SOXS Pipeline

SOXS Consortium Science Meeting. Naples.
June 24-27th, 2024.

Dave Young (QUB), Marco Landoni (INAF) & Stephen Smartt (Oxford)

Overview

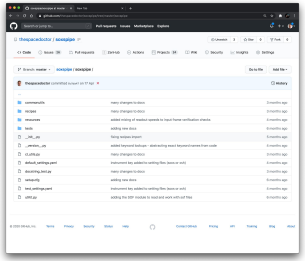
- SOXS Pipeline: the key facts
- High-level science requirements
- What science products to expect
- SOXS automated data-reduction and dataflow
- Quality Control and instrument health monitoring
- Next steps

soxspipe: key facts

soxspipe ...

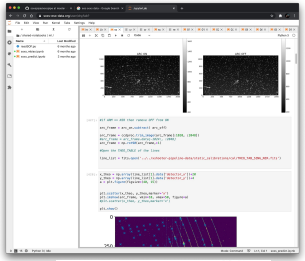
- is an open-source python package

Development Environment & Infrastructure



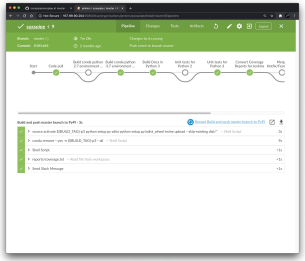
GitHub

Version Control via git and Github as a 'centralised' remote repository (also for project planning and issue tracking).



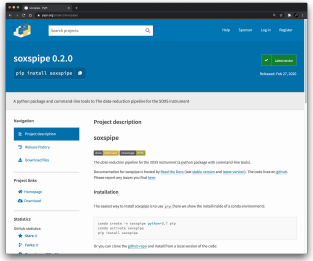
jupyter

Jupyter notebooks for development, investigation, prototyping, visualisation …



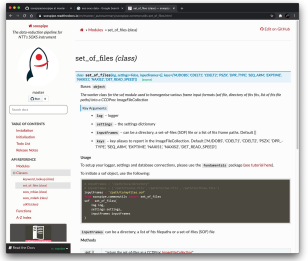
Jenkins

Code testing and continuous integrate via Jenkins server and declarative pipeline.



python Package Index

PyPi and conda for production code distribution.



SPHINX

Read the Docs

Docs written in docstrings and markdown files that live beside the code.

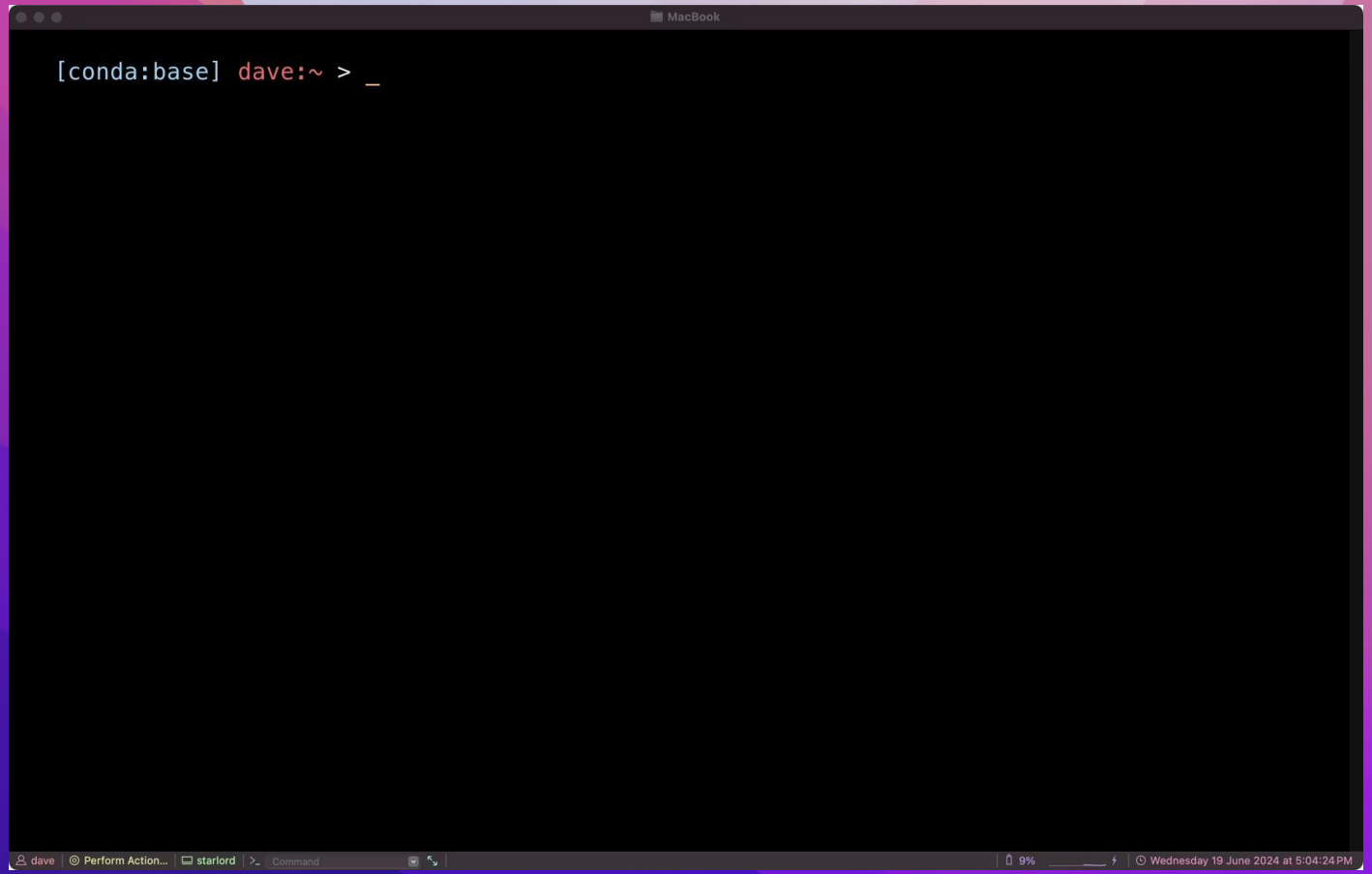
https://github.com/thespacedoctor/soxspipe
https://soxspipe.readthedocs.io/

soxspipe: key facts

soxspipe ...

