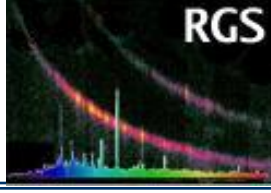


RGS data reduction and analysis of point-like sources

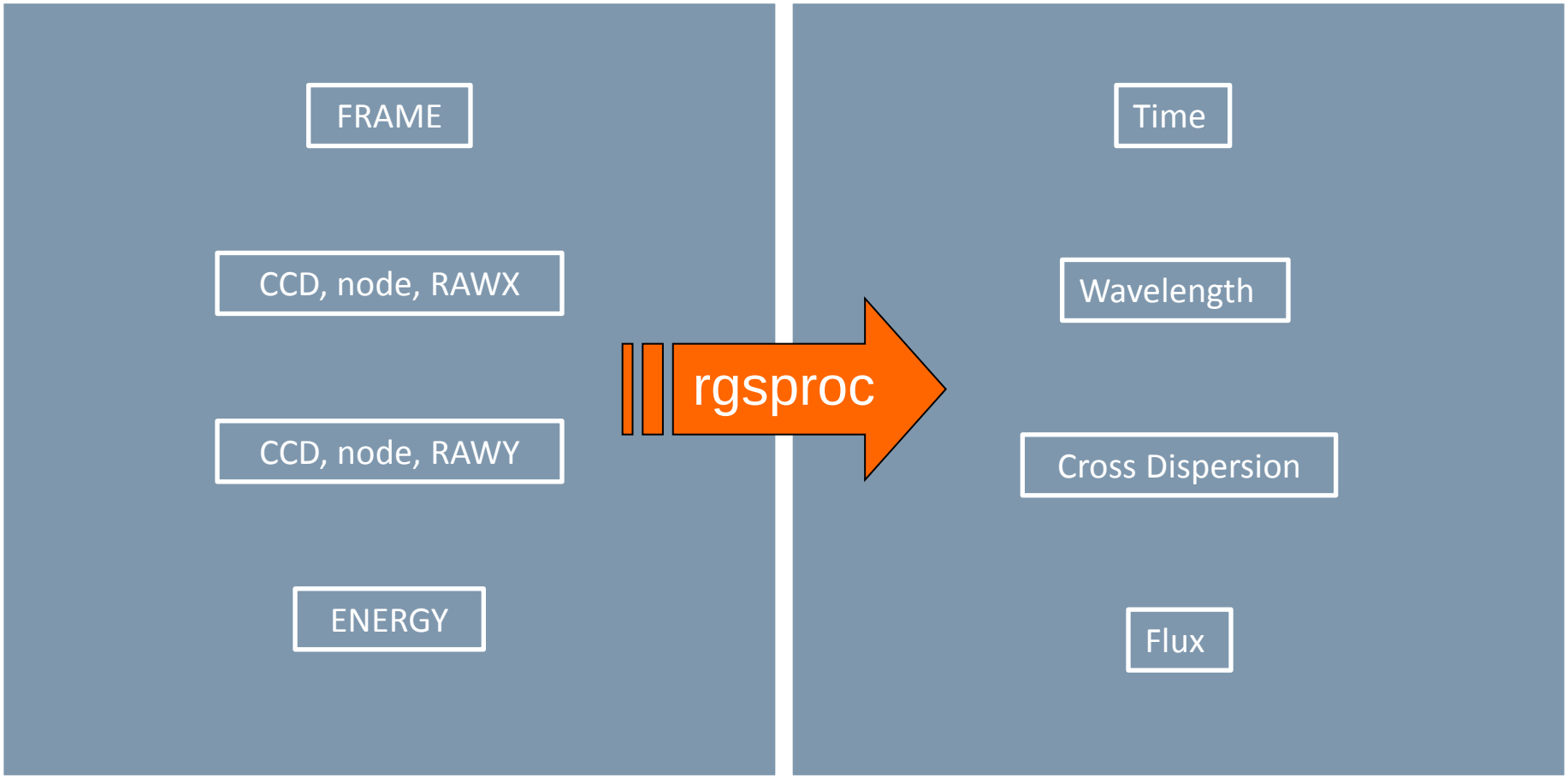
Rosario González-Riestra

XMM-Newton SOC
ESAC

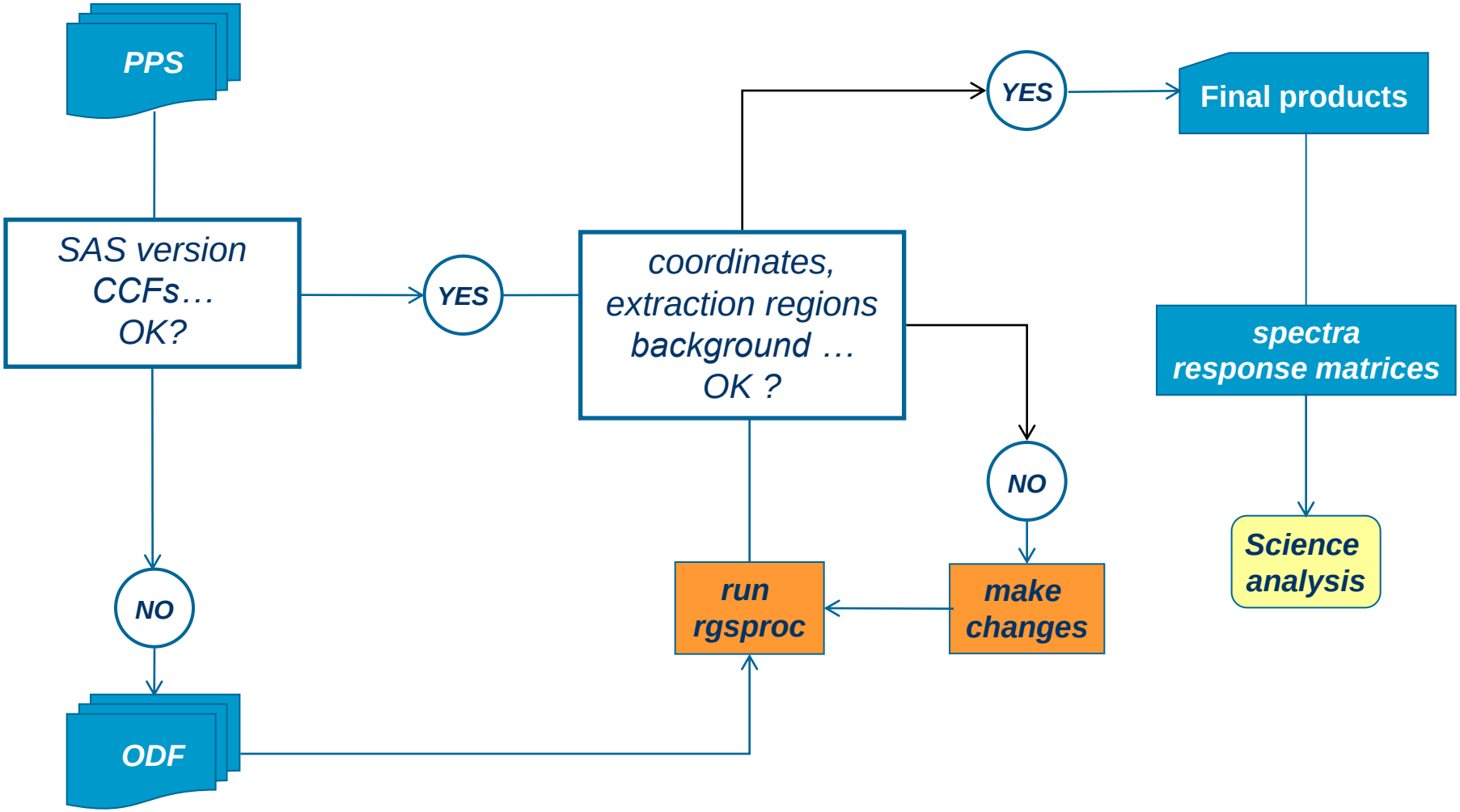
Processing RGS data (I)

from....

to...



