The WFC3 IR Grism Data Reduction Cookbook

for Automated Extraction and Calibration of Slitless Spectra

Harald Kuntschner, Martin Kümmel, Jeremy R. Walsh
Space Telescope European Coordinating Facility

Editor: Janice C. Lee, *STScI*

Version 1.3
March 11, 2013

Contents

1 List of acronyms

aXe	Software package for extraction of spectra from HST grism observations
aXeSIM	Simulation package for HST grism spectroscopy
aXe2web	Stand alone python package to produce grism spectra Web pages
Direct image	“Normal” image of the target field taken with a given filter
ERS	Early Release Science
FITS	The Flexible Image Transport System
FOV	Field-of-View
Grism image	Grism image taken with one of the grisms inserted into the path
IR	Infrared
PET	Pixel Extraction Table
ST-ECF	Space Telescope European Coordinating Facility
`STSDAS`	Space Telescope Science Data Analysis System
WCS	World Coordinate System
WFC3	Wide-Field-Camera 3

2 Introduction

Slitless spectra are often considered 'difficult' in comparison with long slit spectra to extract and calibrate. This arises from the fact that the zero point of the wavelength scale is not fixed, but set by the position of the dispersed object in the field, since there is no physical slit. Moreover, the overlap of spectra can lead to spectral confusion. These difficulties can be surmounted by dedicated software.

This cookbook presents a short guide through a basic WFC3 IR grism (G141) data reduction performed with the aXe data reduction package (Pirzkal et al. 2001, Kümmel et al. 2009). Some suggestions about quality checks of the results are provided in Sec. 5, including simulations with the package aXeSIM. Throughout the document we use the aXe package released as part of STSDAS 3.16 in 2013 April and aXeSIM package version 1.3.

The purpose of this document is to provide a *concise* presentation of all steps involved in a basic aXe data reduction. For more detailed information please refer to the aXe and aXeSIM manuals provided on the Web (aXe Manual and aXeSIM Manual). The instructions and commands needed to perform the example data reduction are all provided so that the user can reproduce the results and modify the commands for their own data and requirements.

The essence of the aXe software is to drive the extraction by a catalogue of the objects which are in the field of the slitless spectrum image, and by parameter files which calibrate the positioning of the spectra, the wavelength dispersion solution and the conversion from extracted detector units to physical units. The aXe software has been designed to operate automatically given a specified instrument mode, the object catalogue and the requisite data. It enables extraction of thousands of spectra without user interaction.

Important remarks or comments are indicated throughout the cookbook by the following symbol:

☞

For example,

☞ *Cutting and pasting from the cookbook PDF file to run* `PyRAF` *commands works on most operating systems. However, depending on your detailed setup (fonts, language setting), and especially when quotation marks and underscores are included, this has caused problems. If in doubt simply type the command directly into your shell.*

2.1 aXe and aXeSIM

The software packages aXe and aXeSIM (Pirzkal et al. 2001, Kümmel et al. 2009) are distributed as sub-packages to `STSDAS` under `analysis.slitless.axe` and `analysis.slitless.axesim`. For installation instructions please refer to the Web pages for aXe and aXeSIM. Further information on the `STSDAS` package can be found at `STSDAS` Home.

☞ *The aXe software package cannot work directly on data taken in sub-array mode. Data taken in sub-array mode must first be embedded in a full frame, before reduction with aXe. A python script which does this is available from the WFC3 Grism Resource Webpage under "aXe Spectral Extraction Software."*

Table 1: Summary of input data files used in this cookbook

FILE	FILTER	DATE-OBS	TIME-OBS	EXPTIME	POSTARG1	POSTARG2
`ib6o23rsq_flt.fits`	G141	2009-10-03	14:43:57	1102.935669	-10.012000	5.058000
`ib6o23rtq_flt.fits`	F140W	2009-10-03	15:02:54	202.932922	-18.838249	5.026986
`ib6o23ruq_flt.fits`	G141	2009-10-03	15:07:35	1002.935364	9.988000	5.050000
`ib6o23rwq_flt.fits`	F140W	2009-10-03	15:24:52	202.932922	1.161730	5.053683
`ib6o23ryq_flt.fits`	G141	2009-10-03	16:15:13	1102.935669	9.971000	-5.045000
`ib6o23rzq_flt.fits`	F140W	2009-10-03	16:34:10	202.932922	1.162243	-5.041331
`ib6o23s0q_flt.fits`	G141	2009-10-03	16:38:51	1002.935364	-9.958000	-5.054000
`ib6o23s2q_flt.fits`	F140W	2009-10-03	16:56:08	202.932922	-18.766710	-5.084905

2.2 Calibration and reference files

The aXe data reduction software is not instrument specific[1] but instrument parameters are provided through a configuration file. Reference files for flux calibration, as well as special flat fields, also must be provided. To facilitate an accurate global background subtraction, a master sky is provided for both grisms. The files are available from the WFC3 grism Web pages at STScI (see also Kuntschner et al. 2009).

3 Example dataset

The data used in this cookbook originates from the WFC3 Early Release Science (ERS) II campaign (PID: 11359, PI: R. O'Connell). The WFC3 IR channel has a field of view of 123 x 136 arcsec at a resolution of 0.13 arcsec/pixel. The ERS II grism field lies in the north-central region of Chandra Deep Field South (J2000 53.071121 -27.709646 [degrees]). The data reductions for the two IR grisms (G102 and G141) are very similar (Straughn et al. 2010) and differ only in the use of separate input data, configuration and reference files; therefore we present in this cookbook only the example reduction for the G141 grism.

There are four individual exposures in the F140W filter and four individual grism G141 images. The images were dithered mostly along the x-axis. Fig. 1 shows the general layout of the observed field and Table 1 provides information on filenames, exposures times and dither steps ("POSTARG1" and "POSTARG2").

☞ *The input files for this example data reduction were taken from a standard* `calwf3` *pipeline reduction of the observing program. Note, that aXe works only with the* `_flt.fits` *files and does not make use of any of the other files (e.g.,* `_drz.fits`*) which are produced by the standard* `calwf3` *pipeline.*

4 Reducing WFC3 IR grism data

We first give an overview of the basic steps involved in extracting spectra from grism images that typically contain traces from several hundred objects (see Fig. 2). Each of the steps below will then

[1]The aXe package has been extensively used for HST/ACS grism spectroscopy (e.g., Pirzkal et al. 2001; 2004; 2009, Pasquali et al. 2006) and also for the HST NICMOS G141 grism (Freudling et al. 2008).

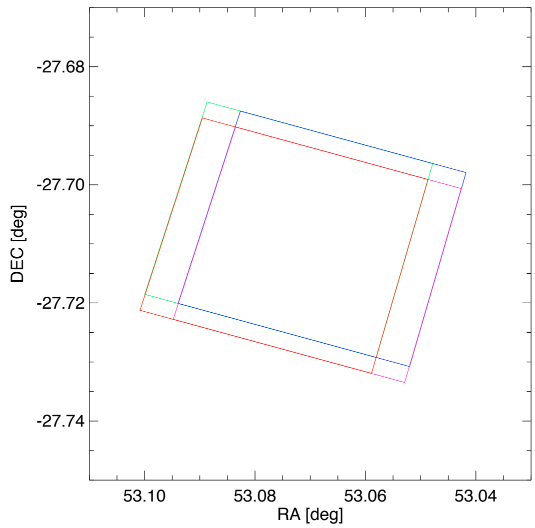

Figure 1: Layout of the ERS II F140W observations. The colors represent different pointings.

be described in more detail in the sections that follow.

- **Make the direct image catalogue (`SExtractor`; §4.3).** This step generates information on object position and size for all objects in the field and creates a master object catalogue. These measurements are later used to define extraction boxes and calculate wavelength solutions in the extraction step. Typically, the object information is derived from a drizzled direct image which captures the full FoV (§4.2).

- **Project the master object catalogue positions to the coordinate system of individual direct images (`iolprep`; §4.4).** Since the individual images typically are dithered, the master object catalogue positions need to be transformed to the coordinate system of each direct image–grism image pair.

- **Prepare the grism images and remove sky background (`axeprep`; §4.6).** In this step the units of the grism images are checked and a scaled master sky background is subtracted to achieve a homogeneous background level ready for spectral extraction.

- **Extract sets of pixels containing spectra for each object. (`axecore`; §4.7).** For each individual grism image the 2-dimensional spectra of all objects in the transformed catalogue are extracted and stored in a Pixel Extraction Table (PET).

- **Combine all available spectra of each object using drizzle techniques (`axedrizzle`; §4.8).** For each object, the available 2-dimensional spectra are combined with the help of drizzle techniques. The results are saved in the form of calibrated, 2-dimensional spectral FITS images and as extracted 1-dimensional FITS tables, including information on counts, fluxes, and an estimate of the spectral overlap by other objects (hereafter referred to as "contamination").

- **View results for all extracted objects on webpages (`aXe2web`; §5.2).** For convenient visualization and quality control, Web pages showing the basic results (direct image cutout of

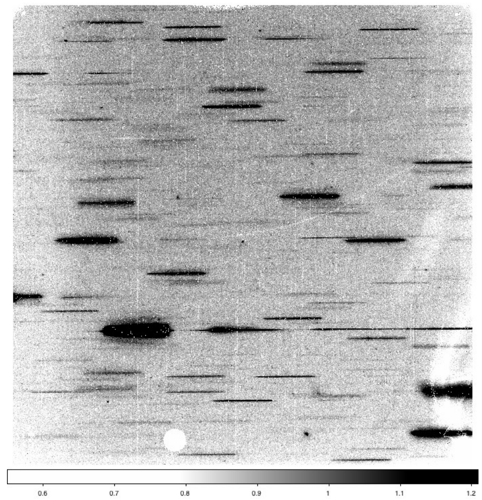

Figure 2: Example of one G141 grism exposure (`ib6o23rsq_flt.fits`) showing spectral traces of more than 100 sources.

the object, 2-dimensional spectral FITS image and extracted 1-dimensional spectrum in counts and flux units) for each object may be produced.