- is a python package
- installs with one command in < 60 sec*

* assuming miniconda or anaconda is already installed


MacBook
[conda:base] dave:~ > _
dave
Perform Action...
starlord
Command
9%
Wednesday 19 June 2024 at 5:04:24 PM

soxspipe: key facts

soxspipe ...

- is a python package
- installs with one command in < 60 sec
- is ‘driven’ from the command-line

Usage

```
[conda:soxspipe] dave:~ > soxspipe -h
Documentation for soxspipe can be found here: http://soxspipe.readthedocs.org

Usage:
    soxspipe prep <workspaceDirectory>
    soxspipe [-q] reduce all <workspaceDirectory> [-s <pathToSettingsFile>]
    soxspipe session ((ls|new|<sessionId>)|new <sessionId>)
    soxspipe [-Vx] mdark <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile>]
    soxspipe [-Vx] mbias <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile>]
    soxspipe [-Vx] disp_sol <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile> --poly=<od>]
    soxspipe [-Vx] order_centres <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile> --poly=<ooww>]
    soxspipe [-Vx] mflat <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile>]
    soxspipe [-Vx] spat_sol <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile> --poly=<oowwss>]
    soxspipe [-Vx] stare <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile>]
    soxspipe [-Vx] nod <inputFrames> [-o <outputDirectory> -s <pathToSettingsFile>]

Options:
    prep                                prepare a folder of raw data (workspace) for data reduction
    session ls                          list all available data-reduction sessions in the workspace
    session new [<sessionId>]           start a new data-reduction session, optionally give a name up to 16 characters A-Z, a-z, 0-9 and/or _-
    session <sessionId>                 use an existing data-reduction session (use `session ls` to see all IDs)
    reduce all                          reduce all of the data in a workspace.

    mbias                               the master bias recipe
    mdark                               the master dark recipe
    mflat                               the master flat recipe
    disp_sol                            the disp solution recipe
    order_centres                       the order centres recipe
    spat_sol                            the spatial solution recipe
    stare                               reduce stare mode science frames
    nod                                 reduce nodding mode science frames

    inputFrames                         path to a directory of frames or a set-of-files file

    -q, --quitOnFail                    stop the pipeline if a recipe fails
    -h, --help                          show this help message
    -v, --version                       show version
    -s, --settings <pathToSettingsFile>    the settings file
    -V, --verbose                       more verbose output
    -x, --overwrite                     more verbose output
    --poly=<ORDERS>                     polynomial degrees (overrides parameters found in setting file). oowwss = order_x,order_y,wavelength_x,w
avelength_y,slit_x,slit_y e.g. 345435. od = order,dispersion-axis
[conda:soxspipe] dave:~ > _
```

recipes

sof files

soxspipe: key facts

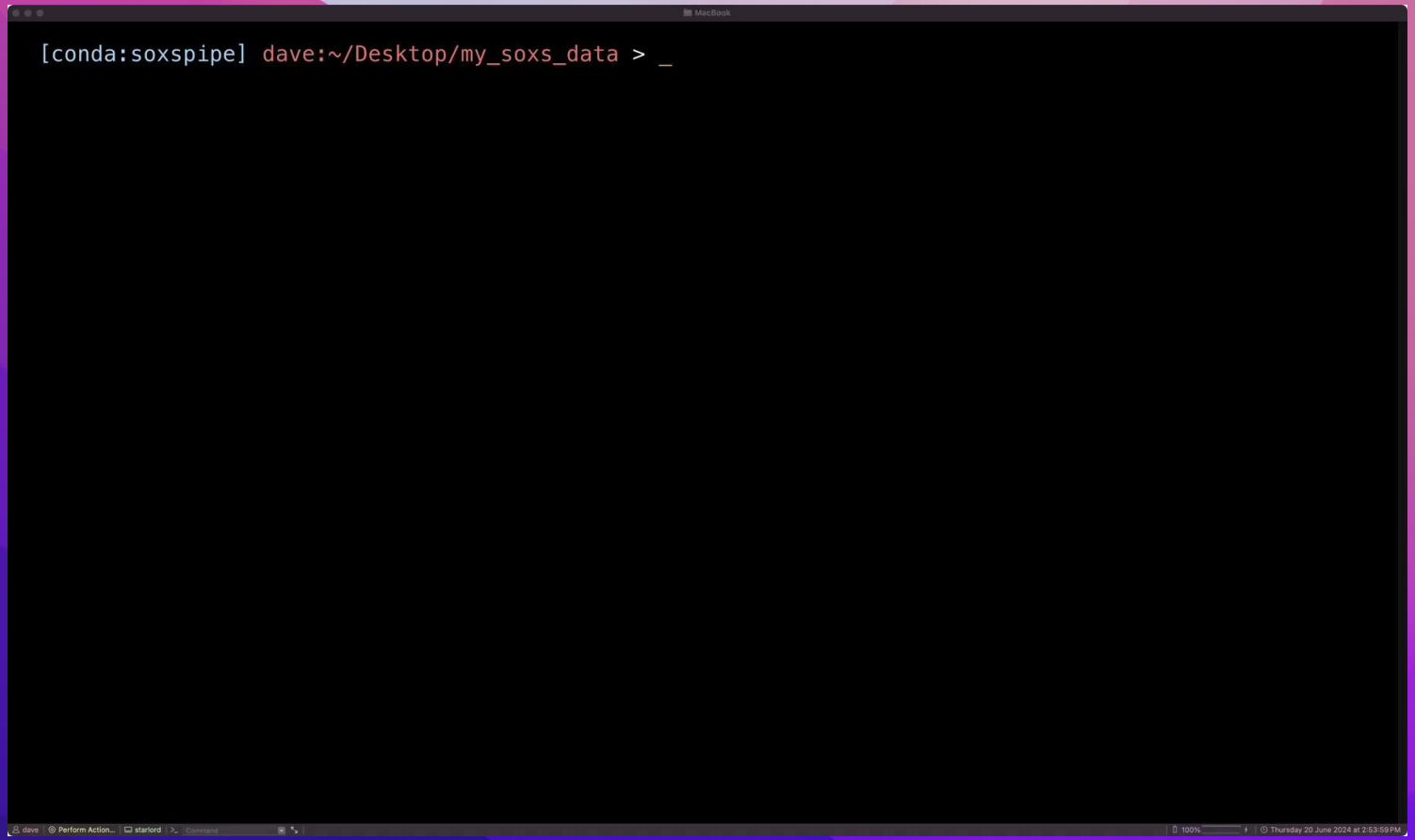
soxspipe ...