Processing RGS data (II)



The RGS processing task: rgsproc

meta-task: interface to ≈ 20 SAS tasks (that can also be run separately)

controlled by ≈ 80 parameter switches

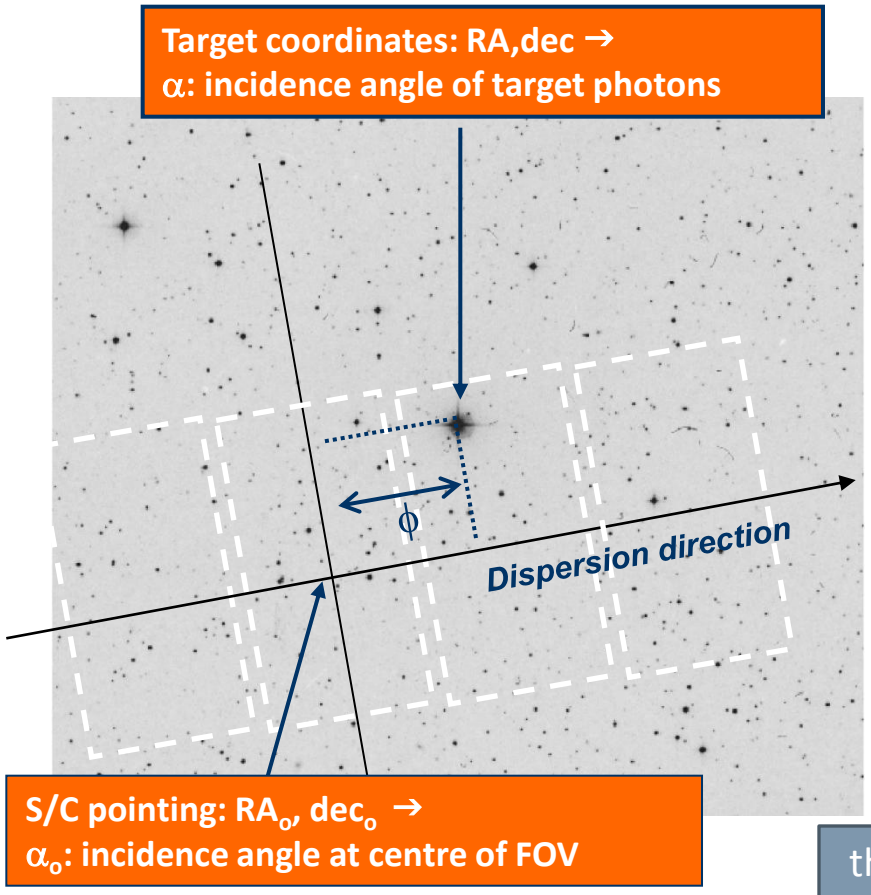
six entry and final points ("*processing stages*")

1:events	*2:angles*
3:filter	*4:spectra*
5:fluxing	*6:lightcurve*

produces filtered event lists, light curves, spectra and response matrices

the quality of the results depends critically on the accuracy of the coordinates of the source

A simplified scheme of the RGS FOV (and why the coordinates are so important!)



$$\lambda = (\cos \beta - \cos \alpha)\ \ d / m$$

measured *known*

$$\alpha = \alpha_o + \phi\ F / L$$

known *known*

$$\phi = f\ (RA, dec, RA_o, dec_o, P.A.)$$

input to rgsproc *known from S/C attitude history file*

the wavelength scale and the effective area depend on the position of the source in the FOV

1 arcsec ≈ 2.3 mÅ (45 km/s at 15 Å)

rgsproc: what does it do?

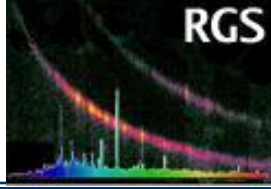
Stage	Task	Purpose	Output
Events	atthkgen	generates attitude file	**Source list and intermediate combined event list**
	attfilter	filters the attitude file	
	hkgtigen	generates housekeeping GTIs	
	rgsoffsetcalc	uses the diagnostic mode data for offset calculation	
	rgssources	creates the list of sources to processed	
	rgsframes	flags bad frames, convert RAW[XY] to readout node reference system , creates GTI for telemetry drops, calculates dead time	
	rgsenergy	performs energy calibrations, i.e. creates the PI column	
	rgsbadpix	flags bad pixels from CCF and from own analysis	
	rgsevents	reconstructs events: total energy ,pattern and coordinates	
	evlistcomb	concatenates event list	

Source independent

Angles	**rgsangles**	performs aspect correction	**Final event list**
Filter	**rgsfilter**	filters event list, removing unwanted frames and events and adding exposure maps	
Spectra	**rgsregions**	computes background and source extraction regions for each source	**Source and background spectra**
	rgsspectrum	extracts source and background spectra	
	rgsbkgmodel	generates model background	
Fluxing	**rgsrmfgen**	creates response matrix	**Response matrices and fluxed spectrum**
	rgsfluxer	combines several RGS spectra into a single "fluxed" spectrum	
Lightcurve	**rgslccorr**	creates background-subtracted light curves	**Light curve**

Source dependent

rgsproc: parameters



entrystage -- optional,type=string,default=events; (re)start processing at

finalstage -- optional,type=string,default=fluxing; conclude processing at

withinstexpids -- optional,type=bool,d

instexpids -- optional,type=list<string>

orders -- optional,type=list<int>,const

expunge -- optional,type=list<string>,c

withprefix -- optional,type=bool,defau

prefix -- optional,type=string; prefix fo

timestep -- optional,type=real,constra

driftlimit -- optional,type=real,constra

withepicset -- optional,type=bool,defa

epicset -- optional,type=dataset; name

withsrc -- optional,type=bool,default=

srclabel -- optional,type=string,default

srcrate -- optional,type=real,constrain

srcstyle -- optional,type=string,default

srcra -- optional,type=real,default=-99

srcdec -- optional,type=real,default=-9

srcdisp -- optional,type=real,default=0

srcxdsp -- optional,type=real,default=0

attstyle -- optional,type=string,default

attra -- optional,type=real,default=-99

attdec -- optional,type=real,default=-9

attapos -- optional,type=real,default=-

calcoffsets -- optional,type=bool,defau

withoffsethistogram -- optional,type=

withdiagoffset -- optional,type=bool,d

withgain -- optional,type=bool,default

withcti -- optional,type=bool,default=

withadvisory -- optional,type=bool,de

withfoundhot -- optional,type=bool,default=yes; include found hot pixels

pixnoiselimit -- option

colnoiselimit -- option

pixsharpness -- option

colsharpness -- option

detcoord -- optional,t

withdetcoord -- optio

reconstruct -- optiona

betabinning -- optiona

betabinref -- optional,

betabinwidth -- optio

nbetabins -- optional,

betamin -- optional,ty

betamax -- optional,ty

xdispbinning -- option

xdispbinref -- optiona

xdispbinwidth -- optio

nxdispbins -- optional

xdispmin -- optional,t

xdispmax -- optional,t

withpointingcolumn --

withmlambdacolumn

rejflags -- optional,typ

driftbinsize -- optiona

withcombmap -- optio

auxgtitables -- optiona

spatialres -- optional,type=int,constraints="spatialres in [2:]",default=5; vertices per node across spatial regions

orderres -- optional,type=int,constraints="orderres in [2:]",default=3; vertices per node across order regions

xpsfincl -- optional,type=real,constraints="xpsfincl in [0:100]",default=90; percent of cross-dispersion PSF included

xpsfexcl -- optional,type=real,constraints="xpsfexcl in [0:100]",default=95; percent of cross-dispersion PSF excluded from background

pdistincl -- optional,type=real,constraints="pdistincl in [0:100]",default=90; percent of pulse-height distribution included

procsrcsexpr -- optional,type=string,default=INDEX==#PRIMESRC; select which sources to process

exclsrcsexpr -- optional,type=string,default=INDEX==#PRIMESRC; select which sources to exclude from background

bkgcorrect -- optional,type=bool,default=yes; background correct the spectra

withbkgset -- optional,type=bool,default=yes; generate background spectra

edgechannels -- optional,type=int,constraints="edgechannels in [0:]",default=2; flag chip-edge channels as dubious

rebin -- optional,type=int,constraints="rebin in [1:]",default=1; dispersion channel rebinning factor

withfracexp -- optional,type=bool,default=no; add FRAC_EXP column to output files

exposed -- optional,type=real,constraints="exposed in [0:1]",default=0.1; minimum good fractional exposure

badquality -- optional,type=int,constraints="badquality in [0:5]",default=1; QUALITY used to flag bad channels

rmfbins -- optional,type=int,constraints="rmfbins in [1:]",default=4000; number of response energy bins

fftdim -- optional,type=int,constraints="fftdim in [1:5]",default=3; convolution size exponent

withmirrorpsf -- optional,type=bool,default=yes; whether to include the standard mirror PSF distribution

angdistset -- optional,type=file,default=angdist.txt; user-defined angular distribution file

withangdist -- optional,type=bool,default=no; whether to include a custom angular distribution

mergeorders -- optional,type=bool,default=no; flux all orders together

flxformat -- optional,type=string,default=dal; output file format

flxmode -- optional,type=string,default=wavelength; computational mode

flxmin -- optional,type=real,constraints="flxmin in [0:]",default=4; minimum output wavelength or energy

flxmax -- optional,type=real,constraints="flxmax in [0:]",default=39; maximum output wavelength or energy