For the purpose of this cookbook we assume that the `PyRAF STSDAS` implementation of the aXe and aXeSIM packages is used. Commands issued on an xterm are indicated by the following symbol

```
>
```

whereas commands issued within a `PyRAF` environment are shown as

```
-->
```

For example, starting `PyRAF` and loading the aXe package would be shown as

```
> pyraf
--> stsdas
--> analysis
--> slitless
--> axe
```

In general `PyRAF` commands can also be conveniently run in Python. Below we show an example of a `PyRAF` command and the equivalent script for Python.

```
PyRAF:

--> axecore G141.lis WFC3.IR.G141.V2.5.conf extrfwhm=4.0 drzfwhm=3.0 back=no
   backfwhm=0.0 orient=no slitless_geom=no cont_model=gauss sampling=drizzle
```

An equivalent Python script would read as:

```
from pyraf import iraf
from iraf import stsdas, analysis, slitless, axe
iraf.axecore(inlist="G141.lis", configs="WFC3.IR.G141.V2.5.conf", extrfwhm=4.0,
   drzfwhm=3.0, back="no", backfwhm=0.0, orient="no", slitless_geom="no",
   cont_model="gauss", sampling="drizzle")
```

4.1 Setup of directories and files

Before any data reduction can start a sensible directory structure should be created and environment variables set. The directory structure we use here has the top level directory `G141_example` with sub-directories (e.g., `save, IMDRIZZLE, ...`) given as follows:

```
G141_example
    save
    IMDRIZZLE
    CONF
    DATA
    OUTPUT
    DRIZZLE
```

A brief description of the purpose for each directory is first given. The commands required to carry out the directory set-up then follow.

- **save**: A pristine copy of the original input data is kept in this directory. Before an aXe run, all content is copied to the 'DATA' directory. This separation is needed since the aXe reduction modifies the input data and it is best to keep a clean copy in case one wants to re-run the reduction with a different parameter set.
- **IMDRIZZLE**: This directory is used to create a deep direct image of the field with the `PyRAF` `astrodrizzle` or `multidrizzle` tasks. It is important to have a separate directory since the drizzle tasks modify the input images and this can lead to conflicts with other software packages such as aXe.
- **CONF**: All necessary aXe configuration and aXe reference files are stored here.
- **DATA**: This directory contains the input data `_flt.fits` files that will modified during the aXe reduction.
- **OUTPUT**: This directory stores intermediate output products and the results of the extractions from single grism images.
- **DRIZZLE**: The results of the `axedrizzle` reduction are stored here. These files comprise the final results where all the available grism data is combined together.

The steps that need to be carried out to complete the setup for the reduction process are:

- Step 1: Put the complete set of direct images and grism images in `_flt.fits` format (see Tab. 1) in the directory `G141_example`.
- Step 2: Create subdirectories and move the input data into the `save` directory.

Table 2: Required G141 Configuration and Reference Files

`WFC3.IR.G141.V2.5.conf`	G141 configuration file
`WFC3.IR.G141.0th.sens.1.fits`	G141 sensitivity 0^{th} order
`WFC3.IR.G141.1st.sens.2.fits`	G141 sensitivity $+1^{st}$ order
`WFC3.IR.G141.2nd.sens.2.fits`	G141 sensitivity $+2^{nd}$ order
`WFC3.IR.G141.3rd.sens.2.fits`	G141 sensitivity $+3^{rd}$ order
`WFC3.IR.G141.m1st.sens.2.5.fits`	G141 sensitivity -1^{st} order
`WFC3.IR.G141.flat.2.fits`	aXe G141 flat-field cube
`WFC3.IR.G141.sky.V1.0.fits`	G141 master sky image

```
> mkdir save IMDRIZZLE CONF DATA OUTPUT DRIZZLE
> mv *_flt.fits save
```

- Step 3: Point with environment variables to these directories by e.g., executing the following commands in your shell or placing equivalent commands in your `.bashrc` file.

  ```
  export AXE_IMAGE_PATH=DATA
  export AXE_CONFIG_PATH=CONF
  export AXE_OUTPUT_PATH=OUTPUT
  export AXE_DRIZZLE_PATH=DRIZZLE
  ```

  If you are using a tcsh/csh shell, please use the equivalent commands such as: `setenv AXE_IMAGE_PATH DATA`.

  Note that `axedrizzle` *does not accept path+file names that are longer than 80 characters, so please specify an* `AXE_DRIZZLE_PATH` *accordingly. This issue will be addressed a future version of aXe.*

- Step 4: Put the appropriate aXe configuration and aXe reference files and the master sky image into the `CONF` directory (see also Sec. 2.2 and the WFC3 Web pages). For the G141 grism the required files are listed in Table 4.1.

- Step 5: The final step of this section is to get an overview of the data by checking the data type and other information. It is often convenient to prepare ASCII files containing lists of images which one wants to work on. Information for each image can be derived from simple `IRAF` commands such as those given below.

```
> cd save
> pyraf
--> hedit *_flt.fits[1] FILTER . update-
ib6o23rsq_flt.fits[1],FILTER = G141
ib6o23rtq_flt.fits[1],FILTER = F140W
ib6o23ruq_flt.fits[1],FILTER = G141
ib6o23rwq_flt.fits[1],FILTER = F140W
ib6o23ryq_flt.fits[1],FILTER = G141
ib6o23rzq_flt.fits[1],FILTER = F140W
ib6o23s0q_flt.fits[1],FILTER = G141
ib6o23s2q_flt.fits[1],FILTER = F140W
```

An ASCII file list of F140W images named `F140W.lis` can be prepared with the help of an text editor. The `IRAF` task `hselect` may also be used to generate a file list based on header keyword values. The resulting file looks like this:

```
--> !more F140W.lis
ib6o23rtq_flt.fits
ib6o23rwq_flt.fits
ib6o23rzq_flt.fits
ib6o23s2q_flt.fits
```

In order to keep a clean copy of the original input data we copy the direct image files into the `IMDRIZZLE` directory where we will produce the combined deep image and the master object catalogue.

```
--> copy @F140W.lis ../IMDRIZZLE
--> copy F140W.lis ../IMDRIZZLE
```

4.2 Combination of direct imaging: `astrodrizzle` (and `multidrizzle`)

A deep image of all exposures in the observed FoV needs to be prepared. There are many advantages to combining multiple exposures. The most important ones are the rejection of deviant pixels (e.g. remaining cosmic rays, hot or bad pixels), and the increase in depth. Since in this example we have four dithered direct images available, we combine them using the `PyRAF` task `astrodrizzle` in the `STSDAS` package `DrizzlePac`.

We note that `astrodrizzle` *is new software that replaced* `multidrizzle` *in 2012. The* `aXe` *package, however, was originally written to support* `Multidrizzle` *formatted astrometric information. Beginning with* `aXe` *released in* `STSDAS` *version 3.16, aXe reduction tasks can accommodate the use of new Astrodrizzle products, while remaining back-compatible with Multidrizzle. For more detailed information on* `Astrodrizzle` *please see the* `DrizzlePac` *website. Here, use of both* `astrodrizzle` *and* `multidrizzle` *are illustrated, but adoption of* `astrodrizzle` *is strongly recommended since* `multidrizzle` *is no longer supported by STScI. In the basic grism reduction illustrated in this cookbook, a switch to Astrodrizzle products will only affect the aXe routine* `iolprep` *as described in §4.4 below.*

4.2.1 `Multidrizzle`

Here, we use the final version of `multidrizzle` which is available in STSDAS. For `multidrizzle` to work properly, we need to set up the local `"iref"` directory if this is not already available. Point with an environment variable to the directory in your shell or place the command in your `.bashrc` file as follows. Copy the needed version of the `IDCTAB` reference table (`u1r16228i_idc.fits` for the data used in this cookbook; see the STScI WFC3 Calibration Website), containing `multidrizzle`-style information on the geometric distortions, into the directory. Putting the `IDCTAB` table in the local directory with the direct images also works. For more details please refer to the WFC3 Data Handbook.

```
> export iref=/mydisk/myiref/
> cp u1r16228i_idc.fits /mydisk/myiref/
```

If you are instead using tcsh/csh, set the environment variable via `setenv iref /mydisk/myiref/`. In `PyRAF` continue with the direct image combination.

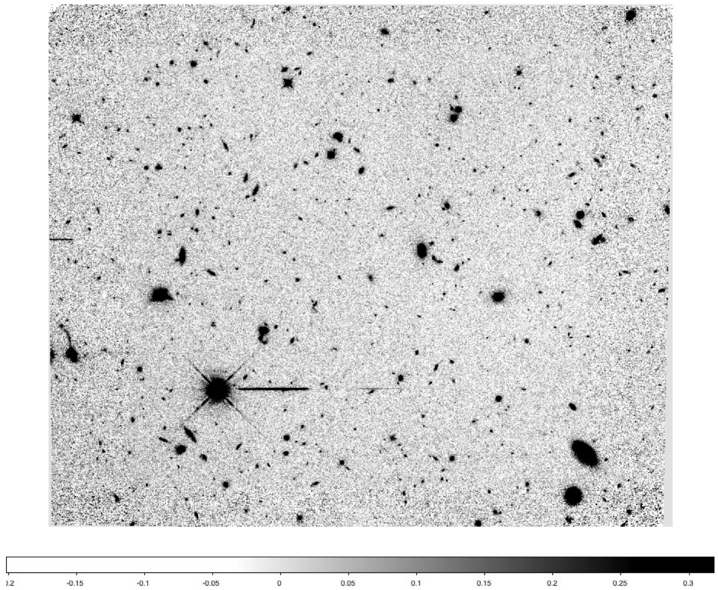

Figure 3: The “drizzled” direct image (F140W_drz.fits; [SCI] extension) showing the full FoV and covering about 160 × 133 arcsec. The horizontal streak next to the bright star originates from persistence effects on the detector caused by preceding grism observations of the same field (see also Tab. 1).

```
--> cd ../IMDRIZZLE
--> stsdas
--> dither
--> unlearn multidrizzle
--> multidrizzle '@F140W.lis' output='F140W' final_rot='INDEF'
```

The output is the co-added image `F140W_drz.fits` (see Fig. 3). Some artifacts (e.g., persistence, dead pixels) are clearly visible, and this should be kept in mind later on during catalogue creation.