- is a python package
- installs with one command in < 60 sec
- is ‘driven’ from the command-line
- uses the ESO pipeline concepts of recipes and sets-of-files
- is very simple to use 'out-of-the-box'

A typical data reduction checklist

1. Download your dataset from ESO SAF into a single folder.
2. Open the terminal and change directory into the folder.
3. Activate the soxspipe conda environment with `conda activate soxspipe`
4. Run `soxspipe prep .`
5. Run `soxspipe reduce all .`

a typical reduction



soxspipe: key facts

soxspipe ...

- is a python package
- installs with one command in < 60 sec
- is ‘driven’ from the command-line
- uses the ESO pipeline concepts of recipes and sets-of-files
- is very simple to use 'out-of-the-box'
- is highly tunable with settings in a single yaml file

soxspipe.yaml

/Users/Dave/Dropbox/Mac/Desktop/soxspipe_sandbox/xsh_5_eso_097_feedback_objects/sessions/base/soxspipe.yaml

soxspipe.yaml

```yaml
tune-pipeline: False

soxs-mbias:
    # INDIVIDUAL RAW BIAS FRAME CLIPPING PARAMETERS. DATA IS CLIPPED BEFORE STACKING.
    frame-clipping-sigma: 3
    frame-clipping-iterations: 1
    # STACKED FRAME CLIPPING PARAMETERS. DATA IS CLIPPED AFTER STACKING.
    stacked-clipping-sigma: 5
    stacked-clipping-iterations: 3

soxs-mdark:
    # INDIVIDUAL RAW DARK FRAME CLIPPING PARAMETERS. DATA IS CLIPPED BEFORE STACKING.
    frame-clipping-sigma: 3
    frame-clipping-iterations: 1
    stacked-clipping-sigma: 5
    stacked-clipping-iterations: 5
    clipping-lower-sigma: 3
    clipping-upper-sigma: 3
    clipping-iteration-count: 5

soxs-disp-solution:
    uvb:
        # SIZE OF STAMP USED TO IDENTIFY ARC-LINES IN PINHOLE IMAGE
        pixel-window-size: 10
        # MINIMUM SIGNIFICANCE REQUIRED FOR ARC-LINE TO BE CONSIDERED 'DETECTED'
        pinhole-detection-thres-sigma: 3
        # DEGREE OF ORDER TERM WHEN FITTING DISPERSION SOLUTION POLYNOMIALS [X,Y]
        order-deg: [4,4]
        # DEGREE OF WAVELENGTH TERM WHEN FITTING DISPERSION SOLUTION POLYNOMIALS [X,Y]
        wavelength-deg: [4,5]
        # CLIPPING LIMIT (MEDIAN AND MAD) WHEN FITTING GLOBAL POLYNOMIAL TO DISPERSION SOLUTION
        poly-fitting-residual-clipping-sigma: 5
        # MAXIMUM NUMBER OF CLIPPING ITERATIONS WHEN FITTING GLOBAL POLYNOMIAL TO DISPERSION SOLUTION
        poly-clipping-iteration-limit: 7
```

LSP-cspell, INSERT MODE, Line 6, Column 50 Spaces: 4 YAML

SOXS Pipeline: key facts

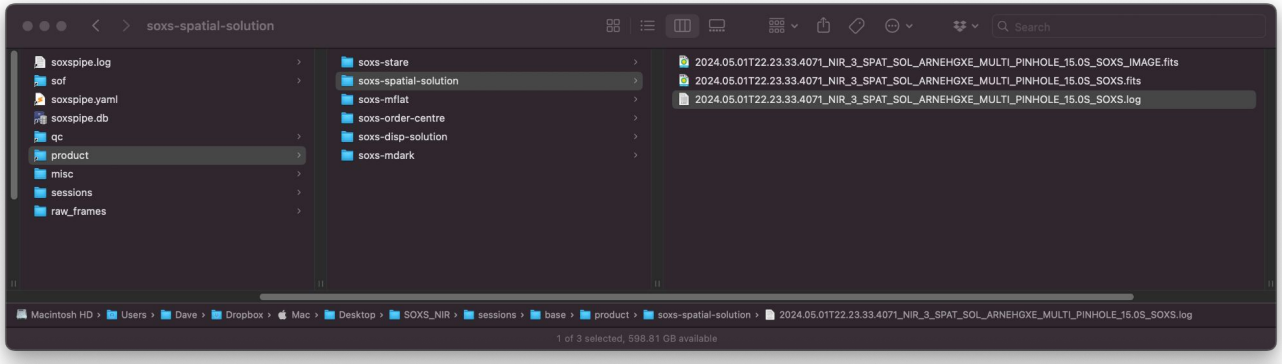
`soxspipe` ...

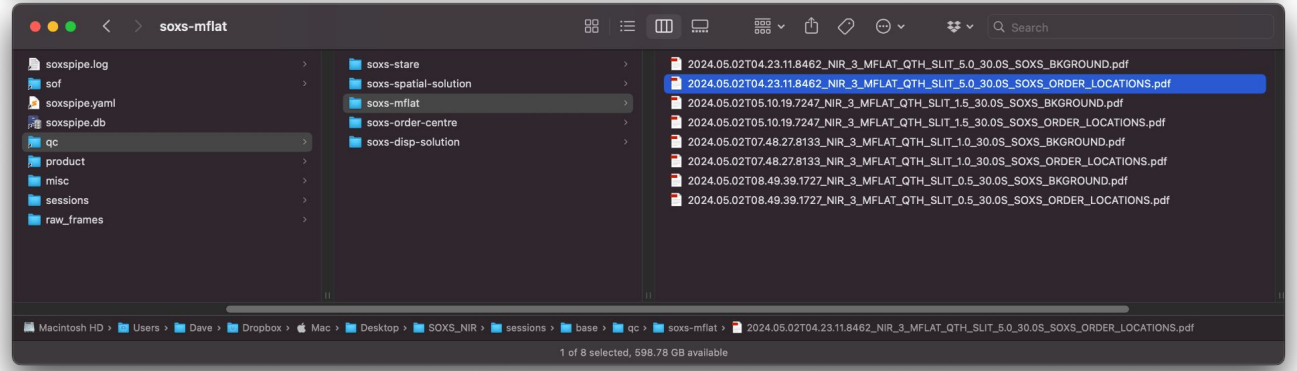
- uses ‘sessions’ to reduce a single dataset multiple times with different settings
- has a built-in intelligent data-organiser
- reduces both SOXS and Xshooter data
- employs logging throughout (to stdout and to file)
- generates many useful QC metrics and plots alongside data products