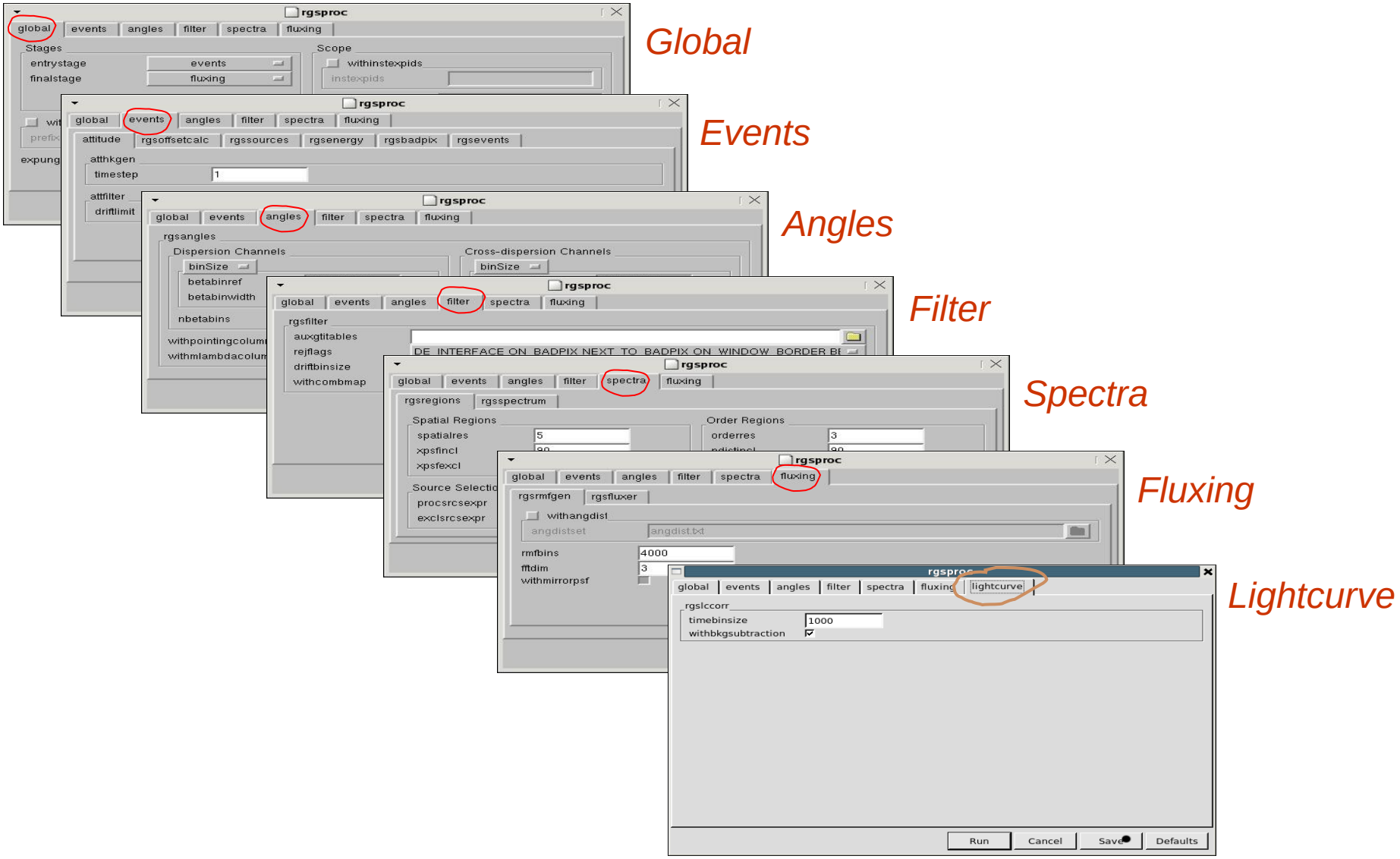
flxbins -- optional,type=int,constraints="flxbins in [1:]",default=3400; number of output bins

flxquality -- optional,type=list<int>,constraints="flxquality in [0:5]",default=0; usable quality in promotion order

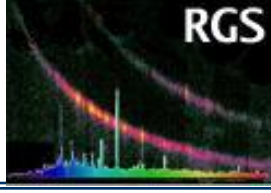
withflxnan -- optional,type=bool,default=no; enable alternate NaN value

flxnan -- optional,type=real,default=0; alternate NaN value

rgsproc: the GUI



What do I get after the processing?



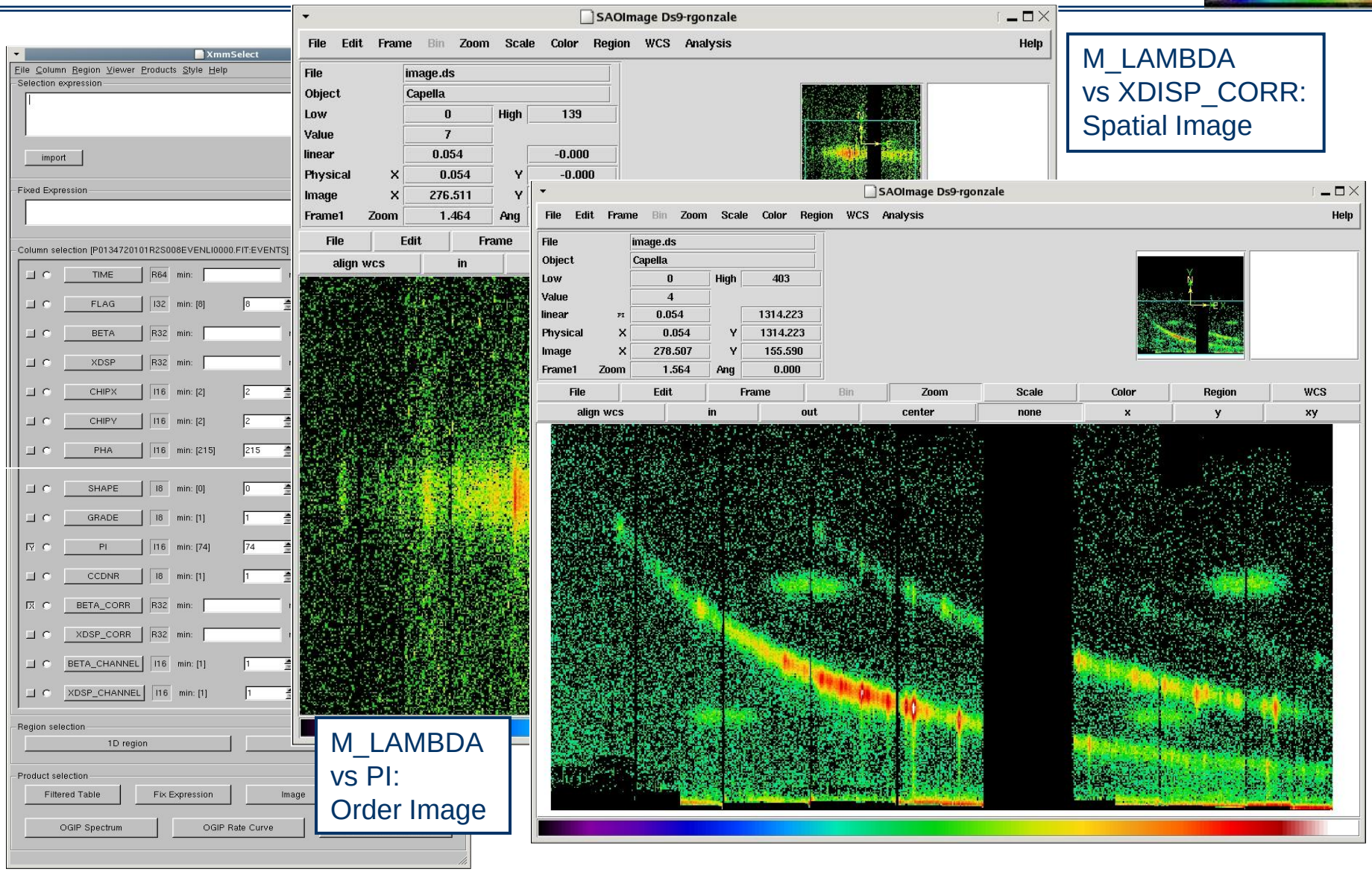
For each RGS and exposure:

File	Content	rgsproc (default)	PPS
P0123456701R1S004EVENLI0000.FIT	Filtered Event List	Y	Y
P0123456701R1S004SRCLI_0000.FIT	Source List	Y (coord from proposal)	Y (coord from EPIC src list)
P0123456701R1S004BGSPEC1001.FIT P0123456701R1S004BGSPEC2001.FIT	Background Spectra (1st and 2nd order)	Y	Y
P0123456701R1S004SRSPEC1001.FIT P0123456701R1S004SRSPEC2001.FIT	Source Spectra (1st and 2nd order)	N	Y
P0123456701R1S004SBSPEC1001.FIT P0123456701R1S004SBSPEC2001.FIT	Source+Bkg Spectra (1st and 2nd order)	Y	Y
P0123456701R1S004RSPMAT1001.FIT P0123456701R1S004RSPMAT2001.FIT	Response Matrices (1st and 2nd order)	Y	only for 1st order
P0123456701R1S004IMAGE_0000.FIT P0123456701R1S004IMAGE_0000.PNG	Dispersion-CrossDispersion Image	N	Y
P0123456701R1S004ORDIMG0000.FIT P0123456701R1S004ORDIMG0000.PNG	Dispersion-Energy Image	N	Y
P0123456701R1S004EXPMAP0000.FIT	Exposure Map	Y	Y
P0123456701R1S004FBKTSR0000.FIT	Flare Background Timeseries	N	Y
P0123456701R1S004MBSPEC1000.FIT P0123456701R1S004MBSPEC2000.FIT	Model Background Spectra	N	Y
P0123456701R1S004SRTSR_1001.FIT P0123456701R1S004BGTSR_1001.FIT	Source and Background Lightcurves	Y	Y

For each observation:

File	Content	rgsproc (default)	PPS
P0123456701OBX000fluxed1000.FIT P0123456701OBX000fluxed2000.FIT	Source Fluxed Spectra (1st and 2nd order)	Y	Y