4.2.2 `Astrodrizzle`

Here, we use `astrodrizzle` in `STSDAS drizzlepac`. The steps are analogous to those above for `multidrizzle`, except that the need for the `IDCTAB` reference table is now obviated.

```
--> import drizzlepac
--> from drizzlepac import astrodrizzle
--> unlearn astrodrizzle
--> astrodrizzle.AstroDrizzle('@F140W.lis',output='F140W',build=yes)
```

4.3 Source detection: `SExtractor`

Now that we have a deep, combined F140W image of the observed field, we can derive the needed information on object positions, sizes and major axis orientation angle from the image. A convenient and readily available task for this is `SExtractor` (Bertin & Arnouts 1996; see http://www.astromatic.net/software/sextractor).

- Step 1: generate a SExtractor default file (`default.sex`) and adjust the parameters file. For this simple example we used the default setup with the following adjustments[2]. Higher values of the following parameters will select brighter sources.

```
DETECT_MINAREA   7              # minimum number of pixels above threshold
DETECT_THRESH    3.0            # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
ANALYSIS_THRESH  3.0            # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
```

- Step 2: isolate the science and weight extensions of the co-added F140W image.

```
--> imcopy F140W_drz.fits[SCI] F140W_drz_sci.fits
--> imcopy F140W_drz.fits[WHT] F140W_drz_wht.fits
```

- Step 3: adjust the file responsible for the output list content (`default.param`). The necessary fields for aXe are:

```
NUMBER, X_IMAGE, Y_IMAGE, X_WORLD, Y_WORLD, A_IMAGE, B_IMAGE,
THETA_IMAGE, A_WORLD, B_WORLD, THETA_WORLD, MAG_AUTO
```

It is important that all SExtractor output fields given above are present. Additional fields do not disturb the following reduction.

- Step 4: run SExtractor to produce the master object catalogue. We use in this example the output catalogue name `F140W.cat`.

```
> sex -c default.sex -WEIGHT_IMAGE F140W_drz_wht.fits -MAG_ZEROPOINT 26.46
    -CATALOG_NAME F140W.cat F140W_drz_sci.fits
```

An example of the output catalogue file (`F140W.cat`) from the SExtractor task is given in the following:

```
#   1 NUMBER          Running object number
#   2 X_IMAGE         Object position along x                      [pixel]
#   3 Y_IMAGE         Object position along y                      [pixel]
#   4 X_WORLD         Barycenter position along world x axis       [deg]
#   5 Y_WORLD         Barycenter position along world y axis       [deg]
#   6 A_IMAGE         Profile RMS along major axis                 [pixel]
#   7 B_IMAGE         Profile RMS along minor axis                 [pixel]
#   8 THETA_IMAGE     Position angle (CCW/x)                       [deg]
#   9 A_WORLD         Profile RMS along major axis (world units)   [deg]
#  10 B_WORLD         Profile RMS along minor axis (world units)   [deg]
#  11 THETA_WORLD     Position angle (CCW/world-x)                 [deg]
#  12 MAG_AUTO        Kron-like elliptical aperture magnitude      [mag]
1   1068.231    145.935 5.3084350628e+01 -2.7692458920e+01    12.533     6.509 -44.3 0.0004379272 0.0002476719  52.8  18.2001
2   1043.301     60.690 5.3082445468e+01 -2.7689782110e+01     5.692     5.356 -43.5 0.0002022316 0.0001913925  52.0  18.2335
3   1154.522     29.426 5.3086403599e+01 -2.7687624369e+01     2.540     2.087 -66.2 8.912657e-05 7.600131e-05  80.3  22.6103
...
```

[2] A default SExtractor file can be dumped with the command `sex -dd > default.sex` and then modified with an editor. You will also need a `default.conv` file which you an easily find on the internet if it is not available on your system.

☞ *The creation of the master object catalogue is fundamental to the data reduction, since all further reduction steps critically depend on it. Only objects correctly detected at this stage can later on be extracted from the grism images. Furthermore, spurious object detections will lead to wrong contamination predictions and should therefore be avoided. We strongly recommend to overplot the SExtractor source catalogue (*`F140W.cat`*) on the drizzled direct image (*`F140W_drz.fits`*) as shown in Fig. 4. This can be achieved by producing, for example, a region file to be displayed in* `ds9` *or in* `IRAF` *with* `tvmark`*. An introduction to* `ds9` *can be found at e.g. http://chandra-ed.harvard.edu/learning_ds9overview.html. Please adjust the SExtractor list of detection parameters to achieve the desired result.*

☞ *It is also important to set the correct ABmag zero-point (*`-MAG_ZEROPOINT 26.46` *for F140W) in the SExtractor run, since these magnitudes (MAG_AUTO) are used by the aXe software to estimate the level of spectral overlap between sources (i.e. contamination). Always use the latest information on zero-points provided at http://www.stsci.edu/hst/wfc3/phot_zp_lbn.*

☞ *If the SExtractor information for individual objects or object classes is inadequate, then one can edit the master object catalogue with an editor or a scripting language. In this way one could, for example, also insert objects for which the position and size is known from other observations but fail to be detected on the WFC3 imaging. Object positions may e.g. be determined with the* `IRAF` *task* `imexam`*.*

☞ *If SExtractor cannot estimate a magnitude for an object it assigns a magnitude of "99". This value of magnitude leads to an error in the aXe reduction software. Either replace the magnitude with a realistic number or remove the object from the Master catalogue.*

☞ *SExtractor can be run with the option to produce a so called "segmentation map" (use* `-CHECKIMAGE_TYPE SEGMENTATION` *in the SExtractor call). The resulting extra FITS image is called* `check.fits` *and displays patches corresponding to pixels attributed to each object. The value of each patch corresponds to the SExtractor catalogue number.*

4.4 Projection of master object catalog to individual observations: `iolprep`

For typical grism observations, one obtains for each grism exposure a direct image at the same (or close to the same) position on the sky. The purpose of this section is to project the deep master object catalogue back to the individual direct images and create an individual catalogue for each grism image. If `multidrizzle` products are used, this step must be carried out in the same data directory where the `multidrizzle` coefficient files are stored (i.e. `IMDRIZZLE`).

We also use this step to prepare information on spectral cross-contamination estimates which aXe will derive in a later step if the catalogues are setup correctly. In its basic mode, the aXe software needs to know the magnitudes of all objects in the catalogue. For further details and information on computing more sophisticated contamination estimates see the aXe Manual.

- Step 1: modify the master object source list so that contamination estimates can be derived. This is triggered by renaming the column for the source brightness, MAG_AUTO to MAG_F1392 (column #12 in our example), with the pivot wavelength of the F140W filter being 1392 nm. We call this catalogue `F140W_prep.cat`. The pivot wavelength for all WFC3 filters is provided at http://www.stsci.edu/hst/wfc3/phot_zp_lbn.

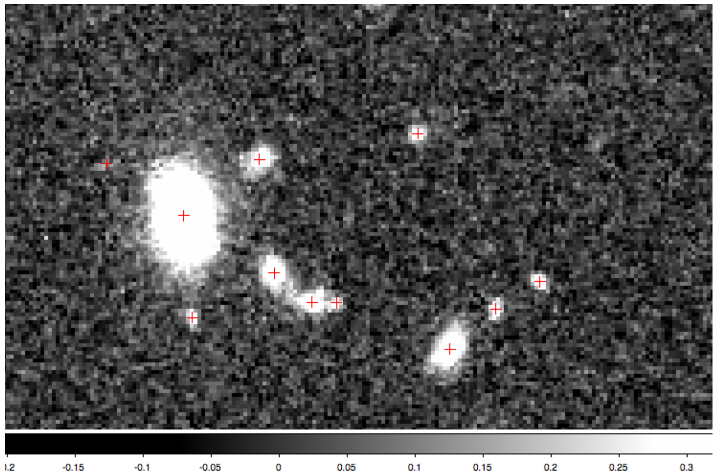

Figure 4: An F140W image subregion (24 × 14 arcsec) with sources indicated (plus signs) as detected by SExtractor.

- Step 2: load the aXe package and run the aXe task `iolprep` on the drizzled output image `F140W_drz.fits`

  ```
  --> axe
  --> unlearn iolprep
  ```

  For multidrizzled images:

  ```
  --> iolprep F140W_drz.fits F140W_prep.cat dimension_in='+100,0,0,0'
  ```

  For astrodrizzled images, one additional parameter must be set to direct the task to use the astrometric solution stored by astrodrizzle in the header, instead of the previous multidrizzle formatted astrometry:

  ```
  --> iolprep F140W_drz.fits F140W_prep.cat dimension_in='+100,0,0,0' useMdriz=no
  ```

  This `aXe` task uses the information in the drizzled image to produce a separate object catalogue for each individual input image. The four, comma separated values specified for the `dimension_in` parameter define the region (in units of pixels) outside the input image FoV which should be taken into account. This is important since for WFC3 IR G102 and G141 grism spectroscopy the $+1^{st}$ order spectrum is not located at the projected position of the object, but rather shifted to increasing x pixel position (see Fig. 5).