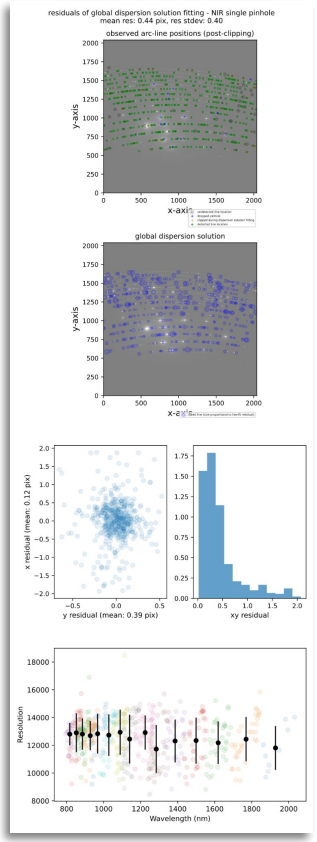
recipe products and qc metrics

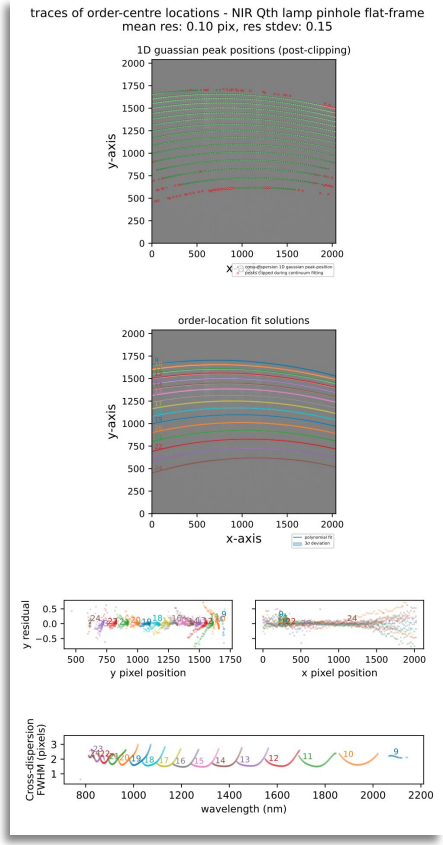
```
# SOXS-DISP-SOLUTION QC METRICS
+-----------+-----------+-----------+-------------------------------------------------------------------------+
|   qc_name |   qc_value |   qc_unit |                                                              qc_comment |
|-----------+-----------+-----------+-------------------------------------------------------------------------|
|     CLINE |  103       |     lines |       Total number of detected lines clipped during solution fitting |
|     NLINE |  628       |     lines |                        Number of lines detected in single pinhole frame |
|     PLINE |    0.95586 |           | Proportion of input line-list lines detected on single pinhole frame |
|     TLINE |  657       |     lines |                              Total number of line in single line-list |
+-----------+-----------+-----------+-------------------------------------------------------------------------+

# SOXS-DISP-SOLUTION RECIPE PRODUCTS & QC OUTPUTS
+---------------------+-----------------------------------------------------------------------------------------+-------------+---------------------+---------+-----------------------------------+
|       product_label |                                                                               file_name |   file_type |        obs_date_utc |   label |                      product_desc |
|---------------------+-----------------------------------------------------------------------------------------+-------------+---------------------+---------+-----------------------------------|
|            DISP_MAP |            2024.05.01T21.45.37.4048_NIR_3_DISP_SOL_ARNEHGXE_PINHOLE_15.0S_SOXS.fits | FITS Table | 2024-05-01T21:45:37.4048 |    PROD |  NIR first pass dispersion solution |
|     DISP_MAP_LINES |  2024.05.01T21.45.37.4048_NIR_3_DISP_SOL_ARNEHGXE_PINHOLE_15.0S_SOXS_FITTED_LINES.fits |        FITS | 2024-05-01T21:45:37.4048 |      QC | NIR dispersion solution fitted lines |
| DISP_MAP_LINES_MISSING |  2024.05.01T21.45.37.4048_NIR_3_DISP_SOL_ARNEHGXE_PINHOLE_15.0S_SOXS_MISSED_LINES.fits |        FITS | 2024-05-01T21:45:37.4048 |      QC |          NIR undetected arc lines |
|        DISP_MAP_RES | 2024.05.01T21.45.37.4048_NIR_3_DISP_SOL_ARNEHGXE_PINHOLE_15.0S_SOXS_RESIDUALS_534500.pdf |         PDF | 2024-05-01T21:45:37.4048 |      QC |    NIR dispersion solution QC plots |
+---------------------+-----------------------------------------------------------------------------------------+-------------+---------------------+---------+-----------------------------------+

Recipe Command: soxspipe disp_sol sof/2024.05.01T21.45.37.4048_NIR_3_DISP_SOL_ARNEHGXE_PINHOLE_15.0S_SOXS.sof -s ./sessions/base/soxspipe.yaml
Recipe Run Time: 9s
```