The filtered event list



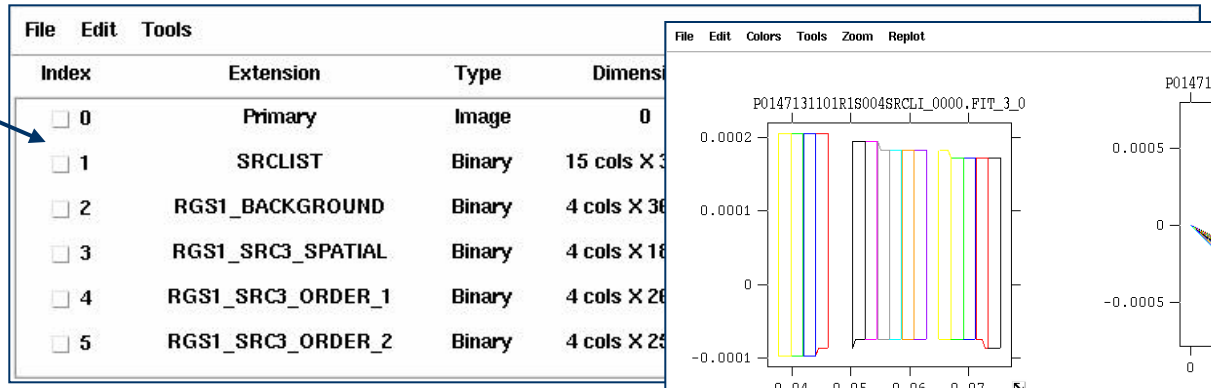
M_LAMBDA vs XDISP_CORR: Spatial Image

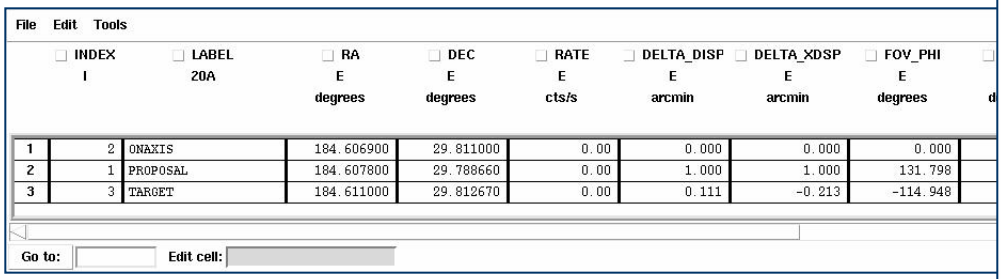
M_LAMBDA vs PI: Order Image

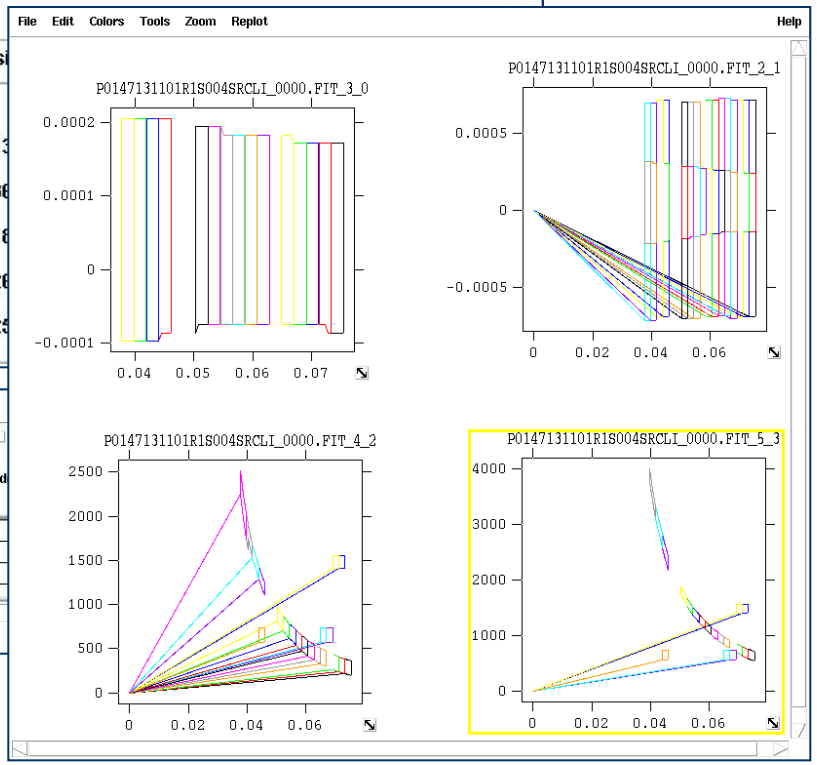
The source list

List of coordinates

Extraction regions





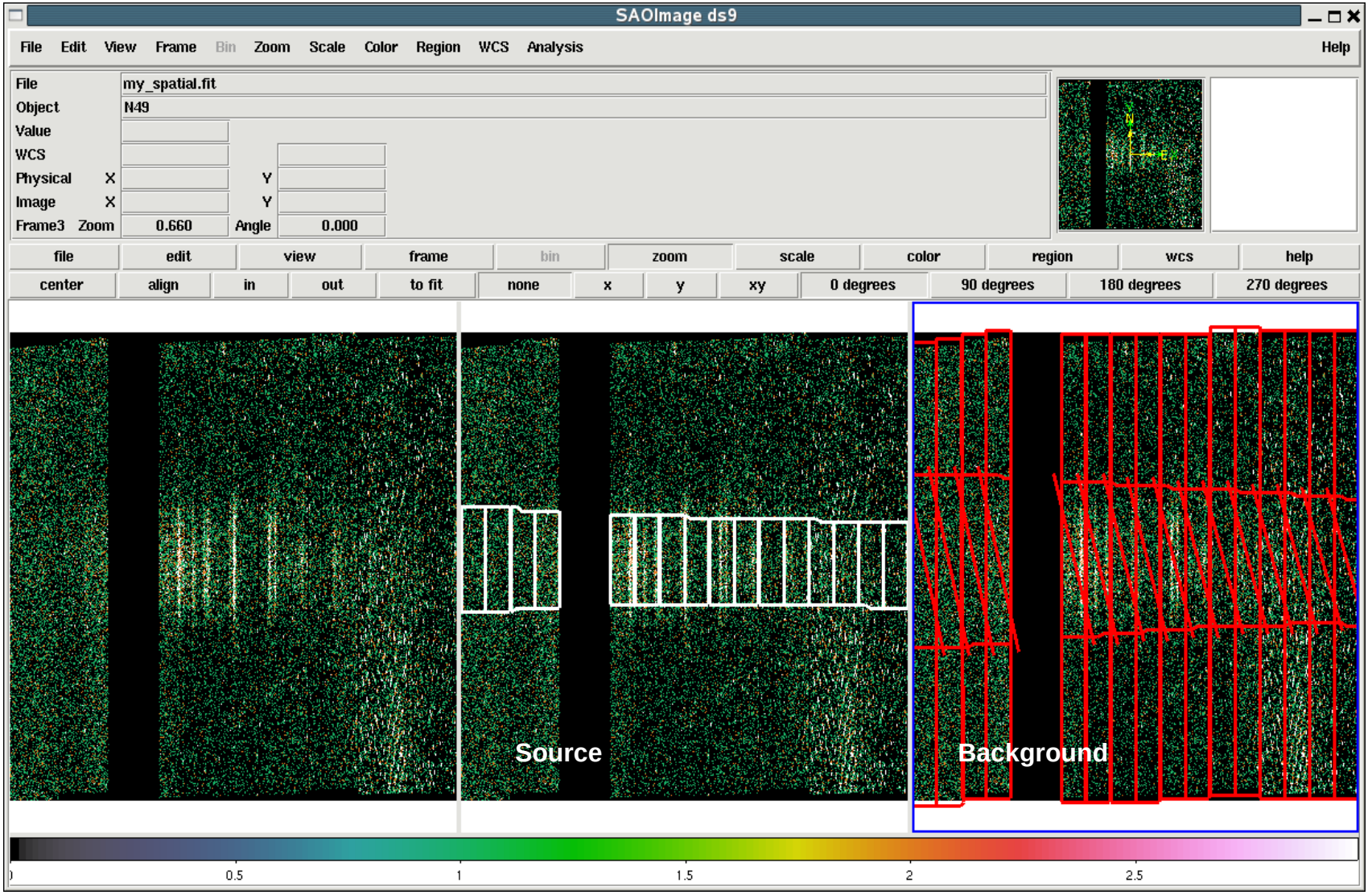


Sources can be added and/or modified with **`rgssources`**

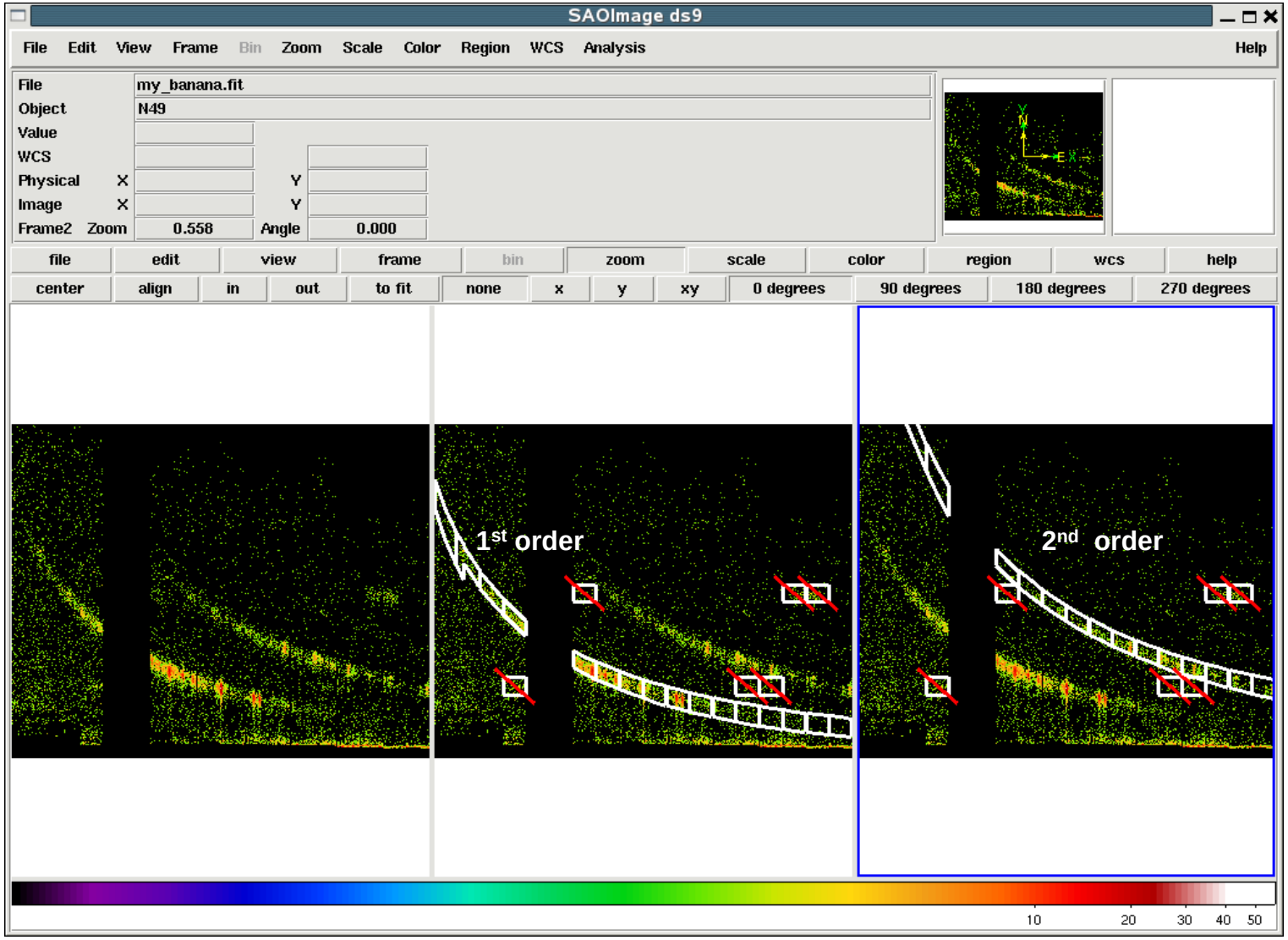
Region definitions can be changed with **`rgsregions`**. Defaults are:

- source spatial: 95% of x-dispersion PSF inclusion
- background spatial: 98% of x-dispersion PSF exclusion
- order mask: 95% of pulse-height distribution