The task `iolprep` generates an individual object catalogue for each of the four input direct images. An example (`ib6o23rtq_flt_1.cat`) can be seen below.

```
...
# 10 B_WORLD  Profile RMS along minor axis (world units)  [deg]
# 11 THETA_WORLD  Position angle (CCW/world-x)  [deg]
# 12 MAG_F1392  Kron-like elliptical aperture magnitude  [mag]
1  865.301  159.466  5.3084350628e+01 -2.7692458920e+01  12.533  6.509 -47.5  0.0004379272  0.0002476719  52.8  18.2001
2  842.747  66.900  5.3082445468e+01 -2.7689782110e+01  5.692  5.356 -46.8  0.0002022316  0.0001913925  52.0  18.2335
3  949.859  32.507  5.3086403599e+01 -2.7687624369e+01  2.540  2.087 -68.0  0.0000891266  0.0000760013  80.3  22.6103
...
```

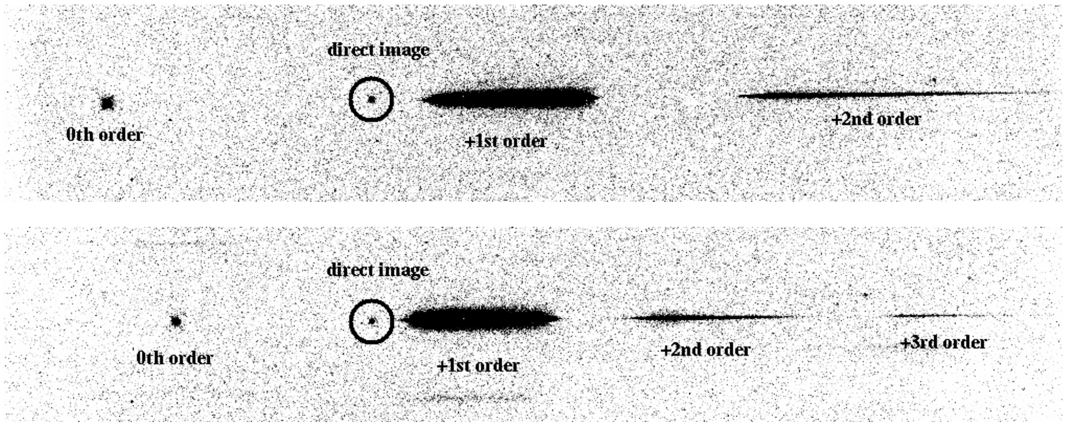

Figure 5: Relative position of direct image position and grism spectra. **Top:** G102 grism observation of the flux standard star GD153 (program 11552) with a F098M direct image (circled) superimposed to illustrate the relative positions. Spectral orders 0, +1, and +2 can be seen on the image. **Bottom:** G141 grism observation of the flux standard star GD153 (program 11552) with a F140W direct image (circled) superimposed to illustrate the relative positions. Spectral orders 0, +1, +2, and +3 can be seen in the image. The images show the full extent of the detector in the x-axis (1014 pixels) and about 200 pixels in the y-axis.

4.5 Building Input Image Lists

In order to run the data reduction tasks, we need to determine the correct associations between direct image and grism images for the whole dataset. In this section we describe the simple steps needed to achieve this.

- Step 1: determine for each grism image the associated direct image. For example, this can be performed with the tasks `dfits` and `fitsort` (see ESO's "Science Archive Stand-alone FITS Tools in ANSI C"). Alternatively, `IRAF hedit` or `hselect` can be used. As can be seen from the list below, there is always a pair of grism and direct image taken at a similar position.

```
> cd mypath/G141_example/save
> dfits *_flt.fits | fitsort FILTER POSTARG1 POSTARG2 RA_TARG DEC_TARG

FILE               FILTER  POSTARG1   POSTARG2   RA_TARG            DEC_TARG
ib6o23rsq_flt.fits G141    -10.012000 5.058000   5.307083333333E+01 -2.771111111111E+01
ib6o23rtq_flt.fits F140W   -18.838249 5.026986   5.307083333333E+01 -2.771111111111E+01

ib6o23ruq_flt.fits G141    9.988000   5.050000   5.307083333333E+01 -2.771111111111E+01
ib6o23rwq_flt.fits F140W   1.161730   5.053683   5.307083333333E+01 -2.771111111111E+01

ib6o23ryq_flt.fits G141    9.971000   -5.045000   5.307083333333E+01 -2.771111111111E+01
ib6o23rzq_flt.fits F140W   1.162243   -5.041331   5.307083333333E+01 -2.771111111111E+01

ib6o23s0q_flt.fits G141    -9.958000  -5.054000   5.307083333333E+01 -2.771111111111E+01
ib6o23s2q_flt.fits F140W   -18.766710 -5.084905   5.307083333333E+01 -2.771111111111E+01
```

- Step 2: prepare an Input Image List (G141.lis) with an editor of your choice. This ASCII file lists in a given row for each grism image the object list to be extracted and the associated direct image. The format is

  ```
  [name of grism image] [object catalogue] [name of direct image]
  ```

  In our example, the file G141.lis would be:

  ```
  ib6o23rsq_flt.fits ib6o23rtq_flt_1.cat ib6o23rtq_flt.fits
  ib6o23ruq_flt.fits ib6o23rwq_flt_1.cat ib6o23rwq_flt.fits
  ib6o23ryq_flt.fits ib6o23rzq_flt_1.cat ib6o23rzq_flt.fits
  ib6o23s0q_flt.fits ib6o23s2q_flt_1.cat ib6o23s2q_flt.fits
  ```

- Step 3: Copy the Input Image List (`G141.lis`) from the `save` directory to the `G141_example` directory level. Copy all grism images and direct images from the `save` directory and the associated catalogues (e.g., `ib6o23rtq_flt_1.cat`) from the `IMDRIZZLE` directory to the `DATA` directory.

  ```
  > cd mypath/G141_example
  > cp save/G141.lis .
  > cd DATA
  > cp ../save/*.fits .
  > cp ../IMDRIZZLE/i*.cat .
  ```

4.6 Preparing the spectral extraction: `axeprep`

Before the real spectral extraction can start, the grism images need to be pre-processed. For WFC3 IR grism images the main purpose here is the subtraction of a scaled master sky background image. For other HST grism images (e.g., ACS/WFC or NICMOS) this task is also used to convert the grism images to units of $[e^{-}/s]$.

Due to the combination of ubiquitous overlap of the various grism orders and large scale detector flat-field structure, the background exhibits a marked structure. This background can be removed by subtracting a scaled master sky background (see Fig. 6) from each grism image. The scaling is necessary since overall background levels vary significantly. For example, the G141 background typically shows average values between $0.9 - 2.4$ e/s. Besides the Input Image List (`G141.lis`; including the object catalogue) and the name of the master sky background (`WFC3.IR.G141.sky.V1.0.fits`) this task also requires as input the configuration file for this grism mode (`WFC3.IR.G141.V2.5.conf`). This is mainly needed here to determine grism image regions free of spectral traces from objects and thus use those "true" sky regions to scale the master background.

☞ *If the catalogues are too deep then one can end up in a situation with essentially no free sky regions. In such a case it is worth generating a special configuration file for this task with a lower object magnitude limit. See the aXe Manual for more details.*

```
--> cd mypath/G141_example
--> unlearn axeprep
--> axeprep G141.lis WFC3.IR.G141.V2.5.conf backgr='yes'
      backims='WFC3.IR.G141.sky.V1.0.fits' norm='no'
```

Figure 6: **Left:** The G141grism image `ib6o23rsq_flt.fits` before background subtraction. Structure caused by detector sensitivity variations is most prominently seen in the lower right corner. **Middle:** Master sky background for the G141 grism `WFC3.IR.G141.sky.V1.0.fits`. This image was created by combining more than 100 publicly available grism observations and carefully excluding regions with spectral signatures from objects. The dark circular region at the bottom is a region of zero sensitivity on the detector. **Right:** The background subtracted grism image `ib6o23rsq_flt.fits`. Note, that the small scale as well as large scale background structure has been removed.

☞ *Note, that the* `axeprep` *task has physically changed the* `_flt.fits` *files for the grism images in the* `DATA` *directory by removing the background. If one wants to start a clean new reduction for whatever reason, the files need to be copied again from the* `save` *directory.*

Congratulations! At this point you have concluded the preparations and are finally ready to run the spectral extraction tasks.

4.7 Extracting individual spectra: `axecore`

This section describes the core spectral extraction task which produces 2-dimensional and 1-dimensional spectra for each object in the Input Object List (the projected catalogue) for a given grism image. The task also applies the flat-field information which is noteworthy since the flat-field information is stored in a 3-dimensional, cube format. In slitless spectroscopy every pixel can in principle see signal from a portion of a spectrum with a given but a priori unknown wavelength. The response function of a detector is generally wavelength dependent and for the WFC3 IR detector this information was determined during thermal vacuum ground testing and stored in a flat-field cube.

In the aXecore task, the object position and size from a given F140W direct image is projected to the associated G141 grism image. The region containing the spectral information for each source and grism order (dubbed a "beam") is then determined and the pixel values are flat-fielded using the wavelength solution for this object. Furthermore, potential contamination from neighboring and/or overlapping objects is estimated. As main output, the task produces 2-dimensional spectra ("stamp" images; `.STP.fits` files) and 1-dimensional summed up spectra (`.SPC.fits` files), where information for each object in the catalogue is stored in a separate FITS extension (for more detailed information on the output formats see Sec. 4.9).

The task, which will run for a while (∼30 min, depending on processing power), is called as follows. In order to start with a clean setup it is prudent to 'unlearn' the parameter set for axecore.

```
--> unlearn axecore
```

```
--> axecore G141.lis WFC3.IR.G141.V2.5.conf extrfwhm=4.0 drzfwhm=3.0
       back='no' backfwhm=0.0 orient='no'  slitless_geom='no'
       cont_model='gauss' sampling='drizzle'
```

The main inputs to the task are the Input Image List (`G141.lis`) and the configuration file for the G141 grism (`WFC3.IR.G141.V2.5.conf`). Further important parameters and their meaning are given in Table 3 (see Appendix A.3 for a full list of axecore parameters).