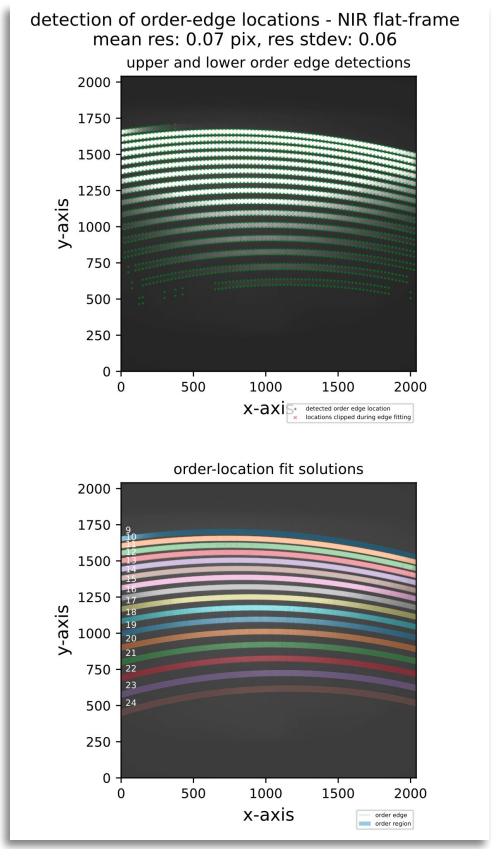
Logging and QC Plots

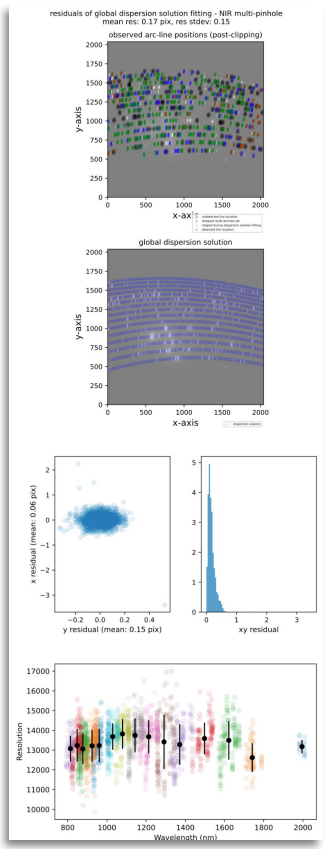





residuals of global dispersion solution fitting - NIR single pinhole
mean res: 0.44 pix, res stdev: 0.40
observed arc-line positions (post-clipping)
global dispersion solution
y-axis
x-axis
x residual (mean: 0.12 pix)
y residual (mean: 0.39 pix)
xy residual
Resolution
Wavelength (nm)


traces of order-centre locations - NIR Qth lamp pinhole flat-frame
mean res: 0.10 pix, res stdev: 0.15
1D guassian peak positions (post-clipping)
order-location fit solutions
y-axis
x-axis
y residual
y pixel position
x pixel position
Cross-dispersion FWHM (pixels)
wavelength (nm)


detection of order-edge locations - NIR flat-frame
mean res: 0.07 pix, res stdev: 0.06
upper and lower order edge detections
order-location fit solutions
y-axis
x-axis


residuals of global dispersion solution fitting - NIR multi-pinhole
mean res: 0.17 pix, res stdev: 0.15
observed arc-line positions (post-clipping)
global dispersion solution
y-axis
x-axis
x residual (mean: 0.06 pix)
y residual (mean: 0.15 pix)
xy residual
Resolution
Wavelength (nm)

Pipeline Science Requirements

1. The SOXS pipeline shall run on a machine in La Silla on all SOXS frames taken, reducing a data set within 10 minutes of the OB completion (with a goal of 5 minutes). It shall run automatically on all point source targets above an AB magnitude of $r = 19$ (with a goal of reaching $r = 20$). Below this magnitude, the pipeline should run automatically but may require user interaction to optimise the object extraction.

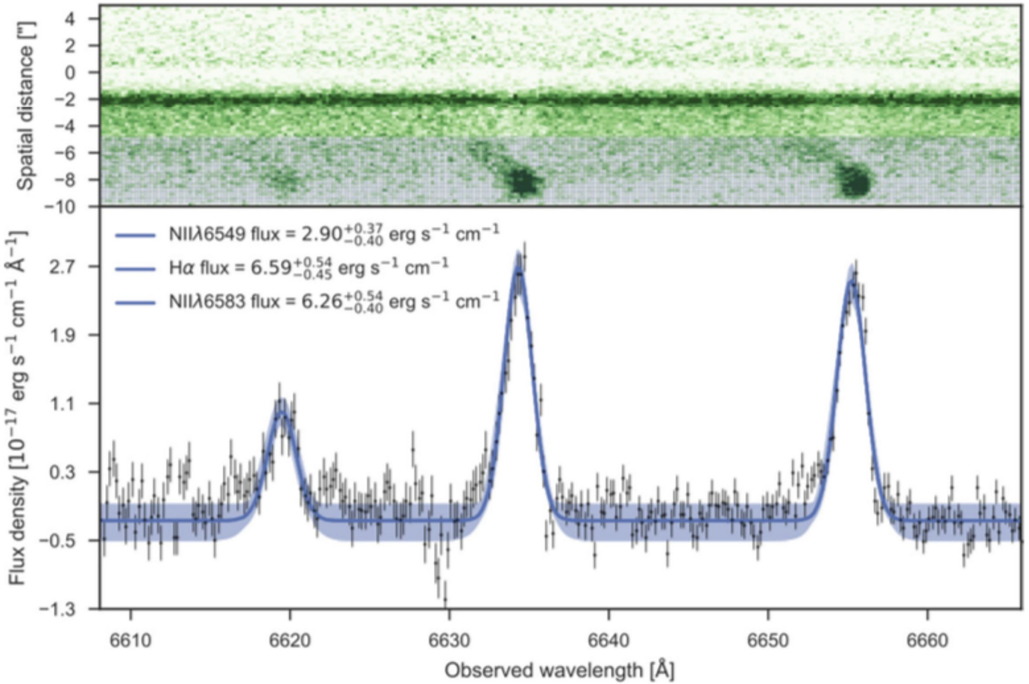
2. The pipeline shall produce science-ready data products from this automated pipeline

3. The acquisition camera data shall be detrended (bias, flat-field and bad pixel masking) and automatically calibrated (astrometrically and photometrically). The *griz* filter images shall be astrometrically and photometrically calibrated using *Refcat2* (Tonry et al., 2018), and the *u*- and *y*-bands will be calibrated when an all-sky catalogue of *u*- and *y*-band photometry becomes available. The SOXS team will define a strategy for objects with complex backgrounds and for which automated photometry may be challenging.