The event list and the extraction regions



The event list and the extraction regions



The light curve

generated with **rgslccorr**

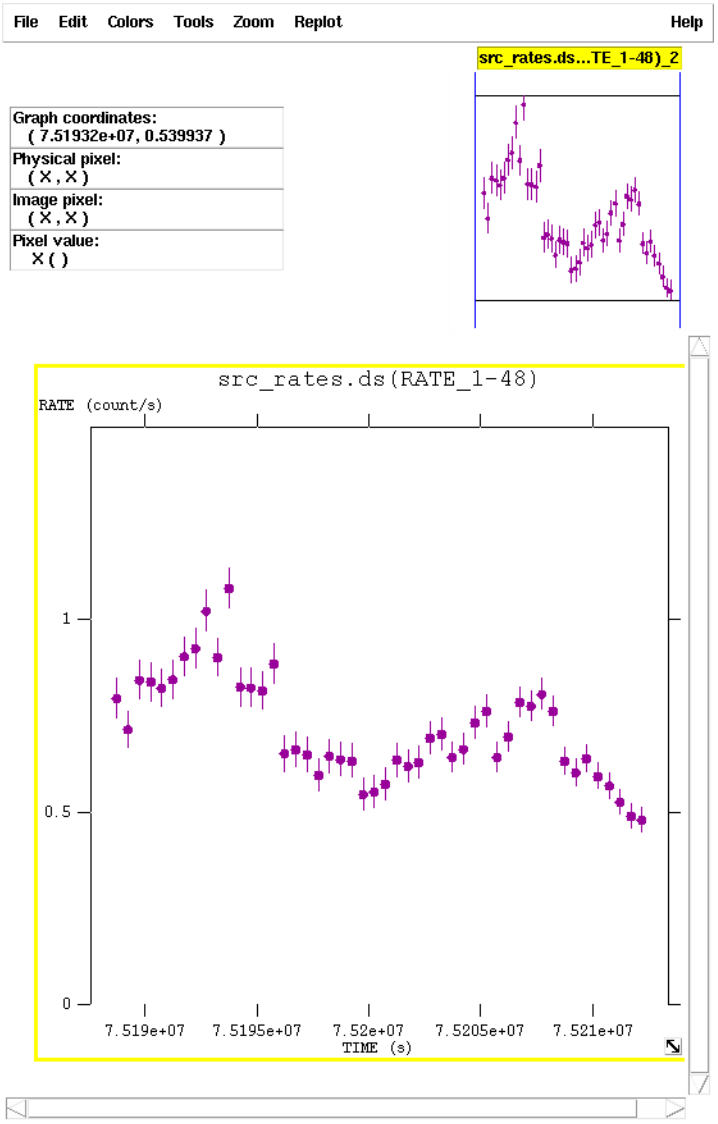
filters the event list file using the source and background region from the source list

light curves are extracted for 1st and 2nd orders

default in **rgsproc**: with background subtraction and time bin of 1000 s

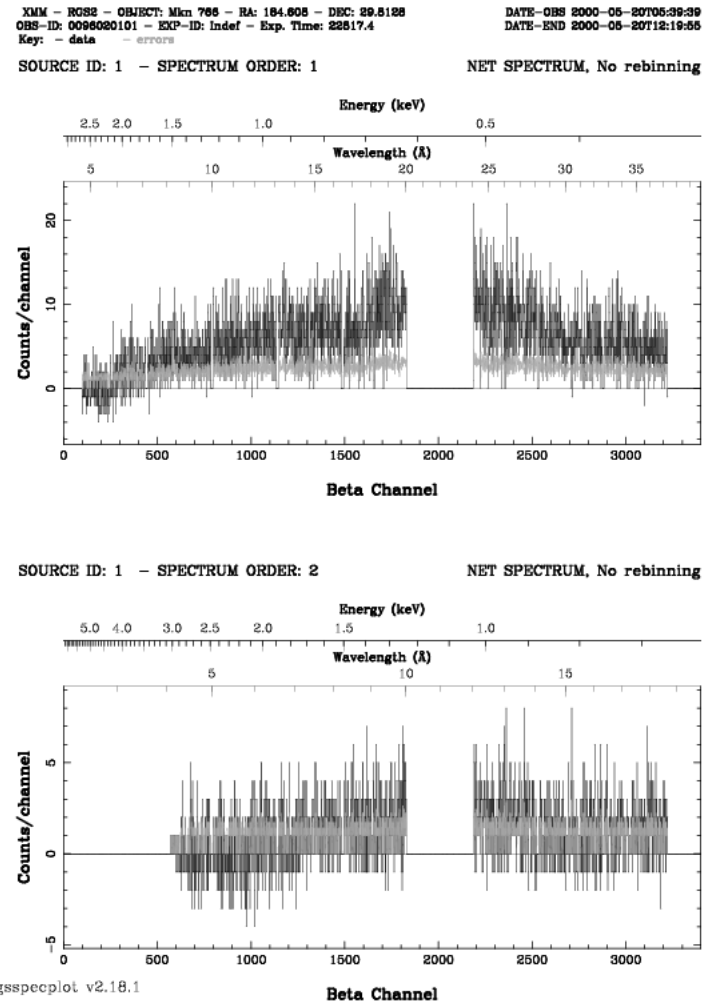
rgslccorr allows the user to select a range of CCDs/wavelength/energy

the background light curve is produced separately



The extracted spectrum

- generated with **rgsspectrum**
- spectra are extracted for 1st and 2nd orders
- default since SAS 10.0: wavelength space
- the **rgsproc** default is the total source spectrum (i.e. NOT background subtracted)
- the background spectrum is produced separately
- a model background can be generated optionally with **rgsbkgmodel**



The fluxed spectrum

RGS extracted spectra (channel/counts) can be converted to physical units

Å vs. photons cm^{-2} s^{-1} Å $^{-1}$

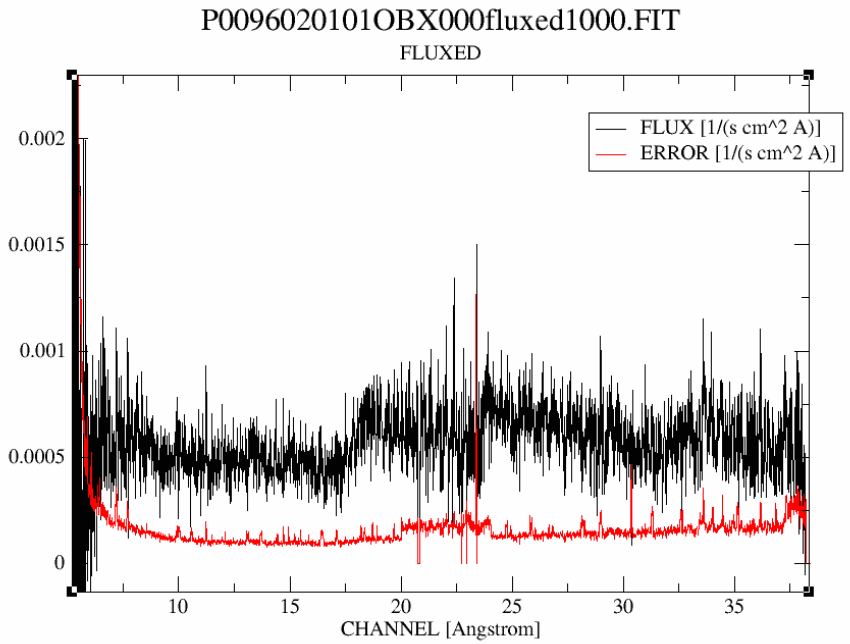
(*fluxed*) with **`rgsfluxer`**

This task also merges several spectra to (e.g.) on a single spectrum increasing the signal-to-noise ratio

fluxed spectra must be used with care in spectral fitting codes, since they do not take into account the effects of redistribution.