Table 3: Important `axecore` parameters

Parameter	Value	Comment
`extrfwhm`	4.0	Multiplicative factor for object mask image
`drzfwhm`	3.0	Multiplicative factor for object mask image after drizzle
`back`	'no'	Create background PET?
`backfwhm`	0.0	Multiplicative factor for background mask image
`orient`	'no'	Use tilted extraction?
`slitless_geom`	'no'	Optimize virtual slit geometry?
`cont_model`	'gauss'	Setting contamination model
`sampling`	'drizzle'	Setting the sample type

The extraction width `extrfwhm` is specified to be ± 4 times the projected width of the source in the direction perpendicular to the spectrum trace. Exactly how this is interpreted in terms of the `A_IMAGE` and `B_IMAGE` values in the Sextractor master object catalogue depends on the values of the parameters `orient` and `slitless_geom`. The aXe manual provides more details. In this simple example we extract the spectrum with a perpendicular pseudo slit orientation, regardless of the orientation of the (extended) sources on the sky. The extraction of the final spectra will be performed from the drizzled, combined spectral images with an extraction width of `drzfwhm` (see Sec. 4.8) and for technical reasons `extrfwhm` should be at least 1 unit larger than `drzfwhm`. For the WFC3/IR grisms a final extraction factor of `drzfwhm` = 3 corresponds for point sources to extracting an aperture of 6 pixels and thus about 90% of the flux (for aperture corrections see Kuntschner et al. 2009).

Another important parameter to set in the `axecore` task is the type of contamination model to be used. The 'gauss' model provides quantitive estimates of spectral overlap based on a Gaussian model of the source luminosity distribution (SExtractor `A_IMAGE` and `B_IMAGE`) and its magnitude (`MAG_F1392` in the projected source catalogue).

Since we want to combine the data from different exposures with the axedrizzle task (see Sec. 4.8) and also provide a linear wavelength sampling for the extracted 2-dimensional spectral images, we choose here the sampling parameter to be 'drizzle'. For more details please refer to the aXe Manual.

The various output files of this task are stored in the `OUTPUT` directory. For a detailed description of the output and data formats we refer the reader to Sec. 4.9

4.8 Combining dithered exposures: `axedrizzle`

In this section we use the task `aXedrizzle` to co-add the individual, dithered exposures ('beams') for each object to obtain a deep 2-dimensional spectrum with a pre-defined, constant dispersion in wavelength direction and a linear spatial sampling. From this product, the final 1-dimensional spectra are extracted for each object.

Similar techniques used in the `astrodrizzle` and `multidrizzle` tasks for direct imaging are used by `aXedrizzle` to combine spectral images, and in the process identify and reject deviating pixels (e.g. undetected cosmic rays, un-flagged bad pixels). This is accomplished by individually drizzling the 2-dimensional spectra, median-combining them, blotting them back and then identifying deviating pixels by comparing the original 2-dimensional spectra with the blotted back versions.

The setup for the main task `aXedrizzle` is performed in a pre-processing step (`drzprep`).

```
--> unlearn drzprep
--> drzprep G141.lis WFC3.IR.G141.V2.5.conf back='no'
```

In this preparatory step, flat-fielded spectral images are produced from our internal data storage format ("PET" files) and also drizzle coefficients are determined. The images are stored in the `.DPP.fits` files.

The main combination of individual 2-dimensional spectra is carried out with the task `axedrizzle`:

```
--> unlearn axedrizzle
--> axedrizzle G141.lis WFC3.IR.G141.V2.5.conf 4.0 3.0
              back='no' driz_separate='yes' combine_nsigmas="4.0 3.0"
              driz_cr_snr="5.0 4.0" driz_cr_grow=1 driz_cr_scale="1.2 0.7"
```

Apart from the Input Image List and the configuration file, the `axedrizzle` task requires a large set of parameters. The parameters `infwhm` and `outfwhm` (the 3^{rd} and 4^{th} input parameters in the `axedrizzle` call above, i.e. 4.0 and 3.0) mirror the `axecore` parameters `extrfwhm` and `drzfwhm` and must be set to the same values. Furthermore, the parameter 'back' must be set to the same value in the tasks `axecore`, `drzprep` and `axedrizzle`.

In this example, we use standard drizzle and rejection parameters which have the same meaning as in the `astrodrizzle` and `multidrizzle` tasks. It is strongly recommended to carefully set and test these parameters against a "basic" drizzled spectral image (`driz_separate='no'`) before any extracted spectra are used for scientific purposes.

With the completion of the `axedrizzle` task, the actual data-reduction of grism spectra is concluded and the output data is ready for inspection. In Sec. 4.9 we describe in some detail the various output products and formats. Note that currently the drizzling is only performed for the $+1^{st}$ order.

4.9 Description and summary of output products

In the course of the data reduction described above, many output files are produced including 1-dimensional spectra and 2-dimensional spectral ('stamp') images, which are stored in the `OUTPUT` and `DRIZZLE` directories. All results for one particular object are marked with the so called *object ID*, which is the entry in the column `NUMBER` in the SExtractor master object catalogue (see Sec. 4.3) plus one letter indicating the extracted grism order ($+1^{st}$ order = A, 0^{th} order = B, $+2^{nd}$ order = C, ...- see aXe Manual and configuration file for further details). Typically the throughput of the grism is concentrated in the $+1^{st}$ order and is therefore used for the extraction of the science spectra.

The extracted 1-dimensional spectra are stored in the `SPC`-files. Grism 'stamp' images are stored in the `STP`-files, and the co-added, 2-dimensional, drizzled spectral images are the `MEF`-files. Important information on the extraction geometry is stored in the `OAF`-files. We will now explain the most important details of these files and the data format.

4.9.1 `SPC`-file – extracted 1-dimensional spectra

The `SPC`-files have the suffix `.SPC.fits`, e.g. `OUTPUT/ib6o23rsq_flt_2.SPC.fits`. While an `SPC`-file in the `OUTPUT` directory (e.g. `ib6o23rsq_flt.fits`) contains all individually extracted spectra from one grism image (including all BEAMS or spectral orders), only the `SPC`-file in the `DRIZZLE` directory, `DRIZZLE/aXeWFC3_2.SPC.fits`, contains the $+1^{st}$ order 1-dimensional spectra extracted from all co-added drizzled images.

The `SPC`-files are multi-extension FITS tables. A list of all extensions can be obtained with the `PyRAF` task `catfits`. Alternatively the FITS files can be explored with TOPCAT, the `IDL` task `mrdfits.pro`, or any other FITS reader of your choice.

```
--> stsdas
--> catfits DRIZZLE/aXeWFC3_2.SPC.fits
EXT#  FITSNAME      FILENAME     EXTVE DIMENS    BITPI OBJECT

0     DRIZZLE/aXeWF                               16
1       BINTABLE    BEAM_1A                      15Fx201R          1
2       BINTABLE    BEAM_10A                     15Fx186R          10
3       BINTABLE    BEAM_100A                    15Fx186R          100
4       BINTABLE    BEAM_101A                    15Fx185R          101
5       BINTABLE    BEAM_102A                    15Fx185R          102
6       BINTABLE    BEAM_103A                    15Fx186R          103
...
```

The $+1^{st}$ order spectrum of the object with the ID 101 is stored in the extentsion with the name `BEAM_101A`, which has the index 4 in the above example. Only $+1^{st}$ order spectra ('beams') are drizzled, hence all extension names end with the letter `A`. Every extension of the `SPC`-file is itself a FITS table with the columns:

```
--> tlcol DRIZZLE/aXeWFC3_2.SPC.fits[beam_101A]
# DRIZZLE/aXeWFC3_2.SPC.fits[BEAM_101A]
ID                CH*60       %-60s ""
N                 S             %11d ""
LAMBDA            R           %15.7g ANGSTROM
TCOUNT            R           %15.7g COUNT
BCOUNT            R           %15.7g COUNT
COUNT             R           %15.7g COUNT
TERROR            R           %15.7g COUNT
BERROR            R           %15.7g COUNT
ERROR             R           %15.7g COUNT
FLUX              R           %15.7g "PHYSICAL UNITS"
FERROR            R           %15.7g "PHYSICAL UNITS"
WEIGHT            R           %15.7g PIXEL
CONTAM            R           %15.7g FLAG
DLAMBDA           R           %15.7g ANGSTROM
DQ                S             %11d DQ
```

The most important columns are:

- `LAMBDA`: the wavelength in Ångstrom
- `COUNT`: the spectrum in units of counts $[e^{-}/s]$
- `ERROR`: the error for `COUNT`
- `FLUX`: the spectrum in $[erg/cm^2/s/\text{Å}]$

- FERROR: the error for FLUX

Plotting the spectrum in [counts/sec] of object 101 in PyRAF can be done via:

```
--> sgraph "DRIZZLE/aXeWFC3_2.SPC.fits[beam_101A] LAMBDA COUNT"
```

4.9.2 STP-file – extracted 2-dimensional spectral images

The STP-files have the suffix .STP.fits and follow the identical naming scheme as the SPC-files. Consequently, the file OUTPUT/ib6o23rsq_flt_2.STP.fits contains all grism stamp images from the grism image ib6o23rsq_flt.fits and DRIZZLE/aXeWFC3_2.STP.fits the grism stamps from all drizzled images.

The STP-files are multi-extension FITS images. A list of all extensions is obtained with the PyRAF task catfits in the STSDAS package:

```
--> catfits OUTPUT/ib6o23rsq_flt_2.STP.fits
EXT#  FITSNAME      FILENAME              EXTVE DIMENS       BITPI OBJECT

0     ib6o23rsq_flt ib6o23rsq_flt.fits                       16
1       IMAGE       BEAM_1A                     134x78       -32   1
2       IMAGE       WEIG_1A                     134x78       -32   1
3       IMAGE       BEAM_1B                     33x77        -32   1
4       IMAGE       WEIG_1B                     33x77        -32   1
5       IMAGE       BEAM_1C                     10x10        -32   1
6       IMAGE       WEIG_1C                     10x10        -32   1
7       IMAGE       BEAM_1D                     10x10        -32   1
8       IMAGE       WEIG_1D                     10x10        -32   1
9       IMAGE       BEAM_1E                     208x78       -32   1
10      IMAGE       WEIG_1E                     208x78       -32   1
11      IMAGE       BEAM_2A                     155x38       -32   2
12      IMAGE       WEIG_2A                     155x38       -32   2
13      IMAGE       BEAM_2B                     33x36        -32   2
14      IMAGE       WEIG_2B                     33x36        -32   2
...
```

The grism stamp of object 198, $+1^{st}$ order, is stored in the extension with the name BEAM_198A. Please note that the results from the individual extractions contain also the other spectral orders (-1^{st}, 0^{th}, $+2^{nd}$, $+3^{rd}$ orders) as indicated with the letters E, B, C, D in the extension names.