4. Science-ready spectral data products are defined as:

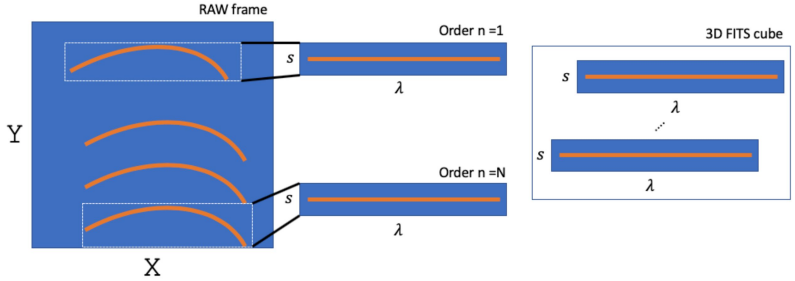
Product	Description
1D Source Spectra	1D spectra in FITS binary table format, one for each arm. Each FITS spectrum file will contain 4 extensions: 1. Wavelength- and flux-calibrated spectra with absolute flux correction via scaling to acquisition image source photometry, 2. an additional spectrum with correction for telluric absorption via MOLECFIT, 3. the variance array and 4. the sky-background spectra.
1D Merged Source Spectrum	1D UV-VIS & NIR merged spectrum in FITS binary table format with PDF visualisation. This spectrum will be rebinned to a common pixel scale foreach arm. This spectrum file will also have the same 4 extensions described above.
2D Source Spectra	A 2D FITS image for each spectral arm containing wavelength and flux calibrated spectra (no other corrections applied) allowing users to perform source extraction with their tool of choice. Note that rectification of the curved orders in the NIR introduces a source of correlated noise not present in extractions performed on the un-straightened orders as done by the pipeline.
Acquisition Camera Images	*ugrizy* astrometrically and photometrically (*griz* only) calibrated to Refcat2 (Tonry et al. 2018).

RESEARCH LETTER



AT2017gfo (GW170817)
XShooter, Pian et al. 2017

Produce 2D distortion corrected, orders merged pre-extraction spectrum for each arm (rectification)



Pipeline Science Requirements

5. Quality control data products shall be defined as:

   (a) Individual bias frames for read-out-noise calculation.

   (b) Master bias frames for checking bias level variations.

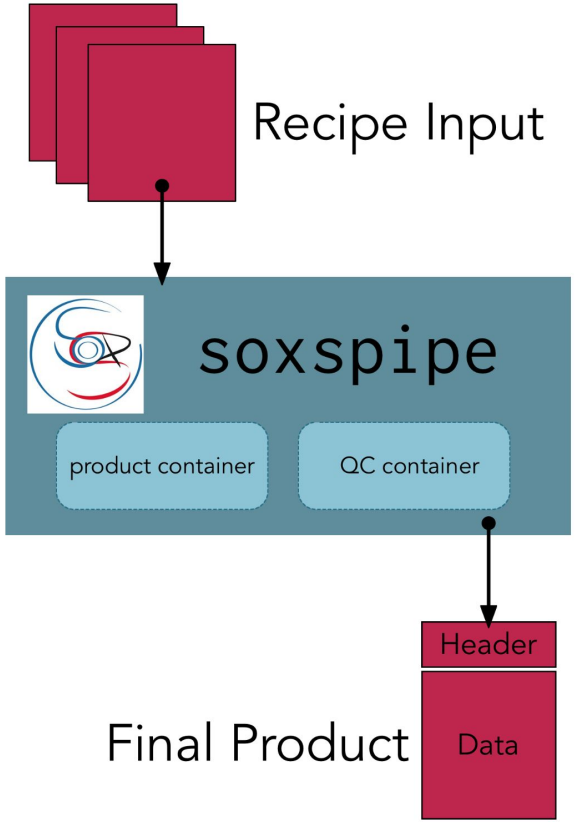
   (c) Master dark frames for bad pixel maps.

   (d) Master, rectified 2D flats fields to allow cross-check of suspected spectral artefacts.

   (e) PDF plots of the spectrum and QC parameters.

   (f) An archive of sensitivity curves shall be maintained and used to validate each new sensitivity curve.

Pipeline QC Reporting



Architecture of pipeline recipes now includes a ‘product’ container and a ‘qc’ container.

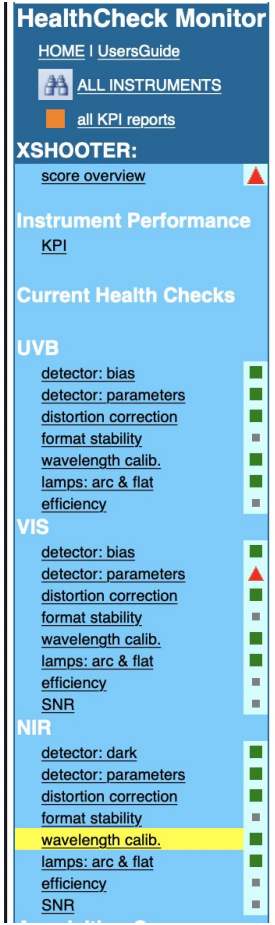
These containers persist throughout the lifecycle of the recipe and can be updated/added to at any point.

Contents of containers are used to create the final product when a recipe is complete, with QCs being written to the FITS header(s).