from RGS countrate to physical flux =>

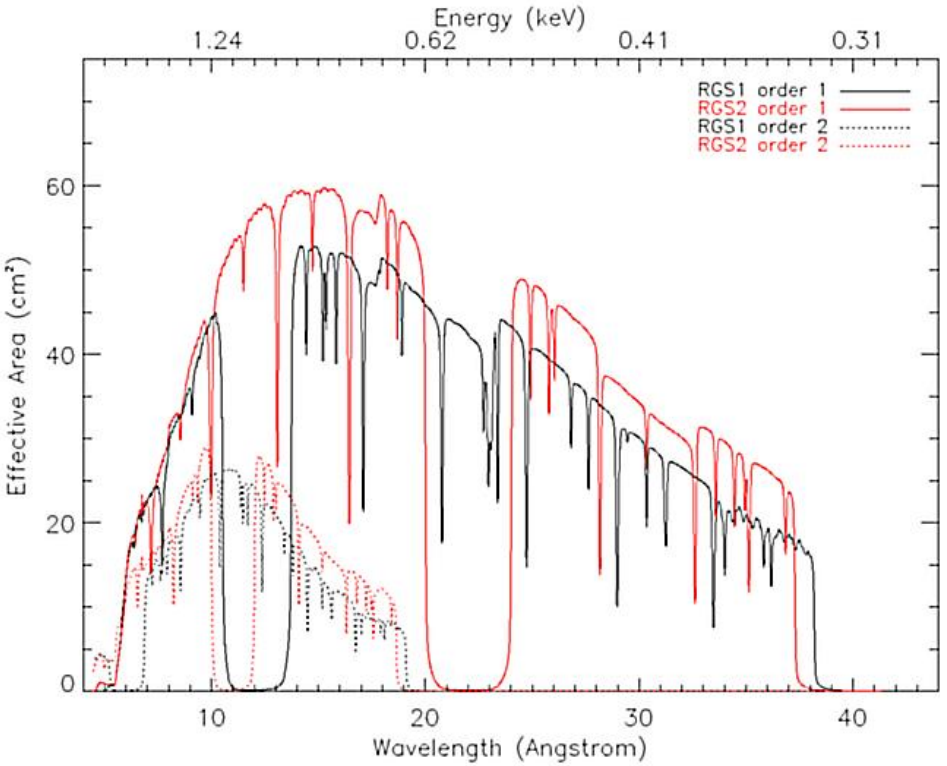
extracted spectrum + response matrix



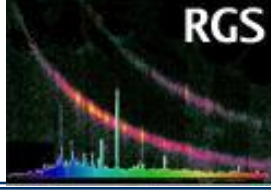
The response matrices

Response files for each source and order are computed with **`rgsrmfgen`**

- Effective area depends on data selections both in PI and XDISP
- Inter-chip gaps
- Bad columns
- Wiggles due to sampling of data selection regions
- Failing CCD chains
- Instrumental edges



Should I reprocess the data?



SAS version of pipeline products

- **PROC0 and PROCDATE**

The calibration index file

- **are there new calibration files?**

Source coordinates and extraction regions

- **are they right?**

Background

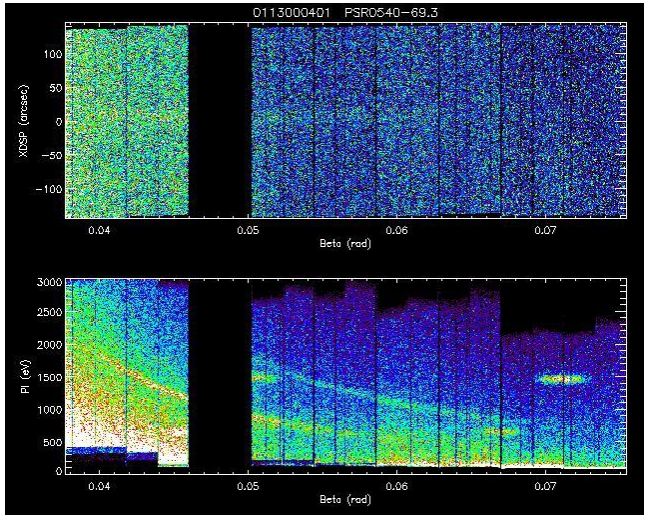
- **was the observation affected by flares?**

Multiple sources and regions

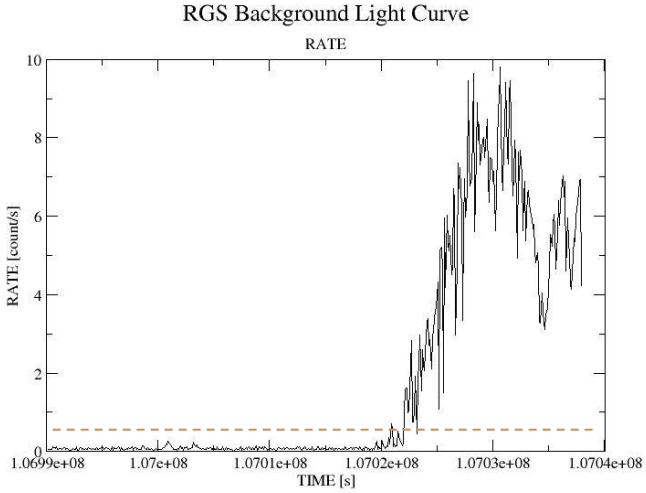
- **are there several sources within the FOV?**

High background (I)

Proton flare during observation ⇒ decrease in signal-to-noise !



Derive the binned (10/100 sec) light curve of the background region of CCD#9 with **`evselect`** or **`xmmselect`** [or use Pipeline file "FBKTSR"]

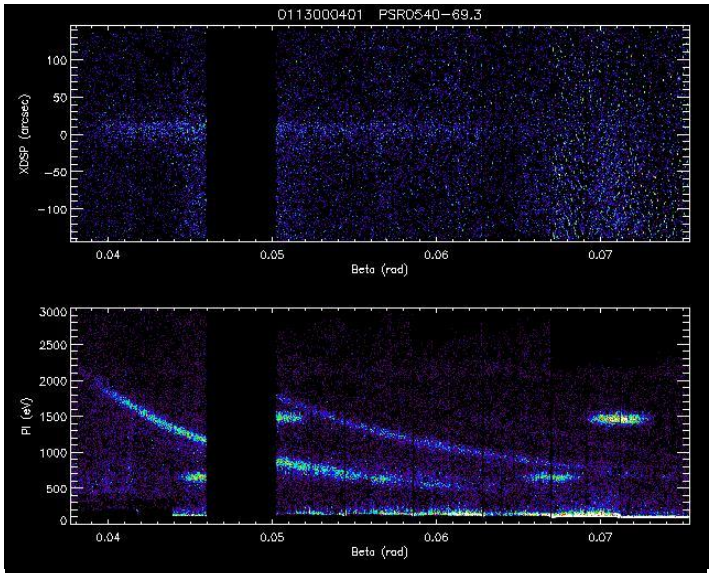


Select periods with (e.g) < 0.5 c/s and generate GTIs with **`tabgtigen`**

High background (II)

re-run `rgsproc` from the 'filter' stage:

```
rgsproc entrystage=filter auxgtitables=mygti.ds
```



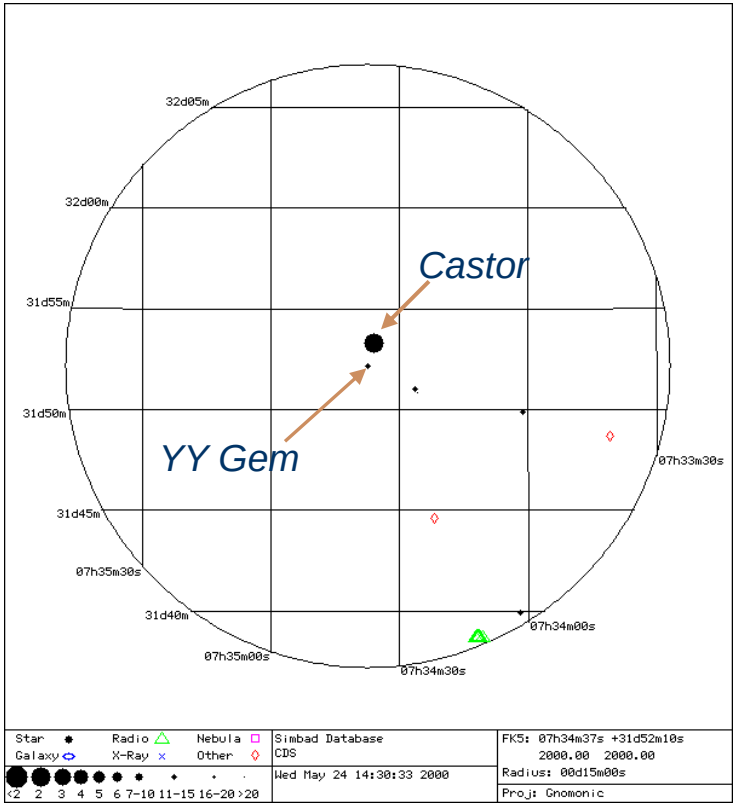
Same observation filtered to

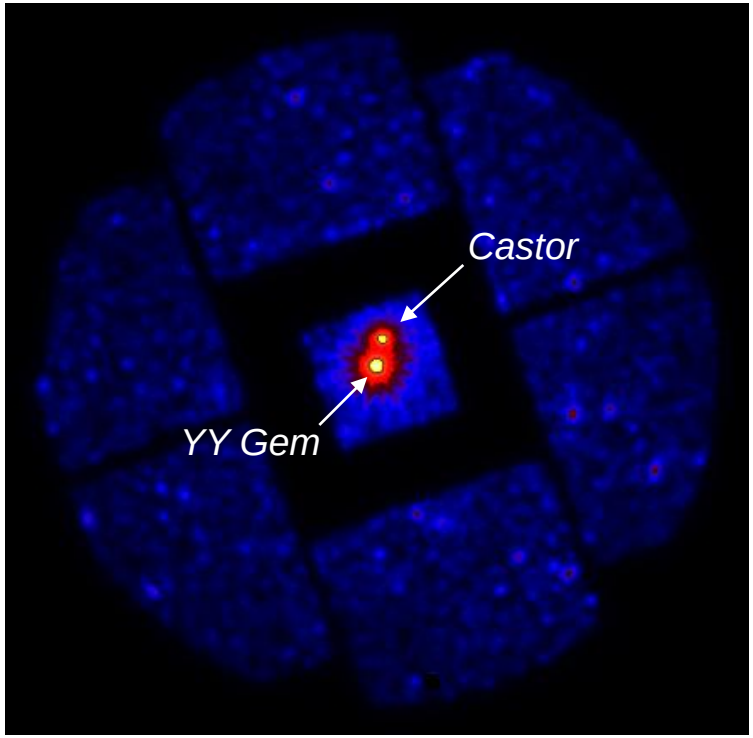
CCD9 < 0.5 counts/sec

Multiples sources and masks (I)