The STP-file extensions can be displayed conveniently with any FITS-image viewer such as ds9.

4.9.3 MEF-file – calibrated spectral stamps

The MEF-files are named as DRIZZLE/aXeWFC3_mef_ID<object ID>.fits, e.g. the MEF-file of object 101 is DRIZZLE/aXeWFC3_mef_ID101.fits. They exist for drizzled objects only, thus all reside in the directory DRIZZLE.

The MEF-files are multi-extension FITS images. In contrast to the STP-files, they contain a fully calibrated and linearized wavelength, as well as spatial coordinate, system. Furthermore, they provide information on errors and an estimate of the contamination. *We recommend to use the MEF-files rather than the STP-files as input for any science investigations.* A list of all extensions in a MEF-file can be obtained with the PyRAF task catfits in the STSDAS package:

```
--> catfits DRIZZLE/aXeWFC3_mef_ID101.fits
EXT#  FITSNAME      FILENAME EXTVE DIMENS       BITPI OBJECT

0     DRIZZLE/aXeWF                             8
1       IMAGE       SCI      1      213x34      -32   101
2       IMAGE       ERR      1      213x34      -32   101
3       IMAGE       EXPT     1      213x34      -32   101
4       IMAGE       CON      1      213x34      -32   101
5       IMAGE       WHT             213x34      -32
```

The important extensions are:

- `SCI`: the co-added (drizzled) flux values in units of $[e^-/s]$
- `ERR`: the error for `SCI`
- `EXPT`: the effective exposure time for each pixel in units of $[s]$
- `CON`: an estimate of the contamination from neighbouring object spectra $[e^-/s]$

The `MEF`-files can be displayed conveniently with any FITS-image viewer, such as `ds9`, which will also display the coordinate system along the wavelength and cross-dispersion axis.

☞ *Since the* `MEF`*-files are fully calibrated with a WCS header showing the wavelength in x and the spatial direction in y, advanced users can make individual spectral extractions from the* `MEF`*-files using customized extraction boxes.*

☞ *The* `STP`*-file extensions of the drizzled images,* `DRIZZLE/aXeWFC3_2.STP.fits` *contain almost the identical information as the* `SCI`*-extension in the corresponding* `MEF`*-files. While the* `STP`*-file is necessary for the web page creation (see Sec. 5.2), the* `MEF`*-files with the WCS header and additional information should preferrentially be used when examining single object results.*

4.9.4 `OAF`-file – extraction geometry

The `OAF`-files have the suffix `.OAF`, e.g. `OUTPUT/ib6o23rsq_flt_2.OAF` or `DRIZZLE/aXeWFC3_2.OAF`. `OAF`-files are ASCII files (i.e. not FITS !) that contain all geometrical extraction information for all spectral orders and objects. `OAF`-files can be displayed via:

```
> more DRIZZLE/aXeWFC3_2.OAF
APERTURE 1
  BEAM A
     REFPIXEL1A 14.0 42.0
     CORNERS1A 1.0 1.0 201.0 1.0 201.0 84.0 1.0 84.0
     CURVE1A   1 0.0 0.0
     WIDTH1A   27.89
     ORIENT1A  90.0
     SLITGEOM1A 0.0 0.0 7.95 0.0
     IGNORE1A   0
  BEAM END
APERTURE END
APERTURE 10
```

```
  BEAM A
     REFPIXEL10A 14.0 14.0
     CORNERS10A 1.0 1.0 186.0 1.0 186.0 28.0 1.0 28.0
     CURVE10A   1 0.0 0.0
     WIDTH10A   7.15
     ORIENT10A  90.0
     SLITGEOM10A 0.0 0.0 1.82 0.0
     IGNORE10A   0
  BEAM END
APERTURE END
```

The information for one object is enclosed with `APERTURE <ID> ... APERTURE END`. For each object the information on the various spectral orders are enclosed in `BEAM <letter> ...BEAM END`; `<letter>` being the corresponding letter for the spectral order.

Probably the most important information in the `OAF`-files are the extraction widths, given with the keyword `WIDTH<ID><letter>`.

The line:

```
WIDTH1A   27.98
```

in the file `DRIZZLE/aXeWFC3_2.OAF` means that the extraction width of the object number 1 was 27.98 pix on each side of the trace. This means the the spectrum `DRIZZLE/aXeWFC3_2.SPC.fits[BEAM_1A]` was extracted from the `SCI`-extension of the `MEF`-file `DRIZZLE/aXeWFC3_mef_ID1.fits` with the total extraction width of $\pm 27.98 = 2.0 * 27.98 = 55.96$ pix.

5 Quality control and simulations with aXeSIM

Running highly automated tasks as described in this cookbook enables the extraction of several 100 or 1000 spectra from deep grism data. Nevertheless, it is still mandatory to check and validate the products before any science investigation can commence. In Sects. 5.1, 5.2 and 5.3 we describe basic checks and tools which can help to validate data quality and understand the limits of the observations.

5.1 Basic data products check

Here we provide a list of suggested, basic checks users can carry out to validate the data reduction quality. Some hints on troubleshooting are given in Sec. 6.

- **Check grism background after `axeprep`.** After running the `axeprep` task with a master sky the resulting grism images should show a smooth background level with zero counts.
- **Contamination:** A convenient means of checking the quality of the contamination estimates is to blink the grism image (`ib6o23rsq_flt.fits`) with the contamination image (`ib6o23rsq_flt_2.CONT.fits`) in, e.g. `ds9`. In our example the files are stored in the `OUTPUT` directory. The trace locations and flux levels of the contamination image should resemble the real data.
- **Individual extraction versus drizzled spectrum:** The quality of the drizzled 1-dimensional 1^{st} order spectra (file: `aXeWFC3_2.SPC.fits`; `DRIZZLE` directory) can be checked by overplotting individual extractions (files: `ib6o23rsq_flt_2.SPC.fits, ib6o23ruq_flt_2.SPC.fits, ib6o23ryq_flt_2.SPC.fits, ib6o23s0q_flt_2.SPC.fits`; `OUTPUT` directory).
- **Compare achieved S/N with expected values from the ETC.** For a true point source, compare the observed S/N with the expectations from the ETC.

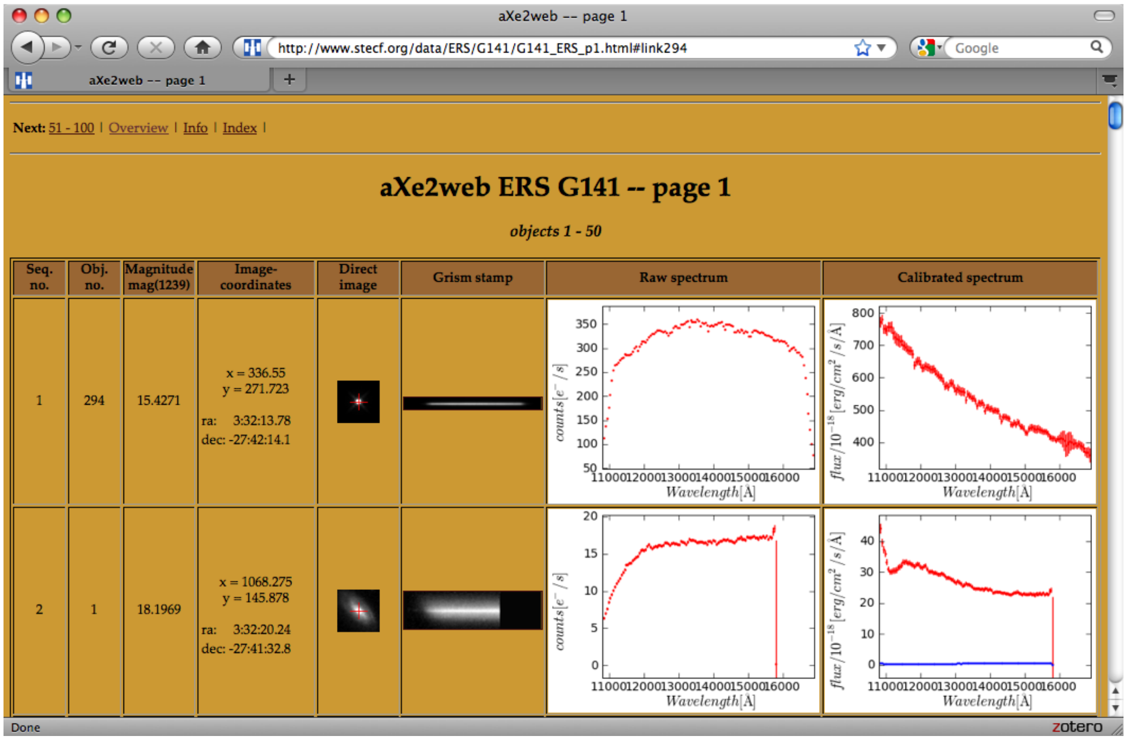

Figure 7: An HTML page for the ERS G141 grism data as produced by aXe2web. The 1^{st} column with an index number is followed by the object ID number, the object brightness and a column with coordinate information. The following columns show a direct image stamp (6.5×6.5 arcsec), the 2-dimensional spectrum image and the extracted 1-dimensional spectrum in units of [e^{-}/s] and [erg/cm^2/s/Å]. If non-zero, an estimate of the contamination level (blue) is provided for the spectrum in physical units.

