDEAD = 6 /	Total number of fibers disabled
h.nfiber	NFIBER = 160 /	Total number of fibers in this module
h.nfiduc	NFIDUC = 10 /	Total number of fiducial positions
h.ra	RA = ' 17:45:34.767' /	RA being tracked
h.dec	DEC = '+05:29:47.19' /	DEC being tracked
h.equinox	EQUINOX = ' 2000.00' /	Equinox of coordinates
h.jd	JD = 2454626.41452 /	Julian Date at start of observation
h.start	START = ' 13:55:59.2' /	Local sidereal time at start of observation
h.xbin	XBIN = 2 /	Binning factor in x axis
h.ybin	YBIN = 2 /	Binning factor in y axis
h.exptime	EXPTIME = ' 600.01000' /	[s] Exposure time
h.darktime	DARKTIME = 600.283 /	[s] Integration time

Table A3 shows the components of the fiber data. These are loaded into an IDL array structure f when a pipeline file is read. This array is indexed by fiber number and stored in the fits file header under parameter names F001xxx, F002xxx, F003xxx etc.

The control data (see Table A1) provides a complete record of the associated files and parameters used in the pipeline process to generate data held in the fits output files. The header can be viewed as a text file using pipeline utility View.

TableA3 Fiber data

Structure f(#)	Fit header value	Descriptor
f(#).stat	F001STAT= '1 ' /	1:valid fiber
		2:fiducial fiber
		3:dead fiber (recordered in AF2/WYFFOS header)
		4:dead fiber located in gap between halves of CCD
		5:invalid fiber in case of single fiber extraction
f(#).line	F001LINE= '142 ' /	Line number on CCD image (see Circ module)
f(#).type	F001TYPE= 'P ' /	P:fiber allocated to star target
		S:fiber allocated to a sky target
		C:fiber set in Circle set up
		X: unused fiber, parked or dead
f(#).name	F001NAME= 'JCO7_512' /	Object name (from file CONFIGURE)
f(#).mag	F001MAGN= '16.935 ' /	Object magnitude (from file CONFIGURE)
f(#).ra	F001RA = '17 47 09.970' /	Fiber RA
f(#).dec	F001DEC = '+05 29 06.42' /	Fiber Dec
f(#).x	F001X = -2607.00000000 /	Fiber x position (microns)
f(#).y	F001Y = 82024.0000000 /	Fiber y position (microns)
f(#).gain	F001GAIN= 0.918600 /	Fiber relative transmission (see Circ module)
f(#).sky	F001SKY = 1.00000 /	Scaling of median sky subtraction (see Star module)