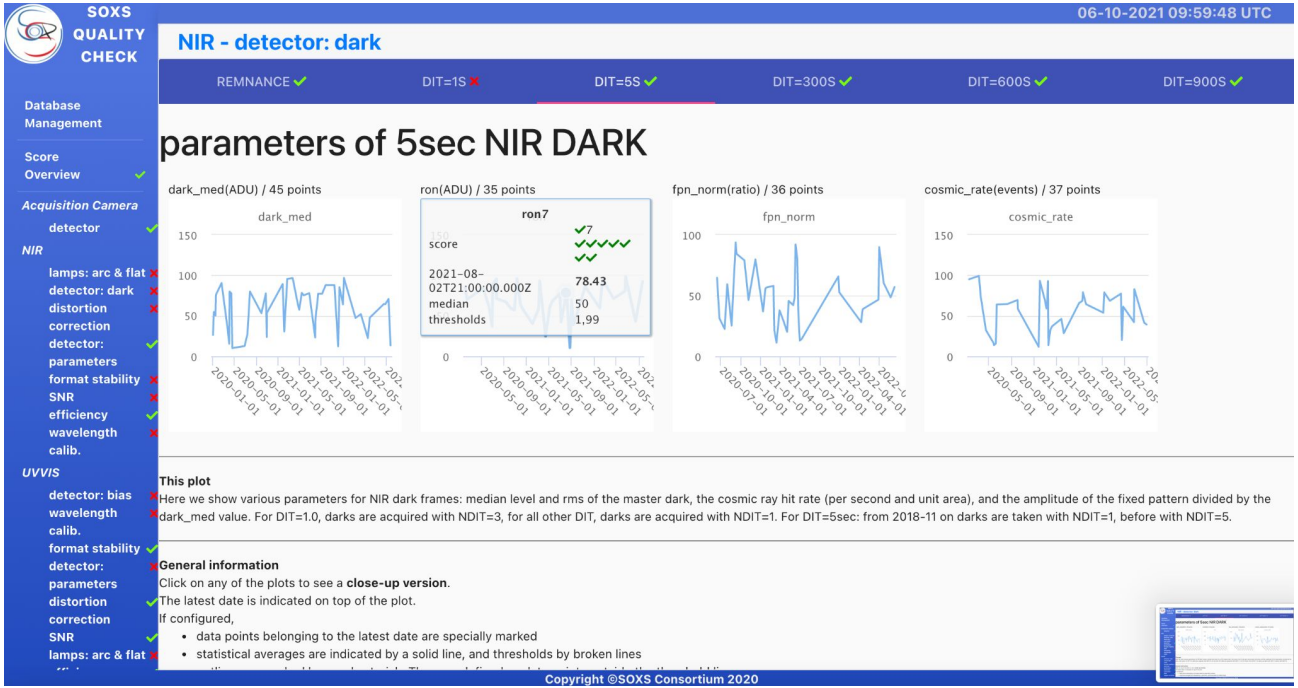
Pipeline QC Reporting

9 Groups of QCs (following structure of XShooter Health Check Monitor)

1. Detector: Bias QCs
2. Detector: Dark QCs
3. Detector: Parameters QCs
4. Distortion Correction QCs
5. Lamps: Arc & flats QCs
6. Wavelength Calibration QCs
7. Format Stability QCs
8. Efficiency QCs
9. SNR QCs



SOXS Instrument Health Monitoring Webpage



Pipeline Science Requirements

5. Quality control data products shall be defined as:

   (a) Individual bias frames for read-out-noise calculation.

   (b) Master bias frames for checking bias level variations.

   (c) Master dark frames for bad pixel maps.

   (d) Master, rectified 2D flats fields to allow cross-check of suspected spectral artefacts.

   (e) PDF plots of the spectrum and QC parameters.

   (f) An archive of sensitivity curves shall be maintained and used to validate each new sensitivity curve.

6. The pipeline shall be designed such that an ESO science user with their own NTT time outside the GTO can install and run the pipeline to reduce their specific data. This may be either from the original raw frames or starting from the 2D frames.

7. The pipeline shall have an optimal extraction routine.

Pipeline Science Requirements

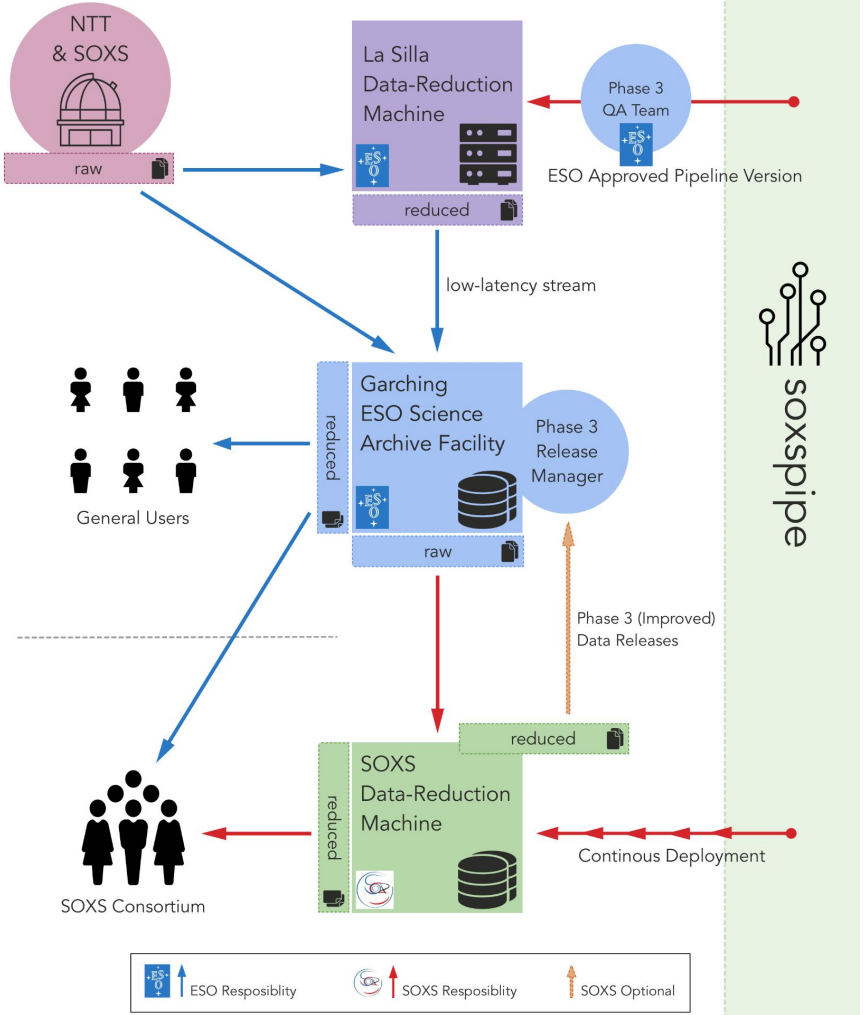
8. A cosmic-ray rejection algorithm shall be run on the UV-VIS images before extraction occurs.

9. The wavelength calibration shall be achieved by using pinhole arc-line exposures from either afternoon or early twilight. The pipeline will check the wavelength calibration with the skyline positions and apply a correction (e.g. for wavelength drift or flexure) as required.

10. The pipeline shall reduce data taken in stare, nodding and offset modes.

11. The pipeline should allow a user to carry out their own wavelength calibration - for example, if a user takes arc frames before and after a scientific exposure then the pipeline should allow the user to use these to interpolate a solution at the time of the science exposure. This will not be provided automatically as a science product but the pipeline should be designed such that a user can run it in this mode.

SOXS automated data-reduction and dataflow


NTT
& SOXS
raw
La Silla
Data-Reduction
Machine
reduced
Phase 3
QA Team
ESO Approved Pipeline Version
low-latency stream
Garching
ESO Science
Archive Facility
reduced
raw
Phase 3
Release
Manager
General Users
Phase 3 (Improved)
Data Releases
reduced
SOXS
Data-Reduction
Machine
reduced
SOXS Consortium
Continous Deployment
soxspipe
ESO Resposiblity
SOXS Resposiblity
SOXS Optional

Next Steps

- Continue to prioritise helping the Padova team during integrations and PAE
- Flux calibration. Code is completed, but robustness testing needed before release.
- Image rectification.
- Phase 3 compliance.
- Ironing out the logistics of automated data-reduce and data-flow into ESO SAF.
- ACQ camera image reduction
- Bug finding and fixing (call for beta-testers)

A soxspipe Recipe API

class **_base_recipe_**(*log, settings=False*) [source]

Bases: `object`

The base recipe class which all other recipes inherit

Key Arguments:

- `log` – logger
- `settings` – the settings dictionary

Usage

To use this base recipe to create a new `soxspipe` recipe, have a look at the code for one of the simpler receipes (e.g. `soxs_mbias`) - copy and modify the code.