5.2 Web page creation with `aXe2web` for visual inspection

The main purpose of this section is to create Web pages for easy visualization of the aXe output products and basic quality checks. The Web pages (see Fig. 7) provide, for each extracted source in the master object catalogue, information on object magnitude, coordinates, a stamp image extracted from the direct image and the full spectral information.

Besides the Web pages, which show the details on individual objects, there is also a master index page, a page with an overview over all objects, as well as an info page which summarizes the parameter setup used for the `aXe2web` run. All pages are crosslinked to enable fast navigation within the dataset.

`aXe2web` is a stand alone python code which is not distributed as part of the `PyRAF STSDAS` package, however, it can be obtained from the following link, including documentation: `aXe2web`.

The following steps need to be carried out in order to produce the Web pages:

- Step 1: create a new directory

```
> mkdir visualization
> cd visualization
```

- Step 2: copy all needed input files to this directory. The input files are:
  - the detection image `F140W_drz_sci.fits`
  - the magnitude column adjusted master object catalogue `F140W_prep.cat`
  - the extracted 1-dimensional spectra from the DRIZZLE directory (`aXeWFC3_2.SPC.fits`)
  - the stamp image file from the DRIZZLE directory (`aXeWFC3_2.STP.fits`)

  ```
  > cp ../IMDRIZZLE/F140W_drz_sci.fits .
  > cp ../IMDRIZZLE/F140W_prep.cat .
  > cp ../DRIZZLE/aXeWFC3_2.SPC.fits .
  > cp ../DRIZZLE/aXeWFC3_2.STP.fits .
  ```

- Step 3: make a parameter ASCII file to define the input, such as 'G141.par':

  ```
  #
  # Example 1
  #
  pagename = G141_ERS
  direct_image = F140W_drz_sci.fits
  stamp_image = aXeWFC3_2.STP.fits
  spectr_table = aXeWFC3_2.SPC.fits
  sextract_cat = F140W_prep.cat
  sort_column = MAG_F1392+
  page_header = "aXe2web ERS G141"
  pagesize=50
  error_bars=yes
  mark_sources=yes
  outputpath=web_pages
  lambda_range=11000,17000
  #
  ```

- Step 4: run `aXe2web` with the parameter file:

  ```
  > aXe2web --parfile G141.par
  ```

The Web pages created by `aXe2web` provide a convenient means for obtaining an overview over several 100 spectra and also help to quickly identify data reduction problems.

5.3 Simulations with aXeSIM

Simulating slitless spectroscopic images with aXeSIM can assist in interpreting the spectra extracted from a data set. As an example, we simulate continuum point sources at various flux levels in order to estimate the SNR as a function of the F140W broad band magnitude.

The identical directory setup generated for the extraction described in Sec. 4 can be used in the simulations. Moreover the configuration and calibration files necessary for aXeSIM are identical to the ones needed for aXe.

The simulation is prepared as follows:

```
> cp DATA/ib6o23rtq_flt_1.cat DATA/SNR_sim.cat
```

With a text editor, the file `SNR_sim.cat` is modified to provide the object parameters for the simulations. For example, we set up a configuration of sources aligned in x direction (X = 500) and varying Y-position. The source magnitudes are also set to different values. The file reads as follows:

```
# 1  NUMBER  Running object number
# 2  X_IMAGE  Object position along x  [pixel]
# 3  Y_IMAGE  Object position along y  [pixel]
# 4  X_WORLD  Barycenter position along world x axis  [deg]
# 5  Y_WORLD  Barycenter position along world y axis  [deg]
# 6  A_IMAGE  Profile RMS along major axis  [pixel]
# 7  B_IMAGE  Profile RMS along minor axis  [pixel]
# 8  THETA_IMAGE  Position angle (CCW/x)  [deg]
# 9  A_WORLD  Profile RMS along major axis (world units)  [deg]
# 10 B_WORLD  Profile RMS along minor axis (world units)  [deg]
# 11 THETA_WORLD  Position angle (CCW/world-x)  [deg]
# 12 MAG_F1392  Kron-like elliptical aperture magnitude  [mag]
 1  500.0  400.0  2.91087e+02  9.893e+00  1.0  1.0  90.0  1.0  1.0  90.0  20.0
 2  500.0  420.0  2.91087e+02  9.893e+00  1.0  1.0  90.0  1.0  1.0  90.0  20.5
 3  500.0  440.0  2.91087e+02  9.894e+00  1.0  1.0  90.0  1.0  1.0  90.0  21.0
 4  500.0  460.0  2.91087e+02  9.895e+00  1.0  1.0  90.0  1.0  1.0  90.0  21.0
 5  500.0  480.0  2.91087e+02  9.895e+00  1.0  1.0  90.0  1.0  1.0  90.0  22.5
 6  500.0  500.0  2.91087e+02  9.896e+00  1.0  1.0  90.0  1.0  1.0  90.0  22.5
 7  500.0  520.0  2.91086e+02  9.897e+00  1.0  1.0  90.0  1.0  1.0  90.0  23.0
 8  500.0  540.0  2.91086e+02  9.897e+00  1.0  1.0  90.0  1.0  1.0  90.0  23.0
 9  500.0  560.0  2.91086e+02  9.898e+00  1.0  1.0  90.0  1.0  1.0  90.0  23.5
10  500.0  580.0  2.91086e+02  9.899e+00  1.0  1.0  90.0  1.0  1.0  90.0  23.5
11  500.0  600.0  2.91086e+02  9.899e+00  1.0  1.0  90.0  1.0  1.0  90.0  24.0
12  500.0  620.0  2.91086e+02  9.900e+00  1.0  1.0  90.0  1.0  1.0  90.0  24.0
```

In total, 12 objects with different brightness in the F140W filter are simulated. The objects are located at positions starting at $(x, y) = (500, 400)$ and then in steps of $\Delta y = 20$.

In the setup prepared for the extraction, a simulation of a grism image with an exposure time of $1\,hour$ and a typical sky background of $0.9\,e/s$ is performed with:

```
> pyraf
--> stsdas
--> slitless
--> axesim
--> simdata incat='SNR_sim.cat' config='WFC3.IR.G141.V2.5.conf'
            output_root='WFC3_G141_SNR' bck_flux_disp=0.9  silent='yes'
            exptime_disp=3600.0 extraction='yes'
```

The provided background value (`bck_flux_disp`) is interpreted to be constant across the FoV and can be adjusted to a specific observed value. Furthermore, one can provide a full background image instead of a constant value if the shape of the background (see Fig. 6) is important.

Fig. 8 shows the central part of the resulting simulated image `WFC3_G141_SNR_slitless.fits`. For the brighter simulated objects one can see the 0^{th}, $+1^{st}$ and $+2^{nd}$ orders from left to right. As expected, the traces become fainter or vanish with increasing y-axis position.

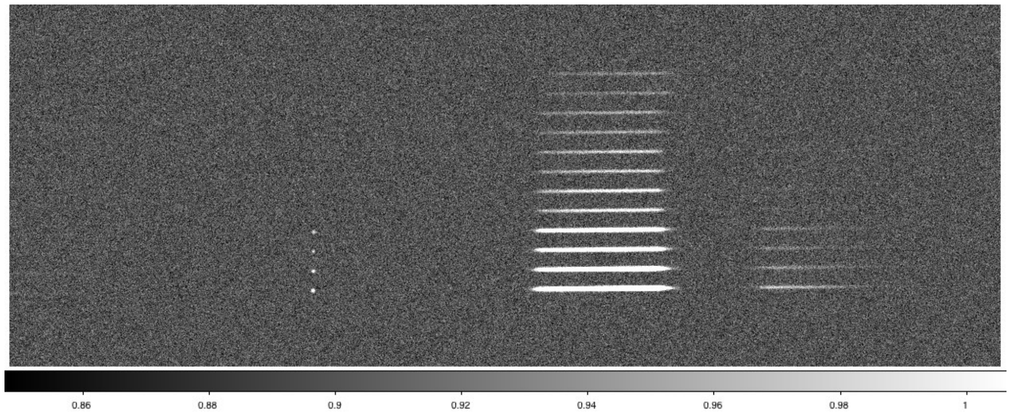

Figure 8: The simulated G141 image `WFC3_G141_SNR_slitless.fits` (central cutout with full x-extension and $\sim 370\,\mathrm{pix}$ in y). In this example, only point-like objects are shown.

An additional output of the simulations are the spectra extracted from the simulated grism image using methods (and even the code) from aXe. The extracted spectra are stored in the `SPC`-file `WFC3_G141_SNR_slitless_2.SPC.fits`. An analysis of the SNR of all spectra stored in the `SPC`-file, e.g. using `DER_SNR.py`[3], then leads to the SNR as a function of F140W magnitude as shown in Fig. 9. The SNR computed from the extracted spectra in `DRIZZLE/aXeWFC3_2.SPC.fits` can then be compared to the simulated numbers in Fig. 9. Since the size of an object is a strong function in S/N, only observed and simulated objects of similar size and shape should be compared.

For a more advanced use of the simulation package, including the simulation of direct images and user-defined input spectra, please see the examples provided in the `aXeSIM` Manual. Also a test package for the `aXeSIM` software is available from this Web page. The test package contains all necessary input data files (e.g. example input spectra, filter transmission curves[4]) to run the example simulations.

6 Troubleshooting

In this section we provide a, necessarily incomplete, list of commonly encountered problems and suggest possible solutions.

- **There is no spectrum of my favorite object.** Please check if the object is listed in the master object catalogue and was properly identified by `SExtractor`, including the information on size and major axis position angle.

- **The aXe software claims that a given file does not exist although I'm sure it's there in the right place.** When using *absolute* path names in deep directory structures the path name can get very long. If the maximum number of characters that `IRAF` allows for command input is exceeded then the above error occurs. Either use relative path names or move the directory structure such that the path names are shortened.

[3] `DER_SNR.py`

[4] The filter transmission curves were cut at 1% transmission, if a higher accuracy is needed then the full filter transmission curves should be re-retrieved from STScI.