If there are several sources in the FOV (YY Gem/ Castor) ...

Field with two sources separated by 71”
primary source: YY Gem, secondary source: Castor





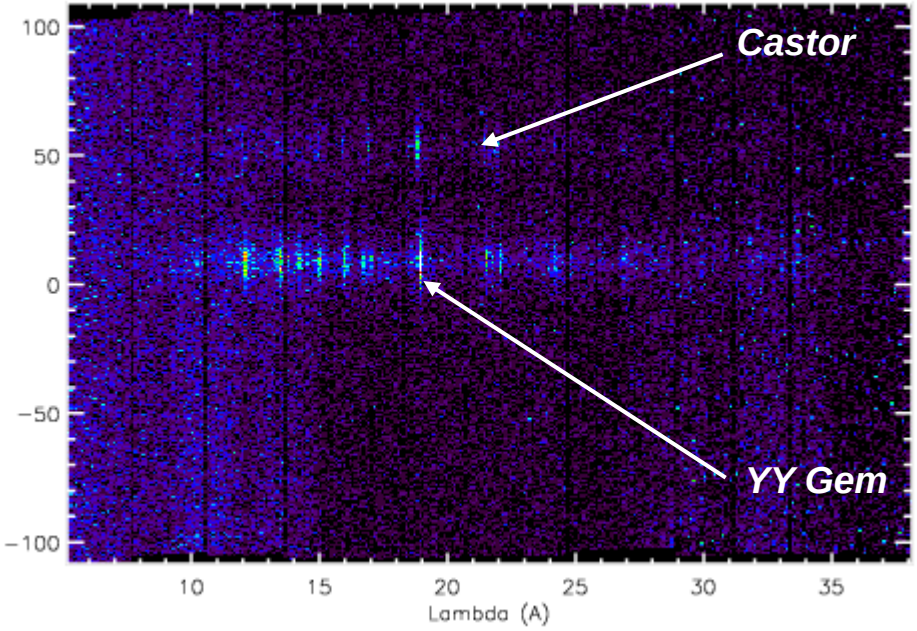
EPIC MOS image

Multiples sources and masks (II)

Use **rgssources** to add the coordinates of Castor to the SRCLIST table

Use **rgsregions** to create new region masks excluding both sources from the background

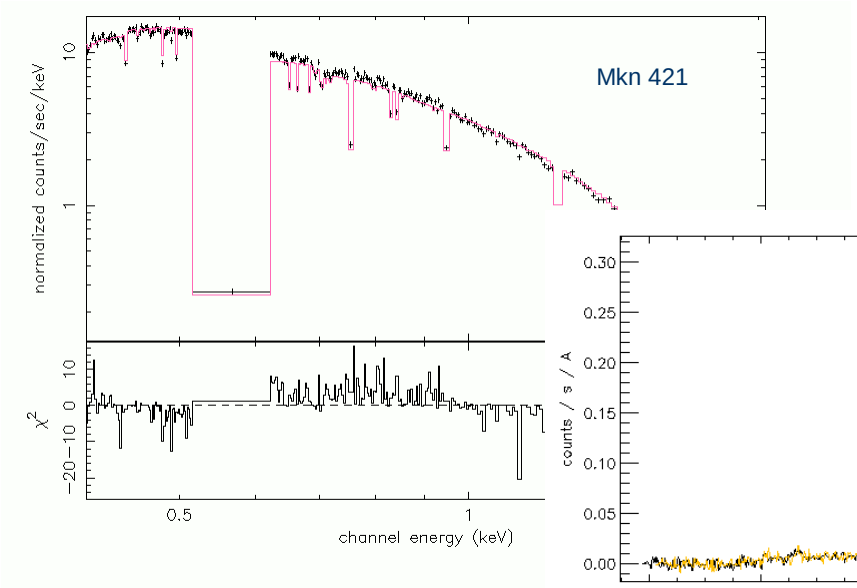
Run **rgsproc entrystage=spectra** to extract the new spectrum

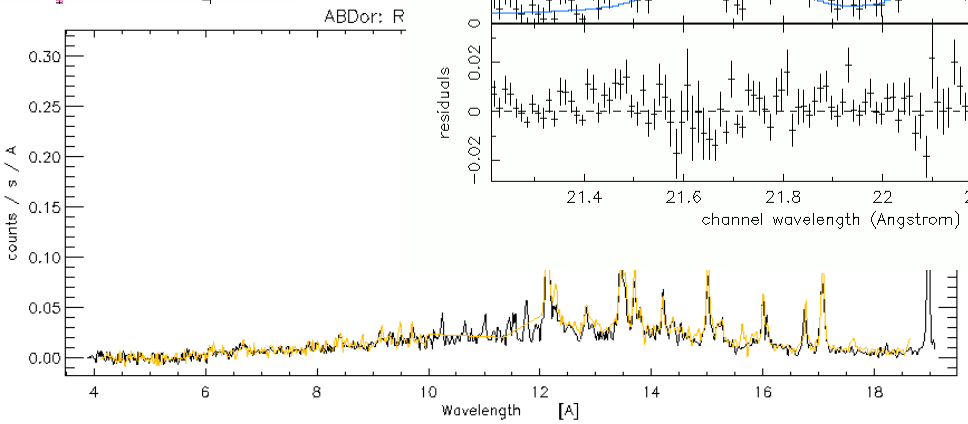


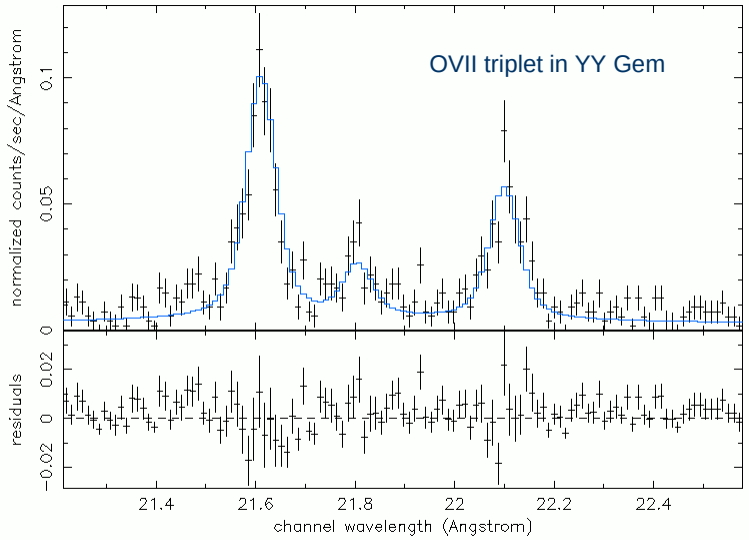
Data analysis

Extracted source + background spectra + response matrix

Work using specific packages, e.g. XSPEC, SPEX, Sherpa, PintOfAle, ISIS..







Summary

Data processing with `rgsproc`

- events
- angles
- filter
- spectra
- fluxing
- lightcurve

Checks

- PPS version and calibration files
- coordinates, sources and regions
- high background, flares

Data analysis

- response matrices and effective area
- fitting (model *vs* observation)