Methods

`clean_up` ()	*remove intermediate files once recipe is complete*
`prepare_frames` ([save])	*prepare all frames in the input data*
`prepare_single_frame` (frame[, save])	*prepare a single raw frame by converting to electron counts and adding mask and uncertainty extensions*

A soxspipe Recipe API

```python
from ._base_recipe_ import _base_recipe_

class soxs_mbias(_base_recipe_):

    def __init__(
            self,
            log,
            settings=False,
            inputFrames=[]
    ):

        ...
```

class **soxs_mbias**(*log, settings=False, inputFrames=[]*) [source]

Bases: `soxspipe.recipes._base_recipe_._base_recipe_`

The soxs_mbias recipe

Key Arguments

- `log` – logger
- `settings` – the settings dictionary
- `inputFrames` – input fits frames. Can be a directory, a set-of-files (SOF) file or a list of fits frame paths. Default []

Methods

`clean_up` ()	*remove intermediate files once recipe is complete*
`prepare_frames` ([save])	*prepare all frames in the input data*
`prepare_single_frame` (frame[, save])	*prepare a single raw frame by converting to electron counts and adding mask and uncertainty extensions*
`produce_product` ()	*The code to generate the product of the soxs_mbias recipe*
`verify_input_frames` ()	*verify the input frame match those required by the soxs_mbias recipe*

https://www.eso.org/sci/software/pipelines/installation/software_prerequisites.html

ESO Pipelines and EsoReflex Software Prerequisites for Source-based Installations

Home | About | RPM Installation | MacPorts Installation | Source Kits | Support

EsoReflex 2.9.0 and newer supports Python based recipes as long as EsoRex is compiled with one of the following prerequisites:

SOXS

SOXS

Son Of X-Shooter

NTT Nasmyth Focus, La Silla, Chile

Science Operations to begin Mid-2022

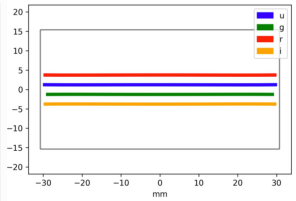
UV-VIS Arm

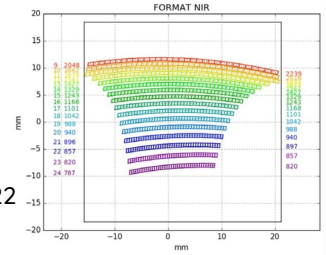
NIR Arm

A&G Camera

Parameter	Value
Detector	e2V CCD44–82
Pixel-Size	15 μm
Array-Size	2048 × 4096 px; 30.7 x 61.4 mm
Array-Scale	0.28 arcsec/px
Peak Signal	200,000 e^-/px
Gain	Slow: 0.6 ± 0.1 e^-/ADU Fast: 2 ± 0.2 e^-/ADU
Read noise (rms)	Slow: <3 e^- Fast: <8 e^-
Dark current @ 153K	< 0.00001 e^-/s/px
Resolution (R)	3500–7000 (≃ 4500 mean)
Wavelength Range	350–850nm
Slit Widths	0.5, 1.0, 1.5, 5.0 arcsec
Slit Height	11 arcsec
Grating Blaze Angle	41°
Orders (quasi)	4

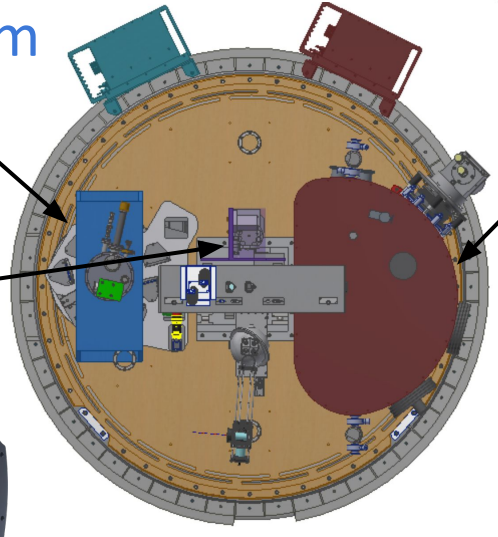
UV-VIS Spectrograph/CCD Characteristics





Parameter	Value
Detector	Teledyne H2RG
Pixel-Size	18 μm
Array-Size	2048 × 2048 px
Array-Scale	0.25 arcsec/px
Read noise (rms)	Double correlated: < 20 e^- 16 Fowler pairs < 7 e^-
Dark current @ 40K	< 0.005 e^-/s/px
Resolution (R)	≃ 5000 (1 arcsec slit)
Wavelength Range	800–2000 nm
Slit Widths	0.5, 1.0, 1.5, 5.0 arcsec
Slit Height	11 arcsec
Grating Blaze Angle	44°
Detector Operating Temp	40K
Spectrograph Operating Temp	150K
Orders	15

NIR Spectrograph/Array Characteristics



A face-on view of SOXS on the NTT rotator flange.
Figure 2 of Schipani, P. et al. (2018)

Parameter	Value
Camera	Andor iKon M934
Detector	BEX2-DD
Pixel-Size	13 μm
Array-Size	1024 × 1024; 13.3 x 13.3 mm
Array-Scale	0.205 arcsec/px
Peak Signal	130000 e^-/px
Dark Current @ 173 K	0.00012 e^-/s/px
Read noise (rms)	2.9 e^-
Filters	*u, g, r, i, z, y, V*

Acquisition Camera Characteristics

- medium resolution spectrograph (R≃4500) capable of simultaneously observing 350–2000nm (U- to H-band).
- limiting magnitude of R~20 (3600sec, S/N~10).
- primary science objective to study the transient sky; classifying and following transients discovered by all-sky imaging surveys (PanSTARRS, ATLAS, ZTF, LSST).
- Will respond to rapid and long-term Target of Opportunity (ToO) requests.
- SOXS consortium will be allocated 900 NTT nights over 5 years (50% time).
- ESO community can apply for the remaining time.