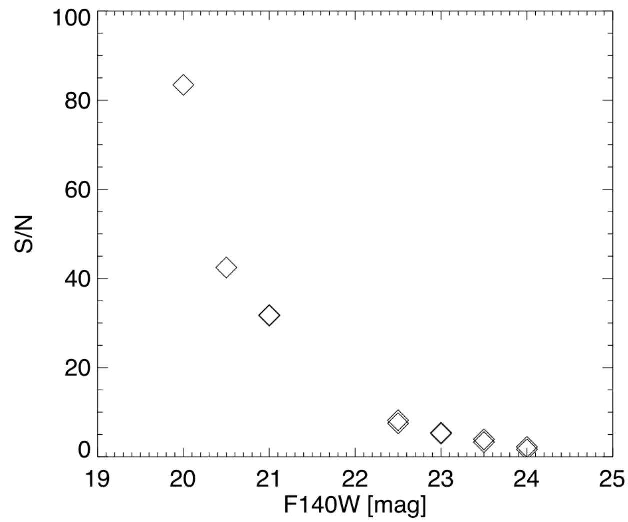

Figure 9: The Signal-to-Noise-Ratio of simulated point source observations as a function of the F140W magnitude (aXeSIM package).

- **Contamination estimates seem to be unrealistically high.** Please check if the flux columns in the master object catalogue (`MAG_F1392` in `F140W_prep.cat`) are set correctly and that pivot filter wavelength are given in units of [nm]. Note, that a wavelength must be specified and that listing only `MAG_AUTO` will lead to a error-exit of the software.
- **Do not run axeprep more than once on grism images.** Multiple runs will result in the header keyword for the background level to be set wrongly and subsequent reduction steps will fail.
- **Imperfect background subtraction.** aXe can also be used with a local background subtraction (see aXe Manual) which can help to achieve better results for individual objects.
- **Use of `_flt.fits` files.** The input data images for the aXe software are in `_flt.fits` format. Please do not use `_drz.fits` images since this will lead to wrongly applied distortion corrections.
- **Magnitude 99 for objects in Master Catalogue.** If SExtractor cannot estimate a magnitude for an object it assigns a magnitude of "99". This value of magnitude leads to an error in the aXe reduction software. Either replace the magnitude with a realistic number or remove the object from the Master catalogue.

7 Acknowledgements

For invaluable comments on early versions of the cookbook we would like to thank Kim Nilsson, Howard Bushouse and Nimish Hathi.

References

Bertin E., Arnouts S., 1996, A&AS, 117, 393

Freudling W., Kümmel M., Haase J., Hook R., Kuntschner H., Lombardi M., Micol A., Stoehr F., Walsh J., 2008, A&A, 490, 1165

Kümmel M., Walsh J. R., Pirzkal N., Kuntschner H., Pasquali A., 2009, PASP, 121, 59

Kuntschner H., Bushouse H., Kümmel M., Walsh J. R., 2009, Technical report, WFC3 SMOV proposal 11552: Calibration of the G141 grism (WFC3-2009-17)

Pasquali A., Pirzkal N., Larsen S., Walsh J. R., Kümmel M., 2006, PASP, 118, 270

Pirzkal N., Burgasser A. J., Malhotra S., Holwerda B. W., Sahu K. C., Rhoads J. E., Xu C., Bochanski J. J., Walsh J. R., Windhorst R. A., Hathi N. P., Cohen S. H., 2009, ApJ, 695, 1591

Pirzkal N., Pasquali A., Demleitner M., 2001, Space Telescope European Coordinating Facility Newsletter, 29, 5

Pirzkal N., Xu C., Malhotra S., et al., 2004, ApJS, 154, 501

Straughn A. N., Kuntschner H., Kuemmel M., et al., 2010, AJ in press (arXiv:1005.3071)

A Parameter set of main aXe tasks

In this Appendix we list the full, default parameter setup of all aXe tasks used in this cookbook.

A.1 iolprep parameters

```
--> lpar iolprep

 mdrizzle_ima = ""              Drizzled image
    input_cat = ""              Input catalogue
(dimension_in = "0,0,0,0")      Additional pixels on the left,right,bottom,top
    (useMdriz = yes)            Use multidrizzle header info for coord transform
                                (False for Astrodrizzle)
        (mode = "al")
```

A.2 axeprep parameters

```
--> lpar axeprep
       inlist = ""              List with input images
      configs = ""              Configuration filenames for CCD's
      (backgr = yes)            Subtract background image?
     (backims = "")             Background image(s)
     (backped = "")             Background podest mage
       (mfwhm = 2.0)            Multiplicative factor for mask image
        (norm = yes)            Normalize image by exposure time?
    (gaincorr = no)             Apply gain correction?
        (mode = "al")
```

A.3 aXecore parameters

```
--> lpar axecore
       inlist = ""              List with input images
      configs = ""              Configuration filenames for CCD's
     extrfwhm = 4.0             Multiplicative factor for object mask image
      drzfwhm = 3.0             Multiplicative factor for object mask image after axedrizzle
    (fconfigs = "")             Finge configuratione filenames for CCD's
        (back = no)             Create also the background PET

                                GOL2AF parameters:
    (backfwhm = 6.0)            Multiplicative factor for background mask image
      (orient = yes)            Use tilted extraction?
(slitless_geo = yes)            Use geometry optimized for slitless data?
     (exclude = no)             Remove faint objects from the results?
 (lambda_mark = 800.0)          Wavelength  to apply MMAG_EXTRACT and MMAG_MARK?

                                PETCONT parameters:
  (cont_model = "gauss")        Which contamination model?
 (model_scale = 3.0)            Scale factor for the model width?
```

```
  (inter_type = "linear")       Which interpolation type?
  (lambda_psf = 800.0)          Wavelength the psf was measured?

                                BACKEST parameters:
          (np = )               Number of points to use for computation
      (interp = )               Type of interpolation to perform
   (niter_med = )               Number of kappa-sigma klippings on median?
   (niter_fit = )               Number of kappa-sigma klippings on fit?
       (kappa = )               Kappa?
(smooth_lengt = )               Smoothing length?
 (smooth_fwhm = )               FWHM in background smoothing?

                                PET2SPC, STAMPS:
      (spectr = yes)            Extract spectra and stamp images?
    (adj_sens = no)             Using size-adjusted sensitivity curve?
     (weights = no)             Compute (and use) optimal weights?
    (sampling = "drizzle")      Which sampling type?
        (mode = "al")
```

A.4 drzprep parameters

```
--> lpar tdrzprep
       inlist = ""              Name of image list
      configs = ""              Configuration filenames for the CCD's
    (opt_extr = no)             Use optimal extraction in aXedrizzle?
        (back = no)             Generate ALSO a background DPP file using a PET?
        (mode = "al")
```

A.5 axedrizzle parameters

```
--> lpar axedrizzle
       inlist = ""              Name of image list
      configs = ""              Configuration filenames for the CCD's
       infwhm =                 Multiplicative factor in DPP's
      outfwhm =                 List of multiplicative factors for drizzled data
        (back = no)             Drizzling a background image?
(driz_separat = no)             Drizzling to separate image and CCR reject?
       (clean = yes)            Remove temporary files?

                                MEDIAN IMAGE parameters:
(combine_type = "median")       Type of combine operation?
(combine_mask = 0.7)            Percentage of weight image below which is is flagged as bad?
(combine_nsig = "4.0 3.0")      Significance for accepting minimum instead of median (for minmed)?
(combine_nlow = 0)              Number of low pixels to reject?
(combine_nhig = 0)              Number of high pixels to reject?
(combine_lthr = INDEF)          Lower threshold for clipping input pixel values?
(combine_hthr = INDEF)          Upper threshold for clipping input pixel values?
```

```
(combine_grow = 1.0)            Radius (pixels) for neighbor rejection?

                                BLOT BACK parameters:
 (blot_interp = "poly5")        Interpolant (nearest,linear,poly3,poly5,sinc)?
 (blot_sinscl = 1.0)            Scale for sinc interpolation kernel?

                                REMOVE COSMIC RAYS parameters:
 (driz_cr_snr = "5.0 4.0")      Driz_cr.SNR parameter?
(driz_cr_grow = 1.0)            Driz_cr_grow parameter
(driz_cr_scal = "1.2 0.7")      Driz_cr.scale parameter

                                SPECTRAL EXTRACTION parameters:
     (makespc = yes)            Making the final SPC and STP?
    (adj_sens = no)             Using size-adjusted sensitivity curve?
    (opt_extr = no)             Using optimal extraction?
        (mode = "al")
```

B Parameter set of main aXeSIM task

B.1 simdata parameters

```
--> lpar simdata
        incat = ""              Model object table name
       config = ""              aXe configuration filename
 (output_root = "")             Root for the output names
      (silent = yes)            Run in silent mode

                                PREPSPECTRA parameters:
 (inlist_spec = "")             List of template spectra (fits or ascii)
  (tpass_flux = "")             Total passband for flux scaling

                                PREPIMAGES parameter:
  (inlist_ima = "")             List of template images (fits)

                                SIMDISPIM parameters:
  (lambda_psf = )               Wavelength the object shape was determined at [nm]
     (nx_disp = )               Dimension of simulated dispersed image in x (if non-default)
     (ny_disp = )               Dimension of simulated dispersed image in y (if non-default)
(exptime_disp = )               Exposure time for simulated dispersed image
(bck_flux_dis = "0.0")          Backgrounds flux/image for dispersed image in [e/s]
  (extraction = yes)            Do a dummy extraction on the dispersed image

                                SIMDIRIM parameters:
(tpass_direct = "")             Name of the total passband file (fits or ascii)
      (nx_dir = )               Dimension of simulated direct image in x (if non-default)
      (ny_dir = )               Dimension of simulated direct image in y (if non-default)
 (exptime_dir = )               Exposure time for simulated direct image
```

```
(bck_flux_dir = "0.0")           Backgrounds flux/image for direct image in [e/s]
     (version = "1.21 (2009Apr01)") aXeSIM version number
        (mode = "al")
```